

**LABCAR-RTC V5.4.1**  
**Real-Time Execution Connector**  
Benutzerhandbuch



## Copyright

---

Die Angaben in diesem Schriftstück dürfen nicht ohne gesonderte Mitteilung der ETAS GmbH geändert werden. Desweiteren geht die ETAS GmbH mit diesem Schriftstück keine weiteren Verpflichtungen ein. Die darin dargestellte Software wird auf Basis eines allgemeinen Lizenzvertrages oder einer Einzellizenz geliefert. Benutzung und Vervielfältigung ist nur in Übereinstimmung mit den vertraglichen Abmachungen gestattet.

Unter keinen Umständen darf ein Teil dieser Veröffentlichung in irgendeiner Form ohne schriftliche Genehmigung der ETAS GmbH kopiert, vervielfältigt, in einem Retrievalsystem gespeichert oder in eine andere Sprache übersetzt werden.

© **Copyright 2003 - 2016** ETAS GmbH, Stuttgart

Die verwendeten Bezeichnungen und Namen sind Warenzeichen oder Handelsnamen ihrer entsprechenden Eigentümer.

V5.4.1 R01 DE - 08.2016

---

## Inhalt

1	Der RTIO-Editor	23
1.1	Die Bedienelemente des RTIO-Editors	23
1.1.1	Das Menü des RTIO-Editors	24
1.1.2	Die Symbolleiste	32
1.2	Items	33
1.2.1	Das Kontextmenü der Liste „Items“	33
1.3	Konfigurationsregister	42
1.3.1	Allgemeine Hinweise	43
1.3.2	Standardoptionen im Register „Globals“	45
1.3.3	Standardoptionen im Register „Groups“	47
1.3.4	Standardoptionen im Register „Signals“	48
1.3.5	Standardoptionen im Register „Data“	49
1.3.6	Das Kontextmenü der Register	50
1.3.7	Optionen	52
2	ES1220.1 CAN Board	55
2.1	ES1220-CAN Subsystem	56
2.1.1	Globals (ES1220-CAN Subsystem)	56
2.2	CAN-CTRL Subsystem	57
2.2.1	Globals (CAN-CTRL Subsystem)	57
2.3	FLEX-CAN Subsystem	58
2.4	CONFIG-CTRL Device	58
2.4.1	Groups (CONFIG-CTRL Device)	58
2.4.2	Signals (CONFIG-CTRL Device)	59
2.5	MESSAGE Device	62
2.5.1	Globals (MESSAGE Device)	62
2.5.2	Groups (MESSAGE Device)	63
2.5.3	Signals (MESSAGE Device)	64
2.6	CAN-IO Device	67

2.6.1	Globals (CAN-IO Device) . . . . .	67
2.6.2	Groups (CAN-IO Device) . . . . .	70
2.6.3	Signals (CAN-IO Device) . . . . .	72
3	ES1302.1 A/D Board . . . . .	75
3.1	ES1302-AD Device . . . . .	75
3.1.1	Globals (ES1302-AD Device) . . . . .	75
3.1.2	Groups (ES1302-AD Device) . . . . .	76
3.1.3	Signals (ES1302-AD Device) . . . . .	77
4	ES1321.1 PWM I/O Board . . . . .	79
4.1	ES1321 Subsystem . . . . .	80
4.1.1	Globals (ES1321 Subsystem) . . . . .	80
4.2	ES1321-In Subsystem . . . . .	81
4.2.1	Globals (ES1321-In Subsystem) . . . . .	81
4.3	ES1321-In-HW Device . . . . .	82
4.3.1	Globals (ES1321-In-HW Device) . . . . .	82
4.3.2	Groups (ES1321-In-HW Device) . . . . .	83
4.4	ES1321-In-Meas Device . . . . .	84
4.4.1	Globals (ES1321-In-Meas Device) . . . . .	84
4.4.2	Signals (ES1321-In-Meas Device) . . . . .	84
4.5	Die Messverfahren . . . . .	87
4.5.1	Pulsweitenmessungen . . . . .	88
4.5.2	Additive Pulsweitenmessungen . . . . .	89
4.5.3	Puls- und Flanken <span>z</span> ählung . . . . .	89
4.5.4	Frequenz- und Zykluszeitmessungen . . . . .	91
4.5.5	Tastverhältnismessungen . . . . .	92
4.5.6	Pegelmessungen . . . . .	92
4.5.7	Positionsverfolgung an zwei-phasigen Schrittmotoren . . . . .	93
4.6	ES1321-In-SENT Device . . . . .	94
4.6.1	Globals (ES1321-In-SENT Device) . . . . .	94
4.6.2	Signals (ES1321-In-SENT Device) . . . . .	97
4.6.3	Data (ES1321-In-SENT Device) . . . . .	98
4.7	ES1321-In-ASM Device . . . . .	100
4.7.1	Globals (ES1321-In-ASM Device) . . . . .	100
4.7.2	Signals (ES1321-In-ASM Device) . . . . .	103
4.8	ES1321-Out Subsystem . . . . .	104
4.8.1	Globals (ES1321-Out Subsystem) . . . . .	104
4.9	ES1321-Out-Dig-Pwm Device . . . . .	105
4.9.1	Signals (ES1321-Out-Dig-Pwm Device) . . . . .	105
4.9.2	Data (ES1321-Out-Dig-Pwm Device) . . . . .	107
4.10	ES1321-Out-Multi-Pulse Device . . . . .	108
4.10.1	Globals (ES1321-Out-Multi-Pulse Device) . . . . .	108
4.10.2	Data (ES1321-Out-Multi-Pulse Device) . . . . .	111
4.11	ES1321-Out-SENT Device . . . . .	112
4.11.1	Globals (ES1321-Out-SENT Device) . . . . .	112
4.11.2	Data (ES1321-Out-SENT Device) . . . . .	115
5	ES1330.1 PWM I/O Counter Board . . . . .	119
5.1	ES1330-PWM Subsystem . . . . .	119



5.1.1	Globals (ES1330-PWM Subsystem)	119
5.2	PWM-COUNTER Device	120
5.2.1	Globals (PWM-COUNTER Device)	120
5.2.2	Groups (PWM-COUNTER Device)	122
5.2.3	Signals (PWM-COUNTER Device)	122
6	ES1331.1 Signal Generator Board	125
6.1	ES1331-DSP Subsystem	126
6.1.1	Globals (ES1331-DSP Subsystem)	126
6.2	WheelSnsrSim Device	126
6.2.1	Simulation von Raddrehzahlfühlern	126
6.2.2	Globals (WheelSnsrSim Device)	130
6.2.3	Groups (WheelSnsrSim Device)	131
6.2.4	Signals (WheelSnsrSim Device)	132
7	ES1332.1 Arbitrary Signal Generator Board	139
7.1	ES1332-DSP Subsystem	140
7.1.1	Globals (ES1332-DSP Subsystem)	140
7.2	SigGen Subsystem	141
7.2.1	Globals (SigGen Subsystem)	141
7.3	MasterSglGnrtr Device	143
7.3.1	Globals (MasterSglGnrtr Device)	143
7.3.2	Groups (MasterSglGnrtr Device)	145
7.3.3	Signals (MasterSglGnrtr Device)	146
8	ES1334.1 Measurement Board	149
8.1	Funktionsbeschreibung	150
8.1.1	Grundprinzip der Messwertbestimmung	150
8.1.2	Eingangsbeschaltung	151
8.1.3	Messwernerfassung	152
8.1.4	Timeout-Überwachung	152
8.1.5	Enable-Signale für Klopf-Generatoren	153
8.1.6	Aufbau des ES1334.1-RTIO-Baums	154
8.2	ES1334-VMI Subsystem	155
8.2.1	Globals (ES1334-VMI Subsystem)	155
8.3	Hardwarekonfiguration - ES1334-Hw Device	156
8.3.1	Globals (ES1334-Hw Device)	156
8.3.2	Groups (ES1334-Hw Device)	159
8.3.3	Signals (ES1334-Hw Device)	162
8.4	Messkonfiguration - ES1334-Meas Device	164
8.4.1	Globals (ES1334-Meas Device)	164
8.4.2	Groups (ES1334-Meas Device)	164
8.4.3	Signals (ES1334-Meas Device)	166
8.5	Die Messverfahren	171
8.5.1	Pulsweitenmessungen	171
8.5.2	Additive Pulsweitenmessungen	173
8.5.3	Frequenz- und Zykluszeitmessungen	176
8.5.4	Tastverhältnismessungen	178
8.5.5	Vermessung von Flanken: Winkelstempel	180
8.5.6	Pulszählung	182

8.5.7	Pegelmessung	184
9	ES1334.2 Measurement Board	187
9.1	Funktionsbeschreibung	188
9.1.1	Grundprinzip der Messwertbestimmung	188
9.1.2	Eingangsbeschaltung	189
9.1.3	Messwernerfassung	190
9.1.4	Timeout-Überwachung	190
9.1.5	Enable-Signale für Klopf-Generatoren	191
9.1.6	Aufbau des -RTIO-Baums	192
9.2	ES1334.2-VMI Subsystem	193
9.2.1	Globals (ES1334.2-VMI Subsystem)	193
9.3	ES1334.2-Hw Device	194
9.3.1	Globals (ES1334.2-Hw Device)	194
9.3.2	Groups (ES1334.2-Hw Device)	197
9.3.3	Signals (ES1334.2-Hw Device)	201
9.4	Messkonfiguration - ES1334.2-Meas Device	203
9.4.1	Globals (ES1334.2-Meas Device)	203
9.4.2	Groups (ES1334.2-Meas Device)	203
9.4.3	RTIO-Signale der Signalgruppe „MeasVal“	204
9.4.4	Signals (ES1334.2-Meas Device)	205
9.5	Messverfahren	213
9.5.1	Pulsweitenmessungen	213
9.5.2	Additive Pulsweitenmessungen	216
9.5.3	Pulsweitenmessungen mit Enable- oder Validierungssignal	219
9.5.4	Frequenz- und Zykluszeitmessungen	223
9.5.5	Tastverhältnismessungen	225
9.5.6	Relativmessungen zwischen Hardwarekanälen	226
9.5.7	Vermessung von Flanken: Winkelstempel	231
9.5.8	Vermessung von Flanken: Zeitstempel	233
9.5.9	Pulszählung	234
9.5.10	Pegelmessung	236
9.5.11	Positionsverfolgung an zwei-phasigen Schrittmotoren	237
10	ESX335.1 Arbitrary Signal Generator Board	241
10.1	Funktionsbeschreibung	242
10.1.1	Signalgeneratoren	242
10.1.2	MSA-Sensor	242
10.1.3	RPM-Generator	242
10.1.4	Winkelauflösung	242
10.1.5	Wellenformspeicher für Signalgeneratoren	243
10.1.6	Direkter Wellenformzugriff aus der Simulation	243
10.1.7	Ausgangsmultiplexer	244
10.1.8	Klopfsignalgenerator	244
10.1.9	Zündaussetzsteuerung	245
10.1.10	Sequenztabellen	245
10.1.11	Aufbau des ESX335.1 RTIO-Baums	245
10.2	ESX335 Subsystem	246
10.2.1	Globals (ESX335 Subsystem)	246

10.3	ESX335-Powertrain Subsystem	247
10.3.1	Globals (ESX335-Powertrain Subsystem)	247
10.4	ESX335-OutputMux Device	249
10.4.1	Globals (ESX335-OutputMux Device)	249
10.4.2	Groups (ESX335-OutputMux Device)	249
10.4.3	Signals (ESX335-OutputMux Device)	249
10.4.4	Data (ESX335-OutputMux Device)	250
10.5	ESX335-Sig Device	251
10.5.1	Globals (ESX335-Sig Device)	251
10.5.2	Groups (ESX335-Sig Device)	252
10.5.3	Data (ESX335-Sig Device)	252
10.6	ESX335-Rpm Device	256
10.6.1	Globals (ESX335-Rpm Device)	256
10.6.2	Groups (ESX335-Rpm Device)	257
10.6.3	Data (ESX335-Rpm Device)	258
10.7	ESX335-Misfire Device	259
10.7.1	Globals (ESX335-Misfire Device)	259
10.7.2	Groups (ESX335-Misfire Device)	261
10.7.3	Signals (ESX335-Misfire Device)	261
10.7.4	Data (ESX335-Misfire Device)	262
10.8	ESX335-Knock Device	264
10.8.1	Globals (ESX335-Knock Device)	264
10.8.2	Groups (ESX335-Knock Device)	266
10.8.3	Signals (ESX335-Knock Device)	267
10.8.4	Data (ESX335-Knock Device)	268
10.9	Direkter Wellenformzugriff: ESX335-OnlineWaveformAccess Device	271
10.9.1	Data (ESX335-OnlineWaveformAccess Device)	271
10.10	ESX335-MSA-Sensor Device	272
10.10.1	Grundsätzliche Funktionsweise	272
10.10.2	Zahnmittenberechnung	273
10.10.3	Globals (ESX335-MSA-Sensor Device)	274
10.10.4	Groups (ESX335-MSA-Sensor Device)	276
10.10.5	Data (ESX335-MSA-Sensor Device)	277
10.10.6	Fehlerbehandlung	278
10.11	Externe Triggerung der Signalgeneratoren: ESX335-Trigger-Inputs Device	279
10.11.1	Data (ESX335-Trigger-Inputs Device)	280
11	ES1336.1 Angle Synchronous Measurement Board	281
11.1	Grundprinzip der Messwertbestimmung	282
11.1.1	Signalerfassung	282
11.1.2	Komparatorstufen	283
11.1.3	Pulsweiten- und Winkelmessungen	283
11.1.4	Pulsintegration	284
11.1.5	Vermessung von Anstiegs- und Abfallzeiten	285
11.1.6	Spitzenwertvermessung	285
11.1.7	Asynchrone Messungen	285
11.1.8	Winkelsynchrone Messungen	286
11.1.9	Timeouts	286
11.2	ES1336 Subsystem	286

11.2.1	Globals (ES1336 Subsystem) . . . . .	287
11.3	ES1336-Rpm Device . . . . .	288
11.3.1	Globals (ES1336-Rpm Device) . . . . .	288
11.3.2	Groups (ES1336-Rpm Device) . . . . .	290
11.3.3	Signals (ES1336-Rpm Device) . . . . .	292
11.4	ES1336-Hw Device . . . . .	292
11.4.1	Globals (ES1336-Hw Device) . . . . .	292
11.4.2	Groups (ES1336-Hw Device) . . . . .	295
11.4.3	Signals (ES1336-Hw Device) . . . . .	296
11.5	ES1336-Meas Device . . . . .	299
11.5.1	Globals (ES1336-Meas Device) . . . . .	299
11.5.2	Groups (ES1336-Meas Device) . . . . .	299
11.5.3	Signals (ES1336-Meas Device) . . . . .	301
11.6	Die Messverfahren . . . . .	305
11.6.1	Asynchrone Messungen . . . . .	305
11.6.2	Winkelsynchrone Messungen - Pulsweitenmessungen . . . . .	306
11.6.3	Pulsweitenmessungen mit Enable- oder Validierungssignal . . . . .	307
11.6.4	Additive Pulsweitenmessungen . . . . .	310
11.6.5	Vermessung von Flanken: Winkelstempel . . . . .	312
11.6.6	Vermessung von Flanken: Zeitstempel . . . . .	313
11.6.7	Vermessung von Flanken: Anstiegs- und Abfallzeiten . . . . .	314
11.6.8	Spitzenwertmessung . . . . .	316
11.6.9	Pulsintegration . . . . .	316
11.6.10	Pulszählung . . . . .	317
12	ES1337.1 Wheelspeed Sensor Simulation Board . . . . .	319
12.1	ES1337 Subsystem . . . . .	320
12.1.1	Globals (ES1337 Subsystem) . . . . .	320
12.1.2	Globals (ES1337-Wheelsnsrsim Subsystem) . . . . .	320
12.2	ES1337-Wheelsnsrsim-DA Device . . . . .	322
12.2.1	Globals (ES1337-Wheelsnsrsim-DA Device) . . . . .	322
12.2.2	Groups (ES1337-Wheelsnsrsim-DA Device) . . . . .	323
12.2.3	Data (ES1337-Wheelsnsrsim-DA Device) . . . . .	323
12.3	Radsensorsimulation – Gemeinsame Einstellungen . . . . .	324
12.3.1	Gemeinsame Signale aller Sensoren . . . . .	325
12.3.2	Globals (ES1337-Wheelsnsrsim-DF6 Device) . . . . .	327
12.3.3	Groups (ES1337-Wheelsnsrsim-DF6 Device) . . . . .	328
12.3.4	Data (ES1337-Wheelsnsrsim-DF6 Device) . . . . .	328
12.3.5	Globals (ES1337-Wheelsnsrsim-DF10-DF11s Device) . . . . .	330
12.3.6	Groups (ES1337-Wheelsnsrsim-DF10-DF11s Device) . . . . .	331
12.3.7	Data (ES1337-Wheelsnsrsim-DF10-DF11s Device) . . . . .	332
12.3.8	Globals (ES1337-Wheelsnsrsim-DF10-RotDir Device) . . . . .	334
12.3.9	Groups (ES1337-Wheelsnsrsim-DF10-RotDir Device) . . . . .	336
12.3.10	Data (ES1337-Wheelsnsrsim-DF10-RotDir Device) . . . . .	337
12.3.11	Globals (ES1337-Wheelsnsrsim-DF11i Device) . . . . .	339
12.3.12	Groups (ES1337-Wheelsnsrsim-DF11i Device) . . . . .	341
12.3.13	Data (ES1337-Wheelsnsrsim-DF11i Device) . . . . .	342
12.3.14	Globals (ES1337-Wheelsnsrsim-VDA Device) . . . . .	344
12.3.15	Groups (ES1337-Wheelsnsrsim-VDA Device) . . . . .	346



12.3.16	Data (ES1337-Wheelsnsrsim-VDA Device) . . . . .	347
13	ES1385 Resistor Cascade Board . . . . .	351
13.1	ES1385 Device . . . . .	351
13.1.1	Globals (ES1385 Device) . . . . .	351
13.1.2	Groups (ES1385 Device) . . . . .	354
13.1.3	Signals (ES1385 Device) . . . . .	356
14	ES1391.1 Power Supply Controller Board . . . . .	357
14.1	ES1391-PWR Subsystem . . . . .	358
14.1.1	Globals (ES1391-PWR Subsystem) . . . . .	358
14.2	ES1391-PwrCtrl Device . . . . .	361
14.2.1	Globals (ES1391-PwrCtrl Device) . . . . .	361
14.2.2	Groups (ES1391-PwrCtrl Device) . . . . .	364
14.2.3	Signals (ES1391-PwrCtrl Device) . . . . .	368
14.3	ES1391-SwCtrl Device . . . . .	370
14.3.1	Globals (ES1391-SwCtrl Device) . . . . .	370
14.3.2	Groups (ES1391-SwCtrl Device) . . . . .	371
14.3.3	Signals (ES1391-SwCtrl Device) . . . . .	374
15	ES1650.1 Piggyback Carrier Board . . . . .	377
15.1	ES1650-CB Subsystem . . . . .	377
15.1.1	Globals (ES1650-CB Subsystem) . . . . .	377
15.2	PB1650ADC1 . . . . .	378
15.2.1	Globals (PB1650ADC1 Device) . . . . .	378
15.2.2	Groups (PB1650ADC1 Device) . . . . .	378
15.2.3	Signals (PB1650ADC1 Device) . . . . .	379
15.2.4	Datentypen und Wertebereiche der Signale . . . . .	380
15.3	PB1650DAC1 . . . . .	381
15.3.1	Globals (PB1650DAC1 Device) . . . . .	381
15.3.2	Groups (PB1650DAC1 Device) . . . . .	381
15.3.3	Signals (PB1650DAC1 Device) . . . . .	382
15.3.4	Datentypen und Wertebereiche der Signale . . . . .	383
15.4	PB1650DIO1 . . . . .	384
15.4.1	Globals (PB1650DIO1 Device) . . . . .	384
15.4.2	Groups (PB1650DIO1 Device) . . . . .	384
15.4.3	Signals (PB1650DIO1 Device) . . . . .	385
15.4.4	Datentypen und Wertebereiche der Signale . . . . .	385
15.5	PB1650DIO2 . . . . .	386
15.5.1	Globals (PB1650DIO2 Device) . . . . .	386
15.5.2	Groups (PB1650DIO2 Device) . . . . .	386
15.5.3	Signals (PB1650DIO2 Device) . . . . .	387
15.5.4	Datentypen und Wertebereiche der Signale . . . . .	387
15.6	PB1650REL1 . . . . .	388
15.6.1	Globals (PB1650REL1 Device) . . . . .	388
15.6.2	Groups (PB1650REL1 Device) . . . . .	389
15.6.3	Signals (PB1650REL1 Device) . . . . .	389
15.6.4	Datentypen und Wertebereiche der Signale . . . . .	389

16	ES1651.1 Carrier Board	391
16.1	ES1651-CB Subsystem	392
16.1.1	Globals (ES1651-CB Subsystem)	392
16.1.2	Versteckte Optionsfelder	392
16.2	ES1651-CTRL Device	394
16.2.1	Globals (ES1651-CTRL Device)	394
16.2.2	Versteckte Optionsfelder	396
16.2.3	Groups (ES1651-CTRL Device)	398
16.2.4	Signals (ES1651-CTRL Device)	400
16.2.5	Data (ES1651-CTRL Device)	401
16.3	ES1651-CAN Subsystem	402
16.3.1	Globals (ES1651-CAN Subsystem)	402
16.4	CAN-CTRL Subsystem	403
16.4.1	Globals (CAN-CTRL Subsystem)	403
16.4.2	Globals (CAN-IO Device)	407
16.4.3	Groups (CAN-IO Device)	410
16.4.4	Die CAN-Botschaften als Signalgruppen	410
16.4.5	Signals (CAN-IO Device)	412
16.4.6	Data (CAN-IO Device)	415
16.4.7	Globals (CAN-ESTAT Device)	416
16.4.8	Groups (CAN-ESTAT Device)	416
16.4.9	Die Signale der Signalgruppe „EStatIn“	417
16.4.10	Signals (CAN-ESTAT Device)	419
16.4.11	Data (CAN-ESTAT Device)	420
16.5	PB1651ADC1	421
16.5.1	Globals (PB1651ADC1 Device)	422
16.5.2	Groups (PB1651ADC1 Device)	426
16.5.3	Signals (PB1651ADC1 Device)	428
16.5.4	Data (PB1651ADC1 Device)	431
16.6	PB1651PWM1	432
16.6.1	Globals (PB1651PWM1 Subsystem)	433
16.6.2	Das PB1651PWM1-In-Hw Device	438
16.6.3	Globals (PB1651PWM1-In-Hw Device)	438
16.6.4	Groups (PB1651PWM1-In-Hw Device)	439
16.6.5	Signals (PB1651PWM1-In-Hw Device)	440
16.6.6	Das PB1651PWM1-In-Meas Device	440
16.6.7	Globals (PB1651PWM1-In-Meas Device)	441
16.6.8	Groups (PB1651PWM1-In-Meas Device)	441
16.6.9	Die RTIO-Signale der Signalgruppe „MeasVal“	442
16.6.10	Signals (PB1651PWM1-In-Meas Device)	444
16.6.11	Die Messverfahren	447
16.6.12	Pulsweitenmessungen	449
16.6.13	Additive Pulsweitenmessungen	451
16.6.14	Puls- und Flankenählung	452
16.6.15	Frequenz- und Zykluszeitmessungen	455
16.6.16	Tastverhältnismessungen	457
16.6.17	Pegelmessungen	458
16.6.18	Das PB1651PWM1-Out Device	459

16.6.19	Globals (PB1651PWM1-Out Device) . . . . .	459
16.6.20	Groups (PB1651PWM1-Out Device) . . . . .	459
16.6.21	Signals (PB1651PWM1-Out Device) . . . . .	464
17	ES4315-VXI - VME64x/VXI Adapter . . . . .	471
17.1	ES4315-VXI Subsystem . . . . .	471
17.1.1	Globals (ES4315-VXI Subsystem) . . . . .	471
17.2	ES4315-CTRL Device . . . . .	472
17.2.1	Globals (ES4315-CTRL Device) . . . . .	472
17.2.2	Groups (ES4315-CTRL Device) . . . . .	474
17.2.3	Signals (ES4315-CTRL Device) . . . . .	475
17.2.4	Datentypen und Wertebereiche . . . . .	475
17.3	ES4320-XSG - PWM und arbiträre Signalgenerierung . . . . .	476
17.3.1	Globals (ES4320-XSG Subsystem) . . . . .	477
17.3.2	ES4320-CTRL Device . . . . .	477
17.3.3	Globals (ES4320-CTRL Device) . . . . .	478
17.3.4	Groups (ES4320-CTRL Device) . . . . .	480
17.3.5	Signals (ES4320-CTRL Device) . . . . .	481
17.3.6	Datentypen und Wertebereiche . . . . .	482
17.3.7	ES4320-RPM Device . . . . .	482
17.3.8	Globals (ES4320-RPM Device) . . . . .	483
17.3.9	Groups (ES4320-RPM Device) . . . . .	484
17.3.10	Signals (ES4320-RPM Device) . . . . .	487
17.3.11	Datentypen und Wertebereiche . . . . .	489
17.3.12	ES4320-SIG Subsystem . . . . .	489
17.3.13	ES4320-SIG-CTRL Device . . . . .	489
17.3.14	Groups (ES4320-SIG-CTRL Device) . . . . .	489
17.3.15	Signals (ES4320-SIG-CTRL Device) . . . . .	491
17.3.16	Datentypen und Wertebereiche . . . . .	491
17.3.17	Globals (ES4320-SIG-PWM Device) . . . . .	492
17.3.18	Groups (ES4320-SIG-PWM Device) . . . . .	493
17.3.19	Signals (ES4320-SIG-PWM Device) . . . . .	494
17.3.20	Datentypen und Wertebereiche . . . . .	495
17.3.21	Globals (ES4320-SIG-ARB Device) . . . . .	495
17.3.22	Groups (ES4320-SIG-ARB Device) . . . . .	500
17.3.23	Signals (ES4320-SIG-ARB Device) . . . . .	501
17.3.24	Datentypen und Wertebereiche . . . . .	502
17.3.25	ES4320-SIG-KNOCK Device . . . . .	503
17.3.26	Globals (ES4320-SIG-KNOCK Device) . . . . .	504
17.3.27	Groups (ES4320-SIG-KNOCK Device) . . . . .	508
17.3.28	Signals (ES4320-SIG-KNOCK Device) . . . . .	509
17.3.29	ES4320-KNOCK-Ctrl Device . . . . .	515
17.3.30	Globals (ES4320-KNOCK-Ctrl Device) . . . . .	515
17.3.31	Groups (ES4320-KNOCK-Ctrl Device) . . . . .	516
17.3.32	Signals (ES4320-KNOCK-Ctrl Device) . . . . .	517
17.3.33	ES4320-Misfire Device . . . . .	518
17.3.34	Globals (ES4320-Misfire Device) . . . . .	519
17.3.35	Groups (ES4320-Misfire Device) . . . . .	522
17.3.36	Signals (ES4320-Misfire Device) . . . . .	522

17.4	ES4330-XMI - VXI Signal Measurement Board	526
17.4.1	Funktionsbeschreibung	526
17.4.2	Aufbau des ES4330 RTIO Baums	531
17.4.3	Globale Einstellungen - ES4330-XMI Subsystem	532
17.4.4	Globals (ES4330-XMI Subsystem)	533
17.4.5	Hardwarekonfiguration - ES4330-Hw Device	534
17.4.6	Globals (ES4330-Hw Device)	534
17.4.7	Groups (ES4330-Hw Device)	536
17.4.8	RTIO-Signale der Signalgruppe „Control“	537
17.4.9	RTIO-Signale der Signalgruppe „Level“	538
17.4.10	Signals (ES4330-Hw Device)	539
17.4.11	Messkonfiguration - ES4330-Meas Device	542
17.4.12	Globals (ES4330-Meas Device)	542
17.4.13	Groups (ES4330-Meas Device)	542
17.4.14	RTIO-Signale der Signalgruppe „MeasVal“	543
17.4.15	Signals (ES4330-Meas Device)	545
17.4.16	Messverfahren - Pulsweitenmessungen	551
17.4.17	Messverfahren - Additive Pulsweitenmessungen	554
17.4.18	Messverfahren - Pulsweitenmessungen mit Enable- oder Validierungssignal	557
17.4.19	Messverfahren - Frequenz- und Zykluszeitmessungen	561
17.4.20	Messverfahren - Tastverhältnismessungen	563
17.4.21	Messverfahren - Relativmessungen zwischen Hardwarekanälen	564
17.4.22	Messverfahren - Vermessung von Flanken: Winkelstempel	569
17.4.23	Messverfahren - Vermessung von Flanken: Zeitstempel	571
17.4.24	Messverfahren - Pulszählung	572
17.4.25	Messverfahren - Pegelmessung	574
17.4.26	Messverfahren - Positionsverfolgung an zwei-phasigen Schrittmotoren	575
17.5	ES4350 Carrier Board	578
17.5.1	Aufbau des ES4350-RTIO-Baums	578
17.5.2	Globals (ES4350-CB Subsystem)	579
17.5.3	Das ES4350-CTRL Device	579
17.5.4	Globals (ES4350-CTRL Device)	582
17.5.5	Groups (ES4350-CTRL Device)	583
17.5.6	RTIO-Signale der Signalgruppe „SyncLvlCtrl“	584
17.5.7	Signals (ES4350-CTRL Device)	585
17.6	PB4350DAC1 I/O-Modul	587
17.6.1	Aufbau des PB4350DAC1-RTIO-Baums	587
17.6.2	Globals (PB4350DAC1 Device)	588
17.6.3	Groups (PB4350DAC Device)	592
17.6.4	RTIO-Signale der Signalgruppe „AnaOut“	593
17.6.5	RTIO-Signale der Signalgruppe „CutOff“	594
17.6.6	Signals (PB4350DAC Device)	594
18	ES4408-System	597
18.1	ES4408-Load-Chassis Subsystem	598
18.1.1	Globals (ES4408-Load-Chassis Subsystem)	598
18.2	ES4408-Ctrl Device	602



18.2.1	Globals (ES4408-Ctrl Device) . . . . .	602
18.2.2	Data (ES4408-Ctrl Device) . . . . .	603
18.3	ES4434-Conf-Load Device . . . . .	604
18.3.1	Globals (ES4434-Conf-Load Device) . . . . .	604
18.4	ES4435-Current-Sources Device . . . . .	605
18.4.1	Globals (ES4435-Current-Sources Device) . . . . .	605
18.4.2	Signals (ES4435-Current-Sources Device) . . . . .	606
18.4.3	Data (ES4435-Current-Sources Device) . . . . .	607
18.5	ES4450-RB-CR-Load Device . . . . .	608
18.5.1	Globals (ES4450-RB-CR-Load Device) . . . . .	608
18.5.2	Signals (ES4450-RB-CR-Load Device) . . . . .	609
18.5.3	Data (ES4450-RB-CR-Load Device) . . . . .	610
18.6	ES4456-RB-Piezo-Load Device . . . . .	611
18.6.1	Globals (ES4456-RB-Piezo-Load Device) . . . . .	611
18.6.2	Signals (ES4456-RB-Piezo-Load Device) . . . . .	612
18.6.3	Data (ES4456-RB-Piezo-Load Device) . . . . .	612
18.7	ES44XX-Variable-Load . . . . .	614
18.7.1	Globals (ES44XX-Variable-Load) . . . . .	615
18.7.2	Groups (ES44XX-Variable-Load) . . . . .	616
18.7.3	Signals (ES44XX-Variable-Load) . . . . .	617
18.7.4	Data (ES44XX-Variable-Load Device) . . . . .	619
18.8	Injektorsimulation mit ES4452.1 (Benziner) und ES4457.1 (Diesel) . . . . .	622
18.8.1	HDEV5 Injektorsimulation (CVO) mit ES4452.1 . . . . .	622
18.8.2	CR12-x Injektorsimulation (VCC/VCA) mit ES4457.1 . . . . .	623
18.8.3	Passive Injektorsimulation . . . . .	624
19	ES5300.1 Housing . . . . .	627
19.1	ES5300-Chassis System . . . . .	627
19.1.1	Globals (ES5300-Chassis System) . . . . .	628
19.2	ES5300-Ctrl Subsystem . . . . .	628
19.3	ES5300-BattNode-Ctrl Device . . . . .	629
19.3.1	Globals (ES5300-BattNode-Ctrl) . . . . .	629
19.3.2	Groups (ES5300-Batt-Node) . . . . .	629
19.3.3	Data (ES5300-Batt-Node) . . . . .	630
20	ES5321.1 PWM I/O Board . . . . .	631
20.1	ES5321 Subsystem . . . . .	633
20.1.1	Globals (ES5321 Subsystem) . . . . .	633
20.2	ES5321-RPM Device . . . . .	634
20.2.1	Globals (ES5321-RPM) . . . . .	634
20.2.2	Groups (ES5321-RPM) . . . . .	635
20.2.3	Signals (ES5321-RPM) . . . . .	635
20.3	ES5321-In Subsystem . . . . .	638
20.3.1	Globals (ES5321-In Subsystem) . . . . .	638
20.4	ES5321-In-HW Device . . . . .	638
20.4.1	Globals (ES5321-In-HW Device) . . . . .	638
20.4.2	Groups (ES5321-In-HW Device) . . . . .	639
20.4.3	Signals (ES5321-In-HW Device) . . . . .	639
20.5	ES5321-In-Meas Device . . . . .	641

20.5.1	Globals (ES5321-In-Meas Device) . . . . .	641
20.5.2	Signals (ES5321-In-Meas Device) . . . . .	641
20.6	Die Messverfahren . . . . .	644
20.6.1	Pulsweitenmessungen . . . . .	645
20.6.2	Additive Pulsweitenmessungen . . . . .	646
20.6.3	Frequenz- und Zykluszeitmessungen . . . . .	647
20.6.4	Tastverhältnismessungen . . . . .	647
20.6.5	Pegelmessungen . . . . .	648
20.7	ES5321-In-SENT Device . . . . .	648
20.7.1	Globals (ES5321-In-SENT Device) . . . . .	648
20.7.2	Groups (ES5321-In-SENT Device) . . . . .	650
20.7.3	Signals (ES5321-In-SENT Device) . . . . .	651
20.8	ES5321-Out Subsystem . . . . .	653
20.8.1	Globals (ES5321-Out Subsystem) . . . . .	653
20.9	ES5321-Out-Dig-Pwm Device . . . . .	654
20.9.1	Globals (ES5321-Out-Dig-Pwm Device) . . . . .	654
20.9.2	Signals (ES5321-Out-Dig-Pwm Device) . . . . .	654
20.10	ES5321-Out-Multi-Pulse Device . . . . .	658
20.10.1	Globals (ES5321-Out-Multi-Pulse Device) . . . . .	658
20.10.2	Signals (ES5321-Out-Multi-Pulse Device) . . . . .	660
20.11	ES5321-Out-SENT Device . . . . .	661
20.11.1	Globals (ES5321-Out-SENT Device) . . . . .	661
20.11.2	Signals (ES5321-Out-SENT Device) . . . . .	664
21	ES5338.1 Carrier Board for Wheel Speed Sensor Simulation . . . . .	669
21.1	ES5338 Subsystem . . . . .	670
21.1.1	Globals (ES5338 Subsystem) . . . . .	670
21.2	ES5338-Current-Out Device . . . . .	671
21.2.1	Globals (ES5338-Current-Out Device) . . . . .	671
21.2.2	Signals (ES5338-Current-Out Device) . . . . .	672
21.3	Radsensorsimulation – Gemeinsame Einstellungen . . . . .	673
21.3.1	Gemeinsame Signale aller Sensoren . . . . .	673
21.3.2	Globals (ES5338-DF10-DF11s Device) . . . . .	675
21.3.3	Groups (ES5338-DF10-DF11s Device) . . . . .	676
21.3.4	Signals (ES5338-DF10-DF11s Device) . . . . .	677
21.3.5	Globals (ES5338-DF11i Device) . . . . .	679
21.3.6	Groups (ES5338-DF11i Device) . . . . .	681
21.3.7	Signals (ES5338-DF11i Device) . . . . .	682
21.3.8	Globals (ES5338-VDA Device) . . . . .	684
21.3.9	Groups (ES5338-VDA Device) . . . . .	686
21.3.10	Signals (ES5338-VDA Device) . . . . .	687
21.4	ES5338-PSI5 Device . . . . .	690
21.4.1	Globals (ES5338-PSI5 Device) . . . . .	690
21.4.2	Signals (ES5338-PSI5 Device) . . . . .	695
22	ES5340.1/2 Electric Drive Simulation Board . . . . .	697
22.1	ES5340-Master . . . . .	699
22.1.1	Globals (ES5340-Master Subsystem) . . . . .	699
22.2	ES5340-RPM – Drehzahleinheit . . . . .	700

22.2.1	ES5340-RPM – Drehzahleinheit . . . . .	704
22.2.2	Globals (ES5340-RPM) . . . . .	705
22.2.3	Groups (ES5340-RPM) . . . . .	707
22.2.4	Data (ES5340-RPM) . . . . .	707
22.3	ES5340-Analog-In – Analoge Eingänge . . . . .	710
22.3.1	ES5340-Analog-In – Analoge Eingänge . . . . .	711
22.3.2	Signals (ES5340-Analog-In) . . . . .	711
22.3.3	Data (ES5340-Analog-In) . . . . .	712
22.4	ES5340-Digital-In – Digitale Eingänge . . . . .	713
22.4.1	ES5340-Dig-In-HW – Konfiguration der digitalen Eingänge . . . . .	713
22.4.2	Globals (ES5340-Dig-In-Hw) . . . . .	713
22.4.3	ES5340-Dig-In-Meas – Nicht synchronisierte Zeit-/Frequenzmessungen 714	
22.4.4	Die Messverfahren . . . . .	715
22.4.5	Groups (ES5340-Dig-In-Meas) . . . . .	718
22.4.6	Signals (ES5340-Dig-In-Meas) . . . . .	718
22.4.7	Data (ES5340-Dig-In-Meas Device) . . . . .	720
22.4.8	ES5340-Dig-In-Inverter-Meas – Vermessung der Inverter-Ansteuersig- nale . . . . .	720
22.4.9	Globals (ES5340-Dig-In-Inverter-Meas) . . . . .	721
22.4.10	Groups (ES5340-Dig-In-Inverter-Meas) . . . . .	723
22.4.11	Signals (ES5340-Dig-In-Inverter-Meas) . . . . .	724
22.4.12	Data (ES5340-Dig-In-Inverter-Meas) . . . . .	725
22.5	ES5340-Digital-Out – Digitale Ausgänge . . . . .	726
22.5.1	ES5340-Digital-Direct-Out . . . . .	726
22.5.2	Globals (ES5340-Digital-Direct-Out) . . . . .	727
22.5.3	Groups (ES5340-Digital-Direct-Out) . . . . .	727
22.5.4	Signals (ES5340-Digital-Direct-Out) . . . . .	727
22.5.5	Data (ES5340-Digital-Direct-Out) . . . . .	728
22.5.6	ES5340-Digital-Position-Sensor . . . . .	729
22.5.7	Globals (ES5340-Digital-Position-Sensor) . . . . .	730
22.5.8	Groups (ES5340-Digital-Position-Sensor) . . . . .	731
22.5.9	Signals (ES5340-Digital-Position-Sensor) . . . . .	731
22.5.10	Data (ES5340-Digital-Position-Sensor) . . . . .	732
22.5.11	ES5340-Digital-Arbitrary . . . . .	733
22.5.12	Globals (ES5340-Digital-Arbitrary) . . . . .	733
22.5.13	Groups (ES5340-Digital-Arbitrary) . . . . .	735
22.5.14	Signals (ES5340-Digital-Arbitrary) . . . . .	735
22.5.15	Data (ES5340-Digital-Arbitrary) . . . . .	735
22.5.16	ES5340-PWM-Output . . . . .	736
22.5.17	Globals (ES5340-PWM-Output) . . . . .	736
22.5.18	Groups (ES5340-PWM-Output) . . . . .	736
22.5.19	Signals (ES5340-PWM-Output) . . . . .	737
22.5.20	Data (ES5340-PWM-Output) . . . . .	737
22.6	ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer . . . . .	738
22.6.1	Signals (ES5340-Digital-Out-Mux) . . . . .	738
22.6.2	Data (ES5340-Digital-Out-Mux) . . . . .	740
22.7	ES5340-Analog-Out – Analoge Ausgänge . . . . .	741

22.7.1	ES5340-Analog-Direct-Out . . . . .	742
22.7.2	Globals (ES5340-Analog-Direct-Out) . . . . .	742
22.7.3	Groups (ES5340-Analog-Direct-Out) . . . . .	742
22.7.4	Signals (ES5340-Analog-Direct-Out) . . . . .	742
22.7.5	Data (ES5340-Analog-Direct-Out) . . . . .	743
22.7.6	ES5340-Sine-Extrapolated . . . . .	744
22.7.7	Globals (ES5340-Sine-Extrapolated) . . . . .	744
22.7.8	Groups (ES5340-Sine-Extrapolated) . . . . .	745
22.7.9	Signals (ES5340-Sine-Extrapolated) . . . . .	745
22.7.10	Data (ES5340-Sine-Extrapolated) . . . . .	746
22.7.11	ES5340-Sine-Encoder . . . . .	747
22.7.12	Globals (ES5340-Sine-Encoder) . . . . .	748
22.7.13	Groups (ES5340-Sine-Encoder) . . . . .	749
22.7.14	Signals (ES5340-Sine-Encoder) . . . . .	749
22.7.15	Data (ES5340-Sine-Encoder) . . . . .	750
22.7.16	ES5340-Resolver . . . . .	752
22.7.17	Globals (ES5340-Resolver) . . . . .	753
22.7.18	Groups (ES5340-Resolver) . . . . .	754
22.7.19	Signals (ES5340-Resolver) . . . . .	754
22.7.20	Data (ES5340-Resolver) . . . . .	755
22.7.21	ES5340-Analog-Arbitrary . . . . .	756
22.7.22	Globals (ES5340-Analog-Arbitrary) . . . . .	757
22.7.23	Groups (ES5340-Analog-Arbitrary) . . . . .	758
22.7.24	Signals (ES5340-Analog-Arbitrary) . . . . .	758
22.7.25	Data (ES5340-Analog-Arbitrary) . . . . .	759
22.8	ES5340-Analog-Out-Mux – Analoger Ausgangsmultiplexer . . . . .	761
22.8.1	Signals (ES5340-Analog-Out-Mux) . . . . .	761
22.8.2	Data (ES5340-Analog-Out-Mux) . . . . .	763
22.9	ES5340-Slave . . . . .	764
22.10	ES5340-Digital-Out (Slave) . . . . .	764
22.11	ES5340-Digital-Out-Mux (Slave) . . . . .	764
22.12	ES5340-Analog-Out (Slave) . . . . .	764
22.13	ES5340-Analog-Out-Mux (Slave) . . . . .	764
22.14	ES5340-PMSM-1.0.0 – PMSM-FPGA-Modell . . . . .	765
22.14.1	Globals (ES5340-PMSM-1.0.0) . . . . .	766
22.14.2	Allgemeines . . . . .	766
22.14.3	Invertermodell . . . . .	767
22.14.4	PMSM-Modell . . . . .	768
22.14.5	Mechanikmodell . . . . .	768
22.14.6	Groups (ES5340-PMSM-1.0.0) . . . . .	770
22.14.7	Signals (ES5340-PMSM-1.0.0) . . . . .	770
22.14.8	Data (ES5340-PMSM-1.0.0) . . . . .	775
22.15	ES5340-IM-1.0.0 – IM-FPGA-Modell . . . . .	779
22.15.1	Globals (ES5340-IM-1.0.0) . . . . .	780
22.15.2	Allgemeines . . . . .	780
22.15.3	Invertermodell . . . . .	781
22.15.4	IM-Modell . . . . .	782
22.15.5	Mechanikmodell . . . . .	782



22.15.6	Groups (ES5340-IM-1.0.0) . . . . .	784
22.15.7	Signals (ES5340-IM-1.0.0) . . . . .	784
22.15.8	Data (ES5340-IM-1.0.0) . . . . .	789
23	ES5340.2 Internal Combustion Engine Application . . . . .	793
23.1	ES5340-Master . . . . .	794
23.1.1	Globals (ES5340-Master Subsystem) . . . . .	794
23.2	ES5340-RPM – Drehzahleinheit . . . . .	795
23.2.1	ES5340-RPM – Drehzahleinheit . . . . .	796
23.2.2	Globals (ES5340-RPM) . . . . .	796
23.2.3	Groups (ES5340-RPM) . . . . .	797
23.2.4	Data (ES5340-RPM) . . . . .	798
23.3	ES5340-Analog-In – Analoge Eingänge . . . . .	799
23.3.1	ES5340-Analog-In – Analoge Eingänge . . . . .	799
23.3.2	Signals (ES5340-Analog-In) . . . . .	799
23.3.3	Data (ES5340-Analog-In) . . . . .	800
23.4	ES5340-Digital-Out – Digitale Ausgänge . . . . .	801
23.4.1	ES5340-Digital-Direct-Out . . . . .	802
23.4.2	Globals (ES5340-Digital-Direct-Out) . . . . .	802
23.4.3	Groups (ES5340-Digital-Direct-Out) . . . . .	802
23.4.4	Signals (ES5340-Digital-Direct-Out) . . . . .	802
23.4.5	Data (ES5340-Digital-Direct-Out) . . . . .	803
23.4.6	ES5340-PWM-Output . . . . .	804
23.4.7	Globals (ES5340-PWM-Output) . . . . .	804
23.4.8	Groups (ES5340-PWM-Output) . . . . .	804
23.4.9	Signals (ES5340-PWM-Output) . . . . .	804
23.4.10	Data (ES5340-PWM-Output) . . . . .	805
23.5	ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer . . . . .	806
23.5.1	Signals (ES5340-Digital-Out-Mux) . . . . .	806
23.5.2	Data (ES5340-Digital-Out-Mux) . . . . .	808
23.6	ES5340-Analog-Out – Analoge Ausgänge . . . . .	809
23.6.1	ES5340-Analog-Direct-Out . . . . .	809
23.6.2	Globals (ES5340-Analog-Direct-Out) . . . . .	809
23.6.3	Groups (ES5340-Analog-Direct-Out) . . . . .	809
23.6.4	Signals (ES5340-Analog-Direct-Out) . . . . .	810
23.6.5	Data (ES5340-Analog-Direct-Out) . . . . .	810
23.7	ES5340-Analog-Out-Mux – Analoger Ausgangsmultiplexer . . . . .	811
23.7.1	Signals (ES5340-Analog-Out-Mux) . . . . .	811
23.7.2	Data (ES5340-Analog-Out-Mux) . . . . .	812
23.8	ES5340-Slave . . . . .	813
23.9	ES5340-Digital-Out (Slave) . . . . .	813
23.10	ES5340-Digital-Out-Mux (Slave) . . . . .	813
23.11	ES5340-Analog-Out (Slave) . . . . .	813
23.12	ES5340-Analog-Out-Mux (Slave) . . . . .	813
23.13	ES5340-Measure Subsystem . . . . .	814
23.13.1	Grundprinzip der Messwertbestimmung . . . . .	814
23.13.2	Globals (ES5340-Measure Subsystem) . . . . .	816
23.14	ES5340-HW Device . . . . .	817
23.14.1	Groups (ES5340-HW Device) . . . . .	817

23.14.2	Signals (ES5340-HW Device) . . . . .	819
23.15	ES5340-MeasTime Device . . . . .	821
23.15.1	Nicht synchronisierte Zeit-/Frequenzmessungen . . . . .	821
23.15.2	Die Messverfahren . . . . .	822
23.15.3	Globals (ES5340-MeasTime Device) . . . . .	824
23.15.4	Groups (ES5340-MeasTime Device) . . . . .	824
23.15.5	Signals (ES5340-MeasTime Device) . . . . .	825
23.16	ES5340-MeasAngle Device . . . . .	828
23.16.1	Pulsweitenmessungen . . . . .	828
23.16.2	Pulsweitenmessungen mit Validierungssignal . . . . .	829
23.16.3	Additive Pulsweitenmessungen . . . . .	831
23.16.4	Vermessung von Flanken: Winkelstempel . . . . .	833
23.16.5	Vermessung von Flanken: Zeitstempel . . . . .	834
23.16.6	Pulszählung . . . . .	835
23.16.7	Globals (ES5340-MeasAngle Device) . . . . .	836
23.16.8	Groups (ES5340-MeasAngle Device) . . . . .	836
23.16.9	Signals (ES5340-MeasAngle Device) . . . . .	838
23.17	ES5340-RailPump Device . . . . .	841
23.17.1	Globals (ES5340-RailPump Device) . . . . .	841
23.17.2	Signals (ES5340-RailPump Device) . . . . .	843
23.18	Arbiträre Signalgeneratoren . . . . .	845
23.18.1	RPM-Generator . . . . .	845
23.18.2	Wellenformspeicher für Signalgeneratoren . . . . .	845
23.18.3	Klopfsignalgenerator . . . . .	846
23.18.4	Zündaussetzsteuerung . . . . .	847
23.18.5	Sequenztabellen . . . . .	847
23.18.6	MSA-Sensor . . . . .	848
23.19	ES5340-SigGen Subssystem . . . . .	849
23.19.1	Globals (ES5340-SigGen Subsystem) . . . . .	849
23.20	ES5340-SigGen Device . . . . .	851
23.20.1	Globals (ES5340-Analog Device) . . . . .	851
23.20.2	Groups (ES5340-Analog Device) . . . . .	852
23.20.3	Data (ES5340-Analog Device) . . . . .	852
23.21	ES5340-Misfire Device . . . . .	856
23.21.1	Globals (ES5340-Misfire Device) . . . . .	856
23.21.2	Groups (ES5340-Misfire Device) . . . . .	858
23.21.3	Signals (ES5340-Misfire Device) . . . . .	858
23.21.4	Data (ES5340-Misfire Device) . . . . .	859
23.22	ES5340-Knock Device . . . . .	861
23.22.1	Globals (ES5340-Knock Device) . . . . .	861
23.22.2	Groups (ES5340-Knock Device) . . . . .	863
23.22.3	Signals (ES5340-Knock Device) . . . . .	864
23.22.4	Data (ES5340-Knock Device) . . . . .	865
23.23	ES5340-MSA-Sensor Device . . . . .	868
23.23.1	Grundsätzliche Funktionsweise . . . . .	868
23.23.2	Zahnmittenberechnung . . . . .	868
23.23.3	Globals (ES5340-MSA-Sensor Device) . . . . .	869
23.23.4	Groups (ES5340-MSA-Sensor Device) . . . . .	870

23.23.5	Data (ES5340-MSA-Sensor Device)	871
23.23.6	Fehlerbehandlung	872
24	ES5350.1 Analog Board	873
24.1	ES5350 Subsystem	873
24.1.1	Globals (ES5350 Subsystem)	873
24.2	ES5350-Analog-In Device	874
24.2.1	Globals (ES5350-Analog-In Device)	874
24.2.2	Groups (ES5350-Analog-In Device)	874
24.2.3	Signals (ES5350-Analog-In Device)	875
24.3	ES5350-Analog-Out Device	876
24.3.1	Globals (ES5350-Analog-Out Device)	876
24.3.2	Groups (ES5350-Analog-Out Device)	877
24.3.3	Signals (ES5350-Analog-Out Device)	877
24.4	Module-Status Device	879
25	ES5385.1 Carrier Board for Resistor Cascade	881
25.1	ES5385Device	881
25.1.1	Globals (ES5385 Device)	881
25.2	ES5385-ResistorCascade Device	881
25.2.1	Globals (ES5385-ResistorCascade Device)	881
25.2.2	Groups (ES5385-ResistorCascade Device:Groups)	882
25.2.3	Signals (ES5385-ResistorCascade Device)	884
26	ES5392.1 High Current Switch Board	887
26.1	ES5392 Subsystem	887
26.1.1	Globals (ES5392 Subsystem)	887
26.2	ES5392-PwrCtrl Device	888
26.2.1	Globals (ES5392-PwrCtrl Device)	888
26.2.2	Groups (ES5392-PwrCtrl Device)	889
26.2.3	Signals (ES5392-PwrCtrl Device)	890
26.3	ES5392-SwCtrl-External Device	891
26.3.1	Globals (ES5392-SwCtrl-External Device)	891
26.3.2	Groups (ES5392-SwCtrl-External Device)	892
26.3.3	Signals (ES5392-SwCtrl-External Device)	892
26.3.4	Signals (ES5392-SwCtrl-DigOut)	894
26.3.5	Signals (ES5392-SwCtrl-HighCurrent)	895
26.4	ES5392-Meas-HighCurrent Device	897
26.4.1	Globals (ES5392-Meas-HighCurrent Device)	897
26.4.2	Groups (ES5392-Meas-HighCurrent Device)	897
26.4.3	Signals (ES5392-Meas-HighCurrent Device)	897
26.5	ES5392-MRC-Simulation Device	899
26.5.1	Globals (ES5392-MRC-Simulation Device)	899
26.5.2	Groups (ES5392-MRC-Simulation Device)	899
26.5.3	Signals (ES5392-MRC-Simulation Device)	900
26.6	ES5392-Out-Dig-Pwm Device	901
26.6.1	Globals (ES5392-Out-Dig-Pwm Device)	901
26.6.2	Groups (ES5392-Out-Dig-Pwm Device)	901
26.6.3	Signals (ES5392-Out-Dig-Pwm Device)	902
26.7	ES5392-Out-Multi-Pulse Device	903

26.7.1	Globals (ES5392-Out-Multi-Pulse Device) . . . . .	903
26.7.2	Groups (ES5392-Out-Multi-Pulse Device) . . . . .	904
26.7.3	Signals (ES5392-Out-Multi-Pulse Device) . . . . .	904
26.8	Module-Status Device . . . . .	904
27	IXXAT iPCI-I XC16/PCI CAN Interfacekarte . . . . .	905
27.1	IXXAT-XC16-CAN Subsystem . . . . .	906
27.1.1	Globals (IXXAT-XC16-CAN Subsystem) . . . . .	906
27.1.2	Versteckte Optionsfelder . . . . .	906
27.2	IXXAT-XC16-CAN-CTRL Subsystem . . . . .	908
27.2.1	Globals (IXXAT-XC16-CAN-CTRL Subsystem) . . . . .	908
27.3	IXXAT-XC16-CAN_IO Device . . . . .	911
27.3.1	Globals (IXXAT-XC16-CAN_IO Device) . . . . .	911
27.3.2	Groups (IXXAT-XC16-CAN_IO Device) . . . . .	913
27.3.3	Die CAN-Botschaften als Signalgruppen . . . . .	914
27.3.4	Signals (IXXAT-XC16-CAN_IO Device) . . . . .	915
27.3.5	Data (IXXAT-XC16-CAN_IO Device) . . . . .	918
28	Das RTIO-Package für Transputer-basierte Hardware . . . . .	919
28.1	ES4205 und ES1206/ES1381 - Buskopplung . . . . .	920
28.1.1	Globals (ES4205 oder ES1206-ES1381-ES4206 Subsystem) . . . . .	920
28.2	MTS-UBATT/UBATT-POW - Batteriesteuerung . . . . .	921
28.2.1	Globals (MTS-UBATT Device) . . . . .	922
28.2.2	Groups (MTS-UBATT Device) . . . . .	925
28.2.3	Signals (MTS-UBATT Device) . . . . .	926
28.2.4	Datentypen und Wertebereiche . . . . .	928
28.3	TS-PWM- und arbiträre Signalgenerierung . . . . .	929
28.3.1	Globals (TS Subsystem) . . . . .	930
28.3.2	Globals (TsPWM Device) . . . . .	931
28.3.3	Groups (TsPWM Device) . . . . .	931
28.3.4	Signals (TsPWM Device) . . . . .	932
28.3.5	Datentypen und Wertebereiche . . . . .	933
28.3.6	TsRPM Device . . . . .	933
28.3.7	Globals (TsRPM Device) . . . . .	935
28.3.8	Groups (TsRPM Device) . . . . .	936
28.3.9	Signals (TsRPM Device) . . . . .	937
28.3.10	Datentypen und Wertebereiche . . . . .	938
28.3.11	Globals (TsSIG Subsystem) . . . . .	939
28.3.12	Globals (TsSIGGen Device) . . . . .	939
28.3.13	Groups (TsSIGGen Device) . . . . .	941
28.3.14	Signals (TsSIGGen Device) . . . . .	942
28.3.15	TsKnockGen Device . . . . .	945
28.3.16	Globals (TsKnockGen Device) . . . . .	946
28.3.17	Groups (TsKnockGen Device) . . . . .	949
28.3.18	Signals (TsKnockGen Device) . . . . .	950
28.3.19	TsMisfire Device . . . . .	955
28.3.20	Globals (TsMisfire Device) . . . . .	956
28.3.21	Groups (TsMisfire Device) . . . . .	959
28.3.22	Signals (TsMisfire Device) . . . . .	960



28.4	TMIO - Digitale Signalerfassung und I/O	962
28.4.1	Grundprinzip der Messwertbestimmung	962
28.4.2	Messwernerfassung	964
28.4.3	Signalauswertung	965
28.4.4	Timeout-Erkennung	973
28.4.5	Winkelsegmentierung	973
28.4.6	Globals (TMIO Device)	974
28.4.7	Groups (TMIO Device)	975
28.4.8	Signals (TMIO Device)	977
28.4.9	Datentypen und Wertebereiche	980
28.4.10	Signal-Konfigurationsparameter	981
28.5	TDAC - Analoge Signalerfassung und -generierung	985
28.5.1	Globals (TDAC Device)	985
28.5.2	Groups (TDAC Device)	987
28.5.3	Signals (TDAC Device)	988
28.5.4	Wertebereiche	991
28.6	TRS422	993
28.6.1	Globals (TRS422 Subsystem)	993
28.7	T8Module Subsystem	994
28.7.1	Globals (T8Module Subsystem)	994
28.8	T8IO	995
28.8.1	Globals (T8IO Subsystem)	995
28.9	LOAD20A14C Hochstrom-Lastschaltkarte	996
28.9.1	Globals (Load20A14C Subsystem)	997
28.9.2	Globals (ErrorSim Device)	997
28.9.3	Groups (ErrorSim Device)	998
28.9.4	Signals (ErrorSim Device)	999
28.9.5	Datentypen und Wertebereiche	1000
28.9.6	Globals (LoadSwitch Device)	1001
28.9.7	Groups (LoadSwitch Device)	1001
28.9.8	Signals (LoadSwitch Device)	1002
28.9.9	Datentypen und Wertebereiche	1003
29	Module Status	1005
30	ETAS Kontaktinformation	1007
	Index	1009



## 1 Der RTIO-Editor

LABCAR-RTC (Real-Time Execution Connector) ist eine Zusatzsoftware zu LABCAR-OPERATOR, die zur Integration und Konfiguration der Hardwarekomponenten für HiL-Echtzeitexperimente benötigt wird.

Der RTIO-Editor ist die Bedienoberfläche von LABCAR-RTC, in der die Hardwarekonfiguration für das jeweilige LABCAR-OPERATOR Projekt erstellt wird.

Dieses Kapitel enthält folgende Informationen:

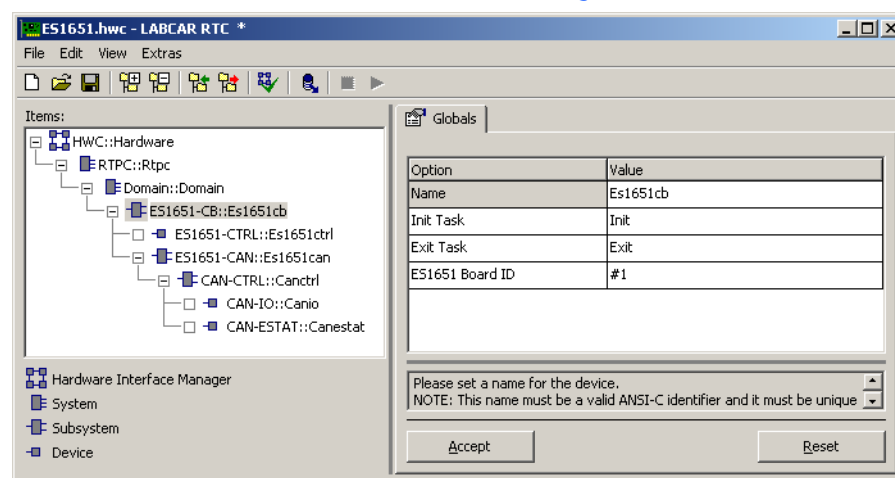
- „Die Bedienelemente des RTIO-Editors“ auf Seite 23
  - „Das Menü des RTIO-Editors“ auf Seite 24
  - „Die Symbolleiste“ auf Seite 32
- „Items“ auf Seite 33
- „Konfigurationsregister“ auf Seite 42

### 1.1 Die Bedienelemente des RTIO-Editors

In diesem Abschnitt werden die Bedienelemente des RTIO-Editors vorgestellt.

#### RTIO-Editor starten

- Öffnen Sie ein Projekt in LABCAR-OPERATOR.
  - Wählen Sie **Project** → **RTIO Editor**.
- Der RTIO Editor wird geöffnet.



**Abb. 1-1** Die Bedienoberfläche des RTIO mit einer geöffneten Hardwarekonfiguration

Die Benutzeroberfläche besteht aus der Menüleiste, einer Schaltflächenleiste, der Liste „Items“ und einem Satz Registerkarten, über die die meisten Modifikationen und Eingaben vorgenommen werden können.

Unterhalb der Liste „Items“ sind die verschiedenen Elemente der Hardwarekonfiguration mit ihren Symbolen aufgeführt, unterhalb der Registerkarten ist ein Textfeld, das Kurzinformationen zur ausgewählten Zelle enthält.

Mit den Schaltflächen **Accept** und **Reset** übernehmen bzw. verwerfen Sie vorgenommene Änderungen.

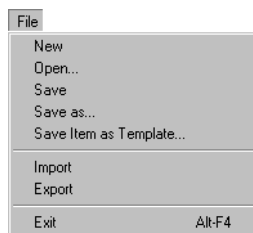
### 1.1.1 Das Menü des RTIO-Editors

In diesem Abschnitt finden Sie eine Beschreibung der Menüeinträge.

#### **Hinweis**

*Je nachdem, in welcher Betriebsart sich der RTIO-Editor befindet, sind die Menüpunkte verfügbar oder gesperrt. Die Buchstaben „E“ (Edit-Modus) und „R“ (Runtime-Modus) geben Aufschluss darüber, wann ein Menüpunkt zur Verfügung steht.*

#### Das Menü „File“

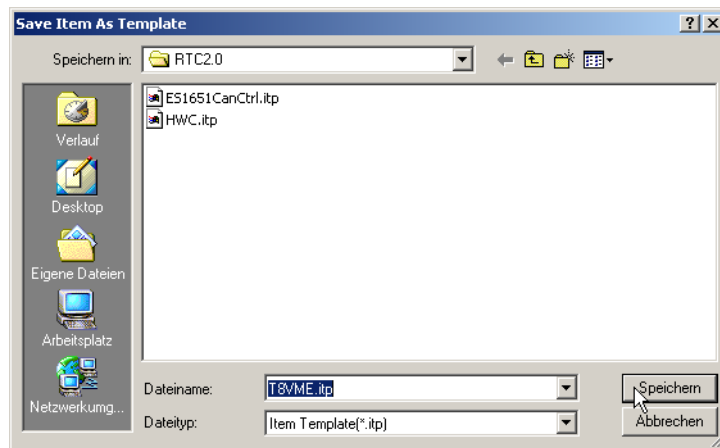


- **File → New (E)**  
Legt eine neue Hardwarekonfiguration an.
- **File → Open... (E, R)**  
Damit kann über ein Dateiauswahlfenster eine bestehende Hardwarekonfiguration in den RTIO-Editor geladen werden.
- **File → Save (E, R)**  
Sichert die aktuelle Hardwarekonfiguration.
- **File → Save As... (E, R)**  
Sichert die aktuelle Hardwarekonfiguration unter einem anderen Namen.

- **File** → **Save Item As Template...** (E)

Speichert die Einstellungen des gewählten Items als Template (\*.itp) für alle Items dieses Typs (siehe hierzu auch „Register „Templates““ auf Seite 53).

Im Dialogfenster „Save Item As Template“ kann das Item als Vorlage in dem angegebenen Verzeichnis gespeichert werden.

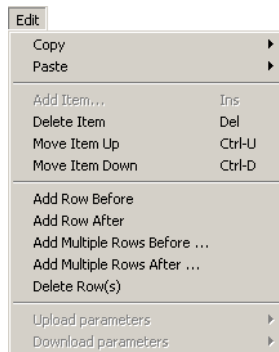


#### **Hinweis**

Beim Öffnen des Dialogs wird immer der Dateipfad verwendet, der in den „Optionen“-Einstellungen unter „Template Path“ voreingestellt ist (siehe „Register „Templates““ auf Seite 53). Der vorgeschlagene Dateiname darf nicht geändert werden.

- **File** → **Export** (E)  
Exportiert die aktuell geöffnete Hardwarekonfiguration unter Angabe eines neuen Dateinamens als \*.hwc-Datei oder als \*.csv-Datei
- **File** → **Exit** (E, R)  
Schließt den RTIO-Editor

## Das Menü „Edit“



- **Edit → Copy →**
  - **Item(s) (E)**

Erstellt eine Kopie des gewählten Items einschließlich des ggf. vorhandenen Teilbaumes und überträgt diese in einen internen Puffer des RTIO-Editors.

Der Puffer ist verfügbar, solange der RTIO-Editor geöffnet ist.
  - **Item(s) for Export (E)**

Funktioniert wie **Edit → Copy → Item(s)**, nur wird hier die Kopie in eine externe Datei geschrieben. Diese Funktion eignet sich dazu, Teilkonfigurationen auszutauschen.
  - **Item Data (E)**

Kopiert nur die Daten eines Items in einen internen Puffer (diese Funktion bezieht sich nicht auf ganze Teilbäume). Diese Funktion kann dazu verwendet werden, die Einstellungen zwischen zwei Items auszutauschen.
- **Edit → Paste →**
  - **To Selected Item (E)**

Fügt ein über **Edit → Copy → Item(s)** kopiertes Item (ggf. mit Teilbaum) unter das gerade gewählte Item ein. Folgende Voraussetzungen sind notwendig, damit die Funktion gewährleistet ist.

    1. Die Item-Typen müssen übereinstimmen.
    2. Es muss möglich sein, das Item bzw. den Teilbaum auch unter das gewählte Item einzufügen (hier gilt das Gleiche wie bei **Edit → Add Item...**).
  - **To Selected Item From Import (E)**

Funktioniert gleich wie **Edit → Paste → To Selected Item**, nur erfolgt hier das Einlesen eines zuvor in einer Datei gespeicherten Items bzw. Teilbaums. Es gelten dieselben Beschränkungen wie beim vorigen Menüpunkt.

- **Data To Selected Item** (E)

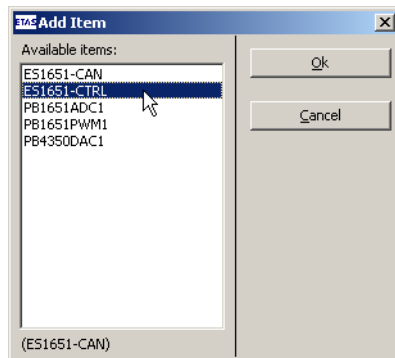
Überschreibt die Daten des gewählten Items mit denen, die über **Edit → Copy → Item Data** in den Puffer geschrieben wurden.

Im Gegensatz zu den vorhergehenden **Edit → Paste →** Menüpunkten gibt es hier auch die Möglichkeit, Daten zwischen ähnlichen bzw. kompatiblen Items auszutauschen. Diese Funktion ist nur dann möglich, wenn der Typ des gewählten Items gleich oder kompatibel zu dem Item ist, von dem die Daten kopiert wurden. Darüberhinaus muss auch die Struktur nahezu identisch sein; d.h. die Anzahl der Reihen in den Tabellen in den Registern „Groups“, „Signals“ und „Data“ muss übereinstimmen.

- **Edit → Add Item...**

Über diese Menüfunktion kann ein neues Hardware-Item in die Hardwarekonfiguration eingefügt werden.

Ein Auswahlfenster öffnet sich, das alle verfügbaren Items der nächst tieferliegenden Hierarchiestufe anzeigt.



Das gewählte Element wird dann in die Hardwarekonfiguration eingefügt.

### Hinweis

*Abhängig von der bereits erstellten Konfiguration kann (auch nach erfolgreicher Auswahl eines Items) das Einfügen mit einer Fehlermeldung abgebrochen werden. Dies ist z. B. dann der Fall, wenn die Anzahl zur Verfügung stehender Ressourcen (Ports, Steckplätze) bereits aufgebraucht ist oder wenn eine bestimmte Einfügereihenfolge notwendig ist.*

- **Edit → Delete Item** (E)

Über diese Menüfunktion kann das markierte Element aus der Hardwarekonfiguration gelöscht werden.

Das Entfernen des obersten Items der Hierarchie (Hardware Interface Manager) ist nicht möglich.

### Hinweis

*Manchmal wird der Versuch, ein Element aus der Hardwarekonfiguration zu entfernen, mit einer Fehlermeldung abgebrochen. Das ist zum Beispiel dann der Fall, wenn eine bestimmte Löschrreihenfolge eingehalten werden muss.*

- **Edit → Move Item Up** (E)  
Dient zum Ändern der Item-Reihenfolge innerhalb einer Hierarchiestufe in der Hardwarekonfiguration. Die Menüfunktion bewegt das ausgewählte Item um ein Element nach oben.
- **Edit → Move Item Down** (E)  
Dient zum Ändern der Item-Reihenfolge innerhalb einer Hierarchiestufe in der Hardwarekonfiguration. Die Menüfunktion bewegt das ausgewählte Item um ein Element nach unten.
- **Edit → Add Row Before** (E)  
Dieser Menüpunkt ist nur für Items des Typs „Device“ verfügbar, bei denen es möglich ist, Signalgruppen- bzw. Signalanzahl dynamisch zu ändern.

Der Menüpunkt ist unter folgenden Voraussetzungen verfügbar und wählbar:

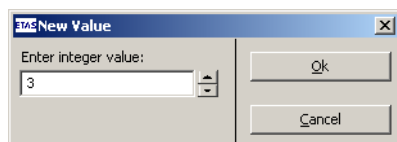
1. Es ist ein Item vom Typ „Device“ ausgewählt, das über „dynamische“ Eigenschaften verfügt (z.B. CAN-IO Device).
2. Das Register „Groups“ oder „Signals“ ist ausgewählt.
3. Ein Eintrag der Spalte „No.“ ist ausgewählt (1. Spalte in der Tabelle).

Damit wird eine Zeile vor der gewählten Zelle bzw. dem gewählten Bereich eingefügt.

#### Hinweis

*Systembedingt ist die Größe der Tabelle oft beschränkt. In diesem Fall bleibt das Ausführen dieser Funktion wirkungslos. Da es sich hier um eine strukturelle Änderung der Tabelle handelt, lässt sich diese Änderung nicht mehr durch die Schaltfläche **Reset** wieder rückgängig machen.*

- **Edit → Add Row After** (E)  
Funktioniert wie **Add Row Before**, nur wird hier eine neue Zeile nach der gewählten Zelle bzw. dem gewählten Bereich eingefügt.
- **Edit → Add Multiple Rows Before...** (E)  
Funktioniert wie **Add Row Before**, nur können hier gleich mehrere Zeilen eingefügt werden. Die Anzahl der neu einzufügenden Zeilen kann in einem Dialogfenster eingegeben werden:



- **Edit → Add Multiple Rows After...** (E)  
Funktioniert wie unter **Add Multiple Rows Before** beschrieben, nur werden hier neue Zeilen nach der gewählten Zelle bzw. dem gewählten Bereich eingefügt.



- **Edit → Delete Row(s)** (E)  
Löscht die Zeile bzw. den Bereich, der in der Spalte „No.“ markiert ist.

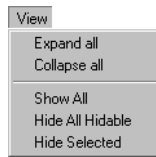
#### **Hinweis**

*Aus Systemgründen muss mindestens eine Signalgruppe bzw. ein Signal in der Tabelle verbleiben. Beim Versuch, die letzte Reihe zu löschen, erscheint eine Fehlermeldung.*

- **Edit → Upload Parameters →**
  - **for selected item** (R)  
Damit werden alle im Runtime-Modus veränderlichen Werte des ausgewählten Items vom Target ausgelesen und die aktuell im RTIO-Editor geladene Hardwarekonfiguration mit diesen Werten überschrieben.
  - **for selected item (recursive)** (R)  
Die Funktion ist identisch zur vorher beschriebenen mit dem Unterschied, dass hier auch die Parameter der unterhalb des gewählten Items liegenden RTIO-Baumstruktur aktualisiert werden.
  - **all items** (R)  
Die Funktion ist identisch zur vorher beschriebenen mit dem Unterschied, dass hier die Parameter der gesamten RTIO-Baumstruktur aktualisiert werden.
- **Edit → Download Parameters →**
  - **for selected item** (R)  
Damit werden alle im Runtime-Modus veränderlichen Werte des ausgewählten Items zum Target heruntergeladen. Wird die Funktion ausgeführt, so werden alle entsprechenden Einstellungen im C-Code-Modul mit den aktuellen Werten vom RTIO-Editor überschrieben.
  - **for selected item (recursive)** (R)  
Die Funktion ist identisch zur vorher beschriebenen mit dem Unterschied, dass hier auch die Parameter der unterhalb des gewählten Items liegenden RTIO-Baumstruktur aktualisiert werden.
  - **all items** (R)  
Die Funktion ist identisch zur vorher beschriebenen mit dem Unterschied, dass hier die Parameter der gesamten RTIO-Baumstruktur aktualisiert werden.

### Das Menü „View“

---



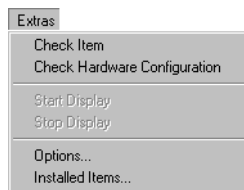
- **View → Expand All** (E, R)  
Faltet den Hierarchiebaum der RTIO-Items (siehe „Items“ auf Seite 33) vollständig auf.
- **View → Collapse All** ((E, R)  
Verbirgt den Hierarchiebaum der RTIO-Items (siehe „Items“ auf Seite 33) vollständig.
- **View → Show All** (E, R)  
Dieser Menüpunkt blendet alle darstellbaren Optionen/Spalten einer Registerkarte ein.  
  
Die Tabelle einer Registerkarte kann so gestaltet sein, dass einige der möglichen Optionen bzw. Spalten standardmäßig nicht eingeblendet sind. Dies hat den Vorteil, dass wenig benutzte oder selten gebrauchte Optionen ausgeblendet und so die Tabelle übersichtlicher gestalten werden kann.

#### Hinweis

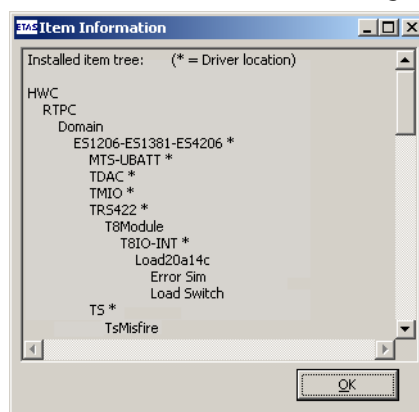
*Wird anschließend die Tabelle mit der Schaltfläche **Accept** bestätigt und die Hardwarekonfiguration gesichert, steht nach dem erneuten Laden die Tabelle wieder so zur Verfügung, wie sie gesichert wurde, d.h. neu eingeblendete Optionen/Spalten bleiben eingeblendet.*

- **View → Hide All Hidable** (E, R)  
Bildet das Gegenstück zum vorigen Menüpunkt, d.h. alle ausblendbaren Optionen/Spalten werden ausgeblendet.
- **View → Hide Selected** (E, R)  
Blendet die aktuell gewählte Option bzw. Spalte aus, sofern sie ausblendbar ist.

## Das Menü „Extras“



- **Extras → Check Item (E)**  
Diese Funktion ermöglicht eine explizite Überprüfung aller Einstellungen des ausgewählten Items. Diese Überprüfung erfolgt auch implizit vor jeder RTIO-Codegenerierung.  
Erst wenn diese Überprüfung ohne Fehler abgeschlossen werden konnte, ist eine RTIO-Codegenerierung überhaupt möglich. Warnungen sind zulässig, sollten aber im Einzelfall immer beachtet werden.
- **Extras → Check Hardware Configuration (E)**  
Führt eine Überprüfung der gesamten Hardwarekonfiguration durch.
- **Extras → Start Display (R)**  
Startet die Anzeige der Spalte „Data“ im „Data“ Register der Items vom Typ „Device“.
- **Extras → Stop Display (R)**  
Stoppt die Anzeige der „Data“ Spalte im „Data“ Register der Items vom Typ „Device“.
- **Extras → Options... (E, R)**  
Öffnet ein Dialogfenster mit mehreren Registern, in denen verschiedene Optionen eingestellt werden können (siehe „Optionen“ auf Seite 52).
- **Extras → Installed Items... (E, R)**  
Öffnet das Fenster „Item Information“, in dem alle verfügbaren Hardware-Items in ihrer Hierarchie aufgelistet werden.



### 1.1.2 Die Symbolleiste

In diesem Abschnitt finden Sie eine Beschreibung der über die Schaltflächen der Symbolleiste erreichbaren Funktionen.



1. **New**

Legt eine neue Hardwarekonfiguration an.

Gleichbedeutend mit **File → New**.

2. **Open**

Damit kann über ein Dateiauswahlfenster eine bestehende Hardwarekonfiguration in den RTIO-Editor geladen werden.

Gleichbedeutend mit **File → Open...**

3. **Save**

Sichert die aktuelle Hardwarekonfiguration.

Gleichbedeutend mit **File → Save**.

4. **Expand all**

Faltet den Hierarchiebaum der RTIO-Items (siehe „Items“ auf Seite 33) vollständig auf.

Gleichbedeutend mit **View → Expand all**.

5. **Collapse all**

Verbirgt den Hierarchiebaum der RTIO-Items (siehe „Items“ auf Seite 33) vollständig.

Gleichbedeutend mit **View → Collapse all**.

6. **Add Item...**

Hiermit kann ein neues Hardware-Item in die Hardwarekonfiguration eingefügt werden.

Gleichbedeutend mit **Edit → Add Item...**

7. **Delete Item**

Hiermit wird das ausgewählte Element aus der Hardwarekonfiguration gelöscht.

Gleichbedeutend mit **Edit → Delete Item**.

8. **Check Hardware Configuration**

Dieser Menüpunkt ermöglicht eine explizite Überprüfung aller Einstellungen hinsichtlich Konsistenz innerhalb der Hardwarekonfiguration.

Gleichbedeutend mit **Extras → Check Hardware Configuration**.

9. **Export**

Damit kann die aktuell geöffnete Hardwarekonfiguration unter Angabe eines neuen Dateinamens als \*.hwc-Datei oder als \*.csv-Datei exportiert werden.

Gleichbedeutend mit **File → Export**.

### 10. Stop Display

Stoppt die Anzeige der „Data“ Spalte in den „Data“ Registern der Items vom Typ „Device“.

Gleichbedeutend mit **Extras** → **Stop Display**.

### 11. Start Display

Startet die Anzeige der Spalte „Data“ in den „Data“ Registern von den Items vom Typ „Device“.

Gleichbedeutend mit **Extras** → **Start Display**.

## 1.2 Items

Die baumartige Liste „Items“ dient zur Anzeige der Struktur der aktuellen Hardwarekonfiguration.

Der Ausdruck „Item“ bezeichnet ganz allgemein ein Element einer Hardwarekonfiguration. Das kann entweder eine gesamte Einschubkarte sein oder nur eine funktionell in sich abgeschlossene Einheit.

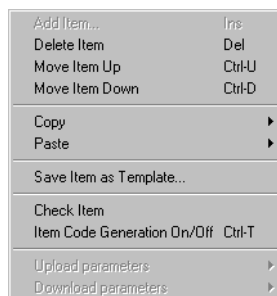
### Hinweis

*Eine Hardware ist dann vollständig spezifiziert, wenn jeder Endknoten des Hardware-Baums vom Typ „Device“ ist.*

Die Register rechts neben der Liste „Items“ zeigen immer die zum ausgewählten Item zugehörigen Einstellungen an.

### 1.2.1 Das Kontextmenü der Liste „Items“

Rechtsklicken in der Liste „Items“ öffnet das Kontextmenü.



**Abb. 1-2** Das Kontextmenü der Liste „Items“

Über dieses Kontextmenü sind verschiedene Aktionen möglich, die im Folgenden beschrieben werden.

#### **Ein Item hinzufügen**

- Um ein Item zur Hardwarekonfiguration hinzuzufügen, wählen Sie **Add Item....**

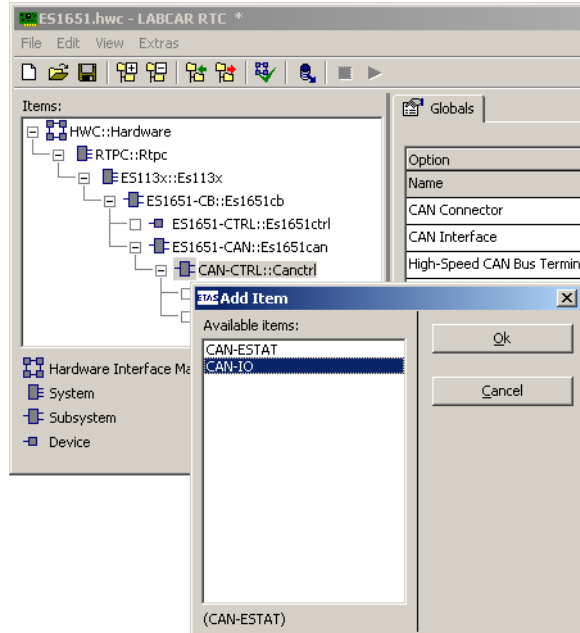
oder

- Drücken Sie <EINFG>.

oder



- Klicken Sie die Schaltfläche **Add Item....**  
Das Fenster „Add Item“ wird geöffnet.



- Wählen Sie aus der Liste der verfügbaren Items der nächst tieferliegenden Hierarchiestufe das einzufügende.
- Klicken Sie **OK**.  
Das gewählte Item wird eingefügt.

### Ein Item entfernen

- Um ein Item aus der Hardwarekonfiguration zu entfernen, markieren Sie dieses und wählen Sie **Delete Item**.

oder

- Drücken Sie <ENTF>.

oder



- Klicken Sie die Schaltfläche **Delete Item**.  
Das gewählte Item wird entfernt.

#### Hinweis

*Sollten unterhalb des zu entfernenden Items weitere Items vorhanden sein, werden diese ebenfalls entfernt!*

### Ein Item nach oben bewegen

---

- Wenn Sie ein Item (innerhalb der Hierarchiestufe) nach oben bewegen wollen, wählen Sie **Move Item Up**.

*oder*

- Drücken Sie <CTRL-U>.  
Das gewählte Item wird nach oben bewegt.

### Ein Item nach unten bewegen

---

- Wenn Sie ein Item (innerhalb der Hierarchiestufe) nach unten bewegen wollen, wählen Sie **Move Item Down**.

*oder*

- Drücken Sie <CTRL-D>.  
Das gewählte Item wird nach unten bewegt.

### Ein Item kopieren

---

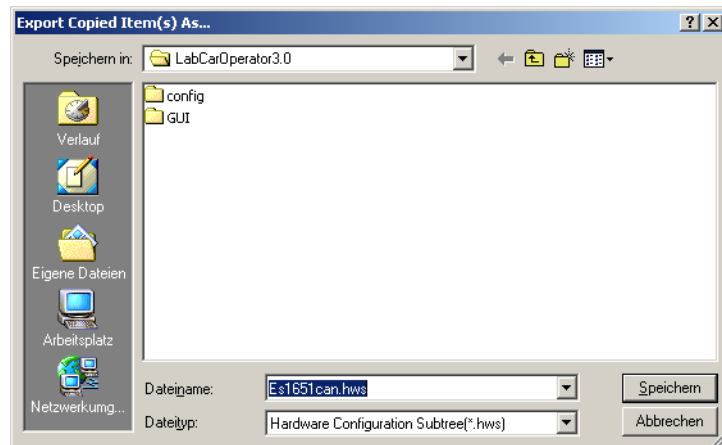
- Um ein Item (samt der eventuell darunter befindlichen Items) zu kopieren, wählen Sie **Copy → Item(s)**.

*oder*

- Drücken Sie <STRG-C>.  
Das gewählte Item oder der gewählte Teilbaum werden in den Puffer kopiert.

## Ein Item für den Export kopieren

- Um ein Item (samt der eventuell darunter befindlichen Items) zu in eine Datei kopieren, wählen Sie **Copy → Item(s) for Export**.  
Das Dateiauswahlfenster „Export Copied Item(s) As...“ wird geöffnet.



- Wählen Sie einen Dateinamen und klicken Sie **Speichern**.  
Der gewählte Teilbaum der Hardwarekonfiguration wird in einer Datei mit der Erweiterung „hws“ (Hardware Configuration Subtree) gespeichert.

## Daten (Einstellungen) eines Items kopieren

- Um die Daten eines einzelnen Items in den Puffer zu kopieren, wählen Sie **Copy → Item Data**.  
Die Einstellungen des gewählten Items werden in den Puffer kopiert.

### Hinweis

*Dabei werden nur die Daten eines einzelnen Items, nicht aber die der darunterliegenden Items kopiert.*

## Ein Item einfügen

- Um ein im Puffer befindliches Item (samt der eventuell darunter befindlichen Items) in Ihre Hardwarekonfiguration einzufügen, markieren Sie das Item, unterhalb dessen eingefügt werden soll.
- Wählen Sie **Paste → to selected Item**.  
*oder*



- Drücken Sie <STRG-V>.  
Das im Puffer befindliche Item (samt der eventuell darunter befindlichen Items) wird an der gewünschten Stelle eingefügt.

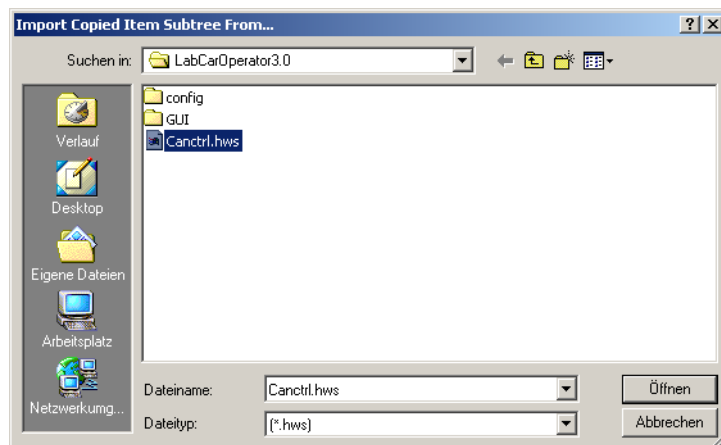
#### Hinweis

Beachten Sie hierzu die in „Das Menü „Edit““ auf Seite 26 genannten Bedingungen.

### Ein Item aus einer zuvor exportierten Datei einfügen

- Um ein zuvor in eine Datei exportiertes Item (inklusive der eventuell darunter befindlichen Items, siehe „Ein Item für den Export kopieren“ auf Seite 36) wieder einzufügen, wählen Sie **Paste** → **to selected Item from Import**.

Das Dateiauswahlfenster „Import Copied Items Subtree From...“ wird geöffnet.



- Wählen Sie eine Datei (\*.hws) und klicken Sie **Öffnen**.

#### Hinweis

Beachten Sie hierzu die in „Das Menü „Edit““ auf Seite 26 genannten Bedingungen.

### Daten (Einstellungen) eines Items einfügen

---

- Um Daten eines Items einzufügen, die über **Copy** → **Item Data** in den Puffer kopiert wurden, wählen Sie **Paste** → **Data to selected Item**.  
Die Daten werden kopiert.

#### **Hinweis**

Beachten Sie hierzu die in „Das Menü „Edit““ auf Seite 26 genannten Bedingungen.

### Ein Item als Vorlage speichern

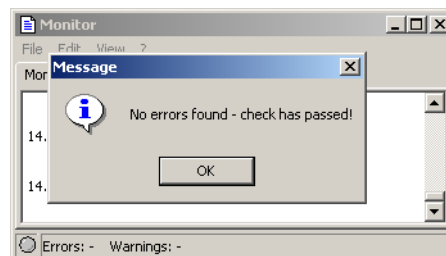
---

- Um ein bestimmtes Item als Vorlage zu speichern, wählen Sie **Save Item as Template...** .  
Näheres dazu finden Sie in der Beschreibung dieses Menüpunktes in „Das Menü „File““ auf Seite 24.

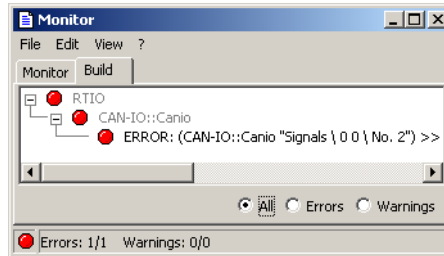
### Konfiguration eines einzelnen Items überprüfen

---

- Um die Konfiguration eines einzelnen Items zu überprüfen, wählen Sie **Check Item**.  
Die Konfiguration wird überprüft. Wenn diese In Ordnung ist, erhalten Sie eine entsprechende Meldung



Enthält die Konfiguration Fehler, so werden im Fenster „Messages“ entsprechende Meldungen ausgegeben.



### Hinweis

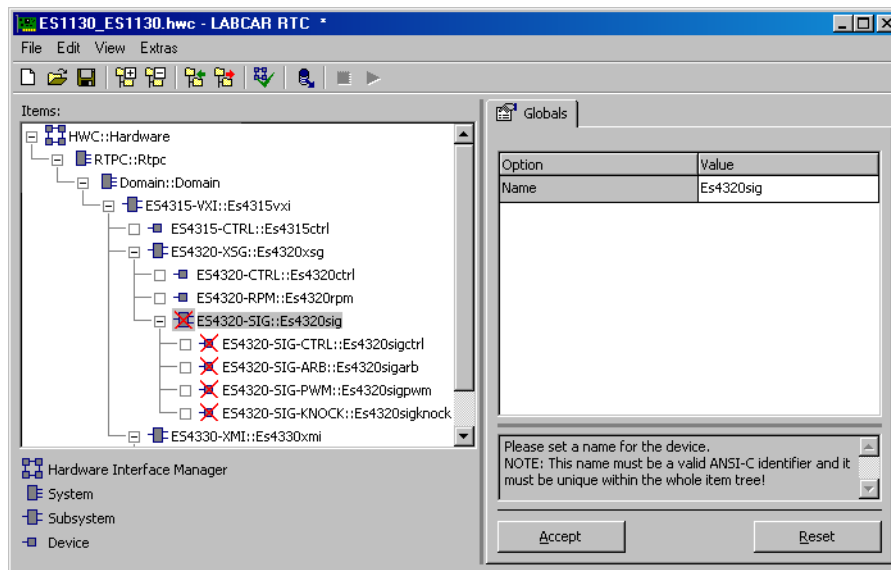
Dabei wird nur die Konfiguration des gewählten Items überprüft - für die Überprüfung der gesamten Hardwarekonfiguration wählen Sie den Menüpunkt **Check Hardware Configuration** (siehe „Das Menü „Extras““ auf Seite 31).

## Codegenerierung für Item deaktivieren

- Um die Codegenerierung für ein Item (einschließlich der darunter befindlichen Items) zu deaktivieren, markieren Sie dieses.
- Im Kontextmenü wählen Sie **Item Code Generation On/Off ...**.

oder

- Drücken Sie <STRG-T>.  
Die Codegenerierung für dieses Item (einschließlich der darunter befindlichen Items) wird deaktiviert.



Von der Codegenerierung ausgenommene Items werden durch ein rotes Kreuz gekennzeichnet.

Die Deaktivierung lässt sich auf demselben Weg wieder rückgängig machen.

### Parameter vom Target hochladen

- Um die aktuellen Werte der Hardwarekonfiguration eines gewählten Items vom Target in den RTIO-Editor zu laden, wählen Sie **Upload Parameters** → **for selected item**.
- Um die aktuellen Werte der Hardwarekonfiguration eines gewählten Items inklusive der darunterliegenden Items vom Target in den RTIO-Editor zu laden, wählen Sie **Upload Parameters** → **for selected item (recursive)**.
- Um die aktuellen Werte der Hardwarekonfiguration aller vorhandenen Items vom Target in den RTIO-Editor zu laden, wählen Sie **Upload Parameters** → **all items**.

### Parameter zum Target herunterladen

- Um im RTIO-Editor geänderte Einstellungen eines ausgewählten Items der Hardwarekonfiguration zum Target herunterzuladen, wählen Sie **Download parameters** → **for selected item**.

- Um im RTIO-Editor geänderte Einstellungen eines ausgewählten Items der Hardwarekonfiguration inklusive der darunter liegenden Items zum Target herunterzuladen, wählen Sie **Download parameters** → **for selected item (recursive)**.
- Um im RTIO-Editor geänderte Einstellungen aller vorhandenen Items der Hardwarekonfiguration zum Target herunterzuladen, wählen Sie **Download parameters** → **all items**.

**Hinweis**

*Hoch- und Herunterladen vom und zum Target ist natürlich nur möglich, wenn ein Experiment geöffnet ist.*

### 1.3 Konfigurationsregister

Je nach dem ausgewählten Item in der Liste „Items“ werden unterschiedliche Registerkarten angezeigt, über die die individuellen Einstellungen für die Items vorgenommen werden können.

Ist das gewählte Item vom Typ „Device“, so werden vier Register eingeblendet; bei allen anderen Elementtypen gibt es nur das Register „Globals“.

Die Aufgabenverteilung auf die Register ist wie folgt:

- **Globals**

Hier finden sich alle globalen Einstellungen, die das gesamte Element betreffen (z.B. VMEbus-Adresse der I/O-Karte).

- **Groups**

Hier befinden sich alle Einstellungen, die eine Signalgruppe betreffen (z.B. Verstärkungsfaktor für einen mehrkanaligen AD-Wandler oder den Identifier einer CAN-Botschaft).

#### Hinweis

*Eine Signalgruppe zeichnet sich dadurch aus, dass alle in ihr enthaltenen Signale die gleiche Übertragungsrichtung (send oder receive) aufweisen und zum gleichen Zeitpunkt oder im selben Raster erfasst werden.*

- **Signals**

In diesem Register werden alle signalspezifischen Einstellungen vorgenommen (z.B. Umrechnungsformel für ein Signal).

- **Data**

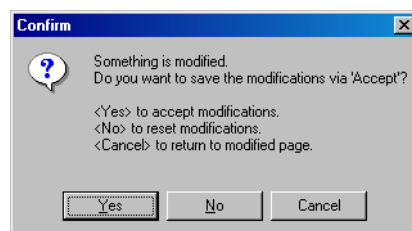
In diesem Register werden die Daten der einzelnen Signale angezeigt bzw. können diese editiert werden (je nachdem ob die „Direction“ der Signalgruppe den Wert „receive“ oder „send“ aufweist).

Darüber hinaus ist jedes Register noch mit den zwei Schaltflächen **Accept** und **Reset** ausgerüstet, die folgende Aufgabe haben:

**Accept:** (Edit-Modus und Runtime-Modus)

Sobald ein Register erstmalig oder erneut aufgebaut wird, wird von dem darin enthaltenen Datensatz eine Kopie erstellt, die dann in der Daten-Tabelle angezeigt wird. Jede Änderung der Daten modifiziert nur die Kopie der Originaldaten.

Ist das Editieren beendet, so muss der Anwender durch Drücken der Schaltfläche **Accept** diese veränderten Daten auf die Originaldaten explizit zurückschreiben. Er wird dazu aufgefordert, falls er versehentlich vorzeitig das Register verlassen möchte:



Sichern Sie die Änderung mit **Yes**, verwerfen Sie die vorgenommenen Änderungen mit **No** oder brechen Sie den Wechsel mit **Cancel** ab.

Dieser Mechanismus ist notwendig, um unter allen Umständen einen konsistenten Datensatz zu gewährleisten, da zum Teil innerhalb der „Items“-Struktur vielfache Abhängigkeiten zu berücksichtigen sind.

**Accept:** (nur Runtime-Modus)

Bei einigen Elementen können manche Einstellungen auch noch während des „Runtime“-Betriebs verändert werden. Dies ist immer dann möglich, wenn die entsprechende Hardware-Komponente dies erlaubt.

Solche Einstellungen kann man im Runtime-Modus daran erkennen, dass sie nicht „disabled“, also editierbar sind. Werden so ausgerüstete Zellen der Tabelle verändert, so löst ein Klicken auf **Accept** nicht nur einen Abgleich der Daten aus, sondern auch eine Aktualisierung aller referenzierten Parameter.

#### **Hinweis**

*Um den Treiber tatsächlich über einen veränderten Datenstand und die notwendige Rekonfiguration zu informieren, ist eine laufende „Config“-Task zwingend notwendig. Ohne diese kann der Treiber die Daten nicht verarbeiten.*

#### **Reset:**

Diese Schaltfläche macht alle nach dem letzten Klicken von **Accept** vorgenommenen Änderungen rückgängig.

### 1.3.1 Allgemeine Hinweise

---

#### *Option / Zelle editieren*

---

In der Tabelle eines Registers kann durch Anklicken mit der Maus die betreffende Zelle editiert werden. Die Navigation innerhalb der Tabelle kann auch mit den Pfeiltasten erfolgen, wobei das Drücken der Leertaste den Editor auf die gerade ausgewählte Zelle öffnet.

Manche Spalten sind auch multi-selektionsfähig, d.h. es können gleichzeitig mehrere Zellen innerhalb der Spalte gewählt werden (z.B. „Formula“-Spalte). Eine Mehrfachauswahl kann mit einem linken Mausklick und gleichzeitigem „Ziehen“ der Maus über die Spalte oder durch zweiten linken Mausklick und Betätigen der <SHIFT>-Taste erfolgen. Auch ein Navigieren über Tastatur mit gleichzeitig gedrückter <SHIFT>-Taste ist möglich.

#### *Editierbarkeit*

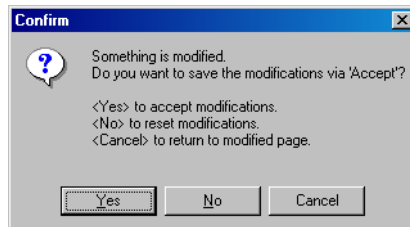
---

Nicht alle Optionen können vom Anwender verändert werden. Manche Optionen sind ständig gesperrt und dienen nur zur Anzeige von bestimmten Statuswerten. Andere Optionen können zeitweise gesperrt oder eingabebereit sein, was unter Umständen von anderen Optionseinstellungen in diesem oder einem anderen Item in der Items-Hierarchie abhängig sein kann.

**Tipp:** Öffnen Sie die Hardwarekonfiguration im Runtime-Modus; blenden Sie alle Spalten oder Optionen ein und beobachten Sie, welche Werte dann noch änderbar sind.

### Modifiziert-Zustand

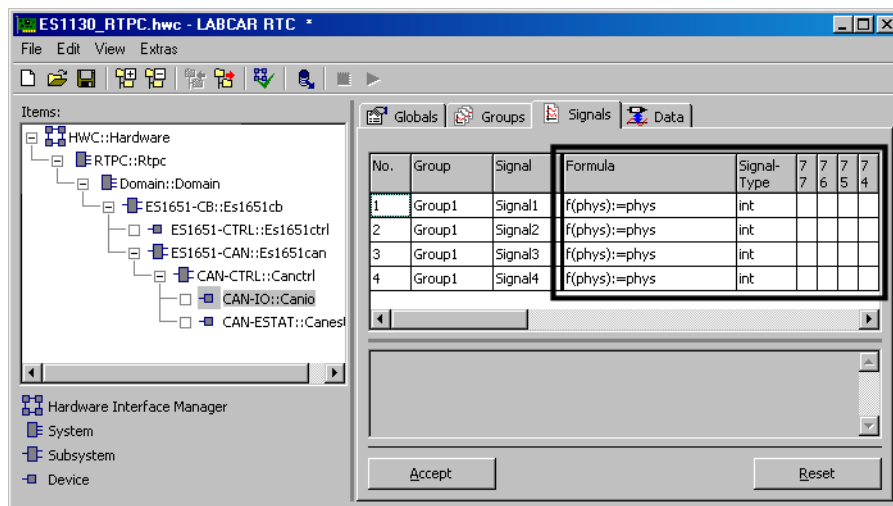
Der RTIO-Editor registriert, ob ein Wert der Tabelle bereits geändert wurde oder sich gerade im Zustand der Bearbeitung befindet. Wird dabei ein anderes Item bzw. Register angewählt, ohne dass die Änderung zuvor über die Schaltfläche **Accept** gültig gemacht wurde, erhalten Sie folgende Abfrage:



Sichern Sie die Änderung mit **Yes**, verwerfen Sie die vorgenommenen Änderungen mit **No** oder brechen Sie den Wechsel in ein anderes Register mit **Cancel** ab.

### Scrollen der Tabelle

Die Tabelle kann unter Umständen zu groß für die Darstellung in einem Register sein. In diesem Fall werden „Scrollbars“ eingeblendet.



**Abb. 1-3** Scrollbarer Bereich eines Konfigurationsregisters

Mit ihnen wird dann aber nicht die gesamte Tabelle verschoben, sondern nur der variable Teil (markierter Bereich in Abb. 1-3); der übrige Teil bleibt unverändert.

### Hinweis

Unter Umständen kann es vorkommen, dass im linken (statischen) Teil der Tabelle so viele Spalten eingeblendet sind, dass der rechte scrollbare Teil nicht mehr in das Register passt, so dass der Anwender vermeintlich keine Reaktion auf ein Scrollen sieht. Dieses Problem kann entweder durch ein Vergrößern des RTIO-Editors oder durch ein Ausblenden nicht benötigter statischer Spalten behoben werden.



### Spaltenbreite ändern

Die Spaltenbreite kann durch linken Mausklick auf die Spaltenbegrenzung mit anschließendem Festhalten nach Belieben verändert werden. Die neu festgelegten Spaltenbreiten werden dann mit der Hardwarekonfiguration mit gesichert, wenn sie zuvor über die Schaltfläche **Accept** bestätigt wurden.

Sie werden dann nach erneutem Laden wieder wie gewünscht eingestellt.

### Hilfe-Text

Im unteren Teil des Registers ist ein Textfeld angebracht, das die Möglichkeit bietet, einen kurzen Hilfetext für die gerade ausgewählte Option anzuzeigen.

### 1.3.2 Standardoptionen im Register „Globals“

In diesem Abschnitt werden die im Register „Globals“ geführten Standard-Optionen beschrieben. Weitere Item-spezifische Optionen finden Sie in den entsprechenden Abschnitten bei den jeweiligen Boards.

#### Name

Hier kann eine individueller Name für das Item vorgegeben werden. Der Name muss ein gültiger ANSI-C kompatibler Name sein und im gesamten Hardware-Baum eindeutig sein.

Diese Option ist in jedem „Globals“-Register vorhanden.

#### Init Task

Hier muss die Task ausgewählt werden, in der der Hardware-Treiber initialisiert werden soll. Das ist in der Regel eine „echte“ Init-Task. In Ausnahmefällen kann es auch eine „Software“-Task sein, das ist jedoch ungewöhnlich und zieht bei der Überprüfung eine Warnung nach sich.

Diese Option ist immer dann vorhanden, wenn das Item mit einem Low-Level-Treiber in Verbindung steht.

#### **Hinweis**

*In jedem Fall muss bei der Abarbeitung der Tasks sichergestellt sein, dass diese Task erstmalig ausgeführt wird, bevor irgendeine andere Task, die mit diesem Treiber in Verbindung steht (Start, Config...), ausgeführt werden kann. Desweiteren muss darauf geachtet werden, dass der „Init“-Aufruf erst wieder erfolgen darf, wenn zuvor ein „Exit“-Aufruf erfolgt ist. Falls nicht, so kommt es zu einem Laufzeitfehler im Experiment was eine „Sperrung“ des Treibers nach sich zieht (Driver Lock)!*

Eine in sich konsistente Task-Reihenfolge muss in jedem Fall eingehalten werden:

1. Init Task
2. Start Task
3. Stop Task
4. Exit Task

### Start Task

Ist diese Einstellung vorhanden, so kann hier eine separate Task angegeben werden, die den Treiber startet. Ist dies nicht notwendig, so kann hier in der Regel die gleiche Task angegeben werden wie bei der „Init Task“.

Die „Start Task“ wird nur von einigen Hardware-Treibern unterstützt.

#### **Hinweis**

*Task-Reihenfolge unbedingt beachten (siehe „Init Task“ auf Seite 45)!*

### Stop Task

Ist diese Einstellung vorhanden, so kann hier eine separate Task angegeben werden, die den Treiber anhält. Ist dies nicht notwendig, so kann hier in der Regel die gleiche Task angegeben werden wie bei der „Exit Task“.

Die „Stop Task“ wird nur von einigen Hardware-Treibern unterstützt.

#### **Hinweis**

*Task-Reihenfolge unbedingt beachten (siehe „Init Task“ auf Seite 45)!*

### Exit Task

Hier muss die Task ausgewählt werden, in der der Hardware-Treiber deinitialisiert werden soll. Das ist in der Regel eine „echte“ Exit-Task (= inactive Init-Task), es könnte aber auch eine „Software“-Task sein, was aber eher ungewöhnlich ist und bei der Überprüfung eine Warnung nach sich zieht.

Eine „Exit Task“ ist immer dann vorhanden, wenn das Item mit einem Low-Level-Treiber in Verbindung steht.

#### **Hinweis**

*Task-Reihenfolge unbedingt beachten (siehe „Init Task“ auf Seite 45)!*

### Config Task

Eine „Config Task“ ist immer dann vorhanden, wenn irgendein Element im Hardware-Treiber im Runtime-Modus verändert werden kann.

Wenn eine Änderung vorgenommen werden soll, muss hier eine Alarm-Task ausgewählt werden. Die Zykluszeit der Task ist unkritisch und besagt nur, wie schnell der Treiber nach Drücken der Schaltfläche **Accept** auf die geänderten Werte reagiert. Eine niedrige Zykluszeit (< 50 ms) bedeutet aber auch eine unnötig hohe „Hintergrundlast“. In der Regel sollte man hier Tasks mit einer Zykluszeit zwischen 200 ms und 1 s vorsehen.

Soll keine Änderung vorgenommen werden, wählen Sie eine Software-Task aus, die nicht aufgerufen wird.

### Analyze Task

Die hier ausgewählte Task muss immer eine Software-Task sein.

Eine „Analyze Task“ kann von Treibern benötigt werden, die im Interrupt-Betrieb arbeiten.

#### **Hinweis**

*Diese Task muss **ausschließlich** für das Element bzw. den betreffenden Hardware-Treiber zur Verfügung stehen und darf von keinem anderen Element bzw. von Anwender-Prozessen mitbenutzt werden. Auch zwei gleichartige Elemente dürfen sich **nicht** die gleiche Task teilen!*

#### *Acknowledge Task*

---

Hier gilt dasselbe wie bei der „Analyze Task“.

#### *Device Manager Task*

---

Diese Task wird von einigen Treibern benötigt, um unabhängig vom eigentlichen Datenaustausch eine gewisse Grundversorgung des Treibers sicherzustellen (z.B. Fehlerbehandlung, Bus-Überwachung).

Hier muss in der Regel eine Timer-Task vorgegeben werden. Die Zykluszeit ist stark unterschiedlich und wird im Hilfetext des Registers bzw. in der entsprechenden Item-Dokumentation genauer spezifiziert.

#### *Version*

---

Anzeige der Versionsbezeichnung des Items, welche zur Versions-Überprüfung mit dem Low-Level-Hardware-Treibern herangezogen werden kann.

„Version“ ist standardmäßig ausgeblendet.

#### *Format*

---

Anzeige des Formats des Items, welches zur Kompatibilitäts-Überprüfung mit dem Low-Level-Hardware-Treibern herangezogen werden kann.

„Format“ ist standardmäßig ausgeblendet.

### 1.3.3 Standardoptionen im Register „Groups“

---

In diesem Abschnitt werden die im Register „Groups“ geführten Standard-Optionen beschrieben. Weitere Item-spezifische Optionen finden Sie in den entsprechenden Abschnitten bei den jeweiligen Boards.

#### *No.*

---

Ist für die Nummerierung der Zeilen zuständig, damit z.B. Fehlermeldungen zugeordnet werden können.

#### *Device*

---

Hier wird der zugehörige Item-Name angezeigt.

„Device“ ist standardmäßig ausgeblendet.

### *Group*

---

Hier wird die Bezeichnung der Signalgruppe angegeben. Dieser Name ist in der Regel editierbar, wobei der Name aber in diesem Fall ein gültiger ANSI-C Name sein muss.

#### **Hinweis**

*Alle Signalgruppen innerhalb eines Devices müssen unterschiedliche Namen besitzen!*

### *Direction*

---

Gibt die Übertragungsrichtung der betreffenden Signalgruppe an. Falls die Einstellung gesperrt ist, so ist die Übertragungsrichtung hardwaremäßig fest vorgegeben und kann nicht geändert werden (z.B. A/D-Karte immer „receive“, D/A-Karte immer „send“). Bei einigen Devices (z.B. CAN-IO) ist die Übertragungsrichtung frei wählbar.

### *Task*

---

Hier wird die Task angegeben, in der die Datenübertragung erfolgen soll. Auch die Vorgabe mehrerer Tasks ist möglich - dann wird in jeder der Tasks die Datenübertragung durchgeführt. Hier sind in der Regel Timer-Tasks oder Software-Tasks zulässig. Bei manchen Devices ist hier auch keine Angabe möglich; dann erfolgt die Datenübertragung auf andere Weise, in der Regel Interrupt-gesteuert.

#### 1.3.4 Standardoptionen im Register „Signals“

---

In diesem Abschnitt werden die im Register „Signals“ geführten Standard-Optionen beschrieben. Weitere Item-spezifische Optionen finden Sie in den entsprechenden Abschnitten bei den jeweiligen Boards.

### *No.*

---

Ist für die Nummerierung der Zeilen zuständig, damit z.B. Fehlermeldungen zugeordnet werden können.

### *Device*

---

Hier wird der zugehörige Elementname angezeigt.

„Device“ ist standardmäßig ausgeblendet.

### *Group*

---

Hier wird der zugehörige Signalgruppen-Name angezeigt. Bei manchen Devices wie z.B. „CAN-IO“ ist hier die Zuordnung des Signals zu einer Signalgruppe auch frei wählbar.

„Group“ ist standardmäßig ausgeblendet.

### *Direction*

---

Hier wird die Übertragungsrichtung der entsprechenden Signalgruppe (und damit auch des Signals) angezeigt.

„Direction“ ist standardmäßig ausgeblendet.

*Task*


---

Hier wird die zugeordnete Task angezeigt, in der die betreffende Signalgruppe übertragen wird.

„Task“ ist standardmäßig ausgeblendet.

*Signal*


---

Hier kann ein Signalname festgelegt werden. Bei einigen Devices ist die Bezeichnung der Signale auch fest und kann vom Anwender nicht verändert werden. Falls eine Eingabe möglich ist, so muss der Name den ANSI-C Richtlinien entsprechen.

**Hinweis**

*Alle Namen von Signalen, die einer Signalgruppe zugeordnet sind, müssen unterschiedlich sein!*

*Formula*


---

In dieser Spalte kann bei Bedarf ein Signal-Wert mit einer linearen Umrechnung versehen werden. Standardmäßig ist hier eine 1:1-Umrechnung gewählt (Formel  $f(\text{phys}) := \text{phys}$ ). Bei einigen Devices ist diese Formel fest vorgegeben und die Spalte gesperrt.

## 1.3.5 Standardoptionen im Register „Data“

---

In diesem Abschnitt werden die im Register „Data“ geführten Standard-Optionen beschrieben. Weitere Item-spezifische Optionen finden Sie in den entsprechenden Abschnitten bei den jeweiligen Boards.

*No.*


---

Ist für die Nummerierung der Zeilen zuständig, damit z.B. Fehlermeldungen zugeordnet werden können.

*Device*


---

Hier wird der zugehörige Item-Name angezeigt.

„Device“ ist standardmäßig ausgeblendet.

*Group*


---

Hier wird der zugehörige Signalgruppen-Name angezeigt.

„Group“ ist standardmäßig ausgeblendet.

*Direction*


---

Hier wird die Übertragungsrichtung der entsprechenden Signalgruppe (und damit auch des Signals) angezeigt.

„Direction“ ist standardmäßig ausgeblendet.

*Task*


---

Hier wird die zugeordnete Task angezeigt, in der die betreffende Signalgruppe übertragen wird.

„Task“ ist standardmäßig ausgeblendet.

### Signal

---

Hier wird der Signalname angezeigt.

### Data

---

In dieser Spalte werden die Werte des Hardwaresignals angezeigt bzw. können Werte eingegeben werden.

### Explanation

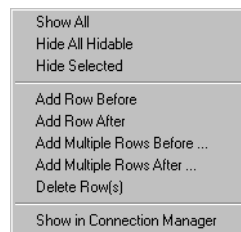
---

Diese Spalte enthält Erläuterungen zu den Signalen. Sie ist nicht bei allen Items vorhanden.

## 1.3.6 Das Kontextmenü der Register

---

Rechtsklicken in einem der Register öffnet ein Kontextmenü, über das alle oder einige der folgenden Menüpunkte erreichbar sind.



**Abb. 1-4** Das Kontextmenü der Register

- **Show All** (E, R)

Dieser Menüpunkt blendet alle darstellbaren Optionen/Spalten einer Registerkarte ein.

Die Tabelle einer Registerkarte kann so gestaltet sein, dass einige der möglichen Optionen bzw. Spalten standardmäßig nicht eingeblendet sind. Dies hat den Vorteil, dass wenig benutzte oder selten gebrauchte Optionen ausgeblendet und so die Tabelle übersichtlicher gestalten werden kann.

#### Hinweis

*Wird anschließend die Tabelle mit der Schaltfläche **Accept** bestätigt und die Hardwarekonfiguration gesichert, steht nach dem erneuten Laden die Tabelle wieder so zur Verfügung, wie sie gesichert wurde, d.h. neu eingeblendete Optionen/Spalten bleiben eingeblendet.*

- **Hide All Hidable** (E, R)

Bildet das Gegenstück zum vorigen Menüpunkt, d.h. alle ausblendbaren Optionen/Spalten werden ausgeblendet.

- **Hide Selected** (E, R)

Blendet die aktuell gewählte Option bzw. Spalte aus, sofern sie ausblendbar ist.

- **Add Row Before** (E)

Dieser Menüpunkt ist nur für Items des Typs „Device“ verfügbar, sofern sie die Möglichkeit bieten, ihre Signalgruppen- bzw. Signal-Anzahl dynamisch zu ändern.

Unter folgenden Voraussetzungen ist der Menüpunkt verfügbar und wählbar:

1. Es ist ein Item vom Typ „Device“ ausgewählt, das über „dynamische“ Eigenschaften verfügt (z.B. CAN-IO Device).
2. Das Register „Groups“ oder „Signals“ ist ausgewählt.
3. Ein Eintrag der Spalte „No.“ ist ausgewählt (1. Spalte in der Tabelle).

Dieser Menüpunkt fügt eine Zeile vor der gewählten Zelle bzw. dem gewählten Bereich ein.

#### Hinweis

*Systembedingt ist die Größe der Tabelle oft beschränkt. In diesem Fall bleibt das Ausführen dieser Funktion wirkungslos. Da es sich hier um eine strukturelle Änderung der Tabelle handelt, lässt sich diese Änderung nicht mehr durch die Schaltfläche **Reset** wieder rückgängig machen.*

- **Add Row After**

Funktioniert wie **Add Row Before**, nur wird hier eine neue Zeile nach der gewählten Zelle bzw. dem gewählten Bereich eingefügt.

- **Add Multiple Rows Before**

Funktioniert wie **Add Row Before**, nur können hier gleich mehrere Zeilen eingefügt werden. Die Anzahl der neu einzufügenden Zeilen kann in einem Dialogfenster eingegeben werden:

- **Add Multiple Rows After**

Funktioniert wie unter **Add Multiple Rows Before** beschrieben, nur werden hier neue Zeilen nach der gewählten Zelle bzw. dem gewählten Bereich eingefügt.

- **Delete Row(s)**

Löscht die Zeile bzw. den Bereich, der in der Spalte „No.“ markiert ist.

#### Hinweis

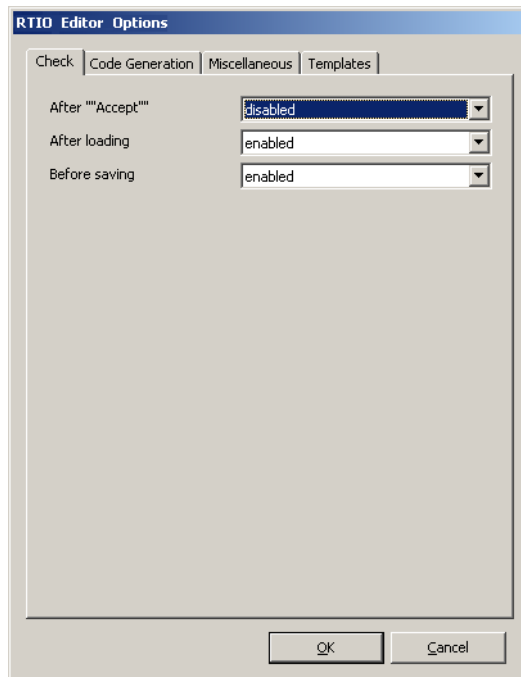
*Aus Systemgründen muss mindestens eine Signalgruppe bzw. ein Signal in der Tabelle verbleiben. Beim Versuch, die letzte Reihe zu löschen, erscheint eine Fehlermeldung.*

- **Show in Connection Manager**

Bei Auswahl dieses Menüpunktes wird der Connection Manager geöffnet. Es wird das ausgewählte Hardwaresignal angezeigt (in der Liste „Inputs“ oder „Outputs“) und eine eventuell vorhandene Verbindungen zu anderen (Modell- oder Hardware-) Ein- oder Ausgängen dargestellt.

### 1.3.7 Optionen

Über den Menüpunkt **Extras** → **Options...** gelangen Sie zu einem Fenster mit mehreren Registern, in denen eine Reihe von Optionen eingestellt werden können.



**Abb. 1-5** Das Fenster „RTIO Editor Options“  
Register „Check“

Hier kann festgelegt werden, bei welchen Gelegenheiten automatische Überprüfungen der Hardwarekonfiguration bzw. der Datenkonsistenz durchgeführt werden.

- After „Accept“  
Aktiviert („enabled“) bzw. deaktiviert („disabled“) die automatische Überprüfung nach Drücken der Schaltfläche **Accept**.  
Aus Performancegründen wird hier nur die Datenkonsistenz innerhalb der gerade ausgewählten Registerkarte überprüft.
- After loading: enabled/disabled  
Aktiviert („enabled“) bzw. deaktiviert („disabled“) die Überprüfung nach dem Laden einer neuen Hardwarekonfiguration.
- Before saving: enabled/disabled  
Aktiviert („enabled“) bzw. deaktiviert („disabled“) die automatische Überprüfung der Hardwarekonfiguration vor dem Sichern einer Konfiguration.

#### Register „Code Generation“

Hier können bestimmte Optionen für die Codegenerierung eingestellt werden.



- Automatic repair  
Aktiviert („yes“) bzw. deaktiviert („no“) die Möglichkeit, bestimmte Fehler in der Hardwarekonfiguration automatisch zu beheben. Diese Möglichkeit wird aber zur Zeit nur sehr eingeschränkt unterstützt.
- Minutes type  
Legt den Umfang des Protokolls fest („no/small/detailed“), das bei der RTIO-Codegenerierung erzeugt wird.

#### *Register „Miscellaneous“*

---

Die beiden Optionen in diesem Register legen das Format fest, in dem die Werte in der Spalte „Data“ in den „Data“-Registerkarten der Devices angezeigt werden.

- Data digits after decimal point: 1...6
- Data digits before decimal point: 1...9

Darstellung: <Data digits before...>.<Data digits after...>

#### *Register „Templates“*

---

In diesem Register werden Optionen zu den Item-Templates festgelegt.

Item-Templates dienen dazu, die vom System vorgegebenen Default-Werte mit eigenen Einstellungen zu überschreiben. Dazu werden bei aktivierter Template-Option jedesmal beim Einfügen von Items (Menü **Edit** → **Add Item...**) die vom System vorgegebenen Standardwerte durch diejenigen überschrieben, die in dem Template-Verzeichnis gesichert wurden (Menü **File** → **Save Item As Template...**).

- Template Path  
Hier kann das Verzeichnis gewählt werden, das als Ablage für die Item-Templates verwendet wird.
- Use Item Templates  
Aktiviert („yes“) bzw. deaktiviert („no“) die Verwendung von Item-Templates.



## 2 **ES1220.1 CAN Board**

---

Das ES1220.1 CAN Board dient als CAN-Schnittstelle in VMEbus-Systemen. Die Einschubkarte verfügt über einen digitalen Signalprozessor (DSP), der unabhängig vom Hauptprozessor des VME-Systems arbeitet. Die Karte stellt vier CAN-Kanäle zur Verfügung, die über getrennte CAN-Controller verfügen. Jeder der vier CAN-Controller vom Typ Intel 82527 besitzt 15 Puffer zur Aufnahme von eingehenden oder zu versendenden CAN-Nachrichten. 14 dieser Puffer können sowohl Sende- als auch Empfangsnachrichten verarbeiten. Der 15. Puffer kann ausschließlich Empfangsnachrichten verarbeiten. Durch den Einsatz eines DSP lassen sich hohe Datenraten und kurze Antwortzeiten auf eingehende CAN-Nachrichten realisieren.

Die Karte umfasst unter anderen folgende Funktionen:

- Digitaler Signalprozessor TMS320F206
- Vier CAN-Controller vom Typ Intel 82527
- 15 Nachrichtenpuffer pro CAN-Controller (14 Sende- & Empfangspuffer, 1 Empfangspuffer)

Die ES1220.1 kann in VMEbus-Systemen überall dort eingesetzt werden, wo CAN-Nachrichten erzeugt und empfangen werden sollen. Der digitale Signalprozessor (DSP) ermöglicht eine flexible Nachrichtenverarbeitung unabhängig vom Systemprozessor.

Beispielhafte Einsatzgebiete sind:

- Simulation von Fahrzeugbaugruppen CAN-Schnittstelle
- Simulation des Fahrzeug-CAN-Bus
- Steuerung von Lastnachbildungen

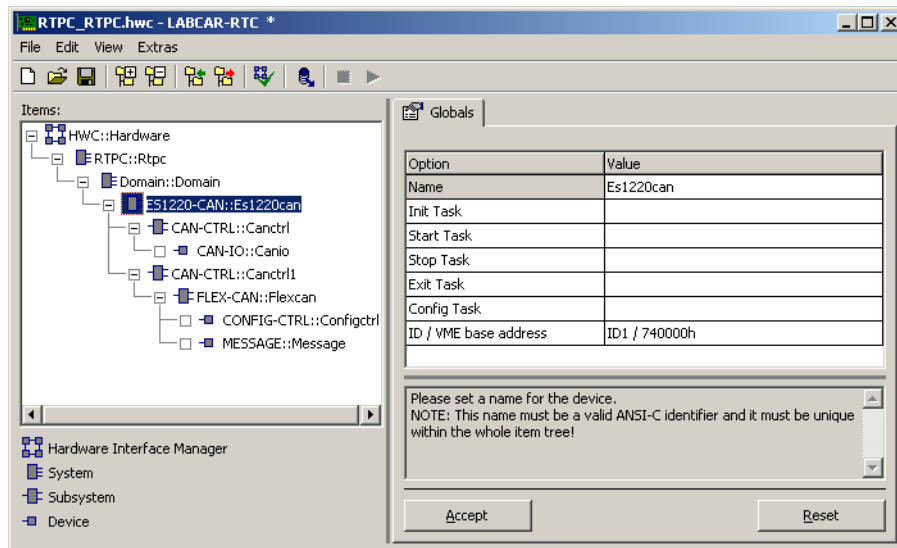
Dieses Dokument beschreibt die RTIO-Einbindung der ES1220.1.

Im RTIO-Editor wird das ES1220.1 CAN Board durch Auswahl des ES1220-CAN Subsystems eingebunden.

## 2.1 ES1220-CAN Subsystem

### 2.1.1 Globals (ES1220-CAN Subsystem)

In der Registerkarte „Globals“ des ES1220-CAN Subsystems kann der Anwender im Auswahlfeld „ID / VME base address“ die VMEbus-Basisadresse seiner ES1220.1 angeben (siehe Abb. 2-1). Es stehen 3 unterschiedliche Basisadressen zur Auswahl (ID1/740000h, ID2/780000h und ID3/7C0000h).



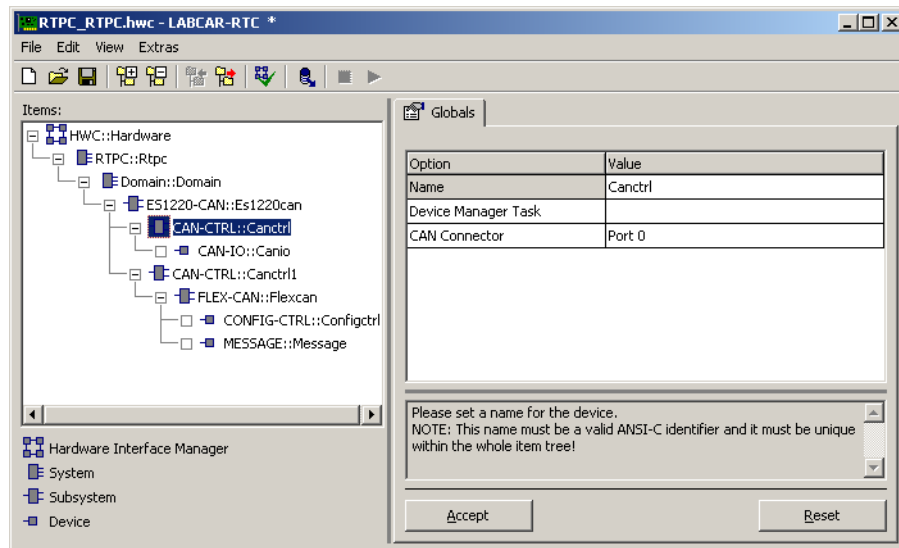
**Abb. 2-1** Die Registerkarte „Globals“ des ES1220-CAN Subsystems

Dem ES1220-CAN Subsystem können bis zu vier CAN-CTRL Subsysteme zugeordnet werden. Diese CAN-CTRL Subsysteme entsprechen den vier auf der Karte vorhandenen Intel 82527 CAN-Controllern.

## 2.2 CAN-CTRL Subsystem

### 2.2.1 Globals (CAN-CTRL Subsystem)

In der Registerkarte „Globals“ eines CAN-CTRL Subsystems (siehe Abb. 2-2) wird dem CAN-CTRL Subsystem ein physikalischer CAN-Controller bzw. CAN-Anschluss zugeordnet. Die CAN-Anschlüsse auf der Frontplatte werden von oben nach unten mit 1 bis 4 durchnummeriert.



**Abb. 2-2** Die Registerkarte „Globals“ eines CAN-CTRL Subsystems

#### *Device Manager Task*

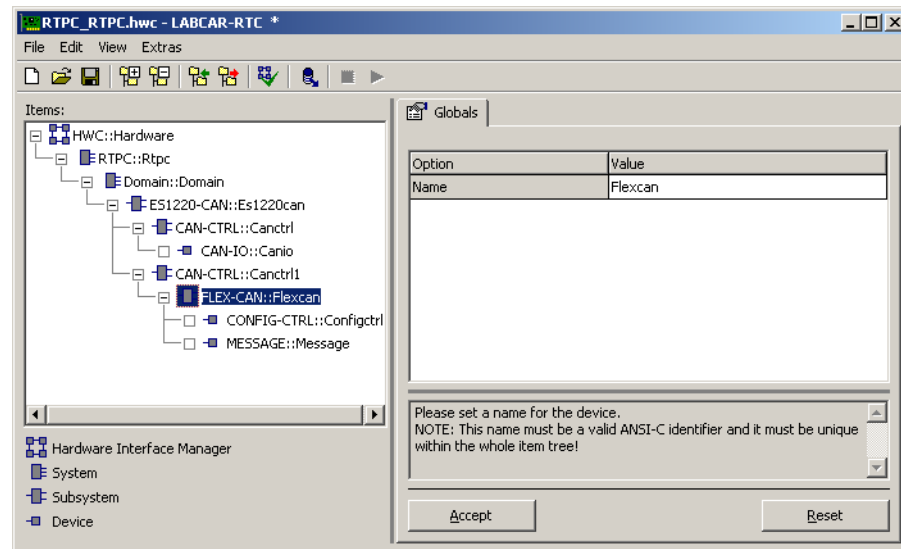
Die Device Manager API Funktion dient der Behandlung von „Bus-Off“-Ereignissen und deren Behebung. Der Device Manager Prozess sollte einer periodisch aktivierten Task zugewiesen werden - eine „task activation period“ von 100 ms oder sogar länger ist hinreichend.

Die Zuordnung einer Task ist optional. Wenn keine Task zugeordnet wird, findet kein „Bus-Off handling“ statt, aber Codegenerierung ist trotzdem möglich.

Jeder der 4 CAN-Controller der ES1220.1 kann für zwei unterschiedliche Anwendungen konfiguriert werden.

Die sogenannte FlexCan-Anwendung erlaubt die Nutzung aller Softwareoptionen, die ein CAN-Controller bietet. So ist z.B. die Änderung der Nachrichten-Id, des Frame-Typs und der Übertragungsrichtung eines Nachrichtenpuffers zur Laufzeit in Echtzeit möglich. Weitere Einstellungen wie z.B. die Baudrate und die Akzeptanzfilter können ebenfalls zur Laufzeit konfiguriert werden.

Die FlexCan-Anwendung wird ausgewählt, indem man dem CAN-CTRL Subsystem ein FLEX-CAN Subsystem zuordnet (siehe Abb. 2-3). Alle Einstellungen eines FLEX-CAN Subsystems können über RTIO-Signale vom Simulationsmodell aus gesteuert werden.



**Abb. 2-3** Der ES1220-Item-Baum mit FLEX-CAN Subsystem und CAN-IO Device

## 2.3 FLEX-CAN Subsystem

Dem FLEX-CAN Subsystem können ein CONFIG-CTRL Device und bis zu 15 MESSAGE Devices zugeordnet werden. Jedes der 15 MESSAGE Devices dient zur Konfiguration eines der 15 physikalisch auf einem CAN-Controller vorhandenen CAN-Nachrichtenpuffer. Das CONFIG-CTRL Device dient zur Konfiguration der restlichen Parameter eines CAN-Controllers. Diese Parameter wirken sich auf das Verhalten aller 15 Nachrichtenpuffer aus.

### Hinweis

*Bevor einem FLEX-CAN Subsystem ein MESSAGE Device zugeordnet werden kann, muss zuerst ein CONFIG-CTRL Device eingefügt werden.*

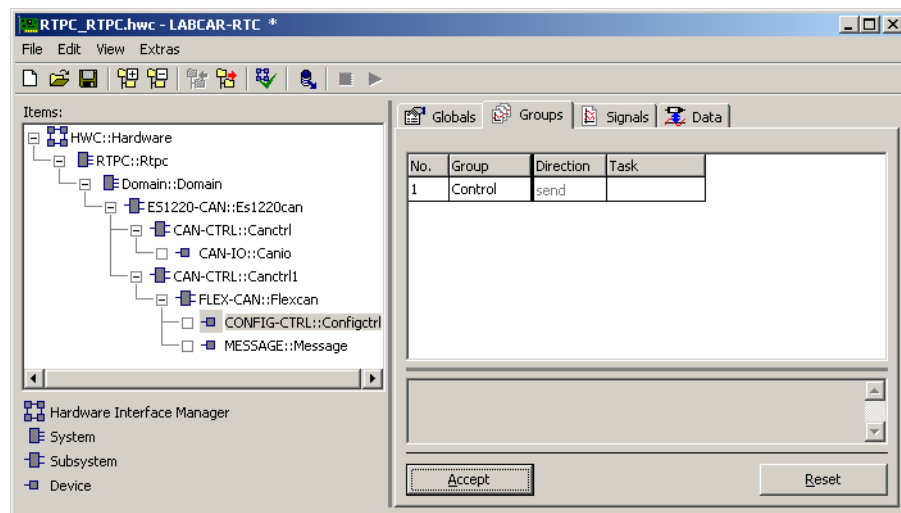
## 2.4 CONFIG-CTRL Device

### 2.4.1 Groups (CONFIG-CTRL Device)

Das CONFIG-CTRL Device besitzt eine Signalgruppe mit der Bezeichnung „Control“, mit der die globalen, für alle 15 Nachrichtenpuffer eines CAN-Controllers gültigen Einstellungen konfiguriert werden. Diese Konfigurationsparameter sind:

- Baudrate
- globale Akzeptanzmaske für CAN-Nachrichten im „Standard Format“
- globale Akzeptanzmaske für CAN-Nachrichten im „Extended Format“
- lokale Akzeptanzmaske für Nachrichtenpuffer 15

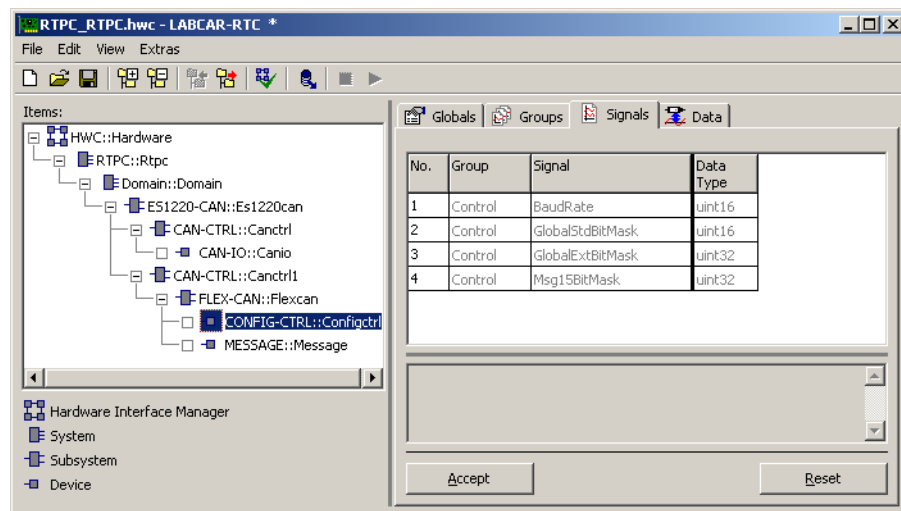
Im Gegensatz zur sonst üblichen Vorgehensweise kann die Berechnung der „Control“ Signalgruppe in der Registerkarte „Groups“ keiner Task des Echtzeitbetriebsystems zugeordnet werden (siehe Abb. 2-4). Die Berechnung der „Control“ Signalgruppe eines CONFIG-CTRL Devices erfolgt implizit in den „Start“ und „Config“ Prozessen des RTIO-Treibers. Diese Prozesse werden in der Registerkarte „Globals“ des ES1220-CAN Subsystems (siehe Abb. 2-1 auf Seite 56) Tasks des Echtzeitbetriebsystems zugeordnet.



**Abb. 2-4** Die Registerkarte „Groups“ des CONFIG-CTRL Devices

2.4.2 Signals (CONFIG-CTRL Device)

Abb. 2-5 zeigt die Registerkarte „Signals“ des CONFIG-CTRL Devices.



**Abb. 2-5** Die Registerkarte „Signals“ des CONFIG-CTRL Devices

Der Empfangszweig eines Intel 82527 CAN-Controllers enthält zwei globale Filtermasken, über die gekennzeichnet werden kann, welche der in den Nachrichtepuffern eingetragenen Identifier-Bits mit denen einer empfangenen CAN-Nachricht übereinstimmen müssen, damit diese in den zugehörigen Nachrichtepuffer übernommen wird.

Diese Masken bestimmen somit, welche Bits für die Nachrichtenfilterung relevant sind. Auf diese Weise lassen sich die einzelnen Puffer entweder auf den Empfang je eines einzigen Identifiers beschränken oder für den Empfang von Identifiergruppen erweitern. Eine der beiden globalen Filtermasken, die mit dem RTIO-Signal *GlobalStdBitMask* konfiguriert wird, ist für die Filterung von CAN-Nachrichten mit Standard (11 Bit) Identifiern zuständig. Die zweite globale Filtermaske, die mit dem RTIO-Signal *GlobalExtBitMask* konfiguriert wird, ist für die Filterung von CAN-Nachrichten mit Extended (29 Bit) Identifiern zuständig. Dem Nachrichtenpuffer 15 (der nur für den Empfang, aber nicht für das Senden von CAN-Nachrichten konfiguriert werden kann) ist eine weitere, lokale Filtermaske vorgeschaltet. Diese Filtermaske wird mit dem RTIO-Signal *Msg15BitMask* konfiguriert. Die relevanten Identifier-Bits werden durch eine „1“ in den Filtermasken gekennzeichnet, die irrelevanten durch eine „0“.

Im Empfangsfall wird zunächst unter Berücksichtigung der globalen Filtermasken geprüft, ob die vorliegende Nachricht in einen der ersten 14 Nachrichtenpuffer übernommen werden kann. Ist dies nicht der Fall, so wird die globale Maske des Nachrichtenformats („Standard“ oder „Extended“) mit der lokalen Maske von Nachrichtenpuffer 15 bitweise UND-verknüpft und auf der Basis der dann resultierenden Filtermaske geprüft, ob die Nachricht in diesen Nachrichtenpuffer übernommen werden darf. Ist dies nicht möglich, so wird die Nachricht verworfen.

Werden z.B alle Nachrichtenpuffer als Empfangspuffer konfiguriert, so erlaubt es der Filtermechanismus, dass in jeden dieser Puffer nur jeweils eine bestimmte Nachricht übernommen wird, während alle anderen Nachrichten in den Nachrichtenpuffer 15 übernommen werden. Hierzu müssen in den beiden globalen Filtermasken alle Bits als relevant (Bitwert 1) definiert werden und in der lokalen Filtermaske für Nachrichtenpuffer 15 alle Bits als nicht relevant (Bitwert 0).

In Tab. 2-1 sind die RTIO-Signale des CONFIG-CTRL Devices zusammengefasst.

#### **Hinweis**

*Wie in „Groups (CONFIG-CTRL Device)“ auf Seite 58 beschrieben, wird die Signalgruppe „Control“ des CONFIG-CTRL Devices implizit in den „Start“ und „Config“ Prozessen des RTIO-Treibers verarbeitet. Daher muss durch den Anwender sichergestellt werden, dass die RTIO-Signale der „Control“ Signalgruppe vor der ersten Verarbeitung eines „Start“ oder „Config“ Prozesses mit zulässigen Werten initialisiert wurden.*



RTIO-Signal	Datentyp	Beschreibung
BaudRate	uint16	Datenübertragungsrate in kBaud Zulässige Werte: 10, 20, 50, 100, 125, 250, 500, 1000
GlobalStdBitMask	uint16	Globale Akzeptanz-Bitmaske für CAN-Nachrichten im „Standard Format“. Die untersten bzw. niederwertigsten 11 Bit werden verarbeitet. Zulässige Werte: 0x0000 ... 0x07FF Bitwert 0: Don't Care. Es wird eine 0 oder 1 auf dieser Bitposition akzeptiert. Bitwert 1: Must Match. Der Bitwert einer empfangenen CAN-Nachrichten muss in dieser Bitposition exakt mit dem Bitwert des Identifiers eines Nachrichtenpuffers übereinstimmen
GlobalExtBitMask	uint32	Globale Akzeptanz-Bitmaske für CAN-Nachrichten im „Extended Format“. Die untersten bzw. niederwertigsten 29 Bit werden verarbeitet. Zulässige Werte: 0x0000 0000 ... 0x1FFF FFFF Bitwert 0: Don't Care Bitwert 1: Must Match
Msg15BitMask	uint32	Lokale Akzeptanz-Bitmaske für Nachrichtenpuffer 15. Die untersten bzw. niederwertigsten 29 Bit werden verarbeitet. Zulässige Werte: 0x0000 0000 ... 0x1FFF FFFF Bitwert 0: Don't care Bitwert 1: Must match <b>Achtung:</b> Die lokale Akzeptanz-Bitmaske für Nachrichtenpuffer 15 wird mit den globalen Akzeptanz-Bitmasken bitweise UND-verknüpft. Das bedeutet, dass jedes Bit das in der globalen Akzeptanz-Bitmaske des CAN-Nachrichtenformat als „don't care“ definiert ist automatisch auch ein „don't care“ Bit in der Akzeptanz-Bitmaske für Nachrichtenpuffer 15 ist.

**Tab. 2-1** Die RTIO-Signale des CONFIG-CTRL Devices

#### Hinweis

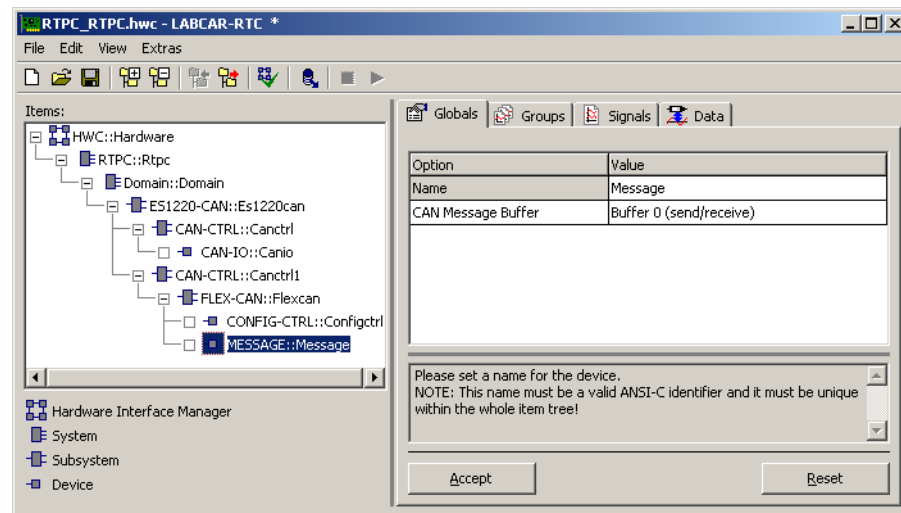
*Wenn in diesem Dokument Datentypen angegeben werden, so sind das die Datentypen, die der RTIO-Treiber intern verwendet.*

## 2.5 MESSAGE Device

### 2.5.1 Globals (MESSAGE Device)

Das MESSAGE Device dient zur Konfiguration eines Nachrichtenpuffers des Intel 82527 CAN-Controllers.

In der Registerkarte „Globals“ eines MESSAGE Devices (siehe Abb. 2-6) wird im Auswahlfeld „CAN Message Buffer“ das MESSAGE Device einem physikalischen Nachrichtenpuffer des CAN-Controllers zugeordnet.

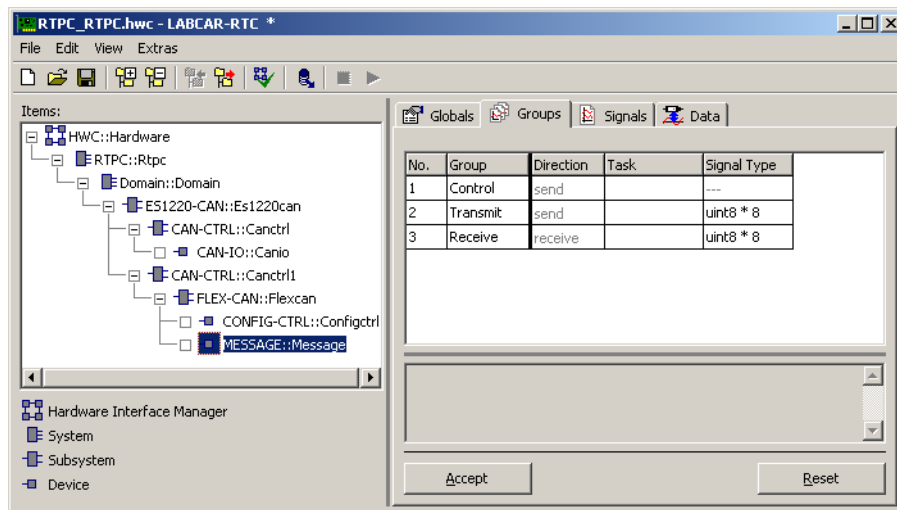


**Abb. 2-6** Die Registerkarte „Globals“ eines MESSAGE Devices

## 2.5.2 Groups (MESSAGE Device)

Ein MESSAGE Device besitzt drei Signalgruppen (siehe Abb. 2-7):

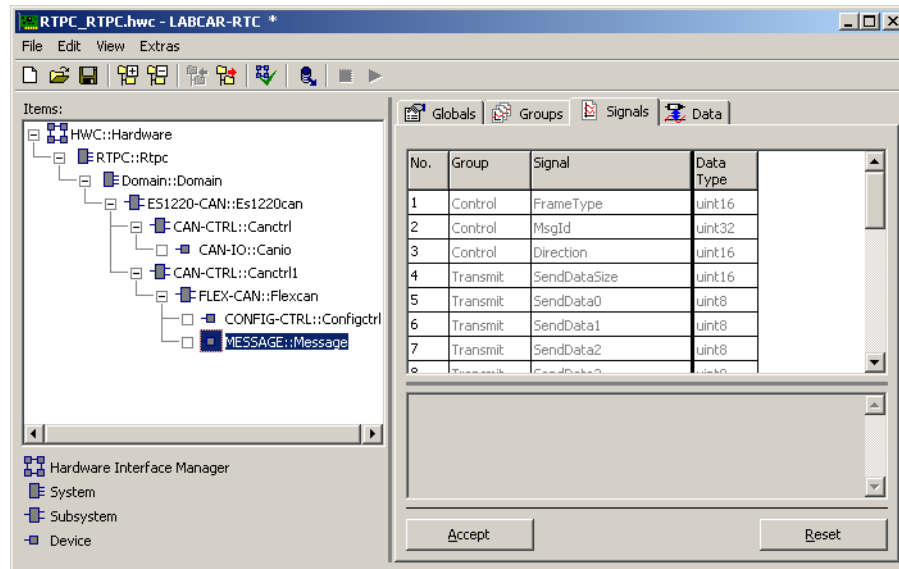
- Control  
Die „Control“ Signalgruppe wird vom Simulationstarget zur ES1220.1 übertragen. Diese Signalgruppe enthält alle Signale, die sowohl das Sende- als auch das Empfangsverhalten eines Nachrichtenpuffer betreffen.
- Transmit  
Die „Transmit“ Signalgruppe wird ebenfalls vom Simulationstarget zur ES1220.1 übertragen. Diese Signalgruppe enthält alle Signale, die das Sendeverhalten eines Nachrichtenpuffers konfigurieren.
- Receive  
Die „Receive“ Signalgruppe wird von der ES1220.1 zum Simulationstarget übertragen und enthält alle Signale, die die Daten einer empfangenen CAN-Nachricht beschreiben.



**Abb. 2-7** Die Registerkarte „Groups“ eines MESSAGE Devices

## 2.5.3 Signals (MESSAGE Device)

Abb. 2-8 zeigt die Registerkarte „Signals“ eines MESSAGE Devices.



**Abb. 2-8** Die Registerkarte „Signals“ eines MESSAGE Devices

#### *FrameType*

Mit dem Signal *FrameType* wird das Format (Standard oder Extended) der CAN-Nachrichten festgelegt, die im Sendebetrieb vom Puffer gesendet werden und im Empfangsbetrieb vom Nachrichtenpuffer empfangen werden.

#### *MsgId*

Ist der Nachrichtenpuffer für einen Sendebetrieb konfiguriert, so wird mit dem Signal *MsgId* der Identifier einer zu versendenden Nachricht definiert. Ist der Nachrichtenpuffer für einen Empfangsbetrieb konfiguriert, so wird mit dem Signal *MsgId* der Identifier der CAN-Nachrichten definiert, die in den Nachrichtenpuffer übernommen werden. Welche Nachrichten im Empfangsbetrieb in einen Nachrichtenpuffer übernommen werden hängt zusätzlich noch von den Einstellungen der globalen und lokalen Akzeptanz-Bit-Masken (siehe Abschnitt 2.4.2 auf Seite 59) ab.

#### *Direction*

Das *Direction* Signal konfiguriert die Übertragungsrichtung des zugehörigen CAN-Nachrichtenpuffers. Ein Puffer kann entweder für Sendebetrieb oder Empfangsbetrieb konfiguriert werden.

#### *SendDataSize und SendData<sub>x</sub>*

Ist der CAN-Nachrichtenpuffer für einen Sendebetrieb konfiguriert, so legen das Signal *SendDataSize* die Anzahl und die Signale *SendData<sub>x</sub>* ( $x = 0, 1, \dots, 7$ ) die Inhalte der zu versendenden Datenbytes fest. Im Empfangsbetrieb sind diese Signale ohne Bedeutung.

Die Signale *RecDataNewMes*, *RecDataSize*, *RecDataFrameType*, *RecDataMsgId* und *RecData<sub>x</sub>* ( $x = 0, 1, \dots, 7$ ) sind nur für den Empfangsbetrieb eines Puffers relevant.

#### *RecDataNewMes*

---

Das Signal *RecDataNewMes* gibt an, ob seit dem letzten Lesezugriff auf den CAN-Nachrichtenpuffer eine neue CAN-Nachricht empfangen wurde.

#### *RecDataSize*

---

Das Signal *RecDataSize* gibt die Anzahl der Datenbytes in der empfangenen CAN-Nachricht an.

#### *RecDataFrameType*

---

Das Signal *RecDataFrameType* gibt das Nachrichtenformat der empfangenen CAN-Nachricht an. Diese Information ist insofern redundant, als dass sie immer mit dem eingestellten Wert des Signals *FrameType* übereinstimmen muss.

#### *RecDataMsgId*

---

Das Signal *RecDataMsgId* gibt den Identifier der empfangenen CAN-Nachricht an. Der Wert von *RecDataMsgId* muss nicht zwingend mit dem eingestellten Wert für *MsgId* übereinstimmen. Das muss nur dann der Fall sein, wenn in den Akzeptanz-Bit-Masken (siehe Abschnitt 2.4.2 auf Seite 59) sämtliche Bits auf „Must Match“ eingestellt sind.

#### *RecDataax*

---

Die Signale *RecDataax* enthalten die Werte der Datenbytes der empfangenen CAN-Nachricht.

Tab. 2-2 enthält die Eigenschaften der RTIO-Signale des MESSAGE Device.

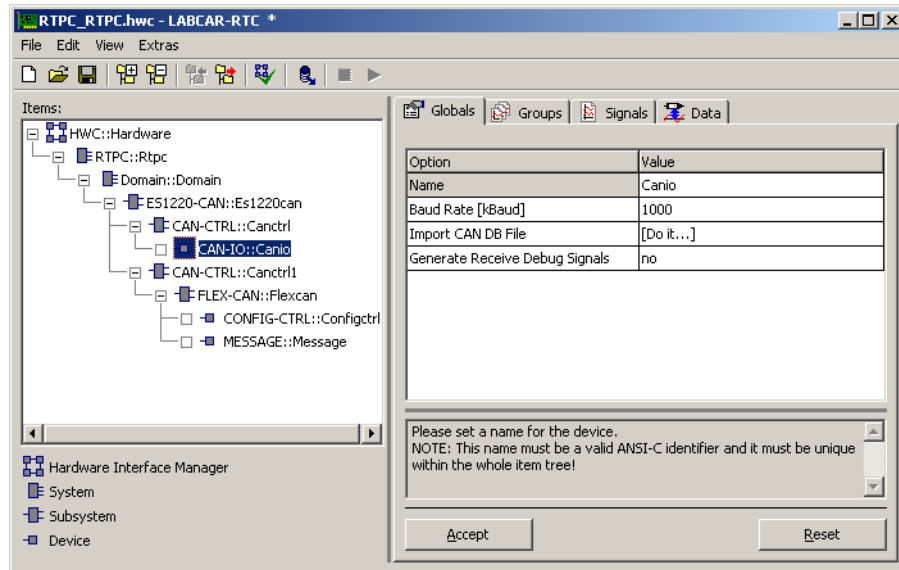
RTIO-Signal	Datentyp	Signalgruppe	Beschreibung
FrameType	uint16	Control	Nachrichtenformat; 0: Standard-Format (11-Bit-Identifizier) 1: Extended-Format (29-Bit-Identifizier)
MsgId	uint32	Control	Nachricht-Identifizier
Direction	uint16	Control	Übertragungsrichtung; 0: Empfang 1: Senden
SendDataSize	uint16	Transmit	Länge der Sendenachricht in Bytes; Zulässige Werte: 0 ... 8
SendData0 ... SendData7	uint8	Transmit	Datenbytes
RecData- NewMes	uint16	Transmit	Indikator für den Empfang einer neuen CAN-Nachricht; 0: Es wurde keine neue CAN-Nach- richt seit dem letzten Lesezugriff auf den Nachrichtenpuffer empfangen 1: Neue CAN-Nachricht empfangen
RecDataSize	uint16	Receive	Anzahl der empfangenen Datenbytes; Wertebereich: 0 ... 8
RecDataFrame- Type	uint16	Receive	Format der empfangenen CAN-Nach- richt 0: Standard-Format (11-Bit-Identifizier) 1: Extended-Format (29-Bit-Identifizier)
RecDataMsgId	uint32	Receive	Identifizier der empfangenen CAN- Nachricht
RecData0 ... RecData7	uint8	Receive	Empfangene Datenbytes

**Tab. 2-2** Die Signale des MESSAGE Device

## 2.6 CAN-IO Device

### 2.6.1 Globals (CAN-IO Device)

Das ES1220-CAN-IO Device bietet dem Anwender eine Benutzeroberfläche, mit der die CAN-Botschaften und -Signale sehr einfach spezifiziert werden können. Sie basiert im Prinzip auch auf der Flex-CAN Einbindung, nur wird hier zur Vereinfachung auf die Laufzeit-Konfigurierbarkeit verzichtet.



**Abb. 2-9** Die Registerkarte „Globals“ des CAN-IO Devices

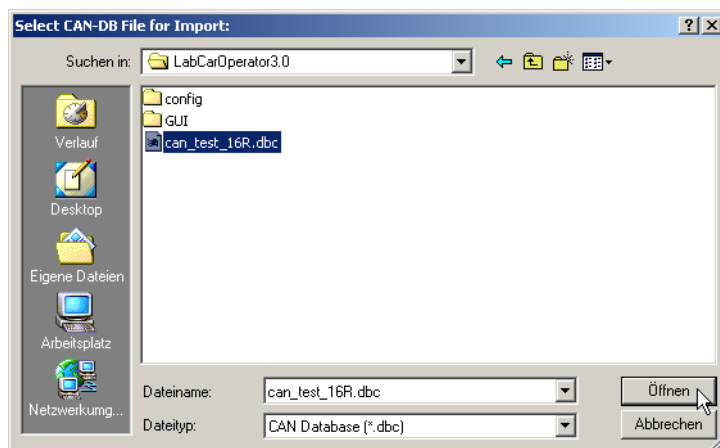
#### *Baud Rate [kBaud]*

Hier kann die gewünschte Baud Rate ausgewählt werden. Es stehen hierfür die Standard-Baudraten (1000, 500, 250, 125, 100, 50, 20, 10 kBaud) zur Verfügung. Im Gegensatz zum ES1222 CAN-IO-Item besteht hier nicht die Möglichkeit, eine <special Timing> Einstellung vorzunehmen.

#### *Import CAN DB File*

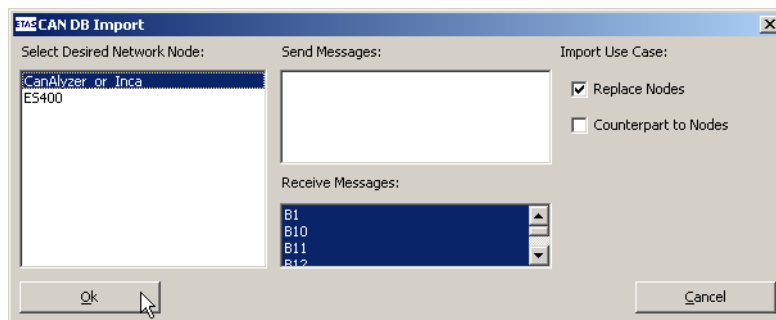
Über diese Option kann eine CAN Datenbank-Datei eingelesen werden, welche mit dem CANdb Datenverwaltungsprogramm der Firma Vector Informatik erstellt wurde. Mit Hilfe dieser Datei können nun bei Bedarf automatisch CAN-Botschaften und -Signale angelegt werden.

Nach Drücken von [Do it...] öffnet sich folgendes Dialogfenster:



In diesem Dialogfenster kann die gewünschte zu importierende CAN DB-Datei ausgewählt werden.

Nach Betätigen der Schaltfläche **Open** öffnet sich nachfolgendes Dialogfenster, indem das weitere Verfahren spezifiziert werden kann:



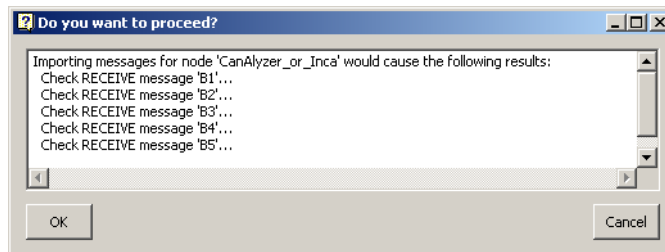
**Abb. 2-10** CAN DB Import Dialog

In einer CAN DB Datei werden üblicherweise mehrere Knoten eines CAN Netzwerkes beschrieben. Alle vorhandenen Knoten werden in der linken Liste (Network Nodes) aufgelistet. Die beiden Listen rechts davon (Send Messages / Receive Messages) listen alle für den aktuell selektierten Knoten definierten CAN Botschaften auf. Die Botschaften, die selektiert sind, werden für den Import herangezogen. Zusätzlich kann der „Import Use Case“ noch ausgewählt werden. Die Option **Replace Nodes** bedeutet, dass das CAN-IO Device die Rolle des Netzwerkknotens übernimmt; d.h. eine Sendebotschaft des Knotens wird auch als Sendebotschaft des CAN-IO Devices umgesetzt usw.

Die Option **Counterpart to Nodes** bedeutet, dass das CAN-IO Device das Gegenstück zum Netzwerkknoten bildet; d.h. eine Sendebotschaft des Knotens ruft eine Receive-Botschaft hervor usw.



Nach Drücken der Schaltfläche **OK** erfolgt eine Überprüfung der zu importierenden Daten (CAN-Messages und -Signale) mit den vorhandenen Signalgruppen und Signalen. Das Resultat dieser Überprüfung wird anschließend in dem nachfolgenden Dialogfenster angezeigt:

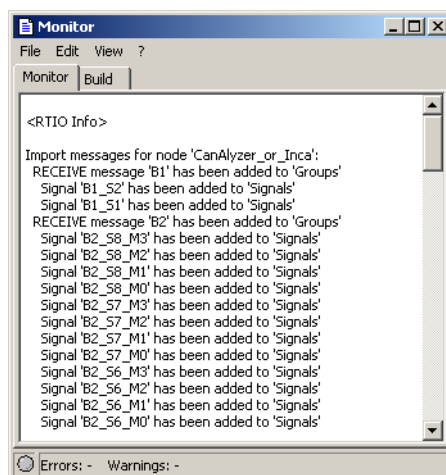


**Abb. 2-11** Dialogfenster „CAN DB Check“

Bis jetzt ist noch nichts an dem vorhandenen CAN-IO Device verändert worden. Erst nach Drücken der Schaltfläche **OK** wird der tatsächliche Importvorgang gestartet.

Der Import-Vorgang fügt die importierten Messages den Signalgruppen hinzu und sorgt auch für die Definition der entsprechenden Signale.

Nach Beendigung des Imports erscheint im Fenster „Monitor“ das detaillierte Protokoll des Importvorgangs:



**Abb. 2-12** Fenster „Monitor“ mit Importprotokoll

### Beim Import bitte beachten :

Beim Importieren einer CAN DB-Datei werden nur 29 Bit (Identifier *extended*) bzw. 11 Bit (Identifier *standard*; siehe „Identifier Type“ auf Seite 71) automatisch in das Identifier-Feld eingefügt.

Wenn im CAN-Controller Item der Identifier *standard* gewählt wurde, die CAN DB-Datei aber Signale mit 29 Bit-Identifiern ( $ID > 2^{31}$ ) enthält, geschieht folgendes:

- 11 Bit (Bits [28..18]) werden automatisch in das Identifier-Feld geschrieben. Die übrigen Bits (MSB) werden verworfen.
- Im Monitorfenster werden Warnungen angezeigt.

Da nur ein Identifier ausgewählt werden kann, kommt es bei CAN DB-Dateien, die sowohl *standard* als auch *extended* Identifier enthalten, zu Konflikten.

### Generate Receive Debug Signals

Ist diese Option eingeschaltet (= `yes`), so wird für jede „receive“-Signalgruppe zwei zusätzliche Signale generiert.

- `<GroupName>_Diag_dT`
- `<GroupName>_Diag_Rec`

Das Signal `...dT` gibt die Differenz in Sekunden zur vorangegangenen empfangenen Botschaft an.

#### **Hinweis**

*Dieses Signal kann beim normalen Botschaftsempfang (IRQ = no) verwendet werden, um den Empfang zu überwachen. Wird z.B. der CANbus unterbrochen, so erhöht sich der Wert laufend im Raster der Empfangstask. Aus Performance-Gründen ist dieser Wert nicht gegen Überlauf geschützt, der bei ca. 300 s auftritt!*

Im Gegensatz hierzu kann beim Botschaftsempfang mit Interrupt (IRQ = `yes`) keine echte Empfangsüberwachung durchgeführt werden, da die Berechnung des Wertes erst in der Interrupt-Task erfolgt. Da beim Ausbleiben des Empfangs diese Berechnung auch nicht stattfindet, bleibt somit der alte Wert eingefroren. Damit kann beim Interrupt-Empfang nicht unterschieden werden, ob der Empfang in einem exakt regelmäßigen Raster erfolgt oder unterbrochen ist.

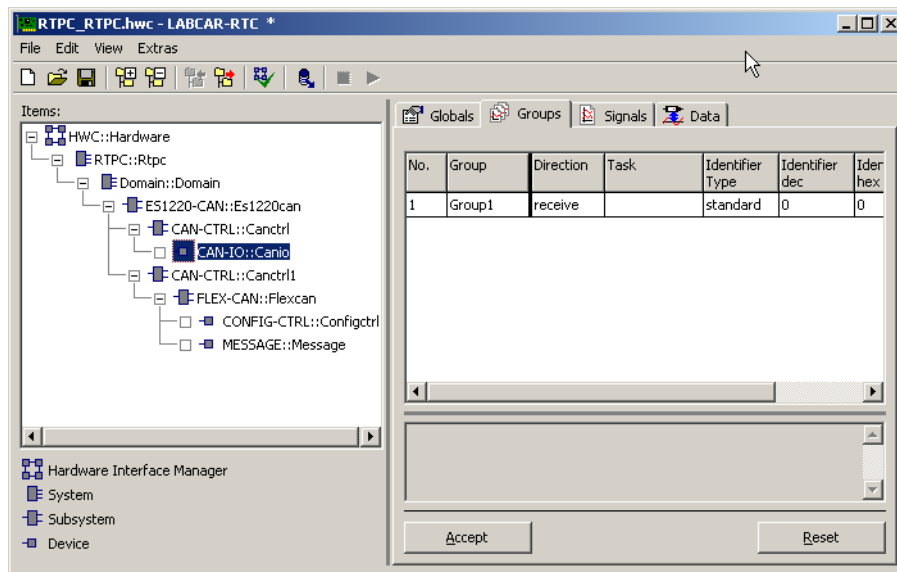
Das Signal `...Rec` wird bei jedem Botschaftsempfang mit dem Wert `true` beschrieben. Wird dieses Signal auf eine Send-Receive-Message abgebildet und diese von der Anwendung nach jedem Lesen auf `false` zurückgesetzt, so kann die Anwendung sehr leicht feststellen, ob zwischen den Rechenzyklen ein Botschaftsempfang stattgefunden hat.

## 2.6.2 Groups (CAN-IO Device)

In dieser CAN-IO Registerkarte werden die CAN-Botschaften in Form von Signalgruppen spezifiziert.

#### **Hinweis**

*Im Gegensatz zum ES1222 CAN-IO Devices ist hier die Anzahl der Signalgruppen auf 15 begrenzt. Die ersten 14 Signalgruppen können wahlweise für „send“ und „receive“ Übertragungsrichtung verwendet werden; die 15. Signalgruppe kann ausschließlich für eine Empfangsbotschaft („receive“) verwendet werden. Darüberhinaus unterstützt dieses CAN-IO Device keine im Interrupt-Betrieb empfangenen Botschaften.*



**Abb. 2-13** Die Registerkarte „Groups“ des CAN-IO Devices

### Hinweis

Bei Änderung des Signalgruppennamens, des Signalnamens oder der Signalrichtung ('Direction') wird die die eventuell zugeordnete ASD Message u.U. nicht mehr automatisch zugeordnet und muss manuell neu zugeordnet werden.

### Direction

Hier kann die Richtung der CAN-Botschaft bestimmt werden („send“ = Sendebotschaft, „receive“ = Empfangsbotschaft).

### Task

Hier wird die Task angegeben, in welcher die Botschaft gesendet, bzw. empfangen werden soll. Soll eine Empfangsbotschaft im Interrupt-Betrieb empfangen werden, so wird diese Einstellung zurückgesetzt und gesperrt.

### Identifier Type

Hier kann eingestellt werden, ob der Identifier-Type im Standard oder Extended Format interpretiert werden soll.

Bei CAN-Botschaften kann zwischen *standard frames* mit 11-Bit Identifiern oder *extended frames* mit 29-Bit Identifiern gewählt werden. In dieser Zeile kann die Länge des Identifier-Felds (*standard/extended*) festgelegt werden.

Wenn der Identifier *standard* gewählt wird, ist im Register „Groups“ des CAN-IO Device (siehe „Identifier dec/hex“ weiter unten) nur die Eingabe von 11 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die *most significant* Bits (MSB) abgeschnitten Eine Warnung wird **nicht** angezeigt.

Wenn der Identifier *extended* gewählt wird, ist nur die Eingabe von 29 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die *most significant* Bits (MSB) abgeschnitten Eine Warnung wird **nicht** angezeigt

*Identifier dec/hex*

Hier muss der Botschafts-Identifier eingegeben werden. Je nach gewählter Identifiertyp-Einstellung in dem darüberliegenden CAN-Controller Item (standard oder extended) kann hier der Wert unterschiedlich groß werden:

**Standard identifier:** 11 Bit 2 047 dec 7 FF hex  
**Extended identifier:** 29 Bit 536 870 911 dec F FF FF FF hex

Jede Signalgruppe bzw. CAN-Botschaft muss einen anderen Identifier besitzen.

*Length [Byte]*

Legt fest, wieviel Nutzdatenbytes die betreffende Botschaft übertragen kann (1..8).

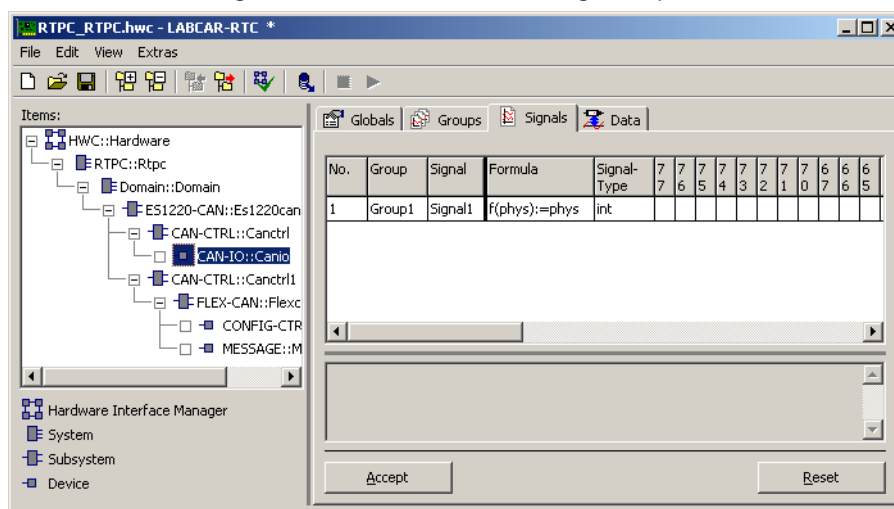
*Activated Task\**

Für „receive“-CAN-Botschaften<sup>1</sup> (Signalgruppen) kann hier eine Software-Task angegeben werden, die immer dann aktiviert wird, wenn die relevante Signalgruppe empfangen und die Empfangstask ausgeführt wurde. Die hier eingegebene Task kann z.B. zur Nachbearbeitung verwendet werden.

Diese Spalte ist standardmäßig nicht eingeblendet.

2.6.3 Signals (CAN-IO Device)

In dieser CAN-IO Registerkarte werden die CAN-Signale spezifiziert.



**Abb. 2-14** Die Registerkarte „Signals“ des CAN-IO Devices

**Hinweis**

Neue Signale lassen sich über das Kontextmenü im Register „Signals“ anlegen (siehe hierzu „Das Kontextmenü der Register“ auf Seite 50).

<sup>1</sup> Es ist möglich, eine solche Task auch für „send“-Botschaften zu spezifizieren, auch wenn es keine empfohlene Anwendung für diese Verwendung gibt. (Da der Sendeprozess asynchron abläuft, ist nicht sichergestellt, ob die CAN-Botschaft gesendet ist, wenn die Task aktiviert wird.)

### Group

---

Hier erfolgt die Zuordnung eines Signals zu der gewünschten Signalgruppe.

### Signal Type

---

Hier wird der Signaltyp, in welchem das Signal über den CAN-Bus übertragen wird, festgelegt.

Es stehen folgende Einstellungen zur Verfügung:

Signaltyp	Datentyp
int	Kennzeichnet ein vorzeichenbehaftetes Signal im Standard 2er-Komplement Datenformat (max. 32 Bit)
(s)int	Kennzeichnet ein vorzeichenbehaftetes Signal, in dem als höchstwertiges Bit das Vorzeichen und anschließend der Absolutwert des Signals übertragen wird. Ist das Vorzeichen-Bit gesetzt, so handelt es sich dabei um eine negative Zahl (max. 32 Bit)
uint	Kennzeichnet ein vorzeichenloses Signal (max. 32 Bit)
bool	Kennzeichnet ein Boolesches Signal. Hierbei darf in der Bit-Matrix nur ein einzelnes Bit markiert sein.
real	Kennzeichnet ein Fließkommawert im „Standard IEEE Float (4 Byte)“ Format. Hierbei dürfen entsprechend nur 32 Bit in der Bit-Matrix markiert sein.

Eine Übersicht über die IEEE-Fließkomma-Formate finden Sie in folgender Tabelle:

Format	Sign	Exponent	Fraction
Float	1 Bit	8 Bit	24 Bit
Double (nicht unterstützt)	1 Bit	11 Bit	52 Bit

### 7654321.. (Bitmatrix)

---

Eine CAN-Botschaft kann bis zu 8 Datenbytes übertragen. In einer Bit-Matrix kann für jedes Signal festgelegt werden, welche Bits dieses Signal benötigt, bzw. belegt (ein Signal = eine Reihe).

Der Aufbau der Spalten ist wie folgt:

7	7	...	0	0	Byte-Nummer
7	6	...	1	0	Bit-Nummer

### Bedeutung der Bit-Felder:

Leeres Feld	Das entsprechende Signal verwendet das Bit nicht
Belegtes Feld	Das Signal benötigt dieses Bit an dieser Position
„X“-Feld	Das entsprechende Bit steht zur Datenübertragung nicht zur Verfügung, weil das Signal weniger Nutzdatenbytes hat (siehe Einstellung „Length“ im Register „Groups“).

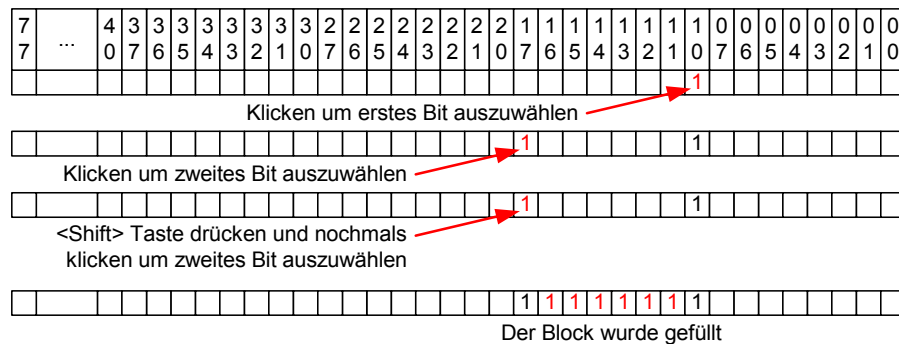
### Bedienung der Bit-Felder:

Über die Pfeiltasten der Tastatur kann die gewünschte Bit-Zelle ausgewählt werden.

Mit der Leertaste kann die Zelle zwischen „unbelegt“ (Zelle leer) und „1“ (Bit belegt) „getoggelt“ werden.

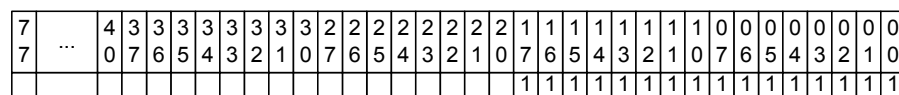
Ein Anklicken mit der Maus „toggelt“ die entsprechende Zelle ebenso zwischen „unbelegt“ und „1“. Wird während des Klickens zusätzlich noch die <ALT>-Taste gedrückt, so wird der Wert „1“ bis „9“ inkrementiert (wichtig für „Blockbildung“).

Mehrere Bits gleichzeitig kann man folgendermaßen selektieren:

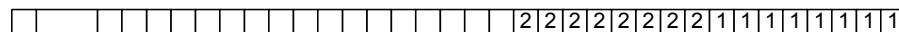


Mit Hilfe der verschiedenen Zahlen („1111 2222...“) lassen sich Datenblöcke bilden, mit denen praktisch jedes übertragene Signal beschrieben werden kann. Die Zahlen, aus denen die Datenblöcke gebildet werden, haben dabei die Bedeutung, dass der Block mit der höchsten Zahl („2222“) den Block angibt, der bei der Übertragung die höchstwertigsten Bits enthält. Der Block mit der kleinsten Zahl („1111“) enthält bei der Übertragung die niederwertigsten Bits. Mit den zur Verfügung stehenden Zahlen (1...9) lässt sich folglich ein Signal mit bis zu 9 Bit-Blöcken beschreiben. Da die Darstellung der Bits genau der Form entspricht, in der ein Intel Signal übertragen wird, ist die Blockbildung für die Darstellung von Signalen im Motorola Format notwendig, sobald das Signal eine Länge von 8 Bit überschreitet.

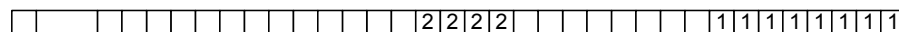
Beispiele für die Definition von verschiedenen Signalen:



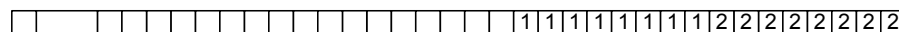
16 Bit Signal im Intel Format



Andere Beschreibung des selben Signals



12 Bit Signal (Intel Format) mit Lücke



16 Bit Signal im Motorola Format

### 3 ES1302.1 A/D Board

Das ES1302.1 A/D Board dient der analogen Datenerfassung in VMEbus-Systemen. Die Karte ist für mittlere Auflösungen und Datenraten konzipiert. Sie umfasst unter anderen folgende Funktionen:

- Analog-Digital-Umsetzer mit 12 Bit Auflösung und einer maximalen Gesamtabtastrate von 300 kHz
- 16 einpolige oder 8 differentielle analoge Eingangskanäle
- Vier Eingangsspannungsbereiche und vier Verstärkungsfaktoren per Software einstellbar
- FIFO-Speicher für Messwerte und Kanalnummer

Die ES1302.1 kann in VMEbus-Systemen überall dort eingesetzt werden, wo analoge Eingangssignale mit mittlerer Auflösung erfasst werden sollen.

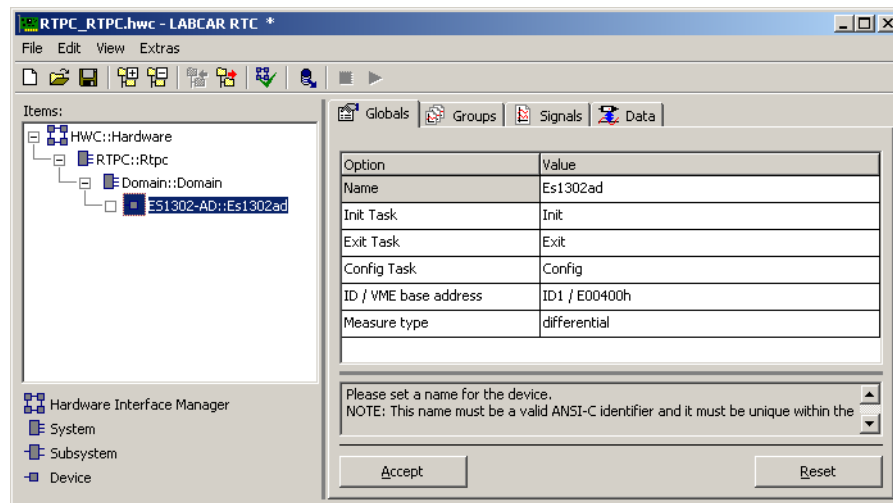
Beispielhafte Einsatzgebiete sind:

- Erfassung analoger Gebersignale, beispielsweise Motortemperatur, Öltemperatur, Fahrpedalstellung
- Ventilstrommessung mit vorgeschaltetem I/U-Wandler

#### 3.1 ES1302-AD Device

##### 3.1.1 Globals (ES1302-AD Device)

In diesem Abschnitt werden die globalen Einstellungen des ES1302-AD Device beschrieben.



**Abb. 3-1** Die Registerkarte „Globals“ des ES1302-AD Device

##### *Init Task*

In dieser Zeile wird die Task zur Initialisierung der ES1302.1 zugeordnet (Type: Init / Application Mode: active).

##### *Exit Task*

In dieser Zeile wird die Task zugeordnet, die bei einem Stopp des Experiments mit der ES1302.1 ausgeführt werden soll (Type: Init / Application Mode: inactive).

### Config Task

Diese Task kann dazu verwendet werden, um die ES1302.1 während des Experiments zu rekonfigurieren. Bitte erzeugen Sie im Betriebssystemeditor eine Software-Task (die nicht aufgerufen werden darf), wenn Sie die ES1302.1 nicht rekonfigurieren wollen.

### ID / VME base address

Im Auswahlfeld „ID / VME base address“ die VMEbus-Basisadresse der ES1302.1 angeben. Es stehen 5 unterschiedliche Basisadressen zur Auswahl (ID1/E00400h, ID2/E00500h, ID3/E00600h, ID4/FE0100h und ID5/FE0200h).

### Measure type

Im Auswahlfeld „Measure type“ wird die Art der Messung festgelegt. Es stehen 3 Optionen zur Auswahl:

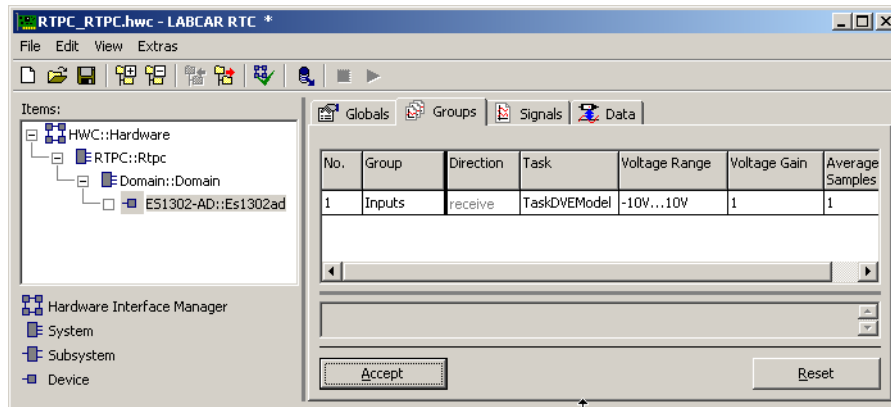
- single-ended: Es werden 16 einpolige analoge Eingangskanäle erfasst.
- differential: Es werden 8 differentielle analoge Eingangskanäle erfasst.
- reference: Es werden die 4 internen Referenzspannungskanäle vermessen.

### Hinweis

Für die korrekte Messung der Eingangsspannungen ist es erforderlich, dass alle unbenutzten Eingänge auf einem definierten Potential liegen.

### 3.1.2 Groups (ES1302-AD Device)

In diesem Abschnitt werden die die Signalgruppen betreffenden Einstellungen des ES1302-AD Device beschrieben.



**Abb. 3-2** Die Registerkarte „Groups“ des ES1302-AD Device

### Voltage Range

Im Feld „Voltage Range“ wird der Eingangsspannungsbereich der Messkanäle festgelegt. Es stehen 4 Bereiche (0 V... 5.0 V, -5.0 V... 5.0 V, 0 V... 10.0 V und -10.0 V... 10.0 V) zur Auswahl.



### Voltage Gain

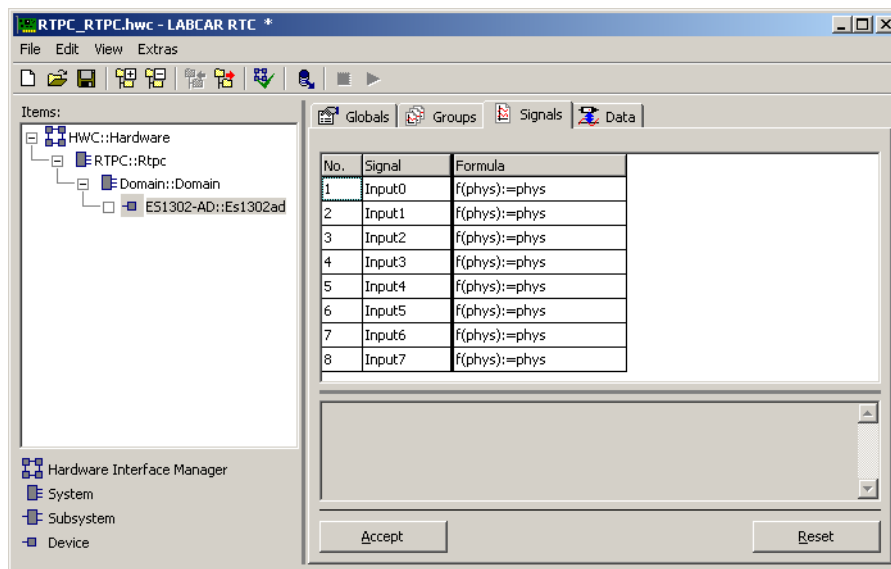
Im Feld „Voltage Gain“ wird eine Eingangsspannungsverstärkung für die Messkanäle festgelegt. Es stehen 4 Verstärkungsfaktoren (1, 2, 4, 8) zur Auswahl. Zu beachten ist, dass sich dadurch der Eingangsspannungsbereich nicht verändert, d.h. die verstärkte Spannung muss innerhalb des eingestellten Spannungsbereiches liegen!

### Average Samples

Im Feld „Average Samples“ wird festgelegt, aus wieviel Abtastwerten ein analoger Spannungswert durch Mittelwertbildung berechnet wird. Durch Mittelwertbildung kann eine Stör- oder Rauschunterdrückung erzielt werden - es ist jedoch zu beachten, dass pro Messkanal und pro Abtastung etwa 3  $\mu$ s benötigt werden.

### 3.1.3 Signals (ES1302-AD Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des ES1302-AD Device beschrieben.



**Abb. 3-3** Die Registerkarte „Signals“ des ES1302-AD Device

Die Anzahl der Signale ist abhängig von der im Feld „Measure type“ der Registerkarte „Globals“ eingestellten Messart:

- single-ended: 16 Eingänge
- differential: 8 Eingänge
- reference: 4 Eingänge

Die Signale *Inputx* sind vom Datentyp „float“ bzw. „real32“. Die Signale geben die gemessene Spannung am zugehörigen Messkanal in V an.

#### **Hinweis**

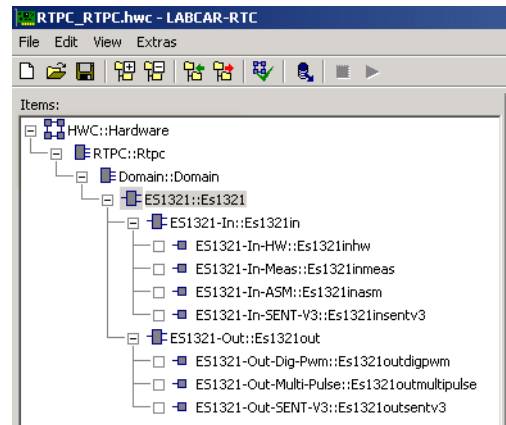
Wenn in diesem Dokument Datentypen angegeben werden, so sind das die Datentypen, die der RTIO-Treiber intern verwendet.



## 4 ES1321.1 PWM I/O Board

Das ES1321.1 PWM I/O Board dient der Ausgabe und Erfassung von pulswertenmodulierten Signalen. Der typische Anwendungsfall für die ES1321.1 in LAB-CAR-Projekten ist die Nachbildung von Fahrzeugsensoren und die Ansteuerung von Aktoren.

Im RTIO-Editor wird das ES1321.1 PWM I/O Board durch Auswahl des ES1321 Subsystems eingebunden.



**Abb. 4-1** RTIO-Hardwarebeschreibung mit ES1321.1 PWM I/O Board

Das ES1321.1 PWM I/O Board besitzt 24 Eingangskanäle zur Erfassung von pulswertenmodulierten Signalen – die Konfiguration dieser Eingangskanäle erfolgt mit dem ES1321-In-HW Subsystem. Neben diesem Subsystem können dem ES1321-RTIO-Element bis zu vier ES1321-In Subsysteme zugeordnet werden.

Das ES1321-Out Subsystem kann aus jeweils einem ES1321-Dig-Pwm, bis zu zwei ES1321-Multi-Pulse Devices und bis zu vier ES1321-Out-SENT Devices bestehen.

### *Messfunktionen des ES1321.1 PWM I/O Boards*

Die Eingänge der ES1321.1 verfügen über folgende Messfunktionen:

- Frequenz
- Cycle Time
- Hightime / Lowtime
- Additive Hightime / Additive Lowtime
- Number of Low/High Pulses
- Number of Low/High Edges
- Duty Cycle für high und low actives Signal
- Positionszähler für Schrittmotoren
- SENT-Receive

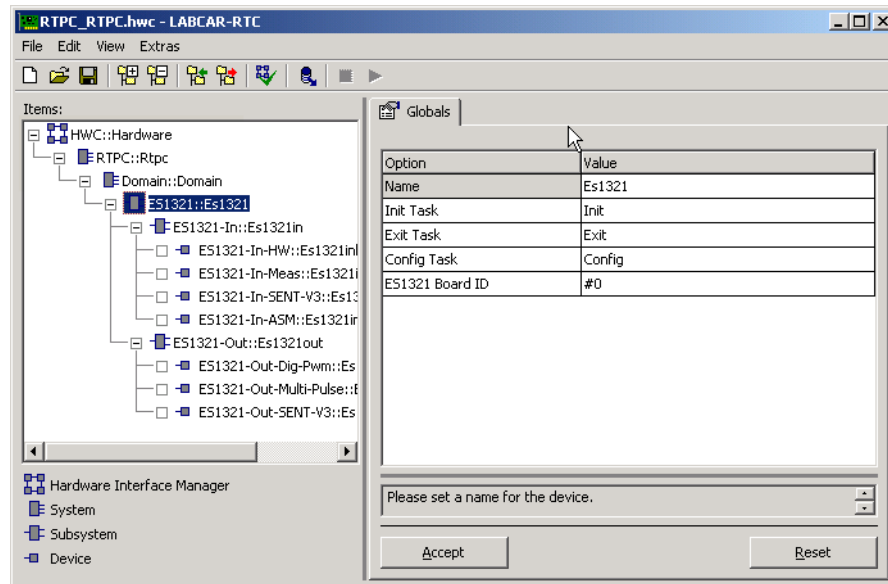
Die Ausgänge der ES1321.1 verfügen über folgende Funktionen:

- Digital Output
- PWM Signalerzeugung
- Multi-Pulse
- SENT-Transmit

## 4.1 ES1321 Subsystem

### 4.1.1 Globals (ES1321 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES1321 Subsystems beschrieben.



**Abb. 4-2** Die Registerkarte „Globals“ des ES1321 Subsystems

#### *ES1321 Board ID*

Dieses Optionsfeld dient zur Identifizierung des ES1321.1 PWM I/O Boards. Es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1321.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist.

Die Nummerierung der ES1321.1 im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1) in Richtung ansteigender Slot-Nummerierung, beginnend mit 1. Die ermittelte Nummer der ES1321.1 ist in dem Listenfeld „ES1321 Board ID“ einzustellen.

Im RTIO-Editor können bis zu zwanzig ES1321.1 PWM I/O Boards pro Baugruppenträger eingebunden werden.

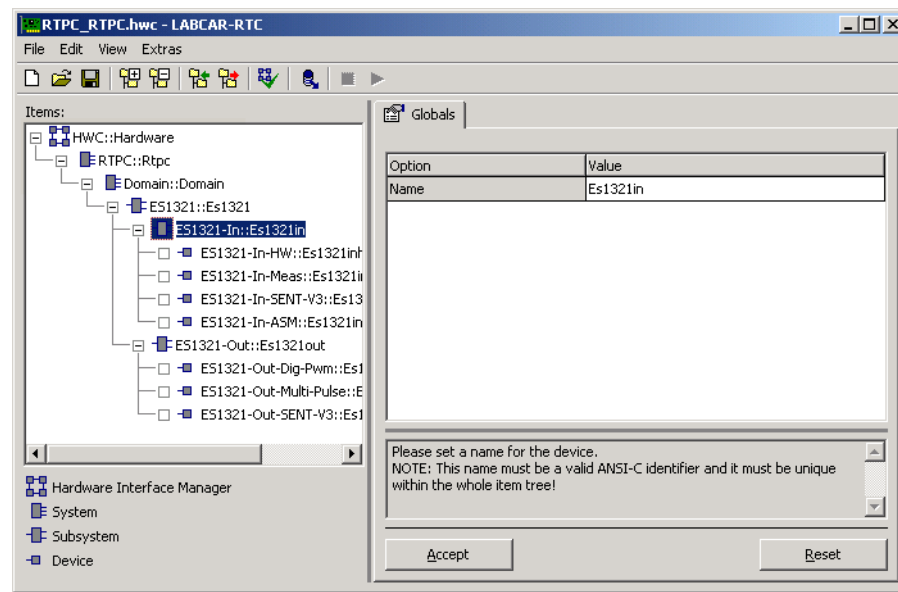
Dieser RTIO-Parameter ist nicht während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

## 4.2 ES1321-In Subsystem

### 4.2.1 Globals (ES1321-In Subsystem)

Das ES1321-In Subsystem dient zur Konfiguration der Messeingänge des ES1321.1 PWM I/O Boards.

Abb. 4-3 zeigt die Registerkarte „Globals“ des ES1321-In Subsystems.



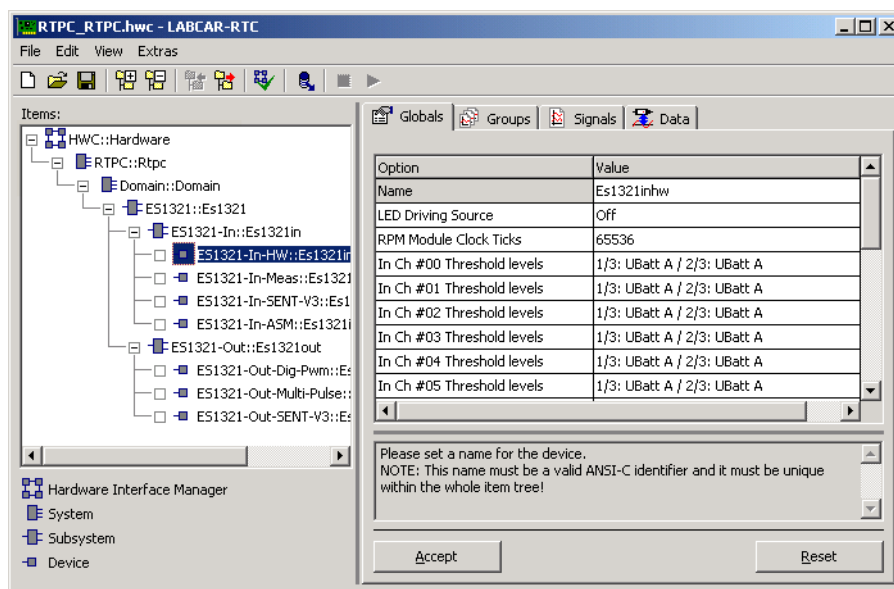
**Abb. 4-3** Die Registerkarte „Globals“ des ES1321-In Subsystems

## 4.3 ES1321-In-HW Device

### 4.3.1 Globals (ES1321-In-HW Device)

Das ES1321-In-HW Device dient zur Konfiguration der Kanal-LED und der Einstellung der Schaltschwellen der Eingangskomparatoren.

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 4-4** Die Registerkarte „Globals“ des ES1321-In-HW Device

#### *LED Driving Source*

In Abhängigkeit von diesem Signal wird die Channel-LED der ES1321.1 angesteuert. Bei Auswahl von „Input Channel #xx“ wird die LED vom Signalpegel des Kanals „xx“ angesteuert.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

#### *In Ch #xx Threshold Levels ...*

Für die Schaltschwellen des Eingangskomparators stehen folgende Möglichkeiten zur Auswahl:

- 1/3: UBatt A / 2/3: UBatt A

Bei dieser Option werden die Schaltschwellen für die Auswertung des Eingangspiegels auf 1/3 bzw. 2/3 der Referenz UBatt\_A ausgewählt.

- 1/3: UBatt B / 2/3: UBatt B

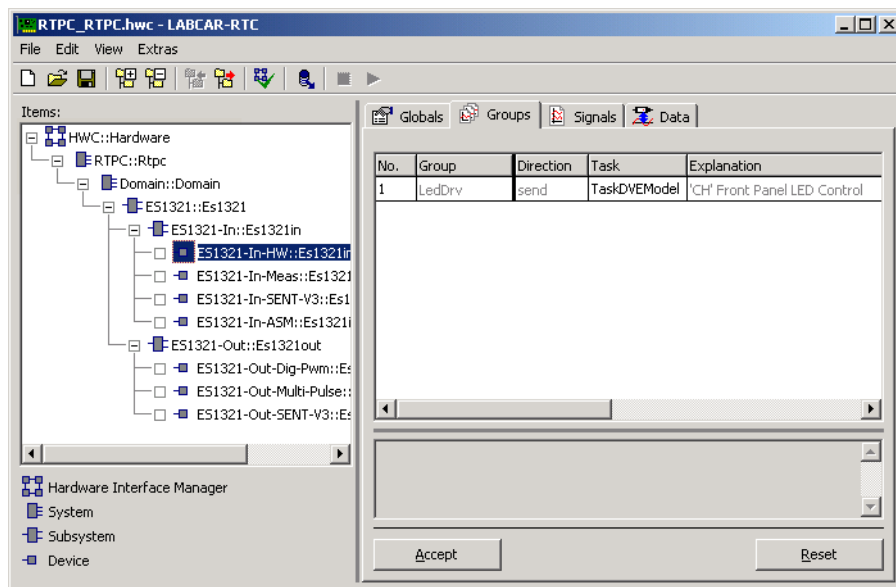
Bei dieser Option werden die Schaltschwellen für die Auswertung des Eingangspiegels auf 1/3 bzw. 2/3 der Referenz UBatt\_B ausgewählt.

- userdefined

Hier gibt der Anwender 1/3 der Referenzspannung ein – dieser Wert stellt die untere Schaltschwelle dar. Die obere Schaltschwelle beträgt 2/3 der Referenzspannung.

#### 4.3.2 Groups (ES1321-In-HW Device)

Das ES1321-In-HW device besitzt eine Signalgruppe „LedDrv“ zum Setzen der LED.



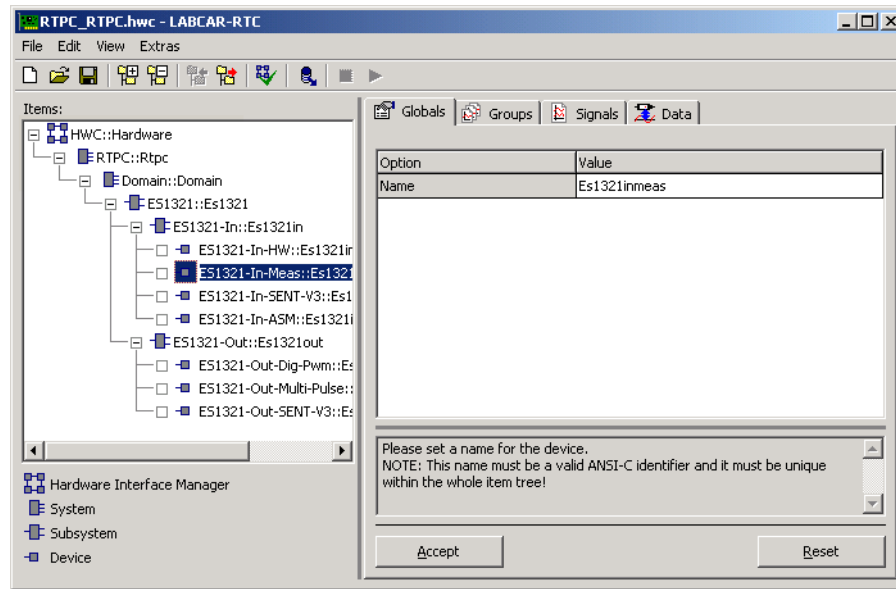
**Abb. 4-5** Die Registerkarte „Groups“ des ES1321-In-HW Device

## 4.4 ES1321-In-Meas Device

### 4.4.1 Globals (ES1321-In-Meas Device)

Das ES1321-In-Meas Device dient zur Konfiguration der Messfunktionen auf den 24 Eingangskanälen.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.

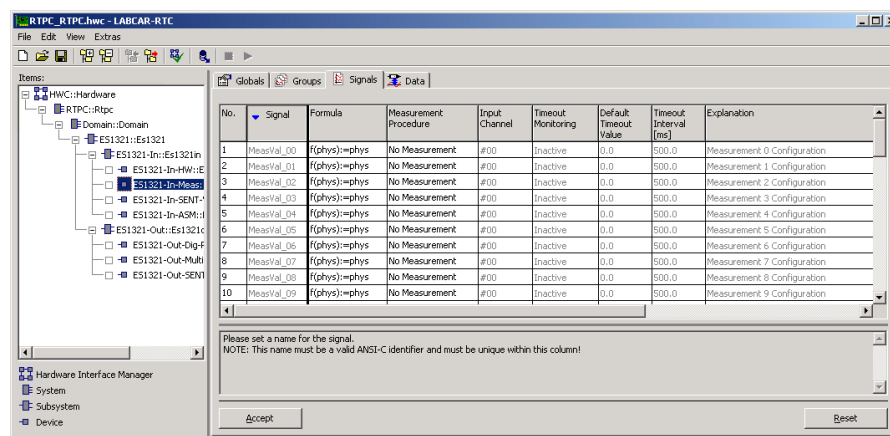


**Abb. 4-6** Die Registerkarte „Globals“ des ES1321-In-Meas Device

### 4.4.2 Signals (ES1321-In-Meas Device)

In der Registerkarte „Signals“ wird die Definition der 24 Messungen eines ES1321-In-Meas Devices durchgeführt. Alle Optionsfelder online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Die folgende Abbildung zeigt die Signale der Registerkarte „Signals“.



**Abb. 4-7** Die Registerkarte „Signals“ des ES1321-In-Meas Device



### *Measurement Procedure*

---

In diesem Listenfeld wird das Messverfahren eingestellt. Im Abschnitt „Die Messverfahren“ auf Seite 87 finden Sie eine detaillierte Beschreibung und Konfigurationsrichtlinien zu den einzelnen Verfahren.

Die Messungen finden jeweils an der steigenden Flanke des Eingangssignals statt.

### *Input Channel*

---

In diesem Listenfeld wird der Eingangskanal der ES1321.1 eingestellt, auf dem die Messung durchgeführt werden soll.

### *Timeout Monitoring*

---

Definition der Timeout-Überwachung für die zugehörige Messung. Die folgenden Einstellungen sind möglich:

- „Inactive“: Keine Timeout-Überwachung.
- „Intvl Predef“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl InpDep“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.

### *Default Timeout Value*

---

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls der „Intvl Predef“ Timeout-Überwachungsmodus eingestellt ist.

Die Timeout-Bedingung ist erfüllt, wenn einer der vier ausgewerteten Zeitstempel außerhalb der eingestellten Timeout-Wertes liegt.

### *Messmethoden und Timeout-Behandlung*

---

Wenn bei „Timeout Monitoring“ die Option „Intvl InpDep“ gewählt wurde, so werden – je nach eingestelltem Messverfahren – die folgenden Werte ausgegeben.

<b>Messmethode</b>	<b>Wert (Input Level Dependent)</b>
No Measurement	-
High Time [µs]	0
Low Time [µs]	0
Additive High Time [µs]	0
Additive Low Time [µs]	0
Number of Low Pulses	0
Number of High Pulses	0
Number of Rising Edges	0
Number of Falling Edges	0
Cycle Time --/-- [µs]	Die Zeit, die seit dem letztem gültigen Zeitstempel vergangen ist

Messmethode	Wert (Input Level Dependent)
Frequency --/-- [Hz]	Kehrwert der Zeit, die seit dem letztem gültigen Zeitstempel vergangen ist
Duty Cycle L/(L+H)	1: Eingangsspiegel „Low“ 0: Eingangsspiegel „High“
Duty Cycle H/(L+H)	1: Eingangsspiegel „High“ 0: Eingangsspiegel „Low“
Level (Active High)	0
Level (Active Low)	0
Stepper A	0
Stepper B	0

## 4.5 Die Messverfahren

### Begriffsdefinitionen

#### Aktive Signalflanke

Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Flanke:

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### Öffnende Flanke

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



### Das Grundprinzip der Messwertberechnung

Die Messung und Auswertung der Messdaten findet beim ES1321.1 PWM I/O Board im wesentlichen auf der Baugruppe statt. Diese werden vom internen Prozessor der Karte ausgewertet und über ein Dual-Ported RAM über den VMEbus zur ES1130 und Real-Time PC weitergegeben. Für jede Messmethode wird jeweils ein 32-Bit Wert übertragen, der gegebenenfalls noch vom RTIO Treiber weiterverarbeitet wird.

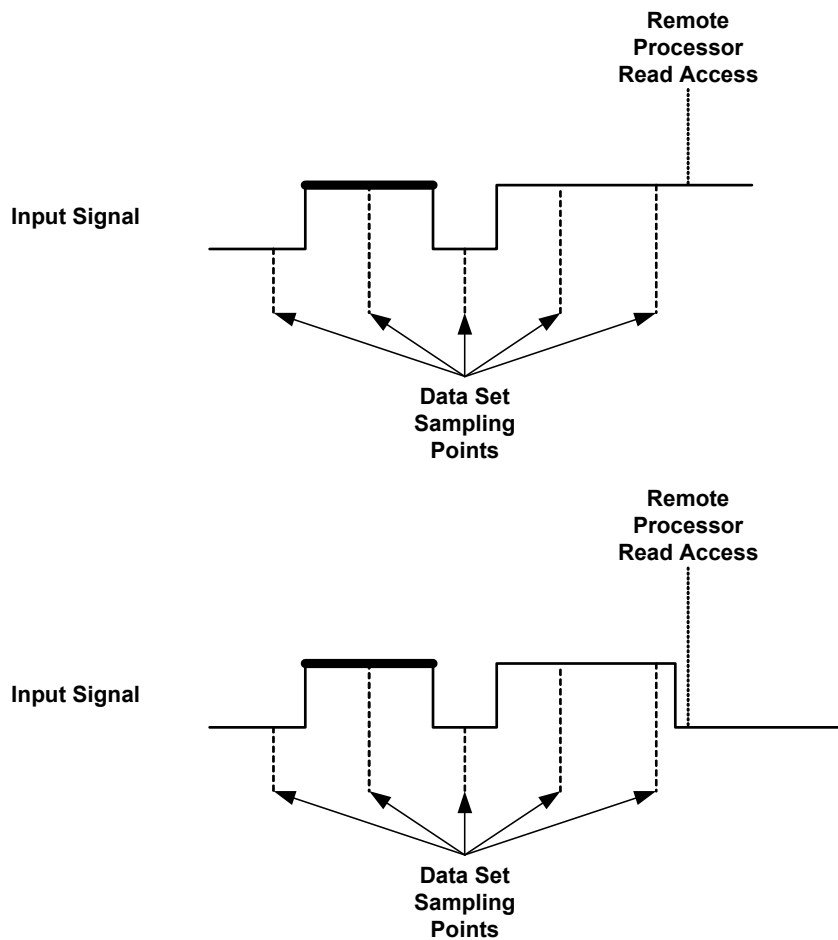
Die Messungen besitzen eine Auflösung von 20 ns.

#### 4.5.1 Pulsweitenmessungen

Mit den Messverfahren

- High Time [ $\mu\text{s}$ ]
- Low Time [ $\mu\text{s}$ ]

werden die High- und Low-Zeiten der Pulse eines PWM-Eingangs vermessen. Es wird die Pulsweite des dem letzten Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehenden Pulses gemessen. Abb. 4-8 zeigt dies beispielhaft für eine High-Zeit-Messung, der gemessene Puls ist mit fetter Strichstärke hervorgehoben.



**Abb. 4-8** High-Zeit-Messung (der gemessene Puls ist mit fetter Strichstärke hervorgehoben)

Eine Timeout-Bedingung liegt dann vor, wenn mindestens eine der vier Flanken älter ist als der aktuelle Zeitstempel abzüglich der Timeout-Zeit.

Die Timeout-Behandlung ist bei allen Messverfahren gleich.

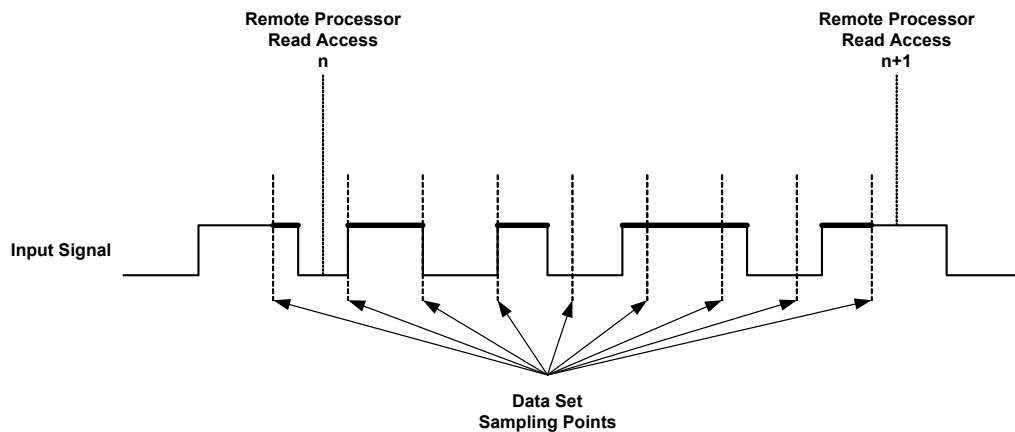
#### 4.5.2 Additive Pulsweitenmessungen

Wird der Leseprozess für die „MeasVal“-Signalgruppe wiederholt (im Spezialfall periodisch) aktiviert, so liefern die Messverfahren

- Additive High Time [ $\mu\text{s}$ ]
- Additive Low Time [ $\mu\text{s}$ ]

die Zeit, in der das zu vermessende Signal zwischen zwei aufeinanderfolgenden Aktivierungen des Leseprozesses einen aktiven Pegel angenommen hat.

Abb. 4-9 veranschaulicht die Messwertberechnung am Beispiel einer additiven High-Zeit-Messung. Die Summe der fett markierten Liniensegmente ist die additive High-Zeit, die bei der (n+1)-ten Aktivierung des Leseprozesses geliefert wird.



**Abb. 4-9** Additive High-Zeit-Messung: Illustration der Messwertberechnung  
Eine Timeout-Bedingung liegt dann vor, wenn mindestens eine der vier Flanken älter ist als der aktuelle Zeitstempel abzüglich der Timeout-Zeit.

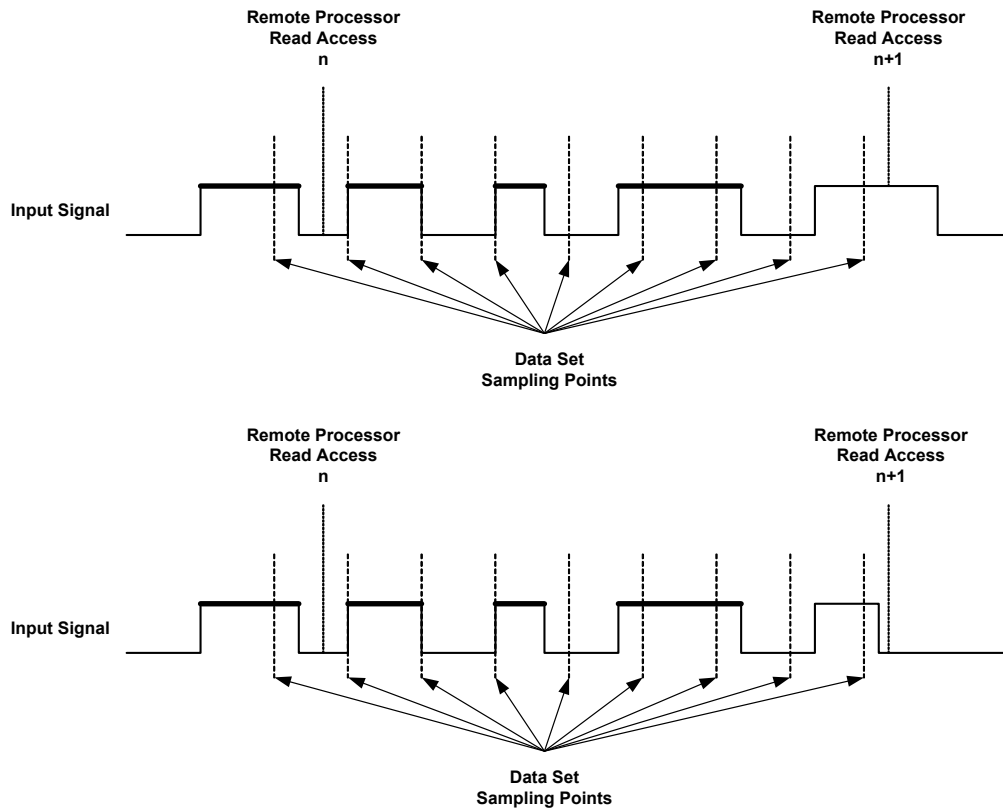
#### 4.5.3 Puls- und Flankenanzählung

Wird der Leseprozess für die „MeasVal“-Signalgruppe wiederholt (im Spezialfall periodisch) aktiviert, so liefern die Messverfahren

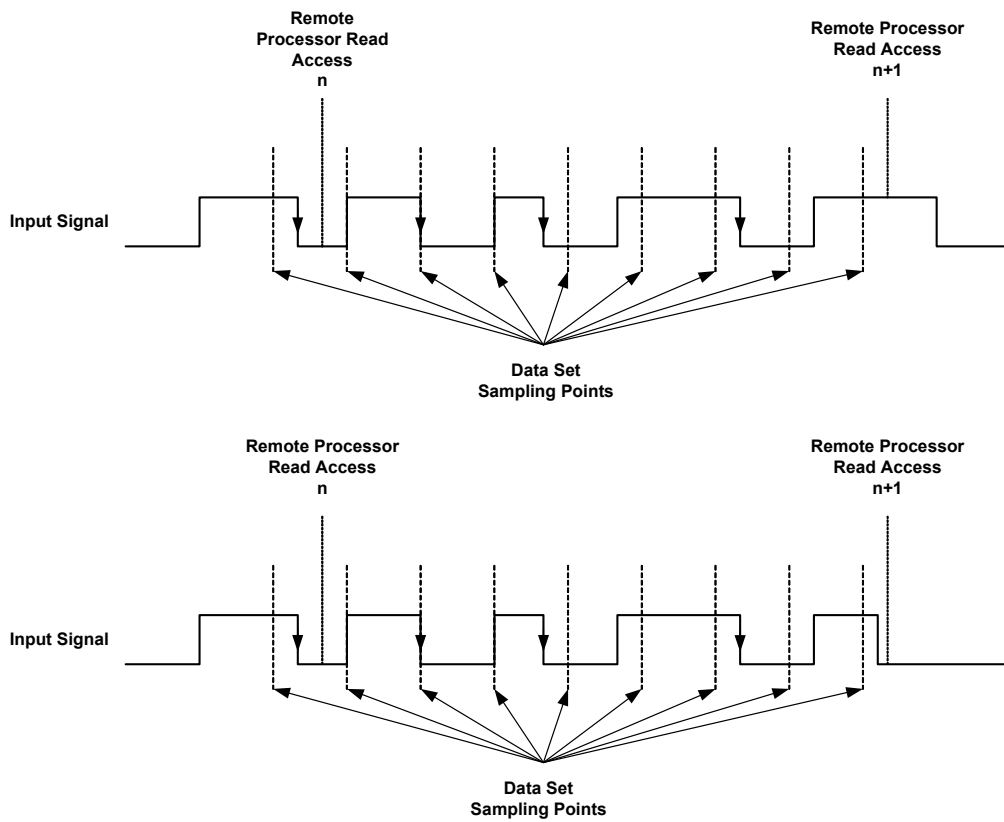
- Number of Low Pulses
- Number of High Pulses
- Number of Rising Edges
- Number of Falling Edges

die Anzahl der aktiven Pulse bzw. Signalfanken zwischen zwei aufeinanderfolgenden Aktivierungen des Leseprozesses. Abb. 4-10 veranschaulicht die Methodik der Puls-Zählung am Beispiel einer High-Puls-Zählung im Detail.

Gezählt werden alle Pulse, deren schließende Flanke in die Zeitspanne zwischen den Aktivierungen des Leseprozesses fällt. Wenn man es exakt formulieren will, ist der Zeitraum bestimmt durch die den Aktivierungen unmittelbar vorausgehenden Datenerfassungszeitpunkte. Was die Flanken-zählung anbelangt, so ist dort der Auswertungszeitraum exakt der gleiche (siehe Abb. 4-11 auf Seite 91).



**Abb. 4-10** High-Puls-Zählung: Die Anzahl der mit fetter Strichstärke markierten Pulse wird beim (n+1)-ten Lesevorgang als Messwert zurückgegeben



**Abb. 4-11** Zählung fallender Flanken: Die Zahl der markierten fallenden Flanken wird beim (n+1)-ten Lesezugriff als Messwert zurückgegeben

4.5.4 Frequenz- und Zykluszeitmessungen

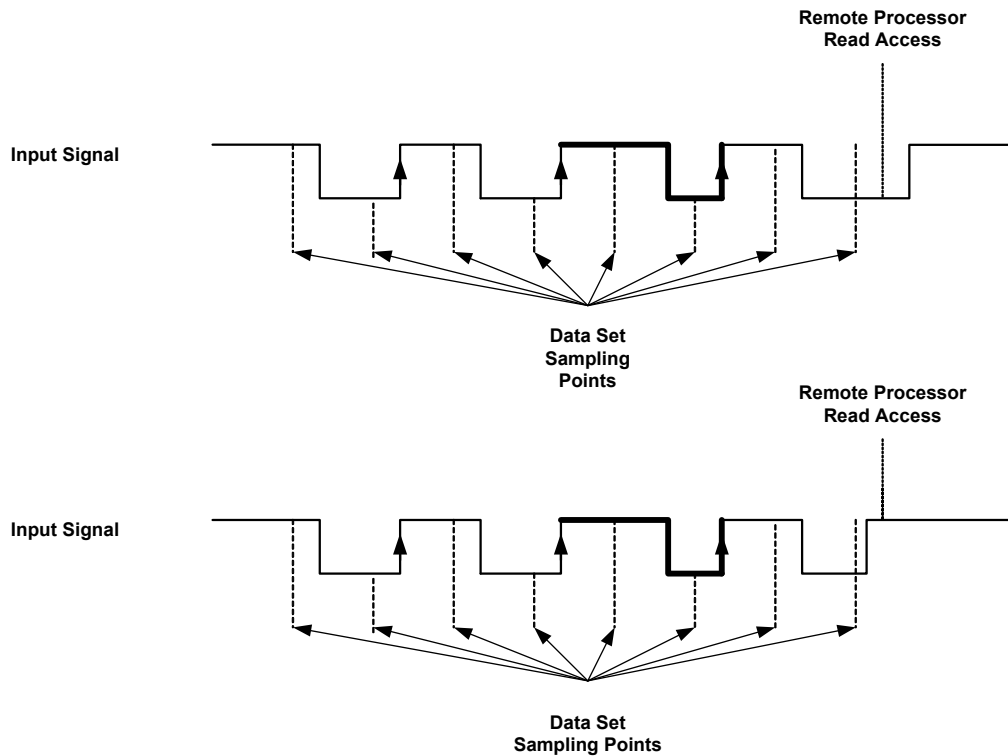
Die Messverfahren

- Cycle Time [ $\mu\text{s}$ ]
- Frequency [Hz]

liefern die Zykluszeit bzw. die Frequenz des Signals an einem PWM-Eingang gemessen an steigenden oder fallenden Flanken. Es wird jeweils die dem letzten<sup>1</sup> Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehende aktive Signalflanke mit zugehöriger Periode ausgewertet.

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt

Abb. 4-12 zeigt die ausgewertete Periode beispielhaft für eine Frequenz- oder Zykluszeitmessung an ansteigenden Flanken.



**Abb. 4-12** Ausgewertete Periode (fett gezeichnet) bei einer Frequenz- oder Zykluszeitmessung an ansteigenden Flanken

#### 4.5.5 Tastverhältnismessungen

Beim ES1321.1 PWM I/O Board stehen die nachfolgenden Tastverhältnismessungen zur Verfügung:

- Duty Cycle L/(L+H)
- Duty Cycle H/(L+H)

Bei diesen Messungen wird eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt. Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest.

Was die Auswahl der Periode des Eingangssignals anbelangt, auf Basis derer die Tastverhältnisberechnung durchgeführt wird, so gilt exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 4.5.4 auf Seite 91 beschrieben wurde.

#### 4.5.6 Pegelmessungen

Die Messverfahren

- Level (Active High)
- Level (Active Low)

liefern den Pegel eines PWM-Eingangs in Form einer aktiv-/inaktiv-Information. „0“ bedeutet das Signal ist inaktiv, „1“ bedeutet das Signal ist aktiv.



4.5.7 Positionsverfolgung an zwei-phasigen Schrittmotoren

*Funktionsbeschreibung*

Die ES1321.1-Firmware ermöglicht die Positionsverfolgung und damit prinzipiell die Erfassung von Rotorstellung und -drehzahl von zwei-phasigen Schrittmotoren. Dazu sind zwei Hardwarekanäle und ein Softwarekanal bzw. Messkanal erforderlich.

Das Messverfahren „Stepper A“ oder „Stepper B“ kann unter „Measurement Procedure“ ausgewählt werden. Die Hardwarekanäle sind wie folgt zugeordnet:

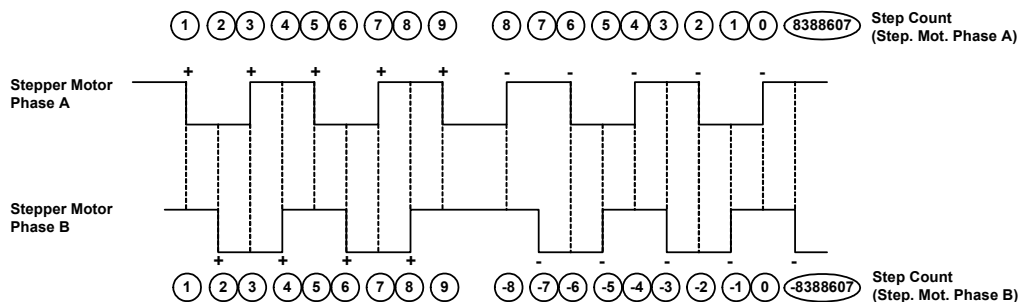
- Software-Kanal 0 → Phase A → Hardware-Kanal 0  
Phase B → Hardware-Kanal 1
- Software-Kanal 1 → Phase A → Hardware-Kanal 2  
Phase B → Hardware-Kanal 3
- Software-Kanal 2 → Phase A → Hardware-Kanal 4  
Phase B → Hardware-Kanal 5

...

**Hinweis**

*Es müssen immer zwei benachbarte Kanäle zu einem Paar zusammengefasst werden (Kanal 0 und Kanal 1), (Kanal 2 und Kanal 3) usw.). Kanal 0 entspricht beim ersten Paar „Phase A“ und Kanal 1 „Phase B“. Bei den weiteren Kanälen ist die Zuordnung analog.*

Im Feld „Reference Channel“ eines Messkanals ist auf die Nummer des anderen an der Messung beteiligten Messkanals zu verweisen.



**Abb. 4-13** Darstellung der Arbeitsweise der Positionsverfolgung an zwei-phasigen Schrittmotoren

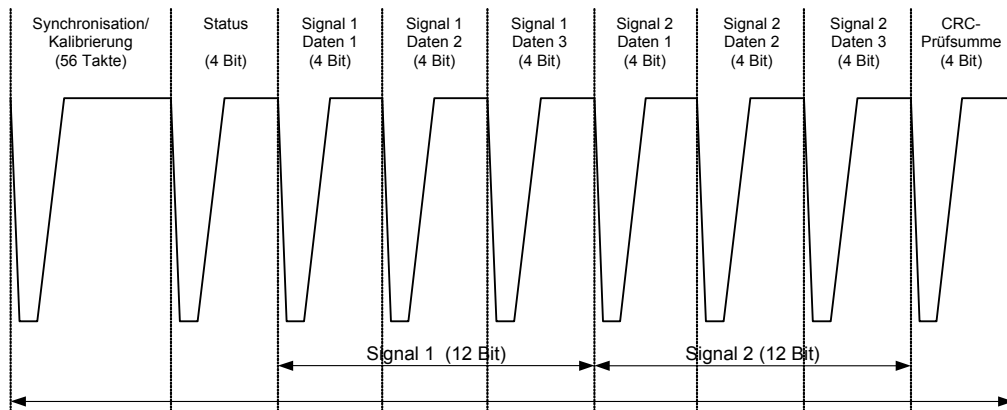
**Hinweis**

*Die Positionsverfolgung erfolgt modulo  $2^{29}$ . Inkrementieren bei einem Positionsstand von  $(2^{29}-1)$  erzeugt einen Wert des Positionszählers von 0. Dekrementieren bei einem Positionsstand von 0 erzeugt einen Wert des Positionszählers der betragsmäßig gleich  $(2^{29}-1)$  ist.*

## 4.6 ES1321-In-SENT Device

### 4.6.1 Globals (ES1321-In-SENT Device)

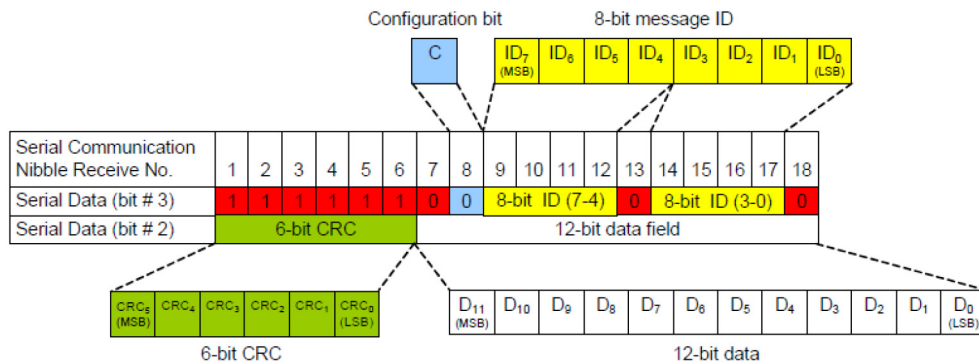
Dieses RTIO-Element ermöglicht den Empfang eines Signals gemäß SENT-Datenprotokoll. Es können bis zu zwei ES1321-In-SENT Devices eingefügt werden, d.h. hardwareseitig sind zwei parallel arbeitende Empfangseinheiten vorhanden.



**Abb. 4-14** Format eines Signals gemäß SENT-Datenprotokoll

#### Serial Messages

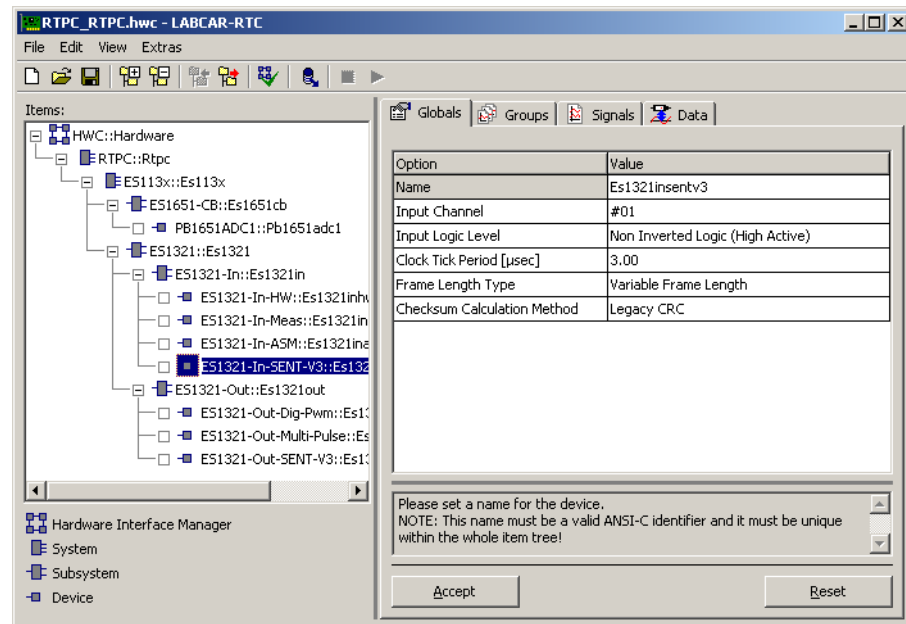
Bit 2 und Bit 3 des Status-Nibble stehen für den sogenannten Serial Message Channel (SMC) zur Verfügung.



Dabei werden über 16 oder 18 Frames die Daten der Bits 2 und 3 des Status-Nibble aufgesammelt und im Speicher der ES1321.1 zwischengespeichert.

Die Auswertung der Daten wird in einem C-Code-Modul in LABCAR-RTPC durchgeführt. Das Modul ist als Open Source Code verfügbar und befindet sich im Ordner \MISC des LABCAR-OPERATOR Installationsmediums.

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES1321-In-SENT Devices.



**Abb. 4-15** Die Registerkarte „Globals“ des ES1321-In-SENT Devices

#### *Input Channel*

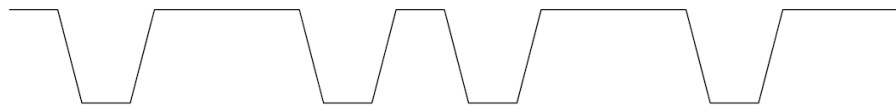
Jeder beliebige Eingabekanal kann verwendet werden – dabei werden die Pegel- bzw. Schwelleneinstellungen dieses Kanals im ES1321-In-HW Device verwendet. Dabei ist darauf zu achten, dass diese der Pegelspezifikation des SENT-Transmitters entsprechen (typisch: inaktiv = GND, aktiv = 5 Volt).

#### *Input Logic Level*

Diese Einstellung legt fest, ob die Pegel der SENT-Signale invertiert oder nicht invertiert ausgegeben werden.

- Non inverted Logic (High Active)

Abb. 4-16 zeigt den Signalverlauf einer nicht-invertierten Logik.



**Abb. 4-16** Nicht-invertierte Logik im Kontext der SENT-Spezifikation

- Inverted Logic (Low Active)

Abb. 4-17 zeigt den Signalverlauf einer invertierten Logik.



**Abb. 4-17** Invertierte Logik im Kontext der SENT-Spezifikation

### *Clock Tick Period [ $\mu$ sec]*

---

Länge (in  $\mu$ s) eines „Clock Ticks“ – daraus ergibt sich dann die Länge des „Calibration/Synchronization Pulse“ und der „Nibble Pulses“.

### *Frame Length Type*

---

Die Übertragungsart der SENT-Frames.

- Variable Frame Length  
Ein SENT-Frame wird unmittelbar nach dem anderen erwartet. Es gibt keinen „Pause“ Puls.
- Constant Frame Length  
Zwischen zwei SENT-Frames wird ein „Pause“ Puls erwartet. Die Länge des „Pause“ Pulses ist so gewählt, dass der komplette SENT-Frame (inkl. „Pause“) eine konstante Länge hat.

### *Checksum Calculation Method*

---

Die Berechnungsmethode für die 4-Bit-Prüfsumme in der ES1321.1.

- None  
Es findet keine Prüfsummenüberprüfung statt.
- Recommended CRC  
Überprüfung der Prüfsumme entsprechend der SENT Spezifikation Jan 2010.  
Sollte die berechnete Prüfsumme nicht mit der übertragenen übereinstimmen, wird der „ChecksumErrorCounter“ erhöht und der Datenwert nicht übernommen.
- Legacy CRC  
Überprüfung der Prüfsumme in der ES1321.1 entsprechend der SENT Spezifikation April 2007 und Feb 2008.  
Sollte die berechnete Prüfsumme nicht mit der übertragenen übereinstimmen, wird der „ChecksumErrorCounter“ erhöht und der Datenwert nicht übernommen.

4.6.2 Signals (ES1321-In-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES1321-In-SENT Devices.

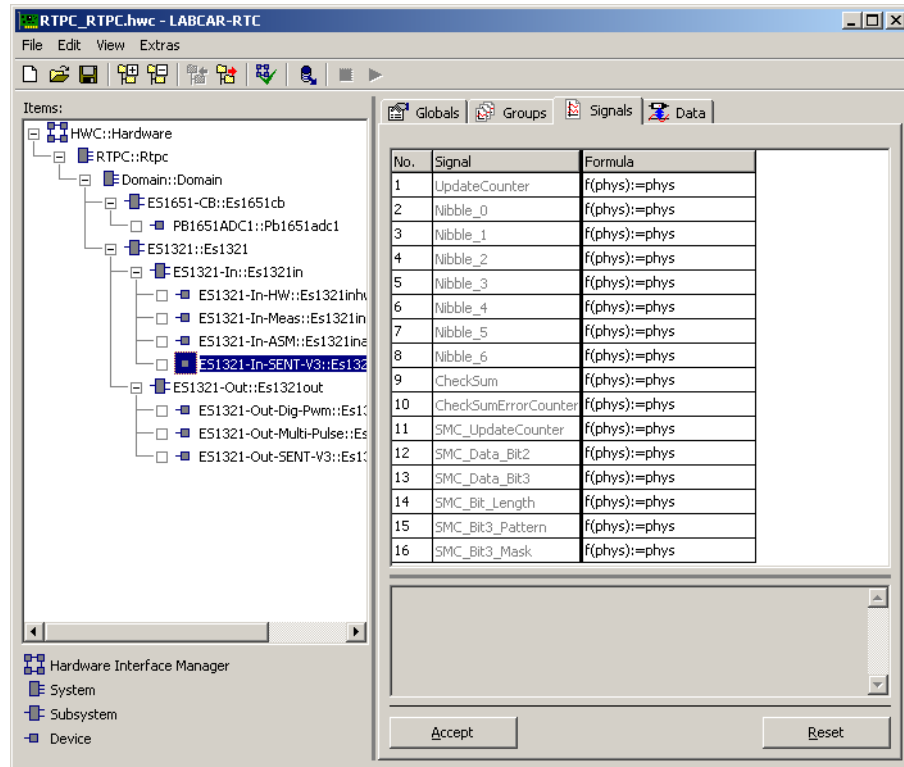
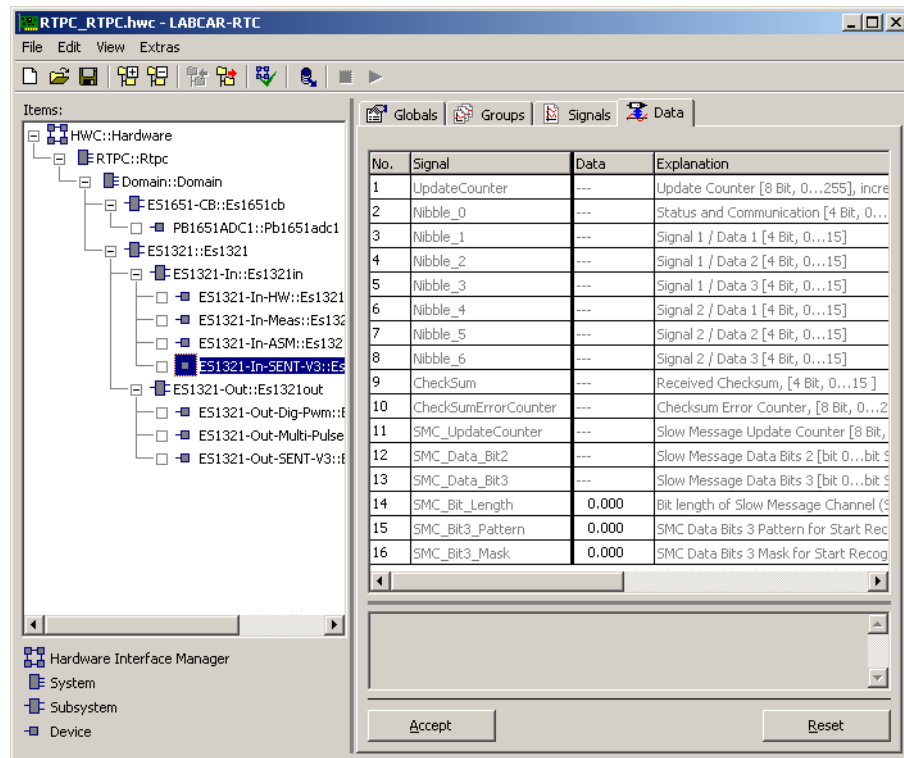


Abb. 4-18 Die Registerkarte „Signals“ des ES1321-In-SENT Devices

### 4.6.3 Data (ES1321-In-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES1321-In-SENT Devices.



**Abb. 4-19** Die Registerkarte „Data“ des ES1321-In-SENT Devices

Das ES1321-In-SENT Device implementiert eine Empfangssignalgruppe „SentIn“. Diese liefert die uninterpretierten Daten des letzten, vollständig empfangenen SENT-Datenwortes und Statusinformationen. Die Interpretation der Daten ist abhängig vom Protokolltyp des Senders.

#### *UpdateCounter*

Umlaufender Zähler, der inkrementiert wird, wenn neue Daten zur Verfügung stehen – der Wertebereich beträgt 0...255.

#### *Nibble\_0 ... Nibble\_6*

Werte der 7 Nibbles (1 Status-Nibble und 2x3 Daten-Nibbles) des zuletzt empfangenen SENT-Datenwortes – der gültige Wertebereich beträgt 0...15.

#### *CheckSum*

Vom SENT-Frame übertragene Prüfsumme.

#### *CheckSumErrorCounter*

Umlaufender Zähler, der inkrementiert wird, wenn eine SENT-Botschaft in der ES1321.1 vollständig empfangen wurde, aber die Prüfsumme falsch war.

### *SMC\_UpdateCounter*

---

Umlaufender 8-Bit-Zähler, der inkrementiert wird, wenn die ES1321.1 eine SMC-Botschaft vollständig empfangen hat, welche nun an den Signalen „SMC\_Data\_Bit2“ und „SMC\_Data\_Bit3“ zur Auswertung zur Verfügung steht.

### *SMC\_Data\_Bit2*

---

Datenwort, welches den empfangenen Bit 2-Datenstrom bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4). Die höheren Bitpositionen enthalten die später empfangenen Daten.

### *SMC\_Data\_Bit3*

---

Datenwort, welches den empfangenen Bit 3-Datenstrom bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4). Die höheren Bitpositionen enthalten die später empfangenen Daten.

### *SMC\_Bit\_Length*

---

Eingangssignal zur Steuerung der SMC-Empfangseinheit der ES1321.1. Der Wert „0“ deaktiviert den SMC-Empfang, ein Wert > 0 (... 31) aktiviert den SMC-Empfang und gibt die Anzahl der Bits für einen SMC-Zyklus vor.

Als gültige Werte entsprechend der SENT-Spezifikation sind 16 (Short Serial Message) bzw. 18 (Enhanced Serial Message) definiert.

### *SMC\_Bit3\_Pattern*

---

Eingangssignal, zur Steuerung der SMC Empfangseinheit der ES1321.1. Es gibt ein Bit-Muster zur Gültigkeitsprüfung einer SMC für Bit 3 vor.

Eine Botschaft wird als gültig erkannt wenn gilt:

```
(SMC_Data_Bit3 bitwiseXor SMC_Bit3_Pattern)
bitwiseAnd SMC_Bit3_Mask == 0.
```

D.h. „SMC\_Bit3\_Pattern“ definiert den Wert der relevanten Bits.

### *SMC\_Bit3\_Mask*

---

Eingangssignal zur Steuerung der SMC Empfangseinheit der ES1321.1. Sie gibt eine Bit-Maske zur Gültigkeitsprüfung einer SMC für Bit 3 vor.

Eine Botschaft wird als gültig erkannt wenn gilt:

```
SMC_Data_Bit3 bitwiseXor SMC_Bit3_Pattern)
bitwiseAnd SMC_Bit3_Mask == 0
```

D.h. „SMC\_Bit3\_Mask“ markiert die relevanten Bitpositionen mit einer Eins.

Für den Empfang einer SMC Botschaft stellt die ES1321.1 und die RTIO Einbindung nur generische Funktionen zur Verfügung. Die Steuerung, Dekodierung der Daten und die Überprüfung der Prüfsumme erfolgt ausserhalb der RTIO.

Ein C-Code-Modul „SentInSMC\_Control“ mit Grundfunktionen steht hierzu als Vorlage zur Verfügung. Alle SMC\_\* Signale der RTIO werden dazu mit den entsprechenden Ports des C-Code-Moduls verbunden.

Das C-Code-Modul muss jedoch noch an die spezifische Anwendung angepasst werden durch:

- Festlegung „SMC Format Mode“ entsprechend der SENT Spezifikation
- Umrechnung der Datenwerte der benötigten Msg-IDs in physikalische Größen
- Bereitstellen der Datenwerte an den Output-Ports

Folgende Werte sind gemäß J2716 Jan 2010 definiert:

- 16 Bit Short Serial Message (E\_SMC\_16Bit)
  - SMC\_Bit3\_Pattern = 0x8000;
  - SMC\_Bit3\_Mask = 0xFFFF;
  - SMC\_Bit\_Length = 16;
- 18 Bit Enhanced Serial Message (12 Bit Data, 8 Bit Msg ID, E\_SMC\_18BitC0):
  - SMC\_Bit3\_Pattern = 0x3F000;
  - SMC\_Bit3\_Mask = 0x3F421;
  - SMC\_Bit\_Length = 18;
- 18 Bit Enhanced Serial Message (16 Bit Data, 4 Bit Msg ID, E\_SMC\_18BitC1):
  - SMC\_Bit3\_Pattern = 0x3F400;
  - SMC\_Bit3\_Mask = 0x3F421;
  - SMC\_Bit\_Length = 18;

## 4.7 ES1321-In-ASM Device

---

### 4.7.1 Globals (ES1321-In-ASM Device)

---

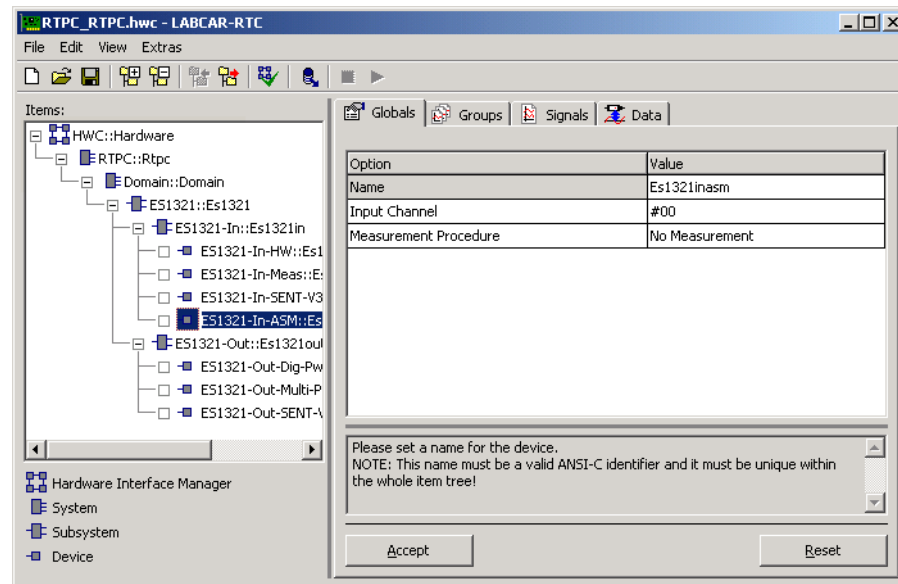
Dieses RTIO-Element ermöglicht es, mit der ES1321.1 bestimmte winkelsynchrone Messfunktionen (siehe „Measurement Procedure“ auf Seite 101) durchzuführen. Grundsätzlich wird dabei der Winkel einer bestimmten Flanke ermittelt, die innerhalb eines Winkelfensters auftritt.

Als Eingang dient dabei ein beliebiger Eingangskanal, welcher im ES1321-In-HW Device entsprechend konfiguriert werden muss.

Für jedes ES1321-In-ASM Device stehen dabei vier Messkanäle des selben Messverfahrens, aber mit unterschiedlichen Winkelfenstern, zur Verfügung. Es können zwei ES1321-In-ASM Devices eingefügt werden, d.h. es stehen Ressourcen für insgesamt acht Messkanäle zur Verfügung.



Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES1321-In-ASM Devices.



**Abb. 4-20** Die Registerkarte „Globals“ des ES1321-In-ASM Devices

In diesem Register können folgende Parameter eingestellt werden:

#### *Input Channel*

Hier wird der zu verwendende Eingangskanal ausgewählt – dabei werden die Pegel- und Schwellwerteneinstellungen des ES1321-In-HW Devices verwendet. Alle vier Software-Messkanäle dieses ES1321-In-ASM Devices beziehen sich auf diesen Hardware-Kanal.

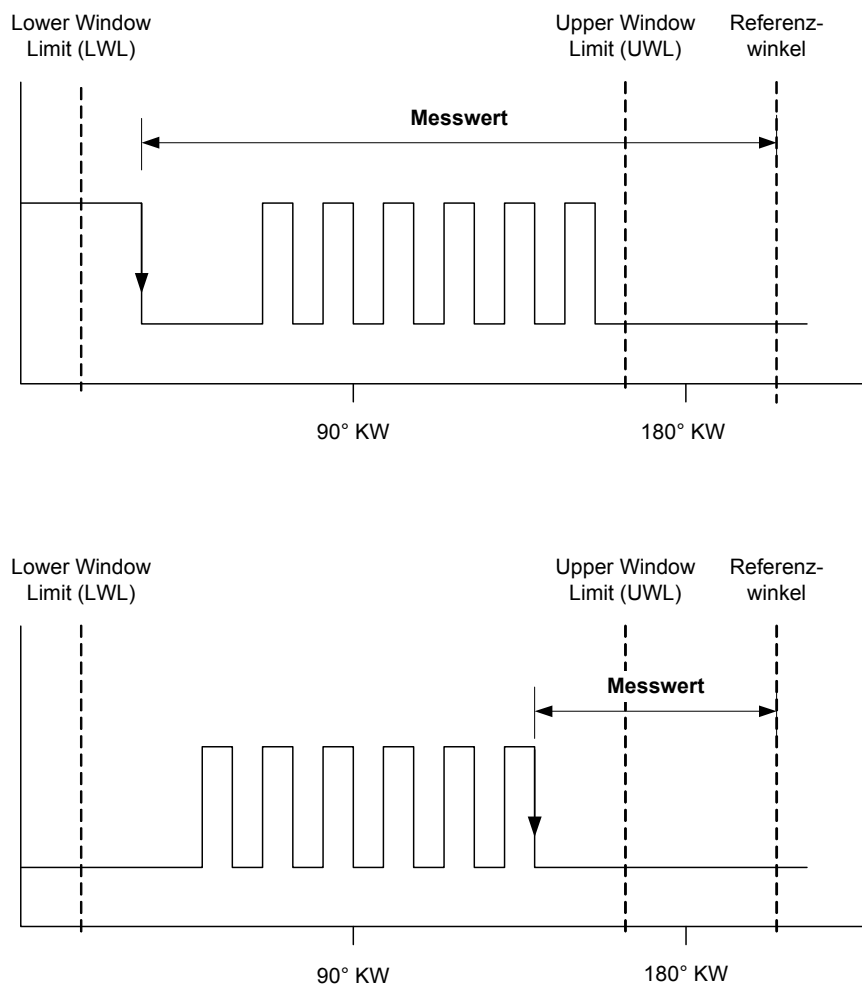
#### *Measurement Procedure*

Hier findet die Auswahl des verwendeten Messverfahrens für alle vier Software-Messkanäle des RTIO-Elements statt – insgesamt stehen folgende vier Messmethoden zur Verfügung:

- Angle of First Rising Edge of Pulse Sequence [°CA]
- Angle of First Falling Edge of Pulse Sequence [°CA]
- Angle of Last Rising Edge of Pulse Sequence [°CA]
- Angle of Last Falling Edge of Pulse Sequence [°CA]

Dabei wird jeweils der Winkelwert der ersten bzw. letzten steigenden bzw. fallenden Flanke gemessen, welche innerhalb eines Winkelfensters (definiert durch LWL und UWL, siehe Abb. 4-21 auf Seite 102) aufgetreten ist. Der Messwert ergibt sich aus einem Referenzwinkelwert abzüglich dieses Winkels. Flanken vor dem Referenzwinkel ergeben ein positives Ergebnis, Flanken nach dem Referenzwinkel ein negatives Ergebnis.

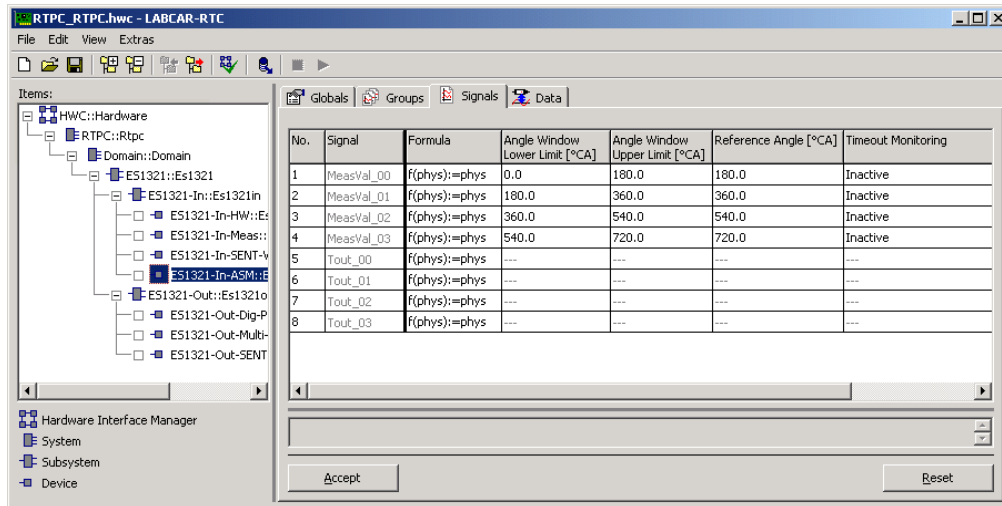
Tritt keine entsprechende Flanke im Winkelfenster auf, wird kein Ergebnis übermittelt und es wird eine Timeout-Bedingung erkannt.



**Abb. 4-21** Messwerte bei „Angle of First Falling Edge of Pulse Sequence“ (oben) und „Angle of Last Falling Edge of Pulse Sequence“ (unten)

#### 4.7.2 Signals (ES1321-In-ASM Device)

Das ES1321-In-ASM Device implementiert eine „MeasVal“ Empfangssignalgruppe.



**Abb. 4-22** Die Registerkarte „Signals“ des ES1321-In-ASM Devices

Folgende Signale werden angelegt:

*MeasVal\_xx [xx = 00...03]*

Die Ergebniswerte für Software-Messkanäle 0...3 (Wertebereich - 720.0...720.0 °KW, -8888.0 falls Kanal unbenutzt)

*Tout\_xx [xx= 00...03]*

Timeout-Signale für die Software-Messkanäle 0...3

- 0 = kein Timeout
- 1 = Timeout erkannt
- 2 = Timeout nicht konfiguriert
- 3 = Winkelinterface nicht synchron

In der Registerkarte „Signals“ können spezifisch für jeden Messkanal folgende Einstellungen vorgenommen werden.

*Angle Window Lower Limit*

Der Startwinkel des Messfensters (entspricht der Größe „LWL“ in Abb. 4-21 auf Seite 102).

*Angle Window Upper Limit*

Der Stoppwinkel des Fensters (entspricht der Größe „UWL“ in Abb. 4-21 auf Seite 102)

*Reference Angle*

Der Bezugswinkel für die Berechnung des Messwertes.

### Timeout Monitoring

Die Betriebsart für Timeout-Behandlung.

- Inactive  
Keine Timeout-Erkennung, der vorhergehende Wert wird übermittelt
- Intvl Predef  
Der Ersatzwert für den Timeout-Fall wird übermittelt

### Default Timeout Value

Der Ersatzwert für den Timeout-Fall.

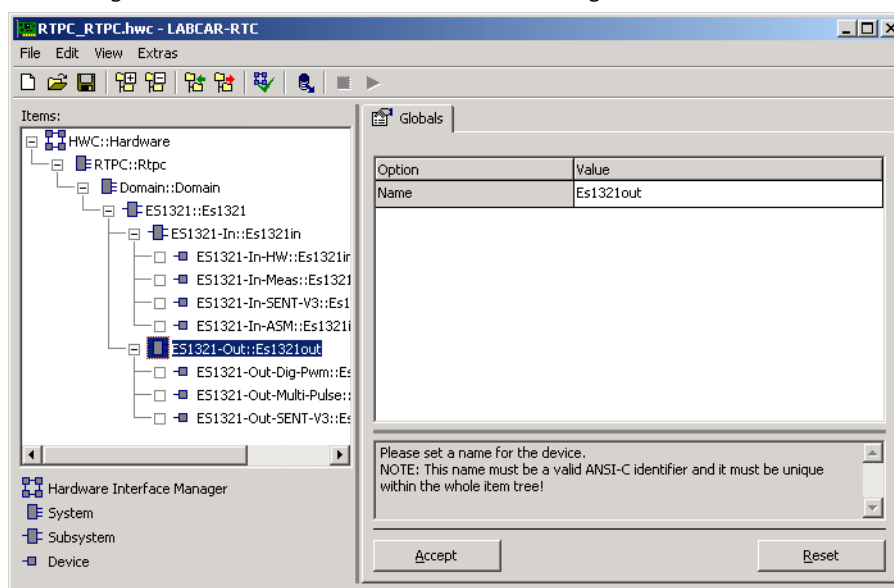
## 4.8 ES1321-Out Subsystem

### 4.8.1 Globals (ES1321-Out Subsystem)

Das ES1321-Out Subsystem dient zur Konfiguration und Ansteuerung der 16 Ausgangskanäle, an denen digitale Signale, pulsweitenmodulierte Signale, Pulssequenzen und SENT-Signale ausgegeben werden können. Es ist zudem möglich, die Ausgänge gegen zwei unterschiedliche positive Referenzspannungen und gegen Masse zu schalten.

Unter dieses Subsystem kann man ein ES1321-Out-Dig-Pwm, zwei ES1321-Out-Multi-Pulse und bis zu vier ES1321-Out-SENT Devices einfügen.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.

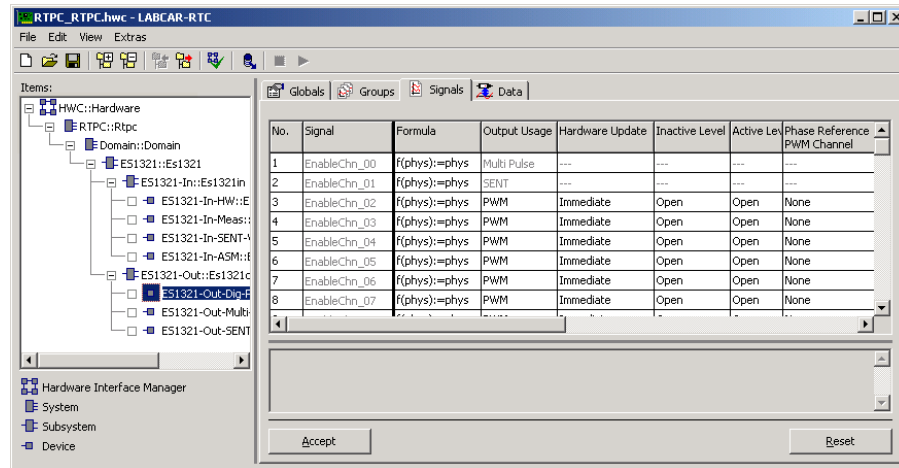


**Abb. 4-23** Die Registerkarte „Globals“ des ES1321-Out Subsystem

## 4.9 ES1321-Out-Dig-Pwm Device

### 4.9.1 Signals (ES1321-Out-Dig-Pwm Device)

Abb. 4-24 zeigt die Registerkarte „Signals“ des ES1321-Out-Dig-Pwm Devices.



**Abb. 4-24** Die Registerkarte „Signals“ des ES1321-Out-Dig-Pwm Devices

In diesem Register können Sie folgende Optionen festlegen:

#### Output Usage

In dieser Ansicht kann zwischen PWM und Digital Out ausgewählt werden.

Ist bereits ein ES1321-Out-Multi-Pulse oder ein ES1321-Out-SENT Device hinzugefügt worden, werden in dieser Ansicht die dafür verwendeten Ausgänge ausgegraut.

- PWM  
Erzeugung eines pulsweitenmodulierten Signals – beschrieben durch Frequenz, Tastverhältnis und Phasenlage zu einem anderen PWM-Ausgang
- Digital Out  
Bei dieser Konfiguration des Ausgangskanals werden Signalpegel generiert.

#### Hardware Update

In diesem Listenfeld wird eingestellt, wann vom Modell oder vom Anwender an den „Frequency\_x“- und „DutyCycle\_x“-RTIO-Signalen (x = 0...15) vorgenommenen Wertänderungen von der ES1321.1-Hardware übernommen werden.

- Immediate  
Ein neuer Wert wird am Ausgang sofort sichtbar, d.h. der aktuelle Impuls wird abgebrochen und mit neuen Werten gestartet.
- Cycle End  
Ein neuer Wert wird erst nach Beendigung des kompletten Impulses am Ausgang sichtbar.
- RTIO Controlled  
Ein neuer Wert wird nach „0 → 1“ Übergang des „SyncSgl“-Signals am Ausgang sichtbar.

### *Inactive Level*

---

In diesem Listenfeld wird der inaktive Zustand des PWM-Signals eingestellt. Der inaktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, so nimmt es während 40% der Periodendauer den inaktiven Signalzustand an.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

### *Active Level*

---

In diesem Listenfeld wird der aktive Zustand des PWM-Signals eingestellt.

Der aktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, nimmt es während 60% der Periodendauer den aktiven Signalzustand an.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

### *Phase Reference PWM Channel*

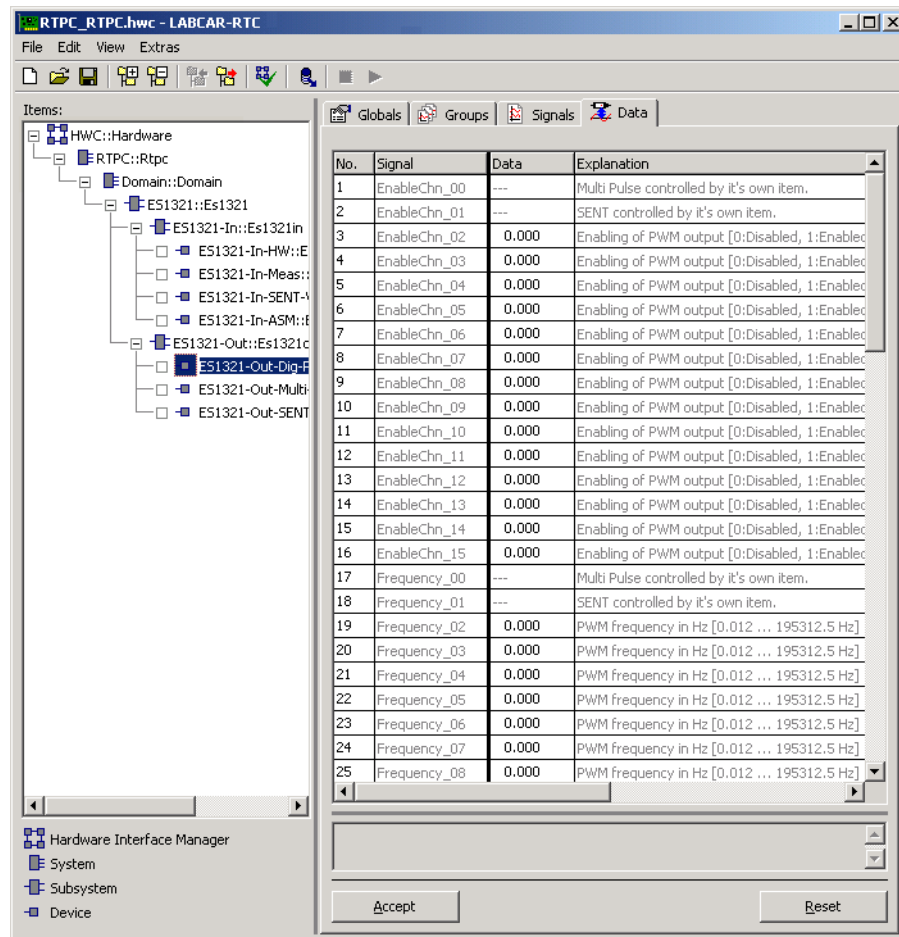
---

Die ES1321.1 bietet die Möglichkeit, Phasenverschiebungen zwischen Ausgabekanälen mit gleicher PWM-Frequenz definiert einzustellen. Hierzu ist einer dieser PWM-Ausgänge als Phasen-Referenz-Kanal zu bestimmen.

Im Listenfeld „Phase Reference PWM Channel“ der übrigen - zu diesem Referenz-Kanal in definierten Phasenbeziehungen stehenden - Ausgänge ist dann die Nummer des Referenz-Kanals einzutragen.

#### 4.9.2 Data (ES1321-Out-Dig-Pwm Device)

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES1321-Out-Dig-Pwm Devices.



**Abb. 4-25** Die Registerkarte „Data“ des ES1321-Out-Dig-Pwm Devices

Die Registerkarte „Data“ enthält folgende Signale:

- EnableChn\_00...EnableChn\_15  
Aktiviert oder deaktiviert die Signalausgabe
  - = 0: Ausgang ist hochohmig, Signalausgabe ist inaktiv
  - = 1: Signalausgabe ist aktiv
- Frequency\_00..Frequency\_15  
Frequenz (in Hz) des jeweiligen PWM-Signals
- DutyCycle\_00..DutyCycle\_15  
Mit den Signalen „DutyCycle\_x“ wird das Tastverhältnis des jeweils zugeordneten PWM-Ausgangs eingestellt. Tastverhältnisse werden mit Werten zwischen 0,0 und 1,0 definiert. 0,0 entspricht einem Tastverhältnis von 0%, 1,0 einem Tastverhältnis von 100%.

- Phase  
Phasenverschiebung (in °) gegenüber dem Referenzkanal. Ein Wert von 360° entspricht einer Phasenverschiebung um einen kompletten Puls.
- SyncSgl  
Die ES1321.1 bietet die Möglichkeit, den Zeitpunkt zu steuern, an dem von RTIO bzw. dem Simulationsmodell an das Board übermittelten Frequenz- und Phasendaten von der Hardware übernommen werden. Die Steuerung erfolgt über die „Hardware Update“-Option der Registerkarte „Signals“ (siehe „Hardware Update“ auf Seite 105).  
Wird in diesem Listenfeld für einen oder mehrere Kanäle die Option „RTIO Controlled“ eingestellt, so werden die an diese Kanäle übermittelten Frequenz- und Tasterhältnisdaten erst dann von der Hardware übernommen, wenn auf dem „SyncSgl“-Signal ein Übergang von 0 nach 1 detektiert wurde.
- ChnState\_00...ChnState\_15  
Jeder Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers:
  - Bit 0: Ausgangspegel
  - Bit 1: Überstromerkennung am Low-Side Switch
  - Bit 2: Überstromerkennung am High-Side Switch
  - Bit 3: Fehler erkannt

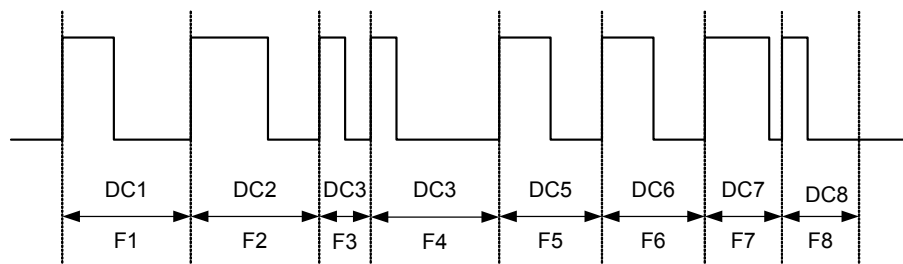
## 4.10 ES1321-Out-Multi-Pulse Device

---

### 4.10.1 Globals (ES1321-Out-Multi-Pulse Device)

---

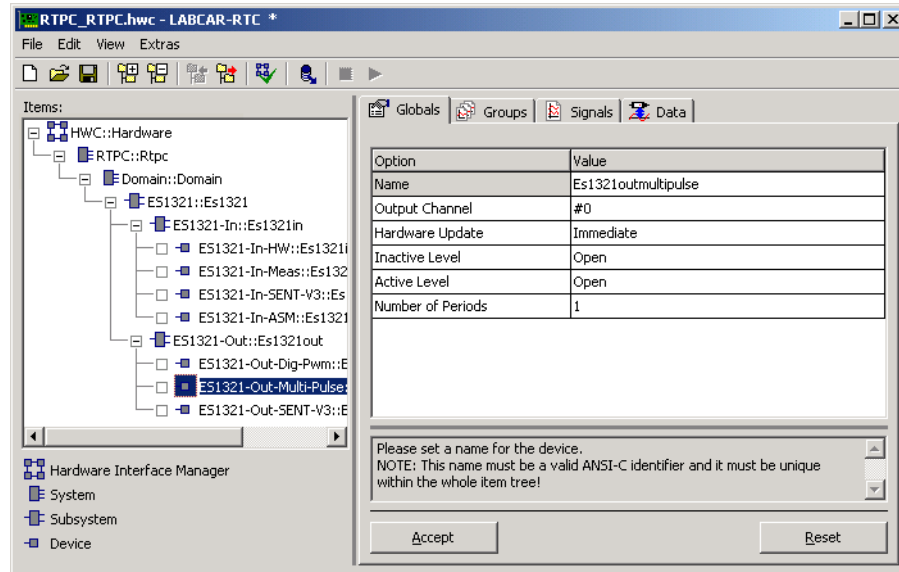
Eine Multi-Pulse-Sequenz kann aus bis zu acht Pulsen bestehen, von denen jeder durch eine Frequenz und ein Tastverhältnis definiert ist.



**Abb. 4-26** Eine Abfolge von acht Pulsen ( $F_x$  = Frequenz von Puls  $x$ ,  $DC_x$  = Tastverhältnis von Puls  $x$ )



Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES1321-Multi-Pulse Devices.



**Abb. 4-27** Die Registerkarte „Globals“ des ES1321-Multi-Pulse Devices

Die Bedeutung der verfügbaren Optionen wird im Folgenden beschrieben.

#### *Output Channel ID*

Der Hardware-Kanal der ES1321.1, der als Multi-Pulse-Ausgang verwendet wird.

#### *Hardware Update*

In diesem Listenfeld wird eingestellt, wann vom Modell oder vom Anwender am Pulspaket vorgenommenen Änderungen von der ES1321.1-Hardware übernommen werden.

- Immediate  
Ein neuer Wert wird am Ausgang sofort sichtbar, d.h. der aktuelle Impuls wird abgebrochen und mit neuen Werten gestartet.
- Cycle End  
Ein neuer Wert wird nach Beendigung des kompletten Impulses am Ausgang sichtbar.
- RTIO Controlled  
Ein neuer Wert wird nach „0 → 1“ Übergang des „SyncSgl“-Signals am Ausgang sichtbar.

#### *Inactive Level*

In diesem Listenfeld wird der inaktive Zustand des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.

- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

#### *Active Level*

---

In diesem Listenfeld wird der aktive Zustand des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

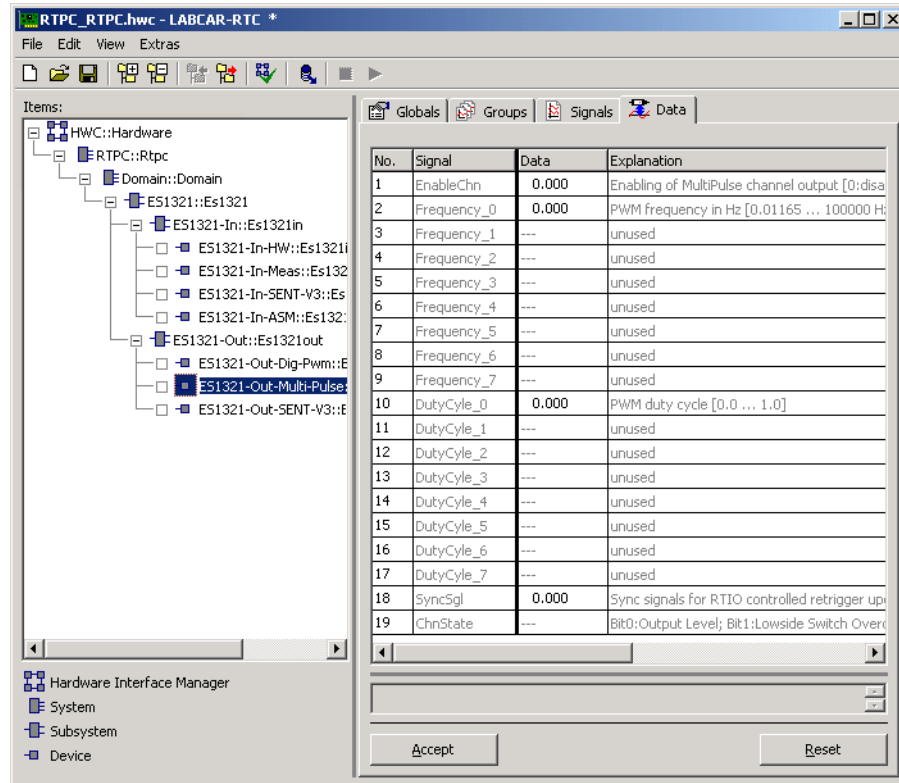
#### *Number of Periods*

---

Anzahl der Pulse der Sequenz (1..8)

## 4.10.2 Data (ES1321-Out-Multi-Pulse Device)

Die folgende Abbildung zeigt die Signale der Registerkarte „Data“



**Abb. 4-28** Die Registerkarte „Data“ des ES1321-Multi-Pulse Devices

Die Registerkarte „Data“ enthält folgende Signale:

- EnableChn  
Aktiviert oder deaktiviert den Multi-Pulse-Kanal
  - = 0: Ausgang ist hochohmig, Signalausgabe ist inaktiv
  - = 1: Signalausgabe ist aktiv
- Frequency\_0 .. Frequency\_7  
Frequenz (in Hz) des jeweiligen Pulses
- DutyCycle\_0 .. DutyCycle\_7  
Mit den Signalen „DutyCycle\_x“ wird das Tastverhältnis des jeweiligen Pulses eingestellt. Tastverhältnisse werden mit Werten zwischen 0,0 und 1,0 definiert. 0,0 entspricht einem Tastverhältnis von 0%, 1,0 einem Tastverhältnis von 100%.
- SyncSgl  
Die ES1321.1 bietet die Möglichkeit, den Zeitpunkt zu steuern, an dem von RTIO bzw. dem Simulationsmodell an das Board übermittelten Pulspaket-Daten von der Hardware übernommen werden. Die Steuerung erfolgt über die „Hardware Update“-Option der Registerkarte „Signals“ (siehe „Hardware Update“ auf Seite 109). Wird in diesem Listenfeld für einen oder mehrere Kanäle die Option „RTIO Controlled“ eingestellt, so werden

die an diese Kanäle übermittelten Pulspaket-Daten erst dann von der Hardware übernommen, wenn auf dem „SyncSgl“-Signal ein Übergang von 0 nach 1 detektiert wurde.

- ChnState

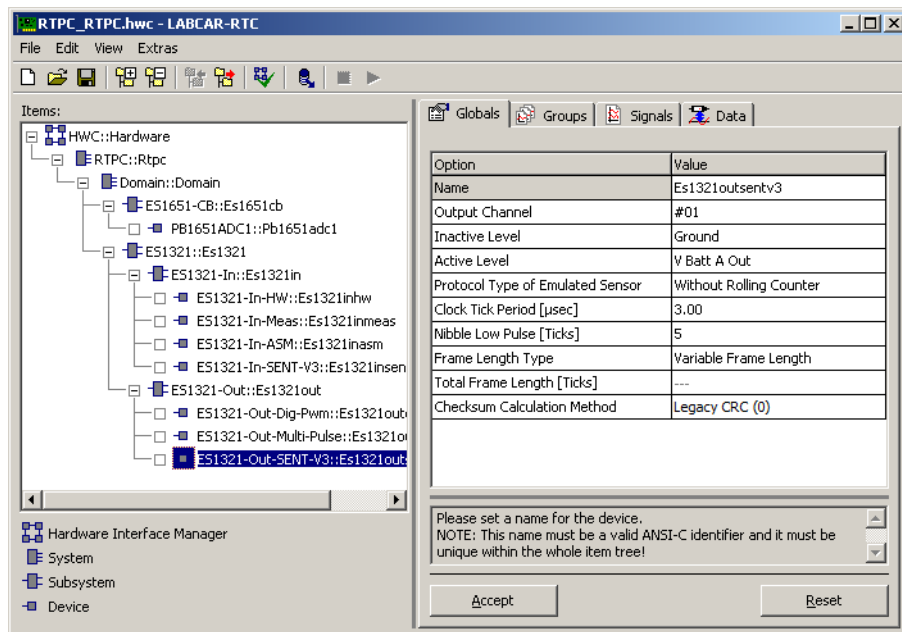
Der Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers:

- Bit 0: Ausgangspegel
- Bit 1: Überstromerkennung am Low-Side Switch
- Bit 2: Überstromerkennung am High-Side Switch
- Bit 3: Fehler erkannt

## 4.11 ES1321-Out-SENT Device

### 4.11.1 Globals (ES1321-Out-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES1321-Out-SENT Devices.



**Abb. 4-29** Die Registerkarte „Globals“ des ES1321-Out-SENT Devices

In dieser Registerkarte werden folgende Optionen eingestellt:

#### *Output Channel*

Auswahl der Hardwarekanals, der als SENT-Transmitter verwendet werden soll.

#### *Inactive Level*

In diesem Listenfeld wird der inaktive Zustand (logische Null) des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Ausgang ist hochohmig
- Ground  
Ausgang liegt auf Massepotential
- V Batt A Out  
Der Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der Ausgang gibt Batteriespannung B aus.

**Hinweis**

*Wenn ein nicht-invertierter Signalverlauf ausgegeben werden soll, dann sollte für den inaktiven Level „Ground“ eingetragen sein.*

Active Level

In diesem Listenfeld wird der aktive Zustand (logische Eins) des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Ausgang ist hochohmig
- Ground  
Ausgang liegt auf Massepotential
- V Batt A Out  
Der Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der Ausgang gibt Batteriespannung B aus.

**Hinweis**

*Wenn ein nicht-invertierter Signalverlauf ausgegeben werden soll, dann sollte für den aktiven Level „V Batt A“ oder „V Batt B“ eingetragen sein.*

Protocol Type of Emulated Sensor

Auswahl des verwendeten Protokolls nach der SENT-Spezifikation. Es stehen folgende Auswahlmöglichkeiten zur Verfügung:

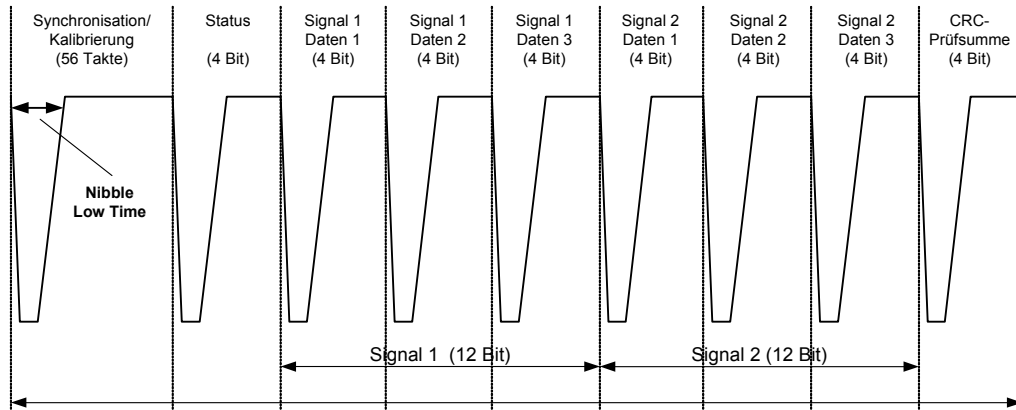
- Without Rolling Counter  
Die Werte von „Nibble\_4“ und „Nibble\_5“ (wie in der Registerkarte „Data“ angezeigt) werden berücksichtigt.
- With Rolling Counter Nibble 4/5  
Ein 8-Bit-Zählerwert, der mit den Nibble 4 und 5 übertragen wird. Die Werte von „Nibble\_4“ und „Nibble\_5“ (wie in der Registerkarte „Data“ angezeigt) werden nicht berücksichtigt.

Clock Tick Period [ $\mu$ sec]

Länge des „Clock Tick“, der kleinsten Zeiteinheit des SENT-Protokolls. Die Angabe erfolgt in  $\mu$ sec mit einer Auflösung von 20 nsec.

### Nibble Low Pulse [Ticks]

Anzahl der Ticks für den Low Pulse eines Nibbles – gemäß SENT-Spezifikation Jan 2010 sollte der Wert größer als 4 sein.



### Frame Length Type

Der Übertragungsmodus der SENT Frames.

- Variable Frame Length  
Ein SENT Frame wird unmittelbar nach dem anderen erwartet. Es gibt keinen „Pause“ Puls.
- Constant Frame Length  
Zwischen zwei SENT Frames wird ein „Pause“ Puls erwartet. Die Länge des „Pause“ Pulses ist gewählt, das das komplette SENT Frame (inkl. „Pause“ Puls) eine konstante Länge hat.

### Total Frame Length [Ticks]

Wenn für der Einstellung „Frame Length Type“ die Auswahl „Constant Frame Length“ gewählt wird, dann kann man hier die Länge des Frames einstellen.

### Checksum Calculation Method

Mit dieser Einstellung wird die Berechnungsmethode der 4-Bit Prüfsumme festgelegt.

- Recommended CRC (0) \*  
Berechnung entsprechend der SENT Spezifikation Jan 2010
- Legacy CRC (1) \*  
Berechnung entsprechend der SENT Spezifikation April 2007 und Feb 2008
- CRC = 0x0 (2) \*  
CRC Nibble hat immer den Wert „0x0“.
- CRC = 0xF (3) \*  
CRC Nibble hat immer den Wert „0xF“.

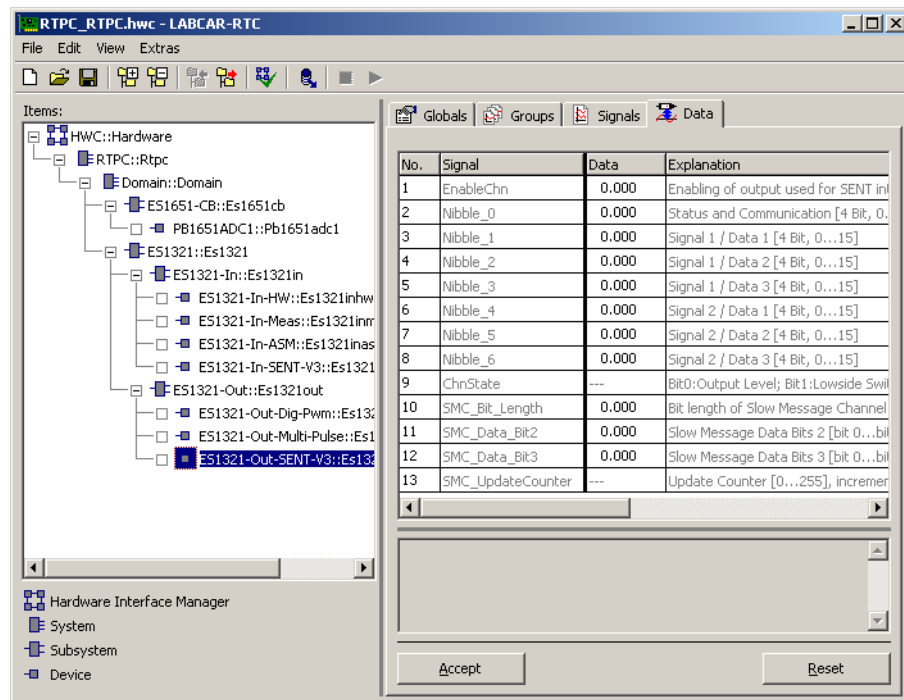
- CRC Mode 0...3 by Signal  
Mit dem Signal „Checksum\_Mode\_Data“ (siehe „Checksum\_Mode\_Data“ auf Seite 117) kann eine der Berechnungsmethoden 0...3 ausgewählt werden (jeder andere Wert führt zu CRC = 0).
- CRC Data 0...15 by Signal  
Das CRC Nibble kann mit dem Signal „Checksum\_Mode\_Data“ direkt vorgegeben werden.

### Hinweis

\* Die Werte in Klammern geben die jeweilige Modenummer des Eingangssignals für die Einstellung "CRC Mode ... by Signal" an.

#### 4.11.2 Data (ES1321-Out-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES1321-Out-SENT Devices.



**Abb. 4-30** Die Registerkarte „Data“ des ES1321-Out-SENT Devices

Die Registerkarte „Data“ enthält folgende Signale:

#### EnableChn

Aktiviert oder deaktiviert den für die SENT-Ausgabe verwendeten Hardwarekanal.

- = 0: Ausgang ist hochohmig, Signalausgabe ist inaktiv
- = 1: Signalausgabe ist aktiv

#### Nibble\_0..Nibble\_6

Der Wert der einzelnen Nibbles.

### *ChnState*

---

4-Bit Fehlerzustandskodierung des verwendeten Hardwarekanals. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers.

- Bit 0: Ausgangspegel
- Bit 1: Überstromerkennung am Low-Side Switch
- Bit 2: Überstromerkennung am High-Side Switch
- Bit 3: Fehler erkannt

### *SMC\_Bit\_Length*

---

Eingangssignal zur Steuerung der SMC Sendeeinheit der ES1321.1.

Der Wert „0“ deaktiviert SMC-Senden, d.h. es werden Bit 2 und Bit 3 des Nibble 0 übertragen. Ein Wert > 0 (... 31) aktiviert die Einblendung der Bits 2 und 3 aus den jeweiligen Datenwörtern. Der Wert gibt die Anzahl der Bits für einen SMC-Zyklus vor. Als gültige Werte sind 16 (Short Serial Message) bzw. 18 (Enhanced Serial Message) definiert.

### *SMC\_Data\_Bit2*

---

Eingangssignal, welches das Datenwort für Bit 2 bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4).

### *SMC\_Data\_Bit3*

---

Eingangssignal, welches das Datenwort für Bit 3 bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4).

### *SMC\_UpdateCounter*

---

Umlaufender 8 Bit Zähler – wird inkrementiert, wenn eine SMC-Botschaft vollständig übertragen wurde. Außerdem wird signalisiert, dass die ES1321.1 die Ausgabe einer neuen SMC Botschaft mit den aktuellen Werten von SMC Data\_Bit2 und SMC\_Data\_Bit3 gestartet hat. Der Zähler kann somit zur Synchronisation bei zyklischer Ausgabe einer Liste von Message IDs verwendet werden.

Es werden die Werte von SMC\_Data\_Bit2 und SMC\_Data\_Bit3 an dem Zeitpunkt übernommen, an dem eine neue Short Serial Message ausgegeben wird. Nach dieser Datenübernahme sollten die Werte SMC\_Data\_Bit2 und SMC\_Data\_Bit3 mit neuen Daten gefüllt werden. Dies wird dem Rechenmodell dadurch signalisiert, indem der Wert von „SMC\_UpdateCounter“ inkrementiert wird.

Für das Senden einer SMC Botschaft stellt die ES1321.1 und die RTIO-Einbindung nur generische Funktionen zur Verfügung. Die Steuerung, Kodierung der Daten und Berechnung der Prüfsumme erfolgt ausserhalb der RTIO. Dafür steht ein C-Code-Modul „SentOutSMC\_Control“ mit Grundfunktionen als Vorlage zur Verfügung. Alle SMC\_\* Signale der RTIO werden dazu mit den entsprechenden Ports des C-Code-Moduls verbunden werden.

Das C-Code-Modul muss jedoch noch an die spezifische Anwendung angepasst werden durch:

- Festlegung von „SMC Format Mode“
- Festlegung von „SMC\_Sequence“ (zyklische Ausgabe einer Msg\_ID Reihe)
- Bereitstellen der physikalischen Datenwerte an den Input-Ports



- Umrechnen der Datenwerte und Zuweisen der Werte in „DataField[MsgId]“

#### *Checksum\_Mode\_Data*

---

Wenn für „Checksum Calculation Method“ (siehe Seite 114) „CRC Mode 0...3 by Signal“ gewählt wurde, kann hier eine der Berechnungsmethoden 0...3 festgelegt werden.

Wenn für „Checksum Calculation Method“ „CRC Data 0...15 by Signal“ gewählt wurde, kann hier das CRC Nibble direkt vorgegeben werden.



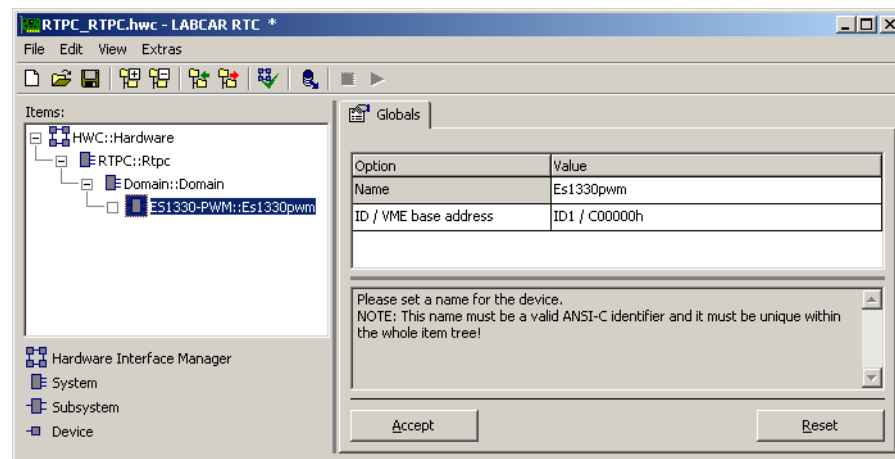
## 5 ES1330.1 PWM I/O Counter Board

Das ES1330.1 PWM I/O Counter Board ermöglicht die Erfassung und Ausgabe von PWM-Signalen. Die ES1330.1 ist mit sechs Zählerbausteinen (Am9513A) bestückt. Jeder Zählerbaustein stellt fünf 16-Bit breite Zähler mit einer Zählfrequenz von 4 MHz zur Verfügung. Dreißig Zählereingänge und sechzehn Zählerausgänge sind auf der ES1330.1 in sechs interne Ports „Port 1 .. Port 6“ aufgeteilt. Jeder Zählerbaustein belegt einen dieser Ports. Der Zugriff auf die Ports geschieht über den SUB-D Stecker der ES1330.1. In der Items-Liste des RTIO-Editors wird das ES1330.1 PWM I/O Counter Board als Subsystem und der Zählerbaustein als Device eingefügt.

### 5.1 ES1330-PWM Subsystem

#### 5.1.1 Globals (ES1330-PWM Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES1330-PWM Subsystems beschrieben.



**Abb. 5-1** Die Registerkarte „Globals“ des ES1330-PWM Subsystems

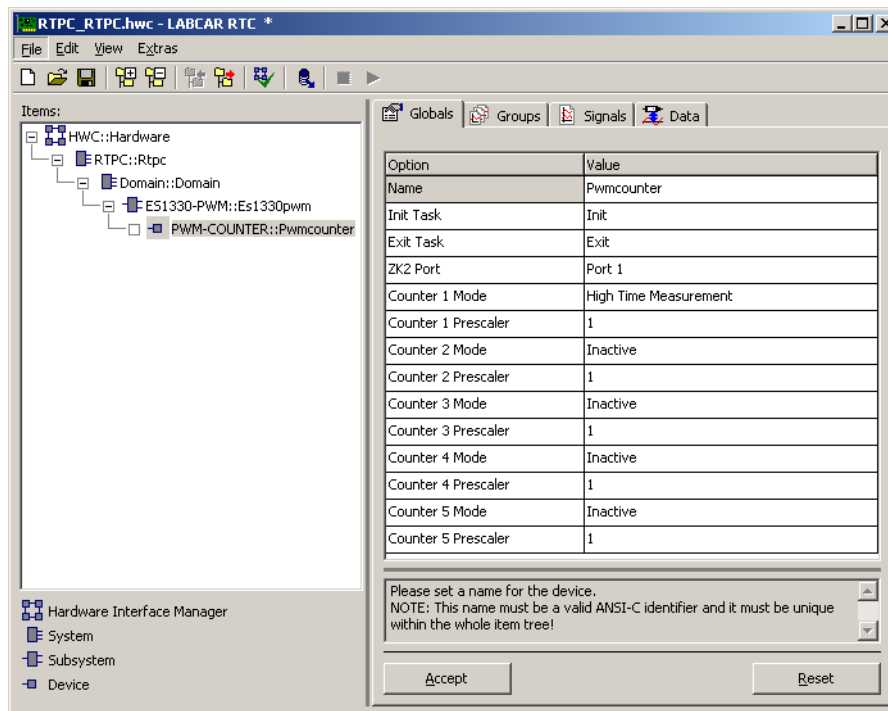
#### *ID / VME base address*

In dieser Zeile wird die Einstellung der VME-Basisadresse vorgenommen. Diese Einstellung muss mit den Jumper-Einstellungen der ES1330.1 übereinstimmen. Für die ES1330.1 können vier unterschiedliche VME-Basisadressen gewählt werden (ID1/C00000h, ID2/C00100h, ID3/C00200h, ID4/C00300h), dadurch können bis zu vier ES1330.1 PWM I/O Counter Board in einem ETAS Experimentalsystem betrieben werden. Die ES1330.1 belegt einen Adressbereich von 256 Worten ab der Basisadresse.

## 5.2 PWM-COUNTER Device

### 5.2.1 Globals (PWM-COUNTER Device)

In diesem Abschnitt werden die globalen Einstellungen des PWM-COUNTER Device beschrieben.



**Abb. 5-2** Die Registerkarte „Globals“ des PWM-COUNTER Device

#### *Init Task*

In dieser Zeile wird die Task zur Initialisierung der ES1330.1 zugeordnet (Type: Init/Application Mode: active).

#### *Exit Task*

In dieser Zeile wird die Task zugeordnet, die bei einem Stopp des Experiments mit der ES1330.1 ausgeführt werden soll (Type: Init / Application Mode: inactive).

#### *ZK2 Port*

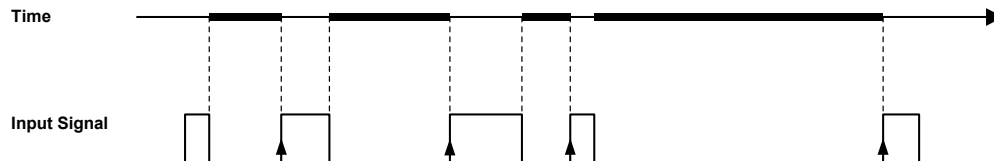
In dieser Zeile wird der Port (und damit der Zählerbaustein) zugeordnet. Jeder Zählerbaustein belegt einen eigenen Port.

#### *Counter 1 Mode .. Counter 5 Mode*

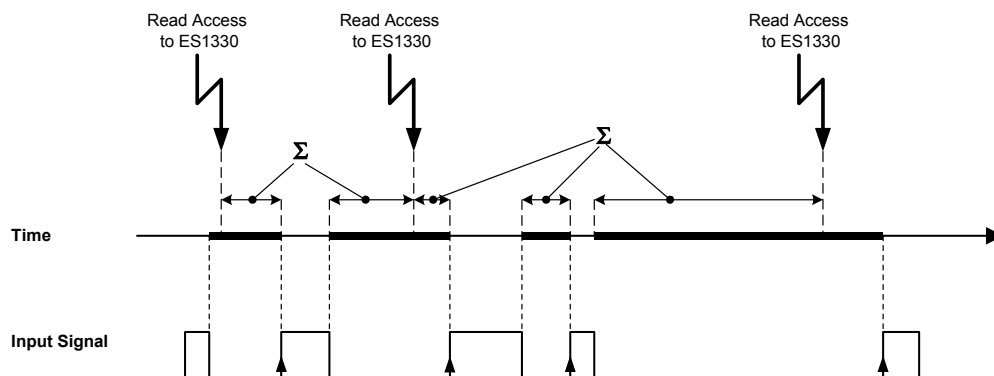
In diesen Zeilen wird der Zählermodus für jeden der fünf Zähler des Zählerbausteins gewählt. Die folgende Liste zeigt, welche Zählermodi ausgewählt werden können.

- Inactive  
Der Zähler wird nicht verwendet.

- PWM-Generator (PWM\_Gen)  
Der Zähler wird genutzt, um PWM-Signale zu erzeugen.
- Period-Measurement (P\_Meas)  
Der Zähler wird genutzt, um die Periode eines PWM-Signals zu messen.
- Low Time Measurement  
Bei diesem Modus wird die Dauer eines jeden Pulses mit aktivem Pegel (hier: low-aktiv) gemessen.



- High Time Measurement  
Bei diesem Modus wird die Dauer eines jeden Pulses mit aktivem Pegel (hier: high-aktiv) gemessen.
- Additive Low Time Measurement  
Das Simulationstarget erfasst Messwerte von der ES1330.1 in der Regel in einem periodischen Raster. Bei den additiven Messungen wird die Zeit gemessen, in der das Signal zwischen zwei aufeinanderfolgenden Leseoperationen auf das Messregister aktiv (hier: low-aktiv) war.



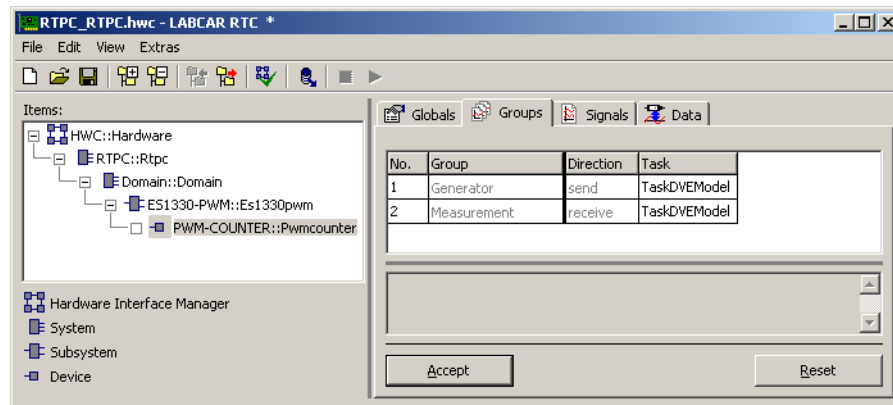
- Additive High Time Measurement  
Das Simulationstarget erfasst Messwerte von der ES1330.1 in der Regel in einem periodischen Raster. Bei den additiven Messungen wird die Zeit gemessen, in der das Signal zwischen zwei aufeinanderfolgenden Leseoperationen auf das Messregister aktiv (hier: high-aktiv) war.+

#### Counter 1 Prescaler .. Counter 5 Prescaler

In diesen Zeilen kann ein Vorteiler für die Eingangsfrequenz von 4 MHz genutzt werden. Die geteilte Eingangsfrequenz dient dann als Taktsignal für die ersten vier Zähler. Der Vorteiler ermöglicht das Erzeugen und Auswerten von PWM-Signalen mit einer längeren Periodendauer, allerdings nimmt dadurch die Auflösung der PWM-Signale ab. Folgende Vorteiler können gewählt werden: 1, 10, 100, 1000 und 10000.

### 5.2.2 Groups (PWM-COUNTER Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des PWM-COUNTER Device beschrieben.

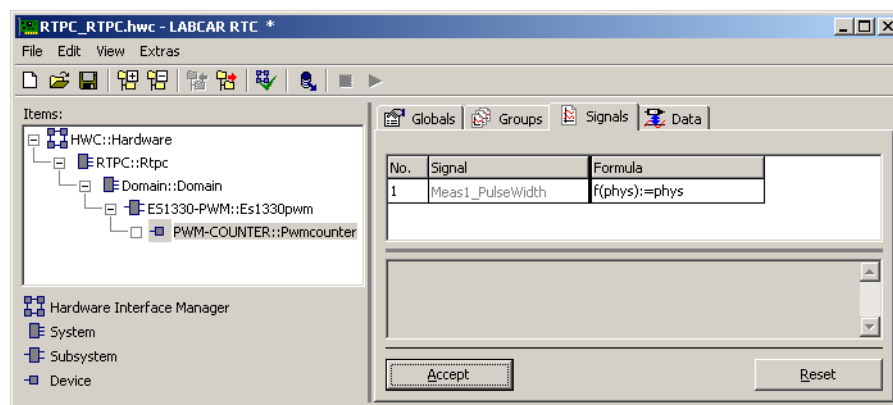


**Abb. 5-3** Die Registerkarte „Groups“ des PWM-COUNTER Device

Für das ES1330-PWM Subsystem sind in der Registerkarte „Groups“ keine Item-spezifischen Spalten definiert.

### 5.2.3 Signals (PWM-COUNTER Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PWM-COUNTER Device beschrieben.



**Abb. 5-4** Die Registerkarte „Signals“ des PWM-COUNTER Device

No.

Die Anzahl und Bedeutung der Signale wird durch die „ZK2 Port“ und „Counter X Mode“ Einstellungen aus der Registerkarte „Globals“ festgelegt.

*Signal*

In dieser Spalte sind Namen (ANSI-C) für die Signale fest vorgegeben. Es gibt folgende Namenskonvention (X = Nummer des Zählers):

- GenX\_Frequency  
Frequenz des PWM-Ausgabekanals in Hz
- GenX\_DutyCycle  
Tastgrad des PWM-Ausgabekanals in %

- MeasX\_Period  
Periodendauer des PWM-Auswertekanals in sec
- MeasX\_PulseWidth  
Ergebnis der Pulsweitenmessung (Einzelpulse oder additiv)





## 6 ES1331.1 Signal Generator Board

Das ES1331.1 Signal Generator Board dient zur Erzeugung von analogen und digitalen Ausgabesignalen in VMEbus-Systemen. Die Einschubkarte verfügt über einen eigenen digitalen Signalprozessor (DSP), der unabhängig vom Hauptprozessor des VMEbus-Systems arbeitet. Durch den Einsatz eines DSP ist die Signalform der Ausgabesignale frei programmierbar und auch hohe Anforderungen an die Änderungsgeschwindigkeit von Amplitude, Frequenz und Pulsbreite lassen sich leicht erfüllen.

Zur Erzeugung der analogen Signale enthält die Einschubkarte vier Digital-Analog-Umsetzer mit 12 Bit Auflösung und einer maximalen Umsetzzeit von 6  $\mu$ s. Der Ausgangsspannungsbereich beträgt -10 V bis +10 V (bezogen auf das Massepotential des VMEbus-Systems).

Die digitalen Ausgabesignale werden über ein Aufsteckmodul generiert. Dort stehen acht Zähler zur Verfügung. Je zwei Zähler bilden einen Ausgabekanal. Die Adress-, Daten- und Steuerleitungen des DSP sind zum Aufsteckmodul geführt.

Die Karte umfasst unter anderen folgende Funktionen:

- Digitaler Signalprozessor TMS320C203 / TMS320C206
- Vier Digital-Analog-Umsetzer mit 12 Bit Auflösung und einem Ausgangsspannungsbereich von -10 V bis +10 V
- Aufsteckmodul zur Generierung digitaler Signale

Das ES1331.1 Signal Generator Board kann in VMEbus-Systemen überall da eingesetzt werden, wo analoge und digitale Signalverläufe mit unterschiedlichen Frequenzen, Amplituden und Pulsbreiten erzeugt werden sollen. Der digitale Signalprozessor (DSP) ermöglicht eine flexible Signalerzeugung unabhängig vom Systemprozessor.

Beispielhafte Einsatzgebiete sind:

- Generierung von aktiven oder passiven Drehzahlgebersignalen für Fahrdynamik-Systeme
- Generierung von Drehzahlsignalen für Getriebesteuergeräte

### Aufbau des ES1331 RTIO-Item-Baums

In RTIO wird die ES1331.1 durch Auswahl des ES1331-DSP Subsystems eingebunden. Auf der Ebene des ES1331-DSP Subsystems kann die gewünschte Anwendung der ES1331.1 ausgewählt bzw. festgelegt werden.

Hier ist das WheelSnrSim Device zur Simulation von aktiven und passiven Drehzahlfühlern wählbar (siehe Abb. 6-1). Falls in Zukunft weitere Anwendungen der ES1331.1 implementiert werden, so werden die zugehörigen RTIO-Items auf dieser Ebene eingebaut werden.

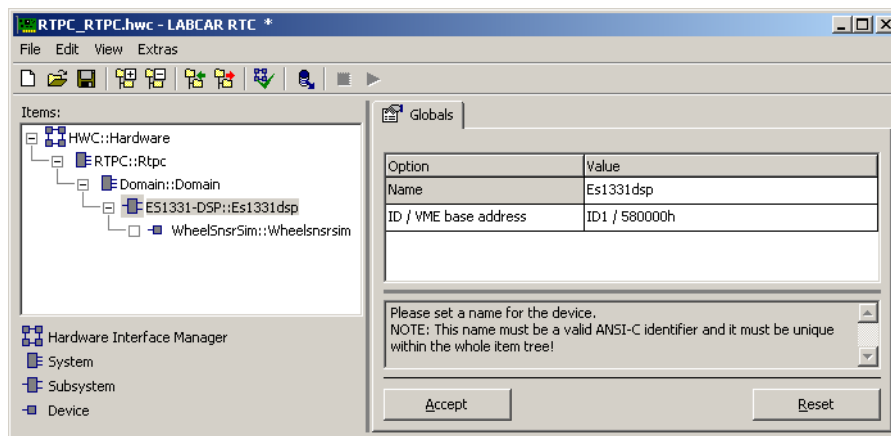
#### **Hinweis**

*Dem ES1331-DSP Subsystem kann nur ein WheelSnrSim Device zugeordnet werden. Der Versuch, ein weiteres hinzuzufügen, führt zu einer Fehlermeldung.*

## 6.1 ES1331-DSP Subsystem

### 6.1.1 Globals (ES1331-DSP Subsystem)

In der Registerkarte „Globals“ des ES1331-DSP Subsystems wird im Auswahlfeld „ID / VME base address“ die VMEbus-Basisadresse der ES1331.1 angegeben (siehe Abb. 6-1). Es stehen 2 Basisadressen zur Auswahl (ID1/580000h und ID2/5C0000h).



**Abb. 6-1** Die Registerkarte „Globals“ des ES1331-DSP Subsystems

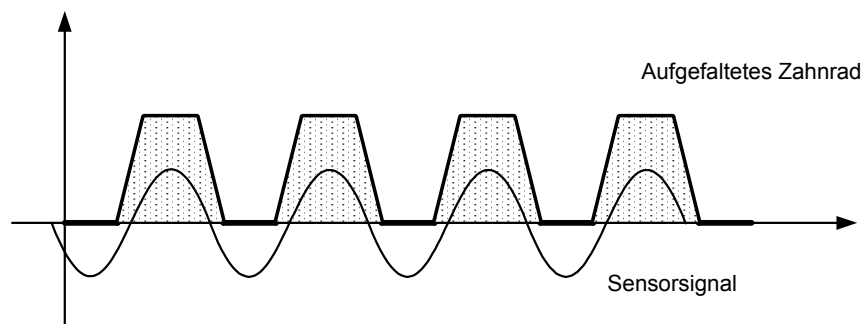
## 6.2 WheelSnrSim Device

### 6.2.1 Simulation von Raddrehzahlfühlern

In diesem Abschnitt finden Sie eine Einführung in die Signalausgabe bei passiven und aktiven Drehzahlfühlern.

#### *Signalausgabe bei passiven Raddrehzahlfühlern*

Ein passiver Drehzahlfühler besteht aus einem Hallsensor, der die Drehung eines Zahnrads erfasst, das auf dem Rad eines Kraftfahrzeugs angebracht ist. Das Sensorsignal kann durch eine Sinusfunktion angenähert werden (siehe Abb. 6-2). Jeder Zahn eines rotierenden Zahnrads erzeugt eine Periode der Sinuswelle. Eine Umdrehung des Zahnrads entspricht einer Radumdrehung.



**Abb. 6-2** Das Sensorsignal eines passiven Drehzahlfühlern

Führt man die nachfolgenden Formelzeichen ein:

$v_{\text{Rad}}$	Radgeschwindigkeit in m/s
$r_{\text{Rad}}$	Radradius in m
$T_{\text{Rad}}$	Periodendauer einer Rad- bzw. Zahnradumdrehung in s
$T_{\text{Sinus}}$	Periodendauer einer Sinuswelle in s
$f_{\text{Sinus}}$	Frequenz der Sinuswelle in Hz
$n_{\text{Zähne}}$	Anzahl der Zähne des Zahnrads

so kann man die folgende Beziehung zwischen der Periodendauer  $T_{\text{Sinus}}$  der Sinuswelle und der Dauer einer Radumdrehung  $T_{\text{Rad}}$  formulieren:

$$\text{Gln. 6-1} \quad T_{\text{Sinus}} = T_{\text{Rad}} / n_{\text{Zähne}}$$

Die Beziehung zwischen der Radgeschwindigkeit  $v_{\text{Rad}}$  und der Dauer einer Radumdrehung  $T_{\text{Rad}}$  lautet:

$$\text{Gln. 6-2} \quad v_{\text{Rad}} = 2\pi r_{\text{Rad}} / T_{\text{Rad}}$$

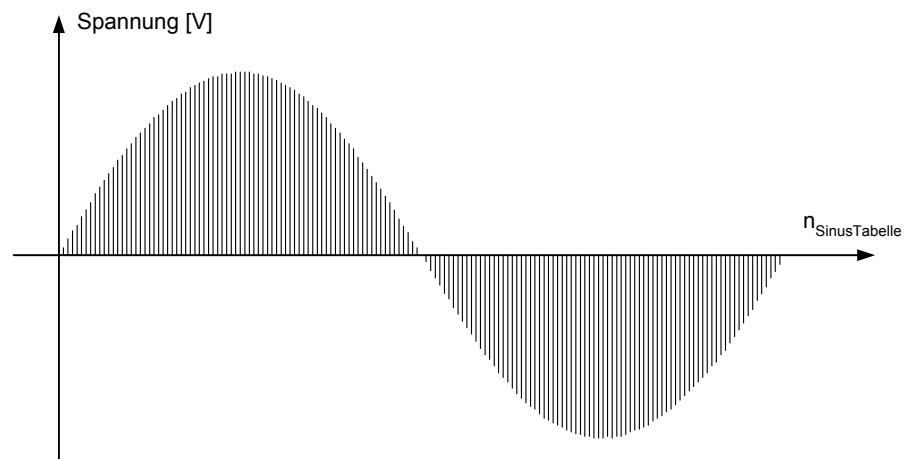
Setzt man Gln. 6-2 in Gln. 6-1 ein, so erhält man für die Periodendauer und die Frequenz der Sinuswelle

$$\text{Gln. 6-3} \quad T_{\text{Sinus}} = 2\pi r_{\text{Rad}} / (n_{\text{Zähne}} * v_{\text{Rad}})$$

und

$$\text{Gln. 6-4} \quad f_{\text{Sinus}} = (n_{\text{Zähne}} * v_{\text{Rad}}) / 2\pi r_{\text{Rad}}$$

Die Sinuswelle eines passiven Drehzahlfühlers wird vom DSP-Prozessor und einem DA-Konverter erzeugt. Sie wird im Datenspeicher des DSP-Prozessors durch  $n_{\text{SinusTabelle}}$  äquidistante Spannungspunkte dargestellt (siehe Abb. 6-3).

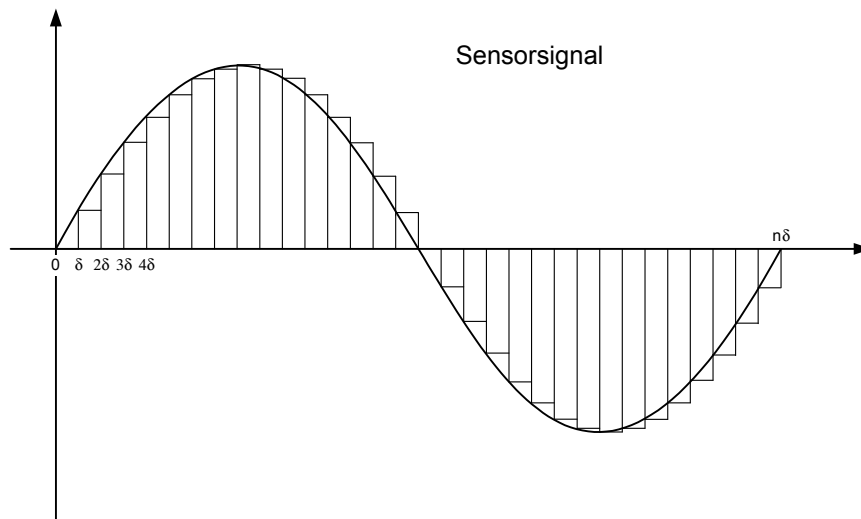


**Abb. 6-3** Darstellung der Sinuswelle im Datenspeicher des DSP-Prozessors

Um eine Sinuswelle mit variabler Frequenz zu erzeugen gibt es prinzipiell zwei Möglichkeiten:

1. Man verwendet eine **variable Austaktperiode**  $\Delta T_{\text{Austakt}}$ , mit der die Stützpunkte der Sinuswelle bei **konstanter Sprungweite**  $\delta$  ausgetaktet werden (siehe Abb. 6-4).  
 $\delta = 1$  bedeutet, dass jeder Stützpunkt ausgetaktet wird;  $\delta = n$  bedeutet, dass jeder n-te Stützpunkt ausgetaktet wird.

2. Man verwendet eine **konstante Austaktperiode**  $\Delta T_{\text{Austakt}}$  und eine **variable Sprungweite**  $\delta$  in der Sinustabelle (siehe Abb. 6-4)..



**Abb. 6-4** Erzeugung einer variablen Sinusfrequenz (siehe Text)

Die DSP-Firmware der ES1331.1 verwendet zur Erzeugung einer variablen Sinusfrequenz die zweite Methode.

An äquidistanten Zeitpunkten ( $\Rightarrow$  konstante Austaktperiode  $\Delta T_{\text{Austakt}}$ ) wird ein Tabellenpunkt der Sinustabelle ausgegeben. Nach der Ausgabe eines Spannungswerts wird die Nummer des Ausgabepunktes in der Sinustabelle um eine variable Sprungweite  $\delta$  (die eine Funktion der Sinusfrequenz ist) erhöht. Die konstante Austaktperiode  $\Delta T_{\text{Austakt}}$  ist ein Vielfaches der Zykluszeit des DSP-Prozessors  $T_{\text{DspZyklus}}$ .

$$\text{Gln. 6-5} \quad \Delta T_{\text{Austakt}} = \lambda * T_{\text{DspZyklus}}$$

Die Zykluszeit des DSP-Prozessors  $T_{\text{DspZyklus}}$  beträgt bei der Simulation eines passiven Drehzahlfühlers  $7 \mu\text{s}$ . Mit  $\lambda$  sei die Proportionalitätskonstante bezeichnet. Sie berechnet sich aus der Größe des Parameters „ShiftCounter“ der DSP-Firmware mit der Beziehung:

$$\text{Gln. 6-6} \quad \lambda = 2^{(\text{ShiftCounter} + 1)} - 1$$

Mit dem Parameter „ShiftCounter“ kann die Austaktperiode des DSP zur Simulation von niedrigen Sinusfrequenzen vergrößert werden.

Zwischen der Sprungweite  $\delta$ , der Anzahl der Sinustabellen-Punkte  $n_{\text{SinusTabelle}}$  und der Anzahl der Sprünge zur Erzeugung einer kompletten Sinuswelle  $n$  besteht folgender Zusammenhang:

$$\text{Gln. 6-7} \quad n_{\text{SinusTabelle}} = n * \delta$$

Für die Periodendauer  $T_{\text{Sinus}}$  der Sinuswelle gilt:

$$\text{Gln. 6-8} \quad T_{\text{Sinus}} = n * \Delta T_{\text{Austakt}}$$

Löst man Gln. 6-8 nach  $\Delta T_{\text{Austakt}}$  auf, eliminiert  $n$  mittels Gln. 6-7 und setzt die dann resultierende Gleichung in Gln. 6-5 ein, so erhält man:

$$\text{Gln. 6-9} \quad \delta = (n_{\text{SinusTabelle}} * \lambda * T_{\text{DspZyklus}}) / T_{\text{Sinus}}$$

Für die Sinusfrequenz  $f_{\text{Sinus}}$  folgt damit:

$$\text{Gln. 6-10} \quad f_{\text{Sinus}} = \delta / (n_{\text{SinusTabelle}} * \lambda * T_{\text{DspZyklus}})$$

mit  $T_{\text{DspZyklus}} = 7 \mu\text{s}$

Mit Gln. 6-3 erhält man aus Gln. 6-9:

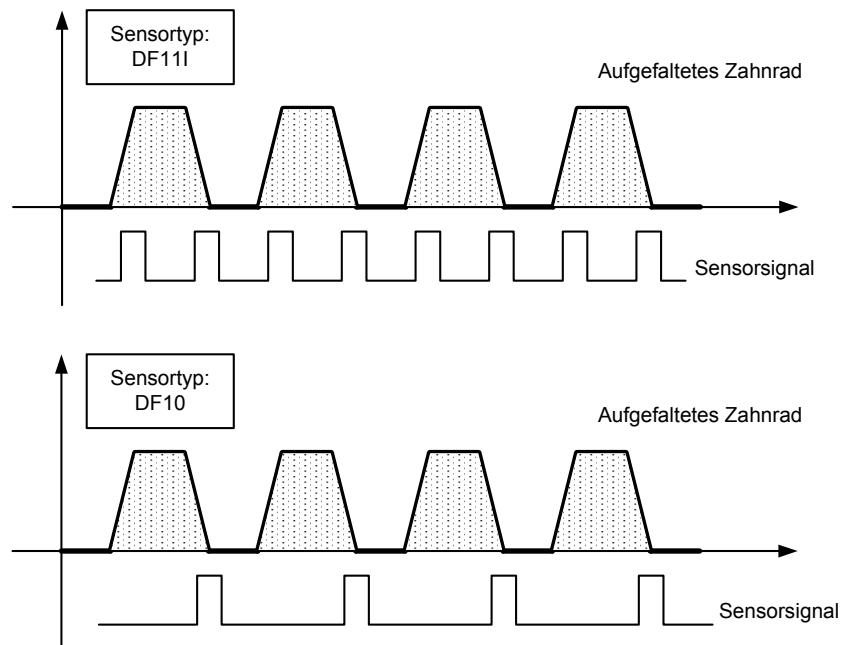
$$\text{Gln. 6-11} \quad \delta = (n_{\text{SinusTabelle}} * \lambda * T_{\text{DspZyklus}} * n_{\text{Zähne}} * v_{\text{Rad}}) / 2\pi r_{\text{Rad}}$$

mit  $T_{\text{DspZyklus}} = 7 \mu\text{s}$

#### Signalausgabe bei aktiven Raddrehzahlfühlern

Der Aufbau eines aktiven Drehzahlfühlers ist gleich dem eines passiven, jedoch wird anstatt eines Analogsignals ein digitales PWM-Signal erzeugt. Die RTIO-Einbindung und die DSP-Firmware der ES1331.1 erlauben die Simulation zweier Typen von aktiven Drehzahlfühlern:

- Bei Drehzahlfühlern vom Typ DF11I wird ein PWM-Impuls bei jeder Detektion einer Kante eines Zahns erzeugt (siehe Abb. 6-5).
- Bei Drehzahlfühlern vom Typ DF10 wird nur dann ein PWM-Impuls erzeugt, wenn die ,abfallende Flanke' eines Zahns detektiert wird (siehe Abb. 6-5).



**Abb. 6-5** Sensorsignal eines DF11I und eines DF10 Drehzahlfühlers

Bezeichnet man mit  $T_{\text{PWM}}$  und  $f_{\text{PWM}}$  Periodendauer und Frequenz des PWM-Signals, so kann man die folgenden Beziehungen formulieren:

$$\text{Gln. 6-12} \quad T_{\text{PWM}} = T_{\text{Rad}} / \Psi * n_{\text{Zähne}}$$

Der Faktor  $\Psi$  ist bei Drehzahlfühlern vom Typ DF11I gleich 2 und bei Drehzahlfühlern vom Typ DF10 gleich 1.

Mit Gln. 6-2 erhält man:

$$\text{Gln. 6-13} \quad T_{\text{PWM}} = 2\pi r_{\text{Rad}} / (\Psi * n_{\text{Zähne}} * v_{\text{Rad}})$$

und

$$\text{Gln. 6-14} \quad f_{\text{PWM}} = (\Psi * n_{\text{Zähne}} * v_{\text{Rad}}) / 2\pi r_{\text{Rad}}$$

Die DSP-Firmware referenziert immer dann den nächsten Zahn eines Zahnrads, wenn ein interner 22-Bit-Zähler überläuft. Der 22-Bit-Zähler wird mit einer Zykluszeit  $T_{\text{DspZyklus}}$  von  $4,5 \mu\text{s}$  getaktet. Für die Periodendauer  $T_{\text{PWM}}$  des PWM-Signals gilt somit die Beziehung:

$$\text{Gln. 6-15} \quad T_{\text{PWM}} = (2^{\kappa} * T_{\text{DspZyklus}}) / (\delta * \Psi)$$

und

$$\text{Gln. 6-16} \quad f_{\text{PWM}} = (\delta * \Psi) / (2^{\kappa} * T_{\text{DspZyklus}})$$

$\kappa$  bezeichnet dabei die Bitlänge des Zählers ( $\kappa = 22$ ) und  $\delta$  die Schrittweite des Zählers.

Aus Gln. 6-15 erhält man für  $\delta$ :

$$\text{Gln. 6-17} \quad \delta = (2^{\kappa} * T_{\text{DspZyklus}}) / (T_{\text{PWM}} * \Psi)$$

und

$$\text{Gln. 6-18} \quad \delta = (2^{\kappa} * f_{\text{PWM}} * T_{\text{DspZyklus}}) / \Psi$$

wobei  $T_{\text{DspZyklus}} = 4,5 \mu\text{s}$ ,  $\kappa = 22$ ,  $\Psi = 2$  (DF11l) und  $\Psi = 1$  (DF10).

Mit Gln. 6-13 erhält man

$$\text{Gln. 6-19} \quad \delta = (2^{\kappa} * n_{\text{Zähne}} * v_{\text{Rad}} * T_{\text{DspZyklus}}) / 2\pi r_{\text{Rad}}$$

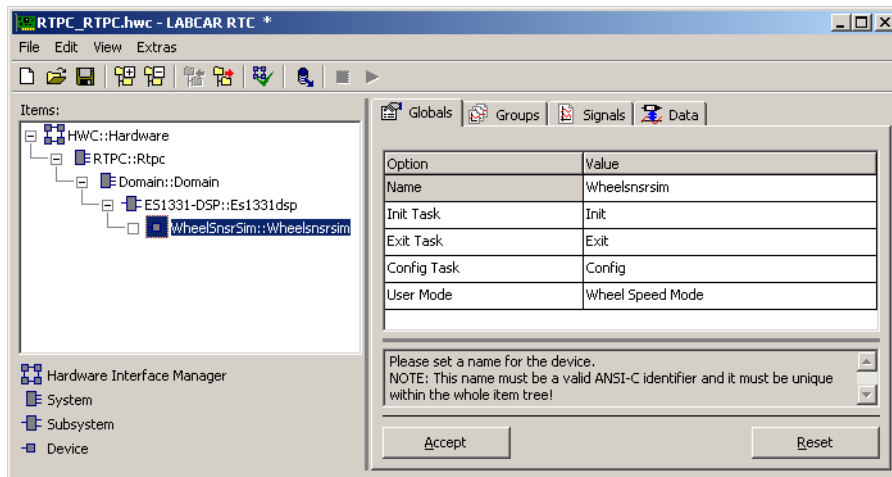
wobei  $T_{\text{DspZyklus}} = 4,5 \mu\text{s}$  und  $\kappa = 22$ .

### 6.2.2 Globals (WheelSnsrSim Device)

Die Frequenz des Sinussignals eines passiven Drehzahlfühlers oder die Frequenz des PWM-Signals eines aktiven Drehzahlfühlers kann vom Anwender auf drei unterschiedliche Arten vorgegeben werden. Die Festlegung, auf welche Art die Signalfrequenz vorgegeben wird erfolgt im Auswahlfeld „User Mode“ der Registerkarte „Globals“ eines WheelSnsrSim Device (siehe Abb. 6-6).

- Wird die Option „Wheel Speed Mode“ gewählt, so erfolgt die Festlegung der Signalfrequenz durch Vorgabe einer Radgeschwindigkeit. Aus dieser Radgeschwindigkeit wird nach Gln. 6-4 auf Seite 127 die Sinusfrequenz eines passiven Drehzahlfühlers und nach Gln. 6-14 auf Seite 129 die PWM-Frequenz eines aktiven Drehzahlfühlers bestimmt.
- Bei Wahl der Option „Frequency Mode“ erfolgt die Festlegung der Signalfrequenz direkt durch Vorgabe der Frequenz (in Hz).

- Im „Expert Mode“ wird zur Festlegung der Signalfrequenz der  $\delta$ -Wert vorgegeben. Nach Gln. 6-10 auf Seite 128 ist damit die Sinusfrequenz passiver Drehzahlfühler und nach Gln. 6-16 auf Seite 130 die PWM-Frequenz aktiver Drehzahlfühler definiert.



**Abb. 6-6** Die Registerkarte „Globals“ eines WheelSnrSim Device

In Tab. 6-1 sind die Felder der Registerkarte „Globals“ des WheelSnrSim Device aufgelistet, die zur Konfiguration des Echtzeitbetriebsystems dienen.

Feld	Bedeutung	Bemerkung
Init Task	Zuordnung des Treiber-Initialisierungsprozesses zu einer ERCOS <sup>EK</sup> -Task.	ERCOS <sup>EK</sup> -Task: Type: „Init“ Application Mode: „Active“
Exit Task	Zuordnung des Exit-Prozesses des Treibers zu einer ERCOS <sup>EK</sup> -Task.	ERCOS <sup>EK</sup> -Task: Type: „Init“ Application Mode: „Inactive“
Config Task	Zuordnung des Treiber Konfigurationsprozesses zu einer ERCOS <sup>EK</sup> -Task.	ERCOS <sup>EK</sup> -Task: Type: „Alarm“ Application Mode: „Active“ Period: 100 ms – 1000 ms

**Tab. 6-1** Die Felder der Registerkarte „Globals“ des WheelSnrSim Item zur Konfiguration des Echtzeitbetriebsystems

### 6.2.3 Groups (WheelSnrSim Device)

Ein WheelSnrSim Device besitzt drei Signalgruppen (siehe Abb. 6-7 auf Seite 132).

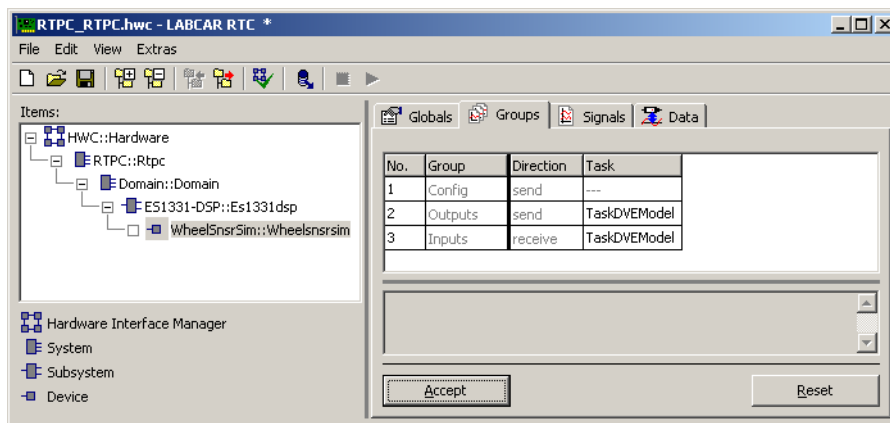
- Die Signalgruppe *Config* wird vom Simulationstarget zur ES1331.1 übertragen. Die RTIO-Signale der Signalgruppe sind Konfigurationsparameter der DSP-Firmware, d.h., diese Signale werden von der DSP-Firmware nur während der Initialisierungsphase verarbeitet. Aus diesem Grund hat eine Änderung eines RTIO-Signals der Signalgruppe *Config* eine Initialisierung der DSP-Firmware zur Folge. Diese Initialisierung kann in Abhängigkeit vom zu simulierenden Drehzahlfühlertyp und den gewählten Konfigurationsparametern bis zu mehreren Sekunden dauern.

Der Signalgruppe *Config* kann in der Registerkarte „Groups“ keine Task des Echtzeitbetriebsystems zugeordnet werden. Die Berechnung der *Config* Signalgruppe erfolgt implizit im „Init“ und dem „Config“ Prozess des RTIO-Treibers.

### **Hinweis**

Die Zuordnung der „Init“ und „Config“ Prozesse zu Tasks des Echtzeitbetriebsystems erfolgt in den Feldern „Init Task“ und „Config Task“ der Registerkarte „Globals“ des WheelSnrSim Device (vgl. dazu die Erläuterungen in Tab. 6-1 auf Seite 131).

- Die Signalgruppe *Outputs* wird vom Simulationstarget zur ES1331.1 übertragen. Diese Signalgruppe enthält alle Signale zur Steuerung der Drehzahlfühlersimulation in Echtzeit.
- Die Signalgruppe *Inputs* wird von der ES1331.1 zum Simulationstarget übertragen. Diese Signalgruppe enthält Statusinformationen der DSP-Firmware.



**Abb. 6-7** Die Registerkarte „Groups“ des WheelSnrSim Device

#### 6.2.4 Signals (WheelSnrSim Device)

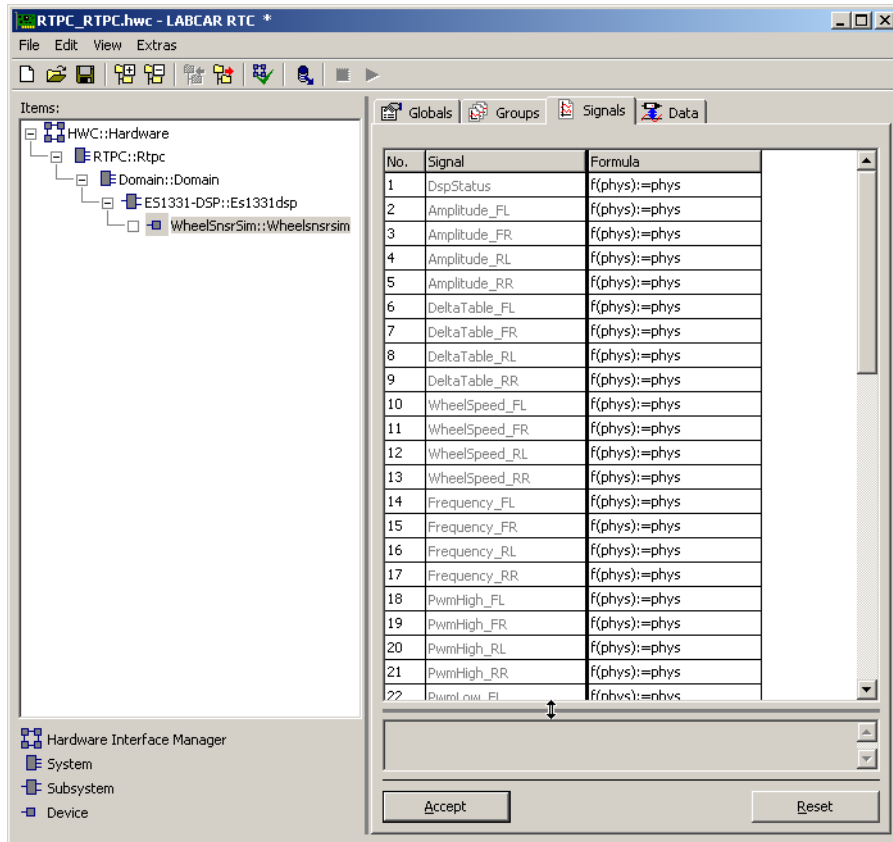
Abb. 6-8 zeigt die Registerkarte „Signals“ eines WheelSnrSim Device.

Die Endungen *„\_FL“*, *„\_FR“*, *„\_RL“*, *„\_RR“* in den Signalnamen bezeichnen das Rad. *„\_FL“* steht für das vordere, linke Rad, *„\_FR“* für das vordere, rechte Rad, *„\_RL“* für das hintere, linke Rad und *„\_RR“* für das hintere, rechte Rad.

### **Hinweis**

Der RTIO-Treiber für die ES1331.1 überprüft nicht, ob die einzelnen RTIO-Signale zulässige und sinnvolle Werte aufweisen. Dies sicherzustellen obliegt dem Anwender der ES1331.1 und deren RTIO-Einbindung.





**Abb. 6-8** Die Registerkarte „Signals“ des WheelSnrSim Device  
Die RTIO-Signale der Signalgruppe Inputs

Das RTIO-Signal *DspStatus* zeigt den Status der DSP-Firmware an. In Tab. 6-2 sind die Eigenschaften der RTIO-Signale der *Inputs* Signalgruppe zusammengefasst.

RTIO-Signal	Datentyp	Signalgruppe	Beschreibung
DspStatus	uint16	Inputs	Status der ES1331.1 DSP-Firmware 1: Initialized 2: Running 4: Active Running 8: Active Initialized

**Tab. 6-2** Die RTIO-Signale der Signalgruppe Inputs

**Hinweis**

Wenn in Tab. 6-2, Tab. 6-3 und Tab. 6-4 Datentypen angegeben werden, so sind dies die Datentypen, die der RTIO-Treiber intern verwendet.

*Die RTIO-Signale der Signalgruppe Outputs*

Die RTIO-Signale *Amplitude\_XX* sind nur bei der Simulation eines passiven Drehzahlfühlers von Bedeutung und geben die Amplitude des Sinussignals bzw. Drehzahlfühlersignals in Volt an.

Die RTIO-Signale *DeltaTable\_XX* geben bei der Simulation eines passiven Drehzahlfühlers die Sprungweite in der Sinustabelle nach Gln. 6-11 auf Seite 129 an. Bei der Simulation eines aktiven Drehzahlfühlers geben diese Signale die Schrittweite  $\delta$  des internen Zählers nach Gln. 6-19 auf Seite 130 an. Diese RTIO-Signale sind nur dann von Bedeutung, wenn in der Registerkarte „Globals“ im Feld „User Mode“ die Einstellung „Expert Mode“ gewählt wird.

Die RTIO-Signale *WheelSpeed\_XX* geben die Radgeschwindigkeit in km/h an. Falls in der Registerkarte „Globals“ im Feld „User Mode“ die Einstellung „Wheel Speed Mode“ gewählt wird, werden mit diesen RTIO-Signalen die Sinusfrequenzen der passiven Drehzahlfühler und die PWM-Frequenzen der aktiven Drehzahlfühler festgelegt. Wird im Feld „User Mode“ eine andere Einstellung gewählt, so haben diese RTIO-Signale keine Bedeutung.

Mit den RTIO-Signalen *Frequency\_XX* werden die Signalfrequenzen der aktiven und passiven Drehzahlfühler vorgegeben, falls im Feld „User Mode“ der Registerkarte „Globals“ die Einstellung „Frequency Mode“ gewählt wird. Wird im Feld „User Mode“ eine andere Einstellung gewählt, so haben diese RTIO-Signale keine Bedeutung.

Die RTIO-Signale *PwmHigh\_XX* sind nur bei der Simulation eines aktiven Drehzahlfühlers von Bedeutung. Diese Signale definieren die Höchstdauer der High-Phase der PWM-Signale. Für diese Dauer  $T_{PwmHighMax}$  gilt:

$$\text{Gln. 6-20 } T_{PwmHighMax} = ( PwmHigh\_XX + 1 ) * T_{DspZyklus}$$

Die Zykluszeit  $T_{DspZyklus}$  beträgt 4,5  $\mu$ s.

Die RTIO-Signale *PwmLow\_XX* sind nur bei der Simulation eines aktiven Drehzahlfühlers von Bedeutung. Diese Signale definieren die Mindestdauer der Low-Phase der PWM-Signale. Für diese Dauer  $T_{PwmLowMin}$  gilt:

$$\text{Gln. 6-21 } T_{PwmLowMin} = ( PwmLow\_XX + 1 ) * T_{DspZyklus}$$

Die Zykluszeit  $T_{DspZyklus}$  beträgt 4,5  $\mu$ s.

#### Hinweis

*Durch die Vorgabe der Signalfrequenz, High-Phase und Low-Phase wäre ein PWM-Signal überbestimmt. Insofern kann vom Anwender nur die Frequenz sowie die Höchstdauer  $T_{PwmHighMax}$  der High-Phase und die Mindestdauer  $T_{PwmLowMin}$  der Low-Phase des PWM-Signals vorgegeben werden. Die Frequenzvorgabe wird generell eingehalten und falls die Mindestdauer  $T_{PwmLowMin}$  der Low-Phase nicht verletzt wird, ist die Dauer der High-Phase gleich  $T_{PwmHighMax}$ .*

Die RTIO-Signale *NoMove\_XX* legen fest, ob sich das jeweilige Rad drehen soll oder still stehen soll.

Die RTIO-Signale *Info\_XX*, *ErrorPulse\_XX* und *Mask\_XX* sind für zukünftige Erweiterungen gedacht und haben in der aktuellen Version des RTIO-Treibers keine Funktion.

Die RTIO-Signale *WheelRadius\_XX* geben den Radradius in Meter an. Diese RTIO-Signale sind nur dann von Bedeutung, wenn die Frequenzen der Drehzahlfühlersignale durch Vorgabe einer Radgeschwindigkeit festgelegt werden, wenn also im Feld „User Mode“ der Registerkarte „Globals“ die Einstellung „Wheel Speed Mode“ gewählt wird.

Die Eigenschaften der RTIO-Signale der *Outputs* Signalgruppe sind in Tab. 6-3 zusammengefasst.

RTIO-Signal	Datentyp	Signalgruppe	Beschreibung
Amplitude_XX	real32	Outputs	Amplituden der Sinussignale bzw. Drehzahlfühlersignale in V. Nur von Bedeutung bei der Simulation eines passiven Drehzahlfühlers. Zulässiger Wertebereich: 0.0 V ... 10.0 V
DeltaTable_XX	uint16	Outputs	Passiver Drehzahlfühler: Sprungweite $\delta$ in der Sinustabelle nach Gln. 6-11 auf Seite 129. Aktiver Drehzahlfühler: Schrittweite $\delta$ der internen Zähler nach Gln. 6-19 auf Seite 130. Nur im User Mode „Expert Mode“ von Bedeutung.
WheelSpeed_XX	real32	Outputs	Radgeschwindigkeit in km/h. Nur im User Mode „Wheel Speed Mode“ von Bedeutung.
Frequency_XX	real32	Outputs	Frequenz des Drehzahlfühlersignals in Hz. Nur im User Mode „Frequency Mode“ von Bedeutung.
PwmHigh_XX	uint16	Outputs	Höchstdauer der High-Phase der PWM-Signale. Nur von Bedeutung bei der Simulation eines aktiven Drehzahlfühlers. Zulässiger Wertebereich: 0 ... 2047
PwmLow_XX	uint16	Outputs	Mindestdauer der Low-Phase der PWM-Signale. Nur von Bedeutung bei der Simulation eines aktiven Drehzahlfühlers. Zulässiger Wertebereich: 0 ... 31
NoMove_XX	uint16	Outputs	Stillstandsanzeige Nur von Bedeutung bei der Simulation eines aktiven Drehzahlfühlers. Empfohlener Wertebereich: 0: Kein Stillstand 1: Stillstand

RTIO-Signal	Datentyp	Signalgruppe	Beschreibung
Info_XX	uint16	Outputs	Für zukünftige Erweiterungen
ErrorPulse_XX	uint16	Outputs	Für zukünftige Erweiterungen
Mask_XX	uint16	Outputs	Für zukünftige Erweiterungen
WheelRadius_XX	real32	Outputs	Radradius in m. Nur im User Mode „Wheel Speed Mode“ von Bedeutung.

**Tab. 6-3** Die RTIO-Signale der Signalgruppe *Outputs*

*Die RTIO-Signale der Signalgruppe Config*

Bei allen RTIO-Signalen der Signalgruppe *Config* handelt es sich um Konfigurationsparameter der DSP-Firmware. Eine Änderung eines Konfigurationsparameters hat eine Initialisierung der DSP-Firmware zur Folge.

Mit dem *DspMode* RTIO-Signal wird die Art des zu simulierenden Drehzahlfühlers festgelegt:

- Der Wert 3 steht für die Simulation eines passiven Drehzahlfühlers.
- Mit dem Wert 1 wird ein aktiver Drehzahlfühler vom Typ DF111 simuliert.
- Mit dem Wert 2 wird ein aktiver Drehzahlfühler vom Typ DF10 simuliert.

Das RTIO-Signal *TabCount* ist nur bei der Simulation eines passiven Drehzahlfühlers von Bedeutung. Mit diesem Signal wird die Anzahl der Stützpunkte der Sinustabelle festgelegt (Parameter  $n_{\text{SinusTabelle}}$  in Gln. 6-7 auf Seite 128 und Gln. 6-9 bis Gln. 6-11 auf Seite 129). Ein empfohlener Richtwert ist  $\text{TabCount} = 4096$ .

Das RTIO-Signal *ShiftCounter* ist ebenfalls nur bei der Simulation eines passiven Drehzahlfühlers von Bedeutung. Mit diesem Signal wird der Zahlenwert für den internen DSP-Parameter „ShiftCounter“ (siehe Gln. 6-6 auf Seite 128) festgelegt. Ein empfohlener Richtwert ist  $\text{ShiftCounter} = 8$ .

Die RTIO-Signale *TeethCount\_XX* geben die Zähnezahl der Drehzahlfühler-Zahnäder an. Ein empfohlener Richtwert ist  $\text{TeethCount\_XX} = 48$ .

Die RTIO-Signale *DAC0* und *DAC1* sind nur bei der Simulation eines aktiven Drehzahlfühlers von Bedeutung. Mit diesen RTIO-Signalen werden die Ausgangsspannungen der beiden auf die Frontplatten-Buchse geführten DA-Wandler-Ausgänge festgelegt.

In Tab. 6-4 sind die Eigenschaften der RTIO-Signale der „Config“ Signalgruppe zusammengefasst.

RTIO-Signal	Daten- typ	Signal- gruppe	Beschreibung
DspMode	uint16	Config	Art des zu simulierenden Drehzahlfühlers. Zulässige Werte: 1: Aktiver Drehzahlfühler vom Typ DF111 2: Aktiver Drehzahlfühler vom Typ DF10 3: Passiver Drehzahlfühler
TabCount	uint16	Config	Anzahl Stützpunkte der Sinustabelle. Parameter $n_{\text{SinusTabelle}}$ in Gln. 6-7 auf Seite 128. Nur von Bedeutung bei der Simulation eines passiven Drehzahlfühlers. Empfohlener Wertebereich: 32 ... 16384. Ein empfohlener Richtwert ist $\text{TabCount} = 4096$
ShiftCounter	uint16	Config	Parameter ShiftCounter in Gln. 6-6. Nur von Bedeutung bei der Simulation eines passiven Drehzahlfühlers. Empfohlener Wertebereich: 1 ... 15. Ein empfohlener Richtwert ist $\text{ShiftCounter} = 8$
TeethCount_XX	uint16	Config	Zähnezahl des Drehzahlfühler-Zahnrad. Parameter $n_{\text{Zähne}}$ in Gln. 6-1 auf Seite 127 und Gln. 6-12 auf Seite 129. Empfohlener Wertebereich: 1 ... 128. Ein empfohlener Richtwert ist $\text{TeethCount\_XX} = 48$
DAC0, DAC1	real32	Config	DA-Wandler Ausgangsspannungen in V. Nur von Bedeutung bei der Simulation eines aktiven Drehzahlfühlers. Zulässiger Wertebereich: -10.0 V ... 10.0 V

**Tab. 6-4** Die RTIO-Signale der Signalgruppe „Config“



## 7 ES1332.1 Arbitrary Signal Generator Board

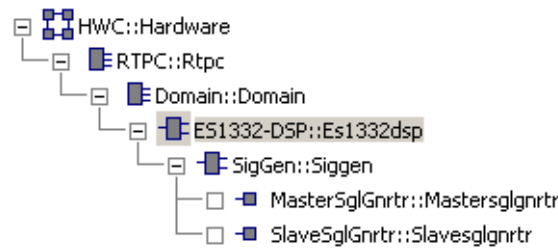
Das ES1332.1 Arbitrary Signal Generator Board ist eine VMEbus-Karte zur Erzeugung beliebig geformter Signalverläufe. Sie besitzt 2 Signalgeneratoren mit jeweils 2 Ausgabekanälen. Die Signalgeneratoren können nur synchron zueinander betrieben werden.

Die RTIO-Einbindung der ES1332.1 ist auf die beabsichtigte Anwendung im Automotive-Umfeld nämlich die Erzeugung drehzahlsynchroner Mess- und Steuersignale zugeschnitten. Beispiele sind:

- Drehzahlsignal - wird vom Motorsteuergerät zur Erfassung der Drehzahl und des Kurbelwellenwinkels eingesetzt -
- Nockenwellensignal - wird vom Steuergerät zur Zylindererkennung benötigt

### Aufbau des ES1332.1 RTIO Item-Baums

In RTIO wird die ES1332.1 durch Auswahl des ES1332-DSP Subsystems eingebunden.



**Abb. 7-1** ES1332 RTIO Item-Baum

Auf der Ebene des ES1332-DSP Subsystems kann die gewünschte Anwendung der ES1332.1 ausgewählt bzw. festgelegt werden. Bis dato ist hier nur das SigGen Subsystem zur arbiträren Signalgenerierung wählbar (siehe Abb. 7-1). Falls in Zukunft weitere Anwendungen der ES1332.1 implementiert werden, so werden die zugehörigen RTIO Items auf dieser Ebene eingebaut werden.

Auf der Ebene des SigGen Subsystems kann ein MasterSglGnrtr und ein SlaveSglGnrtr Device eingefügt werden. Diese Devices entsprechen den beiden physikalisch vorhandenen Signalgeneratoren.

#### **Hinweis**

*Ein SlaveSglGnrtr Device kann nicht ohne MasterSglGnrtr Device existieren, daher muss das Master Item zuerst eingefügt werden (wird durch die Software sichergestellt).*

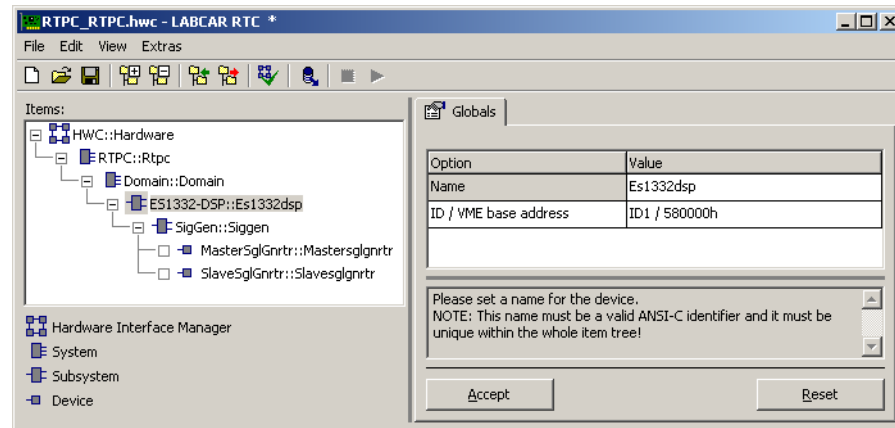
Die Unterscheidung zwischen Master und Slave Generator ist erforderlich aufgrund der synchronen Betriebsweise der beiden Signalgeneratoren, die dazu führt, dass es für beide Signalgeneratoren eine gemeinsame Motordrehzahl bzw. Signalfrequenz gibt. Nur ein Signalgenerator (der Master-Generator) kann die Motordrehzahl bzw. die Signalfrequenz vorgeben, die dann auch für den Slave-Generator gilt.

Eine weitere Echtzeitgröße, die beiden Generatoren gemeinsam ist, ist der aktuelle Kurbelwellenwinkel, der von der ES1332.1 zum Simulationstarget vom Master-Generator-Item, jedoch nicht vom Slave-Generator-Item übertragen wird.

## 7.1 ES1332-DSP Subsystem

### 7.1.1 Globals (ES1332-DSP Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES1332-DSP Subsystems beschrieben.



**Abb. 7-2** Die Registerkarte „Globals“ des ES1332-DSP Subsystems  
*ID / VME base address*

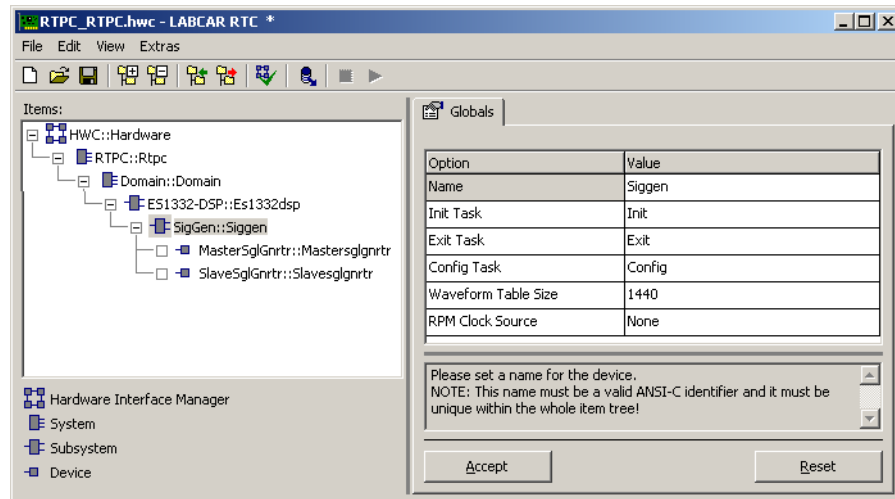
Im Auswahlfeld „ID / VME base address“ wird die VMEbus-Basisadresse der ES1332.1 angegeben. Es stehen 2 Basisadressen zur Auswahl (ID1/580000h und ID2/5C0000h).



## 7.2 SigGen Subsystem

### 7.2.1 Globals (SigGen Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des SigGen Subsystems beschrieben.



**Abb. 7-3** Die Registerkarte „Globals“ des SigGen Subsystems

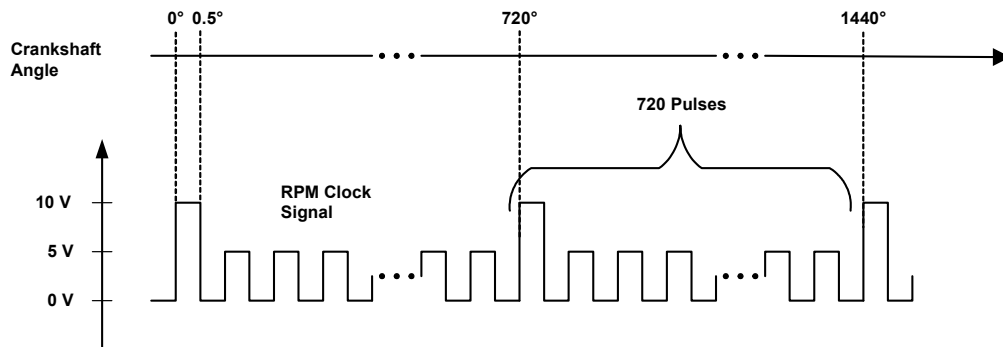
Option	Bedeutung	Wertebereich/ Bemerkung
Init Task	Zuordnung des Treiberinitialisierungsprozesses zu einer ERCON <sup>EK</sup> -Task	ERCON <sup>EK</sup> -Task: Type: 'Init' Application Mode: 'Active'
Exit Task	Zuordnung des Exit-Prozesses des Treibers zu einer ERCON <sup>EK</sup> -Task	ERCON <sup>EK</sup> -Task: Type: 'Init' Application Mode: 'Inactive'
Config Task	Zuordnung des Treiberkonfigurationsprozesses zu einer ERCON <sup>EK</sup> -Task	ERCON <sup>EK</sup> -Task: Type: 'Alarm' Application Mode: 'Active' Period: 100 ms - 500 ms
RPM Clock Source	Quelle des RPM-Taktsignals	- None - Master Generator Channel A - Master Generator Channel B - Slave Generator Channel A - Slave Generator Channel B
Waveform Table Size	Signalbankgröße in Punkten	2 - 1440

**Tab. 7-1** Felder der Registerkarte „Globals“ des SigGen Subsystems

### RPM Clock Source

Mit dieser Option wird festgelegt, welcher der vier Kanäle ein RPM-Taktsignal ausgibt. Das RPM-Taktsignal wird zur Synchronisation der Hardware bezüglich der momentanen Motordrehzahl verwendet.

Ein 10 V-Puls signalisiert einen Nulldurchgang der Nockenwelle, alle anderen Taktpulse besitzen eine Amplitude von 5 V. Alle 0,5° Kurbelwellenwinkel wird eine steigende oder fallende Flanke ausgegeben.



**Abb. 7-4** RPM-Taktsignal

#### Hinweis

Wenn ein Kanal zur Ausgabe des RPM-Taktsignals konfiguriert wird, gelten die folgenden Einschränkungen:

1. Die Option „Waveform Table Size“ wird auf den Wert 1440 gesetzt.
2. Die Signalbank 4 des Signalgenerators, der den RPM-Takt ausgibt, wird für dieses Taktsignal verwendet und ist damit reserviert.
3. Die RTIO-Signale *ChnX\_SigSel*, *ChnX\_Mode* und *ChnX\_PhaseShift* (siehe Tab. 7-3 auf Seite 147) sind bei dem Kanal, der den RPM-Takt ausgibt, nicht verfügbar.

#### Waveform Table Size

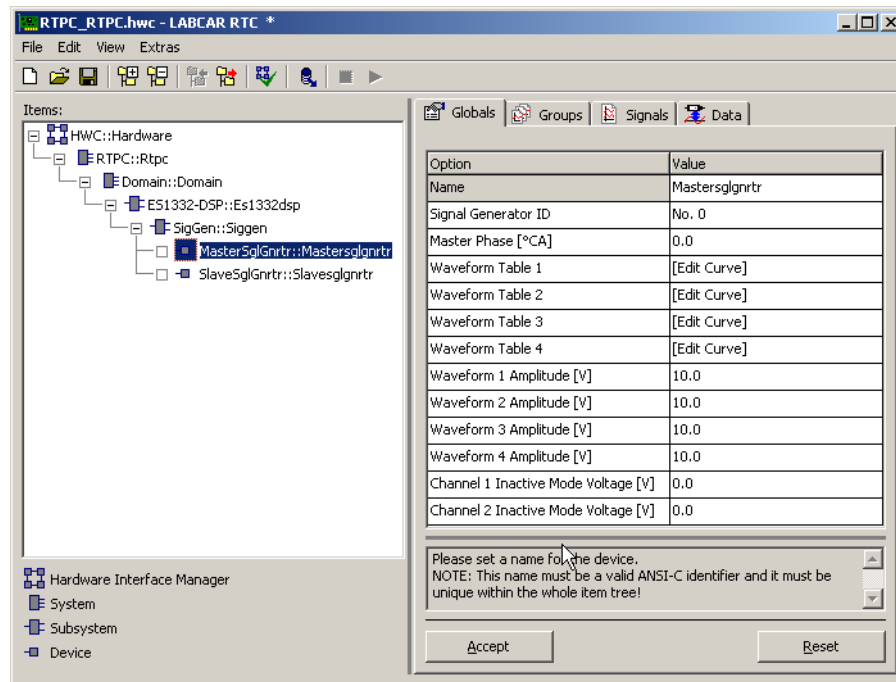
Im Feld „Waveform Table Size“ wird die Signalbankgröße  $n_{\text{Bank}}$  festgelegt, d.h. die Anzahl der Stützpunkte der Signalverläufe.

Die Signalbankgröße kann zwischen 2 und 1440 gewählt werden, sie muss jedoch immer größer oder gleich der Anzahl der Signalform-Stützpunkte sein. Die obere Grenze von 1440 Stützpunkten ist aus der Anwendung der ES1332.1 im Automotive-Umfeld (Motorsteuergeräte-Test) abgeleitet. Bei dieser Anwendung werden in den Signalbanken die Kurvenverläufe bei 2 Motorumdrehungen, d.h. bei 2 Umdrehungen der Kurbelwelle abgelegt (720° Kurbelwelle). Mit 1440 Stützpunkten lassen sich diese Signale mit einer Winkelauflösung von 0,5° beschreiben.

## 7.3 MasterSglGnrtr Device

### 7.3.1 Globals (MasterSglGnrtr Device)

In diesem Abschnitt werden die globalen Einstellungen des MasterSglGnrtr Device beschrieben. Auf das SlaveSglGnrtr Device wird nur dort eingegangen, wo Unterschiede zum MasterSglGnrtr Device bestehen.



**Abb. 7-5** Die Registerkarte „Globals“ des MasterSglGnrtr Device

Option	Bedeutung	Wertebereich/ Bemerkung
Signal Generator ID	Zuordnung eines physikalischen Signalgenerators zum RTIO Item	Generator 1 oder Generator 2
Master Phase [°CA]	Für beide Ausgabekanäle geltende Phasenverschiebung der Signale in Grad Kurbelwellenwinkel	-720,0 ... 720,0
Waveform Table x (x = 1, 2, 3, 4)	Tabelleneditor zur Definition der Signalform	
Waveform x Amplitude [V] (x = 1, 2, 3, 4)	Amplitude des Signalverlaufs einer Signalbank in V	0,0 ... 10,0
Channel x Inactive Mode Voltage [V] (x = 1, 2)	Ausgabespannung eines Kanals im inaktiven Modus in V	-10,0 ... 10,0

**Tab. 7-2** Felder der Registerkarte „Globals“ der MasterSglGnrtr/SlaveSglGnrtr Device

### Signal Generator ID

Im Feld „Signal Generator ID“ wird der physikalische Signalgenerator (Generator 1 oder Generator 2) dem RTIO Item zugeordnet.

### Master Phase [°CA]

Im Feld „Master Phase [°CA]“ kann eine Phasenverschiebung, die für beide Ausgabekanäle gilt, eingestellt werden. Wiederum in Anlehnung an die beabsichtigte Anwendung im Automotive-Umfeld zum Motorsteuergeräte-Test wird diese Phasenverschiebung in Grad Kurbelwellenwinkel spezifiziert. Bei einem Winkel von  $\pm 360^\circ$  wird der Signalverlauf um die halbe Signalbankgröße verschoben, bei einem Winkel von  $\pm 720^\circ$  wird der Signalverlauf um die gesamte Signalbankgröße verschoben, d.h. es wird dann effektiv nicht verschoben.

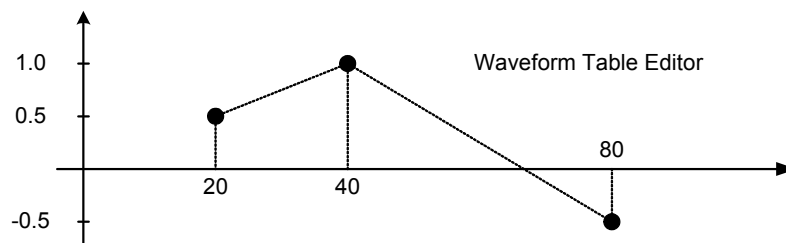
### Waveform Table x

In den Feldern „Waveform Table x“ ( $x = 1, 2, 3, 4$ ) können die Editoren zur Definition der Signalformen der einem Generator zugeordneten Signalverläufe geöffnet werden.

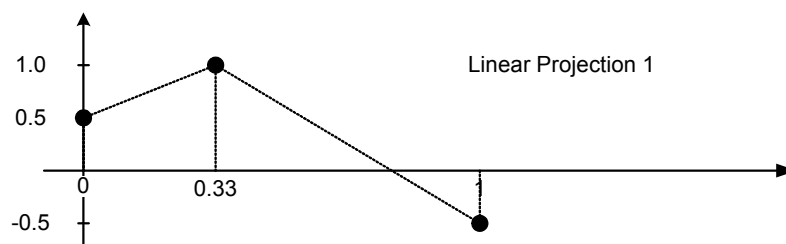
Es gibt keine Beschränkung für den Wertebereich der einzugebenden x-Werte. Die Anzahl der Stützstellen muss im Bereich von 2 bis 1024 liegen. Der Wertebereich für die z-Werte ist -1.0 bis 1.0.

Die Abbildung des in dieser Tabelle definierten Signalverlaufs auf die interne Signaltabelle ist im Folgenden beschrieben.

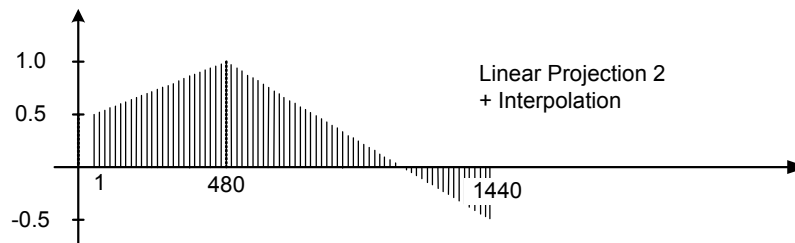
Die in die Tabelle eingegebenen Werte zur Festlegung des Signalverlaufs sind z.B. wie folgt:



1. In einem ersten Schritt wird der Bereich der eingegebenen x-Werte (hier: 20 - 80) durch eine lineare Transformation auf das Intervall  $[0, 1]$  abgebildet.



2. Der zweite Schritte besteht in einer Abbildung dieses Intervalls auf die Bankgröße, die durch den Parameter „Waveform Tabel Size“ (hier: 1440) festgelegt wird und einer linearen Interpolation aller in diesem Intervall vorhandenen Punkte.



#### Waveform x Amplitude [V]

In den Feldern „Waveform x Amplitude [V]“ (x = 1, 2, 3, 4) werden die Amplituden der Signalverläufe festgelegt.

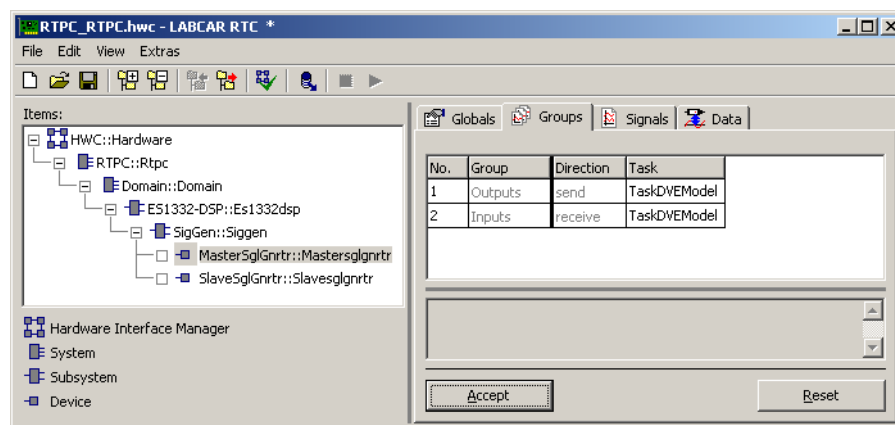
#### Channel x Inactive Mode Voltage [V]

Die Ausgabekanäle eines Generators können in 2 verschiedenen Modi betrieben werden. Im sogenannten *aktiven Modus* wird der gerade aktive Signalverlauf am Kanalausgang ausgegeben. Im *inaktiven Modus* wird eine Gleichspannung ausgegeben. Der Wert dieser Gleichspannung wird in den Eingabefeldern „Channel x Inactive Mode Voltage [V]“ (x = 1, 2) für beide Kanäle eines Generators getrennt definiert. Eine Modusumschaltung zur Laufzeit ist in Echtzeit möglich.

Bis auf die „Signal Generator ID“ Option können sämtliche Optionen der Registerkarte „Globals“ zur Laufzeit umkonfiguriert werden.

### 7.3.2 Groups (MasterSglGnrtr Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des MasterSglGnrtr Device beschrieben.



**Abb. 7-6** Die Registerkarte „Groups“ des MasterSglGnrtr Device

Das MasterSglGnrtr Device bietet zwei Signalgruppen:

#### Outputs

Die Signale der Signalgruppe „Outputs“ werden vom Simulationstarget zur ES1332.1 übertragen.

### Inputs

Die Signale der Signalgruppe „Inputs“ werden von der ES1332.1 zum Simulationsstart übertragen. Eine detaillierte Beschreibung der Signale finden Sie im Abschnitt 7.3.3 auf Seite 146.

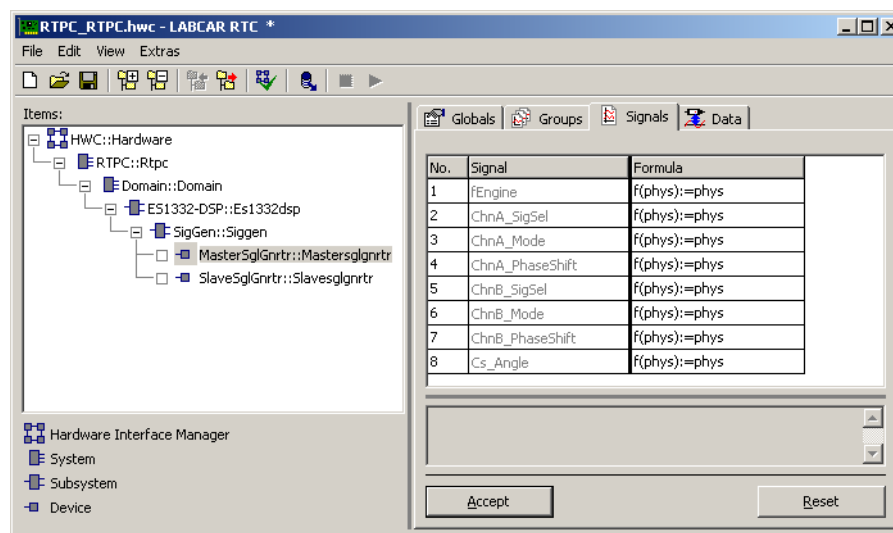
In der Registerkarte „Groups“ (siehe Abb. 7-6) werden die Prozesse, in denen diese Signalgruppen übertragen werden, Tasks des Echtzeitbetriebssystems zugeordnet. Die Tasks sollten den Typ „Alarm“ und den Application Mode „Active“ haben.

#### Hinweis

*Das SlaveSglGnrtr Device weist keine „Inputs“ Signalgruppe auf, da der aktuelle Kurbelwellenwinkel, der in der „Inputs“ Signalgruppe übertragen wird, nur vom Master-Generator übermittelt wird.*

### 7.3.3 Signals (MasterSglGnrtr Device)

Die Signale sind fest vorgegeben. Die Registerkarte „Signals“ listet die einzelnen Signale auf.



**Abb. 7-7** Die Registerkarte „Signals“ des MasterSglGnrtr Device

In Tab. 7-3 sind die Eigenschaften der Signale zusammengefasst.

Signal	Bedeutung	Datentyp	
fEngine	Motordrehzahl in U/min	real32	4 / WaveformTableSize ... 10,8×10 <sup>6</sup> / WaveformTableSize Auflösung: 2 U/min
ChnX_SigSel X = A, B	Auswahl der (aktiven) Signalbank	uint8	0: Signalbank 1 1: Signalbank 2 2: Signalbank 3 3: Signalbank 4
ChnX_Mode X = A, B	Modusumschaltung	uint8	0: Aktiver Modus 1: Inaktiver Modus
ChnX_PhaseShift X = A, B	Phasenverschiebung in Grad Kurbelwellenwinkel	real32	-720.0 ... 720.0
Cs_Angle	Aktueller Kurbelwellenwinkel in Grad	real32	0 ... 720.0

**Tab. 7-3** Die Signale des MasterSglGnrtr Device

#### Hinweis

Wenn in diesem Dokument Datentypen angegeben werden, so sind das die Datentypen, die der RTIO-Treiber intern verwendet.

#### Hinweis

Ein SlaveSglGnrtr Device weist die gleichen Signale auf wie ein MasterSglGnrtr Device, es fehlen jedoch das Signal fEngine zur Steuerung der Motordrehzahl und das Signal Cs\_Angle zum Lesen des Kurbelwellenwinkels.

#### fEngine

Bei der Motorsteuergeräte-Simulation wird die Ausgabefrequenz der Signalverläufe durch die Motordrehzahl definiert. Außerdem werden in den Signalbanken die Signalverläufe zweier aufeinanderfolgender Motorumdrehungen abgelegt. Daher besteht zwischen der Motordrehzahl bzw. Motorfrequenz  $f_{\text{Engine}}$  und der Signalfrequenz  $f_S$  folgender Zusammenhang:

$$f_{\text{Engine}} = 2 \times f_S$$

Das Signal fEngine dient zur Vorgabe der Motordrehzahl in U/min. Die minimal und maximal einstellbare Motordrehzahl sind über

$$f_T = n_{\text{Bank}} \times f_S$$

$$T_S = n_{\text{Bank}} \times T_T$$

aus der minimalen und maximalen Austaktfrequenz  $f_{T,Min}$  und  $f_{T,Max}$  (siehe Tab. 7-4) zu berechnen. Die Auflösung der Motordrehzahl beträgt 2 U/min.

Maximale Austaktfrequenz $f_{T,Max}$	90 kHz
Minimale Austaktfrequenz $f_{T,Min}$	33 mHz

**Tab. 7-4** Maximale und minimale Austaktfrequenz

#### *ChnX\_SigSel*

---

Mit den Signalen *ChnX\_SigSel* (X = A, B) werden für die beiden Kanäle eines Signalgenerators die aktiven Signalbänke festgelegt.

#### *ChnX\_Mode*

---

Mit den Signalen *ChnX\_Mode* (X = A, B) werden die Modi der beiden Kanäle eines Signalgenerators festgelegt.

#### *ChnX\_PhaseShift*

---

Zu den Phasenverschiebungen, die mit den *ChnX\_PhaseShift* (X = A, B) Signalen getrennt für beide Kanäle definiert werden, addiert sich die in Abschnitt 7.3.1 beschriebene, für beide Kanäle gemeinsam gültige Grundverschiebung „Master Phase [°CA]“. Auch die Phasenverschiebung der *ChnX\_PhaseShift* Signale wird in Grad Kurbelwellenwinkel angegeben. Bei einer Phasenverschiebung von  $\pm 360^\circ$  wird der Signalverlauf um die halbe Signalbankgröße verschoben.

#### *Cs\_Angle*

---

Das *Cs\_Angle* Signal gibt den aktuellen Kurbelwellenwinkel in Grad zurück. Durch Division mit 720 und Multiplikation mit der Signalbankgröße  $n_{Bank}$  lässt sich daraus die Nummer des aktuell ausgegebenen Signalbankstützpunkts berechnen (bereinigt von jeglicher Phasenverschiebung, d.h. bei Phasenverschiebung 0).



## 8 ES1334.1 Measurement Board

### Begriffsdefinitionen

#### Aktive Signalflanke:

Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal:

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Pulsflanke:

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal:

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### LWL:

Steht für **L**ower **W**indow **L**imit und bezeichnet die untere Grenze des Winkelfensters bei drehzahlsynchronen Messungen.

#### Öffnende Pulsflanke:

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



#### UWL:

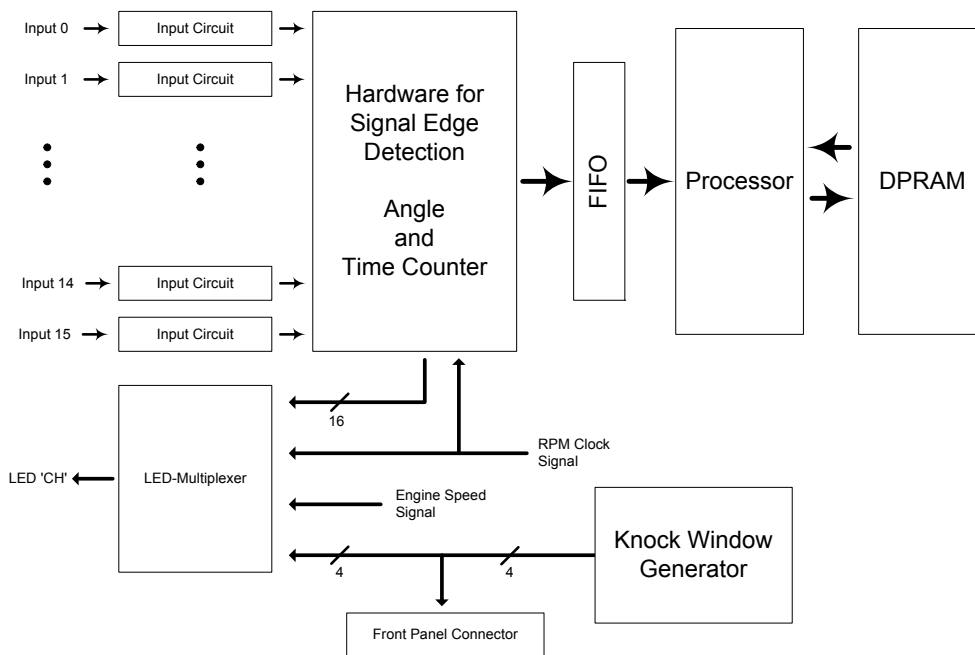
Steht für **U**pper **W**indow **L**imit und bezeichnet die obere Grenze des Winkelfensters bei drehzahlsynchronen Messungen.

## 8.1 Funktionsbeschreibung

### 8.1.1 Grundprinzip der Messwertbestimmung

Um eine möglichst große Flexibilität bei der Auswertung anliegender Eingangssignale zu erreichen, basiert das ES1334.1 Measurement Board auf folgendem Grundprinzip:

Zuerst wird jedes Eingangssignal separat (analog) aufbereitet und dann mit einem (per Software einstellbaren) Schwellwert verglichen, um die Umsetzung des analogen Eingangssignals in eine digitale 0/1- bzw. inaktiv/aktiv-Information zu realisieren. Die so erhaltene binäre Pegelinformation der Signale an den 16 Eingangskanälen wird ständig von einer Hardwareschaltung auf Änderungen hin überprüft, d.h. es wird eine Erkennung von "Flanken" auf einem oder mehreren Eingangsbits durchgeführt. Wechselt nun zumindest ein Eingangssignal von aktiv auf inaktiv bzw. umgekehrt, so speichert die Schaltung die aktuellen Zustände der 16 Eingänge und ruft die momentanen Werte zweier integrierter Zähler ab, die die aktuelle Zeit und den aktuellen Kurbelwellenwinkel angeben. Diese Informationen werden hardwaregesteuert in einen FIFO-Puffer übertragen.



**Abb. 8-1** Blockschaltbild des ES1334.1 Measurement Board

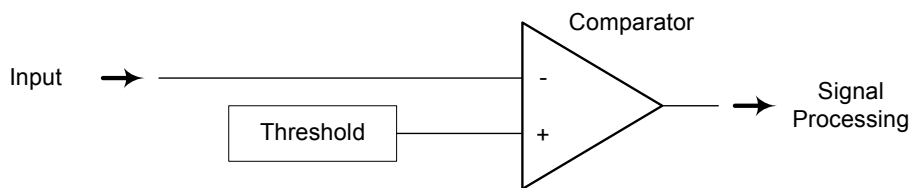
Die auf den Prozessor der ES1334.1 heruntergeladene Firmware kann nun die generierten Werte aus dem FIFO-Puffer auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben. Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind, wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten softwaretechnischen Aufbereitung der Messwerte bewirkt, dass die Auslastung des Prozessors direkt von der Zahl der erkannten Flanken (also den Frequenzen der Eingangssignale) abhängig ist. Der Vorteil dieses Konzepts: Die Hardware erfasst lediglich sämtliche Änderungen und versieht sie mit

Winkel- und Zeitstempeln. Die Auswertung erfolgt hingegen per Software, ist demzufolge verhältnismäßig leicht zu ändern bzw. anzupassen. ES1334.1-Firmware und RTIO-Treiber bieten ein breites Spektrum an implementierten Messverfahren. Das eingesetzte Konzept ermöglicht die Implementierung weiterer Messverfahren. Für die Realisierung genügt eine Anpassung der ES1334.1.1-Firmware und des RTIO-Treibers.

### 8.1.2 Eingangsbeschaltung

Die in Abb. 8-1 auf Seite 150 angedeutete Eingangsbeschaltung eines Hardwarekanals der ES1334.1 ist in Abb. 8-2 vereinfacht dargestellt. Sie besteht im Wesentlichen in einem Vergleich mit einem durch den Anwender definierten Schwellwert.



**Abb. 8-2** Vereinfachte Darstellung der Eingangsbeschaltung eines ES1334.1-Hardwarekanals.

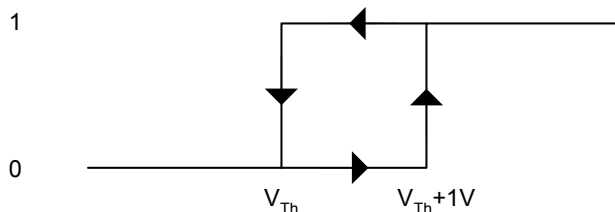
Durch diesen Vergleich wird die Umsetzung des analogen Eingangssignals in eine digitale 0/1-Information erreicht. Der Schwellwert ist derart definiert, dass sämtliche am Karteneingang anliegenden Spannungen, die größer als dieser Wert sind, als „1“ interpretiert werden. Alle Spannungen unterhalb dieses Schwellwertes werden als „0“ interpretiert.

Die Komparator-Schwellen können für die Kanäle 0 bis 7 und 8 bis 15 jeweils gemeinsam auf einen der nachfolgenden Werte eingestellt werden:

- 2.5 V
- (Batteriespannung A) /2
- (Batteriespannung B) /2
- Externe Referenz-Spannung

Gerade im Kfz-Umfeld kommen sehr häufig verhältnismäßig langsame, also niederfrequente Signale zum Einsatz, die zudem an den Flanken sehr oft auch große Anstiegs- bzw. Abfallzeiten aufweisen. Werden nun solche Signale mit einem vorgegebenen Schwellwert verglichen, so besteht – insbesondere wenn das Signal noch ein gewisses Rauschen aufweist – die Gefahr, dass beim langsamen Über- bzw. Unterschreiten der Schwelle das Vergleichsergebnis einige Male zwischen „1“ und „0“ hin und her wechselt. Um diese unerwünschten Nebeneffekte zu vermeiden, weisen die Vergleichsschaltungen auf der ES1334.1 eine

Hysterese auf, d.h. der Schwellwert für den Wechsel von "0" auf "1" liegt etwas höher als der Schwellwert für den Wechsel von "1" nach "0". Die Größe der Hysterese beträgt 1 V (siehe Abb. 8-3).



**Abb. 8-3** Komparator-Hysterese.  $V_{Th}$  bezeichnet den eingestellten Schwellwert.

### 8.1.3 Messwerterfassung

Der ES1334.1-RTIO-Treiber stellt dem Anwender 16 frei konfigurierbare Messwerte (im folgenden Software- bzw. Messkanäle genannt) zur Verfügung. Zwischen den Messkanälen und den Hardwarekanälen der ES1334.1 besteht eine 1:1-Beziehung, d.h. Messkanal 0 führt Messungen auf Hardwarekanal 0 durch, Messkanal 1 auf Hardwarekanal 1 usw.

Für jeden der 16 Messkanäle kann unabhängig von allen anderen Messkanälen definiert werden,

- wie das anliegende Signal auszuwerten ist,
- wann und wie es auf fehlende Pulse hin untersucht werden soll (*Timeout-Überwachung*) und
- ob die Auswertung dauernd stattfinden soll oder nur innerhalb eines bestimmten Bereiches des Kurbelwellenwinkels (*Winkelsegmentierung*).

Für einige Messungen lässt sich weiter festlegen, der wievielte Puls vermessen werden soll.

### 8.1.4 Timeout-Überwachung

Wie bereits in „Grundprinzip der Messwertbestimmung“ auf Seite 150 erläutert, arbeitet die ES1334.1 flankengesteuert, d.h. die Auswertung der anliegenden externen Signale findet ausschließlich an deren Flanken statt. Dies hat jedoch zur Folge, dass auch die generierten Messwerte, die der ES1334.1-RTIO-Treiber liefert, nur nach Flanken aktualisiert werden. Ein Signal, welches bereits einmal von der Karte ausgewertet wurde, würde folglich auch dann noch den zuletzt generierten gültigen Messwert liefern, wenn es inzwischen z.B. vollkommen ausgefallen ist, also keinerlei Flanken mehr aufweist.

Um Signale, die ganz oder zeitweise ausfallen, auch korrekt zu interpretieren, bietet der ES1334.1-RTIO-Treiber die Möglichkeit der Timeout-Überwachung. Jeder einzelne Messwert kann hierzu in bestimmten Abständen daraufhin getestet werden, ob er durch neu aufgetretene Flanken am Eingangssignal aktualisiert wurde. Dieser Test kann sowohl in vom Anwender definierbaren Zeitabständen als auch jeweils an vom Anwender vorgegebenen Kurbelwellenwinkeln erfolgen.

Wird während einer derartigen Überprüfung festgestellt, dass seit der letzten Überprüfung auf Timeout keinerlei Flanke von der Hardware detektiert wurde, so kann der entsprechende Messwert auf zwei verschiedene Arten modifiziert werden. Entweder wird er auf einen vorgegebenen Timeout-Wert gesetzt oder aber in Abhängigkeit vom aktuellen Pegel (high oder low) des Eingangssignals

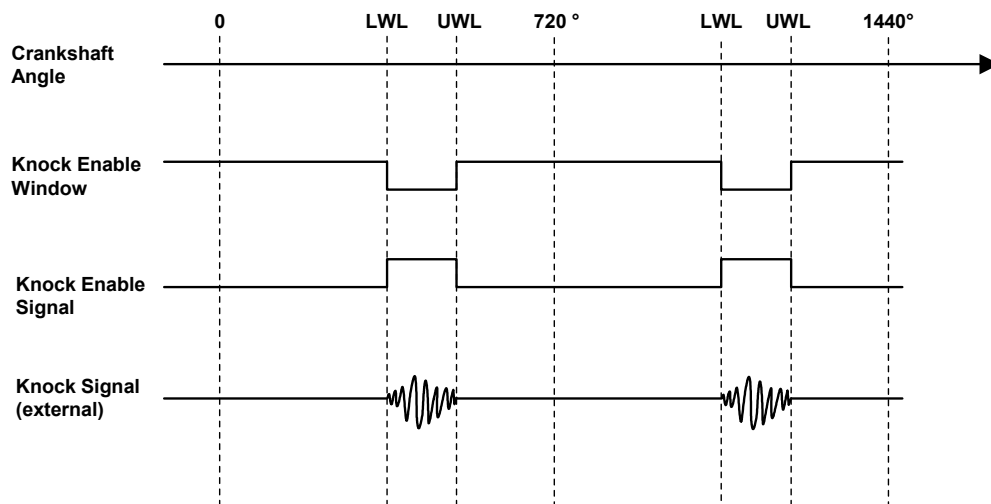
ermittelt. Dies ist insbesondere zur korrekten Auswertung von Tastverhältnissen erforderlich (ein inaktiver Eingang resultiert hierbei in den Messwerten 0.0 bzw. 1.0 – je nach Pegel des Eingangssignals).

Insgesamt ergeben sich somit sechs Möglichkeiten der Timeout-Überwachung:

- Überprüfung in Zeitabständen von  $x$  ms: Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung alle  $720^\circ$  bei  $x^\circ$  Kurbelwelle: Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung alle  $720^\circ$  bei  $x_1^\circ$  Kurbelwelle und alle  $x_2$  ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien erfüllt sind. Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung in Zeitabständen von  $x$  ms: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle  $720^\circ$  bei  $x^\circ$  Kurbelwelle: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle  $720^\circ$  bei  $x_1^\circ$  Kurbelwelle und alle  $x_2$  ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien erfüllt sind. Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.

#### 8.1.5 Enable-Signale für Klopf-Generatoren

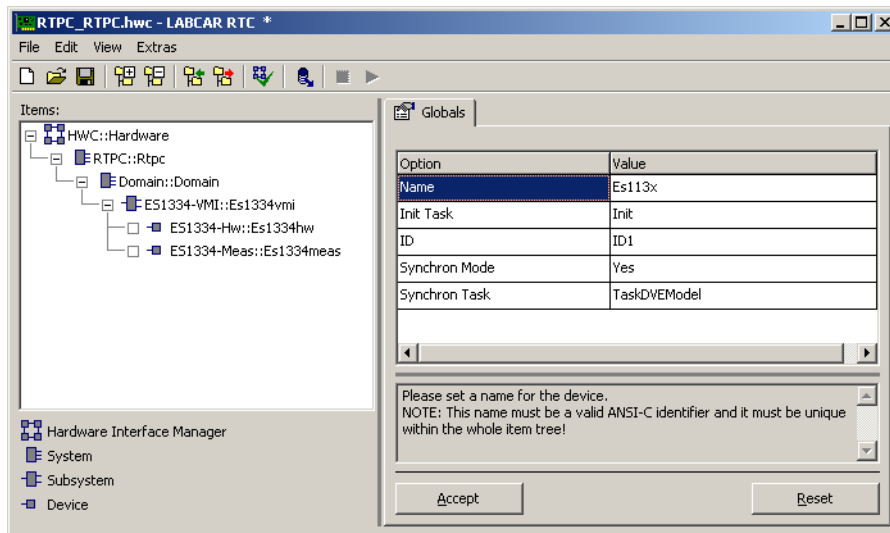
Die ES1334.1 ermöglicht auch die Generierung von vier digitalen Signalen mit TTL-Pegel zur Aktivierung bzw. Deaktivierung von Klopf-Generatoren. Bei diesen „Knock Enable“-Signalen handelt es sich um drehzahlsynchrone Signale, die innerhalb eines vom Anwender definierbaren Winkelsegments („Knock Enable Window“) aktiv geschaltet werden. Die Definition des aktiven Signalpegels (aktiv-high oder aktiv-low) ist ebenfalls konfigurierbar (Abb. 8-4).



**Abb. 8-4** High-aktives Klopf-Enable-Signal zur Aktivierung und Deaktivierung eines externen Klopfgenerators.

### 8.1.6 Aufbau des ES1334.1-RTIO-Baums

Im RTIO-Editor wird das ES1334.1 Measurement Board durch Auswahl des „ES1334-VMI“ Systems eingebunden.



**Abb. 8-5** RTIO-Hardwarebeschreibung mit eingebundener ES1334.1

Die ES1334.1 bietet 16 Hardwarekanäle zur Erfassung und Aufbereitung analoger und digitaler Signale sowie 4 Ausgangssignale zur Aktivierung und Deaktivierung von externen Klopfgeneratoren. Die Konfiguration der 16 Hardwarekanäle und der 4 Klopf-Enable-Ausgänge erfolgt mit dem ES1334-Hw Device, das dem ES1334-VMI Subsystem zugeordnet wird. Neben diesem Item lässt sich dem ES1334-VMI Subsystem ein sogenanntes ES1334-Meas Device zuordnen. Mit diesem Item lassen sich bis zu 16 Messungen spezifizieren.

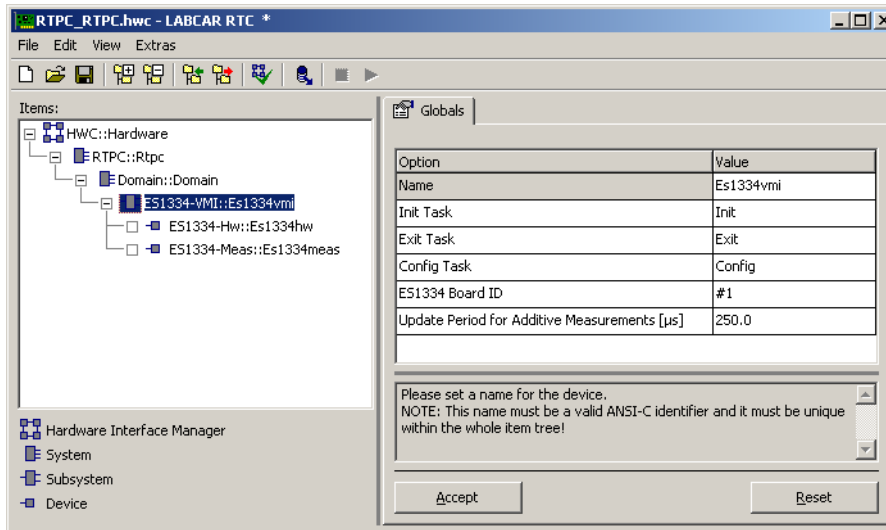
#### **Hinweis**

*Bevor ein ES1334-Meas Device eingefügt werden kann, muss zunächst ein ES1334-Hw Device zur Konfiguration der ES1334.1-Hardware eingefügt werden.*

## 8.2 ES1334-VMI Subsystem

### 8.2.1 Globals (ES1334-VMI Subsystem)

Abb. 8-6 zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 8-6** Die Registerkarte „Globals“ des ES1334-VMI Subsystems

#### Hinweis

Wenn das ES1334.1 Measurement Board in eine LABCAR-RTC-Hardwarekonfiguration eingebunden wird, müssen die beiden Adress-Schalter SW1 und SW2 auf dem Board die Einstellung „0x00“ haben!

#### ES1334 Board ID

Dieses Optionsfeld dient zur Identifizierung der ES1334.1 – es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1334.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist. Die Nummerierung der ES1334.1-Boards im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Nummerierung) beginnend mit 1. Die ermittelte Nummer der ES1334.1 ist in dem Listenfeld „ES1334 Board ID“ einzustellen. Dieser RTIO-Parameter ist nicht zur Laufzeit editierbar.

Im RTIO-Editor können bis zu zwölf ES1334.1 Measurement Board pro Baugruppenträger eingebunden werden.

#### Update Period for Additive Measurements [ $\mu$ s]

Dieser Parameter ist nur für Software- bzw. Messkanäle von Bedeutung, die additive Zeitmessungen im asynchronen Modus durchführen. Er legt fest, in welchen Zeitabständen spätestens eine Aktualisierung der additiven Zeit stattfindet. Weitere Informationen hierzu enthält der Abschnitt „Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen“ auf Seite 174. Der Parameter ist zur Laufzeit des Modells auf dem Experimentaltarget konfigurierbar.

Tab. 8-1 fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
ES1334 Board ID	uint32	Nein	Identifizierung der ES1334.1. Zur Auswahl stehen #1 bis #12.
Update Period for Additive Measurements [ $\mu$ s]	real32	Ja	Messwert-Aktualisierungsrate für asynchrone additive Messungen in Mikrosekunden. Minimal: 50 $\mu$ s Maximal: 500 ms

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

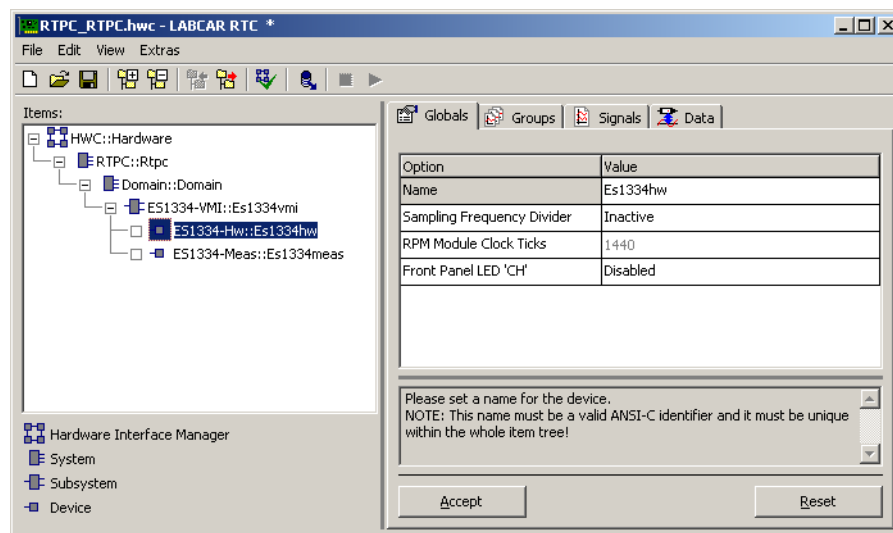
**Tab. 8-1** ES1334-VMI Subsystem: Konfigurationsparameter der Registerkarte „Globals“

### 8.3 Hardwarekonfiguration - ES1334-Hw Device

Das ES1334-Hw Device dient zur Konfiguration und Ansteuerung der ES1334.1 Hardware. Von den einzelnen Hardwarekanälen unabhängige Einstellungen werden in der Registerkarte „Globals“ dieses Items durchgeführt. In der Registerkarte „Signals“ werden kanalspezifische Einstellungen vorgenommen.

#### 8.3.1 Globals (ES1334-Hw Device)

Abb. 8-7 zeigt die RTIO-Parameter der Registerkarte „Globals“. Tab. 8-2 auf Seite 159 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 8-7** Die Registerkarte „Globals“ des ES1334-Hw Device

#### *Sampling Frequency Divider*

Die ES1334.1 tastet die Eingangssignale an den Hardwarekanälen periodisch ab. Die maximale Abtastfrequenz ist durch die Frequenz des internen Taktgebers bestimmt, die 4 MHz beträgt. Mit einem Frequenzteiler kann die Abtastfrequenz reduziert werden. Die Konfiguration des Frequenzteilers erfolgt im Optionsfeld

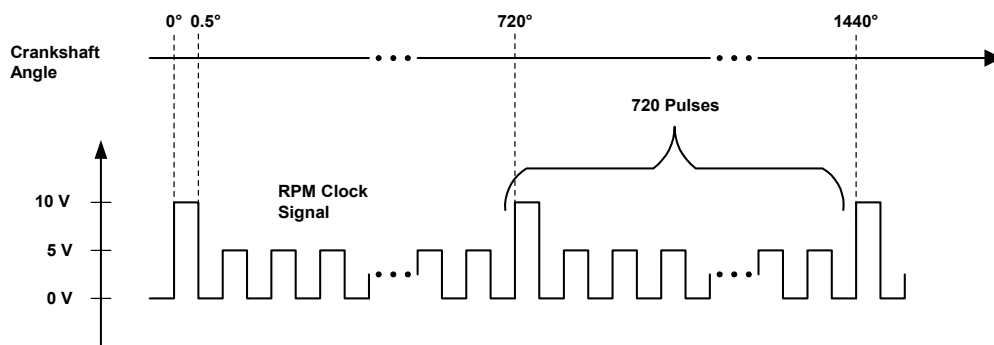


„Sampling Frequency Divider“ der Registerkarte „Globals“. Die Frequenzteilung kann deaktiviert oder auf die Teiler 2, 4, 8, 16 oder 32 konfiguriert werden. Es ist zu beachten, dass nur Eingangspulse mit einer Dauer größer als die Abtastperiodendauer sicher erfasst werden

Der Parameter ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

#### RPM Module Clock Ticks

Zur Durchführung von zur Motordrehzahl synchronen Messungen benötigt die ES1334.1 Informationen über die Winkelstellung des Motors. Diese Informationen ermittelt die ES1334.1 aus dem Winkeltaktsignal, das ihr an der Frontplatte über den „SYNC“-Eingang zugeführt wird. Das Winkeltaktsignal (siehe Abb. 8-8) wird von einem ES1332.1 Arbitrary Signal Generator Board erzeugt und ermöglicht die Synchronisation auf den Nulldurchgang des Nockenwellenwinkels und die Berechnung der Winkelstellung des Motors.



**Abb. 8-8** Winkeltaktsignal zur Synchronisation auf den Nulldurchgang des Nockenwellenwinkels und Berechnung der Winkelstellung des Motors

Das numerische Eingabefeld „RPM Module Clock Ticks“ dient zur Definition der Winkelauflösung, d.h. der Anzahl der Winkeltakte pro Nockenwellenumdrehung (720° Kurbelwelle). Diese Option ist fest, d.h. nicht editierbar auf den Wert 1440 eingestellt, da das von ES1332.1-Karten generierte Winkeltaktsignal diese Anzahl an Ticks bzw. Flanken erzeugt. Der eingestellte Wert von 1440 Winkeltakten pro Umdrehung der Nockenwelle entspricht einer Winkelauflösung von 0.5 °CA (°CA = Grad Kurbelwelle).

#### Front Panel LED „CH“

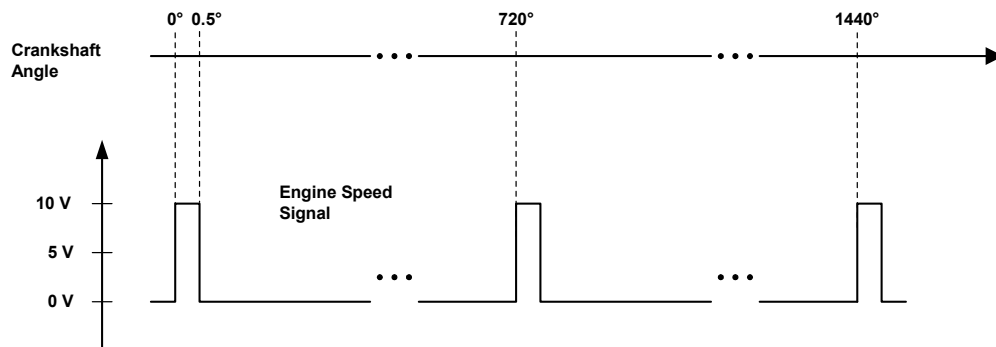
Auf der Frontplatte der ES1334.1-Karte befindet sich eine LED mit der Bezeichnung „CH“. Mit dieser LED lässt sich der Signalpegel

- auf einem der 16 Hardwarekanäle oder
- auf einem der 4 Klopf-Enable-Ausgänge oder
- des Winkeltaktsignals (siehe Abb. 8-8 auf Seite 157) oder
- des Synchronisationssignals auf den Nulldurchgang des Nockenwellenwinkels (siehe Abb. 8-9)

optisch anzeigen. Weist das Signal einen Low-Pegel auf, so wird die LED dunkel geschaltet. Weist das Signal einen High-Pegel auf, so wird die LED hell geschaltet. Die Definition von Low- und High-Pegel ist bei den 16 Eingangskanälen durch die Schwellspannung des Eingangskomparators des jeweiligen Kanals

gegeben. Bei den Klopf-Enable-Ausgängen und dem Winkeltaktsignal ist die Schaltschwelle zur Unterscheidung von High- und Low-Pegel 2,5 V, beim Synchronisationssignal auf den Nulldurchgang des Nockenwellenwinkels beträgt die Schaltschwelle 7,5 V.

Die Registerkarte „Globals“ bietet für die LED „CH“ ein Optionsfeld „Front Panel LED „CH““ mit dem das Signal eingestellt wird, das die LED ansteuert, bzw. mit dem die LED deaktiviert werden kann. Das Optionsfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 8-9** Synchronisationssignal auf den Nulldurchgang des Nockenwellenwinkels.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Sampling Frequency Divider	uint8	Ja	Frequenzteiler für Eingangssignal-Abtastung 0: Keine Frequenzteilung (4 MHz Abtastung) 1: Teilung durch 2 (2 MHz) 2: Teilung durch 4 (1 MHz) 3: Teilung durch 8 (500 kHz) 4: Teilung durch 16 (250 kHz) 5: Teilung durch 32 (125 kHz)
RPM Module Clock Ticks	uint32	Nein	Winkelauflösung Ist fest, nicht editierbar auf 1440 eingestellt.
Front Panel LED „CH“	sint8	Ja	Signal bzw. Kanal der auf die Frontplatten LED „CH“ geschaltet ist bzw. Deaktivierung der LED. -1: Deaktivierung 0...15: Hardwarekanal #0... #15 16...19: Klopf-Enable-Ausgang #0...#3 20: Winkeltaktsignal (RPM Clock) 21: Synchronisationssignal auf den Nulldurchgang des Nockenwellenwinkels (Engine Speed).
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 8-2** ES1334-Hw Device: Konfigurationsparameter der Registerkarte „Globals“

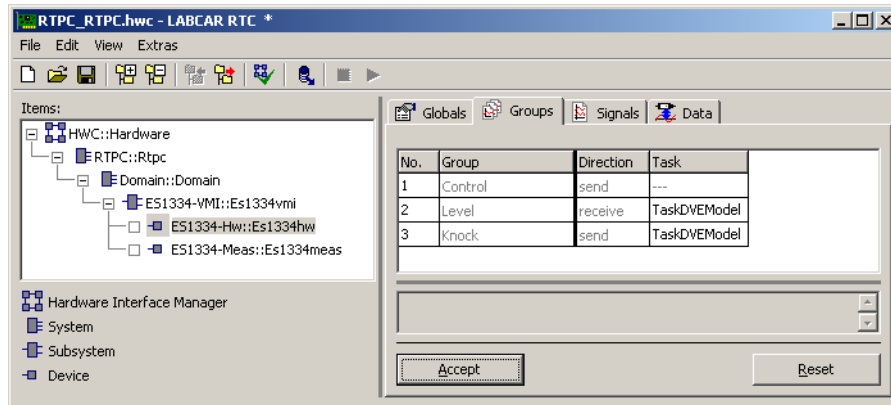
### 8.3.2 Groups (ES1334-Hw Device)

Das ES1334-Hw Device besitzt drei Signalgruppen (siehe Abb. 8-10). Die Signalgruppe „Control“ ist für zukünftige Erweiterungen gedacht. Da sie noch keine Bedeutung hat, kann dieser Signalgruppe keine Task des Echtzeitbetriebssystems zugeordnet werden.

Die Signalgruppe „Level“ wird von der ES1334.1 zum Experimentaltarget übertragen. Diese Signalgruppe enthält die aktuellen Pegelinformationen aller Hardwarekanäle auf denen Messungen durchgeführt werden.

Die Signalgruppe „Knock“ wird vom Experimentaltarget zur ES1334.1 übertragen. Mit dieser Signalgruppe werden die Klopf-Enable-Ausgänge der Karte ein- und ausgeschaltet.

Den „Level“- und „Knock“-Signalgruppen sind ERCOS<sup>EK</sup>-Betriebssystem-Tasks zuzuordnen. Falls man die Pegelinformationen im Simulationsmodell auswertet, wird man der Signalgruppe „Level“ eine Task vom Typ „Timer“ zuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik bzw. der Periodendauer der zu erfassenden Signale.



**Abb. 8-10** Die Registerkarte „Groups“ des ES1334-Hw Device

*RTIO-Signale der Signalgruppe „Level“*

Die Signalgruppe „Level“ umfasst ein RTIO-Signal „LvlBitField\_0“, das als Bitfeld zu interpretieren ist. Das Bitfeld umfasst 16 Bit, in ihm sind die Pegelinformationen der Hardwarekanäle 0 bis 15 codiert. Tab. 8-3 fasst die Eigenschaften des RTIO-Signals zusammen.

#### **Hinweis**

*Nur die Pegelbits von Hardwarekanälen, bei denen die Flankenerkennung aktiviert ist, sind im Bitfeld „LvlBitField\_0“ gültig. Die Flankenerkennung auf einem Hardwarekanal ist aktiviert, wenn auf ihm eine Messung durchgeführt wird.*

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
LvlBitField_0	uint16	Bitfeld mit Pegelinformationen der Hardwarekanäle 0 bis 15. Kanal 0: LSB (Least Significant Bit) Kanal 15: MSB (Most Significant Bit) Bitwert 0: Low Pegel Bitwert 1: High Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 8-3** ES1334-Hw Device: Die RTIO-Signale der Signalgruppe „Level“

RTIO-Signale der Signalgruppe „Knock“

Die Signalgruppe „Knock“ umfasst 4 RTIO-Signale, mit denen die 4 Klopf-Enable-Ausgänge der Karte ein- und ausgeschaltet werden können. Tab. 8-4 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
KnckEnable_0	bool	Ein-/Ausschalten der Klopf-Enable-Ausgänge 0 bis 3
...		
KnckEnable_3		0: Ausschalten 1: Einschalten

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

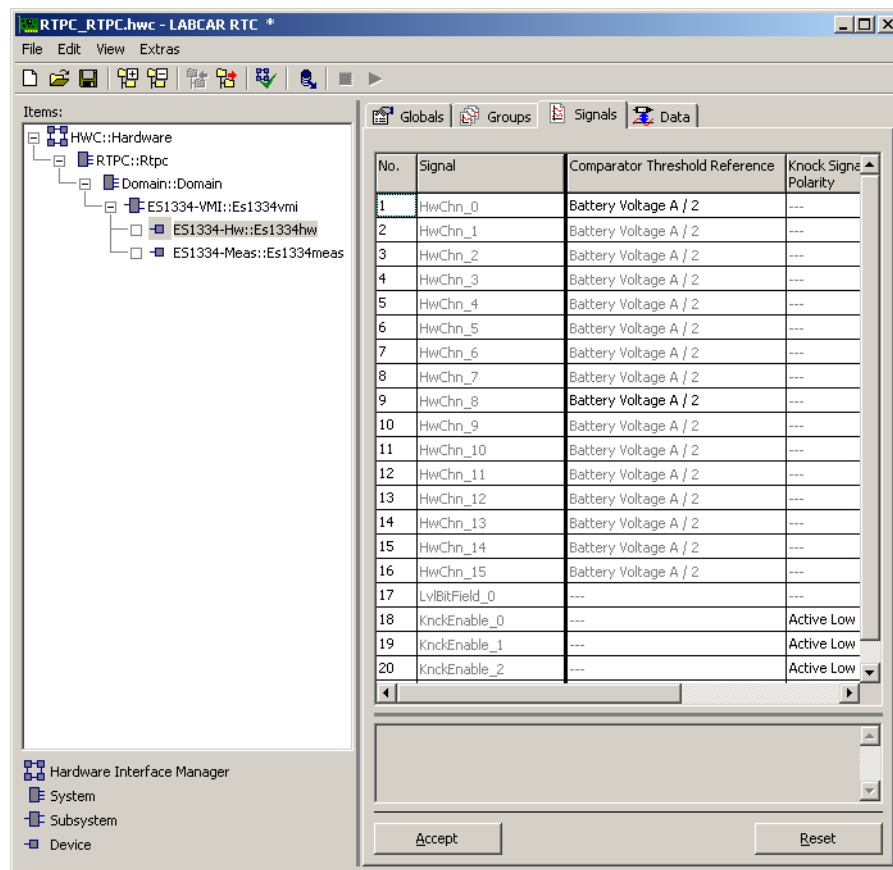
**Tab. 8-4** ES1334-Hw Device: Die RTIO-Signale der Signalgruppe „Knock“

**Hinweis**

Die „Knock“-Signalgruppe wird nur dann in das DPRAM der ES1334.1 geschrieben, wenn sich eines ihrer RTIO-Signale im Wert verändert hat.

### 8.3.3 Signals (ES1334-Hw Device)

In der Registerkarte „Signals“ wird die Konfiguration der 16 Hardwarekanäle einer ES1334.1 durchgeführt. Abb. 8-11 zeigt die RTIO-Parameter der Registerkarte „Signals“. Tab. 8-5 auf Seite 163 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 8-11** Die Registerkarte „Signals“ des ES1334-Hw Items

#### *Comparator Threshold Reference*

Das Optionsfeld „Comparator Threshold Reference“ dient zur Festlegung der Schwellspannung eines Eingangskomparators. Die Komparatorschwellen können für die Kanäle 0 bis 7 und 8 bis 15 nur jeweils gemeinsam auf einen der nachfolgenden Werte eingestellt werden:

- 2,5 V
- (Batteriespannung A)/2
- (Batteriespannung B)/2
- Externe Referenz-Spannung

#### *Knock Signal Polarity*

Polarität des Klopf-Enable-Signals (aktiv-low oder aktiv-high).

*Knock Window Lower CA Limit*

Untere Grenze des Klopfensters in Grad Kurbelwelle.

*Knock Window Upper CA Limit*

Obere Grenze des Klopfensters in Grad Kurbelwelle.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Comparator Threshold Reference	uint8	Ja	Komparatorreferenz 0: 2,5 V 1: (Battery Voltage A)/2 2: (Battery Voltage B)/2 3: External Reference Voltage
Knock Signal Polarity	uint8	Ja	Polarität des Klopf-Enable-Signals 0: Active Low 1: Active High
Knock Window Lower CA Limit	real32	Ja	Untere Grenze des Klopfensters in Grad Kurbelwelle. Wertebereich: -720 °CA bis 720 °CA
Knock Window Upper CA Limit	real32	Ja	Obere Grenze des Klopfensters in Grad Kurbelwelle. Wertebereich: -720 °CA bis 720 °CA

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

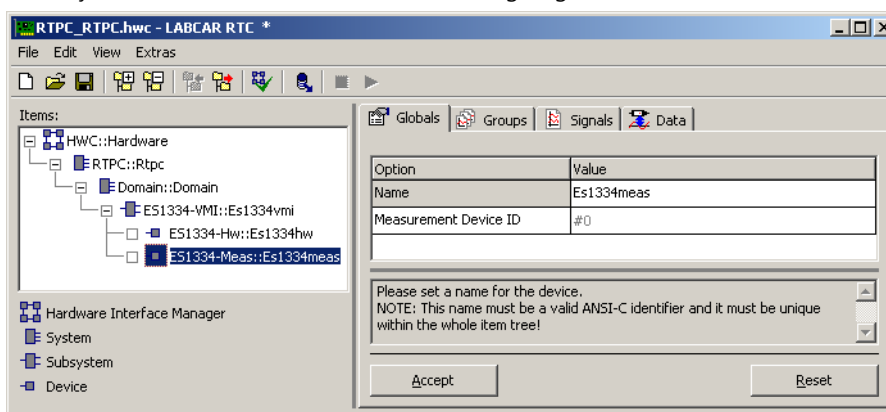
**Tab. 8-5** ES1334-Hw Device: Konfigurationsparameter der Registerkarte „Signals“

## 8.4 Messkonfiguration - ES1334-Meas Device

Das ES1334-Meas Device dient zur Spezifikation und Konfiguration von Messungen. Es bietet 16 Softwarekanäle (im folgenden Messkanäle genannt), die in einer 1:1-Beziehung den 16 Hardwarekanälen zugeordnet sind, d.h. Messkanal 0 führt Messungen auf Hardwarekanal 0 durch, Messkanal 1 auf Hardwarekanal 1 usw.

### 8.4.1 Globals (ES1334-Meas Device)

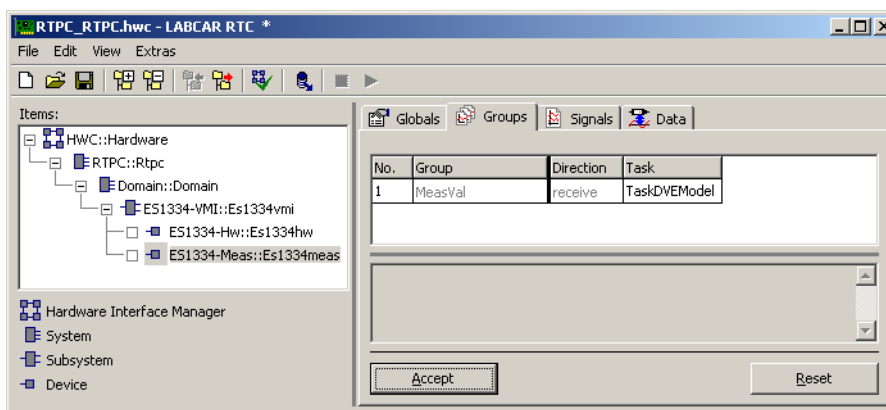
In der Registerkarte „Globals“ (Abb. 8-12) sind keine Einstellungen vorzunehmen. Im Optionsfeld „Measurement Device ID“ wird dem ES1334-Meas Device vom System eine Nummer zur Identifizierung zugewiesen.



**Abb. 8-12** Die Registerkarte „Globals“ des ES1334-Meas Device

### 8.4.2 Groups (ES1334-Meas Device)

Das ES1334-Meas Device besitzt eine Signalgruppe (Abb. 8-13), die von der ES1334.1 zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen. Dieser Signalgruppe ist eine ERCOS<sup>EK</sup>-Betriebsystem-Task vom Typ „Timer“ zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.



**Abb. 8-13** Die Registerkarte „Groups“ des ES1334-Meas Device



### RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 33 RTIO-Signale. Das „TriggerBitField\_0“-Signal ist als Bitfeld zu interpretieren. Es umfasst 16 Bit. In diesem Bitfeld sind die Trigger- oder Update-Daten der 16 Messkanäle codiert, d.h. es zeigt an, welche Messwerte seit der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der Messwert des zugehörigen Messkanals neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeout oder aufgrund einer regulären Messwertberechnung erfolgte, in beiden Fällen wird das Aktualisierungs-Bit des Messkanals gesetzt.

Die RTIO-Signale „MeasVal\_0“ bis „MeasVal\_15“ enthalten die Messwerte der 16 Messkanäle. Es handelt sich dabei um regulär ermittelte Messwerte oder im Falle eines Timeout um den für diesen Fall vorgesehenen Timeout-Wert. Falls ein Messkanal nicht verwendet wird, wird dem zugehörigen Messwert -8888.0 zugewiesen. Die physikalische Einheit der Messwerte richtet sich nach dem Messverfahren:

- Zeitmessungen (Zeitstempel, (additive) Pulsdauer-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Frequenzmessungen erfolgen in Hertz
- Winkelmessungen und Winkelstempel werden in Grad Kurbelwelle (°CA) angegeben
- alle anderen Messungen (Tastverhältnisse, Pulszählung, Pegelmessungen) sind dimensionslos

Die RTIO-Signale „Tout\_0“ bis „Tout\_15“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messkanal.

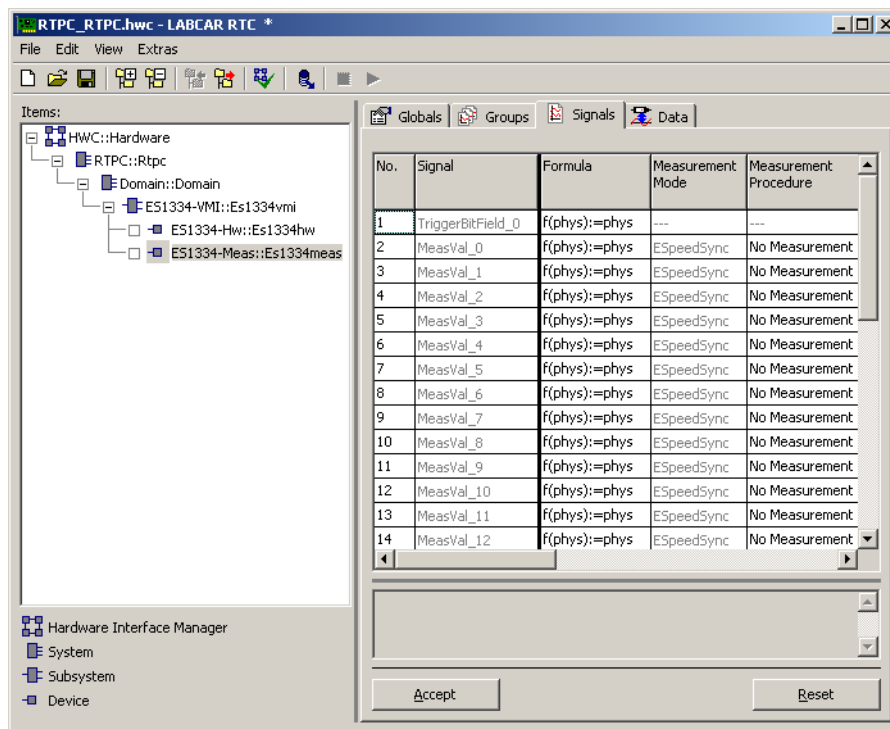
Tab. 8-6 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
TriggerBitField_0	uint16	Bitfeld mit Update-Informationen der 16 Messkanäle. Messkanal 0: LSB Messkanal 15: MSB Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasVal_0 ... MeasVal_15	real64	Messwert Falls der Messkanal nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: Zeitmessungen erfolgen in Mikrosekunden Frequenzmessungen erfolgen in Hertz Winkelmessungen erfolgen in °CA
Tout_0 ... Tout_15	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout aufgetreten 1: Timeout aufgetreten 2: Timeout-Überwachung ist inaktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 8-6** ES1334-Meas Device: Die RTIO-Signale der Signalgruppe „MeasVal“

### 8.4.3 Signals (ES1334-Meas Device)

In der Registerkarte „Signals“ wird die Konfiguration der 16 Messkanäle eines ES1334-Meas Device durchgeführt. Abb. 8-14 zeigt die RTIO-Parameter der Registerkarte „Signals“. Tab. 8-7 auf Seite 169 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 8-14** Die Registerkarte Signals des ES1334-Meas Device

#### Measurement Mode

Das Optionsfeld „Measurement Mode“ definiert den Modus, in dem eine Messung durchgeführt wird. Es stehen zwei Optionen zur Auswahl:

- „ESpeedSync“: Durchführung von drehzahlsynchronen bzw. winkelsynchronen Messungen. Die Durchführung von drehzahlsynchronen Messungen erfordert, dass die ES1334.1 über den „SYNC“-Eingang an der Frontplatte mit einem Winkeltaktsignal versorgt wird.
- „Asynchronous“: Durchführung von asynchronen Messungen.

#### Measurement Procedure

In dieser Listbox wird das Messverfahren ausgewählt. Detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren finden Sie ab Abschnitt 8.5 auf Seite 171. In der Listbox bedeutet die Notation --/-- eine Auswertung an steigenden Flanken, die Notation --\-- hingegen eine Auswertung an fallenden Flanken.

Wird in der Listbox die Option „No Measurement“ eingestellt, so wird keine Messung durchgeführt und die Flankenerkennung auf dem zugeordneten Hardwarekanal wird deaktiviert.

#### **Hinweis**

*Zur Vermeidung unnötiger Rechenzeiten sind nicht benötigte Messkanäle zu deaktivieren.*

#### *Timeout Monitoring*

---

Definition der Timeout-Überwachung für den zugehörigen Messkanal (siehe auch „Timeout-Überwachung“ auf Seite 152).

Die folgenden Einstellungen sind möglich:

- „Inactive“: Keine Timeout-Überwachung.
- „Intvl Predef“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl InpDep“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.
- „CS Angle Predef“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „CS Angle InpDep“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.
- „Intvl & CS Angle Predef“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel und Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Ein Timeout wird erkannt, wenn eine oder beide Timeout-Kriterien verletzt sind. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl & CS Angle InpDep“: Wie „Intvl & CS Angle Predef“, jedoch ist der Messwert im Timeout-Fall vom Pegel des Eingangssignals abhängig.

#### *Default Timeout Value*

---

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls einer der „Intvl Predef“, „CS Angle Predef“ oder „Intvl & CS Angle Predef“ Timeout-Überwachungsmodi eingestellt ist.

#### *Timeout Interval [ms]*

---

Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Nur relevant für die zeitbasierten Timeout-Überwachungsmodi („Intvl Predef“, „Intvl InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Timeout Check Point*

---

Kurbelwellenwinkel an dem Timeout-Kontrollen durchgeführt werden. Nur relevant für die winkelbasierten Timeout-Überwachungsmodi („CS Angle Predef“, „CS Angle InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Lower Limit*

---

Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.

#### *CS Angle Upper Limit*

---

Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“. Die obere Winkelfenstergrenze wird zudem bei additiven Zeitmessungen und Pulszählungen zum Transfer des ermittelten Messwertes in das DPRAM der ES1334.1 genutzt.

#### *CS Angle Reference*

---

Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Measurement Mode	uint8	Ja	Messmodus 0: „Asynchronous“: Asynchrone Messungen 1: „ESpeedSync“: Drehzahl- bzw. winkelsynchrone Messungen
Measurement Procedure	sint32	Ja	Auswahl des Messverfahrens oder Deaktivierung des Messkanals.  („Messverfahren“ Parameterwert) Deaktivierung des Messkanals -1 „Hightime (Pulse Qual.) [µs]“ 0 „Lowtime (Pulse Qual.) [µs]“ 1 „Hightime (Edge Qual.) [µs]“ 59 „Lowtime (Edge Qual.) [µs]“ 60  „Additive Hightime [µs]“ 2 „Additive Lowtime [µs]“ 3  „Cycle Time --/-- [µs]“ 18 „Cycle Time --\-- [µs]“ 19 „Frequency --/-- [Hz]“ 4 „Frequency --\-- [Hz]“ 5  „Duty Factor L/H --/--“ 6 „Duty Factor L/H --\--“ 7 „Duty Factor H/L --/--“ 8 „Duty Factor H/L --\--“ 9 „Duty Cycle L/(L+H) --/--“ 10 „Duty Cycle L/(L+H) --\--“ 11 „Duty Cycle H/(L+H) --/--“ 12 „Duty Cycle H/(L+H) --\--“ 13  „Rising Edge --/-- [deg]“ 14 „Falling Edge --\-- [deg]“ 15  „Number of Low-Pulses“ 16 „Number of High-Pulses“ 17  „Level (Active High)“ 49 „Level (Active Low)“ 50
Timeout Monitoring	uint8	Ja	Timeout-Überwachung: 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“ 3: „CS Angle Predef“ 4: „CS Angle InpDep“ 5: „Intvl & CS Angle Predef“ 6: „Intvl & CS Angle InpDep“

**Tab. 8-7** ES1334-Meas Device: Konfigurationsparameter der Registerkarte „Signals“

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant in den „... Predef“-Modi zur Timeout-Überwachung.
Timeout Interval [ms]	real32	Ja	Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Wertebereich: $\geq 1.0$ ms Nur relevant in den "Intvl ..."-Modi zur Timeout-Überwachung
CS Angle Timeout Check Point	real32	Ja	Kurbelwellenwinkel an dem Timeout-Kontrollen durchgeführt werden. Wertebereich: $-720$ °CA bis $720$ °CA Nur relevant in den „CS Angle ...“-Modi zur Timeout-Überwachung.
CS Angle Lower Limit	real32	Ja	Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °CA bis $720$ °CA Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Upper Limit	real32	Ja	Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °CA bis $720$ °CA Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Reference	real32	Ja	Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung. Wertebereich: $-720$ °CA bis $720$ °CA.
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 8-7** ES1334-Meas Device: Konfigurationsparameter der Registerkarte „Signals“

## 8.5 Die Messverfahren

---

### 8.5.1 Pulsweitenmessungen

---

In diesem Abschnitt werden die Eigenschaften von Pulsweitenmessungen beschrieben. Es wird dabei eine Unterscheidung in „puls-qualifizierte“ und „flanken-qualifizierte“ Messungen vorgenommen. Im einzelnen wird die Funktionsweise folgender Messungen erläutert:

#### *Puls-qualifizierte Pulsweitenmessungen*

---

- „Hightime (Pulse Qual.) [µs]“
- „Lowtime (Pulse Qual.) [µs]“

#### *Flanken-qualifizierte Pulsweitenmessungen*

---

- „Hightime (Edge Qual.) [µs]“
- „Lowtime (Edge Qual.) [µs]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

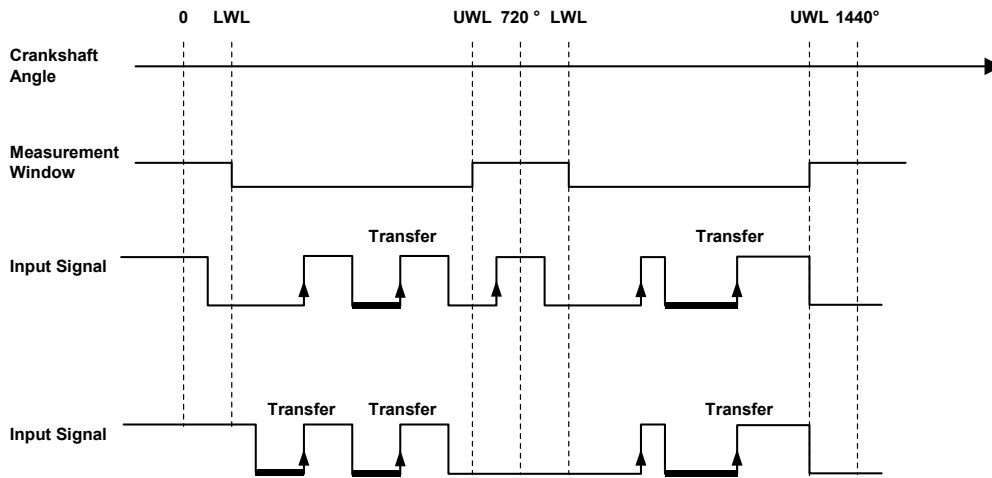
#### *Funktionsbeschreibung: Drehzahlsynchrone Pulsweitenmessungen*

---

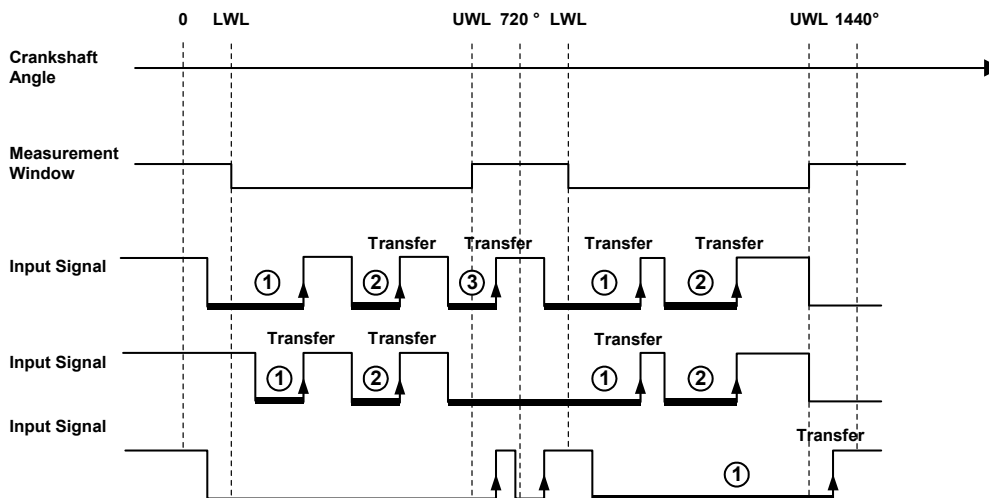
Bei den *puls-qualifizierten Messungen* werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen (gültiger Puls). Abb. 8-15 veranschaulicht die Funktionsweise einer puls-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [µs]“).

Bei den *flanken-qualifizierten Messungen* werden solche Pulse vermessen bei denen die öffnende oder die schließende Flanke innerhalb eines (beliebigen) Winkelfensters liegen (gültiger Puls). Pulse, bei denen beide Flanken außerhalb eines Winkelfensters liegen, werden nicht vermessen bzw. berücksichtigt. Abb. 8-16 veranschaulicht die Funktionsweise einer flanken-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung („Lowtime (Edge Qual.) [µs]“).

Die Messungen erfassen jeden gültigen Puls. Der Transfer der gemessenen Pulsweite in das DPRAM der ES1334.1 erfolgt an der schließenden Flanke des Pulses.



**Abb. 8-15** Darstellung der Funktionsweise einer puls-qualifizierten, low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.



**Abb. 8-16** Darstellung der Funktionsweise einer flanken-qualifizierten, low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Edge Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben. Die Zuordnung der Pulse zu den Winkelfenstern und die Zählung der Pulse innerhalb eines Winkelfensters ist ebenfalls dargestellt.



### Funktionsbeschreibung: Asynchrone Pulsweitenmessungen

Bei den asynchronen Pulsweitenmessungen ist eine Unterscheidung in *puls-qualifizierte* und *flanken-qualifizierte* Messungen nicht möglich bzw. nicht nötig, da es keine Fensterfunktionalität zur Selektion von Pulsen gibt. Die puls-qualifizierten Pulsweitenmessungen haben im asynchronen Modus die gleiche Funktionalität wie die flanken-qualifizierten Messungen.

Es wird die Pulsweite jedes Pulses mit aktivem Pegel gemessen. Der Messwert wird an der schließenden Pulsflanke ins DPRAM der ES1334.1 geschrieben.

### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-8 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	- Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, so wird als Messwert die Zeitdauer seit der letzten öffnenden Flanke des Signals ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-8** Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

## 8.5.2 Additive Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften additiver Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen dargestellt:

- „Additive Hightime [µs]“
- „Additive Lowtime [µs]“

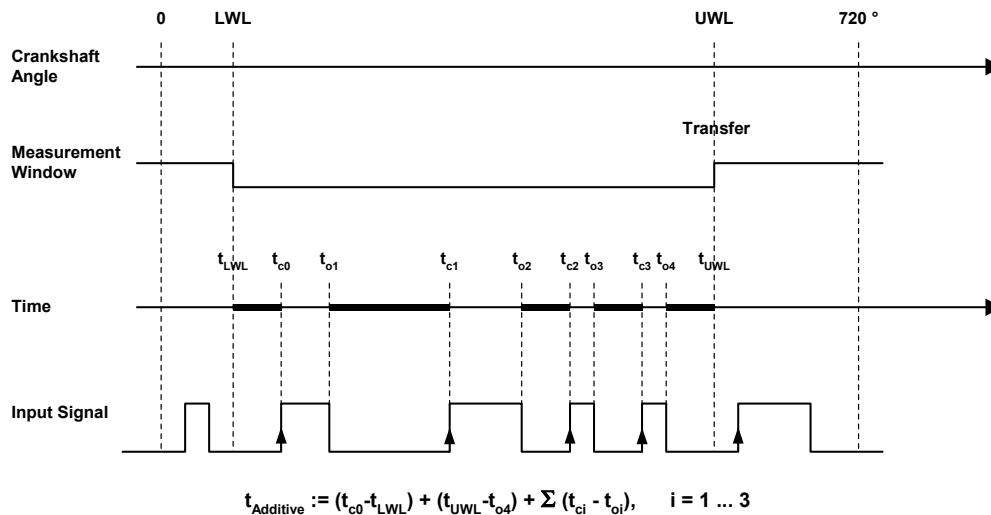
Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus
- Timeout-Überwachung

### Funktionsbeschreibung: Drehzahlsynchrone additive Pulsweitenmessungen

Die additive Zeit ergibt sich als Summe aller Zeitsegmente innerhalb eines Winkelfensters in denen das Signal aktiv ist, unabhängig davon, ob sich die öffnenden oder schließenden Flanken der Pulse innerhalb oder außerhalb eines Winkelfensters befinden. Abb. 8-17 zeigt beispielhaft die Messwertberechnung bei einer additiven Lowzeit-Messung.

Die additive Zeit wird an der oberen Winkelfenstergrenze in das DPRAM der ES1334.1 transferiert.



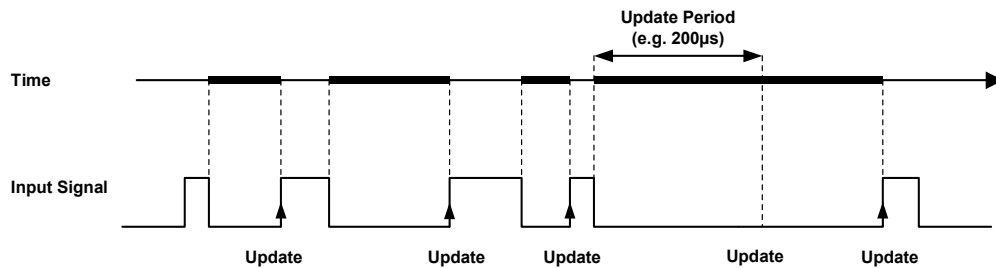
**Abb. 8-17** Darstellung der Funktionsweise einer drehzahlsynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [μs]“). Die additive Zeit setzt sich aus der Summe der fett markierten Liniensegmente zusammen.

### Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen

Bei asynchronen additiven Pulsweitenmessungen wird die Zeit summiert, in der das zu vermessende Signal aktiv ist. Die Aktualisierung der Summe im DPRAM erfolgt an jeder schließenden Pulsflanke, spätestens jedoch nach einer vom Anwender definierten Aktualisierungs-Periode. Abb. 8-18 zeigt anhand einer additiven Lowzeit-Messung, wann die Aktualisierung der additiven Zeit erfolgt. Die Einstellung der Aktualisierungs-Periode durch den Anwender erfolgt mit dem „Update Period for Additive Measurements [μs]“ Parameter in der Registerkarte „Globals“ des ES1334-VMI Item. Eine Initialisierung oder eine Konfiguration der ES1334.1 setzen die summierte Zeit auf 0 zurück.

#### Hinweis

Die Erfassung der additiven Zeit erfolgt modulo  $2^{52} \mu\text{s}$ , d.h. nach  $(2^{52} - 1) \mu\text{s}$  (entspricht etwa 143 Jahren) springt die additive Zeit wieder auf 0 um.  $(2^{52} - 1)$  ist die größte ganze Zahl des Typs „double precision“, die mit einer Auflösung von 1,0 darstellbar ist.



**Abb. 8-18** Darstellung der Funktionsweise einer asynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [µs]“). Die additive Zeit setzt sich aus den fett markierten Zeitsegmenten zusammen. Die Zeitpunkte, an denen die Summe im DPRAM aktualisiert wird, sind ebenfalls dargestellt.

#### Timeout-Überwachung

Die Bedingung, wann ein Timeout ausgelöst wird, ist in Tab. 8-9 für verschiedene Verfahren zur Timeout-Überwachung aufgelistet. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-10 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „Intvl Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde <i>und</i> das Signal inaktiv war. Es spielt dabei im drehzahlsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„CS Angle InpDep“ „CS Angle Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde. Es spielt dabei im drehzahlsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„Intvl & CS Angle InpDep“ „Intvl & CS Angle Predef“	Die Timeout-Bedingung für diese Überwachungsverfahren ergibt sich durch eine ODER-Verknüpfung der Timeout-Bedingung der „Intvl InpDep“- und „Intvl Predef“-Verfahren mit der Timeout-Bedingung der „CS Angle InpDep“- und „CS Angle Predef“-Verfahren.

**Tab. 8-9** Additive Pulsweitenmessungen: Timeout-Bedingung für verschiedene Verfahren zur Timeout-Überwachung

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	<i>Asynchroner Messmodus:</i> Als Messwert im Timeout-Fall wird die bis zum Zeitpunkt der Überprüfung auf Timeout ermittelte additive Zeit ausgegeben. <i>Drehzahlsynchroner Messmodus:</i> Als Messwert im Timeout-Fall wird 0 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-10** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 8.5.3 Frequenz- und Zykluszeitmessungen

Die ES1334.1 ermöglicht Messungen von Frequenzen und Zykluszeiten an steigenden und fallenden Signalfanken. Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Messverfahren zur Verfügung:

- „Cycle Time --/-- [µs]“
- „Cycle Time --\-- [µs]“
- „Frequency --/-- [Hz]“
- „Frequency --\-- [Hz]“

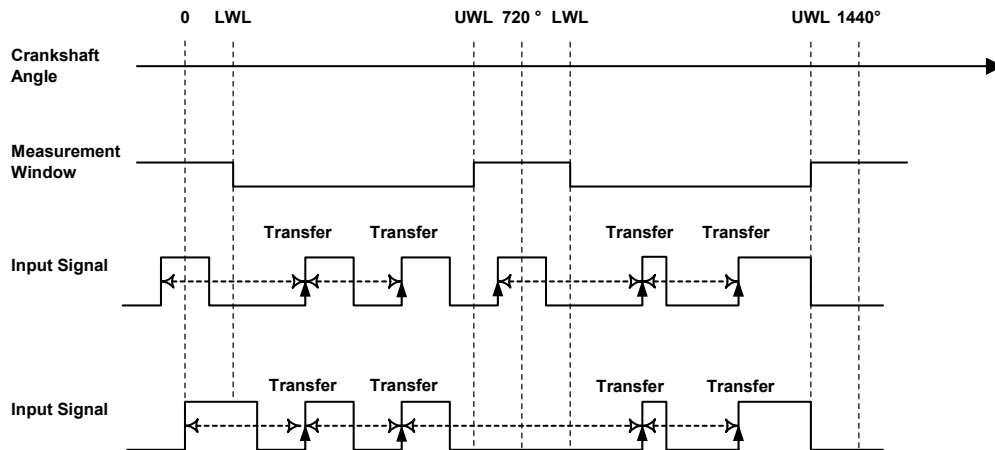
Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus
- Timeout-Überwachung

#### *Funktionsbeschreibung*

Im asynchronen Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalfanke. Im drehzahlsynchronen (winkelsegmentierten) Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalfanke, die innerhalb eines Win-

kelfensters liegt. Abb. 8-19 veranschaulicht die Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalfanken.



**Abb. 8-19** Darstellung der Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalfanken

**Hinweis**

Wird bei einer winkelsegmentierten Zykluszeit- oder Frequenzmessung der gesamte Winkelbereich von 720° CA freigeschaltet, so verhält sich diese Messung wie die entsprechende Messung im asynchronen Modus.

*Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-11 angegeben.

**Timeout-Überwachung**

**Messwert im Timeout-Fall**

„Intvl InpDep“  
 „CS Angle InpDep“  
 „Intvl & CS Angle InpDep“

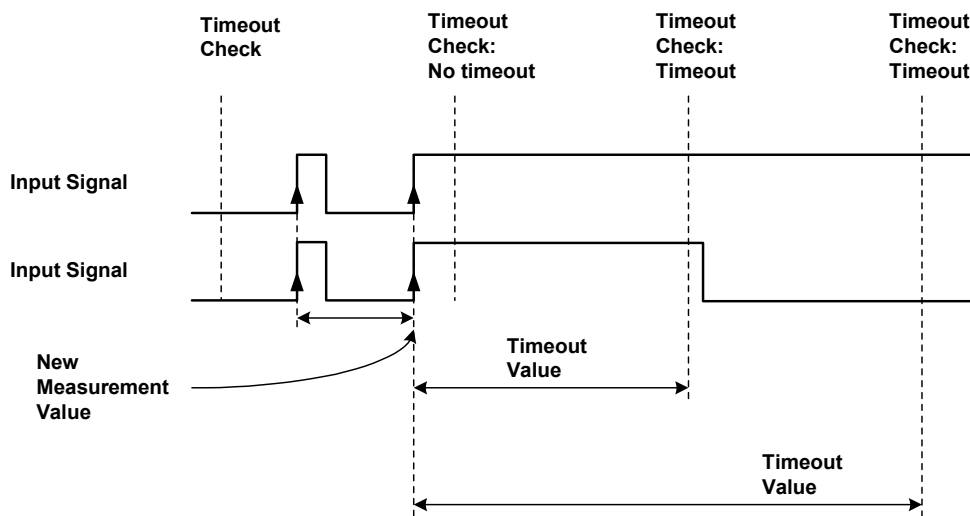
*Zykluszeitmessungen:*  
 Als Messwert wird im Timeout-Fall die Zeitdauer zwischen dem Zeitpunkt des Auftretens der letzten aktiven Flanke des Eingangssignals und dem Zeitpunkt, an dem die Überprüfung auf Timeout durchgeführt wird, ausgegeben (siehe Abb. 8-20).

*Frequenzmessungen:*  
 Die Messwertberechnung im Timeout-Fall erfolgt wie bei den Zykluszeitmessungen, jedoch wird aus der ermittelten Zeitdauer durch Kehrwertbildung eine resultierende Frequenz ermittelt.

„Intvl Predef“  
 „CS Angle Predef“  
 „Intvl & CS Angle Predef“

Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-11** Frequenz- und Zykluszeitmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung.



**Abb. 8-20** Frequenz- und Zykluszeitmessungen an ansteigenden Flanken: Darstellung der Messwertberechnung im Timeout-Fall (der Messwert ist in der Darstellung als „Timeout Value“ bezeichnet) bei den „... InpDep“-Verfahren zur Timeout-Überwachung

8.5.4 Tastverhältnismessungen

Tastverhältnisse (z.B. von PWM-Signalen) können mit der ES1334.1 auf unterschiedlichste Arten ermittelt werden. Bei den Tastverhältnismessungen wird unterschieden in Messungen, die die High-Phase eines Signals ins Verhältnis zur Low-Phase setzen (oder umgekehrt) und Messungen, die die High-Phase (oder die Low-Phase) eines Signals ins Verhältnis zur Zyklusdauer setzen.

Der erste Typ von Tastverhältnismessung wird im Folgenden als Tastverhältnis<sup>P/P</sup> bezeichnet. „P/P“ soll dabei andeuten, dass zwei Pulsdauern zueinander ins Verhältnis gesetzt werden.

Der zweite Typ von Tastverhältnismessung wird im Folgenden als Tastverhältnis<sup>P/C</sup> bezeichnet. „P/C“ soll andeuten, dass hier eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt wird.

$$\text{Tastverhältnis }^{P/P} : \frac{L}{H}, \frac{H}{L} \quad 0 \leq \text{Tastverhältnis }^{P/P} \leq \infty$$

$$\text{Tastverhältnis }^{P/C} : \frac{L}{L+H}, \frac{H}{L+H} \quad 0 \leq \text{Tastverhältnis }^{P/C} \leq 1$$

Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Tastverhältnismessungen vom Typ „P/P“

- „Duty Factor L/H --/--“
- „Duty Factor L/H --\--“
- „Duty Factor H/L --/--“
- „Duty Factor H/L --\--“

und Tastverhältnismessungen vom Typ „P/C“

- „Duty Cycle L/(L+H) --/--“

- „Duty Cycle L/(L+H) --\--“
- „Duty Cycle H/(L+H) --/--“
- „Duty Cycle H/(L+H) --\--“

zur Verfügung.

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

#### *Funktionsbeschreibung*

---

Was die Messwertberechnung im asynchronen und drehzahlsynchronen (winkelsegmentierten) Messmodus anbelangt, so gilt bei Tastverhältnismessungen exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 8.5.3 auf Seite 176 beschrieben wurde.

#### *Timeout-Überwachung*

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-12 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	<p>Tastverhältnis <math>P/P</math>: <math>\frac{L}{H}, \frac{H}{L}</math></p> <p>Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/H-Messungen oder „low“ bei H/L-Messungen ist. Der Messwert ist MAXREAL32 (<math>3.40282347 \times 10^{38}</math>), falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/H-Messungen oder „high“ bei H/L-Messungen ist.</p> <p>Tastverhältnis <math>P/C</math>: <math>\frac{L}{L+H}, \frac{H}{L+H}</math></p> <p>Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/(L+H)-Messungen oder „low“ bei H/(L+H)-Messungen ist. Der Messwert ist 1.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/(L+H)-Messungen oder „high“ bei H/(L+H)-Messungen ist.</p>
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-12** Tastverhältnismessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 8.5.5 Vermessung von Flanken: Winkelstempel

#### Funktionsbeschreibung

Die nachfolgenden Messverfahren vermessen den Kurbelwellenwinkel aktiver Signalflanken auf dem zugeordneten Hardwarekanal relativ zum eingestellten Referenzwinkel.

- „Rising Edge --/-- [deg]“
- „Falling Edge --\-- [deg]“

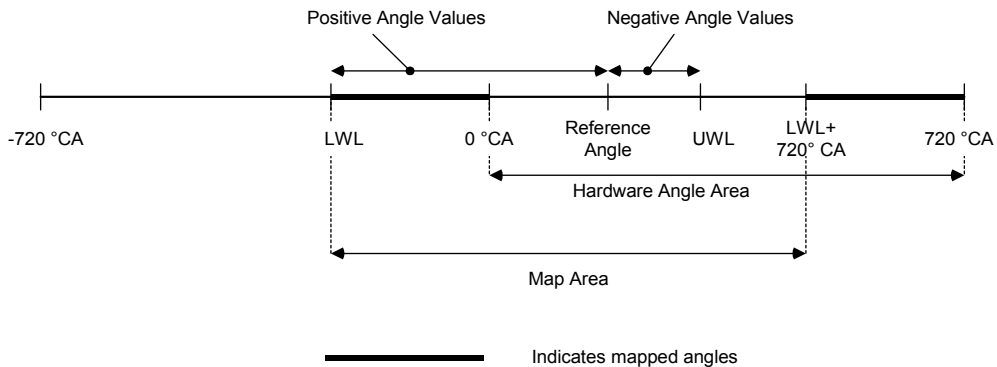
Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus
- „CS Angle Reference“: Definiert den Referenzwinkel der Messung
- Timeout-Überwachung



Der von der Hardware gelieferte Kurbelwellenwinkel einer Flanke liegt grundsätzlich im Bereich  $[0^\circ \text{ CA}, 720^\circ \text{ CA}]$ .

Im *drehzahlsynchronen Messmodus* wird der gemessene Winkel von der ES1334.1 in den Bereich  $[\text{LWL}, \text{LWL} + 720^\circ \text{ CA}]$  abgebildet und anschließend die Differenz mit dem Referenzwinkel berechnet. Diese Differenz ist der von der Messung gelieferte Messwert, sie ist positiv, wenn der abgebildete Winkel kleiner als der Referenzwinkel ist (vgl. Abb. 8-21).



**Abb. 8-21** Funktionsweise einer Winkelmessung im drehzahlsynchronen Messmodus

Da der Referenzwinkel im Bereich  $[-720^\circ \text{ CA}, 720^\circ \text{ CA}]$  liegen kann, liefern die Messfunktionen im drehzahlsynchronen Messmodus Winkelwerte aus dem Bereich  $[-1440^\circ \text{ CA}, 1440^\circ \text{ CA}]$ . Im *asynchronen Messmodus* findet keine Abbildung der von der Hardware gelieferten Kurbelwellenwinkel statt, so dass in diesem Messmodus die Winkelwerte im Bereich  $[-1440^\circ \text{ CA}, 720^\circ \text{ CA}]$  liegen.

Im *drehzahlsynchronen Messmodus* wird jede aktive Flanke, die innerhalb eines Winkelfensters liegt, gemessen. Im *asynchronen Messmodus* wird der Kurbelwellenwinkel jeder aktiven Flanke gemessen.

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-13 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der Low-Pegel, bei einer Messung an fallenden Flanken der High-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der High-Pegel, bei einer Messung an fallenden Flanken der Low-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-13** Winkelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 8.5.6 Pulszählung

---

In diesem Abschnitt werden die Eigenschaften folgender Messverfahren zur Pulszählung beschrieben:

- „Number of Low-Pulses“
- „Number of High-Pulses“

Zulässige Messoptionen sind:

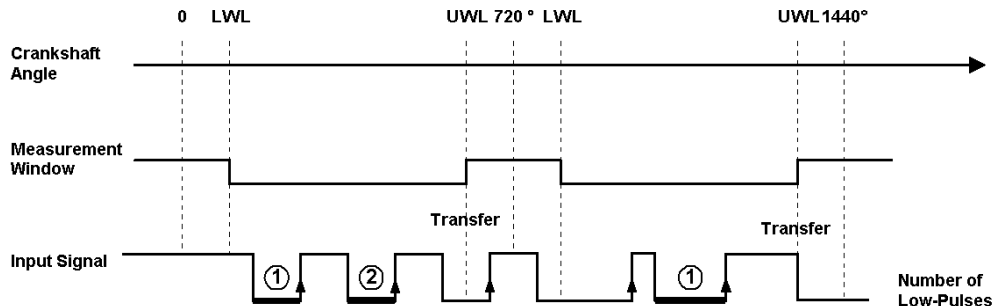
- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus
- Timeout-Überwachung

#### Funktionsbeschreibung

---

Die Messverfahren „Number of Low-Pulses“ und „Number of High-Pulses“ erfassen im *drehzahlsynchronen Messmodus* die Anzahl der gültigen aktiven Pulse innerhalb eines Winkelfensters. Gültige Pulse sind Pulse, die mit öffnender

und schließender Flanke in ein- und demselben Winkelfenster liegen. Die ermittelte Summe wird an der oberen Winkelfenstergrenze in das DPRAM übertragen (Abb. 8-22).



**Abb. 8-22** Darstellung der Funktionsweise drehzahlsynchroner Messverfahren zur Pulszählung (Messverfahren „Number of Low-Pulses“). Pulse, die gezählt werden, sind durch erhöhte Linienstärke hervorgehoben. Die Summe wird jeweils an der oberen Winkelfenstergrenze in das DPRAM transferiert.

Im *asynchronen Messmodus* wird die Gesamtzahl der aktiven Pulse seit der letzten Initialisierung oder Konfiguration der ES1334.1 gemessen. Die Summe wird an jeder schließenden Pulsflanke aktualisiert.

#### **Hinweis**

Die Pulszählung erfolgt modulo  $2^{52}$ , d.h. nach  $2^{52} - 1$  Pulsen springt die Summe wieder auf 0 um. ( $2^{52} - 1$ ) ist die größte ganze Zahl des Typs „double precision“, die mit einer Auflösung von 1,0 darstellbar ist.

#### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein gültiger aktiver Puls aufgetreten ist. Im drehzahlsynchronen Messmodus ist ein gültiger aktiver Puls ein aktiver Puls, dessen öffnende und schließende Flanke

innerhalb ein und desselben Winkelfensters liegen. Im asynchronen Messmodus ist jeder aktive Puls ein gültiger aktiver Puls. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-14 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Zählung von Low-Pulsen der High-Pegel und bei einer Zählung von High-Pulsen der Low-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Zählung von Low-Pulsen der Low-Pegel und bei einer Zählung von High-Pulsen der High-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-14** Pulszählung: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 8.5.7 Pegelmessung

#### *Funktionsbeschreibung*

Die Messverfahren

- „Level (Active High)“
- „Level (Active Low)“

dienen zur Erfassung der aktiven Pegelzustände auf einem Hardwarekanal. Ein aktiver Pegel wird durch den Messwert „Eins“ (1), ein inaktiver Pegel durch den Messwert „Null“ (0) signalisiert. Im *asynchronen Messmodus* wird der Messwert an jeder Flanke des Eingangssignals aktualisiert. Im *drehzahlsynchronen Messmodus* findet eine Aktualisierung nur an Signalfanken statt, die innerhalb des Winkelfensters auftreten.

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus
- Timeout-Überwachung

### *Timeout-Überwachung*

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 8-15 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert „0“ ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert „1“ ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 8-15** Pegelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung



## 9 ES1334.2 Measurement Board

### Begriffsdefinitionen

#### Aktive Signalflanke:

Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal:

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Pulsflanke:

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal:

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### LWL:

Steht für **L**ower **W**indow **L**imit und bezeichnet die untere Grenze des Winkelfensters bei winkelsynchronen Messungen.

#### Öffnende Pulsflanke:

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



#### UWL:

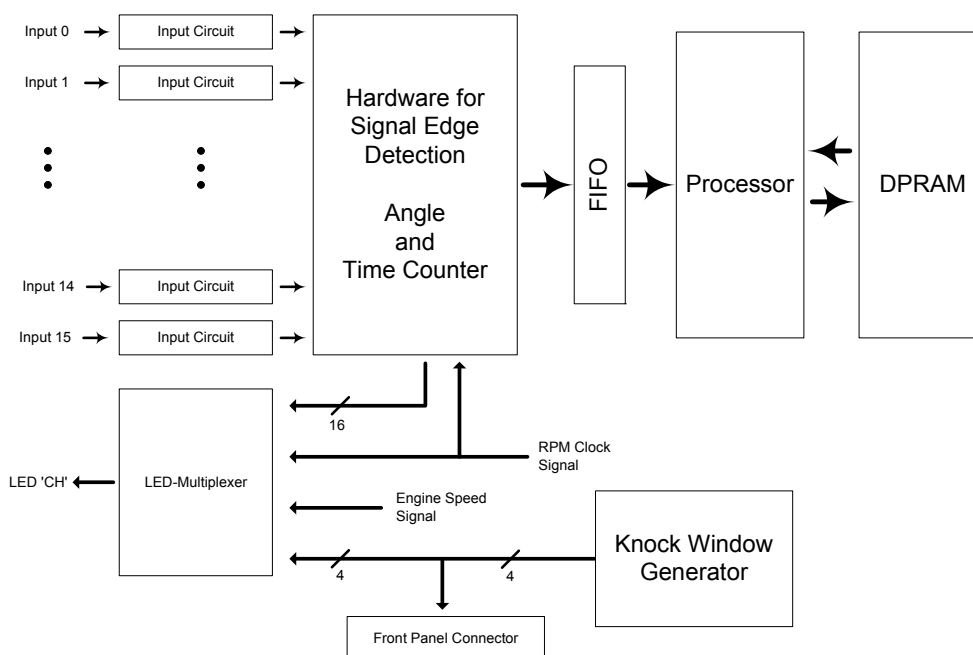
Steht für **U**pper **W**indow **L**imit und bezeichnet die obere Grenze des Winkelfensters bei winkelsynchronen Messungen.

## 9.1 Funktionsbeschreibung

### 9.1.1 Grundprinzip der Messwertbestimmung

Um eine möglichst große Flexibilität bei der Auswertung anliegender Eingangssignale zu erreichen, basiert die ES1334.2 auf folgendem Grundprinzip:

Zuerst wird jedes Eingangssignal separat (analog) aufbereitet und dann mit einem (per Software einstellbaren) Schwellwert verglichen, um die Umsetzung des analogen Eingangssignals in eine digitale 0/1- bzw. inaktiv/aktiv-Information zu realisieren. Die so erhaltene binäre Pegelinformation der Signale an den 16 Eingangskanälen wird ständig von einer Hardwareschaltung auf Änderungen hin überprüft, d.h. es wird eine Erkennung von „Flanken“ auf einem oder mehreren Eingangsbits durchgeführt. Wechselt nun zumindest ein Eingangssignal von aktiv auf inaktiv bzw. umgekehrt, so speichert die Schaltung die aktuellen Zustände der 16 Eingänge und ruft die momentanen Werte zweier integrierter Zähler ab, die die aktuelle Zeit und den aktuellen Kurbelwellenwinkel angeben. Diese Informationen werden hardware-gesteuert in einen FIFO-Puffer übertragen.



**Abb. 9-1** Blockschaltbild des ES1334.2 Measurement Board

Die Prozessor-Firmware der ES1334.2 kann nun die generierten Werte aus dem FIFO-Puffer auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben. Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind, wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

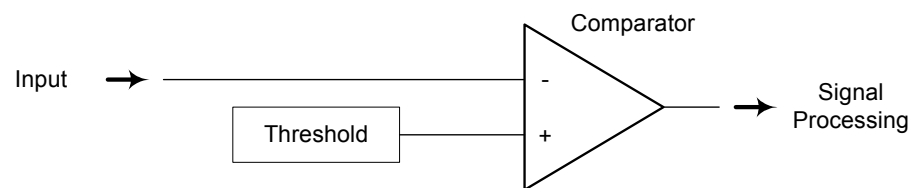
Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten software-technischen Aufbereitung der Messwerte bewirkt, dass die Auslastung des Prozessors direkt von der Zahl der erkannten Flanken (also den Frequenzen der Eingangssignale) abhängig ist. Der Vorteil dieses Konzepts: Die Hardware erfasst lediglich sämtliche Änderungen und versieht sie mit



Winkel- und Zeitstempeln. Die Auswertung erfolgt hingegen per Software, ist demzufolge verhältnismäßig leicht zu ändern bzw. anzupassen. Firmware und RTIO-Treiber bieten ein breites Spektrum an implementierten Messverfahren (vgl. Seite 213 ff.). Das eingesetzte Konzept ermöglicht die Implementierung weiterer Messverfahren. Für die Realisierung genügt eine Anpassung der -Firmware und des RTIO-Treibers.

### 9.1.2 Eingangsbeschaltung

Die in Abb. 9-1 auf Seite 188 angedeutete Eingangsbeschaltung eines Hardwarekanals der ES1334.2 ist in Abb. 9-2 vereinfacht dargestellt. Sie besteht im Wesentlichen aus einem Vergleich mit einem durch den Anwender definierten Schwellwert.



**Abb. 9-2** Vereinfachte Darstellung der Eingangsbeschaltung eines -Hardwarekanals

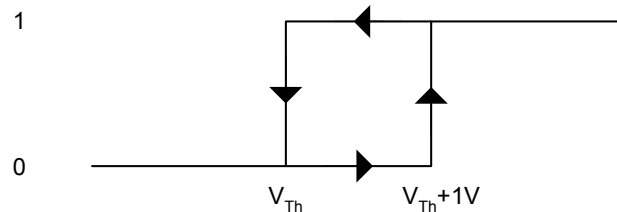
Durch diesen Vergleich wird die Umsetzung des analogen Eingangssignals in eine digitale 0/1-Information erreicht. Der Schwellwert ist derart definiert, dass sämtliche am Karteneingang anliegenden Spannungen, die größer als dieser Wert sind, als „1“ interpretiert werden. Alle Spannungen unterhalb dieses Schwellwertes werden als „0“ interpretiert.

Die Komparatorschwellen können für die Kanäle 0 bis 7 und 8 bis 15 jeweils gemeinsam auf einen der nachfolgenden Werte eingestellt werden:

- 2.5 V
- (Batteriespannung A)/2
- (Batteriespannung B)/2
- Externe Referenz-Spannung

Gerade im Kfz-Umfeld kommen sehr häufig verhältnismäßig langsame, also niederfrequente Signale zum Einsatz, die zudem an den Flanken sehr oft auch große Anstiegs- bzw. Abfallzeiten aufweisen. Werden nun solche Signale mit einem vorgegebenen Schwellwert verglichen, so besteht - insbesondere wenn das Signal noch ein gewisses Rauschen aufweist - die Gefahr, dass beim langsamen Über- bzw. Unterschreiten der Schwelle das Vergleichsergebnis einige Male zwischen „1“ und „0“ hin und her wechselt. Um diese unerwünschten Nebeneffekte zu vermeiden, weisen die Vergleichsschaltungen auf der ES1334.2 eine

Hysterese auf, d.h. der Schwellwert für den Wechsel von „0“ auf „1“ liegt etwas höher als der Schwellwert für den Wechsel von „1“ nach „0“. Die Größe der Hysterese beträgt 1 V (siehe Abb. 9-3).



**Abb. 9-3** Komparator-Hysterese.  $V_{Th}$  bezeichnet den eingestellten Schwellwert

### 9.1.3 Messwernerfassung

Der -RTIO-Treiber stellt dem Anwender bis zu 64 frei konfigurierbare Messwerte (im folgenden Software- bzw. Messkanäle genannt) zur Verfügung, wobei jeder einzelne Messwert von jedem beliebigen Hardwarekanal abhängen kann. Die Zuordnung von Hardwarekanälen auf der einen und Messkanal auf der anderen Seite ist somit nicht durch den -RTIO-Treiber vorgegeben, d.h. bei der Nutzung eines Laborautos für mehrere Projekte ist es nicht zwingend notwendig, bedeutungsgleiche Signale immer an denselben Hardwarekanälen der ES1334.2 anzuschließen.

Darüber hinaus werden dem Anwender mit den 64 Messkanälen viermal so viele Ausgangsgrößen zur Verfügung gestellt, wie die Karte Hardwarekanäle aufweist. Hierdurch ist es problemlos möglich, aus einem einzigen Eingangssignal mehrere Messwerte zu ermitteln, wie z.B. Frequenz und Tastverhältnis eines PWM-Signals. Eine doppelte Verdrahtung ein und desselben Signals auf verschiedene Kanäle der Karte entfällt.

Für jeden dieser Messkanäle kann (neben dem zugrundeliegenden Hardwarekanal) vollkommen unabhängig von allen anderen Messwerten definiert werden,

- wie das anliegende Signal auszuwerten ist,
- wann und wie es auf fehlende Pulse hin untersucht werden soll (Timeout-Überwachung) und
- ob die Auswertung dauernd stattfinden soll oder nur innerhalb eines bestimmten Bereiches des Kurbelwellenwinkels (Winkelsegmentierung).
- Für einige Messungen lässt sich weiter festlegen, der wievielte Puls vermessen werden soll.

### 9.1.4 Timeout-Überwachung

Wie bereits erläutert, arbeitet die ES1334.2 flankengesteuert, d.h. die Auswertung der anliegenden externen Signale findet ausschließlich an deren Flanken statt. Dies hat jedoch zur Folge, dass auch die generierten Messwerte, die der -RTIO-Treiber liefert, nur nach Flanken aktualisiert werden. Ein Signal, welches bereits einmal von der Karte ausgewertet wurde, würde folglich auch dann noch den zuletzt generierten gültigen Messwert liefern, wenn es inzwischen z.B. vollkommen ausgefallen ist, also keinerlei Flanken mehr aufweist.

Um Signale, die ganz oder zeitweise ausfallen, auch korrekt zu interpretieren, bietet der -RTIO-Treiber die Möglichkeit der Timeout-Überwachung. Jeder einzelne Messwert kann hierzu in bestimmten Abständen daraufhin getestet wer-

den, ob er durch neu aufgetretene Flanken am Eingangssignal aktualisiert wurde. Dieser Test kann sowohl in vom Anwender spezifizierbaren Zeitabständen als auch jeweils an vom Anwender vorgegebenen Kurbelwellenwinkeln erfolgen.

Wird während einer derartigen Überprüfung festgestellt, dass seit der letzten Überprüfung keinerlei Flanke von der Hardware detektiert wurde, so kann der entsprechende Messwert auf zwei verschiedene Arten modifiziert werden. Entweder wird er auf einen vorgegebenen Timeout-Wert gesetzt oder aber in Abhängigkeit vom aktuellen Pegel (high oder low) des Eingangssignals ermittelt. Dies ist insbesondere zur korrekten Auswertung von Tastverhältnissen erforderlich (ein inaktiver Eingang resultiert hierbei in den Messwerten 0.0 bzw. 1.0 - je nach Pegel des Eingangssignals).

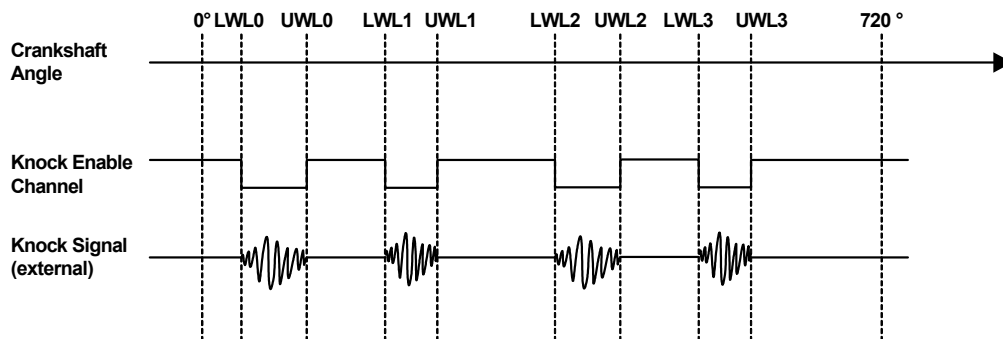
Insgesamt ergeben sich somit sechs Möglichkeiten der Timeout-Überwachung:

- Überprüfung in Zeitabständen von x ms: Messwert im Timeout-Fall vordefiniert auf Wert y.
- Überprüfung alle 720° bei x° Kurbelwelle: Messwert im Timeout-Fall vordefiniert auf Wert y.
- Überprüfung alle 720° bei x1° Kurbelwelle und alle x2 ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien erfüllt sind. Messwert im Timeout-Fall vordefiniert auf Wert y.
- Überprüfung in Zeitabständen von x ms: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle 720° bei x° Kurbelwelle: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle 720° bei x1° Kurbelwelle und alle x2 ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien erfüllt sind. Der Messwert im Timeout-Fall ist abhängig vom Pegel des Eingangssignals.

#### 9.1.5 Enable-Signale für Klopf-Generatoren

Die ES1334.2 bietet als zusätzliches Feature die Generierung von 4 digitalen Signalen mit TTL-Pegel zur Aktivierung bzw. Deaktivierung von Klopf-Generatoren. Bei diesen Klopf-Enable-Signalen handelt es sich um winkelsynchrone Signale, die innerhalb von Anwender definierbaren Winkelsegmenten (Klopfenster) aktiv geschaltet werden (Abb. 9-4). Die aktiven Signalpegel (aktiv-high oder aktiv-low)

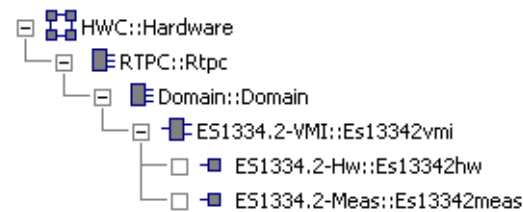
der Klopf-Enable-Signale sind ebenso konfigurierbar wie die Position und die Anzahl der Winkelsegmente pro 720 °KW. Bis zu 12 Winkelsegmente pro 720 °KW werden unterstützt.



**Abb. 9-4** Low-aktives Klopf-Enable-Signal zur Aktivierung und Deaktivierung eines externen Klopf-Generators

#### 9.1.6 Aufbau des -RTIO-Baums

Im RTIO-Editor wird das ES1334.2 Measurement Board durch Auswahl des ES1334.2-VMI Subsystems eingebunden.



**Abb. 9-5** RTIO-Hardwarebeschreibung mit eingebundener ES1334.2

Die ES1334.2 bietet 16 Hardwarekanäle zur Erfassung und Aufbereitung analoger und digitaler Signale sowie 4 Ausgangssignale zur Aktivierung und Deaktivierung von externen Klopfgeneratoren. Die Konfiguration der 16 Hardwarekanäle und der 4 Klopf-Enable-Ausgänge erfolgt mit dem ES1334.2-Hw Device, das dem ES1334.2-VMI Subsystem zugeordnet wird. Neben diesem Element lassen sich dem ES1334.2-VMI Subsystem bis zu zwei sogenannte ES1334.2-Meas Devices zuordnen. Mit jedem dieser Devices lassen sich bis zu 32 Messungen spezifizieren. Insgesamt bieten RTIO-Treiber und Firmware der ES1334.2 damit 64 Messungen bzw. Messkanäle.

#### **Hinweis**

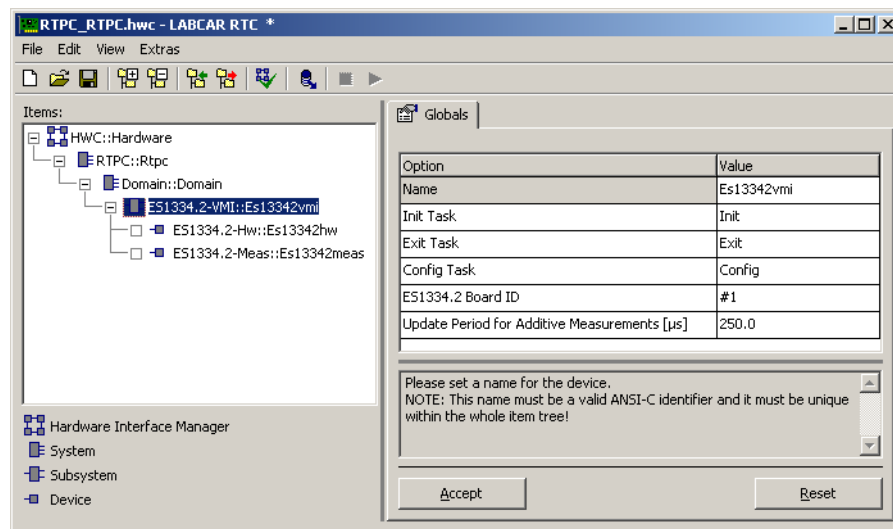
*Bevor ein ES1334.2-Meas Device eingefügt werden kann, muss zunächst ein ES1334.2-Hw Device zur Konfiguration der ES1334.2-Hardware eingefügt werden.*

## 9.2 ES1334.2-VMI Subsystem

### 9.2.1 Globals (ES1334.2-VMI Subsystem)

Das ES1334.2-VMI Subsystem dient zur Einstellung von RTIO-Parametern, die global wirksam sind, d.h. die auf alle ES1334.2-Meas Devices wirken.

Abb. 9-6 zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 9-6** Die Registerkarte „Globals“ des ES1334.2-VMI-Subsystems

#### Hinweis

Wenn das ES1334.2 Measurement Board in eine LABCAR-RTC-Hardwarekonfiguration eingebunden wird, müssen die beiden Adress-Schalter SW1 und SW2 auf dem Board die Einstellung „0x00“ haben!

#### ES1334.2 Board ID

Dieses Optionsfeld dient zur Identifizierung der ES1334.2. Es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1334.2 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist. Die Nummerierung der ES1334.2 Boards im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Numerierung) beginnend mit 1. Die ermittelte Nummer der ES1334.2 ist in dem Listenfeld „ES1334.2 Board ID“ einzustellen. Dieser RTIO-Parameter ist nicht online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar. Im RTIO-Editor können bis zu 20 ES1334.2 Boards pro Baugruppenträger eingebunden werden.

#### Update Period for Additive Measurements [µs]

Dieser Parameter ist nur für Software- bzw. Messkanäle von Bedeutung, die additive Zeitmessungen im asynchronen Modus durchführen. Er legt fest, in welchen Zeitabständen spätestens eine Aktualisierung der additiven Zeit stattfindet. Weitere Informationen enthält der Abschnitt „Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen“ auf Seite 217. Der Parameter ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Tab. 9-1 fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
ES1334.2 Board ID	uint32	Nein	Identifizierung der ES1334.2. Zur Auswahl stehen #1 bis #20.
Update Period for Additive Measurements [ $\mu$ s]	real32	Ja	Messwert-Aktualisierungsrate für asynchrone additive Messungen in Mikrosekunden. - Minimal: 50 $\mu$ s - Maximal: 500 ms

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

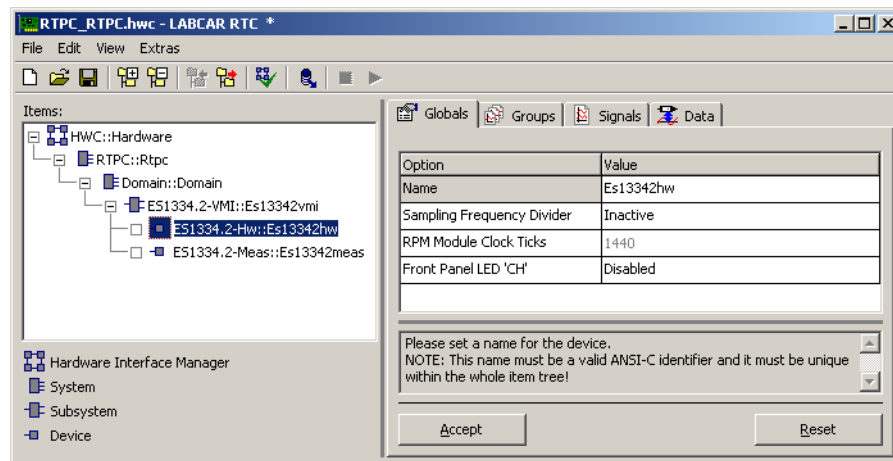
**Tab. 9-1** ES1334.2-VMI Subsystem: Konfigurationsparameter der Registerkarte „Globals“

### 9.3 ES1334.2-Hw Device

Das ES1334.2-Hw Device dient zur Konfiguration und Ansteuerung der ES1334.2-Hardware. Von den einzelnen Hardwarekanälen unabhängige Einstellungen werden in der Registerkarte „Globals“ dieses Elements durchgeführt. Kanalspezifische Einstellungen werden in der Registerkarte „Signals“ vorgenommen.

#### 9.3.1 Globals (ES1334.2-Hw Device)

Abb. 9-7 zeigt die RTIO-Parameter der Registerkarte „Globals“. Tab. 9-2 auf Seite 197 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 9-7** Die Registerkarte „Globals“ des ES1334.2-Hw Device

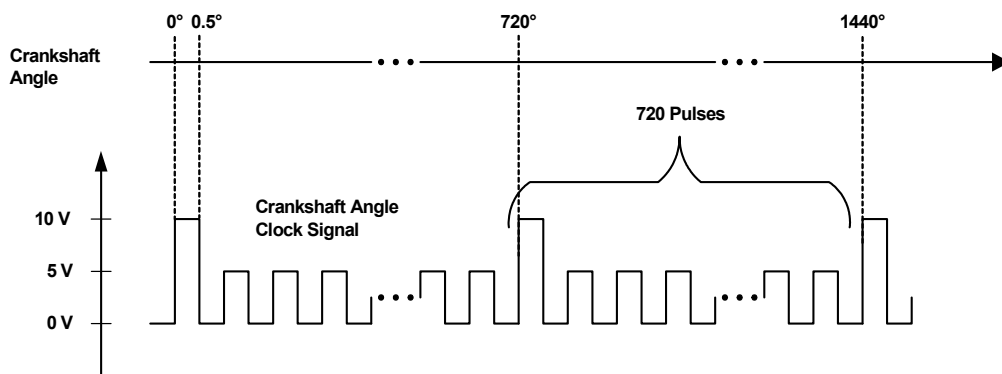
#### Sampling Frequency Divider

Die ES1334.2 tastet die Eingangssignale an den Hardwarekanälen periodisch ab. Die maximale Abtastrate ist durch die Frequenz des internen Taktgebers bestimmt, die 4 MHz beträgt. Mit einem Frequenzteiler kann die Abtastfrequenz reduziert werden. Die Konfiguration des Frequenzteilers erfolgt im Optionsfeld „Sampling Frequency Divider“ der Registerkarte „Globals“. Die Frequenzteilung kann deaktiviert oder auf die Teiler 2, 4, 8, 16 oder 32 konfiguriert werden. Es ist

zu beachten, dass nur Eingangspulse mit einer Dauer größer als die Abtastperiodendauer sicher erfasst werden. Der Parameter ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

#### *RPM Module Clock Ticks*

Zur Durchführung von zum Kurbelwellenwinkel synchronen Messungen benötigt die ES1334.2 Informationen über die Winkelstellung des Motors. Diese Informationen ermittelt die ES1334.2 aus dem Winkeltaktsignal, das ihr an der Frontplatte über den „SYNC“-Eingang zugeführt wird. Das Winkeltaktsignal (siehe Abb. 9-8) wird von einem ES1332.1 Arbitrary Signal Generator Board erzeugt und ermöglicht die Synchronisation auf den Nulldurchgang des Nockenwellenwinkels und die Berechnung der Winkelstellung des Motors.



**Abb. 9-8** Winkeltaktsignal zur Synchronisation auf den Nulldurchgang des Nockenwellenwinkels und Berechnung der Winkelstellung des Motors

Das numerische Eingabefeld „RPM Module Clock Ticks“ dient zur Definition der Winkelauflösung, d.h. der Anzahl der Winkeltakte pro Nockenwellenumdrehung (720 °KW). Diese Option ist fest auf den Wert 1440 eingestellt, da das von ES1332.1 Arbitrary Signal Generator Boards generierte Winkeltaktsignal diese Anzahl an Ticks bzw. Flanken erzeugt. Der eingestellte Wert von 1440 Winkeltakten pro Umdrehung der Nockenwelle entspricht einer Winkelauflösung von 0.5 °KW (°KW = Grad Kurbelwelle).

#### *Front Panel LED „CH“*

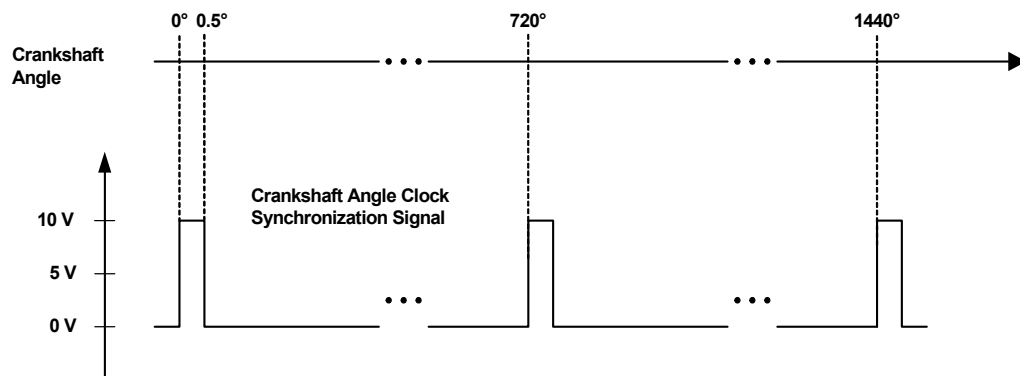
Auf der Frontplatte der ES1334.2 befindet sich eine LED mit der Bezeichnung „CH“. Mit dieser LED lässt sich der Signalpegel

- auf einem der 16 Hardwarekanäle oder
- auf einem der 4 Klopf-Enable-Ausgänge oder
- des Winkeltaktsignals (siehe Abb. 9-8) oder
- des Synchronisations-Signals auf den Nulldurchgang des Nockenwellenwinkels (siehe Abb. 9-9)

optisch anzeigen. Weist das Signal einen Low-Pegel auf, so wird die LED dunkel geschaltet. Weist das Signal einen High-Pegel auf, so wird die LED hell geschaltet. Die Definition von Low- und High-Pegel ist bei den 16 Eingangskanälen durch die Schwellspannung des Eingangskomparators des jeweiligen Kanals gegeben. Bei den Klopf-Enable-Ausgängen und dem Winkeltaktsignal ist die

Schaltswelle zur Unterscheidung von High- und Low-Pegel 2,5 V, beim Synchronisations-Signal auf den Nulldurchgang des Nockenwellenwinkels beträgt die Schaltswelle 7,5 V.

Die Registerkarte „Globals“ bietet für die LED „CH“ ein Optionsfeld „Front Panel LED 'CH'“, in dem das Signal eingestellt wird, das die LED ansteuert, bzw. mit dem die LED deaktiviert werden kann. Das Optionsfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 9-9** Synchronisationssignal auf den Nulldurchgang des Nockenwellenwinkels



Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Sampling Frequency Divider	uint8	Ja	Frequenzteiler für Eingangssignalabtastung 0: Keine Frequenzteilung (4 MHz Abtastung) 1: Teilung durch 2 (2 MHz) 2: Teilung durch 4 (1 MHz) 3: Teilung durch 8 (500 kHz) 4: Teilung durch 16 (250 kHz) 5: Teilung durch 32 (125 kHz)
RPM Module Clock Ticks	uint32	Nein	Winkelauflösung (fest auf 1440 eingestellt).
Front Panel LED „CH“	sint8	Ja	Signal bzw. Kanal der auf die Frontplatten LED „CH“ geschaltet ist bzw. Deaktivierung der LED. -1: Deaktivierung 0...15: Hardwarekanal #0... #15 16...19: Klopf-Enable-Ausgang #0...#3 20: Winkeltaktsignal (RPM Clock) 21: Synchronisationssignal auf den Nulldurchgang des Nockenwellenwinkels (Engine Speed)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 9-2** ES1334.2-Hw Device: Konfigurationsparameter der Registerkarte „Globals“

### 9.3.2 Groups (ES1334.2-Hw Device)

Das ES1334.2-Hw Device besitzt drei Signalgruppen (siehe Abb. 9-10 auf Seite 198).

Die Signalgruppe „Control“ wird vom Experimentaltarget zur ES1334.2 übertragen. Mit dieser Signalgruppe wird die Flankenerkennung auf den einzelnen Hardwarekanälen der Karte ein- und ausgeschaltet.

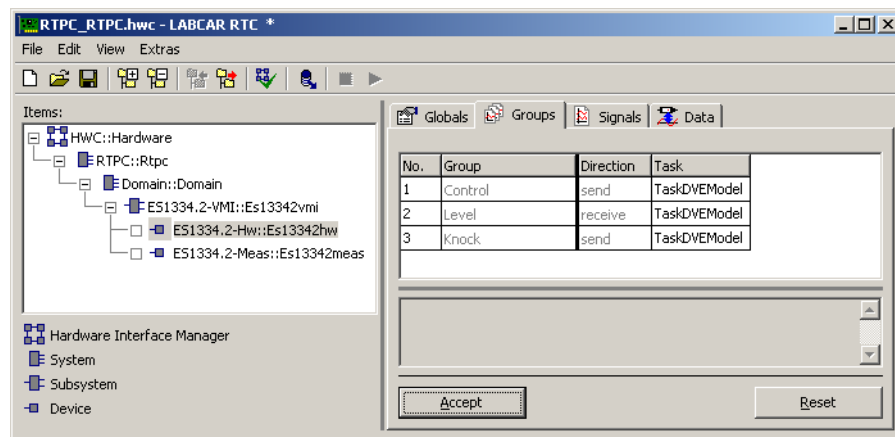
Die Signalgruppe „Level“ wird in umgekehrter Richtung von der ES1334.2 zum Experimentaltarget übertragen. Diese Signalgruppe enthält die aktuellen Pegelinformationen aller Hardwarekanäle, auf denen Messungen durchgeführt werden.

Die Signalgruppe „Knock“ wird vom Experimentaltarget zur ES1334.2 übertragen. Mit dieser Signalgruppe werden die Winkelfenster der Klopf-Enable-Ausgänge der Karte selektiv ein- und ausgeschaltet.

Den Signalgruppen sind ERCOS<sup>EK</sup>-Betriebssystem-Tasks zuzuordnen. Für die „Control“-Signalgruppe wird man üblicherweise eine Task mit dem Aktivierungstyp „Alarm“ und verhältnismäßig großer Aktivierungsperiode (z.B. 100 ms) wählen, da das Ein- und Ausschalten der Hardwarekanäle in der Regel kein hoch-

dynamischer Vorgang ist. Falls das Ein- und Ausschalten der Hardwarekanäle nur beim Modellstart bzw. Modellstop erfolgen soll, so ist es ausreichend die „Control“-Signalgruppe der „Init“ und der „Exit“ Task des Modells zuzuordnen.

Falls man die Pegelinformationen im Simulationsmodell auswertet, wird man der Signalgruppe „Level“ eine Task mit Aktivierungstyp „Alarm“ zuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik bzw. der Periodendauer der zu erfassenden Signale.



**Abb. 9-10** Die Registerkarte „Groups“ des ES1334.2-Hw-Device  
RTIO-Signale der Signalgruppe „Control“

Die Signalgruppe „Control“ umfasst 16 RTIO-Signale, mit denen die Flanken-erkennung auf den 16 Hardwarekanälen aktiviert bzw. deaktiviert werden kann. Tab. 9-3 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
ChnEnable_0	bool	Ein-/Ausschalten der Hardwarekanäle 0 bis 15
...		0: Deaktivierung der Flankenerkennung
ChnEnable_15		1: Aktivierung der Flankenerkennung

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 9-3** ES1334.2-Hw Device: Die RTIO-Signale der Signalgruppe „Control“

#### Hinweis

Die „Control“-Signalgruppe wird nur dann in das DPRAM der ES1334.2 übertragen, wenn sich eines ihrer RTIO-Signale im Wert verändert hat.

### *RTIO-Signale der Signalgruppe „Level“*

Die Signalgruppe „Level“ umfasst ein RTIO-Signal „LvlBitField\_0“, das als Bitfeld zu interpretieren ist. Das Bitfeld umfasst 16 Bit, in ihm sind die Pegelinformationen der Hardwarekanäle 0 bis 15 codiert. Tab. 9-4 fasst die Eigenschaften des RTIO-Signals zusammen.

#### **Hinweis**

*Nur die Pegelbits von Hardwarekanälen, bei denen die Flankenerkennung aktiviert ist, sind im Bitfeld „LvlBitField\_0“ gültig. Zur Aktivierung und Deaktivierung der Flankenerkennung auf einem Hardwarekanal siehe Abschnitt „RTIO-Signale der Signalgruppe „Control““ auf Seite 198.*

<b>RTIO-Signal</b>	<b>Datentyp*</b>	<b>Bemerkung / Wertebereich</b>
LvlBitField_0	uint16	Bitfeld mit Pegelinformationen der Hardwarekanäle 0 bis 15. Kanal 0: LSB (Least Significant Bit) Kanal 15: MSB (Most Significant Bit) Bitwert 0: Low Pegel Bitwert 1: High Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 9-4** ES1334.2-Hw Device: Die RTIO-Signale der Signalgruppe „Level“

### *RTIO-Signale der Signalgruppe „Knock“*

Die ES1334.2 bietet 4 Klopf-Enable-Ausgänge. Für jeden dieser Ausgänge können bis zu 12 Winkelfenster definiert werden, innerhalb derer der jeweilige Ausgang aktiv geschaltet wird. Die Signalgruppe „Knock“ umfasst 4 RTIO-Signale (KnckEnable\_0 bis KnckEnable\_3), mit denen die Winkelfenster der 4 Klopf-Enable-Ausgänge der Karte selektiv ein- und ausgeschaltet werden können. Die RTIO-Signale sind als Bitfelder zu interpretieren: Das niederwertigste Bit (Bit 0) aktiviert bzw. deaktiviert Winkelfenster 0, Bit 1 aktiviert bzw. deaktiviert Winkelfenster 1 usw.

Tab. 9-5 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
KnckEnable_0	uint16	Ein-/Ausschalten der Klopf-Enable-Ausgänge 0
...		- 3
KnckEnable_3		Bitfelder:
		Winkelfenster 0: Bit 0 (LSB)
		Winkelfenster 1: Bit 1
		...
		Winkelfenster 11: Bit 11
		Bitwert 0: Deaktivierung/Ausschalten
		Bitwert 1: Aktivierung/Einschalten

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 9-5** ES1334.2-Hw Device: Die RTIO-Signale der Signalgruppe „Knock“

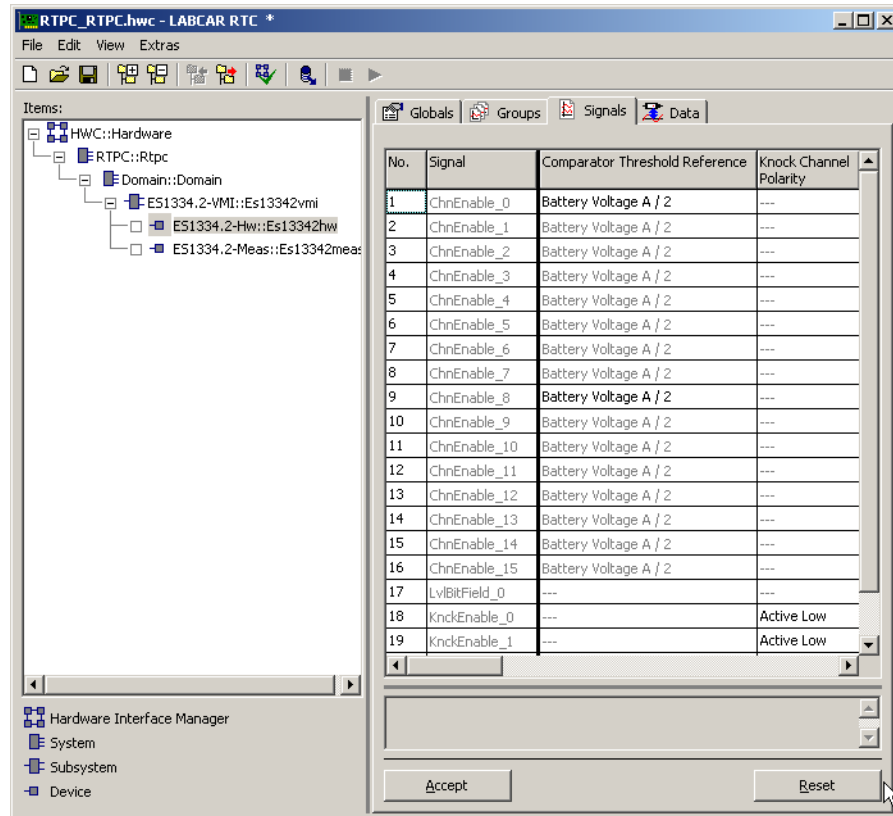
#### Hinweis

*Die „Knock“-Signalgruppe wird nur dann in das DPRAM der ES1334.2 übertragen, wenn sich eines ihrer RTIO-Signale im Wert verändert hat.*

### 9.3.3 Signals (ES1334.2-Hw Device)

In der Registerkarte „Signals“ wird die Konfiguration der 16 Hardwareeingangskanäle und der 4 Klopf-Enable-Ausgänge einer ES1334.2 durchgeführt. Abb. 9-11 zeigt die RTIO-Parameter der Registerkarte „Signals“.

Tab. 9-6 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 9-11** Die Registerkarte „Signals“ des ES1334.2-Hw Device

#### *Comparator Threshold Reference*

Das Optionsfeld „Comparator Threshold Reference“ dient zur Festlegung der Schwellspannung eines Eingangskomparators. Die Komparatorschwellen können für die Kanäle 0 bis 7 und 8 bis 15 nur jeweils gemeinsam auf einen der nachfolgenden Werte eingestellt werden:

- 2.5 V
- (Batteriespannung A)/2
- (Batteriespannung B)/2
- Externe Referenz-Spannung

#### *Knock Channel Polarity*

Polarität des Klopf-Enable-Ausgangs (aktiv-low oder aktiv-high).

# of Knock Windows

Anzahl der Winkelfenster des Klopf-Enable-Ausgangs. Es können bis zu 12 Winkelfenster definiert werden.

*LWL#x (x=0, 1, ... 11)*

Untere Grenze des jeweiligen Winkelfensters in Grad Kurbelwelle. LWL steht für Lower Window Limit.

*UWL#x (x=0, 1, ... 11)*

Obere Grenze des jeweiligen Winkelfensters in Grad Kurbelwelle. UWL steht für Upper Window Limit.

Bei der Definition der Winkelfenster ist zu beachten, dass sich die Winkelfenster eines Klopf-Enable-Ausgangs nicht überlappen dürfen. Die Fenstergrenzen müssen im Bereich [-720.0 °KW, +720 ° KW] liegen und die Größe eines Fensters muss kleiner als 720 °KW sein.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Comparator Threshold Reference	uint8	Ja	Komparatorreferenz 0: 2.5 V 1: (Battery Voltage A)/2 2: (Battery Voltage B)/2 3: External Reference Voltage
Knock Channel Polarity	uint8	Ja	Polarität des Klopf-Enable-Signals 0: Active Low 1: Active High
# of Knock Windows	uint8	Ja	Anzahl der Winkelfenster des Klopf-Enable-Ausgangs (1, 2, ... 12)
LWL#x (x=0, 1, ... 11)	real32	Ja	Untere Grenze des jeweiligen Winkelfensters in Grad Kurbelwelle. Wertebereich: -720 °KW bis 720 °KW
UWL#x (x=0, 1, ... 11)	real32	Ja	Obere Grenze des Winkelfensters in Grad Kurbelwelle. Wertebereich: -720 °KW bis 720 °KW

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

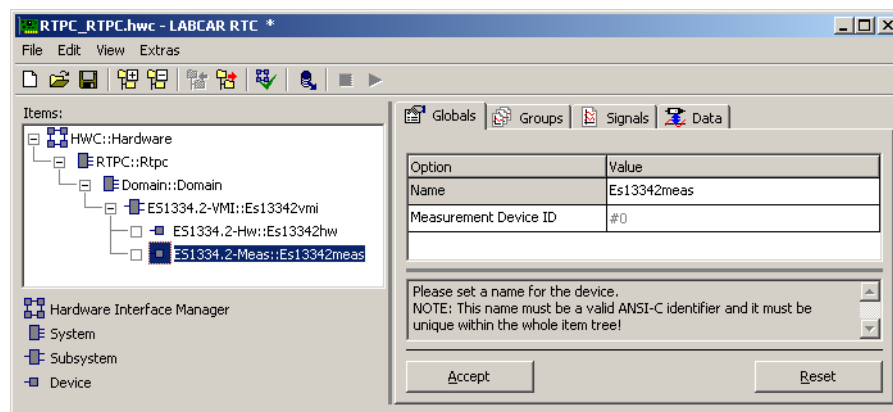
**Tab. 9-6** ES1334.2-Hw Device: Konfigurationsparameter der Registerkarte „Signals“

## 9.4 Messkonfiguration - ES1334.2-Meas Device

Das ES1334.2-Meas Device dient zur Spezifikation und Konfiguration von Messungen. Jedes ES1334.2-Meas Device bietet 32 Softwarekanäle (im Folgenden Messkanäle genannt), die frei auf die 16 Hardwareeingangskanäle der ES1334.2 verschaltet werden können. Bis zu zwei ES1334.2-Meas Devices werden pro ES1334.2 unterstützt, so dass insgesamt 64 Messkanäle konfiguriert werden können.

### 9.4.1 Globals (ES1334.2-Meas Device)

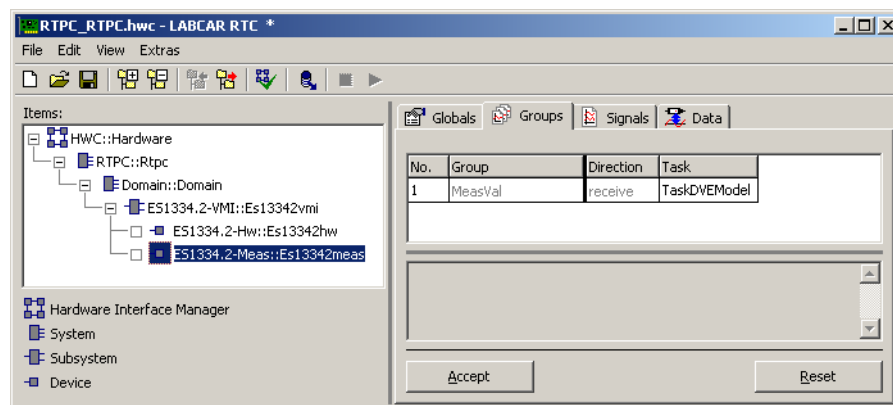
In der Registerkarte „Globals“ (Abb. 9-12) sind keine Einstellungen vorzunehmen. Im Optionsfeld „Measurement Device ID“ wird dem ES1334.2-Meas Device vom System eine Nummer zur Identifizierung zugewiesen.



**Abb. 9-12** Die Registerkarte „Globals“ des ES1334.2-Meas Device

### 9.4.2 Groups (ES1334.2-Meas Device)

Das ES1334.2-Meas Device besitzt eine Signalgruppe (Abb. 9-13), die von der ES1334.2 zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen. Dieser Signalgruppe ist eine ERCOS<sup>EK</sup>-Betriebssystem-Task mit dem Aktivierungstyp „Alarm“ zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.



**Abb. 9-13** Die Registerkarte „Groups“ des ES1334.2-Meas Device

### 9.4.3 RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 65 RTIO-Signale. Das „TriggerBitField\_0“-Signal ist als Bitfeld zu interpretieren. Es umfasst 32 Bit. In diesem Bitfeld sind die Trigger- oder Update-Daten der 32 Messkanäle codiert, d.h., es zeigt an, welche Messwerte seit der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der Messwert des zugehörigen Messkanals neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeouts oder aufgrund einer regulären Messwertberechnung erfolgte, in beiden Fällen wird das Aktualisierungs-Bit des Messkanals gesetzt.

Die RTIO-Signale „MeasVal\_0“ bis „MeasVal\_31“ enthalten die Messwerte der 32 Messkanäle. Es handelt sich dabei um regulär ermittelte Messwerte oder im Falle eines Timeout um den für diesen Fall vorgesehenen Timeout-Wert. Falls ein Messkanal nicht verwendet wird, wird dem zugehörigen Messwert -8888.0 zugewiesen. Die physikalische Einheit der Messwerte richtet sich nach dem Messverfahren:

- Zeitmessungen (Zeitstempel, (additive) Pulsdauer-Messungen, Periodendauern) werden in Mikrosekunden angegeben.
- Frequenzmessungen erfolgen in Hertz.
- Winkelmessungen und Winkelstempel werden in Grad Kurbelwelle (°KW) angegeben.
- Alle anderen Messungen (Tastverhältnisse, Pulszählung, Schrittmotormessungen, Pegel-messungen) sind dimensionslos.

Die RTIO-Signale „Tout\_0“ bis „Tout\_31“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messkanal.



Tab. 9-7 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
TriggerBitField_0	uint32	Bitfeld mit Update-Informationen der 32 Messkanäle. Messkanal 0: LSB Messkanal 31: MSB Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasVal_0 ... MeasVal_31	real64	Messwert Falls der Messkanal nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in Mikrosekunden - Frequenzmessungen erfolgen in Hertz - Winkelmessungen erfolgen in °KW
Tout_0 ... Tout_31	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout aufgetreten 1: Timeout aufgetreten 2: Timeout-Überwachung ist inaktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

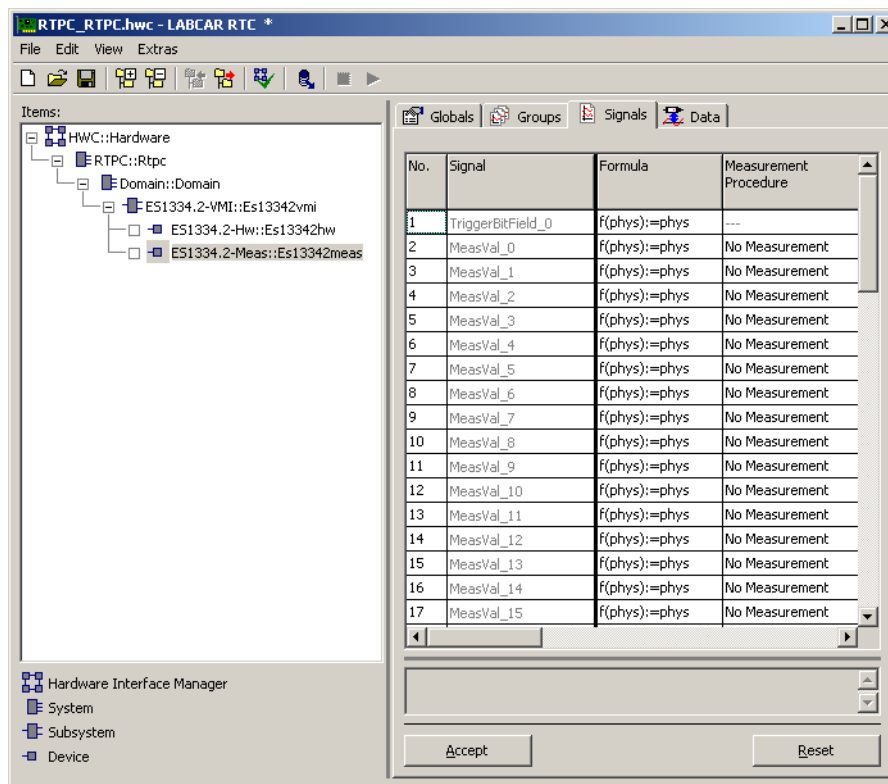
**Tab. 9-7** ES1334.2-Meas Device: Die RTIO-Signale der Signalgruppe „Meas-Val“

#### 9.4.4 Signals (ES1334.2-Meas Device)

In der Registerkarte „Signals“ wird die Konfiguration der 32 Messkanäle eines ES1334.2-Meas Device durchgeführt.

Abb. 9-14 zeigt die RTIO-Parameter der Registerkarte „Signals“. Tab. 9-8 auf Seite 212 fasst die Eigenschaften der einzelnen Parameter zusammen.

Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 9-14** Die Registerkarte „Signals“ des ES1334.2-Meas Device

#### *Measurement Procedure*

In dieser Listbox wird das Messverfahren ausgewählt. Ab Seite 213 finden Sie detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren. In der Listbox bedeutet die Notation --/-- eine Auswertung an steigenden Flanken, die Notation --\-- hingegen eine Auswertung an fallenden Flanken.

Wird in der Listbox die Option „No Measurement“ eingestellt, so wird keine Messung durchgeführt.

#### **Hinweis**

*Zur Vermeidung unnötiger Rechenzeit sind nicht benötigte Messkanäle zu deaktivieren.*

#### *Hardware Channel*

In dieser Listbox wird der Hardwarekanal festgelegt auf dem die Messung durchgeführt werden soll.

#### *Measurement Mode*

Das Optionsfeld „Measurement Mode“ definiert den Modus, in dem eine Messung durchgeführt wird. Es stehen zwei Optionen zur Auswahl:

- ESpeedSync  
Durchführung von drehzahlsynchronen bzw. winkelsynchronen Messungen. Zur Durchführung von winkelsynchronen Messungen ist es erforderlich, dass die ES1334.2 über den „SYNC“-Eingang mit einem Winkeltaktsignal versorgt wird.
- Asynchronous  
Durchführung von asynchronen Messungen

#### *Reference Channel*

---

Einige Messverfahren benötigen zur Durchführung einen zweiten Hardwarekanal oder einen zweiten Messkanal. Die Spezifikation dieses Referenzkanals für die Messung erfolgt in der Listbox „Reference Channel“.

Bei den in Abschnitt 9.5.3 auf Seite 219 beschriebenen Pulsweitenmessungen mit Enable- oder Validierungssignal ist der Referenzkanal ein Hardwarekanal, nämlich der Kanal, an dem das Enable- oder Validierungssignal anliegt.

Bei den in Abschnitt 9.5.6 auf Seite 226 beschriebenen Relativmessungen zwischen Hardwarekanälen und bei den in Abschnitt 9.5.11 auf Seite 237 beschriebenen Messungen zur Positionsverfolgung an zwei-phasigen Schrittmotoren ist der Referenzkanal ein Messkanal.

Bei allen anderen Messverfahren hat der in der Listbox „Reference Channel“ eingestellte Wert keine Bedeutung.

#### *(Max) Pulse (Count)*

---

Die Bedeutung dieses Optionsfelds ist nicht eindeutig, sie hängt vom jeweiligen Messverfahren ab. Bei puls- und flankenselektiven Messungen wird in diesem Optionsfeld die Nummer des Pulses oder der Flanke angegeben, die vermessen werden sollen. Bei additiven Messungen und Pulszählungen wird in diesem Optionsfeld die Maximalzahl von zu berücksichtigenden Pulsen angegeben.

#### *Timeout Monitoring*

---

Definition der Timeout-Überwachung für den zugehörigen Messkanal (siehe auch Abschnitt „Timeout-Überwachung“ auf Seite 190). Die folgenden Einstellungen sind möglich:

- Inactive  
Keine Timeout-Überwachung.
- Intvl Predef  
Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- Intvl InpDep  
Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.

- **CS Angle Predef**  
Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- **CS Angle InpDep**  
Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.
- **Intvl & CS Angle Predef**  
Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel und Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Ein Timeout wird erkannt, wenn eine oder beide Timeout-Kriterien verletzt sind. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- **Intvl & CS Angle InpDep**  
Wie „Intvl & CS Angle Predef“ jedoch ist der Messwert im Timeout-Fall vom Pegel des Eingangssignals abhängig.

#### *Default Timeout Value*

---

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls einer der „Intvl Predef“, „CS Angle Predef“ oder „Intvl & CS Angle Predef“ Timeout-Überwachungsmodi eingestellt ist.

#### *Timeout Interval [ms]*

---

Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Nur relevant für die zeitbasierten Timeout-Überwachungsmodi („Intvl Predef“, „Intvl InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Timeout Check Point*

---

Kurbelwellenwinkel an dem Timeout-Kontrollen durchgeführt werden. Nur relevant für die winkel-basierten Timeout-Überwachungsmodi („CS Angle Predef“, „CS Angle InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Lower Limit*

---

Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im winkelsynchronen Messmodus „ESpeedSync“.

#### *CS Angle Upper Limit*

---

Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im winkelsynchronen Messmodus „ESpeedSync“. Die obere Winkelfenstergrenze wird zudem bei additiven Zeitmessungen und Pulszählungen zum Transfer des ermittelten Messwertes in das DPRAM der ES1334.2 genutzt.

### *CS Angle Reference*

---

Winkelbezugspunkt in Grad Kurbelwelle, auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Measurement Procedure	sint32	Ja	<p>Auswahl des Messverfahrens oder Deaktivierung des Messkanals.</p> <p>(„Messverfahren“                      Parameterwert)</p> <p>Deaktivierung des Messkanals                      -1</p> <p>„Hightime (Pulse Qual.) [µs]“                      0</p> <p>„Lowtime (Pulse Qual.) [µs]“                      1</p> <p>„Hightime (Edge Qual.) [µs]“                      59</p> <p>„Lowtime (Edge Qual.) [µs]“                      60</p> <p>„H-Time n-th Pulse (Pu Qual.) [µs]“                      28</p> <p>„L-Time n-th Pulse (Pu Qual.) [µs]“                      29</p> <p>„H-Time n-th Pulse (Eg Qual.) [µs]“                      61</p> <p>„L-Time n-th Pulse (Eg Qual.) [µs]“                      62</p> <p>„Hightime using H-Enable [µs]“                      20</p> <p>„Lowtime using H-Enable [µs]“                      24</p> <p>„Hightime using L-Enable [µs]“                      21</p> <p>„Lowtime using L-Enable [µs]“                      25</p> <p>„Hightime using H-Validate [µs]“                      22</p> <p>„Lowtime using H-Validate [µs]“                      26</p> <p>„Hightime using L-Validate [µs]“                      23</p> <p>„Lowtime using L-Validate [µs]“                      27</p> <p>„H-Time n-th Pulse (H-Ena) [µs]“                      32</p> <p>„L-Time n-th Pulse (H-Ena) [µs]“                      36</p> <p>„H-Time n-th Pulse (L-Ena) [µs]“                      33</p> <p>„L-Time n-th Pulse (L-Ena) [µs]“                      37</p> <p>„H-Time n-th Pulse (H-Val) [µs]“                      34</p> <p>„L-Time n-th Pulse (H-Val) [µs]“                      38</p> <p>„H-Time n-th Pulse (L-Val) [µs]“                      35</p> <p>„L-Time n-th Pulse (L-Val) [µs]“                      39</p> <p>„Time from H-Ena to n-th edge [µs]“                      46</p> <p>„Time from L-Ena to n-th edge [µs]“                      45</p> <p>„Time from Ena to n-th --/-- [µs]“                      43</p> <p>„Time from Ena to n-th --\-- [µs]“                      44</p>

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
			„Additive Hightime [µs]“ 2
			„Additive Lowtime [µs]“ 3
			„Cycle Time --/-- [µs]“ 18
			„Cycle Time --\-- [µs]“ 19
			„Frequency --/-- [Hz]“ 4
			„Frequency --\-- [Hz]“ 5
			„Duty Factor L/H --/--“ 6
			„Duty Factor L/H --\--“ 7
			„Duty Factor H/L --/--“ 8
			„Duty Factor H/L --\--“ 9
			„Duty Cycle L/(L+H) --/--“ 10
			„Duty Cycle L/(L+H) --\--“ 11
			„Duty Cycle H/(L+H) --/--“ 12
			„Duty Cycle H/(L+H) --\--“ 13
			„Rising Edge --/-- [deg]“ 14
			„Falling Edge --\-- [deg]“ 15
			„Rising Edge of n-th Pulse [deg]“ 30
			„Falling Edge of n-th Pulse [deg]“ 31
			„Time Stamp --/-- [µs]“ 47
			„Time Stamp --\-- [µs]“ 48
			„Number of Low-Pulses“ 16
			„Number of High-Pulses“ 17
			„Total Number of L-Pulses“ 40
			„Total Number of H-Pulses“ 41
			„Level (Active High)“ 49
			„Level (Active Low)“ 50
			„Step Count (Step. Mot. Phase A)“ 42
			„Step Count (Step. Mot. Phase B)“ 63
Hardware Channel	sint8	Ja	Hardwarekanal, auf dem die Messung durchgeführt wird. 0 ... 15 Kanalnummer
Measurement Mode	uint8	Ja	Messmodus 0: „Asynchronous“: Asynchrone Messungen 1: „ESpeedSync“: Drehzahl- bzw. winkelsynchrone Messungen
Reference Channel	sint8	Ja	Referenzkanal der Messung -1: Kein Referenzkanal 0 ... 15: Hardwarekanal 0 bis 15 16 ... 47: Messkanal 0 bis 31

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
(Max) Pulse (Count)	uint32	Ja	Bei puls- und flankenselektiven Messungen: Die Nummer des Pulses bzw. der Flanke, die vermessen werden soll. Bei additiven Messungen und Pulszählungen: Die Maximalzahl von zu berücksichtigenden Pulsen. Wertebereich: $\geq 0$
Timeout Monitoring	uint8	Ja	Timeout-Überwachung: 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“ 3: „CS Angle Predef“ 4: „CS Angle InpDep“ 5: „Intvl & CS Angle Predef“ 6: „Intvl & CS Angle InpDep“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant in den „... Predef“-Modi zur Timeout-Überwachung.
Timeout Interval [ms]	real32	Ja	Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Wertebereich: $\geq 1.0$ ms Nur relevant in den „Intvl ...“-Modi zur Timeout-Überwachung
CS Angle Timeout Check Point	real32	Ja	Kurbelwellenwinkel an dem Timeout-Kontrollen durchgeführt werden. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant in den „CS Angle ...“-Modi zur Timeout-Überwachung.
CS Angle Lower Limit	real32	Ja	Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Upper Limit	real32	Ja	Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Reference	real32	Ja	Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung. Wertebereich: $-720$ °KW bis $720$ °KW.
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 9-8** ES1334.2-Meas Device: Konfigurationsparameter der Registerkarte „Signals“



## 9.5 Messverfahren

---

### 9.5.1 Pulsweitenmessungen

---

In diesem Abschnitt werden die Eigenschaften von Pulsweitenmessungen beschrieben. Es wird dabei eine Unterscheidung in „puls-qualifizierte“ und „flanken-qualifizierte“ Messungen vorgenommen. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

#### *Puls-qualifizierte Pulsweitenmessungen*

---

- „Hightime (Pulse Qual.) [µs]“
- „Lowtime (Pulse Qual.) [µs]“
- „H-Time n-th Pulse (Pu Qual.) [µs]“
- „L-Time n-th Pulse (Pu Qual.) [µs]“

#### *Flanken-qualifizierte Pulsweitenmessungen*

---

- „Hightime (Edge Qual.) [µs]“
- „Lowtime (Edge Qual.) [µs]“
- „H-Time n-th Pulse (Eg Qual.) [µs]“
- „L-Time n-th Pulse (Eg Qual.) [µs]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Puls-Selektion (nur bei „H-Time n-th Pulse ...“ und „L-Time n-th Pulse ...“): „(Max) Pulse (Count)“ definiert die Nummer des zu vermessenden Pulses.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung

#### *Funktionsbeschreibung: Winkelsynchrone Pulsweitenmessungen*

---

Bei den **puls-qualifizierten Messungen** werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen (gültiger Puls). Abb. 9-15 auf Seite 215 veranschaulicht die Funktionsweise einer puls-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [µs]“).

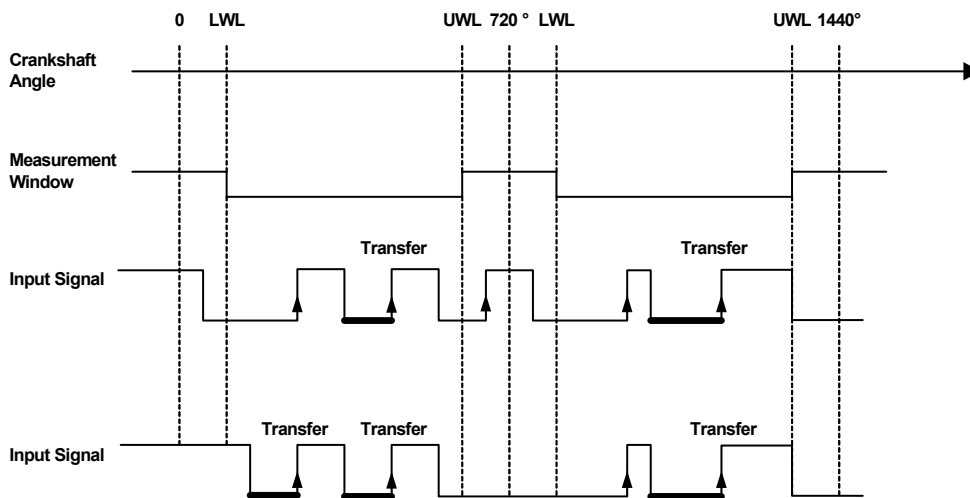
Bei den **flanken-qualifizierten Messungen** werden solche Pulse vermessen, bei denen die öffnende oder die schließende Flanke innerhalb eines (beliebigen) Winkelfensters liegen (gültiger Puls). Pulse, bei denen beide Flanken außerhalb eines Winkelfensters liegen, werden nicht vermessen bzw. berücksichtigt. Abb. 9-16 veranschaulicht die Funktionsweise einer flanken-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung („Lowtime (Edge Qual.) [µs]“).

Bei den **nicht puls-selektiven Messungen** („Hightime (Pulse Qual.) [µs]“, „Lowtime (Pulse Qual.) [µs]“, „Hightime (Edge Qual.) [µs]“, „Lowtime (Edge Qual.) [µs]“) wird jeder gültige Puls vermessen. Der Transfer der gemessenen Pulsweite in das DPRAM der ES1334.2 erfolgt an der schließenden Flanke des Pulses.

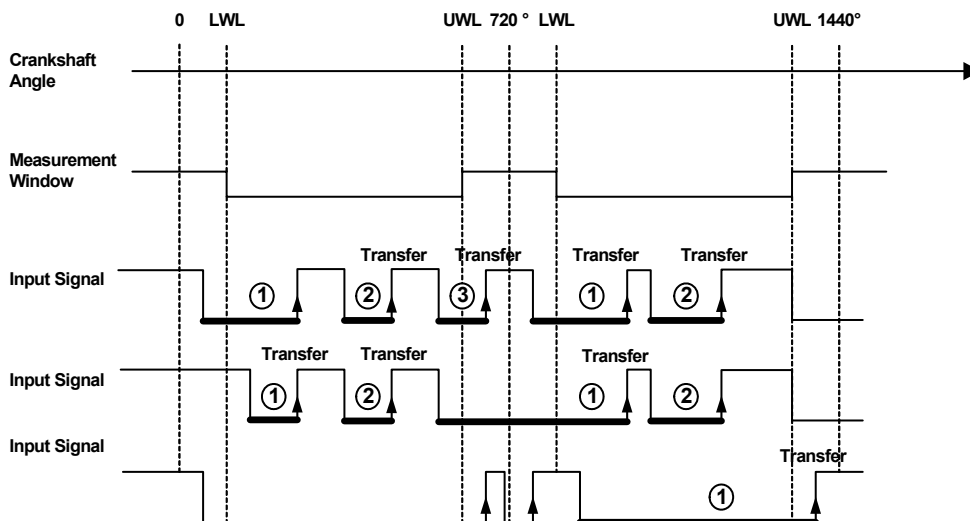
Bei den **puls-selektiven Messungen** („H-Time n-th Pulse (Pu Qual.) [ $\mu\text{s}$ ]“, „L-Time n-th Pulse (Pu Qual.) [ $\mu\text{s}$ ]“, „H-Time n-th Pulse (Eg Qual.) [ $\mu\text{s}$ ]“, „L-Time n-th Pulse (Eg Qual.) [ $\mu\text{s}$ ]“) wird der n-te gültige Puls innerhalb eines Winkelfensters gemessen. Die Pulsnummer wird mit dem Parameter „(Max) Pulse (Count)“ der Registerkarte „Signals“ definiert. Der Transfer der gemessenen Pulsweite erfolgt an der schließenden Flanke des n-ten gültigen Pulses.

**Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*



**Abb. 9-15** Darstellung der Funktionsweise einer puls-qualifizierten, low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.



**Abb. 9-16** Darstellung der Funktionsweise einer flanken-qualifizierten, low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Edge Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben. Die Zuordnung der Pulse zu den Winkelfenstern und die Zählung der Pulse innerhalb eines Winkelfensters ist ebenfalls dargestellt.

*Funktionsbeschreibung: Asynchrone Pulsweitenmessungen*

Bei den asynchronen Pulsweitenmessungen ist eine Unterscheidung in **puls-qualifizierte** und **flanken-qualifizierte Messungen** nicht möglich bzw. nicht nötig, da es keine Fensterfunktionalität zur Selektion von Pulsen gibt. Die puls-qualifizierten Pulsweitenmessungen haben im asynchronen Modus die gleiche Funktionalität wie die flanken-qualifizierten Messungen.

Die **nicht puls-selektiven Messungen** vermessen die Pulsweite jedes Pulses mit aktivem Pegel und transferieren den Messwert ins DPRAM an der schließenden Flanke des Pulses.

Die **puls-selektiven Pulsweitenmessungen** dürften im asynchronen Modus wohl keinen praktischen Anwendungsfall haben. Es wird die Pulsweite des n-ten Pulses mit aktivem Pegel nach der Initialisierung oder einer Konfiguration der ES1334.2 vermessen. Der Messwert wird an der schließenden Flanke des vermessenen Pulses ins DPRAM transferiert.

#### **Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

#### *Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-9 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	- Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, so wird als Messwert die Zeitdauer seit der letzten öffnenden Flanke des Signals ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-9** Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 9.5.2 Additive Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften additiver Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen dargestellt:

- „Additive Hightime [µs]“
- „Additive Lowtime [µs]“

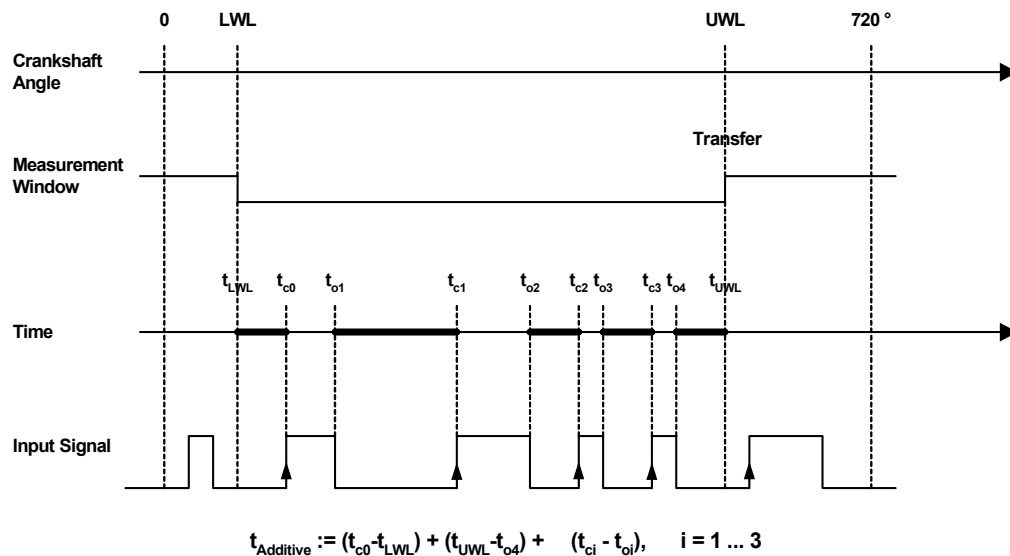
Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung

### Funktionsbeschreibung: Winkelsynchrone additive Pulsweitenmessungen

Die additive Zeit ergibt sich als Summe aller Zeitsegmente innerhalb eines Winkelfensters in denen das Signal aktiv ist, unabhängig davon, ob sich die öffnenden oder schließenden Flanken der Pulse innerhalb oder außerhalb eines Winkelfensters befinden. Abb. 9-17 zeigt beispielhaft die Messwertberechnung bei einer additiven Lowzeit-Messung.

Die additive Zeit wird an der oberen Winkelfenstergrenze in das DPRAM der ES1334.2 transferiert.



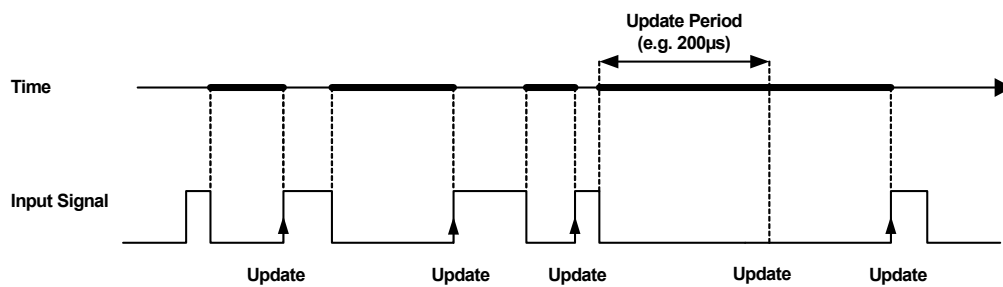
**Abb. 9-17** Darstellung der Funktionsweise einer winkelsynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [µs]“). Die additive Zeit setzt sich aus der Summe der fett markierten Linien-segmente zusammen.

### Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen

Bei asynchronen additiven Pulsweitenmessungen wird die Zeit summiert, in der das zu vermessende Signal aktiv ist. Die Aktualisierung der Summe im DPRAM erfolgt an jeder schließenden Pulsflanke spätestens jedoch nach einer vom Anwender definierten Aktualisierungs-Periode. Abb. 9-18 zeigt anhand einer additiven Lowzeit-Messung wann die Aktualisierung der additiven Zeit erfolgt. Die Einstellung der Aktualisierungs-Periode durch den Anwender erfolgt mit dem „Update Period for Additive Measurements [µs]“-Parameter in der Registerkarte „Globals“ des ES1334.2-VMI Subsystems. Eine Initialisierung oder eine Konfiguration der ES1334.2 setzen die summierte Zeit auf 0 zurück.

#### Hinweis

Die Erfassung der additiven Zeit erfolgt modulo  $2^{52} \mu\text{s}$ , d.h., nach  $(2^{52} - 1) \mu\text{s}$  (entspricht etwa 143 Jahren) springt die additive Zeit wieder auf 0 um.  $(2^{52} - 1)$  ist die größte ganze double precision Zahl, die mit einer Auflösung von 1.0 darstellbar ist.



**Abb. 9-18** Darstellung der Funktionsweise einer asynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [ $\mu\text{s}$ ]“). Die additive Zeit setzt sich aus den fett markierten Zeitsegmenten zusammen. Die Zeitpunkte an denen die Summe im DPRAM aktualisiert wird, sind ebenfalls dargestellt.

#### Timeout-Überwachung

Die Bedingung, wann ein Timeout ausgelöst wird, ist in Tab. 9-10 für verschiedene Verfahren zur Timeout-Überwachung aufgelistet. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-11 angegeben.

Timeout-Überwachung	Timeout-Bedingung
„Intvl InpDep“ „Intvl Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde <i>und</i> das Signal inaktiv war. Es spielt dabei im winkelsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„CS Angle InpDep“ „CS Angle Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde. Es spielt dabei im winkelsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„Intvl & CS Angle InpDep“ „Intvl & CS Angle Predef“	Die Timeout-Bedingung für diese Überwachungsverfahren ergibt sich durch eine ODER-Verknüpfung der Timeout-Bedingung der „Intvl InpDep“- und „Intvl Predef“-Verfahren mit der Timeout-Bedingung der „CS Angle InpDep“- und „CS Angle Predef“-Verfahren.

**Tab. 9-10** Additive Pulsweitenmessungen: Timeout-Bedingung für verschiedene Verfahren zur Timeout-Überwachung

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	<i>Asynchroner Messmodus:</i> Als Messwert im Timeout-Fall wird die bis zum Zeitpunkt der Überprüfung auf Timeout ermittelte additive Zeit ausgegeben. <i>Winkelsynchroner Messmodus:</i> Als Messwert im Timeout-Fall wird 0 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

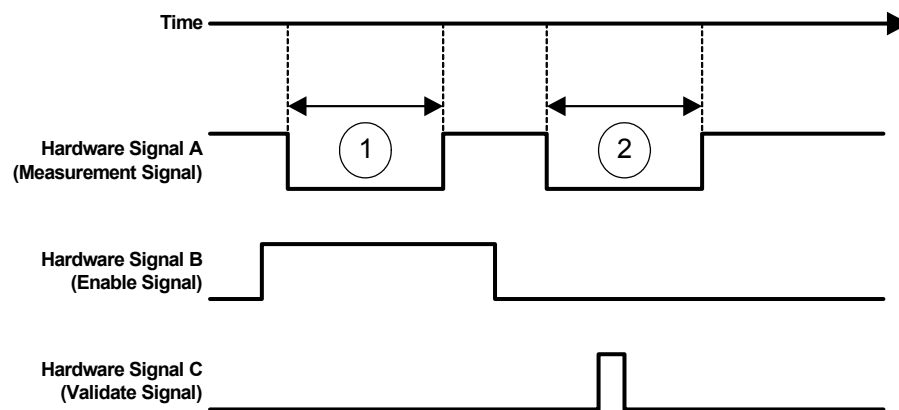
**Tab. 9-11** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 9.5.3 Pulsweitenmessungen mit Enable- oder Validierungssignal

Neben der direkten Erfassung der Pulsweiten eines von allen anderen Eingängen unabhängigen Signals bietet die ES1334.2 noch die Möglichkeit, Pulsweiten eines Signals in Abhängigkeit eines zweiten Eingangs zu messen. Dieses zweite Eingangssignal wird hierbei als Enable- bzw. Validate-Signal interpretiert, wie in Abb. 9-19 am Beispiel einer validierten Low-Zeit-Messung skizziert ist:

Wird die ES1334.2 beispielsweise derart konfiguriert, dass Low-Zeiten eines Eingangssignals A vermessen werden sollen, wobei ein zweites Eingangssignal B als high-aktives Enable-Signal genutzt werden soll, so würde vom Eingangssignal aus Abb. 9-19 lediglich der mit „1“ gekennzeichnete Puls vermessen. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Enable“ werden ausschließlich die Pulse vermessen, die **vollständig** von einem aktiven Enable-Puls **umschlossen sind**.

Wird hingegen ein weiteres Eingangssignal C als high-aktives Validate-Signal genutzt, so würde im obigen Beispiel lediglich der mit 2 gekennzeichnete Puls vermessen werden. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Validate“ werden ausschließlich die Pulse vermessen, die ihrerseits einen aktiven Validate-Puls **vollständig umschließen**.



**Abb. 9-19** Validierte Zeitmessungen mit der ES1334.2

Der ES1334.2-RTIO-Treiber bietet Pulsweitenmessungen mit Enable- oder Validate-Option an. Die Enable- und Validate-Pulse können dabei wahlweise als high-aktiv oder auch als low-aktiv definiert werden. Im Einzelnen wird in diesem Abschnitt die Funktionsweise folgender Messungen erläutert:

- „Hightime using H-Enable [ $\mu\text{s}$ ]“
- „Hightime using L-Enable [ $\mu\text{s}$ ]“
- „Hightime using H-Validate [ $\mu\text{s}$ ]“
- „Hightime using L-Validate [ $\mu\text{s}$ ]“
- „Lowtime using H-Enable [ $\mu\text{s}$ ]“
- „Lowtime using L-Enable [ $\mu\text{s}$ ]“
- „Lowtime using H-Validate [ $\mu\text{s}$ ]“
- „Lowtime using L-Validate [ $\mu\text{s}$ ]“
- „H-Time n-th Pulse (H-Ena) [ $\mu\text{s}$ ]“
- „H-Time n-th Pulse (L-Ena) [ $\mu\text{s}$ ]“
- „H-Time n-th Pulse (H-Val) [ $\mu\text{s}$ ]“
- „H-Time n-th Pulse (L-Val) [ $\mu\text{s}$ ]“
- „L-Time n-th Pulse (H-Ena) [ $\mu\text{s}$ ]“
- „L-Time n-th Pulse (L-Ena) [ $\mu\text{s}$ ]“
- „L-Time n-th Pulse (H-Val) [ $\mu\text{s}$ ]“
- „L-Time n-th Pulse (L-Val) [ $\mu\text{s}$ ]“

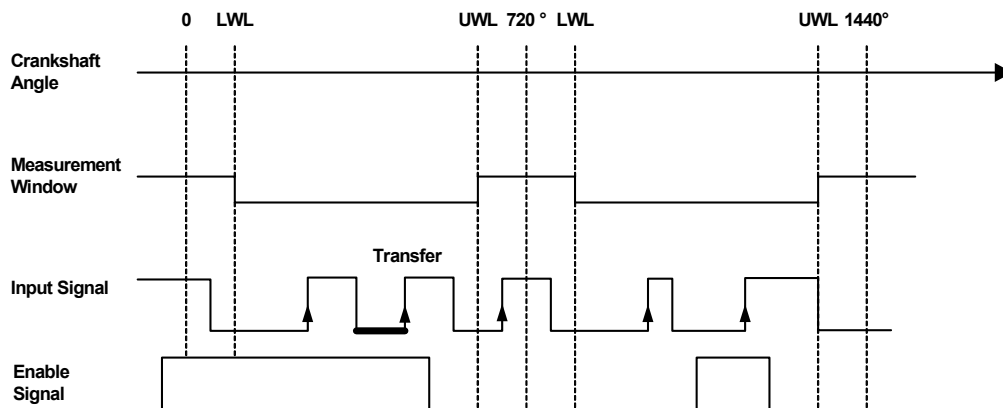
Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- „Reference Channel“: Hardwarekanal, an dem das Enable- bzw. Validate-Signal anliegt.
- Puls-Selektion (nur bei „H-Time n-th Pulse ...“ und „L-Time n-th Pulse ...“): „(Max) Pulse (Count)“ definiert die Nummer des zu vermessenden Pulses.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung



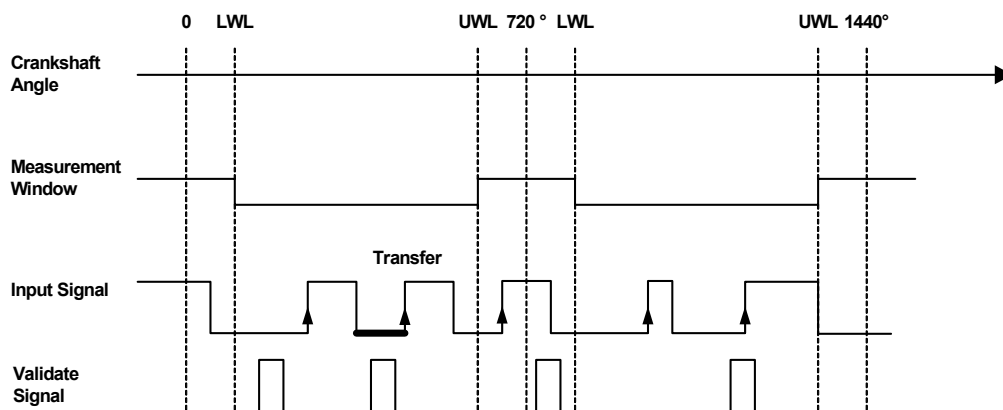
### Funktionsbeschreibung - Winkelsynchrone Pulsweitenmessungen mit Enable- oder Validierungssignal

Abb. 9-20 veranschaulicht die Funktionsweise von winkelsynchronen Pulsweitenmessungen mit Enable-Option anhand einer Low-Zeit-Messung mit high-aktivem Enable-Signal. Es werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig von einem aktiven Enable-Puls umschlossen sind (gültiger Puls).



**Abb. 9-20** Darstellung der Funktionsweise einer winkelsynchronen Pulsweitenmessung mit Enable-Signal (Messverfahren „Lowtime using H-Enable [ $\mu\text{s}$ ]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.

Abb. 9-21 veranschaulicht die Funktionsweise von winkelsynchronen Pulsweitenmessungen mit Validate-Option anhand einer Low-Zeit-Messung mit high-aktivem Validate-Signal. Es werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig einen aktiven Validate-Puls umschließen (gültiger Puls).



**Abb. 9-21** Darstellung der Funktionsweise einer winkelsynchronen Pulsweitenmessung mit Validate-Signal (Messverfahren „Lowtime using H-Validate [ $\mu\text{s}$ ]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.

Bei den **nicht puls-selektiven Messungen** wird jeder gültige Puls vermessen; der Transfer der gemessenen Pulsweite in das DPRAM der ES1334.2 erfolgt an der schließenden Flanke des Pulses.

Bei den **puls-selektiven Messungen** wird der n-te gültige Puls innerhalb eines Winkelfensters gemessen. Die Pulsnummer wird mit dem Parameter „(Max) Pulse (Count)“ der Registerkarte „Signals“ definiert. Der Transfer der gemessenen Pulsweite erfolgt an der schließenden Flanke des n-ten gültigen Pulses.

#### Hinweis

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

#### Funktionsbeschreibung - Asynchrone Pulsweitenmessungen mit Enable- oder Validierungssignal

Die nicht **puls-selektiven Messungen** vermessen die Pulsweite jedes durch ein Enable- oder Validate-Signal validierten Pulses mit aktivem Pegel und transferieren den Messwert ins DPRAM an der schließenden Flanke des Pulses.

Die **puls-selektiven Pulsweitenmessungen** dürften im asynchronen Modus wohl keinen praktischen Anwendungsfall haben. Es wird die Pulsweite des n-ten durch ein Enable- oder Validate-Signal validierten Pulses mit aktivem Pegel nach der Initialisierung oder einer Konfiguration der ES1334.2 vermessen. Der Messwert wird an der schließenden Flanke des vermessenen Pulses ins DPRAM transferiert.

#### Hinweis

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

#### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-12 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	- Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, so wird als Messwert die Zeitdauer seit der letzten öffnenden Flanke des Signals ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-12** Pulsweitenmessungen mit Enable- oder Validate-Signal: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 9.5.4 Frequenz- und Zykluszeitmessungen

Die ES1334.2 ermöglicht Messungen von Frequenzen und Zykluszeiten an steigenden und fallenden Signalfanken. Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Messverfahren zur Verfügung:

- „Cycle Time --/-- [ $\mu\text{s}$ ]“
- „Cycle Time --\-- [ $\mu\text{s}$ ]“
- „Frequency --/-- [Hz]“
- „Frequency --\-- [Hz]“

Zulässige Messoptionen sind:

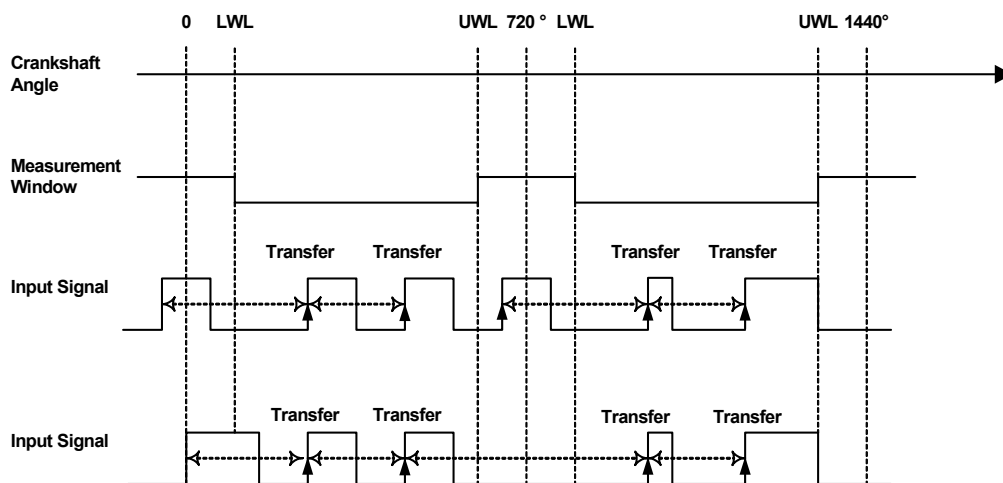
- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus
- Timeout-Überwachung

#### Funktionsbeschreibung

Im asynchronen Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalfanke. Im winkelsynchronen (winkelsegmentierten) Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalfanke, die innerhalb eines Winkelfensters liegt. Abb. 9-22 veranschaulicht die Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalfanken.

#### Hinweis

Wird bei einer winkelsegmentierten Zykluszeit- oder Frequenzmessung der gesamte Winkelbereich von  $720^\circ$  KW freigeschaltet, so verhält sich diese Messung wie die entsprechende Messung im asynchronen Modus.



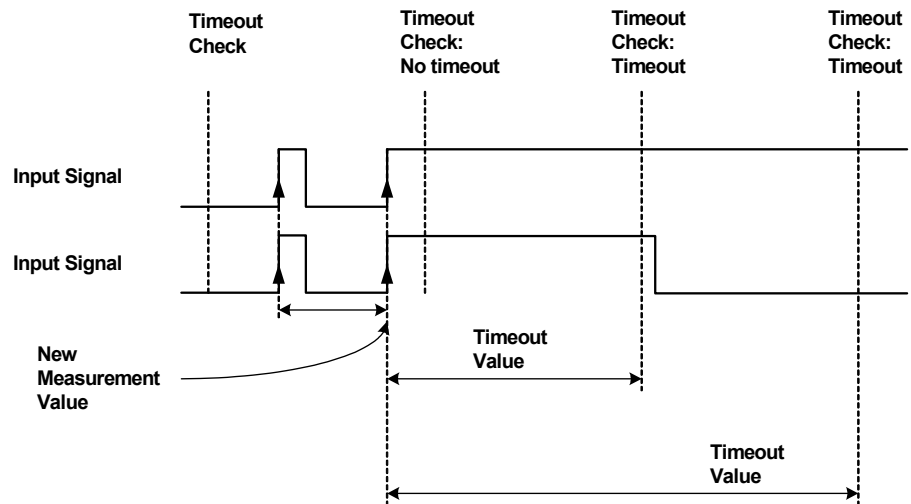
**Abb. 9-22** Darstellung der Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalfanken

*Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-13 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	<i>Zykluszeitmessungen:</i> Als Messwert wird im Timeout-Fall die Zeitdauer zwischen dem Zeitpunkt des Auftretens der letzten aktiven Flanke des Eingangssignals und dem Zeitpunkt, an dem die Überprüfung auf Timeout durchgeführt wird, ausgegeben (siehe Abb. 9-23). <i>Frequenzmessungen:</i> Die Messwertberechnung im Timeout-Fall erfolgt wie bei den Zykluszeitmessungen, jedoch wird aus der ermittelten Zeitdauer durch Kehrwertbildung eine resultierende Frequenz ermittelt.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-13** Frequenz- und Zykluszeitmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung



**Abb. 9-23** Frequenz- und Zykluszeitmessungen an ansteigenden Flanken: Darstellung der Messwertberechnung im Timeout-Fall (der Messwert ist als „Timeout Value“ bezeichnet) bei den „... InpDep“-Verfahren zur Timeout-Überwachung.

### 9.5.5 Tastverhältnismessungen

Tastverhältnisse (z.B. von PWM-Signalen) können mit der ES1334.2 auf unterschiedlichste Arten ermittelt werden. Bei den Tastverhältnismessungen wird unterschieden zwischen Messungen, die die High-Phase eines Signals ins Verhältnis zur Low-Phase setzen (oder umgekehrt) und Messungen, die die High-Phase (oder die Low-Phase) eines Signals ins Verhältnis zur Zyklusdauer setzen. Der erste Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis<sup>P/P</sup> bezeichnet. „P/P“ soll dabei andeuten, dass zwei Pulsdauern zueinander ins Verhältnis gesetzt werden. Der zweite Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis<sup>P/C</sup> bezeichnet. „P/C“ soll andeuten, dass hier eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt wird.

$$\text{Tastverhältnis}^{\text{P/P}}: \frac{L}{H}, \frac{H}{L} \quad 0 \leq \text{Tastverhältnis}^{\text{P/P}} \leq \infty$$

$$\text{Tastverhältnis}^{\text{P/C}}: \frac{L}{L+H}, \frac{H}{L+H} \quad 0 \leq \text{Tastverhältnis}^{\text{P/C}} \leq 1$$

Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Tastverhältnismessungen vom Typ „P/P“

- „Duty Factor L“H --/--“
- „Duty Factor L/H --\--“
- „Duty Factor H/L --/--“
- „Duty Factor H/L --\--“

und Tastverhältnismessungen vom Typ „P/C“

- „Duty Cycle L/(L+H) --/--“
- „Duty Cycle L/(L+H) --\--“
- „Duty Cycle H/(L+H) --/--“
- „Duty Cycle H/(L+H) --\--“

zur Verfügung.

Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung

#### Funktionsbeschreibung

Was die Messwertberechnung im asynchronen und winkelsynchronen (winkelsegmentierten) Messmodus anbelangt, so gilt bei Tastverhältnismessungen exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 9.5.4 auf Seite 223 beschrieben wurde.

### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-14 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	$\text{Tastverhältnis } P/P: \frac{L}{H}, \frac{H}{L}$ <p>- Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/H-Messungen oder „low“ bei H/L-Messungen ist. - Der Messwert ist MAXREAL32 (3.40282347 x10<sup>38</sup>), falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/H-Messungen oder „high“ bei H/L-Messungen ist.</p>
	$\text{Tastverhältnis } P/C: \frac{L}{L+H}, \frac{H}{L+H}$ <p>Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/(L+H)-Messungen oder „low“ bei H/(L+H)-Messungen ist. Der Messwert ist 1.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/(L+H)-Messungen oder „high“ bei H/(L+H)-Messungen ist.</p>
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-14** Tastverhältnismessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

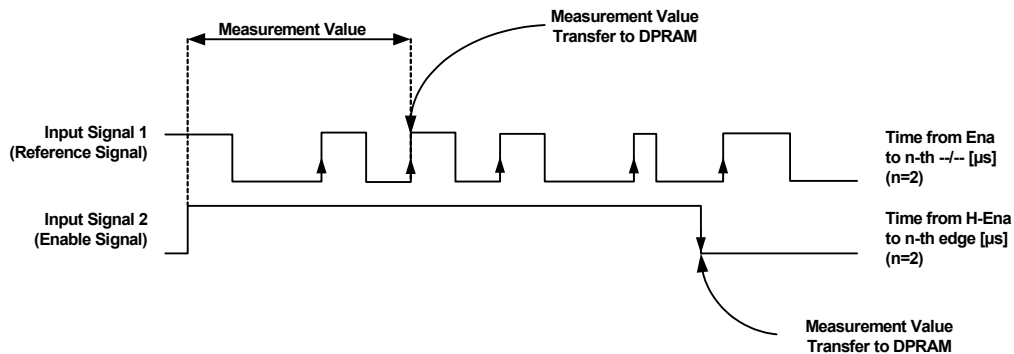
#### 9.5.6 Relativmessungen zwischen Hardwarekanälen

##### Funktionsbeschreibung

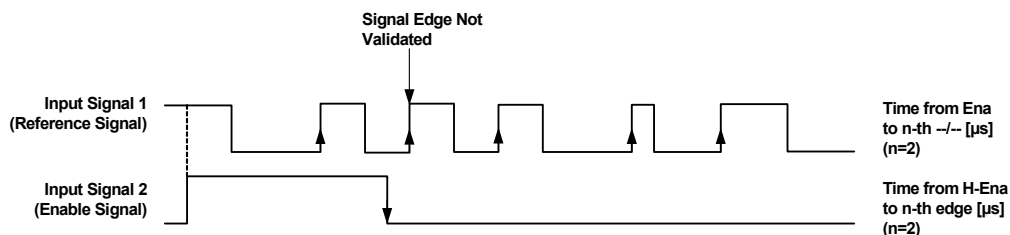
Den bisherigen Messverfahren war gemeinsam, dass die vermessenen Zeiten, Frequenzen, Winkel und Tastverhältnisse durch die Signalfanken eines Hardwarekanals definiert waren. In diesem Abschnitt werden Messverfahren beschrieben, in denen Ereignisse (Flanken) auf zwei Hardwarekanälen relativ zueinander vermessen werden.

Die Messfunktionalität soll anhand von Abb. 9-24 erläutert werden. Vermessen wird die Zeit von der öffnenden Flanke eines Enable-Pulses bis zur n-ten aktiven Flanke auf einem Referenz-Signal. Voraussetzung für die Ermittlung eines Messwerts ist jedoch, dass die n-te aktive Flanke durch den Enable-Puls validiert wird, d.h. die Flanke muss während der aktiven Phase des Enable-Signals auftreten.

Die Zählung der aktiven Flanken des Referenz-Signals wird mit der öffnenden Flanke des Enable-Pulses auf 0 zurückgesetzt. Der Parameter n zur Flanken-Selektion wird im Optionsfeld „(Max) Pulse (Count)“ in der Registerkarte „Signals“ eingestellt. Abb. 9-25 zeigt einen Signalverlauf bei dem keine Validierung des Referenz-Signals vorliegt.



**Abb. 9-24** Funktionsweise der Messverfahren zur Vermessung von Ereignissen (Flanken) auf unterschiedlichen Signalen bzw. Hardwarekanälen relativ zueinander. Dargestellt ist eine Messung mit high-aktivem Enable-Signal. Der RTIO-Parameter zur Flankenselektion („(Max) Pulse (Count)“ in der Registerkarte „Signals“) ist größer als 0.



**Abb. 9-25** Vermessung von Flanken auf unterschiedlichen Hardwarekanälen: Signalverlauf mit high-aktivem Enable-Signal und nicht validierter n-ter aktiver Signalfanke des Referenz-Signals

Die Durchführung der beschriebenen Messungen erfordert grundsätzlich die Konfiguration zweier Mess- bzw. Softwarekanäle. Einem Messkanal ist das Enable-Signal zuzuordnen. In Abhängigkeit davon, ob das Enable-Signal low-aktiv oder high-aktiv ist, ist dem Messkanal eines der beiden nachfolgenden Messverfahren zuzuweisen:

- „Time from L-Ena to n-th edge [µs]“
- „Time from H-Ena to n-th edge [µs]“

Zulässige Messooptionen sind:

- „Reference Channel“: Verweist auf den Messkanal, dem das Referenz-Signal zugeordnet ist.
- Timeout-Überwachung

Die Einstellungen aller anderen Messooptionen werden ignoriert.

Dem anderen Messkanal ist das Referenz-Signal zuzuordnen. In Abhängigkeit davon, ob die aktive Flanke des Referenz-Signals ansteigend oder fallend ist, ist dem Messkanal eines der beiden nachfolgenden Messverfahren zuzuweisen:

- „Time from Ena to n-th --/-- [μs]“
- „Time from Ena to n-th --\-- [μs]“

Zulässige Messoptionen sind:

- „Reference Channel“: Verweist auf den Messkanal, dem das Enable-Signal zugeordnet ist.
- Flanken-Selektion: „(Max) Pulse (Count)“ definiert die Nummer der zu vermessenden aktiven Flanke des Referenz-Signals.

Die Einstellungen aller anderen Messoptionen werden ignoriert.

### Hinweis

*Winkelsegmentierung über Winkelfenster wird nicht unterstützt. Aus diesem Grund gibt es keinen Unterschied in der Funktionalität von Messungen im asynchronen und im winkelsynchronen Messmodus.*

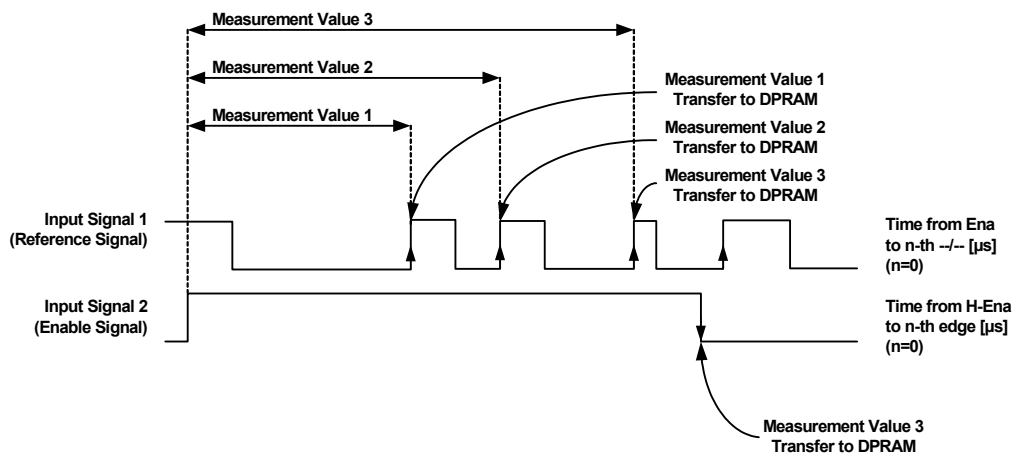
Grundsätzlich liefern beide Messkanäle einer Messung den ermittelten Messwert. Nur der Zeitpunkt, an dem der Messwert in das DPRAM der ES1334.2 übertragen wird, ist bei den beiden Messkanälen unterschiedlich.

Messkanäle, die auf die Messverfahren „Time from Ena to n-th --/-- [μs]“ oder „Time from Ena to n-th --\-- [μs]“ konfiguriert sind, übertragen den Messwert zum Zeitpunkt des Auftretens der n-ten validierten aktiven Flanke (siehe Abb. 9-24 auf Seite 227).

Messkanäle, die auf die Messverfahren „Time from L-Ena to n-th edge [μs]“ oder „Time from H-Ena to n-th edge [μs]“ konfiguriert sind, übertragen den Messwert zum Zeitpunkt des Auftretens der schließenden Flanke des Enable-Pulses (siehe Abb. 9-24 auf Seite 227).

Ein besonderes Messverhalten ergibt sich, wenn der Parameter zur Flanken-Selektion „(Max) Pulse (Count)“ in den Messverfahren „Time from Ena to n-th --/-- [μs]“ oder „Time from Ena to n-th --\-- [μs]“ zu 0 gesetzt wird. In diesem Fall liefern die Messverfahren „Time from L-Ena to n-th edge [μs]“ und „Time from H-Ena to n-th edge [μs]“ die Zeit von der öffnenden Flanke des Enable-Pulses bis zur **letzten** validierten aktiven Flanke des Referenz-Signals. Die Übertragung des Messwerts in das DPRAM erfolgt **an der schließenden Flanke des Enable-Pulses** (Abb. 9-26 auf Seite 229). Die Messverfahren „Time from Ena to n-th --/-- [μs]“ und „Time from Ena to n-th --\-- [μs]“ liefern in diesem Fall die Zeit von der öffnenden Flanke des Enable-Pulses zu **jeder** validierten aktiven Flanke des Referenz-Signals. Die Übertragung der Messwerte ins DPRAM erfolgt **zum Zeitpunkt des Auftretens der jeweiligen validierten aktiven Flanke**.





**Abb. 9-26** Funktionsweise der Messverfahren zur Vermessung von Ereignissen (Flanken) auf unterschiedlichen Signalen bzw. Hardwarekanälen relativ zueinander. Dargestellt ist eine Messung mit high-aktivem Enable-Signal. Der RTIO-Parameter zur Flanken-Selektion („(Max) Pulse (Count)“ in der Registerkarte „Signals“) ist gleich 0.

Tab. 9-15 fasst Konfigurationsrichtlinien und Messwerte der Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen zusammen.

RTIO-Parameter der Registerkarte „Signals“	Enable-Signal	Referenz-Signal
Hardware Channel	Nummer des Hardwarekanals, der das Enable-Signal führt	Nummer des Hardwarekanals, der das Referenz-Signal führt
Measurement Mode	„Asynchronous“ oder „ESpeedSync“. Es gibt keine Unterschiede zwischen den Modi.	„Asynchronous“ oder „ESpeedSync“. Es gibt keine Unterschiede zwischen den Modi.
Measurement Procedure	„Time from L-Ena to n-th edge [µs]“ oder „Time from H-Ena to n-th edge [µs]“ je nachdem, ob das Enable-Signal low- oder high-aktiv ist.	„Time from Ena to n-th --/- [µs]“ oder „Time from Ena to n-th --\-- [µs]“ je nachdem, ob die aktive Flanke des Referenz-Signals steigend oder fallend ist.
Reference Channel	Nummer des Messkanals, dem das Referenz-Signal zugewiesen wurde.	Nummer des Messkanals, dem das Enable-Signal zugewiesen wurde.
Messwert bei Einstellung (Max) Pulse (Count) = 0	Die Zeit von der öffnenden Flanke des Enable-Pulses zur <b>letzten</b> validierten aktiven Flanke des Referenz-Signals wird gemessen und <b>an der schließenden Flanke des Enable-Pulses</b> ins DPRAM übertragen.	Die Zeit von der öffnenden Flanke des Enable-Pulses zu <b>jeder</b> validierten aktiven Flanke des Referenz-Signals wird gemessen und <b>zum Zeitpunkt des Auftretens der jeweiligen aktiven Flanke</b> ins DPRAM übertragen.
Messwert bei Einstellung (Max) Pulse (Count) > 0	Die Zeit von der öffnenden Flanke des Enable-Pulses bis zur n-ten validierten aktiven Flanke des Referenz-Signals wird gemessen und <b>an der schließenden Flanke des Enable-Pulses</b> ins DPRAM übertragen.	Die Zeit von der öffnenden Flanke des Enable-Pulses bis zur n-ten validierten aktiven Flanke des Referenz-Signals wird gemessen und <b>zum Zeitpunkt des Auftretens der n-ten validierten aktiven Flanke</b> ins DPRAM übertragen.

**Tab. 9-15** Konfigurationsrichtlinien und Messwerte der Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-16 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	- Falls der Signalpegel des Enable-Signals zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. - Falls der Signalpegel des Enable-Signals zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-16** Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 9.5.7 Vermessung von Flanken: Winkelstempel

---

##### Funktionsbeschreibung

---

Die nachfolgenden Messverfahren vermessen den Kurbelwellenwinkel aktiver Signalfanken auf dem zugeordneten Hardwarekanal relativ zum eingestellten Referenzwinkel.

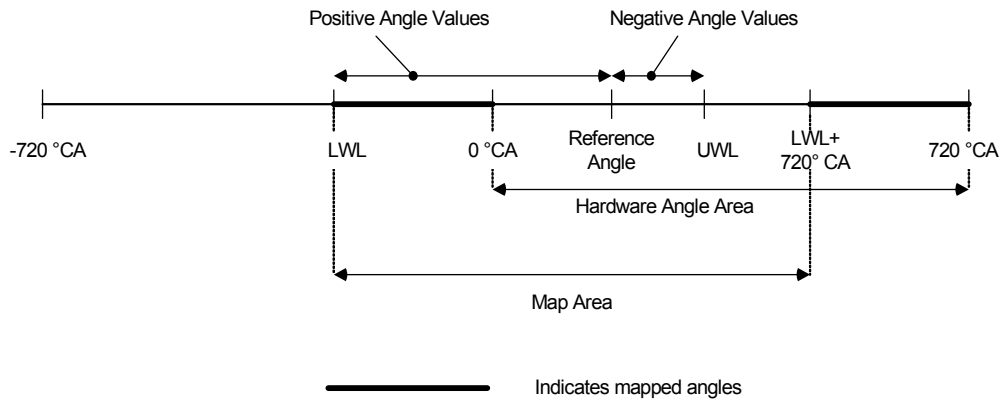
- „Rising Edge --/-- [deg]“
- „Falling Edge --\-- [deg]“
- „Rising Edge of n-th Pulse [deg]“
- „Falling Edge of n-th Pulse [deg]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Flanken-Selektion (nur bei „Rising Edge of n-th Pulse [deg]“ und „Falling Edge of n-th Pulse [deg]“): „(Max) Pulse (Count)“ definiert die Nummer der zu vermessenden Flanke.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus
- „CS Angle Reference“: Definiert den Referenzwinkel der Messung.
- Timeout-Überwachung

Der von der Hardware gelieferte Kurbelwellenwinkel einer Flanke liegt grundsätzlich im Bereich [0 °KW, 720 °KW]. Im **winkelsynchronen Messmodus** wird der gemessene Winkel von der ES1334.2-Firmware in den Bereich [LWL, LWL + 720 °KW] abgebildet und anschließend die Differenz mit dem Referenzwinkel

berechnet. Diese Differenz ist der von der Messung gelieferte Messwert, sie ist positiv, wenn der abgebildete Winkel kleiner als der Referenzwinkel ist (vgl. Abb. 9-27).



**Abb. 9-27** Funktionsweise einer Winkelmessung im winkelsynchronen Messmodus

Da der Referenzwinkel im Bereich  $[-720\text{ °KW}, 720\text{ °KW}]$  liegen kann, liefern die Messfunktionen im winkelsynchronen Messmodus Winkelwerte aus dem Bereich  $[-1440\text{ °KW}, 1440\text{ °KW}]$ . Im **asynchronen Messmodus** findet keine Abbildung der von der Hardware gelieferten Kurbelwellenwinkel statt, so dass in diesem Messmodus die Winkelwerte im Bereich  $[-1440\text{ °KW}, 720\text{ °KW}]$  liegen.

Im **winkelsynchronen Messmodus** erfassen die **nicht flanken-selektiven Messungen** („Rising Edge --/-- [deg]“, „Falling Edge --\-- [deg]“) jede aktive Flanke innerhalb eines Winkelfensters. Die **flanken-selektiven Messungen** („Rising Edge of n-th Pulse [deg]“, „Falling Edge of n-th Pulse [deg]“) vermessen die n-te aktive Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flanken-zähler auf 0 zurückgesetzt.

Im **asynchronen Messmodus** erfassen die **nicht flanken-selektiven Messungen** den Kurbelwellenwinkel jeder aktiven Flanke. Die **flanken-selektiven Messungen** ergeben in diesem Messmodus wenig Sinn, da der Flanken-zähler nicht zurückgesetzt wird.

#### Hinweis

Wird bei den flanken-selektiven Messungen im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht flanken-selektiven Messungen.

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-17 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der Low-Pegel, bei einer Messung an fallenden Flanken der High-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der High-Pegel, bei einer Messung an fallenden Flanken der Low-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-17** Winkelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 9.5.8 Vermessung von Flanken: Zeitstempel

---

##### Funktionsbeschreibung

---

Die nachfolgenden Messverfahren erfassen den Zeitpunkt aktiver Signalfanken auf dem zugeordneten Hardwarekanal.

- „Time Stamp --/-- [μs]“
- „Time Stamp --\-- [μs]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- „(Max) Pulse (Count)“: Definiert die Nummer der zu vermessenden Flanke
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus
- Timeout-Überwachung

Im **winkelsynchronen Messmodus** wird der Zeitpunkt des Auftretens der ersten aktiven Signalfanke innerhalb eines Winkelfensters gemessen. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenzähler auf 0 zurückgesetzt. Wird im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt, so wird jede aktive Flanke innerhalb eines Winkelfensters gemessen.

Im **asynchronen Messmodus** sollte im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt sein, dann wird der Zeitpunkt des Auftretens jeder aktiven Signalfanke gemessen. Ist die Flankennummer ungleich 0, so wird einmalig der Zeitpunkt der n-ten aktiven Flanke nach einer Initialisierung oder einem Konfigurationsschritt gemessen.

Die Zeiten werden relativ zum Zeitpunkt der letzten Initialisierung oder Konfiguration der ES1334.2 angegeben. Mit anderen Worten eine Initialisierung oder Konfiguration der ES1334.2 setzt die Uhr auf 0 zurück.

#### *Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-18 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert im Timeout-Fall wird der letzte (vor dem Timeout) berechnete Messwert beibehalten.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-18** Zeitstempelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

### 9.5.9 Pulszählung

In diesem Abschnitt werden die Eigenschaften folgender Messverfahren zur Pulszählung beschrieben:

- „Number of Low-Pulses“
- „Number of High-Pulses“
- „Total Number of L-Pulses“
- „Total Number of H-Pulses“

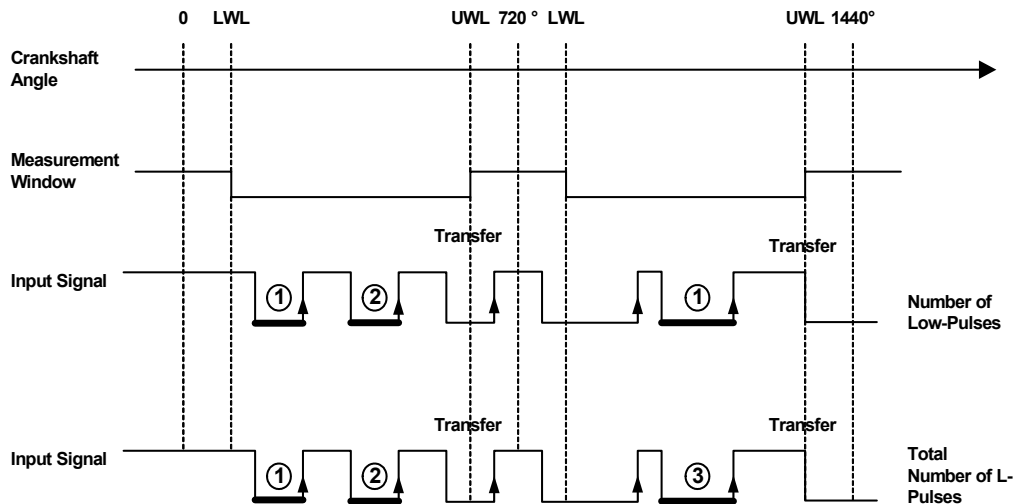
Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung

#### *Funktionsbeschreibung*

Die Messverfahren „Number of Low-Pulses“ und „Number of High-Pulses“ erfassen im **winkelsynchronen Messmodus** die Anzahl der gültigen aktiven Pulse innerhalb eines Winkelfensters. Gültige Pulse sind Pulse, die mit öffnender und schließender Flanke in ein- und demselben Winkelfenster liegen. Die ermittelte Summe wird an der oberen Winkelfenstergrenze in das DPRAM übertragen. Die Messverfahren „Total Number of L-Pulses“ und „Total Number of H-Pulses“

erfassen im winkelsynchronen Messmodus die Summe aller gültigen Pulse seit der letzten Initialisierung oder Konfiguration der ES1334.2. Die Summe wird an jeder oberen Winkelfenstergrenze aktualisiert (Abb. 9-28).



**Abb. 9-28** Darstellung der Funktionsweise winkelsynchroner Messverfahren zur Pulszählung (Messverfahren „Number of Low-Pulses“ und „Total Number of L-Pulses“). Pulse, die gezählt werden, sind durch erhöhte Linienstärke hervorgehoben. Die Summe wird jeweils an der oberen Winkelfenstergrenze in das DPRAM transferiert.

Im **asynchronen Messmodus** besitzen die „Number of ...“ und „Total Number of ...“ Messungen identische Funktionalität. Es wird die Gesamtzahl der aktiven Pulse seit der letzten Initialisierung oder Konfiguration der ES1334.2 gemessen. Die Summe wird an jeder schließenden Pulsflanke aktualisiert.

#### Hinweis

Die Pulszählung erfolgt modulo  $2^{52}$ , d.h. nach  $(2^{52} - 1)$  Pulsen springt die Summe wieder auf 0 um.  $(2^{52} - 1)$  ist die größte ganze double precision Zahl, die mit einer Auflösung von 1.0 darstellbar ist.

#### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein gültiger aktiver Puls aufgetreten ist. Im winkelsynchronen Messmodus ist ein gültiger aktiver Puls ein aktiver Puls, dessen öffnende und schließende Flanke inner-

halb ein und desselben Winkelfensters liegen. Im asynchronen Messmodus ist jeder aktive Puls ein gültiger aktiver Puls. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-19 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Zählung von Low-Pulsen der High-Pegel und bei einer Zählung von High-Pulsen der Low-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Zählung von Low-Pulsen der Low-Pegel und bei einer Zählung von High-Pulsen der High-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-19** Pulszählung: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 9.5.10 Pegelmessung

##### *Funktionsbeschreibung*

Die Messverfahren

- „Level (Active High)“
- „Level (Active Low)“

dienen zur Erfassung der aktiven Pegelzustände auf einem Hardwarekanal. Ein aktiver Pegel wird durch den Messwert „1“, ein inaktiver Pegel durch den Messwert „0“ signalisiert. Im **asynchronen Messmodus** wird der Messwert an jeder Flanke des Eingangssignals aktualisiert. Im **winkelsynchronen Messmodus** findet eine Aktualisierung nur an Signalflanken statt, die innerhalb des Winkelfensters auftreten.

Zulässige Messoptionen sind:

- „Measurement Mode“: Winkelsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im winkelsynchronen Messmodus.
- Timeout-Überwachung



### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 9-20 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert „0“ ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert „1“ ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 9-20** Pegelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 9.5.11 Positionsverfolgung an zwei-phasigen Schrittmotoren

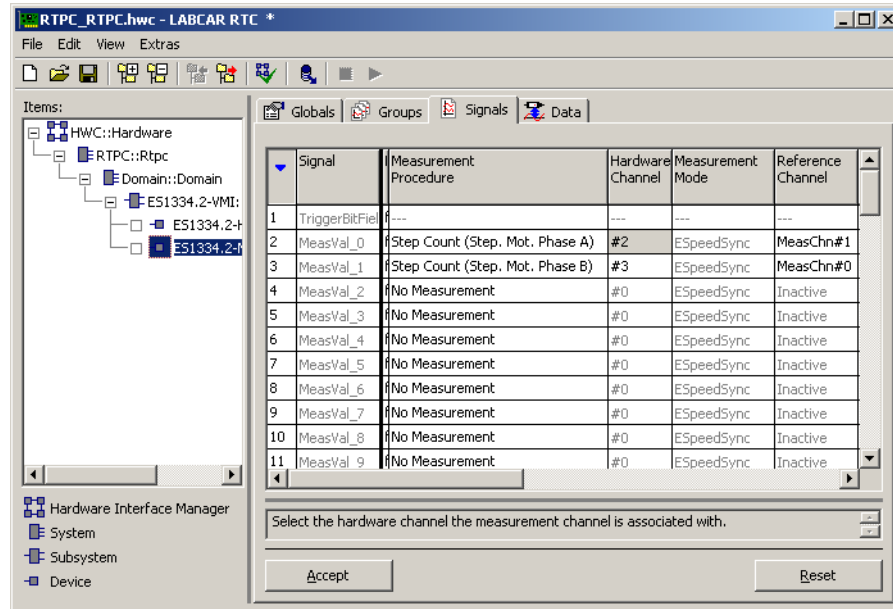
---

##### Funktionsbeschreibung

---

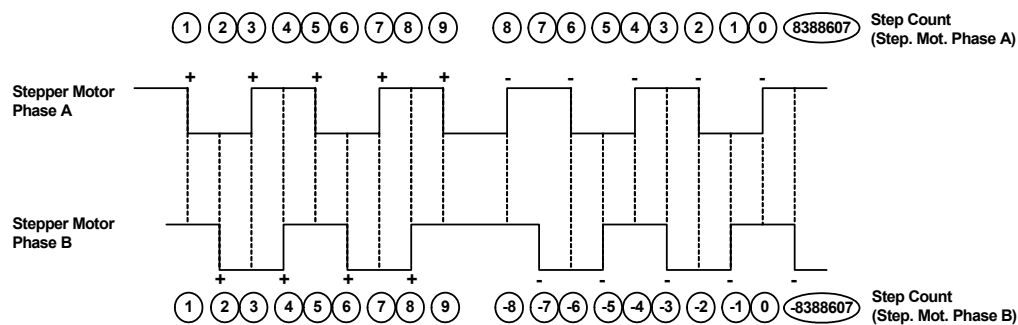
Die ES1334.2-Firmware ermöglicht die Positionsverfolgung und damit prinzipiell die Erfassung von Rotorstellung und -drehzahl von zwei-phasigen Schrittmotoren. Dazu sind zwei Hardwarekanäle und zwei Softwarekanäle bzw. Messkanäle erforderlich. An die beiden Hardwarekanäle werden die Ansteuersignale der beiden Phasenwicklungen eines zwei-phasigen Schrittmotors angelegt. Einem Messkanal ist das Messverfahren „Step Count (Step. Mot. Phase A)“ zuzuweisen und dem anderen Messkanal „Step Count (Step. Mot. Phase B)“. Im Feld „Hardware Channel“ der Messkanäle wird jeweils auf die Nummer des Hardwarekanals verwiesen an dem das Ansteuersignal für die jeweilige Phasenwicklung des zwei-phasigen Schrittmotors anliegt. Im Feld „Reference Channel“ eines Messkanals ist auf die Nummer des anderen an der Messung beteiligten Messkanals zu verweisen. Abb. 9-29 auf Seite 238 veranschaulicht die Vorgehensweise bei der Konfiguration. Die Ansteuersignale der Phasenwicklungen liegen an den

Hardwarekanälen 2 und 3 an. Die Messungen werden an den Messkanälen 0 und 1 vorgenommen. In Tab. 9-21 auf Seite 239 sind die Konfigurationsrichtlinien zusammengefasst.



**Abb. 9-29** Konfiguration einer Messung zur Positionsverfolgung an zwei-phasigen Schrittmotoren

An jeder Flanke des zwei-kanaligen Signals wird ein Positionszähler in- bzw. dekrementiert (Abb. 9-30) und der Wert des Positionszählers im DPRAM der ES1334.2 aktualisiert. Der Messkanal dem das Messverfahren „Step Count (Step. Mot. Phase A)“ zugeordnet ist, liefert stets einen nicht negativen Positionswert. Im Messwert des Messkanals dem das Messverfahren „Step Count (Step. Mot. Phase B)“ zugeordnet ist, ist im Vorzeichen des Positionswerts die Drehrichtung des Motors codiert. Einen aufsteigenden Positionszähler erhält man, wenn das Signal an „Phase A“ dem Signal an „Phase B“ vorausleitet.



**Abb. 9-30** Darstellung der Arbeitsweise der Positionsverfolgung an zwei-phasigen Schrittmotoren

**Hinweis**

Die Positionsverfolgung erfolgt modulo  $2^{23}$ . Inkrementieren bei einem Positionsstand von  $(2^{23}-1)$  erzeugt einen Wert des Positionszählers von 0. Dekrementieren bei einem Positionsstand von 0 erzeugt einen Wert des Positionszählers der betragsmäßig gleich  $(2^{23}-1)$  ist.

RTIO-Parameter der Registerkarte „Signals“	Phase A	Phase B
Hardware Channel	Nummer des Hardwarekanals, an dem das Ansteuersignal für Phasenwicklung A anliegt.	Nummer des Hardwarekanals an dem das Ansteuersignal für Phasenwicklung B anliegt.
Measurement Mode	Ohne Bedeutung	Ohne Bedeutung
Measurement Procedure	„Step Count (Step. Mot. Phase A)“	„Step Count (Step. Mot. Phase B)“
Reference Channel	Nummer des Messkanals, dem das Phase B Signal zugewiesen wurde.	Nummer des Messkanals, dem das Phase A Signal zugewiesen wurde.
Messwert	Aktueller Wert des Positionszählers modulo $2^{23}$	Aktueller Wert des Positionszählers modulo $2^{23}$ . Im Vorzeichen ist die Drehrichtung des Motors codiert.

**Tab. 9-21** Konfigurationsrichtlinien und Messwerte der Messverfahren zur Positionsverfolgung an zwei-phasigen Schrittmotoren

Timeout-Überwachung

Die Messverfahren zur Positionsverfolgung an zwei-phasigen Schrittmotoren bieten keine Timeout-Überwachung.



## 10 ESX335.1 Arbitrary Signal Generator Board

Ein ESX335.1 Arbitrary Signal Generator Board verfügt über sechs Signalausgänge, die flexibel genutzt werden können - jeder Ausgang kann mit einem der zur Verfügung stehenden internen Signale belegt werden. Als interne Signale stehen die Ausgänge von sechs arbiträren Signalgeneratoren sowie einem Klopf-generator mit vier internen Ausgängen zur Verfügung.

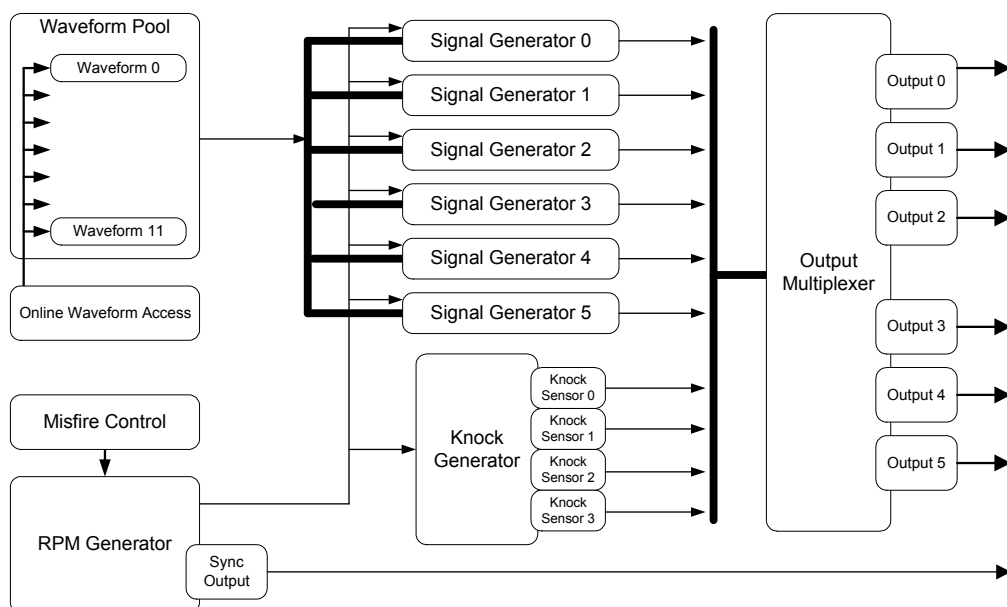
Die sechs arbiträren Signalgeneratoren können über einen zentralen Drehzahlgenerator (RPM-Generator) oder über eine individuelle lokale Frequenz getaktet werden. Eine individuelle Phasenverschiebung ist dabei bei jedem der arbiträren Signalgeneratoren möglich.

Es stehen zentral zwölf Wellenformen zur Verfügung, die wahlfrei von den sechs Signalgeneratoren ausgetaktet werden können. Die maximale Auflösung beträgt dabei 65536 Stützstellen. Die Wellenformen können online über das laufende Simulationsmodell verändert werden.

Die Drehzahl kann über die Zündaussetzsteuerung (Misfire-Control) moduliert werden. Dies ermöglicht winkelbezogene Drehzahlschwankungen um z.B. Zündaussetzer nachzubilden zu können.

Der Klopfsignalgenerator erzeugt die bei einem Verbrennungsmotor auftretenden Körperschallgeräusche, die durch Klopfen verursacht werden. Dabei ist die Frequenz und die Hüllkurve des Klopfsignals konfigurierbar. Eine zylinderindividuelle Zuordnung auf einen von vier internen Ausgängen des Klopfsignalgenerators ermöglicht die Nachbildung der Klopfsignale auch komplexerer Motoren.

Sowohl bei Zündaussetzern als auch beim Klopfgenerator werden bis zu zwölf Zylinder unterstützt.



## 10.1 Funktionsbeschreibung

---

### 10.1.1 Signalgeneratoren

---

Auf dem ESX335.1 Arbitrary Signal Generator Board stehen sechs arbiträre Signalgeneratoren zur Verfügung. Jeder der Signalgeneratoren kann dabei eine der zwölf zentral zur Verfügung stehenden Wellenformen abspielen. Als Taktquellen stehen ein zentraler RPM-Generator und ein pro Signalgenerator vorhandener lokaler Taktgenerator (Maximalfrequenz: 1 MHz) zur Verfügung.

Pro Signalgenerator kann eine individuelle Grundphase sowie eine zusätzliche Phasenverschiebung gewählt werden. Die Geschwindigkeit, mit der eine Änderung der Phasenverschiebung wirkt, ist wählbar.

Bei Verwendung des lokalen Taktgenerators kann die Frequenz des Taktgenerators, der Trigger-Modus (single shot, continuous) sowie ein Triggersignal vorgegeben werden.

Die Amplitude des internen Ausgangs-Signal des Signalgenerators kann zwischen 0.0 und 1.0 verändert werden.

### 10.1.2 MSA-Sensor

---

Eine besondere Verwendung der Signalgeneratoren ist die Nachbildung von Kurbelwellensensoren mit Drehrichtungserkennung (MSA-Sensoren). Ein Zahnimpuls hat dabei keine feste Winkelbreite sondern eine feste Pulsdauer. Außerdem ist als Ausgabesignal ein Low-aktives Open-Collector-Signal vorgegeben.

Wird ein MSA-Sensor RTIO-Element verwendet, so wird für alle Wellenformverläufe während der Konfiguration (potentielle) Zahnmittelinformationen berechnet und im Wellenformspeicher abgelegt. Möglicherweise sind aber nicht alle Wellenformen für diesen Algorithmus geeignet – bei ungeeigneten Wellenformen wird bei Auswahl der entsprechenden Wellenform eine Fehlermeldung übermittelt.

### 10.1.3 RPM-Generator

---

Das ESX335.1 Arbitrary Signal Generator Board verfügt über einen zentralen Drehzahlgenerator (RPM-Generator), der ein motordrehzahlspezifisches Taktsignal ausgibt. Dieses Taktsignal kann von den Signalgeneratoren zum Austakten der Wellenformen verwendet werden. Die maximale Drehzahl beträgt 30000 rpm, die Auflösung in etwa 0,1 rpm. Das Drehzahlsignal selbst kann über einen Misfire-Generator moduliert werden.

Eine Kopplung mehrerer ESX335.1 Arbitrary Signal Generator Boards ist möglich. Dabei muss genau eine der verwendeten ESX335.1 als RPM-Master konfiguriert sein, alle anderen ESX335.1 müssen als RPM-Slave konfiguriert werden. Damit steuert der RPM-Master die Drehzahl für alle ESX335.1 im System. Bei Verwendung einer einzigen ESX335.1 im System muss diese als RPM-Master konfiguriert sein.

Das Drehzahlsignal kann zu Messzwecken auf den Anschluss „SYNC“ (auf der Frontplatte) der ESX335.1 gelegt werden (siehe „Sync Port“ auf Seite 256).

### 10.1.4 Winkelauflösung

---

Die maximale Winkelauflösung beträgt 65536 Punkte pro Zyklus. Bei einem typischen Viertaktmotor mit einer Periode von 720 °KW entspricht das einer Winkelauflösung von ca. 0.01 °KW.

Die Auflösung kann in Zweierpotenzen bis auf 16 Punkte reduziert werden - dies entspricht einer Winkelauflösung von ca. 45 °KW.

#### 10.1.5 Wellenformspeicher für Signalgeneratoren

Es stehen zwölf Wellenformen zur Verfügung, die von den sechs arbiträren Signalgeneratoren verwendet werden können. Die Wellenformen können vom Anwender über Tabellen beschrieben werden. Über ein Interpolationsverfahren wird der Signalverlauf in der Tabelle in die jeweilige Wellenform geschrieben.

##### **Wellenformauflösung:**

Die maximale Auflösung einer Wellenform wird durch die maximal mögliche Zahl von 65536 Stützstellen bestimmt. Auch hier kann die Auflösung in Zweierpotenzen bis auf 16 Punkte reduziert werden; dabei ist zu beachten, dass die Auflösung ( $1/(\text{Anzahl der Stützstellen})$ ) einer Wellenform kleiner oder gleich der Winkelauflösung sein muss. Typischerweise sollte die Auflösung einer Wellenform mit der Winkelauflösung übereinstimmen.

Die Wellenformen werden von den Signalgeneratoren ausgetaktet. Dabei kann entweder der zentrale RPM-Generator als Taktquelle fungieren, oder es wird ein lokaler Frequenzgenerator (maximale Frequenz: 1 MHz) im Signalgenerator verwendet.

##### **Wellenformauflösung kleiner als Winkelauflösung:**

Falls über den Signalgenerator (unter Verwendung des lokalen Frequenzgenerators) ein hochfrequentes Signal ausgegeben werden soll, ist es eventuell nötig, die Auflösung einer Wellenform kleiner als die Winkelauflösung zu halten.

Das folgende Beispiel illustriert das Vorgehen:

Angenommen, es soll ein Sinussignal von 40 kHz ausgegeben werden, die Signaltabelle beschreibt eine einzige Sinus-Periode. Die Winkelauflösung beträgt 65536 Punkte. Durch die maximale Frequenz des lokalen Taktgenerators von 1 MHz ergibt sich dadurch eine maximale Signalfrequenz für das Sinussignal von  $1 \text{ MHz}/65536 = 15,25 \text{ Hz}$ , was natürlich weit von den gewünschten 40 kHz entfernt ist. Durch Verkleinern der Wellenformauflösung auf z.B. 16 Stützstellen wird das Sinussignal mehrfach hintereinander (konkret  $65536/16 = 4096$ -fach) in der Wellenform mit 65536 Stützstellen abgelegt. Damit ergibt sich eine resultierende Maximalfrequenz für das Sinussignal von  $1 \text{ MHz}/16 = 62,5 \text{ kHz}$ , was über der gewünschten Frequenz von 40 kHz liegt. Durch eine entsprechende Absenkung der lokalen Taktfrequenz auf 640 kHz kann man damit das gewünschte Sinussignal mit 40 kHz erzeugen.

Das Beispiel zeigt, dass durch eine Verkleinerung der Wellenformauflösung gegenüber der Winkelauflösung die Wellenformauflösung nicht wirklich verkleinert wird. Es wird lediglich das Signal der Signaltabelle mehrfach hintereinander in die Wellenform geschrieben und damit die „sichtbare“ Auflösung verkleinert.

#### 10.1.6 Direkter Wellenformzugriff aus der Simulation

Um aus dem laufenden Simulationsmodell die Wellenformen ändern zu können, gibt es einen speziellen Zugriff zur Veränderung der Wellenformen zur Laufzeit.

Dabei können pro Simulationstakt bis zu 16 Stützstellen einer der zwölf vorhandenen Wellenformen verändert werden. Dabei wird die Stützstellennummer, die Anzahl der zu veränderten Stützstellen sowie die neuen Datenwerte zur Beschreibung verwendet.

### 10.1.7 Ausgangsmultiplexer

---

Der Ausgangsmultiplexer ermöglicht die Zuordnung der internen Signale auf die sechs vorhandenen physikalischen Ausgänge der ESX335.1. Dabei kann der Ausgangstreiber (Analog, Digital ohne Pull-Up-Widerstand, Digital mit Pull-Up-Widerstand, Open-Collector) sowie die Referenzspannung des D/A-Wandlers (interne oder externe Referenzspannung) gewählt werden. Ein internes Signal kann dabei bei Bedarf auf mehrere physikalische Ausgänge mit verschiedener Beschaltung gelegt werden.

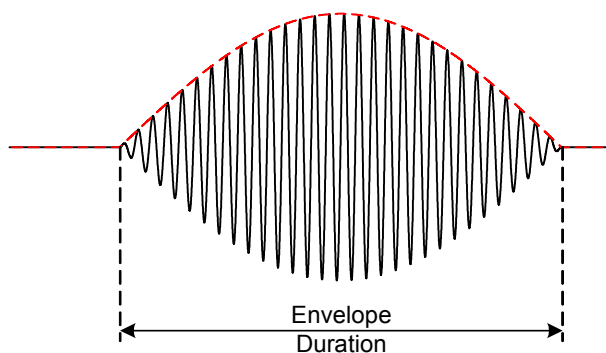
Zu beachten ist, dass die internen Signale einen Wertebereich von -1.0 bis +1.0 aufweisen. Erst durch den D/A-Wandler im Ausgangsmultiplexer wird aus dem internen Signal tatsächlich ein Spannungssignal erzeugt.

### 10.1.8 Klopfsignalgenerator

---

Die bei einem Verbrennungsmotor auftretenden Klopfgeräusche können durch den Klopfsignalgenerator nachgebildet werden. Ein Klopfpaket besteht aus einzelnen Klopfpaketen. Ein Klopfpaket selbst besteht aus einer Sinusschwingung mit wählbarer Frequenz und einer die Sinusschwingung modulierenden Hüllkurve mit definierbarer Dauer.

Die nachfolgende Abbildung zeigt ein einzelnes Klopfpaket. Dabei wird als Hüllkurve eine Sinus-Halbwellenform verwendet.



**Abb. 10-1** Ein Klopfpaket

Auch eine nicht-klopfende Verbrennung erzeugt Geräusche, die von einem realen Körperschall-Klopfensensor erfasst werden. Über die Steuerung der Amplitude des Klopfsignals wird zwischen einer korrekten und einer klopfenden Verbrennung unterschieden.

Zusätzlich gibt es noch eine stochastische Variation der Amplitude eines Klopfpakets. Damit werden die im realen Betrieb auftretenden Änderungen der Klopfsignale nachgebildet.

Pro Zylinder kann nun individuell die Winkellage (in °KW) eines Klopfsignals sowie das Auftreten des Klopfereignisses per Wahrscheinlichkeitswert oder mittels Sequenztabelle (siehe „Sequenztabelle“ auf Seite 245) gesteuert werden.

Der Klopfsignalgenerator verfügt über vier interne Ausgänge. Für jeden Ausgang kann gewählt werden, welche Zylinder den jeweiligen Ausgang bedienen. Speziell bei viel-zylindrigen Motoren ist zu beachten, dass ein Klopfereignis auf einem Ausgang ein noch nicht vollständig abgelaufenes Klopfsignal auf dem selben Ausgang unterbricht.



### 10.1.9 Zündaussetzsteuerung

Zur Nachbildung von Zündaussetzern ist eine entsprechende Steuerung auf dem ESX335.1 Arbitrary Signal Generator Board vorhanden, die eine Modulation der Drehzahl des RPM-Generators in einem bestimmten Winkelbereich bewirkt. Dabei ist es möglich, die Drehzahl über den Kurbelwellenwinkel relativ zur vorgegebenen Drehzahl des RPM-Generators zu verändern (verringern/erhöhen um den Faktor 0,01 bis 2,0). In der typischen Anwendung zur Nachbildung von Zündaussetzern wird die Drehzahl gegenüber der vorgegebenen Drehzahl verringert.

Der Startwirkungswinkel der Drehzahlmodulation ist zylinderindividuell festlegbar. Über einen Wahrscheinlichkeitswert oder über Sequenztabelle (siehe „Sequenztabelle“ auf Seite 245) kann das Wirken der Drehzahlmodulation pro Zylinder gesteuert werden.

Die Drehzahlmodulation kann über vier vorhandenen Modulationsprofile vorgegeben werden, die den Modulationsverlauf über eine komplette Periode von 720 °KW darstellen. Dabei entspricht der Wert 1.0 einer nicht vorhandenen Modulation, ein Wert von 0.01 verringert die Drehzahl auf 1% der vorgegebenen Drehzahl, der Wert 2.0 verdoppelt die vorgegebene Drehzahl. Pro Zylinder kann individuell eines der vier vorhandenen Modulationsprofile ausgewählt werden.

### 10.1.10 Sequenztabelle

Beim Zündaussetzgenerator und beim Klopfsignalgenerator kommen Sequenztabelle zum Einsatz, die es dem Benutzer ermöglichen, komplexe Abläufe von Klopf- bzw. Fehlzündungen darzustellen.

Dabei wird eine Tabelle mit maximal 100 Stützstellen verwendet. Nach Starten der Sequenz wird pro Periode eine Stützstelle vorangegangen. Ein Wert größer als 0.5 an der jeweiligen Stützstelle bedeutet dabei, dass ein Ereignis (Klopfen bzw. Fehlzündung) in dieser Periode auftritt.

Nach 100 Stützstellen wird sofort von vorne begonnen („Sequence trigger = continuous“), oder das Abspielen wird beendet („Sequence trigger = Single Shot“) und muss über ein entsprechendes Triggersignal erneut gestartet werden.

Es ist grundsätzlich möglich, pro Zylinder eine individuelle Sequenz vorzugeben. Jedoch gibt es sowohl beim Zündaussetz- als auch beim Klopfgenerator eine gemeinsame Sequenz („Common Sequence“), auf die alle Zylinder zugreifen können. Das erleichtert das rasche Einstellen von Sequenzen, die für mehrere Zylinder verwendet werden sollen.

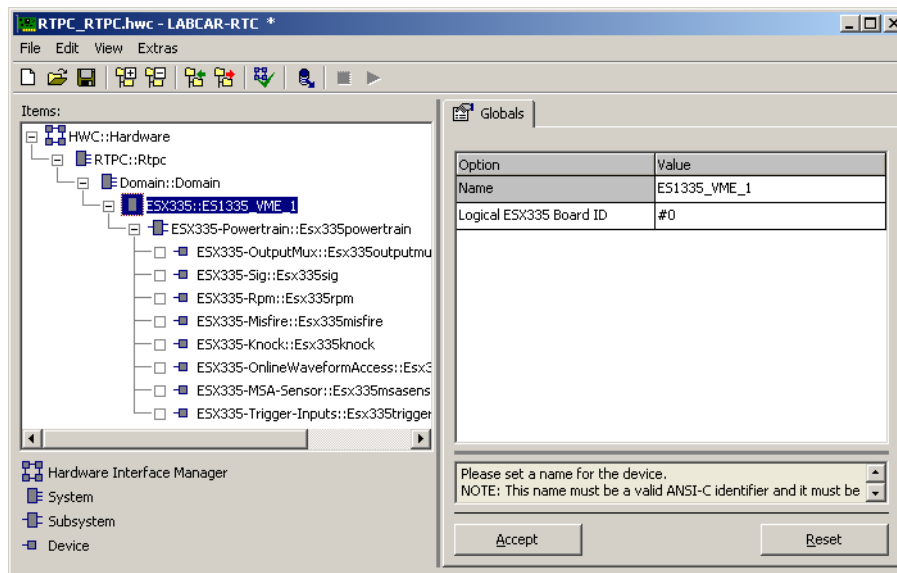
### 10.1.11 Aufbau des ESX335.1 RTIO-Baums

Das ESX335.1 Arbitrary Signal Generator Board wird im RTIO-Editor durch Auswahl eines ESX335 Subsystems gefolgt von einem ESX335-Powertrain Subsystem eingebunden.

## 10.2 ESX335 Subsystem

### 10.2.1 Globals (ESX335 Subsystem)

Das ESX335 Subsystem dient zur Einstellung von Parametern, die global gültig sind, d.h. die auf alle ESX335.1 RTIO-Elemente wirken.



**Abb. 10-2** Die Registerkarte „Globals“ des ESX335 Subsystems

#### *Logical ESX335 Board ID*

Dieses Optionsfeld dient zur Identifizierung der ESX335.1. Es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ESX335.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist.

Die Nummerierung der Karten im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Nummerierung) beginnend mit 1. Die ermittelte Nummer der ESX335.1 ist in dem Listenfeld „Logical ESX335 Board ID“ einzustellen. Im RTIO-Editor können bis zu 20 ESX335.1 pro Baugruppenträger eingebunden werden.

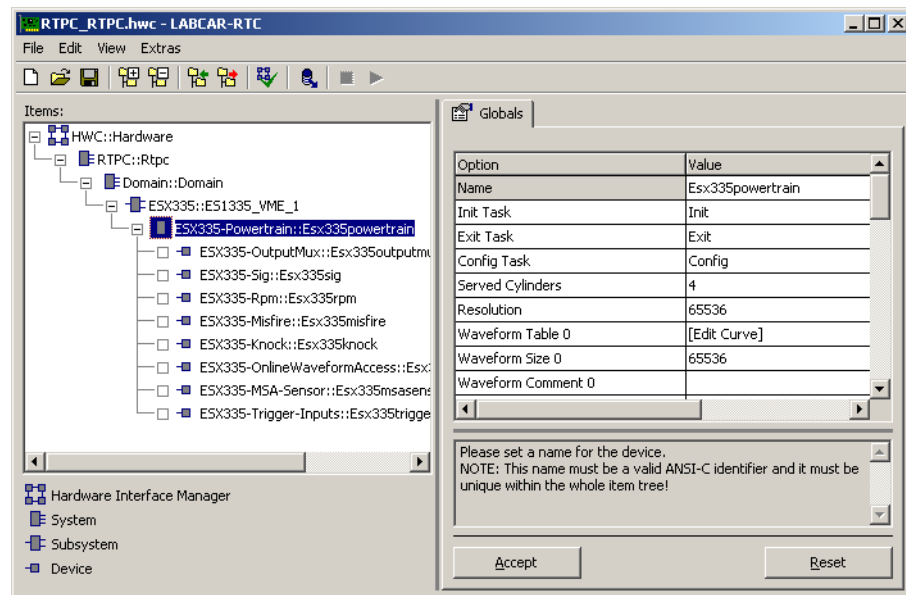
Dieser RTIO-Parameter ist nicht online, (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

## 10.3 ESX335-Powertrain Subsystem

### 10.3.1 Globals (ESX335-Powertrain Subsystem)

Das ESX335-Powertrain Subsystem dient zur Einstellung von RTIO-Parametern, die global wirksam sind, d.h. die auf alle Elemente des ESX335-Powertrain Subsystems wirken.

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-3** Die Registerkarte „Globals“ des ESX335-Powertrain Subsystems  
*Served Cylinders*

Hier wird die Anzahl der unterstützten Zylinder beim Klopfsignalgenerator und bei der Zündaussetzsteuerung eingestellt. Mögliche Werte sind 1 ... 12.

#### *Resolution*

Dieses Optionsfeld dient zur Festlegung der Winkelauflösung. Dabei sind Werte von 16 bis 65536 in Zweierpotenzen möglich.

#### *Waveform Table 0...11*

Hier können die Signalformen der zwölf vorhandenen Wellenformen vorgegeben werden.

#### *Waveform Size 0...11*

Die Stützstellenanzahl der zwölf vorhandenen Wellenformen können über die entsprechenden Optionsfelder vorgegeben werden (siehe auch „Wellenformauf-  
lösung“ auf Seite 243 und „Wellenformauf-  
lösung kleiner als Winkelauf-  
lösung“ auf Seite 243).

Waveform Comment 0...11

Um einen einfachen Überblick über die Verwendung der zwölf Wellenformen zu haben, kann ein individueller Kommentar vergeben werden.

**Hinweis**

*Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.*

MSA Sensors: Logic Level of Teeth

Dieser Parameter legt fest, ob bei der Berechnung der Zahnmitten (für einen MSA-Sensor) ein Signal mit negativer Logik (neg. Puls entspricht „Zahn“) oder positiver Logik (pos. Puls entspricht „Zahn“) zu interpretieren ist. Diese Einstellung gilt für alle 12 Wellenformen.

Der Parameter ist online editierbar, jedoch müssen bei einer Änderung alle Wellenformen neu berechnet und in den Wellenformspeicher auf der Karte geschrieben werden.

**Hinweis**

*Dieser Parameter wird nur dann angezeigt, wenn mindestens ein ESX335-MSA-Sensor Device eingefügt wurde.*

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Served Cylinders	uint32	Ja	Unterstützte Zylinder für Klopfen und Fehlzündung Wertebereich: 1 ... 12
Resolution	uint32	Ja	Winkelauflösung Wertebereich: 16 ... 65536 in Zweierpotenzen
Waveform Table 0...11	Table	Ja	Die jeweilige Wellenform als Tabelle. Wertebereich: -1.0 ... +1.0
Waveform Size 0...11	uint32	Ja	Stützstellenanzahl der jeweiligen Wellenform Wertebereich: 16 ... 65536 in Zweierpotenzen
Waveform Comment 0...11	Identifizier	Nein	Kommentar zur jeweiligen Wellenform
MSA Sensors: Logic Level of Teeth	bool	Ja	Interpretation der Wellenform für MSA-Sensor Wertebereich: 0=FALSE (neg. Logik)  , 1=TRUE (pos. Logik)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-1** ESX335-Powertrain Subsystem: Konfigurationsparameter der Registerkarte „Globals“

## 10.4 ESX335-OutputMux Device

### 10.4.1 Globals (ESX335-OutputMux Device)

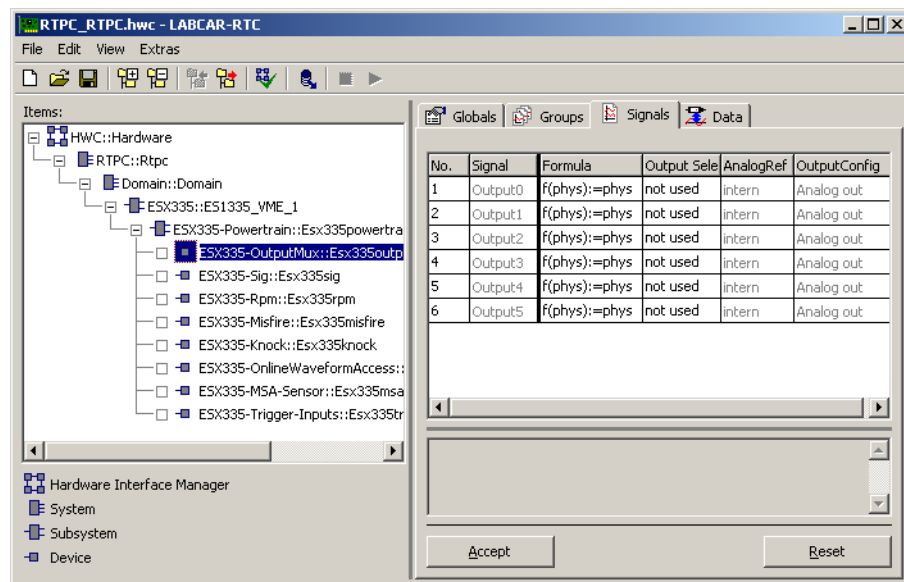
Das ESX335-Mux Device dient zur Konfiguration und Ansteuerung des Ausgangsmultiplexers der ESX335.1. In der Registerkarte „Globals“ gibt es keine Einstellmöglichkeit. In der Registerkarte „Signals“ werden ausgangsspezifische Einstellungen vorgenommen.

### 10.4.2 Groups (ESX335-OutputMux Device)

Das ESX335-OutputMux besitzt eine Signalgruppe zur Steuerung der Ausgangsrelais.

### 10.4.3 Signals (ESX335-OutputMux Device)

In der Registerkarte „Signals“ wird die Konfiguration der sechs Ausgangsmultiplexer durchgeführt. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. In Tab. 10-2 auf Seite 250 sind die Eigenschaften der einzelnen Parameter noch einmal zusammengefasst. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 10-4** Die Registerkarte „Signals“ des ESX335-OutputMux Device

#### Output Select

Das Optionsfeld „Output Select“ dient zur Auswahl des internen Signals, das auf den jeweiligen Ausgang gelegt werden soll. Es wird dabei eine Liste aller möglichen internen Signalquellen (Signalgeneratoren, Klopfsignalgenerator) zur Auswahl angeboten. Ein Klopfsignalgenerator verfügt über vier interne Ausgänge, die mit #0 bis #3 bezeichnet werden.

#### AnalogRef

Über dieses Optionsfeld kann die Referenzspannung für den D/A-Wandler des jeweiligen Ausgangskanals gewählt werden (siehe hierzu „Ausgangsmultiplexer“ auf Seite 244). Mögliche Werte sind „intern“ für eine interne Referenzspannung von 10 V und „extern“ für eine extern anliegende Referenzspannung.

*OutputConfig*

Über dieses Optionsfeld kann der Ausgangstreiber des jeweiligen Kanals gewählt werden (siehe hierzu „Ausgangsmultiplexer“ auf Seite 244). Mögliche Werte sind „Analog out“ für die analoge Ausgangsstufe, „Digital out with pull-up“ für eine digitale Ausgangsstufe mit internem Pull-Up-Widerstand gegen 5 V sowie „Digital out without pull-up“ für eine digitale Ausgangsstufe ohne internen Pull-Up-Widerstand.

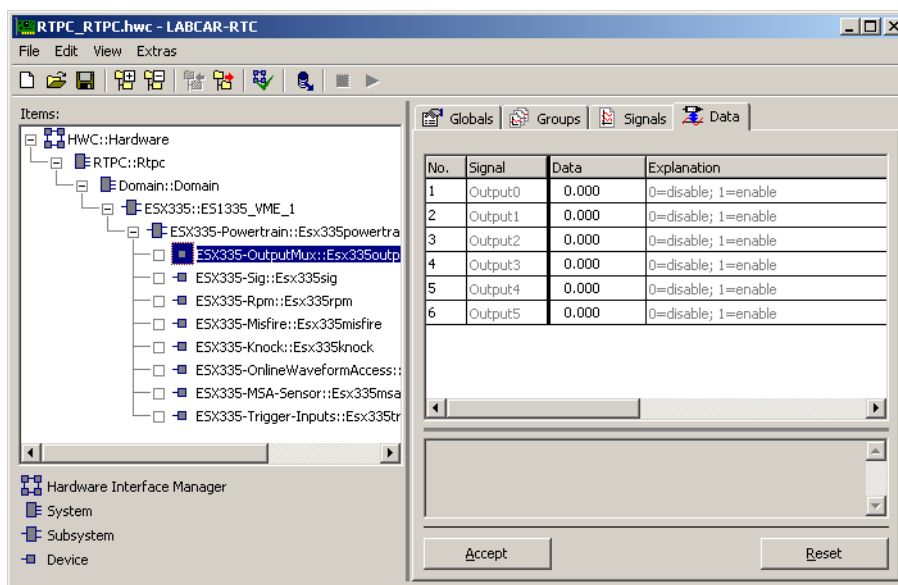
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Output Select	uint8	Ja	Selektion eines internen Signals 0-5: Signalgenerator 6-9: Klopfgenerator #0 - #3
Analog Ref	uint8	Ja	Referenzspannung für D/A-Wandler 0: intern 1: extern
OutputConfig	uint8	Ja	Ausgangstreiber 0: Analog 1: Digital ohne Pull-Up 2: Digital mit Pull-Up

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-2** ESX335-OutputMux Device: Konfigurationsparameter der Registerkarte „Signals“

10.4.4 Data (ESX335-OutputMux Device)

In der vorhandenen Signalgruppe werden sechs RTIO-Signale verarbeitet, die das Schließen des Ausgangsrelais eines jeden Kanals steuern.



**Abb. 10-5** Die Registerkarte „Data“ des ESX335-OutputMux Device

Die folgende Tabelle fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Output0 ... Output 5	uint8	Ein/Ausschalten der Ausgänge 0 - 5 0: Ausschalten des Ausgangs 1: Einschalten des Ausgangs
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

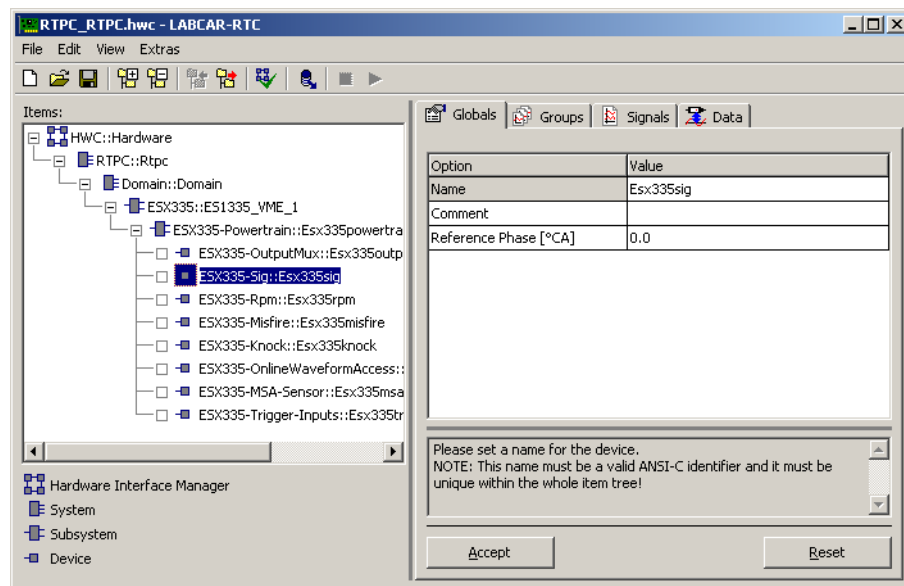
**Tab. 10-3** ESX335-OutputMux Device: Die RTIO-Signale der Registerkarte „Data“

## 10.5 ESX335-Sig Device

Das ESX335-Sig Device dient zur Konfiguration und Ansteuerung eines der sechs arbiträren Signalgeneratoren auf dem ESX335.1 Arbitrary Signal Generator Board.

### 10.5.1 Globals (ESX335-Sig Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-6** Die Registerkarte „Globals“ des ESX335-Sig Device

#### Comment

Für diesen Signalgenerator kann ein beliebiger Kommentar vergeben werden.

#### Hinweis

Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.

#### Reference Phase

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Signalgenerators. Die tatsächliche Phase wird aus der Addition dieses Werts mit dem Signal „PhaseRef“ (siehe „PhaseRef“ auf Seite 253) bestimmt.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Comment	Identifizier	Nein	Benutzerspezifischer Kommentar
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich 0 ... 720 °KW

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-4** ESX335-Sig Device: Konfigurationsparameter der Registerkarte „Globals“

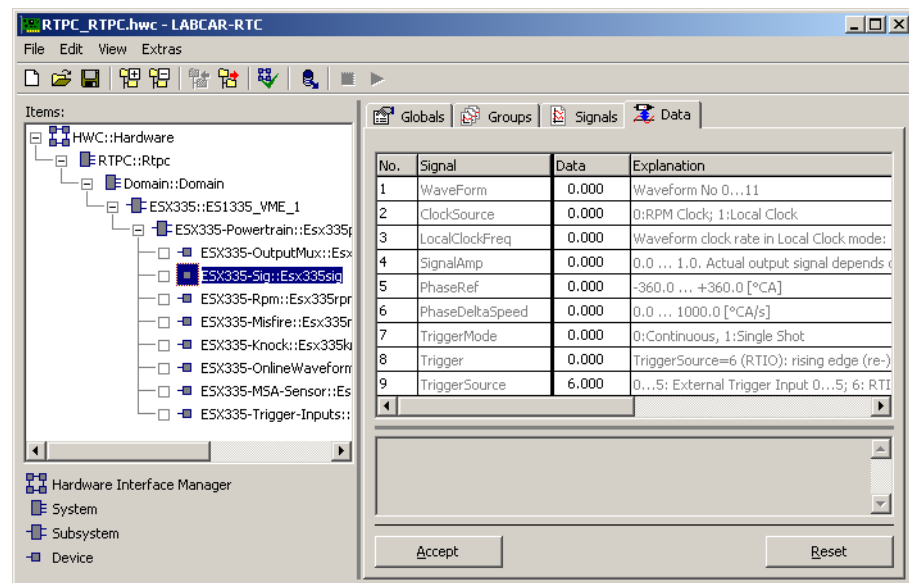
10.5.2 Groups (ESX335-Sig Device)

Das ESX335-Sig Device besitzt eine Signalgruppe, mit der der Signalgenerator gesteuert wird.

10.5.3 Data (ESX335-Sig Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung des Signalgenerators aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 10-7** Die Registerkarte „Data“ des ESX335-Sig Device

*WaveForm*

Über das Signal „WaveForm“ kann die Wellenform ausgewählt werden, die vom Signalgenerator ausgetaktet werden soll. Es sind dabei Werte von 0 bis 11 möglich, die den zentralen Wellenformen im ESX335-Powertrain Subsystem entsprechen.



### *ClockSource*

---

Über das Signal „ClockSource“ wird die Taktquelle für den Signalgenerator bestimmt. Dabei kann zwischen dem zentralen Drehzahlgenerator (RPM-Generator) und einem lokalen Taktgenerator ausgewählt werden. Der Wert 0 entspricht dabei dem zentralen Drehzahlgenerator, der Wert 1 dem lokalen Taktgenerator.

### *LocalClockFreq*

---

Das Signal „LocalClockFreq“ steuert die Frequenz des lokalen Taktgenerators. Das Signal ist nur dann wirksam, wenn das Signal „ClockSource“ den Wert 1 (= lokaler Taktgenerator) besitzt. Die maximale Frequenz des lokalen Taktgenerators beträgt 1 MHz.

### *SignalAmp*

---

Die Ausgangsamplitude des Signalgenerators wird über das Signal „SignalAmp“ gesteuert. Der Wertebereich geht dabei von 0.0 bis 1.0.

Zu beachten ist, dass erst im Ausgangsmultiplexer aus dem internen Signal mit dem Wertebereich -1.0 bis +1.0 ein Spannungssignal gebildet wird.

### *PhaseRef*

---

Dieses Signal beschreibt eine Phasenverschiebung, die zum globalen Parameter „ReferencePhase“ addiert wird. Damit kann eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden. Zu beachten ist, dass die Phasenverschiebung mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeed“) durchgeführt wird. Daher muss das Signal „PhaseDeltaSpeed“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.

### *PhaseDeltaSpeed*

---

Die Phasenänderungsgeschwindigkeit wird mit dem Signal „PhaseDeltaSpeed“ gesteuert. Eine Änderung des Werts von Signal „PhaseRef“ wirkt nicht unendlich schnell, sondern die Phase wird mit dieser Geschwindigkeit verstellt. Der Wertebereich reicht von 0 (es findet keine Verstellung der Phase statt) bis 1000 °KW/s.

### *TriggerMode*

---

#### **Hinweis**

*Das Signal „TriggerMode“ ist nur dann von Bedeutung, wenn der lokale Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist (siehe „ClockSource“ auf Seite 253).*

Der lokale Taktgenerator kann kontinuierlich arbeiten („TriggerMode“ hat den Wert 0) oder er stoppt nach einmaligen Abspielen („Single Shot“) einer kompletten Wellenform („TriggerMode“ hat den Wert 1).

Eine Phasenverschiebung wird auch im Betrieb mit dem lokalen Taktgenerator berücksichtigt. Damit kann der Startpunkt des Abspielens gesteuert werden.

Ein Start bzw. Neustart ist durch das Triggersignal „Trigger“ möglich.

Zu beachten ist, dass immer die komplette Wellenform über die Gesamtanzahl aller Stützstellen abgespielt wird. Wenn die Stützstellenanzahl der Wellenform kleiner als die Winkelauflösung gewählt ist, dann wird das gewünschte Ausgangssignal mehrfach im Wellenformspeicher hintereinander abgelegt. Dann wird beim Abspielen im Modus „Single Shot“ das gewünschte Ausgangssignal mehrfach abgespielt (siehe hierzu „Wellenformauflösung kleiner als Winkelauflösung“ auf Seite 243). Aus diesem Grund ist es für die Anwendung „Single Shot“ in der Regel sinnvoll, die Auflösung ( $=1/\text{Stützstellenanzahl}$ ) der Wellenform gleich der Winkelauflösung zu halten.

### *Trigger*

---

#### **Hinweis**

*Das Signal „Trigger“ ist nur dann von Bedeutung, wenn der lokale Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist (siehe „ClockSource“ auf Seite 253).*

Das Signal „Trigger“ ermöglicht das Starten des lokalen Taktgenerators. Eine Flanke von 0 → 1 auf dem „Trigger“-Signal startet das Austakten der Wellenform mit dem lokalen Taktgenerator. Eine entsprechende Flanke führt in allen Fällen zum Neustart einer Austaktung, auch wenn das vorherige Abspielen noch nicht beendet ist oder auch im kontinuierlichen Betriebsmodus.

### *TriggerSource*

---

#### **Hinweis**

*Das Signal „TriggerSource“ ist nur dann von Bedeutung, wenn der lokale Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist (siehe „ClockSource“ auf Seite 253).*

Mit diesem Signal kann die Triggerung der Signalgeneratoren zwischen Software (LABCAR-RTC) oder über die externen Eingänge des PB1335TRIG-Moduls gewählt werden.

Bei Verwendung von externen Triggern muss „Triggersource“ auf 1 gesetzt werden – gleichzeitig muss der Wert von „LocalClockFreq“ ungleich Null sein.

#### **Hinweis**

*Das Signal „TriggerSource“ ist nur bei neu erzeugten ESX335-Sig Devices vorhanden – bereits existierende Devices werden nicht automatisch erweitert und auch nach einer erneuten Codegenerierung bleibt der Wert intern auf 6 (kein externer Trigger).*

*Wenn Sie mit dem Signal „TriggerSource“ arbeiten wollen, müssen Sie das vorhandene ESX335-Sig Device löschen und erneut einfügen.*

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
WaveForm	uint8	Ja	Auswahl der Wellenform 0 - 11
ClockSource	uint8	Ja	Taktquelle des Signalgenerators 0: Drehzahlgenerator 1: Lokaler Taktgenerator
LocalClockFreq	real32	Ja	Frequenz des lokalen Signalgenerators in Hertz. 0.0 - 1000000.0 [Hz]
SignalAmp	uint16	Ja	Signalamplitude des Ausgangssignals 0...65535 entspricht 0.0...1.0
PhaseRef	sint16	Ja	Additive Phasenverschiebung -32768...32767 entspricht -360...+360 [°KW]
PhaseDeltaSpeed	uint16	Ja	Phasenänderungsgeschwindigkeit 0 ... 1000 [°KW/s]
TriggerMode	uint8	Ja	Trigger-Modus 0: Kontinuierlich 1: Single-Shot
Trigger	uint8	Ja	Triggersignal Flanke von 0 → 1 startet das Abspielen
TriggerSource	uint8	Ja	Triggerquelle Wertebereich: 0..6 0..5 = ext. Triggersignal 6 = LABCAR-RTC (default)
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

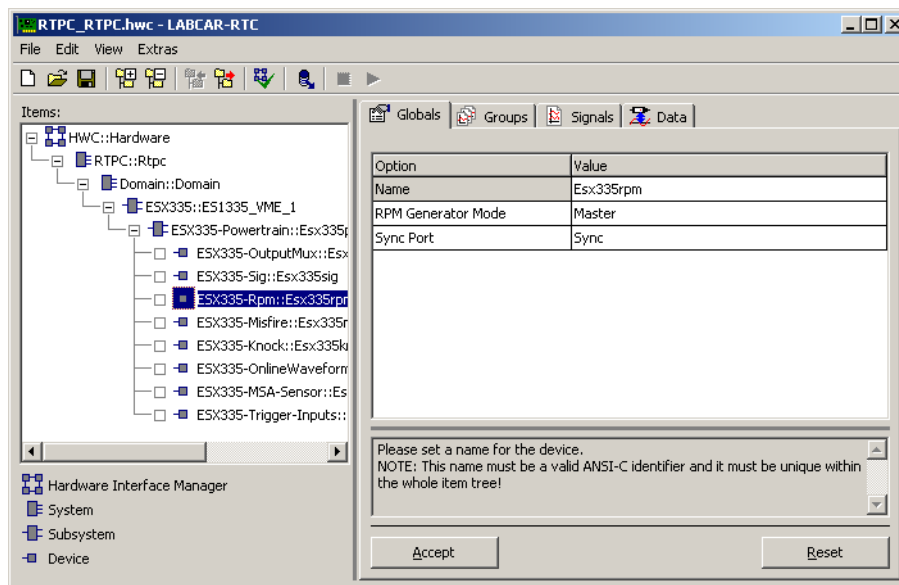
**Tab. 10-5** ESX335-Sig Device: Signale der Registerkarte „Data“

## 10.6 ESX335-Rpm Device

Das ESX335-Rpm Device dient zur Konfiguration und Ansteuerung des Drehzahlgenerators auf dem ESX335.1 Arbitrary Signal Generator Board.

### 10.6.1 Globals (ESX335-Rpm Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-8** Die Registerkarte „Globals“ des ESX335-Rpm Device

#### RPM Generator Mode

Wenn mehrere ESX335.1 Arbitrary Signal Generator Boards in einem VMEbus-System verwendet werden sollen, so ist eine davon im RPM Generator Mode „Master“ zu konfigurieren, alle anderen Karten sind als „Slave“ zu konfigurieren. Die als „Master“ konfigurierte Karte gibt dann die Drehzahl und die Winkelinformation für alle im System befindlichen Karten vor. Bei Nutzung einer einzigen ESX335.1 ist der RPM Generator Mode auf „Master“ zu stellen.

#### Sync Port

Der Anschluss „SYNC“ auf der Frontblende des ESX335.1 Arbitrary Signal Generator Boards kann mit verschiedenen, drehzahlbezogenen Signalen belegt werden. Dies erleichtert die Inbetriebnahme und/oder kann für Diagnosezwecke genutzt werden. Das Ausgangssignal am Anschluss „SYNC“ ist ein 5 V Digitalsignal.

Folgende Einstellungen sind möglich:

- Sync: Eine kurzes High-Signal bei 0 °KW wird ausgegeben.
- Clock - Das Drehzahl-Taktsignal, mit dem die Wellenformen der verbundenen Signalgeneratoren ausgetaktet werden. Sowohl bei einer steigenden als auch bei einer fallenden Flanke des Signals wird die jeweils nächste Stützstelle der Wellenform ausgetaktet.

- Direction: Richtungsanzeige: Ein 5 V Signal entspricht der positiven Drehrichtung des Drehzahlgenerators. Bei einer negativen Drehrichtung liegen 0 V an.
- Engine Speed: Drehzahlsignal - das Signal ist bei einem aktuellen Kurbelwellenwinkel zwischen 0 und 360 °KW auf 5 V, bei einem aktuellen Kurbelwellen-Winkel zwischen 360 und 720 °KW auf 0 V.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
RPM Generator Mode	uint8	Ja	0: Slave, 1: Master
Sync Port	uint8	Ja	Belegung des „SYNC“-Anschlusses: 0: Sync 1: Clock 2: Direction 3: Engine Speed

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

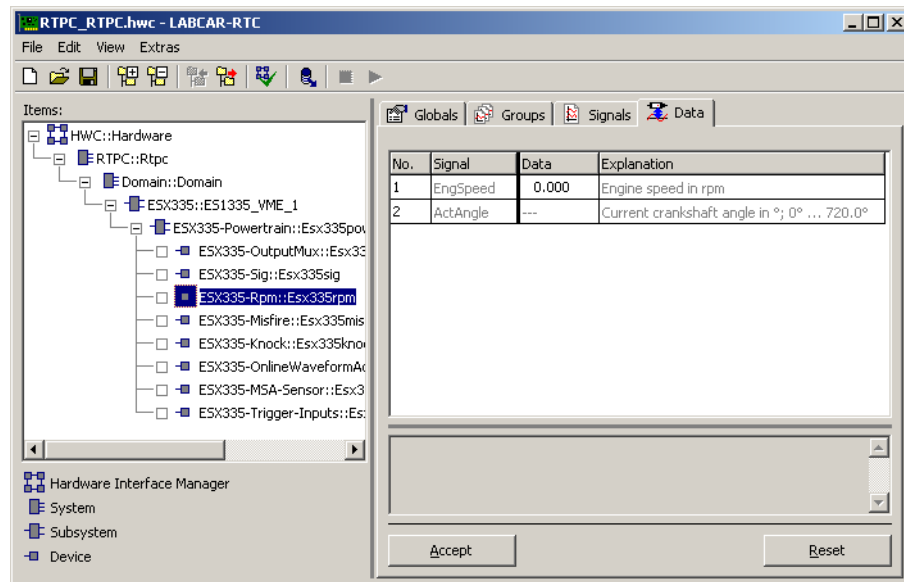
**Tab. 10-6** ESX335-Rpm Device: Konfigurationsparameter der Registerkarte „Globals“

#### 10.6.2 Groups (ESX335-Rpm Device)

Das ESX335-Rpm Device besitzt eine Signalgruppe „Outputs“ zum Setzen der aktuellen Drehzahl und eine Signalgruppe „Inputs“ zum Lesen des aktuellen Kurbelwellenwinkels vom ESX335.1 Arbitrary Signal Generator Board.

## 10.6.3 Data (ESX335-Rpm Device)

In der Registerkarte „Data“ werden die Signale zur Kommunikation mit dem Drehzahlgenerator aufgelistet. Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“.



**Abb. 10-9** Die Registerkarte „Data“ des ESX335-Rpm Device

### EngSpeed

Über das Signal „EngSpeed“ wird die Motordrehzahl, d.h. die Taktfrequenz des Drehzahlgenerators gesteuert. Die Drehzahl wird dabei in Umdrehungen pro Minute (rpm) bis zu einem Maximalwert von 30000 rpm angegeben.

Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.

### ActAngle

Das Signal „ActAngle“ liefert den aktuellen Kurbelwellenwinkel des Drehzahlgenerators - der Wertebereich liegt dabei zwischen 0 und 720 °KW.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
EngSpeed	real32	Ja	Motordrehzahl in [rpm] Wertebereich: -30000 ... +30000 [rpm]
ActAngle	real32	---	Aktueller Kurbelwellenwinkel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

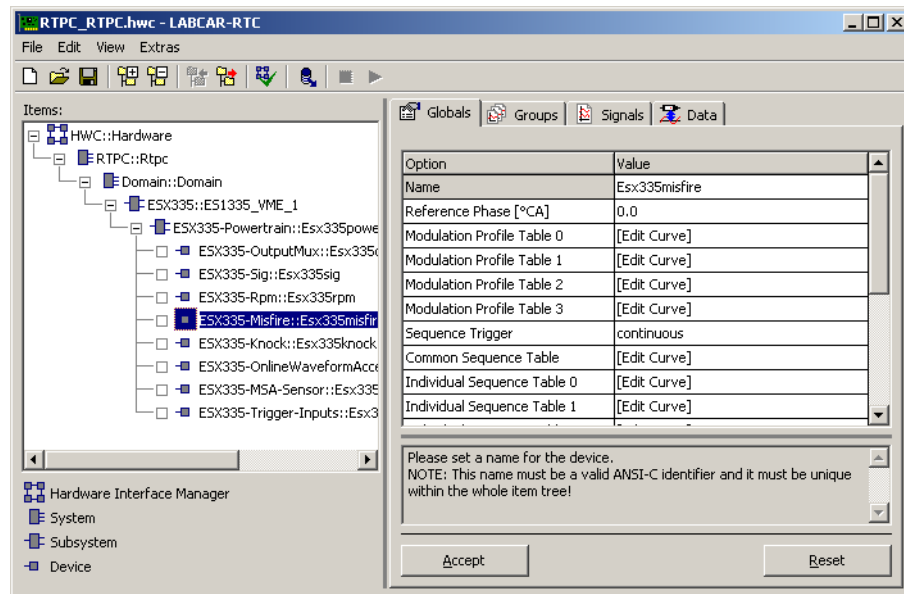
**Tab. 10-7** ESX335-Rpm Device: Konfigurationsparameter der Registerkarte „Data“

## 10.7 ESX335-Misfire Device

Das ESX335-Misfire Device dient zur Konfiguration und Ansteuerung der Zündaussetzsteuerung auf dem ESX335.1 Arbitrary Signal Generator Board.

### 10.7.1 Globals (ESX335-Misfire Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-10** Die Registerkarte „Globals“ des ESX335-Misfire Device

#### Reference Phase

Dieser Parameter beschreibt die Grund-Phasenverschiebung der Zündaussetzsteuerung. Die tatsächliche zylinderspezifische Phase wird aus der Addition dieses Werts mit dem jeweiligen Parameter „Start Angle“ aus der Registerkarte „Signals“ bestimmt.

#### Modulation Profile Table 0 ... 3

Es stehen vier Modulationsprofile zur Verfügung, die eine Modulation der Vorgabedrehzahl über einen bestimmten Winkelbereich vorgeben (siehe „Zündaussetzsteuerung“ auf Seite 245). Die Modulationsprofile können über den Tabelleneditor vorgegeben werden. In einer Modulationskurve ist die Drehzahlmodulation über dem Kurbelwellenwinkel aufgetragen. Dabei entspricht ein Wert von 1 an einem Winkel der unveränderten Übernahme der Vorgabedrehzahl. Ein Wert kleiner 1 (bis 0.01) verringert die Drehzahl. Ein Wert größer 1 entspricht einer Drehzahlerhöhung.

Der erste Stützstellenwert der Signaltabelle entspricht dem jeweiligen zylinder-spezifischem Startwinkel in °KW. Die Signaltabelle beschreibt eine komplette Periode von 720 °KW.

Die Drehzahlmodulation von mehreren Zylindern können sich überlagern - dabei werden die einzelnen zylinderbezogenen Modulationswerte miteinander multipliziert.

*Sequence Trigger*

Die Aktivierung der Drehzahlmodulation kann für jeden Zylinder über Sequenztabelle gesteuert werden (siehe „Sequenztabelle“ auf Seite 245). Über diesen Parameter wird entschieden, ob ein kontinuierliches Abspielen der Sequenztabelle („continuous“) oder ein einmaliges Abspielen („single shot“) der Sequenztabelle gewünscht wird. Das Starten des Abspielens erfolgt über das Signal „Sqnc\_Trigger“ (siehe „Sqnc\_Trigger“ auf Seite 263).

*Common Sequence Table, Individual Sequence Table 0 ... 11*

Zur Steuerung einer Drehzahlmodulation stehen 12 zylinderindividuelle Sequenztabelle zur Verfügung (siehe „Sequenztabelle“ auf Seite 245). Zusätzlich steht eine gemeinsam nutzbare Sequenztabelle „Common Sequence Table“ zur Verfügung, die es ermöglicht, einen Sequenzablauf einfach auf mehreren Zylindern zu nutzen.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich 0 ... 720 °KW
Modulation Profile 0...3	table	Ja	Die Stützstellen werden auf eine komplette Periode (720 °KW) abgebildet. Die Werte liegen innerhalb des Wertebereichs von 0.01 bis 2.0
Sequence Trigger	uint8	Ja	Abspielmodus der Sequenztabelle: Continuous: 0 Single Shot: 1
Common Sequence Table Individual Sequence Table 0...11	table	Ja	Sequenztabelle

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-8** ESX335-Misfire Device: Konfigurationsparameter der Registerkarte „Globals“

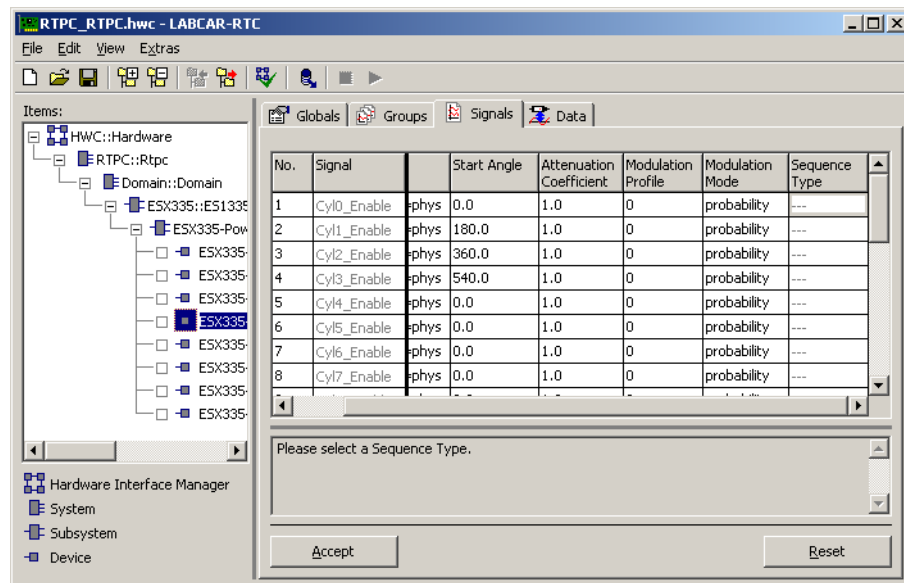


### 10.7.2 Groups (ESX335-Misfire Device)

Das ESX335-Misfire Device besitzt eine Signalgruppe, in der die Signale der Zündaussetzsteuerung verarbeitet werden.

### 10.7.3 Signals (ESX335-Misfire Device)

In der Registerkarte „Signals“ wird die Konfiguration der Zündaussetzsteuerung durchgeführt. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 10-11** Die Registerkarte „Signals“ des ESX335-Misfire Device

#### *Start Angle*

Das Optionsfeld „Start Angle“ dient zur Festlegung des Startwinkels des Zündaussetzer-Modulationsprofils für den jeweiligen Zylinder. Zu diesem Winkel wird der Parameter „Reference Phase“ aus der Registerkarte „Globals“ (siehe „Reference Phase“ auf Seite 259) addiert.

#### *Attenuation Coefficient*

Über diesen Parameter kann die Stärke der Drehzahlmodulation zylinderindividuell reduziert werden. Ein Wert von 1.0 bedeutet, dass die im ausgewählten Modulationsprofil beschriebene Drehzahlmodulation vollständig angewendet wird. Ein Wert kleiner 1 reduziert die Wirkung anteilig, ein Wert von 0.0 schließlich führt dazu, dass für den Zylinder keine Drehzahlmodulation durchgeführt wird.

#### *Modulation Profile*

Pro Zylinder kann eines der vier vorhandenen Modulationsprofile der Registerkarte „Globals“ ausgewählt werden.

### Modulation Mode

Das Vorhandensein eines Zündaussetzers in einer Periode kann entweder über eine Wahrscheinlichkeitsfunktion oder über die Sequenztabelle gesteuert werden. Über diesen Parameter kann zwischen diesen beiden Betriebsarten umgeschaltet werden.

### Sequence Type

Bei Auswahl des „Modulation Mode = sequence“ kann hier festgelegt werden, ob die gemeinsam nutzbare Sequenztabelle („common“) oder die zylinder-spezifische Sequenztabelle („individual“) verwendet werden soll (siehe auch „Sequenztabelle“ auf Seite 245).

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Start Angle	real32	Ja	Startwinkel der Drehzahlmodulation 0.0 ... 720.0 °KW
Attenuation Coefficient	real32	Ja	Anteilige Wirkung des Modulationsprofils 0.0 ... 1.0
Modulation Profile	uint8	Ja	Auswahl des Modulationsprofils für den Zylinder 0 ... 3
Modulation Mode	uint8	Ja	Wahrscheinlichkeit oder Sequenztabelle für die Steuerung der Zündaussetzung 0: Wahrscheinlichkeit 1: Sequenztabelle
Sequence Type	uint8	Ja	Auswahl: Gemeinsame oder zylinderindividuelle Sequenz. 0: Gemeinsame Sequenz 1: Zylinderindividuelle Sequenz

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

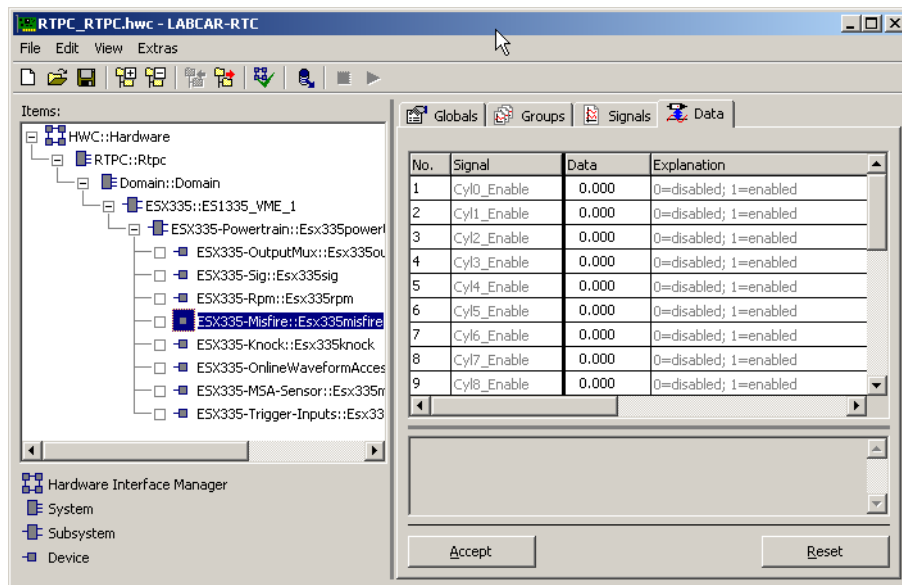
**Tab. 10-9** ESX335-Misfire Device: Konfigurationsparameter der Registerkarte „Signals“

#### 10.7.4 Data (ESX335-Misfire Device)

In der Registerkarte „Data“ werden die RTIO-Signale dargestellt, die zur Steuerung der Zündaussetzeinheit vorhanden sind.

#### Hinweis

*Beachten Sie bitte, dass der Parameter „Served Cylinders“ (siehe „Served Cylinders“ auf Seite 247) aus dem übergeordneten ESX335-Powertrain Subsystem korrekt gesetzt ist, da dieser Wert bei der Verarbeitung der Signale verwendet wird!*



**Abb. 10-12** Die Registerkarte „Data“ des ESX335-Misfire Device

#### *Cyl0\_Enable ... Cyl11\_Enable*

Diese zwölf Signale aktivieren die Drehzahlmodulation für den jeweiligen Zylinder.

#### *Cyl0\_Probability ... Cyl11\_Probability*

Über diese Signale wird die Wahrscheinlichkeit für das Auftreten einer Fehlzündung für einen Zylinder gesteuert. Das Signal wird nur berücksichtigt, wenn der entsprechende Zylinder zur Nutzung der Wahrscheinlichkeitsfunktion konfiguriert wurde („Modulation Mode“ auf Seite 262).

#### *Sqnc\_Trigger*

Dieses Signal stellt den Trigger dar, mit dem das Abspielen der Sequenztabellen gestartet wird. Eine steigende Flanke von 0 → 1 auf diesem Signal startet das Abspielen. Eine noch nicht vollständig abgelaufene Sequenz wird durch eine steigende Flanke auf dem Triggersignal abgebrochen und neu gestartet (re-trigger).

Die folgende Tabelle fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Cyl0_Enable ... Cyl11_Enable	uint8	Ein- / Ausschalten der Ausgänge 0: Ausschalten des Ausgangs 1: Einschalten des Ausgangs
Cyl0_Probability .. Cyl11_Probability	uint16	Wahrscheinlichkeit für einen Zündaussetzer Wertebereich: 0 bis 10000 entspricht 0 ... 1.0
Sqnc_Trigger	uint8	Starten bzw. Neustarten des Abspielens der Sequenztabellen Eine Flanke 0 → 1 dient als Trigger

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

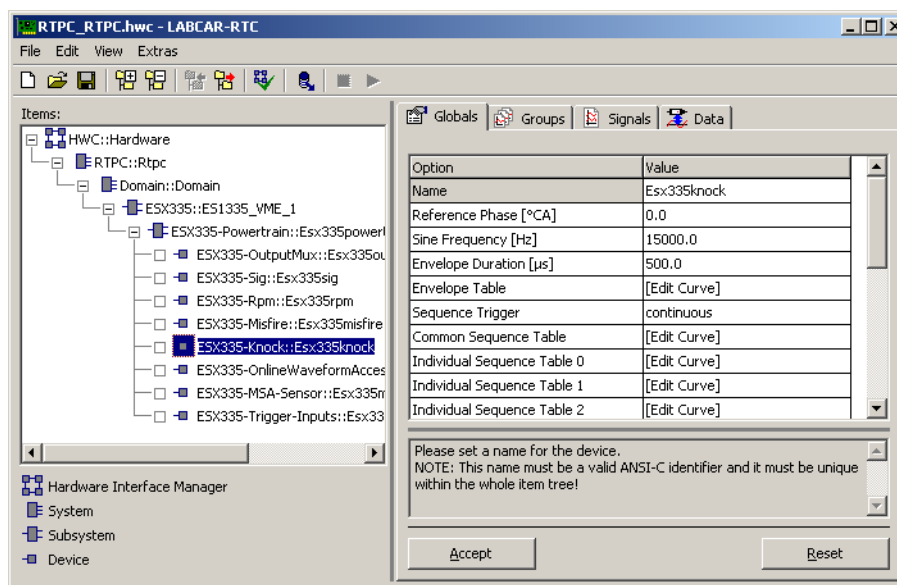
**Tab. 10-10** ESX335-Misfire Device: Die RTIO-Signale der Registerkarte „Data“

## 10.8 ESX335-Knock Device

Das ESX335-Knock Device dient zur Konfiguration und Ansteuerung des Klopf-signalgenerators auf dem ESX335.1 Arbitrary Signal Generator Board.

### 10.8.1 Globals (ESX335-Knock Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-13** Die Registerkarte „Globals“ des ESX335-Knock Device

#### Reference Phase

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Klopf-signalgenerators. Die tatsächliche zylinderspezifische Phase wird aus der Addition dieses Werts mit dem jeweiligen Parameter „Base Angle“ aus der Registerkarte „Signals“ (siehe „Base Angle“ auf Seite 267) und dem Signal „Cyln\_AngleOffset“ (siehe „Cyl0\_AngleOffset ... Cyl11\_AngleOffset“ auf Seite 270) bestimmt.

### *Sine Frequency*

Dieser Parameter beschreibt die Frequenz des Sinussignals, aus dem das Klopfsignal besteht. Dieses Sinussignal wird mit einer Hüllkurve moduliert.

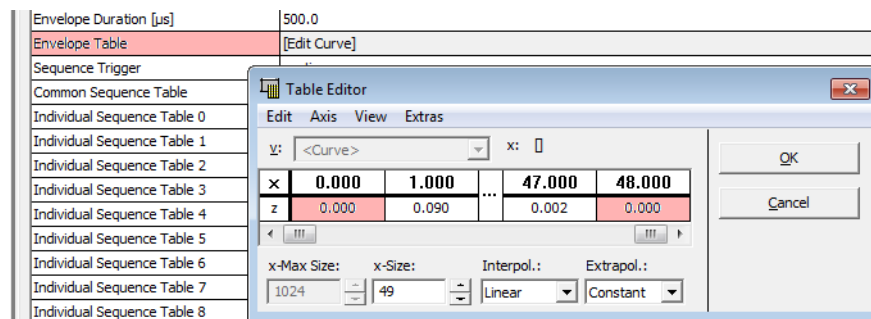
### *Envelope Duration*

Dieser Parameter beschreibt die Dauer eines Klopfsignals in  $\mu\text{s}$ .

### *Envelope Table*

Die Form der Hüllkurve wird durch diese Tabelle beschrieben. Die Länge der Hüllkurve wird durch den Parameter „Envelope Duration“ vorgegeben (siehe hierzu „Klopfsignalgenerator“ auf Seite 244).

Hüllkurven müssen immer mit dem Wert „0“ beginnen und mit dem Wert „0“ enden.



### *Sequence Trigger*

Mit diesem Parameter kann zwischen „Single-Shot“ und der kontinuierlichen Triggerung der Sequenztabelle ausgewählt werden. Siehe hierzu „Sequenztabelle“ auf Seite 245.

### *Common Sequence Table, Individual Sequence Table 0 ... 11*

Es stehen 12 zylinderindividuelle Sequenztabelle zur Steuerung der Erzeugung von Klopfpaketen zur Verfügung (siehe „Sequenztabelle“ auf Seite 245). Zusätzlich steht eine gemeinsam nutzbare Sequenztabelle „Common Sequence Table“ zur Verfügung, die es ermöglicht, einen Sequenzablauf einfach auf mehreren Zylindern zu nutzen.

### *Cylinder to Sensor Mapping Table 0 ... 3*

Der Klopfsignalgenerator verfügt über vier Ausgänge. In jeder der vier „Cylinder to Sensor Mapping“ Zuordnungstabellen wird für den jeweiligen Ausgang definiert, von welchen Zylindern die generierten Klopfsignale ausgehen werden. Damit besteht die Möglichkeit, einen Zylinder auf mehrere interne Ausgänge auszugeben, indem man den entsprechenden Zylinder mehrfach in der Zuordnungstabelle auswählt.

Die Tabellen bestehen jeweils aus bis zu 12 Stützstellen, die die Zylinder 0 bis 11 repräsentieren. Ein Wert von 1.0 an einer Stützstelle entspricht der Zuordnung des jeweiligen Zylinders an den entsprechenden Ausgang. Ein Wert von 0.0 entspricht einer nicht vorhandenen Zuordnung des Zylinders an den Ausgang (siehe auch „Sequenztabelle“ auf Seite 245).

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich 0 ... 720 °KW
Sine Frequency	real32	Ja	Sinusfrequenz der Klopfpakete im Bereich 1000 - 20000 Hz
Envelope Duration	real32	Ja	Dauer der Hüllkurve in µs. Wertebereich: 50-12000µs
Envelope Table	table	Ja	Hüllkurvensignal
Sequence Trigger	uint8	Ja	Auswahl des Triggermodus für die Sequenztabelle. 0: Kontinuierlich 1: Single Shot
Common Sequence Table	table	Ja	Für alle Zylinder gemeinsam nutzbare Sequenztabelle
Individual Sequence Table 0 ... 11	table	Ja	Zylinderindividuelle Sequenztabelle
Cylinder to Sensor Mapping Table 0 ... 3	table	Ja	Tabellen, in denen einem Ausgang die zugehörigen Zylinder zugeordnet werden können. Die Stützstellen entsprechen den 12 Zylindern - ein Wert von größer 0.5 bedeutet, dass der entsprechende Zylinder auf dem Ausgang das Klopfsignal ausgibt.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

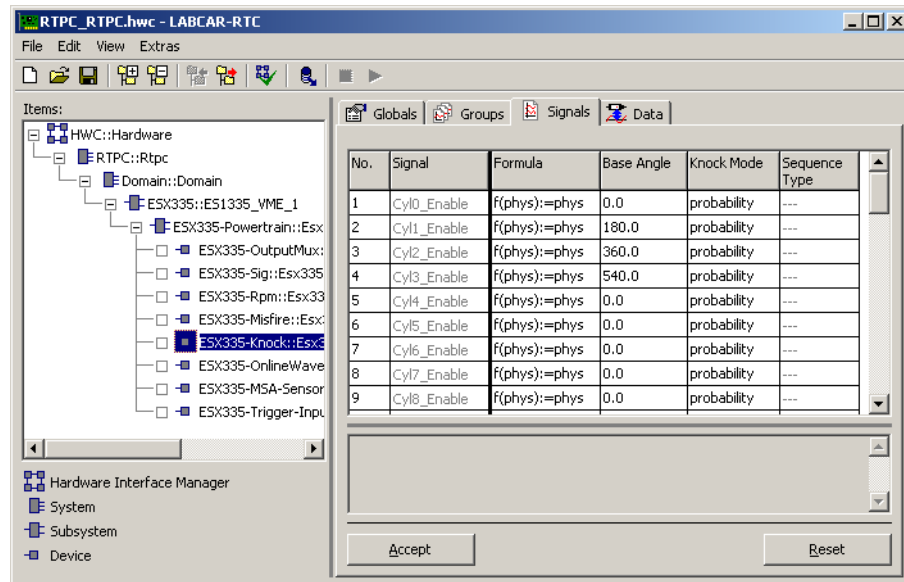
**Tab. 10-11** ESX335-Knock Device: Konfigurationsparameter der Registerkarte „Globals“

#### 10.8.2 Groups (ESX335-Knock Device)

Das ESX335-Knock Device besitzt eine Signalgruppe, in der die Signale der Zündaussetzsteuerung verarbeitet werden.

## 10.8.3 Signals (ESX335-Knock Device)

In der Registerkarte „Signals“ wird die Konfiguration des Klopfgenerators durchgeführt. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 10-14** Die Registerkarte „Signals“ des ESX335-Knock Device

#### Base Angle

Das Optionsfeld „Base Angle“ dient zusammen mit dem Parameter „Reference Phase“ aus der Registerkarte „Globals“ als Grundwinkel zur Steuerung des Startwinkels eines Klopfpakets. Zu diesem Startwinkel wird noch das zylinderspezifische Signal „Cylx\_AngleOffset“ addiert, das eine Veränderung zur Laufzeit ermöglicht. Zu beachten ist, dass das Signal „Cylx\_AngleOffset“ nur über einen Winkelbereich von  $\pm 127$  °KW variiert werden kann.

#### Knock Mode

Das Auftreten von Klopfereignissen in einer Periode von 720 °KW kann entweder über eine Wahrscheinlichkeitsfunktion oder über die Sequenztabellen gesteuert werden. Über diesen Parameter kann zwischen diesen beiden Betriebsarten umgeschaltet werden.

### Sequence Type

Bei Auswahl des „Modulation Mode = sequence“ kann hier ausgewählt werden, ob die gemeinsam nutzbare Sequenztabelle „common“ oder die zylinderindividuelle Sequenztabelle („individual“) verwendet werden soll (siehe auch „Sequenztabellen“ auf Seite 245).

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Base Angle	real32	Ja	Basisausgabewinkel für Klopfpakete 0.0 ... 720.0 °KW
Knock Mode	uint8	Ja	Wahrscheinlichkeit oder Sequenztabelle für die Ausgabe von Klopfpakete 0: Wahrscheinlichkeit 1: Sequenztabelle
Sequence Type	uint8	Ja	Auswahl: Gemeinsame oder zylinderindividuelle Sequenz 0: Gemeinsame Sequenz 1: Zylinderindividuelle Sequenz

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-12** ESX335-Knock Device: Konfigurationsparameter der Registerkarte „Signals“

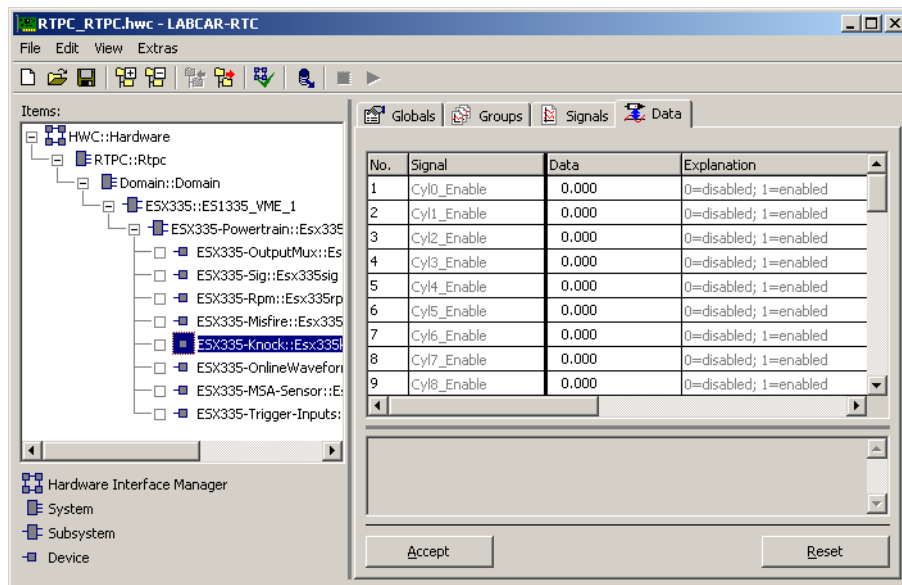
#### 10.8.4 Data (ESX335-Knock Device)

In der vorhandenen Signalgruppe werden RTIO-Signale dargestellt, die zur Steuerung des Klopfsignalgenerators vorhanden sind.

#### Hinweis

*Beachten Sie bitte, dass der Parameter „Served Cylinders“ im übergeordneten ESX335-Powertrain Subsystem („Served Cylinders“ auf Seite 247) korrekt gesetzt ist, da dieser Wert bei der Verarbeitung der Signale verwendet wird!*





**Abb. 10-15** ESX335-Knock Device: Konfigurationsparameter der Registerkarte „Data“

#### *Cyl0\_Enable ... Cyl11\_Enable*

Diese zwölf Signale aktivieren die Generierung von Klopfpaketen für unkontrollierte Verbrennung für die jeweiligen Zylinder.

#### *Cyl0\_Probability ... Cyl11\_Probability*

Über diese Signale wird die Wahrscheinlichkeit für das Auftreten einer klopfenden Verbrennung für einen Zylinder gesteuert. Das Signal wird nur berücksichtigt, wenn der entsprechende Zylinder zur Nutzung der Wahrscheinlichkeitsfunktion konfiguriert wurde.

#### *Sqnc\_Trigger*

Dieses Signal bildet den Trigger, mit dem das Abspielen der Sequenztabellen gestartet werden kann. Eine steigende Flanke von 0 → 1 auf diesem Signal startet das Abspielen. Eine noch nicht vollständig abgelaufene Sequenz wird durch eine steigende Flanke auf dem Triggersignal abgebrochen und neu gestartet (re-trigger).

#### *NoiseAmpl*

Dieses Signal steuert eine stochastische Variation der Amplituden der Klopfpakete. Dabei findet bei einem Wert von 0.0 keine Variation der Amplitude statt, bei einem Wert von 0.1 kann die Amplitude um 10% der maximalen Ausgangsspannung schwanken.

Diese Variation wirkt auf die Ausgabe aller Klopfpakete (klopfend und nicht klopfend).

NoKnockAmpl

Bei einer Verbrennung ohne aufgetretenes Klopfen können dennoch Klopfpakete ausgegeben werden. Die Amplitude dieser Klopfpakete werden mit diesem Signal beschreiben. Typischerweise wird die Amplitude von „NoKnockAmpl“ kleiner als die Amplituden bei einem aufgetretenen Klopfereignis gewählt.

Cyl0\_KnockAmpl ... Cyl11\_KnockAmpl

Über dieser 12 Signale können die zylinderspezifischen Amplituden für eine klopfende Verbrennung angegeben werden.

Cyl0\_AngleOffset ... Cyl11\_AngleOffset

Der Ausgabewinkel der Klopfpakete kann durch dieses Signal zylinderindividuell um  $\pm 127$  °KW in Schritten von 1 °KW variiert werden. Als Basiswinkel dient die Summe von „Reference Phase“ (Registerkarte „Globals“) und des jeweiligen „Base Angles“ (Registerkarte „Signals“).

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Cyl0_Enable ... Cyl11_Enable	uint8	Steuerung der Klopfsignale für die einzelnen Zylinder 0: Klopfpakete für Klopfen möglich 1: Klopfpakete für Klopfen nicht möglich
Cyl0_Probability .. Cyl11_Probability	uint16	Wahrscheinlichkeit für Klopfen Wertebereich: 0 bis 10000 (entspricht 0.0 bis 1.0)
Sqnc_Trigger	uint8	Starten bzw. Neustarten des Abspielens der Sequenztabellen Eine Flanke 0 → 1 dient als Trigger
NoiseAmpl	uint16	Stochastische Amplitudenvariation Wertebereich 0 bis 10000 (entspricht 0.0 bis 1.0)
NoKnockAmpl	uint16	Amplitude bei nicht-klopfender Verbrennung Wertebereich 0 bis 10000 (entspricht 0.0 bis 1.0)
Cyl0_KnockAmpl ... Cyl11_KnockAmpl	uint16	Amplitude bei klopfender Verbrennung Wertebereich 0 bis 10000 (entspricht 0.0 bis 1.0)
Cyl0_AngleOffset ... Cyl11_AngleOffset	sint8	Variation des Klopfwinkels Wertebereich -127 bis +127 (Auflösung 1 °KW)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-13** ESX335-Knock Device: Die RTIO-Signale der Registerkarte „Data“

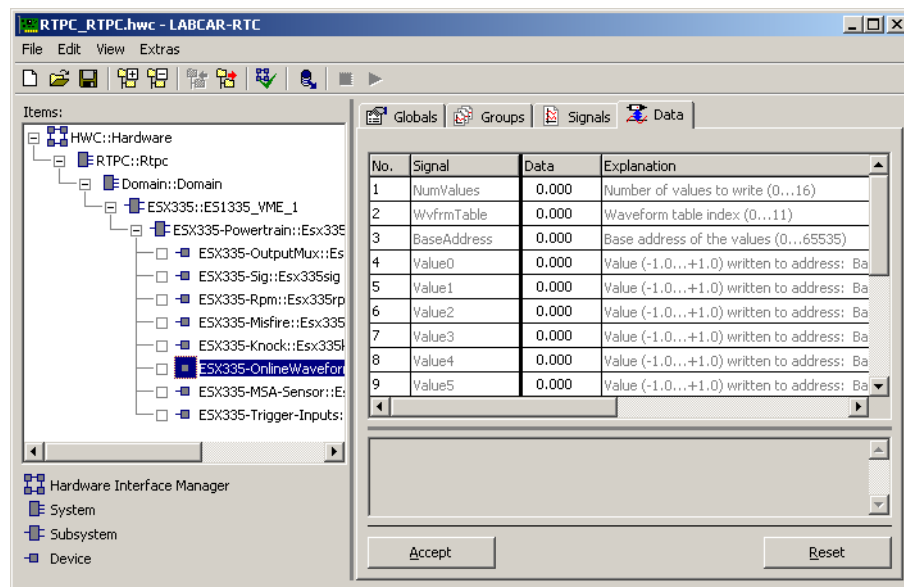
## 10.9 Direkter Wellenformzugriff: ESX335-OnlineWaveformAccess Device

Über das ESX335-OnlineWaveformAccess Device können in der laufenden Simulation Wellenformen verändert werden.

Es gibt keinerlei Konfigurationsparameter, es werden ausschließlich Signale zur Veränderung der Wellenformen verwendet.

### 10.9.1 Data (ESX335-OnlineWaveformAccess Device)

In der Signalgruppe „Ctrl“ werden RTIO-Signale zum Zugriff auf die Wellenformen verwendet.



**Abb. 10-16** ESX335-OnlineWaveformAccess Device: Signale der Registerkarte „Data“

#### *NumValues*

Dieses Signal gibt an, wieviel Datenwerte in einer Wellenform geändert werden sollen. Ein Wert von 0 bedeutet, dass keine Änderung an einer Wellenform stattfinden soll. Maximal 16 Werte können pro Simulationsschritt geändert werden.

#### *WvfrmTable*

Die zu verändernde Wellenform wird über dieses Signal bestimmt. Mögliche Werte sind dabei 0 bis 11 für die zwölf Wellenformen, die im RTIO-Element „ESX335-Powertrain“ definiert sind.

#### *BaseAddress*

Dieses Signal gibt die erste Stützstelle innerhalb der ausgewählten Wellenform an, deren Wert geändert werden soll. Der Wertebereich ist 0 bis 65535, wobei zu beachten ist, dass ausschließlich Werte ausgegeben werden können, die innerhalb der globalen Winkelauflösung (Parameter „Resolution“ im RTIO-Element ESX335-Powertrain) liegen.

Value0 ... Value15

Diese 16 Werte werden - beginnend mit „BaseAddress“ - an „NumValues“ aufsteigenden Stützstellen in die Wellenform „WvfrmTable“ geschrieben.

Die folgende Tabelle fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
NumValues	uint8	Anzahl der zu verändernden Werte 0...16
WvfrmTable	uint8	Die zu verändernde Wellenform 0 ... 11
BaseAddress	uint16	Erste Stützstelle in der zu verändernden Wellenform 0 ... 65535
Value0 ... Value15	sint16	Datenwerte die in die Wellenform zu schreiben sind. -32768 ... 32767

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-14** ESX335-DirectWaveformAccess Device: Die RTIO-Signale der Registerkarte „Data“

## 10.10 ESX335-MSA-Sensor Device

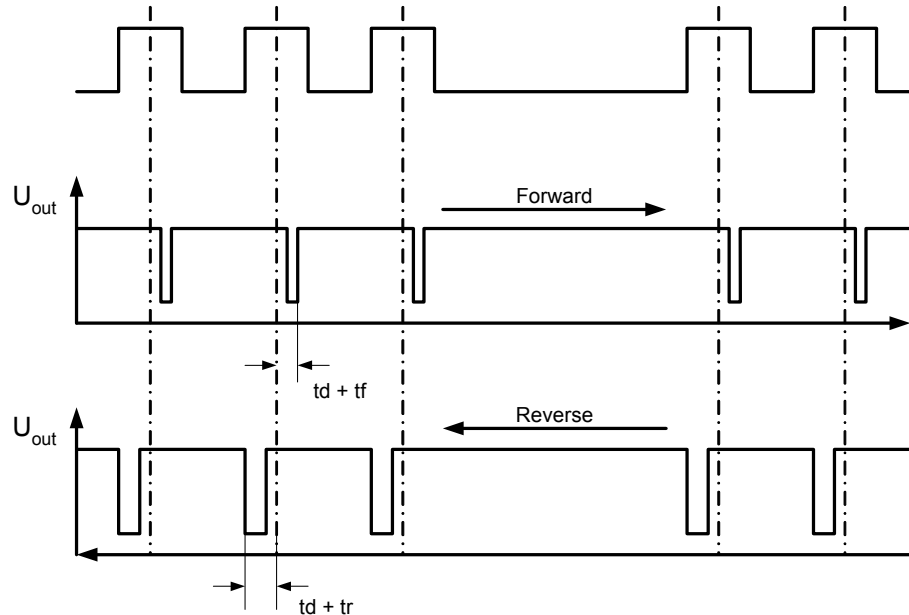
Das ESX335-MSA-Sensor Device dient zur Konfiguration und Ansteuerung eines der sechs Signalgeneratoren auf dem ESX335.1 Arbitrary Signal Generator Board zur Simulation eines MSA-Kurbelwellensensors mit Drehrichtungserkennung.

Da die ausgegebenen Pulse eine feste Zeitdauer haben und somit nicht durch Austakten einer Signaltabelle erzeugt werden können, wurde zur Darstellung eines MSA-Sensors ein spezielles Device eingeführt. Im Vergleich zu einem ESX335-Sig Device wurde die Registerkarte „Globals“ um Parameter zur Spezifikation des Sensors erweitert.

### 10.10.1 Grundsätzliche Funktionsweise

Die MSA-Sensor-Simulation ist in der Lage, dieselben Wellenverläufe des Zahnsignals wie die bisher verwendeten (Hall-)Sensoren zu verwenden. Zur Darstellung eines MSA-Sensors ist jedoch die Kenntnis der Winkelposition der Mitte eines jeden Zahnes des Kurbelwellenrades erforderlich. Für jeden Zahn wird bei der Zahnmitte ein Impuls fester Länge ausgegeben – im realen Sensor werden die Zahnmitten elektronisch ermittelt.

In der Konfigurationsphase werden diese Winkelpositionen aus dem Verlauf der Wellenform berechnet und im Wellenformspeicher abgelegt. Da die Wellenform mittels des Signals "Waveform" in Echtzeit aus einer der vorgegebenen 12 Wellenformen ausgewählt werden kann, müssen alle Wellenformen diesem Verfahren unterzogen werden.



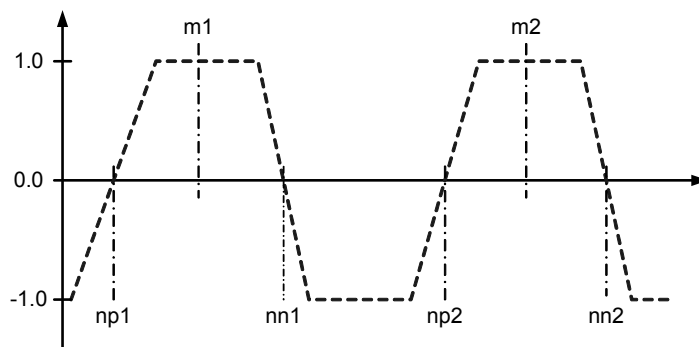
**Abb. 10-17** Kurbelwellenzähne und Sensorsignale bei einem MSA-Sensor

#### 10.10.2 Zahnmittenberechnung

Dazu wird zunächst bei allen bereits zwischen -1.0 und 1.0 normierten Wellenformverläufen jeweils die Positionen der negativ und der positiv verlaufenden Nulldurchgänge ermittelt.

Bei positiver Logik definiert sich ein Zahn als Bereich zwischen einem positiven und dem folgenden negativen Nulldurchgang (siehe Abb. 10-18). Die Zahnmitte ist dementsprechend die arithmetische Mittel der beiden Positionen (bei negativer Logik entsprechend umgekehrt):

Es seien np1, np2 die von unten nach oben schneidenden (positive) Nulldurchgänge und nn1, nn2 die von oben nach unten schneidenden (negative) Nulldurchgänge. Dann ergeben sich die Zahnmittenpositionen m1, m2 zu  $m1 = (np1 + nn1) / 2$  und  $m2 = (np2 + nn2) / 2$ .

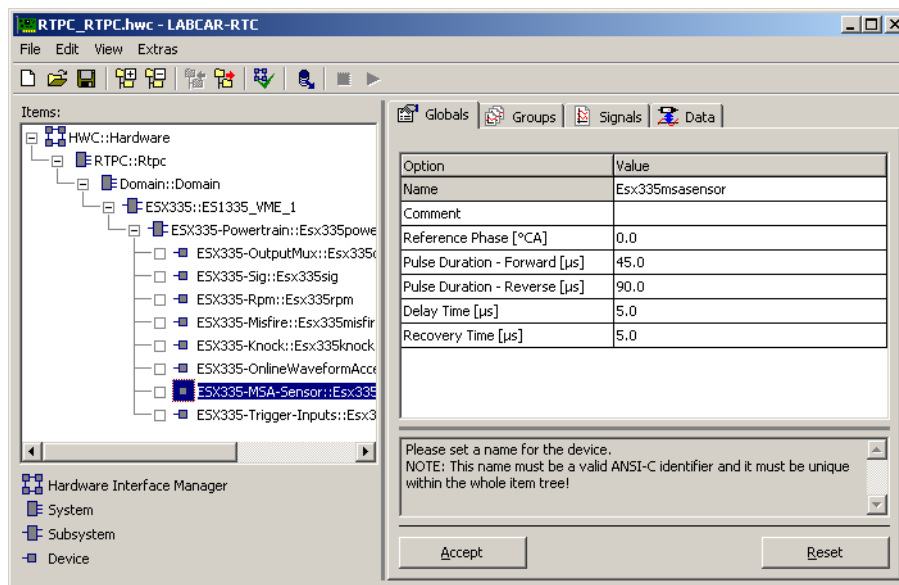


**Abb. 10-18** Mittenpositionsberechnung (bei nicht ganz symmetrischem Zahnverlauf und positiver Logik)

Die Einstellung der Logik erfolgt für alle Kurvenformen gemeinsam in der Registerkarte „Globals“ des ESX335-Powertrain Devices (siehe „MSA Sensors: Logic Level of Teeth“ auf Seite 248).

10.10.3 Globals (ESX335-MSA-Sensor Device)

Das ESX335-MSA-Sensor Device dient zur Konfiguration und Ansteuerung eines der Signalgeneratoren auf dem ESX335.1 Arbitrary Signal Generator Board. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 10-19** Die Registerkarte „Globals“ des ESX335-MSA-Sensor Device  
Die Bedeutung der Optionen wird im Folgenden beschrieben.

### *Comment*

---

Ein beliebiger Kommentar für diesen Signalgenerator.

### **Hinweis**

*Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.*

### *Reference Phase [° CA]*

---

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Signalgenerators. Die tatsächliche Phase wird aus der Addition dieses Werts zu dem Signal „PhaseRef“ ermittelt.

### *Pulse Duration Forward [μs]*

---

Dieser Parameter definiert die Pulsdauer eines Zahnes bei positiver Drehrichtung (vorwärts) ( $t_f$  in Abb. 10-17 auf Seite 273).

### *Pulse Duration Reverse [μs]*

---

Dieser Parameter definiert die Pulsdauer eines Zahnes bei negativer Drehrichtung (rückwärts) ( $t_r$  in Abb. 10-17 auf Seite 273).

### *Delay Time [μs]*

---

Dieser Parameter definiert die Länge der Verzögerungszeit ( $t_d$  in Abb. 10-17 auf Seite 273).

### *Recovery Time [μs]*

---

Dieser Parameter definiert die Länge eines (in bestimmten Situationen der Drehrichtungsumkehr) erzwungenen Impulses.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Comment	Identifizier	Nein	Benutzerspezifischer Kommentar
Reference Phase	real32	Ja	Grund-Phasenverschiebung in ° KW, Wertebereich 0...720 °KW
Pulse Duration - Forward	real32	Ja	Pulsdauer $t_f$ bei Vorwärtsrotation, Wertebereich 10.0...1000.0 $\mu$ s
Pulse Duration - Reverse	real32	Ja	Pulsdauer $t_r$ bei Rückwärtsrotation, Wertebereich 10.0...1000.0 $\mu$ s
Delay Time	real32	Ja	Verzögerungszeit $t_d$ gemäß Sensorspezifikation Wertebereich 1.0...100.0 $\mu$ s
Recovery Time	real32	Ja	Pulsdauer $t_v$ gemäß Sensorspezifikation, Wertebereich 1.0...100.0 $\mu$ s

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-15** ESX335-MSA-Sensor: Konfigurationsparameter der Registerkarte „Globals“

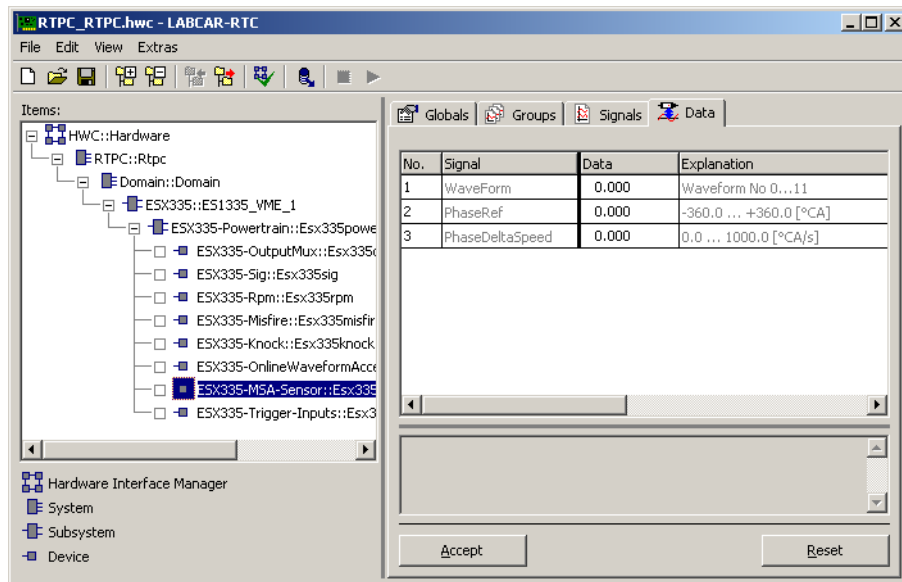
#### 10.10.4 Groups (ESX335-MSA-Sensor Device)

Das ESX335-MSA-Sensor Device besitzt eine Signalgruppe, mit der der MSA-Generator gesteuert wird.



### 10.10.5 Data (ESX335-MSA-Sensor Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung des Signalgenerators aufgelistet.



**Abb. 10-20** Die Registerkarte „Data“ des ESX335-MSA-Sensor Device

Die Bedeutung der Signale wird im Folgenden beschrieben.

#### *WaveForm*

Über das Signal „WaveForm“ wird die Wellenform ausgewählt, die vom Signalgenerator ausgetaktet werden soll. Es sind dabei Werte von 0 bis 11 möglich, die den zentralen Wellenformspeichern im ESX335-Powertrain Device entsprechen.

#### *PhaseRef*

Dieses Signal beschreibt eine Phasenverschiebung, die zum globalen Parameter „ReferencePhase“ addiert wird. Damit kann eine modellgesteuerte Phasenverschiebung zur Laufzeit generiert werden.

Zu beachten ist, dass eine Änderung der Phasenverschiebung mit einer endlichen Geschwindigkeit (gesteuert über den Parameter „PhaseDeltaSpeed“) durchgeführt wird – daher muss das Signal „PhaseDeltaSpeed“ für die meisten Anwendung einen Wert  $> 0$  aufweisen.

PhaseDeltaSpeed

Die Änderungsgeschwindigkeit der Phasenverschiebung wird mit dem Signal „PhaseDeltaSpeed“ gesteuert, da eine Änderung des Werts von „PhaseRef“ nur mit endlicher Geschwindigkeit vonstatten geht. Der Wertebereich reicht von 0 (d.h. es findet keine Verstellung der Phase statt) bis zu 1000° KW/s.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
WaveForm	uint8	Ja	Auswahl der Wellenform 0-11
PhaseRef	sint16	Ja	Additive Phasenverschiebung, Wertebereich: 32768...32767 (entspricht -360...+360 [° KW])
PhaseDeltaSpeed	uint16	Ja	Phasenänderungsgeschwindigkeit 0...1000 [° KW/s]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 10-16** ESX335-MSA-Sensor: Signale der Registerkarte „Data“

## 10.10.6 Fehlerbehandlung

Die Ausgabe einer Fehlermeldung bei ungeeigneten Wellenformverläufen erfolgt erst dann, wenn eine solche Wellenform tatsächlich als Basis für die Ausgabe eines MSA-Sensors benutzt wird.

Eine solche Fehlermeldung hat folgende Form:

```
"Waveform <x> not suitable for MSA sensor implemented
by SigGen #<y> because <error> No signal generated"
```

Dabei bedeutet <x> die Nummer der ungeeigneten Wellenform 0-11 und #<y> die Instanznummer des verwendeten Signalgenerators 0...5.

Eine Beschreibung der Fehlermeldungen <error> finden Sie in der folgenden Tabelle:

Fehlermeldung <error>	Beschreibung
...too many zero crossings (> 2048)...	Es wurden zu viele Nulldurchgänge detektiert, um eine sinnvolle Ausgabe für einen MSA-Sensor zu erzeugen.
...number of positive and negative zero crossings do not match...	Es wurde eine unterschiedliche Anzahl von positiven und negativen Nulldurchgängen ermittelt. Kurvenform ist nicht zur Darstellung eines MSA-Sensors geeignet.
...too few zero crossings (< 2)...	Es wurden zu wenige Nulldurchgänge detektiert, um eine sinnvolle Ausgabe für einen MSA-Sensor darzustellen.
...minimum distance between some tooth centers < 0.4 °CA...	Mindestens zwei der errechneten Zahnmittenspositionen liegt zu dicht nebeneinander, um von der Ausgabereinheit sicher verarbeitet werden zu können. Es ist Mindestabstand der Stützstellen von etwa 0,4 °KW erforderlich.

**Tab. 10-17** Liste der möglichen Fehlermeldungen bei ungeeigneten Wellenformen

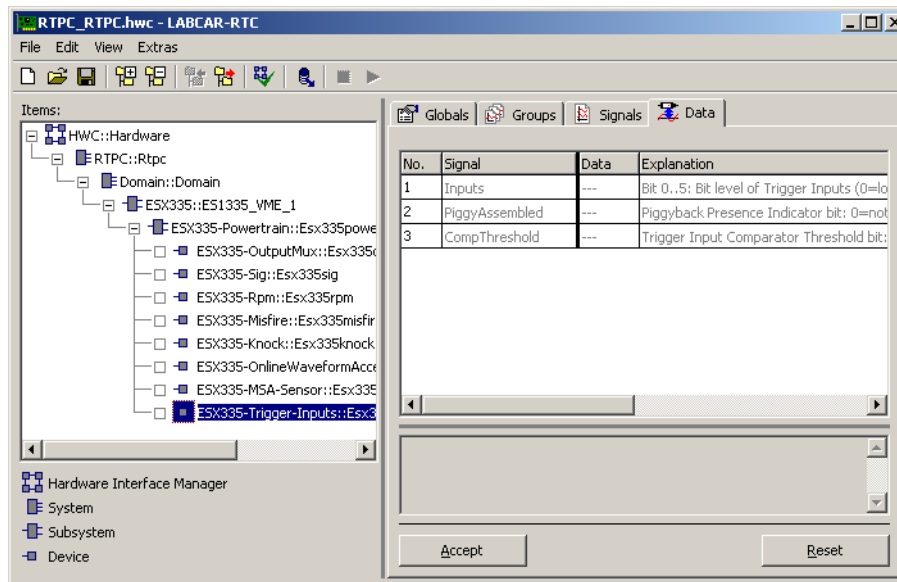
## 10.11 Externe Triggerung der Signalgeneratoren: ESX335-Trigger-Inputs Device

Wenn ein PB1335TRIG Trigger Module zur Triggerung der Signalgeneratoren mit externen Signalen eingesetzt wird, so kann das ESX335-Trigger-Inputs Device verwendet werden, um Statusinformationen zu erhalten.

Darüber hinaus gibt es keine weiteren Konfigurationsparameter.

## 10.11.1 Data (ESX335-Trigger-Inputs Device)

In der Registerkarte „Data“ werden die Signale zum Status des PB1335TRIG-Moduls dargestellt.



**Abb. 10-21** ESX335-Trigger-Inputs Device: Signale der Registerkarte „Data“  
Die Bedeutung der Signale wird im Folgenden beschrieben.

### Inputs

Mit diesem Signal werden die Pegel der Triggereingänge 0 - 5 ermittelt. Dargestellt wird eine Ganzzahl, die in Binärdarstellung ausgewertet wird:

- Bit  $n = 0$  bedeutet Eingang  $n = \text{low}$
- Bit  $n = 1$  bedeutet Eingang  $n = \text{high}$

Beispiel: Die dargestellte Zahl ist 15, binär 001111, also sind die Eingänge 0...3 „high“, Eingänge 4 und 5 „low“.

### PiggyAssembled

Mit diesem Bit wird ermittelt, ob ein PB1335TRIG-Modul auf der ESX335 montiert ist.

### CompThreshold

Liefert die Konfiguration der Triggerschwelle (per Steckbrücke auf dem Modul) als Bitwert:

- Bit = 0: Die Triggerschwelle beträgt typisch  $UBatt/2$  (siehe Benutzerhandbuch der ESX335.1)
- Bit = 1: Die Triggerschwelle beträgt typisch 2,6 V

## 11 ES1336.1 Angle Synchronous Measurement Board

### Begriffsdefinitionen

#### Aktive Signalflanke

Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Pulsflanke

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### LWL

Steht für Lower Window Limit und bezeichnet die untere Grenze des Winkel Fensters bei winkelsynchronen Messungen.

#### Öffnende Pulsflanke

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



#### UWL

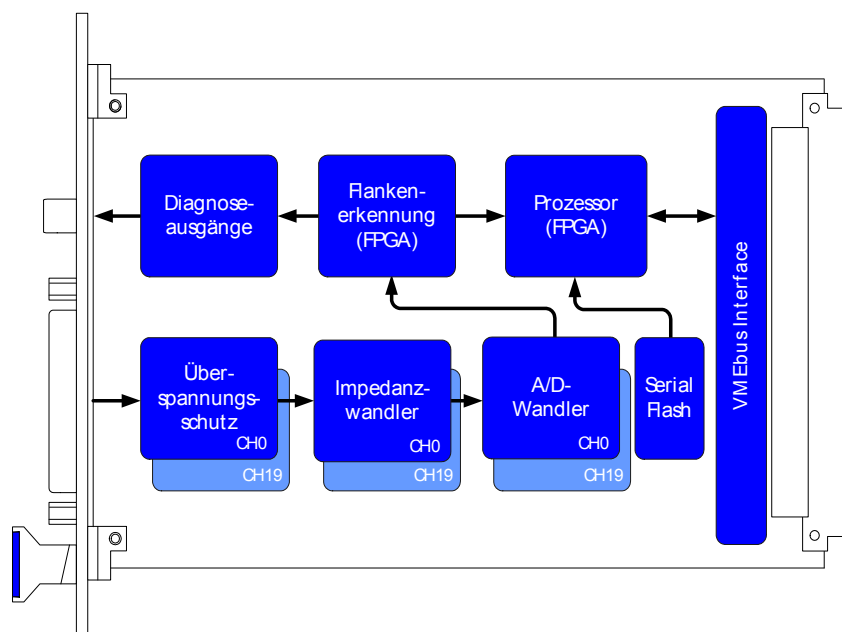
Steht für Upper Window Limit und bezeichnet die obere Grenze des Winkel Fensters bei winkelsynchronen Messungen.

## 11.1 Grundprinzip der Messwertbestimmung

### 11.1.1 Signalerfassung

Das ES1336.1 Angle Synchronous Measurement Board besitzt 20 analoge Eingangskanäle mit einem Spannungsbereich von 0 V bis 40 V und zwei Eingänge für Batteriespannungen mit einem Spannungsbereich von 0 V bis 60 V. Die analogen Eingangsspannungen werden über eine Schutz- und Konditionierungsschaltung auf 12-Bit A/D-Wandler geführt.

Die gewandelten Signale werden einem FPGA zugeführt und sind der Ausgangspunkt für die weitere Signalverarbeitung. Die Abtastrate der A/D-Wandler beträgt 12,5 MHz auf jedem Eingangskanal.



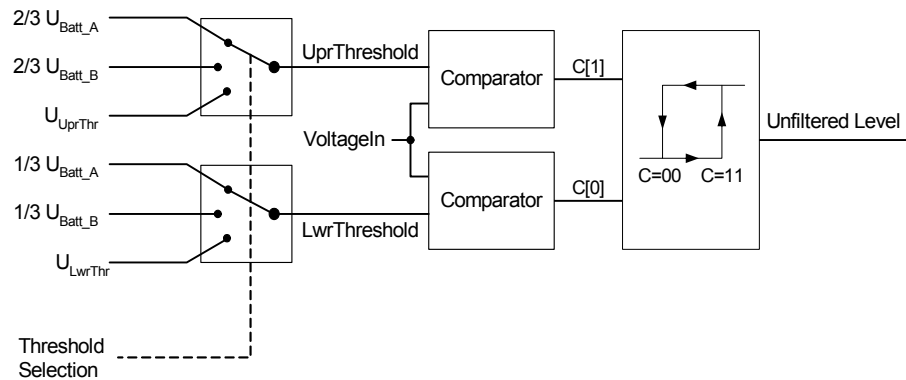
**Abb. 11-1** Blockschaltbild des ES1336.1 Angle Synchronous Measurement Board

Ähnlich verläuft die Verarbeitung der beiden Batteriespannungen, die über eine Konditionierungsschaltung auf einen 10-bit A/D-Wandler geführt werden. Die Abtastrate dieses A/D-Wandlers ist 78,125 kHz pro Batteriespannung. Die gewandelten Batteriespannungen werden ebenfalls dem FPGA zugeführt.

Für winkelsynchrone Messungen bietet die ES1336.1 im FPGA ein Winkeltakt-Interface, das sowohl im Master- als auch im Slave-Betrieb arbeiten kann. Im Master-Betrieb generiert das FPGA den Winkeltakt und stellt ihn anderen Boards im VMEbus-System auf der Backplane zur Verfügung. Im Slave-Betrieb synchronisiert sich das Board auf einen auf der VMEbus-Backplane verfügbaren Winkeltakt.

### 11.1.2 Komparatorstufen

Für die weitere Messwerterfassung ist es erforderlich, aus den gemessenen Eingangsspannungen eine binäre Pegelinformation zu extrahieren. Hierzu wird jede Eingangsspannung auf eine Komparatorstufe mit Hystereseverhalten geführt (Abb. 11-2).



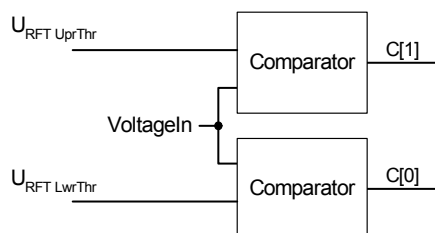
**Abb. 11-2** Komparatorstufe

Die Schwellspannungen der Komparatoren können vom Anwender auf drei verschiedene Arten konfiguriert werden:

- Untere Schwelle = 1/3 Batteriespannung A  
Obere Schwelle = 2/3 Batteriespannung A
- Untere Schwelle = 1/3 Batteriespannung B  
Obere Schwelle = 2/3 Batteriespannung B
- Vorgabe der Schwellwerte durch Anwender

Die ES1336.1 ermöglicht auch die Vermessung der Anstiegs- und Abfallzeiten von Signalen. Diese Zeiten werden üblicherweise zwischen 10% und 90% der Signalamplitude gemessen.

Da diese Einstellung der Schwellwerte nicht immer auch für die Pegelextraktion gewünscht ist, besitzt die ES1336.1 separate Komparatoren für die Vermessung von Anstiegs- und Abfallzeiten. Die Komparatorschwellen werden durch den Anwender in Volt vorgegeben (Abb. 11-3).



**Abb. 11-3** Komparatorstufe für die Messung der Signalanstiegs- und -abfallzeiten

### 11.1.3 Pulsweiten- und Winkelmessungen

Die Pegelinformation eines jeden Kanals wird ständig auf Änderungen hin überprüft, d.h. es wird eine Erkennung von Signalfanken durchgeführt. Bei Auftreten einer Flanke werden die aktuelle Zeit und der aktuelle Kurbelwellenwinkel in einem FPGA-internen Speicher abgelegt. Die Prozessor-Firmware der ES1336.1

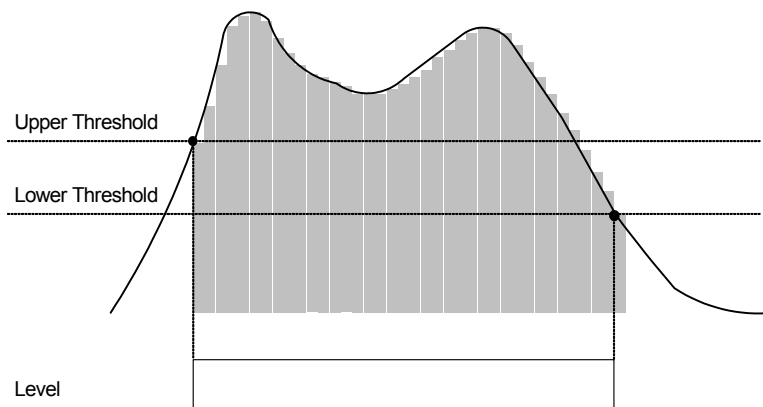
kann nun die erfassten Werte aus dem Speicher auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben.

Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind, wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten softwaretechnischen Aufbereitung der Messwerte hat den Vorteil, dass zur Implementierung weiterer Messverfahren (neben den bereits implementierten) eine Anpassung der Firmware und des RTIO-Treibers genügt.

#### 11.1.4 Pulsintegration

Die Arbeitsweise der Pulsintegration wird in Abb. 11-4 beschrieben – sie zeigt das analoge Eingangssignal zusammen mit den beiden Komparatorschwellen und dem von den Komparatoren generierten Pegelsignal. Wird die obere Komparatorschwelle überschritten, so werden die mit einer Frequenz von 12,5 MHz erfassten A/D-Werte solange akkumuliert, bis das Analogsignal wieder die untere Komparatorschwelle unterschreitet. Das Integral berechnet sich über das Produkt der erhaltenen Summe und dem A/D-Wandler-Abtastintervall von 80 ns.

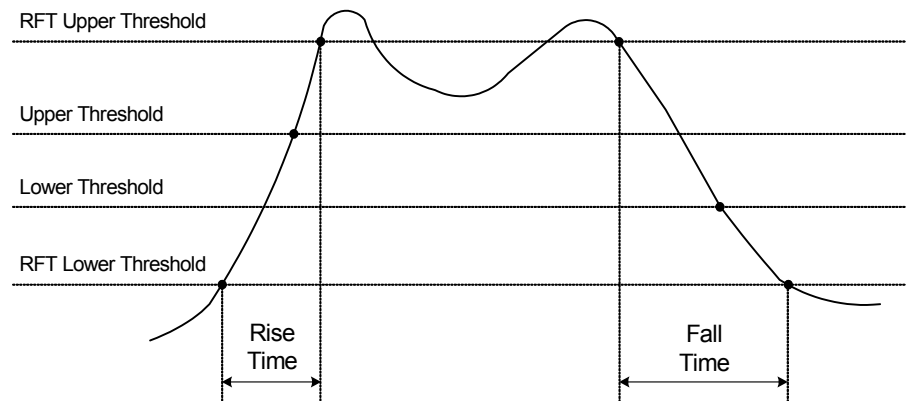


**Abb. 11-4** Pulsintegration



### 11.1.5 Vermessung von Anstiegs- und Abfallzeiten

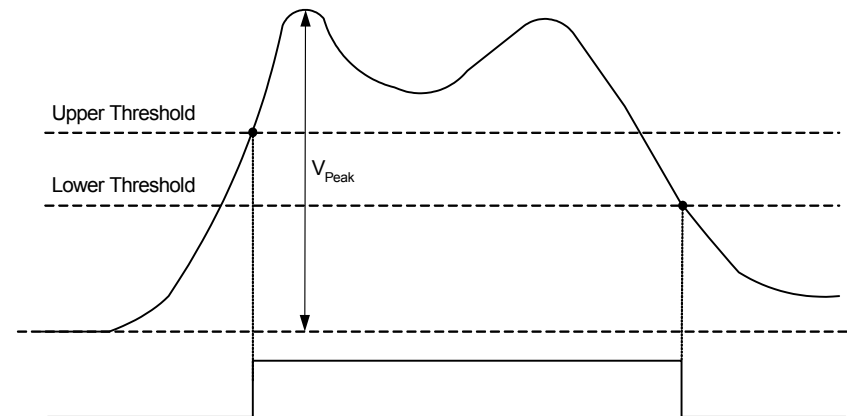
Wie bereits erwähnt, besitzt die ES1336.1 separate Komparatoren für die Vermessung von Signalanstieg- und Signalabfallzeiten an. Die Zeiten werden wie in Abb. 11-5 dargestellt gemessen. Die Auflösung der gemessenen Zeiten ist bestimmt durch die Abtastfrequenz der AVD-Wandler von 12,5 MHz – die Auflösung beträgt demzufolge 80 ns.



**Abb. 11-5** Vermessung von Anstiegs- und Abfallzeiten

### 11.1.6 Spitzenwertvermessung

Die Spitzenwertvermessung ist aktiv, sobald das Analogsignal die obere Komparatorschwelle überschreitet. Unterschreitet das Signal die untere Komparatorschwelle, wird sie deaktiviert. Auch die Spitzenwertvermessung basiert auf mit 12,5 MHz abgetasteten Analogsignalen.



**Abb. 11-6** Spitzenwertverfassung

### 11.1.7 Asynchrone Messungen

Bei asynchronen Messungen wird der entsprechende Messwert (z.B. Frequenz, Tastverhältnis oder Hightime) auf Basis der aktuellsten im Speicher verfügbaren Flanken-Einträge berechnet.

Bei diesen Messungen kann eine zeitsynchrone Timeout-Überwachung aktiviert werden. Wenn innerhalb einer vom Anwender definierbaren Zeitspanne keine Flanke auf dem zugehörigen Hardwarekanal detektiert wird, so wird ein Timeout erkannt.

### 11.1.8 Winkelsynchrone Messungen

---

Charakteristisch für winkelsynchrone Messungen sind Winkelfenster, die durch eine untere Winkelfenstergrenze (LWL) in °KW und eine obere Winkelfenstergrenze (UWL) in °KW spezifiziert werden.

Pro Hardwarekanal kann der Anwender bis zu drei Winkelfenster definieren, die sich überlappen dürfen, deren Größe jedoch jeweils 720 °KW nicht überschreiten darf. Der Anwender definiert Messungen der Art

- Hightime des n-ten Pulses im Winkelfensters
- Winkel der n-ten fallenden Flanke im Winkelfenster
- Integralwert des n-ten (High-)Pulses im Winkelfenster
- Additive Low-Zeit im Winkelfenster.

Die Berechnung der Messwerte erfolgt jeweils dann, wenn der Kurbelwellenwinkel die obere Winkelfenstergrenze überschritten hat. Auch für winkelsynchrone Messungen kann eine Timeoutüberwachung aktiviert werden. Wenn an der oberen Winkelfenstergrenze festgestellt wird, dass der Messwert nicht berechnet werden kann, z.B. wenn die Hightime des vierten Pulses im Winkelfenster zu vermessen ist, jedoch nur drei Pulse aufgetreten sind, so wird ein Timeout erkannt.

### 11.1.9 Timeouts

---

Wird ein Timeout festgestellt, so wird dies (bei aktivierter Timeout-Überwachung) durch Setzen eines Timeout-Flags angezeigt. Gleichzeitig wird im Timeout-Fall auch der dann gültige Messwert durch die Timeout-Überwachung definiert.

Hierbei gibt es zwei Möglichkeiten: Entweder wird ein vom Anwender vorgegebener Messwert an das Simulationsmodell weitergeben oder ein von der jeweiligen Messmethode definierter Wert. Die Entscheidung welcher der beiden Werte weitergegeben wird, trifft der Anwender, wobei jedoch das zweite Verfahren (ein von der Messmethode definierter Wert) nur bei einigen Messverfahren zur Verfügung steht.

## 11.2 ES1336 Subsystem

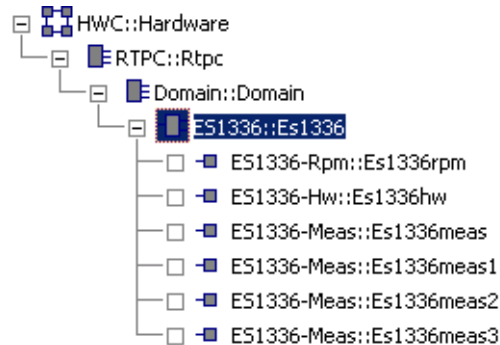
---

Der ES1336.1-RTIO-Treiber stellt dem Anwender bis zu 288 frei konfigurierbare Messwerte (im folgenden auch Software- bzw. Messkanäle genannt) zur Verfügung, wobei jeder einzelne Messwert von jedem beliebigen Hardwarekanal abhängen kann. Die Zuordnung von Hardwarekanälen auf der einen und Messwert auf der anderen Seite ist somit nicht durch den ES1336.1-RTIO-Treiber vorgegeben, d.h. bei der Nutzung eines Laborautos für mehrere Projekte ist es nicht zwingend notwendig, bedeutungsgleiche Signale immer an den selben Hardwarekanälen des ES1336.1 Angle Synchronous Measurement Board anzuschließen.

Darüber hinaus werden dem Anwender mit den 288 Messkanälen etwa 14-mal so viele Ausgangsgrößen zur Verfügung gestellt, wie das Board Hardwarekanäle aufweist. Hierdurch ist es problemlos möglich, aus einem einzigen Eingangssignal mehrere Messwerte zu ermitteln, wie z.B. Frequenz und Tastverhältnis eines PWM-Signals. Eine doppelte Verdrahtung ein und desselben Signals auf verschiedene Kanäle des Boards entfällt.

### Aufbau des ES1336.1-RTIO-Baums

Im RTIO-Editor wird das ES1336.1 Angle Synchronous Measurement Board durch Auswahl des ES1336 Subsystems eingebunden.



**Abb. 11-7** RTIO-Hardwarebeschreibung mit eingebundener ES1336.1

Die Konfiguration der 20 Hardwarekanäle und der 8 Diagnoseausgänge erfolgt mit dem ES1336-Hw Device, das dem ES1336 Subsystem zugeordnet wird.

Die Konfiguration des Winkeltakt-Interfaces wird mit dem ES1336-Rpm Device durchgeführt, das ebenfalls dem ES1336 Subsystem zugeordnet wird.

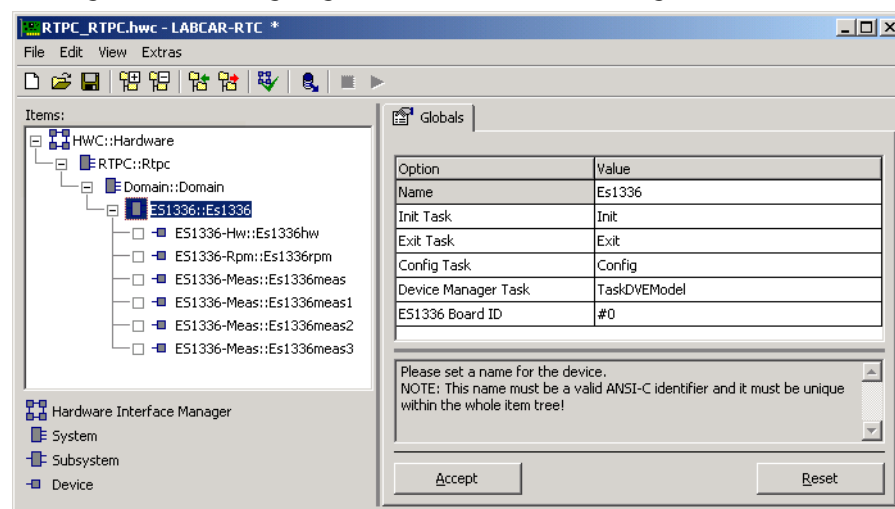
Zur Spezifikation von Messungen lassen sich dem ES1336 Subsystem bis zu neun ES1336-Meas Devices zuordnen. Mit jedem dieser Devices wiederum lassen sich bis zu 32 Messungen definieren – insgesamt ermöglichen RTIO-Treiber und Firmware damit also 288 Messungen.

#### Hinweis

*Bevor ein ES1336-Meas Device eingefügt werden kann, muss zunächst ein ES1336-Hw Device zur Konfiguration der ES1336.1-Hardware eingefügt werden.*

### 11.2.1 Globals (ES1336 Subsystem)

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 11-8** Die Registerkarte „Globals“ des ES1336 Subsystems

### Device Manager Task

Diese Task steuert das Auslesen und den Transport der Messdaten von der ES1336.1 über die ES1130 auf den Echtzeit-PC. Bei der Task-Einstellung sollte die Voreinstellung beibehalten werden.

### ES1336 Board ID

Dieses Optionsfeld dient zur Identifizierung der ES1336.1 – es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1336.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist.

Im RTIO-Editor können bis zu 20 ES1336.1 Boards pro Baugruppenträger eingebunden werden. Die Nummerierung der ES1336.1 Boards im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Nummerierung) beginnend mit 1. Die ermittelte Nummer der ES1336.1 ist im Feld „ES1336 Board ID“ anzugeben.

Dieser RTIO-Parameter ist nicht während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

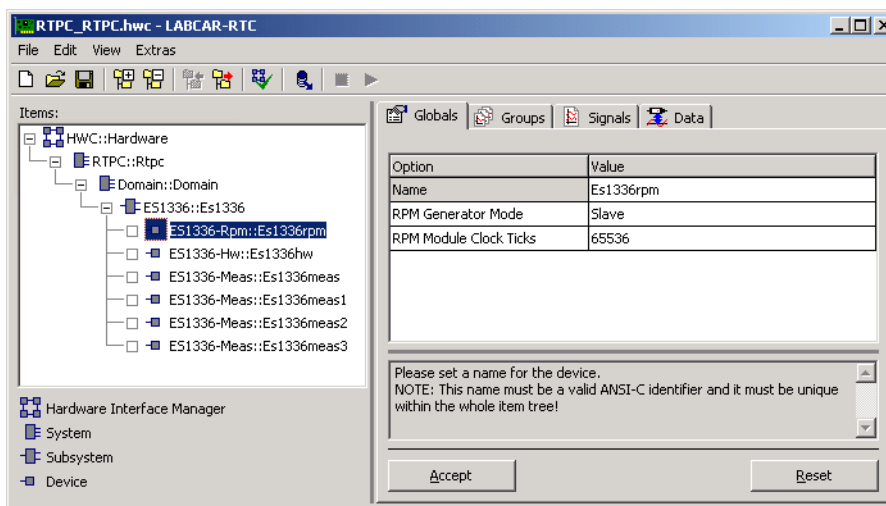
## 11.3 ES1336-Rpm Device

### 11.3.1 Globals (ES1336-Rpm Device)

Das ES1336-Rpm-Device dient zur Konfiguration und Ansteuerung des Winkeltakt-Interfaces.

Zur Durchführung von zum Kurbelwellenwinkel synchronen Messungen benötigt die ES1336.1 Informationen über die Winkelstellung der Kurbelwelle. Diese Informationen ermittelt die ES1336.1 aus dem Winkeltaktsignal, das sie entweder selbst erzeugt oder von der VMEbus-Backplane abgreift.

Abb. 11-9 zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 11-9** Die Registerkarte „Globals“ des ES1336-Rpm Device

Die Bedeutung der einzelnen Parameter wird im Folgenden beschrieben – Tab. 11-1 auf Seite 290 fasst deren Eigenschaften zusammen.

### *RPM Generator Mode*

---

Mit dieser Option wird der Modus definiert in dem das Winkeltakt-Interface arbeitet.

- Slave  
Im Slave-Betrieb synchronisiert sich die ES1336.1 auf einen dem Board über die VMEbus-Backplane zugeführten Winkeltakt.
- Master  
Im Master-Betrieb generiert das Board den Winkeltakt selbst und stellt diesen auf der VMEbus-Backplane zur Verfügung, so dass sich andere Boards im VMEbus-System auf diesen Takt synchronisieren können.
- Local Master  
Im Local Master-Betrieb generiert das Board den Winkeltakt ebenfalls selbst, der Takt wird jedoch nicht auf die VMEbus Backplane geschaltet. Das Optionsfeld ist online editierbar.

### *RPM Module Clock Ticks*

---

Das Auswahlfeld „RPM Module Clock Ticks“ dient zur Definition der Winkelauflösung, d.h. der Anzahl der Winkeltakte pro Nockenwellenumdrehung (720° Kurbelwelle). Die Minimalzahl der Winkeltakte ist  $2^4 = 16$  was einer Winkelauflösung von 45 °KW (°KW = Grad Kurbelwelle) entspricht. Die Maximalzahl der Winkeltakte von  $2^{16} = 65536$  entspricht einer Auflösung von 0,011 °KW.

Das Optionsfeld ist während der Laufzeit des Modells auf dem Experimentaltarget editierbar..

#### **Hinweis**

*Neben der ES1336.1 besitzen auch das ES1321.1 PWM I/O Board und das ES1335.1 Arbitrary Signal Generator Board ein Winkeltakt-Interface, das Master-fähig ist. Prinzipiell kann jedoch nur ein Board in einem VMEbus-System Winkeltakt-Master sein (d.h. den Winkeltakt auf die VMEbus-Backplane einspeisen).*

*Alle anderen ES1336.1 bzw. ES1335.1 und ES1321.1 müssen dann im Slave-Betrieb arbeiten oder als lokaler Master konfiguriert sein. Außerdem muss die Winkelauflösung aller Slave-Boards mit der Winkelauflösung des zugehörigen Masters übereinstimmen.*

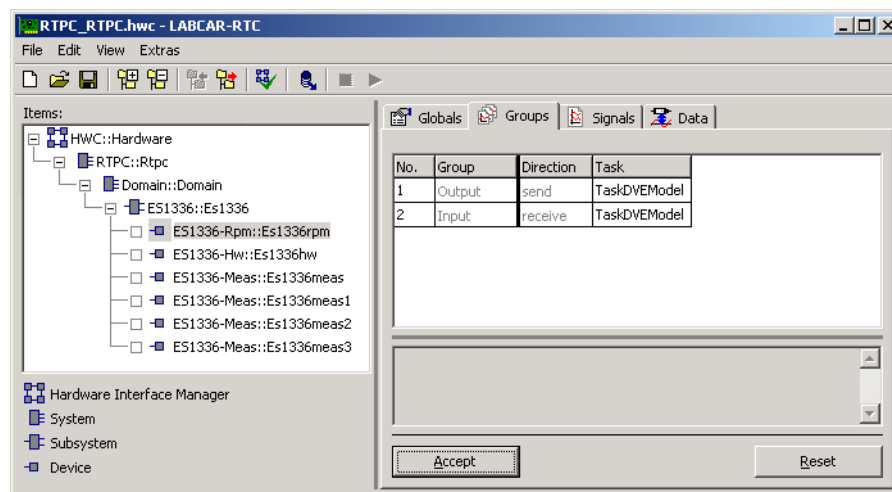
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
RPM Generator Mode	uint8	Ja	Modus des Winkeltakt-Interface 0: Slave 1: Lokaler Master 2: (System-)Master
RPM Module Clock Ticks	uint32	Ja	Winkelauflösung: $2^4$ bis $2^{16}$ in Zweier-Potenz-Schritten

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-1** ES1336-Rpm-Device: Konfigurationsparameter der Registerkarte „Globals“

### 11.3.2 Groups (ES1336-Rpm Device)

Das ES1336-Rpm Device besitzt zwei Signalgruppen



**Abb. 11-10** Die Registerkarte „Groups“ des ES1336-Rpm Device

Die Signalgruppe „Output“ wird vom Experimentaltarget zur ES1336.1 übertragen. Diese Signalgruppe transportiert die Vorgabe für die Motordrehzahl. Die Motordrehzahl bzw. die Signalgruppe ist nur dann von Bedeutung, wenn für „RPM Generator Mode“ Master oder Local Master eingestellt ist.

Die Signalgruppe „Input“ wird in umgekehrter Richtung von der ES1336.1 zum Experimentaltarget übertragen. Diese Signalgruppe transportiert den aktuellen Kurbelwellenwinkel.

Den Signalgruppen sind Betriebssystem-Tasks zuzuordnen – für beide Signalgruppen wird man üblicherweise eine Task mit periodischer Aktivierung wählen. Da die Motordrehzahl vom Motor-Modell berechnet wird und der Kurbelwellenwinkel vom Modell verarbeitet wird, wird man als Aktivierungsperiode das Rechenraster der Modell-Task wählen.

### Die RTIO-Signale der Signalgruppe „Output“

Die Signalgruppe „Output“ umfasst ein RTIO-Signal „EngSpeed“, die Vorgabe für die Motordrehzahl.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
EngSpeed	real32	Motordrehzahl in rpm Minimale Motordrehzahl: 0 rpm Maximale Motordrehzahl: siehe Tab. 11-3. Auflösung: 0,001 rpm Das Signal ist nur dann von Bedeutung, wenn das Winkeltakt-Interface im Master-Modus arbeitet.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-2** ES1336-Rpm-Device: Die RTIO-Signale der Signalgruppe „Output“

$r_{\text{ANGULAR}}$	16	32	64	128	256	512	1024
$n_{\text{Engine,Max}}$ [rpm]	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$1.9 \cdot 10^6$
$r_{\text{ANGULAR}}$	2048	4096	8192	16384	32768	65536	
$n_{\text{Engine,Max}}$ [rpm]	$960 \cdot 10^3$	$480 \cdot 10^3$	$240 \cdot 10^3$	$120 \cdot 10^3$	$60 \cdot 10^3$	$30 \cdot 10^3$	

**Tab. 11-3** Maximale Motordrehzahl ( $n_{\text{Engine,Max}}$ ) in Abhängigkeit von der Winkelauflösung ( $r_{\text{Angular}}$ )

### Die RTIO-Signale der Signalgruppe „Input“

Die Signalgruppe „Input“ enthält das RTIO-Signal „ActAngle“, den aktuellen Kurbelwellenwinkel in °KW.

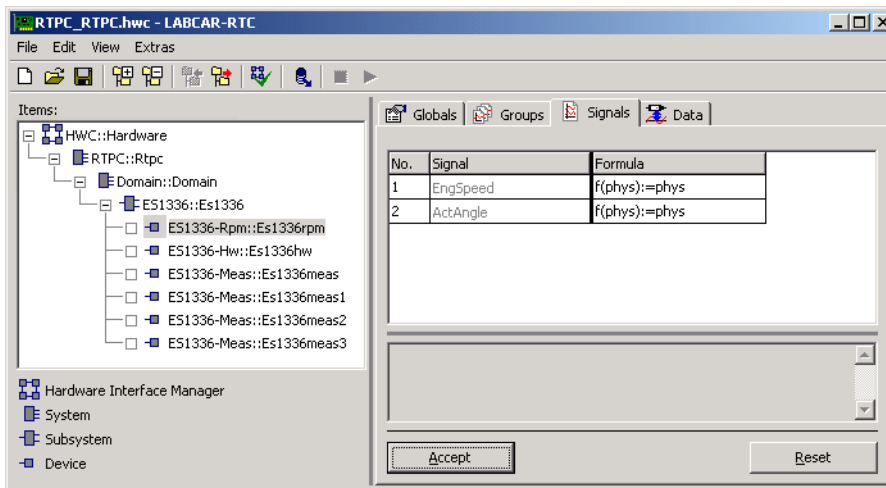
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
ActAngle	real32	Aktueller Kurbelwellenwinkel in °KW. Wertebereich: [0.0 °KW, 720.0 °KW]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-4** ES1336-Rpm-Device: Die RTIO-Signale der Signalgruppe „Input“

### 11.3.3 Signals (ES1336-Rpm Device)

In der Registerkarte „Signals“ sind keine Konfigurationseinstellungen vorzunehmen.



**Abb. 11-11** Die Registerkarte „Signals“ des ES1336-Rpm Device

## 11.4 ES1336-Hw Device

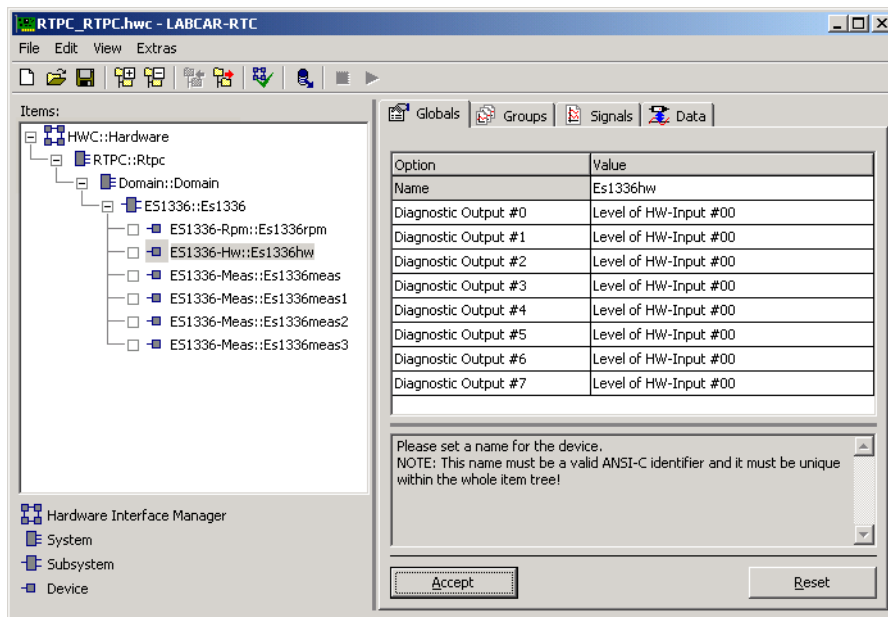
### 11.4.1 Globals (ES1336-Hw Device)

Das ES1336-Hw Device dient zur Konfiguration und Ansteuerung der ES1336.1. Von den einzelnen Hardwarekanälen unabhängige Einstellungen werden in der Registerkarte „Globals“ dieses Devices durchgeführt. In der Registerkarte „Signals“ werden kanalspezifische Einstellungen vorgenommen. Die Erscheinungsform der Registerkarte „Globals“ ist davon abhängig, ob ein ES1336-Rpm-Device eingebunden ist oder nicht.

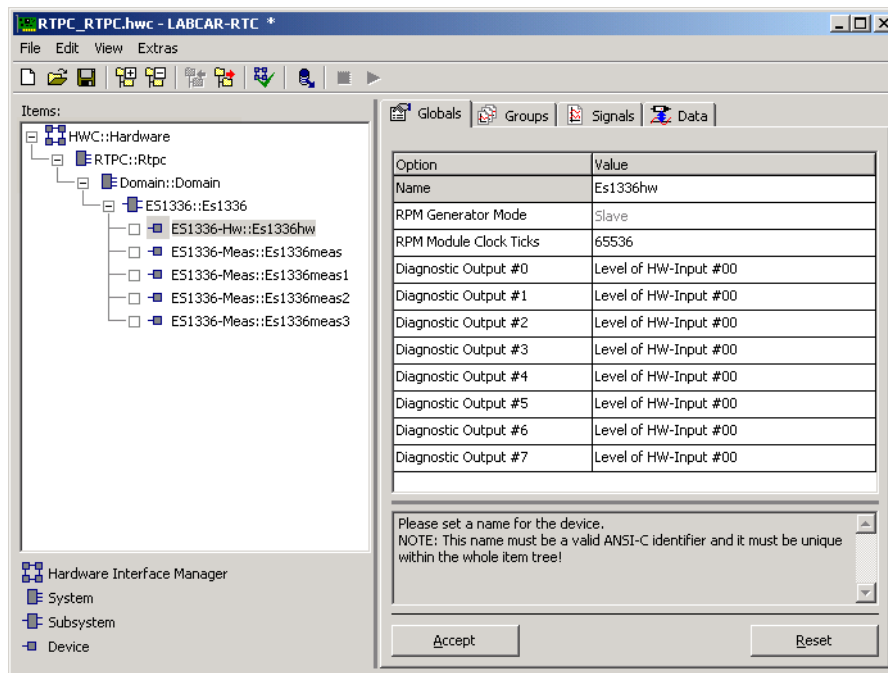
Abb. 11-12 zeigt die RTIO-Parameter der Registerkarte „Globals“ bei eingebundenem ES1336-Rpm-Device. Tab. 11-5 fasst die Eigenschaften dieser Parameter zusammen.

Abb. 11-13 zeigt die RTIO-Parameter der Registerkarte „Globals“ bei fehlendem ES1336-Rpm-Device. In diesem Fall kann das Winkeltakt-Interface der ES1336.1 nur im Slave-Modus betrieben werden. Das Optionsfeld „RPM Generator Mode“ ist deaktiviert. Im Optionsfeld „RPM Module Clock Ticks“ wird die Auflösung des Winkeltakts eingestellt. Nähere Informationen zu diesen beiden Optionsfeldern finden Sie in Abschnitt „Globals (ES1336-Rpm Device)“ auf Seite 288.





**Abb. 11-12** Die Registerkarte „Globals“ des ES1336-Hw Device bei eingebundenem ES1336-Rpm-Device



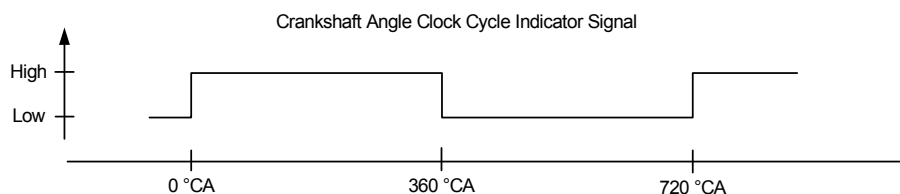
**Abb. 11-13** Die Registerkarte „Globals“ des ES1336-Hw Device bei fehlendem ES1336-Rpm Device

### Diagnostic Output #x (x = 0, 1, ... 7)

Die ES1336.1 bietet die Möglichkeit, FPGA-interne Signale auf die acht Diagnose-Ausgänge zu legen. Mit dem Listenfeld „Diagnostic Output #x (x = 0, 1, ... 7)“ wird definiert, welches Signal auf dem zugehörigen Diagnose-Ausgang ausgegeben wird.

Folgende Signale stehen zur Auswahl:

- Level of HW-Input #xx (xx = 00, 01, ... 19)  
Bei diesen Signalen handelt es sich um die Pegelsignale nach dem Schwellwertvergleich der einzelnen Hardwarekanäle („Unfiltered Level“ in Abb. 11-2 auf Seite 283).
- CA Clock: Cycle Indicator  
Hierbei handelt es sich um ein winkelsynchrones Signal, das sich gut als Triggersignal eignet. Der Verlauf des Signals ist Abb. 11-14 zu entnehmen.



**Abb. 11-14** Winkeltakt-Phasen-Indikator-Signal

Das Optionsfeld ist während der Laufzeit des Modells auf dem Experimentaltarget editierbar..

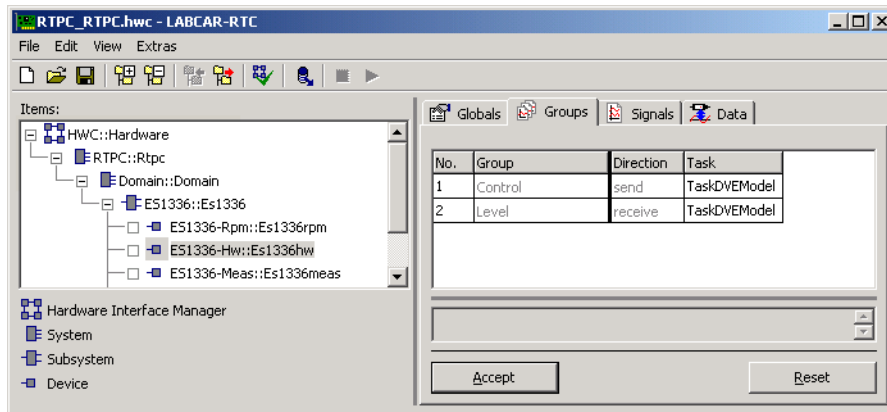
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Diagnostic Output #x (x = 0, 1, 2 ... 7)	uint8	Ja	Signal auf Diagnoseausgang x 0: Pegel von Hardwaresignal 0 1: Pegel von Hardwaresignal 1 ... 19: Pegel von Hardwaresignal 19 31: Winkeltakt-Phasen-Indikator-Signal

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-5** ES1336-Hw Device: Konfigurationsparameter der Registerkarte „Globals“

## 11.4.2 Groups (ES1336-Hw Device)

Das ES1336-Hw Device besitzt zwei Signalgruppen.



**Abb. 11-15** Die Registerkarte „Groups“ des ES1336-Hw Device

Die Signalgruppe „Control“ wird vom Experimentaltarget zur ES1336.1 übertragen. Mit dieser Signalgruppe wird die Messwertberechnung auf den einzelnen Hardwarekanälen des Boards ein- und ausgeschaltet.

Die Signalgruppe „Level“ wird in umgekehrter Richtung von der ES1336.1 zum Experimentaltarget übertragen. Diese Signalgruppe enthält die aktuellen Pegelinformationen aller Hardwarekanäle.

Den Signalgruppen sind Betriebssystem-Tasks zuzuordnen. Für die Signalgruppe „Control“ wird man üblicherweise eine Task mit periodischer Aktivierung und verhältnismäßig großer Aktivierungsperiode (z.B. 100 ms) wählen, da das Ein- und Ausschalten der Hardwarekanäle in der Regel kein hochdynamischer Vorgang ist. Wenn das Ein- und Ausschalten der Hardwarekanäle nur beim Modellstart bzw. Modellstop erfolgen soll, so ist es ausreichend, die Signalgruppe „Control“ der Init Task und der Exit Task des Modells zuzuordnen.

Falls man im Simulationsmodell auch Pegelinformationen auswertet, wird man der Signalgruppe „Level“ eine Task mit periodischer Aktivierung zuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik bzw. der Periodendauer der zu erfassenden Signale.

#### *RTIO-Signale der Signalgruppe „Control“*

Die Signalgruppe „Control“ umfasst 20 RTIO-Signale, mit denen die Messwertberechnung auf den 20 Hardwarekanälen aktiviert bzw. deaktiviert werden kann.

Tab. 11-6 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
EnableChn_00	bool	Ein-/Ausschalten der Messwertberechnung auf den Hardwarekanälen 0 bis 19
...		
EnableChn_19		0: Ausschalten 1: Einschalten

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-6** ES1336-Hw Device: Die RTIO-Signale der Signalgruppe „Control“

### RTIO-Signale der Signalgruppe „Level“

Die Signalgruppe „Level“ umfasst ein RTIO-Signal „LvlBitField\_0“, das als Bitfeld zu interpretieren ist. Das Bitfeld umfasst 20 Bit, in ihm sind die Pegelinformationen der Hardwarekanäle 0 bis 19 codiert.

Tab. 11-7 fasst die Eigenschaften des RTIO-Signals zusammen.

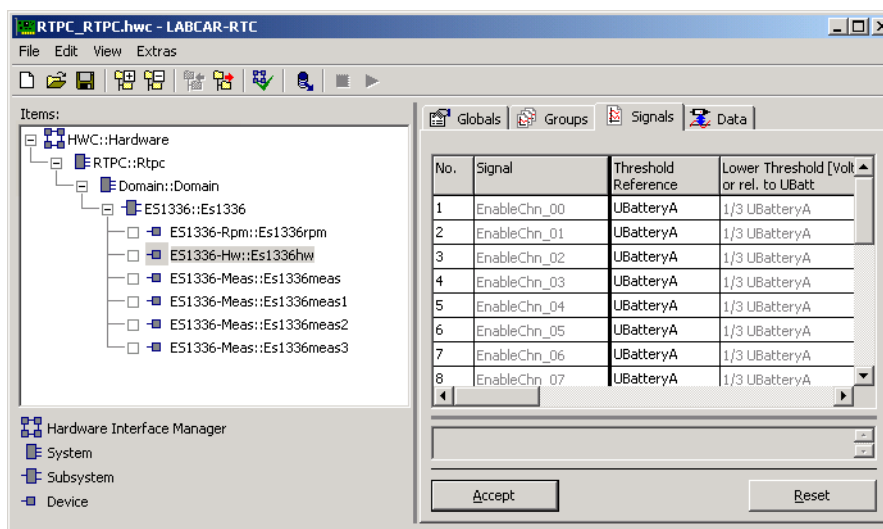
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
LvlBitField_0	uint32	Bitfeld mit Pegelinformationen der Hardwarekanäle 0 bis 19: Kanal 0: Bit 0, LSB (Least Significant Bit) Kanal 19: Bit 19 Bitwert 0: Low Pegel Bitwert 1: High Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-7** ES1336-Hw Device: Die RTIO-Signale der Signalgruppe „Level“

#### 11.4.3 Signals (ES1336-Hw Device)

In der Registerkarte „Signals“ wird die Konfiguration der 20 Hardwarekanäle einer ES1336.1 durchgeführt. Abb. 11-16 zeigt die RTIO-Parameter der Registerkarte „Signals“.



**Abb. 11-16** Die Registerkarte „Signals“ des ES1336-Hw Device

Die Eigenschaften der einzelnen Parameter werden im Folgenden beschrieben – eine Zusammenfassung finden Sie in Tab. 11-8 auf Seite 298. Alle Parameter sind während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

#### Threshold Reference

Mit dem Optionsfeld „Threshold Reference“ werden die Referenzspannungen der Komparatoren zur Pegelermittlung (Abb. 11-2 auf Seite 283) definiert. Die Schwellspannungen der Komparatoren können auf drei verschiedene Arten konfiguriert werden:

- **UBatteryA**  
In diesem Fall entspricht die untere Schwelle 1/3 der Batteriespannung A und die obere Schwelle 2/3 der Batteriespannung A.
- **UBatteryB**  
In diesem Fall entspricht die untere Schwelle 1/3 der Batteriespannung B und die obere Schwelle 2/3 der Batteriespannung B.
- **Absolute Levels**  
Die Komparatorschwellen werden explizit in Volt vorgegeben. Die untere Komparatorschwelle wird im numerischen Eingabefeld „Lower Threshold [Volt] or rel. to UBatt“ definiert, die obere Komparatorschwelle wird im numerischen Eingabefeld „Upper Threshold [Volt] or rel. to UBatt“ definiert

*Lower Threshold [Volt] or rel. to UBatt*

---

*Upper Threshold [Volt] or rel. to UBatt*

---

Diese beiden numerischen Eingabefelder sind nur editierbar, wenn im Feld „Threshold Reference“ die Option „Absolute Levels“ ausgewählt ist. Die Eingangskomparatoren eines Hardwarekanals besitzen ein Hystereseverhalten. Die Eingabefelder definieren die untere und obere Schaltschwelle der Komparatorhysterese in Volt.

*Lower Rise/Fall Threshold [Volt]*

---

*Upper Rise/Fall Threshold [Volt]*

---

Diese beiden numerischen Eingabefelder definieren die untere und obere Schwelle der Komparatoren zur Vermessung von Anstiegs- und Abfallzeiten (Abb. 11-5 auf Seite 285). Die Schwellwerte sind in Volt anzugeben.

*Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)*

---

*Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)*

---

Auf jedem Hardwarekanal können zur Durchführung von winkelsynchronen Messungen drei Winkelfenster definiert werden. „Angle Window #x Lower Limit [°CA]“ definiert jeweils die untere Winkelfenstergrenze in °KW, „Angle Window #x Upper Limit [°CA]“ definiert jeweils die obere Winkelfenstergrenze in °KW.

Die Grenzen können im Bereich [-720.0 °KW, 720 °KW[ eingestellt werden mit den Einschränkungen, dass ein Winkelfenster nicht größer als 720 °KW sein darf und die untere Winkelfenstergrenze kleiner als die obere Winkelfenstergrenze sein muss.

*Asynchronous Timeout [ms]*

Bei asynchronen Messungen kann eine zeitsynchrone Timeout-Überwachung aktiviert werden. Wenn innerhalb der im numerischen Eingabefeld „Asynchronous Timeout [ms]“ definierten Zeitspanne keine Flanke auf dem zugehörigen Hardwarekanal detektiert wird, so wird ein Timeout erkannt. Die Timeout-Zeit ist in Millisekunden anzugeben.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Threshold Reference	uint8	Ja	Komparatorreferenz 0: Absolute Levels 1: UBatteryA 2: UBatteryB
Lower Threshold [Volt] or rel. to UBatt	real64	Ja	Untere Komparatorschwelle in Volt. Wertebereich: [0.0 V, 40.0 V] Nur editierbar, wenn im Feld „Threshold Reference“ die Option „Absolute Levels“ eingestellt ist.
Upper Threshold [Volt] or rel. to UBatt	real64	Ja	Obere Komparatorschwelle in Volt. Wertebereich: [0.0 V, 40.0 V] Nur editierbar, wenn im Feld „Threshold Reference“ die Option „Absolute Levels“ eingestellt ist.
Lower Rise/Fall Threshold [Volt]	real64	Ja	Untere Komparatorschwelle für Anstiegs- und Abfallzeitmessungen in Volt. Wertebereich: [0.0 V, 40.0 V]
Upper Rise/Fall Threshold [Volt]	real64	Ja	Obere Komparatorschwelle für Anstiegs- und Abfallzeitmessungen in Volt. Wertebereich: [0.0 V, 40.0 V]
Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)	real64	Ja	Untere Grenze des jeweiligen Winkelfensters in °KW. Wertebereich: [-720 °KW, 720 °KW]
Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)	real64	Ja	Obere Grenze des jeweiligen Winkelfensters in °KW. Wertebereich: [-720 °KW, 720 °KW]
Asynchronous Timeout [ms]	uint16	Ja	Timeout-Zeit für asynchrone Messungen Wertebereich: [1 ms, 32767 ms]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

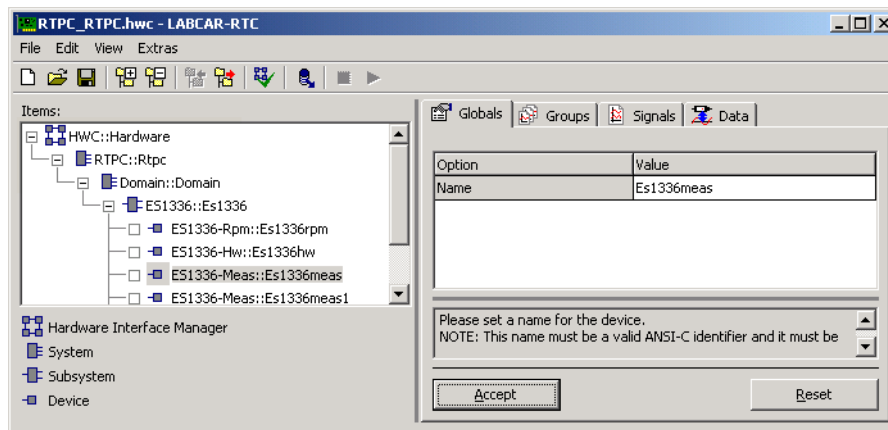
**Tab. 11-8** ES1336-Hw Device: Konfigurationsparameter der Registerkarte „Signals“

## 11.5 ES1336-Meas Device

### 11.5.1 Globals (ES1336-Meas Device)

Das ES1336-Meas Device dient zur Spezifikation und Konfiguration von Messungen. Jedes ES1336-Meas Device bietet 32 Messungen, die frei auf die 20 Hardwarekanäle der ES1336.1 verschaltet werden können. Bis zu 9 ES1336-Meas Devices werden pro ES1336.1 unterstützt, so dass insgesamt 288 Messungen konfiguriert werden können.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.

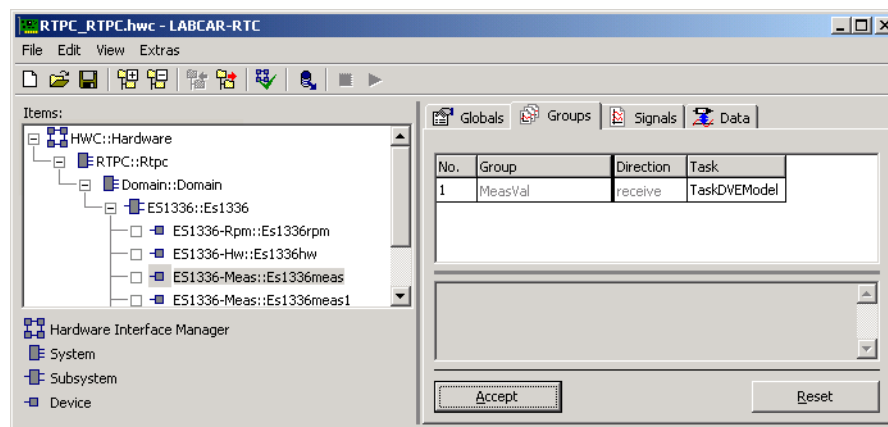


**Abb. 11-17** Die Registerkarte „Globals“ des ES1336-Meas Device

### 11.5.2 Groups (ES1336-Meas Device)

Das ES1336-Meas Device besitzt eine Signalgruppe (Abb. 11-18), die von der ES1336.1 zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen.

Dieser Signalgruppe ist eine Betriebssystem-Task mit periodischer Aktivierung zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.



**Abb. 11-18** Die Registerkarte „Groups“ des ES1336-Meas Device

### Die RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 65 RTIO-Signale. Das Signal „TriggerBitField\_00“ ist als Bitfeld zu interpretieren – es umfasst 32 Bit. In diesem Bitfeld sind die Trigger- oder Update-Daten der 32 Messungen codiert, d.h. es zeigt an, welche Messwerte seit der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der Messwert der zugehörigen Messung neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeouts oder aufgrund einer regulären Messwertberechnung erfolgte – in beiden Fällen wird das Aktualisierungs-Bit der Messung gesetzt.

Die RTIO-Signale „MeasVal\_00“ bis „MeasVal\_31“ enthalten die Messwerte der 32 Messungen. Es handelt sich dabei um regulär ermittelte Messwerte oder (im Falle eines Timeouts) um den für diesen Fall vorgesehenen Timeout-Wert.

Wenn eine Messung nicht verwendet wird, wird dem zugehörigen Messwert 8888.0 zugewiesen. Die physikalische Einheit der Messwerte hängt vom Messverfahren ab:

- Zeitmessungen (Zeitstempel, (additive) Pulsweiten-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Frequenzmessungen erfolgen in Hertz
- Winkelmessungen und Winkelstempel werden in Grad Kurbelwelle (°KW) angegeben
- Spannungsmessungen erfolgen in Volt
- Integralbestimmungen in  $V \cdot \mu s$
- alle anderen Messungen (Tastverhältnisse, Pulszählung) sind dimensionslos

Die RTIO-Signale „Tout\_00“ bis „Tout\_31“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messwert.



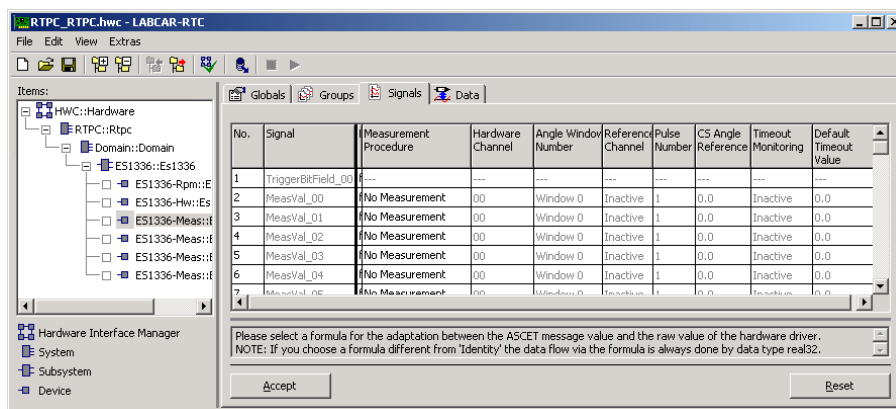
Tab. 11-9 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
TriggerBitField_00	uint32	Bitfeld mit Update-Informationen der 32 Messungen. Messung 0: LSB Messung 31: MSB Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasVal_00 ... MeasVal_31	real64	Messwert. Wenn die Messung nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in $\mu$ s - Frequenzmessungen erfolgen in Hz - Winkelmessungen erfolgen in $^{\circ}$ KW - Spannungsmessungen erfolgen in V - Integralbestimmungen erfolgen in $V \cdot \mu$ s
Tout_00 ... Tout_31	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout festgestellt 1: Timeout festgestellt 2: Timeout-Überwachung ist inaktiv

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-9** ES1336-Meas Device: Die RTIO-Signale der Signalgruppe „MeasVal Signals (ES1336-Meas Device)

In der Registerkarte „Signals“ wird die Konfiguration der 32 Messungen eines ES1336-Meas Device durchgeführt. Abb. 11-19 zeigt die RTIO-Parameter der Registerkarte „Signals“ – Tab. 11-10 auf Seite 305 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 11-19** Die Registerkarte „Signals“ des ES1336-Meas Device  
Alle Parameter sind während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

### *Measurement Procedure*

---

In diesem Listenfeld wird das Messverfahren ausgewählt. In den Abschnitten ab Seite 305 finden Sie detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren. Im Listenfeld bedeutet die Notation --/-- eine Auswertung an steigenden Flanken, die Notation --\-- hingegen eine Auswertung an fallenden Flanken.

Wird im Listenfeld die Option „No Measurement“ eingestellt, so wird keine Messung durchgeführt.

#### **Hinweis**

*Bitte deaktivieren Sie Messungen, die nicht benötigt werden, da diese unnötige Rechenzeiten verursachen.*

### *Hardware Channel*

---

In diesem Listenfeld wird der Hardwarekanal festgelegt, auf dem die Messung durchgeführt werden soll.

### *Angle Window Number*

---

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „Measurement Procedure“ ein winkelsynchrones Messverfahren ausgewählt ist. Es wird das Winkelfenster definiert, innerhalb dessen die Messungen durchgeführt werden. Drei Winkelfenster stehen zur Auswahl, es sind die Winkelfenster des zugeordneten Hardwarekanals (siehe hierzu „Signals (ES1336-Hw Device)“ auf Seite 296).

### *Reference Channel*

---

Die in Abschnitt 11.6.3 auf Seite 307 beschriebenen Pulsweitenmessungen mit Enable- oder Validierungssignal benötigen zur Durchführung einen zweiten Hardwarekanal: Der Kanal an dem das Enable- oder Validierungssignal anliegt. Das Feld „Reference Channel“ zeigt den Hardwarekanal an, an den (bei Auswahl einer solchen Messung) das Enable- bzw. Validierungssignal anzuschließen ist.

Das Feld ist ein reines Anzeigefeld, ist also nicht editierbar und hat bei allen anderen Messungen keine Bedeutung.

### *Pulse Number*

---

Dieses numerische Eingabefeld ist nur editierbar, wenn im Listenfeld „Measurement Procedure“ ein winkelsynchrones Messverfahren ausgewählt ist. Bei puls- und flankenselektiven Messungen wird in diesem Feld die Nummer des Pulses bzw. der Flanke angegeben, der bzw. die vermessen werden soll.

Bei additiven Messungen und Pulszählungen hat dieses Feld keine Bedeutung und ist ebenfalls deaktiviert.

### *CS Angle Reference*

---

Der Winkelbezugspunkt in Grad Kurbelwelle, auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung und deaktiviert.

### *Timeout Monitoring*

---

Bei der Timeout-Überwachung muss zwischen asynchronen und winkelsynchronen Messungen unterschieden werden. Bei asynchronen Messungen erfolgt die Überprüfung auf Timeout in periodischen Abständen. Die Zeitdauer kann für jeden Hardwarekanal getrennt eingestellt werden (siehe hierzu auch „Signals (ES1336-Hw Device)“ auf Seite 296). Wird während der Timeout-Periode keine Flanke auf dem zur Messung gehörigen Hardwarekanal festgestellt, so wird ein Timeout erkannt. Jede Flanke auf dem Hardwarekanal setzt die Timeout-Überwachung zurück.

Bei winkelsynchronen Messungen erfolgt die Überprüfung auf Timeout alle 720 °KW jeweils an der oberen Winkelfenstergrenze. Es wird ein Timeout erkannt, wenn kein Messwert berechnet werden konnte. Als Beispiel wird ein Timeout erkannt, wenn die Hightime des fünften Pulses im Winkelfenster zu vermessen ist, jedoch nur vier Pulse auftraten.

Mit dem Listenfeld „Timeout Monitoring“ kann die Timeout-Überwachung deaktiviert (Option „Inactive“) oder aktiviert (Optionen „Intvl Predef“ und „Intvl InpDep“) werden. Der Unterschied zwischen den Optionen „Intvl Predef“ und „Intvl InpDep“ besteht im Messwert, der im Timeout-Fall an das Modell übermittelt wird. Ist die Option „Intvl Predef“ eingestellt, so wird in diesem Fall der im numerischen Eingabefeld „Default Timeout Value“ eingetragene Wert übermittelt. Bei der Option „Intvl InpDep“ ist der zurück gelieferte Wert abhängig vom Messverfahren, wobei diese Option jedoch nur bei einigen Messverfahren zur Verfügung steht. Näheres hierzu finden Sie in den Abschnitten ab Seite 305.

### *Default Timeout Value*

---

Dieses numerische Eingabefeld ist nur editierbar, wenn im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt ist. Das Eingabefeld definiert in diesem Fall den Messwert, der im Timeout-Fall an das Modell übermittelt wird.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Measurement Procedure	uint32	Ja	Messverfahren Messverfahren (Parameterwert) „No Measurement“ (0) „Angular: Additive Hightime [µs]“ (42) „Angular: Additive Lowtime [µs]“ (43) „Angular: Rising Edge of n-th Pulse [°CA]“ (44) „Angular: Falling Edge of n-th Pulse [°CA]“ (45) „Angular: H-Time n-th Pulse (Pu Qual.) [µs]“ (32) „Angular: L-Time n-th Pulse (Pu Qual.) [µs]“ (33) „Angular: Time Stamp of n-th Rising Edge [µs]“ (46) „Angular: Time Stamp of n-th Falling Edge [µs]“ (47) „Angular: Number of High-Pulses“ (48) „Angular: Number of Low-Pulses“ (49) „Angular: H-Time n-th Pulse (H-Enable) [µs]“ (34) „Angular: H-Time n-th Pulse (L-Enable) [µs]“ (35) „Angular: H-Time n-th Pulse (H-Valid.) [µs]“ (36) „Angular: H-Time n-th Pulse (L-Valid.) [µs]“ (37) „Angular: L-Time n-th Pulse (H-Enable) [µs]“ (38) „Angular: L-Time n-th Pulse (L-Enable) [µs]“ (39) „Angular: L-Time n-th Pulse (H-Valid.) [µs]“ (40) „Angular: L-Time n-th Pulse (L-Valid.) [µs]“ (41) „Angular: Peak-Amplitude of n-th Pulse [V]“ (50) „Angular: Integral U*dt of n-th Pulse [µVs]“ (51) „Angular: Rising Time of n-th Pulse [µs]“ (52) „Angular: Falling Time of n-th Pulse [µs]“ (53) „Asynchron: Frequency --/-- [Hz]“ (1) „Asynchron: Cycle Time --/-- [µs]“ (2) „Asynchron: Duty Cycle H/(L+H) --/--“ (3) „Asynchron: Duty Cycle L/(L+H) --/--“ (4) „Asynchron: Hightime [µs]“ (5) „Asynchron: Lowtime [µs]“ (6) „Asynchron: Rising Time [µs]“ (7) „Asynchron: Falling Time [µs]“ (8) „Asynchron: Input Voltage [V]“ (9)
Hardware Channel	uint8	Ja	Hardwarekanal, auf dem die Messung durchgeführt wird. 0 ... 19: Kanalnummer
Angle Window Number	uint8	Ja	Winkelfensterauswahl (Wertebereich: 0, 1, 2) Nur relevant / editierbar bei winkelsynchronen Messungen.
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Reference Channel	-	Nein	Referenzkanal bei Messungen mit Enable- und Validierungssignal. Reines Anzeigefeld, nicht editierbar.
Pulse Number	uint16	Ja	Bei puls- und flankenselektiven Messungen: Die Nummer des Pulses bzw. der Flanke, die vermessen werden soll (Wertebereich: 1, 2, ... 32)
CS Angle Reference	real64	Ja	Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen (Wertebereich: [-720.0 °KW, 720.0 °KW])
Timeout Monitoring	uint8	Ja	Timeout-Überwachung. 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“  „Intvl InpDep“ steht nur bei folgenden Messverfahren zur Verfügung: - „Asynchron: Duty Cycle H/(L+H) --/--“ - „Asynchron: Duty Cycle L/(L+H) --/--“ - „Angular: Additive Hightime [µs]“ - „Angular: Additive Lowtime [µs]“ - „Angular: Time Stamp of n-th Rising Edge [µs]“ - „Angular: Time Stamp of n-th Falling Edge [µs]“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant im „Intvl Predef“-Modus zur Timeout-Überwachung.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 11-10** ES1336-Meas Device: Konfigurationsparameter der Registerkarte „Signals“

## 11.6 Die Messverfahren

### 11.6.1 Asynchrone Messungen

Die Berechnung von asynchronen Messwerten findet immer dann statt, wenn keine winkelsynchronen Messwerte zu berechnen sind, mindestens jedoch einmal pro Nockenwellenumdrehung (720 °KW).

Für die Berechnung

- der Frequenz („Asynchron: Frequency --/-- [Hz]“)
- der Periodendauer („Asynchron: Cycle Time --/-- [µs]“)
- von Tastverhältnissen („Asynchron: Duty Cycle H/(L+H) --/--“, „Asynchron: Duty Cycle L/(L+H) --/--“)
- von Pulsweiten („Asynchron: Hightime [µs]“, „Asynchron: Lowtime [µs]“)
- und von Anstiegs- bzw. Abfallzeiten („Asynchron: Rising Time [µs]“, „Asynchron: Falling Time [µs]“)

werden jeweils die drei letzten vor dem Berechnungszeitpunkt detektierten Flanken ausgewertet.

Zur Spannungsmessung (Messverfahren „Asynchron: Input Voltage [V]“) werden zum Berechnungszeitpunkt 128 Analogwerte vom A/D-Wandler ausgelesen und daraus der Mittelwert gebildet.

Für asynchrone Messungen kann eine zeitsynchrone Timeout-Überwachung aktiviert werden. Ist diese aktiv, so wird genau dann ein Timeout ausgelöst, wenn innerhalb des Timeout-Intervalls keine Flanke auf dem zugehörigen Hardwarekanal detektiert wurde. Jede Flanke auf dem Hardwarekanal setzt die Überwachung zurück.

Bei den Tastverhältnismessungen

- „Asynchron: Duty Cycle H/(L+H) --/--“
- „Asynchron: Duty Cycle L/(L+H) --/--“

kann der Timeout-Modus „Intvl InpDep“ eingestellt werden.

Die Messwerte, die im Timeout-Fall an das Modell übermittelt werden, sind in Tab. 11-11 aufgelistet.

Messverfahren	Signalpegel zum Zeitpunkt der Überprüfung auf Timeout	Messwert im Timeout-Fall
Asynchron: Duty Cycle H/(L+H) --/--	low	0
	high	1
Asynchron: Duty Cycle L/(L+H) --/--	low	1
	high	0

**Tab. 11-11** Asynchrone Messungen: Messwert im Timeout-Fall im Timeout-Modus „Intvl InpDep“

### 11.6.2 Winkelsynchrone Messungen - Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften von Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: H-Time n-th Pulse (Pu Qual.) [ $\mu\text{s}$ ]
- Angular: L-Time n-th Pulse (Pu Qual.) [ $\mu\text{s}$ ]

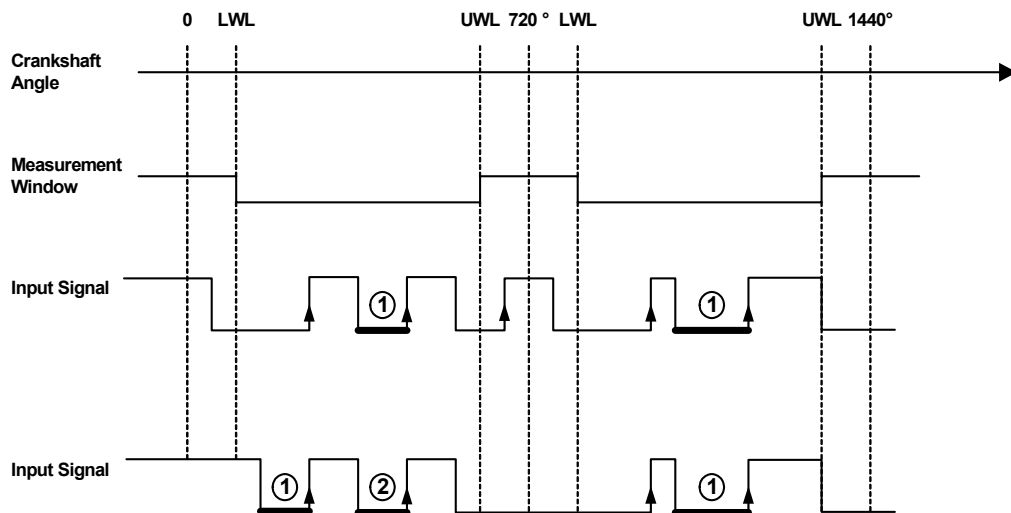
Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer des zu vermessenden Pulses
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

### Funktionsbeschreibung

Bei den Pulsweitenmessungen handelt es sich um puls-qualifizierte Messungen, d.h., es werden nur solche Pulse vermessen und gezählt, deren öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen (gültiger Puls). Ermittelt wird die Pulsweite des n-ten gültigen Pulses im gewählten Winkelfenster. Der Messwert wird an der oberen Winkelfenstergrenze berechnet.

Abb. 11-20 veranschaulicht die Funktionsweise anhand einer low-aktiven Pulsweitenmessung.



**Abb. 11-20** Darstellung der Funktionsweise einer puls-qualifizierten, low-aktiven Pulsweitenmessung (Messverfahren Angular: L-Time n-th Pulse (Pu Qual.) [ $\mu$ s]). Vermessene Pulse bzw. Pulse die gezählt werden sind mit erhöhter Linienstärke hervorgehoben.

### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, da zum Beispiel die Lowzeit des fünften Pulses zu vermessen ist aber nur vier Pulse auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt. Die Option „Intvl InpDep“ wird nicht unterstützt.

#### 11.6.3 Pulsweitenmessungen mit Enable- oder Validierungssignal

Neben der direkten Erfassung der Pulsweiten eines von allen anderen Hardwarekanälen unabhängigen Signals bietet die ES1336.1 noch die Möglichkeit, Pulsweiten eines Signals in Abhängigkeit von Ereignissen auf einem zweiten Hardwarekanal zu messen. Dieses zweite Signal wird hierbei als Enable- bzw. Validierungssignal bezeichnet.

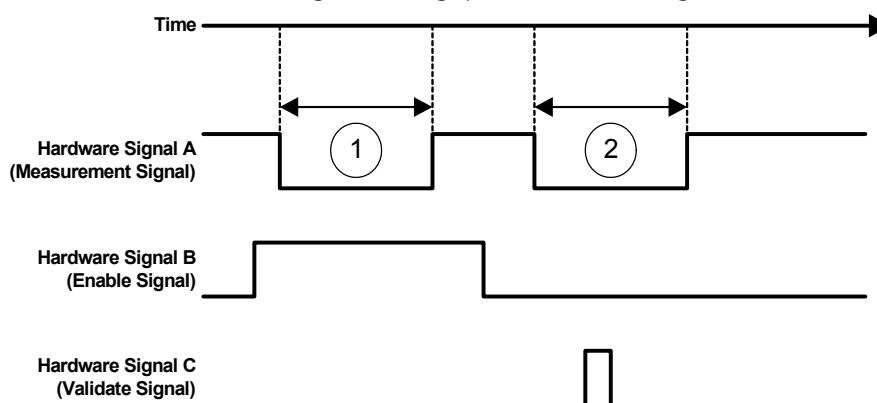
#### Pulsweitenmessungen mit Enablesignal

Bei einer Pulsweitenmessung mit Enablesignal wird ein Puls auf dem Messkanal nur dann vermessen bzw. gezählt, wenn er von einem Puls auf dem zweiten Kanal (im folgenden Enablekanal genannt) vollständig umschlossen wird. Abb.

11-21 zeigt ein low-aktives Messsignal und ein high-aktives Enablesignal. Von den beiden Lowpulsen des Messsignals wird lediglich der erste „enabled“, nur er wird von einem Highpuls des Enablekanals vollständig umschlossen.

#### *Pulsweitenmessungen mit Validierungssignal*

Bei einer Pulsweitenmessung mit Validierungssignal wird ein Puls auf dem Messkanal nur dann vermessen bzw. gezählt, wenn er einen Puls auf dem zweiten Kanal (im folgenden Validierungskanal genannt) vollständig umschließt. Abb. 11-21 zeigt ein low-aktives Messsignal und ein high-aktives Validierungssignal. Von den beiden Lowpulsen des Messsignals wird lediglich der zweite validiert, nur er umschließt vollständig einem Highpuls des Validierungskanals.



**Abb. 11-21** Pulsweitenmessungen mit Enable- und Validierungssignal

Die ES1336.1 bietet Pulsweitenmessungen mit Enable- oder Validierungsoption an. Die Enable- und Validierungspulse können dabei wahlweise als high-aktiv oder auch als low-aktiv definiert werden. Im Einzelnen sind folgende Messungen möglich:

- Angular: H-Time n-th Pulse (H-Enable) [ $\mu$ s]
- Angular: H-Time n-th Pulse (L-Enable) [ $\mu$ s]
- Angular: H-Time n-th Pulse (H-Valid.) [ $\mu$ s]
- Angular: H-Time n-th Pulse (L-Valid.) [ $\mu$ s]
- Angular: L-Time n-th Pulse (H-Enable) [ $\mu$ s]
- Angular: L-Time n-th Pulse (L-Enable) [ $\mu$ s]
- Angular: L-Time n-th Pulse (H-Valid.) [ $\mu$ s]
- Angular: L-Time n-th Pulse (L-Valid.) [ $\mu$ s]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Messkanal
- Pulse Number  
Definiert die Nummer des zu vermessenden Pulses
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird



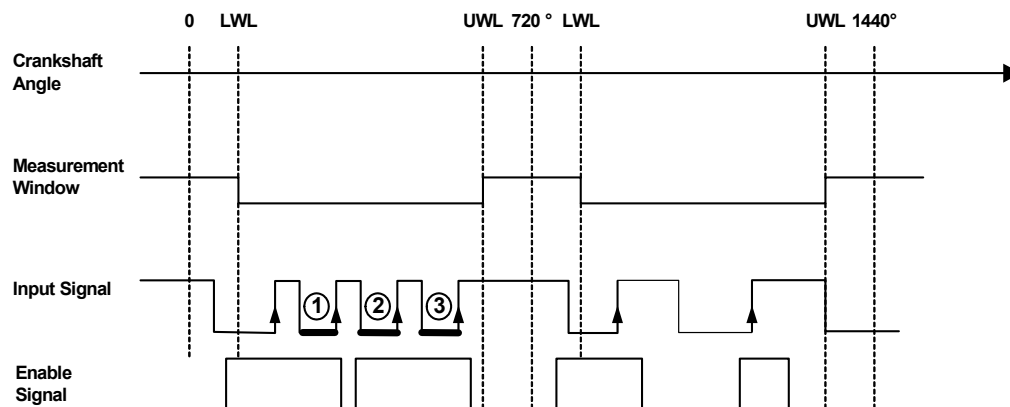
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

### Hinweis

Die Nummer des Enable- bzw. Validierungskanals wird vorgegeben. Er wird im Listenfeld „Reference Channel“ angezeigt. Die Hardwarekanäle 0 und 1 können ein Paar bilden wobei 0 als Messkanal gewählt werden kann, Kanal 1 ist dann der Enable- bzw. Validierungskanal. Es kann jedoch auch Kanal 1 als Messkanal gewählt werden, in diesem Fall wäre Kanal 0 der Enable- bzw. Validierungskanal. Das selbe gilt jeweils für die Kanalpaare (2, 3), (4, 5), (6, 7), (8, 9), (10, 11), (12, 13), (14, 15), (16, 17), (18, 19).

### Funktionsbeschreibung

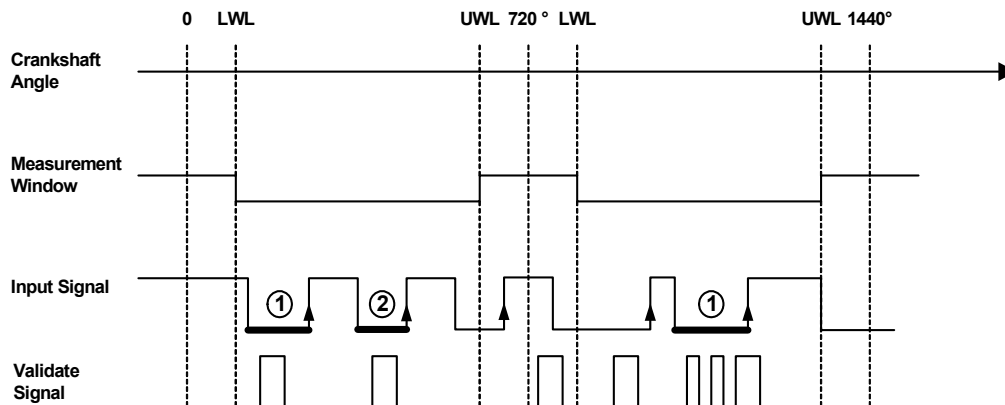
Abb. 11-22 veranschaulicht die Funktionsweise von winkelsynchronen Pulsweitenmessungen mit Enableoption anhand einer Low-Zeit-Messung mit high-aktivem Enablesignal. Es werden nur solche Pulse berücksichtigt, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig von einem aktiven Enable-Puls umschlossen sind („aktivierter Puls“).



**Abb. 11-22** Darstellung der Funktionsweise einer winkelsynchronen Pulsweitenmessung mit Enable-Signal (Messverfahren „Angular: L-Time n-th Pulse (H-Enable) [µs]“). Aktivierte Pulse sind mit erhöhter Linienstärke hervorgehoben.

Abb. 11-23 veranschaulicht die Funktionsweise von winkelsynchronen Pulsweitenmessungen mit Validierungsoption anhand einer Low-Zeit-Messung mit high-aktivem Validierungssignal. Es werden nur solche Pulse berücksichtigt, deren öff-

nende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig einen aktiven Validierungspuls umschließen (validierter Puls).



**Abb. 11-23** Darstellung der Funktionsweise einer winkelsynchronen Pulsweitenmessung mit Validierungssignal (Messverfahren „Angular: L-Time n-th Pulse (H-Valid.) [μs]“). Validierte Pulse sind mit erhöhter Linienstärke hervorgehoben.

Es wird der n-te aktivierte oder validierte Puls innerhalb eines Winkelfensters gemessen. Die Pulsnummer wird im Listenfeld „Pulse Number“ der Registerkarte „Signals“ definiert.

Die Übermittlung der gemessenen Pulsweite an das Modell erfolgt an der oberen Winkelfenstergrenze.

#### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, da zum Beispiel die Lowzeit des fünften validierten bzw. aktivierten Pulses zu vermessen ist, aber nur vier solche Pulse auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt. Die Option „Intvl InpDep“ wird nicht unterstützt.

#### 11.6.4 Additive Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften additiver Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Additive Hightime [μs]
- Angular: Additive Lowtime [μs]

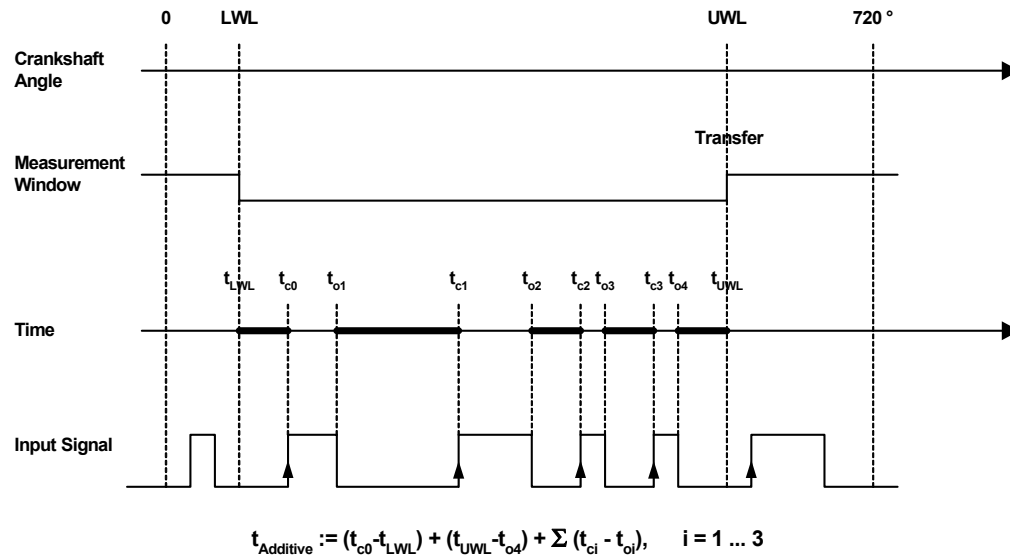
Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird
- Timeout Monitoring  
Alle drei Optionen Inactive, Intvl InpDep und Intvl Predef werden unterstützt.

- Default Timeout Value

### Funktionsbeschreibung

Die additive Zeit ergibt sich als Summe aller Zeitsegmente innerhalb eines Winkelfensters, in denen das Signal aktiv ist, unabhängig davon, ob sich die öffnenden oder schließenden Flanken der Pulse innerhalb oder außerhalb eines Winkelfensters befinden. Abb. 11-24 zeigt beispielhaft die Messwertberechnung bei einer additiven Lowzeit-Messung. Die additive Zeit wird an der oberen Winkelfenstergrenze an das Modell übermittelt.



**Abb. 11-24** Darstellung der Funktionsweise einer winkelsynchronen additiven Lowzeit-Messung (Messverfahren Angular: Additive Lowtime [ $\mu\text{s}$ ]). Die additive Zeit setzt sich aus der Summe der fett markierten Liniensegmente zusammen.

### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle  $720^\circ \text{KW}$  an der oberen Winkelfenstergrenze durchgeführt. Ein Timeout wird erkannt, wenn innerhalb des Winkelfensters keine Flanke auftrat.

Tab. 11-12 listet die Messwerte auf, die im Timeout-Fall an das Modell übergeben werden, wenn im Listenfeld „Timeout Monitoring“ die Einstellung „Intvl InpDep“ ausgewählt ist.

Messverfahren	Signalpegel im Winkelfenster	Messwert im Timeout-Fall
Angular:	low	0
Additive Hightime [µs]	high	Zeitdifferenz: Zeitpunkt obere Winkelfenstergrenze – Zeitpunkt untere Winkelfenstergrenze ( $t_{UWL} - t_{LWL}$ )
Angular:	low	Zeitdifferenz: Zeitpunkt obere Winkelfenstergrenze – Zeitpunkt untere Winkelfenstergrenze ( $t_{UWL} - t_{LWL}$ )
Additive Lowtime [µs]	high	0

**Tab. 11-12** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei Einstellung von „Intvl InpDep“ im Listenfeld „Timeout Monitoring“

### 11.6.5 Vermessung von Flanken: Winkelstempel

In diesem Abschnitt wird die Erfassung der Winkelstempel von Flanken beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Rising Edge of n-th Pulse [°CA]
- Angular: Falling Edge of n-th Pulse [°CA]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer der zu vermessenden Flanke
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird
- CS Angle Reference  
Referenzwinkel
- Timeout Monitoring  
Entweder Inactive oder Intvl Predef, die Option „Intvl InpDep“ wird nicht unterstützt
- Default Timeout Value

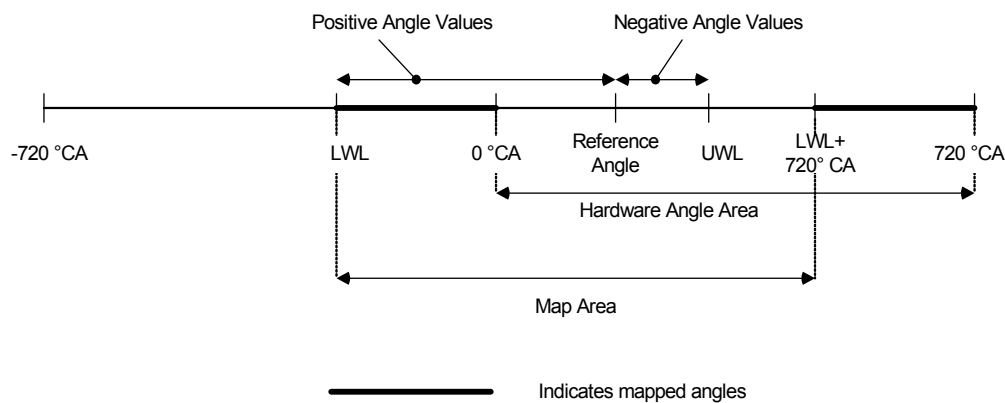
#### Funktionsbeschreibung

Die Messungen erfassen den Winkel der n-ten steigenden oder fallenden Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenzähler auf 0 zurückgesetzt.

Der von der Hardware gelieferte Kurbelwellenwinkel einer Flanke liegt grundsätzlich im Bereich [0 °KW, 720 °KW[. Der gemessene Winkel wird von der ES1336.1-Firmware in den Bereich [LWL, LWL + 720 °KW[ abgebildet und anschließend die Differenz mit dem Referenzwinkel berechnet. Diese Differenz ist der von der Messung gelieferte Messwert, sie ist positiv, wenn der fragliche Winkel kleiner als der Referenzwinkel ist (vgl. Abb. 11-25 auf Seite 313).

Ein Beispiel hierzu: Es sei ein Winkelfenster von  $[-480\text{ }^\circ\text{KW}, 120\text{ }^\circ\text{KW}]$  definiert worden. Der Referenzwinkel soll bei  $-80\text{ }^\circ\text{KW}$  liegen. In einem ersten Fall soll der gemessene Winkel bei  $60\text{ }^\circ\text{KW}$  liegen. Für Winkelwerte zwischen  $0\text{ }^\circ\text{KW}$  und  $(-480\text{ }^\circ\text{KW} + 720\text{ }^\circ\text{KW} = 240\text{ }^\circ\text{KW})$  ist keine Abbildung erforderlich, der gelieferte Messwert ergibt sich zu  $(-80\text{ }^\circ\text{KW} - 60\text{ }^\circ\text{KW} = -140\text{ }^\circ\text{KW})$ .

In einem zweiten Fall soll der gemessene Winkel bei  $540\text{ }^\circ\text{KW}$  liegen. Dieser Winkel ist größer als  $240\text{ }^\circ\text{KW}$  und wird so abgebildet, dass er im Winkelfenster zu liegen kommt. Es werden also  $720\text{ }^\circ\text{KW}$  vom Winkelwert abgezogen ( $540\text{ }^\circ\text{KW} - 720\text{ }^\circ\text{KW} = -180\text{ }^\circ\text{KW}$ ). Der Messwert ergibt sich zu  $(-80\text{ }^\circ\text{KW} + 180\text{ }^\circ\text{KW} = 100\text{ }^\circ\text{KW})$ .



**Abb. 11-25** Funktionsweise einer Winkelmessung.

Da der Referenzwinkel im Bereich  $[-720\text{ }^\circ\text{KW}, 720\text{ }^\circ\text{KW}]$  liegen kann, liefern die Messfunktionen Winkelwerte aus dem Bereich  $[-1440\text{ }^\circ\text{KW}, 1440\text{ }^\circ\text{KW}]$ .

#### **Hinweis**

*Die Auflösung der gemessenen Winkel wird durch den Winkeltaktgenerator vorgegeben.*

#### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle  $720\text{ }^\circ\text{KW}$  an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, da z.B. der Winkel der fünften steigenden Flanke zu vermessen ist, aber nur vier solche Flanken auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt. Die Option „Intvl InpDep“ wird nicht unterstützt.

#### 11.6.6 Vermessung von Flanken: Zeitstempel

In diesem Abschnitt wird die Erfassung der Zeitstempel von Flanken beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Time Stamp of n-th Rising Edge [ $\mu\text{s}$ ]
- Angular: Time Stamp of n-th Falling Edge [ $\mu\text{s}$ ]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal

- Pulse Number  
Definiert die Nummer der zu vermessenden Flanke
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird.
- Timeout Monitoring  
Entweder „Inactive“, „Intvl Predef“ oder „Intvl InpDep“
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messungen erfassen den Zeitstempel der n-ten steigenden oder fallenden Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenzähler auf 0 zurückgesetzt.

#### **Hinweis**

*Der gelieferte Zeitstempel hat keinen Bezug zu anderen Zeitstempeln im LABCAR-System. Es ist ein rein ES1336.1-interner Zeitstempel mit einer Auflösung von 80 ns.*

#### *Timeout-Überwachung*

---

Bei aktiver Timeout-Überwachung wird alle 720 °KW an der oberen Winkelfenstergrenze eine Überprüfung auf Timeout durchgeführt. Ein Timeout wird erkannt, wenn der Messwert nicht berechnet werden konnte, da z.B. der Zeitstempel der fünften steigenden Flanke zu erfassen ist, aber nur vier solche Flanken auftraten.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl InpDep“ eingestellt, so wird als Messwert der Zeitstempel an das Modell übermittelt, der in der Nockenwellenumdrehung ermittelt wurde, in der letztmalig kein Timeout auftrat. Mit anderen Worten: Es wird einfach der alte Zeitstempel beibehalten.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

#### 11.6.7 Vermessung von Flanken: Anstiegs- und Abfallzeiten

---

In diesem Abschnitt wird die Vermessung der Anstiegs- und Abfallzeiten von Flanken beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Rising Time of n-th Pulse [ $\mu$ s]
- Angular: Falling Time of n-th Pulse [ $\mu$ s]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer der zu vermessenden Flanke
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird

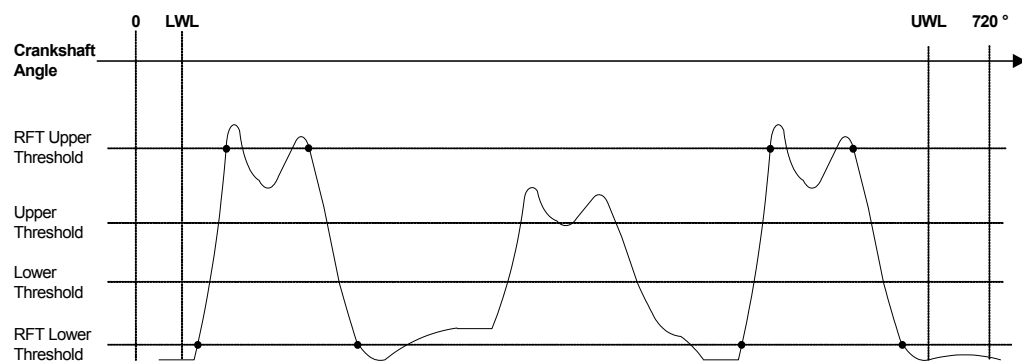
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

### Funktionsbeschreibung

Die Messungen erfassen die Anstiegszeit der n-ten steigenden oder die Abfallzeit der n-ten fallenden Flanke innerhalb eines Winkelfensters. Die Auflösung der Messwerte beträgt 80 ns.

Es ist zu beachten, dass für die Vermessung von Anstiegs- und Abfallzeiten und damit auch für die damit verbundene Zählung der Flanken **andere Komparatorschwellen** maßgebend sind. Dies soll anhand von Abb. 11-26 erläutert werden: Führt man auf dem abgebildeten Signal eine Messung des Winkels der n-ten steigenden Flanke durch, so sind für diese Messung die Komparatorschwellen „Lower Threshold“ und „Upper Threshold“ maßgebend. Mit diesen Schwellen treten im Winkelfenster daher drei steigende Flanken auf.

Für Anstiegs- und Abfallzeitmessungen hingegen sind die Schwellen „**RFT Lower Threshold**“ und „**RFT Upper Threshold**“ maßgebend. Mit diesen Schwellen treten daher zwei steigende und zwei fallende Flanken auf.



**Abb. 11-26** Flankenzählung bei Vermessung von Anstiegs- und Abfallzeiten

### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, weil z.B. die Anstiegszeit der fünften steigenden Flanke zu vermessen ist, aber nur vier solche Flanken auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

Die Option „Intvl InpDep“ wird nicht unterstützt.

### 11.6.8 Spitzenwertmessung

---

In diesem Abschnitt werden die Eigenschaften des folgenden Messverfahren zur Spitzenwerterfassung beschrieben:

- Angular: Peak-Amplitude of n-th Pulse [V]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird
- Pulse Number  
Definiert die Nummer des zu vermessenden Pulses
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messung erfasst die Spitzenspannung des n-ten Pulses innerhalb eines Winkelfensters. Damit ein Puls vermessen bzw. gezählt wird, muss sowohl seine öffnende als auch seine schließende Pulsflanke in ein- und demselben Winkelfenster liegen.

Die gemessene Spannung hat eine Auflösung von 12 Bit – der Messbereich reicht von 0 bis 40 Volt.

#### *Timeout-Überwachung*

---

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, weil z.B. der Spitzenwert des fünften Pulses zu vermessen ist, aber nur vier Pulse auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

Die Option „Intvl InpDep“ wird nicht unterstützt.

### 11.6.9 Pulsintegration

---

In diesem Abschnitt werden die Eigenschaften des folgenden Messverfahrens zur Pulsintegration beschrieben:

- Angular: Integral  $U \cdot dt$  of n-th Pulse [ $V \cdot \mu s$ ]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird
- Pulse Number  
Definiert die Nummer des zu vermessenden Pulses



- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messung erfasst das Integral des n-ten Pulses innerhalb eines Winkelfensters. Damit ein Puls vermessen bzw. gezählt wird, muss sowohl seine öffnende als auch seine schließende Pulsflanke in ein- und demselben Winkelfenster liegen.

Die Methode der Pulsintegration ist im Abschnitt „Pulsintegration“ auf Seite 284 erläutert.

#### *Timeout-Überwachung*

---

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, weil z.B. das Integral des fünften Pulses zu vermessen ist, aber nur vier Pulse auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

Die Option „Intvl InpDep“ wird nicht unterstützt.

### 11.6.10 Pulszählung

---

In diesem Abschnitt werden die Eigenschaften folgender Messverfahren zur Pulszählung beschrieben:

- Angular: Number of High-Pulses
- Angular: Number of Low-Pulses

Zulässige Messoptionen sind:

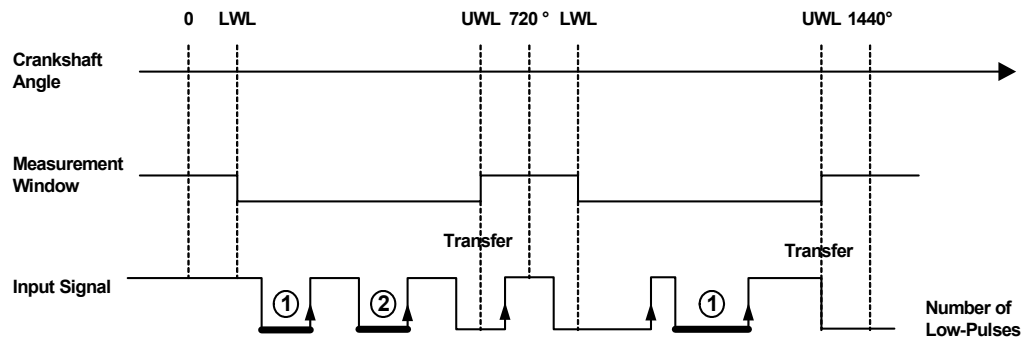
- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messverfahren erfassen die Anzahl der gültigen aktiven Pulse innerhalb eines Winkelfensters. Gültige Pulse sind Pulse, die mit öffnender und schließender Flanke in ein- und demselben Winkelfenster liegen.

Die ermittelte Summe wird an der oberen Winkelfenstergrenze an das Modell übermittelt. (Abb. 11-27).



**Abb. 11-27** Darstellung der Funktionsweise winkelsynchroner Messverfahren zur Pulszählung (Messverfahren Angular: Number of Low-Pulses). Pulse, die gezählt werden, sind durch erhöhte Linienstärke hervorgehoben.

#### *Timeout-Überwachung*

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Ein Timeout wird erkannt, wenn keine Flanke im Winkelfenster auftrat.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

## 12 ES1337.1 Wheelspeed Sensor Simulation Board

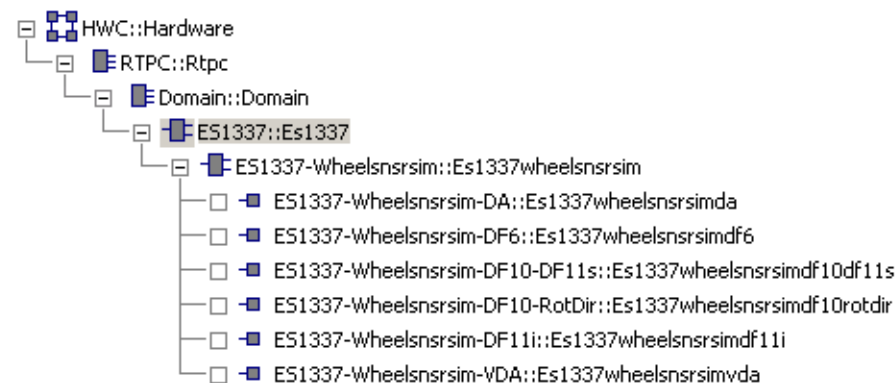
Das ES1337.1 Wheelspeed Sensor Simulation Board besitzt vier unabhängige Signalgeneratoren zur Erzeugung von verschiedenen Sensorsignalen. Folgende Arten von Sensoren können damit nachgebildet werden:

- Passive analoge Sensoren mit sinusförmigem Ausgangssignal (Typ: DF6)
- Aktive digitale Sensoren mit einem Strominterface mit zwei Stromleveln (Typ: DF10)
- Aktive digitale Sensoren mit einem Strominterface mit drei Stromleveln und Vorwärts-/Rückwärtskodierung (Typ: DF10-RotDir)
- Aktive digitale Sensoren mit einem Strominterface mit zwei Stromleveln und zusätzlichen Informationen (Typ: DF11i)
- Aktive digitale Sensoren mit einem Strominterface mit drei Stromleveln und zusätzlichen Informationen (Typ: VDA)

Darüber hinaus stehen zwei weitere, galvanisch isolierte steuerbare Spannungsausgänge zur Verfügung.

### Aufbau des ES1337.1 RTIO-Baums

Im RTIO-Editor wird das ES1337.1 Wheelspeed Sensor Simulation Board durch Auswahl des ES1337 Subsystems gefolgt von einem ES1337-Wheelsnsrsim Subsystem eingebunden.



**Abb. 12-1** RTIO-Hardwarebeschreibung mit eingebundener ES1337.1

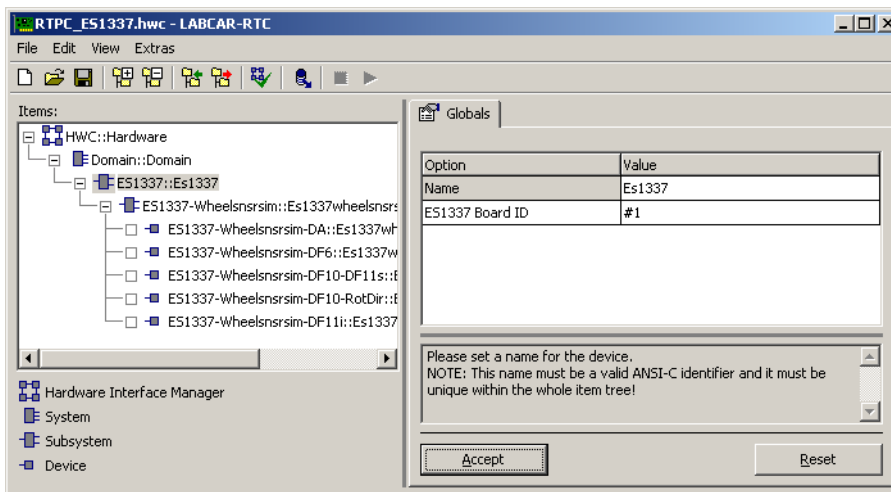
Unter dem ES1337-Wheelsnsrsim Subsystem können ein ES1337-Wheelsnsrsim-DA Device und bis zu vier Devices zur Simulation von Raddrehzahlsensoren eingefügt werden. Diese Devices können in beliebiger Weise kombiniert werden.

Mit ES1337-Wheelsnsrsim-DA Devices können die Ausgabekanäle 4 und 5 zur Ausgabe von analogen Spannungen verwendet werden. Die Kanäle 0 ...3 werden zur Ausgabe der simulierten Sensorsignale benutzt (Strom- oder Spannungssignal – abhängig vom simulierten Sensortyp).

## 12.1 ES1337 Subsystem

### 12.1.1 Globals (ES1337 Subsystem)

Das ES1337 Subsystem dient zur Einstellung von Parametern, die global gültig sind, d.h. die auf alle ES1337.1 RTIO-Elemente wirken.



**Abb. 12-2** Die Registerkarte „Globals“ des ES1337 Subsystems

#### *ES1337 Board ID*

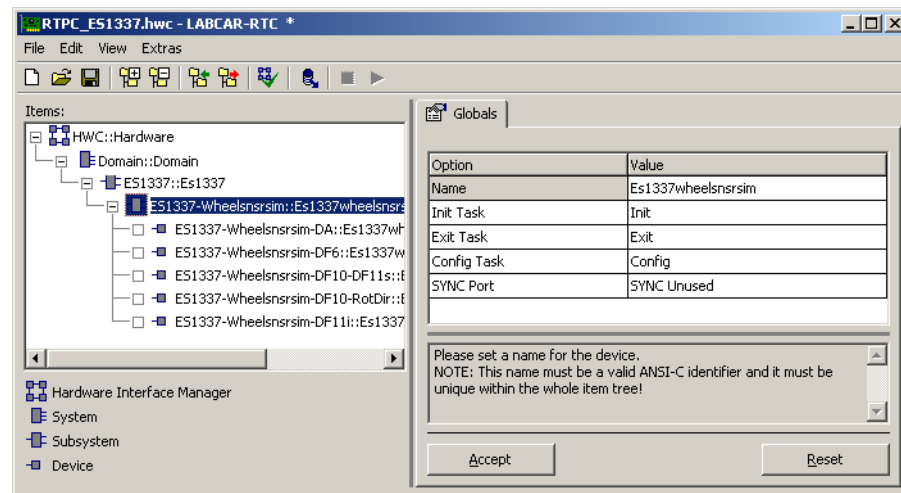
Dieses Optionsfeld dient zur Identifizierung der ES1337.1. Es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1337.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist.

Die Nummerierung der Karten im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Nummerierung) beginnend mit 1. Die ermittelte Nummer der ES1337.1 ist in dem Listenfeld „ES1337 Board ID“ einzustellen. Im RTIO-Editor können bis zu 20 ES1337.1 pro Baugruppenträger eingebunden werden.

### 12.1.2 Globals (ES1337-Wheelsnrsim Subsystem)

Das ES1337-Wheelsnrsim Subsystem dient zur Einstellung von Parametern, die global wirksam sind, d.h. die auf alle Elemente des ES1337-Wheelsnrsim Subsystems wirken.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-3** Die Registerkarte „Globals“ des ES1337-Wheelsnsrsim Subsystems. Das Optionsfeld „Output U/I Ratio“ ist erst nach Auswahl von **View → Show All** sichtbar.

#### *Init Task, Exit Task, Config Task*

In diesen Optionsfeldern können Sie die Tasks auswählen, denen die generierten Init,- Exit- und Config-Prozesse des Treibers zugeordnet werden.

#### *SYNC Port*

In diesem Optionsfeld können Sie festlegen, von welchem Kanal/Ausgang das Synchronisationssignal am Anschluss „SYNC“ auf der Frontplatte stammt. Dieses Signal ist während des gesamten Zahnes 0 des gewählten Sensors aktiv.

#### *Output U/I Ratio*

In diesem Optionsfeld können Sie die Steilheit des (dem D/A-Wandlers nachgeschalteten) U/I-Wandlers einstellen. Der Wert wird in mA/Uref vorgegeben, wobei immer Uref=10 Volt. Der Standardwert ist  $40 \text{ mA}/10 \text{ V} = 4 \text{ mA/V}$  (passend zur Ausgangsschaltung der ES1337.1)

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Init Task, Exit Task Config Task	Auswahl	Nein	Tasks für die Init-, Exit- und Config-Prozesse
SYNC Port	uint8	Ja	„Zahn 0“-Impuls des gewählten Ausgangs/Sensors Wertebereich: Output 0 ... Output 3
Output U/I Ratio	real32	Nein	Umrechnungsfaktor für Spannungs-Strom Wandler (Uref=10V) Wertebereich: 1.0 ...1000.0 [mA/Uref], Default: 40 mA/Uref = 4 mA/V

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

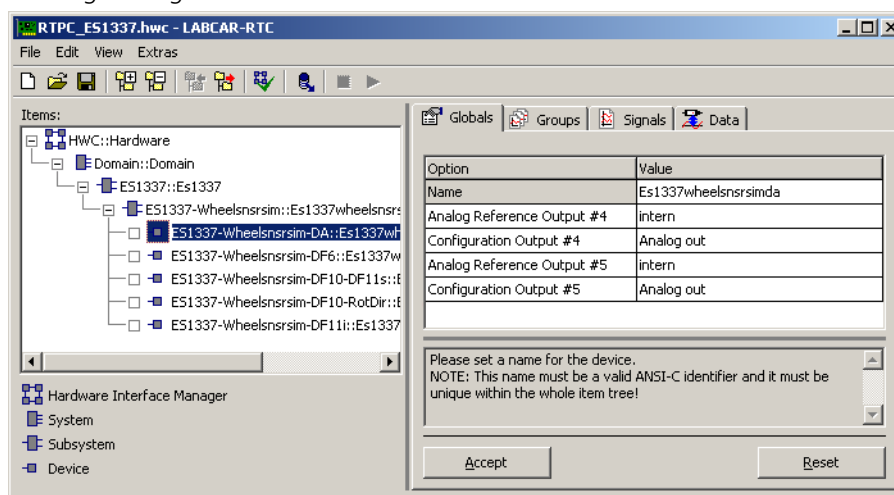
**Tab. 12-1** ES1337-Wheelsnsrsim Subsystem: Konfigurationsparameter der Registerkarte „Globals“

## 12.2 ES1337-Wheelsnsrsim-DA Device

Mit dem ES1337-Wheelsnsrsim-DA Device können Sie die als analoge Spannungsausgänge nutzbaren Kanäle 4 und 5 ansteuern und konfigurieren.

### 12.2.1 Globals (ES1337-Wheelsnsrsim-DA Device)

In der Registerkarte „Globals“ können ausgangsspezifische Konfigurationseinstellungen vorgenommen werden.



**Abb. 12-4** Die Registerkarte „Globals“ des ES1337-Wheelsnsrsim-DA Device

#### Analog Reference Output #4, #5:

In diesen Optionsfeldern können Sie den Ursprung der Referenzspannungen der analogen Kanäle 4 und 5 vorgeben. Zur Auswahl stehen die interne Referenz von 10 V oder eine am externen Referenzspannungseingang des Kanals angelegte Spannung zwischen -10,0 V und +10,0 V.

Die tatsächliche Ausgangsspannung des jeweiligen Kanals ist dann das Produkt aus der Referenzspannung und dem Steuersignals. Diese Einstellung ist nur bei Analogkonfiguration (s.u.) des Kanals von Bedeutung.

#### Configuration Output #4, #5

In diesen Optionsfeldern können Sie Konfiguration der Ausgänge 4 und 5 verändern. Möglich ist ein Betrieb als Analogausgang oder als Binärausgang, entweder als Open Drain oder mit zusätzlichen Stromquelle als Pull-Up nach +5 V. In der Konfiguration als Binärausgang ist der Ausgang low bei einem Ansteuersignal kleiner 0.1 und high bei gleich oder größer 0.1.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Analog Reference Output #4, #5	uint	Ja	Ursprung der Referenzspannung Wertebereich: Intern   Extern
Configuration Output #4, #5	uint	Ja	Konfiguration des Ausgangs Wertebereich: Analog Out   Digital without Pull-Up   Digital with Internal Pull-Up

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-2** ES1337-Wheelsnsrsim-DA Subsystem: Konfigurationsparameter der Registerkarte „Globals“

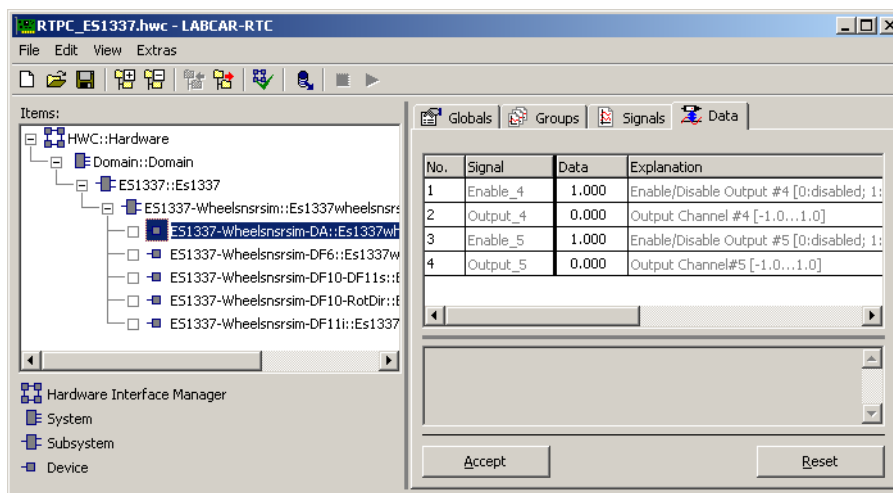
#### 12.2.2 Groups (ES1337-Wheelsnsrsim-DA Device)

Das ES1337-Wheelsnsrsim-DA besitzt eine Signalgruppe „Output“ zur Steuerung der Analogausgänge.

#### 12.2.3 Data (ES1337-Wheelsnsrsim-DA Device)

In der Signalgruppe werden pro analogem Kanal je zwei Signale verarbeitet. Mit den Signalen „Enable\_4“ und „Enable\_5“ können Sie den jeweiligen Kanal aktivieren bzw. deaktivieren. Im deaktivierten Zustand ist der Ausgang des Kanals hochohmig.

Mit den Signalen „Output\_4“, „Output\_5“ können Sie den jeweiligen Ausgang ansteuern. Im Analogbetrieb ergibt sich das Produkt aus diesem Signal und der Referenzspannung als Ausgangsspannung. Im einem der Binärmodi ergibt sich ein logischer High Pegel bei einem Signalwert gleich oder größer 0.1.



**Abb. 12-5** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-DA Device  
Die folgende Tabelle fasst die Eigenschaften der Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Enable_4, Enable_5	uint8	Ein/Ausschalten der Ausgänge 4, 5 0: Ausschalten des Ausgangs 1: Einschalten des Ausgangs
Output_4, Output_5	uint8	Ausgangswert, Wertebereich: -1.0...1.0

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-3** ES1337-Wheelsnsrsim-DA Device: Die Signale der Registerkarte  
„Data“

### 12.3 Radsensorsimulation – Gemeinsame Einstellungen

Die RTIO-Elemente zur Darstellung der simulierten Sensoren sind einheitlich aufgebaut und unterscheiden sich nur durch die spezifischen Eigenschaften der verschiedenen Sensoren.

#### *Analog Output*

Jeder Sensor kann über das Optionsfeld „Analog Output“ in der Registerkarte „Globals“ einem Ausgang zugeordnet werden. Hierbei besteht folgende feste Zuordnung zwischen Montageposition des Sensors und den Analogausgang:

Analogausgang	Radposition des simulierten Sensors
# 0	vorne links (FL)
# 1	vorne rechts (FR)
# 2	hinten links (RL)
# 3	hinten rechts (RR)



### Operation Mode

---

Mit dem Optionsfeld „Operation Mode“ können Sie festlegen, ob die Raddrehzahl über Geschwindigkeit, Zähnezahl und Radradius oder direkt über die Zahnfrequenz des Sensors vorgegeben werden soll.

### Zahnzahl, Verhalten bei Zahnfehlern

---

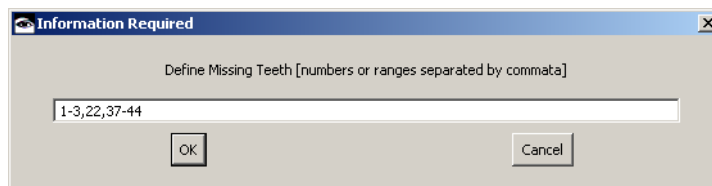
Außer der Zahnzahl (bzw. Polpaarzahl) pro Umdrehung können Sie für jeden Zahn definieren, ob dieser regulär oder fehlerhaft ausgegeben werden soll. Bei allen fehlerhaften Zähnen wird der im Optionsfeld definierte Spannung- oder Stromwert während des gesamten Zahns ausgegeben. Die Wirksamkeit von Zahnfehlern ist noch vom Signal „ToothFaultEnable“ abhängig.

Die Definition von Zahnfehlern erfolgt über einen eigenen Eingabedialog, wobei einzelne durch Kommata getrennte Zahnnummern oder Zahnbereiche eingegeben werden können. Bereiche werden hierbei in der Form

<erster Zahn> + <-> + <letzter Zahn>

definiert.

Die Nummerierung beginnt bei 0, d. h. für ein Rad mit 48 Zähnen/Polpaaren ist der Bereich 0 – 47.



**Abb. 12-6** Eingabedialog zur Definition von Zahnfehlern

## 12.3.1 Gemeinsame Signale aller Sensoren

---

### Enable

---

Mit diesem Signal wird der Sensor aktiviert bzw. deaktiviert. Im inaktiven Zustand ist der zugehörige Ausgang hochohmig geschaltet. Beim Aktivieren wird der Sensor neu gestartet und der Ausgang an den Spannungs- (beim Typ „DF6“) bzw. den Stromausgang (alle anderen Sensortypen) geschaltet. Beim Sensorneustart wird mit Zahn 0 gestartet.

#### **Hinweis**

*Bitte beachten Sie, dass die Umschaltung des Ausgangs mittels Relais erfolgt und daher gewissen Schalt- bzw. Prellzeiten auftreten.*

### Frequency

---

Mit diesem Signal können Sie die Frequenz des Sensors direkt vorgeben (nur wirksam in der Betriebsart „Frequency“, siehe „Operation Mode“ auf Seite 325).

Bei Sensoren mit Drehrichtungserkennung wird durch einen negativen Wert die Drehrichtung umgekehrt (positiver Wert: vorwärts bzw. Normaldrehrichtung, negativer Wert: rückwärts). Bei Sensoren ohne Drehrichtungserkennung wird das Vorzeichen ignoriert.

### *WheelSpeed, WheelRadius*

---

Mit diesem Signal können Sie die Radgeschwindigkeit und den Radradius vorgeben. In der Betriebsart „WheelSpeed“ wird aus diesen Werten zusammen mit der Zähne-/Polpaarzahl die Frequenz wie folgt berechnet.

$$f [\text{Hz}] = \frac{v [\text{km/h}] \cdot z}{r [\text{m}] \cdot 2\pi \cdot 3,6}$$

In der Betriebsart „Frequency“ haben die Signale keine Wirkung. Für einen negativen Wert der Radgeschwindigkeit gilt dasselbe, wie bei „Frequency“, d.h. für Sensoren mit Drehrichtungserkennung wird eine Drehrichtungsumkehr bzw. Rückwärtsdrehen simuliert.

### *ToothFaultEnable*

---

Mit diesem Signal können Sie die in der Registerkarte „Globals“ gemachten Einstellungen für Zahnfehler aktivieren (= 1) bzw. deaktivieren (= 0). Für jeden Zahn bzw. jedes Polpaar kann dabei festgelegt werden, ob reguläre oder fehlerhafte Ausgabe erfolgen soll. Für alle fehlerhaften Zähne eines Sensors wird ein festgelegter Pegel (beim Typ „DF6“ Spannung, bei anderen Sensortypen Strom) ausgegeben.

### *Übernahme von Signaländerungen an den Ausgängen*

---

Änderungen der Frequenz oder Radgeschwindigkeit werden mit dem Beschreiben des Registers am darauffolgenden Polwechsel (halber Zahn) am jeweiligen Ausgang wirksam. Ein Ausnahme bildet der Sensortyp „DF6“, bei welchem die Änderungen unmittelbar wirksam werden.

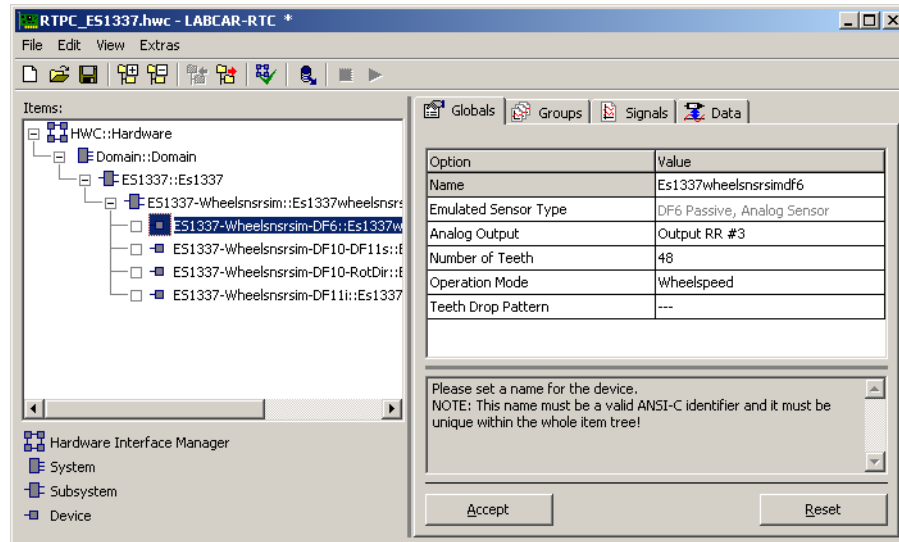
Protokollinformationen (Typ „DF11i“ und „VDA“) werden nicht mit Zahn oder Polwechseln synchronisiert. Es werden die zum Zeitpunkt des Registerzugriffs zur Verfügung stehenden Werte benutzt.

Auch die De-/Aktivierung der Zahnfehlersimulation mittels des Signals „Tooth-FaultEnable“ wird nicht mit Zahn-, Pol- oder Umdrehungswechseln synchronisiert. Damit können bei ungünstigen Umschaltzeitpunkten Teilausgaben von Protokoll- oder Zahninformationen erfolgen.

### 12.3.2 Globals (ES1337-Wheelsnsrsim-DF6 Device)

Mit dem ES1337-Wheelsnsrsim-DF6 Device können Sie an einem der vier Sensorausgänge einen Radsensor des Typs „DF6“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-7** Die Registerkarte „Globals“ des ES1337-Wheelsnsrsim-DF6 Device

*Name*

---

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### *Analog Output*

---

In dieser Option können Sie den Ausgang festlegen, auf welchen der simulierte Sensor geschaltet werden soll.

#### *Number of Teeth*

---

In diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### *Operation Mode*

---

In dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

#### *Teeth Drop Pattern*

---

In dieser Option können Sie ein Zahnfehlermuster definieren. Bei einem als fehlerhaft definierten Zahn wird der Spannungswert 0 V ausgegeben

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Analog Output	uint8	Nein	Verwendeter Ausgang [0 = FL #0   1 = FR #1   2 = RL #2   3 = RR #3]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128]
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-4** ES1337-Wheelsnsrsim-DF6 Device: Konfigurationsparameter der Registerkarte „Globals“

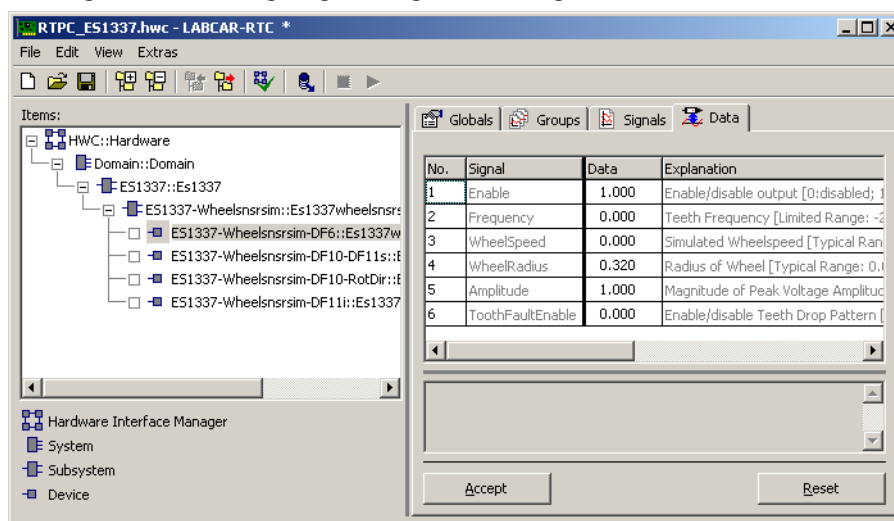
12.3.3 Groups (ES1337-Wheelsnsrsim-DF6 Device)

Das ES1337-Wheelsnsrsim-DF6 Device besitzt eine Signalgruppe „Output“ zur Steuerung der Sensorsimulation.

12.3.4 Data (ES1337-Wheelsnsrsim-DF6 Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung der Sensorsimulation aufgelistet.

Die folgende Abbildung zeigt die Signale der Registerkarte „Data“:



**Abb. 12-8** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-DF6 Device

Enable

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

Frequency

Mit diesem Signal können Sie die Zahnfrequenz direkt vorgeben (nur bei Betriebsart „Frequency“).

WheelSpeed, WheelRadius

Mit diesen Signalen können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (nur bei der Betriebsart „Wheelspeed“).

Amplitude

Über dieses Signal können Sie den Betrag des Spitzenwerts der Sinusspannung am Ausgang einstellen.

ToothFaultEnable

Mit diesem Signal können Sie das in der „Globals“ Registerkarte definierte Zahnfehlermuster aktivieren bzw. deaktivieren.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-2500...2500]
WheelSpeed	real32	Ja	Raddrehzahl [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
Amplitude	real32	Ja	Spitzenamplitude [0.0...10.0 V]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0, 1]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-5** ES1337-Wheelsnsrsim-DF6 Device: Signale der Registerkarte „Data“

Hinweis

Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 2500 Hz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert..

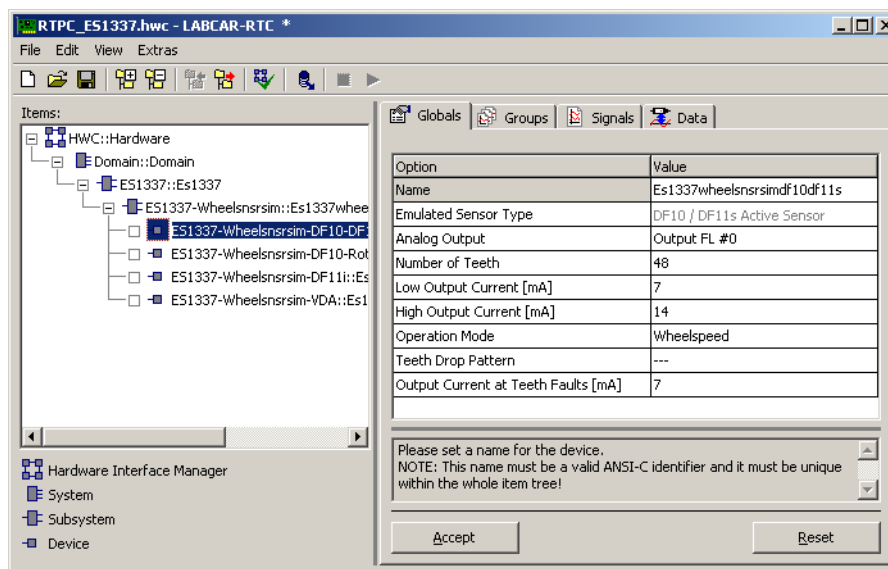
Übernahme von Signaländerungen

Frequenz-, Raddrehzahl- und Amplitudenänderungen werden unmittelbar bei Aufruf der entsprechenden Task wirksam. Die Amplitude ist beim Typ „DF6“ als Signal ausgeführt, um z.B. eine drehzahlabhängige Amplitude realisieren zu können.

### 12.3.5 Globals (ES1337-Wheelsnrsim-DF10-DF11s Device)

Mit dem ES1337-Wheelsnrsim-DF10-DF11s Device können Sie an einem der vier Sensorausgänge einen Radsensor vom Typ „DF10“ bzw. „DF11s“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-9** Die Registerkarte „Globals“ des ES1337-Wheelsnrsim-DF10-DF11s Device

#### Name

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### Analog Output

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

#### Number of Teeth

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### Low Output Current

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### High Output Current

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

#### Operation Mode

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

*Teeth Drop Pattern*

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

*Output Current at Teeth Faults*

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definiertem Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Analog Output	uint8	Nein	Verwendeter Ausgang [0 = FL #0   1 = FR #1   2 = RL #2   3 = RR #3]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 14 mA
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Fault	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-6** ES1337-Wheelsnsrsim-DF10-DF11s Device: Konfigurationsparameter der Registerkarte „Globals“

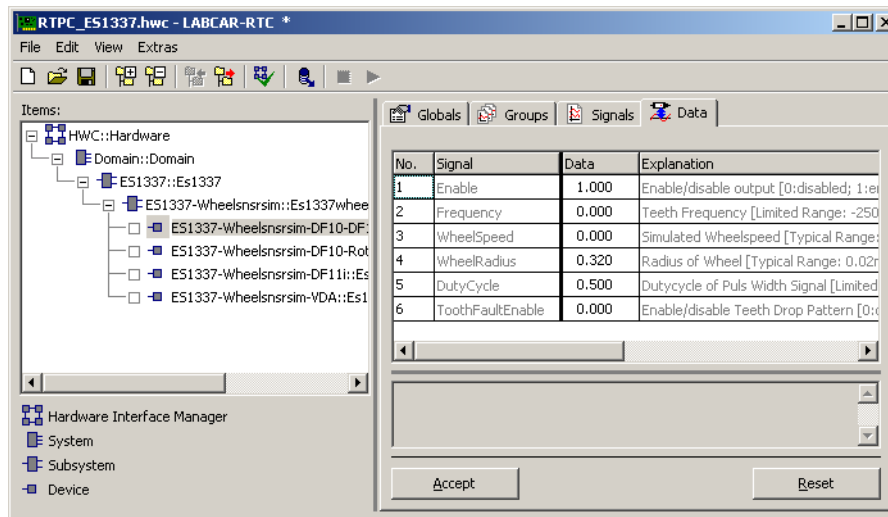
### 12.3.6 Groups (ES1337-Wheelsnsrsim-DF10-DF11s Device)

Das ES1337-Wheelsnsrsim-DF10-DF11s Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

### 12.3.7 Data (ES1337-Wheelsnsrsim-DF10-DF11s Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung der Sensor-simulation aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 12-10** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-DF10-DF11s Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0,5 entsteht ein symmetrisches Ausgangssignal.

#### *ToothFaultEnable*

Mit diesem Signal können Sie das in der „Globals“ Registerkarte definierte Zahnfehlermuster aktivieren bzw. deaktivieren.



Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-2500...2500 Hz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0, 1]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-7** ES1337-Wheelsnsrsim-DF10-DF11s Device: Signale der Registerkarte „Data“

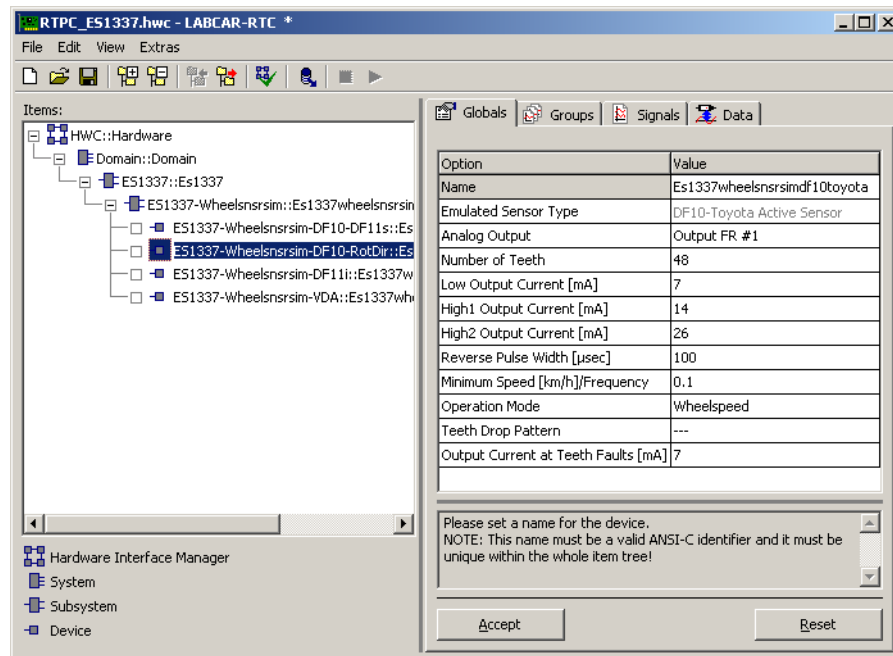
#### Hinweis

*Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 2500 Hz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*

### 12.3.8 Globals (ES1337-Wheelsnsrsim-DF10-RotDir Device)

Mit dem ES1337-Wheelsnsrsim-DF10-RotDir Device können Sie mit einem der vier Sensorausgänge einen Radsensor vom Typ „DF10-RotDir“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-11** Die Registerkarte „Globals“ des ES1337-Wheelsnsrsim-DF10-RotDir Device

#### Name

Sie können einen individuellen Bezeichner für diesen Sensor vergeben.

#### **Hinweis**

*Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen, da dieser Name in den Variablen- und Parameternamen benutzt wird.*

#### Analog Output

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

#### Number of Teeth

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### Low Output Current

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### High1 Output Current

Mit diesem Parameter können Sie den mittleren Stromwert vorgeben.

#### High2 Output Current

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

#### *Reverse Pulse Width*

---

Mit diesem Parameter können Sie die Dauer der Pulse vorgeben, welche bei umgekehrter Drehrichtung an den Polflanken erzeugt wird.

#### *Minimum Speed/Frequency Threshold*

---

Mit dieser Option können Sie die minimale Raddrehzahl (Betriebsart „Wheel-speed“) bzw. Frequenz (Betriebsart „Frequency“) festlegen, ab welcher ein Stillstandverhalten simuliert wird.

#### *Operation Mode*

---

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

#### *Teeth Drop Pattern*

---

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

### Output Current at Teeth Faults

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definiertem Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Analog Output	uint8	Nein	Verwendeter Ausgang [0 = FL #0   1 = FR #1   2 = RL #2   3 = RR #3]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High1 Output Current	uint8	Ja	Mittlerer Stromwert [0...35 mA], Default: 14 mA
High2 Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 28 mA
Reverse Pulse Width	uint8	Ja	Dauer Reverspuls [50...150 µA] Default: 100 µA
Minimum Speed/Frequency Threshold	real32	Ja	Schwelle für Stillstandsverhalten [-100.0...100.0 km/h/Hz]
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Fault	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 12-8** ES1337-Wheelsnsrsim-DF10-RotDir Device: Konfigurationsparameter der Registerkarte „Globals“

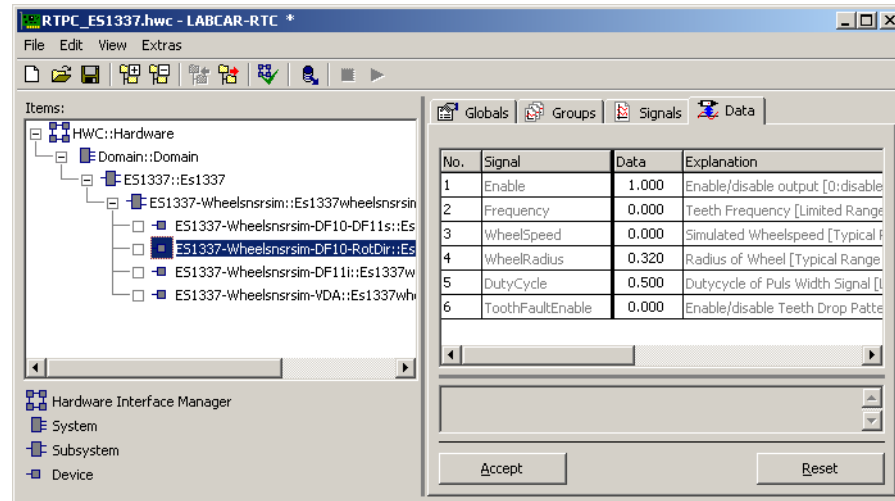
#### 12.3.9 Groups (ES1337-Wheelsnsrsim-DF10-RotDir Device)

Das ES1337-Wheelsnsrsim-DF10-RotDir Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

### 12.3.10 Data (ES1337-Wheelsnsrsim-DF10-RotDir Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung der Sensorsimulation aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 12-12** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-DF10-RotDir Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0.5 entsteht ein symmetrisches Ausgangssignal.

#### *ToothFaultEnable*

Mit diesem Signal können Sie das in der Registerkarte „Globals“ definierte Zahnfehlermuster aktivieren bzw. deaktivieren.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-1600...2500 Hz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
DutyCycle	real32	Ja	Tastverhältnis [0.25...0.75]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0, 1]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-9** ES1337-Wheelsnsrsim-DF10-RotDir Device: Signale der Registerkarte „Data“

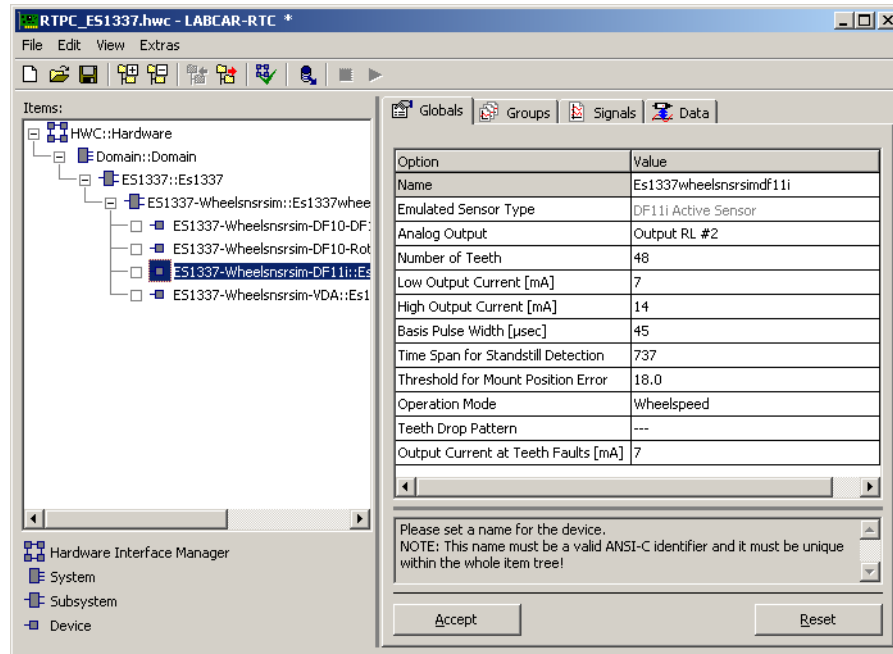
#### Hinweis

*Die resultierende Frequenz wird unabhängig von der Betriebsart auf den Bereich -1600...2500 Hz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*

### 12.3.11 Globals (ES1337-Wheelsnrsim-DF11i Device)

Mit dem ES1337-Wheelsnrsim-DF11i Device können Sie mit einem der vier Sensorausgänge einen Radsensor vom Typ „DF11i“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-13** Die Registerkarte „Globals“ des ES1337-Wheelsnrsim-DF11i Device

#### Name

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### Analog Output

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

#### Number of Teeth

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### Low Output Current

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### High Output Current

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

### *Basic Pulse Width*

---

Mit diesem Parameter können Sie die Basispulsweite vorgeben. Andere Pulsbreiten sind von diesem Wert abgeleitet wie z.B.

- Luftspaltreserve
- Drehrichtung Links
- Drehrichtung Rechts
- Drehrichtung Links und Einbaulage
- Drehrichtung Rechts und Einbaulage

### *Timespan for Standstill Detection*

---

Mit diesem Parameter können Sie die Zeitdauer vorgeben, nach der der Stillstand des Rades erkannt und der Stillstandsimpuls ausgegeben wird. Die Stillstandsüberwachung wird mit einem Zahnimpuls oder Stillstandsimpuls zurückgesetzt, d.h. bei Stillstand werden Stillstandsimpulse in diesen Zeitabständen ausgegeben.

### *Threshold for Mount Position Error*

---

Mit dieser Option können Sie den Grenzwert zur Ausgabe eines Lagefehlers über die Raddrehzahl (Betriebsart „Wheelspeed“) bzw. die Frequenz (Betriebsart „Frequency“) festlegen. Unterschreitet die Raddrehzahl bzw. die Frequenz diesen Wert, so wird das Fehlersignal gesetzt.

### *Operation Mode*

---

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

### *Teeth Drop Pattern*

---

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

### *Output Current at Teeth Faults*

---

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definiertem Zahn ausgegeben werden soll.



Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Analog Output	uint8	Nein	Verwendeter Ausgang [0 = FL #0   1 = FR #1 2 = RL #2   3 = RR #3]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0 ...35 mA], Default: 14 mA
Basic Pulse Width	uint8	Ja	Basispulsdauer [37...53 µs], Default: 45 µs
Minimum Speed/ Frequency Threshold	real32	Ja	Schwelle für Stillstandsverhalten [-100.0...100.0 km/h/Hz]
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Fault	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-10** ES1337-Wheelsnsrsim-DF11i Device: Konfigurationsparameter der Registerkarte „Globals“

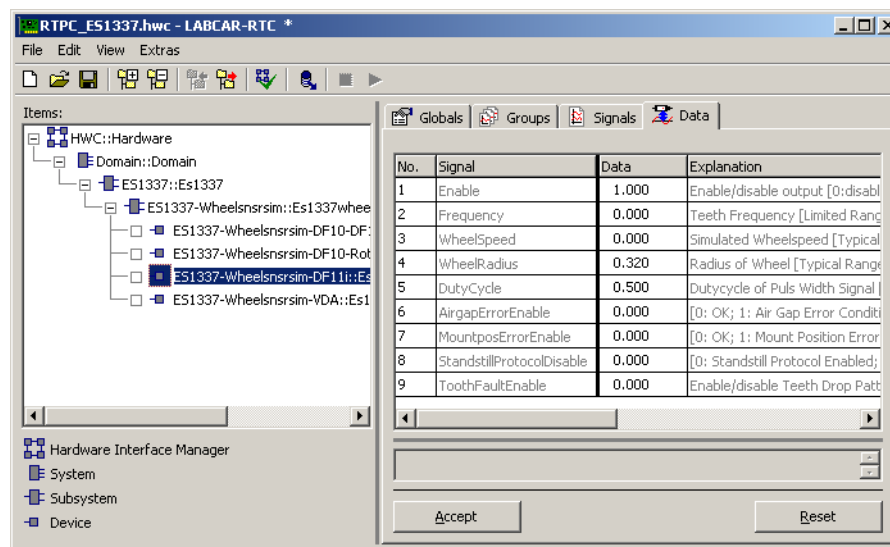
#### 12.3.12 Groups (ES1337-Wheelsnsrsim-DF11i Device)

Das ES1337-Wheelsnsrsim-DF11i Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

### 12.3.13 Data (ES1337-Wheelsnsrsim-DF11i Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung der Sensorsimulation aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 12-14** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-DF11i Device  
*Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0.5 entsteht eine symmetrisches Ausgangssignal.

#### *AirgapErrorEnable*

Mit diesem Signal können Sie die Bedingung „Luftspaltfehler“ aktivieren bzw. deaktivieren.

#### *MountposErrorEnable*

Mit diesem Signal können Sie die Bedingung „Lagefehler“ aktivieren bzw. deaktivieren.

StandstillProtocolDisable

Mit diesem Signal können Sie das für diesen Sensor bei Stillstand definierte Verhalten aktivieren bzw. deaktivieren.

ToothFaultEnable

Mit diesem Signal können Sie das in der „Globals“ Registerkarte definierte Zahnfehlermuster aktivieren bzw. deaktivieren.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-10000...10000 Hz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95]
AirgapError-Enable	real32	Ja	Ausgabe Bedingung „Luftspaltfehler“ [0.0 = Aus, 1.0 = Aktiv]
MountposError-Enable	real32	Ja	Aktivierung Lagefehler [0.0 = Aus, 1.0 = Aktiv]
Standstill-ProtocolDisable	real32	Ja	Aktivierung Stillstandprotokoll [0.0 = Aus, 1.0 = Aktiv]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0 = Aus, 1 = Aktiv]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-11** ES1337-Wheelsnsrsim-DF11i Device: Signale der Registerkarte „Data“

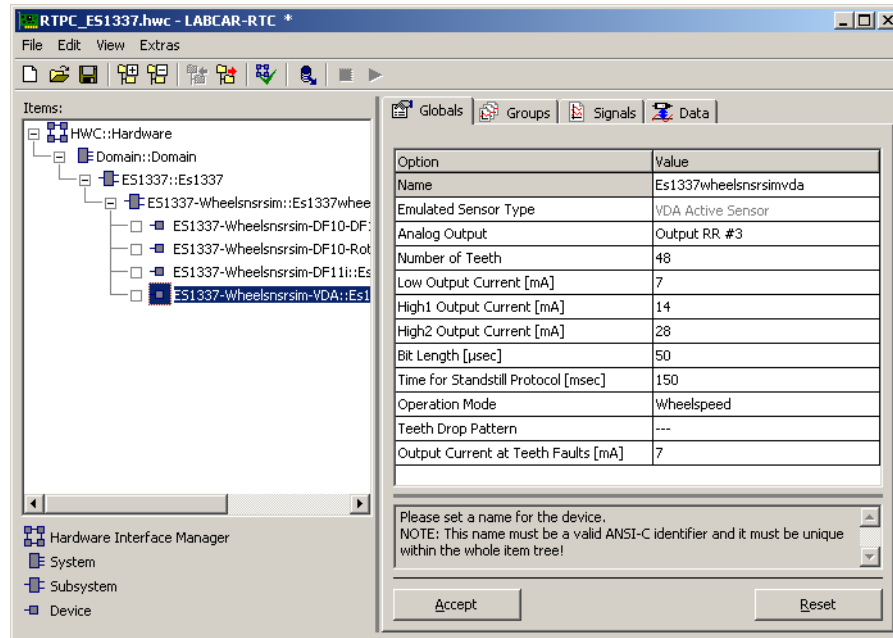
Hinweis

Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 10000 Hz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.

### 12.3.14 Globals (ES1337-Wheelsnsrsim-VDA Device)

Mit dem ES1337-Wheelsnsrsim-VDA Device können Sie mit einem der vier Sensorausgänge einen Radsensor des Typs „VDA“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 12-15** Die Registerkarte „Globals“ des ES1337-Wheelsnsrsim-VDA Device

#### Name

Ein eindeutiger Bezeichner für dieses Device.

#### Hinweis

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### Analog Output

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

#### Number of Teeth

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### Low Output Current

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### High1 Output Current

Mit diesem Parameter können Sie den mittleren Stromwert vorgeben.

#### High2 Output Current

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

### *Bit Length*

---

Mit diesem Parameter können Sie die Basispulswerte vorgeben. Weitere Pulswerte sind von diesem Wert abgeleitet wie

- Geschwindigkeitspuls
- Datenpuls
- Vorbit (Initial Bit)

### *Time for Standstill Protocol*

---

Mit diesem Parameter können Sie die Zeitdauer vorgeben, nach der der Stillstand des Rades erkannt und der Stillstandsimpuls ausgegeben wird. Die Stillstandsüberwachung wird mit einem Zahnimpuls oder Stillstandsimpuls zurückgesetzt, d.h. bei Stillstand werden Stillstandsimpulse in diesen Zeitabständen ausgegeben.

### *Operation Mode*

---

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

### *Teeth Drop Pattern*

---

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

### *Output Current at Teeth Faults*

---

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definierten Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Analog Output	uint8	Nein	Verwendeter Ausgang [0 = FL #0   1 = FR #1 2 = RL #2   3 = RR #3]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 14 mA
Bit Length	uint8	Ja	Basispulsdauer [20...80 µs], Default: 50 µs
Time for Standstill Protocol	uint8	Ja	Schwelle für Stillstandsverhalten [-100.0 ... 100.0 km/h/Hz]
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Faults	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 12-12** ES1337-Wheelsnsrsim-VDA Device: Konfigurationsparameter der Registerkarte „Globals“

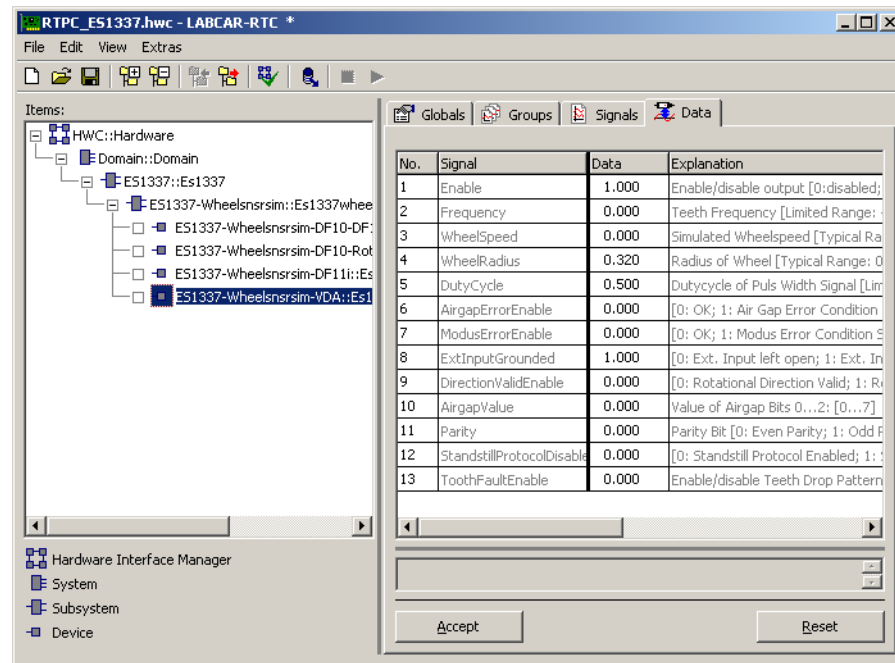
#### 12.3.15 Groups (ES1337-Wheelsnsrsim-VDA Device)

Das ES1337-Wheelsnsrsim-VDA Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

### 12.3.16 Data (ES1337-Wheelsnsrsim-VDA Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung der Sensorsimulation aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 12-16** Die Registerkarte „Data“ des ES1337-Wheelsnsrsim-VDA Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0,5 entsteht ein symmetrisches Ausgangssignal.

#### *AirgapErrorEnable*

Mit diesem Signal können Sie die Bedingung „Luftspaltfehler“ aktivieren bzw. deaktivieren (Protokollbit 0).

#### *ModusErrorEnable*

Mit diesem Signal können Sie die Bedingung „Modusfehler“ aktivieren bzw. deaktivieren (Protokollbit 1).

*ExtInputGrounded*

Mit diesem Signal können Sie das zugehörige Protokollbit 2 beeinflussen. Eine Null signalisiert „Externer Eingang offen“.

*DirectionValidEnable*

Mit diesem Signal können Sie das zugehörige Protokollbit 3 beeinflussen. Eine Null signalisiert, dass die übertragene Drehrichtung gültig ist.

*AirgapValue*

Mit diesem Signal können Sie den Wert der zugehörigen Protokollbits 5..7 beeinflussen. Der Luftspaltwert kann sich im Bereich zwischen 0 und 7 bewegen.

*Parity*

Mit diesem Signal können Sie die Paritätsbildung beeinflussen (Protokollbit 8). Ein Wert von Null entspricht gerader Parität.

Die Bedeutung der Protokollbits ist in der folgenden Tabelle zusammengefasst.

Bit	Bedeutung	Kodierung
0	Fehlerbit Luftspaltreserve	0 = korrekt 1 = Luftspaltreserve
1	Frei vergebbar	
2	Frei vergebbar	
3	Gültigkeit der Drehrichtung	0 = gültig 1 = ungültig
4	Drehrichtung - wird aus dem Vorzeichen von „WheelSpeed“ oder „Frequency“ berechnet	0 = positiv 1 = negativ
5	Frei vergebbar	
6	Frei vergebbar	
7	Frei vergebbar	
8	Paritäts-Bit	Wird auf 0 oder 1 gesetzt, um gerade Parität (inkl. dem Paritäts-Bit selbst) beizubehalten.

**Tab. 12-13** Bedeutung der neun Protokollbits

*StandstillProtocolDisable*

Mit diesem Signal können Sie das Stillstandsprotokollverhalten des Sensors beeinflussen

*ToothFaultEnable*

Mit diesem Signal können Sie das in der Registerkarte „Globals“ definierte Zahnfehlermuster aktivieren bzw. deaktivieren.



Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-5000...5000 Hz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95]
AirgapErrorEnable	real32	Ja	Aktivierung „Luftspaltfehler“ [0.0 = aus, 1.0 = aktiv]
ModusErrorEnable	real32	Ja	Aktivierung „Lagefehler“ [0.0 = aus, 1.0 = aktiv]
ExtInputGrounded	real32	Ja	Pegel ext. Eingang [0.0 = offen, 1.0 = Ground]
DirectionValidEnable	real32	Ja	Signalisierung Drehrichtung gültig [0.0 = Drehrichtung gültig, 1.0 = Drehrichtung ungültig]
AirgapValue	real32	Ja	Wert Luftspalt [0...7]
Parity	real32	Ja	Einstellung Parität [0.0 = gerade, 1.0 = ungerade]
Standstill-ProtocolDisable	real32	Ja	Aktivierung Stillstandprotokoll [0.0 = aktiv, 1.0 = inaktiv]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0 = aus, 1 = aktiv]
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 12-14** ES1337-Wheelsnsrsim-VDA Device: Signale der Registerkarte „Data“

#### Hinweis

*Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 5000 Hz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*



## 13 ES1385 Resistor Cascade Board

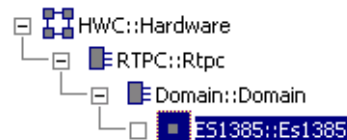
Das ES1385 Resistor Cascade Board dient im LABCAR-Umfeld als Widerstandskarte zur Simulation von Temperatur- bzw. Lambda-Sensoren.

### Hinweis

Was die Werte der Widerstandskaskaden anbelangt, gibt es mehrere Varianten des ES1385 Resistor Cascade Boards – die genaue Spezifikation finden Sie im jeweiligen Benutzerhandbuch.

### Aufbau des ES1385 RTIO-Baums

Ein ES1385 Resistor Cascade Board wird im VMEbus-Baugruppenträger verbaut und ist dort direkt über den VMEbus mit dem Experimentaltarget verbunden. Die beschriebene Hardware-Anbindung spiegelt sich im RTIO-Baum wider.

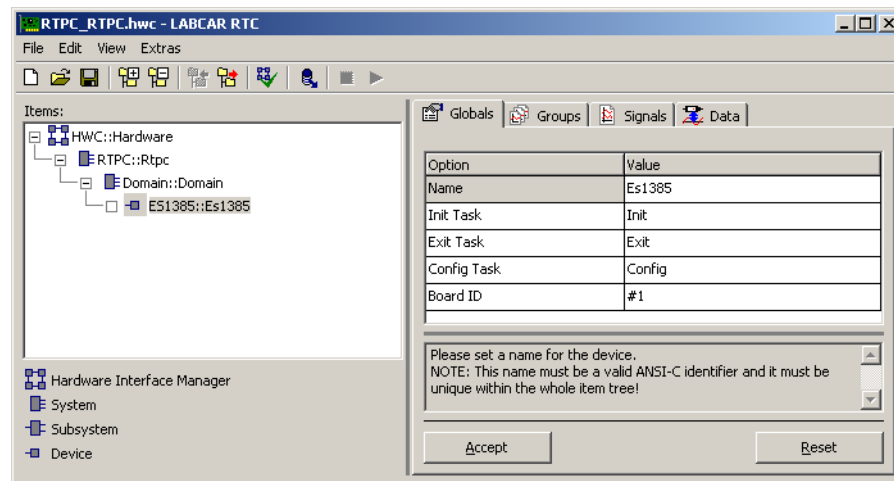


**Abb. 13-1** RTIO-Hardwarebeschreibung mit eingebundener ES1385

### 13.1 ES1385 Device

#### 13.1.1 Globals (ES1385 Device)

Die Registerkarte „Globals“ eines ES1385 Device ist in Abb. 13-2 dargestellt.



**Abb. 13-2** Die Registerkarte „Globals“ des ES1385 Device

### Hinweis

Wenn das ES1385 Resistor Cascade Board in eine LABCAR-RTC-Hardwarekonfiguration eingebunden wird, müssen die beiden Adress-Schalter SW1 und SW2 auf dem Board die Einstellung „0x00“ haben!

*Board ID*

Dieses Optionsfeld dient zur Identifizierung des ES1385 Resistor Cascade Board. Es stellt die Zuordnung zwischen der RTIO-Hardwarebeschreibung und einer bestimmten ES1385 im VMEbus-Baugruppenträger her. Die Nummerierung der ES1385 im Baugruppenträger erfolgt dabei von links nach rechts beginnend mit 1. Es können bis zu 20 ES1385 pro Baugruppenträger eingebunden werden.

Dieser RTIO-Parameter ist nicht zur Laufzeit einstellbar.

Tab. 13-1 fasst die Eigenschaften des „Board ID“-Parameters zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Board ID	uint32	Nein	Identifizier der ES1385. Zur Auswahl stehen #1 bis #20.

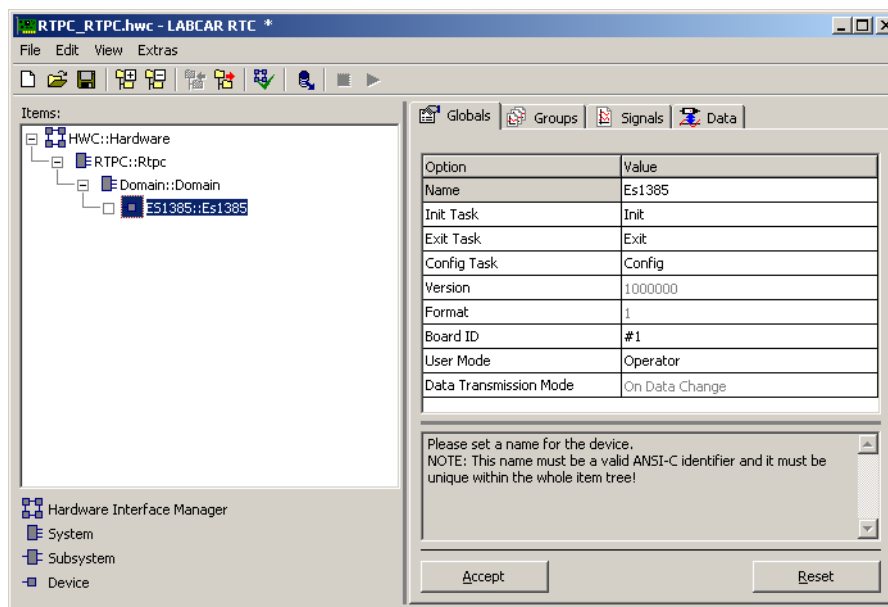
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-1** ES1385 Device: Konfigurationsparameter der Registerkarte „Globals“

*Versteckte Optionsfelder*

Weitere Optionsfelder (siehe Abb. 13-3) können sichtbar gemacht werden, wenn Sie mit der rechten Maustaste in die Registerkarte „Globals“ klicken und **Show All** wählen.

Diese Optionsfelder erlauben es ETAS-Servicepersonal im Falle einer Fehlersuche zusätzliche Informationen über das ES1385 Resistor Cascade Board zu erhalten. Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten daher vom Anwender auch nicht geändert werden.



**Abb. 13-3** Die Registerkarte „Globals“ des ES1385 Device mit allen Optionsfeldern

### User Mode

---

Mit dem Parameter „User Mode“ wird RTIO-Anwendermodus eingestellt. Wählen Sie in diesem Listenfeld die Option „Operator“.

Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Im Modus „Supervisor“ werden in der Initialisierungsphase des RTIO-Treibers die Daten des nichtflüchtigen Speichers (wie z.B. die Kalibrierungswerte der Kaskaden) ausgelesen und dem Anwender in LABCAR-OPERATOR im Fenster „Hardware Info“ sichtbar gemacht. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

### Data Transmission Mode

---

Dieses Listenfeld ist nur editierbar, falls im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeit-Daten vom Experimentaltarget zum ES1385 Resistor Cascade Board übertragen werden.

Ist die Option „On Data Change“ aktiviert, werden die Daten nur dann übertragen, wenn sich wenigstens ein Datum geändert hat. Ist die Option „Every Interval“ aktiviert, so werden die Echtzeit-Daten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.

In Tab. 13-2 sind die Eigenschaften der versteckten RTIO-Parameter der Registerkarte „Globals“ zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
User Mode	uint8	Ja	Anwender-Modus 0: Operator 1: Supervisor „Operator“ ist Voreinstellung. „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche.
Data Transmission Mode	uint8	Ja	Übertragungsmodus der Echtzeit-Daten 0: On Data Change 1: Every Interval „On Data Change“ ist Voreinstellung. Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-2** ES1385 Device: Versteckte Konfigurationsparameter der Registerkarte „Globals“

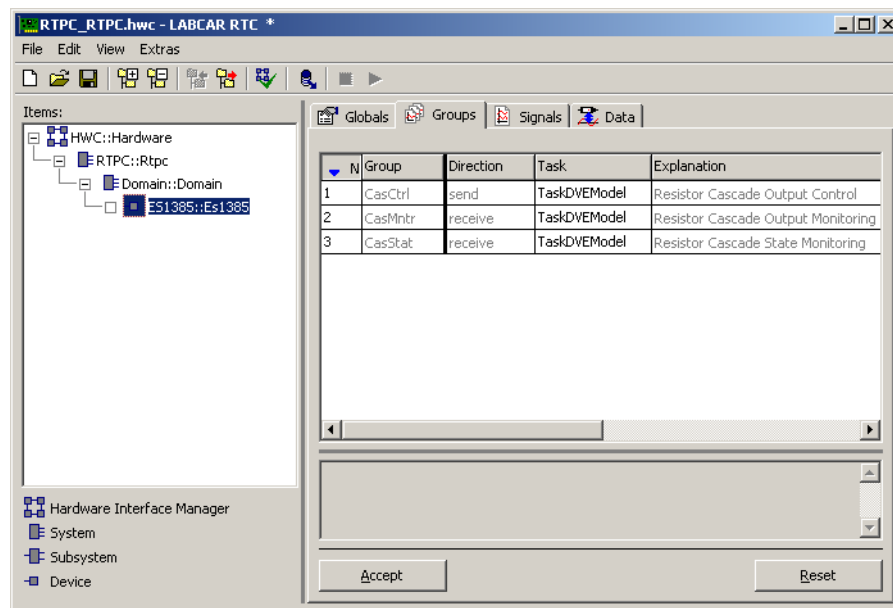
### 13.1.2 Groups (ES1385 Device)

Das ES1385 Device besitzt drei Signalgruppen, die vom Experimentaltarget zum ES1385 Resistor Cascade Board bzw. von der ES1385 zum Experimentaltarget übertragen werden (siehe Abb. 13-4 auf Seite 354).

- Die **Signalgruppe „CasCtrl“** transportiert die vom Anwender spezifizierten und von der RTIO aufbereiteten Widerstandswerte der Kaskaden zur ES1385. Die vom Anwender spezifizierten Widerstandswerte werden unter Berücksichtigung der Kalibrierung und Monotonie mit Hilfe eines Näherungsverfahrens zum real möglichen Widerstandswert akkumuliert. Erst danach erfolgt die Übertragung zur Hardware.
- Die **Signalgruppe „CasMntr“** stellt die vom RTIO Low-Level-Treiber gebildeten realen Widerstandswerte für den Anwender dar. Die Aktivierungsperiode dieser Signalgruppe sollte der Aktivierungsperiode der Signalgruppe „CasCtrl“ entsprechen.
- Die **Signalgruppe „CasStat“** transportiert aktuelle Zustandsinformationen der unabhängigen Widerstandskaskaden zur RTIO. Dadurch lassen sich die einzelnen Fehlerzustände der Hardware über die RTIO darstellen bzw. überwachen.

Jeder Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erzeugenden Widerstandswerte bzw. der zu überwachenden Statusinformationen.

Abb. 13-4 zeigt die Registerkarte „Groups“ eines ES1385 Device.



**Abb. 13-4** Die Registerkarte „Groups“ des ES1385 Device

*RTIO-Signale der Signalgruppe „CasCtrl“*

Die Signalgruppe „CasCtrl“ umfasst 6 RTIO-Signale „Ctrl\_R\_0“ bis „Ctrl\_R\_5“, die vom Datentyp „uint32“ sind. Wird die Widerstandskaskade  $x$  ( $x = 0, 1, 2, 3, 4, 5$ ) dahingehend konfiguriert, dass sie von der RTIO aus gesteuert wird, so definiert der Wert des RTIO-Signals „Ctrl\_R\_x“ den vom Anwender spezifizierten Soll-Widerstandswert der zugehörigen Kaskade in Ohm.

Wird die Widerstandskaskade nicht von der RTIO gesteuert, so hat das Signal keine Bedeutung.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Ctrl_R_0 bis Ctrl_R_5	uint32	Vom Anwender definierter Widerstandswert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-3** ES1385 Device: Die RTIO-Signale der Signalgruppe „CasCtrl“

*RTIO-Signale der Signalgruppe „CasMntr“*

Die Signalgruppe „CasMntr“ umfasst 6 RTIO-Signale „Mntr\_R\_0“ bis „Mntr\_R\_5“, die vom Datentyp „real32“ sind. Sie wird dazu verwendet, um den zur Hardware übertragenen Ist-Widerstandswert einer Kaskade darzustellen. Jeder vom Anwender definierte Widerstandswert muss unter Berücksichtigung der Kalibrierungswerte und Monotonie mit Hilfe eines Näherungsverfahrens gebildet werden.

Tab. 13-4 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Mntr_R_0 bis Mntr_R_5	real32	Tatsächlicher zur Hardware übertragener Widerstandswert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-4** ES1385 Device: Die RTIO-Signale der Signalgruppe „CasMntr“

*RTIO-Signale der Signalgruppe „CasStat“*

Die Signalgruppe „CasStat“ umfasst 6 RTIO-Signale „Stat\_C\_0“ bis „Stat\_C\_5“, die vom Datentyp „uint16“ sind. Sie wird dazu verwendet, um den aktuellen Zustand der Hardwarekanäle darzustellen.

Tab. 13-5 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Stat_C_0 bis Stat_C_5	uint16	Zustand der Widerstandskaskade 0: kein Fehler 1: Überstrom wurde erfasst

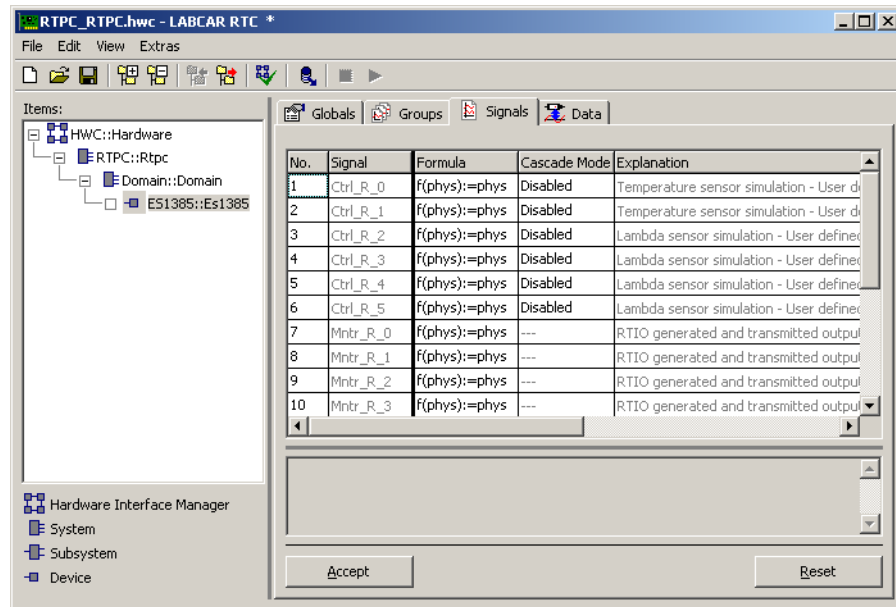
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-5** ES1385 Device: Die RTIO-Signale der Signalgruppe „CasStat“

13.1.3 Signals (ES1385 Device)

In der Registerkarte „Signals“ wird die Konfiguration der Widerstandskaskaden einer ES1385 durchgeführt.

Abb. 13-5 zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 13-5** Die Registerkarte „Signals“ des ES1385 Device

Cascade Mode

Dieses Optionsfeld dient zur Konfiguration der Widerstandskaskade. Mit „Disabled“ wird die Kaskade x (x = 0, 1, 2, 3, 4, 5) deaktiviert, d.h. alle Relais sind geöffnet. In diesem Zustand können keine Widerstandswerte eingegeben und zur Hardware übermittelt werden.

„Enabled“ aktiviert die Kaskade x (x = 0, 1, 2, 3, 4, 5), d.h. nur in diesem Zustand können Widerstandswerte über die RTIO zur Hardware transportiert werden.

Tab. 13-6 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Cascade Mode	uint8	Ja	Konfiguriert die Widerstandskaskaden: 0: „Disabled“, Kaskade ist deaktiviert 1: „Enabled“, Kaskade ist aktiviert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 13-6** ES1385 Device: Konfigurationsparameter der Registerkarte „Signals“



## 14 ES1391.1 Power Supply Controller Board

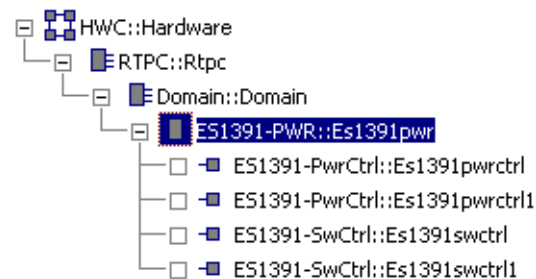
Das ES1391.1 Power Supply Controller Board dient der Fernsteuerung von externen Netzteilen zur Bereitstellung der Batteriespannung für LABCAR-Anwendungen und (in Verbindung mit den ES1392 High Current Switch Boards) zum Schalten der Batterieknoten im LABCAR.

Neben diesen beiden speziellen Anwendungen ist die ES1391.1 ganz allgemein für Aufgaben geeignet, die die Erfassung und Ausgabe von unipolaren, analogen Signalen sowie die Erfassung und Ausgabe von Digitalsignalen erfordern.

Beim Entwurf der RTIO-Benutzeroberfläche für die ES1391.1 wurde Wert auf eine einfache, intuitive Bedienbarkeit für den Anwender gelegt. Das gilt sowohl für die Anwendungen des Boards zur Fernsteuerung von Netzteilen und ES1392 High Current Switch Boards als auch für den allgemeineren Fall der analogen und digitalen Ein- und Ausgabe.

### *Aufbau des ES1391.1-RTIO-Baums*

Im RTIO-Editor wird die ES1391.1 durch Auswahl des ES1391-PWR Device eingebunden.



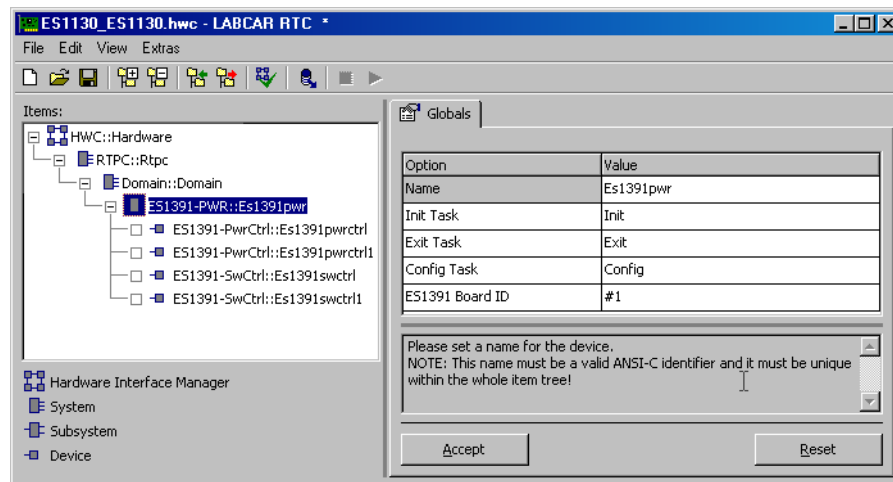
**Abb. 14-1** RTIO-Hardwarebeschreibung mit eingebundener ES1391.1

Die ES1391.1 bietet zwei identische Funktionseinheiten zur Fernsteuerung von externen Netzteilen und zwei identische Funktionseinheiten zur Steuerung von ES1392 High Current Switch Boards. Dieser hardwaretechnische Aufbau spiegelt sich im RTIO-Baum wider. Der Anwender hat die Möglichkeit, bis zu zwei ES1391-PwrCtrl Device zur Fernsteuerung externer Netzteile und bis zu zwei ES1391-SwCtrl Device zur Steuerung von ES1392 High Current Switch Boards einzubinden.

## 14.1 ES1391-PWR Subsystem

### 14.1.1 Globals (ES1391-PWR Subsystem)

Abb. 14-2 zeigt die RTIO-Parameter in der Registerkarte „Globals“.



**Abb. 14-2** Die Registerkarte „Globals“ des ES1391-PWR Subsystems

#### Hinweis

Wenn das ES1391.1 Power Supply Controller Board in eine LABCAR-RTC-Hardwarekonfiguration eingebunden wird, müssen die beiden Adress-Schalter SW1 und SW2 auf dem Board die Einstellung „0x00“ haben!

#### ES1391 Board ID

Dieses Listenfeld dient zur Identifizierung der ES1391.1. Es stellt die Zuordnung her zwischen der RTIO-Hardwarebeschreibung und der ES1391.1 im VMEbus-Baugruppenträger, für die diese Beschreibung gültig ist. Die Nummerierung der ES1391.1 Boards im Baugruppenträger erfolgt dabei von links nach rechts (links = VMEbus-Slot Nr. 1, in Richtung ansteigender Slot-Nummerierung) beginnend mit 1. Die ermittelte Nummer des ES1391.1 Boards ist in dem Listenfeld „ES1391 Board ID“ einzustellen. Dieser RTIO-Parameter ist nicht zur Laufzeit editierbar. Im RTIO-Editor können bis zu 20 ES1391.1-Boards pro Baugruppenträger eingebunden werden.

Tab. 14-1 fasst die Eigenschaften des „Board ID“-Parameters zusammen.

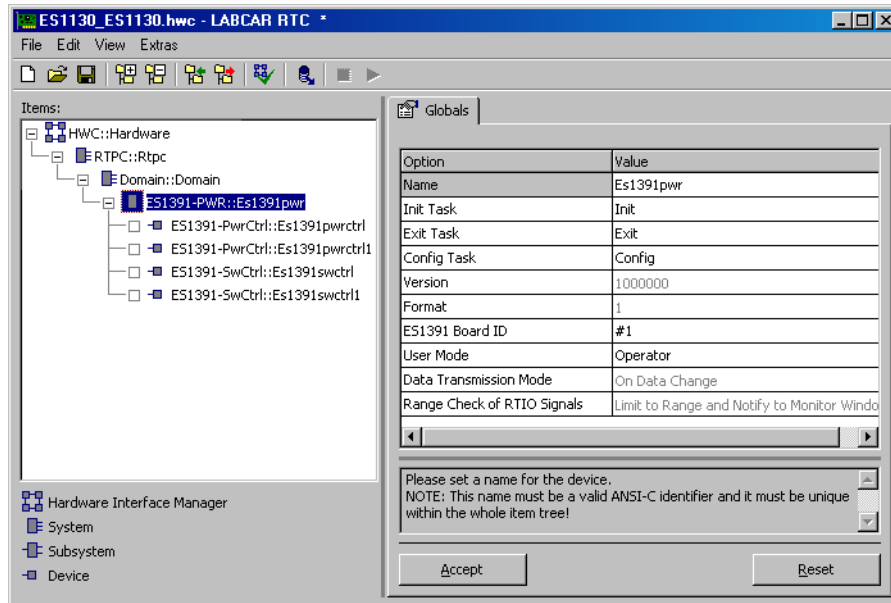
Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
ES1391 Board ID	uint32	Nein	Identifizierung der ES1391.1. Zur Auswahl stehen #1 bis #20.
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 14-1** ES1391-PWR Subsystem: Konfigurationsparameter der Registerkarte „Globals“

### Versteckte Optionsfelder

Weitere Optionsfelder (siehe Abb. 14-3) können sichtbar gemacht werden, wenn Sie mit der rechten Maustaste in die Registerkarte „Globals“ klicken und die **Show All** wählen.

Diese Optionsfelder erlauben es ETAS-Servicepersonal im Falle einer Fehlersuche zusätzliche Informationen über die ES1391.1 zu erhalten. Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten daher vom Anwender auch nicht geändert werden.



**Abb. 14-3** Die Registerkarte „Globals“ des ES1391-PWR Subsystems mit allen Optionsfeldern

### User Mode

Mit dem Parameter „User Mode“ wird RTIO-Anwendermodus eingestellt. Wählen Sie in diesem Listenfeld die Option „Operator“.

Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Unter anderem werden bei aktiviertem „Supervisor“-Modus die Inhalte des nicht flüchtigen Datenspeichers der ES1391.1 und die Inhalte des EEPROMs eventuell angeschlossener ES1392 High Current Switch Boards während der Initialisierung in das „Target Debug“-Fenster ausgegeben.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

### Data Transmission Mode

Dieses Listenfeld ist nur editierbar, falls im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeit-Daten vom Experimentaltarget zur ES1391.1 übertragen werden. Ist die Option „On Data Change“ aktiviert, werden die Daten nur dann übertragen, wenn sich wenigstens ein Datum geändert hat. Ist die Option „Every Interval“ aktiviert, so werden die Echtzeit-Daten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.

#### *Range Check of RTIO Signals*

Dieses Listenfeld ist nur editierbar, falls im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Mit ihm wird festgelegt, in welchem Umfang die RTIO-Signale zur Laufzeit einer Wertebereich-Prüfung unterzogen werden.

Wird die Option „No Check“ eingestellt, so findet keine -Prüfung des Wertebereichs statt.

Ist die Option „Limit to Range“ aktiviert, so werden die RTIO-Signale auf ihren zulässigen Wertebereich begrenzt. Ist zum Beispiel für das RTIO-Signal x ein Wertebereich zwischen a und b definiert, so wird auf a begrenzt, falls x kleiner als a ist und auf b, falls x größer als b ist. Ist die Option „Limit to Range and Notify to Monitor Window“ eingestellt, so werden die RTIO-Signale auf ihren zulässigen Wertebereich begrenzt und zusätzlich wird noch eine entsprechende Nachricht im Fehlerfenster („Monitor Window“) ausgegeben.

Das Listenfeld „Range Check of RTIO Signals“ ist nicht zur Laufzeit editierbar und hat die Voreinstellung „Limit to Range and Notify to Monitor Window“.

In Tab. 14-2 sind die Eigenschaften der versteckten RTIO-Parameter der Registerkarte „Globals“ zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
User Mode	uint8	Ja	Anwender-Modus 0: Operator 1: Supervisor „Operator“ ist Voreinstellung. „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche.
Data Transmission Mode	uint8	Ja	Übertragungsmodus der Echtzeitdaten 0: On Data Change 1: Every Interval „On Data Change“ ist Voreinstellung. Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“.
Range Check of RTIO Signals	-	Nein	Wertebereichsprüfung für RTIO-Signale. „Limit to Range and Notify to Monitor Window“ ist Voreinstellung. Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

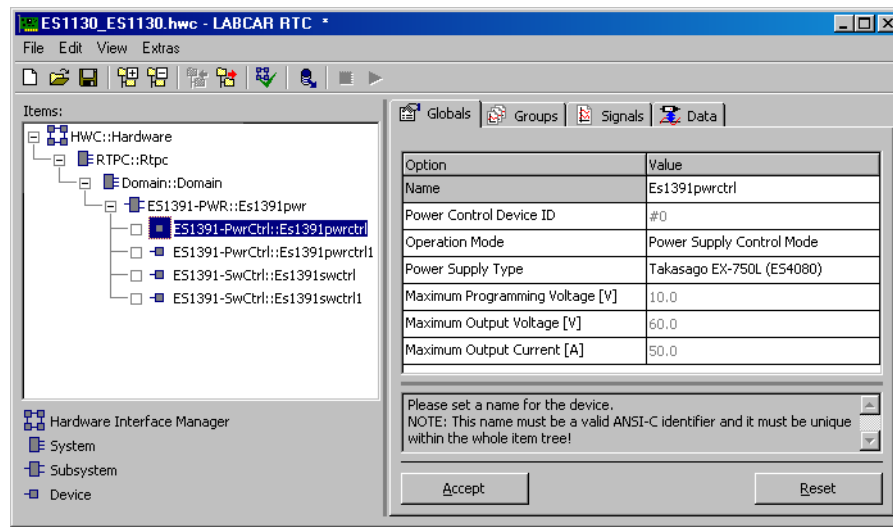
**Tab. 14-2** ES1391-PWR Subsystem: Versteckte Konfigurationsparameter der Registerkarte „Globals“

## 14.2 ES1391-PwrCtrl Device

### 14.2.1 Globals (ES1391-PwrCtrl Device)

Das ES1391-PwrCtrl Device dient zur Fernsteuerung eines externen Netzteils, es kann jedoch auch für Aufgaben der analogen und digitalen Ein- und Ausgabe konfiguriert werden.

Die Registerkarte „Globals“ des ES1391-PwrCtrl Device ist in Abb. 14-4 dargestellt.



**Abb. 14-4** Die Registerkarte „Globals“ des ES1391-PwrCtrl Device

#### *Power Control Device ID*

Dieses Listenfeld ordnet das ES1391-PwrCtrl Device einem der beiden Einheiten zur Netzteil-Fernsteuerung zu.

Das Listenfeld ist nicht editierbar, sein Wert wird automatisch eingetragen. Der Optionswert „#0“ bezeichnet Hardware-Einheit „PwrCtrl0“ und der Optionswert „#1“ bezeichnet Hardware-Einheit „PwrCtrl1“.

#### *Operation Mode*

Dieses Listenfeld legt fest, in welchem Modus das ES1391-PwrCtrl Device und die zugehörige Hardware-Einheit betrieben werden. Es stehen zwei Modi zur Auswahl:

- Die Einstellung „Power Supply Control Mode“ konfiguriert das ES1391-PwrCtrl Device für die Aufgabe der Netzteil-Fernsteuerung.
- Die Einstellung „Analog & Digital IO Mode“ konfiguriert das ES1391-PwrCtrl Device für die Aufgaben der analogen und digitalen Ein- und Ausgabe.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Power Supply Control Mode“.

### *Power Supply Type*

---

In diesem Listenfeld wird der Typ des zu steuernden externen Netzteils angegeben. Das Listenfeld ist nur dann editierbar, wenn im Listenfeld „Operation Mode“ die Einstellung „Power Supply Control Mode“ gewählt wurde. Im Betriebsmodus „Analog & Digital IO Mode“ hat es keine Bedeutung.

Neben einigen vordefinierten Netzteilen kann in diesem Listenfeld die Option „Other Power Supply Types“ zur Fernsteuerung eines kundenspezifischen externen Netzteils eingestellt werden. Das Listenfeld ist zur Laufzeit editierbar.

### *Maximum Programming Voltage [V]*

---

Dieses und die folgenden beiden numerischen Eingabefelder dienen zur Beschreibung der Eigenschaften eines kundenspezifischen externen Netzteils.

Sie sind nur dann editierbar, wenn das ES1391-PwrCtrl Device für die Steuerung eines externen Netzteil konfiguriert ist und im Listenfeld „Power Supply Type“ die Einstellung „Other Power Supply Types“ aktiv ist. In allen anderen Fällen werden die Werte in diesen numerischen Eingabefeldern automatisch gesetzt. Alle drei Eingabefelder sind zur Laufzeit editierbar.

Im Eingabefeld „Maximum Programming Voltage [V]“ ist die maximale Programmierspannung des externen Netzteils anzugeben. Es ist zu beachten, dass Programmierspannungen größer als 10 V von der ES1391.1-Hardware nicht generiert werden können.

### *Maximum Output Voltage [V]*

---

Im Eingabefeld „Maximum Output Voltage [V]“ ist die maximale Ausgangsspannung des Netzteils anzugeben. Das ist die Spannung, die das Netzteil bei einer Programmierspannung gleich der maximalen Programmierspannung im CV-Modus (CV = Constant Voltage) an seinem Ausgang abgibt.

### *Maximum Output Current [A]*

---

Im Eingabefeld „Maximum Output Current [A]“ ist der maximale Ausgangsstrom des Netzteils anzugeben. Das ist der Strom, den das Netzteil bei einer Programmierspannung gleich der maximalen Programmierspannung im CC-Modus (CC = Constant Current) an seinem Ausgang abgibt.

In Tab. 14-3 sind die Eigenschaften der einzelnen RTIO-Parameter der Registerkarte „Globals“ zusammengefasst.

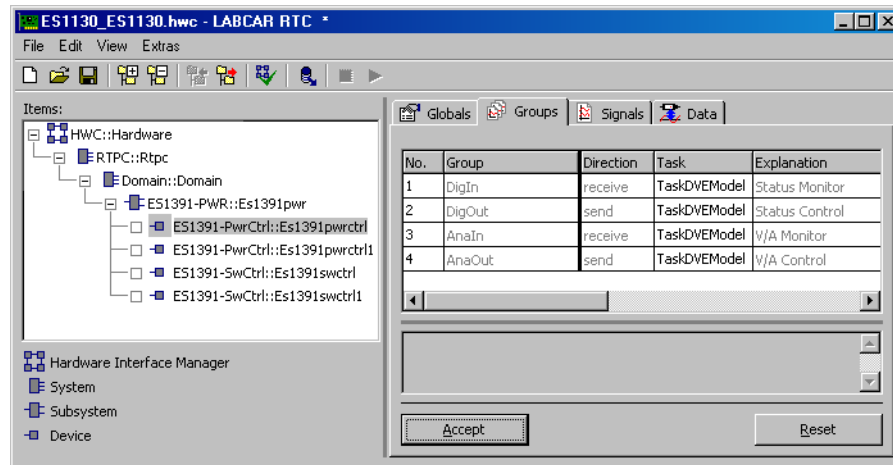
Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Power Control Device ID	uint32	Nein	Zuordnung einer Hardware-Einheit zur Netzteil-Fernsteuerung zum ES1391-PwrCtrl Device 0: Hardware-Einheit „PwrCtrl0“ 1: Hardware-Einheit „PwrCtrl1“ Ist nicht editierbar, wird automatisch eingestellt.
Operation Mode	uint8	Ja	Betriebsmodus 0: Analog & Digital IO Mode 1: Power Supply Control Mode „Power Supply Control Mode“ ist Voreinstellung
Power Supply Type	uint32	Ja	Typ des externen Netzteils 0: Takasago EX-375L 1: Takasago EX-750L 2: Takasago EX-1500L 255: Other Power Supply Types Nur editierbar im Betriebsmodus „Power Supply Control Mode“.
Maximum Programming Voltage [V]	real32	Ja	Maximale Programmierspannung des externen Netzteils in V. Zur Beschreibung kundenspezifischer externer Netzteile. Wertebereich: > 0 V
Maximum Output Voltage [V]	real32	Ja	Maximale Ausgangsspannung des externen Netzteils in V. Zur Beschreibung kundenspezifischer externer Netzteile. Wertebereich: > 0 V
Maximum Output Current [A]	real32	Ja	Maximaler Ausgangsstrom des externen Netzteils in A. Zur Beschreibung kundenspezifischer externer Netzteile. Wertebereich: > 0 A

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-3** ES1391-PwrCtrl Subsystem: Konfigurationsparameter der Registerkarte „Globals“

## 14.2.2 Groups (ES1391-PwrCtrl Device)

Das ES1391-PwrCtrl Device besitzt vier Signalgruppen.



**Abb. 14-5** Die Registerkarte „Groups“ des ES1391-PwrCtrl Device

- „DigIn“  
Diese Signalgruppe wird von der ES1391.1 zum Experimentaltarget übertragen. Sie transportiert die Statusinformationen der digitalen Ausgänge eines externen Netzteils oder allgemeiner die Statusinformationen der digitalen Eingänge der Hardware-Einheit zur Netzteil-Fernsteuerung.
- „DigOut“  
Diese Signalgruppe wird vom Experimentaltarget zur ES1391.1 übertragen. Sie transportiert Steuerinformationen für die digitalen Eingänge des externen Netzteils oder allgemeiner für die digitalen Ausgänge der Hardware-Einheit.
- „AnaIn“  
Diese Signalgruppe wird von der ES1391.1 zum Experimentaltarget übertragen. Sie transportiert die Werte für Ist-Strom und Ist-Spannung des externen Netzteils oder allgemeiner die Spannungswerte an den analogen Eingängen der Hardware-Einheit.
- „AnaOut“  
Diese Signalgruppe wird vom Experimentaltarget zur ES1391.1 übertragen. Sie transportiert Werte für Soll-Spannung und Soll-Strom des externen Netzteils oder allgemeiner die Spannungswerte für die analogen Ausgänge der Hardware-Einheit.

Den Signalgruppen sind Tasks des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise für alle Signalgruppen Tasks mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erfassenden und auszugebenden Signale sowie nach den Eigenschaften der ES1391.1-Firmware. Die ES1391.1-Firmware erfasst die analogen und digitalen Eingänge im 1-Millisekunden-Raster und aktualisiert ihre analogen und digitalen Ausgänge ebenfalls im 1-Millisekunden-Raster. Dieses Raster begrenzt sinnvolle Aktivierungsperioden der RTIO-Tasks auf eine Millisekunde.



### RTIO-Signale der Signalgruppe „DigIn“

Die Signalgruppe „DigIn“ umfasst fünf RTIO-Signale „DigIn\_0“ bis „DigIn\_2“ und „DigOutSts\_0“, „DigOutSts\_1“, die vom Datentyp „bool“ sind. Die Signale „DigIn\_x“ (x = 0, 1, 2) zeigen an, ob der digitale Eingang x aktiv (Wert ≠ 0) oder inaktiv (Wert = 0) ist. Die Signale „DigOutSts\_x“ (x = 0, 1) zeigen an, ob der digitale Ausgang x aktiv (Wert ≠ 0) oder inaktiv (Wert = 0) ist. Der aktive Pegel ist durch die Polarität (high-aktiv oder low-aktiv) des Ein- oder Ausgangs festgelegt. Die Polarität wird in der Registerkarte „Signals“ für die digitalen Ein- und Ausgänge angegeben.

Tab. 14-4 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
DigIn_x (x = 0, 1, 2)	bool	Statuserfassung der digitalen Ein- und Ausgänge
DigOutSts_x (x = 0, 1)		= 0: Ein-/Ausgang ist inaktiv ≠ 0: Ein-/Ausgang ist aktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 14-4** ES1391-PwrCtrl Device: Die RTIO-Signale der Signalgruppe „DigIn“  
Befindet sich das ES1391-PwrCtrl Device zur Fernsteuerung eines Netzteils im Betriebsmodus „Power Supply Control Mode“, so haben die einzelnen RTIO-Signale der Signalgruppe „DigIn“ eine vordefinierte Bedeutung, die in Tab. 14-5 angegeben ist.

RTIO-Signal	Bedeutung	Signalwerte
DigIn_0	Alarm-Eingang	= 0: Alarm inaktiv ≠ 0: Alarm aktiv
DigIn_1	CV-Modus CV = Constant Voltage	= 0: CV-Modus inaktiv ≠ 0: Netzteil ist im CV-Modus
DigIn_2	CC-Modus CC = Constant Current	= 0: CC-Modus inaktiv ≠ 0: Netzteil ist im CC-Modus
DigOutSts_0	Ausgang des Netzteils	= 0: Ausgang des Netzteils ist eingeschaltet ≠ 0: Ausgang des Netzteils ist abgeschaltet
DigOutSts_1	Spannungsversorgung des Netzteils	= 0: Das Netzteil ist an das Spannungsnetz angeschlossen. ≠ 0: Das Netzteil ist vom Spannungsnetz abgetrennt.

**Tab. 14-5** Vordefinierte Bedeutung der RTIO-Signale der Signalgruppe „DigIn“ (im Betriebsmodus „Power Supply Control Mode“ zur Fernsteuerung eines Netzteils)

### RTIO-Signale der Signalgruppe „DigOut“

Die Signalgruppe „DigOut“ umfasst zwei RTIO-Signale „DigOutCtrl\_0“ und „DigOutCtrl\_1“, die vom Datentyp „bool“ sind. Mit diesen Signalen werden die beiden digitalen Ausgänge einer Hardware-Einheit zur Netzteil-Steuerung akti-

viert (Wert  $\neq 0$ ) und deaktiviert (Wert = 0). Der aktive Pegel ist durch die Polarität (high-aktiv oder low-aktiv) der Ausgänge festgelegt. Die Polarität der Ausgänge wird in der Registerkarte „Signals“ angegeben.

Tab. 14-6 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
DigOutCtrl_x (x = 0, 1)	bool	Steuerung der digitalen Ausgänge = 0: Deaktivierung des Ausgangs $\neq 0$ : Aktivierung des Ausgangs
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 14-6** ES1391-PwrCtrl Device: Die RTIO-Signale der Signalgruppe „DigOut“

Befindet sich das ES1391-PwrCtrl Device zur Fernsteuerung eines Netzteils im Betriebsmodus „Power Supply Control Mode“, so haben die einzelnen RTIO-Signale der Signalgruppe „DigOut“ eine vordefinierte Bedeutung, die in Tab. 14-7 angegeben ist.

RTIO-Signal	Bedeutung	Signalwerte
DigOutCtrl_0	Ausgang des Netzteils	= 0: Ausgang des Netzteils abschalten $\neq 0$ : Ausgang des Netzteils einschalten
DigOutCtrl_1	Spannungsversorgung des Netzteils	= 0: Das Netzteil an das Spannungsnetz anschließen $\neq 0$ : Das Netzteil vom Spannungsnetz trennen

**Tab. 14-7** Vordefinierte Bedeutung der RTIO-Signale der Signalgruppe „DigOut“ (im Betriebsmodus „Power Supply Control Mode“ zur Fernsteuerung eines Netzteils)

#### Hinweis

Das ES1391-SwCtrl Device bietet die Möglichkeit, als Reaktion auf ein aktives Alarm-Signal einer angeschlossenen ES1392-Karte die (von der ES1391.1 gesteuerten) Netzteile abzuschalten. Abschalten bedeutet dabei, dass die Ausgänge der Netzteile deaktiviert werden.

Wird ein ES1391-SwCtrl Device entsprechend konfiguriert, kann es sein, dass das Steuersignal „DigOutCtrl\_0“ zum Ein- und Abschalten eines Netzteil-Ausgangs vom Statussignal „DigOutSts\_0“ abweicht.

#### RTIO-Signale der Signalgruppe „AnaIn“

Die Signalgruppe „AnaIn“ umfasst zwei RTIO-Signale „AnaIn\_0“ und „AnaIn\_1“, die vom Datentyp „real32“ sind.

Wird das ES1391-PwrCtrl Device im Modus zur digitalen und analogen Ein- und Ausgabe betrieben, so geben die RTIO-Signale die Spannungen an den beiden analogen Eingängen der Hardware-Einheit an. Der Wertebereich der RTIO-Signale ist dann 0.0 V bis 10.0 V.

Wird das ES1391-PwrCtrl Device im Modus zur Netzteil-Fernsteuerung betrieben, so gibt das RTIO-Signal „Analn\_0“ die aktuelle Ausgangsspannung des Netzteils in Volt und „Analn\_1“ den aktuellen Ausgangsstrom des Netzteils in Ampere an. Tab. 14-8 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Analn_0	real32	Analoger Eingang 0 Betriebsmodus „Analog & Digital IO Mode“: Eingangsspannung in Volt (0.0 V bis 10.0 V) Betriebsmodus „Power Supply Control Mode“: Aktuelle Ausgangsspannung des Netzteils in Volt
Analn_1	real32	Analoger Eingang 1 Betriebsmodus „Analog & Digital IO Mode“: Eingangsspannung in Volt (0.0 V bis 10.0 V) Betriebsmodus „Power Supply Control Mode“: Aktueller Ausgangsstrom des Netzteils in Ampere

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-8** ES1391-PwrCtrl Device: Die RTIO-Signale der Signalgruppe „Analn“  
*RTIO-Signale der Signalgruppe „AnaOut“*

Die Signalgruppe „AnaOut“ umfasst zwei RTIO-Signale „AnaOut\_0“ und „AnaOut\_1“, die vom Datentyp „real32“ sind.

Wird das ES1391-PwrCtrl Device im Modus zur digitalen und analogen Ein- und Ausgabe betrieben, so definieren die RTIO-Signale die Spannungen an den beiden analogen Ausgängen der Hardware-Einheit. Der Wertebereich der RTIO-Signale ist dann 0.0 V bis 10.0 V.

Wird das ES1391-PwrCtrl Device im Modus zur Netzteil-Fernsteuerung betrieben, so definiert das RTIO-Signal „AnaOut\_0“ die Ausgangsspannung<sup>1</sup> des Netzteils in Volt und „AnaOut\_1“ den Ausgangsstrom<sup>2</sup> des Netzteils in Ampere.

<sup>1</sup>. Im CV-Modus (CV = Constant Voltage) des Netzteils

<sup>2</sup>. Im CC-Modus (CC = Constant Current) des Netzteils

Tab. 14-9 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
AnaOut_0	real32	Analoger Ausgang 0 Betriebsmodus „Analog & Digital IO Mode“: Ausgangsspannung in Volt (0.0 V bis 10.0 V) Betriebsmodus „Power Supply Control Mode“: Soll-Ausgangsspannung des Netzteils in Volt
AnaOut_1	real32	Analoger Eingang 1 Betriebsmodus „Analog & Digital IO Mode“: Ausgangsspannung in Volt (0.0 V bis 10.0 V) Betriebsmodus „Power Supply Control Mode“: Soll-Ausgangsstrom des Netzteils in Ampere

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-9** ES1391-PwrCtrl Device: Die RTIO-Signale der Signalgruppe „Ana-Out“

#### 14.2.3 Signals (ES1391-PwrCtrl Device)

In der Registerkarte „Signals“ wird die Konfiguration der digitalen und analogen Ein-/Ausgänge einer Hardware-Einheit zur Netzteil-Fernsteuerung durchgeführt. Abb. 14-6 zeigt die RTIO-Parameter der Registerkarte „Signals“.

Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

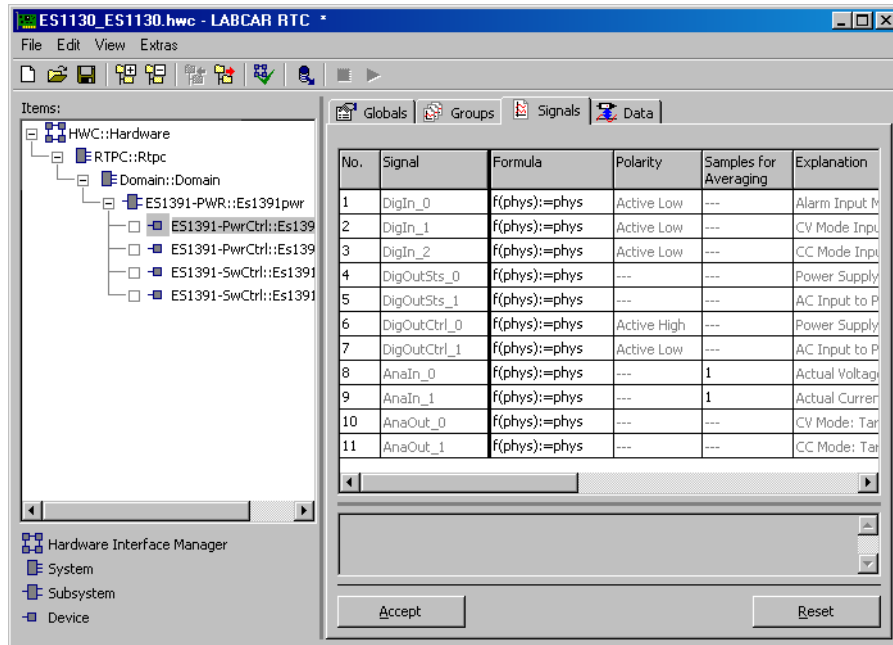
##### *Polarity*

In diesem Listenfeld wird die Polarität (high-aktiv oder low-aktiv) eines digitalen Ein- oder Ausgangs angegeben. Wird das ES1391-PwrCtrl Device zur Fernsteuerung eines der vordefinierten Takasago-Netzteile verwendet, so werden die Polaritätswerte für die einzelnen digitalen Ein- und Ausgänge automatisch eingestellt - sie sind in diesem Fall durch den Anwender nicht editierbar. In allen anderen Fällen sind sie editierbar.

##### *Samples for Averaging*

Der Spannungswert eines analogen Eingangs kann als Mittelwert mehrerer Abtastwerte ermittelt werden. In diesem Eingabefeld wird (für die analogen Eingänge) die Anzahl der Abtastwerte angegeben, aus denen dieser Mittelwert berechnet wird. Die Maximalzahl der Abtastwerte zur Bildung des Mittelwerts beträgt 100.

Die ES1391.1-Firmware tastet die analogen Eingänge im 1-Millisekunden-Raster ab. Im gleichen Raster wird auch der Mittelwert aktualisiert. Bei einer Berechnung des Mittelwerts über n Abtastwerte werden also jeweils die Abtastwerte der letzten n Millisekunden herangezogen.



**Abb. 14-6** Die Registerkarte „Signals“ des „ES1391-PwrCtrl“ Device (bei Fernsteuerung eines Takasago-Netzteils)

Tab. 14-10 fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Polarity	bool	Ja	Polarität des digitalen Ein- oder Ausgangs = 0: Low-aktiv ≠ 0: High-aktiv
Samples for Averaging	uint32	Ja	Analoger Eingang: Anzahl der Abtastwerte zur Berechnung des Mittelwerts. Wertebereich: 1 ... 100

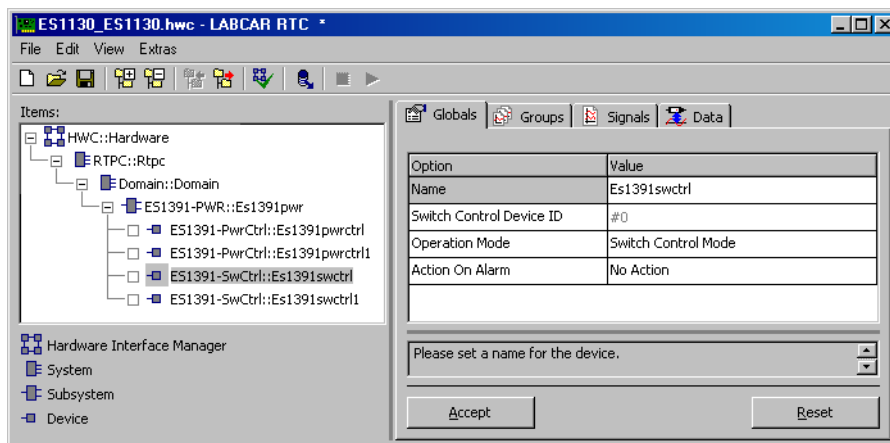
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-10** ES1391-PwrCtrl Device: Konfigurationsparameter der Registerkarte „Signals“

## 14.3 ES1391-SwCtrl Device

### 14.3.1 Globals (ES1391-SwCtrl Device)

Das ES1391-SwCtrl Device dient zur Steuerung eines ES1392 High Current Switch Board - es kann jedoch auch für Aufgaben der digitalen Ein- und Ausgabe konfiguriert werden. Die Registerkarte „Globals“ eines ES1391-SwCtrl Device ist in Abb. 14-7 dargestellt.



**Abb. 14-7** Die Registerkarte „Globals“ des ES1391-SwCtrl Device  
*Switch Control Device ID*

Dieses Listenfeld ordnet das ES1391-SwCtrl Device einem der beiden auf der ES1391.1 vorhandenen Steuereinheiten zu.

Das Listenfeld ist nicht editierbar, sein Wert wird automatisch eingetragen. Der Optionswert „#0“ bezeichnet Hardware-Einheit „SwCtrl0“ und der Optionswert „#1“ bezeichnet Hardware-Einheit „SwCtrl1“.

#### *Operation Mode*

Dieses Listenfeld legt fest, in welchem Modus das ES1391-SwCtrl Device und die zugehörige Hardware-Einheit betrieben werden. Es stehen zwei Modi zur Auswahl:

- Die Einstellung „Switch Control Mode“ konfiguriert das ES1391-SwCtrl Device für die Steuerung von ES1392 High Current Switch Boards.
- Die Einstellung „Digital IO Mode“ konfiguriert das ES1391-SwCtrl Device für die digitale Ein- und Ausgabe.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Switch Control Mode“.

#### *Action On Alarm*

In diesem Listenfeld wird festgelegt, welche Aktionen in Reaktion auf ein aktives Alarmsignal des an die Hardware-Einheit angeschlossenen ES1392 High Current Switch Board durchgeführt werden. Es ist nur dann editierbar bzw. hat nur dann eine Bedeutung, wenn das ES1391-SwCtrl Device im Betriebsmodus „Switch Control Mode“ zur ES1392-Steuerung verwendet wird. Als Reaktion auf ein akti-

ves Alarmsignal des ES1392 High Current Switch Board können eine oder beide der an die Hardware-Einheiten zur Netzteil-Fernsteuerung angeschlossenen Netzteile abgeschaltet werden.

Das Listenfeld ist zur Laufzeit editierbar.

In Tab. 14-11 sind die Eigenschaften der einzelnen RTIO-Parameter der Registerkarte „Globals“ zusammen gefasst.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Switch Control Device ID	uint32	Nein	Zuordnung einer Hardware-Einheit für die die Steuerung einer ES1392 zum ES1391-SwCtrl Device 0: Hardware-Einheit „SwCtrl0“ 1: Hardware-Einheit „SwCtrl1“ Ist nicht editierbar, wird automatisch eingestellt.
Operation Mode	uint8	Ja	Betriebsmodus 0: Digital IO Mode 1: Switch Control Mode „Switch Control Mode“ ist Voreinstellung.
Action On Alarm	uint8	Ja	Reaktion auf aktives ES1392-Alarm-Signal 0: Keine Reaktion 1: Abschaltung von Netzteil an Port „PwrCtrl0“ 2: Abschaltung von Netzteil an Port „PwrCtrl1“ 3: Abschaltung beider Netzteile Nur editierbar im Betriebsmodus „Switch Control Mode“.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-11** ES1391-SwCtrl Device: Konfigurationsparameter der Registerkarte „Globals“

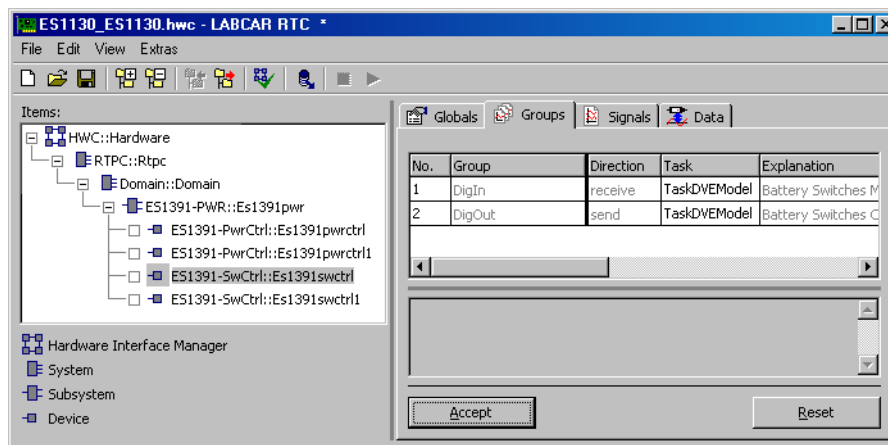
### 14.3.2 Groups (ES1391-SwCtrl Device)

Das ES1391-SwCtrl Device besitzt zwei Signalgruppen (siehe Abb. 14-8).

Die Signalgruppe „DigIn“ wird von der ES1391.1 zum Experimentaltarget übertragen. Sie transportiert die Statusinformationen der digitalen Ausgänge eines ES1392 High Current Switch Board oder allgemeiner die Statusinformationen der digitalen Eingänge der zugeordneten Hardware-Einheit.

Die Signalgruppe „DigOut“ wird vom Experimentaltarget zur ES1391.1 übertragen. Sie transportiert Steuerinformationen für die digitalen Eingänge eines ES1392 High Current Switch Board oder allgemeiner für die digitalen Ausgänge der zugeordneten Hardware-Einheit.

Den Signalgruppen sind Tasks des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise für alle Signalgruppen Tasks mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erfassenden und auszugebenden Signale sowie nach den Eigenschaften der ES1391.1. Die ES1391.1 erfasst die digitalen Eingänge im 1-Millisekunden-Raster und aktualisiert ihre digitalen Ausgänge ebenfalls im 1-Millisekunden-Raster. Dieses Raster begrenzt sinnvolle Aktivierungsperioden der RTIO-Tasks auf eine Millisekunde.



**Abb. 14-8** Die Registerkarte „Groups“ des ES1391-SwCtrl Device  
RTIO-Signale der Signalgruppe „DigIn“

Die Signalgruppe „DigIn“ umfasst acht RTIO-Signale „DigIn\_0“, „DigIn\_1“ und „DigOutSts\_0“ bis „DigOutSts\_5“, die vom Datentyp „bool“ sind.

Die Signale „DigIn\_x“ (x = 0, 1) zeigen an, ob der digitale Eingang x aktiv (Wert ≠ 0) oder inaktiv (Wert = 0) ist. Die Signale „DigOutSts\_x“ (x = 0, 1, ... 5) zeigen an, ob der digitale Ausgang x aktiv (Wert ≠ 0) oder inaktiv (Wert = 0) ist. Der aktive Pegel ist durch die Polarität (high-aktiv oder low-aktiv) des Ein- oder Ausganges festgelegt. Die Polarität wird in der Registerkarte „Signals“ angegeben. Tab. 14-12 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
DigIn_x (x = 0, 1)	bool	Stuserfassung der digitalen Ein- und Ausgänge
DigOutSts_x (x = 0, 1, ... 5)		= 0: Ein- / Ausgang ist inaktiv ≠ 0: Ein- / Ausgang ist aktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 14-12** ES1391-SwCtrl Device: Die RTIO-Signale der Signalgruppe „DigIn“



Ist das ES1391-SwCtrl Device im Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board, so haben die einzelnen RTIO-Signale der Signalgruppe „DigIn“ eine vordefinierte Bedeutung, die in Tab. 14-13 angegeben ist.

RTIO-Signal	Bedeutung	Signalwerte
DigIn_0	Hauptrelais-Eingang	= 0: Hauptrelais offen ≠ 0: Hauptrelais geschlossen
DigIn_1	Alarm-Eingang	= 0: Alarm inaktiv ≠ 0: Alarm aktiv
DigOutSts_0 bis DigOutSts_4	Hoch-Strom-Schalter 0 bis Hoch-Strom-Schalter 4	0: Schalter offen ≠ 0: Schalter geschlossen
DigOutSts_5	Konfiguration von Hoch-Strom-Schalter 0	= 0: Schalter 0 schaltet nach Batteriespannung ≠ 0: Schalter 0 schaltet nach Batteriemasse

**Tab. 14-13** Vordefinierte Bedeutung der RTIO-Signale der Signalgruppe „DigIn“ (im Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board)

#### *RTIO-Signale der Signalgruppe „DigOut“*

Die Signalgruppe „DigOut“ umfasst sechs RTIO-Signale „DigOutCtrl\_0“ bis „DigOutCtrl\_5“, die vom Datentyp „bool“ sind. Mit diesen Signalen werden die digitalen Ausgänge einer Hardware-Einheit aktiviert (Wert ≠ 0) und deaktiviert (Wert = 0). Der aktive Pegel ist durch die Polarität (high-aktiv oder low-aktiv) der Ausgänge festgelegt. Die Polarität der Ausgänge wird in der Registerkarte „Signals“ angegeben.

Tab. 14-14 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
DigOutCtrl_x (x=0, 1, ... 5)	bool	Steuerung der digitalen Ausgänge = 0: Deaktivierung des Ausgangs ≠ 0: Aktivierung des Ausgangs
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 14-14** ES1391-SwCtrl Device: Die RTIO-Signale der Signalgruppe „DigOut“

Ist das ES1391-SwCtrl Device im Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board, so haben die einzelnen RTIO-Signale der Signalgruppe „DigOut“ eine vordefinierte Bedeutung, die in Tab. 14-15 angegeben ist.

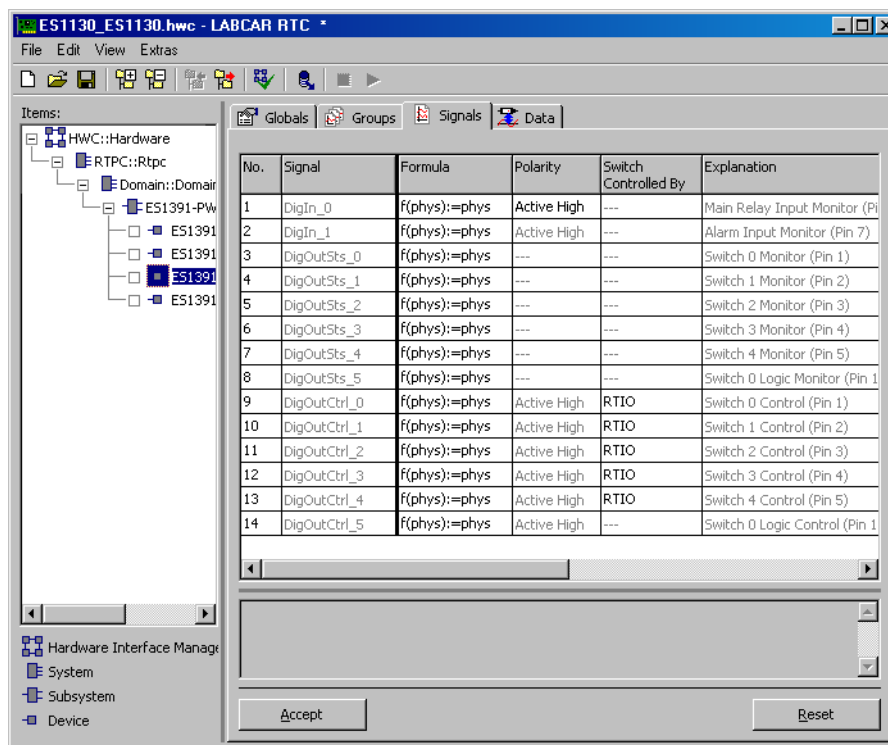
RTIO-Signal	Bedeutung	Signalwerte
DigOutCtrl_0 bis DigOutCtrl_4	Hoch-Strom-Schalter 0 bis Hoch-Strom-Schalter 4	= 0: Schalter öffnen ≠ 0: Schalter schließen
DigOutCtrl_5	Konfiguration von Hoch-Strom-Schalter 0	= 0: Schalter 0 schaltet nach Batteriespannung ≠ 0: Schalter 0 schaltet nach Batteriemasse

**Tab. 14-15** Vordefinierte Bedeutung der RTIO-Signale der Signalgruppe „DigOut“ (m Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board)

### 14.3.3 Signals (ES1391-SwCtrl Device)

In der Registerkarte „Signals“ erfolgt die Konfiguration der digitalen Ein- und Ausgänge einer Hardwareeinheit. Alle Parameter sind online (d.h. während der Laufzeit des RTIO-Treibers auf dem Experimentaltarget) editierbar.

Abb. 14-9 zeigt die RTIO-Parameter der Registerkarte „Signals“.



**Abb. 14-9** Die Registerkarte „Globals“ des ES1391-SwCtrl Device (im Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board)

### Polarity

In diesem Listenfeld wird die Polarität (high-aktiv oder low-aktiv) eines digitalen Ein- oder Ausgangs angegeben.

Wird das ES1391-SwCtrl Device im Betriebsmodus „Switch Control Mode“ zur Steuerung eines ES1392 High Current Switch Board verwendet, so werden die Polaritätswerte für die einzelnen digitalen Ein- und Ausgänge automatisch eingestellt - sie sind in diesem Fall durch den Anwender nicht editierbar. Ausnahme ist der digitale Eingang 0 (der Hauptrelais-Eingang), dessen Polarität editierbar ist.

Wird das ES1391-SwCtrl Device im Betriebsmodus „Digital IO Mode“ zur digitalen Ein-/Ausgabe verwendet, so sind die Polaritätsfelder aller Ein- und Ausgänge editierbar.

### Switch Controlled By

Dieses Listenfeld ist nur relevant und editierbar, wenn das ES1391-SwCtrl Device zur Steuerung eines ES1392 High Current Switch Board (d.h. im Betriebsmodus „Switch Control Mode“) verwendet wird.

Es legt fest, ob der zugeordnete Hoch-Strom-Schalter des ES1392 High Current Switch Board über RTIO (Einstellung „RTIO“) oder den Zustand des Hauptrelais (Einstellung „Main Relay“) ein- und ausgeschaltet wird. Wird ein Hochstromschalter vom Zustand des Hauptrelais gesteuert, so wird er geschlossen, falls das Hauptrelais geschlossen ist und geöffnet, falls das Hauptrelais offen ist.

Wird das ES1391-SwCtrl Device im Betriebsmodus „Digital IO Mode“ zur digitalen Ein-/Ausgabe verwendet, so wird dieses Listenfeld für alle digitalen Ausgänge automatisch auf die Einstellung „RTIO“ gesetzt und ist nicht editierbar.

Tab. 14-16 fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Polarity	bool	Ja	Polarität des digitalen Ein- oder Ausgangs = 0: Low-aktiv ≠ 0: High-aktiv
Switch Controlled By	bool	Ja	Steuerquelle für digitalen Ausgang = 0: RTIO ≠ 0: Hauptrelais

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 14-16** ES1391-SwCtrl Device: Konfigurationsparameter der Registerkarte „Signals“



## 15 ES1650.1 Piggyback Carrier Board

Das ES1650.1 Piggyback Carrier Board wird in VMEbus-Systemen als Trägerkarte für Aufsteckmodule eingesetzt. Auf der ES1650.1 sind zwei Steckplätze für Aufsteckmodule vorhanden. Es stehen Aufsteckmodule für unterschiedliche Aufgaben zur Verfügung, z.B. digitale und analoge Ein- und Ausgabemodule sowie Relaismodule mit Schaltfunktionen.

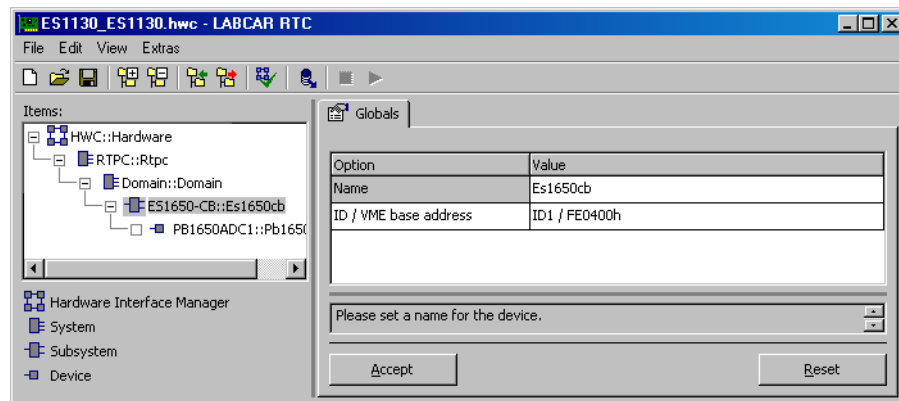
Dieses Dokument beschreibt die RTIO-Einbindung der ES1650.1 sowie die RTIO-Einbindungen der auf diese Trägerkarte aufsteckbaren Module:

- PB1650ADC1 (auf Seite 378)
- PB1650DAC1 (auf Seite 381)
- PB1650DIO1 (auf Seite 384)
- PB1650DIO2 (auf Seite 386)
- PB1650REL1 (auf Seite 388)

### 15.1 ES1650-CB Subsystem

#### 15.1.1 Globals (ES1650-CB Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES1650-CB Subsystems beschrieben.



**Abb. 15-1** Die Registerkarte „Globals“ des ES1650-CB Subsystems

*ID / VME base address*

Im Auswahlfeld „ID / VME base address“ wird die VMEbus-Basisadresse der verwendeten ES1650.1 angegeben. Es stehen 16 unterschiedliche Basisadressen zur Auswahl (ID1/FE0400h, ID2/FE0C00h, ID3/FE1400h, ID4/FE1C00h, ID5/FE2400h, ID6/FE2C00h, ID7/FE3400h, ID8/FE3C00h, ID9/FE4400h, ID10/FE4C00h, ID11/FE5400h, ID12/FE5C00h, ID13/FE6400h, ID14/FE6C00h, ID15/FE7400h und ID16/FE7C00h). Die Größe des Adressbereichs der ES1650.1 kann mit Steckbrücken auf 256 Byte oder 8 kByte konfiguriert werden. Sämtliche in RTIO eingebundenen Aufsteckmodule weisen einen Adressbereich von 256 Bytes auf. Die RTIO-Einbindung der ES1650.1 unterstützt keine Aufsteckmodule mit 8 kByte Adressraum.

## 15.2 PB1650ADC1

Das PB1650ADC1 Aufsteckmodul dient der analogen Datenerfassung in Verbindung mit der ES1650.1. Das Modul ist für mittlere Auflösungen und kleine bis mittlere Datenraten konzipiert. Es umfasst unter anderen folgende Funktionen:

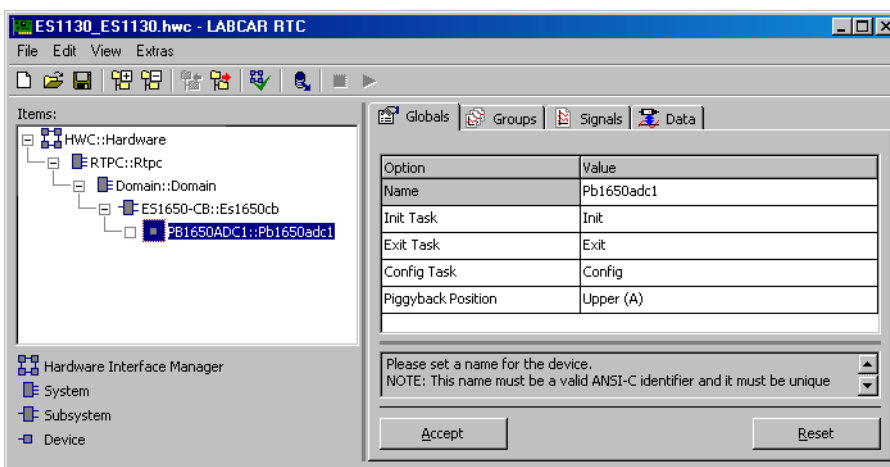
- Analog-Digital-Umsetzung mit 12 Bit Auflösung
- Acht unipolare oder bipolare Eingangskanäle, per Software programmierbar
- Vier Eingangsspannungsbereiche

Das PB1650ADC1 Aufsteckmodul kann in VMEbus-Systemen überall da eingesetzt werden, wo analoge Eingangssignale erfasst werden sollen. Beispielhafte Einsatzgebiete sind:

- Erfassung analoger Gebersignale, beispielsweise Motortemperatur, Öltemperatur, Fahrpedalstellung.
- Erfassung analoger Ausgabegrößen des Steuergeräts.
- Nachbildung von analogen Sensoren, die eine Steuer- oder Referenzspannung vom Steuergerät erhalten.

### 15.2.1 Globals (PB1650ADC1 Device)

In diesem Abschnitt werden die globalen Einstellungen des PB1650ADC1 Devices beschrieben.



**Abb. 15-2** Die Registerkarte „Globals“ des PB1650ADC1 Devices

#### *Piggyback Position*

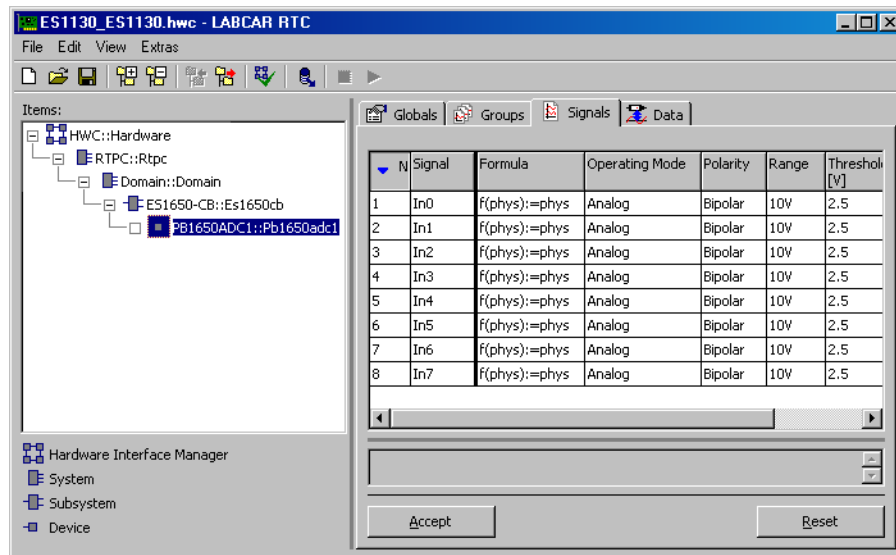
Im Auswahlfeld „Piggyback Position“ wird angegeben, ob das Aufsteckmodul auf dem oberen oder unteren Steckplatz der ES1650.1 steckt.

### 15.2.2 Groups (PB1650ADC1 Device)

Die im Register „Groups“ verfügbaren Optionen sind im Abschnitt „Standardoptionen im Register „Groups““ auf Seite 47 beschrieben.

### 15.2.3 Signals (PB1650ADC1 Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PB1650ADC1 Devices beschrieben.



**Abb. 15-3** Die Registerkarte „Signals“ des PB1650ADC1 Devices

#### *Operating Mode*

Jeder der 8 Hardwarekanäle eines PB1650ADC1 Aufsteckmoduls kann in vier unterschiedlichen Modi betrieben werden, die im Auswahlfeld „Operating Mode“ für jeden Kanal getrennt eingestellt werden (siehe Abb. 15-3).

- Disabled: Der Kanal ist deaktiviert.
- Analog: Der Kanal ist zur analogen Messwerterfassung konfiguriert.  
Das zugehörige Signal  $In_x$  ( $x = 0, 1, \dots, 7$ ) gibt die gemessene Spannung in Volt an.
- Comparator: Der Kanal ist als Komparator konfiguriert.  
Der zugehörige Signalwert  $In_x$  ist gleich 0, wenn die Eingangsspannung des Kanals kleiner als eine vom Anwender einstellbare Spannungsschwelle ist, anderenfalls ist der Signalwert gleich 1.
- Comparator (abs): Der Kanal ist als absoluter Komparator konfiguriert.  
Der zugehörige Signalwert  $In_x$  ist gleich 0, wenn der Betrag der Eingangsspannung des Kanals kleiner als eine vom Anwender einstellbare Schwellenspannung ist, anderenfalls ist der Signalwert gleich 1.

#### *Polarity/Range*

Im Auswahlfeld „Polarity“ wird die Spannungspolarität eines Messkanals eingestellt und im Auswahlfeld „Range“ wird der Spannungsbereich angegeben.

Im Gegensatz zur Polarität ist der Spannungsbereich eines Kanals nicht Softwarekonfigurierbar, sondern wird durch Lötbrücken eingestellt. Mit den beiden Auswahlfeldern „Polarity“ und „Range“ lassen sich für jeden Kanal insgesamt 4 Messbereiche einstellen (0 V ... 5.0 V; -5.0 V ... 5.0 V; 0 V ... 10.0 V; -10.0 V ... 10.0 V).

Die Einstellungen in den Feldern „Polarity“ und „Range“ sind für alle Betriebsmodi bis auf „Disabled“ relevant.

#### *Threshold [V]*

---

Die Einstellung im Feld „Threshold [V]“ ist nur für die beiden Komparator-Modi von Bedeutung. Hier wird die Schwellspannung für den Komparator festgelegt.

#### *AD Average Samples*

---

Hier wird festgelegt, aus wieviel Abtastwerten ein analoger Spannungswert durch Mittelwertbildung berechnet wird. Durch Mittelwertbildung kann eine Stör- oder Rauschunterdrückung erzielt werden, es ist jedoch zu beachten, dass für eine Abtastung pro Kanal etwa 43  $\mu$ s benötigt werden.

Die Einstellung im Feld „AD Average Samples“ ist für alle Betriebsmodi bis auf „Disabled“ von Bedeutung.

### 15.2.4 Datentypen und Wertebereiche der Signale

---

Die 8 Signale  $In_x$  ( $x = 0, 1, 2, \dots, 7$ ) des PB1650ADC1 Aufsteckmoduls sind vom Datentyp „float“ bzw. „real32“. Ein Kanal im Modus „Disabled“ liefert den Wert 0.0. Ein Kanal im Modus „Analog“ liefert die gemessene Spannung in Volt. Ein Kanal im Modus „Comparator“ oder „Comparator (abs)“ liefert entweder 0.0 oder 1.0 (vgl. Abschnitt 15.2.3 auf Seite 379).

#### **Hinweis**

*Wenn in diesem Dokument Datentypen angegeben werden, so sind dies die Datentypen, die der RTIO-Treiber intern verwendet.*



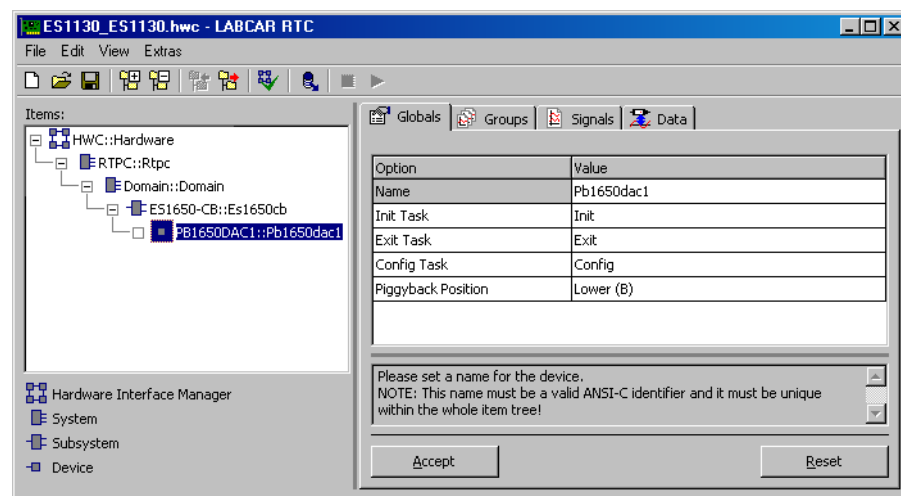
## 15.3 PB1650DAC1

Das PB1650DAC1 Aufsteckmodul dient der Erzeugung von analogen Ausgangssignalen in Verbindung mit der ES1650.1. Das Modul umfasst unter anderen folgende Funktionen:

- Digital-Analog-Umsetzung mit 12 Bit Auflösung
- Vier wahlweise uni- oder bipolare Ausgangskanäle (Ausgangsspannungsbereich 0.0 V... 10.0 V oder -10.0 V... 10.0 V)
- Analoge Ausgangskanäle galvanisch vom VMEbus-System getrennt

### 15.3.1 Globals (PB1650DAC1 Device)

In diesem Abschnitt werden die globalen Einstellungen des PB1650DAC1 Devices beschrieben.



**Abb. 15-4** Die Registerkarte „Globals“ des PB1650DAC1 Devices  
*Piggyback Position*

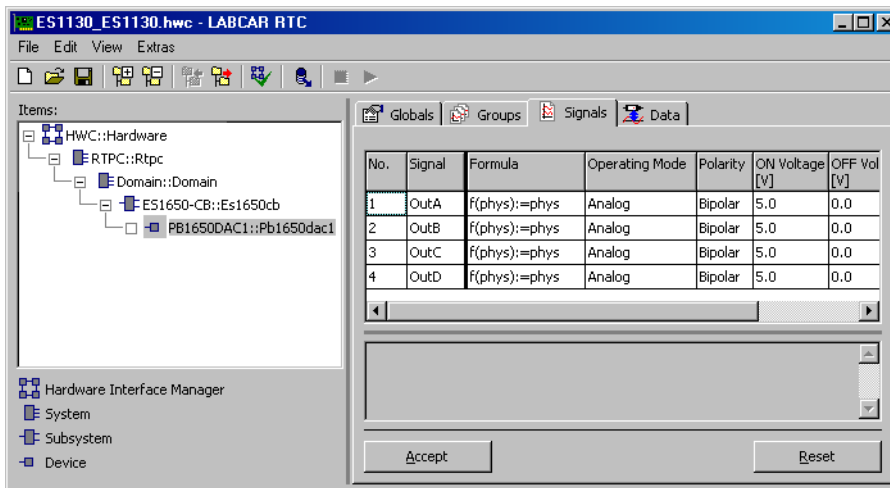
Im Auswahlfeld „Piggyback Position“ wird angegeben, ob das Aufsteckmodul auf dem oberen oder unteren Steckplatz der ES1650.1 steckt.

### 15.3.2 Groups (PB1650DAC1 Device)

Die im Register „Groups“ verfügbaren Optionen sind im Abschnitt „Standardoptionen im Register „Groups““ auf Seite 47 beschrieben.

### 15.3.3 Signals (PB1650DAC1 Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PB1650DAC1 Devices beschrieben.



**Abb. 15-5** Die Registerkarte „Signals“ des PB1650DAC1 Devices

#### *Operating Mode*

Jeder der 4 Hardwarekanäle eines PB1650DAC1 Aufsteckmoduls kann in vier unterschiedlichen Modi betrieben werden, die im Auswahlfeld „Operating Mode“ für jeden Kanal getrennt eingestellt werden (siehe Abb. 15-5).

- Disabled: Der Kanal ist deaktiviert.
- Analog: Der Kanal ist zur analogen Spannungsausgabe konfiguriert.  
Das zugehörige Signal *Outx* ( $x = A, B, C, D$ ) gibt die auszugebende Spannung in Volt an.
- Comparator: Der Kanal ist als Komparator konfiguriert.  
Ist der Signalwert *Outx* kleiner als ein vom Anwender einstellbarer Schwellwert „Comparator Threshold“, so wird am Ausgang die im Feld „OFF Voltage [V]“ eingestellte Spannung ausgegeben, anderenfalls wird die im Feld „ON Voltage [V]“ eingestellte Spannung ausgegeben.
- Comparator (abs): Der Kanal ist als absoluter Komparator konfiguriert.  
Ist der Signalwert „Outx“ betragsmäßig kleiner als ein vom Anwender einstellbarer Schwellwert „Comparator Threshold“, so wird am Ausgang die im Feld „OFF Voltage [V]“ eingestellte Spannung ausgegeben, anderenfalls wird die im Feld „ON Voltage [V]“ eingestellte Spannung ausgegeben.

#### *Polarity*

Im Auswahlfeld „Polarity“ wird die Spannungspolarität eines Messkanals angegeben. Es ist zu beachten, dass diese Spannungspolarität nicht Software-konfigurierbar ist, sondern durch Lötbrücken eingestellt wird.

Der Ausgangsspannungsbereich eines Messkanals beträgt (abhängig von der Einstellung im Feld „Polarity“) entweder 0.0 V...10.0 V oder -10.0 V...10.0 V.

#### ON Voltage [V], OFF Voltage [V] und Comparator Threshold

Die Felder „Comparator Threshold“, „OFF Voltage [V]“ und „ON Voltage [V]“ sind nur in den beiden Komparator-Modi relevant. Die zulässigen Werte für „OFF Voltage [V]“ und „ON Voltage [V]“ richten sich nach dem gewählten Spannungsbereich.

Der zulässige Wertebereich für „Comparator Threshold“ ist im Kanalmodus Comparator -1.0 ... 1.0 und im Kanalmodus Comparator (abs) 0.0 ... 1.0.

#### 15.3.4 Datentypen und Wertebereiche der Signale

Die 4 Signale *Outx* ( $x = A, B, C, D$ ) des PB1650DAC1 Aufsteckmoduls sind vom Datentyp `float` bzw. `real32`. Der Wert *Outx* eines Kanals im Modus „Disabled“ wird ignoriert. Der Wert *Outx* eines Kanals im Modus „Analog“ gibt die auszugebende Spannung in Volt an. Liegt der Wert *Outx* außerhalb des zulässigen Bereichs, so wird er auf den nächstliegenden zulässigen Wert begrenzt. Der Wert *Outx* eines Kanals im Modus „Comparator“ oder „Comparator (abs)“ muss zwischen -1.0 und 1.0 liegen. Ein Wert *Outx* außerhalb des zulässigen Bereichs wird auf den nächstliegenden zulässigen Wert begrenzt.

## 15.4 PB1650DIO1

Das PB1650DIO1 Aufsteckmodul dient der digitalen Ein- und Ausgabe von Schaltzuständen über jeweils acht galvanisch getrennte Ein- und Ausgänge. Das Modul bietet dabei unter anderen folgende Funktionen:

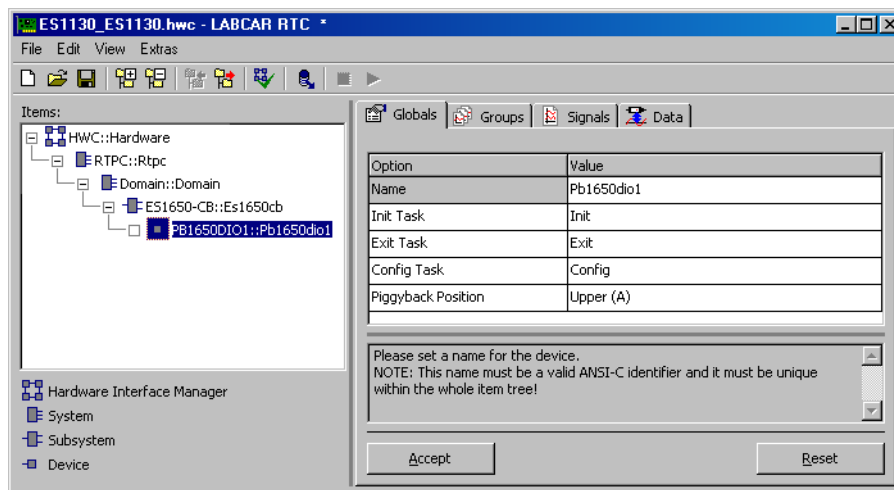
- Acht parallele, galvanisch getrennte digitale Eingangskanäle (bis zu 80 V Gleichspannung)
- Acht parallele, galvanisch getrennte digitale Ausgangskanäle (bis zu 500 mA)

Das PB1650DIO1 Aufsteckmodul wird in Verbindung mit der ES1650.1 zur Erfassung und Generierung binärer Schaltsignale eingesetzt. Beispielhafte Einsatzgebiete sind:

- Erfassung von Schaltausgängen des Steuergerätes (Ansteuerung von Magnetventilen, Rückfahrlichtrelais)
- Simulation eines Schalters (Handbremsschalter)

### 15.4.1 Globals (PB1650DIO1 Device)

In diesem Abschnitt werden die globalen Einstellungen des PB1650DIO1 Devices beschrieben.



**Abb. 15-6** Die Registerkarte „Globals“ des PB1650DIO1 Devices

#### *Piggyback Position*

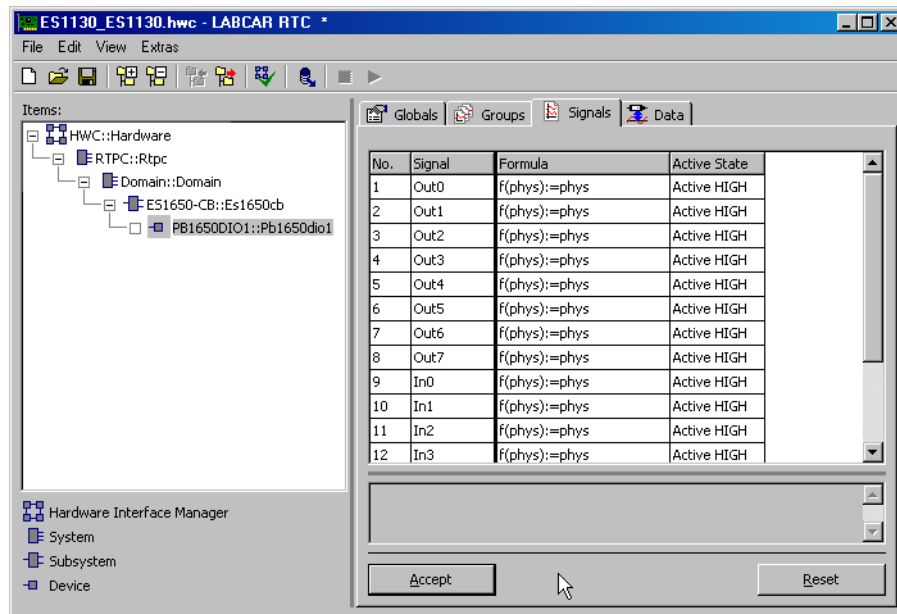
Im Auswahlfeld „Piggyback Position“ wird angegeben, ob das Aufsteckmodul auf dem oberen oder unteren Steckplatz der ES1650.1 steckt.

### 15.4.2 Groups (PB1650DIO1 Device)

Die im Register „Groups“ verfügbaren Optionen sind im Abschnitt „Standardoptionen im Register „Groups““ auf Seite 47 beschrieben.

### 15.4.3 Signals (PB1650DIO1 Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PB1650DIO1 Devices beschrieben.



**Abb. 15-7** Die Registerkarte „Signals“ des PB1650DIO1 Device

#### Active State

Für jeden Ein- bzw. Ausgang eines PB1650DIO1 Aufsteckmoduls kann im Auswahlfeld „Active State“ angegeben werden, ob das zugehörige Signal low aktiv (Active LOW) oder high aktiv (Active HIGH) ist.

Bei einem Eingang  $In_x$  ( $x = 0, 1 \dots 7$ ) bedeutet der Wert 0, dass das angelegte Signal inaktiv ist, der Wert 1 bedeutet, dass das angelegte Signal aktiv ist.

Bei einem Ausgang  $Out_x$  ( $x = 0, 1, \dots 7$ ) bedeutet der Wert 0, dass das zugehörige Signal inaktiv geschaltet wird, der Wert 1 bedeutet, dass das zugehörige Signal aktiv geschaltet wird.

### 15.4.4 Datentypen und Wertebereiche der Signale

Die 8 Signale  $Out_x$  ( $x = 0, 1, \dots 7$ ) des PB1650DIO1 Aufsteckmoduls sind vom Datentyp `uint8`. Das Signal wird vom RTIO-Treiber als Boolesche Größe interpretiert. Der Wert 0 entspricht `FALSE`, ein Wert ungleich 0 entspricht `TRUE`.

Die 8 Signale  $In_x$  ( $x = 0, 1, \dots 7$ ) sind vom Datentyp `uint8`. Der zurückgelieferte Signalwert ist entweder 0 oder 1.

## 15.5 PB1650DIO2

Das PB1650DIO2 Aufsteckmodul dient der digitalen Ein- und Ausgabe von Schaltzuständen über jeweils acht galvanisch getrennte Ein- und Ausgänge. Das Modul bietet dabei unter anderen folgende Funktionen:

- Acht parallele, galvanisch getrennte, digitale Eingangskanäle
- Acht parallele, galvanisch getrennte, digitale Ausgangskanäle
- Vier parallele, galvanisch getrennte Steuerleitungen

Das PB1650DIO2 Aufsteckmodul wird in Verbindung mit der ES1650.1 zur Erfassung und Generierung binärer Schaltsignale mit TTL-Signalpegeln eingesetzt. Beispielhafte Einsatzgebiete sind:

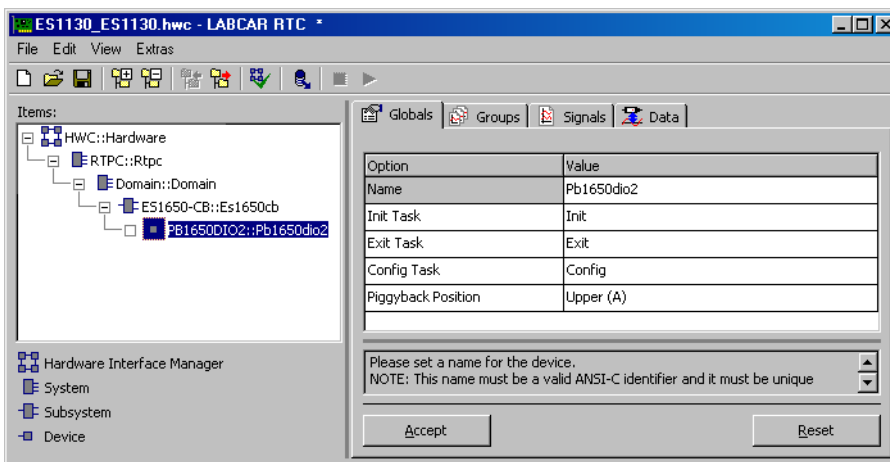
- Erfassung von Schaltausgängen des Steuergerätes (Ansteuerung von Magnetventilen, Rückfahrlichtrelais)
- Simulation eines Schalters (Handbremsschalter)

### Hinweis

*Der RTIO-Treiber des PB1650DIO2 Aufsteckmoduls verwendet zwei der vier zur Verfügung stehenden Steuerleitungen als zusätzliche Eingänge und die restlichen zwei Steuerleitungen als zusätzliche Ausgänge, so dass insgesamt 10 Eingänge und 10 Ausgänge zur Verfügung stehen.*

### 15.5.1 Globals (PB1650DIO2 Device)

Dieser Abschnitt erklärt die globalen Einstellungen des PB1650DIO2 Device.



**Abb. 15-8** Die Registerkarte „Globals“ des PB1650DIO2 Device

#### *Piggyback Position*

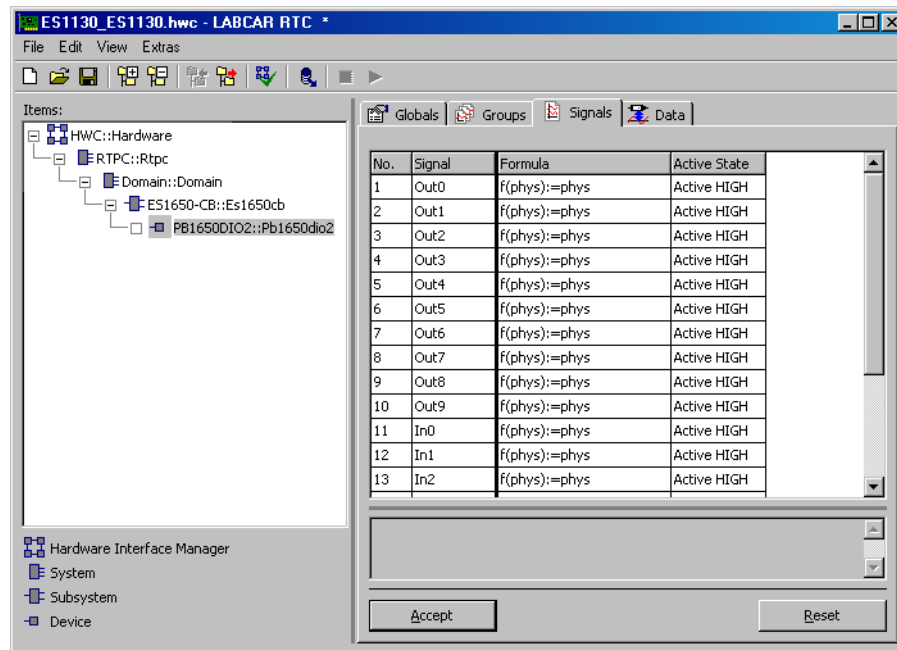
Im Auswahlfeld „Piggyback Position“ an, ob das Aufsteckmodul auf dem oberen oder unteren Steckplatz der ES1650.1 steckt (siehe Abb. 15-8).

### 15.5.2 Groups (PB1650DIO2 Device)

Die im Register „Groups“ verfügbaren Optionen sind im Abschnitt „Standardoptionen im Register „Groups““ auf Seite 47 beschrieben.

### 15.5.3 Signals (PB1650DIO2 Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PB1650DIO2 Devices beschrieben.



**Abb. 15-9** Die Registerkarte „Signals“ des PB1650DIO2 Devices

#### Active State

Für jeden Ein- bzw. Ausgang eines PB1650DIO2 Aufsteckmoduls kann im Auswahlfeld „Active State“ angegeben werden, ob das zugehörige Signal low aktiv ('Active LOW') oder high aktiv ('Active HIGH') ist (siehe Abb. 15-9).

Bei einem Eingang  $In_x$  ( $x = 0, 1 \dots 9$ ) bedeutet der Wert 0, dass das angelegte Signal inaktiv ist, der Wert 1 bedeutet, dass das angelegte Signal aktiv ist.

Bei einem Ausgang  $Out_x$  ( $x = 0, 1, \dots 9$ ) bedeutet der Wert 0, dass das zugehörige Signal inaktiv geschaltet wird, der Wert 1 bedeutet, dass das zugehörige Signal aktiv geschaltet wird.

### 15.5.4 Datentypen und Wertebereiche der Signale

Die 10 Signale  $Out_x$  ( $x = 0, 1, \dots 9$ ) des PB1650DIO2 Aufsteckmoduls sind vom Datentyp `uint8`. Das Signal wird vom RTIO-Treiber als Boolesche Größe interpretiert. Der Wert 0 entspricht `FALSE`, ein Wert ungleich 0 entspricht `TRUE`.

Die 10 Signale  $In_x$  ( $x = 0, 1, \dots 9$ ) sind vom Datentyp `uint8`. Der zurückgelieferte Signalwert ist entweder 0 oder 1.

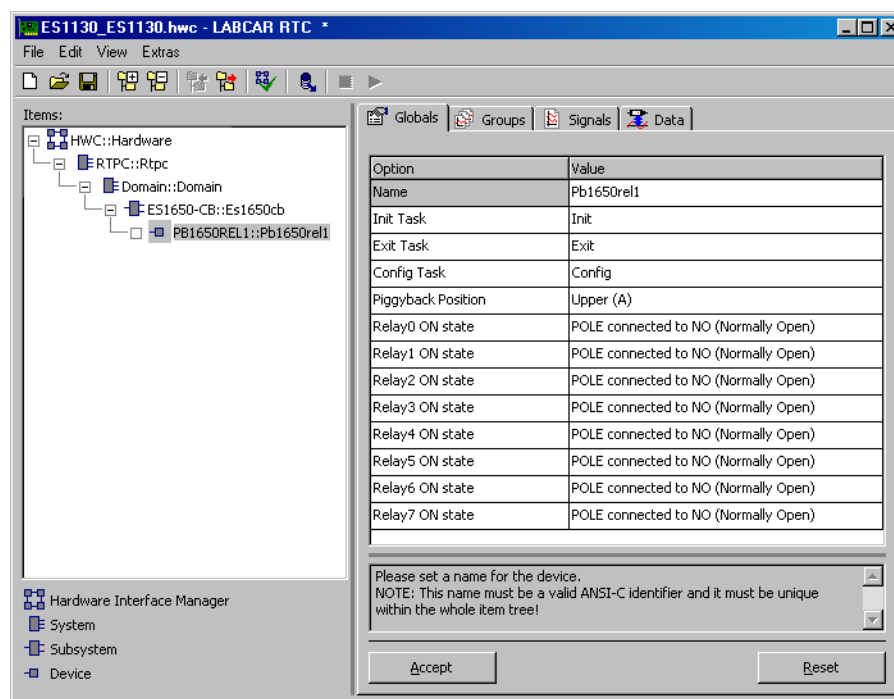
## 15.6 PB1650REL1

Das Aufsteckmodul PB1650REL1 stellt acht galvanisch getrennte Schalter im Spannungsbereich bis 175 V zur Verfügung. Es wird zusammen mit der ES1650.1 als Schaltermodul eingesetzt.

Typische Anwendungen sind die Simulation von manuellen Schaltern oder das Schalten von Lasten, die nicht mit Halbleiterschaltern geschaltet werden können. Weiter kann das PB1650REL1 Aufsteckmodul als analoger Multiplexer eingesetzt werden.

### 15.6.1 Globals (PB1650REL1 Device)

In diesem Abschnitt werden die globalen Einstellungen des PB1650REL1 Devices beschrieben.



**Abb. 15-10** Die Registerkarte „Globals“ des PB1650REL1 Devices

#### *Piggyback Position*

Im Auswahlfeld „Piggyback Position“ wird angegeben, ob das Aufsteckmodul auf dem oberen oder unteren Steckplatz der ES1650.1 steckt.

#### *Relayx ON state*

In den Auswahlfeldern „Relayx ON state“ (x = 0, 1, ... 7) wird festgelegt, welche Schalterstellung für das Relais x der geschlossene Schalterzustand ist.

„POLE connected to NO (Normally Open)“ bedeutet, dass bei geschlossenem Schalter der Anschluss „POLE“ des Relais mit dem Anschluss „NO“ (Normally Open) verbunden ist.

„POLE connected to NC (Normally Closed)“ bedeutet, dass bei geschlossenem Schalter der Anschluss „POLE“ des Relais mit dem Anschluss „NC“ (Normally Closed) verbunden ist.

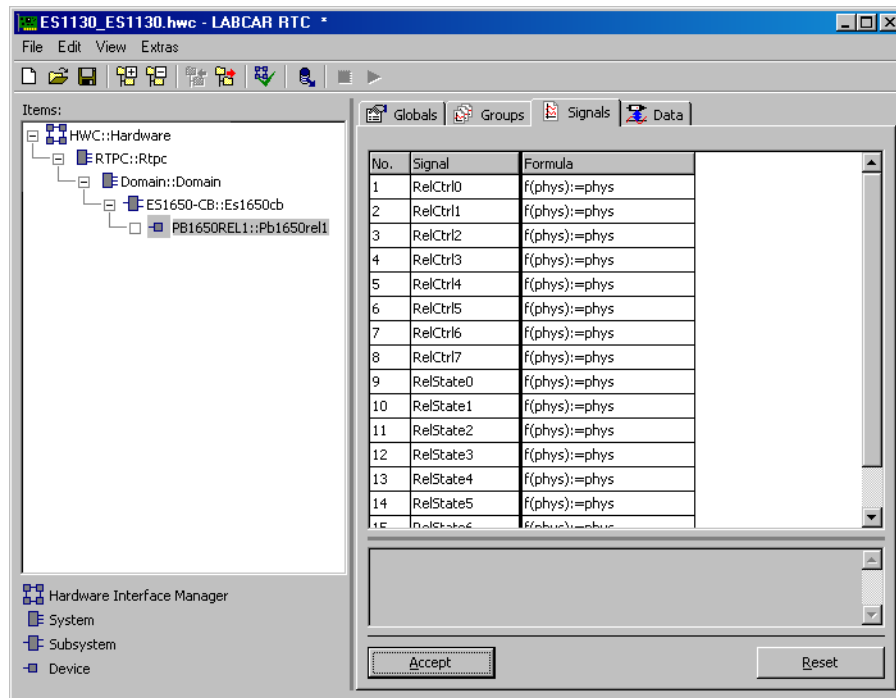


### 15.6.2 Groups (PB1650REL1 Device)

Die im Register „Groups“ verfügbaren Optionen sind im Abschnitt „Standardoptionen im Register „Groups““ auf Seite 47 beschrieben.

### 15.6.3 Signals (PB1650REL1 Device)

In diesem Abschnitt werden die signalspezifischen Einstellungen des PB1650REL1 Devices beschrieben.



**Abb. 15-11** Die Registerkarte „Signals“ des PB1650REL1 Devices

- Die Ausgänge *RelCtrl<sub>x</sub>* ( $x = 0, 1, \dots, 7$ ) steuern die Schalterstellung des zugeordneten Relais  $x$ .  
Ein Wert 0 für *RelCtrl<sub>x</sub>* öffnet das Relais  $x$ . Ein Wert ungleich 0 für *RelCtrl<sub>x</sub>* schließt das Relais  $x$ .
- Die Eingänge *RelStatex* ( $x = 0, 1, \dots, 7$ ) zeigen die momentane Schalterstellung des zugeordneten Relais  $x$  an.  
Hat *RelStatex* den Wert 0, so ist das Relais  $x$  geöffnet. Hat *RelStatex* den Wert 1, so ist das Relais  $x$  geschlossen.

Was jeweils als geöffneter und geschlossener Relaiszustand gilt, wird in der Registerkarte „Globals“ definiert (siehe „Relayx ON state“ auf Seite 388).

### 15.6.4 Datentypen und Wertebereiche der Signale

#### *RelCtrl<sub>x</sub>*

Die 8 Signale *RelCtrl<sub>x</sub>* ( $x = 0, 1, \dots, 7$ ) des PB1650REL1 Devices sind vom Datentyp `uint8`. Das Signal wird vom RTIO-Treiber als Boolesche Größe interpretiert. Der Wert 0 entspricht `FALSE`, ein Wert ungleich 0 entspricht `TRUE`.

### *RelStatex*

---

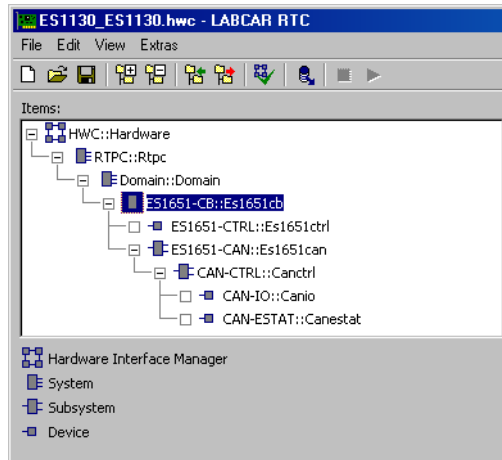
Die 8 Signale *RelStatex* ( $x = 0, 1, \dots, 7$ ) sind vom Datentyp `uint8`. Der zurückgelieferte Signalwert ist entweder 0 oder 1.

## 16 ES1651.1 Carrier Board

Dieser Abschnitt beschreibt die RTIO-Einbindung des ES1651.1 Carrier Board, das im LABCAR-Umfeld als Trägerkarte für bis zu vier I/O-Module eingesetzt wird. Des Weiteren bietet die Karte einen Winkeltakt-Bus sowie eine Anbindung an den CAN-Bus über zwei identische CAN-Controller.

### *Aufbau des ES1651.1 RTIO-Baums*

Ein ES1651.1 Carrier Board wird im VMEbus-Baugruppenträger verbaut und dort ist sie direkt über den VMEbus mit dem Experimentaltarget verbunden. Die beschriebene Hardware-Anbindung spiegelt sich im RTIO-Baum wider (Abb. 16-1).



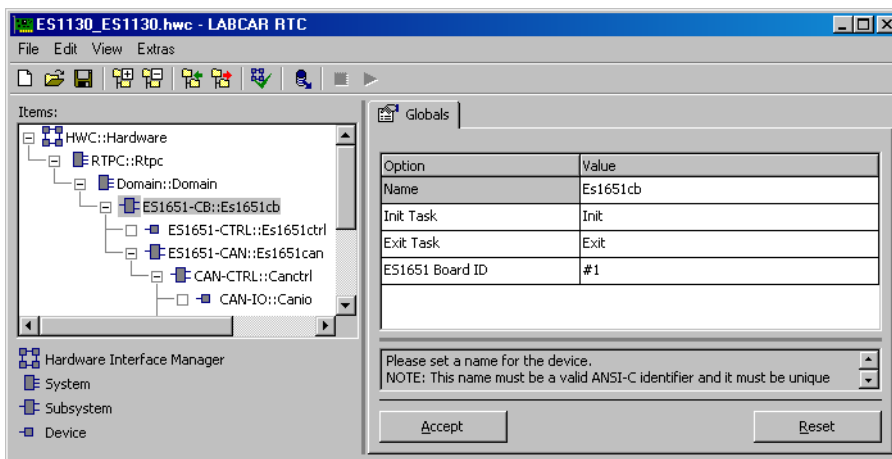
**Abb. 16-1** RTIO-Hardwarebeschreibung mit eingebundener ES1651.1

Die Hardware eines ES1651.1 Carrier Board besitzt einen Synchronisationsbus mit vier Leitungen sowie - speziell für den Test von Motorsteuergeräten - einen Winkeltaktbus, über den die I/O-Module zur drehzahlsynchronen Signalerfassung bzw. Signalgenerierung auf einen Winkeltakt synchronisiert werden können. Die Konfiguration und Ansteuerung der Busleitungen erfolgt mit dem ES1651-CTRL RTIO-Element. Die Konfiguration und Ansteuerung des CAN-Bus wird über das RTIO-Element „ES1651-CAN“ durchgeführt.

## 16.1 ES1651-CB Subsystem

### 16.1.1 Globals (ES1651-CB Subsystem)

Abb. 16-2 zeigt die Registerkarte „Globals“ eines ES1651-CB Subsystems.



**Abb. 16-2** Die Registerkarte „Globals“ des ES1651-CB Subsystems  
*ES1651 Board ID*

Dieses Optionsfeld dient zur Identifizierung des ES1651.1 Carrier Board. Es stellt die Zuordnung zwischen der RTIO-Hardwarebeschreibung und der ES1651.1 im VMEbus-Baugruppenträger her, für die diese Beschreibung gültig ist. Die Nummerierung der ES1651.1 Carrier Boards im Baugruppenträger erfolgt dabei von links nach rechts (beginnend mit 1). Es können bis zu 20 ES1651.1 pro Baugruppenträger eingebunden werden.

Dieser RTIO-Parameter ist nicht zur Laufzeit einstellbar.

Tab. 16-1 fasst die Eigenschaften des Parameters „ES1651 Board ID“ zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
ES1651 Board ID	uint32	Nein	Identifizierung der ES1651.1. Zur Auswahl stehen #1 bis #20.
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

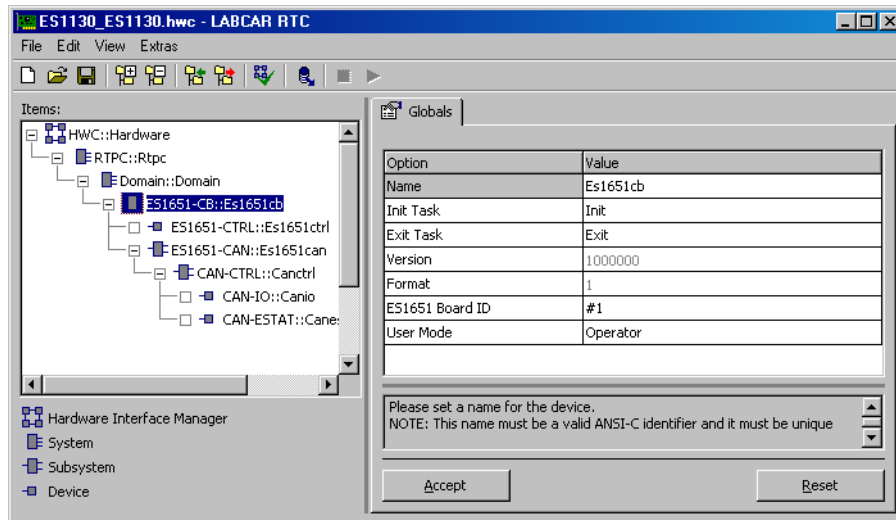
**Tab. 16-1** ES1651-CB Subsystem: Konfigurationsparameter der Registerkarte „Globals“

### 16.1.2 Versteckte Optionsfelder

Wird mit der rechten Maustaste in die Registerkarte „Globals“ geklickt und anschließend im erscheinenden kontextsensitiven Menü die Option „Show all Options“ gewählt, so werden weitere Optionsfelder sichtbar (Abb. 16-3).

Diese Optionsfelder erlauben es ETAS-Servicepersonal bei einer Fehlersuche zusätzliche Informationen über das ES1651.1 Carrier Board zu erhalten. Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten vom Anwender nicht abgeändert werden.

Abb. 16-3 zeigt die Registerkarte „Globals“ mit versteckten Optionen eines ES1651-CB Subsystems.



**Abb. 16-3** Die Registerkarte „Globals“ des ES1651-CB Subsystem mit versteckten Optionen

*User Mode*

Stellt den RTIO-Anwendermodus ein. Anwender sollten in diesem Listenfeld die Option „Operator“ wählen. Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

Tab. 16-2 fasst die Eigenschaften der Optionen zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
User Mode	uint8	Ja	Anwender-Modus 0: „Operator“ (Voreinstellung) 1: „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche.

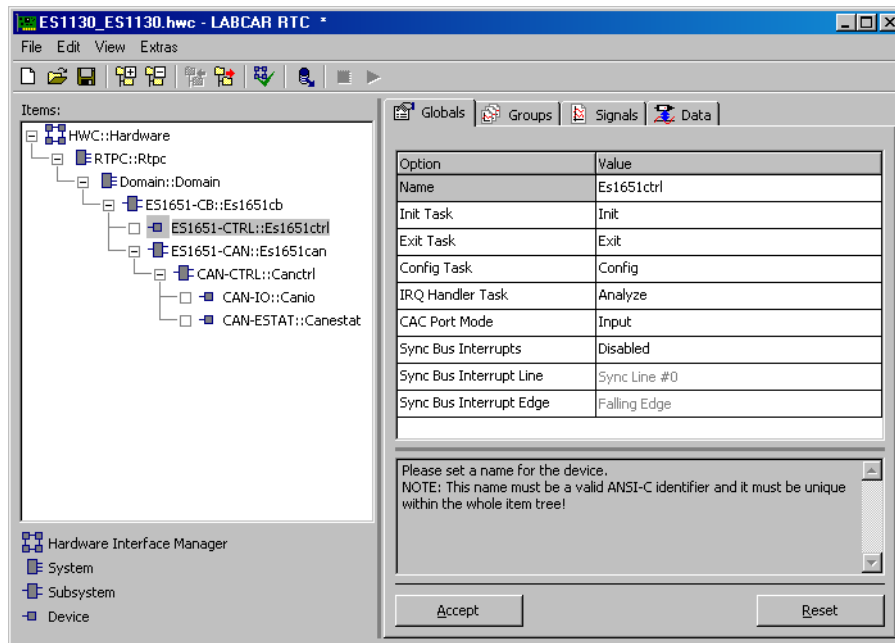
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-2** ES1651-CB Subsystem: Optionen der Registerkarte „Globals“

## 16.2 ES1651-CTRL Device

### 16.2.1 Globals (ES1651-CTRL Device)

Abb. 16-4 zeigt die Registerkarte „Globals“ eines ES1651-CTRL Device.



**Abb. 16-4** Die Registerkarte „Globals“ des ES1651-CTRL Device

#### *Analyze Task*

Zur Unterstützung von Interrupts des ES1651.1 Carrier Board wird eine zusätzliche „Analyze Task“ benötigt. Diese Task muss im Projekteditor in der Taskliste als Software-Task angelegt werden - die Anzahl der möglichen Taskaktivierungen muss zwischen 2 und 255 betragen.

#### *CAC Port Mode*

Dieses Optionsfeld dient zur Konfiguration des 9-poligen Sub-D-Anschlusses an der Frontplatte des ES1651.1 Carrier Board bezüglich des Winkeltaktbusses.

Der Anschluss kann dabei auf vier verschiedene Arten konfiguriert werden. Die Einstellung „Isolated“ entkoppelt den Anschluss vom Winkeltakt-Bus. Über die Option „Output“ können CAC-Signale der ES1651.1 externer Hardware zur Verfügung gestellt werden. Im Eingangsmodus „Input“ werden Winkeltaktsignale für Trägerboardhardware bzw. für die I/O-Module durch externe Quellen zur Verfügung gestellt. „Input ES1332 CAC“ gewährleistet die Verarbeitung kombinierter CAC-Synchronisations- und Taktsignale, welche von der ES1332 erzeugt werden können.

#### *Sync Bus Interrupts*

Dieses Optionsfeld aktiviert bzw. deaktiviert die Fähigkeit zur Erzeugung von VMEbus-Interrupts durch Signalübertragungen auf dem Synchronisationsbus. „Enabled“ bedeutet „Interrupt-Erzeugung ist aktiv“ und „Disabled“ bedeutet „Interrupt-Erzeugung inaktiv“.

#### *Sync Bus Interrupt Line*

---

Dieses Optionsfeld dient zur Einstellung der Interruptquelle, d.h. welche der vier Sync-Leitungen befähigt wird, den VMEbus-Interrupt bei einer Signalübertragung zu erzeugen. Diese Option ist nur dann editierbar, wenn die Option „Sync Bus Interrupts“ auf „Enabled“ steht.

#### *Sync Bus Interrupt Edge*

---

Dieses Optionsfeld dient zur Wahl der Flankentriggerung für die Generierung von VMEbus-Interrupts. Ein Interrupt wird dabei entweder bei einer fallenden oder steigenden Flanke erzeugt. Diese Option ist nur dann editierbar, wenn die Option „Sync Bus Interrupts“ auf „Enabled“ steht.

Tab. 16-3 fasst die Eigenschaften der Konfigurationsparameters zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
CAC Port Mode	uint8	Ja	Konfiguration des Sub-D-Anschlusses an der Frontplatte bezüglich des Winkeltaktbusses: 0: Anschluss ist von Winkeltaktbus abgekoppelt 1: Anschluss operiert als Ausgang, welcher die CAC-Signale externer Hardware bereitstellt 2: Anschluss operiert als Eingang und stellt CAC-Signale von der Trägerboardhardware oder von I/O-Modulen zur Verfügung 3: Anschluss operiert als Eingang und verarbeitet kombinierte CAC-Synchronisations- und Taktsignale, welche von der ES1332 erzeugt werden können
Sync Bus Interrupts	bool	Ja	Erzeugung von VMEbus-Interrupts durch Signalübertragungen auf dem Synchronisationsbus: False: deaktiviert True: aktiviert
Sync Bus Interrupt Line	uint8	Ja/Nein	Quelle für die VMEbus-Interruptgenerierung: 0: Sync-Leitung 0 1: Sync-Leitung 1 2: Sync-Leitung 2 3: Sync-Leitung 3
Sync Bus Interrupt Edge	uint8	Ja/Nein	Flankentriggerung für die Erzeugung von VMEbus-Interrupts: 0: Triggerung bei fallender Flanke 1: Triggerung bei steigender Flanke

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-3** ES1651-CB Device: Konfigurationsparameter der Registerkarte „Globals“

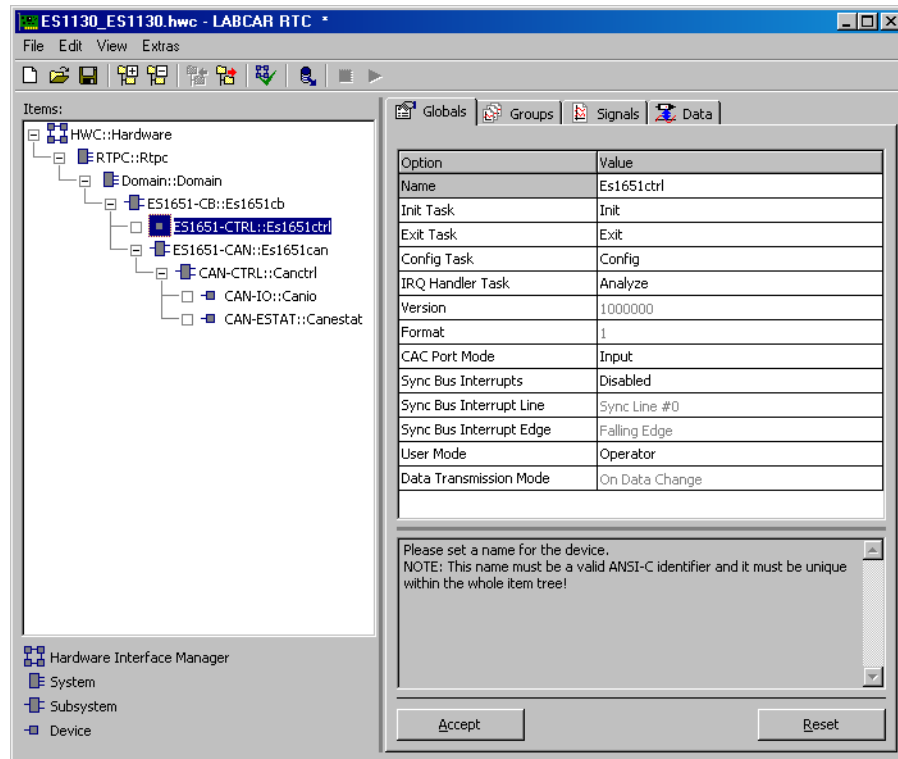
### 16.2.2 Versteckte Optionsfelder

Wird mit der rechten Maustaste in die Registerkarte „Globals“ geklickt und anschließend im erscheinenden kontext-sensitiven Menü die Option „Show all Options“ gewählt, so werden weitere Optionsfelder sichtbar (Abb. 16-5). Diese Optionsfelder erlauben es ETAS-Servicepersonal, bei einer Fehlersuche zusätzliche Informationen über das ES1651.1 Carrier Board zu erhalten.

Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten vom Anwender nicht abgeändert werden.



Abb. 16-5 zeigt die Registerkarte „Globals“ mit versteckten Optionen eines ES1651-CTRL Devices.



**Abb. 16-5** Die Registerkarte „Globals“ des ES1651-CTRL Devices mit versteckten Optionen

#### *User Mode*

Stellt den RTIO-Anwendermodus ein. Anwender sollten in diesem Listenfeld die Option „Operator“ wählen. Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

#### *Data Transmission Mode*

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeitdaten vom Experimentaltarget zum ES1651.1 Carrier Board übertragen werden.

Ist die Option „On Data Change“ aktiviert, werden die Daten einer Signalgruppe nur dann übertragen, wenn sich wenigstens ein Datum geändert hat. Ist die Option „Every Interval“ aktiviert, so werden die Echtzeitdaten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.

Tab. 16-4 fasst die Eigenschaften der Optionen zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
User Mode	uint8	Ja	Anwender-Modus 0: „Operator“ ist Voreinstellung 1: „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche
Data Transmission Mode	uint8	Ja	Übertragungsmodus der Echtzeitdaten 0: „On Data Change“ 1: „Every Interval“ „On Data Change“ ist Voreinstellung Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-4** ES1651-CTRL Device: Optionen der Registerkarte „Globals“

### 16.2.3 Groups (ES1651-CTRL Device)

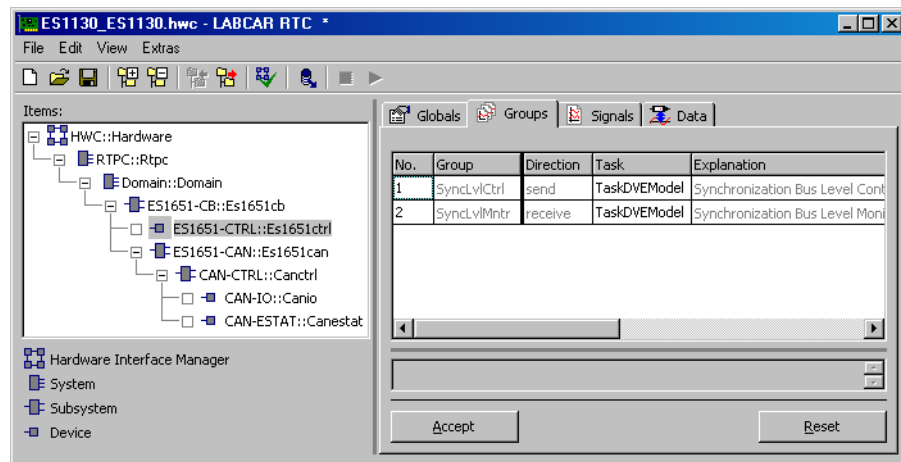
Das ES1651-CTRL Device besitzt zwei Signalgruppen, die vom Experimentaltarget zum ES1651.1 Carrier Board bzw. von der ES1651.1 zum Experimentaltarget übertragen werden (siehe Abb. 16-6).

Die Signalgruppe „SyncLvlCtrl“ transportiert Pegelinformationen für die Sync-Leitungen, die von RTIO aus gesteuert werden, zur ES1651.1.

Die Signalgruppe „SyncLvlMntr“ transportiert Pegelinformationen der Sync-Leitungen zur RTIO. Dadurch lassen sich die einzelnen Pegelzustände der Sync-Leitungen über die RTIO darstellen bzw. überwachen.

Jeder Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erzeugenden bzw. zu überwachenden Synchronisationssignale. Sinnvolle Aktivierungsperioden sind durch Laufzeiten der Firmware auf den Trägerkarten auf 1 Millisekunde nach unten begrenzt.

Abb. 16-6 zeigt die Registerkarte „Groups“ eines ES1651-CTRL Devices.



**Abb. 16-6** Die Registerkarte „Groups“ des ES1651-CTRL Devices

*RTIO-Signale der Signalgruppe „SynclvlCtrl“*

Die Signalgruppe „SynclvlCtrl“ umfasst vier RTIO-Signale „SynclvlCtrl\_0“ bis „SynclvlCtrl\_3“, die vom Datentyp „bool“ sind. Wird die Sync-Leitung x (x = 0, 1, 2, 3) dahingehend konfiguriert, dass sie von der RTIO aus gesteuert wird, so definiert der Wert des RTIO-Signals „SynclvlCtrl\_x“ den Pegel der Leitung. „0“ bedeutet dabei einen Low-Pegel, „1“ bedeutet einen High-Pegel. Wird die Sync-Leitung nicht von der RTIO gesteuert, so hat das Signal keine Bedeutung - die Spalte „Data“ der Sync-Leitungen, welche nicht von der RTIO gesteuert werden, ist inaktiv.

Tab. 16-5 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
SynclvlCtrl_0 bis SynclvlCtrl_3	bool	Ja	Pegelsteuerung auf Sync-Leitung x 0: Low-Pegel 1: High-Pegel
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 16-5** RTIO-Signale der Gruppe „SynclvlCtrl“

*RTIO-Signale der Signalgruppe „SynclvlMntr“*

Die Signalgruppe „SynclvlMntr“ umfasst vier RTIO-Signale „SynclvlMntr\_0“ bis „SynclvlMntr\_3“, die vom Datentyp „bool“ sind. Sie wird dazu verwendet, den aktuellen Pegel der betreffenden Sync-Leitung zu überwachen.

„0“ bedeutet dabei einen Low-Pegel, „1“ bedeutet einen High-Pegel auf der konfigurierten Sync-Leitung. Wird der Signalgruppe eine Task mit periodischer Aktivierung zugeordnet, so ist die zyklische Überwachung aller Sync-Leitungen dauerhaft aktiv.

Tab. 16-6 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
SyncLvlMntr_0 bis SyncLvlMntr_3	bool	Nein	Pegelüberwachung auf Sync-Leitung x 0: Low-Pegel 1: High-Pegel

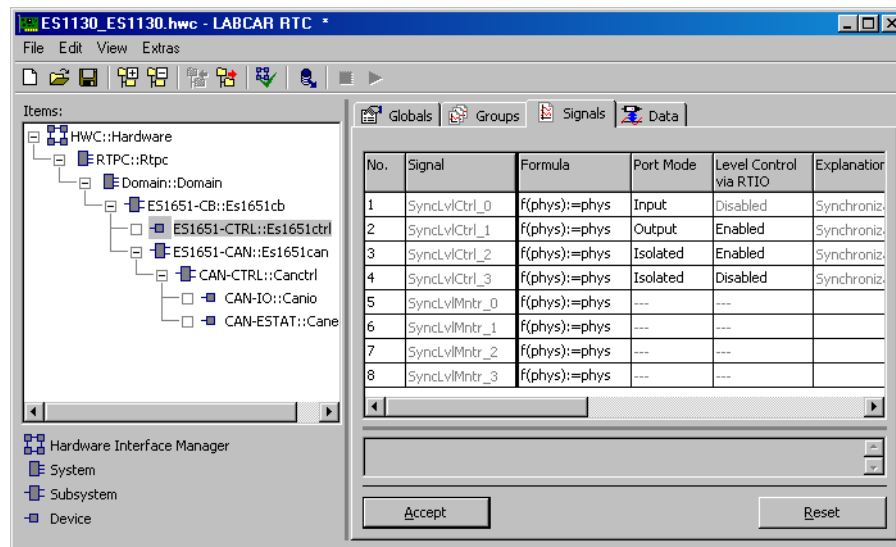
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-6** RTIO-Signale der Gruppe „SyncLvlMntr“

16.2.4 Signals (ES1651-CTRL Device)

In der Registerkarte „Signals“ wird die Konfiguration der Sync-Leitungen eines ES1651.1 Carrier Board durchgeführt. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Abb. 16-7 zeigt die Registerkarte „Signals“ eines ES1651-CTRL Devices.



**Abb. 16-7** Die Registerkarte „Signals“ des ES1651-CTRL Devices

*Port Mode*

Dieses Optionsfeld dient zur Konfiguration der Sync-Leitungen bezüglich des an der Frontplatte befindlichen Sub-D-Anschlusses.

Mit „Isolated“ kann die Sync-Leitung x (x = 0, 1, 2, 3) vom Anschluss abgekoppelt werden. „Output“ konfiguriert die Sync-Leitung als Ausgang, wodurch Synchronisationssignale externer Hardware zur Verfügung gestellt werden kann. Über die Option „Input“ werden Synchronisationssignale der Trägerboardhardware bzw. den I/O-Modulen zur Verfügung gestellt.

*Level Control via RTIO*

Wird die Sync-Leitung x (x = 0, 1, 2, 3) mit der „Port Mode“ Option „Isolated“ oder „Output“ konfiguriert, wird die Einstellungsmöglichkeit dieses Optionsfeldes aktiv. Die Pegelsteuerung der Sync-Leitung x (x = 0, 1, 2, 3) kann nun über

die RTIO bzw. das Modell erfolgen. Dementsprechend wird das zugehörige Eingabefeld in der Registerkarte „Data“ editierbar. „0“ bedeutet einen Low-Pegel und „1“ einen High-Pegel auf der Sync-Leitung x (x = 0, 1, 2, 3).

Tab. 16-7 fasst die Eigenschaften der RTIO-Signale zusammen.

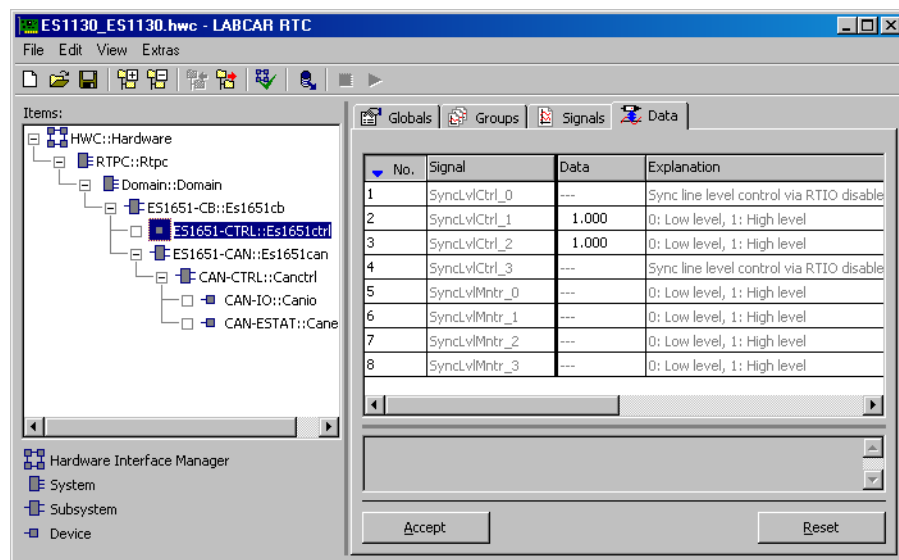
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Port Mode	uint8	Ja	Konfiguriert Sync-Leitungen bezüglich des Sub-D-Anschlusses an der Frontplatte: 0: „Isolated“, Sync-Leitung ist vom Anschluss entkoppelt 1: „Output“, Sync-Signale werden externer Hardware zur Verfügung gestellt 2: „Input“, Sync-Leitung dient als Eingang, Signale werden eingespeist
Level Control via RTIO	uint	Ja/Nein	Steuerung der Pegelzustände der Sync-Leitungen mittels RTIO: 0: „Disabled“, Steuerung erfolgt nicht über RTIO 1: „Enabled“, Steuerung der Pegel erfolgt über RTIO

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-7** RTIO-Signale der Registerkarte „Signals“

16.2.5 Data (ES1651-CTRL Device)

Abb. 16-8 zeigt die Registerkarte „Data“ eines ES1651-CTRL Devices.



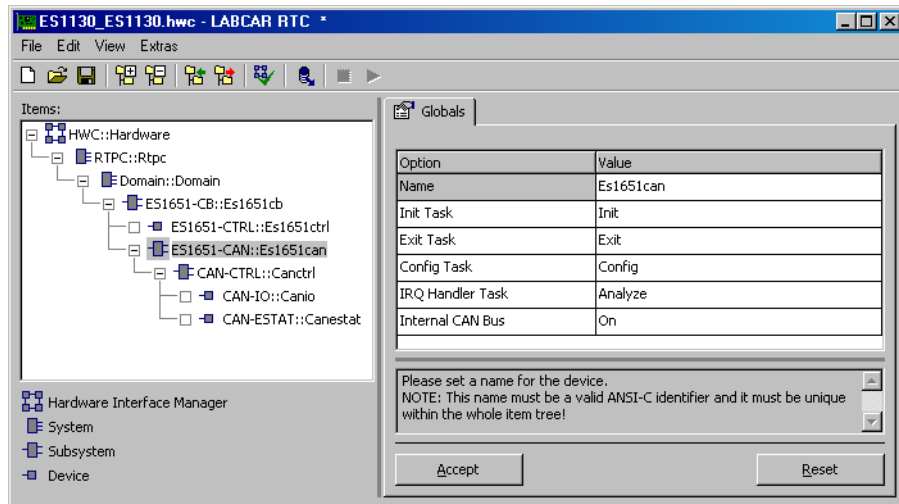
**Abb. 16-8** Die Registerkarte „Data“ des ES1651-CTRL Devices

Man erkennt, dass die Signale „SyncLvlCtrl\_1“ und „SyncLvlCtrl\_2“ dazu verwendet werden, um die Pegel auf den entsprechenden Sync-Leitungen zu steuern. Beide Pegel wurden über die RTIO „High“ aktiv gesetzt.

### 16.3 ES1651-CAN Subsystem

#### 16.3.1 Globals (ES1651-CAN Subsystem)

Abb. 16-9 zeigt die Registerkarte „Globals“ eines ES1651-CAN Subsystems.



**Abb. 16-9** Die Registerkarte „Globals“ des ES1651-CAN Subsystems

#### *IRQ Handler Task*

Für die Unterstützung von Interrupts des ES1651.1 Carrier Board wird eine zusätzliche Task benötigt. Diese Task muss im Projekteditor in der Taskliste als Software-Task angelegt werden. Die Anzahl der möglichen Taskaktivierungen muss zwischen 2 und 255 liegen.

#### *Internal CAN Bus*

Dieses Optionsfeld dient zur internen Verbindung der beiden CAN-Controller untereinander. „Off“ bedeutet die beiden CAN-Controller sind nicht verbunden und „On“ steht für die hardwaretechnische Kopplung der CAN-Controller.

Tab. 16-8 fasst die Eigenschaften der Konfigurationsparameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Internal CAN Bus	bool	Ja	Konfiguration der CAN-Controller 0: „Off“, CAN-Controller entkoppelt 1: „On“, CAN-Controller „on Board“ verbunden

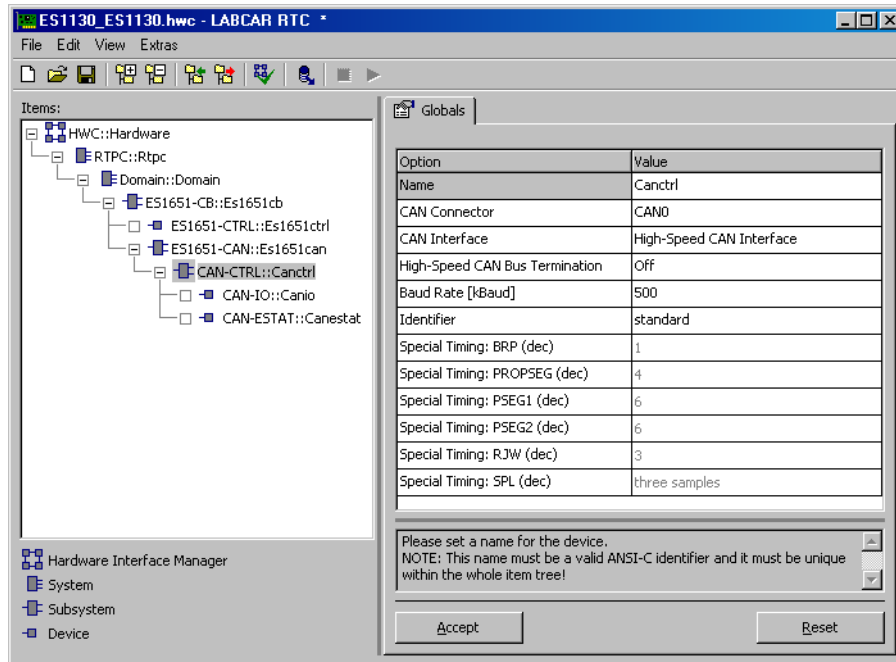
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-8** ES1651-CAN Parameter der Registerkarte „Globals“

## 16.4 CAN-CTRL Subsystem

### 16.4.1 Globals (CAN-CTRL Subsystem)

Abb. 16-10 zeigt die Registerkarte „Globals“ eines CAN-CTRL Subsystems.



**Abb. 16-10** Die Registerkarte „Globals“ des CAN-CTRL Subsystems

#### *CAN Connector*

Dieses Optionsfeld dient zur Zuordnung des CAN-CTRL Subsystem zu einem CAN-Controller bzw. CAN-Anschluss an der Frontplatte der ES1651.1. Hierbei ist zu beachten, dass ein CAN-CTRL Subsystem nur einem einzigen CAN-Controller zugeordnet werden kann.

#### *CAN Interface*

Mit diesem Optionsfeld kann zwischen der „High-Speed“ und „Low-Speed“ CAN-Funktionalität gewechselt werden. Die Option „High-Speed CAN Interface“ definiert den CAN-Controller als High-Speed-Schnittstelle, während „Fault-Tolerant CAN Interface“ den CAN-Controller als Low-Speed-Schnittstelle definiert.

#### *High-Speed CAN Bus*

Über diese Option kann ein 120 Ohm Abschlusswiderstand für die High-Speed-CAN-Schnittstelle geschaltet werden. Diese Option ist nur bei der „High-Speed CAN Interface“ Konfiguration aktiv und online editierbar. „On“ bedeutet dabei die „Terminierung aktiv“ und „Off“ steht für „Terminierung inaktiv“.

### *Baud Rate [kBaud]*

---

In diesem Optionsfeld kann die Übertragungsrate des CAN-Controllers festgelegt werden. Es stehen hierfür die Standard-Baudraten (1000, 500, 250, 125, 100, 50, 20, 10 kBaud) zur Verfügung. Mit Hilfe der Einstellung „Special Timing“ wird die Low-Level-Ansteuerung des CAN-Controllers bezüglich Bit-Timing und Baud Rate aktiviert.

Bei der Konfiguration der „Baud Rate“ werden automatisch die Optionsfelder „Special Timing“ bezüglich der eingestellten Übertragungsrate mit Werten konfiguriert und zur Firmware transferiert.

### *Special Timing: BRP (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Baud Rate Prescaler“, welcher aus dem Eingangstakt des CAN-Controllers die Baud Rate bestimmt. Der Wertebereich dieser Einstellung beträgt 0 - 255.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

### *Special Timing: PROPSEG (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung der „Propagation Segment Time“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 7.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

### *Special Timing: PSEG1 (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Phase Segment 1“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 7.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

### *Special Timing: PSEG2 (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Phase Segment 2“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 7.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

### *Special Timing: RJW (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung der „Resynchronization Jump Width“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 3.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.



### *Special Timing: SPL (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Sampling Mode“ und bestimmt, wie oft das Signal abgetastet wird, um den logischen Zustand zu ermitteln. Die möglichen Einstellungen sind 1 (einmal abgetastet) oder 3 (dreimal abgetastet).

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

Die Formel zur Berechnung der CAN-Bus Baud Rate lautet:

$$\text{Baud Rate} = 20 \text{ MHz} / [(\text{BRP} + 1) \times (4 + \text{PROPSEG} + \text{TSEG1} + \text{TSEG2})]$$

### *Identifier*

---

In dieser Zeile kann die Länge des Identifier-Felds der CAN-Botschaft ausgewählt werden. Bei CAN-Botschaften kann grundsätzlich zwischen „standard“ Frames mit 11-Bit Identifiern oder „extended“ Frames mit 29-Bit Identifiern gewählt werden.

#### **Hinweis**

*Pro CAN-Controller muss bezüglich der Länge des Identifiers eine einheitliche Einstellung vorliegen.*

#### **Hinweis**

*Wenn der Identifier „standard“ gewählt wird, ist in der Registerkarte „Groups“ des „CAN-IO“ RTIO-Elements nur die Eingabe von 11 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die „most significant bits“ (MSB) abgeschnitten. Eine Warnung wird nicht angezeigt.*

#### **Hinweis**

*Wenn der Identifier „extended“ gewählt wird, ist nur die Eingabe von 29 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die „most significant bits“ (MSB) abgeschnitten. Eine Warnung wird nicht angezeigt.*

Tab. 16-9 fasst die Eigenschaften der Konfigurationsparameter zusammen.

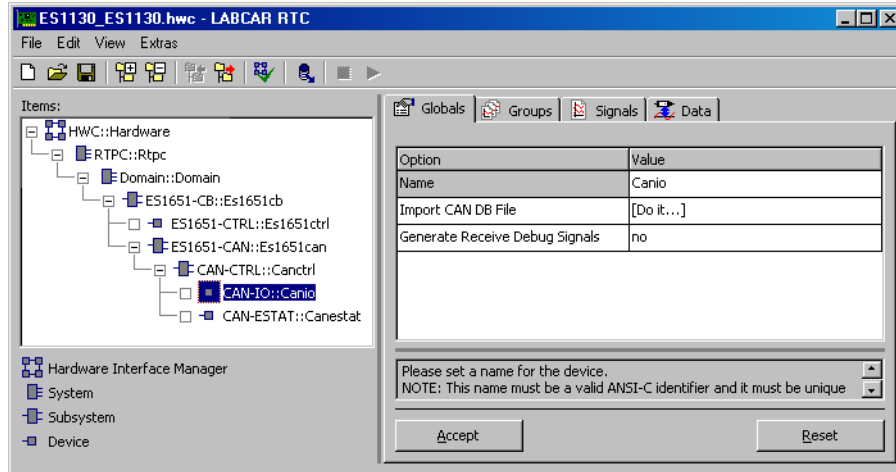
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
CAN Connector	uint32	Nein	Zuordnung zwischen CAN-CTRL Subsystem und CAN-Controller: 0: „CAN0“, CAN-Controller 0 1: „CAN1“, CAN-Controller 1
CAN Interface	uint8	Ja	Konfiguration des CAN-Controllers: 0: „Fault-Tolerant CAN Interface“ 1: „High-Speed CAN Interface“
High-Speed CAN Bus	uint8	Ja	Zuschaltung des Abschlusswiderstandes im Falle der High-Speed-CAN-Funktionalität: 0: „Off“, Terminierung inaktiv 1: „On“, Terminierung aktiv
Identifier	uint8	Nein	Identifier Länge der CAN-Botschaft: 0: „standard“, 11-Bit Länge 1: „extended“, 29-Bit Länge
Special Timing: BPR (dec)	uint32	Ja/Nein	Bestimmung der Baud Rate aus dem Eingangstakt: Wertebereich: 0 bis 255
Special Timing: PROPSEG (dec)	uint8	Ja/Nein	„Propagation Segment Time“ Wertebereich: 0 bis 7
Special Timing: PSEG1 (dec)	uint8	Ja/Nein	„Phase Segment 1“ Wertebereich: 0 bis 7
Special Timing: PSEG2 (dec)	uint8	Ja/Nein	„Phase Segment 2“ Wertebereich: 0 bis 7
Special Timing: RJW (dec)	uint8	Ja/Nein	„Resynchronization Jump Width“ Wertebereich: 0 bis 3
Special Timing: SPL (dec)	uint8	Ja/Nein	Abtastrate des Signals: 0: „one sample“, garantiert eine höhere Übertragungsgeschwindigkeit 1: „three samples“, garantiert eine höhere Signalqualität

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-9** Konfigurationsparameter der Registerkarte „Globals“

## 16.4.2 Globals (CAN-IO Device)

Abb. 16-11 zeigt die Registerkarte „Globals“ eines CAN-IO Devices.

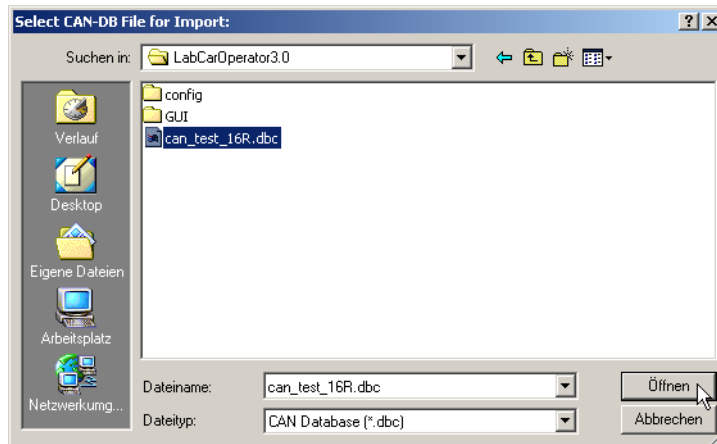


**Abb. 16-11** Die Registerkarte „Globals“ des CAN-IO Devices

#### Import CAN DB File

Über diese Option kann eine CAN-Datenbank-Datei eingelesen werden, welche mit dem CANdb Datenverwaltungsprogramm der Firma Vector Informatik erstellt wurde. Mit Hilfe dieser Datei können nun bei Bedarf automatisch CAN-Botschaften und -Signale angelegt werden.

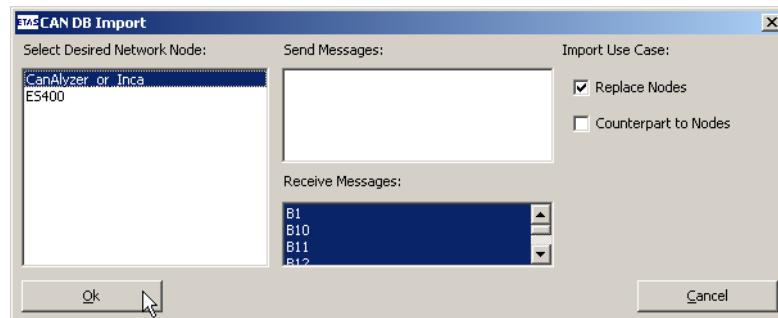
Nach Drücken von [Do it...] öffnet sich folgendes Dialogfenster:



**Abb. 16-12** CANdb Auswahl

In diesem Dialogfenster kann die gewünschte zu importierende CAN DB-Datei ausgewählt werden.

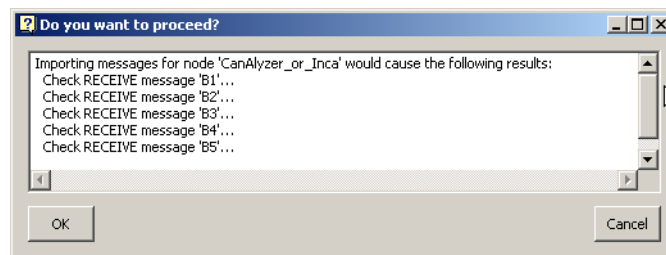
Nach Betätigen der Schaltfläche **Open** öffnet sich nachfolgendes Dialogfenster, indem das weitere Verfahren spezifiziert werden kann:



**Abb. 16-13** CANdb Import Dialog

In einer CAN DB Datei werden üblicherweise mehrere Knoten eines CAN-Netzwerkes beschrieben. Alle vorhandenen Knoten werden in der linken Liste (Network Nodes) aufgelistet. Die beiden Listen rechts davon (Send Messages/Receive Messages) listen alle für den aktuell selektierten Knoten definierten CAN Botschaften auf. Die Botschaften, die selektiert sind, werden für den Import herangezogen. Zusätzlich kann der „Import Use Case“ noch ausgewählt werden. Die Option „Replace Nodes“ bedeutet, dass das CAN-IO Device die Rolle des Netzwerkknotens übernimmt; d.h. eine Sendebotschaft des Knotens wird auch als Sendebotschaft des CAN-IO Devices umgesetzt usw. Die Option „Counterpart to Nodes“ bedeutet, dass das CAN-IO Device das Gegenstück zum Netzwerkknoten bildet; d.h. eine Sendebotschaft des Knotens ruft eine Receive-Botschaft hervor usw.

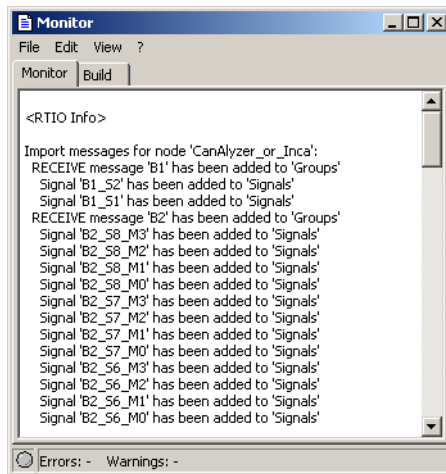
Nach Drücken der Schaltfläche **OK** erfolgt eine Überprüfung der zu importierenden Daten (CAN-Messages und -Signale) mit den vorhandenen Signalgruppen und Signalen. Das Resultat dieser Überprüfung wird anschließend in einem Dialogfenster angezeigt:



**Abb. 16-14** CANdb Import Check

Bis jetzt ist an dem vorhandenen CAN-IO Device noch nichts verändert worden. Erst nach Drücken von **OK** wird der tatsächliche Importvorgang gestartet. Der Importvorgang fügt die importierten Messages den Signalgruppen hinzu und sorgt auch für die Definition der entsprechenden Signale.

Nach Beendigung des Imports erscheint im Fenster „Monitor“ das detaillierte Protokoll des Importvorgangs:



**Abb. 16-15** CANdb Import Protokoll

### Hinweis

Beim Importieren einer CAN DB-Datei werden nur 29 Bit (Identifizier „extended“) bzw. 11 Bit (Identifizier „standard“) automatisch in das Identifizier-Feld eingefügt. Wenn im CAN-CTRL Subsystem der Identifizier „standard“ gewählt wurde, die CAN DB-Datei aber Signale mit 29 Bit-Identifiern (ID > 231) enthält, geschieht Folgendes:

- 11 Bit (Bits [28... 18]) werden automatisch in das Identifizier-Feld geschrieben. Die übrigen Bits (MSB) werden verworfen.
- Im Monitorfenster werden Warnungen angezeigt. Da nur ein Identifizier ausgewählt werden kann, kommt es bei CAN DBDateien, die sowohl „standard“ als auch „extended“ Identifizier enthalten, zu Konflikten.

### Generate Receive Debug Signals

Ist diese Option eingeschaltet (= yes), so werden für jede „receive“-Signalgruppe zwei zusätzliche Signale generiert.

- <GroupName>\_Diag\_dT
- <GroupName>\_Diag\_Rec

Das Signal „...dT“ gibt die Differenz in Sekunden zur vorangegangenen empfangenen Botschaft an.

### Hinweis

Dieses Signal kann beim normalen Botschaftsempfang (IRQ = no) verwendet werden, um den Empfang zu überwachen. Wird z.B. der CANbus unterbrochen, so erhöht sich der Wert laufend im Raster der Empfangstask. Aus Performance-Gründen ist dieser Wert nicht gegen Überlauf geschützt, der bei ca. 300 s auftritt!

Im Gegensatz hierzu kann beim Botschaftsempfang mit Interrupt (IRQ = yes) keine echte Empfangsüberwachung durchgeführt werden, da die Berechnung des Wertes erst in der Interrupt-Task erfolgt. Da beim Ausbleiben des Empfangs

diese Berechnung auch nicht stattfindet, bleibt somit der alte Wert eingefroren. Damit kann beim Interrupt-Empfang nicht unterschieden werden, ob der Empfang in einem exakt regelmäßigen Raster erfolgt oder unterbrochen ist.

Das Signal „...Rec“ wird bei jedem Botschaftsempfang mit dem Wert „true“ beschrieben. Wird dieses Signal auf eine Send-Receive-Message abgebildet und diese von der Anwendung nach jedem Lesen auf „false“ zurückgesetzt, so kann die Anwendung sehr leicht feststellen, ob zwischen den Rechenzyklen ein Botschaftsempfang stattgefunden hat.

#### 16.4.3 Groups (CAN-IO Device)

In dieser Registerkarte werden die CAN-Botschaften in Form von Signalgruppen spezifiziert.

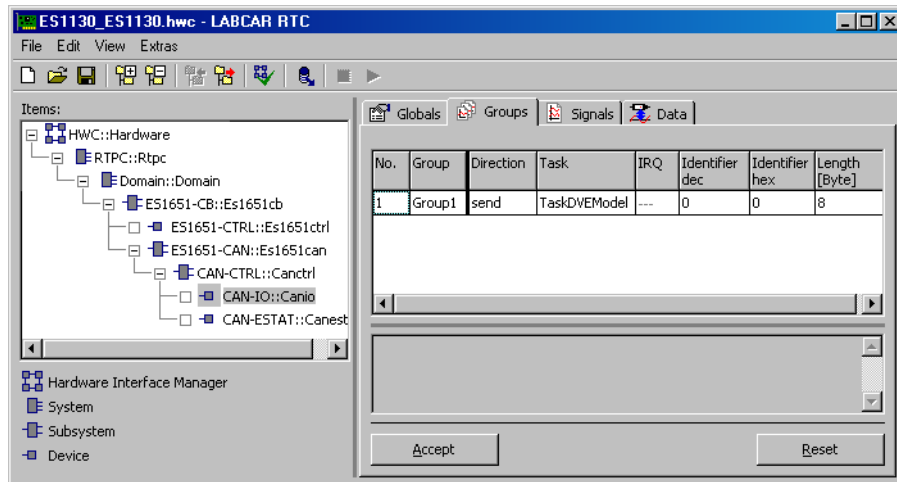
##### **Hinweis**

*Neue Signalgruppen bzw. CAN-Botschaften kann man über das Kontextmenü im Register „Groups“ anlegen.*

##### **Hinweis**

*Bei Änderung des Signalgruppennamens, des Signalnamens oder der Signalrichtung „Direction“ wird die eventuell zugeordnete Message u.U. nicht mehr automatisch zugeordnet und muss manuell neu zugeordnet werden.*

Abb. 16-16 zeigt die Registerkarte „Globals“ eines CAN-IO Devices.



**Abb. 16-16** Die Registerkarte „Groups“ des CAN-IO Devices

#### 16.4.4 Die CAN-Botschaften als Signalgruppen

##### *Direction*

Hier kann die Richtung der CAN-Botschaft bestimmt werden („send“ = Sendebotschaft, „receive“ = Empfangsbotschaft).

##### *Task*

Hier wird die Task angegeben, in welcher die Botschaft gesendet, bzw. empfangen werden soll. Soll eine Empfangsbotschaft im Interrupt-Betrieb empfangen werden, so wird diese Einstellung zurückgesetzt und gesperrt.

### *IRQ*

---

Diese Option gibt an, ob die betreffende Empfangsbotschaft im Interrupt-Betrieb empfangen werden soll oder nicht. Für den „normalen“ Empfang muss die CAN-Botschaft in einer Task mindestens so oft „gepollt“ werden, wie sie von der Gegenstation versendet werden kann.

Bei dieser Betriebsart sind solche CAN-Botschaften problematisch, die in keinem festen Raster versendet werden oder sogar nur sporadisch auftauchen. Für diese ist der Interrupt-Empfang ideal, da er eine Botschaftsverarbeitung genau dann anstößt, wenn die Botschaft empfangen wurde.

### *Identifier (dec / hex)*

---

Hier muss der Botschafts-Identifizierer eingegeben werden. Je nach gewählter Identifiziertyp-Einstellung in dem darüberliegenden CAN-CTRL Subsystem („standard“ oder „extended“) kann hier der Wert unterschiedlich groß werden:

Jede Signalgruppe bzw. CAN-Botschaft muss einen anderen Identifizierer besitzen.

- „Standard“ Identifizierer: 11 Bit 2 047 dec 7 FF hex
- „Extended“ Identifizierer: 29 Bit 536 870 911 dec F FF FF FF hex

### *Length [Byte]*

---

Legt die Anzahl der Nutzdatenbytes, welche die betreffende CAN-Botschaft übertragen kann, fest (1 bis 8 Bytes).

### *Activated Task*

---

Für „receive“-CAN-Botschaften<sup>1</sup> (Signalgruppen) kann hier eine Software-Task angegeben werden, die immer dann aktiviert wird, wenn die relevante Signalgruppe empfangen und die Empfangstask ausgeführt wurde. Die hier eingegebene Task kann z.B. zur Nachbearbeitung verwendet werden. Diese Spalte ist standardmäßig nicht eingeblendet.

### *Prescaler*

---

Hier kann bei einer im Interrupt-Betrieb empfangenen Botschaft eingestellt werden, ab wann tatsächlich ein VMEbus-Interrupt ausgelöst wird, um die Daten zu übertragen. Die Standardeinstellung „1“ besagt, dass mit jeder empfangenen Botschaft auch eine Datenübertragung stattfindet. Wird dieser Wert z.B. auf „2“ erhöht, so wird der VMEbus-Interrupt erst mit jeder zweiten empfangenen Botschaft ausgelöst, welcher dann die Daten der zuletzt empfangenen Botschaft überträgt. Diese Spalte ist standardmäßig nicht eingeblendet.

### **Hinweis**

*Eine Vergrößerung dieses Wertes sollte erwogen werden, wenn die entsprechende Botschaft in sehr schneller Folge empfangen wird und damit die VMEbus-Interruptlast zu sehr erhöht.*

Tab. 16-10 fasst die Eigenschaften der Konfigurationsparameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Direction	uint8	Nein	Übertragungsrichtung der CAN-Botschaft: 0x0: „send“, Sender 0x1: „receive“, Empfänger 0x80: „irq“, Empfänger mittels Interrupt
Identifizier	uint32	Nein	Identifizier der CAN-Botschaft: „standard“: 11-Bit „extended“: 29-Bit
Length [Byte]	uint8	Nein	Festlegung der Nutzdatenlänge der CAN-Botschaft: Wertebereich: 1 bis 8
Prescaler	uint16	Nein	Anzahl der CAN-Botschaften für Interrupt Erzeugung: Wertebereich: 1 bis 100

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-10** Konfigurationsparameter der Registerkarte „Groups“

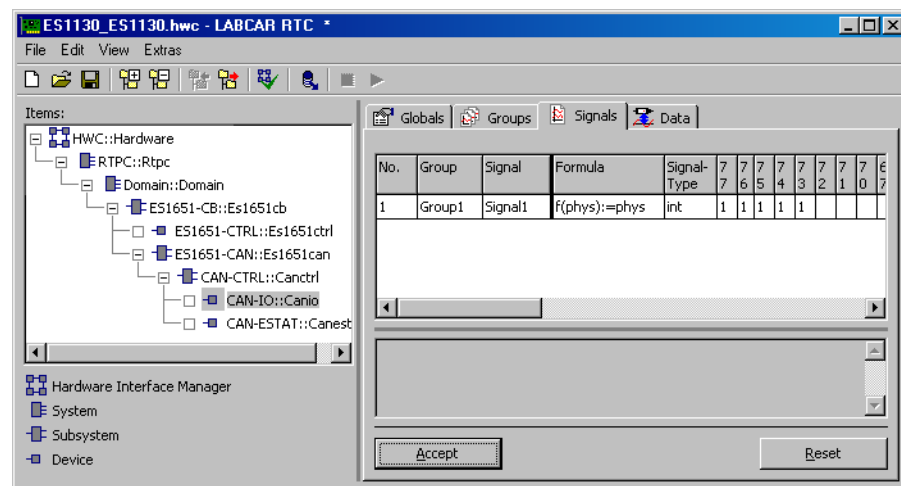
16.4.5 Signals (CAN-IO Device)

In der Registerkarte „Signals“ werden die CAN-Botschaften weiter spezifiziert.

**Hinweis**

Neue CAN-Signale lassen sich über das Kontextmenü im Register „Signals“ anlegen.

Abb. 16-17 zeigt die Registerkarte „Signals“ des CAN-IO Devices.



**Abb. 16-17** Die Registerkarte „Signals“ des CAN-IO Devices

*Group*

Hier erfolgt die Zuordnung eines Signals zu der gewünschten Signalgruppe.



### Signal Type

---

Hier wird der Signaltyp festgelegt, wie das Signal über den CAN-Bus übertragen wird.

Tab. 16-11 zeigt einen Überblick über die möglichen Signaltypen:

Signaltyp	Datentyp
int	Kennzeichnet ein vorzeichenbehaftetes Signal im Standard 2er-Komplement Datenformat (max. 32 Bit)
(s) int	Kennzeichnet ein vorzeichenbehaftetes Signal, in dem als höchstwertiges Bit das Vorzeichen und anschließend der Absolutwert des Signals übertragen wird. Ist das Vorzeichen-Bit gesetzt, so handelt es sich dabei um eine negative Zahl (max. 32 Bit)
uint	Kennzeichnet ein vorzeichenloses Signal (max. 32 Bit)
bool	Kennzeichnet ein Boolesches Signal. Hierbei darf in der Bit-Matrix nur ein einzelnes Bit markiert sein.
real	Kennzeichnet ein Fließkommawert im „Standard IEEE Float (4 Byte)“ Format. Hierbei dürfen entsprechend nur 32 Bit in der Bit-Matrix markiert sein. Aufbau des Datentyps nach IEEE: - Sign: 1 Bit - Exponent: 8 Bit - Fraction: 24 Bit

**Tab. 16-11** Signaltypen

### Bitmatrix

---

Eine CAN-Botschaft kann bis zu 8 Datenbytes übertragen. In einer Bitmatrix kann für jedes Signal festgelegt werden, welche Bits dieses Signal benötigt bzw. belegt.

Der Aufbau der Spalten ist wie folgt:

7	7	...	0	0	Byte-Nummer
7	6		1	0	Bit-Nummer

### Bedeutung der Bit-Felder:

- Leeres Feld  
Das entsprechende Signal verwendet das Bit nicht
- Belegtes Feld  
Das Signal benötigt dieses Bit an dieser Position
- „x“-Feld  
Das entsprechende Bit steht zur Datenübertragung nicht zur Verfügung, weil das Signal weniger Nutzdatenbytes hat (siehe Einstellung „Length“ im Register „Groups“).

**Bedienung der Bit-Felder:**

Über die Pfeiltasten der Tastatur kann die gewünschte Bit-Zelle ausgewählt werden. Mit der Leertaste kann die Zelle zwischen „unbelegt“ (Zelle leer) und „1“ (Bit belegt) „getoggelt“ werden.

Ein Anklicken mit der Maus „toggelt“ die entsprechende Zelle ebenso zwischen „unbelegt“ und „1“. Wird während des Klickens zusätzlich noch die <Alt>-Taste gedrückt, so wird der Wert „1“ bis „9“ inkrementiert (Blockbildung).

Mehrere Bits gleichzeitig kann man folgendermaßen selektieren:

1. Erstes Bit auswählen

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0		
			5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	Group1	Signal1																														

2. Zweites Bit auswählen

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	
			5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	Group1	Signal1																														

3. <Shift>-Taste drücken und gleichzeitig nochmals zweites Bit selektieren, - dann wird automatisch der gesamte Zwischenblock selektiert.

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	
			5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	Group1	Signal1																														

Mit Hilfe der verschiedenen Zahlen („1111 2222...“) lassen sich Datenblöcke bilden, mit denen praktisch jedes übertragene Signal beschrieben werden kann. Die Zahlen, aus denen die Datenblöcke gebildet werden, haben dabei die Bedeutung, dass der Block mit der höchsten Zahl („2222“) den Block angibt, der bei der Übertragung die höchstwertigsten Bits enthält. Der Block mit der kleinsten Zahl („1111“) enthält bei der Übertragung die niederwertigsten Bits.

Mit den zur Verfügung stehenden Zahlen (1...9) lässt sich folglich ein Signal mit bis zu 9 Bit-Blöcken beschreiben. Da die Darstellung der Bits genau der Form entspricht, in der ein Intel-Signal übertragen wird, ist die Blockbildung für die Darstellung von Signalen im Motorola-Format notwendig, sobald das Signal eine Länge von 8 Bit überschreitet.

**Beispiele für die Definition von verschiedenen Signalen:**

16 Bit Signal im Intel Format:

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0		
			6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	Group1	Signal1																															

Andere Beschreibung desselben Signals:

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0		
			6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	Group1	Signal1																															

12 Bit Signal im Intel Format mit Lücke:

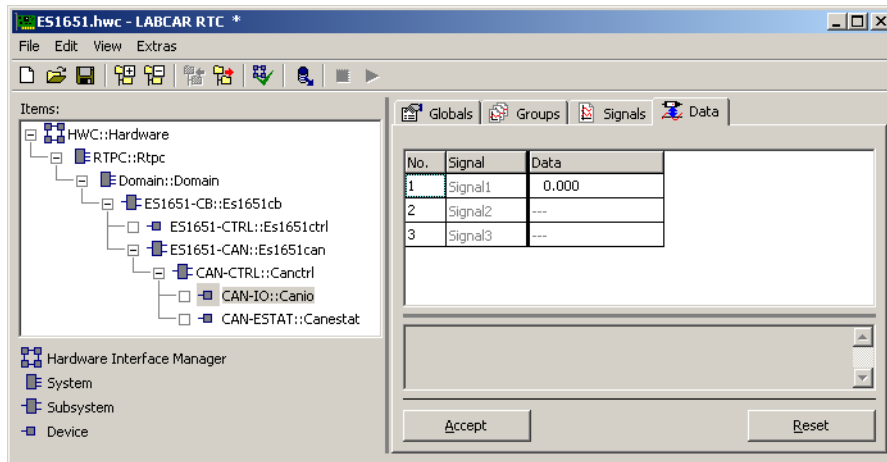
No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0
			6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4
1	Group1	Signal1													2	2	2												

16 Bit Signal im Motorola Format:

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0
			6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4
1	Group1	Signal1															1	1	1	1	1	1	1	1	1	1	1	1	1

16.4.6 Data (CAN-IO Device)

Abb. 16-18 zeigt die Registerkarte „Data“ eines CAN-IO Devices.

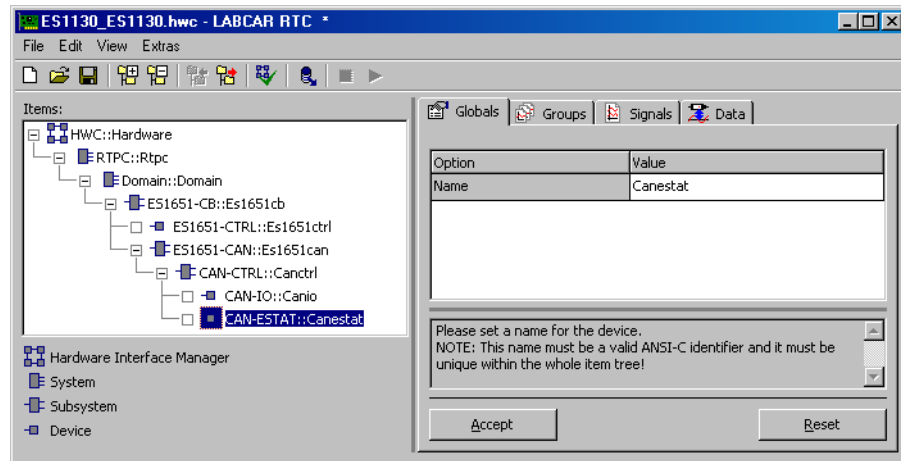


**Abb. 16-18** Die Registerkarte „Data“ des CAN-IO Devices

Man erkennt in der Registerkarte, dass das Signal „Signal1“ dazu verwendet wird, um eine CAN-Botschaft zu versenden. Im „Data“-Feld können im „online“ Modus die Nutzdaten belegt werden. Im „offline“ Modus sind die „Data“-Felder der Empfangsbotschaften „Signal2“ und „Signal3“ nicht aktiv, im „online“ Modus werden dort die empfangenen Nutzdaten der entsprechenden CAN-Botschaften dargestellt.

16.4.7 Globals (CAN-ESTAT Device)

Abb. 16-19 zeigt die Registerkarte „Globals“ eines CAN-ESTAT RTIO-Elements.



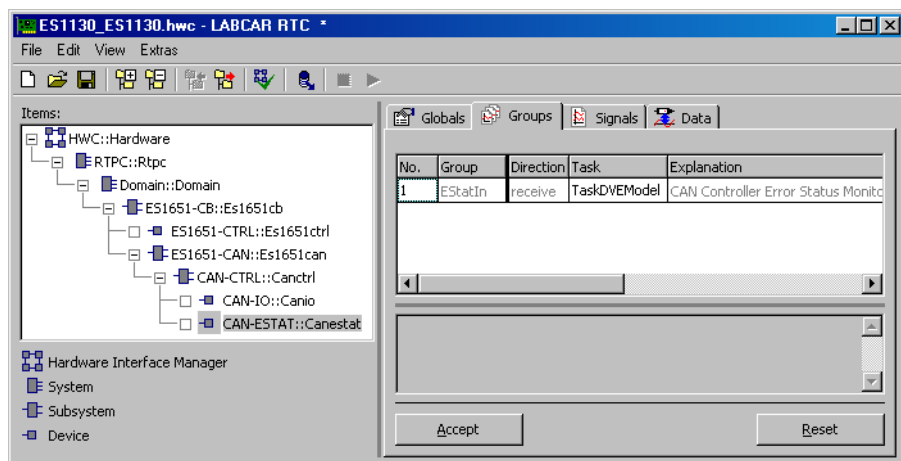
**Abb. 16-19** Die Registerkarte „Globals“ des CAN-ESTAT Device

16.4.8 Groups (CAN-ESTAT Device)

Das CAN-ESTAT Device besitzt eine Signalgruppe, die vom ES1651.1 Carrier Board zum Experimentaltarget übertragen wird (siehe Abb. 16-20). Die Signalgruppe „EStatIn“ transportiert Statusinformation und Fehlerzustände der einzelnen CAN-Controller zur RTIO.

Der Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erzeugenden bzw. zu überwachenden Synchronisationssignale. Sinnvolle Aktivierungsperioden sind durch die Aktualisierungsrate der Zustandsinformationen der Firmware auf den Trägerkarten auf 500 Millisekunden nach unten begrenzt.

Abb. 16-20 zeigt die Registerkarte „Groups“ eines CAN-ESTAT Devices.



**Abb. 16-20** Die Registerkarte „Groups“ des CAN-ESTAT Devices

#### 16.4.9 Die Signale der Signalgruppe „EStatIn“

Die Signalgruppe „EStatIn“ umfasst acht RTIO-Signale „EStatIn\_0“ bis „EStatIn\_7“, die vom Datentyp „bool“ (Signale 0..5) bzw. „uint8“ (Signale 6 und 7) sind. Für die Überwachung der beiden CAN-Controller ist keine weitere Konfiguration notwendig. Sobald ein Task des Echtzeit-Betriebssystems der Signalgruppe zugeordnet wurde, werden die Zustands- und Fehlerinformationen im „online“ Modus zur RTIO übertragen.

Tab. 16-12 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
EstatIn_0	bool	Nein	TXWARN: Reflektiert den Zustand der Sendeeinheit des betreffenden CAN-Controllers: 0: Fehlerzähler < 96 1: Fehlerzähler >= 96
EstatIn_1	bool	Nein	RXWARN: Reflektiert den Zustand der Empfangseinheit des betreffenden CAN-Controllers: 0: Fehlerzähler < 96 1: Fehlerzähler >= 96
EstatIn_2	bool	Nein	IDLE: Reflektiert Aktivitäten auf CAN-Bus: 0: Aktivitäten auf dem CAN-Bus 1: keine Aktivitäten auf dem CAN-Bus
EstatIn_3	bool	Nein	FCS: Zustand des CAN-Controllers im Fehlerfall: 0: CAN-Controller ist nicht im Zustand „error active“ 1: CAN-Controller ist im Zustand „error active“
EstatIn_4	bool	Nein	FCS: Zustand des CAN-Controllers im Fehlerfall: 0: CAN-Controller ist nicht im Zustand „error passive“ 1: CAN-Controller ist im Zustand „error passive“
EstatIn_5	bool	Nein	BOFFINT: Dieses Flag indiziert die Anforderung eines Interrupts beim Übergang des CAN-Controllers in den „bus off“ Zustand: 0: keine Interrupt-Anforderung 1: Interrupt-Anforderung

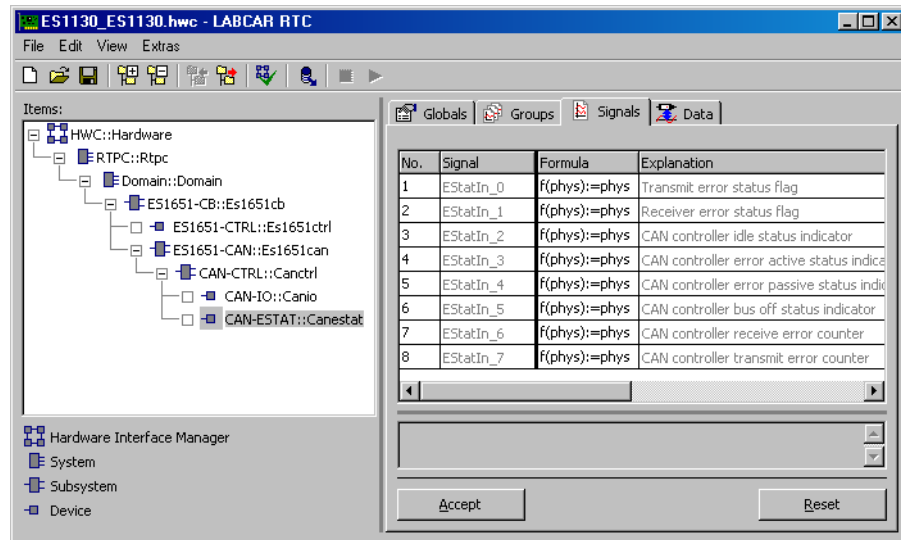
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
EstatIn_6	uint8	Nein	RXECTR: Fehlerzähler der Empfangseinheit des CAN-Controllers
EstatIn_7	uint8	Nein	TXECTR: Fehlerzähler der Sendeeinheit des CAN-Controllers

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-12** RTIO-Signale der Registerkarte „Signals“

16.4.10 Signals (CAN-ESTAT Device)

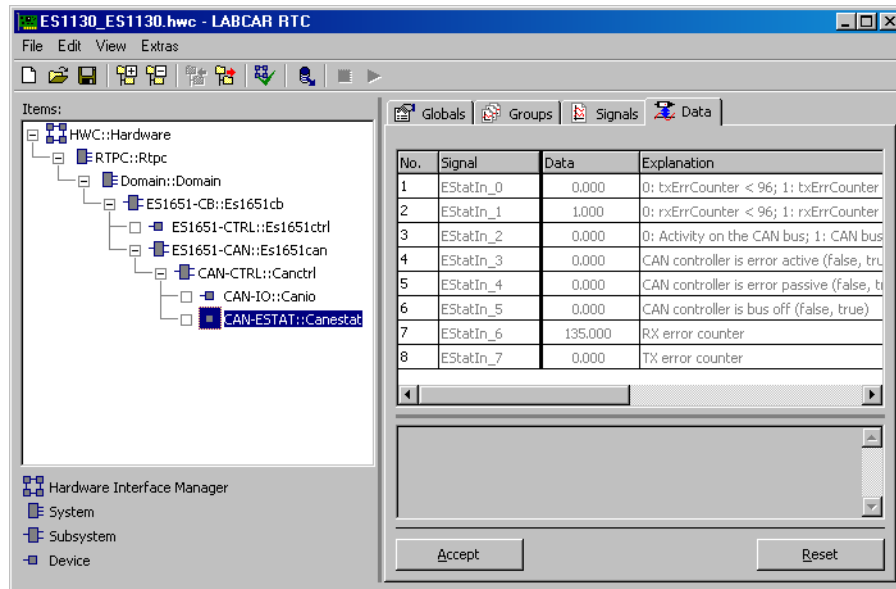
Abb. 16-21 zeigt die Registerkarte „Signals“ eines CAN-ESTAT Devices.



**Abb. 16-21** Die Registerkarte „Signals“ des CAN-ESTAT Devices

## 16.4.11 Data (CAN-ESTAT Device)

Abb. 16-22 zeigt die Registerkarte „Data“ eines CAN-ESTAT Devices.



**Abb. 16-22** Die Registerkarte „Data“ des CAN-ESTAT Devices

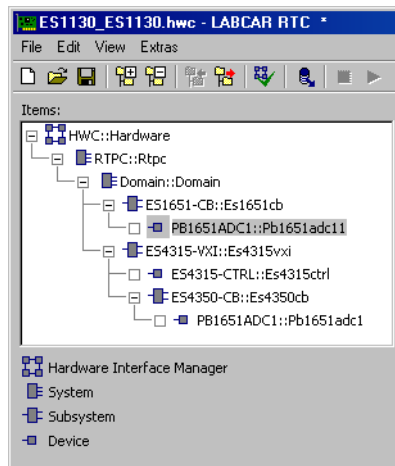
Obige Abbildung („online“ Modus) zeigt einen als Empfänger konfigurierten CAN-Controller im Fehlerfall. Es wurden 135 Empfangsfehler protokolliert und dementsprechend wurde das „RXWARN“-Flag auf „true“ gesetzt. Aktivitäten auf dem CAN-Bus wurden im Augenblick der Aufnahme keine registriert.



## 16.5 PB1651ADC1

Das PB1651ADC1-Modul wird zur Erfassung von analogen Eingangssignalen eingesetzt wie z.B. Motortemperatur, Fahrpedalstellung oder anderer analoger Ausgangsgrößen des Steuergerätes.

Ein PB1651ADC1-Modul wird im VMEbus-Baugruppenträger auf einem ES1651.1 Carrier Board oder im VXI-Baugruppenträger auf einem ES4350 Carrier Board verbaut. Die beschriebene Hardware-Anbindung spiegelt sich im RTIO-Baum wider (Abb. 16-23). Das PB1651ADC1 RTIO-Element eines PB1651ADC1-Moduls wird direkt dem ES1651-CB oder ES4350-CB RTIO-Element zugeordnet.



**Abb. 16-23** RTIO-Hardwarebeschreibung mit PB1651ADC1-Modul

Das PB1651ADC1-Modul besitzt 16 Eingangskanäle zur Erfassung von analogen Eingangssignalen. Die Konfiguration dieser Eingangskanäle erfolgt mit dem PB1651ADC1 RTIO-Element. Für die Messwerteerfassung stehen grundsätzlich zwei Betriebsmodi zur Verfügung. Der Modus „Analog“ liefert die Eingangsgrößen als Spannungswert im Intervall 0 bis 10 V, während der Modus „Comparator“ eine Vergleichsoperation bezüglich einer zu spezifizierenden Vergleichsspannung durchführt.

16.5.1 Globals (PB1651ADC1 Device)

In diesem Abschnitt werden die globalen Einstellungen des PB1651ADC1 Devices beschrieben.

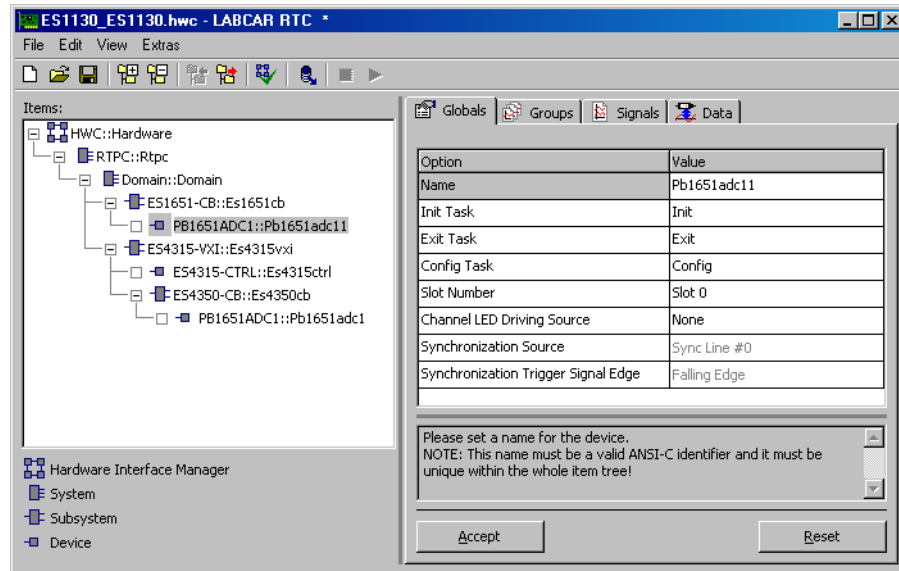


Abb. 16-24 Die Registerkarte „Globals“ des PB1651ADC1 Devices

Slot Number

In diesem Listenfeld ist die Nummer des Steckplatzes eines ES1651.1 Carrier Board bzw. eines ES4350 Carrier Boards anzugeben, in dem das PB165ADC1-Modul steckt. ES4350 Carrier Boards weisen sechs Modul-Steckplätze auf, deren Nummern auf der Frontplatte aufgedruckt sind. Die Nummerierung der vier Steckplätze eines ES1651.1 Carrier Board ist in Abb. 16-25 gezeigt.

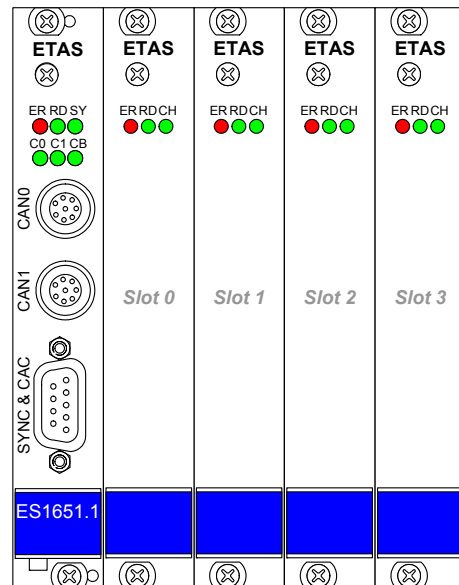


Abb. 16-25 Nummerierung der Modulsteckplätze eines ES1651.1 Carrier Board

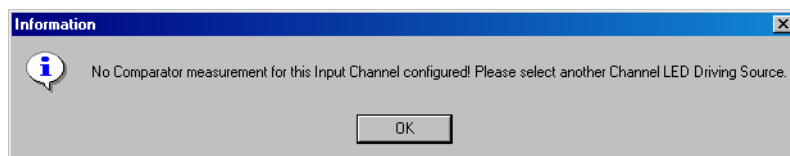
### *Channel LED Driving Source*

---

In diesem Optionsfeld kann die Ansteuerung der „CH“-LED auf dem PB1651ADC1-Modul festgelegt werden. Mit der Einstellung „RTIO“ kann die „CH“-LED über die RTIO bzw. das Simulationsmodell angesteuert werden.

Wird für eine Eingangsmessgröße der Betriebsmodus „Comparator“ gewählt, kann jeweils die „CH“-LED dazu verwendet werden, das Ergebnis der Vergleichsoperation sichtbar zu machen. Ist die Eingangsspannung dabei kleiner als die eingestellte Vergleichsspannung, so ist die „CH“-LED aus, ist die Eingangsspannung größer, so wird die LED aktiviert.

Diese Funktion ist aber nur dann aktiviert, wenn für den entsprechenden Eingangskanal die Vergleichsoperation „Comparator“ aktiviert wurde. Ist dies nicht der Fall, wird eine Fehlermeldung ausgegeben (siehe Abb. 16-26).



**Abb. 16-26** Fehlerhafte Konfiguration der „CH“-LED

### *Synchronization Source*

---

Diese Option dient zur Synchronisierung der analogen Eingangskanäle auf ein externes Ereignis. Das Listenfeld ist nur dann editierbar, wenn für einen Eingangskanal im Listenfeld „Synchronization Mode“ (in der Registerkarte „Signals“) die Option „Hardware Triggered“ ausgewählt wurde.

Hier ist die Leitung des Synchronisationsbusses einzustellen, deren aktive Flanken eine Aktualisierung der Spannungen an den Kanälen eines PB1651ADC1-Moduls anstossen. Der Synchronisationsbus eines ES1651.1 Carrier Board besitzt vier Leitungen, der eines ES4350 Carrier Boards sechs Leitungen. Das Listenfeld ist online editierbar.

### *Synchronization Trigger Signal Edge*

---

In diesem Listenfeld wird die aktive Flanke (steigend oder fallend) eingestellt, die eine Aktualisierung der Spannungen an den Kanälen eines PB1651ADC1-Moduls anstößt. Wie das Listenfeld „Synchronization Source“ ist dieses Listenfeld nur dann editierbar, wenn in einem Listenfeld „Synchronization Mode“ (in der Registerkarte „Signals“) die Option „Hardware Triggered“ eingestellt ist. Das Listenfeld ist online editierbar.

Tab. 16-13 fasst die Eigenschaften der Konfigurationsparameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wer tebereich
Slot Number	uint32	Nein	Steckplatz-Nummer des PB1651ADC1 Moduls auf der Trägerkarte ES1651: 0, 1, 2, oder 3 ES4350: 0, 1, ... 5
Channel LED Driving Source	sint8	Ja	Steuerquelle für „CH“-LED -2: „None“ (Keine Steuerquelle) -1: „RTIO“ ** 0: „Input Channel #0“ ... 15: „Input Channel #15“
Synchronization Source	uint8	Ja/Nein	Synchronisationsleitung: 0: Sync-Leitung 0 1: Sync-Leitung 1 2: Sync-Leitung 2 3: Sync-Leitung 3 4: Sync-Leitung 4 *** 5: Sync-Leitung 5 ***
Synchronization Trigger Signal Edge	bool	Ja/Nein	Synchronisations-Triggerflanke: 0: Fallende Flanke 1: Steigende Flanke

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet.

\*\* Wird die „CH“-LED über die RTIO angesteuert, ist die LED aktiv beim Senden einer „1“ und inaktiv bei „0“.

\*\*\* Sync-Leitung 4 und 5 nur beim ES4350 Carrier Board

**Tab. 16-13** PB1651ADC1 Device: Konfigurationsparameter der Registerkarte „Globals“

#### Versteckte Optionsfelder

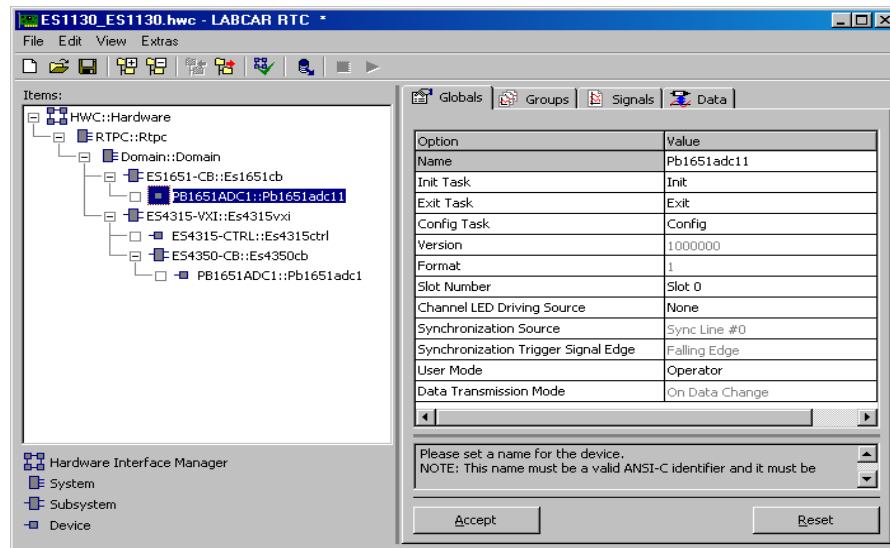
Wird mit der rechten Maustaste in die Registerkarte „Globals“ geklickt und im Kontextmenü **Show all Options** gewählt, werden weitere Optionsfelder sichtbar (Abb. 16-27). Diese Optionsfelder erlauben es dem ETAS-Servicepersonal, bei einer Fehlersuche zusätzliche Informationen über das PB1651ADC1-Modul zu erhalten.

#### Hinweis

*Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten vom Anwender nicht abgeändert werden.*

Im „Supervisor“ Modus werden in der Initialisierungsphase des RTIO-Treibers die Daten des nichtflüchtigen Speichers (wie z.B. Kalibrierungswerte des Moduls) ausgelesen und dem Anwender in LABCAR-OPERATOR „Hardware Info“-Feld sichtbar gemacht.

Die Registerkarte „Globals“ mit den versteckten Optionen eines PB1651ADC1 Devices ist in Abb. 16-27 dargestellt.



**Abb. 16-27** Die Registerkarte „Globals“ des PB1651ADC1 Devices mit allen Optionen sichtbar

#### *User Mode*

Stellt den RTIO-Anwendermodus ein. Anwender sollten in diesem Listenfeld die Option „Operator“ wählen. Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

#### *Data Transmission Mode*

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeitdaten vom Experimentaltarget zum PB1651ADC1-Modul übertragen werden.

Ist die Option „On Data Change“ aktiviert, werden die Daten einer Signalgruppe nur dann übertragen, wenn sich wenigstens ein Datum geändert hat. Ist die Option „Every Interval“ aktiviert, so werden die Echtzeitdaten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.

In Tab. 16-14 sind die Eigenschaften der Optionen zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
User Mode	uint8	Ja	Anwendermodus 0: „Operator“ ist Voreinstellung 1: „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche
Data Transmission Mode	uint8	Ja	Übertragungsmodus der Echtzeitdaten ** 0: „On Data Change“ 1: „Every Interval“ „On Data Change“ ist Voreinstellung

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

\*\* Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“

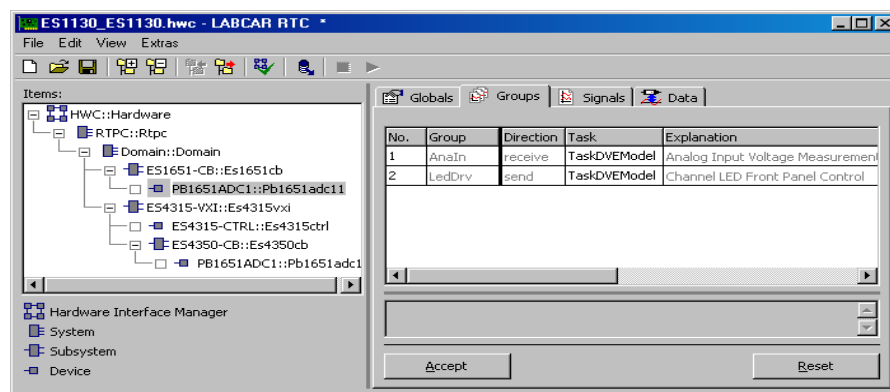
**Tab. 16-14** ES1651ADC1 Device: Konfigurationsparameter der Registerkarte „Globals“

16.5.2 Groups (PB1651ADC1 Device)

Das PB1651ADC1 Device besitzt zwei Signalgruppen, die vom Experimentaltarget über das Carrier Board zum Modul bzw. vom Modul über das Carrier Board zum Experimentaltarget übertragen werden. Die Signalgruppe „Analn“ transportiert 16-Bit breite analoge Messwerte der 16 Eingangskanäle vom Modul zur RTIO, während die Signalgruppe „LedDrv“ für die Ansteuerung der „CH“-LED zur Verfügung steht.

Jeder Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen - üblicherweise wird man eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu vermessenden analogen Eingangssignale bzw. der Ansteuerung der LED z.B. aus dem Simulationsmodell. Sinnvolle Aktivierungsperioden sind durch Laufzeiten der Firmware auf den Trägerkarten auf 1 ms nach unten begrenzt.

Abb. 16-28 zeigt die Registerkarte „Groups“ eines PB1651ADC1 Devices.



**Abb. 16-28** Die Registerkarte „Groups“ des PB1651ADC Devices

### Die RTIO-Signale der Signalgruppe „Analn“

Die Signalgruppe „Analn“ umfasst 16 RTIO-Signale „Analn\_0“ bis „Analn\_15“, die vom Datentyp „real32“ sind. Es stehen grundsätzlich zwei Messverfahren zur Verfügung.

Das Verfahren „Analog“ liefert analoge Spannungswerte im Intervall von 0,0 bis 10,0 Volt.

Das Messverfahren „Comparator“ realisiert einen Spannungsvergleich der Eingangsspannung mit einer vom Anwender zu spezifizierenden Vergleichsspannung. Ist die Eingangsspannung dabei kleiner als die Vergleichsspannung, wird eine „0“ ausgegeben, andernfalls eine „1“.

Das Ergebnis dieses Messverfahrens kann über die „CH“-LED auch sichtbar gemacht werden. Ist eine Eingangsgröße mit dem Verfahren „Comparator“ konfiguriert, so kann in der Registerkarte „Globals“ diese Größe für die Option „Channel LED Driving Source“ als Quelle dienen.

Tab. 16-15 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Analn_0 bis Analn_15	real32	Nein	Messwertefassung der beiden Messverfahren: 0 -10 V: Messverfahren „Analog“ 0 oder 1: Messverfahren „Comparator“ **

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet.

\*\* Ist die „CH“-LED Ansteuerung für eine Vergleichsoperation aktiviert, ist die LED aus, wenn die Eingangsspannung kleiner ist als die Vergleichsspannung, andernfalls ist die LED an.

### **Tab. 16-15** RTIO-Signale der Signalgruppe „Analn“

#### Die RTIO-Signale der Signalgruppe „LedDrv“

Die Signalgruppe „LedDrv“ umfasst ein RTIO-Signal „LedDrv“ vom Datentyp „bool“. Sie wird dazu verwendet, um die „CH“-LED an der Frontplatte des PB1651ADC1-Moduls zu steuern. „0“ bedeutet dabei „LED inaktiv“, „1“ bedeutet „LED aktiviert“.

Das Signal ist nur dann online editierbar, wenn (in der Registerkarte „Globals“) die Option „Channel LED Driving Source“ mit dem Wert „RTIO“ konfiguriert wird.

Tab. 16-16 fasst die Eigenschaften des RTIO-Signals zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
LedDrv	bool	Ja/Nein	Ansteuerung der „CH“-LED: 0: LED inaktiv 1: LED aktiv

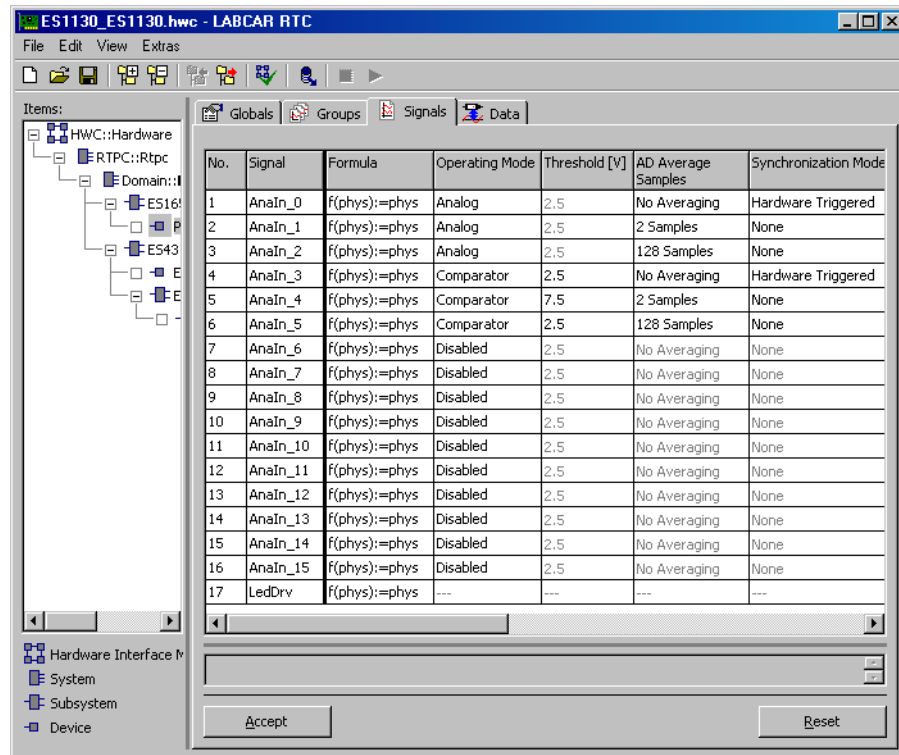
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

### **Tab. 16-16** RTIO-Signale der Signalgruppe „LedDrv“

### 16.5.3 Signals (PB1651ADC1 Device)

In der Registerkarte „Signals“ wird die Konfiguration der Eingangskanäle des PB1651ADC1-Moduls durchgeführt. Alle Parameter der Signalgruppe „AnaIn“ sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Abb. 16-29 zeigt die Registerkarte „Signals“ eines PB1651ADC1 Devices.



**Abb. 16-29** Die Registerkarte „Signals“ des PB1651ADC1 Devices

#### Operating Mode

Jeder der 16 Eingangskanäle eines PB1651ADC1-Moduls kann in drei unterschiedlichen Modi betrieben werden, die im Auswahlfeld „Operating Mode“ für jeden Kanal getrennt eingestellt werden:

- **Disabled**  
Der Eingangskanal ist deaktiviert.
- **Analog**  
Der Kanal ist für analoge Messwerterfassung konfiguriert. Das zugehörige Signal  $x$  ( $x = 0, 1, \dots, 15$ ) gibt die gemessene Spannung in Volt an.
- **Comparator**  
Der Kanal ist als Komparator konfiguriert. Der zugehörige Signalwert  $x$  ( $x = 0, 1, \dots, 15$ ) ist gleich 0, wenn die Eingangsspannung des Kanals kleiner als eine vom Anwender einstellbare Spannungsschwelle ist, andernfalls ist der Signalwert gleich 1.



### *Threshold [V]*

---

Die Einstellung im Feld „Threshold [V]“ ist nur für den Komparatormodus von Bedeutung, d.h. sowohl „online“ als auch „offline“ editierbar. Hier wird die Schwellspannung für den Komparator festgelegt.

### *AD Average Samples*

---

Hier wird festgelegt, aus wie vielen Abtastwerten ein analoger Spannungswert durch Mittelwertbildung berechnet wird. Durch die Mittelwertbildung kann eine Stör- oder Rauschunterdrückung erzielt werden. Die Einstellung im Feld „AD Average Samples“ ist für alle Betriebsmodi bis auf „Disabled“ von Bedeutung.

Pro Kanal stehen acht Konfigurationsmöglichkeiten zur Verfügung:

- No Averaging (Mittelwertbildung ist deaktiviert)
- 2 Samples
- 4 Samples
- 8 Samples
- 16 Samples
- 32 Samples
- 64 Samples
- 128 Samples

### *Synchronization Mode*

---

Dieses Listenfeld legt fest, wann vom PB1651ADC1-Modul analoge Eingangssignale von der Hardware erfasst und über das Carrier Board zum RTIO-Treiber bzw. zur RTIO transferiert werden.

Wird in diesem Listenfeld die Option „None“ eingestellt, so wird die Erfassung von Eingangssignalen auf dem PB1651ADC1-Modul für den betreffenden Kanal sofort aktiviert.

Bei der Option „Hardware Triggered“ erfolgt eine Aktualisierung nur dann, wenn auf der im Listenfeld „Synchronization Source“ eingestellten Leitung des Synchronisationsbusses eines ES1651.1 Carrier Board oder eines ES4350 Carrier Boards eine aktive Flanke erfasst wurde. Auf diese Weise ist eine Synchronisierung der analogen Eingabe auf ein externes Ereignis möglich.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar, wenn der „Operating Mode“ mit den Optionen „Analog“ oder „Comparator“ konfiguriert wurde.

In Tab. 16-17 sind die Eigenschaften der RTIO-Signale zusammengefasst.

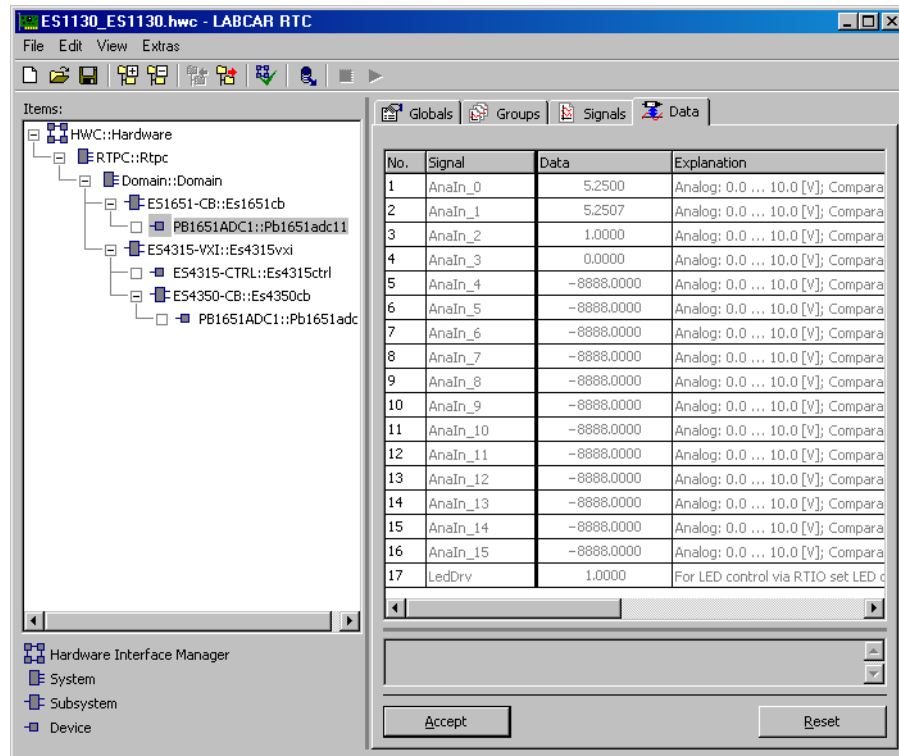
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Operating Mode	uint8	Ja	Messverfahren für die Eingangskanäle: 0: „Disabled“: Kanal deaktiviert 1: „Analog“: Analoge Spannungserfassung 2: „Comparator“: Vergleichsoperation
Threshold	real32	Ja/Nein	Definition der Vergleichsspannungen im Intervall von 0 bis 10 Volt
AD Average Samples	uint8	Ja/ Nein	Einstellungen für Mittelwertbildung eines jeden Eingangskanals: 0: „No Averaging“: Keine Mittelwertbildung 1: „2 Samples“ 2: „4 Samples“ 3: „8 Samples“ 4: „16 Samples“ 5: „32 Samples“ 6: „64 Samples“ 7: „128 Samples“
Synchronization Mode	uint8	Ja/ Nein	Auswahl der Triggerung für die Messwerterfassung: 0: „Hardware Triggered“: Die Messwerterfassung wird (mittels einer Sync-Leitung) auf ein externes Ereignis getriggert 1: „None“: Es erfolgt keine externe Triggerung

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-17** RTIO-Signale der Registerkarte „Signals“

## 16.5.4 Data (PB1651ADC1 Device)

Abb. 16-30 zeigt die Registerkarte „Data“ eines PB1651ADC1 Devices.



**Abb. 16-30** Die Registerkarte „Data“ des PB1651ADC1 Devices

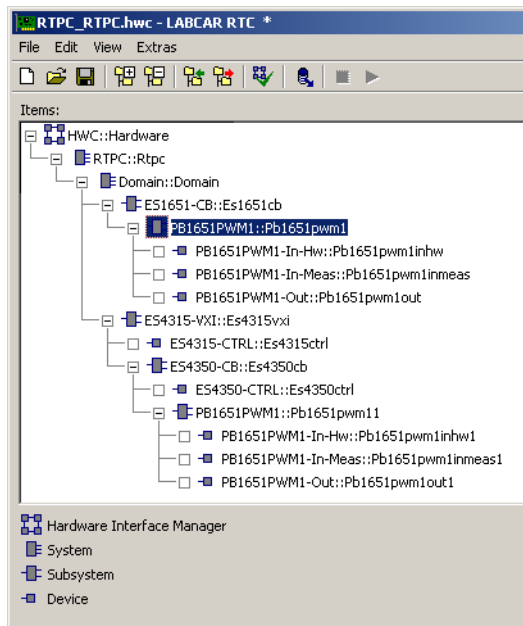
Die Abbildung zeigt die Registerkarte „Data“ im „online“ Modus. Für die ersten vier Eingangskanäle wurden Messmodi ausgewählt, die restlichen Kanäle sind deaktiviert. Eingangskanal 1 und 2 befinden sich im Messmodus „Analog“, d.h. es werden analoge Eingangsspannungen im Intervall von 0 V bis 10 V vermessen und dargestellt. Die beiden nächsten Kanäle werden im Messmodus „Comparator“ betrieben, d.h. es wird ein Spannungsvergleich mit einer eingestellten Vergleichsspannung durchgeführt. Der Eingangskanal 3 zeigt eine „1“, d.h. die vermessene Eingangsspannung ist größer als die spezifizierte Vergleichsspannung. Das Messergebnis von Kanal 4 ist genau umgekehrt, das Eingangssignal ist kleiner.

Das RTIO-Signal 17 ist für die „CH“-LED Ansteuerung über das Simulationsmodell bzw. einer ASCET-SD-Message konfiguriert. Momentan ist eine „1“ sichtbar, d.h. die „CH“-LED an der Frontplatte des PB1651ADC1-Moduls ist an.

## 16.6 PB1651PWM1

Das PB1651PWM1-Modul dient der Ausgabe und Erfassung von pulsweitenmodulierten Signalen. Typischer Anwendungsfall des Moduls in LABCAR-Projekten ist die Nachbildung von Fahrzeugsensoren und die Ansteuerung von Aktoren.

Im RTIO-Editor wird das PB1651PWM1 I/O-Modul durch Auswahl des PB1651PWM1-RTIO-Elements eingebunden. Dieses Element kann entweder einem ES1651.1 Carrier Board (VMEbus) oder einem ES4350 Carrier Board (VXIbus) zugeordnet werden. Abb. 16-31 zeigt die Anbindung von PB1651PWM1-Modulen an beide Typen von Trägerkarten.



**Abb. 16-31** RTIO-Hardwarebeschreibung mit eingebundenen PB1651PWM1-Modulen

Das PB1651PWM1 I/O-Modul bietet 24 Eingangskanäle zur Erfassung von pulsweitenmodulierten Signalen. Die Konfiguration dieser Eingangskanäle erfolgt mit dem PB1651PWM1-In-Hw Device.

Neben diesem Element können dem PB1651PWM1-RTIO-Element bis zu vier PB1651PWM1-In-Meas Devices und ein PB1651PWM1-Out Device zugeordnet werden.

Mit jedem der PB1651PWM1-In-Meas Device lassen sich bis zu 24 Messungen (Frequenz, Pulsweite, Tastverhältnis usw. auf den Signalen der Eingangskanäle spezifizieren. Insgesamt bieten RTIO-Treiber und Firmware der PB1651PWM1-Moduls damit 96 mögliche Messungen an.

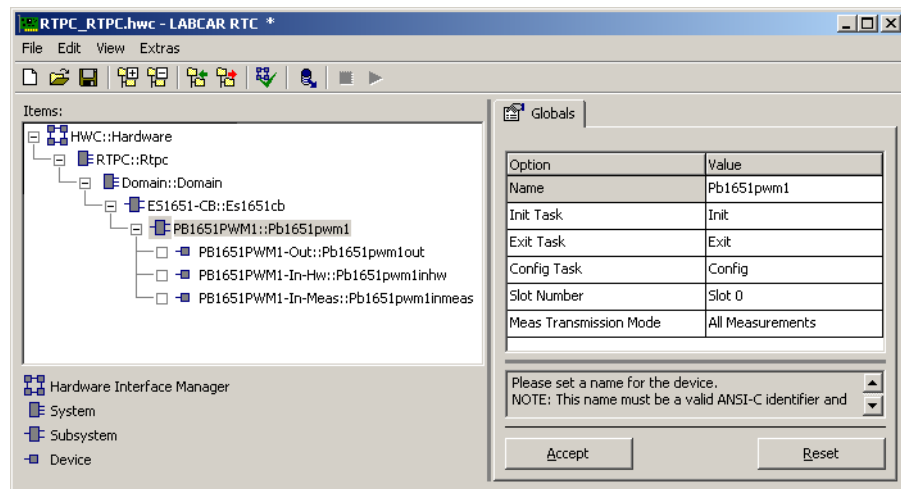
Die Konfiguration und Ansteuerung der 16 Kanäle eines PB1651PWM1-Moduls zur Ausgabe pulswidenmodulierter Signale erfolgt mit dem PB1651PWM1-Out Device.

**Hinweis**

*Bevor ein „PB1651PWM1-In-Meas“-RTIO-Element eingefügt werden kann, muss zunächst ein „PB1651PWM1-In-Hw“-Element zur Konfiguration der Eingangskanäle eingefügt werden!*

16.6.1 Globals (PB1651PWM1 Subsystem)

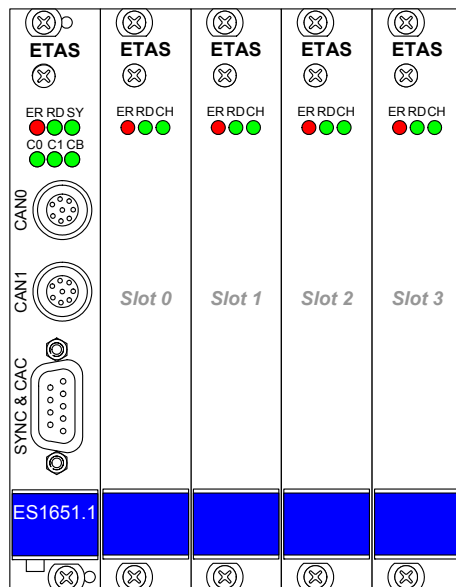
In diesem Abschnitt werden die globalen Einstellungen des PB1651PWM1 Subsystems beschrieben.



**Abb. 16-32** Die Registerkarte „Globals“ des PB1651PWM1 Subsystems

### Slot Number

In diesem Listenfeld ist die Nummer des Steckplatzes eines ES1651.1 Carrier Board bzw. eines ES4350 Carrier Boards einzustellen, in dem das PB1651PWM1-Modul steckt. ES4350 Carrier Boards weisen sechs Modul-Steckplätze auf, deren Nummern auf der Frontplatte aufgedruckt sind. Die Nummerierung der vier Steckplätze eines ES1651.1 Carrier Board ist in Abb. 16-33 gezeigt.



**Abb. 16-33** Nummerierung der Modul-Steckplätze eines ES1651.1 Carrier Board

### Meas Transmission Mode

Mit Hilfe dieser Option wird direkt der Datentransfer zwischen I/O-Modul und dem Experimentaltarget festgelegt. Für die Signalvermessung stehen vier unterschiedliche Konfigurationen zur Verfügung.

- All Measurements  
Alle Messdaten werden zum Experimentaltarget übertragen. Dadurch können in der RTIO alle Messverfahren selektiert werden.
- Timestamp Measurements  
Für dieses Messverfahren werden nur die Zeitstempel zum Experimentaltarget übertragen. Dadurch können in der RTIO alle Messverfahren exklusiv der „additiven“ Verfahren selektiert werden.
- Additive Measurements  
In diesem Modus können nur die „additiven“ Messverfahren ausgewählt werden. Zudem steht für diese Konfiguration keine Timeout-Erkennung bereit.

- Level Measurements

In diesem Modus werden nur die Signalpegel der Eingangskanäle zum Experimentaltarget transferiert. Zudem steht für diese Konfiguration keine Timeout-Erkennung bereit.

#### **Hinweis**

*Die Konfiguration des Datentransfers kann nur vor der Codegenerierung erfolgen, d.h. diese Option ist nicht online editierbar.*

#### **Hinweis**

*Für die Konfigurationen „Additive Measurements“ und „Level Measurements“ steht die "Timeout" Detektierung zur Laufzeit nicht zur Verfügung.*

Tab. 16-18 fasst die Eigenschaften der Konfigurationsparameter der Registerkarte „Globals“ zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Slot Number	uint32	Nein	Steckplatz-Nummer des PB1651PWM1 Moduls auf der Trägerkarte ES1651: 0, 1, 2, oder 3 ES4350: 0, 1, ... 5
Meas Transmission Mode	uint8	Nein	Konfiguration des Datentransfers bei der Signalvermessung: 0: Alle Messdaten werden übertragen 1: Nur die Zeitstempel werden übertragen 2: Nur die Additiven Daten werden übertragen 3: Nur die Signalpegel werden übertragen

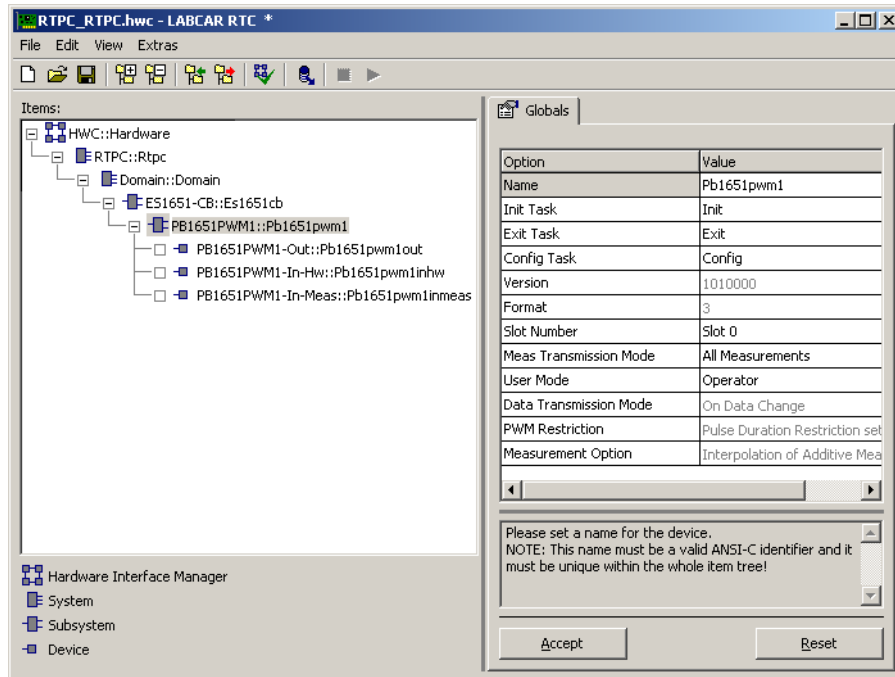
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-18** ES1651PWM1 Subsystem: Konfigurationsparameter der Registerkarte „Globals“

#### *Versteckte Optionsfelder*

Wird mit der rechten Maustaste auf die Registerkarte „Globals“ geklickt und im Kontextmenü **Show all Options** gewählt, werden weitere Optionsfelder sichtbar (siehe Abb. 16-34).

Diese Optionsfelder erlauben es dem ETAS-Servicepersonal, bei einer Fehlersuche zusätzliche Informationen über das PB1651PWM1-Modul zu erhalten. Sie sind nicht für den Anwender gedacht und die Voreinstellungen sollten vom Anwender nicht abgeändert werden.



**Abb. 16-34** Die Registerkarte „Globals“ des PB1651PWM1 Subsystems mit allen Optionen sichtbar

### User Mode

Mit dieser Option wird der RTIO-Anwendermodus eingestellt. Anwender sollten in diesem Listenfeld die Option „Operator“ wählen. Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für das ETAS-Servicepersonal gedacht. Unter anderem werden bei aktiviertem „Supervisor“-Modus die Inhalte des nicht flüchtigen Datenspeichers des PB1651PWM1-Moduls während der Initialisierung in das „Target Debug“-Fenster ausgegeben. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

### Data Transmission Mode

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeitdaten vom Experimentaltarget zum PB1651PWM1-Modul übertragen werden. Ist die Option „On Data Change“ aktiviert, werden die Daten einer Signalgruppe nur dann übertragen, wenn sich wenigstens ein Datum geändert hat.

Ist die Option „Every Interval“ aktiviert, so werden die Echtzeitdaten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.



In Tab. 16-19 sind die Eigenschaften der versteckten RTIO-Parameter der Registerkarte „Globals“ zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
User Mode	uint	Ja	Anwendermodus 0: Operator (Voreinstellung) 1: Supervisor (aktiviert zusätzliche Optionen zur Fehlersuche)
Data Transmission Mode	uint	Ja	Übertragungsmodus 0: On Data Change (Voreinstellung) 1: Every Interval Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

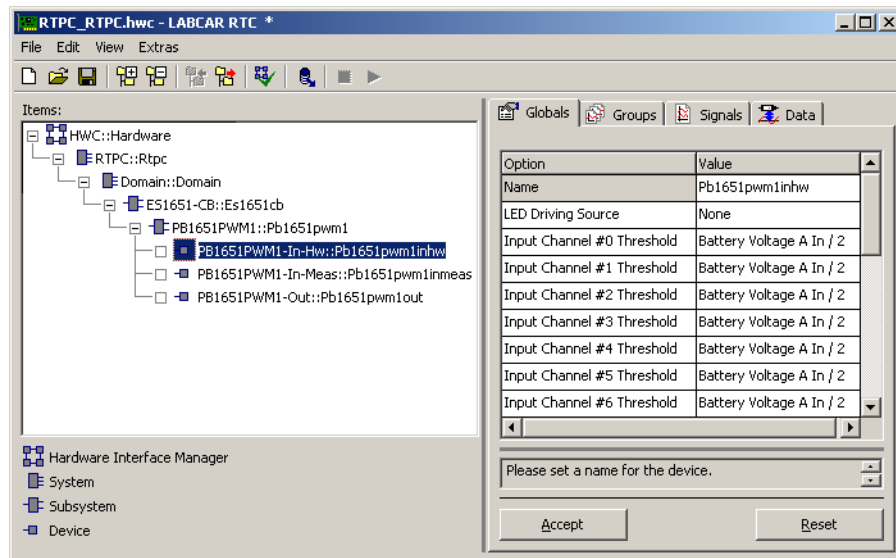
**Tab. 16-19** ES1651PWM1 Subsystem: Konfigurationsparameter der Registerkarte „Globals“

## 16.6.2 Das PB1651PWM1-In-Hw Device

Das PB1651PWM1-In-Hw Device dient zur Konfiguration der 24 Eingangskanäle des PWM-Moduls und zur Konfiguration und Ansteuerung der „CH“-LED in der Frontplatte des Moduls.

## 16.6.3 Globals (PB1651PWM1-In-Hw Device)

Abb. 16-35 zeigt die Registerkarte „Globals“ des PB1651PWM1-In-Hw Device.



**Abb. 16-35** Die Registerkarte „Globals“ des PB1651PWM1-In-Hw Device

### LED Driving Source

Mit diesem Listenfeld wird die Steuerquelle für die „CH“-LED in der Frontplatte des PB1651PWM1-Moduls definiert. Als Steuerquelle kann einer der 24 Eingangskanäle oder „RTIO“ eingestellt werden. Wird ein Eingangskanal als Steuerquelle eingestellt, wird die LED eingeschaltet, wenn der Kanal einen High-Pegel aufweist und ausgeschaltet, wenn er einen Low-Pegel aufweist.

Wird „RTIO“ als Steuerquelle eingestellt, so kann die LED gesteuert vom Simulationsmodell ein- bzw. ausgeschaltet werden. Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

### Input Channel #x Threshold ( $x = 0, 1, \dots, 23$ )

Die an den Eingängen des Moduls anliegenden PWM-Signale werden durch Komparatoren in den Eingangsstufen der Kanäle auf TTL-Pegel umgesetzt. Mit diesen Listenfeldern werden die Komparator-Schwelspannungen der Eingangskanäle eingestellt. Zur Auswahl stehen zwei Spannungen „Battery Voltage A In/2“ und „Battery Voltage B In/2“.

Die Spannungen „Battery Voltage A In“ und „Battery Voltage B In“ werden über den Steckverbinder in der Frontplatte des PB1651PWM1-Moduls eingespeist. Bevor sie auf die Komparatoren geführt werden, werden sie durch Spannungsteiler-Schaltungen um den Faktor 2 geteilt.

Die Listenfelder sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Die Eigenschaften der Konfigurationsparameter der Registerkarte „Globals“ sind in Tab. 16-20 zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
LED Driving Source	sint8	Ja	Steuerquelle für „CH“ LED -2: „None“ (keine Steuerquelle) -1: „RTIO“ 0: „Input Channel #0“ ... 23: „Input Channel #23“
Input Channel #x Threshold (x = 0, 1, ... 23)	uint8	Ja	Schwellspannung für Eingangskomparator 0: „Battery Voltage A In/2“ 1: „Battery Voltage B In/2“

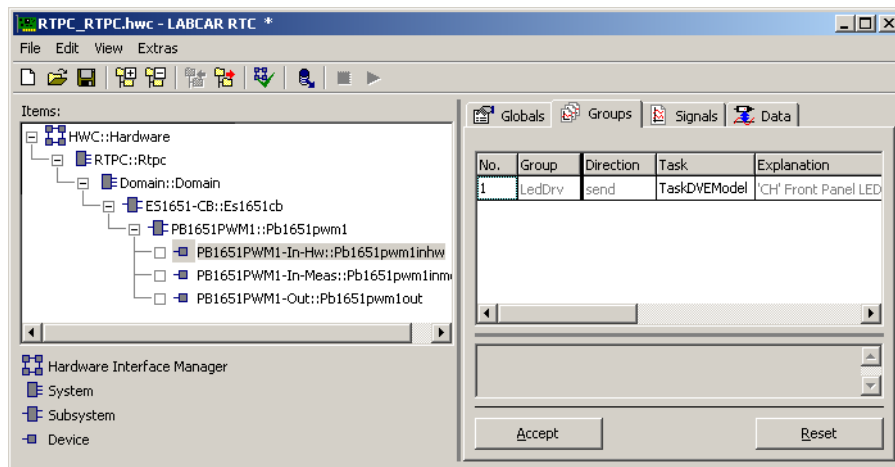
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-20** PB1651PWM1-In-Hw Device: Konfigurationsparameter der Registerkarte „Globals“

16.6.4 Groups (PB1651PWM1-In-Hw Device)

Das PB1651PWM1-In-Hw Device besitzt eine Signalgruppe, die vom Experimentaltarget zum PB1651PWM1-Modul übertragen wird (siehe Abb. 16-36). Sie transportiert die An-/Aus-Information für die „CH“-LED in der Frontplatte des Moduls. Diese Information wird von der Hardware allerdings nur dann ausgewertet, falls im Optionsfeld „LED Driving Source“ der Registerkarte „Globals“ die Einstellung „RTIO“ eingestellt ist.

Der Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen - üblicherweise wird man eine Task mit periodischer Aktivierung und einer Aktivierungsperiode größer als 100 Millisekunden wählen.



**Abb. 16-36** Die Registerkarte „Groups“ des PB1651PWM1-In-Hw Device  
*Die RTIO-Signale der Signalgruppe „LedDrv“*

Die Signalgruppe umfasst ein RTIO-Signal mit der Bezeichnung „LedDrv“, das vom Datentyp „bool“ ist. Es definiert den Schaltzustand der „CH“-LED. Der Wert 0 schaltet die LED aus; ein Wert ungleich 0 schaltet die LED ein.

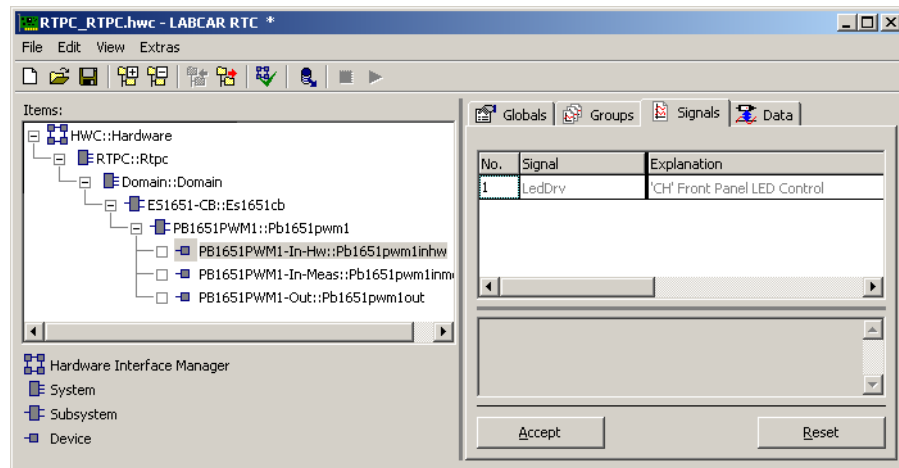
Tab. 16-21 fasst die Eigenschaften des RTIO-Signals zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
LedDrv	bool	Steuersignal für „CH“ LED 0: LED aus ≠ 0: LED ein
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 16-21** PB1651PWM1-In-Hw Device: Die RTIO-Signale der Signalgruppe „LedDrv“

### 16.6.5 Signals (PB1651PWM1-In-Hw Device)

Das PB1651PWM1-In-Hw Device besitzt keine signalspezifischen Optionen. Daher sind in der Registerkarte „Signals“ (siehe Abb. 16-37) keine Einstellungen vorzunehmen.



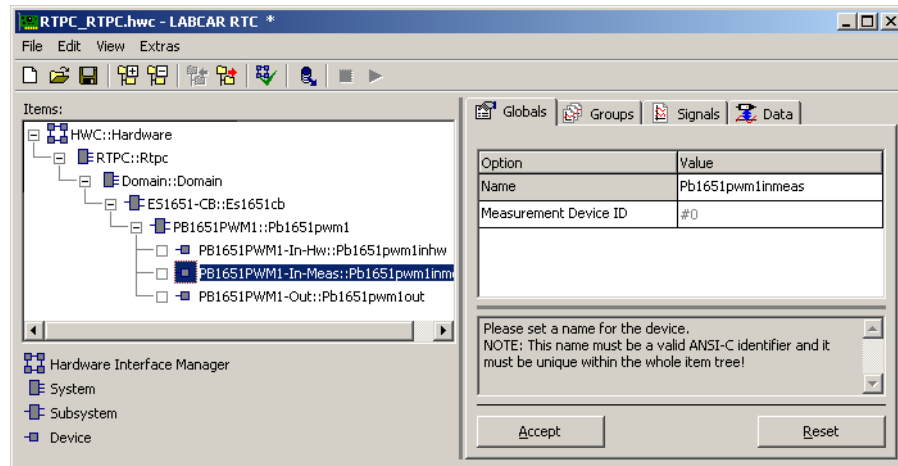
**Abb. 16-37** Die Registerkarte „Signals“ des PB1651PWM1-In-Hw Device

### 16.6.6 Das PB1651PWM1-In-Meas Device

Das PB1651PWM1-In-Meas Device dient zur Definition von Messungen auf den 24 Eingangskanälen des Moduls. Pro PB1651PWM1-Modul werden bis zu vier PB1651PWM1-In-Meas Devices unterstützt. Mit jedem Element lassen 24 Messungen definieren, so dass insgesamt 96 Messungen pro Modul möglich sind.

16.6.7 Globals (PB1651PWM1-In-Meas Device)

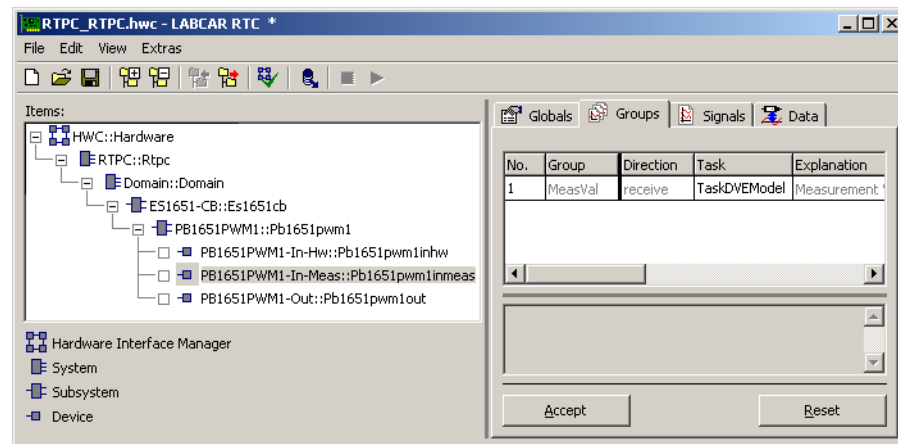
In der Registerkarte „Globals“ (siehe Abb. 16-38) sind keine Einstellungen vorzunehmen. Im Optionsfeld „Measurement Device ID“ wird dem PB1651PWM1-In-Meas Device vom System eine Zahl zwischen 0 und 3 zur Identifizierung zugewiesen.



**Abb. 16-38** Die Registerkarte „Globals“ des PB1651PWM1-In-Meas Device

16.6.8 Groups (PB1651PWM1-In-Meas Device)

Das PB1651PWM1-In-Meas Device besitzt eine Signalgruppe (siehe Abb. 16-39), die vom PB1651PWM1-Modul zum Experimentaltarget übertragen wird. Sie transportiert sämtliche Messdaten wie die eigentlichen Messwerte, Update- und Timeout-Informationen.



**Abb. 16-39** Die Registerkarte „Groups“ des PB1651PWM1-In-Meas Device

Der Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen - üblicherweise wird man eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik und der Periodendauer der zu vermessenden Signale.

### 16.6.9 Die RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 49 RTIO-Signale. Das Signal „TriggerBitField\_0“ ist als Bitfeld zu interpretieren - es umfasst 24 Bit. In diesem Bitfeld sind Aktualisierungsindikatoren für die 24 Messungen codiert, d.h. es zeigt an, welche Messwerte bei der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der Messwert der zugehörigen Messung neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeout oder aufgrund einer regulären Messwertberechnung erfolgte - in beiden Fällen wird das Indikator-Bit des Messkanals gesetzt.

Da die Messwernerfassung zwischen der Trägerkarte und dem PWM-Modul asynchron zur Taskzykluszeit der Messwernerfassung auf dem Modul verläuft, entsprechen die RTIO-Signale „MeasCatch\_0“ und „MeasCatch\_1“ der Zykluszeit der Werterfassung auf dem Modul.

Die RTIO-Signale „MeasVal\_0“ bis „MeasVal\_23“ enthalten die Messwerte der 24 Messungen. Es handelt sich dabei um regulär ermittelte Messwerte oder - im Falle eines Timeouts - um den für diesen Fall vorgesehenen Timeout-Wert. Falls eine Messung nicht verwendet wird, wird dem zugehörigen Messwert -8888.0 zugewiesen. Die physikalische Einheit der Messwerte richtet sich nach dem Messverfahren:

- Zeitmessungen (additive) Pulsdauer-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Frequenzmessungen erfolgen in Hertz
- alle anderen Messungen (Tastverhältnisse, Pulszählung, Pegelmessungen) sind dimensionslos

Die RTIO-Signale „Tout\_0“ bis „Tout\_23“ enthalten das Ergebnis der Timeout-Überwachung für die jeweilige Messung.

#### **Hinweis**

*Bei der Belegung der Messkanäle ist zu beachten, dass die Messwerte der 24 Eingangskanäle in zwei Blöcken vom PWM-Modul zur Trägerkarte transferiert werden (1. Block: Kanal 0-11; 2. Block: Kanal 12-23).*

*Ist die verwendete Kanalanzahl kleiner als 12, so ist aus Performancegründen darauf zu achten, dass nur Kanäle ausgewählt werden, die im selben Block liegen. Dadurch wird sichergestellt, dass keine unnötigen Daten übertragen und keine Ressourcen verschwendet werden.*

Tab. 16-22 fasst die Eigenschaften des RTIO-Signals zusammen.

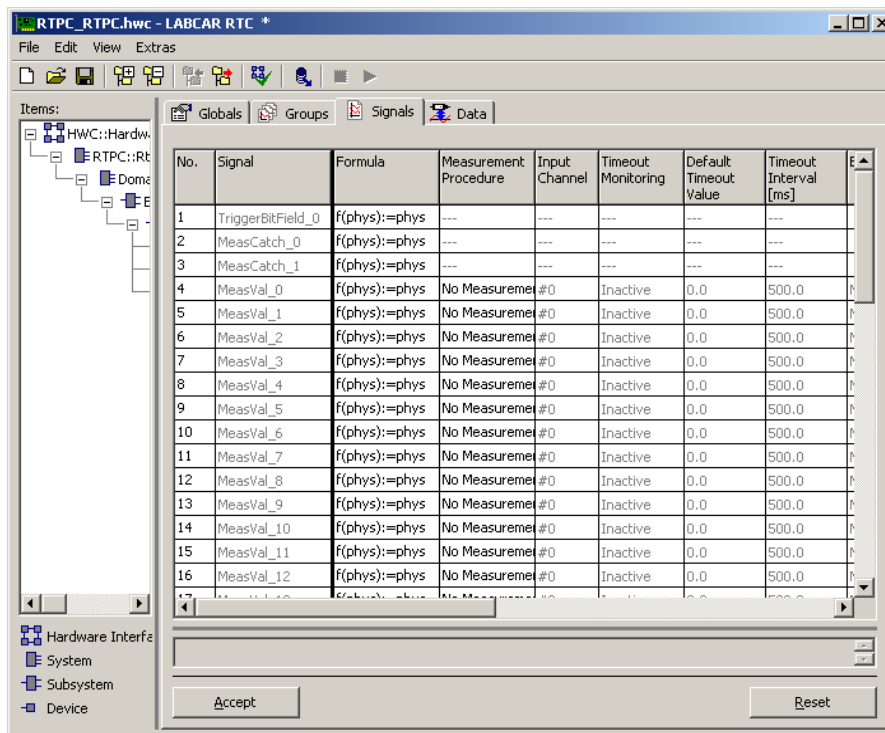
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
TriggerBitField	uint32	Bitfeld mit Aktualisierungsindikatoren für die 24 Messungen Messung 0: LSB (Least Significant Bit) Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasCatch_0 ... MeasCatch_1	real64	Zykluszeit der Messwernerfassung auf dem PWM Modul Falls die Messung nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in Mikrosekunden
MeasVal_0 ... MeasVal_23	real64	Messwert Falls die Messung nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in Mikrosekunden - Frequenzmessungen erfolgen in Hertz
Tout_0 ... Tout_23	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout aufgetreten 1: Timeout aufgetreten 2: Timeout-Überwachung ist inaktiv

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-22** PB1651PWM1-In-Meas Device: Die RTIO-Signale der Signalgruppe „MeasVal“

## 16.6.10 Signals (PB1651PWM1-In-Meas Device)

In der Registerkarte „Signals“ wird die Definition der 24 Messungen eines PB1651PWM1-In-Meas Devices durchgeführt. Abb. 16-40 zeigt die Optionsfelder der Registerkarte „Signals“. Alle Optionsfelder sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 16-40** Die Registerkarte „Signals“ des PB1651PWM1-In-Meas Device  
*Measurement Procedure*

In diesem Listenfeld wird das Messverfahren eingestellt. Im Abschnitt „Die Messverfahren“ auf Seite 447 finden Sie eine detaillierte Beschreibung und Konfigurationsrichtlinien zu den einzelnen Verfahren.

In dem Listenfeld bedeutet die Notation --/-- eine Auswertung an steigenden Flanken, die Notation --\-- hingegen eine Auswertung an fallenden Flanken.

#### **Hinweis**

*Zur Vermeidung unnötiger Rechenzeiten sind nicht benötigte Messungen zu deaktivieren, d.h. im Listenfeld ist die Option „No Measurement“ einzustellen.*

#### *Input Channel*

In diesem Listenfeld wird der Eingangskanal des Moduls eingestellt, auf dem die Messung durchgeführt werden soll. Das Listenfeld ist nur dann editierbar, wenn für die Messung ein Messverfahren eingestellt wurde.



### *Timeout Monitoring*

---

Definition der Timeout-Überwachung für die zugehörige Messung. Die folgenden Einstellungen sind möglich:

- „Inactive“: Keine Timeout-Überwachung.
- „Intvl Predef“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl InpDep“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.

Das Listenfeld ist nur dann editierbar, wenn für die Messung ein Messverfahren eingestellt wurde.

### *Default Timeout Value*

---

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls der „Intvl Predef“ Timeout-Überwachungsmodus eingestellt ist.

Das Eingabefeld ist nur dann editierbar, wenn für die Messung ein Messverfahren und der Modus „Intvl Predef“ zur Timeout-Überwachung eingestellt wurde.

### *Timeout Interval [ms]*

---

Zeitabstand, in dem die Messung auf einen Timeout geprüft wird. Das Intervall wird in Millisekunden angegeben.

Das Eingabefeld ist nur dann editierbar, wenn für die Messung ein Messverfahren und einer der beiden Modi „Intvl Predef“ oder „Intvl InpDep“ zur Timeout-Überwachung eingestellt wurde.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Measurement Procedure	sint32	Ja	Definition des Messverfahrens. „Messverfahren“ Parameterwert „No Measurement“ -1 „High Time [µs]“ 0 „Low Time [µs]“ 1 „Additive High Time [µs]“ 2 „Additive Low Time [µs]“ 3 „Number of Low Pulses“ 18 „Number of High Pulses“ 19 „Number of Rising Edges“ 20 „Number of Falling Edges“ 21 „Cycle Time --/-- [µs]“ 4 „Cycle Time --\-- [µs]“ 5 „Frequency --/-- [Hz]“ 6 „Frequency --\-- [Hz]“ 7 „Duty Factor L/H --/--“ 8 „Duty Factor L/H --\--“ 9 „Duty Factor H/L --/--“ 10 „Duty Factor H/L --\--“ 11 „Duty Cycle L/(L+H) --/--“ 12 „Duty Cycle L/(L+H) --\--“ 13 „Duty Cycle H/(L+H) --/--“ 14 „Duty Cycle H/(L+H) --\--“ 15 „Level (Active High)“ 16 „Level (Active Low)“ 17
Input Channel	uint8	Ja	Eingangskanal auf dem die Messung durchgeführt wird. 0 ... 23: Kanalnummer
Timeout Monitoring	uint8	Ja	Timeout-Überwachung. 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant in dem Überwachungsmodus „Intvl Predef“.
Timeout Interval [ms]	real32	Ja	Zeitabstand (in Millisekunden) in dem die Messung auf Timeout überprüft wird. Wertebereich: $\geq 1.0$ ms
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 16-23** PB1651PWM1-In-Meas Device: Konfigurationsparameter der Registerkarte „Signals“

## 16.6.11 Die Messverfahren

*Begriffsdefinitionen***Aktive Signalflanke**

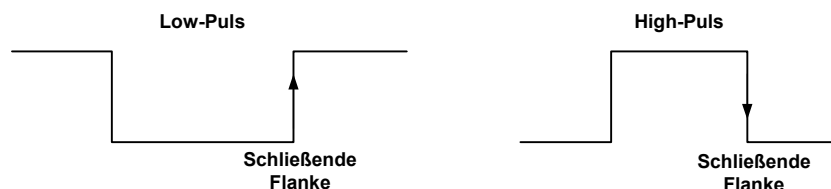
Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

**Aktives Signal**

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

**Schließende Flanke:**

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.

**Inaktives Signal**

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

**Öffnende Flanke**

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.

*Das Grundprinzip der Messwertberechnung*

Der PB1651PWM1-RTIO-Treiber berechnet die Messwerte der einzelnen Messverfahren aus einem einheitlichen Satz von Daten, die er zu Beginn der Messwertberechnung aus dem Dual-Ported-RAM-Speicher des ES4350 Carrier Board oder des ES1651.1 Carrier Board ausliest. Der Datensatz wird von der Firmware der Trägerkarte periodisch aktualisiert, die Zugriffe des RTIO-Treibers auf diesen Datensatz erfolgen asynchron zu dieser Periode.

Der Datensatz hat folgenden Aufbau:

- Ein 32-Bit Zeitstempel, der den Zeitpunkt angibt, an dem die PB1651PWM1-Hardware die Daten erfasst und im Dual-Ported-RAM abgelegt hat. Dieser Zeitpunkt wird nachfolgend als Datenerfassungszeitpunkt bezeichnet.
- Vier 32-Bit Datenwörter pro PWM-Eingang, die die Zeitstempel und Übergangsrichtung der letzten vier, dem Datenerfassungszeitpunkt vorangehenden Flanken auf dem PWM-Eingang codieren. Das oberste Bit (Bit 32) des jeweiligen Datenwortes codiert die Übergangsrichtung der Flanke (1: steigend, 0: fallend). Die unteren 31 Bit enthalten den Zeitstempel, wann die jeweilige Flanke aufgetreten ist.
- Ein 32-Bit Datenwort pro PWM-Eingang, das die kumulierte High-Zeit des PWM-Eingangs seit der Initialisierung des RTIO-Treibers und der PB1651PWM1-Hardware angibt.
- Ein 32-Bit Datenwort pro PWM-Eingang, das die kumulierte Low-Zeit des PWM-Eingangs seit der Initialisierung des RTIO-Treibers und der PB1651PWM1-Hardware angibt.
- Ein 16-Bit Datenwort pro PWM-Eingang, das die Anzahl der steigenden Flanken auf dem PWM-Eingang seit Initialisierung des RTIO-Treibers und der PB1651PWM1-Hardware angibt.
- Ein 16-Bit Datenwort pro PWM-Eingang, das die Anzahl der fallenden Flanken auf dem PWM-Eingang seit Initialisierung des RTIO-Treibers und der PB1651PWM1-Hardware angibt.

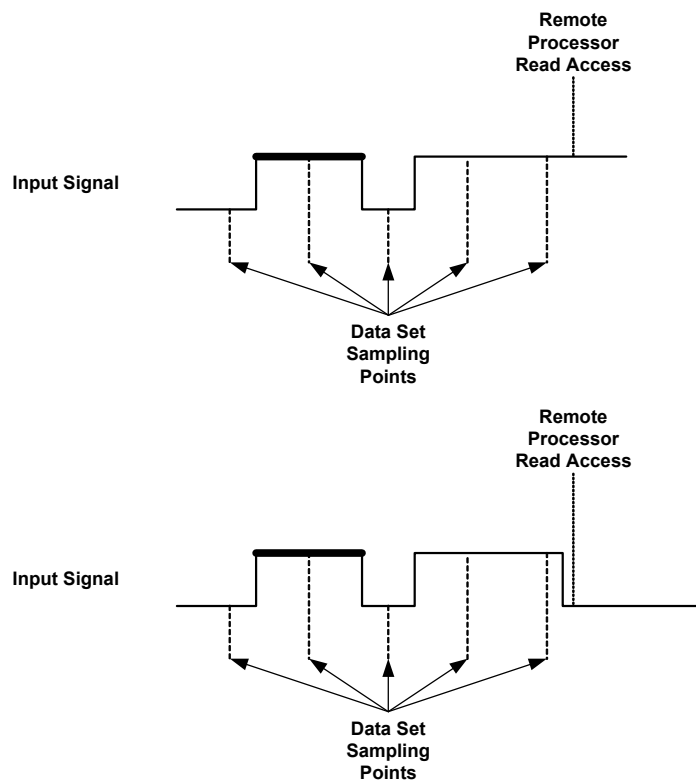
Die Zeitstempel weisen eine Auflösung von 20 ns auf.

### 16.6.12 Pulsweitenmessungen

Mit den Messverfahren

- High Time [ $\mu\text{s}$ ]
- Low Time [ $\mu\text{s}$ ]

werden die High- und Low-Zeiten der Pulse eines PWM-Eingangs vermessen. Es wird die Pulsweite des dem letzten Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehenden Pulses gemessen. Abb. 16-41 zeigt dies beispielhaft für eine High-Zeit-Messung, der gemessene Puls ist mit fetter Strichstärke hervorgehoben.



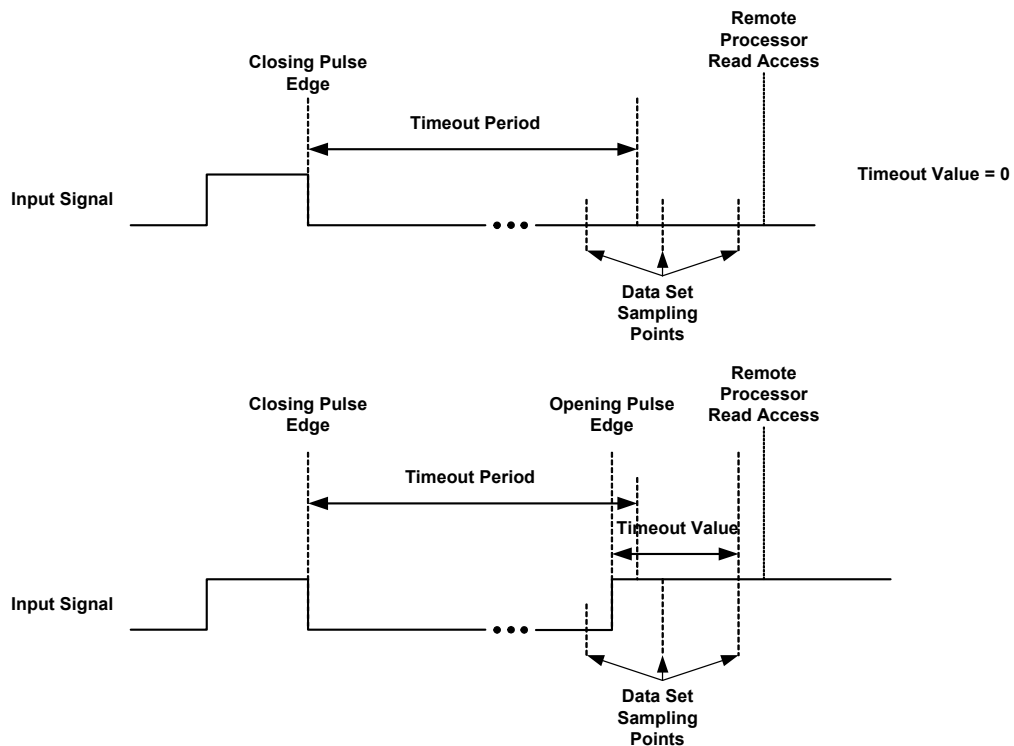
**Abb. 16-41** High-Zeit-Messung (der gemessene Puls ist mit fetter Strichstärke hervorgehoben)

Ist eine Timeout-Überwachung für die Messung aktiviert, so wird genau dann ein Timeout ausgelöst, wenn die Zeitdifferenz zwischen dem letzten Datenerfassungszeitpunkt und der letzten schließenden Pulsflanke größer als die eingestellte Timeout-Periode ist.

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-24 angegeben..

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	- Ist der Signalpegel zum Datenerfassungszeitpunkt inaktiv, so wird als Messwert 0 ausgegeben (Abb. 16-42). - Ist der Signalpegel zum Datenerfassungszeitpunkt aktiv, so wird als Messwert die Zeitdifferenz zwischen dem Datenerfassungszeitpunkt und der letzten öffnenden Flanke des Pulses ausgegeben (Abb. 16-42).
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 16-24** Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung



**Abb. 16-42** Pulsweitenmessungen: Messwert im Timeout-Fall im „Intvl InpDep“-Modus am Beispiel einer High-Zeit-Messung

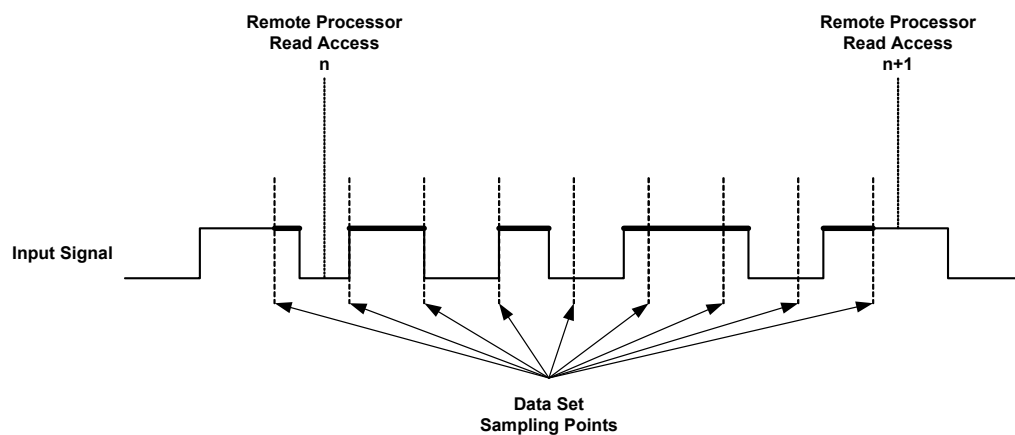
16.6.13 Additive Pulsweitenmessungen

Wird der Leseprozess für die „MeasVal“-Signalgruppe wiederholt (im Spezialfall periodisch) aktiviert, so liefern die Messverfahren

- Additive High Time [ $\mu\text{s}$ ]
- Additive Low Time [ $\mu\text{s}$ ]

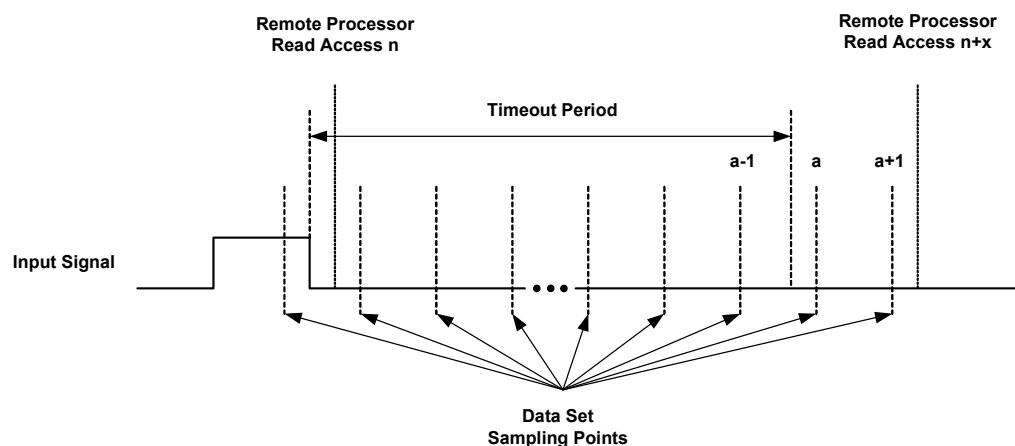
die Zeit, in der das zu vermessende Signal zwischen zwei aufeinanderfolgenden Aktivierungen des Leseprozesses insgesamt einen aktiven Pegel angenommen hat.

Abb. 16-43 veranschaulicht die Messwertberechnung am Beispiel einer additiven High-Zeit-Messung. Die Summe der fett markierten Liniensegmente ist die additive High-Zeit, die bei der (n+1)-ten Aktivierung des Leseprozesses geliefert wird.



**Abb. 16-43** Additive High-Zeit-Messung: Illustration der Messwertberechnung

Ist eine Timeout-Überwachung für die Messung aktiviert, so wird genau dann ein Timeout erkannt, wenn die Zeitdifferenz zwischen dem letzten Datenerfassungszeitpunkt und der letzten Pulsflanke des Signals (egal ob schließend oder öffnend) größer als die eingestellte Timeout-Periode ist. In dem in Abb. 16-44 dargestellten Fall läuft die Timeout-Zeit zwischen Datenerfassungszeitpunkt (a-1) und a ab. Der RTIO-Treiber liest kurz nach dem Datenerfassungszeitpunkt (a+1) Daten aus dem DPRAM der Trägerkarte aus und stellt einen Timeout fest.



**Abb. 16-44** Timeouts bei additiven Pulsweitenmessungen

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-25 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	Der Messwert im Timeout-Fall ist wie auch im Normalfall ohne Timeout gleich der Summe der aktiven Zeiten des Signals zwischen zwei Aktivierungen des Leseprozesses. - Diese Zeit ist 0, falls der Signalpegel während dieser Zeit inaktiv war. - Diese Zeit ist gleich der Aktivierungsperiode, falls der Signalpegel während dieser Zeit aktiv war.
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 16-25** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 16.6.14 Puls- und Flanken-zählung

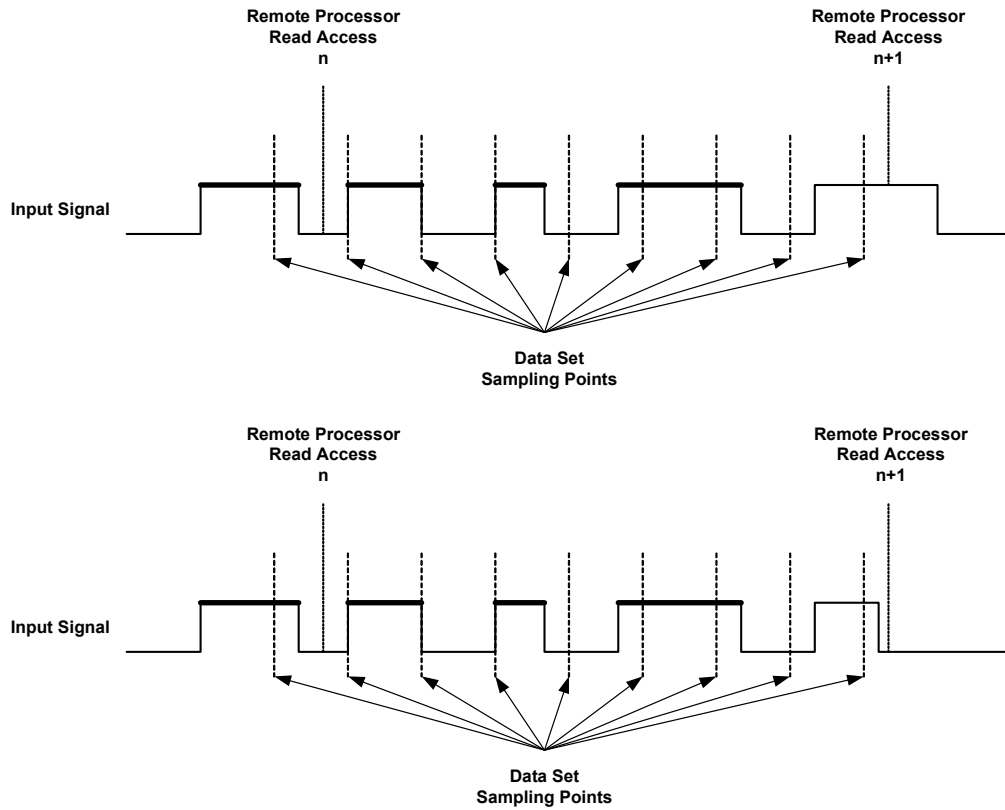
Wird der Leseprozess für die „MeasVal“-Signalgruppe wiederholt (im Spezialfall periodisch) aktiviert, so liefern die Messverfahren

- „Number of Low Pulses“
- „Number of High Pulses“
- „Number of Rising Edges“
- „Number of Falling Edges“

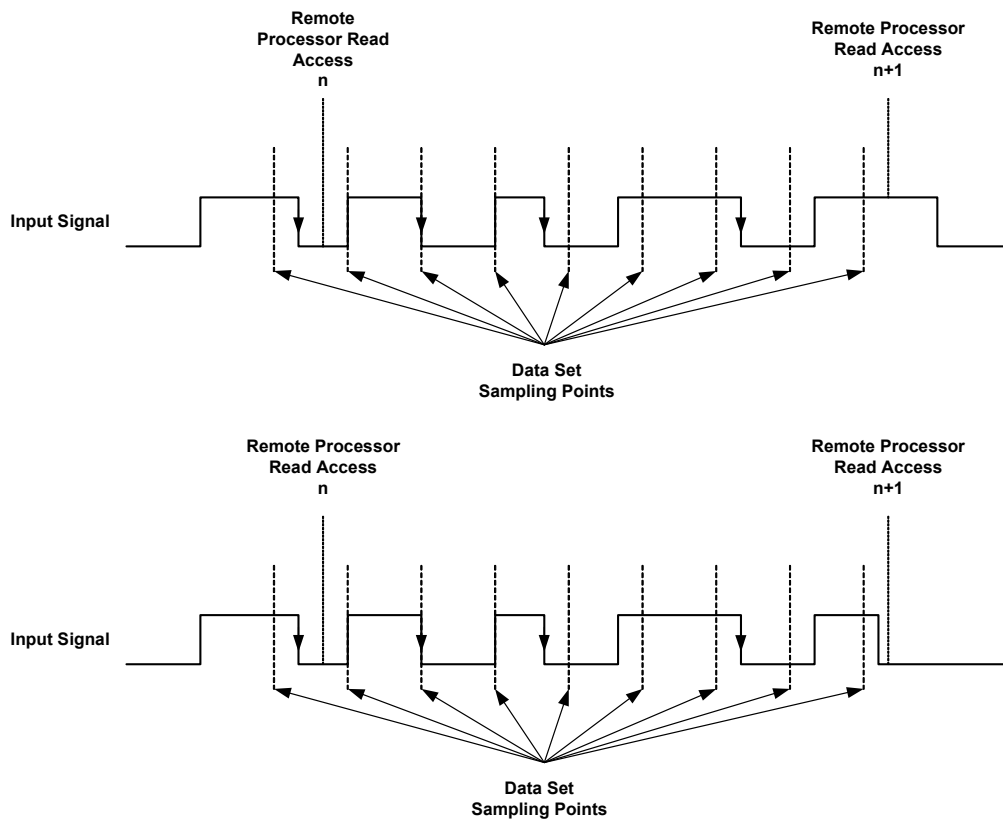
die Anzahl der aktiven Pulse bzw. Signalflanken zwischen zwei aufeinanderfolgenden Aktivierungen des Leseprozesses. Abb. 16-45 veranschaulicht die Methodik der Puls-Zählung am Beispiel einer High-Puls-Zählung im Detail. Gezählt werden alle Pulse, deren schließende Flanke in die Zeitspanne zwischen den Aktivierungen des Leseprozesses fällt. Wenn man es exakt formulieren will,



ist der Zeitraum bestimmt durch die den Aktivierungen unmittelbar vorausgehenden Datenerfassungszeitpunkte. Was die Flanken-zählung anbelangt, so ist dort der Auswertungszeitraum exakt der gleiche (Abb. 16-46).



**Abb. 16-45** High-Puls-Zählung: Die Anzahl der mit fetter Strichstärke markierten Pulse wird beim (n+1)-ten Lesevorgang als Messwert zurückgegeben



**Abb. 16-46** Zählung fallender Flanken: Die Zahl der markierten fallenden Flanken wird beim (n+1)-ten Lesezugriff als Messwert zurückgegeben

Ist eine Timeout-Überwachung für die Messung aktiviert, so wird genau dann ein Timeout ausgelöst, wenn die Zeitdifferenz zwischen dem letzten<sup>1</sup> Datenerfassungszeitpunkt und der letzten schließenden Flanke (Pulszählung) bzw. aktiven Flanke (Flanken-zählung) des Signals größer als die eingestellte Timeout-Periode ist.

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-26 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	Als Messwert im Timeout-Fall wird 0 ausgegeben.
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 16-26** Puls- und Flanken-zählungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt.

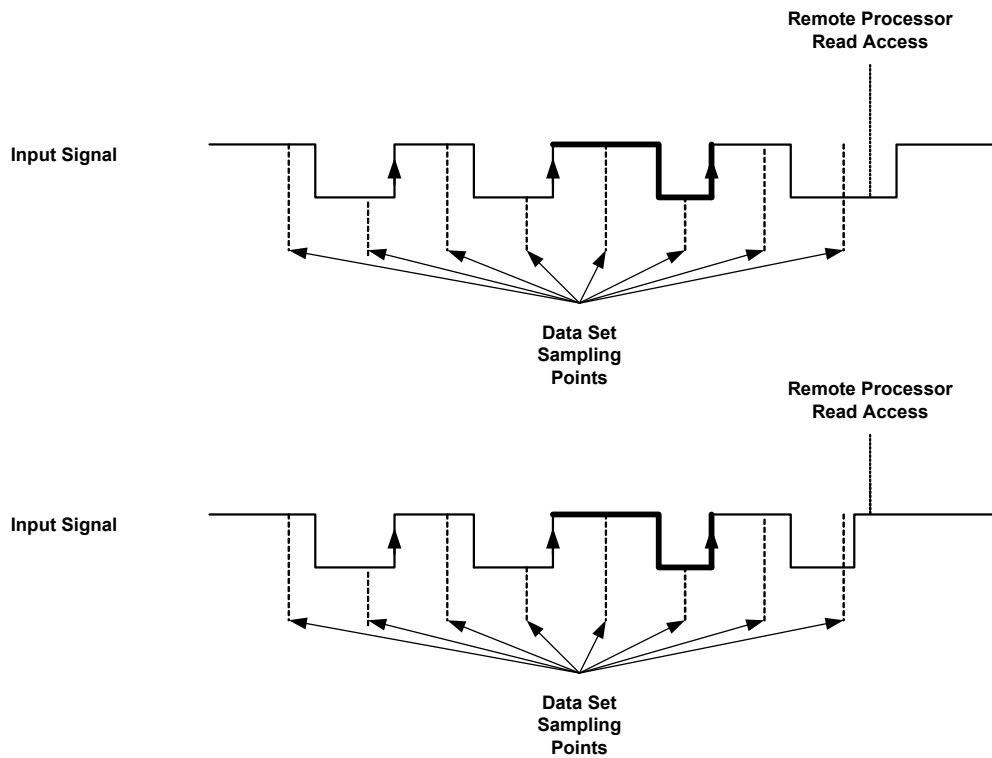
### 16.6.15 Frequenz- und Zykluszeitmessungen

Die Messverfahren

- „Cycle Time --/-- [ $\mu\text{s}$ ]“
- „Cycle Time --\-- [ $\mu\text{s}$ ]“
- „Frequency --/-- [Hz]“
- „Frequency --\-- [Hz]“

liefern die Zykluszeit bzw. die Frequenz des Signals an einem PWM-Eingang gemessen an steigenden oder fallenden Flanken. Es wird jeweils die dem letzten<sup>1</sup> Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehende aktive Signalflanke mit zugehöriger Periode ausgewertet.

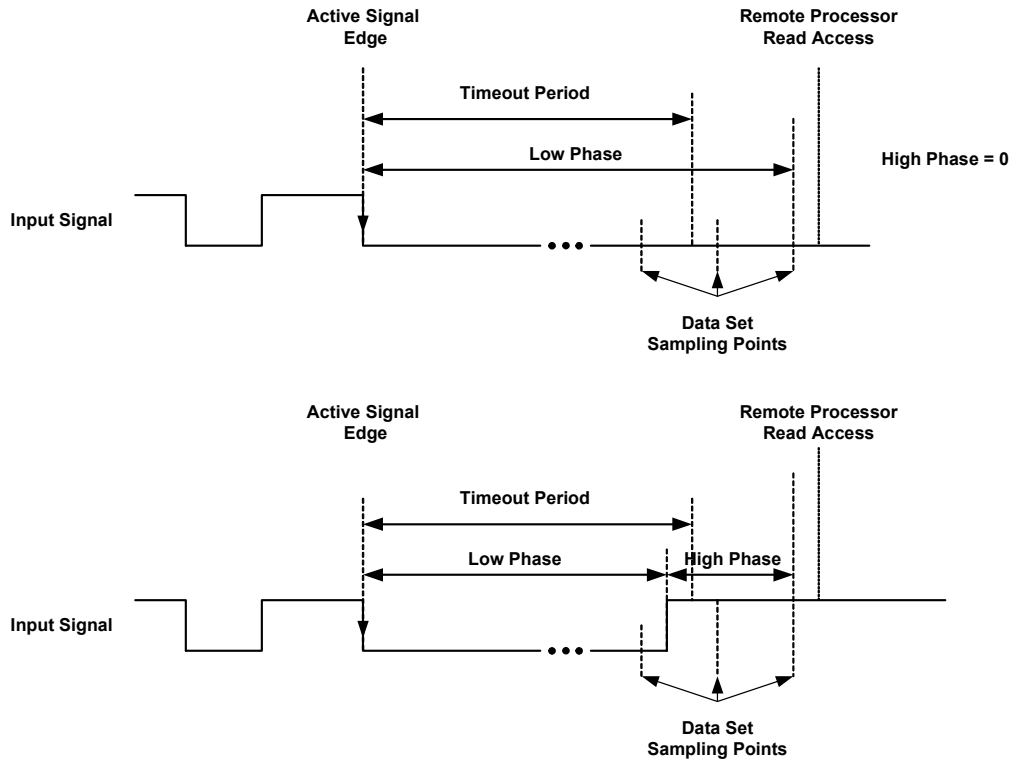
Abb. 16-47 zeigt die ausgewertete Periode beispielhaft für eine Frequenz- oder Zykluszeitmessung an ansteigenden Flanken.



**Abb. 16-47** Ausgewertete Periode (fett gezeichnet) bei einer Frequenz- oder Zykluszeitmessung an ansteigenden Flanken

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt.

Ist eine Timeout-Überwachung für die Messung aktiviert, so wird genau dann ein Timeout ausgelöst, wenn die Zeitdifferenz zwischen dem letzten<sup>1</sup> Datenerfassungszeitpunkt und der letzten aktiven Signalfanke größer als die eingestellte Timeout-Periode ist (Abb. 16-48).



**Abb. 16-48** Frequenz- und Zykluszeitmessungen: Messwert im Timeout-Fall beim Modus „Intvl InpDep“ zur Timeout-Überwachung

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-27 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	Bei Zykluszeitmessungen wird die Zeitdifferenz zwischen dem letzten Datenerfassungszeitpunkt und der letzten aktiven Signalfanke ausgegeben. Bei Frequenzmessungen der dazu reziproke Wert.
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 16-27** Frequenz- und Zykluszeitmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt.

### 16.6.16 Tastverhältnismessungen

Tastverhältnisse können mit dem PB1651PWM1-Modul auf unterschiedlichste Arten ermittelt werden. Bei den Tastverhältnismessungen wird unterschieden zwischen Messungen, die die High-Phase eines Signals ins Verhältnis zur Low-Phase setzen (oder umgekehrt) und Messungen, die die High-Phase (oder die Low-Phase) eines Signals ins Verhältnis zur Zyklusdauer setzen.

Der erste Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis P/P bezeichnet, „P/P“ soll dabei andeuten, dass zwei Pulsdauern zueinander ins Verhältnis gesetzt werden. Der zweite Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis P/C bezeichnet, „P/C“ soll andeuten, dass hier eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt wird.

$$\text{Tastverhältnis } \overset{\text{P/P}}{\text{P/P}} : \frac{L}{H}, \frac{H}{L} \quad 0 \leq \text{Tastverhältnis } \overset{\text{P/P}}{\text{P/P}} \leq \infty$$

$$\text{Tastverhältnis } \overset{\text{P/C}}{\text{P/C}} : \frac{L}{L+H}, \frac{H}{L+H} \quad 0 \leq \text{Tastverhältnis } \overset{\text{P/C}}{\text{P/C}} \leq 1$$

Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Tastverhältnismessungen vom Typ „P/P“

- „Duty Factor L/H --/--“
- „Duty Factor L/H --\--“
- „Duty Factor H/L --/--“
- „Duty Factor H/L --\--“

und Tastverhältnismessungen vom Typ „P/C“

- „Duty Cycle L/(L+H) --/--“
- „Duty Cycle L/(L+H) --\--“
- „Duty Cycle H/(L+H) --/--“
- „Duty Cycle H/(L+H) --\--“

zur Verfügung.

Was die Auswahl der Periode des Eingangssignals anbelangt, auf Basis derer die Tastverhältnisberechnung durchgeführt wird, so gilt exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 16.6.15 auf Seite 455 beschrieben wurde.

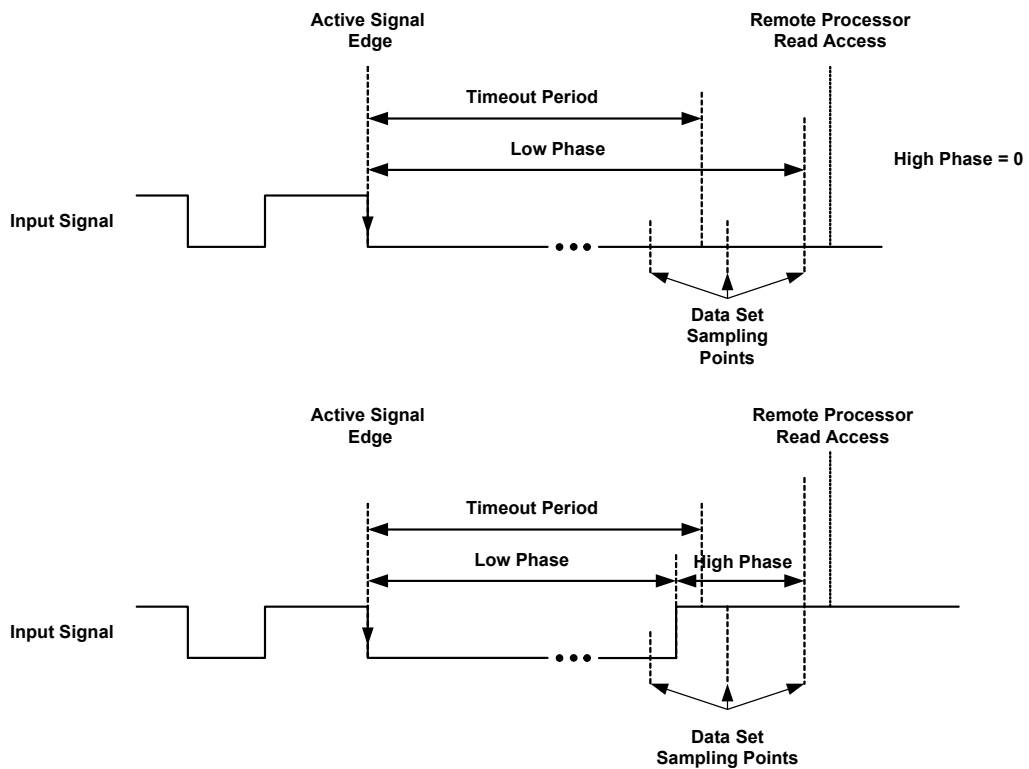
Auch was die Timeout-Überwachung anbelangt, gilt das in Abschnitt 16.6.15 auf Seite 455 Geschriebene: Ein Timeout wird genau dann ausgelöst, wenn die Zeitdifferenz zwischen dem letzten<sup>1</sup> Datenerfassungszeitpunkt und der letzten aktiven Signalfanke größer als die eingestellte Timeout-Periode ist.

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt.

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-28 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	Im Timeout-Fall werden die High-Phase und Low-Phase wie in Abb. 16-49 auf Seite 458 für den Fall von Tastverhältnissberechnungen an fallenden Flanken dargestellt bestimmt. Auf der Basis dieser Werte wird dann das Tastverhältnis berechnet.
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 16-28** Tastverhältnismessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung



**Abb. 16-49** Tastverhältnismessungen an fallenden Flanken: Bestimmung der High-Phase und Low-Phase des Signals im Timeout-Fall

16.6.17 Pegelmessungen

Die Messverfahren

- „Level (Active High)“
- „Level (Active Low)“

liefern den Pegel eines PWM-Eingangs in Form einer Aktiv-/Inaktiv-Information. „0“ bedeutet das Signal ist inaktiv, „1“ bedeutet das Signal ist aktiv.

Ist eine Timeout-Überwachung für die Messung aktiviert, so wird genau dann ein Timeout ausgelöst, wenn die Zeitdifferenz zwischen dem letzten<sup>1</sup> Datenerfassungszeitpunkt und der letzten Pulsflanke des Signals (egal ob schließend oder öffnend) größer als die eingestellte Timeout-Periode ist.

Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 16-29 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“	Der Messwert im Timeout-Fall ist gleich dem Signal-Zustand am letzten Datenerfassungszeitpunkt, also: - 0, falls der Signalpegel am letzten Datenerfassungszeitpunkt inaktiv war. - 1, falls der Signalpegel am letzten Datenerfassungszeitpunkt aktiv war.
„Intvl Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

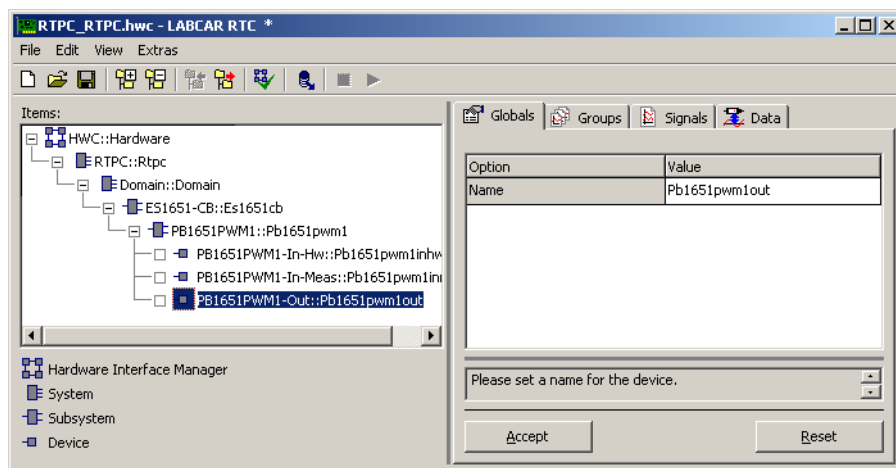
**Tab. 16-29** Pegelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

16.6.18 Das PB1651PWM1-Out Device

Das PB1651PWM1-Out Device dient zur Konfiguration und Ansteuerung der 16 Kanäle eines PB1651PWM1-Moduls zur PWM-Ausgabe.

16.6.19 Globals (PB1651PWM1-Out Device)

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.



**Abb. 16-50** Die Registerkarte „Globals“ des PB1651PWM1-Out Device

16.6.20 Groups (PB1651PWM1-Out Device)

Das PB1651PWM1-Out Device besitzt zwei Signalgruppen (siehe Abb. 16-51), die vom Experimentaltarget zum PB1651PWM1-Modul übertragen werden.

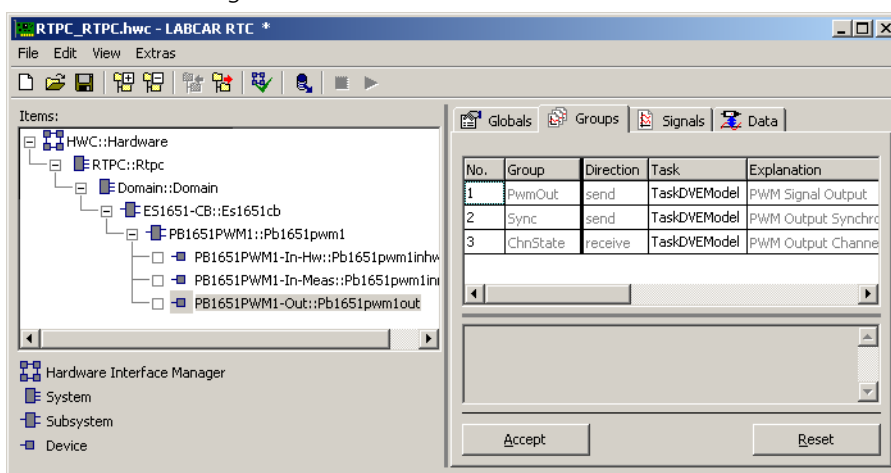
<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt.

Die Signalgruppe „PwmOut“ transportiert neben den Frequenz-, Tastverhältnis- und Phasendaten für die einzelnen PWM-Ausgänge noch Steuersignale zum Ein- und Ausschalten der PWM-Ausgabe.

Die Signalgruppe „Sync“ überträgt ein Synchronisationssignal, mit dem die Übernahme geänderter Frequenz- und Tastverhältnisdaten durch die PB1651PWM1-Hardware von RTIO aus synchronisiert werden kann.

Die Signalgruppe „ChnState“ empfängt Zustandssignale, welche den aktuellen Zustand der Ausgangstreiber der zugehörigen Ausgabekanäle beschreiben.

Den Signalgruppen sind Tasks des Echtzeit-Betriebssystems zuzuordnen. Für die „PwmOut“-Signalgruppe wird man eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der auszugebenden Signale. Für die „Sync“-Signalgruppe wird man eine Task mit periodischer oder Software-Aktivierung wählen.

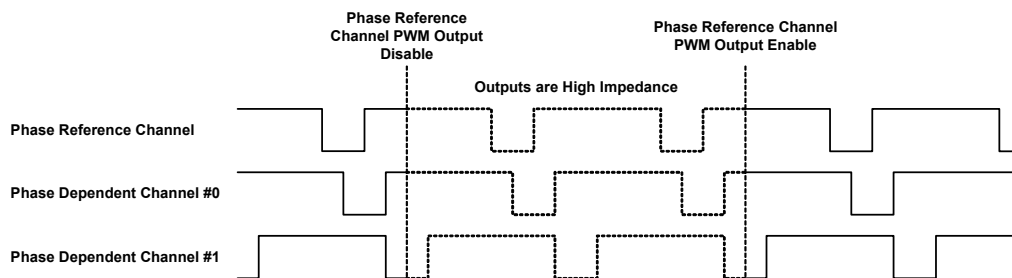


**Abb. 16-51** Die Registerkarte „Groups“ des PB1651PWM1-Out Device  
Die RTIO-Signale der Signalgruppe „PwmOut“

Die Signalgruppe „PwmOut“ umfasst 64 RTIO-Signale.

Mit den Signalen „EnableChn\_0“ bis „EnableChn\_15“ kann der jeweils zugeordnete PWM-Ausgang zwischen dem Zustand der PWM-Signal-Ausgabe und dem hochohmigen Zustand geschaltet werden. Mit diesen Signalen lässt sich somit im Rahmen einer Fehlersimulation zum Beispiel ein Leitungsbruch bei einem Aktor nachbilden.

Das konkrete Verhalten der PB1651PWM1-Hard- und Firmware bei einer Deaktivierung der PWM-Ausgabe soll anhand von Abb. 16-52 erläutert werden:



**Abb. 16-52** Aktivierung und Deaktivierung der PWM-Ausgabe von Kanälen, die in einem definierten Phasenbezug zueinander stehen



Stehen mehrere Kanäle in einer definierten Phasenbeziehung zueinander, so wird bei einer Deaktivierung bzw. Aktivierung der PWM-Ausgabe des Phasen-Referenz-Kanals auch die PWM-Ausgabe der phasenverschobenen Kanäle deaktiviert bzw. aktiviert. Die „EnableChn\_x“-RTIO-Signale der zu einem Phasen-Referenz-Kanal phasenverschobenen Kanäle werden ignoriert.

Wird die PWM-Ausgabe eines Kanals nach einer Deaktivierung wieder aktiviert, so wird die PWM-Ausgabe so fortgesetzt, als hätte die Unterbrechung nicht stattgefunden.

Mit den Signalen „Frequency\_0“ bis „Frequency\_15“ wird die Signalfrequenz des jeweils zugeordneten PWM-Ausgangs eingestellt. Der Frequenzbereich ist 0 bis 100 kHz und die Frequenzauflösung beträgt 0.01 Hz.

Mit den Signalen „DutyCycle\_0“ bis „DutyCycle\_15“ wird das Tastverhältnis des jeweils zugeordneten PWM-Ausgangs eingestellt. Tastverhältnisse werden mit Werten zwischen 0,0 und 1,0 definiert. 0,0 entspricht einem Tastverhältnis von 0%, 1,0 einem Tastverhältnis von 100%.

PB1651PWM1-Hardware und Firmware bieten die Möglichkeit, Phasenverschiebungen zwischen Ausgabe-Kanälen mit gleicher PWM-Frequenz definiert einzustellen. Hierzu ist einer dieser PWM-Ausgänge als Phasen-Referenz-Kanal zu bestimmen. Die Phasenlage der übrigen, in definierten Phasenbeziehungen zum Referenz-Kanal stehenden PWM-Ausgänge wird mit den diesen Ausgängen zugeordneten Phasen-Signalen „Phase\_x“ (x = 0, 1, ... 15) eingestellt.

Wird die Werteübernahme der Signalgenerierung auf der Hardware über die Synchronisationsleitungen der Trägerkarte gesteuert, so kann die Phasenlage der einzelnen PWM-Ausgänge ebenfalls nachträglich über die zugeordneten Phasen-Signale verändert werden.

Tab. 16-30 fasst die Eigenschaften des RTIO-Signals zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
EnableChn_0 ... EnableChn_15	bool	Aktivierung und Deaktivierung der PWM-Signal-Ausgabe. 0: Ausgang ist hochohmig, PWM-Signal-Ausgabe ist inaktiv 1: PWM-Signal-Ausgabe ist aktiv. Hat keine Bedeutung bei PWM-Ausgängen mit Phasen-Referenz-Kanal.
Frequency_0 ... Frequency_15	real32	Signalfrequenz - Frequenzbereich: 0 bis 100 kHz - Frequenzauflösung: 0,01 Hz - Hat keine Bedeutung bei PWM-Ausgängen mit Phasen-Referenz-Kanal.
DutyCycle_0 ... DutyCycle_15	real32	Tastverhältnis; Wertebereich: 0.0 ... 1.0
Phase_0 ... Phase_15	real32	Phasenlage; Wertebereich: -360.0° ... 360.0° Hat keine Bedeutung bei PWM-Ausgängen ohne Phasen-Referenz-Kanal

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-30** PB1651PWM1-Out Device: Die RTIO-Signale der Signalgruppe „PwmOut“

*Die RTIO-Signale der Signalgruppe „Sync“*

Die Signalgruppe „Sync“ enthält ein RTIO-Signal mit der Bezeichnung „SyncSgl“.

PB1651PWM1 RTIO-Treiber und Firmware bieten die Möglichkeit, den Zeitpunkt zu steuern, an dem von RTIO bzw. dem Simulationsmodell an das Modul übermittelte Frequenz- und Phasendaten von der Hardware übernommen werden. Die Steuerung erfolgt über die „Hardware Update“-Option der Registerkarte „Signals“. Wird in diesem Listenfeld für einen oder mehrere Kanäle die Option „RTIO Controlled“ eingestellt, so werden die an diese Kanäle übermittelten Frequenz- und Phasendaten erst dann von der Hardware übernommen, wenn von der Modul-Firmware auf dem „SyncSgl“-RTIO-Signal ein Übergang von 0 nach 1 detektiert wurde.

Tab. 16-31 fasst die Eigenschaften des RTIO-Signals zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
SyncSgl	bool	Ein Übergang von 0 nach 1 bewirkt, dass geeignet konfigurierte PWM-Ausgänge geänderte Frequenz- und Phasendaten an ihren Ausgängen aktualisieren.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-31** PB1651PWM1-Out Device: Die RTIO-Signale der Signalgruppe „Sync“

### Die RTIO-Signale der Signalgruppe „ChnState“

Die Signalgruppe „ChnState“ umfasst 16 RTIO-Signale mit der Bezeichnung „ChnState\_0“ bis „ChnState\_15“. Die Signalgruppe dient zur unabhängigen Erfassung der aktuellen Zustände der einzelnen Ausgangstreiber für die PWM-Signalgenerierung.

Tab. 16-32 fasst die Eigenschaften der RTIO-Signale zusammen.

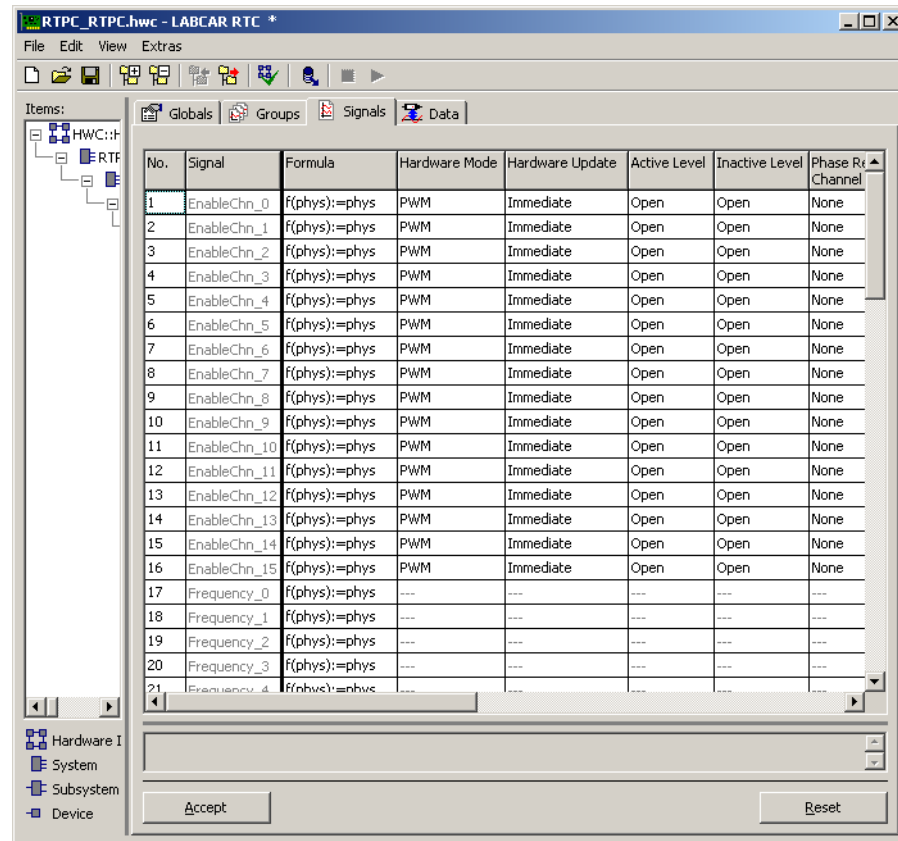
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
ChnState_0 ... ChnState_15	uint8	Jeder Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreiber: - Bit0: Lowside - Bit1: Highside A - Bit2: Highside B - Bit3: Channel Level Ist eines der ersten 3 Bits gesetzt, so wurde bei der aktuellen Konfiguration des Ausgangstreiber ein Überstrom detektiert. Das vierte Bit beschreibt den aktuellen Signalpegel (high oder low) des entsprechenden Ausgabekanal.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 16-32** PB1651PWM1-Out Device: Die RTIO-Signale der Signalgruppe „ChnState“

## 16.6.21 Signals (PB1651PWM1-Out Device)

In der Registerkarte „Signals“ wird die Konfiguration der 16 Ausgänge eines PB1651PWM1-Moduls durchgeführt. Abb. 16-53 zeigt die Optionsfelder der Registerkarte „Signals“. Alle Optionsfelder sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 16-53** Die Registerkarte „Signals“ des PB1651PWM1-Out Device

#### Hardware Mode

Jeder Ausgangskanal kann in zwei unterschiedlichen Modi betrieben werden „PWM“ und „Digital Out“.

- PWM
  - Ist diese Option aktiv, so können PWM-Signale mit einer minimalen Impulsdauer von 2  $\mu$ s generiert werden.
- Digital Out
  - Bei dieser Konfiguration des Ausgangskanals werden Signalpegel generiert. Der Frequenzbereich ist von 0 Hz bis 100 Hz (mit einer Auflösung von 0,01 Hz) begrenzt. Die Signalpegel werden über die RTIO-Signale

„DutyCycle\_x“ (x = 0, 1, ... 15) definiert. Werte kleiner 0,5 definieren einen „Low“-Pegel und Werte größer oder gleich 0,5 erzeugen einen „High“-Pegel.

### Hinweis

Im Mode „Digital Out“ können die weiteren Konfigurationsoptionen „Hardware Update“ und „Phase Reference Channel“ nicht verändert werden. Stehen Ausgangskanäle im Modus „PWM“ über die Option „Phase Reference Channel“ in Beziehung, so wird bei der Auswahl des Modus „Digital Out“ der Grundzustand hergestellt, d.h. alle Beziehungen hinsichtlich „Phase Reference Channel“ werden gelöscht.

### Hardware Update

In diesem Listenfeld wird eingestellt, wann vom Modell oder vom Anwender an den „Frequency\_x“- und „DutyCycle\_x“-RTIO-Signalen (x = 0, 1, ... 15) vorgenommene Wertänderungen von der PB1651PWM1-Hardware übernommen werden.

Folgende Einstellungen sind möglich:

- **Immediate**  
Wertänderungen werden sofort von der Hardware übernommen.
- **RTIO Controlled**  
Wertänderungen werden erst dann übernommen, wenn die Modul-Firmware einen Übergang von 0 nach 1 auf dem „SyncSgl“-RTIO-Signal detektiert (siehe hierzu auch „Die RTIO-Signale der Signalgruppe „Sync““ auf Seite 462).
- **Cycle End**  
Wertänderungen werden erst nach Abschluss der aktuellen PWM-Periode des jeweiligen PWM-Ausgangs übernommen.
- **Sync Line #x --\--**  
Wertänderungen werden erst an der nächsten fallenden Flanke auf Synchronisations-Leitung x übernommen.
- **Sync Line #x --/--**  
Wertänderungen werden erst an der nächsten steigenden Flanke auf Synchronisations-Leitung x übernommen.

In Verbindung mit den zugehörigen Phasen-Signalen kann die Phasenlage der einzelnen PWM-Ausgänge eingestellt werden. Nach der Konfiguration einer der oben genannten Optionen werden die betreffenden Phasen-Signale ebenfalls freigeschaltet.

### Hinweis

Die Einstellungen im Listenfeld „Hardware Update“ wirken sich nur auf die „Frequency\_x“- und „DutyCycle\_x“-RTIO-Signale aus. Wertänderungen der „EnableChn\_x“-RTIO-Signale (x = 0, 1, ... 15) werden generell sofort von der Hardware übernommen. Wertänderungen eines RTIO-Signals „Phase\_x“ (x = 0, 1, ... 15) werden generell nach Abschluss der aktuellen PWM-Periode des dem PWM-Ausgang x zugeordneten Phasen-Referenz-Kanals übernommen.

**Hinweis**

Bei Kanälen, die in einer definierten Phasenbeziehung zu einem Referenz-Kanal stehen, ist dieses Listenfeld deaktiviert - es werden implizit die Einstellungen des Phasen-Referenz-Kanals in diesem Feld übernommen.

*Active Level*

Der aktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, nimmt es während 60% der Periodendauer den aktiven Signalzustand an.

In diesem Listenfeld wird der aktive Zustand des PWM-Signals eingestellt. Zur Auswahl stehen die folgenden Optionen:

- **Open**  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- **Ground**  
Der PWM-Ausgang liegt auf Masse.
- **V Batt A Out**  
Der PWM-Ausgang gibt Batteriespannung A aus.
- **V Batt B Out**  
Der PWM-Ausgang gibt Batteriespannung B aus.

*Inactive Level*

Der inaktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, so nimmt es während 40% der Periodendauer den inaktiven Signalzustand an.

In diesem Listenfeld wird der inaktive Zustand des PWM-Signals eingestellt. Zur Auswahl stehen die folgenden Optionen:

- **Open**  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- **Ground**  
Der PWM-Ausgang liegt auf Masse.
- **V Batt A Out**  
Der PWM-Ausgang gibt Batteriespannung A aus.
- **V Batt B Out**  
Der PWM-Ausgang gibt Batteriespannung B aus.

*Phase Reference Channel*

PB1651PWM1-Hardware und Firmware bieten die Möglichkeit, Phasenverschiebungen zwischen Ausgabe-Kanälen mit gleicher PWM-Frequenz definiert einzustellen. Hierzu ist einer dieser PWM-Ausgänge als Phasen-Referenz-Kanal zu

bestimmen. Im Listenfeld „Phase Reference Channel“ der übrigen - zu diesem Referenz-Kanal in definierten Phasenbeziehungen stehenden - Ausgänge ist dann die Nummer des Referenz-Kanals einzutragen.

Wird ein PWM-Ausgang dahingehend konfiguriert, dass er in einer definierten Phasenbeziehung zu einem Referenz-Kanal steht (im Listenfeld „Phase Reference Channel“ ist ein Wert ungleich „None“ eingestellt), so ergeben sich daraus die folgenden Konsequenzen:

- Da phasenverschobener PWM-Ausgang und Phasen-Referenz-Kanal die gleiche PWM-Frequenz aufweisen müssen, wird das RTIO-Signal „Frequency\_x“ des phasenverschobenen PWM-Ausgangs von der Firmware ignoriert. Es wird automatisch der „Frequency\_x“-Wert des Phasen-Referenz-Kanals übernommen.
- Das Listenfeld „Hardware Update“ des phasenverschobenen PWM-Ausgangs ist deaktiviert. Für den phasenverschobenen PWM-Ausgang wird automatisch die Einstellung des Phasen-Referenz-Kanals in diesem Feld übernommen.

Ist im Listenfeld „Hardware Update“ des Phasen-Referenz-Kanals und damit, wie gerade beschrieben, automatisch auch im Listenfeld „Hardware Update“ des phasenverschobenen PWM-Ausgangs die Option „Cycle End“ eingestellt, so ist darunter auch beim phasenverschobenen PWM-Ausgang das Ende der PWM-Periode des Phasen-Referenz-Kanals zu verstehen.

- Verkettete Phasenbeziehungen sind nicht erlaubt, d.h. wenn mehrere PWM-Ausgänge in definierten Phasenbeziehungen miteinander stehen, so muss für diese Kanäle genau ein PWM-Ausgang als Phasen-Referenz-Kanal bestimmt werden. Dieser Kanal muss dann bei den restlichen phasenverschobenen Ausgängen im Listenfeld „Phase Reference Channel“ eingestellt werden. Für den Phasen-Referenz-Kanal ist im Listenfeld „Phase Reference Channel“ die Option „None“ einzustellen.
- Der Wert des RTIO-Signals „Phase\_x“ eines Phasen-Referenz-Kanals wird ignoriert.

Die Eigenschaften der einzelnen Parameter sind in Tab. 16-33 zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Hardware Mode	uint8	Ja	Art der Signalgenerierung PWM: 0 Digital Out: 1
Hardware Update	uint8	Ja	Zeitpunkt bzw. Ereignis, an dem die PB1651PWM1-Hardware eine geänderte Frequenz oder ein geändertes Tastverhältnis am zugeordneten Ausgang übernimmt. „Zeitpunkt/Ereignis“ Parameterwert „Immediate: 0 „RTIO Controlled: 1 „Cycle End“: 14 „Sync Line #0 --/--“: 2 „Sync Line #1 --/--“: 3 „Sync Line #2 --/--“: 4 „Sync Line #3 --/--“: 5 „Sync Line #4 --/--“: 6 „Sync Line #5 --/--“: 7 „Sync Line #0 --/--“: 8 „Sync Line #1 --/--“: 9 „Sync Line #2 --/--“: 10 „Sync Line #3 --/--“: 11 „Sync Line #4 --/--“: 12 „Sync Line #5 --/--“: 13
Active Level	uint8	Ja	Aktiver Signalzustand „Open“: 0 „Ground“: 1 „V Batt A Out“: 2 „V Batt B Out“: 3
Inactive Level	uint8	Ja	Inaktiver Signalzustand „Open“: 0 „Ground“: 1 „V Batt A Out“: 2 „V Batt B Out“: 3
Phase Reference Channel	sint8	Ja	Phasen-Referenz-Kanal „None“: -1 #0 ... #15: 0 ... 15
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 16-33** PB1651PWM1-Out Device: Konfigurationsparameter der Registerkarte „Signals“



**Hinweis**

*\*\* Die Optionen „Sync Line #4 --/--“, „Sync Line #5 --/--“, „Sync Line #4 --\--“ und „Sync Line #5 --\--“ sind nur dann verfügbar, wenn das Modul auf ein ES4350 Carrier Board gesteckt ist.*



## 17 ES4315-VXI - VME64x/VXI Adapter

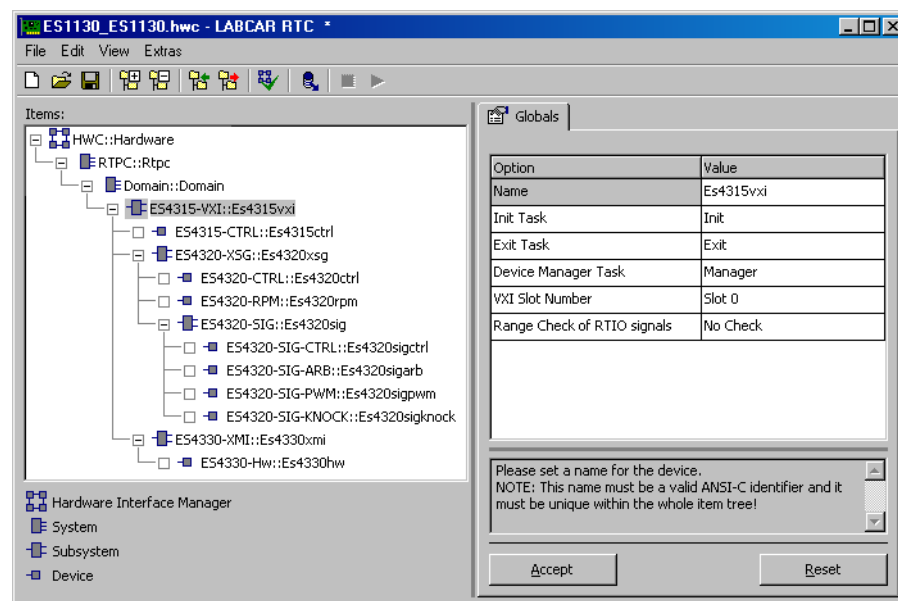
Der ES4315 VME64x/VXI Adapter fungiert als Bindeglied zwischen dem VME64x- und dem VXI-Teil des ES4300-Racks. Der Adapter ist fest im ES4300 Gehäuse verbaut und durch den Anwender nicht zugänglich. Er wird VME-seitig als Slavekarte angesprochen und erscheint VXI-seitig als Slot 0 Controller. Damit kann eine VME- Masterkarte zur Steuerung des VXI-Systems verwendet werden.

Die Einbindung eines ES4300 VXI-Systems im RTIO-Editor erfolgt durch Einfügen eines ES4315-VXI Subsystem Items innerhalb eines ES1130 Systems. Es kann jeweils nur ein ES4315-VXI Subsystem in ein ES1130 System eingefügt werden, da physikalisch der ES1130 Simulationsrechner im VME64x-Teil des ES4300-Racks eingesteckt ist. Alle weiteren VXI-Komponenten, wie die VXI-Boards können Sie unterhalb des ES4315-VXI Items eingefügen. Für das ES4315-VXI Subsystem ist ein RTIO-Treiber implementiert, welcher die Initialisierung des gesamten VXI-Systems und die Verwaltung der VXI-Systemressourcen übernimmt.

### 17.1 ES4315-VXI Subsystem

#### 17.1.1 Globals (ES4315-VXI Subsystem)

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten des ES4315-VXI Subsystems beschrieben.



**Abb. 17-1** Die Registerkarte „Globals“ des ES4315-VXI Subsystems

#### *Device Manager Task*

Sie können eine Task definieren, in welcher die Device Manager API Funktion des Treiber aufgerufen wird. In dieser Version ist hier allerdings keine Funktionalität hinterlegt.

### VXI Slot Number

Diese Option bezeichnet die VXI Slot Position des ES4315-VME64x/VXI Adapters. Sie können die Auswahl Slot 0 nicht ändern, da der VME64/VXI Adapter immer Slot 0 des VXI-Systems entspricht. Der erste nutzbare Slot im VXI-Gehäuse entspricht daher Slot 1.

### Range Check of RTIO Signals

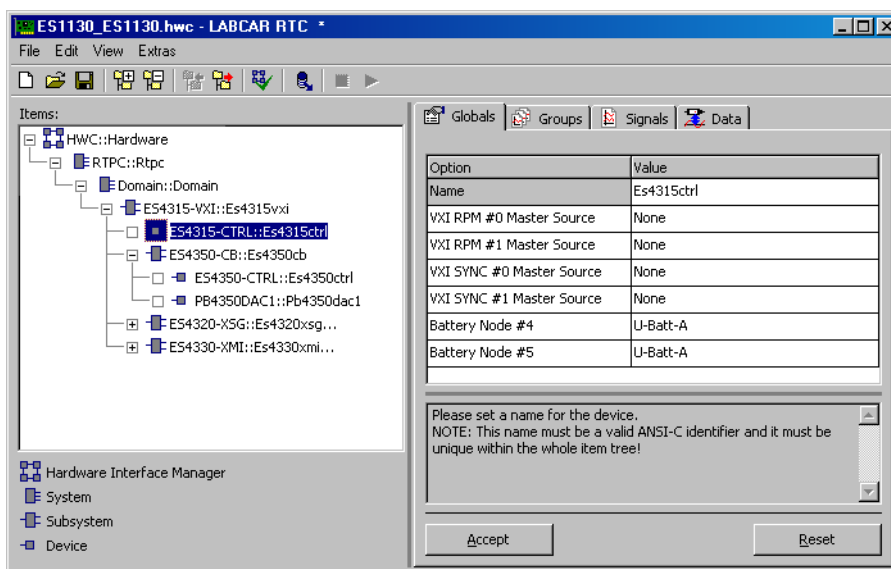
Mit dieser Option können Sie wählen, inwiefern der Wertebereich der RTIO-Signale zur Laufzeit überprüft werden soll. Die Einstellung kann von allen hierarchisch untergeordneten VXI-Kartentreibern verwendet werden. Folgende Einstellungen sind möglich. Sie wird in der Regel bei der Codegenerierung auf alle Signale sowohl von Send- als auch von Receive-Signalgruppen angewendet.

Einstellwert	Beschreibung
No Check	Es wird keine Prüfung vorgenommen
Limit to Range	Die Signale werden auf die von der Codegenerierung definierten Bereiche begrenzt.
Limit to Range and Notify to Monitor Window	Die Signale werden auf die von der Codegenerierung definierten Bereiche begrenzt und es erfolgt bei Bereichsüber- oder Unterschreitung eine Fehlermeldung in das Monitor Ausgabefenster. Die Fehlermeldung enthält den Namen des RTIO-Signals und des RTIO-Prozesses.

## 17.2 ES4315-CTRL Device

### 17.2.1 Globals (ES4315-CTRL Device)

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten des ES4315-CTRL Devices beschrieben.



**Abb. 17-2** Die Registerkarte „Globals“ des ES4315-CTRL Devices

#### VXI RPM #0 Master Source, VXI RPM #1 Master Source

In LABCAR-Projekten ist es häufig erforderlich, Abläufe untereinander zu synchronisieren. Projekte zum Test von Motor-Steuergeräten zum Beispiel erfordern eine exakte Winkelsynchronisation zwischen der Generierung von Einspritz- und Zündsignalen auf dem Steuergerät und deren Vermessung auf Test-Hardware (z. B. ES4330-Karten). Motor-Steuergerät und Test-Hardware werden hierzu in LABCAR-Projekten mit demselben Kurbelwellen-Winkeltakt versorgt.

ES4300 basierte LABCAR-Systeme bieten hierfür auf der ES4300-Backplane zwei voneinander unabhängige Winkeltaktbusse mit jeweils 3 Leitungen. Eine Busleitung dient zur Synchronisation auf den Nulldurchgang des Kurbelwellen-Winkels, die zweite Leitung überträgt den eigentlichen Winkeltakt und auf der dritten Leitung ist die Drehrichtung des Motors codiert.

Ein Winkeltaktbus kann - wie jeder Hardwarebus - nur von einer Quelle getrieben werden. Mögliche Quellen sind neben den Winkeltakt-Generatoren auf ES4320-Karten auch ES4350-Karten.

Abnehmer für Winkeltakt-Signale sind:

- ES4330-Karten, die damit eine exakte Vermessung drehzahlsynchroner Signale durchführen.
- im Slave-Modus betriebene ES4320-Karten, die damit ihre Signalgeneratoren auf einen externen Winkeltakt synchronisieren.
- ES4350-Karten, die den Winkeltakt an die Aufsteckmodule weiterleiten.

Mit diesen Listenfeldern können Sie definieren, von welcher Quelle die Winkeltaktsignale für die beiden Busse auf der Backplane der ES4300-Signalbox stammen.

Die Listenfelder werden dynamisch aufgebaut. Wird eine ES4320-Karte in die Hardwarebeschreibung eingefügt, so werden die vier Winkeltakt-Generatoren (RPM #0 ... RPM #3) dieser Karte zur Auswahl angeboten. Sind die beiden externen Winkeltaktports SYNC0 und SYNC1 an der Frontplatte der ES4320-Karte als Eingänge konfiguriert, so werden sie ebenfalls zur Auswahl angeboten (Ext RPM #0 und Ext RPM #1). Nähere Informationen zur Konfiguration der externen Winkeltaktports einer ES4320-Karte finden Sie im Abschnitt „Sync #0, Sync #1 Configuration“ auf Seite 479.

Mögliche weitere Quellen sind neben den Winkeltakt-Generatoren auf ES4320-Karten auch ES4350-Karten. Wird eine ES4350-Karte in die Hardwarebeschreibung eingefügt, so können deren interne Winkeltaktbusse (RPM #0 und RPM #1) auf die Backplanebusse geschaltet werden. Zum Entstehungszeitpunkt dieser Dokumentation ist die sinnvolle Verwendung dieser Möglichkeit jedoch eingeschränkt, da noch keine Aufsteckmodule für die Winkeltakt-Generierung auf ES4350-Karten zur Verfügung stehen.

#### VXI SYNC #0 Master Source, VXI SYNC #1 Master Source

Neben den beschriebenen Bussen zur Synchronisation auf den Kurbelwellenwinkeltakt weist die ES4300-Backplane zwei weitere Leitungen auf, die der Synchronisation von Hardware-Einheiten auf verschiedenen VXIbus-Karten einer ES4300-Signalbox dienen. Die Listenfelder „VXI SYNC #0 Master Source“ und „VXI SYNC #1 Master Source“ dienen zur Festlegung der Hardwareeinheiten, die diese beiden Leitungen treiben. Die Listenfelder werden dynamisch aufgebaut. Wird eine ES4350-Karte der RTIO-Hardwarebeschreibung hinzugefügt, so

werden die sechs Leitungen des internen Synchronisationsbusses dieser Karte der Auswahl hinzugefügt. Jeder dieser sechs Sync-Leitungen einer ES4350-Karte kann auf eine oder beide Backplane-Leitungen geschaltet werden.

*Battery Node Control Source*

Die Einstellung der Batterieknotenzustände erfolgt in der Signalgruppe „BnStates“ des ES4315-CTRL Items. Für jeden der acht Batterieknoten kann dabei vorgegeben werden, ob er ein- oder ausgeschaltet ist.

**Hinweis**

*Somit ist diese Option für den Anwender normalerweise nicht sichtbar, wobei die Defaulteinstellung „All ES43XX Boards“ bewirkt, dass die Batterieknoten-information zu allen Karten per Befehl übertragen wird.*

*Battery Node #4*

Mit dieser Option können Sie wählen, welche Spannung (UBatt\_A oder UBatt\_B) auf Batterieknoten 4 geschaltet werden kann.

*Battery Node #5*

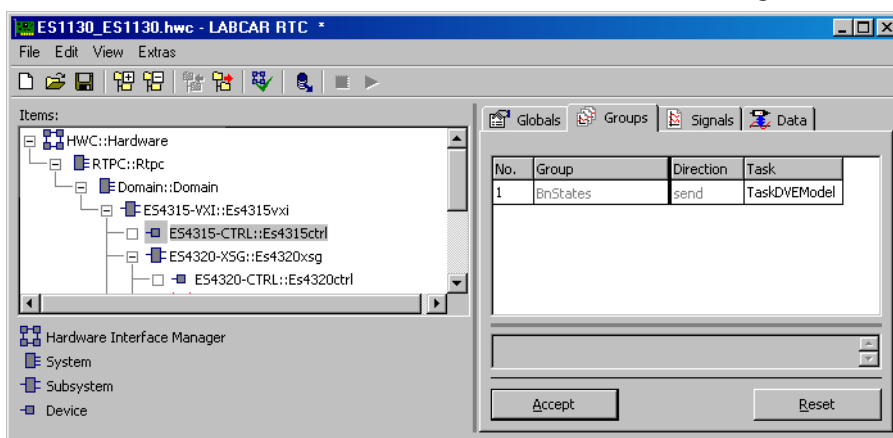
Mit dieser Option können Sie wählen, welche Spannung (UBatt\_A oder UBatt\_B) auf Batterieknoten 5 geschaltet werden kann.

**Hinweis**

*Es sind grundsätzlich zwei verschiedene Batteriespannungen UBatt\_A und UBatt\_B vorgesehen. Die Batterieknoten 0 bis 3 können nur mit UBatt\_A, die Batterieknoten 5 bis 7 nur mit UBatt\_B beschaltet werden. Für die Batterieknoten 4 und 5 ist wählbar, ob UBatt\_A oder UBatt\_B geschaltet wird. Diese LAB-CAR Systemeinstellung ist zur Laufzeit nicht veränderlich und wird dem VXI-System mit diesen Einstellungen mitgeteilt.*

17.2.2 Groups (ES4315-CTRL Device)

Das Device implementiert eine Send-Signalgruppe „BnStates“. Diese können Sie dazu benutzen, um dem VXI-System die Zustände der Batterieknoten mitzuteilen. Eine Kommunikation mit dem VXI-System wird vom Treiber erst dann ange-stossen, wenn sich mindestens der Zustand eines Batterieknotens geändert hat.



**Abb. 17-3** Die Registerkarte „Groups“ des ES4315-CTRL Devices

17.2.3 Signals (ES4315-CTRL Device)

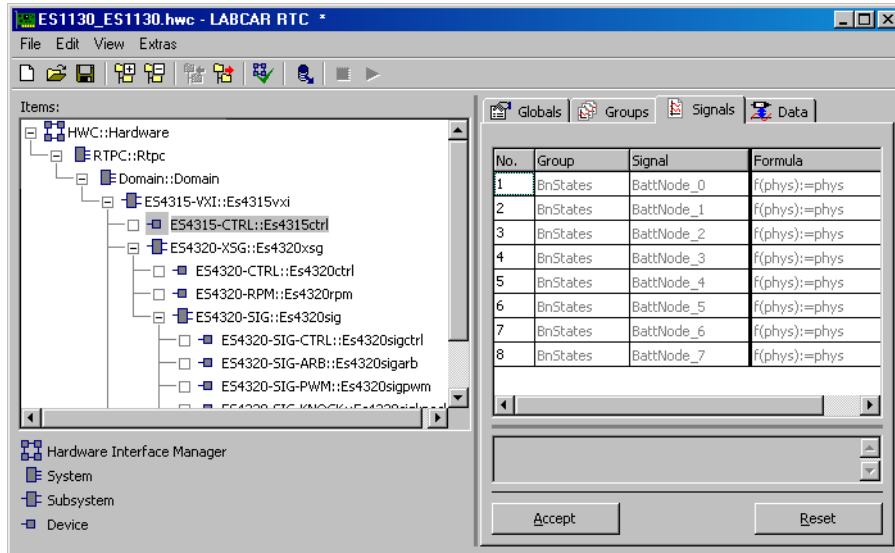


Abb. 17-4 Die Registerkarte „Signals“ des ES4315-CTRL Devices

Signal	Wirkungs- richtung	Beschreibung
BattNode_0 ... BattNode_7	send	Schaltzustand des Batterieknotens 0 ... 7

17.2.4 Datentypen und Wertebereiche

Signal	Datentypen	Mögliche Werte
BattNode_0 ... BattNode_7	bool	false [0] = Batterieknoten ausgeschaltet true [1] = Batterieknoten eingeschaltet

### 17.3 ES4320-XSG - PWM und arbiträre Signalgenerierung

---

Die ES4320-XSG-Karte (eXtended Signal Generator board) ermöglicht Erzeugung von PWM-Signalen und frei programmierbaren (arbiträren) Signalen mit Zündaussetzerfunktionalität und Klopfsignalen. Die Karte besitzt hierzu zwölf Signalgeneratoren (Kanäle). Jeder Kanal kann entweder als PWM-Kanal, als Arbiträrkanal oder als Klopfkanal benutzt werden.

#### *Gemeinsame Eigenschaften der Kanäle:*

---

- Galvanische Trennung möglich
- Zwei analoge und mehrere binäre Ausgangsbeschaltungen möglich
- Cut-Out-Relais zum Auftrennen des Kanals
- Einstellbare Pull-Up Strombegrenzung für binäre Ausgangsmodi
- Synchroner Steuerung der Kanäle in Dreiergruppen

#### *Eigenschaften Arbiträrkanal*

---

- Beschreibung des Signalverlaufs mit bis zu vier Signalbänken
- Vorgabe einer externen Referenzspannung möglich
- Kombination mit Zündaussetzergenerierung möglich

#### *Eigenschaften PWM-Kanal*

---

- Vorgabe von Frequenz, Tastverhältnis
- Vorgabe der Spannungswerte für Low und High Pegel möglich

#### *Eigenschaften Klopfkanal*

---

- Klopfsignalgenerierung für bis zu 20 Zylinder
- Vorgabe der Klopfsignalwahrscheinlichkeit, der Rauscheigenschaft und der Klopfhüllkurve.

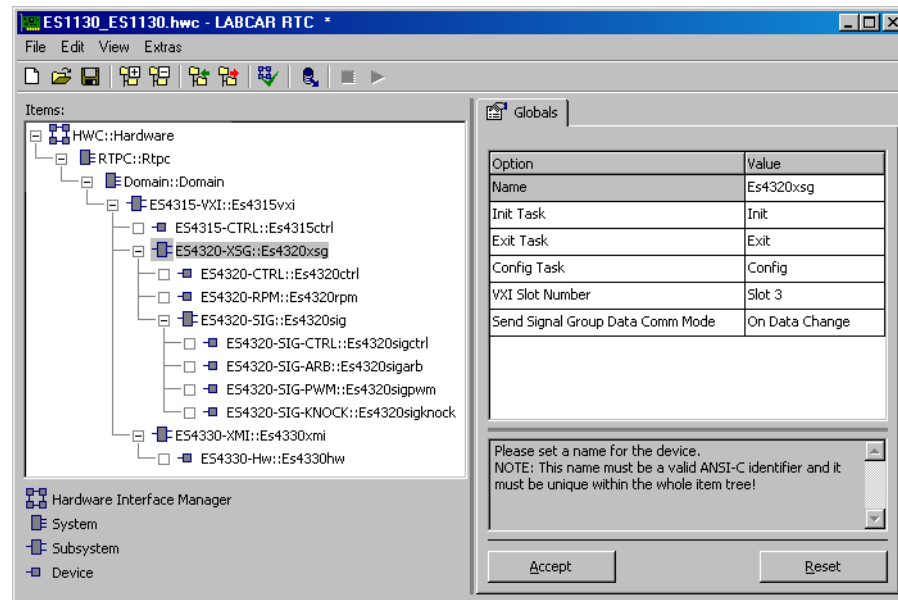
Eine weitere Einheit der Karte enthält vier unabhängige Drehzahl- (RPM) Generatoren. Diese ermöglichen die Erzeugung steuerbarer Drehsignale mit Startmarke und Richtungsinformation. Diese RPM Signale werden zur Ansteuerung der arbiträren Signalgeneratoren und zur Erfassung von Drehwinkel-synchronen Signalen im ES4330 VXI Signal Measurement Board (ES4330-XMI) benötigt.

Die Einbindung einer ES4320-Karte erfolgt durch Einfügen eines ES4320-XSG Items in ein ES4315-VXI Subsystem im RTIO-Editor. Es können mehrere ES4320-Karten in ein ES4315-VXI Subsystem eingefügt werden. Sie können jedoch insgesamt maximal 9 VXI-Karten eingefügen, da jedes Board einen VXI-Slot belegt.



### 17.3.1 Globals (ES4320-XSG Subsystem)

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten des ES4320-XSG Items beschrieben.



**Abb. 17-5** Die Registerkarte „Globals“ des ES4320-XSG Subsystems

#### *VXI Slot Number*

Mit dieser Option können Sie die ES4320-Karte einem VXI-Slot zuordnen. Aus der Zuordnung aller beschriebenen VXI-Karten zu Slots ergibt sich die tatsächliche Anordnung der Karten im ES4300 VXI-Gehäuse. Weiterhin muss der ausgewählte Eintrag mit der tatsächlichen Bestückung im ES4300-Gehäuse übereinstimmen. Ist dies nicht der Fall, wird der RTIO-Treiber deaktiviert und die im RTIO-Editor beschriebene Karte ist nicht benutzbar.

#### *Send Signal Group Data Comm Mode*

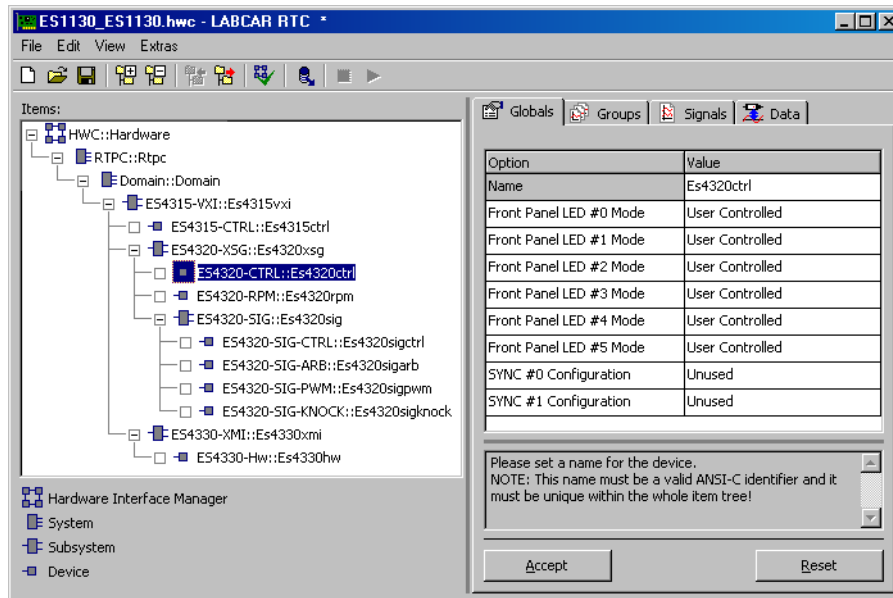
Mit dieser Option können Sie wählen, wie die Art der Kommunikation im ES1130-Treiber bei Signalgruppen gehandhabt wird, die Daten zum ES4320 Board übertragen. Die Daten können entweder unbedingt bei jeder Taskaktivierung (Einstellung „Every Interval“) oder bei Taskaktivierung und geänderten Signalgruppendaten (Einstellung „On Data Change“) übertragen werden. Dabei ist es hinreichend, dass sich ein beliebiges Signal der Signalgruppe gegenüber der letzten erfolgreichen Kommunikation verändert hat.

Die Einstellung dieser Option gilt für alle Send-Signalgruppen der betreffenden ES4320-Karte.

### 17.3.2 ES4320-CTRL Device

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten des ES4320-CTRL Devices beschrieben. In diesem Device sind Signalgruppen untergebracht und können Einstellungen getätigt werden, welche unabhängig von einzelnen RPM-Generatoren oder einzelnen SIG-ARB- bzw. SIG-PWM- Kanälen wirken.

## 17.3.3 Globals (ES4320-CTRL Device)



**Abb. 17-6** Die Registerkarte „Globals“ des ES4320-CTRL Devices

*Front Panel LED #0 ... Front Panel LED #5 Mode*

Mit diesen Optionen können Sie festlegen, von welchen Quellen die entsprechende LED-Anzeigen an der Frontplatte der ES4320-Karte angesteuert werden. Die Einstellung „User Controlled“ ermöglicht dabei, die Ansteuerung der LEDs durch das Modell über die Signalgruppe „StateLEDs“ im selben Device. Weiterhin können mit den LEDs die Pegel der externen Disable-Eingänge „Dis\_0 ... Dis\_3“ und die Triggermarken verschiedener RPM Triggerquellen angezeigt werden. Da die Triggermarken sehr kurz sind, werden jedoch nicht die Pegel dargestellt, vielmehr ändert die LED bei jedem Triggerereignis seinen Anzeigezustand.

Einstellwert	Beschreibung
Disabled	Die LED wird nicht angesteuert.
User Controlled	Die LED wird durch Vorgaben des Benutzers (entsprechendes Signal LED_0 ... LED_5 der Signalgruppe „StateLEDs“) angesteuert.
Dis_0 Level ... Dis_3 Level	Die LED zeigt den Pegel des entsprechenden externen Disable Eingangs an (TTL High = LED an).

Einstellwert	Beschreibung
RPM #0 Trg Toggle ...	Die LED ändert ihren Zustand bei jeder Triggermarke des RPM-Signals des ausgewählten RPM-Generators auf diesem ES4320 Board.
RPM #3 Trg Toggle	
Sync #0 Trg Toggle ...	Die LED ändert ihren Zustand bei jeder Triggermarke des gewählten Sync-Signals (d. h. das zugrunde liegende RPM-Signal wird von einem anderen System eingespeist).
Sync #1 Trg Toggle	
VXI-RPM #0 Trg Toggle ... VXI-RPM #1 Trg Toggle	Die LED ändert ihren Zustand bei jeder Triggermarke des RPM-Signals des ausgewählten VXI-RPM-Signals (d.h. das zugrunde liegende RPM-Signal wird von einer anderen ES4320-Karte in diesem VXI-System erzeugt).

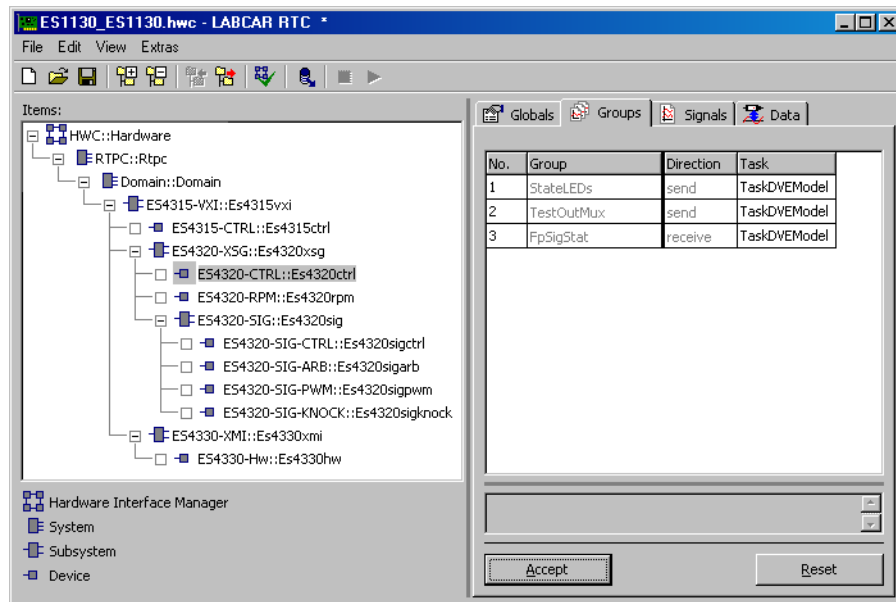
#### *Sync #0, Sync #1 Configuration*

Mit diesen Optionen wird Wirkungsweise und Richtung der SYNC0 und SYNC1 Signale an der Frontplatte festgelegt. Jede Buchse kann für eine RPM-Signal entweder als Ein- oder als Ausgang fungieren. Bei der Funktion als Ausgang kann ein auf der Karte erzeugtes RPM-Signal oder eines der beiden VXI-RPM Signale ausgegeben werden. Bei der Konfiguration als Eingang können Sync #0 (Ext-RPM #0) bzw Sync #1 (Ext-RPM #1) mit einem externen RPM-Signal beaufschlagt und somit als Trigger- oder Taktquelle für ARB-Kanäle ausgewählt werden. Die SYNC Konfigurationseinstellungen können zur Laufzeit nicht geändert werden.

Einstellwert	Beschreibung
Disabled	SYNC wird nicht verwendet
Input	Das externe Synchronisationssignal kann zur Synchronisierung eines RPM-Signals oder als direktes Synchronisationssignal eines Arbitärsignals verwendet werden.
Output RPM #0	Das Signal des RPM-Generators 0 ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.
Output RPM #1	Das Signal des RPM-Generators 1 ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.
Output RPM #2	Das Signal des RPM-Generators 2 ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.
Output RPM #3	Das Signal des RPM-Generators 3 ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.
Output VXI-RPM #0	Das Signal RPM #0 der VXI-Backplane ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.
Output VXI-RPM #1	Das Signal RPM #1 der VXI-Backplane ist mit dem Anschluss „SYNC“ auf der Frontplatte verbunden.

### 17.3.4 Groups (ES4320-CTRL Device)

Das ES4320-CTRL Device implementiert drei Signalgruppen, welche Sie für folgende Aufgaben benutzen können.



**Abb. 17-7** Die Registerkarte „Groups“ des ES4320-CTRL Devices

#### *StateLEds*

Mit diese Send-Signalgruppe wird der Zustand einer oder mehrerer der sechs LEDs an der Frontplatte definiert, falls die Ansteuerung der jeweiligen LED durch das Modell erfolgt (Einstellwert „User Controlled“ der Option „Front Panel LED #0 ... 5“, siehe auch Abschnitt 17.3.3 auf Seite 478).

#### *TestOutMux*

Mit dieser Send-Signalgruppe wird festgelegt, mit welchen Signalen die jeweils zwei analogen und TTL Testausgänge auf der Frontplatte beschaltet werden.

#### *FpSigStat*

Diese Receive-Signalgruppe ermöglicht es, die logischen Zustände der Trigger- und Disableingänge der Frontplatte abzufragen.

17.3.5 Signals (ES4320-CTRL Device)

In diesem Abschnitt werden die Signale des ES4320-CTRL Devices beschrieben.

Signal	Wirkungsrichtung	Beschreibung
LED_0 ... LED_6	send	Festlegung der Zustände der Frontplatten- LEDs im Modus „UserDefined“
TestOutMuxAna_0 („TST0“)*, TestOutMuxAna_1 („TST1“)*	send	Auswahl der Quellen für die beiden Analogtestausgänge an der Frontplatte
TestOutMuxTTL_0 („TST2“)*, TestOutMuxTTL_1 („TST3“)*	send	Auswahl der Quellen für die beiden TTL-Testausgänge an der Frontplatte
FpTrigger_0 ... FpTrigger_3	send	Zustände der Triggereingänge an der Frontplatte
ExtDisable_0 ... ExtDisable_3	send	Zustände der Disable-Eingänge an der Frontplatte

\* In der Klammer sind die jeweiligen Ausgänge des ES4320 Boards angegeben, die diese Signale führen.

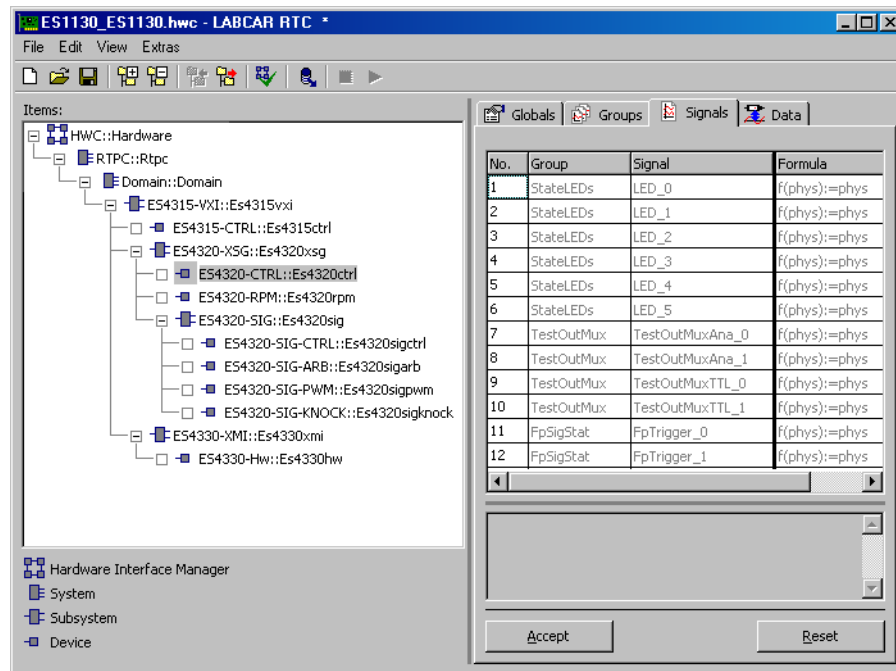


Abb. 17-8 Die Registerkarte „Signals“ des ES4320-CTRL Devices

## 17.3.6 Datentypen und Wertebereiche

Signal	Datentypen	Mögliche Werte
LED_0 ... LED_6	bool	false [0] = LED aus true [1] = LED an
TestOutMuxAna_0 („TST0“)*, TestOutMuxAna_1 („TST1“)*	uint	0 ... 11 = Analogsignal des PWM-/ARB Kanals der ID 0 ... 11
TestOutMuxTTL_0 („TST2“)*, TestOutMuxTTL_1 („TST3“)*	uint	0, 1 = Trigger von ExtRpm_0, ExtRpm_1 2, 3 = Trigger von VxiRpm_0, VxiRpm_1 4, 5 = Taktsignal von ExtRpm_0, ExtRpm_1 6, 7 = Taktsignal von VxiRpm_0, VxiRpm_1 8 ... 11 = Triggersignal von RPM-Generator #0 ... #3 12 ... 15 = Taktsignal von RPM-Generator #0 ... #3 des Boards 16 ... 27 = lokales Taktsignal des PWM-/ARB Kanals der ID 0 ... 11 32 ... 35 = Zusätzliche Triggermarken (Additional Trigger) 0 ... 3 40 ... 45 = interne Testpunkte
FpTrigger_0 ... FpTrigger_3	bool	false [0] = TTL low am jeweiligen Eingang true [1] = TTL high am jeweiligen Eingang
ExtDisable_0 ... ExtDisable_3	bool	false [0] = TTL low am jeweiligen Eingang true [1] = TTL high am jeweiligen Eingang
* In der Klammer sind die jeweiligen Ausgänge des ES4320 Boards angegeben, die diese Signale führen.		

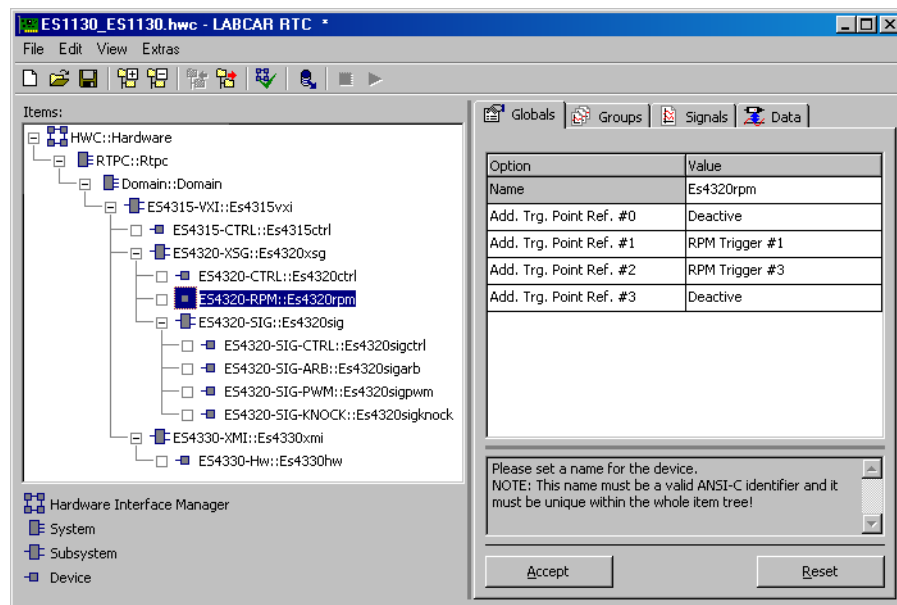
## 17.3.7 ES4320-RPM Device

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten des ES4320-RPM Devices beschrieben.

In diesem Device sind Signalgruppen definiert und können Einstellungen vorgenommen werden, welche für Betrieb und Konfiguration der vier RPM-Generatoren erforderlich sind. Jeder der vier RPM-Generatoren ist in der Lage, ein Drehwinkelsignal (RPM-Signal) bestehend aus Takt, Richtung und Bezugsmarke zu erzeugen. Diese RPM-Signale können nun von einem Arbiträrkanal auf dieser Karte verwendet werden und/oder über die VXI-RPM #0 oder VXI-RPM #1 Ressourcen anderer Karten (ES4320 oder ES4330) in diesem VXI-System zur Verfügung gestellt werden.

## 17.3.8 Globals (ES4320-RPM Device)

In dieser Registerkarte können Sie Konfigurationseinstellungen vornehmen, welche auf alle vier RPM-Generatoren gemeinsam wirken.



**Abb. 17-9** Die Registerkarte „Globals“ des ES4320-RPM Devices

*Add. Trg. Point Ref. #0 ... Add Trg Point Ref. #3*

Mit dieser Option können Sie festlegen, auf welchen RPM-Generator sich die zusätzlichen Triggermarken beziehen. Die zusätzliche Triggermarke ergibt sich dann jeweils aus der Nullmarke des gewählten RPM-Generators und der in der Signalgruppe „AddTrgPhase“ definierten Phasenverschiebung für diese zusätzliche Triggermarke.

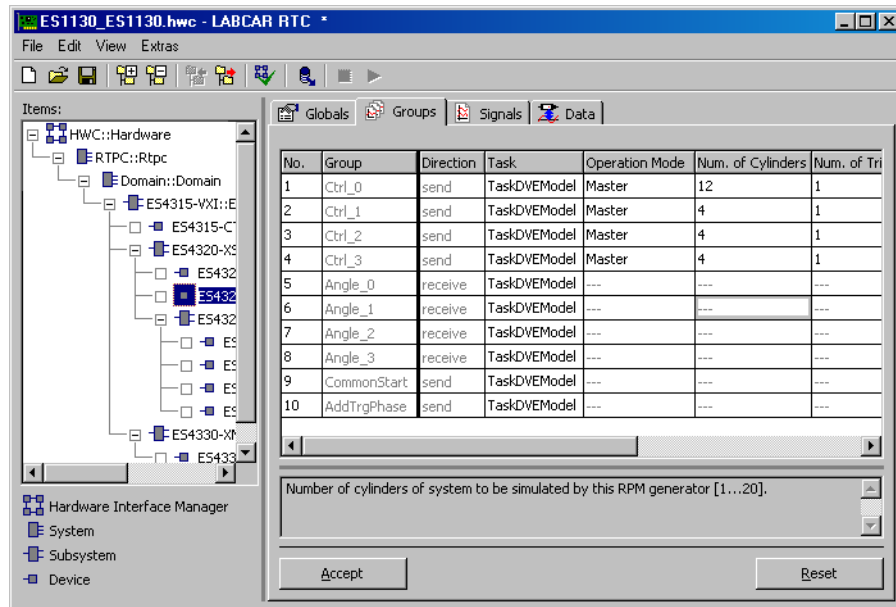
Bitte beachten Sie, dass die Nullmarke des RPM-Generators nicht mit einer einzelnen RPM-Triggermarke zusammenfallen muss, da zwischen Nullmarke und erstem Trigger noch eine Phasenverschiebung spezifiziert sein kann (Signale „TrgPhase\_<X>“ in Signalgruppe „Ctrl\_<X>“ des RPM Devices).

#### **Hinweis**

*Die Erzeugung einer zusätzlichen Triggermarke wird abgeschaltet, wenn ein KNOCK-Ctrl Item eingefügt wird, welches diesselbe ID besitzt (siehe Abschnitt „ES4320-KNOCK-Ctrl Device“ auf Seite 515)*

### 17.3.9 Groups (ES4320-RPM Device)

Das ES4320-RPM Device implementiert Signalgruppen zur Steuerung und zur Rückgabe der Winkelposition der vier RPM-Generatoren. Weiterhin sind in dieser Registerkarte die generatorspezifischen Konfigurationseinstellungen angeordnet.



**Abb. 17-10** Die Registerkarte „Groups“ des ES4320-RPM Devices

#### *Ctrl\_0 ... Ctrl\_3*

Mit diesen Send-Signalgruppen werden die RPM-Generatoren #0 ... #3 gesteuert.

#### *Angle\_0 ... Angle\_3*

Diese Receive-Signalgruppen stellen die aktuelle Drehwinkelinformation der RPM-Generatoren #0 ... #3 bereit.

#### *CommonStart*

Diese Send-Signalgruppe ermöglicht es, dass mehrere RPM-Generatoren synchron gestartet werden können.

#### *AddTrgPhase*

Diese Send-Signalgruppe ermöglicht es, die Phasenverschiebungen der vier zusätzlichen Triggermarken einzustellen. Den Signalgruppen Ctrl\_0 ... Ctrl\_3 sind Einstelloptionen zugeordnet. Damit können die den RPM-Generatoren #0 ... #3 zugeordneten Konfigurationsdaten eingestellt werden.

#### *Task*

Hier wird die Aktivierungszeit der RPM-Generatoren zwischen Hostsystem und der ES4320 definiert.



### Operation Mode

---

Hier wird definiert, ob der entsprechende RPM-Generator als Master-Generator fungieren soll oder durch ein externes RPM-Signal synchronisiert wird.

Einstellwert	Beschreibung
Master	Der RPM-Generator generiert das RPM-Signal selbstständig.
Slave of RPM #0 RPM #1 RPM #2 RPM #3	Der RPM-Generator übernimmt das RPM-Signal des RPM-Generators #0 RPM-Generators #1 RPM-Generators #2 RPM-Generators #3
Slave of VXI-RPM #0 VXI-RPM #1	Der RPM-Generator übernimmt das RPM-Signal #0 der ES4300 VXI-Backplane RPM-Signal #1 der ES4300 VXI-Backplane
Slave of Sync #0 Slave of Sync #1	Der RPM-Generator übernimmt das RPM-Signal, das über den Sync0-Steckverbinder der Frontplatte eingespeist wird.* Der RPM-Generator übernimmt das RPM-Signal, das über den Sync1-Steckverbinder der Frontplatte eingespeist wird.*

\* Hierzu muss im ESXS4320-CTRL Device der Parameter „Sync #n Configuration“ auf Input konfiguriert werden (siehe Abschnitt „Globals (ES4320-CTRL Device)“ auf Seite 478).

### Num. of Cylinders

---

Dieser Parameter legt die Anzahl der Zylinder fest. Dieser Parameter wird für die Klopfsignalgenerierung verwendet.

### Num. of Triggers

---

Mit dieser Option können Sie die Anzahl der Triggermarken definieren, welche pro Periode des RPM-Signals (720 °KW) generiert werden (1...16). Bei mehr als einer Triggermarke sind diese winkelläquidistant über die Periode des RPM-Signals verteilt. Für „Num. of Triggers“ kann nur ein Wert eingestellt werden, dessen ganzzahliges Vielfaches „Period Length“ ergibt.

### Period Length

---

Mit dieser Option können Sie die Anzahl der Takte festlegen, welche pro Periode des RPM-Signals generiert werden. Für „Period Length“ kann nur ein ganzzahliges Vielfaches von „Num. of Triggers“ eingestellt werden.

### Disable Source

Mit dieser Option können Sie wählen, ob einer der vier Disable-Eingänge Dis\_0 ... Dis\_3 an der Frontplatte für die Aktivierungssteuerung des RPM-Kanals verwendet werden soll. Die Wirkungsweise des Disable-Eingangs auf den RPM-Kanal ist von der Einstellung der Option „Disable Mode“ abhängig.

Einstellwert	Beschreibung
Deactive	Der RPM-Generator ist unabhängig von Zustand der Disable-Eingänge ständig aktiv.
Dis_0	Eingang Dis_0 steuert den RPM-Generator
Dis_1	Eingang Dis_1 steuert den RPM-Generator
Dis_2	Eingang Dis_2 steuert den RPM-Generator
Dis_3	Eingang Dis_3 steuert den RPM-Generator

### Disable Mode

Mit dieser Option können Sie festlegen, in welcher Art und Weise der festgelegte Disable-Eingang auf den RPM-Generator wirkt. Die Einstellung hat keine Auswirkung, falls bei der Option „Disable Source“ der Wert „Deactive“ eingestellt wird.

Einstellwert	Beschreibung
Cont. Low Active	Das Signal am Disable-Eingang wird TTL-Low-aktiv mit dem Takt des RPM-Generators verknüpft. D.h. der RPM-Generator ist bei TTL-High-Pegel angehalten.
Cont. High Active	Das Signal am Disable-Eingang wird TTL-High-aktiv mit dem Takt des RPM-Generators verknüpft. D.h. der RPM-Generator ist bei TTL-Low-Pegel angehalten.

### Hinweis

*Bei asynchronem Start wird der Slave erst mit dem nächsten Trigger des Masters wieder auf diesen synchronisiert.*

Der folgende RPM-Parameter ist im Slave-Modus unwirksam:

- EngSpeed\_*n*

Die folgenden RPM-Parameter müssen im Slave-Modus denen des Masters entsprechen:

- TrgEna\_*n*
- StartAngle\_*n*
- StartAngleEna\_*n*

Die folgenden RPM-Parameter können im Slave-Modus frei verwendet werden:

- StartMode\_*n*
- StartFlag
- TrgPhase\_*n*

*Datentypen und Wertebereiche*

Signal	Datentypen	Mögliche Werte
Num of Cylinders	uint8	2...20
Num of Triggers	uint8	1...16
Period Length	uint8	2...16384 (Anzahl der Takte je Kurbelwellenumdrehung)
Disable Source*	uint8	siehe „Disable Source“ auf Seite 486
Disable Mode	uint8	siehe „Disable Mode“ auf Seite 486
* Sobald ein RPM-Kanal als Slave konfiguriert ist, hat diese Einstellung keine Auswirkung. „Disable Source“ ist nur am Master-RPM möglich.		

## 17.3.10 Signals (ES4320-RPM Device)

In diesem Abschnitt werden die Signale des ES4320-RPM Items beschrieben. Die RPM-Generator-spezifischen Steuersignale sind für jeden RPM-Generator jeweils getrennt vorhanden und mit dem Suffix `_0`, `_1`, `_2` bzw. `_3` versehen.

Signal	Wirkungsrichtung	Beschreibung
StartMode_0 ... StartMode_3	send	Definiert, ob Start des RPM-Generators vom lokalen oder vom gemeinsamen (globalen) Startflag abhängig ist.
StartFlag_0 ... StartFlag_3	send	Lokales Startflag für betreffenden RPM-Kanal.
EngSpeed_0 ... EngSpeed_3	send	Vorgabewert der Winkelgeschwindigkeit des RPM-Generators.
TrgPhase_0 ... TrgPhase_3	send	Vorgabewert für Phasenverschiebung des ersten Triggerimpulses.
StartAngle_0 ... StartAngle_3	send	Vorgabewert für Startwinkel des RPM-Generators. Dies ist der Winkelwert, bei welchem der Generator starten bzw. anhalten kann. Die Funktion ist nur wirksam, falls zum Start- bzw. Stop-Zeitpunkt das betreffende „StartAngleEna_<X>“ Flag aktiviert ist. Ist dies nicht der Fall, so erfolgt Start bzw. Stop beim Winkelwert 0.0 °.

Signal	Wirkungs- richtung	Beschreibung
TrgEna_0 ... TrgEna_3	send	Mit diesem Signal können einzelne Triggermarken des RPM-Signals ausgeblendet werden. Bit 0 entspricht dabei der ersten Triggermarke usw. Die Anzahl der Triggermarken pro Periode wird durch die Option „Num. of Triggers“ in der Registerkarte „Globals“ konfiguriert.
StartAngleEna_0 ... StartAngleEna_3	send	Flag zur Aktivierung des vorgegebenen Startwinkels beim Starten / Stoppen des RPM-Generators #0 ... #3.
ActAngle_0 ... ActAngle_3	receive	Aktuelle Drehwinkelposition des RPM-Generators #0 ... #3.
StartFlag	send	Globales Startflag für synchronen Start mehrerer RPM-Generatoren.
Phase_0 ... Phase_3	send	Wert der Phasenverschiebung der zusätzlichen Triggermarken gegenüber dem zugrunde liegenden RPM-Trigger. Die zusätzlichen Triggermarken können als Trigger für SIG-ARB Kanäle verwendet werden.

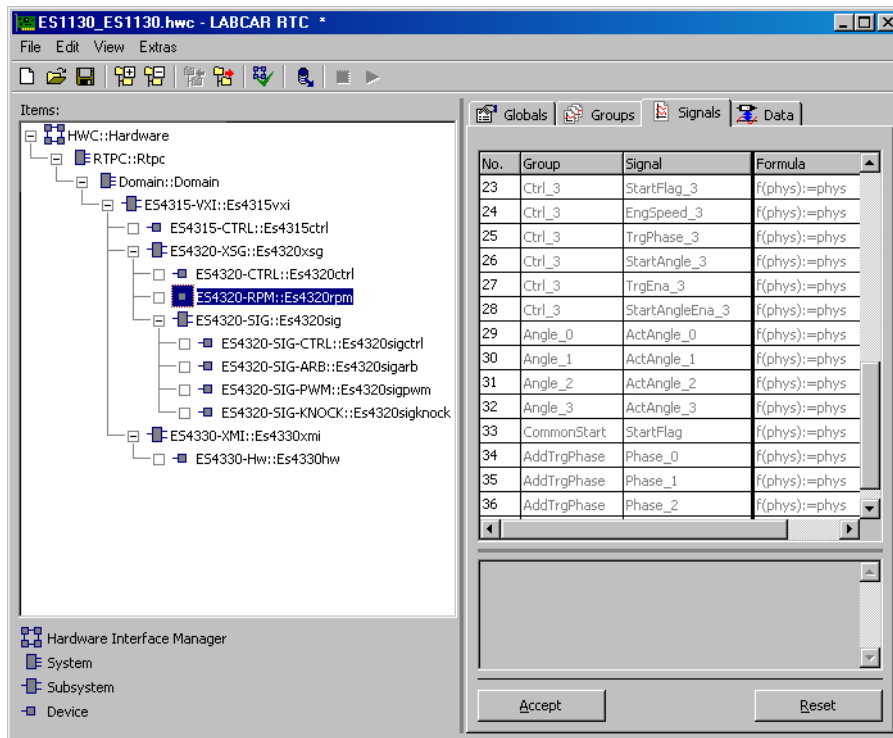


Abb. 17-11 Die Registerkarte „Signals“ des ES4320-RPM Devices

### 17.3.11 Datentypen und Wertebereiche

Signal	Datentypen	Mögliche Werte
StartMode_0 ... StartMode_3	uint8 / bool	0 [false]= local: Start-/Stop-Steuerung erfolgt durch lokales Startflag 1 [true] = global: Start-/Stop-Steuerung erfolgt durch globales Startflag
StartFlag_0 ... StartFlag_3	uint8 / bool	0 [false] = RPM-Generator wird angehalten 1 [true] = RPM-Generator wird gestartet
EngSpeed_0 ... EngSpeed_3	real32	(0 ... 1,44 x10 <sup>8</sup> ) / Period Length [1/min]
TrgPhase_0 ... TrgPhase_3	real32	0 ... 720.0° KW
StartAngle_0 ... StartAngle_3	real32	0 ... 720.0° KW
TrgEna_0 ... TrgEna_3	uint16	0x0001 ... 0xFFFF:
StartAngleEna_0 ... StartAngleEna_3	uint8 / bool	0 [false] = Start-/Stop-Winkel = 0.0 °KW 1 [true] = eingestellter Start-/Stop-Winkel aktiv
ActAngle_0 ... ActAngle_3	real32	0 ... 720.0° KW
StartFlag	uint8 / bool	0 [false] = betroffene RPM-Generatoren werden angehalten 1 [true] = betroffene RPM-Generatoren werden gestartet
Phase_0 ... Phase_3	real32	0 ... 720.0° KW

### 17.3.12 ES4320-SIG Subsystem

Das ES4320-SIG Subsystem bietet keine Einstellmöglichkeiten. Es dient vielmehr als Container für Items, welche die zwölf Ausgangskanäle der ES4320-Karte beschreiben. Jeder der zwölf Kanäle kann entweder als PWM- oder als Arbiträrkanal benutzt werden. Bei Letzterem werden die Werte der Hüllkurve mittels einer Tabelle vorgegeben. Somit können beliebige analoge Kurvenverläufe erzeugt werden.

### 17.3.13 ES4320-SIG-CTRL Device

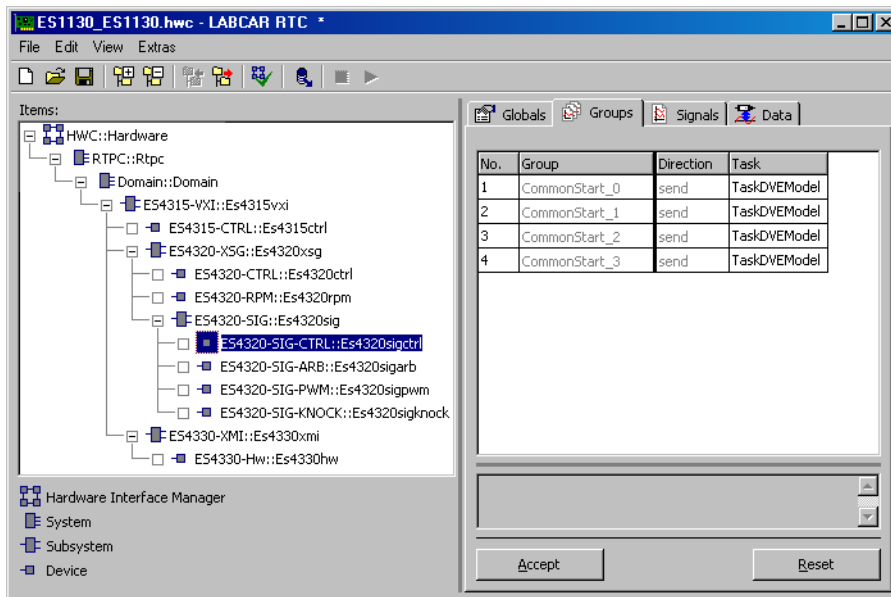
In diesem Abschnitt werden die Signalgruppen des ES4320-SIG-CTRL Devices beschrieben. Diese Signalgruppen können Auswirkungen auf alle zwölf Kanäle haben, unabhängig davon, ob diese als PWM- oder Arbiträrkanal konfiguriert sind.

### 17.3.14 Groups (ES4320-SIG-CTRL Device)

Das ES4320-SIG-CTRL Device implementiert vier Signalgruppen, welche zum synchronen Start mehrerer Ausgangskanäle verwendet werden können. Jede der Signalgruppen wirkt dabei auf jeweils drei Ausgangskanäle. Allerdings dürfen

die drei Ausgangskanäle nicht mit unterschiedlichen Funktionen konfiguriert sein. Dies bedeutet, dass PWM- und Arbiträrfunktion in der Dreiergruppe nicht gemischt verwendet werden dürfen. Es ist jedoch möglich, einen oder zwei Kanäle nicht zu verwenden, d. h. unkonfiguriert zu lassen.

Falls diese Voraussetzung nicht erfüllt ist, hat die entsprechende Signalgruppe „CommonStart“ keine Auswirkung.



**Abb. 17-12** Die Registerkarte „Groups“ des ES4320-SIG-CTRL Devices

#### *CommonStart\_0*

Mit dieser Send-Signalgruppe können die Kanäle mit der ID 0, 1, 2 synchron gestartet werden.

#### *CommonStart\_1*

Mit dieser Send-Signalgruppe können die Kanäle mit der ID 3, 4, 5 synchron gestartet werden.

#### *CommonStart\_2*

Mit dieser Send-Signalgruppe können die Kanäle mit der ID 6, 7, 8 synchron gestartet werden.

#### *CommonStart\_3*

Mit dieser Send-Signalgruppe können die Kanäle mit der ID 9, 10, 11 synchron gestartet werden.

17.3.15 Signals (ES4320-SIG-CTRL Device)

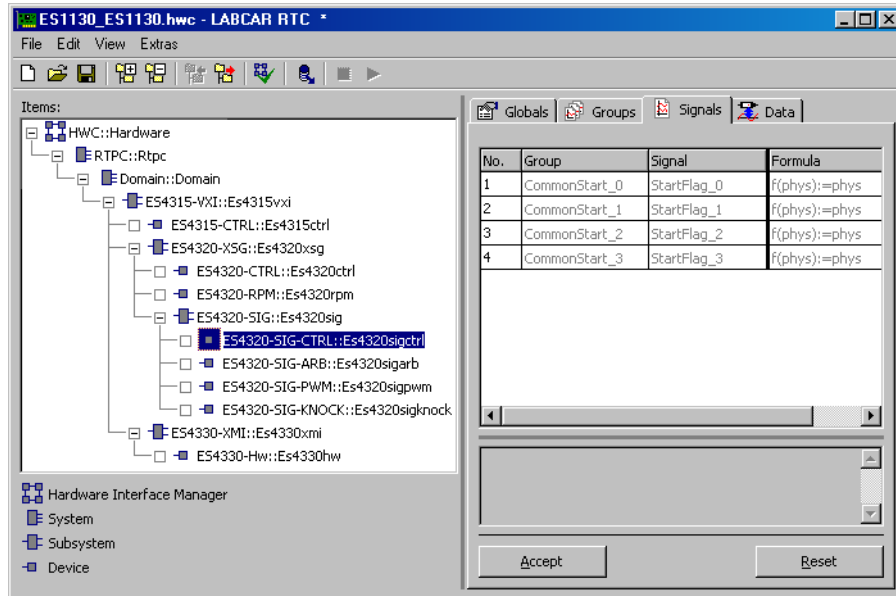


Abb. 17-13 Die Registerkarte „Signals“ des ES4320-SIG-CTRL Devices

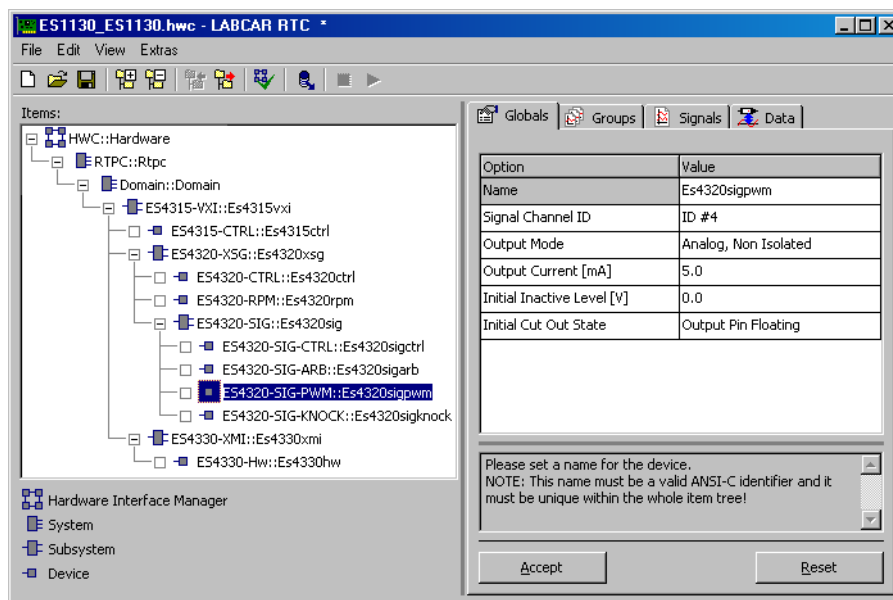
Signal	Wirkungs- richtung	Beschreibung
StartFlag_0	send	Globales Startflag für synchronen Start der Kanäle mit der ID 0, 1, 2.
StartFlag_1	send	Globales Startflag für synchronen Start der Kanäle mit der ID 3, 4, 5.
StartFlag_2	send	Globales Startflag für synchronen Start der Kanäle mit der ID 6, 7, 8.
StartFlag_3	send	Globales Startflag für synchronen Start der Kanäle mit der ID 9, 10, 11.

17.3.16 Datentypen und Wertebereiche

Signal	Datentypen	Mögliche Werte
StartFlag_0	uint / bool	0 [false] = betroffene ARB- oder PWM-Kanäle werden angehalten
...		
StartFlag_3		1 [true] = betroffene ARB- oder PWM-Kanäle werden gestartet

## 17.3.17 Globals (ES4320-SIG-PWM Device)

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten und die Signalgruppe des ES4320-SIG-PWM Device beschrieben. Dieses Device implementiert einen PWM-Ausgang unter Verwendung eines Hardwarekanals.



**Abb. 17-14** Die Registerkarte „Globals“ des ES4320-SIG-PWM Devices

Folgende Konfigurationseinstellungen sind in der Registerkarte „Globals“ möglich:

#### *Signal Channel ID*

Mit dieser Option können Sie festlegen, welcher der Kanäle 0 ... 11 für die PWM Funktion verwendet werden soll. Ein Kanal kann nicht mehrfach verwendet werden. Weiterhin kann der synchrone Start von Kanälen nur jeweils für die Kanäle 0, 1, 2 bzw. 3, 4, 5 bzw. 6, 7, 8 bzw. 9, 10, 11 erfolgen.

#### *Output Mode*

Mit dieser Option können Sie festlegen, welche Ausgangsbeschaltung für den Kanal benutzt werden soll. Es sind zwei analoge Modi (mit und ohne galvanischer Trennung) vorgesehen. Alle anderen Modi resultieren in einem binären Ausgangspegel, wobei der Binärwert mittels eines Schmitt-Triggers mit TTL-Eingangspannung gebildet wird. Dies sind entweder Pull-Down- (Open Collector) oder Pull-Down-/ Pull-Up-Modi mit Bezug auf eine Batteriespannung (Batterieknoten 0 ... 7 oder UBatt A bzw. UBatt B).

#### *Output Current*

Mit dieser Option können Sie den Wert der Ausgangsstrombegrenzung im Bereich von 5 ... 15 mA vorgeben. Dieser Wert bezeichnet den maximalen Strom im Pull-Up-Betrieb bei binären Modi (nicht Open Collector). Der Pull-Down-Strom ist bei den binären Ausgangsmodi auf 100 mA begrenzt.



### *Initial Inactive Level*

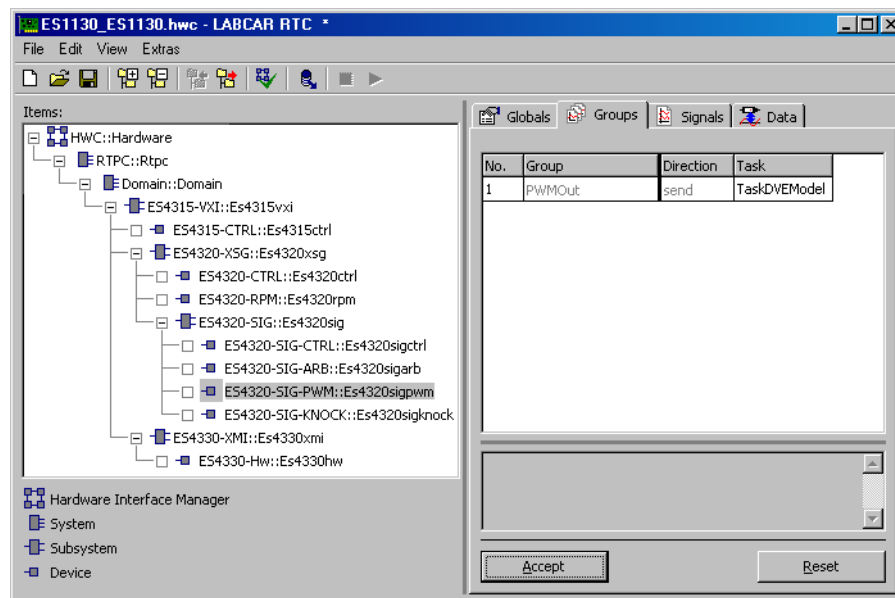
Mit dieser Option können Sie den Initialisierungswert des Low- bzw. Inaktiv-Pegels des Ausgangs festlegen. Dieser Pegel wird unmittelbar nach der Initialisierung wirksam, noch bevor Pegelwerte mittels der Signalgruppe „PWMOut“ vorgegeben werden. Bei binären Ausgangsmodi (siehe „Output Mode“) ergibt sich ein Zustand entsprechend der Schaltschwelle des Schmitt-Triggers.

### *Initial Cut Out State*

Mit dieser Option können Sie einen definierten Initialisierungszustand des Unterbrechungsrelais vorgeben. Der Zustand wird unmittelbar nach der Initialisierung eingestellt, noch bevor durch die Signalgruppe „PWMOut“ ein anderer Wert für das Relais vorgegeben wird.

## 17.3.18 Groups (ES4320-SIG-PWM Device)

Das Device implementiert eine Send-Signalgruppe „PWMOut“ zur Steuerung des PWM-Kanals.



**Abb. 17-15** Die Registerkarte „Groups“ des ES4320-SIG-PWM Devices

## 17.3.19 Signals (ES4320-SIG-PWM Device)

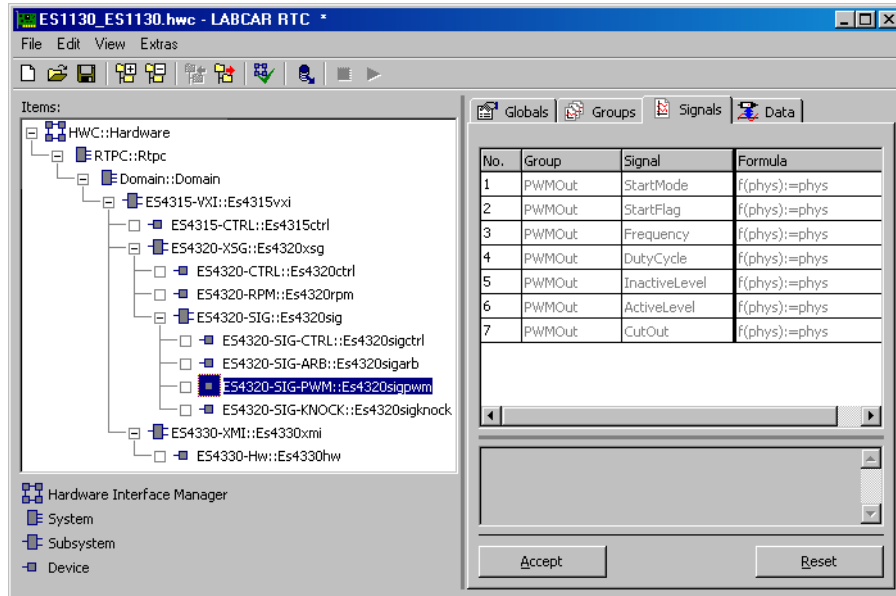


Abb. 17-16 Die Registerkarte „Signals“ des ES4320-SIG-PWM Devices

Signal	Wirkungs- richtung	Beschreibung
StartMode	send	Definiert, ob der Start des PWM-Kanals vom lokalen oder einem gemeinsamen (globalen) Startflag abhängig ist.
StartFlag	send	Lokales Startflag für diesen PWM-Kanal.
Frequency	send	Frequenz des PWM-Signals
DutyCycle	send	Tastgrad des PWM-Signals
InactiveLevel	send	Spannungswert für den Inaktiv (Low) Zustand des PWM-Signals
ActiveLevel	send	Spannungswert für den Aktiv (High) Zustand des PWM-Signals
CutOut	send	Zustand des Unterbrechungsrelais am Ausgang

17.3.20 Datentypen und Wertebereiche

Signal	Datentypen	Mögliche Werte
StartMode	uint8 / bool	0 [local] = PWM-Kanal wird vom lokalen StartFlag gesteuert 1 [global] = PWM-Kanal wird vom zugehörigen globalen StartFlag gesteuert
StartFlag	uint8 / bool	0 [false] = PWM-Kanal angehalten 1 [true] = PWM-Kanal gestartet
Frequency	real32	0.0 ... 10000.0 Hz
DutyCycle	real32	0.0 ... 1.0
InactiveLevel	real32	0.0 ... 10.0 Volt
ActiveLevel	real32	0.0 ... 10.0 Volt
CutOut	uint8 / bool	0 [false] = Unterbrechungsrelais geöffnet 1 [true] = Unterbrechungsrelais geschlossen

17.3.21 Globals (ES4320-SIG-ARB Device)

In diesem Abschnitt werden die spezifischen Einstellmöglichkeiten und die Signalgruppe des ES4320-SIG-ARB Device beschrieben. Dieses Device ermöglicht es, unter Verwendung eines HW Kanals ein arbiträres Ausgangssignal zu erzeugen.

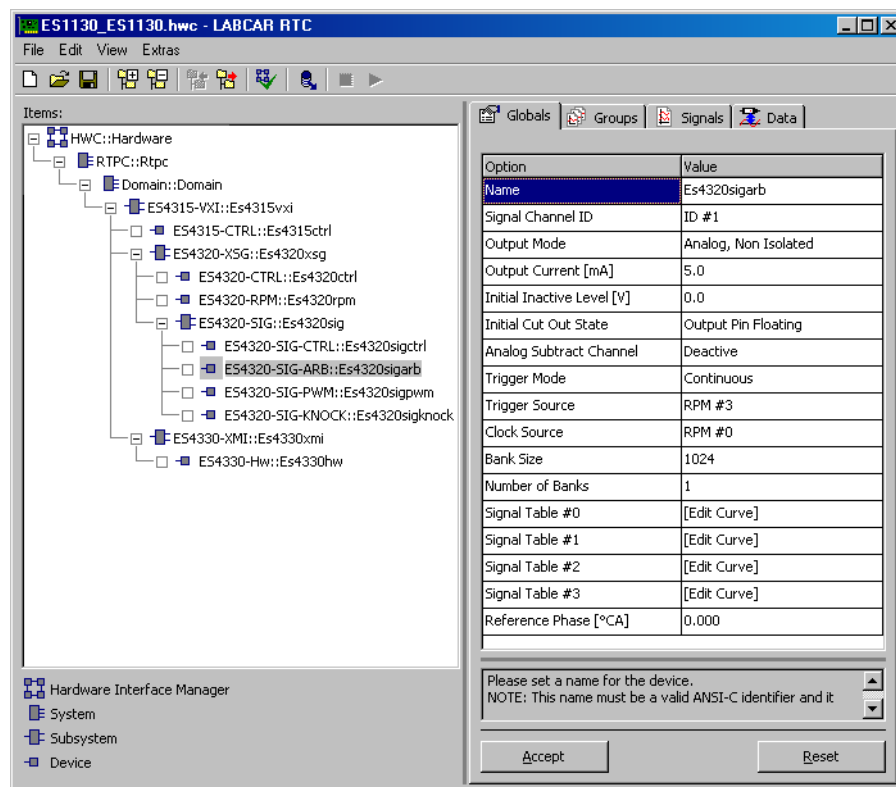


Abb. 17-17 Die Registerkarte „Globals“ des ES4320-SIG-ARB Devices

Folgende Konfigurationseinstellungen sind in der Registerkarte „Globals“ möglich:

#### *Signal Channel ID*

---

Mit dieser Option können Sie festlegen, mit welchem der Kanäle 0 ... 11 das arbiträre Ausgangssignal erzeugt werden soll. Ein Kanal kann nicht mehrfach verwendet werden. Als weitere Einschränkung kann der synchrone Start von Kanälen nur jeweils für die Kanäle 0, 1, 2 bzw. 3, 4, 5 bzw. 6, 7, 8 bzw. 9, 10, 11 erfolgen.

#### *Output Mode*

---

Mit dieser Option können Sie festlegen, welche Ausgangsbeschaltung für den Kanal benutzt werden soll.

Es sind zwei analoge Modi (mit und ohne galvanischer Trennung) vorgesehen. Alle anderen Modi resultieren in einem binären Ausgangspegel, wobei der Binärwert mittels eines Schmitt-Triggers mit TTL-Eingangspegel gebildet wird. Dies sind entweder Pull-Down- (Open Collector) oder Pull-Down- / Pull-Up Modi mit Bezug auf eine Batteriespannung (Batterieknoten 0 ... 7 oder UBatt A bzw. UBatt B).

<b>Einstellwert</b>	<b>Beschreibung</b>
Analog Non Isolated	Analoger Ausgangstreiber ohne galvanische Trennung. Vorgabe einer externen Referenzspannung möglich.
Analog Isolated	Analoger Ausgangstreiber, galvanisch getrennt
Open Collector	Open Collector mit Pull-Down-Strombegrenzung von 100 mA
Pull-Up U-Batt-A Pull-Up U-Batt-B	Wie Open Collector, zusätzlich Pull-Up nach UBatt A bzw. UBatt B mit einstellbarer Strombegrenzung
Pull-Up Node #0 ... Pull-Up Node #7	Wie Open Collector, zusätzlich Pull-Up nach Batterieknoten #0 bis #7 mit einstellbarer Strombegrenzung

#### *Output Current*

---

Mit dieser Option können Sie den Wert der Ausgangsstrombegrenzung im Bereich von 5 ... 15 mA vorgeben. Dieser Wert bezeichnet den maximalen Strom im Pull-Up-Betrieb bei binären Modi (nicht Open Collector). Der Pull-Down-Strom ist bei den binären Ausgangsmodi auf 100 mA begrenzt.

#### *Initial Inactive Level*

---

Mit dieser Option können Sie den Initialisierungswert des Low- bzw. Inaktiv-Pegels des Ausgangs festlegen. Dieser Pegel wird unmittelbar nach der Initialisierung wirksam, noch bevor Pegelwerte mittels der Signalgruppe „ArbOut“ vorgegeben werden. Bei binären Ausgangsmodi (siehe „Output Mode“) ergibt sich ein Zustand entsprechend der Schaltschwelle des Schmitt-Triggers.

### *Initial Cut Out State*

---

Mit dieser Option können Sie einen definierten Initialisierungszustand des Unterbrechungsrelais vorgeben. Der Zustand wird unmittelbar nach der Initialisierung eingestellt, noch bevor durch die Signalgruppe „ArbOut“ ein anderer Wert für das Relais vorgegeben wird.

### *Analog Subtract Channel*

---

Mit dieser Option können Sie festlegen, ob ein bzw. welcher Kanal von dem aktuellen Kanal subtrahiert werden soll. Es wird dabei der Spannungswert des festgelegten Kanals analog-seitig subtrahiert. Ist ein binärer Ausgangsmode eingestellt, so ergibt sich aus der Spannungsdifferenz ein Zustand am Ausgang gemäss der Schaltschwelle des Schmitt-Triggers. Soll keine Subtraktion des Spannungswerts eines anderen Kanals erfolgen, so muss der aktuelle Kanal bzw. der Eintrag „Deactive“ eingestellt werden.

### *Trigger Mode*

---

Mit dieser Option können Sie festlegen, wie sich das arbiträre Ausgangssignal verhält, wenn die Signalbank weniger Stützstellen enthält als zwischen zwei RPM-Triggermarken Taktimpulse bereitgestellt werden. Der Signalbankinhalt ist dann vor Auftreten der nächsten RPM-Triggermarke bereits vollständig ausgetaktet.

Beim Einstellwert „Continuous“ wird nach der letzten Stützstelle der Signalbankinhalt fortlaufend zyklisch ausgegeben. Beim Einstellwert „Wait for Trigger“ hingegen bleibt der Wert der letzten Stützstelle am Ausgang stehen. Bei der nächsten Triggermarke des RPM-Signals wird unabhängig vom Vorzustand mit dem erneuten Austakten der ausgewählten Signalbank begonnen.

### **Hinweis**

*Bitte beachten Sie, dass im Triggermodus „Continuous“ keine Phasenverschiebung des Arbiträrsignals gegenüber der Triggermarke eingestellt werden kann. Die Einstelloption Reference Phase und das Signal PhaseRef der Signalgruppe „ArbOut“ in diesem Device haben daher keine Wirkung!*

*Trigger Source*

Mit dieser Option können Sie festlegen, welche Signalquelle als Trigger benutzt werden soll. Triggerpunkt ist die steigende Flanke des ausgewählten Signals. Folgende Triggerquellen sind möglich:

<b>Einstellwert</b>	<b>Beschreibung</b>
RPM #0 ... RPM #3	Es wird das Triggersignal des RPM-Generators 0 ... 3 dieser ES4320-Karte verwendet.
Sync #0, Sync #1	Es wird das Triggersignal des an der Frontplatte eingespeisten RPM-Signals SYNC0 bzw. SYNC1 verwendet. Diese Einstellung ist nur möglich, wenn SYNC0 bzw. SYNC1 als Eingänge konfiguriert sind.
VXI-RPM #0, VXI-RPM #1	Es wird das Triggersignal der über VXI verfügbaren RPM-Ressource VXI RPM #0 bzw. VXI RPM #1 verwendet. Diese Ressourcen können von einem RPM-Generator einer ES4320-Karte des VXI-Systems gespeist werden. Die diesbezügliche Einstellung erfolgt im ES4315-CTRL Device.
Add. Trigger #0 ... Add. Trigger #3	Es wird eine der im ES4320-RPM Device definierten zusätzlichen Triggermarken Add. Trigger #0 ... Add. Trigger #3 als Triggerquelle benutzt.
Front Panel Trigger #0 ... Front Panel Trigger #3	Es wird einer der vier TRIG-Eingänge der Frontplatte als Triggersignal benutzt. Diese Signale müssen von einer externen Quelle eingespeist werden (High-aktiv, TTL-Pegel).
SIG-Channel #0 ... SIG-Channel #11	Es wird der binäre Ausgangswert eines PWM- oder Arbiträrkanals dieser ES4320-Karte als Triggersignal benutzt. Es kann nur ein Kanal aus der jeweiligen Sechsergruppe des Arbiträrkanals ausgewählt werden. Benutzt z. B. dieser Arbiträrkanal die Ausgangskanal-ID #1, so können die Ausgangskanäle ID #0, #2, #3, #4 oder #5 als Triggerquelle benutzt werden.

### Clock Source

---

Mit dieser Option können Sie vorgeben, welche Signalquelle als Takt zum Aus-takten der Signalbänke verwendet werden soll. Folgende Taktquellen stehen zur Verfügung:

Einstellwert	Beschreibung
RPM #0 ... RPM #3	Es wird das Taktsignal des RPM-Generators 0 ... 3 verwendet.
Sync #0, Sync #1	Es wird das Taktsignal des an der Frontplatte eingespeisten RPM-Signals Sync #0 bzw. Sync #1 verwendet. Diese Einstellung ist nur möglich, wenn SYNC0 bzw. SYNC1 als Eingänge konfiguriert sind.
VXI-RPM #0, VXI-RPM #1	Es wird das Triggersignal der über VXI verfügbaren RPM-Res-source VXI RPM #0 bzw. VXI RPM #1 verwendet. Diese Res-sourcen können von einem RPM-Generator einer ES4320-Karte des VXI Systems gespeist werden. Die diesbezügliche Einstellung erfolgt im ES4315-CTRL Device.
Local	Es wird die durch das Signal „LocalClockFreq“ der Signal-gruppe „ArbOut“ in diesem Device definierte Taktfrequenz verwendet. Es wird kein Trigger verwendet. Der mit der Option „Trigger Source“ hierfür eingestellte Wert bleibt ohne Wirkung.
1 MHz	Es wird ein fester Takt von 1 MHz verwendet. Es wird kein Trigger verwendet. Der mit der Option „Trigger Source“ hier-für eingestellte Wert bleibt ohne Wirkung.

### Bank Size

---

Mit dieser Option können Sie die Anzahl der Stützstellen einer einzelnen Bank vorgeben. Dies entspricht dann einer Periode des Arbiträrsignals. Werden Takt und Trigger durch einen RPM-Generator vorgegeben, so muss der Wert von „Bank Size“ des Arbiträrkanals mit dem Wert von „Period Length“ des benutz-ten RPM-Generators übereinstimmen.

### Number of Banks

---

Mit dieser Option können Sie die Anzahl der Bänke für diesen Arbiträrkanal ein-stellen. Es können prinzipiell eine, zwei oder vier Bänke benutzt werden - bei einer „Bank Size“ bis 8192 maximal 4 Bänke, bei einer „Bank Size“ bis 16384 maximal 2 Bänke.

Es kann jedoch immer nur eine Bank zu einem Zeitpunkt aktiv geschaltet sein, d.h. den Spannungsverlauf des Ausgangssignals bestimmen. Mit dem Signal „BankSelect“ in der Send-Signalgruppe „ArbOut“ dieses Items wird die aktuell gültige Bank ausgewählt.

### Signal Table #0 ... Signal Table #3

---

Mit dieser Option können Sie die Stützstellen der Signalbankinhalte definieren. Durch Klicken auf das Eingabefeld "[Edit Curve]" öffnet sich ein Tabelleneditor. Sie können nun die Anzahl der Stützstellen verändern und die Stützstellen und Datenwerte verändern. Die X Stützstellen müssen monoton steigen und es kön-nen maximal 1024 Stützstellen definiert werden. Der absolute Wertebereich der

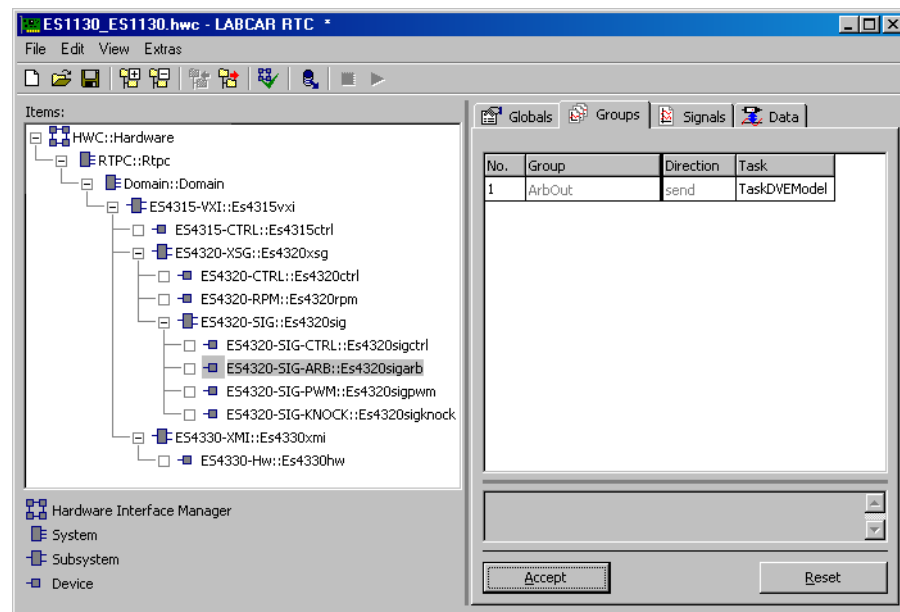
X- bzw. Z-Werte spielt dabei keine Rolle. Vielmehr werden bei der Berechnung der auszugebenden Signalbänke die Z-Werte zwischen 0 und 1 und die X-Werte auf eine Periode des Arbiträrsignals normiert. Somit brauchen nur diejenigen Stützstellen des Signalverlaufes spezifiziert werden, bei welchen sich das Signal ändert.

#### Reference Phase

Mit dieser Option können Sie eine zusätzliche Phasenverschiebung zwischen dem RPM-Trigger und dem Signalverlauf der gewählten Signalbank vorgeben. Der Gesamtwert der Phasenverschiebung ergibt sich aus dem hier eingestellten Wert und dem Wert des Signals „PhaseRef“ der Signalgruppe „ArbOut“ in diesem Item. Der hier eingestellte Wert kann sich im Bereich zwischen  $-720,0^\circ$  ...  $+720,0^\circ$  bewegen. Diese Option ist nur beim Trigger-Modus „Wait for Trigger“ wirksam. Beim Triggermodus „Continuous“ ist der hier eingestellte Wert bedeutungslos.

#### 17.3.22 Groups (ES4320-SIG-ARB Device)

Das Device implementiert ein Send-Signalgruppe „ArbOut“ zur Steuerung des Abiträrkanals.



**Abb. 17-18** Die Registerkarte „Groups“ des ES4320-SIG-ARB Devices



17.3.23 Signals (ES4320-SIG-ARB Device)

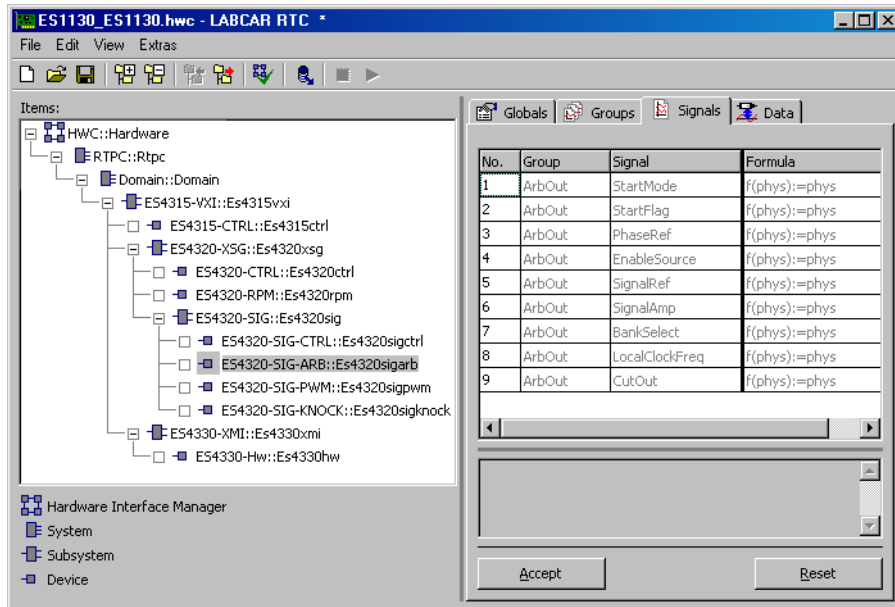


Abb. 17-19 Die Registerkarte „Signals“ des ES4320-SIG-ARB Devices

Signal	Wirkungs- richtung	Beschreibung
StartMode	send	Definiert, ob Start des Arbiträrkanal vom lokalen oder einem gemeinsamen (globalen) Startflag abhängig ist.
StartFlag	send	Lokales Startflag für diesen Arbiträrkanal
PhaseRef	send	Phasenverschiebung zwischen RPM-Triggermarke und Start der Signaltabelle. Nur wirksam bei Triggermodus „Wait for Trigger“. Eine Änderung dieser Größe wird sofort, d. h. bei der Berechnung der nächsten auszu-taktenden Stützstelle wirksam.
EnableSource	send	Mit diesem Signal kann der Takt gesteuert werden. Der Takt kann hierbei unbedingt ein- bzw. auszuschaltet werden oder er kann mit dem binären Ausgangswert eines anderen Kanals verknüpft werden. Dies ist jedoch nur mit einem anderen Kanal aus der entsprechenden Sechsergruppe (ID #0 ... ID #5 bzw. ID #6 ... ID #11) möglich.
SignalRef	send	Legt fest, ob als Referenzspannung der interne Vorgabewert „SignalAmp“ oder die extern angelegte Referenzspannung benutzt wird. Die externe Referenzspannung ist dabei immer auf die Analogmasse der Karte bezogen und damit nicht potentialfrei.

Signal	Wirkungsrichtung	Beschreibung
SignalAmp	send	Vorgabewert für die interne Referenzspannung. Damit ergibt sich der Ausgangsspannungsbereich von 0 Volt bis zu diesem Spannungswert.
BankSelect	send	Selektor, welcher das unmittelbare Umschalten zwischen den Signalbänken während des laufenden Betriebs ermöglicht.
LocalClockFreq	send	Frequenz des lokalen Taktes. Diese wird zum Austakten des Signalbankinhaltes benutzt, falls bei der Option „Clock Source“ der Wert „local“ eingestellt ist.
CutOut	send	Zustand des Unterbrechungsrelais am Ausgang des Kanals

### 17.3.24 Datentypen und Wertebereiche

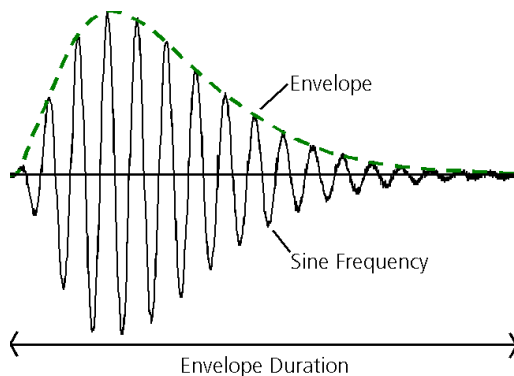
Signal	Datentypen	Mögliche Werte
StartMode	uint8/bool	0 [local] = Arbiträrkanal wird vom lokalen StartFlag gesteuert. 1 [global] = Arbiträrkanal wird vom zugehörigen globalen StartFlag gesteuert.
StartFlag	uint8/bool	0 [false] = Arbiträrkanal angehalten. 1 [true] = Arbiträrkanal gestartet.
PhaseRef	real32	-720,0 ° ... 720,0 ° KW
EnableSource	uint8	0 [off] = 1...6 [channel dependent] = 8 [unconditional] =
SignalRef	uint8/bool	0 [intern] = interne, einstellbare Referenzspannung wird benutzt 1 [extern] = externe Referenzspannung wird benutzt
SignalAmp	real32	0 ... 10,0 Volt
BankSelect	uint8	0 = Ausgabe von Signalbank 0 1 = Ausgabe von Signalbank 1 2 = Ausgabe von Signalbank 2 3 = Ausgabe von Signalbank 3
LocalClockFreq	real32	0,0 ... 1,0 MHz
CutOut	uint8/bool	0 [false] = Unterbrechungsrelais geöffnet 1 [true] = Unterbrechungsrelais geschlossen

### 17.3.25 ES4320-SIG-KNOCK Device

Der Klopfgenerator wird zur Simulation von Signalen von Klopfensoren verwendet. Eine unkontrollierte Verbrennung des Luft-Kraftstoff-Gemisches führt zu Druckschwankungen, die durch Körperschallsensoren aufgenommen werden. Das Motorsteuergerät kann Klopfen innerhalb eines bestimmten Frequenzbereiches (abhängig von der Amplitude der Schwingungen) erfassen.

Klopfen kann für jeden einzelnen Zylinder separat generiert werden, entweder nach einem Zufallsmuster oder nach einem festgelegten Muster. Dauer, Signalform und die Amplitudenmodulation (Hüllkurve) des Klopfsignals können frei definiert werden.

Abb. 17-20 zeigt ein Beispiel eines Klopfsignals („Klopfpaket“). Normalerweise ist dies ein hochfrequentes Sinussignal, das mit einer sich langsam verändernder Einhüllenden moduliert ist.



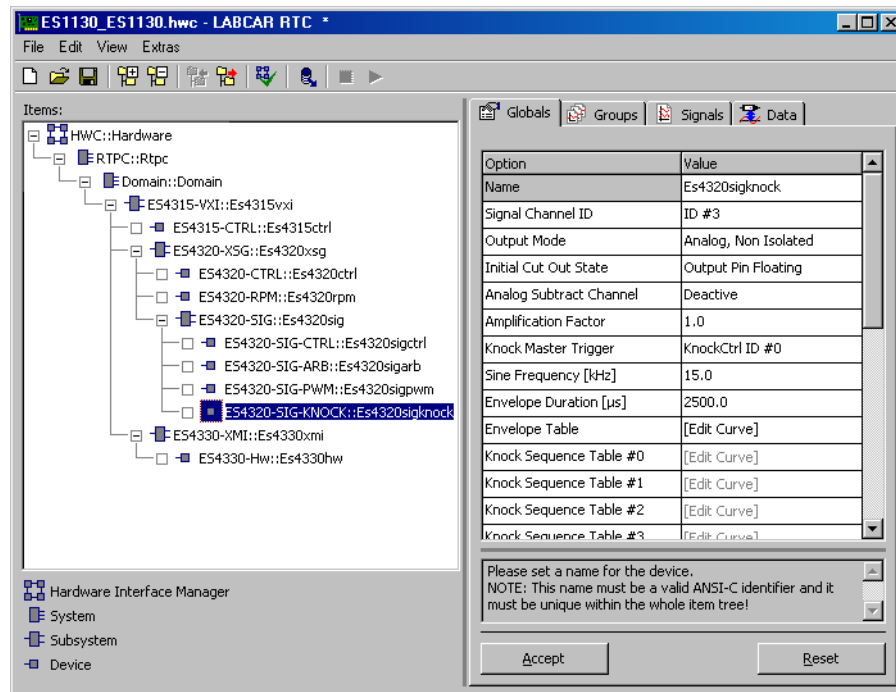
**Abb. 17-20** Klopfsignal (durchgezogen) und Einhüllende (gestrichelt)

#### **Hinweis**

Der Klopfgenerator benötigt die Angabe der Anzahl der vorhandenen Zylinder. Die Einstellung der Zylinderanzahl erfolgt in dem Register „Groups“ des ES4320-RPM Devices (siehe „Groups (ES4320-RPM Device)“ auf Seite 484).

## 17.3.26 Globals (ES4320-SIG-KNOCK Device)

In diesem Abschnitt werden die globalen Einstellungen des ES4320-SIG-KNOCK Devices beschrieben.



**Abb. 17-21** Die Registerkarte „Globals“ des ES4320-SIG-KNOCK Devices

### *Signal Channel ID*

Mit dieser Option können Sie festlegen, welcher der Kanäle 0..11 für die Klopf-signalgenerierung verwendet werden soll. Ein Kanal kann nicht mehrfach verwendet werden. Maximal vier Kanäle der ES4320 können für die Klopf-signalgenerierung verwendet werden.

Dieser Parameter ist nicht editierbar im Laufzeitmodus.

### *Output Mode*

Mit dieser Option können Sie festlegen, welche Ausgangsbeschaltung für den Kanal verwendet wird. Es sind zwei analoge Modi (mit und ohne galvanischer Trennung) vorgesehen.

### *Initial Cut Out State*

Mit dieser Option können Sie einen definierten Initialisierungszustand des Unterbrechungsrelais vorgeben. Der Zustand wird unmittelbar nach der Initialisierung eingestellt, noch bevor durch die Signalgruppe „KnockOut“ ein anderer Wert für das Relais vorgegeben wird.

### *Analog Subtract Channel*

Mit dieser Option können Sie festlegen, ob ein bzw. welcher Kanal von dem aktuellen Klopf-generator-Kanal subtrahiert werden soll. Es wird dabei der Spannungswert des festgelegten Kanals analog-seitig subtrahiert. Ist ein binärer Ausgangsmode eingestellt, so ergibt sich aus der Spannungsdifferenz ein Zustand

am Ausgang gemäss der Schaltschwelle des Schmitt-Triggers. Soll keine Subtraktion des Spannungswerts eines anderen Kanals erfolgen, so muss der aktuelle Kanal bzw. der Eintrag „Deactive“ eingestellt werden.

#### *Amplification Factor*

---

Klopfsignale haben eine geringe Amplitude und sind damit bei der Übertragung zum Steuergerät sehr anfällig gegenüber Störungen. Um dies zu vermeiden, werden die Signale bei der Ausgabe an den Trennadapter verstärkt. Ein gewöhnlich im Trennadapter installierter Abschwächer kompensiert diese Verstärkung zur Weitergabe an das Steuergerät. Der „Amplification Factor“ muss damit gleich dem Abschwächungsfaktor des Abschwächers des Trennadapters gewählt werden.

Der Parameter „Amplification Factor“ ist im Laufzeitmodus editierbar.

#### *Knock Master Trigger*

---

Mit dieser Option können Sie festlegen, über welchen Klopf-Master-Trigger der Klopf-Signalgenerator angesteuert wird. Dabei muss zwischen KnockCtrl ID #0 ... #3 ausgewählt werden.

Die Zuordnung zwischen „KnockControl ID“ und RPM-Generator wird im ES4320-KNOCK-Ctrl Item definiert. Alle Einstellungen, die im ES4320-RPM Device definiert sind (wie z.B. Anzahl der Zylinder oder „Period Length“), werden für die Klopfsignalgenerierung verwendet. In Abhängigkeit von der Zylinderanzahl wird unter „ES4320-RPM Device - Globals“ (siehe Seite 483) der entsprechende „Add. Trigger Point Ref.“ verwendet, um an den entsprechenden Zylinderpositionen Triggermarken zu erzeugen, an denen die Klopfsignalgenerierung aktiviert wird. Aus diesem Grund wird der Parameter „Add. Trigger Point Ref.“ bei Verwendung des RPMs für die Klopfsignalgenerierung deaktiviert (d.h. es ist keine Einstellung durch den Anwender möglich).

#### *Sine Frequency*

---

Mit dieser Option können Sie festlegen, mit welcher Sinus-Grundfrequenz Klopfsignale generiert werden. Diese Einstellung wirkt sich auf alle Zylinder aus, die im ES4320-SIG-KNOCK Device definiert sind.

Das resultierende Klopfsignal ergibt sich aus der Sinus-Grundschiwingung multipliziert mit der Dauer und Kurvenform der Umhüllenden.

#### *Envelope Duration [μs]*

---

Mit dieser Option können Sie festlegen, wie lange das Klopfpaket ausgegeben wird.

Ein Klopfsignal für einen Zylinder (siehe auch Abb. 17-20 auf Seite 503) wird definiert durch die Kurvenform der Umhüllenden (definiert durch „Envelope Table“), die Dauer der Umhüllenden (definiert durch „Envelope Duration“) und der Sinus-Grundfrequenz des Klopfsignals (definiert durch „Sine Frequency“).

#### *Envelope Table*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Definition der Hüllkurve und der Dauer des Klopfsignals. Die x-Achse spezifiziert die Zeit in μs. Der x-Wert des ersten Punktes muss 0.0 sein. Die Dauer des Klopfsignals

muss größer oder gleich der Periode des Klopfsignals sein. Die z-Achse spezifiziert die (normalisierte) Form der Hüllkurve - die z-Werte müssen zwischen 0.0 und 1.0 liegen.

Die Gleichung der Defaulthüllkurve lautet:

$$y = 25.0 * e^{2.0} * x^{2.0} * e^{(-10 x)} \quad \text{wobei } 25.0 * e^{2.0} = 184.7264025$$

Der Parameter „Envelope Table“ ist im Laufzeitmodus editierbar.

#### *Knock Sequence Table #0...#19*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor, mit dem interaktiv das Muster für das Auftreten der Klopfsignale festgelegt werden kann. Ein Muster ist jeweils 100 Zündvorgänge lang. Die x-Achse hat keine Bedeutung und wird vernachlässigt. Ein z-Wert von 1 (Eins) bedeutet, dass für die aktuelle Zündung klopfen simuliert werden soll, während eine 0 (Null) für „kein Klopfen“ steht.

Die Parameter „Knock Sequence Table #n“ sind im Laufzeitmodus editierbar.

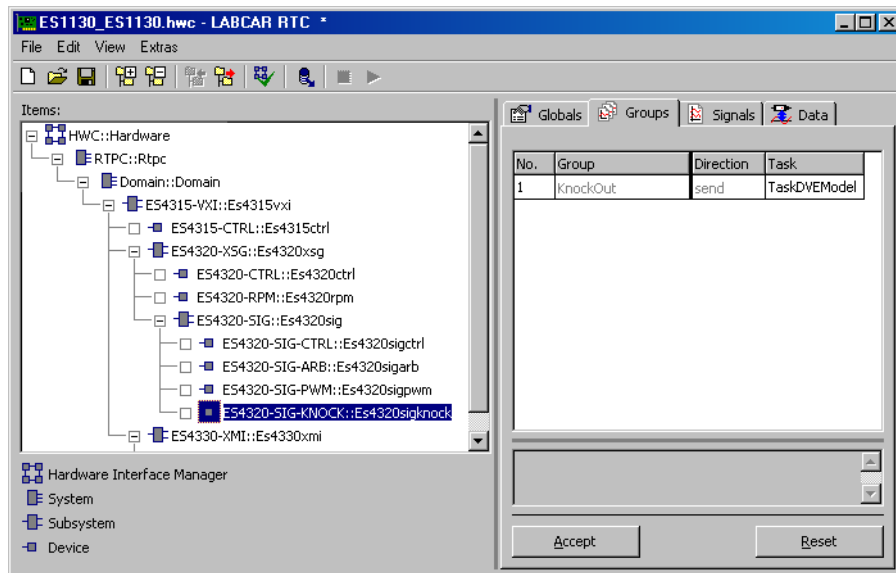
In Tab. 17-1 sind die Eigenschaften der globalen RTIO-Parameters zusammengefasst.

RTIO Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Signal Channel ID	uint32	Nein	Kanal, der zur Klopfsignal-erzeugung verwendet wird. Gültige Werte: 0..11
Output Mode	uint8	Ja	Ausgangsbeschaltung für den Kanal: - Analog ohne galv. Trennung - Analog mit galv. Trennung
Initial Cut Out State	uint8	Ja	Initialisierungszustand des Unterbrecherrelais
Analog Subtract Channel	uint8	Ja	Definition des Kanals, der vom aktuellen Kanal abgezogen werden soll.
Amplification Factor	real32	Ja	Verstärkungsfaktor zur Kompensation der externen Signalabschwächung. Defaultwert: 1.0 (keine externe Abschwächung vorhanden) Gültiger Bereich: 0.1 ... 10.0
Knock Master Trigger	uint8	Nein	KnockCtrl ID #0 ... #3
Sine Frequency	real32	Ja	Frequenz der Sinus-Grundschiwingung des Klopfsignals in kHz. Defaultwert: 15 kHz Gültiger Bereich: 1 kHz - 20 kHz
Envelope Duration [ $\mu$ s]	real32	Ja	Dauer der Klopfsignalumhüllenden Defaultwert: 2500 $\mu$ s Gültiger Bereich: 2 $\mu$ s - 12000 $\mu$ s
Envelope Table	1-dim. Table	Ja	Normalisierte Hüllkurve des Klopfsignals x-Achse: Nr. der Stützstelle Bereich für z-Werte: 0.0 ... 1.0
Knock Sequence Table #0 ... Knock Sequence Table #19	1-dim. Table	Ja	Klopfmuster. Jedes Muster ist 100 Zündvorgänge lang. Ein z-Wert von 1 bedeutet, dass für die aktuelle Zündung Klopfen simuliert wird. Ein z-Wert von 0 bedeutet „kein Klopfen“ für die aktuelle Zündung. x-Werte werden ignoriert.

**Tab. 17-1** Globale Parameter des ES4320-SIG-KNOCK Devices

### 17.3.27 Groups (ES4320-SIG-KNOCK Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des ES4320-SIG-KNOCK Devices beschrieben.



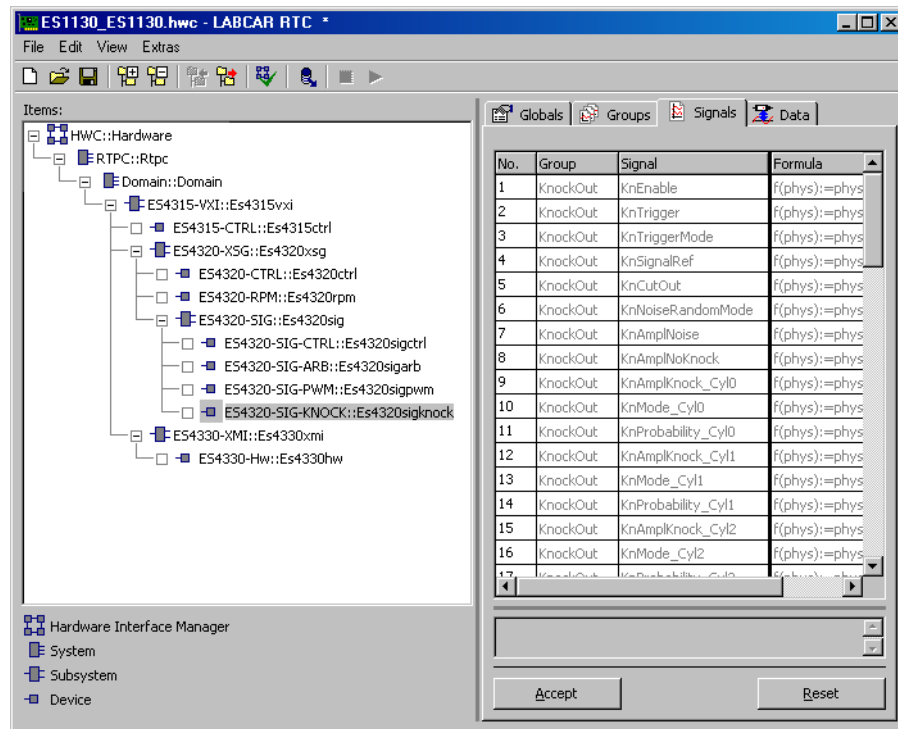
**Abb. 17-22** Die Registerkarte „Groups“ des ES4320-SIG-KNOCK Devices  
*KnockOut*

KnockOut ist die Task, in der das Klopfen ausgegeben wird.



### 17.3.28 Signals (ES4320-SIG-KNOCK Device)

In diesem Abschnitt werden die Signale beschrieben, mit denen der Klopfgenerator gesteuert wird.



**Abb. 17-23** Die Registerkarte „Signals“ des ES4320-SIG-KNOCK Devices

#### *KnEnable*

Das RTIO-Signal „KnEnable“ wird verwendet, um den Klopfgenerator an- und auszuschalten. Ein Signalwert von 0 schaltet den Klopfgenerator ab, ein Signalwert von 1 schaltet den Klopfgenerator an.

#### *KnTrigger*

Ein 0 → 1-Übergang des RTIO-Signals „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder).

Das Signal „KnTrigger“ ist nur aktiv, wenn der Klopfgenerator im Triggermodus „Single Shot“ betrieben wird. Der Triggermodus wird mit dem RTIO-Signal „KnTriggerMode“ eingestellt (siehe Seite 509).

#### *KnTriggerMode*

Das RTIO-Signal „KnTriggerMode“ wird zur Steuerung des Triggermodus des Klopfgenerators verwendet.

Wenn das Signal auf 0 gesetzt wird, arbeitet der Klopfgenerator im Triggermodus „Single Shot“. Ein 0 → 1-Übergang des RTIO-Signals „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder). Danach unterbleibt die Klopfgenerierung bis zum Auftreten eines erneuten Triggervorganges.

Wenn das Signal auf 1 gesetzt wird, arbeitet der Klopfgenerator im Modus „Continuous“. In diesem Modus ist die Klopfsimulation ohne eine Steuerung durch einen Trigger permanent aktiv. In diesem Triggermodus ist das RTIO-Signal „KnTrigger“ nicht aktiv.

#### *KnSignalRef*

---

Das RTIO-Signal „KnSignalRef“ legen die Quelle der Referenzspannung des D/A-Wandlers des Klopfgenerators fest. Wenn der Wert auf 0 gesetzt ist, wird die interne Referenzspannung (10 V) gewählt. Ist der Wert auf 1 gesetzt, wird die Referenzspannung des D/A-Wandlers extern eingespeist.

#### **Hinweis**

*Wenn Sie die interne Referenzspannung verwenden, beziehen sich die RTIO Signalamplituden "KnAmplNoKnock", "KnAmplKnock\_Cyln" und "KnAmplNoise" auf eine Referenzspannung von 10 V - die Signalwerte sind auf einen Bereich von (0.0 ... 10.0)/Amplification Factor beschränkt.*

*Wenn Sie eine externe Referenzspannung verwenden, geben die RTIO Signalamplituden "KnAmplNoKnock", "KnAmplKnock\_Cyln" und "KnAmplNoise" einen Faktor an, der mit der externen Referenzspannung multipliziert wird. Die Signalwerte sind auf einen Bereich von (0.0 ... 1.0)/Amplification Factor beschränkt.*

*Die externe Referenzspannung ist dabei immer auf die Analogmasse der Karte bezogen und damit nicht potentialfrei!*

#### *KnCutOut*

---

Das Signal „KnCutOut“ dient zum An- und Abschalten des Ausgangskanals. Ein Signalwert von 0 schaltet den Ausgang des Kanals ab, ein Signalwert von 1 schaltet den Ausgang des Kanals an.

#### *KnNoiseRandomMode*

---

Das Signal „KnNoiseRandom Mode“ legt fest, ob die Modulation der Klopfsignale bei einer Motorumdrehung für jeden Zylinder einzeln (= 0) oder ein Muster für alle Zylinder (= 1) erzeugt wird.

#### *KnAmplNoise*

---

Viele Steuergeräte interpretieren ein mehrfaches Auftreten von Klopfsignalen derselben Amplitude als Sensorstörung. Um dies zu verhindern, können die simulierten Klopfsignale über einen Zufallsgenerator amplitudenmoduliert werden.

Das RTIO-Signal „KnAmplNoise“ legt die Amplitude fest, mit der das Klopfsignal nach dem Zufallsprinzip moduliert wird. Dieses Signal wird verwendet, um statistischen Schwankungen bei der Verbrennung und damit bei den Signalen des Klopfensors zu simulieren.

Die Formel für das Klopfsignal lautet für den Modus „Random“:

$$A(x) = (\text{KnAmplKnock\_Cyln} - \text{KnAmplNoKnock}) * \text{KnProbability} + \text{KnAmplNoKnock} + \text{KnAmplNoise} * \text{Random Value}[0..1]$$

Für den Modus „Sequence“:

$$A(x) = (\text{KnAmplKnock\_Cyl}n - \text{KnAmplNoKnock}) * \text{KnockSequenceTable} \#n + \text{KnAmplNoKnock} + \text{KnAmplNoise} * \text{RandomValue}[0..1]$$

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

*KnAmplNoKnock*

---

Amplitude des Klopfsignals bei kontrollierter Verbrennung. Das RTIO-Signal „KnAmplNoKnock“ legt die Amplitude des Klopfsignals fest, wenn für die aktuelle Zündung kein Klopfsignal generiert werden soll.

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

*KnAmplKnock\_Cyl0 ... KnAmplKnock\_Cyl19*

---

Amplitude des Klopfsignals bei unkontrollierter Verbrennung. Die RTIO-Signale „KnAmplKnock\_Cyl $n$ “ legen die Amplitude des Klopfsignals für Zylinder  $n$  fest, wenn für die aktuelle Zündung ein Klopfsignal generiert werden soll.

In Abhängigkeit von der Klopfwahrscheinlichkeit „KnProbability\_Cyl $n$ “ wird aus beiden Signalen die Klopfspannung generiert.

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

*KnMode\_Cyl0 ... KnMode\_Cyl19*

---

Die RTIO-Signale „KnMode\_Cyl0“ bis „KnMode\_Cyl19“ legen die Betriebsart für die Klopfgenerierung im jeweiligen Zylinder fest.

- 0 = disabled  
Keine Klopfsignalgenerierung für den jeweiligen Zylinder.
- 1 = sequence  
Für die in „Knock Sequence Table # $n$ “ mit 1 definierten Motorumdrehungen wird ein Klopfsignal erzeugt.
- 2 = random  
Es wird bei jeder Motorumdrehung ein Klopfsignal generiert in Abhängigkeit von der Klopfwahrscheinlichkeit „KnProbability\_Cyl $n$ “.

*KnProbability\_Cyl0 ... KnProbability\_Cyl19*

---

Mit den RTIO-Signalen „KnProbability\_Cyl0“ bis „KnProbability\_Cyl19“ wird die Wahrscheinlichkeit festgelegt, dass während der aktuellen Zündung eine unkontrollierte Verbrennung auftritt. Diese Signale sind aber nur dann aktiv, wenn das entsprechende „KnMode\_Cyl $n$ “ RTIO-Signal auf „Random Mode“ gesetzt ist.

Der gültige Wertebereich geht von 0.0 bis 1.0. Ein Wert von 0.0 bedeutet, dass eine unkontrollierte Verbrennung nie auftritt, ein Wert von 1.0 bedeutet, dass es bei jeder Zündung im entsprechenden Zylinder zu einer unkontrollierten Verbrennung kommt.

**Hinweis**

*Die Indizes 0 bis 19 der RTIO-Signale „KnMode“ und „KnProbability“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*

Tab. 17-2 enthält eine Zusammenfassung aller Signale des ES4320-SIG-KNOCK Device.

RTIO-Signale	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
KnEnable	uint8 / bool	ja	An- und Abschalten des Klopfgenerators. 0: Abgeschaltet 1: Angeschaltet
KnTrigger	uint8 / bool	ja	Triggersignal für Klopfgenerator. Ein 0 → 1-Übergang des RTIO-Signal „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder). Nur aktiv im Triggermodus „single shot“.
KnTriggerMode	uint8 / bool	ja	Triggermodus des Klopfgenerators 0: Triggermodus „single shot“ 1: Triggermodus „continuous“.
KnSignalRef	uint8 / bool	ja	D/A-Wandler-Bezugsspannung des Klopfgeneratorsausgangs 0: Interne 10 V Referenzspannung 1: Externe Bezugsspannung
KnCutOut	uint8 / bool	ja	An- und Abschalten des jeweiligen Ausgangskanals 0: Ausgang abgeschaltet 1: Ausgang angeschaltet
KnNoiseRandom-Mode	uint8 / bool	ja	Art der Klopfmustererzeugung 0: für jeden Zylinder einzeln 1: ein Muster für alle Zylinder
KnAmplNoise	real32	ja	Amplitude (in Volt), mit der das Klopfsignal (per Zufallsgenerator) moduliert wird Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor
KnAmplNoKnock	real32	ja	Amplitude des Klopfsignals bei kontrollierter Verbrennung Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor

RTIO-Signale	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
KnAmpl-Knock_Cyl0 ... KnAmpl-Knock_Cyl19	real32	ja	Amplitude des Klopfsignals bei unkontrollierter Verbrennung in Zylinder n. Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor
KnMode_Cyl0 ... KnMode_Cyl19	uint8	ja	Betriebsart für die Klopfgenerierung im jeweiligen Zylinder. 0: Disabled 1: Pattern mode 2: Random mode
KnProbability_Cyl0 ... KnProbability_Cyl19	real32	ja	Wahrscheinlichkeit, dass während der aktuellen Zündung eine unkontrollierte Verbrennung auftritt. Nur aktiv im Modus „Random“. Gültige Werte: 0.0 (kein Klopfen) ... 1.0 (immer Klopfen)

**Tab. 17-2** Signale des ES4320-SIG-KNOCK Devices

### 17.3.29 ES4320-KNOCK-Ctrl Device

Um einen Klopfgenerator betreiben zu können, muss für dessen Triggern ein ES4320-KNOCK-Ctrl Device ausgewählt und konfiguriert werden. Auf der ES4320 stehen bis zu 4 unabhängige Knock-Control Einheiten zur Verfügung.

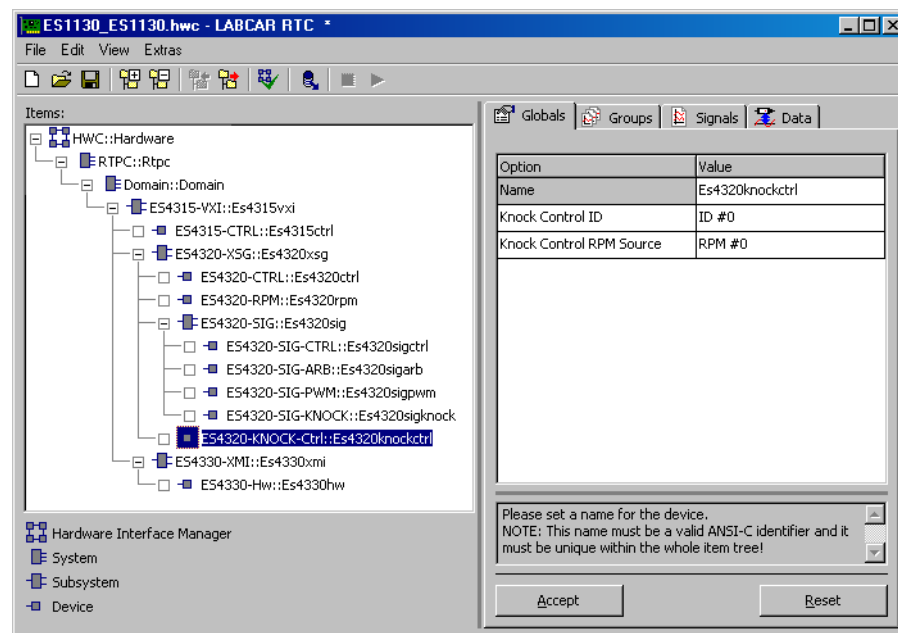
Für jeden Zylinder eines Motors, für den Klopfen simuliert werden soll, muss in der 720° KW-Ebene ein Triggersignal erzeugt werden. Das Triggersignal wird dem Klopfgenerator über eine der „Additional Trigger“-Leitungen zugeführt.

#### **Hinweis**

*Für die Triggersignalleitung mit der „KnockCtrl ID“ ID #n wird immer der Additional Trigger „Add. Trigger #n“ verwendet. Damit steht nach Einfügen eines KNOCK-Ctrl Devices mit der entsprechenden ID der zugehörige Additional Trigger nicht mehr für andere Zwecke zur Verfügung. Unter dem „Globals“ Register des ES4320-RPM Devices wird der entsprechende Eintrag dann deaktiviert.*

### 17.3.30 Globals (ES4320-KNOCK-Ctrl Device)

In diesem Abschnitt werden die globalen Einstellungen des ES4320-KNOCK-Ctrl Devices beschrieben.



**Abb. 17-24** Die Registerkarte „Globals“ des ES4320-KNOCK-Ctrl Devices  
*Knock Control ID*

Mit dieser Option können Sie festlegen, welche ID das KNOCK-Ctrl Device hat. Mit dem Parameter „Knock Master Trigger“ des ES4320-SIG-KNOCK Devices wird über diese ID die Referenz zu dem verwendeten RPM-Generator hergestellt. Mögliche Einstellungen sind: ID #0 ... ID #3.

Dieser Parameter ist nicht zur Laufzeit editierbar.

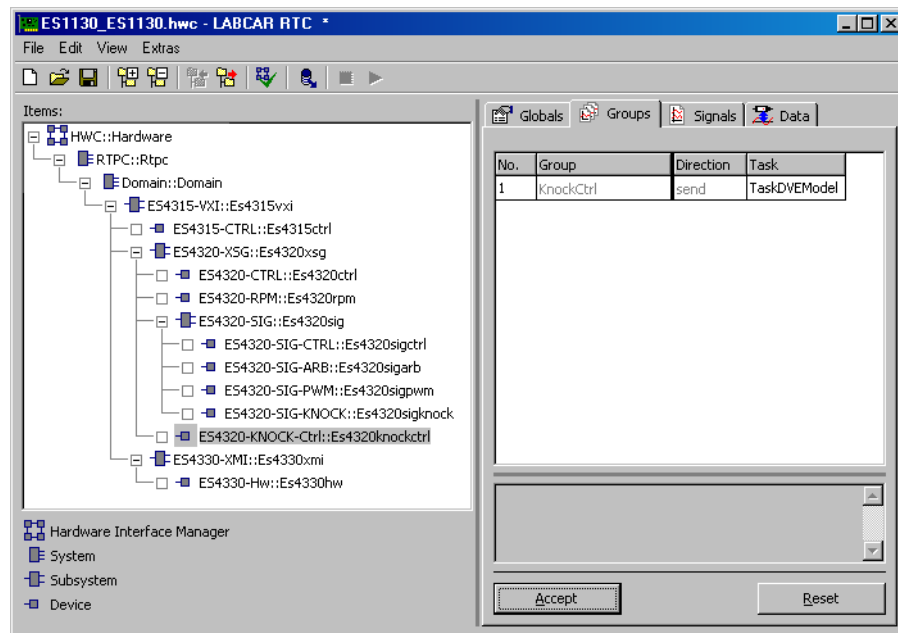
### Knock Control RPM Source

Mit dieser Option können Sie festlegen, welcher RPM-Generator für die Klopfsignalgenerierung verwendet werden soll. Dieser kann sowohl Master als auch Slave sein. Mögliche Einstellungen sind: RPM #0 ... RPM #3.

Dieser Parameter ist nicht zur Laufzeit editierbar.

#### 17.3.31 Groups (ES4320-KNOCK-Ctrl Device)

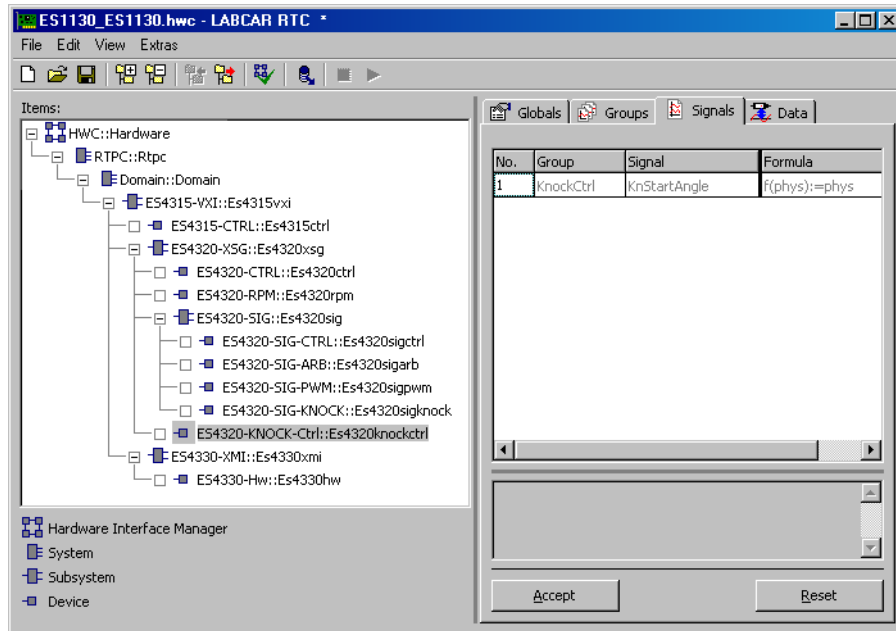
Das ES4320-KNOCK-Ctrl Device implementiert eine Signalgruppe „KnockCtrl“, welche für die Echtzeitkommunikation verwendet wird.



**Abb. 17-25** Die Registerkarte „Groups“ des ES4320-KNOCK-Ctrl Devices



17.3.32 Signals (ES4320-KNOCK-Ctrl Device)



**Abb. 17-26** Die Registerkarte „Signals“ des ES4320-KNOCK-Ctrl Devices *KnStartAngle*

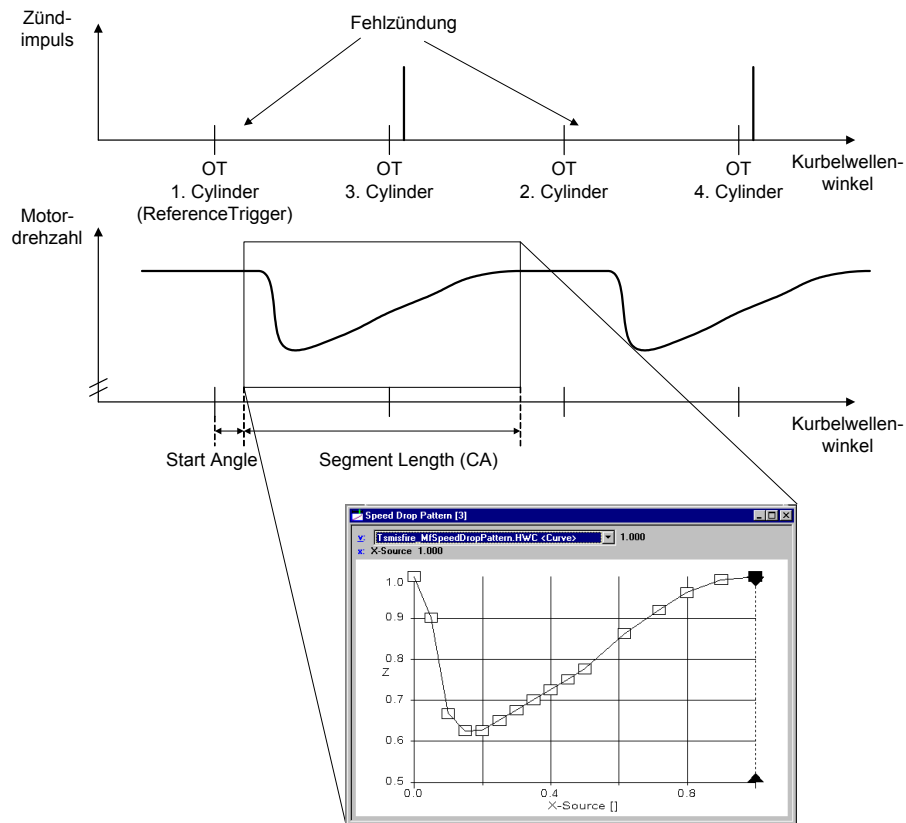
Dies ist der Vorgabewert für die Phasenverschiebung der Klopfsignalgenerierung.

Signal	Datentyp	Mögliche Werte
KnStartAngle	real32	-720° ... +720° KW

## 17.3.33 ES4320-Misfire Device

Der Fehlzündungsgenerator simuliert einen Abfall der Motordrehzahl nach einer Fehlzündung und ist somit für Steuergeräte geeignet, die eine Fehlzündung anhand des nachfolgenden Abfalls der Drehzahl identifizieren.

Abb. 17-27 zeigt den Zusammenhang zwischen Fehlzündung und dem Abfall der Motordrehzahl.



**Abb. 17-27** Fehlzündung und Abfall der Motordrehzahl

Die Fehlzündungen können für jeden einzelnen Zylinder simuliert werden - die maximale Zahl der Zylinder liegt bei 20. Die Fehlzündungen können per Zufalls-generator mit einer festgelegten Wahrscheinlichkeit oder nach einem vorgegebenen Muster generiert werden. Diese Muster haben eine maximale Länge von jeweils 100 Zündungen (pro Zylinder) und können beliebig oft wiederholt werden.

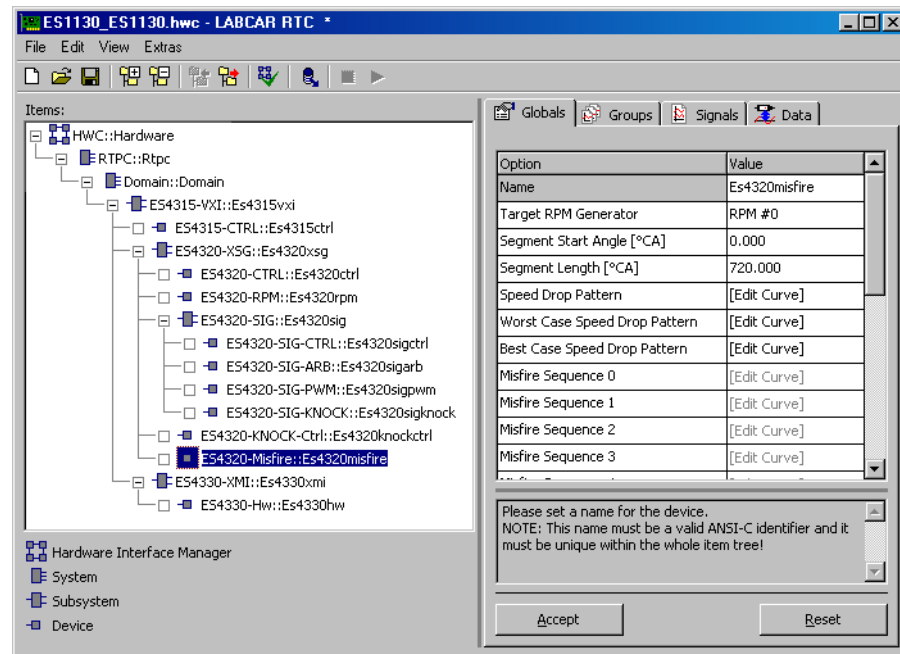
Es ist außerdem möglich, für jeden Zylinder einen Faktor anzugeben, der den Drehzahlabfall für jeden Zylinder einzeln bestimmt.

#### **Hinweis**

*Der Fehlzündungsgenerator benötigt die Angabe der Anzahl der vorhandenen Zylinder. Die Einstellung der Zylinderanzahl erfolgt in dem Register „Groups“ des ES4320-RPM Devices (siehe „Groups (ES4320-RPM Device)“ auf Seite 484).*

### 17.3.34 Globals (ES4320-Misfire Device)

In diesem Abschnitt werden die globalen Einstellungen des ES4320-Misfire Devices beschrieben.



**Abb. 17-28** Die Registerkarte „Globals“ des ES4320-Misfire Devices

#### *Target RPM Generator*

Der Parameter „Target RPM Generator“ wird verwendet, um den RPM-Generator festzulegen, auf dem Fehlzündungen simuliert werden sollen.

#### *Segment Start Angle [°CA]*

Der Parameter „Segment Start Angle“ wird verwendet, um den Winkel zwischen OT und dem Beginn des Abfalls der Motordrehzahl aufgrund der Fehlzündung festzulegen (siehe Abb. 17-27 auf Seite 518).

„Segment Start Angle“ wird in °KW angegeben und kann Werte zwischen -720.0 °KW und 720 °KW annehmen. Ein positiver Startwinkel bedeutet, dass der Beginn nach OT liegt.

Dieser Parameter ist im Laufzeitmodus editierbar.

#### *Segment Length [°CA]*

Der Parameter „Segment Length“ bestimmt die Länge (in °KW), über die der Abfall der Motordrehzahl anhält (siehe Abb. 17-27 auf Seite 518).

„Segment Length“ wird in °KW angegeben und kann Werte zwischen 0 °KW und 720 °KW annehmen.

Dieser Parameter ist im Laufzeitmodus editierbar.

#### *Speed Drop Pattern*

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Festlegung der Charakteristik des Drehzahlabfalls nach einer Fehlzündung.

Die x-Achse hat dabei keine Bedeutung - die Dauer des Drehzahlabfalls (in °KW) wird mit dem Parameter „Segment Length“ festgelegt (siehe Abb. 17-27 auf Seite 518).

Die z-Achse bestimmt den Abfall der Motordrehzahl. Der zulässige Wertebereich reicht von 0.0 bis 2.0. Ein Wert von 0.5 beispielsweise bedeutet, dass die Motordrehzahl halbiert wird.

Ganz allgemein wird die Motordrehzahl während der Fehlzündungssimulation dadurch festgelegt, dass die Drehzahl vor Auftreten der Fehlzündung mit dem Faktor für den Abfall der Drehzahl (z-Wert) multipliziert wird.

Die durch aufeinanderfolgenden Fehlzündungen hervorgerufenen Drehzahländerungen können auch überlappen, z. B. wenn bei einem 4-Zylinder-Motor gilt: „Segment Length“ > 180 °KW.

Die Tabelle „Speed Drop Pattern“ ist im Laufzeitmodus editierbar.

#### *Worst Case Speed Drop/Best Case Speed Drop*

---

Mit dieser Option legen Sie fest, mit welcher Charakteristik der Drehzahlabfall nach einer Fehlzündung erfolgen soll. In Verbindung mit „Best Case Speed Drop“ und „MfMode\_Cyln“ = 3 wird die resultierende Charakteristik berechnet.

$$\text{DropPatternValue} = \text{RandomValue} * \text{WorstCasePattern} + (1 - \text{RandomValue}) * \text{BestCasePattern}$$

#### *Misfire Sequence 0 ... Misfire Sequence 19*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Festlegung des Musters für das Auftreten der Fehlzündungen. Diese Muster sind jeweils 100 Zündvorgänge lang. Die x-Achse besitzt keine Bedeutung.

Ein z-Wert von 1 bedeutet, dass für den aktuellen Zündvorgang eine Fehlzündung simuliert werden soll, während ein z-Wert von 0 bedeutet, dass für den aktuellen Zündvorgang keine Fehlzündung simuliert werden soll.

Die Tabelle „Misfire Sequence nn“ sind im Laufzeitmodus editierbar.

#### **Hinweis**

*Die Indizes 0 bis 19 der RTIO-Parameter „Misfire Sequence“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*

In Tab. 17-3 sind die Eigenschaften der globalen Parameter des ES4320-Misfire Devices zusammengefasst.

RTIO Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Segment Start Angle	real32	Yes	Der Winkel zwischen OT (reference trigger) und dem Beginn des Drehzahlabfalls aufgrund der Fehlzündung. Gültiger Wertebereich: -720.0 °KW ... 720.0 °KW
Segment Length	real32	Yes	Dauer des Drehzahlabfalls (in °KW) nach der Fehlzündung Gültiger Wertebereich: 0 °KW ... 720.0 °KW
Speed Drop Pattern	1-dim. Table	Yes	Charakteristik des Drehzahlabfalls nach Fehlzündung z-Werte: 0.0 ... 2.0 Die x-Werte haben keine Bedeutung.
Worst Case Speed Drop / Best Case Speed Drop	1-dim. Table	Yes	Charakteristik des Drehzahlabfalls nach Fehlzündung z-Werte: 0.0 ... 2.0 Die x-Werte haben keine Bedeutung.
Misfire Sequence 0 ... Misfire Sequence 11	1-dim. Table	Yes	Fehlzündungsmuster. Jedes Muster ist 100 Zündvorgänge lang z-Wert = 1: Für den aktuellen Zündvorgang wird eine Fehlzündung simuliert z-Wert = 0: Für den aktuellen Zündvorgang wird keine Fehlzündung simuliert Die x-Werte haben keine Bedeutung.

**Tab. 17-3** Globale RTIO-Parameter des ES4320-Misfire Devices

17.3.35 Groups (ES4320-Misfire Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des ES4320-Misfire Devices beschrieben.

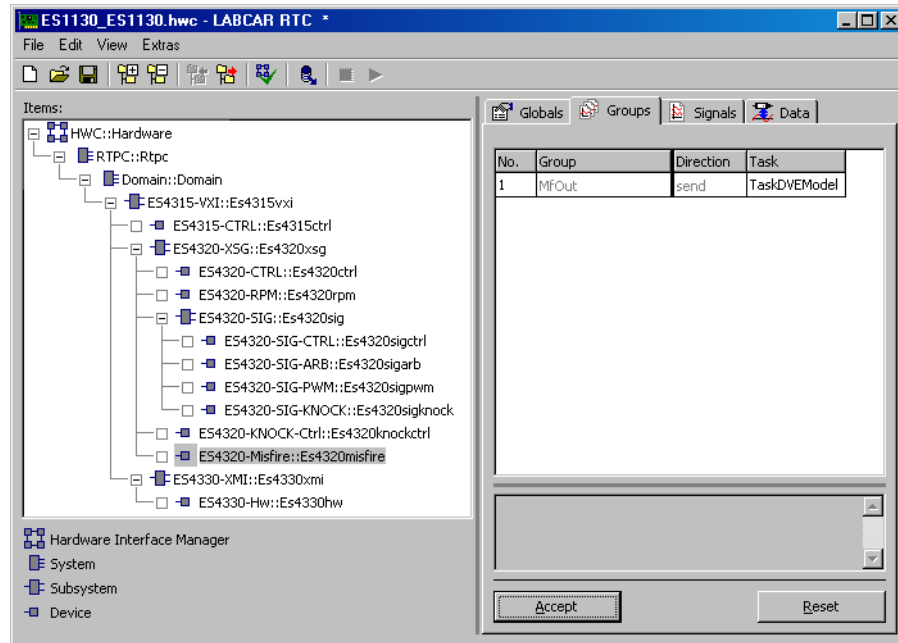


Abb. 17-29 Die Registerkarte „Groups“ des ES4320-Misfire Devices

17.3.36 Signals (ES4320-Misfire Device)

Die Registerkarte „Signals“ listet die einzelnen Signale des ES4320-Misfire Devices auf.

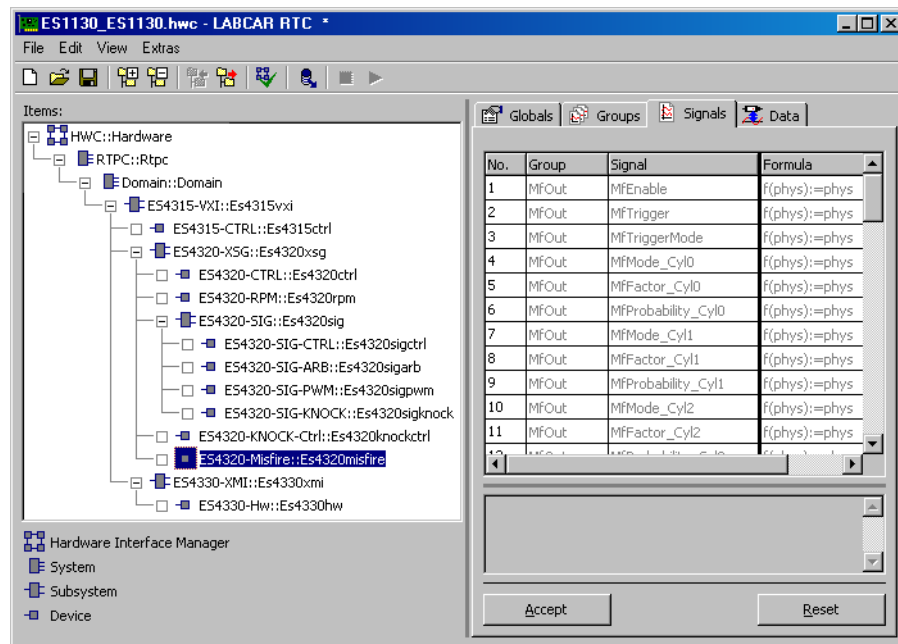


Abb. 17-30 Die Registerkarte „Signals“ des ES4320-Misfire Devices

### *MfEnable*

---

Das RTIO-Signal „MfEnable“ wird verwendet, um den Fehlzündungsgenerator an- oder abzuschalten. Ein Wert von 0 schaltet den Fehlzündungsgenerator ab, ein Wert von 1 schaltet den Fehlzündungsgenerator an.

### *MfTrigger*

---

Ein 0 → 1 Übergang des Signals „MfTrigger“ löst eine Fehlzündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus.

Das Signal „MfTrigger“ ist nur aktiv, wenn der Fehlzündungsgenerator im Triggermodus „Single Shot“ betrieben wird. Der Triggermodus wird über das Signal „MfTriggerMode“ eingestellt.

### *MfTriggerMode*

---

Das RTIO-Signal „MfTriggerMode“ dient der Steuerung des Triggermodus des Fehlzündungsgenerators.

Wenn das Signal auf „0“ gesetzt wird, arbeitet der Fehlzündungsgenerator im Triggermodus „SingleShot“.

Ein 0 → 1 Übergang des Signals „MfTrigger“ löst eine Fehlzündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus. Danach wird die Simulation von Fehlzündungen ausgesetzt bis zum Auftreten des nächsten Triggers.

Wenn das Signal auf „1“ gesetzt wird, arbeitet der Fehlzündungsgenerator im Triggermodus „Continuous“. In diesem Modus ist die Simulation von Fehlzündungen ständig aktiv. In diesem Modus ist das Signal „MfTrigger“ nicht aktiv.

### *MfMode\_Cyl0 ... MfMode\_Cyl19*

---

Die RTIO-Signale „MfMode\_Cyl0“ bis „MfMode\_Cyl11“ bestimmen die Betriebsart des Fehlzündungsgenerators für den jeweiligen Zylinder.

- Disabled Mode (MfMode\_Cyln = 0)  
Der Fehlzündungsgenerator für den jeweiligen Zylinder ist nicht aktiv.
- Pattern Mode (MfMode\_Cyln = 1)  
Die entsprechende Tabelle „Misfire Sequence nn“ bestimmt das Muster der Fehlzündungen für den jeweiligen Zylinder.
- Random Mode (MfMode\_Cyln = 2)  
Die Simulation von Fehlzündungen für den aktuelle Zündvorgang wird von einem Zufallsgenerator gesteuert. Die Wahrscheinlichkeit für das Auftreten von Fehlzündungen wird über das entsprechende Signal „MfProbability“ festgelegt.
- Worst/Best Mode (MfMode\_Cyln = 3)  
Die Simulation von Fehlzündungen wird über die entsprechenden Tabellen gesteuert (siehe „Worst Case Speed Drop/Best Case Speed Drop“ auf Seite 520).

### *MfFactor\_Cyl0 ... MfFactor\_Cyl19*

---

Die RTIO-Signale „MfFactor\_Cyl0“ bis „MfFactor\_Cyl19“ spezifizieren den Korrekturfaktor für den Abfall der Motordrehzahl nach der Fehlzündung für den jeweiligen Zylinder.

Ein Faktor von 0.7 bedeutet, dass lediglich 70 % des in der Tabelle „Speed Drop Pattern“ angegebenen Betrages der Drehzahlverringerng wirksam werden.

Gültiger Wertebereich: 0.0 ... 1.0.

*MfProbability\_Cyl0 ... MfProbability\_Cyl19*

Die RTIO-Signale „MfProbability\_Cyl0“ bis „MfProbability\_Cyl19“ bestimmen die Wahrscheinlichkeit für eine Fehlzündung während der aktuellen Zündung.

Diese RTIO-Signale sind nur aktiv, wenn das entsprechende Signal „MfMode\_Cyln“ auf „Random Mode“ gesetzt ist.

Der gültige Wertebereich ist 0.0 bis 1.0. Ein Wert von 0.0 bedeutet, dass eine Fehlzündung nie auftritt, ein Wert von 1.0 bedeutet, dass für jede Zündung im jeweiligen Zylinder eine Fehlzündung auftritt.

#### **Hinweis**

*Die Indizes 0 bis 19 der RTIO-Parameter „MfMode\_Cyln“, „MfFactor\_Cyln“ und „MfProbability\_Cyln“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*



Tab. 17-4 fasst die Eigenschaften der RTIO-Signale des ES4320-Misfire Devices zusammen.

RTIO Signal	Datentyp	Kommentar
MfEnable	uint8 / bool	An- und Abschalten des Fehlzündungs- generators 0: Abschalten 1: Anschalten
MfTrigger	uint8 / bool	Triggersignal des Fehlzündungsgenerators. Ein 0 → 1 Übergang löst eine Fehl- zündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus. Nur aktiv im Triggermodus „Single Shot“.
MfTriggerMode	uint8 / bool	Triggermodus des Fehlzündungsgenera- tors. 0: Triggermodus „Single Shot“ 1: Triggermodus „Continuous“
MfMode_Cyl0 ... MfMode_Cyl19	uint8	Betriebsart des Fehlzündungsgenerators für den jeweiligen Zylinder 0: Disabled 1: Pattern mode 2: Random mode 3: Best/worst case mode
MfFactor_Cyl0 ... MfFactor_Cyl19	real32	Korrekturfaktor für den Abfall der Motor- drehzahl nach der Fehlzündung für den jeweiligen Zylinder Gültige Werte: 0.0 ... 1.0
MfProbability_Cyl 0 ... MfProbability_Cyl 19	real32	Wahrscheinlichkeit für eine Fehlzündung während der aktuellen Zündung. Nur aktiv im „Random Mode“. Gültige Werte: 0.0 (keine Fehlzündung) ... 1.0 (dauernde Fehlzündung)

**Tab. 17-4** RTIO-Signale des ES4320-Misfire Devices

## 17.4 ES4330-XMI - VXI Signal Measurement Board

### Begriffsdefinitionen

#### Aktive Signalflanke:

Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal:

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Pulsflanke:

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal:

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### LWL:

Steht für **L**ower **W**indow **L**imit und bezeichnet die untere Grenze des Winkelfensters bei drehzahlsynchronen Messungen.

#### Öffnende Pulsflanke:

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



#### UWL:

Steht für **U**pper **W**indow **L**imit und bezeichnet die obere Grenze des Winkelfensters bei drehzahlsynchronen Messungen.

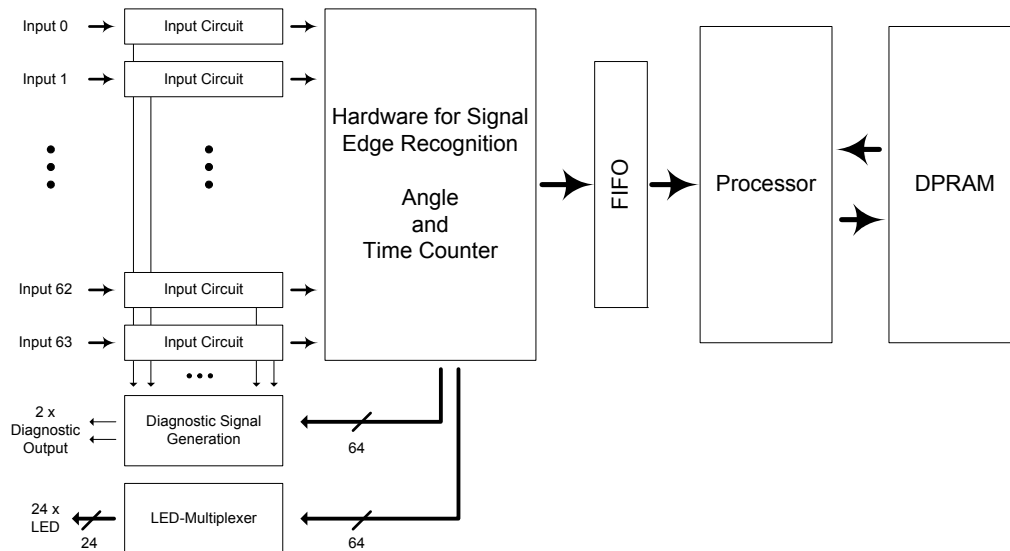
### 17.4.1 Funktionsbeschreibung

#### Grundprinzip der Messwertbestimmung

Um eine möglichst große Flexibilität bei der Auswertung anliegender Eingangssignale zu erreichen, basiert die ES4330-Karte auf folgendem Grundprinzip:

Zuerst wird jedes Eingangssignal separat (analog) aufbereitet, dann mit einem (für jedes Signal einzeln softwaretechnisch einstellbaren) Schwellwert verglichen, um die Umsetzung des analogen Eingangssignals in eine digitale 0/1- bzw. inaktiv/aktiv-Information zu realisieren. Die so erhaltene binäre Pegelinformation der

Signale an den 64 Eingangskanälen wird ständig von einer Hardwareschaltung auf Änderungen hin überprüft, d. h., es wird eine Erkennung von „Flanken“ auf einem oder mehreren Eingangsbits durchgeführt. Wechselt nun zumindest ein Eingangssignal von aktiv auf inaktiv bzw. umgekehrt, so speichert die Schaltung die aktuellen Zustände der 64 Eingänge und ruft die momentanen Werte zweier integrierter Zähler ab, die die aktuelle Zeit und den aktuellen Kurbelwellenwinkel angeben. Diese Informationen werden hardwaregesteuert in einen FIFO-Puffer übertragen.



**Abb. 17-31** Blockschaltbild der ES4330-Karte

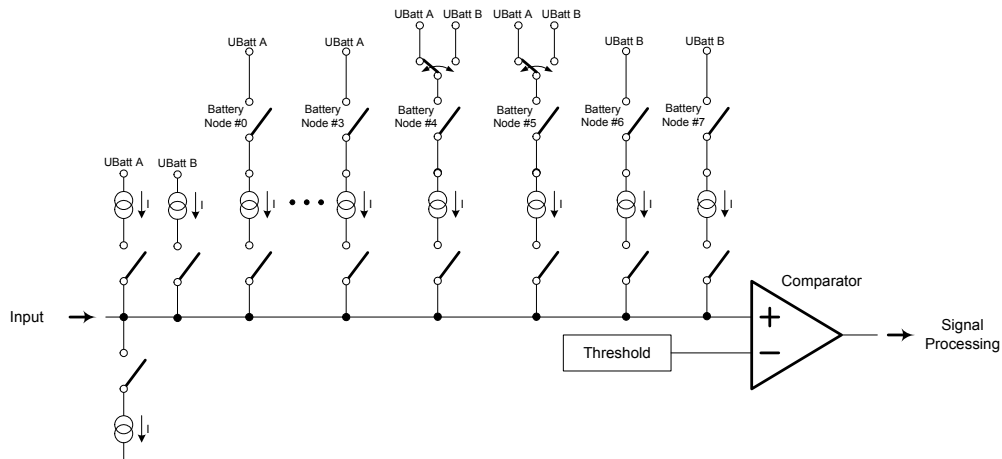
Die auf den Prozessor der ES4330-Karte heruntergeladene Firmware, kann nun die generierten Werte aus dem FIFO-Puffer auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben. Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind, wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten softwaretechnischen Aufbereitung der Messwerte bewirkt, dass die Auslastung des Prozessors direkt von der Zahl der erkannten Flanken (also den Frequenzen der Eingangssignale) abhängig ist. Der Vorteil dieses Konzepts: Die Hardware erfasst lediglich sämtliche Änderungen und versieht sie mit Winkel- und Zeitstempeln. Die Auswertung erfolgt hingegen per Software, ist demzufolge verhältnismäßig leicht zu ändern bzw. anzupassen. ES4330 Firmware und RTIO-Treiber bieten ein breites Spektrum an implementierten Messverfahren (vgl. Seite 551 ff.). Das eingesetzte Konzept ermöglicht die Implementierung weiterer Messverfahren - für die Realisierung genügt eine Anpassung der ES4330-Firmware und des RTIO-Treibers.

### Eingangsbeschaltung

Die in Abb. 17-31 auf Seite 527 bereits angedeutete Eingangsbeschaltung eines jeden Hardwarekanals der ES4330-Karte ist in Abb. 17-32 detaillierter dargestellt. Sie gliedert sich in zwei Einheiten, die nachfolgend erläutert werden:

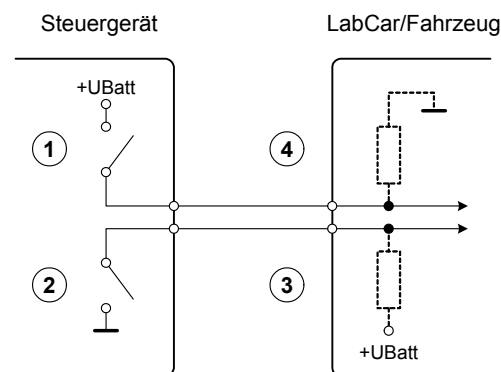
- Realisierung einer „Pull-Up“- / „Pull-Down“-Funktionalität und
- Schwellwertvergleich.



**Abb. 17-32** Vereinfachte Darstellung der Eingangsbeschaltung eines ES4330 Hardwarekanals

### Realisierung einer „Pull-Up“- bzw. „Pull-Down“-Funktionalität

Viele Steuergeräte-Ausgänge sind derart aufgebaut, dass nur die aktiven Pegel der Signale vom Steuergerät selbst generiert werden, d. h., die Ausgänge stellen reine Schalter gegen die Batteriespannung bzw. die Batteriemasse dar, wie es schematisch in der folgenden Abbildung (Punkte 1 und 2) skizziert ist. Als Beispiel für eine solche Beschaltung seien hier die Open-Collector-Ausgänge erwähnt, die funktionell einen Schalter gegen die Batteriemasse (2) realisieren.



**Abb. 17-33** Notwendigkeit von Pull-Up- / Pull-Down-Beschaltungen

Im realen Fahrzeug dienen derartige Signale zur Ansteuerung von Aktoren, die beispielsweise direkt an die Batteriespannung angeschlossen sind. Durch das Schalten des Steuerungssignals gegen Batteriemasse wird also in einer solchen Situation lediglich ein Stromkreis geschlossen.

Wird nun ein Steuergerät am LABCAR betrieben, und werden hierbei die Aktoren durch die ES4330-Karte ersetzt, so sind zwei Punkte zu beachten:

- Steuersignale, wie sie oben beschrieben sind, haben nur im aktiven Zustand einen wohldefinierten Pegel, im inaktiven Zustand sind sie (aus Steuergeräte-Sicht) nicht definiert. Um ein solches Signal mit der ES4330-Karte vermessen zu können, muss der inaktive Zustand des Signals folglich von der ES4330-Karte vorgegeben werden.
- Im aktiven Zustand fließt bei einem angeschlossenen Aktor ein Strom, der in einigen Steuergeräten gemessen wird, um eine Fehlererkennung zu ermöglichen (fließt trotz aktivem Signal kein Strom, so muss ein Fehler am Aktor vorliegen). Dieser Stromfluß muss also auch dann ermöglicht werden, wenn statt des Aktors die ES4330-Karte angeschlossen ist.

Das Standardverfahren zur Lösung dieser Probleme wäre der Einsatz von Pull-Up- bzw. Pull-Down-Widerständen, wie sie in Abb. 17-33 (Punkte 3 und 4) skizziert sind. Diese würden zum einen den Spannungspegel des inaktiven Signals vorgeben, zum anderen könnte bei einem anliegenden aktiven Signal über die Widerstände ein Strom fließen.

Nachteil dieser Konstruktion wäre allerdings, dass die Größe des fließenden Stroms direkt von der Batteriespannung abhängig wäre ( $I = U / R$ ); eine simulationsgesteuerte Modifikation der Batteriespannung würde sich somit u. U. direkt auf die Fehlererkennung im Steuergerät auswirken. Aus diesem Grunde finden auf der ES4330-Karte Konstantstromquellen Verwendung wie in Abb. 17-32 auf Seite 528 skizziert sind. Diese erfüllen die gleiche Funktionalität wie entsprechende Pull-Up- bzw. Pull-Down-Widerstände, jedoch ist hierbei für verschiedene Batteriespannungen nicht der Widerstand konstant und der Strom variabel, sondern (wie der Name schon sagt) der Strom konstant und der resultierende (scheinbare) Widerstand variabel.

#### *Schwellwertvergleich*

---

Neben der Realisierung der Pull-Up- und Pull-Down-Funktionalität besteht die Beschaltung eines ES4330-Eingangs im Wesentlichen aus einem Vergleich mit einem durch den Anwender definierten Schwellwert. Durch diesen Vergleich wird die Umsetzung des analogen Eingangssignals in eine digitale 0/1-Information erreicht. Der Schwellwert ist derart definiert, dass sämtliche am Karteneingang anliegenden Spannungen, die größer als dieser Wert sind, als „1“ interpretiert werden. Alle Spannungen unterhalb dieses Schwellwertes werden als „0“ interpretiert.

Der Schwellwert kann für jeden Kanal unabhängig von allen anderen Kanälen definiert werden. Für jeden Kanal ist er alternativ absolut im Bereich von 0 V bis 60 V spezifizierbar oder relativ zur Batteriespannung A oder B.

Gerade im Kfz-Umfeld kommen sehr häufig verhältnismäßig langsame, also niederfrequente Signale zum Einsatz, die zudem an den Flanken sehr oft auch große Anstiegs- bzw. Abfallzeiten aufweisen. Werden nun solche Signale mit einem vorgegebenen Schwellwert verglichen, so besteht – insbesondere wenn das Signal noch ein gewisses Rauschen aufweist – die Gefahr, dass beim langsamen Über- bzw. Unterschreiten der Schwelle das Vergleichsergebnis einige Male zwischen „1“ und „0“ hin und her wechselt. Um diese unerwünschten Nebeneffekte zu vermeiden, weisen die Vergleichsschaltungen auf der ES4330-Karte eine Hysterese auf, d. h., der Schwellwert für den Wechsel von „0“ auf „1“ liegt etwas höher als der Schwellwert für den Wechsel von „1“ nach „0“. Die Größe der Hysterese ist per Software einstellbar.

### Messwerterfassung

Der ES4330 RTIO-Treiber stellt dem Anwender bis zu 256 frei konfigurierbare Messwerte (im folgenden Software- bzw. Messkanäle genannt) zur Verfügung, wobei jeder einzelne Messwert von jedem beliebigen Hardwarekanal abhängen kann. Die Zuordnung von Hardwarekanälen auf der einen und Messkanal auf der anderen Seite ist somit nicht durch den ES4330 RTIO-Treiber vorgegeben, d. h., bei der Nutzung eines Laborautos für mehrere Projekte ist es nicht zwingend notwendig, bedeutungsgleiche Signale immer an denselben Hardwarekanälen der ES4330-Karte anzuschließen.

Darüberhinaus werden dem Anwender mit den 256 Messkanälen viermal so viele Ausgangsgrößen zur Verfügung gestellt, wie die Karte Hardwarekanäle aufweist. Hierdurch ist es problemlos möglich, aus einem einzigen Eingangssignal mehrere Messwerte zu ermitteln, wie z.B. Frequenz und Tastverhältnis eines PWM-Signals. Eine doppelte Verdrahtung ein und desselben Signals auf verschiedene Kanäle der Karte entfällt.

Für jeden dieser Messkanäle kann (neben dem zugrundeliegenden Hardwarekanal) vollkommen unabhängig von allen anderen Messwerten definiert werden,

- wie das anliegende Signal auszuwerten ist,
- wann und wie es auf fehlende Pulse hin untersucht werden soll (*Timeout-Überwachung*) und
- ob die Auswertung dauernd stattfinden soll oder nur innerhalb eines bestimmten Bereiches des Kurbelwellenwinkels (*Winkelsegmentierung*).
- Für einige Messungen läßt sich weiter festlegen, der wievielte Puls vermessen werden soll.

### Timeout-Überwachung

Wie weiter oben bereits erläutert, arbeitet die ES4330-Karte flankengesteuert, d. h., die Auswertung der anliegenden externen Signale findet ausschließlich an deren Flanken statt. Dies hat jedoch zur Folge, dass auch die generierten Messwerte, die der ES4330 RTIO-Treiber liefert, nur nach Flanken aktualisiert werden. Ein Signal, welches bereits einmal von der Karte ausgewertet wurde, würde folglich auch dann noch den zuletzt generierten gültigen Messwert liefern, wenn es inzwischen z.B. vollkommen ausgefallen ist, also keinerlei Flanken mehr aufweist.

Um Signale, die ganz oder zeitweise ausfallen, auch korrekt zu interpretieren, bietet der ES4330 RTIO-Treiber die Möglichkeit der Timeout-Überwachung. Jeder einzelne Messwert kann hierzu in bestimmten Abständen daraufhin getestet werden, ob er durch neu aufgetretene Flanken am Eingangssignal aktualisiert wurde. Dieser Test kann sowohl in vom Anwender spezifizierbaren Zeitabständen als auch jeweils an vom Anwender vorgegebenen aktuellen Kurbelwellenwinkeln erfolgen.

Wird während einer derartigen Überprüfung festgestellt, dass seit der letzten Überprüfung keinerlei Flanke von der Hardware detektiert wurde, so kann der entsprechende Messwert auf zwei verschiedene Arten modifiziert werden. Entweder wird er auf einen vorgegebenen Timeout-Wert gesetzt oder aber in Abhängigkeit vom aktuellen Pegel (high oder low) des Eingangssignals ermittelt. Dies ist insbesondere zur korrekten Auswertung von Tastverhältnissen erforderlich (ein inaktiver Eingang resultiert hierbei in den Messwerten 0.0 bzw. 1.0 – je nach Pegel des Eingangssignals).

Insgesamt ergeben sich somit sechs Möglichkeiten der Timeout-Überwachung:

- Überprüfung in Zeitabständen von  $x$  ms: Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung alle  $720^\circ$  bei  $x^\circ$  Kurbelwelle: Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung alle  $720^\circ$  bei  $x_1^\circ$  Kurbelwelle und alle  $x_2$  ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien auftreten. Messwert im Timeout-Fall vordefiniert auf Wert  $y$ .
- Überprüfung in Zeitabständen von  $x$  ms: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle  $720^\circ$  bei  $x^\circ$  Kurbelwelle: Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.
- Überprüfung alle  $720^\circ$  bei  $x_1^\circ$  Kurbelwelle und alle  $x_2$  ms. Ein Timeout wird festgestellt, wenn eine oder beide Timeout-Kriterien auftreten. Messwert im Timeout-Fall abhängig vom Pegel des Eingangssignals.

#### 17.4.2 Aufbau des ES4330 RTIO Baums

---

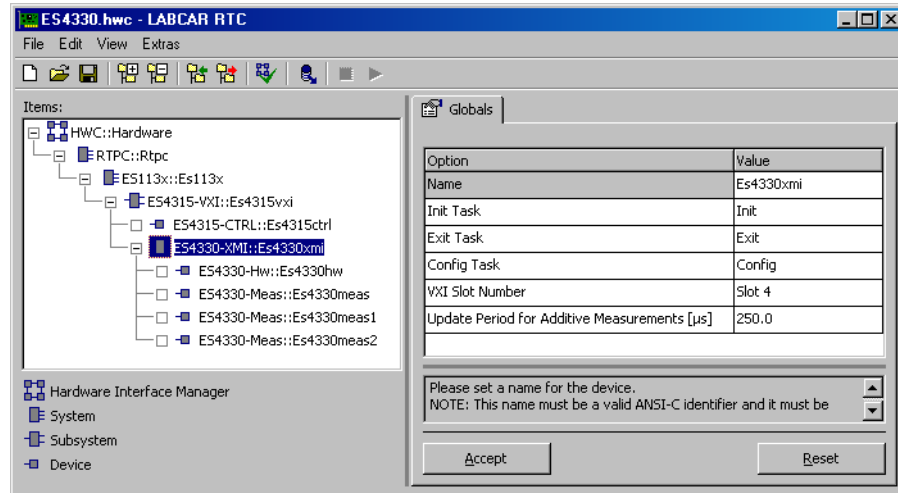
Mit der ES4300-Signalbox wurde von ETAS ein kombiniertes VMEbus-/VXIbus-System entwickelt. Die beiden Bussysteme sind mit einander über eine ES4315-Karte, die die Protokolle der beiden Busse umsetzt, gekoppelt. Die ES4315-Karte befindet sich in Slot 0 des VXIbus und in Slot 2 des VMEbus. Ein in der ES4300 installierter VMEbus Master (üblicherweise eine ES1130-Karte) übernimmt zusätzlich die Aufgaben eines VXIbus Systemcontrollers. Diese Funktionalität ist im RTIO-Treiber der ES4315-Karte enthalten. Die ES4330-Karte kann in den Slots 1 bis 9 des VXIbus-Systems eingebaut werden.

##### **Hinweis**

*VXIbus-Slots werden von 0 ausgehend durchnummeriert, VMEbus-Slots werden beginnend mit 1 nummeriert.*

Die beschriebene Hardware-Anbindung des VXIbus an den VMEbus spiegelt sich im RTIO-Baum wider (Abb. 17-34). Die Einbindung des VXIbus-Teils einer ES4300-Signalbox in eine RTIO-Hardwarebeschreibung beginnt mit dem Hinzu-

fügen eines ES4315-VXI Items an ein Experimentaltarget. Dem ES4315-VXI Item sind dann die sich im VXIbus-Teil des ES4300-Systems befindlichen VXIbus-Karten hinzuzufügen.



**Abb. 17-34** RTIO Hardwarebeschreibung mit ES4315 VMEbus/VXIbus-Adapter und ES4330-Karte

Die ES4330 Karte bietet 64 Hardwarekanäle zur Erfassung und Aufbereitung analoger und digitaler Signale. Die Konfiguration der 64 Hardwarekanäle erfolgt mit dem ES4330-Hw Device, das dem ES4330-XMI Subsystem zugeordnet wird. Neben diesem Device lassen sich dem ES4330-XMI Subsystem bis zu 8 sogenannter ES4330-Meas Devices zuordnen. Mit jedem dieser Items lassen sich bis zu 32 Messungen spezifizieren. Insgesamt bieten RTIO-Treiber und Firmware der ES4330-Karte damit 256 Messungen bzw. Messkanäle.

#### **Hinweis**

*Bevor ein ES4330-Meas Device eingefügt werden kann, muss zunächst ein ES4330-Hw Device zur Konfiguration der ES4330 Hardware eingefügt werden.*

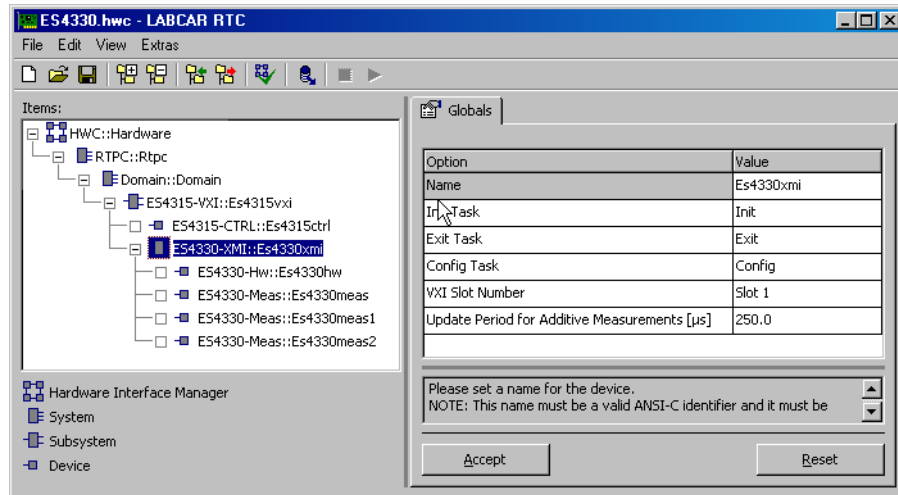
#### 17.4.3 Globale Einstellungen - ES4330-XMI Subsystem

Das ES4330-XMI Subsystem dient zur Einstellung von RTIO-Parametern, die global wirksam sind, d. h. die auf alle ES4330-Meas Devices wirken.



17.4.4 Globals (ES4330-XMI Subsystem)

Abb. 17-35 zeigt die RTIO-Parameter der Registerkarte „Globals“. Tab. 17-5 auf Seite 533 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 17-35** Die Registerkarte „Globals“ des ES4330-XMI Subsystems

*VXI Slot Number*

Nummer des VXIbus-Slots, in dem die ES4330-Karte eingebaut ist. Die Numerierung der sichtbaren VXIbus-Slots innerhalb der ES4300 Signalbox beginnt mit 1 und verläuft von links nach rechts. Der ES4315 VXIbus- / VMEbus-Adapter ist fest in den für den Anwender nicht sichtbaren VXIbus-Slot 0 eingebaut. Die VXIbus-Slot-Nummer ist nicht zur Laufzeit editierbar.

*Update Period for Additive Measurements [µs]*

Dieser Parameter ist nur für Software- bzw. Messkanäle von Bedeutung, die additive Zeitmessungen im asynchronen Modus durchführen. Er legt fest, in welchen Zeitabständen spätestens eine Aktualisierung der additiven Zeit stattfindet. Weitere Informationen finden Sie in „Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen“ auf Seite 555. Der Parameter ist zur Laufzeit des Modells konfigurierbar.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
VXI Slot Number	uint32	Nein	VXIbus Slot Nummer Zur Auswahl stehen „Slot 1“ bis „Slot 9“.
Update Period for Additive Measurements [µs]	real32	Ja	Messwert-Aktualisierungsrate für asynchrone additive Messungen in Mikrosekunden. Minimal: 50 µs Maximal: 500 ms

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

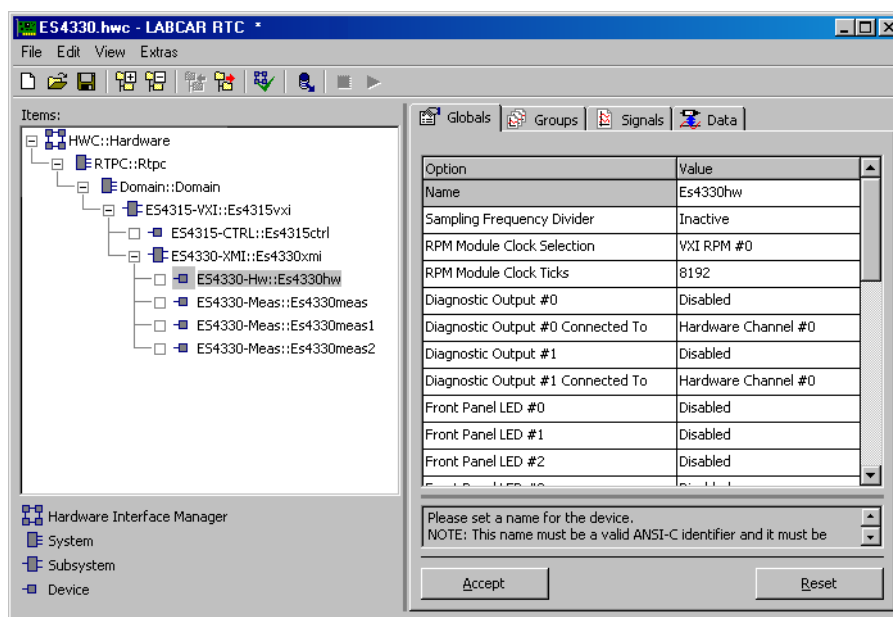
**Tab. 17-5** ES4330-XMI Subsystem: Konfigurationsparameter der Registerkarte „Globals“

#### 17.4.5 Hardwarekonfiguration - ES4330-Hw Device

Das ES4330-Hw Device dient zur Konfiguration und Ansteuerung der ES4330 Hardware. Von den einzelnen Hardwarekanälen unabhängige Einstellungen werden in der Registerkarte „Globals“ dieses Items durchgeführt. In der Registerkarte „Signals“ werden kanalspezifische Einstellungen vorgenommen.

#### 17.4.6 Globals (ES4330-Hw Device)

Abb. 17-36 zeigt die RTIO-Parameter der Registerkarte „Globals“. Tab. 17-6 auf Seite 536 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 17-36** Die Registerkarte „Globals“ des ES4330-Hw Devices

#### *Sampling Frequency Divider*

Die ES4330 tastet die Eingangssignale an den Hardwarekanälen periodisch ab. Die maximale Abtastrate ist durch die Frequenz des internen Taktgebers bestimmt, die 4 MHz beträgt. Mit einem Frequenzteiler kann die Abtastfrequenz reduziert werden. Die Konfiguration des Frequenzteilers erfolgt im Optionsfeld „Sampling Frequency Divider“ der Registerkarte „Globals“. Die Frequenzteilung kann deaktiviert oder auf die Teiler 2, 4, 8, 16 oder 32 konfiguriert werden. Es ist zu beachten, dass nur Eingangspulse mit einer Dauer größer als die Abtastperiodendauer sicher erfasst werden.

#### *RPM Module Clock Ticks - RPM Module Clock Selection*

Zur Durchführung von zur Motordrehzahl synchronen Messungen benötigt die ES4330-Karte Informationen über die Winkelstellung und Drehrichtung des Motors. Diese Informationen stehen ES4330-Karten an der VXIbus Backplane zur Verfügung. Das ES4300-System bietet auf der VXIbus-Backplane zwei Winkeltakt-Leitungen mit jeweils drei Signalen, ein Synchronisationssignal auf den Nulldurchgang des Kurbel- bzw. Nockenwellenwinkels, ein Signal, in dem die

Drehrichtung des Motors kodiert ist und dem eigentlichen Taktsignal, in dem die Winkelstellung des Motors kodiert ist. Mit dem Optionsfeld „RPM Module Clock Selection“ wird festgelegt, welche der beiden Winkeltakt-Leitungen (Option „VXI RPM #0“ oder „VXI RPM #1“) die ES4330 mit dem Winkeltakt versorgt. Wird die Option „None“ eingestellt wird, so wird die ES4330-Karte mit keinem Winkeltakt versorgt.

Das numerische Eingabefeld „RPM Module Clock Ticks“ dient zur Definition der Winkelauflösung, d. h. die Anzahl der Winkeltakte pro Nockenwellenumdrehung (720° Kurbelwelle). Wird als Beispiel 8192 eingestellt, so arbeitet die ES4330 Messerfassung mit einer Winkelauflösung von  $720^{\circ}\text{KW} / 8192 = 0.088^{\circ}\text{KW}$  ( $^{\circ}\text{KW} = \text{Grad Kurbelwelle}$ ).

*Diagnostic Output #0, Diagnostic Output #0 Connected To, Diagnostic Output #1, Diagnostic Output #1 Connected To*

---

Die ES4330-Karte bietet auf der Frontplatte zwei Anschlüsse für Test- und Diagnosezwecke. Über diese Anschlüsse können zwei voneinander unabhängige Signale zur Beobachtung und Auswertung z.B. direkt an ein Oszilloskop angeschlossen werden. Als Testsignale stehen die Eingangssignale aller 64 Hardwarekanäle zur Verfügung. Die Auswahl des Hardwarekanals erfolgt in dem Optionsfeld „Diagnostic Output #x Connected To“ ( $x = 0, 1$ ).

Zusätzlich zur Wahl des eigentlichen Signals hat der Anwender die Möglichkeit festzulegen, von welcher Stelle des Signalflusses aus das Signal an den Testanschluß ausgegeben werden soll. Zum einen kann es als analoges Signal unmittelbar vor dem Komparator des Hardwarekanals abgegriffen werden, wobei das Testsignal dann technisch bedingt eine Ausgangsamplitude von 1/6 der Amplitude des Eingangssignals aufweist. Zum anderen hat der Anwender die Möglichkeit, das Testsignal unmittelbar hinter dem Komparator des Hardwarekanals, also als TTL-Signal, abzugreifen. Durch diese Möglichkeit des Signalabgriffes vor und hinter dem Komparator ist eine direkte Kontrolle des eingestellten Schwellwertes sehr einfach möglich. Die Auswahl des Signaltyps erfolgt im Optionsfeld „Diagnostic Output #x“ ( $x = 0, 1$ ), zusätzlich bietet dieses Optionsfeld noch die Möglichkeit, den Testausgang zu deaktivieren.

*Front Panel LED #0 ... Front Panel LED #23*

---

Auf der Frontplatte der ES4330-Karte befinden sich 24 LEDs. Mit jeder LED läßt sich der Signalpegel auf einem Hardwarekanal optisch anzeigen. Weist das Signal einen Low-Pegel auf, so wird die LED dunkel geschaltet. Weist das Signal einen High-Pegel auf, so wird die LED hell geschaltet. Die Definition von Low- und High-Pegel ist durch die Schwellspannung des Eingangskomparators des

jeweiligen Hardwarekanals gegeben. Die Registerkarte „Globals“ bietet für jede LED ein Optionsfeld „Front Panel LED #x“ (x = 0, 1, ...23) mit dem der zugeordnete Hardwarekanal eingestellt bzw. die LED deaktiviert werden kann.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Sampling Frequency Divider	uint8	Ja	Frequenzteiler für Eingangssignal-Abtastung 0: Keine Frequenzteilung (4 MHz Abtastung) 1: Teilung durch 2 (2 MHz) 2: Teilung durch 4 (1 MHz) 3: Teilung durch 8 (500 kHz) 4: Teilung durch 16 (250 kHz) 5: Teilung durch 32 (125 kHz)
RPM Module Clock Ticks	uint32	Ja	Winkelauflösung Wertebereich: 1 ... 65536
RPM Module Clock Selection	uint8	Ja	Quelle für Winkeltakt 0: Kein Winkeltakt 1: RPM Module #0 2: RPM Module #1
Diagnostic Output #0 Diagnostic Output #1	sint8	Ja	Aktivierung und Deaktivierung der Diagnoseausgänge sowie Auswahl des Signaltyps (analog oder TTL) -1: Deaktivierung des Diagnoseausgangs 0: TTL Signalausgabe 1: Analog Signalausgabe (Amplitude ist 1/6 der Amplitude des Eingangssignals)
Diagnostic Output #0 Connected To Diagnostic Output #1 Connected To	uint8	Ja	Hardwarekanal, der auf den Diagnoseausgang geschaltet ist. Wertebereich: 0 ... 63
Front Panel LED #0 ... Front Panel LED #23	sint8	Ja	Hardwarekanal, der auf die Frontplatten LED geschaltet ist bzw. Deaktivierung der LED. -1: Deaktivierung 0 ... 63: Hardwarekanal

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-6** ES4330-Hw Device: Konfigurationsparameter der Registerkarte „Globals“

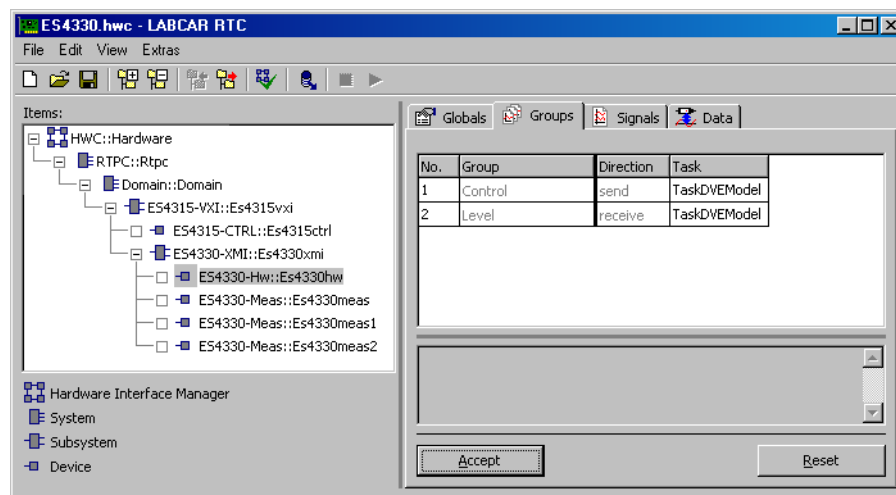
#### 17.4.7 Groups (ES4330-Hw Device)

Das ES4330-Hw Device besitzt zwei Signalgruppen (Abb. 17-37). Die Signalgruppe „Control“ wird vom Experimentaltarget zur ES4330-Karte übertragen. Mit dieser Signalgruppe wird die Flankenerkennung auf den einzelnen Hard-

warekanälen der Karte ein- und ausgeschaltet. Die Signalgruppe „Level“ wird in umgekehrter Richtung von der ES4330-Karte zum Experimentaltarget übertragen. Diese Signalgruppe enthält die aktuellen Pegelinformationen aller eingeschalteten Hardwarekanäle.

Den beiden Signalgruppen sind ERCOS<sup>EK</sup> Betriebssystem-Tasks zuzuordnen. Für die „Control“-Signalgruppe wird man üblicherweise eine Task mit „Type“ „Alarm“ und verhältnismäßig großer Aktivierungsperiode (z.B. 100 ms) wählen, da das Ein- und Ausschalten der Hardwarekanäle in der Regel kein hochdynamischer Vorgang ist. Falls das Ein- und Ausschalten der Hardwarekanäle nur beim Modellstart bzw. Modellstop erfolgen soll, so ist es ausreichend, die „Control“-Signalgruppe der „Init“ und der „Exit“ Task des Modells zuzuordnen.

Falls man die Pegelinformationen im Simulationsmodell auswertet, wird man der Signalgruppe „Level“ eine Task mit „Type“ „Alarm“ zuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik bzw. der Periodendauer der zu erfassenden Signale.



**Abb. 17-37** Die Registerkarte „Groups“ des ES4330-Hw Devices

17.4.8 RTIO-Signale der Signalgruppe „Control“

Die Signalgruppe „Control“ umfasst 64 RTIO-Signale, mit denen die Flanken-erkennung auf den 64 Hardwarekanälen aktiviert bzw. deaktiviert werden kann. Tab. 17-7 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
ChnEnable_0	bool	Ein- / Ausschalten der Hardwarekanäle 0 bis 63
...		0: Deaktivierung der Flankenerkennung
ChnEnable_63		1: Aktivierung der Flankenerkennung
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 17-7** ES4330-Hw Device: Die RTIO-Signale der Signalgruppe „Control“

**Hinweis**

Die „Control“-Signalgruppe wird nur dann in das DPRAM der ES4330-Karte geschrieben, wenn sich eines ihrer RTIO-Signale im Wert verändert hat.

## 17.4.9 RTIO-Signale der Signalgruppe „Level“

Die Signalgruppe „Level“ umfasst 2 RTIO-Signale, die als Bitfelder zu interpretieren sind. Die Bitfelder umfassen jeweils 32 Bit. Im Bitfeld „LvlBitField\_0“ sind die Pegelinformationen der Hardwarekanäle 0 bis 31 kodiert, im Bitfeld „LvlBitField\_1“ die der Hardwarekanäle 32 bis 63. Tab. 17-8 fasst die Eigenschaften der RTIO-Signale zusammen.

**Hinweis**

*Nur die Pegelbits von Hardwarekanälen, bei denen die Flankenerkennung aktiviert ist, sind in den Bitfeldern „LvlBitField\_0“ und „LvlBitField\_1“ gültig. Zur Aktivierung und Deaktivierung der Flankenerkennung auf einem Hardwarekanal siehe „RTIO-Signale der Signalgruppe „Control““ auf Seite 537.*

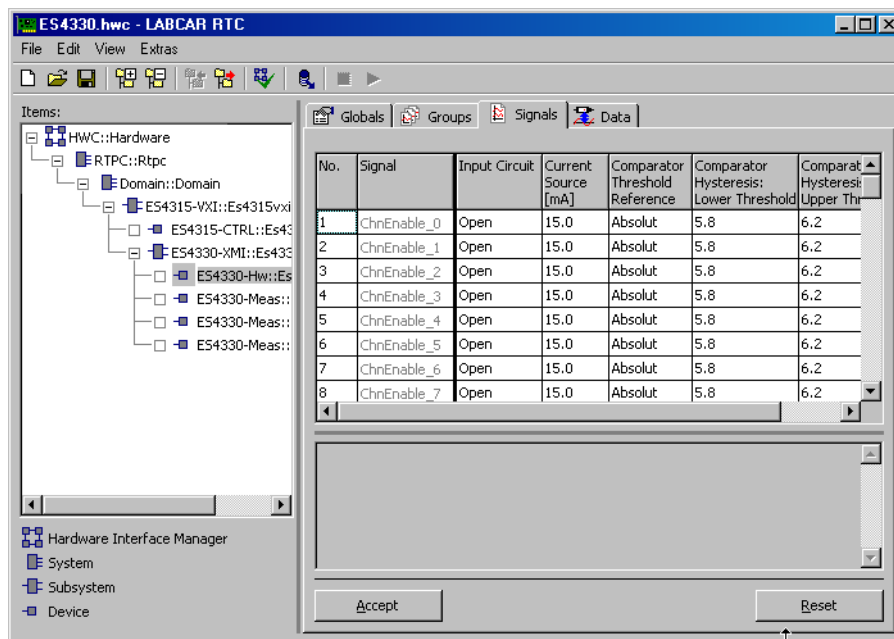
RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
LvlBitField_0	uint32	Bitfeld mit Pegelinformationen der Hardwarekanäle 0 bis 31. Kanal 0: LSB (Least Significant Bit) Kanal 31: MSB (Most Significant Bit) Bitwert 0: Low Pegel Bitwert 1: High Pegel
LvlBitField_1	uint32	Bitfeld mit Pegelinformationen der Hardwarekanäle 32 bis 63. Kanal 32: LSB Kanal 63: MSB Bitwert 0: Low Pegel Bitwert 1: High Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-8** ES4330-Hw Device: Die RTIO-Signale der Signalgruppe „Level“

## 17.4.10 Signals (ES4330-Hw Device)

In der Registerkarte „Signals“ wird die Konfiguration der 64 Hardwarekanäle einer ES4330-Karte durchgeführt. Abb. 17-38 zeigt die RTIO-Parameter der Registerkarte „Signals“. Tab. 17-9 auf Seite 541 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 17-38** Die Registerkarte „Signals“ des ES4330-Hw Devices

#### *Input Circuit*

Mit diesem Optionsfeld wird die Eingangsbeschaltung eines Hardwarekanals spezifiziert. Die verfügbaren Optionen sind aus dem Schaltbild eines Hardwarekanals (siehe Abb. 17-32 auf Seite 528) abgeleitet. Neben der direkten Verschaltung des Eingangssignals auf den Eingangskomparator (Option „Open“) kann eine Pull-Up-Beschaltung gegen die Batteriespannungen A oder B oder eine Pull-Down-Beschaltung gegen Batteriemasse geschaltet werden. Auch eine Pull-Up-Beschaltung gegen einen der acht Batterieknoten des ES4300-Systems ist möglich. Die in Abb. 17-32 auf Seite 528 dargestellte wahlweise Verschaltung der Batterieknoten 4 und 5 gegen die Batteriespannung A oder B sowie das Ein- und Ausschalten der einzelnen Batterieknoten wird nicht vom RTIO-Treiber der ES4330-Karte vorgenommen. Diese ES4300-Systemeigenschaften werden vom RTIO-Treiber der ES4315-Karte konfiguriert und überwacht.

#### *Current Source [mA]*

Im numerischen Eingabefeld „Current Source [mA]“ wird die Stromstärke der Konstantstromquellen eines Hardwarekanals definiert. Die maximal zulässige Stromstärke hängt von der Eingangsbeschaltung und der Größe der Batteriespannung ab (siehe Tab. 17-9 auf Seite 541).

### *Comparator Threshold Reference*

---

Das Optionsfeld „Comparator Threshold Reference“ dient zur Festlegung, ob die Schwellspannung eines Eingangskomparators durch Vorgabe einer Spannung in Volt definiert (Option „Absolut“) oder relativ zu den Batteriespannungen A oder B angegeben wird.

### *Comparator Hysteresis: Lower Threshold - Comparator Hysteresis: Upper Threshold*

---

Die Schwellspannung der Eingangskomparatoren eines Hardwarekanals besitzt ein Hystereseverhalten. Die numerischen Eingabefelder „Comparator Hysteresis: Lower Threshold“ und „Comparator Hysteresis: Upper Threshold“ dienen zur Festlegung der Komparatorhysterese und damit der Schwellspannung.

Wird im Optionsfeld „Comparator Threshold Reference“ des Hardwarekanals die Option „Absolut“ gewählt, so werden die numerischen Werte in den „Comparator Hysteresis“ Feldern als Spannungen in Volt interpretiert. Der zulässige Wertebereich ist 0 V bis 60 V.

Wird im Optionsfeld „Comparator Threshold Reference“ des Hardwarekanals die Option „Relative to UBatt\_A“ oder „Relative to UBatt\_B“ eingestellt, so ergeben sich die Komparator-Hysterese durch Multiplikation der numerischen Werte in den „Comparator Hysteresis“ Feldern mit der jeweiligen Batteriespannung. Die Einträge in den „Comparator Hysteresis“ Feldern müssen in diesem Fall zwischen 0 und 1 liegen.



Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Input Circuit	uint8	Ja	Festlegung der Eingangsbeschaltung 0: „Open“. Das Eingangssignal wird direkt auf den Eingangskomparator geschaltet. 1: „Pull-Up to UBatt_A“ 2: „Pull-Up to UBatt_B“ 3: „Pull-Down to Battery Ground“ 4 .. 11: „Pull-Up to Battery Node #0...#7“
Current Source [mA]	real32	Ja	Stromstärke der Konstantstromquelle in mA. Minimum: 0.0 mA Das Maximum hängt von der Eingangsbeschaltung und der Größe der Batteriespannung ab: 15 mA, wenn der Eingang über eine Pull-Down-Beschaltung auf Batteriemasse geschaltet ist. 15 mA bei einer Pull-Up-Beschaltung gegen eine Batteriespannung > 30 V. 30 mA bei einer Pull-Up-Beschaltung gegen eine Batteriespannung zwischen 0 V und 30 V.
Comparator Threshold Reference	uint8	Ja	Komparatorreferenz 1: „Relative to UBatt_A“ 2: „Relative to UBatt_B“ 3: „Absolut“
Comparator Hysteresis: Lower Threshold	real32	Ja	Untere Komparatorschwelle Wertebereich: 0.0 ... 1.0 bei relativer Komparatorreferenz 0.0 V ... 60 V bei absoluter Komparatorreferenz
Comparator Hysteresis: Upper Threshold	real32	Ja	Obere Komparatorschwelle Wertebereich: 0.0 ... 1.0 bei relativer Komparatorreferenz 0.0 V ... 60 V bei absoluter Komparatorreferenz

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

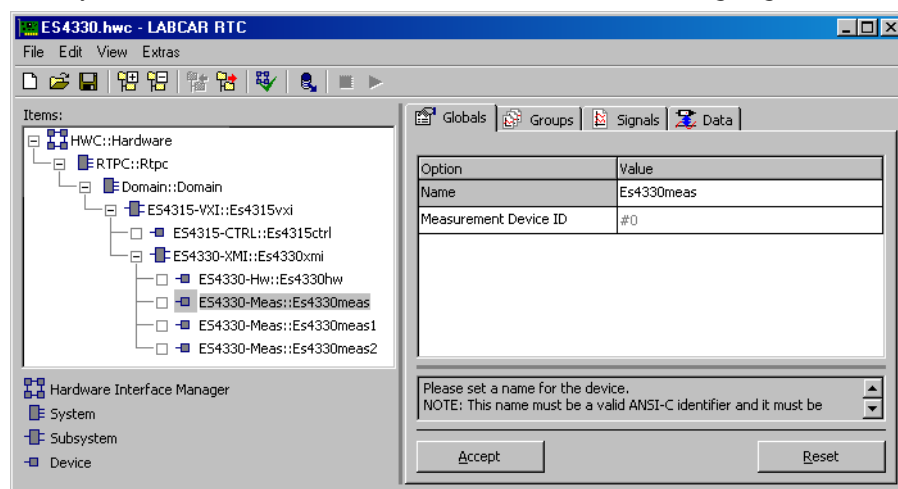
**Tab. 17-9** ES4330-Hw Device: Konfigurationsparameter der Registerkarte „Signals“

#### 17.4.11 Messkonfiguration - ES4330-Meas Device

Das ES4330-Meas Device dient zur Spezifikation und Konfiguration von Messungen. Jedes ES4330-Meas Device bietet 32 Softwarekanäle (im folgenden Messkanäle genannt), die frei auf die 64 Hardwarekanäle der ES4330-Karte verschaltet werden können. Bis zu 8 ES4330-Meas Devices werden pro ES4330-Karte unterstützt, so dass insgesamt 256 Messkanäle konfiguriert werden können.

#### 17.4.12 Globals (ES4330-Meas Device)

In der Registerkarte „Globals“ (Abb. 17-39) sind keine Einstellungen vorzunehmen. Im Optionsfeld „Measurement Device ID“ wird dem ES4330-Meas Device vom System eine Nummer zwischen 0 und 7 zur Identifizierung zugewiesen.

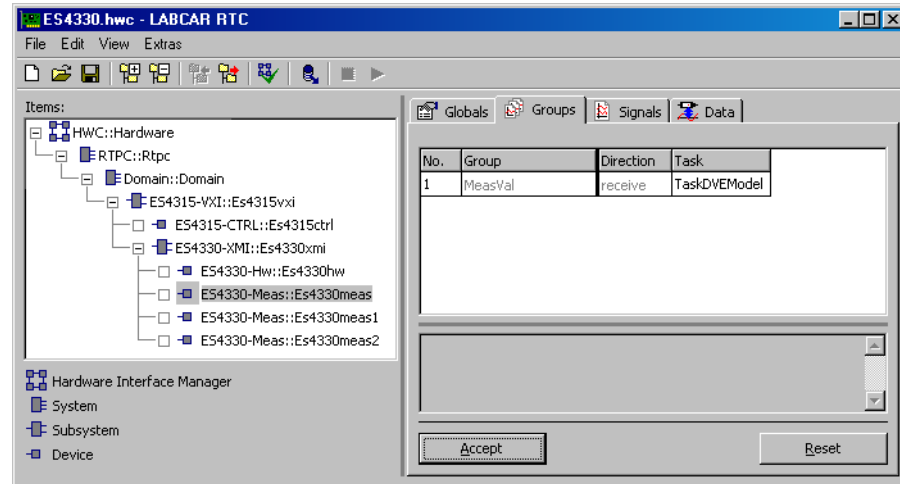


**Abb. 17-39** Die Registerkarte „Globals“ des ES4330-Meas Device

#### 17.4.13 Groups (ES4330-Meas Device)

Das ES4330-Meas Device besitzt eine Signalgruppe (Abb. 17-40), die von der ES4330-Karte zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen. Dieser Signal-

gruppe ist eine ERCOS<sup>EK</sup> Betriebssystem-Task mit dem „Type“ „Alarm“ zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.



**Abb. 17-40** Die Registerkarte „Groups“ des ES4330-Meas Devices

#### 17.4.14 RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 65 RTIO-Signale. Das „TriggerBitField\_0“-Signal ist als Bitfeld zu interpretieren. Es umfasst 32 Bit. In diesem Bitfeld sind die Trigger- oder Update-Daten der 32 Messkanäle kodiert, d. h., es zeigt an, welche Messwerte seit der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der Messwert des zugehörigen Messkanals neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeout oder aufgrund einer regulären Messwertberechnung erfolgte, in beiden Fällen wird das Aktualisierungsbit des Messkanals gesetzt.

Die RTIO-Signale „MeasVal\_0“ bis „MeasVal\_31“ enthalten die Messwerte der 32 Messkanäle. Es handelt sich dabei um regulär ermittelte Messwerte oder im Falle eines Timeout um den für diesen Fall vorgesehenen Timeout-Wert. Falls ein Messkanal nicht verwendet wird, wird dem zugehörigen Messwert -8888.0 zugewiesen. Die physikalische Einheit der Messwerte richtet sich nach dem Messverfahren:

- Zeitmessungen (Zeitstempel, (additive) Pulsdauer-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Frequenzmessungen erfolgen in Hertz
- Winkelmessungen und Winkelstempel werden in Grad Kurbelwelle (°KW) angegeben
- alle anderen Messungen (Tastverhältnisse, Pulszählung, Schrittmotormessungen, Pegelmessungen) sind dimensionslos

Die RTIO-Signale „Tout\_0“ bis „Tout\_31“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messkanal.

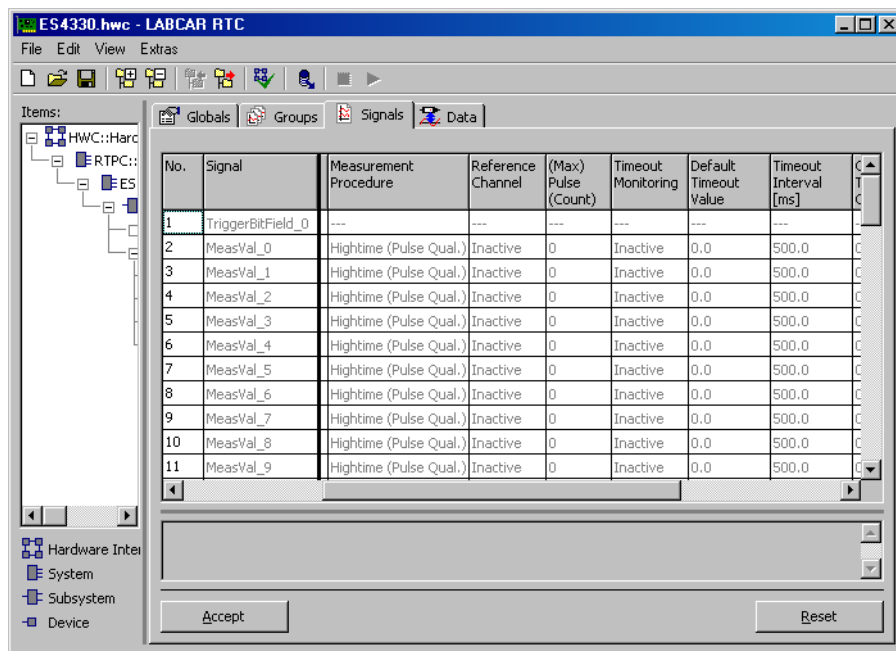
Tab. 17-10 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
TriggerBitField_0	uint32	Bitfeld mit Triggerinformationen der 32 Messkanäle. Messkanal 0: LSB Messkanal 31: MSB Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasVal_0 ... MeasVal_31	real64	Messwert Falls der Messkanal nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: Zeitmessungen erfolgen in Mikrosekunden Frequenzmessungen erfolgen in Hertz Winkelmessungen erfolgen in °KW
Tout_0 ... Tout_31	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout aufgetreten 1: Timeout aufgetreten 2: Timeout-Überwachung ist inaktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 17-10** ES4330-Meas Device: Die RTIO-Signale der Signalgruppe „MeasVal“

### 17.4.15 Signals (ES4330-Meas Device)

In der Registerkarte „Signals“ wird die Konfiguration der 32 Messkanäle eines ES4330-Meas Device durchgeführt. Abb. 17-41 zeigt die RTIO-Parameter der Registerkarte „Signals“. Tab. 17-11 auf Seite 551 fasst die Eigenschaften der einzelnen Parameter zusammen. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 17-41** Die Registerkarte „Signals“ des ES4330-Meas Devices

#### Hardware Channel

In dieser Listbox wird der Hardwarekanal festgelegt, auf dem die Messung durchgeführt werden soll. Wird die Einstellung „Inactive“ gewählt, so wird keine Messung durchgeführt, d. h., der Messkanal ist deaktiviert.

#### Hinweis

Zur Vermeidung unnötiger Rechenzeiten sind nicht benötigte Messkanäle zu deaktivieren.

#### Measurement Mode

Das Optionsfeld „Measurement Mode“ definiert den Modus, in dem eine Messung durchgeführt wird. Es stehen zwei Optionen zur Auswahl:

- „ESpeedSync“: Durchführung von drehzahlsynchronen bzw. winkelsynchronen Messungen. Die Durchführung von drehzahlsynchronen Messungen erfordert, dass die ES4330-Karte über die Backplane der ES4300 Signalbox mit einem Winkeltakt (Winkelstellung und Drehrichtung des Motors) versorgt wird. Nähere Informationen zur Konfiguration der Winkeltaktquelle finden Sie in „RPM Module Clock Ticks - RPM Module Clock Selection“ auf Seite 534.
- „Asynchronous“: Durchführung von asynchronen Messungen.

### *Measurement Procedure*

---

In dieser Listbox wird das Messverfahren ausgewählt. Die Abschnitte 17.4.16 - 17.4.26 enthalten detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren. In der Listbox bedeutet die Notation -- / -- eine Auswertung an steigenden Flanken, die Notation -- \ -- hingegen eine Auswertung an fallenden Flanken

### *Reference Channel*

---

Einige Messverfahren benötigen zur Durchführung einen zweiten Hardwarekanal oder einen zweiten Messkanal. Die Spezifikation dieses Referenzkanals für die Messung erfolgt in der Listbox „Reference Channel“.

Bei den in „Messverfahren - Pulsweitenmessungen mit Enable- oder Validierungssignal“ auf Seite 557 beschriebenen Pulsweitenmessungen mit Enable- oder Validierungssignal ist der Referenzkanal ein Hardwarekanal, der Kanal an dem das Enable- oder Validierungssignal anliegt.

Bei den in „Messverfahren - Relativmessungen zwischen Hardwarekanälen“ auf Seite 564 beschriebenen Relativmessungen zwischen Hardwarekanälen und bei den in „Messverfahren - Positionsverfolgung an zwei-phasigen Schrittmotoren“ auf Seite 575 beschriebenen Messungen zur Positionsverfolgung an zwei-phasigen Schrittmotoren ist der Referenzkanal ein Messkanal.

Bei allen anderen Messverfahren hat der in der Listbox „Reference Channel“ eingestellte Wert keine Bedeutung.

### *(Max) Pulse (Count)*

---

Die Bedeutung dieses Optionsfelds ist nicht eindeutig, sie hängt vom jeweiligen Messverfahren ab. Bei puls- und flankenselektiven Messungen wird in diesem Optionsfeld die Nummer des Pulses oder der Flanke angegeben, die vermessen werden soll. Bei additiven Messungen und Pulszählungen wird in diesem Optionsfeld die Maximalzahl von zu berücksichtigenden Pulsen angegeben.

### *Timeout Monitoring*

---

Definition der Timeout-Überwachung für den zugehörigen Messkanal (siehe auch „Timeout-Überwachung“ auf Seite 530). Die folgenden Einstellungen sind möglich:

- „Inactive“: Keine Timeout-Überwachung.
- „Intvl Predef“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl InpDep“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.
- „CS Angle Predef“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.

- „CS Angle InpDep“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.
- „Intvl & CS Angle Predef“: Timeout-Überwachung alle 720° Kurbelwellendrehung und zwar an dem im Optionsfeld „CS Angle Timeout Check Point“ definierten Kurbelwellenwinkel und Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Ein Timeout wird erkannt, wenn eine oder beide Timeout-Kriterien verletzt sind. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl & CS Angle InpDep“: Wie „Intvl & CS Angle Predef“, jedoch ist der Messwert im Timeout-Fall vom Pegel des Eingangssignals abhängig.

#### *Default Timeout Value*

---

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls einer der „Intvl Predef“, „CS Angle Predef“ oder „Intvl & CS Angle Predef“ Timeout-Überwachungsmodi eingestellt ist.

#### *Timeout Interval [ms]*

---

Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Nur relevant für die zeitbasierten Timeout-Überwachungsmodi („Intvl Predef“, „Intvl InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Timeout Check Point*

---

Kurbelwellenwinkel an dem Timeout-Kontrollen durchgeführt werden. Nur relevant für die winkelbasierten Timeout-Überwachungsmodi („CS Angle Predef“, „CS Angle InpDep“, „Intvl & CS Angle Predef“, „Intvl & CS Angle InpDep“).

#### *CS Angle Lower Limit*

---

Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.

#### *CS Angle Upper Limit*

---

Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“. Die obere Winkelfenstergrenze wird zudem bei additiven Zeitmessungen und Pulszählungen zum Transfer des ermittelten Messwertes in das DPRAM der ES4330-Karte genutzt.

#### *CS Angle Reference*

---

Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Hardware Channel	sint8	Ja	Hardwarekanal, auf dem die Messung durchgeführt wird, und Deaktivierung des Messkanals -1: Deaktivierung des Messkanals 0 ... 63: Kanalnummer
Measurement Mode	uint8	Ja	Messmodus 0: „Asynchronous“: Asynchrone Messungen 1: „ESpeedSync“: Drehzahl- bzw. winkelsynchrone Messungen
Measurement Procedure	uint32	Ja	Auswahl des Messverfahrens. Messverfahren      Parameterwert „Hightime (Pulse Qual.) [µs]“ 0 „Lowtime (Pulse Qual.) [µs]“ 1 „Hightime (Edge Qual.) [µs]“ 59 „Lowtime (Edge Qual.) [µs]“ 60 „H-Time n-th Pulse (Pu Qual.) [µs]“ 28 „L-Time n-th Pulse (Pu Qual.) [µs]“ 29 „H-Time n-th Pulse (Eg Qual.) [µs]“ 61 „L-Time n-th Pulse (Eg Qual.) [µs]“ 62  „Hightime using H-Enable [µs]“ 20 „Lowtime using H-Enable [µs]“ 24 „Hightime using L-Enable [µs]“ 21 „Lowtime using L-Enable [µs]“ 25 „Hightime using H-Validate [µs]“ 22 „Lowtime using H-Validate [µs]“ 26 „Hightime using L-Validate [µs]“ 23 „Lowtime using L-Validate [µs]“ 27 „H-Time n-th Pulse (H-Ena) [µs]“ 32 „L-Time n-th Pulse (H-Ena) [µs]“ 36 „H-Time n-th Pulse (L-Ena) [µs]“ 33 „L-Time n-th Pulse (L-Ena) [µs]“ 37 „H-Time n-th Pulse (H-Val) [µs]“ 34 „L-Time n-th Pulse (H-Val) [µs]“ 38



Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
			„H-Time n-th Pulse (L-Val) [µs]“ 35 „L-Time n-th Pulse (L-Val) [µs]“ 39  „Time from H-Ena to n-th edge [µs]“ 46 „Time from L-Ena to n-th edge [µs]“ 45 „Time from Ena to n-th --/-- [µs]“ 43 „Time from Ena to n-th --\-- [µs]“ 44  „Additive Hightime [µs]“ 2 „Additive Lowtime [µs]“ 3  „Cycle Time --/-- [µs]“ 18 „Cycle Time --\-- [µs]“ 19 „Frequency --/-- [Hz]“ 4 „Frequency --\-- [Hz]“ 5 „Duty Factor L/H --/--“ 6 „Duty Factor L/H --\--“ 7 „Duty Factor H/L --/--“ 8 „Duty Factor H/L --\--“ 9 „Duty Cycle L/(L+H) --/--“ 10 „Duty Cycle L/(L+H) --\--“ 11 „Duty Cycle H/(L+H) --/--“ 12 „Duty Cycle H/(L+H) --\--“ 13  „Rising Edge --/-- [deg]“ 14 „Falling Edge --\-- [deg]“ 15 „Rising Edge of n-th Pulse [deg]“ 30 „Falling Edge of n-th Pulse [deg]“ 31  „Time Stamp --/-- [µs]“ 47 „Time Stamp --\-- [µs]“ 48  „Number of Low-Pulses“ 16 „Number of High-Pulses“ 17 „Total Number of L-Pulses“ 40 „Total Number of H-Pulses“ 41  „Level (Active High)“ 49 „Level (Active Low)“ 50 „Step Count (Step. Mot. Phase A)“ 42 „Step Count (Step. Mot. Phase B)“ 63
Reference Channel	sint8	Ja	Referenzkanal der Messung -1: Kein Referenzkanal 0 ... 63: Hardwarekanal 0 bis 63 64 ... 95: Messkanal 0 bis 31

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
(Max) Pulse (Count)	uint32	Ja	Bei puls- und flankenselektiven Messungen: Die Nummer des Pulses bzw. der Flanke, die vermessen werden soll. Bei additiven Messungen und Pulszählungen: Die Maximalzahl von zu berücksichtigenden Pulsen. Wertebereich: $\geq 0$
Timeout Monitoring	uint8	Ja	Timeout-Überwachung. 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“ 3: „CS Angle Predef“ 4: „CS Angle InpDep“ 5: „Intvl & CS Angle Predef“ 6: „Intvl & CS Angle InpDep“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant in den „... Predef“ Modi zur Timeout-Überwachung.
Timeout Interval [ms]	real32	Ja	Zeit zwischen zwei Timeout-Kontrollen in Millisekunden. Wertebereich: $\geq 1.0$ ms Nur relevant in den „Intvl ...“ Modi zur Timeout-Überwachung
CS Angle Timeout Check Point	real32	Ja	Kurbelwellenwinkel, bei dem Timeout-Kontrollen durchgeführt werden. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant in den „CS Angle ...“ Modi zur Timeout-Überwachung.
CS Angle Lower Limit	real32	Ja	Untere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Upper Limit	real32	Ja	Obere Grenze des zur Messung freigegebenen Winkelfensters in Grad Kurbelwelle. Wertebereich: $-720$ °KW bis $720$ °KW Nur relevant im drehzahlsynchronen Messmodus „ESpeedSync“.
CS Angle Reference	real32	Ja	Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung. Wertebereich: $-720$ °KW bis $720$ °KW
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 17-11** ES4330-Meas Device: Konfigurationsparameter der Registerkarte „Signals“

#### 17.4.16 Messverfahren - Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften von Pulsweitenmessungen beschrieben. Es wird dabei eine Unterscheidung in „puls-qualifizierte“ und „flanken-qualifizierte“ Messungen vorgenommen. Im einzelnen wird die Funktionsweise folgender Messungen erläutert:

##### *Puls-qualifizierte Pulsweitenmessungen*

- „Hightime (Pulse Qual.) [μs]“
- „Lowtime (Pulse Qual.) [μs]“
- „H-Time n-th Pulse (Pu Qual.) [μs]“
- „L-Time n-th Pulse (Pu Qual.) [μs]“

##### *Flanken-qualifizierte Pulsweitenmessungen*

- „Hightime (Edge Qual.) [μs]“
- „Lowtime (Edge Qual.) [μs]“
- „H-Time n-th Pulse (Eg Qual.) [μs]“
- „L-Time n-th Pulse (Eg Qual.) [μs]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Puls-Selektion (nur bei „H-Time n-th Pulse ...“ und „L-Time n-th Pulse ...“): „(Max) Pulse (Count)“ definiert die Nummer des zu vermessenden Pulses.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

##### *Funktionsbeschreibung: Drehzahlsynchrone Pulsweitenmessungen*

Bei den *puls-qualifizierten Messungen* werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen (gültiger Puls). Abb. 17-42 auf Seite 553 veranschaulicht die Funktionsweise einer puls-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [μs]“).

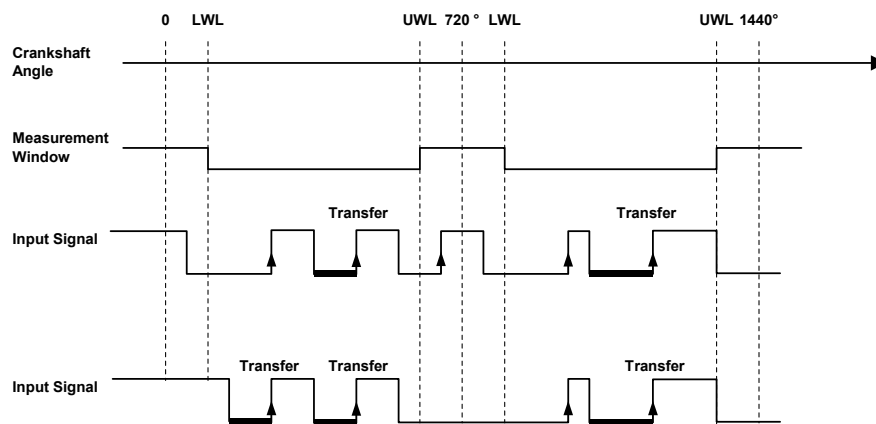
Bei den *flanken-qualifizierten Messungen* werden solche Pulse vermessen bei denen die öffnende oder die schließende Flanke innerhalb eines (beliebigen) Winkelfensters liegen (gültiger Puls). Pulse, bei denen beide Flanken außerhalb eines Winkelfensters liegen, werden nicht vermessen bzw. berücksichtigt. Abb. 17-43 auf Seite 553 veranschaulicht die Funktionsweise einer flanken-qualifizierten Messung anhand einer low-aktiven Pulsweitenmessung („Lowtime (Edge Qual.) [μs]“).

Bei den *nicht puls-selektiven Messungen* („Hightime (Pulse Qual.) [μs]“, „Lowtime (Pulse Qual.) [μs]“, „Hightime (Edge Qual.) [μs]“, „Lowtime (Edge Qual.) [μs]“) wird jeder gültige Puls vermessen, der Transfer der gemessenen Pulsweite in das DPRAM der ES4330-Karte erfolgt an der schließenden Flanke des Pulses.

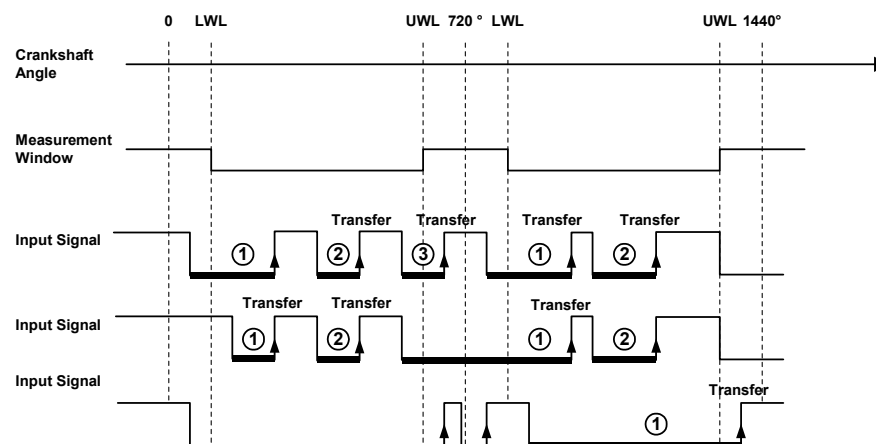
Bei den *puls-selektiven Messungen* („H-Time n-th Pulse (Pu Qual.) [μs]“, „L-Time n-th Pulse (Pu Qual.) [μs]“, „H-Time n-th Pulse (Eg Qual.) [μs]“, „L-Time n-th Pulse (Eg Qual.) [μs]“) wird der n-te gültige Puls innerhalb eines Winkelfensters gemessen. Die Pulsnummer wird mit dem Parameter „(Max) Pulse (Count)“ der Registerkarte „Signals“ definiert. Der Transfer der gemessenen Pulsweite erfolgt an der schließenden Flanke des n-ten gültigen Pulses.

**Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*



**Abb. 17-42** Darstellung der Funktionsweise einer puls-qualifizierten low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Pulse Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.



**Abb. 17-43** Darstellung der Funktionsweise einer flanken-qualifizierten low-aktiven Pulsweitenmessung (Messverfahren „Lowtime (Edge Qual.) [µs]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben. Die Zuordnung der Pulse zu den Winkelfenstern und die Zählung der Pulse innerhalb eines Winkelfensters ist ebenfalls dargestellt.

*Funktionsbeschreibung: Asynchrone Pulsweitenmessungen*

Bei den asynchronen Pulsweitenmessungen ist eine Unterscheidung in *puls-qualifizierte* und *flanken-qualifizierte Messungen* nicht möglich bzw. nicht nötig, da es keine Fensterfunktionalität zur Selektion von Pulsen gibt. Die puls-qualifizierten Pulsweitenmessungen haben im asynchronen Modus die gleiche Funktionalität wie die flanken-qualifizierten Messungen.

Die *nicht puls-selektiven Messungen* vermessen die Pulsweite jedes Pulses mit aktivem Pegel und transferieren den Messwert ins DPRAM an der schließenden Flanke des Pulses.

Die *puls-selektiven Pulsweitenmessungen* dürften im asynchronen Modus wohl keinen praktischen Anwendungsfall haben. Es wird die Pulsweite des n-ten Pulses mit aktivem Pegel nach der Initialisierung oder einer Konfiguration der ES4330-Karte vermessen. Der Messwert wird an der schließenden Flanke des vermessenen Pulses ins DPRAM transferiert.

#### **Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

#### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-12 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, so wird als Messwert die Zeitdauer seit der letzten öffnenden Flanke des Signals ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-12** Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.17 Messverfahren - Additive Pulsweitenmessungen

In diesem Abschnitt werden die Eigenschaften additiver Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen dargestellt:

- „Additive Hightime [µs]“
- „Additive Lowtime [µs]“

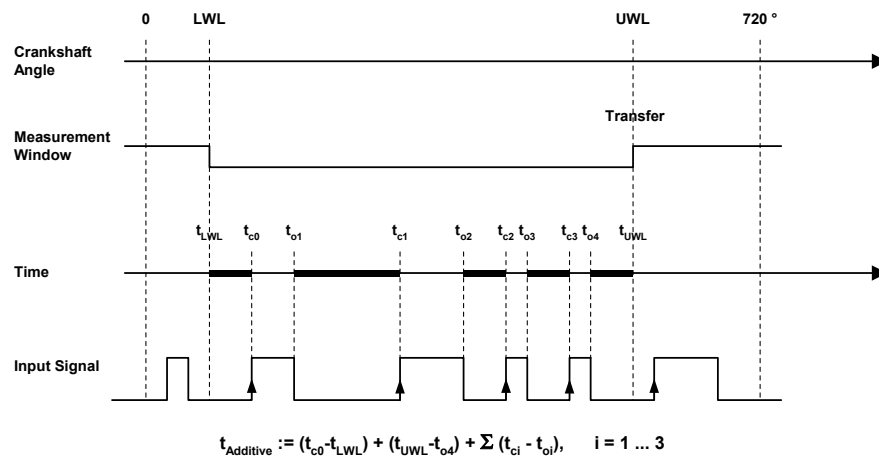
Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

### Funktionsbeschreibung: Drehzahlsynchrone additive Pulsweitenmessungen

Die additive Zeit ergibt sich als Summe aller Zeitsegmente innerhalb eines Winkelfensters, in denen das Signal aktiv ist, unabhängig davon, ob sich die öffnenden oder schließenden Flanken der Pulse innerhalb oder außerhalb eines Winkelfensters befinden. Abb. 17-44 zeigt beispielhaft die Messwertberechnung bei einer additiven Lowzeit-Messung.

Die additive Zeit wird an der oberen Winkelfenstergrenze in das DPRAM der ES4330-Karte transferiert.



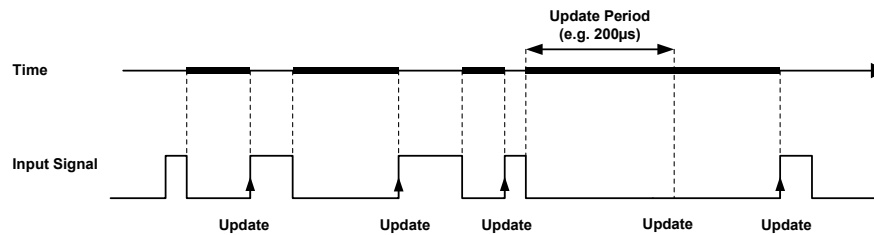
**Abb. 17-44** Darstellung der Funktionsweise einer drehzahlsynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [μs]“). Die additive Zeit setzt sich aus der Summe der fett markierten Liniensegmente zusammen.

### Funktionsbeschreibung: Asynchrone additive Pulsweitenmessungen

Bei asynchronen additiven Pulsweitenmessungen wird die Zeit summiert, in der das zu vermessende Signal aktiv ist. Die Aktualisierung der Summe im DPRAM erfolgt an jeder schließenden Pulsflanke spätestens jedoch nach einer vom Anwender definierten Aktualisierungs-Periode. Abb. 17-45 zeigt anhand einer additiven Lowzeit-Messung wann die Aktualisierung der additiven Zeit erfolgt. Die Einstellung der Aktualisierungs-Periode durch den Anwender erfolgt mit dem „Update Period for Additive Measurements [μs]“ Parameter in der Registerkarte „Globals“ des ES4330-XMI Subsystems. Eine Initialisierung oder eine Konfiguration der ES4330-Karte setzen die summierte Zeit auf 0 zurück.

#### **Hinweis**

Die Erfassung der additiven Zeit erfolgt modulo  $2^{52} \mu\text{s}$ , d. h., nach  $(2^{52}-1) \mu\text{s}$  (entspricht etwa 143 Jahren) springt die additive Zeit wieder auf 0 um.  $(2^{52}-1)$  ist die größte ganze Zahl des Typs „double precision“, die mit einer Auflösung von 1.0 darstellbar ist.



**Abb. 17-45** Darstellung der Funktionsweise einer asynchronen additiven Lowzeit-Messung (Messverfahren „Additive Lowtime [ $\mu\text{s}$ ]“). Die additive Zeit setzt sich aus den fett markierten Zeitsegmenten zusammen. Die Zeitpunkte an denen die Summe im DPRAM aktualisiert wird, sind ebenfalls dargestellt.

#### Timeout-Überwachung

Die Bedingung, wann ein Timeout ausgelöst wird, ist in Tab. 17-13 für verschiedene Verfahren zur Timeout-Überwachung aufgelistet. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-14 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „Intvl Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde <i>und</i> das Signal inaktiv war. Es spielt dabei im drehzahlsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„CS Angle InpDep“ „CS Angle Predef“	Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout keine Flanke (fallend oder steigend) auf dem Hardwarekanal bzw. Eingangskanal detektiert wurde. Es spielt dabei im drehzahlsynchronen Messmodus keine Rolle, ob die Flanken innerhalb oder außerhalb des Winkelfensters auftreten.
„Intvl & CS Angle InpDep“ „Intvl & CS Angle Predef“	Die Timeout-Bedingung für diese Überwachungsverfahren ergibt sich durch eine oder-Verknüpfung der Timeout-Bedingung der „Intvl InpDep“- und „Intvl Predef“-Verfahren mit der Timeout-Bedingung der „CS Angle InpDep“- und „CS Angle Predef“-Verfahren.

**Tab. 17-13** Additive Pulsweitenmessungen: Timeout-Bedingung für verschiedene Verfahren zur Timeout-Überwachung



Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Asynchroner Messmodus: Als Messwert im Timeout-Fall wird die bis zum Zeitpunkt der Überprüfung auf Timeout ermittelte additive Zeit ausgegeben. Drehzahlsynchroner Messmodus: Als Messwert im Timeout-Fall wird 0 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

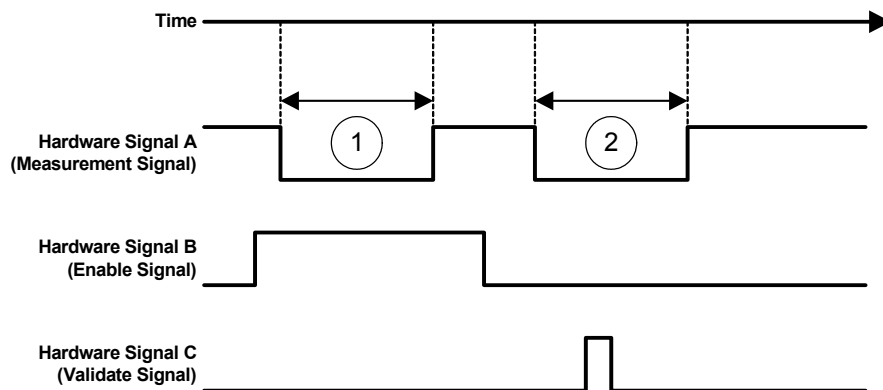
**Tab. 17-14** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung.

#### 17.4.18 Messverfahren - Pulsweitenmessungen mit Enable- oder Validierungssignal

Neben der direkten Erfassung der Pulsweiten eines von allen anderen Eingängen unabhängigen Signals bietet die ES4330-Karte noch die Möglichkeit, Pulsweiten eines Signals in Abhängigkeit eines zweiten Eingangs zu messen. Dieses zweite Eingangssignal wird hierbei als Enable- bzw. Validate-Signal interpretiert, wie in Abb. 17-46 auf Seite 558 am Beispiel einer validierten Low-Zeit-Messung skizziert ist:

Wird die ES4330-Karte beispielsweise derart konfiguriert, dass Low-Zeiten eines Eingangssignals A vermessen werden sollen, wobei ein zweites Eingangssignal B als high-aktives Enable-Signal genutzt werden soll, so würde vom Eingangssignal aus Abb. 17-46 auf Seite 558 lediglich der mit 1 gekennzeichnete Puls vermessen. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Enable“ werden ausschließlich die Pulse vermessen, die *vollständig* von einem aktiven Enable-Puls *umschlossen* sind.

Wird hingegen ein weiteres Eingangssignal C als high-aktives Validate-Signal genutzt, so würde im obigen Beispiel lediglich der mit 2 gekennzeichnete Puls vermessen werden. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Validate“ werden ausschließlich die Pulse vermessen, die ihrerseits einen aktiven Validate-Puls *vollständig umschließen*.



**Abb. 17-46** Validierte Zeitmessungen mit der ES4330-Karte

Der ES4330 RTIO-Treiber bietet Pulsweitenmessungen mit Enable- oder Validate-Option an. Die Enable- und Validate-Pulse können dabei wahlweise als high-aktiv oder auch als low-aktiv definiert werden. Im einzelnen wird in diesem Abschnitt die Funktionsweise folgender Messungen erläutert:

- „Hightime using H-Enable [ $\mu$ s]“
- „Hightime using L-Enable [ $\mu$ s]“
- „Hightime using H-Validate [ $\mu$ s]“
- „Hightime using L-Validate [ $\mu$ s]“
- „Lowtime using H-Enable [ $\mu$ s]“
- „Lowtime using L-Enable [ $\mu$ s]“
- „Lowtime using H-Validate [ $\mu$ s]“
- „Lowtime using L-Validate [ $\mu$ s]“
- „H-Time n-th Pulse (H-Ena) [ $\mu$ s]“
- „H-Time n-th Pulse (L-Ena) [ $\mu$ s]“
- „H-Time n-th Pulse (H-Val) [ $\mu$ s]“
- „H-Time n-th Pulse (L-Val) [ $\mu$ s]“
- „L-Time n-th Pulse (H-Ena) [ $\mu$ s]“
- „L-Time n-th Pulse (L-Ena) [ $\mu$ s]“
- „L-Time n-th Pulse (H-Val) [ $\mu$ s]“
- „L-Time n-th Pulse (L-Val) [ $\mu$ s]“

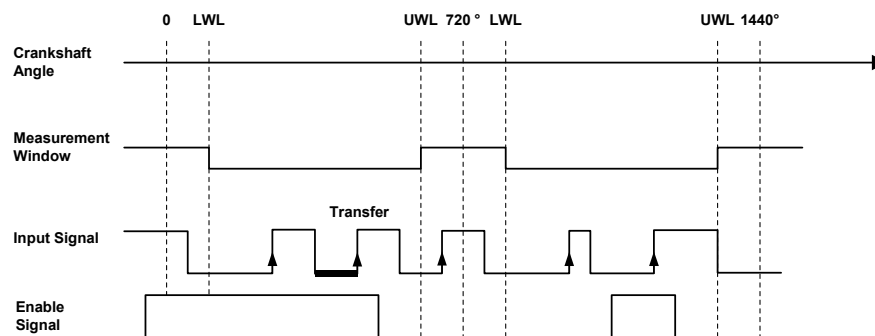
Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- „Reference Channel“: Hardwarekanal, an dem das Enable- bzw. Validate-Signal anliegt.
- Puls-Selektion (nur bei „H-Time n-th Pulse ...“ und „L-Time n-th Pulse ...“): „(Max) Pulse (Count)“ definiert die Nummer des zu vermessenden Pulses.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.

- Timeout-Überwachung

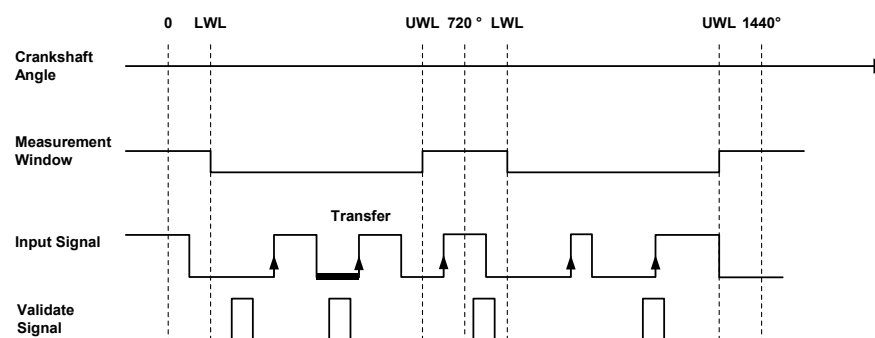
*Funktionsbeschreibung: Drehzahlsynchrone Pulsweitenmessungen mit Enable- oder Validierungssignal*

Abb. 17-47 veranschaulicht die Funktionsweise von drehzahlsynchronen Pulsweitenmessungen mit Enable-Option anhand einer Low-Zeit-Messung mit high-aktivem Enable-Signal. Es werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig von einem aktiven Enable-Puls umschlossen sind (gültiger Puls).



**Abb. 17-47** Darstellung der Funktionsweise einer drehzahlsynchronen Pulsweitenmessung mit Enable-Signal (Messverfahren „Lowtime using H-Enable [ $\mu\text{s}$ ]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.

Abb. 17-48 veranschaulicht die Funktionsweise von drehzahlsynchronen Pulsweitenmessungen mit Validate-Option anhand einer Low-Zeit-Messung mit high-aktivem Validate-Signal. Es werden nur solche Pulse vermessen, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkelfensters liegen und die vollständig einen aktiven Validate-Puls umschließen (gültiger Puls).



**Abb. 17-48** Darstellung der Funktionsweise einer drehzahlsynchronen Pulsweitenmessung mit Validate-Signal (Messverfahren „Lowtime using H-Validate [ $\mu\text{s}$ ]“). Vermessene Pulse sind mit erhöhter Linienstärke hervorgehoben.

Bei den *nicht puls-selektiven Messungen* wird jeder gültige Puls vermessen - der Transfer der gemessenen Pulsweite in das DPRAM der ES4330-Karte erfolgt an der schließenden Flanke des Pulses.

Bei den *puls-selektiven Messungen* wird der n-te gültige Puls innerhalb eines Winkelfensters gemessen. Die Pulsnummer wird mit dem Parameter „(Max) Pulse (Count)“ der Registerkarte „Signals“ definiert. Der Transfer der gemessenen Pulsweite erfolgt an der schließenden Flanke des n-ten gültigen Pulses.

#### **Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer „0“ eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

*Funktionsbeschreibung: Asynchrone Pulsweitenmessungen mit Enable- oder Validierungssignal*

Die *nicht puls-selektiven Messungen* vermessen die Pulsweite jedes durch ein Enable- oder Validate-Signal validierten Pulses mit aktivem Pegel und transferieren den Messwert ins DPRAM an der schließenden Flanke des Pulses.

Die *puls-selektiven Pulsweitenmessungen* dürften im asynchronen Modus wohl keinen praktischen Anwendungsfall haben. Es wird die Pulsweite des n-ten durch ein Enable- oder Validate-Signal validierten Pulses mit aktivem Pegel nach der Initialisierung oder einer Konfiguration der ES4330-Karte vermessen. Der Messwert wird an der schließenden Flanke des vermessenen Pulses ins DPRAM transferiert.

#### **Hinweis**

*Wird bei den puls-selektiven Messungen als Pulsnummer „0“ eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht puls-selektiven Messungen.*

*Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-15 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, so wird als Messwert die Zeitdauer seit der letzten öffnenden Flanke des Signals ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-15** Pulsweitenmessungen mit Enable- oder Validate-Signal: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.19 Messverfahren - Frequenz- und Zykluszeitmessungen

Die ES4330-Karte ermöglicht Messungen von Frequenzen und Zykluszeiten an steigenden und fallenden Signalflanken. Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Messverfahren zur Verfügung:

- „Cycle Time --/-- [ $\mu\text{s}$ ]“
- „Cycle Time --\-- [ $\mu\text{s}$ ]“
- „Frequency --/-- [Hz]“
- „Frequency --\-- [Hz]“

Zulässige Messoptionen sind:

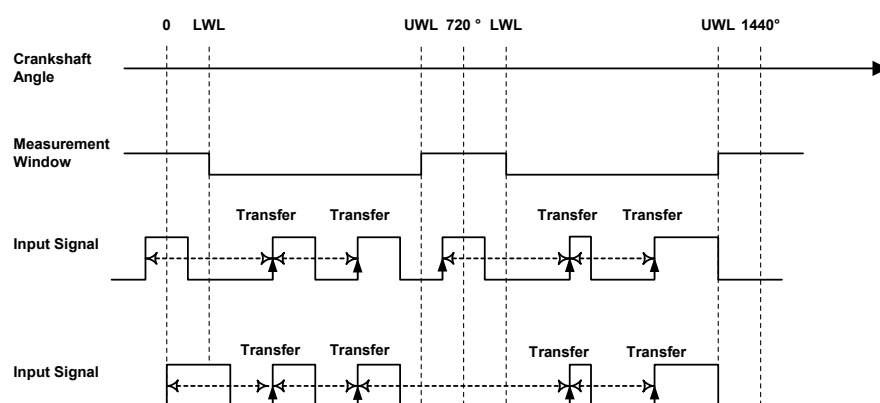
- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

#### Funktionsbeschreibung

Im asynchronen Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalflanke. Im drehzahlsynchronen (winkelsegmentierten) Messmodus erfolgt die Messwertberechnung an jeder aktiven Signalflanke, die innerhalb eines Winkelfensters liegt. Abb. 17-49 veranschaulicht die Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalflanken.

#### Hinweis

Wird bei einer winkelsegmentierten Zykluszeit- oder Frequenzmessung der gesamte Winkelbereich von  $720^\circ$  KW freigeschaltet, so verhält sich diese Messung wie die entsprechende Messung im asynchronen Modus.



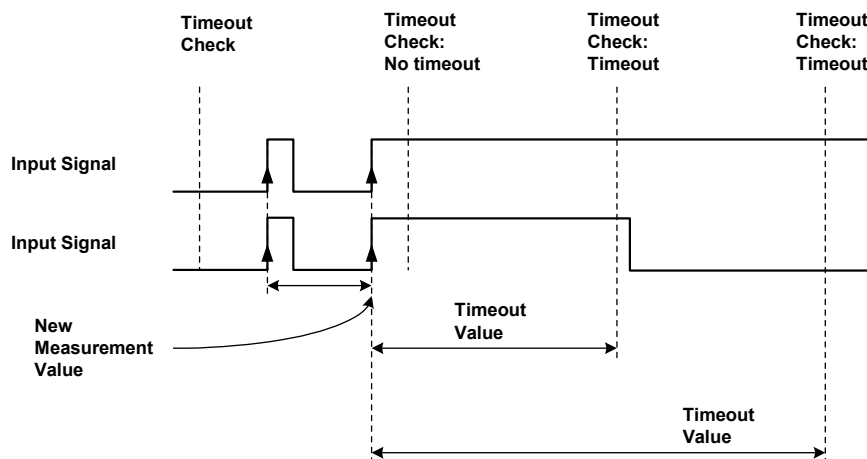
**Abb. 17-49** Darstellung der Messwertberechnung bei einer winkelsegmentierten Frequenz- oder Zykluszeitmessung an steigenden Signalflanken

### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-16 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	<p><i>Zykluszeitmessungen:</i> Als Messwert wird im Timeout-Fall die Zeitdauer zwischen dem Zeitpunkt des Auftretens der letzten aktiven Flanke des Eingangssignals und dem Zeitpunkt, an dem die Überprüfung auf Timeout durchgeführt wird, ausgegeben (siehe Abb. 17-50).</p> <p><i>Frequenzmessungen:</i> Die Messwertberechnung im Timeout-Fall erfolgt wie bei den Zykluszeitmessungen, jedoch wird aus der ermittelten Zeitdauer durch Kehrwertbildung eine resultierende Frequenz ermittelt.</p>
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-16** Frequenz- und Zykluszeitmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung



**Abb. 17-50** Frequenz- und Zykluszeitmessungen an ansteigenden Flanken: Darstellung der Messwertberechnung im Timeout-Fall (der Messwert ist in der Darstellung als „Timeout Value“ bezeichnet) bei den „... InpDep“-Verfahren zur Timeout-Überwachung

#### 17.4.20 Messverfahren - Tastverhältnismessungen

Tastverhältnisse (z.B. von PWM-Signalen) können mit der ES4330-Karte auf unterschiedlichste Arten ermittelt werden. Bei den Tastverhältnismessungen wird unterschieden zwischen Messungen, die die High-Phase eines Signals ins Verhältnis zur Low-Phase setzen (oder umgekehrt) und Messungen, die die High-Phase (oder die Low-Phase) eines Signals ins Verhältnis zur Zyklusdauer setzen. Der erste Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis<sup>P/P</sup> bezeichnet, „P/P“ soll dabei andeuten, dass zwei Pulsdauern zueinander ins Verhältnis gesetzt werden.

Der zweite Typ von Tastverhältnismessung wird im folgenden als Tastverhältnis<sup>P/C</sup> bezeichnet, „P/C“ soll andeuten, dass hier eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt wird.

$$\text{Tastverhältnis } \overset{P/P}{P/P} : \frac{L}{H}, \frac{H}{L} \quad 0 \leq \text{Tastverhältnis } \overset{P/P}{P/P} \leq \infty$$

$$\text{Tastverhältnis } \overset{P/C}{P/C} : \frac{L}{L+H}, \frac{H}{L+H} \quad 0 \leq \text{Tastverhältnis } \overset{P/C}{P/C} \leq 1$$

Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest. Es stehen die nachfolgenden Tastverhältnismessungen vom Typ „P/P“ zur Verfügung:

- „Duty Factor L/H --/--“
- „Duty Factor L/H --\--“
- „Duty Factor H/L --/--“
- „Duty Factor H/L --\--“

Es stehen die folgenden Tastverhältnismessungen vom Typ „P/C“ zur Verfügung:

- „Duty Cycle L/(L+H) --/--“
- „Duty Cycle L/(L+H) --\--“
- „Duty Cycle H/(L+H) --/--“
- „Duty Cycle H/(L+H) --\--“

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

##### Funktionsbeschreibung

Was die Messwertberechnung im asynchronen und drehzahlsynchronen (winkelsegmentierten) Messmodus anbelangt, so gilt bei Tastverhältnismessungen exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 17.4.19 auf Seite 561 beschrieben wurde.

##### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-17 auf Seite 564 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Tastverhältnis $P/P$ : $\frac{L}{H}, \frac{H}{L}$ <ul style="list-style-type: none"> <li>• Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/H-Messungen oder „low“ bei H/L-Messungen ist.</li> <li>• Der Messwert ist MAXREAL32, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/H-Messungen oder „high“ bei H/L-Messungen ist.</li> </ul>
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Tastverhältnis $P/C$ : $\frac{L}{L+H}, \frac{H}{L+H}$ <ul style="list-style-type: none"> <li>• Der Messwert ist 0.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „high“ bei L/(L+H)-Messungen oder „low“ bei H/(L+H)-Messungen ist.</li> <li>• Der Messwert ist 1.0, falls der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout „low“ bei L/(L+H)-Messungen oder „high“ bei H/(L+H)-Messungen ist.</li> </ul>
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-17** Tastverhältnismessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

17.4.21 Messverfahren - Relativmessungen zwischen Hardwarekanälen

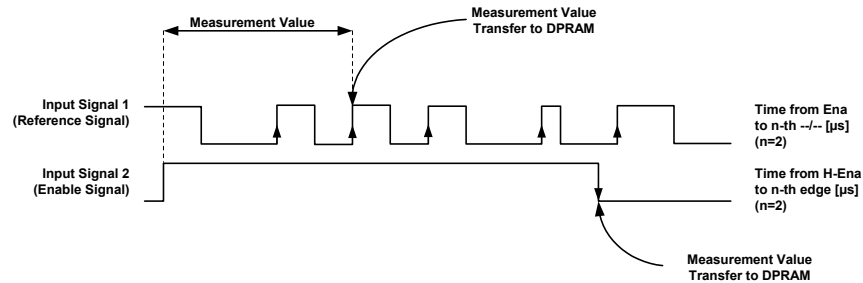
*Funktionsbeschreibung*

Den bisherigen Messverfahren war gemeinsam, dass die vermessenen Zeiten, Frequenzen, Winkel und Tastverhältnisse durch die Signalfanken eines Hardwarekanals definiert waren. In diesem Abschnitt werden Messverfahren beschrieben, in denen Ereignisse (Flanken) auf zwei Hardwarekanälen relativ zueinander vermessen werden.

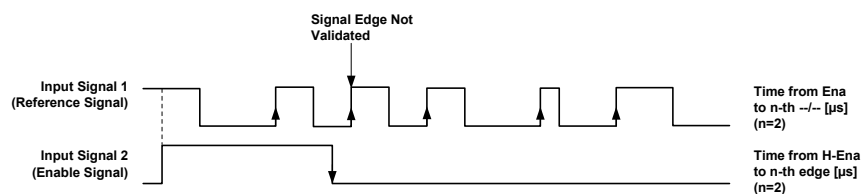
Die Messfunktionalität soll anhand von Abb. 17-51 erläutert werden. Vermessen wird die Zeit von der öffnenden Flanke eines Enable-Pulses bis zur n-ten aktiven Flanke auf einem Referenz-Signal. Voraussetzung für die Ermittlung eines Messwerts ist jedoch, dass die n-te aktive Flanke durch den Enable-Puls validiert wird, d. h., die Flanke muss während der aktiven Phase des Enable-Signals auftreten. Die Zählung der aktiven Flanken des Referenz-Signals wird mit der öffnenden Flanke des Enable-Pulses auf 0 zurückgesetzt. Der Parameter n zur Flankenselek-



tion wird im Optionsfeld „(Max) Pulse (Count)“ in der Registerkarte „Signals“ eingestellt. Abb. 17-52 zeigt einen Signalverlauf, bei dem keine Validierung des Referenz-Signals vorliegt.



**Abb. 17-51** Funktionsweise der Messverfahren zur Vermessung von Ereignissen (Flanken) auf unterschiedlichen Signalen bzw. Hardwarekanälen relativ zueinander. Dargestellt ist eine Messung mit high-aktivem Enable-Signal. Der RTIO-Parameter zur Flanken-Selektion („(Max) Pulse (Count)“ in der Registerkarte „Signals“) ist größer als 0.



**Abb. 17-52** Vermessung von Flanken auf unterschiedlichen Hardwarekanälen: Signalverlauf mit high-aktivem Enable-Signal und nicht validierter n-ter aktiver Signalfanke des Referenz-Signals.

Die Durchführung der beschriebenen Messungen erfordert grundsätzlich die Konfiguration zweier Mess- bzw. Softwarekanäle. Einem Messkanal ist das Enable-Signal zuzuordnen. In Abhängigkeit davon, ob das Enable-Signal low-aktiv oder high-aktiv ist, ist dem Messkanal eines der beiden nachfolgenden Messverfahren zuzuweisen:

- „Time from L-Ena to n-th edge [µs]“
- „Time from H-Ena to n-th edge [µs]“

Zulässige Messoptionen sind:

- „Reference Channel“: Verweist auf den Messkanal, dem das Referenz-Signal zugeordnet ist.
- Timeout-Überwachung

Die Einstellungen aller anderen Messoptionen werden ignoriert.

Dem anderen Messkanal ist das Referenz-Signal zuzuordnen. In Abhängigkeit davon, ob die aktive Flanke des Referenz-Signals ansteigend oder fallend ist, ist dem Messkanal eines der beiden nachfolgenden Messverfahren zuzuweisen:

- „Time from Ena to n-th --/-- [µs]“

- „Time from Ena to n-th --\-- [ $\mu$ s]“

Zulässige Messoptionen sind:

- „Reference Channel“: Verweist auf den Messkanal, dem das Enable-Signal zugeordnet ist.
- Flanken-Selektion: „(Max) Pulse (Count)“ definiert die Nummer der zu vermessenden aktiven Flanke des Referenz-Signals.

Die Einstellungen aller anderen Messoptionen werden ignoriert.

#### **Hinweis**

*Winkelsegmentierung über Winkelfenster wird nicht unterstützt. Aus diesem Grund gibt es keinen Unterschied in der Funktionalität der Messungen im asynchronen und im drehzahlsynchronen Messmodus.*

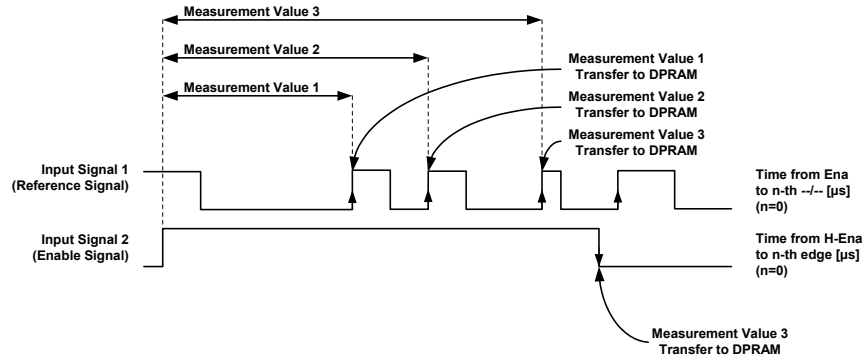
Grundsätzlich liefern beide Messkanäle einer Messung den ermittelten Messwert. Nur der Zeitpunkt, an dem der Messwert in das DPRAM der ES4330-Karte übertragen wird, ist bei den beiden Messkanälen unterschiedlich.

Messkanäle, die auf die Messverfahren „Time from Ena to n-th --/-- [ $\mu$ s]“ oder „Time from Ena to n-th --\-- [ $\mu$ s]“ konfiguriert sind, übertragen den Messwert zum Zeitpunkt des Auftretens der n-ten validierten aktiven Flanke (siehe Abb. 17-51).

Messkanäle, die auf die Messverfahren „Time from L-Ena to n-th edge [ $\mu$ s]“ oder „Time from H-Ena to n-th edge [ $\mu$ s]“ konfiguriert sind, übertragen den Messwert zum Zeitpunkt des Auftretens der schließenden Flanke des Enable-Pulses (siehe Abb. 17-51).

Ein besonderes Messverhalten ergibt sich, wenn der Parameter zur Flanken-Selektion „(Max) Pulse (Count)“ in den Messverfahren „Time from Ena to n-th --/-- [ $\mu$ s]“ oder „Time from Ena to n-th --\-- [ $\mu$ s]“ zu 0 gesetzt wird. In diesem Fall liefern die Messverfahren „Time from L-Ena to n-th edge [ $\mu$ s]“ und „Time from H-Ena to n-th edge [ $\mu$ s]“ die Zeit von der öffnenden Flanke des Enable-Pulses bis zur *letzten* validierten aktiven Flanke des Referenz-Signals. Die Übertragung des Messwerts in das DPRAM erfolgt *an der schließenden Flanke des Enable-Pulses* (Abb. 17-53). Die Messverfahren „Time from Ena to n-th --/-- [ $\mu$ s]“ und „Time from Ena to n-th --\-- [ $\mu$ s]“ liefern in diesem Fall die Zeit von der öffnenden

Flanke des Enable-Pulses zu *jeder* validierten aktiven Flanke des Referenz-Signals. Die Übertragung der Messwerte ins DPRAM erfolgt zum Zeitpunkt des Auftretens der jeweiligen validierten aktiven Flanke (Abb. 17-53).



**Abb. 17-53** Funktionsweise der Messverfahren zur Vermessung von Ereignissen (Flanken) auf unterschiedlichen Signalen bzw. Hardwarekanälen relativ zueinander. Dargestellt ist eine Messung mit high-aktivem Enable-Signal. Der RTIO-Parameter zur Flanken-Selektion („(Max) Pulse (Count)“ in der Registerkarte „Signals“) ist gleich 0.

Tab. 17-18 fasst Konfigurationsrichtlinien und Messwerte der Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen zusammen.

RTIO-Parameter der Registerkarte „Signals“	Enable-Signal	Referenz-Signal
Hardware Channel	Nummer des Hardwarekanals, der das Enable-Signal führt.	Nummer des Hardwarekanals, der das Referenz-Signal führt.
Measurement Mode	„Asynchronous“ oder „ESpeedSync“. Es gibt keine Unterschiede zwischen den Modi.	„Asynchronous“ oder „ESpeedSync“. Es gibt keine Unterschiede zwischen den Modi.
Measurement Procedure	„Time from L-Ena to n-th edge [ $\mu$ s]“ oder „Time from H-Ena to n-th edge [ $\mu$ s]“ je nachdem, ob das Enable-Signal low- oder high-aktiv ist.	„Time from Ena to n-th --/-- [ $\mu$ s]“ oder „Time from Ena to n-th --\-- [ $\mu$ s]“ je nachdem, ob die aktive Flanke des Referenz-Signals steigend oder fallend ist.
Reference Channel	Nummer des Messkanals, dem das Referenz-Signal zugewiesen wurde.	Nummer des Messkanals, dem das Enable-Signal zugewiesen wurde.
Messwert bei Einstellung (Max) Pulse (Count) = 0	Die Zeit von der öffnenden Flanke des Enable-Pulses zur <i>letzten</i> validierten aktiven Flanke des Referenz-Signals wird gemessen und <i>an der schließenden Flanke des Enable-Pulses</i> ins DPRAM übertragen.	Die Zeit von der öffnenden Flanke des Enable-Pulses zu <i>jeder</i> validierten aktiven Flanke des Referenz-Signals wird gemessen und <i>zum Zeitpunkt des Auftretens der jeweiligen aktiven Flanke</i> ins DPRAM übertragen.
Messwert bei Einstellung (Max) Pulse (Count) > 0	Die Zeit von der öffnenden Flanke des Enable-Pulses bis zur n-ten validierten aktiven Flanke des Referenz-Signals wird gemessen und <i>an der schließenden Flanke des Enable-Pulses</i> ins DPRAM übertragen.	Die Zeit von der öffnenden Flanke des Enable-Pulses bis zur n-ten validierten aktiven Flanke des Referenz-Signals wird gemessen und <i>zum Zeitpunkt des Auftretens der n-ten validierten aktiven Flanke</i> ins DPRAM übertragen.

**Tab. 17-18** Konfigurationsrichtlinien und Messwerte der Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-19 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Falls der Signalpegel des Enable-Signals zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Falls der Signalpegel des Enable-Signals zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-19** Messverfahren zur Vermessung von Flanken auf unterschiedlichen Hardwarekanälen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.22 Messverfahren - Vermessung von Flanken: Winkelstempel

---

### Funktionsbeschreibung

---

Die nachfolgenden Messverfahren vermessen den Kurbelwellenwinkel aktiver Signalfanken auf dem zugeordneten Hardwarekanal relativ zum eingestellten Referenzwinkel.

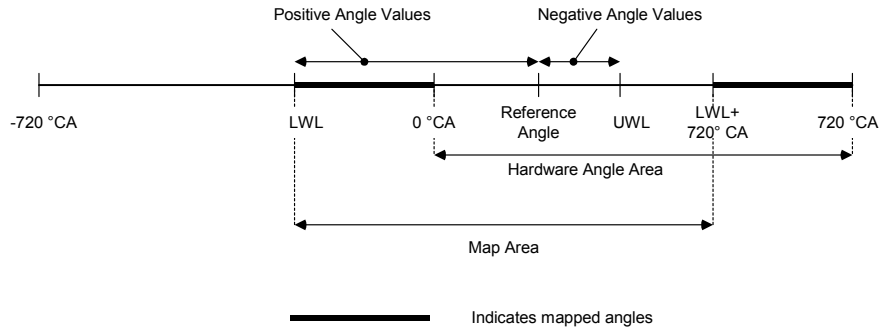
- „Rising Edge --/-- [deg]“
- „Falling Edge --\-- [deg]“
- „Rising Edge of n-th Pulse [deg]“
- „Falling Edge of n-th Pulse [deg]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Flanken-Selektion (nur bei „Rising Edge of n-th Pulse [deg]“ und „Falling Edge of n-th Pulse [deg]“): „(Max) Pulse (Count)“ definiert die Nummer der zu vermessenden Flanke.
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- „CS Angle Reference“: Definiert den Referenzwinkel der Messung.
- Timeout-Überwachung

Der von der Hardware gelieferte Kurbelwellenwinkel einer Flanke liegt grundsätzlich im Bereich [0 °KW, 720 °KW]. Im *drehzahlsynchronen Messmodus* wird der gemessene Winkel von der ES4330-Firmware in den Bereich [LWL, LWL + 720 °KW] abgebildet und anschließend die Differenz mit dem Referenzwinkel

berechnet. Diese Differenz ist der von der Messung gelieferte Messwert, sie ist positiv, wenn der abgebildete Winkel kleiner als der Referenzwinkel ist (vgl. Abb. 17-54).



**Abb. 17-54** Funktionsweise einer Winkelmessung im drehzahlsynchronen Messmodus

Da der Referenzwinkel im Bereich  $[-720^\circ \text{KW}, 720^\circ \text{KW}]$  liegen kann, liefern die Messfunktionen im drehzahlsynchronen Messmodus Winkelwerte aus dem Bereich  $]-1440^\circ \text{KW}, 1440^\circ \text{KW}]$ . Im *asynchronen Messmodus* findet keine Abbildung der von der Hardware gelieferten Kurbelwellenwinkel statt, so dass in diesem Messmodus die Winkelwerte im Bereich  $]-1440^\circ \text{KW}, 720^\circ \text{KW}]$  liegen.

Im *drehzahlsynchronen Messmodus* erfassen die *nicht flanken-selektiven Messungen* („Rising Edge --/-- [deg]“, „Falling Edge --\-- [deg]“) jede aktive Flanke innerhalb eines Winkelfensters. Die *flanken-selektiven Messungen* („Rising Edge of n-th Pulse [deg]“, „Falling Edge of n-th Pulse [deg]“) vermessen die n-te aktive Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenähler auf 0 zurückgesetzt.

Im *asynchronen Messmodus* erfassen die *nicht flanken-selektiven Messungen* den Kurbelwellenwinkel jeder aktiven Flanke. Die *flanken-selektiven Messungen* ergeben in diesem Messmodus wenig Sinn, da der Flankenähler nicht zurückgesetzt wird.

#### Hinweis

Wird bei den flanken-selektiven Messungen im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt, so verhalten sich diese Messungen wie die jeweiligen nicht flanken-selektiven Messungen.

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-20 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der Low-Pegel, bei einer Messung an fallenden Flanken der High-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Messung an ansteigenden Flanken der High-Pegel, bei einer Messung an fallenden Flanken der Low-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-20** Winkelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.23 Messverfahren - Vermessung von Flanken: Zeitstempel

---

##### Funktionsbeschreibung

---

Die nachfolgenden Messverfahren erfassen den Zeitpunkt aktiver Signalfanken auf dem zugeordneten Hardwarekanal.

- „Time Stamp --/-- [µs]“
- „Time Stamp --\-- [µs]“

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- „(Max) Pulse (Count)“: Definiert die Nummer der zu vermessenden Flanke
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

Im *drehzahlsynchronen Messmodus* wird der Zeitpunkt des Auftretens der n-ten aktiven Signalfanke innerhalb eines Winkelfensters gemessen. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenzähler auf 0 zurückgesetzt. Wird im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt, so wird jede aktive Flanke innerhalb eines Winkelfensters gemessen.

Im *asynchronen Messmodus* sollte im Optionsfeld „(Max) Pulse (Count)“ als Flankennummer 0 eingestellt sein, dann wird der Zeitpunkt des Auftretens jeder aktiven Signalflanke gemessen. Ist die Flankennummer ungleich 0, so wird einmalig der Zeitpunkt der n-ten aktiven Flanke nach einer Initialisierung oder einem Konfigurationsschritt gemessen.

Die Zeiten werden relativ zum Zeitpunkt der letzten Initialisierung oder Konfiguration der ES4330-Karte angegeben. Mit anderen Worten: Eine Initialisierung oder Konfiguration der ES4330-Karte setzt die Uhr auf 0 zurück.

#### *Timeout-Überwachung*

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-21 angegeben.

<b>Timeout-Überwachung</b>	<b>Messwert im Timeout-Fall</b>
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert im Timeout-Fall wird der letzte (vor dem Timeout berechnete) Messwert beibehalten.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-21** Zeitstempelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.24 Messverfahren - Pulszählung

In diesem Abschnitt werden die Eigenschaften folgender Messverfahren zur Pulszählung beschrieben:

- „Number of Low-Pulses“
- „Number of High-Pulses“
- „Total Number of L-Pulses“
- „Total Number of H-Pulses“

Zulässige Messoptionen sind:

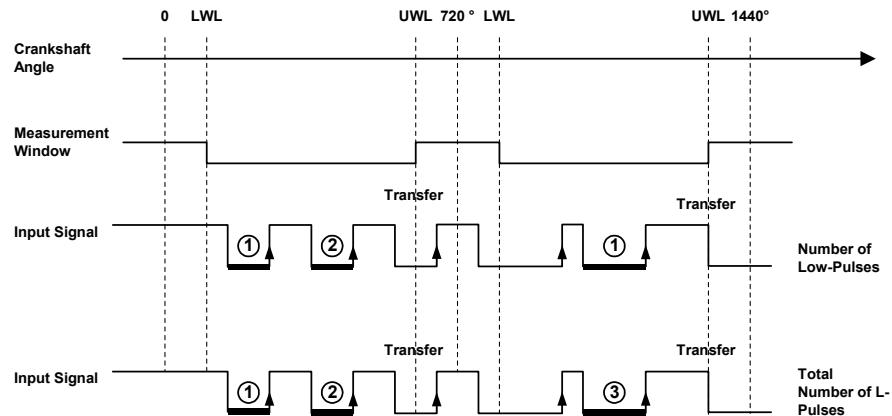
- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

#### *Funktionsbeschreibung*

Die Messverfahren „Number of Low-Pulses“ und „Number of High-Pulses“ erfassen im *drehzahlsynchronen Messmodus* die Anzahl der gültigen aktiven Pulse innerhalb eines Winkelfensters. Gültige Pulse sind Pulse, die mit öffnender und schließender Flanke in ein- und demselben Winkelfenster liegen. Die ermittelte Summe wird an der oberen Winkelfenstergrenze in das DPRAM übertragen. Die Messverfahren „Total Number of L-Pulses“ und „Total Number of H-Pulses“



erfassen im drehzahlsynchronen Messmodus die Summe aller gültigen Pulse seit der letzten Initialisierung oder Konfiguration der ES4330-Karte. Die Summe wird an jeder oberen Winkelfenstergrenze aktualisiert (Abb. 17-55).



**Abb. 17-55** Darstellung der Funktionsweise drehzahlsynchroner Messverfahren zur Pulszählung (Messverfahren „Number of Low-Pulses“ und „Total Number of L-Pulses“). Pulse die gezählt wurden sind durch erhöhte Linienstärke hervorgehoben. Die Summe wird jeweils an der oberen Winkelfenstergrenze in das DPRAM transferiert.

Im *asynchronen Messmodus* besitzen die „Number of ...“ und „Total Number of ...“ Messungen identische Funktionalität. Es wird die Gesamtzahl der aktiven Pulse seit der letzten Initialisierung oder Konfiguration der ES4330-Karte gemessen. Die Summe wird an jeder schließenden Pulsflanke aktualisiert.

#### **Hinweis**

Die Pulszählung erfolgt modulo  $2^{52}$ , d. h., nach  $(2^{52}-1)$  Pulsen springt die Summe wieder auf 0 um.  $(2^{52}-1)$  ist die größte ganze Zahl vom Typ „double precision“, die mit einer Auflösung von 1.0 darstellbar ist.

#### Timeout-Überwachung

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein gültiger aktiver Puls aufgetreten ist. Im drehzahlsynchronen Messmodus ist ein gültiger aktiver Puls ein aktiver Puls, dessen öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen. Im asynchronen Messmodus ist jeder aktive Puls ein gültiger aktiver Puls. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-22 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. Ein inaktiver Signalpegel ist bei einer Zählung von Low-Pulsen der High-Pegel und bei einer Zählung von High-Pulsen der Low-Pegel. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben. Ein aktiver Signalpegel ist bei einer Zählung von Low-Pulsen der Low-Pegel und bei einer Zählung von High-Pulsen der High-Pegel.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-22** Pulszählung: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.25 Messverfahren - Pegelmessung

##### *Funktionsbeschreibung*

Die Messverfahren

- „Level (Active High)“
- „Level (Active Low)“

dienen zur Erfassung der aktiven Pegelzustände auf einem Hardwarekanal. Ein aktiver Pegel wird durch den Messwert „Eins“ (1), ein inaktiver Pegel durch den Messwert „Null“ (0) signalisiert. Im *asynchronen Messmodus* wird der Messwert an jeder Flanke des Eingangssignals aktualisiert. Im *drehzahlsynchronen Messmodus* findet eine Aktualisierung nur an Signalfanken statt, die innerhalb des Winkelfensters auftreten.

Zulässige Messoptionen sind:

- „Measurement Mode“: Drehzahlsynchroner oder asynchroner Messmodus
- Winkelfenster („CS Angle Lower Limit“ und „CS Angle Upper Limit“) im drehzahlsynchronen Messmodus.
- Timeout-Überwachung

### Timeout-Überwachung

---

Ein Timeout wird ausgelöst, falls seit der letzten Überprüfung auf Timeout kein Messwert berechnet bzw. ermittelt werden konnte. Der Messwert, der im Timeout-Fall ausgegeben wird, ist in Tab. 17-23 angegeben.

Timeout-Überwachung	Messwert im Timeout-Fall
„Intvl InpDep“ „CS Angle InpDep“ „Intvl & CS Angle InpDep“	Als Messwert wird der Signalpegel zum Zeitpunkt der Überprüfung auf Timeout ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung inaktiv ist, wird als Messwert 0 ausgegeben. - Falls der Signalpegel zum Zeitpunkt der Timeout-Überprüfung aktiv ist, wird als Messwert 1 ausgegeben.
„Intvl Predef“ „CS Angle Predef“ „Intvl & CS Angle Predef“	Es wird der im Optionsfeld „Default Timeout Value“ eingestellte Wert als Messwert ausgegeben.

**Tab. 17-23** Pegelmessungen: Messwert im Timeout-Fall bei verschiedenen Verfahren zur Timeout-Überwachung

#### 17.4.26 Messverfahren - Positionsverfolgung an zwei-phasigen Schrittmotoren

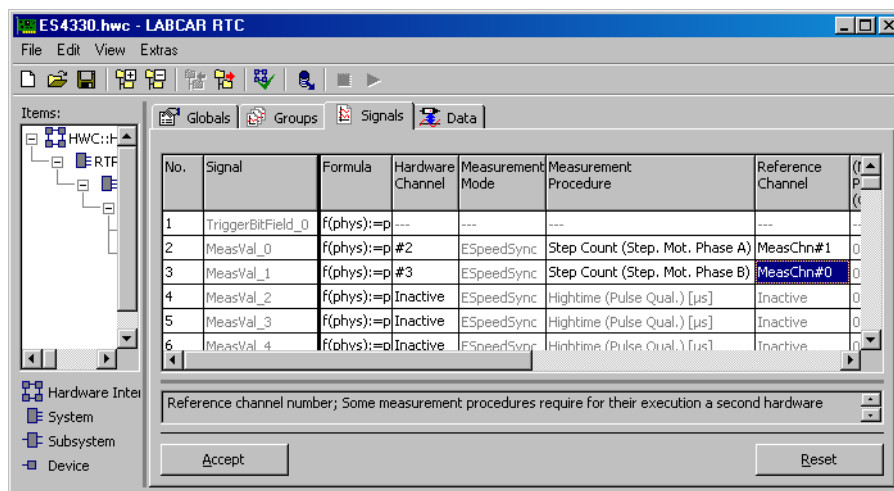
---

##### Funktionsbeschreibung

---

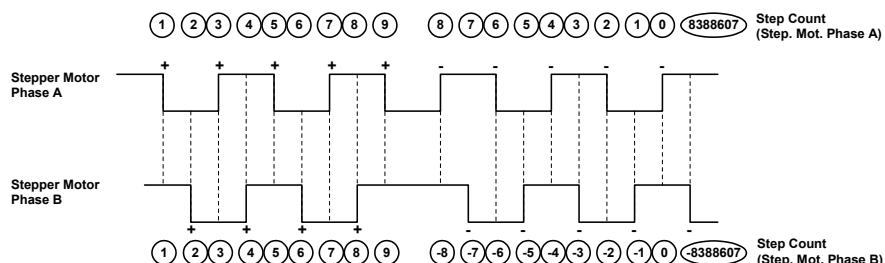
Die ES4330-Firmware ermöglicht die Positionsverfolgung und damit prinzipiell die Erfassung von Rotorstellung und -drehzahl von zwei-phasigen Schrittmotoren. Dazu sind zwei Hardwarekanäle und zwei Softwarekanäle bzw. Messkanäle erforderlich. An die beiden Hardwarekanäle werden die Ansteuersignale der beiden Phasenwicklungen eines zwei-phasigen Schrittmotors angelegt. Einem Messkanal ist das Messverfahren 'Step Count (Step. Mot. Phase A)' zuzuweisen und dem anderen Messkanal 'Step Count (Step. Mot. Phase B)'. Im Feld „Hardware Channel“ der Messkanäle wird jeweils auf die Nummer des Hardwarekanals verwiesen, an dem das Ansteuersignal für die jeweilige Phasenwicklung des zwei-phasigen Schrittmotors anliegt. Im Feld „Reference Channel“ eines Messkanals ist auf die Nummer des anderen an der Messung beteiligten Messkanals zu verweisen. Abb. 17-56 veranschaulicht die Vorgehensweise bei der Konfiguration. Die Ansteuersignale der Phasenwicklungen liegen an den Hardwarekanälen 2 und 3 an. Die Messungen werden an den Messkanälen 0 und 1 vorgenommen.

Die Konfigurationsrichtlinien sind in Tab. 17-24 zusammengefasst.



**Abb. 17-56** Konfiguration einer Messung zur Positionsverfolgung an zwei-phasigen Schrittmotoren

An jeder Flanke des zweikanaligen Signals wird ein Positionszähler in- bzw. dekrementiert (Abb. 17-57) und der Wert des Positionszählers im DPRAM der ES4330-Karte aktualisiert. Der Messkanal dem das Messverfahren „Step Count (Step. Mot. Phase A)“ zugeordnet ist, liefert stets einen nicht negativen Positionswert. Im Messwert des Messkanals dem das Messverfahren „Step Count (Step. Mot. Phase B)“ zugeordnet ist, ist im Vorzeichen des Positionswerts die Drehrichtung des Motors kodiert. Einen aufsteigenden Positionszähler erhält man, wenn das Signal an „Phase A“ dem Signal an „Phase B“ vorausliegt (Abb. 17-57).



**Abb. 17-57** Darstellung der Arbeitsweise der Positionsverfolgung an zwei-phasigen Schrittmotoren

**Hinweis**

Die Positionsverfolgung erfolgt modulo  $2^{23}$ . Inkrementieren bei einem Positionszähler von  $(2^{23}-1)$  erzeugt einen Wert des Positionszählers von 0. Dekrementieren bei einem Positionszähler von 0 erzeugt einen Wert des Positionszählers der betragsmäßig gleich  $(2^{23}-1)$  ist.

RTIO-Parameter der Registerkarte „Signals“	Phase A	Phase B
Hardware Channel	Nummer des Hardwarekanals an dem das Ansteuersignal für Phasenwicklung A anliegt.	Nummer des Hardwarekanals an dem das Ansteuersignal für Phasenwicklung B anliegt.
Measurement Mode	Ohne Bedeutung	Ohne Bedeutung
Measurement Procedure	„Step Count (Step. Mot. Phase A)“	„Step Count (Step. Mot. Phase B)“
Reference Channel	Nummer des Messkanals dem das Phase B Signal zugewiesen wurde.	Nummer des Messkanals dem das Phase A Signal zugewiesen wurde.
Messwert	Aktueller Wert des Positionszählers modulo $2^{23}$ .	Aktueller Wert des Positionszählers modulo $2^{23}$ . Im Vorzeichen ist die Drehrichtung des Motors kodiert.

**Tab. 17-24** Konfigurationsrichtlinien und Messwerte der Messverfahren zur Positionsverfolgung an zwei-phasigen Schrittmotoren

#### *Timeout-Überwachung*

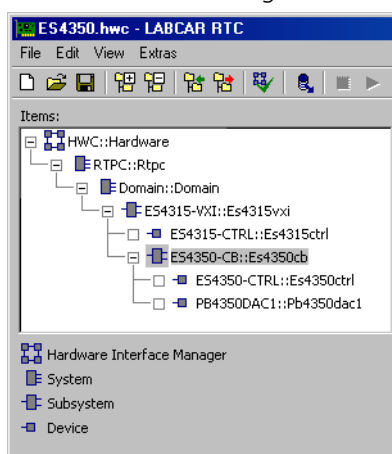
Die Messverfahren zur Positionsverfolgung an zwei-phasigen Schrittmotoren bieten keine Timeout-Überwachung.

## 17.5 ES4350 Carrier Board

In diesem Abschnitt wird die RTIO-Einbindung des ES4350 Carrier Boards beschrieben, das im LABCAR-Umfeld als Trägerkarte für I/O-Module eingesetzt wird.

### 17.5.1 Aufbau des ES4350-RTIO-Baums

Ein ES4350 Carrier Board wird im VXIbus-Teil einer ES4300-Signalbox verbaut. Über den ES4315 VME64x/VXI Adapter der Signalbox wird sie an das im VMEbus-Teil der Signalbox befindliche Experimentaltarget angebunden. Die beschriebene Hardware-Anbindung spiegelt sich im RTIO-Baum wider (Abb. 17-58). Das ES4350-CB Subsystem einer ES4350-Karte wird dem ES4315-VXI Subsystem einer ES4315-Karte zugeordnet.

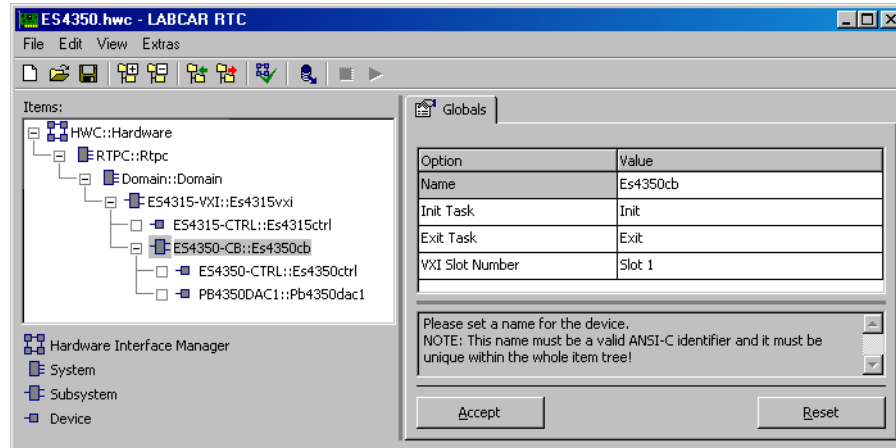


**Abb. 17-58** RTIO-Hardwarebeschreibung mit eingebundenem ES4350 Carrier Board

Die Hardware von ES4350-Trägerkarten besitzt einen Synchronisationbus mit sechs Leitungen sowie zwei identische Busse mit Signalen zur Synchronisation auf den Kurbelwellen-Winkel speziell für den Test von Motorsteuergeräten. Die Konfiguration und Ansteuerung der Busleitungen erfolgt mit dem ES4350-CTRL Device.

17.5.2 Globals (ES4350-CB Subsystem)

Abb. 17-59 zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 17-59** Die Registerkarte „Globals“ des ES4350-CB Subsystems

*VXI Slot Number*

Die Nummer des VXIbus-Slots, in dem das ES4350 Carrier Board eingebaut ist.

Die Nummerierung der sichtbaren VXIbus-Slots innerhalb der ES4300 Signalbox beginnt mit 1 und verläuft von links nach rechts. Der ES4315 VME64x/VXI-Adapter ist fest in den (für den Anwender nicht sichtbaren) VXIbus-Slot 0 eingebaut.

Der Parameter ist nicht online editierbar. In Tab. 17-25 sind die Eigenschaften des Parameters „VXI Slot Number“ zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
VXI Slot Number	uint32	Nein	VXIbus-Slot in dem die ES4350-Karte steckt.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-25** ES4350-CB Subsystem: Konfigurationsparameter der Registerkarte „Globals“

17.5.3 Das ES4350-CTRL Device

Das ES4350-CTRL Device dient der Konfiguration und der Ansteuerung der ES4350-Winkeltakt- und Synchronisationsbusse.

*Funktionsbeschreibung*

In LABCAR-Projekten ist es häufig erforderlich, Abläufe untereinander zu synchronisieren. Projekte zum Test von Motorsteuergeräten zum Beispiel erfordern eine exakte Winkel-Synchronisation zwischen der Generierung von Einspritz- und Zündsignalen auf dem Steuergerät und deren Vermessung auf Test-Hardware (z. B. ES4330-Karten). Motorsteuergerät und Testhardware werden hierzu in LABCAR-Projekten mit demselben Kurbelwellen-Winkeltakt versorgt.

ES4300 basierte LABCAR-Systeme bieten hierfür auf der ES4300-Backplane zwei von einander unabhängige Winkeltakt-Busse mit jeweils 3 Leitungen (siehe Abb. 17-60 auf Seite 581). Eine Busleitung dient zur Synchronisation auf den Null-

durchgang des Kurbelwellenwinkels, die zweite Leitung überträgt den eigentlichen Winkeltakt und auf der dritten Leitung ist die Drehrichtung des Motors codiert.

Ein Winkeltaktbus kann wie jeder Hardwarebus nur von einer Quelle getrieben werden. Mögliche Quellen sind neben den Winkeltakt-Generatoren auf ES4320-Karten auch ES4350-Karten. Zum Entstehungszeitpunkt dieser Dokumentation ist diese Möglichkeit jedoch eingeschränkt, da noch keine Aufsteckmodule für die Winkeltakt-Generierung auf ES4350-Karten zur Verfügung stehen.

Abnehmer für Winkeltakt-Signale sind:

- ES4330-Karten, die damit eine exakte Vermessung drehzahlsynchroner Signale durchführen
- Im Slave-Modus betriebene ES4320-Karten, die damit ihre Signalgeneratoren auf einen externen Winkeltakt synchronisieren
- ES4350-Karten, die den Winkeltakt an die Aufsteckmodule weiterleiten

Eine ES4350-Karte besitzt intern zwei Winkeltakt-Busse. Jeder dieser Busse kann

- von einem der beiden Backplane-Winkeltakt-Busse getrieben werden oder
- vom FPGA der Karte
- oder von einem der sechs Aufsteckmodule mit der Einschränkung, dass es gegenwärtig (wie bereits erwähnt) noch kein geeignetes Modul gibt.

Neben den Winkeltaktbussen besitzen ES4300-Backplane und ES4350-Karte weitere Leitungen, die der Synchronisation von Hardwareeinheiten untereinander dienen. Der aus sechs Leitungen bestehende Synchronisationsbus einer ES4350-Karte ist an das FPGA der Karte und an die sechs Aufsteckmodule geführt. Im Gegensatz zum einem Winkeltakt-Bus dessen Leitungen eine Einheit bilden, sind die Leitungen des Synchronisationsbusses voneinander unabhängig. Ein Aufsteckmodul kann damit z.B. eine Sync-Leitung treiben und gleichzeitig auf Ereignisse auf einer anderen Sync-Leitung reagieren.

Eine Sync-Bus-Leitung kann nur von einer Quelle getrieben werden. Mögliche Quellen sind die Aufsteckmodule sowie die Software (Real-Time Execution Connector), die die Pegel auf den Sync-Leitungen des ES4350 Carrier Board durch Zugriffe auf interne Register der Karte steuern kann.



Jede der sechs Sync-Leitungen einer ES4350-Karte kann auf eine der beiden Sync-Leitungen auf der Backplane einer ES4300 Signalbox geschaltet werden. Auf diese Weise lassen sich Hardware und Aufsteckmodule auf unterschiedlichen ES4350-Karten synchronisieren.

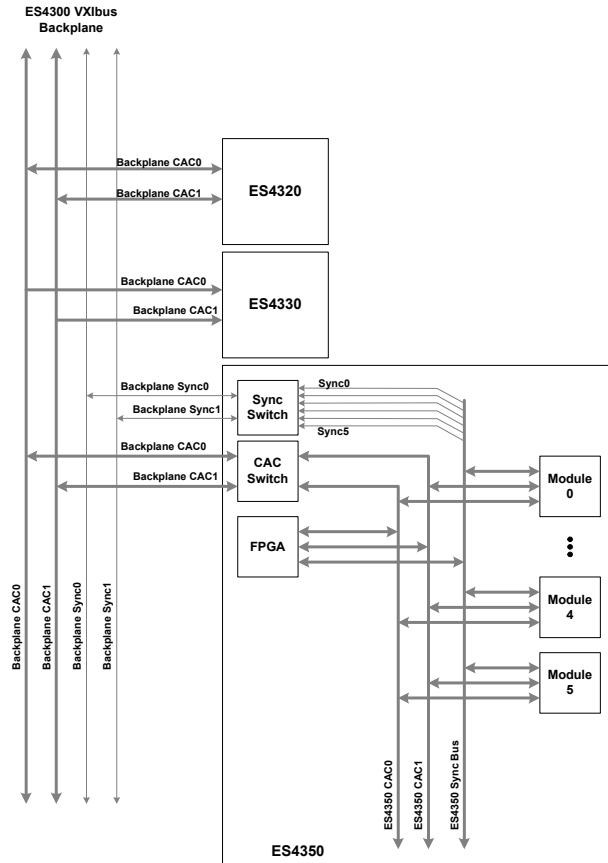
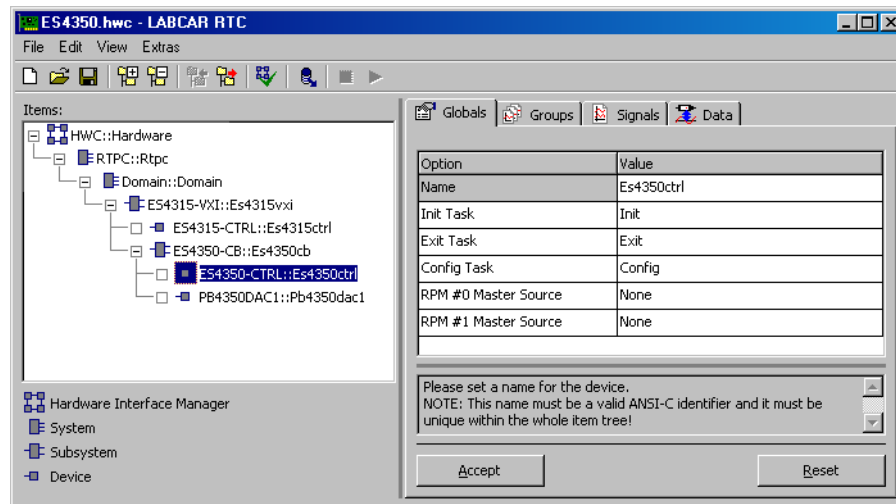


Abb. 17-60 ES4350 Winkeltakt-Bus und Synchronisationsbus

## 17.5.4 Globals (ES4350-CTRL Device)

Die Registerkarte „Globals“ eines ES4350-CTRL Device ist in Abb. 17-61 dargestellt.



**Abb. 17-61** Die Registerkarte „Globals“ des ES4350-CTRL Device

#### *RPM #0 Master Source, RPM #1 Master Source*

In diesem Listenfeld wird die Quelle festgelegt, die den internen Winkelaktbus 0 bzw. 1 eines ES4350 Carrier Board treibt. Das Listenfeld wird dynamisch aufgebaut. Feste Optionen sind:

- **None**  
Der Winkelaktbus wird nicht getrieben.
- **VXI Backplane RPM #0**  
Der Winkelaktbus wird vom Winkelaktbus 0 der ES4300 Backplane getrieben.
- **VXI Backplane RPM #1**  
Der Winkelaktbus wird vom Winkelaktbus 1 der ES4300 Backplane getrieben.

Falls ein Aufsteckmodul eingebunden wird, das die Fähigkeit besitzt, Winkelakt-signale zu erzeugen, so wird dieses Aufsteckmodul in der Liste ebenfalls zur Auswahl angeboten.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

In Tab. 17-26 sind die Eigenschaften der einzelnen RTIO-Parameter der Registerkarte „Globals“ zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
RPM #0 Master Source	uint8	Ja	Quelle für interne Winkeltakt-Leitung 0 und 1: 0: "None" 1: "VXI Backplane RPM #0" 2: "VXI Backplane RPM #1" 3: reserviert 4: Aufsteckmodul in Slot 0 5: Aufsteckmodul in Slot 1 6: Aufsteckmodul in Slot 2 7: Aufsteckmodul in Slot 3 8: Aufsteckmodul in Slot 4 9: Aufsteckmodul in Slot 5
RPM #1 Master Source			

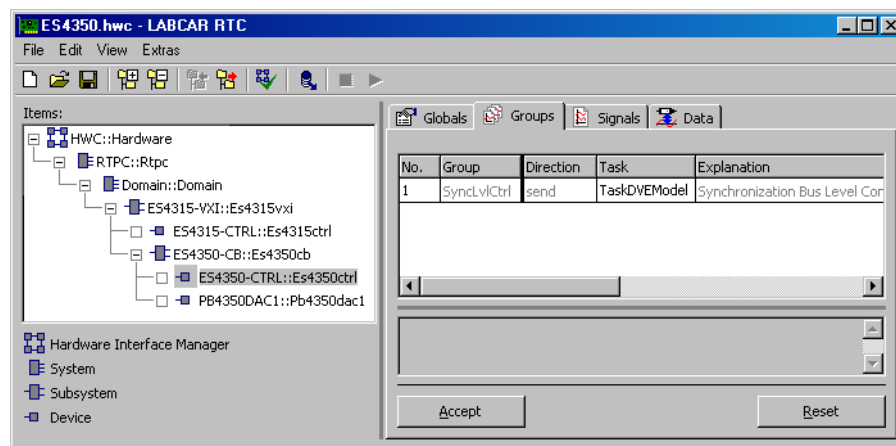
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-26** ES4350-CTRL Device: Konfigurationsparameter der Registerkarte „Globals“

17.5.5 Groups (ES4350-CTRL Device)

Das ES4350-CTRL Device besitzt eine Signalgruppe, die vom Experimentaltarget zum ES4350 Carrier Board übertragen wird (siehe Abb. 17-62). Sie transportiert Pegelinformationen für die Sync-Leitungen einer ES4350, die von RTIO aus angetrieben bzw. gesteuert werden.

Der Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erzeugenden Synchronisationssignale. Sinnvolle Aktivierungsperioden sind durch Laufzeiten der Firmware auf den Trägerkarten auf 1 ms nach unten begrenzt.



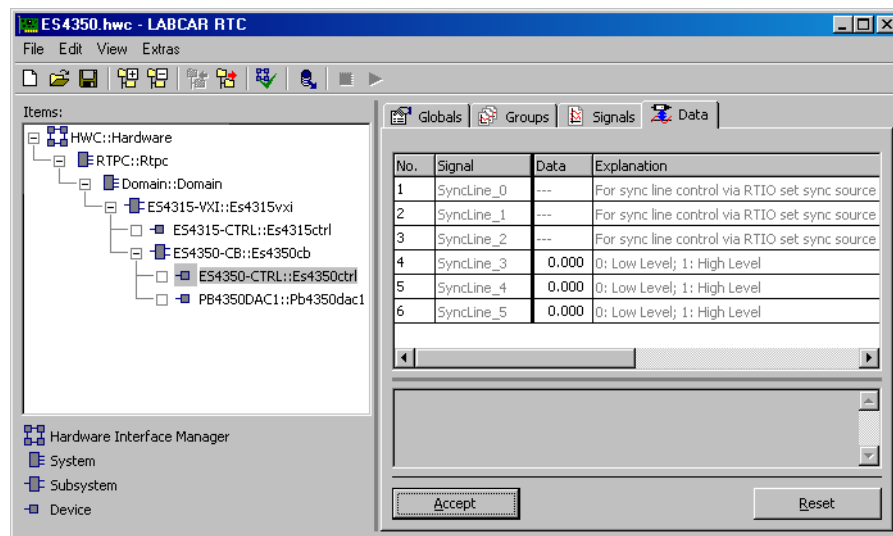
**Abb. 17-62** Die Registerkarte „Groups“ des ES4350-CTRL Device

### 17.5.6 RTIO-Signale der Signalgruppe „SynclvlCtrl“

Die Signalgruppe „SynclvlCtrl“ umfasst sechs RTIO-Signale „Syncline\_0“ bis „Syncline\_5“, die vom Datentyp „bool“ sind.

Wird Sync-Leitung x (x = 0, 1, ... 5) dahingehend konfiguriert, dass sie von RTIO aus getrieben bzw. gesteuert wird, so definiert der Wert des RTIO-Signals „Syncline\_x“ den Pegel der Leitung. „0“ bedeutet dabei einen Low-Pegel, „1“ bedeutet einen High-Pegel.

Wird Sync-Leitung x nicht von RTIO gesteuert, so hat das Signal keine Bedeutung. Dies ist auch schon optisch in der Registerkarte „Data“ des RTIO-Elements sichtbar: Die „Data“-Spalte der Sync-Leitungen, die nicht von RTIO gesteuert werden, sind inaktiv. In Abb. 17-63 z. B. werden die Sync-Leitungen 0 bis 2 nicht von RTIO gesteuert, die Pegel der Sync-Leitung 3 bis 5 hingegen werden von RTIO gesteuert.



**Abb. 17-63** Die Registerkarte „Data“ des ES4350-CTRL Devices

Tab. 17-27 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Syncline_x (x = 0, 1, ... 5)	bool	Pegelsteuerung auf Sync-Leitung x 0: Low-Pegel 1: High-Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-27** ES4350-CTRL Device: Die RTIO-Signale der Signalgruppe „SynclvlCtrl“

## 17.5.7 Signals (ES4350-CTRL Device)

In der Registerkarte „Signals“ wird die Konfiguration der Sync-Leitungen eines ES4350 Carrier Board durchgeführt.

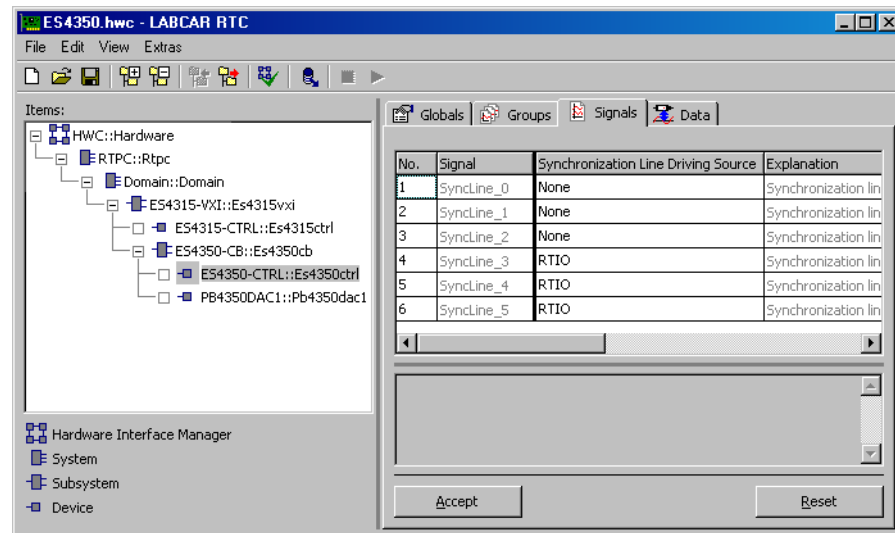
Abb. 17-64 zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

#### *Synchronization Line Driving Source*

In diesem Listenfeld wird die Quelle festgelegt, die die Sync-Leitung treibt. Das Listenfeld wird dynamisch aufgebaut. Feste Optionen sind:

- **None**  
Die Sync-Leitung wird nicht getrieben.
- **VXI Backplane SYNC #0**  
Die Sync-Leitung wird von Sync-Leitung 0 der ES4300 Backplane getrieben.
- **VXI Backplane SYNC #1**  
Die Sync-Leitung wird vom Sync-Leitung 1 der ES4300 Backplane getrieben.
- **RTIO**  
Der Pegel auf der Sync-Leitung wird von RTIO gesteuert.

Falls ein I/O-Modul eingebunden wird, das die Fähigkeit besitzt, Sync-Leitungen zu treiben, so wird dieses Modul in der Liste ebenfalls zur Auswahl angeboten.



**Abb. 17-64** Die Registerkarte „Signals“ des ES4350-CTRL Devices

Tab. 17-28 fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Synchronization Line Driving Source	uint8	Ja	Quelle für Sync-Leitung 0: None 1: VXI Backplane SYNC #0 2: VXI Backplane SYNC #1 3: RTIO 4: Aufsteckmodul in Slot 0 5: Aufsteckmodul in Slot 1 6: Aufsteckmodul in Slot 2 7: Aufsteckmodul in Slot 3 8: Aufsteckmodul in Slot 4 9: Aufsteckmodul in Slot 5

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-28** ES4350-CTRL Device: Konfigurationsparameter der Registerkarte „Signals“

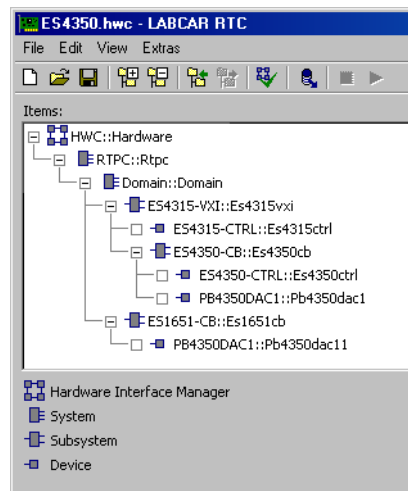
## 17.6 PB4350DAC1 I/O-Modul

In diesem Abschnitt wird die RTIO-Einbindung des PB4350DAC1-Moduls beschrieben.

Das PB4350DAC1-Modul ist ganz allgemein für Aufgaben geeignet, die die Ausgabe von unipolaren, analogen Spannungen erfordern. Typischer Anwendungsfall des Moduls in LABCAR-Projekten ist die Nachbildung von Fahrzeugsensoren.

### 17.6.1 Aufbau des PB4350DAC1-RTIO-Baums

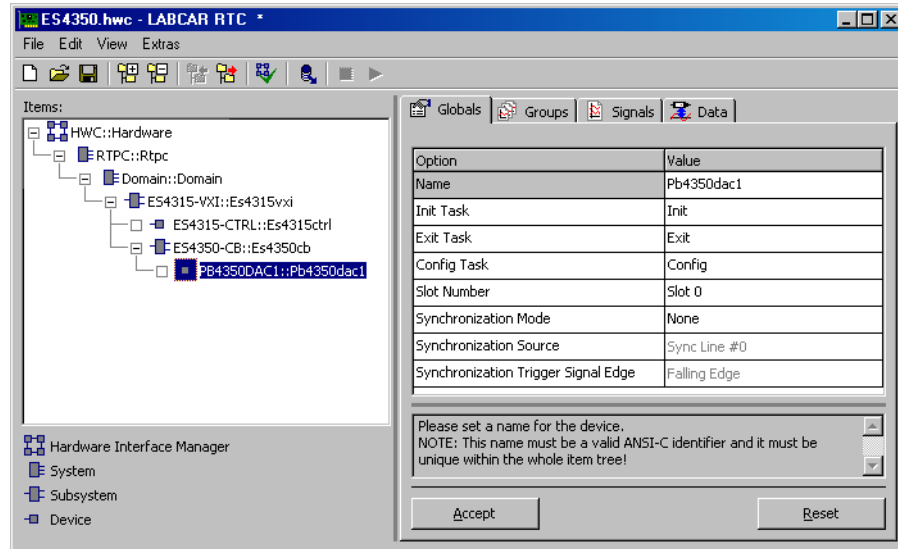
Im RTIO-Editor wird das PB4350DAC1-Modul durch Auswahl des PB4350DAC1 Device eingebunden. Dieses Element kann entweder einer ES1651 Carrier Board oder eines ES4350 Carrier Board zugeordnet werden. Abb. 17-65 zeigt die Anbindung von PB4350DAC1-Modulen an beide Typen von Trägerkarten.



**Abb. 17-65** RTIO-Hardwarebeschreibung mit eingebundenen PB4350DAC1-Modulen

17.6.2 Globals (PB4350DAC1 Device)

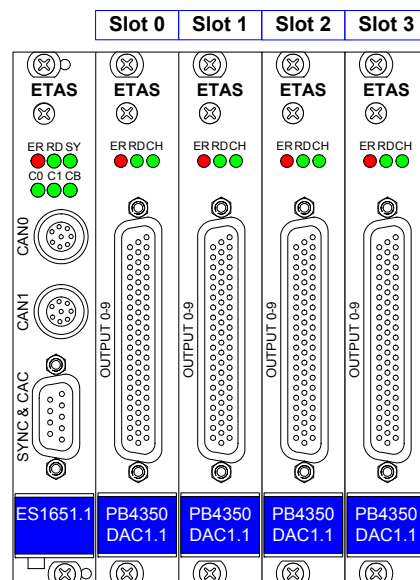
Abb. 17-66 zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 17-66** Die Registerkarte „Globals“ des PB4350DAC1 Device  
*Slot Number*

In diesem Listenfeld ist die Nummer des Steckplatzes des ES1651 Carrier Board bzw. des ES4350 Carrier Board einzustellen, in dem das PB4350DAC1-Modul steckt. ES4350 Carrier Boards weisen sechs Modul-Steckplätze auf, deren Nummern auf der Frontplatte aufgedruckt sind.

Die Nummerierung der vier Steckplätze eines ES1651 Carrier Boards ist in Abb. 17-67 gezeigt.



**Abb. 17-67** Nummerierung der Modul-Steckplätze einer ES1651-Trägerkarte



### *Synchronization Mode*

---

In diesem Listenfeld wird festgelegt, wann die von der PC-Software (Real-Time Execution Connector) bzw. vom RTIO-Treiber an das PB4350DAC1-Modul übermittelten analogen Ausgabewerte von der Hardware des PB4350DAC1-Moduls übernommen werden.

Wird in diesem Listenfeld die Option „None“ eingestellt, so werden die von der PC-Software übermittelten analogen Ausgabewerte sofort an den Ausgängen des PB4350DAC1-Moduls aktualisiert.

Bei der Option „Hardware Triggered“ erfolgt eine Aktualisierung erst, wenn auf der im Listenfeld „Synchronization Source“ definierten Leitung des Synchronisationsbusses eines ES1651 Carrier Boards oder eines ES4350 Carrier Boards eine aktive Flanke erfasst wurde. Auf diese Weise ist eine Synchronisierung der analogen Ausgabe auf ein externes Ereignis möglich.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

### *Synchronization Source*

---

Hier ist die Leitung des Synchronisationsbusses anzugeben, deren aktive Flanken eine Aktualisierung der Spannungen an den Kanälen eines PB4350DAC1-Moduls anstoßen.

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „Synchronization Mode“ (s.o.) die Option „Hardware Triggered“ (Synchronisierung der analogen Ausgabekanäle auf ein externes Ereignis) eingestellt ist.

Der Synchronisationsbus eines ES1651 Carrier Boards umfasst vier Leitungen, der eines ES4350 Carrier Boards umfasst sechs Leitungen.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

### *Synchronization Trigger Signal Edge*

---

In diesem Listenfeld wird die aktive Flanke („Rising Edge“ = steigende Flanke oder „Falling Edge“ = fallende Flanke) einzustellen, die eine Aktualisierung der Spannungen an den Kanälen eines PB4350DAC1-Moduls anstößt.

Ebenso wie das Listenfeld „Synchronization Source“ ist dieses Listenfeld nur editierbar, falls im Listenfeld „Synchronization Mode“ die Option „Hardware Triggered“ eingestellt ist.

Das Listenfeld ist online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Tab. 17-29 fasst die Eigenschaften der Konfigurations-Parameter der Registerkarte „Globals“ zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Slot Number	uint32	Nein	Steckplatz-Nummer des Moduls auf der Trägerkarte. ES1651: 0, 1, 2, oder 3 ES4350: 0, 1, ... 5
Synchronization Mode	uint8	Ja	Synchronisations-Modus 1: Keine Synchronisierung 0: Hardware Triggered
Synchronization Source	uint8	Ja	Synchronisations-Leitung 0: Sync-Leitung 0 1: Sync-Leitung 1 2: Sync-Leitung 2 3: Sync-Leitung 3 4: Sync-Leitung 4 5: Sync-Leitung 5 (4 und 5 nur bei ES4350-Trägerkarten!)
Synchronization Trigger Signal Edge	bool	Ja	Synchronisations-Trigger-Flanke 0: Fallende Flanke 1: Steigende Flanke

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-29** PB4350DAC1 Device: Konfigurationsparameter der Registerkarte „Globals“

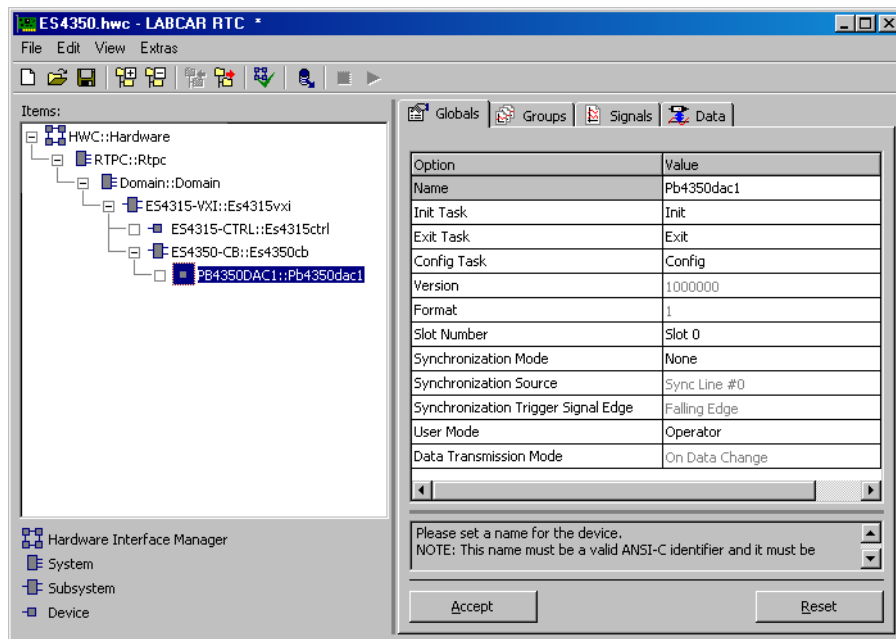
#### Versteckte Optionsfelder

Wird die Registerkarte „Globals“ mit der rechten Maustaste angeklickt und anschließend im Kontextmenü die Option „Show all Options“ gewählt, so werden weitere Optionsfelder sichtbar (siehe Abb. 17-68).

Diese Optionsfelder erlauben es dem ETAS-Servicepersonal, bei einer Fehlersuche zusätzliche Informationen über das PB4350DAC1-Modul zu erhalten.

#### Hinweis

*Diese Optionsfelder sind nicht für den Anwender gedacht - die Voreinstellungen sollten deshalb vom Anwender nicht geändert werden!*



**Abb. 17-68** Die Registerkarte „Globals“ des PB4350DAC1 Device (mit allen Optionsfeldern)

#### *User Mode*

Stellt den RTIO-Anwendermodus ein.

Anwender sollten in diesem Listenfeld die Option „Operator“ wählen. Der Anwendermodus „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche und ist nur für ETAS-Servicepersonal gedacht. Bei aktiviertem „Supervisor“-Modus werden unter Anderem die Inhalte des nicht flüchtigen Datenspeichers des PB4350DAC1-Moduls während der Initialisierung in das „Target Debug“-Fenster ausgegeben.

Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „Operator“.

#### *Data Transmission Mode*

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „User Mode“ die Option „Supervisor“ eingestellt ist. Es legt fest, wann Echtzeit-Daten vom Experimental-target zum PB4350DAC1-Modul übertragen werden.

Ist die Option „On Data Change“ aktiviert, werden die Daten einer Signalgruppe nur dann übertragen, wenn sich wenigstens ein Datum geändert hat. Ist die Option „Every Interval“ aktiviert, so werden die Echtzeit-Daten in jedem Rechenschritt der zugeordneten Echtzeit-Task übertragen. Das Listenfeld ist zur Laufzeit editierbar und hat die Voreinstellung „On Data Change“.

In Tab. 17-30 sind die Eigenschaften der versteckten RTIO-Parameter der Registerkarte "Globals" zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
User Mode	uint8	Ja	Anwender-Modus 0: Operator 1: Supervisor „Operator“ ist Voreinstellung. „Supervisor“ aktiviert zusätzliche Optionen zur Fehlersuche.
Data Transmission Mode	uint8	Ja	Übertragungsmodus der Echtzeit-Daten 0: On Data Change 1: Every Interval „On Data Change“ ist Voreinstellung. Optionsfeld ist nur editierbar, wenn „User Mode“ = „Supervisor“

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-30** PB4350DAC1 Device: Versteckte Konfigurationsparameter der Registerkarte „Globals“

### 17.6.3 Groups (PB4350DAC Device)

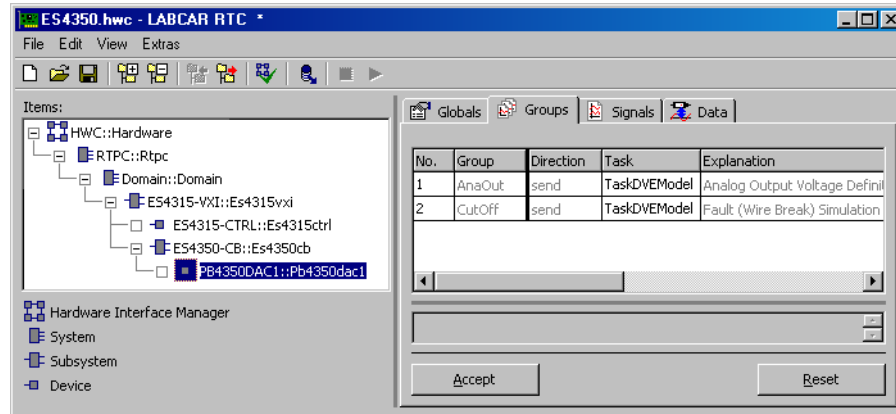
Das PB4350DAC1 Device verfügt über zwei Signalgruppen (siehe Abb. 17-69), die beide vom Experimentaltarget zum PB4350DAC1-Modul übertragen werden.

Die Signalgruppe „AnaOut“ transportiert die Spannungswerte für die zehn Analogausgänge eines Moduls.

Zur Fehlersimulation (z. B. zur Simulation eines Leitungsbruchs in der Zuleitung eines Fahrzeugsensors zum Steuergerät) bietet der PB4350DAC1-RTIO-Treiber die Signalgruppe „CutOff“. Mit dieser Signalgruppe lassen sich die analogen Ausgänge eines Moduls zwischen dem Normalbetrieb der analogen Spannungsangabe und dem Fehlerfall eines hochohmigen Ausgangs schalten.

Den Signalgruppen sind Tasks des Echtzeit-Betriebssystems zuzuordnen. Für die Signalgruppe „AnaOut“ zur analogen Signalausgabe wird man üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der auszugebenden Signale. Sinnvolle Aktivierungsperioden sind durch Laufzeiten der Firmware auf den Trägerkarten sowie dem PB4350DAC1-Modul auf 500 µs nach unten begrenzt. Für die Signalgruppe

„CutOff“ zur Fehlersimulation wird man üblicherweise Tasks mit Software-Aktivierung oder periodische Tasks mit langsamer Aktivierung im Bereich größer 10 ms wählen.



**Abb. 17-69** Die Registerkarte „Groups des PB4350DAC1 Device

17.6.4 RTIO-Signale der Signalgruppe „AnaOut“

Die Signalgruppe „AnaOut“ umfasst zehn RTIO-Signale („AnaOut\_0“ bis „AnaOut\_9“), die vom Datentyp „real32“ sind. Sie definieren die Spannungen an den analogen Ausgängen des PB4350DAC1-Moduls.

Der Wertebereich der RTIO-Signale geht 0.0 V bis 10.0 V, falls der zugeordnete DA-Ausgang mit interner Spannungsreferenz betrieben wird, bzw. von 0.0 bis 1.0, falls der zugeordnete DA-Ausgang mit externer Spannungsreferenz betrieben wird.

Tab. 17-31 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Bemerkung / Wertebereich
AnaOut_0 bis AnaOut_9	real32	Analoger Ausgang 0 bis 9: Interne Spannungsreferenz: Ausgangsspannung in Volt (0.0 V bis 10.0 V) Externe Spannungsreferenz: Ausgangsspannung in Prozent der externen Referenzspannung Wertebereich: 0.0 (entspricht 0 %) bis 1.0 (entspricht 100 %)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-31** PB4350DAC1 Device: Die RTIO-Signale der Signalgruppe „AnaOut“

### 17.6.5 RTIO-Signale der Signalgruppe „CutOff“

Die Signalgruppe „CutOff“ umfasst zehn RTIO-Signale („CutOff\_0“ bis „CutOff\_9“), die vom Datentyp „bool“ sind. Sie definieren, ob der zugeordnete DA-Ausgang im Modus zur analogen Spannungsausgabe oder im Modus zur Fehlersimulation betrieben wird.

#### **Hinweis**

*Um sicherzustellen, dass die Schaltbefehle umgesetzt werden, sollten die Relais zur Unterbrechung der Ausgangssignale maximal alle 50 ms angesteuert werden.*

Tab. 17-32 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Bemerkung / Wertebereich
CutOff_0 bis CutOff_9	bool	Betriebsmodus analoger Ausgang 0 bis 9: 0: Analoge Spannungsausgabe (ist die Voreinstellung) 1: Fehlersimulation

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 17-32** PB4350DAC1 Device: Die RTIO-Signale der Signalgruppe „CutOff“

### 17.6.6 Signals (PB4350DAC Device)

In der Registerkarte „Signals“ wird die Konfiguration der DA-Ausgänge eines PB4350DAC1-Moduls durchgeführt.

Abb. 17-70 zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Reference

In diesem Listenfeld wird die Spannungsreferenz (interne 10 Volt-Spannungsreferenz oder externe Spannungsreferenz) eines DA-Ausgangs angegeben.

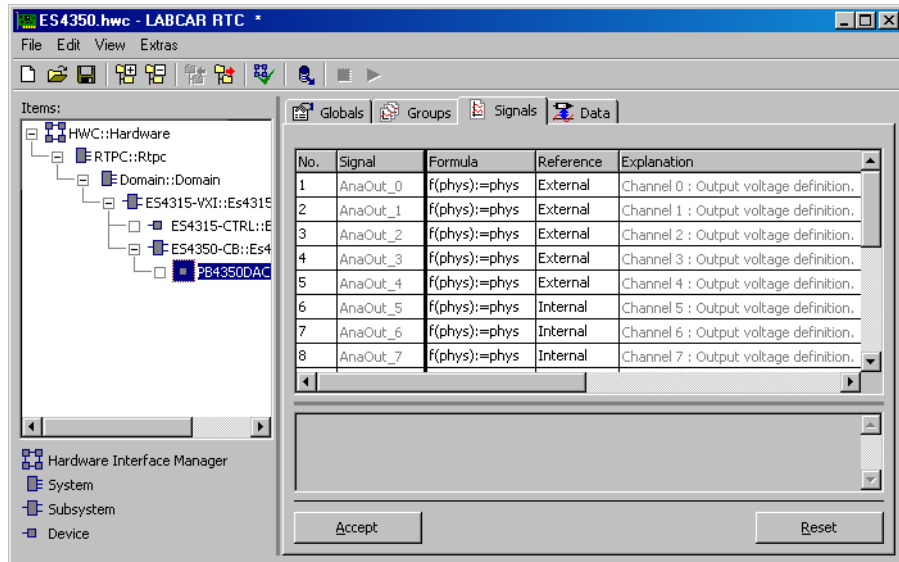


Abb. 17-70 Die Registerkarte „Signals“ des PB4350DAC1 Device

In Tab. 17-33 sind die Eigenschaften der einzelnen Parameter zusammengefasst.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung / Wertebereich
Reference	bool	Ja	Spannungsreferenz des DA-Ausgangs 0: Externe Referenzspannung 1: Interne 10 V-Referenzspannung (Voreinstellung)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

Tab. 17-33 PB4350DAC1 Device: Konfigurationsparameter der Registerkarte „Signals“





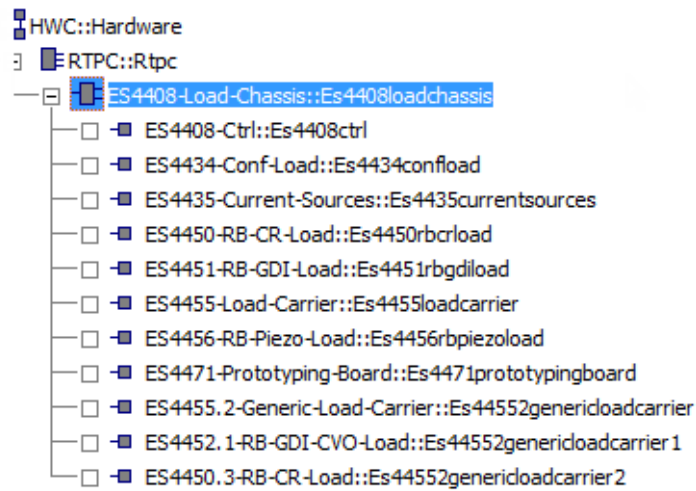
## 18 ES4408-System

Das ES4408.1 Load Chassis und die dazugehörigen Boards bilden ein System zur Simulation elektrischer Lasten. Die Einschubkarten ermöglichen die Nachbildung verschiedenster Lasten, um das Verhalten von Steuergeräten in einem HIL-System zu testen. Hierzu zählen insbesondere Pull-Up/Pull-Down-Lasten (bis 150 mA) und die Simulation von elektromagnetischen und Piezo-Injektoren.

Zur Kommunikation mit den Boards im ES4408.1 Load Chassis via Ethernet-Schnittstelle wird das ES4408CON.1 Communication Interface benötigt.

### Aufbau des ES4408-RTIO-Baums

Im RTIO-Editor wird das ES4408.1 Load Chassis durch Auswahl des ES4408-Load-Chassis Subsystems eingebunden. Dieses Subsystem wird einem RTPC-Experimentaltarget zugeordnet.



**Abb. 18-1** RTIO-Hardwarebeschreibung mit eingebundenem ES4408.1 Load Chassis und weiteren Boards in diesem

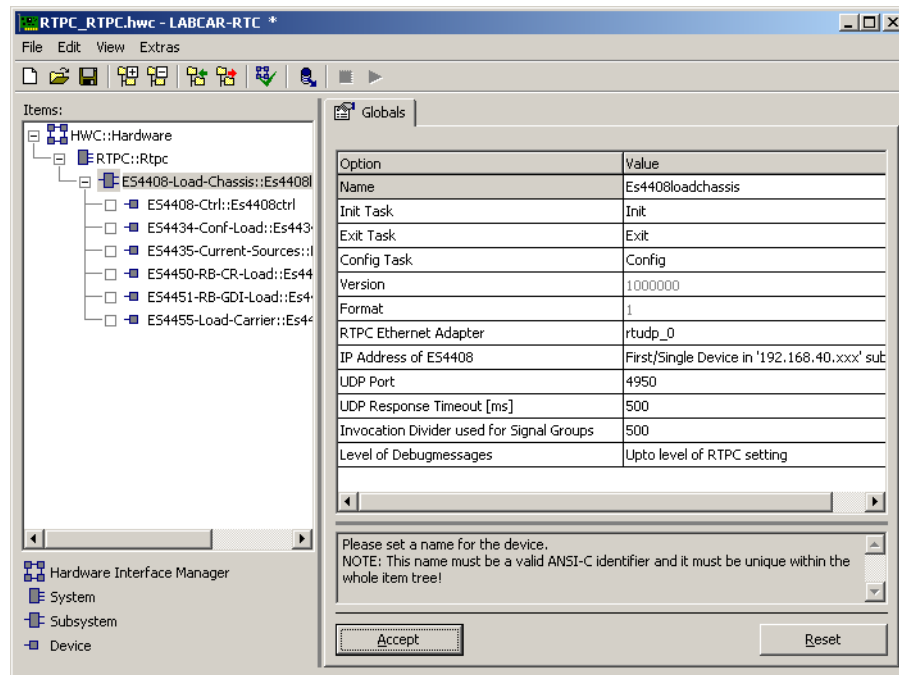
Darunter folgen unmittelbar die Instanzen der eingebauten Karten als RTIO-Devices.

## 18.1 ES4408-Load-Chassis Subsystem

### 18.1.1 Globals (ES4408-Load-Chassis Subsystem)

In der Registerkarte „Globals“ werden alle Einstellungen vorgenommen, die sich auf das ES4408.1 Load Chassis und die Karten darin beziehen.

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4408-Load-Chassis Subsystems nach Auswahl von **View → Show All**.



**Abb. 18-2** Die Registerkarte „Globals“ des ES4408-Load-Chassis Subsystems.

In diesem Register können die folgenden Parameter eingestellt werden.

#### **Hinweis**

Die mit \* gekennzeichneten Optionen sind nur nach Auswahl von **View → Show All** sichtbar.

#### **Name**

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!

#### **Init Task, Exit Task, Config Task**

In diesen Optionsfeldern können Sie die Tasks auswählen, in denen die generierten Init-, Exit- und Config-Prozesse eingehängt werden.

### *RTPC Ethernet Adapter*

---

Hier wird der logische Ethernet-Adapter des Real-Time PCs ausgewählt, der für die Kommunikation mit dem ES4408.1 Load Chassis via rtudp (Real-Time UDP) verwendet wird. Für ein ES4408.1 Load Chassis ist auch ein Anschluss des Typs „nonrtudp“ möglich.

### *IP Address of ES4408*

---

Hier weisen Sie dem ES4408.1 Load Chassis eine IP-Adresse im Bereich zwischen 192.168.40.50 und 192.168.40.70 zu.

Darüber hinaus kann auch einfach das erste oder zweite im Netzwerk gefundene Gerät spezifiziert werden (wird mit einem Broadcast im Subnetz ermittelt).

### **Hinweis**

Die mit \* gekennzeichnete Optionen werden nur nach Auswahl von [View](#) → [Show All](#) angezeigt.

### *UDP Port\**

---

Der für die Kommunikation zu verwendende UDP-Port des Ethernet-Adapters (Default: 4950).

### *UDP Response Timeout [ms]\**

---

Timeout für ein UDP-Antwortpaket vom ES4408.1 Load Chassis (Default: 500).

### *Invocation Divider used for Signal Groups\**

---

Die Einstellung „Invocation Divider...“ ermöglicht es, die Zahl der Anfragen beim ES4408.1 Load Chassis in sinnvollen Grenzen zu halten.

Die Prozesse einer Signalgruppe sind normalerweise in eine einzige, schnelle Simulations-Task eingehängt, die Anfragen bei der ES4408.1 beziehen sich dagegen typischerweise auf Statusinformationen und müssen nicht unter Echtzeitbedingungen verhandelt werden.

Eine vernünftige Zeitspanne bei der Abfrage solcher Informationen beträgt 1 - 5 Sekunden. UDP-Anfragen in Signalgruppen sind als „non-blocking“ implementiert und werden somit auch über mehrfache Taskaufrufe hinweg verarbeitet.

### *Level of Debug Messages\**

---

Die Option „Level of Debug Messages“ ermöglicht eine Einstellung für den Level der Fehlermeldungen, die unabhängig sind von dem des Real-Time PCs oder dem Rest des Systems.

Außerdem kann „SCPI Debugging“ aktiviert werden – in dieser Betriebsart gibt jeder SCPI-Befehl eine Antwort zurück, unabhängig davon, ob ein Fehler aufgetreten ist oder nicht.

In der folgenden Tabelle werden die möglichen Einstellungen erläutert:

<b>Einstellung</b>	<b>Bedeutung</b>
Upto Level of RTPC setting	Der in LABCAR-RTPC eingestellte Level wird angewandt.
Upto LOG_ERR	Nur Emergencies, Criticals, Alerts und Errors werden geloggt
Upto LOG_WARNING	Wie LOG_ERROR plus Warnings
Upto LOG_NOTICE	Wie LOG_WARNING plus Notices
Upto LOG_INFO	Wie LOG_NOTICE plus Information
Upto LOG_DEBUG	Alle verfügbaren Informationen werden aufgezeichnet
...+ SCPI DEB ON	Kann zusätzlich zu jeder der obigen Optionen gewählt werden: Aktiviert „SCPI Debugging“

**Tab. 18-1** Mögliche Einstellungen für „Level of Debug Messages“

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

#### **Hinweis**

Die mit \* gekennzeichneten Optionen sind nur nach Auswahl von **View** → **Show All** sichtbar.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	String	nein	Name des Subsystems
Init Task, Exit Task, Config Task	Auswahl	nein	Tasks für die Init-, Exit- und Config-Prozesse
RTPC Ethernet Adapter	int	nein	Der Ethernet-Adapter des Real-Time PCs für die Kommunikation mit dem ES4408.1 Load Chassis. Wertebereich: rtudp_0...rtudp_3, nonrtudp
IP Address of ES4408	int	nein	Dem ES4408.1 Load Chassis zugewiesene IP-Adresse Wertebereich: 'First/Single Device'   'Second Device'  192.168.40.x , x = 50...70
UDP Port*	int	nein	Der UDP-Port des Ethernet-Adapters (Default: 4950) Wertebereich: 1000...65535
UDP Response Timeout [ms]*	int	nein	Timeout für die Kommunikation mit dem ES4408.1 Load Chassis (Default: 500) Wertebereich: 10...10000 ms
Invocation Divider used for Signal Groups*	int	nein	Kommunikation einer Signalgruppe findet nur bei jedem n-ten Aufruf der Task statt (Default: 500) Wertebereich: 10...10000
Level of Debugmessages*	int	ja	Einstellungen für den Umfang von Debug-Meldungen (siehe Tab. 18-1 auf Seite 600)

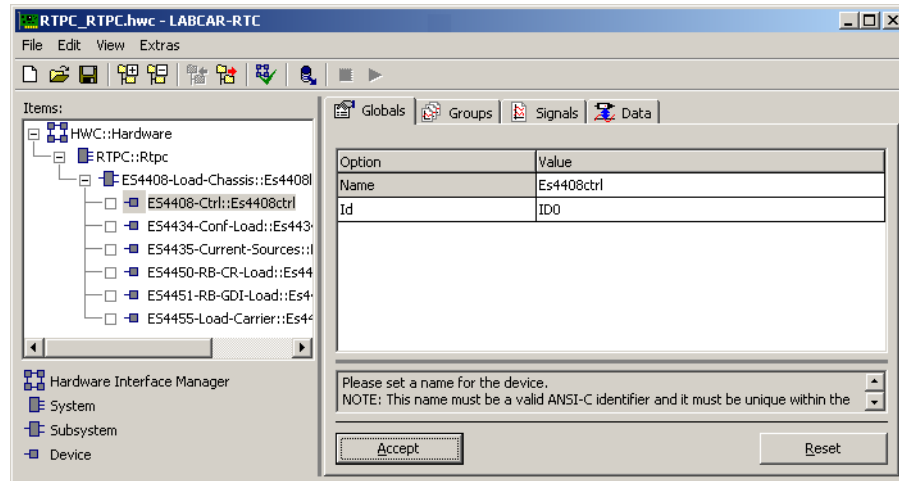
**Tab. 18-2** ES4408-Load-Chassis Subsystem: Konfigurationsparameter der Registerkarte „Globals“

## 18.2 ES4408-Ctrl Device

Das ES4408-Ctrl Device dient zur Einstellung von global gültigen Parametern des ES4408CON.1 Communication Interface.

### 18.2.1 Globals (ES4408-Ctrl Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4408-Ctrl Device.



**Abb. 18-3** Die Registerkarte „Globals“ des ES4408-Ctrl Device

Dieses Register enthält folgende Parameter:

Name

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

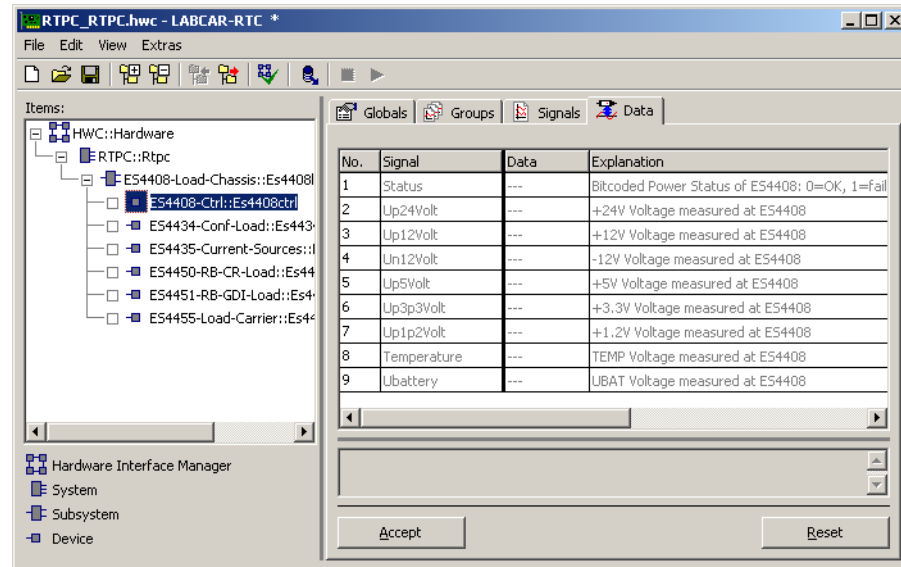
Id

Identifizier des ES4408CON.1 Communication Interface

## 18.2.2 Data (ES4408-Ctrl Device)

In der Registerkarte „Data“ des ES4408-Ctrl Device werden Statusinformationen zu den Versorgungsspannungen, zu Temperatur und Batteriespannung ausgegeben.

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES4408-Ctrl Device.



**Abb. 18-4** Die Registerkarte „Data“ des ES4408-Ctrl Device

Folgende Signaldaten werden angezeigt.

### *Status*

Der allgemeine Status der Spannungsversorgung des ES4408.1 Load Chassis.

Ein gesetztes Bit 0, 1, 2, 3 oder 4 bedeutet jeweils das Versagen der Spannungen +12 V, -12 V, +5 V, +3,3 V oder +1,2 V

### *Up24V, Up12V, Un12V, Up5V, Up3p3V, Up1p2V*

Hier werden die im laufenden Betrieb gemessenen Werte der Versorgungsspannungen dargestellt.

### *Temperature*

Vom Temperatursensor im ES4408.1 Load Chassis gemessener Wert.

### *Ubattery*

Aktueller Wert der in das ES4408.1 Load Chassis eingespeisten Batteriespannung.

Die folgende Tabelle fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp	Bemerkung
Status	uint8	Wenn Bit 0, 1, 2, 3 oder 4 sind gesetzt ist, bedeutet dies einen Fehler bei den entsprechenden Versorgungsspannungen +12 V, -12 V, +5 V, +3,3 V oder +1,2 V
Up24V	real32	Messwert für +24 V-Versorgungsspannung
Up12V	real32	Messwert für +12 V-Versorgungsspannung
Un12V	real32	Messwert für -12 V-Versorgungsspannung
Up5V	real32	Messwert für +5 V-Versorgungsspannung
Up3p3V	real32	Messwert für +3,3 V-Versorgungsspannung
Up1p2V	real32	Messwert für +1,2 V-Versorgungsspannung
Temperature	real32	Messwert für Spannung am Temperatursensor
Ubattery	real32	Messwert für Batteriespannung
Module Present	uint8	= 0: Das diesem RTIO-Item zugeordnete Board ist nicht vorhanden = 1: Das diesem RTIO-Item zugeordnete Board ist vorhanden und aktiv

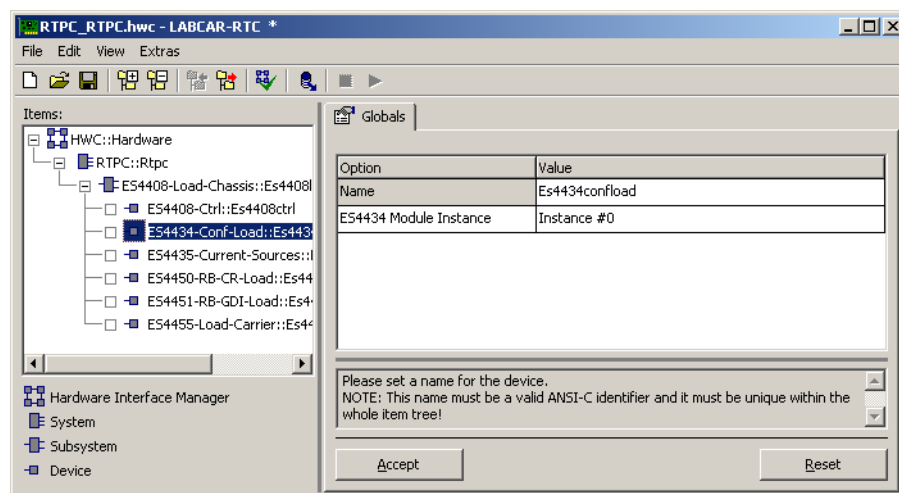
**Abb. 18-5** ES4408-Ctrl Device: Signale der Registerkarte „Data“

### 18.3 ES4434-Conf-Load Device

Die Konfiguration des ES4434.1 Configurable Load Board findet auf dem Board statt, so daß softwareseitig keine Einstellungen erforderlich sind.

#### 18.3.1 Globals (ES4434-Conf-Load Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4434-Conf-Load Device.



**Abb. 18-6** Die Registerkarte „Globals“ des ES4434-Conf-Load Device  
Dieses Register enthält folgende Parameter:



Name

Ein eindeutiger Bezeichner für dieses Device.

**Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

ES4434 Module Instance

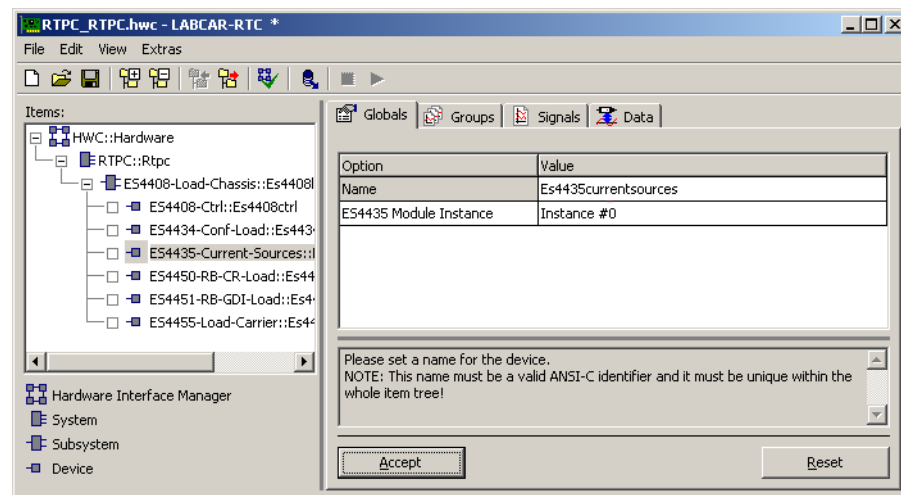
Die internen Bezeichnungen dieser Instanz eines ES4434.1 Configurable Load Board (Instance #0 ... Instance #6).

## 18.4 ES4435-Current-Sources Device

Im Gegensatz zum ES4434.1 Configurable Load Board wird das ES4435.1 Current Source Load Board fast ausschließlich von LABCAR-RTC konfiguriert und gesteuert.

## 18.4.1 Globals (ES4435-Current-Sources Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4435-Current-Sources Device.



**Abb. 18-7** Die Registerkarte „Globals“ des ES4435-Current-Sources Device

In diesem Register können folgende Parameter eingestellt werden:

Name

Ein eindeutiger Bezeichner für dieses Device.

**Hinweis**

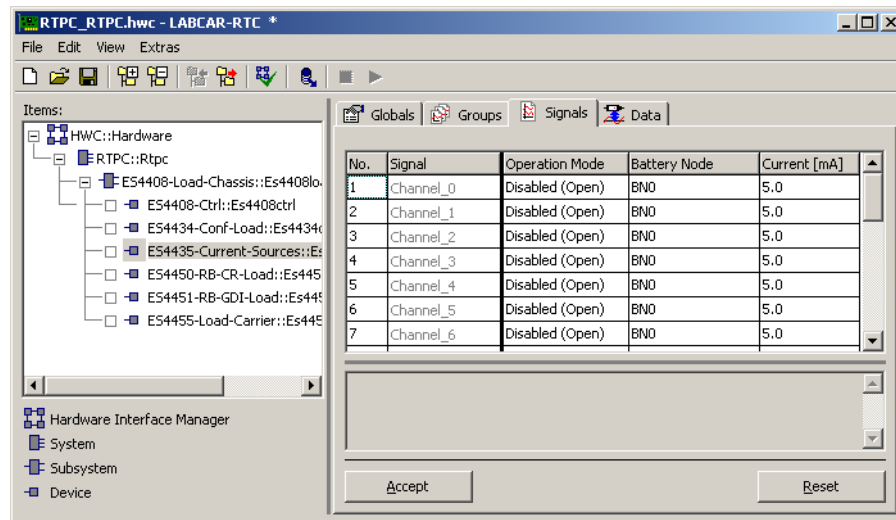
*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

ES4435 Module Instance

Die internen Bezeichnungen dieser Instanz eines ES4435.1 Current Source Load Board (Instance #0 ... Instance #6).

## 18.4.2 Signals (ES4435-Current-Sources Device)

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES4435-Current-Sources Device.



**Abb. 18-8** Die Registerkarte „Signals“ des ES4435-Current-Sources Device  
In diesem Register können für jeden der 24 Kanäle folgende Parameter eingestellt werden:

### *Operation Mode*

Hier können die Stromquellen für Pull-Up- oder Pull-Down-Betrieb konfiguriert werden oder deaktiviert werden (d.h. weder gegen einen Batterieknoten noch Batteriemasse konfiguriert).

### *Battery Node*

Hier kann die Batteriespannung gewählt werden, an die die Stromquelle verbunden wird. Zur Auswahl stehen die Batterieknoten 0..5, Dauerplus („PERM“) und keine Spannung („NONE“).

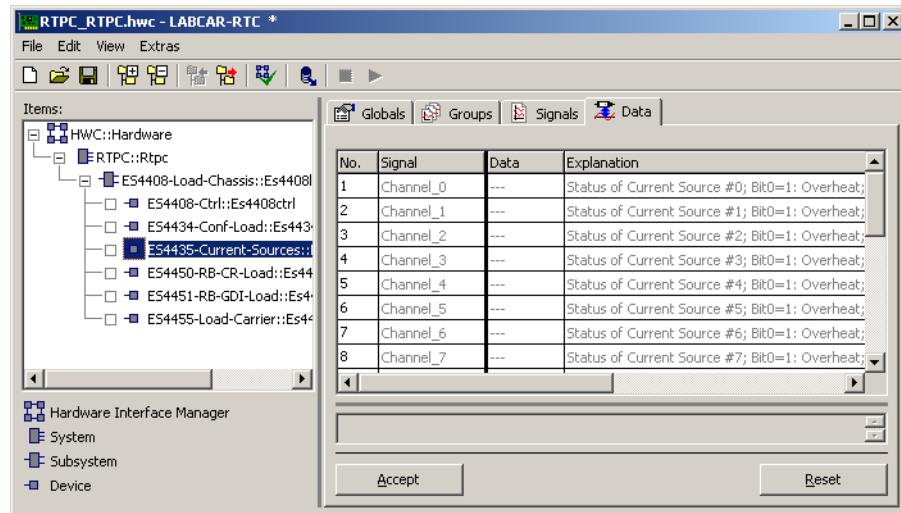
### *Current*

Hier wird der Strom eingestellt, den die Stromquelle liefern soll (5..150 mA).

18.4.3 Data (ES4435-Current-Sources Device)

In der Registerkarte „Data“ des ES4435-Current-Sources Device werden Statusinformationen zu jedem Kanal ausgegeben.

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES4435-Current-Sources Device.



**Abb. 18-9** Die Registerkarte „Data“ des ES4435-Current-Sources Device  
Die Bedeutung der Statusdaten finden Sie in der folgenden Tabelle:

RTIO-Signal	Datentyp	Bemerkung
Channel_xx [xx= 0...23]	uint8	Bit 0 = 1: Übertemperatur Bit 1 = 1: Leistung überschreitet 3 W (Strom wird reduziert)
Module Present	uint8	= 0: Das diesem RTIO-Item zugeordnete Board ist nicht vorhanden = 1: Das diesem RTIO-Item zugeordnete Board ist vorhanden und aktiv

**Tab. 18-3** ES4435-Current-Sources Device: Signale der Registerkarte „Data“

## 18.5 ES4450-RB-CR-Load Device

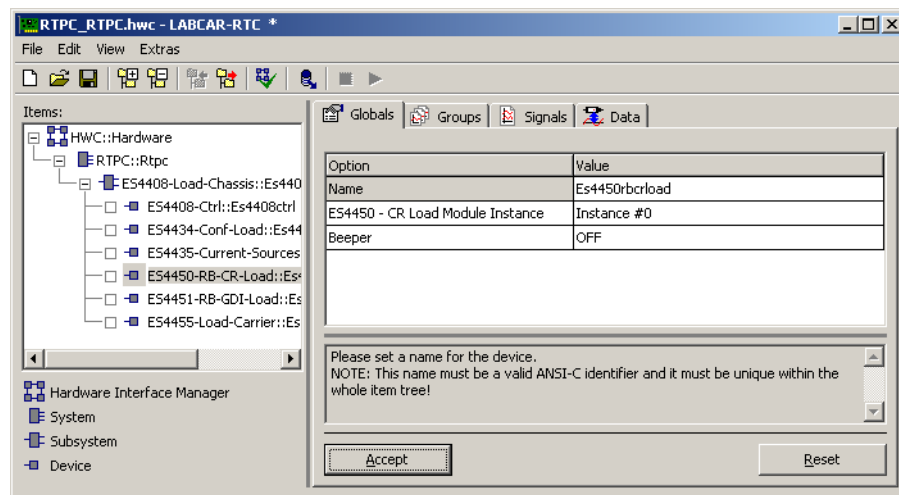
Das ES4450-RB-CR-Load Device repräsentiert ein im ES4408.1 Load Chassis eingebautes ES4450.2 Load Board for 4 RB CRS Injectors.

### **Hinweis**

*Die RTIO-Konfiguration eines ES4451.3 Load Board for 4 RB GDI Injectors oder eines ES4455.1 Load Board ist identisch mit der Konfiguration des hier beschriebenen ES4450.2 Load Board for 4 RB CRS Injectors und wird daher nicht gesondert beschrieben.*

### 18.5.1 Globals (ES4450-RB-CR-Load Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4450-RB-CR-Load Device.



**Abb. 18-10** Die Registerkarte „Globals“ des ES4450-RB-CR-Load Device

Dieses Register enthält folgende Parameter:

#### *Name*

Ein eindeutiger Bezeichner für dieses Device.

### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### *ES4450 - CR Load Module Instance*

Die internen Bezeichnungen dieser Instanz eines ES4450.2 Load Board for 4 RB CRS Injectors (Instance #0...Instance #2).

### Beeper

Mit dieser Option können Sie die Funktion für die akustische Signalisierung (beim Überschreiten der in der Registerkarte „Signals“ einstellbaren Schaltschwellen) aktivieren bzw. deaktivieren.

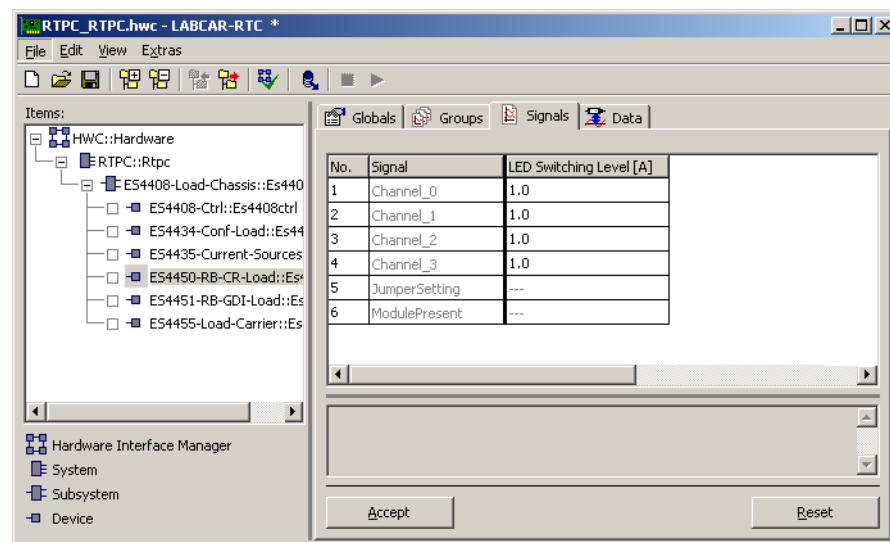
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Beeper	uint8	ja	0 = Off, 1 = On

**Tab. 18-4** ES4450-RB-CR-Load Device: Konfigurationsparameter der Registerkarte „Globals“

### 18.5.2 Signals (ES4450-RB-CR-Load Device)

In der Registerkarte „Signals“ des ES4450-RB-CR-Load Device können die Schaltschwellen für die LED-Anzeige auf der Frontplatte eingestellt werden.

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES4450-RB-CR-Load Device.



**Abb. 18-11** Die Registerkarte „Signals“ des ES4450-RB-CR-Load Device

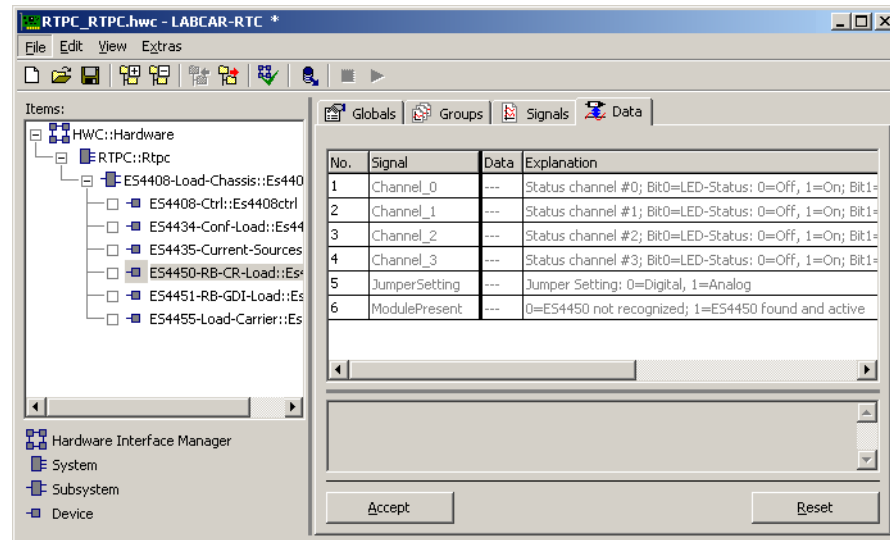
*Channel<sub>n</sub> (n = 0...3)*

Hier können Sie den Stromwert einstellen, nach dessen Überschreitung die entsprechende LED auf der Frontplatte leuchtet.

### 18.5.3 Data (ES4450-RB-CR-Load Device)

In der Registerkarte „Data“ des ES4450-RB-CR-Load Device werden Statusinformationen zu jedem Kanal ausgegeben.

Die folgende Abbildung zeigt die Registerkarte „Data“ des ES4450-RB-CR-Load Device.



**Abb. 18-12** Die Registerkarte „Data“ des ES4450-RB-CR-Load Device

Die Bedeutung der Statusdaten finden Sie in der folgenden Tabelle:

RTIO-Signal	Datentyp	Bemerkung
Channel_x [x = 0...3]	uint8	Bit 0 = LED-Status: 0 = LED an, 1 = LED aus Bit 1 = Hysterese-Status: 0 = low, 1 = high
Jumper Setting	uint8	Position des Steckbrückenleiste JP200: 0 = Strom wird digital ausgegeben 1 = Strom wird analog ausgegeben
Module Present	uint8	= 0: Das diesem RTIO-Item zugeordnete Board ist nicht vorhanden = 1: Das diesem RTIO-Item zugeordnete Board ist vorhanden und aktiv

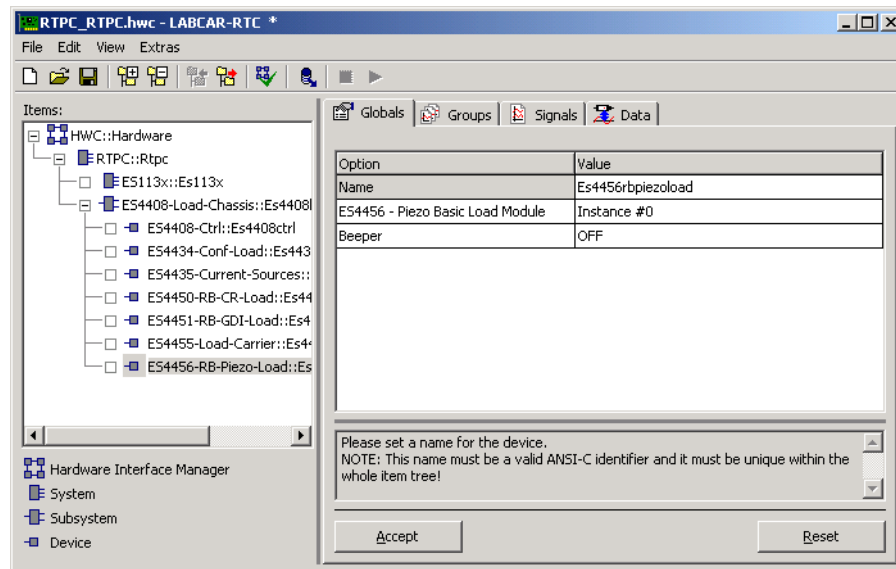
**Tab. 18-5** ES4450-RB-CR-Load Device: Signale der Registerkarte „Data“

## 18.6 ES4456-RB-Piezo-Load Device

Das ES4456-RB-Piezo-Load Device repräsentiert ein ES4456.1 Load Board for 8 RB Piezo Injectors.

### 18.6.1 Globals (ES4456-RB-Piezo-Load Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES4456-RB-Piezo-Load Device.



**Abb. 18-13** Die Registerkarte „Globals“ des ES4456-RB-Piezo-Load Device  
Dieses Register enthält folgende Parameter:

#### Name

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### ES4456 - Piezo Basic Load Module

Die internen Bezeichnungen dieser Instanz für ein ES4456.1 Load Board for 8 RB Piezo Injectors (Instance #0...Instance #2).

### Beeper

Mit dieser Option können Sie die Funktion für die akustische Signalisierung (beim Überschreiten der in der Registerkarte „Signals“ einstellbaren Schaltschwellen) aktivieren bzw. deaktivieren.

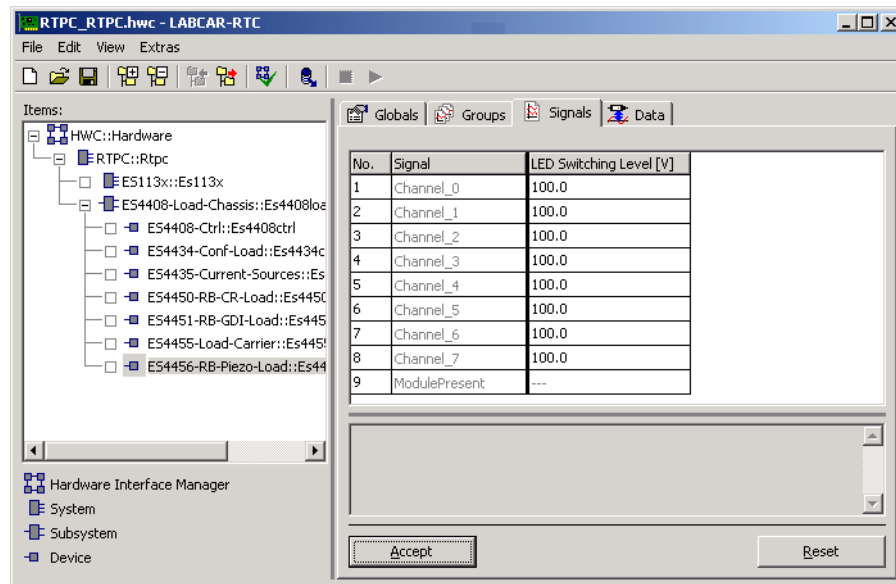
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Beeper	uint8	ja	0 = Off, 1 = On

**Tab. 18-6** ES4456-RB-Piezo-Load Device: Konfigurationsparameter der Registerkarte „Globals“

### 18.6.2 Signals (ES4456-RB-Piezo-Load Device)

In der Registerkarte „Signals“ des ES4456-RB-Piezo-Load Device können die Schaltschwellen für die LED-Anzeige auf der Frontplatte eingestellt werden.

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES4456-RB-Piezo-Load Device.



**Abb. 18-14** Die Registerkarte „Signals“ des ES4456-RB-Piezo-Load Device  $Channel_n$  ( $n = 0...7$ )

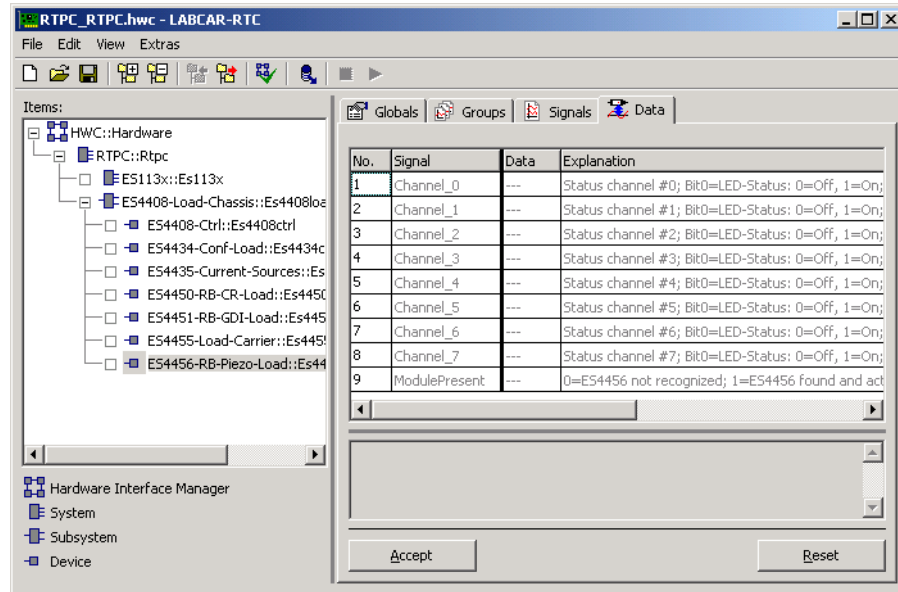
Hier können Sie den Spannungswert einstellen, nach dessen Überschreitung die entsprechende LED auf der Frontplatte leuchtet (0...250 V).

### 18.6.3 Data (ES4456-RB-Piezo-Load Device)

In der Registerkarte „Data“ des ES4456-RB-Piezo-Load Device werden Statusinformationen zu jedem Kanal ausgegeben.



Die folgende Abbildung zeigt die Registerkarte „Data“ des ES4456-RB-Piezo-Load Device.



**Abb. 18-15** Die Registerkarte „Data“ des ES4456-RB-Piezo-Load Device

Die Bedeutung der Statusdaten finden Sie in der folgenden Tabelle:

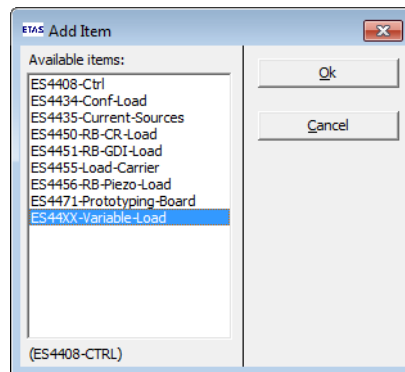
RTIO-Signal	Datentyp	Bemerkung
Channel_x [x = 0...7]	uint8	Bit 0 = LED-Status: 0 = LED an, 1 = LED aus Bit 1 = Überlast-Status: 0 = no, 1 = yes
Module Present	uint8	= 0: Das diesem RTIO-Item zugeordnete Board ist nicht vorhanden = 1: Das diesem RTIO-Item zugeordnete Board ist vorhanden und aktiv

**Tab. 18-7** ES4456-RB-Piezo-Load Device: Signale der Registerkarte „Data“

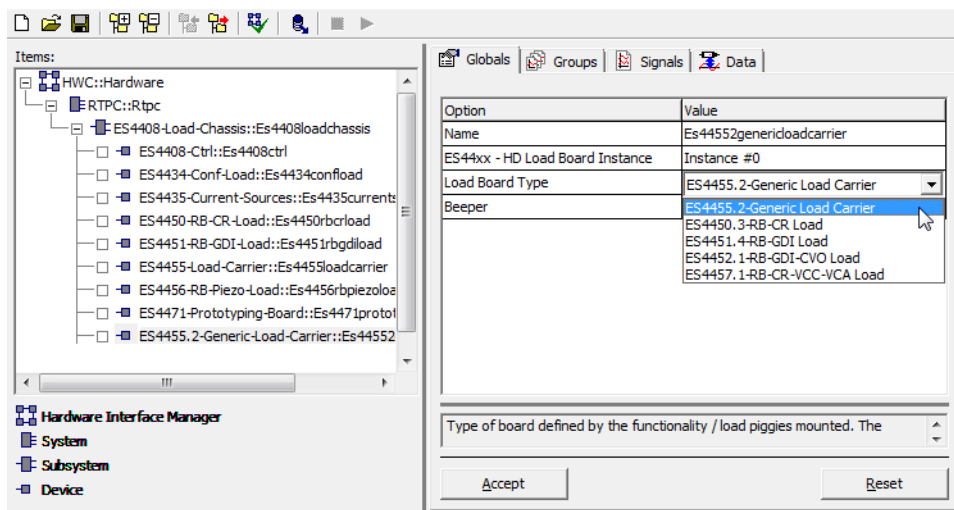
## 18.7 ES44XX-Variable-Load

Die ES44XX Variable-Load-Karten werden mittels eines einzelnen, generischen RTIO-Items dargestellt. Die Umschaltung zwischen den verschiedenen Kartentypen kann später mithilfe des Parameters „Load Board Type“ im Register „Globals“ vorgenommen werden (siehe „Load Board Type“ auf Seite 616).

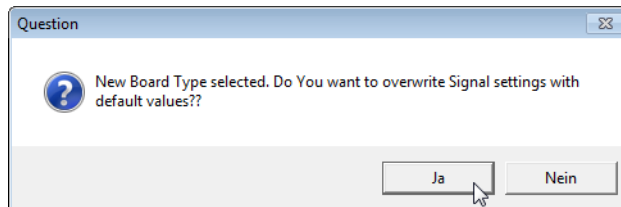
- Um eine Variable-Load-Karte in den RTIO-Baum einzufügen, selektieren Sie ein ES4408-Load-Chassis Item und wählen **Add Item** im Kontextmenü
- Sie können nun das Element „ES44XX-Variable-Load“ auswählen und das Item einfügen. Bei Einfügen ändert sich dessen Anzeigenamen in „ES4455.2-Generic-Load-Carrier“ - dies ist die generische Kartenvariante der variablen Last.



- Um eine andere Kartenvariante zu wählen, selektieren Sie diese mit dem Element „Load Board Type“.



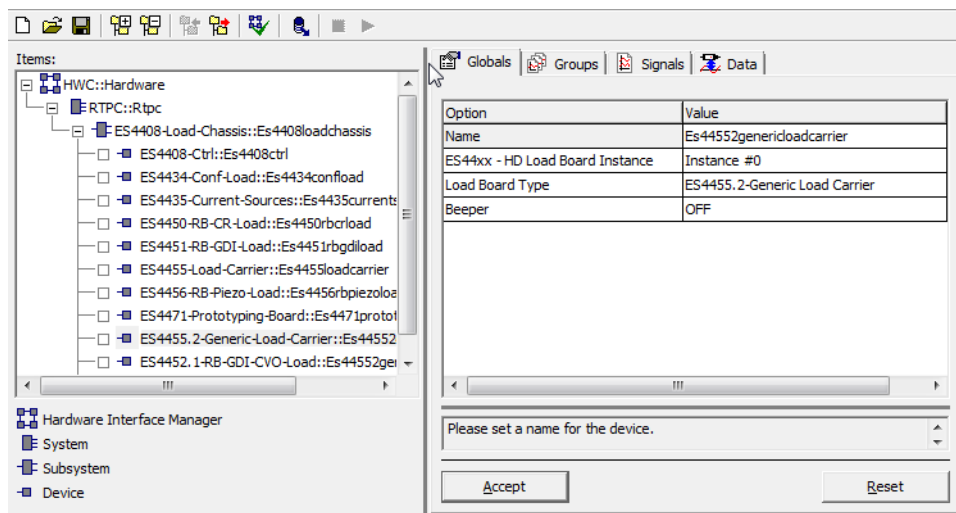
- Wählen Sie , ob Sie die den Parametersatz in der Registerkarte „Signals“ beibehalten oder diesen mit den kartenspezifischen Standardwerten überschreiben wollen.



- Um die Änderungen durchzuführen, klicken Sie **Accept**.

### 18.7.1 Globals (ES44XX-Variable-Load)

Die Registerkarte „Globals“ ist für alle Boards des Typs „Variable Load“ identisch.



**Abb. 18-16** Die Registerkarte „Globals“ des ES44XX-Variable-Load Device

#### Name

Der C-Name des Items – dieser wird als Prefix für alle globalen Bezeichner verwendet. Zur besseren Übersichtlichkeit kann es sinnvoll sein, diesen entsprechend anzupassen (Voreinstellung: „Es44552genericloadcarrier“).

#### ES44xx – HD Load Board Instance

Die Karteninstanz (Instance #0 ... Instance #2). Es können maximal drei „Variable Load“ Boards beliebigen Typs eingefügt werden.

#### Hinweis

*Eine Prüfung der Instanzen beim Konfigurieren findet nicht statt, d.h. es wird nicht verhindert, dass mehrere Karten gleichen Typs mit der gleichen Instanz konfiguriert werden. Bei der Konsistenzprüfung (beim Speichern der Konfiguration) findet diese Überprüfung aber statt.*

*Es ist auch zu beachten, dass Karten verschiedener Generationen gleichen Namens (z. B. ES4455.1 und ES4455.2) gemeinsam nummeriert werden.*

### Load Board Type

Der Typ der Lastkarte – daraus wird ein „SCPI Board Name“ abgeleitet, der zur Kommunikation mit der ES4408\_CON benutzt wird. Die kartenspezifischen Funktionalitäten sind in der ES4408\_CON-Firmware und im HDC der Lastkarte gekapselt.

Anzeigename	SCPI Board Name	Funktion
ES4455.2-Generic Load Carrier	ES4455_2	Generischer Kartenträger ohne spezifische Lasten
ES4450.3-RB-CR Load	ES4450_3	Diesel-Lasten
ES4451.4-RB-GDI Load	ES4451_4	Benziner-Lasten
ES4452.1-RB-GDI-CVO Load	ES4452	Benziner-Lasten mit CVO
ES4457.1-RB-CR-VCC-VCA Load	ES4457	Diesel-Lasten mit VCC/VCA

**Tab. 18-8** Mögliche Einstellungen für „Load Board Type“

### Beeper

Ein-/Ausschalten des Signalgebers der Lasten.

#### Hinweis

*Der Signalgeber ertönt nur dann, wenn mindestens eine der Lasten angesteuert wird.*

Darüberhinaus gibt es versteckte Einstellparameter, welche mit dem Kontextmenüpunkt **Show All** sichtbar gemacht werden können. Diese werden als Steuergrößen für die SCPI-Kommunikation der Signalgruppe „Control“ verwendet und sollten normalerweise nicht verändert werden.

### 18.7.2 Groups (ES44XX-Variable-Load)

Es werden vier Signalgruppen zur Verfügung gestellt, davon sind drei „Receive“ Signalgruppen.

Name der Signalgruppe	Wirkungsrichtung	Funktion
Status	Receive	Pegel der „I_Dig_Out“ Signale und Status der Last und der Schalter
Control	Send	Echtzeitsteuersignale der Lasten
Measure	Receive	Momentanwerte von Strom und Spannung der Lasten
Presence	Receive	Presence/Driver Ready status

**Tab. 18-9** Signalgruppen des ES44XX-Variable-Load Device

Die „Receive“ Signalgruppen werden zusammen mit allen anderen „Receive“ Signalgruppen des ES4408 Subsystems sequentiell gemäß den Einstellungen in der Registerkarte „Globals“ bedient.

Die „Send“ Signalgruppe hingegen wird asynchron zu den „Receive“ Signalgruppen abgearbeitet, d.h. SCPI-Kommandosequenzen können mit höherer Geschwindigkeit an eine ES4408\_COM kommuniziert werden.

Jedoch darf von der SCPI-Kommandosequenz der „Control“ Signalgruppe keine Rückmeldung erfolgen, da diese die Zuordnung der SCPI-Rückmeldungen zu den „Receive“ Signalgruppen beeinflussen können. Daher wird bei dem Debug-Level „SCPI DEBUG ON“ das Senden von SCPI-Sequenzen der „Control“ Signalgruppe unterbunden.

Generell wird eine SCPI-Sequenz nur bei Änderungen der zugrunde liegenden Eingangssignale generiert, d.h. ohne Signaländerung erfolgt keine SCPI-Kommunikation. Die maximale Häufigkeit einer SCPI-Kommunikation der Signalgruppe „Control“ wird durch eine Vorhaltezeit begrenzt, die verhindert, dass der SCPI-Kanal überlastet wird. Der Wert wird in der Registerkarte „Globals“ durch den normalerweise nicht sichtbaren Parameter „Control Guard Time“ festgelegt.

### 18.7.3 Signals (ES44XX-Variable-Load)

In der Registerkarte Signals können die Einstellungen der vier Lastkanäle über die Signale „I\_Dig\_Out\_0“ ... „I\_Dig\_Out\_3“ vorgenommen werden.

Ein Item wird durch eine Reihe von Einstellparametern und einer Kurvenform definiert – je nach gewähltem Lastkartentyp ist nur der Teil der für diesen Lastkartentyp benötigten Einstelloptionen sichtbar.

No.	Signal	End of Control Threshold VT1 [Volt]	Holding Phase Threshold VT2 [Volt]	Booster Phase Threshold VT4 [Volt]	End of Energizing Threshold IT1 [Ampere]	End of Control Threshold IT2 [Ampere]	Opening Valve Threshold IT3 [Ampere]	Current Rise Time Measurement IT4 [Ampere]	Valve Closing Delay [microseconds]	I-Dig-Out Offset [microseconds]
1	I_Dig_Out_0	-15.0	5.0	40.0	1.0	2.0	6.0	12.0	[Edit Curve]	0.0
2	I_Dig_Out_1	-15.0	5.0	40.0	1.0	2.0	6.0	12.0	[Edit Curve]	0.0
3	I_Dig_Out_2	-15.0	5.0	40.0	1.0	2.0	6.0	12.0	[Edit Curve]	0.0
4	I_Dig_Out_3	-15.0	5.0	40.0	1.0	2.0	6.0	12.0	[Edit Curve]	0.0
5	LoadActive_0	---	---	---	---	---	---	---	---	---
6	LoadActive_1	---	---	---	---	---	---	---	---	---
7	LoadActive_2	---	---	---	---	---	---	---	---	---
8	LoadActive_3	---	---	---	---	---	---	---	---	---
9	Enable_0	---	---	---	---	---	---	---	---	---
10	Enable_1	---	---	---	---	---	---	---	---	---
11	Enable_2	---	---	---	---	---	---	---	---	---
12	Enable_3	---	---	---	---	---	---	---	---	---
13	Offset_x_0	---	---	---	---	---	---	---	---	---
14	Offset_x_1	---	---	---	---	---	---	---	---	---
15	Offset_x_2	---	---	---	---	---	---	---	---	---
16	Offset_x_3	---	---	---	---	---	---	---	---	---
17	Offset_y_0	---	---	---	---	---	---	---	---	---
18	Offset_y_1	---	---	---	---	---	---	---	---	---
19	Offset_y_2	---	---	---	---	---	---	---	---	---
20	Offset_y_3	---	---	---	---	---	---	---	---	---
21	AnalogMuxer	---	---	---	---	---	---	---	---	---
22	I_Load_0	---	---	---	---	---	---	---	---	---
23	I_Load_1	---	---	---	---	---	---	---	---	---
24	I_Load_2	---	---	---	---	---	---	---	---	---
25	I_Load_3	---	---	---	---	---	---	---	---	---
26	U_Load_0	---	---	---	---	---	---	---	---	---
27	U_Load_1	---	---	---	---	---	---	---	---	---
28	U_Load_2	---	---	---	---	---	---	---	---	---
29	U_Load_3	---	---	---	---	---	---	---	---	---
30	I_Dig_Out_Status_0	---	---	---	---	---	---	---	---	---
31	I_Dig_Out_Status_1	---	---	---	---	---	---	---	---	---
32	I_Dig_Out_Status_2	---	---	---	---	---	---	---	---	---
33	I_Dig_Out_Status_3	---	---	---	---	---	---	---	---	---
34	ModulePresent	---	---	---	---	---	---	---	---	---

Abb. 18-17 Die Registerkarte „Signals“ des ES44XX-Variable-Load Device

Parameter	ES4455.2	ES4450.3	ES4451.4	ES4452	ES4457
VT1 - End of Control Threshold [ $\pm 128.0$ V]	Sichtbar -15.0	Sichtbar -15.0	Sichtbar -15.0	Sichtbar -15.0	Sichtbar -15.0
VT2 - Holding Phase Threshold [ $\pm 128.0$ V]	Sichtbar 5.0	Sichtbar 5.0	Sichtbar 5.0	Sichtbar 5.0	Sichtbar 5.0
VT3 - Reserved [ $\pm 128$ V]	Unsichtbar	Unsichtbar	Unsichtbar	Unsichtbar	Unsichtbar
VT4 - Booster Phase Threshold [ $\pm 128.0$ V]	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0
IT1 - End of Energizing Threshold [ $\pm 32.0$ A]	Unsichtbar 1.0	Unsichtbar 1.0	Unsichtbar 1.0	Sichtbar 1.0	Sichtbar 1.0
IT2 - End of Control Threshold [ $\pm 32.0$ A]	Sichtbar 2.0	Sichtbar 2.0	Sichtbar 2.0	Sichtbar 2.0	Sichtbar 2.0
IT3 - Opening Valve Threshold [ $\pm 32.0$ A]	Sichtbar 6.0	Sichtbar 6.0	Sichtbar 6.0	Sichtbar 6.0	Sichtbar 6.0
IT4 - Current Rise Time measurement [ $\pm 32.0$ A]	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0	Sichtbar 40.0
Valve Closing Delay t_close = f(t_control) [ $\mu$ s]	Unsichtbar Default-Tabelle	Unsichtbar Default-Tabelle	Unsichtbar Default-Tabelle	Sichtbar Default-Tabelle	Sichtbar Default-Tabelle
I-Dig-Out-Offset [ $\mu$ s]	Sichtbar 0.0	Sichtbar 0.0	Sichtbar 0.0	Sichtbar 0.0	Sichtbar 0.0

**Tab. 18-10** Die Parameter der Registerkarte „Signals“ und deren Sichtbarkeit und Defaultwerte  
Die Bedeutung der Schwellen VT1...4 und IT1...IT4 ist in Abschnitt 18.8.1 auf Seite 622 beschrieben.

Valve Closing Delay [microseconds]

Diese Einstelloption ist nur für CVO von Bedeutung.

Im FPGA der Lastkarte wird ein zusätzlicher Korrekturwert für die Einspritzzeit berechnet. Eingangsgröße ist die Ansteuerzeit des Ventils in  $\mu$ s, Ausgangsgröße ist der Korrekturwert des Einspritzendes, welches sowohl das „I\_Dig\_Out“ Ausgangssignal verlängert als auch die Generierung des CVO-Dips verschiebt.

Die Tabelle beschreibt bis zu 32 Punkte dieser Kurve, welche zur Konfigurationszeit in die ES4408 geladen wird.

I-Dig-Out-Offset [microseconds]

Diese Einstelloption verschiebt die öffnende Flanke des Einspritzsignals, d.h. die Einspritzzeit wird reduziert. Der Parameter kann als Korrekturwert benutzt werden. Im Gegensatz zur Korrektur des Einspritzendes kann hier jedoch nur ein statischer Korrekturwert eingestellt werden.

18.7.4 Data (ES44XX-Variable-Load Device)

In der Registerkarte „Data“ werden Pegel- und Statusinformationen der Empfangssignalgruppen dargestellt sowie Eingabewerte für die „Control“ Signalgruppe eingegeben.

No.	Signal	Data	Explanation
1	I_Dig_Out_0	---	Level of I-Dig-Out Load-Channel #0 [0=Low, 1=High]
2	I_Dig_Out_1	---	Level of I-Dig-Out Load-Channel #1 [0=Low, 1=High]
3	I_Dig_Out_2	---	Level of I-Dig-Out Load-Channel #2 [0=Low, 1=High]
4	I_Dig_Out_3	---	Level of I-Dig-Out Load-Channel #3 [0=Low, 1=High]
5	LoadActive_0	1.000	Switch to (dis-)connect Load #0 [0=Switch is Open; 1= Load is Connected]
6	LoadActive_1	1.000	Switch to (dis-)connect Load #1 [0=Switch is Open; 1= Load is Connected]
7	LoadActive_2	1.000	Switch to (dis-)connect Load #2 [0=Switch is Open; 1= Load is Connected]
8	LoadActive_3	1.000	Switch to (dis-)connect Load #3 [0=Switch is Open; 1= Load is Connected]
9	Enable_0	0.000	CVO: [0: Disable; 1=Enable Dip at Valve Closure]; VCC/VCA: [0: CRI1.16; 1=VCC/VCA Enabled; 2=VCC/VCA with Delay]
10	Enable_1	0.000	CVO: [0: Disable; 1=Enable Dip at Valve Closure]; VCC/VCA: [0: CRI1.16; 1=VCC/VCA Enabled; 2=VCC/VCA with Delay]
11	Enable_2	0.000	CVO: [0: Disable; 1=Enable Dip at Valve Closure]; VCC/VCA: [0: CRI1.16; 1=VCC/VCA Enabled; 2=VCC/VCA with Delay]
12	Enable_3	0.000	CVO: [0: Disable; 1=Enable Dip at Valve Closure]; VCC/VCA: [0: CRI1.16; 1=VCC/VCA Enabled; 2=VCC/VCA with Delay]
13	Offset_x_0	0.000	CVO only: Time Table x-Axis Offset Channel #0 added to x value of Time Table (before interpolation) [+/-500.0 microseconds]
14	Offset_x_1	0.000	CVO only: Time Table x-Axis Offset Channel #1 added to x value of Time Table (before interpolation) [+/-500.0 microseconds]
15	Offset_x_2	0.000	CVO only: Time Table x-Axis Offset Channel #2 added to x value of Time Table (before interpolation) [+/-500.0 microseconds]
16	Offset_x_3	0.000	CVO only: Time Table x-Axis Offset Channel #3 added to x value of Time Table (before interpolation) [+/-500.0 microseconds]
17	Offset_y_0	0.000	CVO only: Time Table y-Axis Offset Channel #0 added to y value of Time Table (after interpolation) [+/-500.0 microseconds]
18	Offset_y_1	0.000	CVO only: Time Table y-Axis Offset Channel #1 added to y value of Time Table (after interpolation) [+/-500.0 microseconds]
19	Offset_y_2	0.000	CVO only: Time Table y-Axis Offset Channel #2 added to y value of Time Table (after interpolation) [+/-500.0 microseconds]
20	Offset_y_3	0.000	CVO only: Time Table y-Axis Offset Channel #3 added to y value of Time Table (after interpolation) [+/-500.0 microseconds]
21	AnalogMuxer	0.000	Muxer for Analog Output Usage: [0...3=I_Load_0...3; 4...7=U_Load_0...3]
22	I_Load_0	---	Momentary Current through Load-Channel 0 [A]
23	I_Load_1	---	Momentary Current through Load-Channel 1 [A]
24	I_Load_2	---	Momentary Current through Load-Channel 2 [A]
25	I_Load_3	---	Momentary Current through Load-Channel 3 [A]
26	U_Load_0	---	Momentary Voltage over Load-Channel 0 [V]
27	U_Load_1	---	Momentary Voltage over Load-Channel 1 [V]
28	U_Load_2	---	Momentary Voltage over Load-Channel 2 [V]
29	U_Load_3	---	Momentary Voltage over Load-Channel 3 [V]
30	I_Dig_Out_Status_0	---	Err. Stat. Chan. #0: Bit0=I-Dig-Out Ovld; Bit1,2,3=Switch 1,2,3 Ovld; Bit4=Power Diss. [Bit=0: No Error, Bit=1: Error]
31	I_Dig_Out_Status_1	---	Err. Stat. Chan. #1: Bit0=I-Dig-Out Ovld; Bit1,2,3=Switch 1,2,3 Ovld; Bit4=Power Diss. [Bit=0: No Error, Bit=1: Error]
32	I_Dig_Out_Status_2	---	Err. Stat. Chan. #2: Bit0=I-Dig-Out Ovld; Bit1,2,3=Switch 1,2,3 Ovld; Bit4=Power Diss. [Bit=0: No Error, Bit=1: Error]
33	I_Dig_Out_Status_3	---	Err. Stat. Chan. #3: Bit0=I-Dig-Out Ovld; Bit1,2,3=Switch 1,2,3 Ovld; Bit4=Power Diss. [Bit=0: No Error, Bit=1: Error]
34	ModulePresent	---	Presence Status of ES44xx Load-Board [-1= No Info; 0=No Board; 1=Board Found and Active]

**Abb. 18-18** Die Registerkarte „Data“ mit eingblendeter Signalgruppe

*I\_Dig\_Out\_0...I\_Dig\_Out\_3*

Diese Signale liefern den Pegel des „I\_Dig\_Out“ Signals des jeweiligen Kanals.

Wertebereich: 0 = Low, 1 = High

*LoadActive\_0... LoadActive\_3*

Mit diesen Steuersignalen kann die Last des jeweiligen Kanales mittels eines Relais abgeworfen werden.

Wertebereich: 0 = Last abgeworfen, 1 = Last angeschlossen (Default)

*Enable\_0...Enable\_3*

Mit diesen Steuersignalen können bei den Lastkartentypen ES4452.1 (CVO) und ES4457.1 (VCC/VCA) spezifische Funktionen gesteuert werden.

- Beim CVO-Lastkartentyp kann mit diesen Signalen die Simulation der Unstetigkeit des Spannungssignales beim Schließen des Ventils (Dip) ein- bzw. ausgeschaltet werden.

Wertebereich: 0 = kein Dip, 1 = Dip wird generiert

- Beim VCC/VCA Lastkartentyp kann mit diesen Signalen zwischen Legacy-Betrieb (Injektor CRI1.16), VCC/VCA-Mode und VCC/VCA-Mode mit Verzögerung umgeschaltet werden.

Wertebereich:

- 0 = Legacy
- 1 = VCC/VCA an
- 2 = VCC/VCA mit zusätzlicher Verzögerung

*Offset\_x\_0...Offset\_x\_3, Offset\_y\_0...Offset\_y\_3*

Mit diesen Signalen kann beim CVO Lastkartentyp der Zeitpunkt des Auftretens des Dips beeinflusst werden.

Die „Offset\_x“ Signale werden vor der Interpolation in der „Valve Closing Delay“ Tabelle vor der Tabelleninterpolation zur Ansteuerdauer addiert. Die „Offset\_y“ Werte hingegen werden auf den resultierenden y-Wert der Tabelle addiert. Der resultierende Zeitwert bildet die Verzugszeit nach Ansteuerende zur Generierung des Dips.

Alle Offset-Signale sind Zeitwerte mit einem Wertebereich von  $\pm 500 \mu\text{s}$  – Werte außerhalb dieses Bereichs werden entsprechend begrenzt.

*AnalogMuxer*

Mit diesem Signal wird gesteuert, welcher der Lastströme oder eine der Lastspannungen auf den analogen Ausgang geschaltet wird. Der jeweilige Wertebereich wird hierbei auf den  $\pm 10 \text{ V}$ -Bereich des Analogausgangs skaliert.

Ströme:  $-32 \text{ A} \dots +32 \text{ A} \rightarrow -10 \text{ V} \dots +10 \text{ V}$

Spannungen:  $-128 \text{ V} \dots +128 \text{ V} \rightarrow -10 \text{ V} \dots +10 \text{ V}$

*I\_Load\_0...I\_Load\_3, U\_Load\_0...U\_Load\_3*

Diese Signale liefern die momentan gemessenen Spannungs- bzw. Stromwerte an den bzw. durch die Lasten. Die Messungen sind nicht zeitlich synchronisiert.

Ströme haben einen max. Wertebereich von  $\pm 32 \text{ A}$ , Spannungen von  $\pm 128 \text{ V}$ .



### *I\_Dig\_Out\_Status\_0... I\_Dig\_Out\_Status\_3*

Diese Signale stellen Status- und Fehlerinformationen der Lastkanäle zur Verfügung. Die Signale sind bitcodiert, wobei Bit = 0 „kein Fehler“ und Bit = 1 „Fehler“ bedeutet.

- Bit 0  
Fehlersignal des jeweiligen „I\_Dig\_Out“ Ausgangs, z. B. bei einem Kurzschluss desselben. Das Fehlerbit steht an, solange am Ausgang nicht erwartete Wert gelesen wird.
- Bits 1, 2, 3  
Überlastsignale der Schalter 1, 2,3. Diese Fehlerbits werden aus der Überstromerkennung der FETs abgeleitet. Die Fehlerbits werden erst beim Anhalten bzw. der nächsten Initialisierung des Experiments zurückgesetzt.
- Bit 4  
Abschaltung des Kanals wegen zu hoher Verlustleistung. Bei jedem Kanal wird die Temperatur aufgrund der Verlustleistung im HDC bilanziert. Bei einem zu hohen Wert wird die Last solange abgeworfen, bis wieder ein unkritischer Wert erreicht ist.

### *ModulePresent*

Dieses Signal zeigt an, ob die gewählte Lastkarteninstanz vorhanden und verwendbar ist.

Wertebereich:

- -1 = Keine Daten verfügbar
- 0 = keine verwendbare Karte
- 1 = Karte vorhanden und verwendbar

## 18.8 Injektorsimulation mit ES4452.1 (Benziner) und ES4457.1 (Diesel)

Die Produkte ES4452.1 (Benziner) und ES4457.1 (Diesel) ermöglichen die Simulation von „intelligenten/aktiven“ Injektoren bzw. neuen Regelalgorithmen zur hochgenauen Treibstoffdosierung in Steuergeräten.

Die Produkte ES4451.4 (Benziner) und ES4450.3 (Diesel) sind passiver Natur und simulieren die elektronische Last eines Injektors. Das Vorgängerprodukt der ES4451.4 ist die ES4451.3 (basierend auf dem ES4455.1 Load Board), das der ES4450.3 ist die ES4450.2 (basierend auf dem ES4455.1 Load Board).

### 18.8.1 HDEV5 Injektorsimulation (CVO) mit ES4452.1

Die ES4452.1 simuliert HDEV5 Injektoren und ermöglicht die Stimulation von CVO-Regelalgorithmen. CVO steht für „Controlled Valve Operation“ und nutzt physikalische Effekte bzw. Merkmale in elektrischen Signalen bei der Injektoransteuerung zur genauen Bestimmung des tatsächlichen Injektorschließzeitpunktes.

Ein Signal „I\_Dig\_Out“ ist low-aktiv und stellt normalerweise die Ventilöffnungszeit dar. Das Signal „I\_Dig\_Out“ der jeweiligen Injektorsimulation kann einer winkelsynchronen Zeitvermessung eines HIL-Systems (z.B. ES5340.2-ICE) zugeführt werden, um einem Verbrennungsmotormodell die aktuelle Einspritzzeit zu kommunizieren.

Das Auftreten der fallende Flanke des Signals „I\_Dig\_Out“ kann entweder durch die Verstellung der Strom-/Spannungsschwelle (siehe Tab. 18-11 auf Seite 626) herbeigeführt werden oder über den Parameter „I\_Dig\_Out\_Offset“.

#### Strom- und Spannungsschwellen

Um die gewünschte Funktion zu erhalten, müssen diverse Strom- und Spannungsschwellen konfiguriert werden. Im Folgenden werden diese Schwellen anhand von typischen Strom- Spannungskurven für CVO und VCC beschrieben sowie deren Einfluss auf die Generierung der „I\_Dig\_Out“-Signale.

Die Ansteuerung des Injektors kann grob in vier Bereiche unterteilt werden (siehe Abb. 18-19 auf Seite 623):

1. Booster Phase
2. Pickup Phase
3. Holding Phase
4. Switching-off Phase

Diese vier Bereiche werden in verschiedenen produktinternen Kombinationen über Schwelleneinstellung bzw. Triggerung ermittelt.

#### Timetable

Bei der CVO-Funktion des simulierten Injektors ist es wichtig, in LABCAR-RTC eine sogenannte Timetable zu konfigurieren. Die Timetable sollte mindestens die Abhängigkeit des Auftretens des Merkmals (tab) zur Injektoransteuerdauer (ti) abbilden.

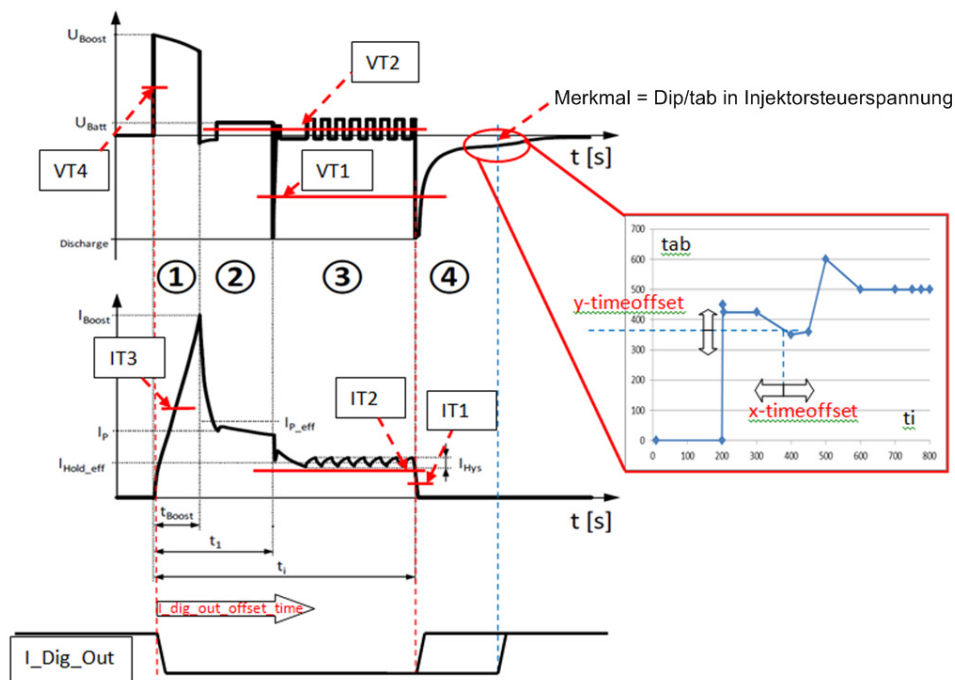
Die Injektoransteuerdauer wird über die Zeitdifferenz zwischen „Überschreiten von VT4“ und dem letzten „Unterschreiten der VT1-Schwelle“ mit dazu passenden Stromschwellen berechnet. Eine Korrektur der daraus resultierenden Dauer bzw. Start/Beginn des Signals „I\_Dig\_Out“ kann über die Einstellung von Schwellen, der „I\_Dig\_Out\_Offset\_Time“, der Timetable und den x/y-Time-

offsets erfolgen. Weitere Einflussgrößen können Treibstoffqualität, Raildruck, usw. sein – die passenden Kurven sollten Sie bei Ihrem Injektorlieferanten erfragen.

Als Beispiel wird in Abb. 18-19 eine „unrealistische“ (eventuell für SW-Tests relevante) Kurve dargestellt (Achseneinheit in Mikrosekunden). Mittels Echtzeitparametern (x/y-Timeoffset) kann die Lage der Kurve während der Laufzeit über das Modell beeinflusst werden. Dies wird durch eine simple Addition der Timetablewerte mit den x/y-Timeoffset-Werten erreicht.

Es wird empfohlen, die Defaultwerte der Strom-/Spannungsschwellen zu verwenden (siehe Tab. 18-11 auf Seite 626).

Die Amplitude des Merkmals kann nicht direkt manipuliert werden und ist abhängig von der eingekoppelten Energie in die Injektorsimulation. Es besteht der qualitative Zusammenhang, dass die Amplitude des Merkmals umso größer ist, je früher das Merkmal nach der Schnelllöschung (letztes VT2-Ereignis) erzeugt werden soll.



**Abb. 18-19** Typischer HDEV5 CVO-Injektorstrom-/Injektorspannungsverlauf mit Merkmal und „I\_Dig\_Out“ Signal

18.8.2 CR12-x Injektorsimulation (VCC/VCA) mit ES4457.1

Die ES4457.1 simuliert CR12-x Injektoren und ermöglicht die Stimulation von VCC/VCA Regelalgorithmen. VCC/VCA steht für „Valve Closing Control“ bzw. „Valve Closing Adjustment“ und nutzt physikalische Effekte bzw. Merkmale in elektrischen Signalen bei der Injektoransteuerung zur genauen Bestimmung von tatsächlichen Injektoröffnungszeiten.

Ein typischer Stromverlauf eines CR12-x Injektors ist in Abb. 18-20 dargestellt. Die einstellbaren Stromschwellwerte sind in rot eingezeichnet und können in der Registerkarte „Signals“ verändert werden. Die voreingestellten Werte und deren Funktion sind in Tab. 18-11 auf Seite 626 beschrieben.



Schwelle	ES4455.2	ES4452.1 (CVO-Benziner)	ES4457.1 (VCC-Diesel)	ES4451.4 (Benziner)	ES4450.3 (Diesel)
Injektorstromschwelle IT1 (intern)	Keine Funktion	Interne Verwendung (Defaultwert: 1 A)	Interne Verwendung (Defaultwert: 1 A)	Interne Verwendung (Defaultwert: 1 A)	Interne Verwendung (Defaultwert: 1 A)
Injektorstromschwelle IT2 (intern)	Keine Funktion	Interne Verwendung (Defaultwert: 2 A). Muss deutlich unter dem minimalen Wert des Haltestromripples sein.	Ansteuerende; Steigende Flanke von I_Dig_Out_x (wenn VCC deaktiviert); Verlängerung von I_Dig_Out_x auf Strommaximum des Merkmals (VCC aktiviert). Muss deutlich unter dem minimalen Wert des Haltestromripples sein (Defaultwert: 1 A).	Ansteuerende; Steigende Flanke von I_Dig_Out_x	Ansteuerende; Steigende Flanke von I_Dig_Out_x
Injektorstromschwelle IT3	Keine Funktion	Keine Funktion	Keine Funktion	Ansteuerbeginn; Fallende Flanke von I_Dig_Out_x	Ansteuerbeginn; Fallende Flanke von I_Dig_Out_x
Injektorstromschwelle IT4 (intern)	Keine Funktion	Interne Verwendung (Defaultwert: 12 A)	Interne Verwendung (Defaultwert: 12 A)	Keine Funktion	Keine Funktion

Schwelle	ES4455.2	ES4452.1 (CVO-Benziner)	ES4457.1 (VCC-Diesel)	ES4451.4 (Benziner)	ES4450.3 (Diesel)
Injektorspannungsschwelle VT1	Keine Funktion	Ansteuerende bzw. Pickupphase-Detektion (Defaultwert: -15 V); Steigende Flanke von I_Dig_Out_x (wenn CVO deaktiviert); Start tab-Generierung in Abhängigkeit von $t_i$ = Zeit zwischen VT4 und VT1 Ereignis wenn IT1 unterschritten wurde, bzw. timetable-Wert	Pickupphase-Detektion (Defaultwert: -15 V)	Keine Funktion	Keine Funktion
Injektorspannungsschwelle VT2	Keine Funktion	Detektion der PWM Signale in der Haltephase (Defaultwert: 5 V)	Detektion der PWM-Signale in der Haltephase (Defaultwert: 5 V)	Keine Funktion	Keine Funktion
Injektorspannungsschwelle VT4	Keine Funktion	Ansteuerbeginn; Fallende Flanke von I_Dig_Out_x	Ansteuerbeginn; Fallende Flanke von I_Dig_Out_x	Keine Funktion	Keine Funktion

**Tab. 18-11** Schwellwerte und deren Funktion/Einfluss auf „I\_Dig\_Out“ bzw. Merkmalsgenerierung

## 19 ES5300.1 Housing

Das ES5300.1 Housing ist ein Systemgehäuse zum Aufbau eines Hardware-in-the-Loop Testsystems. Digitale und analoge Schnittstellen zum Steuergerät in Form von PCI Express, SPI oder I<sup>2</sup>C basierten Karten können in das ES5300.1 Housing eingebaut werden.

### 19.1 ES5300-Chassis System

Dieses RTIO-Element repräsentiert ein ES5300.1 Housing und ermöglicht eine hierarchische Darstellung der Einsteckkarten.

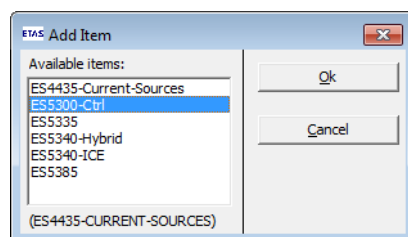
In einem ES5300.1 Housing sind prinzipiell zum Einbau vorgesehen:

- ETAS und 3rd Party PCIe Steckkarten im PC Formfaktor mittels eines ES5370.1 Adapters
- PCIe-Steckkarten im ES5300 Formfaktor \*
- SPI-Steckkarten im ES5300 Formfaktor
- ES443x Lastkarten mittels ES5371.1 Adapter
- ES445x Lastkarten mittels ES5372.1 Adapter \*

\* derzeit noch keine Unterstützung vorhanden.

In dem RTIO-System können jedoch nur die ETAS I/O-Karten dargestellt werden. Bezüglich der Handhabung der ETAS Karten gelten folgende Unterschiede und Regeln, die zu beachten sind:

- Die Instanzierung der PCIe-Karten gilt systemweit, wobei gleichartige Karten in aufsteigender Reihenfolge nummeriert werden, beginnend bei „ES5300-Chassis“ Instanz 0, Slot 0.
- Die Anordnung der PCIe-Karten mit PC-Formfaktor in einem ES5300-Chassis System dient nur der Übersichtlichkeit. Sie können auch direkt unter dem „RTPC“ RTIO-Element angeordnet werden. Die Auswahl der Karte erfolgt einzig über die Instanznummer.
- SPI-Karten gleichen Typs hingegen werden innerhalb jedes „ES5300-Chassis“ RTIO-Elements aufsteigend nummeriert, beginnend mit Slot 0 und Instanznummer 0.
- Die für die ES44xx-Lastkarten nötigen Adapter sind RTIO-seitig transparent und werden daher nicht dargestellt.

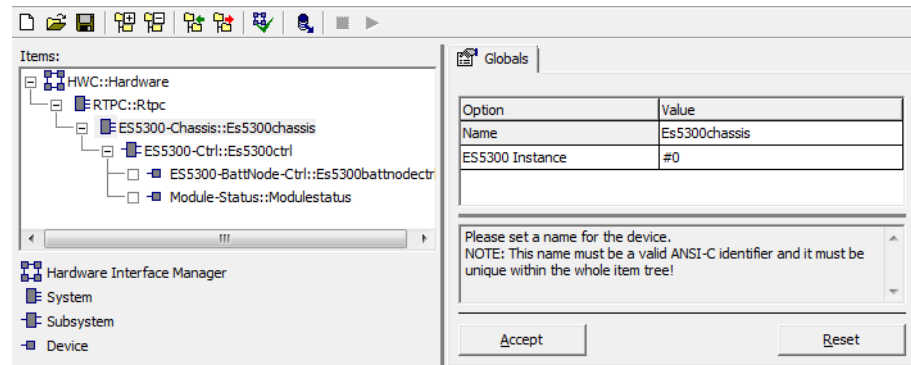


**Abb. 19-1** Die unter einem ES5300-Chassis verfügbaren RTIO Elemente

### 19.1.1 Globals (ES5300-Chassis System)

In der Registerkarte „Globals“ des ES5300-Chassis RTIO-Elements kann dessen Instanznummer verändert werden. Die Vergabe der Instanznummern bei mehreren ES5300 Systemen hängt von der Topologie des PCIe- und des SYNC-Busses ab.

Die Instanznummer wird von den RTIO-Treibern der SPI-Karten und dem Treiber benötigt, der dem ES5300-Ctrl Subsystem zugeordnet ist.

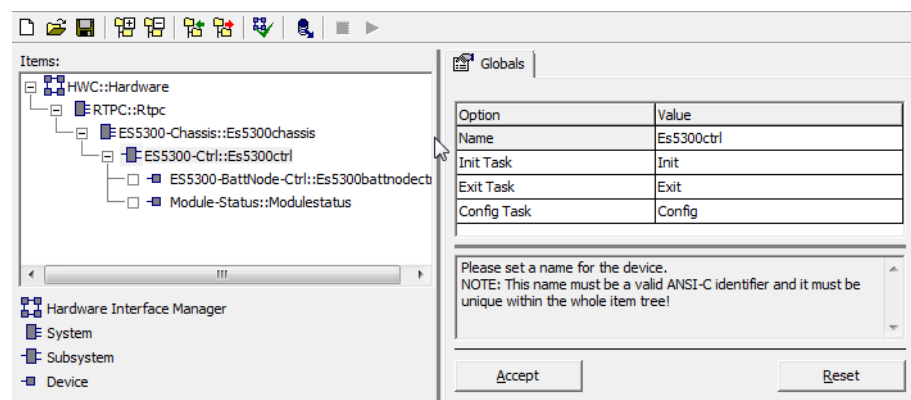


**Abb. 19-2** Die Registerkarte „Globals“ des ES5300-Chassis Systems

#### Hinweis

*Wenn nur ein ES5300.1-A Housing verwendet wird, sollte „ES5300 Instance“ den Wert „#0“ haben – wird zusätzlich ein ES5300.1-B Housing eingesetzt, muss dafür die „ES5300 Instance“ auf den Wert „#1“ gesetzt werden!*

### 19.2 ES5300-Ctrl Subsystem



**Abb. 19-3** Die Registerkarte „Globals“ des ES5300-Ctrl Subsystems

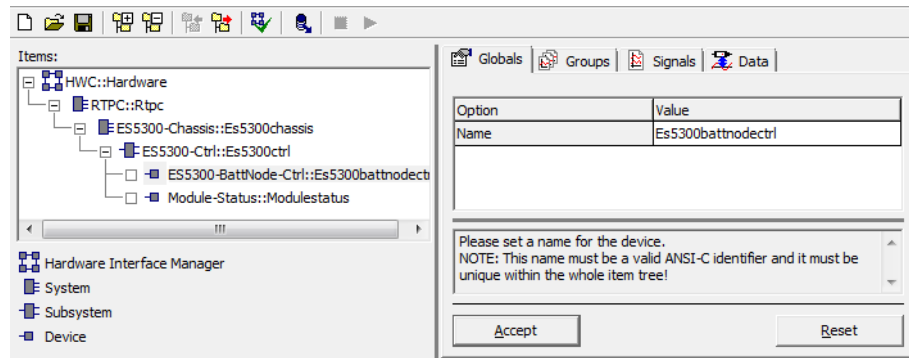


### 19.3 ES5300-BattNode-Ctrl Device

Das ES5300-BattNode-Ctrl Device stellt die Treiberebene des ES5300.1 Housing dar – der Treiber ermöglicht die Ansteuerung der Batterieknotsignale.

#### 19.3.1 Globals (ES5300-BattNode-Ctrl)

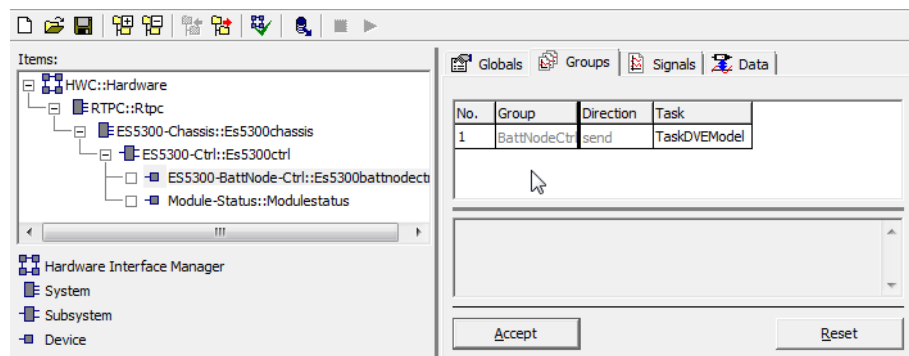
In der Registerkarte „Globals“ kann der Devicename verändert werden. Er wird bei der C-Codegenerierung als Prefix für alle globalen Bezeichner verwendet.



**Abb. 19-4** Die Registerkarte „Globals“ des ES5300-BattNode-Ctrl Device

#### 19.3.2 Groups (ES5300-Batt-Node)

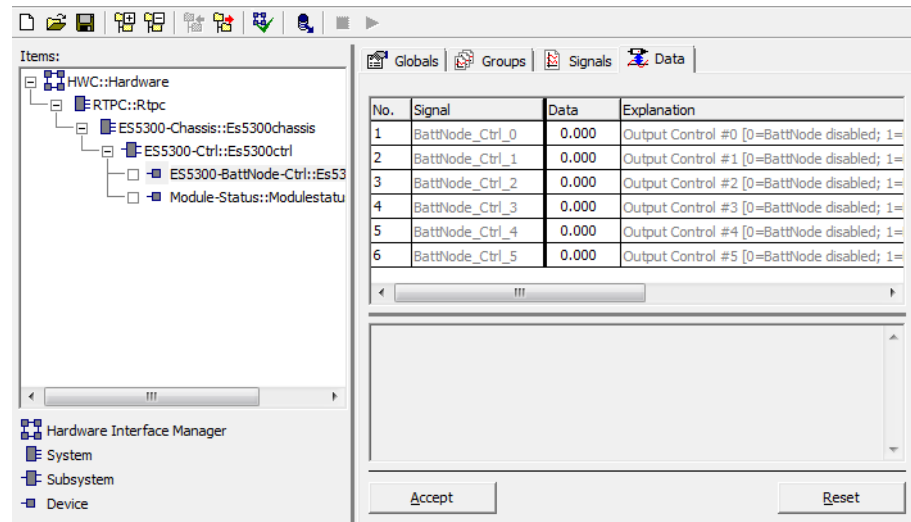
Das ES5300-Batt-Node Device implementiert eine Send-Signalgruppe „BattNodeCtrl“.



**Abb. 19-5** Die Registerkarte „Groups“ des ES5300-BattNode-Ctrl Device

### 19.3.3 Data (ES5300-Batt-Node)

Zur Ansteuerung der auf die Backplane geführten Signale für die Batterieknotensteuerung stehen sechs Signale zur Verfügung.

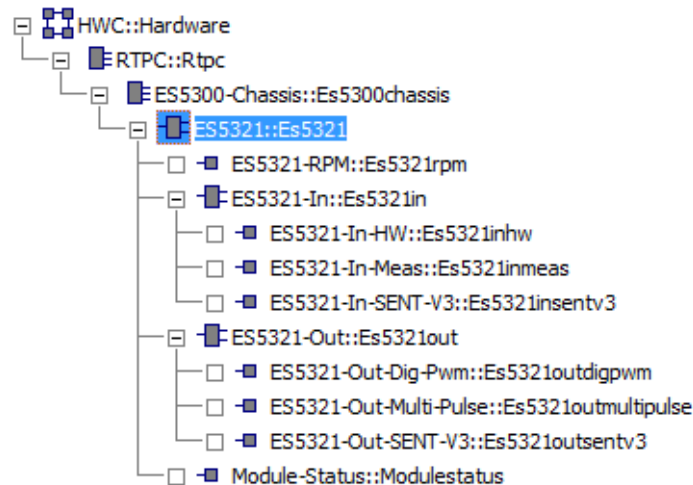


**Abb. 19-6** Die Registerkarte „Data“ des ES5300-BattNode-Ctrl Device

## 20 ES5321.1 PWM I/O Board

Das ES5321.1 PWM I/O Board dient der Ausgabe und Erfassung von pulswertenmodulierten Signalen. Der typische Anwendungsfall für die ES5321.1 in LAB-CAR-Projekten ist die Nachbildung von Fahrzeugsensoren und die Ansteuerung von Aktoren.

Im RTIO-Editor wird das ES5321.1 PWM I/O Board durch Auswahl des ES5321 Subsystems eingebunden.



**Abb. 20-1** RTIO-Hardwarebeschreibung mit ES5321.1 PWM I/O Board

Das ES5321.1 PWM I/O Board besitzt 48 Eingangskanäle (vier galvanisch getrennte Gruppen mit jeweils zwölf Kanälen) zur Erfassung von pulswertenmodulierten Signalen – die Konfiguration dieser Eingangskanäle erfolgt mit dem ES5321-In-HW Subsystem.

Neben diesem Subsystem können dem ES5321-RTIO-Element bis zu vier ES5321-In-SENT und acht ES5321-In-Meas Subsysteme zugeordnet werden.

Die bis zu vier ES5321-Out Subsysteme können aus jeweils einem ES5321-Out-Dig-Pwm, bis zu vier ES5321-Out-Multi-Pulse Devices und bis zu zwölf ES5321-Out-SENT Devices bestehen.

### *Messfunktionen des ES5321.1 PWM I/O Boards*

---

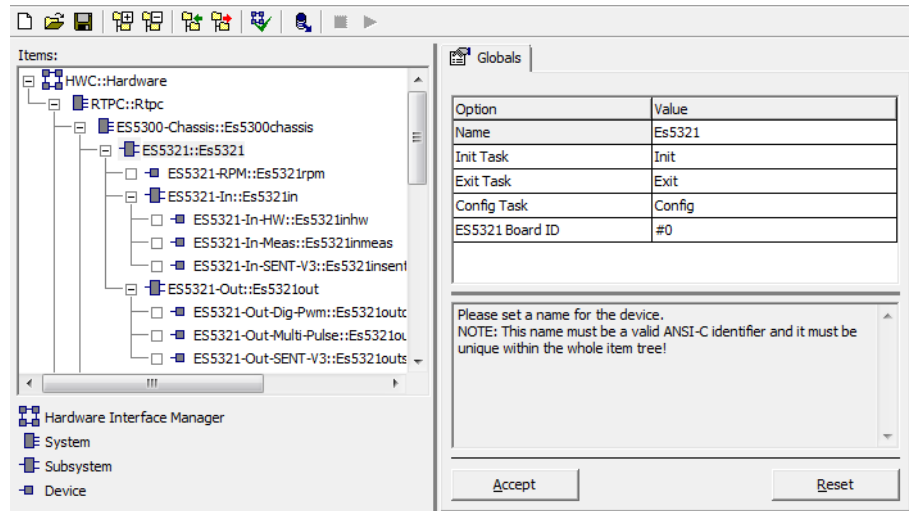
Die Eingänge der ES5321.1 verfügen über folgende Messfunktionen:

- Level (Active High)
- Level (Active Low)
- High Time [ $\mu$ s]
- Low Time [ $\mu$ s] (yes)
- Additive High Time [ $\mu$ s]
- Additive Low Time [ $\mu$ s]
- Cycle Time --/-- [ $\mu$ s]
- Frequency --/-- [Hz]
- Duty Cycle L/(L+H) --/--
- Duty Cycle H/(L+H) --/--
- Cycle Time --\-- [ $\mu$ s]
- Frequency --\-- [Hz]
- Duty Cycle L/(L+H) --\--
- Duty Cycle H/(L+H) --\--

## 20.1 ES5321 Subsystem

### 20.1.1 Globals (ES5321 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES5321 Subsystems beschrieben.



**Abb. 20-2** Die Registerkarte „Globals“ des ES5321 Subsystems

#### ES5321 Board ID

Dieses Optionsfeld dient zur Identifizierung des ES5321.1 PWM I/O Boards.

Im RTIO-Editor können bis zu zehn ES5321.1 PWM I/O Boards pro ES5300 Bau-  
gruppenträger eingebunden werden.

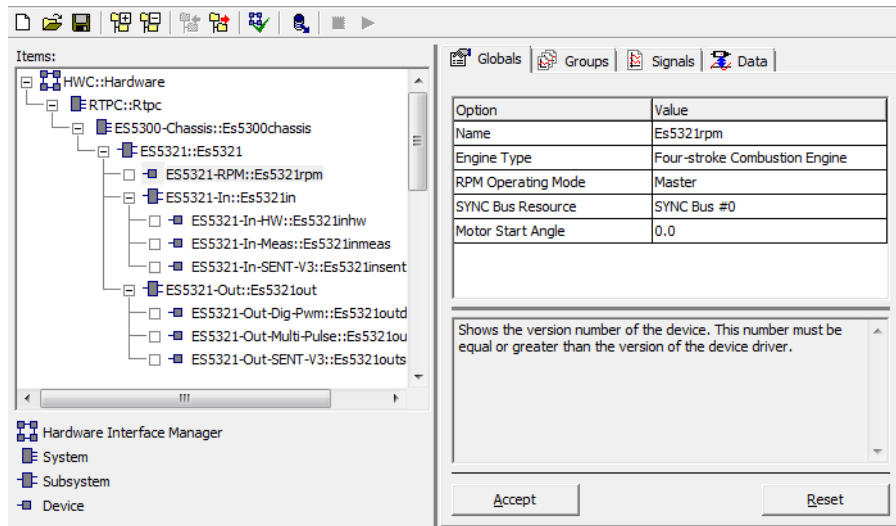
Dieser RTIO-Parameter ist während der Laufzeit des Modells auf dem Experimen-  
taltarget nicht editierbar.

## 20.2 ES5321-RPM Device

Das ES5321-RPM Device dient zu Konfiguration und Steuerung der ES5321-Drehzahleinheit.

### 20.2.1 Globals (ES5321-RPM)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 20-3** Die Registerkarte „Globals“ des ES5321-RPM Device

#### *Engine Type*

Auswahl des Motortyps.

#### *RPM Operating Mode*

Auswahl zwischen Master- oder Slave-Betrieb.

#### **Hinweis**

Die Optionen „Engine Type“ und „RPM Operating Mode“ sind nur sichtbar, wenn zuvor im Menü des Editors die Einstellung **View** → **Show All** gewählt wurde!

#### *SYNC Bus Resource*

Das Winkeltaktsignal wird über Synchronisationsbus-Leitungen zwischen den Karten transportiert. Mit diesem Konfigurationsparameter wird die Busleitung ausgewählt, über die der Winkeltakt im Slave-Betrieb eingelesen wird oder im Master-Betrieb ausgegeben wird.

Wird die Drehzahleinheit auf Master-Betrieb konfiguriert, so steht zusätzlich noch die Option „None“ zur Auswahl – in diesem Fall wird ein Winkeltakt generiert und intern auf der Karte verwendet, jedoch nicht auf einen der beiden Synchronisationsbusse ausgegeben.

#### **Hinweis**

*Bitte beachten Sie, dass es auf einem Synchronisationsbus nur jeweils einen Winkeltakt-Master geben darf!*

#### Motor Start Angle

Der Winkel, bei dem der Motor nach der Modellinitialisierung losläuft.

Die folgende Tabelle fasst die Eigenschaften der einzelnen Konfigurationsparameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Engine Type	int	Nein	0: Viertakt-Verbrennungsmotor 1: E-Motor 2: Zweitakt-Verbrennungsmotor
RPM Operating Mode	int	Ja	0: Slave 1: Master
SYNC Bus Resource	int	Ja	0: SYNC Bus #0 1: SYNC Bus #1 255: None (nur bei Masterbetrieb)
Motor Start Angle	float	Ja	0.0 °RW...360.0 °RW: E-Motor 0.0 °KW...360.0 °KW: Zweitaktmotor 0.0 °KW...720.0 °KW: Viertaktmotor

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 20-1** Die Registerkarte „Globals“ des ES5321-RPM Device

#### **Hinweis**

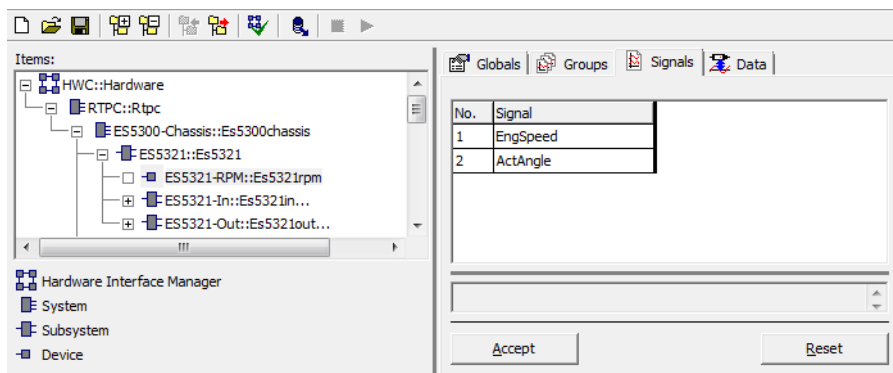
*Beachten Sie bitte, dass je nach hier gewähltem Motortyp („Engine Type“) die Winkelgrenzen unterschiedlich sind (720 ° bzw. 360 °). Dies kann sich auch entsprechend auf den Wertebereich anderer Parameter auswirken!*

#### 20.2.2 Groups (ES5321-RPM)

Das ES5321-RPM Device besitzt eine Signalgruppe „Outputs“ zur Steuerung der Drehzahleinheit und eine Signalgruppe „Inputs“ zum Lesen von aktuellen Statusinformationen der Drehzahleinheit.

#### 20.2.3 Signals (ES5321-RPM)

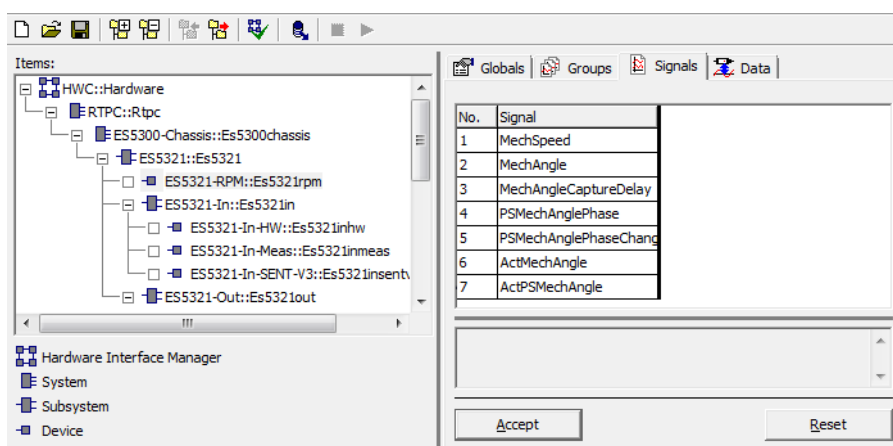
Die in der Registerkarte „Signals“ angebotenen Signale zur Steuerung und Statusüberwachung der Drehzahleinheit sind vom gewählten Motortyp abhängig. Abb. 20-4 auf Seite 636 zeigt die RTIO-Signale für den Motortyp „Verbrennungsmotor“, Abb. 20-5 auf Seite 636 zeigt die RTIO-Signale für den Motortyp „E-Motor“. In Tab. 20-2 auf Seite 636 und Tab. 20-3 auf Seite 637 sind die Signaleigenschaften noch einmal zusammengefasst.



**Abb. 20-4** Die Registerkarte „Signals“ des ES5321-RPM Device (Verbrennungsmotor)

Signal-name	Daten-typ	Anmerkungen
EngSpeed	float	Motordrehzahl in Umdrehungen pro Minute Wertebereich: Viertaktmotor: -60000.0 rpm...+60000.0 rpm, Auflösung: 0.001 rpm Zweitaktmotor: -30000.0 rpm...+30000.0 rpm, Auflösung: 0.0005 rpm Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.
ActAngle	float	Aktueller Kurbelwellenwinkel in °KW Wertebereich: 0.0 °KW ... 720.0 °KW (oder 360 °KW) Auflösung: 0.011 °KW (oder 0.0055 °KW)

**Tab. 20-2** ES5321-RPM Device: RTIO-Signale (Verbrennungsmotor)



**Abb. 20-5** Die Registerkarte „Signals“ des ES5321-RPM Device (E-Motor)



Signalname	Datentyp	Anmerkungen
MechSpeed	float	Motordrehzahl in Umdrehungen pro Minute Wertebereich: -30000.0 rpm ... +30000.0 rpm Auflösung: 0.001 rpm Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.
MechAngle	float	Zielwinkelvorgabe in Grad Wertebereich: 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW Dieses Signal ist nur dann von Bedeutung, wenn die Drehzahleinheit als „Master mit Winkeladaption“ betrieben wird.
MechAngle-CaptureDelay	float	Dieses Signal definiert die Zeit (in µs), um die die Zielwerte für Drehzahl und Winkel verzögert (bezogen auf das Triggerereignis) von der Hardware übernommen werden. Maximaler Wert: 1000.0 µs Dieses Signal ist nur dann von Bedeutung, wenn die Drehzahleinheit als „Master mit Winkeladaption“ betrieben wird.
PSMechAngle-Phase	float	Offset des phasenverschobenen mechanischen Winkels relativ zum Rotorwinkel Wertebereich: -360.0 °RW ... 360.0 °RW Auflösung: 0.0055 °RW
PSMechAngle-PhaseChange-Speed	float	Änderungsgeschwindigkeit des phasenverschobenen mechanischen Winkels relativ zum Rotorwinkel nach einem „PSMechAnglePhase“-Sprung (in Grad pro Sekunde) Maximale Änderungsgeschwindigkeit: 1000.0 °RW/s Wird hier als Wert 0 eingetragen, so wird die Ziel-Phasenverschiebung sofort eingestellt.
ActMechAngle	float	Aktueller Rotorwinkel in Grad Wertebereich 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW
ActPSMechAngle	float	Aktueller Wert des zum Rotorwinkel phasenverschobenen Winkels in Grad Wertebereich 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW

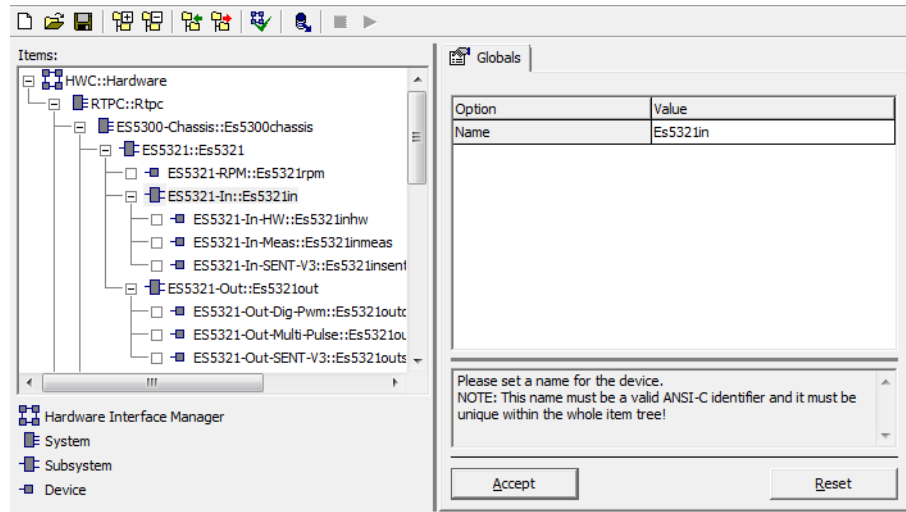
**Tab. 20-3** ES5321-RPM Device: RTIO-Signale (E-Motor)

## 20.3 ES5321-In Subsystem

### 20.3.1 Globals (ES5321-In Subsystem)

Das ES5321-In Subsystem dient zur Konfiguration der Messeingänge des ES5321.1 PWM I/O Boards.

Abb. 20-6 zeigt die Registerkarte „Globals“ des ES5321-In Subsystems.



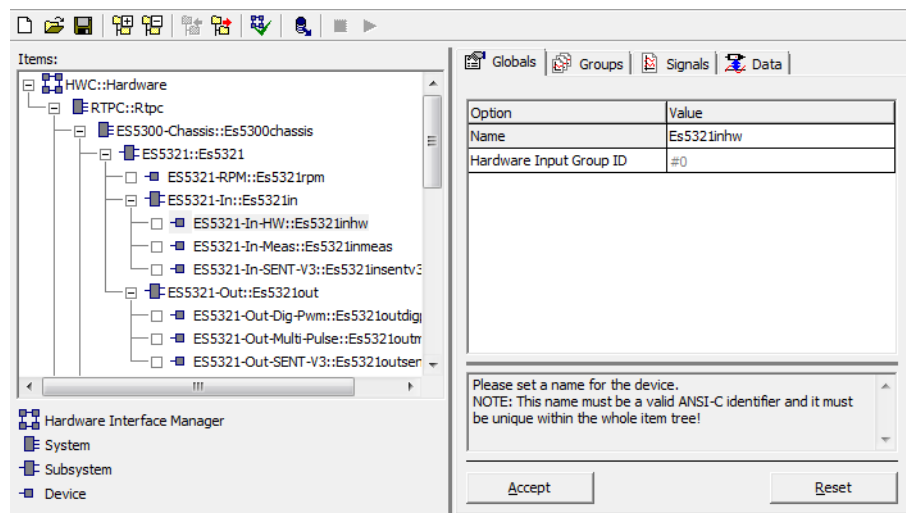
**Abb. 20-6** Die Registerkarte „Globals“ des ES5321-In Subsystems

## 20.4 ES5321-In-HW Device

### 20.4.1 Globals (ES5321-In-HW Device)

Das ES5321-In-HW Device dient zur Aktivierung der Kanäle und der Einstellung der Schaltschwellen der Eingangskomparatoren.

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



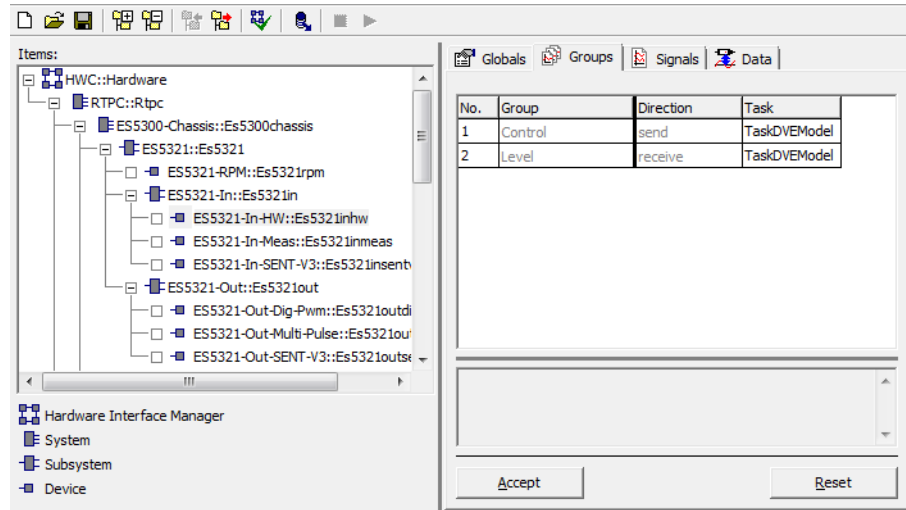
**Abb. 20-7** Die Registerkarte „Globals“ des ES5321-In-HW Device

*Hardware Input Group*

Die (von den anderen drei Gruppen galvanisch getrennte) Gruppe von Eingängen (0...3).

20.4.2 Groups (ES5321-In-HW Device)

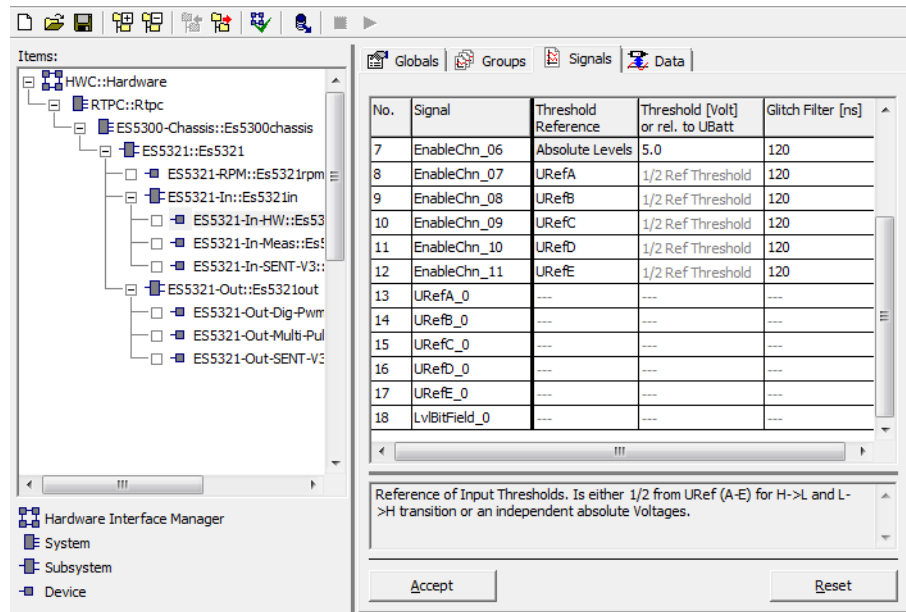
Das ES5321-In-HW Device besitzt eine Send-Signalgruppe „Control“ zur Steuerung des Boards und eine Receive-Signalgruppe „Level“ zum Einlesen der Eingangswerte.



**Abb. 20-8** Die Registerkarte „Groups“ des ES5321-In-HW Device

20.4.3 Signals (ES5321-In-HW Device)

Die folgende Abbildung zeigt die Signale des ES5321-In-HW Device.



**Abb. 20-9** Die Registerkarte „Signals“ des ES5321-In-HW Device

Signalname	Datentyp	Anmerkungen
EnableChn_xx	bool	0: Eingang wird nicht verwendet 1: Eingang wird verwendet (Default)
URefA...URefE	real32	Referenzspannungen [0...10 V]
LvlBitField_0	uint32	Bitfeld, das die Pegelinformationen der Eingänge enthält

**Tab. 20-4** Die Signale des ES5321-In-HW Device

Für die Signale gibt es folgende Parameter:

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Threshold Reference	uint8	Ja	Referenzspannung für Schwellwerte. Entweder eine der Batteriespannungen oder Absolutwert.
Threshold or relative to UBatt	real32	Ja	Schwellwerte (1/2 Uref oder Spannung)
Glitch Filter	uint16	Ja	Werte kleiner 120 werden auf 0.0 abgebildet. Werte zwischen 120 und 5000 werden auf den nächsthöheren, durch 8 teilbaren Wert aufgerundet Default: 120 ns (Auflösung 8 ns)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

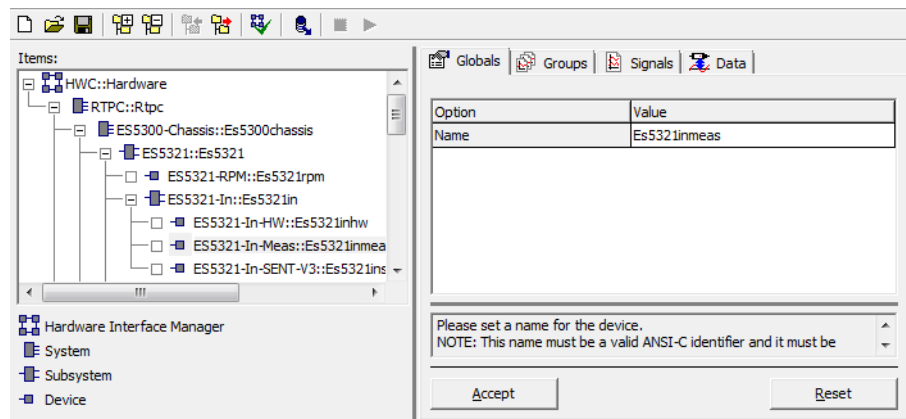
**Tab. 20-5** ES5321-In-HW Device: Konfigurationsparameter der Registerkarte „Signals“

## 20.5 ES5321-In-Meas Device

### 20.5.1 Globals (ES5321-In-Meas Device)

Das ES5321-In-Meas Device dient zur Konfiguration der Messfunktionen auf den 24 Eingangskanälen.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.



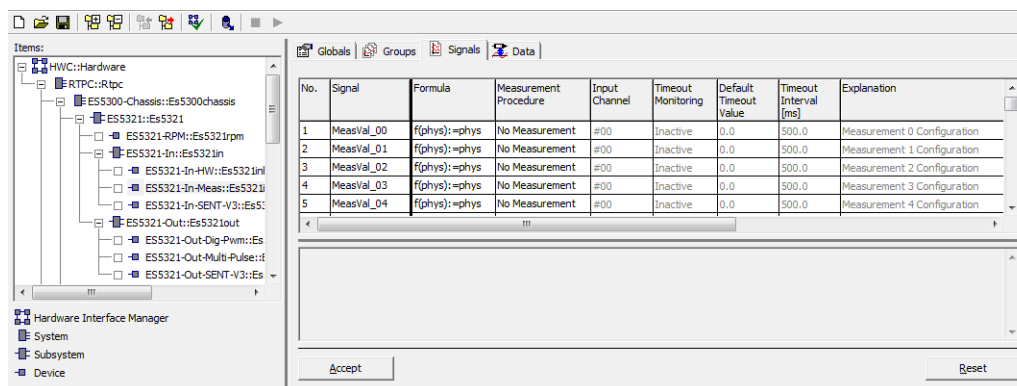
**Abb. 20-10** Die Registerkarte „Globals“ des ES5321-In-Meas Device

### 20.5.2 Signals (ES5321-In-Meas Device)

In der Registerkarte „Signals“ wird die Definition der 192 Messungen (bei acht ES5321-In-Meas Devices) eines ES5321-In-Meas Devices durchgeführt, die in Blöcken von je 24 Kanälen unterteilt werden.

Alle Optionsfelder online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.

Die folgende Abbildung zeigt die Signale der Registerkarte „Signals“.



**Abb. 20-11** Die Registerkarte „Signals“ des ES5321-In-Meas Device

Signalname	Datentyp	Anmerkungen
MeasVal_n (n = 0..23)	real32	Die 24 Messungen
Tout_n (n = 0..23)	real32	Konfiguration der Timeouts 0: kein Timeout 1: Timeout 2: nicht verwendet

**Tab. 20-6** Die Signale des ES5321-In-Meas Device

Für diese Signale gibt es folgende Konfigurationsparameter:

#### *Measurement Procedure*

In diesem Listenfeld wird das Messverfahren eingestellt. Im Abschnitt „Die Messverfahren“ auf Seite 644 finden Sie eine detaillierte Beschreibung und Konfigurationsrichtlinien zu den einzelnen Verfahren.

#### *Input Channel*

In diesem Listenfeld wird der Eingangskanal der ES5321.1 eingestellt, auf dem die Messung durchgeführt werden soll.

#### *Timeout Monitoring*

Definition der Timeout-Überwachung für die zugehörige Messung. Die folgenden Einstellungen sind möglich:

- „Inactive“: Keine Timeout-Überwachung.
- „Intvl Predef“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist der im Optionsfeld „Default Timeout Value“ eingestellte Wert.
- „Intvl InpDep“: Timeout-Überwachung in den im Optionsfeld „Timeout Interval [ms]“ definierten Zeitabständen. Der Messwert im Timeout-Fall ist vom Pegel des Eingangssignals abhängig.

#### *Default Timeout Value*

Der in diesem numerischen Eingabefeld eingestellte Wert wird als Messwert im Timeout-Fall ausgegeben, falls der „Intvl Predef“ Timeout-Überwachungsmodus eingestellt ist.

Die Timeout-Bedingung ist erfüllt, wenn einer der vier ausgewerteten Zeitstempel außerhalb der eingestellten Timeout-Wertes liegt.

#### *Timeout Interval [ms]*

Intervall für Überwachung der Zeitüberschreitung. Dieser Wert ist nur in den Timeout-Betriebsarten „Intvl Predef“ und „Intvl InpDef“ editierbar.

- Defaultwert: 500.0 [ms]
- Einstellbereich: 0.1 ... 10000.0 [ms]

### Messmethoden und Timeout-Behandlung

Wenn bei „Timeout Monitoring“ die Option „Intvl InpDep“ gewählt wurde, so werden – je nach eingestelltem Messverfahren – die folgenden Werte ausgegeben.

Messmethode	Wert (Input Level Dependent)
No Measurement	-
High Time [ $\mu$ s]	0
Low Time [ $\mu$ s]	0
Additive High Time [ $\mu$ s] (free-running)	0
Additive Low Time [ $\mu$ s] (free-running)	0
Additive High Time [ $\mu$ s] (delta)	0
Additive Low Time [ $\mu$ s] (delta)	0
Cycle Time --/-- [ $\mu$ s] Cycle Time --\-- [ $\mu$ s]	Die Zeit, die seit dem letztem gültigen Zeitstempel vergangen ist
Frequency --/-- [Hz] Frequency --\-- [Hz]	Kehrwert der Zeit, die seit dem letztem gültigen Zeitstempel vergangen ist
Duty Cycle L/(L+H) --/-- Duty Cycle L/(L+H) --\--	1: Eingangspegel „Low“ 0: Eingangspegel „High“
Duty Cycle H/(L+H) --/-- Duty Cycle H/(L+H) --\--	1: Eingangspegel „High“ 0: Eingangspegel „Low“
Level (Active High)	0
Level (Active Low)	0

## 20.6 Die Messverfahren

---

### Begriffsdefinitionen

---

#### Aktive Signalflanke

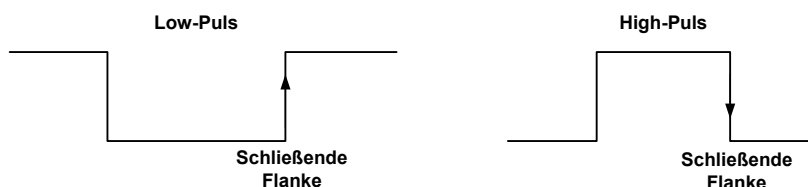
Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Flanke:

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### Öffnende Flanke

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



### Das Grundprinzip der Messwertberechnung

---

Die Messdaten (Zeitstempel von detektierten Flanken) werden per DMA (Direct Memory Access) vom ES5321.1 PWM I/O Board zum Real-Time PC übertragen und dort ausgewertet.

Die Auswertung erfolgt im RTIO-Treiber, der dann den Messwert und den Timeoutwert an die GUI weiterleitet.

Die Messungen besitzen eine Auflösung von 8 ns.

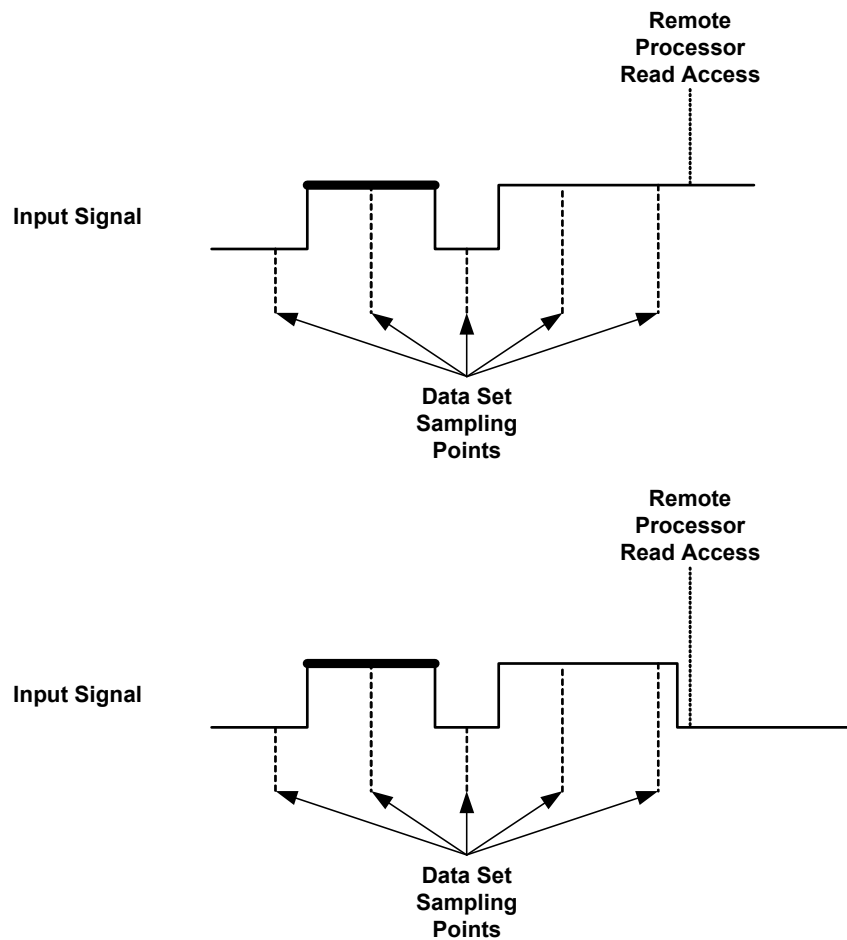


### 20.6.1 Pulsweitenmessungen

Mit den Messverfahren

- High Time [ $\mu\text{s}$ ]
- Low Time [ $\mu\text{s}$ ]

werden die High- und Low-Zeiten der Pulse eines PWM-Eingangs vermessen. Es wird die Pulsweite des dem letzten Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehenden Pulses gemessen. Abb. 20-12 zeigt dies beispielhaft für eine High-Zeit-Messung, der gemessene Puls ist mit fetter Strichstärke hervorgehoben.



**Abb. 20-12** High-Zeit-Messung (der gemessene Puls ist mit fetter Strichstärke hervorgehoben)

Eine Timeout-Bedingung liegt dann vor, wenn mindestens eine der vier Flanken älter ist als der aktuelle Zeitstempel abzüglich der Timeout-Zeit.

Die Timeout-Behandlung ist bei allen Messverfahren gleich.

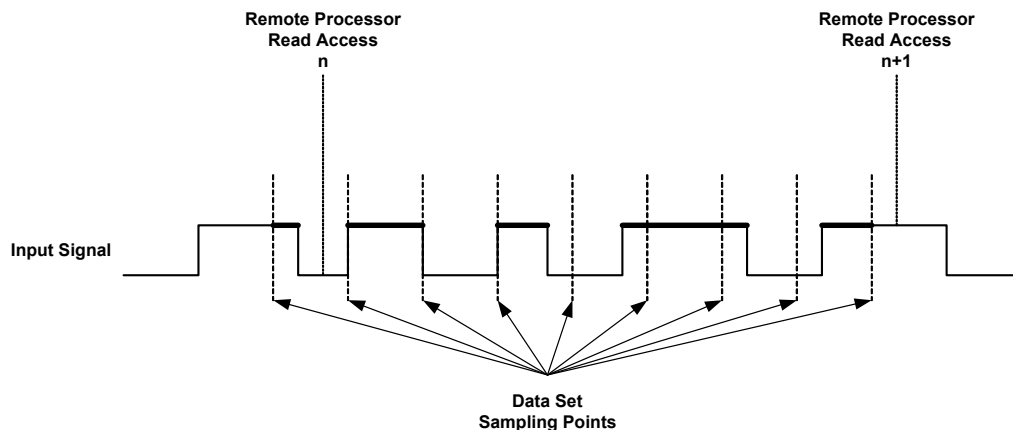
## 20.6.2 Additive Pulsweitenmessungen

Wird der Leseprozess für die „MeasVal“-Signalgruppe wiederholt (im Spezialfall periodisch) aktiviert, so liefern die Messverfahren

- Additive High Time [ $\mu\text{s}$ ]
- Additive Low Time [ $\mu\text{s}$ ]

die Zeit, in der das zu vermessende Signal zwischen zwei aufeinanderfolgenden Aktivierungen des Leseprozesses einen aktiven Pegel angenommen hat.

Abb. 20-13 veranschaulicht die Messwertberechnung am Beispiel einer additiven High-Zeit-Messung. Die Summe der fett markierten Liniensegmente ist die additive High-Zeit, die bei der (n+1)-ten Aktivierung des Leseprozesses geliefert wird.



**Abb. 20-13** Additive High-Zeit-Messung: Illustration der Messwertberechnung  
Eine Timeout-Bedingung liegt dann vor, wenn mindestens eine der vier Flanken älter ist als der aktuelle Zeitstempel abzüglich der Timeout-Zeit.

### „Free running“ und „delta“

Die beiden Messmethoden „Additive High Time“ und „Additive Low Time“ gibt es in zwei Varianten:

- Free running  
Diese Methode liefert die „Additive High/Low Time“ seit Beginn der Simulation. Die Zeit wird in „Clock Ticks“ zu je 8 ns gemessen und als 64-Bit Integer gespeichert.
- Delta  
Diese Methode liefert die „Additive High/Low Time“ zwischen zwei Aufrufen der Task, die die Messfunktion ausführt.

#### **Hinweis**

Da zur Messung der additiven Zeiten Hardware-intern ein 32-Bit Zähler verwendet wird, darf die Zeit zwischen zwei Aufrufen der Messfunktion nicht mehr als  $34,38 \text{ s}$  ( $2^{32} * 8 \text{ ns}$ ) betragen. Andernfalls kann nicht garantiert werden, dass die gemessenen Daten („delta“ und „free-running“) korrekt sind!

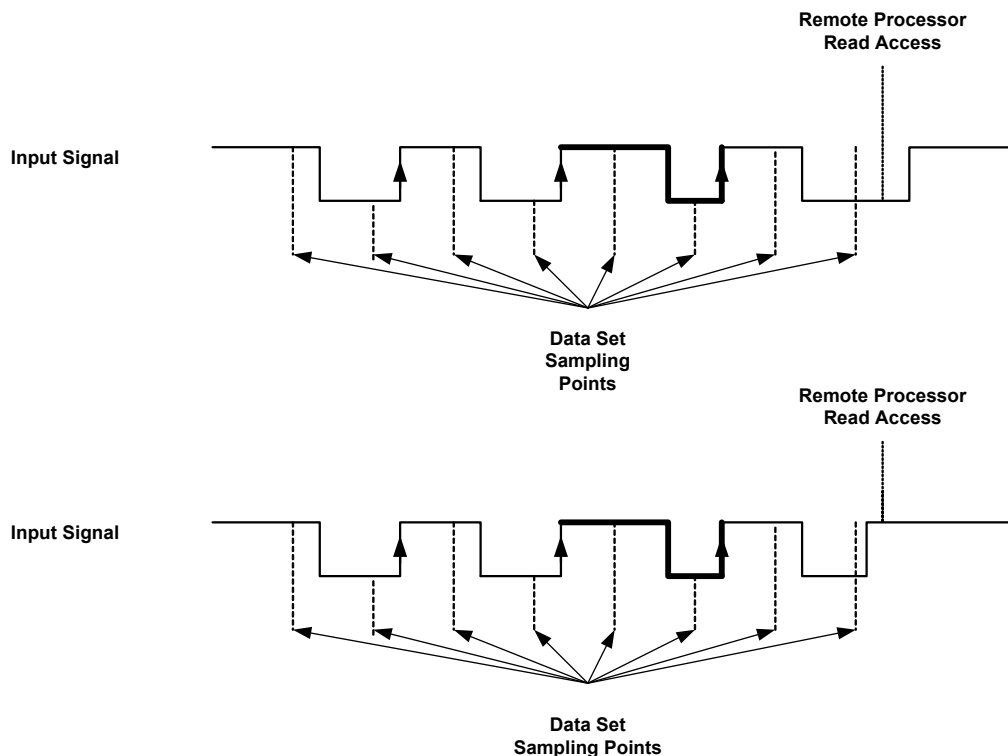
### 20.6.3 Frequenz- und Zykluszeitmessungen

Die Messverfahren

- Cycle Time [ $\mu\text{s}$ ]
- Frequency [Hz]

liefern die Zykluszeit bzw. die Frequenz des Signals an einem PWM-Eingang gemessen an steigenden oder fallenden Flanken. Es wird jeweils die dem letzten<sup>1</sup> Datenerfassungszeitpunkt zeitlich unmittelbar vorausgehende aktive Signalflanke mit zugehöriger Periode ausgewertet.

Abb. 20-14 zeigt die ausgewertete Periode beispielhaft für eine Frequenz- oder Zykluszeitmessung an ansteigenden Flanken.



**Abb. 20-14** Ausgewertete Periode (fett gezeichnet) bei einer Frequenz- oder Zykluszeitmessung an ansteigenden Flanken

### 20.6.4 Tastverhältnismessungen

Beim ES5321.1 PWM I/O Board stehen die nachfolgenden Tastverhältnismessungen zur Verfügung:

- Duty Cycle  $L/(L+H)$
- Duty Cycle  $H/(L+H)$

Bei diesen Messungen wird eine Pulsdauer ins Verhältnis zur Zykluszeit des Signals gesetzt. Der Anwender legt die Flanke, an der der Messwert berechnet wird, mit der Auswahl des Messverfahrens fest.

<sup>1</sup>. Der dem Lesezugriff des RTIO-Treibers auf den Datensatz unmittelbar vorausgehende Datenerfassungszeitpunkt

Was die Auswahl der Periode des Eingangssignals anbelangt, auf Basis derer die Tastverhältnisberechnung durchgeführt wird, so gilt exakt das, was für Frequenz- und Zykluszeitmessungen in Abschnitt 20.6.3 auf Seite 647 beschrieben wurde.

20.6.5 Pegelmessungen

Die Messverfahren

- Level (Active High)
- Level (Active Low)

liefern den Pegel eines PWM-Eingangs in Form einer aktiv-/inaktiv-Information. „0“ bedeutet das Signal ist inaktiv, „1“ bedeutet das Signal ist aktiv.

20.7 ES5321-In-SENT Device

20.7.1 Globals (ES5321-In-SENT Device)

Dieses RTIO-Element ermöglicht den Empfang eines Signals gemäß SENT-Datenprotokoll. Es können bis zu vier ES5321-In-SENT Devices eingefügt werden.

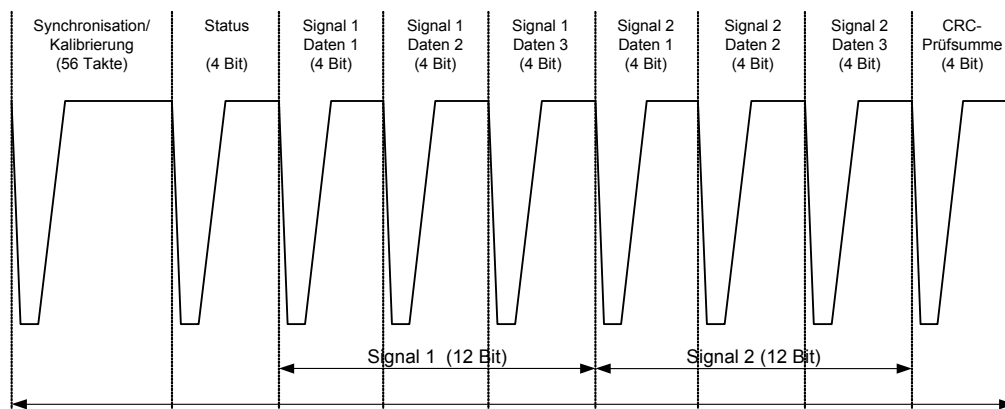


Abb. 20-15 Format eines Signals gemäß SENT-Datenprotokoll

Serial Messages

Bit 2 und Bit 3 des Status-Nibble stehen für den sogenannten Serial Message Channel (SMC) zur Verfügung.

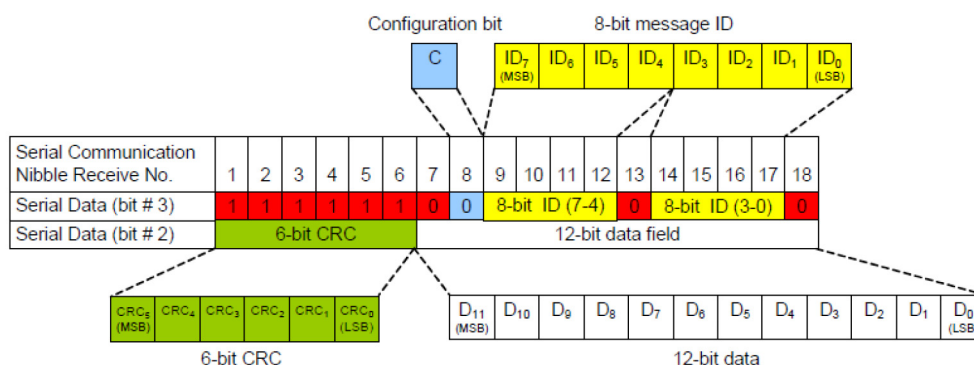
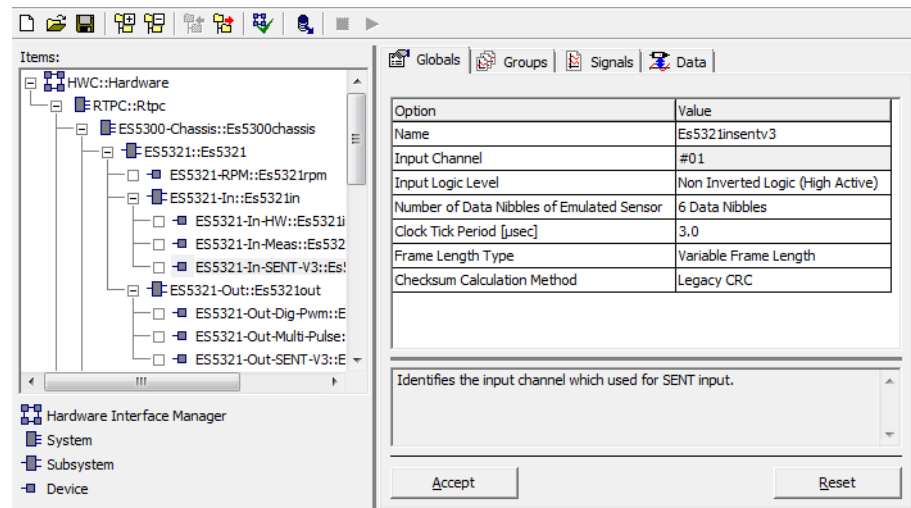


Abb. 20-16 Enhanced Serial Message Format mit 12-Bit Data Field und 8-Bit Message ID

Dabei werden über 16 oder 18 Frames die Daten der Bits 2 und 3 des Status-Nibble aufgesammelt und im Speicher der ES5321.1 zwischengespeichert.

Die Auswertung der Daten wird in einem C-Code-Modul in LABCAR-RTPC durchgeführt. Das Modul ist als Open Source Code verfügbar und befindet sich im Ordner \MISC des LABCAR-OPERATOR Installationsmediums.

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES5321-In-SENT Devices.



**Abb. 20-17** Die Registerkarte „Globals“ des ES5321-In-SENT Devices

#### *Input Channel*

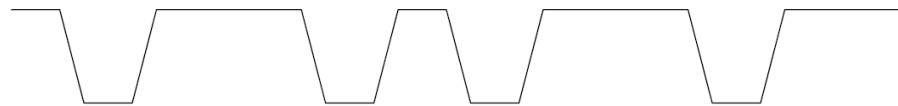
Jeder beliebige Eingabekanal kann verwendet werden – dabei werden die Pegel- bzw. Schwelleneinstellungen dieses Kanals im ES5321-In-HW Device verwendet. Dabei ist darauf zu achten, dass diese der Pegelspezifikation des SENT-Transmitters entsprechen (typisch: inaktiv = GND, aktiv = 5 Volt).

#### *Input Logic Level*

Diese Einstellung legt fest, ob die Pegel der SENT-Signale invertiert oder nicht invertiert ausgegeben werden.

- Non inverted Logic (High Active)

Abb. 20-18 zeigt den Signalverlauf einer nicht-invertierten Logik.



**Abb. 20-18** Nicht-invertierte Logik im Kontext der SENT-Spezifikation

- Inverted Logic (Low Active)

Abb. 20-19 zeigt den Signalverlauf einer invertierten Logik.



**Abb. 20-19** Invertierte Logik im Kontext der SENT-Spezifikation

#### *Number of Data Nibbles of Emulated Sensor*

---

Mit diesem Parameter lässt sich die Anzahl der Datennibbles des emulierten Sensors einstellen.

#### *Clock Tick Period [ $\mu$ sec]*

---

Länge (in  $\mu$ s) eines „Clock Ticks“ – daraus ergibt sich dann die Länge des „Calibration/Synchronization Pulse“ und der „Nibble Pulses“.

#### *Frame Length Type*

---

Die Übertragungsart der SENT-Frames.

- Variable Frame Length  
Ein SENT-Frame wird unmittelbar nach dem anderen erwartet. Es gibt keinen „Pause“ Puls.
- Constant Frame Length  
Zwischen zwei SENT-Frames wird ein „Pause“ Puls erwartet. Die Länge des „Pause“ Pulses ist so gewählt, dass der komplette SENT-Frame (inkl. „Pause“) eine konstante Länge hat.

#### *Checksum Calculation Method*

---

Die Berechnungsmethode für die 4-Bit-Prüfsumme in der ES5321.1.

- None  
Es findet keine Prüfsummenüberprüfung statt.
- Recommended CRC  
Überprüfung der Prüfsumme entsprechend der SENT Spezifikation Jan 2010.  
Sollte die berechnete Prüfsumme nicht mit der übertragenen übereinstimmen, wird der „ChecksumErrorCounter“ erhöht und der Datenwert nicht übernommen.
- Legacy CRC  
Überprüfung der Prüfsumme in der ES5321.1 entsprechend der SENT Spezifikation April 2007 und Feb 2008.  
Sollte die berechnete Prüfsumme nicht mit der übertragenen übereinstimmen, wird der „ChecksumErrorCounter“ erhöht und der Datenwert nicht übernommen.

### 20.7.2 Groups (ES5321-In-SENT Device)

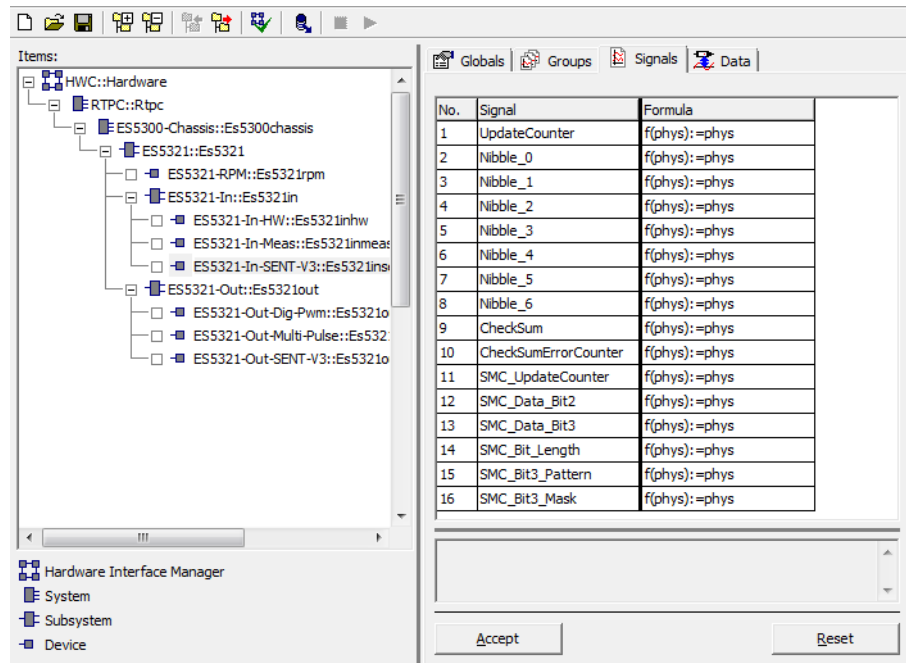
---

Das ES5321-In-SENT Device implementiert eine Receive-Signalgruppe „SentIn“. Diese liefert die uninterpretierten Daten des letzten, vollständig empfangenen SENT-Datenwortes und Statusinformationen.

Die Interpretation der Daten ist abhängig vom Protokolltyp des Senders.

## 20.7.3 Signals (ES5321-In-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES5321-In-SENT Devices.



**Abb. 20-20** Die Registerkarte „Signals“ des ES5321-In-SENT Devices

### Signale der Receive-Signalgruppe "SentIn":

#### *UpdateCounter*

Umlaufender Zähler, der inkrementiert wird, wenn neue Daten zur Verfügung stehen – der Wertebereich beträgt 0...255.

#### *Nibble\_0 ... Nibble\_6*

Werte der 7 Nibbles (1 Status-Nibble und 2x3 Daten-Nibbles) des zuletzt empfangenen SENT-Datenwortes – der gültige Wertebereich beträgt 0...15.

#### *Checksum*

Vom SENT-Frame übertragene Prüfsumme.

#### *ChecksumErrorCounter*

Umlaufender Zähler, der inkrementiert wird, wenn eine SENT-Botschaft in der ES5321.1 vollständig empfangen wurde, aber die Prüfsumme falsch war.

### Signale der Receive-Signalgruppe "SentInSmc":

#### *SMC\_UpdateCounter*

Umlaufender 8-Bit-Zähler, der inkrementiert wird, wenn die ES5321.1 eine SMC-Botschaft vollständig empfangen hat, welche nun an den Signalen „SMC\_Data\_Bit2“ und „SMC\_Data\_Bit3“ zur Auswertung zur Verfügung steht.

SMC\_Data\_Bit2

Datenwort, welches den empfangenen Bit 2-Datenstrom bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4). Die höheren Bitpositionen enthalten die später empfangenen Daten.

SMC\_Data\_Bit3

Datenwort, welches den empfangenen Bit 3-Datenstrom bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4). Die höheren Bitpositionen enthalten die später empfangenen Daten.

**Signale der Send-Signalgruppe "SentInSmcCtrl":**SMC\_Bit\_Length

Eingangssignal zur Steuerung der SMC-Empfangseinheit der ES5321.1. Der Wert „0“ deaktiviert den SMC-Empfang, ein Wert > 0 (... 31) aktiviert den SMC-Empfang und gibt die Anzahl der Bits für einen SMC-Zyklus vor.

Als gültige Werte entsprechend der SENT-Spezifikation sind 16 (Short Serial Message) bzw. 18 (Enhanced Serial Message) definiert.

SMC\_Bit3\_Pattern

Eingangssignal zur Steuerung der SMC Empfangseinheit der ES5321.1. Es gibt ein Bit-Muster zur Gültigkeitsprüfung einer SMC für Bit 3 vor.

Eine Botschaft wird als gültig erkannt wenn gilt:

```
(SMC_Data_Bit3 bitwiseXor SMC_Bit3_Pattern)
bitwiseAnd SMC_Bit3_Mask == 0.
```

D.h. „SMC\_Bit3\_Pattern“ definiert den Wert der relevanten Bits.

SMC\_Bit3\_Mask

Eingangssignal zur Steuerung der SMC Empfangseinheit der ES5321.1. Sie gibt eine Bit-Maske zur Gültigkeitsprüfung einer SMC für Bit 3 vor.

Eine Botschaft wird als gültig erkannt wenn gilt:

```
SMC_Data_Bit3 bitwiseXor SMC_Bit3_Pattern)
bitwiseAnd SMC_Bit3_Mask == 0
```

D.h. „SMC\_Bit3\_Mask“ markiert die relevanten Bitpositionen mit einer Eins.

Für den Empfang einer SMC Botschaft stellt die ES5321.1 und die RTIO Einbindung nur generische Funktionen zur Verfügung. Die Steuerung, Dekodierung der Daten und die Überprüfung der Prüfsumme erfolgt außerhalb der RTIO.

Ein C-Code-Modul „SentInSMC\_Control“ mit Grundfunktionen steht hierzu als Vorlage zur Verfügung. Alle SMC\_\* Signale der RTIO werden dazu mit den entsprechenden Ports des C-Code-Moduls verbunden.

Das C-Code-Modul muss jedoch noch an die spezifische Anwendung angepasst werden durch:

- Festlegung „SMC Format Mode“ entsprechend der SENT Spezifikation
- Umrechnung der Datenwerte der benötigten Msg-IDs in physikalische Größen
- Bereitstellen der Datenwerte an den Output-Ports



Folgende Werte sind gemäß J2716 Jan 2010 definiert:

- 16 Bit Short Serial Message (E\_SMC\_16Bit)
  - SMC\_Bit3\_Pattern = 0x8000;
  - SMC\_Bit3\_Mask = 0xFFFF;
  - SMC\_Bit\_Length = 16;
- 18 Bit Enhanced Serial Message (12 Bit Data, 8 Bit Msg ID, E\_SMC\_18BitC0):
  - SMC\_Bit3\_Pattern = 0x3F000;
  - SMC\_Bit3\_Mask = 0x3F421;
  - SMC\_Bit\_Length = 18;
- 18 Bit Enhanced Serial Message (16 Bit Data, 4 Bit Msg ID, E\_SMC\_18BitC1):
  - SMC\_Bit3\_Pattern = 0x3F400;
  - SMC\_Bit3\_Mask = 0x3F421;
  - SMC\_Bit\_Length = 18;

## 20.8 ES5321-Out Subsystem

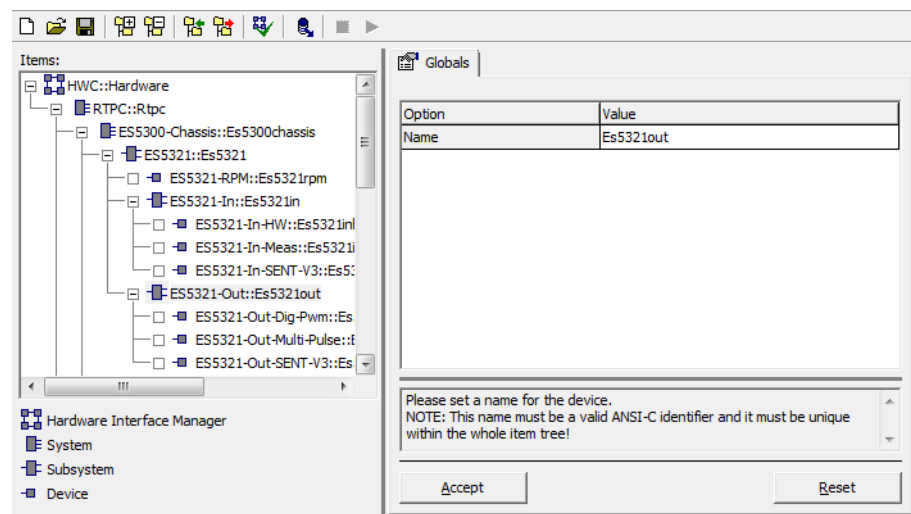
### 20.8.1 Globals (ES5321-Out Subsystem)

Das ES5321-Out Subsystem dient zur Konfiguration und Ansteuerung der 32 Ausgangskanäle, an denen digitale Signale, pulsweitenmodulierte Signale, Pulssequenzen und SENT-Signale ausgegeben werden können.

Außerdem kann hier definiert werden, welcher Level „Inactive“ sein soll und welcher „Active“.

Unter dieses Subsystem kann man vier ES5321-Out-Dig-Pwm, vier ES5321-Out-Multi-Pulse und 16 ES5321-Out-SENT Devices einfügen.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.



**Abb. 20-21** Die Registerkarte „Globals“ des ES5321-Out Subsystem

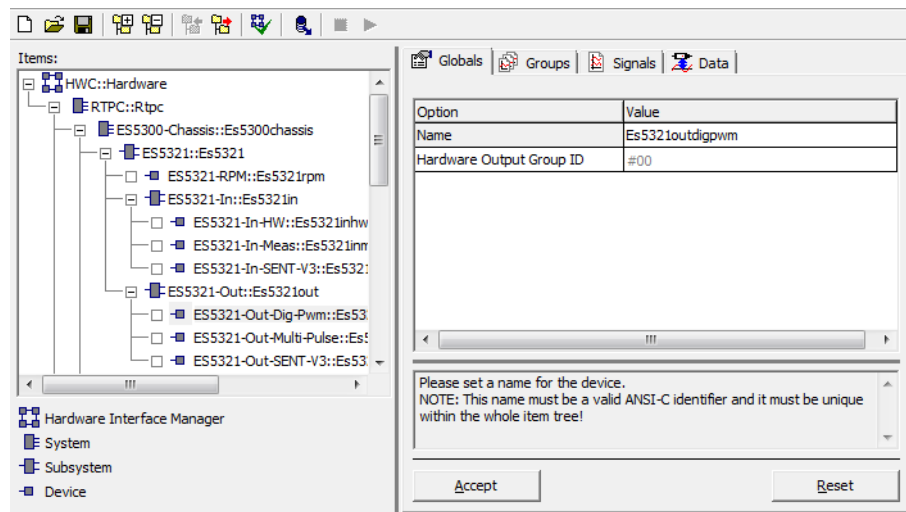
## 20.9 ES5321-Out-Dig-Pwm Device

In einer Hardwarekonfiguration können bis zu vier ES5321-Out-Dig-PWM Subsysteme eingefügt werden (je ein Subsystem für eine galvanisch getrennte Ausgangsgruppe).

Jedes dieser Items steuert acht Ausgangskanäle, so dass in Summe 32 Ausgänge konfiguriert werden können.

### 20.9.1 Globals (ES5321-Out-Dig-Pwm Device)

Das ES5321-Out-Dig-Pwm Device dient zur Ausgabe von PWM-Signalen. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.

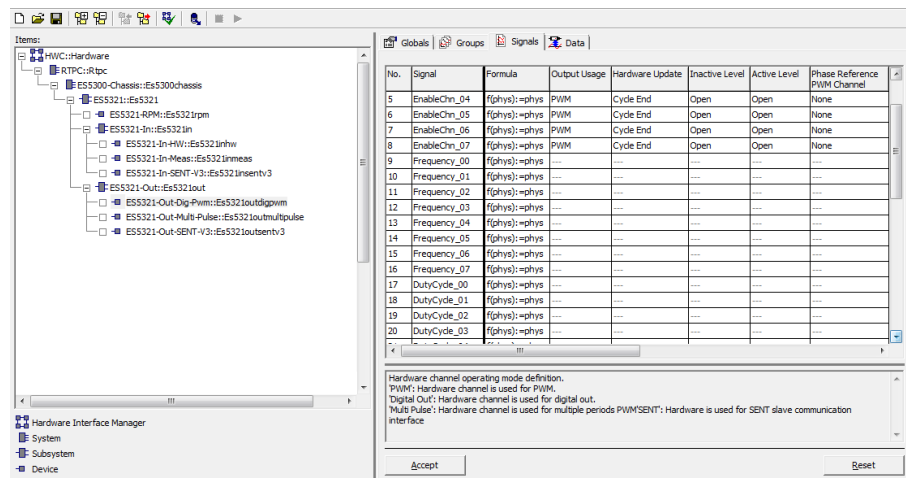


**Abb. 20-22** Die Registerkarte „Globals“ des ES5321-Out-Dig-Pwm Devices  
*Hardware Input Group*

Die (von den anderen galvanisch getrennte) Gruppe von Eingängen (0...3).

### 20.9.2 Signals (ES5321-Out-Dig-Pwm Device)

Abb. 20-23 zeigt die Registerkarte „Signals“ des ES5321-Out-Dig-Pwm Devices.



**Abb. 20-23** Die Registerkarte „Signals“ des ES5321-Out-Dig-Pwm Devices

**Signale der Send-Signalgruppe „PwmOut“:**

Signalname	Datentyp	Anmerkungen
EnableChn_xx	bool	0: Eingang wird nicht verwendet (Default) 1: Eingang wird verwendet Ohne Bedeutung, wenn eine Phasenbeziehung zu einem anderen Kanal hergestellt wurde (siehe „Phase Reference PWM Channel“ auf Seite 657).
Frequency_xx	real32	PWM-Frequenz [0.0 Hz...100000.0 Hz] Ohne Bedeutung, wenn eine Phasenbeziehung zu einem anderen Kanal hergestellt wurde - in diesem Fall wird die Frequenz des Masterkanals verwendet.
DutyCycle_xx	real32	Tastverhältnis [0.0...1.0]
Phase_xx	real32	Wenn eine Phasenbeziehung zu einem anderen Kanal besteht, kann hier der Wert der Phasenverschiebung eingestellt werden [-360.0°...+360 °]
Ground_Shift_GDN	bool	Aktiviert/deaktiviert einen Ground-Shift für GND
Ground_Shift_UBattA	bool	Aktiviert/deaktiviert einen Ground-Shift für UBatt A
Ground_Shift_UBattB	bool	Aktiviert/deaktiviert einen Ground-Shift für UBatt B

**Tab. 20-7** Die Signale des ES5321-Out-Dig-Pwm Device - Signalgruppe „PwmOut“

**Signale der Receive-Signalgruppe „ChnState“:**

Signalname	Datentyp	Anmerkungen
ChnState_xx	uint8	Jeder Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers: - Bit 0: Ausgangspegel (0 = inaktiv, 1 = aktiv) - Bit 1: Überstrom am Low-Side FET (1 = aktiv) - Bit 2: Überstrom am High-Side FET (1 = aktiv) - Bit 3: Fehler erkannt (1 = aktiv)

**Tab. 20-8** Die Signale des ES5321-Out-Dig-Pwm Device - Signalgruppe „ChnState“

Für die Signale können Sie folgende Optionen festlegen:

#### *Output Usage*

---

In dieser Ansicht kann zwischen PWM und Digital Out ausgewählt werden.

Ist bereits ein ES5321-Out-Multi-Pulse (oder ein ES5321-Out-SENT Device hinzugefügt worden, werden in dieser Ansicht die dafür verwendeten Ausgänge ausgegraut.

- PWM  
Erzeugung eines pulswertenmodulierten Signals – beschrieben durch Frequenz, Tastverhältnis und Phasenlage zu einem anderen PWM-Ausgang
- Digital Out  
Bei dieser Konfiguration des Ausgangskanals werden Signalpegel generiert.

#### *Hardware Update*

---

In diesem Listenfeld wird eingestellt, wann vom Modell oder vom Anwender an den „Frequency\_x“- und „DutyCycle\_x“-RTIO-Signalen (x = 0...7) vorgenommenen Wertänderungen von der ES5321.1-Hardware übernommen werden.

- Immediate  
Ein neuer Wert wird am Ausgang sofort sichtbar, d.h. der aktuelle Impuls wird abgebrochen und mit neuen Werten gestartet.
- Cycle End  
Ein neuer Wert wird erst nach Beendigung des kompletten Impulses am Ausgang sichtbar.

#### *Inactive Level*

---

In diesem Listenfeld wird der inaktive Zustand des PWM-Signals eingestellt. Der inaktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, so nimmt es während 40% der Periodendauer den inaktiven Signalzustand an.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

### Active Level

In diesem Listenfeld wird der aktive Zustand des PWM-Signals eingestellt.

Der aktive Zustand eines PWM-Signals ist folgendermaßen definiert: Ist für ein PWM-Signal ein Tastverhältnis von 60% eingestellt, nimmt es während 60% der Periodendauer den aktiven Signalzustand an.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

### **Hinweis**

*Die gleichzeitige Verwendung von Batteriespannung A und Batteriespannung B als Active/Inaktive Level ist nicht zulässig!*

### Phase Reference PWM Channel

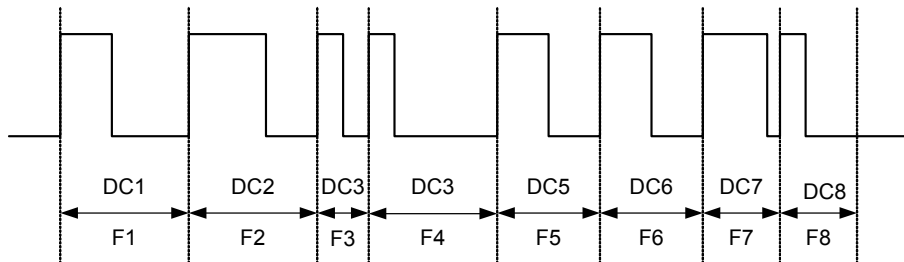
Die ES5321.1 bietet die Möglichkeit, Phasenverschiebungen zwischen Ausgabekanälen mit gleicher PWM-Frequenz definiert einzustellen. Hierzu ist einer dieser PWM-Ausgänge als Phasen-Referenz-Kanal zu bestimmen.

Im Listenfeld „Phase Reference PWM Channel“ der übrigen, zu diesem Referenz-Kanal in definierten Phasenbeziehungen stehenden, Ausgänge ist dann die Nummer des Referenz-Kanals einzutragen.

## 20.10 ES5321-Out-Multi-Pulse Device

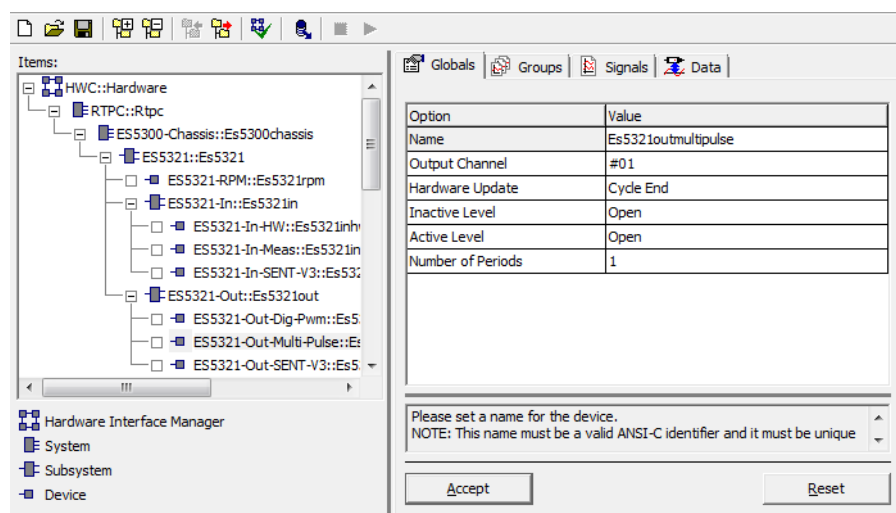
### 20.10.1 Globals (ES5321-Out-Multi-Pulse Device)

Eine Multi-Pulse-Sequenz kann aus bis zu acht Pulsen bestehen, von denen jeder durch eine Frequenz und ein Tastverhältnis definiert ist.



**Abb. 20-24** Eine Abfolge von acht Pulsen ( $F_x$  = Frequenz von Puls  $x$ ,  $DC_x$  = Tastverhältnis von Puls  $x$ )

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES5321-Multi-Pulse Devices.



**Abb. 20-25** Die Registerkarte „Globals“ des ES5321-Multi-Pulse Devices

Die Bedeutung der verfügbaren Optionen wird im Folgenden beschrieben.

#### *Output Channel*

Der Hardware-Kanal der ES5321.1, der als Multi-Pulse-Ausgang verwendet wird.

#### *Hardware Update*

In diesem Listenfeld wird eingestellt, wann vom Modell oder vom Anwender am Pulspaket vorgenommenen Änderungen von der ES5321.1-Hardware übernommen werden.

- Immediate

Ein neuer Wert wird am Ausgang sofort sichtbar, d.h. der aktuelle Impuls wird abgebrochen und mit neuen Werten gestartet.

- Cycle End  
Ein neuer Wert wird nach Beendigung des kompletten Impulses am Ausgang sichtbar.
- RTIO Controlled  
Ein neuer Wert wird nach „0 → 1“ Übergang des „SyncSgl“-Signals am Ausgang sichtbar.

#### *Inactive Level*

---

In diesem Listenfeld wird der inaktive Zustand des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

#### *Active Level*

---

In diesem Listenfeld wird der aktive Zustand des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Der PWM-Ausgang ist hochohmig und kann über einen externen Pull-Up- oder Pull-Down-Widerstand auf einen definierten Spannungspegel eingestellt werden.
- Ground  
Der PWM-Ausgang liegt auf Masse.
- V Batt A Out  
Der PWM-Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der PWM-Ausgang gibt Batteriespannung B aus.

#### **Hinweis**

*Die gleichzeitige Verwendung von Batteriespannung A und Batteriespannung B als Active/Inaktive Level ist nicht zulässig!*

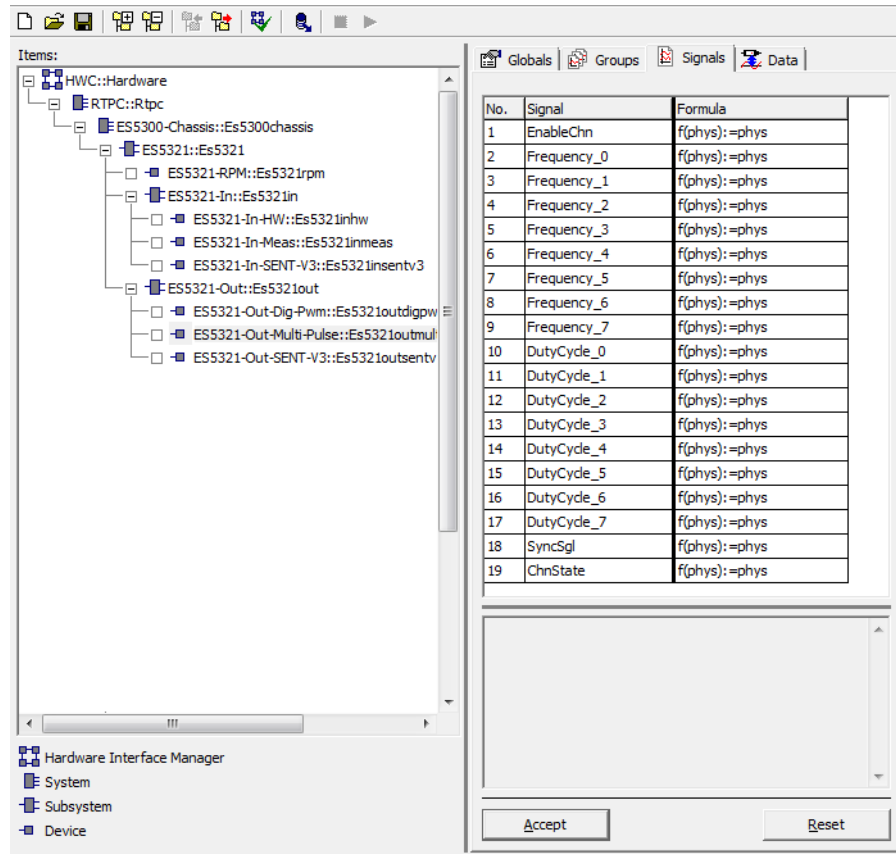
#### *Number of Periods*

---

Anzahl der Pulse der Sequenz (1..8)

20.10.2 Signals (ES5321-Out-Multi-Pulse Device)

Die folgende Abbildung zeigt die Signale der Registerkarte „Signals“



**Abb. 20-26** Die Registerkarte „Signals“ des ES5321-Multi-Pulse Devices

Die Registerkarte „Signals“ enthält folgende Signale:

**Signale der Send-Signalgruppe „MultiPulseOut“ :**

Signalname	Datentyp	Anmerkungen
EnableChn	bool	0: Eingang wird nicht verwendet (Default) 1: Eingang wird verwendet
Frequency_xx	real32	PWM-Frequenz in Hz [0.01 Hz... 100000.0 Hz]
DutyCycle_xx	real32	Tastverhältnis [0.0... 1.0]
SyncSgl	bool	Ist bei der Option „Hardware Update“ (siehe „Hardware Update“ auf Seite 658) die Einstellung „RTIO Controlled“ gewählt, so werden die an diesen Kanal übermittelten Puls-paket-Daten erst dann von der Hardware übernommen, nachdem auf dem „SyncSgl“-Signal ein Übergang von 0 nach 1 detektiert wurde.

**Tab. 20-9** Die Signale des ES5321-Out-Dig-Pwm Device - Signalgruppe „PwmOut“



**Signale der Receive-Signalgruppe „ChnState“ :**

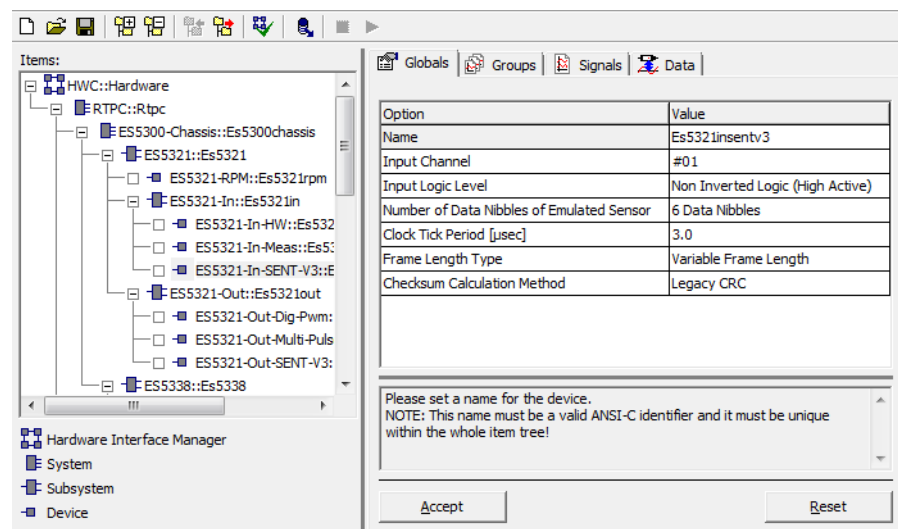
Signalname	Datentyp	Anmerkungen
ChnState	uint8	Jeder Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers: - Bit 0: Ausgangspegel (0 = inaktiv, 1 = aktiv) - Bit 1: Überstrom am Low-Side FET (1 = aktiv) - Bit 2: Überstrom am High-Side FET (1 = aktiv) - Bit 3: Fehler erkannt (1 = aktiv)

**Tab. 20-10** Die Signale des ES5321-Multi-Pulse Device - Signalgruppe „ChnState“

## 20.11 ES5321-Out-SENT Device

## 20.11.1 Globals (ES5321-Out-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES5321-Out-SENT Devices.

**Abb. 20-27** Die Registerkarte „Globals“ des ES5321-Out-SENT Devices

In dieser Registerkarte werden folgende Optionen eingestellt:

*Output Channel*

Auswahl der Hardwarekanals, der als SENT-Transmitter verwendet werden soll. Es können maximal 16 ES5321-Out-SENT Devices konfiguriert werden.

*Inactive Level*

In diesem Listenfeld wird der inaktive Zustand (logische Null) des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Ausgang ist hochohmig

- Ground  
Ausgang liegt auf Massepotential
- V Batt A Out  
Der Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der Ausgang gibt Batteriespannung B aus.

**Hinweis**

*Wenn ein nicht-invertierter Signalverlauf ausgegeben werden soll, dann sollte für den inaktiven Level „Ground“ eingetragen sein.*

Active Level

In diesem Listenfeld wird der aktive Zustand (logische Eins) des Signals eingestellt.

Zur Auswahl stehen die folgenden Optionen:

- Open  
Ausgang ist hochohmig
- Ground  
Ausgang liegt auf Massepotential
- V Batt A Out  
Der Ausgang gibt Batteriespannung A aus.
- V Batt B Out  
Der Ausgang gibt Batteriespannung B aus.

**Hinweis**

*Wenn ein nicht-invertierter Signalverlauf ausgegeben werden soll, dann sollte für den aktiven Level „V Batt A“ oder „V Batt B“ eingetragen sein.*

**Hinweis**

*Die gleichzeitige Verwendung von Batteriespannung A und Batteriespannung B als Active/Inaktive Level ist nicht zulässig!*

Protocol Type of Emulated Sensor

Auswahl des verwendeten Protokolls nach der SENT-Spezifikation. Es stehen folgende Auswahlmöglichkeiten zur Verfügung:

- Without Rolling Counter  
Die Werte von „Nibble\_4“ und „Nibble\_5“ (wie in der Registerkarte „Data“ angezeigt) werden berücksichtigt.
- With Rolling Counter Nibble 4/5  
Ein 8-Bit-Zählerwert, der mit den Nibble 4 und 5 übertragen wird. Die Werte von „Nibble\_4“ und „Nibble\_5“ (wie in der Registerkarte „Data“ angezeigt) werden nicht berücksichtigt.
- *n* Data Nibbles  
Anzahl der im Protokoll definierten Daten-Nibbles

### *Clock Tick Period [µsec]*

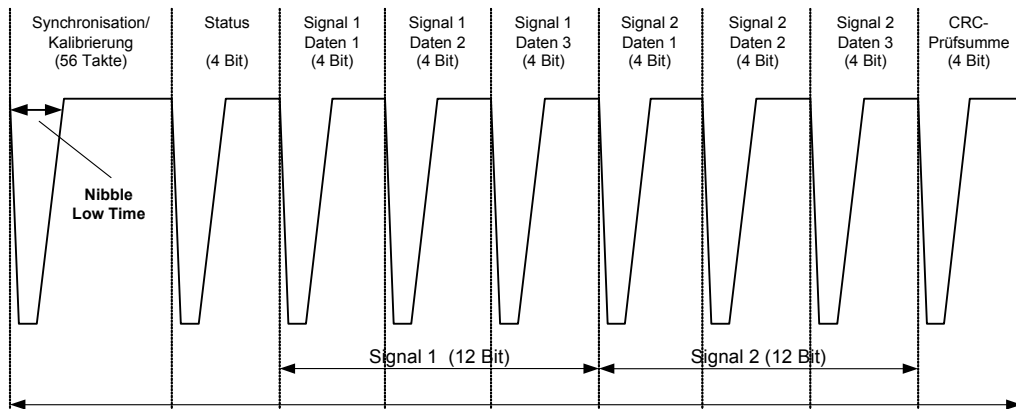
---

Länge des „Clock Tick“, der kleinsten Zeiteinheit des SENT-Protokolls (1...100 µs, Auflösung 8 ns). Die kleinstmögliche Länge beim ES5321-Out-SENT Device beträgt 1,504 µs.

### *Nibble Low Pulse [Ticks]*

---

Anzahl der Ticks für den Low Pulse eines Nibbles – gemäß SENT-Spezifikation Jan 2010 sollte der Wert größer als 4 sein.



### *Frame Length Type*

---

Der Übertragungsmodus der SENT Frames.

- Variable Frame Length  
Ein SENT Frame wird unmittelbar nach dem anderen erwartet. Es gibt keinen „Pause“ Puls.
- Constant Frame Length  
Zwischen zwei SENT Frames wird ein „Pause“ Puls erwartet. Die Länge des „Pause“ Pulses ist gewählt, das das komplette SENT Frame (inkl. „Pause“ Puls) eine konstante Länge hat.

### *Total Frame Length [Ticks]*

---

Wenn für der Einstellung „Frame Length Type“ die Auswahl „Constant Frame Length“ gewählt wird, dann kann die Länge des Frames eingestellt werden.

### *Checksum Calculation Method*

---

Mit dieser Einstellung wird die Berechnungsmethode der 4-Bit Prüfsumme festgelegt.

- Legacy CRC (0) \*  
Berechnung entsprechend der SENT Spezifikation April 2007 und Feb 2008
- Recommended CRC (1) \*  
Berechnung entsprechend der SENT Spezifikation Jan 2010

- CRC = 0x0 (2) \*  
CRC Nibble hat immer den Wert „0x0“.
- CRC = 0xF (3) \*  
CRC Nibble hat immer den Wert „0xF“.
- CRC Mode 0...3 by Signal  
Mit dem Signal „Checksum\_Mode\_Data“ (siehe „Checksum\_Mode\_Data“ auf Seite 665) kann eine der Berechnungsmethoden 0...3 ausgewählt werden (jeder andere Wert führt zu CRC = 0).
- CRC Data 0...15 by Signal  
Das CRC Nibble kann mit dem Signal „Checksum\_Mode\_Data“ direkt vorgegeben werden.

### Hinweis

\* Die Werte in Klammern geben die jeweilige Modenummer des Eingangssignals für die Einstellung "CRC Mode 0...3 by Signal" an.

#### 20.11.2 Signals (ES5321-Out-SENT Device)

Die folgende Abbildung zeigt die Registerkarte „Signals“ des ES5321-Out-SENT Devices.

No.	Signal	Formula
1	EnableChn	f(phys)=phys
2	Nibble_0	f(phys)=phys
3	Nibble_1	f(phys)=phys
4	Nibble_2	f(phys)=phys
5	Nibble_3	f(phys)=phys
6	Nibble_4	f(phys)=phys
7	Nibble_5	f(phys)=phys
8	Nibble_6	f(phys)=phys
9	ChnState	f(phys)=phys
10	SMC_Bit_Length	f(phys)=phys
11	SMC_Data_Bit2	f(phys)=phys
12	SMC_Data_Bit3	f(phys)=phys
13	SMC_UpdateCounter	f(phys)=phys
14	Checksum_Mode_Data	f(phys)=phys

**Abb. 20-28** Die Registerkarte „Signals“ des ES5321-Out-SENT Devices

Die Registerkarte „Signals“ enthält folgende Signale:

### **Signale der Send-Signalgruppe „SENTOut“:**

#### *EnableChn*

---

Aktiviert oder deaktiviert den für die SENT-Ausgabe verwendeten Hardwarekanal.

- = 0: Ausgang ist hochohmig, Signalausgabe ist inaktiv
- = 1: Signalausgabe ist aktiv

#### *Nibble\_0..Nibble\_6*

---

Der Wert der einzelnen Nibbles.

#### *Checksum\_Mode\_Data*

---

Wenn für „Checksum Calculation Method“ (siehe Seite 663) „CRC Mode 0...3 by Signal“ gewählt wurde, kann hier eine der Berechnungsmethoden 0...3 festgelegt werden.

Wenn für „Checksum Calculation Method“ „CRC Data 0...15 by Signal“ gewählt wurde, kann hier das CRC Nibble direkt vorgegeben werden.

### **Signale der Send-Signalgruppe „ChnState“:**

#### *ChnState*

---

4-Bit Fehlerzustandskodierung des verwendeten Hardwarekanals. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers.

- Bit 0: Ausgangspegel (0 = inaktiv, 1 = aktiv)
- Bit 1: Überstromerkennung am Low-Side FET (1 = aktiv)
- Bit 2: Überstromerkennung am High-Side FET (1 = aktiv)
- Bit 3: Fehler erkannt (1 = aktiv)

### **Signale der Send-Signalgruppe „SentOutSmc“:**

#### *SMC\_Bit\_Length*

---

Eingangssignal zur Steuerung der SMC Sendeeinheit der ES5321.1.

Der Wert „0“ deaktiviert SMC-Senden, d.h. es werden Bit 2 und Bit 3 des Nibble 0 übertragen. Ein Wert > 0 (... 31) aktiviert die Einblendung der Bits 2 und 3 aus den jeweiligen Datenwörtern. Der Wert gibt die Anzahl der Bits für einen SMC-Zyklus vor. Als gültige Werte sind 16 (Short Serial Message) bzw. 18 (Enhanced Serial Message) definiert.

#### *SMC\_Data\_Bit2*

---

Eingangssignal, welches das Datenwort für Bit 2 bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4).

#### *SMC\_Data\_Bit3*

---

Eingangssignal, welches das Datenwort für Bit 3 bereithält (siehe SAE J2716 Version Jan 2010, Kap.5.2.4).

**Signale der Receive-Signalgruppe „SentOutSmcState“:***SMC\_UpdateCounter*

---

Umlaufender 8 Bit Zähler – wird inkrementiert, wenn eine SMC-Botschaft vollständig übertragen wurde. Außerdem wird signalisiert, dass die ES5321.1 die Ausgabe einer neuen SMC-Botschaft mit den aktuellen Werten von SMC Data\_Bit2 und SMC\_Data\_Bit3 gestartet hat. Der Zähler kann somit zur Synchronisation bei zyklischer Ausgabe einer Liste von Message IDs verwendet werden.

Es werden die Werte von SMC\_Data\_Bit2 und SMC\_Data\_Bit3 an dem Zeitpunkt übernommen, an dem eine neue Short Serial Message ausgegeben wird. Nach dieser Datenübernahme sollten die Werte SMC\_Data\_Bit2 und SMC\_Data\_Bit3 mit neuen Daten gefüllt werden. Dies wird dem Rechenmodell dadurch signalisiert, indem der Wert von „SMC\_UpdateCounter“ inkrementiert wird.

Für das Senden einer SMC Botschaft stellt die ES5321.1 und die RTIO-Einbindung nur generische Funktionen zur Verfügung. Die Steuerung, Kodierung der Daten und Berechnung der Prüfsumme erfolgt ausserhalb der RTIO. Dafür steht ein C-Code-Modul „SentOutSMC\_Control“ mit Grundfunktionen als Vorlage zur Verfügung. Alle SMC\_\* Signale der RTIO werden dazu mit den entsprechenden Ports des C-Code-Moduls verbunden werden.

Das C-Code-Modul muss jedoch noch an die spezifische Anwendung angepasst werden durch:

- Festlegung von „SMC Format Mode“
- Festlegung von „SMC\_Sequence“ (zyklische Ausgabe einer Msg\_ID Reihe)
- Bereitstellen der physikalischen Datenwerte an den Input-Ports
- Umrechnen der Datenwerte und Zuweisen der Werte in „DataField[MsgId]“

Signalname	Datentyp	Anmerkungen
EnableChn	bool	0: Eingang wird nicht verwendet (Default) 1: Eingang wird verwendet
Nibble_xx	uint8	Wert des Nibble
ChnState	uint8	Jeder Ausgabekanal besitzt eine 4-Bit Zustandskodierung. Die einzelnen Bits beschreiben dabei unterschiedliche Einstellungen und die daraus resultierenden Zustände des Ausgangstreibers: - Bit 0: Ausgangspegel (0 = inaktiv, 1 = aktiv) - Bit 1: Überstrom am Low-Side FET (1 = aktiv) - Bit 2: Überstrom am High-Side FET (1 = aktiv) - Bit 3: Fehler erkannt (1 = aktiv)
SMC_Update-Counter	uint8	Zähler für Serial Messages [0..15]
SMC_Bit_Length	uint8	Bit-Länge Serial Message [0..31], 0 = SMC inaktiv
SMC_Data_Bit2	uint32	Eingangssignal, welches das Datenwort für Bit 2 bereithält
SMC_Data_Bit3	uint32	Eingangssignal, welches das Datenwort für Bit 3 bereithält
Checksum_Mode_Data	uint8	„Checksum Calculation Method“ = „CRC Mode 0...3 by Signal“: - 0: Legacy - 1: Recommended - 2: CRC = 0x0 - 3: CRC = 0xF „Checksum Calculation Method“ = „CRC Data 0...15 by Signal“: CRC Value [0..15]

**Tab. 20-11** Die Signale des ES5321-Out-SENT Device





## 21 ES5338.1 Carrier Board for Wheel Speed Sensor Simulation

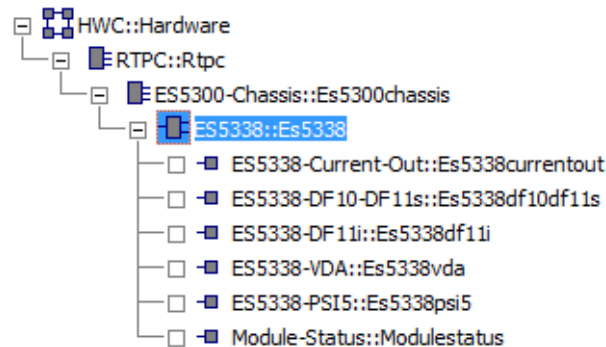
Mit dem ES5338.1 Carrier Board for Wheel Speed Sensor Simulation können folgende Arten von Sensoren nachgebildet werden:

- Aktive digitale Sensoren mit einem Strominterface mit zwei Stromleveln (Typ: DF10/DF11s)
- Aktive digitale Sensoren mit einem Strominterface mit zwei Stromleveln und zusätzlichen Informationen (Typ: DF11i)
- Aktive digitale Sensoren mit einem Strominterface mit drei Stromleveln und zusätzlichen Informationen (Typ: VDA)
- PSI5 Sensor-Signale in Anlehnung an PSI5 Standards V1.3 und V2.1

Darüber hinaus ist eine direkte Ansteuerung der Stromquelle über das Simulationsmodell („Direct Current Output“) möglich.

### Aufbau des ES5338.1 RTIO-Baums

Im RTIO-Editor wird das ES5338.1 Carrier Board for Wheel Speed Sensor Simulation durch Auswahl eines ES5338 Subsystems eingebunden.



**Abb. 21-1** RTIO-Hardwarebeschreibung mit eingebundener ES5338.1

Ein ES5300-Chassis System kann bis zu zehn ES5338 Subsysteme enthalten, von denen jedes wiederum sechs Module zur Sensorsimulation aufnehmen kann.

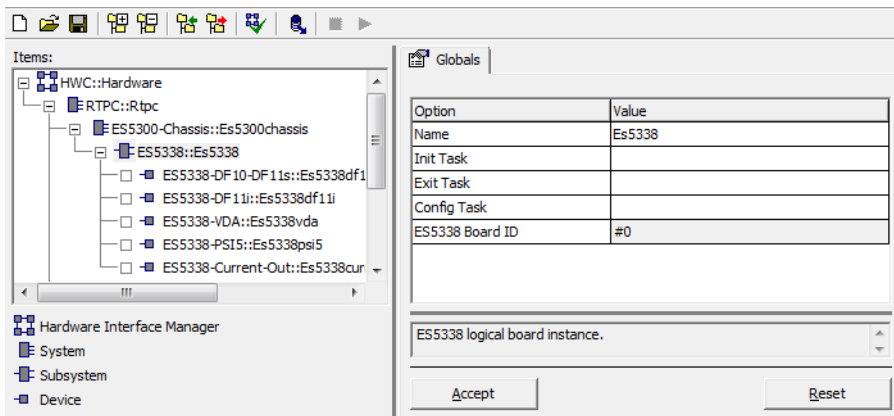
## 21.1 ES5338 Subsystem

---

### 21.1.1 Globals (ES5338 Subsystem)

---

Das ES5338 Subsystem dient zur Einstellung von Parametern, die global gültig sind, d.h. die auf alle ES5338.1 RTIO-Elemente wirken.



**Abb. 21-2** Die Registerkarte „Globals“ des ES5338 Subsystems

*Init Task, Exit Task, Config Task*

---

Die Tasks für Start-Up, Shut-Down und Konfiguration des Boards.

*ES5338 Board ID*

---

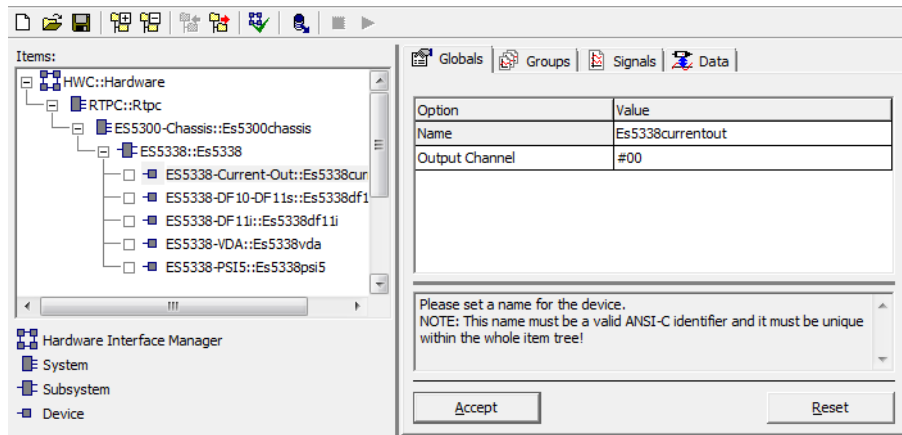
Dieses Optionsfeld dient zur Identifizierung der ES5338.1 – im RTIO-Editor können bis zu zehn ES5338.1 pro ES5300 eingebunden werden.

## 21.2 ES5338-Current-Out Device

Mit dem ES5338-Current-Out Device ist die direkte Ansteuerung der Stromquelle über das Simulationsmodell möglich.

### 21.2.1 Globals (ES5338-Current-Out Device)

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 21-3** Die Registerkarte „Globals“ des ES5338-Current-Out Device

#### Name

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### Output Channel

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll (siehe auch „Output Channel“ auf Seite 673).

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

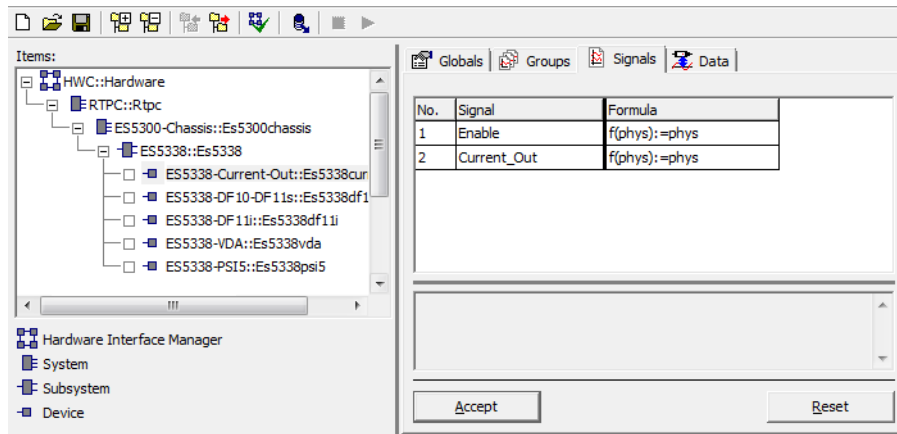
Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Output Channel	uint8	Nein	Verwendeter Ausgang [0...5]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 21-1** ES5338-Current-Out Device: Konfigurationsparameter der Registerkarte „Globals“

### 21.2.2 Signals (ES5338-Current-Out Device)

Das ES5338-Current-Out Device besitzt eine Send-Signalgruppe mit den folgenden Signalen.



**Abb. 21-4** Die Registerkarte „Signals“ des ES5338-Current-Out Device  
*Enable*

Mit diesem Signal wird die Stromquelle aktiviert bzw. deaktiviert.

*Current\_Out*

Der auszugebende Strom

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Current_Out	real32	Ja	Strom [0.0... 120.0 mA]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 21-2** ES5338-Current-Out Device: Signale

## 21.3 Radsensorsimulation – Gemeinsame Einstellungen

Die RTIO-Elemente zur Darstellung der simulierten Sensoren sind einheitlich aufgebaut und unterscheiden sich nur durch die spezifischen Eigenschaften der verschiedenen Sensoren.

### *Output Channel*

Jeder Sensor kann über das Optionsfeld „Output Channel“ in der Registerkarte „Globals“ einem Ausgang zugeordnet werden.

### *Operation Mode*

Mit dem Optionsfeld „Operation Mode“ können Sie festlegen, ob die Raddrehzahl über Geschwindigkeit, Zähnezah und Radradius oder direkt über die Zahnfrequenz des Sensors vorgegeben werden soll.

### *Zahnzahl („Number of Teeth“), Verhalten bei Zahnfehlern („Teeth Drop Pattern“)*

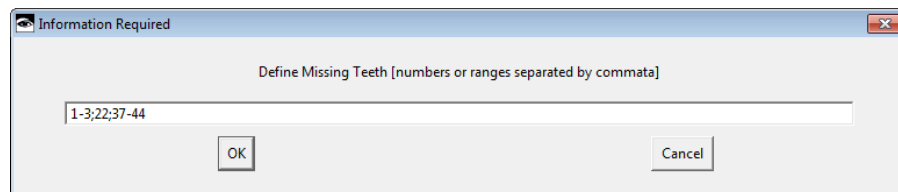
Außer der Zahnzahl (bzw. Polpaarzahl) pro Umdrehung können Sie für jeden Zahn definieren, ob dieser regulär oder fehlerhaft ausgegeben werden soll. Bei allen fehlerhaften Zähnen wird der im Optionsfeld definierte Spannungs- oder Stromwert während des gesamten Zahns ausgegeben. Die Wirksamkeit von Zahnfehlern ist noch vom Signal „ToothFaultEnable“ abhängig.

Die Definition von Zahnfehlern erfolgt über einen eigenen Eingabedialog, wobei einzelne durch Kommata getrennte Zahnnummern oder Zahnbereiche eingegeben werden können. Bereiche werden hierbei in der Form

`<erster Zahn> + <-> + <letzter Zahn>`

definiert.

Die Nummerierung beginnt bei 0, d. h. für ein Rad mit 48 Zähnen/Polpaaren ist der Bereich 0 – 47.



**Abb. 21-5** Eingabedialog zur Definition von Zahnfehlern

### 21.3.1 Gemeinsame Signale aller Sensoren

#### *Enable*

Mit diesem Signal wird der Sensor aktiviert bzw. deaktiviert. Im inaktiven Zustand ist der zugehörige Ausgang hochohmig geschaltet. Beim Aktivieren wird der Sensor neu gestartet und der Ausgang an den Spannungs- (beim Typ „DF10“) bzw. den Stromausgang (alle anderen Sensortypen) geschaltet. Beim Sensorneu-start wird mit Zahn 0 gestartet.

#### **Hinweis**

*Bitte beachten Sie, dass die Umschaltung des Ausgangs mittels Relais erfolgt und daher gewisse Schalt- bzw. Prellzeiten auftreten.*

### *Frequency*

---

Mit diesem Signal können Sie die Frequenz des Sensors direkt vorgeben (nur wirksam in der Betriebsart „Frequency“, siehe „Operation Mode“ auf Seite 673).

Bei Sensoren mit Drehrichtungserkennung wird durch einen negativen Wert die Drehrichtung umgekehrt (positiver Wert: vorwärts bzw. Normaldrehrichtung, negativer Wert: rückwärts). Bei Sensoren ohne Drehrichtungserkennung wird das Vorzeichen ignoriert.

### *WheelSpeed, WheelRadius*

---

Mit diesem Signal können Sie die Radgeschwindigkeit  $v$  und den Radradius  $r$  vorgeben. In der Betriebsart „Wheelspeed“ wird aus diesen Werten zusammen mit der Zähne-/Polpaarzahl  $z$  die Frequenz wie folgt berechnet:

$$f [\text{Hz}] = \frac{v [\text{km/h}] \cdot z}{r [\text{m}] \cdot 2\pi \cdot 3,6}$$

In der Betriebsart „Frequency“ haben die Signale keine Wirkung. Für einen negativen Wert der Radgeschwindigkeit gilt dasselbe, wie bei „Frequency“, d.h. für Sensoren mit Drehrichtungserkennung wird eine Drehrichtungsumkehr bzw. Rückwärtsdrehen simuliert.

### *ToothFaultEnable*

---

Mit diesem Signal können Sie die in der Registerkarte „Globals“ gemachten Einstellungen für Zahnfehler aktivieren (= 1) bzw. deaktivieren (= 0). Für jeden Zahn bzw. jedes Polpaar kann dabei festgelegt werden, ob reguläre oder fehlerhafte Ausgabe erfolgen soll. Für alle fehlerhaften Zähne eines Sensors wird ein festgelegter Strom-Pegel ausgegeben.

### *Übernahme von Signaländerungen an den Ausgängen*

---

Änderungen der Frequenz oder Radgeschwindigkeit werden mit dem Beschreiben des Registers am darauffolgenden Polwechsel (halber Zahn) am jeweiligen Ausgang wirksam.

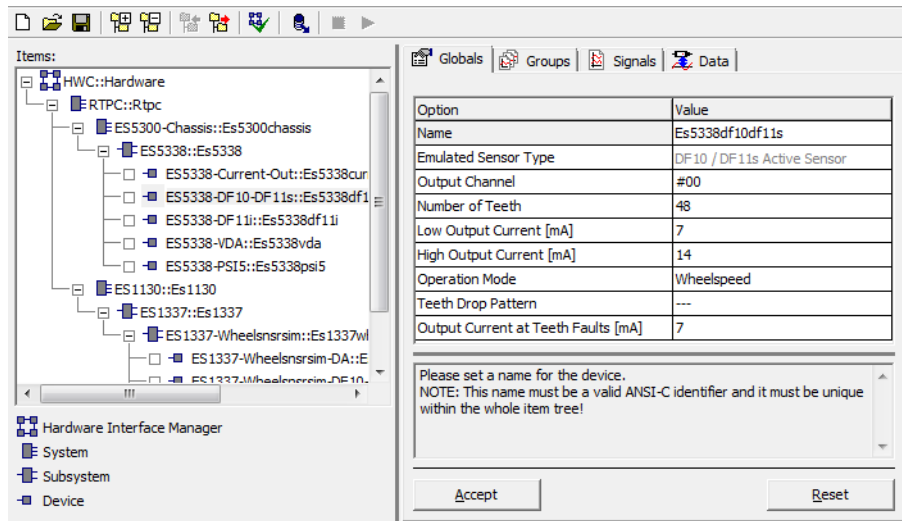
Protokollinformationen (Typ „DF11i“ und „VDA“) werden nicht mit Zahn oder Polwechseln synchronisiert. Es werden die zum Zeitpunkt des Registerzugriffs zur Verfügung stehenden Werte benutzt.

Auch die De-/Aktivierung der Zahnfehlersimulation mittels des Signals „ToothFaultEnable“ wird nicht mit Zahn-, Pol- oder Umdrehungswechseln synchronisiert. Damit können bei ungünstigen Umschaltzeitpunkten Teilausgaben von Protokoll- oder Zahninformationen erfolgen.

### 21.3.2 Globals (ES5338-DF10-DF11s Device)

Mit dem ES5338-DF10-DF11s Device können Sie einen Radsensor vom Typ „DF10“ bzw. „DF11s“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 21-6** Die Registerkarte „Globals“ des ES5338-DF10-DF11s Device

#### *Name*

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### *Output Channel*

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll (siehe auch „Output Channel“ auf Seite 673).

#### *Number of Teeth*

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### *Low Output Current*

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### *High Output Current*

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

#### *Operation Mode*

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen (siehe auch „Operation Mode“ auf Seite 673).

### Teeth Drop Pattern

Mit dieser Option können Sie ein Zahnfehlermuster definieren (siehe auch „Zahnzahl („Number of Teeth“), Verhalten bei Zahnfehlern („Teeth Drop Pattern“)) auf Seite 673).

### Output Current at Teeth Faults

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definiertem Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Output Channel	uint8	Nein	Verwendeter Ausgang [0...5]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 14 mA
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Fault	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 21-3** ES5338-DF10-DF11s Device: Konfigurationsparameter der Registerkarte „Globals“

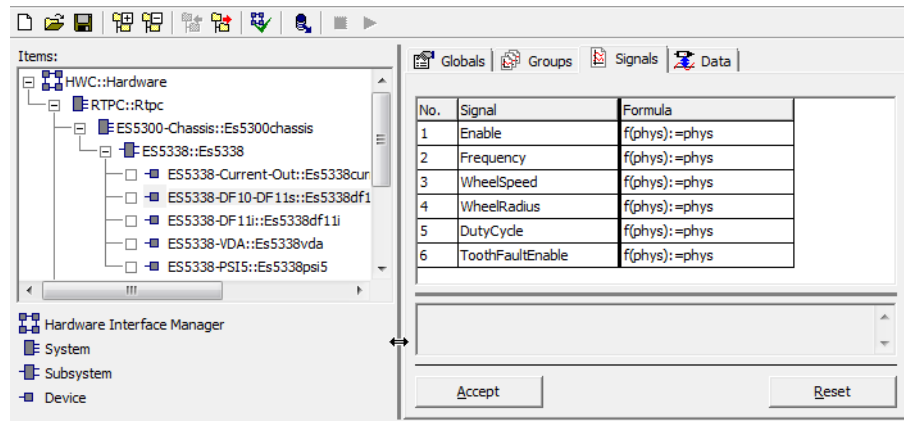
#### 21.3.3 Groups (ES5338-DF10-DF11s Device)

Das ES5338-DF10-DF11s Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.



### 21.3.4 Signals (ES5338-DF10-DF11s Device)

In der Registerkarte „Signals“ werden die Signale zur Steuerung der Sensor-simulation aufgelistet.



**Abb. 21-7** Die Registerkarte „Signals“ des ES5338-DF10-DF11s Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0,5 entsteht ein symmetrisches Ausgangssignal.

#### *ToothFaultEnable*

Mit diesem Signal können Sie das in der „Globals“ Registerkarte definierte Zahnfehlermuster aktivieren bzw. deaktivieren (siehe „Teeth Drop Pattern“ auf Seite 676).

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-20...20 kHz] (nur bei Betriebsart „Frequency“)
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h] (nur bei Betriebsart „WheelSpeed“)
WheelRadius	real32	Ja	Radradius [0.02...2 m] (nur bei Betriebsart „WheelSpeed“)
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95] (nur bei Betriebsart „WheelSpeed“)
ToothFaultEnable	bool	Ja	Zahnfehleraktivierung [0, 1]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 21-4** ES5338-DF10-DF11s Device: Signale

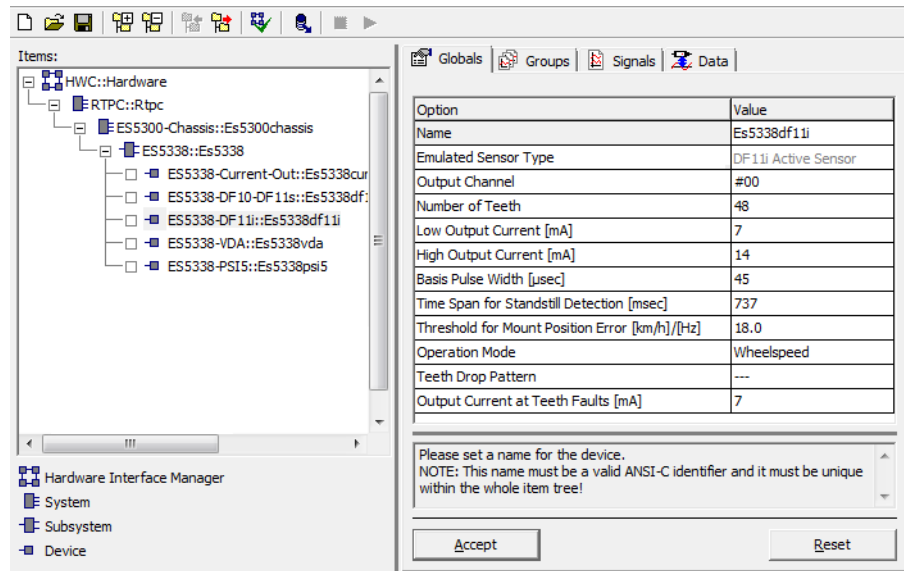
#### Hinweis

*Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 20 kHz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*

## 21.3.5 Globals (ES5338-DF11i Device)

Mit dem ES5338-DF11i Device können Sie mit einem der vier Sensorausgänge einen Radsensor vom Typ „DF11i“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 21-8** Die Registerkarte „Globals“ des ES5338-DF11i Device

### Name

Ein eindeutiger Bezeichner für dieses Device.

### Hinweis

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

### Output Channel

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

### Number of Teeth

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

### Low Output Current

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

### High Output Current

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.

### *Basic Pulse Width*

---

Mit diesem Parameter können Sie die Basispulsweite vorgeben. Andere Pulsbreiten sind von diesem Wert abgeleitet wie z.B.

- Luftspaltreserve
- Drehrichtung Links
- Drehrichtung Rechts
- Drehrichtung Links und Einbaulage
- Drehrichtung Rechts und Einbaulage

### *Time Span for Standstill Detection*

---

Mit diesem Parameter können Sie die Zeitdauer vorgeben, nach der der Stillstand des Rades erkannt und der Stillstandsimpuls ausgegeben wird. Die Stillstandsüberwachung wird mit einem Zahnimpuls oder Stillstandsimpuls zurückgesetzt, d.h. bei Stillstand werden Stillstandsimpulse in diesen Zeitabständen ausgegeben.

### *Threshold for Mount Position Error*

---

Mit dieser Option können Sie den Grenzwert zur Ausgabe eines Lagefehlers über die Raddrehzahl (Betriebsart „Wheelspeed“) bzw. die Frequenz (Betriebsart „Frequency“) festlegen. Unterschreitet die Raddrehzahl bzw. die Frequenz diesen Wert, so wird das Fehlersignal gesetzt.

### *Operation Mode*

---

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

### *Teeth Drop Pattern*

---

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

### *Output Current at Teeth Faults*

---

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definiertem Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Output Channel	uint8	Nein	Verwendeter Ausgang [0...5]]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 14 mA
Basic Pulse Width	uint8	Ja	Basispulsdauer [37...53 µs], Default: 45 µs
Time Span for Standstill Detection	uint8	Ja	Zeitspanne für Detektion eines stehenden Rades [100 ms...1000 ms], Default: 737 ms
Threshold for Mount Position Error	real32	Ja	Schwelle für Stillstandsverhalten [-100.0...100.0 km/h] oder [-500.0...500.0 Hz]
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche
Output Current at Teeth Fault	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

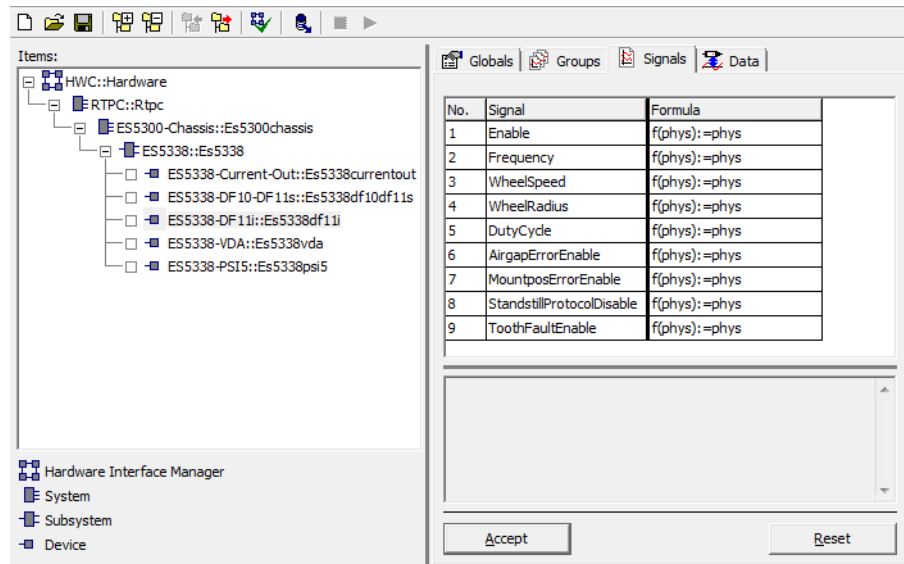
**Tab. 21-5** ES5338-DF11i Device: Konfigurationsparameter der Registerkarte „Globals“

### 21.3.6 Groups (ES5338-DF11i Device)

Das ES5338-DF11i Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

### 21.3.7 Signals (ES5338-DF11i Device)

In der Registerkarte „Signals“ werden die Signale zur Steuerung der Sensorsimulation aufgelistet.



**Abb. 21-9** Die Registerkarte „Signals“ des ES5338-DF11i Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0.5 entsteht ein symmetrisches Ausgangssignal.

#### *AirgapErrorEnable*

Mit diesem Signal können Sie die Bedingung „Luftspaltfehler“ aktivieren bzw. deaktivieren.

#### *MountposErrorEnable*

Mit diesem Signal können Sie die Bedingung „Lagefehler“ aktivieren bzw. deaktivieren.

#### *StandstillProtocolDisable*

Mit diesem Signal können Sie das für diesen Sensor bei Stillstand definierte Verhalten aktivieren bzw. deaktivieren.

ToothFaultEnable

Mit diesem Signal können Sie das in der „Globals“ Registerkarte definierte Zahnfehlermuster aktivieren bzw. deaktivieren.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1]
Frequency	real32	Ja	Frequenz [-20...20 kHz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m]
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95]
AirgapErrorEnable	real32	Ja	Ausgabe Bedingung „Luftspaltfehler“ [0.0 = Aus, 1.0 = Aktiv]
MountposError-Enable	real32	Ja	Aktivierung Lagefehler [0.0 = Aus, 1.0 = Aktiv]
Standstill-ProtocolDisable	real32	Ja	Aktivierung Stillstandprotokoll [0.0 = Aus, 1.0 = Aktiv]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0 = Aus, 1 = Aktiv]
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 21-6** ES5338-DF11i Device: Signale

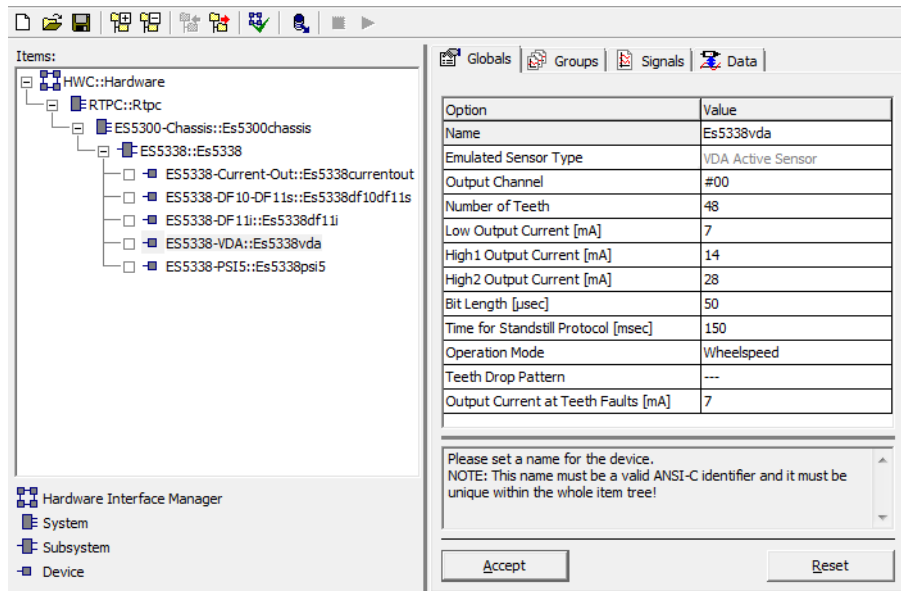
**Hinweis**

*Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 20 kHz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*

### 21.3.8 Globals (ES5338-VDA Device)

Mit dem ES5338-VDA Device können Sie einen Radsensor des Typs „VDA“ simulieren.

Die nachfolgende Abbildung zeigt die Parameter der Registerkarte „Globals“.



**Abb. 21-10** Die Registerkarte „Globals“ des ES5338-VDA Device

#### *Name*

Ein eindeutiger Bezeichner für dieses Device.

#### **Hinweis**

*Da dieser Name für Variablen- und Parameternamen verwendet wird, sind Leer- oder Sonderzeichen nicht zulässig!*

#### *Output Channel*

Mit dieser Option können Sie den Ausgang festlegen, auf den der simulierte Sensor geschaltet werden soll.

#### *Number of Teeth*

Mit diesem Parameter können Sie die Zahn- bzw. Polpaarzahl festlegen.

#### *Low Output Current*

Mit diesem Parameter können Sie den unteren Stromwert vorgeben.

#### *High1 Output Current*

Mit diesem Parameter können Sie den mittleren Stromwert vorgeben.

#### *High2 Output Current*

Mit diesem Parameter können Sie den oberen Stromwert vorgeben.



### *Bit Length*

---

Mit diesem Parameter können Sie die Basispulsweite vorgeben. Weitere Pulsweiten sind von diesem Wert abgeleitet wie

- Geschwindigkeitspuls
- Datenpuls
- Vorbit (Initial Bit)

### *Time for Standstill Protocol*

---

Mit diesem Parameter können Sie die Zeitdauer vorgeben, nach der der Stillstand des Rades erkannt und der Stillstandsimpuls ausgegeben wird. Die Stillstandsüberwachung wird mit einem Zahnimpuls oder Stillstandsimpuls zurückgesetzt, d.h. bei Stillstand werden Stillstandsimpulse in diesen Zeitabständen ausgegeben.

### *Operation Mode*

---

Mit dieser Option können Sie die Betriebsart zur Vorgabe der Raddrehzahl festlegen.

### *Teeth Drop Pattern*

---

Mit dieser Option können Sie ein Zahnfehlermuster definieren.

### *Output Current at Teeth Faults*

---

Mit diesem Parameter können Sie festlegen, welcher Stromwert bei einem als fehlerhaft definierten Zahn ausgegeben werden soll.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Name	Identifizier	Nein	Benutzerspezifischer Name
Output Channel	uint8	Nein	Verwendeter Ausgang [0...5]
Number of Teeth	uint8	Ja	Zähne-/Polpaarzahl [1...128], Default: 48
Low Output Current	uint8	Ja	Unterer Stromwert [0...35 mA], Default: 7 mA
High Output Current	uint8	Ja	Oberer Stromwert [0...35 mA], Default: 14 mA
Bit Length	uint8	Ja	Basispulsdauer [20...80 µs], Default: 50 µs
Time for Standstill Protocol [msec]	uint16	Ja	Zeitspanne für Detektion eines stehenden Rades [50 ms...255 ms], Default: 150 ms
Operation Mode	uint8	Ja	Betriebsart zur Vorgabe der Drehzahländerung [0 = Wheelspeed   1 = Frequency]
Teeth Drop Pattern	Kennlinie	Ja	Liste der als fehlerhaft zu simulierenden Zähne/Polpaare oder Zahn-/Polpaarbereiche, Default: kein Zahnfehler
Output Current at Teeth Faults	uint8	Ja	Stromwert bei Zahnfehler [0...35 mA], Default: 7 mA

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

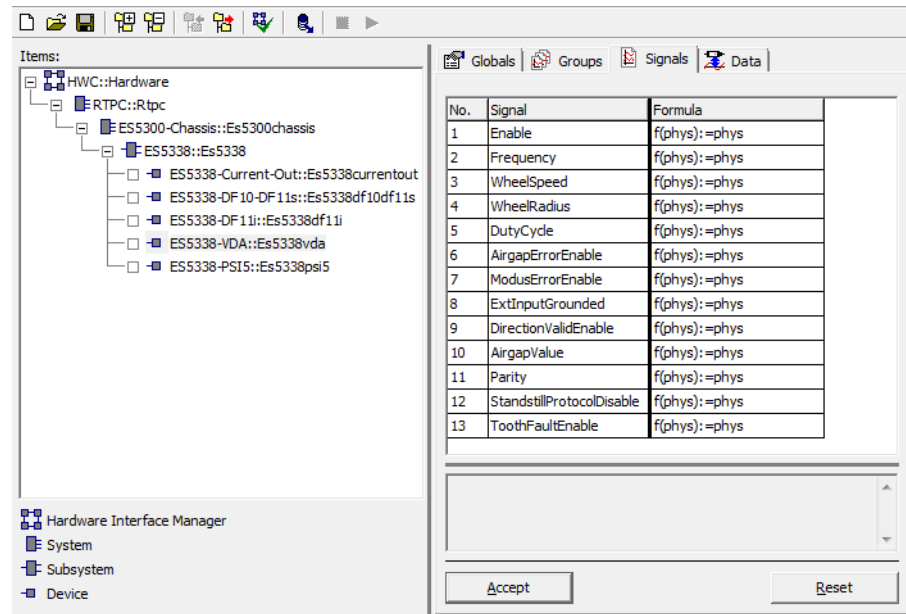
**Tab. 21-7** ES5338-VDA Device: Konfigurationsparameter der Registerkarte „Globals“

### 21.3.9 Groups (ES5338-VDA Device)

Das ES5338-VDA Device besitzt eine Signalgruppe „Output“, mit der die Sensorsimulation gesteuert wird.

## 21.3.10 Signals (ES5338-VDA Device)

In der Registerkarte „Signals“ werden die Signale zur Steuerung der Sensor-simulation aufgelistet.



**Abb. 21-11** Die Registerkarte „Signals“ des ES5338-VDA Device

#### *Enable*

Mit diesem Signal können Sie den Ausgang aktivieren bzw. deaktivieren.

#### *Frequency*

Über dieses Signal können Sie die Zahnfrequenz direkt vorgeben (für die Betriebsart „Frequency“).

#### *WheelSpeed, WheelRadius*

Über diese Signale können Sie die Radgeschwindigkeit bzw. den Radradius vorgeben (für die Betriebsart „Wheelspeed“).

#### *DutyCycle*

Über dieses Signal können Sie das Tastverhältnis der beiden Pole eines Zahns definieren. Bei einem Tastverhältnis von 0,5 entsteht ein symmetrisches Ausgangssignal.

#### *AirgapErrorEnable*

Mit diesem Signal können Sie die Bedingung „Luftspaltfehler“ aktivieren bzw. deaktivieren (Protokollbit 0).

#### *ModusErrorEnable*

Mit diesem Signal können Sie die Bedingung „Modusfehler“ aktivieren bzw. deaktivieren (Protokollbit 1).

*ExtInputGrounded*

Mit diesem Signal können Sie das zugehörige Protokollbit 2 beeinflussen. Eine Null signalisiert „Externer Eingang offen“.

*DirectionValidEnable*

Mit diesem Signal können Sie das zugehörige Protokollbit 3 beeinflussen. Eine Null signalisiert, dass die übertragene Drehrichtung gültig ist.

*AirgapValue*

Mit diesem Signal können Sie den Wert der zugehörigen Protokollbits 5...7 beeinflussen. Der Luftspaltwert kann sich im Bereich zwischen 0 und 7 bewegen.

*Parity*

Mit diesem Signal können Sie die Paritätsbildung beeinflussen (Protokollbit 8). Ein Wert von Null entspricht gerader Parität.

Die Bedeutung der Protokollbits ist in der folgenden Tabelle zusammengefasst.

Bit	Bedeutung	Kodierung
0	Fehlerbit Luftspaltreserve	0 = korrekt 1 = Luftspaltreserve
1	Frei vergebbar	
2	Frei vergebbar	
3	Gültigkeit der Drehrichtung	0 = gültig 1 = ungültig
4	Drehrichtung - wird aus dem Vorzeichen von „WheelSpeed“ oder „Frequency“ berechnet	0 = positiv 1 = negativ
5	Frei vergebbar	
6	Frei vergebbar	
7	Frei vergebbar	
8	Paritäts-Bit	Wird auf 0 oder 1 gesetzt, um gerade Parität (inkl. dem Paritäts-Bit selbst) beizubehalten.

**Tab. 21-8** Bedeutung der neun Protokollbits

*StandstillProtocolDisable*

Mit diesem Signal können Sie das Stillstandsprotokollverhalten des Sensors beeinflussen

*ToothFaultEnable*

Mit diesem Signal können Sie das in der Registerkarte „Globals“ definierte Zahnfehlermuster aktivieren bzw. deaktivieren.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Sensoraktivierung [0, 1], Default: 1
Frequency	real32	Ja	Frequenz [-20...20 kHz]
WheelSpeed	real32	Ja	Radgeschwindigkeit [-1000.0...1000.0 km/h]
WheelRadius	real32	Ja	Radradius [0.02...2 m], Default: 0,32 m
DutyCycle	real32	Ja	Tastverhältnis [0.05...0.95], Default: 0,5
AirgapErrorEnable	real32	Ja	Aktivierung „Luftspaltfehler“ [0.0 = aus, 1.0 = aktiv]
ModusErrorEnable	real32	Ja	Aktivierung „Lagefehler“ [0.0 = aus, 1.0 = aktiv]
ExtInputGrounded	real32	Ja	Pegel ext. Eingang [0.0 = offen, 1.0 = Ground]
DirectionValidEnable	real32	Ja	Signalisierung Drehrichtung gültig [0.0 = Drehrichtung gültig, 1.0 = Drehrichtung ungültig]
AirgapValue	real32	Ja	Wert Luftspalt [0...7]
Parity	real32	Ja	Einstellung Parität [0.0 = gerade, 1.0 = ungerade]
Standstill-ProtocolDisable	real32	Ja	Aktivierung Stillstandprotokoll [0.0 = aktiv, 1.0 = inaktiv]
ToothFaultEnable	bool	Ja	Zahnfehlerabschaltung [0 = aus, 1 = aktiv]
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 21-9** ES5338-VDA Device: Signale

**Hinweis**

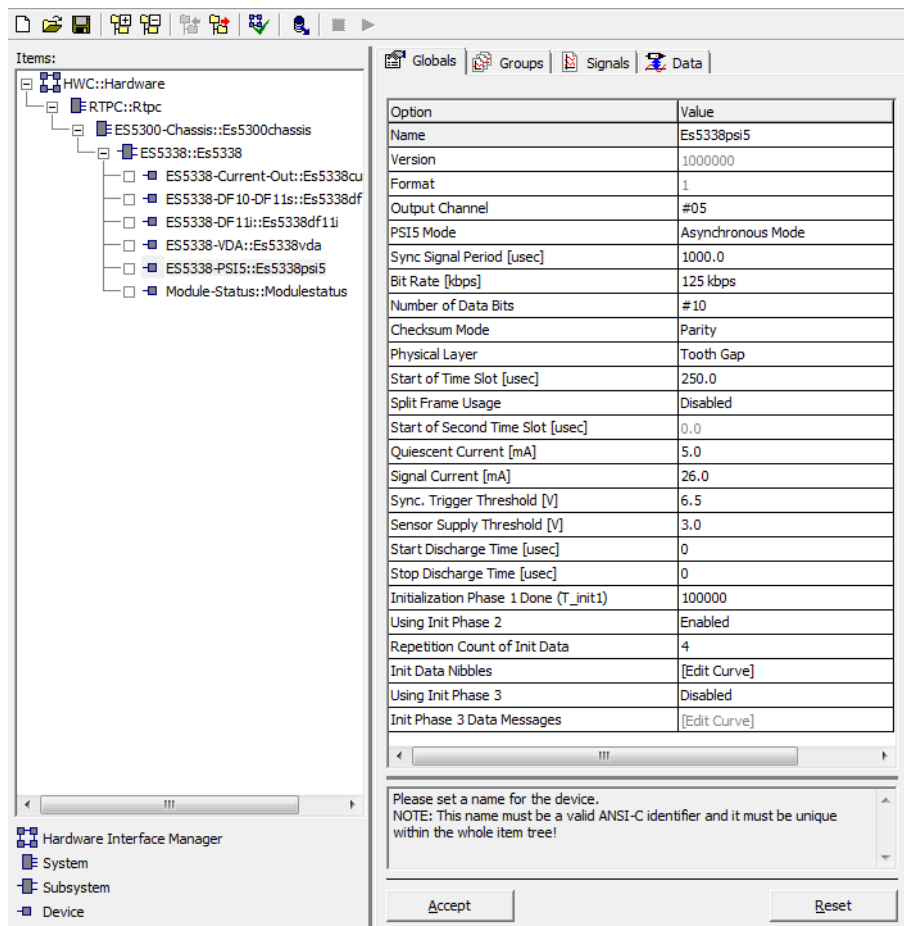
*Der Betrag der resultierenden Frequenz wird unabhängig von der Betriebsart auf 20 kHz begrenzt – bei Überschreitung wird eine entsprechende Fehlermeldung generiert.*

## 21.4 ES5338-PSI5 Device

Mit dem ES5338-PSI5 Device wird die Ausgabe der PSI5-Signale konfiguriert.

### 21.4.1 Globals (ES5338-PSI5 Device)

In der Registerkarte „Globals“ können folgende Einstellungen vorgenommen werden:



**Abb. 21-12** Die Registerkarte „Globals“ des ES5338-PSI5 Device

#### *Output Channel*

Mit dieser Option können Sie den Ausgang festlegen, auf dem das PSI5-Signal ausgegeben werden soll.

#### *PSI5 Mode*

Der Kommunikationsmodus des Sensors:

- Asynchronous Mode
- Sync. Parallel Bus Mode
- Sync. Universal Bus Mode

---

*Sync Signal Period [ $\mu\text{sec}$ ]*

---

Definiert die maximale Periodenlänge des Sync-Signals (0...6553  $\mu\text{s}$ , Auflösung 0,1  $\mu\text{s}$ ).

---

*Bit Rate [kpbs]*

---

Bitrate des Sensors (125 kbps, 189 kbps)

---

*Number of Data Bits*

---

Definiert die Anzahl der Datenbits des PSI5-Sensors (10...28).

---

*Checksum Mode*

---

Definiert die Methode zur Berechnung der Prüfsumme (Parity, CRC).

---

*Physical Layer*

---

Definiert die Kommunikationsmethode zwischen Sensor und Steuergeräte auf dem physikalischen Layer (Tooth Gap, Pulse Width).

---

*Start of Time Slot [ $\mu\text{sec}$ ]*

---

Definiert den Zeitpunkt (nach Erhalt des Sync-Pulses), ab dem die Sensorsimulation die PSI5-Daten sendet.

---

*Split Frame Usage*

---

Aktiviert bzw. deaktiviert den „Split Frame“ Modus.

Dabei werden die Nutzdaten auf zwei Datenpakete aufgeteilt: Beispielsweise kann man 16 Bit Nutzdaten in 2 x 8 Bit aufteilen und in Frames mit 10 Datenbits versenden.

---

*Start of Second Time Slot [ $\mu\text{sec}$ ]*

---

Definiert den Zeitpunkt (nach Erhalt des Sync-Pulses), ab dem die PSI5-Daten des zweiten Frames gesendet werden.

---

*Quiescent Current [mA]*

---

Definiert den Ruhestrom bzw. die normale Stromaufnahme eines PSI5-Sensors (= die logische „0“ in der Sensor-Steuergerätekommunikation).

---

*Signal Current [mA]*

---

Die erhöhte Stromaufnahme bei Darstellung einer logischen „1“ in der Sensor-Steuergerätekommunikation.

---

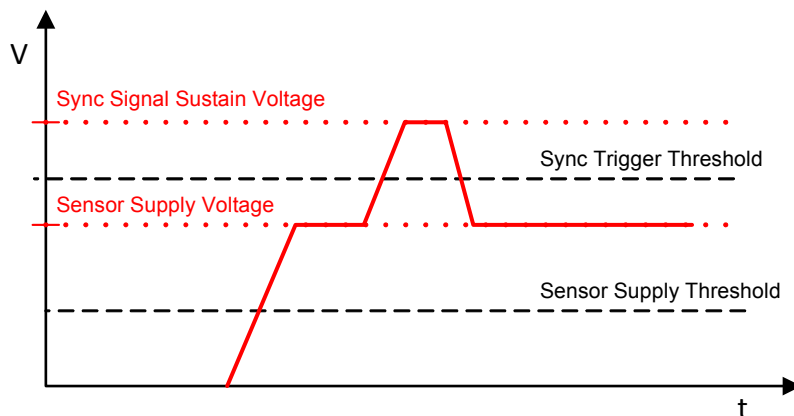
*Sync Trigger Threshold [V]*

---

Definiert den Spannungswert, ab dem die Hardware einen Spannungspuls vom Steuergerät als solchen erkennt (siehe auch Abb. 21-13 auf Seite 692).

Sensor Supply Threshold [V]

Definiert einen Schwellwert, ab dem die ES5338.1 die Versorgungsspannung als gültig akzeptiert. Bereich: 0...60 V, typischer Wert: Versorgungsspannung/2 (siehe auch Abb. 21-13).



**Abb. 21-13** Typischer Signalverlauf mit Schwellenspannungen

Start Discharge Time

Definiert den Startzeitpunkt einer erhöhten Stromaufnahme des PSI5-Sensors. Nur sichtbar, wenn **View** → **Show All** gewählt wurde.

Stop Discharge Time

Definiert den Endzeitpunkt einer erhöhten Stromaufnahme des PSI5-Sensors. Nur sichtbar, wenn **View** → **Show All** gewählt wurde.

Initialization Phase 1 Done ( $T_{init1}$ )

Definiert den Zeitpunkt, zu dem die Initialisierungsphase 1 abgeschlossen ist und der Sensor mit dem Senden der Init Phase 2 Daten-Nibbles beginnen kann.

Using Init Phase 2

Aktiviert/deaktiviert das „Init Phase 2“ Feature.

Wenn dieser Parameter deaktiviert ist, werden die Parameter „Repetition Count“ und „Init Data Nibbles“ ausgegraut.

Repetition Count of Init Data

Definiert wie häufig die Init Phase Data Nibbles gesendet werden.

Init Data Nibbles

Tabelle zu Speicherung der Init Phase Data Nibbles. Die Anzahl der Einträge in dieser Tabelle entspricht auch die Anzahl der Init Phase Data Nibbles.

Using Init Phase 3

Definiert ob eine „Init Phase 3“ verwendet werden soll oder nicht.



### *Init Phase 3 Data Messages*

Nach Ende der Init Phase 2 liefert ein Sensor entweder „OK“ (in Form von speziellen Datenpaketen) oder einen Fehlercode zurück. Wenn eine Init Phase 3 verwendet wird, können die Daten hiermit definiert werden.

Die folgenden Tabelle fasst die Konfigurationsparameter zusammen:

<b>Parameter bzw. Optionsfeld</b>	<b>Datentyp*</b>	<b>Online editierbar</b>	<b>Bemerkung/Wertebereich</b>
Name	Identifizier	Nein	Benutzerspezifischer Name
Output Channel	uint8	Nein	Der verwendete Ausgang
PSI5 Mode	uint8	Ja	Kommunikationsmodus: - Asynchronous Mode - Sync. Parallel Bus Mode - Sync. Universal Bus Mode Default: Asynchronous Mode
Sync Signal Period	real32	Ja	Maximale Periodendauer des Sync-Signals [0...6553 µs], Auflösung: 0,1 µs, Default: 1000.0 µs
Bit Rate [kpbs]	uint8	Ja	Bitrate des Sensors [125 kpbs, 189 kpbs] Default: 125 kpbs
Number of Data Bits	uint8	Ja	Anzahl der Datenbits [10...28], Default: 10 Bits
Checksum Mode	uint8	Ja	Berechnung der Prüfsumme [CRC, Parity], Default: Parity
Physical Layer	uint8	Ja	Kommunikation zum Steuergerät auf Physical Layer [Tooth Gap, Pulse Width] Default: Tooth Gap
Start of Time Slot	real32	Ja	Beginn des Time Slots [0...6553 µs], Auflösung: 0,1 µs Default: 250.0 µs
Split Frame Usage	uint8	Ja	Verwendung des „Split Frame“ Modus [Enable, Disable] Default: Disable
Start of Second Time Slot	real32	Ja	Beginn des zweiten Time Slots [0...6553 µs], Auflösung: 0,1 µs Default: 0.0 µs
Quiescent Current	real32	Ja	Ruhestrom des Sensors [0... mA] (keine obere Grenze) Default: 10 mA

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Signal Current	real32	Ja	Stromaufnahme bei Darstellung einer logischen „1“ [0... mA] (keine obere Grenze) Default: 26 mA
Sync Trigger Threshold	real32	Ja	Schwelle für Pulserkennung [0...60 V] Default: 3.5 V
Sensor Supply Threshold	real32	Ja	Schwelle für Versorgungsspannung [0...60 V] Default: 2.0 V
Start Discharge Time *	real32	Ja	Beginn der erhöhten Stromaufnahme [0... µsec] (keine obere Grenze) Default: 0 µsec
Stop Discharge Time *	real32	Ja	Ende der erhöhten Stromaufnahme [0... µsec] (keine obere Grenze) Default: 0 µsec
Initialization Phase 1 Done (T_init1)	real32	Ja	Ende „Initialization Phase 1“ [0... µsec] (keine obere Grenze) Default: 100000 µsec
Using Init Phase 2	uint8	Ja	Aktiviert/deaktiviert die Initialization Phase 2 des PSI5-Sensors [Enable / Disable] Default: Enabled
Repetition Count of Init Data	real32	Ja	Wiederholungen „Senden der Initialisierungsdaten“ [0...255] Default: 4
Init Data Nibbles	Table	Ja	Werden als Wellenform (Curve) eingegeben – jeder Punkt repräsentiert ein Daten-Nibble.
Using Init Phase 3	uint8	Ja	Aktiviert/deaktiviert die Initialization Phase 3 des PSI5-Sensors [Enable / Disable] Default: Disabled
Init Phase 3 Data Messages	Table	Ja	Enthält Informationen über die Init Phase 3 Data Messages
* Nur sichtbar, wenn <b>View → Show All</b> gewählt			

**Tab. 21-10** ES5338-PSI5 Device: Konfigurationsparameter der Registerkarte „Globals“

21.4.2 Signals (ES5338-PSI5 Device)

Das ES5338-PSI5 Device besitzt folgende Signale:

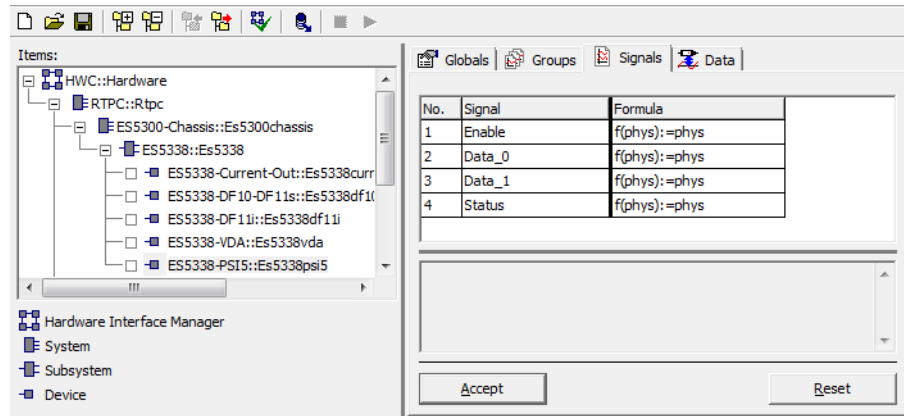


Abb. 21-14 Die Registerkarte „Signals“ des ES5338-PSI5 Device

Signale der Signalgruppe „Send“:

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Enable	bool	Ja	Aktiviert/deaktiviert den Signalausgang [0,1], Default: 1 (= Ausgangs aktiv)
Data_0	real64	Ja	Datenregister 0 des PSI5-Sensors
Data_1	real64	Ja	Datenregister 1 des PSI5-Sensors (nur im „Split Frame“-Modus)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

Signale der Signalgruppe „Receive“:

Parameter bzw. Optionsfeld	Datentyp*	Bemerkung/Wertebereich
Status	real64	Status- und Errorbits des PSI5-Sensors.

**Statusbits Bit 0...Bit 2:**

- 000000: Sensor stromlos / Reset
- 000001: Sensor in „Init Phase 1“
- 000010: Sensor in „Init Phase 2“
- 000011: Sensor in „Init Phase 3“
- 000100: Sensor in Betrieb

**Error Bits Bit 4...Bit 5:**

- 010000: Überspannung detektiert
- 100000: Überlast detektiert, Ausgangsstrom verringert.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet



## 22 ES5340.1/2 Electric Drive Simulation Board

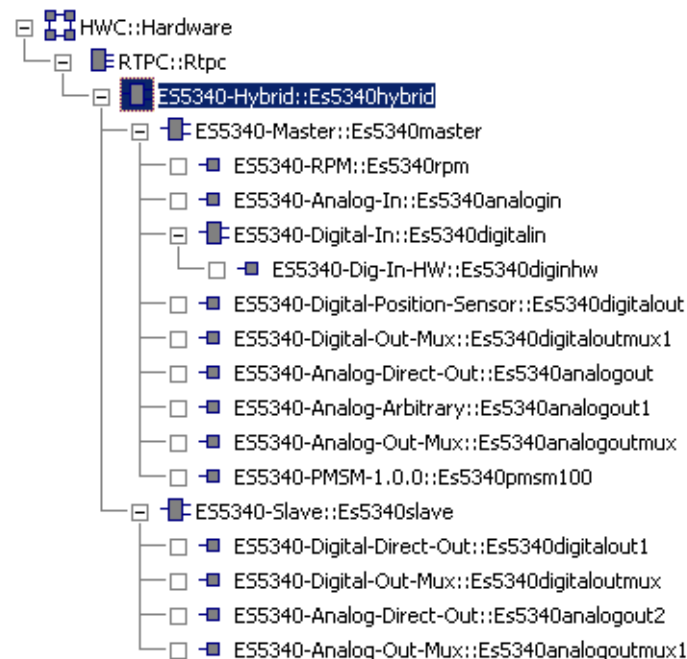
Das ES5340.1/2 Electric Drive Simulation Board mit PCI-Express-Schnittstelle dient zum Testen von Steuergeräten für Wechselrichter/E-Motoren auf Signalebene und kann als Multi-I/O-Karte zum Testen von Steuergeräten aus anderen Bereichen (z.B. Antriebsstrang oder Batteriemangement) eingesetzt werden.

Näheres zu Funktionsweise und Simulation des elektrischen Antriebs können Sie dem ES5340.1/2 Electric Drive Simulation Board Benutzerhandbuch entnehmen.

### Aufbau der RTIO-Hierarchie

Im Folgenden wird der Aufbau der RTIO-Hierarchie im RTC Editor sowie die Konfigurations- und Echtzeitdaten der einzelnen Items näher beschrieben.

Um im LABCAR RTC Editor eine neue ES5340.1/2-Hierarchie aufzubauen, fügen Sie ein ES5340-Hybrid Subsystem auf der RTPC-Systemebene ein. Dieses RTIO-Element dient als Container für die Beschreibung einer Master- und wahlweise einer Slave-Karte und bietet keine weiteren Einstellmöglichkeiten.



**Abb. 22-1** Vollständiger ES5340-Hybrid RTIO-Baum mit Master- und Slavekarte

### Electric Drive Simulation Master Board (ES5340.1/2-M)

Dieses Board enthält den vollen Funktionsumfang:

#### I/O:

- 8 Analogausgänge (inkl. Simulation von analogen Drehgebersignalen)
- 8 Digitalausgänge (inkl. Simulation von digitalen Drehgebersignalen)
- 20 Digitaleingänge
- 4 Analogeingänge

**Modellunterstützung:**

- FPGA-Modell erhältlich
- RTPC-Modell möglich

Um die Masterkarte benutzen zu können, fügen Sie im LABCAR RTC Editor auf der ES5340.1/2-Hybrid Subsystemebene ein ES5340.1/2-Master Subsystem (siehe „ES5340-Master“ auf Seite 699) ein.

***Electric Drive Simulation Slave Board (ES5340.1/2-S)***

---

Dieses Board weist einen gegenüber dem Masterboard reduzierten Funktionsumfang auf:

**I/O:**

- 6 Analogausgänge (ohne Simulation von analogen Drehgebersignalen)
- 6 Digitalausgänge (ohne Simulation von digitalen Drehgebersignalen)

Die Quelle der Ausgangssignale sind „Analog-/Digital-Direct-Out“-Ausgänge der Slave-Karte und insbesondere die Werte eines FPGA-Modells der ES5340.1-M.

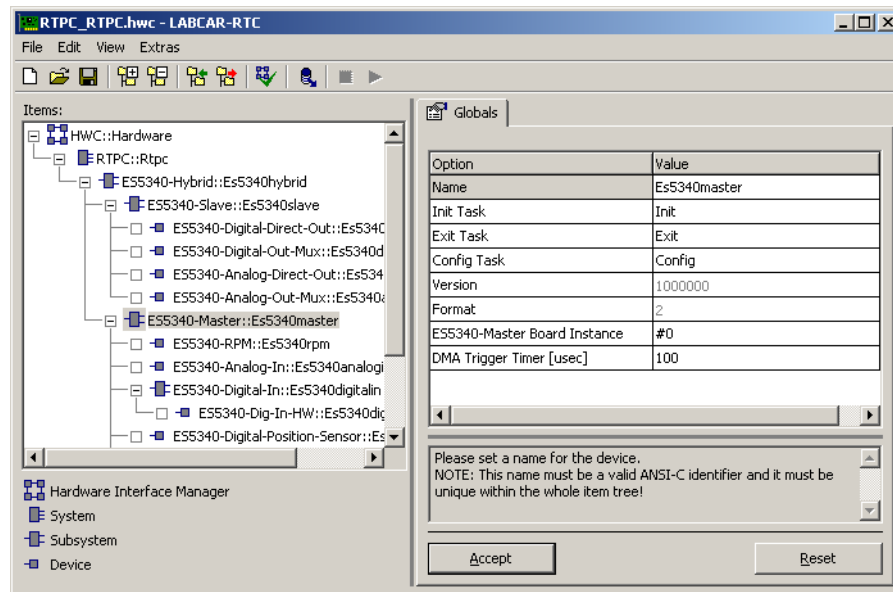
Um die Slavekarte benutzen zu können, fügen Sie im LABCAR RTC Editor auf der ES5340.1/2-Hybrid Subsystemebene ein ES5340-Slave Subsystem (siehe „ES5340-Slave“ auf Seite 764) ein.

## 22.1 ES5340-Master

Das ES5340-Master Subsystem dient zur Konfiguration einer ES5340-Masterkarte.

### 22.1.1 Globals (ES5340-Master Subsystem)

In diesem Abschnitt können Sie globale Einstellungen für die ES5340-Masterkarte vornehmen.



**Abb. 22-2** Registerkarte „Globals“ des ES5340-Master Subsystems (Option „Show All“)

#### *ES5340-Master Board Instance*

Diese Optionsfeld dient zur Festlegung der Karteninstanz, die diesem RTIO-Subsystem zugeordnet ist. Die Nummerierung beginnt bei 0 und die Karten eines Typs werden (ähnlich wie beim VMEbus-System) von links nach rechts aufsteigend nummeriert<sup>1</sup>.

Hierzu muss jedoch der interne Winkeltaktbus (SYNC-Interface) über alle Karten verbunden sein (siehe Benutzerhandbuch) – die Karteninstanz kann während des Experiments nicht verändert werden.

#### *DMA Trigger Time [usec]*

Dieses Feld wird nur angezeigt, wenn im Kontextmenü (rechte Maustaste) der Eintrag „Show All“ gewählt wurde.

Sie können hier den zeitlichen Abstand für die Aktivierung zweier asynchroner DMA-Übertragungen wählen – die gemessenen Daten werden von der ES5340.1/2-Masterkarte direkt in den PC-Hauptspeicher geschrieben.

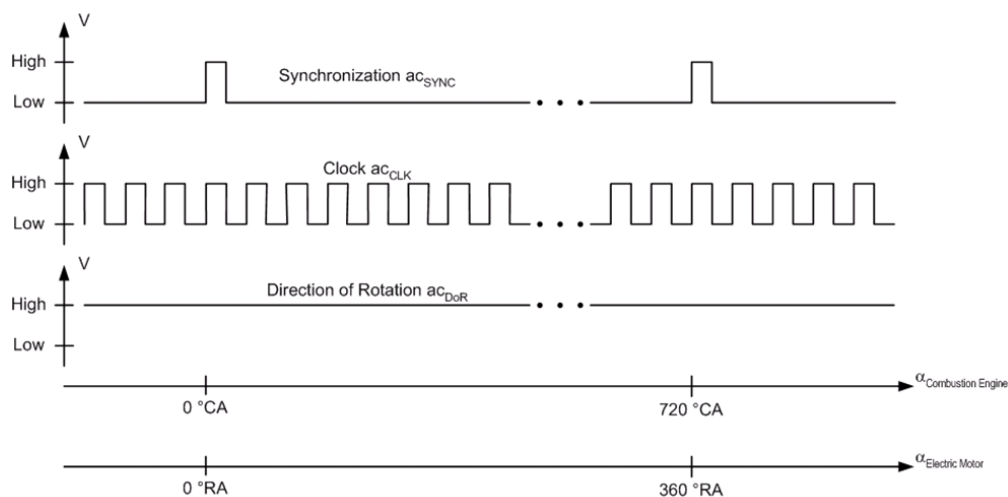
<sup>1</sup> Gilt nur unter der Voraussetzung, dass die Karten über die SYNC-Bus-Leitungen verbunden sind. Ansonsten erfolgt die Nummerierung zufällig bzw. hardwareabhängig.

Dies betrifft die letzten vier Zeitstempel aller 20 digitalen Eingänge, die aktuellen Messwerte der vier Analogwerte, die RTIO-Eingangsgrößen der Drehzahleinheit und bei Verwendung eines FPGA-Modell dessen RTIO-Eingangsgrößen. Dieses Datum kann während der Simulation durch Rekonfiguration verändert werden.

- Defaultwert: 100  $\mu$ s
- Einstellbereich: 5  $\mu$ s ... 1 s

## 22.2 ES5340-RPM – Drehzahleinheit

Die winkelbasierte Synchronisation zwischen LABCAR-Karten erfolgt über ein aus drei Komponenten bestehendes Winkeltaktsignal (Abb. 22-3). Das Signal „ac<sub>SYNC</sub>“ zeigt einen Nulldurchgang des Kurbelwellenwinkels beim Verbrennungsmotor bzw. des Rotorwinkels beim Elektromotor an, „ac<sub>CLK</sub>“ transportiert den eigentlichen Winkeltakt, „ac<sub>DoR</sub>“ codiert die Drehrichtung des Motors.



**Abb. 22-3** Die drei Komponenten des Winkeltaktsignals

Das Taktsignal bietet eine Auflösung  $\delta_\alpha$  von 16 Bit pro 720°KW des Verbrennungsmotors bzw. 360°RW des Elektromotors.

$$\delta_{\alpha, \text{Verbrennungsmotor}} = \frac{720^\circ KW}{2^{16}} = 0,011^\circ KW$$

**Gln. 22-1**

$$\delta_{\alpha, \text{Elektromotor}} = \frac{360^\circ RW}{2^{16}} = 0,0055^\circ RW$$

**Gln. 22-2**

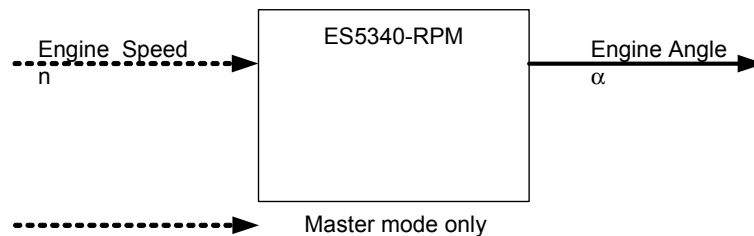
Die Drehzahleinheit der ES5340.1/2 kann im Slave- und Master-Modus betrieben werden. Im Slave-Modus synchronisiert sich die Drehzahleinheit auf einen (von einer anderen Karte im System) generierten Winkeltakt. Im Master-Modus generiert die Drehzahleinheit den Winkeltakt für andere Karten im System. Zur Übertragung des Winkeltakts zwischen den Karten stehen zwei Busse zur Verfügung. An einem Bus kann es jeweils nur einen Master geben.



Die Steuer- und Statussignale der Drehzahleinheit sind abhängig vom simulierten Motortyp. Bei der Simulation von Verbrennungsmotoren ist die Motordrehzahl  $n$  (in Umdrehungen pro Minute) die Eingangsgröße der Drehzahleinheit. Sie ist nur dann relevant, wenn die Drehzahleinheit für den Masterbetrieb konfiguriert ist. Als Ausgangsgröße wird die momentane Motorposition  $\alpha$  (in Grad Kurbelwelle) geliefert.

Signal	Richtung	Beschreibung
$n$	Eingang	Motordrehzahl in Umdrehungen pro Minute
$\alpha$	Ausgang	Motorwinkel in Grad Kurbelwelle

**Tab. 22-1** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Verbrennungsmotoren

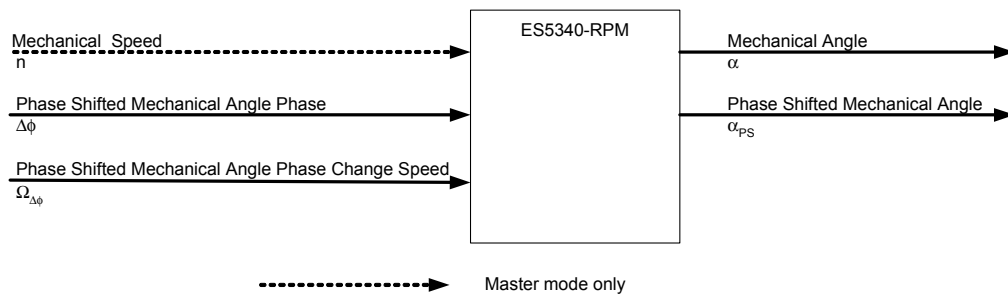


**Abb. 22-4** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Verbrennungsmotoren

Die Eingangs- und Ausgangsgrößen bei der Simulation von Elektromotoren sind in Tab. 22-2 aufgelistet bzw. in Abb. 22-5 dargestellt. Neben der aktuellen Motorposition  $\alpha$  wird hier noch der Wert eines zum Motorwinkel phasenverschobenen Winkels  $\alpha_{PS}$  ausgegeben. Die Phasenlage  $\Delta\phi$  relativ zum Motorwinkel sowie die Phasennachführungsgeschwindigkeit  $\Omega_{\Delta\phi}$  werden über entsprechende Eingänge der Drehzahleinheit gesteuert.

Signal	Richtung	Beschreibung
$n$	Eingang	Motordrehzahl in Umdrehungen pro Minute
$\Delta\phi$	Eingang	Phasenlage des phasenverschobenen mechanischen Winkels in °RW
$\Omega_{\Delta\phi}$	Eingang	Nachführungsgeschwindigkeit in °RW pro Sekunde des phasenverschobenen mechanischen Winkels bei einer Änderung des Sollwerts
$\alpha$	Ausgang	Rotorposition bzw. mechanischer Winkel in °RW
$\alpha_{PS}$	Ausgang	phasenverschobener mechanischer Winkel in °RW

**Tab. 22-2** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Elektromotoren im einfachen Masterbetrieb



**Abb. 22-5** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Elektromotoren im einfachen Master-Betrieb

Neben den einfachen Masterbetrieb bietet die Drehzahleinheit bei der Simulation von Elektromotoren einen weiteren Betriebsmodus an, den sogenannten „Masterbetrieb mit Winkeladaption“. Die Funktionsweise dieser Betriebsart wird im Folgenden anhand Abb. 22-6 auf Seite 703 erläutert.

Dies Abbildung zeigt den prinzipiellen Ablauf beim Steuergerätestest mit Hilfe eines LABCARs. Die Messdaten werden zu Beginn eines Simulationszyklus von der LABCAR-Hardware in das Simulationsmodell eingelesen. Das Fahrzeugmodell wird basierend auf diesen Daten gerechnet und nach Abschluss der Berechnung werden die berechneten Steuergrößen an die LABCAR-Hardware ausgegeben.

Im Gegensatz zum einfachen Masterbetrieb, bei dem nur die Motordrehzahl als Steuergröße der Drehzahleinheit zugeführt wird, wird beim Masterbetrieb mit Winkeladaption zusätzlich ein Winkelwert vom Modell berechnet und an die Drehzahleinheit übergeben.

Dieser Winkelwert ( $\alpha_T(t_1)$ ,  $\alpha_T(t_2)$ ,  $\alpha_T(t_3)$  usw.) stellt den Winkel dar, den der Motor zu Beginn des jeweils nächsten Simulationszyklus (Zeitpunkte  $t_1$ ,  $t_2$ ,  $t_3$  usw.) haben sollte.

In der Realität kann bzw. wird der tatsächliche Winkel des Motors (dargestellt als  $\alpha_{HW}(t_1)$ ) von diesem Zielwert abweichen. Die Drehzahleinheit berechnet deshalb intern ein Drehzahladditiv, das diese Winkeldifferenz bis zum Beginn des nächsten Simulationszyklus ( $t_2$  in diesem Fall) ausgleicht:

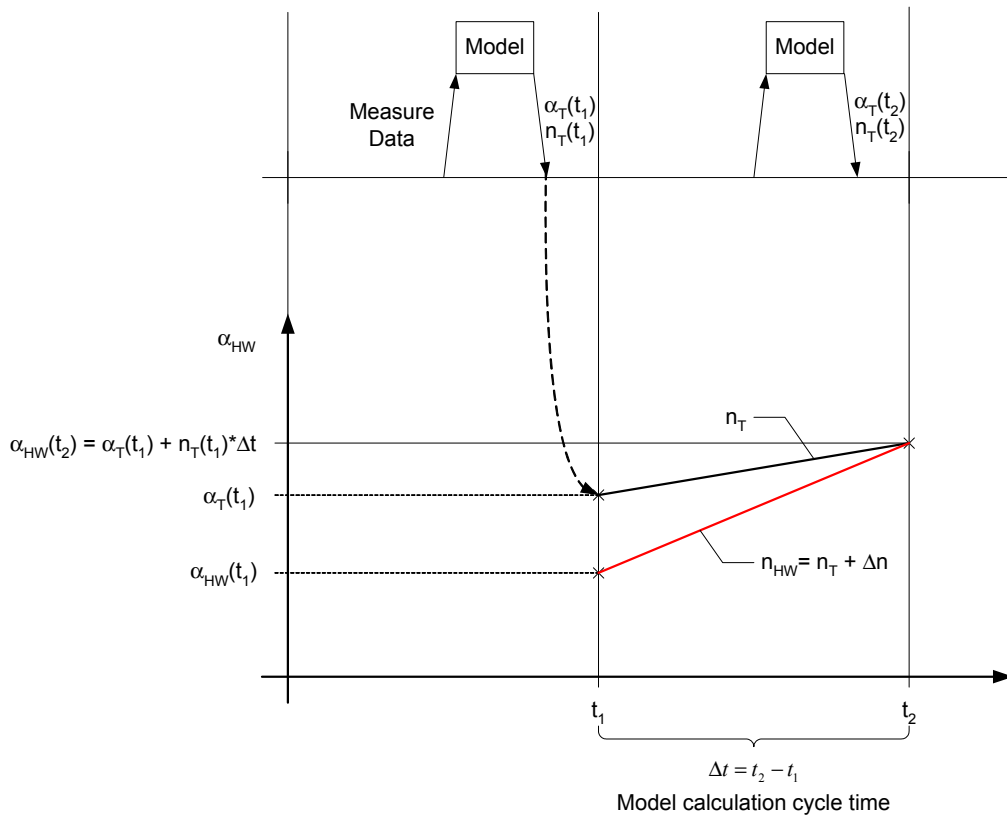
$$n_{HW} = n_T + \frac{\alpha_T - \alpha_{HW}}{6\Delta t}$$

**Gln. 22-3**

$$\Delta n = \frac{\alpha_T - \alpha_{HW}}{6\Delta t}$$

**Gln. 22-4**

wobei  $n$  in Umdrehungen pro Minute,  $\alpha$  in Grad Rotorwinkel,  $\Delta t$  in Sekunden.



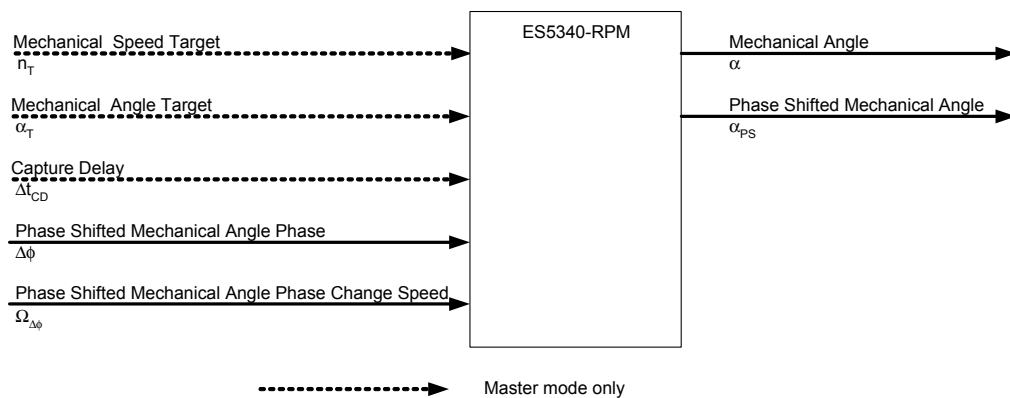
**Abb. 22-6** Masterbetrieb mit Winkeladaption

Die Eingangs- und Ausgangsgrößen bei der Simulation von Elektromotoren im Master-Betrieb mit Winkeladaption sind in Tab. 22-3 auf Seite 704 aufgelistet bzw. in Abb. 22-7 auf Seite 704 dargestellt.

Mit der Steuergröße  $\Delta t_{CD}$  kann zusätzlich noch eine Zeitverschiebung für die Übernahme der Zielgrößen  $n_T$  und  $\alpha_T$  durch die Hardware (bezogen auf den jeweiligen Beginn des Simulationszyklus) eingestellt werden. Das heißt, die Hardware übernimmt die Zielgrößen  $n_T$  und  $\alpha_T$  nicht zum Zeitpunkt  $t_v$  sondern zum Zeitpunkt  $t_v + \Delta t_{CD}$ .

Signal	Richtung	Beschreibung
$n_T$	Eingang	Ziel-Motordrehzahl in Umdrehungen pro Minute
$\alpha_T$	Eingang	Zielwert des mechanischen Winkels zu Beginn des nächsten Simulationszyklus in °RW
$\Delta t_{CD}$	Eingang	Zeitverschiebung betreffend die Übernahme von Ziel-Motordrehzahl und Zielwinkel
$\Delta\phi$	Eingang	Phasenlage des phasenverschobenen mechanischen Winkels in °RW.
$\Omega_{\Delta\phi}$	Eingang	Nachführungsgeschwindigkeit in °RW pro Sekunde des phasenverschobenen mechanischen Winkels bei einer Änderung des Sollwerts
$\alpha$	Ausgang	Rotorposition bzw. mechanischer Winkel in °RW
$\alpha_{PS}$	Ausgang	Phasenverschobenen mechanischer Winkel in °RW

**Tab. 22-3** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Elektromotoren im Master-Betrieb mit Winkeladaption.



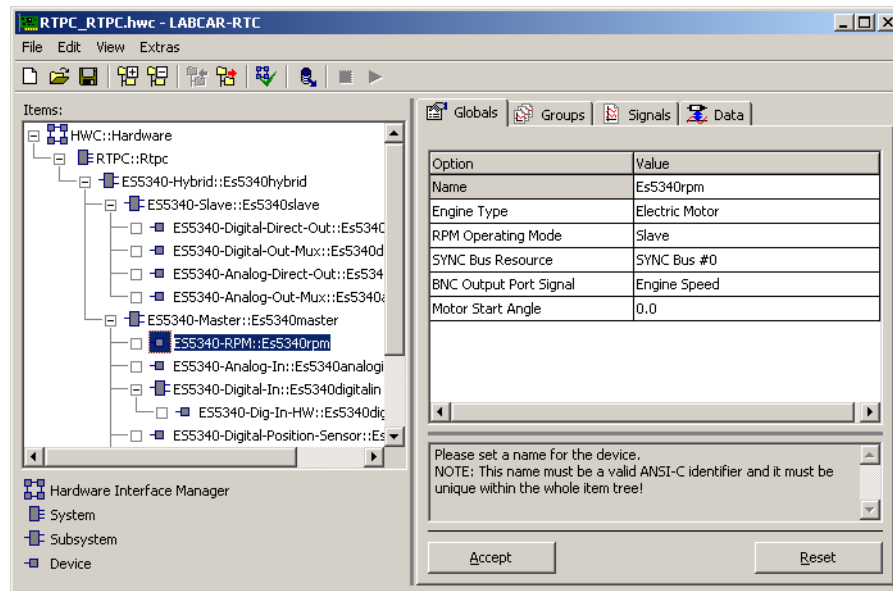
**Abb. 22-7** Steuer- und Statussignale der Drehzahleinheit bei der Simulation von Elektromotoren im Master-Betrieb mit Winkeladaption

### 22.2.1 ES5340-RPM – Drehzahleinheit

Das ES5340-RPM Device dient zu Konfiguration und Steuerung der ES5340-Drehzahleinheit.

## 22.2.2 Globals (ES5340-RPM)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-8** Die Registerkarte „Globals“ des ES5340-RPM Device

#### *Engine Type*

Auswahl des Motortyps.

#### *RPM Operating Mode*

Auswahl zwischen Master- oder Slave-Betrieb. Beim Elektromotor steht neben dem einfachen Master-Betrieb noch der Master-Betrieb mit Winkeladaption zur Auswahl.

#### *SYNC Bus Resource*

Das Winkeltaktsignal wird über Synchronisationsbus-Leitungen zwischen den Karten transportiert. Mit diesem Konfigurationsparameter wird die Busleitung ausgewählt, über die der Winkeltakt im Slave-Betrieb eingelesen wird oder im Master-Betrieb ausgegeben wird.

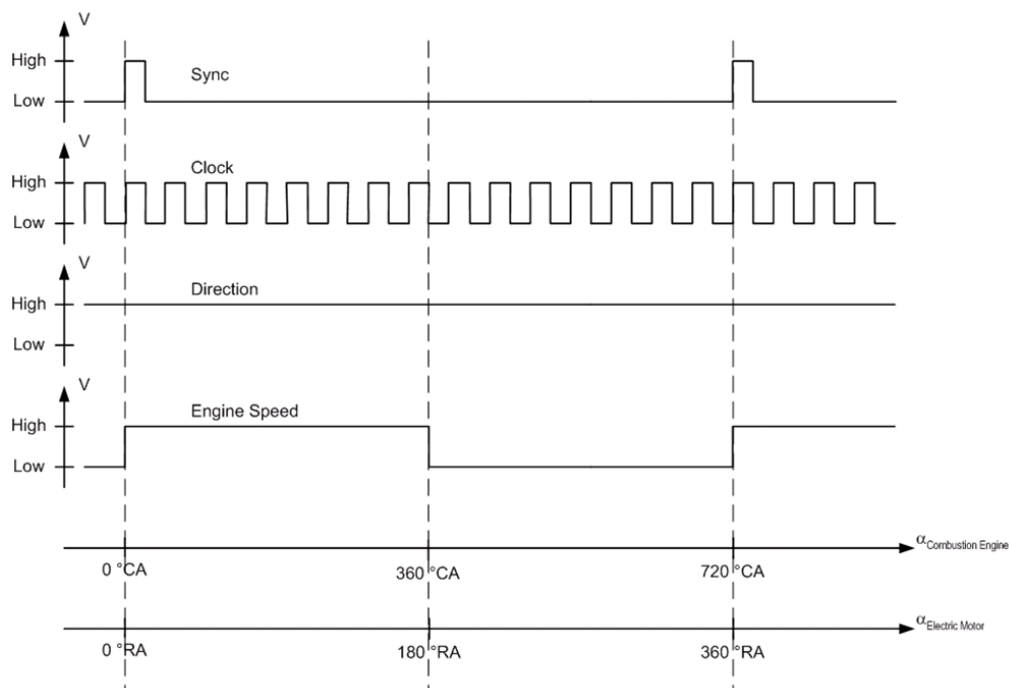
Wird die Drehzahleinheit auf Master-Betrieb konfiguriert, so steht zusätzlich noch die Option „None“ zur Auswahl – in diesem Fall wird ein Winkeltakt generiert und intern auf der Karte verwendet, jedoch nicht auf einen der beiden Synchronisationsbusse ausgegeben.

#### **Hinweis**

*Bitte beachten Sie, dass es auf einem Synchronisationsbus nur jeweils einen Winkeltakt-Master geben darf!*

#### *BNC Output Port Signal*

Mit diesem Auswahlfeld wird das Ausgabesignal der BNC-Buchse an der Frontplatte des ES5340.1/2 Electric Drive Simulation Board definiert. Zur Auswahl stehen die in Abb. 22-9 dargestellten Signale.



**Abb. 22-9** Ausgabesignal an BNC-Buchse  
*Motor Start Angle*

Der Winkel, bei dem der Motor nach der Modellinitialisierung losläuft.

Die folgende Tabelle fasst die Eigenschaften der einzelnen Konfigurationsparameter zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Engine Type	int	Nein	0: Verbrennungsmotor 1: E-Motor
RPM Operating Mode	int	Ja	0: Slave 1: Master 2: Master with Angle Adaption
SYNC Bus Resource	int	Ja	0: SYNC Bus #0 1: SYNC Bus #1 255: None (nur bei Masterbetrieb)
BNC Output Port Signal	int	Ja	0: Sync 1: Clock 2: Direction 3: Engine Speed
Motor Start Angle	float	Ja	0.0 °RW...360.0 °RW: E-Motor 0.0 °KW...720.0 °KW: Verbrennungsmotor

**Tab. 22-4** Die Registerkarte „Globals“ des ES5340-RPM Device

### 22.2.3 Groups (ES5340-RPM)

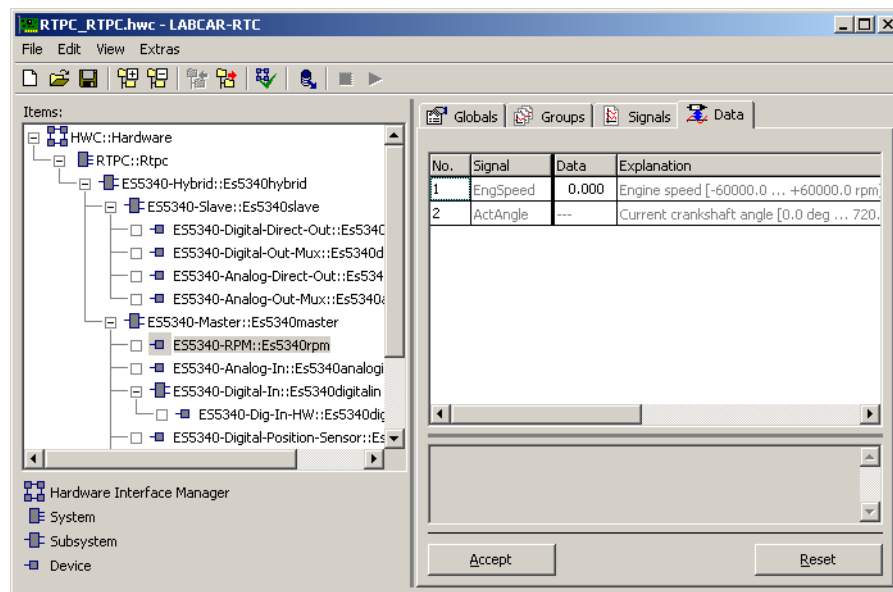
Das ES5340-RPM Device besitzt eine Signalgruppe „Outputs“ zur Steuerung der Drehzahleinheit und eine Signalgruppe „Inputs“ zum Lesen von aktuellen Statusinformationen der Drehzahleinheit.

#### Hinweis

Bei der Zuordnung einer Task des Echtzeitbetriebssystems zur Signalgruppe „Outputs“ muss folgende Einschränkung beachtet werden: Wird der Konfigurationsparameter „RPM Operating Mode“ auf „Master with Angle Adaption“ eingestellt und **gleichzeitig** ein „ES5340-Dig-In-Inverter-Meas“ Device zur Durchführung von Inverter-Messungen eingebunden, so muss der „Meas“-Signalgruppe des „ES5340-Dig-In-Inverter-Meas“ Device und der „Outputs“-Signalgruppe des „ES5340-RPM“ Device die gleiche Task zugeordnet werden.

### 22.2.4 Data (ES5340-RPM)

Die in der Registerkarte „Data“ angebotenen Signale zur Steuerung und Statusüberwachung der Drehzahleinheit sind vom gewählten Motortyp abhängig. Abb. 22-10 auf Seite 707 zeigt die RTIO-Signale für den Motortyp „Verbrennungsmotor“ – in Tab. 22-5 auf Seite 708 sind die Signaleigenschaften noch einmal zusammengefasst.

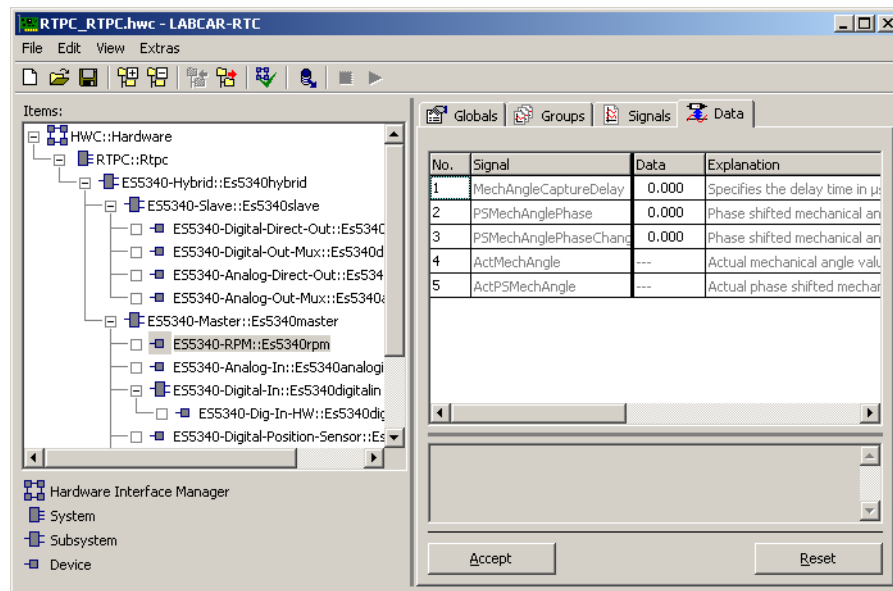


**Abb. 22-10** Die Registerkarte „Data“ des ES5340-RPM Device (Verbrennungsmotor)

Signal-name	Daten-typ	Anmerkungen
EngSpeed	float	Motordrehzahl in Umdrehungen pro Minute Wertebereich: -60000.0 rpm ... +60000.0 rpm Auflösung: 0.001 rpm Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.
ActAngle	float	Aktueller Kurbelwellenwinkel in °KW Wertebereich: 0.0 °KW ... 720.0 °KW Auflösung: 0.011 °KW

**Tab. 22-5** ES5340-RPM Device: RTIO-Signale (Verbrennungsmotor)

Die RTIO-Signale für den Motortyp „Elektromotor“ sind in Abb. 22-11 dargestellt und in Tab. 22-6 auf Seite 709 kurz erläutert.



**Abb. 22-11** Die Registerkarte „Data“ des ES5340-RPM Device (Elektromotor)



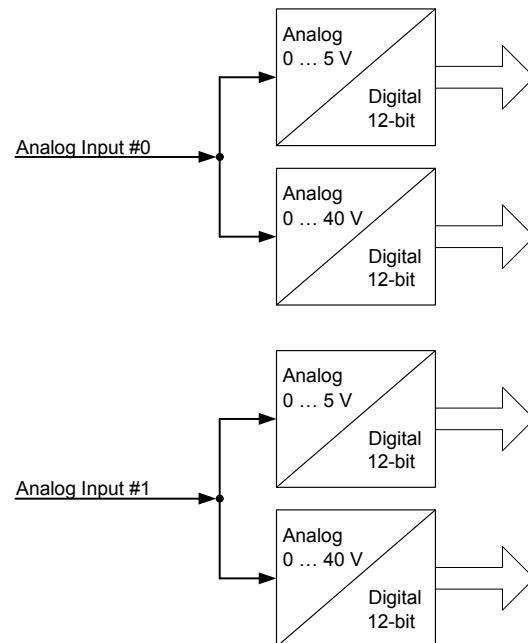
Signalname	Datentyp	Anmerkungen
MechSpeed	float	Motordrehzahl in Umdrehungen pro Minute Wertebereich: -30000.0 rpm ... +30000.0 rpm Auflösung: 0.001 rpm Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.
MechAngle	float	Zielwinkelvorgabe in Grad Wertebereich: 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW Dieses Signal ist nur dann von Bedeutung, wenn die Drehzahleinheit als „Master mit Winkeladaption“ betrieben wird.
MechAngle-CaptureDelay	float	Dieses Signal definiert die Zeit (in µs), um die die Zielwerte für Drehzahl und Winkel verzögert (bezogen auf das Triggerereignis) von der Hardware übernommen werden. Maximaler Wert: 1000.0 µs Dieses Signal ist nur dann von Bedeutung, wenn die Drehzahleinheit als „Master mit Winkeladaption“ betrieben wird.
PSMechAngle-Phase	float	Offset des phasenverschobenen mechanischen Winkels relativ zum Rotorwinkel Wertebereich: -360.0 °RW ... 360.0 °RW Auflösung: 0.0055 °RW
PSMechAngle-PhaseChange-Speed	float	Änderungsgeschwindigkeit des phasenverschobenen mechanischen Winkels relativ zum Rotorwinkel nach einem „PSMechAngle“-Sprung (in Grad pro Sekunde) Maximale Änderungsgeschwindigkeit: 1000.0 °RW/s Wird hier als Wert 0 eingetragen, so wird die Ziel-Phasenverschiebung sofort eingestellt.
ActMechAngle	float	Aktueller Rotorwinkel in Grad Wertebereich 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW
ActPSMechAngle	float	Aktueller Wert des zum Rotorwinkel phasenverschobenen Winkels in Grad Wertebereich 0.0 °RW ... 360 °RW Auflösung: 0.0055 °RW

**Tab. 22-6** ES5340-RPM Device: RTIO-Signale (Elektromotor)

## 22.3 ES5340-Analog-In – Analoge Eingänge

Das ES5340.1/2 Electric Drive Simulation Board besitzt vier analoge Eingänge mit einer Spannungsfestigkeit von  $\pm 60$  V – jeder dieser Eingänge wird auf einen Kanal eines 4-fach AD-Wandlers geführt.

Jeweils zwei AD-Wandler weisen einen Spannungsbereich von 0 bis 5 V auf, bzw. von 0 bis 40 V auf. Jeder Kanal eines AD-Wandlers wandelt mit einer Rate von 500000 Samples pro Sekunde.



**Abb. 22-12** Schematische Darstellung der Analog-Einheit.

22.3.1 ES5340-Analog-In – Analoge Eingänge

Das „ES5340-Analog-In“ Device dient zur Konfiguration der AD-Wandler und zur Erfassung der anliegenden Analogspannungen.

22.3.2 Signals (ES5340-Analog-In)

In der Registerkarte „Signals“ kann für jeden Kanal des Analogwandlers eine Mittelwertbildung über die erfassten AD-Werte konfiguriert werden.

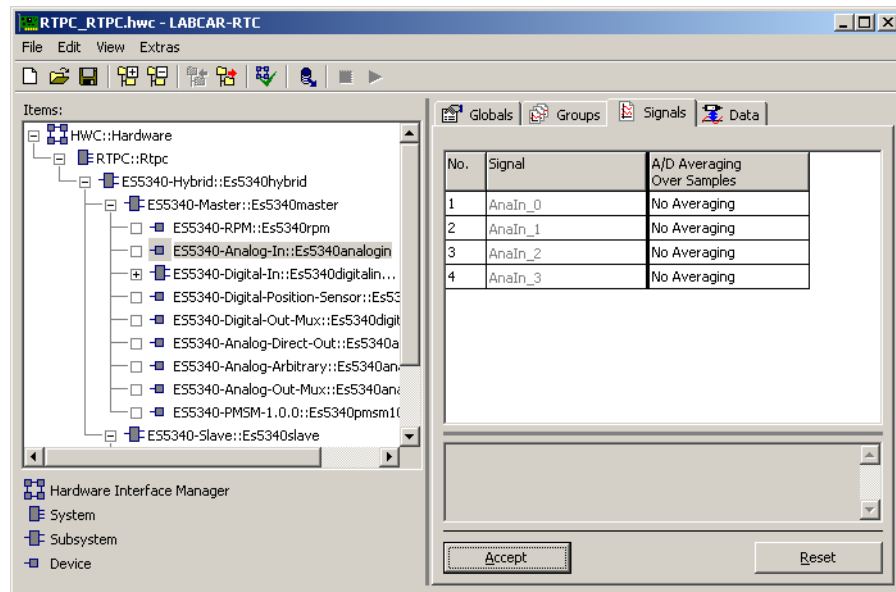


Abb. 22-13 Die Registerkarte „Signals“ des ES5340-Analog-In Device

Parameter	Daten- typ	Online editier- bar	Anmerkungen
A/D Averaging over Samples	int	Ja	0: Keine Mittelwertbildung 1: Mittelwertbildung über 2 Samples 2: Mittelwertbildung über 4 Samples 3: Mittelwertbildung über 8 Samples 4: Mittelwertbildung über 16 Samples 5: Mittelwertbildung über 32 Samples 6: Mittelwertbildung über 64 Samples 7: Mittelwertbildung über 128 Samples

Tab. 22-7 ES5340-Analog-In Device: Konfigurationsparameter der Registerkarte „Signals“

22.3.3 Data (ES5340-Analog-In)

In der Registerkarte „Data“ werden die erfassten (und optional gemittelten) Spannungswerte in Volt angezeigt.

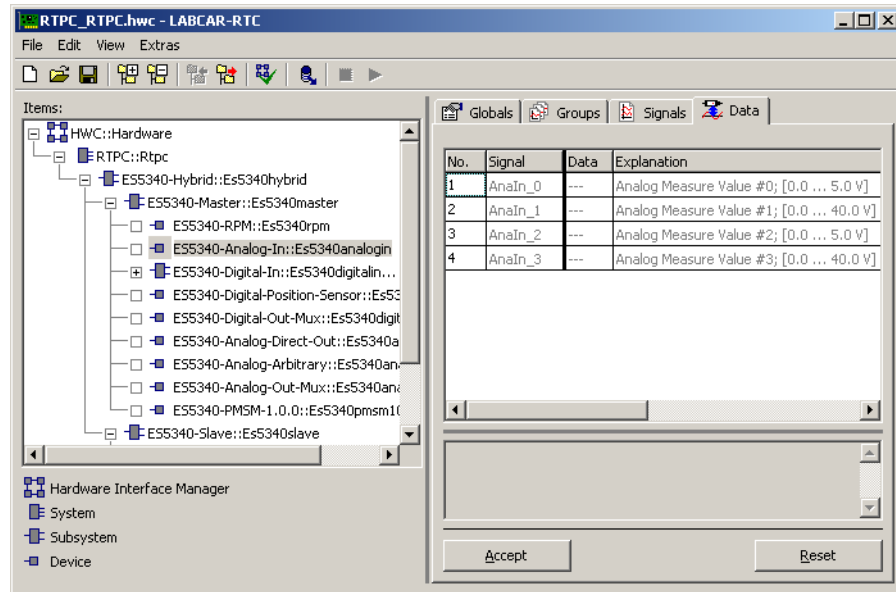


Abb. 22-14 Die Registerkarte „Data“ des ES5340-Analog-In Device

Signal-name	Daten-typ	Anmerkungen
Analn_0	float	Analoger Eingang 0 (misst im Bereich von 0.0...5.0 V)
Analn_1	float	Analoger Eingang 1 (misst im Bereich 0.0...40.0 V)
Analn_2	float	Analoger Eingang 2 (misst im Bereich von 0.0...5.0 V)
Analn_3	float	Analoger Eingang 3 (misst im Bereich 0.0...40.0 V)

Tab. 22-8 ES5340-Analog-In Device: Signale der Registerkarte „Data“

## 22.4 ES5340-Digital-In – Digitale Eingänge

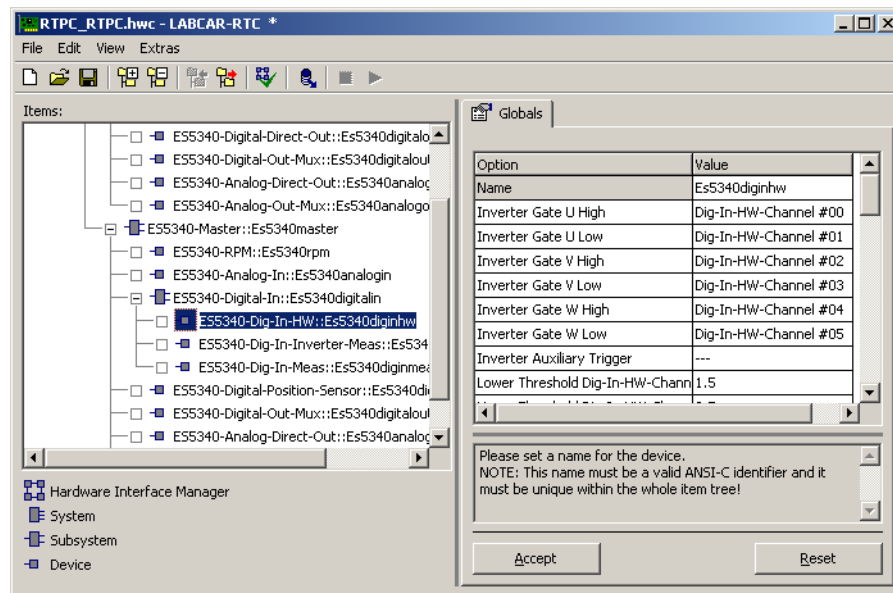
Dieses RTIO-Element dient nur zur Hierarchisierung. Es können keine weiteren Einstellungen vorgenommen werden.

### 22.4.1 ES5340-Dig-In-HW – Konfiguration der digitalen Eingänge

Hier können Sie die Verwendung der 20 digitalen Eingänge konfigurieren. Die Eingänge können zum Einen zur allgemeinen Erfassung zeitvariabler Signale (z.B. PWM-Signale), zum Anderen für spezielle E-Motor-Messfunktionen verwendet werden.

### 22.4.2 Globals (ES5340-Dig-In-Hw)

In diesem Abschnitt können Sie den Hardware-Eingangskanälen spezielle Inverterfunktionen zuweisen und die Schaltschwellen für die Eingänge festlegen.



**Abb. 22-15** Die Registerkarte „Globals“ des ES5340-Dig-In-Hw Device

#### *Inverter Gate {U, V, W} {High, Low}*

Diese Optionsfelder dienen der Zuordnung der Inverter-Ansteuersignale zu Hardware-Eingangskanälen. Es müssen die Kanäle eingestellt werden, die mit den entsprechenden Ausgängen des Hybrid-Steuergerätes verbunden sind. Diese Zuordnung wird bei der Invertersignalvermessung durch das ES5340-Dig-In-Inverter-Meas Device bzw. durch das FPGA-Modell der E-Motor-Simulation (ES5340-PMSM Device) benötigt.

- Defaultwerte: Dig-In-HW Kanäle 0...5

#### *Inverter Auxiliary Trigger*

Dieses Optionsfeld dient der Zuordnung des Inverter-Hilfstriggersignales zu einem Hardware-Eingangskanal. Das Hilfssignal muss entweder in der Mitte oder am Beginn der PWM-Phase eine steigende oder fallende Flanke erzeugen.

Diese Einstellung wird nur für die entsprechenden Betriebsarten des ES5340-Dig-In-Inverter-Meas Devices benötigt.

- Defaultwert: --- (keine Zuordnung)

#### *Lower Threshold Dig-In HW Channel #<xx>*

---

Diese Optionsfelder beschreiben den Spannungswert am Eingang #<xx>, bei dessen Unterschreiten ein Logikübergang von High → Low ausgelöst wird.

- Defaultwert: 1,5 Volt
- Einstellbereich: 0,0 ... 10,0 Volt

#### *Upper Threshold Dig-In HW Channel #<xx>*

---

Diese Optionsfelder beschreiben den Spannungswert am Eingang #<xx>, bei dessen Überschreiten ein Logikübergang von Low → High ausgelöst wird.

- Defaultwert: 3,5 Volt
- Einstellbereich: 0,0 ... 10,0 Volt

Wird für die untere Schwelle eines Kanals ein höherer Wert als für die obere angegeben, so werden die beiden Schwellen automatisch vertauscht.

### 22.4.3 ES5340-Dig-In-Meas – Nicht synchronisierte Zeit-/Frequenzmessungen

---

Mit diesem RTIO-Device können Sie Zeit-, Frequenz- und Tastverhältnismessungen an periodischen Signalen durchführen oder den Binärwert eines Eingangs erfassen. In einem RTIO-Device können maximal zehn Messkanäle konfiguriert werden und es können maximal vier dieser RTIO-Devices in ein ES5340-Digital-In Subsystem eingefügt werden. Insgesamt stehen maximal 40 Softwarekanäle zur Verfügung.

Die einzelnen Messungen sind voneinander unabhängig und auch prinzipiell unabhängig von anderweitiger Verwendung der Eingänge (Invertersignalvermessung).

#### *Prinzip der Messwerterfassung und Berechnung*

---

Die Messwerterfassung erfolgt auf dem ES5340.1/2 Electric Drive Simulation Board – hierbei werden die 20 Eingangssignale entsprechend den (im ES5340-Dig-In-HW Device) vorgegebenen Schaltschwellen digitalisiert.

Für jeden sich ergebenden Flankenwechsel eines Hardwarekanals wird ein Datenwort generiert, welches aus einem 31-Bit Zeitstempel eines lokalen Zählers und einem Bit für die Flankeninformation besteht. Der Zeitstempel hat eine Auflösung von 8 ns (125 MHz-Taktung), so dass eine Zeit von max. 17,28 s dargestellt werden kann. Eine „1“ als Flankenbit signalisiert eine positive Flanke, d.h. der Pegel ist danach „1“.

Die Berechnung der Messwerte erfolgt im Treiber des Real-Time PC. Hierzu werden für jeden Hardwarekanal die Daten der jeweils letzten vier Flankenwechsel mittels des zyklischen DMA-Transfers in den Arbeitsspeicher des Real-Time PC übertragen. Bei hinreichend niedriger DMA-Transferzykluszeit (Defaultwert: 100 µs) stehen im Treiber alle Informationen zur Berechnung der Messverfahren mit niedriger Latenz zur Verfügung.

Falls bei einem Hardwarekanal seit Beginn der Datenerfassung noch keine vier Flankenwechsel aufgetreten sind, so wird beim Start der Datenerfassung für alle vier Flanken das gleiche Datenwort generiert. Dieses besteht aus dem aktuellen Zeitstempel und einer Flankeninformation, welche den aktuellen Pegel widerspiegelt.

Diese einfachen Kodierungsregeln ermöglichen es, gültige von ungültigen Daten unterscheiden:

- Der aktuelle Pegel kann auch ohne Flankenwechsel ermittelt werden.
- Ungültige Flankendaten können daran erkannt werden, dass das Flankenbit sich gegenüber dem vorigen Datum nicht geändert hat.

Die Messverfahren zur Rückgabe des Pegels liefern daher immer gültige Daten. Bei allen anderen Messverfahren müssen ausreichend Flanken vorhanden sein, was vom Verfahren und dem aktuellen Zustand abhängt.

Kann kein Messwert berechnet werden, so wird als Wert -8888.0 zurückgegeben (sofern keine Zeitüberschreitung vorliegt bzw. überwacht wird).

#### 22.4.4 Die Messverfahren

Im Folgenden finden Sie eine Beschreibung der möglichen Verfahren zur Vermessung digitaler Signale.

##### **Hinweis**

*Generell dürfen die gemessenen Zeiten zwischen zwei Flankenereignissen den Abbildungsbereich des 31-Bit-Zählers (17,28 s) nicht überschreiten. Ist dies der Fall, kommt es zu einem Zählerüberlauf und damit zu falschen Messergebnissen.*

##### *Frequenz- und Periodendauermessung*

Die Messverfahren

- Cycle Time --/-- [ $\mu$ s]
- Cycle Time --\-- [ $\mu$ s]
- Frequency --/-- [Hz]
- Frequency --\-- [Hz]

liefern die Periodendauer bzw. Frequenz des Signals am jeweiligen Eingang, gemessen zwischen zwei steigenden Flanken (--/--) bzw. zwischen zwei fallenden Flanken (--\--).

##### *High/Low Zeitmessung*

Die Messverfahren

- High Time [ $\mu$ s]
- Low Time [ $\mu$ s]

liefern die Zeit zwischen letzter steigender und darauffolgender fallender Flanke (High Time) bzw. letzter fallender und steigender Flanke (Low Time).

##### *Tastverhältnismessungen*

Die Messverfahren

- Duty Cycle L/(L+H) --/--
- Duty Cycle L/(L+H) --\--
- Duty Cycle H/(L+H) --/--
- Duty Cycle H/(L+H) --\--

liefern das Verhältnis der Low- bzw. High-Zeit, bezogen auf die Periodendauer jeweils von steigender zu steigender Flanke (--/--) bzw. von fallender zu fallender Flanke (--\--).

### *Pegelmessung*

---

Die Messverfahren

- Level (Active High)
- Level (Active Low)

liefern den Pegel eines PWM-Eingangs in Form einer aktiv-/inaktiv-Information. „0“ bedeutet das Signal ist inaktiv, „1“ bedeutet das Signal ist aktiv.

### *Verhalten bei Zeitüberschreitung*

---

Für jeden Messkanal kann eine Überwachung von Zeitüberschreitungen eingestellt werden. Bei aktivierter Überwachung können der Default-Rückgabewert und das Überwachungsintervall verändert werden. Der Zustand „Zeitüberschreitung“ ist erreicht, wenn die Differenz zwischen der aktuellen Zeit einer DMA-Übertragung und der Zeit des letzten Flankenwechsels des Kanals die vorgegebene Intervallzeit überschreitet. Maximal kann eine Intervallzeit von 10 s angegeben werden. Bis zu diesem Zeitpunkt wird der letzte Messwert übertragen (bzw. -8888.0, falls noch keine gültiger Wert vorhanden ist).

Bei Erreichen der Zeitüberschreitung ist der Messwert von der Betriebsart abhängig. Prinzipiell sind drei Betriebsarten zur Überwachung bei Zeitüberschreitung möglich:

- Inactive  
Keine Überwachung – es wird der letzte gemessene Wert (bzw. -8888.0, falls noch kein Wert vorliegt) zurückgegeben. Das Signal „Tout\_<xx>“ hat immer den Wert „2 = not used“.
- Intvl Predef  
Im Falle einer Zeitüberschreitung das Signal „Tout\_<xx>“ auf „1 = Timeout“ gesetzt und der vordefinierte Wert zurückgegeben.



- Intvl InpDep

Das Signal „Tout\_<xx>“ wird auf „1 = Timeout“ gesetzt und ein vom Messverfahren und ggf. vom Eingangspegel abhängiger Wert zurückgegeben (siehe Tab. 22-9).

Messverfahren	Wert (Input Level Dependent)
No Measurement	-8888.0
High Time [ $\mu$ s]	0
Low Time [ $\mu$ s]	0
Cycle Time --/-- [ $\mu$ s]	Zeit seit letzter Flanke [ $\mu$ s]
Cycle Time --\-- [ $\mu$ s]	Zeit seit letzter Flanke [ $\mu$ s]
Frequency --/-- [Hz]	Kehrwert der Zeit seit letzter Flanke [Hz]
Frequency --\-- [Hz]	Kehrwert der Zeit seit letzter Flanke [Hz]
Duty Cycle L/(L+H) --/--	1.0 bei Low-Pegel, 0.0 bei High-Pegel am Eingang
Duty Cycle L/(L+H) --\--	1.0 bei Low-Pegel, 0.0 bei High-Pegel am Eingang
Duty Cycle H/(L+H) --/--	0.0 bei Low-Pegel, 1.0 bei High-Pegel am Eingang
Duty Cycle H/(L+H) --\--	0.0 bei Low-Pegel, 1.0 bei High-Pegel am Eingang
Level (Active High)	0
Level (Active Low)	0

**Tab. 22-9** Rückgabewerte bei Zeitüberschreitung (Betriebsart „Intvl InpDep“)

Bei den Werten, die bei Messungen von Frequenz- und Periodendauer übermittelt werden, ist der Überlauf des 31-Bit breiten Zeitzählers berücksichtigt, d.h. diese können im Rahmen der 64-Bit-Gleitpunktdarstellung beliebig kleine bzw. große Werte (Frequenz bzw. Zeit) annehmen.

Der Zustand der Zeitüberschreitung wird verlassen, wenn das 32-Bit-Datenwort der neuesten Flanke (Zeitstempel + Flankeninformation) sich gegenüber jenem ändert, das bei Eintritt der Zeitüberschreitung übermittelt wurde. Dadurch können auch Zeitüberschreitungen dargestellt werden, welche den Überlaufwert des 31-Bit Zeitzählers (17,28 s) überschreiten. Das Signal „Tout\_<xx>“ wird auf den Wert „0 = no Timeout“ gesetzt.

22.4.5 Groups (ES5340-Dig-In-Meas)

Hier können Sie der Signalgruppe eine Task zuweisen – alle Messkanäle dieser Signalgruppe werden in dieser Task berechnet.

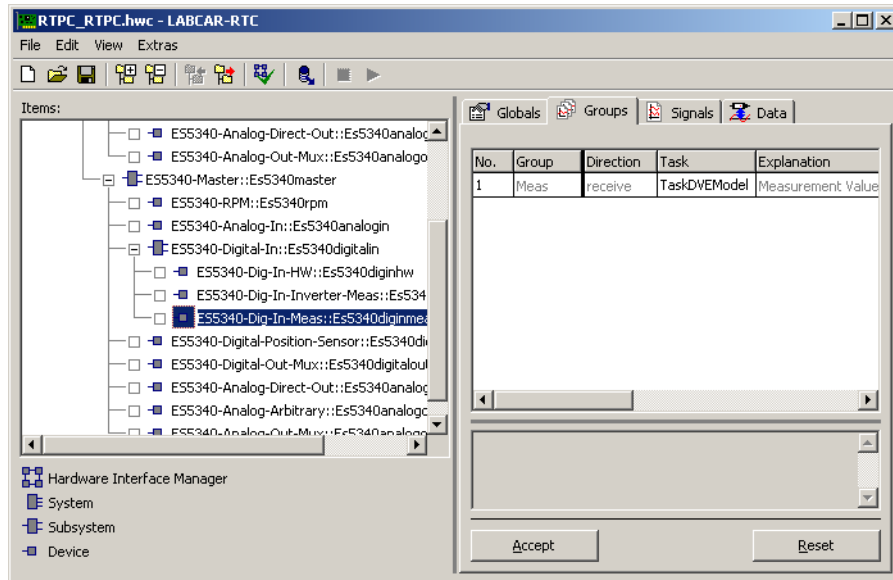


Abb. 22-16 Die Registerkarte „Groups“ des ES5340-Dig-In-Meas Device

22.4.6 Signals (ES5340-Dig-In-Meas)

Hier können Sie die Eigenschaften der Messkanäle konfigurieren. Die Konfigurationsparameter gelten für alle Messwerte „MeasVal\_<xx>“. Alle Parameter sind online konfigurierbar.

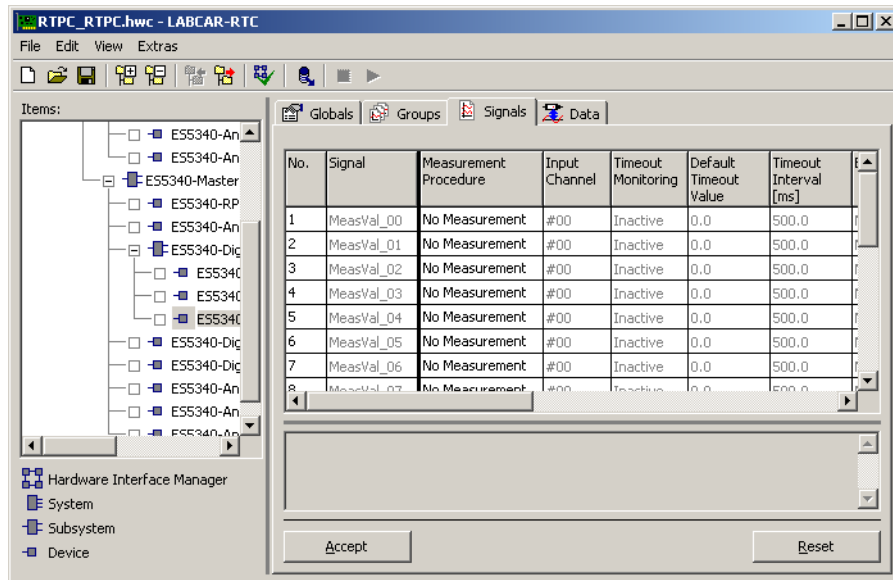


Abb. 22-17 Die Registerkarte „Signals“ des ES5340-Dig-In-Meas Device

### *Measurement Procedure*

---

Das Messverfahren (siehe „Die Messverfahren“ auf Seite 715). Bei Auswahl von „No Measurement“ wird kein Wert berechnet und -8888.0 übermittelt.

- Defaultwert: No Measurement

### *Input Channel*

---

Der Hardware-Eingangskanal, auf den sich das Messverfahren bezieht.

- Defaultwert: #00
- Einstellbereich: #00, #01, ..., #19

### *Timeout Monitoring*

---

Die Betriebsart für die Überwachung bei Zeitüberschreitung (siehe „Verhalten bei Zeitüberschreitung“ auf Seite 716).

- Defaultwert: Inactive

### *Default Timeout Value*

---

Rückgabewert bei der Timeout-Betriebsart „Intvl InpDep“. Dieser Wert ist nur dann editierbar, wenn für den Messkanal diese Timeout-Betriebsart eingestellt ist.

- Defaultwert: 0.0
- Einstellbereich: 64-Bit Gleitkommazahl

### *Timeout Interval [ms]*

---

Intervall für Überwachung der Zeitüberschreitung. Dieser Wert ist nur in den Timeout-Betriebsarten „Intvl Predef“ und „Intvl InpDef“ editierbar.

- Defaultwert: 500.0 [ms]
- Einstellbereich: 0.1 ... 10000.0 [ms]

22.4.7 Data (ES5340-Dig-In-Meas Device)

Hier werden die Messergebnisse und die zugehörigen Timeout-Flags angezeigt.

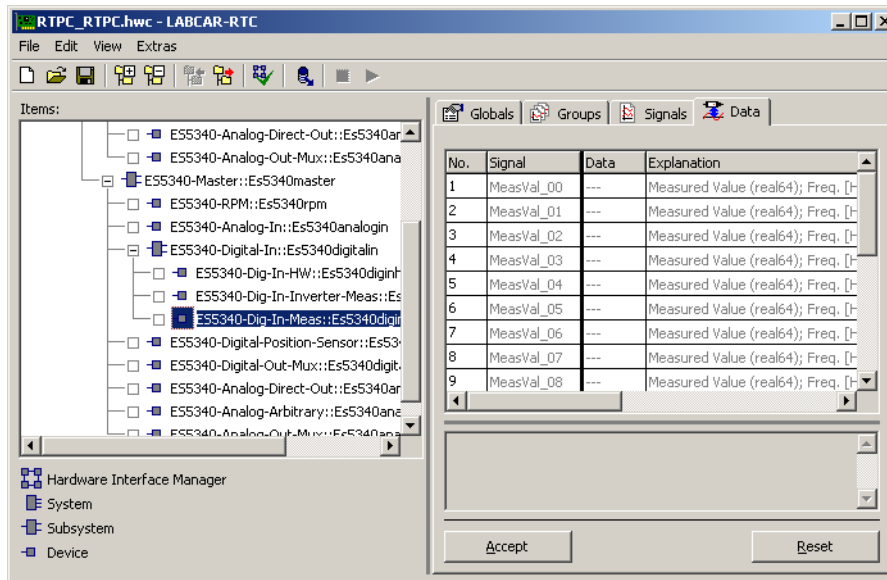


Abb. 22-18 Die Registerkarte „Data“ des ES5340-Dig-In-Meas Device

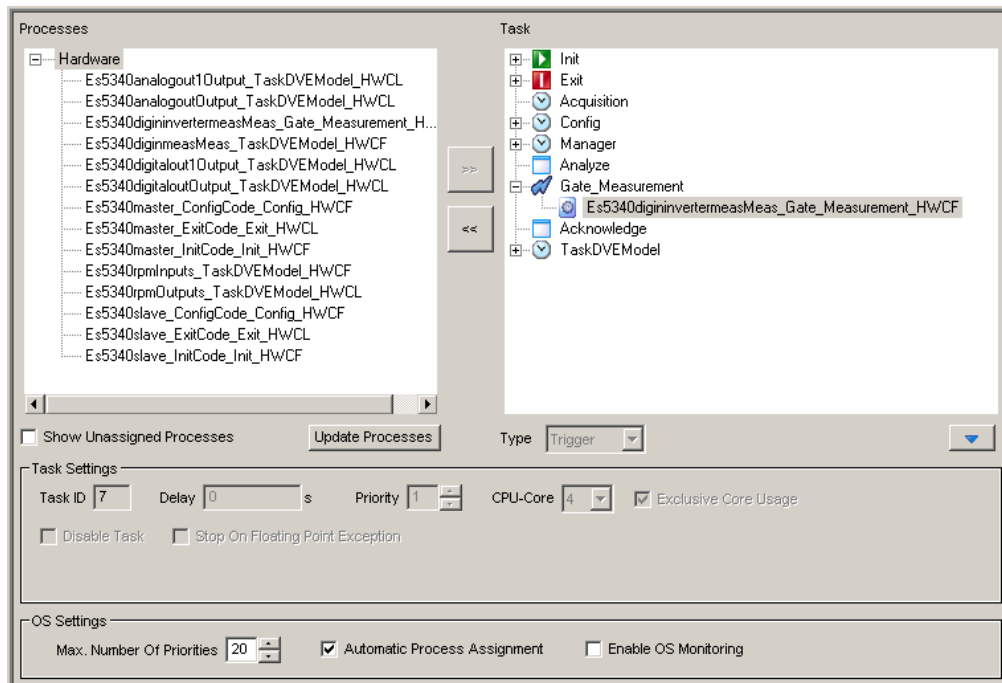
RTIO-Signal	Daten-typ	Signal-gruppe	Beschreibung
MeasVal_00 - MeasVal_09	real64	Meas	Messergebnis, je nach Messverfahren Zeit [µs], Frequenz [Hz], Verhältniszahl, - 8888.0 für ungültigen Messwert oder Timeout-Defaultwert
Tout_00 - Tout_09	real64	Meas	Signalisiert Vorgabe bzw. Zustand der Zeitüberschreitung 0: kein Timeout, Überwachung bei Zeitüberschreitung aktiv 1: im Timeout, Überwachung bei Zeitüberschreitung aktiv 2: keine Überwachung bei Zeitüberschreitung

Tab. 22-10 Die RTIO-Signale der Signalgruppe „Meas“

22.4.8 ES5340-Dig-In-Inverter-Meas – Vermessung der Inverter-Ansteuersignale

Mit diesem RTIO-Device können Sie die Ansteuersignale für eine permanent-erregte Synchronmaschine vermessen und eine Task mit niedriger Latenz aktivieren, wenn neue Messdaten zur Verfügung stehen. Die eigentliche Messung erfolgt in der ES5340.1/2. Die Messwerte werden mittels DMA-Transfer in den Arbeitsspeicher des Real-Time PC übertragen. Die Aktivierung einer Task erfolgt durch Abfrage eines Zählers in den übertragenen Messdaten durch einen Software-Scheduler.

Hierzu ist es erforderlich, dass im Abschnitt „OS-Configuration“ von LABCAR-IP einen Task von Typ „Trigger“ angelegt wurde. Weiterhin muss die Option „Exclusive Core Usage“ aktiviert sein und diese Task exklusiv mit einem der vorhandenen CPU-Kerne verknüpft werden.



**Abb. 22-19** Einstellung der Trigger Task in der OS Configuration in LABCAR-IP

Die Verknüpfung der Task mit der Invertervermessung erfolgt durch Zuweisung der „Meas“ Signalgruppe in der Registerkarte „Groups“.

Nur dadurch wird die erforderliche niedere Latenz erreicht – dafür erforderlich ist ein Real-Time PC mit einem Prozessor mit mindestens drei physikalischen Kernen.

Die Zuordnung der Inverteransteuersignale zu Hardware-Eingangskanälen erfolgt im ES5340-Dig-In-HW Device. Fehlt dieses Device, wird beim Prüfen der Hardwarekonfiguration im RTIO-Editor eine Fehlermeldung erzeugt und es wird kein Code generiert.

#### 22.4.9 Globals (ES5340-Dig-In-Inverter-Meas)

Hier können Sie Vorgaben zum Trigger Mode und zur PWM-Periode machen.

##### *Trigger Mode*

In diesem Auswahlfeld wird die Art der Triggerung festgelegt. Sie erfolgt entweder durch Regeneration des Taktes aus den Ansteuersignalen oder mittels Hilfstigger.

Die Modi mit Hilfstigger setzen voraus, dass im ES5340-Dig-In-HW Device das Hilfstiggersignal zu einem Hardwareeingang zugewiesen wurde. Ist dies nicht der Fall, so erfolgt eine Fehlermeldung und die Einstellung eines Triggermodus mittels Hilfstigger ist nicht möglich.

### Nominal PWM Period of Center Aligned PWM [usec]

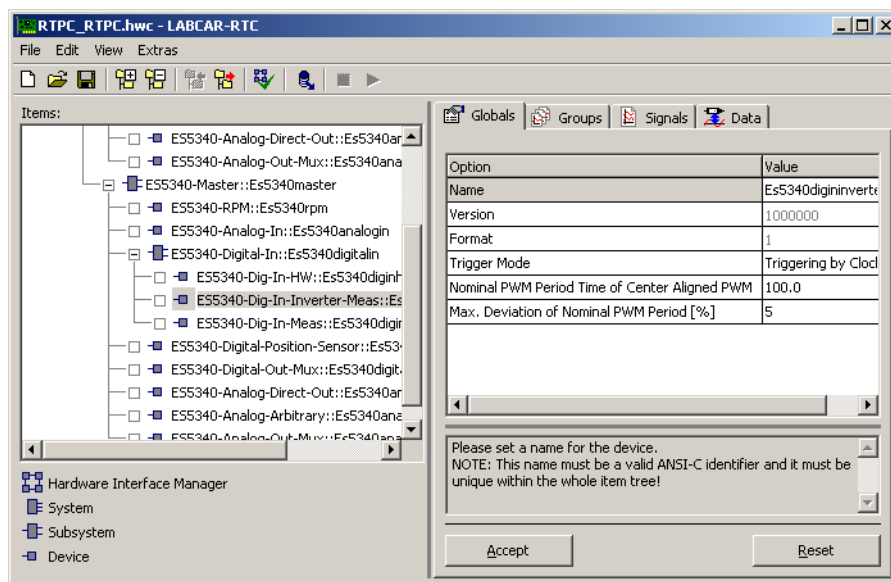
Hier wird der Sollwert der Periodendauer des PWM-Signales festgelegt. Dieser wird zur Berechnung der Tastverhältnisse benötigt, damit diese bereits zur Mitte eines PWM-Pulses erfolgen kann.

- Defaultwert: 100.0  $\mu$ s
- Einstellbereich: 0...500.0  $\mu$ s

### Nominal PWM Period Deviation of Center Aligned PWM [%]

Diese Option beschreibt die max. zulässige Abweichung der PWM-Periode vom Sollwert (in Prozent). Die Einstellung ist nur in der Ansichtsoption „Show All“ sicht- und veränderbar.

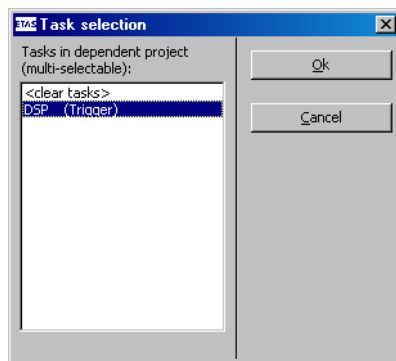
- Defaultwert: 5%
- Einstellbereich: 0%...100%



**Abb. 22-20** Die Registerkarte „Globals“ des ES5340-Dig-In-Inverter-Meas Device (Ansichtsoption „Show All“)

#### 22.4.10 Groups (ES5340-Dig-In-Inverter-Meas)

In diesem Abschnitt können Sie Taskzuweisungen für die Invertervermessung (Signalgruppe „Meas“) und ggf. einer weiteren Steuersignalgruppe („Ctrl“) vornehmen. Bei der Invertervermessung ist dies auch gleichzeitig die Task, welche durch die neuen Messdaten bzw. eine neue DMA-Übertragung der mittenzentrierten PWM-Vermessung getriggert wird.



**Abb. 22-21** Auswahldialog für „Trigger“ Task

Es werden nur Tasks vom Typ „Trigger“ zur Auswahl angezeigt, welche mit dem Attribut „Exclusive Core Usage“ versehen sind.

Falls hier keine Task zur Auswahl steht, gehen Sie wie folgt vor:

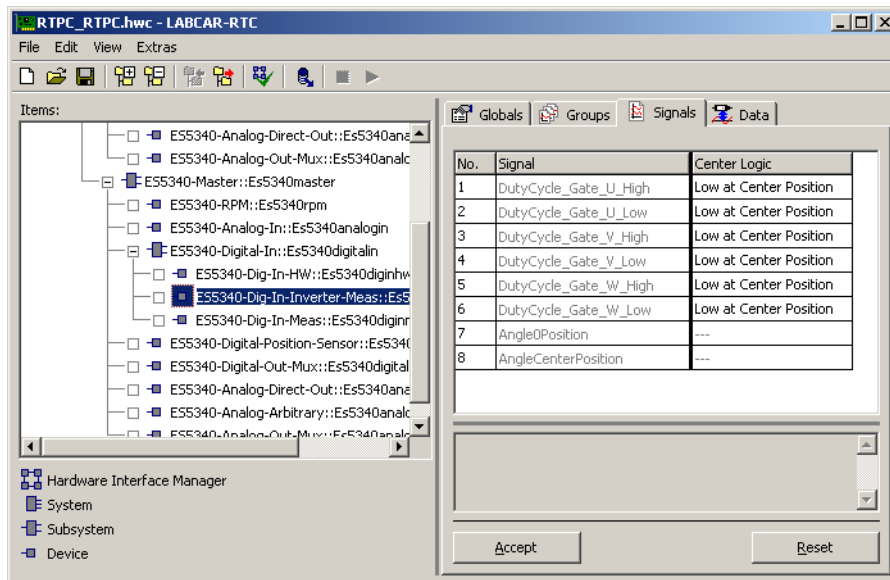
#### **Task vom Typ „Trigger“ erstellen**

- Speichern Sie bis dato gemachte Änderungen im LABCAR-RTC Editor und schließen Sie diesen.
- Gehen Sie in LABCAR-IP in das Register „OS Configuration“.
- Fügen Sie eine neue Task hinzu bzw. ändern Sie eine bestehende wie folgt:
  - Type: „Trigger“
  - erweiterten Task-Einstellungen: „Exclusive Core Usage“ (mit Angabe eines CPU-Kerns)
- Speichern Sie die Änderungen in LABCAR-IP und öffnen Sie den LABCAR-RTC Editor erneut.
- Weisen Sie der Signalgruppe die neue bzw. geänderte Task zu.

Weiterhin wird bei den Triggermodi mit Hilfstrigger eine Sendesignalgruppe „Ctrl“ angelegt. Diese dient zur Einstellung zusätzlicher Verzögerungszeiten für die sechs Inverteransteuersignale (Gate U,V,W / High, Low) und das Hilfstriggersignal.

## 22.4.11 Signals (ES5340-Dig-In-Inverter-Meas)

In dieser Registerkarte können Sie Einstellungen für den Triggermodus „Triggering by Clock Recovery from PWM Signals“ vornehmen.



**Abb. 22-22** Die Registerkarte „Signals“ des ES5340-Dig-In-Inv-Meas Device  
*Center Logic*

Diese Einstellungen legen die Polarität der sechs Inverteransteuersignale durch Angabe des Pegels in der Mitte des jeweiligen PWM-Ansteuerimpulses fest. Die Parameter sind für die Triggermodi mit Hilfstriggersignal ohne Bedeutung und nicht sichtbar.

- Defaultwert: „Low at Center Position“
- Einstellbereich: „Low at Center Position“, „High at Center Position“



## 22.4.12 Data (ES5340-Dig-In-Inverter-Meas)

Hier werden das Messergebnis der Tastverhältnisse der Inverteransteuersignale und einige Winkelpositionen angezeigt. Bei den Triggermodi mit Hilfstriggersignal sind als Ausgangsgrößen zusätzlich die Signale zur Vorgabe der Verzögerungszeiten der Inverteransteuersignale vorhanden.

RTIO-Signal	Datentyp	Signalgruppe	Beschreibung
DutyCyle_Gate_U_High DutyCyle_Gate_U_Low DutyCyle_Gate_V_High DutyCyle_Gate_V_Low DutyCyle_Gate_W_High DutyCyle_Gate_W_Low	real64*	Meas	Messergebnis der Tastverhältnisse der Inverteransteuersignale (Duty Cycle H/(H+L)) als Verhältniszahl [0.0 ... 1.0]
Angle0Position	real64	Meas	Startwinkelposition [0.0 °RW ... 360.0 °RW]
AngleCenterPosition	real64	Meas	Mittenwinkelposition [0.0 °RW ... 360.0 °RW]
Delay_Gate_U_High Delay_Gate_U_Low Delay_Gate_V_High Delay_Gate_V_Low Delay_Gate_W_High Delay_Gate_W_Low Delay_Aux_Trigger	real64**	Ctrl	

\* von 16-Bit Registern abgeleitet

\*\* abgebildet auf 11-Bit Register

**Tab. 22-11** Die RTIO-Signale der Signalgruppen „Meas“ und „Ctrl“

## 22.5 ES5340-Digital-Out – Digitale Ausgänge

Das ES5340-Digital-Out RTIO-Element kann in vier verschiedenen Betriebsarten verwendet werden.

Mode	Beschreibung
Digital Direct Out	Ansteuerung eines digitalen Ausgangs (siehe „ES5340-Digital-Direct-Out“ auf Seite 726)
Digital Position Sensor	Simulation eines Lagesensors des Typs „Digital Position Sensor“ (siehe „ES5340-Digital-Position-Sensor“ auf Seite 729)
Digital Arbitrary	Ausgabe einer frei konfigurierbaren digitalen Wellenform (siehe „ES5340-Digital-Arbitrary“ auf Seite 733)
PWM Output	Ausgabe von PWM-Signalen (siehe „ES5340-PWM-Output“ auf Seite 736)

**Tab. 22-12** Betriebsarten des ES5340-Digital-Out RTIO-Elements

### Hinweis

*Nähere Informationen zu den Betriebsarten finden Sie im Benutzerhandbuch zum ES5340.1/2 Electric Drive Simulation Board.*

Die Konfiguration des RTIO-Elements ist für jeden Ausgabemodus unterschiedlich – abhängig davon ändert sich auch das Aussehen des RTIO-Elements ES5340-Digital-Out.

Für die Ausgabe von Digitalwerten verwendet die ES5340.1/2 ein zweistufiges Verfahren:

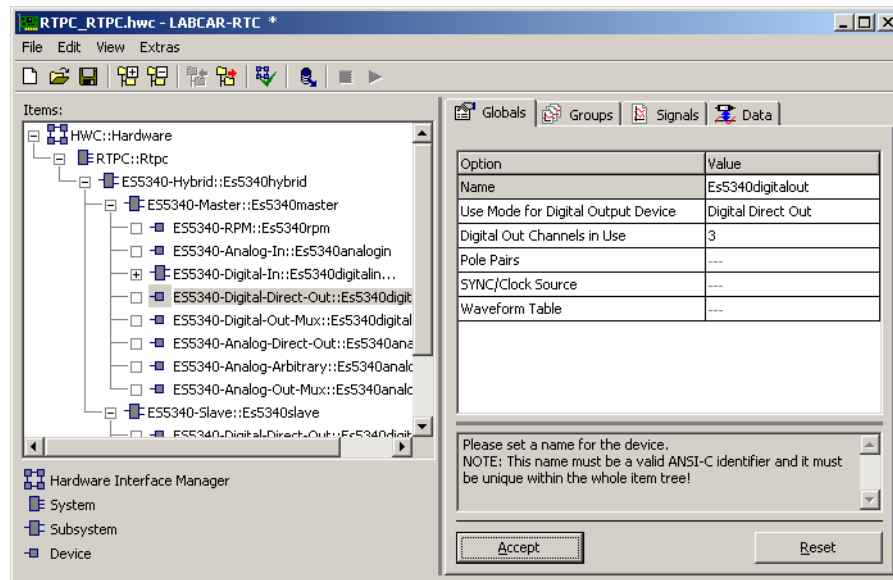
1. Die Ausgabewerte werden über das ES5340-Digital-Out RTIO-Element konfiguriert. Hier können mehr Signale konfiguriert werden als Ausgangskanäle auf der ES5340.1/2 zur Verfügung stehen (max. je 8 Kanäle im Modus „Digital-Direct-Out“ und 8 Kanäle für die anderen digitalen Ausgabemodi).
2. Die Auswahl der Werte, die tatsächlich ausgegeben werden, erfolgt über das ES5340-Digital-Out-Mux RTIO-Element (siehe „ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer“ auf Seite 738).

### 22.5.1 ES5340-Digital-Direct-Out

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Digital-Out RTIO-Elements mit dem Ausgabemodus „Digital Direct Out“. In dieser Betriebsart können ein bis acht digitale Ausgänge direkt angesteuert werden.

### 22.5.2 Globals (ES5340-Digital-Direct-Out)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-23** Die Registerkarte „Globals“ des ES5340-Digital-Direct-Out RTIO-Elements

*Use Mode for Digital Output Device*

Auswahl des Ausgabemodus.

*Digital Out Channels in Use*

Anzahl der verwendeten digitalen Ausgabekanäle (max. 8).

### 22.5.3 Groups (ES5340-Digital-Direct-Out)

Das ES5340-Digital-Direct-Out-RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

### 22.5.4 Signals (ES5340-Digital-Direct-Out)

Hier sind keine weiteren Einstellungen notwendig.

22.5.5 Data (ES5340-Digital-Direct-Out)

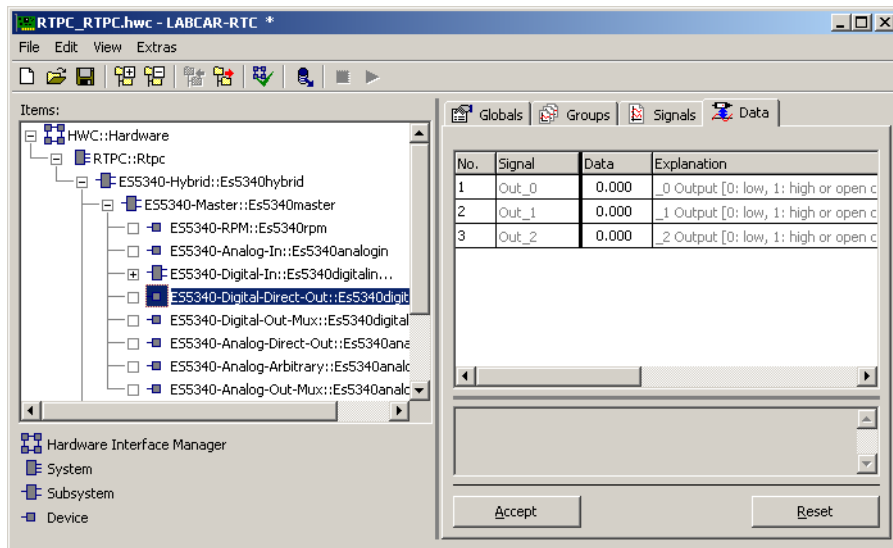


Abb. 22-24 Die Registerkarte „Data“ ES5340-Digital-Direct-Out RTIO-Elements

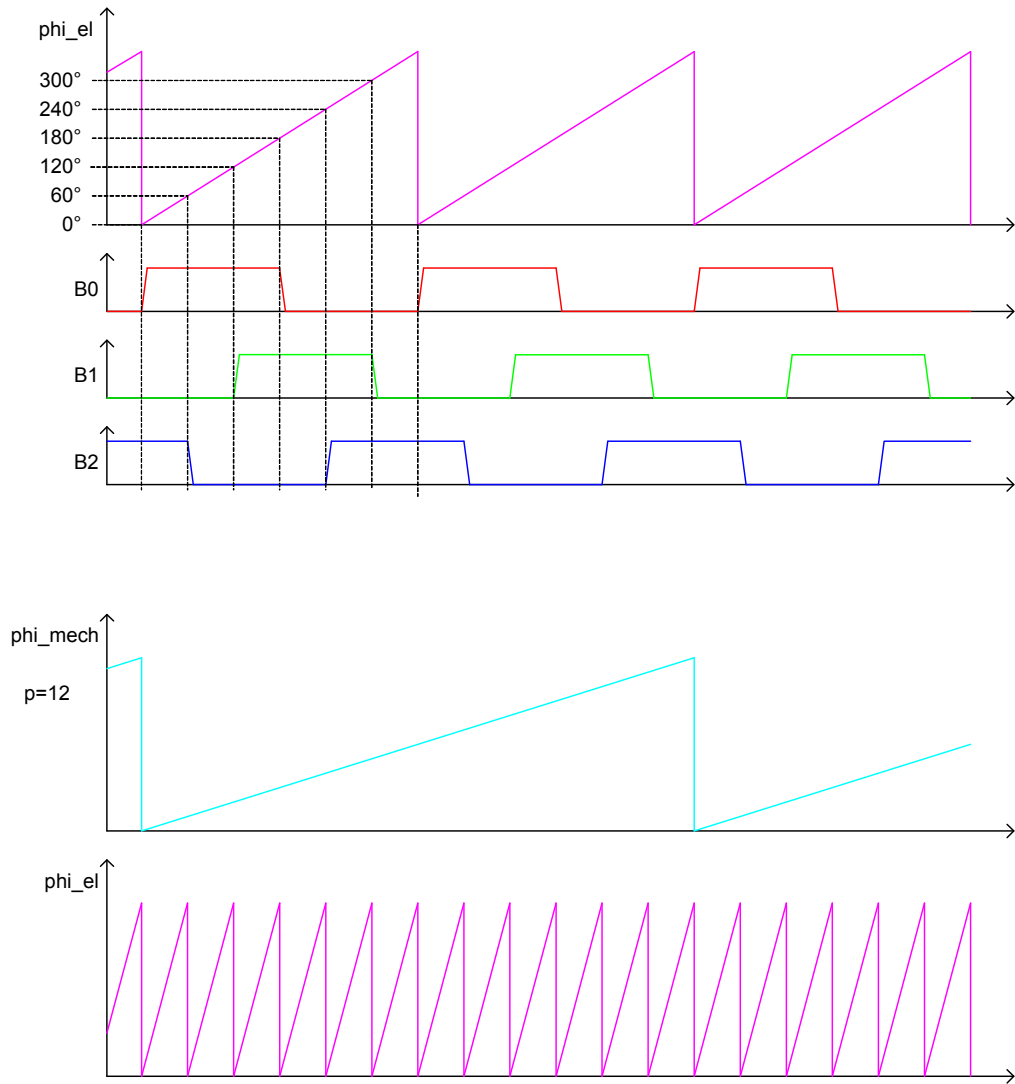
Signalname (n = 0...7)	Datentyp*	Anmerkungen
OutValue_n	uint8	Digitaler Ausgabewert 0: low; 1: high oder open collector

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

Tab. 22-13 ES5340-Digital-Direct-Out: Signale der Registerkarte „Data“

22.5.6 ES5340-Digital-Position-Sensor

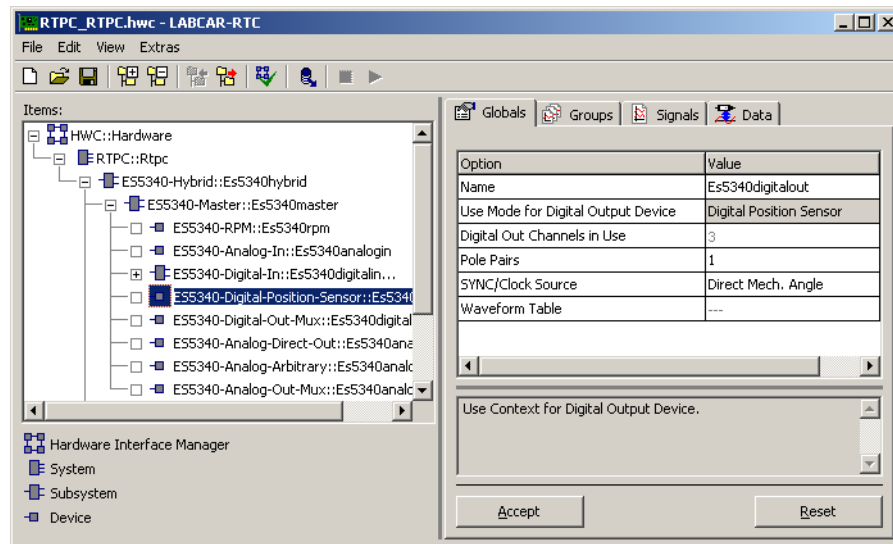
Die Betriebsart „Digital Position Sensor“ ermöglicht die Nachbildung eines digitalen Lagesensors und belegt intern drei arbiträre Signalgeneratoren.



**Abb. 22-25** Digital Position Sensor – Signalverläufe

## 22.5.7 Globals (ES5340-Digital-Position-Sensor)

Die folgende Abbildung zeigt die Konfigurationsparameter der Registerkarte „Globals“



**Abb. 22-26** Die Registerkarte „Globals“ des ES5340-Digital-Position-Sensor RTIO-Elements

#### *Use Mode for Digital Output Device*

Auswahl des Ausgabemodus.

#### *Pole Pairs*

Anzahl der Polpaare (maximal 64).

#### **Hinweis**

*Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.*

#### *SYNC/Clock Source*

Die digitalen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird. Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch die Signale „PhaseRefB0“, „PhaseRefB1“ und „PhaseRefB2“ definiert.

- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

### Hinweis

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben. Es stehen zwei Einstellungen zur Verfügung:

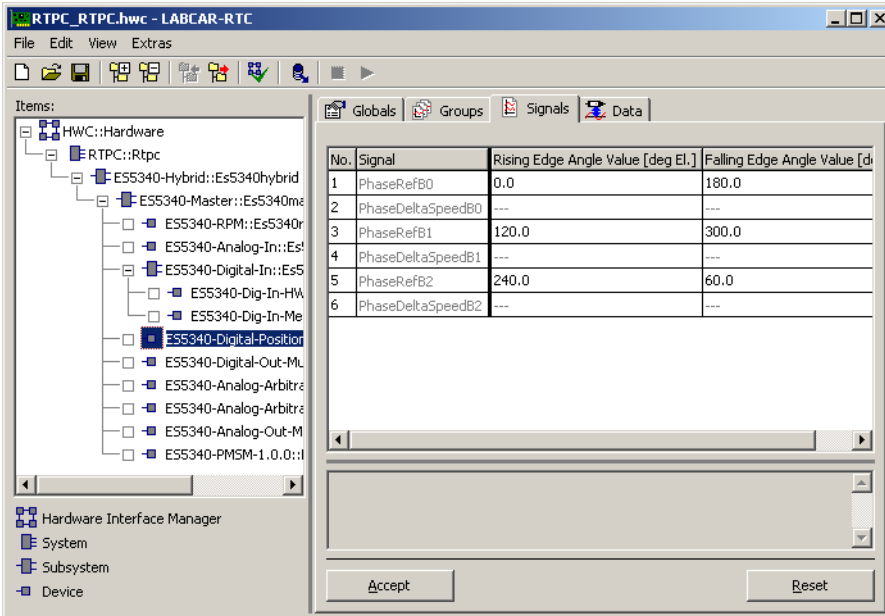
- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

#### 22.5.8 Groups (ES5340-Digital-Position-Sensor)

Das ES5340-Digital-Position-Sensor RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

#### 22.5.9 Signals (ES5340-Digital-Position-Sensor)

Die folgende Abbildung zeigt die Signale des ES5340-Digital-Position-Sensor RTIO-Elements.



No.	Signal	Rising Edge Angle Value [deg El.]	Falling Edge Angle Value [d
1	PhaseRefB0	0.0	180.0
2	PhaseDeltaSpeedB0	---	---
3	PhaseRefB1	120.0	300.0
4	PhaseDeltaSpeedB1	---	---
5	PhaseRefB2	240.0	60.0
6	PhaseDeltaSpeedB2	---	---

**Abb. 22-27** Die Registerkarte „Signals“ des ES5340-Digital-Position-Sensor RTIO-Elements

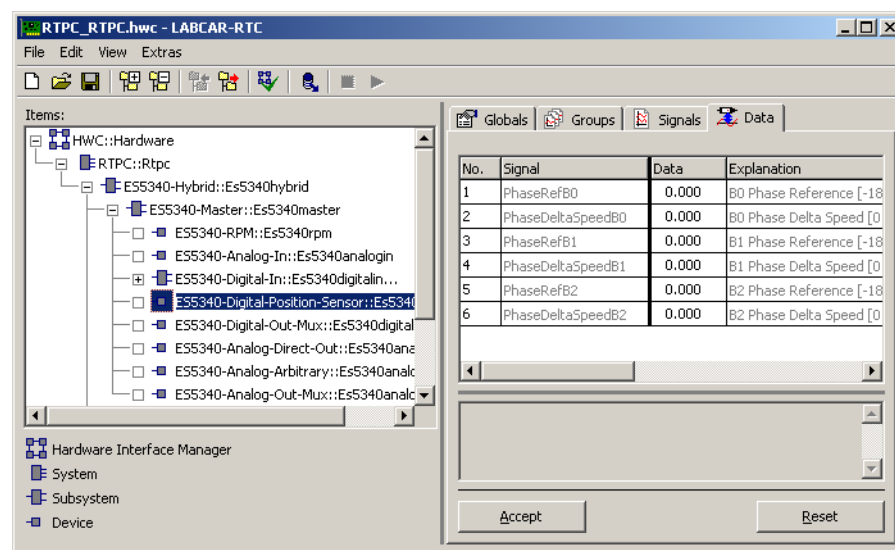
### *Rising Edge Angle Value [deg El.]*

Winkelwert für der steigende Flanke des Digitalsignals – beim Einfügen des Items werden die Defaultwerte aus Abb. 22-27 eingestellt.

### *Falling Edge Angle Value [deg El.]*

Winkelwert für der fallenden Flanke des Digitalsignals – beim Einfügen des Items werden die Defaultwerte aus Abb. 22-27 eingestellt.

## 22.5.10 Data (ES5340-Digital-Position-Sensor)



**Abb. 22-28** Die Registerkarte „Data“ des ES5340-Digital-Position-Sensor RTIO-Elements

### *PhaseRefBn (n = 0...2)*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der zur Laufzeit eine modellgesteuerte Phasenverschiebung durchgeführt werden kann.

#### **Hinweis**

Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedBn“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedBn“ für die meisten Anwendungen einen Wert > 0 besitzen.

### *PhaseDeltaSpeedBn (n = 0...2)*

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedBn“ gesteuert – eine Änderung des Signals „PhaseRefBx“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000 deg mech/s.



Signalname	Datentyp*	Anmerkungen
PhaseRefB0	sint16	B0 Phase Referenz [-180.0...180.0]
PhaseDeltaSpeedB0	uint16	B0 Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]
PhaseRefB1	sint16	B1 Phase Referenz [-180.0...180.0]
PhaseDeltaSpeedB1	uint16	B1 Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]
PhaseRefB2	sint16	B2 Phase Referenz [-180.0...180.0]
PhaseDeltaSpeedB2	uint16	B2 Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

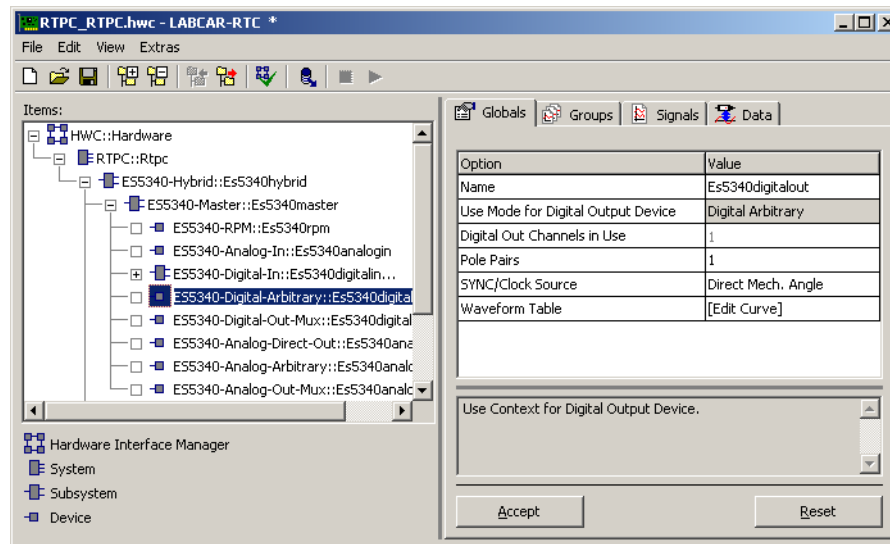
**Tab. 22-14** ES5340-Digital-Position-Sensor: Signale der Registerkarte „Data“

22.5.11 ES5340-Digital-Arbitrary

In der Betriebsart „Digital Arbitrary“ wird ein arbiträrer Signalgenerator verwendet, um (mit Hilfe des RPM-Generators) eine digitale Kurvenform zu erzeugen.

22.5.12 Globals (ES5340-Digital-Arbitrary)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-29** Die Registerkarte „Globals“ des ES5340-Digital-Arbitrary RTIO-Elements

*Use Mode for Digital Output Device*

Auswahl des Ausgabemodus.

### *Digital Out Channels in Use*

---

In der Betriebsart „Digital Arbitrary“ hat diese Einstellung keine Bedeutung, da pro RTIO-Element immer nur ein arbiträrer Signalgenerator verwendet wird.

### *Pole Pairs*

---

Anzahl der Polpaare (maximal 64).

#### **Hinweis**

*Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.*

### *SYNC/Clock Source*

---

Die digitalen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird. Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch das Signal „PhaseValue“ definiert.
- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

#### **Hinweis**

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben. Es stehen zwei Einstellungen zur Verfügung:

- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

### *Waveform Table*

---

Hier kann die Signalform des Ausgabesignals durch eine Tabelle vorgegeben werden. Hierbei werden die z-Werte der Tabelle innerhalb des Intervalls -1.0 ... 1.0, die zur Verfügung stehenden x-Werte innerhalb eines Zyklusses des RPM-Generators skaliert. Die interpolierten, analogen z-Werte werden in Binärwerte umgewandelt (Schwellwert: 0.1 (uint16-Ausgabewert: 0xCCC), d.h.  $< 0,1 \rightarrow$  low,  $\geq 0,1 \rightarrow$  high).

### 22.5.13 Groups (ES5340-Digital-Arbitrary)

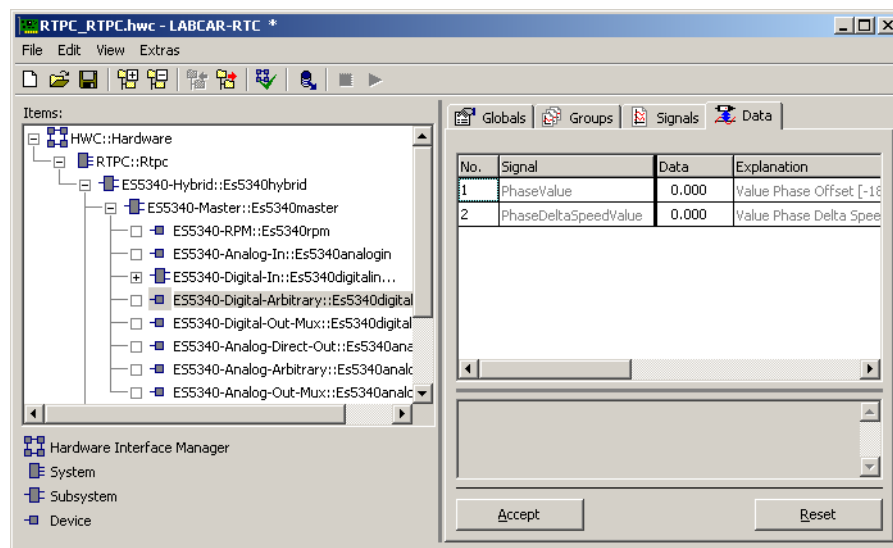
Das ES5340-Digital-Arbitrary RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

### 22.5.14 Signals (ES5340-Digital-Arbitrary)

Hier sind keine weiteren Einstellungen notwendig.

### 22.5.15 Data (ES5340-Digital-Arbitrary)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-30** Die Registerkarte „Data“ des ES5340-Digital-Arbitrary RTIO-Elements

#### *PhaseValue*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der zur Laufzeit eine modellgesteuerte Phasenverschiebung durchgeführt werden kann.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedValue“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedValue“ für die meisten Anwendungen einen Wert > 0 besitzen.*

#### *PhaseDeltaSpeedValue*

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeed“ gesteuert – eine Änderung des Signals „PhaseValue“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000 deg mech/s.

Signalname	Datentyp*	Anmerkungen
PhaseValue	sint16	Lokale Phasenverschiebung [-180.0...180.0 deg El. Angle]
PhaseDeltaSpeedValue	uint16	Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

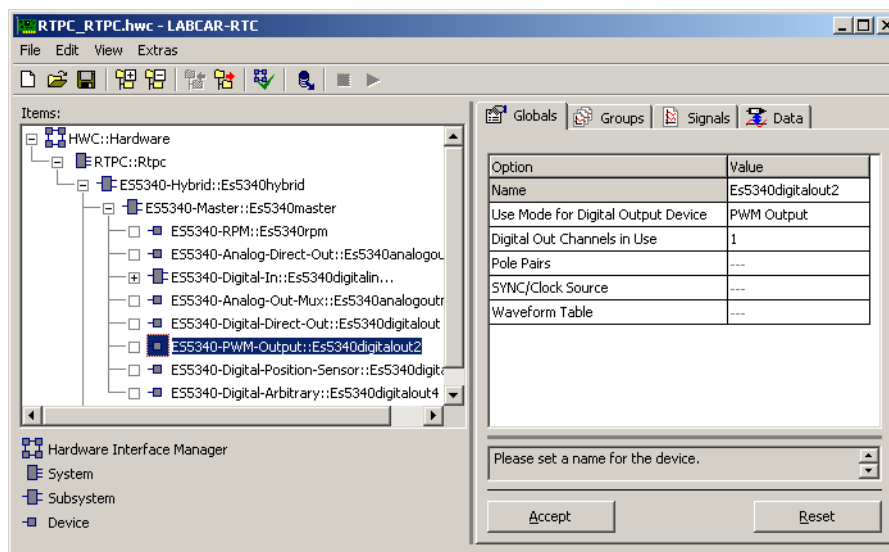
**Tab. 22-15** ES5340-Digital-Arbitrary: Signale der Registerkarte „Data“

22.5.16 ES5340-PWM-Output

In der Betriebsart „PWM Output“ können bis zu acht digitale Ausgänge für die PWM-Ausgabe konfiguriert werden (nur positive Logik!). Dabei wird für jeden PWM-Kanal Frequenz und Tastverhältnis separat vorgegeben. Die Übernahme der neuen Daten erfolgt jeweils am Ende eines PWM-Zyklus.

22.5.17 Globals (ES5340-PWM-Output)

Die folgende Abbildung zeigt die Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-31** Die Registerkarte „Globals“ des ES5340-PWM-Output-Elements  
*Use Mode for Digital Output Device*

Auswahl des Ausgabemodus.

*Digital Out Channels in Use*

Anzahl der verwendeten digitalen Ausgabekanäle (max. 8).

22.5.18 Groups (ES5340-PWM-Output)

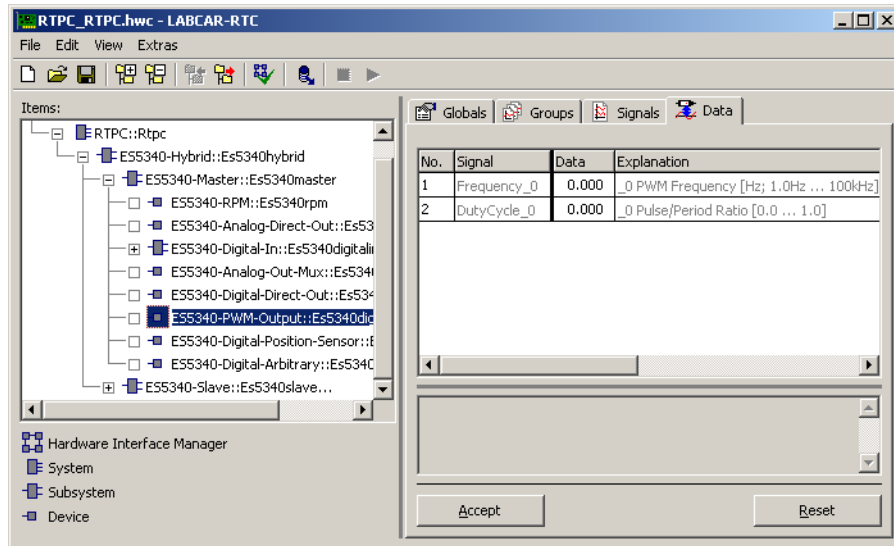
Das ES5340-PWM-Output besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators bzw der jeweiligen Signalgeneratoren.

22.5.19 Signals (ES5340-PWM-Output)

Hier sind keine weiteren Einstellungen notwendig.

22.5.20 Data (ES5340-PWM-Output)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-32** Die Registerkarte „Data“ des ES5340-PWM-Output-RTIO-Elements *Frequency<sub>n</sub>*

Die Frequenz des PWM-Signals am Ausgang n (n = 0...7).

*DutyCycle<sub>n</sub>*

Das Tastverhältnis des PWM-Signals am Ausgang n (n = 0...7).

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Frequency <sub>n</sub>	uint32	Frequenz des PWM Signals (1.0Hz...100kHz)
DutyCycle <sub>n</sub>	uint32	Tastverhältnis (0.0...1.0)

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-16** ES5340-PWM-Output: Signale der Registerkarte „Data“

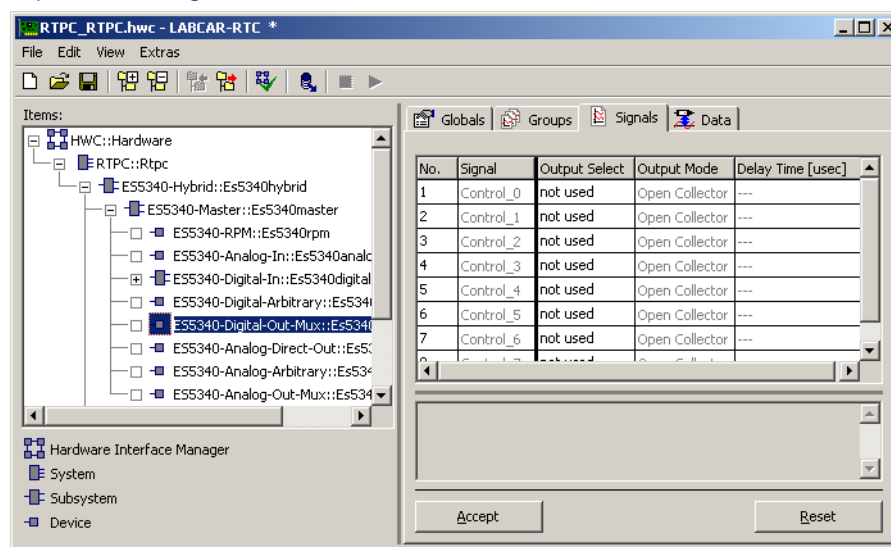
## 22.6 ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer

In dem RTIO-Element „ES5340-Digital-Out-Mux“ werden (zuvor im RTIO-Element „ES5340-Digital-Out“ konfigurierte) Signale den tatsächlich vorhandenen digitalen Signalausgängen des ES5340.1/2 Electric Drive Simulation Board zugewiesen.

### 22.6.1 Signals (ES5340-Digital-Out-Mux)

In der Registerkarte „Signals“ werden die acht (ES5340.1/2-Master) bzw. sechs (ES5340.1/2-Slave) digitalen Ausgänge konfiguriert.

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 22-33** Die Registerkarte „Signals“ des ES5340-Digital-Out-Mux RTIO-Elements

#### Output Select

Das Optionsfeld „Output Select“ dient zur Auswahl des internen Signals, das auf den jeweiligen Ausgang gelegt werden soll.

Es wird dabei die Liste aller internen Signalquellen zur Auswahl angeboten:

- Ausgabegrößen aller Digital-Out RTIO-Elemente
- digitale Messwerte der Inverter-Gate-Steuerung
- digitale Ausgangsgrößen eines FPGA-Modells (sofern vorhanden)

Sofern ein FPGA-Modell digitale Ausgangsgrößen bereitstellt, können diese auch über eine angeschlossene ES5340.1/2-Slave Karte ausgegeben werden. Dies setzt die Verbindung der beiden Karten über ein Gigabit-Link voraus.

#### Hinweis

*Beim ES5340.1/2 Electric Drive Simulation Board können digitale Messwerte des Inverter-Gates direkt oder zeitlich verzögert wieder ausgegeben werden (dies ist z.B. für die Diagnose einiger Steuergeräte notwendig). Hierzu muss das ES5340-Dig-In-HW RTIO-Element im RTIO-Baum vorhanden sein.*

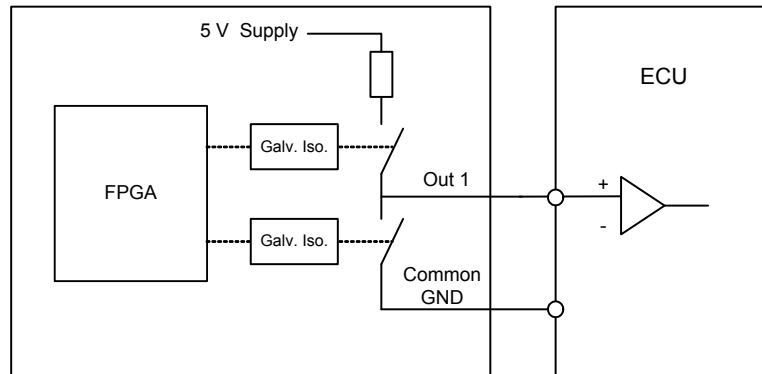
### Output Mode

---

Es kann zwischen zwei verschiedenen Ausgabearten unterschieden werden:

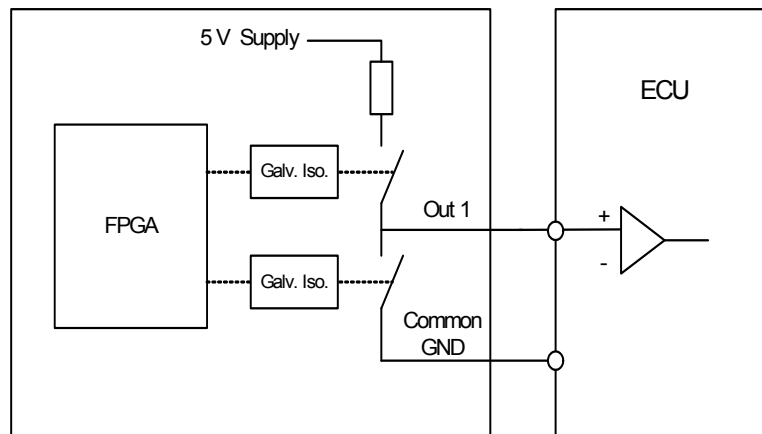
- Pull-Up to +5V

Hier wird ein interner Pull-Up-Widerstand an 5 V geschaltet



- Open Collector

Hier wird ein externer Pull-Up-Widerstand erwartet.



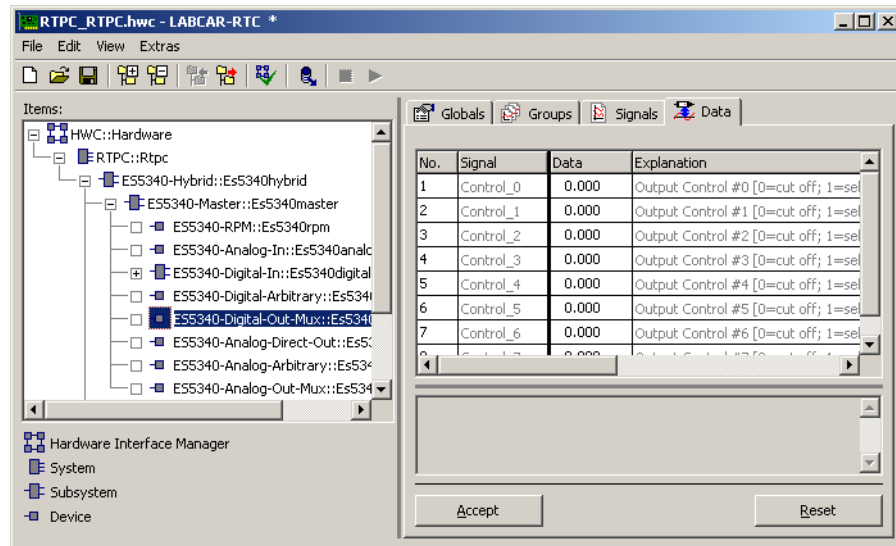
### Delay Time

---

Die Zeitverzögerung, mit der die Inverter-Gate-Messsignale wieder ausgegeben werden. Es sind Werte von 0.008  $\mu$ s bis 5  $\mu$ s möglich.

22.6.2 Data (ES5340-Digital-Out-Mux)

In der vorhandenen Signalgruppe werden acht RTIO-Signale verarbeitet, die das Schließen des Ausgangsrelais eines jeden Kanals steuern. Weiterhin ist es möglich, die Ausgänge für eine Fehlersimulation dauerhaft auf GND oder +5 V zu legen.



**Abb. 22-34** Die Registerkarte „Data“ des ES5340-Digital-Out-Mux RTIO-Elements

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Control_n	uint8	0: Ausgang getrennt 1: Ausgang verbunden 2: Kurzschluss nach GND 3: Kurzschluss nach +5 V

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-17** ES5340-Digital-Arbitrary: Signale der Registerkarte „Data“



## 22.7 ES5340-Analog-Out – Analoge Ausgänge

Das ES5340-Analog-Out RTIO-Element kann in fünf verschiedenen Betriebsarten verwendet werden.

Mode	Beschreibung
Analog Direct Out	Ausgabe eines vorgegebenen Spannungswertes
Sine Extrapolated	Ausgabe eines sinusförmigen Spannungsverlaufs
Sine Encoder	Simulation eines Lagesensors des Typs „Sine Encoder“
Resolver	Simulation eines Lagesensors des Typs „Resolver“
Analog Arbitrary	Ausgabe eine frei konfigurierbaren Wellenform

**Tab. 22-18** Betriebsarten des ES5340-Analog-Out RTIO-Elements

### Hinweis

*Nähere Informationen zu den Betriebsarten finden Sie im Benutzerhandbuch des ES5340.1/2 Electric Drive Simulation Board.*

Die Konfiguration des RTIO-Elements ist für jeden Ausgabemodus unterschiedlich – abhängig von diesem ändert sich auch das Aussehen des ES5340-Analog-Out RTIO-Elements.

Für die Ausgabe von Analogwerten verwendet die ES5340.1/2 ein zweistufiges Verfahren:

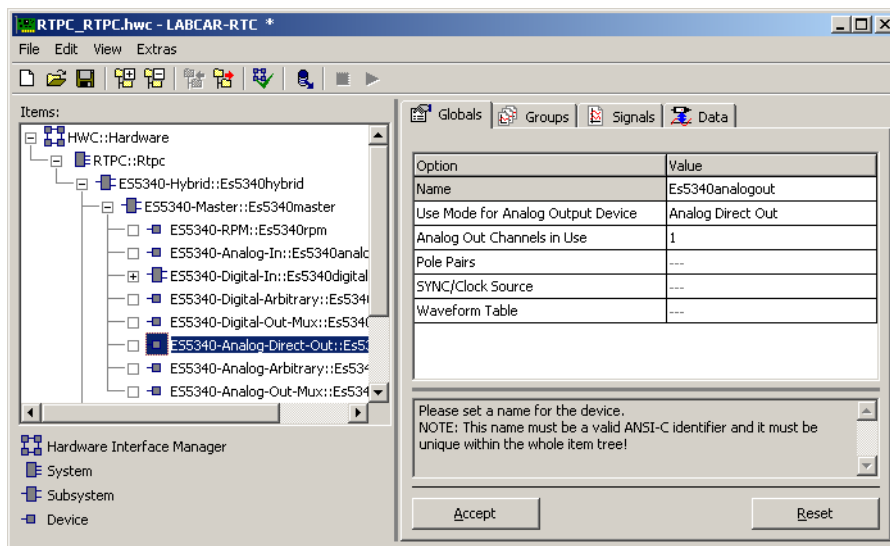
1. Die Ausgabewerte werden über das RTIO-Element ES5340-Analog-Out konfiguriert. Hier können mehr Signale konfiguriert werden als Ausgangskanäle auf der ES5340.1/2 zur Verfügung stehen: Jeweils 8 Kanäle vom Typ „Analog Direct Out“ und 8 Kanäle des Typs „Sine Extrapolated“, „Sine Encoder“, „Resolver“ oder „Analog Arbitrary“.
2. Die Auswahl der Werte, die tatsächlich ausgegeben werden, erfolgt über das ES5340-Analog-Out-Mux RTIO-Element (siehe „ES5340-Analog-Out-Mux – Analoger Ausgangsmultiplexer“ auf Seite 761).

### 22.7.1 ES5340-Analog-Direct-Out

In der Betriebsart „Analog Direct Out“ können Sie 1 bis 8 Analogausgänge zur unmittelbaren Ausgabe eines analogen Wertes verwenden.

### 22.7.2 Globals (ES5340-Analog-Direct-Out)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-35** Die Registerkarte „Globals“ des ES5340-Analog-Direct-Out RTIO-Elements

#### Use Mode for Analog Output Device

Auswahl des Ausgabemodus.

#### Analog Out Channels in Use

Anzahl der verwendeten analogen Ausgabekanäle (max. 8).

### 22.7.3 Groups (ES5340-Analog-Direct-Out)

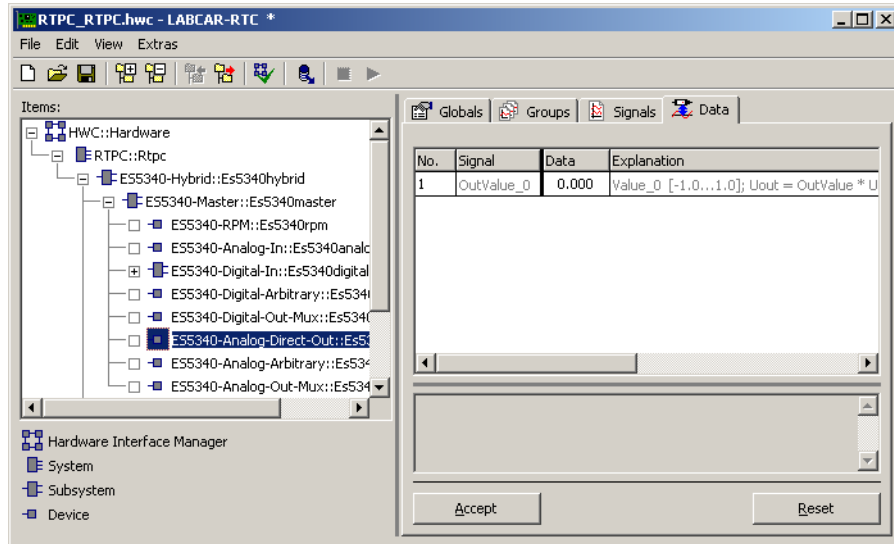
Das ES5340-Analog-Direct-Out RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

### 22.7.4 Signals (ES5340-Analog-Direct-Out)

Hier sind keine weiteren Einstellungen notwendig.

22.7.5 Data (ES5340-Analog-Direct-Out)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-36** Die Registerkarte „Data“ des ES5340-Analog-Direct-Out RTIO-Elements

*OutValue\_x*

Relative Amplitude der Ausgabe – wird mit der im Analogmultiplexer festgelegten Referenzspannung multipliziert.

Signalname (n = 0...5)	Datentyp	Anmerkungen
OutValue_n	sint16	Analoger Ausgabewert im Bereich -1.0 ... 1.0; Uout = OutValue * Uref

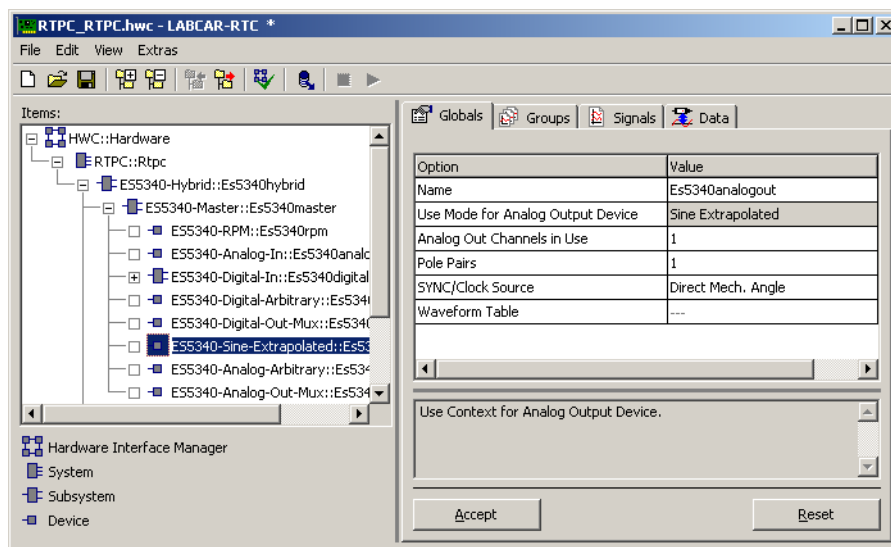
**Tab. 22-19** ES5340-Analog-Direct-Out: Signale der Registerkarte „Data“

### 22.7.6 ES5340-Sine-Extrapolated

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Analog-Out RTIO-Elements in der Betriebsart „Sine Extrapolated“. Diese dient zur Ausgabe eines oder mehrerer Sinussignale, welche (bezüglich Phase und Amplitude) getrennt voneinander einstellbar sind.

### 22.7.7 Globals (ES5340-Sine-Extrapolated)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-37** Die Registerkarte „Globals“ des ES5340-Sine-Extrapolated RTIO-Elements

#### *Use Mode for Analog Output Device*

Auswahl des Ausgabemodus.

#### *Analog Out Channels in Use*

Anzahl der verwendeten arbiträren Signalgeneratoren (max. 8). Die Steuerungssignale werden entsprechend der Kanalanzahl erzeugt.

#### *Pole Pairs*

Anzahl der Polpaare (max. 64).

#### **Hinweis**

Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.

### SYNC/Clock Source

Die analogen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird. Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch das Signal „PhaseValue\_n“ definiert.
- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

#### **Hinweis**

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben.

Es stehen zwei Einstellungen zur Verfügung:

- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

#### 22.7.8 Groups (ES5340-Sine-Extrapolated)

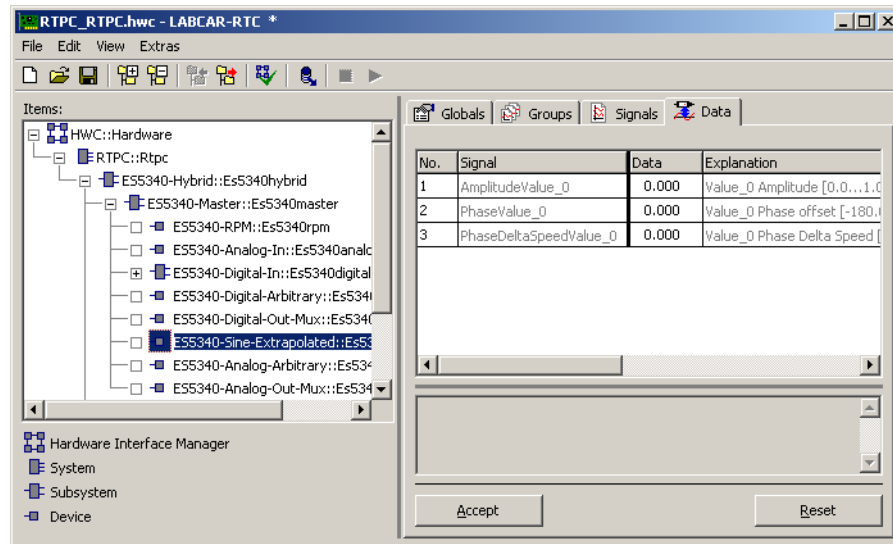
Das ES5340-Sine-Extrapolated RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung jeweiligen Signalgenerators.

#### 22.7.9 Signals (ES5340-Sine-Extrapolated)

Hier sind keine weiteren Einstellungen notwendig.

## 22.7.10 Data (ES5340-Sine-Extrapolated)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-38** Die Registerkarte „Data“ des ES5340-Sine-Extrapolated RTIO-Element

#### *AmplitudeValue<sub>n</sub> (n = 0...2)*

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeValue<sub>n</sub>“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

#### *PhaseValue<sub>n</sub> (n = 0...2)*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedValue<sub>n</sub>“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedValue<sub>n</sub>“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.*

#### *PhaseDeltaSpeedValue<sub>n</sub> (n = 0...2)*

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedValue<sub>n</sub>“ gesteuert – eine Änderung des Signals „PhaseValue<sub>n</sub>“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

Signalname	Datentyp*	Anmerkungen
AmplitudeValue_n	uint16	Signalamplitude des Ausgangssignals [0.0...1.0]
PhaseValue_n	sint16	Phasenverschiebung [-180.0...180.0 deg mech]
PhaseDeltaSpeedValue_n	uint16	Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-20** ES5340-Sine-Extrapolated: Signale der Registerkarte „Data“

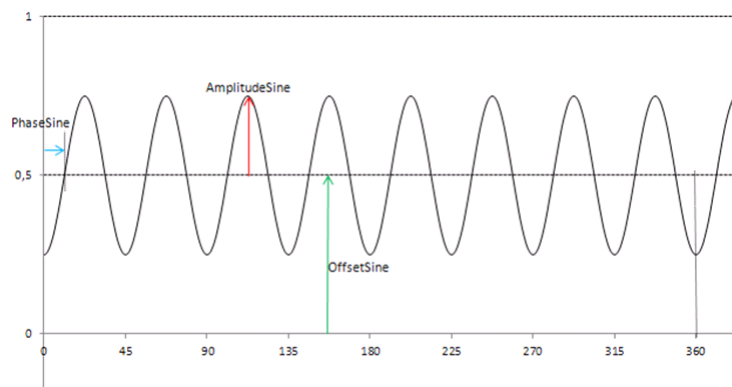
### 22.7.11 ES5340-Sine-Encoder

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Analog-Out RTIO-Elements in der Betriebsart „Sine Encoder“.

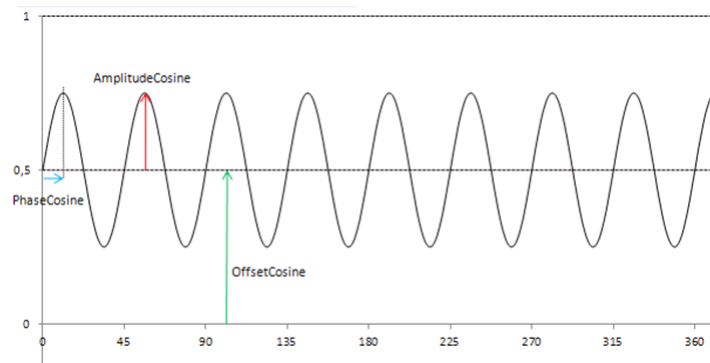
In dieser Betriebsart werden zwei arbiträre Signalgeneratoren zur Generierung von Sinussignalen verwendet, die zueinander um 90° phasenverschoben sind (Sinus-, Kosinussignal). Diese Signale werden unmittelbar ausgegeben und dienen zur Nachbildung des entsprechenden Sensors.

In den folgenden Abbildungen sind Signalverläufe gezeigt mit den Parametern:

- Pole Pairs: 8
- AmplitudeSine: 0,25
- OffsetSine: 0,5
- PhaseSine: -90,0°
- AmplitudeCosine: 0,25
- OffsetCosine: 0,5
- PhaseCosine: -90,0°



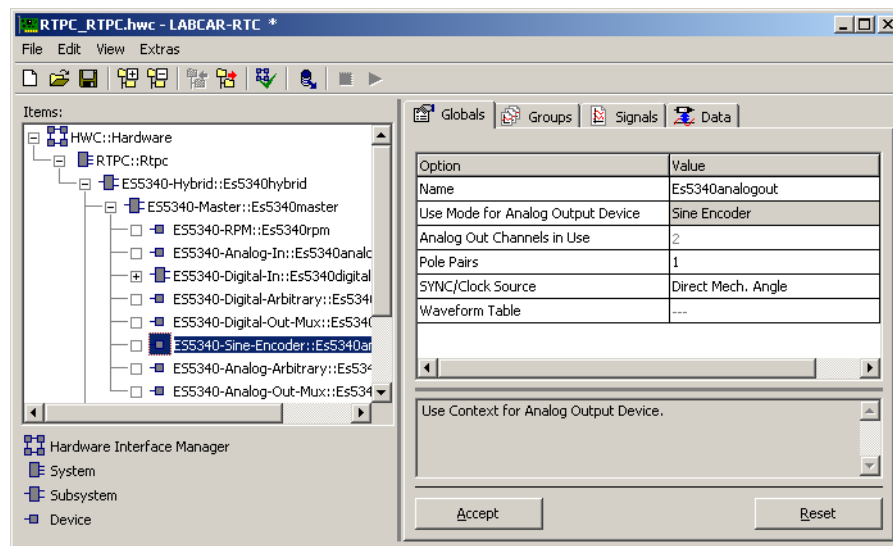
**Abb. 22-39** Sine-Encoder: Sinuskanal



**Abb. 22-40** Sine-Encoder: Kosinuskanal

### 22.7.12 Globals (ES5340-Sine-Encoder)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-41** Die Registerkarte „Globals“ des ES5340-Sine-Encoder RTIO-Elements

#### *Use Mode for Analog Output Device*

Auswahl des Ausgabemodus.

#### *Pole Pairs*

Anzahl der Polpaare (max. 64).

#### **Hinweis**

Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.



### SYNC/Clock Source

Die analogen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird.

Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch die Signale „PhaseSine“ und „PhaseCosine“ definiert.
- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

### Hinweis

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben. Es stehen zwei Einstellungen zur Verfügung:

- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

#### 22.7.13 Groups (ES5340-Sine-Encoder)

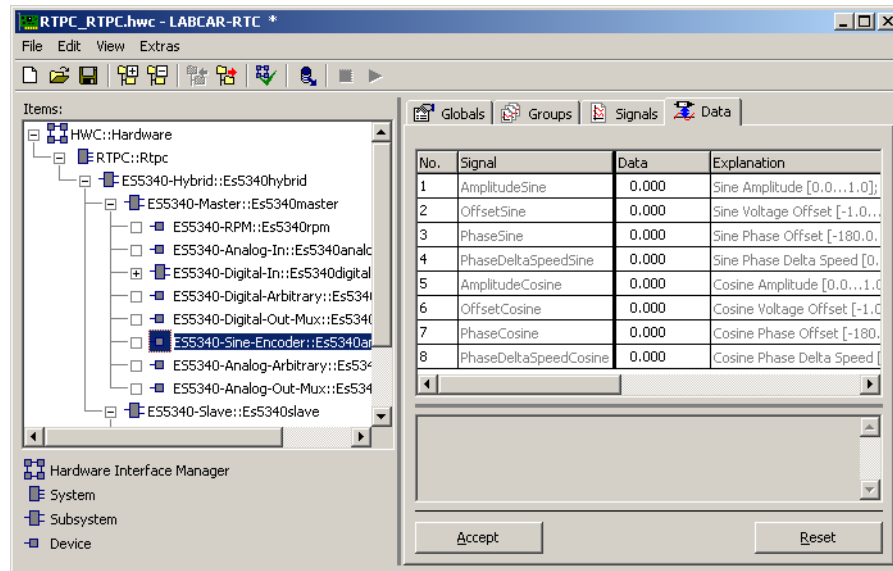
Das ES5340-Sine-Encoder RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

#### 22.7.14 Signals (ES5340-Sine-Encoder)

Hier sind keine weiteren Einstellungen notwendig.

## 22.7.15 Data (ES5340-Sine-Encoder)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-42** Die Registerkarte „Data“ des ES5340-Sine-Encoder RTIO-Elements *AmplitudeSine*

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeSine“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

#### *OffsetSine*

Der Offset des Signalgenerators wird mit dem Signal „OffsetSine“ gesteuert (Wertebereich: -1.0...1.0). Der Ausgabewert wird um den Wert von „OffsetSine“ verschoben.

Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer gebildet. Überschreitet die Summe aus „OffsetSine“ und „AmplitudeSine“ den Wertebereich -1.0 bis +1.0, wird der Ausgabewert auf diesen Wertebereich begrenzt.

#### *PhaseSine*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der zur Laufzeit eine modellgesteuerte Phasenverschiebung durchgeführt werden kann.

#### **Hinweis**

Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedSine“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedSine“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.

#### *PhaseDeltaSpeedSine*

---

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedSine“ gesteuert – eine Änderung des Signals „PhaseSine“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

#### *AmplitudeCosine*

---

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeCosine“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

#### *OffsetCosine*

---

Der Offsetwert des Signalgenerators wird mit dem Signal „OffsetSine“ gesteuert – der Wertebereich geht von -1.0 ... 1.0. Der Ausgabewert wird um den Wert von OffsetSine verschoben.

Zu beachten ist, dass erst im Ausgangsmultiplexer aus dem internen Signal mit dem Wertebereich -1.0 bis +1.0 ein Spannungssignal gebildet wird. Überschreitet die Summe aus „OffsetCosine“ und „AmplitudeCosine“ den Wertebereich -1.0 bis +1.0, wird der Ausgabewert auf diesen Wertebereich begrenzt.

#### *PhaseCosine*

---

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedCosine“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedCosine“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.*

#### *PhaseDeltaSpeedCosine*

---

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedCosine“ gesteuert – eine Änderung des Signals „PhaseCosine“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

Signalname	Datentyp*	Anmerkungen
AmplitudeSine	uint16	Signalamplitude des Ausgangssignals [0.0 ... 1.0]
OffsetSine	sint16	Offset Wert für das Ausgangssignal [-1.0 ... 1.0]
PhaseSine	sint16	Phasenverschiebung [-180.0 ... 180.0 deg mech]
PhaseDelta-SpeedSine	uint16	Phasenänderungsgeschwindigkeit [0.0 ... 1000.0 deg mech/s]
AmplitudeCosine	uint16	Signalamplitude des Ausgangssignals [0.0 ... 1.0]
OffsetCosine	sint16	Offset Wert für das Ausgangssignal [-1.0 ... 1.0]
PhaseCosine	sint16	Phasenverschiebung [-180.0 ... 180.0 deg mech]
PhaseDelta-SpeedCosine	uint16	Phasenänderungsgeschwindigkeit [0.0 ... 1000.0 deg mech/s]

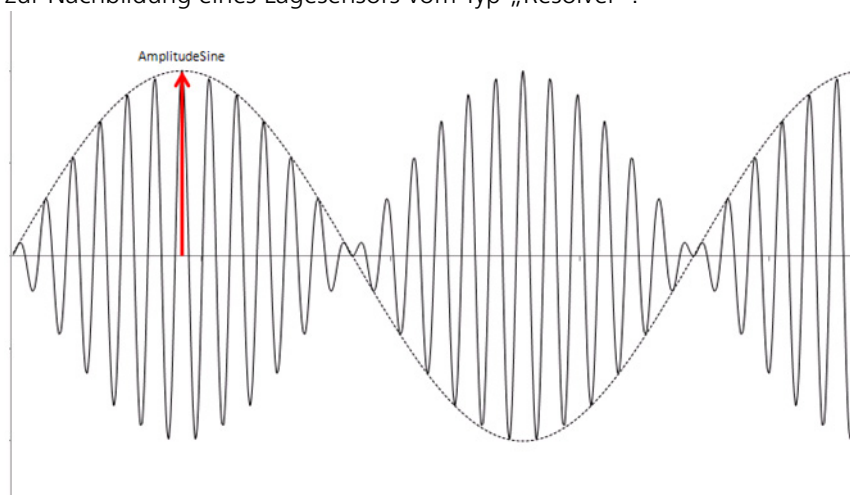
\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-21** ES5340-Sine-Encoder: Signale der Registerkarte „Data“

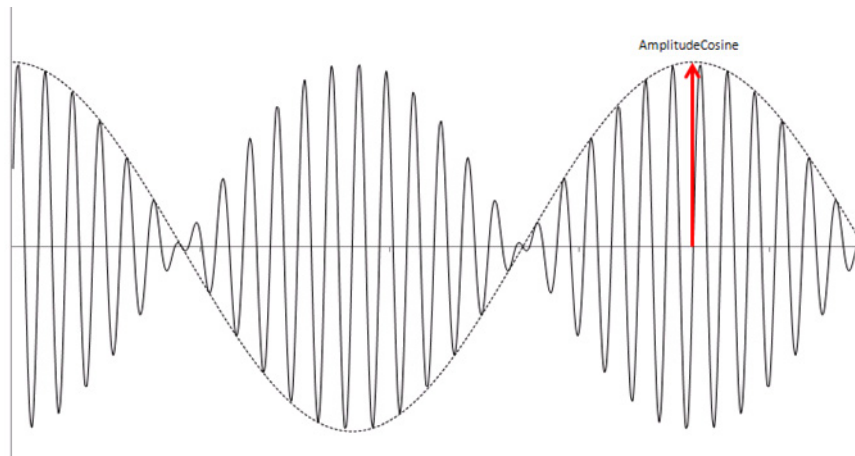
#### 22.7.16 ES5340-Resolver

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Resolver RTIO-Elements in der Betriebsart „Resolver“.

Diese Betriebsart verwendet zwei arbiträre Signalgeneratoren zur Generierung von Sinussignalen, die zueinander um 90° phasenverschoben sind (Sinus- und Kosinussignal). Diese Signale bilden die Hüllkurve eines höherfrequenten Signals, welches mittels Hardwaremultiplizierer erzeugt wird. Diese Kombination dient zur Nachbildung eines Lagesensors vom Typ „Resolver“.



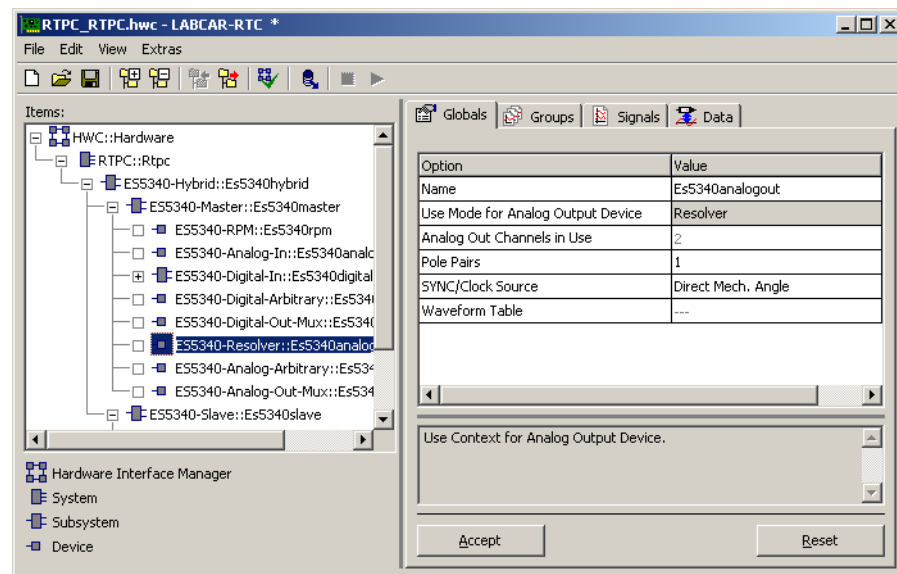
**Abb. 22-43** Resolver: Sinuskanal



**Abb. 22-44** Resolver: Kosinuskanal

### 22.7.17 Globals (ES5340-Resolver)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-45** Die Registerkarte „Globals“ des ES5340-Resolver RTIO-Elements  
*Use Mode for Analog Output Device*

Auswahl des Ausgabemodus.

*Analog Out Channels in Use*

Diese Einstellung hat beim Ausgabemodus „Resolver“ keine Bedeutung (intern werden zwei arbiträre Signalgeneratoren mit aufeinanderfolgender ID verwendet).

### *Pole Pairs*

---

Anzahl der Polpaare (max. 64).

#### **Hinweis**

*Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.*

### *SYNC/Clock Source*

---

Die analogen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird. Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch die Signale „PhaseSine“ und „PhaseCosine“ definiert.
- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

#### **Hinweis**

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben. Es stehen zwei Einstellungen zur Verfügung:

- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

#### 22.7.18 Groups (ES5340-Resolver)

---

Das ES5340-Resolver RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

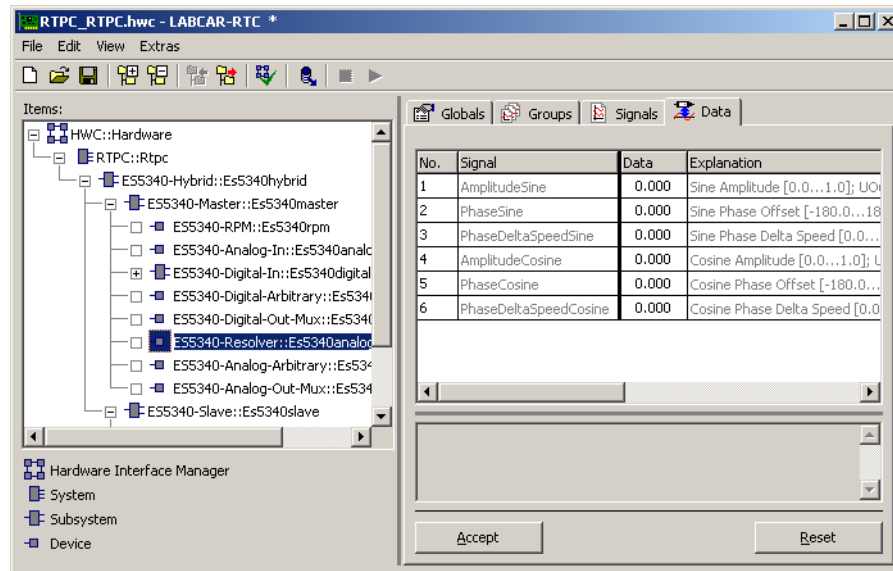
#### 22.7.19 Signals (ES5340-Resolver)

---

Hier sind keine weiteren Einstellungen notwendig.

## 22.7.20 Data (ES5340-Resolver)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-46** Die Registerkarte „Data“ des ES5340-Resolver RTIO-Elements

#### *AmplitudeSine*

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeSine“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

#### *PhaseSine*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedSine“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedSine“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.*

#### *PhaseDeltaSpeedSine*

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedSine“ gesteuert – eine Änderung des Signals „PhaseSine“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

### AmplitudeCosine

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeCosine“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

### PhaseCosine

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedCosine“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedCosine“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.*

### PhaseDeltaSpeedCosine

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedCosine“ gesteuert – eine Änderung des Signals „PhaseCosine“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

Signalname	Datentyp*	Anmerkungen
AmplitudeSine	uint16	Signalamplitude des Ausgangssignals [0.0 ... 1.0]
PhaseSine	sint16	Phasenverschiebung [-180.0 ... 180.0 deg mech]
PhaseDeltaSpeedSine	uint16	Phasenänderungsgeschwindigkeit [0.0 ... 1000.0 deg mech/s]
AmplitudeCosine	uint16	Signalamplitude des Ausgangssignals [0.0 ... 1.0]
PhaseCosine	sint16	Phasenverschiebung [-180.0 ... 180.0 deg mech]
PhaseDeltaSpeedCosine	uint16	Phasenänderungsgeschwindigkeit [0.0 ... 1000.0 deg mech/s]

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-22** ES5340-Resolver: Signale der Registerkarte „Data“

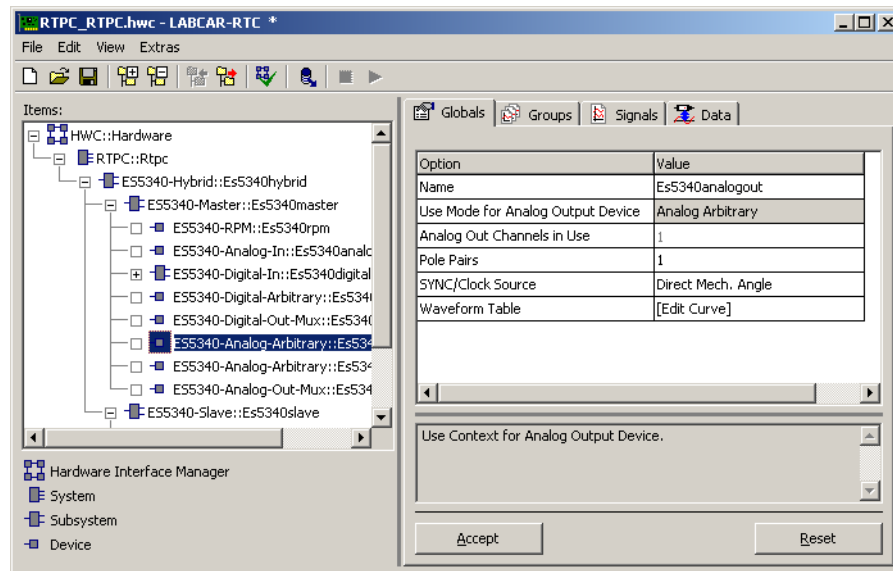
#### 22.7.21 ES5340-Analog-Arbitrary

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Analog-Arbitrary RTIO-Elements in der Betriebsart „Analog Arbitrary“. In dieser Betriebsart wird ein arbiträrer Signalgenerator verwendet, um mithilfe des RPM-Generators eine beliebige Kurvenform zu erzeugen.



## 22.7.22 Globals (ES5340-Analog-Arbitrary)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-47** Die Registerkarte „Globals“ des ES5340-Analog-Arbitrary RTIO-Elements

#### *Use Mode for Analog Output Device*

Auswahl des Ausgabemodus.

#### *Analog Out Channels in Use*

Diese Einstellung hat für den Ausgabemodus „Analog Arbitrary“ keine Bedeutung.

#### *Pole Pairs*

Anzahl der Polpaare (max. 64).

#### **Hinweis**

Eine Polpaarzahl von  $n$  bewirkt eine Erhöhung der Ausgabefrequenz um den Faktor  $n$  gegenüber der vom RPM-Generator vorgegebenen Grundfrequenz.

### SYNC/Clock Source

Die analogen Signale werden abhängig vom mechanischen Winkel ausgegeben, der von der ES5340.1/2 generiert und dem LABCAR-Modell über das RTIO-Element „ES5340-RPM“ (Signal „ActMechAngle“) zur Verfügung gestellt wird. Mit dem ES5340.1/2 Electric Drive Simulation Board ist es möglich, die Phase des mechanischen Winkels zu verschieben.

Es gibt dabei jeweils eine lokale Phasenverschiebung und eine globale Phasenverschiebung:

- Die **lokale Phasenverschiebung** wird im Register „Data“ durch das Signal „PhaseValue“ definiert.
- Die **globale Phasenverschiebung** wird in dem RTIO-Element „ES5340-RPM“ (im Register „Data“) durch das Signal „PSMechAnglePhase“ definiert.

#### **Hinweis**

*Die lokale Phasenverschiebung ist bezogen auf die Phase des erzeugten Signals, dessen Frequenz sich aus der Motordrehzahl multipliziert mit der Anzahl der Polpaare ergibt.*

Mit dem Konzept der globalen Phasenverschiebung ist es möglich, die Phasenlage einer Gruppe von analogen oder digitalen Ausgaben synchron zu verschieben. Es stehen zwei Einstellungen zur Verfügung:

- **Direct Mech. Angle**  
Zur Berechnung der Phasenlage wird nur der Wert der lokalen Phasenverschiebung berücksichtigt.
- **Phase Shifted Mech. Angle**  
Zur Berechnung der Phasenlage werden die Werte der lokalen und der globalen Phasenverschiebung berücksichtigt.

### Waveform Table

Hier kann die Signalform des Ausgabesignals in Form einer Tabelle vorgegeben werden. Hierbei werden die z-Werte der Tabelle innerhalb des Intervalls -1.0 ... 1.0, die zur Verfügung stehenden x-Werte innerhalb eines Zyklus des RPM-Generators skaliert.

#### 22.7.23 Groups (ES5340-Analog-Arbitrary)

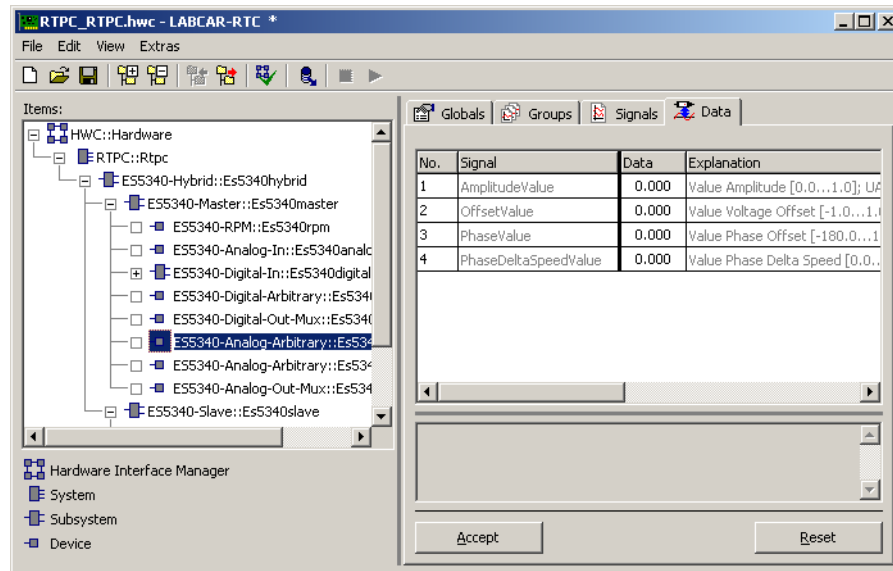
Das ES5340-Analog-Arbitrary RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

#### 22.7.24 Signals (ES5340-Analog-Arbitrary)

Hier sind keine weiteren Einstellungen notwendig.

## 22.7.25 Data (ES5340-Analog-Arbitrary)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 22-48** Die Registerkarte „Data“ des ES5340-Analog-Arbitrary RTIO-Elements

#### *AmplitudeValue*

Die Ausgangsamplitude des Signalgenerators wird über das Signal „AmplitudeValue“ gesteuert (Wertebereich: 0.0...1.0). Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer aus diesem Signal und der jeweiligen Referenzspannung gebildet.

#### *OffsetValue*

Der Offset des Signalgenerators wird mit dem Signal „OffsetValue“ gesteuert (Wertebereich: -1.0...1.0). Der Ausgabewert wird um den Wert von „OffsetValue“ verschoben.

Das tatsächliche Spannungssignal wird erst im Ausgangsmultiplexer gebildet. Überschreitet die Summe aus „OffsetValue“ und „AmplitudeValue“ den Wertebereich -1.0 bis +1.0, wird der Ausgabewert auf diesen Wertebereich begrenzt.

#### *PhaseValue*

Dieses Signal beschreibt eine lokale Phasenverschiebung, mit der eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden.

#### **Hinweis**

*Die Phasenverschiebung wird mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeedValue“) durchgeführt. Daher muss das Signal „PhaseDeltaSpeedValue“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.*

PhaseDeltaSpeedValue

Die Phasenänderungsgeschwindigkeit wird über das Signal „PhaseDeltaSpeedValue“ gesteuert – eine Änderung des Signals „PhaseValue“ wirkt nicht unendlich schnell, sondern erfolgt mit dieser Geschwindigkeit.

Der Wertebereich reicht von 0 (die Verstellung der Phase findet sprunghaft statt) bis 1000° mech/s.

Signalname	Datentyp*	Anmerkungen
AmplitudeValue	uint16	Signalamplitude des Ausgangssignals [0.0...1.0]
OffsetValue	sint16	Offset für das Ausgangssignal [-1.0...1.0]
PhaseValue	sint16	Phasenverschiebung [-180.0...180.0 deg mech]
PhaseDeltaSpeedValue	uint16	Phasenänderungsgeschwindigkeit [0.0...1000.0 deg mech/s]

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 22-23** ES5340-Analog-Arbitrary: Signale der Registerkarte „Data“

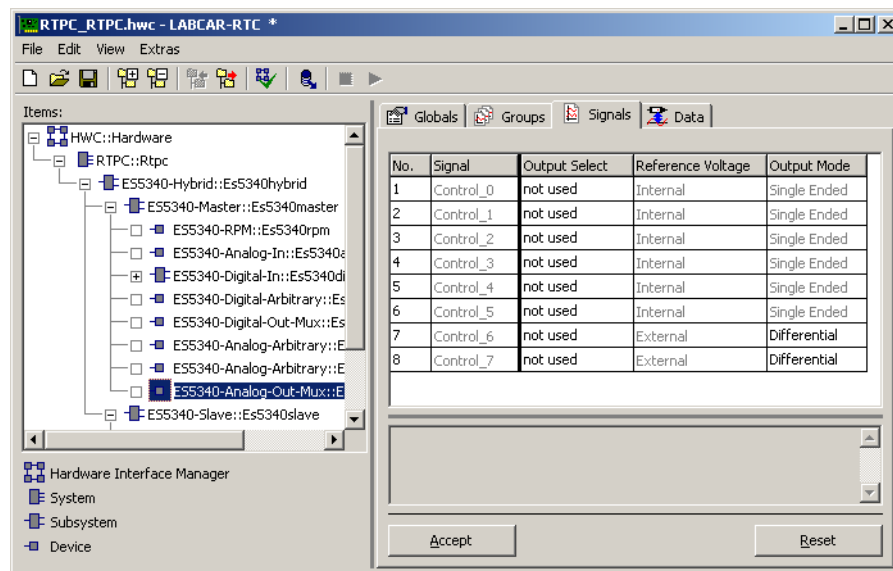
## 22.8 ES5340-Analog-Out-Mux – Analoger Ausgangsmultiplexer

In dem RTIO-Element „ES5340-Analog-Out-Mux“ werden (zuvor im RTIO-Element „ES5340-Analog-Out“ konfigurierte) Signale den tatsächlich vorhandenen analogen Signalausgängen des ES5340.1/2 Electric Drive Simulation Board zugewiesen.

### 22.8.1 Signals (ES5340-Analog-Out-Mux)

In der Registerkarte „Signals“ werden die acht (ES5340-Master) bzw. sechs (ES5340-Slave) analogen Ausgänge konfiguriert. Dabei können die Eigenschaften des Analogausgangs sowie die Quelle des Ausgabewertes festgelegt werden.

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“ – die Eigenschaften der einzelnen Parameter sind in Tab. 22-24 auf Seite 763 zusammengefasst. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 22-49** Die Registerkarte „Signals“ des ES5340-Analog-Out-Mux RTIO-Elements

#### *Output Select*

Das Optionsfeld „Output Select“ dient zur Auswahl des internen Signals, das auf den jeweiligen Ausgang gelegt werden soll. Es wird dabei die Liste aller internen Signalquellen zur Auswahl angeboten. Die Liste setzt sich zusammen aus den möglichen Ausgabegrößen aller Analog-Out RTIO-Elemente und ggf. aus analogen Ausgabegrößen eines FPGA-Modells.

Die Anzahl der möglichen Ausgabegrößen eines Analog-Out RTIO-Elements hängt von dessen Belegung der Arbiträrgeneratoren und Direktausgabe-Kanäle ab.

Sofern ein FPGA-Modell analoge Ausgangsgrößen bereitstellt, können diese auch über eine angeschlossene ES5340.1/.2-Slave Karte ausgegeben werden. Dies setzt die Verbindung der beiden Karten über ein Gigabit-Link voraus.

**Hinweis**

*Zur Generierung der Resolver-Signale können nur die Kanäle 6 und 7 verwendet werden, da hierfür eine spezielle Signalkonditionierung auf der ES5340.1/.2 benötigt wird.*

**Reference Voltage**

Über dieses Optionsfeld kann die Referenzspannung für den D/A-Wandler des jeweiligen Ausgangskanals gewählt werden. Mögliche Werte sind „Internal“ (für eine interne Referenzspannung von 10 V) und „External“ (für eine extern anliegende Referenzspannung).

**Output Mode**

Dieser Parameter ist nur für die Kanäle 6 und 7 veränderbar. Es kann dabei zwischen einem Single-Ended- und einem differentiellen Ausgangssignal umgeschaltet werden.

**Hinweis**

*Für die korrekte Funktion der Resolversignal-Generierung ist die Einstellung „differential“ notwendig*

Bezüglich der beiden Parameter sind die folgenden Kombinationen zugelassen:

- Reference Voltage: Internal / Output Mode: Single Ended
- Reference Voltage: External / Output Mode: Single Ended
- Reference Voltage: External / Output Mode: Differential

Der Output Mode für die Kanäle 6 und 7 ist immer identisch.

22.8.2 Data (ES5340-Analog-Out-Mux)

In der vorhandenen Signalgruppe werden acht RTIO-Signale verarbeitet, die das Schließen des Ausgangsrelais eines jeden Kanals steuern. Weiterhin ist es möglich, für eine Fehlersimulation die Ausgänge permanent auf GND oder Uref zu legen.

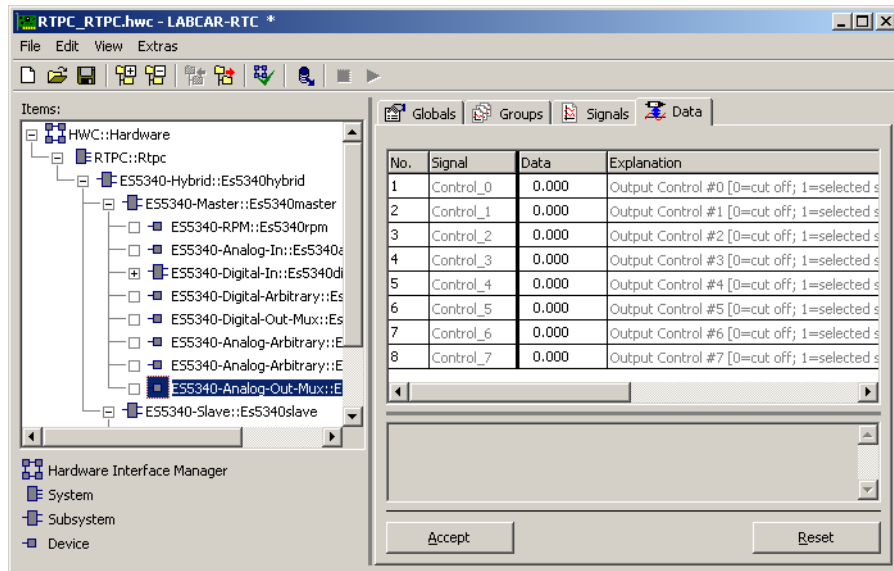


Abb. 22-50 Die Registerkarte „Data“ des ES5340-Analog-Out-Mux RTIO-Elements

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Control_n	uint8	0 = cut off 1 = ausgewählte Quelle 2 = verbunden mit GND 3 = verbunden mit URef

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

Tab. 22-24 ES5340-Analog-Out-Mux: Signale der Registerkarte „Data“

**Hinweis**  
Ist der Output Mode auf den Wert „Differential“ gesetzt, kann keine Gleichspannung ausgegeben werden.

Wenn gleichzeitig „Reference Voltage“ auf „Internal“ eingestellt ist, wird im Fall 3 (Kurzschluss nach Uref) am Ausgang 0 V ausgegeben. Ist hingegen „Reference Voltage“ auf „External“ eingestellt, wird im Fall 3 (Kurzschluss nach Uref) die am externen Referenzeingang („Excitation P/N“) angelegte Wechselspannung mit voller Amplitude ausgegeben.

## 22.9 ES5340-Slave

---

Das ES5340-Slave Subsystem dient zur Konfiguration einer ES5340-Slavekarte.

## 22.10 ES5340-Digital-Out (Slave)

---

Die Ausgabe von digitalen Signalen ist bei der Slave-Karte nur in der Betriebsart „Digital Direct Out“ mit maximal sechs Kanälen möglich.

Eine Beschreibung des ES5340-Digital-Direct-Out RTIO-Elements finden Sie im Abschnitt „ES5340-Digital-Direct-Out“ auf Seite 726.

## 22.11 ES5340-Digital-Out-Mux (Slave)

---

In dem RTIO-Element „ES5340-Digital-Out-Mux“ werden die Signale gewählt, die auf den 6 digitalen Signalausgängen der Slave-Karte ausgegeben werden.

Ausgewählt werden können folgende Signale:

- Die Signale des Typs „Digital Direct Out“ der Slave-Karte
- Die digitalen Ausgangsgrößen eines Modells der Master-Karte

Nähere Informationen zur Konfiguration der Signalausgänge finden Sie im Abschnitt „Signals (ES5340-Digital-Out-Mux)“ auf Seite 738.

## 22.12 ES5340-Analog-Out (Slave)

---

Die Ausgabe von analogen Signalen ist bei der Slave-Karte nur in der Betriebsart „Analog Direct Out“ mit maximal sechs Kanälen möglich.

Eine Beschreibung des ES5340-Analog-Direct-Out RTIO-Elements finden Sie im Abschnitt „ES5340-Analog-Direct-Out“ auf Seite 742.

## 22.13 ES5340-Analog-Out-Mux (Slave)

---

In dem RTIO-Element „ES5340-Analog-Out-Mux“ werden die Signale gewählt, die auf den 6 analogen Signalausgängen der Slave-Karte ausgegeben werden.

Ausgewählt werden können folgende Signale:

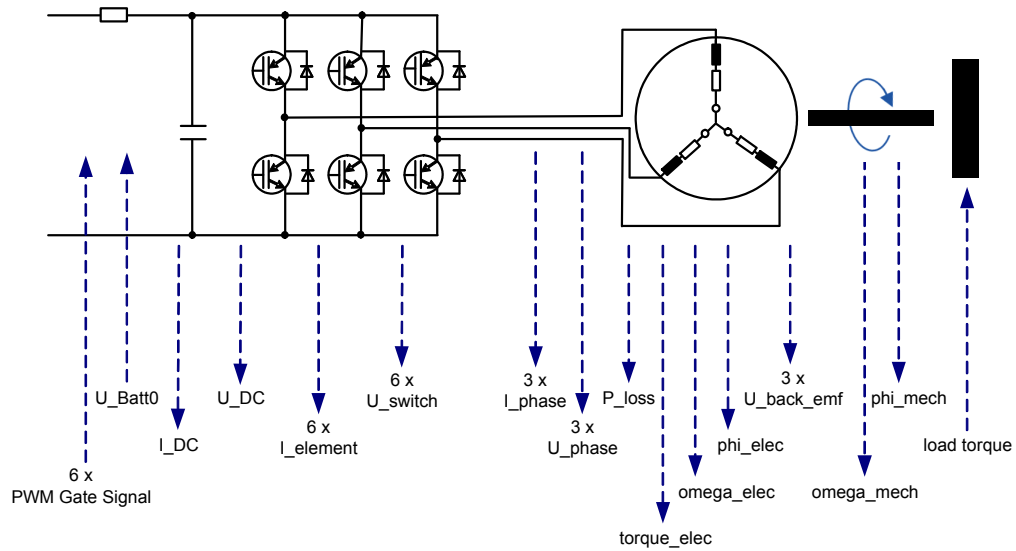
- Die Signale des Typs „Analog Direct Out“ der Slave-Karte
- Die analogen Ausgangsgrößen eines Modells der Master-Karte

Nähere Informationen zur Konfiguration der Signalausgänge finden Sie im Abschnitt „Signals (ES5340-Analog-Out-Mux)“ auf Seite 761.



## 22.14 ES5340-PMSM-1.0.0 – PMSM-FPGA-Modell

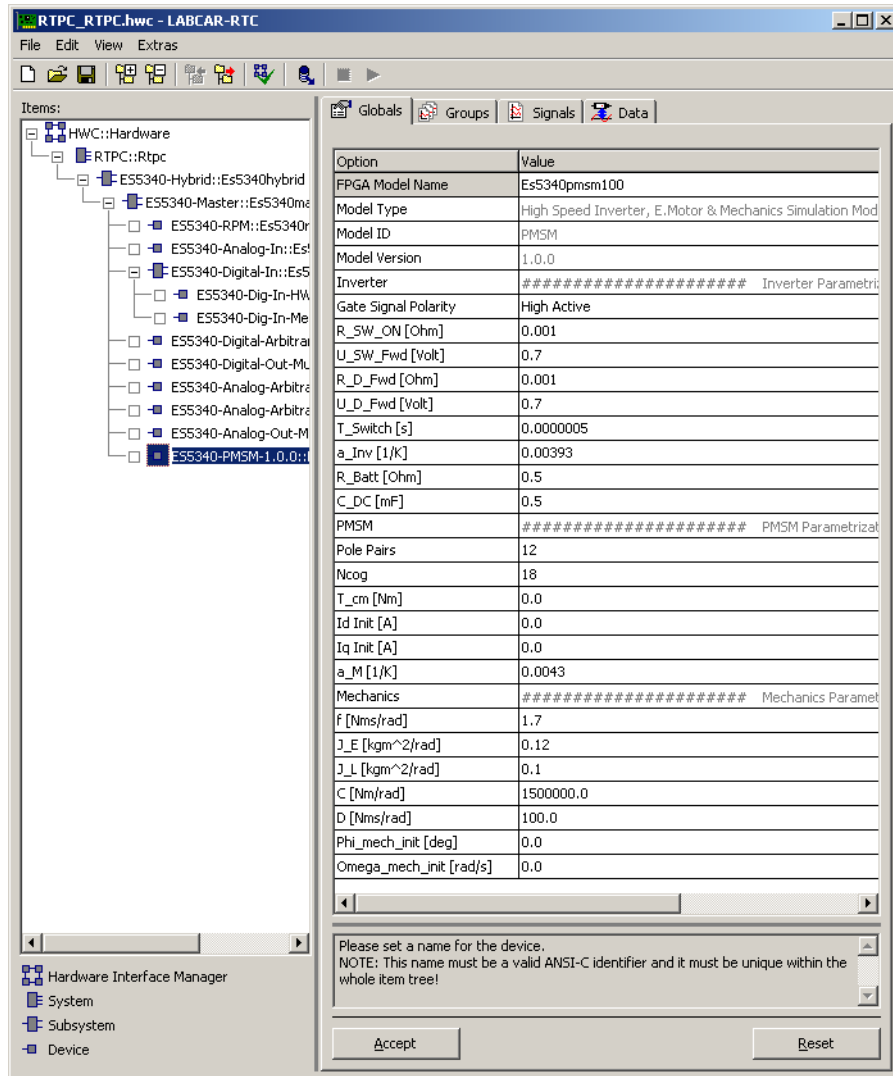
Das ES5340-PMSM-1.0.0 RTIO-Element dient zur Konfiguration des im FPGA der ES5340.1/2 implementierten Wechselrichter-, PMSM- und Mechanikmodells.



**Abb. 22-51** Blockdiagramm des ES5340-PMSM-1.0.0 Modells

22.14.1 Globals (ES5340-PMSM-1.0.0)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 22-52** Die Registerkarte „Globals“ des ES5340-PMSM-1.0.0 RTIO-Elements

Die Registerkarte „Globals“ ist unterteilt in die 4 Bereiche

- Allgemeines
- Invertermodell
- PMSM-Modell
- Mechanikmodell

22.14.2 Allgemeines

*FPGA Model Name*

Bezeichnung des Modells in der RTIO.

*Model Type*


---

 Anzeige des Modelltyps.
*Model ID*


---

 Anzeige der Modell-ID.
*Model Version*


---

 Anzeige der Modellversion.

## 22.14.3 Invertermodell

*Gate Signal Polarity*


---

 Mit dieser Einstellung kann die Polarität der an 6 (von 20) digitalen Eingängen angelegten Inverter-Gatesignale festgelegt werden.
*R\_SW\_ON*


---

 Widerstandswert eines Inverterschalters im leitenden Zustand bei 20 °C.
*U\_SW\_FWD*


---

 Spannung in Durchlassrichtung über einem Inverterschalter im leitenden Zustand.
*R\_D\_FWD*


---

 Widerstandswert der Freilaufdiode eines Inverterschalters im leitenden Zustand bei 20°C.
*U\_D\_FWD*


---

 Spannung in Durchlassrichtung über einer Diode im leitenden Zustand eines Inverterschalters.
*T\_Switch*


---

 Umschaltzeit eines Inverterschalters.
*a\_Inv*


---

 Temperaturkoeffizient der ohmschen Größen des Inverters. Mit diesem Koeffizienten lässt sich für R\_SW\_ON und R\_D\_SW eine Temperaturabhängigkeit wie folgt darstellen:

$$\begin{aligned}
 R_{SW\_ON}(\Theta_{Inv}) &= \\
 &R_{SW\_ON}(20\text{ °C}) + (1 + a_{Inv} * (\Theta_{Inv} - 20\text{ °C})) \\
 R_{D\_FWD}(\Theta_{Inv}) &= \\
 &R_{D\_FWD}(20\text{ °C}) + (1 + a_{Inv} * (\Theta_{Inv} - 20\text{ °C}))
 \end{aligned}$$

*R\_Batt*


---

 Innenwiderstand der Batterie. Für diese Größe wird keine Temperaturabhängigkeit simuliert.

Wird dieser Wert zu 0 gesetzt, wird das gesamte RC-Glied (bestehend aus R\_Batt und C\_DC) deaktiviert unabhängig davon, welchen Wert die Größe C\_DC besitzt.

Ist das RC-Glied deaktiviert, gilt:

$$I_{\text{Batt}} = I_{\text{DC}}$$

$$U_{\text{DC}} = U_{\text{Batt0}}$$

$$C_{\text{DC}}$$

Kapazität des Zwischenkreises.

Wird dieser Wert zu 0 gesetzt, wird das gesamte RC-Glied (bestehend aus R\_Batt und DC) deaktiviert, unabhängig davon, welchen Wert die Größe R\_Batt besitzt.

$$I_{\text{Batt}} = I_{\text{DC}}$$

$$U_{\text{DC}} = U_{\text{Batt0}}$$

#### 22.14.4 PMSM-Modell

##### *Pole Pairs*

Die Anzahl der Polpaare der permanent-erregten Synchronmaschine.

##### *Ncog*

Die Anzahl an Rastpositionen der permanent-erregten Synchronmaschine.

##### *T\_cm*

Der Scheitelwert des Rastmoments der permanent-erregten Synchronmaschine.

##### *Id\_Init*

d-Komponente des Initialisierungswerts des Statorstroms im rotorfesten d-q-Koordinatensystem.

##### *Iq\_Init*

q-Komponente des Initialisierungswerts des Statorstroms im rotorfesten d-q-Koordinatensystem.

##### *a\_M*

Temperaturkoeffizient der ohmschen Größen der permanent-erregten Synchronmaschine. Mit diesem Koeffizienten lässt sich für den Wicklungswiderstand R\_d bzw. R\_q eine Temperaturabhängigkeit wie folgt darstellen:

$$R_d(\text{Theta}_M) = R_d(20 \text{ °C}) + (1 + a_M * (\text{Theta}_M - 20 \text{ °C}))$$

$$R_q(\text{Theta}_M) = R_q(20 \text{ °C}) + (1 + a_M * (\text{Theta}_M - 20 \text{ °C}))$$

R\_d(Theta\_M) und R\_q(Theta\_M) sind die im Modell wirksamen Größen. Diese berechnen sich nach obigen Formeln in Abhängigkeit von den Widerstandswerten bei 20 °C (R\_d(20 °C) und R\_q(20 °C)), von der aktuellen Maschinentemperatur Theta\_M und vom Temperaturkoeffizienten a\_M (siehe Tab. 22-28 auf Seite 778).

#### 22.14.5 Mechanikmodell

Im Mechanikmodell wird ein Zwei-Massensystem bestehend aus PMSM-Maschine und Last simuliert.

Die Trägheiten dieser beiden Massen werden mit den Größen J\_E (PMSM-Maschine) und J\_L (Last bzw. Triebstrangs) vorgegeben – die Reibung des Gesamtsystems wird über den Parameter f eingestellt.

Beide Massen sind über ein Feder-Dämpfer-Element miteinander gekoppelt (Federkonstante C, Dämpfung D).

*f*

---

Reibung des mechanischen Gesamtsystems.

*J\_E*

---

Massenträgheit der Maschine.

*J\_L*

---

Massenträgheit der Last bzw. des Triebstrangs.

*C*

---

Torsionssteifigkeit bzw. Federkonstante des Kopplungselements zwischen zwei Massen.

*D*

---

Dämpfung des Kopplungselements zwischen zwei Massen.

*Phi\_mech\_init*

---

Initialisierungswert des mechanischen Lagewinkels der elektrischen Maschine/ des Rotors.

*Omega\_mech\_init*

---

Initialisierungswert der mechanischen Winkelgeschwindigkeit des E-Motors/ des Rotors.

Die folgende Tabelle fasst die Konfigurationsparameter des Modells mit Datentyp, Wertebereich und physikalischer Einheit zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Gate Signal Polarity	int	Ja	High Active Low Active
R_SW_ON	float	Ja	0,0 .. 0,05 $\Omega$
U_SW_FWD	float	Ja	0,0 .. 3,0 V
R_D_FWD	float	Ja	0,0 .. 0,05 $\Omega$
U_D_FWD	float	Ja	0,0 .. 3,0 V
T_Switch	float	Ja	0,0000002 .. 0,000001 s
a_Inv	float	Ja	0,0 .. 0,01 1/K
R_Batt	float	Ja	0,01 .. 10,0 $\Omega$
C_DC	float	Ja	0,01 .. 10,0 mF
Pole Pairs	float	Ja	1.. 64
Ncog	float	Ja	1 ..50
T_cm	float	Ja	0,0 .. 100,0 Nm
Id_Init	float	Ja	0,0 .. 1500,0 A
Iq_Init	float	Ja	0,0 .. 1500,0 A
a_M	float	Ja	0,0 .. 0,01 1/K
f	float	Ja	0,0 .. 10,0 Nms
J_E	float	Ja	0,0 .. 1,0 kgm <sup>2</sup>
J_L	float	Ja	0,0 .. 10,0 kgm <sup>2</sup>
C	float	Ja	0,0 .. 10000000,0 Nm/rad
D	float	Ja	0,0 .. 1000,0 Nms/rad
Phi_mech_init	float	Ja	0,0 .. 360,0 °
Omega_mech_init	float	Ja	-833,333 .. +833,333 rad/s

**Tab. 22-25** ES5340-PMSM-1.0.0 RTIO-Element: Konfigurationsparameter der Registerkarte „Globals“

#### 22.14.6 Groups (ES5340-PMSM-1.0.0)

Das ES5340-PMSM-1.0.0 RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des Motormodells und eine Signalgruppe „Inputs“ zum Lesen von aktuellen Statusinformationen und Simulationsgrößen des Motormodells.

#### 22.14.7 Signals (ES5340-PMSM-1.0.0)

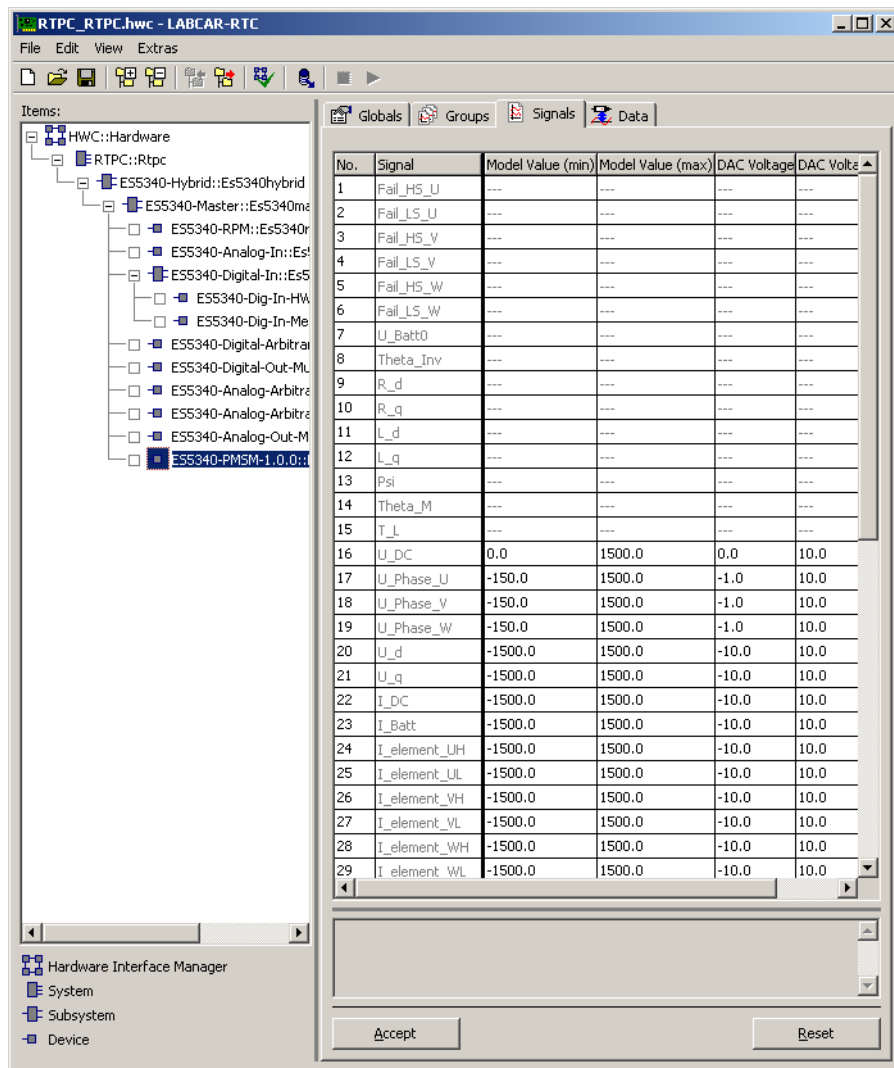
In der Registerkarte „Signals“ kann für jede analoge Modellausgabegröße festgelegt werden, welcher Wertebereich auf welchen Spannungsbereich abgebildet wird, wenn diese Größe im analogen Ausgangsmultiplexer für die Analogausgabe ausgewählt wird.

Dabei wird stets der physikalische Wert „Model Value (min)“ auf den Spannungswert „DAC Voltage (min)“ bzw. der physikalische Wert „Model Value (max)“ auf den Spannungswert „DAC Voltage (max)“ abgebildet.

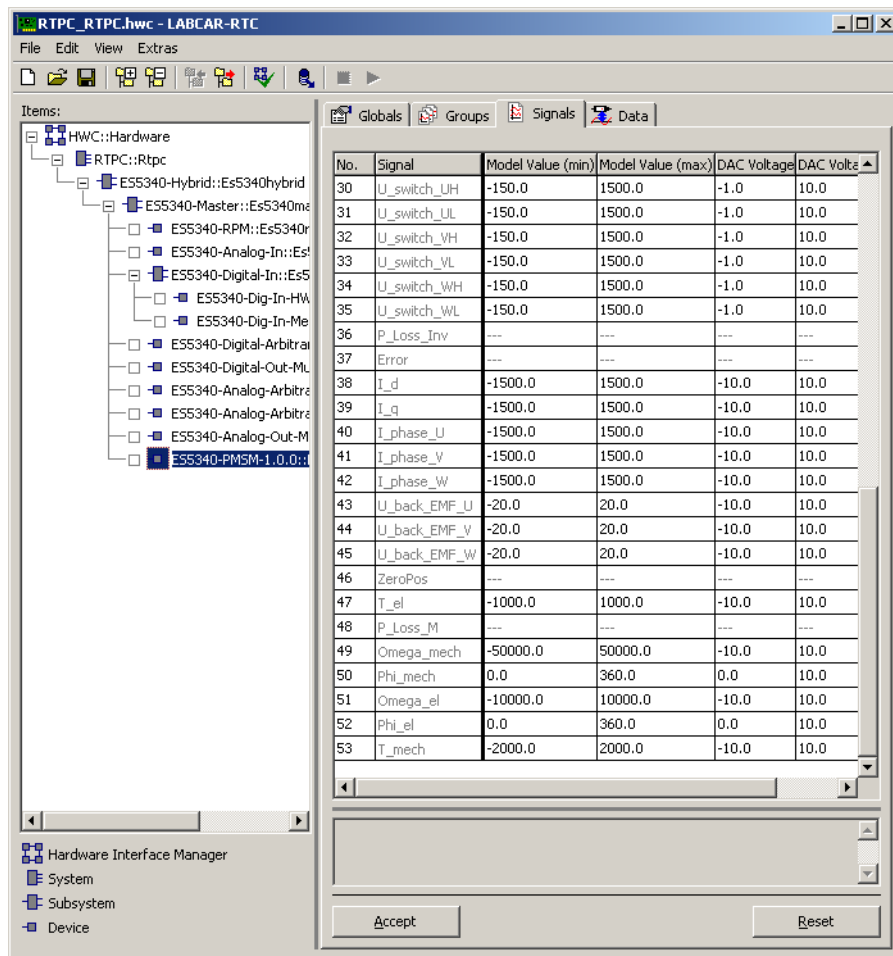
Die Min/Max-Werte für „DAC Voltage“ begrenzen gleichzeitig die Ausgangsspannung – es können keine Spannungswerte eingestellt werden, die außerhalb dieses Bereiches liegen. Wird der physikalische Wertebereich einer Simulationsgröße überschritten, stellen die Min/Max-Werte für „DAC Voltage“ eine absolute Grenze dar.

**Hinweis**

*Wird im analogen Ausgangsmultiplexer jedoch der Fehlerzustand 3 (shorted to Uref) aktiviert, ist diese Begrenzung nicht wirksam! Das bedeutet, dass die durch den Fehler definierte Spannung am Ausgang angelegt wird und diese die oben erwähnte Begrenzung u.U. überschreitet!*



**Abb. 22-53** Die Registerkarte „Signals“ des ES5340-PMSM-1.0.0 RTIO-Elements (Teil 1)



**Abb. 22-54** Die Registerkarte „Signals“ des ES5340-PMSM-1.0.0 RTIO-Elements (Teil 2)

Die folgende Tabelle fasst die Konfigurationsparameter der Registerkarte „Signals“ mit Datentyp und Wertebereich zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Model Value (min)	float	Nein	Wertebereich in Abhängigkeit von der Modellgröße
Model Value (max)	float	Nein	Wertebereich in Abhängigkeit von der Modellgröße
DAC Voltage (min)	float	Nein	Wertebereich: -10,0 V .. +10,0 V
DAC Voltage (max)	float	Nein	Wertebereich: -10,0 V .. +10,0 V

**Tab. 22-26** ES5340-PMSM-1.0.0 RTIO-Element: Konfigurationsparameter der Registerkarte „Signals“

Die folgende Tabelle enthält die Wertebereiche der Skalierungsparameter von analogen Modellgrößen auf analoge Ausgangsspannungsbereiche:



Parameter	Model Value (min)	Model Value (max)	DAC Voltage (min)	DAC Voltage (max)	Bedeutung
U_DC	0	1500	-10	10	Spannung am Zwischenkreiskondensator [V]
U_Phase_x*	-150	1500	-10	10	Phasenspannung der Phase x bezogen auf das Massepotential des Zwischenkreises [V]
U_d	-1500	1500	-10	10	d-Komponente der Spannungen im rotorfesten d-q-Koordinatensystem [V]
U_q	-1500	1500	-10	10	q-Komponente der Spannungen im rotorfesten d-q-Koordinatensystem [V]
I_DC	-1500	1500	-10	10	Strom in den Wechselrichter [A]
I_Batt	-1500	1500	-10	10	Strom durch den nachgebildeten Batterieinnenwiderstand [A]
I_element_xH*, I_element_xL*	-1500	1500	-10	10	Elementstrom durch den High-Side- bzw. Low-Side-Pfad der Phase x [A]
U_switch_xH*, U_switch_xL*	-150	1500	-10	10	Spannung über dem High-Side-, bzw. Low-Side-Schalter der Phase x [A]
I_d	-1500	1500	-10	10	d-Komponente der Ströme im rotorfesten d-q-Koordinatensystem [A]
I_q	-1500	1500	-10	10	q-Komponente der Ströme im rotorfesten d-q-Koordinatensystem [A]
I_phase_x*	-1500	1500	-10	10	Phasenstrom in der Phase x [A]
U_back_EMF_x*	-1500	1500	-10	10	Induzierte Spannung Phase x [U]
T_el	-1000	1000	-10	10	Elektrisch erzeugtes Moment [Nm]
Omega_mech	-50000	50000	-10	10	Mechanische Winkelgeschwindigkeit [rad/s]
Phi_mech	0	360	-10	10	Mechanischer Winkel, Rotorwinkel [°]
Omega_el	-50000	50000	-10	10	Elektrische Winkelgeschwindigkeit [rad/s]

Parameter	Model Value (min)	Model Value (max)	DAC Voltage (min)	DAC Voltage (max)	Bedeutung
Phi_el	0	360	-10	10	Elektrischer Winkel [°]
T_mech	-2000	2000	-10	10	Effektiv wirksames mechanisches Moment [Nm]

\*x = U,V,W

**Tab. 22-27** ES5340-PMSM-1.0.0 RTIO-Element: Wertebereiche für Signale

**Hinweis**

*DAC Voltage (min) muss stets kleiner sein als DAC Voltage (max).*

22.14.8 Data (ES5340-PMSM-1.0.0)

Die folgende Abbildung zeigt die Registerkarte „Data“.

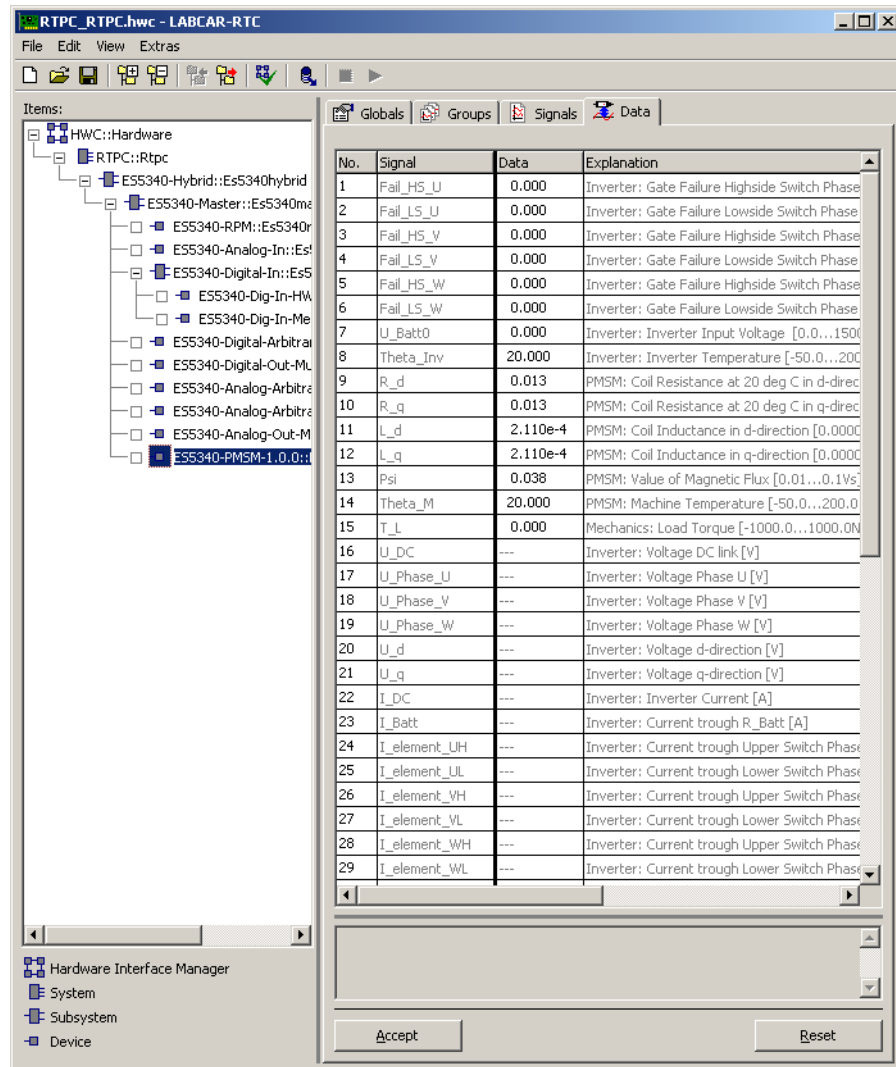
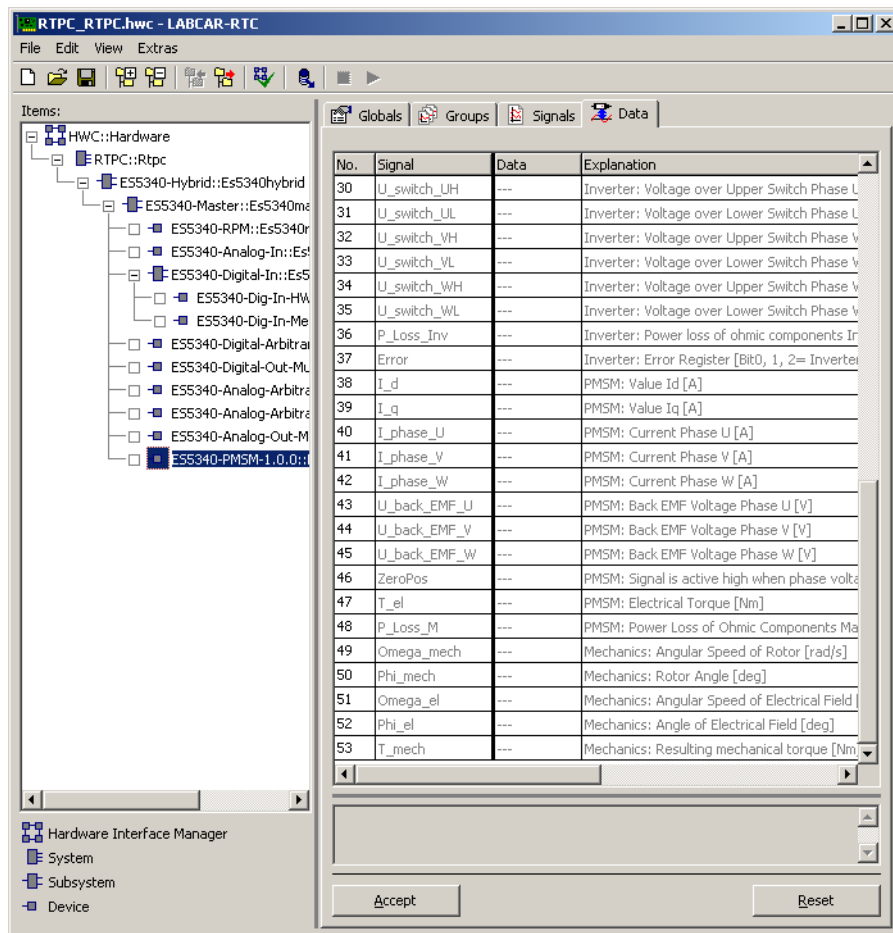


Abb. 22-55 Die Registerkarte „Data“ des ES5340-PMSM-1.0.0 RTIO-Elements (Teil 1)



**Abb. 22-56** Die Registerkarte „Data“ des ES5340-PMSM-1.0.0 RTIO-Elements (Teil 2)

In der folgenden Tabelle sind die Signale der Registerkarte „Data“ zusammengefasst.

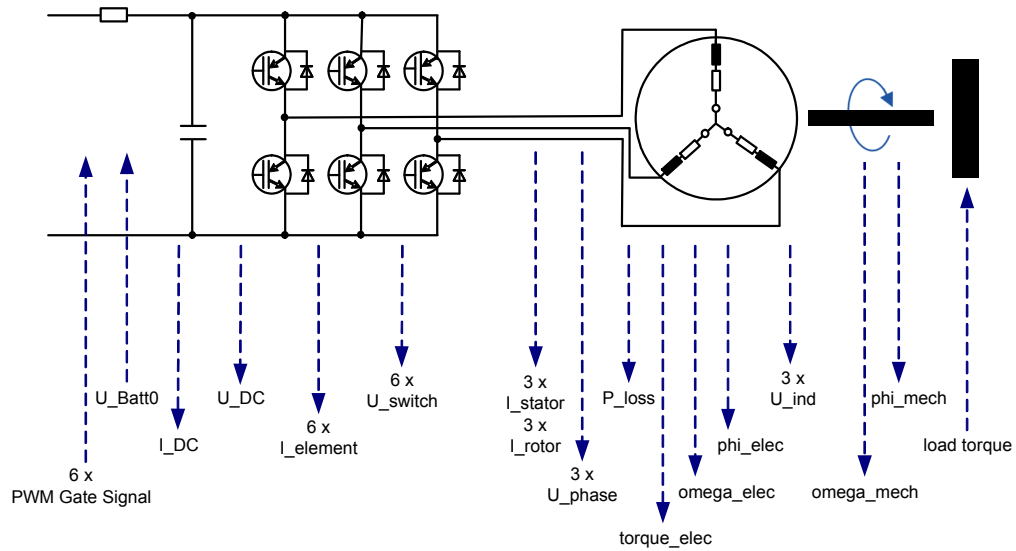
Signalname	Datentyp	Richtung	Anmerkungen
Fail_HS_x*	float	Eingang	Wechselrichter: Gate-Fehler High-Side Schalter Phase x* 0 = normale Funktion 1 = Schalter permanent aus 2 = Schalter permanent an
Fail_LS_x*	float	Eingang	Wechselrichter: Gate-Fehler Low-Side Schalter Phase x* 0 = normale Funktion 1 = Schalter permanent aus 2 = Schalter permanent an
U_Batt0	float	Eingang	Zwischenkreiseingangsspannung/ Batteriespannung Wertebereich: 0.0 .. 1500.0 V
Theta_Inv	float	Eingang	Temperatur des Wechselrichters Wertebereich: -50,0 .. 200,0 °C
R_d	float	Eingang	d-Komponente des Stator-Wicklungswiderstands bei 20 °C Wertebereich: 0,01 .. 1,0 Ω
R_q	float	Eingang	q-Komponente des Stator-Wicklungswiderstands bei 20 °C Wertebereich: 0,01 .. 1,0 Ω
L_d	float	Eingang	d-Komponente der Stator-Wicklungsinduktivität Wertebereich: 0,00008 .. 0,001 H
L_q	float	Eingang	q-Komponente der Stator-Wicklungsinduktivität Wertebereich: 0,00008 .. 0,001 H
Psi	float	Eingang	Magnetischer Fluss erzeugt durch Permanentmagnete der Maschine Wertebereich: 0,01 .. 0,1 Vs
Theta_M	float	Eingang	Temperatur in der Maschine (zur Berechnung der Wärmeabhängigkeit ohmscher Größen) Wertebereich: -50,0 .. 200,0 °C
T_L	float	Eingang	Lastmoment Positiver Wert: Moment wirkt in negativer Drehrichtung, Bremsmoment Negativer Wert: Moment wirkt in positiver Drehrichtung, beschleunigendes Moment Wertebereich: -1000,0 .. 1000,0 Nm
U_DC	float	Ausgang	Spannung am Zwischenkreiskondensator [V]
U_Phase_x*	float	Ausgang	Phasenspannung der Phase x bezogen auf das Massepotential des Zwischenkreises [V]
U_d	float	Ausgang	d-Komponente der Spannungen im rotorfesten dq-Koordinatensystem [V]
U_q	float	Ausgang	q-Komponente der Spannungen im rotorfesten dq-Koordinatensystem [V]

Signalname	Datentyp	Richtung	Anmerkungen
I_DC	float	Ausgang	Strom in den Wechselrichter [A]
I_Batt	float	Ausgang	Strom durch den nachgebildeten Batterieinnenwiderstand [A]
I_element_xH*, I_element_xL*	float	Ausgang	Elementstrom durch den High-Side- bzw. Low-Side-Pfad der Phase x [A]
U_switch_xH*, U_switch_xL*	float	Ausgang	Spannung über dem High-Side- bzw. Low-Side-Schalter der Phase x [A]
P_Loss_Inv	float	Ausgang	Verlustleistung der ohmschen Komponenten des Wechselrichters [W]
Error	float	Ausgang	Bitvektor in Gleitkommadarstellung: Bit 0: Wechselrichter-Kurzschluss High-Side/Low-Side Phase U Bit 1: Wechselrichter-Kurzschluss High-Side/Low-Side Phase V Bit 2: Wechselrichter-Kurzschluss High-Side/Low-Side Phase W
I_d	float	Ausgang	d-Komponente der Ströme im rotorfesten d-q-Koordinatensystem [A]
I_q	float	Ausgang	q-Komponente der Ströme im rotorfesten d-q-Koordinatensystem [A]
I_phase_x*	float	Ausgang	Phasenstrom in der Phase x [A]
U_back_EMF_x*	float	Ausgang	Induzierte Spannung in Phase x [V]
ZeroPos	float	Ausgang	1, wenn $U\_back\_EMF\_U \geq U\_back\_EMF\_V$ 0, wenn $U\_back\_EMF\_U < U\_back\_EMF\_V$
T_el	float	Ausgang	Elektrisch erzeugtes Moment [Nm]
P_Loss_M	float	Ausgang	Verlustleistung der ohmschen Komponenten der Maschine [W]
Omega_mech	float	Ausgang	Mechanische Winkelgeschwindigkeit [rad/s]
Phi_mech	float	Ausgang	Mechanischer Winkel, Rotorwinkel [°]
Omega_el	float	Ausgang	Elektrische Winkelgeschwindigkeit [rad/s]
Phi_el	float	Ausgang	Elektrischer Winkel [°]
T_mech	float	Ausgang	Effektiv wirksames mechanisches Moment [Nm]
* x = U,V,W			

**Tab. 22-28** ES5340-PMSM-1.0.0 RTIO-Element: Signale der Registerkarte „Data“

22.15 ES5340-IM-1.0.0 – IM-FPGA-Modell

Das ES5340-IM-1.0.0 RTIO-Element dient zur Konfiguration des im FPGA der ES5340.1/2 implementierten Wechselrichter-, IM- und Mechanikmodells.



**Abb. 22-57** Blockdiagramm des ES5340-IM-1.0.0 Modells

## 22.15.1 Globals (ES5340-IM-1.0.0)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.

Option	Value
FPGA Model Name	Es5340im100
Model Type	High Speed Inverter, Induction Motor & Mechanics Simulation Model
Model ID	IM
Model Version	1.0.0
Inverter	##### Inverter Parametrization #####
Gate Signal Polarity	High Active
R_SW_ON [Ohm]	0.001
U_SW_Fwd [Volt]	0.7
R_D_Fwd [Ohm]	0.001
U_D_Fwd [Volt]	0.7
T_Switch [s]	0.0000005
a_Inv [1/K]	0.00393
R_Batt [Ohm]	0.5
C_DC [mF]	0.5
IM	##### Induction Machine Parametrization #####
Pole Pairs	12
I_Stator_U_Init [A]	0.0
I_Stator_V_Init [A]	0.0
I_Stator_W_Init [A]	0.0
I_Rotor_U_Init [A]	0.0
I_Rotor_V_Init [A]	0.0
I_Rotor_W_Init [A]	0.0
a_M [1/K]	0.0043
Mechanics	##### Mechanics Parametrization #####
f [Nms/rad]	1.7
J_E [kgm <sup>2</sup> ]	0.12
J_L [kgm <sup>2</sup> ]	0.1
C [Nm/rad]	1500000.0
D [Nms/rad]	100.0
Phi_mech_init [deg]	0.0
Omega_mech_init [rad/s]	0.0

**Abb. 22-58** Die Registerkarte „Globals“ des ES5340-IM-1.0.0 RTIO-Elements  
Die Registerkarte „Globals“ ist in vier Bereiche unterteilt:

- Allgemeines
- Invertermodell
- IM-Modell
- Mechanikmodell

## 22.15.2 Allgemeines

*FPGA Model Name*

Bezeichnung des Modells in der RTIO.

*Model Type*

Anzeige des Modelltyps.



*Model ID*


---

Anzeige der Modell-ID.

*Model Version*


---

Anzeige der Modellversion.

## 22.15.3 Invertermodell

*Gate Signal Polarity*


---

Mit dieser Einstellung kann die Polarität der an 6 (von 20) digitalen Eingängen angelegten Inverter-Gatesignale festgelegt werden.

*R\_SW\_ON*


---

Widerstandswert eines Inverterschalters im leitenden Zustand bei 20 °C.

*U\_SW\_FWD*


---

Spannung in Durchlassrichtung über einem Inverterschalter im leitenden Zustand.

*R\_D\_FWD*


---

Widerstandswert der Freilaufdiode eines Inverterschalters im leitenden Zustand bei 20°C.

*U\_D\_FWD*


---

Spannungsabfall in Durchlassrichtung über einer Diode im leitenden Zustand eines Inverterschalters.

*T\_Switch*


---

Umschaltzeit eines Inverterschalters.

*a\_Inv*


---

Temperaturkoeffizient der ohmschen Größen des Inverters. Mit diesem Koeffizienten lässt sich für R\_SW\_ON und R\_D\_SW eine Temperaturabhängigkeit wie folgt darstellen:

$$R\_SW\_ON(\Theta\_Inv) = R\_SW\_On(20\text{ °C}) + (1 + a\_Inv * (\Theta\_Inv - 20\text{ °C}) )$$

$$R\_D\_FWD(\Theta\_Inv) = R\_D\_FWD(20\text{ °C}) + (1 + a\_Inv * (\Theta\_Inv - 20\text{ °C}) )$$

*R\_Batt*


---

Innenwiderstand der Batterie. Für diese Größe wird keine Temperaturabhängigkeit simuliert.

Wird dieser Wert zu 0 gesetzt, wird von der Software ein minimaler Wert = 0.01 eingesetzt. Dann wird das gesamte RC-Glied (bestehend aus R\_Batt und C\_DC) deaktiviert unabhängig davon, welchen Wert die Größe C\_DC besitzt.

Ist das RC-Glied deaktiviert, gilt:

$$I\_Batt = I\_DC$$

$$U\_DC = U\_Batt0$$

C\_DC

Kapazität des Zwischenkreises.

Wird dieser Wert zu 0 gesetzt, wird von der Software ein minimaler Wert = 0.01 eingesetzt. Dann wird das gesamte RC-Glied (bestehend aus R\_Batt und DC) deaktiviert, unabhängig davon, welchen Wert die Größe R\_Batt besitzt.

$$I_{Batt} = I_{DC}$$

$$U_{DC} = U_{Batt0}$$

## 22.15.4 IM-Modell

Pole Pairs

Die Anzahl der Polpaare der Induktionsmaschine.

T\_cm

Der Scheitelwert des Rastmoments der Induktionsmaschine.

I\_Stator\_X\_Init (X = U,V,W)

Initialisierungswert des Statorstroms der Phase X.

I\_Rotor\_X\_Init (X = U,V,W)

Initialisierungswert des Rotorstroms der Phase X.

a\_M

Temperaturkoeffizient der ohmschen Größen der Induktionsmaschine.

Für die Berechnung der Wicklungswiderstände R\_s1 (Stator) bzw. R\_s2 (Rotor) wird mit diesen Temperaturkoeffizienten nach folgender Gleichung eine Temperaturabhängigkeit realisiert:

$$R_{s1}(\Theta_M) = R_{s1}(20\text{ °C}) + (1 + a_M * (\Theta_M - 20\text{ °C}))$$

$$R_{s2}(\Theta_M) = R_{s2}(20\text{ °C}) + (1 + a_M * (\Theta_M - 20\text{ °C}))$$

R\_s1(Theta\_M) und R\_s2(Theta\_M) sind die im Modell wirksamen Größen. Diese berechnen sich nach obigen Formeln in Abhängigkeit von den Widerstandswerten bei 20 °C (R\_s1(20 °C) und R\_s2(20 °C), Eingabe in der Registerkarte „Signals“ als R\_s1 und R\_s2), von der aktuellen Maschinentemperatur Theta\_M und vom Temperaturkoeffizienten a\_M (siehe Tab. 22-32 auf Seite 791).

## 22.15.5 Mechanikmodell

Im Mechanikmodell wird ein Zwei-Massensystem – bestehend aus dem Rotor der Induktionsmaschine und der Last – simuliert.

Die Trägheiten dieser beiden Massen werden mit den Größen J\_E (Induktionsmaschine) und J\_L (Last bzw. Triebstrangs) vorgegeben – die Reibung des Gesamtsystems wird über den Parameter f eingestellt.

Beide Massen sind über ein Feder-Dämpfer-Element miteinander gekoppelt (Federkonstante C, Dämpfung D).

f

Reibung des mechanischen Gesamtsystems.

*J\_E*

---

Massenträgheit des Rotors der Induktionsmaschine.

*J\_L*

---

Massenträgheit der Last bzw. des Triebstrangs.

*C*

---

Torsionssteifigkeit bzw. Federkonstante des Kopplungselements zwischen zwei Massen.

*D*

---

Dämpfung des Kopplungselements zwischen zwei Massen.

*Phi\_mech\_init*

---

Initialisierungswert des mechanischen Lagewinkels der elektrischen Maschine/ des Rotors.

*Omega\_mech\_init*

---

Initialisierungswert der mechanischen Winkelgeschwindigkeit des E-Motors/ des Rotors.

Die folgende Tabelle fasst die Konfigurationsparameter des Modells mit Datentyp, Wertebereich und physikalischer Einheit zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Gate Signal Polarity	int	Ja	High Active Low Active
R_SW_ON	float	Ja	0,0 .. 0,05 $\Omega$
U_SW_FWD	float	Ja	0,0 .. 3,0 V
R_D_FWD	float	Ja	0,0 .. 0,05 $\Omega$
U_D_FWD	float	Ja	0,0 .. 3,0 V
T_Switch	float	Ja	0,0000002 .. 0,000001 s
a_Inv	float	Ja	0,0 .. 0,01 1/K
R_Batt	float	Ja	0,01 .. 10,0 $\Omega$
C_DC	float	Ja	0,01 .. 10,0 mF
Pole Pairs	float	Ja	1.. 16
I_Stator_X_Init (X = U,V,W)	float	Ja	-1500,0 .. 1500,0 A
I_Rotor_X_Init (X = U,V,W)	float	Ja	-1500 .. 1500,0 A
a_M	float	Ja	0,0 .. 0,01 1/K
f	float	Ja	0,0 .. 10,0 Nms
J_E	float	Ja	0,0 .. 1,0 kgm <sup>2</sup>
J_L	float	Ja	0,0 .. 10,0 kgm <sup>2</sup>
C	float	Ja	0,0 .. 10000000,0 Nm/rad
D	float	Ja	0,0 .. 1000,0 Nms/rad
Phi_mech_init	float	Ja	0,0 .. 360,0 °
Omega_mech_init	float	Ja	-833,333 .. +833,333 rad/s

**Tab. 22-29** ES5340-IM-1.0.0 RTIO-Element: Konfigurationsparameter der Registerkarte „Globals“

#### 22.15.6 Groups (ES5340-IM-1.0.0)

Das ES5340-IM-1.0.0 RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des Motormodells und eine Signalgruppe „Inputs“ zum Lesen von aktuellen Statusinformationen und Simulationsgrößen des Motormodells.

#### 22.15.7 Signals (ES5340-IM-1.0.0)

In der Registerkarte „Signals“ kann für jede analoge Modellausgabegröße festgelegt werden, welcher Wertebereich auf welchen Spannungsbereich abgebildet wird, wenn diese Größe im analogen Ausgangsmultiplexer für die Analogausgabe ausgewählt wird.

Dabei wird stets der physikalische Wert „Model Value (min)“ auf den Spannungswert „DAC Voltage (min)“ bzw. der physikalische Wert „Model Value (max)“ auf den Spannungswert „DAC Voltage (max)“ abgebildet.

Die Min/Max-Werte für „DAC Voltage“ begrenzen gleichzeitig die Ausgangsspannung – es können keine Spannungswerte eingestellt werden, die außerhalb dieses Bereiches liegen. Wird der physikalische Wertebereich einer Simulationsgröße überschritten, stellen die Min/Max-Werte für „DAC Voltage“ eine absolute Grenze dar.

**Hinweis**

*Wird im analogen Ausgangsmultiplexer jedoch der Fehlerzustand 3 (shorted to Uref) aktiviert, ist diese Begrenzung nicht wirksam! Das bedeutet, dass die durch den Fehler definierte Spannung am Ausgang angelegt wird und diese die oben erwähnte Begrenzung u.U. überschreitet!*

No.	Signal	Model Value (min)	Model Value (max)	DAC Voltage (min)	DAC Voltage (max)
1	Fail_HS_U	---	---	---	---
2	Fail_LS_U	---	---	---	---
3	Fail_HS_V	---	---	---	---
4	Fail_LS_V	---	---	---	---
5	Fail_HS_W	---	---	---	---
6	Fail_LS_W	---	---	---	---
7	U_Batt0	---	---	---	---
8	Theta_Inv	---	---	---	---
9	R_s1	---	---	---	---
10	R_s2	---	---	---	---
11	L_m	---	---	---	---
12	L_s1	---	---	---	---
13	L_s2	---	---	---	---
14	Theta_M	---	---	---	---
15	T_L	---	---	---	---
16	I_Batt	-1500.0	1500.0	-10.0	10.0
17	U_DC	0.0	1500.0	0.0	10.0
18	I_DC	-1500.0	1500.0	-10.0	10.0
19	I_element_UH	-1500.0	1500.0	-10.0	10.0
20	I_element_UL	-1500.0	1500.0	-10.0	10.0
21	I_element_VH	-1500.0	1500.0	-10.0	10.0
22	I_element_VL	-1500.0	1500.0	-10.0	10.0
23	I_element_WH	-1500.0	1500.0	-10.0	10.0
24	I_element_WL	-1500.0	1500.0	-10.0	10.0
25	U_switch_UH	-150.0	1500.0	-1.0	10.0
26	U_switch_UL	-150.0	1500.0	-1.0	10.0
27	U_switch_VH	-150.0	1500.0	-1.0	10.0
28	U_switch_VL	-150.0	1500.0	-1.0	10.0
29	U_switch_WH	-150.0	1500.0	-1.0	10.0
30	U_switch_WL	-150.0	1500.0	-1.0	10.0
31	U_phase_U	-150.0	1500.0	-1.0	10.0
32	U_phase_V	-150.0	1500.0	-1.0	10.0
33	U_phase_W	-150.0	1500.0	-1.0	10.0
34	P_Loss_Inv	---	---	---	---
35	Error	0.0	0.0	0.0	0.0
36	I_stator_U	-1500.0	1500.0	-10.0	10.0
37	I_stator_V	-1500.0	1500.0	-10.0	10.0
38	I_stator_W	-1500.0	1500.0	-10.0	10.0
39	I_rotor_U	-1500.0	1500.0	-10.0	10.0
40	I_rotor_V	-1500.0	1500.0	-10.0	10.0
41	I_rotor_W	-1500.0	1500.0	-10.0	10.0
42	U_ind_U	-1000.0	1000.0	-10.0	10.0
43	U_ind_V	-1000.0	1000.0	-10.0	10.0
44	U_ind_W	-1000.0	1000.0	-10.0	10.0
45	T_el	-1000.0	1000.0	-10.0	10.0
46	P_Loss_M	---	---	---	---
47	Omega_mech	-50000.0	50000.0	-10.0	10.0
48	Phi_mech	0.0	360.0	0.0	10.0
49	Omega_el	-10000.0	10000.0	-10.0	10.0
50	Phi_el	0.0	360.0	0.0	10.0
51	T_mech	-2000.0	2000.0	-10.0	10.0

**Abb. 22-59** Die Registerkarte „Signals“ des ES5340-IM-1.0.0 RTIO-Elements

Die folgende Tabelle fasst die Konfigurationsparameter der Registerkarte „Signals“ mit Datentyp und Wertebereich zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Model Value (min)	float	Nein	Wertebereich in Abhängigkeit von der Modellgröße
Model Value (max)	float	Nein	Wertebereich in Abhängigkeit von der Modellgröße
DAC Voltage (min)	float	Nein	Wertebereich: -10,0 V .. +10,0 V
DAC Voltage (max)	float	Nein	Wertebereich: -10,0 V .. +10,0 V

**Tab. 22-30** ES5340-IM-1.0.0 RTIO-Element: Konfigurationsparameter der Registerkarte „Signals“

Die folgende Tabelle enthält die Wertebereiche der Skalierungsparameter von analogen Modellgrößen auf analoge Ausgangsspannungsbereiche:

Parameter	Model Value (min)	Model Value (max)	DAC Voltage (min)	DAC Voltage (max)	Bedeutung
L_Batt	-1500	1500	-10	10	Strom durch den nachgebildeten Batterie-Innenwiderstand [A]
U_DC	0	1500	-10	10	Spannung am Zwischenkreiskondensator [V]
I_DC	-1500	1500	-10	10	Strom in den Wechselrichter [A]
I_element_xH*, I_element_xL*	-1500	1500	-10	10	Elementstrom durch den High-Side- bzw. Low-Side-Pfad der Phase x [A]
U_switch_xH*, U_switch_xL*	-150	1500	-10	10	Spannung über dem High-Side-, bzw. Low-Side-Schalter der Phase x [A]
U_phase_x	-150	1500	-10	10	Phasenspannung der Phase x bezogen auf das Massepotential des Zwischenkreises [V]
Error	–	–	-10	10	Gehaltenes Overflow-Bit aus Fehler-Register
I_stator_x*	-1500	1500	-10	10	Strom in Statorwicklung Phase x [A]
I_rotor_x*	-1500	1500	-10	10	Strom in Rotorwicklung Phase x [A]
I_phase_x*	-1500	1500	-10	10	Phasenstrom in der Phase x [A]
U_ind_x*	-1500	1500	-10	10	Induzierte Spannung der Phase x [U]
T_el	-1000	1000	-10	10	Elektrisch erzeugtes Drehmoment [Nm]
Omega_mech	-50000	50000	-10	10	Mechanische Winkelgeschwindigkeit [rad/s]
Phi_mech	0	360	-10	10	Mechanischer Winkel, Rotorwinkel [°]
Omega_el	-50000	50000	-10	10	Elektrische Winkelgeschwindigkeit [rad/s]
Phi_el	0	360	-10	10	Elektrischer Winkel [°]
T_mech	-2000	2000	-10	10	Effektiv wirksames mechanisches Drehmoment [Nm]
*x = U,V,W					

**Tab. 22-31** ES5340-IM-1.0.0 RTIO-Element: Wertebereiche für Signale



**Hinweis**

*DAC Voltage (min) muss stets kleiner sein als DAC Voltage (max).*

22.15.8 Data (ES5340-IM-1.0.0)

Die folgende Abbildung zeigt die Registerkarte „Data“.

No.	Signal	Data	Explanation
1	Fail_HS_U	0.000	Inverter: Gate Failure Highside Switch Phase U [0=normal operation; 1=stuck at low; 2=
2	Fail_LS_U	0.000	Inverter: Gate Failure Lowside Switch Phase U [0=normal operation; 1=stuck at low; 2=
3	Fail_HS_V	0.000	Inverter: Gate Failure Highside Switch Phase V [0=normal operation; 1=stuck at low; 2=
4	Fail_LS_V	0.000	Inverter: Gate Failure Lowside Switch Phase V [0=normal operation; 1=stuck at low; 2=
5	Fail_HS_W	0.000	Inverter: Gate Failure Highside Switch Phase W [0=normal operation; 1=stuck at low; 2=
6	Fail_LS_W	0.000	Inverter: Gate Failure Lowside Switch Phase W [0=normal operation; 1=stuck at low; 2=
7	U_Batt0	0.000	Inverter: Inverter Input Voltage [0.0...1500V]
8	Theta_Inv	20.000	Inverter: Inverter Temperature [-50.0...200.0 deg C]
9	R_s1	0.100	IM: Stator Resistance at 20 deg C [0.01...1.0Ohm]
10	R_s2	0.100	IM: Rotor Resistance at 20 deg C [0.01...1.0Ohm]
11	L_m	2.110e-4	IM: Effective Inductance [0.00008...0.001H]
12	L_s1	2.110e-4	IM: Leakage Inductance Stator [0.00008...0.001H]
13	L_s2	2.110e-4	IM: Leakage Inductance Rotor [0.00008...0.001H]
14	Theta_M	20.000	IM: Machine Temperature [-50.0...200.0 deg C]
15	T_L	0.000	Mechanics: Load Torque [-1000.0...1000.0Nm]
16	I_Batt	---	Inverter: Current trough R_Batt [A]
17	U_DC	---	Inverter: Voltage DC link [V]
18	I_DC	---	Inverter: Inverter Current [A]
19	I_element_UH	---	Inverter: Current trough Upper Switch Phase U [A]
20	I_element_UL	---	Inverter: Current trough Lower Switch Phase U [A]
21	I_element_VH	---	Inverter: Current trough Upper Switch Phase V [A]
22	I_element_VL	---	Inverter: Current trough Lower Switch Phase V [A]
23	I_element_WH	---	Inverter: Current trough Upper Switch Phase W [A]
24	I_element_WL	---	Inverter: Current trough Lower Switch Phase W [A]
25	U_switch_UH	---	Inverter: Voltage over Upper Switch Phase U [V]
26	U_switch_UL	---	Inverter: Voltage over Lower Switch Phase U [V]
27	U_switch_VH	---	Inverter: Voltage over Upper Switch Phase V [V]
28	U_switch_VL	---	Inverter: Voltage over Lower Switch Phase V [V]
29	U_switch_WH	---	Inverter: Voltage over Upper Switch Phase W [V]
30	U_switch_WL	---	Inverter: Voltage over Lower Switch Phase W [V]
31	U_phase_U	---	Inverter: Voltage Phase U [V]
32	U_phase_V	---	Inverter: Voltage Phase V [V]
33	U_phase_W	---	Inverter: Voltage Phase W [V]
34	P_Loss_Inv	---	Inverter: Power loss of ohmic components Inverter [W]
35	Error	---	Inverter: Error Register [Bit0, 1, 2= Inverter-KS HS-LSPhase U, V, W]
36	I_stator_U	---	IM: Stator Current Phase U [A]
37	I_stator_V	---	IM: Stator Current Phase V [A]
38	I_stator_W	---	IM: Stator Current Phase W [A]
39	I_rotor_U	---	IM: Rotor Current Phase U [A]
40	I_rotor_V	---	IM: Rotor Current Phase V [A]
41	I_rotor_W	---	IM: Rotor Current Phase W [A]
42	U_ind_U	---	IM: Induced Voltage Phase U [V]
43	U_ind_V	---	IM: Induced Voltage Phase V [V]
44	U_ind_W	---	IM: Induced Voltage Phase W [V]
45	T_el	---	IM: Electrical Torque [Nm]
46	P_Loss_M	---	IM: Power Loss of Ohmic Components Machine [W]
47	Omega_mech	---	Mechanics: Angular Speed of Rotor [rad/s]
48	Phi_mech	---	Mechanics: Rotor Angle [deg]
49	Omega_el	---	Mechanics: Angular Speed of Electrical Field [rad/s]
50	Phi_el	---	Mechanics: Angle of Electrical Field [deg]
51	T_mech	---	Mechanics: Resulting mechanical torque [Nm]

**Abb. 22-60** Die Registerkarte „Data“ des ES5340-IM-1.0.0 RTIO-Elements

In der folgenden Tabelle sind die Signale der Registerkarte „Data“ zusammengefasst.

Signalname	Datentyp	Richtung	Anmerkungen
Fail_HS_x*	float	Eingang	Wechselrichter: Gate-Fehler High-Side Schalter Phase x* 0 = normale Funktion 1 = Schalter permanent aus 2 = Schalter permanent an
Fail_LS_x*	float	Eingang	Wechselrichter: Gate-Fehler Low-Side Schalter Phase x* 0 = normale Funktion 1 = Schalter permanent aus 2 = Schalter permanent an
U_Batt0	float	Eingang	Zwischenkreiseingangsspannung/ Batteriespannung Wertebereich: 0.0 .. 1500.0 V
Theta_Inv	float	Eingang	Temperatur des Wechselrichters Wertebereich: -50,0 .. 200,0 °C
R_s1	float	Eingang	Stator-Wicklungswiderstand bei 20 °C Wertebereich: 0,01 .. 1,0 Ω
R_s2	float	Eingang	Rotor-Wicklungswiderstand bei 20 °C Wertebereich: 0,01 .. 1,0 Ω
L_m	float	Eingang	Effektive Induktivität Wertebereich: 0,00008 .. 0,001 H
L_s1	float	Eingang	Stator-Wicklungsinduktivität Wertebereich: 0,00008 .. 0,001 H
L_s2	float	Eingang	Rotor-Wicklungsinduktivität Wertebereich: 0,00008 .. 0,001 H
Theta_M	float	Eingang	Temperatur in der Maschine (zur Berechnung der Wärmeabhängigkeit ohmscher Größen) Wertebereich: -50,0 .. 200,0 °C
T_L	float	Eingang	Lastmoment Positiver Wert: Moment wirkt in negativer Drehrichtung, Bremsmoment Negativer Wert: Moment wirkt in positiver Drehrichtung, beschleunigendes Moment Wertebereich: -1000,0 .. 1000,0 Nm
U_DC	float	Ausgang	Spannung am Zwischenkreiskondensator [V]
I_DC	float	Ausgang	Strom in den Wechselrichter [A]
I_element_xH*, I_element_xL*	float	Ausgang	Elementstrom durch den High-Side- bzw. Low-Side-Pfad der Phase x [A]
U_switch_xH*, U_switch_xL*	float	Ausgang	Spannung über dem High-Side- bzw. Low-Side-Schalter der Phase x [A]
U_Phase_x*	float	Ausgang	Phasenspannung der Phase x bezogen auf das Massepotential des Zwischenkreises [V]
P_Loss_Inv	float	Ausgang	Verlustleistung der ohmschen Komponenten des Wechselrichters [W]

Signalname	Datentyp	Richtung	Anmerkungen
I_Batt	float	Ausgang	Strom durch den nachgebildeten Batterieinnenwiderstand [A]
Error	float	Ausgang	Bitvektor in Gleitkommadarstellung: Bit 0: Wechselrichter-Kurzschluss High-Side/Low-Side Phase U Bit 1: Wechselrichter-Kurzschluss High-Side/Low-Side Phase V Bit 2: Wechselrichter-Kurzschluss High-Side/Low-Side Phase W Bit 3: Überlauf bei einem 52-Bit Festkommawert im IM-Modell Bit 4: Überlauf bei einem 52-Bit Festkommawert im Mechanik-Modell
I_stator_x*	float	Ausgang	Strom in Statorwicklung Phase x [A]
I_rotor_x*	float	Ausgang	Strom in Rotorwicklung Phase x [A]
U_ind_x*	float	Ausgang	Induzierte Spannung in Phase x [V]
T_el	float	Ausgang	Elektrisch erzeugtes Moment [Nm]
P_Loss_M	float	Ausgang	Verlustleistung der ohmschen Komponenten der Maschine [W]
Omega_mech	float	Ausgang	Mechanische Winkelgeschwindigkeit [rad/s]
Phi_mech	float	Ausgang	Mechanischer Winkel, Rotorwinkel [°]
Omega_el	float	Ausgang	Elektrische Winkelgeschwindigkeit [rad/s]
Phi_el	float	Ausgang	Elektrischer Winkel [°]
T_mech	float	Ausgang	Effektiv wirksames mechanisches Moment [Nm]
* x = U,V,W			

**Tab. 22-32** ES5340-IM-1.0.0 RTIO-Element: Signale der Registerkarte „Data“



## 23 ES5340.2 Internal Combustion Engine Application

In diesem Kapitel wird die Hardwarekonfiguration der ES5340.2 Internal Combustion Engine Application (kurz: ES5340.2-ICE) beschrieben. Näheres zu Funktionsweise der Karte finden Sie im ES5340.2 Internal Combustion Engine Application Benutzerhandbuch.

### Aufbau der RTIO-Hierarchie

Im Folgenden wird der Aufbau der RTIO-Hierarchie im RTC Editor sowie die Konfigurations- und Echtzeitdaten der einzelnen Items näher beschrieben.

Um im LABCAR RTC Editor eine neue ES5340.2-ICE-Hierarchie aufzubauen, fügen Sie ein ES5340-ICE Subsystem auf der RTPC-Systemebene ein. Dieses RTIO-Element dient als Container für die Beschreibung einer Master- und wahlweise einer Slave-Karte sowie der Subsysteme zur Signalvermessung und Signalgenerierung.

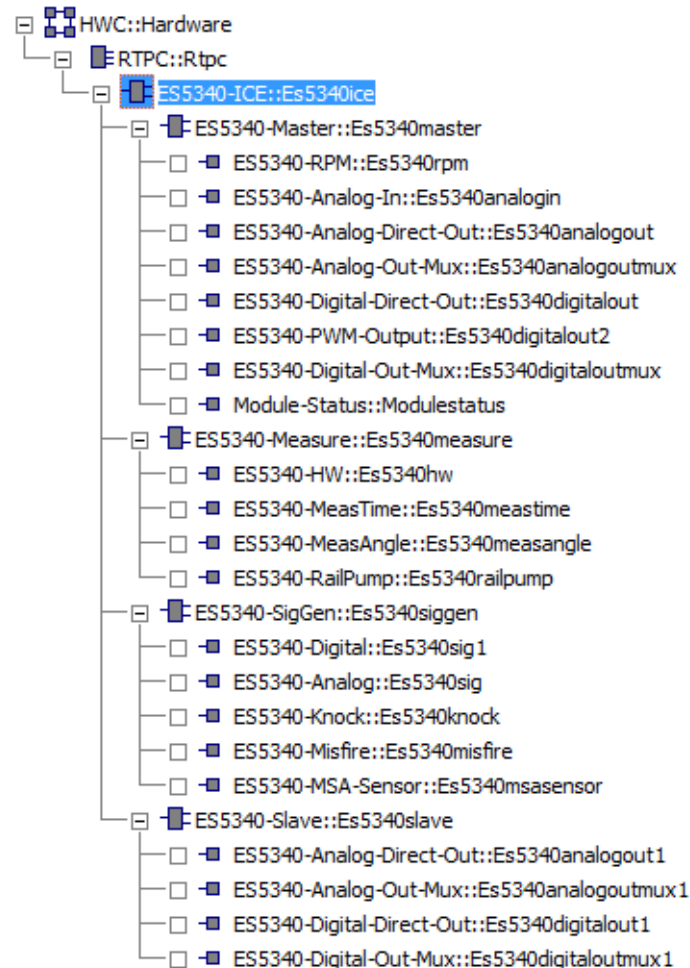


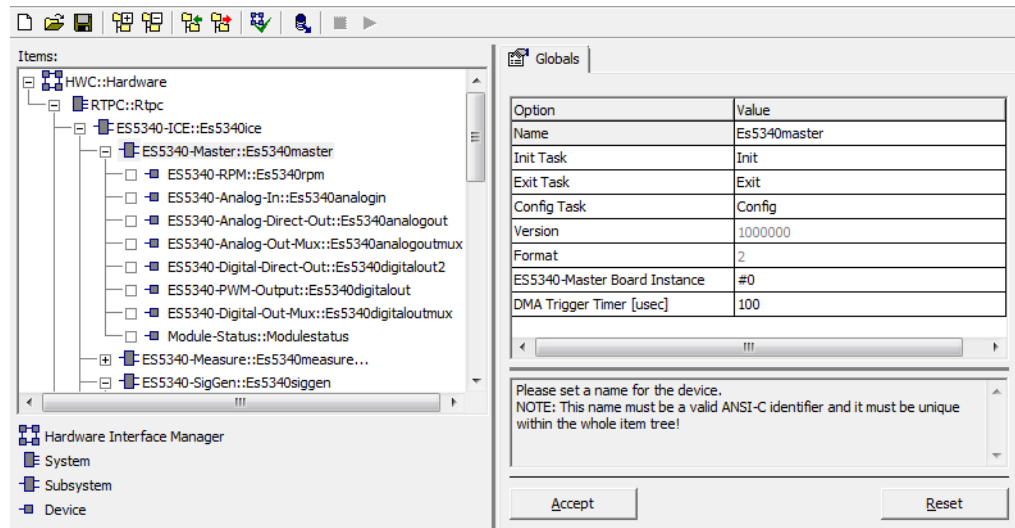
Abb. 23-1 Vollständiger ES5340-ICE RTIO-Baum

## 23.1 ES5340-Master

Das ES5340-Master Subsystem enthält die mit der ES5340-Hybrid gemeinsam genutzten RTIO Elemente und deren Einstellungen.

### 23.1.1 Globals (ES5340-Master Subsystem)

In diesem Abschnitt können Sie globale Einstellungen für die ES5340-Masterkarte vornehmen.



**Abb. 23-2** Registerkarte „Globals“ des ES5340-Master Subsystems (Option „Show All“)

#### *ES5340-Master Board Instance*

Diese Optionsfeld dient zur Festlegung der Karteninstanz, die diesem RTIO-Subsystem zugeordnet ist. Die Nummerierung beginnt bei 0 und die Karten eines Typs werden (ähnlich wie beim VMEbus-System) von links nach rechts aufsteigend nummeriert.<sup>1</sup>

Hierzu muss jedoch der interne Winkeltaktbus (SYNC-Interface) über alle Karten verbunden sein (siehe Benutzerhandbuch) – die Karteninstanz kann während des Experiments nicht verändert werden.

#### *DMA Trigger Time [μsec]*

Dieses Feld wird nur angezeigt, wenn im Kontextmenü (rechte Maustaste) der Eintrag „Show All“ gewählt wurde.

Sie können hier den zeitlichen Abstand für die Aktivierung zweier asynchroner DMA-Übertragungen wählen – die gemessenen Daten werden von der ES5340.2-ICE-Masterkarte direkt in den PC-Hauptspeicher geschrieben.

<sup>1</sup>. Gilt nur unter der Voraussetzung, dass die Karten über die SYNC-Bus-Leitungen verbunden sind. Ansonsten erfolgt die Nummerierung zufällig bzw. hardwareabhängig.

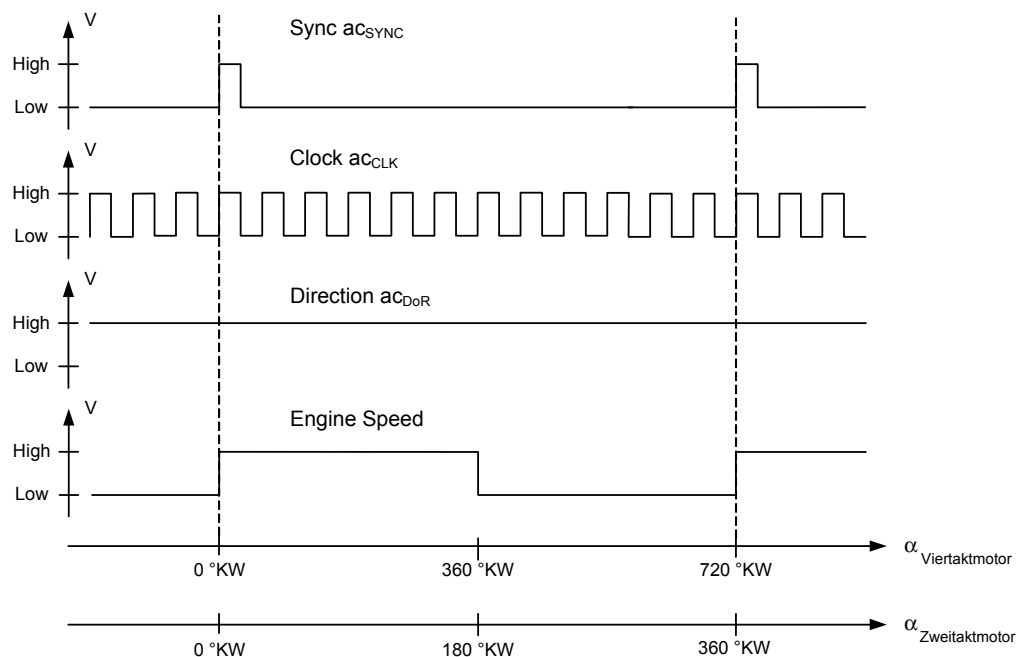
Dies betrifft die letzten vier Zeitstempel aller 20 digitalen Eingänge, die aktuellen Messwerte der vier Analogwerte, die RTIO-Eingangsgrößen der Drehzahleinheit und Statusinformationen für Klopfen.

Dieses Datum kann während der Simulation durch Rekonfiguration verändert werden.

- Defaultwert: 100  $\mu$ s
- Einstellbereich: 5  $\mu$ s ... 1 s

## 23.2 ES5340-RPM – Drehzahleinheit

Die winkelbasierte Synchronisation zwischen LABCAR-Karten erfolgt über ein aus drei Komponenten bestehendes Winkeltaktsignal (Abb. 23-3). Das Signal „ac<sub>SYNC</sub>“ zeigt einen Nulldurchgang des Kurbelwellenwinkels beim Verbrennungsmotor an, „ac<sub>CLK</sub>“ transportiert den eigentlichen Winkeltakt, „ac<sub>DoR</sub>“ codiert die Drehrichtung des Motors.



**Abb. 23-3** Die drei Komponenten des Winkeltaktsignals

Das Taktsignal bietet eine Auflösung  $\delta_{\alpha}$  von 16 Bit pro 720°KW des Viertaktmotors bzw. 360°KW des Zweitaktmotors.

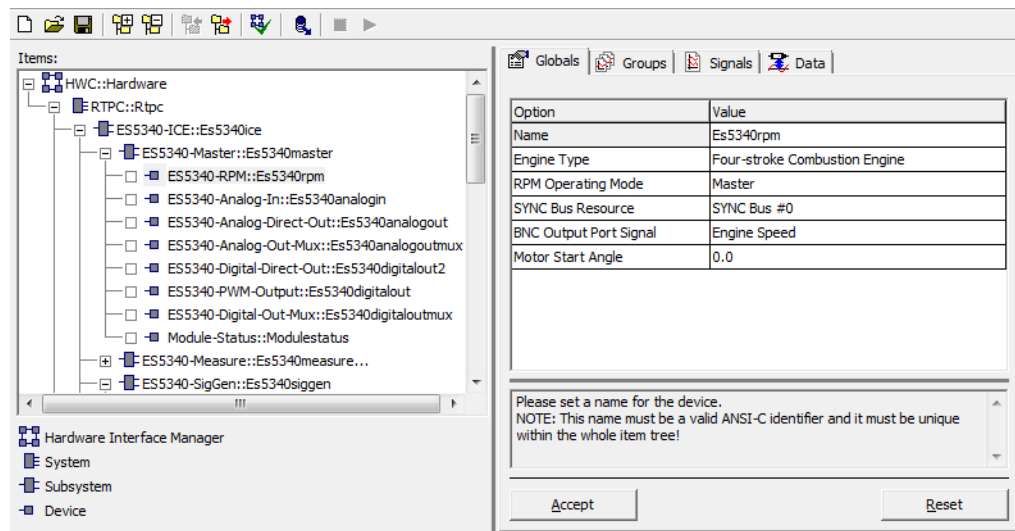
Die Drehzahleinheit der ES5340.2-ICE kann im Slave- und Master-Modus betrieben werden. Im Slave-Modus synchronisiert sich die Drehzahleinheit auf einen (von einer anderen Karte im System) generierten Winkeltakt. Im Master-Modus generiert die Drehzahleinheit den Winkeltakt für andere Karten im System. Zur Übertragung des Winkeltakts zwischen den Karten stehen zwei Busse zur Verfügung. An einem Bus kann es jeweils nur einen Master geben.

### 23.2.1 ES5340-RPM – Drehzahleinheit

Das ES5340-RPM Device dient zu Konfiguration und Steuerung der ES5340-Drehzahleinheit.

### 23.2.2 Globals (ES5340-RPM)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 23-4** Die Registerkarte „Globals“ des ES5340-RPM Device

#### Engine Type

Auswahl des Motortyps.

#### RPM Operating Mode

Auswahl zwischen Master- oder Slave-Betrieb.

#### SYNC Bus Resource

Das Winkeltaktsignal wird über Synchronisationsbus-Leitungen zwischen den Karten transportiert. Mit diesem Konfigurationsparameter wird die Busleitung ausgewählt, über die der Winkeltakt im Slave-Betrieb eingelesen wird oder im Master-Betrieb ausgegeben wird.

Wird die Drehzahleinheit auf Master-Betrieb konfiguriert, so steht zusätzlich noch die Option „None“ zur Auswahl – in diesem Fall wird ein Winkeltakt generiert und intern auf der Karte verwendet, jedoch nicht auf einen der beiden Synchronisationsbusse ausgegeben.

#### **Hinweis**

*Bitte beachten Sie, dass es auf einem Synchronisationsbus nur jeweils einen Winkeltakt-Master geben darf!*

#### BNC Output Port Signal

Mit diesem Auswahlfeld wird das Ausgabesignal der BNC-Buchse an der Frontplatte der ES5340.2-ICE definiert. Zur Auswahl stehen die in Abb. 23-3 auf Seite 795 dargestellten Signale.



### Motor Start Angle

Der Winkel, bei dem der Motor nach der Modellinitialisierung losläuft.

Die folgende Tabelle fasst die Eigenschaften der einzelnen Konfigurationsparameter zusammen.

Parameter	Datentyp	Online editierbar	Anmerkungen
Engine Type	int	Nein	0: Viertakt-Verbrennungsmotor 1: Zweitakt-Verbrennungsmotor
RPM Operating Mode	int	Ja	0: Slave 1: Master
SYNC Bus Resource	int	Ja	0: SYNC Bus #0 1: SYNC Bus #1 255: None (nur bei Masterbetrieb)
BNC Output Port Signal	int	Ja	0: Sync 1: Clock 2: Direction 3: Engine Speed
Motor Start Angle	float	Ja	0.0 °KW...360.0 °KW: Zweitaktmotor 0.0 °KW...720.0 °KW: Viertaktmotor

**Tab. 23-1** Die Registerkarte „Globals“ des ES5340-RPM Device

#### Hinweis

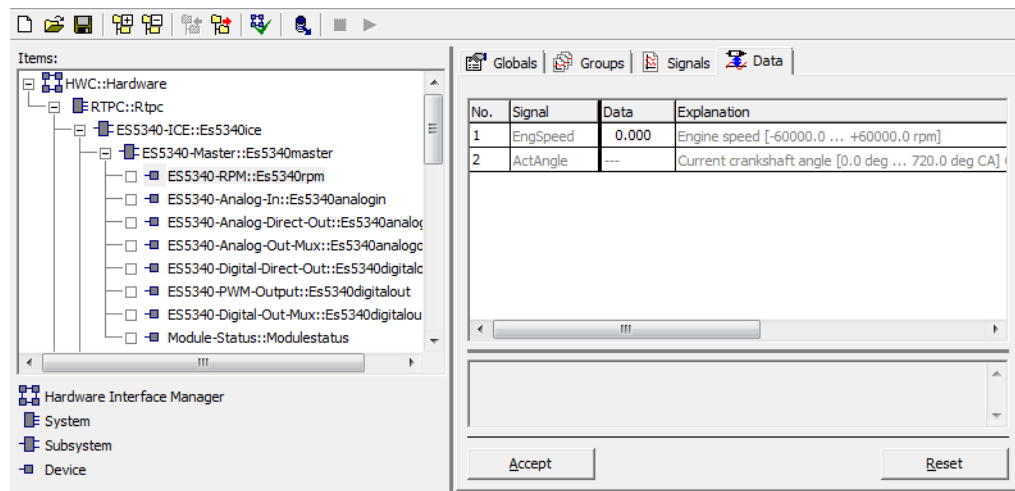
*Beachten Sie bitte, dass je nach hier gewähltem Motortyp („Engine Type“) die Winkelgrenzen unterschiedlich sind (720 ° bzw. 360 °). Dies kann sich auch entsprechend auf den Wertebereich anderer Parameter auswirken!*

### 23.2.3 Groups (ES5340-RPM)

Das ES5340-RPM Device besitzt eine Signalgruppe „Outputs“ zur Steuerung der Drehzahleinheit und eine Signalgruppe „Inputs“ zum Lesen von aktuellen Statusinformationen der Drehzahleinheit.

## 23.2.4 Data (ES5340-RPM)

Die in der Registerkarte „Data“ angebotenen Signale zur Steuerung und Statusüberwachung der Drehzahlseinheit sind vom gewählten Motortyp abhängig. Abb. 23-5 auf Seite 798 zeigt die RTIO-Signale für den Motortyp „Viertaktmotor“ – in Tab. 23-2 auf Seite 798 sind die Signaleigenschaften noch einmal zusammengefasst.



**Abb. 23-5** Die Registerkarte „Data“ des ES5340-RPM Device (Viertaktmotor)

Signal-name	Datentyp	Anmerkungen
EngSpeed	float	Motordrehzahl in Umdrehungen pro Minute Wertebereich: Viertaktmotor: -60000.0 rpm ... +60000.0 rpm, Auflösung: 0.001 rpm Zweitaktmotor: -30000.0 rpm ... +30000.0 rpm, Auflösung: 0.0005 rpm Ein rückwärts drehender Motor kann durch Vorgabe einer negativen Drehzahl nachgebildet werden.
ActAngle	float	Aktueller Kurbelwellenwinkel in °KW Wertebereich: 0.0 °KW ... 720.0 °KW (oder 360 °KW) Auflösung: 0.011 °KW (oder 0.0055 °KW)

**Tab. 23-2** ES5340-RPM Device: RTIO-Signale

### 23.3 ES5340-Analog-In – Analoge Eingänge

Die ES5340.2-ICE besitzt vier analoge Eingänge mit einer Spannungsfestigkeit von  $\pm 60\text{ V}$  – jeder dieser Eingänge wird auf einen Kanal eines AD-Wandlers geführt.

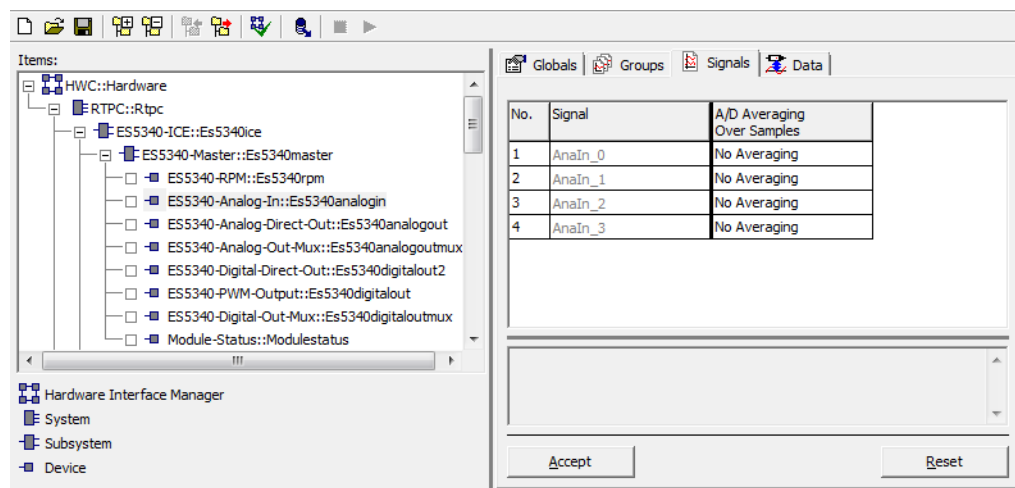
Jeweils zwei AD-Wandler weisen einen Spannungsbereich von 0 bis 5 V auf, bzw. von 0 bis 40 V auf. Jeder Kanal eines AD-Wandlers wandelt mit einer Rate von 500 kSamples pro Sekunde.

#### 23.3.1 ES5340-Analog-In – Analoge Eingänge

Das „ES5340-Analog-In“ Device dient zur Konfiguration der AD-Wandler und zur Erfassung der anliegenden Analogspannungen.

#### 23.3.2 Signals (ES5340-Analog-In)

In der Registerkarte „Signals“ kann für jeden Kanal des Analogwandlers eine Mittelwertbildung über die erfassten AD-Werte konfiguriert werden.



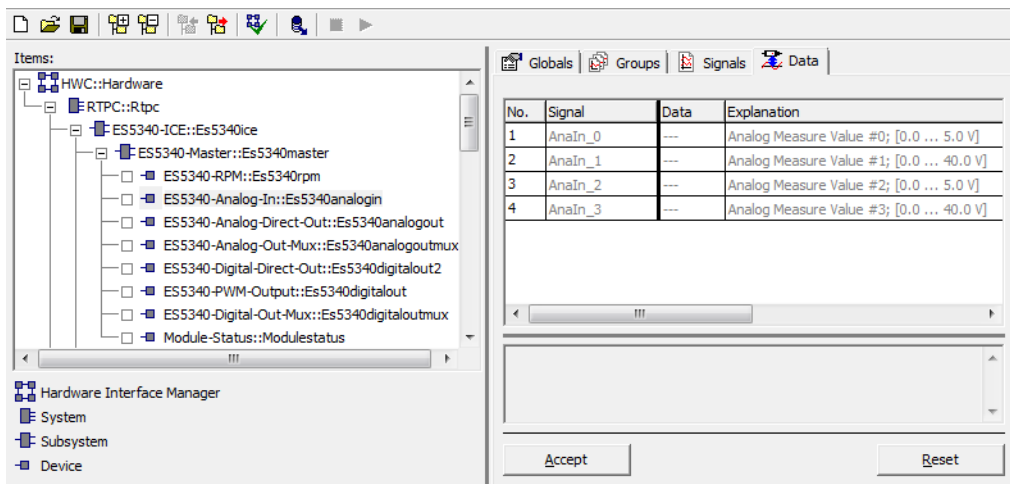
**Abb. 23-6** Die Registerkarte „Signals“ des ES5340-Analog-In Device

Parameter	Daten- typ	Online editier- bar	Anmerkungen
A/D Averaging over Samples	int	Ja	0: Keine Mittelwertbildung 1: Mittelwertbildung über 2 Samples 2: Mittelwertbildung über 4 Samples 3: Mittelwertbildung über 8 Samples 4: Mittelwertbildung über 16 Samples 5: Mittelwertbildung über 32 Samples 6: Mittelwertbildung über 64 Samples 7: Mittelwertbildung über 128 Samples

**Tab. 23-3** ES5340-Analog-In Device: Konfigurationsparameter der Registerkarte „Signals“

23.3.3 Data (ES5340-Analog-In)

In der Registerkarte „Data“ werden die Momentanwerte der Spannungen in Volt angezeigt.



**Abb. 23-7** Die Registerkarte „Data“ des ES5340-Analog-In Device

Signal-name	Daten-typ	Anmerkungen
Analn_0	float	Analoger Eingang 0 (misst im Bereich von 0.0...5.0 V)
Analn_1	float	Analoger Eingang 1 (misst im Bereich 0.0...40.0 V)
Analn_2	float	Analoger Eingang 2 (misst im Bereich von 0.0...5.0 V)
Analn_3	float	Analoger Eingang 3 (misst im Bereich 0.0...40.0 V)

**Tab. 23-4** ES5340-Analog-In Device: Signale der Registerkarte „Data“

## 23.4 ES5340-Digital-Out – Digitale Ausgänge

Das ES5340-Digital-Out RTIO-Element kann in zwei verschiedenen Betriebsarten verwendet werden.

Mode	Beschreibung
Digital Direct Out	Ansteuerung eines digitalen Ausgangs (siehe „ES5340-Digital-Direct-Out“ auf Seite 802)
PWM Output	Ausgabe von PWM-Signalen (siehe „Globals (ES5340-Digital-Direct-Out)“ auf Seite 802)

**Tab. 23-5** Betriebsarten des ES5340-Digital-Out RTIO-Elements

Die Konfiguration des RTIO-Elements ist für jeden Ausgabemodus unterschiedlich – abhängig davon ändert sich auch das Aussehen des RTIO-Elements ES5340-Digital-Out.

Für die Ausgabe von Digitalwerten verwendet die ES5340.2-ICE ein zweistufiges Verfahren:

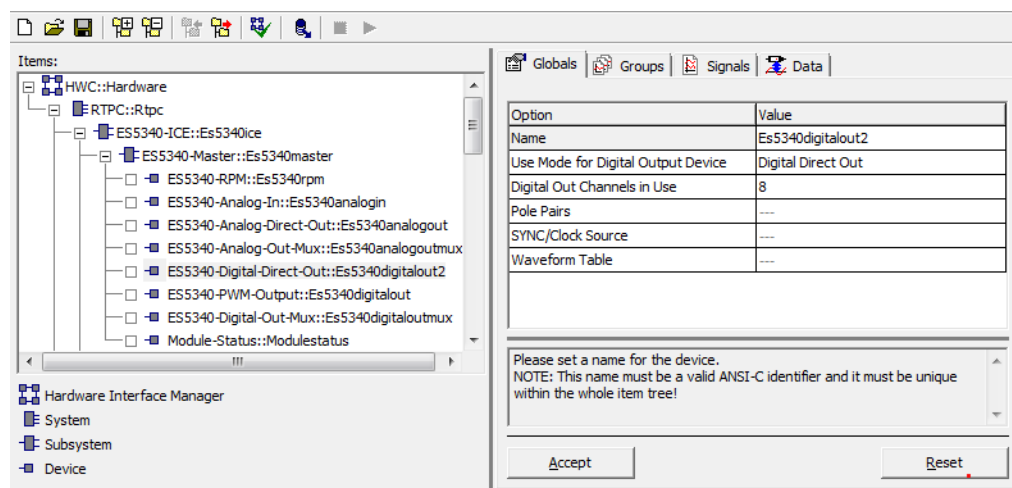
1. Die Ausgabewerte werden über das ES5340-Digital-Out RTIO-Element konfiguriert. Hier können mehr Signale konfiguriert werden als Ausgangskanäle auf der ES5340.2-ICE zur Verfügung stehen (max. je 8 Kanäle im Modus „Digital-Direct-Out“ und 8 Kanäle für die anderen digitalen Ausgabemodi).
2. Die Auswahl der Werte, die tatsächlich ausgegeben werden, erfolgt über das ES5340-Digital-Out-Mux RTIO-Element (siehe „ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer“ auf Seite 806).

### 23.4.1 ES5340-Digital-Direct-Out

Dieser Abschnitt beschreibt die Konfiguration des ES5340-Digital-Out RTIO-Elements mit dem Ausgabemodus „Digital Direct Out“. In dieser Betriebsart können ein bis acht digitale Ausgänge direkt angesteuert werden.

### 23.4.2 Globals (ES5340-Digital-Direct-Out)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 23-8** Die Registerkarte „Globals“ des ES5340-Digital-Direct-Out RTIO-Elements

#### Use Mode for Digital Output Device

Auswahl des Ausgabemodus.

#### Digital Out Channels in Use

Anzahl der konfigurierbaren digitalen Ausgabekanäle (max. 8).

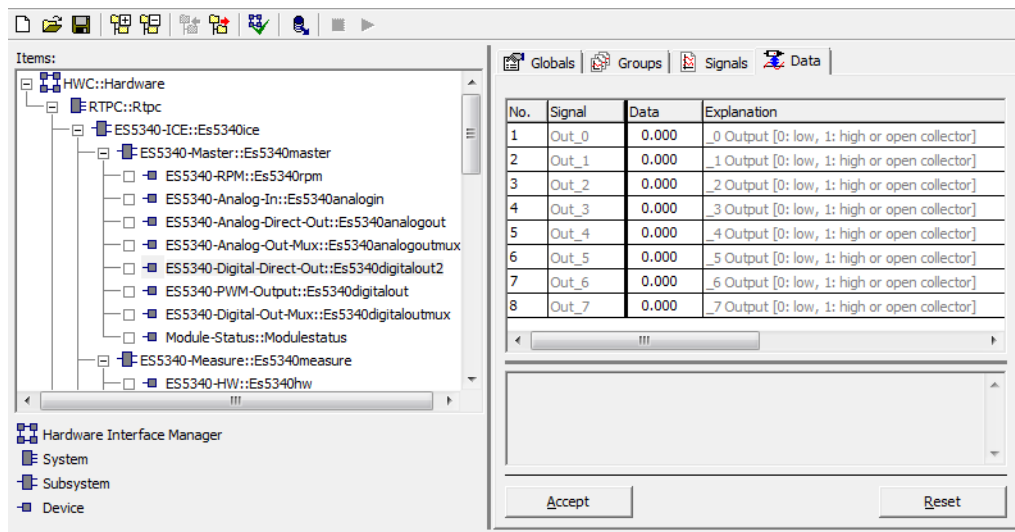
### 23.4.3 Groups (ES5340-Digital-Direct-Out)

Das ES5340-Digital-Direct-Out-RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators.

### 23.4.4 Signals (ES5340-Digital-Direct-Out)

Hier sind keine weiteren Einstellungen notwendig.

23.4.5 Data (ES5340-Digital-Direct-Out)



**Abb. 23-9** Die Registerkarte „Data“ ES5340-Digital-Direct-Out RTIO-Elements

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Out_n	uint8	Digitaler Ausgabewert 0: low; 1: high oder open collector

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-6** ES5340-Digital-Direct-Out: Signale der Registerkarte „Data“

### 23.4.6 ES5340-PWM-Output

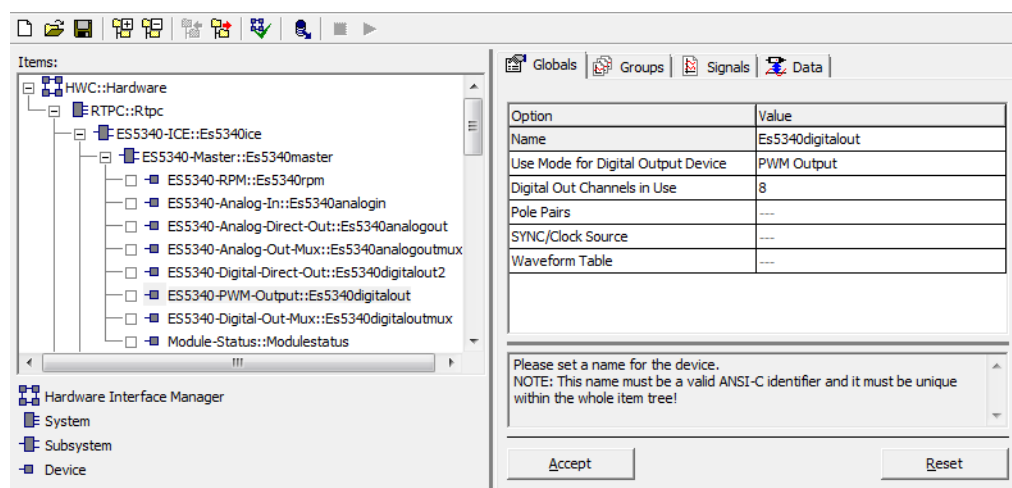
In der Betriebsart „PWM Output“ können bis zu acht digitale Ausgänge für die PWM-Ausgabe konfiguriert werden (nur positive Logik!). Dabei wird für jeden PWM-Kanal Frequenz und Tastverhältnis separat vorgegeben. Die Übernahme der neuen Daten erfolgt jeweils am Ende eines PWM-Zyklus.

#### **Hinweis**

*Beträgt die eingestellte Frequenz weniger als 1 Hz, so erfolgt die Übernahme von neuen Werten sofort!*

### 23.4.7 Globals (ES5340-PWM-Output)

Die folgende Abbildung zeigt die Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 23-10** Die Registerkarte „Globals“ des ES5340-PWM-Output-Elements  
*Use Mode for Digital Output Device*

Auswahl des Ausgabemodus.

*Digital Out Channels in Use*

Anzahl der konfigurierbaren digitalen Ausgabekanäle (max. 8).

### 23.4.8 Groups (ES5340-PWM-Output)

Das ES5340-PWM-Output besitzt eine Signalgruppe „Outputs“ zur Steuerung des jeweiligen Signalgenerators bzw der jeweiligen Signalgeneratoren.

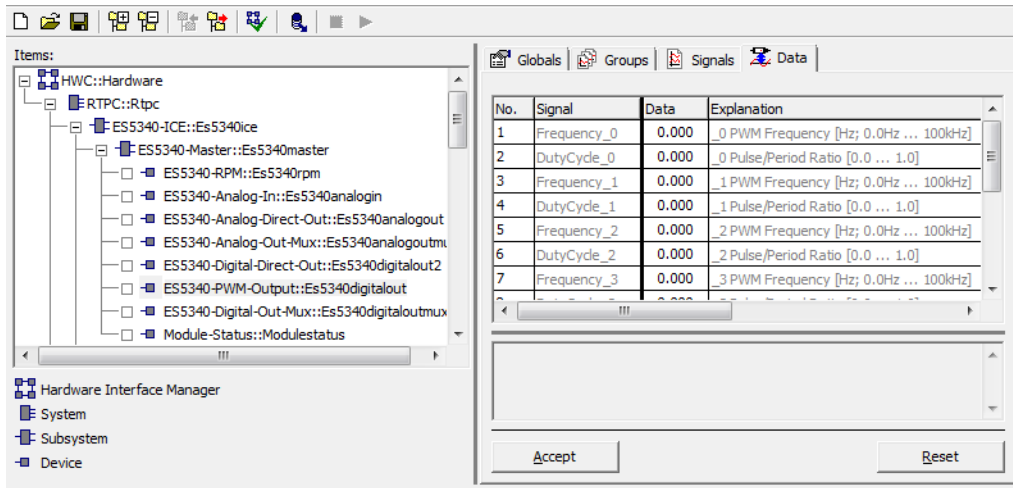
### 23.4.9 Signals (ES5340-PWM-Output)

Hier sind keine weiteren Einstellungen notwendig.



23.4.10 Data (ES5340-PWM-Output)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 23-11** Die Registerkarte „Data“ des ES5340-PWM-Output-RTIO-Elements *Frequency<sub>n</sub>*

Die Frequenz des Signals am PWM-Generator n (n = 0...7).

*DutyCycle<sub>n</sub>*

Das Tastverhältnis des PWM-Signals am Ausgang n (n = 0...7).

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Frequency <sub>n</sub>	uint32	Frequenz des PWM Signals (0 Hz...100 kHz)
DutyCycle <sub>n</sub>	uint32	Tastverhältnis (0.0...1.0)

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-7** ES5340-PWM-Output: Signale der Registerkarte „Data“

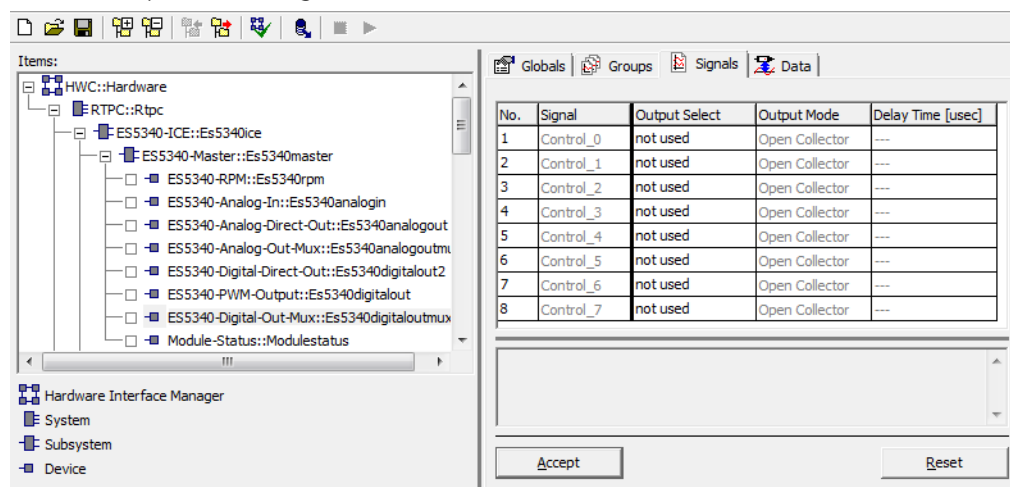
## 23.5 ES5340-Digital-Out-Mux – Digitaler Ausgangsmultiplexer

In dem RTIO-Element „ES5340-Digital-Out-Mux“ werden (zuvor in den RTIO-Elementen „ES5340-Digital-Out“, „ES5340-MSA-Sensor“ und „ES5340-Analog“ des „ES5340-SigGen“ Subsystems konfigurierte) Signale den tatsächlich vorhandenen digitalen Signalausgängen der ES5340.2-ICE zugewiesen.

### 23.5.1 Signals (ES5340-Digital-Out-Mux)

In der Registerkarte „Signals“ werden die digitalen Ausgänge konfiguriert.

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 23-12** Die Registerkarte „Signals“ des ES5340-Digital-Out-Mux RTIO-Elements

#### Output Select

Das Optionsfeld „Output Select“ dient zur Auswahl des internen Signals, das auf den jeweiligen Ausgang gelegt werden soll.

Es wird dabei die Liste aller internen Signalquellen zur Auswahl angeboten:

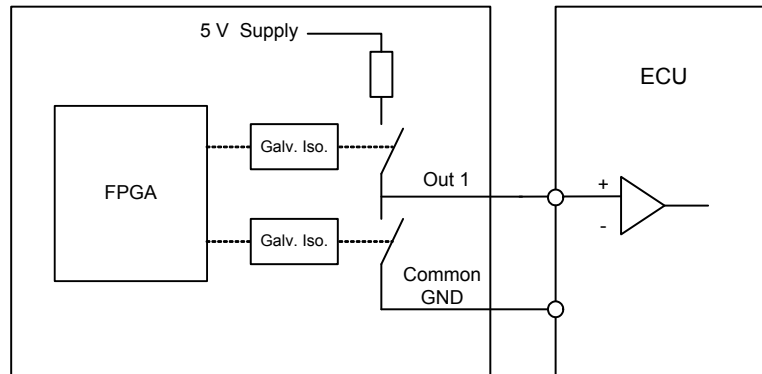
- Ausgabegrößen aller Digital-Out RTIO-Elemente (ES5340-Digital-Direct-Out und ES5340-PWM-Output)
- Digitale Signale der arbiträren Signalgeneratoren (ES5340-Digital) (siehe „ES5340-SigGen Subsystem“ auf Seite 849)
- MSA-Sensorsignal

### Output Mode

Es kann zwischen zwei verschiedenen Ausgabearten unterschieden werden:

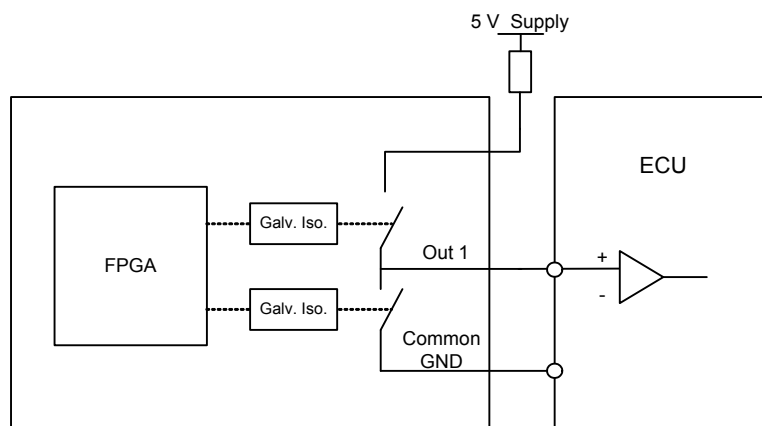
- Pull-Up to +5V

Hier wird ein interner Pull-Up-Widerstand an 5 V geschaltet



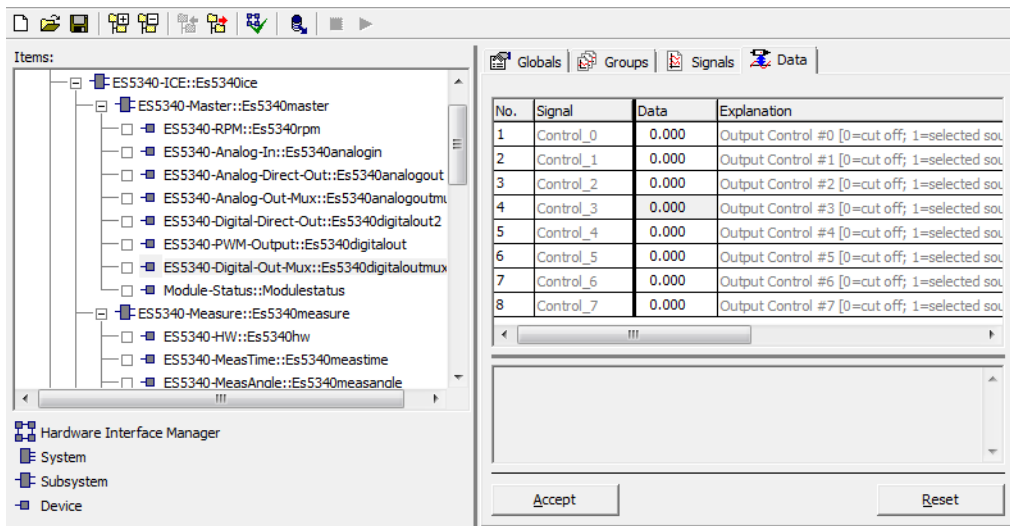
- Open Collector

Hier wird ein externer Pull-Up-Widerstand erwartet.



23.5.2 Data (ES5340-Digital-Out-Mux)

In der vorhandenen Signalgruppe werden acht RTIO-Signale verarbeitet, die das Schließen des Ausgangsrelais eines jeden Kanals steuern. Weiterhin ist es möglich, die Ausgänge für eine Fehlersimulation dauerhaft auf GND oder +5 V zu legen.



**Abb. 23-13** Die Registerkarte „Data“ des ES5340-Digital-Out-Mux RTIO-Elements

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Control_n	uint8	0: Ausgang getrennt 1: Ausgang verbunden 2: Kurzschluss nach GND 3: Kurzschluss nach +5 V

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-8** ES5340-Digital-Out-Mux: Signale der Registerkarte „Data“

## 23.6 ES5340-Analog-Out – Analoge Ausgänge

Das ES5340-Analog-Out RTIO-Element kann in nur Betriebsart „Analog Direct Out“ (Ausgabe einer definierten Spannung) verwendet werden.

Für die Ausgabe von Analogwerten verwendet die ES5340.2-ICE ein zweistufiges Verfahren:

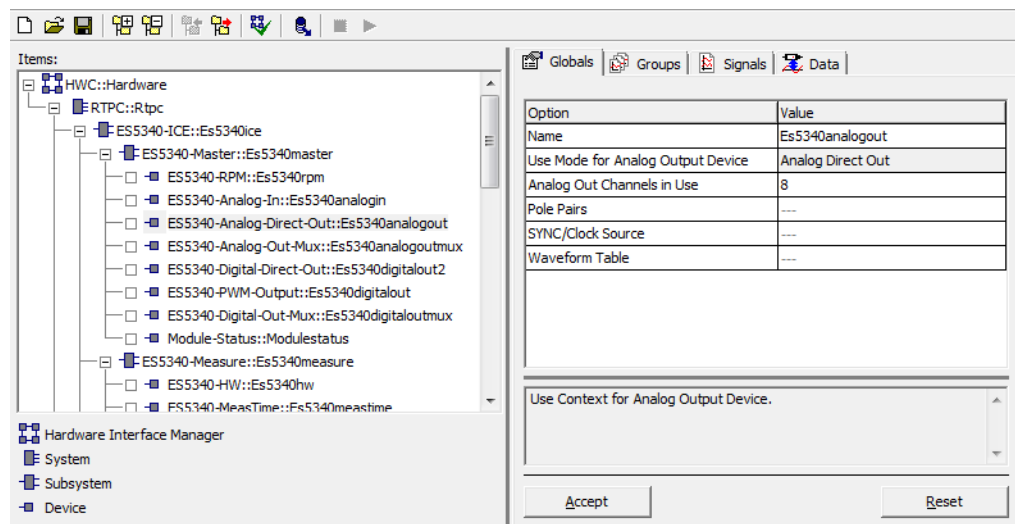
1. Die Ausgabewerte werden über das RTIO-Element ES5340-Analog-Out konfiguriert.
2. Die Auswahl der Werte, die tatsächlich ausgegeben werden, erfolgt über das ES5340-Analog-Out-Mux RTIO-Element (siehe „ES5340-Analog-Out-Mux – Analoges Ausgangsmultiplexer“ auf Seite 811).

### 23.6.1 ES5340-Analog-Direct-Out

In der Betriebsart „Analog Direct Out“ können Sie 1 bis 8 Analogausgänge zur unmittelbaren Ausgabe eines analogen Wertes konfigurieren.

### 23.6.2 Globals (ES5340-Analog-Direct-Out)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 23-14** Die Registerkarte „Globals“ des ES5340-Analog-Direct-Out RTIO-Elements

#### *Use Mode for Analog Output Device*

Auswahl des Ausgabemodus.

#### *Analog Out Channels in Use*

Anzahl der konfigurierbaren analogen Ausgabekanäle (max. 8).

### 23.6.3 Groups (ES5340-Analog-Direct-Out)

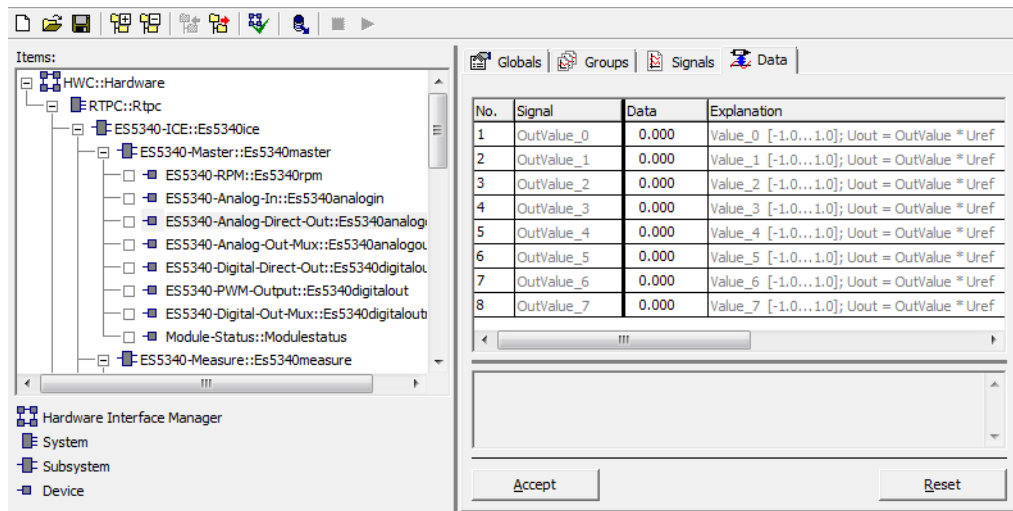
Das ES5340-Analog-Direct-Out RTIO-Element besitzt eine Signalgruppe „Outputs“ zur Steuerung der Ausgabe von Analogsignalen.

### 23.6.4 Signals (ES5340-Analog-Direct-Out)

Hier sind keine weiteren Einstellungen notwendig.

### 23.6.5 Data (ES5340-Analog-Direct-Out)

Die folgende Abbildung zeigt die Registerkarte „Data“.



**Abb. 23-15** Die Registerkarte „Data“ des ES5340-Analog-Direct-Out RTIO-Elements

#### OutValue\_x

Relative Amplitude der Ausgabe – wird mit der im Analogmultiplexer festgelegten Referenzspannung multipliziert.

Signalname	Datentyp	Anmerkungen
<b>(n = 0...7)</b>		

OutValue_n	sint16	Analoger Ausgabewert im Bereich -1.0 ... 1.0; Uout = OutValue * Uref
------------	--------	---

**Tab. 23-9** ES5340-Analog-Direct-Out: Signale der Registerkarte „Data“

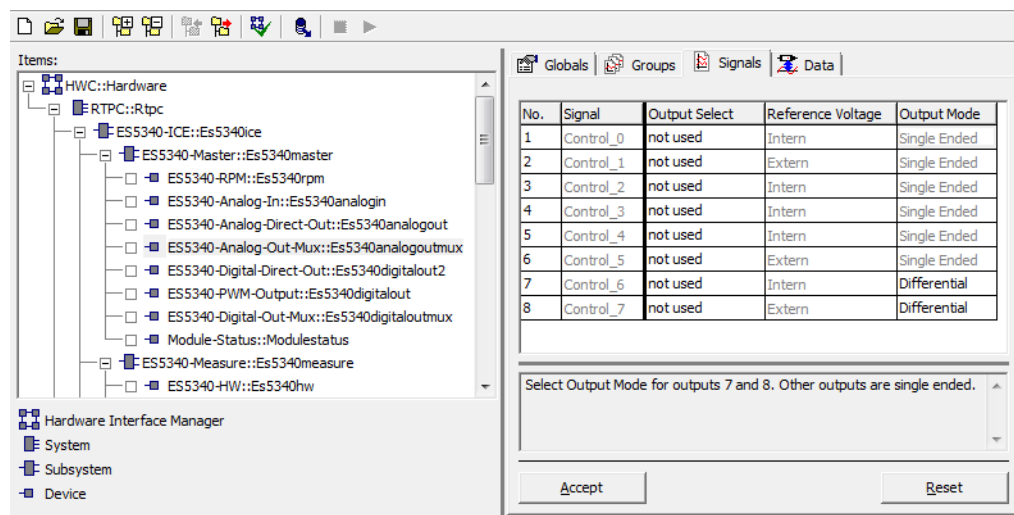
## 23.7 ES5340-Analog-Out-Mux – Analoger Ausgangsmultiplexer

In dem RTIO-Element „ES5340-Analog-Out-Mux“ werden (zuvor im RTIO-Element „ES5340-Analog-Out“ und in den Quellen des „ES5340SigGen“ Subsystems konfigurierte – siehe weiter unten) Signale den tatsächlich vorhandenen analogen Signalausgängen der ES5340.2-ICE zugewiesen.

### 23.7.1 Signals (ES5340-Analog-Out-Mux)

In der Registerkarte „Signals“ werden die acht (ES5340-Master) bzw. sechs (ES5340-Slave) analogen Ausgänge konfiguriert. Dabei können die Eigenschaften des Analogausgangs sowie die Quelle des Ausgabewertes festgelegt werden.

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“ – die Eigenschaften der einzelnen Parameter sind in Tab. 23-10 auf Seite 812 zusammengefasst. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 23-16** Die Registerkarte „Signals“ des ES5340-Analog-Out-Mux RTIO-Elements

#### Output Select

Das Optionsfeld „Output Select“ dient zur Auswahl des internen Signals, das auf den jeweiligen Ausgang gelegt werden soll. Es wird dabei die Liste aller internen Signalquellen zur Auswahl angeboten:

- Signale aus ES5340-Analog-Direct-Out
- Signale aus ES5340-Analog (arbiträre Signalgeneratoren)
- Signale aus ES5340-Knock (Klopfgeneratoren)

#### Reference Voltage

Über dieses Optionsfeld kann die Referenzspannung für den D/A-Wandler des jeweiligen Ausgangskanals gewählt werden. Mögliche Werte sind „Internal“ (für eine interne Referenzspannung von 10 V) und „External“ (für eine extern anliegende Referenzspannung von 12 V).

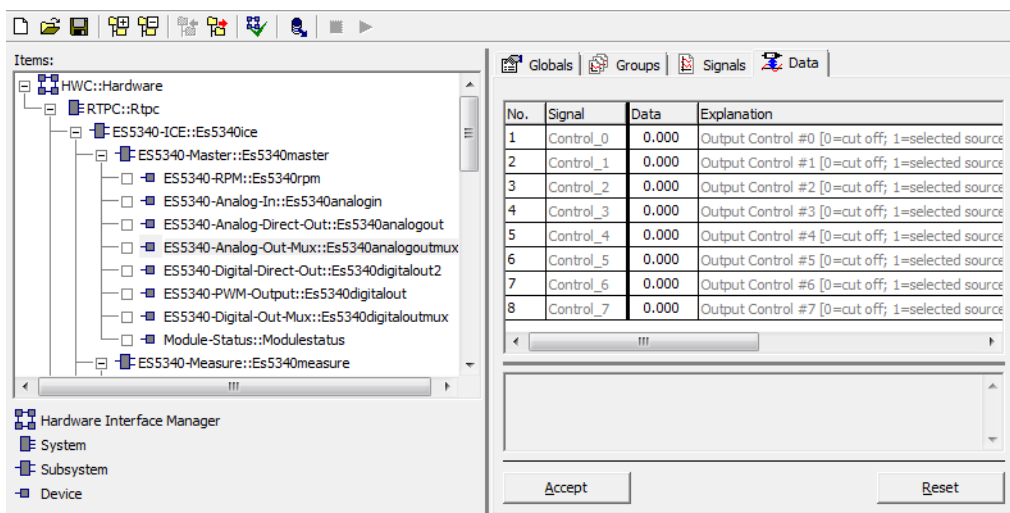
Output Mode

Dieser Parameter ist nur für die Kanäle 6 und 7 veränderbar. Es kann dabei zwischen einem Single-Ended- und einem differentiellen Ausgangssignal umgeschaltet werden.

Der Output Mode für die Kanäle 6 und 7 ist immer identisch.

23.7.2 Data (ES5340-Analog-Out-Mux)

In der vorhandenen Signalgruppe werden acht RTIO-Signale verarbeitet, die das Schließen des Ausgangsrelais eines jeden Kanals steuern. Weiterhin ist es möglich, für eine Fehlersimulation die Ausgänge permanent auf GND oder Uref zu legen.



**Abb. 23-17** Die Registerkarte „Data“ des ES5340-Analog-Out-Mux RTIO-Elements

Signalname (n = 0...7)	Datentyp*	Anmerkungen
Control_n	uint8	0: Ausgang getrennt 1: Ausgang verbunden 2: Kurzschluss nach GND 3: Kurzschluss nach +5 V

\*Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-10** ES5340-Analog-Out-Mux: Signale der Registerkarte „Data“

**Hinweis**  
Ist der Output Mode auf den Wert „Differential“ gesetzt, kann keine Gleichspannung ausgegeben werden.

Wenn gleichzeitig „Reference Voltage“ auf „Internal“ eingestellt ist, wird im Fall 3 (Kurzschluss nach Uref) am Ausgang 0 V ausgegeben. Ist hingegen „Reference Voltage“ auf „External“ eingestellt, wird im Fall 3 (Kurzschluss nach Uref) die am externen Referenzeingang („Excitation P/N“) angelegte Wechselspannung mit voller Amplitude ausgegeben.



### 23.8 ES5340-Slave

---

Das ES5340-Slave Subsystem dient zur Konfiguration einer ES5340-Slavekarte.

### 23.9 ES5340-Digital-Out (Slave)

---

Die Ausgabe von digitalen Signalen ist bei der Slave-Karte nur in der Betriebsart „Digital Direct Out“ mit maximal sechs Kanälen möglich.

Eine Beschreibung des ES5340-Digital-Direct-Out RTIO-Elements finden Sie im Abschnitt „ES5340-Digital-Direct-Out“ auf Seite 802.

### 23.10 ES5340-Digital-Out-Mux (Slave)

---

In dem RTIO-Element „ES5340-Digital-Out-Mux“ werden die Signale gewählt, die auf den 6 digitalen Signalausgängen der Slave-Karte ausgegeben werden.

Ausgewählt werden können folgende Signale:

- Die Signale des Typs „Digital Direct Out“ der Slave-Karte
- Die digitalen Ausgangsgrößen eines Modells der Master-Karte

Nähere Informationen zur Konfiguration der Signalausgänge finden Sie im Abschnitt „Signals (ES5340-Digital-Out-Mux)“ auf Seite 806.

### 23.11 ES5340-Analog-Out (Slave)

---

Die Ausgabe von analogen Signalen ist bei der Slave-Karte nur in der Betriebsart „Analog Direct Out“ mit maximal sechs Kanälen möglich.

Eine Beschreibung des ES5340-Analog-Direct-Out RTIO-Elements finden Sie im Abschnitt „ES5340-Analog-Direct-Out“ auf Seite 809.

### 23.12 ES5340-Analog-Out-Mux (Slave)

---

In dem RTIO-Element „ES5340-Analog-Out-Mux“ werden die Signale gewählt, die auf den 6 analogen Signalausgängen der Slave-Karte ausgegeben werden.

Ausgewählt werden können Signale des Typs „Analog Direct Out“ der Slave-Karte.

Nähere Informationen zur Konfiguration der Signalausgänge finden Sie im Abschnitt „Signals (ES5340-Analog-Out-Mux)“ auf Seite 811.

## 23.13 ES5340-Measure Subsystem

### Begriffsdefinitionen

#### Aktive Signalflanke

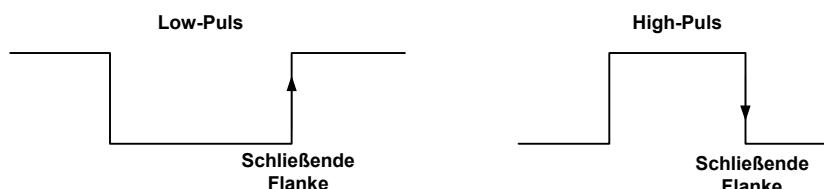
Die aktive Signalflanke ist die steigende Flanke bei Messungen, die an steigenden Flanken durchgeführt werden, bzw. die fallende Flanke bei Messungen, die an fallenden Flanken durchgeführt werden.

#### Aktives Signal

Ein Signal wird als aktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen low oder während High-Zeit-Messungen high ist.

#### Schließende Pulsflanke

Die steigende Flanke ist die schließende Flanke eines Low-Pulses. Die fallende Flanke ist die schließende Flanke eines High-Pulses.



#### Inaktives Signal

Ein Signal wird als inaktiv bezeichnet, wenn sein Pegel während Low-Zeit-Messungen high oder während High-Zeit-Messungen low ist.

#### LWL

Steht für Lower Window Limit und bezeichnet die untere Grenze des Winkel Fensters bei winkelsynchronen Messungen.

#### Öffnende Pulsflanke

Die fallende Flanke ist die öffnende Flanke eines Low-Pulses. Die steigende Flanke ist die öffnende Flanke eines High-Pulses.



#### UWL

Steht für Upper Window Limit und bezeichnet die obere Grenze des Winkel Fensters bei winkelsynchronen Messungen.

### 23.13.1 Grundprinzip der Messwertbestimmung

#### Komparatorstufen

Für die weitere Messwerterfassung ist es erforderlich, aus den gemessenen Eingangsspannungen eine binäre Pegelinformation zu extrahieren. Hierzu wird jede Eingangsspannung auf eine Komparatorstufe mit Hystereseverhalten geführt.

Die Spannungsschwellen der Komparatoren können vom Anwender auf verschiedene Arten konfiguriert werden:

- Absolute Level
- Untere Schwelle =  $1/3 \text{ URefX}$  (Modellgrößen) ( $X = A\dots E$ )  
Obere Schwelle =  $2/3 \text{ URefX}$  (Modellgrößen) ( $X = A\dots E$ )
- Untere Schwelle =  $1/3 \text{ Analog-In } n$  ( $n = 0\dots3$ )  
Obere Schwelle =  $2/3 \text{ Analog-In } n$  ( $n = 0\dots3$ )

#### *Pulsweiten- und Winkelmessungen*

---

Die Pegelinformation eines jeden Kanals wird ständig auf Änderungen hin überprüft, d.h. es wird eine Erkennung von Signalfanken durchgeführt. Bei Auftreten einer Flanke werden die aktuelle Zeit und der aktuelle Kurbelwellenwinkel in einem FPGA-internen Speicher abgelegt. Die Prozessor-Firmware der ES5340.2-ICE kann nun die erfassten Werte aus dem Speicher auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben.

Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind, wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten softwaretechnischen Aufbereitung der Messwerte hat den Vorteil, dass zur Implementierung weiterer Messverfahren (neben den bereits implementierten) eine Anpassung der Firmware und des RTIO-Treibers genügt.

#### *Zeitbasierte (asynchrone) Messungen*

---

Bei asynchronen Messungen wird der entsprechende Messwert (z.B. Frequenz, Tastverhältnis oder Hightime) auf Basis der aktuellsten im Speicher verfügbaren Flanken-Einträge berechnet.

Bei diesen Messungen kann eine zeitsynchrone Timeout-Überwachung aktiviert werden. Wenn innerhalb einer vom Anwender definierbaren Zeitspanne keine Flanke auf dem zugehörigen Hardwarekanal detektiert wird, so wird ein Timeout erkannt.

#### *Winkelbasierte (synchrone) Messungen*

---

Charakteristisch für winkelsynchrone Messungen sind Winkelfenster, die durch eine untere Winkelfenstergrenze (LWL) in °KW und eine obere Winkelfenstergrenze (UWL) in °KW spezifiziert werden.

Pro Hardwarekanal kann der Anwender bis zu drei Winkelfenster definieren, die sich überlappen dürfen, deren Größe jedoch jeweils 720 °KW nicht überschreiten darf. Der Anwender definiert Messungen der Art

- Hightime des n-ten Pulses im Winkelfensters
- Winkel der n-ten fallenden Flanke im Winkelfenster
- Additive Low-Zeit im Winkelfenster.

Die Berechnung der Messwerte erfolgt jeweils dann, wenn der Kurbelwellenwinkel die obere Winkelfenstergrenze überschritten hat. Auch für winkelsynchrone Messungen kann eine Timeoutüberwachung aktiviert werden. Wenn an der oberen Winkelfenstergrenze festgestellt wird, dass der Messwert nicht berechnet

werden kann, z.B. wenn die Hightime des vierten Pulses im Winkelfenster zu vermessen ist, jedoch nur drei Pulse aufgetreten sind, so wird ein Timeout erkannt.

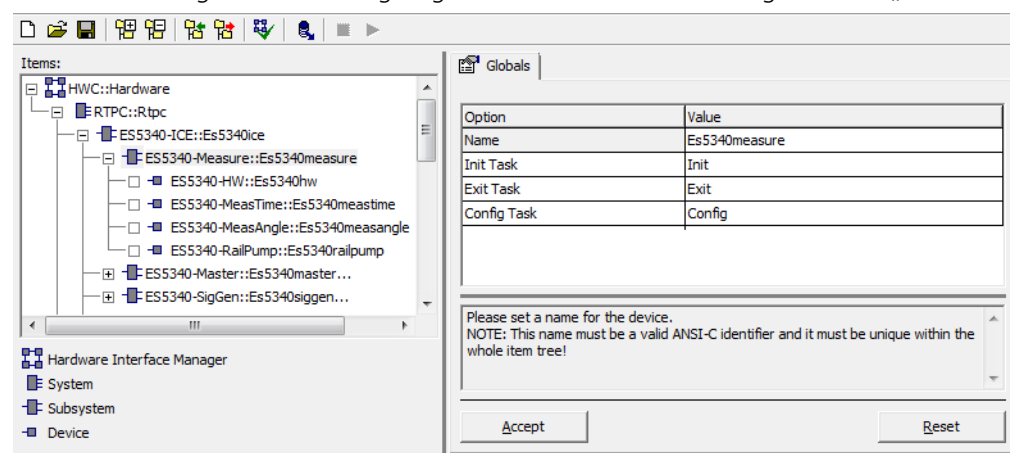
### Timeouts

Wird ein Timeout festgestellt, so wird dies (bei aktivierter Timeout-Überwachung) durch Setzen eines Timeout-Flags angezeigt. Gleichzeitig wird im Timeout-Fall auch der dann gültige Messwert durch die Timeout-Überwachung definiert.

Hierbei gibt es zwei Möglichkeiten: Entweder wird ein vom Anwender vorgegebener Messwert an das Simulationsmodell weitergeben oder ein von der jeweiligen Messmethode definierter Wert. Die Entscheidung welcher der beiden Werte weitergegeben wird, trifft der Anwender, wobei jedoch das zweite Verfahren (ein von der Messmethode definierter Wert) nur bei einigen Messverfahren zur Verfügung steht.

### 23.13.2 Globals (ES5340-Measure Subsystem)

Die folgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



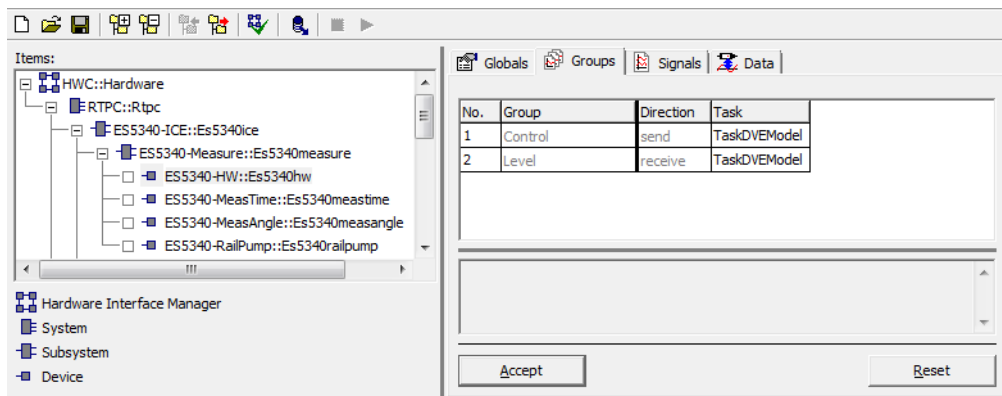
**Abb. 23-18** Die Registerkarte „Globals“ des ES5340-Measure Subsystems

## 23.14 ES5340-HW Device

Das ES5340-HW Device dient zur Konfiguration und Ansteuerung der ES5340.2-ICE.

### 23.14.1 Groups (ES5340-HW Device)

Das ES5340-HW Device besitzt zwei Signalgruppen.



**Abb. 23-19** Die Registerkarte „Groups“ des ES5340-HW Device

Die Signalgruppe „Control“ wird vom Experimentaltarget zur ES5340.2-ICE übertragen. Mit dieser Signalgruppe wird die Messwertberechnung auf den einzelnen Hardwarekanälen des Boards ein- und ausgeschaltet.

Die Signalgruppe „Level“ wird in umgekehrter Richtung von der ES5340.2-ICE zum Experimentaltarget übertragen. Diese Signalgruppe enthält die aktuellen Pegelinformationen aller Hardwarekanäle.

Den Signalgruppen sind Betriebssystem-Tasks zuzuordnen. Für die Signalgruppe „Control“ wird man üblicherweise eine Task mit periodischer Aktivierung und verhältnismäßig großer Aktivierungsperiode (z.B. 100 ms) wählen, da das Ein- und Ausschalten der Hardwarekanäle in der Regel kein hochdynamischer Vorgang ist. Wenn das Ein- und Ausschalten der Hardwarekanäle nur beim Modellstart bzw. Modellstop erfolgen soll, so ist es ausreichend, die Signalgruppe „Control“ der Init Task und der Exit Task des Modells zuzuordnen.

Falls man im Simulationsmodell auch Pegelinformationen auswertet, wird man der Signalgruppe „Level“ eine Task mit periodischer Aktivierung zuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik bzw. der Periodendauer der zu erfassenden Signale.

*RTIO-Signale der Signalgruppe „Control“*

Die Signalgruppe „Control“ umfasst 20 RTIO-Signale, mit denen die Messwertberechnung auf den 20 Hardwarekanälen aktiviert bzw. deaktiviert werden kann.

Tab. 23-11 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
EnableChn_00 ... EnableChn_19	bool	Ein-/Ausschalten der Messwertberechnung auf den Hardwarekanälen 0 bis 19 0: Ausschalten 1: Einschalten
UrefA...UrefE	float	Definition der fünf Spannungen für die Komparatorstufe

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-11** ES5340-HW Device: Die RTIO-Signale der Signalgruppe „Control“

*RTIO-Signale der Signalgruppe „Level“*

Die Signalgruppe „Level“ umfasst ein RTIO-Signal „LvlBitField\_0“, das als Bitfeld zu interpretieren ist. Das Bitfeld umfasst 20 Bit, in ihm sind die Pegelinformationen der Hardwarekanäle 0 bis 19 codiert.

Tab. 23-12 fasst die Eigenschaften des RTIO-Signals zusammen.

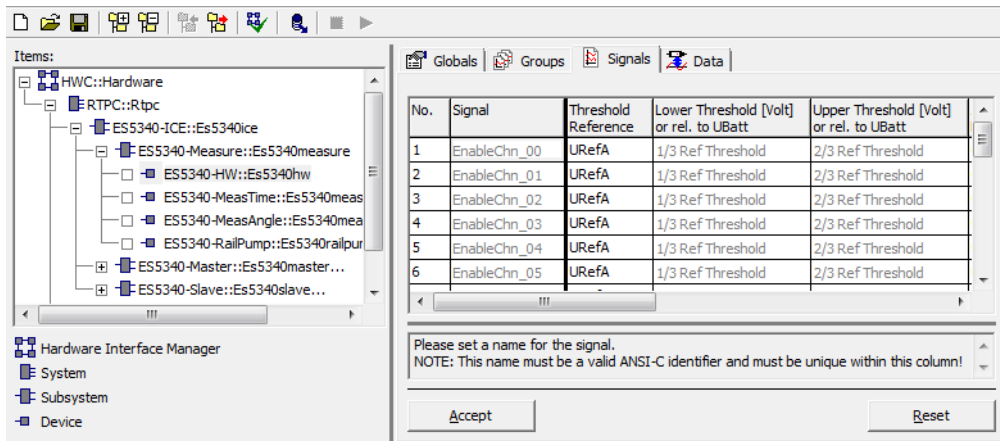
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
LvlBitField_0	uint32	Bitfeld mit Pegelinformationen der Hardwarekanäle 0 bis 19: Kanal 0: Bit 0, LSB (Least Significant Bit) Kanal 19: Bit 19 Bitwert 0: Low Pegel Bitwert 1: High Pegel

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-12** ES5340-HW Device: Die RTIO-Signale der Signalgruppe „Level“

### 23.14.2 Signals (ES5340-HW Device)

In der Registerkarte „Signals“ wird die Konfiguration der 20 Hardwarekanäle einer ES5340.2-ICE durchgeführt. Abb. 23-20 zeigt die RTIO-Parameter der Registerkarte „Signals“.



**Abb. 23-20** Die Registerkarte „Signals“ des ES5340-HW Device

Die Eigenschaften der einzelnen Parameter werden im Folgenden beschrieben. Alle Parameter sind während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

#### Threshold Reference

Mit dem Optionsfeld „Threshold Reference“ werden die Referenzspannungen der Komparatoren definiert. Die Schwellspannungen der Komparatoren können auf drei verschiedene Arten konfiguriert werden:

- Absolute Levels  
Die Komparatorschwellen werden explizit in Volt vorgegeben. Die untere Komparatorschwelle wird im numerischen Eingabefeld „Lower Threshold [Volt] or rel. to UBatt“ definiert, die obere Komparatorschwelle wird im numerischen Eingabefeld „Upper Threshold [Volt] or rel. to UBatt“ definiert.
- URefX (X = A...E)  
In diesem Fall entspricht die untere Schwelle 1/3 der Batteriespannung URefX und die obere Schwelle 2/3 der Batteriespannung URefX.
- Analog-In n (n = 0..3)  
In diesem Fall entspricht die untere Schwelle 1/3 der Spannung an Analog\_In n und die obere Schwelle 2/3 der Spannung an Analog\_In n.

#### Lower Threshold [Volt] or rel. to UBatt

#### Upper Threshold [Volt] or rel. to UBatt

Diese beiden numerischen Eingabefelder sind nur editierbar, wenn im Feld „Threshold Reference“ die Option „Absolute Levels“ ausgewählt ist. Die Eingangskomparatoren eines Hardwarekanals besitzen ein Hystereseverhalten. Die Eingabefelder definieren die untere und obere Schaltschwelle der Komparatorhysterese in Volt.

*Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)*

---

*Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)*

---

Auf jedem Hardwarekanal können zur Durchführung von winkelsynchronen Messungen drei Winkelfenster definiert werden. „Angle Window #x Lower Limit [°CA]“ definiert jeweils die untere Winkelfenstergrenze in °KW, „Angle Window #x Upper Limit [°CA]“ definiert jeweils die obere Winkelfenstergrenze in °KW.

Die Grenzen können im Bereich [-720.0 °KW, 720 °KW[ (Viertaktmotor) bzw. im Bereich [-360.0 °KW, 360 °KW[ (Zweitaktmotor) eingestellt werden mit den Einschränkungen, dass ein Winkelfenster nicht größer als 720 °KW (bzw. 360 °KW) sein darf und die untere Winkelfenstergrenze kleiner als die obere Winkelfenstergrenze sein muss.



## 23.15 ES5340-MeasTime Device

---

### 23.15.1 Nicht synchronisierte Zeit-/Frequenzmessungen

---

Mit diesem RTIO-Device können Sie Zeit-, Frequenz- und Tastverhältnismessungen an periodischen Signalen durchführen oder den Binärwert eines Eingangs erfassen. In einem RTIO-Device können maximal zehn Messkanäle konfiguriert werden und es können maximal vier dieser RTIO-Devices in ein ES5340-Digital-In Subsystem eingefügt werden. Insgesamt stehen maximal 40 Softwarekanäle zur Verfügung.

Die einzelnen Messungen sind voneinander unabhängig und auch prinzipiell unabhängig von anderweitiger Verwendung der Eingänge (synchrone Signalvermessung).

#### *Prinzip der Messwerterfassung und Berechnung*

---

Die Messwerterfassung erfolgt auf der ES5340.2-ICE – hierbei werden die 20 Eingangssignale entsprechend den (im ES5340-HW Device) vorgegebenen Schaltschwellen digitalisiert.

Für jeden sich ergebenden Flankenwechsel eines Hardwarekanals wird ein Datenwort generiert, welches aus einem 31-Bit Zeitstempel eines lokalen Zählers und einem Bit für die Flankeninformation besteht. Der Zeitstempel hat eine Auflösung von 8 ns (125 MHz-Taktung), so dass eine Zeit von max. 17,28 s dargestellt werden kann. Eine „1“ als Flankenbit signalisiert eine positive Flanke, d.h. der Pegel ist danach „1“.

Die Berechnung der Messwerte erfolgt im Treiber des Real-Time PC. Hierzu werden für jeden Hardwarekanal die Daten der jeweils letzten vier Flankenwechsel mittels des zyklischen DMA-Transfers in den Arbeitsspeicher des Real-Time PC übertragen. Bei hinreichend niedriger DMA-Transferzykluszeit (Defaultwert: 100 µs) stehen im Treiber alle Informationen zur Berechnung der Messverfahren mit niedriger Latenz zur Verfügung.

Falls bei einem Hardwarekanal seit Beginn der Datenerfassung noch keine vier Flankenwechsel aufgetreten sind, so wird beim Start der Datenerfassung für alle vier Flanken das gleiche Datenwort generiert. Dieses besteht aus dem aktuellen Zeitstempel und einer Flankeninformation, welche den aktuellen Pegel widerspiegelt.

Diese einfachen Kodierungsregeln ermöglichen es, gültige von ungültigen Daten zu unterscheiden:

- Der aktuelle Pegel kann auch ohne Flankenwechsel ermittelt werden.
- Ungültige Flankendaten können daran erkannt werden, dass das Flankenbit sich gegenüber dem vorigen Datum nicht geändert hat.

Die Messverfahren zur Rückgabe des Pegels liefern daher immer gültige Daten. Bei allen anderen Messverfahren müssen ausreichend Flanken vorhanden sein, was vom Verfahren und dem aktuellen Zustand abhängt.

Kann kein Messwert berechnet werden, so wird als Wert -8888.0 zurückgegeben (sofern keine Zeitüberschreitung vorliegt bzw. überwacht wird).

## 23.15.2 Die Messverfahren

Im Folgenden finden Sie eine Beschreibung der möglichen Verfahren zur Vermessung digitaler Signale.

**Hinweis**

*Generell dürfen die gemessenen Zeiten zwischen zwei Flankenereignissen den Abbildungsbereich des 31-Bit-Zählers (17,28 s) nicht überschreiten. Ist dies der Fall, kommt es zu einem Zählerüberlauf und damit zu falschen Messergebnissen.*

*Frequenz- und Periodendauermessung*

Die Messverfahren

- Cycle Time --/-- [ $\mu$ s]
- Cycle Time --\-- [ $\mu$ s]
- Frequency --/-- [Hz]
- Frequency --\-- [Hz]

liefern die Periodendauer bzw. Frequenz des Signals am jeweiligen Eingang, gemessen zwischen zwei steigenden Flanken (--/--) bzw. zwischen zwei fallenden Flanken (--\--).

*High/Low Zeitmessung*

Die Messverfahren

- High Time [ $\mu$ s]
- Low Time [ $\mu$ s]

liefern die Zeit zwischen letzter steigender und darauffolgender fallender Flanke (High Time) bzw. letzter fallender und steigender Flanke (Low Time).

*Tastverhältnismessungen*

Die Messverfahren

- Duty Cycle L/(L+H) --/--
- Duty Cycle L/(L+H) --\--
- Duty Cycle H/(L+H) --/--
- Duty Cycle H/(L+H) --\--

liefern das Verhältnis der Low- bzw. High-Zeit, bezogen auf die Periodendauer jeweils von steigender zu steigender Flanke (--/--) bzw. von fallender zu fallender Flanke (--\--).

*Pegelmessung*

Die Messverfahren

- Level (Active High)
- Level (Active Low)

liefern den Pegel eines PWM-Eingangs in Form einer aktiv-/inaktiv-Information. „0“ bedeutet das Signal ist inaktiv, „1“ bedeutet das Signal ist aktiv.

### Verhalten bei Zeitüberschreitung

Für jeden Messkanal kann eine Überwachung von Zeitüberschreitungen eingestellt werden. Bei aktivierter Überwachung können der Default-Rückgabewert und das Überwachungsintervall verändert werden. Der Zustand „Zeitüberschreitung“ ist erreicht, wenn die Differenz zwischen der aktuellen Zeit einer DMA-Übertragung und der Zeit des letzten Flankenwechsels des Kanals die vorgegebene Intervallzeit überschreitet. Maximal kann eine Intervallzeit von 10 s angegeben werden. Bis zu diesem Zeitpunkt wird der letzte Messwert übertragen (bzw. -8888.0, falls noch keine gültiger Wert vorhanden ist).

Bei Erreichen der Zeitüberschreitung ist der Messwert von der Betriebsart abhängig. Prinzipiell sind drei Betriebsarten zur Überwachung bei Zeitüberschreitung möglich:

- Inactive  
Keine Überwachung – es wird der letzte gemessene Wert (bzw. -8888.0, falls noch kein Wert vorliegt) zurückgegeben. Das Signal „Tout\_<xx>“ hat immer den Wert „2 = not used“.
- Intvl Predef  
Im Falle einer Zeitüberschreitung das Signal „Tout\_<xx>“ auf „1 = Timeout“ gesetzt und der vordefinierte Wert zurückgegeben.
- Intvl InpDep  
Das Signal „Tout\_<xx>“ wird auf „1 = Timeout“ gesetzt und ein vom Messverfahren und ggf. vom Eingangspegel abhängiger Wert zurückgegeben (siehe Tab. 23-13).

Messverfahren	Wert (Input Level Dependent)
No Measurement	-8888.0
High Time [ $\mu$ s]	0
Low Time [ $\mu$ s]	0
Cycle Time --/-- [ $\mu$ s]	Zeit seit letzter Flanke [ $\mu$ s]
Cycle Time --\-- [ $\mu$ s]	Zeit seit letzter Flanke [ $\mu$ s]
Frequency --/-- [Hz]	Kehrwert der Zeit seit letzter Flanke [Hz]
Frequency --\-- [Hz]	Kehrwert der Zeit seit letzter Flanke [Hz]
Duty Cycle L/(L+H) --/--	1.0 bei Low-Pegel, 0.0 bei High-Pegel am Eingang
Duty Cycle L/(L+H) --\--	1.0 bei Low-Pegel, 0.0 bei High-Pegel am Eingang
Duty Cycle H/(L+H) --/--	0.0 bei Low-Pegel, 1.0 bei High-Pegel am Eingang
Duty Cycle H/(L+H) --\--	0.0 bei Low-Pegel, 1.0 bei High-Pegel am Eingang
Level (Active High)	0
Level (Active Low)	0

**Tab. 23-13** Rückgabewerte bei Zeitüberschreitung (Betriebsart „Intvl InpDep“)

Bei den Werten, die bei Messungen von Frequenz- und Periodendauer übermittelt werden, ist der Überlauf des 31-Bit breiten Zeitzählers berücksichtigt, d.h. diese können im Rahmen der 64-Bit-Gleitpunktdarstellung beliebig kleine bzw. große Werte (Frequenz bzw. Zeit) annehmen.

Der Zustand der Zeitüberschreitung wird verlassen, wenn das 32-Bit-Datenwort der neuesten Flanke (Zeitstempel + Flankeninformation) sich gegenüber jenem ändert, das bei Eintritt der Zeitüberschreitung übermittelt wurde. Dadurch kön-

nen auch Zeitüberschreitungen dargestellt werden, welche den Überlaufwert des 31-Bit Zeitzählers (17,28 s) überschreiten. Das Signal „Tout\_<xx>“ wird auf den Wert „0 = no Timeout“ gesetzt.

### 23.15.3 Globals (ES5340-MeasTime Device)

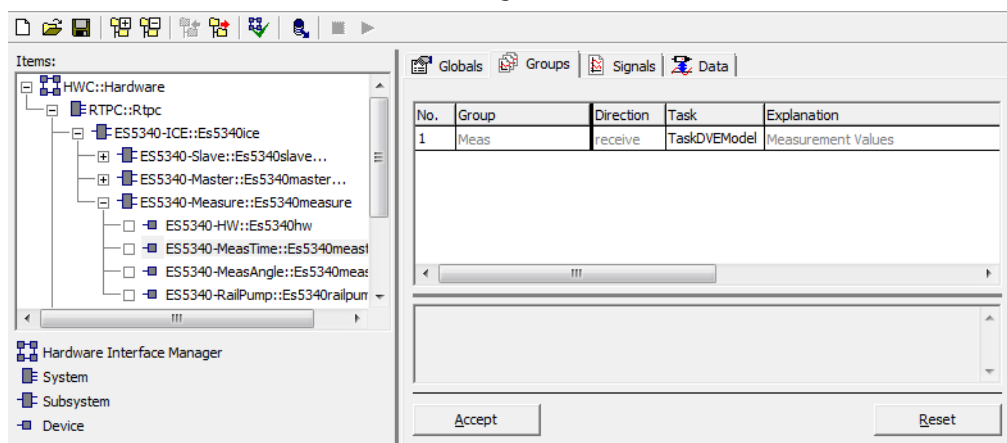
Das ES5340-MeasTime Device dient zur Spezifikation und Konfiguration von zeitbasierten (asynchronen) Messungen. Jedes ES5340-MeasTime Device bietet 10 Messungen, die frei auf die 20 Hardwarekanäle der ES5340.2-ICE verschaltet werden können. Bis zu vier ES5340-MeasTime Devices werden pro ES5340.2-ICE unterstützt, so dass insgesamt 40 Messungen konfiguriert werden können.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.

### 23.15.4 Groups (ES5340-MeasTime Device)

Das ES5340-MeasTime Device besitzt eine Signalgruppe (Abb. 23-21), die von der ES5340.2-ICE zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen.

Dieser Signalgruppe ist eine Betriebssystem-Task mit periodischer Aktivierung zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.



**Abb. 23-21** Die Registerkarte „Groups“ des ES5340-MeasTime Device  
Die RTIO-Signale der Signalgruppe „Meas“

Die Signalgruppe „Meas“ umfasst 20 RTIO-Signale.

Die RTIO-Signale „MeasVal\_00“ bis „MeasVal\_09“ enthalten die Messwerte der 10 Messungen. Es handelt sich dabei um regulär ermittelte Messwerte oder (im Falle eines Timeouts) um den für diesen Fall vorgesehenen Timeout-Wert.

Wenn eine Messung nicht verwendet wird, wird dem zugehörigen Messwert 8888.0 zugewiesen. Die physikalische Einheit der Messwerte hängt vom Messverfahren ab:

- Zeitmessungen (Zeitstempel, (additive) Pulsweiten-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Frequenzmessungen erfolgen in Hertz
- Spannungsmessungen erfolgen in Volt
- alle anderen Messungen (Tastverhältnisse, Pulszählung) sind dimensionslos

Die RTIO-Signale „Tout\_00“ bis „Tout\_09“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messwert.

Tab. 23-14 fasst die Eigenschaften der RTIO-Signale zusammen.

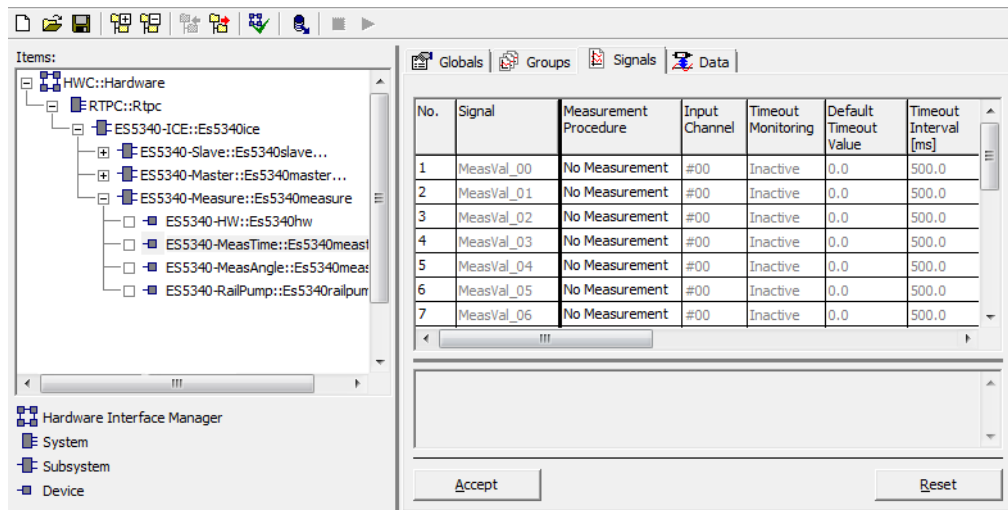
RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
MeasVal_00 ... MeasVal_09	real64	Messwert. Wenn die Messung nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in µs - Frequenzmessungen erfolgen in Hz - Winkelmessungen erfolgen in °KW - Spannungsmessungen erfolgen in V
Tout_00 ... Tout_09	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout festgestellt 1: Timeout festgestellt 2: Timeout-Überwachung ist inaktiv

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-14** ES5340-MeasTime Device: Die RTIO-Signale der Signalgruppe „Meas

23.15.5 Signals (ES5340-MeasTime Device)

In der Registerkarte „Signals“ wird die Konfiguration der 32 Messungen eines ES5340-MeasTime Device durchgeführt. Abb. 23-22 zeigt die RTIO-Parameter der Registerkarte „Signals“ – Tab. 23-15 auf Seite 827 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 23-22** Die Registerkarte „Signals“ des ES5340-MeasTime Device

Alle Parameter sind während der Laufzeit des Modells auf dem Experimentaltar get editierbar.

### *Measurement Procedure*

---

In diesem Listenfeld wird das Messverfahren ausgewählt. Im Abschnitt 23.15.2 auf Seite 822 finden Sie detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren. Im Listenfeld bedeutet die Notation --/-- eine Auswertung an steigenden Flanken, die Notation --\-- hingegen eine Auswertung an fallenden Flanken.

Wird im Listenfeld die Option „No Measurement“ eingestellt, so wird keine Messung durchgeführt.

#### **Hinweis**

*Bitte deaktivieren Sie Messungen, die nicht benötigt werden, da diese unnötige Rechenzeiten verursachen.*

### *Input Channel*

---

In diesem Listenfeld wird der Hardwarekanal festgelegt, auf dem die Messung durchgeführt werden soll.

### *Timeout Monitoring*

---

Bei asynchronen Messungen erfolgt die Überprüfung auf Timeout in periodischen Abständen. Die Zeitdauer kann für jeden Hardwarekanal getrennt eingestellt werden. Wird während der Timeout-Periode keine Flanke auf dem zur Messung gehörigen Hardwarekanal festgestellt, so wird ein Timeout erkannt. Jede Flanke auf dem Hardwarekanal setzt die Timeout-Überwachung zurück.

Mit dem Listenfeld „Timeout Monitoring“ kann die Timeout-Überwachung deaktiviert (Option „Inactive“) oder aktiviert (Optionen „Intvl Predef“ und „Intvl InpDep“) werden. Der Unterschied zwischen den Optionen „Intvl Predef“ und „Intvl InpDep“ besteht im Messwert, der im Timeout-Fall an das Modell übermittelt wird. Ist die Option „Intvl Predef“ eingestellt, so wird in diesem Fall der im numerischen Eingabefeld „Default Timeout Value“ eingetragene Wert übermittelt. Bei der Option „Intvl InpDep“ ist der zurück gelieferte Wert abhängig vom Messverfahren, wobei diese Option jedoch nur bei einigen Messverfahren zur Verfügung steht.

### *Default Timeout Value*

---

Dieses numerische Eingabefeld ist nur editierbar, wenn im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt ist. Das Eingabefeld definiert in diesem Fall den Messwert, der im Timeout-Fall an das Modell übermittelt wird.

### *Timeout Interval*

---

Hiermit lässt sich das Timeout Interval festlegen – der Wertebereich liegt zwischen 0.1 ms und 10000 ms.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Measurement Procedure	uint32	Ja	Messverfahren: „No Measurement“ „High Time [µs]“ „Low Time [µs]“ „Cycle Time --/-- [µs]“ „Cycle Time --\-- [µs]“ „Frequency --/-- [Hz]“ „Frequency --\-- [Hz]“ „Duty Cycle L/(L+H) --/--“ „Duty Cycle L/(L+H) --\--“ „Duty Cycle H/(L+H) --/--“ „Duty Cycle H/(L+H) --\--“ Level (Active High) Level (Active Low)
Input Channel	uint8	Ja	Hardwarekanal, auf dem die Messung durchgeführt wird. 00 ... 19: Kanalnummer
Timeout Monitoring	uint8	Ja	Timeout-Überwachung. 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant im „Intvl Predef“-Modus zur Timeout-Überwachung.
Timeout Interval	float	Ja	0.1...10000 ms
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

**Tab. 23-15** ES5340-MeasTime Device: Konfigurationsparameter der Registerkarte „Signals“

## 23.16 ES5340-MeasAngle Device

---

### 23.16.1 Pulsweitenmessungen

---

In diesem Abschnitt werden die Eigenschaften von Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: H-Time n-th Pulse (Pu Qual.) [ $\mu$ s]
- Angular: L-Time n-th Pulse (Pu Qual.) [ $\mu$ s]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer des zu vermessenden Pulses
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Bei den Pulsweitenmessungen handelt es sich um puls-qualifizierte Messungen, d.h. es werden nur solche Pulse vermessen und gezählt, deren öffnende und schließende Flanke innerhalb ein und desselben Winkelfensters liegen (gültiger Puls). Ermittelt wird die Pulsweite des n-ten gültigen Pulses im gewählten Winkelfenster. Der Messwert wird an der oberen Winkelfenstergrenze berechnet.





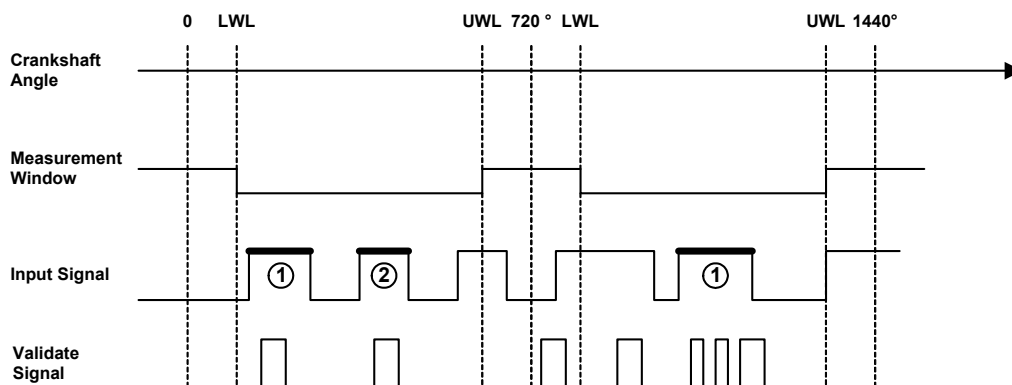
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“ (die Option „Intvl InpDep“ wird nicht unterstützt)
- Default Timeout Value

### Hinweis

Die Nummer des Validierungskanals wird vorgegeben. Er wird im Listenfeld „Reference Channel“ angezeigt. Die Hardwarekanäle 0 und 1 können ein Paar bilden wobei 0 als Messkanal gewählt werden kann, Kanal 1 ist dann der Validierungskanal. Es kann jedoch auch Kanal 1 als Messkanal gewählt werden, in diesem Fall wäre Kanal 0 der Validierungskanal. Das Selbe gilt jeweils für die Kanalpaare (2, 3), (4, 5), (6, 7), (8, 9), (10, 11), (12, 13), (14, 15), (16, 17), (18, 19).

### Funktionsbeschreibung

Abb. 23-24 veranschaulicht die Funktionsweise von winkelsynchronen Pulsweitenmessungen mit Validierungsoption anhand einer High-Zeit-Messung mit high-aktivem Validierungssignal. Es werden nur solche Pulse berücksichtigt, deren öffnende und schließende Flanke innerhalb ein- und desselben Winkel Fensters liegen und die vollständig einen aktiven Validierungspuls umschließen (validierter Puls).



**Abb. 23-24** Darstellung der Funktionsweise einer winkelsynchronen Pulsweitenmessung mit Validierungssignal (Messverfahren „Angular: H-Time n-th Pulse (H-Valid.) [ $\mu$ s]“). Validierte Pulse sind mit erhöhter Linienstärke hervorgehoben.

Es wird der n-te validierte Puls innerhalb eines Winkel Fensters gemessen. Die Pulsnummer wird im Listenfeld „Pulse Number“ der Registerkarte „Signals“ definiert.

Die Übermittlung der gemessenen Pulsweite an das Modell erfolgt an der oberen Winkel Fenstergrenze.

### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720° KW an der oberen Winkel Fenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, da zum Beispiel die Lowzeit des fünften validier-

ten bzw. aktivierten Pulses zu vermessen ist, aber nur vier solche Pulse auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt. Die Option „Intvl InpDep“ wird nicht unterstützt.

### 23.16.3 Additive Pulsweitenmessungen

---

In diesem Abschnitt werden die Eigenschaften additiver Pulsweitenmessungen beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Additive Hightime [ $\mu\text{s}$ ]
- Angular: Additive Lowtime [ $\mu\text{s}$ ]

Zulässige Messoptionen sind:

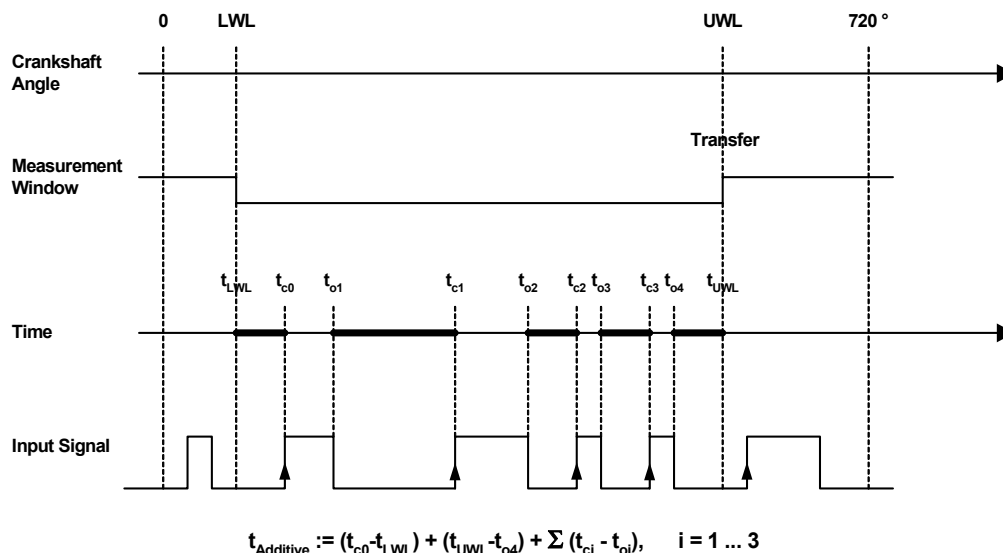
- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird
- Timeout Monitoring  
Alle drei Optionen Inactive, Intvl InpDep und Intvl Predef werden unterstützt.
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die additive Zeit ergibt sich als Summe aller Zeitsegmente innerhalb eines Winkelfensters, in denen das Signal aktiv ist, unabhängig davon, ob sich die öffnenden oder schließenden Flanken der Pulse innerhalb oder außerhalb eines

Winkelfensters befinden. Abb. 23-25 zeigt beispielhaft die Messwertberechnung bei einer additiven Lowzeit-Messung. Die additive Zeit wird an der oberen Winkelfenstergrenze an das Modell übermittelt.



**Abb. 23-25** Darstellung der Funktionsweise einer winkelsynchronen additiven Lowzeit-Messung (Messverfahren Angular: Additive Lowtime [µs]). Die additive Zeit setzt sich aus der Summe der fett markierten Liniensegmente zusammen.

*Timeout-Überwachung*

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Ein Timeout wird erkannt, wenn innerhalb des Winkelfensters keine Flanke auftrat.

Tab. 23-16 listet die Messwerte auf, die im Timeout-Fall an das Modell übergeben werden, wenn im Listenfeld „Timeout Monitoring“ die Einstellung „Intvl InpDep“ ausgewählt ist.

Messverfahren	Signalpegel im Winkelfenster	Messwert im Timeout-Fall
Angular:	low	0
Additive Hightime [µs]	high	Zeitdifferenz: Zeitpunkt obere Winkelfenstergrenze – Zeitpunkt untere Winkelfenstergrenze (t <sub>UWL</sub> – t <sub>LWL</sub> )
Angular:	low	Zeitdifferenz: Zeitpunkt obere Winkelfenstergrenze – Zeitpunkt untere Winkelfenstergrenze (t <sub>UWL</sub> – t <sub>LWL</sub> )
Additive Lowtime [µs]	high	0

**Tab. 23-16** Additive Pulsweitenmessungen: Messwert im Timeout-Fall bei Einstellung von „Intvl InpDep“ im Listenfeld „Timeout Monitoring“

#### 23.16.4 Vermessung von Flanken: Winkelstempel

---

In diesem Abschnitt wird die Erfassung der Winkelstempel von Flanken beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Rising Edge of n-th Pulse [°CA]
- Angular: Falling Edge of n-th Pulse [°CA]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer der zu vermessenden Flanke
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird
- CS Angle Reference  
Referenzwinkel
- Timeout Monitoring  
Entweder Inactive oder Intvl Predef, die Option „Intvl InpDep“ wird nicht unterstützt
- Default Timeout Value

##### *Funktionsbeschreibung*

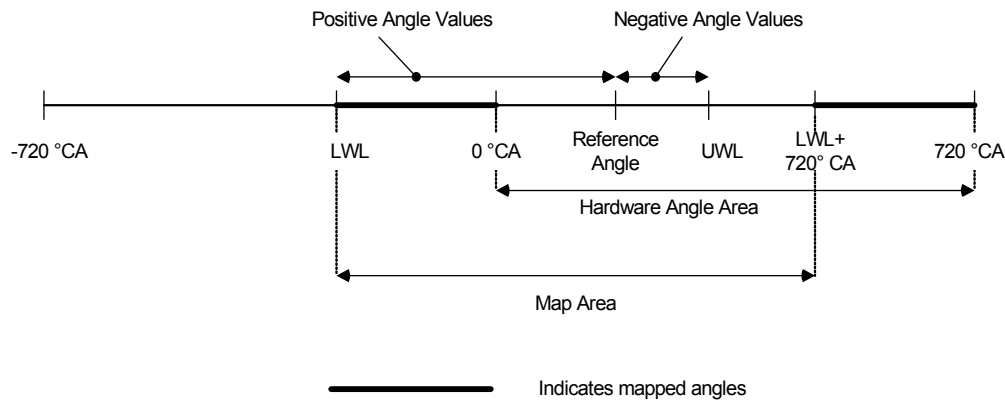
---

Die Messungen erfassen den Winkel der n-ten steigenden oder fallenden Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenähler auf 0 zurückgesetzt.

Der von der Hardware gelieferte Kurbelwellenwinkel einer Flanke liegt grundsätzlich im Bereich [0 °KW, 720 °KW]. Der gemessene Winkel wird von der ES5340.2-ICE-Firmware in den Bereich [LWL, LWL + 720 °KW] abgebildet und anschließend die Differenz mit dem Referenzwinkel berechnet. Diese Differenz ist der von der Messung gelieferte Messwert, sie ist positiv, wenn der fragliche Winkel kleiner als der Referenzwinkel ist (vgl. Abb. 23-26 auf Seite 834).

Ein Beispiel hierzu: Es sei ein Winkelfenster von [-480 °KW, 120 °KW] definiert worden. Der Referenzwinkel soll bei -80 °KW liegen. In einem ersten Fall soll der gemessene Winkel bei 60 °KW liegen. Für Winkelwerte zwischen 0 °KW und (-480 °KW + 720 °KW = 240 °KW) ist keine Abbildung erforderlich, der gelieferte Messwert ergibt sich zu (-80 °KW - 60 °KW = -140 °KW).

In einem zweiten Fall soll der gemessene Winkel bei 540 °KW liegen. Dieser Winkel ist größer als 240 °KW und wird so abgebildet, dass er im Winkelfenster zu liegen kommt. Es werden also 720 °KW vom Winkelwert abgezogen ( $540 \text{ °KW} - 720 \text{ °KW} = -180 \text{ °KW}$ ). Der Messwert ergibt sich zu ( $-80 \text{ °KW} + 180 \text{ °KW} = 100 \text{ °KW}$ ).



**Abb. 23-26** Funktionsweise einer Winkelmessung.

Da der Referenzwinkel im Bereich  $[-720 \text{ °KW}, 720 \text{ °KW}]$  liegen kann, liefern die Messfunktionen Winkelwerte aus dem Bereich  $[-1440 \text{ °KW}, 1440 \text{ °KW}]$ .

#### **Hinweis**

Die Auflösung der gemessenen Winkel wird durch den Winkeltaktgenerator vorgegeben.

#### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Wenn der Messwert nicht berechnet werden konnte, da z.B. der Winkel der fünften steigenden Flanke zu vermessen ist, aber nur vier solche Flanken auftraten, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt. Die Option „Intvl InpDep“ wird nicht unterstützt.

#### 23.16.5 Vermessung von Flanken: Zeitstempel

In diesem Abschnitt wird die Erfassung der Zeitstempel von Flanken beschrieben. Im Einzelnen wird die Funktionsweise folgender Messungen erläutert:

- Angular: Time Stamp of n-th Rising Edge [ $\mu\text{s}$ ]
- Angular: Time Stamp of n-th Falling Edge [ $\mu\text{s}$ ]

Zulässige Messoptionen sind:

- Hardware Channel  
Definiert den Hardwarekanal
- Pulse Number  
Definiert die Nummer der zu vermessenden Flanke
- Angle Window Number  
Definiert die Nummer des Winkelfensters in dem gemessen wird.

- Timeout Monitoring  
Entweder „Inactive“, „Intvl Predef“ oder „Intvl InpDep“
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messungen erfassen den Zeitstempel der n-ten steigenden oder fallenden Flanke innerhalb eines Winkelfensters. Mit dem Überschreiten der unteren Winkelfenstergrenze wird der Flankenzähler auf 0 zurückgesetzt.

#### **Hinweis**

*Der gelieferte Zeitstempel hat keinen Bezug zu anderen Zeitstempeln im LABCAR-System. Es ist ein rein ES5340.2-ICE-interner Zeitstempel mit einer Auflösung von 8 ns.*

#### *Timeout-Überwachung*

---

Bei aktiver Timeout-Überwachung wird alle 720 °KW an der oberen Winkelfenstergrenze eine Überprüfung auf Timeout durchgeführt. Ein Timeout wird erkannt, wenn der Messwert nicht berechnet werden konnte, da z.B. der Zeitstempel der fünften steigenden Flanke zu erfassen ist, aber nur vier solche Flanken auftraten.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl InpDep“ eingestellt, so wird als Messwert der Zeitstempel an das Modell übermittelt, der in der Nockenwellenumdrehung ermittelt wurde, in der letztmalig kein Timeout auftrat. Mit anderen Worten: Es wird einfach der alte Zeitstempel beibehalten.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

### 23.16.6 Pulszählung

---

In diesem Abschnitt werden die Eigenschaften folgender Messverfahren zur Pulszählung beschrieben:

- Angular: Number of High-Pulses
- Angular: Number of Low-Pulses

Zulässige Messoptionen sind:

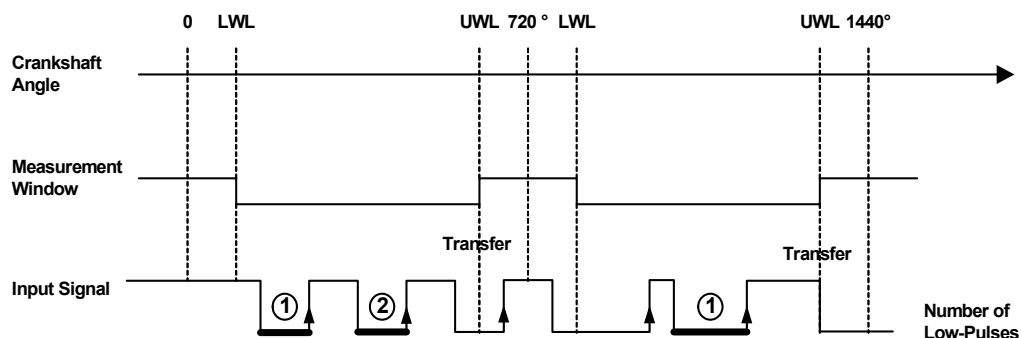
- Hardware Channel  
Definiert den Hardwarekanal
- Angle Window Number  
Definiert die Nummer des Winkelfensters, in dem gemessen wird
- Timeout Monitoring  
Entweder „Inactive“ oder „Intvl Predef“
- Default Timeout Value

#### *Funktionsbeschreibung*

---

Die Messverfahren erfassen die Anzahl der gültigen aktiven Pulse innerhalb eines Winkelfensters. Gültige Pulse sind Pulse, die mit öffnender und schließender Flanke in ein- und demselben Winkelfenster liegen.

Die ermittelte Summe wird an der oberen Winkelfenstergrenze an das Modell übermittelt. (Abb. 23-27).



**Abb. 23-27** Darstellung der Funktionsweise winkelsynchroner Messverfahren zur Pulszählung (Messverfahren Angular: Number of Low-Pulses). Pulse, die gezählt werden, sind durch erhöhte Linienstärke hervorgehoben.

#### Timeout-Überwachung

Bei aktiver Timeout-Überwachung wird eine Überprüfung auf Timeout alle 720 °KW an der oberen Winkelfenstergrenze durchgeführt. Ein Timeout wird erkannt, wenn keine Flanke im Winkelfenster auftrat.

Ist im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt, so wird der vom Anwender definierte Default Timeout Value an das Modell als Messwert übermittelt.

#### 23.16.7 Globals (ES5340-MeasAngle Device)

Das ES5340-MeasAngle Device dient zur Spezifikation und Konfiguration von winkelbasierten (synchronen) Messungen. Jedes ES5340-MeasAngle Device bietet 32 Messungen, die frei auf die 20 Hardwarekanäle der ES5340.2-ICE verschaltet werden können. Bis zu 9 ES5340-MeasAngle Devices werden pro ES5340.2-ICE unterstützt, so dass insgesamt 288 Messungen konfiguriert werden können.

In der Registerkarte „Globals“ sind keine Einstellungen vorzunehmen.

#### 23.16.8 Groups (ES5340-MeasAngle Device)

Das ES5340-MeasAngle Device besitzt eine Signalgruppe, die von der ES5340.2-ICE zum Experimentaltarget übertragen wird. Sie enthält sämtliche Messdaten wie Messwerte, Trigger- und Timeout-Informationen.

Dieser Signalgruppe ist eine Betriebssystem-Task mit periodischer Aktivierung zuzuordnen. Die Aktivierungsperiode richtet sich nach der Dynamik und Periodendauer der zu vermessenden Signale.

#### Die RTIO-Signale der Signalgruppe „MeasVal“

Die Signalgruppe „MeasVal“ umfasst 65 RTIO-Signale. Das Signal „TriggerBitField\_00“ ist als Bitfeld zu interpretieren – es umfasst 32 Bit. In diesem Bitfeld sind die Trigger- oder Update-Daten der 32 Messungen codiert, d.h. es zeigt an, welche Messwerte seit der letzten Aktivierung des Leseprozesses für die Signalgruppe „MeasVal“ aktualisiert wurden. Ist ein Bit gesetzt, so wurde der



Messwert der zugehörigen Messung neu ermittelt. Es spielt dabei keine Rolle, ob die Aktualisierung des Messwerts infolge eines Timeouts oder aufgrund einer regulären Messwertberechnung erfolgte – in beiden Fällen wird das Aktualisierungs-Bit der Messung gesetzt.

Die RTIO-Signale „MeasVal\_00“ bis „MeasVal\_31“ enthalten die Messwerte der 32 Messungen. Es handelt sich dabei um regulär ermittelte Messwerte oder (im Falle eines Timeouts) um den für diesen Fall vorgesehenen Timeout-Wert.

Wenn eine Messung nicht verwendet wird, wird dem zugehörigen Messwert 8888.0 zugewiesen. Die physikalische Einheit der Messwerte hängt vom Messverfahren ab:

- Zeitmessungen (Zeitstempel, (additive) Pulsweiten-Messungen, Periodendauern) werden in Mikrosekunden angegeben
- Winkelmessungen und Winkelstempel werden in Grad Kurbelwelle (°KW) angegeben
- alle anderen Messungen (Tastverhältnisse, Pulszählung) sind dimensionslos

Die RTIO-Signale „Tout\_00“ bis „Tout\_31“ enthalten die Ergebnisse der Timeout-Überwachung für den jeweiligen Messwert.

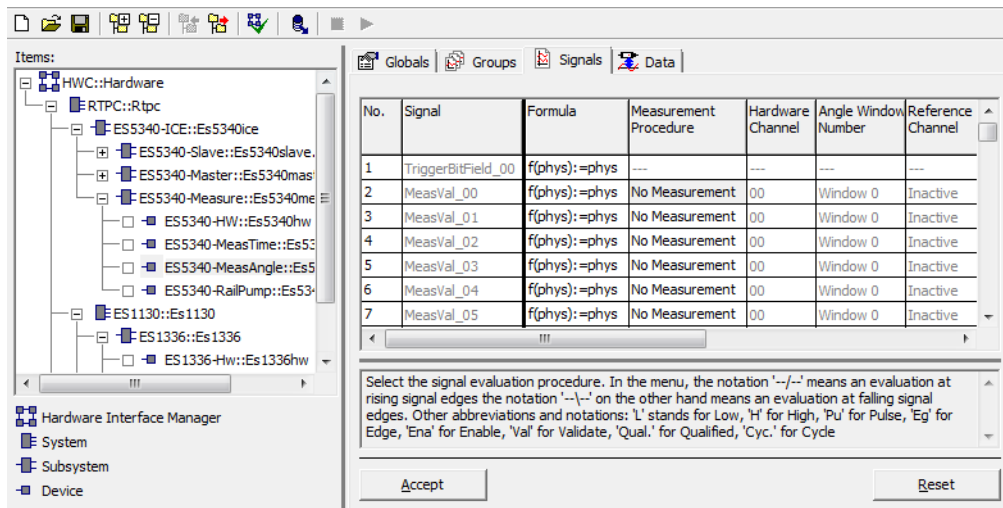
Tab. 23-17 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
TriggerBitField_00	uint32	Bitfeld mit Update-Informationen der 32 Messungen. Messung 0: LSB Messung 31: MSB Bitwert 0: Messwert ist unverändert Bitwert 1: Messwert wurde aktualisiert
MeasVal_00 ... MeasVal_31	real64	Messwert. Wenn die Messung nicht genutzt wird, wird als Wert -8888.0 ausgegeben. Physikalische Einheit des Messwerts: - Zeitmessungen erfolgen in $\mu\text{s}$ - Frequenzmessungen erfolgen in Hz - Winkelmessungen erfolgen in °KW - Spannungsmessungen erfolgen in V - Integralbestimmungen erfolgen in $\text{V} \cdot \mu\text{s}$
Tout_00 ... Tout_31	uint8	Ergebnis der Timeout-Überwachung 0: Kein Timeout festgestellt 1: Timeout festgestellt 2: Timeout-Überwachung ist inaktiv
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 23-17** ES5340-MeasAngle Device: Die RTIO-Signale der Signalgruppe „MeasVal

### 23.16.9 Signals (ES5340-MeasAngle Device)

In der Registerkarte „Signals“ wird die Konfiguration der 32 Messungen eines ES5340-MeasAngle Device durchgeführt. Abb. 23-28 zeigt die RTIO-Parameter der Registerkarte „Signals“ – Tab. 23-18 auf Seite 840 fasst die Eigenschaften der einzelnen Parameter zusammen.



**Abb. 23-28** Die Registerkarte „Signals“ des ES5340-MeasAngle Device

Alle Parameter sind während der Laufzeit des Modells auf dem Experimentaltarget editierbar.

#### Measurement Procedure

In diesem Listenfeld wird das Messverfahren ausgewählt. In den Abschnitten ab Seite 828 finden Sie detaillierte Beschreibungen und Konfigurationsrichtlinien zu den einzelnen Verfahren. Im Listenfeld bedeutet die Notation '-/-' eine Auswertung an steigenden Flanken, die Notation '-\-' hingegen eine Auswertung an fallenden Flanken.

Wird im Listenfeld die Option „No Measurement“ eingestellt, so wird keine Messung durchgeführt.

#### **Hinweis**

*Bitte deaktivieren Sie Messungen, die nicht benötigt werden, da diese unnötige Rechenzeiten verursachen.*

#### Hardware Channel

In diesem Listenfeld wird der Hardwarekanal festgelegt, auf dem die Messung durchgeführt werden soll.

#### Angle Window Number

Dieses Listenfeld ist nur editierbar, wenn im Listenfeld „Measurement Procedure“ ein winkelsynchrones Messverfahren ausgewählt ist. Es wird das Winkel Fenster definiert, innerhalb dessen die Messungen durchgeführt werden. Drei Winkel Fenster stehen zur Auswahl, es sind die Winkel Fenster des zugeordneten Hardwarekanals (siehe hierzu „Signals (ES5340-HW Device)“ auf Seite 819).

### *Reference Channel*

---

Die in Abschnitt 23.16.2 auf Seite 829 beschriebenen Pulsweitenmessungen mit Validierungssignal benötigen zur Durchführung einen zweiten Hardwarekanal: Der Kanal an dem das Enable- oder Validierungssignal anliegt. Das Feld „Reference Channel“ zeigt den Hardwarekanal an, an den (bei Auswahl einer solchen Messung) das Enable- bzw. Validierungssignal anzuschließen ist.

Das Feld ist ein reines Anzeigefeld, ist also nicht editierbar und hat bei allen anderen Messungen keine Bedeutung.

### *Pulse Number*

---

Dieses numerische Eingabefeld ist nur editierbar, wenn im Listenfeld „Measurement Procedure“ ein winkelsynchrones Messverfahren ausgewählt ist. Bei puls- und flankenselektiven Messungen wird in diesem Feld die Nummer des Pulses bzw. der Flanke angegeben, der bzw. die vermessen werden soll.

Bei additiven Messungen und Pulszählungen hat dieses Feld keine Bedeutung und ist ebenfalls deaktiviert.

### *CS Angle Reference*

---

Der Winkelbezugspunkt in Grad Kurbelwelle, auf den sich alle Winkelmessungen beziehen. Für alle anderen Messungen ist dieser Parameter ohne Bedeutung und deaktiviert.

### *Timeout Monitoring*

---

Bei winkelsynchronen Messungen erfolgt die Überprüfung auf Timeout alle 720 °KW jeweils an der oberen Winkelfenstergrenze. Es wird ein Timeout erkannt, wenn kein Messwert berechnet werden konnte. Als Beispiel wird ein Timeout erkannt, wenn die Hightime des fünften Pulses im Winkelfenster zu vermessen ist, jedoch nur vier Pulse auftraten.

Mit dem Listenfeld „Timeout Monitoring“ kann die Timeout-Überwachung deaktiviert (Option „Inactive“) oder aktiviert (Optionen „Intvl Predef“ und „Intvl InpDep“) werden. Der Unterschied zwischen den Optionen „Intvl Predef“ und „Intvl InpDep“ besteht im Messwert, der im Timeout-Fall an das Modell übermittelt wird. Ist die Option „Intvl Predef“ eingestellt, so wird in diesem Fall der im numerischen Eingabefeld „Default Timeout Value“ eingetragene Wert übermittelt. Bei der Option „Intvl InpDep“ ist der zurück gelieferte Wert abhängig vom Messverfahren, wobei diese Option jedoch nur bei einigen Messverfahren zur Verfügung steht. Näheres hierzu finden Sie in den Abschnitten ab Seite 828.

### *Default Timeout Value*

---

Dieses numerische Eingabefeld ist nur editierbar, wenn im Listenfeld „Timeout Monitoring“ die Option „Intvl Predef“ eingestellt ist. Das Eingabefeld definiert in diesem Fall den Messwert, der im Timeout-Fall an das Modell übermittelt wird.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Measurement Procedure	uint32	Ja	Messverfahren: „No Measurement“ „Angular: Additive Hightime [µs]“ „Angular: Additive Lowtime [µs]“ „Angular: Rising Edge of n-th Pulse [°CA]“ „Angular: Falling Edge of n-th Pulse [°CA]“ „Angular: H-Time n-th Pulse (H-Valid.) [µs]“ „Angular: H-Time n-th Pulse (L-Valid.) [µs]“ „Angular: H-Time n-th Pulse (Pu Qual.) [µs]“ „Angular: L-Time n-th Pulse (Pu Qual.) [µs]“ „Angular: Time Stamp of n-th Rising Edge [µs]“ „Angular: Time Stamp of n-th Falling Edge [µs]“ „Angular: Number of High-Pulses“ „Angular: Number of Low-Pulses“
Hardware Channel	uint8	Ja	Hardwarekanal, auf dem die Messung durchgeführt wird. 0 ... 19: Kanalnummer
Angle Window Number	uint8	Ja	Winkelfensterauswahl (Wertebereich: 0, 1, 2) Nur relevant / editierbar bei winkelsynchronen Messungen.
Reference Channel	-	Nein	Referenzkanal bei Messungen mit Enable- und Validierungssignal. Reines Anzeigefeld, nicht editierbar.
Pulse Number	uint16	Ja	Bei puls- und flankenselektiven Messungen: Die Nummer des Pulses bzw. der Flanke, die vermessen werden soll (Wertebereich: 1, 2, ... 32)
CS Angle Reference	real64	Ja	Winkelbezugspunkt in Grad Kurbelwelle auf den sich alle Winkelmessungen beziehen (Wertebereich: [-720.0 °KW, 720.0 °KW])
Timeout Monitoring	uint8	Ja	Timeout-Überwachung. 0: „Inactive“ 1: „Intvl Predef“ 2: „Intvl InpDep“
Default Timeout Value	real32	Ja	Messwert im Timeout-Fall. Nur relevant im „Intvl Predef“-Modus zur Timeout-Überwachung.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

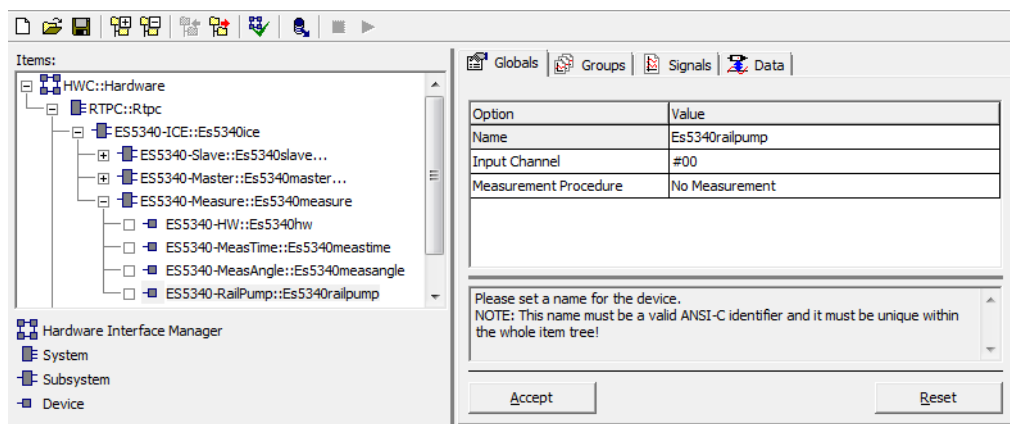
**Tab. 23-18** ES5340-MeasAngle Device: Konfigurationsparameter der Registerkarte „Signals“

## 23.17 ES5340-RailPump Device

### 23.17.1 Globals (ES5340-RailPump Device)

Für jedes ES5340-RailPump Device stehen vier Messkanäle des selben Messverfahrens, aber mit unterschiedlichen Winkelfenstern, zur Verfügung. Es können zwei ES5340-RailPump Devices eingefügt werden, d.h. es stehen Ressourcen für insgesamt acht Messkanäle zur Verfügung.

Die folgende Abbildung zeigt die Registerkarte „Globals“ des ES5340-RailPump Devices.



**Abb. 23-29** Die Registerkarte „Globals“ des ES5340-RailPump Devices

In diesem Register können folgende Parameter eingestellt werden:

#### *Input Channel*

Hier wird der zu verwendende Eingangskanal ausgewählt – dabei werden die Schwellwerteinstellungen des ES5340-HW Devices verwendet. Alle vier Software-Messkanäle dieses ES5340-RailPump Devices beziehen sich auf diesen Hardware-Kanal.

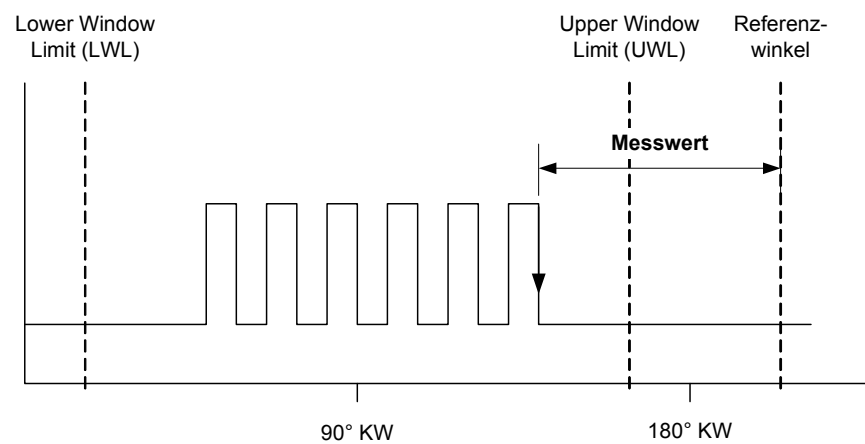
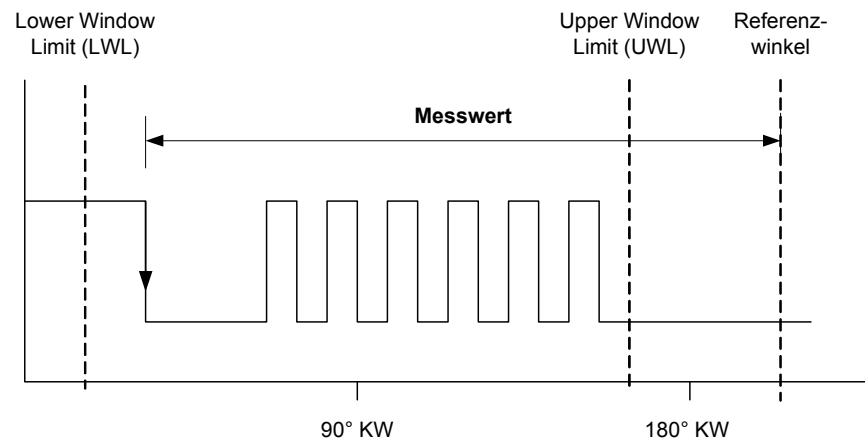
#### *Measurement Procedure*

Hier findet die Auswahl des verwendeten Messverfahrens für alle vier Software-Messkanäle des RTIO-Elements statt – insgesamt stehen folgende vier Messmethoden zur Verfügung:

- Angle of First Rising Edge of Pulse Sequence [°CA]
- Angle of First Falling Edge of Pulse Sequence [°CA]
- Angle of Last Rising Edge of Pulse Sequence [°CA]
- Angle of Last Falling Edge of Pulse Sequence [°CA]

Dabei wird jeweils der Winkelwert der ersten bzw. letzten steigenden bzw. fallenden Flanke gemessen, welche innerhalb eines Winkelfensters (definiert durch LWL und UWL, siehe Abb. 23-30 auf Seite 842) aufgetreten ist. Der Messwert ergibt sich aus einem Referenzwinkelwert abzüglich dieses Winkels. Flanken vor dem Referenzwinkel ergeben ein positives Ergebnis, Flanken nach dem Referenzwinkel ein negatives Ergebnis.

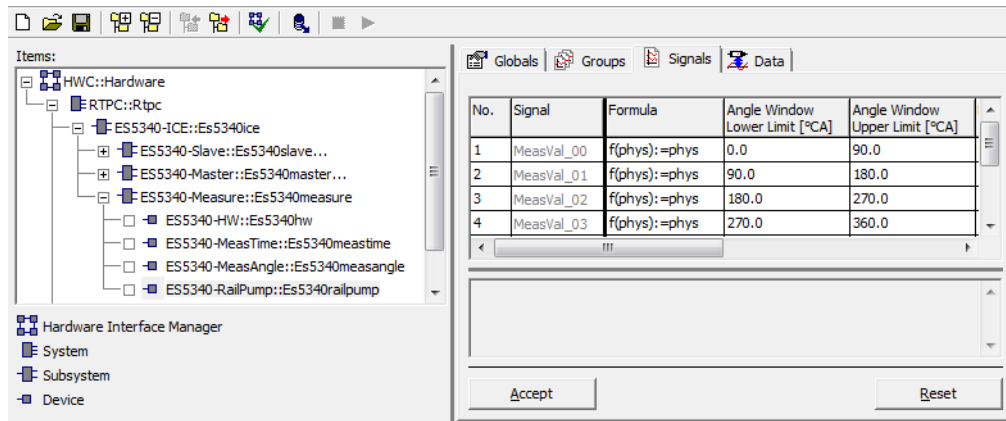
Tritt keine entsprechende Flanke im Winkelfenster auf, wird kein Ergebnis übermittelt und es wird eine Timeout-Bedingung erkannt.



**Abb. 23-30** Messwerte bei „Angle of First Falling Edge of Pulse Sequence“ (oben) und „Angle of Last Falling Edge of Pulse Sequence“ (unten)

### 23.17.2 Signals (ES5340-RailPump Device)

Das ES5340-RailPump Device implementiert eine „MeasVal“ Empfangssignalgruppe.



**Abb. 23-31** Die Registerkarte „Signals“ des ES5340-RailPump Devices

Folgende Signale werden angelegt:

*MeasVal\_xx [xx = 00...03]*

Die Ergebniswerte für Software-Messkanäle 0...3 (Wertebereich -720.0...720.0 °KW, -8888.0 falls Kanal unbenutzt)

*Tout\_xx [xx= 00...03]*

Timeout-Signale für die Software-Messkanäle 0...3

- 0 = kein Timeout
- 1 = Timeout erkannt
- 2 = Timeout nicht konfiguriert
- 4 = Winkelinterface nicht synchron

*Offset Angle*

Um diesen Winkel können Winkelfenster und Referenzwinkel gegenüber dem Kurbelwellenwinkel verschoben werden.

In der Registerkarte „Signals“ können spezifisch für jeden Messkanal folgende Einstellungen vorgenommen werden.

*Angle Window Lower Limit*

Der Startwinkel des Messfensters (entspricht der Größe „LWL“ in Abb. 23-30 auf Seite 842).

*Angle Window Upper Limit*

Der Stoppwinkel des Fensters (entspricht der Größe „UWL“ in Abb. 23-30 auf Seite 842)

*Reference Angle*

Der Bezugswinkel für die Berechnung des Messwertes.

### *Timeout Monitoring*

---

Die Betriebsart für Timeout-Behandlung.

- Inactive  
Keine Timeout-Erkennung, der vorhergehende Wert wird übermittelt
- Intvl Predef  
Der Ersatzwert für den Timeout-Fall wird übermittelt

### *Default Timeout Value*

---

Der Ersatzwert für den Timeout-Fall.



## 23.18 Arbiträre Signalgeneratoren

---

Auf der ES5340.2-ICE stehen acht analoge und acht digitale Signalgeneratoren zur Verfügung. Jeder der Signalgeneratoren kann dabei eine der 16 zur Verfügung stehenden Wellenformen abspielen. Als Taktquellen stehen ein zentraler RPM-Generator und ein pro Signalgenerator vorhandener variabler Taktgenerator (Maximalfrequenz: 1 MHz) zur Verfügung.

Pro Signalgenerator kann eine individuelle Grundphase sowie eine zusätzliche Phasenverschiebung gewählt werden. Die Geschwindigkeit, mit der eine Änderung der Phasenverschiebung wirkt, ist wählbar.

Bei Verwendung des variablen Taktgenerators kann die Frequenz des Taktgenerators, der Trigger-Modus (single shot, continuous) sowie ein Triggersignal vorgegeben werden.

Die Amplitude des internen Ausgangssignals des Signalgenerators kann zwischen 0.0 und 1.0 verändert werden.

### 23.18.1 RPM-Generator

---

Die ES5340.2-ICE verfügt über einen zentralen Drehzahlgenerator (RPM-Generator), der ein motordrehzahlspezifisches Taktsignal ausgibt. Dieses Taktsignal kann von den Signalgeneratoren zum Austakten der Wellenformen verwendet werden. Die maximale Drehzahl beträgt 30000 rpm, die Auflösung ca. 0,1 rpm. Das Drehzahlsignal selbst kann über einen Misfire-Generator moduliert werden.

Das Drehzahlsignal kann zu Messzwecken auf den BNC-Anschluss auf der Frontplatte der ES5340.2-ICE gelegt werden.

#### *Winkelauflösung*

---

Die Winkelauflösung beträgt 65536 Punkte pro Zyklus. Bei einem typischen Viertaktmotor mit einer Periode von 720 °KW entspricht das einer Winkelauflösung von ca. 0.01 °KW.

### 23.18.2 Wellenformspeicher für Signalgeneratoren

---

Es stehen 16 Wellenformen zur Verfügung, die von den arbiträren Signalgeneratoren verwendet werden können. Die Wellenformen können vom Anwender über Tabellen beschrieben werden. Über ein Interpolationsverfahren wird der Signalverlauf in der Tabelle in die jeweilige Wellenform geschrieben.

#### **Wellenformauflösung:**

Die maximale Auflösung einer Wellenform wird durch die maximal mögliche Zahl von 65536 Stützstellen bestimmt. Auch hier kann die Auflösung in Zweierpotenzen bis auf 16 Punkte reduziert werden; dabei ist zu beachten, dass die Auflösung ( $1/(\text{Anzahl der Stützstellen})$ ) einer Wellenform kleiner oder gleich der Winkelauflösung sein muss. Typischerweise sollte die Auflösung einer Wellenform mit der Winkelauflösung übereinstimmen.

Die Wellenformen werden von den Signalgeneratoren ausgetaktet. Dabei kann entweder der zentrale RPM-Generator als Taktquelle fungieren, oder es wird ein variabler Frequenzgenerator (maximale Frequenz: 1 MHz) im Signalgenerator verwendet.

### Wellenformauflösung kleiner als Winkelauflösung:

Falls über den Signalgenerator (unter Verwendung des variablen Frequenzgenerators) ein hochfrequentes Signal ausgegeben werden soll, ist es eventuell nötig, die Auflösung einer Wellenform kleiner als die Winkelauflösung zu halten.

Das folgende Beispiel illustriert das Vorgehen:

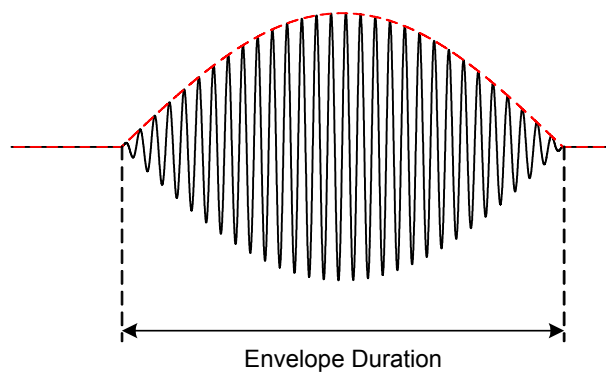
Angenommen, es soll ein Sinussignal von 40 kHz ausgegeben werden, die Signaltabelle beschreibt eine einzige Sinus-Periode. Die Winkelauflösung beträgt 65536 Punkte. Durch die maximale Frequenz des variablen Taktgenerators von 1 MHz ergibt sich dadurch eine maximale Signalfrequenz für das Sinussignal von  $1 \text{ MHz}/65536 = 15,25 \text{ Hz}$ , was natürlich weit von den gewünschten 40 kHz entfernt ist. Durch Verkleinern der Wellenformauflösung auf z.B. 16 Stützstellen wird das Sinussignal mehrfach hintereinander (konkret  $65536/16 = 4096$ -fach) in der Wellenform mit 65536 Stützstellen abgelegt. Damit ergibt sich eine resultierende Maximalfrequenz für das Sinussignal von  $1 \text{ MHz}/16 = 62,5 \text{ kHz}$ , was über der gewünschten Frequenz von 40 kHz liegt. Durch eine entsprechende Absenkung der variablen Taktfrequenz auf 640 kHz kann man damit das gewünschte Sinussignal mit 40 kHz erzeugen.

Das Beispiel zeigt, dass durch eine Verkleinerung der Wellenformauflösung gegenüber der Winkelauflösung die Wellenformauflösung nicht wirklich verkleinert wird. Es wird lediglich das Signal der Signaltabelle mehrfach hintereinander in die Wellenform geschrieben und damit die „sichtbare“ Auflösung verkleinert.

#### 23.18.3 Klopfsignalgenerator

Die bei einem Verbrennungsmotor auftretenden Klopfgeräusche können durch den Klopfsignalgenerator nachgebildet werden. Ein Klopfpaket besteht aus einzelnen Klopfpaketen. Ein Klopfpaket selbst besteht aus einer Sinusschwingung mit wählbarer Frequenz und einer die Sinusschwingung modulierenden Hüllkurve mit definierbarer Dauer.

Die nachfolgende Abbildung zeigt ein einzelnes Klopfpaket. Dabei wird als Hüllkurve eine Sinus-Halbwellen verwendet.



**Abb. 23-32** Ein Klopfpaket

Auch eine nicht-klopfende Verbrennung erzeugt Geräusche, die von einem realen Körperschall-Klopfensensor erfasst werden. Über die Steuerung der Amplitude des Klopfsignals wird zwischen einer korrekten und einer klopfenden Verbrennung unterschieden.

Zusätzlich gibt es noch eine stochastische Variation der Amplitude eines Klopfpakets. Damit werden die im realen Betrieb auftretenden Änderungen der Klopfsignale nachgebildet.

Ein gewisses Rauschen ist auch dann vorhanden, wenn gerade kein Klopfpaket ausgegeben wird. Dieses Grundrauschen wird z.B. benötigt, um die initiale Diagnose des Sensors bestehen zu können. Moderne Steuergeräte behandeln nicht rauschende Eingänge als fehlerhaft oder nicht vorhanden.

Pro Zylinder kann nun individuell die Winkellage (in °KW) eines Klopfsignals sowie das Auftreten des Klopfereignisses per Wahrscheinlichkeitswert oder mittels Sequenztabelle (siehe „Sequenztabelle“ auf Seite 847) gesteuert werden.

Der Klopfsignalgenerator verfügt über vier interne Ausgänge. Für jeden Ausgang kann gewählt werden, welche Zylinder den jeweiligen Ausgang bedienen. Bei viel-zylindrigen Fahrzeugen ist dabei wichtig, dass sich einzelne Klopfpakete auch überlagern können.

#### **Hinweis**

*Es können sich maximal vier Wellenformen überlagern!*

#### 23.18.4 Zündaussetzsteuerung

---

Zur Nachbildung von Zündaussetzern ist eine entsprechende Steuerung auf der ES5340.2-ICE vorhanden, die eine Modulation der Drehzahl des RPM-Generators in einem bestimmten Winkelbereich bewirkt. Dabei ist es möglich, die Drehzahl über den Kurbelwellenwinkel relativ zur vorgegebenen Drehzahl des RPM-Generators zu verändern (verringern/erhöhen um den Faktor 0,01 bis 2,0). In der typischen Anwendung zur Nachbildung von Zündaussetzern wird die Drehzahl gegenüber der vorgegebenen Drehzahl verringert.

Der Startwirkungswinkel der Drehzahlmodulation ist zylinderindividuell festlegbar. Über einen Wahrscheinlichkeitswert oder über Sequenztabelle (siehe „Sequenztabelle“ auf Seite 847) kann das Wirken der Drehzahlmodulation pro Zylinder gesteuert werden.

Die Drehzahlmodulation kann über vier vorhandenen Modulationsprofile vorgegeben werden, die den Modulationsverlauf über eine komplette Periode von 720 °KW (bzw. 360 ° beim Zweitaktmotor) darstellen. Dabei entspricht der Wert 1.0 einer nicht vorhandenen Modulation, ein Wert von 0.01 verringert die Drehzahl auf 1% der vorgegebenen Drehzahl, der Wert 2.0 verdoppelt die vorgegebene Drehzahl. Pro Zylinder kann individuell eines der vier vorhandenen Modulationsprofile ausgewählt werden.

#### 23.18.5 Sequenztabelle

---

Beim Zündaussetzgenerator und beim Klopfsignalgenerator kommen Sequenztabelle zum Einsatz, die es dem Benutzer ermöglichen, komplexe Abläufe von Klopf- bzw. Fehlzündungen darzustellen.

Dabei wird eine Tabelle mit maximal 100 Stützstellen verwendet. Nach Starten der Sequenz wird pro Periode eine Stützstelle vorangegangen. Bei Fehlzündungen bedeutet dabei ein Wert größer als 0.5 an der jeweiligen Stützstelle, dass eine Fehlzündung in dieser Periode auftritt. Beim Klopfsignalgenerator kann man über diesen Wert in der Tabelle zusätzlich festlegen, wie stark der Klopfsensor das Klopfsignal wahrnehmen soll (nahe Zylinder: großer Wert, ferne Zylinder: kleiner Wert).

Nach 100 Stützstellen wird sofort von vorne begonnen („Sequence trigger = continuous“), oder das Abspielen wird beendet („Sequence trigger = Single Shot“) und muss über ein entsprechendes Triggersignal erneut gestartet werden.

Es ist grundsätzlich möglich, pro Zylinder eine individuelle Sequenz vorzugeben. Jedoch gibt es sowohl beim Zündaussetz- als auch beim Klopfgenerator eine gemeinsame Sequenz („Common Sequence“), auf die alle Zylinder zugreifen können. Das erleichtert das rasche Einstellen von Sequenzen, die für mehrere Zylinder verwendet werden sollen.

#### 23.18.6 MSA-Sensor

---

Eine besondere Verwendung der Signalgeneratoren ist die Nachbildung von Kurbelwellensensoren mit Drehrichtungserkennung (MSA-Sensoren). Ein Zahnpuls hat dabei keine feste Winkelbreite sondern eine feste Pulsdauer. Außerdem ist als Ausgabesignal ein Low-aktives Open-Collector-Signal vorgegeben.

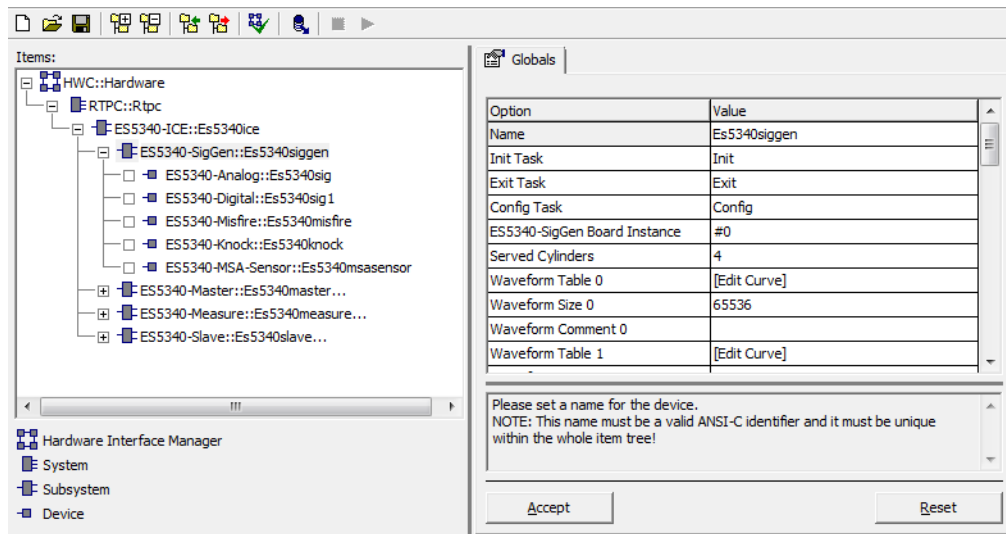
Wird ein MSA-Sensor RTIO-Element verwendet, so wird für alle Wellenformverläufe während der Konfiguration (potentielle) Zahnmitteinformationen berechnet und im Wellenformspeicher abgelegt. Möglicherweise sind aber nicht alle Wellenformen für diesen Algorithmus geeignet – bei ungeeigneten Wellenformen wird bei Auswahl der entsprechenden Wellenform eine Fehlermeldung übermittelt.

## 23.19 ES5340-SigGen Subsystem

### 23.19.1 Globals (ES5340-SigGen Subsystem)

Das ES5340-SigGen Subsystem dient zur Einstellung von RTIO-Parametern, die global wirksam sind, d.h. die auf alle Elemente des ES5340-SigGen Subsystems wirken.

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 23-33** Die Registerkarte „Globals“ des ES5340-SigGen Subsystems

#### *Served Cylinders*

Hier wird die Anzahl der unterstützten Zylinder beim Klopfsignalgenerator und bei der Zündaussetzsteuerung eingestellt. Mögliche Werte sind 1 ... 12.

#### *Waveform Table 0... 15*

Hier können die Signalformen der 16 vorhandenen Wellenformen vorgegeben werden.

#### *Waveform Size 0... 15*

Die Stützstellenanzahl der 16 vorhandenen Wellenformen können über die entsprechenden Optionsfelder vorgegeben werden (siehe auch „Wellenformauflösung“ auf Seite 845 und „Wellenformauflösung kleiner als Winkelauflösung“ auf Seite 846).

#### *Waveform Comment 0... 15*

Um einen einfachen Überblick über die Verwendung der 16 Wellenformen zu haben, kann ein individueller Kommentar vergeben werden.

#### **Hinweis**

*Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.*

MSA Sensors: Logic of Input Waveform

Dieser Parameter legt fest, ob bei der Berechnung der Zahnmitten (für einen MSA-Sensor) ein Signal mit negativer Logik (neg. Puls entspricht „Zahn“) oder positiver Logik (pos. Puls entspricht „Zahn“) zu interpretieren ist. Diese Einstellung gilt für alle 16 Wellenformen.

Der Parameter ist online editierbar, jedoch müssen bei einer Änderung alle Wellenformen neu berechnet und in den Wellenformspeicher auf der Karte geschrieben werden.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Served Cylinders	uint32	Ja	Unterstützte Zylinder für Klopfen und Fehlzündung Wertebereich: 1 ... 12
Resolution	uint32	Ja	Winkelauflösung Wertebereich: 16 ... 65536 in Zweierpotenzen
Waveform Table 0...15	Table	Ja	Die jeweilige Wellenform als Tabelle. Wertebereich: -1.0 ... +1.0
Waveform Size 0...15	uint32	Ja	Stützstellenanzahl der jeweiligen Wellenform Wertebereich: 16 ... 65536 in Zweierpotenzen
Waveform Comment 0...15	Identifizier	Nein	Kommentar zur jeweiligen Wellenform
MSA Sensors: Logic of Input Waveform	bool	Ja	Interpretation der Wellenform für MSA-Sensor Wertebereich: 0 = FALSE (neg. Logik)  , 1 = TRUE (pos. Logik)

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-19** ES5340-SigGen Subsystem: Konfigurationsparameter der Registerkarte „Globals“

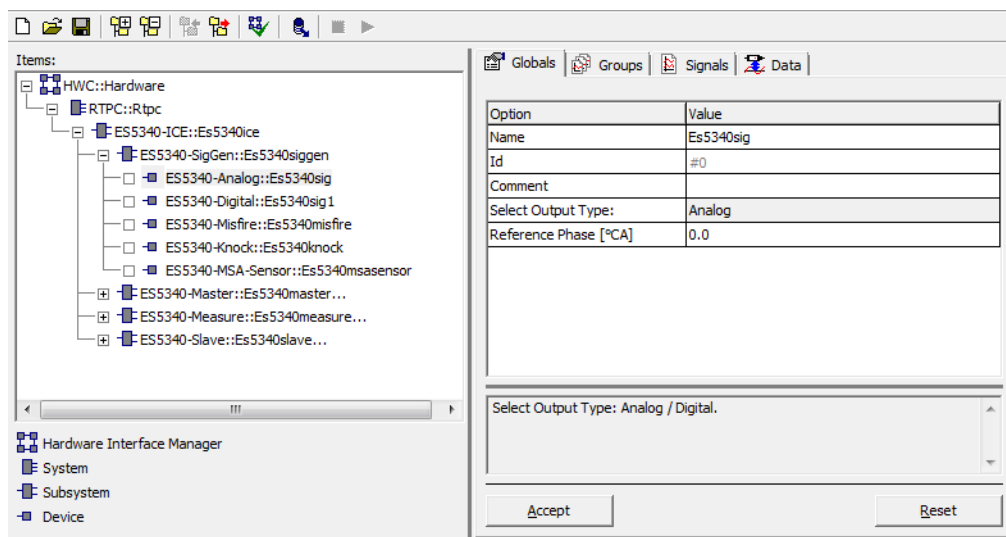
## 23.20 ES5340-SigGen Device

Das ES5340-Analog Device dient zur Konfiguration und Ansteuerung eines der arbiträren Signalgeneratoren auf der ES5340.2-ICE.

Es können bis zu acht analoge und acht digitale Signalgeneratoren eingebunden werden.

### 23.20.1 Globals (ES5340-Analog Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 23-34** Die Registerkarte „Globals“ des ES5340-Analog Device

#### Comment

Für diesen Signalgenerator kann ein beliebiger Kommentar vergeben werden.

#### Hinweis

Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.

#### Select Output Type

Analog oder digital.

#### Reference Phase

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Signalgenerators. Die tatsächliche Phase wird aus der Addition dieses Werts mit dem Signal „PhaseRef“ (siehe „PhaseRef“ auf Seite 853) bestimmt.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Comment	Identifizier	Nein	Benutzerspezifischer Kommentar
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich -720...720 °KW

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-20** ES5340-Analog Device: Konfigurationsparameter der Registerkarte „Globals“

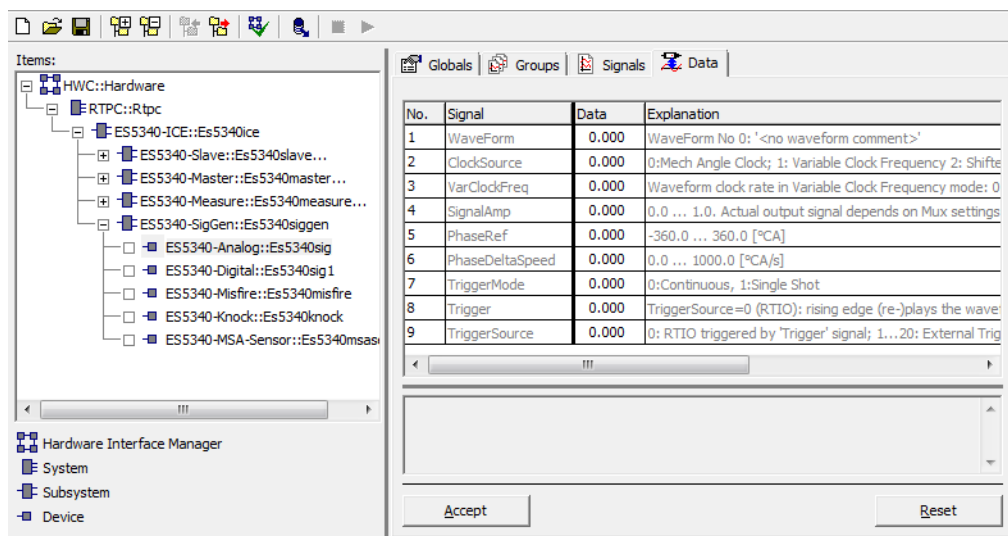
23.20.2 Groups (ES5340-Analog Device)

Das ES5340-Analog Device besitzt eine Signalgruppe, mit der der Signalgenerator gesteuert wird.

23.20.3 Data (ES5340-Analog Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung des Signalgenerators aufgelistet.

Die folgende Abbildung zeigt die Signale in der Registerkarte „Data“:



**Abb. 23-35** Die Registerkarte „Data“ des ES5340-Analog Device

*WaveForm*

Über das Signal „WaveForm“ kann die Wellenform ausgewählt werden, die vom Signalgenerator ausgetaktet werden soll. Es sind dabei Werte von 0 bis 15 möglich, die den zentralen Wellenformen im ES5340-SigGen Subsystem entsprechen.



### *ClockSource*

---

Über das Signal „ClockSource“ wird die Taktquelle für den Signalgenerator bestimmt. Dabei kann zwischen dem zentralen Drehzahlgenerator (RPM-Generator) und einem variablen Taktgenerator ausgewählt werden. Der Wert 0 entspricht dabei dem zentralen Drehzahlgenerator, der Wert 1 dem variablen Taktgenerator und der Wert 2 dem zentralen Taktgenerator mit einem Offset („Shifted Mech Angle Clock“).

### *VarClockFreq*

---

Das Signal „VarClockFreq“ steuert die Frequenz des variablen Taktgenerators. Das Signal ist nur dann wirksam, wenn das Signal „ClockSource“ den Wert 1 (= variabler Taktgenerator) besitzt. Die maximale Frequenz des variablen Taktgenerators beträgt 1 MHz.

### *SignalAmp*

---

Die Ausgangsamplitude des Signalgenerators wird über das Signal „SignalAmp“ gesteuert. Der Wertebereich geht dabei von 0.0 bis 1.0.

Zu beachten ist, dass erst im Ausgangsmultiplexer aus dem internen Signal mit dem Wertebereich -1.0 bis +1.0 ein Spannungssignal gebildet wird.

### *PhaseRef*

---

Dieses Signal beschreibt eine Phasenverschiebung, die zum globalen Parameter „ReferencePhase“ addiert wird. Damit kann eine modellgesteuerte Phasenverschiebung zur Laufzeit durchgeführt werden. Zu beachten ist, dass die Phasenverschiebung mit einer endlichen Phasenänderungsgeschwindigkeit (gesteuert durch den Parameter „PhaseDeltaSpeed“) durchgeführt wird. Daher muss das Signal „PhaseDeltaSpeed“ für die meisten Anwendungen einen Wert größer als 0 aufweisen.

### *PhaseDeltaSpeed*

---

Die Phasenänderungsgeschwindigkeit wird mit dem Signal „PhaseDeltaSpeed“ gesteuert. Eine Änderung des Werts von Signal „PhaseRef“ wirkt nicht unendlich schnell, sondern die Phase wird mit dieser Geschwindigkeit verstellt. Der Wertebereich reicht von 0 (es findet keine Verstellung der Phase statt) bis 1000 °KW/s.

### *TriggerMode*

---

#### **Hinweis**

Das Signal „TriggerMode“ ist nur dann von Bedeutung, wenn der variable Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist (siehe „ClockSource“ auf Seite 853).

Der variable Taktgenerator kann kontinuierlich arbeiten („TriggerMode“ hat den Wert 0) oder er stoppt nach einmaligen Abspielen („Single Shot“) einer kompletten Wellenform („TriggerMode“ hat den Wert 1).

Eine Phasenverschiebung wird auch im Betrieb mit dem variablen Taktgenerator berücksichtigt. Damit kann der Startpunkt des Abspielens gesteuert werden.

Ein Start bzw. Neustart ist durch das Triggersignal „Trigger“ möglich.

Zu beachten ist, dass immer die komplette Wellenform über die Gesamtanzahl aller Stützstellen abgespielt wird. Wenn die Stützstellenanzahl der Wellenform kleiner als die Winkelauflösung gewählt ist, dann wird das gewünschte Ausgangssignal mehrfach im Wellenformspeicher hintereinander abgelegt. Dann wird beim Abspielen im Modus „Single Shot“ das gewünschte Ausgangssignal mehrfach abgespielt (siehe hierzu „Wellenformauflösung kleiner als Winkelauflösung“ auf Seite 846).

### *Trigger*

---

#### **Hinweis**

*Das Signal „Trigger“ ist nur dann von Bedeutung, wenn der variable Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist (siehe „Clock-Source“ auf Seite 853).*

Das Signal „Trigger“ ermöglicht das Starten des variablen Taktgenerators. Eine Flanke von 0 → 1 auf dem „Trigger“-Signal startet das Austakten der Wellenform mit dem variablen Taktgenerator. Eine entsprechende Flanke führt in allen Fällen zum Neustart einer Austaktung, auch wenn das vorherige Abspielen noch nicht beendet ist oder auch im kontinuierlichen Betriebsmodus.

### *TriggerSource*

---

#### **Hinweis**

*Das Signal „TriggerSource“ ist nur dann von Bedeutung, wenn der variable Taktgenerator als Taktquelle für den Signalgenerator ausgewählt ist („Clock-Source“ auf Seite 853).*

Mit diesem Signal kann die Triggerung der Signalgeneratoren über Software (LABCAR-RTC) oder durch einen der digitalen Eingänge 1 bis 20 konfiguriert werden – eine steigende Flanke auf dem ausgewählten Eingang triggert das Austakten der Wellenform.

Bei Verwendung von externen Triggern muss „TriggerSource“ auf 1 gesetzt werden – gleichzeitig muss der Wert von „VarClockFreq“ ungleich Null sein.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
WaveForm	uint8	Ja	Auswahl der Wellenform 0 - 11
ClockSource	uint8	Ja	Taktquelle des Signalgenerators 0: Mech Angle Clock 1: Variable Taktfrequenz 2: Shifted Mech Angle Clock
VarClockFreq	real32	Ja	Frequenz des variablen Signalgenerators in Hertz. 0.0 - 1000000.0 [Hz]
SignalAmp	uint16	Ja	Signalamplitude des Ausgangssignals 0...65535 entspricht 0.0...1.0
PhaseRef	sint16	Ja	Additive Phasenverschiebung -32768...32767 entspricht -360...+360 [°KW]
PhaseDeltaSpeed	uint16	Ja	Phasenänderungsgeschwindigkeit 0 ... 1000 [°KW/s]
TriggerMode	uint8	Ja	Trigger-Modus 0: Kontinuierlich 1: Single-Shot
Trigger	uint8	Ja	Triggersignal Flanke von 0 → 1 startet das Abspielen
TriggerSource	uint8	Ja	Triggerquelle Wertebereich: 0...20 0 = LABCAR-RTC (default) 1...20 = ext. Triggersignal
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet			

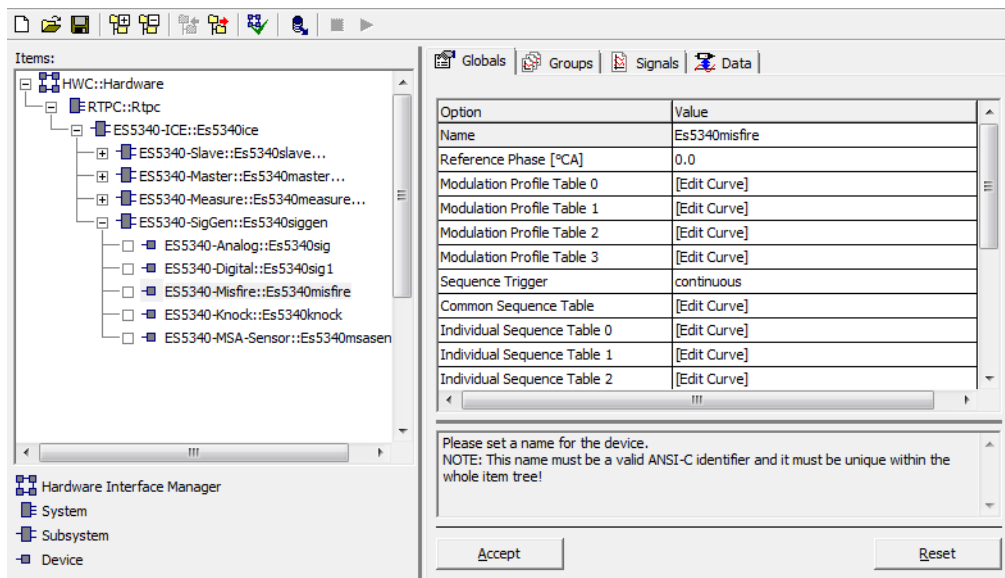
**Tab. 23-21** ES5340-Analog Device: Signale der Registerkarte „Data“

## 23.21 ES5340-Misfire Device

Das ES5340-Misfire Device dient zur Konfiguration und Ansteuerung der Zündaussetzsteuerung auf der ES5340.2-ICE.

### 23.21.1 Globals (ES5340-Misfire Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 23-36** Die Registerkarte „Globals“ des ES5340-Misfire Device

#### *Reference Phase*

Dieser Parameter beschreibt die Grund-Phasenverschiebung der Zündaussetzsteuerung. Die tatsächliche zylinderspezifische Phase wird aus der Addition dieses Werts mit dem jeweiligen Parameter „Start Angle“ aus der Registerkarte „Signals“ bestimmt.

#### *Modulation Profile Table 0 ... 3*

Es stehen vier Modulationsprofile zur Verfügung, die eine Modulation der Vorgabedrehzahl über einen bestimmten Winkelbereich vorgeben (siehe „Zündaussetzsteuerung“ auf Seite 847). Die Modulationsprofile können über den Tabelleneditor vorgegeben werden. In einer Modulationskurve ist die Drehzahlmodulation über dem Kurbelwellenwinkel aufgetragen. Dabei entspricht ein Wert von 1 an einem Winkel der unveränderten Übernahme der Vorgabedrehzahl. Ein Wert kleiner 1 (bis 0.01) verringert die Drehzahl. Ein Wert größer 1 entspricht einer Drehzahlerhöhung.

Der erste Stützstellenwert der Signaltabelle entspricht dem jeweiligen zylinderspezifischem Startwinkel in °KW. Die Signaltabelle beschreibt eine komplette Periode von 720 °KW.

Die Drehzahlmodulation von mehreren Zylindern können sich überlagern - dabei werden die einzelnen zylinderbezogenen Modulationswerte miteinander multipliziert.

### Sequence Trigger

Die Aktivierung der Drehzahlmodulation kann für jeden Zylinder über Sequenztabelle gesteuert werden (siehe „Sequenztabelle“ auf Seite 847). Über diesen Parameter wird entschieden, ob ein kontinuierliches Abspielen der Sequenztabelle („continuous“) oder ein einmaliges Abspielen („single shot“) der Sequenztabelle gewünscht wird. Das Starten des Abspielens erfolgt über das Signal „Sqnc\_Trigger“ (siehe „Sqnc\_Trigger“ auf Seite 860).

#### Common Sequence Table, Individual Sequence Table 0 ... 11

Zur Steuerung einer Drehzahlmodulation stehen 12 zylinderindividuelle Sequenztabelle zur Verfügung (siehe „Sequenztabelle“ auf Seite 847). Zusätzlich steht eine gemeinsam nutzbare Sequenztabelle „Common Sequence Table“ zur Verfügung, die es ermöglicht, einen Sequenzablauf einfach auf mehreren Zylindern zu nutzen.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich 0 ... 720 °KW
Modulation Profile 0...3	table	Ja	Die Stützstellen werden auf eine komplette Periode (720 °KW) abgebildet. Die Werte liegen innerhalb des Wertebereichs von 0.01 bis 2.0
Sequence Trigger	uint8	Ja	Abspielmodus der Sequenztabelle: Continuous: 0 Single Shot: 1
Common Sequence Table Individual Sequence Table 0...11	table	Ja	Sequenztabelle

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

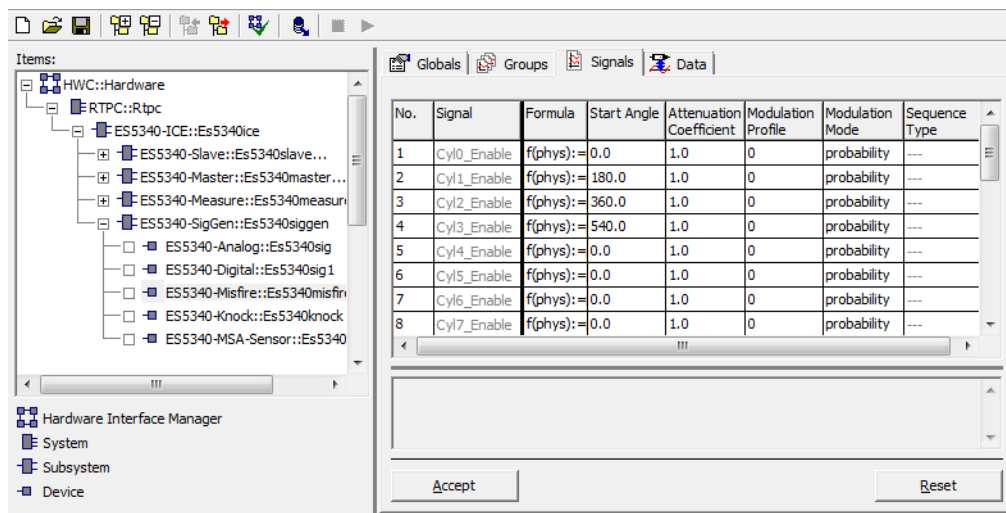
**Tab. 23-22** ES5340-Misfire Device: Konfigurationsparameter der Registerkarte „Globals“

### 23.21.2 Groups (ES5340-Misfire Device)

Das ES5340-Misfire Device besitzt eine Signalgruppe, in der die Signale der Zündaussetzsteuerung verarbeitet werden.

### 23.21.3 Signals (ES5340-Misfire Device)

In der Registerkarte „Signals“ wird die Konfiguration der Zündaussetzsteuerung durchgeführt. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 23-37** Die Registerkarte „Signals“ des ES5340-Misfire Device

#### *Start Angle*

Das Optionsfeld „Start Angle“ dient zur Festlegung des Startwinkels des Zündaussetzer-Modulationsprofils für den jeweiligen Zylinder. Zu diesem Winkel wird der Parameter „Reference Phase“ aus der Registerkarte „Globals“ (siehe „Reference Phase“ auf Seite 856) addiert.

#### *Attenuation Coefficient*

Über diesen Parameter kann die Stärke der Drehzahlmodulation zylinderindividuell reduziert werden. Ein Wert von 1.0 bedeutet, dass die im ausgewählten Modulationsprofil beschriebene Drehzahlmodulation vollständig angewendet wird. Ein Wert kleiner 1 reduziert die Wirkung anteilig, ein Wert von 0.0 schließlich führt dazu, dass für den Zylinder keine Drehzahlmodulation durchgeführt wird.

#### *Modulation Profile*

Pro Zylinder kann eines der vier vorhandenen Modulationsprofile der Registerkarte „Globals“ ausgewählt werden.

#### *Modulation Mode*

Das Vorhandensein eines Zündaussetzers in einer Periode kann entweder über eine Wahrscheinlichkeitsfunktion oder über die Sequenztabelle gesteuert werden. Über diesen Parameter kann zwischen diesen beiden Betriebsarten umgeschaltet werden.

### Sequence Type

Bei Auswahl des „Modulation Mode = sequence“ kann hier festgelegt werden, ob die gemeinsam nutzbare Sequenztabelle („common“) oder die zylinder-spezifische Sequenztabelle („individual“) verwendet werden soll (siehe auch „Sequenztabellen“ auf Seite 847).

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Start Angle	real32	Ja	Startwinkel der Drehzahlmodulation 0.0 ... 720.0 °KW
Attenuation Coefficient	real32	Ja	Anteilige Wirkung des Modulationsprofils 0.0 ... 1.0
Modulation Profile	uint8	Ja	Auswahl des Modulationsprofils für den Zylinder 0 ... 3
Modulation Mode	uint8	Ja	Wahrscheinlichkeit („probability“) oder Sequenztabelle („sequence“) für die Steuerung der Zündaussetzung
Sequence Type	uint8	Ja	Auswahl: Gemeinsame („common“) oder zylinderindividuelle („individual“) Sequenz.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

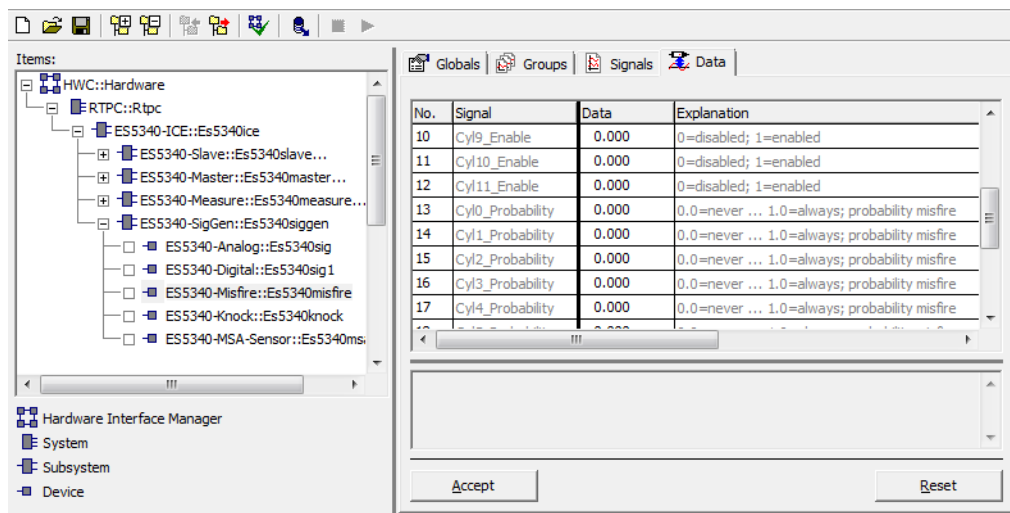
**Tab. 23-23** ES5340-Misfire Device: Konfigurationsparameter der Registerkarte „Signals“

#### 23.21.4 Data (ES5340-Misfire Device)

In der Registerkarte „Data“ werden die RTIO-Signale dargestellt, die zur Steuerung der Zündaussetzeinheit vorhanden sind.

#### Hinweis

*Beachten Sie bitte, dass der Parameter „Served Cylinders“ (siehe „Served Cylinders“ auf Seite 849) aus dem übergeordneten ES5340-SigGen Subsystem korrekt gesetzt ist, da dieser Wert bei der Verarbeitung der Signale verwendet wird!*



**Abb. 23-38** Die Registerkarte „Data“ des ES5340-Misfire Device

#### *Cyl0\_Enable ... Cyl11\_Enable*

Diese zwölf Signale aktivieren die Drehzahlmodulation für den jeweiligen Zylinder.

#### *Cyl0\_Probability ... Cyl11\_Probability*

Über diese Signale wird die Wahrscheinlichkeit für das Auftreten einer Fehlzündung für einen Zylinder gesteuert. Das Signal wird nur berücksichtigt, wenn der entsprechende Zylinder zur Nutzung der Wahrscheinlichkeitsfunktion konfiguriert wurde („Modulation Mode“ auf Seite 858).

#### *Sqnc\_Trigger*

Dieses Signal stellt den Trigger dar, mit dem das Abspielen der Sequenztabellen gestartet wird. Eine steigende Flanke von 0 → 1 auf diesem Signal startet das Abspielen. Eine noch nicht vollständig abgelaufene Sequenz wird durch eine steigende Flanke auf dem Triggersignal abgebrochen und neu gestartet (re-trigger).

Die folgende Tabelle fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Cyl0_Enable ... Cyl11_Enable	uint8	Ein- / Ausschalten der Ausgänge 0: Ausschalten des Ausgangs 1: Einschalten des Ausgangs
Cyl0_Probability .. Cyl11_Probability	uint16	Wahrscheinlichkeit für einen Zündaussetzer Wertebereich: 0 bis 10000 entspricht 0 ... 1.0
Sqnc_Trigger	uint8	Starten bzw. Neustarten des Abspielens der Sequenztabellen Eine Flanke 0 → 1 dient als Trigger
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 23-24** ES5340-Misfire Device: Die RTIO-Signale der Registerkarte „Data“

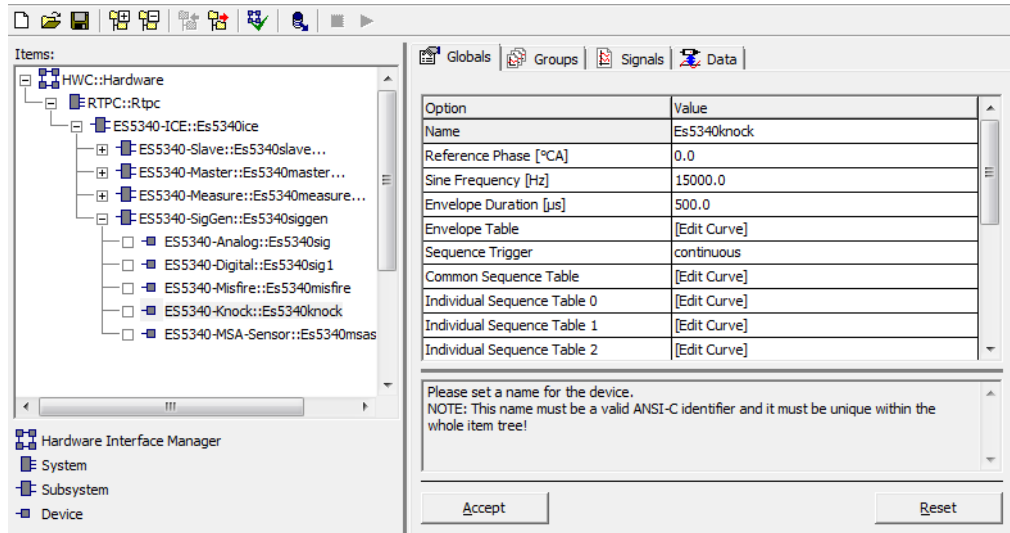


## 23.22 ES5340-Knock Device

Das ES5340-Knock Device dient zur Konfiguration und Ansteuerung des Klopfsignalgenerators auf der ES5340.2-ICE.

### 23.22.1 Globals (ES5340-Knock Device)

Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 23-39** Die Registerkarte „Globals“ des ES5340-Knock Device

#### *Reference Phase*

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Klopfsignalgenerators. Die tatsächliche zylinderspezifische Phase wird aus der Addition dieses Werts mit dem jeweiligen Parameter „Base Angle“ aus der Registerkarte „Signals“ (siehe „Base Angle“ auf Seite 864) und dem Signal „Cyln\_AngleOffset“ (siehe „Cyl0\_AngleOffset ... Cyl11\_AngleOffset“ auf Seite 867) bestimmt.

#### *Sine Frequency*

Dieser Parameter beschreibt die Frequenz des Sinussignals, aus dem das Klopfsignal besteht. Dieses Sinussignal wird mit einer Hüllkurve moduliert.

#### *Envelope Duration*

Dieser Parameter beschreibt die Dauer eines Klopfsignals in µs.

#### *Envelope Table*

Die Form der Hüllkurve wird durch diese Tabelle beschrieben. Die Länge der Hüllkurve wird durch den Parameter „Envelope Duration“ vorgegeben (siehe hierzu „Klopfsignalgenerator“ auf Seite 846).

#### *Sequence Trigger*

Mit diesem Parameter kann zwischen „Single-Shot“ und der kontinuierlichen Triggerung der Sequenztabelle ausgewählt werden.

### Common Sequence Table, Individual Sequence Table 0 ... 11

Es stehen 12 zylinderindividuelle Sequenztabelle zur Steuerung der Erzeugung von Klopfpaketen zur Verfügung. Zusätzlich steht eine gemeinsam nutzbare Sequenztabelle „Common Sequence Table“ zur Verfügung, die es ermöglicht, einen Sequenzablauf einfach auf mehreren Zylindern zu nutzen.

### Cylinder to Sensor Mapping Table 0 ... 3

Der Klopfsignalgenerator verfügt über vier Ausgänge. In jeder der vier „Cylinder to Sensor Mapping“ Zuordnungstabellen wird für den jeweiligen Ausgang definiert, von welchen Zylindern die generierten Klopfsignale ausgehen werden. Damit besteht die Möglichkeit, einen Zylinder auf mehrere interne Ausgänge auszugeben, indem man den entsprechenden Zylinder mehrfach in der Zuordnungstabelle auswählt.

Die Tabellen bestehen jeweils aus bis zu 12 Stützstellen, die die Zylinder 0 bis 11 repräsentieren. Ein Wert von 1.0 an einer Stützstelle entspricht der Zuordnung des jeweiligen Zylinders an den entsprechenden Ausgang. Ein Wert von 0.0 entspricht einer nicht vorhandenen Zuordnung des Zylinders an den Ausgang. Werte zwischen 0.0 und 1.0 geben die Amplitude des auftretenden Klopfsignals an.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Reference Phase	real32	Ja	Grund-Phasenverschiebung in °KW. Wertebereich 0 ... 720 °KW
Sine Frequency	real32	Ja	Sinusfrequenz der Klopfpakete im Bereich 1000 - 20000 Hz
Envelope Duration	real32	Ja	Dauer der Hüllkurve in µs. Wertebereich: 50-12000 µs
Envelope Table	table	Ja	Hüllkurvensignal
Sequence Trigger	uint8	Ja	Auswahl des Triggermodus für die Sequenztabelle. 0: Kontinuierlich 1: Single Shot
Common Sequence Table	table	Ja	Für alle Zylinder gemeinsam nutzbare Sequenztabelle
Individual Sequence Table 0 ... 11	table	Ja	Zylinderindividuelle Sequenztabelle
Cylinder to Sensor Mapping Table 0 ... 3	table	Ja	Tabellen, in denen einem Ausgang die zugehörigen Zylinder zugeordnet werden können. Die Stützstellen entsprechen den 12 Zylindern - ein Wert von größer 0.5 bedeutet, dass der entsprechende Zylinder auf dem Ausgang das Klopfsignal ausgibt.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

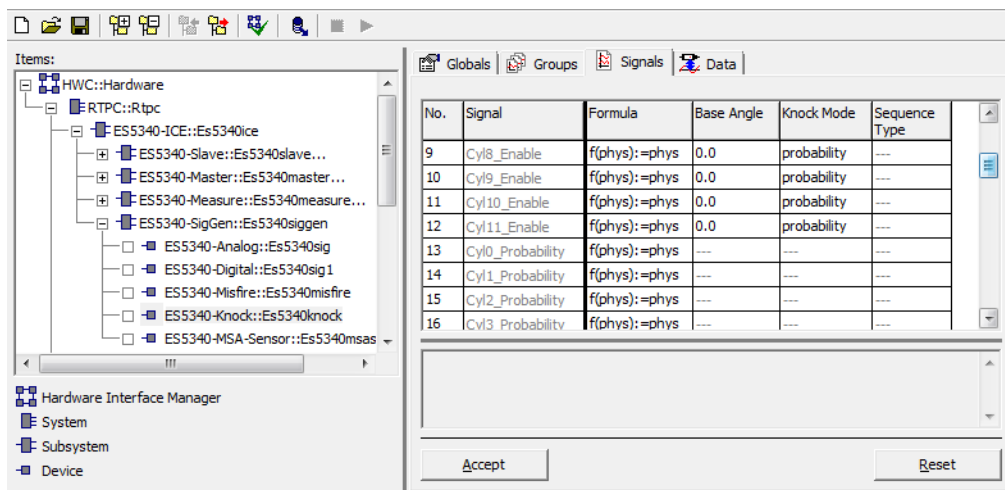
**Tab. 23-25** ES5340-Knock Device: Konfigurationsparameter der Registerkarte „Globals“

### 23.22.2 Groups (ES5340-Knock Device)

Das ES5340-Knock Device besitzt eine Signalgruppe, in der die Signale der Zündaussetzsteuerung verarbeitet werden.

### 23.22.3 Signals (ES5340-Knock Device)

In der Registerkarte „Signals“ wird die Konfiguration des Klopfgenerators durchgeführt. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 23-40** Die Registerkarte „Signals“ des ES5340-Knock Device

#### *Base Angle*

Das Optionsfeld „Base Angle“ dient zusammen mit dem Parameter „Reference Phase“ aus der Registerkarte „Globals“ als Grundwinkel zur Steuerung des Startwinkels eines Klopfpakets. Zu diesem Startwinkel wird noch das zylinderspezifische Signal „Cylx\_AngleOffset“ addiert, das eine Veränderung zur Laufzeit ermöglicht. Zu beachten ist, dass das Signal „Cylx\_AngleOffset“ nur über einen Winkelbereich von  $\pm 127$  °KW variiert werden kann.

#### *Knock Mode*

Das Auftreten von Klopfereignissen in einer Periode von 720 °KW (bzw. 360 ° beim Zweitaktmotor) kann entweder über eine Wahrscheinlichkeitsfunktion oder über die Sequenztabellen gesteuert werden. Über diesen Parameter kann zwischen diesen beiden Betriebsarten umgeschaltet werden.

Sequence Type

Bei Auswahl des „Modulation Mode = sequence“ kann hier ausgewählt werden, ob die gemeinsam nutzbare Sequenztabelle „common“ oder die zylinderindividuelle Sequenztabelle („individual“) verwendet werden soll.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Base Angle	real32	Ja	Basisausgabewinkel für Klopfpakete 0.0 ... 720.0 °KW
Knock Mode	uint8	Ja	Wahrscheinlichkeit („probability“) oder Sequenztabelle („sequence“) für die Ausgabe von Klopfpaketen
Sequence Type	uint8	Ja	Auswahl: Gemeinsame („common“) oder zylinderindividuelle („individual“) Sequenz

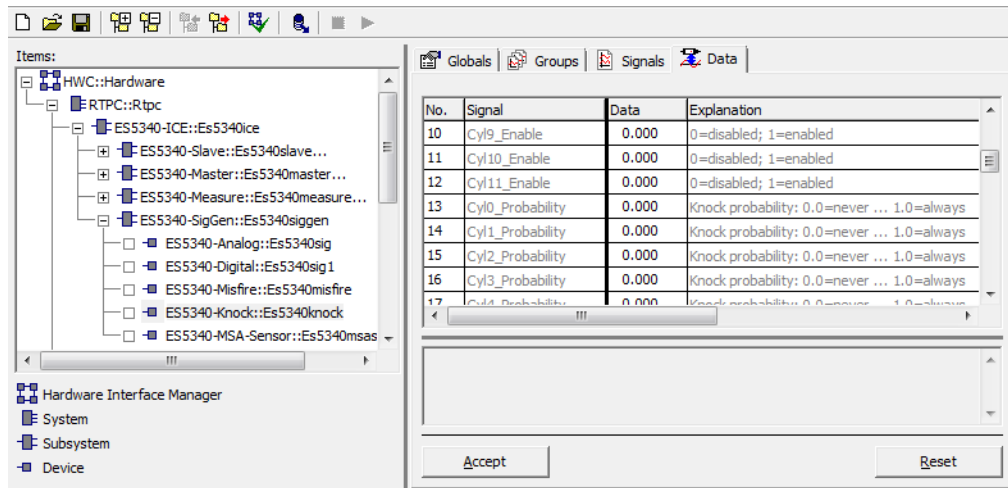
\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-26** ES5340-Knock Device: Konfigurationsparameter der Registerkarte „Signals“

23.22.4 Data (ES5340-Knock Device)

In der vorhandenen Signalgruppe werden RTIO-Signale dargestellt, die zur Steuerung des Klopfsignalgenerators vorhanden sind.

**Hinweis**  
*Beachten Sie bitte, dass der Parameter „Served Cylinders“ im übergeordneten ES5340-SigGen Subsystem („Served Cylinders“ auf Seite 849) korrekt gesetzt ist, da dieser Wert bei der Verarbeitung der Signale verwendet wird!*



**Abb. 23-41** ES5340-Knock Device: Konfigurationsparameter der Registerkarte „Data“

---

*Cyl0\_Enable ... Cyl11\_Enable*

---

Diese zwölf Signale aktivieren die Generierung von Klopfpaketen für unkontrollierte Verbrennung für die jeweiligen Zylinder.

---

*Cyl0\_Probability ... Cyl11\_Probability*

---

Über diese Signale wird die Wahrscheinlichkeit für das Auftreten einer klopfenden Verbrennung für einen Zylinder gesteuert. Das Signal wird nur berücksichtigt, wenn der entsprechende Zylinder zur Nutzung der Wahrscheinlichkeitsfunktion konfiguriert wurde.

---

*Sqnc\_Trigger*

---

Dieses Signal bildet den Trigger, mit dem das Abspielen der Sequenztabelle gestartet werden kann. Eine steigende Flanke von 0 → 1 auf diesem Signal startet das Abspielen. Eine noch nicht vollständig abgelaufene Sequenz wird durch eine steigende Flanke auf dem Triggersignal abgebrochen und neu gestartet (re-trigger).

---

*NoiseAmpl*

---

Dieses Signal steuert eine stochastische Variation der Amplituden der Klopfpakete. Dabei findet bei einem Wert von 0.0 keine Variation der Amplitude statt, bei einem Wert von 0.1 kann die Amplitude um 10% der maximalen Ausgangsspannung schwanken.

Diese Variation wirkt auf die Ausgabe aller Klopfpakete (klopfend und nicht klopfend).

---

*NoKnockAmpl*

---

Bei einer Verbrennung ohne aufgetretenes Klopfen können dennoch Klopfpakete ausgegeben werden. Die Amplitude dieser Klopfpakete werden mit diesem Signal beschrieben. Typischerweise wird die Amplitude von „NoKnockAmpl“ kleiner als die Amplituden bei einem aufgetretenen Klopfereignis gewählt.

---

*Cyl0\_KnockAmpl ... Cyl11\_KnockAmpl*

---

Über dieser 12 Signale können die zylinderspezifischen Amplituden für eine klopfende Verbrennung angegeben werden.

Cyl0\_AngleOffset ... Cyl11\_AngleOffset

Der Ausgabewinkel der Klopfpakete kann durch dieses Signal zylinderindividuell um  $\pm 127$  °KW in Schritten von 1 °KW variiert werden. Als Basiswinkel dient die Summe von „Reference Phase“ (Registerkarte „Globals“) und des jeweiligen „Base Angles“ (Registerkarte „Signals“).

RTIO-Signal	Datentyp*	Bemerkung/Wertebereich
Cyl0_Enable ... Cyl11_Enable	uint8	Steuerung der Klopfsignale für die einzelnen Zylinder 0: Klopfpakete für Klopfen möglich 1: Klopfpakete für Klopfen nicht möglich
Cyl0_Probability .. Cyl11_Probability	uint16	0.0 bis 1.0
Sqnc_Trigger	uint8	Starten bzw. Neustarten des Abspielens der Sequenztabelle Eine Flanke 0 → 1 dient als Trigger
NoiseAmpl	uint16	0.0 bis 1.0
NoKnockAmpl	uint16	0.0 bis 1.0
Cyl0_KnockAmpl ... Cyl11_KnockAmpl	uint16	0.0 bis 1.0
Cyl0_AngleOffset ... Cyl11_AngleOffset	sint8	Variation des Klopfwinkels Wertebereich -127 bis +127 (Auflösung 1 °KW)
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 23-27** ES5340-Knock Device: Die RTIO-Signale der Registerkarte „Data“

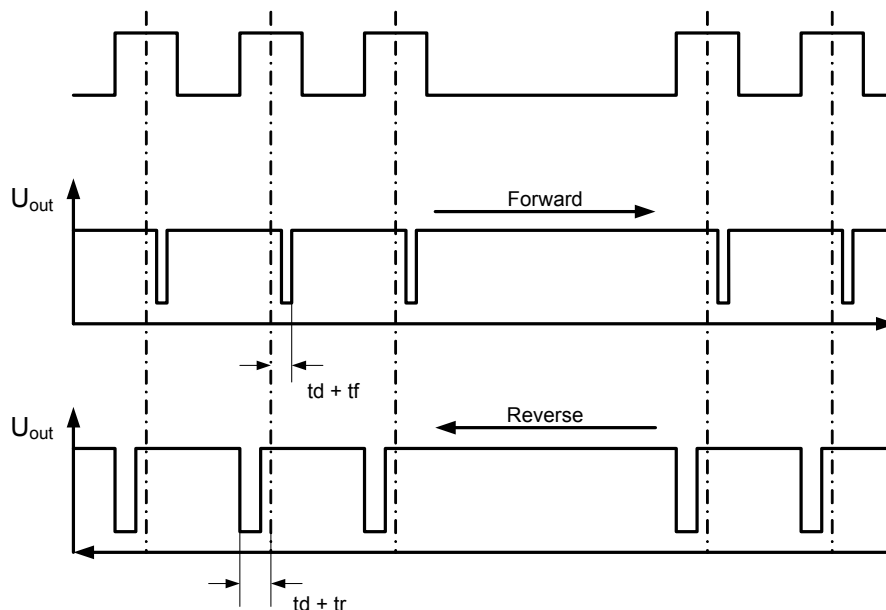
## 23.23 ES5340-MSA-Sensor Device

Das ES5340-MSA-Sensor Device dient zur Simulation eines MSA-Kurbelwellensensors mit Drehrichtungserkennung.

### 23.23.1 Grundsätzliche Funktionsweise

Die MSA-Sensor-Simulation ist in der Lage, dieselben Wellenverläufe des Zahnsignals wie die bisher verwendeten (Hall-)Sensoren zu verwenden. Zur Darstellung eines MSA-Sensors ist jedoch die Kenntnis der Winkelposition der Mitte eines jeden Zahnes des Kurbelwellenrades erforderlich. Für jeden Zahn wird bei der Zahnmitte ein Impuls fester Länge ausgegeben – im realen Sensor werden die Zahnmitten elektronisch ermittelt.

In der Konfigurationsphase werden diese Winkelpositionen aus dem Verlauf der Wellenform berechnet und im Wellenformspeicher abgelegt. Da die Wellenform mittels des Signals "Waveform" in Echtzeit aus einer der vorgegebenen 16 Wellenformen ausgewählt werden kann, müssen alle Wellenformen diesem Verfahren unterzogen werden.



**Abb. 23-42** Kurbelwellenzähne und Sensorsignale bei einem MSA-Sensor

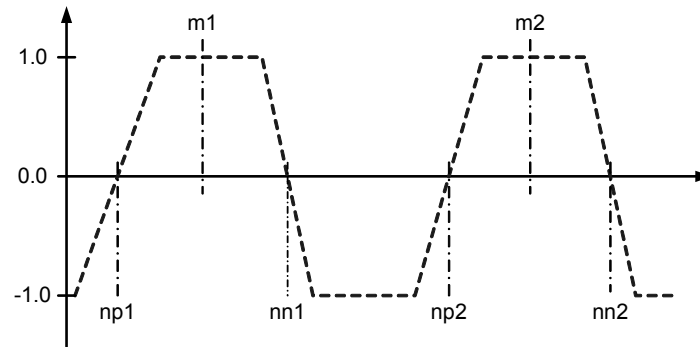
### 23.23.2 Zahnmittenberechnung

Dazu wird zunächst bei allen bereits zwischen -1.0 und 1.0 normierten Wellenformverläufen jeweils die Positionen der negativ und der positiv verlaufenden Nulldurchgänge ermittelt.

Bei positiver Logik definiert sich ein Zahn als Bereich zwischen einem positiven und dem folgenden negativen Nulldurchgang (siehe Abb. 23-43 auf Seite 869). Die Zahnmitte ist dementsprechend die arithmetische Mittel der beiden Positionen (bei negativer Logik entsprechend umgekehrt):



Es seien np1, np2 die von unten nach oben schneidenden (positive) Nulldurchgänge und nn1, nn2 die von oben nach unten schneidenden (negative) Nulldurchgänge. Dann ergeben sich die Zahnmittenpositionen m1, m2 zu  $m1 = (np1 + nn1) / 2$  und  $m2 = (np2 + nn2) / 2$ .

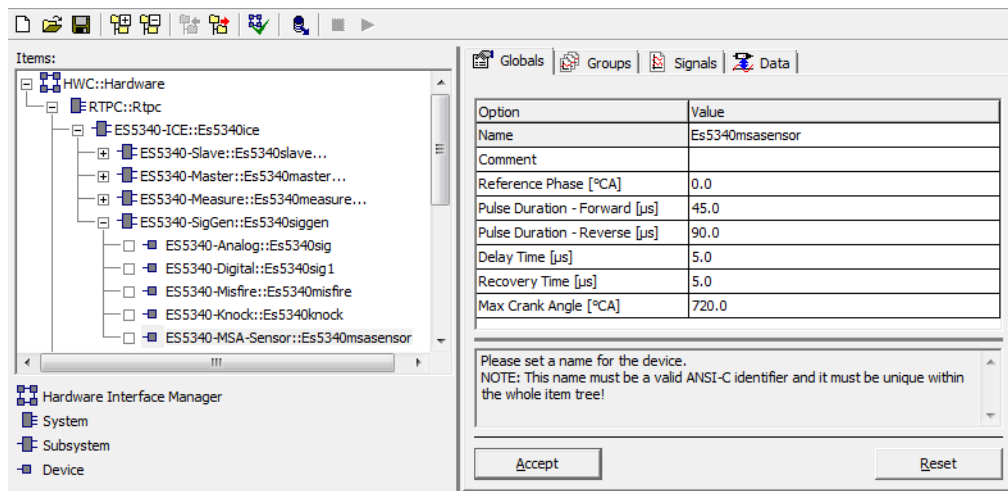


**Abb. 23-43** Mittenpositionsberechnung (bei nicht ganz symmetrischem Zahnverlauf und positiver Logik)

Die Einstellung der Logik erfolgt für alle Kurvenformen gemeinsam in der Registerkarte „Globals“ des ES5340-SigGen Devices (siehe „MSA Sensors: Logic of Input Waveform“ auf Seite 850).

### 23.23.3 Globals (ES5340-MSA-Sensor Device)

Das ES5340-MSA-Sensor Device dient zur Konfiguration und Ansteuerung eines der Signalgeneratoren auf der ES5340.2-ICE. Die nachfolgende Abbildung zeigt die RTIO-Parameter der Registerkarte „Globals“.



**Abb. 23-44** Die Registerkarte „Globals“ des ES5340-MSA-Sensor Device

Die Bedeutung der Optionen wird im Folgenden beschrieben.

#### Comment

Ein beliebiger Kommentar für diesen Signalgenerator.

#### Hinweis

Das Kommentarfeld erlaubt keine Eingabe von Leer- oder Sonderzeichen.

Reference Phase [° CA]

Dieser Parameter beschreibt die Grund-Phasenverschiebung des Signalgenerators. Die tatsächliche Phase wird aus der Addition dieses Werts zu dem Signal „PhaseRef“ ermittelt.

Pulse Duration - Forward [µs]

Dieser Parameter definiert die Pulsdauer eines Zahnes bei positiver Drehrichtung (vorwärts) (tf in Abb. 23-42 auf Seite 868).

Pulse Duration - Reverse [µs]

Dieser Parameter definiert die Pulsdauer eines Zahnes bei negativer Drehrichtung (rückwärts) (tr in Abb. 23-42 auf Seite 868).

Delay Time [µs]

Dieser Parameter definiert die Länge der Verzögerungszeit (td in Abb. 23-42 auf Seite 868).

Recovery Time [µs]

Dieser Parameter definiert die Länge eines (in bestimmten Situationen der Drehrichtungsumkehr) erzwungenen Impulses.

Die nachfolgende Tabelle fasst die Eigenschaften der einzelnen Parameter zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
Comment	Identifizier	Nein	Benutzerspezifischer Kommentar
Reference Phase	real32	Ja	Grund-Phasenverschiebung in ° KW, Wertebereich 0...720 °KW
Pulse Duration - Forward	real32	Ja	Pulsdauer tf bei Vorwärtsrotation, Wertebereich 10.0...1000.0 µs
Pulse Duration - Reverse	real32	Ja	Pulsdauer tr bei Rückwärtsrotation, Wertebereich 10.0...1000.0 µs
Delay Time	real32	Ja	Verzögerungszeit td gemäß Sensorspezifikation Wertebereich 1.0...100.0 µs
Recovery Time	real32	Ja	Pulsdauer tv gemäß Sensorspezifikation Wertebereich 1.0...100.0 µs

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

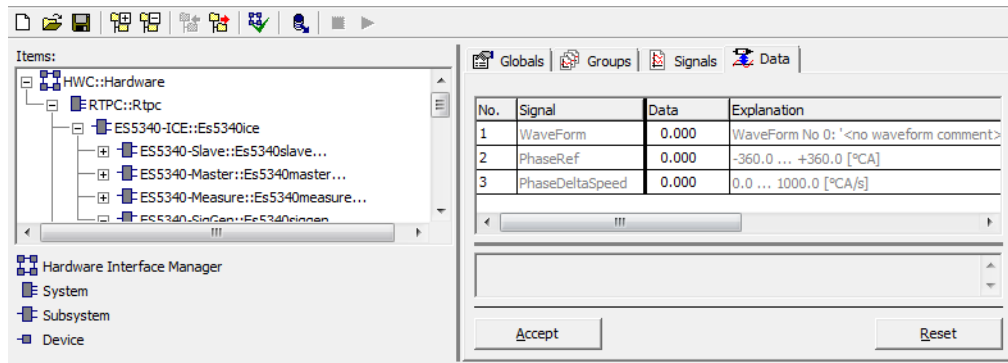
**Tab. 23-28** ES5340-MSA-Sensor: Konfigurationsparameter der Registerkarte „Globals“

#### 23.23.4 Groups (ES5340-MSA-Sensor Device)

Das ES5340-MSA-Sensor Device besitzt eine Signalgruppe, mit der der MSA-Generator gesteuert wird.

23.23.5 Data (ES5340-MSA-Sensor Device)

In der Registerkarte „Data“ werden die Signale zur Steuerung des Signalgenerators aufgelistet.



**Abb. 23-45** Die Registerkarte „Data“ des ES5340-MSA-Sensor Device

Die Bedeutung der Signale wird im Folgenden beschrieben.

*WaveForm*

Über das Signal „WaveForm“ wird die Wellenform ausgewählt, die vom Signalgenerator ausgetaktet werden soll. Es sind dabei Werte von 0 bis 15 möglich, die den zentralen Wellenformspeichern im ES5340-SigGen Device entsprechen.

*PhaseRef*

Dieses Signal beschreibt eine Phasenverschiebung, die zum globalen Parameter „ReferencePhase“ addiert wird. Damit kann eine modellgesteuerte Phasenverschiebung zur Laufzeit generiert werden.

Zu beachten ist, dass eine Änderung der Phasenverschiebung mit einer endlichen Geschwindigkeit (gesteuert über den Parameter „PhaseDeltaSpeed“) durchgeführt wird – daher muss das Signal „PhaseDeltaSpeed“ für die meisten Anwendung einen Wert > 0 aufweisen.

*PhaseDeltaSpeed*

Die Änderungsgeschwindigkeit der Phasenverschiebung wird mit dem Signal „PhaseDeltaSpeed“ gesteuert, da eine Änderung des Werts von „PhaseRef“ nur mit endlicher Geschwindigkeit vonstatten geht. Der Wertebereich reicht von 0 (d.h. es findet keine Verstellung der Phase statt) bis zu 1000° KW/s.

Parameter bzw. Optionsfeld	Datentyp*	Online editierbar	Bemerkung/Wertebereich
WaveForm	uint8	Ja	Auswahl der Wellenform 0-11
PhaseRef	sint16	Ja	Additive Phasenverschiebung, Wertebereich: 32768...32767 (entspricht -360...+360 [° KW])
PhaseDeltaSpeed	uint16	Ja	Phasenänderungsgeschwindigkeit 0...1000 [° KW/s]

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 23-29** ES5340-MSA-Sensor: Signale der Registerkarte „Data“

### 23.23.6 Fehlerbehandlung

Die Ausgabe einer Fehlermeldung bei ungeeigneten Wellenformverläufen erfolgt erst dann, wenn eine solche Wellenform tatsächlich als Basis für die Ausgabe eines MSA-Sensors benutzt wird.

Eine solche Fehlermeldung hat folgende Form:

```
"Waveform <x> not suitable for MSA sensor implemented
  by SigGen #<y> because <error> No signal generated"
```

Dabei bedeutet <x> die Nummer der ungeeigneten Wellenform 0-15 und #<y> die Instanznummer des verwendeten Signalgenerators 0...7.

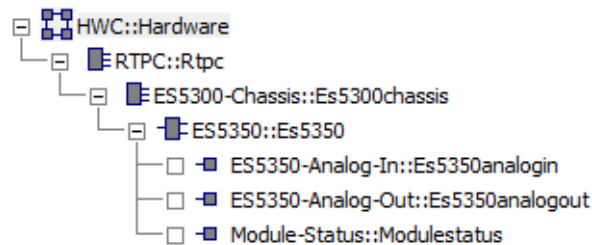
Eine Beschreibung der Fehlermeldungen <error> finden Sie in der folgenden Tabelle:

Fehlermeldung <error>	Beschreibung
...too many zero crossings (> 2048)...	Es wurden zu viele Nulldurchgänge detektiert, um eine sinnvolle Ausgabe für einen MSA-Sensor zu erzeugen.
...number of positive and negative zero crossings do not match...	Es wurde eine unterschiedliche Anzahl von positiven und negativen Nulldurchgängen ermittelt. Kurvenform ist nicht zur Darstellung eines MSA-Sensors geeignet.
...too few zero crossings (< 2)...	Es wurden zu wenige Nulldurchgänge detektiert, um eine sinnvolle Ausgabe für einen MSA-Sensor darzustellen.
...minimum distance between some tooth centers < 0.4 °CA...	Mindestens zwei der errechneten Zahnmittenspositionen liegt zu dicht nebeneinander, um von der Ausgabereinheit sicher verarbeitet werden zu können. Es ist Mindestabstand der Stützstellen von etwa 0,4 °KW erforderlich.

**Tab. 23-30** Liste der möglichen Fehlermeldungen bei ungeeigneten Wellenformen

## 24 ES5350.1 Analog Board

Das ES5350.1 Analog Board dient zur Erzeugung und Messung analoger Signale. Im RTIO-Editor wird das ES5350.1 Analog Board durch Auswahl des ES5350 Subsystems eingebunden.

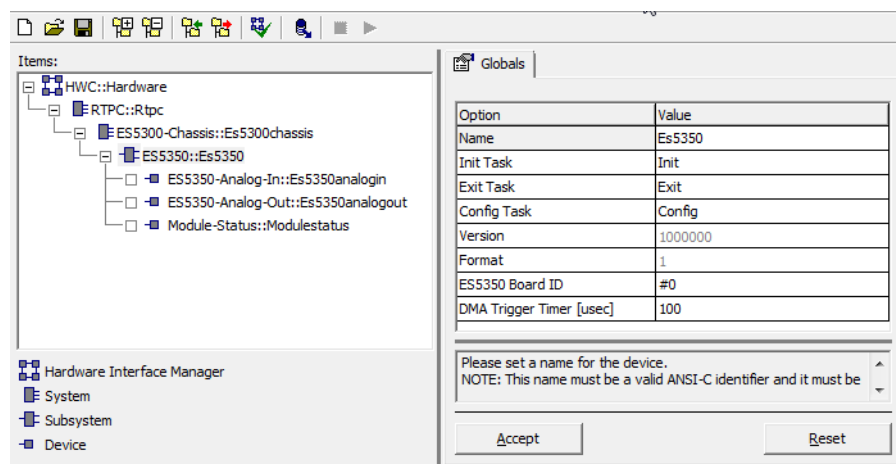


**Abb. 24-1** RTIO-Hardwarebeschreibung mit ES5350.1 Analog Board

### 24.1 ES5350 Subsystem

#### 24.1.1 Globals (ES5350 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES5350 Subsystems beschrieben.



**Abb. 24-2** Die Registerkarte „Globals“ des ES5350 Subsystems

#### *ES5350 Board ID*

Im RTIO-Editor können bis zu 16 ES5350.1-Boards pro ES5300 Baugruppen-träger eingebunden werden. Diese Option dient zur Identifizierung des ES5350.1 Analog Boards – nummeriert von links (Steckplatz 0) nach rechts.

Dieser RTIO-Parameter ist während der Laufzeit des Modells auf dem Experimentaltarget nicht editierbar.

Wenn im Menü des Editors die Option **View → Show All** gewählt wurde, wird die folgende Option sichtbar:

*DMA Trigger Timer [μsec]*

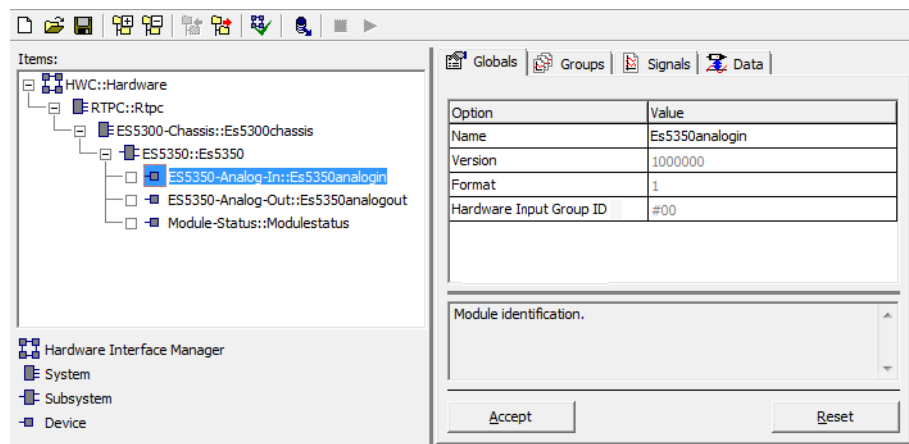
Hier kann der zeitliche Abstand für die Aktivierung zweier asynchroner DMA-Übertragungen gewählt werden – die aktuellen Spannungen an den zehn analogen Eingängen werden direkt in den PC-Hauptspeicher geschrieben.

- Defaultwert: 100 μs
- Einstellbereich: 5 μs ... 1 s

## 24.2 ES5350-Analog-In Device

### 24.2.1 Globals (ES5350-Analog-In Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



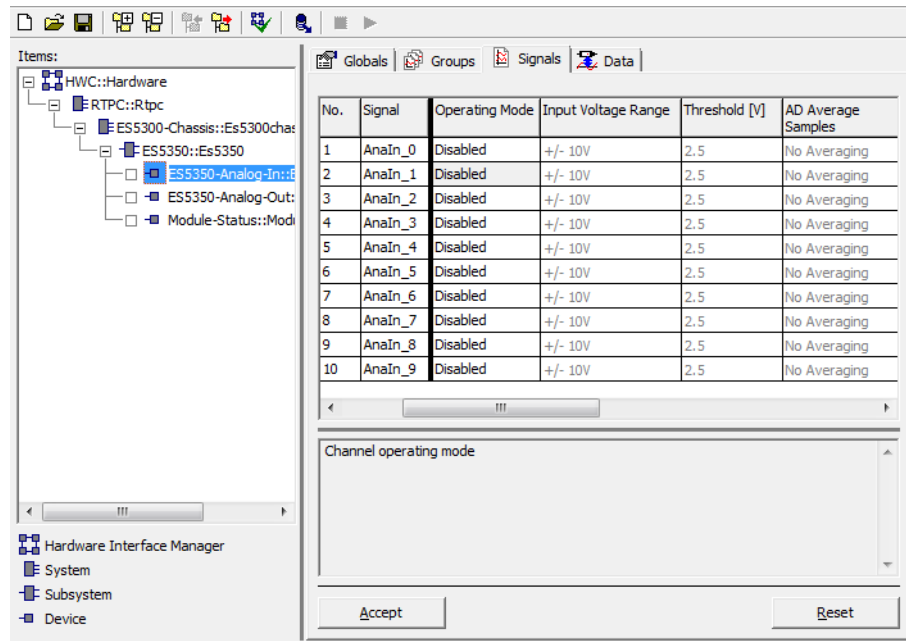
**Abb. 24-3** Die Registerkarte „Globals“ des ES5350-Analog-In Device

### 24.2.2 Groups (ES5350-Analog-In Device)

Das ES5350-Analog-In Device besitzt eine Receive-Gruppe „Analn“ für die Messwerte der analogen Eingänge.

24.2.3 Signals (ES5350-Analog-In Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 24-4** Die Registerkarte „Signals“ des ES5350-Analog-In Device

Signalname	Anmerkungen
AnaIn_x	Analoge Spannung am Eingang #x
(x = 0...9)	

**Tab. 24-1** Die Signale des ES5350-Analog-In Device

Für die Eingangssignale gibt es folgende Parameter:

Parameter bzw. Optionsfeld	Online editierbar	Bemerkung/Wertebereich
Operating Mode	Ja	- <b>Disabled</b> Eingang ist deaktiviert - <b>Analog</b> Spannungsmessung/Wertebereich siehe „Input Voltage Range“ - <b>Comparator</b> Eingang arbeitet als Komparator [0.0, 1.0] mit Hysterese (Werte siehe „Input Voltage Range“)
Input Voltage Range	Ja	Eingangsspannungsbereich (Hysterese) - [-1 V...+1 V] ( $\pm 0.05$ V) - [-10 V...+10 V] ( $\pm 0.5$ V) - [-60 V...+60 V] ( $\pm 0.5$ V)
Threshold [V]	Ja	Spannungsschwelle des Komparators
AD Average Samples	Ja	Anzahl der Samples zur Mittelwertbildung

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

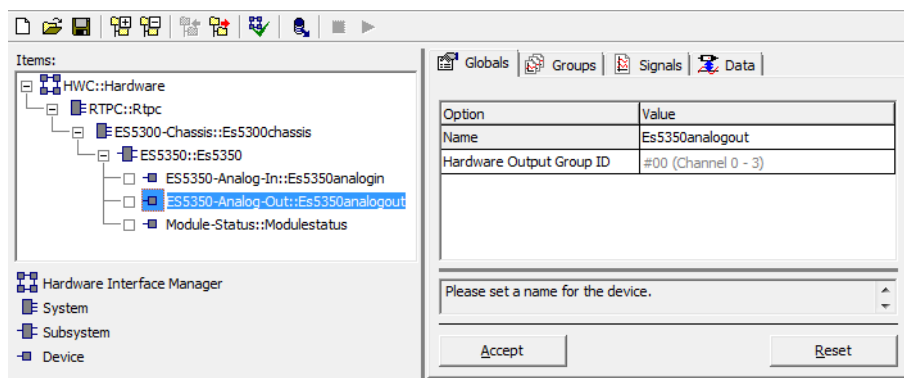
**Tab. 24-2** ES5350-Analog-In Device: Konfigurationsparameter der Registerkarte „Signals“

### 24.3 ES5350-Analog-Out Device

Unter einem ES5350 Subsystem können bis zu fünf Analog-Out Devices eingefügt werden.

#### 24.3.1 Globals (ES5350-Analog-Out Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 24-5** Die Registerkarte „Globals“ des ES5350-Analog-Out Device

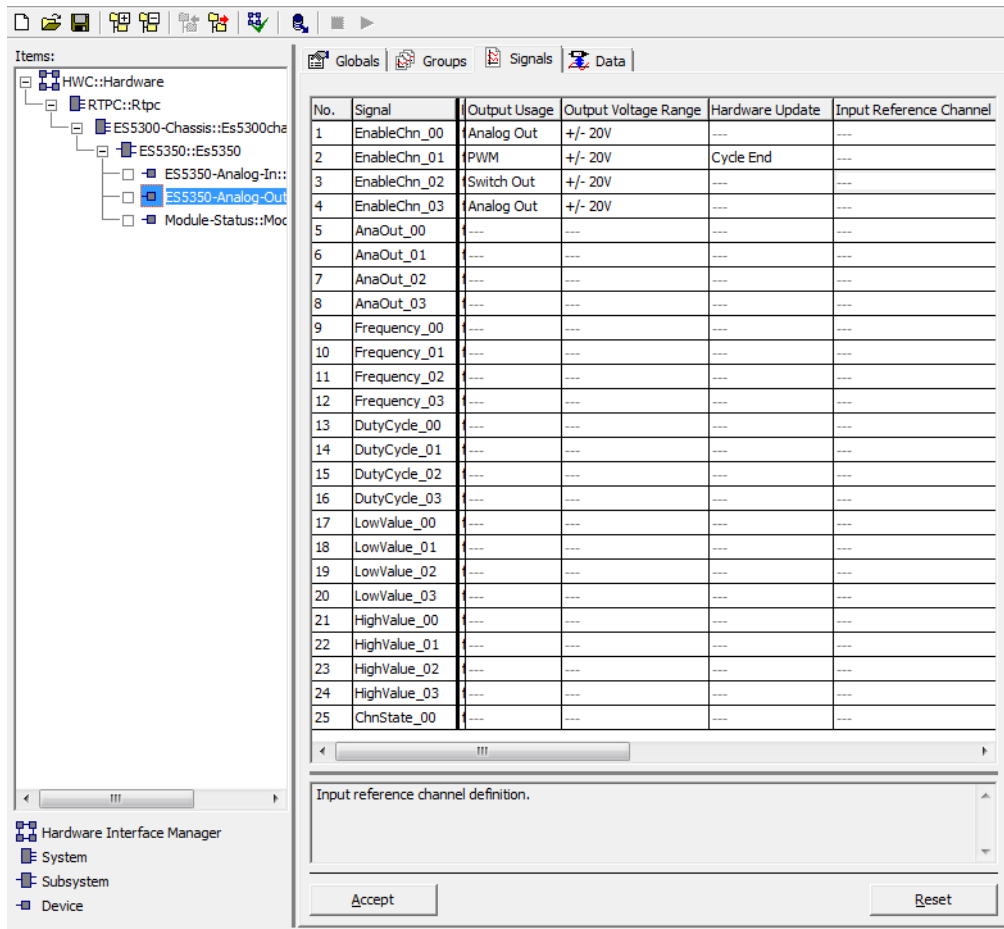


24.3.2 Groups (ES5350-Analog-Out Device)

Das ES5350-Analog-Out Device besitzt eine Send-Gruppe „AnaOut“ für die analogen Ausgangssignale und eine Receive-Gruppe „ChnState“ für Statusinformationen.

24.3.3 Signals (ES5350-Analog-Out Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 24-6** Die Registerkarte „Signals“ des ES5350-Analog-Out Device

Signalname	Anmerkungen
EnableChn_xx	0.0 = Kanal deaktiviert, sonst Kanal aktiviert
AnaOut_xx (x = 00...03)	Je nach Einstellung für Parameter „Output Usage“ in Tab. 24-4 auf Seite 879: - Analog Out: Ausgangsspannung [-20.0..+20.0 V] oder (wenn „Output Voltage Range“ = „Input Dep“) -1.0..1.0 - PWM: ohne Bedeutung - Switch Out: 0.0 (LowValue_xx)..1.0 (HighValue_xx)
Frequency_xx	PWM-Frequenz Wertebereich: 0.0..1000.0 Hz
DutyCycle_xx	Tastverhältnis des PWM-Signals Wertebereich: 0.0...1.0
LowValue_xx	Switch Out: Wenn bei „Output Voltage Range“ (siehe Tab. 24-4 auf Seite 879) $\pm 20$ V gewählt wurde: Unterer Wert der Ausgangsspannung [V]. Wenn „Output Voltage Range“ = „Input Dep“: -1.0...1.0 des Eingangswertes.
HighValue_xx	Switch Out: Wenn bei „Output Voltage Range“ (siehe Tab. 24-4 auf Seite 879) $\pm 20$ V gewählt wurde: Oberer Wert der Ausgangsspannung [V]. Wenn „Output Voltage Range“ = „Input Dep“: -1.0...1.0 des Eingangswertes.
ChnState_xx	Übertemperatur: Bits 0 - 3 (kanalselektiv, Bit 0 entspricht Kanal 0 usw.) 0 = Keine Übertemperatur -1 = Übertemperatur Überlast: Bit 4 (gilt für die gesamte Ausgangsgruppe) 0 = Keine Überlast -1 = Überlast

**Tab. 24-3** Die Signale des ES5350-Analog-Out Device

Für die Signale gibt es folgende Parameter:

Parameter bzw. Optionsfeld	Online editierbar	Bemerkung/Wertebereich
Output Usage	Ja	Betriebsart des Kanals: <ul style="list-style-type: none"> <li>- <b>Analog Out</b> Kanal gibt analoge Signale aus</li> <li>- <b>PWM</b> Kanal gibt PWM-Signale aus</li> <li>- <b>Switch Out</b> Ausgangsspannung wechselt zwischen „LowValue_xx“ und „HighValue xx“</li> </ul>
Output Voltage Range	Ja	Spannungsbereich: <ul style="list-style-type: none"> <li>- <b>[-20...+20 V]</b></li> <li>- <b>Input Dependent</b> - erfordert Definition eines „Input Reference Channel“ (s.u.)</li> </ul>
Hardware Update	Ja	Definiert, wann (bei PWM-Signalen) Änderungen von Frequenz oder Tastverhältnis übernommen werden. <ul style="list-style-type: none"> <li>- <b>Immediate</b> Änderungen werden sofort übernommen.</li> <li>- <b>Cycle End</b> Änderungen werden erst nach dem Ende der aktuellen Periode übernommen.</li> </ul>
Input Reference Channel	Ja	Referenz-Eingangskanal (#0..#9), wenn bei „Output Voltage Range“ der Wert „Input Dep“ gewählt wurde.

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 24-4** ES5350-Analog-Out Device: Konfigurationsparameter der Registerkarte „Signals“

## 24.4 Module-Status Device

Siehe „Module Status“ auf Seite 1005.



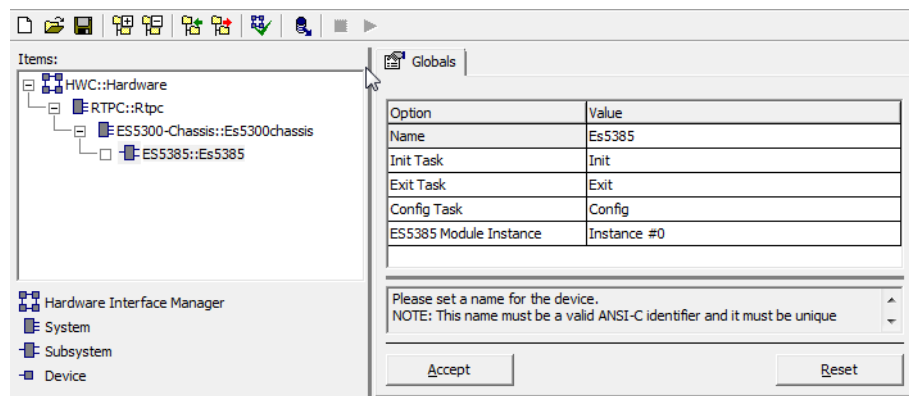
## 25 ES5385.1 Carrier Board for Resistor Cascade

Das ES5385.1 Carrier Board for Resistor Cascade dient im LABCAR-Umfeld als Widerstandskarte zur Simulation von Temperatur- bzw. Lambda-Sensoren und wird in das ES5300.1 Housing integriert.

### 25.1 ES5385Device

#### 25.1.1 Globals (ES5385 Device)

Die Registerkarte „Globals“ eines ES5385 Device ist in Abb. 25-1 dargestellt.



**Abb. 25-1** Die Registerkarte „Globals“ des ES5385 Device

#### Name

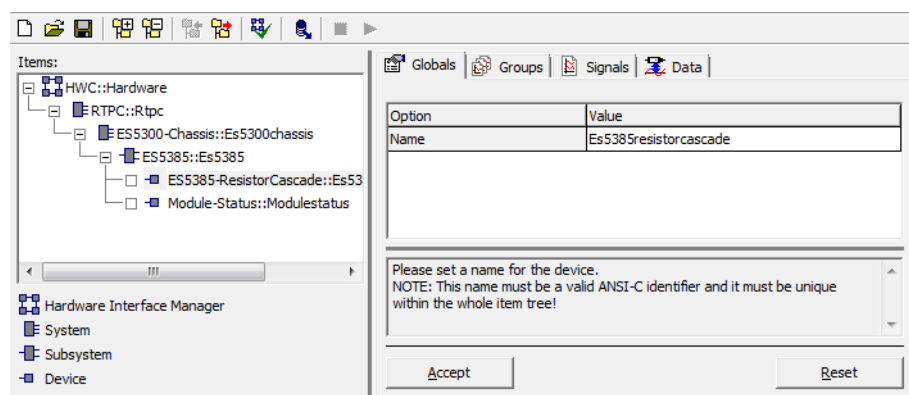
Der C-Name des Items – dieser wird als Prefix für alle globalen Bezeichner verwendet.

#### ES5385 Module Instance

Die Modul-Instanz.

### 25.2 ES5385-ResistorCascade Device

#### 25.2.1 Globals (ES5385-ResistorCascade Device)



**Abb. 25-2** Die Registerkarte „Globals“ des ES5385-ResistorCascade Device

### Name

Der C-Name des Items – dieser wird als Prefix für alle globalen Bezeichner verwendet.

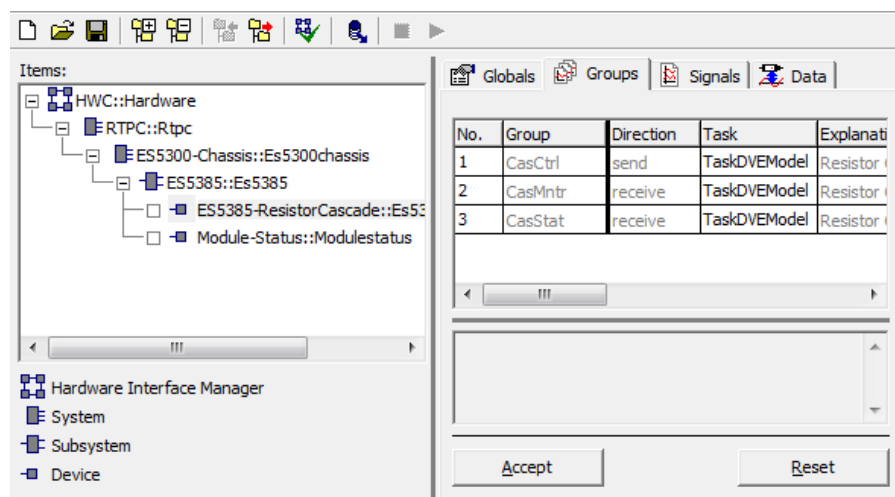
### 25.2.2 Groups (ES5385-ResistorCascade Device:Groups)

Das ES5385-ResistorCascade Device besitzt drei Signalgruppen, die vom Experimentaltarget zum ES5385.1 Carrier Board for Resistor Cascade bzw. von der ES5385.1 zum Experimentaltarget übertragen werden (siehe Abb. 25-3 auf Seite 882).

- Die **Signalgruppe „CasCtrl“** transportiert die vom Anwender spezifizierten und von der RTIO aufbereiteten Widerstandswerte der Kaskaden zur ES5385.1. Die vom Anwender spezifizierten Widerstandswerte werden unter Berücksichtigung der Kalibrierung und Monotonie mit Hilfe eines Näherungsverfahrens zum real möglichen Widerstandswert akkumuliert. Erst danach erfolgt die Übertragung zur Hardware.
- Die **Signalgruppe „CasMntr“** stellt die vom RTIO Low-Level-Treiber gebildeten realen Widerstandswerte für den Anwender dar. Die Aktivierungsperiode dieser Signalgruppe sollte der Aktivierungsperiode der Signalgruppe „CasCtrl“ entsprechen.
- Die **Signalgruppe „CasStat“** transportiert aktuelle Zustandsinformationen der unabhängigen Widerstandskaskaden zur RTIO. Dadurch lassen sich die einzelnen Fehlerzustände der Hardware über die RTIO darstellen bzw. überwachen.

Jeder Signalgruppe ist eine Task des Echtzeit-Betriebssystems zuzuordnen. Man wird üblicherweise eine Task mit periodischer Aktivierung wählen. Die Aktivierungsperiode richtet sich nach der Dynamik der zu erzeugenden Widerstandswerte bzw. der zu überwachenden Statusinformationen.

Abb. 25-3 zeigt die Registerkarte „Groups“ eines ES5385-ResistorCascade Device.



**Abb. 25-3** Die Registerkarte „Groups“ des ES5385-ResistorCascade Device

*RTIO-Signale der Signalgruppe „CasCtrl“*

Die Signalgruppe „CasCtrl“ umfasst 12 RTIO-Signale „Ctrl\_R\_0“ bis „Ctrl\_R\_11“, die vom Datentyp „uint32“ sind. Wird die Widerstandskaskade x (x = 0...11) dahingehend konfiguriert, dass sie von der RTIO aus gesteuert wird, so definiert der Wert des RTIO-Signals "Ctrl\_R\_x" den vom Anwender spezifizierten Soll-Widerstandswert der zugehörigen Kaskade in Ohm.

Wird die Widerstandskaskade nicht von der RTIO gesteuert, so hat das Signal keine Bedeutung.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Ctrl_R_0 bis Ctrl_R_11	uint32	Vom Anwender definierter Widerstandswert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 25-1** ES5385-ResistorCascade Device: Die RTIO-Signale der Signalgruppe „CasCtrl“

*RTIO-Signale der Signalgruppe „CasMntr“*

Die Signalgruppe „CasMntr“ umfasst 12 RTIO-Signale „Mntr\_R\_0“ bis „Mntr\_R\_11“, die vom Datentyp „real32“ sind. Sie wird dazu verwendet, um den zur Hardware übertragenen Ist-Widerstandswert einer Kaskade darzustellen. Jeder vom Anwender definierte Widerstandswert muss unter Berücksichtigung der Kalibrierungswerte und Monotonie mit Hilfe eines Näherungsverfahrens gebildet werden.

Tab. 25-2 fasst die Eigenschaften der RTIO-Signale zusammen.

RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Mntr_R_0 bis Mntr_R_11	real32	Tatsächlicher zur Hardware übertragener Widerstandswert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

**Tab. 25-2** ES5385-ResistorCascade Device: Die RTIO-Signale der Signalgruppe „CasMntr“

*RTIO-Signale der Signalgruppe „CasStat“*

Die Signalgruppe „CasStat“ umfasst 12 RTIO-Signale „Stat\_C\_0“ bis „Stat\_C\_11“, die vom Datentyp „uint16“ sind. Sie wird dazu verwendet, um den aktuellen Zustand der Hardwarekanäle darzustellen.

Tab. 25-3 fasst die Eigenschaften der RTIO-Signale zusammen.

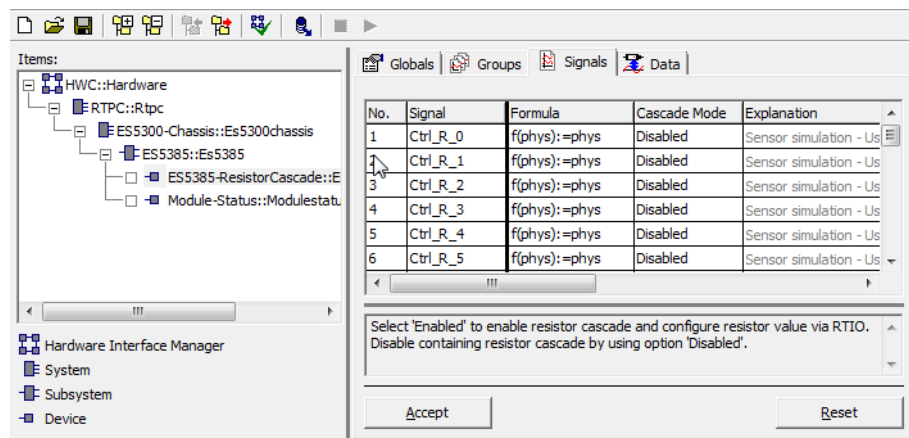
RTIO-Signal	Datentyp*	Bemerkung / Wertebereich
Stat_C_0 bis Stat_C_11	uint16	Zustand der Widerstandskaskade 0: kein Fehler 1: Überstrom wurde erfasst
* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet		

**Tab. 25-3** ES5385-ResistorCascade Device: Die RTIO-Signale der Signalgruppe „CasStat“

### 25.2.3 Signals (ES5385-ResistorCascade Device)

In der Registerkarte „Signals“ wird die Konfiguration der Widerstandskaskaden einer ES5385.1 durchgeführt.

Abb. 25-4 zeigt die RTIO-Parameter der Registerkarte „Signals“. Alle Parameter sind online (d.h. während der Laufzeit des Modells auf dem Experimentaltarget) editierbar.



**Abb. 25-4** Die Registerkarte „Signals“ des ES5385-ResistorCascade Device  
*Cascade Mode*

Dieses Optionsfeld dient zur Konfiguration der Widerstandskaskade. Mit „Disabled“ wird die Kaskade x (x = 0...11) deaktiviert, d.h. alle Relais sind geöffnet. In diesem Zustand können keine Widerstandswerte eingegeben und zur Hardware übermittelt werden.

„Enabled“ aktiviert die Kaskade x (x = 0...11), d.h. nur in diesem Zustand können Widerstandswerte über die RTIO zur Hardware transportiert werden.



Tab. 25-4 fasst die Eigenschaften der RTIO-Signale zusammen.

Parameter bzw. Optionsfeld	Datentyp*	Editierbar im Laufzeitmodus	Bemerkung / Wertebereich
Cascade Mode	uint8	Ja	Konfiguriert die Widerstandskaskaden: 0: „Disabled“, Kaskade ist deaktiviert 1: „Enabled“, Kaskade ist aktiviert

\* Datentyp, den der RTIO-Treiber intern für den Parameter verwendet

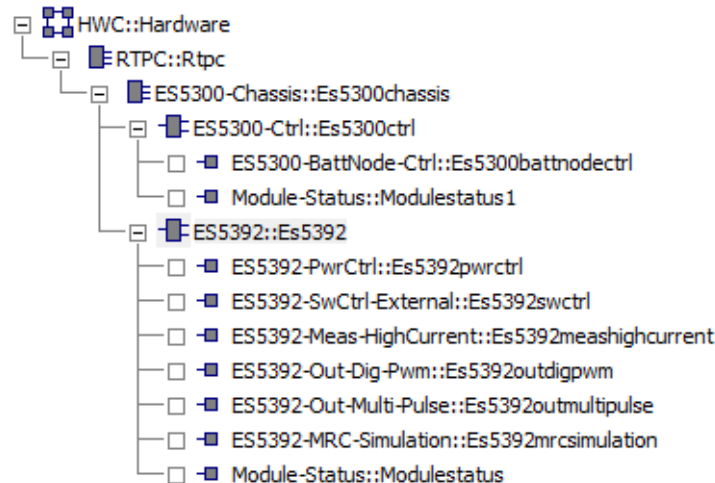
**Tab. 25-4** ES5385-ResistorCascade Device: Konfigurationsparameter der Registerkarte „Signals“



## 26 ES5392.1 High Current Switch Board

Das ES5392.1 High Current Switch Board dient zur Steuerung einer externen Spannungsversorgung und zum Schalten von Batterieknoten.

Im RTIO-Editor wird das ES5392.1 High Current Switch Board durch Auswahl des ES5392 Subsystems eingebunden.

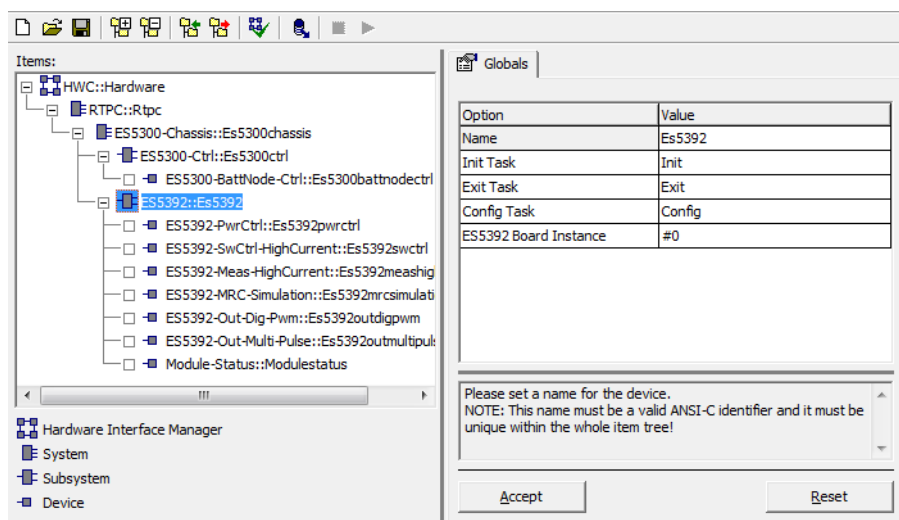


**Abb. 26-1** RTIO-Hardwarebeschreibung mit ES5392.1 High Current Switch Board

### 26.1 ES5392 Subsystem

#### 26.1.1 Globals (ES5392 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des ES5392 Subsystems beschrieben.



**Abb. 26-2** Die Registerkarte „Globals“ des ES5392 Subsystems

### ES5392 Board Instance

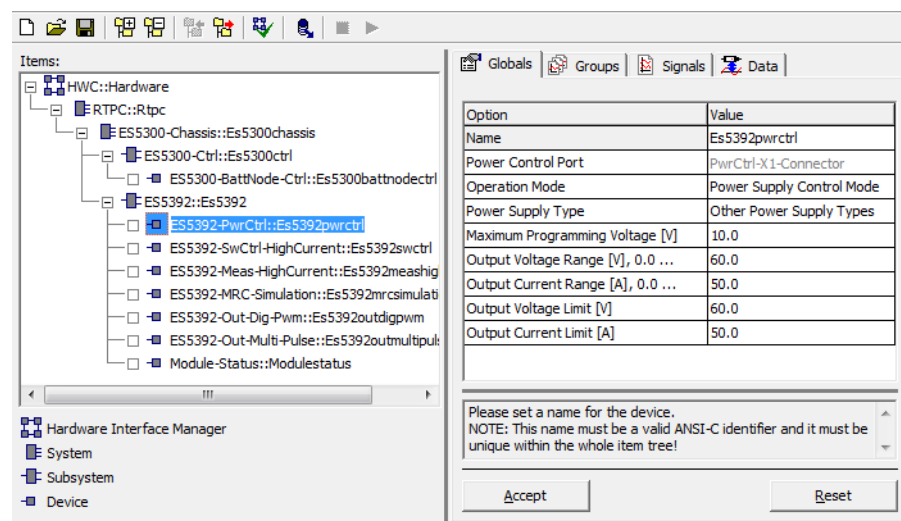
Im RTIO-Editor können bis zu vier ES5392.1-Boards pro ES5300 Baugruppen-träger eingebunden werden. Diese Option dient zur Identifizierung des ES5392.1 High Current Switch Boards – nummeriert von links (Steckplatz 0) nach rechts.

Dieser RTIO-Parameter ist während der Laufzeit des Modells auf dem Experimentaltarget nicht editierbar.

## 26.2 ES5392-PwrCtrl Device

### 26.2.1 Globals (ES5392-PwrCtrl Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 26-3** Die Registerkarte „Globals“ des ES5392-PwrCtrl Device

#### Power Control Port

Das ES5392.1 High Current Switch Board hat einen Anschluss „PwrCtrl “ (X1) auf der Frontplatte.

#### Operation Mode

Betriebsart der Stromversorgungseinheit:

- **Analog & Digital IO Mode**  
Für allgemeine digitale/analoge I/O-Operationen
- **Power Supply Control Mode**  
Ansteuerung einer Stromversorgungseinheit, die unter „Power Supply Type“ gewählt werden kann.

#### Power Supply Type

Auswahl der verwendeten Stromversorgungseinheit.

Wenn Sie keines der vordefinierten Geräte verwenden, wählen Sie „Other Power Supply Types“. Sie können dann die folgenden Parameter entsprechend anpassen.

#### *Maximum Programming Voltage [V]*

---

Maximale Steuerspannung in V.

#### **Hinweis**

*Steuerspannungen > 10 V können mit der ES5392.1 nicht erzeugt werden!*

#### *Output Voltage Range [V]*

---

Maximale Ausgangsspannung in V. Diese Spannung wird ausgegeben, wenn die Steuerspannung den maximalen Wert hat und die Stromversorgungseinheit im Modus „Constant Voltage“ (CV) arbeitet.

#### *Output Current Range [A]*

---

Maximaler Ausgangsstrom in A. Diese Strom wird ausgegeben, wenn die Steuerspannung den maximalen Wert hat und die Stromversorgungseinheit im Modus „Constant Current“ (CC) arbeitet.

#### *Output Voltage Limit [V]*

---

Begrenzung der Ausgangsspannung - dieser Wert wird nicht überschritten, auch wenn eine höhere Ausgangsspannung programmiert wurde.

#### *Output Current Limit [A]*

---

Begrenzung des Ausgangsstroms - dieser Wert wird nicht überschritten, auch wenn ein höherer Ausgangsstrom programmiert wurde.

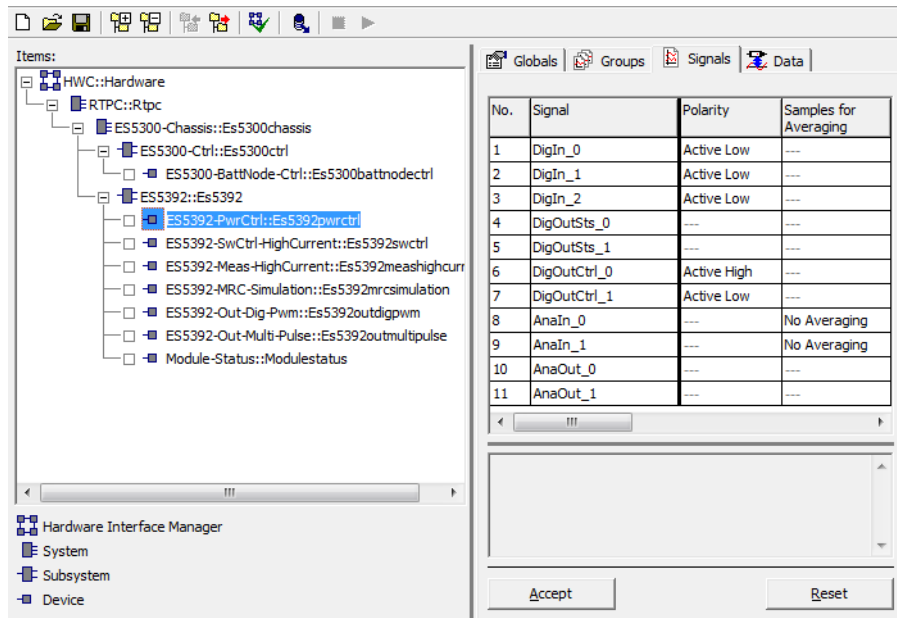
#### 26.2.2 Groups (ES5392-PwrCtrl Device)

---

Das ES5392-PwrCtrl Device besitzt vier Gruppen für die jeweils analogen oder digitalen Ein- oder Ausgänge.

## 26.2.3 Signals (ES5392-PwrCtrl Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-4** Die Registerkarte „Signals“ des ES5392-PwrCtrl Device

Signalname	Anmerkungen
DigIn_x	0: Alarm Input Monitor (X1 Pin 8 / DGND = Pin 11,12) 1: CV Mode Input Monitor (X1 Pin 2 / DGND = Pin 11,12) 2: CC Mode Input Monitor (X1 Pin 7 / DGND = Pin 11,12)
DigOutSts_x	0: Power Supply Output Monitor (X1 Pin 6 / DGND = Pin 11,12) 1: AC Input to Power Supply Monitor (X1 Pin 1 / DGND = Pin 11,12)
DigOutCtrl_x	0: Power Supply Output Control (X1 Pin 6 / DGND = Pin 11,12) 1: AC Input to Power Supply Control (X1 Pin 1 / DGND = Pin 11,12)
AnaIn_x	0: Aktuelle Spannung (X1 Pin 4 / AGND = Pin 3) 1: Aktueller Strom (X1 Pin 9 / AGND = Pin 14)
AnaOut_x	0: CV Mode: Sollspannung, CC Mode: Spannungslimit (X1 Pin 5 / AGND = Pin 13) 1: CC Mode: Sollstrom, CV Mode: Stromlimit (X1 Pin 10 / AGND = Pin 15)

**Tab. 26-1** Die Signale des ES5392-PwrCtrl Device

Für die Signale gibt es folgende Parameter:

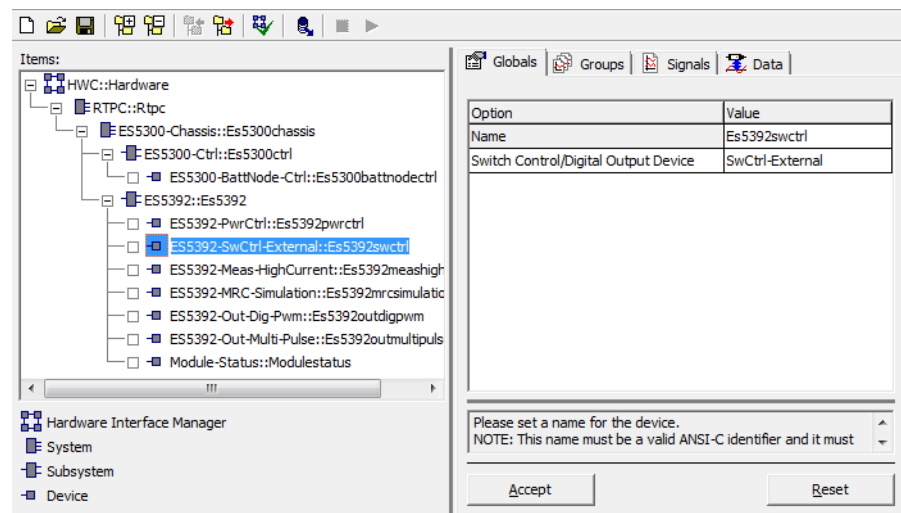
Parameter bzw. Optionsfeld	Online editierbar	Bemerkung/Wertebereich
Polarity	Ja	Polarität der Signale
Samples for Averaging	Ja	Anzahl der Samples zur Mittelwertbildung

**Tab. 26-2** ES5392-PwrCtrl Device: Konfigurationsparameter der Registerkarte „Signals“

## 26.3 ES5392-SwCtrl-External Device

### 26.3.1 Globals (ES5392-SwCtrl-External Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 26-5** Die Registerkarte „Globals“ des ES5392-SwCtrl-External Device  
*Switch Control/Digital Output Device*

Spezifikation des Ausgangs (Switch Control/Digital Output).

Folgende Konfigurationen sind möglich:

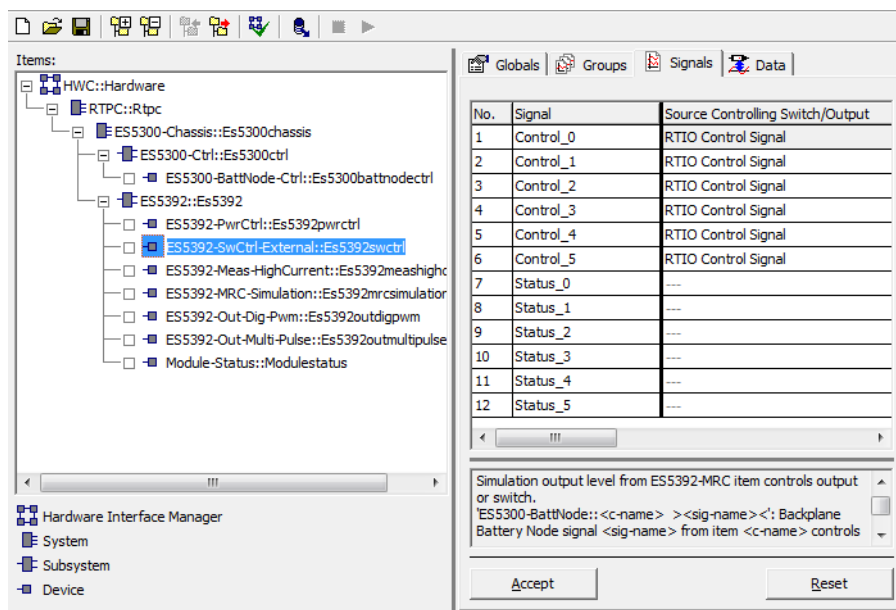
- **SwCtrl-External**  
Signale werden am Anschluss „SwCtrl“ (X2) zur Verfügung gestellt (siehe „Signals (ES5392-SwCtrl-External Device)“ auf Seite 892).
- **SwCtrl-DigOut**  
Weitere Signale werden am Anschluss X4 zur Verfügung gestellt (siehe „Signals (ES5392-SwCtrl-DigOut)“ auf Seite 894).
- **SwCtrl-HighCurrent**  
Zur Steuerung der Hochstromschalter – die geschalteten Batteriespannungen werden Anschluss X4 zur Verfügung gestellt (siehe „Signals (ES5392-SwCtrl-HighCurrent)“ auf Seite 895).

26.3.2 Groups (ES5392-SwCtrl-External Device)

Das ES5392-SwCtrl-External Device besitzt zwei Gruppen für Steuer- und Statussignale.

26.3.3 Signals (ES5392-SwCtrl-External Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-6** Die Registerkarte „Signals“ des ES5392-SwCtrl-External Device

Signalname	Anmerkungen
Control_x (x = 0...5)	Digitaler Ausgang 0: 0 V 1: 5 V
Status_x (x = 0...5)	Status des Schalters oder Ausgangs #x 0: Kein Fehler 1: Fehler

**Tab. 26-3** Die Signale des ES5392-SwCtrl-External Device

Zur Steuerung der Ausgänge gibt es folgenden Parameter:



### Source Controlling Switch/Output

---

- **RTIO Control Signal**  
Ausgang/Schalter wird vom RTIO-Signal dieses Kanals gesteuert.
- **ES5392-MRC-Simulation::  
Ausgang/Schalter wird vom Ausgangspegel der Main-Relay-Simulation gesteuert (siehe „Globals (ES5392-MRC-Simulation Device)“ auf Seite 899).**
- **ES5300-BattNode::  
Ausgang/Schalter wird vom Batterieknotensignal <sig-name> des Items <c-name> gesteuert (siehe „ES5300-BattNode-Ctrl Device“ auf Seite 629).**
- **ES5392-Out-Dig-Pwm::  
Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (positive Polarität).**
- **ES5392-Out-Dig-Pwm::  
Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (invertierte Polarität).**
- **ES592-Out-Multi-Pulse::  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (positive Polarität).**
- **ES592-Out-Multi-Pulse::  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (invertierte Polarität).**

26.3.4 Signals (ES5392-SwCtrl-DigOut)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.

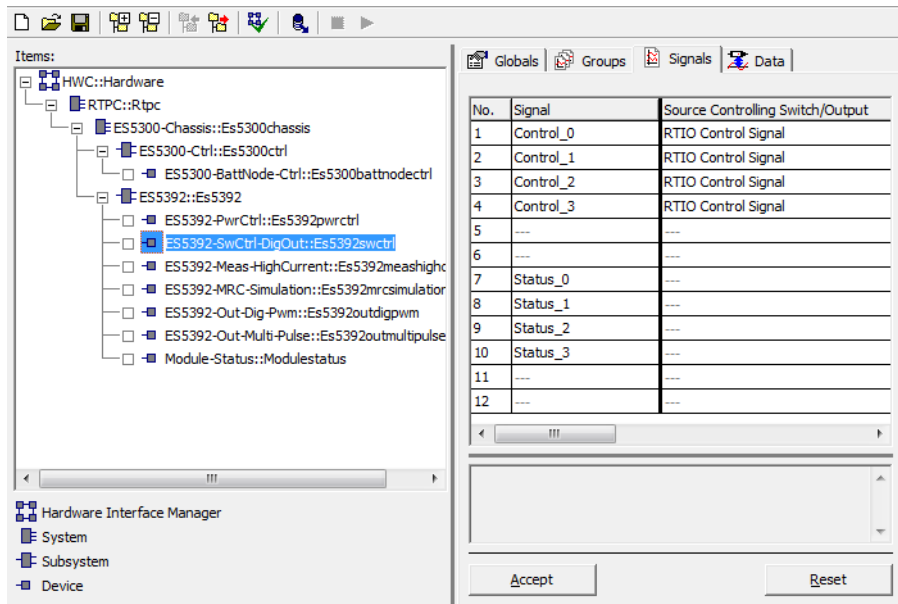


Abb. 26-7 Die Registerkarte „Signals“ des ES5392-SwCtrl-DigOut Device

Signalname	Anmerkungen
Control_x (x = 0...3)	Digitaler Ausgang 0: 0 V 1: 5 V
Status_x (x = 0...3)	Status des Schalters oder Ausgangs #x 0: Kein Fehler 1: Fehler

Tab. 26-4 Die Signale des ES5392-SwCtrl-DigOut Device

Zur Steuerung der Ausgänge gibt es folgenden Parameter:

Source Controlling Switch/Output

- RTIO Control Signal**  
 Ausgang/Schalter wird vom RTIO-Signal dieses Kanals gesteuert.
- ES5392-MRC-Simulation::<c-name> ><sig-name><**  
 Ausgang/Schalter wird vom Ausgangspegel der Main-Relay-Simulation (siehe „Globals (ES5392-MRC-Simulation Device)“ auf Seite 899) gesteuert.
- ES5300-BattNode::<c-name> ><sig-name><**  
 Ausgang/Schalter wird vom Batterienotensignal <sig-name> des Items <c-name> gesteuert (siehe „ES5300-BattNode-Ctrl Device“ auf Seite 629).
- ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=High/On**  
 Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (positive Polarität).

- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=Low/Off**  
Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (invertierte Polarität).
- **ES592-Out-Multi-Pulse::<c-name>: Active=High/On**  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (positive Polarität).
- **ES592-Out-Multi-Pulse::<c-name>: Active=Low/Off**  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (invertierte Polarität).

26.3.5 Signals (ES5392-SwCtrl-HighCurrent)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.

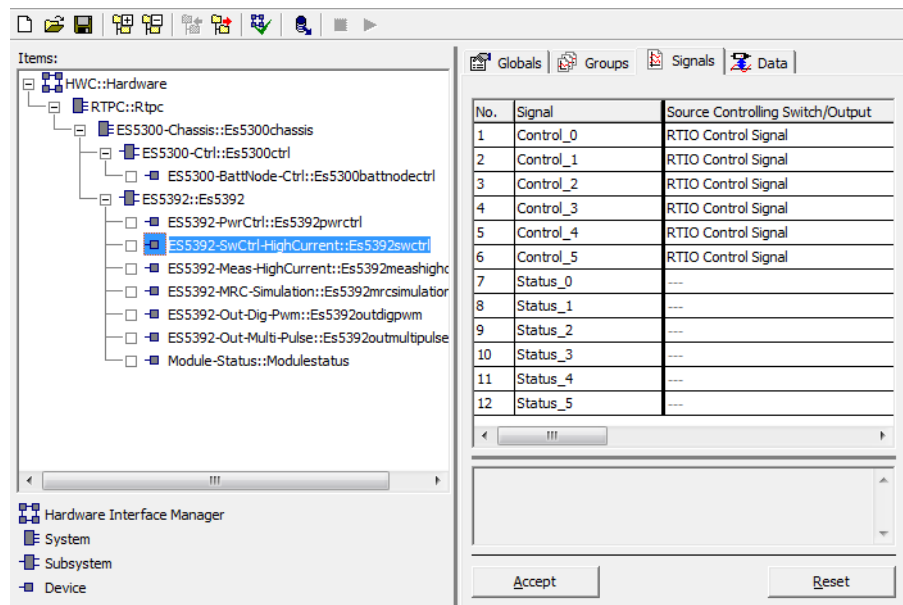


Abb. 26-8 Die Registerkarte „Signals“ des ES5392-SwCtrl-DigOut Device

Signalname	Anmerkungen
Control_x (x = 0...5)	Digitaler Ausgang 0: 0 V 1: 5 V
Status_x (x = 0...5)	Status des Schalters oder Ausganges #x 0: Kein Fehler 1: Fehler

Tab. 26-5 Die Signale des ES5392-SwCtrl-DigOut Device

Zur Steuerung der Ausgänge gibt es folgenden Parameter:

*Source Controlling Switch/Output*

---

- **RTIO Control Signal**  
Ausgang/Schalter wird vom RTIO-Signal dieses Kanals gesteuert.
- **ES5392-MRC-Simulation::  
Ausgang/Schalter wird vom Ausgangspegel der Main-Relay-Simulation (siehe „Globals (ES5392-MRC-Simulation Device)“ auf Seite 899) gesteuert.**
- **ES5300-BattNode::  
Ausgang/Schalter wird vom Batterieknotensignal <sig-name> des Items <c-name> gesteuert (siehe „ES5300-BattNode-Ctrl Device“ auf Seite 629).**
- **ES5392-Out-Dig-Pwm::  
Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (positive Polarität).**
- **ES5392-Out-Dig-Pwm::  
Ausgang/Schalter wird vom PWM-Kanal <x> gesteuert (invertierte Polarität).**
- **ES592-Out-Multi-Pulse::  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (positive Polarität).**
- **ES592-Out-Multi-Pulse::  
Ausgang/Schalter wird vom Multi-Pulse-Kanal <x> gesteuert (invertierte Polarität).**

*Cut-Off Limit [A]*

---

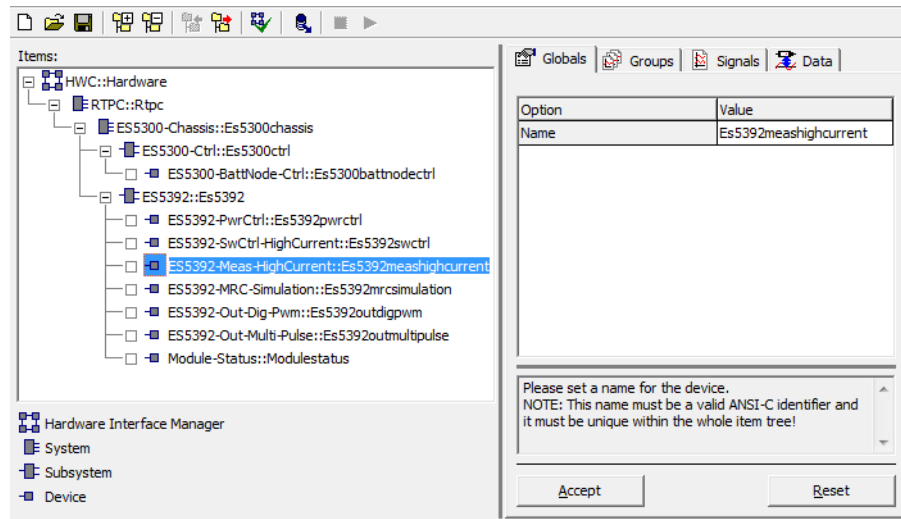
Der Wert des Auslösestroms der selbstheilenden Sicherung.

Wenn Temperatur und Strom wieder akzeptable Werte angenommen haben, schaltet die Sicherung wieder ein.

## 26.4 ES5392-Meas-HighCurrent Device

### 26.4.1 Globals (ES5392-Meas-HighCurrent Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



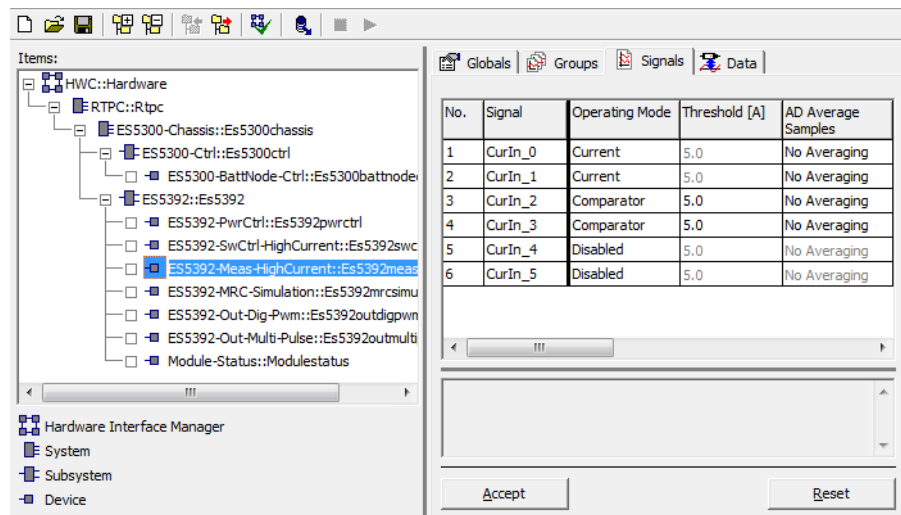
**Abb. 26-9** Die Registerkarte „Globals“ des ES5392-Meas-HighCurrent Device

### 26.4.2 Groups (ES5392-Meas-HighCurrent Device)

Das ES5392-Meas-HighCurrent Device besitzt eine Receive-Signalgruppe „CurrentIn“.

### 26.4.3 Signals (ES5392-Meas-HighCurrent Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-10** Die Registerkarte „Signals“ des ES5392-Meas-HighCurrent Device

Signalname	Anmerkungen
CurIn_x	Konfiguration der Strommessung für Kanal x (x = 0...5) Wertebereich: -15.0 A...+15.0 A

**Tab. 26-6** Die Signale des ES5392-Meas-HighCurrent Device

Für die Signale gibt es folgende Parameter:

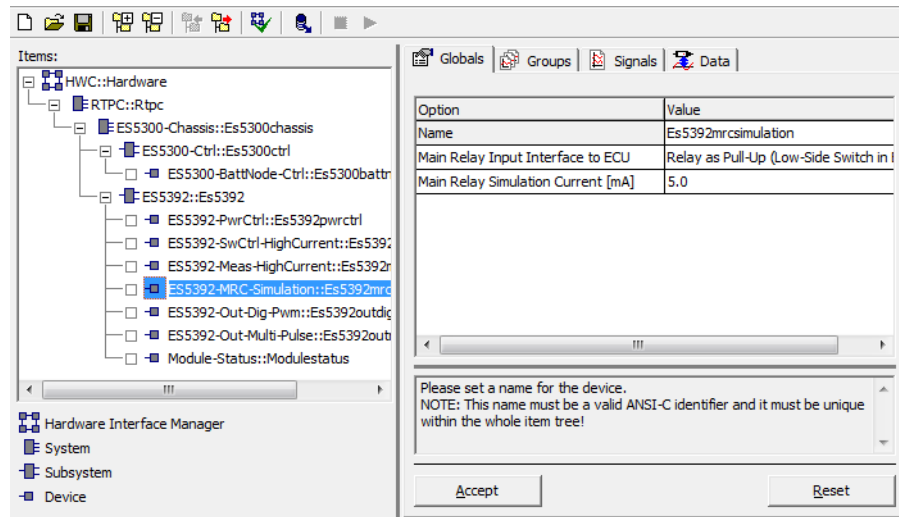
Parameter bzw. Optionsfeld	Online editierbar	Bemerkung/Wertebereich
Operating Mode	Ja	- Disabled: CurIn_x =: 0.0 - Current: CurIn_x enthält den gemessenen Strom - Comparator: CurIn_x = 1.0 (true), wenn der eingestellte Schwellwert überschritten wurde
Threshold [A]	Ja	Komparatorschwelle Ist der gemessene Strom gleich oder größer als dieser Wert, wird das Signal auf „true“ (1.0) gesetzt
AD Average Samples	Ja	Anzahl der Samples zur Mittelwertbildung

**Tab. 26-7** ES5392-Meas-HighCurrent Device: Konfigurationsparameter der Registerkarte „Signals“

## 26.5 ES5392-MRC-Simulation Device

### 26.5.1 Globals (ES5392-MRC-Simulation Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 26-11** Die Registerkarte „Globals“ des ES5392-MRC-Simulation Device  
*Main Relay Input Interface to ECU*

Ermöglicht die Simulation des „Main Relay“ für das Steuergerät.

Mögliche Einstellungen:

- Disabled
- Relay as Pull-Up (zu +UBatt)
- Relay as Pull-Down (zu GND)

*Main Relay Simulation Current [mA]*

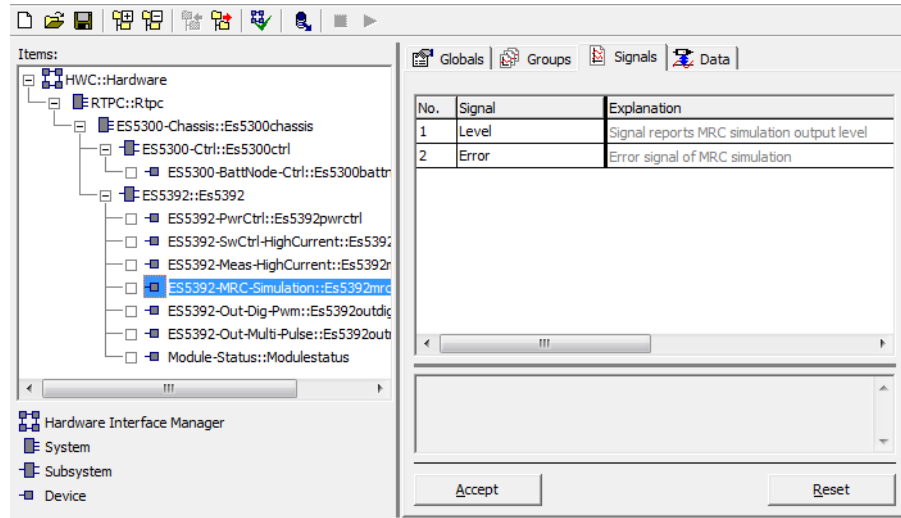
Strom (in mA) der Stromquelle

### 26.5.2 Groups (ES5392-MRC-Simulation Device)

Das ES5392-MRC-Simulation Device besitzt eine Receive-Signalgruppe „Status“ für Pegel- und Fehlerinformationen.

26.5.3 Signals (ES5392-MRC-Simulation Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-12** Die Registerkarte „Signals“ des ES5392-MRC-Simulation Device

Signalname	Anmerkungen
Level	Status des Ausgangs der MRC-Simulation: 0 = Off 1 = On
Error	Fehlerstatus des MRC-Ausgangs 0 = Ok 1 = Error

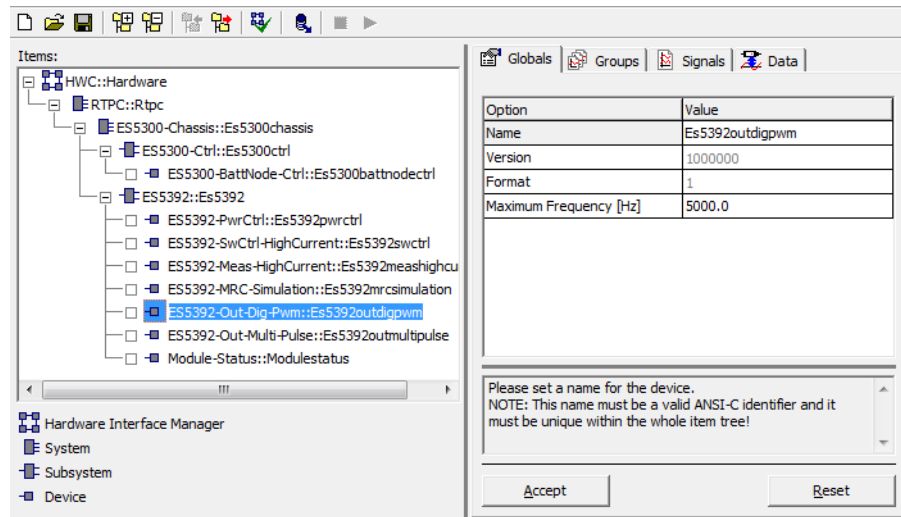
**Tab. 26-8** Die Signale des ES5392-MRC-Simulation Device



## 26.6 ES5392-Out-Dig-Pwm Device

### 26.6.1 Globals (ES5392-Out-Dig-Pwm Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 26-13** Die Registerkarte „Globals“ des ES5392-Out-Dig-Pwm Device  
Wenn im Menü des Editors die Option **View → Show All** gewählt wurde, wird die folgende Option sichtbar:

*Maximum Frequency [Hz]*

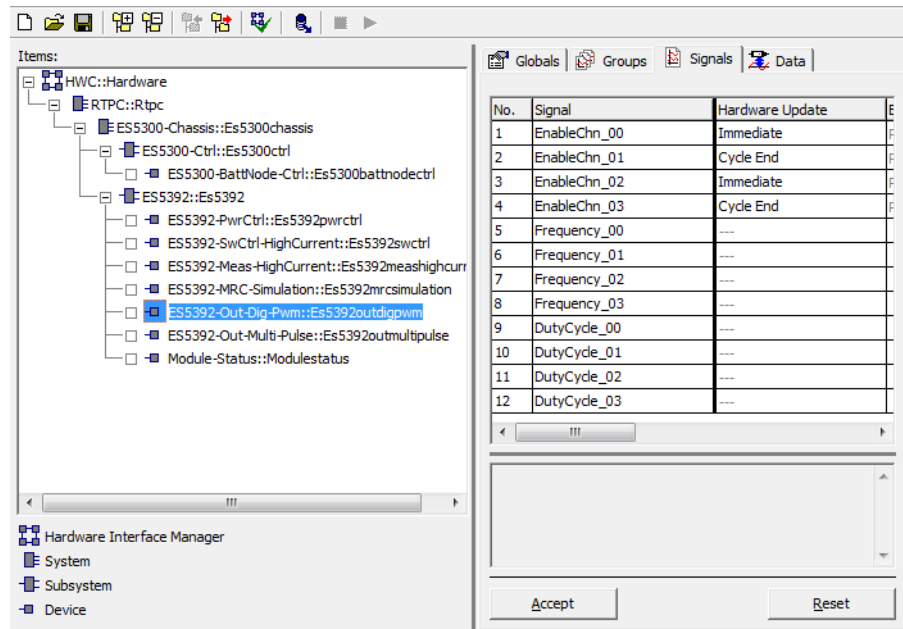
Maximale einstellbare Frequenz (in Hz) für alle PWM-Einheiten diese Devices.

### 26.6.2 Groups (ES5392-Out-Dig-Pwm Device)

Das ES5392-Out-Dig-Pwm Device besitzt eine Send-Gruppe „Control“ für die Steuersignale der PWM-Kanäle.

## 26.6.3 Signals (ES5392-Out-Dig-Pwm Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-14** Die Registerkarte „Signals“ des ES5392-Out-Dig-Pwm Device

Signalname	Anmerkungen
EnableChn_xx	Aktivierung des PWM-Kanals 'xx': 0 = Deaktiviert 1 = Aktiviert
Frequency	Frequenz des PWM-Signals in Hz, Wertebereich: $4.547 \times 10^{-5} \dots 5000.0$
DutyCycle	Tastverhältnis des PWM-Signals. Wertebereich: 0.0...1.0

**Tab. 26-9** Die Signale des ES5392-Out-Dig-Pwm Device

#### Hardware Update

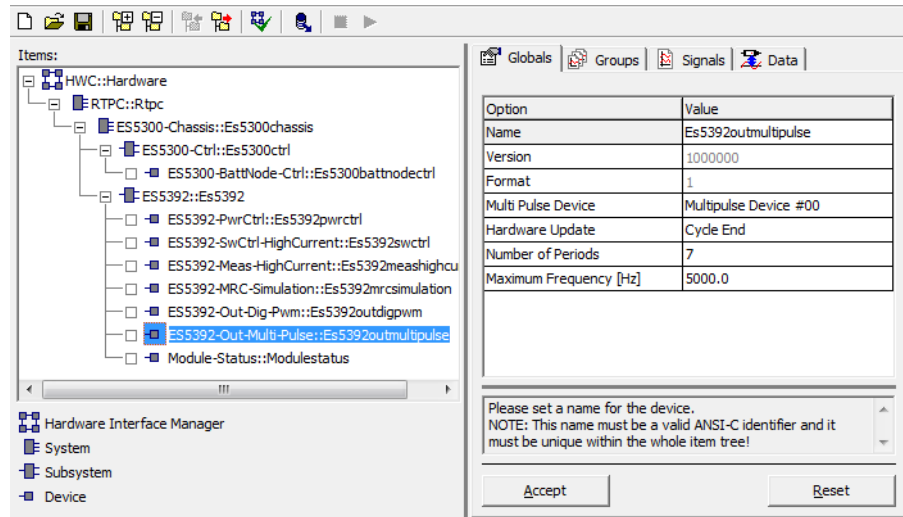
Für das Signal „EnableChn“ gibt es die Option „Hardware Update“, mit der definiert wird, wann Änderungen von Frequenz oder Tastverhältnis übernommen werden.

- **Immediate**  
Änderungen werden sofort übernommen.
- **Cycle End**  
Änderungen werden erst nach dem Ende der aktuellen Periode übernommen.

## 26.7 ES5392-Out-Multi-Pulse Device

### 26.7.1 Globals (ES5392-Out-Multi-Pulse Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Globals“.



**Abb. 26-15** Die Registerkarte „Globals“ des ES5392-Out-Multi-Pulse Device  
*Multi Pulse Device*

Definiert den Ausgangskanal für das Multipulse-Signal (0...3).

#### *Hardware Update*

Hiermit wird definiert, wann Änderungen von Frequenz oder Tastverhältnis übernommen werden:

- **RTIO Controlled**  
Änderungen werden an den steigenden Flanken des Signals „SyncSgl“ übernommen.
- **Immediate**  
Änderungen werden sofort übernommen.
- **Cycle End**  
Änderungen werden erst nach dem Ende der aktuellen Periode übernommen.

#### *Number of Periods*

Anzahl der Perioden (1...8, jeweils definiert durch Frequenz und Tastverhältnis) eines Multipulse-Zyklus.

#### *Maximum Frequency [Hz]*

Maximale einstellbare Frequenz (in Hz) für alle PWM-Einheiten diese Devices.

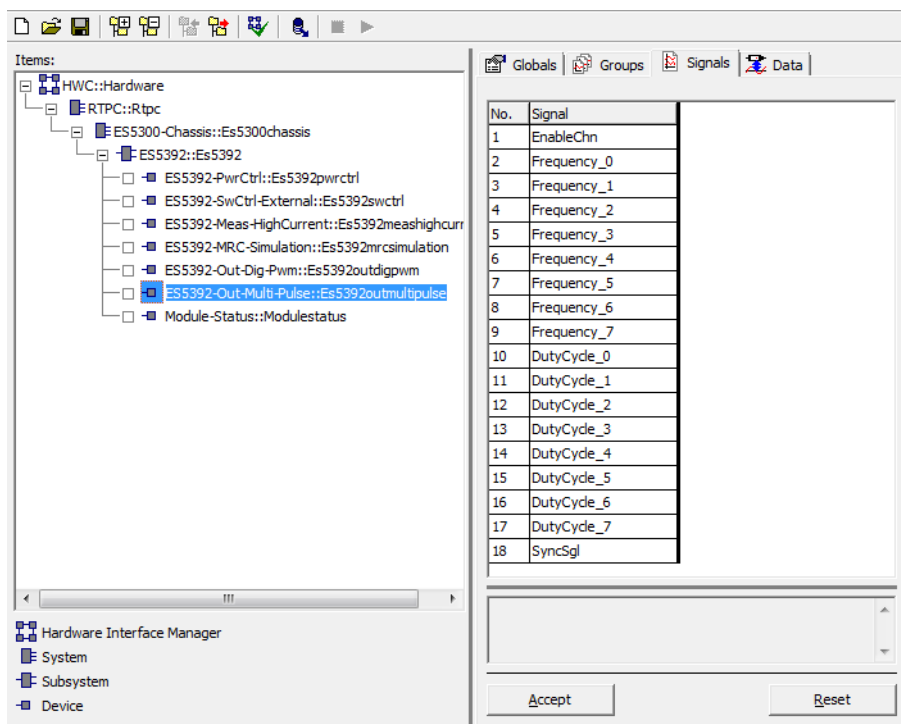
Nur sichtbar, wenn im Menü des Editors die Option **View** → **Show All** gewählt wurde.

### 26.7.2 Groups (ES5392-Out-Multi-Pulse Device)

Das ES5392-Out-Multi-Pulse Device besitzt eine Send-Gruppe „Control“ zur Steuerung der Multipulse-Ausgänge.

### 26.7.3 Signals (ES5392-Out-Multi-Pulse Device)

Die folgende Abbildung zeigt die RTIO-Konfigurationsparameter der Registerkarte „Signals“.



**Abb. 26-16** Die Registerkarte „Signals“ des ES5392-Out-Multi-Pulse Device

Signalname	Anmerkungen
EnableChn	Aktivierung des Multipulse-Kanals: 0 = Deaktiviert 1 = Aktiviert
Frequency_x [x = 0...7]	Frequenz des Signals in Hz Wertebereich: $1.164 \times 10^{-2} \dots 5000.0$
DutyCycle_x [x = 0...7]	Tastverhältnis des PWM-Signals Wertebereich: 0.0...1.0
SyncSgl	Synchronisationssignal für RTIO-gesteuerten Trigger bei 0 → 1-Übergang (siehe „Hardware Update“ auf Seite 903).

## 26.8 Module-Status Device

Siehe „Module Status“ auf Seite 1005.

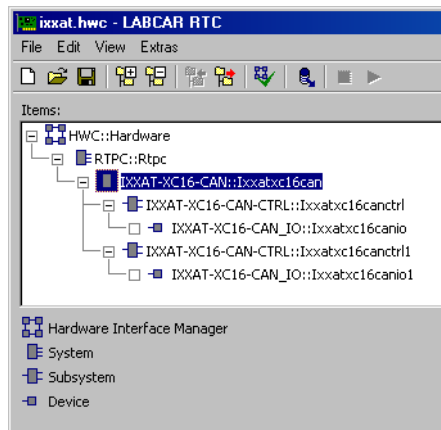
## 27 IXXAT iPCI-I XC16/PCI CAN Interfacekarte

Dieses Dokument beschreibt die RTIO-Einbindung der PCI CAN-Karte „iPCI-I XC16/PCI“ der Firma IXXAT für ETAS RTPC. In der RTIO und auch in dieser Dokumentation wird die Karte als „IXXAT-XC16-CAN“ bezeichnet.

Die Karte verfügt über zwei unabhängig nutzbare CAN-Controller, die - bei entsprechender Bestückung - per Software zwischen „High-Speed“ und „Low-Speed“ CAN-Modus umgeschaltet werden kann.

### *Aufbau des IXXAT-XC16-CAN RTIO-Baums*

Eine IXXAT-XC16-CAN-Karte wird in einen freien PCI-Steckplatz eines Real-Time PC gesteckt und dort ist sie direkt über den PCI-Bus mit dem Experimentaltarget „RTPC“ verbunden. Die beschriebene Hardware-Anbindung spiegelt sich im RTIO-Baum wider (Abb. 27-1). Das „IXXAT-XC16-CAN“ RTIO-Element einer IXXAT-XC16-CAN-Karte wird direkt dem Experimentaltarget „RTPC“ zugeordnet.



**Abb. 27-1** RTIO-Hardwarebeschreibung mit eingebundener IXXAT-XC16-CAN-Karte

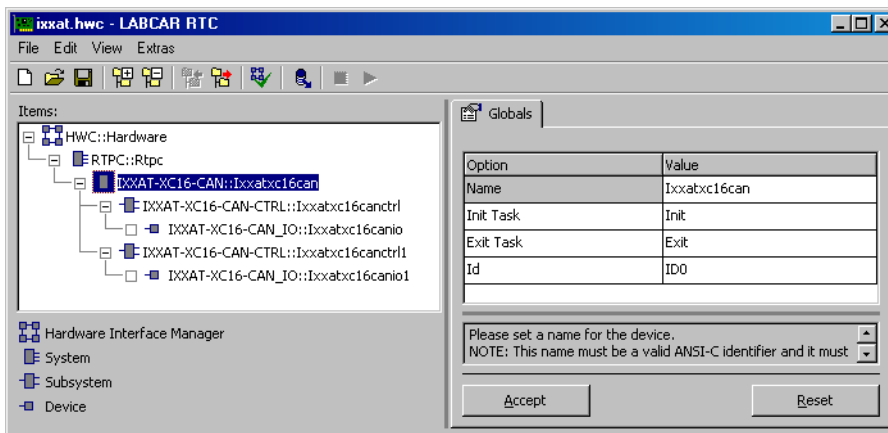
### Hinweis

Abb. 27-1 zeigt die Anbindung der IXXAT-XC16-CAN an ein RTPC-Experimentaltarget, die Anbindung an ein ES1130 Experimentaltarget ist nicht möglich.

## 27.1 IXXAT-XC16-CAN Subsystem

### 27.1.1 Globals (IXXAT-XC16-CAN Subsystem)

Abb. 27-2 zeigt die Registerkarte „Globals“ eines IXXAT-XC16-CAN Subsystems.



**Abb. 27-2** Die Registerkarte „Globals“ des IXXAT-XC16-CAN Subsystems

*Id*

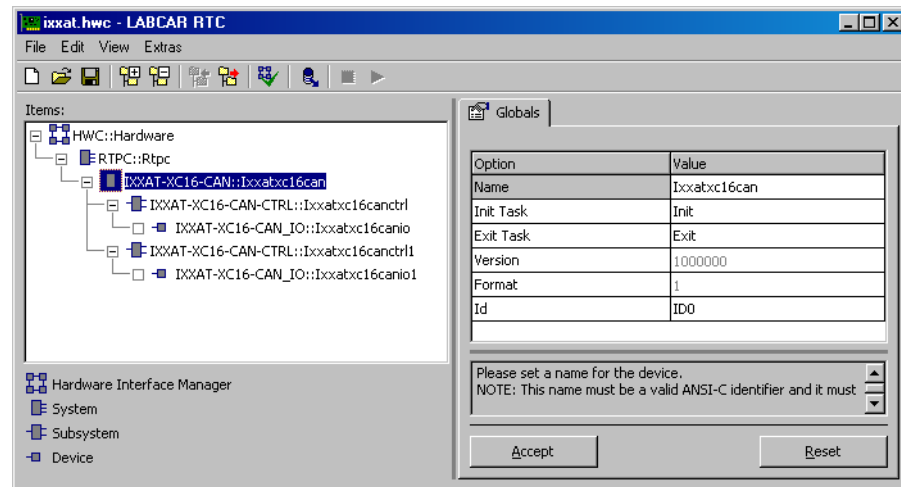
Dieses Optionsfeld dient zur Identifizierung der IXXAT-XC16-CAN-Karte. Es stellt die Zuordnung zwischen der RTIO-Hardwarebeschreibung und der IXXAT-XC16-CAN-Karte im PCI Bus her, für welche diese Beschreibung gültig ist. Die Nummerierung der IXXAT-XC16-CAN-Karten im Real-Time PC hängt vom verwendeten PC ab und ist nicht einheitlich, die Nummerierung beginnt mit 0. Es können bis zu vier IXXAT-XC16-CAN-Karten pro Real-Time PC eingebunden werden.

Dieser RTIO-Parameter ist nicht zur Laufzeit einstellbar.

### 27.1.2 Versteckte Optionsfelder

Wird mit der rechten Maustaste in die Registerkarte „Globals“ geklickt und anschließend im erscheinenden kontextsensitiven Menü die Option „Show all Options“ gewählt, so werden weitere Optionsfelder sichtbar (Abb. 27-3). Diese Optionsfelder erlauben es ETAS-Servicepersonal, bei einer Fehlersuche zusätzliche Informationen über die IXXAT-XC16-CAN-Karte zu erhalten. Sie sind nicht für den Anwender gedacht - die Voreinstellungen sollten vom Anwender nicht geändert werden.

Die Registerkarte „Globals“ mit versteckten Optionen eines IXXAT-XC16-CAN Subsystems ist in Abb. 27-3 dargestellt.



**Abb. 27-3** Die Registerkarte „Globals“ des IXXAT-XC16-CAN Subsystems mit versteckten Optionen

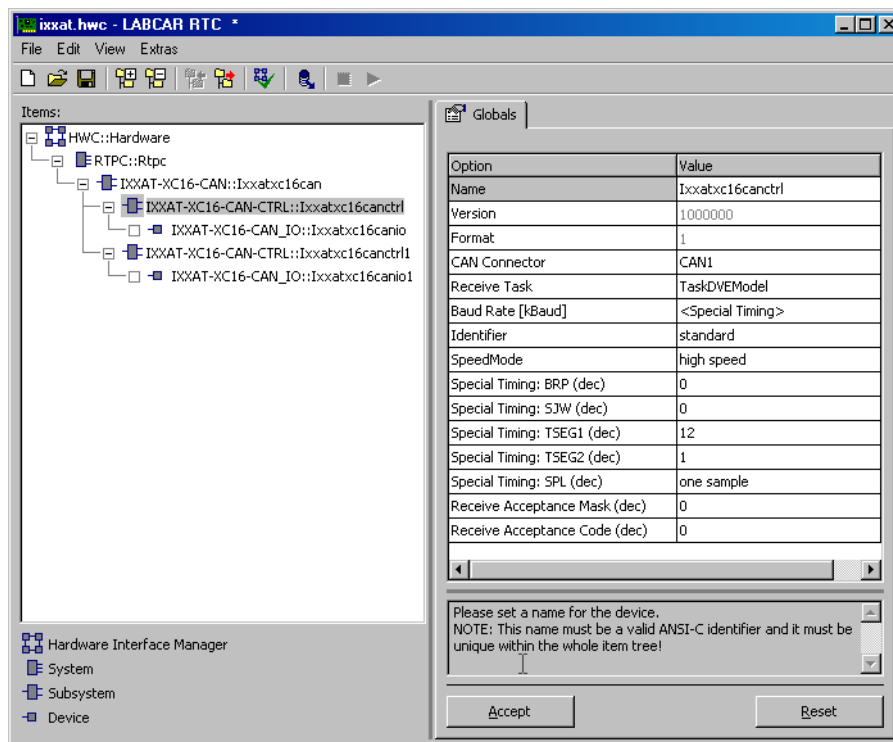
## 27.2 IXXAT-XC16-CAN-CTRL Subsystem

### 27.2.1 Globals (IXXAT-XC16-CAN-CTRL Subsystem)

#### **Hinweis**

*Im Online-Modus sind keine Änderungen an den Einstellungen der IXXAT-XC16-CAN Karte möglich.*

Abb. 27-4 zeigt die Registerkarte „Globals“ eines IXXAT-XC16-CAN-CTRL Subsystems.



**Abb. 27-4** Die Registerkarte „Globals“ eines IXXAT-XC16-CAN-CTRL Subsystems

#### *CAN Connector*

Dieses Optionsfeld dient zur Zuordnung des IXXAT-XC16-CAN-CTRL Subsystems zu einem CAN-Controller bzw. CAN-Anschlusses am Slotblech der IXXAT-XC16-CAN. Hierbei ist zu beachten, dass man ein IXXAT-XC16-CAN-CTRL Subsystem nur einem einzigen CAN-Controller zuordnen kann. Mögliche Werte sind „CAN1“ und „CAN2“.

#### *Receive Task*

Bei der IXXAT-XC16-CAN werden alle CAN Empfangsbotschaften in einer zentralen Funktion verarbeitet und auf die einzelnen Signale verteilt. Die Task, in der diese zentrale Funktion abläuft, kann mit dieser Option ausgewählt werden. Für Empfangsbotschaften besteht daher im IXXAT-XC16-CAN\_IO Device der RTIO keine Möglichkeit, eine Task auszuwählen.



### Baud Rate [kBaud]

In diesem Optionsfeld kann die Übertragungsrate des CAN-Controllers festgelegt werden. Es stehen hierfür die Standard-Baudraten (1000, 500, 250, 125, 100, 50, 20, 10 kBaud) zur Verfügung. Mit Hilfe der Einstellung „Special Timing“ wird die Low-Level-Ansteuerung des CAN-Controllers bezüglich Bit-Timing und Baud Rate aktiviert.

Bei der Konfiguration der „Baud Rate“ werden automatisch die „Special Timing“ Optionsfelder bezüglich der eingestellten Übertragungsrate mit Werten konfiguriert und zur Firmware transferiert.

### Identifizier

In dieser Zeile kann die Länge des Identifizier-Felds der CAN-Botschaft ausgewählt werden. Bei CAN-Botschaften kann grundsätzlich zwischen „standard“ Frames mit 11-Bit Identifiern oder „extended“ Frames mit 29-Bit Identifiern gewählt werden.

#### **Hinweis**

*Pro CAN-Controller muss bezüglich der Länge des Identifiers eine einheitliche Einstellung vorliegen.*

#### **Hinweis**

*Wenn der Identifizier „standard“ gewählt wird, ist in der Registerkarte „Groups“ des IXXAT-XC16-CAN\_IO Devices nur die Eingabe von 11 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die „most significant bits“ (MSB) abgeschnitten. Eine Warnung wird nicht angezeigt.*

#### **Hinweis**

*Wenn der Identifizier „extended“ gewählt wird, ist nur die Eingabe von 29 Bit-Werten erlaubt. Wird ein größerer Wert eingegeben, werden die „most significant bits“ (MSB) abgeschnitten. Eine Warnung wird nicht angezeigt.*

### SpeedMode

Mit diesem Optionsfeld kann zwischen der „High-Speed“ und „Low-Speed“ CAN-Funktionalität gewechselt werden. Die Option „high speed“ definiert den CAN-Controller als „High-Speed“ Schnittstelle während „low speed“ den CAN-Controller als „Fault tolerant Low-Speed“ Schnittstelle definiert.

### Special Timing: BRP (dec)

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Baud Rate Prescaler“, welcher aus dem Eingangstakt des CAN-Controllers die Baud Rate bestimmt. Der Wertebereich dieser Einstellung beträgt 0 - 255.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

---

*Special Timing: SJW (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung der „Resynchronization Jump Width“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 3.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

---

*Special Timing: TSEG1 (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Phase Segment 1“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 7.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

---

*Special Timing: TSEG2 (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Phase Segment 2“ innerhalb des Bit-Timings. Der Wertebereich dieser Einstellung beträgt 0 - 7.

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

---

*Special Timing: SPL (dec)*

---

Diese Option ist nur editierbar, wenn die Option „Baud Rate [kBaud]“ auf „Special Timing“ eingestellt ist. Dieser Parameter dient zur Einstellung des „Sampling Mode“ und bestimmt, wie oft das Signal abgetastet wird, um den logischen Zustand zu ermitteln. Die möglichen Einstellungen sind „one sample“ (einmal abgetastet) oder „three samples“ (dreimal abgetastet).

Mehr Informationen zu dieser Einstellung finden Sie im Datenblatt des CAN-Controllers.

Die Formel zur Berechnung der CAN Bus Baud Rate lautet:

$$\text{Baud Rate} = 8 \text{ MHz} / [(\text{BRP} + 1) \times (3 + \text{TSEG1} + \text{TSEG2})]$$

---

*Receive Acceptance Mask (dec)*

---

Informationen zu dieser Option finden Sie im Datenblatt des CAN-Controllers.

---

*Receive Acceptance Code (dec)*

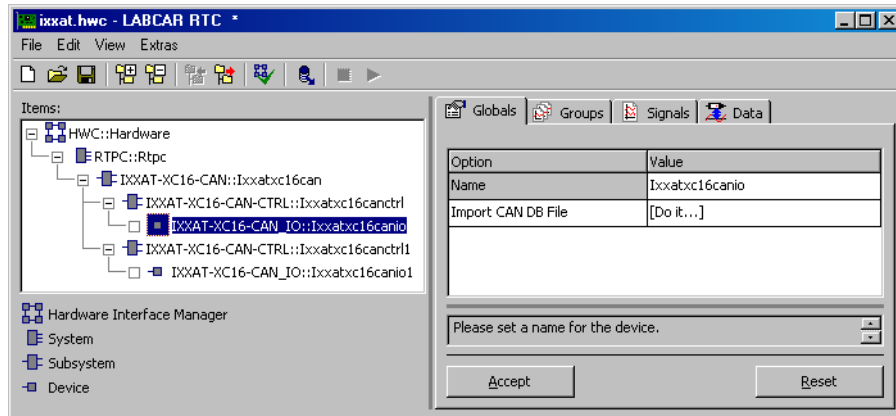
---

Informationen zu dieser Option finden Sie im Datenblatt des CAN-Controllers.

## 27.3 IXXAT-XC16-CAN\_IO Device

### 27.3.1 Globals (IXXAT-XC16-CAN\_IO Device)

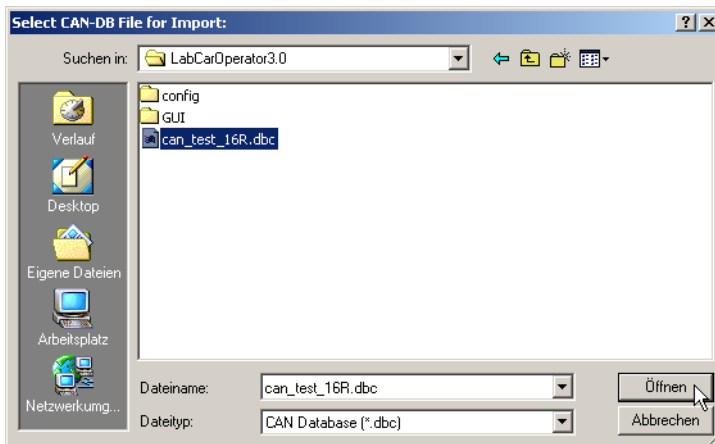
Abb. 27-5 zeigt die Registerkarte „Globals“ eines IXXAT-XC16-CAN\_IO Devices.



**Abb. 27-5** Die Registerkarte „Globals“ des IXXAT-XC16-CAN\_IO Devices  
*Import CAN DB File*

Über diese Option kann eine CAN-Datenbank-Datei eingelesen werden, welche mit dem CANdb Datenverwaltungsprogramm der Firma Vector Informatik erstellt wurde. Mit Hilfe dieser Datei können nun bei Bedarf automatisch CAN-Botschaften und -Signale angelegt werden.

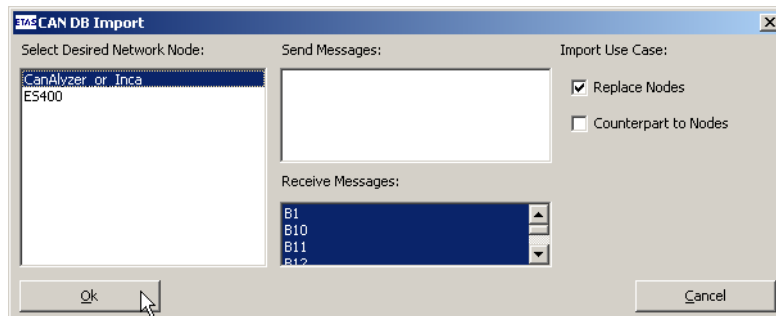
Nach Anklicken von [Do it...] öffnet sich folgendes Dialogfenster:



**Abb. 27-6** CANdb Auswahl

In diesem Dialogfenster kann die gewünschte zu importierende CAN DB-Datei ausgewählt werden.

Nach Betätigen der Schaltfläche **Open** öffnet sich nachfolgendes Dialogfenster, in dem das weitere Verfahren spezifiziert werden kann.

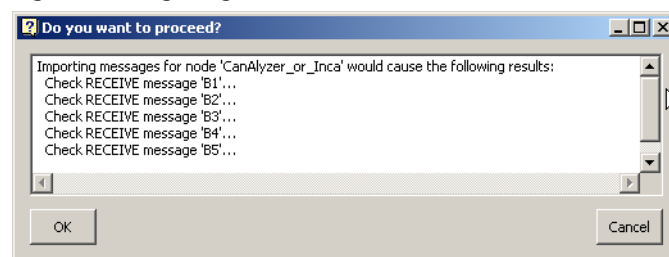


**Abb. 27-7** CANdb Import Dialog

In einer CAN DB-Datei werden üblicherweise mehrere Knoten eines CAN-Netzwerkes beschrieben. Alle vorhandenen Knoten werden in der linken Liste (Network Nodes) aufgelistet. Die beiden Listen rechts davon (Send Messages/Receive Messages) listen alle für den aktuell selektierten Knoten definierten CAN Botschaften auf. Die Botschaften, die selektiert sind, werden für den Import herangezogen. Zusätzlich kann der „Import Use Case“ noch ausgewählt werden.

Die Option „Replace Nodes“ bedeutet, dass das IXXAT-XC 16-CAN\_IO Device die Rolle des Netzwerkknotens übernimmt, d.h. eine Sendebotschaft des Knotens wird auch als Sendebotschaft des IXXAT-XC 16-CAN\_IO Devices umgesetzt usw. Die Option „Counterpart to Nodes“ bedeutet, dass das IXXAT-XC 16-CAN\_IO Device das Gegenstück zum Netzwerkknoten bildet; d.h. eine Sendebotschaft des Knotens ruft eine Receive-Botschaft hervor usw.

Nach Drücken der Schaltfläche **OK** erfolgt eine Überprüfung der zu importierenden Daten (CAN-Messages und -Signale) mit den vorhandenen Signalgruppen und Signalen. Das Resultat dieser Überprüfung wird anschließend in einem Dialogfenster angezeigt.



**Abb. 27-8** CANdb Import Check

Bis jetzt ist an dem vorhandenen IXXAT-XC 16-CAN\_IO Device noch nichts verändert worden. Erst nach Drücken von **OK** wird der tatsächliche Importvorgang gestartet. Der Importvorgang fügt die importierten Messages den Signalgruppen hinzu und sorgt auch für die Definition der entsprechenden Signale.

Nach Beendigung des Imports erscheint im Fenster „Monitor“ das detaillierte Protokoll des Importvorgangs.

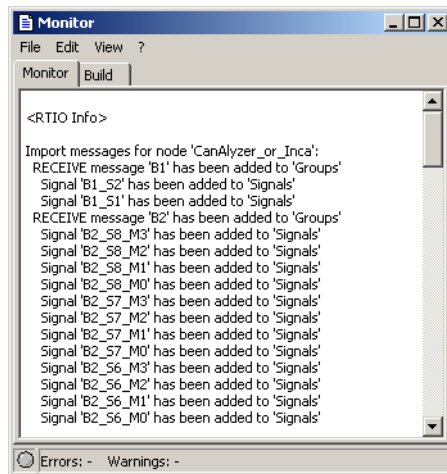


Abb. 27-9 CANdb Importprotokoll

### Hinweis

Beim Importieren einer CAN DB-Datei werden nur 29 Bit (Identifizier „extended“) bzw. 11 Bit (Identifizier „standard“) automatisch in das Identifizier-Feld eingefügt. Wenn im CAN-CTRL Subsystem der Identifizier „standard“ gewählt wurde, die CAN DB-Datei aber Signale mit 29 Bit-Identifiern (ID > 231) enthält, geschieht Folgendes:

- 11 Bit (Bits [28... 18]) werden automatisch in das Identifizier-Feld geschrieben. Die übrigen Bits (MSB) werden verworfen.
- Im Monitorfenster werden Warnungen angezeigt. Da nur ein Identifizier ausgewählt werden kann, kommt es bei CAN DB-Dateien, die sowohl „standard“ als auch „extended“ Identifizier enthalten, zu Konflikten.

### 27.3.2 Groups (IXXAT-XC16-CAN\_IO Device)

In dieser Registerkarte werden die CAN-Botschaften in Form von Signalgruppen spezifiziert.

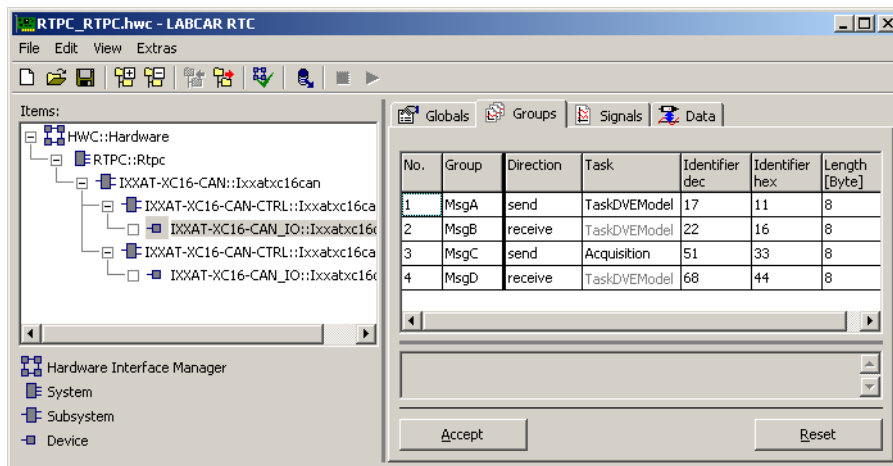
### Hinweis

Neue Signalgruppen bzw. CAN-Botschaften kann man über das Kontextmenü im Register „Groups“ anlegen.

### Hinweis

Bei Änderung des Signalgruppennamens, des Signalnamens oder der Signalrichtung „Direction“ wird die eventuell zugeordnete Message u.U. nicht mehr automatisch zugeordnet und muss manuell neu zugeordnet werden.

Abb. 27-10 zeigt die Registerkarte „Groups“ eines IXXAT-XC16-CAN\_IO Devices.



**Abb. 27-10** Die Registerkarte „Groups“ des IXXAT-XC16-CAN\_IO Devices

27.3.3 Die CAN-Botschaften als Signalgruppen

*Direction*

Hier kann die Richtung der CAN-Botschaft bestimmt werden („send“ = Sendebotschaft, „receive“ = Empfangsbotschaft).

*Task*

Bei Sendebotschaften wird hier die Task angegeben, in welcher die Botschaft gesendet werden soll.

Alle Empfangsbotschaften werden in einer gemeinsamen Task verarbeitet, eine individuelle Einstellung ist hier nicht möglich. Die gemeinsame Task für die Empfangsbotschaften wird hier grau dargestellt; sie kann im IXXAT-XC16-CAN-CTRL Subsystem ausgewählt werden.

*Identifier (dec / hex)*

Hier muss der Botschafts-Identifier eingegeben werden. Je nach gewählter Identifiertyp-Einstellung in dem darüberliegenden IXXAT-XC16-CAN-CTRL Subsystem („standard“ oder „extended“) kann hier der Wert unterschiedlich groß werden:

Jede Signalgruppe bzw. CAN-Botschaft muss einen anderen Identifier besitzen.

- „Standard“ Identifier: 11 Bit 2 047 dec 7 FF hex
- „Extended“ Identifier: 29 Bit 536 870 911 dec F FF FF FF hex

*Length [Byte]*

Legt die Anzahl der Nutzdatenbytes, welche die betreffende CAN-Botschaft übertragen kann, fest (1 bis 8 Bytes).

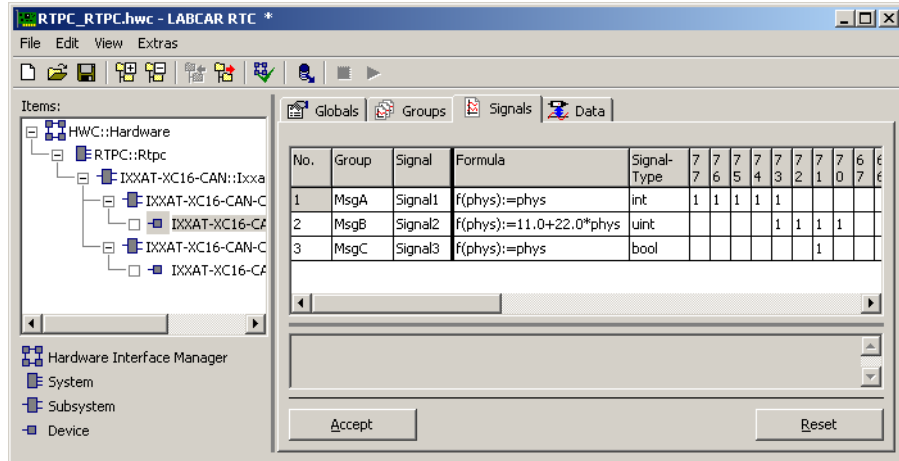
27.3.4 Signals (IXXAT-XC16-CAN\_IO Device)

In der Registerkarte „Signals“ werden die CAN-Botschaften weiter spezifiziert.

**Hinweis**

Neue CAN-Signale lassen sich über das Kontextmenü im Register „Signals“ anlegen.

Abb. 27-11 zeigt die Registerkarte „Signals“ des IXXAT-XC16-CAN\_IO Devices.



**Abb. 27-11** Die Registerkarte „Signals“ des IXXAT-XC16-CAN\_IO Devices

*Group*

Hier erfolgt die Zuordnung eines Signals zu der gewünschten Signalgruppe.

*Signal Type*

Hier wird der Signaltyp festgelegt, wie das Signal über den CAN-Bus übertragen wird.

Tab. 27-1 zeigt einen Überblick über die möglichen Signaltypen:

Signaltyp	Datentyp
int	Kennzeichnet ein vorzeichenbehaftetes Signal im Standard Zwe-Komplement Datenformat (max. 32 Bit)
(s) int	Kennzeichnet ein vorzeichenbehaftetes Signal, in dem als höchstwertiges Bit das Vorzeichen und anschließend der Absolutwert des Signals übertragen wird. Ist das Vorzeichen-Bit gesetzt, so handelt es sich dabei um eine negative Zahl (max. 32 Bit)
uint	Kennzeichnet ein vorzeichenloses Signal (max. 32 Bit)
bool	Kennzeichnet ein Boolesches Signal. Hierbei darf in der Bit-Matrix nur ein einzelnes Bit markiert sein.
real	Kennzeichnet ein Fließkommawert im „Standard IEEE Float (4 Byte)“ Format. Hierbei dürfen entsprechend nur 32 Bit in der Bit-Matrix markiert sein. Aufbau des Datentyps nach IEEE: - Sign: 1 Bit - Exponent: 8 Bit - Fraction: 24 Bit

**Tab. 27-1** Signaltypen

*Bitmatrix*

Eine CAN-Botschaft kann bis zu 8 Datenbytes übertragen. In einer Bitmatrix kann für jedes Signal festgelegt werden, welche Bits dieses Signal benötigt bzw. belegt.

Der Aufbau der Spalten ist wie folgt:

7	7	...	0	0	Byte-Nummer
7	6		1	0	Bit-Nummer

**Bedeutung der Bit-Felder:**

- Leeres Feld  
Das entsprechende Signal verwendet das Bit nicht
- Belegtes Feld  
Das Signal benötigt dieses Bit an dieser Position
- „x“-Feld  
Das entsprechende Bit steht zur Datenübertragung nicht zur Verfügung, weil das Signal weniger Nutzdatenbytes hat (siehe Einstellung „Length“ im Register „Groups“).

**Bedienung der Bit-Felder:**

Über die Pfeiltasten der Tastatur kann die gewünschte Bit-Zelle ausgewählt werden. Ein Anklicken mit der Maus „toggelt“ die entsprechende Zelle zwischen „unbelegt“ und „1“. Wird während des Klickens zusätzlich noch die <Alt>-Taste gedrückt, so wird der Wert „1“ bis „9“ inkrementiert (Blockbildung).



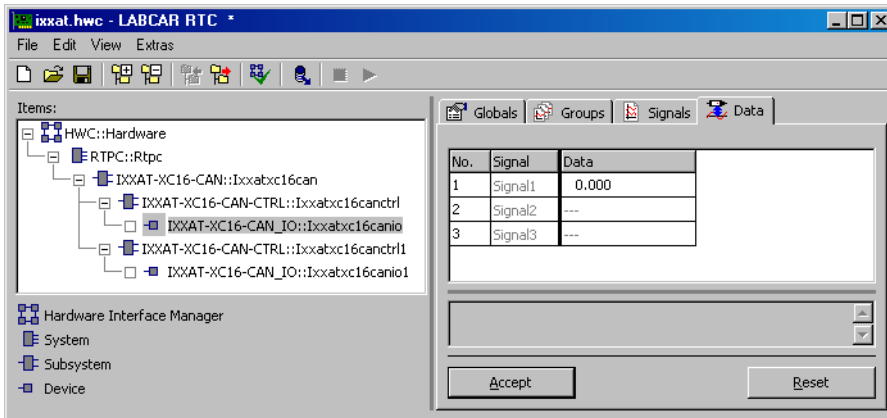


16 Bit Signal im Motorola Format:

No.	Group	Signal	3	3	3	3	3	3	2	2	2	2	2	2	1	1	1	1	1	1	0	0	0	0	0	0	0
			6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6
1	Group1	Signal1													1	1	1	1	1	1	1	1	1	1	1	1	

27.3.5 Data (IXXAT-XC16-CAN\_IO Device)

Abb. 27-12 zeigt die Registerkarte „Data“ eines IXXAT-XC16-CAN\_IO Device Devices.



**Abb. 27-12** Die Registerkarte „Data“ des IXXAT-XC16-CAN\_IO Devices  
 Man erkennt in der Registerkarte „Data“ (Abb. 27-12 - „offline“ Modus), dass das Signal „Signal1“ dazu verwendet wird, um eine CAN-Botschaft zu versenden. Im „Data“-Feld können im „online“ Modus die Nutzdaten belegt werden. Im „offline“ Modus sind die „Data“-Felder der Empfangsbotschaften „Signal2“ und „Signal3“ nicht aktiv, im „online“ Modus werden dort die empfangenen Nutzdaten der entsprechenden CAN-Botschaften dargestellt.

## 28 Das RTIO-Package für Transputer-basierte Hardware

---

Zur Anbindung von Transputer-basierter LabCar2-Hardware an LABCAR-OPERATOR stehen zwei hardwaretechnische Realisierungen zur Verfügung.

Dies ist zum einen das ES4205-System. Dieses System besteht aus einem VMEbus Chassis, welches in die LabCar2-Signalbox eingebaut wird und fünf VMEbus Steckplätze bietet, sowie einer Einschubkarte, die die Verbindung zwischen dem parallelen VMEbus und dem seriellen Transputer-Link-Bus der LabCar2-Hardware herstellt.

### *ES4205 VME64x Integration - ES1206 Serial Interface Board (T8VME)*

---

Die ES4205 Einschubkarte besteht aus einem ES1206 Serial Interface Board (T8VME), das im VMEbus-Teil des Chassis steckt, und einem auf der ES1206 aufgesteckten Piggyback-Modul, welches an das Transputer-Link-System angeschlossen ist. Die Kopplung der beiden Bussysteme erfolgt durch Firmware, die auf dem Transputer der ES1206 läuft. Mit dem ES4205-System lässt sich eine LabCar2 Signalbox an LABCAR-OPERATOR anbinden.

### *ES1206 - ES1381 - ES4206*

---

Die zweite hardwaretechnische Realisierung wird verwendet, wenn mehr als eine LabCar2 Signalbox angesteuert werden soll. VMEbus-seitig besteht die Hardware aus einem ES1206 Serial Interface Board (T8VME) mit aufgestecktem ES1381 Serial Interface Piggyback. Über ein spezielles Transputer-Link-Kabel, das 5 Transputer-Links führt, wird die ES1381 mit einem ES4206 Serial Interface Board in der LabCar2 Signalbox verbunden. Die ES1206/ES1381-Karte wird in einen Standard VMEbus Baugruppenträger wie z. B. das ES4100-System eingebaut. Für jede anzubindende LabCar2 Signalbox ist eine separate ES1206/ES1381-Karte zu verwenden. Die Kopplung der Bussysteme erfolgt wie beim ES4205-System durch Firmware auf dem Transputer der ES1206.

Das RTIO Package enthält alle benötigten Treiber, um die Karten einer LabCar2 Signal- oder Lastbox mit LABCAR-OPERATOR anzusteuern.

### *Inhalt dieses Kapitels*

---

Dieses Kapitel beschreibt die Einstellmöglichkeiten des RTIO-Editors für die ES4205- bzw. die ES1206/ES1381-Karten und der Transputer-basierten LabCar2 Hardware. Informationen zur Funktion der einzelnen Karten finden Sie in der jeweiligen Hardware-Dokumentation. Im Einzelnen werden die folgenden Karten beschrieben:

- ES4205 oder ES1206-ES1381-ES4206 (auf Seite 920)
- MTS-UBATT und UBATT-POW (auf Seite 921)
- TS-PWM- und arbiträre Signalgenerierung (auf Seite 929)
- TMIO (auf Seite 962)
- TDAC (auf Seite 985)
- TRS422 (auf Seite 993)
- T8Module TMIO (auf Seite 994)
- T8IO (auf Seite 995)
- Load20A14C (auf Seite 996)

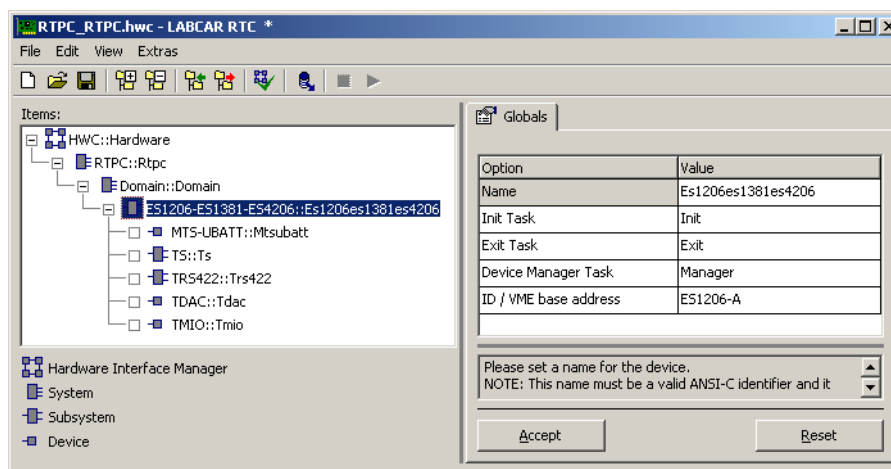
In diesem Kapitel werden nur die Einstellungen beschrieben, die spezifisch für die einzelnen Karten sind. Die Einstellungen, die für alle Karten die gleiche Bedeutung haben, werden im Abschnitt „Konfigurationsregister“ auf Seite 42 beschrieben.

## 28.1 ES4205 und ES1206/ES1381 - Buskopplung

Im RTIO Editor erfolgt die Beschreibung bzw. Einbindung von LabCar2-Hardware durch Zuordnung eines ES4205- oder ES1206-ES1381-ES4206-Subsystems zu einem ES1130-System.

### 28.1.1 Globals (ES4205 oder ES1206-ES1381-ES4206 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des Items beschrieben.



**Abb. 28-1** Die Registerkarte „Globals“ des ES4205 / ES1206-ES1381-ES4206 Subsystems

#### *Device Manager Task*

Geben Sie hier eine Task an, in der die Systemdienste der ES4205 bzw. der ES1206 abgearbeitet werden.

Diese Systemdienste bestehen in der Anzeige von Fehler-, Warn- oder Informationsmeldungen, die von den LabCar2-Karten an den Host-PC gesendet werden und für den Anwender gedacht sind. Diese Meldungen werden zunächst in einen Puffer auf der ES4205 bzw. der ES1206 geschrieben. Aus diesem Puffer werden die Meldungen vom der ES1130 in der Device Manager Task gelesen und an den Host-PC gesendet. Damit der Puffer auf der ES4205 bzw. der ES1206 nicht überläuft und Meldungen verlorengehen, sollte die Aktivierungszeit der Device Manager Task kleiner oder gleich 10 ms betragen.

#### *ID / VME base address*

In diesem Feld wird die VMEbus Basisadresse der ES1206-Karten angegeben. Diese Basisadresse ist in den EPLDs dieser Karten codiert und kann nur von ETAS abgeändert werden. Bitte beachten Sie, dass dieses Feld bei ES4205-Karten deaktiviert ist, da diese generell nur eine VMEbus Basisadresse unterstützen.

## 28.2 MTS-UBATT/UBATT-POW - Batteriesteuerung

---

Die Kombination aus MTS-UBATT-Karte und UBATT-POW-Karte erfüllt im LABCAR primär drei Aufgaben:

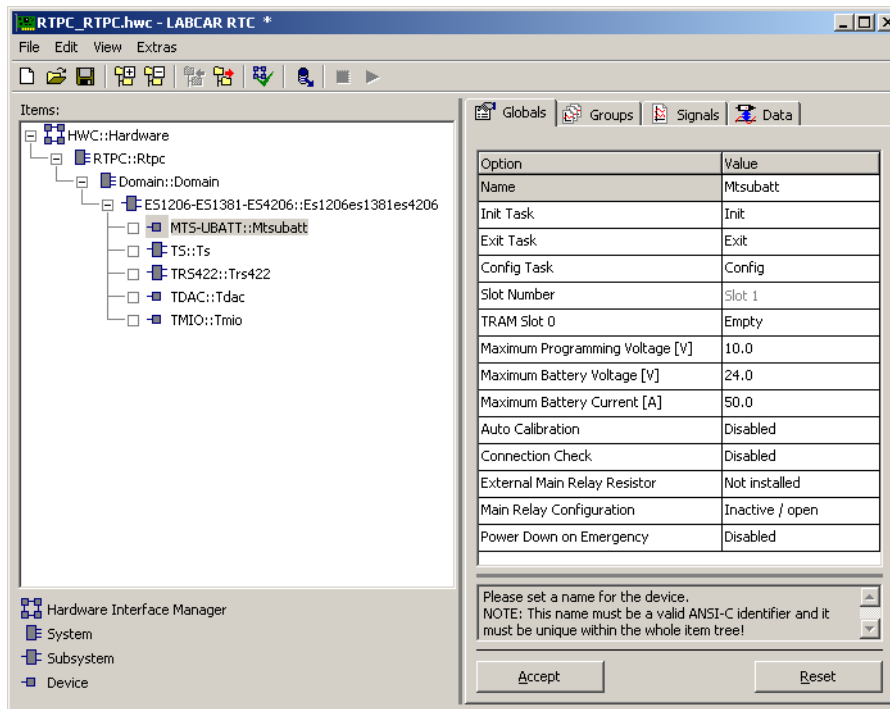
- Trägereinheit für Prozessormodule  
Im Gegensatz zur früheren Software-Generation wird die Berechnung von Modellteilen auf der MTS-UBATT-Karte von LABCAR-OPERATOR nicht mehr unterstützt. Das gesamte Modell wird auf der ES1130 Prozessorkarte oder dem Real-Time PC gerechnet
- Steuerung eines extern an das LABCAR angeschlossenen Netzteils zur Emulation der Batteriespannung
- Nachbildung von „Batterie-Knoten“, also Leistungsschaltern, über die das Steuergerät, die LABCAR-I/O-Hardware und die Lasten mit der Batteriespannung versorgt werden können

Beide Karten bilden zusammen eine Einheit, sind jedoch aus schaltungstechnischen Gründen auf zwei Platinen verteilt. Sowohl die MTS-UBATT-Karte als auch die UBATT-POW-Karte können jeweils nur einmal im LABCAR eingesetzt werden.

Darüber hinaus kommunizieren beide Karten über eine speziell für das LABCAR entwickelte Schnittstelle miteinander - aus diesem Grund kann die UBATT-POW-Karte ausschließlich von der MTS-UBATT-Karte gesteuert werden. Der Einsatz der UBATT-POW-Karte ohne MTS-UBATT-Karte ist nicht möglich. Aus diesem Grund existiert auch kein RTIO-Item für die UBATT-POW-Karte, alle Einstellungen erfolgen über die MTS-UBATT-Karte.

## 28.2.1 Globals (MTS-UBATT Device)

In diesem Abschnitt werden die globalen Einstellungen des MTS-UBATT Devices beschrieben.



**Abb. 28-2** Die Registerkarte „Globals“ des MTS-UBATT Devices

### *Slot Number*

Geben Sie hier den Slot der LabCar2-Signalbox an, in dem die MTS-UBATT-Karte steckt. Die Slots werden von links nach rechts (beginnend mit 1) durchnummeriert. Eine MTS-UBATT-Karte kann nur im ersten Slot einer LabCar2 Signalbox stecken.

### *TRAM Slot 0*

Die MTS-UBATT-Karte bietet Steckplätze für die Aufnahme von bis zu acht TRAM-Modulen (Module, die aus einem Transputer und zugehörigem Hauptspeicher bestehen). Geben Sie hier an, ob der Steckplatz 0 mit einem TRAM-Modul bestückt ist (Option „TRAM Module“) oder nicht (Option „Empty“).

Diese Information wird vom Boot-Algorithmus des LabCar2-Systems benötigt. Dieser Boot-Algorithmus lädt auf Transputern, die keine I/O-Funktionalität ausführen, ein spezielles Programm, den sogenannten Router, das Nachrichten im seriellen LabCar2-System weiterleitet. Daher müssen im Slot 0 der MTS-UBATT-Karte installierte TRAM-Module angegeben werden. TRAM-Module, die in den Slots 1- 7 der MTS-UBATT-Karte installiert sind, müssen nicht angegeben werden, da der Boot-Algorithmus diese Slots durch Setzen einer Link-Verbindung überbrückt.

### *Maximum Programming Voltage [V]*

---

Im Laborauto-Umfeld wird zur Emulation der Batterie ein Netzteil eingesetzt. Solche Netzteile werden fast ausnahmslos auch in Versionen angeboten, die sich elektronisch ansteuern lassen. In der Regel werden hierzu analoge Schnittstellen verwendet. Über eine von außen angelegte „Steuerspannung“ kann dann beispielsweise die Ausgangsspannung des Netzteils eingestellt werden.

Geben Sie hier die maximale Ansteuerspannung (in Volt) des externen Netzteils an. Die Steuerung des Netzteils durch das Modell erfolgt durch die absolute Angabe einer Sollspannung und eines Maximalstroms. Die analoge Schnittstelle der Netzteile ermöglicht in der Regel aber nur eine relative Einstellung (z.B. dient eine Steuerspannung zwischen 0 V und 10 V der linearen Regelung der Ausgangsspannung auf Werte zwischen 0 V und der Maximalspannung des Gerätes).

Um die erforderlichen Umrechnungen der absoluten in relative Werte vornehmen zu können, müssen die Angaben zur maximalen Ausgangsspannung, dem maximalen Ausgangsstrom und der maximalen Ansteuerspannung des externen Netzteils eingetragen werden.

### *Maximum Battery Voltage [V]*

---

Maximale Ausgangsspannung des externen Netzteils in Volt.

Siehe auch „Maximum Programming Voltage [V]“ auf Seite 923.

### *Maximum Battery Current [A]*

---

Maximaler Ausgangsstrom des externen Netzteils in Ampere.

Siehe auch „Maximum Programming Voltage [V]“ auf Seite 923.

### *Auto Calibration*

---

Die MTS-UBATT-Karte verfügt über einen Mechanismus zur Autokalibrierung der analogen Schnittstellen zum und vom externen Netzteil. Jede ausgelieferte Karte ist bereits fertig kalibriert. Sollte jedoch einmal eine Ungenauigkeit in der Einstellung des externen Netztes festgestellt werden, so kann eine erneute Autokalibrierung durch Setzen der Option auf „Enabled“ aktiviert werden. Die eigentliche Kalibrierung wird jedoch erst beim nächsten Initialisieren der Karte durchgeführt.

### *Connection Check*

---

Zur Kommunikation mit der UBATT-POW-Karte verfügt die MTS-UBATT-Karte über eine spezielle serielle Hochgeschwindigkeits-Schnittstelle. Standardmäßig wird die Verbindung zwischen der MTS-UBATT-Karte und der UBATT-POW-Karte bei jeder Kommunikation zwischen den Karten auf korrekte Funktion hin überprüft. Wird hierbei ein Fehler entdeckt, so generiert das Modell eine entsprechende Fehlermeldung. Um dies in einem LABCAR, in dem keine UBATT-POW-Karte installiert ist, zu verhindern, kann der Fehlererkennungsmechanismus durch Setzen der Option auf „Disabled“ abgeschaltet werden.

### *External Main Relay Resistor*

---

Die UBATT-POW-Karte bietet dem Anwender die Möglichkeit, ein vorhandenes Steuergeräte-Signal zur Ansteuerung eines „Hauptrelais“ direkt in Hardware auszuwerten und in die Aktivierung bzw. Deaktivierung der einzelnen Batterieknoten einfließen zu lassen.

Hierzu verfügt die Karte über einen speziellen Eingang, der, softwaretechnisch gesteuert, entweder mit einer Pull-Up- oder einer Pull-Down-Schaltung versehen werden kann, um den Pegel des inaktiven Signalzustandes zu definieren.

Die Pull-Up- bzw. Pull-Down-Schaltung auf der MTS-UBATT-Einschubkarte ist als Konstantstromquelle mit einem Nennstrom von 160 mA realisiert. Um eine Überlastung des angeschlossenen Steuergerätes zu vermeiden, ist es möglich, die Pull-Up- bzw. Pull-Down-Funktion mit einer externen Beschaltung zu realisieren.

Diese Einstellung bestimmt, gegen welchen Batterieknoten der externe Widerstand der Hauptrelaisbeschaltung verdrahtet ist. Falls Sie die interne Pull-Up- bzw. Pull-Down-Schaltung verwenden, wählen Sie „Not installed“.

#### *Main Relay Configuration*

---

Hier kann die Beschaltung des Hauptrelaiseinganges der UBATT-POW-Karte spezifiziert werden. Wird hier „External pull down“ oder „Onboard pull down“ selektiert, so wird das Hauptrelais als aktiv erkannt, sobald sein Signalpegel oberhalb der halben Batteriespannung liegt. Wird „External pull up“ oder „Onboard pull up“ selektiert, so ist das Hauptrelais dann aktiv, wenn sein Signalpegel unterhalb der halben Batteriespannung liegt.

Selektieren Sie „Onboard pull down“ oder „Onboard pull up“ um die interne Konstantstromquelle für die Hauptrelaisbeschaltung auszuwählen. Für die Auswahlen „External pull down“ oder „External pull up“ müssen Sie eine externe Beschaltung einfügen, um den Pegel am Ausgang des Steuergerätes festzulegen.

Die Auswahl „Inactive / open“ deaktiviert jegliche Hauptrelais-Erkennung, d.h., es wird immer als inaktiv gewertet.

#### *Power Down on Emergency*

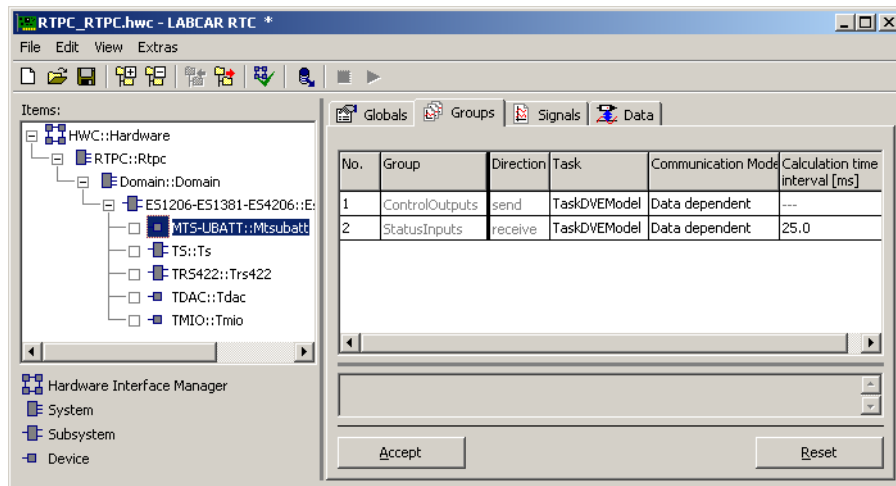
---

Dieses Feld legt fest, wie die Software auf ein aktives „Emergency“ Signal reagiert. Wird „Disabled“ eingestellt, so werden bei einem aktiven *Emergency* Signal alle Batterieknotenschalter geöffnet. Der Wert der Batteriespannung bleibt jedoch unverändert. Wird *Enabled* eingestellt, so wird zusätzlich auch noch die Batteriespannung auf 0 gesetzt, d.h., die Energieversorgung des LABCAR wird abgeschaltet. Das „Emergency“ Signal ist ein Eingangssignal des RTIO-Treibers der Karte.



## 28.2.2 Groups (MTS-UBATT Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des MTS-UBATT Devices beschrieben.



**Abb. 28-3** Die Registerkarte „Groups“ des MTS-UBATT Devices

Für die MTS-UBATT-Karte gibt es zwei Signalgruppen:

- **ControlOutputs**  
Werden vom Modell gesteuert und verändern die Einstellungen der MTS-UBATT-Karte. Über diese Signale werden die Batterieknoten geschaltet sowie die Batteriespannung und der Batteriestrom gesetzt.
- **StatusInputs**  
Beinhalten Informationen zum Hardware-Status.  
Sie dienen der Erfassung von Batteriespannung und -strom sowie der Schalt- und Fehlerzustände der Batterieknoten und des Hauptrelais.

### *Communication Mode*

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der MTS-UBATT-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

### *Calculation time interval [ms]*

Auf dem On-Board-Transputer der MTS-UBATT-Karte läuft ein Prozess, der die Statusinformation der Batterieknoten und des Hauptrelais ermittelt sowie Batteriespannung und -strom misst. In diesem Feld wird das Zeitraster angegeben, in dem dieser Prozess gerechnet bzw. aktiviert wird.

28.2.3 Signals (MTS-UBATT Device)

In diesem Abschnitt werden die Signale der MTS-UBATT-Karte sowie deren spezielle Einstellungen beschrieben.

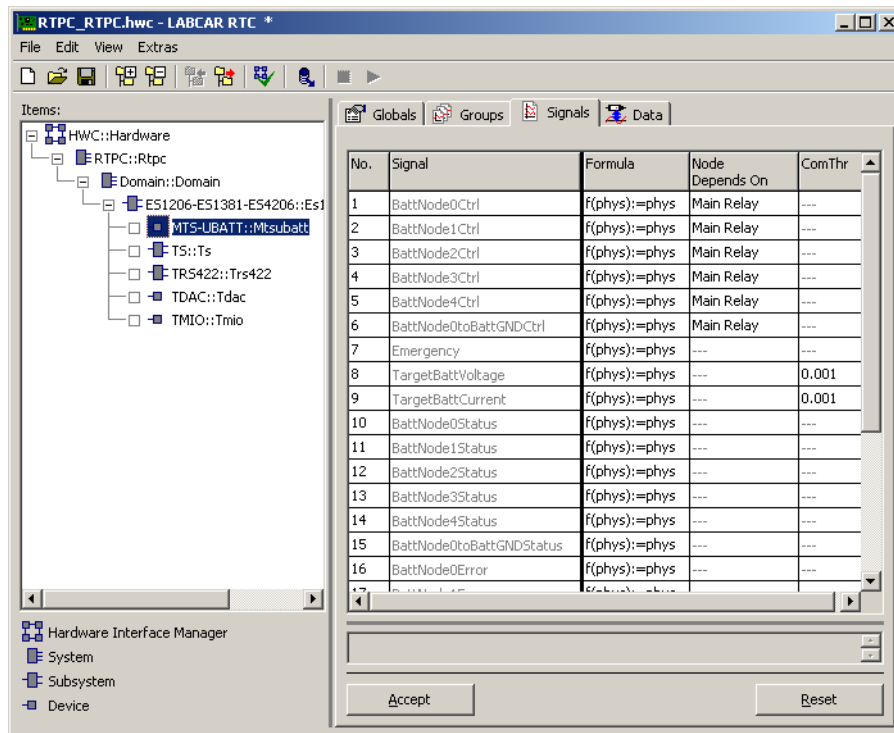


Abb. 28-4 Die Registerkarte „Signals“ des MTS-UBATT Devices

Die Datentypen und die Bedeutung der verschiedenen Werte der einzelnen Signale werden im Abschnitt „Datentypen und Wertebereiche“ auf Seite 928 beschrieben.

Signal	Richtung	Bedeutung
BattNodenCtrl	send	Schaltet den Batterieknoten <i>n</i> gegen Batteriespannung. Ist ein Knoten <i>n</i> derart konfiguriert, dass er <b>nicht</b> vom Zustand des Hauptrelais abhängt, so gibt <i>BattNodenCtrl</i> an, ob er geschaltet ( $\neq 0$ ) oder geöffnet ( $= 0$ ) werden soll. Hängt der Knoten hingegen vom Hauptrelaiszustand ab, so hat das Signal <i>BattNodenCtrl</i> keine Bedeutung.
BattNode0to-BattGNDCtrl	send	Schaltet den Batterieknoten 0 gegen die Batteriemasse. Wird der Batterieknoten 0 vom Hauptrelaiszustand gesteuert, so hat das Signal <i>BattNode0toBattGNDCtrl</i> keine Bedeutung.
Emergency	send	Signal für die Notabschaltung aller Batterieknoten und Deaktivierung des Hauptrelais. Steht die Option „Power Down on Emergency“ auf „enabled“, so wird zusätzlich die Batteriespannung zu 0 gesetzt.

Signal	Richtung	Bedeutung
TargetBattVoltage	send	Sollwert für die Batteriespannung
TargetBattCurrent	send	Maximaler Sollstrom
BattNodenStatus	receive	Schaltzustand des jeweiligen Batterieknotens
BattNode0to-BattGNDStatus	receive	Zeigt an, ob Batterieknoten 0 gegen Masse geschaltet ist.
BattNodenError	receive	Das Signal stellt dem Modell Statusinformationen des Batterieknoten-Schalters $n$ zur Verfügung. Liefert das Signal einen Wert $\neq 0$ , so wurde Knoten $n$ aufgrund eines Fehlerzustandes (wie z.B. Kurzschluss) abgeschaltet. Bei fehlender Last ist der Signalwert auch $\neq 0$ , aber der Knoten ist <i>nicht</i> abgeschaltet.
BattNode0to-BattGNDError	receive	Stellt dem Modell Statusinformationen des Batterieknoten-Schalters 0 nach Masse zur Verfügung (siehe auch Bedeutung von <i>BattNodenError</i> ).
MainRelayStatus	receive	Zustand des Hauptrelais
MainRelayError	receive	Zeigt an, ob in der Beschaltung des Hauptrelais-Signals ein Fehler aufgetreten ist.
ActualBattVoltage	receive	Istwert der Batteriespannung in Volt
ActualBattCurrent	receive	Istwert des Batteriestroms in Ampere

#### *Node Depends On*

---

Diese Einstellung legt fest, ob der jeweilige Batterieknoten vom Modell (Einstellung „Input“) aus gesteuert wird oder vom Zustand des Hauptrelais (Einstellung „Main Relay“) abhängt.

#### *ComThr*

---

Diese Einstellung ist nur von Bedeutung, wenn im Feld „Communication Mode“ der zugehörigen Signalgruppe die Einstellung „Data dependent“ gewählt wurde. In diesem Fall gibt „ComThr“ an, um wieviel sich das zugehörige Signal bezogen auf seinen maximalen Wert geändert haben muss, damit ein Datentransfer zwischen dem Simulationstarget und der MTS-UBATT-Karte stattfindet. Hat man z.B. ein Spannungssignal mit dem Maximalwert 10 V und gibt man für die Kommunikationsschwelle `ComThr` einen Wert von 0.01 an, so muss sich das Spannungssignal um 1%, d.h. 100 mV ändern, damit ein Datentransfer stattfindet.

## 28.2.4 Datentypen und Wertebereiche

In diesem Abschnitt werden die möglichen Datentypen sowie die Bedeutung der verschiedenen Werte der einzelnen Signale beschrieben.

Signal	Datentypen	Mögliche Werte
BattNodenCtrl	int	≠ 0 : Batterieknoten $n$ wird an die Batteriespannung angeschlossen 0 : Batterieknoten $n$ wird von der Batteriespannung abgetrennt
BattNode0to-BattGNDCtrl	int	≠ 0 : Batterieknoten 0 wird an die Batteriemasse angeschlossen 0 : Batterieknoten 0 wird von der Batteriemasse abgetrennt
Emergency	int	≠ 0 : Notfallbetrieb ist aktiviert 0 : Notfallbetrieb ist deaktiviert
TargetBattVoltage	real32	Sollwert der Batteriespannung in Volt
TargetBattCurrent	real32	Maximaler Sollstrom in Ampere
BattNodenStatus	beliebiger ganzzahliger Datentyp	≠ 0 : Batterieknoten $n$ ist an die Batteriespannung angeschlossen 0 : Batterieknoten $n$ ist von der Batteriespannung abgetrennt
BattNode0to-BattGNDCtrl	beliebiger ganzzahliger Datentyp	≠ 0 : Batterieknoten 0 ist an die Batteriemasse angeschlossen 0 : Batterieknoten 0 ist von der Batteriemasse abgetrennt
BattNodenError	beliebiger ganzzahliger Datentyp	≠ 0 : Fehler im Batterieknoten $n$ 0 : Kein Fehler im Batterieknoten $n$
BattNode0to-BattGNDCtrl	beliebiger ganzzahliger Datentyp	≠ 0 : Fehler am Schalter des Batterieknotens 0 nach Masse 0 : Kein Fehler
MainRelayStatus	beliebiger ganzzahliger Datentyp	0 : Hauptrelais ist inaktiv ≠ 0 : Hauptrelais ist aktiv
MainRelayError	beliebiger ganzzahliger Datentyp	≠ 0 : Fehler im Hauptrelais 0: Kein Fehler im Hauptrelais
ActualBattVoltage	real32, real64	Istwert der Batteriespannung in Volt
ActualBattCurrent	real32, real64	Istwert des Batteriestroms in Ampere

### 28.3 TS-PWM- und arbiträre Signalgenerierung

---

Die TS-Karte (Transputer-Signalgenerator-Karte) ist eine LABCAR-Stimulationskarte, die zur Erzeugung von PWM- und frei programmierbaren Signalen mit einer Abtastrate kleiner oder gleich 1 MHz eingesetzt werden kann.

Die Karte beinhaltet zwei Gruppen von Signalgeneratoren:

- Frei programmierbare Signalgeneratoren (arbiträre Signalgeneratoren)  
Dem Anwender stehen insgesamt drei arbiträre Signalgeneratoren zur Verfügung. Jeder Signalgenerator beinhaltet zwei Kanäle, die über die gleiche Steuerlogik synchron zueinander ausgegeben werden.
- Pulsweitenmodulierte Signalgeneratoren (PWM Signalgenerierung)  
Die zweite Generatorgruppe beinhaltet vier PWM-Generatoren. Die Signalverläufe werden bei diesem Signaltyp durch eine einstellbare Periode und ein Tastverhältnis bestimmt. Die PWM-Signale werden asynchron zu den arbiträren Signalen erzeugt.

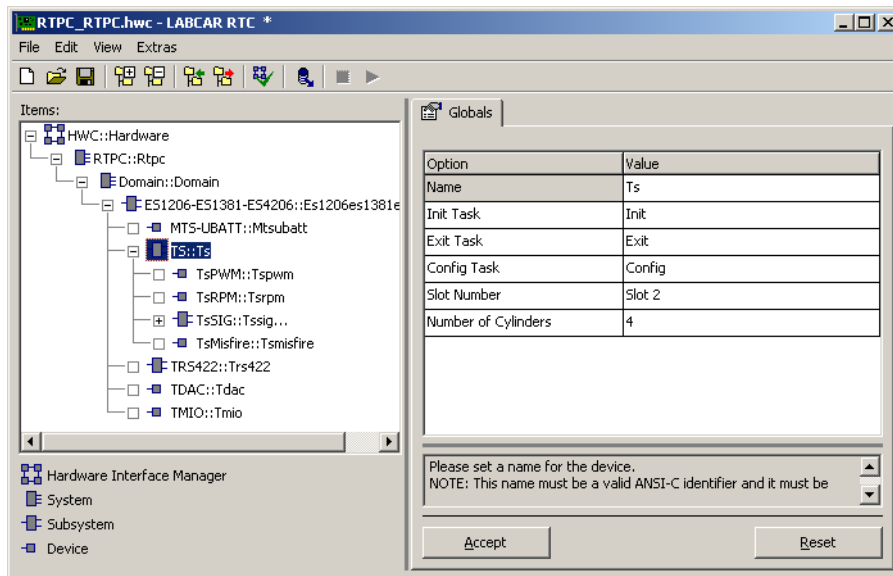
Ein zentrales Drehzahlmodul ergänzt den Funktionsumfang der Karte. Dieses Modul wird zur Erzeugung von drehzahlsynchronen Signalen verwendet und stellt somit eine wesentliche Grundfunktion für die Stimulation von Motorsteuergeräten dar. Das Drehzahlmodul wird zur Steuerung der arbiträren Signalgeneratoren und der digitalen Signalerfassung mit der TMIO-Karte verwendet.

Die TS-Karte wird zur Erzeugung folgender Fahrzeugsignale verwendet:

- Generierung drehzahlsynchroner Signale mit arbiträren Signalgeneratoren
  - Drehzahlsignal (wird vom Motorsteuergerät zur Erfassung der Drehzahl und des Kurbelwellenwinkels eingesetzt.)
  - Nockenwellensignal (das Nockenwellensignal wird vom Steuergerät zur Zylindererkennung benötigt)
  - Klopfsignal (das Klopfsignal wird für die Simulation des Klopfensors verwendet)
- Generierung pulswertenmodulierter Signale mit PWM Signalgeneratoren
  - Raddrehzahlsignal (Simulation der Fahrzeuggeschwindigkeit)
  - Pedalwertgebersignal (Simulation des Gas-, Kupplungs-, Bremspedals)

28.3.1 Globals (TS Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des TS Subsystems beschrieben.



**Abb. 28-5** Die Registerkarte „Globals“ des TS Subsystems

Für die PWM-Signalgenerierung, die arbiträre Signalgenerierung und die Drehzahl-signalgenerierung existieren jeweils separate RTIO-Items, die als Unter-Items dem TS-Item zugeordnet werden können und die jeweils über eigene globale Einstellungen verfügen.

*Slot Number*

Geben Sie hier den Slot der LabCar2 Signalbox an, in dem die TS-Karte steckt. Die Slots werden von links nach rechts beginnend mit eins durchnummeriert.

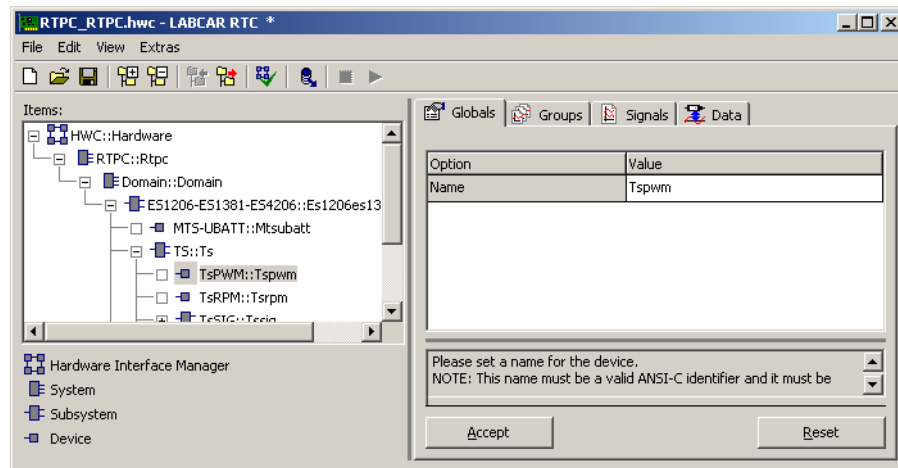
*Number of Cylinders*

Sowohl der Fehlzündungsgenerator als auch der Klopf-generator benötigen die Angabe der Anzahl der vorhandenen Zylinder.

RTIO Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Number of Cylinders	uint8	ja	Der Fehlzündungsgenerator unterstützt 1 bis 12 Zylinder. Der Kopfgenerator unterstützt 1 bis 8 Zylinder.

### 28.3.2 Globals (TsPWM Device)

In diesem Abschnitt werden die globalen Einstellungen des TsPWM Devices beschrieben.

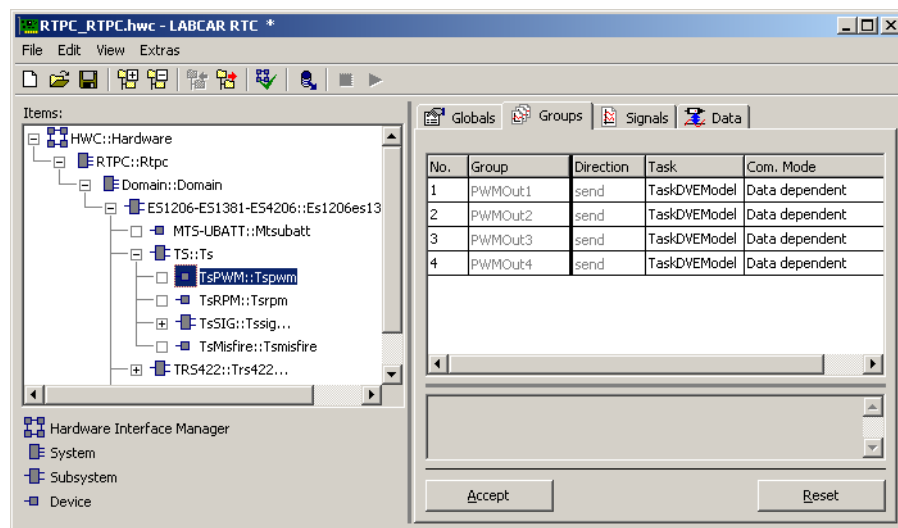


**Abb. 28-6** Die Registerkarte „Globals“ des TsPWM Devices

Über das TsPWM-Device werden die vier PWM-Signalgeneratoren der TS-Karte konfiguriert. Die Konfiguration ist für jeden Generator individuell und kann zur Laufzeit aus dem Modell heraus geändert werden.

### 28.3.3 Groups (TsPWM Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TsPWM Devices beschrieben.



**Abb. 28-7** Die Registerkarte „Groups“ des TsPWM Devices

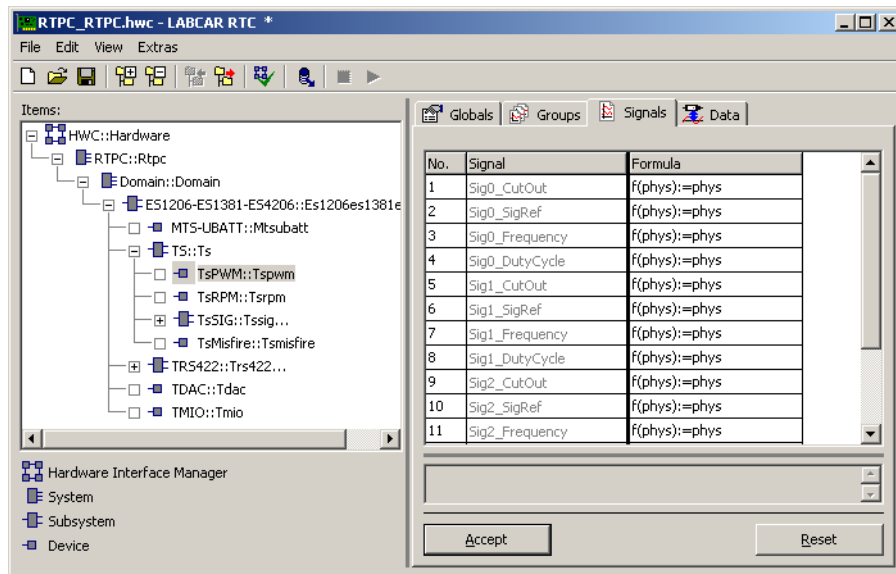
Das TsPWM-Device verfügt über vier Signalgruppen, die jeweils die Einstellungen für einen der vier PWM-Generatoren steuern. Alle Signale haben die Richtung *send*, werden also vom Simulationstarget zur TS-Karte übertragen.

### Com. Mode

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der TS-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

### 28.3.4 Signals (TsPWM Device)

In diesem Abschnitt werden die Signale des TsPWM Devices und deren Einstellungen beschrieben.



**Abb. 28-8** Die Registerkarte „Signals“ des TsPWM Devices

Jeder PWM-Generator kann durch Ändern der folgenden vier Parameter konfiguriert werden.

#### *Sign\_Frequency*

Legt die Frequenz des PWM-Signals  $n$  fest.

#### *Sign\_DutyCycle*

Legt das Tastverhältnis des PWM-Signals  $n$  fest. Das Tastverhältnis ist der Anteil des High-Pegels an der Periodendauer des PWM-Signals. Das Tastverhältnis wird als Faktor der Periodendauer angegeben, z.B. ein Faktor 0,5 entspricht der halben Periodendauer.

#### *Sign\_CutOut*

Schaltet das PWM-Signal  $n$  ein oder aus.

#### *Sign\_SigRef*

Legt die Ausgangs-Endstufe des PWM-Signals fest. Es besteht die Auswahl zwischen einer Open-Collector Endstufe und einer Endstufe mit internem  $2\text{ k}\Omega$  Kollektorwiderstand gegen Batterieknoten 4. Wird die interne Endstufe gewählt, so



muss zur PWM-Ausgabe Batterieknoten 4 eingeschaltet sein. Für jedes PWM-Signal gibt es einen eigenen Eingang für die externe Masse, welcher für ein korrektes Funktionieren des PWM-Generators immer beschaltet sein muss.

### 28.3.5 Datentypen und Wertebereiche

In diesem Abschnitt werden die möglichen Datentypen für die einzelnen Signale sowie die Bedeutung der verschiedenen Signalwerte beschrieben.

Signal	Datentypen	Mögliche Werte
Sign_Frequency	float	0.0 bis 10000.0 [Hz]
Sign_DutyCycle	float	0.0 bis 1.0
Sign_CutOut	bool	false = PWM-Ausgang abgeschaltet true = PWM-Ausgang eingeschaltet
Sign_SigRef	bool	false = Endstufe mit internem 2 kΩ Pull-Up-Widerstand zur Batteriespannung true = Open Collector Endstufe

### 28.3.6 TsRPM Device

Mit dem TsRPM Device der TS-Karte wird der Drehzahl-Signal-Generator konfiguriert. Der Drehzahl-Signal-Generator wird zur Stimulation (Takt und Trigger) der arbiträren Signalgeneratoren verwendet. Er stellt das Taktsignal, das Triggersignal und den Kurbelwellenwinkel bereit.

#### *Taktsignal (Drehzahlgenerator-Ausgang)*

Für eine Motorperiode wird eine Speicherbank mit 8192 Stützstellen ausgetaktet. Eine Motorperiode beim Viertaktmotor umfasst eine Nockenwellenumdrehung bzw. zwei Kurbelwellenumdrehungen. Da die Motordrehzahl die Kurbelwellenfrequenz angibt, hat das Taktsignal die (8192/2)-fache Frequenz der Motordrehzahl (n):

$$f_{\text{Takt}} = n_{\text{Motor}} \cdot \frac{\text{Speicherbankgrösse}}{2}$$

**Gln. 28-1** Allgemeine Berechnung des Drehzahltaktes

Wird die Speicherbankgröße mit 8192 in die Gln. 28-1 eingesetzt und die Taktfrequenz in Hertz umgerechnet, so erhält man folgenden Zusammenhang:

$$\frac{f_{\text{Takt}}}{\text{Hz}} = 68,2667 \cdot \frac{n_{\text{Motor}}}{\text{U/min}}$$

**Gln. 28-2** Drehzahltakt für 8K-Stützstellen, TS-Karte

Die maximale Taktfrequenz beträgt 1,2 MHz, d.h. die Maximaldrehzahl entspricht 17578 U/min.

#### **Hinweis**

*Wird die Speicherbankgröße reduziert (z.B. 2K oder 4K), so wird die maximale Motordrehzahl erhöht. Die Speicherbankreduzierung hat jedoch eine Verschlechterung der Winkelauflösung zur Folge. Wenn Sie die Speicherbankgröße ändern möchten, nehmen Sie bitte Kontakt mit dem LABCAR-Anwenderbetreuer auf.*

#### Kurbelwellenwinkel

Die Winkelauflösung  $\alpha$  beträgt allgemein in Abhängigkeit von der Speicherbankgröße:

$$\alpha = \frac{720^\circ \text{ KW}}{\text{Speicherbankgroesse}}$$

**Gln. 28-3** Winkelauflösung  $\alpha$  in Abhängigkeit von der Speicherbankgröße ( $^\circ$  KW = Grad-Kurbelwelle)

Für 8K-Stützstellen ergibt sich:

$$\alpha = \frac{720^\circ \text{ Kw}}{8192} = 0,0879^\circ \text{ Kw}$$

**Gln. 28-4** Winkelauflösung bei 8K-Stützstellen

Der Kurbelwellenwinkel  $\Gamma$  ergibt sich aus dem aktuell ausgetakteten Stützpunkt  $n$  und der Winkelauflösung  $\alpha$ :

$$\Gamma = n \cdot \alpha$$

**Gln. 28-5** Kurbelwellenwinkel  $\Gamma$

#### Triggersignal (Drehzahlgenerator-Ausgang)

Alle 8192 Stützstellen wird vom Drehzahlgenerator ein Triggersignal mit Pulsbreite einer Taktperiode generiert. Dieser Triggerpuls hat einen Neubeginn einer Motorperiode zur Folge.

Das Triggersignal besitzt folgende Frequenz:

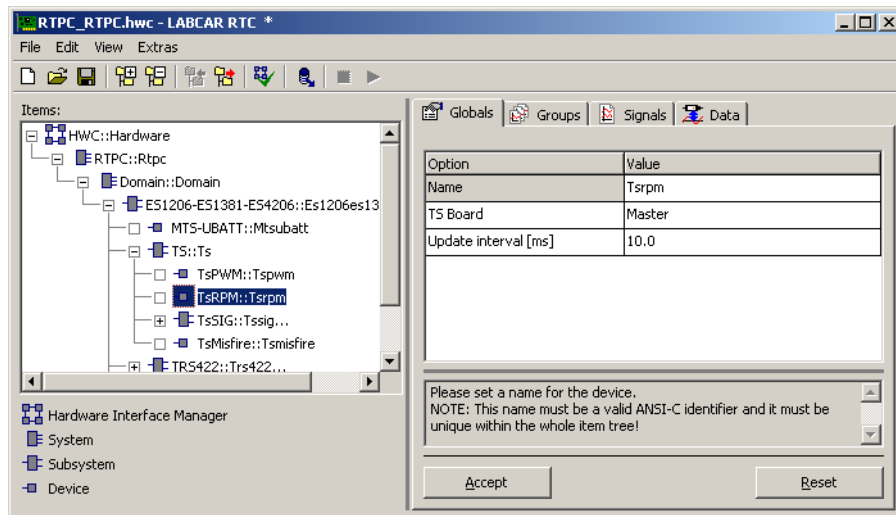
$$f_{\text{Trig}} = \frac{f_{\text{Takt}}}{\text{Speicherbankgroesse}}$$

**Gln. 28-6** Triggerfrequenz für drehzahlsynchrone Signale, TS-Karte

Das Triggersignal wird dem LABCAR-Anwender über eine BNC-Buchse (TFO) auf der TS-Kartenfrontseite für externe Triggerzwecke zur Verfügung gestellt. Alle drehzahlsynchronen Abläufe innerhalb des LABCAR werden mit diesem sogenannten „Master-Trigger“ synchronisiert.

## 28.3.7 Globals (TsRPM Device)

In diesem Abschnitt werden die globalen Einstellungen des TsRPM Devices und deren Einstellungen beschrieben.



**Abb. 28-9** Die Registerkarte „Globals“ des TsRPM Devices

#### *TS Board*

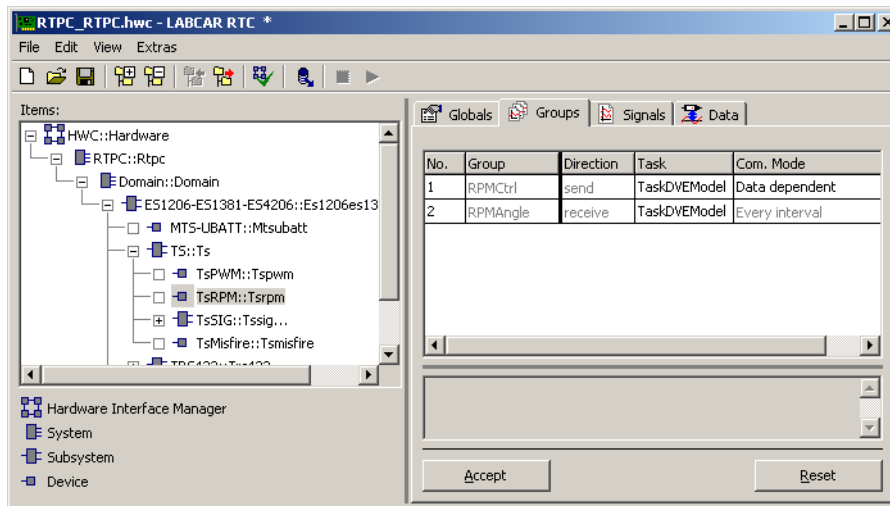
Legt fest, ob die TS-Karte als Master oder Slave betrieben wird. Wenn mehrere TS-Karten in einer Signalbox betrieben werden, muss immer eine als Master definiert sein. Die Master-Karte gibt das Drehzahlsignal vor, alle anderen TS-Karten werden über die Backplane mit diesem Signal synchronisiert.

#### *Update Interval [ms]*

Das Intervall in Millisekunden, in dem der aktuelle Kurbelwellenwinkel an das Modell zurückgegeben wird.

### 28.3.8 Groups (TsRPM Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TsRPM Devices beschrieben.



**Abb. 28-10** Die Registerkarte „Groups“ des TsRPM Devices

Das TsRPM-Device verfügt über zwei Signalgruppen:

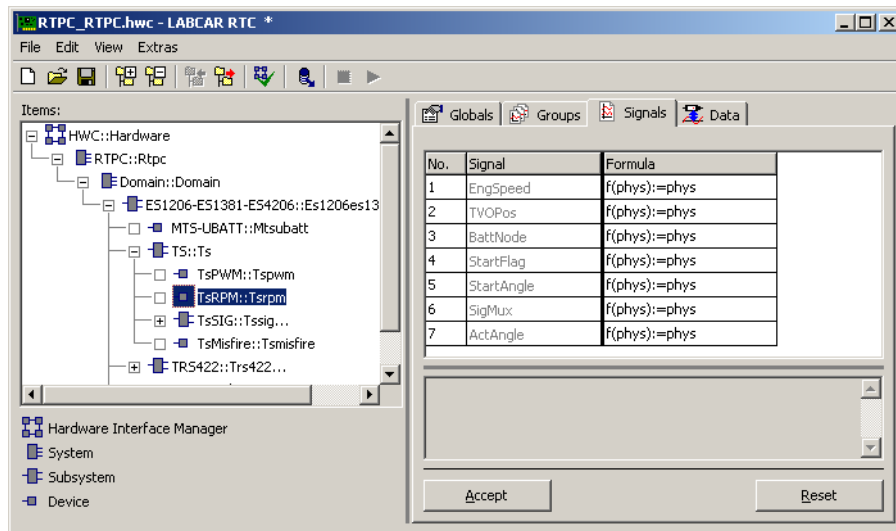
- RPMCtrl  
Beinhaltet alle Signale, die zur Konfiguration des Drehzahlgenerators dienen. Diese Gruppe sollte der Task zugeordnet werden, die im Modell die aktuelle Drehzahl berechnet.
- RPMAngle  
Beinhaltet das Signal, welches den Kurbelwellenwinkel an das Modell zurückgibt. Das Rechenraster dieser Gruppe sollte dem Parameter „Update Interval [ms]“ entsprechen, um unnötige Datenübertragungen zu vermeiden.

#### *Com. Mode*

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der TS-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

## 28.3.9 Signals (TsRPM Device)

In diesem Abschnitt werden die Signale des TsRPM Devices und deren Einstellungen beschrieben.



**Abb. 28-11** Die Registerkarte „Signals“ des TsRPM Devices

Alle Signale des TsRPM-Device, bis auf das Signal *ActAngle*, dienen der Konfiguration der Karte.

Signal	Richtung	Bedeutung
EngSpeed	send	Die vom Modell vorgegebene Sollzahl in Umdrehungen pro Minute.
TVOPos	send	Der Offset in Grad für das Triggersignal TVO, relativ zum Triggersignal TFO. Der maximale Offset beträgt eine Motorperiode (720° KW).
BattNode	send	Schaltet den „Motor“ ein oder aus.
StartFlag	send	Schaltet das Anfahren eines definierten Kurbelwellenwinkels beim Ausschalten des Motors ein und aus.
StartAngle	send	Definiert einen Kurbelwellenwinkel in Grad, der beim Ausschalten des Motors angefahren wird, sofern dies mit dem Signal <i>StartFlag</i> aktiviert wurde.
SigMux	send	Legt den Kanal fest, der am Signal-Multiplexer Ausgang TSTO ausgegeben wird. Dies kann jeder beliebige Kanal der TS-Karte sein.
ActAngle	receive	Der aktuelle Kurbelwellenwinkel in Grad, der an das Modell zurückgegeben wird.

## 28.3.10 Datentypen und Wertebereiche

In diesem Abschnitt werden die Datentypen und Wertebereiche der einzelnen Signale sowie die Bedeutungen der verschiedenen Werte beschrieben.

Signal	Datentypen	Mögliche Werte
EngSpeed	float	0 bis 17578 (Maximaldrehzahl)
TVOPos	float	0 bis 720
BattNode	bool	true = Motor an false = Motor aus
StartFlag	bool	true = „StartAngle“-Funktion aktiviert false = „StartAngle“-Funktion nicht aktiviert
StartAngle	float	0 bis 720
SigMux	int	0 bis 15, Bedeutung siehe Tab. 28-1 auf Seite 938
ActAngle	float	0 bis 720

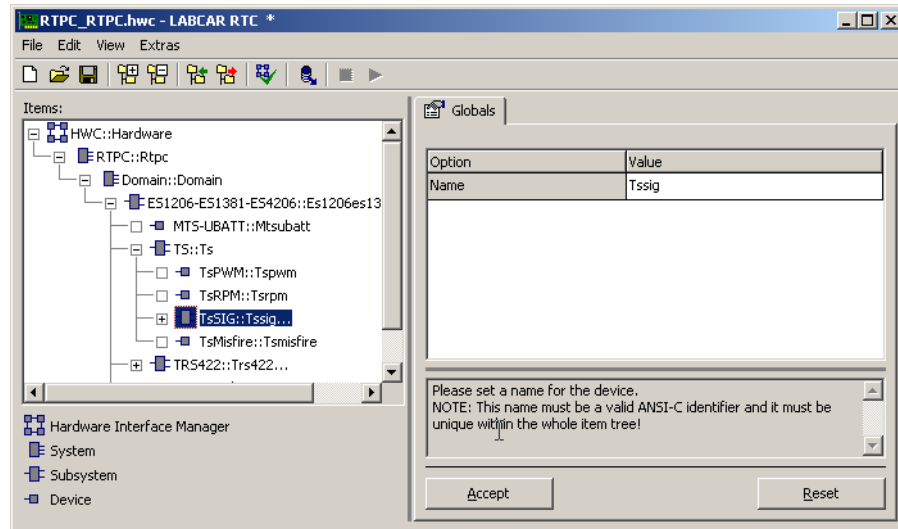
Das Signal *SigMux* kann die folgenden Werte annehmen:

0	Signal 1 (Signalgenerator 1)
1	Signal 2 (Signalgenerator 1)
2	Signal 1 (Signalgenerator 2)
3	Signal 2 (Signalgenerator 2)
4	Signal 1 (Signalgenerator 3)
5	Signal 2 (Signalgenerator 3)
6	PWM-Signal 1
7	PWM-Signal 2
8	PWM-Signal 3
9	PWM-Signal 4
10	Referenzsignal 1 (Signalgenerator 1)
11	Referenzsignal 2 (Signalgenerator 1)
12	Referenzsignal 1 (Signalgenerator 2)
13	Referenzsignal 2 (Signalgenerator 2)
14	Referenzsignal 1 (Signalgenerator 3)
15	Referenzsignal 2 (Signalgenerator 3)

**Tab. 28-1** Werte des „SigMux“ Signals

28.3.11 Globals (TsSIG Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des TsSIG Items beschrieben.

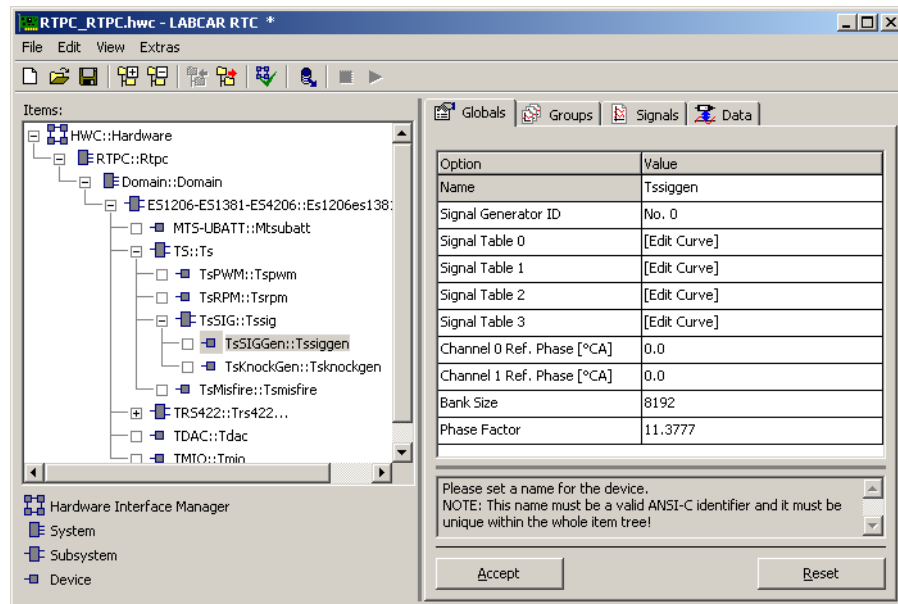


**Abb. 28-12** Die Registerkarte „Globals“ des TsSIG Subsystems

Das TsSIG-Subsystem beinhaltet bis zu drei TsSIGGen-Devices. Jedes dieser Devices ist für die Konfiguration eines der drei arbiträren Signalgeneratoren der TS-Karte zuständig.

28.3.12 Globals (TsSIGGen Device)

In diesem Abschnitt werden die globalen Einstellungen des TsSIGGEN Devices beschrieben.



**Abb. 28-13** Die Registerkarte „Globals“ des TsSIGGEN Devices

Das Register „Globals“ des TsSIGGen-Device beinhaltet die Konfigurationseinstellungen für jeweils einen der drei arbiträren Signalgeneratoren der TS-Karte. Diese können unabhängig voneinander in einem jeweils eigenen Device konfiguriert werden.

#### *Signal Generator ID*

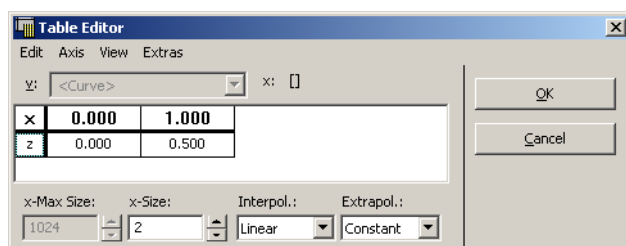
---

Mit dieser Option wird der zu konfigurierende Signalgenerator der TS-Karte ausgewählt.

#### *Signal Table 0-3*

---

Mit dieser Option können die Stützstellen für die jeweilige Signalbank definiert werden. Für jeden Signalgenerator stehen vier Signalbänke zur Verfügung. Durch Klicken auf „[Edit Curve]“ wird der Tabelleneditor geöffnet.



Nun können Sie eine beliebige Anzahl von Stützstellen (maximal 8192) definieren. Auf der x-Achse sind nur aufsteigende Werte zulässig. Der Wertebereich wird so skaliert, dass er auf den Bereich 0 bis 8191 abgebildet wird. Die zusätzlichen Stützstellen werden dabei linear interpoliert. Auf diese Weise brauchen nur die für den Signalverlauf notwendigen Stützstellen angegeben werden. Durch Klicken auf OK beenden Sie die Eingabe der Stützstellen.

Wenn die arbiträren Signalgeneratoren vom Drehzahl-Signal-Generator getriggert werden, wird bei jedem Inkrement des Zählers der Wert ausgegeben, der in der entsprechenden Stützstelle der ausgewählten Speicherbank des Signalgenerators hinterlegt ist. Auf diese Weise ist die Synchronisation des Kurbelwellensignals mit den Signalgeneratoren sichergestellt.

#### **Hinweis**

*Änderungen an den Signalverläufen werden erst mit dem nächsten Triggersignal in den Signalgeneratoren aktiv, um ein definiertes, kurbelwellensynchrones Umschalten der Signalform zu gewährleisten!*

#### *Channel 0 Ref. Phase*

---

Definiert eine Phasenverschiebung zwischen 0 und 720° für den Kanal 0 des Signalgenerators.

#### *Channel 1 Ref. Phase*

---

Definiert eine Phasenverschiebung zwischen 0 und 720° für den Kanal 1 des Signalgenerators.

#### *Bank Size*

---

Legt die Größe der Speicherbank für den Signalgenerator fest. Der Vorgabewert von 8192 sollte nicht verändert werden.

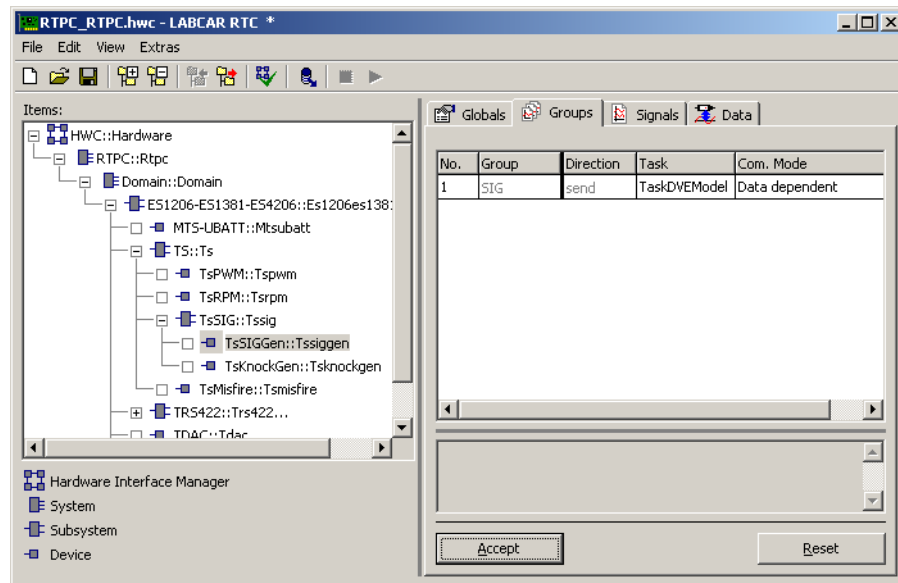


### Phase Factor

Legt die Anzahl der Taktperioden pro Grad Kurbelwellenwinkel fest. „Phase Factor“ mal 720 ergibt die Größe der Speicherbank. Der Vorgabewert von 11.38 sollte nicht verändert werden.

### 28.3.13 Groups (TsSIGGen Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TsSIGGen Devices beschrieben.



**Abb. 28-14** Die Registerkarte „Groups“ des TsSIGGEN Devices

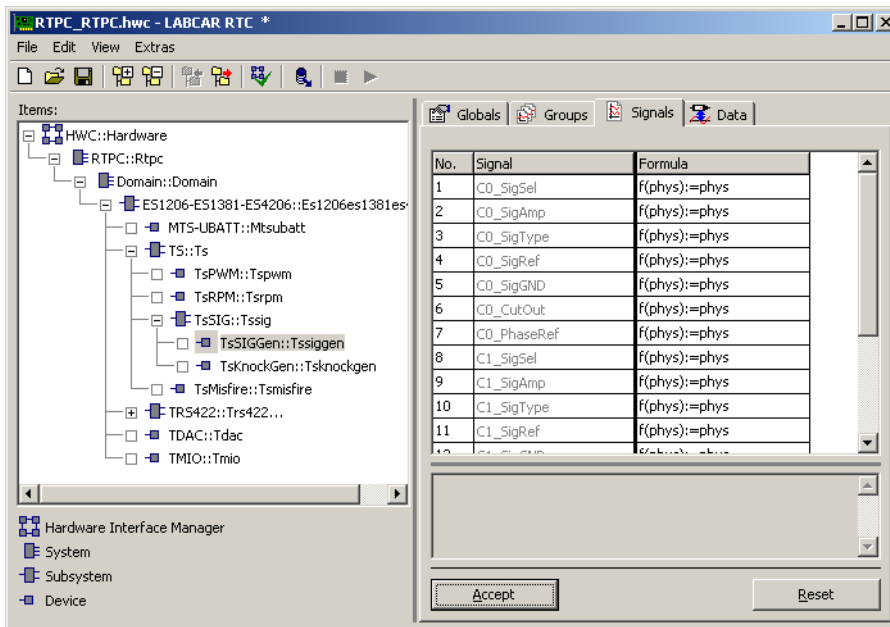
Für das TsSIGGen Device gibt es nur eine Signalgruppe „SIG“. Sie hat die Richtung *send*, d.h. sie wird vom Simulationstarget zur TS-Karte übertragen. Alle Signale der Signalgruppe dienen der Konfiguration des Generators.

#### Com. Mode

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der TS-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

## 28.3.14 Signals (TsSIGGen Device)

In diesem Abschnitt werden die Signale beschrieben, mit denen ein arbiträrer Signalgenerator konfiguriert wird.



**Abb. 28-15** Die Registerkarte „Signals“ des TsSIGGEN Devices

Das TsSIGGen Device wird mit den folgenden Signalen konfiguriert:

Signal	Richtung	Bedeutung
C0_SigSel	send	Legt fest, welche Signalbank am Kanal 0 ausgegeben wird
C0_SigAmp	send	Legt den Verstärkungsfaktor für den Analogausgang von Kanal 0 fest
C0_SigType	send	Auswahl der aktiven Endstufe
C0_SigRef	send	Umschaltung zwischen interner und externer Referenzspannung für Kanal 0
C0_SigGND	send	Auswahl zwischen Batteriemasse und externer Masse für Kanal 0
C0_CutOut	send	Schaltet den Signalgenerator ein und aus
C0_PhaseRef	send	Phasenverschiebung für Kanal 0. Diese Phasenverschiebung addiert sich zu der unter <i>Globals</i> definierten, ist aber, im Gegensatz zu dieser, zur Laufzeit verstellbar.
C1_...	send	Für Kanal 1 stehen jeweils die gleichen Signale wie für Kanal 0 zur Verfügung.
TrigSource	send	Legt die Triggerquelle für den Signalgenerator fest. Als Trigger können der Drehzahlgenerator, sowie verschiedene externe Quellen dienen.

Signal	Richtung	Bedeutung
ClkSource	send	Legt die Taktquelle für den Signalgenerator fest. Als Taktquelle können der Drehzahlgenerator, eine interne Taktquelle, eine 1 MHz Taktquelle, sowie verschiedene externe Quellen dienen.
LocalClkFreq	send	Wenn die interne Taktquelle ausgewählt ist, wird mit diesem Signal deren Frequenz eingestellt.
TrgMode	send	Legt den Triggermodus fest. <ul style="list-style-type: none"> <li>• Wait function (wait for trigger): Nachdem der letzte Datenpunkt ausgetaktet wurde, wird auf den nächsten Trigger gewartet. Solange wird der Pegel des letzten Datenpunktes ausgegeben.</li> <li>• Continuous function: Bei dieser Einstellung besitzt der Trigger eine Reset-Funktion. Mit jedem Trigger wird die Signalausgabe erneut gestartet. Falls kein Trigger anhängig ist, beginnt die Signalausgabe erneut nach Erreichen des letzten Datenpunktes.</li> </ul>

#### *Datentypen und Wertebereiche*

In diesem Abschnitt werden die Datentypen und Wertebereiche der einzelnen Signale sowie die Bedeutungen der verschiedenen Werte beschrieben. Für Kanal 0 (C0\_...) und Kanal 1 (C1\_...) stehen jeweils die gleichen Signale zur Verfügung. Die Optionen „TrigSource“, „ClckSource“ und „LocalClckFreq“ gelten für beide Kanäle.

Signal	Datentyp	Mögliche Werte
C0_SigSel	int	0 bis 3, entspricht Signalbank 0 bis 3
C0_SigAmp	float	0 bis 10 Volt bei interner Referenzspannung 0 bis 1 bei externer Referenzspannung
C0_Sig Type	bool	true = analog false = digital (open-collector switch)
C0_SigRef	bool	true = externe Referenzspannung false = interne Referenzspannung
C0_SigGND	bool	true = Batteriemasse false = externe Masse
C0_CutOut	bool	true = Kanal eingeschaltet false = Kanal hochohmig geschaltet
C0_PhaseRef	float	0 bis 720°
C1_...	wie Kanal 0	wie Kanal 0
TrigSource	int	0 bis 12, siehe Tab. 28-2 auf Seite 944

Signal	Datentyp	Mögliche Werte
ClckSource	int	0 bis 4, siehe Tab. 28-3 auf Seite 944
LocalClckFreq	int	0 Hz bis 780 kHz
TrgMode	int	= 0: continuous function ≠ 0: wait for trigger function

Die Einstellungen der Option „TrigSource“ haben die folgende Bedeutung:

0	Drehzahlgenerator der Master-Karte
1	externe Triggerquelle 0
2	externe Triggerquelle 1
3	externe Triggerquelle 2
4	externe Triggerquelle 3
5 - 9	nicht belegt
10	keine Triggerquelle
11	nicht belegt
12	ODER-Verknüpfung der externen Triggersignale

**Tab. 28-2** Einstellungen der Option „TrigSource“

Die Einstellungen der Option „ClckSource“ haben die folgende Bedeutung:

0	1 Mhz
1	lokale Taktquelle
2	externe Taktquelle 1
3	externe Taktquelle 2
4	Drehzahlgenerator der Master-Karte

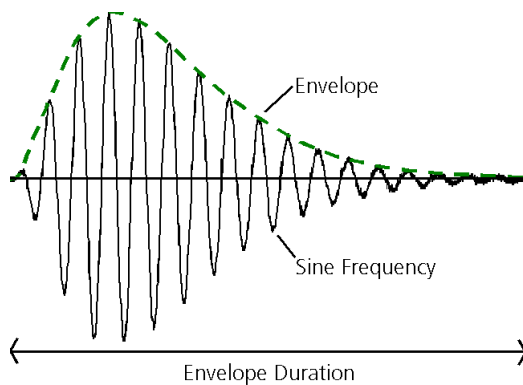
**Tab. 28-3** Einstellungen der Option „ClckSource“

### 28.3.15 TsKnockGen Device

Der Klopfgenerator wird zur Simulation von Signalen von Klopfensensoren verwendet. Eine unkontrollierte Verbrennung des Luft-Kraftstoff-Gemisches führt zu Druckschwankungen, die durch Körperschallsensoren aufgenommen werden. Das Motorsteuergerät kann Klopfen innerhalb eines bestimmten Frequenzbereiches (abhängig von der Amplitude der Schwingungen) erfassen.

Klopfen kann für jeden einzelnen Zylinder separat generiert werden, entweder nach einem Zufallsmuster oder nach einem festgelegten Muster. Dauer, Signalform und die Amplitudenmodulation (Hüllkurve) des Klopfsignals können frei definiert werden.

Abb. 28-16 zeigt ein Beispiel eines Klopfsignals („Klopfpaket“). Normalerweise ist dies ein hochfrequentes Sinussignal, das mit einer sich langsam verändernden Einhüllenden moduliert ist.



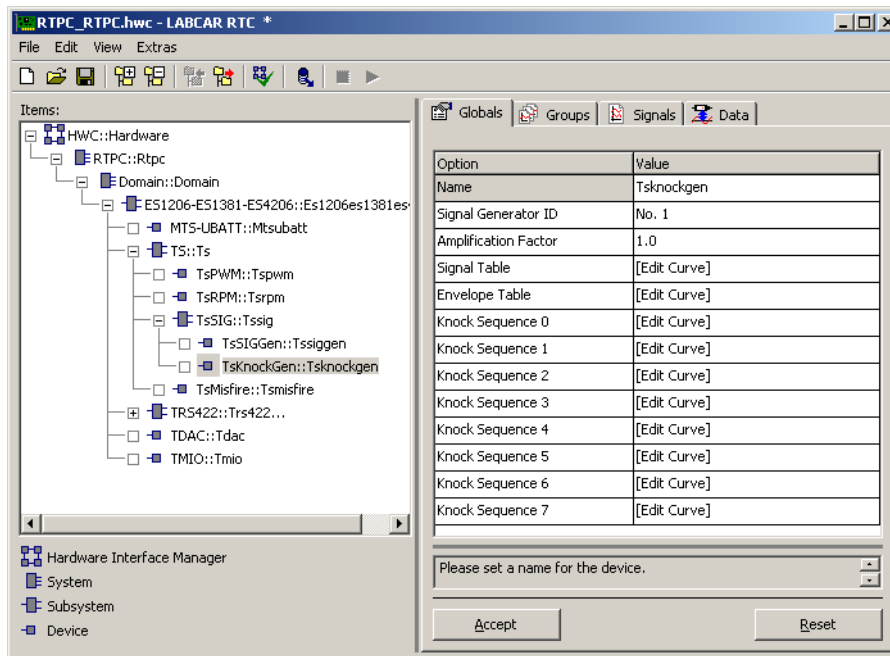
**Abb. 28-16** Klopfsignal (durchgezogen) und Einhüllende (gestrichelt)

#### **Hinweis**

Der Klopfgenerator benötigt die Angabe der Anzahl der vorhandenen Zylinder. Die Einstellung der Zylinderanzahl erfolgt in dem Register „Globals“ des TS Subsystems (siehe Abschnitt ss)

## 28.3.16 Globals (TsKnockGen Device)

In diesem Abschnitt werden die globalen Einstellungen des TsKnockGen Devices beschrieben.



**Abb. 28-17** Die Registerkarte „Globals“ des TsKnockGen Devices

#### *Signal Generator ID*

Die „Signal Generator ID“ legt fest, welche der drei TS Signalgenerierungskarten für die Klopfsignalerzeugung verwendet wird.

Dieser Parameter ist nicht editierbar im Laufzeitmodus.

#### *Amplification Factor*

Klopfsignale haben eine geringe Amplitude und sind damit bei der Übertragung zum Steuergerät sehr anfällig gegenüber Störungen. Um dies zu vermeiden, werden die Signale bei der Ausgabe an den Trennadapter verstärkt. Ein gewöhnlich im Trennadapter installierter Abschwächer kompensiert diese Verstärkung zur Weitergabe an das Steuergerät. Der „Amplification Factor“ muss damit gleich dem Abschwächungsfaktor des Abschwächers des Trennadapters gewählt werden.

Der Parameter „Amplification Factor“ ist im Laufzeitmodus editierbar.

#### *Signal Table*

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Definition der Signalform und der Periode des Klopfsignals. Die x-Achse spezifiziert die Zeit in  $\mu\text{s}$ . Der erste x-Wert muss 0.0 sein - die kleinste Periode des Klopfsignals beträgt 3  $\mu\text{s}$ . Die z-Achse legt die (normalisierte) Form des Klopfsignals fest - die z-Werte müssen zwischen -1.0 und 1.0 liegen.

Der Parameter „Signal Table“ ist im Laufzeitmodus editierbar.

### *Envelope Table*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Definition der Hüllkurve und der Dauer des Klopfsignals. Die x-Achse spezifiziert die Zeit in  $\mu\text{s}$ . Der x-Wert des ersten Punktes muss 0.0 sein. Die Dauer des Klopfsignals muss größer oder gleich der Periode des Klopfsignals sein. Die z-Achse spezifiziert die (normalisierte) Form der Hüllkurve - die z-Werte müssen zwischen 0.0 und 1.0 liegen

Der Parameter „Envelope Table“ ist im Laufzeitmodus editierbar.

### *Knock Sequence 0 ... Knock Sequence 7*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor, mit dem interaktiv das Muster für das Auftreten der Klopfsignale festgelegt werden kann. Ein Muster ist jeweils 100 Zündvorgänge lang. Die x-Achse hat keine Bedeutung und wird vernachlässigt. Ein z-Wert von 1 (Eins) bedeutet, dass für die aktuelle Zündung klopfen simuliert werden soll, während eine 0 (Null) für „kein Klopfen“ steht.

Die Parameter „Knock Sequence n“ sind im Laufzeitmodus editierbar.

In Tab. 28-4 sind die Eigenschaften der globalen RTIO-Parameters zusammengefasst.

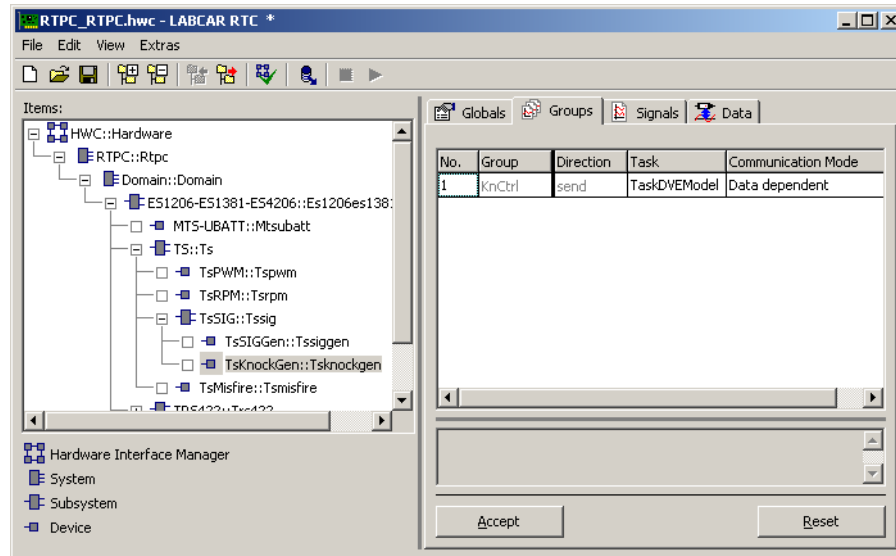
RTIO Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Signal Generator ID	uint32	Nein	TS Signalgenerierungskarte, die zur Klopfsignaerzeugung verwendet wird. Gültige Werte: 1, 2, 3
Amplification Factor	real32	Ja	Verstärkungsfaktor zur Kompensation der externen Signalabschwächung. Defaultwert: 1.0 (keine externe Abschwächung vorhanden) Gültiger Bereich: 0.1 ... 10.0
Signal Table	1-dim. Table	Ja	Normalisierte Klopfsignalform. x-Achse: Zeit in $\mu$ s z-Achse: -1.0 ... 1.0
Envelope Table	1-dim. Table	Ja	Normalisierte Hüllkurve des Klopfsignals x-Achse: Zeit in $\mu$ s Bereich für z-Werte: 0.0 ... 1.0
Knock Sequence 0 ... 7	1-dim. Table	Ja	Klopfmuster. Jedes Muster ist 100 Zündvorgänge lang. Ein z-Wert von 1 bedeutet, dass für die aktuelle Zündung Klopfen simuliert wird. Ein z-Wert von 0 bedeutet „kein Klopfen“ für die aktuelle Zündung. x-Werte werden ignoriert.

**Tab. 28-4** Globale Parameter des TsKnockGen Devices



28.3.17 Groups (TsKnockGen Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TsKnockGen Devices beschrieben.



**Abb. 28-18** Die Registerkarte „Groups“ des TsKnockGen Devices

*Communication Mode*

Der Kommunikationsmodus der Signalgruppe.

Option „Every interval“: Die Signalgruppe wird innerhalb eines jeden Intervalls (Task) gesendet.

Option „Data dependent“: Die Signalgruppe wird immer dann zur TS-Karte gesendet, wenn sich der Wert mindestens eines Signals geändert hat. Diese Option sollte bevorzugt verwendet werden.

Dieser Parameter ist im Laufzeitmodus editierbar.

Tab. 28-5 fasst die Eigenschaften der Signalgruppen-spezifischen RTIO-Parameter zusammen.

RTIO-Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Communication Mode	uint8	Ja	Kommunikationsmodus der Signalgruppe 0: Every interval (task period) 1: Data dependent

**Tab. 28-5** Signalgruppen-spezifische Parameter des TsKnockGen Devices

### 28.3.18 Signals (TsKnockGen Device)

In diesem Abschnitt werden die Signale beschrieben, mit denen der Klopfgenerator konfiguriert wird.

#### *C0\_SigRef, C1\_SigRef*

Die RTIO-Signale „C0\_SigRef“ und „C1\_SigRef“ legen die Quelle der Referenzspannung des D/A-Wandlers jeweils für Kanal 0 und 1 des Klopfgenerators fest. Wenn der Wert auf 0 gesetzt ist, wird die interne Referenzspannung (10 V) gewählt. Ist der Wert auf 1 gesetzt, wird die Referenzspannung des D/A-Wandlers extern eingespeist.

#### **Hinweis**

*Wenn Sie die interne Referenzspannung verwenden, beziehen sich die RTIO Signalamplituden „KnAmplSubLimit“, „KnAmplOverLimit“ und „KnNoiseAmpl“ auf eine Referenzspannung von 10 V - die Signalwerte sind auf einen Bereich von (0.0 ... 10.0)/Amplification Factor beschränkt.*

*Wenn Sie eine externe Referenzspannung verwenden, geben die RTIO Signalamplituden „KnAmplSubLimit“, „KnAmplOverLimit“ und „KnNoiseAmpl“ einen Faktor an, der mit der externen Referenzspannung multipliziert wird. Die Signalwerte sind auf einen Bereich von (0.0 ... 1.0)/Amplification Factor beschränkt.*

#### *C0\_SigGND, C1\_SigGND*

Die Signale „C0\_SigGND“ und „C1\_SigGND“ legen das Bezugspotential für die Klopfgeneratorausgänge 0 und 1 fest. Wird der Signalwert auf 0 (Null) gesetzt, wird die interne LABCAR-Masse verwendet. Wird der Signalwert auf 1 (Eins) gesetzt, wird ein extern bereitgestelltes Bezugspotential verwendet.

#### *C0\_CutOut, C1\_CutOut*

Die Signale „C0\_CutOut“ und „C1\_CutOut“ dienen zum An- und Abschalten des jeweiligen Ausgangskanals. Ein Signalwert von 0 schaltet den Ausgang des Kanals ab, ein Signalwert von 1 schaltet den Ausgang des Kanals an.

#### *KnEnable*

Das RTIO-Signal „KnEnable“ wird verwendet, um den Klopfgenerator an- und auszuschalten. Ein Signalwert von 0 schaltet den Klopfgenerator ab, ein Signalwert von 1 schaltet den Klopfgenerator an.

#### *KnTrigger*

Ein 0 → 1-Übergang des RTIO-Signals „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder).

Das Signal „KnTrigger“ ist nur aktiv, wenn der Klopfgenerator im Triggermodus „Single Shot“ betrieben wird. Der Triggermodus wird mit dem RTIO-Signal „KnTriggerMode“ eingestellt (siehe Seite 950).

#### *KnTriggerMode*

Das RTIO-Signal „KnTriggerMode“ wird zur Steuerung des Triggermodus des Klopfgenerators verwendet.

Wenn das Signal auf 0 gesetzt wird, arbeitet der Klopfgenerator im Triggermodus „Single Shot“. Ein 0 → 1-Übergang des RTIO-Signal „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder). Danach unterbleibt die Klopfgenerierung bis zum Auftreten eines erneuten Triggervorganges.

Wenn das Signal auf 1 gesetzt wird, arbeitet der Klopfgenerator im Modus „Continuous“. In diesem Modus ist die Klopfsimulation ohne eine Steuerung durch einen Trigger permanent aktiv. In diesem Triggermodus ist das RTIO-Signal „KnTrigger“ nicht aktiv.

#### *KnAmplSubLimit*

---

Amplitude des Klopfsignals bei kontrollierter Verbrennung. Das RTIO-Signal „KnAmplSubLimit“ legt die Amplitude des Klopfsignals fest, wenn für die aktuelle Zündung kein Klopfsignal generiert werden soll.

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

#### *KnAmplOverLimit*

---

Amplitude des Klopfsignals bei unkontrollierter Verbrennung. Das RTIO-Signal „KnAmplOverLimit“ legt die Amplitude des Klopfsignals fest, wenn für die aktuelle Zündung ein Klopfsignal generiert werden soll.

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

#### *KnNoiseAmpl*

---

Viele Steuergeräte interpretieren ein mehrfaches Auftreten von Klopfsignalen derselben Amplitude als Sensorstörung. Um dies zu verhindern, können die simulierten Klopfsignale über einen Zufallsgenerator amplitudenmoduliert werden.

Das RTIO-Signal „KnNoiseAmpl“ legt die Amplitude fest, mit der das Klopfsignal nach dem Zufallsprinzip moduliert wird. Dieses Signal wird verwendet, um statistischen Schwankungen bei der Verbrennung und damit bei den Signalen des Klopfensors zu simulieren.

Gültiger Bereich für die Amplitude:

Interne Bezugsspannung gewählt: (0.0 ... 10.0)/Amplification Factor.

Externe Bezugsspannung gewählt: (0.0 ... 1.0)/Amplification Factor

#### *KnStartAngle*

---

Das RTIO-Signal „KnStartAngle“ legt den Beginn des Klopfsignals relativ zum OT des jeweiligen Zylinders fest. Die relative Lage des OT wird in °KW angegeben. Ein positives Vorzeichen bedeutet, dass das Klopfsignal vor OT einsetzt, ein negatives, dass das Klopfsignal nach OT einsetzt.

Gültige Werte für den Kurbelwellenwinkel sind -720 °KW ... 720 °KW.

### *KnMode0 ... KnMode7*

---

Die RTIO-Signale „KnMode0“ bis „KnMode7“ legen die Betriebsart für die Klopfgenerierung im jeweiligen Zylinder fest.

„Disabled Mode“: Wenn das RTIO-Signal gleich 0 gesetzt wird, findet keine Klopfsignalgenerierung für den jeweiligen Zylinder statt.

„Pattern Mode“: Wenn das RTIO-Signal gleich 1 gesetzt wird, bestimmt die zugehörige Tabelle „KnockSequence“, ob ein Klopfsignal während der aktuellen Zündung erzeugt wird oder nicht.

„Random Mode“: Wenn das RTIO-Signal gleich 2 gesetzt wird, bestimmt ein Zufallsgenerator, ob ein Klopfsignal generiert wird oder nicht. Die Wahrscheinlichkeit der Klopfgenerierung wird über das entsprechende RTIO-Signal „KnProbability“ festgelegt.

### *KnProbability0 ... KnProbability7*

---

Mit den RTIO-Signalen „KnProbability0“ bis „KnProbability7“ wird die Wahrscheinlichkeit festgelegt, dass während der aktuellen Zündung eine unkontrollierte Verbrennung auftritt. Diese Signale sind aber nur dann aktiv, wenn das entsprechende „KnMode“ RTIO-Signal auf „Random Mode“ gesetzt ist.

Der gültige Wertebereich geht von 0.0 bis 1.0. Ein Wert von 0.0 bedeutet, dass eine unkontrollierte Verbrennung nie auftritt, ein Wert von 1.0 bedeutet, dass es bei jeder Zündung im entsprechenden zu einer unkontrollierten Verbrennung kommt.

#### **Hinweis**

*Die Indizes 0 bis 7 der RTIO-Signale „KnMode“ und „KnProbability“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*

Tab. 28-6 enthält eine Zusammenfassung aller Signale des TsKnockGen Device.

RTIO-Signale	Daten-typ	Erläuterung
C0_SigRef C1_SigRef	uint8	D/A-Wandler-Bezugsspannung für Kanal 1 und Kanal 2 des Klopfgeneratorsausgangs. 0: Interne 10 V Referenzspannung 1: Externe Bezugsspannung
C0_SigGND C1_SigGND	uint8	Bezugspotential für die Klopfgeneratorsausgang Kanal 1 und Kanal 2 0: interne LABCAR-Masse 1: Externe Masse
C0_CutOut C1_CutOut	uint8	An- und Abschalten des jeweiligen Ausgangskanals 1 und 2. 0: Ausgang abgeschaltet 1: Ausgang angeschaltet
KnEnable	uint8	An- und Abschalten des Klopfgenerators. 0: Abgeschaltet 1: Angeschaltet
KnTrigger	uint8	Triggersignal für Klopfgenerator. Ein 0 → 1-Übergang des RTIO-Signal „KnTrigger“ triggert die Klopfsimulation für den Zeitraum der nächsten 100 Zündvorgänge (pro Zylinder). Nur aktiv im Triggermodus „Single Shot“.
KnTriggerMode	uint8	Triggermodus des Klopfgenerators 0: Triggermodus „Single Shot“ 1: Triggermodus „Continuous“.
KnAmplSubLimit	real32	Amplitude des Klopfsignals bei kontrollierter Verbrennung. Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor
KnAmplOverLimit	real32	Amplitude des Klopfsignals bei unkontrollierter Verbrennung. Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor
KnNoiseAmpl	real32	Amplitude (in Volt), mit der das Klopfsignal (per Zufallsgenerator) moduliert wird. Gültiger Wertebereich: Interne Bezugsspannung: (0.0 ... 10.0)/Amplification Factor. Externe Bezugsspannung: (0.0 ... 1.0)/Amplification Factor

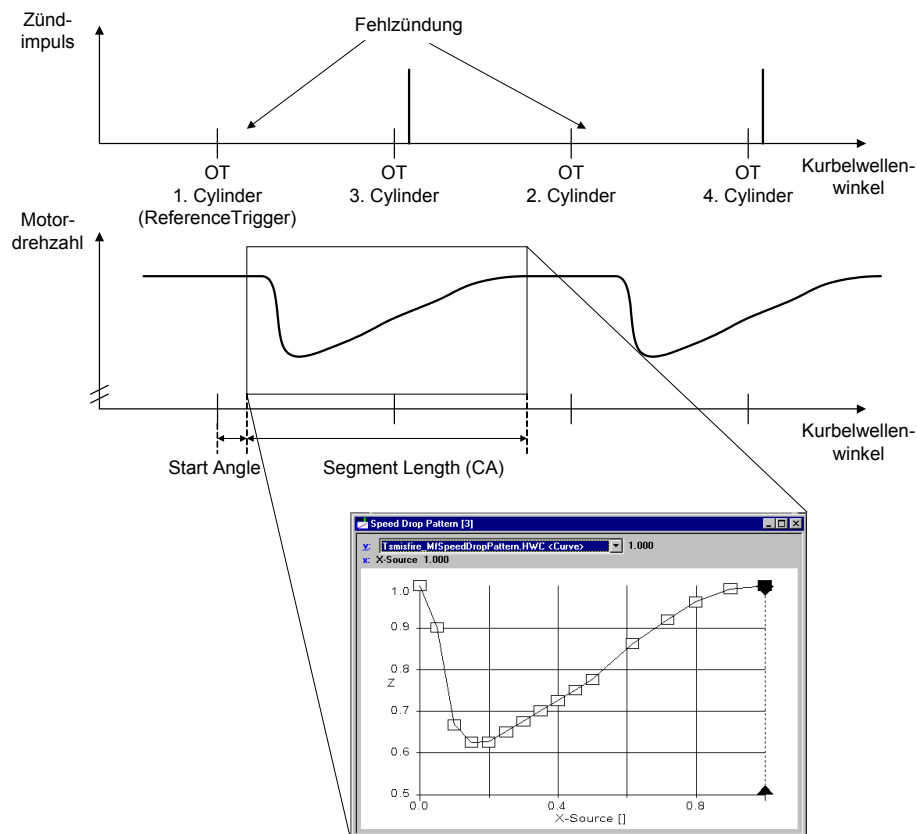
RTIO-Signale	Datentyp	Erläuterung
KnStartAngle	real32	Beginn des Klopfsignals relativ zum OT des jeweiligen Zylinders. Ein positives Vorzeichen bedeutet, dass das Klopfsignal vor OT einsetzt, ein negatives, dass das Klopfsignal nach OT einsetzt. Gültige Werte für den Kurbelwellenwinkel sind -720 °KW ... 720 °KW.
KnMode0 ... KnMode7	real32	Betriebsart für die Klopfgenerierung im jeweiligen Zylinder. 0: Disabled 1: Pattern mode 2: Random mode
KnProbability0 ... KnProbability7	real32	Wahrscheinlichkeit, dass während der aktuellen Zündung eine unkontrollierte Verbrennung auftritt. Nur aktiv im Modus „Random“. Gültige Werte: 0.0 (kein Klopfen) ... 1.0 (immer Klopfen)

**Tab. 28-6** Signale des TsKnockGen Devices

## 28.3.19 TsMisfire Device

Der Fehlzündungsgenerator simuliert einen Abfall der Motordrehzahl nach einer Fehlzündung und ist somit für Steuergeräte geeignet, die eine Fehlzündung anhand des nachfolgenden Abfalls der Drehzahl identifizieren.

Abb. 28-19 zeigt den Zusammenhang zwischen Fehlzündung und dem Abfall der Motordrehzahl.



**Abb. 28-19** Fehlzündung und Abfall der Motordrehzahl

Die Fehlzündungen können für jeden einzelnen Zylinder simuliert werden - die maximale Zahl der Zylinder liegt bei 12.

Die Fehlzündungen können per Zufallsgenerator mit einer festgelegten Wahrscheinlichkeit oder nach einem vorgegebenen Muster generiert werden. Diese Muster haben eine maximale Länge von jeweils 100 Zündungen (pro Zylinder) und kann beliebig oft wiederholt werden.

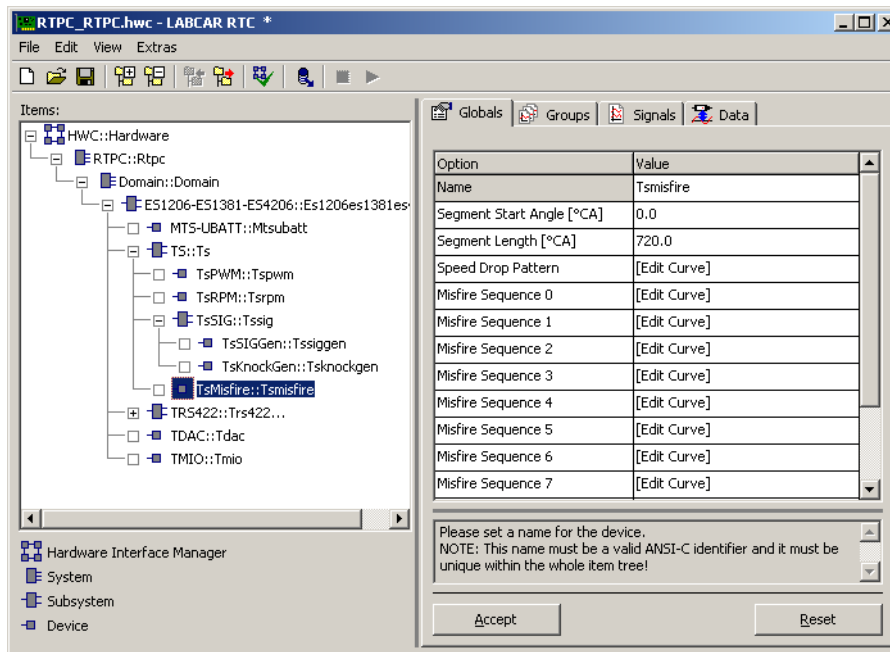
Es ist außerdem möglich, für jeden Zylinder einen Faktor anzugeben, der den Drehzahlabfall für jeden einzelnen Zylinder einzeln bestimmt.

#### **Hinweis**

*Der Fehlzündungsgenerator benötigt die Angabe der Anzahl der vorhandenen Zylinder. Die Einstellung der Zylinderanzahl erfolgt in dem Register „Globals“ des TS Subsystems (siehe Abschnitt „Number of Cylinders“ auf Seite 930)*

## 28.3.20 Globals (TsMisfire Device)

In diesem Abschnitt werden die globalen Einstellungen des TsMisfire Devices beschrieben.



**Abb. 28-20** Die Registerkarte „Globals“ des TsMisfire Devices

#### *Segment Start Angle [°CA]*

Der Parameter „Segment Start Angle“ wird verwendet, um den Winkel zwischen OT und dem Beginn des Abfalls der Motordrehzahl aufgrund der Fehlzündung festzulegen (siehe Abb. 28-19 auf Seite 955).

„Segment Start Angle“ wird in °KW angegeben und kann Werte zwischen -720.0 °KW und 720 °KW annehmen.

Dieser Parameter ist im Laufzeitmodus editierbar.

#### *Segment Length [°CA]*

Der Parameter „Segment Length“ bestimmt die Länge (in °KW), über die der Abfall der Motordrehzahl anhält (siehe Abb. 28-19 auf Seite 955).

„Segment Length“ wird in °KW angegeben und kann Werte zwischen 0 °KW und 720 °KW annehmen.

Dieser Parameter ist im Laufzeitmodus editierbar.

#### *Speed Drop Pattern*

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Festlegung der Charakteristik des Drehzahlabfalls nach einer Fehlzündung.

Die x-Achse hat dabei keine Bedeutung - die Dauer des Drehzahlabfalls (in °KW) wird mit dem Parameter „Segment Length“ festgelegt (siehe Abb. 28-19 auf Seite 955).



Die z-Achse bestimmt den Abfall der Motordrehzahl. Der zulässige Wertebereich reicht von 0.0 bis 2.0. Ein Wert von 0.5 beispielsweise bedeutet, dass die Motordrehzahl halbiert wird.

Ganz allgemein wird die Motordrehzahl während der Fehlzündungssimulation dadurch festgelegt, dass die Drehzahl vor Auftreten der Fehlzündung mit dem Faktor für den Abfall der Drehzahl (z-Wert) multipliziert wird.

Die durch aufeinanderfolgenden Fehlzündungen hervorgerufenen Drehzahländerungen können auch überlappen, z. B. wenn bei einem 4-Zylinder-Motor gilt: „Segment Length“ > 180 °KW.

Die Tabelle „Speed Drop Pattern“ ist im Laufzeitmodus editierbar.

*Misfire Sequence 0 ... Misfire Sequence 11:*

---

Ein Mausklick auf das Feld „Edit Curve“ öffnet den Tabelleneditor zur Festlegung des Musters für das Auftreten der Fehlzündungen. Diese Muster sind jeweils 100 Zündvorgänge lang. Die x-Achse besitzt keine Bedeutung.

Ein z-Wert von 1 bedeutet, dass für den aktuellen Zündvorgang eine Fehlzündung simuliert werden soll, während ein z-Wert von 0 bedeutet, dass für den aktuellen Zündvorgang keine Fehlzündung simuliert werden soll.

Die Tabelle „Misfire Sequence nn“ sind im Laufzeitmodus editierbar.

#### **Hinweis**

*Die Indizes 0 bis 11 der RTIO-Parameter „Misfire Sequence“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*

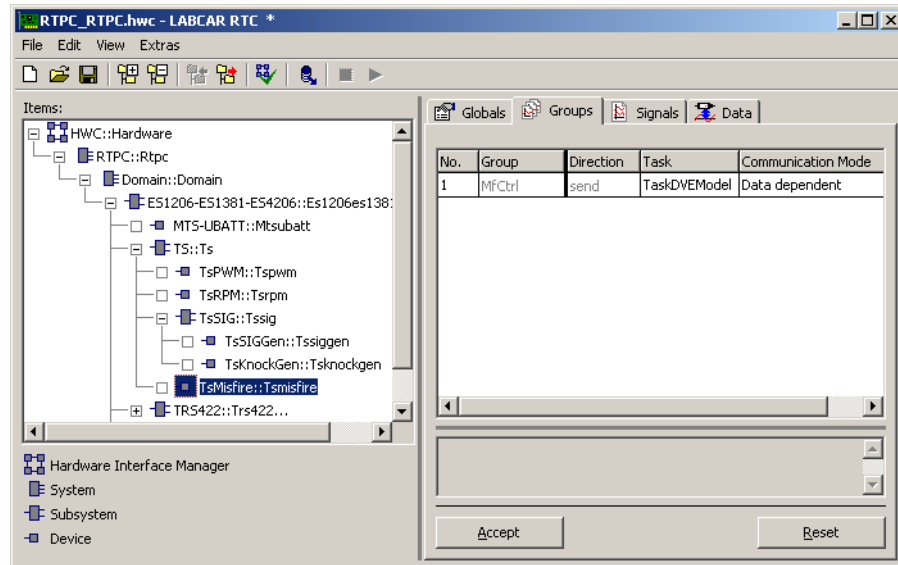
In Tab. 28-7 sind die Eigenschaften der globalen Parameter des TsMisfire Devices zusammengefasst.

RTIO Parameter	Datentyp	Editierbar im Laufzeitmodus	Erläuterung
Segment Start Angle	real32	Yes	Der Winkel zwischen OT (reference trigger) und dem Beginn des Drehzahlabfalls aufgrund der Fehlzündung. Gültiger Wertebereich: -720.0 °KW ... 720.0 °KW
Segment Length	real32	Yes	Dauer des Drehzahlabfalls (in °KW) nach der Fehlzündung Gültiger Wertebereich: 0 °KW ... 720.0 °KW
Speed Drop Pattern	1-dim. Table	Yes	Charakteristik des Drehzahlabfalls nach Fehlzündung z-Werte: 0.0 ... 2.0 Die x-Werte haben keine Bedeutung.
Misfire Sequence 0 ... Misfire Sequence 11	1-dim. Table	Yes	Fehlzündungsmuster. Jedes Muster ist 100 Zündvorgänge lang z-Wert = 1: Für den aktuellen Zündvorgang wird eine Fehlzündung simuliert z-Wert = 0: Für den aktuellen Zündvorgang wird keine Fehlzündung simuliert Die x-Werte haben keine Bedeutung.

**Tab. 28-7** Globale RTIO-Parameter des TsMisfire Devices

28.3.21 Groups (TsMisfire Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TsMisfire Devices beschrieben.



**Abb. 28-21** Die Registerkarte „Groups“ des TsMisfire Devices

*Communication Mode*

Der Kommunikationsmodus der Signalgruppe.

Option „Every interval“: Die Signalgruppe wird innerhalb eines jeden Intervalls (Task) zur TS-Karte gesendet.

Option „Data dependent“: Die Signalgruppe wird immer dann zur TS-Karte gesendet, wenn sich der Wert mindestens eines Signals geändert hat. Diese Option sollte bevorzugt verwendet werden.

Dieser Parameter ist im Laufzeitmodus editierbar.

Tab. 28-8 fasst die Eigenschaften der Signalgruppen-spezifischen RTIO-Parameter zusammen.

RTIO-Parameter	Daten-typ	Editierbar im Laufzeit-modus	Erläuterung
Communication Mode	uint8	Ja	Kommunikationsmodus der Signalgruppe 0: Every interval (task period) 1: Data dependent

**Tab. 28-8** Signalgruppen-spezifische Parameter des TsMisfire Devices

## 28.3.22 Signals (TsMisfire Device)

Die Registerkarte „Signals“ listet die einzelnen Signale des TsMisfire Devices auf.

*MfEnable*

Das RTIO-Signal „MfEnable“ wird verwendet, um den Fehlzündungsgenerator an- oder abzuschalten. Ein Wert von 0 schaltet den Fehlzündungsgenerator ab, ein Wert von 1 schaltet den Fehlzündungsgenerator an.

*MfTrigger*

Ein 0 → 1 Übergang des Signals „MfTrigger“ löst eine Fehlzündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus.

Das Signal „MfTrigger“ ist nur aktiv, wenn der Fehlzündungsgenerator im Triggermodus „Single Shot“ betrieben wird. Der Triggermodus wird über das Signal „MfTriggerMode“ eingestellt.

*MfTriggerMode*

Das RTIO-Signal „MfTriggerMode“ dient der Steuerung des Triggermodus des Fehlzündungsgenerators.

Wenn das Signal auf „0“ gesetzt wird, arbeitet der Fehlzündungsgenerator im Triggermodus „SingleShot“.

Ein 0 → 1 Übergang des Signals „MfTrigger“ löst eine Fehlzündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus. Danach wird die Simulation von Fehlzündungen ausgesetzt bis zum Auftreten des nächsten Triggers.

Wenn das Signal auf „1“ gesetzt wird, arbeitet der Fehlzündungsgenerator im Triggermodus „Continuous“. In diesem Modus ist die Simulation von Fehlzündungen ständig aktiv. In diesem Modus ist das Signal „MfTrigger“ nicht aktiv.

*MfMode0 ... MfMode11*

Die RTIO-Signale „MfMode0“ bis „MfMode11“ bestimmen die Betriebsart des Fehlzündungsgenerators für den jeweiligen Zylinder.

Disabled Mode (MfModenn = 0): Der Fehlzündungsgenerator für den jeweiligen Zylinder ist nicht aktiv.

Pattern Mode (MfModenn = 1): Die entsprechende Tabelle „Misfire Sequence nn“ bestimmt das Muster der Fehlzündungen für den jeweiligen Zylinder.

Random Mode (MfModenn = 2): Die Simulation von Fehlzündungen für den aktuelle Zündvorgang wird von einem Zufallsgenerator gesteuert. Die Wahrscheinlichkeit für das Auftreten von Fehlzündungen wird über das entsprechende Signal „MfProbability“ festgelegt.

*MfFactor0 ... MfFactor11*

Die RTIO-Signale „MfFactor0“ bis „MfFactor11“ spezifizieren den Korrekturfaktor für den Abfall der Motordrehzahl nach der Fehlzündung für den jeweiligen Zylinder.

Ein Faktor von 0.7 bedeutet, dass lediglich 70 % des in der Tabelle „Speed Drop Pattern“ angegebenen Betrages der Drehzahlverringerng wirksam werden.

Gültiger Wertebereich: 0.0 ... 1.0.

MfProbability0 ... MfProbability11

Die RTIO-Signale „MfProbability0“ bis „MfProbability11“ bestimmen die Wahrscheinlichkeit für eine Fehlzündung während der aktuellen Zündung.

Diese RTIO-Signale sind nur aktiv, wenn das entsprechende Signal „MfModenn“ auf „Random Mode“ gesetzt ist.

Der gültige Wertebereich ist 0.0 bis 1.0. Ein Wert von 0.0 bedeutet, dass eine Fehlzündung nie auftritt, ein Wert von 1.0 bedeutet, dass für jede Zündung im jeweiligen Zylinder eine Fehlzündung auftritt.

**Hinweis**

*Die Indizes 0 bis 11 der RTIO-Parameter „MfMode“, „MfFactor“ und „MfProbability“ beziehen sich auf die Zündfolge des Motors. Die Zündfolge ist nicht notwendig identisch mit der Anordnung der Zylinder.*

Tab. 28-9 fasst die Eigenschaften der RTIO-Signals des TsMisfire Devices zusammen.

RTIO Signal	Data Type	Kommentar
MfEnable	uint8	An- und Abschalten des Fehlzündungsgenerators 0: Abschalten 1: Anschalten
MfTrigger	uint8	Triggersignal des Fehlzündungsgenerators. Ein 0 → 1 Übergang löst eine Fehlzündungssimulation für die nächsten 100 Zündvorgänge (je Zylinder) aus. Nur aktiv im Triggermodus „Single Shot“.
MfTriggerMode	uint8	Triggermodus des Fehlzündungsgenerators. 0: Triggermodus „Single Shot“ 1: Triggermodus „Continuous“
MfMode0 ... MfMode11	uint8	Betriebsart des Fehlzündungsgenerators für den jeweiligen Zylinder 0: Disabled 1: Pattern mode 2: Random mode
MfFactor0 ... MfFactor11	real32	Korrekturfaktor für den Abfall der Motordrehzahl nach der Fehlzündung für den jeweiligen Zylinder Gültige Werte: 0.0 ... 1.0
MfProbability0... MfProbability11	real32	Wahrscheinlichkeit für eine Fehlzündung während der aktuellen Zündung. Nur aktiv im „Random Mode“. Gültige Werte: 0.0 (keine Fehlzündung) ... 1.0 (dauernde Fehlzündung)

**Tab. 28-9** RTIO-Signale des TsMisfire Devices

## 28.4 TMIO - Digitale Signalerfassung und I/O

---

Die TMIO-Karte (Transputer-Mess- und I/O-Karte) erfüllt im LABCAR-Umfeld zwei Grundfunktionen:

- Erfassung und Auswertung digitaler Steuergeräte-Signale
- Generierung von Schaltsignalen (als Steuergeräte-Eingänge)

Insgesamt stehen dem Anwender 30 Hardware-Kanäle (0-29) zur Verfügung, die alle zur Signalerfassung herangezogen werden können. Ein gewisser Teil dieser Kanäle (0-15) bietet darüber hinaus die Möglichkeit der Signalgenerierung, d. h. diese Kanäle können *entweder* zur Signalerfassung *oder* zur Stimulation von Steuergeräte-Eingängen genutzt werden.

Insbesondere ermöglichen TMIO-Karten somit:

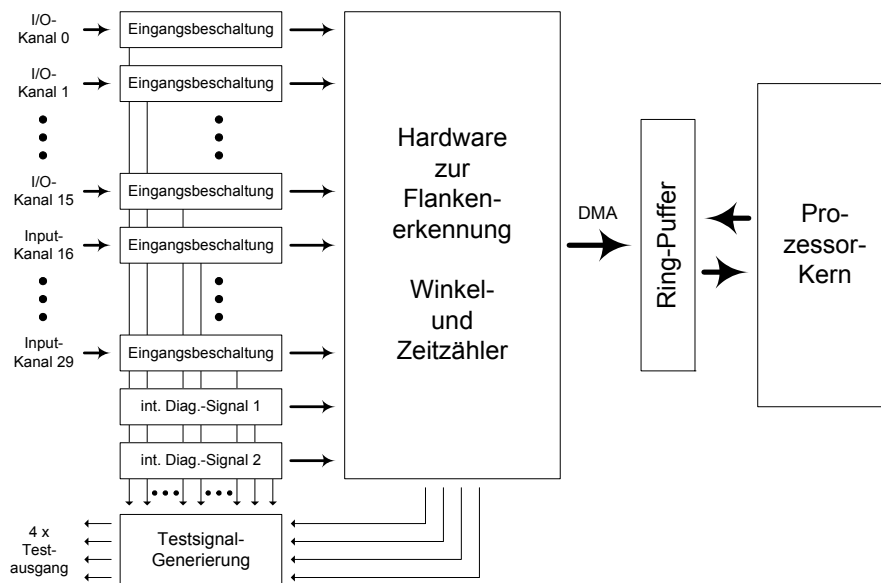
- Erfassung und Auswertung drehzahlsynchroner Signale, z.B. Einspritzsignale, Zündsignale
- Erfassung und Auswertung zeitsynchroner Signale (i.d.R. PWM), z.B. Abgasrückführung, Tankentlüftungsventil
- Erfassung und Auswertung statischer (Schalt-)Signale, z.B. Lüftersteuerung, Fehlerlampe
- Generierung statischer (Schalt-)Signale, z.B. Klimaschalter, Bremsschalter

### 28.4.1 Grundprinzip der Messwertbestimmung

---

Um eine möglichst große Flexibilität bei der Auswertung anliegender Eingangssignale zu erreichen, basiert die TMIO-Karte auf folgendem Grundprinzip:

Zuerst wird jedes Eingangssignal separat (analog) aufbereitet, dann mit einem (für jedes Signal einzeln softwaretechnisch einstellbaren) Schwellwert verglichen, um die Umsetzung des analogen Eingangssignals in eine digitale 0/1- bzw. inaktiv/aktiv-Information zu realisieren. Die so erhaltenen 32 Bit (30 Kanäle zzgl. zweier interner Diagnose-Signale) werden ständig von einer Hardwareschaltung auf Änderungen hin überprüft, d.h. es wird eine Erkennung von „Flanken“ auf einem oder mehreren Eingangsbits durchgeführt. Wechselt nun zumindest ein Eingangssignal von „aktiv“ auf „inaktiv“ bzw. umgekehrt, so speichert die Schaltung die aktuellen Zustände der 32 Eingänge und ruft die momentanen Werte zweier integrierter Zähler ab, die die aktuelle Zeit und den aktuellen Kurbelwellenwinkel angeben. Diese Informationen werden dann mittels DMA (Direct Memory Access), also hardwaregesteuert, in einen Ring-Puffer übertragen.



**Abb. 28-22** Blockschaltbild TMIO-Karte

#### Hinweis

Der Winkelzähler der TMIO-Karte wird von zwei Signalen gesteuert, die von der TS-Karte generiert werden. Das Vorhandensein einer TS-Karte ist für die Bestimmung von Kurbelwellenwinkel-abhängigen Messwerten zwingend erforderlich.

Die Hardware der TMIO-Einschubkarte erfasst auch Pulse, deren Pulsbreite kleiner als die eingestellte Auflösung ist. Solche kurzen Pulse, „Spikes“ genannt, können bei einigen Auswerteverfahren (beispielsweise der Pulszählung o.ä.) zu Fehlinterpretationen führen. Pulse, die kürzer sind als die eingestellte Auflösung, werden daher von der Auswertesoftware *nicht* berücksichtigt.

Die auf den Transputer der TMIO-Karte heruntergeladene Firmware kann nun die generierten Werte aus dem Ring-Puffer auslesen und somit die Information erzeugen, wann und bei welchem Winkel sich welche Eingangssignale wie verändert haben. Aus diesen Informationen wiederum lassen sich sämtliche Messwerte generieren, die letztendlich für den Anwender von Interesse sind wie z.B. Einspritzzeiten, Zündzeitpunkte oder Tastverhältnisse.

Dieses Grundkonzept einer hardwaretechnischen Flankenerkennung mit einer nachgeschalteten softwaretechnischen Aufbereitung der Messwerte bewirkt, dass die Auslastung des integrierten Prozessorkerns direkt von der Zahl der erkannten Flanken (also den Frequenzen der Eingangssignale) abhängig ist. Im Laborauto-Umfeld ist der integrierte Prozessorkern *ausschließlich* zur Erfassung und Auswertung der Eingangssignale vorgesehen. Darüber hinaus ist die Rechenleistung des Prozessors hoch genug um auch bei einer Nutzung aller zur Verfügung stehender Eingangssignale die erforderlichen Messwerte schnell genug ermitteln zu können, ohne hierdurch Echtzeitverletzungen hervorzurufen.

Der Vorteil dieses Konzepts: Die Hardware erfasst lediglich sämtliche Änderungen und versieht sie mit „Winkel-“ und „Zeitstempeln“. Die Auswertung erfolgt hingegen per Software, ist demzufolge verhältnismäßig leicht zu ändern bzw.

anzupassen. Die Firmware der TMIO-Karte bietet verschiedene Formen der Messwertbestimmung (vgl. Abschnitt 28.4.3 „Signalauswertung“ auf Seite 965). Das auf der TMIO-Karte eingesetzte Konzept ermöglicht eine leichte Anpassung: Für die Realisierung neuer Messarten genügt eine Anpassung der TMIO-Firmware und ein Update der RTIO-Einbindung der Karte.

#### 28.4.2 Messwerterfassung

---

Die RTIO-Einbindung der TMIO-Karte stellt dem Anwender insgesamt 64 frei konfigurierbare Messwerte zur Verfügung, wobei jeder einzelne Messwert von jedem beliebigen Hardware-Kanal abhängen kann. Die Zuordnung von Hardware-Kanälen auf der einen und resultierenden Messwerten auf der anderen Seite ist somit nicht durch die TMIO-Karte vorgegeben, d.h. bei der Nutzung eines Laborautos für mehrere Projekte ist es nicht zwingend notwendig, bedeutungsgleiche Signale immer an denselben Hardware-Kanälen der TMIO-Karte anzuschließen.

Darüber hinaus werden dem Anwender mit den generierten 64 Messwerten mehr als doppelt so viele Ausgangsgrößen zur Verfügung gestellt, wie die Karte I/O-Kanäle aufweist. Hierdurch ist es ohne Weiteres möglich, aus einem einzigen Eingangssignal mehrere Messwerte zu ermitteln wie z.B. Frequenz und Tastverhältnis eines PWM-Signals. Eine doppelte Verdrahtung ein und desselben Signals auf verschiedene Kanäle der Karte entfällt.

Für jeden dieser Messwerte kann (neben dem zugrundeliegenden Hardware-Kanal) vollkommen unabhängig von allen anderen Messwerten definiert werden,

- wie das anliegende Signal auszuwerten ist,
- wann und wie es auf fehlende Pulse hin untersucht werden soll (*Timeout-Erkennung*) und
- ob die Auswertung dauernd stattfinden soll oder nur innerhalb eines bestimmten Bereiches des Kurbelwellenwinkels (*Winkelsegmentierung*).

Für einige Messungen lässt sich weiter festlegen, der wievielte Puls vermessen werden soll.

##### *Anmerkung zur Zahl der Messkanäle*

---

Die TMIO-Karte besitzt intern Speicher für 75 Messkanäle - davon sind höchstens 64 durch den Anwender nutzbar bzw. konfigurierbar. Warum es höchstens 64 Messkanäle und nicht genau 64 sind, soll nachfolgend erläutert werden.

Einige Messverfahren erfordern zu ihrer Durchführung intern noch einen zweiten Messkanal. Dieser Messkanal wird von der TMIO-Firmware konfiguriert und ist damit für den Anwender nicht mehr nutzbar bzw. konfigurierbar. Überschreitet die Anzahl der Messkanäle mit Messverfahren, die 2 Kanäle erfordern, die Zahl 11, so stehen dem Anwender nicht mehr alle 64 Messkanäle zur Verfügung.

Allgemein gilt die Anzahl der für den Anwender nutzbaren Messkanäle beträgt:  
 $n = 64 - (\text{Anzahl der Messkanäle mit Messverfahren, die 2 Kanäle erfordern} - 11)$ .

$$n_{\max} = 64$$



Die folgenden Messverfahren erfordern intern noch einen 2. Messkanal:

- Hightime using H-Enable [ $\mu\text{s}$ ]
- Hightime using L-Enable [ $\mu\text{s}$ ]
- Hightime using H-Validate [ $\mu\text{s}$ ]
- Hightime using L-Validate [ $\mu\text{s}$ ]
- Lowtime using H-Enable [ $\mu\text{s}$ ]
- Lowtime using L-Enable [ $\mu\text{s}$ ]
- Lowtime using H-Validate [ $\mu\text{s}$ ]
- Lowtime using L-Validate [ $\mu\text{s}$ ]
- H-Time n-th Pulse (H-Ena) [ $\mu\text{s}$ ]
- H-Time n-th Pulse (L-Ena) [ $\mu\text{s}$ ]
- H-Time n-th Pulse (H-Val) [ $\mu\text{s}$ ]
- H-Time n-th Pulse (L-Val) [ $\mu\text{s}$ ]
- L-Time n-th Pulse (H-Ena) [ $\mu\text{s}$ ]
- L-Time n-th Pulse (L-Ena) [ $\mu\text{s}$ ]
- L-Time n-th Pulse (H-Val) [ $\mu\text{s}$ ]
- L-Time n-th Pulse (L-Val) [ $\mu\text{s}$ ]
- Time to first --/-- (H-En) [ $\mu\text{s}$ ]
- Time to first --\-- (H-En) [ $\mu\text{s}$ ]
- Time to first --/-- (L-En) [ $\mu\text{s}$ ]
- Time to first --\-- (L-En) [ $\mu\text{s}$ ]
- Time to last --/-- (H-En) [ $\mu\text{s}$ ]
- Time to last --\-- (H-En) [ $\mu\text{s}$ ]
- Time to last --/-- (L-En) [ $\mu\text{s}$ ]
- Time to last --\-- (L-En) [ $\mu\text{s}$ ]
- Step Count (Stepper Motor)

### 28.4.3 Signalauswertung

Die RTIO-Einbindung der TMIO-Karte unterstützt eine Vielzahl verschiedener Möglichkeiten der Signalauswertung:

#### *Pegelerfassung*

Zur Erfassung von (statischen) Schaltsignalen bietet die TMIO-Karte die Möglichkeit der Pegelmessung. Ein entsprechend konfigurierter Messwert der TMIO-Karte hängt direkt vom 0/1-Ergebnis des Schwellwertvergleichs für den zugeordneten Hardware-Kanal ab:

- Erfassung von *active high*-Signalen:  
Liefert der Schwellwertvergleich eine „0“ als Ergebnis, so ist der Messwert 0,0, andernfalls ist der Messwert 1,0.

- Erfassung von *active low*-Signalen:

Im Gegensatz zur Auswertung von *active high*-Signalen ist hier der Messwert 1,0 dann, wenn der Vergleich den Wert „0“ liefert. Das Vergleichsergebnis „1“ resultiert im Messwert 0,0.

#### Steckbrief Pegelerfassung:

Messung: „active high Pegel“ / „active low Pegel“

Wertebereich: 0,0; 1,0

Auflösung:

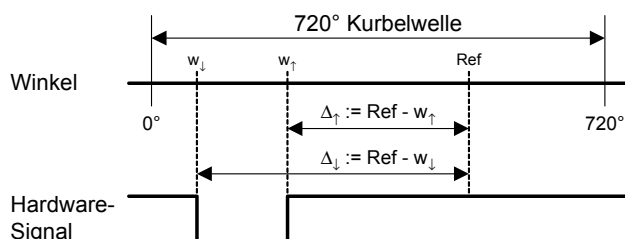
Berechnung: 0,0 / 1,0, falls Eingang < Schwelle  
1,0 / 0,0, falls Eingang > Schwelle

Berechnungszeitpunkt: jede ansteigende und jede abfallende Flanke

Aktualisierung: im nachfolgenden Signalgruppen-Rechenraster

#### Winkelmessungen

Winkelmessungen sind sowohl an steigenden als auch an fallenden Flanken des jeweiligen Eingangssignals möglich. Hierbei kann zu jedem Messwert ein beliebiger Bezugspunkt definiert werden; der Messwert ergibt sich jeweils aus der Differenz zwischen diesem Bezugspunkt und dem aktuellen Winkel (vgl. nachfolgende Abbildung).



**Abb. 28-23** Winkelmessungen mit der TMIO-Karte

Winkelmessungen dienen beispielsweise der Erfassung von Zünd- und Einspritz-Winkeln. Hierbei wird der Referenzpunkt vom Anwender so festgelegt, dass er dem oberen Totpunkt (OT) des entsprechenden Zylinders entspricht. Die TMIO-Firmware liefert dann „Winkel vor OT“ als positive und „Winkel nach OT“ als negative Werte.

Für die Winkelmessungen lässt sich weiter festlegen, der wievielte Puls innerhalb eines Winkelbereichs vermessen werden soll.

#### Steckbrief Winkelmessung:

Messung: „Steigende Flanke“ / „Fallende Flanke“

Wertebereich:  $-720,0^\circ \dots +720,0^\circ$

Auflösung:  $(720,0^\circ / 8192) = 0,088^\circ$

Berechnung:  $\text{Ref} - w^\uparrow / \text{Ref} - w^\downarrow$

Berechnungszeitpunkt:  $\uparrow / \downarrow$

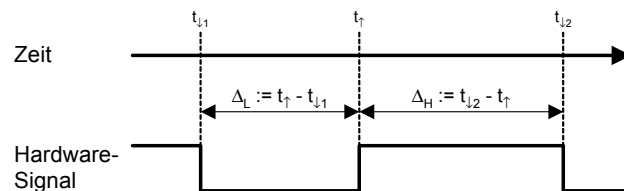
Auswahlbedingung: Winkelsegment  
n-ter Puls

Aktualisierung: im nachfolgenden Signalgruppen-Rechenraster

### High- und Low-Zeiten

Zur Bestimmung von Pulsbreiten bietet die TMIO-Karte sowohl eine High-Zeit- als auch eine Low-Zeit-Messung, wie in der folgenden Abbildung skizziert ist.

Die Messung von High- und Low-Zeiten lässt sich sowohl über einen vorgegebenen Winkelbereich als auch über die Nummer des zu vermessenden Pulses qualifizieren.



**Abb. 28-24** Zeitmessungen mit TMIO-Karte

#### Steckbrief Zeitmessung:

Messung „Low-Zeit“ / „High-Zeit“

Wertebereich:  $0,0 \mu\text{s} \dots +\infty$

Auflösung:  $1,0 \mu\text{s}$

Berechnung:  $t_{\uparrow} - t_{\downarrow} / t_{\downarrow} - t_{\uparrow}$

Berechnungszeitpunkt:  $\uparrow / \downarrow$

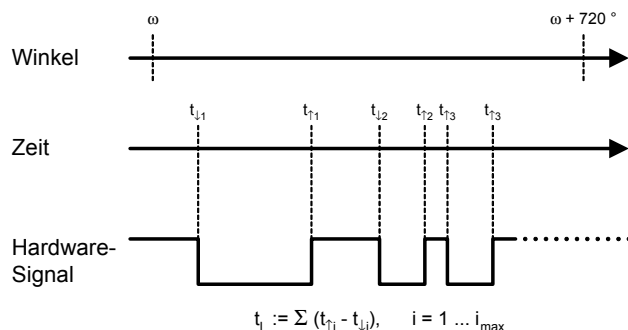
Auswahlbedingung: Winkelsegment  
n-ter Puls

Aktualisierung: im nachfolgenden Signalgruppen-Rechenraster

### Additive Zeiten und Pulszahlen

Neben der Vermessung einzelner Pulsbreiten bietet die TMIO-Karte auch die Möglichkeit, „additive“ Zeiten zu ermitteln (in Abb. 28-25 „Additive Zeitmessungen mit TMIO-Karte“ ist eine additive Low-Zeitmessung beispielhaft skizziert). Hierzu werden alle High- bzw. Low-Pulse innerhalb eines vorgegebenen Winkelsegments (vgl. Abschnitt 28.4.5 „Winkelsegmentierung“ auf Seite 973) bis zu einer vom Anwender vorgegebenen Höchst-Pulszahl aufsummiert. Als

Sonderform dieser Messung wird darüber hinaus eine „Pulszählung“ angeboten, wobei nicht die Zeiten aufsummiert, sondern schlicht die aufgetretenen Pulse gezählt werden.



**Abb. 28-25** Additive Zeitmessungen mit TMIO-Karte

#### Hinweis

Da bei additiven Messungen innerhalb eines freigegebenen Winkelsegments (vgl. Abschnitt 28.4.5 „Winkelsegmentierung“ auf Seite 973) nicht vorhergesagt werden kann, ob innerhalb des gleichen Segments noch weitere Pulse auftreten werden, ist eine Aktualisierung des Messwerts bei diesen Messungen erst am Ende des Segments möglich!

#### Steckbrief additive Zeitmessung:

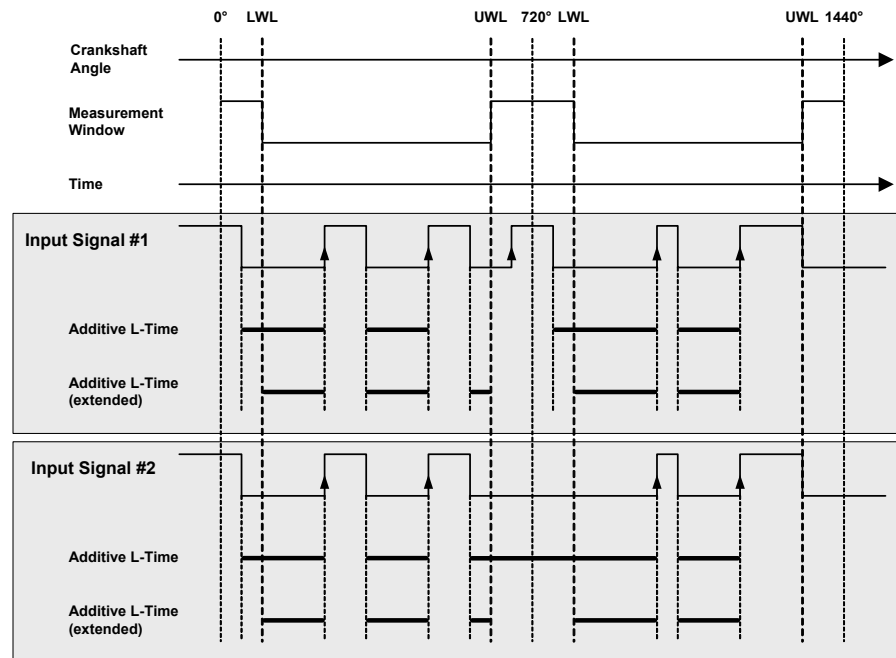
Messung:	„additive Low-Zeit“ / „additive High-Zeit“
Wertebereich:	0,0 $\mu$ s ... $+\infty$
Auflösung:	1,0 $\mu$ s
Berechnung:	$\Sigma(t\uparrow - t\downarrow) / \Sigma(t\downarrow - t\uparrow)$
Berechnungszeitpunkt:	$\uparrow / \downarrow$
Auswahlbedingung:	Winkelsegment
Aktualisierung:	im ersten Signalgruppen-Rechenraster, das auf die obere Grenze des genutzten Winkelsegmentes folgt

#### Steckbrief Pulszählung:

Messung:	„Anzahl Low-Pulse“ / „Anzahl High-Pulse“
Wertebereich:	0 ... $+\infty$
Auflösung:	—
Berechnung:	Anzahl ( $\downarrow\uparrow$ ) / Anzahl ( $\uparrow\downarrow$ )
Berechnungszeitpunkt:	$\uparrow / \downarrow$
Auswahlbedingung:	Winkelsegment fortlaufende Addition
Aktualisierung:	im ersten Signalgruppen-Rechenraster, das auf die obere Grenze des genutzten Winkelsegmentes folgt

Unterschied: „Additive L/H-Time“ und „Additive L/H-Time (extended)“

Der Unterschied zwischen den beiden Messmodi ist in Abb. 28-26 gezeigt.

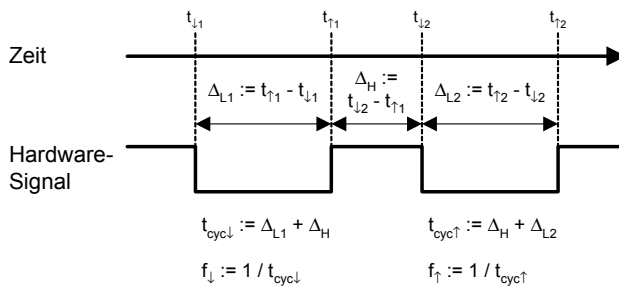


**Abb. 28-26** Additive L-Time und Additive L-Time (extended)

„Additive L-Time“ misst alle Low-Zeiten, deren schließende Flanken sich innerhalb eines Winkelfensters befinden (unabhängig davon, wann die Low-Zeit begonnen hat), während „Additive L-Time (extended)“ nur die Low-Zeiten innerhalb des Winkelfensters addiert (unabhängig von der Lage der Flanken).

Frequenzen und Zykluszeiten

Die TMIO-Karte ermöglicht Vermessungen von Frequenzen und Zykluszeiten, wie in der folgenden Abbildung skizziert. Die Flanke, an der jeweils der Messwert bestimmt werden soll, kann hierbei vom Anwender beliebig vorgegeben werden.



**Abb. 28-27** Frequenzmessungen mit TMIO-Karte.

**Steckbrief Zykluszeitmessung:**

Messung:	„Zykluszeit ↑“ / „Zykluszeit ↓“
Wertebereich:	0,0 $\mu\text{s}$ ... $+\infty$
Auflösung:	1,0 $\mu\text{s}$

**Steckbrief Zykluszeitmessung:**Berechnung:  $(t_{\uparrow} - t_{\downarrow}) + (t_{\downarrow} - t_{\uparrow}) / (t_{\downarrow} - t_{\uparrow}) + (t_{\uparrow} - t_{\downarrow})$ Berechnungszeitpunkt:  $\uparrow / \downarrow$ 

Auswahlbedingung: Winkelsegment

Aktualisierung: im nachfolgenden Signalgruppen-Rechenraster

**Steckbrief Frequenzmessung:**Messung: „Frequenz  $\uparrow$ “ / „Frequenz  $\downarrow$ “Wertebereich: 0,0 Hz ...  $+\infty$ 

Auflösung: abhängig von Zykluszeit-Bestimmung

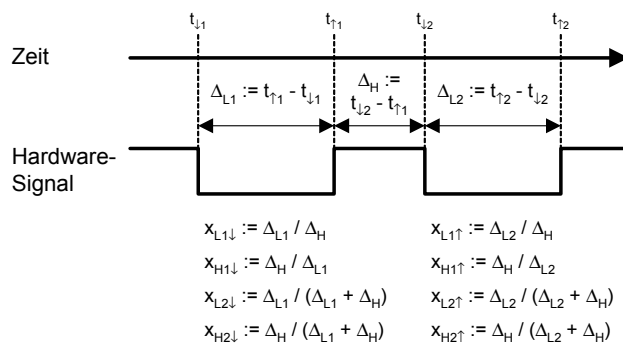
Berechnung:  $1 / ((t_{\uparrow} - t_{\downarrow}) + (t_{\downarrow} - t_{\uparrow})) / 1 / ((t_{\downarrow} - t_{\uparrow}) + (t_{\uparrow} - t_{\downarrow}))$ Berechnungszeitpunkt:  $\uparrow / \downarrow$ 

Auswahlbedingung: Winkelsegment

Aktualisierung: im nachfolgenden Signalgruppen-Rechenraster

*Tastverhältnisse*

Tastverhältnisse (z.B. von PWM-Signalen) können mit der TMIO-Karte auf unterschiedlichste Arten ermittelt werden, wie in der folgenden Abbildung skizziert ist. Auch bei diesen Messungen kann die Flanke, an der jeweils die Auswertung durchgeführt wird, vom Anwender beliebig vorgegeben werden.

**Abb. 28-28** Tastverhältnis-Bestimmung mit TMIO-Karte.

**Steckbrief Bestimmung Tastverhältnis (1):**

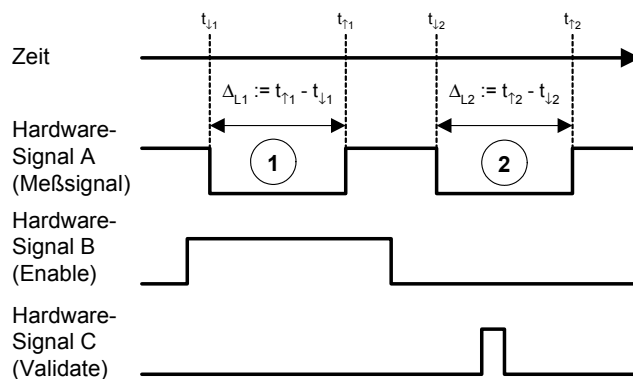
Messung:	„Verhältnis L/H ↑“ / „Verhältnis H/L ↑“ „Verhältnis L/H ↓“ / „Verhältnis H/L ↓“
Wertebereich:	0,0 ... +∞
Auflösung:	abhängig von High- bzw. Low-Zeitbestimmung
Berechnung:	$(t_{L\downarrow} - t_{H\uparrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{L\downarrow} - t_{H\uparrow})$ $(t_{L\downarrow} - t_{H\uparrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{L\downarrow} - t_{H\uparrow})$
Berechnungszeitpunkt:	↑ / ↑ ↓ / ↓
Auswahlbedingung:	Winkelsegment
Aktualisierung:	im nachfolgenden Signalgruppen-Rechenraster

**Steckbrief Bestimmung Tastverhältnis (2):**

Messung:	„Verhältnis L/(L+H) ↑“ / „Verhältnis H/(L+H) ↑“ / „Verhältnis L/(L+H) ↓“ / „Verhältnis H/(L+H) ↓“
Wertebereich:	0,0 ... 1,0
Auflösung:	abhängig von High- bzw. Low-Zeitbestimmung
Berechnung:	$(t_{L\downarrow} - t_{H\uparrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{L\downarrow} - t_{H\uparrow})$ $(t_{L\downarrow} - t_{H\uparrow}) / (t_{L\downarrow} - t_{L\downarrow}) / (t_{H\uparrow} - t_{L\downarrow}) / (t_{L\downarrow} - t_{L\downarrow})$
Berechnungszeitpunkt:	↑ / ↑ ↓ / ↓
Auswahlbedingung:	Winkelsegment
Aktualisierung:	im nachfolgenden Signalgruppen-Rechenraster

*High- und Low-Zeiten bei externer Signalvalidierung*

Neben diesen direkten Auswertungen eines von allen anderen Eingängen unabhängigen Signals bietet die TMIO-Karte noch die Möglichkeit, High- bzw. Low-Zeiten eines Signals in Abhängigkeit eines zweiten Eingangs zu erfassen. Dieses zweite Eingangssignal wird hierbei als „enable“- bzw. „validate“-Signal interpretiert, wie in der folgenden Abbildung am Beispiel einer validierten Low-Zeit-Messung skizziert ist:



**Abb. 28-29** Validierte Zeitmessungen mit TMIO-Karte

Wird die TMIO-Karte beispielsweise derart konfiguriert, dass Low-Zeiten eines Eingangssignals A vermessen werden sollen, wobei ein zweites Eingangssignal B als „active high“ Enable-Signal genutzt werden soll, so würde vom Eingangssignal aus der obenstehenden Abbildung lediglich der mit 1 gekennzeichnete Puls vermessen. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Enable“ werden ausschließlich die Pulse vermessen, die **vollständig** von einem aktiven Enable-Puls **umschlossen** sind.

Wird hingegen ein weiteres Eingangssignal C als „active high“ Validate-Signal genutzt, so würde im obigen Beispiel lediglich der mit 2 gekennzeichnete Puls vermessen werden. Allgemeiner: Bei Nutzung eines zweiten Eingangssignals als „Validate“ werden ausschließlich die Pulse vermessen, die ihrerseits einen aktiven Validate-Puls **vollständig umschließen**.

Die TMIO-Karte bietet sowohl für High- als auch für Low-Zeit-Messungen jeweils die Nutzung von Enable- oder Validate-Signalen an. Diese können dabei wahlweise als „active high“ oder auch als „active low“ definiert werden.

#### Steckbrief validierte Zeitmessung (1)

Messung:	„Low-Zeit mit High- oder Low-Enable“ / „Low-Zeit mit High- oder Low-Validate“
Wertebereich:	0,0 $\mu$ s ... $+\infty$
Auflösung:	1,0 $\mu$ s
Berechnung:	$t_{\uparrow} - t_{\downarrow}$ , falls von aktivem Enable-Puls umschlossen / $t_{\uparrow} - t_{\downarrow}$ , falls hierbei aktiver Validate-Puls umschlossen
Berechnungszeitpunkt:	$\uparrow$
Auswahlbedingung:	Winkelsegment Enable-Signal Validate-Signal n-ter Puls
Aktualisierung:	im nachfolgenden Signalgruppen-Rechenraster

#### Steckbrief validierte Zeitmessung (2):

Messung:	„High-Zeit mit High- oder Low-Enable“ / „High-Zeit mit High- oder Low-Validate“
Wertebereich:	0,0 $\mu$ s ... $+\infty$
Auflösung:	1,0 $\mu$ s
Berechnung:	$t_{\downarrow} - t_{\uparrow}$ , falls von aktivem Enable-Puls umschlossen / $t_{\downarrow} - t_{\uparrow}$ , falls hierbei aktiver Validate-Puls umschlossen
Berechnungszeitpunkt:	$\downarrow$
Auswahlbedingung:	Winkelsegment Enable-Signal Validate-Signal n-ter Puls
Aktualisierung:	im nachfolgenden Signalgruppen-Rechenraster



#### 28.4.4 Timeout-Erkennung

---

Wie bereits in Abschnitt 28.4.1 auf Seite 962 beschrieben, arbeitet die TMIO-Karte flankengesteuert, d.h., die Auswertung der anliegenden externen Signale findet ausschließlich an deren Flanken statt. Dies hat jedoch zur Folge, dass auch die generierten Messwerte, die die TMIO-Karte liefert, nur nach Flanken aktualisiert werden. Ein Signal, welches bereits einmal von der Karte ausgewertet wurde, würde folglich auch dann noch den zuletzt generierten gültigen Messwert „generieren“, wenn es inzwischen z.B. vollkommen ausgefallen ist, also keinerlei Flanken mehr aufweist.

Um Signale, die ganz oder zeitweise ausfallen, auch korrekt zu interpretieren, bietet die TMIO-Karte die Möglichkeit der Timeout-Erkennung. Jeder einzelne von der TMIO-Firmware generierte Messwert kann hierzu in bestimmten Abständen daraufhin getestet werden, ob er durch neu aufgetretene Flanken am Eingangssignal generiert wurde. Dieser Test kann sowohl in vom Anwender spezifizierbaren Zeitabständen als auch jeweils an vom Anwender vorgegebenen aktuellen Kurbelwellen-Winkeln erfolgen.

Wird während einer derartigen Überprüfung festgestellt, dass seit der letzten Überprüfung keinerlei Flanke von der Hardware detektiert wurde, so kann der entsprechende Messwert auf zwei verschiedene Arten modifiziert werden. Entweder wird er auf einen vorgegebenen Timeout-Wert gesetzt oder aber in Abhängigkeit vom aktuellen Zustand des Eingangssignals ermittelt. Dies ist insbesondere zur korrekten Auswertung von Tastverhältnissen erforderlich (ein inaktiver Eingang resultiert hierbei in einem Messwert von 0,0 bzw. 1,0 - je nach Eingangssignal).

Insgesamt ergeben sich somit vier Möglichkeiten der Timeout-Erkennung:

- Überprüfung in Zeitabständen von x ms: Der Messwert im Timeoutfall ist vordefiniert auf Wert y.
- Überprüfung alle 720° bei x° Kurbelwelle: Der Messwert im Timeoutfall ist vordefiniert auf Wert y.
- Überprüfung in Zeitabständen von x ms: Der Messwert im Timeoutfall ist abhängig vom Zustand des Eingangssignals („low“ oder „high“).
- Überprüfung alle 720° bei x° Kurbelwelle: Der Messwert im Timeoutfall ist abhängig vom Zustand des Eingangssignals („low“ oder „high“).

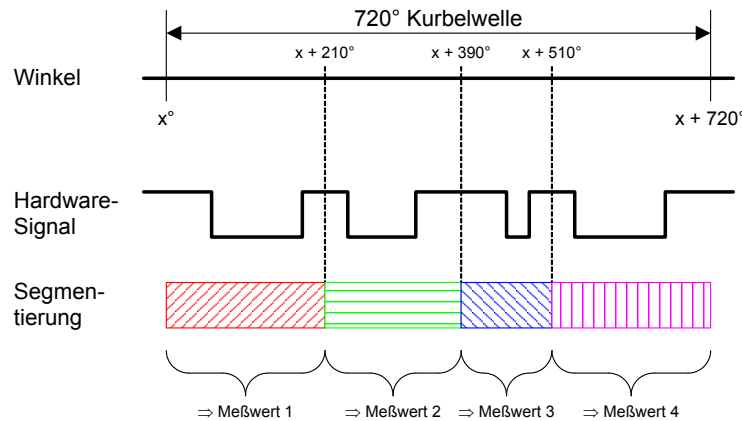
Bei der Einstellung „Überprüfung alle 720° Kurbelwelle“ würde ein stehender Motor nicht zu einem Timeout-Signal führen, da sich der Kurbelwellenwinkel nicht mehr ändert. Um diesem Problem abzuwehren, werden Drehzahlen kleiner als 10 (1/min) als stehender Motor interpretiert und die entsprechenden Messwerte werden auf „Timeout“ gesetzt.

#### 28.4.5 Winkelsegmentierung

---

Drehzahlsynchrone Signale müssen häufig für unterschiedliche Winkelbereiche auch unterschiedlich (und voneinander unabhängig) ausgewertet werden (Beispiel: Zündverteilung). Diesem Umstand wird auf der TMIO-Karte dadurch Rechnung getragen, dass für jede spezifizierte Messung ein Winkelbereich definiert werden kann, in dem die Auswertung des Signals zu erfolgen hat. Sämtliche außerhalb dieses Bereiches liegenden Flanken werden bei der Berechnung des Messwertes ignoriert.

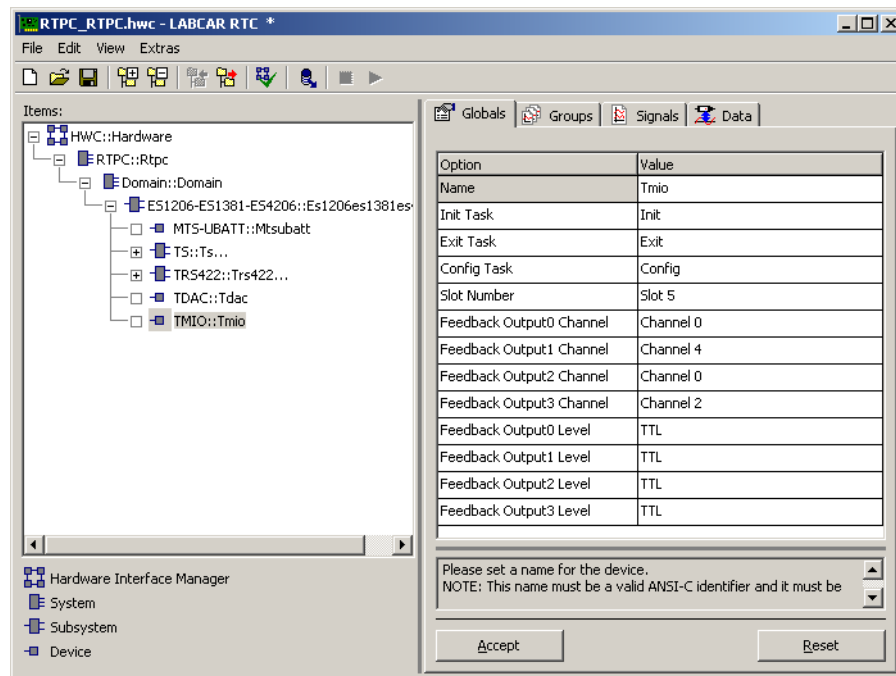
Der für jeden Messwert separat definierbare Winkelbereich darf hierbei beliebig innerhalb einer 720°-Periode liegen, d.h. auch Segmente, die die 0°-Grenze überschreiten, sind problemlos einstellbar. Ein Signal, wie es als Beispiel in der folgenden Abbildung skizziert ist, kann somit sehr einfach auf die vier angegebenen Messwerte „aufgeteilt“ werden. Hierzu wäre es lediglich erforderlich, bei der Konfiguration von vier unterschiedlichen Messwerten jeweils auf denselben Hardware-Kanal zu referenzieren und hierbei die in der Abbildung angegebenen oberen und unteren Grenzen des jeweils freizugebenden Winkelsegments anzugeben.



**Abb. 28-30** Winkelsegmentierung mit TMIO-Karte

28.4.6 Globals (TMIO Device)

In diesem Abschnitt werden die globalen Einstellungen des TMIO Devices beschrieben.



**Abb. 28-31** Die Registerkarte „Globals“ des TMIO Devices

### Slot Number

Geben Sie hier den Slot der LabCar2 Signalbox an, in dem die TMIO-Karte steckt. Die Slots werden von links nach rechts (beginnend mit 1) durchnummeriert.

### Feedback Outputn Channel

Zur direkten Unterstützung einer eventuellen Suche nach Ursachen für eine Fehlfunktion der TMIO-Karte verfügt diese auf der Frontplatte über insgesamt vier BNC-Buchsen. Über diese können vier voneinander unabhängige Testsignale zur Beobachtung und Auswertung z.B. direkt an ein Oszilloskop angeschlossen werden.

Jedes dieser Testsignale bietet dem Anwender die Möglichkeit, über einen *Multiplexer* eines von jeweils 16 Eingangs-Signalen der Karte wieder auszugeben. Die Zusammenstellung dieser 16 möglichen „Signalquellen“ ist für jeden Testausgang unterschiedlich.

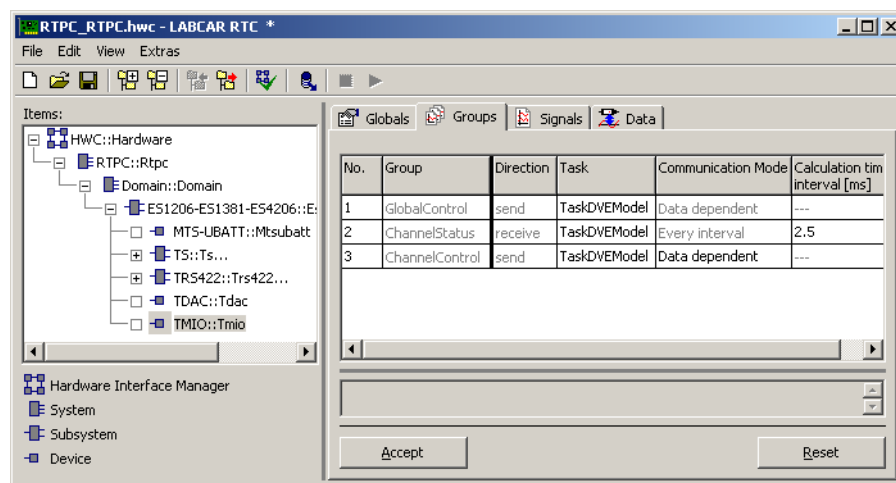
### Feedback Outputn Level

Zusätzlich zur Wahl des eigentlichen Signals hat der Anwender darüber hinaus die Möglichkeit festzulegen, von welcher Stelle des Signalfusses aus das Signal an die Testbuchse ausgegeben werden soll. Zum Einen kann es als analoges Signal unmittelbar vor dem Schwellwert-Vergleicher abgegriffen werden, wobei das Testsignal dann (technisch bedingt) eine Ausgangsamplitude von 40% der Amplitude des Eingangssignals aufweist.

Zum Anderen hat der Anwender die Möglichkeit, das Testsignal unmittelbar hinter dem Schwellwert-Vergleicher, also als TTL-Signal, abzugreifen. Durch diese Möglichkeit des Signalabgriffes vor und hinter dem Vergleicher ist eine direkte Kontrolle des eingestellten Schwellwertes sehr einfach möglich.

## 28.4.7 Groups (TMIO Device)

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TMIO Devices beschrieben.



**Abb. 28-32** Die Registerkarte „Groups“ des TMIO Devices

Die TMIO-Karte verfügt über drei Signalgruppen.

- **GlobalControl**  
Die „GlobalControl“ Signalgruppe beinhaltet das Kurbelwellenwinkel-Signal und das *Enable* Signal mit dem die gesamte Flankenerkennung und Flankenauswertung der TMIO-Karte aktiviert und deaktiviert werden kann. Die Richtung dieser Signale ist *send*, d. h. sie werden vom Simulationsmodell zur TMIO-Karte übertragen.
- **ChannelStatus**  
Die „ChannelStatus“ Signalgruppe beinhaltet sämtliche Mess- und Statussignale der Karte. Die Richtung dieser Signale ist *receive*, d. h. sie werden von der TMIO-Karte zum Simulationsmodell übertragen.
- **ChannelControl**  
Die „ChannelControl“ Signalgruppe beinhaltet die Steuer-Signale zur Einstellung der Triggerschwellen für die Schwellwertvergleiche, die Schaltsignale zur Stimulation von Steuergeräte-Eingängen und die Steuersignale zur Aktivierung bzw. Deaktivierung von Hardwarekanälen. Die Richtung dieser Signale ist *send*.

#### *Communication Mode*

---

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der TMIO-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

#### *Calculation time interval [ms]*

---

Auf dem On-Board-Transputer der TMIO-Karte läuft ein Prozess, der die Messwerterfassung und Statusermittlung der Hardwarekanäle periodisch durchführt und diese Werte in Abhängigkeit von der gewählten Einstellung im Feld „Communication Mode“ entweder immer oder nur bei veränderten Werten zum Simulationstarget überträgt. In diesem Feld wird das Zeitraster angegeben, in dem dieser Prozess gerechnet bzw. aktiviert wird.

28.4.8 Signals (TMIO Device)

In diesem Abschnitt werden die Signale der TMIO-Karte beschrieben sowie die Konfigurationsoptionen der einzelnen Signale.

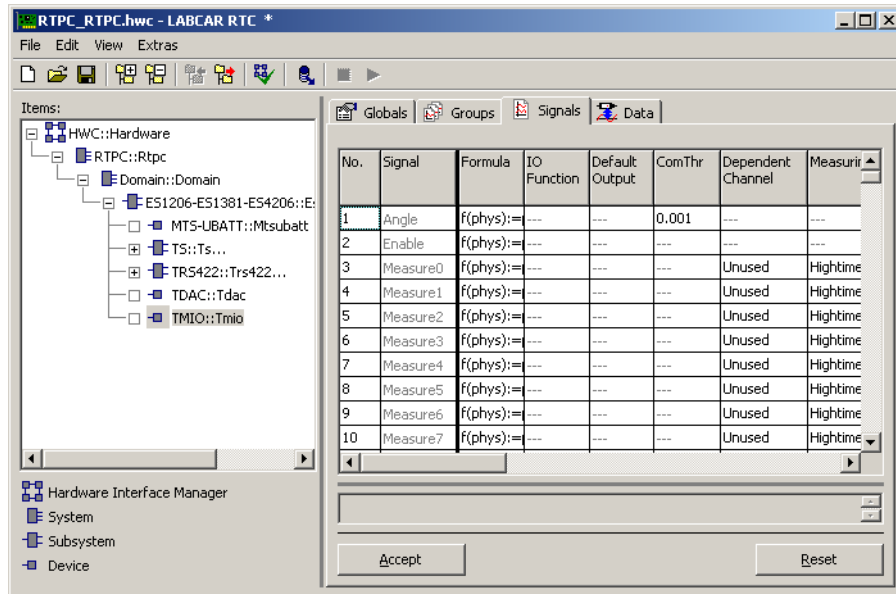


Abb. 28-33 Die Registerkarte „Signals“ des TMIO Devices

Die TMIO-Karte verfügt über die folgenden Signale:

Signal	Richtung	Bedeutung
Angle	send	Über dieses Signal erhält die TMIO-Karte ständig den aktuellen Kurbelwellenwinkel. Diese Information wird intern u.a. für Winkelsegmentierungen oder auch Timeout-Erkennungen zwingend benötigt! Das Signal ist mit dem Kurbelwellenwinkel-Ausgangssignal der TS-Karte zu verbinden.
Enable	send	Über dieses Signal kann die gesamte Flankenerkennung und -auswertung der TMIO-Karte aktiviert (true) bzw. deaktiviert (false) werden. Eine Deaktivierung der Flankenauswertung ist insbesondere dann angeraten, wenn die Signalbox nicht mit der Batteriespannung $U_{BATT}$ versorgt wird. In dieser Situation kann der interne Schwellwertvergleich für alle Kanäle, bei denen dieser relativ zur Batteriespannung eingestellt wird, nicht funktionieren, da diese Spannung eben nicht vorliegt. Folglich würde eine Auswertung der betroffenen Hardwaresignale zu Fehlern in der Simulation führen. Die Signalbox des Laborautos wird über den Batterieknoten 4 der UBATT-POW-Karte mit der Batteriespannung versorgt. Aus diesem Grund sollte das <i>Enable</i> Signal unter normalen Umständen immer mit dem Signal zur Steuerung des Batterieknoten 4 der MTS-UBATT-Karte verbunden werden.
Measure0-63	receive	Messwerte der TMIO-Karte
Timeout0-63	receive	Die <i>Timeout</i> Signale korrespondieren direkt mit den <i>Measure</i> Signalen. Zu jedem Signal <i>Measure i</i> gibt das Signal <i>Timeout i</i> an, ob bei der Ermittlung des Messwertes ein Timeout festgestellt wurde oder nicht.
Trigger0	receive	<i>Trigger0</i> ist ein 32-bit Wert, der Informationen zu den 32 Messwerten <i>Measure0</i> bis <i>Measure31</i> enthält. Jedes Bit gibt an, ob der zugehörige Messwert im letzten Simulationsschritt aktualisiert wurde (das Bit ist dann gesetzt) oder nicht (das Bit ist nicht gesetzt). Die Bitposition 0 gehört zum Messwert <i>Measure0</i> und die Bitposition 31 zum Messwert <i>Measure31</i> .
Trigger1	receive	Wie <i>Trigger0</i> , jedoch für die 32 Messwerte <i>Measure32</i> bis <i>Measure63</i> . Die Bitposition 0 gehört zum Messwert <i>Measure32</i> und die Bitposition 31 zum Messwert <i>Measure63</i> .

Signal	Richtung	Bedeutung
ActiveLevel	receive	<i>ActiveLevel</i> ist ein 32-bit Wert, der in jedem Simulationsschritt die Zustände aller 32 Hardwarekanäle angibt. Liefert der Schwellwertvergleich für Kanal $i$ ( $i = 0 \dots 31$ ) aktuell eine „0“, so ist Bit $i$ von <i>ActiveLevel</i> auf 0 gesetzt, ansonsten auf 1.
CurrentLimit	receive	<i>CurrentLimit</i> ist ein 16-bit Wert, der in jedem Simulationsschritt angibt, ob für die Schalter der I/O-Kanäle 0 bis 15 die Strombegrenzung auf 1 A aktiv ist. Ist Bit $i$ ( $i = 0 \dots 15$ ) auf 1 gesetzt, so wird der Strom für I/O-Kanal $i$ begrenzt, ist das Bit auf 0, so ist die Begrenzung nicht aktiv.
OverCurrentP/N	receive	Weist eines dieser Signale den Wert „true“ auf, so wurden sämtliche Kanäle der TMIO-Karte von der Batteriespannung abgetrennt, da ein Gesamtstrom von mehr als 10 A auf der Karte festgestellt wurde.
TrgThrSgl0-31	send	Die Signale <i>TrgThrSgl0</i> bis <i>TrgThrSgl31</i> legen die Schwellen für die Schwellwertvergleiche der 32 Hardwarekanäle fest. Der für einen Kanal eingestellte Wert wird jedoch nur berücksichtigt, wenn im Konfigurationsfeld „Trigger Threshold Source“ des Signals die Einstellung „Signal“ gewählt wird.
DisableHwChn0-31	send	Steuersignale zur Aktivierung bzw. Deaktivierung von Hardwarekanälen. Ein Wert ungleich 0 deaktiviert den zugehörigen Hardwarekanal. Werden an einem deaktivierten Kanal Messungen vorgenommen, so wird ein Timeout erkannt, falls es sich um eine Messung mit Timeout-Überprüfung handelt.
OutputSetting0-15	send	Für jeden als Schalter bzw. Ausgang genutzten I/O-Kanal $i$ der TMIO-Karte wird während der Simulation die Stellung dieses Schalters über den Wert des Signals <i>OutputSettingi</i> festgelegt.

## 28.4.9 Datentypen und Wertebereiche

In diesem Abschnitt werden die Datentypen und Wertebereiche, sowie die Bedeutung der verschiedenen Werte der Signale der TMIO-Karte beschrieben.

Signal	Datentyp	Mögliche Werte
Angle	float	0 bis 720° Kurbelwellenwinkel
Enable	bool	true = Flankenerkennung und -auswertung ist eingeschaltet false = Die gesamte Flankenerkennung und -auswertung der Karte ist ausgeschaltet
Measure0-63	float	abhängig von Art der Messung
Timeout0-63	uint	0: Bei der Ermittlung des zugehörigen Messwertes wurde <b>kein Timeout</b> festgestellt. 1: Bei der Ermittlung des zugehörigen Messwertes wurde ein <b>Timeout</b> festgestellt. 2: Der zugehörige Messwert wird zur Zeit (gemäß der Parametrierung der Karte) <b>nicht benutzt</b> .
Trigger0	uint32	
Trigger1	uint32	
ActiveLevel	uint32	
CurrentLimit	uint16	
OverCurrentP/N	bool	true = Notabschaltung aktiv false = Notabschaltung nicht aktiv
TrgThrSgl0-31	float	Kanal 0 - 7: 0.0 ... 1.0 (*U <sub>BATT</sub> ) Kanal 8-15: 0.0 V ... 35.0 V Kanal 16-23: 0.0 ... 1.0 (*U <sub>BATT</sub> ) Kanal 24-29: 0.0 V... 35.0 V Kanal Diag1 & Diag2: 0.0 V... 35.0 V
DisableHwChn0-31	bool	true = Deaktivierung des zugehörigen Hardwarekanals false = Aktivierung des zugehörigen Hardwarekanals
OutputSetting0-15	uint	0 - Der Kanal ist hochohmig 1 - Pull-Up-Beschaltung gegen Batteriespannung 2 - Pull-Down-Beschaltung gegen Batteriemasse 3 - Schaltung direkt gegen Batteriespannung 4 - Schaltung direkt gegen Batteriemasse



#### 28.4.10 Signal-Konfigurationsparameter

---

##### *IO Function*

---

Legt die Funktion und die Eingangsbeschaltung der 32 Hardware-Kanäle fest. Folgende Einstellungen sind möglich:

- *lpt*  
Der Kanal wird als Eingang verwendet und hat weder Pull-Up- noch Pull-Down Beschaltung.
- *lpt + PullUp*  
Der Kanal wird als Eingang verwendet und hat Pull-Up-Beschaltung.
- *lpt + PullDown*  
Der Kanal wird als Eingang verwendet und hat Pull-Down-Beschaltung.
- *Output*  
Der Kanal wird als Ausgang verwendet. Diese Option steht nur bei den Hardwarekanälen 0 bis 15 zur Auswahl.

##### *Default Output*

---

Über diese Listbox kann für Kanäle, die als Ausgang genutzt werden sollen, die Standardeinstellung nach der Initialisierung festgelegt werden. Es sind dieselben Optionen verfügbar wie für das Signal *OutputSetting*, welches die Ausgangsbeschaltung zur Laufzeit aus dem Modell heraus verändern kann. Die Einstellung ist nur relevant, wenn der Kanal als Ausgang benutzt wird.

##### *ComThr*

---

Diese Einstellung ist nur von Bedeutung, wenn im Feld „Communication Mode“ der zugehörigen Signalgruppe die Einstellung „Data dependent“ gewählt wurde. In diesem Fall gibt „ComThr“ an, um wieviel sich das zugehörige Signal bezogen auf seinen maximalen Wert geändert haben muss, damit ein Datentransfer zwischen dem Simulationstarget und der TMIO-Karte stattfindet. Hat man z.B. ein Winkelsignal mit dem Maximalwert 720° und gibt man für die Kommunikationsschwelle „ComThr“ einen Wert von 0.01 an, so muss sich das Winkelsignal um 1%, d.h. 7.2° ändern, damit ein Datentransfer stattfindet.

##### *Dependent Channel*

---

Über diese Listbox wird der Hardware-Kanal (0-29, Diag 1 oder Diag 2) spezifiziert, auf den sich die Messwertbestimmung beziehen soll. Wird das Messsignal *Measurei* im Modell überhaupt nicht benötigt, so ist hier (zur Vermeidung unnötiger Rechenzeiten) der Eintrag „Unused“ anzuwählen.

*Measuring Procedure*

Legt fest, welche der in Abschnitt 28.4.3 auf Seite 965 beschriebenen Messungen durchgeführt werden soll. In der Listbox bedeutet die Notation --/-- eine Auswertung an steigenden (↑) Flanken, die Notation --\-- hingegen eine Auswertung an fallenden (↓) Flanken.

Messverfahren	Erläuterung
High-/Low-Time	High- bzw. Low-Zeiten des Signals innerhalb des spezifizierten Winkelsegments
High-/Low-Time of n-th Pulse	High- bzw. Low-Zeit des n-ten Pulses innerhalb des spezifizierten Winkelsegments
Additive High-/Low-Time	Summe der High- bzw. Low-Zeiten des Signals innerhalb des spezifizierten Winkelsegments bis zur maximal vorgegebenen Pulszahl
High-/Low-Time using H-/L- Enable	Summe der High- bzw. Low-Zeiten des Signals innerhalb des spezifizierten Winkelsegments unter Berücksichtigung des Enable-Signals
High-/Low-Time using H-/L-Validate	Summe der High- bzw. Low-Zeiten des Signals innerhalb des spezifizierten Winkelsegments unter Berücksichtigung des Validate-Signals
High-/Low-Time n-th Pulse using H-/L-Enable	High- bzw. Low-Zeit des n-ten Pulses innerhalb des spezifizierten Winkelsegments unter Berücksichtigung des Enable-Signals
High-/Low-Time n-th Pulse using H-/L-Validate	High- bzw. Low-Zeit des n-ten Pulses innerhalb des spezifizierten Winkelsegments unter Berücksichtigung des Validate-Signals
Frequency	Frequenz der ansteigenden bzw. abfallenden Flanken innerhalb des spezifizierten Winkelsegments
Cycle Time	Zykluszeit der ansteigenden bzw. abfallenden Flanken innerhalb des spezifizierten Winkelsegments
Duty Factor	Tastverhältnis Low- zu High-Zeit bzw. High- zu Low-Zeit ab ansteigender bzw. abfallender Flanke innerhalb des spezifizierten Winkelsegments
Duty Cycle	Tastverhältnis Low-Zeit zu Periodendauer bzw. High-Zeit zu Periodendauer ab ansteigender bzw. abfallender Flanke innerhalb des spezifizierten Winkelsegments
Rising/Falling Edge	Winkelmessung der ansteigenden bzw. abfallenden Flanke innerhalb des spezifizierten Winkelsegments
Rising/Falling Edge of n-th Pulse	Winkelmessung der ansteigenden bzw. abfallenden Flanke des n-ten Pulses innerhalb des spezifizierten Winkelsegments

Messverfahren	Erläuterung
Number of High-/Low-Pulses	Zählung der High- bzw. Low-Pulse innerhalb des spezifizierten Winkelsegments
Total Number of High-/Low-Pulses	Zählung der High- bzw. Low-Pulse seit der letzten Initialisierung, der letzten Parameterkonfiguration oder des letzten Timeout
Level (Active High/Low)	Schwellwertvergleich mit der als „Trigger Level“ definierten Spannung

#### *Enable Validate Channel*

---

Das „Enable Validate Channel“ Feld hat nur für bestimmte Messungen eine Bedeutung. Für validierte Zeitmessungen wird in diesem Feld der Enable- bzw. Validate-Kanal spezifiziert, für andere Messungen ist dieses Feld ohne Bedeutung. Nähere Informationen zu diesen Spezialmessungen finden sich im Abschnitt 28.4.3 auf Seite 965.

#### *Timeout Check*

---

Dieses Feld legt die Art der Timeout-Überprüfung für den zugehörigen Messwert fest. Die folgenden Einstellungen sind möglich:

- Inactive  
Keine Timeout-Überprüfung.
- Intvl Predef  
Führt die Timeout-Überprüfung in den im Feld „Timeout Interval [ms]“ definierten Zeitabständen durch. Wird ein Timeout entdeckt, wird der unter „Default Timeout Value“ definierte Wert ausgegeben.
- Intvl InpDep  
Führt die Timeout-Überprüfung in den im Feld „Timeout Interval [ms]“ definierten Zeitabständen durch. Der im Timeout-Fall ausgegebene Messwert ist vom aktuellen Zustand des Eingangssignals abhängig. Wenn z.B. ein PWM-Signal konstant aktiv ist, wird 1.0 ausgegeben, ist es konstant inaktiv, wird 0.0 ausgegeben.
- CS Angle Predef  
Führt die Timeout-Überprüfung alle 720° Kurbelwellendrehung durch, und zwar an dem im Feld „CS Angle Check Point“ definierten Winkel. Wird ein Timeout entdeckt, wird der unter „Default Timeout Value“ definierte Wert ausgegeben.
- CS Angle InpDep  
Führt die Timeout-Überprüfung alle 720° Kurbelwellendrehung durch, und zwar an dem im Feld „CS Angle Check Point“ definierten Winkel. Der im Timeout-Fall ausgegebene Messwert ist vom aktuellen Zustand des Eingangssignals abhängig. Wenn z.B. ein PWM-Signal konstant aktiv ist, wird 1.0 ausgegeben, ist es konstant inaktiv 0.0.

#### *Max Pulse Count*

---

Das Feld „Max Pulse Count“ hat nur für bestimmte Messungen eine Bedeutung. Mit „Max Pulse Count“ kann für additive Messungen bzw. Pulszählungen eine Maximalzahl von zu berücksichtigenden Pulsen angegeben werden.

Bei der Vermessung eines bestimmten Pulses innerhalb eines Intervalls (n-ter Puls) geben Sie hier die Nummer des Pulses an.

#### *Default Timeout Value*

---

Default-Messwert, der im Timeout-Fall bei bestimmten im Feld `Timeout Check` gewählten Einstellungen ausgegeben wird.

#### *Timeout Interval [ms]*

---

Dieses Feld legt das Zeitintervall fest, in dem die Timeout-Überprüfung durchgeführt wird. Dieses Feld hat nur dann eine Bedeutung, wenn im Feld „Timeout Check“ die Optionen „Intvl predef“ oder „Intvl InpDep“ gewählt werden.

#### *CS Angle Check Point*

---

Dieses Feld legt den Kurbelwellenwinkel fest, an dem die Timeout-Überprüfung durchgeführt wird. Dieses Feld hat nur dann eine Bedeutung, wenn im Feld „Timeout Check“ die Optionen „CS Angle predef“ oder „CS Angle InpDep“ gewählt werden.

#### *CS Angle Lower Limit*

---

In diesem Feld wird die untere Grenze des zur Messung freigegebenen Winkelsegments angegeben (vgl. Abschnitt 28.4.5 auf Seite 973). Dieses Feld ist nur für Messungen mit Winkelsegmentierung von Bedeutung.

#### *CS Angle Upper Limit*

---

In diesem Feld wird die obere Grenze des zur Messung freigegebenen Winkelsegments angegeben (vgl. Abschnitt 28.4.5 auf Seite 973). Diese obere Segmentgrenze wird zudem bei additiven Zeitmessungen und Pulszählungen zur Übergabe des ermittelten Endwertes an das Modell genutzt (vgl. Abschnitt 28.4.3 auf Seite 965).

#### *CS Angle Reference*

---

In diesem Feld wird der Winkelbezugspunkt definiert, auf den sich alle Winkelmessungen beziehen. Für sämtliche anderen Messungen ist dieser Wert ohne Bedeutung.

#### *Trigger Threshold Source*

---

Zur Festlegung der Triggerschwellen für die Schwellwertvergleiche auf den einzelnen Hardwarekanälen bestehen zwei Möglichkeiten. Die Triggerschwellen können in Echtzeit aus dem Modell heraus gesteuert werden oder sie können vom Anwender mit einem Konfigurationsparameter in der RTIO-Benutzeroberfläche definiert werden.

In diesem Feld wird festgelegt, ob die Triggerschwelle des zugehörigen Hardwarekanals vom Modell gesteuert (Einstellung „Signal“) oder in der RTIO-Benutzeroberfläche definiert (Einstellung „Parameter“) wird.

#### *Trigger Threshold Parameter*

---

Wird im Feld „Trigger Threshold Source“ des zugehörigen Hardwarekanals die Einstellung „Parameter“ gewählt, so gibt dieses Feld die Triggerschwelle für den Schwellwertvergleich des am Kanal anliegenden Eingangssignals an. Anderenfalls hat dieses Feld keine Bedeutung.

## 28.5 TDAC - Analoge Signalerfassung und -generierung

Die TDAC-Karte (Transputer D/A-Converter-Karte) ermöglicht die Generierung und Erfassung von analogen Spannungswerten und -verläufen. Sie umfasst im einzelnen

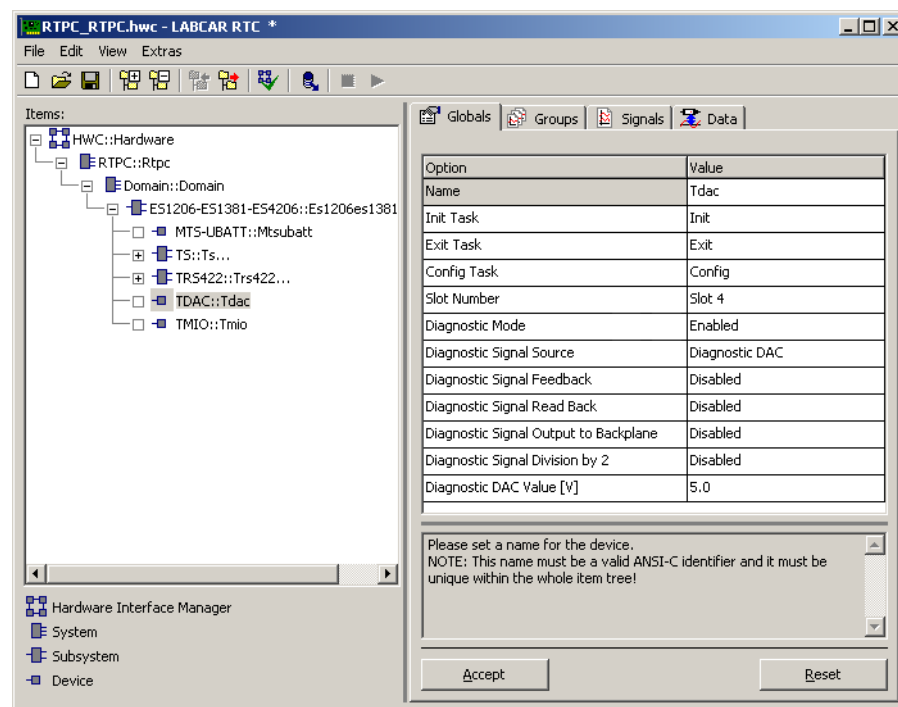
- 14 D/A-Wandler-Kanäle
- 8 A/D-Wandler-Kanäle
- 3 Relais zur freien Beschaltung durch den Anwender.

Die A/D- und D/A-Kanäle können analoge Werte direkt ein- oder ausgeben oder als Komparator genutzt werden. Die D/A Kanäle können sowohl mit interner als auch mit externer Referenzspannung betrieben werden.

Im LABCAR-Umfeld wird durch die TDAC-Karte beispielsweise die Generierung des Motortemperatur-Signals sowie die Erfassung des Winkelsignals der Drosselklappe ermöglicht.

### 28.5.1 Globals (TDAC Device)

In diesem Abschnitt werden die globalen Einstellungen des TDAC Devices beschrieben.



**Abb. 28-34** Die Registerkarte „Globals“ des TDAC Devices

#### Slot Number

Geben Sie hier den Slot der LabCar2 Signalbox an, in dem die TDAC-Karte steckt. Die Slots werden von links nach rechts beginnend mit eins durchnummeriert.

### *Diagnostic Mode*

---

Zum Zweck der Eigendiagnose verfügt die TDAC-Karte über einen 15. D/A-Wandler. Dieser ist über einen Multiplexer, an den auch **alle** anderen D/A-Kanäle angeschlossen sind, mit verschiedenen „Signalempfängern“ verbunden:

- Das Ausgangssignal des Multiplexers wird auf einer speziell für Diagnosezwecke vorgesehenen separaten Buchse an der Frontplatte der Karte ausgegeben. Somit kann ein beliebiger D/A-Wandler-Ausgang z.B. direkt an ein Oszilloskop angeschlossen werden.
- Über einen Schalter kann das Diagnose-Signal auch auf die Backplane der Signalbox ausgegeben werden, um es beispielsweise mit einer TS-Karte oder auch einer zweiten TDAC-Karte auswerten zu können.
- Ebenfalls über einen Schalter kann das Diagnose-Signal aber auch direkt (oder aber von der Amplitude her halbiert) auf den A/D-Wandler geschaltet werden. Somit ist in einem gewissen Umfang ein Selbsttest der Karte möglich.

Mit der Einstellung „enabled“ wird die Generierung des internen Diagnose-signals der Karte aktiviert.

### *Diagnostic Signal Source*

---

Legt das Ausgangssignal des Diagnose-Multiplexers und damit die Quelle für das Diagnosesignal fest. Als Quelle kann jeder beliebige D/A-Kanal (Einstellung „DAC Channel 0-13“), der zusätzliche Diagnose-D/A-Wandler („Diagnostic DAC“) oder die karteninterne Masse („GND“) ausgewählt werden. Wird der Diagnose-D/A-Wandler selektiert, muss dessen Ausgangsspannung mit der Option „Diagnostic DAC Value [V]“ festgelegt werden.

### *Diagnostic Signal Feedback*

---

Steht dieses Feld auf „enabled“, so wird das Ausgangssignal des Diagnose-Multiplexers der TDAC-Karte in das System zur weiteren Verarbeitung (Ausgabe auf der Backplane oder Zurücklesen über den A/D-Wandler) zurückgeführt und gleichzeitig an der Frontplattenbuchse ausgegeben. Wird „disabled“ eingestellt, so steht das Ausgangssignal des Diagnose-Multiplexers dem System nicht zur Verfügung.

### *Diagnostic Signal Read Back*

---

Steht diese Option auf „enabled“, wird das Diagnosesignal auf den A/D-Wandler geführt und erfasst. Zu beachten ist hierbei, dass während der Zeit, in der das Diagnosesignal auf den A/D-Wandler geführt wird, keinerlei Messungen der 8 analogen Eingänge vorgenommen werden können!

Im Normalbetrieb steht die Option auf „disabled“. In diesem Fall werden die Analogsignale der 8 Eingänge auf den A/D-Wandler geführt.

### *Diagnostic Signal Output to Backplane*

---

Wenn die Option auf „enabled“ steht, wird das Diagnosesignal bei aktiviertem Diagnosemodus auf die Backplane geschaltet.

### *Diagnostic Signal Division by 2*

---

Wenn die Option auf „enabled“ steht, wird der gemessene Wert des Diagnose-signals durch 2 dividiert.

### Diagnostic DAC Value [V]

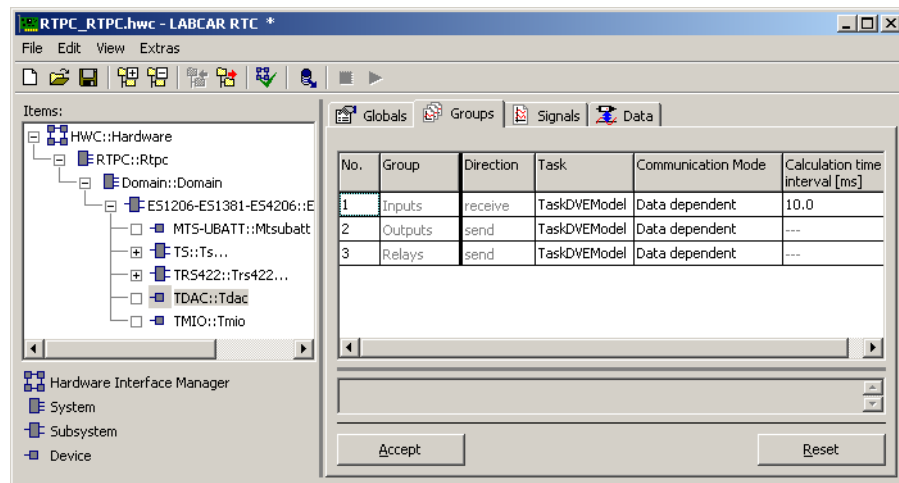
---

Legt die Ausgangsspannung des Diagnose-D/A-Wandlers fest.

### 28.5.2 Groups (TDAC Device)

---

In diesem Abschnitt werden die Signalgruppen-spezifischen Einstellungen des TDAC Devices beschrieben.



**Abb. 28-35** Die Registerkarte „Groups“ des TDAC Devices

Für die Echtzeitkommunikation mit der TDAC-Karte gibt es drei Signalgruppen:

- **Inputs**  
Die Gruppe „Inputs“ umfasst die 8 A/D-Signale und das Diagnosesignal der Karte. Die Werte dieser Signale werden von der TDAC-Karte gemessen und zum Simulationstarget übertragen. Die Übertragungsrichtung der Signale ist also *receive*.
- **Outputs**  
Die Gruppe „Outputs“ umfasst die 14 D/A-Signale der Karte. Diese Signale werden im Simulationsmodell gesetzt und zur TDAC-Karte übertragen. Die Übertragungsrichtung der Signale ist also *send*.
- **Relays**  
Zur Gruppe „Relays“ gehören die drei durch den Anwender beschaltbaren Relais. Auch diese Signale haben die Richtung *send*.

### Communication Mode

---

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der TDAC-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

### Calculation time interval [ms]

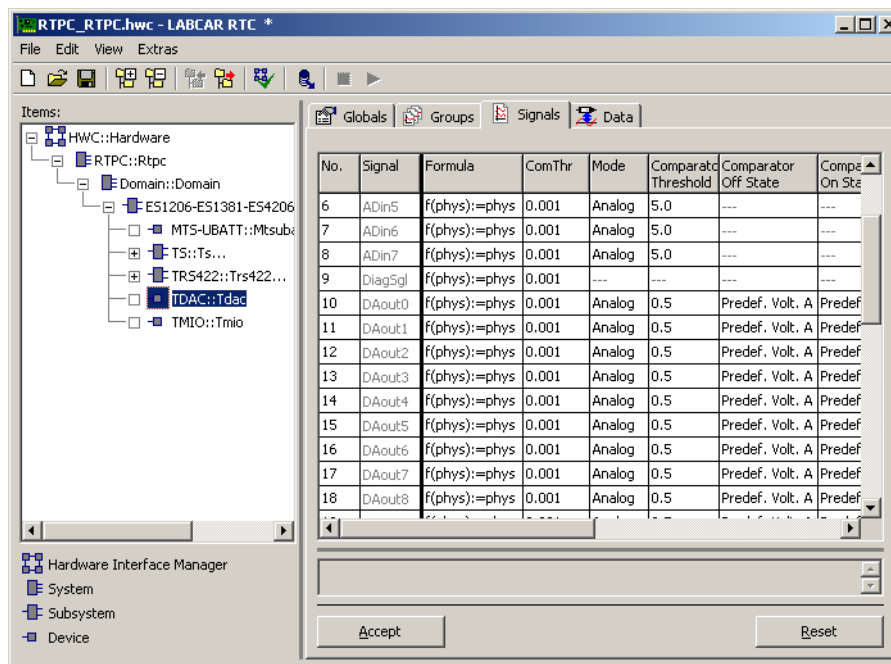
---

Auf dem On-Board-Transputer der TDAC-Karte läuft ein Prozess, der die Werte der 8 A/D-Wandler und des Diagnosesignals periodisch ermittelt und diese Werte in Abhängigkeit von der gewählten Einstellung im Feld „Communication Mode“

entweder immer oder nur bei veränderten Werten zum Simulationstarget überträgt. In diesem Feld wird das Zeitraster angegeben, in dem dieser Prozess gerechnet bzw. aktiviert wird.

### 28.5.3 Signals (TDAC Device)

In diesem Abschnitt werden die Signale der TDAC-Karte und deren Konfigurationsparameter beschrieben. Informationen zu den Wertebereichen und zur Bedeutung der verschiedenen Werte finden Sie im darauffolgenden Abschnitt.



**Abb. 28-36** Die Registerkarte „Signals“ des TDAC Devices

Die TDAC-Karte verfügt über die folgenden Signale:

Signal	Richtung	Datentypen	Bedeutung
ADinn	receive	real32, real64	Signal des A/D-Wandlereingangs <i>n</i>
DiagSgl	receive	real32, real64	Diagnosesignal
DAoutn	send	real32	Signal des D/A-Wandlersausgangs <i>n</i>
DACtrln	send	uint8	Fehlersimulationssignal für D/A-Wandlersausgang <i>n</i>
Reln	send	real32	Signal für Relaisausgang <i>n</i>

#### ComThr

Diese Einstellung ist nur von Bedeutung, wenn im Feld „Communication Mode“ der zugehörigen Signalgruppe die Einstellung „Data dependent“ gewählt wurde. In diesem Fall gibt „ComThr“ an, um wieviel sich das zugehörige Signal bezogen auf seinen maximalen Wert geändert haben muss, damit ein Datentransfer zwischen dem Simulationstarget und der TDAC-Karte stattfindet. Hat man z.B. ein Spannungssignal mit dem Maximalwert 10 V und gibt man für die



Kommunikationsschwelle `ComThr` einen Wert von 0.01 an, so muss sich das Spannungssignal um 1 %, d.h. 100 mV ändern, damit ein Datentransfer stattfindet.

### Mode

---

Mit dieser Option wird zwischen den drei verschiedenen Betriebsmodi eines D/A-Ausgangskanals bzw. eines A/D-Eingangskanals ausgewählt:

- Analog
 

Der Modus „Analog“ dient der Ausgabe analoger Spannungsverläufe auf einem D/A-Ausgangskanal bzw. der Erfassung analoger Spannungsverläufe auf einem A/D-Eingangskanal.
- Comparator
 

Im Modus „Comparator“ funktioniert der Kanal als Komparator. Bei D/A-Ausgangskanälen wird hierbei das Signal *DAOutn* ( $n = 0 \dots 13$ ) eines D/A-Ausgangskanals mit einem Schwellwert, der im Feld „Comparator Threshold“ definiert wird, verglichen. Ist der Wert des Signals *DAOutn* kleiner als der Schwellwert, so wird der im Feld „Comparator Off State“ definierte Kanalzustand eingestellt. Ist der Wert des Signals *DAOutn* größer oder gleich dem Schwellwert, so wird der im Feld „Comparator On State“ definierte Kanalzustand eingestellt. Bei A/D-Eingangskanälen wird die am Kanaleingang anliegende analoge Spannung mit der im Feld „Comparator Threshold“ definierten Schwellspannung verglichen. Ist die Spannung kleiner als die Schwellspannung, so liefert das Eingangssignal *ADinn* ( $n = 0 \dots 7$ ) den Wert 0.0, ist die Spannung größer oder gleich der Schwellspannung, so wird der Wert 1.0 geliefert. Relaisausgangskanäle können nur im Komparatormodus betrieben werden. Ist der Wert des Steuersignals *Reln* ( $n = 0, 1, 2$ ) kleiner als der Schwellwert, so wird der im Feld „Comparator Off State“ definierte Kanalzustand eingestellt. Ist der Wert des Signals *Reln* größer oder gleich dem Schwellwert, so wird der im Feld „Comparator On State“ definierte Kanalzustand eingestellt.
- Disabled
 

Im Modus „Disabled“ wird ein D/A-Ausgangskanal hochohmig geschaltet, ein A/D-Eingangskanal wird komplett abgeschaltet. Diese Abschaltung ist vor allem dann sinnvoll, wenn ein Hardwareeingang der TDAC-Karte überhaupt nicht verdrahtet ist, da der offene analoge Eingang ansonsten die Messwerte der angeschlossenen Kanäle verfälschen könnte.

### Comparator Threshold

---

Schwellenwert für den Komparatormodus. Bei A/D-Eingangskanälen sind Werte zwischen 0 und 10 V zulässig, bei D/A-Ausgangskanälen und Relaisausgangskanälen relative Werte zwischen 0 und 1.

### Comparator Off State

---

Nur für D/A- und Relais-Ausgangskanäle im Komparatormodus relevant. Die möglichen Einstellungen für D/A-Kanäle sind:

- High Impedance
 

Der Kanal ist hochohmig geschaltet.

- Predef. Volt. A  
Die Spannung, die im Feld „DA Predef. Voltage A [V]“ eingestellt ist, wird ausgegeben, wenn der Komparator im Zustand „offstate“ ist.
- +Ubatt  
Der Kanal wird auf die Batteriespannung geschaltet, wenn der Komparator im Zustand „offstate“ ist.
- -Ubatt  
Der Kanal wird auf die Batteriemasse geschaltet, wenn der Komparator im Zustand „offstate“ ist.

Mögliche Zustände für Relaisausgangskanäle:

- Open  
Der Relaiszustand „Open“ entspricht dem Komparatorzustand „offstate“.
- Closed  
Der Relaiszustand „Closed“ entspricht dem Komparatorzustand „offstate“.

#### *Comparator On State*

---

Nur für D/A- und Relais-Ausgangskanäle im Komparatormodus relevant. Die möglichen Werte für D/A-Kanäle sind:

- High Impedance  
Der Kanal ist hochohmig geschaltet.
- Predef. Volt. B  
Die Spannung, die im Feld „DA Predef. Voltage B [V]“ eingestellt ist, wird ausgegeben, wenn der Komparator im Zustand „onstate“ ist.
- +Ubatt  
Der Kanal wird auf die Batteriespannung geschaltet, wenn der Komparator im Zustand „onstate“ ist.
- -Ubatt  
Der Kanal wird auf die Batteriemasse geschaltet, wenn der Komparator im Zustand „onstate“ ist.

Mögliche Zustände für Relais:

- Open  
Der Relaiszustand „Open“ entspricht dem Komparatorzustand „onstate“.
- Closed  
Der Relaiszustand „Closed“ entspricht dem Komparatorzustand „onstate“.

#### *AD Samples*

---

Nur für A/D-Eingangskanäle im Analogmodus relevant. Diese Einstellung legt fest, aus wievielen Abtastwerten des analogen Eingangssignals ein Mittelwert gebildet wird, der dann als Ausgabewert dient. Die Mittelung der Messwerte dient der Rausch- und Störunterdrückung. Zulässige Werte liegen zwischen 1 und 100. Ist 1 als Wert für die Option „AD Samples“ ausgewählt, erfolgt keine

Mittelung. Beachten Sie bitte, dass jede Abtastung ca. 10  $\mu$ s benötigt. Hohe Werte für die Mittelung auf mehreren Kanälen wird die Rechenzeit negativ beeinflussen.

#### *DA Reference*

---

Für jeden der 14 D/A-Ausgangskanäle der TDAC-Karte kann softwaretechnisch zwischen einer internen Referenzspannung von 10 V und einer vom Anwender vorgegebenen externen Referenzspannung umgeschaltet werden. Diese Einstellung ist nur für D/A-Ausgangskanäle im Analogmodus relevant. Für D/A-Ausgangskanäle im Komparatormodus wird die interne Referenzspannung verwendet.

#### *DA Scale*

---

Die Einstellung für den Skalierungsfaktor *DA Scale* ist nur für D/A-Ausgabekanäle im Analogmodus relevant. Wird der Kanal mit der *internen* Referenzspannung betrieben, so kann die Bedeutung des Ausgangssignals *DAout<sub>n</sub>* ( $n = 0, 1 \dots 13$ ) über den in diesem Feld angegebenen Wert beeinflusst werden. Die Ausgangsspannung des D/A-Kanals wird für interne Referenz gemäß folgender Gleichung bestimmt:

$$\text{Spannung [V]} := \text{DAout}_n * \text{DA Scale}$$

Wird also beispielsweise „DA Scale“ auf den Wert 1 gesetzt, so gibt *DAout<sub>n</sub>* offensichtlich direkt die Ausgangsspannung in Volt an, der Blockeingang ist in dieser Situation also *absolut*. Wird für „DA Scale“ hingegen der Wert 10 angegeben, so liegt (für Spannungen zwischen 0 V und 10 V) der entsprechende Wert für *DAout<sub>n</sub>* zwischen 0.0 und 1.0, der Blockeingang ist folglich *relativ*.

#### *DA Predef. Voltage A [V]*

---

Ausgabewert für den Zustand „offstate“ beim Komparatormodus.

#### *DA Predef. Voltage B [V]*

---

Ausgabewert für den Zustand „onstate“ beim Komparatormodus.

### 28.5.4 Wertebereiche

---

In diesem Abschnitt werden die zulässigen Datentypen und Wertebereiche der einzelnen Signale in den verschiedenen Betriebsmodi beschrieben.

#### *ADin*

---

Die A/D Kanäle können in drei Modi betrieben werden:

- Im Modus „Disabled“ wird -1 ausgegeben. -1 wird ebenfalls ausgegeben, falls das Feld „Diagnostic Signal Read Back“ in der Registerkarte *Global* auf „Enabled“ steht.
- Im Modus „Analog“ beträgt der Wertebereich 0 bis 10 Volt.
- Im Modus „Comparator“ wird 0 zurückgegeben, wenn die gemessene Spannung kleiner als der Schwellenwert ist. Ist sie größer oder gleich, wird 1 zurückgegeben.

### *DiagSgl*

---

Wenn die Werte der 8 A/D-Wandlereingänge zurückgelesen werden (das Feld „Diagnostic Signal Read Back“ in der Registerkarte „Globals“ steht auf „Disabled“), wird -1 zurückgegeben, anderenfalls beträgt der Wertebereich 0 bis 10 Volt.

### *DAout*

---

Im Kanalmodus „Analog“ richtet sich der Wertebereich der DAout-Signale danach, ob die interne oder eine externe Referenzspannung für den jeweiligen D/A-Ausgabekanal verwendet wird.

- Wird die interne Referenzspannung verwendet, so gilt für den Wertebereich der DAout-Signale  

$$0 \leq \text{DAout} \leq 10.0 \text{ V} / \text{DA Scale}$$
 „DA Scale“ ist dabei der in der Registerkarte „Signals“ für den Kanal eingestellte Skalierungsfaktor.
- Wird die externe Referenzspannung verwendet, so berechnet sich die Ausgangsspannung des D/A-Kanals nach folgender Gleichung  

$$\text{Spannung [V]} := \text{DAout} * \text{Externe Referenzspannung}$$
 Die Werte für die DAout-Signale müssen demnach zwischen 0.0 und 1.0 liegen.

Im Kanalmodus „Comparator“ sollten die Werte für die DAout-Signale zwischen 0.0 und 1.0 liegen.

Im Kanalmodus „Disabled“ haben die Werte der DAout-Signale keine Bedeutung.

### *DACtrl*

---

Durch die Einstellungen des Signals *DACtrl* können auf dem jeweiligen D/A-Kanal verschiedene Fehlerzustände simuliert werden. Je nach dem Wert, der dem Signal zugewiesen wird, wird am entsprechenden Ausgabekanal ein Fehlerzustand simuliert. Das Signal kann folgende Werte annehmen:

Wert	Bedeutung
0	Normaler Betrieb, keine Fehlersimulation. Das Augabesignal ergibt sich (in Abhängigkeit von der Kanal-Konfiguration) aus dem Wert des DAout-Signals.
1	Geberabfall wird simuliert, der Kanalausgang wird hochohmig geschaltet
2	Kurzschluss gegenüber der Batteriespannung wird simuliert
3	Kurzschluss gegenüber der Batteriemasse wird simuliert

Im Falle der Fehlersimulation, d.h., wenn das DACtrl-Signal die Werte 1, 2 oder 3 annimmt, hat das zum Kanal gehörige DAout-Signal keine Bedeutung.

### *Rel*

---

Die Werte für die Relais-Steuersignale sollten im Bereich zwischen 0.0 und 1.0 liegen.

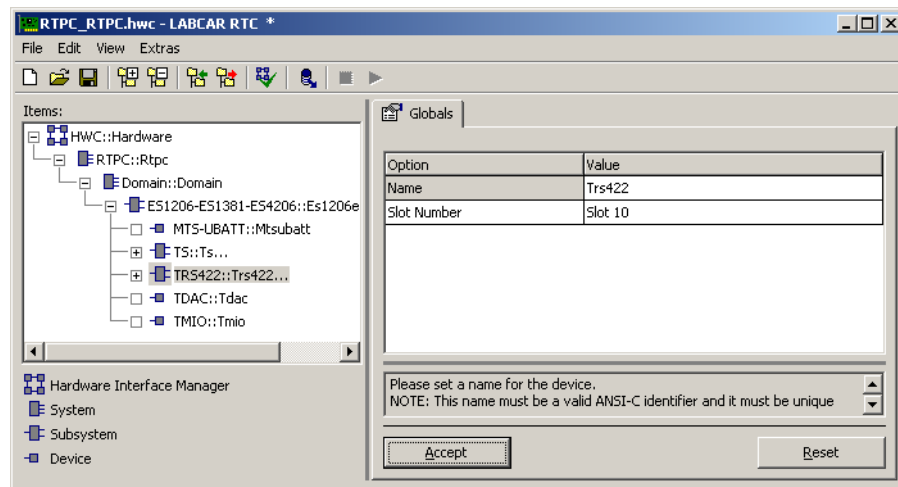
## 28.6 TRS422

Mit der TRS422-Karte ist es möglich, maximal 2 Transputermodule (TRAMs), die im Laborauto-Umfeld als Funktionsmodule eingesetzt werden, mit bis zu drei verschiedenen optionalen LABCAR-Komponenten (VME-Erweiterung, Lastbox, Bedienpult o.ä.) zu verbinden. Als Schnittstelle wird die symmetrische Standard-schnittstelle RS422A eingesetzt. Die LABCAR-OPERATOR Software unterstützt von diesen durch die Hardware gegebenen Möglichkeiten die Ankopplung von einer oder mehreren Lastboxen. Die softwaremäßige Unterstützung einer VME-Erweiterung ist im VMEbus basierten LABCAR System nicht mehr erforderlich.

Zur Ankopplung von Lastboxen werden Transputer-Links der TRAM-Module, die in den beiden TRAM-Slots stecken können, auf Sub-D-Stecker oder Buchsen in der Frontplatte der TRS422-Karten geführt. Über ein serielles Transputer-Link-Kabel lässt sich so eine Verbindung zur T8VME-Einschubkarte einer Lastbox herstellen. Diese T8VME-Einschubkarte zur Ankopplung einer Lastbox wird im folgenden, zur besseren Unterscheidung von T8VME-Karten, die zur Umsetzung des parallelen VMEbus auf den seriellen Transputer-Link-Bus dienen, als T8IO-Karte bezeichnet.

### 28.6.1 Globals (TRS422 Subsystem)

In diesem Abschnitt werden die globalen Einstellungen der TRS422-Karte beschrieben.



**Abb. 28-37** Die Registerkarte „Globals“ des TRS422 Subsystems

#### Slot Number

Geben Sie hier den Slot der LabCar2 Signalbox an, in dem die TRS422-Karte steckt. Die Slots werden von links nach rechts beginnend mit eins durchnummeriert.

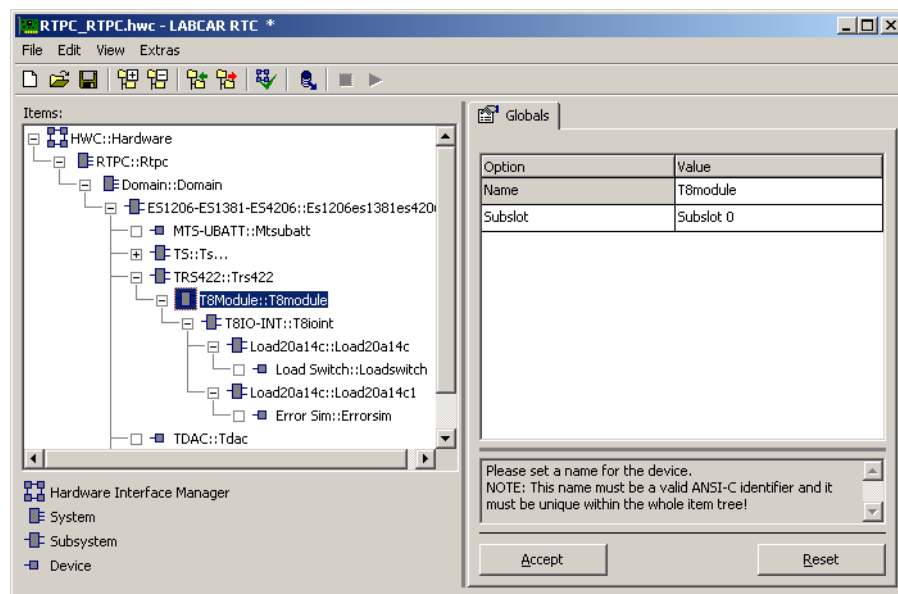
## 28.7 T8Module Subsystem

Dem TRS422 Subsystem können bis zu zwei T8Module Subsystems zugeordnet werden. Diese entsprechen den physikalisch in den Slots der TRS422-Karte gesteckten Aufsteckmodulen. Dabei spielt es keine Rolle, ob es sich bei diesen Modulen um TRAM- oder TCAN-Module handelt.

Lastboxen, die über Transputer-Links der T8-Module angekoppelt sind, werden im Hierarchybaum des RTIO-Editors den T8-Modulen zugeordnet, wie nachfolgend bei der Beschreibung des T8IO Subsystems erläutert wird.

### 28.7.1 Globals (T8Module Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des T8Module Subsystems beschrieben.



**Abb. 28-38** Die Registerkarte „Globals“ des T8Module Subsystems

#### Subslot

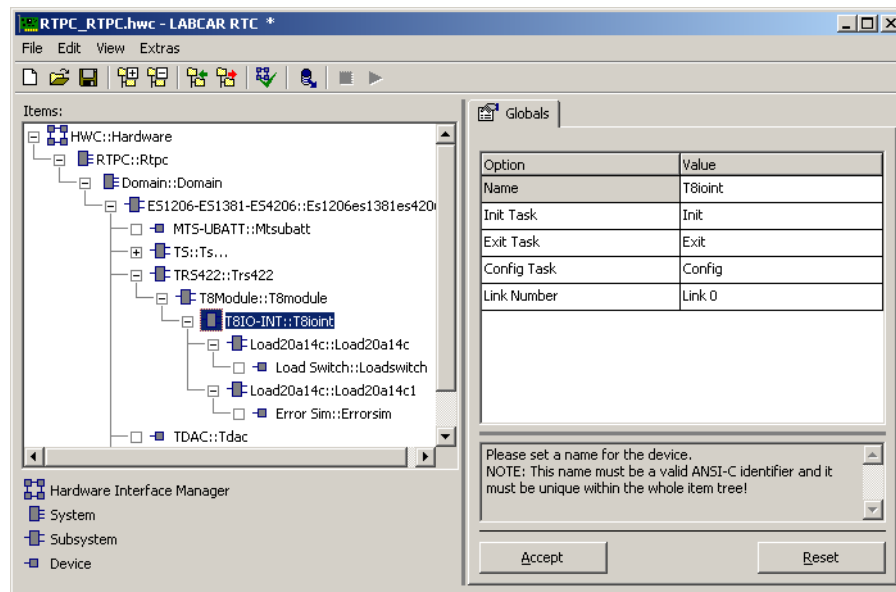
Geben Sie hier den Slot der TRS422-Karte an, in dem das TRAM- oder TCAN-Modul steckt.

## 28.8 T8IO

Die T8IO-Karte dient zum Anschluss einer Lastbox mit VME-Backplane an eine Transputer-basierte Signalbox. Die Karten der Lastbox befinden sich im Hierarchiebaum des RTIO-Editors unterhalb der T8IO-Karte und werden durch diese gesteuert. Im Gegensatz zur früheren Software-Generation wird die Berechnung von Modellteilen auf der T8IO-Karte von LABCAR-OPERATOR nicht mehr unterstützt.

### 28.8.1 Globals (T8IO Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des T8IO Subsystems beschrieben.



**Abb. 28-39** Die Registerkarte „Globals“ des T8IO Subsystems

Die Init- Exit- und Config-Tasks für die T8IO-Karte gelten auch für alle angeschlossenen Lastkarten.

#### *Link Number*

Geben Sie hier an, über welchen Link des zugeordneten T8-Moduls die T8IO-Karte angeschlossen ist.

## 28.9 LOAD20A14C Hochstrom-Lastschaltkarte

---

Die Hochstrom-Lastschaltkarte Load20A14C kann in zwei alternativen Betriebsarten eingesetzt werden:

- als Umschalter von ECU-Anschlüssen zwischen verschiedenen Lasten
- als Kurzschlussimulator für ECU-Anschlüsse

In der ersten Betriebsart lassen sich Originallasten oder Lastnachbildungen softwaregesteuert auf Anschlüsse des Steuergeräts schalten. Hierzu stehen 14 Kanäle zur Verfügung, mit denen jeweils ein Anschluss des Steuergeräts zwischen zwei Lastanschlüssen umgeschaltet werden kann.

Die zweite Betriebsart der Karte bietet die Möglichkeit, Kurzschlüsse der Steuergeräteanschlüsse gegen Masse bzw. die Versorgungsspannung zu erzeugen. Die Lastanschlüsse lassen sich hierzu mit einem Kurzschlussbus verbinden. Wird der Kurzschlussbus gegen Masse oder die Versorgungsspannung geschaltet, misst ein Stromsensor den fließenden Strom. Ein dem Strom proportionales Spannungssignal ist auf einen Anschluss des Control-Ports auf der Frontplatte geführt.

In beiden Betriebsarten vergleicht eine Statusauswertung für den Steuergeräte- oder Last-Anschluss eines jeden Kanals dessen Spannungswert mit zwei festen Schwellwerten. Das Ergebnis des Vergleichs wird mit Leuchtdioden an der Frontplatte angezeigt und steht über ein Statusregister für eine Weiterverarbeitung zur Verfügung.

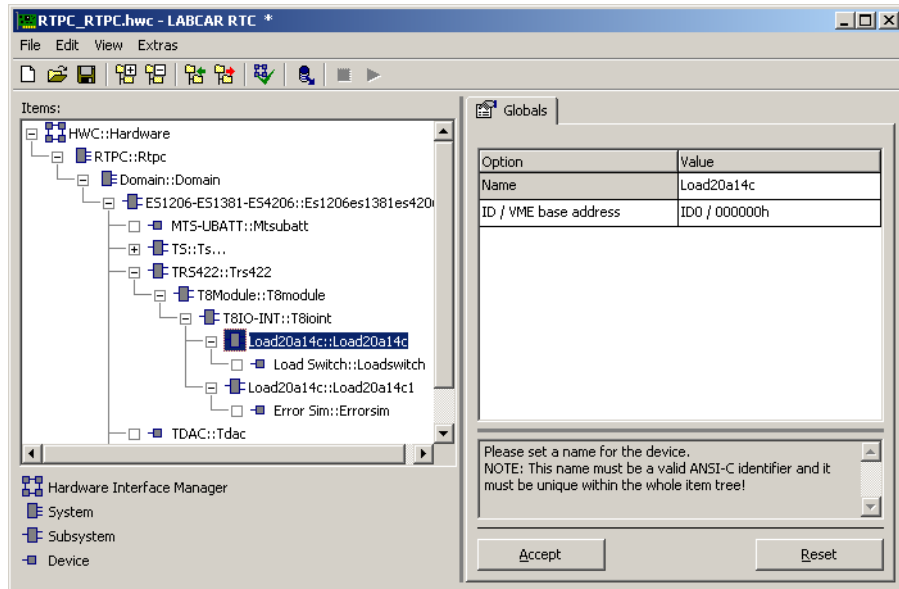
Je ein Steuergeräteanschluss lässt sich softwaregesteuert zur Spannungsmessung auf einen Anschluss des Control-Ports schalten. Dort lässt sich beispielsweise der Spannungsverlauf des Steuergeräteanschluss mit einem Oszilloskop beobachten.

Die Load20A14C-Karte lässt sich entweder als Schalterkarte oder zur Kurzschlussimulation einsetzen. Die Einschubkarte wird von ETAS für den entsprechenden Anwendungsfall konfiguriert. Für beide Betriebsarten gibt es jeweils ein Device im RTIO-Editor. Für jede Instanz einer Load20A14C-Karte kann im RTIO-Editor nur jeweils eines der beiden Devices angelegt werden, welches zur auf der Karte eingestellten Betriebsart passen muss.



28.9.1 Globals (Load20A14C Subsystem)

In diesem Abschnitt werden die globalen Einstellungen des Load20A14C Subsystems beschrieben.

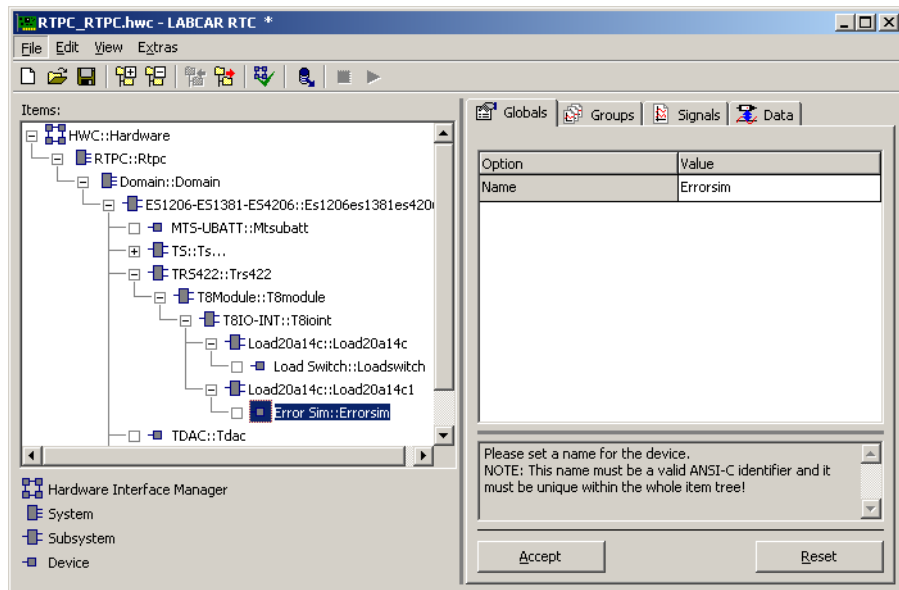


**Abb. 28-40** Die Registerkarte „Globals“ des Load20A14C Subsystems  
*ID / VME base address*

Legt die VME-Basisadresse der Karte fest.

28.9.2 Globals (ErrorSim Device)

In diesem Abschnitt werden die globalen Einstellungen des ErrorSim Devices beschrieben.

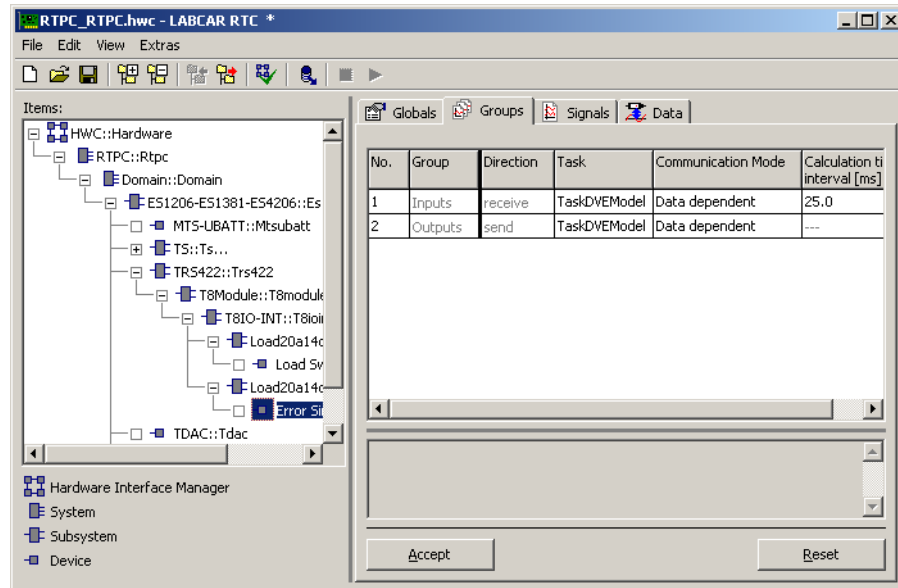


**Abb. 28-41** Die Registerkarte „Globals“ des ErrorSim Devices

Das ErrorSim-Device dient der Steuerung der Load20A14C-Karte in der Betriebsart *Kurzschlussimulation*. Softwaregesteuert lässt sich jeweils ein Steuergeräteaanschluss auf die Batteriespannung oder die LABCAR-Masse schalten. Der dabei fließende Strom wird mit einer Sicherung begrenzt. Der Status der Sicherung steht als Eingangssignal dem Simulationsmodell zur Verfügung.

### 28.9.3 Groups (ErrorSim Device)

In diesem Abschnitt werden die Signalgruppen des ErrorSim Devices beschrieben.



**Abb. 28-42** Die Registerkarte „Groups“ des ErrorSim Devices

Das ErrorSim-Device verfügt über die folgenden Signalgruppen:

Signalgruppe	Richtung	Bedeutung
Inputs	receive	Anzeige des Status der Überstromsicherung und der Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen.
Outputs	send	Auswahl des Messkanals, des ECU-Kanals für die Kurzschlussimulation und der Kurzschluss simulationsart.

#### Communication Mode

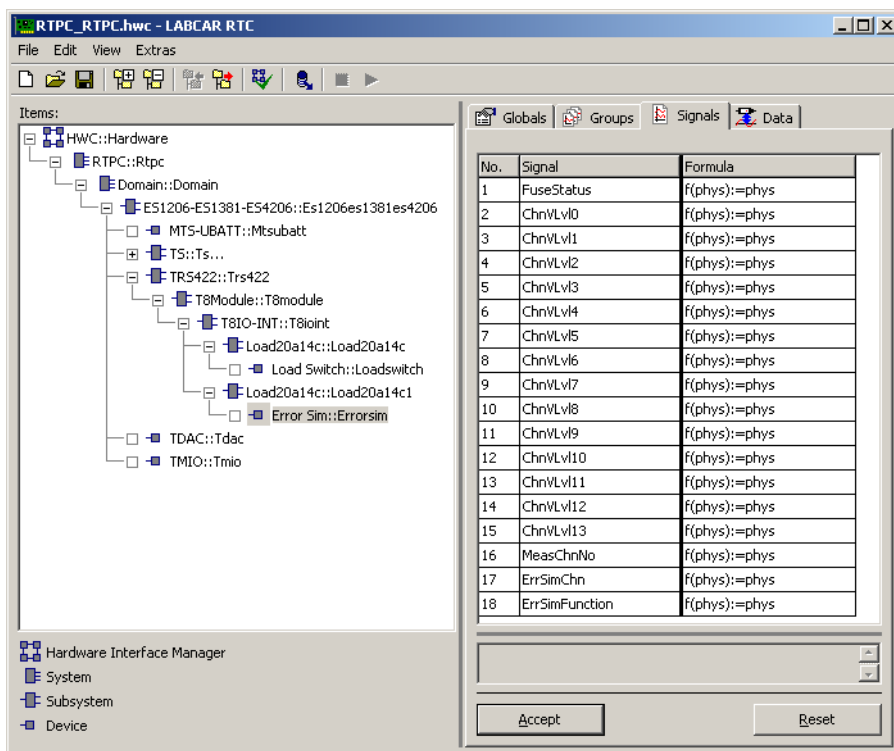
In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der Load20A14C-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

### Calculation time interval [ms]

Auf dem On-Board-Transputer der T8IO-Karte läuft ein Prozess, der den Status der Überstromsicherung und die Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen periodisch ermittelt und diese Daten in Abhängigkeit von der gewählten Einstellung im Feld „Communication Mode“ entweder immer oder nur bei veränderten Werten zum Simulationstarget überträgt. In diesem Feld wird das Zeitraster angegeben, in dem dieser Prozess gerechnet bzw. aktiviert wird.

#### 28.9.4 Signals (ErrorSim Device)

In diesem Abschnitt werden die Signale des ErrorSim Devices und deren Einstellungen beschrieben.



**Abb. 28-43** Die Registerkarte „Signals“ des ErrorSim Devices

Das ErrorSim-Device verfügt über die folgenden Signale:

Signal	Richtung	Bedeutung
FuseStatus	receive	Gibt den Status der Überstromsicherung auf der Karte an
MeasChnNo	send	Wählt den Messkanal aus, der an einen Anschluß des Control-Port auf der Frontplatte der Load20A14C-Karte geschaltet wird

Signal	Richtung	Bedeutung
ChnVLvl0-13	receive	Diese Signale zeigen die Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen an.
ErrSimChn	send	Wählt den Kanal aus, an dem eine Kurzschluß- bzw. Fehlersimulation durchgeführt wird
ErrSimFunction	send	Legt die Art der Kurzschluß- bzw. Fehlersimulation am ausgewählten Kanal fest

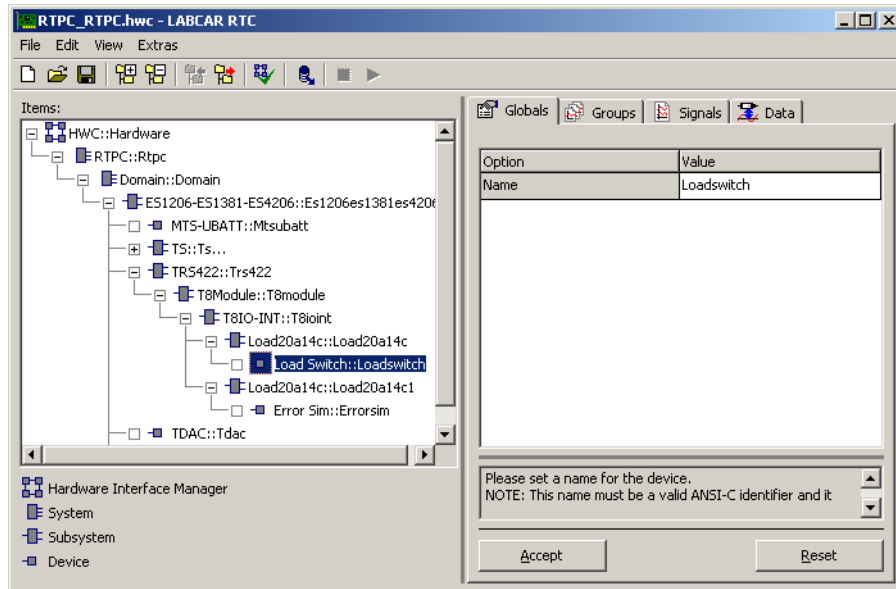
### 28.9.5 Datentypen und Wertebereiche

In diesem Abschnitt werden die zulässigen Datentypen und Wertebereiche der einzelnen Signale des ErrorSim-Device, sowie die Bedeutung der verschiedenen Werte beschrieben.

Signal	Datentyp	Bedeutung
FuseStatus	bool, int	0 bzw. false: Überstromsicherung ist ok ≠ 0 bzw. true: Überstromsicherung ist durchgebrannt
MeasChnNo	int	0 -13: Entspricht der Nummer des Messkanals Andere Werte: Es wird kein Messkanal auf den Control-Port an der Frontplatte durchgeschaltet
ChnVLvl0-13	int	0: $V_{\text{Steuergerätepin}} < 2,5 \text{ V}$ 1: $V_{\text{Steuergerätepin}} > 3,0 \text{ V}$ 2: $2,5 \text{ V} \leq V_{\text{Steuergerätepin}} \leq 3,0 \text{ V}$ -> Kanal ist nicht angeschlossen $V_{\text{Lastpin}}$ = Spannung am Pin des Anschlusses „LOAD A“
ErrSimChn	int	0 -13: Entspricht der Nummer des Fehlersimulationskanals Andere Werte: Es wird kein Kanal zur Fehlersimulation ausgewählt -> Fehlersimulation wird abgeschaltet
ErrSimFunction	int	1: Kurzschluss gegen Batteriespannung 2: Kurzschluss gegen Batteriemasse Andere Werte: Signalunterbrechung

28.9.6 Globals (LoadSwitch Device)

In diesem Abschnitt werden die globalen Einstellungen des LoadSwitch Devices beschrieben.

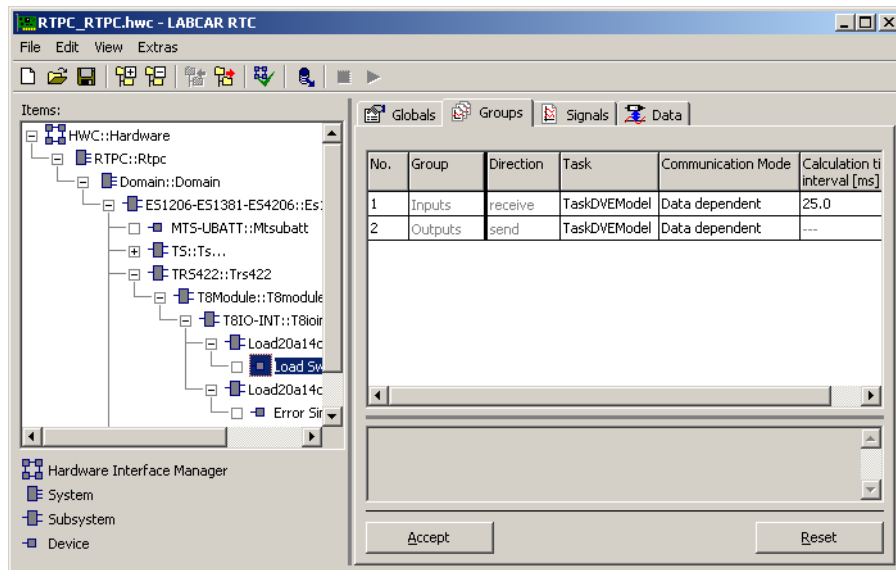


**Abb. 28-44** Die Registerkarte „Globals“ des LoadSwitch Devices

Das LoadSwitch-Device dient der Steuerung der Load20A14C-Karte in der Betriebsart *Lastumschaltung*. Mit jedem Schaltkanal der Karte lässt sich softwaregesteuert ein Anschluss des Steuergeräts zwischen zwei Lastanschlüssen umschalten. Die Lastanschlüsse werden als Load A und Load B bezeichnet.

28.9.7 Groups (LoadSwitch Device)

In diesem Abschnitt werden die Signalgruppen des LoadSwitch Devices beschrieben.



**Abb. 28-45** Die Registerkarte „Groups“ des LoadSwitch Devices

Das LoadSwitch-Device verfügt über die folgenden Signalgruppen:

Signalgruppe	Richtung	Bedeutung
Inputs	receive	Anzeige des Status der Überstromsicherung und der Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen
Outputs	send	Auswahl des Messkanals und Steuerung der Lastumschaltung

#### Communication Mode

In diesem Feld wird festgelegt, wann ein Datentransfer zwischen dem Simulationstarget und der Load20A14C-Karte stattfindet. Wird die Einstellung „Every interval“ gewählt, findet der Datentransfer in jedem Rechenschritt statt. Wird die Einstellung „Data dependent“ gewählt, werden die Daten nur dann übertragen, wenn sie sich geändert haben. Die Einstellung „Data dependent“ wird empfohlen, um den Datentransfer auf dem seriellen Transputer-Link-Bus zu minimieren.

#### Calculation time interval [ms]

Auf dem On-Board-Transputer der T8IO-Karte läuft ein Prozess, der den Status der Überstromsicherung und die Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen periodisch ermittelt und diese Daten in Abhängigkeit von der gewählten Einstellung im Feld „Communication Mode“ entweder immer oder nur bei veränderten Werten zum Simulationstarget überträgt. In diesem Feld wird das Zeitraster angegeben, in dem dieser Prozess gerechnet bzw. aktiviert wird.

### 28.9.8 Signals (LoadSwitch Device)

In diesem Abschnitt werden die Signale des LoadSwitch Devices und deren Einstellungen beschrieben.

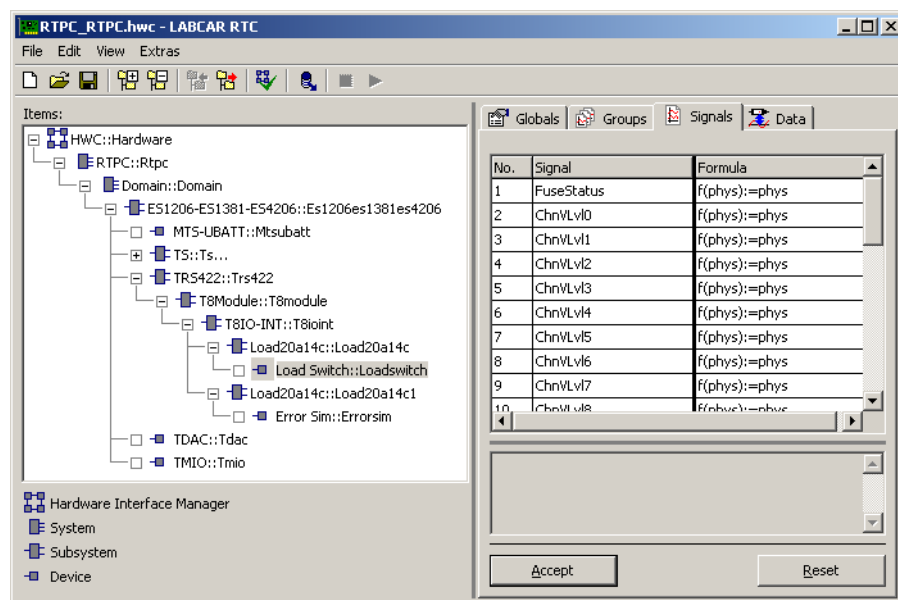


Abb. 28-46 Die Registerkarte „Signals“ des LoadSwitch Devices

Das LoadSwitch-Device verfügt über die folgenden Signale:

Signal	Richtung	Bedeutung
FuseStatus	receive	Gibt den Status der Überstromsicherung auf der Karte an
MeasChnNo	send	Wählt den Messkanal aus, der an einen Anschluss des Control-Port auf der Frontplatte der Load20A14C-Karte geschaltet wird
ChnVlvl0-13	receive	Diese Signale zeigen die Ergebnisse der Schwellwertvergleiche auf den 14 Statuskanälen an
LdSwitchCtrl0-13	send	Legt fest, auf welchen Lasteingang der entsprechende Schaltkanal geschaltet wird

### 28.9.9 Datentypen und Wertebereiche

In diesem Abschnitt werden die zulässigen Datentypen und Wertebereiche der einzelnen Signale des LoadSwitch Devices sowie die Bedeutung der verschiedenen Werte beschrieben.

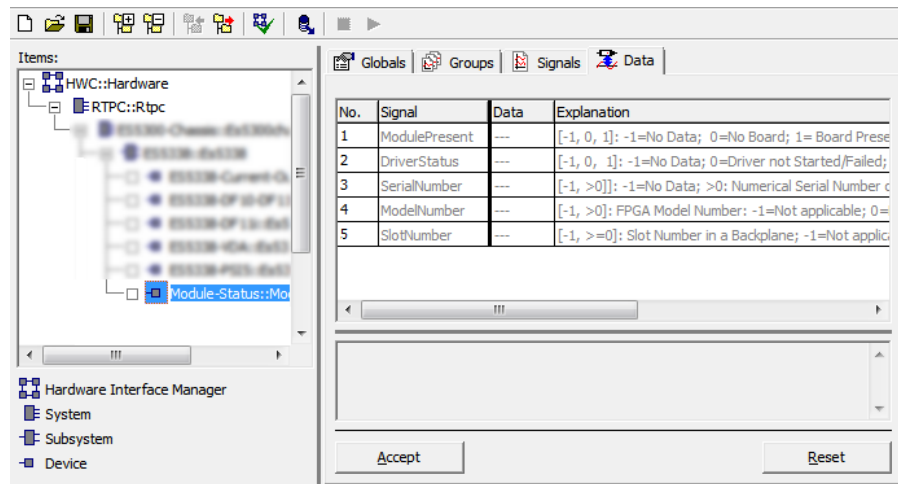
Signal	Datentyp	Bedeutung
FuseStatus	bool, int	0 bzw. false: Überstromsicherung ist ok ≠ 0 bzw. true: Überstromsicherung ist durchgebrannt
MeasChnNo	int	0 -13: Entspricht der Nummer des Messkanals Andere Werte: Es wird kein Messkanal auf den Control-Port an der Frontplatte durchgeschaltet
ChnVlvl0-13	int	0: $V_{\text{Steuergerätepin}} < 2,5 \text{ V}$ 1: $V_{\text{Steuergerätepin}} > 3,0 \text{ V}$ 2: $2,5 \text{ V} \leq V_{\text{Steuergerätepin}} \leq 3,0 \text{ V}$ -> Kanal ist nicht angeschlossen.  $V_{\text{Steuergerätepin}} = \text{Spannung am Steuergerätepin}$
LdSwitchCtrl0-13	int	0 - Schaltung nach Load A 1 - Schaltung nach Load B





## 29 Module Status

Das Module-Status Device liefert Informationen über das Vorhandensein und den Status des Moduls und des Hardwaretreiber sowie Serien- und Modellnummer der jeweiligen Hardware.



**Abb. 29-1** Die Registerkarte „Data“ des Module-Status RTIO-Elements  
*Module Present*

Dieses Signal enthält Informationen über das Vorhandensein der Karte.

*DriverStatus*

Dieses Signal enthält Informationen über den Status des Hardwaretreibers.

*SerialNumber*

Die Seriennummer der Karte (sofern vorhanden).

*ModelNumber*

Die Modellnummer der Karte (sofern vorhanden).

*SlotNumber*

Die Nummer des Steckplatzes in einer bestimmten Backplane (if available).

<b>Signalname</b>	<b>Anmerkungen</b>
ModulePresent	-1: Keine Daten verfügbar 0: Karte nicht vorhanden 1: Karte vorhanden
DriverStatus	-1: Keine Daten verfügbar 0: Treiber nicht gestartet. 1: Treiber gestartet
SerialNumber	-1: Keine Daten verfügbar > 0: Seriennummer
ModelNumber	-1: Keine Daten verfügbar > 0: Modellnummer
SlotNumber	-1: Keine Daten verfügbar 0: Nummer des Steckplatzes

**Tab. 29-1** Module-Status: Signale der Registerkarte „Data“

## 30 **ETAS Kontaktinformation**

---

### *ETAS Hauptsitz*

---

ETAS GmbH

Borsigstraße 14

70469 Stuttgart

Deutschland

Telefon: +49 711 3423-0

Telefax: +49 711 3423-2106

WWW: [www.etas.com](http://www.etas.com)

### *ETAS Regionalgesellschaften und Technischer Support*

---

Informationen zu Ihrem lokalen Vertrieb und zu Ihrem lokalen Technischen Support bzw. den Produkt-Hotlines finden Sie im Internet:

ETAS Regionalgesellschaften      WWW: [www.etas.com/de/contact.php](http://www.etas.com/de/contact.php)

ETAS Technischer Support      WWW: [www.etas.com/de/hotlines.php](http://www.etas.com/de/hotlines.php)



---

## Index

**A**

additive Pulsweitenmessungen 554  
additive Zeit 967

**C**

CAN-CTRL Subsystem 57  
  Globals 57, 403

CAN-ESTAT Device  
  Data 420  
  Globals 416  
  Groups 416  
  Signals 419

CAN-IO Device 67  
  Data 415  
  Globals 67, 407  
  Groups 70, 410  
  Signals 72, 412

CONFIG-CTRL Device 58  
  Groups 58  
  Signals 59

**E**

ErrorSim Device  
  Globals 997  
  Groups 998  
  Signals 999

ES1206-ES1381-ES4206 Subsystem  
  Globals 920

ES1220.1 CAN Board 55

ES1220-CAN Subsystem 56  
  Globals 56

ES1302.1 A/D Board 75

ES1302-AD Device  
  Globals 75  
  Groups 76  
  Signals 77

ES1321 Subsystem  
  Globals 80

ES1321.1 PWM I/O Board 79

ES1321-In Subsystem  
  Globals 81

ES1321-In-ASM Device  
  Globals 100  
  Signals 103

ES1321-In-HW Device  
  Globals 82

ES1321-In-Meas Device  
  Globals 84

ES1321-In-SENT Device  
  Data 98

ES1321-In-SENT Device  
  Globals 94  
  Signals 97

ES1321-Out Subsystem  
  Globals 104

ES1321-Out-Dig-Pwm Device  
  Data 107  
  Signals 105

- ES1321-Out-Multi-Pulse Device
  - Data 111
  - Globals 108
- ES1321-Out-SENT Device
  - Data 115
- ES1330.1 PWM I/O Counter Board 119
- ES1330-PWM Subsystem
  - Globals 119
- ES1331.1 Signal Generator Board 125
- ES1331-DSP Subsystem
  - Globals 126
- ES1332
  - Aufbau des Item-Baums 139
- ES1332.1 Arbitrary Signal Generator Board 139
- ES1332-DSP Subsystem
  - Globals 140
- ES1334
  - RTIO-Baum 154
- ES1334.1 Measurement Board 149
- ES1334.2
  - RTIO-Baum 192
- ES1334.2 Measurement Board 187
- ES1334.2-Hw Device
  - „Control“ Signalgruppe 198
  - „Knock“ Signalgruppe 199
  - „Level“ Signalgruppe 199
  - Globals 194
  - Groups 197
  - Signals 201
- ES1334.2-Meas Device
  - „MeasVal“ Signalgruppe 204
  - Globals 203
  - Groups 203
  - Signals 205
- ES1334.2-VMI Subsystem
  - Globals 193
- ES1334-Hw Device
  - "Knock" Signalgruppe 161
  - "Level" Signalgruppe 160
  - Globals 156
  - Groups 159
  - Signals 162
- ES1334-Meas Device
  - "MeasVal" Signalgruppe 165
  - Globals 164
  - Signals 166
- ES1334-VMI Subsystem
  - Globals 155
- ES1335 Arbitrary Signal Generator Board 241
- ES1335 Subsystem
  - Globals 246
- ES1335-Knock Device
  - Data 268
  - Globals 264
  - Groups 266
  - Signals 267
- ES1335-Misfire Device
  - Data 262
  - Globals 259
  - Groups 261
  - Signals 261
- ES1335-MSA-Sensor Device
  - Data 277
  - Globals 274
  - Groups 276
- ES1335-OnlineWaveformAccess Device
  - Data 271
- ES1335-OutputMux Device
  - Data 250
  - Globals 249
  - Groups 249
  - Signals 249
- ES1335-Powertrain Subsystem
  - Globals 247
- ES1335-Rpm Device
  - Data 258
  - Globals 256
  - Groups 257
- ES1335-Sig Device
  - Data 252
  - Globals 251
  - Groups 252
- ES1335-Trigger-Inputs Device
  - Data 280
- ES1336.1 Angle Synchronous Measurement Board 281
- ES1336-Hw Device
  - Globals 292
  - Groups 295
  - Signals 296
- ES1336-Meas Device
  - Globals 299
  - Groups 299
  - Signals 301
- ES1336-Rpm Device
  - Globals 288
  - Groups 290
  - Signals 292
- ES1337 Subsystem 320
  - Globals 320

- ES1337.1 Wheelspeed Sensor Simulation Board 319
  - Aufbau des RTIO-Baums 319
- ES1337-Wheelsnsrsim Subsystem
  - Globals 320
- ES1337-Wheelsnsrsim-DA Device 322
  - Data 323
  - Globals 322
  - Groups 323
- ES1337-Wheelsnsrsim-DF10-DF11s Device
  - Data 332
  - Globals 330
  - Groups 331
- ES1337-Wheelsnsrsim-DF10-RotDir Device
  - Data 337
  - Globals 334
  - Groups 336
- ES1337-Wheelsnsrsim-DF11i Device
  - Data 342
  - Globals 339
  - Groups 341
- ES1337-Wheelsnsrsim-DF6 Device
  - Data 328
  - Globals 327
  - Groups 328
- ES1337-Wheelsnsrsim-VDA Device
  - Data 347
  - Globals 344
  - Groups 346
- ES1385 Device
  - Globals 351
  - Groups 354
  - Signals 356
- ES1385 Resistor Cascade Board 351
- ES1391.1 Power Supply Controller Board 357
- ES1391-PWR Subsystem
  - Globals 358
- ES1391-PwrCtrl Device
  - Globals 361
  - Groups 364
  - Signals 368
- ES1391-SwCtrl Device
  - Globals 370
  - Groups 371
  - Signals 374
- ES1650.1 Piggyback Carrier Board 377
- ES1650-CB Subsystem
  - Globals 377
- ES1651.1 Carrier Board 391
- ES1651-CAN Subsystem
  - Globals 402
- ES1651-CB Subsystem
  - Globals 392
- ES1651-CTRL Device
  - Data 401
  - Globals 394
  - Groups 398
  - Signals 400
- ES4205 Subsystem
  - Globals 920
- ES4315-CTRL Device
  - Globals 472
  - Groups 474
  - Signals 475
- ES4315-VXI Subsystem
  - Globals 471
- ES4320-CTRL Device
  - Globals 478
  - Groups 480
  - Signals 481
- ES4320-Misfire 518
- ES4320-Misfire Device
  - Globals 519
  - Groups 522
  - Signals 522
- ES4320-RPM Device
  - Globals 483
  - Groups 484
  - Signals 487
- ES4320-SIG-ARB Device
  - Globals 495
  - Groups 500
  - Signals 501
- ES4320-SIG-CTRL Device
  - Groups 489
  - Signals 491
- ES4320-SIG-KNOCK Device
  - Globals 504
  - Groups 508
  - Signals 509
- ES4320-SIG-PWM Device
  - Globals 492
  - Groups 493
  - Signals 494
- ES4320-XSG Subsystem
  - Globals 477
- ES4330
  - Aufbau des RTIO Baums 531

- ES4330-Hw Device
  - Globals 534
  - Groups 536
  - Signalgruppe „Control“ 537
  - Signalgruppe „Level“ 538
  - Signals 539
- ES4330-Meas Device
  - Globals 542
  - Groups 542
  - Signalgruppe „MeasVal“ 543
  - Signals 545
- ES4330-XMI Subsystem
  - Globals 533
- ES4408-Ctrl Device
  - Data 603
  - Globals 602
- ES4408-Load-Chassis Subsystem
  - Globals 598
- ES4434-Conf-Load Device
  - Globals 604
- ES4435-Current-Sources Device
  - Data 607
  - Globals 605
  - Signals 606
- ES4450-RB-CR-Load Device
  - Data 610, 612
  - Globals 608, 611
  - Signals 609, 612
- ES44XX-Variable-Load 614
  - Data 619
  - Globals 615
  - Groups 616
  - Signals 617
- ES5300.1 Housing 627
- ES5300-BattNode-Ctrl Device 629
  - Data 630
  - Globals 629
  - Groups 629
- ES5300-Chassis System 627
  - Globals 628
- ES5300-Ctrl Subsystem 628
- ES5321 Subsystem
  - Globals 633
- ES5321.1 PWM I/O Board 631
- ES5321-In Subsystem
  - Globals 638
- ES5321-In-HW Device
  - Globals 638
  - Groups 639
  - Signals 639
- ES5321-In-Meas Device 641
  - Globals 641
  - Signals 641
- ES5321-In-SENT Device
  - Globals 648
  - Groups 650
  - Signals 651
- ES5321-Out Subsystem 653
  - Globals 653
- ES5321-Out-Dig-Pwm Device
  - Globals 654
  - Signals 654
- ES5321-Out-Multi-Pulse Device 658
  - Globals 658
  - Signals 660
- ES5321-Out-SENT Device 661
  - Globals 661
  - Signals 664
- ES5321-RPM Device 634
  - Globals 634
  - Groups 635
  - Signals 635
- ES5335 Arbitrary Signal Generator
  - PCIe Board 241
- ES5335 Subsystem
  - Globals 246
- ES5335-Knock Device
  - Data 268
  - Globals 264
  - Groups 266
  - Signals 267
- ES5335-Misfire Device
  - Data 262
  - Globals 259
  - Groups 261
  - Signals 261
- ES5335-MSA-Sensor Device
  - Globals 274
  - Groups 276
- ES5335-OnlineWaveformAccess Device
  - Data 271
- ES5335-OutputMux Device
  - Data 250
  - Globals 249
  - Groups 249
  - Signals 249
- ES5335-Powertrain Subsystem
  - Globals 247



- ES5335-Rpm Device
  - Data 258
  - Globals 256
  - Groups 257
- ES5335-Sig Device
  - Data 252
  - Globals 251
  - Groups 252
- ES5338 Subsystem 670
  - Globals 670
- ES5338.1 Wheelspeed Sensor Simulation 669
- ES5338-DF10-DF11s Device
  - Globals 675
  - Groups 676
  - Signals 677
- ES5338-DF11i Device
  - Globals 679
  - Groups 681
  - Signals 682
- ES5338-PSI5 Device
  - Globals 690
  - Signals 695
- ES5338-VDA Device
  - Globals 684
  - Groups 686
  - Signals 687
- ES5340.1 Electric Drive Simulation Board 697, 793
- ES5340-Analog Device
  - Data 852
  - Globals 851
  - Groups 852
- ES5340-Analog-Arbitrary Device 756
  - Data 759
  - Globals 757
  - Groups 758
  - Signals 758
- ES5340-Analog-Direct-Out Device 742, 809
  - Data 743, 810
  - Globals 742, 809
  - Groups 742, 809
  - Signals 742, 810
- ES5340-Analog-In Device 711, 799
  - Data 712, 800
  - Signals 711, 799
- ES5340-Analog-Out 741, 809
- ES5340-Analog-Out-Mux Device 761, 811
  - Data 763, 812
  - Signals 761, 811
- ES5340-Dig-In-HW Device 713
  - Globals 713
- ES5340-Dig-In-Inverter-Meas Device 720
  - Data 725
  - Globals 721
  - Groups 723
  - Signals 724
- ES5340-Dig-In-Meas Device 714
  - Data 720
  - Groups 718
  - Signals 718
- ES5340-Digital-Arbitrary Device 733
  - Data 735
  - Globals 733
  - Groups 735
  - Signals 735
- ES5340-Digital-Direct-Out Device 726, 802
  - Data 728, 803
  - Globals 727, 802
  - Groups 727, 802
  - Signals 727, 802
- ES5340-Digital-In 713
- ES5340-Digital-Out 726, 801
- ES5340-Digital-Out-Mux Device
  - Data 740, 808
  - Signals 738, 806
- ES5340-Digital-Position-Sensor Device 729
  - Data 732
  - Globals 730
  - Groups 731
  - Signals 731
- ES5340-HW Device 817
  - Groups 817
  - Signals 819
- ES5340-IM-1.0.0 Device 779
  - Data 789
  - Globals 780
  - Groups 784
  - Signals 784
- ES5340-Knock Device 861
  - Data 865
  - Globals 861
  - Groups 863
  - Signals 864
- ES5340-Master Subsystem 699, 794
  - Globals 699, 794

- ES5340-MeasAngle Device 828
  - Globals 836
  - Groups 836
  - Signals 838
- ES5340-MeasTime Device 821
  - Globals 824
  - Groups 824
  - Signals 825
- ES5340-Measure Subsystem 814
  - Globals 816
- ES5340-Misfire Device 856
  - Data 859
  - Globals 856
  - Groups 858
  - Signals 858
- ES5340-MSA-Sensor Device 868
  - Data 871
  - Globals 869
  - Groups 870
- ES5340-PMSM-1.0.0 Device 765
  - Data 775
  - Globals 766
  - Groups 770
  - Signals 770
- ES5340-PWM-Output Device 736, 804
  - Data 737, 805
  - Globals 736, 804
  - Groups 736, 804
  - Signals 737, 804
- ES5340-RailPump Device 841
  - Globals 841
  - Signals 843
- ES5340-Resolver Device 752
  - Data 755
  - Globals 753
  - Groups 754
  - Signals 754
- ES5340-RPM Device 704, 796
  - Data 707, 798
  - Globals 705, 796
  - Groups 707, 797
- ES5340-SigGen Device 851
- ES5340-SigGen Subsystem 849
- ES5340-SigGen Subsystem
  - Globals 849
- ES5340-Sine-Encoder Device 747
  - Data 750
  - Globals 748
  - Groups 749
  - Signals 749
- ES5340-Sine-Extrapolated Device 744
  - Data 746
  - Globals 744
  - Groups 745
  - Signals 745
- ES5340-Slave Subsystem 764, 813
- ES5350 Subsystem 873
  - Globals 873
- ES5350.1 Analog Board 873
- ES5350-Analog-In Device
  - Globals 874
  - Groups 874
  - Signals 875
- ES5350-Analog-Out Device
  - Globals 876
  - Groups 877
  - Signals 877
- ES5385 Device
  - Globals 881
- ES5385.1 Carrier Board for Resistor Cascade 881
- ES5385-ResistorCascade Device
  - Globals 881
  - Groups 882
  - Signals 884
- ES5392 Subsystem 887
  - Globals 887
- ES5392.1 High Current Switch Board 887
- ES5392-Meas-HighCurrent Device
  - Globals 897
  - Groups 897
  - Signals 897
- ES5392-MRC-Simulation Device
  - Globals 899
  - Groups 899
  - Signals 900
- ES5392-Out-Dig-Pwm Device
  - Globals 901
  - Groups 901
  - Signals 902
- ES5392-Out-Multi-Pulse Device
  - Globals 903
  - Groups 904
  - Signals 904
- ES5392-PwrCtrl Device
  - Globals 888
  - Groups 889
  - Signals 890
- ES5392-SwCtrl-DigOut Device
  - Signals 894

ES5392-SwCtrl-External Device  
 Globals 891  
 Groups 892  
 Signals 892  
 ES5392-SwCtrl-HighCurrent Device  
 Signals 895  
 ETAS Kontaktinformation 1007

**F**

FLEX-CAN Subsystem 58  
 Frequenz 969  
 Frequenz- und Zykluszeitmessungen  
 561

**H**

High- und Low-Zeit 967  
 externe Signalvalidierung 971

**I**

Injektorsimulation (CVO) 622  
 Injektorsimulation (VCC/VCA) 623  
 Item-Templates 53  
 IXXAT iPCI-I XC16/PCI CAN Interface-  
 karte 905  
 IXXAT-XC16-CAN Subsystem  
 Globals 906  
 IXXAT-XC16-CAN\_IO Device  
 Data 918  
 Globals 911  
 Groups 913  
 Signals 915  
 IXXAT-XC16-CAN-CTRL Subsystem  
 Globals 908

**L**

LOAD20A14C 996  
 Load20A14C Subsystem  
 Globals 997  
 LoadSwitch Device  
 Globals 1001  
 Groups 1001  
 Signals 1002

**M**

MasterSglGnrtr Device  
 Globals 143  
 Groups 145  
 MESSAGE Device 62  
 Globals 62  
 Groups 63  
 Signals 64

**Messverfahren**

additive Pulsweitenmessungen  
 89, 173, 451, 554, 646  
 Frequenz- und Zykluszeitmessun-  
 gen 91, 176, 455, 561, 647  
 Pegelmessung 184, 574  
 Pegelmessungen 92, 458, 648  
 Positionsverfolgung an zwei-pha-  
 sigen Schrittmotoren 575  
 Puls- und Flankenählung 89, 452  
 Pulsweitenmessungen 88, 171,  
 449, 551, 645  
 Pulsweitenmessungen mit Enable-  
 oder Validierungssignal  
 557  
 Pulsählung 182, 572  
 Relativmessungen zwischen Hard-  
 warekanälen 564  
 Tastverhältnismessungen 92, 178,  
 457, 647  
 Vermessung von Flanken 569, 571  
 Vermessung von Flanken: Winkel-  
 stempel 180

Messwertbestimmung 962

Module Status 1005

**Motor**

stehend 973

**MTS-UBATT Device**

Globals 922

Groups 925

Signals 926

MTS-UBATT/UBATT-POW 921

**P**

PB1650ADC1 378

PB1650ADC1 Device

Globals 378

Signals 379

PB1650DAC1 381

PB1650DAC1 Device

Globals 381

Signals 382

PB1650DIO1 384

PB1650DIO1 Device

Globals 384

Signals 385

PB1650DIO2 386

PB1650DIO2 Device

Globals 386

Signals 387

PB1650REL1 388

- PB1650REL1 Device
  - Globals 388
  - Signals 389
- PB1651ADC1 421
- PB1651ADC1 Device
  - Data 431
  - Globals 422
  - Groups 426
  - Signals 428
- PB1651PWM1 432
- PB1651PWM1 Subsystem
  - Globals 433
- PB1651PWM1-In-Meas Device
  - Globals 441
  - Groups 441
  - Signals 444
- PB1651PWM1-Out Device
  - Globals 459
  - Groups 459
  - Signals 464
- Pegelerfassung 965
- Pegelmessung 574
- Pulsbreite
  - kleiner als Auflösung 963
- Pulsweitenmessungen 551
- Pulsweitenmessungen mit Enable-  
oder Validierungssignal 557
- Pulszahl 967
- Pulszählung 572
- PWM-COUNTER Device
  - Globals 120
  - Groups 122
  - Signals 122
- R**
- Raddrehzahlfühler
  - Simulation von 126
- Relativmessungen zwischen Hard-  
warekanälen 564
- RTIO-Editor 23
  - Register „Data“ 49
  - Register „Globals“ 45
  - Register „Groups“ 47
  - Register „Signals“ 48
  - starten 23
- S**
- Schaltsignal
  - Generierung 962
- SigGen Subsystem
  - Globals 141
- SlaveSglGnrtr Device
  - siehe MasterSglGnrtr Device
- Spikes 963
- Steuergeräte-Signal
  - Erfassung 962
- T**
- T8IO 995
- T8IO Subsystem
  - Globals 995
- T8Module Subsystem 994
  - Globals 994
- Tastverhältnis 970
- TDAC 985
- TDAC Device
  - Globals 985
  - Groups 987
  - Signals 988
- Timeout-Erkennung 973
- TMIO 962
  - Messwerterfassung 964
  - Signalauswertung 965
- TMIO Device
  - Globals 974
  - Groups 975
  - Signals 977
- Transputer-basierte Hardware
  - RTIO-Package 919
- TRS422 993
- TRS422 Subsystem
  - Globals 993
- TS Subsystem
  - Globals 930
- TsKnockGen Device 945
  - Globals 946, 949, 950
- TsMisfire Device 955
  - Globals 956
  - Groups 959
  - Signals 960
- TS-PWM 929
- TsPWM Device
  - Globals 931
  - Groups 931
  - Signals 932
- TsRPM Device 933
  - Globals 935
  - Groups 936
  - Signals 937
- TsSIG Subsystem
  - Globals 939

TsSIGGen Device  
  Globals 939  
  Groups 941  
  Signals 942

**V**

Vermessung von Flanken 569, 571

**W**

WheelSnrSim Device 126  
  Globals 130  
  Groups 131  
  Signals 132  
Winkelsegmentierung 973

**Z**

Zykluszeit 969

