LABCAR-RTC V5.4.2
Real-Time Execution Connector
User's Guide

Copyright

V5.4.2 R01 EN - 12.2016

Contents

1 The RTIO Editor

LABCAR-RTC (Real-Time Execution Connector) is an add-on software for LAB-CAR-OPERATOR which is required for the integration and configuration of the hardware components for HiL real-time experiments.

The RTIO Editor is the user interface of LABCAR-RTC in which the hardware configuration for the relevant LABCAR-OPERATOR project is created.

This chapter contains the following information:

1.1 The Control Elements of the RTIO Editor

This section presents the control elements of the RTIO Editor.

To launch the RTIO Editor

- Open a project in LABCAR-OPERATOR.
- Select **Project** → **RTIO Editor**.

  The RTIO Editor opens.

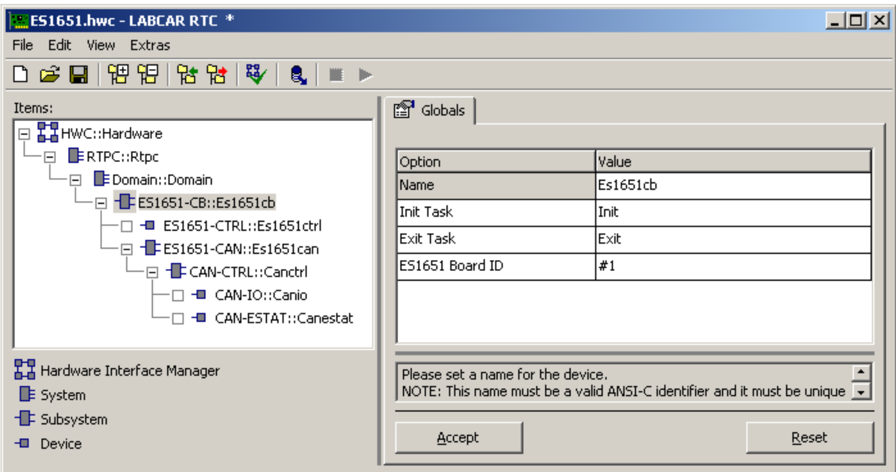


Fig. 1-1 The User Interface of the RTIO with an Open Hardware Configuration

The user interface consists of the menu bar, a button bar, the "Items" list and a set of tabs via which most modifications and entries can be made.

The "Items" list lists the various elements of the hardware configuration with their icons; under the tabs is a text box which contains brief information on the selected cell.

Use the buttons **Accept** and **Reset** to accept or reject changes made.

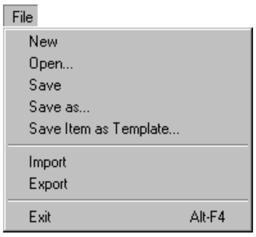
1.1.1 The Menu of the RTIO Editor

This section contains a description of the menu items.

Note

Depending on the operating mode of the RTIO Editor, the menu items are either enabled or disabled. The letters "E" (Edit mode) and "R" (Runtime mode) provide information on when a menu item is available.

The "File" Menu



- **File → New** (E)

  Creates a new hardware configuration.

- **File → Open...** (E, R)

  This makes it possible to load an existing hardware configuration into the RTIO Editor via a file selector window.

- **File → Save** (E, R)

  Saves the current hardware configuration.

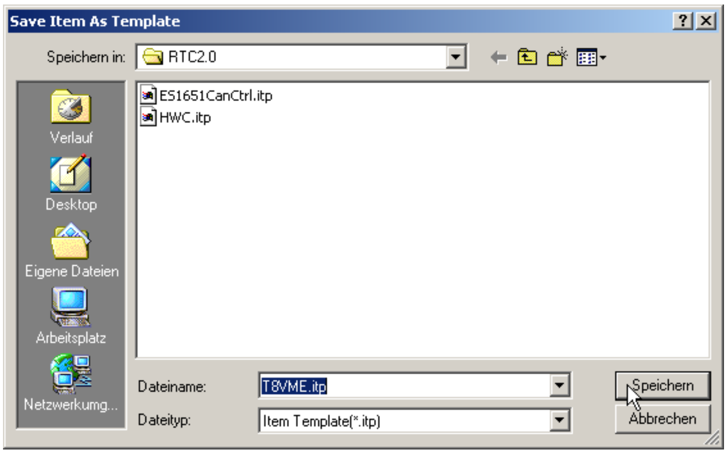
- **File → Save As...** (E, R)

  Saves the current hardware configuration under another name.

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

  Saves the settings of the selected item as a template (*.itp) for all items of this type (see " "Templates" Tab" on page 50).

  In the dialog box "Save Item As Template", the item can be saved as a template in the specified directory.



Note

When opening the dialog box, the file path preset in "Options" under "Template Path" is always used (see ""Templates" Tab" on page 50). The suggested file name must not be changed.

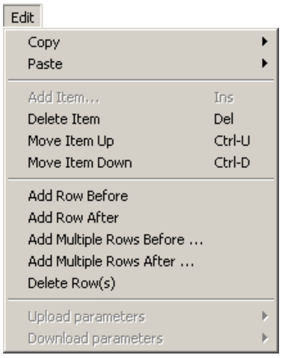
- **File → Export** (E)

  Exports the hardware configuration currently open specifying a new file name as *.hwc file or as *.csv file

- **File → Exit** (E, R)

  Closes the RTIO Editor

The "Edit" Menu



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

    Creates a copy of the selected item, including any existing subtrees, and transfers these to an internal buffer of the RTIO Editor.

    The buffer is available as long as the RTIO Editor is open.
  - **Item(s) for Export** (E)

    Works like **Edit → Copy → Item(s)**, but here the copy is written to an external file. This function is useful for exchanging subconfigurations.
  - **Item Data** (E)

    Only copies the data of an item into an internal buffer (this function does not refer to entire subtrees). This function can be used to exchange the settings between two items.
- **Edit → Paste →**
  - **To Selected Item** (E)

    Adds an item (possibly with subtree) copied with **Edit → Copy → Item(s)** under the item just selected. The following prerequisites are necessary for this to work.

    1. The Item types have to correspond to each other.
    2. It has to be possible to insert the item or the subtree under the selected item (the same is true here as for **Edit → Add Item...**).
  - **To Selected Item From Import** (E)

    Works exactly the same as **Edit → Paste → To Selected Item**, but here, an item or subtree saved previously in a file is read in. The restrictions applicable to the last menu item apply here too.
  - **Data To Selected Item** (E)

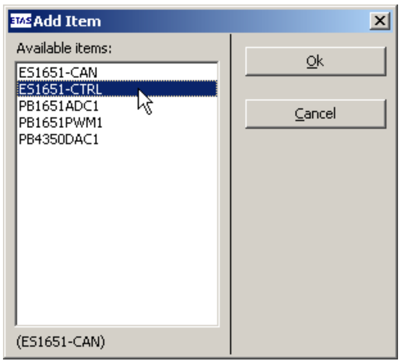
    Overwrites the data of the selected item with the data which was written to the buffer via **Edit → Copy → Item Data**.

    Unlike the previous **Edit → Paste →** menu items, it is possible here to exchange data between similar or compatible items. This function is only possible if the type of selected item is the same as or compatible to the item from which the data was copied. In addition, the structure also has to be virtually identical; i.e. the number of rows in the tables in the "Groups", "Signals" and "Data" tabs has to correspond to each other.

- **Edit → Add Item...**

  This menu item makes it possible to add a new hardware item to the hardware configuration.

  A selector window opens showing all available items of the next hierarchy down.



  The selected element is then inserted into the hardware configuration.

Note

Depending on the configuration already created, pasting may be interrupted with an error message (even after an item has been selected successfully). This is the case, for example, when the number of resources available (ports, slots) has already been used up or when a specific order of insertion is necessary.

- **Edit → Delete Item** (E)

  This menu item is used to delete the marked item from the hardware configuration.

  It is not possible to delete the top item of the hierarchy (Hardware Interface Manager).

Note

Sometimes an attempt to delete an item from the hardware configuration is interrupted with an error message. This is, for example, the case when a specific deletion order has to be adhered to.

- **Edit → Move Item Up** (E)

  This changes the item order within a hierarchy level in the hardware configuration. The menu item moves the selected item one element up.

- **Edit → Move Item Down** (E)

  This changes the item order within a hierarchy level in the hardware configuration. The menu item moves the selected item one element down.

- **Edit → Add Row Before** (E)

  This menu item is only available for items of the type "Device", for which it is possible to dynamically change signal groups or the number of signals.

  The menu item is available and can be selected under the following prerequisites:

  1. An item of the type "Device" which has "dynamic" properties (e.g. CAN-IO Device) has been selected.
  2. The "Groups" or "Signals" tab has been selected.
  3. An item in the column "No." has been selected (first column in the table).

  This results in a row being inserted in front of the selected row or the selected area.

Note

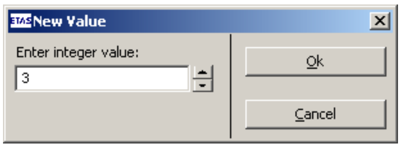
The size of the table is often restricted due to the system. In this case, executing the function has no effect. As this changes the table structurally, the change cannot be undone by the **Reset** *button.*

- **Edit → Add Row After** (E)

  Works exactly the same as **Add Row Before** except a new row is added after the selected cell or the selected area.

- **Edit → Add Multiple Rows Before...** (E)

  Works exactly the same as **Add Row Before** except that several rows can be added here at a time. The number of rows to be added can be entered in a dialog box:

- **Edit → Add Multiple Rows After...** (E)

  Works exactly the same as **Add Multiple Rows Before** except new rows are added after the selected cell or the selected area.

- **Edit → Delete Row(s)** (E)

  Deletes the row or area marked in the "No." column.

Note

For system reasons, at least one signal group or one signal has to remain in the table. If you attempt to delete the last row, an error message is displayed.

- **Edit → Upload Parameters →**
  - **for selected item** (R)

    This results in all values of the selected item which can be modified in Runtime mode being read by the target and the hardware configuration loaded in the RTIO Editor being overwritten with these values.

  – **for selected item (recursive)** (R)

    The function is identical to the one described before with the difference that here the parameters of the RTIO tree structure under the selected item are also updated.

  – **all items** (R)

    The function is identical to the one described before with the difference that here the parameters of the entire RTIO tree structure are updated.

- **Edit → Download Parameters →**

  – **for selected item** (R)

    This means that all values of the selected item which can be changed in Runtime mode are downloaded to the target. If the function is executed, all corresponding settings in the C code module are overwritten with the current values of the RTIO Editor.

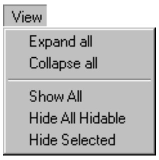
  – **for selected item (recursive)** (R)

    The function is identical to the one described before with the difference that here the parameters of the RTIO tree structure under the selected item are updated.

  – **all items** (R)

    The function is identical to the one described before with the difference that here the parameters of the entire RTIO tree structure are updated.

The "View" Menu



- **View → Expand All** (E, R)

  Expands the hierarchy tree of the RTIO item completely (see "Items" on page 32).

- **View → Collapse All** ((E, R)

  Hides the hierarchy tree of the RTIO item completely (see "Items" on page 32).

- **View → Show All** (E, R)

  This menu shows all displayable options/columns of a tab.

  The table of a tab can be structured so that some of the possible options or columns are not displayed by default. This has the advantage that options which are not used often are hidden and the table can be structured more clearly.

Note

If the table is then confirmed with the ***Accept*** *button and the hardware configuration is saved, the table is once again available in the form in which it was saved, i.e. newly displayed options/columns remain displayed.*

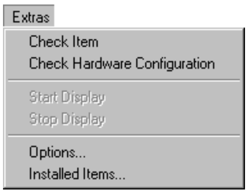
- **View → Hide All Hidable** (E, R0)

  This is the opposite of the previous menu item, i.e. all options/columns which can be hidden are hidden.

- **View → Hide Selected** (E, R)

  Hides the option or column currently selected, providing they can be hidden.

The "Extras" Menu



- **Extras → Check Item** (E)

  This function makes an explicit check of all settings of the selected item possible. This check always takes place implicitly before RTIO code is generated.

  RTIO code generation is not possible until this check has been completed without any errors. Warnings are admissible but should be heeded in each individual case.

- **Extras → Check Hardware Configuration** (E)

  Executes a check of the entire hardware configuration.

- **Extras → Start Display** (R)

  Starts the display of the "Data" column in the "Data" tab of the items of the type "Device".

- **Extras → Stop Display** (R)

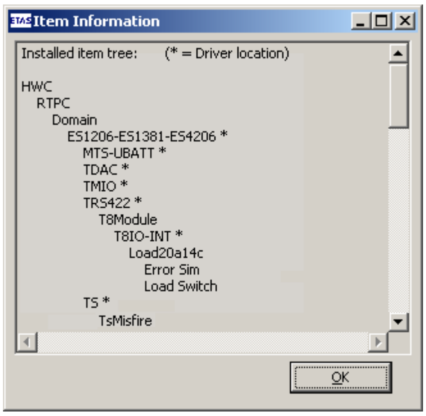
  Stops the display of the "Data" column in the "Data" tab of the items of the type "Device".

- **Extras → Options...** (E, R)

  Opens a dialog box with several tabs in which different options can be set (see "Options" on page 49).

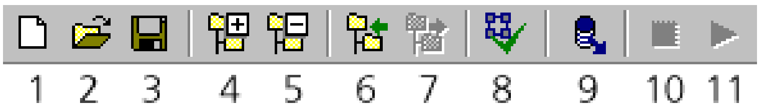
- **Extras → Installed Items...** (E, R)

  Opens the "Item Information" window in which all available hardware items are listed in their hierarchy.



1.1.2 Icon Bar

This section contains a description of the functions you can activate using the buttons of the icon bar.



1. **New**

   Creates a new hardware configuration.

   Exactly the same as **File → New**.

2. **Open**

   This makes it possible to load an existing hardware configuration into the RTIO Editor via a file selector window.

   Exactly the same as **File → Open...**.

3. **Save**

   Saves the current hardware configuration.

   Exactly the same as **File → Save**.

4. **Expand all**

   Expands the hierarchy tree of the RTIO item completely (see "Items" on page 32).

   Exactly the same as **View → Expand all**.

5. **Collapse all**

   Hides the hierarchy tree of the RTIO item completely (see "Items" on page 32).

   Exactly the same as **View → Collapse all**.

6. **Add Item...**

   This makes it possible to add a new hardware item to the hardware configuration.

   Exactly the same as **Edit** → **Add Item...**.

7. **Delete Item**

   This deletes the selected item from the hardware configuration.

   Exactly the same as **Edit** → **Delete Item**.

8. **Check Hardware Configuration**

   This menu item makes an explicit check of all settings possible in terms of the consistency within the hardware configuration.

   Exactly the same as **Extras** → **Check Hardware Configuration**.

9. **Export**

   Exports the hardware configuration currently open specifying a new file name as *.hwc file or as *.csv file

   Exactly the same as **File** → **Export**.

10. **Stop Display**

    Stops the display of the "Data" column in the "Data" tabs of the "Device" items.

    Exactly the same as **Extras** → **Stop Display**.

11. **Start Display**

    Starts the display of the "Data" column in the "Data" tabs of the "Device" items.

    Exactly the same as **Extras** → **Start Display**.

1.2 Items

The tree-like list "Items" is used to display the structure of the current hardware configuration.

The term "Item" is a general term for an element of a hardware configuration. This might be an entire board or simply a unit which in terms of its function is in itself complete.

Note

Hardware is completely specified when each end node of the hardware tree is of the type "Device".

The tabs to the right of the "Items" list always display the relevant settings for the selected item.

1.2.1 The Shortcut Menu of the "Items" List

Right-click the "Items" list to open the shortcut menu.

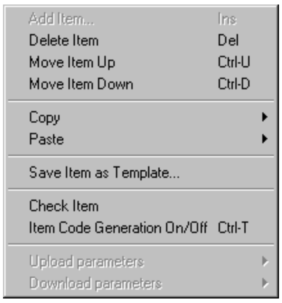


Fig. 1-2 The Shortcut Menu of the "Items" List

This shortcut menu makes various actions possible; these are described below.

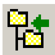
To add an item

- To add an item to the hardware configuration, select **Add Item...**.

or

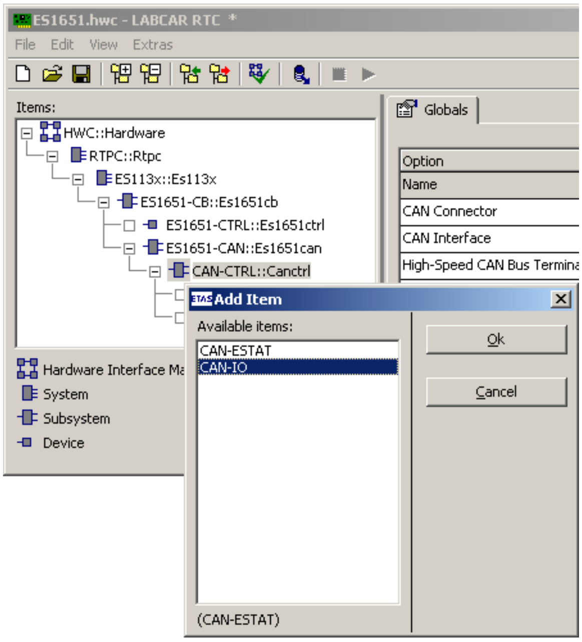
- Press <INS>.

or



- Click the **Add Item...** button.

  The "Add Item" window opens.



- Select the item to be inserted from the list of available items of the hierarchy next down.
- Click **OK**.

  The selected item is inserted.

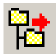
To delete an item

- To delete an item from the hardware configuration, activate it and select **Delete Item**.

or

- Press <DEL>.

or



- Click the **Delete Item** button.

  The selected item is deleted.

> **Note**
>
> *If the item to be deleted has subitems, these are also deleted!*

To move an item up

- If you want to move an item up (within the hierarchy level), select **Move Item Up**.

or

- Press <CTRL-U>.

  The selected item is moved up.

To move an item down

- If you want to move an item down (within the hierarchy level), select **Move Item Down**.

or

- Press <CTRL-D>.

  The selected item is moved down.

To copy an item

- To copy an item (including any subitems), select **Copy** → **Item(s)**.

or

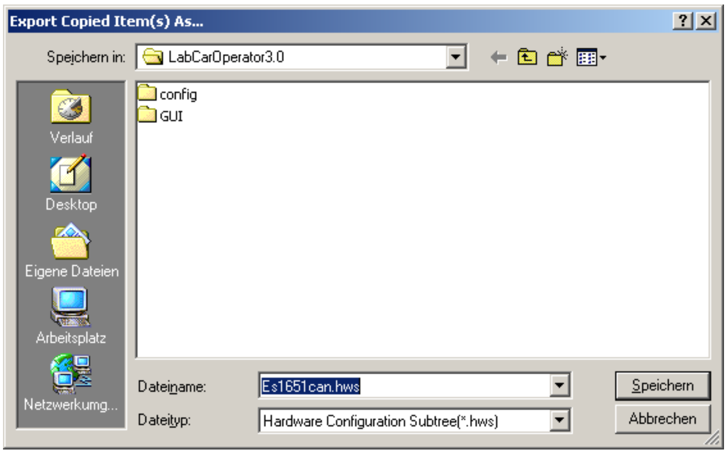
- Press <CTRL-C>.

  The selected item or subtree is copied into the buffer.

To copy an item for export

- To copy an item (including any subitems) to a file, select **Copy** → **Item(s) for Export**.

  The file selector window "Export Copied Item(s) As..." opens.



- Select a file name and click **Save**.

  The selected subtree of the hardware configuration is saved in a file with the extension "hws" (Hardware Configuration Subtree).

To copy data (settings) of an item

- To copy the data of a single item to the buffer, select **Copy** → **Item Data**.

  The settings of the selected item are copied to the buffer.

Note

Only the data of the individual item, and not the subitems, is copied.

To insert an item

- To insert an item (including any subitems) from the buffer into your hardware configuration, activate the item under which it should be inserted.
- Select **Paste** → **to selected Item**.

or

- Press <CTRL-V>.

  The item in the buffer (including any subitems) is added at the desired point.

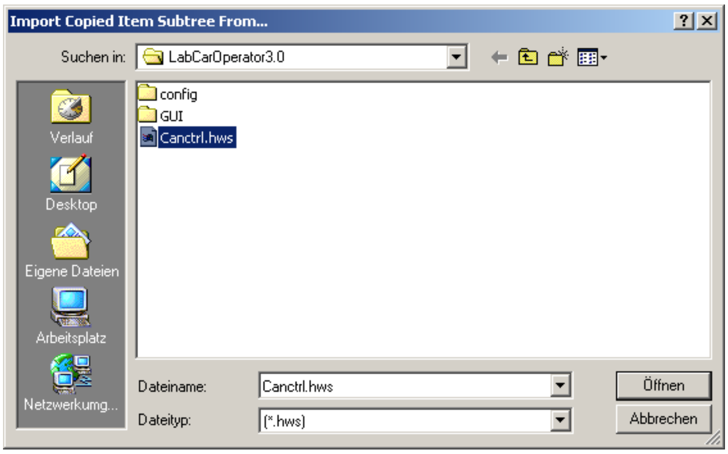
Note

Please observe the conditions listed in the section "The "Edit" Menu" on page 26.

To insert an item from a file exported previously

- To insert an item from a file exported previously (including any subitems, see "To copy an item for export" on page 35), select **Paste** → **to selected Item from Import**.

  The file selector window "Import Copied Items Sub-tree From..." opens.



- Select a file (*.hws) and click **Open**.

Note

Please observe the conditions listed in the sections "The "Edit" Menu" on page 26.

To add data (settings) of an item

- To add data of an item copied to the buffer using **Copy** → **Item Data**, select **Paste** → **Data to selected Item**.

  The data is copied.

Note

Please observe the conditions listed in the sections "The "Edit" Menu" on page 26.

To save an item as a template

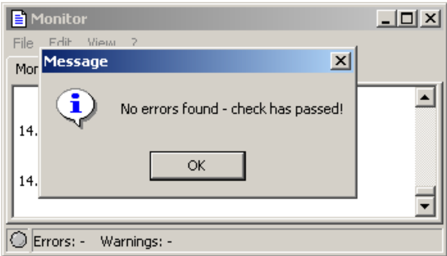
- To save a specific item as a template, select **Save Item as Template...** .

  For more details, refer to "The "File" Menu" on page 24.

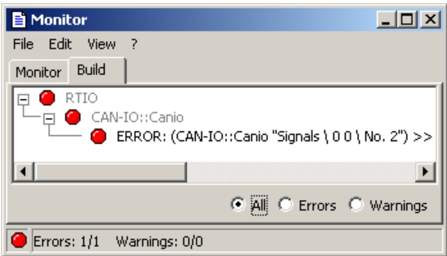
To check the configuration of an individual item

- To check the configuration of an individual item, select **Check Item**.

  The configuration is checked. If it is ok, an appropriate message is displayed.



  If the configuration contains errors, messages are displayed in the "Messages" window.



Note

Only the configuration of the selected item is checked – to check the entire hardware configuration choose **Check Hardware Configuration** *(see "The "Extras" Menu" on page 30).*

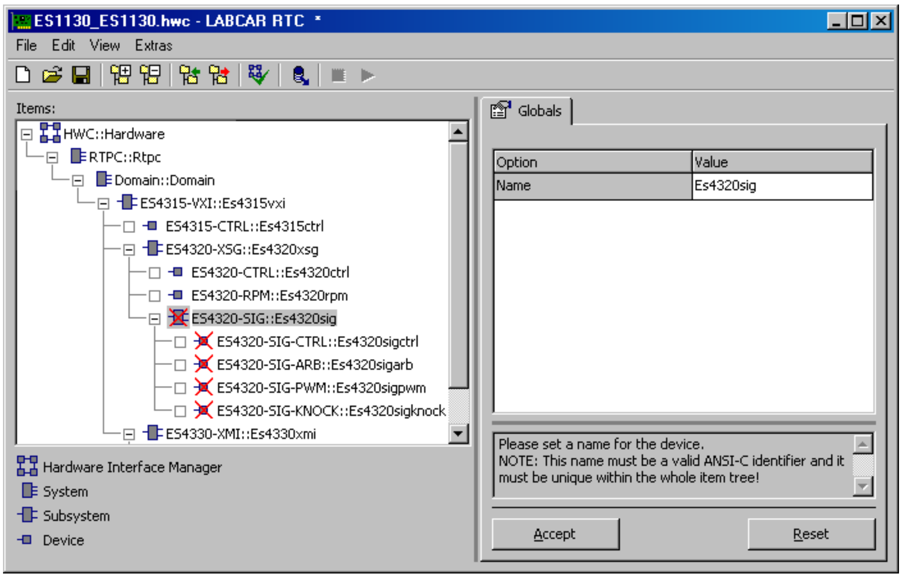
To deactivate code generation for an item

- To deactivate code generation for an item (including any subitems), select it.
- In the shortcut menu, select **Item Code Generation On/Off ...**.

or

- Press <CTRL-T>.

  The code generation for this item (including any subitems) is deactivated.



Items not involved in code generation are indicated by a red cross.

Deactivation can be undone in exactly the same way.

To upload parameters from the target

- To load the current values of the hardware configuration of a selected item from the target into the RTIO Editor, select **Upload Parameters** → **for selected item**.
- To load the current values of the hardware configuration of a selected item including the items below it from the target into the RTIO Editor, select **Upload Parameters** → **for selected item (recursive)**.
- To load the current values of the hardware configuration of all available items from the target into the RTIO Editor, select **Upload Parameters** → **all items**.

To download parameters to the target

- To download changed settings of a selected item of the hardware configuration in the RTIO Editor to the target, select **Download parameters** → **for selected item**.

- To download changed settings of a selected item of the hardware configuration including the items below it in the RTIO Editor to the target, select **Download parameters** → **for selected item (recursive)**.
- To download changed settings of all available items of the hardware configuration in the RTIO Editor to the target, select **Download parameters** → **all items**.

Note

Uploading and downloading from and to the target is of course only possible if an experiment is open.

1.3 Configuration Tabs

Depending on the item selected in the "Items" list, different tabs are displayed. These are used to make individual settings for the items.

If the selected item is of the type "Device", four tabs are displayed; for all other element types, only the "Globals" tab is displayed.

The tasks are divided as follows over the tabs:

- **Globals**

  This includes all global settings which apply to the entire item (e.g. VME-bus address of the I/O board).

- **Groups**

  This contains all settings which apply to a signal group (e.g. amplification factor for a multi-channel AD-converter or the identifier of a CAN message).

  Note

  A signal group is distinguished by the fact that all signals it contains have the same transfer direction (send or receive) and are recorded at the same time or at the same interval.

- **Signals**

  All signal-specific settings are made in this tab (e.g. conversion formulae for a signal).

- **Data**

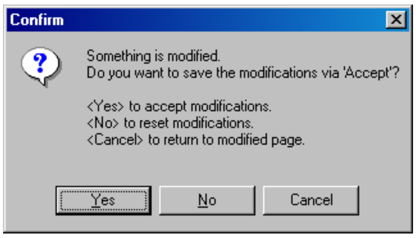
  This tabs shows the data of the individual signals/the data of the individual signals can be edited in this tab (depending on whether the "Direction" of the signal group has the value "receive" or "send").

In addition, each tab has two buttons **Accept** and **Reset** which have the following tasks:

Accept: (Edit mode and Runtime mode)

As soon as a tab is displayed for the first time or redisplayed, a copy is made of the dataset it contains which is then shown in the data table. Every change to the data only modifies the copy of the original data.

Once editing has been completed, the user has to explicitly write the modified data back to the original data by pressing the **Accept** button. A prompt to this effect is displayed if the user wants to quit the tab before saving:



Save the modifications with **Yes**, reject them with **No** or cancel the change with **Cancel**.

This mechanism is necessary to guarantee a consistent dataset in all circumstances as there are a great number of dependencies to be taken into consideration within the "Items" structure.

Accept: (only Runtime mode)

With some items, there are settings which can also be changed during runtime. This is possible whenever the relevant hardware component allows it.

These settings can be recognized in Runtime mode by the fact that they are not "disabled", i.e. can be edited. If table cells equipped in this way are changed, clicking **Accept** not only results in a comparison of the data, but also an update of all referenced parameters.

Note

To inform the driver about a changed dataset and the necessary reconfiguration, a running "Config" task is mandatory. Without this, the driver cannot process the data.

Reset:

This button undoes all changes made after you last clicked **Accept**.

1.3.1 General Notes

Editing Option / Cell

In a tab's table, the relevant cell can be edited by clicking it with the mouse. Navigation within the table can also take place using the cursor keys whereby pressing the space bar opens the editor on the cell currently selected.

Some columns are also multi-selectable, i.e. several cells within the column can be selected simultaneously (e.g. "Formula" column). A multiple selection can be executed by pressing the left mouse button and dragging the mouse across the column at the same time or by a second left mouse click and pressing the <SHIFT> button. Navigating using the keyboard keeping the <SHIFT> key pressed down at the same time is also possible.

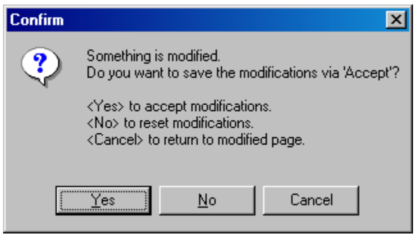
Editability

Not all options can be modified by the user. Some options are permanently disabled and are only used to display certain status values. Other options can be disabled some of the time and ready for input at others which may depend on other option settings in this or in another item in the items hierarchy.

Tip: Open the hardware configuration in Runtime mode; display all columns or options and see which values can still be changed.

Modified State

The RTIO Editor registers whether a value of the table has already been changed or whether it is currently being modified. If another item or tab is selected, without the change having been made valid with the **Accept** button, the following will be displayed:



Save the modifications with **Yes**, reject them with **No** or cancel the change to another tab with **Cancel**.

Scrolling the Table

The table may be too large to display in one tab. In this case, scrollbars are displayed.

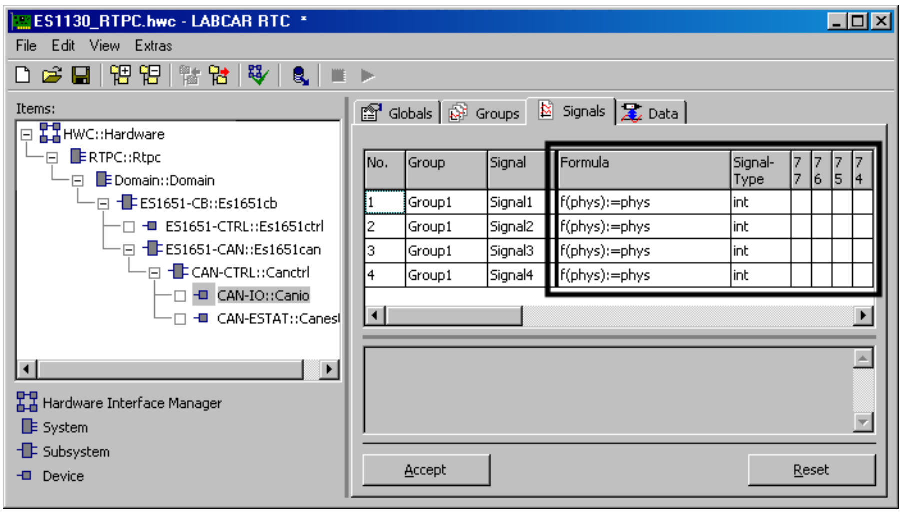


Fig. 1-3 Scrollable Area of a Configuration Tab

These do not, however, move the entire table, but only the variable part (marked area in Fig. 1-3); the rest remains unchanged.

Note

If there is a large number of columns displayed in the left-hand (static) part of the table, the right-hand scrollable part may no longer fit in the tab mistakenly leading the user to believe that scrolling is not actually having any effect. This problem can be solved either by enlarging the RTIO Editor or by hiding static columns which are not required.

Changing Column Width

The column width can be changed by left-clicking the edge of the column and then dragging the edge as required. The newly specified column widths are then saved with the hardware configuration if they are first confirmed with the **Accept** button.

They are then set as required when reloaded.

Help Text

There is a text box in the lower part of the tab which makes it possible to display a short help text for the option currently selected.

1.3.2 Default Options in the "Globals" Tab

This section describes the default options of the "Globals" tab. For details on further item-specific options, refer to the relevant sections for the particular boards.

Name

This is where you can enter an individual name for the item. The name has to be a valid ANSI C compatible name and be unique throughout the entire hardware tree.

This option is available in every "Globals" tab.

Init Task

This is where the task has to be selected in which the hardware driver is to be initialized. This is usually a "real" Init task. In some cases, it may be a "Software" task, but this is very unusual and results in a warning during checking.

This option is always available when the item is connected to a low-level driver.

Note

It must always be ensured that, when processing the tasks, this task is executed before any other task connected to this driver (Start, Config...) can be executed. Please remember that the "Init" call can only take place if an "Exit" call took place before it. If not, there will be runtime errors in the experiment resulting in driver lock!

A consistent task order must always be adhered to:

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

Start Task

It this setting is available, a separate task can be specified here to start the driver. If this is not necessary, the same task as specified for "Init task" can usually be specified.

The "Start task" is supported by only a few hardware drivers.

Note

Make sure you observe the task order (see "Init Task" on page 43)!

Stop Task

If this setting is available, a separate task can be specified here to stop the driver. If this is not necessary, the same task as specified for "Exit Task" can usually be specified.

The "Stop Task" is supported by only a few hardware drivers.

Note

Make sure you observe the task order (see "Init Task" on page 43)!

Exit Task

This is where the task has to be selected in which the hardware driver is to be deinitialized. This is usually a "real" Exit task (= inactive Init task), but could also be a "Software" task although this is very unusual and results in a warning during checking.

An "Exit Task" is always available when the item is connected to a low-level driver.

Note

Make sure you observe the task order (see "Init Task" on page 43)!

Config Task

A "Config task" is always available if an item in the hardware driver can be modified in Runtime mode.

If a change is to be made, an Alarm task has to be selected here. The cycle time of the task is not critical and only shows how fast the driver reacts to the changed values after the **Accept** button has been pressed. A low cycle time (< 50 ms), however, also means an unnecessarily high "background load". Normally tasks with a cycle time between 200 ms and 1 s should be used.

If no change is to be made, select a Software task which is not invoked.

Analyze Task

The task selected here always has to be a Software task.

An "Analyze task" can be needed by drivers which work in Interrupt mode.

Note

*This task must be **exclusively** available to the item or the relevant hardware driver and must not be used by any other item or user process. Even two identical items must **not** share the same task!*

Acknowledge Task

This same is true here as of the "Analyze task".

Device Manager Task

This task is needed by some drivers to ensure a certain basic supply of the driver independent of the actual data exchange (e.g. error recovery, bus monitoring).

Normally a Timer task has to be specified. The cycle time varies immensely and is specified more precisely in the help text of the tab or in the relevant item documentation.

Version

Shows the version definition of the item which can be used to check the version with the low-level hardware drivers.

"Version" is hidden by default.

Format

Shows the format of the item which can be used to check the compatibility with the low-level hardware drivers.

"Format" is hidden by default.

1.3.3 Default Options in the "Groups" Tab

This section describes the default options of the "Groups" tab. For details on further item-specific options, refer to the relevant sections for the particular boards.

No.

Is responsible for the numbering of the rows so that, for example, error messages can be assigned.

Device

This is where the relevant item name is displayed.

"Device" is hidden by default.

Group

This is where the signal group is specified. This name can usually be edited although the name has to be a valid ANSI C name in this case.

Note

All signal groups within a device have to have different names!

Direction

Specifies the transfer direction of the relevant signal group. If the setting is disabled, the transfer direction is fixed by the hardware and cannot be changed (e.g. A/D Board always "receive", D/A Board always "send"). With some devices (e.g. CAN-IO) you can choose the transfer direction.

Task

This is where the task is specified in which data transfer is to take place. It is also possible to specify several tasks – data transfer then takes place in each of the tasks. Here, Timer tasks or Software tasks are usually admissible. With some devices, it is not possible to make any specification; data transfer then takes place in a different way, usually interrupt-controlled.

1.3.4 Default Options in the "Signals" Tab

This section describes the default options of the "Signals" tab.For details on further item-specific options, refer to the relevant sections for the particular boards.

No.

Is responsible for the numbering of the rows so that, for example, error messages can be assigned.

Device

This is where the relevant item name is displayed.

"Device" is hidden by default.

Group

This is where the relevant signal group name is displayed. With some devices, e.g. "CAN-IO", the assignment of the signal to a signal group can also be chosen freely.

"Group" is hidden by default.

Direction

The transfer direction of the relevant signal group (and thus the signal) is displayed here.

"Direction" is hidden by default.

Task

This is where the assigned task is displayed in which the relevant signal group is transferred.

"Task" is hidden by default.

Signal

A signal name can be specified here. With some devices, the signals are already specified and the specification cannot be modified by the user. If an input is possible, the name has to correspond to ANSI C guidelines.

Note

All names of signals which are assigned to a signal group have to differ!

Formula

If necessary, a signal value can be assigned a linear conversion in this column. A 1:1 conversion is selected by default (Formula `f(phys):= phys`). With some devices, this formula is fixed and the column disabled.

1.3.5 Default Options in the "Data" Tab

This section describes the default options of the "Data" tab. For details on further item-specific options, refer to the relevant sections for the particular boards.

No.

Is responsible for the numbering of the rows so that, for example, error messages can be assigned.

Device

This is where the relevant item name is displayed.

"Device" is hidden by default.

Group

This is where the relevant signal group name is displayed.

"Group" is hidden by default.

Direction

The transfer direction of the relevant signal group (and thus the signal) is displayed here.

"Direction" is hidden by default.

Task

This is where the assigned task is displayed in which the relevant signal group is transferred.

"Task" is hidden by default.

Signal

The signal name is displayed here.

Data

The values of the hardware signal are displayed or values can be entered in this column.

Explanation

This column contains explanations about the signals. It is not available for every item.

1.3.6 The Tab Shortcut Menu

Right-clicking in one of the tabs opens a shortcut menu via which all or some of the following menu items can be reached.

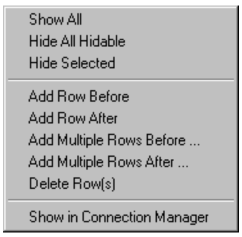


Fig. 1-4 The Tab Shortcut Menu

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

  This menu item shows all displayable options/columns of a tab.

  The table of a tab can be structured so that some of the possible options or columns are not displayed by default. This has the advantage that options which are not used often are hidden and the table can be structured more clearly.

Note

If the table is then confirmed with the **Accept** *button and the hardware configuration is saved, the table is once again available in the form in which it was saved, i.e. newly displayed options/columns remain displayed.*

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

  This is the opposite of the previous menu item, i.e. all options/columns which can be hidden are hidden.

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

  Hides the option or column currently displayed, providing they can be hidden.

- **Add Row Before** (E)

  This menu item is only available for items of the type "Device" providing they make it possible to change their number of signal groups or signals dynamically.
  The menu item is available and can be selected under the following circumstances:

  1. An item of the type "Device" is selected which has "dynamic" properties (e.g. CAN-IO Device).
  2. The "Groups" or "Signals" tab has been selected.
  3. An item in the column "No." has been selected (1st column in the table).

  This menu items adds a line before the selected cell or area.

Note

The size of the table is often restricted due to the system. In this case, executing the function often has no effect. As this is a structural change to the table, this change cannot be undone by the **Reset** *button.*

- **Add Row After**

  Works exactly the same as **Add Row Before** except a new row is added after the selected cell or the selected area.

- **Add Multiple Rows Before**

  Works exactly the same as **Add Row Before** except that several rows can be added here at a time. The number of lines to be added can be entered in a dialog box.

- **Add Multiple Rows After**

  Works exactly the same as **Add Multiple Rows Before**, except new rows are added after the selected cell or the selected area.

- **Delete Row(s)**

  Deletes the row or area which is marked in the column "No.".

Note

For system reasons, at least one signal group or a signal in the table has to remain. When you attempt to delete the last row, an error message is displayed.

- **Show in Connection Manager**

  Selecting this menu item results in the Connection Manager being opened. The selected hardware signal is displayed (in the "Inputs" or "Outputs" list) and any existing connections to other (model or hardware) inputs or outputs are displayed.

1.3.7 Options

Via the menu item **Extras** → **Options...** a window with several tabs is activated in which a range of options can be set.

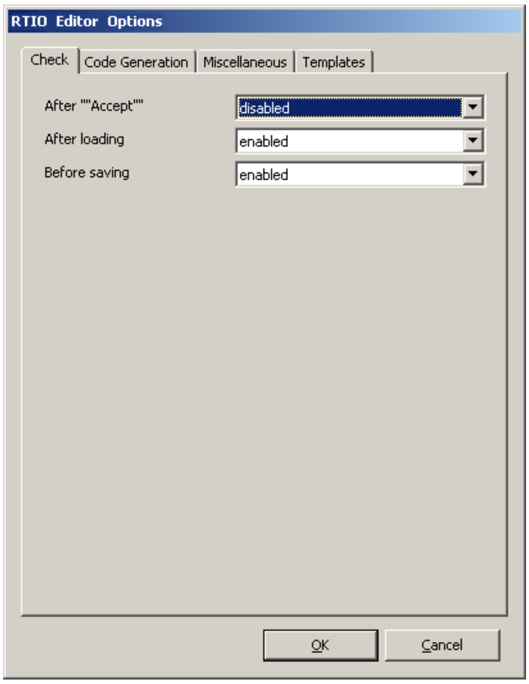


Fig. 1-5 The "RTIO Editor Options" Window

"Check" Tab

This is where you can specify when automatic checks of the hardware configuration or data consistency should take place.

- After "Accept"

  Activates ("enabled") or deactivates ("disabled") the automatic check after the **Accept** button has been pressed.
  For performance reasons, only the data consistency within the tab currently selected is checked.

- After loading: enabled/disabled

  Activates ("enabled") or deactivates ("disabled") the check after a new hardware configuration has been loaded.
- Before saving: enabled/disabled

  Activates ("enabled") or deactivates ("disabled") the automatic check of the hardware configuration before a configuration is saved.

"Code Generation" Tab

This is where specific options for code generation can be set.

- Automatic repair

  Activates ("yes") or deactivates ("no") the possibility to automatically recover certain errors in the hardware configuration. This possibility is only supported to a very limited extent.
- Minutes type

  Specifies the scope of the protocol ("no/small/detailed") generated during RTIO code generation.

"Miscellaneous" Tab

The two options in this tab specify the format in which the values of the "Data" column are displayed in the "Data" tabs of the devices.

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

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

"Templates" Tab

Options for the item templates are specified in this tab.

Item templates are used to overwrite the default values specified by the system with your own settings. With an activated template option, the default values specified by the system are overwritten by those saved in the Template directory (**File → Save Item As Template...**) each time an item is added (**Edit → Add Item...**) .

- Template Path

  This is where the directory can be selected which is to be used as the storage place for the item templates.
- Use Item Templates

  Activates ("yes") or deactivates ("no") the use of item templates.

2 ES1220.1 CAN Board

The ES1220.1 CAN Board is used as a CAN interface in VMEbus systems. The board has a digital signal processor (DSP) which works independently of the VME system. The board provides four CAN channels which have separate CAN controllers. Each of the four CAN controllers of type Intel 82527 has 15 buffers for storing incoming and outgoing CAN messages. 14 of these buffers can process both send and receive messages. The 15th buffer can only process receive messages. By using a DSP, high data rates and short response times to incoming CAN messages are guaranteed.

The board has the following functions:

- digital signal processor TMS320F206
- four CAN controllers of the type Intel 82527
- 15 message buffers per CAN controller (14 send & receive buffers, 1 receive buffer)

The ES1220.1 can be used anywhere in VMEbus systems where CAN messages are to be generated and received. The digital signal processor (DSP) makes it possible to process messages flexibly and independently of the system processor.

Examples of areas of implementation are:

- simulation of vehicle modules CAN interface
- simulation of the vehicle CAN bus
- control of load simulations

This document describes the RTIO integration of the ES1220.1.

In the RTIO-Editor, the ES1220.1 is integrated by selecting the ES1220 CAN item.

2.1 ES1220-CAN Subsystem

2.1.1 Globals (ES1220-CAN Subsystem)

In the "Globals" tab of the ES1220-CAN subsystem, the user can specify the VMEbus base address of his/her ES1220.1 in the "ID / VME base address" field (see Fig. 2-1). You can choose from between 3 different base addresses (`ID1/740000h`, `ID2/780000h` and `ID3/7C0000h`).

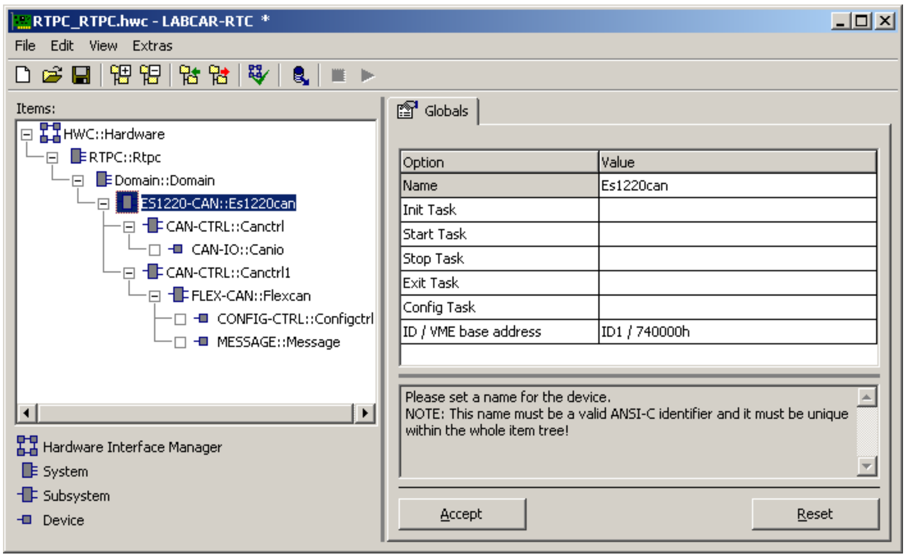


Fig. 2-1 The "Globals" Tab of the ES1220-CAN Item

Up to four CAN-CTRL subsystems can be allocated to the ES1220-CAN subsystem. These CAN-CTRL subsystems correspond to the four Intel 82527 CAN controllers on the board.

2.2 CAN-CTRL Subsystem

2.2.1 Globals (CAN-CTRL Subsystem)

In the "Globals" tab of a CAN-CTRL subsystem (see Fig. 2-2) a physical CAN controller or CAN connector is allocated to the CAN-CTRL subsystem. The CAN connectors on the front panel are numbered from top to bottom with 1 to 4.

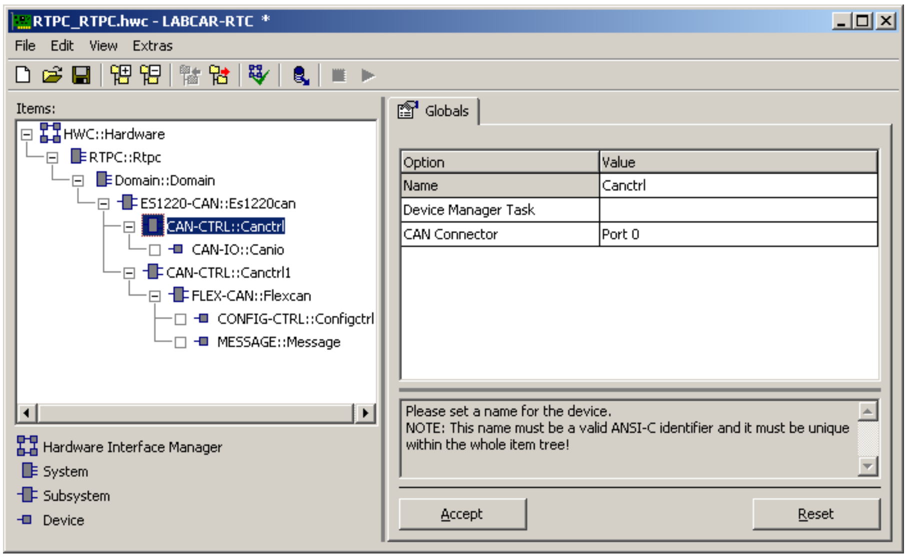


Fig. 2-2 The "Globals" Tab of a CAN-CTRL Subsystem

Device Manager Task

The device manager API function handles "Bus-Off" detection and recovery. The device manager process should be assigned to a periodically activated task. A task activation period of 100 milliseconds or even longer is sufficient.

The device manager task assignment is optional. No "Bus-Off" handling will be done if no task is assigned but code generation is possible.

Each of the 4 CAN controllers of the ES1220.1 can be configured for two different applications.

What is referred to as the FlexCan application allows all software options made available by a CAN controller to be used. For example, modification of message ID, frame type and the transfer direction of a message buffer in runtime are possible in real time. Further options, such as the baud rate and the acceptance filter, can also be configured during runtime.

The FlexCan application is selected by assigning a FLEX-CAN subsystem to the CAN-CTRL subsystem (see Fig. 2-3). All options of a FLEX-CAN subsystem can be controlled from the simulation model using RTIO signals.

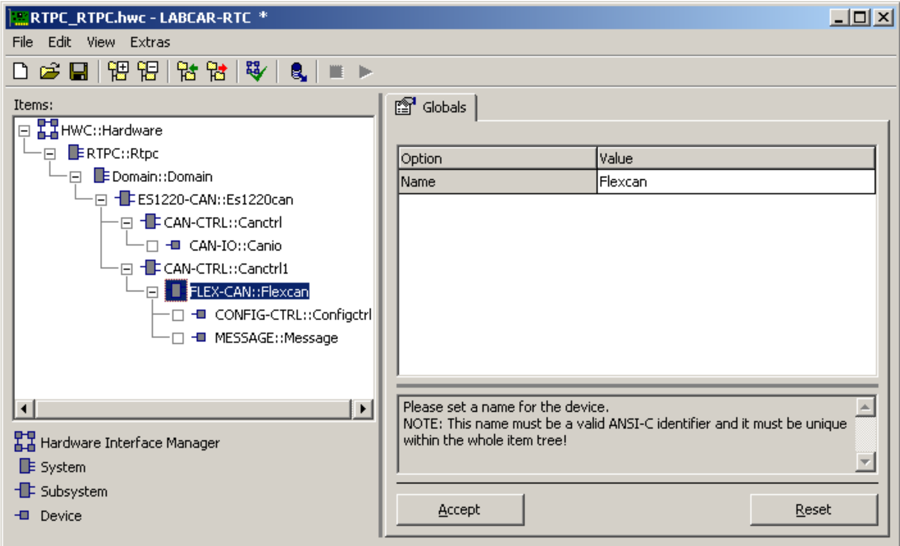


Fig. 2-3 The ES1220 Item Tree with FLEX-CAN Subsystem and CAN-IO Device

2.3 FLEX-CAN Subsystem

A CONFIG-CTRL device and up to 15 MESSAGE devices can be allocated to the FLEX-CAN subsystem. Each of the 15 MESSAGE devices is used to configure one of the 15 CAN message buffers physically present on a CAN controller. The CONFIG-CTRL device is used to configure the remaining parameters of a CAN controller. These parameters have an effect on the response of all 15 message buffers.

Note

A CONFIG-CTRL device has to be added before a MESSAGE device can be assigned to a FLEX-CAN subsystem.

2.4 CONFIG-CTRL Device

2.4.1 Groups (CONFIG-CTRL Device)

The CONFIG-CTRL device has a signal group called "Control" with which the global settings valid for all 15 message buffers of a CAN controller are configured. These configuration parameters are:

- baud rate
- global acceptance mask for CAN messages in "standard format"
- global acceptance mask for CAN messages in "extended format"
- local acceptance mask for message buffer 15

Unlike the procedure normally used, the calculation of the "Control" signal group in the "Groups" tab cannot be assigned to any task of the real-time operating system (see Fig. 2-4). The calculation of the "Control" signal group of a CONFIG-CTRL device takes place implicitly in the "Start" and "Config" processes of the RTIO driver. These processes are allocated to tasks of the real-time operating system in the "Globals" tab of the ES1220-CAN subsystem (see Fig. 2-1 on page 52).

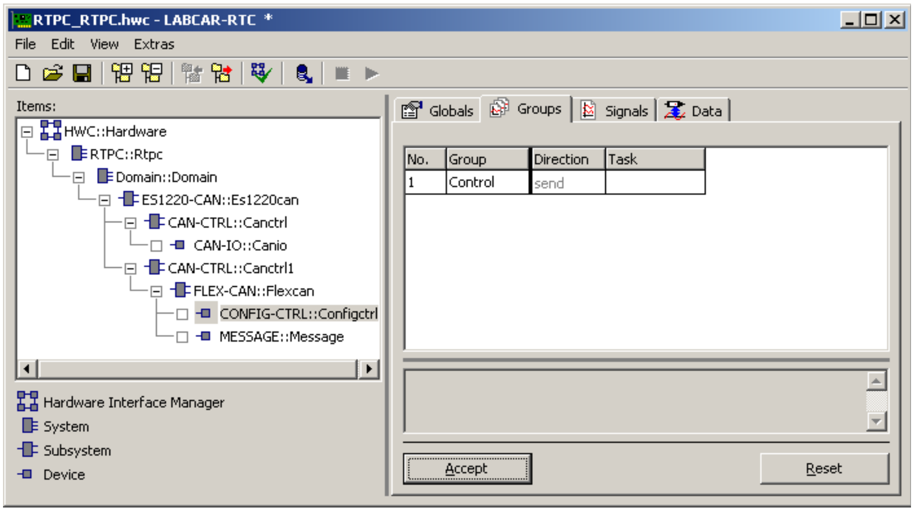


Fig. 2-4 The "Groups" Tab of the CONFIG-CTRL Device

2.4.2 Signals (CONFIG-CTRL Device)

Fig. 2-5 shows the "Signals" tab of the CONFIG-CTRL device.

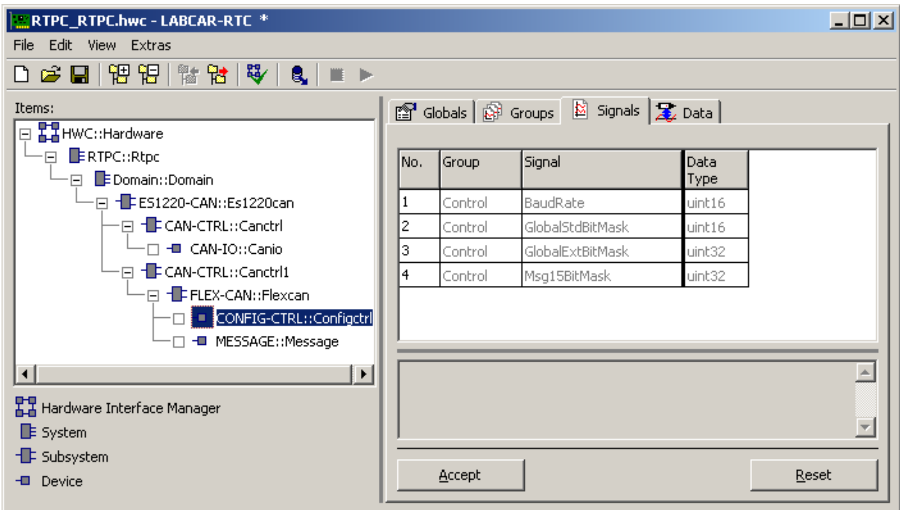


Fig. 2-5 The "Signals" Tab of the CONFIG-CTRL Device

The receive branch of an Intel 82527 CAN controller contains two global filter masks which can be used to indicate which of the identifier bits entered in the message buffers have to correspond to those of a received CAN message for it to be accepted in the relevant message buffer.

These masks thus determine which bits are relevant for message filtering. In this way, the individual buffers can be limited either to receiving one identifier each or extended to receive identifier groups. One of the two global filter masks which is configured with the RTIO signal, *GlobalStdBitMask,* is responsible for filtering CAN messages with standard (11-bit) identifiers. The second global filter mask which is configured with the RTIO signal, *GlobalExtBitMask,* is responsible for filtering CAN messages with extended (29-bit) identifiers. Another local filter mask is preconnected to message buffer 15 (which can be configured to receive only but not send CAN messages). This filter mask is configured with the RTIO signal, *Msg15BitMask*. The relevant identifier bits are shown by a "1" in the filter masks; the irrelevant ones with a "0".

When received, a check is made to see whether the message can be accepted in one of the first 14 message buffers, taking the global filter masks into account. If this is not the case, the global mask of the message format ("Standard" or "Extended") is AND-linked bit-by-bit with the local mask of message buffer 15 and a check is then made to see whether this message can be accepted into this message buffer based on the resulting filter mask. If this is not possible, the message is rejected.

If, for example, all message buffers are configured as receive buffers, the filter mechanism only allows one specific message to be accepted in each of these buffers, and all other messages are accepted into message buffer 15. For this purpose, all bits in the two global filter masks have to be defined as relevant (bit value 1) and all bits in the local filter mask for message buffer 15 as not relevant (bit value 0).

Tab. 2-1 lists the RTIO signals of the CONFIG-CTRL device.

Note

As described in "Groups (CONFIG-CTRL Device)" on page 54, the "Control" signal group of the CONFIG-CTRL device *is processed implicitly in the "Start" and "Config" processes of the RTIO driver. This is why the user must ensure that the RTIO signals of the "Control" signal group are initialized with admissible values before a "Start" or "Config" process is processed for the first time.*

RTIO Signal	Data Type	Description
BaudRate	uint16	Data transfer rate in kBaud Admissible values: 10, 20, 50, 100, 125, 250, 500, 1000
GlobalStdBitMask	uint16	Global acceptance bit mask for CAN messages in "standard format". The lowest or least-significant 11 bits are processed. Admissible values: 0x0000 ... 0x07FF Bit value 0: don't care. A 0 or 1 is accepted at this bit position. Bit value 1: must match. The bit value of a CAN message received must correspond exactly in this bit position with the bit value of the identifier of a message buffer
GlobalExtBitMask	uint32	Global acceptance bit mask for CAN messages in "extended format". The lowest or least-significant 29 bits are processed. Admissible values: 0x0000 0000 ... 0x1FFF FFFF Bit value 0: don't care Bit value 1: must match
Msg15BitMask	uint32	Local acceptance bit mask for message buffer 15. The lowest or least-significant 29 bits are processed. Admissible values: 0x0000 0000 ... 0x1FFF FFFF Bit value 0: don't care Bit value 1: must match **Warning**: The local acceptance bit mask for message buffer 15 is AND-linked bit-by-bit with the global acceptance bit masks. This means that each bit defined in the global acceptance bit mask of the CAN message format as "don't care" is automatically a "don't care" bit in the acceptance bit mask for message buffer 15.

Tab. 2-1 The RTIO Signals of the CONFIG-CTRL Device

Note

When data types are specified in this document, they are the data types used internally by the RTIO driver.

2.5 MESSAGE Device

2.5.1 Globals (MESSAGE Device)

The MESSAGE device is used to configure a message buffer of the Intel 82527 CAN controller.

In the "Globals" tab of a MESSAGE device (see Fig. 2-6), the MESSAGE device is assigned to a physical message buffer of the CAN controller in the "CAN Message Buffer" field.

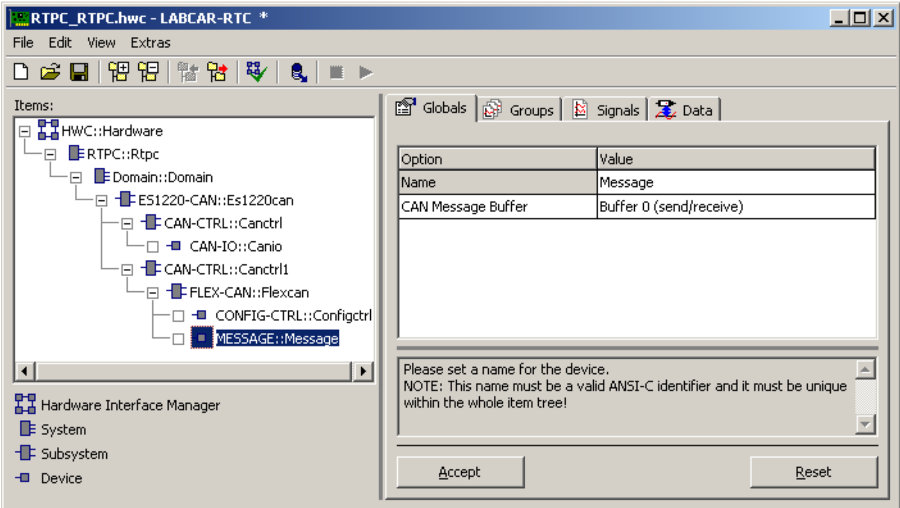


Fig. 2-6 The "Globals" Tab of a MESSAGE Device

2.5.2 Groups (MESSAGE Device)

A MESSAGE device has three signal groups (see Fig. 2-7):

- Control

  The "Control" signal group is transferred from the simulation target to the ES1220.1. This signal group contains all signals which concern both the send and receive behavior of a message buffer.
- Transmit

  The "Transmit" signal group is also transferred from the simulation target to the ES1220.1. This signal group contains all signals which configure the send behavior of a message buffer.
- Receive

  The "Receive" signal group is transferred from the ES1220.1 to the simulation target and contains all signals which describe the data of a received CAN message.

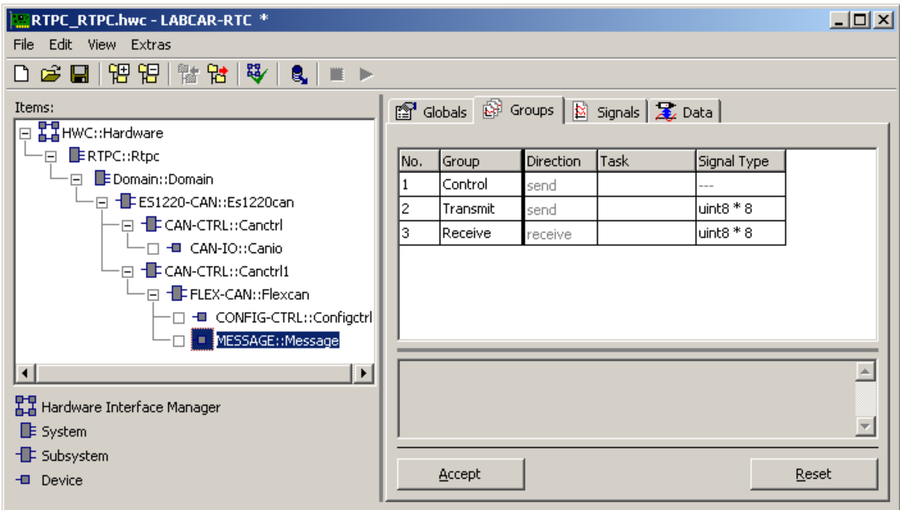


Fig. 2-7 The "Groups" Tab of a MESSAGE Device

2.5.3 Signals (MESSAGE Device)

Fig. 2-8 shows the "Signals" tab of a MESSAGE device.

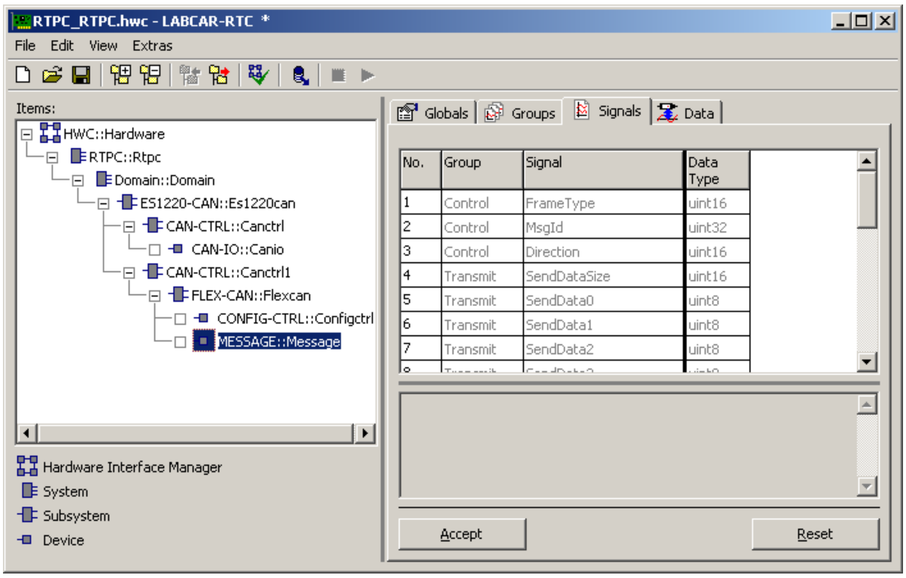


Fig. 2-8 The "Signals" Tab of a MESSAGE Device

FrameType

With the *FrameType* signal, the format (standard or extended) of the CAN messages is determined which are sent in send mode by the buffer and received in receive mode by the message buffer.

MsgId

If the message buffer is configured to send, the identifier of a message to be sent is defined with the *MsgId* signal. If the message buffer is configured to receive, the *MsgId* signal defines the identifier of the CAN messages which are accepted in the message buffer. Which messages are accepted into a message buffer in receive mode also depends on the settings of the global and local acceptance bit masks (see section 2.4.2 on page 55).

Direction

The *Direction* signal configures the transfer direction of the relevant CAN message buffer. A buffer can either be configured to send or receive.

SendDataSize and SendDatax

If the CAN message buffer is configured to send, the *SendDataSize* signal specifies the number and the *SendDatax* (x = 0, 1, ... 7) signals specify the contents of the data types to be sent. These signals are of no significance in receive mode.

The *RecDataNewMes*, *RecDataSize*, *RecDataFrameType*, *RecDataMsgId* and *RecDatax* (x = 0, 1, ... 7) signals are only relevant for the receive mode of a buffer.

RecDataNewMes

The *RecDataNewMes* signal specifies whether a new CAN message has been received since the CAN message buffer was last read.

RecDataSize

The *RecDataSize* signal specifies the number of data bytes in the CAN message received.

RecDataFrameType

The *RecDataFrameType* signal specifies the message format of the received CAN message. This information is redundant since it always has to correspond to the value set of the *FrameType* signal.

RecDataMsgId

The *RecDataMsgId* signal specifies the identifier of the CAN message received. The value of *RecDataMsgId* does not necessarily have to correspond to the value set for *MsgId*. This is only necessary if all bits were set to "must match" in the acceptance bit masks (see section 2.4.2 on page 55).

RecDatax

The *RecDatax* signals contain the values of the data bytes of the CAN message received.

Tab. 2-2 lists the features of the RTIO signals of the MESSAGE device.

RTIO Signal	Data Type	Signal Group	Description
FrameType	uint16	Control	Message format; 0: standard format (11-bit identifier) 1: extended format (29-bit identifier)
MsgId	uint32	Control	Message identifier
Direction	uint16	Control	Transfer direction; 0: receive; 1: send
SendDataSize	uint16	Transmit	Length of the send message in bytes; Admissible values: 0 ... 8
SendData0 ... SendData7	uint8	Transmit	Data bytes
RecData-NewMes	uint16	Transmit	Indicator for the receipt of a new CAN message; 0: no new CAN message has been received since the message buffer was last read 1: new CAN message received
RecDataSize	uint16	Receive	Number of received data bytes; Value range: 0 ... 8
RecDataFrame-Type	uint16	Receive	Format of CAN message received; 0: standard format (11-bit identifier) 1: extended format (29-bit identifier)
RecDataMsgId	uint32	Receive	Identifier of the CAN message received
RecData0 ... RecData7	uint8	Receive	Received data bytes

Tab. 2-2 The Signals of the MESSAGE Device

2.6 CAN-IO Device

2.6.1 Globals (CAN-IO Device)

The ES1220 CAN-IO device provides the user with a user interface with which CAN messages and signals can be specified very easily. It is effectively based on the Flex-CAN integration; but the runtime configurability is done without here to simplify matters.

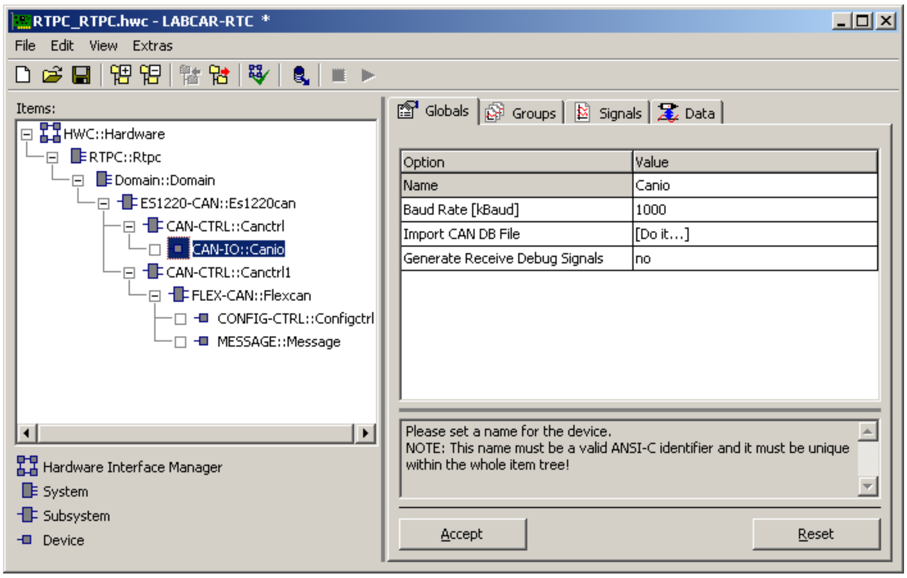


Fig. 2-9 The "Globals" Tab of the CAN-IO Device

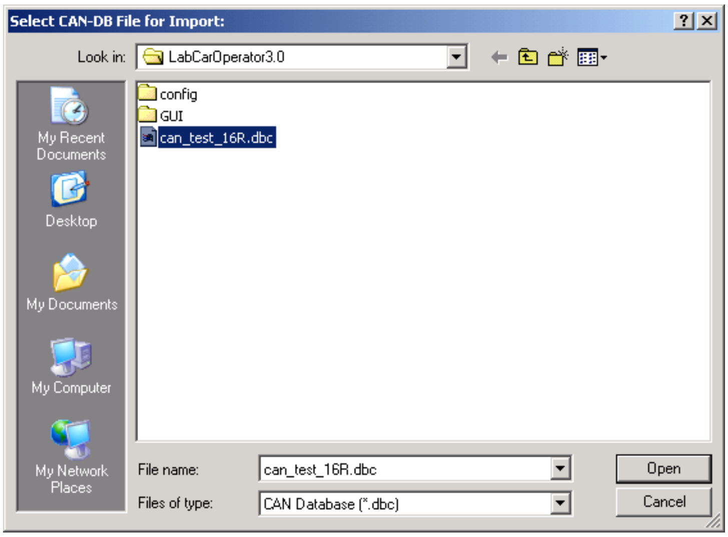
Baud Rate [kBaud]

This is where you can select the required baud rate. The standard baud rates are available (1000, 500, 250, 125, 100, 50, 20,10 kBaud). Unlike the ES1222 CAN IO item, a `<Special Timing>` setting is not possible here.

Import CAN DB File

This option is used to read a CAN database file which was created with the CANdb data management program made by the company Vector Informatik. CAN messages and signals can be generated automatically if necessary using this file.

A dialog box opens after you press [Do it...]:



The CAN DB file to be imported can be selected in this dialog box.

After clicking the "Open" button, the following dialog box is displayed and can be used to specify what happens next:

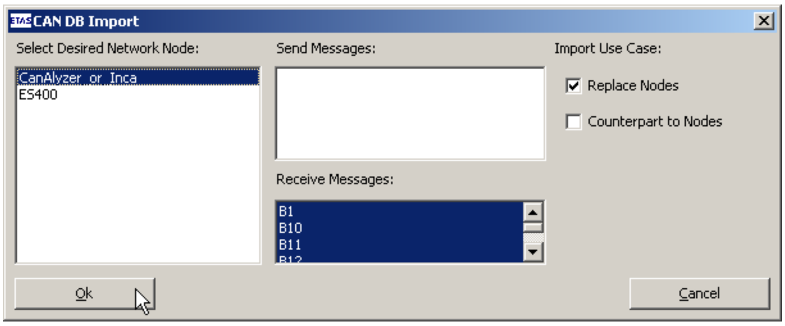


Fig. 2-10 CAN DB Import Dialog

A CAN DB file normally describes several nodes of a CAN network. All existing nodes are listed in the list on the left (Network Nodes). The two lists to the right of it (Send Messages / Receive Messages) list all CAN messages defined for the node currently selected. The messages selected are used for import. "Import Use Case" can also be selected. The "Replace Nodes" option means that the CAN-IO device assumes the role of the network node; i.e. a send message of the node is also used as a send message of the CAN-IO device etc..

The "Counterpart to Nodes" option means that the CAN-IO device is the counterpart to the network node; i.e. a send message of the node results in a receive message etc.

After you confirm with "OK", the data to be imported (CAN messages and signals) with the signal groups and signals is checked. The result of this check is then shown in the dialog box displayed below:

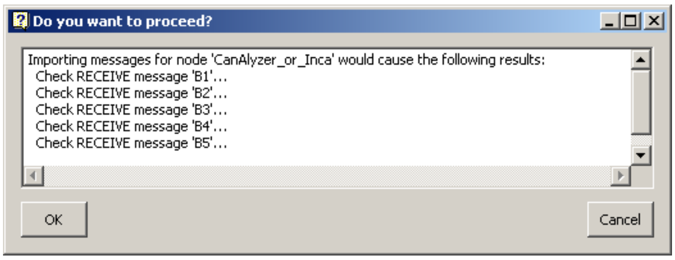


Fig. 2-11 "CAN DB Check" Dialog Box

So far, the existing CAN-IO device has not been changed. The actual import procedure does not start until the "OK" button is activated.

The import procedure inserts the imported messages in the signal groups and ensures that relevant signals are defined.

Once import is completed, the detailed protocol of the import procedure is shown in the "Monitor" dialog box:

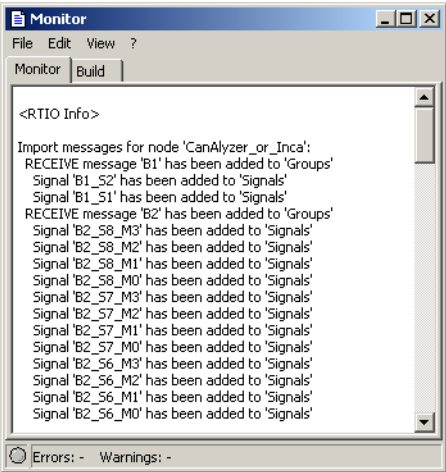


Fig. 2-12 "Monitor" Window with Import Protocol

Attention during import:

When a CAN DB file is imported, only 29 Bit (identifier *extended*), or 11 Bit (identifier *standard*; cf. "Identifier Type" on page 67), are inserted into the identifier field automatically.

When you selected the *standard* identifier in the CAN Controller item, but the CAN DB file contains signals with 29 Bit identifiers (ID > 2^{31}), two things happen:

- 11 bits (bits [28...18]) are inserted into the identifier field. The remaining bits (MSB) are rejected.
- Warnings are displayed in the monitor window.

Since only one identifier can be selected, CAN DB files containing both *standard* and *extended* identifiers cause problems.

Generate Receive Debug Signals

If this option is activated (= `yes`), two additional signals are generated for every "receive" signal group.

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

The *...dT* signal specifies the difference in seconds to the previous message received.

Note

This signal can be used for a normal message receipt (IRQ = no) to monitor the receipt. If, for example, the CANbus is interrupted, the value increases permanently in the grid of the receive task. This value is not protected against overflow (which occurs at approx. 300s) for performance reasons!

In contrast, no real receipt monitoring can be executed when a message is received with an interrupt (`IRQ = yes`) as the calculation of the value does not take place until the interrupt task. As the calculation does not take place if no message is received, the old value is frozen. This means that it is impossible to tell when an interrupt is received, whether the receipt takes place in a regular grid or whether it is interrupted.

The *...Rec* signal is described as `true` every time a message is received. If this signal is mapped to a send-receive message and then reset to `false` by the application every time it is read, the application can easily determine whether a message has been received between the cycles.

2.6.2 Groups (CAN-IO Device)

The CAN messages are specified in the form of signal groups in this CAN-IO tab.

Note

Unlike the ES1222 CAN-IO device, the number of signal groups is limited to 15 here. The first 14 signal groups can be used for both the "send" and "receive" direction; the 15th signal group can only be used for a "receive" message. This CAN-IO device does not support any messages received in interrupt mode.

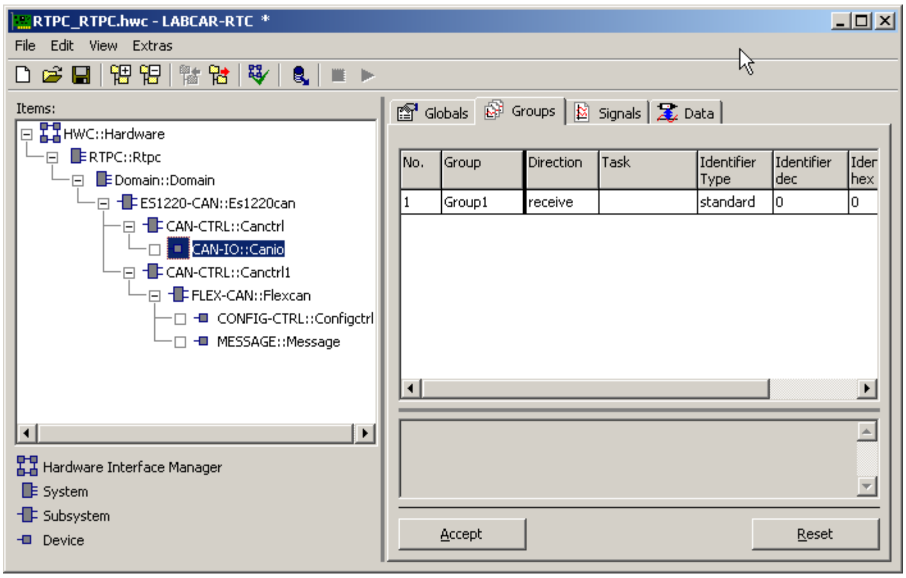


Fig. 2-13 The "Groups" Tab of the CAN-IO Device

Note

After changing the group name, the signal name, or the signal direction, an ASD message mapped previously may not be mapped automatically any longer and then has to be mapped again manually.

Direction

This is where you can determine the direction of the CAN message ("send" = send message, "receive" = receive message).

Task

This is where the task is specified in which the message is to be sent or received. If a receive message is to be received in interrupt mode, this setting is reset and locked.

Identifier Type

This is where you can specify whether the identifier type should be interpreted in standard or in extended format.

With CAN messages, you can choose between standard frames with 11-bit identifiers or extended frames with 29-bit identifiers. The length of the identifier field (standard / extended) can be specified in this line.

When the standard identifier has been selected, only 11 bit values are allowed in the "Groups" tab of the CAN-IO device (cf. "Identifier dec/hex" below). When you enter larger values, the most significant bits (MSB) are truncated. A warning is **not** given.

When the extended identifier has been selected, only 29 bit values are allowed. When you enter larger values, the most significant bits (MSB) are truncated. A warning is **not** given

Identifier dec/hex

This is where the message identifier is entered. The value can vary in size here depending on the setting chosen for the identifier type in the superior CAN controller item (standard or extended):

Standard identifier:	11-bit	`2 047` dec	`7 FF` hex
Extended identifier:	29-bit	`536 870 911` dec	`F FF FF FF` hex

Each signal group or CAN message has to have a unique identifier.

Length [Byte]

Specifies how many useful data bytes the relevant message can transfer (1..8).

*Activated Task**

For "receive" CAN messages[1] (signal groups), a software task may be entered here that is always activated when the relevant signal group has been received, after the receiving task has been executed. The task entered here can, e.g., be used for post-processing like cleaning up.

This column is hidden by default.

2.6.3 Signals (CAN-IO Device)

The CAN I/O signals are specified in this CAN-IO tab.

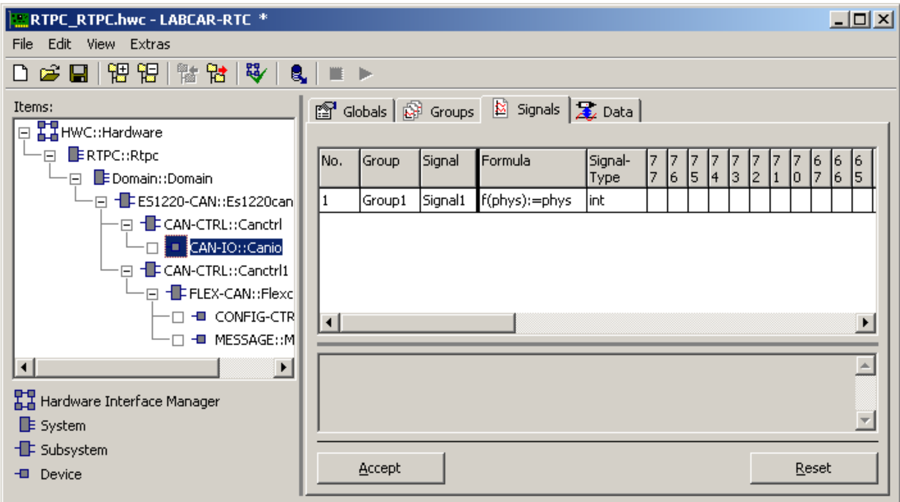


Fig. 2-14 The "Signals" Tab of the CAN-IO Device

Note

New signals can be generated via the context menu in the "Signals" tab (for more details please refer to "The Tab Shortcut Menu" on page 47).

Group

This is where a signal is allocated to the required signal group.

1. It is possible to specify such a task for "send" groups, too, but there is no recommended usage for such a utilization. (As the sending process is handled asynchronously, the CAN message may or may not have been sent when that task is activated.)

Signal Type

This is where the signal type is determined in which the signal is transferred via the CAN bus.

The following settings are possible:

Signal Type	Data Type
int	Characterizes a signed signal in the default complement to two data format (max. 32-bit)
(s)int	Characterizes a signed signal in which the sign is transferred as the most significant bit and then the absolute value of the signal is transferred. If the sign bit is set, this is a negative number (max. 32-bit)
uint	Characterizes an unsigned signal (max. 32-bit)
bool	Characterizes a Boolean signal. Only one bit can be marked in the bit matrix.
real	Characterizes a floating-point value in "standard IEEE float (4 byte)" format. Accordingly, only 32 bits can be marked in the bit matrix.

The following table provides an overview of the IEEE floating-point formats:

Format	Sign	Exponent	Fraction
Float	1 bit	8 bits	24 bits
Double (not supported)	1 bit	11 bits	52 bits

7654321.. (Bit matrix)

A CAN message can transfer up to 8 data bytes. A bit matrix can specify which bits each signal requires or occupies (a signal = a row).

The columns are structured as follows:

7	7	...	0	0	Byte number
7	6	...	1	0	Bit number

Significance of the bit fields:

Empty field	The relevant signal does not use the bit
Occupied field	The signal requires this bit at the position
"X" field	The relevant bit is not available for data transfer because the signal has fewer useful data bytes (see "Length" in the "Groups" tab).

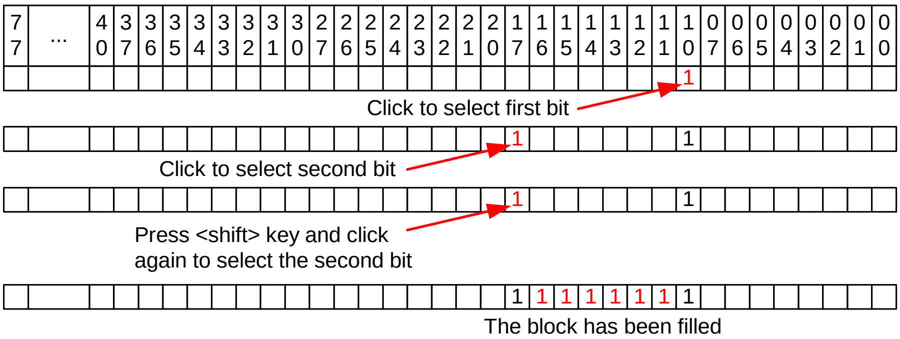
Operating the bit fields:

The required bit cell can be selected using the arrow keys on the keyboard.

Use the space bar to "toggle" between "empty" and "1" (bit assigned).

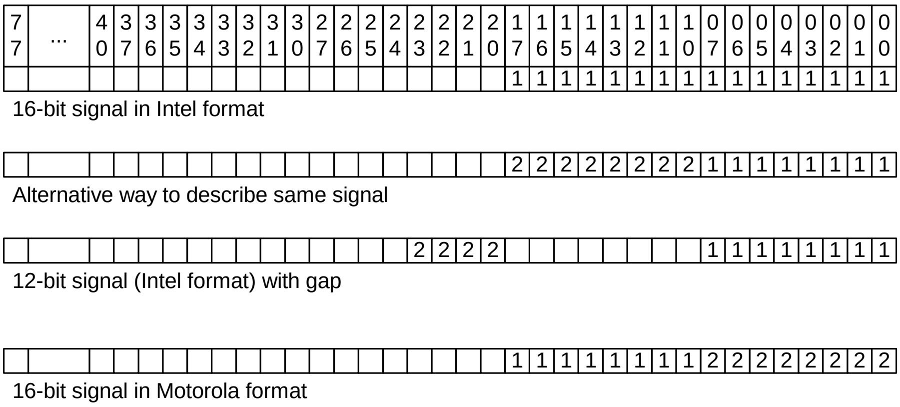
Clicking with the mouse also "toggles" the relevant cell between "empty" and "1". If you press the <Shift> key and click the mouse simultaneously, the value is incremented from "1" to "9" (important for "block building").

You can select several bits simultaneously as follows:



Data blocks can be created using different numbers ("1111 2222...") with which virtually every signal transferred can be described. The numbers used to form the data blocks have the significance that the block with the highest number ("2222") specifies the block which contains the most significant bits during transfer. The block with the smallest number ("1111") contains the least significant bits during transfer. With the numbers available (1...9), you can write a signal with up to 9 bit blocks. As the representation of the bits corresponds exactly to the form in which an Intel signal is transferred, block building is necessary for the representation of signals in Motorola format as soon as the signal is longer than 8 bits.

Examples of the definition of different signals:



3 ES1302.1 A/D Board

The ES1302.1 A/D Board is used for analog data acquisition in VMEbus systems. The board is designed for medium resolutions and data rates. It includes the following functions:

- A/D converters with 12-bit resolution and a maximum overall sampling rate of 300 kHz
- 16 single-ended or 8 differential analog input channels
- four input voltage ranges and four gain factors which can be set by software
- FIFO memory for measured values and channel numbers

The ES1302.1 can be used anywhere in VMEbus systems where analog input signals with medium resolution are to be acquired.

Examples of areas of implementation are:

- acquisition of analog transducer signals, for example engine temperature, oil temperature, accelerator setting
- valve flow measuring with a primary I/U converter

3.1 ES1302-AD Device

3.1.1 Globals (ES1302-AD Device)

This section describes the global options of the ES1302-AD device.

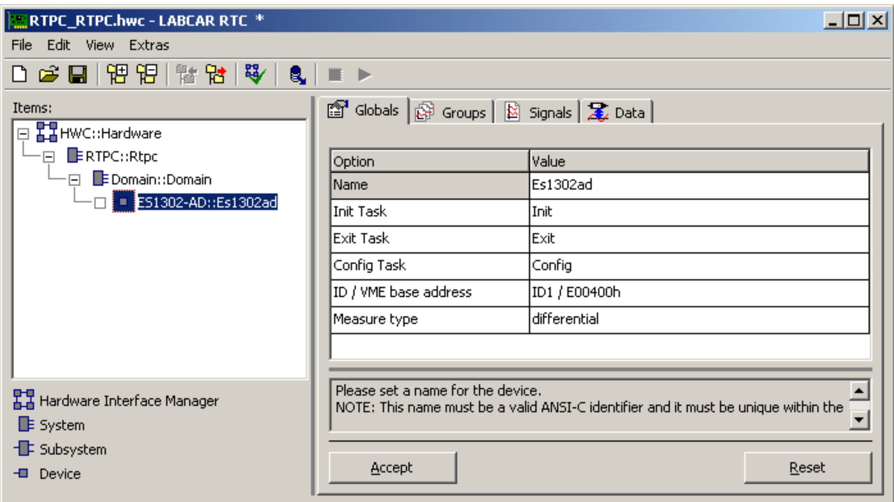


Fig. 3-1 The "Globals" Tab of the ES1302-AD Device

Init Task

The task for initializing the ES1302.1 is assigned in this line (Type: Init / Application Mode: active).

Exit Task

The task to be executed with the ES1302.1 when the experiment stops is assigned in this line (Type: Init / Application Mode: inactive).

Config Task

This task can be used to reconfigure the ES1302.1 during experimentation. Please create a software task (which must *not* be called) in the OS editor when you do not want to reconfigure the ES1302.1.

ID / VME base address

The VMEbus base address of the ES1302.1 is specified in the "ID / VME base address" field. There are 5 different base addresses to choose from (`ID1/E00400h`, `ID2/E00500h`, `ID3/E00600h`, `ID4/FE0100h` and `ID5/FE0200h`).

Measure type

The type of measuring is specified in the "Measure type" field. There are three options:

- single-ended: 16 single-ended analog input channels are measured.
- differential: 8 differential analog input channels are measured.
- reference: the 4 internal reference voltage channels are measured.

Note

To ensure that the input voltages are measured correctly, all unused inputs have to be at a defined potential.

3.1.2 Groups (ES1302-AD Device)

This section describes the signal-group-specific options of the ES1302-AD device.

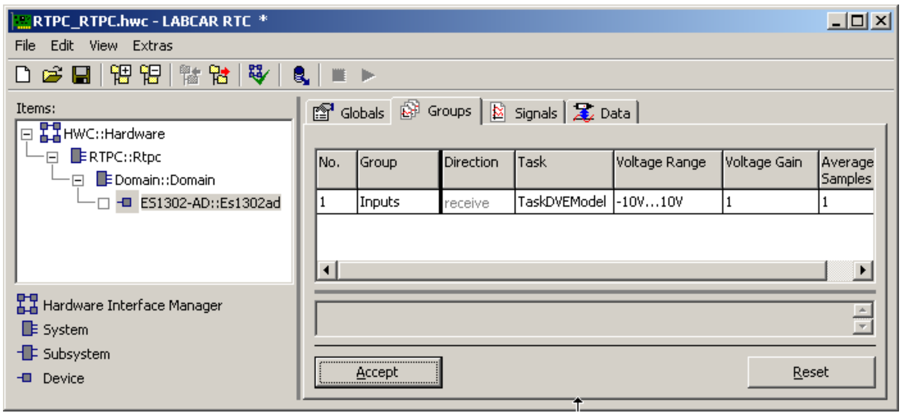


Fig. 3-2 The "Groups" Tab of the ES1302-AD Device

Voltage Range

The input voltage range of the measure channels is specified in the "Voltage Range" field. There are four ranges to choose from (0 V ... 5.0 V, -5.0 V ... 5.0 V, 0 V ... 10.0 V and -10.0 V ... 10.0 V).

Voltage Gain

The "Voltage Gain" field determines an input voltage gain for the measure channels. There are four gain factors to choose from (1, 2, 4, 8). Please note that this does not change the input voltage range, i.e. the voltage gain must be within the voltage range set!

Average Samples

The "Average Samples" field determines how many sample values are used to calculate an average analog voltage value. Averages can ensure interference or noise suppression - please note, however, that approx. 3 µs are required per measure channel and per sampling.

3.1.3 Signals (ES1302-AD Device)

This section describes the signal-specific options of the ES1302-AD device.

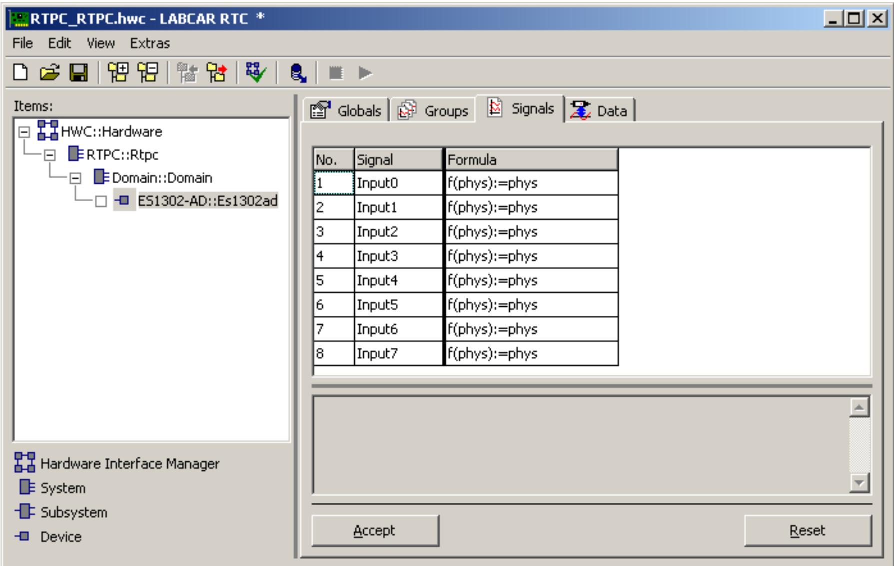


Fig. 3-3 The "Signals" Tab of the ES1302-AD Device

The number of signals depends on the measure type set in the "Measure type" field in the "Globals" tab:

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

The *Inputx* signals are either of data type "float" or "real32". The signals specify the measured voltage at the relevant measure channel in volts.

Note

When data types are specified in this document, they are the data types used internally by the RTIO driver.

4 ES1321.1 PWM I/O Board

The ES1321.1 PWM I/O Board is used to issue and acquire pulse-width modulated signals. A typical example of how the ES1321.1 is used is, e.g., in LABCAR projects where it is used to simulate vehicle sensors and address actuators.

The ES1321.1 PWM I/O Board is integrated in the RTIO-Editor by selecting the ES1321 subsystem.

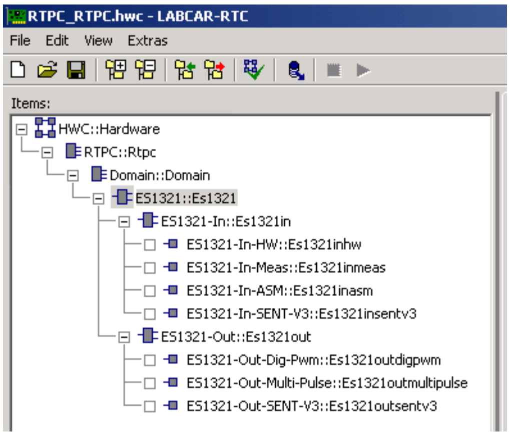


Fig. 4-1 RTIO Hardware Description with ES1321.1 PWM I/O Board

The ES1321.1 PWM I/O Board has 24 input channels for acquiring pulse-width modulated signals. These input channels are configured with the ES1321-In-HW subsystem. In addition to this subsystem, up to four ES1321-In subsystems can be assigned to the ES1321-RTIO element.

An ES1321-Out subsystem can consist of one ES1321-Dig-Pwm, up to two ES1321-Multi-Pulse devices and up to four ES1321-Out-SENT devices.

Measure Functions of the ES1321.1 PWM I/O Board

The inputs of the ES1321.1 have the following measure functions:

- Frequency
- Cycle Time
- Hightime / Lowtime
- Additive Hightime / Additive Lowtime
- Number of Low/High Pulses
- Number of Low/High Edges
- Duty cycle for high and low active signal
- Position tracing for stepper motors
- SENT-Receive

The outputs of the ES1321.1 have the following functions:

- Digital Output
- PWM signal generation
- Multi-Pulse
- SENT-Transmit

4.1 ES1321 Subsystem

4.1.1 Globals (ES1321 Subsystem)

This section describes the global options of the ES1321 subsystem.

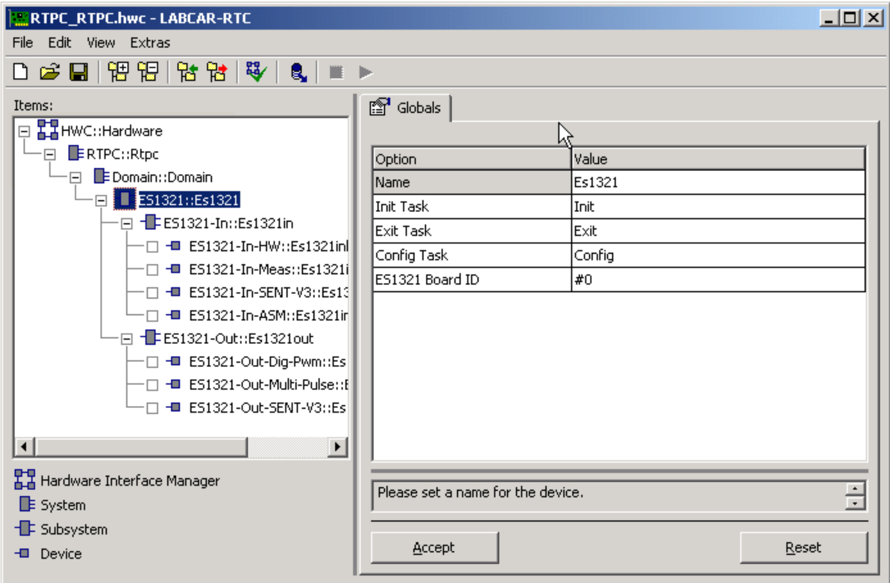


Fig. 4-2 The "Globals" Tab of the ES1321 Subsystem

ES1321 Board ID

This option field is used to identify the ES1321.1 PWM I/O Board. It establishes the assignment between the RTIO hardware description and the ES1321.1 in the VMEbus chassis for which this description is valid.

The numbering of the ES1321.1 in the chassis takes place from left to right (left = VMEbus slot no. 1) with ascending slot numbering, starting with 1. The number of the ES1321.1 determined must be set in the "ES1321 Board ID" list field.

In the RTIO-Editor, up to twenty ES1321.1 PWM I/O Boards can be integrated per chassis.

This RTIO parameter cannot be edited during runtime of the model on the experimental target.

4.2 ES1321-In Subsystem

4.2.1 Globals (ES1321-In Subsystem)

The ES1321-In subsystem is used to configure the measure inputs of the ES1321.1 PWM I/O Board.

Fig. 4-3 shows the "Globals" tab of the ES1321-In Subsystem.

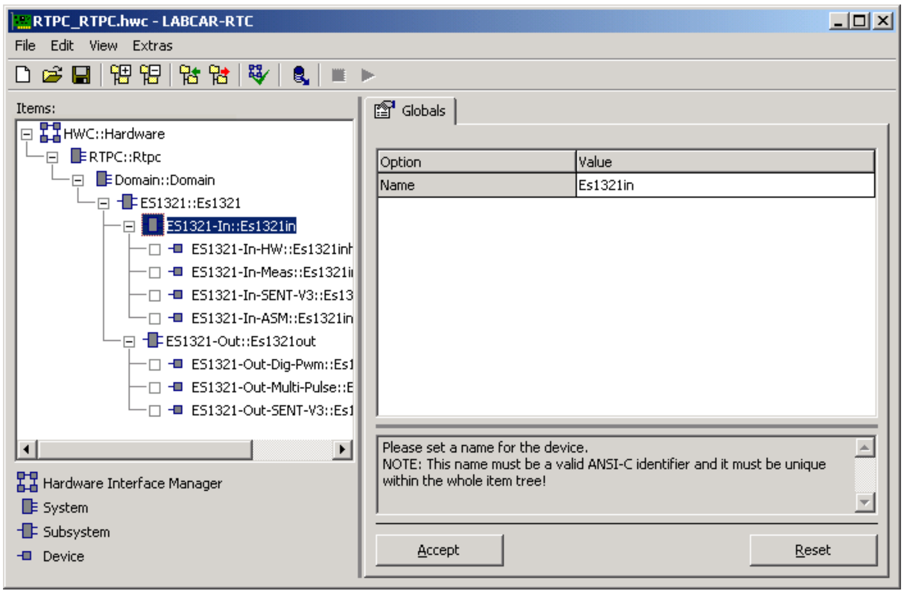


Fig. 4-3 The "Globals" Tab of the ES1321-In Subsystem

4.3 ES1321-In-HW Device)

4.3.1 Globals (ES1321-In-HW Device)

The ES1321-In-HW device is used to configure the channel LED and set the threshold levels of the input comparators.

The following figure shows the RTIO parameters of the "Globals" tab.

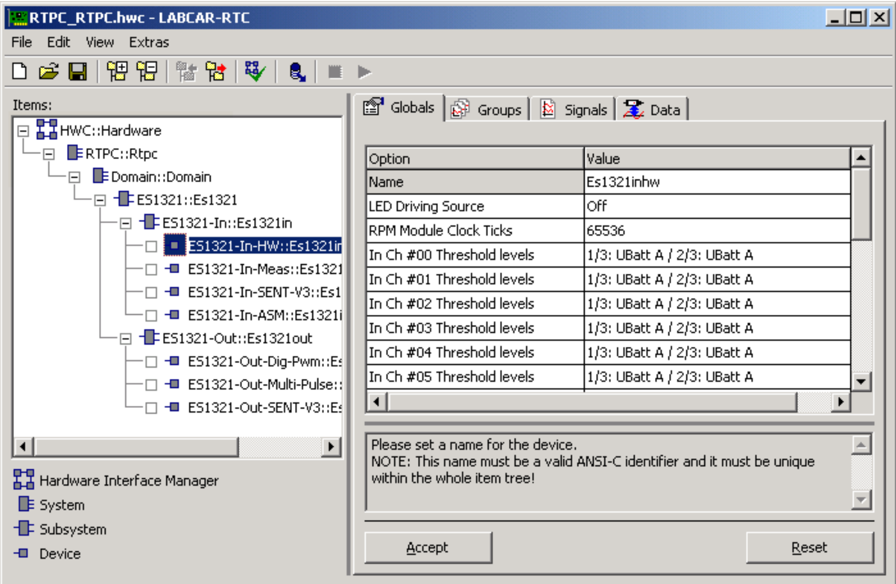


Fig. 4-4 The "Globals" Tab of the ES1321-In-HW Device

LED Driving Source

The channel LED of the ES1321.1 is addressed depending on this signal. If "Input Channel #xx" is selected, the LED is addressed by the signal level of channel "xx".

The list field can be edited online (i.e. during runtime of the model on the experimental target).

In Ch #xx Threshold Levels …

The following options are available for the threshold levels of the input comparator:

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

  With this option, the threshold levels for evaluating the input level are selected at 1/3 or 2/3 of the reference UBatt_A.

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

  With this option, the threshold levels for evaluating the input level are selected at 1/3 or 2/3 of the reference UBatt_B.

- userdefined

  Here, the user specifies 1/3 of the reference voltage – this value represents the lower threshold level. The upper threshold level is 2/3 of the reference voltage.

4.3.2 Groups (ES1321-In-HW Device)

The ES1321-In-HW device has a "LedDrv" signal group for setting the LED.

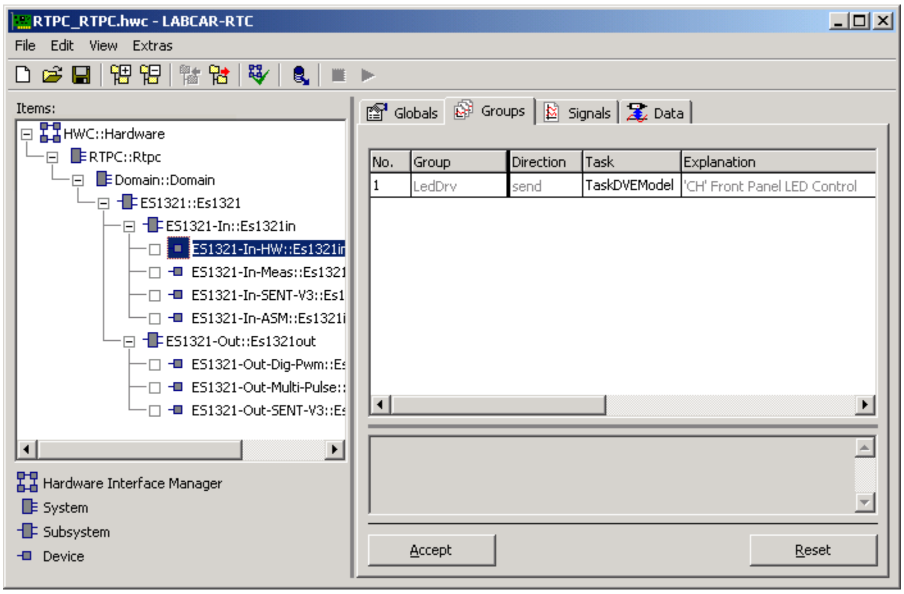


Fig. 4-5 The "Groups" Tab of the ES1321-In-HW Device

4.4 ES1321-In-Meas Device

4.4.1 Globals (ES1321-In-Meas Device)

The ES1321-In-Meas device is used to configure the measure functions on the 24 input channels.

No settings need to be made in the "Globals" tab.

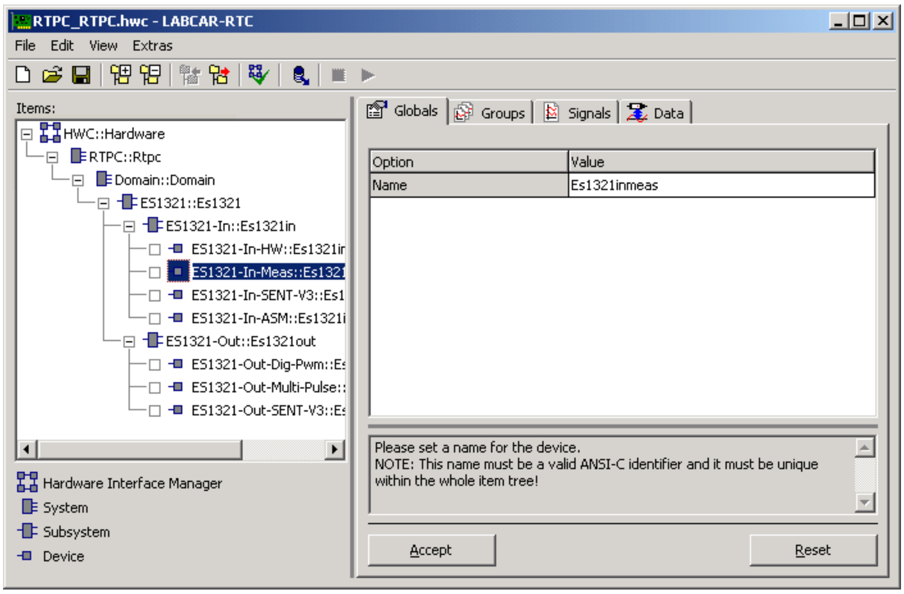


Fig. 4-6 The "Globals" Tab of the ES1321-In-Meas Device

4.4.2 Signals (ES1321-In-Meas Device)

The 24 measurements of an ES1321-In-Meas device are defined in the "Signals" tab. All option fields can be edited online (i.e. during runtime of the model on the experimental target).

The following figure shows the signals of the "Signals" tab.

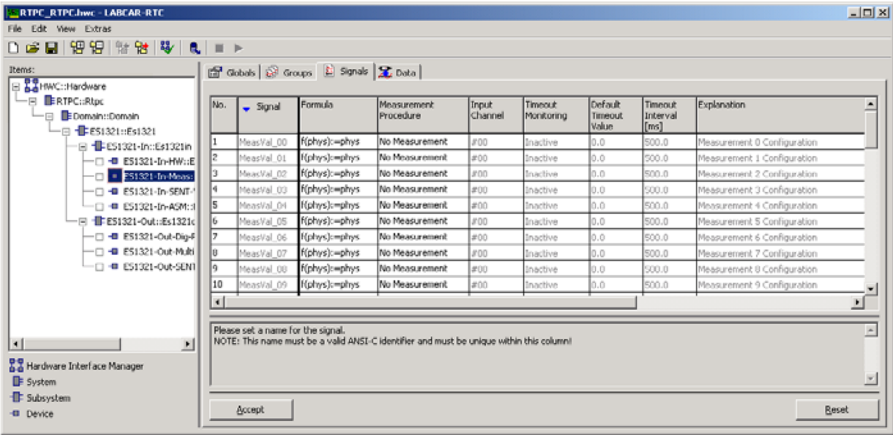


Fig. 4-7 The "Signals" Tab of the ES1321-In-Meas Device

Measurement Procedure

The measurement procedure is set in this list field. Refer to the section "The Measurement Procedures" on page 82 for a detailed description and configuration guidelines on the individual procedures.

Measuring takes place on the rising edge of the input signal.

Input Channel

This is where the input channel of the ES1321.1 to be used for measuring is set.

Timeout Monitoring

Definition of timeout monitoring for the relevant measurement. The following settings are possible:

- "Inactive": no timeout monitoring.
- "Intvl Predef": timeout monitoring in the intervals defined in the "Timeout Interval [ms]" option field. The timeout measure value is the value set in the "Default Timeout Value" option field.
- "Intvl InpDep": timeout monitoring in the intervals defined in the "Timeout Interval [ms]" option field. The timeout measure value depends on the level of the input signal.

Default Timeout Value

The value set in this numeric input field is issued as the timeout measure value if "Intvl Predef" timeout monitoring mode has been set.

The timeout condition is fulfilled if one of the four evaluated timestamps is outside the timeout value set.

Measure Methods and Timeout Handling

If the "Intvl InpDep" option was selected in "Timeout Monitoring", the following values are issued – depending on the measurement procedure set.

Measure Method	Value (Input Level Dependent)
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]	Time passed since last valid timestamp
Frequency --/-- [Hz]	Reciprocal of time passed since last timestamp
Duty Cycle L/(L+H)	1: input level is low 0: input level is high
Duty Cycle H/(L+H)	1: input level is high 0: input level is low
Level (Active High)	0
Level (Active Low)	0
Stepper A	0
Stepper B	0

4.5 The Measurement Procedures

Definitions

Active signal edge:

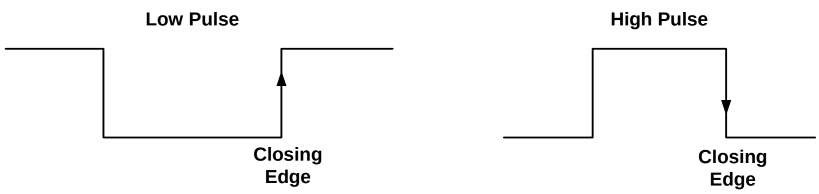
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.

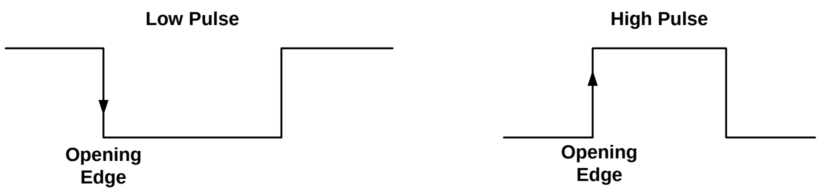


Inactive signal:

A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

Opening edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



The Basic Principle of Measure Value Calculation

The measuring and evaluation of measure data on the ES1321.1 PWM I/O Board basically takes place on the board.

The data is evaluated by the board's internal processor and transferred to the ES1130 and real-time PC using a dual-ported RAM via the VMEbus. In every measure method, a 32-bit value is transferred which may be further processed by the RTIO driver.

The measurements have a resolution of 20 ns.

4.5.1 Pulse-Width Measurements

The measurement procedures

- High Time [µs]
- Low Time [µs]

measure the high and low times of the pulses of a PWM input. The pulse width of the pulse immediately preceding the last time of data acquisition is measured. Fig. 4-8 shows an example of a high-time measurement; the measured pulse is shown in bold print.

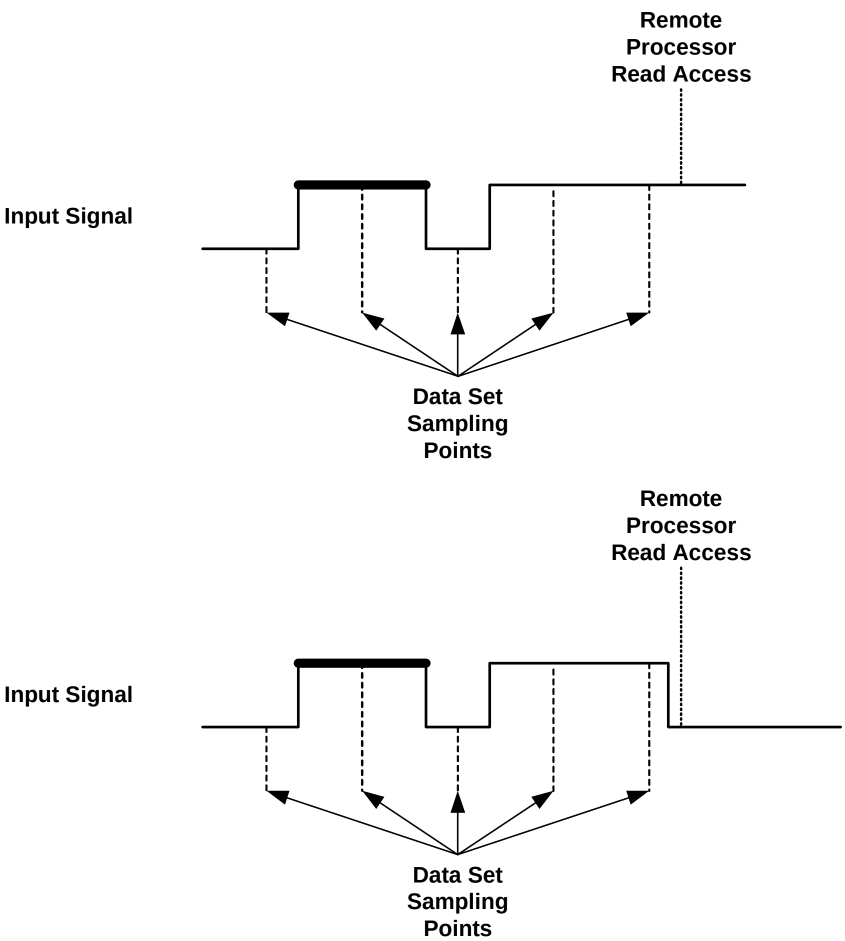


Fig. 4-8 High-Time Measurement (the Measured Pulse is shown in Bold Print)

A timeout occurs when at least one of the four edges is older than the current timestamp minus the timeout time.

Timeout handling is the same for all measurement procedures.

4.5.2 Additive Pulse-Width Measurements

If the read process for the "MeasVal" signal group is activated again (periodically in special cases), the measurement procedures

- Additive High Time [µs]
- Additive Low Time [µs]

return the time in which the signal to be measured has assumed an active level between two consecutive activations of the read process.

Fig. 4-9 shows the measure value calculation using the example of an additive high-time measurement. The total of the line segments marked in bold print is the additive high time which is returned on the (n+1)th activation of the read process.

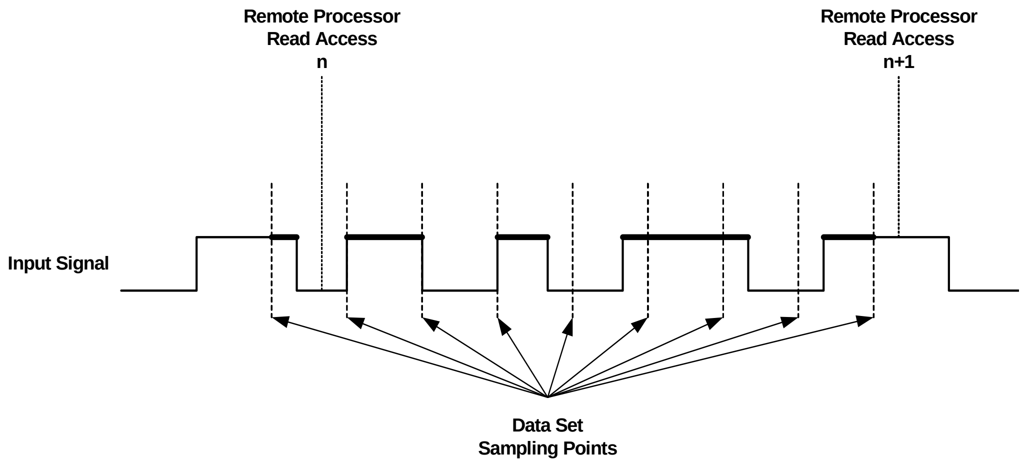


Fig. 4-9 Additive High-Time Measurement: Illustration of Measure Value Calculation

A timeout occurs when at least one of the four edges is older than the current timestamp minus the timeout time.

4.5.3 Pulse and Edge Count

If the read process for the "MeasVal" signal group is activated again (periodically in special cases), the measurement procedures

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

return the number of active pulse or signal edges between two consecutive activations of the read process. Fig. 4-10 shows the method of pulse counting in detail using the example of a high-pulse count.

All pulses whose closing edge is in the period between the activations of the read process are counted. If you want to specify this precisely, the period of time is defined by the times of data acquisition immediately preceding the activations. As far as the edge count is concerned, the evaluation period is exactly the same (Fig. 4-11 on page 86).

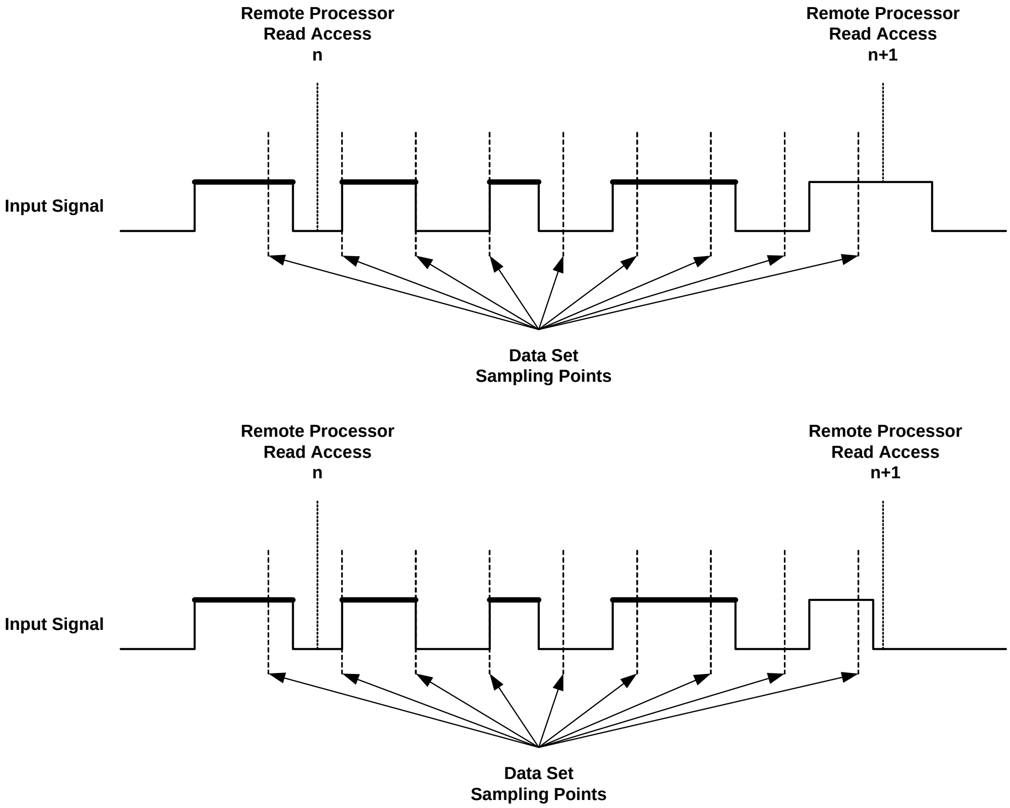


Fig. 4-10 High-Pulse Count: The Number of Pulses Marked in Bold Print is Returned as a Measure Value on the (n+1)th Time of Reading

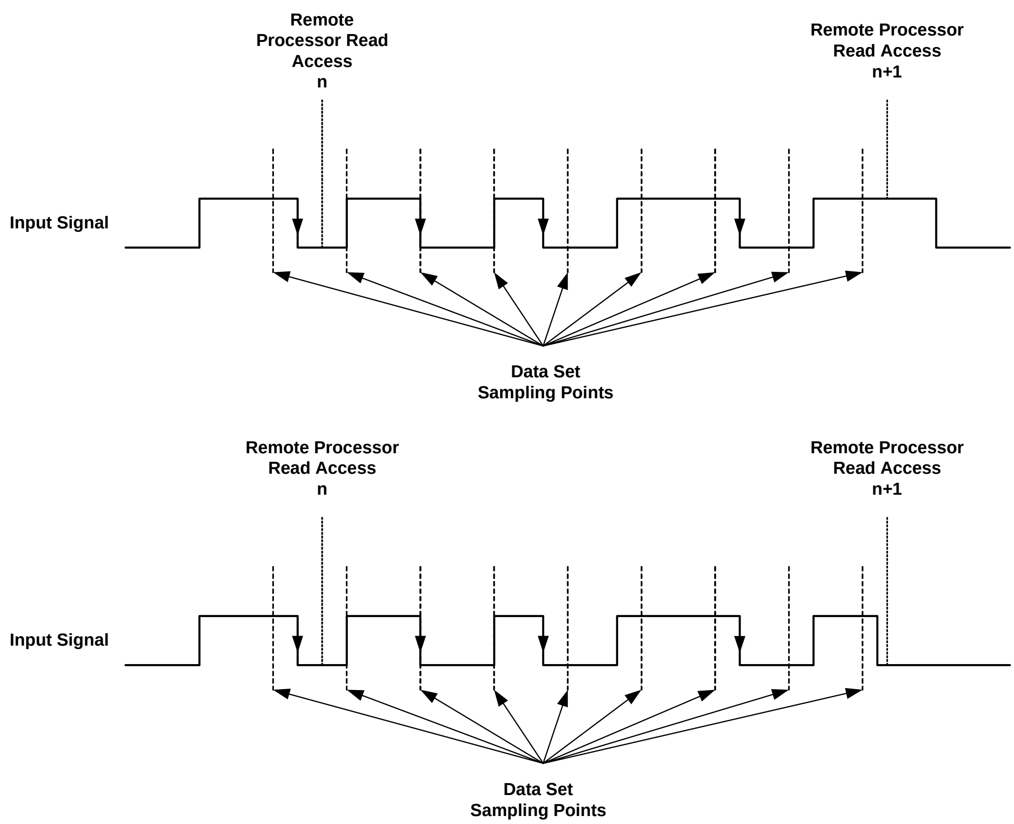


Fig. 4-11 Counting Falling Edges: The Number of Marked Falling Edges is returned as a Measure Value on the (n+1)th Time of Reading

4.5.4 Frequency and Cycle Time Measuring

The measurement procedures

- Cycle Time
- Frequency [Hz]

return the cycle time or the frequency of the signal at a PWM input measured on rising or falling edges. The active signal edge immediately preceding the last[1] time of data acquisition and its corresponding period are evaluated.

1. The time of data acquisition immediately preceding the RTIO driver's read access of the data set

Fig. 4-12 shows the evaluated period using the example of a frequency or cycle measurement on rising edges.

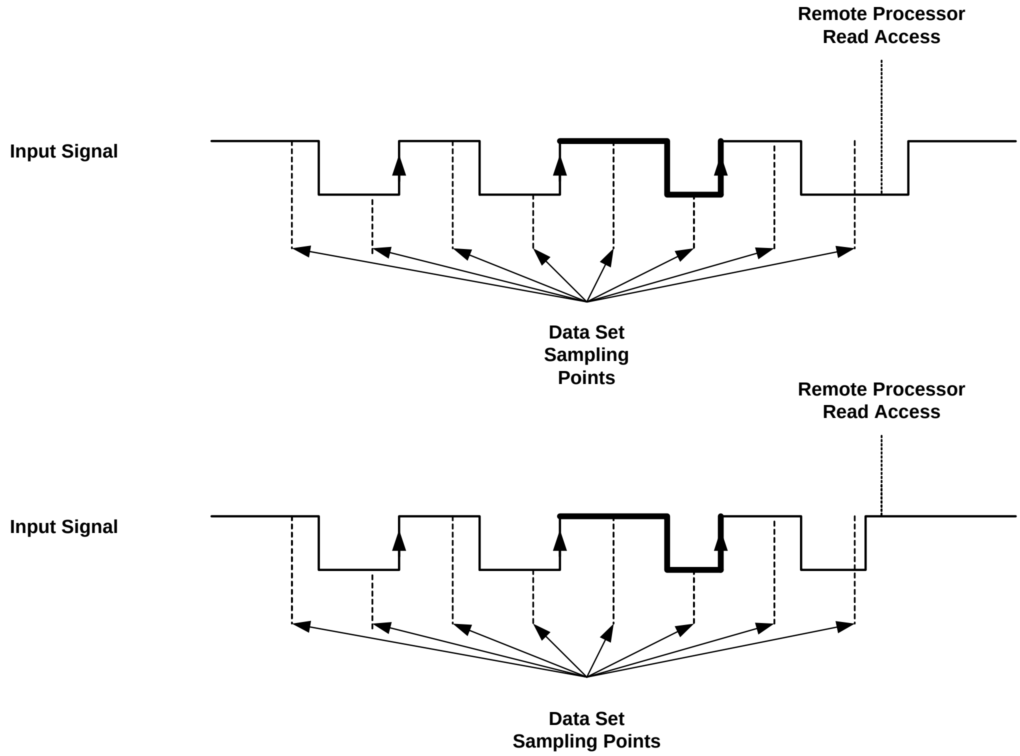


Fig. 4-12 Evaluated Period (in Bold) in a Frequency or Cycle Measurement on Rising Edges

4.5.5 Duty Cycle Measurements

The following duty cycle measurements are available with the ES1321.1 PWM I/O Board:

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

In these measurements, a pulse duration is set in relation to the signal cycle time. The user specifies the edge on which the measure value is calculated when selecting the measurement procedure.

As far as the selection of the period of the input signal is concerned, on the basis of which the duty cycle calculation is executed, exactly the same as was described for frequency and cycle time measurements in section 4.5.4 on page 86 applies.

Note

*In order to detect a **duty cycle of 0 or 1** the measurement must be combined with timeout monitoring "Intvl InpDep":*

- *For PWM signals having a **constant frequency** and a variable duty cycle, "Timeout Interval [ms]" must be set to a time greater than or equal to the cycle time of the PWM input signal.*
- *For PWM signals having a **variable frequency**, "Timeout Interval [ms]" must be set to a time greater than or equal to the maximum cycle time of the PWM input signal.*

At the end of the timeout interval the measurement is set to 0 or 1 according to the measurement method used and the input level.

*If the **frequency of the PWM input signal is measured simultaneously**, the timeout measurement methods "Intvl Predef" and "Intvl InpDep" can be used in order to set the measurement to the desired value:*

- *Timeout monitoring "Intvl Predef" returns a predefined frequency.*
- *Timeout monitoring "Intvl InpDep" returns the reciprocal value of the time having passed since the last valid time stamp.*

4.5.6 Level Measurements

The measurement procedures

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

return the level of a PWM input in the form of active/inactive information. "0" means the signal is inactive; "1" means the signal is active.

4.5.7 Position Tracing on Two-Phase Stepper Motors

Functional Description

The ES1321.1 firmware makes position tracing possible and thus basically the measurement of rotor position and speed of two-phase stepper motors. Two hardware channels and one software or measurement channel are necessary for this purpose.

As measurement procedures, select "Stepper A" or "Stepper B" – the hardware channels are assigned as follows:

software channel 0 → phase A → hardware channel 0
phase B → hardware channel 1
software channel 1 → phase A → hardware channel 2
phase B → hardware channel 3
software channel 2 → phase A → hardware channel 4
phase B → hardware channel 5

...

Note

Two adjacent channels always have to be made into a pair ((channel 0 and channel 1), (channel 2 and channel 3) etc.).
In the first pair, channel 0 corresponds to "Phase A" and channel 1 to "Phase B". The assignment of the other channels follows the same pattern.

In the "Reference Channel" field of one measurement channel, there must be a reference to the number of the other measurement channel involved in the measurement.

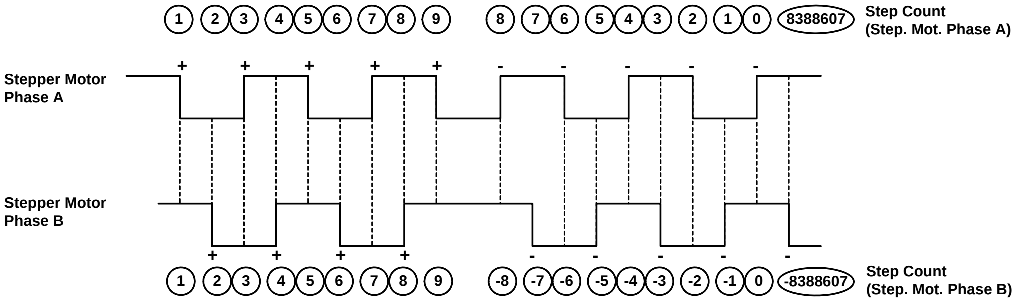


Fig. 4-13 How Position Tracing on Two-Phase Stepper Motors Works

Note

Position tracing takes place modulo 2^{29}. Incrementing at a position level of (2^{29}-1) generates a position counter value of 0. Decrementing with a position level of 0 generates a position counter value which in terms of the amount is (2^{29}-1).

4.6 ES1321-In-SENT Device

4.6.1 Globals (ES1321-In-SENT Device)

This RTIO element makes it possible to receive signals in accordance with the SENT data protocol. Two ES1321-In-SENT devices can be added, i.e. in terms of the hardware, there can be two receiving units working parallel to one another.

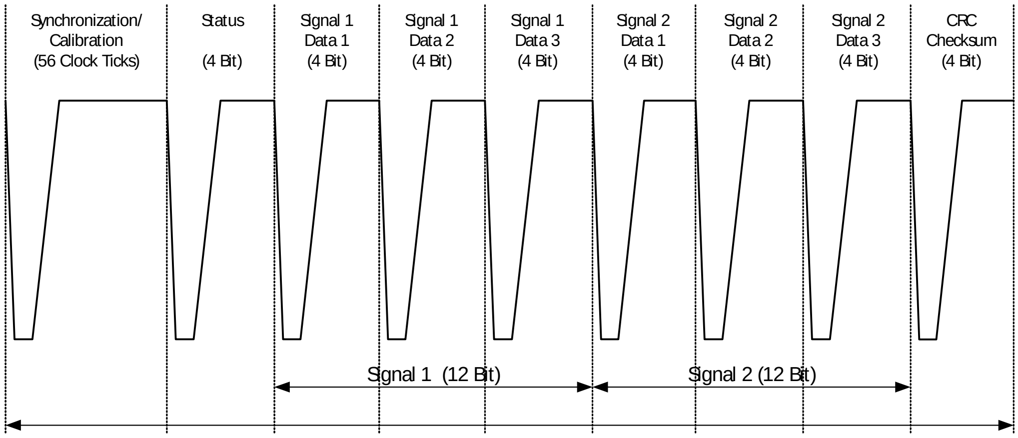
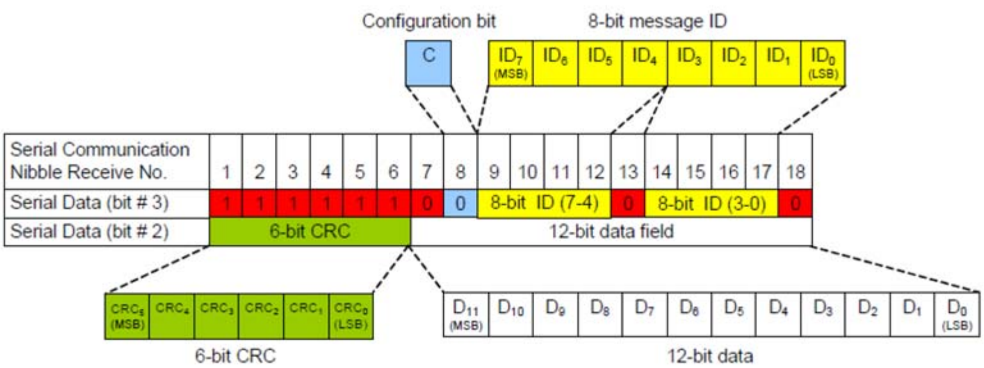


Fig. 4-14 Format of a Signal in acc. with the SENT Data Protocol

Serial Messages Format

Bit 2 and bit 3 of the status nibble are used for the Serial Message Channel (SMC).



For 16 or 18 frames, the data in bit 2 and bit 3 of the status nibble is collected and stored in the memory of the ES1321.1.

The evaluation of the data is done in LABCAR-RTPC using a special C code module. This module is available as open source code and can be found on the LABCAR-OPERATOR installation medium in the `\MISC` folder.

The following figure shows the "Globals" tab of the ES1321-In-SENT device.

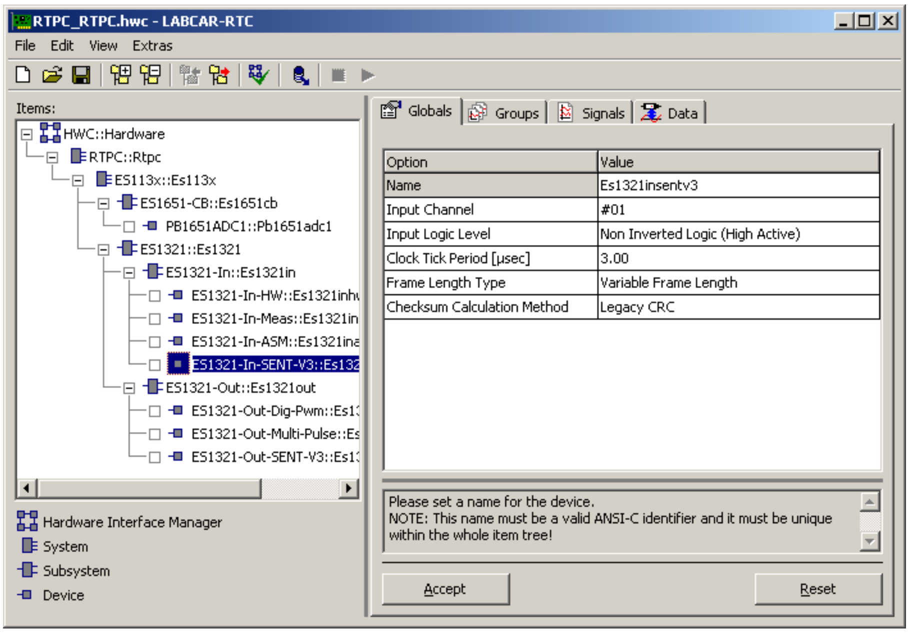


Fig. 4-15 The "Globals" Tab of the ES1321-In-SENT Device

Input Channel

Any input channel can be used – the level and threshold settings of this channel in the ES1321-In-HW device will be used. Please ensure that they correspond to the level specification of the SENT transmitter (typically: inactive = GND, active = 5 Volt).

Input Logic Level

This setting determines whether the level of the SENT signal is output inverted or non-inverted.

- Non inverted Logic (High Active)

  Fig. 4-16 shows a signal with non-inverted logic.

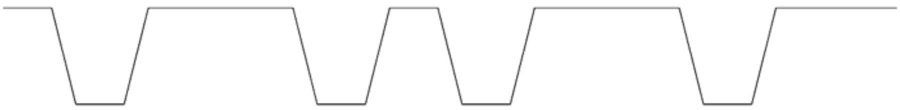


Fig. 4-16 Non-Inverted Logic

- Inverted Logic (Low Active)

  Fig. 4-17 shows a signal with inverted logic.

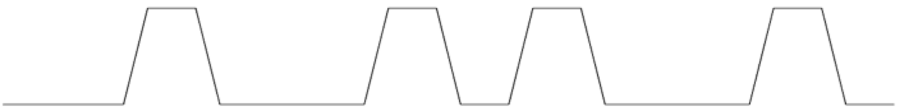


Fig. 4-17 Inverted Logic

Clock Tick Period [µsec]

Length (in µs) of a clock tick – the length of the "calibration/synchronization pulse" and of the "nibble pulses" are derived from this value.

Frame Length Type

The transmission mode of the SENT frames.

- Variable Frame Length

  A SENT frame is expected to follow the preceding frame immediately – there is no "pause" pulse.

- Constant Frame Length

  A "pause" pulse is expected in between two SENT frames. The length of the "pause" pulse is such that the whole frame (including "pause") is of constant length.

Checksum Calculation Method

The calculation method for the 4-bit checksum in the ES1321.1.

- None

  Checksum calculation is disabled.

- Recommended CRC

  Checksum calculation according to SENT specification Jan 2010.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.

- Legacy CRC

  Checksum calculation according to SENT specification April 2007 and Feb 2008.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.

4.6.2 Signals (ES1321-In-SENT Device)

The following figure shows the "Signals" tab of the ES1321-In-SENT device.

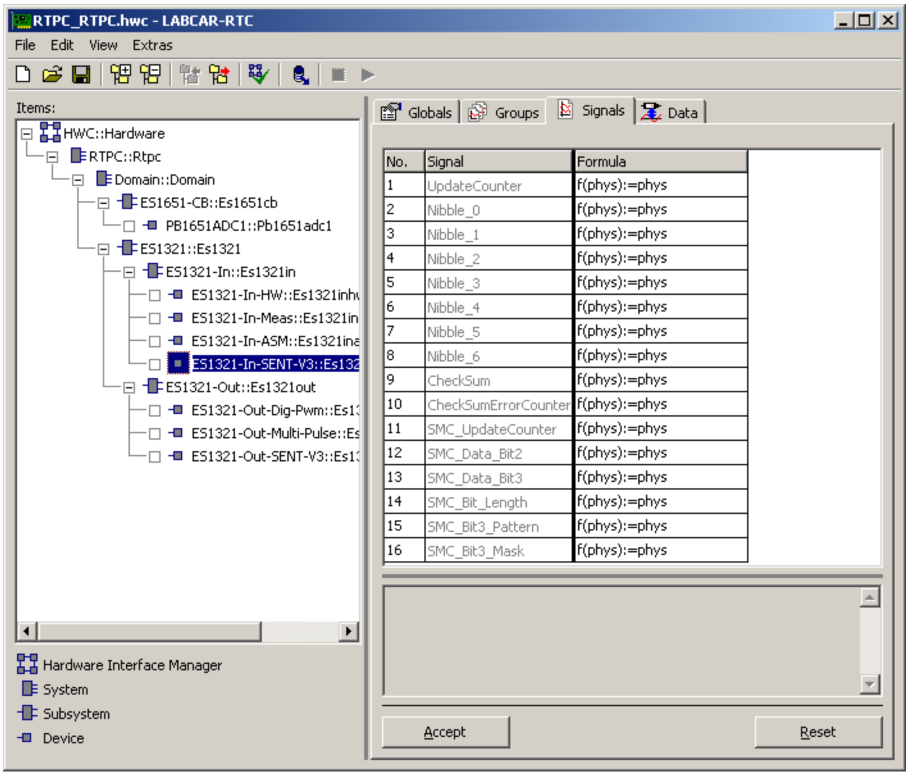


Fig. 4-18 The "Signals" Tab of the ES1321-In-SENT Device

4.6.3 Data (ES1321-In-SENT Device)

The following figure shows the "Data" tab of the ES1321-In-SENT device.

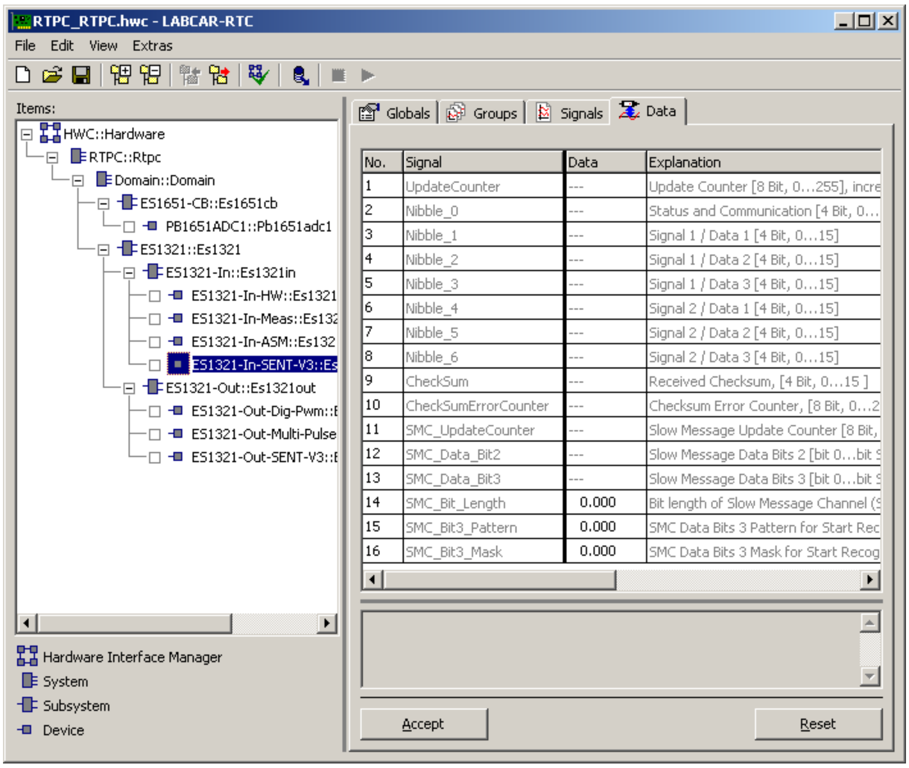


Fig. 4-19 The "Data" Tab of the ES1321-In-SENT Device

The ES1321-In-SENT device implements a receive signal group "SentIn". This supplies the non-interpreted data of the last SENT data word received completely and status information. The interpretation of the data depends on the sender's protocol type.

UpdateCounter

Counter which increases when new data are available – the value range is 0…255.

Nibble_0 ... Nibble_6

Values of the 7 nibbles (1 status nibble and 2x3 data nibbles) of the last SENT data word received – the value range is 0…15.

CheckSum

Checksum transmitted by the SENT frame.

CheckSumErrorCounter

Counter which increases when a SENT message was completely received in the ES1321.1 but the checksum was wrong.

SMC_UpdateCounter

8-bit counter which increases when the ES1321.1 has received a complete SMC message and is ready for evaluation at the signals "SMC_Data_Bit2" and "SMC_Data_Bit3".

SMC_Data_Bit2

Data word containing the bit-2 data stream received (see SAE J2716 version Jan 2010, Ch.5.2.4). The higher bit positions contain data which was received later.

SMC_Data_Bit3

Data word containing the bit-3 data stream received (see SAE J2716 version Jan 2010, Ch.5.2.4). The higher bit positions contain data which was received later.

SMC_Bit_Length

Input signal controlling the SMC receiving unit of the ES1321.1.

"0" disables the reception, a value > 0 (… 31) enables it and defines the number of bits for a SMC cycle. According to the SENT specification, valid values are "16 " (short serial message) or "18 "(enhanced serial message).

SMC_Bit3_Pattern

Input signal controlling the SMC receiving unit of the ES1321.1.

It defines a bit pattern for bit 3 checking the validity of the SMC message.

A message is recognised as valid, when

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

I.e., "SMC_Bit3_Pattern" defines the value of the relevant bits.

SMC_Bit3_Mask

Input signal controlling the SMC receiving unit of the ES1321.1.

It defines a bit mask for bit 3 checking the validity of the SMC message.

A message is recognised as valid, when

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

I.e., "SMC_Bit3_Mask" marks the relevant bit positions with a "1".

The ES1321.1 and the RTIO only provides generic functionality for the reception of SMC messages. Control, decoding of the data and verification of checksums have to take place outside the RTIO.

A C code module template "SentInSMC_Control" with basic functionality is provided. For that purpose all SMC_* signals of the RTIO are connected to the corresponding ports of the C code module.

However, for a specific application the C code module must be adapted by

- defining "SMC format mode" according to the SENT spezification
- converting the data values of the required Msg-IDs into physical values
- providing these values at the output ports

According to SAE J2716 Jan 2010 the following values are defined:

- 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)

This RTIO element makes it possible to execute certain angle-synchronous measure functions with the ES1321.1 (see "Measurement Procedure" on page 97). Basically, the angle of a specific edge which appears within an angle window is determined.

Any input channel can be used as an input. This has to be configured accordingly in the ES1321-In-HW device.

For each ES1321-In-ASM device, four measure channels of the same measurement procedure, but with different angle windows, are available. Two ES1321-In-ASM devices can be added, i.e. resources are available for a total of eight measure channels.

The following figure shows the "Globals" tab of the ES1321-In-ASM device.

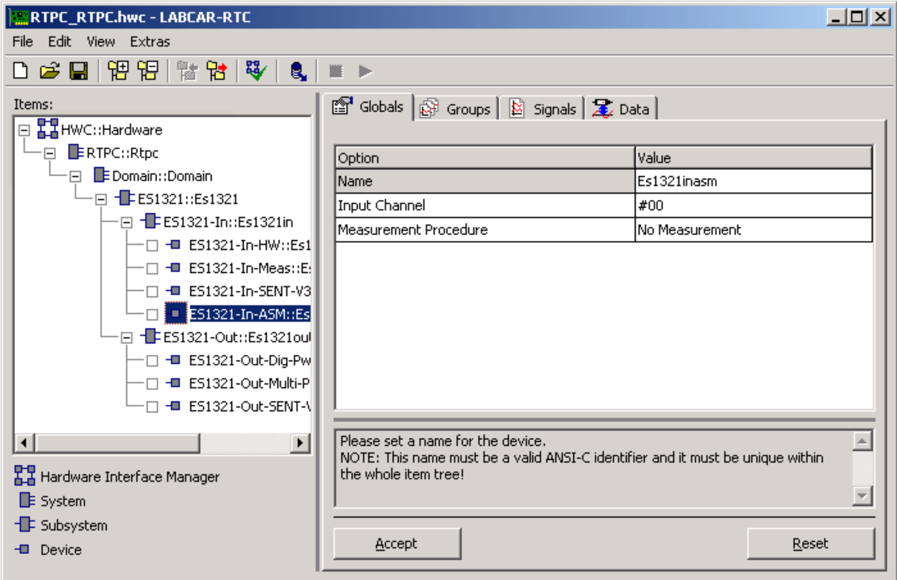


Fig. 4-20 The "Globals" Tab of the ES1321-In-ASM Device

The following parameters can be set in this tab:

Input Channel

The input channel to be used is selected here – the level and threshold value settings of the ES1321-In-HW device are used. All four software measure channels of this ES1321-In-ASM device refer to this hardware channel.

Measurement Procedure

This is where the measurement procedure used is selected for all four software measure channels of the RTIO element – in total, the following four measure methods are available:

- 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]

The angle value of the first or last rising or falling edge (respectively) which has occurred within an angle window (defined by LWL and UWL, see Fig. 4-21 on page 98) is measured in each case. The measure value results from a reference angle value minus this angle. Edges before the reference angle return a positive result, edges after the reference angle return a negative result.

If no corresponding edge occurs in the angle window, no result is transferred and a timeout is detected.

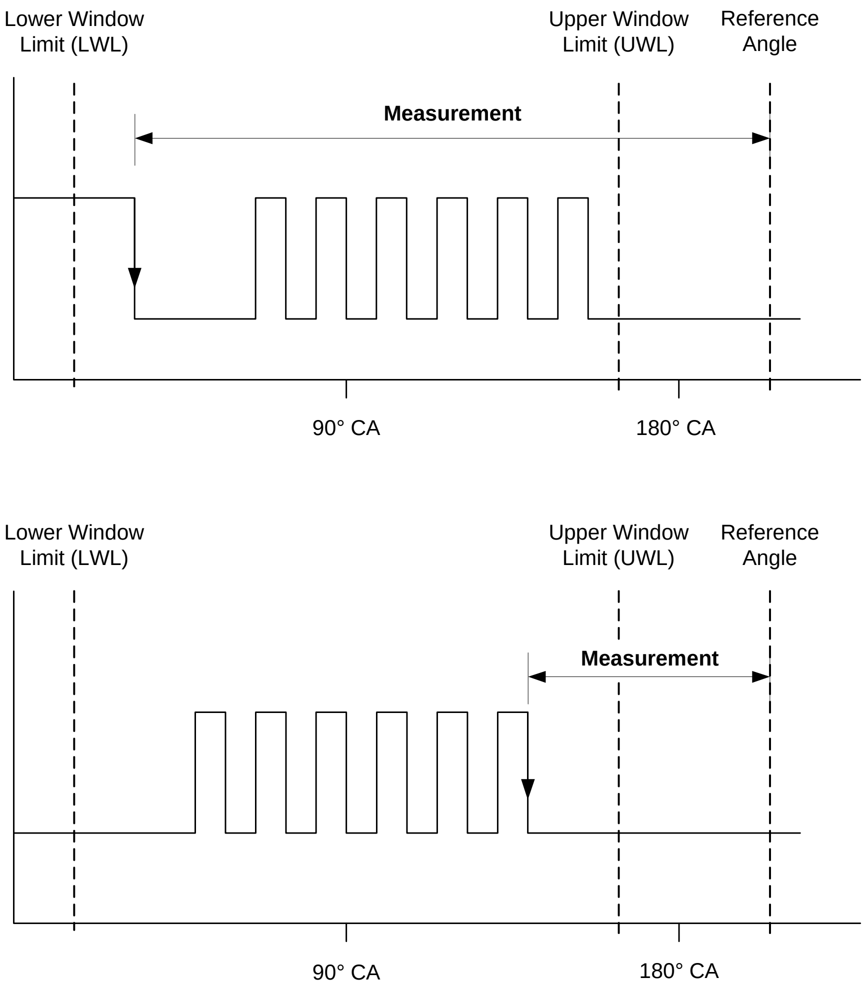


Fig. 4-21 Measure Values with "Angle of First Falling Edge of Pulse Sequence" (top) and "Angle of Last Falling Edge of Pulse Sequence" (bottom)

4.7.2 Signals (ES1321-In-ASM Device)

The ES1321-In-ASM device implements a "MeasVal" receive signal group.

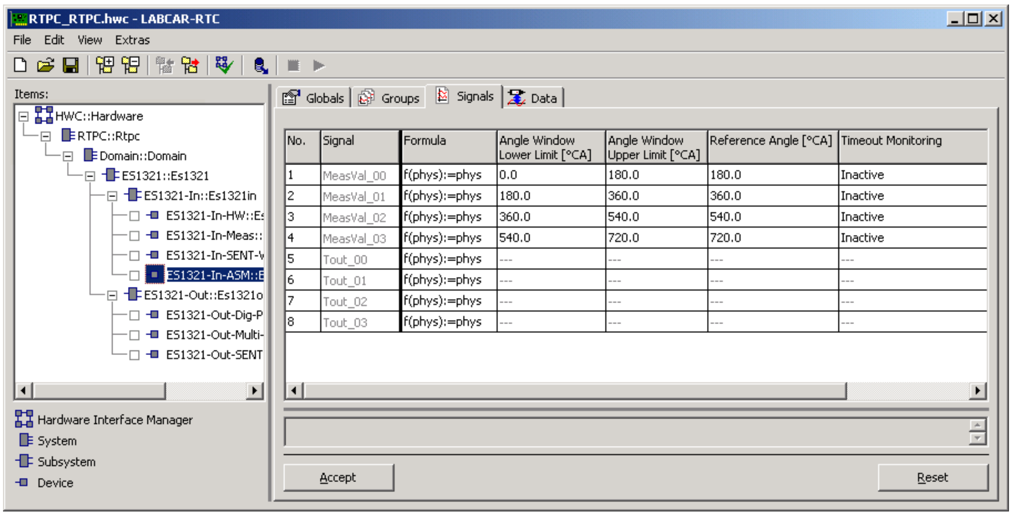


Fig. 4-22 The "Signals" Tab of the ES1321-In-ASM Device

The following signals are created:

MeasVal_xx [xx = 00…03]

The result values for software measure channels 0…3 (value range -720.0…720.0 °CA, -8888.0 if channel not used)

Tout_xx [xx= 00…03]

Timeout signals for the software measure channels 0…3

- 0 = no timeout
- 1 = timeout detected
- 2 = timeout not configured
- 3 = angle interface not synchronous

In the "Signals" tab, the following settings can be made for each individual measure channel.

Angle Window Lower Limit

The starting angle of the measure window (corresponds to "LWL" in Fig. 4-21 on page 98).

Angle Window Upper Limit

The concluding angle of the window (corresponds to "UWL" in Fig. 4-21 on page 98)

Reference Angle

The reference angle for calculating the measure value.

Timeout Monitoring

The operating mode for a timeout.

- Inactive

  No timeout detection; the previous value is transferred

- Intvl Predef

  The substitute value for a timeout is transferred

Default Timeout Value

The substitute value for a timeout.

4.8 ES1321-Out Subsystem

4.8.1 Globals (ES1321-Out Subsystem)

The ES1321-Out subsystem is used to configure and address the 16 output channels at which digital signals, pulse-width modulated signals, pulse sequences and SENT signals can be issued. It is also possible to switch the outputs against two different positive reference voltages and against ground.

An ES1321-Out-Dig-Pwm, two ES1321-Out-Multi-Pulse and up to four ES1321-Out-SENT devices can be added under this subsystem.

No settings need to be made in the "Globals" tab.

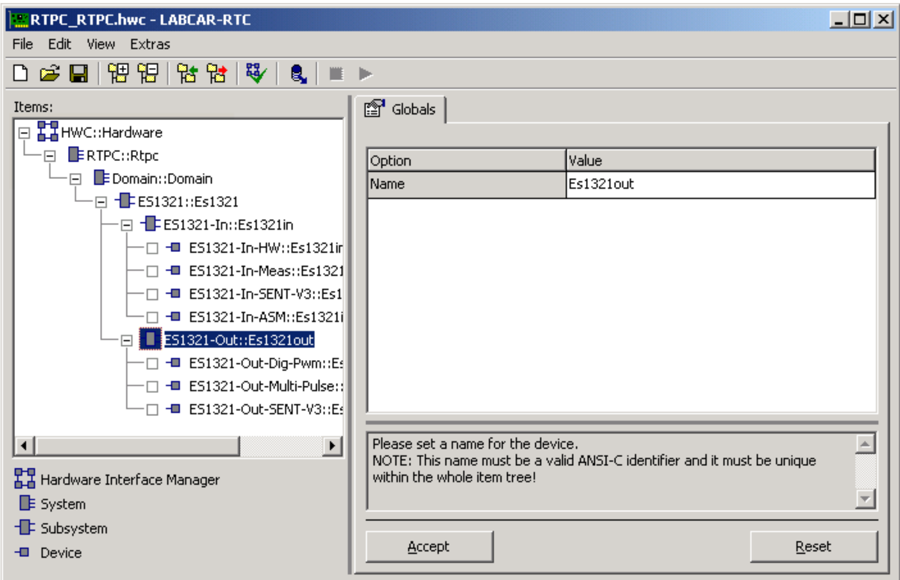


Fig. 4-23 The "Globals" Tab of the ES1321-Out Subsystem

4.9 ES1321-Out-Dig-Pwm Device

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

Fig. 4-24 shows the "Signals" tab of the ES1321-Out-Dig-Pwm device.

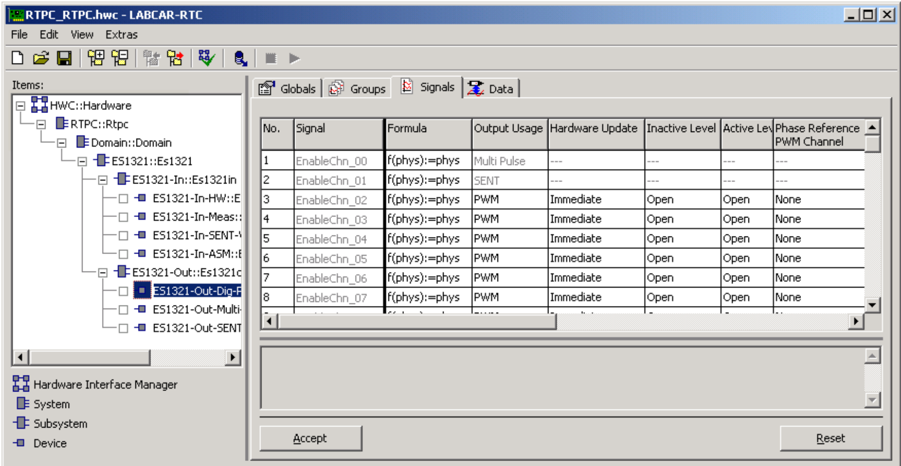


Fig. 4-24 The "Signals" Tab of the ES1321-Out-Dig-Pwm Device

You can define the following options in this tab:

Output Usage

In this view, you can choose between PWM and Digital Out.

If an ES1321-Out-Multi-Pulse or an ES1321-Out-SENT device has already been added, the outputs used for this are grayed out.

- PWM

  Generation of a pulse-width modulated signal – described by frequency, duty cycle and phase relation to another PWM output

- Digital Out

  Signal levels are generated in this configuration of the output channel

Hardware Update

This list field is used to set when value changes made by the model or the user to the "Frequency_x" and "DutyCycle_x" RTIO signals (x = 0...15) are accepted by the ES1321.1 hardware.

- Immediate

  A new value is immediately visible at the output, i.e. the current pulse is cancelled and started with new values.

- Cycle End

  A new value only becomes visible at the output when the complete pulse is ended.

- RTIO Controlled

  A new value becomes visible at the output after a change from "0 → 1" of the "SyncSgl" signal.

Inactive Level

The inactive state of the PWM signal is set in this list field. The inactive state of a PWM signal is defined as follows: if a duty cycle of 60% is set for a PWM signal, it assumes inactive signal state for 40% of the period duration.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Active Level

The active state of the PWM signal is set in this list field.

The active state of a PWM signal is defined as follows: if a duty cycle of 60% is set for a PWM signal, it assumes active signal state for 60% of the period duration.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level using an external pull-up- or pull-down resistor.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Phase Reference PWM Channel

The ES1321.1 makes it possible to define the setting of phase shifts between output channels with the same PWM frequency. One of the PWM outputs must be specified as a phase reference channel for this purpose. T

he number of the reference channel is then to be entered in the "Phase Reference PWM Channel" list field of the remaining outputs which have defined phase relations with this reference channel.

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

The following figure shows the "Data" tab of the ES1321-Out-Dig-Pwm device.

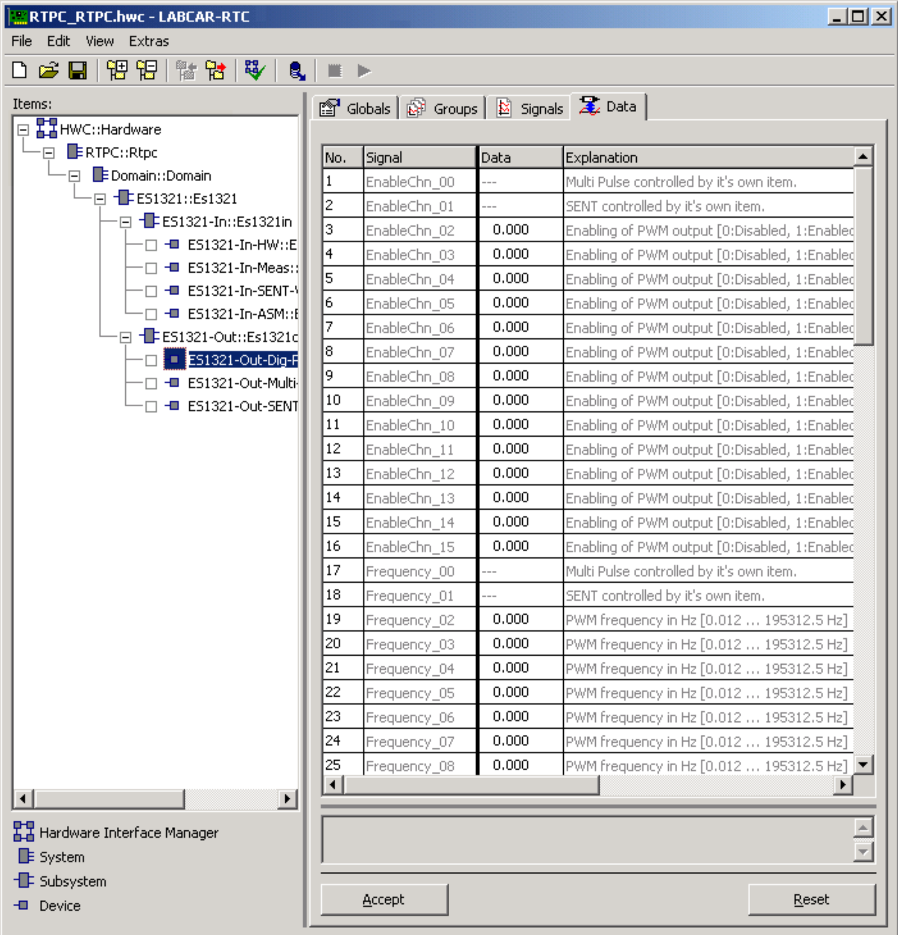


Fig. 4-25 The "Data" Tab of the ES1321-Out-Dig-Pwm Device

The "Data" tab contains the following signals:

- EnableChn_00...EnableChn_15

  Enables or disables signal output

  – = 0: output is high-impedance, signal output is inactive

  – = 1: signal output is active

- Frequency_00..Frequency_15

  Frequency (in Hz) of the relevant PWM signal

- DutyCycle_00..DutyCycle_15

  The duty cycle of the relevant assigned PWM output is set with the "DutyCycle_x" signals. Duty cycles are defined with values between 0.0 and 1.0. 0.0 corresponds to a duty cycle of 0%, 1.0 to a duty cycle of 100%.

- Phase

  Phase shift (in °) with relation to the reference channel. A value of 360° corresponds to a phase shift of a complete pulse.

- SyncSgl

  The ES1321.1 makes it possible to control the time when frequency and phase data transmitted by RTIO or by the simulation model to the board is accepted by the hardware. The time is controlled using the "Hardware Update" option in the "Signals" tab (see "Hardware Update" on page 101).

  If the option "RTIO Controlled" is set in this list field for one or more channels, the frequency and duty cycle data transmitted to these channels is only accepted by the hardware if a transition from 0 to 1 is detected on the "SyncSgl" signal.

- ChnState_00...ChnState_15

  Each output channels has a 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver:

  – Bit 0: Output Level
  – Bit 1: Lowside Switch Overcurrent
  – Bit 2: Highside Switch Overcurrent
  – Bit 3: Error Condition Detected

4.10 ES1321-Out-Multi-Pulse Device

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

A multi-pulse sequence can consist of up to eight pulses with each one being defined by a frequency and a duty cycle.

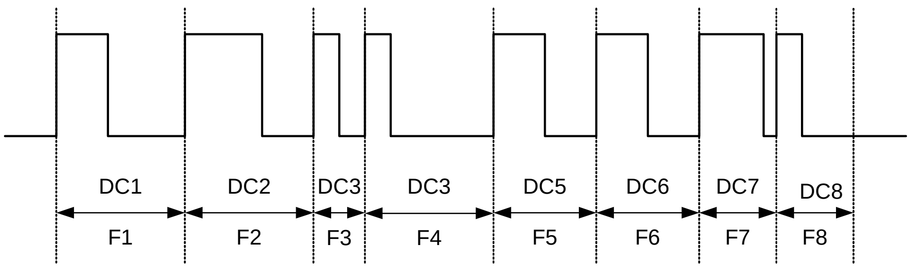


Fig. 4-26 A Sequence of Eight Pulses (Fx = Frequency of Pulse x, DCx = Duty Cycle of Pulse x)

The following figure shows the "Globals" tab of the ES1321-Multi-Pulse device.

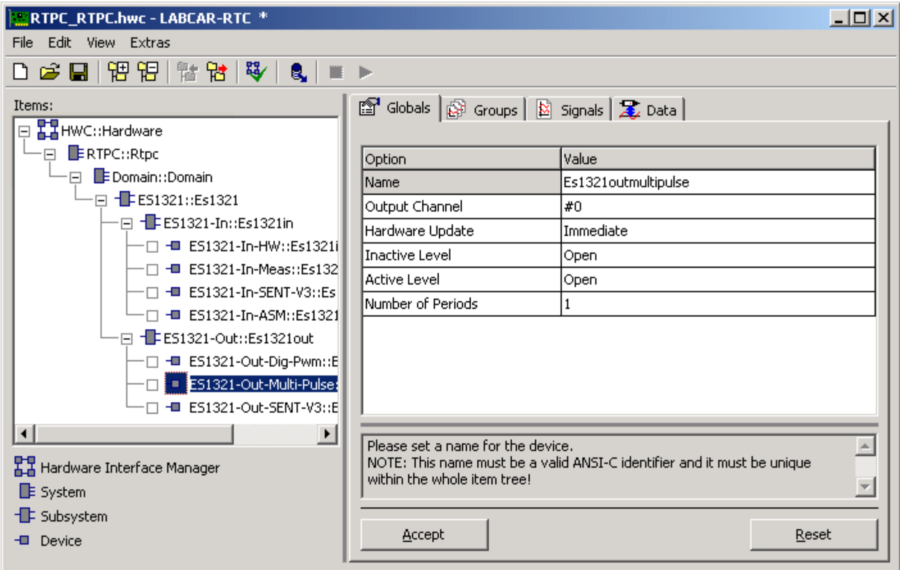


Fig. 4-27 The "Globals" Tab of the ES1321-Multi-Pulse Device

The meaning of the available options is described below.

Output Channel

The hardware channel of the ES1321.1 to be used as a multi-pulse output.

Hardware Update

This list field is used to set when changes made by the model or the user to the pulse package are accepted by the ES1321.1 hardware.

- Immediate

  A new value is immediately visible at the output, i.e. the current pulse is cancelled and started with new values.

- Cycle End

  A new value only becomes visible at the output when the complete pulse is ended.

- RTIO Controlled

  A new value becomes visible at the output after a change from "0 → 1" of the "SyncSgl" signal.

Inactive Level

The inactive state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.

- Ground

  The PWM output is at ground.

- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Active Level

The active state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Number of Periods

Number of pulses of the sequence (1..8)

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

The following figure shows the signals of the "Data" tab

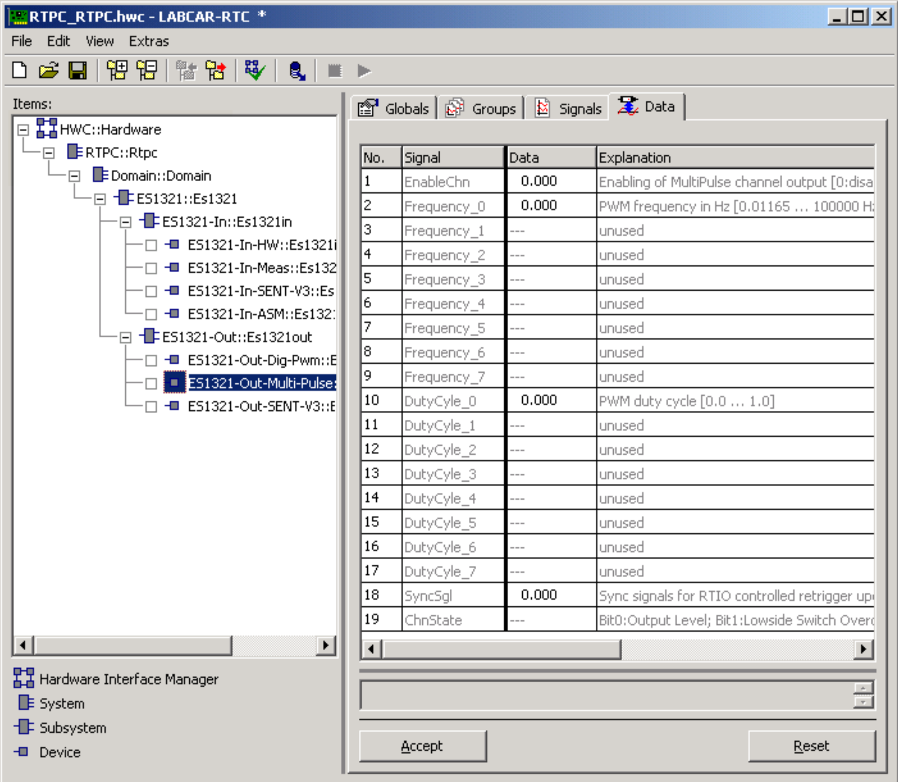


Fig. 4-28 The "Data" Tab of the ES1321-Multi-Pulse Device

The "Data" tab contains the following signals:

- EnableChn

  Enables or disables the multi-pulse channel

  – = 0: output is high-impedance, signal output is inactive

  – = 1: signal output is active

- Frequency_0 .. Frequency_7

  Frequency (in Hz) of the relevant pulse

- DutyCycle_0 .. DutyCycle_7

  The duty cycle of the relevant pulse is set with the "DutyCycle_x" signals. Duty cycles are defined with values between 0.0 and 1.0. 0.0 corresponds to a duty cycle of 0%, 1.0 to a duty cycle of 100%.

- SyncSgl

  The ES1321.1 makes it possible to control the time when pulse package data transmitted by RTIO or by the simulation model to the board is accepted by the hardware. The time is controlled using the "Hardware Update" option in the "Signals" tab (see "Hardware Update" on page 105). If the "RTIO Controlled" option is set in this list field for one or more channels, the pulse package data transmitted to these channels is only accepted by the hardware if a transition from 0 to 1 is detected on the "SyncSgl" signal.

- ChnState

  The output channel has 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver:
  - Bit 0: Output Level
  - Bit 1: Lowside Switch Overcurrent
  - Bit 2: Highside Switch Overcurrent
  - Bit 3: Error Condition Detected

4.11 ES1321-Out-SENT Device

4.11.1 Globals (ES1321-Out-SENT Device)

The following figure shows the "Globals" tab of the ES1321-Out-SENT device.

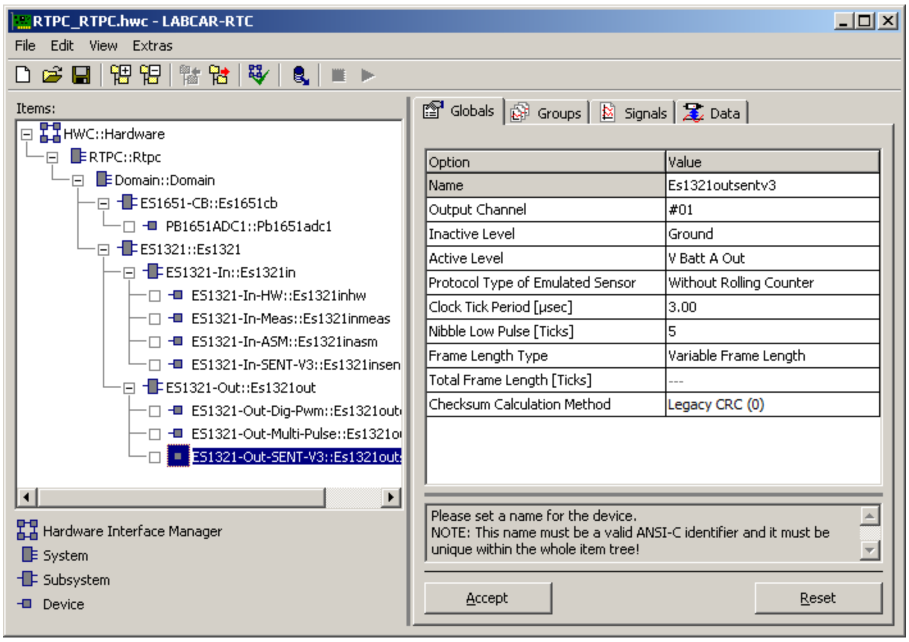


Fig. 4-29 The "Globals" Tab of the ES1321-Out-SENT Device

The following options are set in this tab:

Output Channel

Selection of the hardware channel which is to be used as the SENT transmitter.

Inactive Level

The inactive state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance.
- Ground

  The PWM output is at ground.

- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Note

For a non-inverted output signal, select "Ground" for the inactive level.

Active Level

The active state (of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Note

For a non-inverted output signal, select "V Batt A" or "V Batt B" for the active level.

Protocol Type of Emulated Sensor

Selection of the protocol used in acc. with the SENT specification.

The following options are available:

- Without Rolling Counter

  The values of "Nibble_4" and "Nibble_5" (as shown in the "Data" tab) are considered.
- With Rolling Counter Nibble 4/5

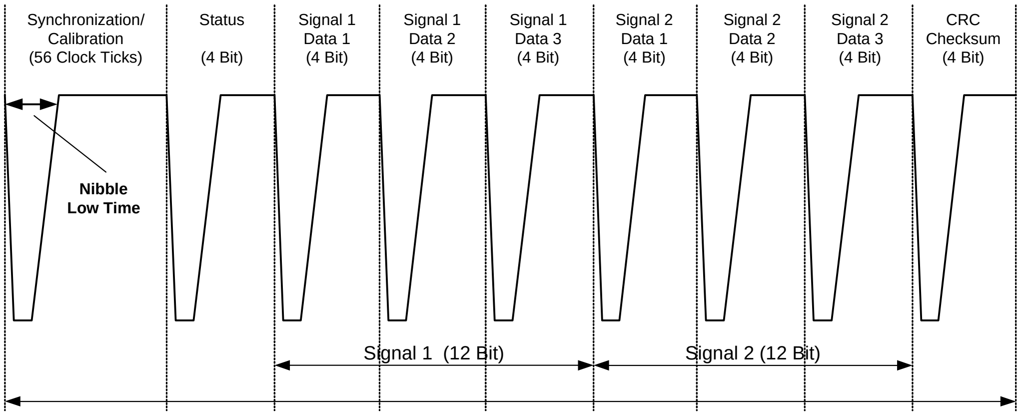
  An 8-bit counter value transmitted with nibble 4 and nibble_5. The values of "Nibble_4" and "Nibble_5" (as shown in the "Data" tab) are not considered.

Clock Tick Period [µsec]

Length of a "clock tick", which is the smallest unit of time in the SENT protocol. Specified in µsec with a resolution of 20 nsec.

Nibble Low Pulse [Ticks]

Number of ticks for the low pulse of a nibble. According to the SENT specification Jan 2010 this value should be > 4.



Frame Length Type

Transmission mode of the SENT frames.

- Variable Frame Length

  A SENT frame is expected to follow the preceding frame immediately – there is no "pause" pulse.
- Constant Frame Length

  A "pause" pulse is expected in between two SENT frames. The length of the "pause" pulse is such that the whole frame (including "pause") is of constant length.

Total Frame Lenght [Ticks]

Setting of the frame length, if "Frame Length Type" is set to "Constant Frame Length".

Checksum Calculation Method

The calculation method for the 4-bit checksum in the ES1321.1.

- Recommended CRC (0) *

  Checksum calculation according to SENT specification Jan 2010.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.
- Legacy CRC (1) *

  Checksum calculation according to SENT specification April 2007 and Feb 2008.
- CRC = 0x0 (2) *

  The value of the CRC nibble is always "0x0".
- CRC = 0xF (3) *

  The value of the CRC nibble is always "0xF".

- CRC Mode 0…3 by Signal

  With the signal "Checksum_Mode_Data" (see "Checksum_Mode_Data" on page 113) one of the calculation methods 0...3 can be selected (any other value yields CRC = 0).
- CRC Data 0…15 by Signal

  With the signal "Checksum_Mode_Data" the CRC nibble can be set explicitely.

Note

** The number in brackets indicates the mode number when "CRC Mode 0…3 by Signal" is selected.*

4.11.2 Data (ES1321-Out-SENT Device)

The following figure shows the "Data" tab of the ES1321-Out-SENT device.

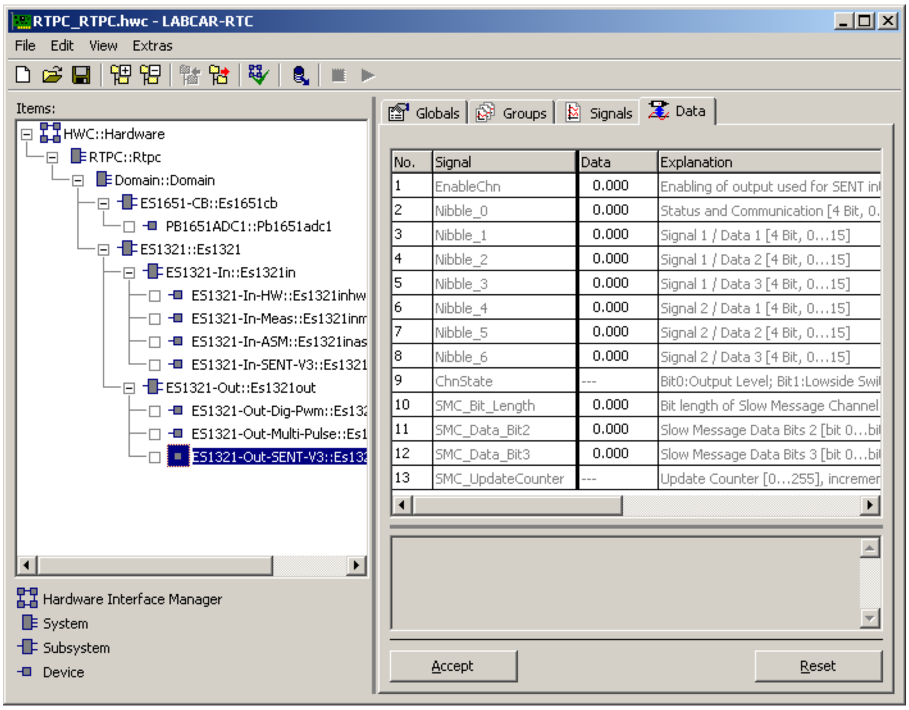


Fig. 4-30 The "Data" Tab of the ES1321-Out-SENT Device

The "Data" tab contains the following signals

EnableChn

Enables or disables the hardware channel used for the SENT output

- = 0: output is high-impedance, signal output is inactive
- = 1: signal output is active

Nibble_0..Nibble_6

The value of the individual nibbles.

ChnState

The 4-bit error state coding of the used hardware channel. The individual bits describe different settings and the resulting states of the output driver:

- Bit 0: Output Level
- Bit 1: Lowside Switch Overcurrent
- Bit 2: Highside Switch Overcurrent
- Bit 3: Error Condition Detected

SMC_Bit_Length

Input signal controlling the SMC sending unit of the ES1321.1.

"0" disables SMC sending, i.e., bit 2 and bit 3 of nibble 0 are transmitted. A value > 0 (… 31) activates the overlay of bit 2 and bit 3 of the respective data words. The value defines the number of bits for a SMC cycle. According to the SENT specification, valid values are "16" (short serial message) or "18" (enhanced serial message).

SMC_Data_Bit2

Input signal carrying the data word for bit 2 (see SAE J2716 version Jan 2010, Ch.5.2.4).

SMC_Data_Bit3

Input signal carrying the data word for bit 3 (see SAE J2716 version Jan 2010, Ch.5.2.4).

SMC_UpdateCounter

8-bit counter which increases when an SMC message was transmitted completely.

Additionally this signalizes that the ES1321.1 has started sending a new SMC message with the current values of SMC Data_Bit2 and SMC_Data_Bit3. Thus the counter can be used for synchronization purposes for the cyclic output of a list of message IDs.

The values of SMC_Data_Bit2 and SMC_Data_Bit3 are transferred at that moment, when a new short serial message is output – after data transfer SMC_Data_Bit2 and SMC_Data_Bit3 should be filled with new data. This again is communicated to the model by incrementing the value of "SMC_UpdateCounter".

The ES1321.1 and the RTIO only provides generic functionality for the reception of SMC messages. Control, decoding of the data and verification of checksums have to take place outside the RTIO.

A C code module template "SentOutSMC_Control" with basic functionality is provided. For that purpose all SMC_* signals of the RTIO are connected to the corresponding ports of the C code module.

However, for a specific application the C code module must be adapted by

- defining "SMC format mode"
- defining "SMC_Sequence" (cyclic output of a series of Msg_IDs
- providing the physical data values at the input ports

- converting the data values and assigning values to "DataField[MsgId]"

Checksum_Mode_Data

If "Checksum Calculation Method" (see page 110) was set to "CRC Mode 0…3 by Signal", one of the calculation methods 0...3 can be set here.

If "Checksum Calculation Method" was set to "CRC Data 0...15 by Signal", the CRC nibble can be set here.

5 ES1330.1 PWM I/O Counter Board

The ES1330.1 PWM I/O Counter Board enables the acquisition and output of PWM signals. The ES1330.1 has six counter components (Am9513A). Each counter component has five 16-bit-wide counters with a count frequency of 4 MHz. Thirty counter inputs and sixteen counter outputs are divided into six internal ports "Port 1 ... Port 6" on the ES1330.1. Every counter component occupies one of these ports. Access to the ports takes place via the SUB-D connector of the ES1330.1. In the RTIO Editor items list, the ES1330.1 PWM I/O Counter Board is added as a subsystem; the counter component as a device.

5.1 ES1330-PWM Subsystem

5.1.1 Globals (ES1330-PWM Subsystem)

This section describes the global options of the ES1330-PWM subsystem.

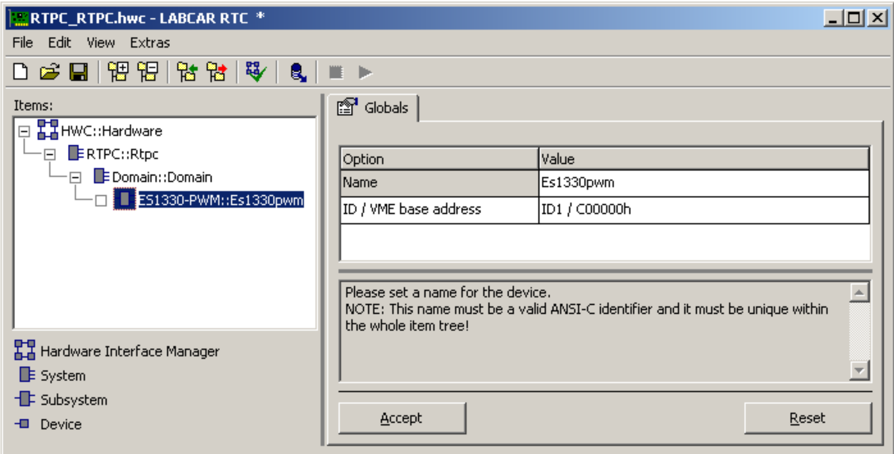


Fig. 5-1 The "Globals" Tab of the ES1330-PWM Subsystem

ID / VME base address

This line is responsible for the setting of the VME base address. This setting has to correspond to the jumper settings of the ES1330.1. Four different VME base addresses can be selected for the ES1330.1 (`ID1/C00000h`, `ID2/C00100h`, `ID3/C00200h`, `ID4/C00300h`); this means that up to four ES1330.1 PWM I/O Counter Boards can be operated in one ES1000 system. The ES1330.1 occupies an address space of 256 words from the base address.

5.2 PWM-COUNTER Device

5.2.1 Globals (PWM-COUNTER Device)

This section describes the global options of the PWM-COUNTER device.

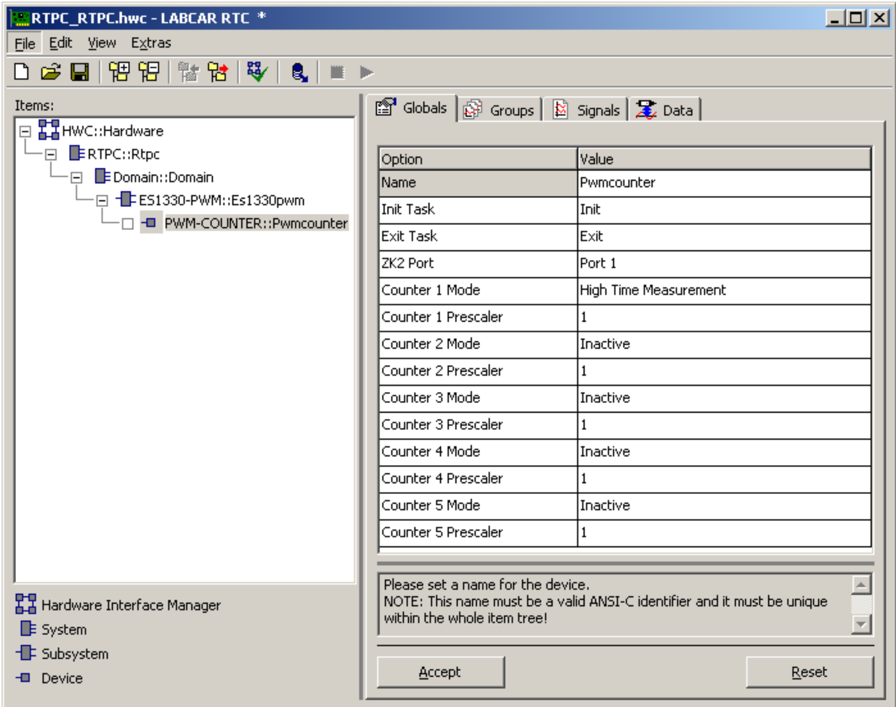


Fig. 5-2 The "Globals" Tab of the PWM-COUNTER Device

Init Task

The task for initializing the ES1330.1 is assigned in this line (Type: Init / Application Mode: active).

Exit Task

The task to be executed with the ES1330.1 when the experiment stops is assigned in this line (Type: Init / Application Mode: inactive).

ZK2 Port

This is where the port (and hence the counter component) is assigned. Every counter component has its own port.

Counter 1 Mode ... Counter 5 Mode

This is where the counter mode is selected for each of the five counters of the counter component. The following list shows the available counter modes.

- inactive

  The counter is not used.

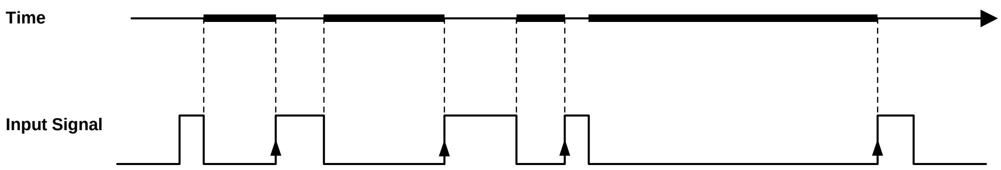
- PWM-Generator (PWM_Gen)

  The counter is used to generate PWM signals.

- Period-Measurement (P_Meas)

  The counter is used to measure the period of a PWM signal.
- Low Time Measurement

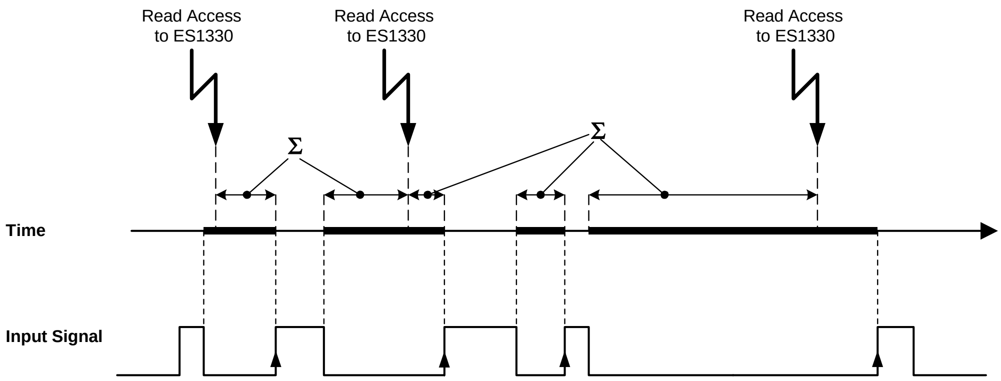
  In this mode, the duration of a pulse with an active level (here: low-active) is measured.



- High Time Measurement

  In this mode, the duration of a pulse with an active level (here: high-active) is measured.
- Additive Low Time Measurement

  The simulation target usually acquires measurements from the ES1330.1 periodically. With additive measurements, the length of time the signal was active (here: low-active) between two consecutive read-accesses of the measure register is measured.



- Additive High Time Measurement

  The simulation target usually acquires measurements from the ES1330.1 periodically. With additive measurements, the length of time the signal was active (here: high-active) between two consecutive read-accesses of the measure tab is measured.

Counter 1 Prescaler ... Counter 5 Prescaler

These lines can be used to specify a prescaler for the input frequency of 4 MHz. The scaled input frequency is used as a clock signal for the first four counters. The prescaler makes it possible to generate and evaluate PWM signals with a longer period duration; the resolution of the PWM signals is, however, reduced because of this. The following prescalers can be selected: 1, 10, 100, 1000 and 10000.

5.2.2 Groups (PWM-COUNTER Device)

This section describes the signal-group-specific options of the PWM-COUNTER device.

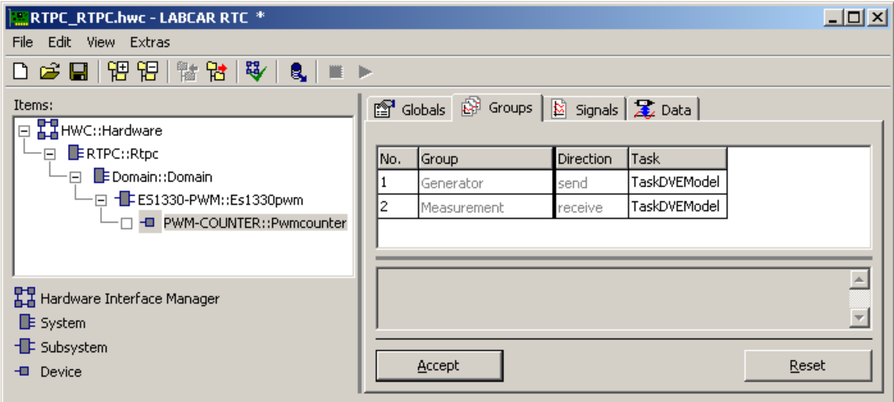


Fig. 5-3 The "Groups" Tab of the PWM-COUNTER Device

There are no item-specific columns defined for the ES1330-PWM subsystem in the "Groups" tab.

5.2.3 Signals (PWM-COUNTER Device)

This section describes the signal-specific options of the PWM-COUNTER device.

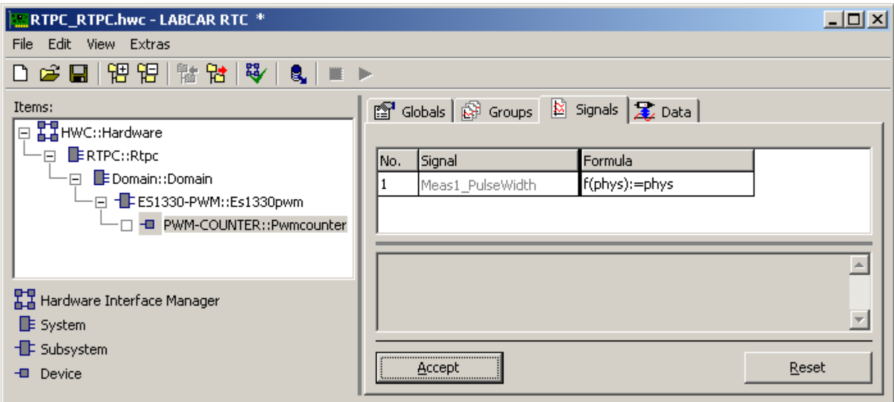


Fig. 5-4 The "Signals" Tab of the PWM-COUNTER Device

No.

The number and significance of the signals is determined by the "ZK2 Port" and "Counter X Mode" settings from the "Globals" tab.

Signal

This is where names (ANSI-C) for the signals are specifically defined. The following naming conventions apply (X = number of the counter):

- `GenX_Frequency`

  Frequency of the PWM output channel in Hz

- `GenX_DutyCycle`

  Duty cycle of the PWM output channel in %

- `MeasX_Period`
  Period duration of the PWM evaluation channel in seconds
- `MeasX_PulseWidth`
  Result of the pulse width measurement (single pulse or additive)

6 ES1331.1 Signal Generator Board

The ES1331.1 Signal Generator Board is used to generate analog and digital output signals in VMEbus systems. The board has its own digital signal processor (DSP) which works independently of the main processor of the VMEbus system. Using a DSP means that the signal form of the output signals can be programmed user-specifically and high demands of the change speed of amplitude, frequency and pulse width are easy to fulfil.

The board has four digital/analog converters with 12-bit resolution and a maximum conversion time of 6 µs for generating analog signals. The output voltage range is -10 V to +10 V (in relation to the ground potential of the VMEbus system).

The digital output signals are generated via a piggyback module with eight counters. Two counters together form an output channel. The address, data and control lines of the DSP are routed to the piggyback.

The board also has the following features:

- TMS320C203 / TMS320C206 digital signal processor
- four digital-analog converters with 12-bit resolution and an output voltage range of -10 V to +10 V
- piggyback for generating digital signals

The ES1331.1 Signal Generator Board can be used in VMEbus systems wherever analog and digital signal traces with varying frequencies, amplitudes and pulse widths are to be generated. The digital signal processor (DSP) makes flexible signal generation possible regardless of the system processor.

Examples of applications are:

- generation of active or passive speed sensor signals for driving-dynamics systems
- generation of speed signals for transmission ECUs

Structure of the ES1331 RTIO Item Tree

In RTIO, the ES1331.1 is integrated by selecting the ES1331-DSP subsystem. The required application of the ES1331.1 can be selected or specified at the level of the ES1331-DSP subsystem.

This is where you can select the WheelSnsrSim device for simulating active and passive speed sensors (see Fig. 6-1). If other applications of the ES1331.1 are implemented at a later date, the relevant RTIO items are incorporated at this level.

Note

Only one WheelSnsrSim device can be allocated to the ES1331-DSP subsystem. If you attempt to add a second device, an error message is displayed.

6.1 ES1331-DSP Subsystem

6.1.1 Globals (ES1331-DSP Subsystem)

The VMEbus base address of the ES1331.1 is specified in the "ID / VME base address" field in the "Globals" tab of the ES1331-DSP subsystem (see Fig. 6-1). There are 2 different base addresses to choose from (`ID1/580000h` and `ID2/5C0000h`).

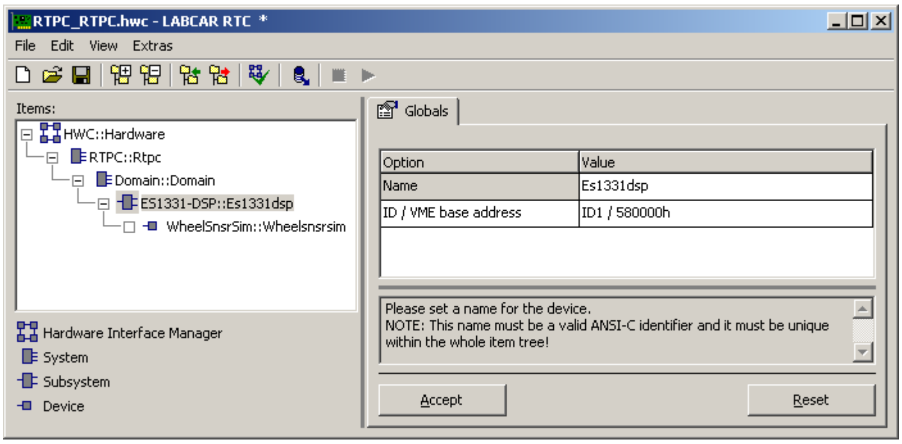


Fig. 6-1 The "Globals" Tab of the ES1331-DSP Subsystem

6.2 WheelSnsrSim Device

6.2.1 Simulation of Wheel Speed Sensors

This section is an introduction to signal output with passive and active speed sensors.

Signal Output with Passive Wheel Speed Sensors

A passive speed sensor consists of a Hall sensor which records the revolution of a toothed gear on the wheel of a vehicle. The sensor signal can be approximated using a sine function (see Fig. 6-2). Every tooth of a rotating toothed gears generates a period of the sine wave. One revolution of the toothed gear is the same as one wheel revolution..

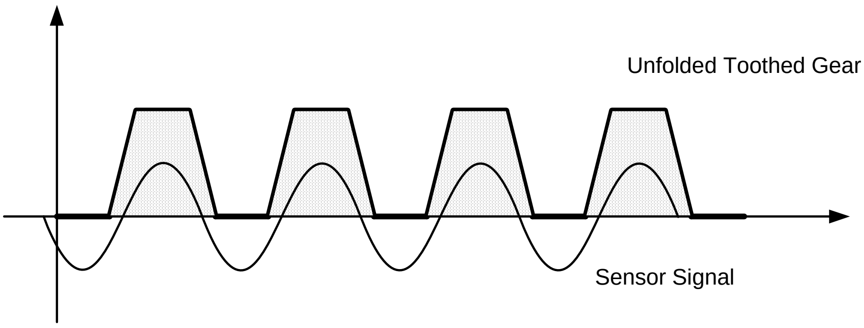


Fig. 6-2 The Sensor Signal of a Passive Speed Sensor

If you use the following symbols:

ν_{Wheel}	Wheel speed in m/s
r_{Wheel}	Wheel radius in m
T_{Wheel}	Period duration of a wheel revolution or toothed gear revolution in s
T_{Sine}	Period duration of a sine wave in s
f_{Sine}	Frequency of the sine wave in Hz
n_{Teeth}	Number of teeth of the toothed gear

you can define the following relation between the period duration T_{Sine} of the sine wave and the duration of a wheel revolution T_{Wheel} as follows:

Equ. 6-1 $T_{Sine} = T_{Wheel} / n_{Teeth}$

The relation between the wheel speed v_{Wheel} and the duration of a wheel revolution T_{Wheel} is:

Equ. 6-2 $v_{Wheel} = 2\pi\, r_{Wheel} / T_{Wheel}$

If you use Equ. 6-2 in Equ. 6-1, the following results for the period duration and the frequency of the sine wave:

Equ. 6-3 $T_{Sine} = 2\pi\, r_{Wheel} / (n_{Teeth} * \nu_{Wheel})$

and:

Equ. 6-4 $f_{Sine} = (n_{Teeth} * \nu_{Wheel}) / 2\pi\, r_{Wheel}$

The sine wave of a passive speed sensor is generated by the DSP processor and a D/A converter. It is represented in the data memory of the DSP processor by $n_{SineTable}$ equidistant voltage points (see Fig. 6-3).

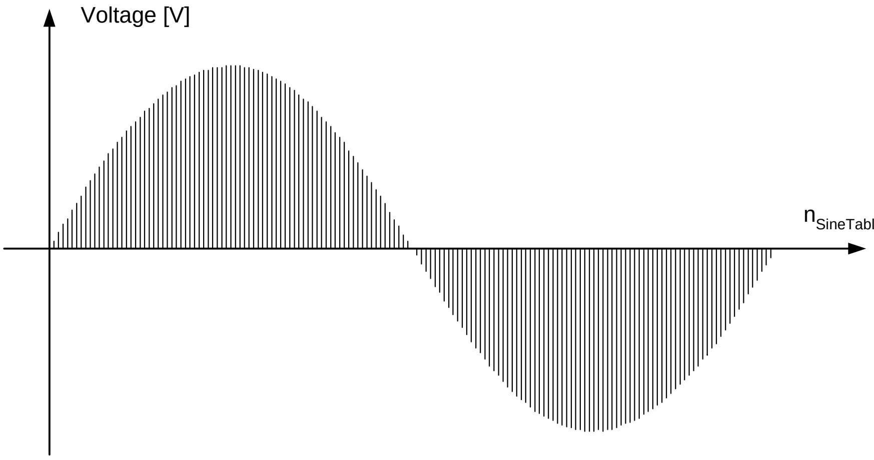


Fig. 6-3 Representation of the Sine Wave in the Data Memory of the DSP Processor

There are two basic ways of generating a sine wave with variable frequency:

1. You use a **variable sampling period** ΔT_{sample}, with which the data samples of the sine wave with a **constant increment** δ are filled with clock discrete values (see Fig. 6-4).
   $\delta = 1$ means that every data sample is filled with clock discrete values;
   $\delta = n$ means that every nth data sample is filled with clock discrete values.

2. You use a **constant sampling period** ΔT_{Sample} and a **variable increment** δ in the sine table (see Fig. 6-4)..

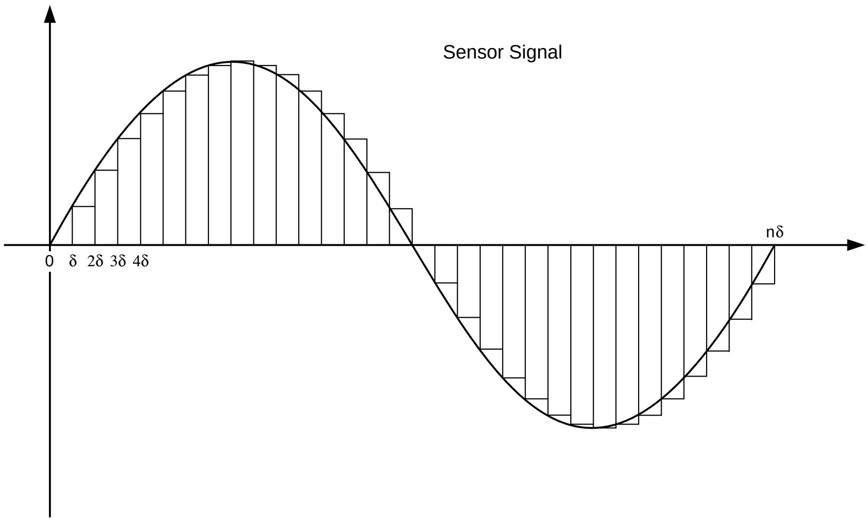


Fig. 6-4 Generation of a Variable Sine Frequency (see Text)

The DSP firmware of the ES1331.1 uses the second method to generate a variable sine frequency.

A table point of the sine table is issued at equidistant points ($\Rightarrow$ constant sampling period ΔT_{Sample}). Once a voltage value has been issued, the number of the output point in the sine table is increased by a variable increment δ (which is a function of the sine frequency). The constant sampling period ΔT_{Sample} is a multiple of the cycle time of the DSP processor $T_{DspCycle}$.

Equ. 6-5 $\Delta T_{Sample} = \lambda * T_{DspCycle}$

The cycle time of the DSP processor $T_{DspCycle}$ is 7 µs in the simulation of a passive speed sensor. λ is the proportionality constant. It is calculated from the size of the "ShiftCounter" parameter of the DSP firmware in the relation:

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

The "ShiftCounter" parameter is used to increase the size of the sampling period of the DSP for simulating low sine frequencies.

There is the following relation between the increment δ, the number of sine-table points $n_{SineTable}$ and the number of steps for generating a complete sine wave n:

Equ. 6-7 $n_{SineTable} = n * \delta$

The following is true for the period duration T_{Sine} of the sine wave:

Equ. 6-8 $T_{Sine} = n * \Delta T_{Sample}$

If you solve Equ. 6-8 in accordance with ΔT_{Sample}, eliminate n using Equ. 6-7 and then use the resulting equation Equ. 6-5, the following results:

Equ. 6-9 $\delta = (n_{SineTable} * \lambda * T_{DspCycle}) / T_{Sine}$

This then results in the following for the sine frequency f_{Sine}:

Equ. 6-10 $f_{Sine} = \delta / (n_{SineTable} * \lambda * T_{DspCycle})$

where $T_{DspCycle}$ = 7 µs.

With Equ. 6-3 you receive the following from Equ. 6-9:

Equ. 6-11 $\delta = (n_{SineTable} * \lambda * T_{DspCycle} * n_{Teeth} * \nu_{Wheel}) / 2\pi r_{Wheel}$

where $T_{DspCycle}$ = 7 µs.

Signal Output with Active Wheel Speed Sensors

The setup of an active speed sensor is the same as that of a passive one, the difference being that a digital PWM signal, not an analog signal, is generated. The RTIO link and the DSP firmware of the ES1331.1 enable two types of active speed sensor to be simulated:

- with speed sensors of type DF11I, a PWM impulse is generated each time the edge of a tooth is detected (see Fig. 6-5)
- with speed sensors of type DF10, a PWM impulse is only generated if the 'trailing edge' of a tooth is detected (see Fig. 6-5)

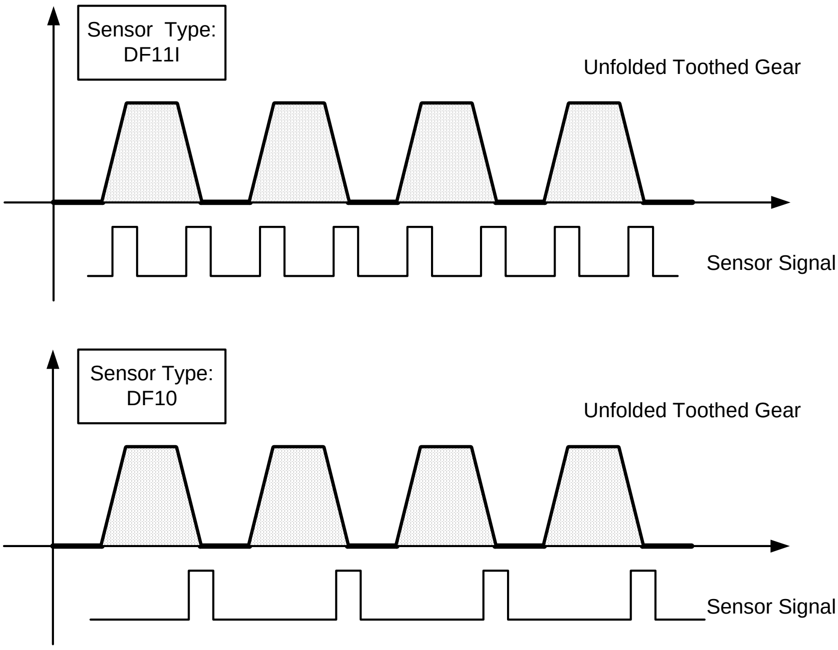


Fig. 6-5 Sensor Signal of a DF11I and a DF10 Speed Sensor

If T_{PWM} and f_{PWM} are the period duration and frequency of the PWM signal, the following relation is possible:

Equ. 6-12 $T_{PWM} = T_{Wheel} / \Psi * n_{Teeth}$

The factor Ψ is 2 with speed sensors of type DF11I and 1 with speed sensors of type DF10.

The following results with Equ. 6-2:

Equ. 6-13 $T_{PWM} = 2\pi r_{Wheel} / (\Psi * n_{Teeth} * \nu_{Wheel})$

and

Equ. 6-14 $f_{PWM} = (\Psi * n_{Teeth} * \nu_{Wheel}) / 2\pi r_{Wheel}$

The DSP firmware always references the next tooth of a toothed gear when an internal 22-bit counter overflows. The 22-bit counter is timed with a cycle time, $T_{DspCycle}$, of 4.5 µs. This then results in the following relation for the period duration T_{PWM} of the PWM signal:

Equ. 6-15 $T_{PWM} = (2^{\kappa} * T_{DspCycle}) / (\delta * \Psi)$

and

Equ. 6-16 $f_{PWM} = (\delta * \Psi) / (2^{\kappa} * T_{DspCycle})$

κ refers to the bit length of the counter ($\kappa = 22$) and δ to the increment of the counter.

Equ. 6-15 results in the following for δ:

Equ. 6-17 $\delta = (2^{\kappa} * T_{DspCycle}) / (T_{PWM} * \Psi)$

and

Equ. 6-18 $\delta = (2^{\kappa} * f_{PWM} * T_{DspCycle}) / \Psi$

where $T_{DspCycle} = 4.5$ µs, $\kappa = 22$, $\Psi = 2$ (DF11I) and $\Psi = 1$ (DF10).

Equ. 6-13 results in the following

Equ. 6-19 $\delta = (2^{\kappa} * n_{Teeth} * v_{Wheel} * T_{DspCycle}) / 2\pi\ r_{Wheel}$

where $T_{DspCycle} = 4.5$ µs and $\kappa = 22$.

6.2.2 Globals (WheelSnsrSim Device)

The frequency of the sine signal of a passive speed sensor or the frequency of the PWM signal of an active speed sensor can be specified by the user in three different ways. You select how the signal frequency is to be specified in the "User Mode" field of the "Globals" tab of a WheelSnsrSim device (see Fig. 6-6).

- If the "Wheel Speed Mode" option is selected, the signal frequency is determined by specifying a wheel speed. This wheel speed is used to determine the sine frequency of a passive speed sensor in accordance with Equ. 6-4 on page 123 and the PWM frequency of an active speed sensor in accordance with Equ. 6-14 on page 125.
- When you select the option "Frequency Mode", the signal frequency is determined directly by specifying the frequency (in Hz).

- In "Expert Mode" the δ value is used to determine the signal frequency. This is used to define the sine frequency of passive speed sensors in accordance with Equ. 6-10 on page 124 and the PWM frequency of active speed sensors in accordance with Equ. 6-16 on page 126.

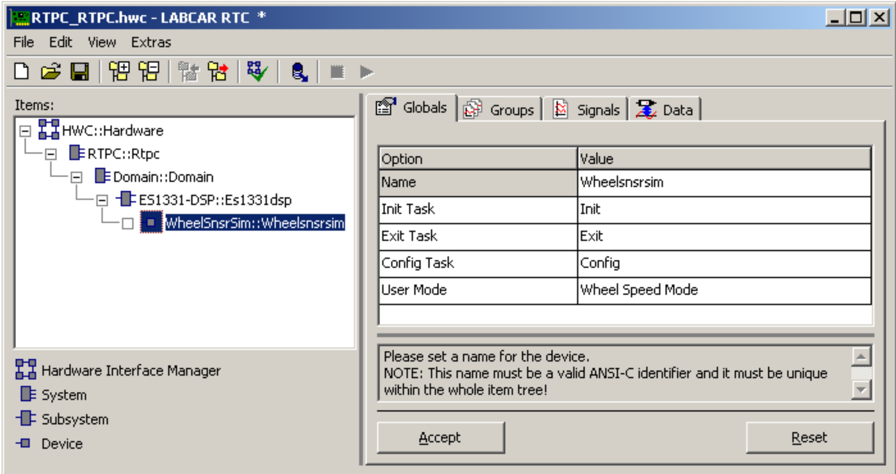


Fig. 6-6 The "Globals" Tab of a WheelSnsrSim Device

Tab. 6-1 lists the fields of the "Globals" tab of the WheelSnsrSim device which are used to configure the real-time operating system.

Field	Meaning	Note
Init Task	Allocation of the driver initialization process to an ERCOS^{EK} task.	ERCOS^{EK} task: *Type:* "Init" *Application Mode:* "Active"
Exit Task	Allocation of the Exit process of the driver to an ERCOS^{EK} task.	ERCOS^{EK} task: *Type:* "Init" *Application Mode:* "Inactive"
Config Task	Allocation of the driver configuration process to an ERCOS^{EK} task.	ERCOS^{EK} task: *Type:* "Alarm" *Application Mode:* "Active" *Period:* 100 ms – 1000 ms

Tab. 6-1 The Fields of the "Globals" Tab of the WheelSnsrSim Item for the Configuration of the Real-Time Operating System

6.2.3 Groups (WheelSnsrSim Device)

A WheelSnsrSim device has three signal groups (see Fig. 6-7 on page 128).

- The *Config* signal group is transferred from the simulation target to the ES1331.1. The RTIO signals of the signal group are configuration parameters of the DSP firmware, i.e., these signals are only processed by the DSP firmware during the initialization phase.
  This is why any change to an RTIO signal of the *Config* signal group results in an initialization of the DSP firmware. This initialization can take several seconds depending on the type of speed sensor to be simulated and the selected configuration parameters.
  The *Config* signal group cannot be allocated any tasks of the real-time

operating system in the "Groups" tab. The calculation of the *Config* signal group takes place implicitly in the "Init" and "Config" process of the RTIO driver.

Note

The allocation of the "Init" and "Config" processes to tasks of the real-time operating system takes place in the "Init Task" and "Config Task" fields of the "Globals" tab of the WheelSnsrSim device (cf. the explanations in Tab. 6-1 on page 127).

- The *Outputs* signal group is transferred from the simulation target to the ES1331.1. This signal group contains all signals for controlling the speed sensor simulation in real time.
- The *Inputs* signal group is transferred from the ES1331.1 to the simulation target. This signal group contains status information of the DSP firmware.

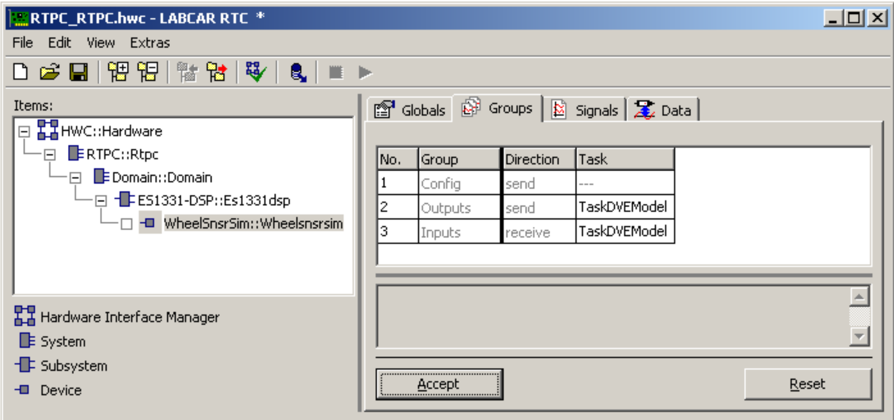


Fig. 6-7 The "Groups" Tab of the WheelSnsrSim Device

6.2.4 Signals (WheelSnsrSim Device)

Fig. 6-8 shows the "Signals" tab of a WheelSnsrSim device.

The endings '_FL', '_FR', '_RL', '_RR' in the signal names specify the wheel. '_FL' is the front left wheel, '_FR' the front right wheel, '_RL' the rear left wheel and '_RR' the rear right wheel.

Note

The RTIO driver for the ES1331.1 Signal Generator Board does not verify whether the individual RTIO signals have admissible and useful values. This is the responsibility of the user of the ES1331.1 board and its RTIO integration.

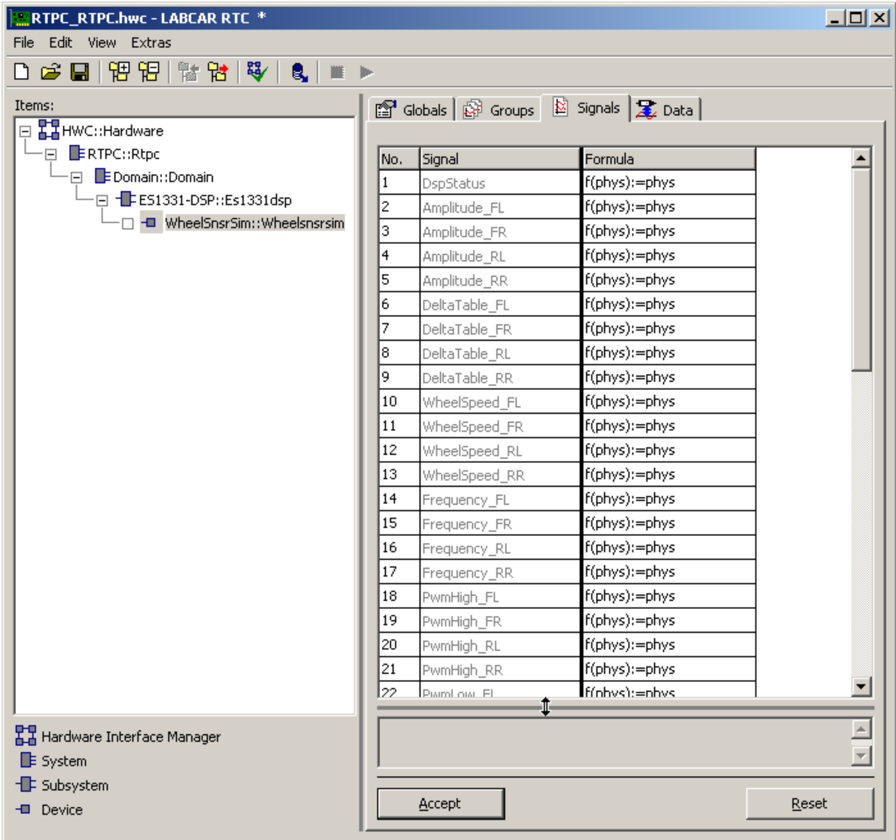


Fig. 6-8 The "Signals" Tab of the WheelSnsrSim Device

The RTIO Signals of the Inputs Signal Group

The RTIO signal *DspStatus* shows the status of the DSP firmware. The features of the RTIO signals of the *Inputs* signal group are listed in Tab. 6-2.

RTIO Signal	Data Type	Signal Group	Description
DspStatus	uint16	Inputs	Status of the ES1331.1 DSP firmware 1: Initialized 2: Running 4: Active Running 8: Active Initialized

Tab. 6-2 The RTIO Signals of the *Inputs* Signal Group

Note

If data types are specified in Tab. 6-2, Tab. 6-3 and Tab. 6-4, these are the data types that the RTIO driver uses internally.

The RTIO Signals of the Outputs Signal Group

The RTIO *Amplitude_XX* signals are only of any significance in the simulation of a passive speed sensor and specify the amplitude of the sine signal or speed sensor signal in volts.

The RTIO *DeltaTable_XX* signals specify the increment δ in the sine table in accordance with Equ. 6-11 on page 125 in the simulation of a passive speed sensor. In the simulation of an active speed sensor, these signals specify the increment δ of the internal counter in accordance with Equ. 6-19 on page 126. These RTIO signals are only of any significance if "Expert Mode" is selected in the "User Mode" field in the "Globals" tab.

The RTIO *WheelSpeed_XX* signals specify the wheel velocity in km/h. If "Wheel Speed Mode" is selected in the "User Mode" field in the "Globals" tab, these RTIO signals determine the sine frequencies of the passive speed sensors and the PWM frequencies of the active speed sensors. If a different option is selected in the "User Mode" field, these RTIO signals are of no significance.

The RTIO *Frequency_XX* signals specify the signal frequencies of the active and passive speed sensors if "Frequency Mode" is selected in the "User Mode" field of the "Globals" tab. If a different option is selected in the "User Mode" field, these RTIO signals are of no significance.

The RTIO *PwmHigh_XX* signals are only of any significance for the simulation of an active speed sensor. These signals define the maximum duration of the high phase of the PWM signals. The following is true for the duration $T_{PwmHighMax}$:

Equ. 6-20 $T_{PwmHighMax} = (\mathit{PwmHigh\_XX} + 1) * T_{DspCycle}$

The cycle time $T_{DspCycle}$ is 4.5 µs.

The RTIO *PwmLow_XX* signals are only of any significance for the simulation of an active speed sensor. These signals define the minimum duration of the low phase of the PWM signals. The following is true for the duration $T_{PwmLowMin}$:

Equ. 6-21 $T_{PwmLowMin} = (\mathit{PwmLow\_XX} + 1) * T_{DspCycle}$

The cycle time $T_{DspCycle}$ is 4.5 µs.

Note

The specification of the signal frequency, high phase and low phase would be too much for a PWM signal. The user can only specify the frequency and the maximum duration $T_{PwmHighMax}$ *of the high phase and the minimum duration* $T_{PwmLowMin}$ *of the low phase of the PWM signal. The frequency specification is generally adhered to and, providing the minimum duration* $T_{PwmLowMin}$ *of the low phase is not violated, the duration of the high phase is* $T_{PwmHighMax}$.

The RTIO *NoMove_XX* signals determine whether the particular wheel should rotate or not.

The RTIO *Info_XX*, *ErrorPulse_XX* and *Mask_XX* signals are intended for future add-ons and have no function in the current version of the RTIO driver.

The RTIO *WheelRadius_XX* signals specify the wheel radius in meters. These RTIO signals are only of significance if the frequencies of the speed sensor signals are determined by the specification of a wheel speed, i.e. if "Wheel Speed Mode" is activated in the "User Mode" field in the "Globals" tab.

The features of the RTIO signals of the *Outputs* signal group are listed in Tab. 6-3.

RTIO Signal	Data Type	Signal Group	Description
Amplitude_XX	real32	Outputs	Amplitudes of the sine signals or speed sensor signals in volts. Only of significance for the simulation of a passive speed sensor. Admissible value range: 0.0 V ... 10.0 V
DeltaTable_XX	uint16	Outputs	Passive speed sensor: increment δ in the sine table in accordance with Equ. 6-11 on page 125. Active speed sensor: increment δ of the internal counters in accordance with Equ. 6-19 on page 126. Only of significance in "Expert Mode".
WheelSpeed_XX	real32	Outputs	Wheel speed in km/h. Only of significance in "Wheel Speed Mode".
Frequency_XX	real32	Outputs	Frequency of the speed sensor signal in Hz. Only of significance in "Frequency Mode".
PwmHigh_XX	uint16	Outputs	Maximum duration of the high phase of the PWM signals. Only of significance for the simulation of an active speed sensor. Admissible value range: 0 ... 2047
PwmLow_XX	uint16	Outputs	Minimum duration of the low phase of the PWM signals. Only of significance for the simulation of an active speed sensor. Admissible value range: 0 ... 31
NoMove_XX	uint16	Outputs	"No move" display Only of significance for the simulation of an active speed sensor. Recommended value range: 0: moving 1: not moving
Info_XX	uint16	Outputs	For future add-ons

RTIO Signal	Data Type	Signal Group	Description
ErrorPulse_XX	uint16	Outputs	For future add-ons
Mask_XX	uint16	Outputs	For future add-ons
WheelRadius_XX	real32	Outputs	Wheel radius in m. Only of significance in "Wheel Speed Mode".

Tab. 6-3 The RTIO Signals of the *Outputs* Signal Group

The RTIO Signals of the Config Signal Group

All RTIO signals of the *Config* signal group are configuration parameters of the DSP firmware. If a configuration parameter is changed, the DSP firmware is initialized.

The *DspMode* RTIO signal determines the type of speed sensor to be simulated:

- 3 stands for the simulation of a passive speed sensor
- 1 results in the simulation of an active speed sensor of type DF11I
- 2 results in the simulation of an active speed sensor of type DF10

The RTIO signal *TabCount* is only of any significance in the simulation of a passive speed sensor. This signals determines the number of data samples in the sine table ($n_{SineTable}$ parameter in Equ. 6-7 on page 124 and Equ. 6-9 to Equ. 6-11 on page 125). A recommended value is *TabCount* = 4096.

The RTIO signal *ShiftCounter* is also only of significance in the simulation of a passive speed sensor. This signal determines the numerical value for the internal DSP parameter "ShiftCounter" (see Equ. 6-6 on page 124). A recommended value is *ShiftCounter* = 8.

The RTIO *TeethCount_XX* signals specify the number of teeth of the speed sensor toothed gears. A recommended value is *TeethCount_XX* = 48.

The RTIO signals *DAC0* and *DAC1* are only of any significance in the simulation of an active speed sensor. These RTIO signals are used to determine the output voltages of the two D/A converter outputs routed to the front panel.

The features of the RTIO signals of the "Config" signal group are listed in Tab. 6-4.

RTIO Signal	Data Type	Signal Group	Description
DspMode	uint16	Config	Type of speed sensor to be simulated. Admissible values: 1: Active speed sensor - DF11I 2: Active speed sensor - DF10 3: Passive speed sensor
TabCount	uint16	Config	Number of data samples of the sine table. $n_{SineTable}$ parameter in Equ. 6-7 on page 124. Only of significance in the simulation of a passive speed sensor. Recommended value range: 32 ... 16384. A recommended value is *TabCount* = 4096
ShiftCounter	uint16	Config	ShiftCounter parameter in Equ. 6-6. Only of significance in the simulation of a passive speed sensor. Recommended value range: 1 ... 15. A recommended value is *ShiftCounter* = 8
TeethCount_XX	uint16	Config	Number of teeth of the speed sensor toothed gear. n_{Teeth} parameter in Equ. 6-1 on page 123 and Equ. 6-12 on page 125. Recommended value range: 1 ... 128. A recommended value is *TeethCount_XX = 48*
DAC0, DAC1	real32	Config	D/A converter output voltages in volts. Only of significance in the simulation of an active speed sensor. Admissible value range: -10.0 V ... 10.0 V

Tab. 6-4 The RTIO Signals of the "Config" Signal Group

7 ES1332.1 Arbitrary Signal Generator Board

The ES1332.1 Arbitrary Signal Generator Board is a VMEbus board for generating signal traces. It has two signal generators each with two output channels. The signal generators can only be operated synchronously to one another.

The RTIO integration of the ES1332.1 is tailored to the intended application in the automotive environment, namely the generation of speed-synchronous measuring and control signals. Examples are:

- speed signal - is used by the engine ECU to measure the speed and the crankshaft angle
- camshaft signal - is required by the ECU for cylinder detection

Structure of the ES1332 RTIO Item Tree

In RTIO, the ES1332.1 is incorporated by selecting the ES1332-DSP subsystem.

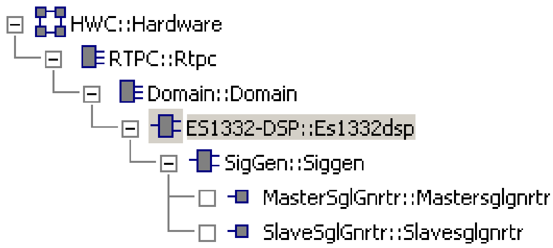


Fig. 7-1 ES1332.1 RTIO Item Tree

The required application of the ES1332.1 can be selected and specified at the level of the ES1332-DSP subsystem. Currently, only the SigGen subsystem for arbitrary signal generation can be selected (see Fig. 7-1). If, in the future, other applications of the ES1332.1 are implemented, the relevant RTIO items will be incorporated at this level.

A MasterSglGnrtr and a SlaveSglGnrtr device can be added at the level of the SigGen subsystem. These devices correspond to the two signal generators physically present.

Note

A SlaveSglGnrtr device cannot exist without a MasterSglGnrtr device, which is why the master item has to be added first (is ensured by the software).

A difference between the master and the slave generator is necessary due to the synchronous operation of the two signal generators which leads to there being a common engine speed or signal frequency for both signal generators. Only one signal generator (the master generator) can specify the engine speed or the signal frequency which is then also valid for the slave generator.

A further real-time value common to both generators is the current crankshaft angle which is transmitted from the ES1332.1 to the simulation target from the master generator item but not from the slave generator item.

7.1 ES1332-DSP Subsystem

7.1.1 Globals (ES1332-DSP Subsystem)

This section describes the global options of the ES1332-DSP subsystem.

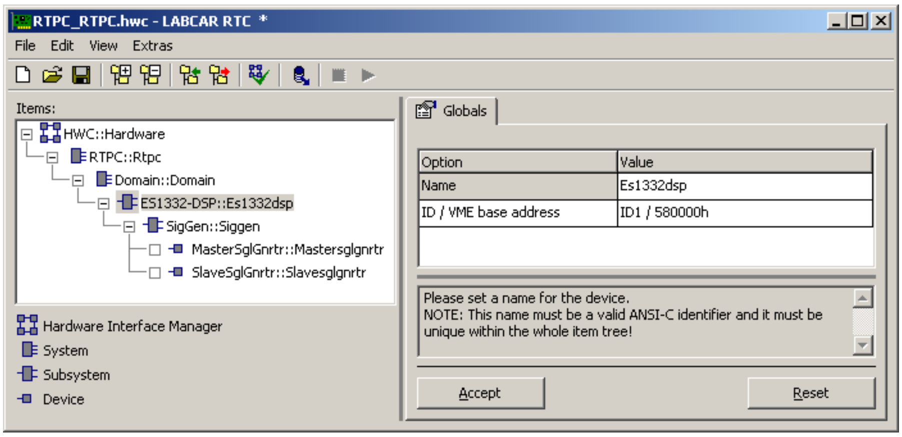


Fig. 7-2 The "Globals" Tab of the ES1332-DSP Subsystem

ID / VME base address

The VMEbus base address of the ES1332.1 is specified in the "ID / VME base address" field. There are 2 base addresses to choose from (`ID1/580000h` and `ID2/5C0000h`).

7.2 SigGen Subsystem

7.2.1 Globals (SigGen Subsystem)

This section describes the global options of the SigGen subsystem.

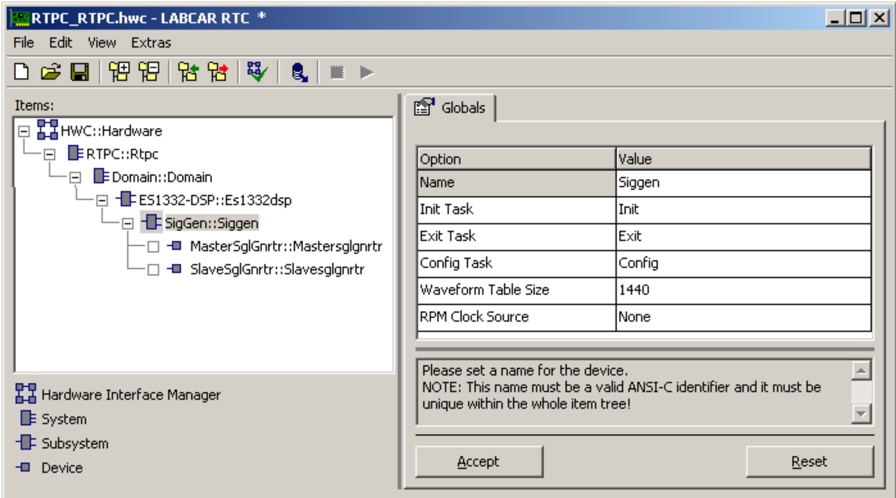


Fig. 7-3 The "Globals" Tab of the SigGen Subsystem

Option	Description	Value Range/ Note
Init Task	Allocation of the driver initialization process to an ERCOS^{EK} task	ERCOS^{EK} task: Type: 'Init' Application mode: 'Active'
Exit Task	Allocation of the exit process of the driver to an ERCOS^{EK} task	ERCOS^{EK} task: Type: 'Init' Application mode: 'Inactive'
Config Task	Allocation of the driver configuration process to an ERCOS^{EK} task	ERCOS^{EK} task: Type: 'Alarm' Application mode: 'Active' Period: 100 ms - 500 ms
RPM Clock Source	Source of the RPM clock signal	- None - Master Generator Channel A - Master Generator Channel B - Slave Generator Channel A - Slave Generator Channel B
Waveform Table Size	Signal bank size in data samples	2 - 1440

Tab. 7-1 Fields of the "Globals" Tab of the SigGen Subsystem

RPM Clock Source

The option specifies which of the four ES1332.1 channels outputs an RPM clock signal. The RPM clock is used to synchronize hardware to the current engine speed.

A 10 V pulse indicates zero-crossing of the camshaft angle, all other clock pulses have an amplitude of 5 V. A rising or falling clock edge occurs every 0.5° crankshaft angle.

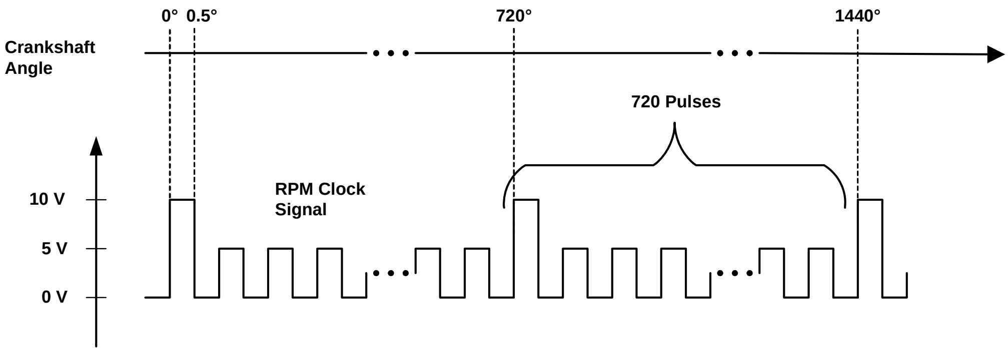


Fig. 7-4 RPM Clock Signals

Note

If a channel is configured for RPM clock output then the following restrictions apply:
1. The "Waveform Table Size" option is forced to 1440.
2. Waveform 4 of the signal generator which outputs the RPM clock is used for the RPM clock signal and hence is reserved.
3. The RTIO signals ChnX_SigSel, ChnX_Mode and ChnX_PhaseShift (see Tab. 7-3 on page 143) of the arb channel which outputs the RPM clock are disabled.

Waveform Table Size

The "Waveform Table Size" specifies the signal bank size n_{Bank}, i.e. the number of data samples of the signal traces.

The signal bank size can be between 2 and 1440; it must, however, always be greater than or equal to the number of signal form data samples. The top limit of 1440 data samples is derived from the application of the ES1332.1 in the automotive sector (engine ECU test). In this application, the curve traces are stored in the signal banks at 2 revolutions, i.e. after 2 revolutions of the crankshaft (720° crankshaft). With 1440 data samples, these signals can be described with an angular resolution of 0.5°.

7.3 MasterSglGnrtr Device

7.3.1 Globals (MasterSglGnrtr Device)

This section describes the global options of the MasterSglGnrtr device. Information is only given separately on the SlaveSglGnrtr device where it differs from the MasterSglGnrtr device.

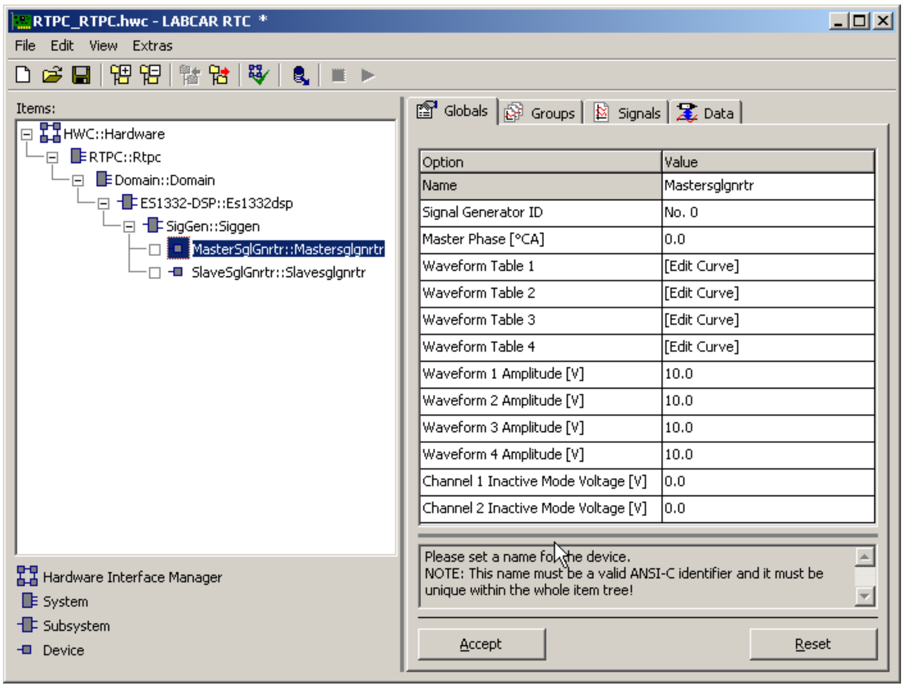


Fig. 7-5 The "Globals" Tab of the MasterSglGnrtr Device

Option	Description	Value Range/Note
Signal Generator ID	Allocation of a physical signal generator to the RTIO item	Generator 1 or Generator 2
Master Phase [°CA]	Phase shift of the signals valid for both output channels in degrees crankshaft angle	-720.0 ... 720.0
Waveform Table x (x = 1, 2, 3, 4)	Table editor to define the signal form	
Waveform x Amplitude [V] (x = 1, 2, 3, 4)	Amplitude of the signal trace of a signal bank in volts	0.0 ... 10.0
Channel x Inactive Mode Voltage [V] (x = 1, 2)	Output voltage of a channel in inactive mode in volts	-10.0 ... 10.0

Tab. 7-2 Fields of the "Globals" Tab of the MasterSglGnrtr/SlaveSglGnrtr Devices

Signal Generator ID

In the "Signal Generator ID" field, the physical signal generator (Generator 1 or Generator 2) is allocated to the RTIO item.

Master Phase [°CA]

In the "Master Phase [°CA]" field, a phase shift valid for both output channels can be set. Again in view of the intended application in the automotive sector for engine ECU tests, this phase shift is specified in degrees crankshaft angle. With an angle of ±360°, the signal trace is moved half the signal bank size; with an angle of ±720°, the signal trace is moved the entire signal bank size, i.e. it is effectively not moved.

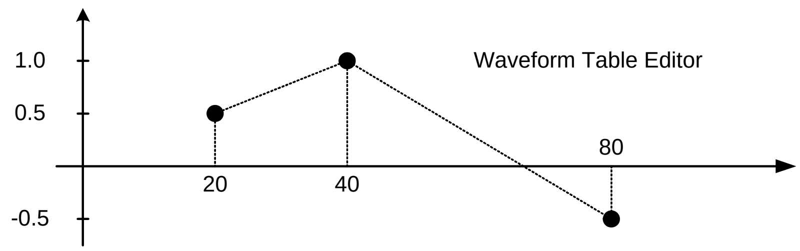
Waveform Table x

The editors for defining the signal forms of the signal traces allocated to a generator can be opened in the "Waveform Table x" fields (x = 1, 2, 3, 4).

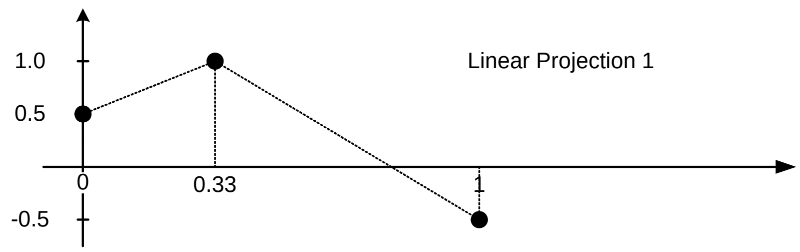
There is no limitation for the value range of the x values to be entered. The number of data samples has to be in the range 2 to 1024. The value range for the z values is -1.0 to 1.0.

The mapping of the signal trace defined in this table to the internal signal table is described below.

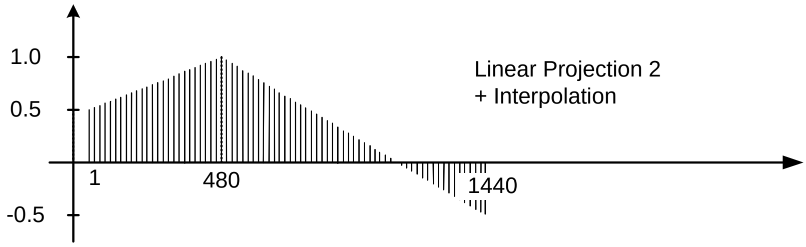
The values entered in the table to define the signal trace are, for example, as follows:



1. First of all, the range of x values entered (here: 20 - 80) is mapped by a linear transformation to the interval [0,1].



2. The second step is the mapping of this interval to the bank size which is determined by the parameter "Waveform Table Size" (here: 1440) and a linear interpolation of all points available in this interval.



Waveform x Amplitude [V]

The amplitudes of the signal traces can be specified in the "Waveform x Amplitude [V]" fields (x = 1, 2, 3, 4).

Channel x Inactive Mode Voltage [V]

The output channels of a generator can be operated in two different modes. In what is referred to as *active mode,* the currently active signal trace is issued at the channel output. In *inactive mode,* a direct voltage is output. The value of this direct voltage is defined separately in the input fields "Channel x Inactive Mode Voltage [V]" (x = 1, 2) for both channels of a generator. You can toggle between modes during runtime in real time.

All options of the "Globals" tab can be reconfigured during runtime apart from the "Signal Generator ID" option.

7.3.2 Groups (MasterSglGnrtr Device)

This section describes the signal-group-specific options of the MasterSglGnrtr device.

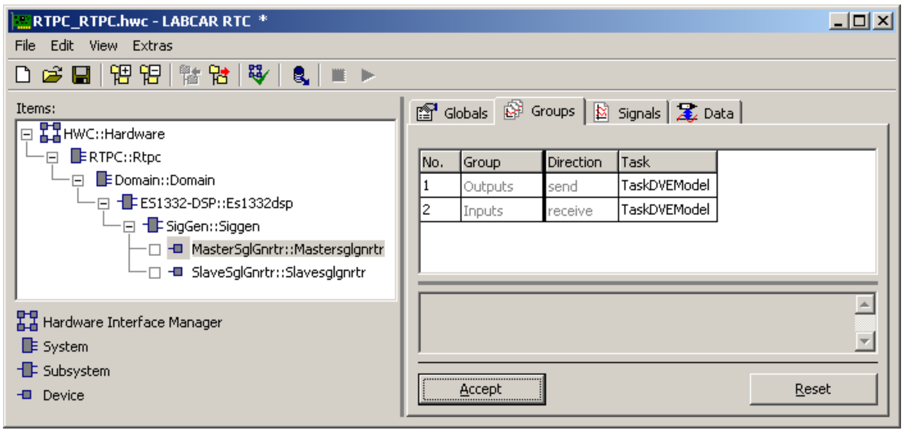


Fig. 7-6 The "Groups" Tab of the MasterSglGnrtr Device

The MasterSglGnrtr device has two signal groups:

Outputs

The signals of the "Outputs" signal group are transmitted from the simulation target to the ES1332.1.

Inputs

The signals of the "Inputs" signal group are transmitted from the ES1332.1 to the simulation target. A detailed description of the signals can be found in section 7.3.3 on page 142.

In the "Groups" tab (see Fig. 7-6), the processes in which these signal groups are transmitted, are assigned to tasks of the real-time operating system. The tasks should have "Alarm" as type and "Active" as application mode.

Note

The SlaveSglGnrtr device has no "Inputs" signal group as the current crankshaft angle, which is transmitted in the "Inputs" signal group, is only transmitted from the master generator.

7.3.3 Signals (MasterSglGnrtr Device)

The signals are pre-set. The "Signals" tab lists the individual signals.

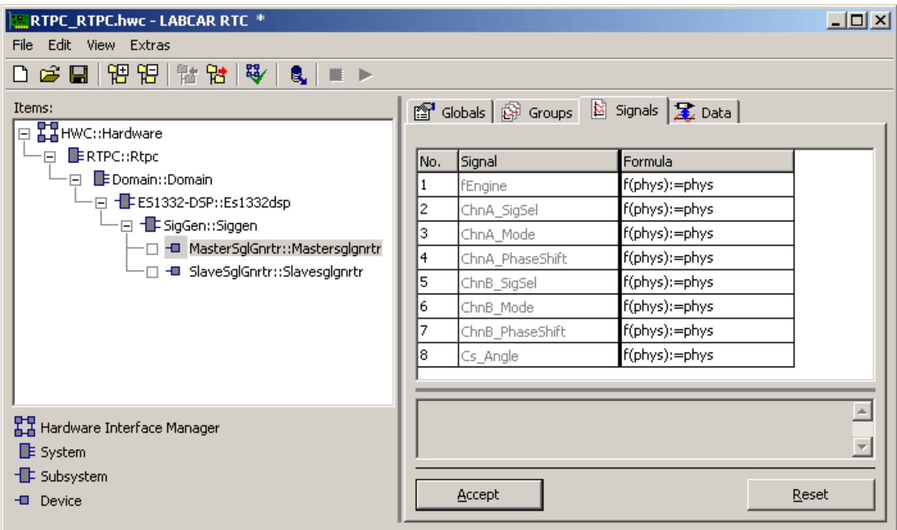


Fig. 7-7 The "Signals" Tab of the MasterSglGnrtr Device

Tab. 7-3 lists the characteristics of the signals:

Signal	Description	Data Type	
fEngine	Engine speed in rpm	real32	4 / WaveformTableSize ... 10.8×10^6 / Waveform-TableSize Resolution: 2 rpm
ChnX_SigSel X = A, B	Selection of the (active) signal bank	uint8	0: signal bank 1 1: signal bank 2 2: signal bank 3 3: signal bank 4
ChnX_Mode X = A, B	Switching mode	uint8	0: active mode 1: inactive mode
ChnX_PhaseShift X = A, B	Phase shift in degrees crankshaft angle	real32	-720.0 ... 720.0
Cs_Angle	Current crankshaft angle in degrees	real32	0 ... 720.0

Tab. 7-3 The Signals of the MasterSglGnrtr Device

Note

When data types are specified in this document, they are the data types used internally by the RTIO driver.

Note

A SlaveSglGnrtr device has the same signals as a MasterSglGnrtr device, apart from the fEngine signal for controlling engine speed and the Cs_Angle signal for reading the crankshaft angle.

fEngine

With the engine ECU simulation, the output frequency of the signal traces is defined by the engine speed. The signal traces of two consecutive engine revolutions are stored in the signal banks. This is why there is the following relation between the engine speed/engine frequency f_{Engine} and the signal frequency f_S:

$f_{Engine} = 2 \times f_S$

The *fEngine* signal is used to specify the engine speed in rpm. The minimum and maximum engine speed that can be set can be calculated using

$f_T = n_{Bank} \times f_S$

$T_S = n_{Bank} \times T_T$

from the minimum and maximum sampling frequency $f_{T,Min}$ and $f_{T,Max}$ (see Tab. 7-4). The resolution of the engine speed is 2 rpm.

Maximum sampling frequency $f_{T,Max}$	90 kHz
Minimum sampling frequency $f_{T,Min}$	33 mHz

Tab. 7-4 Maximum and Minimum Sampling Frequency

ChnX_SigSel

The *ChnX_SigSel* (X = A, B) signals specify the active signal banks for the two channels of a signal generator.

ChnX_Mode

The *ChnX_Mode* (X = A, B) signals specify the modes of the two channels of a a signal generator.

ChnX_PhaseShift

The basic shift "Master Phase [°CA]" described in section 7.3.1 valid jointly for both channels is added to the phase shifts which are defined separately for both channels with the *ChnX_PhaseShift* (X = A, B) signals. The phase shift of the *ChnX_PhaseShift* signals is also specified in degrees crankshaft angle. With a phase shift of ±360°, the signal trace is moved by half the signal bank size.

Cs_Angle

The *Cs_Angle* signal returns the current crankshaft angle in degrees. The number of the signal bank data sample currently issued can be calculated by dividing by 720 and multiplying with the signal bank size n_{Bank}(cleared of all phase shift, i.e. with phase shift 0).

8 ES1334.1 Measurement Board

Definitions

Active signal edge:

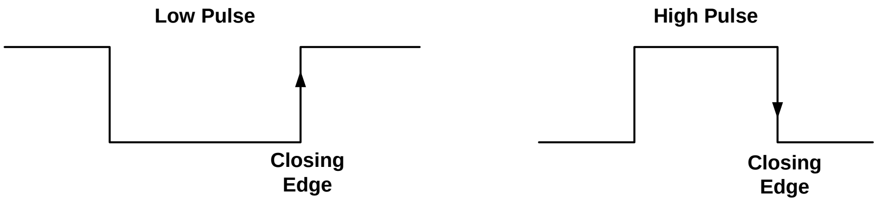
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing pulse edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.



Inactive signal:

A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

LWL:

Stands for **L**ower **W**indow **L**imit and refers to the lower limit of the angle window in speed-synchronous measurements.

Opening pulse edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.

UWL:

Stands for **U**pper **W**indow **L**imit and refers to the upper limit of the angle window in speed-synchronous measurements.

8.1 Functional Description

8.1.1 Basic Principle of Measurement Value Determination

To ensure the greatest possible flexibility in the evaluation of pending input signals, the ES1334.1 Measurement Board is based on the following basic principle:

First of all, each input signal is conditioned separately (analogously), then compared with a threshold value (which can be set using the software) to convert the analog input signal to digital 0/1 or inactive/active information. The binary level information of the signals obtained in this way at the 16 input channels is continuously checked for changes by a hardware circuit, i.e. "edges" are recognized on one or more input bits. If at least one input signal changes from active to inactive or vice versa, the circuit saves the current statuses of the 16 inputs and calls the current values of two integrated counters which specify the current time and the current crankshaft angle. This information is transferred by the hardware to an FIFO buffer.

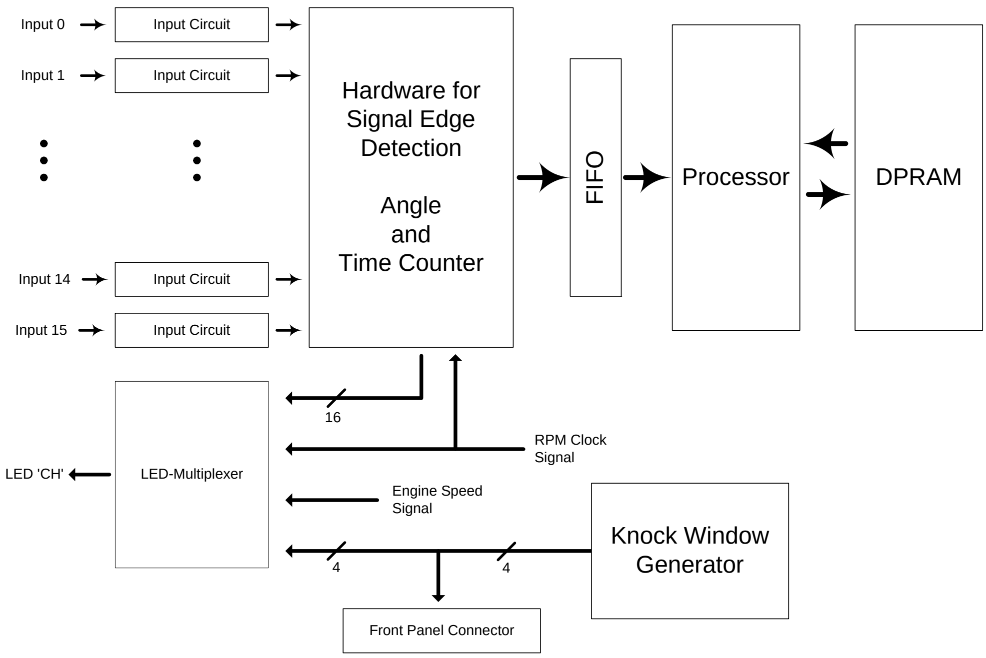


Fig. 8-1 Block Diagram of the ES1334.1 Measurement Board

The firmware downloaded to the processor of the ES1334.1 can now read the generated values from the FIFO buffer and thus generate information on when and at what angle which input signals have changed and how. In turn, this information is used to generate all measurement values which are of interest to the user, such as injection times, ignition times and duty cycles.

This basic concept of hardware edge recognition with post-connected software conditioning of the measurement values means that the usage factor of the processor depends directly on the number of recognized edges (i.e. the frequencies of the input signals). The advantage of this concept is that the hardware simply acquires all the changes and gives them an angle and time stamp. The evaluation, however, is executed by the software and is therefore relatively easy to change/adapt. The ES1334.1 firmware and RTIO driver offer a wide range of implemented measurement procedures. The concept used also makes it possible to implement other measurement procedures by simply adapting the ES1334.1 firmware and the RTIO driver.

8.1.2 Input Circuit

Fig. 8-2 is a simplified representation of the input circuit of a hardware channel on the ES1334.1 already shown in Fig. 8-1 on page 146. It basically consists of a comparison with a threshold value defined by the user.

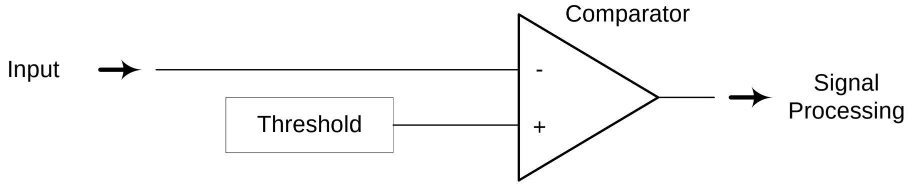


Fig. 8-2 Simplified Representation of the Input Circuit of an ES1334.1 Hardware Channel

This comparison leads to the analog input signal being converted to digital (0/1) information. The threshold value is defined in such a way that all voltages pending at the board input which are greater than this value, are interpreted as "1". All voltages under this threshold value are interpreted as "0".

The comparator thresholds can be set to one of the following values for channels 0 to 7 and 8 to 15 in each case:

- 2.5 V
- (battery voltage A) /2
- (battery voltage B) /2
- external reference voltage

It is often the case in the automotive sector that relatively slow, i.e. low-frequency, signals are used which very often have large rise or fall times at the edges. If this kind of signal is compared to a specified threshold value, there is a danger – particularly if there is noise on the signal line – that the result of the comparison changes backwards and forwards from "1" to "0" a few times with a slow overstepping/undercutting of the threshold. To avoid these unwanted side-effects, the comparison circuits on the ES1334.1 have a hysteresis, i.e. the threshold value for the change from "0" to "1" is slightly higher than the threshold value for the change from "1" to "0". The size of the hysteresis is 1 V (see Fig. 8-3).

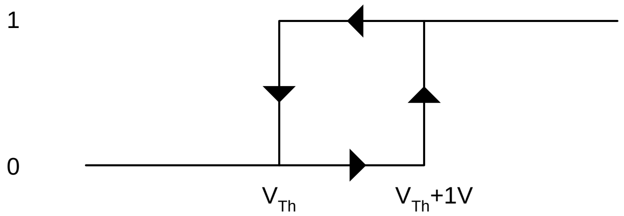


Fig. 8-3 Comparator Hysteresis - V_{Th} is the Threshold Value Set

8.1.3 Measurement Value Acquisition

The ES1334.1 RTIO driver provides the user with 16 freely configurable measurement values (referred to below as software or measurement channels). There is a 1:1 relation between the measurement channels and hardware channels of the ES1334.1, i.e. measurement channel 0 executes measurements on hardware channel 0, measurement channel 1 on hardware channel 1 etc.

The following can be defined for each of the 16 measurement channels completely independently of all others:

- how the pending signal is to be evaluated
- when and how it should be checked for missing pulses (*timeout monitoring*) and
- whether the evaluation should take place continuously or only within a certain range of the crankshaft angle (*angle segmenting*).

For some measurements, you can also determine which pulse is to be measured.

8.1.4 Timeout Monitoring

As already described in the section "Basic Principle of Measurement Value Determination" on page 146, the ES1334.1 is edge-controlled, i.e. the pending external signals are evaluated exclusively at their edges. This, however, means that the generated measurement values returned by the ES1334.1 RTIO driver are also only updated according to edges. A signal which has already been evaluated once by the board would thus continue to return the last valid measurement value generated, even if in the meantime it has completely dropped out, i.e. has no more edges.

The ES1334.1 RTIO driver provides timeout monitoring functionality to ensure that signals which have completely or partially dropped out are interpreted correctly. A test can be carried out at certain intervals for every individual measurement value to see whether it has been updated by new edges at the input signal. This test can take place either in intervals specified by the user or at specific crankshaft angles specified by the user.

If this kind of check shows that no edges have been recognized by the hardware since the last timeout check, the relevant measurement value can be modified in two different ways. It is either set to a specified timeout value or determined depending on the current level (high or low) of the input signal. This is particularly necessary for the correct evaluation of duty cycles (an inactive input results in the measurement values 0.0 or 1.0 – depending on the level of the input signal).

This makes a total of six different timeout monitoring possibilities:

- checking in time intervals of x ms: measurement value in a timeout predefined as value y.
- checking every 720° at x° crankshaft: measurement value in a timeout predefined as value y.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria are fulfilled: measurement value in a timeout predefined as value y.
- checking in time intervals of x ms: measurement value in a timeout depending on the level of the input signal.

- checking every 720° at x° crankshaft: measurement value in a timeout depending on the level of the input signal.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria are fulfilled: measurement value in a timeout depending on the level of the input signal.

8.1.5 Enable Signals for Knock Generators

The ES1334.1 also makes it possible to generate four digital signals with TTL level to activate/deactivate knock generators. These "Knock Enable" signals are speed-synchronous signals which are active within an angle segment defined by the user ("Knock Enable Window"). The definition of the active signal level (active-high or active-low) can also be configured (Fig. 8-4).

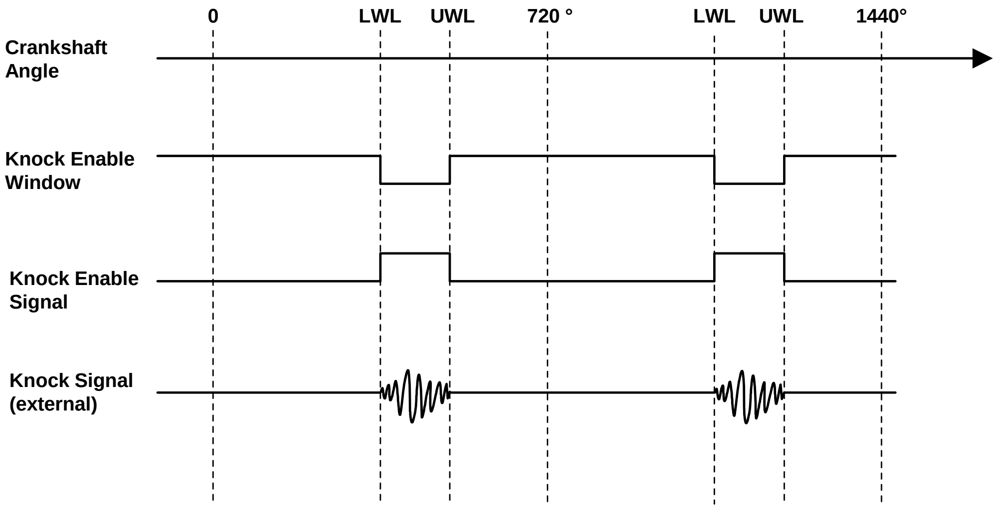


Fig. 8-4 High-Active Knock Enable Signal for Activating and Deactivating an External Knock Generator

8.1.6 The Structure of the ES1334.1 RTIO Tree

The inclusion of the ES1334.1 Measurement Board in an RTIO hardware description starts with the addition of an ES1334-VMI item to an experimental target.

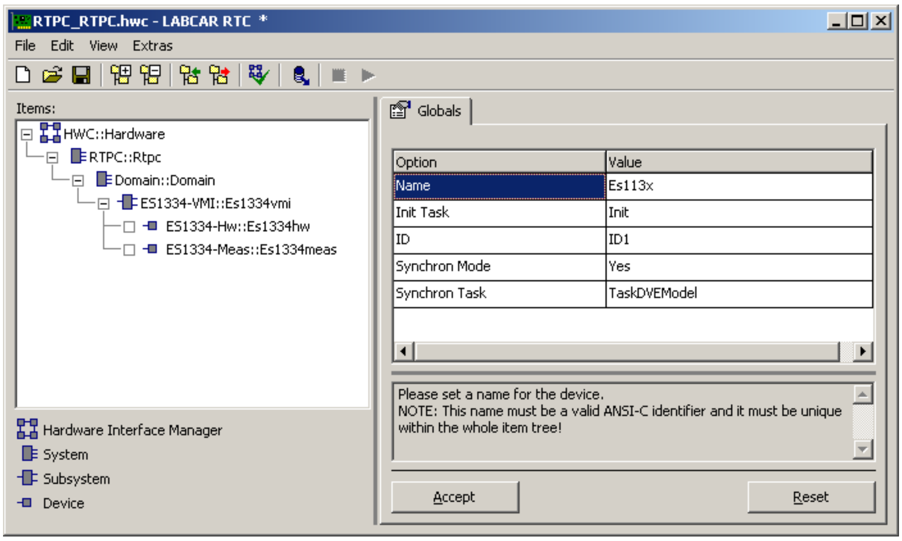


Fig. 8-5 RTIO Hardware Description with ES1334.1 Board

The ES1334.1 board has 16 hardware channels for acquiring and conditioning analog and digital signals plus 4 output signals to activate 4 external knock generators. The 16 hardware channels and the 4 "knock-enable" outputs are configured using the ES1334-Hw device which is assigned to the ES1334-VMI subsystem. In addition to this item, a ES1334-Meas device can be assigned to the ES1334-VMI subsystem. Up to 16 measurements can be specified with each of these items.

Note

Before an ES1334-Meas device item can be added, an ES1334-Hw device must first be added to configure the ES1334.1 *hardware.*

8.2 ES1334-VMI Subsystem

8.2.1 Globals (ES1334-VMI Subsystem)

Fig. 8-6 shows the RTIO parameters of the "Globals" tab.

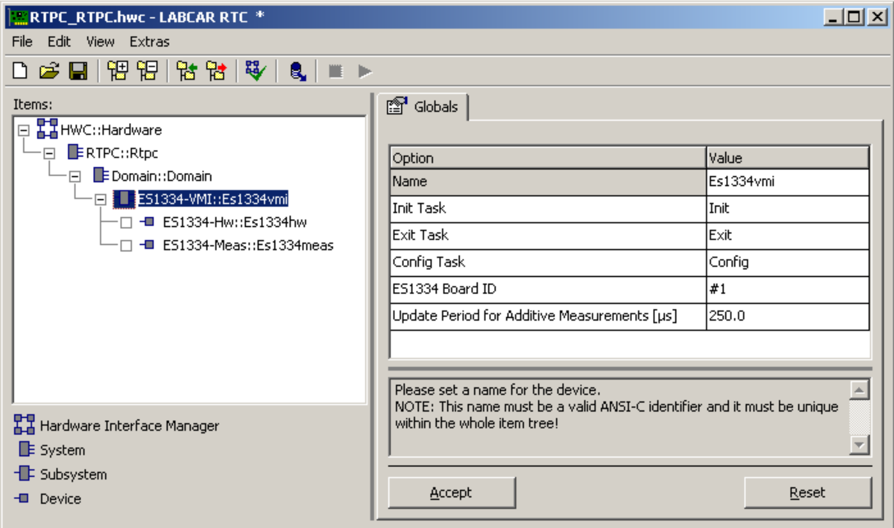


Fig. 8-6 The "Globals" Tab of the ES1334-VMI Subsystem

Note

When the ES1334.1 Measurement Board is integrated into a LABCAR-RTC hardware configuration, both address switches (SW1 and SW2) on the board must be set to "0x00"!

ES1334 Board ID

This option field is used to identify the ES1334.1 board. It establishes the assignment between the RTIO hardware description and the ES1334.1 in the VMEbus chassis, for which this description is valid. The ES1334.1 boards are numbered from left to right starting with 1 in the chassis (left = VMEbus slot no. 1, in ascending order). The number of the ES1334.1 board determined must be entered in the "ES1334 Board ID" list field. This RTIO parameter cannot be edited during runtime.

Up to twelve ES1334.1 boards can be integrated per chassis in the RTIO Editor.

Update Period for Additive Measurements [µs]

This parameter is only of any importance to software and measurement channels which execute additive time measurements in asynchronous mode. It specifies the maximum intervals at which the additive time has to be updated. For more information, refer to "Functional Description: Asynchronous Additive Pulse-Width Measurements" on page 169. The parameter can be configured during runtime of the model.

Tab. 8-1 lists the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
ES1334 Board ID	uint32	No	Identification of the ES1334.1. You can choose from #1 to #12.
Update Period for Additive Measurements [µs]	real32	Yes	Measurement update rate for asynchronous additive measurements in microseconds. Minimum: 50 µs Maximum: 500 ms

* Data type which the RTIO driver uses internally for the parameter

Tab. 8-1 ES1334-VMI Subsystem: Configuration Parameters of the "Globals" Tab

8.3 Hardware Configuration - ES1334-Hw Device

The ES1334-Hw device is used to configure and control the ES1334.1 hardware. Settings which are independent of the individual hardware channels are executed in the "Globals" tab of this item. Channel-specific settings are made in the "Signals" tab.

8.3.1 Globals (ES1334-Hw Device)

Fig. 8-7 shows the RTIO parameters of the "Globals" tab. Tab. 8-2 on page 154 lists the properties of the individual parameters.

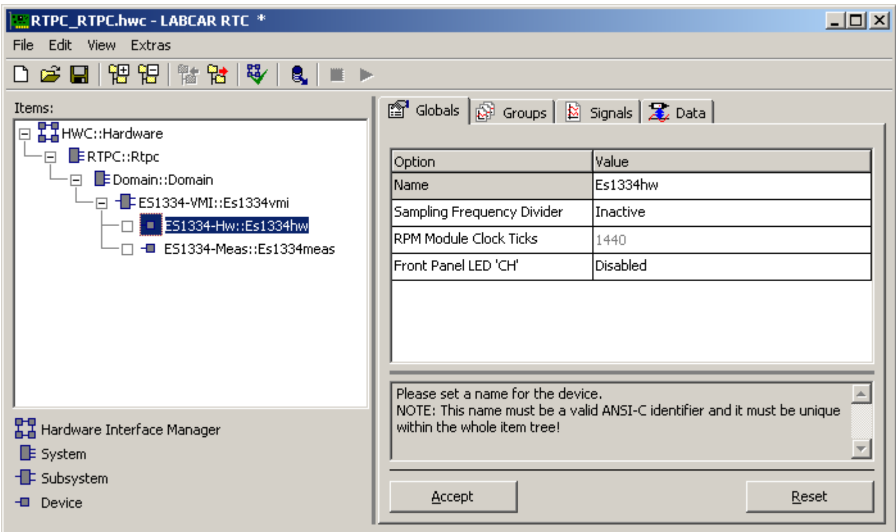


Fig. 8-7 The "Globals" Tab of the ES1334-Hw Device

Sampling Frequency Divider

The ES1334.1 samples the input signals at the hardware channels periodically. The maximum sampling rate is determined by the frequency of the internal clock, which is 4 MHz. The sampling frequency can be reduced with a frequency divider. The frequency divider is configured in the "Sampling Frequency Divider" option field of the "Globals" tab. Frequency division can be disabled or configured to dividers 2, 4, 8, 16 or 32. Please note that only input pulses which last longer than the sampling period can be acquired accurately.

The parameter can be edited online (i.e. during runtime of the model on the experimental target).

RPM Module Clock Ticks

To execute speed-synchronous measurements, the ES1334.1 requires information on the angle position and direction of rotation of the engine.

This information is determined by the ES1334.1 from the angle clock signal supplied on the front panel via the "SYNC" input. The angle clock signal (see Fig. 8-8) is generated by an ES1332.1 Arbitrary Signal Generator Board and makes it possible to synchronize to 0° camshaft and calculate the angle position of the engine.

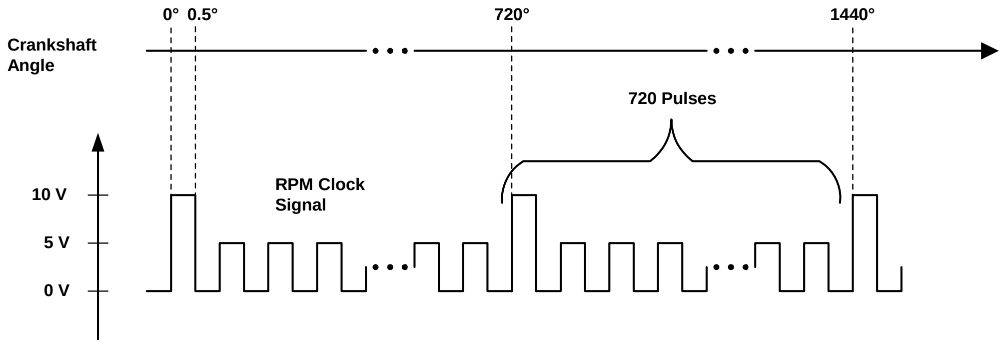


Fig. 8-8 Angle clock signal to synchronize to 0° camshaft and calculate the angle position of the engine

The numeric input field "RPM Module Clock Ticks" is used to define the angular resolution, i.e. the number of angle clocks per camshaft rotation (720° crankshaft). This option is fixed at a value of 1440, i.e. cannot be edited, as the angle clock signal generated by ES1332.1 Arbitrary Signal Generator Board boards generates this number of ticks or edges. The value set, i.e. 1440 angle clocks per camshaft rotation, corresponds to an angular resolution of 0.5 °CA (°CA = degrees crankshaft).

Front Panel "CH" LED

On the front panel of the ES1334.1, there is an LED labeled "CH". The signal level

- on one of the 16 hardware channels or
- on one of the 4 knock enable outputs or
- of the angle clock signal (see Fig. 8-8 on page 153) or
- of the signal for synchronizing to 0° camshaft (see Fig. 8-9)

can be displayed using this LED. If the signal has a low level, the LED is dark; if it has a high level, the LED is light. The definition of low and high level for the 16 input channels comes from the threshold voltage of the input comparator of the relevant channel. With the knock enable outputs and the clock angle signal, the threshold for the distinction of high and low level is 2.5 V; with the signal for synchronizing to 0° camshaft, the threshold value is 7.5 V.

The "Globals" tab has a "Front Panel LED "CH"" option field for the "CH" LED with which the signal is set which addresses the LED or with which the LED can be deactivated. The option field can be edited can be edited online (i.e. during runtime of the model on the experimental target).

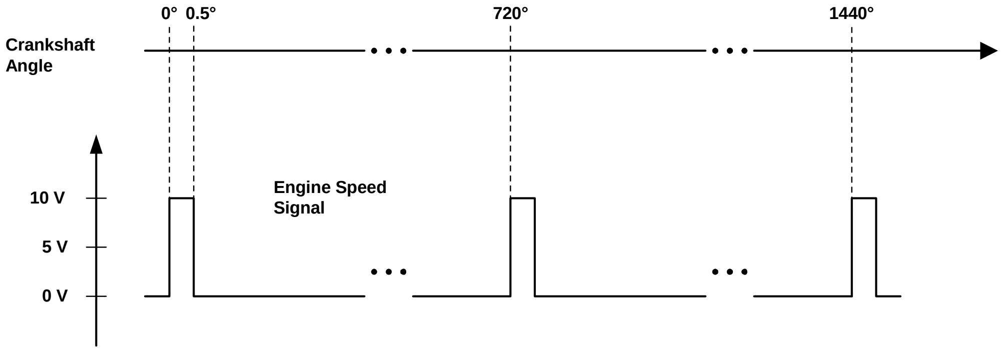


Fig. 8-9 Signal for Synchronizing to 0° Camshaft

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Sampling Frequency Divider	uint8	Yes	Frequency divider for input signal sampling 0: no frequency division (4 MHz) 1: division by 2 (2 MHz) 2: division by 4 (1 MHz) 3: division by 8 (500 kHz) 4: division by 16 (250 kHz) 5: division by 32 (125 kHz)
RPM Module Clock Ticks	uint32	No	Angular resolution Set to 1440 and cannot be edited.
Front Panel LED "CH"	sint8	Yes	Signal or channel which is connected to the front panel LED "CH" or deactivation of the LED. -1: deactivation 0...15: hardware channel #0... #15 16...19: knock-enable-output #0...#3 20: Angle clock signal (RPM Clock) 21: Signal for synchronizing to 0° camshaft (engine speed).
* Data type which the RTIO driver uses internally for the parameter			

Tab. 8-2 ES1334-Hw Device: Configuration Parameters of the "Globals" Tab

8.3.2 Groups (ES1334-Hw Device)

The ES1334-Hw item has three signal groups (Fig. 8-10). The "Control" signal group is intended for future extensions. As it is not currently of any significance, this signal group cannot be assigned any real-time operating system task.

The "Level" signal group is transferred from the ES1334.1 to the experimental target. This signal group contains the current level information of all activated hardware channels.

The "Knock" signal group is transferred from the experimental target to the ES1334.1. This signal group is used to activate/deactivate the knock enable outputs.

ERCOS[EK] operating system tasks must be assigned to both "Level" and "Knock" signal groups. If the level information is evaluated in the simulation model, a task with "Type" "Timer" is assigned to the "Level" signal group. The activation period depends on the dynamic behavior or period duration of the signals to be acquired.

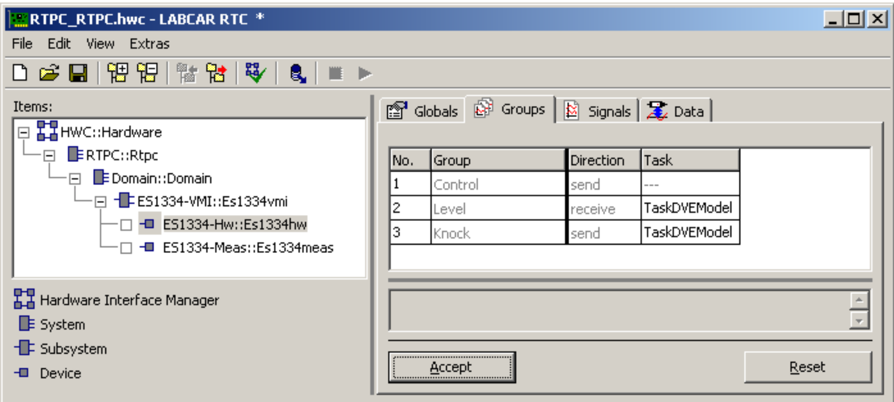


Fig. 8-10 The "Groups" Tab of the ES1334-Hw Device

RTIO Signals of the "Level" Signal Group

The "Level" signal group contains the RTIO signal "LvlBitField_0" which is to be interpreted as bit fields. The bit field has 16 bits which contain coded level information of hardware channels 0 to 15. Tab. 8-3 lists the properties of the RTIO signal.

Note

Only level bits of hardware channels for which edge recognition is activated are valid in the "LvlBitField_0" bit field. Edge recognition on a hardware channel is activated if a measurement is executed on it.

RTIO Signal	Data Type*	Comment / Value Range
LvlBitField_0	uint16	Bit field with level information of hardware channels 0 to 15. Channel 0: LSB (Least Significant Bit) Channel 15: MSB (Most Significant Bit) Bit value 0: low level Bit value 1: high level
* Data type which the RTIO driver uses internally for the parameter		

Tab. 8-3 The RTIO Signals of the "Level" Signal Group

RTIO Signals of the "Knock" Signal Group

The "Knock" signal group consists of 4 RTIO signals with which the 4 knock enable outputs of the board can be activated or deactivated. Tab. 8-4 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
KnckEnable_0 ... KnckEnable_3	bool	Activation/deactivation of the knock enable outputs 0 to 3 0: deactivation 1: activation
* Data type which the RTIO driver uses internally for the parameter		

Tab. 8-4 ES1334-Hw Device: The RTIO Signals of the "Knock" Signal Group

Note

The "Knock" signal group is only written to the DPRAM of the ES1334.1 when the value of one of its RTIO signals has changed.

8.3.3 Signals (ES1334-Hw Device)

The "Signals" tab is used to configure the 16 hardware channels of an ES1334.1. Fig. 8-11 shows the RTIO parameters of the "Signals" tab. Tab. 8-5 on page 158 lists the properties of the individual parameters. All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

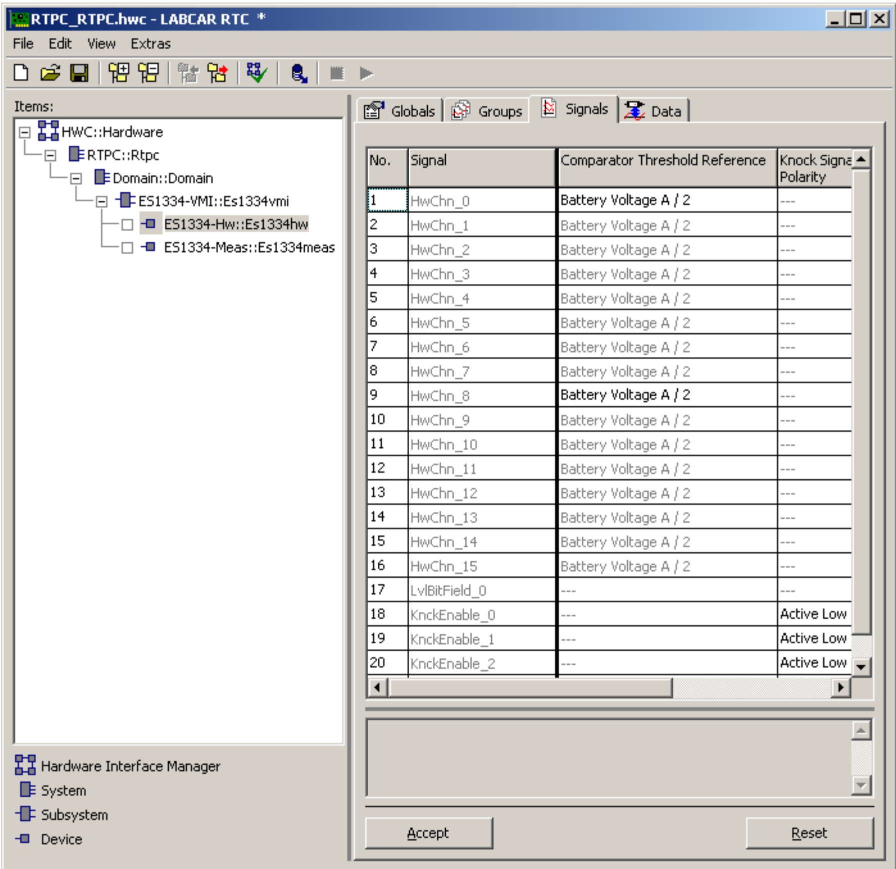


Fig. 8-11 The "Signals" Tab of the ES1334-Hw Item

Comparator Threshold Reference

The "Comparator Threshold Reference" option field is used to specify the threshold voltage of an input comparator. The comparator thresholds can only be set to one of the following values for channels 0 to 7 and 8 to 15 in each case:

- 2.5 V
- (battery voltage A)/2
- (battery voltage B)/2
- external reference voltage

Knock Signal Polarity

Polarity of the "knock enable" signal (low-active or high-active).

Knock Window Lower CA Limit

Lower limit of knock window in °CA.

Knock Window Upper CA Limit

Upper limit of knock window in °CA.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Comparator Threshold Reference	uint8	Yes	Comparator reference 0: 2.5 V 1: (battery voltage A)/2 2: (battery voltage B)/2 3: external reference voltage
Knock Signal Polarity	uint8	Yes	Polarity of "knock enable" signal 0: low-active 1: high-active
Knock Window Lower CA Limit	real32	Yes	Lower limit of knock window in °CA. Value range: -720 °CA to 720 °CA
Knock Window Upper CA Limit	real32	Yes	Upper limit of knock window in °CA. Value range: -720 °CA to 720 °CA

* Data type which the RTIO driver uses internally for the parameter

Tab. 8-5 ES1334-Hw Item: Configuration Parameters of the "Signals" Tab

8.4 Measuring Configuration - ES1334-Meas Device

The ES1334-Meas device is used to specify and configure measurements. It provides 16 software channels (referred to below as measurement channels) which are assigned to the 16 hardware channels in a 1:1 relation, i.e. measurement channel 0 executes measurements on hardware channel 0, measurement channel 1 on hardware channel 1, etc.

8.4.1 Globals (ES1334-Meas Device)

No settings are to be made in the "Globals" tab (Fig. 8-12). In the "Measurement Device ID" option field, a number is assigned to the ES1334-Meas device by the system for identification purposes.

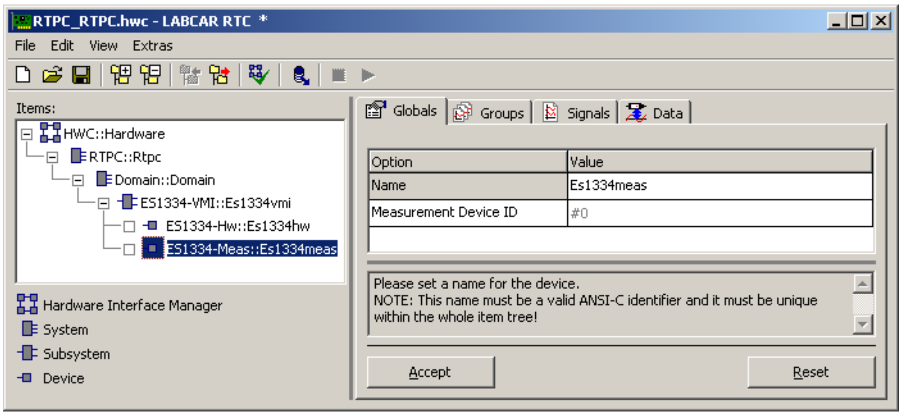


Fig. 8-12 The "Globals" Tab of the ES1334-Meas Item

8.4.2 Groups (ES1334-Meas Device)

The ES1334-Meas device has a signal group (Fig. 8-13) which is transferred to the experimental target by the ES1334.1. It contains all measurement data such as measurement values, trigger and timeout information. An ERCOS^{EK} operating system task with "Type" "Timer" must be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

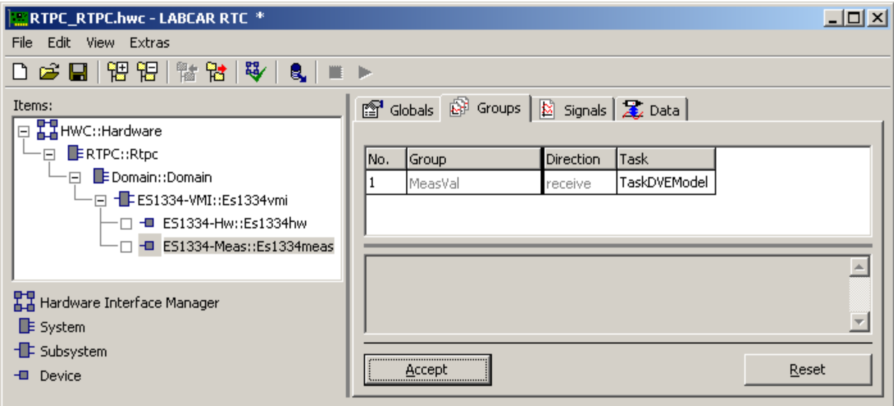


Fig. 8-13 The "Groups" Tab of the ES1334-Meas Item

RTIO Signals of the "MeasVal" Signal Group

The "MeasVal" group consists of 33 RTIO signals. The "TriggerBitField_0" signal must be interpreted as a bit field. It has 16 bits. The trigger or update data of the 16 measurement channels is coded in this bit field, i.e. it shows which measurement values have been updated since the last activation of the read process for the "MeasVal" signal group. If a bit is set, the measurement value of the relevant measurement channel has been newly determined. It is of no importance whether the measurement value was updated due to a timeout or a regular measurement value calculation; the update bit of the measurement channel is set in both cases.

The RTIO signals "MeasVal_0" to "MeasVal_15" contain the measurement values of the 16 measurement channels. These are either measurement values determined in the normal way or, in the case of a timeout, the timeout value intended for this case. If a measurement channel is not used, -8888.0 is assigned to the relevant measurement value. The physical unit of the measurement values depends on the measurement procedure:

- time measurements (time stamp, (additive) pulse duration measurements, period duration) are issued in microseconds
- frequency measurements are in hertz
- angle measurements and angle stamp are specified in degrees crankshaft (°CA)
- all other measurements (duty cycles, pulse count, level measurements) are dimensionless

The RTIO signals "Tout_0" to "Tout_15" contain the results of the timeout monitoring for the relevant measurement channel.

Tab. 8-6 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
TriggerBitField_0	uint16	Bit field with trigger information of the 16 measurement channels. Measurement channel 0: LSB Measurement channel 15: MSB Bit value 0: measurement value is unchanged Bit value 1: measurement value has been updated
MeasVal_0 ... MeasVal_15	real64	Measurement value If the measurement channel is not used, -8888.0 is issued as a value. Physical unit of the measurement value: Time measurements are in microseconds Frequency measurements are in hertz Angle measurements are in °CA
Tout_0 ... Tout_15	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive
* Data type which the RTIO driver uses internally for the parameter		

Tab. 8-6 ES1334-Meas Device: The RTIO Signals of the "MeasVal" Signal Group

8.4.3 Signals (ES1334-Meas Device)

The "Signals" tab is used to configure the 16 measurement channels of an ES1334-Meas device. Fig. 8-14 shows the RTIO parameters of the "Signals" tab. Tab. 8-7 on page 164 lists the properties of the individual parameters. All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

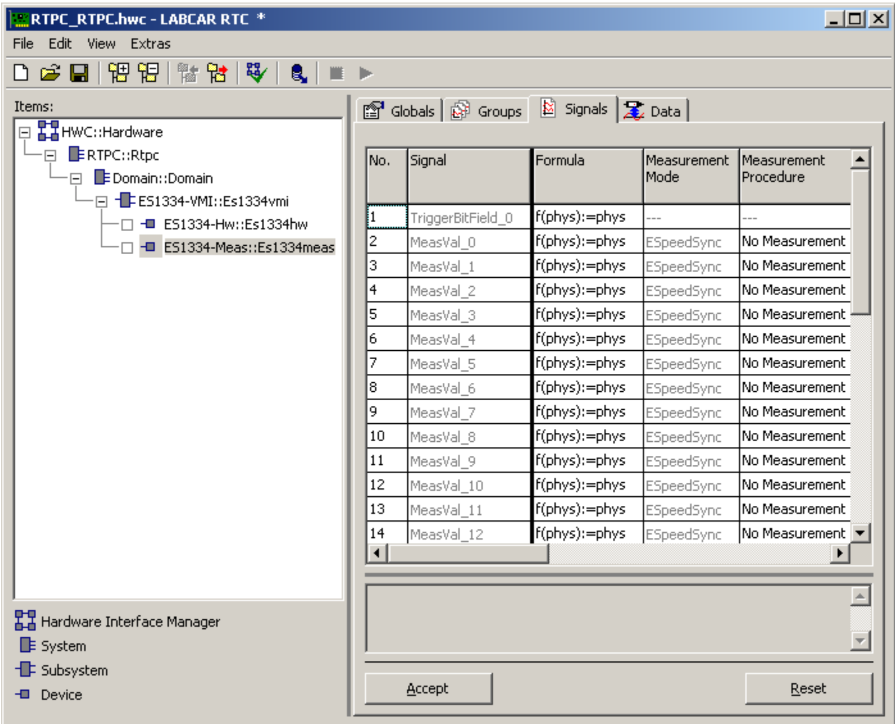


Fig. 8-14 The "Signals" Tab of the ES1334-Meas Item

Measurement Mode

The "Measurement Mode" option field defines the mode in which measuring takes place. There are two options:

- "ESpeedSync": execution of speed-synchronous or angle-synchronous measurements. The execution of speed-synchronous measurements makes it necessary for the ES1334.1 to be supplied with an angle clock signal via the "SYNC" input on the front panel.
- "Asynchronous": execution of asynchronous measurements.

Measurement Procedure

This is where the measurement procedure is selected. Section 8.5 on page 166 - section 8.5.7 on page 179 contain detailed descriptions and configuration guidelines for the individual procedures. The notation - - / - - denotes evaluation at rising edges whereas - - \ - - denotes evaluation at falling edges.

If "No Measurement" is set in the list box, no measurement is executed and edge recognition on the assigned hardware channel is deactivated.

Note

To avoid unnecessary computing time, measurement channels which are not required should be deactivated.

Timeout Monitoring

Definition of the timeout monitoring for the relevant measurement channel (see "Timeout Monitoring" on page 148).

The following settings are possible:

- "Inactive": no timeout monitoring.
- "Intvl Predef": timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "Intvl InpDep": timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout depends on the level of the input signal.
- "CS Angle Predef": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "CS Angle InpDep": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout depends on the level of the input signal.
- "Intvl & CS Angle Predef": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field and timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. A timeout is recognized when one or both timeout criteria are violated. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "Intvl & CS Angle InpDep": like "Intvl & CS Angle Predef", but the measurement value in a timeout depends on the level of the input signal.

Default Timeout Value

The value set in this numeric input field is issued as the measurement value in a timeout if one of the "Intvl Predef", "CS Angle Predef" or "Intvl & CS Angle Predef" timeout monitoring modes is set.

Timeout Interval [ms]

Time between two timeout checks in milliseconds. Only relevant for time-based timeout monitoring modes ("Intvl Predef", "Intvl InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Timeout Check Point

Crankshaft angle at which timeout checks are executed. Only relevant for angle-based timeout monitoring modes ("CS Angle Predef", "CS Angle InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Lower Limit

Lower limit of the angle window released for measurement in degrees crankshaft. Only relevant in the speed-synchronous measurement mode "ESpeedSync".

CS Angle Upper Limit

Upper limit of the angle window released for measurement in degrees crankshaft. Only relevant in the speed-synchronous measurement mode "ESpeedSync". The upper angle window limit is also used for additive time measurements and pulse counts to transfer the determined measurement value to the DPRAM of the ES1334.1.

CS Angle Reference

Angle reference point in degrees crankshaft to which all angle measurements refer. This parameter is of no importance in all other measurements.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Measurement Mode	uint8	Yes	Measurement mode 0: "Asynchronous": asynchronous measurements 1: "ESpeedSync": speed- or angle-synchronous measurements
Measurement Procedure	sint32	Yes	Selection of measurement procedure. ("measurement procedure" param. value) Deactivation of measurement channel: -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	Yes	Timeout monitoring. 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: Configuration Parameters of the "Signals" Tab

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Default Time-out Value	real32	Yes	Measurement value for timeout. Only relevant in "... Predef" modes for timeout monitoring.
Timeout Interval [ms]	real32	Yes	Time between two timeout checks in milliseconds. Value range: ≥ 1.0 ms Only relevant in "Intvl ..." modes for time-out monitoring.
CS Angle Timeout Check Point	real32	Yes	Crankshaft angle at which timeout checks are executed. Value range: -720 °CA to 720 °CA Only relevant in "CS Angle ..." modes for timeout monitoring.
CS Angle Lower Limit	real32	Yes	Lower limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Upper Limit	real32	Yes	Upper limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Reference	real32	Yes	Angle reference point in degrees crank-shaft to which all angle measurements refer. This parameter is of no importance in all other measurements. Value range: -720 °CA to 720 °CA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 8-7 ES1334-Meas Device: Configuration Parameters of the "Signals" Tab

8.5 Measurement Procedures

8.5.1 Pulse-Width Measurements

This section describes the properties of pulse-width measurements. A distinction is made between "pulse-qualified" and "edge-qualified" measurements. Each of the following measurements is described:

Pulse-Qualified Pulse-Width Measurements

- "Hightime (Pulse Qual.) [µs]"
- "Lowtime (Pulse Qual.) [µs]"

Edge-Qualified Pulse-Width Measurements

- "Hightime (Edge Qual.) [µs]"
- "Lowtime (Edge Qual.) [µs]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode.
- timeout monitoring

Functional Description: Speed-Synchronous Pulse-Width Measurements

In *pulse-qualified measurements*, only those pulses are measured whose opening and closing edges are within one and the same angle window (valid pulse). Fig. 8-15 illustrates how pulse-qualified measuring works using a low-active pulse-width measurement (measurement procedure "Lowtime (Pulse Qual.) [µs]").

In *edge-qualified measurements,* only those pulses are measured whose opening or closing edge is within a (any) angle window (valid pulse). Pulses which have both edges outside an angle window are not measured or taken into consideration. Fig. 8-16 illustrates how edge-qualified measuring works using a low-active pulse-width measurement ("Lowtime (Edge Qual.) [µs]").

Every valid pulse is measured; the measured pulse width is transferred to the DPRAM of the ES1334.1 on the closing edge of the pulse.

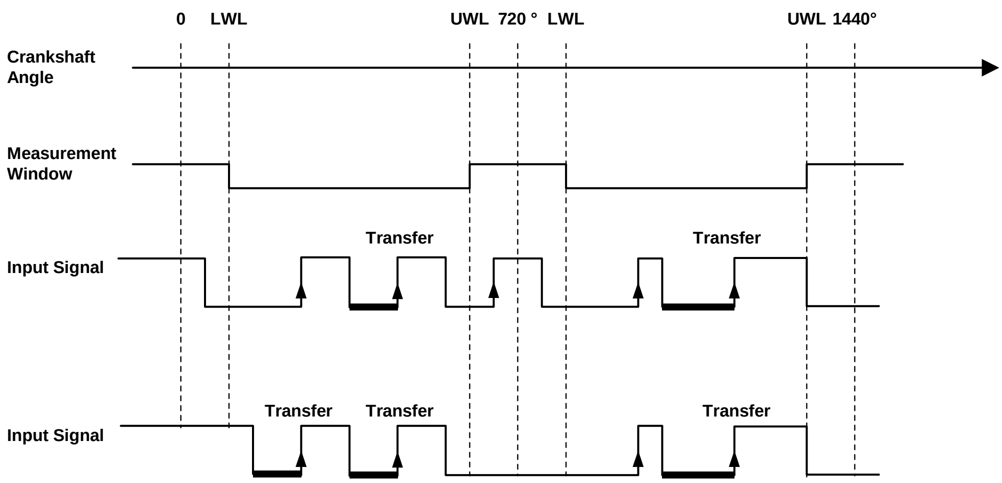


Fig. 8-15 Illustration of how a pulse-qualified low-active pulse-width measurement (measurement procedure "Lowtime (Pulse Qual.) [µs]") works. Measured pulses are shown by a thicker line.

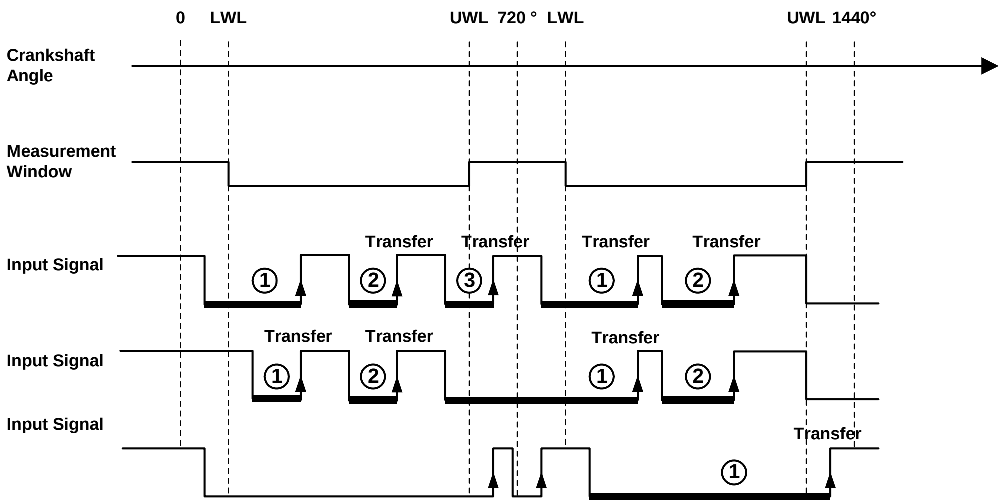


Fig. 8-16 Illustration of how an edge-qualified low-active pulse-width measurement (measurement procedure "Lowtime (Edge Qual.) [µs]") works. Measured pulses are shown by a thicker line. The assignment of the pulses to the angle windows and the counting of the pulses within an angle window is also shown.

Functional Description: Asynchronous Pulse-Width Measurements

With asynchronous pulse-width measurements, it is not possible or not necessary to make a distinction between *pulse-qualified* and *edge-qualified measurements* as there is no window functionality for selecting pulses. The pulse-qualified pulse-width measurements have exactly the same function in asynchronous mode as the edge-qualified measurements.

The pulse width of every pulse with an active level is measured. The measurement value is transferred to the DPRAM of the ES1334.1 on the closing edge of the measured pulse.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 8-8.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	- If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level is active when the timeout check takes place, the length of time since the last opening edge of the signal is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-8 Pulse-width measurements: measurement value in a timeout with different timeout monitoring procedures

8.5.2 Additive Pulse-Width Measurements

This section describes the properties of additive pulse-width measurements. The functioning of the following measurements is described:

- "Additive Hightime [µs]"
- "Additive Lowtime [µs]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description: Speed-Synchronous Additive Pulse-Width Measurements

The additive time results from the total of all time segments within an angle window in which the signal is active, regardless of whether the opening or closing edges of the pulses are inside or outside the angle window. Fig. 8-17 shows an example of measurement value calculation with an additive lowtime measurement.

The additive time is transferred to the DPRAM of the ES1334.1 on the upper angle window limit.

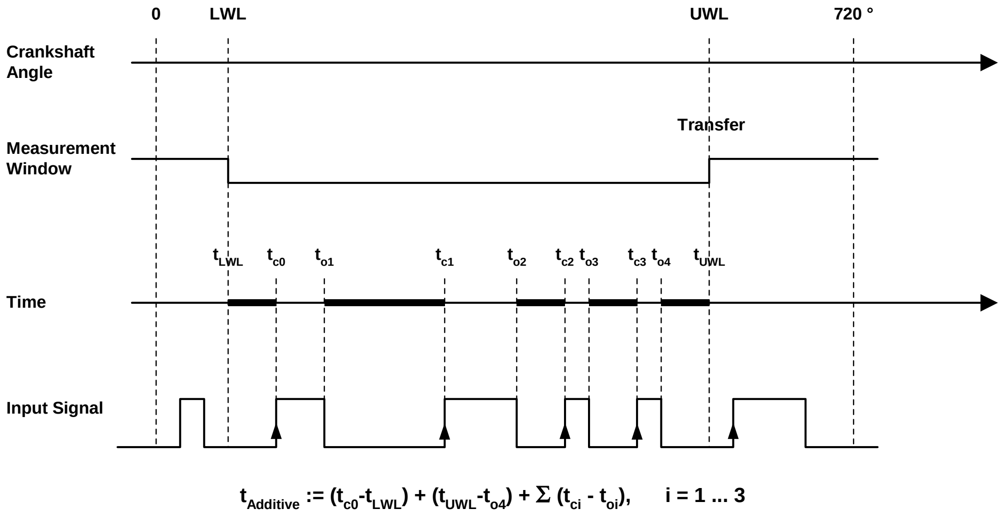


Fig. 8-17 Illustration of how a speed-synchronous additive lowtime measurement (measurement procedure "Additive Lowtime [µs]") works. The additive time is the total of the line segments shown in bold face.

Functional Description: Asynchronous Additive Pulse-Width Measurements

With asynchronous additive pulse-width measurements, the time in which the signal to be measured is active is totalled. The total in the DPRAM is updated at every closing pulse edge and at the latest after an updating period defined by the user. Fig. 8-18 shows when the additive time is updated using an additive low-time measurement. The update period is set by the user in the "Update Period for Additive Measurements [µs]" parameter in the "Globals" tab of the ES1334-VMI item. The totalled time is reset to 0 if the ES1334.1 is initialized or configured.

Note

The additive time is acquired modulo 2^{52} µs, i.e. after (2^{52} - 1) µs (corresponds to approx. 143 years), the additive time returns to 0. (2^{52} - 1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

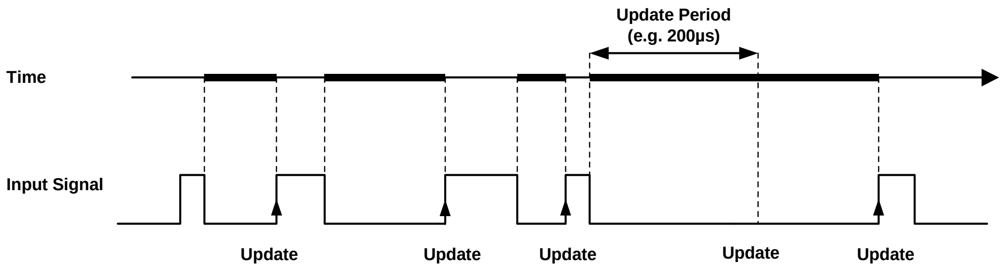


Fig. 8-18 Illustration of how an asynchronous additive lowtime measurement (measurement procedure "Additive Lowtime [µs]") works. The additive time results from the time segments shown in bold face. The times at which the total is updated in the DPRAM are also shown.

Timeout Monitoring

Tab. 8-9 lists the condition of when a timeout is initiated for different timeout monitoring procedures. The measurement value issued in a timeout is specified in Tab. 8-10.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "Intvl Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel *and* the signal has been inactive since the last timeout check. In speed-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"CS Angle InpDep" "CS Angle Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel since the last timeout check. In speed-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"Intvl & CS Angle InpDep" "Intvl & CS Angle Predef"	The timeout condition for this monitoring procedure results from an OR operation of the timeout condition of the "Intvl InpDep" and "Intvl Predef" procedures with the timeout condition of the "CS Angle InpDep" and "CS Angle Predef" procedures.

Tab. 8-9 Additive pulse-width measurements: timeout condition for different timeout monitoring procedures

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	*Asynchronous measurement mode:* The additive time determined up to the point where a check is made for a timeout is issued as measurement value in a timeout. *Speed-synchronous measurement mode:* 0 is issued as measurement value in a timeout.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-10 Additive pulse-width measurements: measurement value in a timeout with different timeout monitoring procedures

8.5.3 Frequency and Cycle Time Measurements

The ES1334.1 makes it possible to measure frequencies and cycle times at rising and falling signal edges. The user specifies the edge at which the measurement value is calculated by selecting a measurement procedure. The following measurement procedures are available:

- "Cycle Time --/-- [µs]"
- "Cycle Time --\-- [µs]"
- "Frequency --/-- [Hz]"
- "Frequency --\-- [Hz]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

In asynchronous measurement mode, measurement values are calculated at every active signal edge. In speed-synchronous (angle-segmented) measurement mode, measurement values are calculated at every active signal edge within an angle window. Fig. 8-19 shows the calculation of measurement values with angle-segmented frequency or cycle time measurement at rising signal edges.

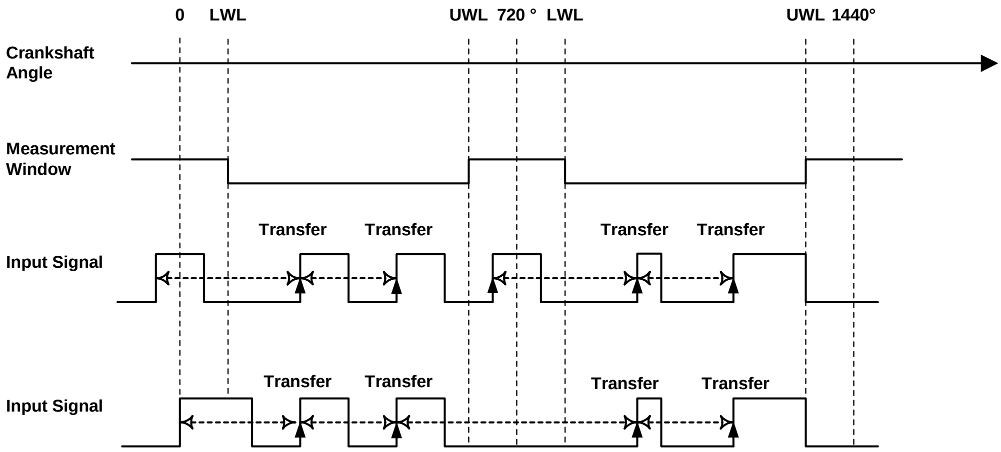


Fig. 8-19 Illustration of measurement value calculation in angle-segmented frequency or cycle time measurement at rising signal edges

Note

If the entire angle range of 720° CA is released in an angle-segmented cycle time or frequency measurement, this measurement behaves like the relevant measurement in asynchronous mode.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 8-11.

Timeout Monitoring	**Measurement Value in a Timeout**
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	*Cycle time measurements:* The length of time between the time the last active edge of the input signal occurred and the time at which a timeout check was executed is issued as measurement value (see Fig. 8-20). *Frequency measurements:* Measurement value calculation in a timeout takes place as for cycle time measurements but a resulting frequency is determined by inversion from the duration determined.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-11 Frequency and cycle time measurements: measurement value in a timeout with different timeout monitoring procedures

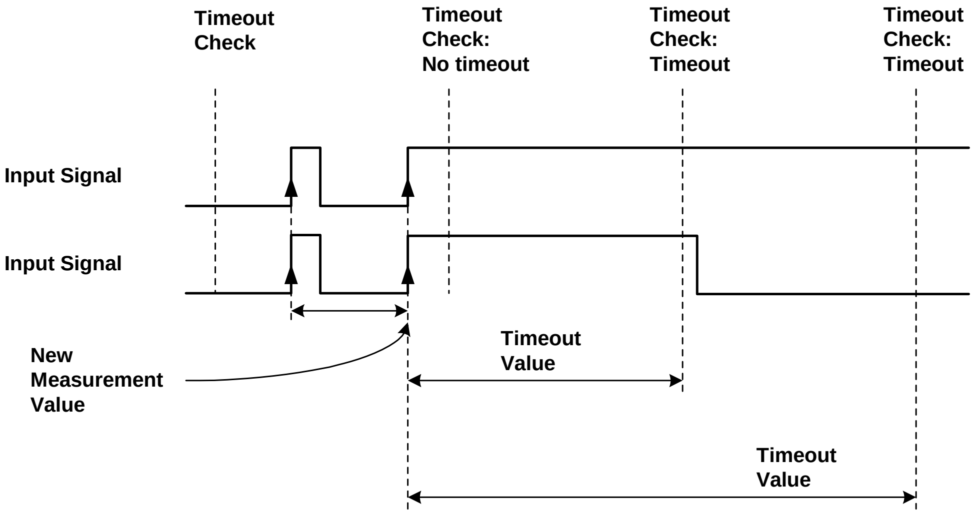


Fig. 8-20 Frequency and cycle time measurements at rising edges: illustration of measurement value calculation in a timeout (the measurement value is referred to in the figure as "Timeout Value") in the "... InpDep" timeout monitoring procedure

8.5.4 Duty Cycle Measurements

Duty cycles (e.g. of PWM signals) can be determined in various ways with the ES1334.1. With duty cycle measurements, a distinction is made between measurements which set the high-phase of a signal in relation to the low-phase (or vice versa) and measurements which set the high-phase (or low-phase) of a signal in relation to the cycle duration.

The first type of duty cycle measurement is referred to below as duty cycle $^{P/P}$; "P/P" shows that two pulse durations are set in relation to one another.

The second type of duty cycle measurement is referred to below as duty cycle $^{P/C}$; "P/C" shows that a pulse duration is set in relation to the cycle time of the signal.

$$\text{Duty Cycle}^{P/P}: \frac{L}{H}, \frac{H}{L} \qquad 0 \le \text{Duty Cycle}^{P/P} \le \infty$$

$$\text{Duty Cycle}^{P/C}: \frac{L}{L+H}, \frac{H}{L+H} \qquad 0 \le \text{Duty Cycle}^{P/C} \le 1$$

The user specifies the edge at which the measurement value is calculated by selecting the measurement procedure. The following "P/P" duty cycle measurements are available:

- "Duty Factor L/H --/--"
- "Duty Factor L/H --\--"
- "Duty Factor H/L --/--"
- "Duty Factor H/L --\--"

The following "P/C" duty cycle measurements are available:

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

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

As far as measurement value calculation in asynchronous and speed-synchronous (angle-segmented) measurement mode is concerned, exactly the same is true of duty cycle measurements as was described for frequency and cycle time measurements in section 8.5.3 on page 171.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 8-12.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	Duty Cycle $^{P/P}$: $\frac{L}{H}$, $\frac{H}{L}$ The measurement value is 0.0 if the signal level is "high" for L/H measurements or "low" for H/L measurements when the timeout check is carried out. The measurement value is MAXREAL32 (3.40282347 x10^{38}) if the signal level is "low" with L/H measurements or "high" with H/L measurements when the timeout check is carried out. Duty Cycle $^{P/C}$: $\frac{L}{L+H}$, $\frac{H}{L+H}$ The measurement value is 0.0 if the signal level is "high" with L/(L+H) measurements or "low" with H/(L+H) measurements when the timeout check is carried out. The measurement value is 1.0 if the signal level is "low" with L/(L+H) measurements or "high" with H/(L+H) measurements when the timeout check is carried out.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-12 Duty cycle measurements: measurement value in a timeout with different timeout monitoring procedures

8.5.5 Measuring Edges: Angle Stamp

Functional Description

The following measurement procedures measure the crankshaft angle of active signal edges on the assigned hardware channel in relation to the reference angle set.

- "Rising Edge --/-- [deg]"
- "Falling Edge --\-- [deg]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode

- "CS Angle Reference": defines the reference angle of the measurement
- timeout monitoring

The crankshaft angle of an edge provided by the hardware is always in the range [0° CA, 720° CA[. In *speed-synchronous measurement mode,* the measured angle is mapped into the range [LWL, LWL + 720°CA[by the ES1334.1 firmware and then the difference is calculated with the reference angle. This difference is the measurement value provided by the measurement; it is positive when the mapped angle is smaller than the reference angle (cf. Fig. 8-21).

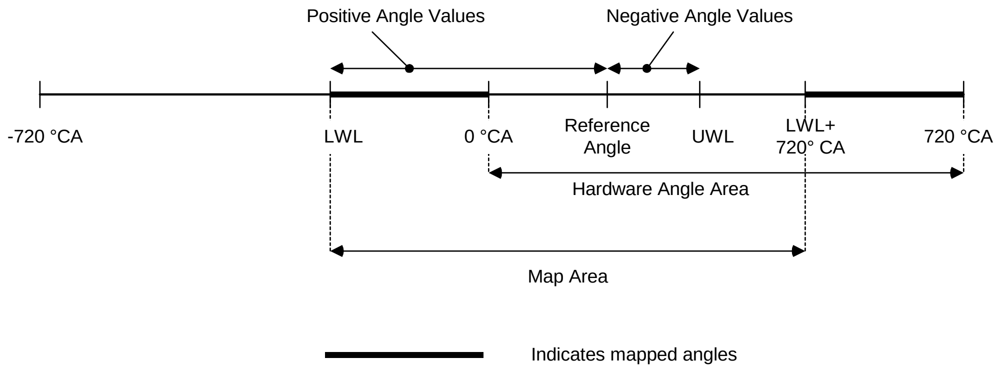


Fig. 8-21 How angle measurement works in speed-synchronous measurement mode

As the reference angle can be in the range [-720° CA, 720° CA], the measurement functions return angle values from the range]-1440 °CA, 1440 °CA] in speed-synchronous measurement mode. In *asynchronous measurement mode,* there is no mapping of the crankshaft angle provided by the hardware which means that the angle values are in the range]-1440 °CA, 720° CA] in this measurement mode.

In *speed-synchronous measurement mode* every active edge within one angle window is measured. In *asynchronous measurement mode* the crankshaft angle of every active edge is measured.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 8-13.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the low level for a measurement at rising edges and the high level for a measurement at falling edges. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the high level for a measurement at rising edges and the low level for a measurement at falling edges.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-13 Angle measurements: measurement value in a timeout with different timeout monitoring procedures

8.5.6 Pulse Count

This section describes the properties of the following pulse-count measurement procedures:

- "Number of Low-Pulses"
- "Number of High-Pulses"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

The "Number of Low-Pulses" and "Number of High-Pulses" measurement procedures acquire the number of valid active pulses within an angle window in *speed-synchronous measurement mode*. Valid pulses are pulses which are in the same angle window with both opening and closing edge. The total calculated is transferred to the DPRAM at the upper angle window limit (Fig. 8-22).

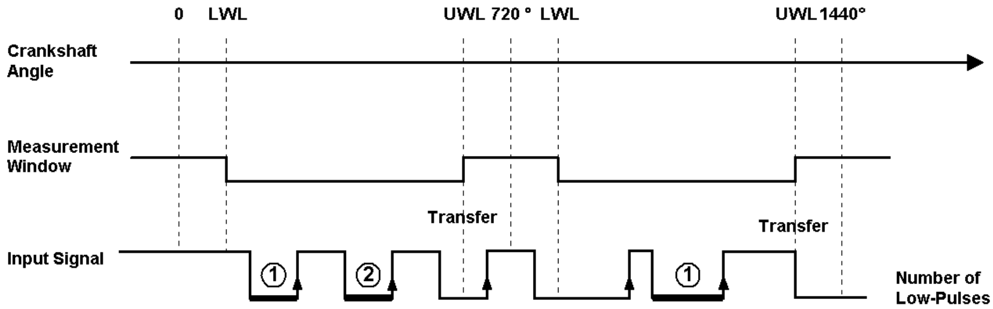


Fig. 8-22 Illustration of how speed-synchronous measurement procedures for pulse count (measurement procedure "Number of Low-Pulses") work. Pulses which have been counted are shown with thicker lines. The total is transferred to the DPRAM at the upper angle window limit.

In *asynchronous measurement mode* the total number of active pulses since the last initialization or configuration of the ES1334.1 is measured. The total is updated at each closing pulse edge.

Note

The pulse count takes place with modulo 2^{52}, i.e. after (2^{52} - 1) pulses, the total returns to 0. (2^{52} - 1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

Timeout Monitoring

A timeout is initiated if no valid active pulse has occurred since the last timeout check. In speed-synchronous measurement mode, a valid active pulse is an active pulse whose opening and closing edges are within one and the same angle window. In asynchronous measurement mode, every active pulse is a valid, active pulse. The measurement value issued in a timeout is specified in Tab. 8-14.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the high level when low pulses are counted and the low level when high pulses are counted. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the low level when low pulses are counted and the high level when high pulses are counted.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-14 Pulse count: measurement value in a timeout with different timeout monitoring procedures

8.5.7 Level Measurement

Functional Description

The measurement procedures

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

are used to acquire the active level states on a hardware channel. An active level is signalized by the measurement value "one" (1); an inactive level is shown by the measurement value "zero" (0). In *asynchronous measurement mode,* the measurement value is updated at each edge of the input signal. In *speed-synchronous measurement mode,* updating only takes place at signal edges which occur within the angle window.

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 8-15.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 8-15 Level measurements: measurement value in a timeout with different timeout monitoring procedures

9 ES1334.2 Measurement Board

Definitions

Active signal edge:

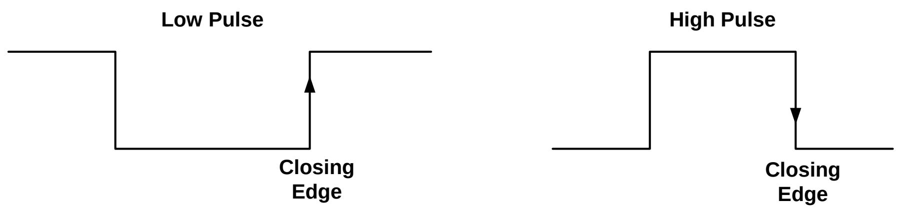
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing pulse edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.



Inactive signal:

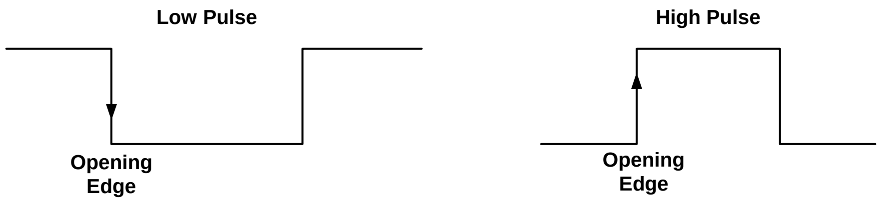
A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

LWL:

Stands for **L**ower **W**indow **L**imit and refers to the lower limit of the angle window in angle-synchronous measurements.

Opening pulse edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



UWL:

Stands for **U**pper **W**indow **L**imit and refers to the upper limit of the angle window in angle-synchronous measurements.

9.1 Functional Description

9.1.1 Basic Principle of Measurement Value Determination

To ensure the greatest possible flexibility in the evaluation of pending input signals, the ES1334.2 is based on the following principle:

First of all, each input signal is conditioned separately (analogously), then compared with a threshold value (which can be set using the software) to convert the analog input signal to digital 0/1 or inactive/active information. The binary level information of the signals obtained in this way at the 16 input channels is continuously checked for changes by a hardware circuit, i.e. "edges" are recognized on one or more input bits. If at least one input signal changes from active to inactive or vice versa, the circuit saves the current statuses of the 16 inputs and calls the current values of two integrated counters which specify the current time and the current crankshaft angle. This information is transferred by the hardware to an FIFO buffer.

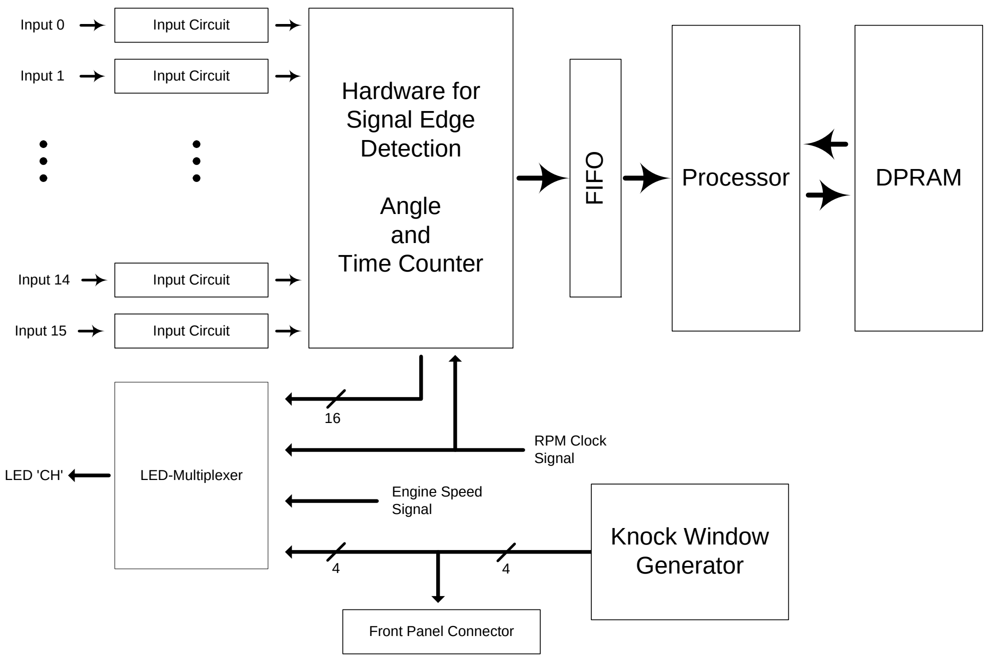


Fig. 9-1 Block Diagram of the ES1334.2 Measurement Board

The processor firmware of the ES1334.2 can now read the generated values from the FIFO buffer and thus generate information on when and at what angle which input signals have changed and how. In turn, this information is used to generate all measurement values which are of interest to the user, such as injection times, ignition times and duty cycles.

This basic concept of hardware edge recognition with post-connected software conditioning of the measurement values means that the usage factor of the processor depends directly on the number of recognized edges (i.e. the frequencies of the input signals). The advantage of this concept is that the hardware simply acquires all the changes and gives them an angle and time stamp. The evaluation, however, is executed by the software and is therefore relatively easy to change/adapt. The ES1334.2 firmware and RTIO driver offer a wide range of implemented measurement procedures (cf. page 204 ff.). The concept used also makes it possible to implement other measurement procedures by simply adapting the ES1334.2 firmware and the RTIO driver.

9.1.2 Input Circuit

Fig. 9-2 is a simplified representation of the input circuit of a hardware channel on the ES1334.2 already shown in Fig. 9-1 on page 182. It basically consists of a comparison with a threshold value defined by the user.

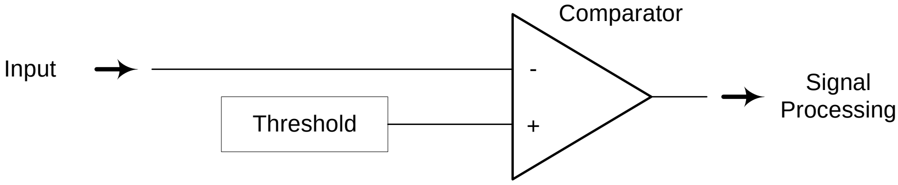


Fig. 9-2 Simplified Representation of the Input Circuit of an ES1334.2 Hardware Channel

This comparison leads to the analog input signal being converted to digital (0/1) information. The threshold value is defined in such a way that all voltages pending at the board input which are greater than this value, are interpreted as "1". All voltages under this threshold value are interpreted as "0".

The comparator thresholds can be set to one of the following values for channels 0 to 7 and 8 to 15 in each case:

- 2.5 V
- (battery voltage A) /2
- (battery voltage B) /2
- external reference voltage

It is often the case in the automotive sector that relatively slow, i.e. low-frequency, signals are used which very often have large rise or fall times at the edges. If this kind of signal is compared to a specified threshold value, there is a danger – particularly if there is noise on the signal line – that the result of the comparison changes backwards and forwards from "1" to "0" a few times with a slow overstepping/undercutting of the threshold. To avoid these unwanted side-effects, the comparison circuits on the ES1334.2 have a hysteresis, i.e. the threshold value for the change from "0" to "1" is slightly higher than the threshold value for the change from "1" to "0". The size of the hysteresis is 1 V (see Fig. 9-3).

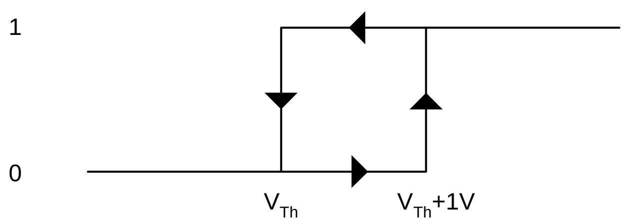


Fig. 9-3 Comparator Hysteresis. V_{Th} is the Threshold Value Set

9.1.3 Measurement Value Acquisition

The ES1334.2 RTIO driver provides the user with up to 64 freely configurable measurement values (referred to below as software or measurement channels) with each individual measurement value being able to depend on any hardware channel. The ES1334.2 RTIO driver therefore does not assign the hardware channels on the one hand and the measurement channel on the other, i.e. when using a lab car for several projects, it is not absolutely necessary always to connect signals with the same meaning to the same hardware channels of the ES1334.2.

In addition, the 64 measurement channels provide the user with four times as many output values as the board has hardware channels. This makes it easy to acquire several measurement values from one single input signal, such as the frequency and duty cycle of a PWM signal. Doubly wiring one and the same signal on different channels of the board is thus superfluous.

The following can be defined for each of the measurement channels (in addition to the hardware channel it is based on) completely independently of all other measurement values:

- how the pending signal is to be evaluated
- when and how it should be checked for missing pulses (timeout monitoring) and
- whether the evaluation should take place continuously or only within a certain range of the crankshaft angle (angle segmenting).

For some measurements, you can also determine which pulse is to be measured.

9.1.4 Timeout Monitoring

As already described, the ES1334.2 is edge-controlled, i.e. the pending external signals are evaluated exclusively at their edges. This, however, means that the generated measurement values returned by the ES1334.2 RTIO driver are also only updated according to edges. A signal which has already been evaluated once by the board would thus continue to return the last valid measurement value generated, even if in the meantime it has completely dropped out, i.e. has no more edges.

The ES1334.2 RTIO driver provides timeout monitoring functionality to ensure that signals which have completely or partially dropped out are interpreted correctly. A test can be carried out at certain intervals for every individual measurement value to see whether it has been updated by new edges at the input signal. This test can take place either in intervals specified by the user or at specific crankshaft angles specified by the user.

If this kind of check shows that no edges have been recognized by the hardware since the last timeout check, the relevant measurement value can be modified in two different ways. It is either set to a specified timeout value or determined depending on the current level (high or low) of the input signal. This is particularly necessary for the correct evaluation of duty cycles (an inactive input results in the measurement values 0.0 or 1.0 – depending on the level of the input signal).

This makes a total of six different timeout monitoring possibilities:

- checking in time intervals of x ms: measurement value in a timeout predefined as value y.

- checking every 720° at x° crankshaft: measurement value in a timeout predefined as value y.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria are fulfilled: measurement value in a timeout predefined as value y.
- checking in time intervals of x ms: measurement value in a timeout depending on the level of the input signal.
- checking every 720° at x° crankshaft: measurement value in a timeout depending on the level of the input signal.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria are fulfilled- The measurement value in a timeout depends on the level of the input signal.

9.1.5 Enable Signals for Knock Generators

The ES1334.2 also makes it possible to generate four digital signals with TTL level to activate/deactivate knock generators. These "Knock Enable" signals are angle-synchronous signals which are active within an angle segment defined by the user (knock window) (Fig. 9-4). The active signal level (active-high or active-low) of the "Knock Enable" signals can be configured as can the position and number of angle segments per 720 °CA. Up to 12 angle segments per 720 °CA are supported.

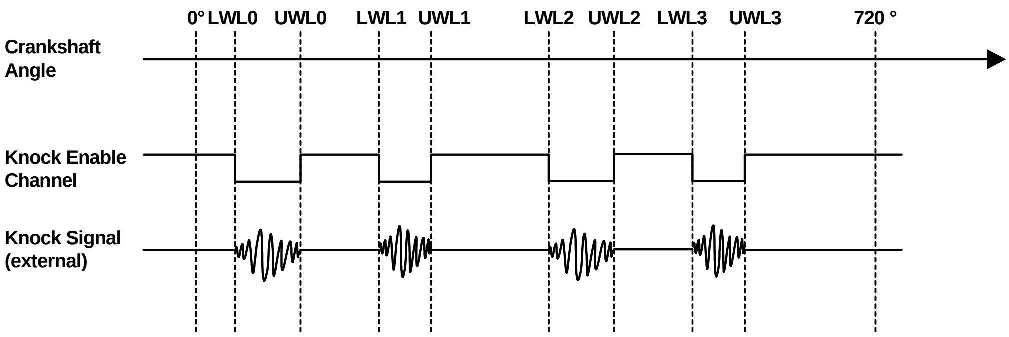


Fig. 9-4 Low-Active Knock Enable Signal for Activating and Deactivating an External Knock Generator

9.1.6 The Structure of the ES1334.2-RTIO Tree

The inclusion of the ES1334.2 in an RTIO Editor starts by selecting the ES1334.2-VMI subsystem.

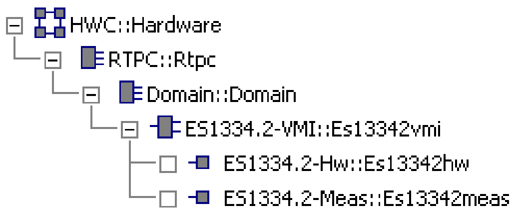


Fig. 9-5 RTIO Hardware Description with ES1334.2

The ES1334.2 has 16 hardware channels for acquiring and conditioning analog and digital signals plus 4 output signals to activate and deactivate external knock generators. The 16 hardware channels and the 4 "knock enable" outputs are configured using the ES1334.2-Hw device which is assigned to the ES1334.2-

VMI subsystem. In addition to this item, up to two ES1334.2-Meas devices can be assigned to the ES1334.2-VMI subsystem. Up to 32 measurements can be specified with each of these devices. This means that the RTIO driver and firmware of the ES1334.2 thus provide 64 measurements or measurement channels.

Note

Before an ES1334.2-Meas device item can be added, an ES1334.2-Hw device must first be added to configure the ES1334.2 hardware.

9.2 ES1334.2-VMI Subsystem

9.2.1 Globals (ES1334.2-VMI Subsystem)

The ES1334.2-VMI Subsystem is used to set RTIO parameters effective globally, i.e. which have an effect on all ES1334.2-Meas devices.

Fig. 9-6 shows the RTIO parameters of the "Globals" tab.

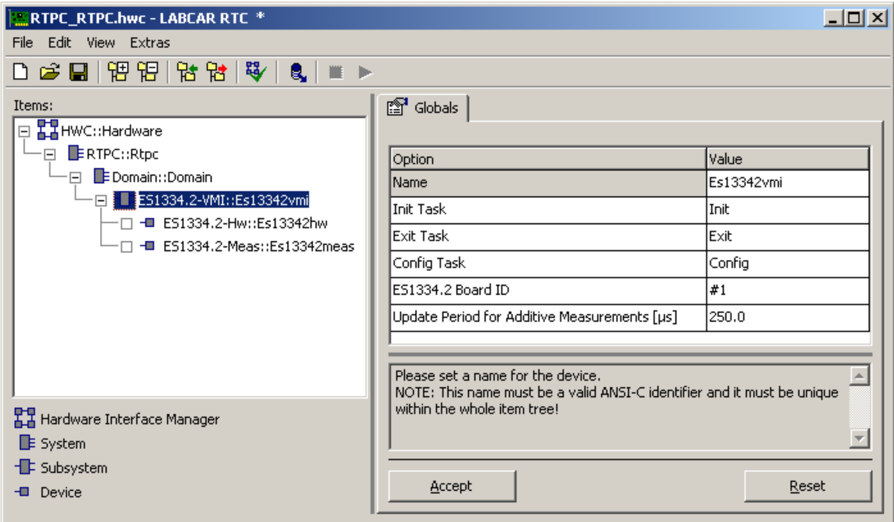


Fig. 9-6 The "Globals" Tab of the ES1334.2-VMI Subsystem

Note

When the ES1334.2 Measurement Board is integrated into a LABCAR-RTC hardware configuration, both address switches (SW1 and SW2) on the board must be set to "0x00"!

ES1334.2 Board ID

This option field is used to identify the ES1334.2. It establishes the assignment between the RTIO hardware description and the ES1334.2 in the VMEbus chassis, for which this description is valid. The ES1334.2 boards are numbered from left to right starting with 1 in the chassis (left = VMEbus slot no. 1, in ascending order). The number of the ES1334.2 board determined must be entered in the "ES1334.2 Board ID" list field. This RTIO parameter cannot be edited can be edited online (i.e. during runtime of the model on the experimental target). Up to 20 ES1334.2 boards can be integrated per chassis in the RTIO Editor.

Update Period for Additive Measurements [µs]

This parameter is only of any importance to software and measurement channels which execute additive time measurements in asynchronous mode. It specifies the maximum intervals at which the additive time has to be updated. For more information, refer to "Functional Description: Asynchronous Additive Pulse-Width Measurements" on page 207. The parameter can be configured during runtime of the model.

Tab. 9-1 lists the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
ES1334.2 Board ID	uint32	No	Identification of the ES1334.2. You can choose from #1 to #20.
Update Period for Additive Measure-ments [µs]	real32	Yes	Measurement update rate for asynchronous additive mea-surements in microseconds. - Minimum: 50 µs - Maximum: 500 ms

* Data type which the RTIO driver uses internally for the parameter

Tab. 9-1 ES1334.2-VMI Subsystem: Configuration Parameters of the "Globals" Tab

9.3 ES1334.2-Hw Device

The ES1334.2-Hw device is used to configure and control the ES1334.2 hardware. Settings which are independent of the individual hardware channels are executed in the "Globals" tab of this item. Channel-specific settings are made in the "Signals" tab.

9.3.1 Globals (ES1334.2-Hw Device)

Fig. 9-7 shows the RTIO parameters of the "Globals" tab. Tab. 9-2 on page 190 lists the properties of the individual parameters.

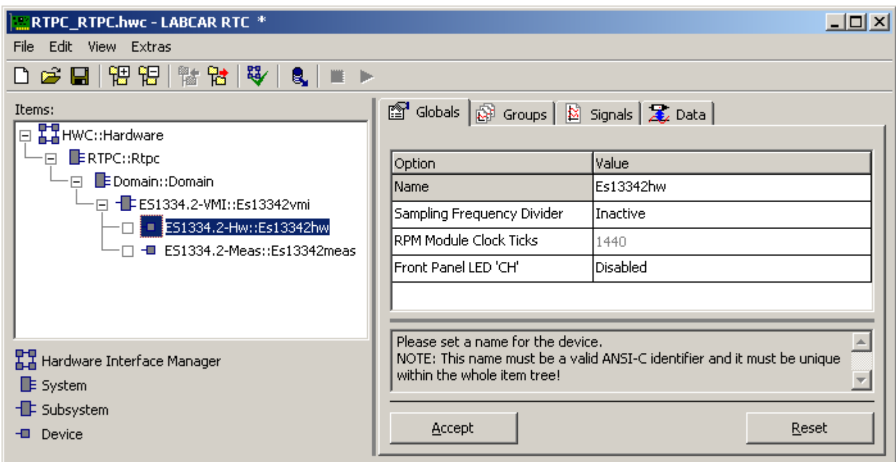


Fig. 9-7 The "Globals" Tab of the ES1334.2-Hw Device

Sampling Frequency Divider

The ES1334.2 samples the input signals at the hardware channels periodically. The maximum sampling rate is determined by the frequency of the internal clock, which is 4 MHz. The sampling frequency can be reduced with a frequency divider. The frequency divider is configured in the "Sampling Frequency Divider" option field of the "Globals" tab. Frequency division can be disabled or configured to dividers 2, 4, 8, 16 or 32. Please note that only input pulses which last longer than the sampling period can be acquired accurately. The parameter can be edited can be edited online (i.e. during runtime of the model on the experimental target).

RPM Module Clock Ticks

To execute speed-synchronous measurements, the ES1334.2 requires information on the angle position of the engine.

This information is determined by the ES1334.2 from the angle clock signal supplied on the front panel via the "SYNC" input. The angle clock signal (see Fig. 9-8) is generated by an ES1332.1 Arbitrary Signal Generator Board and makes it possible to synchronize to 0° camshaft and calculate the angle position of the engine.

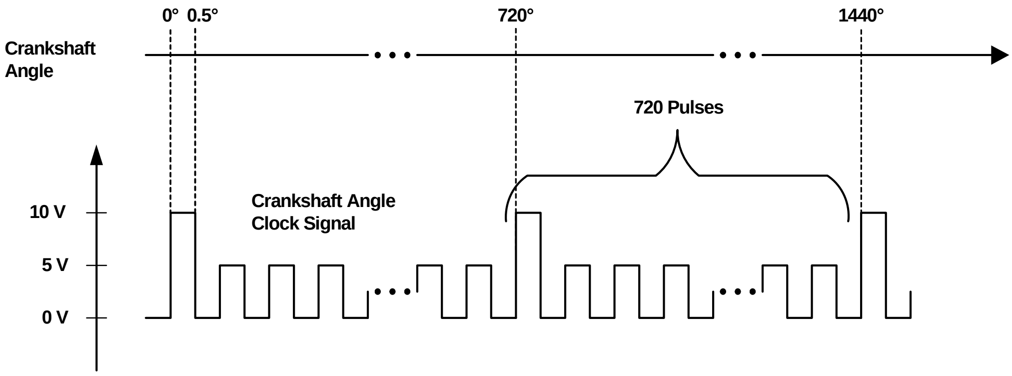


Fig. 9-8 Angle Clock Signal to Synchronize to 0° Camshaft and Calculate the Angle Position of the Engine

The numeric input field "RPM Module Clock Ticks" is used to define the angular resolution, i.e. the number of angle clocks per camshaft rotation (720° crankshaft). This option is fixed at a value of 1440, as the angle clock signal generated by ES1332.1 Arbitrary Signal Generator Boards generates this number of ticks or edges. The value set, i.e. 1440 angle clocks per camshaft rotation, corresponds to an angular resolution of 0.5 °CA (°CA = degrees crankshaft).

Front Panel LED 'CH'

On the front panel of the ES1334.2, there is an LED labeled "CH". The signal level

- on one of the 16 hardware channels or
- on one of the 4 knock enable outputs or
- of the angle clock signal (see Fig. 9-8) or
- of the signal for synchronizing to 0° camshaft (see Fig. 9-9)

can be displayed using this LED. If the signal has a low level, the LED is dark; if it has a high level, the LED is light. The definition of low and high level for the 16 input channels comes from the threshold voltage of the input comparator of the relevant channel. With the knock enable outputs and the clock angle signal, the threshold for the distinction of high and low level is 2.5 V; with the signal for synchronizing to 0° camshaft, the threshold value is 7.5 V.

The "Globals" tab has a "Front Panel LED "CH"" option field for the "CH" LED with which the signal is set which addresses the LED or with which the LED can be deactivated. The option field can be edited can be edited online (i.e. during runtime of the model on the experimental target).

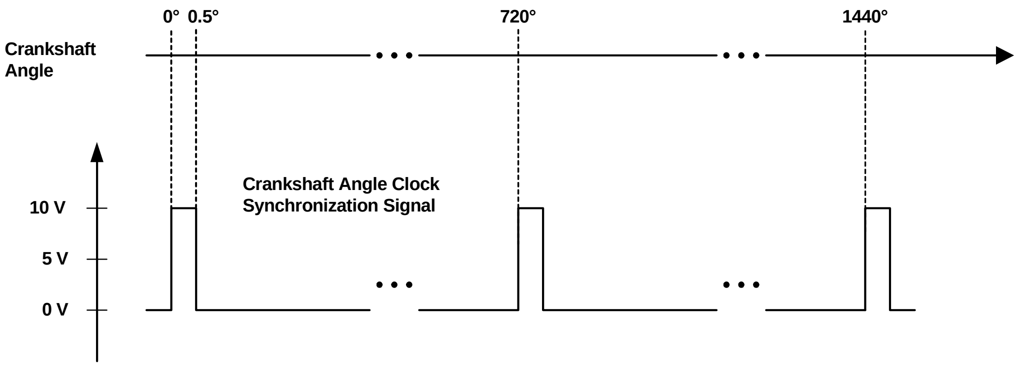


Fig. 9-9 Signal for Synchronizing to 0° Camshaft

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Sampling Frequency Divider	uint8	Yes	Frequency divider for input signal sampling 0: no frequency division (4 MHz sampling) 1: division by 2 (2 MHz) 2: division by 4 (1 MHz) 3: division by 8 (500 kHz) 4: division by 16 (250 kHz) 5: division by 32 (125 kHz)
RPM Module Clock Ticks	uint32	No	Angular resolution (set to 1440)
Front Panel LED "CH"	sint8	Yes	Signal or channel which is connected to the front panel LED "CH" or deactivation of the LED. -1: deactivation 0...15: hardware channel #0... #15 16...19: knock enable output #0...#3 20: angle clock signal (RPM Clock) 21: signal for synchronizing to 0° camshaft (engine speed)
* Data type which the RTIO driver uses internally for the parameter			

Tab. 9-2 ES1334.2-Hw Device: Configuration Parameters of the "Globals" Tab

9.3.2 Groups (ES1334.2-Hw Device)

The ES1334-Hw item has three signal groups (Fig. 9-10).

The "Control" signal group is transferred from the experimental target to the ES1334.2. This signal group activates/deactivates edge recognition on the individual hardware channels of the board.

The "Level" signal group is transferred in the opposite direction from the ES1334.2 to the experimental target. This signal group contains the current level information of all activated hardware channels on which measurements are carried out.

The "Knock" signal group is transferred from the experimental target to the ES1334.2. This signal group is used to activate/deactivate the knock enable outputs.

ERCOS^{EK} operating system tasks must be assigned to the signal groups. Normally a task with the activation type "Alarm" and a relatively large activation period (e.g. 100 ms) is selected for the "Control" signal group as the activation and deactivation of the hardware channels is not usually a highly dynamic procedure. If the activation and deactivation of the hardware channels is only to take place when the model is started or stopped, it is sufficient to assign the "Control" signal group to the "Init" and the "Exit" task of the model.

If the level information is evaluated in the simulation model, a task with the activation type "Alarm" is assigned to the "Level" signal group. The activation period depends on the dynamic behavior or period duration of the signals to be acquired.

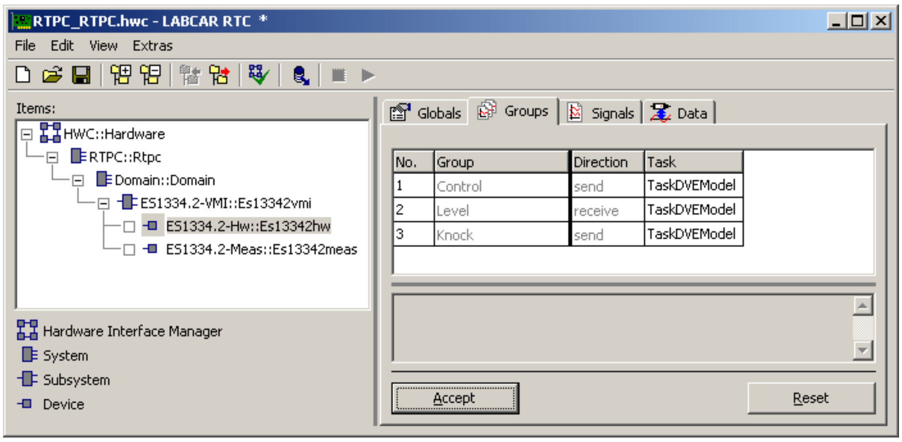


Fig. 9-10 The "Groups" Tab of the ES1334.2-Hw Device

RTIO Signals of the "Control" Signal Group

The "Control" signal group consists of 16 RTIO signals with which edge recognition on the 16 hardware channels is activated or deactivated. Tab. 9-3 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
ChnEnable_0 ... ChnEnable_15	bool	Activation/deactivation of hardware channels 0 to 15 0: deactivation of edge recognition 1: activation of edge recognition
* Data type which the RTIO driver uses internally for the parameter		

Tab. 9-3 ES1334.2-Hw Device: The RTIO Signals of the "Control" Signal Group

Note

The "Control" signal group is only transferred to the DPRAM of the ES1334.2 if one of its RTIO signals changes value.

RTIO Signals of the "Level" Signal Group

The "Level" signal group contains the RTIO signal "LvlBitField_0" which is to be interpreted as a bit field. The bit field has 16 bits which contain coded level information of hardware channels 0 to 15.

Tab. 9-4 lists the properties of the RTIO signal.1

Note

Only level bits of hardware channels for which edge recognition is activated are valid in the "LvlBitField_0" bit field. For details of activating and deactivating edge recognition on a hardware channel, refer to the section "RTIO Signals of the "Control" Signal Group" on page 191.

RTIO Signal	Data Type*	Comment / Value Range
LvlBitField_0	uint16	Bit field with level information of hardware channels 0 to 15. Channel 0: LSB (Least Significant Bit) Channel 15: MSB (Most Significant Bit) Bit value 0: low level Bit value 1: high level
* Data type which the RTIO driver uses internally for the parameter		

Tab. 9-4 ES1334.2-Hw Device: The RTIO Signals of the "Level" Signal Group

RTIO Signals of the "Knock" Signal Group

The ES1334.2 provides 4 knock enable outputs. Up to 12 angle windows can be defined for each of these outputs within which the relevant output is activated. The "Knock" signal group consists of 4 RTIO signals (KnckEnable_0 to KnckEnable_3) with which angle windows of the 4 knock enable outputs of the board can be activated or deactivated. The RTIO signals are to be interpreted as bit fields: the LSB (bit 0) activates or deactivates angle window 0, bit 1 activates or deactivates angle window 1 etc.

Tab. 9-5 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
KnckEnable_0 ... KnckEnable_3	uint16	Activation/deactivation of the knock enable outputs 0 - 3 Bit fields: Angle window 0: Bit 0 (LSB) Angle window 1: Bit 1 ... Angle window 11: Bit 11 Bit value 0: deactivation/power off Bit value 1: activation/power on
* Data type which the RTIO driver uses internally for the parameter		

Tab. 9-5 ES1334.2-Hw Device: The RTIO Signals of the "Knock" Signal Group

Note

The "Knock" signal group is only written to the DPRAM of the ES1334.2 when the value of one of its RTIO signals has changed.

9.3.3 Signals (ES1334.2-Hw Device)

The "Signals" tab is used to configure the 16 hardware channels and the four knock enable outputs of an ES1334.2. Fig. 9-11 shows the RTIO parameters of the "Signals" tab.

Tab. 9-6 lists the properties of the individual parameters. All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

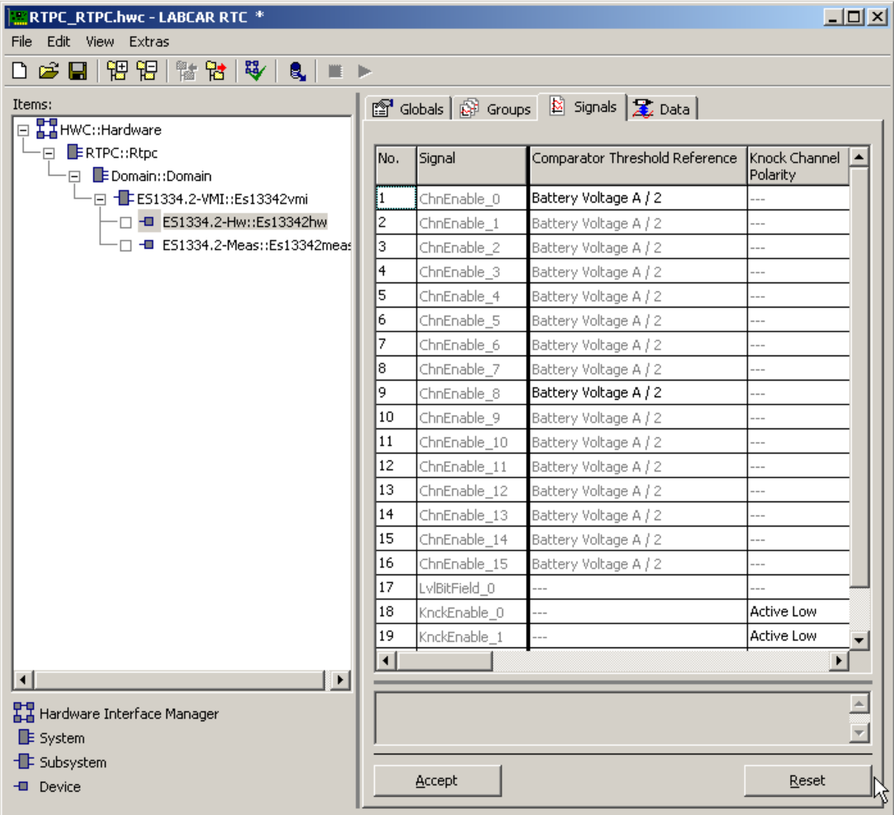


Fig. 9-11 The "Signals" Tab of the ES1334.2-Hw Device

Comparator Threshold Reference

The "Comparator Threshold Reference" option field is used to specify the threshold voltage of an input comparator. The comparator thresholds can only be set to one of the following values for channels 0 to 7 and 8 to 15 in each case:

- 2.5 V
- (battery voltage A)/2
- (battery voltage B)/2
- external reference voltage

Knock Channel Polarity

Polarity of the knock enable output (low-active or high-active).

# of Knock Windows

Number of angle windows of the knock enable output. Up to 12 angle windows can be defined.

LWL#x (x=0, 1, ... 11)

Lower limit of angle window in °CA. LWL stands for Lower Window Limit.

UWL#x (x=0, 1, ... 11)

Upper limit of angle window in °CA. UWL stands for Upper Window Limit.

When defining angle windows, please note that the angle windows of a knock enable output must not overlap. The window limits must be in the range [-720.0 °CA, +720 ° CA] and the size of a window has to be less than 720 °CA.

Parameter or Option Field	**Data Type***	**Can be Edited Online**	**Comment / Value Range**
Comparator Threshold Reference	uint8	Yes	Comparator reference 0: 2.5 V 1: (battery voltage A)/2 2: (battery voltage B)/2 3: External reference voltage
Knock Channel Polarity	uint8	Yes	Polarity of the knock enable signal 0: low-active 1: high-active
# of Knock Windows	uint8	Yes	Number of knock windows of the knock enable output (1, 2, ... 12)
LWL#x (x=0, 1, ... 11)	real32	Yes	Lower limit of knock window in °CA. Value range: -720 °CA to 720 °CA
UWL#x (x=0, 1, ... 11)	real32	Yes	Upper limit of knock window in °CA. Value range: -720 °CA to 720 °CA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 9-6 ES1334.2-Hw Device: Configuration Parameters of the "Signals" Tab

9.4 Measuring Configuration - ES1334.2-Meas Device

The ES1334.2-Meas device is used to specify and configure measurements. Every ES1334.2-Meas device provides 32 software channels (referred to below as measurement channels) which can be freely assigned to the 16 hardware input channels of the ES1334.2. Up to 2 ES1334.2-Meas devices are supported per ES1334.2 making it possible to configure a total of 64 measure channels.

9.4.1 Globals (ES1334.2-Meas Device)

No settings are to be made in the "Globals" tab (Fig. 9-12). In the "Measurement Device ID" option field, a number is assigned to the ES1334.2-Meas device by the system for identification purposes.

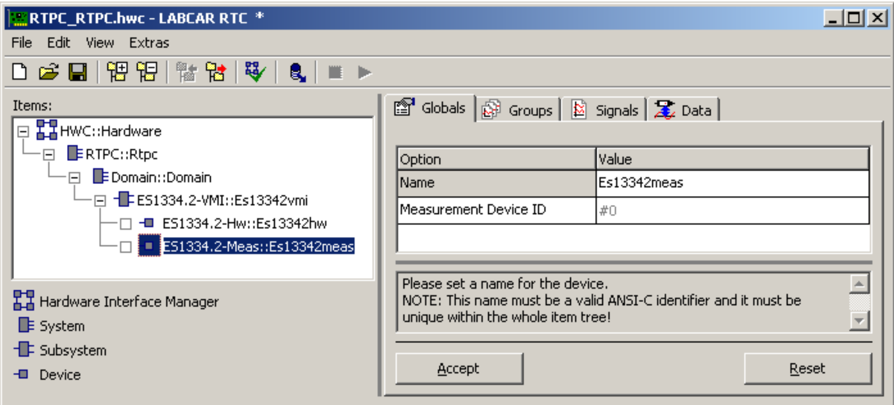


Fig. 9-12 The "Globals" Tab of the ES1334.2-Meas Device

9.4.2 Groups (ES1334.2-Meas Device)

The ES1334.2-Meas device has one signal group (Fig. 9-13) which is transferred to the experimental target by the ES1334.2. It contains all measurement data such as measurement values, trigger and timeout information. An ERCOS^{EK} operating system task with the activation type "Alarm" must be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

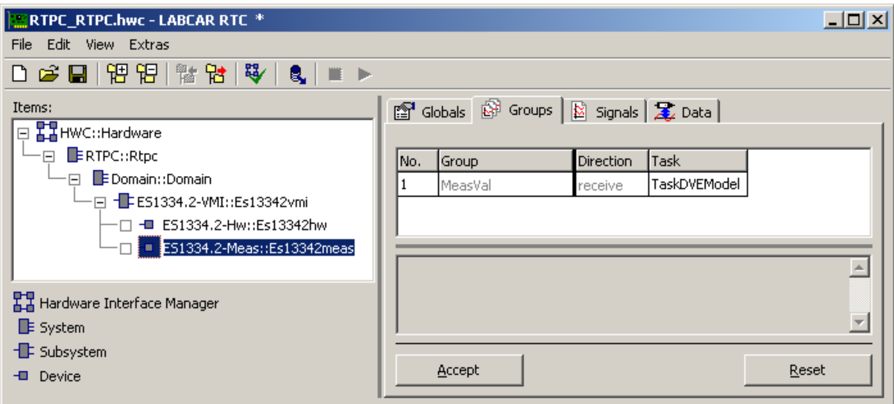


Fig. 9-13 The "Groups" Tab of the ES1334.2-Meas Device

9.4.3 RTIO Signals of the "MeasVal" Signal Group

The "MeasVal" group consists of 65 RTIO signals. The "TriggerBitField_0" signal must be interpreted as a bit field. It has 32 bits. The trigger or update data of the 32 measurement channels is coded in this bit field, i.e. it shows which measurement values have been updated since the last activation of the read process for the "MeasVal" signal group. If a bit is set, the measurement value of the relevant measurement channel has been newly determined. It is of no importance whether the measurement value was updated due to a timeout or a regular measurement value calculation; the update bit of the measurement channel is set in both cases.

The RTIO signals "MeasVal_0" to "MeasVal_31" contain the measurement values of the 32 measurement channels. These are either measurement values determined in the normal way or, in the case of a timeout, the timeout value intended for this case. If a measurement channel is not used, -8888.0 is assigned to the relevant measurement value. The physical unit of the measurement values depends on the measurement procedure:

- time measurements (time stamp, (additive) pulse duration measurements, period duration) are issued in microseconds
- frequency measurements are in hertz
- angle measurements and angle stamp are specified in degrees crankshaft (°CA)
- all other measurements (duty cycles, pulse count, level measurements) are dimensionless

The RTIO signals "Tout_0" to "Tout_31" contain the results of the timeout monitoring for the relevant measurement channel.

Tab. 9-7 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
TriggerBitField_0	uint32	Bit field with update information of the 32 measurement channels. Measurement channel 0: LSB Measurement channel 31: MSB Bit value 0: measurement value is unchanged Bit value 1: measurement value has been updated
MeasVal_0 ... MeasVal_31	real64	Measurement value If the measurement channel is not used, -8888.0 is issued as a value. Physical unit of the measurement value: - time measurements are in microseconds - frequency measurements are in Hertz - angle measurements are in °CA
Tout_0 ... Tout_31	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive
* Data type which the RTIO driver uses internally for the parameter		

Tab. 9-7 ES1334.2-Meas Device: The RTIO Signals of the "MeasVal" Signal Group

9.4.4 Signals (ES1334.2-Meas Device)

The "Signals" tab is used to configure the 32 measurement channels of an ES1334.2-Meas device.

Fig. 9-14 shows the RTIO parameters of the "Signals" tab. Tab. 9-8 on page 203 lists the properties of the individual parameters.

All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

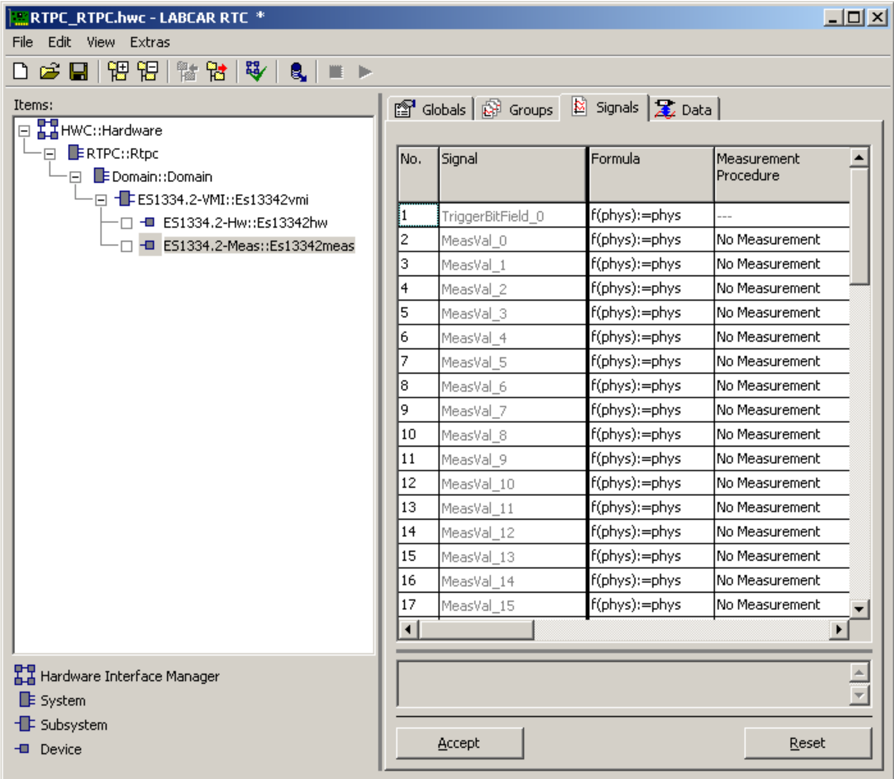


Fig. 9-14 The "Signals" Tab of the ES1334.2-Meas Device

Measurement Procedure

This is where the measurement procedure is selected. From page 204 you can find detailed descriptions and configuration guidelines for the individual procedures. The notation --/-- denotes evaluation at rising edges whereas --\-- denotes evaluation at falling edges.

If "No Measurement" is set in the list box, no measurement is executed.

Note

To avoid unnecessary computing time, measurement channels which are not required should be deactivated.

Hardware Channel

This is where you specify the hardware channel on which measurement is to take place.

Measurement Mode

The "Measurement Mode" option field defines the mode in which measuring takes place. There are two options:

- ESpeedSync

  Execution of speed-synchronous or angle-synchronous measurements. The execution of angle-synchronous measurements makes it necessary for the ES1334.2 to be supplied with an angle clock signal via the "SYNC" input on the front panel.
- Asynchronous

  Execution of asynchronous measurements.

Reference Channel

Some measurement procedures require a second hardware channel or a second measurement channel to be executed. This reference channel is specified in the "Reference Channel" list box.

With the pulse width measurements with Enable or Validate signal described in section 9.5.3 on page 210, the reference channel is a hardware channel, the channel at which the Enable or Validate signal is pending.

With the relative measurements between hardware channels described in section 9.5.6 on page 217 and the measurements for position tracing on two-phase stepper motors described in section 9.5.11 on page 228, the reference channel is a measure channel.

With all other measurement procedures, the value set in the "Reference Channel" list box is of no significance.

(Max) Pulse (Count)

The significance of this option field is not specific but depends on the relevant measurement procedure. With pulse- and edge-selective measurements, the number of the pulse or edge to be measured is specified in this option field. With additive measurements and pulse counts, the maximum number of pulses to be taken into consideration is specified in this option field.

Timeout Monitoring

Definition of timeout monitoring for the relevant measurement channel (see the section "Timeout Monitoring" on page 184). The following settings are possible:

- Inactive

  No timeout monitoring.
- Intvl Predef

  Timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- Intvl InpDep

  Timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout depends on the level of the input signal.
- CS Angle Predef

  Timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.

- CS Angle InpDep

  Timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout depends on the level of the input signal.

- Intvl & CS Angle Predef

  Timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field and timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. A timeout is recognized when one or both timeout criteria are violated. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.

- Intvl & CS Angle InpDep

  Like "Intvl & CS Angle Predef", but the measurement value in a timeout depends on the level of the input signal.

Default Timeout Value

The value set in this numeric input field is issued as the measurement value in a timeout if one of the "Intvl Predef", "CS Angle Predef" or "Intvl & CS Angle Predef" timeout monitoring modes is set.

Timeout Interval [ms]

Time between two timeout checks in milliseconds. Only relevant for time-based timeout monitoring modes ("Intvl Predef", "Intvl InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Timeout Check Point

Crankshaft angle at which timeout checks are executed. Only relevant for angle-based timeout monitoring modes ("CS Angle Predef", "CS Angle InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Lower Limit

Lower limit of the angle window released for measurement in degrees crankshaft. Only relevant in the angle-synchronous measurement mode "ESpeedSync".

CS Angle Upper Limit

Upper limit of the angle window released for measurement in degrees crankshaft. Only relevant in the angle-synchronous measurement mode "ESpeedSync". The upper angle window limit is also used for additive time measurements and pulse counts to transfer the determined measurement value to the DPRAM of the ES1334.2.

CS Angle Reference

Angle reference point in degrees crankshaft to which all angle measurements refer. This parameter is of no importance in all other measurements.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Measure-ment Proce-dure	sint32	Yes	Selection of the measurement procedure or deactivation of the measurement channel. ("measurement procedure" parameter value) Deactivation of measurement channel -1 "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 "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

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
			"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	Yes	Hardware channel on which the measurement is executed. 0 ... 15 channel number
Measurement Mode	uint8	Yes	Measurement mode 0: "Asynchronous": asynchronous measurements 1: "ESpeedSync": speed- or angle-synchronous measurements
Reference Channel	sint8	Yes	Reference channel of the measurement -1: no reference channel 0 ...15: hardware channel 0 to 15 16 ... 47: measure channel 0 to 31

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
(Max) Pulse (Count)	uint32	Yes	With pulse- and edge-selective measurements: the number of the pulse or the edge to be measured. With additive measurements and pulse counts: the maximum number of pulses to be taken into consideration. Value range: ≥ 0
Timeout Monitoring	uint8	Yes	Timeout monitoring: 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	Yes	Measurement value in a timeout. Only relevant in "... Predef" modes for timeout monitoring.
Timeout Interval [ms]	real32	Yes	Time between two timeout checks in milliseconds. Value range: ≥ 1.0 ms Only relevant in "Intvl ..." modes for timeout monitoring.
CS Angle Timeout Check Point	real32	Yes	Crankshaft angle at which timeout checks are executed. Value range: -720 °CA to 720 °CA Only relevant in "CS Angle ..." modes for timeout monitoring.
CS Angle Lower Limit	real32	Yes	Lower limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Upper Limit	real32	Yes	Upper limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Reference	real32	Yes	Angle reference point in degrees crankshaft to which all angle measurements refer. This parameter is of no importance in all other measurements. Value range: -720 °CA to 720 °CA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 9-8 ES1334.2-Meas Device: Configuration Parameters of the "Signals" Tab

9.5 Measurement Procedures

9.5.1 Pulse-Width Measurements

This section describes the properties of pulse-width measurements. A distinction is made between "pulse-qualified" and "edge-qualified" measurements. Each of the following measurements is described:

Pulse-Qualified Pulse-Width Measurements

- "Hightime (Pulse Qual.) [µs]"
- "Lowtime (Pulse Qual.) [µs]"
- "H-Time n-th Pulse (Pu Qual.) [µs]"
- "L-Time n-th Pulse (Pu Qual.) [µs]"

Edge-Qualified Pulse-Width Measurements

- "Hightime (Edge Qual.) [µs]"
- "Lowtime (Edge Qual.) [µs]"
- "H-Time n-th Pulse (Eg Qual.) [µs]"
- "L-Time n-th Pulse (Eg Qual.) [µs]"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- pulse selection (only with "H-Time n-th Pulse ..." and "L-Time n-th Pulse ..."): "(Max) Pulse (Count)" defines the number of the pulse to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Functional Description: Angle-Synchronous Pulse-Width Measurements

In **pulse-qualified measurements**, only those pulses are measured whose opening and closing edges are within one and the same angle window (valid pulse). Fig. 9-15 on page 205 illustrates how pulse-qualified measuring works using a low-active pulse-width measurement (measurement procedure "Lowtime (Pulse Qual.) [µs]").

In **edge-qualified measurements**, those pulses are measured whose opening or closing edge is within any one angle window (valid pulse). Pulses which have both edges outside a angle window are not measured or taken into consideration. Fig. 9-16 illustrates how edge-qualified measuring works using a low-active pulse-width measurement ("Lowtime (Edge Qual.) [µs]").

In **non-pulse-selective measurements** ("Hightime (Pulse Qual.) [µs]", "Lowtime (Pulse Qual.) [µs]", "Hightime (Edge Qual.) [µs]", "Lowtime (Edge Qual.) [µs]"), every valid pulse is measured; the measured pulse width is transferred to the DPRAM of the ES1334.2 on the closing edge of the pulse.

In **pulse-selective measurements** ("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]"), the n-th valid pulse within a angle window is measured.

The pulse number is defined with the "(Max) Pulse (Count)" parameter of the "Signals" tab. The measured pulse width is transferred on the closing edge of the n-th valid pulse.

Note

If 0 is entered as pulse number in pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

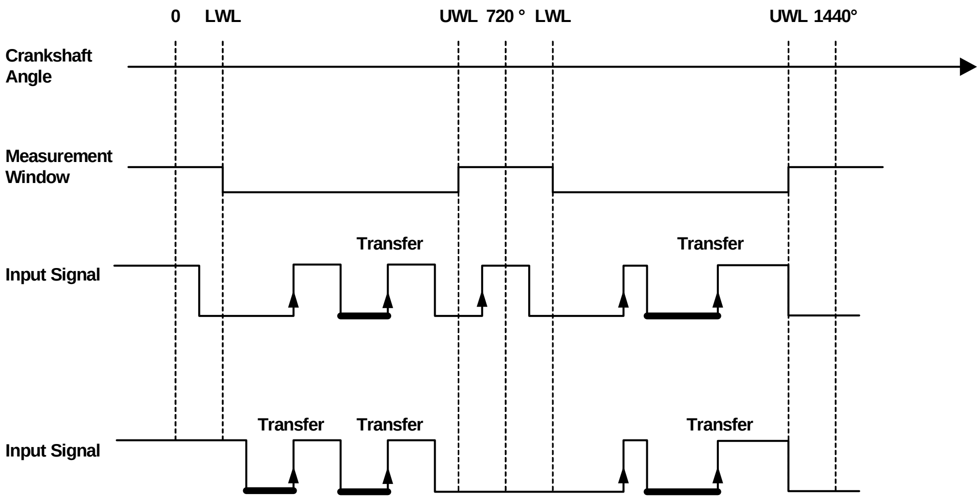


Fig. 9-15 Illustration of how a pulse-qualified, low-active pulse-width measurement ("Lowtime (Pulse Qual.) [µs]" measurement procedure) works. Measured pulses are shown by a thicker line.

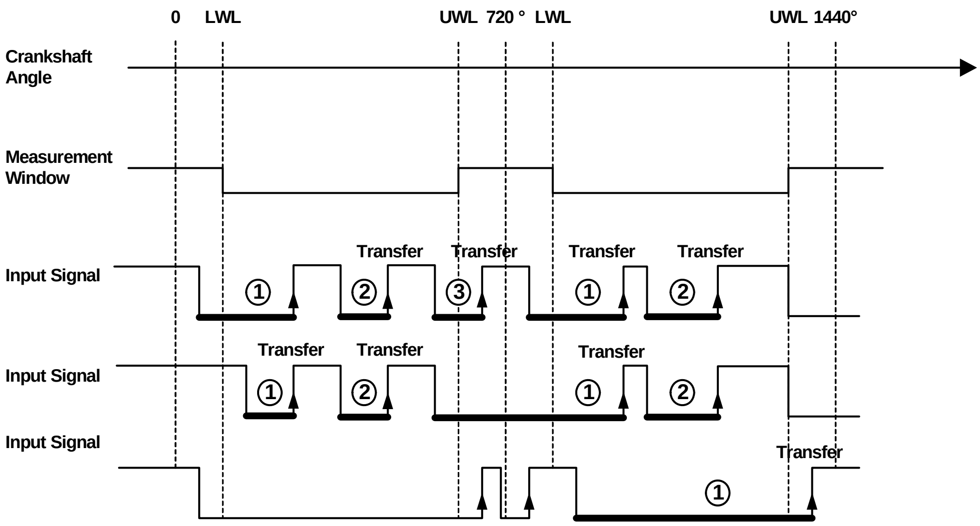


Fig. 9-16 Illustration of how an edge-qualified, low-active pulse-width measurement ("Lowtime (Edge Qual.) [µs]" measurement procedure) works. Measured pulses are shown by a thicker line. The assignment of the pulses to the angle windows and the counting of the pulses within a angle window is also shown.

Functional Description: Asynchronous Pulse-Width Measurements

With asynchronous pulse-width measurements, it is impossible and in fact unnecessary to make a distinction between **pulse-qualified** and **edge-qualified measurements** as there is no window functionality for selecting pulses. The pulse-qualified pulse-width measurements have exactly the same function in asynchronous mode as the edge-qualified measurements.

The **non-pulse-selective measurements** measure the pulse width of every pulse with an active level and transfer the measurement value to the DPRAM on the closing edge of the pulse.

There is virtually no practical use for the **pulse-selective pulse-width measurements** in asynchronous mode. The pulse width of the n-th pulse with an active level after initialization or configuration of the ES1334.2 is measured. The measurement value is transferred to the DPRAM on the closing edge of the pulse.

Note

If 0 is entered as pulse number in pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-9.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	- If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level is active when the timeout check takes place, the length of time which has elapsed since the last opening edge of the signal is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-9 Pulse-Width Measurements: Measurement Value in a Timeout with Different Timeout Monitoring Procedures

9.5.2 Additive Pulse-Width Measurements

This section describes the properties of additive pulse-width measurements. The functioning of the following measurements is described:

- "Additive Hightime [µs]"
- "Additive Lowtime [µs]"

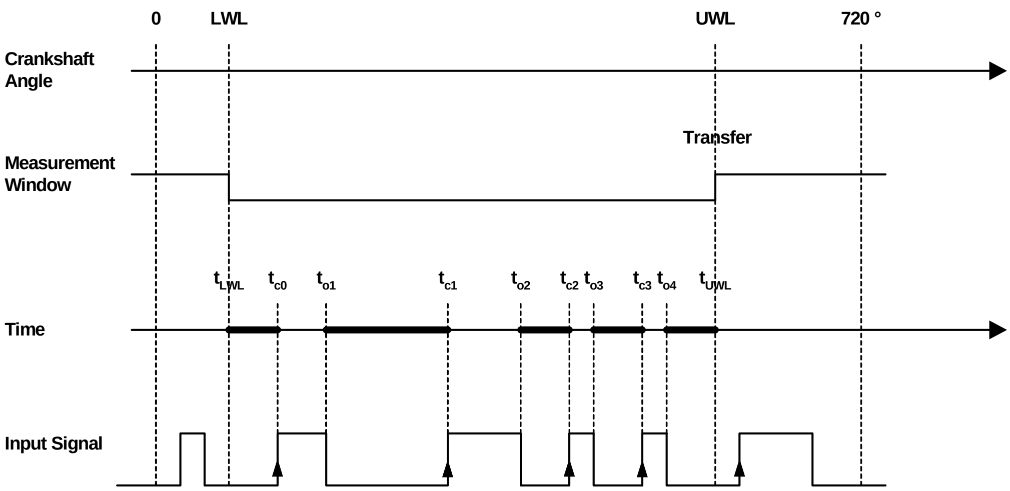
Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- angle windows ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode.
- timeout monitoring

Functional Description: Angle-Synchronous Additive Pulse-Width Measurements

The additive time results from the total of all time segments within a angle window in which the signal is active, regardless of whether the opening or closing edges of the pulses are inside or outside the angle window. Fig. 9-17 shows an example of measurement value calculation with an additive lowtime measurement.

The additive time is transferred to the DPRAM of the ES1334.2 on the upper angle window limit.



$$t_{Additive} := (t_{c0}-t_{LWL}) + (t_{UWL}-t_{o4}) + \quad (t_{ci} - t_{oi}), \quad i = 1 \ldots 3$$

Fig. 9-17 Illustration of how an angle-synchronous additive lowtime measurement ("Additive Lowtime [µs]" measurement procedure) works. The additive time is the total of the line segments shown in bold face.

Functional Description: Asynchronous Additive Pulse-Width Measurements

With asynchronous additive pulse-width measurements, the time in which the signal to be measured is active is totalled. The total in the DPRAM is updated at every closing pulse edge and at the latest after an updating period defined by the user. Fig. 9-18 shows when the additive time is updated using an additive lowtime measurement as an example. The update period is set by the user in the

"Update Period for Additive Measurements [µs]" parameter in the "Globals" tab of the ES1334.2-VMI subsystem. The totalled time is reset to 0 if the ES1334.2 is initialized or configured.

Note

The additive time is acquired modulo 2^{52} µs, i.e. after (2^{52} - 1) µs (corresponds to approx. 143 years), the additive time returns to 0. (2^{52} - 1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

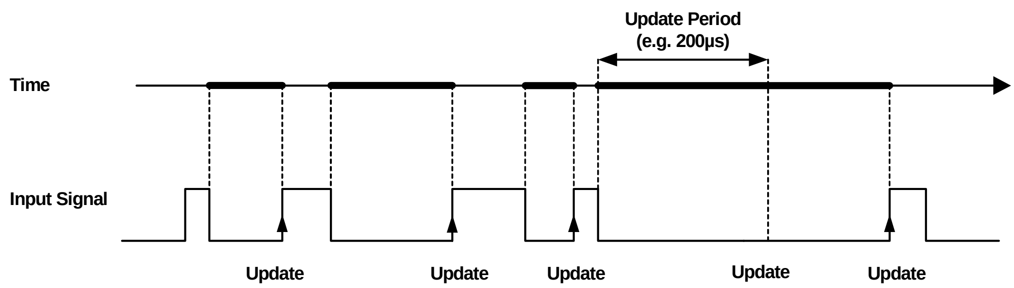


Fig. 9-18 Illustration of how an asynchronous additive lowtime measurement ("Additive Lowtime [µs]" measurement procedure) works. The additive time is the total of the time segments shown in bold face. The times at which the total is updated in the DPRAM are also shown.

Timeout Monitoring

Tab. 9-10 lists the condition of when a timeout is initiated for different timeout monitoring procedures. The measurement value issued in a timeout is specified in Tab. 9-11.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "Intvl Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel *and* the signal has been inactive since the last timeout check. In angle-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"CS Angle InpDep" "CS Angle Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel since the last timeout check. In angle-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"Intvl & CS Angle Inp-Dep" "Intvl & CS Angle Predef"	The timeout condition for this monitoring procedure results from an OR operation of the timeout condition of the "Intvl InpDep" and "Intvl Predef" procedures with the timeout condition of the "CS Angle InpDep" and "CS Angle Predef" procedures.

Tab. 9-10 Additive Pulse-Width Measurements: Timeout Condition for Different Timeout Monitoring Procedures

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle Inp-Dep"	*Asynchronous measurement mode:* The additive time determined up to the point where a check is made for a timeout is issued as measurement value in a timeout. *Angle-synchronous measurement mode:* 0 is issued as measurement value in a timeout.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-11 Additive Pulse-Width Measurements: Measurement Value in a Timeout with Different Timeout Monitoring Procedures

9.5.3 Pulse-Width Measurements with Enable or Validate Signal

In addition to the direct measurement of the pulse widths of a signal independent of all other inputs, the ES1334.2 also makes it possible to measure pulse widths of a signal with reference to a second input. This second input signal is interpreted as an Enable or Validate signal, as shown in Fig. 9-19 using a validated lowtime measurement as an example:

If, for example, the ES1334.2 is configured so that low-imtes of an input signal A are to be measured, with a second input signal B to be used as a high-active Enable signal, only the pulse marked "1" would be measured from the input signal in Fig. 9-19. Or to put it more generally: when using a second input signal as "Enable", only those pulses are measured which **are completely surrounded** by an active Enable pulse.

If, however, an additional input signal C is used as a high-active Validate signal, only the pulse marked "2" would be measured in the above example. Or to put it more generally: when using a second input signal as "Validate", only those pulses are measured which **completely surround** an active Validate pulse.

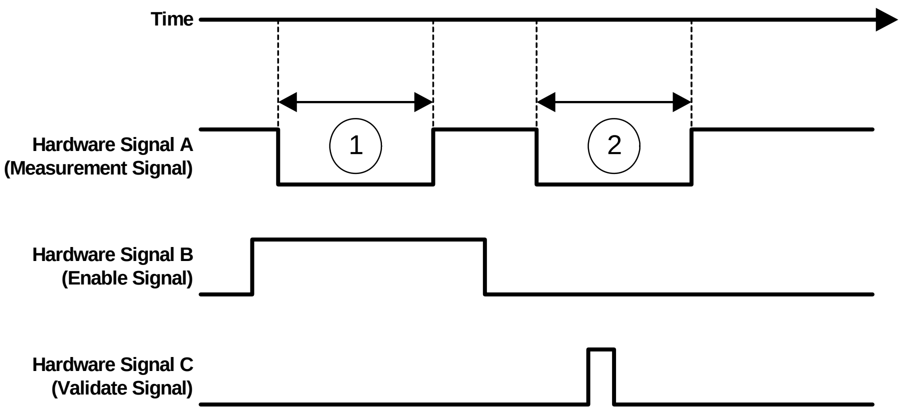


Fig. 9-19 Validated Time Measurements with the ES1334.2

The ES1334.2-RTIO driver offers pulse-width measurements with an Enable or Validate option. The Enable and Validate pulses can either be defined as high-active or as low-active. This section describes how the following measurements work:

- "Hightime using H-Enable [µs]"
- "Hightime using L-Enable [µs]"
- "Hightime using H-Validate [µs]"
- "Hightime using L-Validate [µs]"
- "Lowtime using H-Enable [µs]"
- "Lowtime using L-Enable [µs]"
- "Lowtime using H-Validate [µs]"
- "Lowtime using L-Validate [µs]"
- "H-Time n-th Pulse (H-Ena) [µs]"
- "H-Time n-th Pulse (L-Ena) [µs]"
- "H-Time n-th Pulse (H-Val) [µs]"
- "H-Time n-th Pulse (L-Val) [µs]"

- "L-Time n-th Pulse (H-Ena) [µs]"
- "L-Time n-th Pulse (L-Ena) [µs]"
- "L-Time n-th Pulse (H-Val) [µs]"
- "L-Time n-th Pulse (L-Val) [µs]"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- "Reference Channel": hardware channel at which the Enable or Validate signal is pending
- pulse selection (only with "H-Time n-th Pulse ..." and "L-Time n-th Pulse ..."): "(Max) Pulse (Count)" defines the number of the pulse to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Functional Description - Angle-Synchronous Pulse-Width Measurements with Enable or Validate Signal

Fig. 9-20 shows how angle-synchronous pulse-width measurements with Enable option work using a lowtime measurement with a high-active Enable signal. Only pulses whose opening and closing edge are within one and the same angle window and which are completely surrounded by an active Enable pulse are measured (valid pulse).

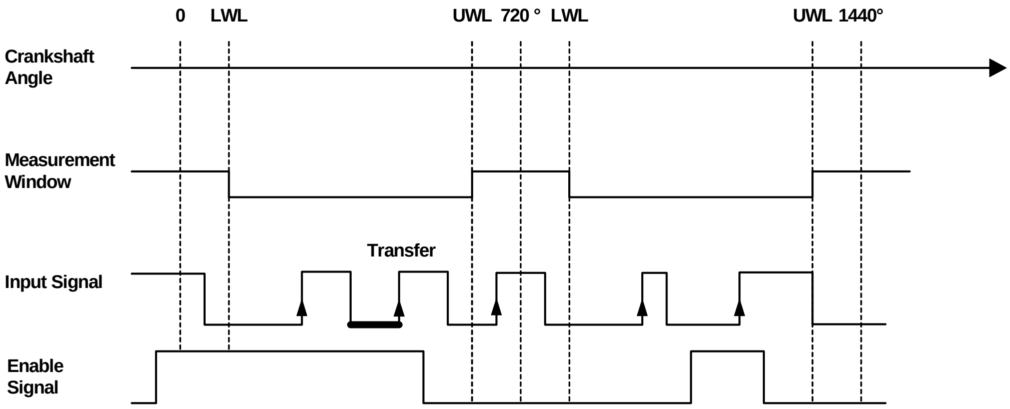


Fig. 9-20 How angle-synchronous pulse-width measurement with Enable signal ("Lowtime using H-Enable [µs]" measurement procedure) works. Thicker lines indicate the pulses measured.

Fig. 9-21 shows how angle-synchronous pulse-width measurements with a Validate option work using a lowtime measurement with a high-active Validate signal. Only pulses whose opening and closing edge are within one and the same angle window and which completely surround an active Validate pulse are measured (valid pulse).

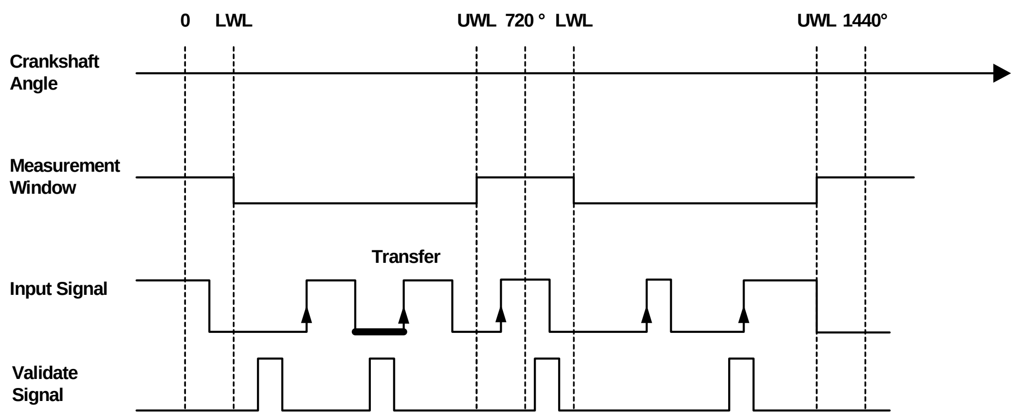


Fig. 9-21 How angle-synchronous pulse-width measurement with Validate signal ("Lowtime using H-Validate [µs]" measurement procedure) works. Thicker lines indicate the pulses measured.

In **non-pulse-selective measurements**, every valid pulse is measured; the measured pulse width is transferred to the DPRAM of the ES1334.2 on the closing edge of the pulse.

In **pulse-selective measurements**, the n-th valid pulse within a angle window is measured. The pulse number is defined with the "(Max) Pulse (Count)" parameter of the "Signals" tab. The measured pulse width is transferred on the closing edge of the n-th valid pulse.

Note

If 0 is entered as pulse number in pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Functional Description - Asynchronous Pulse-Width Measurements with Enable or Validate Signal

The **non-pulse-selective measurements** measure the pulse width of every pulse with an active level validated by an Enable or Validate signal and transfer the measurement value to the DPRAM on the closing edge of the pulse.

The **pulse-selective pulse-width measurements** do not really have a practical use in asynchronous mode. The pulse width of the n-th pulse with an active level validated by an Enable or Validate signal is measured after the ES1334.2 has been initialized or configured. The measurement value is transferred to the DPRAM on the closing edge of the pulse measured.

Note

If 0 is entered as pulse number in pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-12.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle Inp-Dep"	- If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level is active when the timeout check takes place, the length of time which has elapsed since the last opening edge of the signal is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-12 Pulse-width measurements with Enable or Validate signal: measurement value in a timeout with different timeout monitoring procedures

9.5.4 Frequency and Cycle Time Measurements

The ES1334.2 makes it possible to measure frequencies and cycle times at rising and falling signal edges. The user specifies the edge at which the measurement value is calculated by selecting a measurement procedure. The following measurement procedures are available:

- "Cycle Time --/-- [µs]"
- "Cycle Time --\-- [µs]"
- "Frequency --/-- [Hz]"
- "Frequency --\-- [Hz]"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Functional Description

In asynchronous measurement mode, measurement values are calculated at every active signal edge. In angle-synchronous (angle-segmented) measurement mode, measurement values are calculated at every active signal edge within a angle window. Fig. 9-22 shows the calculation of measurement values with angle-segmented frequency or cycle time measurement at rising signal edges.

Note

If the entire angle range of 720° CA is released in an angle-segmented cycle time or frequency measurement, this measurement behaves like the relevant measurement in asynchronous mode.

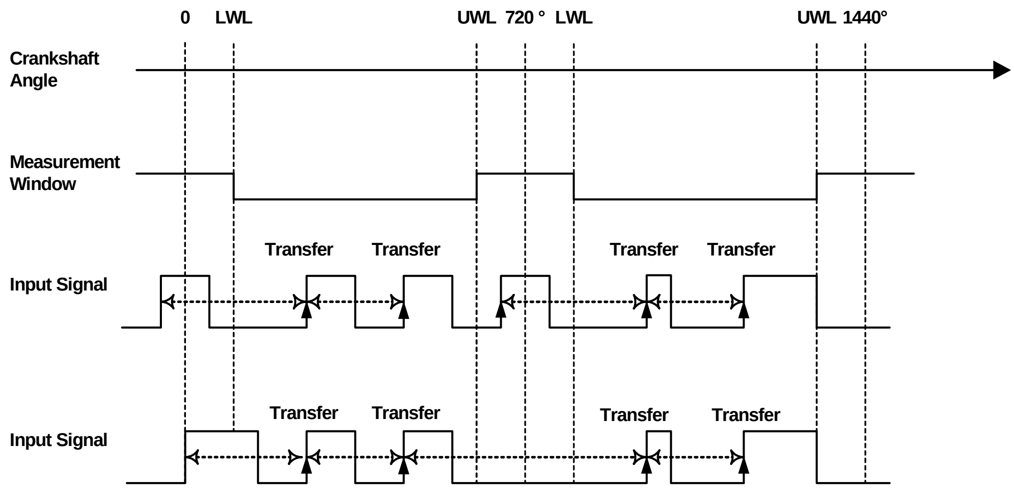


Fig. 9-22 Illustration of measurement value calculation in angle-segmented frequency or cycle time measurement at rising signal edges

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-13.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	*Cycle time measurements:* The length of time between the time the last active edge of the input signal occurred and the time at which a timeout check was executed is issued as measurement value (see Fig. 9-23). *Frequency measurements:* Measurement value calculation in a timeout takes place as for cycle time measurements but a resulting frequency is determined by inversion from the duration determined.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-13 Frequency and cycle time measurements: measurement value in a timeout with different timeout monitoring procedures

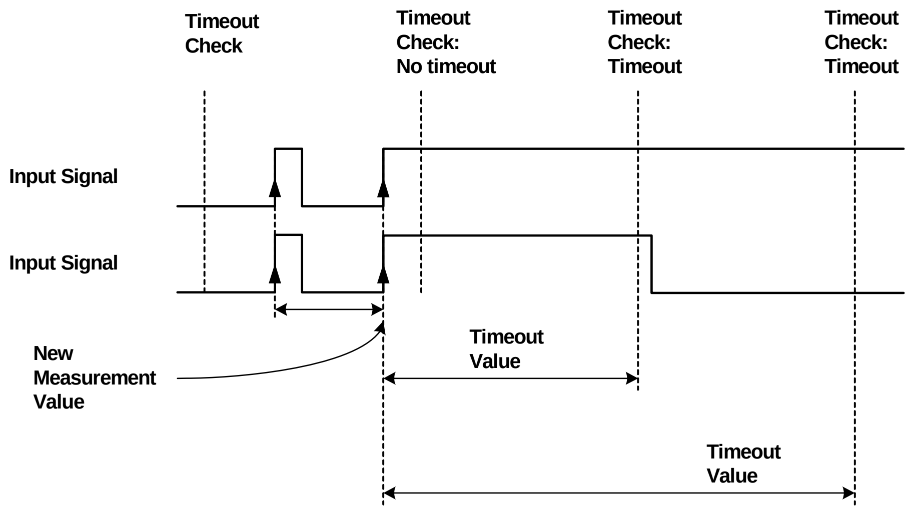


Fig. 9-23 Frequency and cycle time measurements at rising edges: illustration of measurement value calculation in a timeout (the measurement value is referred to as "Timeout Value") in the "... InpDep" timeout monitoring procedure

9.5.5 Duty Cycle Measurements

Duty cycles (e.g. of PWM signals) can be determined in various ways with the ES1334.2. With duty cycle measurements, a distinction is made between measurements which set the high-phase of a signal in relation to the low-phase (or vice versa) and measurements which set the high-phase (or low-phase) of a signal in relation to the cycle duration. The first type of duty cycle measurement is referred to below as duty cycle $^{P/P}$; "P/P" shows that two pulse durations are set in relation to one another. The second type of duty cycle measurement is referred to below as duty cycle $^{P/C}$; "P/C" shows that a pulse duration is set in relation to the cycle time of the signal.

$$\text{Duty Cycle}^{P/P}: \quad \frac{L}{H}, \frac{H}{L} \qquad 0 \leq \text{Duty Cycle}^{P/P} \leq \infty$$

$$\text{Duty Cycle}^{P/C}: \quad \frac{L}{L+H}, \frac{H}{L+H} \qquad 0 \leq \text{Duty Cycle}^{P/C} \leq 1$$

The user specifies the edge at which the measurement value is calculated by selecting the measurement procedure. The following "P/P" duty cycle measurements are available:

- "Duty Factor L/H --/--"
- "Duty Factor L/H --\--"
- "Duty Factor H/L --/--"
- "Duty Factor H/L --\--"

The following "P/C" duty cycle measurements are available:

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

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Functional Description

As far as measurement value calculation in asynchronous and angle-synchronous (angle-segmented) measurement mode is concerned, exactly the same is true of duty cycle measurements as was described for frequency and cycle time measurements in section 9.5.4 on page 213.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-14.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	Duty Cycle $^{P/P}$: $\frac{L}{H}$, $\frac{H}{L}$ - The measurement value is 0.0 if the signal level is "high" for L/H measurements or "low" for H/L measurements when the timeout check is carried out. - The measurement value is MAXREAL32 ($3.40282347 \times 10^{38}$) if the signal level is "low" with L/H measurements or "high" with H/L measurements when the timeout check is carried out. Duty Cycle $^{P/C}$: $\frac{L}{L+H}$, $\frac{H}{L+H}$ The measurement value is 0.0 if the signal level is "high" with L/(L+H) measurements or "low" with H/(L+H) measurements when the timeout check is carried out. The measurement value is 1.0 if the signal level is "low" with L/(L+H) measurements or "high" with H/(L+H) measurements when the timeout check is carried out.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-14 Duty cycle measurements: measurement value in a timeout with different timeout monitoring procedures

9.5.6 Relative Measurements between Hardware Channels

Functional Description

The measurement procedures described so far have all had one thing in common: the fact that the times, frequencies, angles and duty cycles measured were defined by the signal edges of a hardware channel. This section describes measurement procedures in which events (edges) are measured on two hardware channels in relation to one another.

The measurement functionality is explained using Fig. 9-24 as an example. The time from the opening edge of an Enable pulse to the n-th active edge on a reference signal is to be measured. The prerequisite for determining a measurement value is, however, that the n-th active edge is validated by the Enable pulse, i.e. the edge has to occur during the active phase of the Enable signal. The opening edge of the Enable pulse resets counting the active edges of the reference signal to 0. The parameter n for selecting the edge is set in the option field "(Max) Pulse (Count)" in the "Signals" tab. Fig. 9-25 shows a signal trace in which there is no validation of the reference signal.

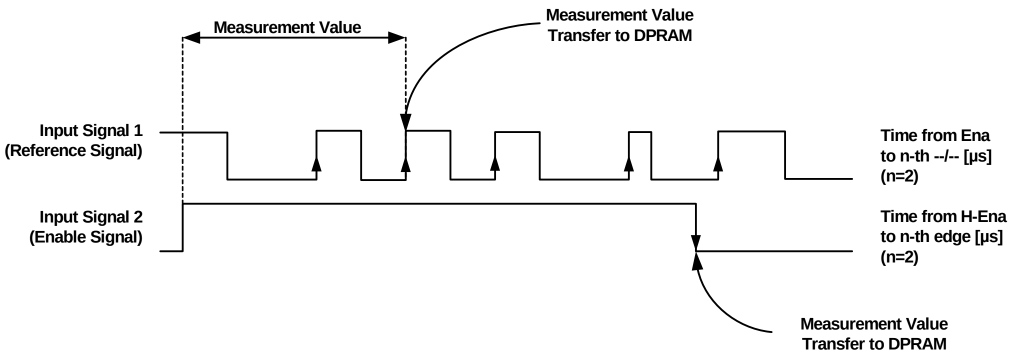


Fig. 9-24 How the measurement procedure for measuring events (edges) on different signals or hardware channels in relation to one another works. The figure shows a measurement with a high-active Enable signal. The RTIO parameter for edge selection ("(Max) Pulse (Count)" in the "Signals" tab) is greater than 0.

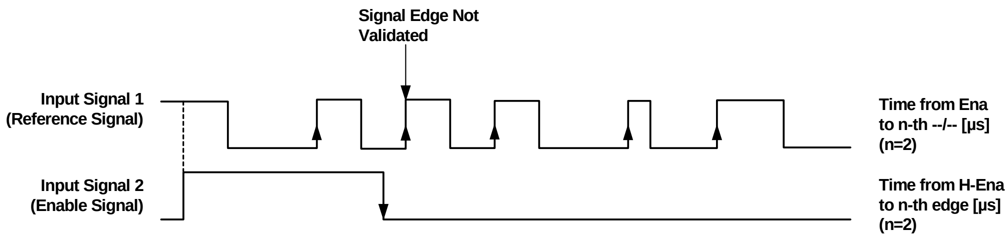


Fig. 9-25 Measuring edges on different hardware channels: signal trace with high-active Enable signal and non-validated n-th active signal edge of the reference signal

For the measurements described to be executed, two measurement or software channels have to be configured. The Enable signal must be assigned to one measurement channel. Depending on whether the Enable signal is low-active or high-active, the measurement channel must be assigned one of the following two measurement procedures:

- "Time from L-Ena to n-th edge [µs]"
- "Time from H-Ena to n-th edge [µs]"

Admissible measurement options are:

- "Reference Channel": refers to the measurement channel to which the reference signal is assigned
- timeout monitoring

The settings of all other measurement options are ignored.

The reference signal must be assigned to the other measurement channel. Depending on whether the active edge of the reference signal is rising or falling, the measurement channel must be assigned one of the following two measurement procedures:

- "Time from Ena to n-th --/-- [µs]"

- "Time from Ena to n-th --\-- [µs]"

Admissible measurement options are:

- "Reference Channel": refers to the measurement channel to which the Enable signal is assigned
- edge selection: "(Max) Pulse (Count)" defines the number of the active edge of the reference signal to be measured

The settings of all other measurement options are ignored.

Note

Segmenting of angle windows is not supported. This is why there is no difference between the functionality of measurements in asynchronous and angle-synchronous measurement mode.

Basically both measurement channels of a measurement return the measurement value determined. The only thing that is different between the two measurement channels is the time at which the measurement value is transferred to the DPRAM of the ES1334.2.

Measurement channels which are configured to the measurement procedure "Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]", transfer the measurement value when the n-th validated active edge occurs (see Fig. 9-24 on page 218).

Measurement channels which are configured to the measurement procedure "Time from L-Ena to n-th edge [µs]" or "Time from H-Ena to n-th edge [µs]", transfer the measurement value on the closing edge of the Enable pulse (see Fig. 9-24 on page 218).

There is a special measurement response when the parameter for edge selection "(Max) Pulse (Count)" is set to 0 in the measurement procedure "Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]". In this case, the measurement procedures "Time from L-Ena to n-th edge [µs]" and "Time from H-Ena to n-th edge [µs]" return the time from the opening edge of the Enable pulse to the **last** validated active edge of the reference signal. The measurement value is transferred to the DPRAM **on the closing edge of the Enable pulse** (Fig. 9-26 on page 220). In this case, the measurement procedures "Time from Ena to n-th --/-- [µs]" and "Time from Ena to n-th --\-- [µs]" return the time from the opening edge of the Enable pulse to **each** validated active edge of the reference signal. The measurement values are transferred to the DPRAM **when the relevant validated active edge occurs**.

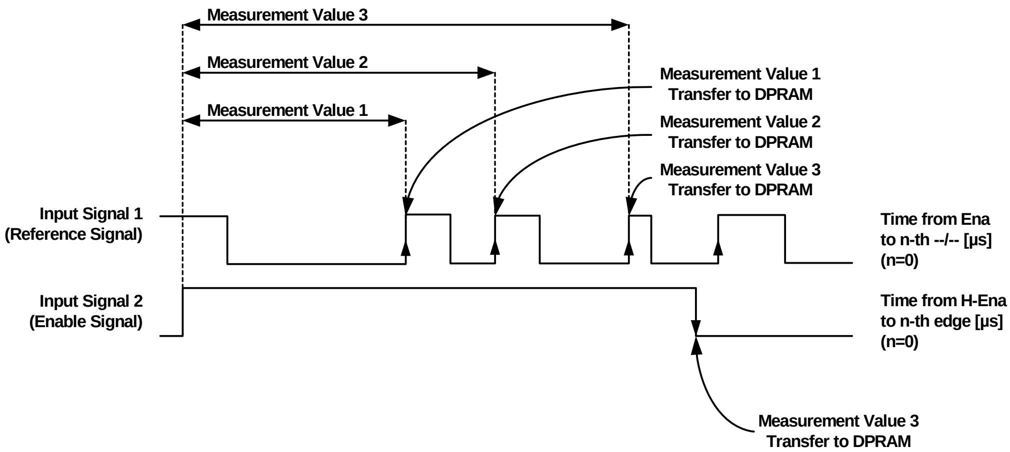


Fig. 9-26 How the measurement procedure for measuring events (edges) on different signals or hardware channels in relation to one another works. The figure shows a measurement with a high-active Enable signal. The RTIO parameter for edge selection ("(Max) Pulse (Count)" in the "Signals" tab) is 0.

Tab. 9-15 lists the configuration guidelines and measurement values of the measurement procedures for measuring edges on different hardware channels.

RTIO Parameter of the "Signals" Tab	Enable Signal	Reference Signal
Hardware Channel	Number of the hardware channel which carries the Enable signal	Number of the hardware channel which carries the reference signal
Measurement Mode	"Asynchronous" or "ESpeedSync". There is no difference between the modes.	"Asynchronous" or "ESpeedSync". There is no difference between the modes.
Measurement Procedure	"Time from L-Ena to n-th edge [µs]" or "Time from H-Ena to n-th edge [µs]" depending on whether the Enable signal is low- or high-active.	"Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]" depending on whether the active edge of the reference signal is rising or falling.
Reference Channel	Number of the measurement channel to which the reference signal was assigned.	Number of the measurement channel to which the Enable signal was assigned.
Measurement Value with the Setting (Max) Pulse (Count) = 0	The time from the opening edge of the Enable pulse to the **last** validated active edge of the reference signal is measured and transferred to the DPRAM **on the closing edge of the Enable pulse**.	The time from the opening edge of the Enable pulse to **every** validated active edge of the reference signal is measured and transferred to the DPRAM **when the relevant active edge occurs**.
Measurement Value with the Setting (Max) Pulse (Count) > 0	The time from the opening edge of the Enable pulse to the n-th validated active edge of the reference signal is measured and transferred to the DPRAM **on the closing edge of the Enable pulse**.	The time from the opening edge of the Enable pulse to the n-th validated active edge of the reference signal is measured and transferred to the DPRAM **when the n-th validated active edge occurs**.

Tab. 9-15 Configuration Guidelines and Measurement Values of the Measurement Procedures for Measuring Edges on Different Hardware Channels

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-16.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	- If the signal level of the Enable signal is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level of the Enable signal is active when the timeout check takes place, 1 is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-16 Measurement procedure for measuring edges on different hardware channels: measurement value in a timeout with different timeout monitoring procedures

9.5.7 Measuring Edges: Angle Stamp

Functional Description

The following measurement procedures measure the crankshaft angle of active signal edges on the assigned hardware channel in relation to the reference angle set.

- "Rising Edge --/-- [deg]"
- "Falling Edge --\-- [deg]"
- "Rising Edge of n-th Pulse [deg]"
- "Falling Edge of n-th Pulse [deg]"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- edge selection (only with "Rising Edge of n-th Pulse [deg]" and "Falling Edge of n-th Pulse [deg]"): "(Max) Pulse (Count)" defines the number of the edge to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- "CS Angle Reference": defines the reference angle of the measurement
- timeout monitoring

The crankshaft angle of an edge provided by the hardware is always in the range [0° CA, 720° CA[. In **angle-synchronous measurement mode**, the measured angle is mapped into the range [LWL, LWL + 720°CA[by the ES1334.2 firmware and then the difference is calculated with the reference angle. This difference is the measurement value provided by the measurement; it is positive when the mapped angle is smaller than the reference angle (cf. Fig. 9-27).

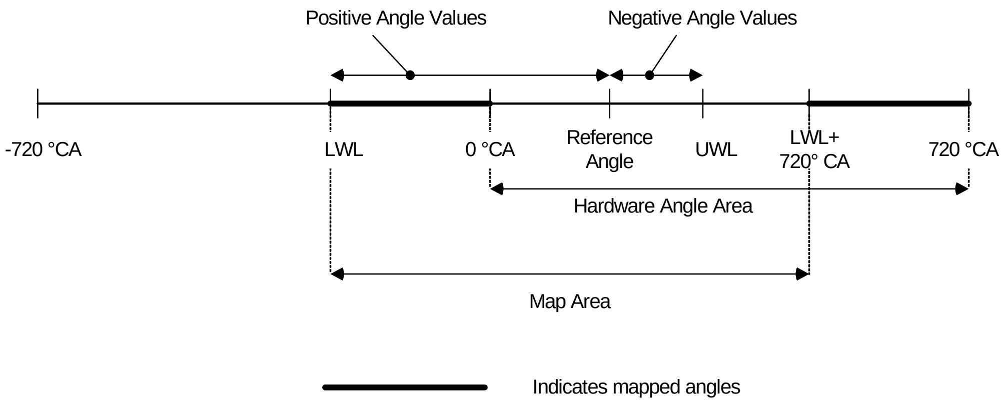


Fig. 9-27 How angle measurement works in angle-synchronous measurement mode

As the reference angle can be in the range [-720° CA, 720° CA], the measurement functions return angle values from the range]-1440 °CA, 1440 °CA] in angle-synchronous measurement mode. In **asynchronous measurement mode**, there is no mapping of the crankshaft angle provided by the hardware which means that the angle values are in the range]-1440 °CA, 720° CA] in this measurement mode.

In **angle-synchronous measurement mode** the **non-edge-selective measurements** ("Rising Edge --/-- [deg]", "Falling Edge --\-- [deg]") measure every active edge within one angle window. The **edge-selective** measurements ("Rising Edge of n-th Pulse [deg]", "Falling Edge of n-th Pulse [deg]") measure the n-th active edge within a angle window. If the lower angle window limit is exceeded, the edge counter is reset to 0.

In **asynchronous measurement mode** the **non-edge-selective measurements** measure the crankshaft angle of every active edge. The **edge-selective** measurements are not much use in this measurement mode as the edge counter is not reset.

Note

If 0 is entered as edge number for the edge-selective measurements in the option field "(Max) Pulse (Count)", these measurements behave like the relevant non-edge-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-17.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the low level for a measurement at rising edges and the high level for a measurement at falling edges. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the high level for a measurement at rising edges and the low level for a measurement at falling edges.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-17 Angle measurements: measurement value in a timeout with different timeout monitoring procedures

9.5.8 Measuring Edges: Time Stamp

Functional Description

The following measurement procedures measure the time of active signal edges on the assigned hardware channel.

- "Time Stamp --/-- [µs]"
- "Time Stamp --\-- [µs]"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- "(Max) Pulse (Count)": defines the number of the edge to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

In **angle-synchronous measurement mode** the time when the n-th active signal edge occurs within a angle window is measured. When the lower angle window limit is exceeded, the edge counter is reset to 0. If 0 is set as edge number in the option field "(Max) Pulse (Count)", every active edge within a angle window is measured.

If 0 is entered as edge number in the "(Max) Pulse (Count)" option field in **asynchronous measurement mode**, the time each active signal edge occurs is measured. If the edge number is not equal to 0, the time of the n-th active edge after an initialization or configuration is measured once.

The times are specified relative to the last time the ES1334.2 was initialized or configured. In other words, the clock is reset to 0 when the ES1334.2 is initialized or configured.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-18.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The last measurement value calculated (before the timeout) is retained as measurement value in a timeout.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-18 Time stamp measurements: measurement value in a timeout with different timeout monitoring procedures

9.5.9 Pulse Count

This section describes the properties of the following pulse-count measurement procedures:

- "Number of Low-Pulses"
- "Number of High-Pulses"
- "Total Number of L-Pulses"
- "Total Number of H-Pulses"

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Functional Description

The "Number of Low-Pulses" and "Number of High-Pulses" measurement procedures acquire the number of valid active pulses within a angle window in **angle-synchronous measurement mode**. Valid pulses are pulses which are in the same angle window with both opening and closing edge. The total calculated is transferred to the DPRAM at the upper angle window limit. The measurement procedures "Total Number of L-Pulses" and "Total Number of H-Pulses" measure the total of all valid pulses since the last initialization or configuration of the ES1334.2 in angle-synchronous measurement mode. The total calculated is updated at each upper angle window limit (Fig. 9-28).

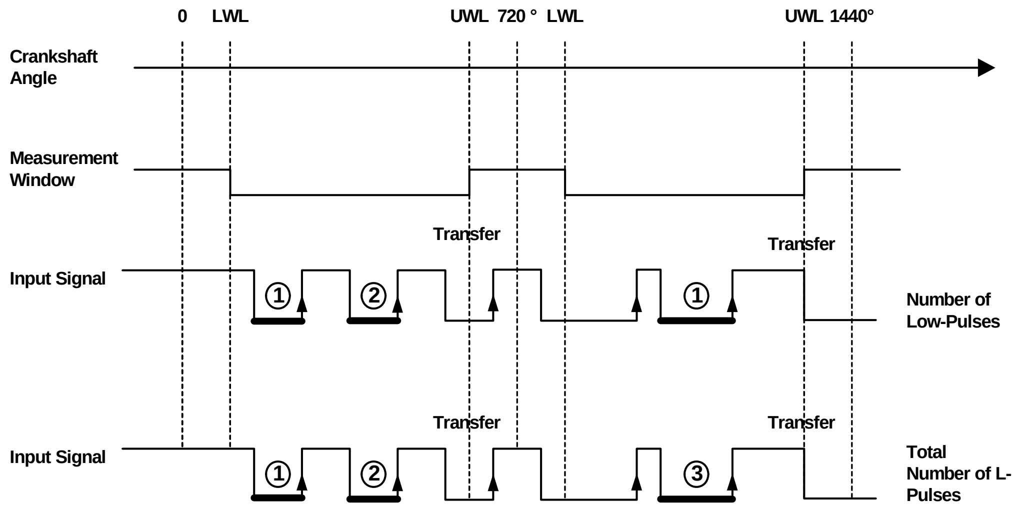


Fig. 9-28 Illustration of how angle-synchronous measurement procedures for pulse count (measurement procedure "Number of Low-Pulses") work. Thicker lines denote the pulses counted. The total is transferred to the DPRAM at the upper angle window limit.

In **asynchronous measurement mode** the "Number of ..." and "Total Number of ..." measurements have identical functionality. The total number of active pulses since the last initialization or configuration of the ES1334.2 is measured. The total is updated at each closing pulse edge.

Note

The pulse count takes place with modulo 2^{52}, i.e. after (2^{52} - 1) pulses, the total returns to 0. (2^{52} - 1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

Timeout Monitoring

A timeout is initiated if no valid active pulse has occurred since the last timeout check. In angle-synchronous measurement mode, a valid active pulse is an active pulse whose opening and closing edges are within one and the same angle window. In asynchronous measurement mode, every active pulse is a valid, active pulse. The measurement value issued in a timeout is specified in Tab. 9-19.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the high level when low pulses are counted and the low level when high pulses are counted. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the low level when low pulses are counted and the high level when high pulses are counted.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-19 Pulse count: measurement value in a timeout with different timeout monitoring procedures

9.5.10 Level Measurement

Functional Description

The measurement procedures

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

are used to acquire the active level states on a hardware channel. An active level is signalized by the measurement value "1"; an inactive level is shown by the measurement value "0". In **asynchronous measurement mode**, the measurement value is updated at each edge of the input signal. In **angle-synchronous measurement mode**, updating only takes place at signal edges which occur within the angle window.

Admissible measurement options are:

- "Measurement Mode": angle-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in angle-synchronous measurement mode
- timeout monitoring

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 9-20.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes pl ace, "0" is issued as measurement value. - If the signal level is active when the timeout check takes place, "1" is issued as measurement value
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 9-20 Level measurements: measurement value in a timeout with different timeout monitoring procedures

9.5.11 Position Tracing on Two-Phase Stepper Motors

Functional Description

The ES1334.2 firmware makes position tracing possible and thus basically the measurement of rotor position and speed of two-phase stepper motors. Two hardware channels and two software or measurement channels are necessary for this purpose. The control signals of the two phases of a two-phase stepper motor are applied to the two hardware channels. The "Step Count (Step. Mot. Phase A)" measurement procedure must be allocated to one measurement channel and "Step Count (Step. Mot. Phase B)" to the other measurement channel. The "Hardware Channel" field of the measurement channels contains a reference to the number of the hardware channel at which the control signal for the relevant phases of the two-phase stepper motor is pending. In the "Reference Channel" field of one measurement channel, there must be a reference to the number of the other measurement channel involved in the measurement. Fig. 9-29 on page 229 shows the procedure involved in the configuration. The control signals of the phases are pending at hardware channels 2 and 3. The measurements are made at measurement channels 0 and 1. Tab. 9-21 on page 230 contains the configuration guidelines.

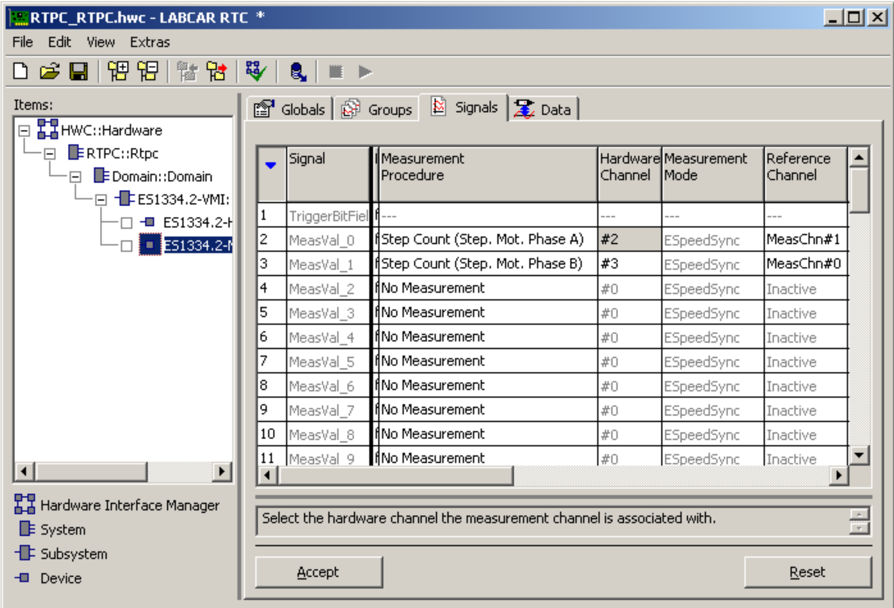


Fig. 9-29 Configuration of a Measurement for Position Tracing with Two-Phase Stepper Motors

On each edge of the two-channel signal, a position counter is incremented or decremented (Fig. 9-30) and the value of the position counter updated in the DPRAM of the ES1334.2. The measurement channel to which the measurement procedure "Step Count (Step. Mot. Phase A)" is assigned always returns a non-negative position value. In the measurement value of the measurement channel to which the measurement procedure "Step Count (Step. Mot. Phase B)" is assigned, the direction of rotation of the motor is coded in the sign of the position value. You obtain an ascending position counter when the signal at "Phase A" is before the signal at "Phase B".

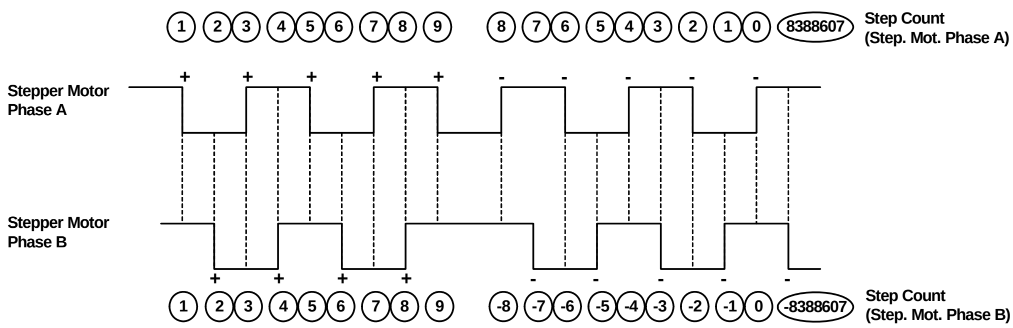


Fig. 9-30 How Position Tracing on Two-Phase Stepper Motors Works

Note

Position tracing takes place modulo 2^{23}. Incrementing at a position level of (2^{23}-1) generates a position counter value of 0. Decrementing with a position level of 0 generates a position counter value which in terms of the amount is (2^{23}-1).

RTIO Parameters of the "Signals" Tab	Phase A	Phase B
Hardware Channel	Number of the hardware channel at which the control signal for phase A is pending.	Number of the hardware channel at which the control signal for phase B is pending.
Measurement Mode	No significance	No significance
Measurement Procedure	"Step Count (Step. Mot. Phase A)"	"Step Count (Step. Mot. Phase B)"
Reference Channel	Number of the measurement channel to which the phase B signal was assigned.	Number of the measurement channel to which the phase A signal was assigned.
Measurement Value	Current value of the position counter modulo 2^{23}	Current value of the position counter modulo 2^{23}. The direction of rotation of the engine is coded in the sign.

Tab. 9-21 Configuration guidelines and measurement values of the measurement procedures for position tracing on two-phase stepper motors

Timeout Monitoring

The measurement procedures for position tracing on two-phase stepper motors have no timeout monitoring.

10 ESX335.1 Arbitrary Signal Generator Board

An ESX335.1 Arbitrary Signal Generator Board has six signal outputs which can be used flexibly – every output can be assigned one of the available internal signals. The outputs of six arbitrary signal generators and of a knock generator with four internal outputs are available as internal signals.

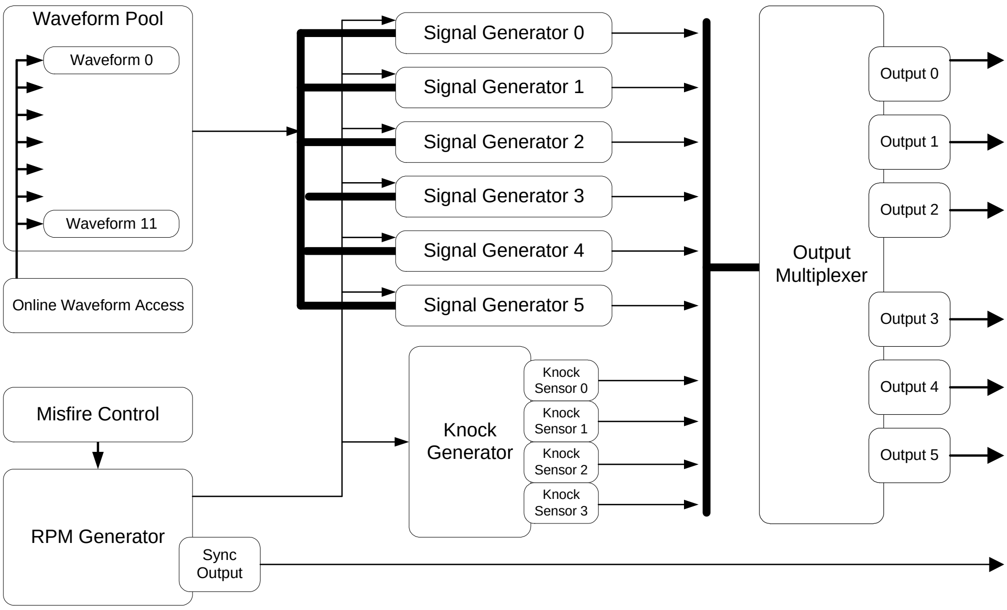
The six arbitrary signal generators can be clocked using a central speed generator (RPM generator) or an individual local frequency. An individual phase shift is possible with each of the arbitrary signal generators.

There are twelve waveforms available centrally any of which can be read out and output by the six signal generators. The maximum resolution is 65536 data points. The waveforms can be modified online via the running simulation model.

The speed can be modulated via misfire control. This makes angle-related speed variations possible to be able to simulate misfiring, for example.

The knock signal generator generates the structure-borne noise which occurs with a combustion engine due to knocking. The frequency and envelope curve of the knock signal can be configured. A cylinder-specific assignment to one of four internal outputs of the knock signal generator makes it possible to simulate knock signals of more complex engines.

Up to twelve cylinders are supported both with misfiring and with the knock generator.



10.1 Functional Description

10.1.1 Signal Generators

There are six arbitrary signal generators available on the ESX335.1 Arbitrary Signal Generator Board. Each of the signal generators can play back one of the twelve waveforms available centrally. A central RPM generator and one local clock generator per signal generator (maximum frequency: 1 MHz) are available as clock sources.

One individual basic phase as well as an additional phase shift can be controlled per signal generator. The speed at which a change of the phase shift takes effect can be defined.

When using the local clock generator, the frequency of the clock generator, the trigger mode (single shot, continuous) and a trigger signal can be specified.

The amplitude of the internal output signal of the signal generator can be varied between 0.0 and 1.0.

10.1.2 MSA Sensor

Signal generators are also used to simulate crankshaft sensors which can detect the direction of rotation (MSA sensors). A tooth pulse has no fixed angle width but a fixed pulse duration. Moreover, the output signal is predefined as being a low-active open collector signal.

If an MSA sensor RTIO element is used, (potential) tooth center information is calculated for all waveform traces during configuration and stored in the waveform pool. However, not all waveforms are necessarily suitable for this algorithm; when an unsuitable waveform is selected, an error message is issued.

10.1.3 RPM Generator

The ESX335.1 Arbitrary Signal Generator Board has a central speed generator (RPM generator) which outputs an engine-speed-specific clock signal. This clock signal can be used by the signal generators to read out and output the waveforms. The maximum speed is 30000 rpm, the resolution in around 0.1 rpm. The speed signal itself can be modulated using a misfire generator.

It is possible to connect several ESX335.1 Arbitrary Signal Generator Boards. Exactly one of the ESX335.1s used must be configured as the RPM master; the other ESX335.1s must be configured as RPM slaves. The RPM master controls the speed for all ESX335.1s in the system. If only one ESX335.1 is being used in the system, it must be configured as RPM master.

For measuring purposes, the speed signal can be applied to the "SYNC" port (on the front panel) of the ESX335.1 (see "Sync Port" on page 246).

10.1.4 Angular Resolution

The maximum angular resolution is 65536 points per cycle. With a typical four-stroke engine with a period of 720 °CA, this corresponds to an angular resolution of around 0.01 °CA.

The resolution can be reduced to 16 points in powers of two – this corresponds to an angular resolution of approx. 45 °CA.

10.1.5 Waveform Pool for Signal Generators

There are twelve waveforms available which can be used by the six arbitrary signal generators. The user can decribe the waveforms with tables. The signal trace in the table is written to the relevant waveform using an interpolation procedure.

Waveform resolution:

The maximum resolution of a waveform is determined by the maximum possible number of 65536 data points. Here too, the resolution can be reduced to 16 points in powers of two; please note that the resolution (1/(number of data points)) of a waveform must be smaller than or equal to the angular resolution. Normally the resolution of a waveform should correspond to the angular resolution.

The waveforms are read out and output by the signal generators. Either the central RPM generator can act as clock source or a local frequency generator (maximum frequency: 1 MHz) in the signal generator is used.

Waveform resolution smaller than angular resolution:

If a high-frequency signal is to be output via the signal generator (using the local frequency generator), it might be necessary to keep the resolution of one waveform smaller than the angular resolution.

The following example illustrates the procedure:

If a sinusoidal signal of 40 kHz is to be output, the signal table describes a single sine period. The angular resolution is 65536 points. Due to a maximum frequency of the local clock generator of 1 MHz, the maximum signal frequency for the sinusoidal signal is 1 MHz/65536 = 15.25 Hz which, of course, is considerably less than the desired 40 kHz. By reducing the waveform resolution to, for example, 16 data points, the sinusoidal signal is stored several times in succession (in fact 65536/16 = 4096-fold) in the waveform with 65536 data points. This results in a total maximum frequency for the sinusoidal signal of 1 MHz/16 = 62.5 kHz, which is above the desired frequency of 40 kHz. Due to a corresponding reduction of the local clock rate to 640 kHz, the desired sinusoidal signal can be generated with 40 kHz.

The example shows that due to a reduction in the waveform resolution in comparison to the angular resolution, the waveform resolution is not really reduced. The signal of the signal table is simply written to the waveform several times in succession and the "visible" resolution thus reduced.

Direct Waveform Access from the Simulation

There is a special form of access for modifying the waveforms in runtime from the running simulation model.

Up to 16 data points of one of the twelve available waveforms can be modified per simulation cycle. The data point number, the total number of data points to be modified and the new data values are used for the description.

Output Multiplexer

The output multiplexer enables the assignment of the internal signals to the six available physical outputs of the ESX335.1. The output driver (analog, digital without pull-up resistance, digital with pull-up resistance, open collector) as well as the reference voltage of the D/A converter (internal or external reference voltage) can be selected. If necessary, an internal signal can be applied to several physical outputs with varying circuits.

Please note that the internal signals have a value range of -1.0 to +1.0. A real voltage signal is generated from the internal signal by the D/A converter in the output multiplexer.

10.1.6 Knock Signal Generator

The knocking which occurs with a combustion engine can be simulated by the knock signal generator. A knock signal consists of individual knock packages. A knock package itself consists of a sinusoidal oscillation with selectable frequency and an envelope curve which modulates the sinusoidal oscillation with a duration which can be defined.

The following figure shows an individual knock package. A sine half wave is used as an envelope curve.

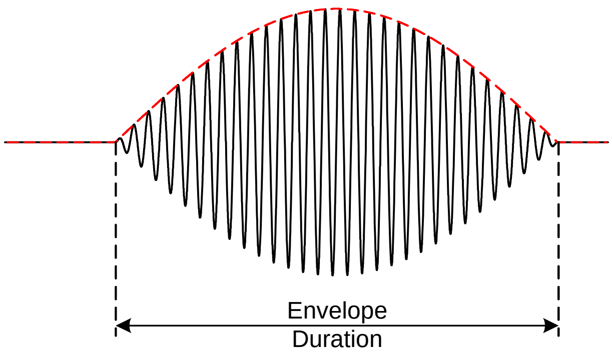


Fig. 10-1 A Knock Package

Non-knocking combustion also generates noises which are acquired by a real structure-borne noise knock sensor. A distinction is made between correct and knocking combustion via the control of the amplitude of the knock signal.

In addition, there is also a stochastic variation of the amplitude of a knock package. This is used for the simulation of variations in the knock signals which occur in real operation.

The angular position (in °CA) of a knock signal as well as the occurrence of the knock event can now be controlled individually for each cylinder using a probability value or sequence tables (see "Sequence Tables" on page 235).

The knock signal generator has four internal outputs. You can select which cylinders serve the relevant output. Please note that, particularly with four-cylinder engines, a knock event at one output interrupts a knock signal at the same output which is not entirely complete.

10.1.7 Misfire Control

A relevant control mechanism is available on the ESX335.1 Arbitrary Signal Generator Board to simulate misfiring; this results in a modulation of the speed of the RPM generator in a specific angle range. It is possible to modify the speed in relation to the specified speed of the RPM generator (reduce/increase by the factor 0.01 to 2.0). When simulating misfiring, the speed is normally reduced in comparison to the defined speed.

The start effect angle of speed modulation can be defined for each individual cylinder. The effect of speed modulation can be controlled for each cylinder using a probability value or sequence tables (see "Sequence Tables" on page 235).

Speed modulation can be defined via four modulation profiles which represent the course of modulation over a complete period of 720 °CA. 1.0 represents a non-existent modulation; 0.01 reduces the speed to 1% of the specified speed; 2.0 doubles the specified speed. One of the four available modulation profiles can be selected individually per cylinder.

10.1.8 Sequence Tables

Sequence tables are used with the misfire generator and the knock signal generator: these make it possible for the user to describe complex knock and misfiring sequences.

A table with a maximum of 100 data points is used for this purpose. Once the sequence has been started, the sequence proceeds one data point per period. A value greater than 0.5 at the relevant data point means that one event (knocking or misfiring) occurs in this period.

After 100 data points, the sequence is either started from the beginning again, or play-back is terminated and has to be restarted via the relevant trigger signal.

It is possible to specify one individual sequence per cylinder. There is, however, one common sequence ("Common Sequence") both with the misfire generator and the knock generator which all cylinders can access. This facilitates the fast setting of sequences which are to be used for several cylinders.

10.1.9 Structure of the ESX335.1 RTIO Tree

The ESX335.1 Arbitrary Signal Generator Board is integrated in the RTIO Editor by selecting an ESX335 Subsystem followed by an ESX335-Powertrain Subsystem.

10.2 ESX335 Subsystem

10.2.1 Globals (ESX335 Subsystem)

The ESX335 Subsystem is used to set parameters which are valid globally, i.e. which have an effect on all ESX335.1 RTIO elements.

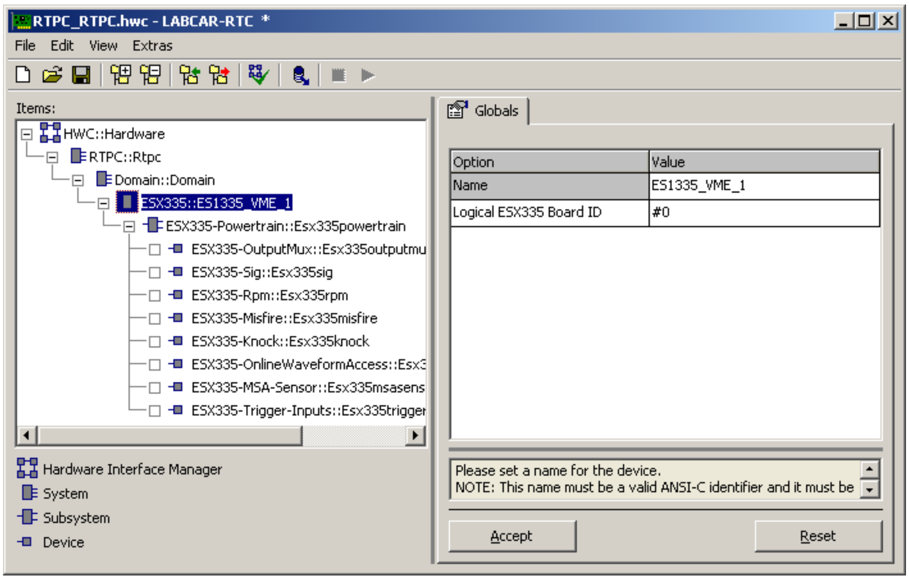


Fig. 10-2 The "Globals" Tab of the ESX335 Subsystem

Locical ESX335 Board ID

This option field is used to identify the ESX335.1. It establishes the assignment between the RTIO hardware description and the ESX335.1 in the VMEbus chassis for which this description is valid.

The numbering of the boards in the chassis takes place from left to right (left = VMEbus slot no. 1, with ascending slot numbering) starting with 1. The number of the ESX335.1 determined must be set in the "Logical ESX335 Board ID" list field. In the RTIO Editor, up to 20 ESX335.1s can be integrated per chassis.

This RTIO parameter cannot be edited online.

10.3 ESX335-Powertrain Subsystem

10.3.1 Globals (ESX335-Powertrain Subsystem)

The ESX335-Powertrain Subsystem is used to set RTIO parameters which are effective globally, i.e. which have an effect on all elements of the ESX335-Powertrain Subsystem.

The following figure shows the RTIO parameters of the "Globals" tab.

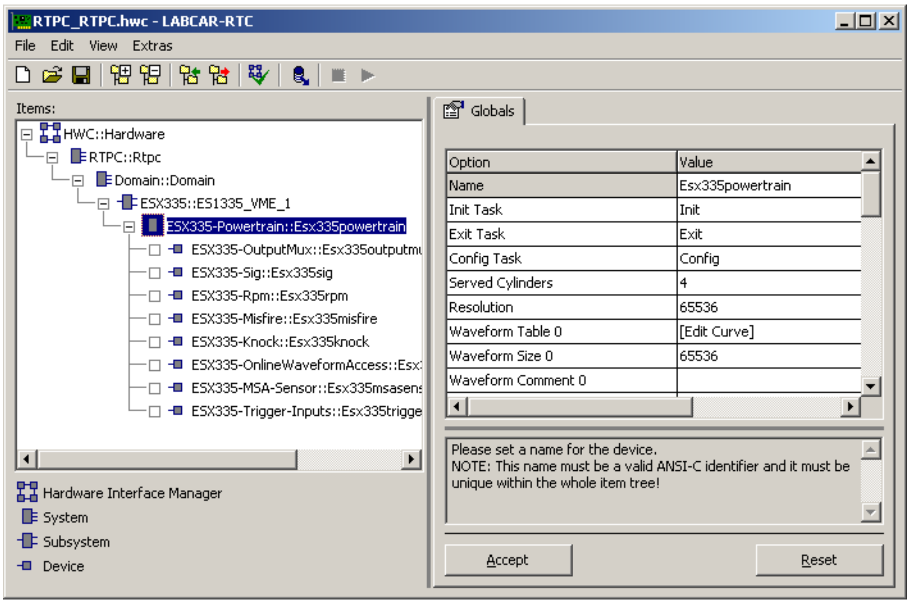


Fig. 10-3 The "Globals" Tab of the ESX335-Powertrain Subsystem

Served Cylinders

This is where the number of cylinders supported by the knock signal generator and with misfire control is set. Possible values are 1 ... 12.

Resolution

This option field is used to define the angular resolution. Values from 16 to 65536 in powers of two are possible.

Waveform Table 0…11

This is where the signal shapes of the twelve waveforms can be specified.

Waveform Size 0…11

The number of data points of the twelve waveforms can be specified using the corresponding options fields (see "Waveform resolution" on page 233 and "Waveform resolution smaller than angular resolution" on page 233).

Waveform Comment 0…11

An individual comment can be specified to obtain a simple overview of how the twelve waveforms are used.

Note

The comment field does not permit blanks or special characters.

MSA Sensors: Logic Level of Teeth

This parameter determines whether a signal with negative logic (neg. pulse corresponds to "tooth") or positive logic (pos. pulse corresponds to "tooth") is to be interpreted when the tooth centers (for an MSA sensor) are calculated. This setting applies to all twelve waveforms.

The parameter can be edited online; however, when a change occurs, all waveforms must be recalculated and written to the board in the waveform pool.

Note

This parameter is only displayed if at least one ESX335-MSA-Sensor device has been added.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Served Cylinders	uint32	Yes	Supported cylinders for knocking and misfiring Value range: 1 … 12
Resolution	uint32	Yes	Angular resolution Value range: 16 … 65536 in powers of two
Waveform Table 0…11	Table	Yes	The relevant waveform as table Value range: -1.0 … +1.0
Waveform Size 0…11	uint32	Yes	Number of data points of the relevant waveform Value range: 16 … 65536 in powers of two
Waveform Comment 0…11	Identifier	No	Comment on the relevant waveform
MSA Sensors: Logic Level of Teeth	bool	Yes	Interpretation of wave form for MSA sensor Value range: 0=FALSE (neg. logic) |, 1=TRUE (pos. logic)

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-1 ESX335-Powertrain Subsystem: Configuration Parameters of the "Globals" Tab

10.4 ESX335-OutputMux Device

10.4.1 Globals (ESX335-OutputMux Device)

The ESX335-Mux device is used to configure and control the output multiplexer of the ESX335.1. This cannot be set in the "Globals" tab, output-specific settings are made in the "Signals" tab.

10.4.2 Groups (ESX335-OutputMux Device)

The ESX335-OutputMux has one signal group for controlling the output relays.

10.4.3 Signals (ESX335-OutputMux Device)

The six output multiplexers are configured in the "Signals" tab. The following figure shows the RTIO parameters of the "Signals" tab. Tab. 10-2 on page 240 summarizes the properties of the individual parameters. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

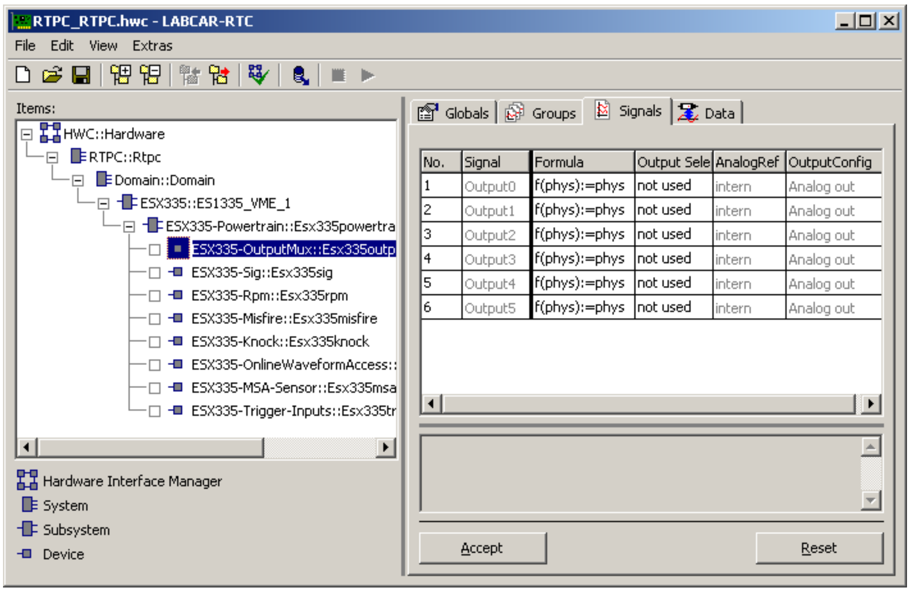


Fig. 10-4 The "Signals" Tab of the ESX335-OutputMux Device

Output Select

The option field "Output Select" is used to select the internal signal which is to be applied to the particular output. A list of all possible internal signal sources (signal generators, knock signal generator) is offered for selection. A knock signal generator has four internal outputs numbered #0 to #3.

AnalogRef

In this option field, you can select the reference voltage for the D/A converter of the relevant output channel (see "Output Multiplexer" on page 233). Possible values are "internal" for an internal reference voltage of 10 V and "external" for a reference voltage applied externally.

OutputConfig

The output driver of the relevant channel can be selected in this option field (see "Output Multiplexer" on page 233). Possible values are "Analog out" for the analog output stage, "Digital out with pull-up" for a digital output stage with an internal pull-up resistance against 5 V as well as "Digital out without pull-up" for a digital output stage without an internal pull-up resistance.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Output Select	uint8	Yes	Selection of an internal signal 0-5: Signal generator 6-9: Knock generator #0 - #3
Analog Ref	uint8	Yes	Reference voltage for D/A converter 0: Internal 1: External
OutputConfig	uint8	Yes	Output driver 0: analog 1: Digital without pull-up 2: Digital with pull-up
* Data type which the RTIO driver uses internally for the parameter			

Tab. 10-2 ESX335-OutputMux Device: Configuration Parameters of the "Signals" Tab

10.4.4 Data (ESX335-OutputMux Device)

The six RTIO signals in this group control the closing of the output relay of each channel.

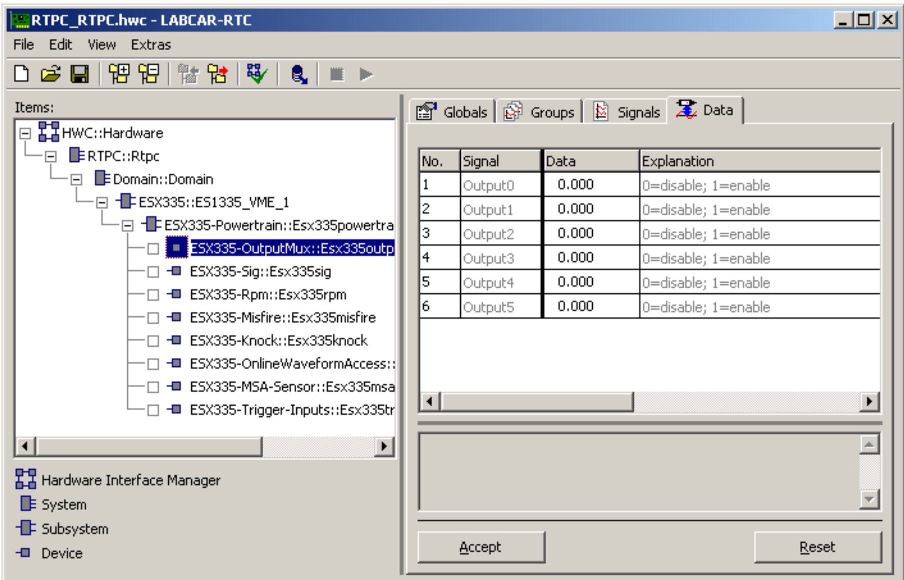


Fig. 10-5 The "Data" Tab of the ESX335-OutputMux Device

The following table summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Output0 ... Output 5	uint8	Enabling/disabling the outputs 0 - 5 0: Disabling the output 1: Enabling the output
* Data type which the RTIO driver uses internally for the parameter		

Tab. 10-3 ESX335-OutputMux RTIO Element: The RTIO Signals of the "Data" Tab

10.5 ESX335-Sig Device

The ESX335-Sig device is used to configure and control one of the six arbitrary signal generators on the ESX335.1 Arbitrary Signal Generator Board.

10.5.1 Globals (ESX335-Sig Device)

The ESX335-Sig device is used to configure and control one of the six arbitrary signal generators on the ESX335.1 Arbitrary Signal Generator Board.

The following figure shows the RTIO parameters of the "Globals" tab.

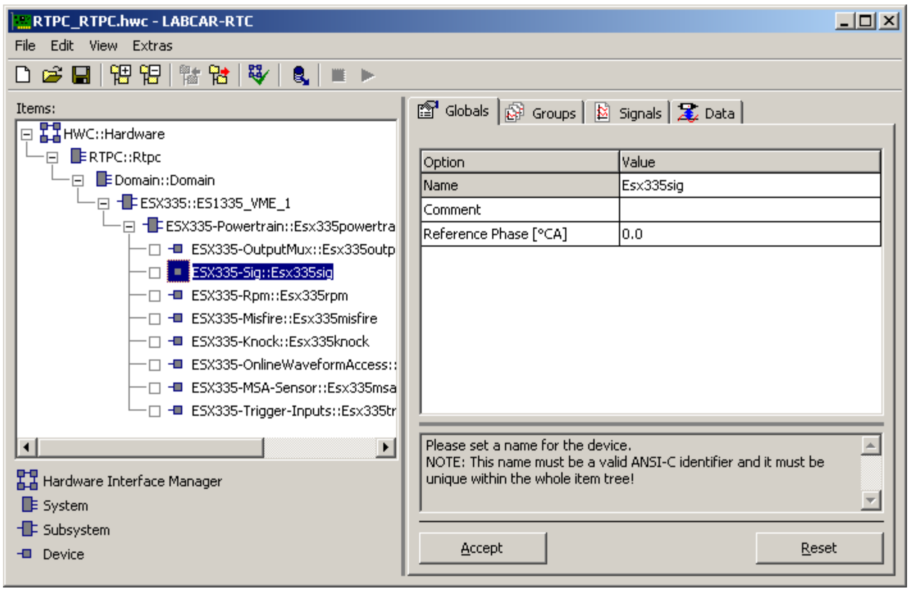


Fig. 10-6 The "Globals" Tab of the ESX335-Sig Device

Comment

A comment can be made for this signal generator.

Note

The comment field does not permit blanks or special characters.

Reference Phase

This parameter describes the basic phase shift of the signal generator. The actual phase is determined by adding this value to the "PhaseRef" signal (see "PhaseRef" on page 243).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Comment	Identi-fier	No	User-specific comment
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range 0 … 720 °CA

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-4 ESX335-Sig Device: Configuration Parameters of the "Globals" Tab

10.5.2 Groups (ESX335-Sig Device)

The ESX335-Sig device has one signal group which is used to control the signal generator.

10.5.3 Data (ESX335-Sig Device)

The "Data" tab lists the signals for controlling the signal generator.

The following figure shows the signals in the "Data" tab:

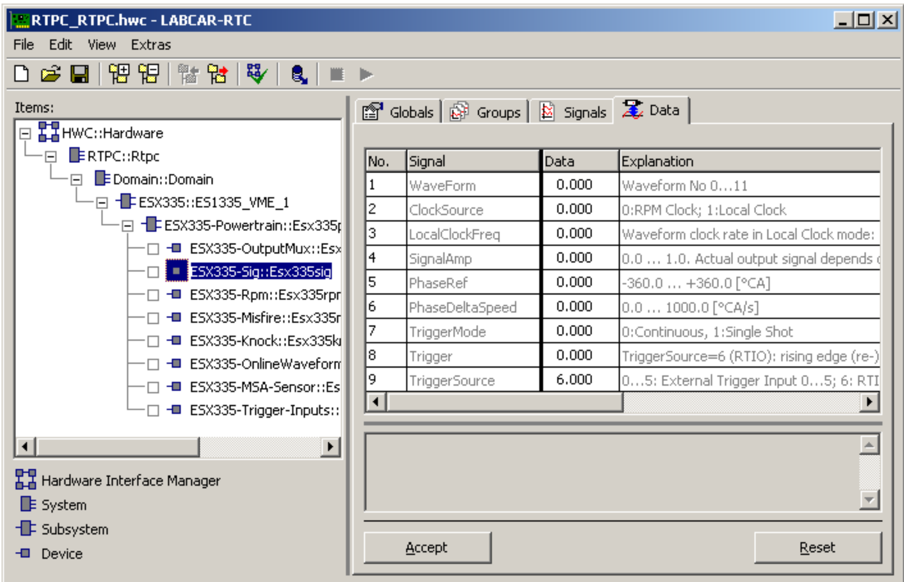


Fig. 10-7 The "Data" Tab of the ESX335-Sig Device

WaveForm

The "WaveForm" signal is used to select the waveform to be read out and output by the signal generator. The values 0 to 11 are possible. These correspond to the central waveforms in the ESX335-Powertrain Subsystem.

ClockSource

The "ClockSource" signal is used to determine the clock source for the signal generator. You can choose between the central speed generator (RPM generator) and a local clock generator. The value 0 corresponds to the central speed generator; the value 1 to the local clock generator.

LocalClockFreq

The "LocalClockFreq" signal controls the frequency of the local clock generator. The signal is only effective if the "ClockSource" signal is 1 (= local clock generator). The maximum frequency of the local clock generator is 1 MHz.

SignalAmp

The output amplitude of the signal generator is controlled via the "SignalAmp" signal. The values range from 0.0 to 1.0.

Please note that a voltage signal is not created until an internal signal is within the value range -1.0 to +1.0.

PhaseRef

This signal describes a phase shift which is added to the global parameter "ReferencePhase". This makes a model-controlled phase shift during runtime possible. Please note that the phase shift is takes place with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeed"). This is why the "PhaseDeltaSpeed" signal must have a value greater than 0 for most applications.

PhaseDeltaSpeed

The phase change speed is controlled by the "PhaseDeltaSpeed" signal. A change of the value of the "PhaseRef" signal does not have an infinitely fast effect; rather the phase is calibrated with this speed. The values range from 0 (no calibration of the phase) to 1000 °CA/s.

TriggerMode

Note

The "TriggerMode" signal is only of importance if the local clock generator is selected as clock source for the signal generator (see "ClockSource" on page 242).

The local clock generator can work continuously ("TriggerMode" is 0) or it stops once a complete waveform has been played back once ("Single Shot") ("TriggerMode" is 1).

A phase shift is also taken into consideration during operation with the local clock generator. This controls the starting point of play-back.

The trigger signal "Trigger" results in a start or new start.

Please note that the complete waveform is always played back over the total number of all data points. If the number of data points of the waveform is selected as being smaller than the angular resolution, the required output signal is stored several times in succession in the waveform pool. The required output signal is then played back several times when play-back takes place in "Single Shot" mode (see "Waveform resolution smaller than angular resolution" on page 233). This is why it is usually a good idea to keep the resolution (= 1/number of data points) of the waveform the same as the angular resolution for the "Single Shot" application.

Trigger

Note

The "TriggerMode" signal is only of importance if the local clock generator is selected as clock source for the signal generator (see "ClockSource" on page 242).

The "Trigger" signal enables you to start the local clock generator. An edge of 0 → 1 on the "Trigger" signal starts the reading out and outputting of the waveform with the local clock generator. In all cases, a corresponding edge leads to reading out and outputting being restarted even if the previous play-back has not been completed or is in continuous mode.

TriggerSource

Note

The "TriggerMode" signal is only of importance if the local clock generator is selected as clock source for the signal generator (see "ClockSource" on page 242).

This signal is used to select whether the signal generators are triggered via software (LABCAR-RTC) or via the external inputs of the PB1335TRIG Module.

When using external triggers, "Triggersource" must be set to 1 – at the same time, the value of "LocalClockFreq" must not be zero.

Note

The "TriggerSource" signal only exists with newly generated ESX335-Sig Devices – existing devices are not automatically extended and the internal value remains as 6 (no external trigger) even after code has been generated once again. If you want to work with the "TriggerSource" signal, you must delete the existing ESX335-Sig device and then add it again.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
WaveForm	uint8	Yes	Selection of the waveform 0 - 11
ClockSource	uint8	Yes	Clock source of the signal generator 0: Speed generator 1: Local clock generator
LocalClockFreq	real32	Yes	Frequency of the local signal generator in Hertz. 0.0 - 1000000.0 [Hz]
SignalAmp	uint16	Yes	Signal amplitude of the output signal 0…65535 corresponds to 0.0…1.0
PhaseRef	sint16	Yes	Additive phase shift -32768…32767 corresponds to -360…+360 [°CA]
PhaseDeltaSpeed	uint16	Yes	Phase change speed 0 … 1000 [°CA/s]
TriggerMode	uint8	Yes	Trigger mode 0: Continuous 1: Single-Shot
Trigger	uint8	Yes	Trigger signal Edge of 0 → 1 starts play-back
TriggerSource	uint8	Yes	Trigger source Value range: 0...6 0...5 = Ext. trigger signal 6 = LABCAR-RTC (default)
* Data type which the RTIO driver uses internally for the parameter			

Tab. 10-5 ESX335-Sig Device: Signals of the "Data" Tab

10.6 ESX335-Rpm Device

The ESX335-Rpm device is used to configure and control the speed generator on the ESX335.1 Arbitrary Signal Generator Board.

10.6.1 Globals (ESX335-Rpm Device)

The following figure shows the RTIO parameters of the "Globals" tab.

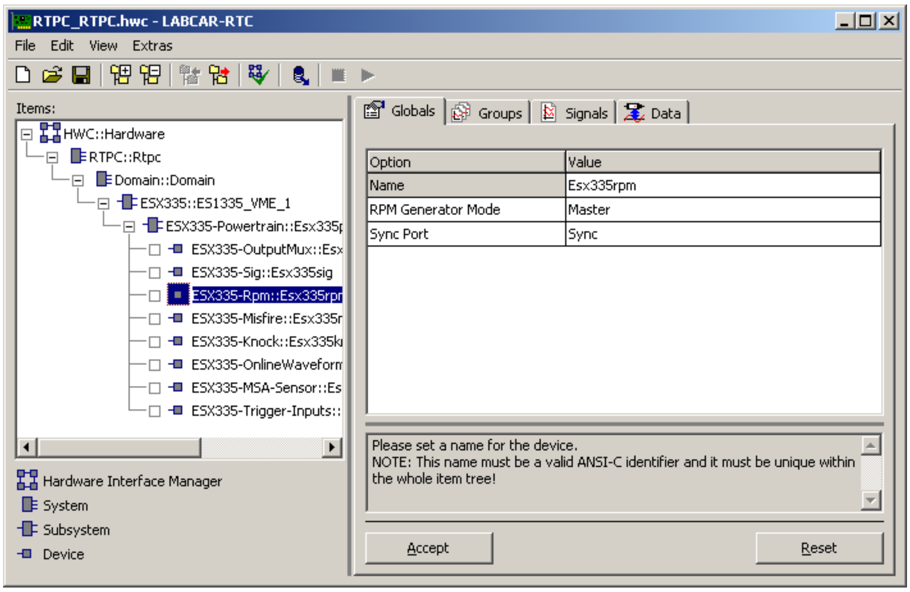


Fig. 10-8 The "Globals" Tab of the ESX335-Rpm Device

RPM Generator Mode

If several ESX335.1 Arbitrary Signal Generator Boards are to be used in a VMEbus system, one must be configured in the RPM Generator Mode "Master"; all other boards are to be configured as slaves. The board configured as "Master" specifies the speed and the angular information for all boards in the system. If you are using a single ESX335.1, the RPM Generator Mode must be set to "Master".

Sync Port

The "SYNC" port on the front panel of the ESX335.1 Arbitrary Signal Generator Board can be assigned different, speed-specific signals. This facilitates startup and/or can be used for diagnostic purposes. The output signal at the "SYNC" port is a 5 V digital signal.

The following settings are possible:

- Sync: A short high signal is issued at around 0 °CA.
- Clock: The speed clock signal with which the waveforms of the connected signal generators are read out and output. The next data point of the waveform is read out and output both with a rising and with a falling edge of the signal.
- Direction: Displays direction - a 5 V signal corresponds to the speed generator turning in a positive direction. A negative direction of rotation is indicated by 0 V.

- Engine Speed: Engine speed signal – the signal is at 5 V with a current crankshaft angle between 0 and 360 °CA and at 0 V with a current crankshaft angle between 360 and 720 °CA.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
RPM Generator Mode	uint8	Yes	0: Slave, 1: Master
Sync Port	uint8	Yes	Assignment of the "SYNC" port: 0: Sync 1: Clock 2: Direction 3: Engine Speed

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-6 ESX335-Rpm Device: Configuration Parameters of the "Globals" Tab

10.6.2 Groups (ESX335-Rpm Device)

The ESX335-Rpm device has a signal group called "Outputs" for setting the current speed and a signal group called "Inputs" for reading the current crankshaft angle from the ESX335.1 Arbitrary Signal Generator Board.

10.6.3 Data (ESX335-Rpm Device)

The "Data" tab lists the signals for communicating with the speed generator. The following figure shows the signals in the "Data" tab.

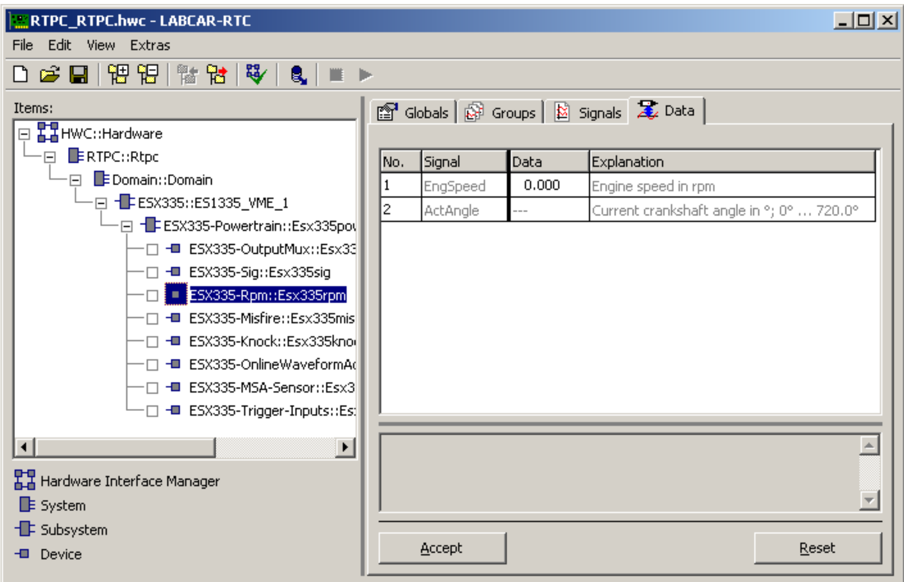


Fig. 10-9 The "Data" Tab of the ESX335-Rpm Device

EngSpeed

The engine speed, i.e. the clock frequency of the speed generator, is controlled using the "EngSpeed" signal. The speed is specified in revolutions per minute (rpm) up to a maximum value of 30000 rpm.

An engine rotating backwards can be simulated by entering a negative speed.

ActAngle

The signal "ActAngle" delivers the current crankshaft angle of the speed generator – the value range is between 0 and 720 °CA.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
EngSpeed	real32	Yes	Engine speed in [rpm] Value range: -30000 … +30000 [rpm]
ActAngle	real32	---	Current crankshaft angle

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-7 ESX335-Rpm Device: Configuration Parameters of the "Data" Tab

10.7 ESX335-Misfire Device

The ESX335-Misfire device is used to configure and control the misfire control on the ESX335.1 Arbitrary Signal Generator Board.

10.7.1 Globals (ESX335-Misfire Device)

The following figure shows the RTIO parameters of the "Globals" tab.

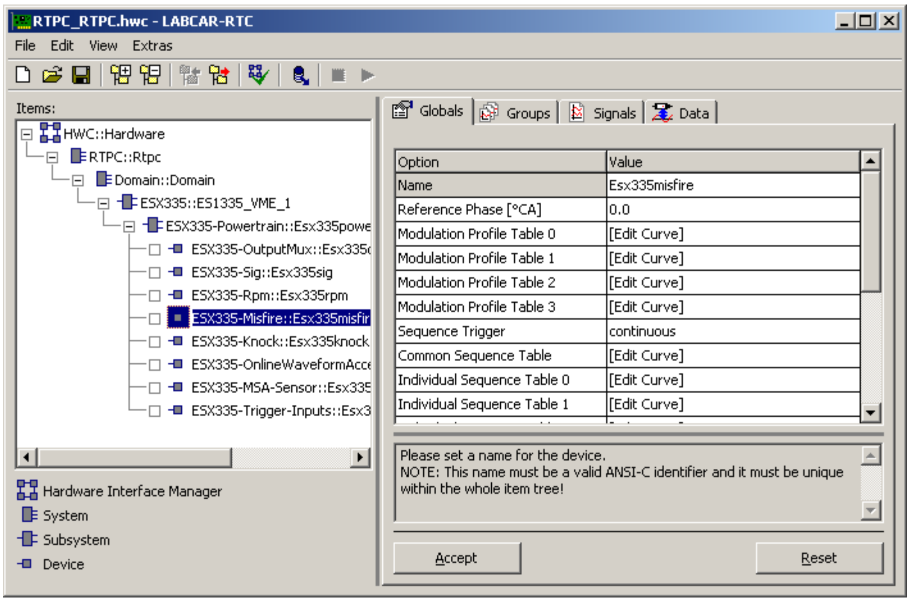


Fig. 10-10 The "Globals" Tab of the ESX335-Misfire Device

Reference Phase

This parameter describes the basic phase shift of the misfire control. The actual cylinder-specific phase is determined by adding this value to the relevant parameter "Start Angle" from the "Signals" tab.

Modulation Profile Table 0 … 3

Four modulation profiles are available which define a modulation of the predefined speed over a specific angle area (see "Misfire Control" on page 234). The modulation profiles can be specified using the table editor. In a modulation curve the speed modulation is plotted against the crankshaft angle. A value of 1 at an angle represents the unchanged acceptance of the predefined speed. A value less than 1 (to 0.01) lowers the speed. A value greater than 1 represents an increase in speed.

The first data point value of the signal table corresponds to the relevant cylinder-specific start angle in °CA. The signal table describes a complete period of 720 °CA.

The speed modulation of several cylinders can overlap – the individual cylinder-related modulation values are multiplied with each other.

Sequence Trigger

The activation of the speed modulation can be controlled for each cylinder using sequence tables (see "Sequence Tables" on page 235). This parameter is used to decide whether the sequence table should be played back continuously ("continuous") or only once ("single shot"). Play-back is started with the signal "Sqnc_Trigger" (see "Sqnc_Trigger" on page 253).

Common Sequence Table, Individual Sequence Table 0 … 11

There are 12 cylinder-individual sequence tables for controlling speed modulation (see "Sequence Tables" on page 235). There is also a "Common Sequence Table" which makes it possible to use one sequence on several cylinders.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range 0 … 720 °CA
Modulation Pro-file 0…3	table	Yes	The data points are mapped to a complete period (720 °CA). The values are within the range 0.01 to 2.0
Sequence Trigger	uint8	Yes	Play-back mode of the sequence tables: Continuous:0 Single Shot: 1
Common Sequence Table Individual Sequence Table 0…11	table	Yes	Sequence tables

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-8 ESX335-Misfire Device: Configuration Parameters of the "Globals" Tab

10.7.2 Groups (ESX335-Misfire Device)

The ESX335-Misfire device has a signal group in which the misfire control signals are processed.

10.7.3 Signals (ESX335-Misfire Device)

Misfire control is configured in the "Signals" tab. The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

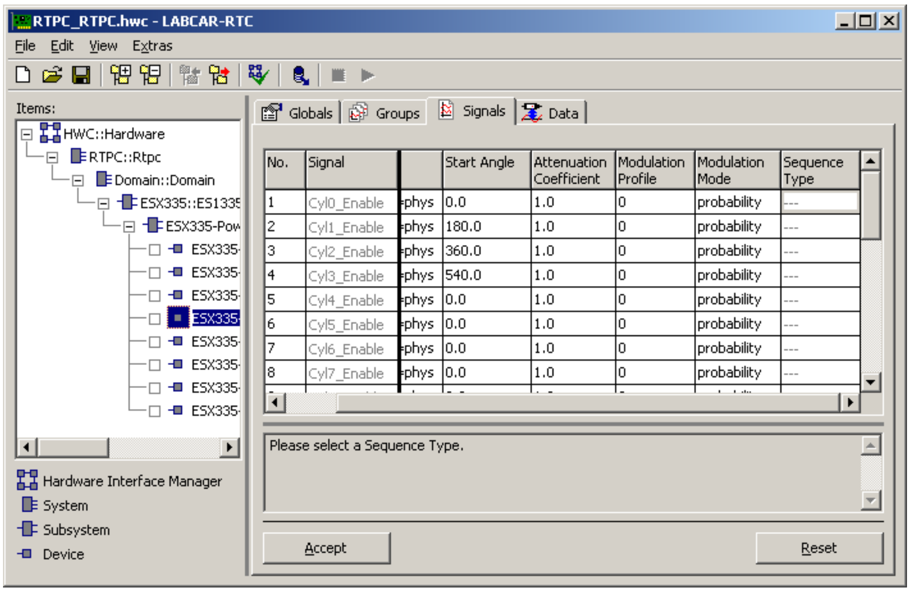


Fig. 10-11 The "Signals" Tab of the ESX335-Misfire Device

Start Angle

The option field "Start Angle" is used to determine the start angle of the misfire modulation profile for the relevant cylinder. The "Reference Phase" parameter from the "Globals" tab (see "Reference Phase" on page 249) is added to this angle.

Attenuation Coefficient

This parameter is used to reduce the degree of speed modulation for each individual cylinder. A value of 1.0 means that the speed modulation described in the selected modulation profile is applied completely. A value less than 1 reduces the effect in the relevant proportion; a value of 0.0 ultimately leads to no speed modulation being executed for the cylinder.

Modulation Profile

One of the four available modulation profiles from the "Globals" tab can be selected per cylinder.

Modulation Mode

The existence of misfiring in a period can be controlled either via a probability function or using sequence tables. You can toggle between these two operating modes using this parameter.

Sequence Type

When selecting "Modulation Mode = sequence", you can specify here whether the common sequence table ("common") or the cylinder-specific sequence table ("individual") is to be used (see "Sequence Tables" on page 235).

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Start Angle	real32	Yes	Start angle of the speed modulation 0.0 … 720.0 °CA
Attenuation Coefficient	real32	Yes	Proportional effect of the modulation profile 0.0 … 1.0
Modulation Profile	uint8	Yes	Selection of the modulation profile for the cylinder 0 … 3
Modulation Mode	uint8	Yes	Probability or sequence table for misfire control 0: Probability 1: Sequence table
Sequence Type	uint8	Yes	Selection: Common or cylinder-individual sequence. 0: Common sequence 1: Cylinder-individual sequence
* Data type which the RTIO driver uses internally for the parameter			

Tab. 10-9 ESX335-Misfire Device: Configuration Parameters of the "Signals" Tab

10.7.4 Data (ESX335-Misfire Device)

The "Data" tab shows the RTIO signals which are available for controlling the misfire unit.

Note

Please ensure that the "Served Cylinders" parameter (see "Served Cylinders" on page 237) from the upstream ESX335-Powertrain Subsystem has been set correctly as this value is used when processing the signals!

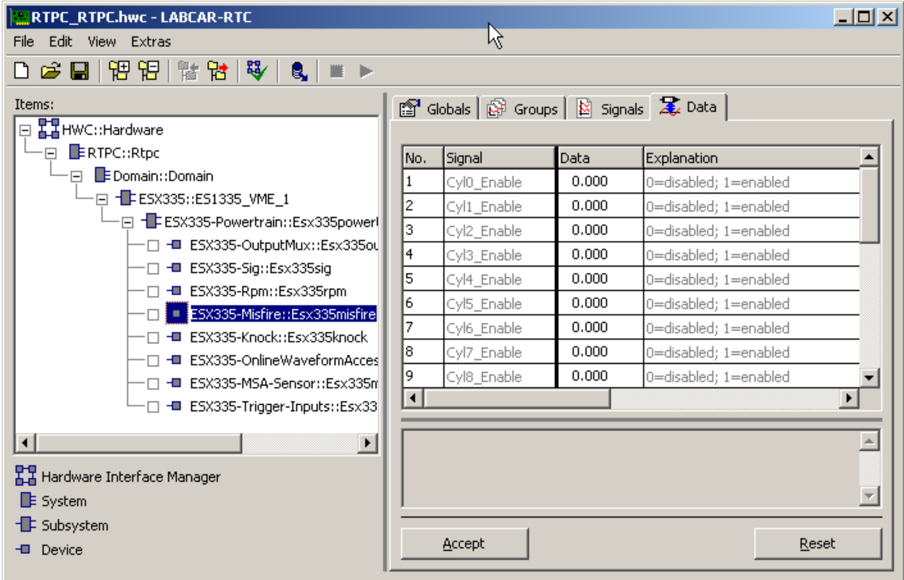


Fig. 10-12 The "Data" Tab of the ESX335-Misfire Device

Cyl0_Enable … Cyl11_Enable

These twelve signals activate the speed modulation for the relevant cylinder.

Cyl0_Probability … Cyl11_Probability

These signals are used to control the probability of the occurrence of misfiring for a cylinder. The signal is only taken into consideration if the relevant cylinder has been configured for using the probability function ("Modulation Mode" on page 251).

Sqnc_Trigger

This signal represents the trigger with which the sequence table play-back is started. A rising edge of 0 → 1 on this signal starts play-back. A sequence which has not run completely is interrupted by a rising edge on the trigger signal and restarted (re-trigger).

The following table summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Cyl0_Enable … Cyl11_Enable	uint8	Enabling/disabling the outputs 0: Disabling the output 1: Enabling the output
Cyl0_Probability … Cyl11_Probability	uint16	Probability of misfiring Value range: 0 to 10000 corresponds to 0 …1.0
Sqnc_Trigger	uint8	Starting or restarting sequence table play-back An edge 0 → 1 is used as trigger
* Data type which the RTIO driver uses internally for the parameter		

Tab. 10-10 ESX335-Misfire Device: The RTIO Signals of the "Data" Tab

10.8 ESX335-Knock Device

The ESX335-Knock device is used to configure and control the knock signal generator on the ESX335.1 Arbitrary Signal Generator Board.

10.8.1 Globals (ESX335-Knock Device)

The following figure shows the RTIO parameters of the "Globals" tab.

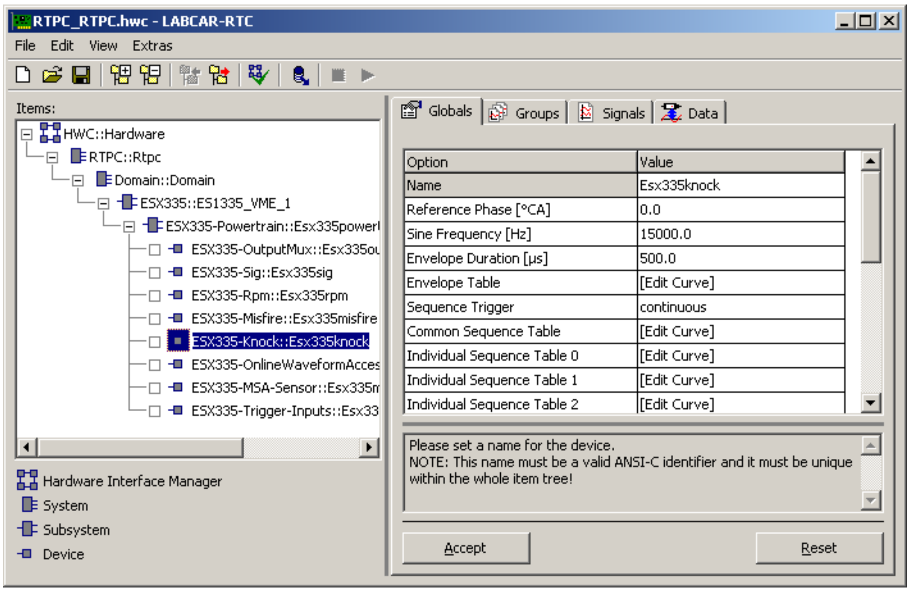


Fig. 10-13 The "Globals" Tab of the ESX335-Knock Device

Reference Phase

This parameter describes the basic phase shift of the knock signal generator. The actual cylinder-specific phase is determined by adding this value to the relevant "Base Angle" parameter from the "Signals" tab (see "Base Angle" on page 257) and the "CylN_AngleOffset" signal (see "Cyl0_AngleOffset … Cyl11_AngleOffset" on page 260).

Sine Frequency

This parameter describes the frequency of the sinusoidal signal of which the knock signal consists. This sinusoidal signal is modulated with an envelope curve.

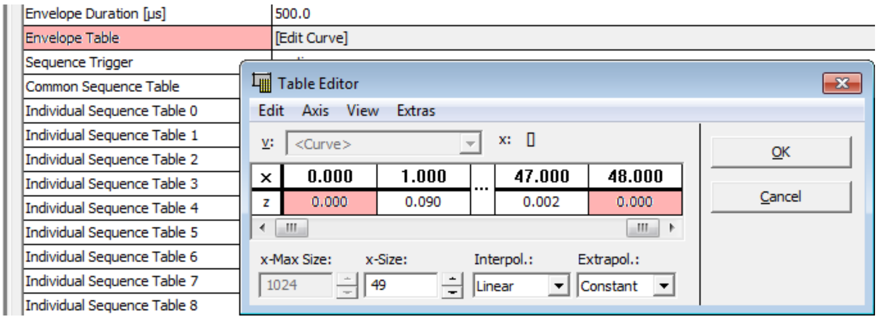
Envelope Duration

This parameter describes the duration of a knock signal in µs.

Envelope Table

The form of the envelope curve is described by this table. The length of the envelope curve is defined by the parameter "Envelope Duration" (see "Knock Signal Generator" on page 234).

Envelope curves must always start and end with the value "0".



Sequence Trigger

With this parameter you can choose between "single shot" and the continuous triggering of the sequence tables. See "Sequence Tables" on page 235.

Common Sequence Table, Individual Sequence Table 0 … 11

There are 12 cylinder-individual sequence tables available for controlling the generation of knock packages (see "Sequence Tables" on page 235). There is also a "Common Sequence Table" which makes it possible to use one sequence on several cylinders.

Cylinder to Sensor Mapping Table 0 … 3

The knock signal generator has four outputs. Each of the four "Cylinder to Sensor Mapping" allocation tables defines which cylinders the generated knock signals are issued from. This makes it possible to output a cylinder on several internal outputs by selecting the relevant cylinder several times in the allocation table.

The tables each consist of up to 12 data points which represent the cylinders 0 to 11. A value of 1.0 at a data point corresponds to the allocation of the relevant cylinder to the relevant output. A value of 0.0 shows that the cylinder is not allocated to the output (see "Sequence Tables" on page 235).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range: 0 … 720 °CA
Sine Frequency	real32	Yes	Sine frequency of the knock packages in the range 1000 - 20000 Hz
Envelope Duration	real32	Yes	Duration of the envelope curve in µs. Value range: 50-12000µs
Envelope Table	table	Yes	Envelope curve signal
Sequence Trigger	uint8	Yes	Selection of the trigger mode for the sequence tables. 0: Continuous 1: Single Shot
Common Sequence Table	table	Yes	Sequence table which can be used for all cylinders together
Individual Sequence Table 0 … 11	table	Yes	Cylinder-individual sequence tables
Cylinder to Sensor Mapping Table 0 … 3	table	Yes	Tables in which the relevant cylinders can be allocated to an output. The data points correspond to the 12 cylinders – a value greater than 0.5 means that the relevant cylinder issues the knock signal at the output.

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-11 ESX335-Knock Device: Configuration Parameters of the "Globals" Tab

10.8.2 Groups (ESX335-Knock Device)

The ESX335-Knock device has a signal group in which the signals of the misfire control are processed.

10.8.3 Signals (ESX335-Knock Device)

The knock generator is configured in the "Signals" tab. The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

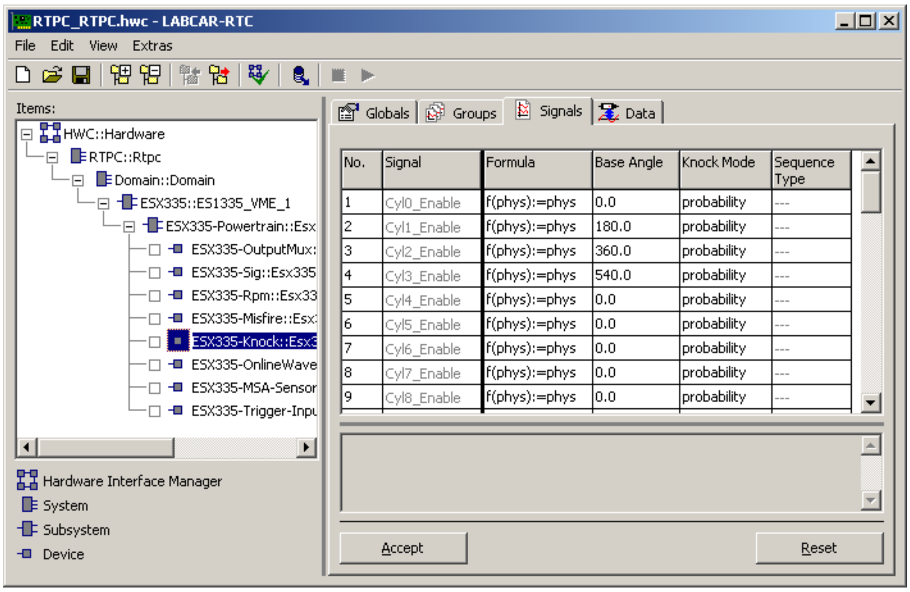


Fig. 10-14 The "Signals" Tab of the ESX335-Knock Device

Base Angle

The option field "Base Angle" is used together with the parameter "Reference Phase" from the "Globals" tab as the base angle for controlling the start angle of a knock package. The cylinder-specific signal "Cylx_AngleOffset" is added to this start angle enabling modification during runtime. Please note that the signal "Cylx_AngleOffset" can only be varied over an angle range of ±127 °CA.

Knock Mode

The occurrence of knock events in a period of 720 °CA can be controlled either via a probability function or using sequence tables. You can toggle between these two operating modes via this parameter.

Sequence Type

When selecting "Modulation Mode = sequence", you can specify here whether the common sequence table ("common") or the cylinder-individual sequence table ("individual") is to be used (see "Sequence Tables" on page 235).

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Base Angle	real32	Yes	Base output angle for knock packages 0.0 … 720.0 °CA
Knock Mode	uint8	Yes	Probability or sequence table for outputting knock packages 0: Probability 1: Sequence table
Sequence Type	uint8	Yes	Selection: Common or cylinder-individual sequence 0: Common sequence 1: Cylinder-individual sequence
* Data type which the RTIO driver uses internally for the parameter			

Tab. 10-12 ESX335-Knock Device: Configuration Parameters of the "Signals" Tab

10.8.4 Data (ESX335-Knock Device)

The existing signal group shows RTIO signals which are available for controlling the knock signal generator.

Note

Please ensure that the "Served Cylinders" parameter in the upstream ESX335-Powertrain Subsystem ("Served Cylinders" on page 237) has been set correctly as this value is used when processing the signals!

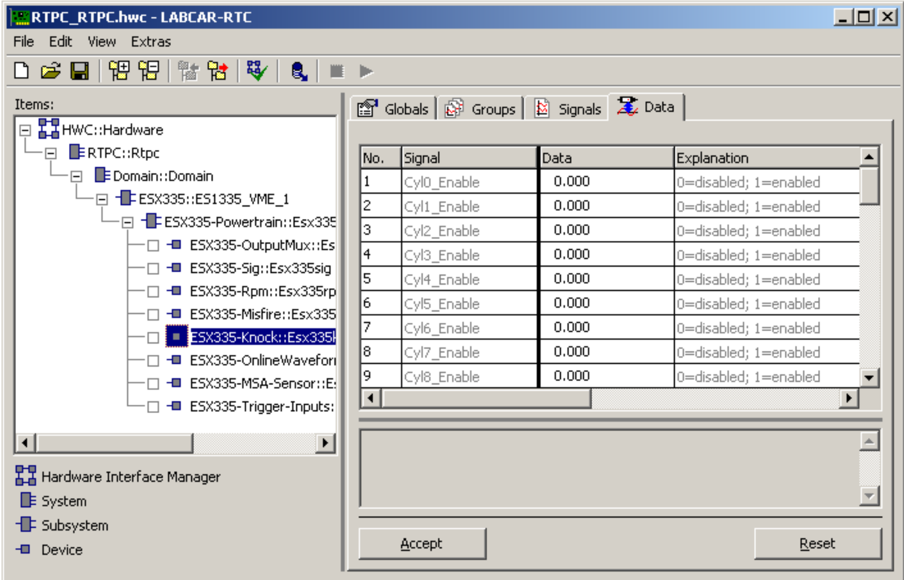


Fig. 10-15 ESX335-Knock Device: Configuration Parameters of the "Data" Tab

Cyl0_Enable … Cyl11_Enable

These twelve signals activate the generation of knock packages for uncontrolled combustion for the relevant cylinders.

Cyl0_Probability … Cyl11_Probability

These signals are used to control the probability of the occurrence of knocking combustion for a cylinder. The signal is only taken into consideration if the relevant cylinder has been configured for using the probability function.

Sqnc_Trigger

This signal represents the trigger with which sequence table play-back can be started. A rising edge of 0 → 1 on this signal starts play-back. A sequence which has not run completely is interrupted by a rising edge on the trigger signal and restarted (re-trigger).

NoiseAmpl

This signal controls a stochastic variation of the amplitudes of the knock packages. There is no variation of the amplitude with a value of 0.0; with a value of 0.1, the amplitude can vary by 10% of the maximum output voltage.

This variation has an effect on the output of all knock packages (knocking and non-knocking).

NoKnockAmpl

Knock packages can still be output in combustion without knocking. The amplitudes of these knock packages are described with this signal. Normally, the amplitude of "NoKnockAmpl" is defined as being smaller than the amplitudes in which knocking occurs.

Cyl0_KnockAmpl … Cyl11_KnockAmpl

The cylinder-specific amplitudes for knocking combustion can be output using these 12 signals.

Cyl0_AngleOffset … Cyl11_AngleOffset

The output angle of the knock packages can be varied with this signal for each individual cylinder by ±127 °CA in intervals of 1 °CA. The total of "Reference Phase" ("Globals" tab) and the relevant "Base Angle" ("Signals" tab) is used as base angle.

RTIO Signal	Data Type*	Comment/Value Range
Cyl0_Enable … Cyl11_Enable	uint8	Control of the knock signals for the individual cylinders 0: Knock packages for knocking possible 1: Knock packages for knocking not possible
Cyl0_Probability ... Cyl11_Probability	uint16	Probability of knocking Value range: 0 to 10000 (corresponds to 0.0 to 1.0)
Sqnc_Trigger	uint8	Starting or restarting sequence table playback An edge 0 → 1 is used as trigger
NoiseAmpl	uint16	Stochastic amplitude variation Value range 0 to 10000 (corresponds to 0.0 to 1.0)
NoKnockAmpl	uint16	Amplitude with non-knocking combustion Value range 0 to 10000 (corresponds to 0.0 to 1.0)
Cyl0_KnockAmpl … Cyl11_KnockAmpl	uint16	Amplitude with knocking combustion Value range 0 to 10000 (corresponds to 0.0 to 1.0)
Cyl0_AngleOffset … Cyl11_AngleOffset	sint8	Variation of the knock angle Value range -127 to +127 (resolution 1 °CA)
* Data type which the RTIO driver uses internally for the parameter		

Tab. 10-13 ESX335-Knock Device: The RTIO Signals of the "Data" Tab

10.9 Direct Waveform Access: ESX335-OnlineWaveformAccess Device

Waveforms can be modified in the running simulation using the ESX335-Online-WaveformAccess device.

There are no configuration parameters; only signals are used to modify the waveforms.

10.9.1 Data (ESX335-OnlineWaveformAccess Device)

In the "Ctrl" signal group, RTIO signals are used to access the waveforms.

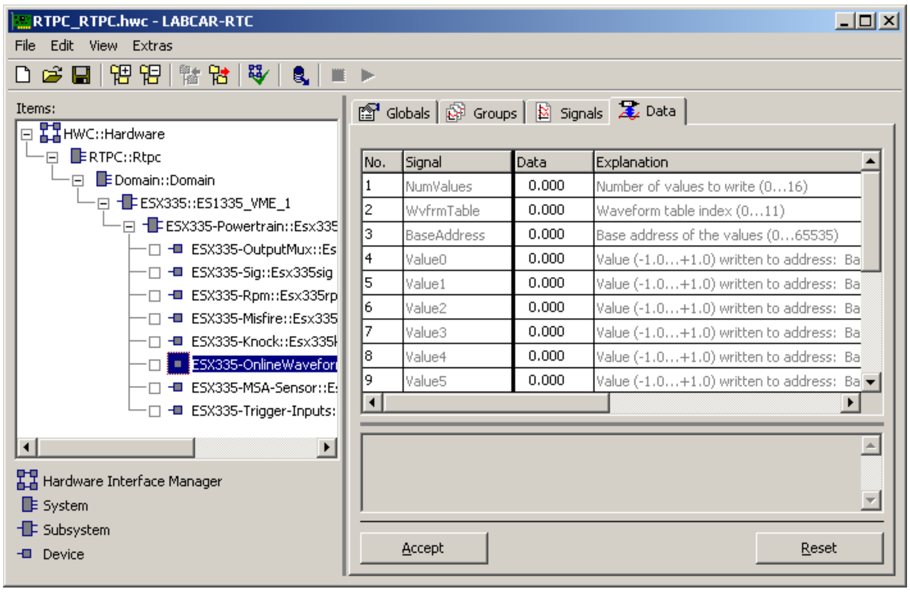


Fig. 10-16 ESX335-OnlineWaveformAccess Device: Signals of the "Data" Tab

NumValues

This signal specifies how many data values in a waveform are to be modified. A value of 0 means that no change is to be made to the waveform. A maximum of 16 values can be modified per simulation step.

WvfrmTable

The waveform to be modified is determined by this signal. Possible values are 0 to 11 for the twelve waveforms which are defined in the RTIO element "ESX335-Powertrain".

BaseAddress

This signal specifies the first data point in the selected waveform whose value is to be changed. The value range is 0 to 65535; only values within the global angular resolution ("Resolution" parameter in the RTIO element ESX335-Powertrain) can be output.

Value0 … Value15

These 16 values are written to the waveform "WvfrmTable" – starting with "BaseAddress" - to "NumValues" ascending data points.

The following table summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
NumValues	uint8	Number of values to be modified 0…16
WvfrmTable	uint8	The waveform to be modified 0 … 11
BaseAddress	uint16	First data point in the waveform to be modified 0 … 65535
Value0 … Value15	sint16	Data values to be written to the waveform. -32768 … 32767

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-14 ESX335-DirectWaveformAccess Device: The RTIO Signals of the "Data" Tab

10.10 ESX335-MSA-Sensor Device

The ESX335-MSA-Sensor device is used to configure and address one of the six signal generators on the ESX335.1 Arbitrary Signal Generator Board to simulate an MSA crankshaft sensor which detects the direction of rotation.

As the output pulses have a fixed duration and therefore cannot be generated by reading out and outputting a signal table, a special device was introduced to represent an MSA sensor. In comparison to an ESX335-Sig device, the "Globals" tab has additional parameters used to define the sensor.

10.10.1 Basic Functioning

The MSA sensor simulation can use the same waveform traces of the tooth signal as the (Hall) sensors used so far. However, to represent an MSA sensor, the angle position of the center of each tooth of the crankshaft gear must be known. A fixed-length pulse is output for each tooth at the tooth center – in the real sensor, the tooth centers are determined electronically.

In the configuration phase, these angle positions are calculated in accordance with the trace of the waveform and stored in the waveform pool. As the waveform can be selected in real time from one of the twelve predefined waveforms using the "Waveform" signal, all waveforms have to undergo this procedure.

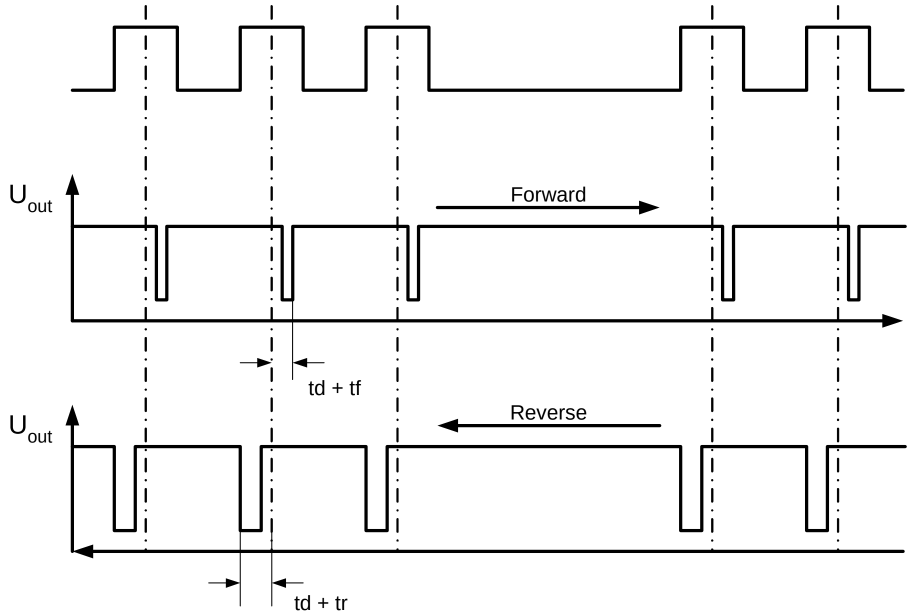


Fig. 10-17 Crankshaft Teeth and Sensor Signals with an MSA Sensor

10.10.2 Tooth Center Calculation

First of all the positions of the negative and positive zero crossings are determined for all waveform traces which have already been normed to a value between -1.0 and 1.0.

In positive logic, a tooth is an area between one positive and the following negative zero crossing (see Fig. 10-18). The tooth center is accordingly the arithmetic center of the two positions (and accordingly vice versa with negative logic):

Let np1, np2 be the (positive) zero crossings from bottom to top and nn1, nn2 the (negative) zero crossings from top to bottom. This results in the tooth center positions m1, m2 being m1 = (np1 + nn1) / 2 and m2 = (np2 + nn2) / 2.

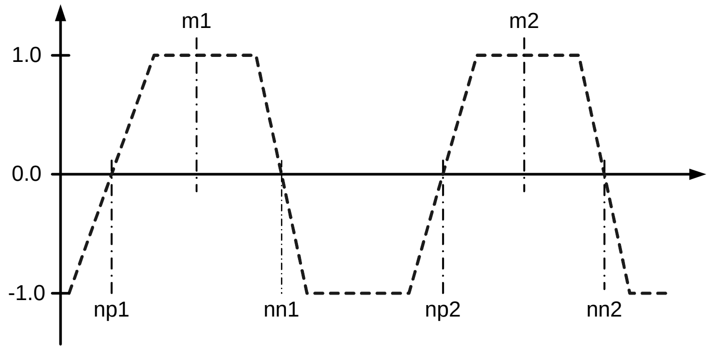


Fig. 10-18 Center Position Calculation (with a not quite symmetrical tooth trace and positive logic)

The logic is set for all curve forms together in the "Globals" tab of the ESX335-Powertrain device (see "MSA Sensors: Logic Level of Teeth" on page 238).

10.10.3 Globals (ESX335-MSA-Sensor Device)

The ESX335-MSA-Sensor device is used to configure and address one of the signal generators on the ESX335.1 Arbitrary Signal Generator Board. The following figure shows the RTIO parameters of the "Globals" tab.

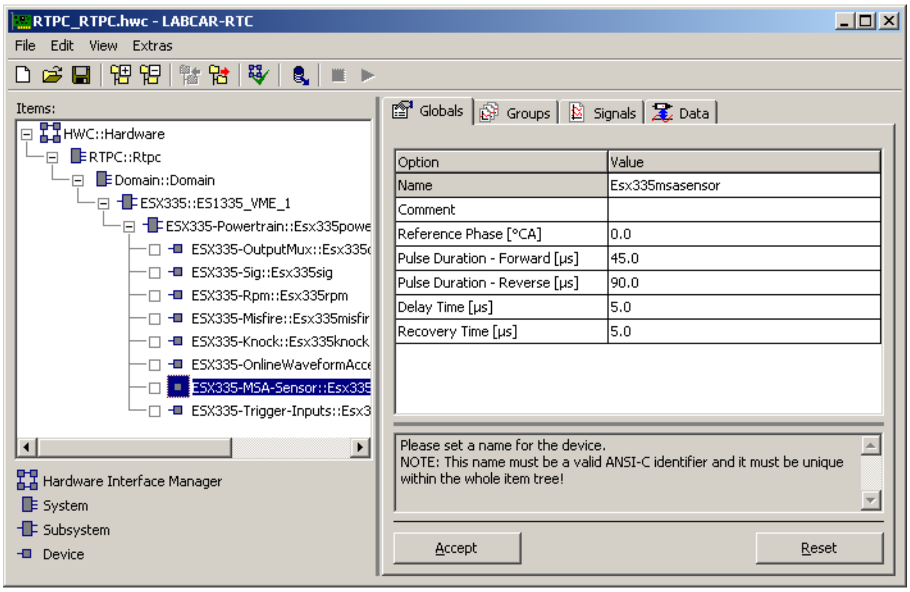


Fig. 10-19 The "Globals" Tab of the ESX335-MSA-Sensor Device

The meaning of the options is described below.

Comment

Any comment for this signal generator.

Note

The comment field does not permit blanks or special characters.

Reference Phase [° CA]

This parameter describes the reference phase shift of the signal generator. The actual phase is determined by adding this value to the signal "PhaseRef".

Pulse Duration Forward [µs]

This parameter defines the pulse duration of a tooth with a positive direction of rotation (forwards) (tf in Fig. 10-17 on page 263).

Pulse Duration Reverse [µs]

This parameter defines the pulse duration of a tooth with a negative direction of rotation (backwards) (tr in Fig. 10-17 on page 263).

Delay Time [µs]

This parameter defines the length of the delay time (td in Fig. 10-17 on page 263).

Recovery Time [µs]

This parameter defines the length of a forced pulse (in certain situations when the direction of rotation is reversed).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Comment	Identi-fier	No	User-specific comment
Reference Phase	real32	Yes	Ref. phase shift in ° CA, Value range 0…720 °CA
Pulse Duration - Forward	real32	Yes	Pulse duration tf with a forwards rotation, value range 10.0…1000.0 µs
Pulse Duration - Reverse	real32	Yes	Pulse duration tr with a reverse rotation, value range 10.0…1000.0 µs
Delay Time	real32	Yes	Delay time td in acc. with sensor specification Value range 1.0…100.0 µs
Recovery Time	real32	Yes	Pulse duration tv in acc. with sensor specification, value range 1.0...100.0 µs

* Data type which the RTIO driver uses internally for the parameter

Tab. 10-15 ESX335-MSA-Sensor: Configuration Parameters of the "Globals" Tab

10.10.4 Groups (ESX335-MSA-Sensor Device)

The ESX335-MSA-Sensor device has one signal group which is used to control the MSA generator.

10.10.5 Data (ESX335-MSA-Sensor Device)

The "Data" tab lists the signals for controlling the signal generator.

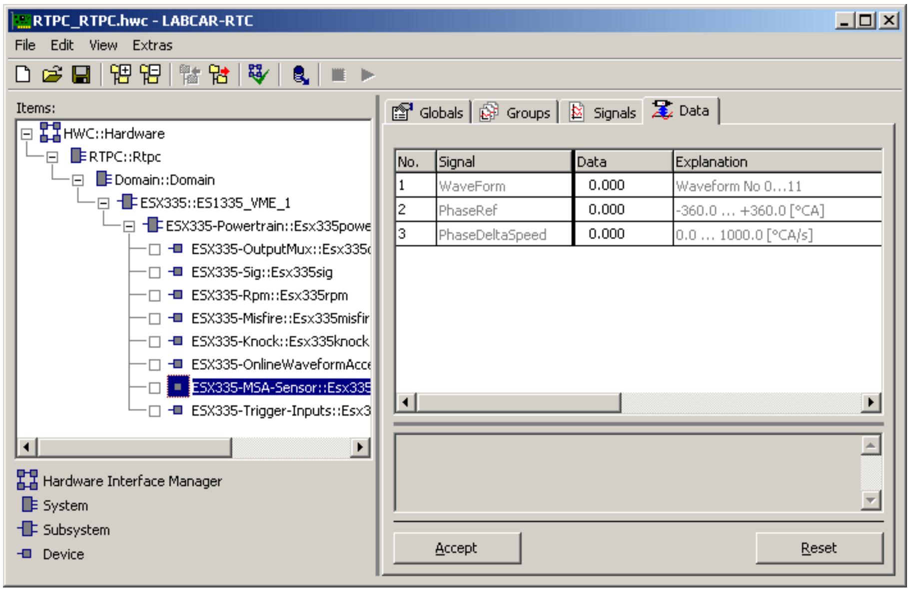


Fig. 10-20 The "Data" Tab of the ESX335-MSA-Sensor Device

The meaning of the signals is described below.

WaveForm

The waveform to be read out and output by the signal generator is selected via the signal "WaveForm". Values from 0 to 11 are possible, corresponding to the central waveform pools in the ESX335-Powertrain device.

PhaseRef

This signal describes a phase shift which is added to the global parameter "ReferencePhase". This enables the generation of a model-controlled phase shift during runtime.

Please note that a phase shift is changed with a finite speed (controlled via the "PhaseDeltaSpeed" parameter) which is why the "PhaseDeltaSpeed" signal must have a value > 0 for most applications.

PhaseDeltaSpeed

The change speed of the phase shift is controlled by the signal "PhaseDeltaSpeed" as a change of the value of "PhaseRef" can only take place with finite speed. The value range is from 0 (i.e. no adjustment of the phase) to 1000° CA/s.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
WaveForm	uint8	Yes	Selection of waveform 0-11
PhaseRef	sint16	Yes	Additive phase shift, value range: 32768…32767 (corresponds to -360…+360 [° CA])
PhaseDeltaSpeed	uint16	Yes	Speed of phase change 0…1000 [° CA/s]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 10-16 ESX335-MSA-Sensor: Signals of the "Data" Tab

10.10.6 Troubleshooting

An error message about unsuitable waveform traces is only output when an unsuitable waveform is actually used as the basis of the output of an MSA sensor.

The error message will have the following format:

```
"Waveform <x> not suitable for MSA sensor implemented
by SigGen #<y> because <error> No signal generated"
```

`<x>` represents the number of the unsuitable waveform 0-11 and `#<y>` the number of the signal generator 0…5 used.

The following table contains a description of the error messages `<error>`:

Error Message <error>	Description
…too many zero crossings (> 2048)…	Too many zero crossings were detected to generate a sensible output for an MSA sensor.
…number of positive and negative zero crossings do not match…	A varying number of positive and negative zero crossings was determined. Curve shape is not suitable for representing an MSA sensor.
…too few zero crossings (< 2)…	Too few zero crossings were detected to represent a sensible output for an MSA sensor.
…minimum distance between some tooth centers < 0.4 °CA…	At least two of the calculated tooth center positions are too close together to be processed correctly by the output unit. A minimum distance of around 0.4 °CA is necessary.

Tab. 10-17 List of Possible Error Messages with Unsuitable Waveforms

10.11 External Triggering of Signal Generators: ESX335-Trigger-Inputs Device

If a PB1335TRIG Trigger Module for triggering signal generators with external signals is used, the ESX335-Trigger-Inputs device can be used to obtain status information.

There are no further configuration parameters.

10.11.1 Data (ESX335-Trigger-Inputs Device)

The "Data" tab lists the signals for the status of the PB1335TRIG Module.

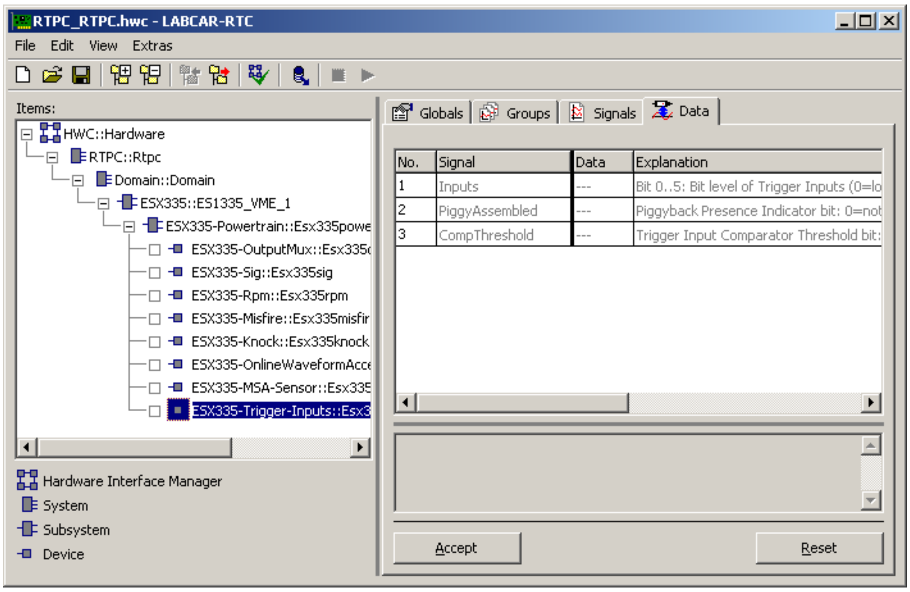


Fig. 10-21 ESX335-Trigger-Inputs Device: Signals of the "Data" Tab

The meaning of the signals is described below.

Inputs

This signal determines the levels of the trigger inputs 0 - 5. A whole integer is displayed and is evaluated in binary representation:

- Bit *n* = 0 means input *n* = low
- Bit *n* = 1 means input *n* = high

Example: The number shown is 15, binary 001111, which means inputs 0...3 are "high" and inputs 4 and 5 are "low".

PiggyAssembled

This bit determines whether a PB1335TRIG Module is assembled on the ESX335.1.

11 ES1336.1 Angle Synchronous Measurement Board

Terms

Active signal edge

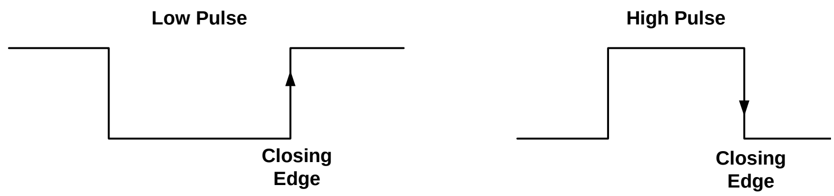
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing pulse edge

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.



Inactive signal

A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

LWL

Stands for Lower Window Limit and refers to the lower limit of the angle window in angle-synchronous measurements.

Opening pulse edge

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.

UWL

Stands for Upper Window Limit and refers to the upper limit of the angle window in angle-synchronous measurements.

11.1 The Basic Principle of Measure Value Calculation

11.1.1 Signal Acquisition

The ES1336.1 Angle Synchronous Measurement Board has 20 analog input channels with a voltage range of 0 V to 40 V and two inputs for battery voltages with a voltage range of 0 V to 60 V. The analog input voltages are routed to 12-bit A/D converters via a protection and conditioning circuit.

The converted signals are supplied to a FPGA and are the starting point for further signal processing. The sampling rate of the A/D converters is 12.5 MHz on every input channel.

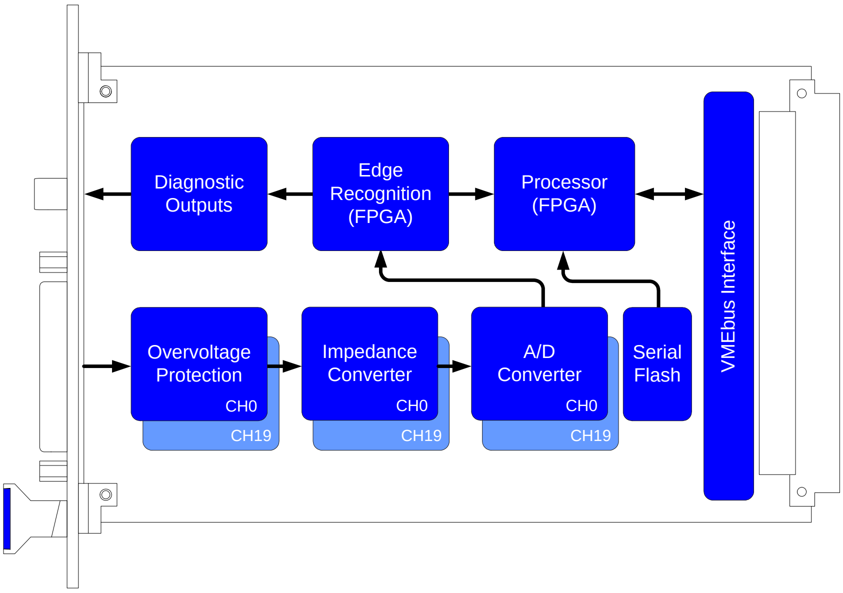


Fig. 11-1 Block Diagram of the ES1336.1 Angle Synchronous Measurement Board

The processing of the two battery voltages is similar; they are routed to a 10-bit A/D converter via a conditioning circuit. The sampling rate of this A/D converter is 78.125 kHz per battery voltage. The FPGA is also supplied with the converted battery voltages.

For angle-synchronous measurements, the ES1336.1 has an angle clock interface in the FPGA which works both in Master and Slave mode. In Master mode, the FPGA generates the angle clock and makes it available to other boards in the VMEbus system on the backplane. In Slave mode, the board is synchronized to an angle clock available on the VMEbus backplane.

11.1.2 Comparator Levels

For further data acquisition, it is necessary to extract binary level information from the measured input voltages. Every input voltage is routed to a comparator level with hysteresis behavior for this purpose (Fig. 11-2).

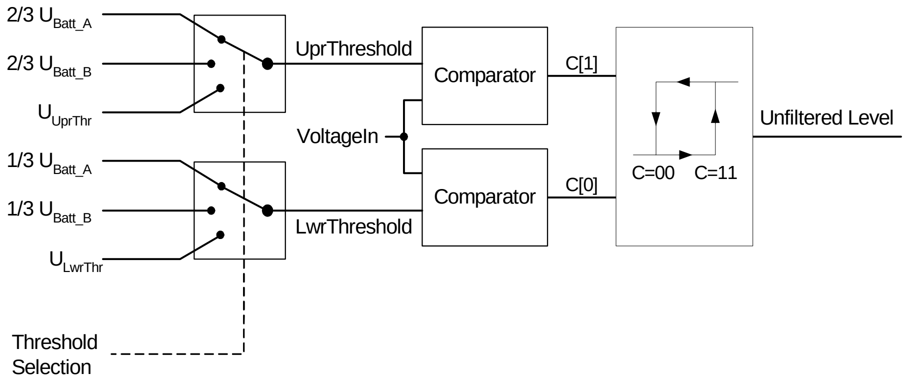


Fig. 11-2 Comparator Level

The user can configure the threshold voltages of the comparators in three different ways:

- Lower threshold = 1/3 battery voltage A
  Upper threshold = 2/3 battery voltage A
- Lower threshold = 1/3 battery voltage B
  Upper threshold = 2/3 battery voltage B
- Specification of the threshold values by the user

The ES1336.1 also makes it possible to measurement signal rise and fall times. These times are usually measured between 10% and 90% of the signal amplitude.

As this setting of the threshold values is not always desired for level extraction, the ES1336.1 has separate comparators for measuring rise and fall times. The comparator thresholds are defined by the user in volts (Fig. 11-3).

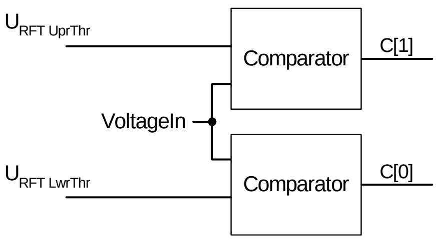


Fig. 11-3 Comparator Level for Measuring Signal Rise and Fall Times

11.1.3 Pulse-Width and Angle Measurements

The level information of every channel is constantly being checked for changes, i.e. signal edges are detected. When an edge occurs, the current time and current crankshaft angle are stored in an FPGA-internal memory. The processor firmware of the ES1336.1 can now read the acquired values from the memory and thus generate information on when and at which angle which input signals have changed.

All measure values which are of interest to the user can then be generated from this information, values such as injection times, ignition times and duty cycles.

This basic concept of hardware-technical edge detection with subsequent software editing of the measure values has the advantage that the firmware and RTIO driver simply have to be adjusted to implement further measurement procedures (in addition to the ones already implemented).

11.1.4 Pulse Integration

Fig. 11-4 describes how pulse integration works – it shows the analog input signal together with the two comparator thresholds and the level signal generated by the comparators. If the upper comparator threshold is exceeded, the A/D values acquired with a frequency of 12.5 MHz are accumulated until such time as the analog signal exceeds the lower comparator threshold again. The integral is calculated from the result of the sum obtained and the A/D converter sample interval of 80 ns.

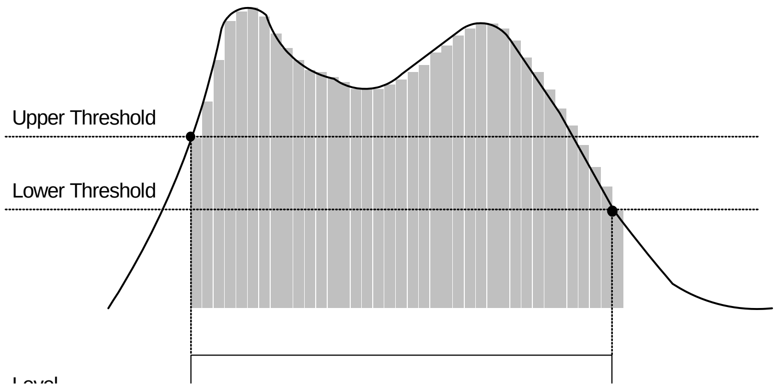


Fig. 11-4 Pulse Integration

11.1.5 Measuring Rise and Fall Times

As has already been mentioned, the ES1336.1 has separate comparators for measuring signal rise and fall times. The times are measured as shown in Fig. 11-5. The resolution of the measured times is determined by the sampling frequency of the A/D converters of 12.5 MHz – the resolution is thus 80 ns.

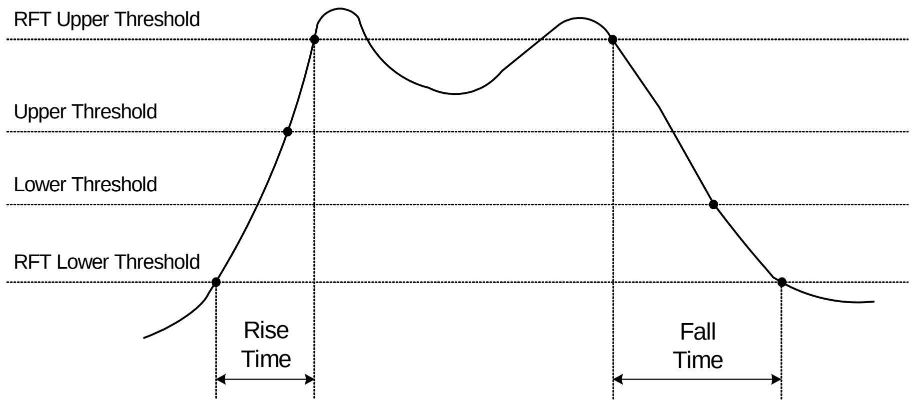


Fig. 11-5 Measuring Rise and Fall Times

11.1.6 Peak Value Measurement

Peak value acquisition is active as soon as the analog signal exceeds the upper comparator threshold. If the signal exceeds the lower comparator threshold, peak value acquisition is deactivated. Peak value acquisition is also based on analog signals sampled with 12.5 MHz.

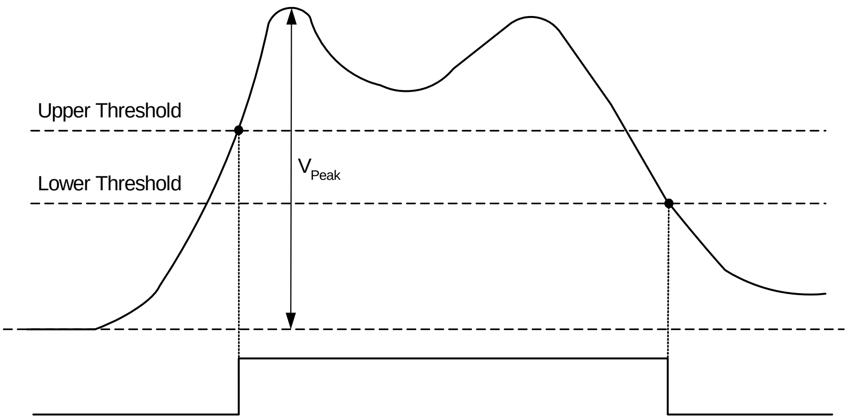


Fig. 11-6 Peak Value Acquisition

11.1.7 Asynchronous Measurements

With asynchronous measurements, the relevant measure value (e.g. frequency, duty cycle or hightime) is calculated on the basis of the most recent edge entries available in the memory.

Time-synchronous timeout monitoring can be activated in these measurements. If no edge is detected on the relevant hardware channel within a period defined by the user, a timeout is detected.

11.1.8 Angle-Synchronous Measurements

Angle windows which are specified by a lower angle window limit (LWL) in CA° and an upper angle window limit (UWL) in °CA are characteristic for angle-synchronous measurements.

The user can define up to three angle windows per hardware channel which can overlap but whose size must not exceed 720 °CA. The user defines measurements of the following kind:

- hightime of the n-th pulse in the angle window
- angle of the n-th falling edge in the angle window
- integral value of the n-th (high) pulse in the angle window
- additive lowtime in the angle window

The calculation of the measure values takes place whenever the crankshaft angle has exceeded the upper angle window limit. Timeout monitoring can also be activated for angle-synchronous measurements. If, at the upper window limit, it is discovered that the measure value cannot be calculated, e.g. if the hightime of the fourth pulse is to be measured in the angle window but only three pulses occurred, a timeout is detected.

11.1.9 Timeouts

A timeout is indicated by the setting of a timeout flag (with activated timeout monitoring). At the same time, the valid measure value is defined by timeout monitoring.

There are two possibilities: Either a measure value specified by the user is transferred to the simulation model or a value defined by the relevant measure method. The user decides which of the two values is transferred whereby the second procedure (a value defined by the measure method) is only available with a few measurement procedures.

11.2 ES1336 Subsystem

The ES1336.1 RTIO driver provides the user with up to 288 freely configurable measure values (also referred to below as software or measure channels) where each individual measure value can depend on any hardware channel. The assignment of hardware channels on the one hand and measure value on the other is thus not defined by the ES1336.1 RTIO driver, i.e. when using a labcar for several projects it is not absolutely necessary to always connect signals of the same kind to the same hardware channels of the ES1336.1 Angle Synchronous Measurement Board.

In addition, the 288 measure channels provide the user with around 14 times as many output values as the board has hardware channels. This means it is simple to determine several measure values from a single input signal, e.g. frequency and duty cycle of a PWM signal. Double wiring of one and the same signal on different channels of the board is not necessary.

Structure of the ES1336.1-RTIO Tree

The ES1336.1 Angle Synchronous Measurement Board is integrated in the RTIO Editor by selecting the ES1336 subsystem.

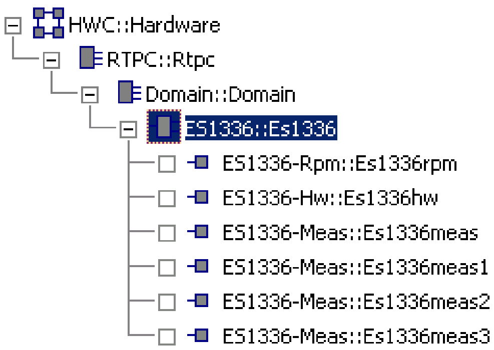


Fig. 11-7 RTIO Hardware Description with an Integrated ES1336.1

The 20 hardware channels and 8 diagnostic outputs are configured with the ES1336-Hw device which is assigned to the ES1336 subsystem.

The angle clock interface is configured with the ES1336-Rpm device which is also assigned to the ES1336 subsystem.

Up to nine ES1336-Meas devices can be assigned to the ES1336 subsystem for specifying measurements. In turn, up to 32 measurements can be defined with each of these devices – this means that the RTIO driver and firmware can enable a total of 288 measurements.

Note

Before an ES1336-Meas device can be integrated, an ES1336-Hw device first has to be integrated for configuring the ES1336.1 hardware.

11.2.1 Globals (ES1336 Subsystem)

The following figure shows the RTIO parameters of the "Globals" tab.

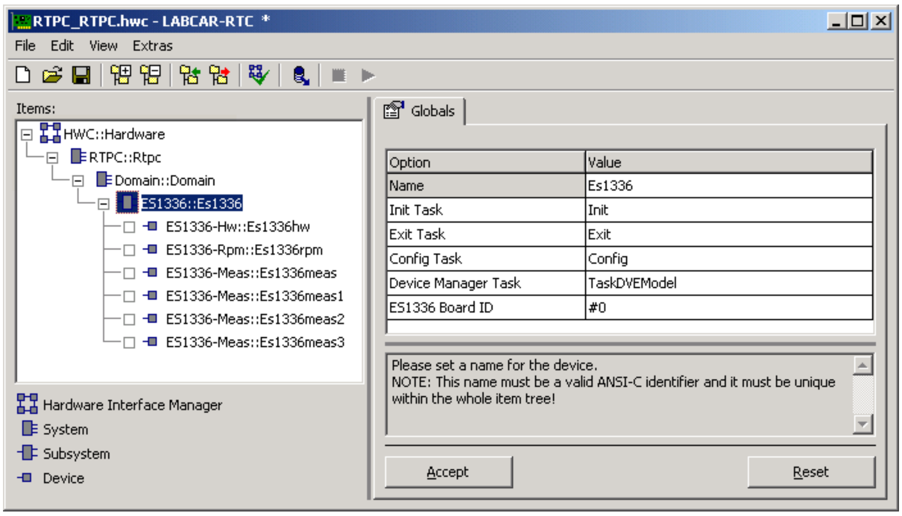


Fig. 11-8 The "Globals" Tab of the ES1336 Subsystem

Device Manager Task

This task controls the reading and transport of measure data from the board to the real-time PC via the ES1130. The presetting should be retained when setting the task.

ES1336 Board ID

This option field is used to identify the ES1336.1 boards – it establishes the assignment between the RTIO hardware description and ES1336.1 in the VME-bus chassis for which this description is valid.

In the RTIO Editor, up to 20 ES1336.1 boards can be integrated per chassis. The ES1336.1 boards in the chassis are numbered from left to right (left = VMEbus slot no. 1, with ascending slot numbering) starting with 1. The number of the ES1336.1 board determined must be specified in the "ES1336 Board ID" field.

This RTIO parameter cannot be edited during runtime of the model on the experimental target.

11.3 ES1336-Rpm Device

11.3.1 Globals (ES1336-Rpm Device)

The ES1336-Rpm device is used to configure and address the clock angle interface.

To execute measurements which are synchronous to the crankshaft angle, the ES1336.1 requires information about the angle position of the crankshaft. The ES1336.1 determines this information from the angle clock signal, which it either creates itself or taps from the VMEbus backplane.

Fig. 11-9 shows the RTIO parameters of the "Globals" tab.

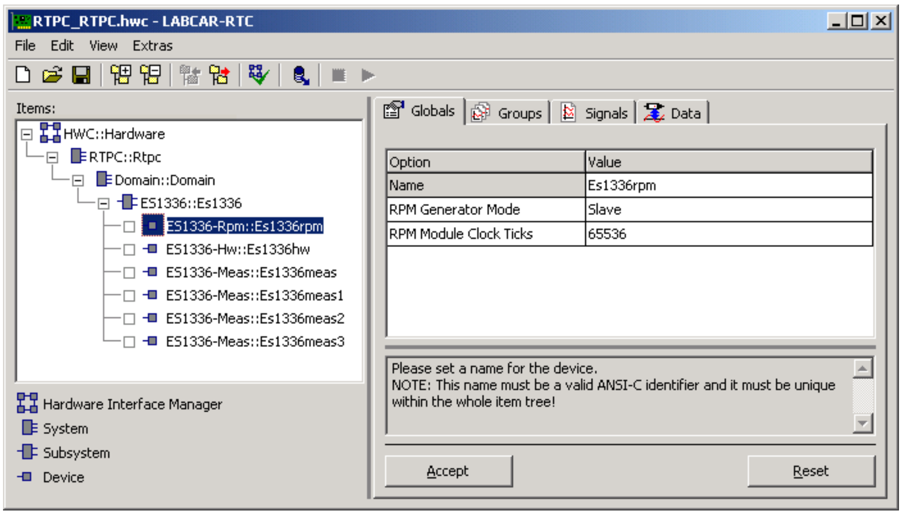


Fig. 11-9 The "Globals" Tab of the ES1336-Rpm Device

The meaning of the individual parameters is described below – Tab. 11-1 on page 277 summarizes their properties.

RPM Generator Mode

This option is used to define the mode in which the angle clock interface works.

- Slave

  In Slave mode, the ES1336.1 is synchronized to an angle clock supplied to the board via the VMEbus backplane.

- Master

  In Master mode, the board generates the angle clock itself and makes it available at the VMEbus backplane so that other boards in the VMEbus system can be synchronized to this clock.

- Local Master

  In Local Master mode, the board also generates the angle clock itself, but the clock is not made available at the VMEbus backplane. The option field can be edited online.

RPM Module Clock Ticks

The "RPM Module Clock Ticks" field is used to define the angular resolution, i.e. the number of angle clocks per camshaft revolution (720° crankshaft). The minimum number of angle clocks is $2^4 = 16$ which represents an angular resolution of 45 °CA (°CA = degrees crankshaft). The maximum number of angle clocks of $2^{16} = 65536$ represents a resolution of 0.011 °CA.

The option field can be edited during runtime of the model on the experimental target..

Note

Not only the ES1336.1 but also the ES1321.1 PWM I/O Board and the ES1335.1 Arbitrary Signal Generator Board have an angle clock interface which can be used as a master. However, only one board in a VMEbus system can be angle clock master (i.e, supply the angle clock at the VMEbus backplane).
All other ES1336.1 and/or ES1335.1 and ES1321.1 boards must then work in Slave mode or be configured as the Local Master. In addition, the angular resolution of all slave boards must correspond to the angular resolution of the relevant master.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
RPM Generator Mode	uint8	Yes	Clock angle interface mode 0: Slave 1: Local Master 2: (System) Master
RPM Module Clock Ticks	uint32	Yes	Angular resolution: 2^4 to 2^{16} in steps of two (2^6, 2^8, ...)

* Data type which the RTIO driver uses internally for the parameter

Tab. 11-1 ES1336-Rpm-Device: Configuration Parameters of the "Globals" Tab

11.3.2 Groups (ES1336-Rpm Device)

The ES1336-Rpm device has two signal groups.

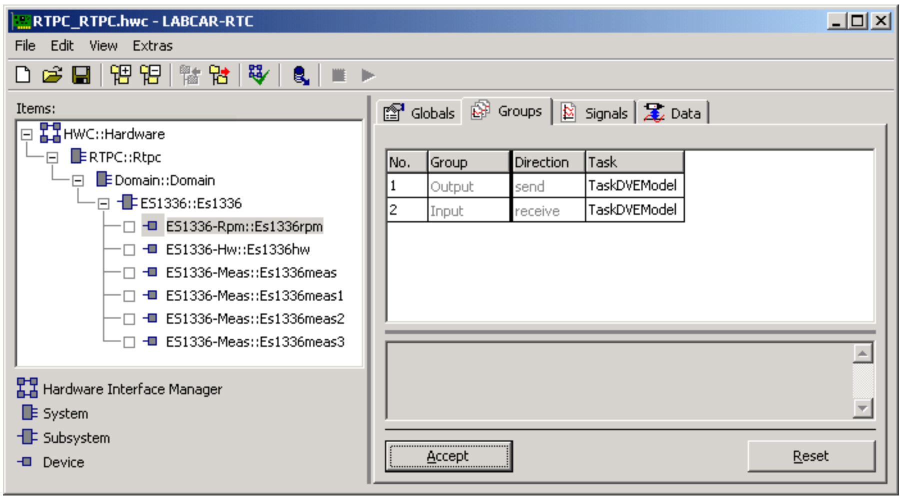


Fig. 11-10 The "Groups" Tab of the ES1336-Rpm Device

The "Output" signal group is transferred from the experimental target to the ES1336.1. This signal group transports the setting for engine speed. The engine speed or signal group is only of importance if Master or Local Master has been set as "RPM Generator Mode".

The "Input" signal group is transferred in the opposite direction, from the ES1336.1 to the experimental target. This signal group transports the current crankshaft angle.

Operating system tasks must be assigned to the signal groups – typically a task with periodic activation is set for both signal groups. As the engine speed is calculated by the engine model and the crankshaft angle is processed by the model, the calculation rate of the model tasks will be selected as activation period.

The RTIO Signals of the "Output" Signal Group

The "Output" signal group contains one RTIO signal, "EngSpeed", the setting for the engine speed.

RTIO Signal	Data Type*	Comment/Value Range
EngSpeed	real32	Engine speed in rpm Minimum engine speed: 0 rpm Maximum engine speed: see Tab. 11-3. Resolution: 0.001 rpm The signal is only of importance if the angle clock interface is working in Master mode.
* Data type which the RTIO driver uses internally for the parameter		

Tab. 11-2 ES1336-Rpm-Device: the RTIO Signals of the "Output" Signal Group

$r_{ANGULAR}$	16	32	64	128	256	512	1024
$n_{Engine,Max}$ [rpm]	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$2.1 \cdot 10^6$	$1.9 \cdot 10^6$
$r_{ANGULAR}$	2048	4096	8192	16384	32768	65536	
$n_{Engine,Max}$ [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 Maximum Engine Speed ($n_{Engine,Max}$) depending on the Angular Resolution ($r_{Angular}$)

The RTIO Signals of the "Input" Signal Group

The "Input" signal group contains the RTIO signal "ActAngle", the current crankshaft angle in °CA.

RTIO Signal	Data Type*	Comment/Value Range
ActAngle	real32	Current crankshaft angle in °CA. Value range: [0.0 °CA, 720.0 °CA[
* Data type which the RTIO driver uses internally for the parameter		

Tab. 11-4 ES1336-Rpm-Device: the RTIO Signals of the "Input" Signal Group

11.3.3 Signals (ES1336-Rpm Device)

No configuration settings are to be made in the "Signals" tab.

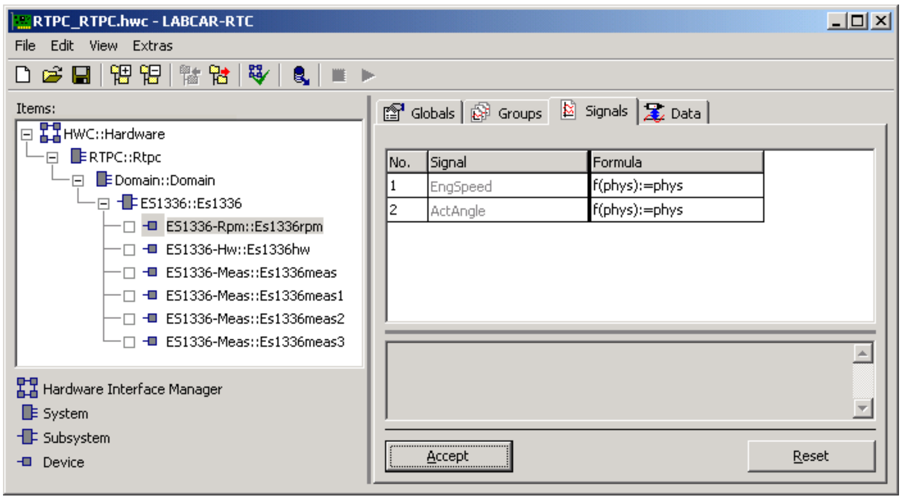


Fig. 11-11 The "Signals" Tab of the ES1336-Rpm Device

11.4 ES1336-Hw Device

11.4.1 Globals (ES1336-Hw Device)

The ES1336-Hw device is used to configure and address the ES1336.1 settings independent of the individual hardware channels are made in the "Globals" tab of this device. Channel-specific settings are made in the "Signals" tab. The appearance of the "Globals" tab changes according to whether an ES1336-Rpm device is integrated or not.

Fig. 11-12 shows the RTIO parameters of the "Globals" tab with an integrated ES1336-Rpm device. Tab. 11-5 summarizes the properties of these parameters.

Fig. 11-13 shows the RTIO parameters of the "Globals" tab with a missing ES1336-Rpm device. In this case, the angle clock interface of the ES1336.1 can only be operated in Slave mode. The "RPM Generator Mode" option field is disabled. The resolution of the angle clock is set in the "RPM Module Clock Ticks" option field. For more details on these two option fields, refer to the section "Globals (ES1336-Rpm Device)" on page 276.

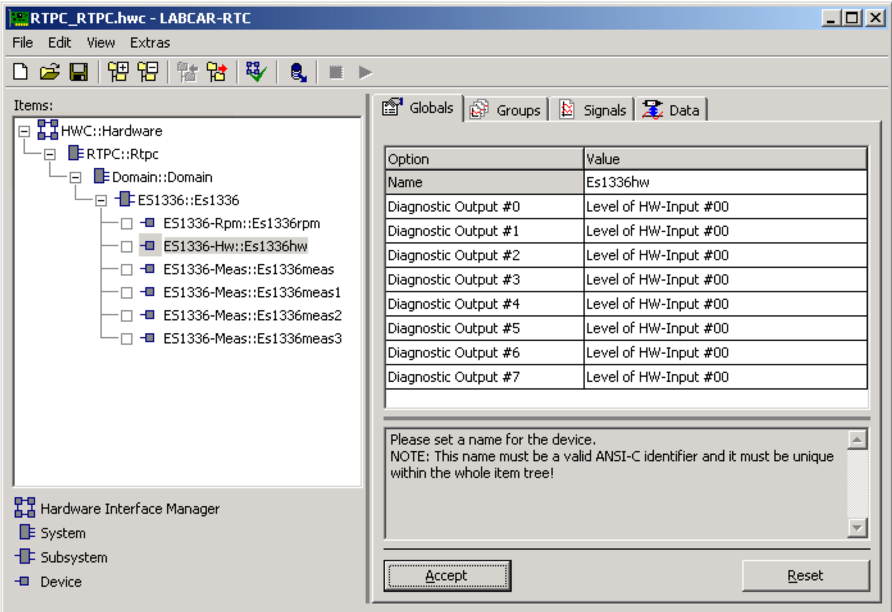


Fig. 11-12 The "Globals" Tab of the ES1336-Hw Device with an Integrated ES1336-Rpm Device

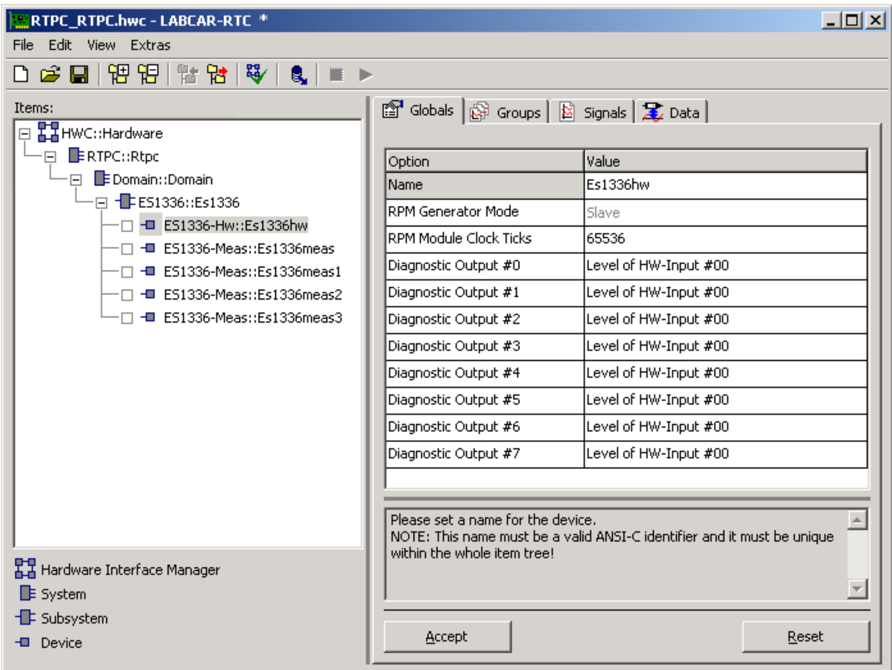


Fig. 11-13 The "Globals" Tab of the ES1336-Hw Device with a Missing ES1336-Rpm Device

Diagnostic Output #x (x = 0, 1, ... 7)

The ES1336.1 makes it possible to apply FPGA-internal signals to the eight diagnostic outputs. The "Diagnostic Output #x (x = 0, 1, ... 7)" list field defines which signal is issued at the relevant diagnostic output.

The following signals are available:

- Level of HW-Input #xx (xx = 00, 01, ... 19)

  These signals are the level signals after the threshold comparison of the individual hardware channels ("Unfiltered Level" in Fig. 11-2 on page 271).
- CA Clock: Cycle Indicator

  This is an angle-synchronous signal which is suitable for use as a trigger signal. For details of the course of the signal, refer to Fig. 11-14.

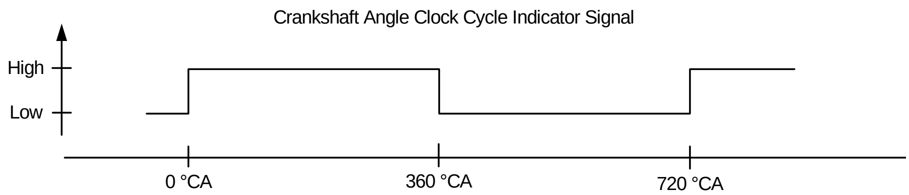


Fig. 11-14 Angle Clock Cycle Indicator Signal

The option field can be edited during runtime of the model on the experimental target..

Parameter or Option Field	Data Type*	Can Be Edited Online	Comment/Value Range
Diagnostic Output #x (x = 0, 1, 2 ... 7)	uint8	Yes	Signal to diagnostic output x 0: Level of hardware signal 0 1: Level of hardware signal 1 ... 19: Level of hardware signal 19 31: Angle clock cycle indicator signal
* Data type which the RTIO driver uses internally for the parameter			

Tab. 11-5 ES1336-Hw Device: Configuration Parameters of the "Globals" Tab

11.4.2 Groups (ES1336-Hw Device)

The ES1336-Hw device has two signal groups.

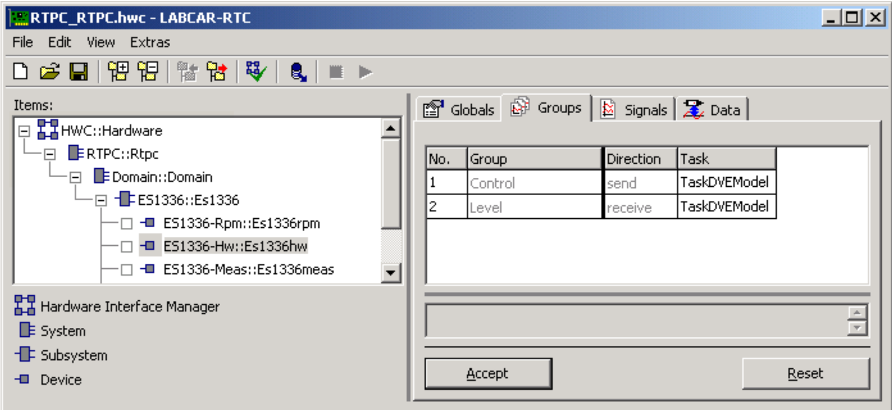


Fig. 11-15 The "Groups" Tab of the ES1336-Hw Device

The "Control" signal group is transferred from the experimental target to the ES1336.1. This signal group is used to enable/disable measure value calculation on the individual hardware channels of the board.

The "Level" signal group is transferred in the opposite direction, from the ES1336.1 to the experimental target. This signal group contains the current level information of all hardware channels.

Operating system tasks are to be assigned to the signal groups. Usually a task with periodic activation and a relatively large activation period (e.g. 100 ms) is selected for the "Control" signal group as the enabling and disabling of the hardware channels is not usually a highly dynamic procedure. If the hardware channels are only to be enabled and disabled when the model is started or stopped, it is sufficient to assign the "Control" signal group to the Init task and the Exit task of the model.

If level information is also evaluated in the simulation model, a task with periodic activation is assigned to the "Level" signal group. The activation period depends on the dynamic behavior or the period duration of the signals to be acquired.

RTIO Signals of the "Control" Signal Group

The "Control" signal group consists of 20 RTIO signals with which measure value calculation on the 20 hardware channels can be enabled or disabled.

Tab. 11-6 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
EnableChn_00 ... EnableChn_19	bool	Enabling/disabling measure value calculation on hardware channels 0 to 19 0: Disable 1: Enable
* Data type which the RTIO driver uses internally for the parameter		

Tab. 11-6 ES1336-Hw Device: the RTIO Signals of the "Control" Signal Group

RTIO Signals of the "Level" Signal Group

The "Level" signal group consists of one RTIO signal, "LvlBitField_0", which is to be interpreted as a bit field. The bit field is a 20-bit field; the level information of hardware channels 0 to 19 is coded in it.

Tab. 11-7 summarizes the properties of the RTIO signal.

RTIO Signal	Data Type*	Comment/Value Range
LvlBitField_0	uint32	Bit field with level information of hardware channels 0 to 19: Channel 0: bit 0, LSB (Least Significant Bit) Channel 19: bit 19 Bit value 0: low level Bit value 1: high level
* Data type which the RTIO driver uses internally for the parameter		

Tab. 11-7 ES1336-Hw Device: the RTIO Signals of the "Level" Signal Group

11.4.3 Signals (ES1336-Hw Device)

The 20 hardware channels of an ES1336.1 are configured in the "Signals" tab. Fig. 11-16 shows the RTIO parameters of the "Signals" tab.

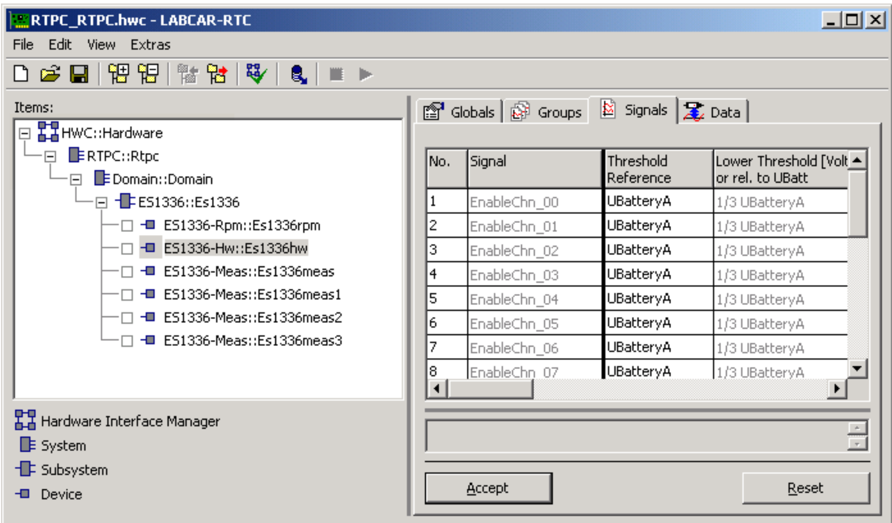


Fig. 11-16 The "Signals" Tab of the ES1336-Hw Device

The properties of the individual parameters are described below – for a summary, refer to Tab. 11-8 on page 286. All parameters can be edited during runtime of the model on the experimental target.

Threshold Reference

The "Threshold Reference" option field is used to define the reference voltages of the comparators for level determination (Fig. 11-2 on page 271). The threshold voltages of the comparators can be configured in three different ways:

- UBatteryA

  In this case, the lower threshold is 1/3 of battery voltage A and the upper threshold is 2/3 of battery voltage A.

- UBatteryB

  In this case the lower threshold is 1/3 of battery voltage B and the upper threshold is 2/3 of battery voltage B.
- Absolute Levels

  The comparator thresholds are defined explicitly in volts. The lower comparator threshold is defined in the numeric input field "Lower Threshold [Volt] or rel. to UBatt", the upper comparator threshold is defined in the numeric input field "Upper Threshold [Volt] or rel. to UBatt".

Lower Threshold [Volt] or rel. to UBatt

Upper Threshold [Volt] or rel. to UBatt

These two numeric input fields can only be edited if "Absolute Levels" has been selected in the "Threshold Reference" field. The input comparators of a hardware channel exhibit hysteresis behavior. The input fields define the upper and lower threshold of the comparator hysteresis in volts.

Lower Rise/Fall Threshold [Volt]

Upper Rise/Fall Threshold [Volt]

These two numeric input fields define the lower and upper threshold of the comparators for measuring rise and fall times (Fig. 11-5 on page 272). The threshold values must be specified in volts.

Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)

Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)

To execute angle-synchronous measurements, three angle windows can be defined on every hardware channel. "Angle Window #x Lower Limit [°CA]" defines each lower angle window limit in °CA, "Angle Window #x Upper Limit [°CA]" defines each upper angle window limit in °CA.

The limits can be set in the range [-720.0 °CA, 720 °CA[with the limitations that an angle window must not be greater than 720 °CA and the lower angle window limit must be smaller than the upper angle window limit.

Asynchronous Timeout [ms]

With asynchronous measurements, time-synchronous timeout monitoring can be activated. If no edge is detected on the relevant hardware channel within the period defined in the numeric input field "Asynchronous Timeout [ms]", a timeout is detected. The timeout time must be specified in milliseconds.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Threshold Reference	uint8	Yes	Comparator reference 0: Absolute Levels 1: UBatteryA 2: UBatteryB
Lower Threshold [Volt] or rel. to UBatt	real64	Yes	Lower comparator threshold in volts. Value range: [0.0 V, 40.0 V] Can only be edited if "Absolute Levels" is set in the "Threshold Reference" field.
Upper Threshold [Volt] or rel. to UBatt	real64	Yes	Upper comparator threshold in volts.Value range: [0.0 V, 40.0 V] Can only be edited if "Absolute Levels" is set in the "Threshold Reference" field.
Lower Rise/Fall Threshold [Volt]	real64	Yes	Lower comparator threshold for rise and fall measurements in volts. Value range: [0.0 V, 40.0 V]
Upper Rise/Fall Threshold [Volt]	real64	Yes	Upper comparator threshold for rise and fall measurements in volts. Value range: [0.0 V, 40.0 V]
Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)	real64	Yes	Lower limit of the relevant angle window in °CA. Value range: [-720 °CA, 720 °CA[
Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)	real64	Yes	Upper limit of the relevant angle window in °CA. Value range: [-720 °CA, 720 °CA[
Asynchronous Timeout [ms]	uint16	Yes	Timeout time for asynchronous measurements. Value range: [1 ms, 32767 ms]

* Data type which the RTIO driver uses internally for the parameter

Tab. 11-8 ES1336-Hw Device: Configuration Parameters ("Signals" Tab)

11.5 ES1336-Meas Device

11.5.1 Globals (ES1336-Meas Device)

The ES1336-Meas device is used to specify and configure measurements. Every ES1336-Meas device offers 32 measurements which can be applied freely to the 20 hardware channels of the ES1336.1. Up to 9 ES1336-Meas devices are supported per ES1336.1 so that a total of 288 measurements can be configured.

No settings are to be made in the "Globals" tab.

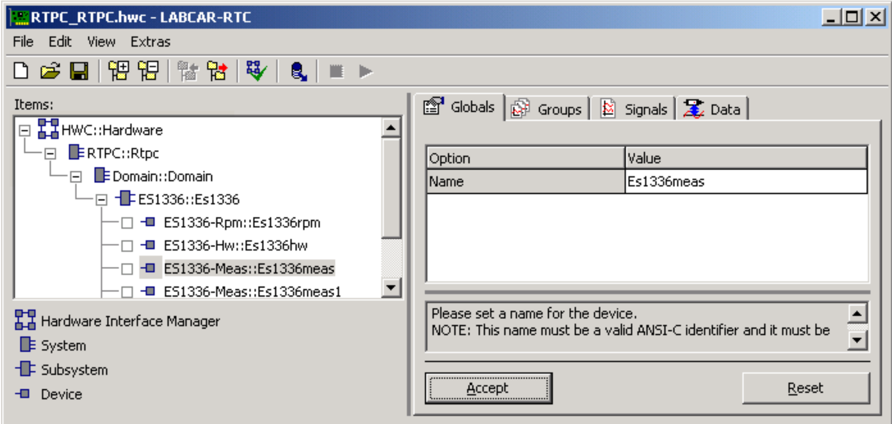


Fig. 11-17 The "Globals" Tab of the ES1336-Meas Device

11.5.2 Groups (ES1336-Meas Device)

The ES1336-Meas device has one signal group (Fig. 11-18) which is transferred from the ES1336.1 to the experimental target. It contains all measure data such as measure values, trigger and timeout information.

An operating system task with periodic activation is to be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

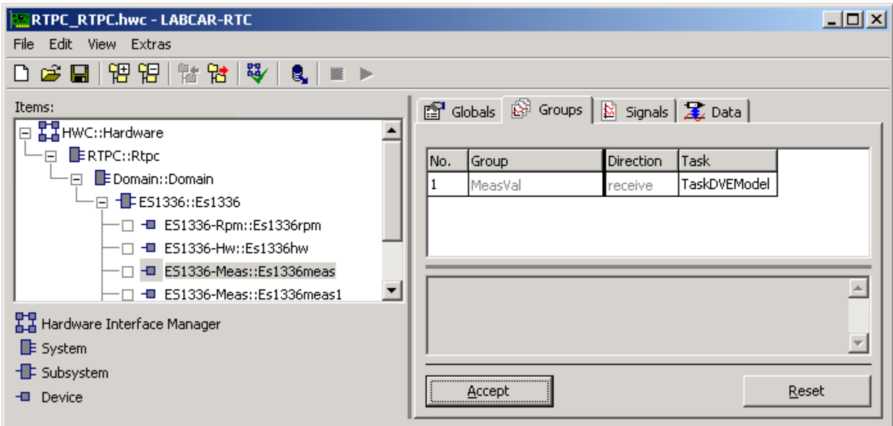


Fig. 11-18 The "Groups" Tab of the ES1336-Meas Device

The RTIO Signals of the "MeasVal" Group

The "MeasVal" group consists of 65 RTIO signals. The "TriggerBitField_00" signal must be interpreted as a bit field. It consists of 32 bits. The trigger or update data of the 32 measurements is coded in this bit field, i.e. it shows which measure values have been updated for the "MeasVal" signal group since the read process was last activated. If a bit is set, the measure value of the relevant measurement has been newly determined. It does not matter whether the measure value was updated as part of a timeout or due to a regular measure value calculation; the update bit of the measurement is set in both cases.

The RTIO signals "MeasVal_00" to "MeasVal_31" contain the measure values of the 32 measurements. These are either measure values which have been determined in the normal way or (in the case of a timeout) the timeout value intended for this case.

If a measurement is not used, 8888.0 is assigned to the relevant measure value. The physical unit of the measure values depends on the measurement procedure:

- time measurements (timestamp, (additive) pulse-width measurements, period durations) are specified in microseconds
- frequency measurements are in hertz
- angle measurements and angle stamps are specified in degrees crankshaft (°CA)
- voltage measurements are in volts
- integral evaluation in V*µs
- all other measurements (duty cycles, pulse count) are dimensionless

The RTIO signals "Tout_00" to "Tout_31" contain the results of the timeout monitoring for the relevant measure value.

Tab. 11-9 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
TriggerBitField_00	uint32	Bit field with update information of the 32 measurements. Measurement 0: LSB Measurement 31: MSB Bit value 0: measure value unchanged Bit value 1: measure value updated
MeasVal_00 ... MeasVal_31	real64	Measure value. If the measurement is not used, -8888.0 is issued as a value. Physical unit of the measure value: - time measurements are in µs - frequency measurements are in Hz - angle measurements are in °CA - voltage measurements are in V - integral evaluations are in V*µs
Tout_00 ... Tout_31	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive
* Data type which the RTIO driver uses internally for the parameter		

Tab. 11-9 ES1336-Meas Device: the RTIO Signals of the "MeasVal" Signal Group

11.5.3 Signals (ES1336-Meas Device)

The 32 measurements of an ES1336-Meas device are configured in the "Signals" tab. Fig. 11-19 shows the RTIO parameters of the "Signals" tab – Tab. 11-10 on page 292 summarizes the properties of the individual parameters.

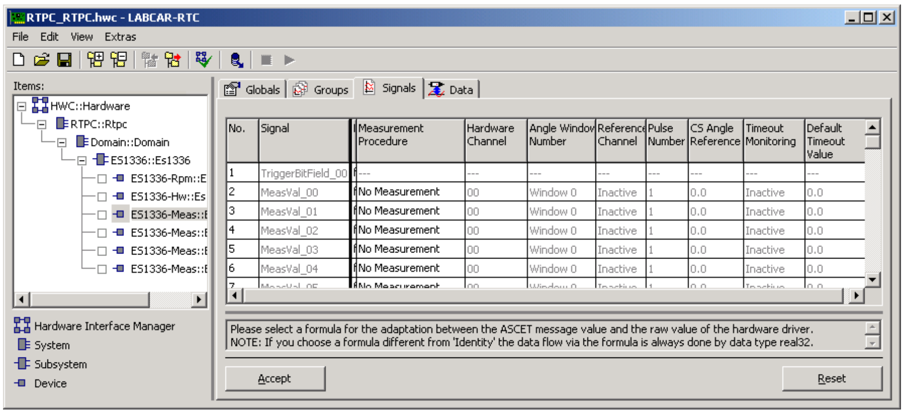


Fig. 11-19 The "Signals" Tab of the ES1336-Meas Device

All parameters can be edited during runtime of the model on the experimental target.

Measurement Procedure

This is where the measurement procedure is selected. The sections on page 292ff contain detailed descriptions and configuration guidelines for the individual procedures. The notation --/-- denotes evaluation at rising edges whereas --\-- denotes evaluation at falling edges.

If "No Measurement" is set in the list field, no measurement takes place.

Note

To avoid unnecessary computing time, measurements which are not required should be deactivated.

Hardware Channel

This is where you specify the hardware channel on which measurement is to take place.

Angle Window Number

This list field can only be edited if an angle-synchronous measurement procedure has been selected in the "Measurement Procedure" list field. The angle window is defined within which the measurements are to be executed. Three angle windows are available; these are the angle windows of the assigned hardware channel (for more information see "Signals (ES1336-Hw Device)" on page 284).

Reference Channel

The pulse-width measurements with Enable or Validate signal described in section 11.6.3 on page 294 require a second hardware channel for the purposes of execution: the channel at which the Enable or Validate signal is pending. The "Reference Channel" field displays the hardware channel at which (when this kind of measurement is selected) the Enable or Validate signal is to be connected.

The field is a display field only, which means it cannot be edited and is of no importance for all other measurements.

Pulse Number

This numeric input field can only be edited if an angle-synchronous measurement procedure has been selected in the "Measurement Procedure" field. With pulse- and edge-selective measurements, the number of the pulse or edge to be measured is specified in this field.

This field is of no importance for additive measurements and pulse counts, and is therefore disabled.

CS Angle Reference

The angle reference in degrees crankshaft to which all angle measurements refer. This parameter is of no importance to all other measurements and is thus disabled.

Timeout Monitoring

A distinction has to be made between asynchronous and angle-synchronous measurements in timeout monitoring. With asynchronous measurements, timeout monitoring takes place in periodic intervals. The time can be set separately for each hardware channel (for more details see "Signals (ES1336-Hw Device)" on page 284). A timeout is determined when no edge is determined on the hardware channel relevant to the measurement during the timeout period. Every edge on the hardware channel resets timeout monitoring.

With angle-synchronous measurements, timeout monitoring takes place every 720 °CA, at the upper angle window limit in each case. A timeout is detected when no measure value could be calculated. For example, a timeout is detected when the hightime of the fifth pulse in the angle window is to be measured, but only four pulses occurred.

In the "Timeout Monitoring" list field, timeout monitoring can be deactivated ("Inactive" option) or activated ("Intvl Predef" and "Intvl InpDep" options). The difference between the options "Intvl Predef" and "Intvl InpDep" is the measure value which, in the case of a timeout, is transferred to the model. If the "Intvl Predef" option is set, the value specified in the numeric input field "Default Timeout Value" is transferred. With the "Intvl InpDep" option, the value returned depends on the measurement procedure, although this option is actually only available with specific measurement procedures. For more details, refer to the sections starting on page 292.

Default Timeout Value

This numeric input field can only be edited if "Intvl Predef" has been set in the "Timeout Monitoring" field. In this case, the input field defines the measure value which is transferred to the model in the case of a timeout.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Measurement Procedure	uint32	Yes	Measurement procedure Measurement procedure (parameter value) "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	Yes	Hardware channel on which the measurement is executed. 0 ... 19: channel number
Angle Window Number	uint8	Yes	Angle window selection (value range: 0, 1, 2) Only relevant / editable with angle-synchronous measurements.
* Data type which the RTIO driver uses internally for the parameter			

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Reference Channel	-	No	Reference channel for measurements with Enable and Validate signal. Display field only; cannot be edited.
Pulse Number	uint16	Yes	With pulse- and edge-selective measurements: the number of the pulse or edge to be measured (value range: 1, 2, ... 32)
CS Angle Reference	real64	Yes	Angle reference in degrees crankshaft to which all angle measurements refer (Value range: [-720.0 °CA, 720.0 °CA[)
Timeout Monitoring	uint8	Yes	Timeout monitoring. 0: "Inactive" 1: "Intvl Predef" 2: "Intvl InpDep" "Intvl InpDep" is only available with the following measurement procedures: - "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	Yes	Measure value in the case of a timeout. Only relevant in the "Intvl Predef" mode for timeout monitoring.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 11-10 ES1336-Meas Device: Configuration Parameters of the "Signals" Tab

11.6 The Measurement Procedures

11.6.1 Asynchronous Measurements

Asynchronous measure values are always calculated when no angle-synchronous measure values are to be calculated, but at least once per camshaft revolution (720 °CA).

The last three edges detected before the calculation time are evaluated for the calculation of

- frequency ("Asynchron: Frequency --/-- [Hz]")
- period duration ("Asynchron: Cycle Time --/-- [µs]")
- duty cycles ("Asynchron: Duty Cycle H/(L+H) --/--", "Asynchron: Duty Cycle L/(L+H) --/--")
- pulse widths ("Asynchron: Hightime [µs]", "Asynchron: Lowtime [µs]")
- and rise and fall times ("Asynchron: Rising Time [µs]", "Asynchron: Falling Time [µs]").

At the time of calculation 128 analog values are read from the A/D converter to measure voltage ("Asynchron: Input Voltage [V]" measurement procedure) and an average calculated.

For asynchronous measurements, time-synchronous timeout monitoring can be activated. If this is active, exactly one timeout is triggered if no edge was detected on the relevant hardware channel within the timeout interval. Every edge on the hardware channel resets monitoring.

With the

- "Asynchron: Duty Cycle H/(L+H) --/--"
- "Asynchron: Duty Cycle L/(L+H) --/--"

duty cycle measurements, the timeout mode "Intvl InpDep" can be set.

The measure values transferred to the model when a timeout occurs are listed in Tab. 11-11.

Measurement Procedure	Signal Level during a Timeout Check	Measure Value in a Timeout
Asynchron: Duty Cycle H/(L+H) --/--	low	0
	high	1
Asynchron: Duty Cycle L/(L+H) --/--	low	1
	high	0

Tab. 11-11 Asynchronous Measurements: Measure Value in a Timeout in the "Intvl InpDep" Timeout Mode

11.6.2 Angle-Synchronous Measurements - Pulse-Width Measurements

This section describes the properties of pulse-width measurements. The functioning of the following measurements is described:

- Angular: H-Time n-th Pulse (Pu Qual.) [µs]
- Angular: L-Time n-th Pulse (Pu Qual.) [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the pulse to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" (the "Intvl InpDep" option is not supported)
- Default Timeout Value

Functional Description

Pulse-width measurements are pulse-qualified measurements, i.e. only those pulses are measured and counted whose opening and closing edge are within one and the same angle window (valid pulse). The pulse-width of the n-th valid pulse in the selected angle window is determined. The measure value is calculated at the upper angle window limit.

Fig. 11-20 shows how this works using the example of a low-active pulse-width measurement.

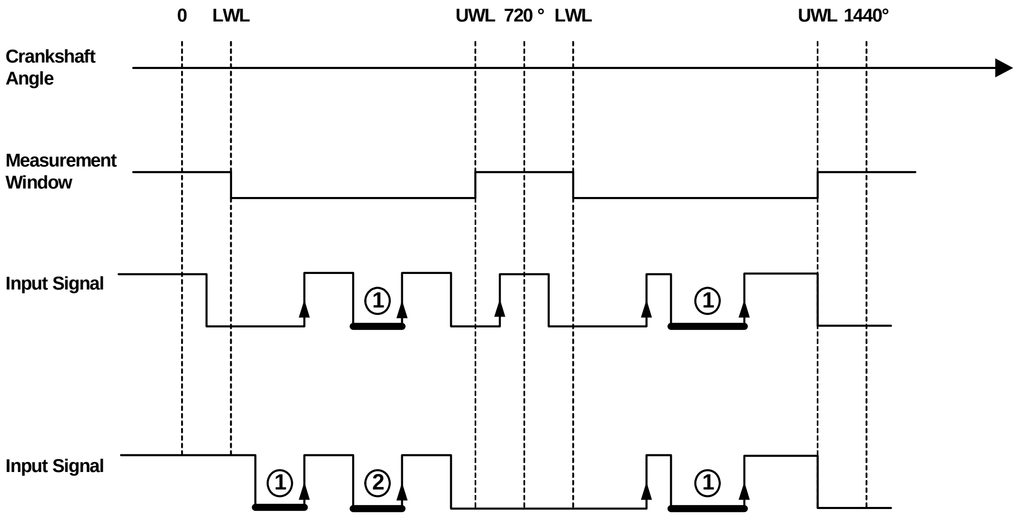


Fig. 11-20 Representation of how pulse-qualified, low-active pulse-width measurement (Angular: L-Time n-th Pulse (Pu Qual.) [µs] measurement procedure) works. Measured pulses or pulses which are counted are indicated by the bold lines.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the lowtime of the fifth pulse is to be measured, but only four pulses occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

11.6.3 Pulse-Width Measurements with Enable or Validate Signal

In addition to the direct acquisition of pulse widths of a signal which is independent of all other hardware channels, the ES1336.1 makes it possible to measure pulse widths of a signal according to events on a second hardware channel. This second signal is referred to as an Enable or Validate signal.

Pulse-Width Measurements with an Enable Signal

In the case of a pulse-width measurement with an Enable signal, a pulse on the measure channel is only measured or counted if it is completely surrounded by a pulse on the second channel (referred to below as the Enable channel). Fig. 11-21 shows a low-active measurement signal and a high-active Enable signal. Only the first of the two low pulses of the measurement signal is "enabled"; it is the only one that is completely surrounded by a high pulse of the Enable channel.

Pulse-Width Measurements with a Validate Signal

In the case of a pulse-width measurement with a Validate signal, a pulse on the measure channel is only measured or counted if it completely surrounds a pulse on the second channel (referred to below as the Validate channel). Fig. 11-21 shows a low-active measurement signal and a high-active Validate signal. Only the second of the two low pulses of the measurement signal is validated; only this pulse completely surrounds a high pulse of the Validate channel.

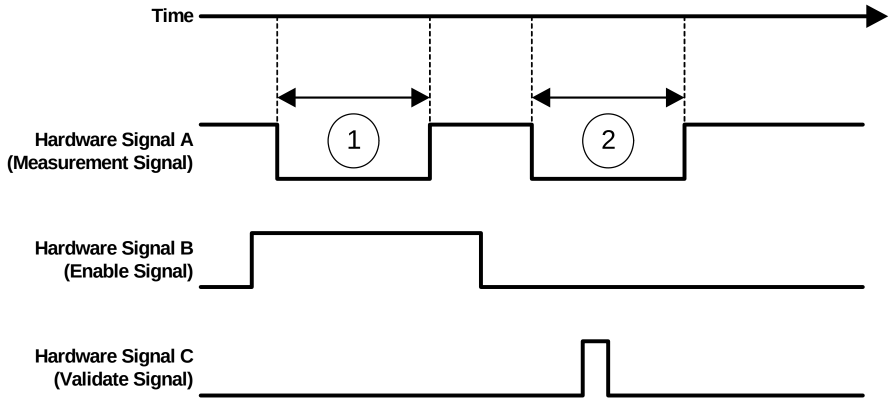


Fig. 11-21 Pulse-Width Measurements with Enable and Validate Signal

The ES1336.1 offers pulse-width measurements with an Enable or Validate option. The Enable and Validate pulses can be defined either as high-active or low-active. The following measurements are possible:

- Angular: H-Time n-th Pulse (H-Enable) [µs]
- Angular: H-Time n-th Pulse (L-Enable) [µs]
- Angular: H-Time n-th Pulse (H-Valid.) [µs]
- Angular: H-Time n-th Pulse (L-Valid.) [µs]
- Angular: L-Time n-th Pulse (H-Enable) [µs]
- Angular: L-Time n-th Pulse (L-Enable) [µs]
- Angular: L-Time n-th Pulse (H-Valid.) [µs]
- Angular: L-Time n-th Pulse (L-Valid.) [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the measure channel
- Pulse Number

  Defines the number of the pulse to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" ("Intvl InpDep" is not supported)
- Default Timeout Value

Note

The number of the Enable or Validate channel is predefined. It is shown in the "Reference Channel" list field. The hardware channels 0 and 1 can make a pair whereby 0 can be selected as a measure channel, channel 1 is then the Enable or Validate channel. Channel 1 can, however, also be selected as measure channel; in this case, channel 0 would be the Enable or Validate channel. The same is true for each of the channel pairs (2, 3), (4, 5), (6, 7), (8, 9), (10, 11), (12, 13), (14, 15), (16, 17), (18, 19).

Functional Description

Fig. 11-22 shows how angle-synchronous pulse-width measurements with Enable option work using a lowtime measurement with a high-active Enable signal. Only those pulses are taken into consideration whose opening and closing edge are within one and the same angle window, and which are completely surrounded by an active Enable pulse ("enabled pulse").

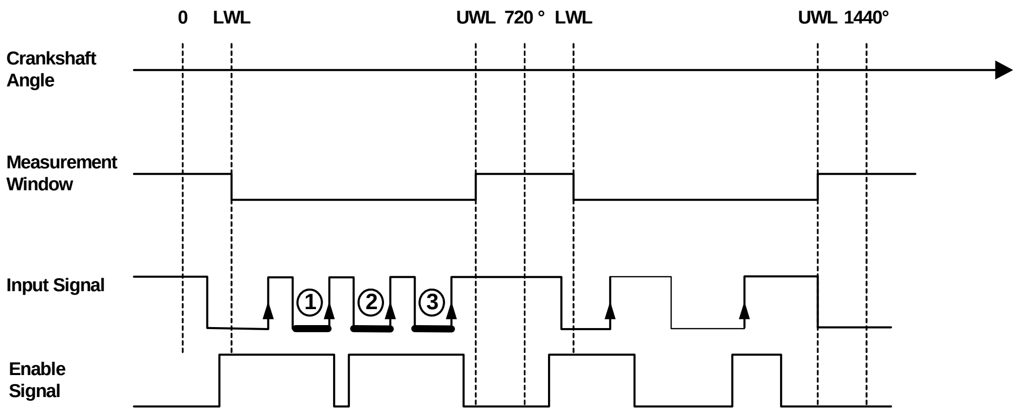


Fig. 11-22 Representation of how angle-synchronous pulse-width measurement works with an Enable signal ("Angular: L-Time n-th Pulse (H-Enable) [µs]" procedure). Enabled pulses are indicated by bold lines.

Fig. 11-23 shows how angle-synchronous pulse-width measurements with Validate option work using a lowtime measurement with a high-active Validate signal. Only those pulses are taken into consideration whose opening and closing edge are within one and the same angle window, and which completely surround an active Validate pulse ("validated pulse").

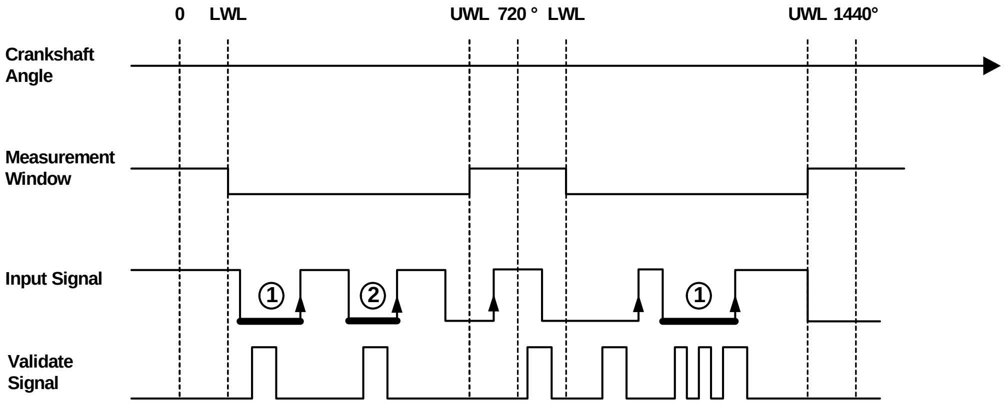


Fig. 11-23 Representation of how an angle-synchronous pulse-width measurement works with a Validate signal ("Angular: L-Time n-th Pulse (H-Valid) [µs]" measurement procedure). Validated pulses are indicated by bold lines.

The n-th enabled or validated pulse within an angle window is measured. The pulse number is defined in the "Pulse Number" list field of the "Signals" tab.

The measured pulse widths are transferred to the model at the upper angle window limit.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the lowtime of the fifth validated or enabled pulse is to be measured, but only four such pulses occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

11.6.4 Additive Pulse-Width Measurements

This section summarizes the properties of additive pulse-width measurements. The functioning of the following measurements is described:

- Angular: Additive Hightime [µs]
- Angular: Additive Lowtime [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Angle Window Number

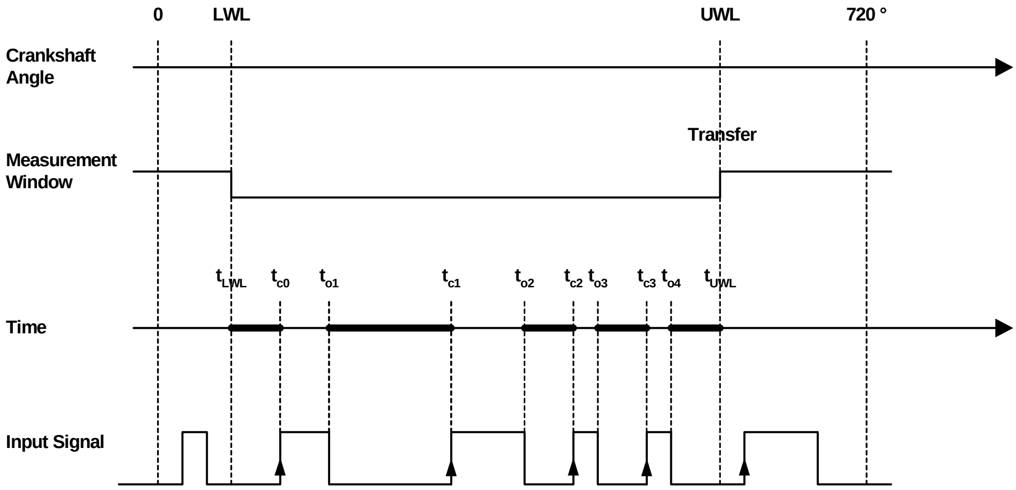
  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  All three options (Inactive, Intvl InpDep and Intvl Predef) are supported.

- Default Timeout Value

Functional Description

The additive time is the total of all time segments within an angle window in which the signal is active, regardless of whether the opening or closing edges of the pulses are inside or outside the angle window. Fig. 11-24 shows an example of measurement value calculation with an additive lowtime measurement. The additive time is transferred to the model at the upper angle window limit.



$$t_{Additive} := (t_{c0}-t_{LWL}) + (t_{UWL}-t_{o4}) + \Sigma\,(t_{ci} - t_{oi}), \quad i = 1 \dots 3$$

Fig. 11-24 Representation of the functioning of an angle-synchronous additive lowtime measurement (Angular: Additive Lowtime [µs] measurement procedure). The additive time consists of the total of the bold line segments.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected when no edge occurred within the angle window.

Tab. 11-12 shows a list of measure values which are transferred to the model if a timeout occurs when "Intvl InpDep" is selected in the "Timeout Monitoring" list field.

Measurement Procedure	Signal Level in the Angle Window	Measure Value in a Timeout
Angular: Additive Hightime [µs]	low	0
	high	Time difference: time of upper angle window limit – time of lower angle window limit ($t_{UWL} - t_{LWL}$)
Angular: Additive Lowtime [µs]	low	Time difference: time of upper angle window limit – time of lower angle window limit ($t_{UWL} - t_{LWL}$)
	high	0

Tab. 11-12 Additive Pulse-Width Measurements: Measure Value in a Timeout when "Intvl InpDep" is set in the "Timeout Monitoring" List Field

11.6.5 Measuring Edges: Angle Stamp

This section describes the acquisition of angle stamps of edges. The functioning of the following measurements is described:

- Angular: Rising Edge of n-th Pulse [°CA]
- Angular: Falling Edge of n-th Pulse [°CA]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the edge to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- CS Angle Reference

  Reference angle
- Timeout Monitoring

  Either Inactive or Intvl Predef; "Intvl InpDep" is not supported
- Default Timeout Value

Functional Description

The measurements determine the angle of the n-th rising or falling edge within an angle window. The edge counter is reset to 0 when the lower angle window limit is exceeded.

The crankshaft angle of an edge delivered by the hardware is generally in the range [0 °CA, 720 °CA[. The measured angle is mapped by the ES1336.1 firmware in the range [LWL, LWL + 720 °CA[and the difference is then calculated using the reference angle. This difference is the measure value provided by measurement; it is positive when the angle in question is less than the reference angle (cf. Fig. 11-25 on page 300).

An example: let’s assume an angle window of [-480 °CA, 120 °CA] has been defined. The reference angle should be around -80 °CA. In a first case, the measured angle should be around 60 °CA. No mapping is necessary for angle values between 0 °CA and (-480 °CA + 720 °CA = 240 °CA); the measure value supplied is (-80 °CA - 60 °CA = -140 °CA).

In a second case, the measured angle should be around 540 °CA. This angle is greater than 240 °CA and is mapped so that it is in the angle window. 720 °CA are thus subtracted from the angle value (540 °CA - 720 °CA = -180 °CA). The measure value is then (-80 °CA + 180 °CA = 100 °CA).

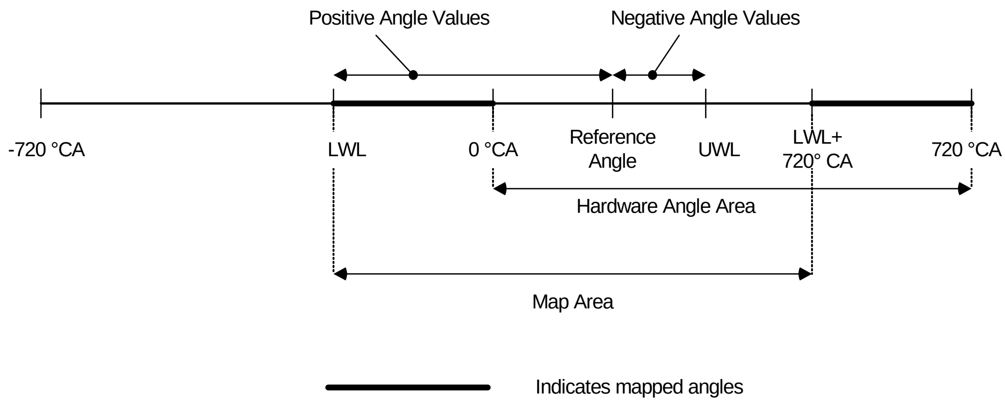


Fig. 11-25 How Angle Measurement Works

As the reference angle can be in the range [-720 °CA, 720 °CA], the measure functions return angle values from the range]-1440 °CA, 1440 °CA].

Note

The resolution of the measured angles is specified by the angle clock generator.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the angle of the fifth rising edge is to be measured, but only four such edges occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

11.6.6 Measuring Edges: Time Stamps

This section describes the acquisition of time stamps of edges. The functioning of the following measurements is described:

- Angular: Time Stamp of n-th Rising Edge [µs]
- Angular: Time Stamp of n-th Falling Edge [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the edge to be measured

- Angle Window Number

  Defines the number of the angle window in which measuring is to take place.
- Timeout Monitoring

  Either "Inactive", "Intvl Predef" or "Intvl InpDep"
- Default Timeout Value

Functional Description

The measurements determine the time stamp of the n-th rising or falling edge within an angle window. The edge counter is reset to 0 if the lower angle window limit is exceeded.

Note

The time stamp returned has no connection to other time stamps in the LABCAR system. It is a purely internal ES1336.1 time stamp with a resolution of 80 ns.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected if the measure value could not be calculated because, for example, the time stamp of the fifth rising edge is to be measured, but only four such edges occurred.

If "Intvl InpDep" is set in the "Timeout Monitoring" list field, the time stamp determined in the last camshaft revolution to have no timeout is transferred as measure value to the model. In other words: the old time stamp is retained.

If "Intvl Predef" is set in the "Timeout Monitoring" list field, the default timeout value defined by the user is transferred to the model as measure value.

11.6.7 Measuring Edges: Rise and Fall Times

This section describes the measuring of rise and fall times of edges. The functioning of the following measurements is described:

- Angular: Rising Time of n-th Pulse [µs]
- Angular: Falling Time of n-th Pulse [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the edge to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" ("Intvl InpDep" is not supported)
- Default Timeout Value

Functional Description

The measurements determine the rise time of the n-th rising or the fall time of the n-th falling edge within an angle window. The resolution of the measure values is 80 ns.

Please note that **different comparator thresholds** are decisive for measuring rise and fall times and thus also for the related counting of edges. This is explained in Fig. 11-26. If the angle of the n-th rising edge is measured on the signal shown, the comparator thresholds "Lower Threshold" and "Upper Threshold" are decisive for this measurement. Three rising edges thus occur with these thresholds in the angle window.

For rise and fall time measurements, however, the thresholds **"RFT Lower Threshold"** and **"RFT Upper Threshold"** are decisive. Two rising and two falling edges thus occur with these thresholds.

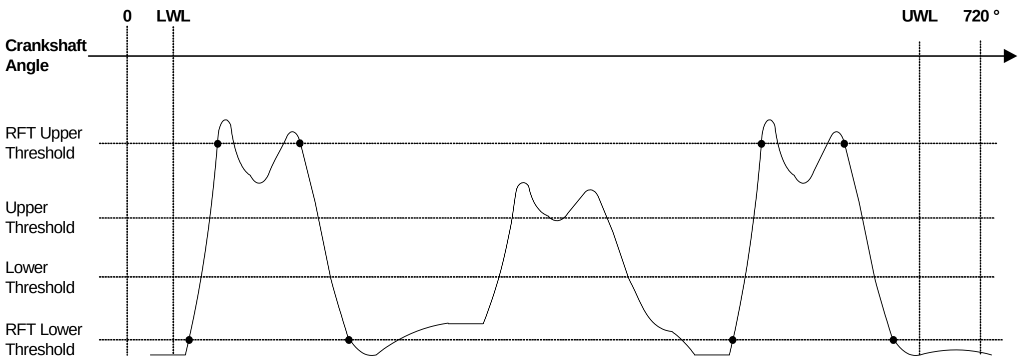


Fig. 11-26 Edge Counting when Measuring Rise and Fall Times

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the rise time of the fifth rising edge is to be measured, but only four such edges occurred, the default timeout value defined by the user is transferred to the model as measure value.

The "Intvl InpDep" option is not supported.

11.6.8 Peak Value Measurement

This section describes the properties of the following measurement procedure for peak value measuring:

- Angular: Peak-Amplitude of n-th Pulse [V]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Pulse Number

  Defines the number of the pulse to be measured

- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" ("Intvl InpDep" is not supported)
- Default Timeout Value

Functional Description

Measurement determines the peak voltage of the n-th pulse within an angle window. For a pulse to be measured or counted, both its opening and closing pulse edge have to be in one and the same angle window.

The measured voltage has a resolution of 12 bits – the measure range is from 0 to 40 volts.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the peak value of the fifth pulse is to be measured, but only four such pulses occurred, the default timeout value defined by the user is transferred to the model as measure value.

The "Intvl InpDep" option is not supported.

11.6.9 Pulse Integration

This section describes the properties of the following measurement procedure for pulse integration:

- Angular: Integral U*dt of n-th pulse [V*µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Pulse Number

  Defines the number of the pulse to be measured
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" ("Intvl InpDep" is not supported)
- Default Timeout Value

Functional Description

The measurement determines the integral of the n-th pulse within an angle window. For a pulse to be measured or counted, both its opening and closing pulse edge have to be in the same angle window.

The pulse integration method is described in the section "Pulse Integration" on page 272.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the integral of the fifth pulse is to be measured, but only four such pulses occurred, the default timeout value defined by the user is transferred to the model as measure value.

The "Intvl InpDep" option is not supported.

11.6.10 Pulse Counting

This section describes the properties of the following measurement procedures for pulse counting:

- Angular: Number of High-Pulses
- Angular: Number of Low-Pulses

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef"
- Default Timeout Value

Functional Description

The measurement procedures determine the number of valid active pulses within an angle window. Valid pulses are pulses which are in the same window with both an opening and a closing edge.

The total determined is transferred to the model at the upper angle window limit. (Fig. 11-27).

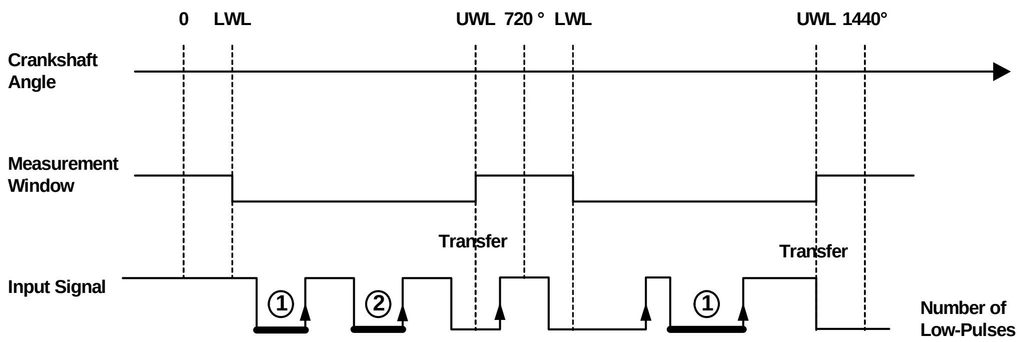


Fig. 11-27 Representation of how angle-synchronous measurement procedure for pulse count (Angular: Number of Low-Pulses Measurement Procedure) takes place. Pulses which are counted are indicated by bold lines.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected when no edge occurred in the angle window.

If "Intvl Predef" is set in the "Timeout Monitoring" list field, the default timeout value defined by the user is transferred to the model as measure value.

12 ES1337.1 Wheelspeed Sensor Simulation Board

The ES1337.1 Wheelspeed Sensor Simulation Board has four independent signal generators for generating various sensor signals. The following types of sensor can be simulated:

- Passive analog sensors with a sinusoidal output signal (type DF6)
- Active digital sensors with a current interface with two current levels (type DF10)
- Active digital sensors with a current interface with three current levels and forwards/backwards coding (type DF10-RotDir)
- Active digital sensors with a current interface with two current levels and additional information (type DF11i)
- Active digital sensors with a current interface with three current levels and additional information (type VDA)

In addition, there are two other galvanically isolated controllable voltage outputs.

The Structure of the ES1337.1 RTIO Tree

In the RTIO Editor, the ES1337.1 Wheelspeed Sensor Simulation Board is integrated by selecting the ES1337 subsystem and then an ES1337-Wheelsnsrsim subsystem.

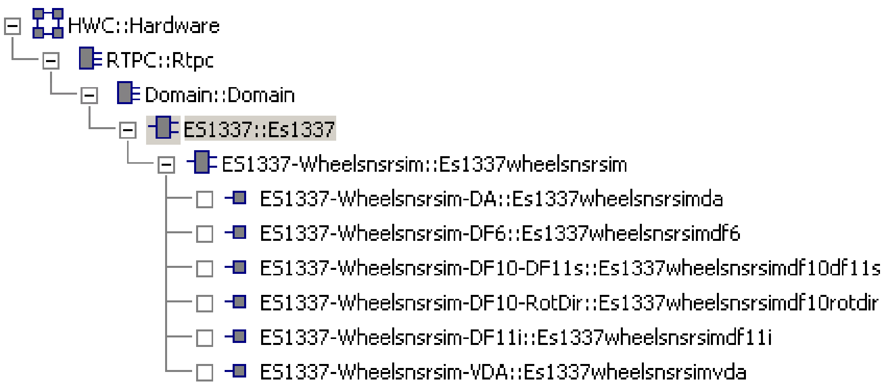


Fig. 12-1 RTIO Hardware Description with Integrated ES1337.1

One ES1337-Wheelsnsrsim-DA device and up to four devices for simulating wheel speed sensors can be added under the ES1337-Wheelsnsrsim subsystem. These devices can be combined in any way.

With ES1337-Wheelsnsrsim-DA devices, output channels 4 and 5 can be used to output analog voltages. Channels 0 …3 are used for outputting simulated sensor signals (current or voltage signal – depending on the type of sensor simulated).

12.1 ES1337 Subsystem

12.1.1 Globals (ES1337 Subsystem)

The ES1337 subsystem is used to set globally valid parameters, i.e. which have an effect on all ES1337.1 RTIO elements.

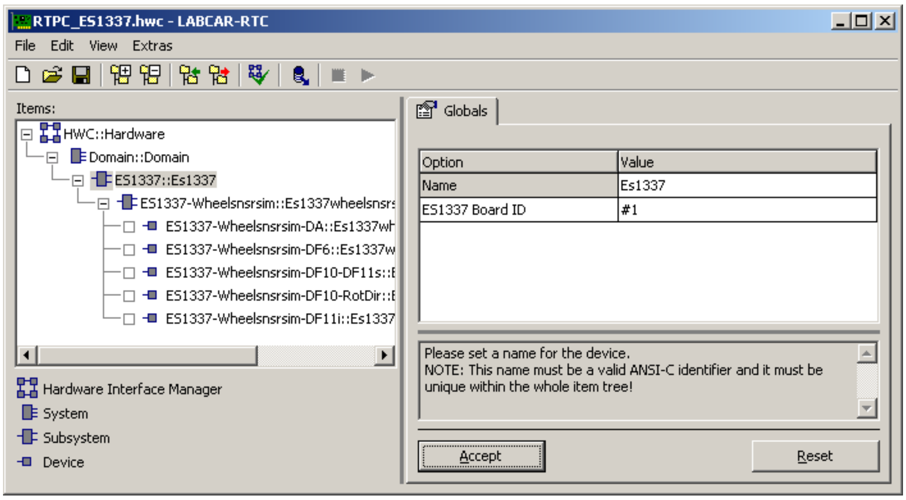


Fig. 12-2 The "Globals" Tab of the ES1337 Subsystem

ES1337 Board ID

This option field is used to identify the ES1337.1. It establishes the assignment between the RTIO hardware description and the ES1337.1 in the VMEbus chassis for which this description is valid.

The boards are numbered from left to right starting with 1 in the chassis (left = VMEbus slot no. 1, in ascending order). The number of the ES1337.1 determined must be entered in the "ES1337 Board ID" list field. Up to 20 ES1337.1s can be integrated per chassis in the RTIO Editor.

12.1.2 Globals (ES1337-Wheelsnsrsim Subsystem)

The ES1337-Wheelsnsrsim subsystem is used to set globally effective parameters, i.e. which have an effect on all elements of the ES1337-Wheelsnsrsim subsystem.

The following figure shows the parameters of the "Globals" tab.

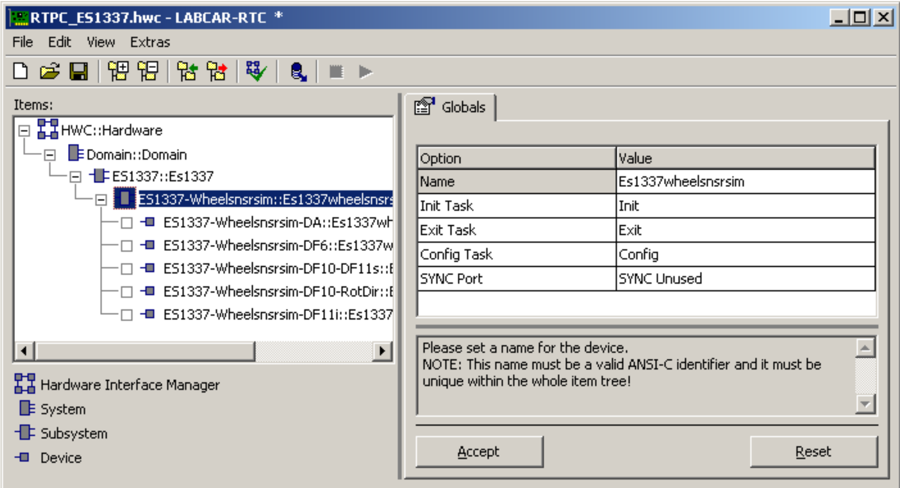


Fig. 12-3 The "Globals" Tab of the ES1337-Wheelsnsrsim Subsystem

The "Output U/I Ratio" option field is only visible after selecting **View → Show All**.

Init Task, Exit Task, Config Task

In these option fields, you can select the tasks to which the generated Init, Exit and Config processes of the driver are to be assigned.

SYNC Port

In this option field, you can specify which channel/output the synchronization signal at the "SYNC" port on the front panel comes from. This signal is active throughout tooth 0 of the selected sensor.

Output U/I Ratio

In this option field, you can set the transconductance of the U/I converter (post-connected to the D/A converter). The value is specified in mA/Uref with Uref=10 Volt (always). The default value is 40 mA/10 V = 4 mA/V (corresponding to the output circuit of the ES1337.1)

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Init Task, Exit Task Config Task	Selec-tion	No	Tasks for the Init, Exit and Config processes
SYNC Port	uint8	Yes	"Tooth 0" pulse of the selected output/sensor Value range: Output 0 … Output 3
Output U/I Ratio	real32	No	Conversion factor for voltage current converter (Uref=10V) Value range: 1.0 …1000.0 [mA/Uref], default: 40 mA/Uref = 4 mA/V
* Data type which the RTIO driver uses internally for the parameter			

Tab. 12-1 ES1337-Wheelsnsrsim Subsystem: Configuration Parameters of the "Globals" Tab

12.2 ES1337-Wheelsnsrsim-DA Device

With the ES1337-Wheelsnsrsim-DA device, you can address and configure channels 4 and 5 which can be used as analog voltage outputs.

12.2.1 Globals (ES1337-Wheelsnsrsim-DA Device)

Output-specific configuration settings can be made in the "Globals" tab.

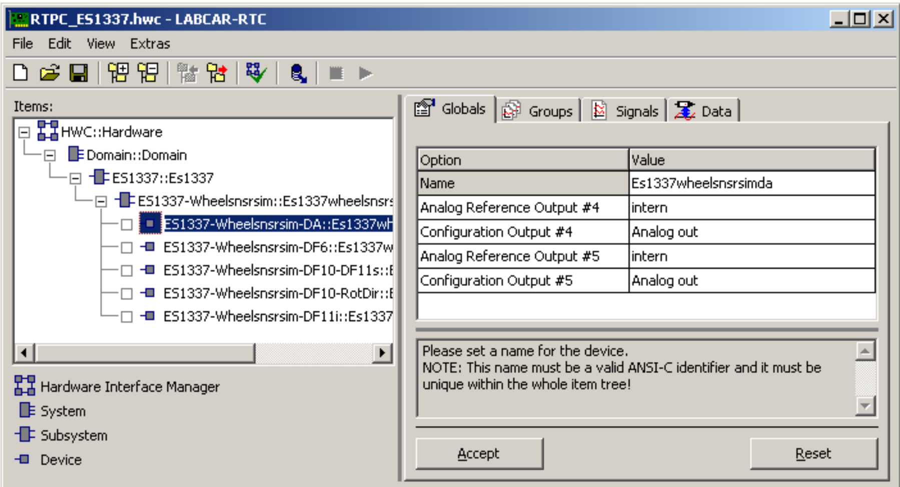


Fig. 12-4 The "Globals" Tab of the ES1337-Wheelsnsrsim-DA Device

Analog Reference Output #4, #5:

In these option fields, you can specify the origin of the reference voltages of the analog channels 4 and 5. You can choose from the internal reference of 10 V or a voltage applied to the external reference voltage input of the channel between -10.0 V and +10.0 V.

The actual output voltage of the relevant channel is then the product of the reference voltage and the control signal. This setting is only of any significance with an analog configuration of the channel (see below).

Configuration Output #4, #5

In these option fields, you can modify the configuration of outputs 4 and 5. Operation as an analog output or as a binary output, either as an open drain or with an additional current source as pull-up to +5 V, is possible. When configured as a binary output, the output is low with a control pulse less than 0.1 and high when equal to or greater than 0.1.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Analog Reference Output #4, #5	uint	Yes	Origin of the reference voltage Value range: Internal | External
Configuration Output #4, #5	uint	Yes	Configuration of the output Value range: Analog Out | Digital without Pull-Up| Digital with Internal Pull-Up

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-2 ES1337-Wheelsnsrsim-DA Subsystem: Configuration Parameters of the "Globals" Tab

12.2.2 Groups (ES1337-Wheelsnsrsim-DA Device)

The ES1337-Wheelsnsrsim-DA device has an "Output" signal group for controlling the analog outputs.

12.2.3 Data (ES1337-Wheelsnsrsim-DA Device)

In the signal group, two signals are processed per analog channel. Use the signals "Enable_4" and "Enable_5" to enable or disable the relevant channel. When disabled, the channel output is high-impedance.

Use the signals "Output_4" and "Output_5" to address the relevant output. In analog operation, the product results from this signal and the reference voltage as output voltage. One of the binary modes results in a logical high level with a signal value equal to or greater than 0.1.

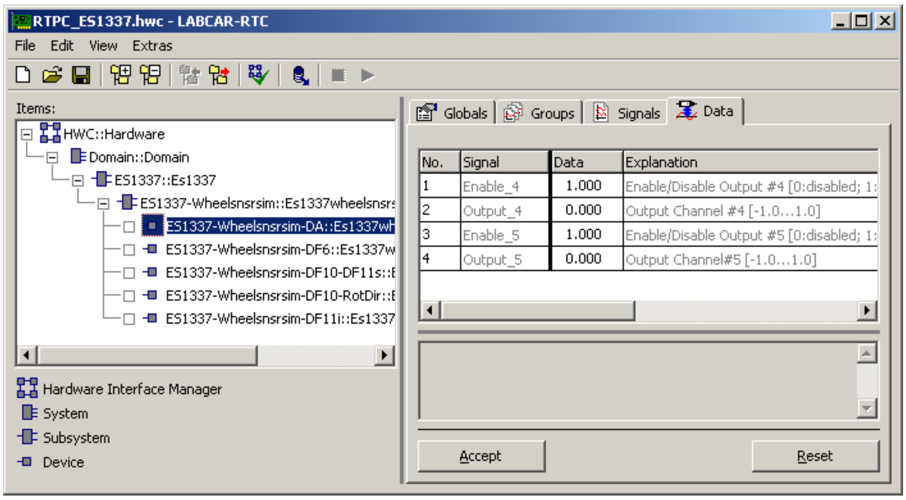


Fig. 12-5 The "Data" Tab of the ES1337-Wheelsnsrsim-DA Device

The following table summarizes the properties of the signals.

RTIO Signal	Data Type*	Comment/Value Range
Enable_4, Enable_5	uint8	Enabling/disabling outputs 4, 5 0: Disabling the output 1: Enabling the output
Output_4, Output_5	uint8	Output value, Value range: -1.0...1.0

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-3 ES1337-Wheelsnsrsim-DA Device: Signals of the "Data" Tab

12.3 Wheel Sensor Simulation – Common Settings

The RTIO elements for representing the simulated sensors have an identical structure and only differ in the specific features of the different sensors.

Analog Output

Every sensor can be assigned to an output in the "Analog Output" option field in the "Globals" tab. The following assignment is fixed between the assembly position of the sensor and the analog output:

Analog Output	Wheel Position of the Simulated Sensor
# 0	Front left (FL)
# 1	Front right (FR)
# 2	Rear left (RL)
# 3	Rear right (RR)

Operation Mode

Use the "Operation Mode" option field to define whether the wheel speed should be specified via velocity, number of teeth, wheel radius or directly via the tooth frequency of the sensor.

Number of Teeth, Reaction to Tooth Faults

Apart from the number of teeth (or pole pairs) per revolution, you can define for each tooth whether it should be output regularly or as a fault. With all defective teeth, the voltage or current value defined in the option field is output throughout the tooth. The effect of tooth faults depends on the "ToothFaultEnable" signal.

Tooth faults are defined via a proprietary input dialog whereby individual tooth numbers or tooth ranges separated by commas can be entered. Ranges are defined in the form

```
<first tooth> + <-> + <last tooth>
```

Numbering starts with 0, i.e. for a wheel with 48 teeth/pole pairs the range is 0 – 47.

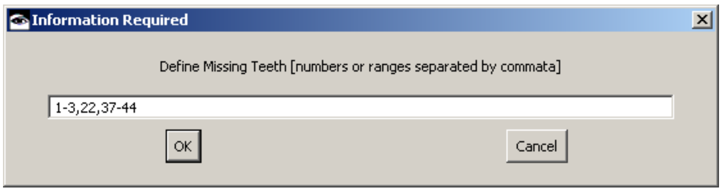


Fig. 12-6 Input Dialog for Defining Missing Teeth

12.3.1 Signals Common to all Sensors

Enable

This signal activates or deactivates the sensor. When inactive, the relevant output is high-impedance. When activated, the sensor is restarted and the output switched to the voltage (with type "DF6") or current output (all other types of sensor). When the sensor is restarted, you start with tooth 0.

Note

Please note that the output is switched using relays which is why certain switching and/or bounce times occur.

Frequency

This signal is used to directly specify the frequency of the sensor (only effective in the operation mode "Frequency", see "Operation Mode" on page 312).

With sensors which detect the direction of rotation, a negative value changes the direction of rotation (positive value: forward or normal direction of rotation, negative value: backwards). The sign is ignored by sensors which cannot detect the direction of rotation.

WheelSpeed, WheelRadius

This signal is used to specify wheel speed and wheel radius. In the operation mode "Wheelspeed", the frequency is calculated as follows from these values together with the number of teeth/pole pairs.

$$f\,[\mathrm{Hz}] = \frac{v\,[\mathrm{km/h}] \cdot z}{r\,[\mathrm{m}] \cdot 2\pi \cdot 3{,}6}$$

In the operation mode "Frequency" the signals have no effect. The same is true for a negative value of wheel speed as with "Frequency", i.e. a change in the direction of rotation or rotating backwards is simulated for sensors which can detect the direction of rotation.

ToothFaultEnable

With this signal, you can enable (= 1) or disable (= 0) the settings for tooth faults made in the "Globals" tab. For every tooth or every pole pair you can determine whether an output should take place regularly or as a fault. A specified level (with a "DF6" voltage and with all other sensor types current) is output for all defective teeth of a sensor.

Accepting Signal Changes at the Outputs

Changes to frequency or wheel speed become effective when the register is written at the subsequent pole change (half tooth) at the relevant output. The sensor type "DF6" is an exception; changes take immediate effect here.

Protocol information (type "DF11i" and "VDA") is not synchronized with tooth or pole changes. The values available when the register is accessed are used.

The enabling/disabling of tooth fault simulation with the "ToothFaultEnable" signal is not synchronized with a change of tooth, pole or revolution either. This could result in incomplete output of protocol or tooth information with an unfavorable switch time.

12.3.2 Globals (ES1337-Wheelsnsrsim-DF6 Device)

With the ES1337-Wheelsnsrsim-DF6 device you can simulate a "DF6" wheel sensor at one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

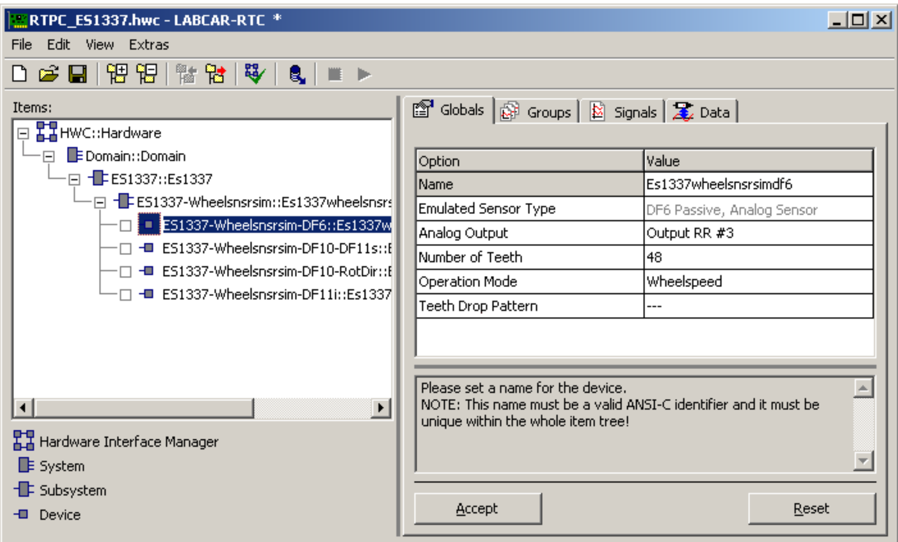


Fig. 12-7 The "Globals" Tab of the ES1337-Wheelsnsrsim-DF6 Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Analog Output

In this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

In this parameter, you can specify the number of teeth or pole pairs.

Operation Mode

In this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

In this option you can define a teeth drop pattern. The voltage value 0 V is output with a tooth defined as defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identifier	No	User-specific name
Analog Output	uint8	No	Output used [0 = FL #0 | 1 = FR #1 |2 = RL #2 | 3 = RR #3]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128]
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simulated as defective
* Data type which the RTIO driver uses internally for the parameter			

Tab. 12-4 ES1337-Wheelsnsrsim-DF6 Device: Configuration Parameters of the "Globals" Tab

12.3.3 WheelsnsrsimGroups (ES1337-Wheelsnsrsim-DF6 Device)

The ES1337-Wheelsnsrsim-DF6 device has a signal group "Output" for controlling sensor simulation.

12.3.4 Data (ES1337-Wheelsnsrsim-DF6 Device)

The signals for controlling sensor simulation are listed in the "Data" tab.

The following figure shows the signals of the "Data" tab:

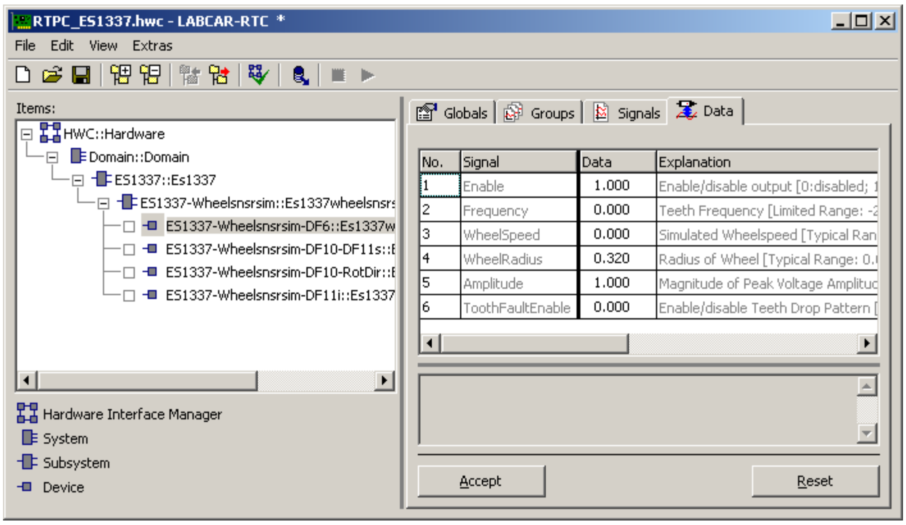


Fig. 12-8 The "Data" Tab of the ES1337-Wheelsnsrsim-DF6 Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (only with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (only with operation mode "Wheelspeed").

Amplitude

Use this signal to set the peak value of the sinusoidal voltage at the output.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-2500...2500]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02…2 m]
Amplitude	real32	Yes	Peak amplitude [0.0…10.0 V]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0, 1]

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-5 ES1337-Wheelsnsrsim-DF6 Device: Signals of the "Data" Tab

Note

The resulting frequency is limited to 2500 Hz regardless of the operation mode – when exceeded, a corresponding error message is generated.

Accepting Signal Changes

Frequency, wheel speed and amplitude changes become effective as soon as the relevant task is called. With type "DF6" the amplitude is a signal so as to be able to realize, for example, a speed-dependent amplitude.

12.3.5 Globals (ES1337-Wheelsnsrsim-DF10-DF11s Device)

With the ES1337-Wheelsnsrsim-DF10-DF11s device you can simulate a "DF10" or "DF11s" wheel sensor at one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

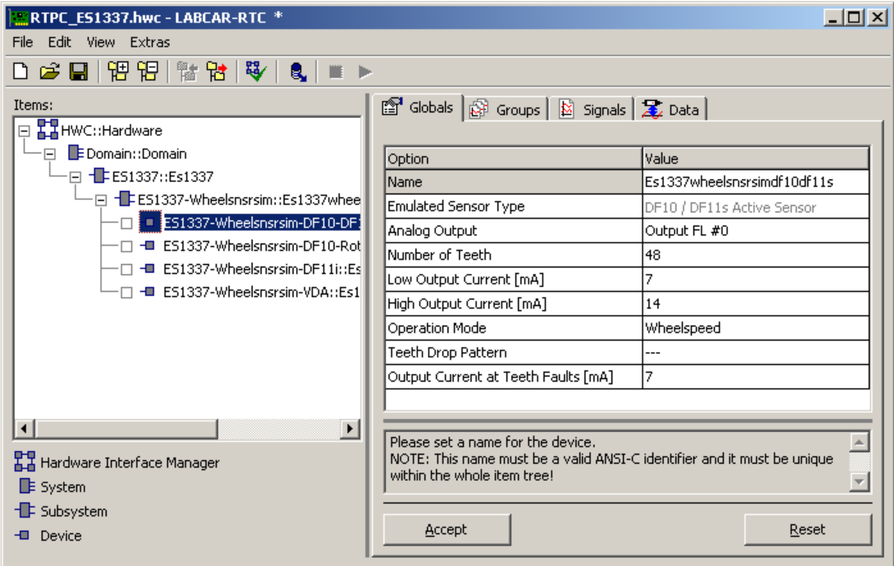


Fig. 12-9 The "Globals" Tab of the ES1337-Wheelsnsrsim-DF10-DF11s Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Analog Output

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High Output Current

With this parameter you can specify the upper current value.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

With this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Analog Output	uint8	No	Output used [0 = FL #0 | 1 = FR #1 |2 = RL #2 | 3 = RR #3]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simulated as defective
Output Current at Teeth Fault	uint8	Yes	Current value when there is a tooth fault [0...35 mA]

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-6 ES1337-Wheelsnsrsim-DF10-DF11s Device: Configuration Parameters of the "Globals" Tab

12.3.6 Groups (ES1337-Wheelsnsrsim-DF10-DF11s Device)

The ES1337-Wheelsnsrsim-DF10-DF11s device has an "Output" signal group with which sensor simulation is controlled.

12.3.7 Data (ES1337-Wheelsnsrsim-DF10-DF11s Device)

The signals for controlling sensor simulation are listed in the "Data" tab.

The following figure shows the signals of the "Data" tab:

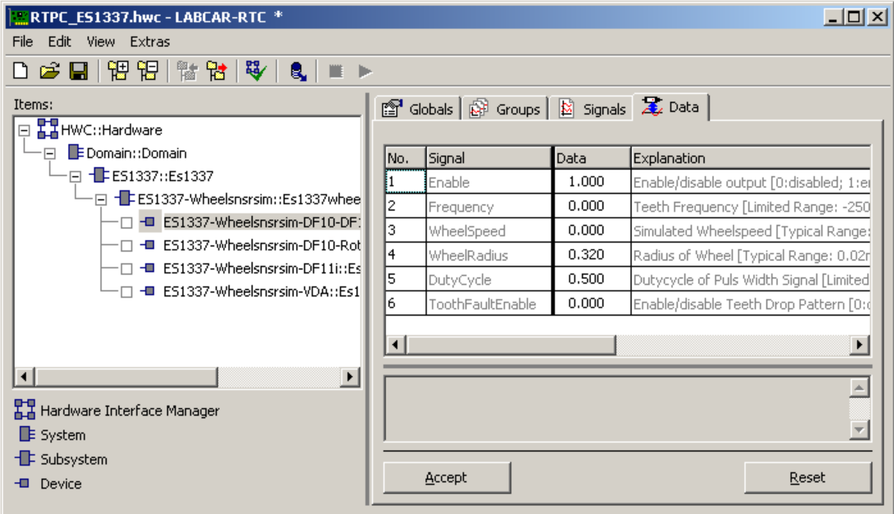


Fig. 12-10 The "Data" Tab of the ES1337-Wheelsnsrsim-DF10-DF11s Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-2500…2500 Hz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02…2 m]
DutyCycle	real32	Yes	Duty cycle [0.05…0.95]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0, 1]

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-7 ES1337-Wheelsnsrsim-DF10-DF11s Device: Signals of the "Data" Tab

Note

The resulting frequency is limited to 2500 Hz regardless of the operation mode – when exceeded, a corresponding error message is generated.

12.3.8 Globals (ES1337-Wheelsnsrsim-DF10-RotDir Device)

With the ES1337-Wheelsnsrsim-DF10-RotDir device you can simulate a "DF10-RotDir" wheel sensor with one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

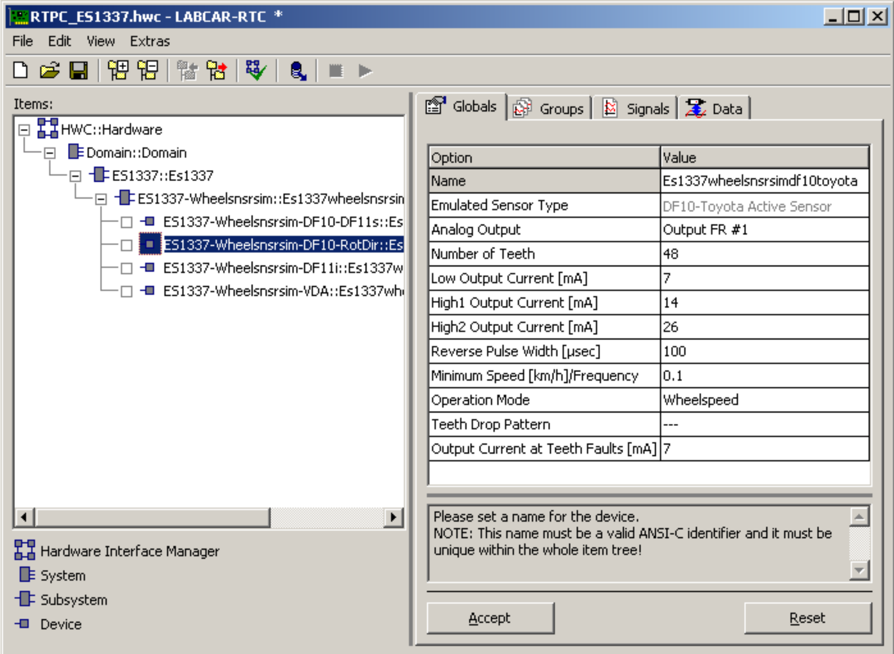


Fig. 12-11 The "Globals" Tab of the ES1337-Wheelsnsrsim-DF10-RotDir Device

Name

You can specify an individual name for this sensor.

Note

The comment field does not permit blanks or special characters as this name is used for variable and parameter names.

Analog Output

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High1 Output Current

With this parameter you can specify the medium current value.

High2 Output Current

With this parameter you can specify the upper current value.

Reverse Pulse Width

With this parameter you can specify the duration of the pulse generated at the pole edges in reverse direction of rotation.

Minimum Speed/Frequency Threshold

With this option you can define the minimum wheel speed (operation mode "Wheelspeed") or frequency (operation mode "Frequency") from which standstill behavior is simulated.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

With this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Analog Output	uint8	No	Output used [0 = FL #0 | 1 = FR #1 |2 = RL #2 | 3 = RR #3]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High1 Output Current	uint8	Yes	Medium current value [0…35 mA], default: 14 mA
High2 Output Current	uint8	Yes	Upper current value [0…35 mA], default: 28 mA
Reverse Pulse Width	uint8	Yes	Duration of reverse pulse [50…150 µA], default: 100 µA
Minimum Speed/ Frequency Thresh-old	real32	Yes	Threshold for standstill behavior [-100.0…100.0 km/h/Hz]
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simu-lated as defective
Output Current at Teeth Fault	uint8	Yes	Current value when there is a tooth fault [0…35 mA], default: 7 mA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 12-8 ES1337-Wheelsnsrsim-DF10-RotDir Device: Configuration Parameters of the "Globals" Tab

12.3.9 Groups (ES1337-Wheelsnsrsim-DF10-RotDir Device)

The ES1337-Wheelsnsrsim-DF10-RotDir device has an "Output" signal group with which sensor simulation is controlled.

12.3.10 Data (ES1337-Wheelsnsrsim-DF10-RotDir Device)

The signals for controlling sensor simulation are listed in the "Data" tab.

The following figure shows the signals of the "Data" tab:

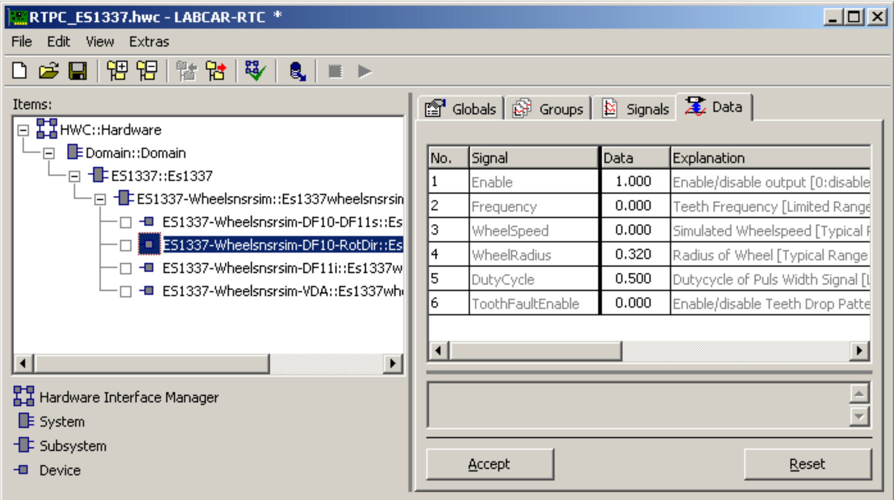


Fig. 12-12 The "Data" Tab of the ES1337-Wheelsnsrsim-DF10-RotDir Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-1600…2500 Hz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02…2 m]
DutyCycle	real32	Yes	Duty cycle [0.25…0.75]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0, 1]

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-9 ES1337-Wheelsnsrsim-DF10-RotDir Device: Signals of the "Data" Tab

Note

The resulting frequency is limited to the range -1600...2500 Hz regardless of the operation mode – when exceeded, a corresponding error message is generated.

12.3.11 Globals (ES1337-Wheelsnsrsim-DF11i Device)

With the ES1337-Wheelsnsrsim-DF11i device you can simulate a "DF11i" wheel sensor with one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

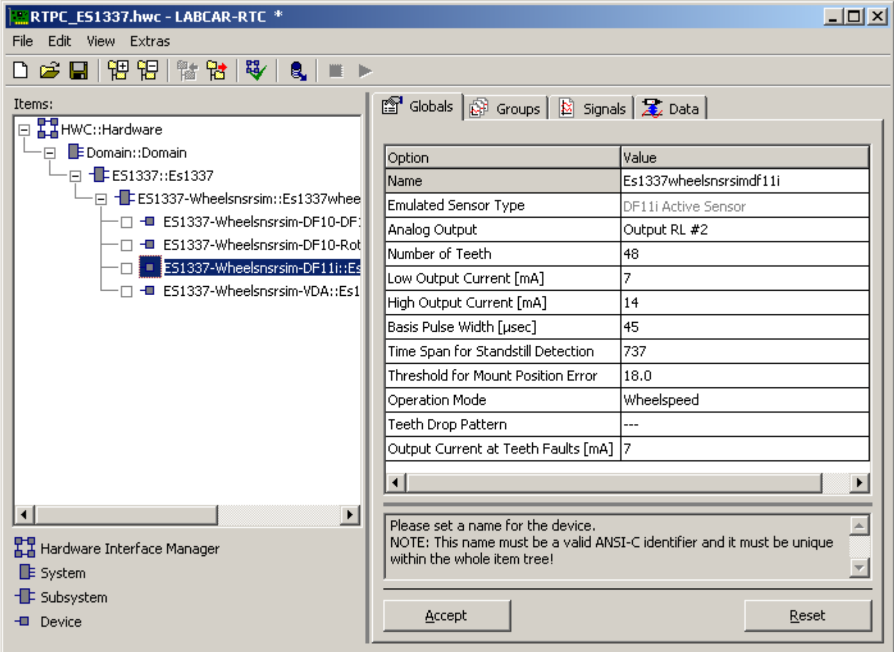


Fig. 12-13 The "Globals" Tab of the ES1337-Wheelsnsrsim-DF11i Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Analog Output

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High Output Current

With this parameter you can specify the upper current value.

Basic Pulse Width

With this parameter you can specify the basic pulse width. Other pulse widths are derived from this value, e.g.

- Air gap limit
- Direction of rotation left
- Direction of rotation right
- Direction of rotation left and mount position
- Direction of rotation right and mount position

Timespan for Standstill Detection

With this parameter you can define the time after which the fact that the wheel is not moving is detected and the standstill pulse is output. Standstill monitoring is reset with a tooth pulse or standstill pulse, i.e. on standstill, standstill pulses are output in these intervals.

Threshold for Mount Position Error

With this option, you can define the threshold after which a position error should be output via wheel speed (operation mode "Wheelspeed") or frequency (operation mode "Frequency"). If the wheel speed or frequency is below this value, the error signal is set.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

In this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Analog Output	uint8	No	Output used [0 = FL #0 | 1 = FR #1 |2 = RL #2 | 3 = RR #3]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Basic Pulse Width	uint8	Yes	Basic pulse width [37...53 µs], default: 45 µs
Minimum Speed/ Frequency Thresh-old	real32	Yes	Threshold for standstill behavior [-100.0…100.0 km/h/Hz]
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simu-lated as defective
Output Current at Teeth Fault	uint8	Yes	Current value when there is a tooth fault [0…35 mA], default: 7 mA

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-10 ES1337-Wheelsnsrsim-DF11i Device: Configuration Parameters of the "Globals" Tab

12.3.12 Groups (ES1337-Wheelsnsrsim-DF11i Device)

The ES1337-Wheelsnsrsim-DF11i device has an "Output" signal group with which sensor simulation is controlled.

12.3.13 Data (ES1337-Wheelsnsrsim-DF11i Device)

The signals for controlling sensor simulation are listed in the "Data" tab.

The following figure shows the signals of the "Data" tab:

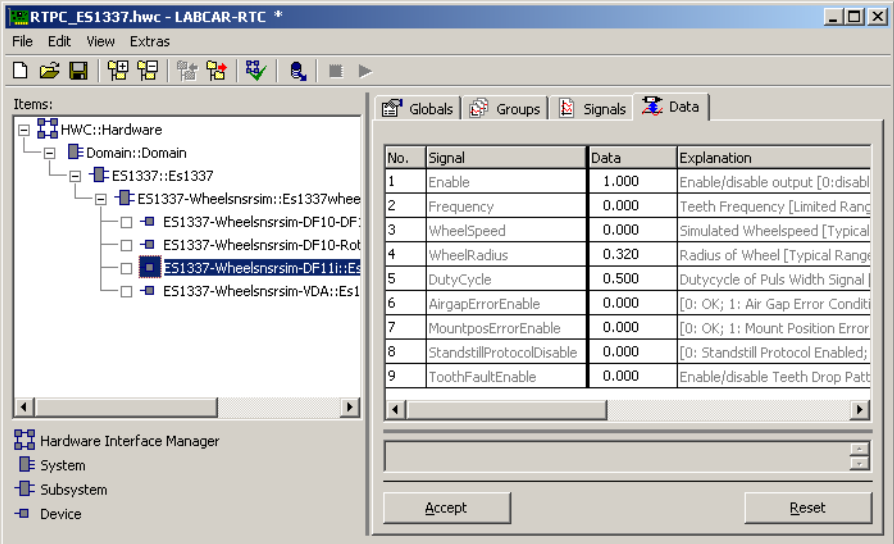


Fig. 12-14 The "Data" Tab of the ES1337-Wheelsnsrsim-DF11i Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

AirgapErrorEnable

Use this signal to enable/disable the "air gap error" condition.

MountposErrorEnable

Use this signal to enable/disable the "mount position error" condition.

StandstillProtocolDisable

Use this signal to enable/disable the behavior defined for this sensor on standstill.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-10000…10000 Hz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02...2 m]
DutyCycle	real32	Yes	Duty cycle [0.05…0.95]
AirgapErrorEnable	real32	Yes	Output "Air gap error" condition [0.0 = off, 1.0 = active]
MountposError-Enable	real32	Yes	Activation of position error [0.0 = off, 1.0 = active]
Standstill-ProtocolDisable	real32	Yes	Activation of standstill protocol [0.0 = off, 1.0 = active]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0 = off, 1 = active]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 12-11 ES1337-Wheelsnsrsim-DF11i Device: Signals of the "Data" Tab

Note

The resulting frequency is limited to 2500 Hz regardless of the operation mode – when exceeded, a corresponding error message is generated.

12.3.14 Globals (ES1337-Wheelsnsrsim-VDA Device)

With the ES1337-Wheelsnsrsim-VDA device you can simulate a "VDA" wheel sensor with one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

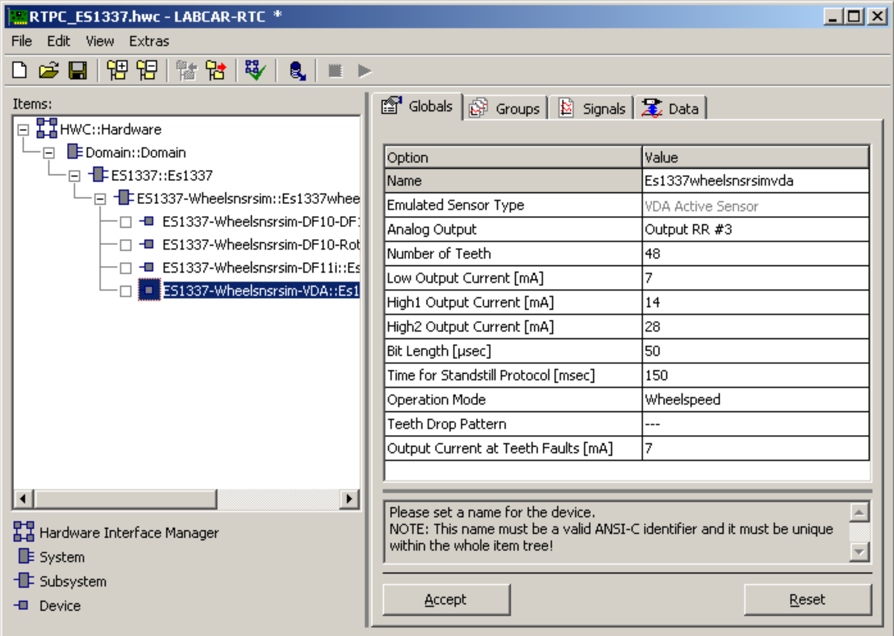


Fig. 12-15 The "Globals" Tab of the ES1337-Wheelsnsrsim-VDA Device

Name

A unique name for this device.

> **Note**
>
> *As this name is used for variable and parameter names, blanks and special characters are not permissible!*

Analog Output

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High1 Output Current

With this parameter you can specify the medium current value.

High2 Output Current

With this parameter you can specify the upper current value.

Bit Length

With this parameter you can specify the basic pulse width. Other pulse widths are derived from this value, for example

- Speed pulse
- Data pulse
- Initial bit

Time for Standstill Protocol

With this parameter you can define the time after which the fact that the wheel is not moving is detected and the standstill pulse is output. Standstill monitoring is reset with a tooth pulse or standstill pulse, i.e. on standstill, standstill pulses are output in these intervals.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

In this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Analog Output	uint8	No	Output used [0 = FL #0 | 1 = FR #1 |2 = RL #2 | 3 = RR #3]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Bit Length	uint8	Yes	Basic pulse width [20…80 µs], default: 50 µs
Time for Standstill Protocol	uint8	Yes	Threshold for standstill behavior [-100.0…100.0 km/h/Hz]
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1= Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simulated as defective
Output Current at Teeth Faults	uint8	Yes	Current value when there is a tooth fault [0…35 mA], default: 7mA

* Data type which the RTIO driver uses internally for the parameter

Tab. 12-12 ES1337-Wheelsnsrsim-VDA Device: Configuration Parameters of the "Globals" Tab

12.3.15 Groups (ES1337-Wheelsnsrsim-VDA Device)

The ES1337-Wheelsnsrsim-VDA device has an "Output" signal group with which sensor simulation is controlled.

12.3.16 Data (ES1337-Wheelsnsrsim-VDA Device)

The signals for controlling sensor simulation are listed in the "Data" tab.

The following figure shows the signals of the "Data" tab:

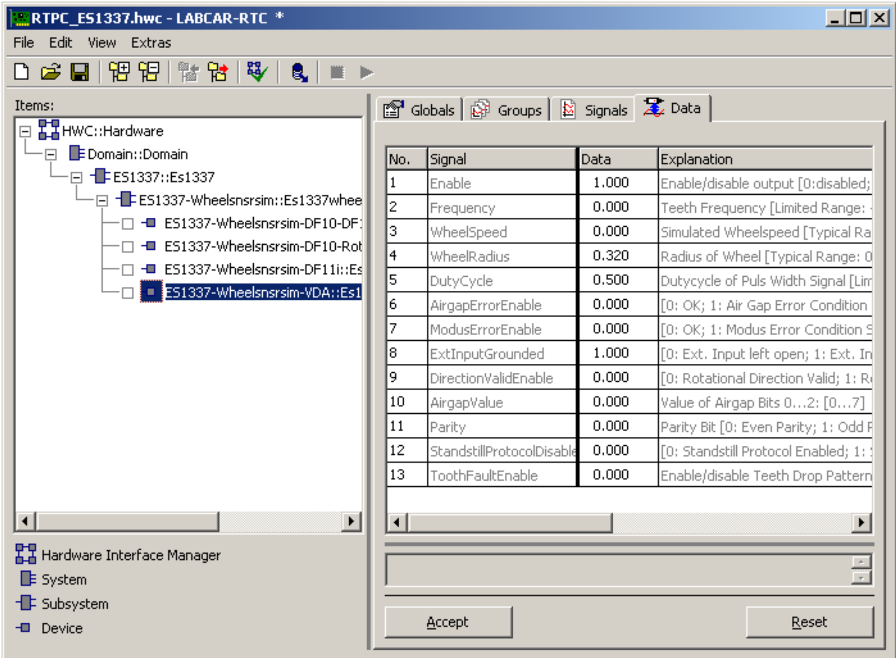


Fig. 12-16 The "Data" Tab of the ES1337-Wheelsnsrsim-VDA Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

AirgapErrorEnable

Use this signal to enable/disable the "AirgapError" condition (protocol bit 0).

ModusErrorEnable

Use this signal to enable/disable the "ModusError" condition (protocol bit 1).

ExtInputGrounded

Use this signal to influence the corresponding protocol bit 2. A zero indicates "External input open".

DirectionValidEnable

Use this signal to influence the corresponding protocol bit 3. A zero indicates that the direction of rotation transferred is valid.

AirgapValue

Use this signal to influence the value of the corresponding protocol bits 5..7 . The air gap value is in a range between 0 and 7.

Parity

Use this signal to influence the parity creation (protocol bit 8). A value of zero corresponds to even parity.

The meaning of the protocol bits is summarized in the following table.

Bit	Meaning	Coding
0	Error bit air gap limit	0 = correct 1 = air gap limit
1	Can be assigned freely	
2	Can be assigned freely	
3	Validity of direction of rotation	0 = valid 1 = invalid
4	Direction of rotation – is calculated from the sign of "WheelSpeed" or "Frequency"	0 = positive 1 = negative
5	Can be assigned freely	
6	Can be assigned freely	
7	Can be assigned freely	
8	Parity bit	Is set to 0 or 1 to retain even parity (incl. the parity bit itself).

Tab. 12-13 Meaning of the Nine Protocol Bits

StandstillProtocolDisable

Use this signal to influence the standstill protocol behavior of the sensor.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-5000…5000 Hz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02...2 m]
DutyCycle	real32	Yes	Duty cycle [0.05…0.95]
AirgapErrorEnable	real32	Yes	Activation of "air gap error" [0.0 = off, 1.0 = active]
ModusErrorEnable	real32	Yes	Activation of "modus error" [0.0 = off, 1.0 = active]
ExtInputGrounded	real32	Yes	Level of ext. input [0.0 = open, 1.0 = ground]
DirectionValidEnable	real32	Yes	Indicates direction of rotation valid [0.0 = direction of rota-tion valid, 1.0 = direction of rotation invalid]
AirgapValue	real32	Yes	Air gap value [0…7]
Parity	real32	Yes	Setting parity [0.0 = even, 1.0 = odd]
Standstill-ProtocolDisable	real32	Yes	Activation of standstill protocol [0.0 = active, 1.0 = inactive]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0 = off, 1 = active]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 12-14 ES1337-Wheelsnsrsim-VDA Device: Signals of the "Data" Tab

Note

The resulting frequency is limited to 5000 Hz regardless of the operation mode – when exceeded, a corresponding error message is generated.

13 ES1385 Resistor Cascade Board

The ES1385 Resistor Cascade Board is used in the LABCAR environment as a resistor board to simulate both temperature and oxygen sensors.

Note

Concerning the specific values of the resistors, several variants of the ES1385 Resistor Cascade Board are available. For details on the specifications of the resistor cascades refer to the users guide of your hardware.

Structure of the ES1385 RTIO Tree

An ES1385 Resistor Cascade Board is integrated in the VMEbus chassis and is connected directly to the experimental target via the VMEbus. The hardware connection described is reflected in the RTIO tree

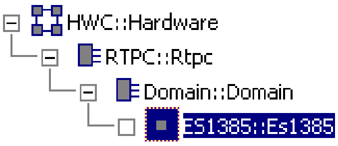


Fig. 13-1 RTIO Hardware Description with Integrated ES1385

13.1 ES1385 Device

13.1.1 Globals (ES1385 Device)

The "Globals" tab of an ES1385 device is shown in Fig. 13-2.

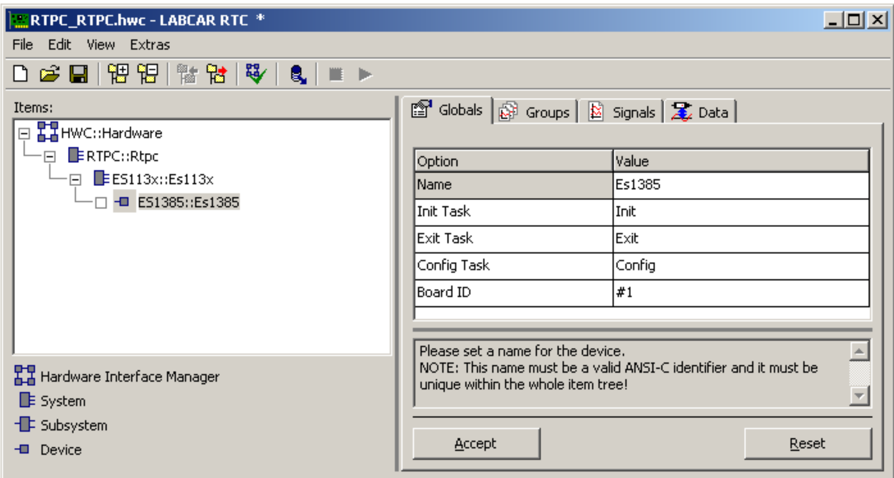


Fig. 13-2 The "Globals" Tab of the ES1385 Device

Note

When the ES1385 Resistor Cascade Board is integrated into a LABCAR-RTC hardware configuration, both address switches (SW1 and SW2) on the board must be set to "0x00"!

Board ID

This option field is used to identify the ES1385 Resistor Cascade Board. It establishes the assignment between the RTIO hardware description and a specific ES1385 in the VMEbus chassis. The ES1385 boards in the chassis are numbered from left to right (starting with 1). Up to 20 ES1385 boards can be integrated per chassis.

This RTIO parameter cannot be set during runtime.

Tab. 13-1 summarizes the properties of the "Board ID" parameter.

Parameter or Option Field	Data Type*	Can be Edited in Runtime Mode	Comment/Value Range
Board ID	uint32	No	Identification of the ES1385 board. You can choose from #1 to #20.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 13-1 ES1385 Device: Configuration Parameter of the "Globals" Tab

Hidden Option Fields

Further option fields (see Fig. 13-3 on page 340) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting "Show all Options".

These option fields make it possible for ETAS service personnel to gain additional information on the ES1385 Resistor Cascade Board when localizing the cause of an error. They are not intended for the user - the default settings should therefore not be changed by the user.

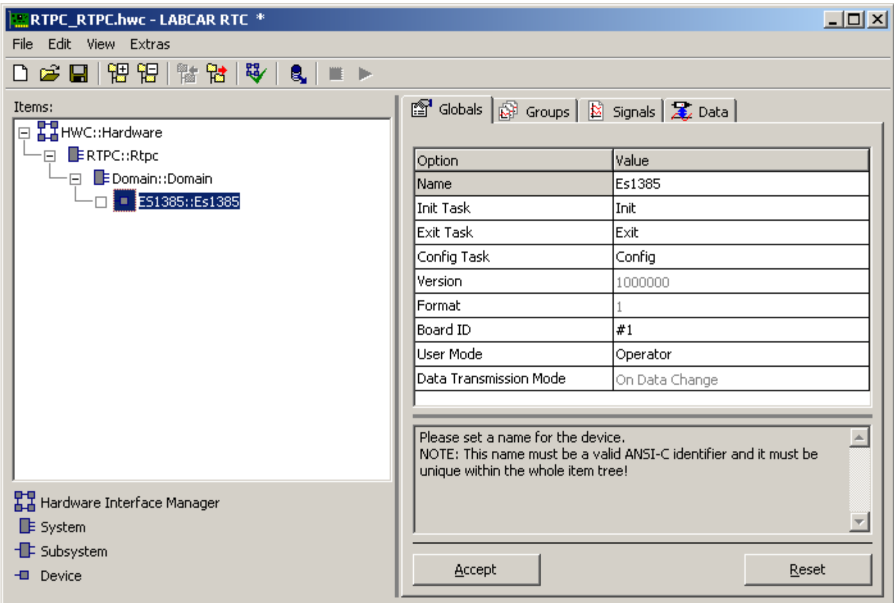


Fig. 13-3 The "Globals" Tab of the ES1385 Device with all Option Fields

User Mode

RTIO user mode is set with the parameter "User Mode". Users should select "Operator" from the list field.

"Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. In "Supervisor" mode, the data of the non-volatile memory (such as the calibration values of the cascades) is read out in the initialization phase of the RTIO driver and made visible to the user in LABCAR-OPERATOR in the "Hardware Info" window.

The list field can be edited during runtime and is set to "Operator" by default.

Data Transmission Mode

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies when real-time data is transmitted from the experimental target to the ES1385 Resistor Cascade Board.

If the "On Data Change" option is activated, data is only transmitted if there has been a change. If the "Every Interval" option is activated, the real-time data is transmitted at every interval of the allocated real-time task.

The list field can be edited during runtime and is set to "On Data Change" by default.

Tab. 13-2 summarizes the properties of the hidden RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited in Runtime Mode	Comment/Value Range
User Mode	uint8	Yes	User mode 0: Operator 1: Supervisor "Operator" is preset. "Supervisor" activates additional options for localizing errors.
Data Transmission Mode	uint8	Yes	Transfer mode of the real-time data 0: On Data Change 1: Every Interval "On Data Change" is preset. The option field can only be edited if "User Mode" = "Supervisor".

* Data type which the RTIO driver uses internally for the parameter

Tab. 13-2 ES1385 Device: Hidden Configuration Parameters of the "Globals" Tab

13.1.2 Groups (ES1385 Device)

The ES1385 device has three signal groups which are transmitted from the experimental target to the ES1385 Resistor Cascade Board or from the ES1385 to the experimental target (see Fig. 13-4 on page 342).

- The **signal group "CasCtrl"** transports the resistance values of the cascades specified by the user and prepared by the RTIO to the ES1385. The resistance values specified by the user are accumulated to obtain the resistance value actually possible by taking calibration and monotony into consideration using an approximation method. Then it is transmitted to the hardware.
- The **signal group "CasMntr"** displays the real resistance values created by the RTIO low-level driver for the user. The activation period of this signal group should correspond to the activation period of the signal group "CasCtrl".
- The **signal group "CasStat"** transports current status information of the independent resistor cascades to the RTIO. This means that the individual error states of the hardware can be displayed/monitored via the RTIO.

One task of the real-time operating system must be assigned to each signal group. A task with periodic activation is usually selected. The activation period depends on the dynamic behavior of the resistance values to be generated or the status information to be monitored.

Fig. 13-4 shows the "Groups" tab of an ES1385 device.

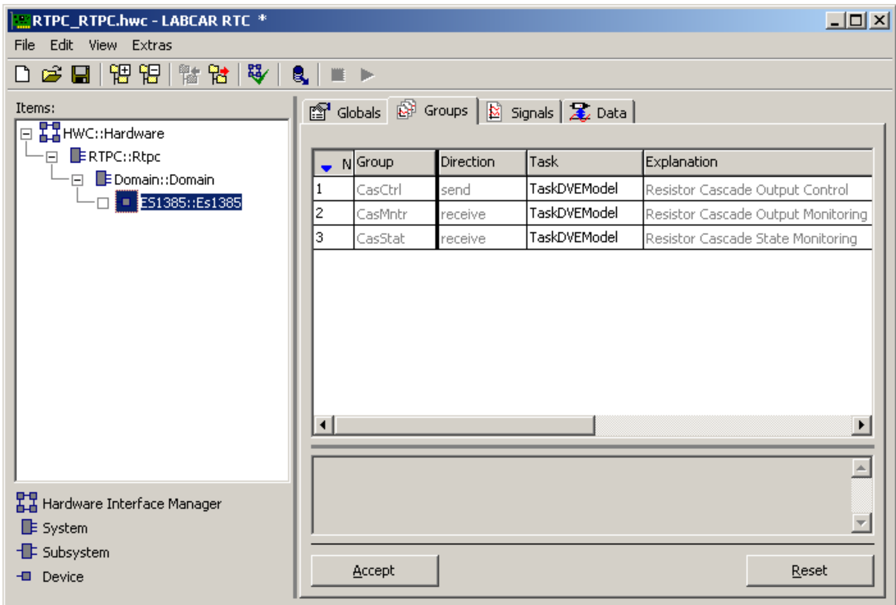


Fig. 13-4 The "Groups" Tab of the ES1385 Device

RTIO Signals of the "CasCtrl" Signal Group

The "CasCtrl" signal group has 6 RTIO signals, "Ctrl_R_0" to "Ctrl_R_5", which are of the data type "uint32". If the resistor cascade x (x = 0, 1, 2, 3, 4, 5) is configured so that it is controlled by the RTIO, the value of the RTIO signal "Ctrl_R_x" defines the target resistance value of the corresponding cascade specified by the user in Ohm.

If the resistor cascade is not controlled by the RTIO, the signal is irrelevant.

RTIO Signal	Data Type*	Comment/Value Range
Ctrl_R_0 to Ctrl_R_5	uint32	Resistance value defined by the user
* Data type which the RTIO driver uses internally for the parameter		

Tab. 13-3 ES1385 Device: The RTIO Signals of the "CasCtrl" Signal Group

RTIO Signals of the "CasMntr" Signal Group

The "CasMntr" signal group has 6 RTIO signals, "Mntr_R_0" to "Mntr_R_5", which are of the data type "real32". It is used to display the actual resistance value of a cascade transmitted to the hardware. Every resistance value defined by the user must be created using an approximation method taking the calibration values and monotony into consideration.

Tab. 13-4 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Mntr_R_0 to Mntr_R_5	real32	The actual resistance value transmitted to the hardware
* Data type which the RTIO driver uses internally for the parameter		

Tab. 13-4 ES1385 Device: The RTIO Signals of the "CasMntr" Signal Group

RTIO Signals of the "CasStat" Signal Group

The "CasStat" signal group has 6 RTIO signals, "Stat_C_0" to "Stat_C_5", which are of the data type "uint16". It is used to display the current status of the hardware channels.

Tab. 13-5 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Stat_C_0 to Stat_C_5	uint16	Status of the resistor cascade 0: no error 1: overcurrent was detected
* Data type which the RTIO driver uses internally for the parameter		

Tab. 13-5 ES1385 Device: The RTIO Signals of the "CasStat" Signal Group

13.1.3 Signals (ES1385 Device)

In the "Signals" tab, the resistor cascades of an ES1385 board are configured.

Fig. 13-5 shows the RTIO parameters of the "Signals" tab. All parameters can be edited during runtime of the RTIO driver on the experimental target.

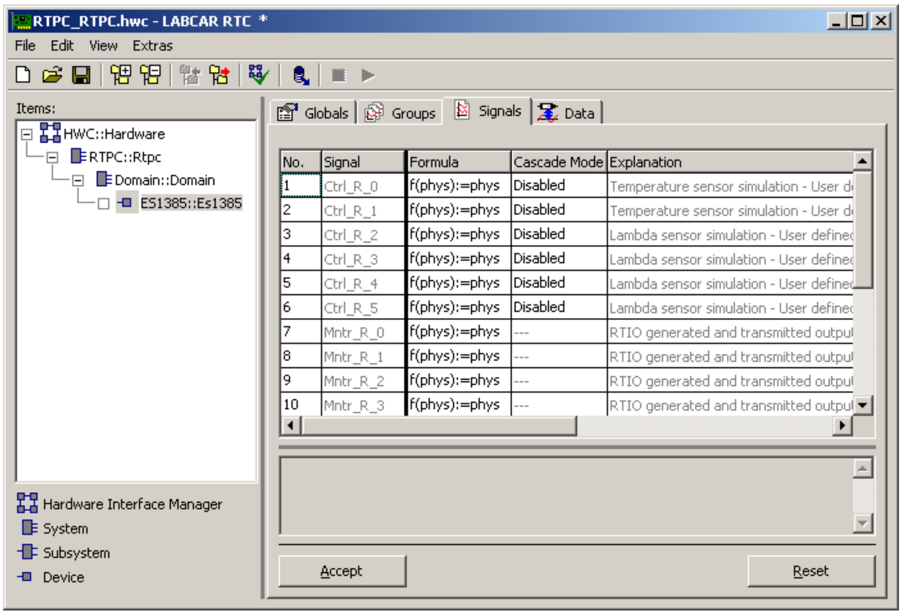


Fig. 13-5 The "Signals" Tab of the ES1385 Device

Cascade Mode

This option field is used to configure the resistor cascade. "Disabled" deactivates the cascade x (x = 0, 1, 2, 3, 4, 5), i.e. all relays are open. No resistance values can be entered and transmitted to the hardware in this state.

"Enabled" activates the cascade x (x = 0, 1, 2, 3, 4, 5), i.e. resistance values can only be transported to the hardware via the RTIO in this state.

Tab. 13-6 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited in Runtime Mode	Comment/Value Range
Cascade Mode	uint8	Yes	Configures the resistor cascades: 0: "Disabled", cascade is deactivated 1: "Enabled", cascade is activated
* Data type which the RTIO driver uses internally for the parameter			

Tab. 13-6 ES1385 Device: Configuration Parameter of the "Signals" Tab

14 ES1650.1 Piggyback Carrier Board

The ES1650.1 Piggyback Carrier Board is used in VMEbus systems as a carrier board for piggybacks. The ES1650.1 has two slots for piggybacks. There are piggyback modules for different tasks, e.g. digital and analog input and output modules as well as relay modules with switch functions.

This document describes the RTIO integration of the ES1650.1 as well as the RTIO integrations of the piggyback modules for this carrier board:

- PB1650ADC1 (on page 346)
- PB1650DAC1 (on page 349)
- PB1650DIO1 (on page 351)
- PB1650DIO2 (on page 353)
- PB1650REL1 (on page 355)

14.1 ES1650-CB Subsystem

14.1.1 Globals (ES1650-CB Subsystem)

This section describes the global options of the ES1650-CB subsystem.

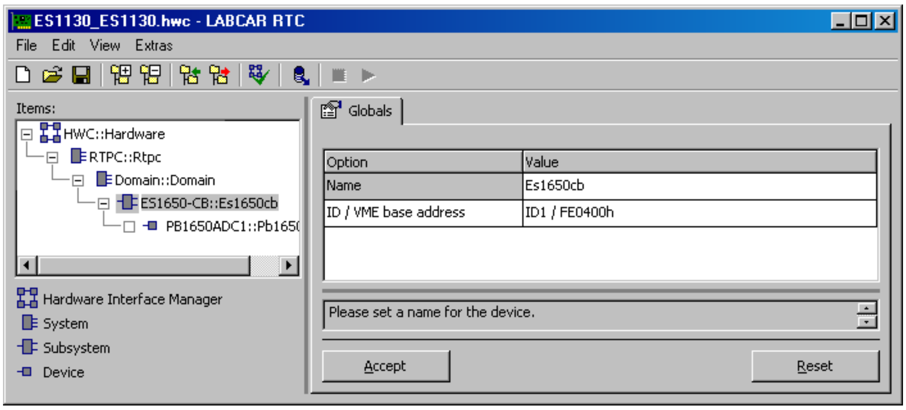


Fig. 14-1 The "Globals" Tab of the ES1650-CB Subsystem

ID / VME base address

The VMEbus base address of the ES1650.1 Piggyback Carrier Board used is specified in the "ID / VME base address" field. There are 16 different base addresses to choose from (`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` and `ID16/FE7C00h`). The size of the address space of the ES1650.1 can be configured with jumpers to 256 bytes or 8 kBytes. All piggybacks incorporated in RTIO have an address space of 256 bytes. The RTIO integration of the ES1650.1 does not support piggybacks with an 8-kByte address space.

14.2 PB1650ADC1

The PB1650ADC1 piggyback is used for analog data measurement in connection with the ES1650.1 Piggyback Carrier Board. The module is designed for medium-sized resolutions and small to medium data rates. It includes the following functions:

- A/D conversion with 12-bit resolution
- eight unipolar or bipolar input channels, which can be programmed by software
- four input voltage ranges

The PB1650ADC1 piggyback can be used anywhere in VMEbus systems where analog input signals are to be measured. Examples of areas of application are:

- measurement of analog transducer signals, e.g. engine temperature, oil temperature, accelerator setting
- measurement of analog output values of the ECU
- simulation of analog sensors which receive a control or reference voltage from the ECU

14.2.1 Globals (PB1650ADC1 Device)

This section describes the global options of the PB1650ADC1 device.

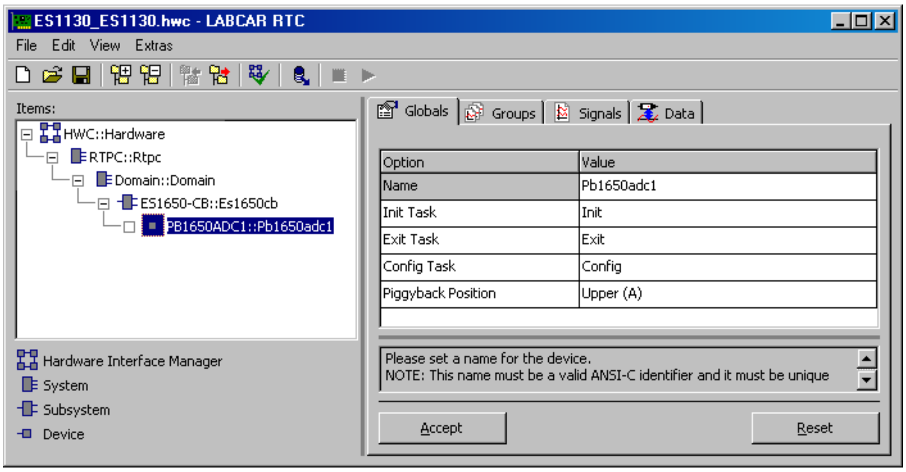


Fig. 14-2 The "Globals" Tab of the PB1650ADC1 Device

Piggyback Position

The "Piggyback Position" field specifies whether the piggyback is in the upper or lower slot of the ES1650.1.

14.2.2 Groups (PB1650ADC1 Device)

The available options in the "Groups" tab are described in section "Default Options in the "Groups" Tab" on page 45.

14.2.3 Signals (PB1650ADC1 Device)

This section describes the signal-specific options of the PB1650ADC1 device.

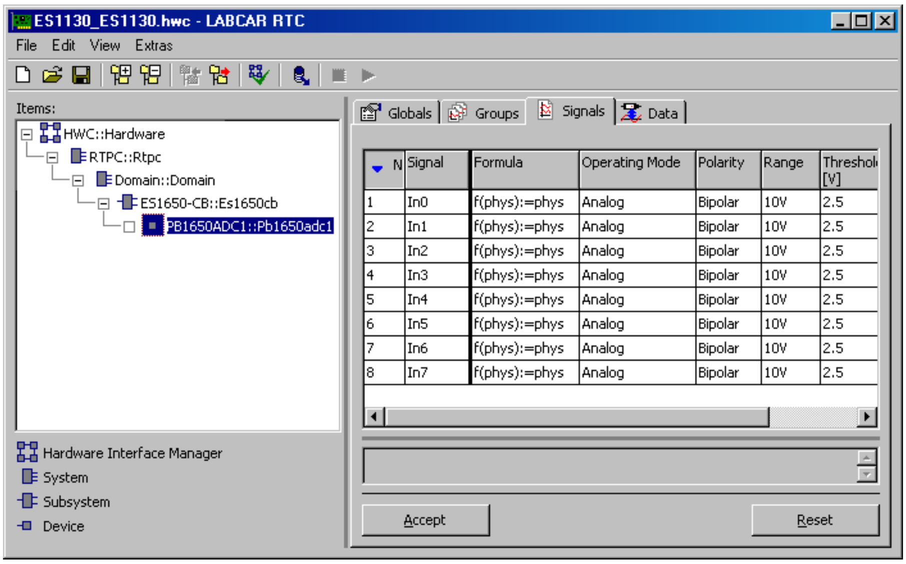


Fig. 14-3 The "Signals" Tab of the PB1650ADC1 Device

Operating Mode

Each of the 8 hardware channels of a PB1650ADC1 piggyback can be operated in four different modes which are set separately for each channel in the "Operating Mode" field (see Fig. 14-3).

- Disabled: the channel is deactivated.
- Analog: the channel is configured for analog measurement acquisition.

  The relevant signal *Inx* (x = 0, 1, ... 7) specifies the measured voltage in volts.
- Comparator: the channel is configured as a comparator.

  The relevant signal value *Inx* is 0 if the input voltage of the channel is smaller than a voltage threshold which can be set by the user; otherwise the signal value is 1.
- Comparator (abs): the channel is configured as an absolute comparator.

  The relevant signal value *Inx* is 0 if the amount of input voltage of the channel is smaller than a threshold voltage which can be set by the user; otherwise the signal value is 1.

Polarity/Range

The "Polarity" field is used to set the voltage polarity of a measure channel and the "Range" field is used to specify the voltage range.

Unlike polarity, the voltage range of a channel cannot be configured by the software, but is set using solder straps. A total of 4 measuring ranges per channel can be set using the two fields, "Polarity" and "Range": (0 V ... 5.0 V; -5.0 V ... 5.0 V; 0 V ... 10.0 V; -10.0 V ... 10.0 V).

The settings in the "Polarity" and "Range" fields are relevant for all operating modes apart from "Disabled".

Threshold [V]

The setting in the "Threshold [V]" field is only of any significance for the two comparator modes. It specifies the threshold voltage for the comparator.

AD Average Samples

This determines how many sample values are used to determine an analog voltage value through averaging. Averaging can result in interference or noise suppression, but please note that approx. 43 µs per channel are required for sampling.

The setting in the "AD Average Samples" field is of significance for all operating modes apart from "Disabled".

14.2.4 Data Types and Value Ranges of the Signals

The 8 *Inx* (x = 0, 1, 2, ... 7) signals of the PB1650ADC1 piggyback are either of data type "float" or "real32". A channel in "Disabled" mode returns the value 0.0. A channel in "Analog" mode returns the measured voltage in volts. A channel in "Comparator" or "Comparator (abs)" mode returns either 0.0 or 1.0 (cf. section 14.2.3 on page 347).

Note

When data types are specified in this document, they are the data types used internally by the RTIO driver.

14.3 PB1650DAC1

The PB1650DAC1 piggyback is used to generate analog output signals in connection with the ES1650.1 Piggyback Carrier Board. The module includes the following functions:

- D/A conversion with 12-bit resolution
- four unipolar or bipolar output channels (output voltage range: 0.0 V ... 10.0 V or -10.0 V ... 10.0 V)
- analog output channels dc decoupled from the VMEbus system

14.3.1 Globals (PB1650DAC1 Device)

This section describes the global options of the PB1650DAC1 device.

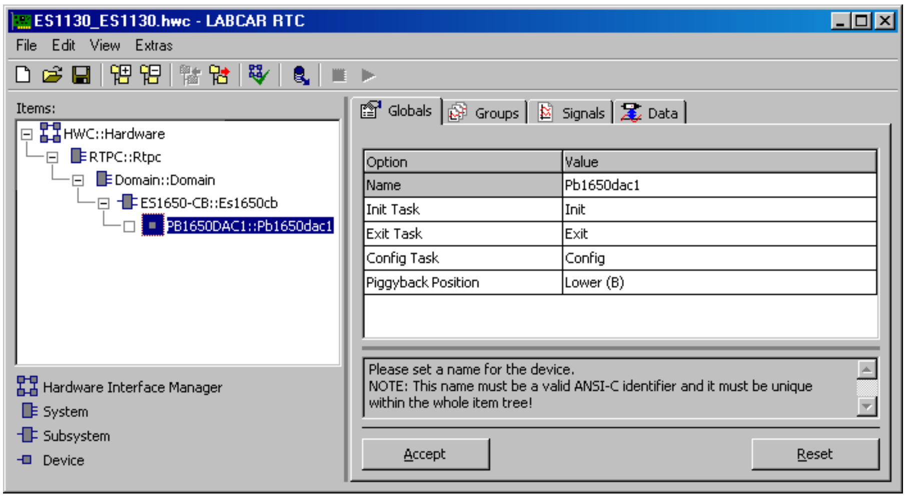


Fig. 14-4 The "Globals" Tab of the PB1650DAC1 Device

Piggyback Position

The "Piggyback Position" field specifies whether the piggyback is in the upper or lower slot of the ES1650.1 .

14.3.2 Groups (PB1650DAC1 Device)

The available options in the "Groups" tab are described in section "Default Options in the "Groups" Tab" on page 45.

14.3.3 Signals (PB1650DAC1 Device)

This section describes the signal-specific options of the PB1650DAC1 device.

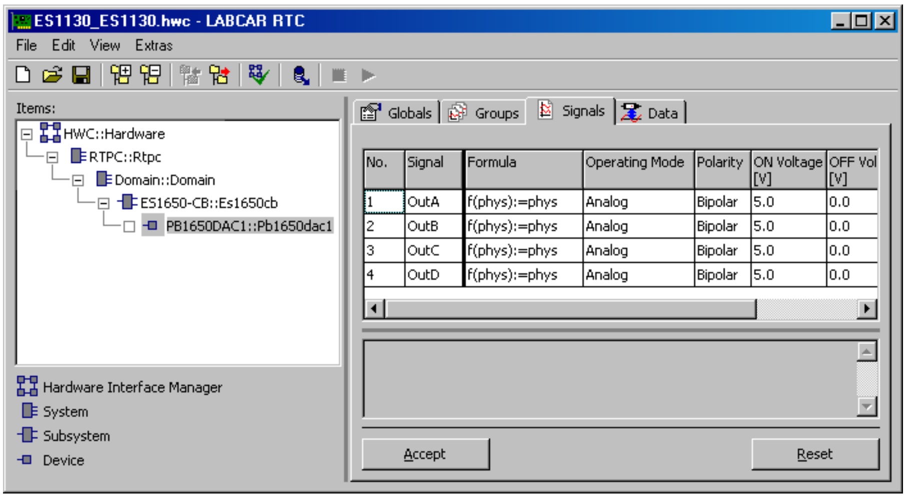


Fig. 14-5 The "Signals" Tab of the PB1650DAC1 Device

Operating Mode

Each of the 4 hardware channels of a PB1650DAC1 piggyback can be operated in four different modes which are set separately for each channel in the "Operating Mode" field (see Fig. 14-5).

- Disabled: the channel is deactivated.
- Analog: the channel is configured for analog voltage output.

  The relevant signal *Outx* (x = A, B, C, D) specifies the voltage to be output in volts.
- Comparator: the channel is configured as a comparator.

  If the *Outx* signal value is smaller than a "Comparator Threshold" set by the user, the voltage set in the "OFF Voltage [V]" field is output at the output; otherwise the voltage set in the "ON Voltage [V]" field is output.
- Comparator (abs): the channel is configured as an absolute comparator.

  If the *Outx* signal value is smaller than a "Comparator Threshold" set by the user, the voltage set in the "OFF Voltage [V]" field is output at the output; otherwise the voltage set in the "ON Voltage [V]" field is output.

Polarity

The voltage polarity of a measure channel is specified In the "Polarity" field. Please note that this voltage polarity cannot be configured using the software but is set using solder straps.

The output voltage range of a measure channel is (depending on the setting in the "Polarity" field) either 0.0 V...10.0 V or -10.0 V...10.0 V.

ON Voltage [V], OFF Voltage [V] and Comparator Threshold

The "Comparator Threshold", "OFF Voltage [V]" and "ON Voltage [V]" fields are only relevant in the two Comparator modes. The permissible values for "OFF Voltage [V]" and "ON Voltage [V]" depend on the selected voltage range.

The admissible value range for "Comparator Threshold" is -1.0 ... 1.0 in channel mode Comparator and 0.0 ... 1.0 in channel mode Comparator (abs).

14.3.4 Data Types and Value Ranges of the Signals

The 4 *Outx* (x = A, B, C, D) signals of the PB1650DAC1 piggyback are of data type "float" or "real32". The *Outx* value of a channel in "Disabled" mode is ignored. The *Outx* value of a channel in "Analog" mode specifies the voltage to be output in volts. If the *Outx* value is outside the permissible range, it is limited to the nearest admissible value. The *Outx* value of a channel in "Comparator" or "Comparator (abs)" mode must be between -1.0 and 1.0. An *Outx* value outside the permissible range is limited to the nearest admissible value.

14.4 PB1650DIO1

The PB1650DIO1 piggyback is used for the digital input and output of switch states via eight separate dc decoupled inputs and outputs. The module also provides the following functions:

- eight parallel, dc decoupled digital input channels (up to 80 V direct voltage)
- eight parallel, dc decoupled digital output channels (up to 500 mA)

The PB1650DIO1 piggyback is used in connection with the ES1650.1 Piggyback Carrier Board to measure and generate binary switch signals. Examples of areas of implementation are:

- acquisition of switch outputs of the ECU (addressing solenoid valves, reverse-light relay)
- simulation of a switch (handbrake switch)

14.4.1 Globals (PB1650DIO1 Device)

This section describes the global options of the PB1650DIO1 device.

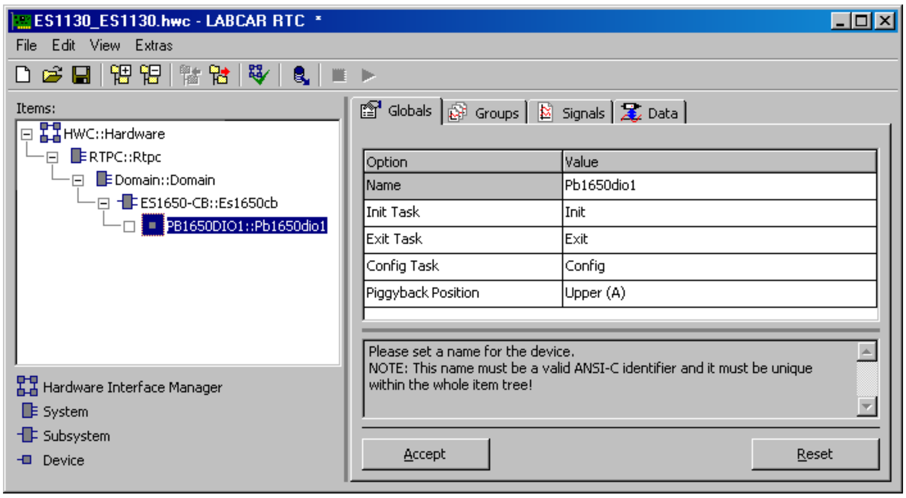


Fig. 14-6 The "Globals" Tab of the PB1650DIO1 Device

Piggyback Position

The "Piggyback Position" field specifies whether the piggyback is in the upper or lower slot of the ES1650.1.

14.4.2 Groups (PB1650DIO1 Device)

The available options in the "Groups" tab are described in section "Default Options in the "Groups" Tab" on page 45.

14.4.3 Signals (PB1650DIO1 Device)

This section describes the signal-specific options of the PB1650DIO1 device.

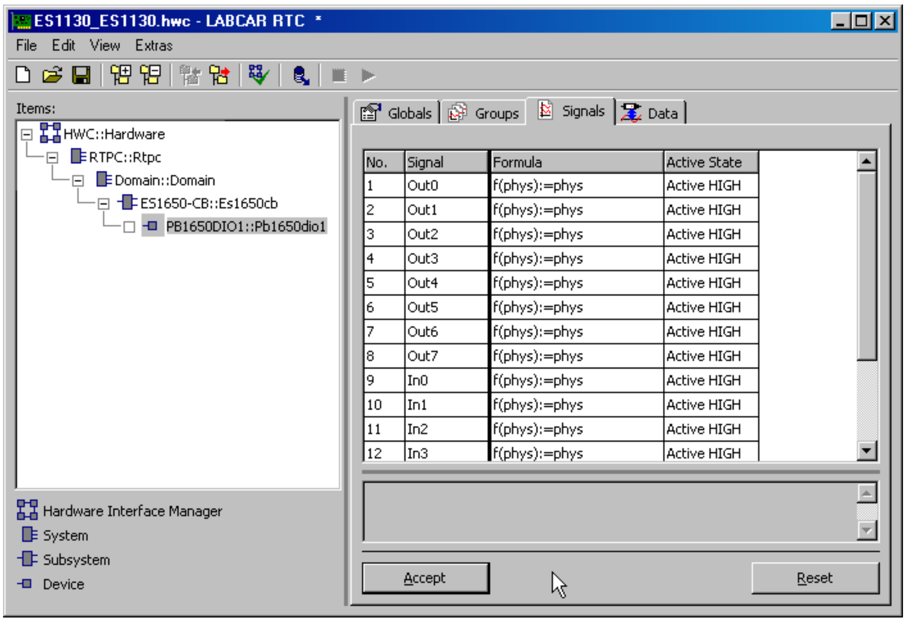


Fig. 14-7 The "Signals" Tab of the PB1650DIO1 Device

Active State

In the "Active State" field, you can specify whether the relevant signal is Active LOW or Active HIGH for each input and output of a PB1650DIO1 piggyback.

With an *Inx* (x = 0, 1 ... 7) input, 0 means that the generated signal is inactive; 1 means that the generated signal is active.

With an *Outx* (x = 0, 1, ... 7) output, 0 means that the relevant signal is switched as inactive; 1 means that the relevant signals is switched as active.

14.4.4 Data Types and Value Ranges of the Signals

The 8 *Outx* (x = 0, 1, ... 7) signals of the PB1650DIO1 piggyback are of data type `uint8`. The signal is interpreted by the RTIO driver as a BOOLEAN value. The value 0 corresponds to `FALSE`; a value not equal to 0 corresponds to `TRUE`.

The 8 *Inx* (x = 0, 1, ... 7) signals are of data type "uint8". The returned signal value is either 0 or 1.

14.5 PB1650DIO2

The PB1650DIO2 piggyback is used for the digital input and output of switch states via eight dc decoupled inputs and outputs. The module provides the following functions:

- eight parallel, dc decoupled, digital input channels
- eight parallel, dc decoupled, digital output channels
- four parallel, dc decoupled control lines

The PB1650DIO2 piggyback is used in conjunction with the ES1650.1 Piggyback Carrier Board to measure and generate binary switch signals with TTL signal levels. Examples of areas of implementation are:

- acquisition of switch outputs of the ECU (addressing of solenoid valves, reverse-light relay)
- simulation of a switch (handbrake switch)

Note

The RTIO driver of the PB1650DIO2 piggyback uses two of the four available control lines as additional inputs and the remaining two control lines as additional outputs so that a total of 10 inputs and 10 outputs are available.

14.5.1 Globals (PB1650DIO2 Device)

This section describes the global options of the PB1650DIO2 device.

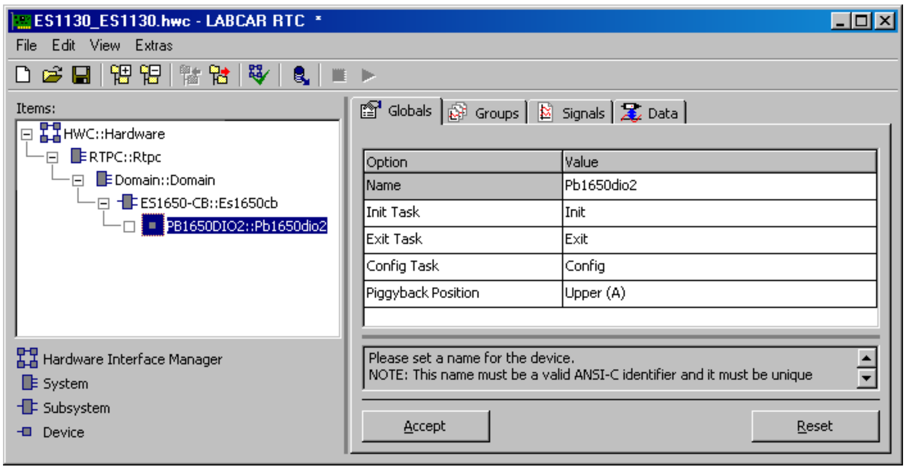


Fig. 14-8 The "Globals" Tab of the PB1650DIO2 Device

Piggyback Position

The "Piggyback Position" field specifies whether the piggyback is in the upper or lower slot of the ES1650.1 (see Fig. 14-8).

14.5.2 Groups (PB1650DIO2 Device)

The available options in the "Groups" tab are described in section "Default Options in the "Groups" Tab" on page 45.

14.5.3 Signals (PB1650DIO2 Device)

This section describes the signal-specific options of the PB1650DIO2 device.

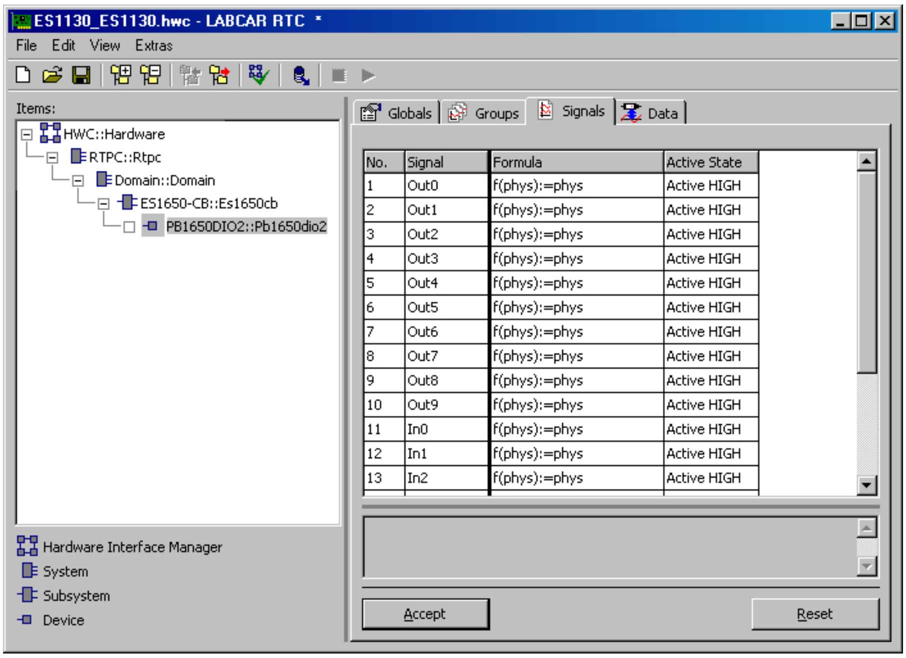


Fig. 14-9 The "Signals" Tab of the PB1650DIO2 Device

Active State

In the "Active State" field, you can specify whether the relevant signal is Active LOW or Active HIGH for each input and output of a PB1650DIO2 piggyback (see Fig. 14-9).

With an *Inx* (x = 0, 1 ... 9) input, 0 means that the generated signal is inactive; 1 means that the generated signal is active.

With an *Outx* (x = 0, 1, ... 9) output, 0 means that the relevant signal is switched as inactive; 1 means that the relevant signal is switched as active.

14.5.4 Data Types and Value Ranges of the Signals

The 10 *Outx* (x = 0, 1, ... 9) signals of the PB1650DIO2 piggyback are of the data type `uint8`. The signal is interpreted by the RTIO driver as a Boolean value. The value 0 corresponds to `FALSE`; a value not equal to 0 corresponds to `TRUE` .

The 10 *Inx* (x = 0, 1, ... 9) signals are of data type `uint8`. The returned signal value is either 0 or 1.

14.6 PB1650REL1

The PB1650REL1 piggyback provides eight dc decoupled switches in the voltage range to 175 V. It is used together with the ES1650.1 Piggyback Carrier Board as a switch module.

Typical applications are the simulation of manual switches or the switching of loads which cannot be switched using semiconductor switches. The PB1650REL1 piggyback can also be used as an analog multiplexer.

14.6.1 Globals (PB1650REL1 Device)

This section describes the global options of the PB1650REL1 device.

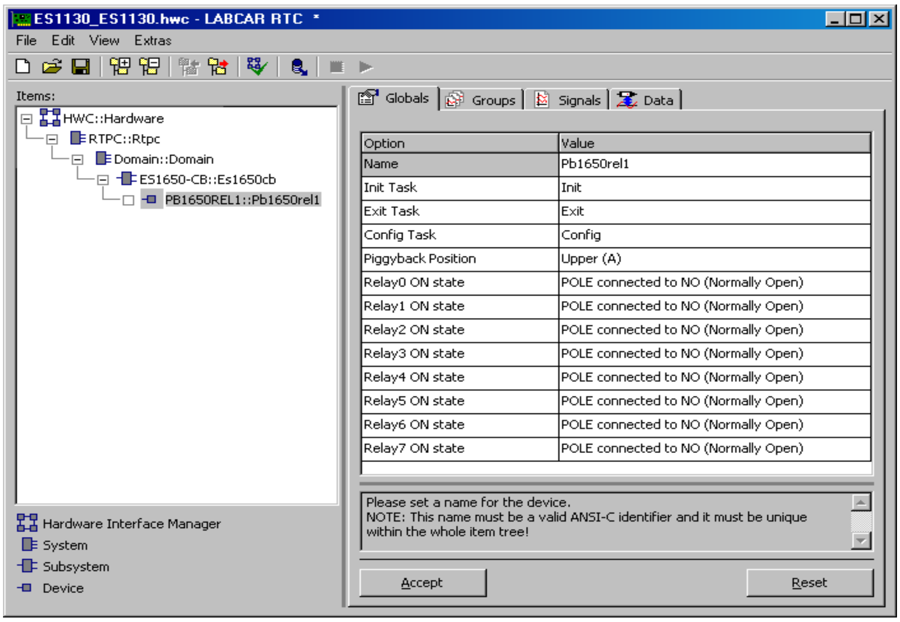


Fig. 14-10 The "Globals" Tab of the PB1650REL1 Device

Piggyback Position

The "Piggyback Position" field specifies whether the piggyback is in the upper or lower slot of the ES1650.1.

Relayx ON state

The "Relayx ON state" (x = 0, 1, ... 7) fields determine which switch setting for relay x is the closed switch state.

"POLE connected to NO (Normally Open)" means that when the switch is closed, the "POLE" connection of the relay is connected to "NO" (Normally Open).

"POLE connected to NC (Normally Closed)" means that when the switch is closed, the "POLE" connection of the relay is connected to "NC" (Normally Closed).

14.6.2 Groups (PB1650REL1 Device)

The available options in the "Groups" tab are described in section "Default Options in the "Groups" Tab" on page 45.

14.6.3 Signals (PB1650REL1 Device)

This section describes the signal-specific options of the PB1650REL1 device.

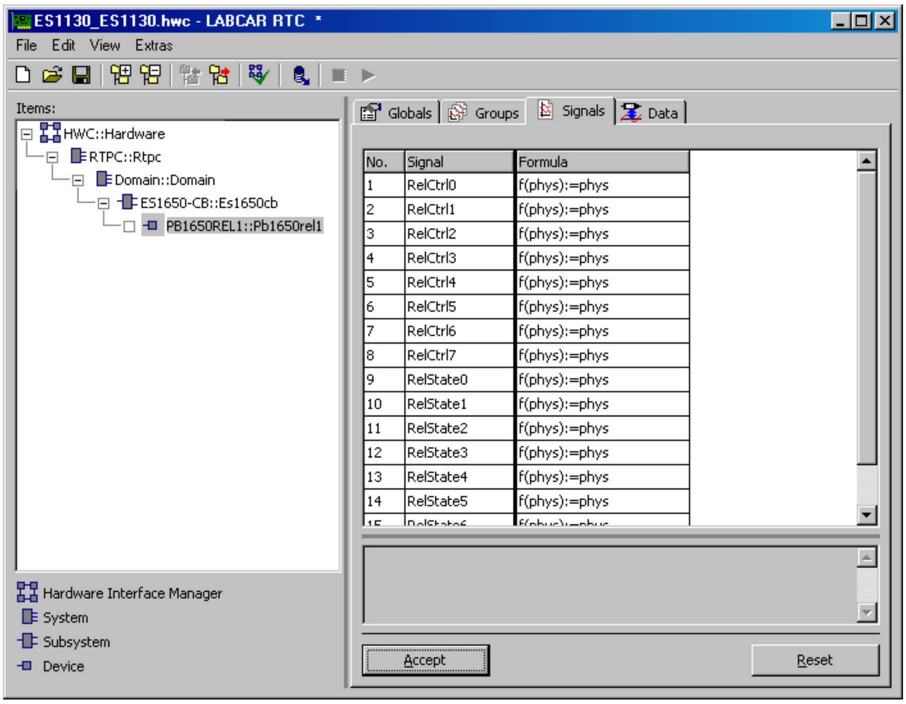


Fig. 14-11 The "Signals" Tab of the PB1650REL1 Device

- The *RelCtrlx* (x = 0, 1, ... 7) outputs control the switch setting of the relevant relay, x.

  If *RelCtrlx* is 0, relay x is open. A value not equal to 0 for *RelCtrlx* closes the relay x.

- The *RelStatex* (x = 0, 1, ... 7) inputs indicate the current switch position of the relevant relay, x.

  If *RelStatex* is 0, relay x is open. If *RelStatex* is 1, relay x is closed.

For a definition of what an open and closed relay is, please see the "Globals" tab ("Relayx ON state" on page 355).

14.6.4 Data Types and Value Ranges of the Signals

RelCtrlx

The 8 *RelCtrlx* (x = 0, 1, ... 7) signals of the PB1650REL1 device are of the data type `uint8`. The signal is interpreted by the RTIO driver as a Boolean value. The value 0 corresponds to `FALSE`; a value not equal to 0 corresponds to `TRUE`.

RelStatex

The 8 *RelStatex* (x = 0, 1, ... 7) signals are of the data type `uint8`. The returned signal value is either 0 or 1.

15 ES1391.1 Power Supply Controller Board

The ES1391.1 Power Supply Controller Board is used for the remote control of external power supplies to provide the battery voltage for LABCAR applications and (when used with an ES1392 High Current Switch Board) to switch battery nodes in LABCAR.

In addition to these two special uses, the ES1391.1 can also be used for tasks requiring the acquisition and output of unipolar, analog signals and the acquisition and output of digital signals.

While the RTIO user interface for the ES1391.1 Power Supply Controller Board was being developed, particular attention was paid to ensuring simple, intuitive operation for the user - both when using the board for the remote control of power supplies and of ES1392 High Current Switch Boards and for its more general use for analog and digital input and output.

Structure of the ES1391.1-RTIO Tree

In RTIO, the ES1391.1 board is integrated by selecting the ES1391-PWR device.

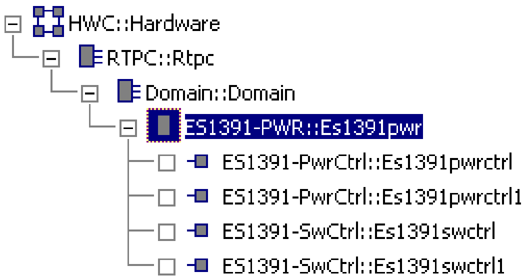


Fig. 15-1 RTIO Hardware Description with Integrated ES1391.1

The ES1391.1 has two identical functional units for the remote control of external power supplies and two identical functional units for the control of ES1392 High Current Switch Boards. This hardware structure is reflected in the RTIO tree. The user can integrate up to two ES1391-PwrCtrl devices for the remote control of external power supplies and up to two ES1391-SwCtrl devices to control ES1392 High Current Switch Boards.

15.1 ES1391-PWR Subsystem

15.1.1 Globals (ES1391-PWR Subsystem)

Fig. 15-2 shows the RTIO parameters of the "Globals" tab.

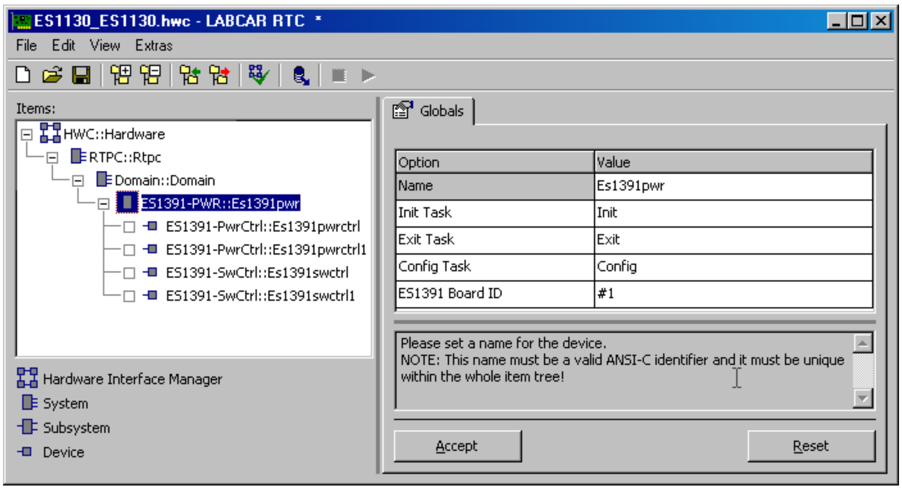


Fig. 15-2 The "Globals" Tab of the ES1391-PWR Subsystem

Note

When the ES1391.1 is integrated into a LABCAR-RTC hardware configuration, both address switches (SW1 and SW2) on the board must be set to "0x00"!

ES1391.1 Board ID

This list field is used to identify the ES1391.1. It establishes the assignment between the RTIO hardware description and the ES1391.1 in the VMEbus chassis, for which this description is valid. The ES1391.1 boards are numbered from left to right starting with 1 in the chassis (left = VMEbus slot no. 1, in ascending order). The number of the ES1391.1 board determined must be entered in the "ES1391 Board ID" list field. This RTIO parameter cannot be edited during runtime. Up to twenty ES1391.1 boards can be integrated per chassis in the RTIO Editor.

Tab. 15-1 lists the properties of the "Board ID" parameter.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
ES1391 Board ID	uint32	No	Identification of the ES1391.1. You can choose from #1 to #20.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 15-1 ES1391-PWR Subsystem: Configuration Parameters of the "Globals" Tab

Hidden Option Fields

Further option fields (see Fig. 15-3) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting **Show All**.

These option fields make it possible for ETAS service personnel to gain additional information on the ES1391.1 when localizing the cause of an error. They are not intended for the user and the default settings should therefore not be changed by the user.

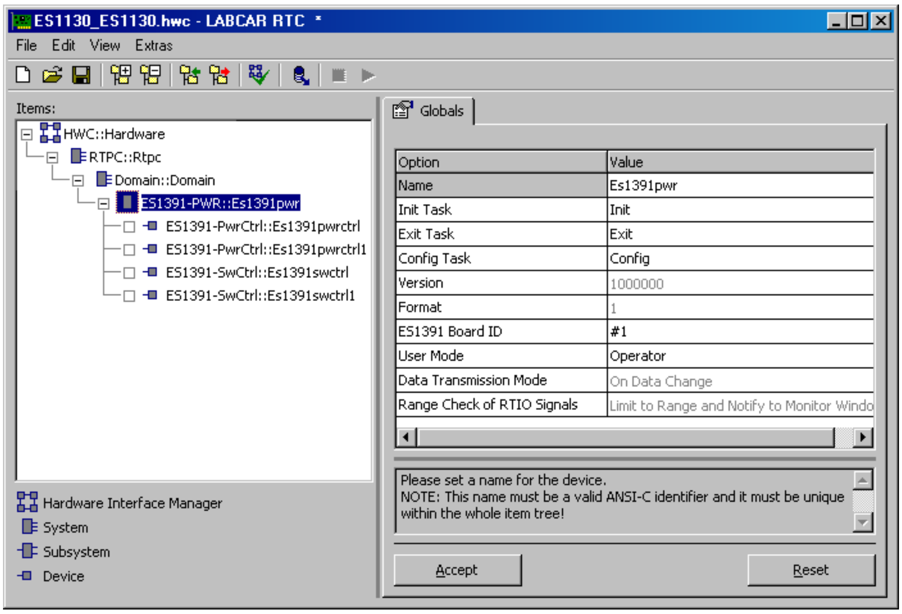


Fig. 15-3 The "Globals" Tab of the ES1391-PWR Subsystem with all Option Fields

User Mode

The "User Mode" parameter sets the RTIO user mode. Select "Operator" from this list field.

"Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. The contents of the non-volatile memory of the ES1391.1 and the contents of the EEPROM of any ES1392 High Current Switch Boards connected, for example, are output to the "Target Debug" window during initialization when "Supervisor" mode is active.

The list field can be edited during runtime and is set to "Operator" by default.

Data Transmission Mode

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies when real-time data is transferred to the experimental target. If the "On Data Change" option is activated, the data is only transferred if there has been a change. If the "Every Interval" option is activated, the real-time data is transferred at every interval of the allocated real-time task.

The list field can be edited during runtime and is set to "On Data Change" by default.

Range Check of RTIO Signals

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies to what extent the RTIO signals are to undergo a range check during runtime.

If the "No Check" option is set, there is no range check.

If the "Limit to Range" option is activated, the RTIO signals are limited to their admissible range. If, for example, a value range between a and b is defined for the RTIO signal x, the check is limited to a, if x is smaller than a and to b, if x is greater than b. If the "Limit to Range and Notify to Monitor Window" option is set, the RTIO signals are limited to their admissible range and a relevant message is displayed in the monitor window.

The "Range Check of RTIO Signals" list field cannot be edited during runtime and is set to "Limit to Range and Notify to Monitor Window" by default.

Tab. 15-2 lists the properties of the hidden RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
User Mode	uint8	Yes	User mode 0: Operator 1: Supervisor "Operator" is set by default. "Supervisor" activates additional options for localizing errors.
Data Transmission Mode	uint8	Yes	Transmission mode for real-time data 0: On Data Change 1: Every Interval "On Data Change" is set by default. The option field can only be edited if "User Mode" = "Supervisor".
Range Check of RTIO Signals	-	No	Range check for RTIO signals. "Limit to Range and Notify to Monitor Window" is set by default. The option field can only be edited if "User Mode" = "Supervisor".
* Data type which the RTIO driver uses internally for the parameter			

Tab. 15-2 ES1391-PWR Subsystem: Hidden Configuration Parameters of the "Globals" Tab.

15.2 ES1391-PwrCtrl Device

15.2.1 Globals (ES1391-PwrCtrl Device)

The ES1391-PwrCtrl device is used for the remote control of an external power supply, but can also be configured for tasks of analog and digital input and output.

Fig. 15-4 shows the "Globals" tab of the ES1391-PwrCtrl device.

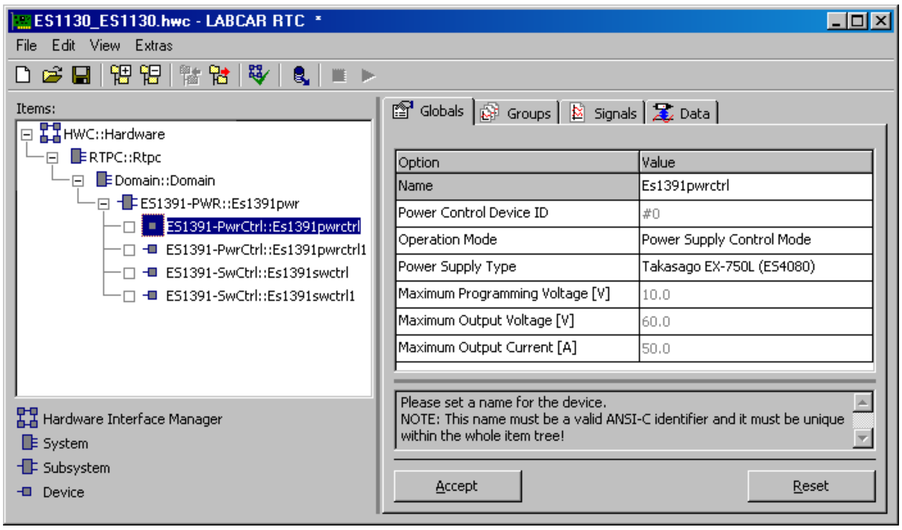


Fig. 15-4 The "Globals" Tab of the ES1391-PwrCtrl Device

Power Control Device ID

This list field allocates the ES1391-PwrCtrl device to one of the two units for the remote control of a power supply.

The list field cannot be edited; its value is entered automatically. The option value "#0" denotes the hardware unit "PwrCtrl0" and option value "#1" "PwrCtrl1".

Operation Mode

This list field determines the operating mode of the ES1391-PwrCtrl device and the relevant hardware unit. There are two modes available.

- The "Power Supply Control Mode" setting configures the "ES1391-PwrCtrl" device for the remote control of a power supply.
- The "Analog & Digital IO Mode" setting configures the "ES1391-PwrCtrl" device for the tasks of analog and digital input and output.

The list field can be edited during runtime and is set to "Power Supply Control Mode" by default.

Power Supply Type

This list field specifies the type of external power supply to be controlled. The list field can only be edited if "Power Supply Control Mode" is selected in the "Operation Mode" list field. It is of no significance in "Analog & Digital IO Mode".

In addition to several predefined power supplies, the "Other Power Supply Types" option can be activated in this list field for the remote control of a customized external power supply. The list field can be edited during runtime.

Maximum Programming Voltage [V]

This and the following two numeric input boxes are used to describe the properties of a customized external power supply.

They can only be edited if the ES1391-PwrCtrl device is configured for controlling an external power supply and "Other Power Supply Types" is active in the "Power Supply Type" list field. In all other cases, the values in these numeric input boxes are set automatically. All three input boxes can be edited during runtime.

The maximum programming voltage of the external power supply is specified in the "Maximum Programming Voltage [V]" input box. Note that programming voltages greater than 10 V cannot be generated by the ES1391.1 hardware.

Maximum Output Voltage [V]

The maximum output voltage of the power supply is specified in the "Maximum Output Voltage [V]" input box. This is the voltage the power supply delivers at its output when the programming voltage is the same as the maximum programming voltage in CV mode (CV = Constant Voltage).

Maximum Output Current [A]

The maximum output current of the power supply is specified in the "Maximum Output Current [A]" input box. This is the current the power supply delivers at its output when the programming voltage is the same as the maximum programming voltage in CC mode (CC = Constant Current).

Tab. 15-3 lists the properties of the individual RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Power Control Device ID	uint32	No	Allocation of a hardware unit for the remote control of a power supply to the ES1391-PwrCtrl device 0: Hardware unit "PwrCtrl0" 1: Hardware unit "PwrCtrl1" Cannot be edited; is set automatically.
Operation Mode	uint8	Yes	Operating mode 0: Analog & Digital IO Mode 1: Power Supply Control Mode "Power Supply Control Mode" is set by default.
Power Supply Type	uint32	Yes	Type of external power supply 0: Takasago EX-375L 1: Takasago EX-750L 2: Takasago EX-1500L 255: Other Power Supply Types Can only be edited in "Power Supply Control Mode".
Maximum Programming Voltage [V]	real32	Yes	Maximum programming voltage of the external power supply in V. To describe customized external power supplies. Value range: > 0 V
Maximum Output Voltage [V]	real32	Yes	Maximum output voltage of the external power supply in V. To describe customized external power supplies. Value range: > 0 V
Maximum Output Current [A]	real32	Yes	Maximum output current of the external power supply in A. To describe customized external power supplies. Value range: > 0 V

* Data type which the RTIO driver uses internally for the parameter

Tab. 15-3 ES1391-PwrCtrl Subsystem: Configuration Parameters of the "Globals" Tab.

15.2.2 Groups (ES1391-PwrCtrl Device)

The ES1391-PwrCtrl device has four signal groups.

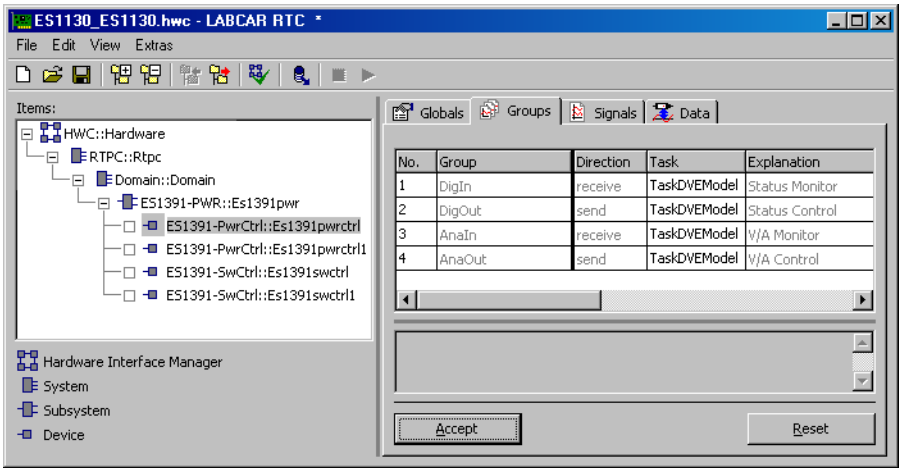


Fig. 15-5 The "Groups" Tab of the ES1391-PwrCtrl Device

- "DigIn"

  This signal group is transferred from the ES1391.1 board to the experimental target. It transports status information of the digital outputs of an external power supply or, more generally speaking, status information of the digital inputs of the hardware unit for remote control of a power supply.

- "DigOut"

  This signal group is transferred from the experimental target to the ES1391.1 board. It transports control information for the digital inputs of the external power supply or, more generally speaking, for the digital outputs of the hardware unit.

- "AnaIn"

  This signal group is transferred from the ES1391.1 board to the experimental target. It transports the actual current and actual voltage values of the external power supply or, more generally speaking, the voltage values at the analog inputs of the hardware unit.

- "AnaOut"

  This signal group is transferred from the experimental target to the ES1391.1 . It transports target voltage and target current values of the external power supply or, more generally speaking, the voltage values for the analog outputs of the hardware unit.

Tasks of the real-time operating system must be allocated to the signal groups. Normally, tasks with periodic activation are selected for all signal groups. The activation period depends on the dynamic behavior of the signals to be acquired and output as well as on the properties of the ES1391.1 firmware. The ES1391.1 firmware acquires the analog and digital inputs at a rate of 1 millisecond and also updates its analog and digital outputs at the same rate. This rate limits useful activation periods of the RTIO tasks to one millisecond.

RTIO Signals of the "DigIn" Signal Group

The "DigIn" signal group has five RTIO signals, "DigIn_0" to "DigIn_2" and "DigOutSts_0" and "DigOutSts_1", which are of the data type "bool". The "DigIn_x" signals (x = 0, 1, 2) show whether the digital input x is actiive (value ≠ 0) or inactive (value = 0). The "DigOutSts_x" signals (x = 0, 1) show whether the digital output x is active (value ≠ 0) or inactive (value = 0). The active level is determined by the polarity (high-active or low-active) of the input or output. The polarity for the digital inputs and outputs is specified in the "Signals" tab. Tab. 15-4 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
DigIn_x (x = 0, 1, 2) DigOutSts_x (x = 0, 1)	bool	Status acquisition of the digital inputs and outputs = 0: Input/output is inactive ≠ 0: Input/output is active
* Data type which the RTIO driver uses internally for the parameter		

Tab. 15-4 ES1391-PwrCtrl Device: The RTIO Signals of the "DigIn" Signal Group

If the ES1391-PwrCtrl device is in "Power Supply Control Mode" for the remote control of a power supply, the individual RTIO signals of the "DigIn" signal group have a predefined meaning, which is listed in Tab. 15-5.

RTIO Signal	Meaning	Signal Values
DigIn_0	Alarm input	= 0: Alarm inactive ≠ 0: Alarm active
DigIn_1	CV mode CV = Constant Voltage	= 0: CV mode inactive ≠ 0: Power supply is in CV mode
DigIn_2	CC mode CC = Constant Current	= 0: CC mode inactive ≠ 0: Power supply is in CC mode
DigOutSts_0	Output of power supply	= 0: Output of the power supply is activated ≠ 0: Output of the power supply is deactivated
DigOutSts_1	Input of power supply	= 0: The power supply is connected to the mains. ≠ 0: The power supply is disconnected from the mains.

Tab. 15-5 Predefined meaning of the RTIO signals of the "DigIn" signal group (in "Power Supply Control Mode" for the remote control of a power supply)

RTIO Signals of the "DigOut" Signal Group

The "DigOut" signal group has two RTIO signals, "DigOutCtrl_0" and "DigOutCtrl_1", which are of the data type "bool". The two digital outputs of a hardware unit for power supply control are activated (value ≠ 0) and deactivated (value = 0) with these signals. The active level is determined by the polarity (high-active or low-active) of the outputs. The polarity of the outputs is specified in the "Signals" tab.

Tab. 15-6 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
DigOutCtrl_x (x = 0, 1)	bool	Control of the digital outputs = 0: Deactivation of the output ≠ 0: Activation of the output
* Data type which the RTIO driver uses internally for the parameter		

Tab. 15-6 ES1391-PwrCtrl Device: The RTIO Signals of the "DigOut" Signal Group

If the ES1391-PwrCtrl device is in "Power Supply Control Mode" for the remote control of a power supply, the individual RTIO signals of the "DigOut" signal group have a predefined meaning, which is listed in Tab. 15-7.

RTIO Signal	Meaning	Signal Values
DigOutCtrl_0	Output of power supply	= 0: Deactivate the output of the power supply ≠ 0: Activate the output of the power supply
DigOutCtrl_1	Input of power supply	= 0: Connect the power supply to the mains ≠ 0: Disconnect the power supply from the mains

Tab. 15-7 Predefined meaning of the RTIO signals of the "DigOut" signal group (in "Power Supply Control Mode" for the remote control of a power supply)

Note

The ES1391.1-SwCtrl device makes it possible to react to an active alarm signal of a connected ES1392 by switching off the power supplies (controlled by the ES1391.1).Switching off means that the outputs of the power supplies are deactivated.
If an ES1391-SwCtrl device is configured accordingly, it is possible that the "DigOutCtrl_0" control signal for activating/deactivating a power supply output deviates from the "DigOutSts_0" status signal.

RTIO Signals of the "AnaIn" Signal Group

The "AnaIn" signal group has two RTIO signals, "AnaIn_0" and "AnaIn_1", which are of the data type "real32".

If the ES1391-PwrCtrl device is being operated in the mode for digital and analog input and output, the RTIO signals specify the voltages at the two analog inputs of the hardware unit. The value range of the RTIO signals is then 0.0 V to 10.0 V.

If the ES1391-PwrCtrl device is being operated in the mode for the remote control of a power supply, the RTIO signal "AnaIn_0" specifies the current output voltage of the power supply in volts and "AnaIn_1" the current output current of the power supply in amperes. Tab. 15-8 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
AnaIn_0	real32	Analog input 0 "Analog & Digital IO Mode" operating mode: Input voltage in volts (0.0 V to 10.0 V) "Power Supply Control Mode" operating mode: Current output voltage of the power supply in volts
AnaIn_1	real32	Analog input 1 "Analog & Digital IO Mode" operating mode: Input voltage in volts (0.0 V to 10.0 V) "Power Supply Control Mode" operating mode: Current output current of the power supply in amperes

* Data type which the RTIO driver uses internally for the parameter

Tab. 15-8 ES1391-PwrCtrl Device: The RTIO Signals of the "AnaIn" Signal Group

RTIO Signals of the "AnaOut" Signal Group

The "AnaOut" signal group has two RTIO signals, "AnaOut_0" and "AnaOut_1", which are of the data type "real32".

If the ES1391-PwrCtrl device is being operated in the mode for digital and analog input and output, the RTIO signals define the voltages at the two analog outputs of the hardware unit. The value range of the RTIO signals is then 0.0 V to 10.0 V.

If the ES1391-PwrCtrl device is being operated in the mode for the remote control of a power supply, the RTIO signal "AnaOut_0" defines the output voltage[1] of the power supply in volts and "AnaOut_1" the output current[2] of the power supply in amperes.

1. In CV mode (CV = Constant Voltage) of the power supply
2. In CC mode (CC = Constant Current) of the power supply

Tab. 15-9 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
AnaOut_0	real32	Analog output 0 "Analog & Digital IO Mode" operating mode: Output voltage in volts (0.0 V to 10.0 V) "Power Supply Control Mode" operating mode: Target output voltage of the power supply in volts
AnaOut_1	real32	Analog input 1 "Analog & Digital IO Mode" operating mode: Output voltage in volts (0.0 V to 10.0 V) "Power Supply Control Mode" operating mode: Target output current of the power supply in amperes

* Data type which the RTIO driver uses internally for the parameter

Tab. 15-9 ES1391-PwrCtrl Device: The RTIO Signals of the "AnaOut" Signal Group

15.2.3 Signals (ES1391-PwrCtrl Device)

The digital and analog inputs/outputs of a hardware unit for controlling power supplies remotely are configured in the "Signals" tab. Fig. 15-6 shows the RTIO parameters of the "Signals" tab.

All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

Polarity

The polarity (high-active or low-active) of a digital input or output is specified in this list field. If the ES1391-PwrCtrl device is being used to remotely control one of the predefined Takasago power supplies, the polarity values for the individual digital inputs and outputs are set automatically - in this case they cannot be edited by the user. They can be edited in all other cases.

Samples for Averaging

The voltage value of an analog input can be determined as an average of several sample values. The number of sample values (for the analog inputs) from which this average value is to be calculated is specified in this input box. The maximum number of sample values that can be used in this calculation is 100.

The ES1391.1 firmware samples the analog inputs at a rate of 1 millisecond. The average value is also updated at the same rate. In a calculation of the average value using n sample values, the sample values of the last n milliseconds are used.

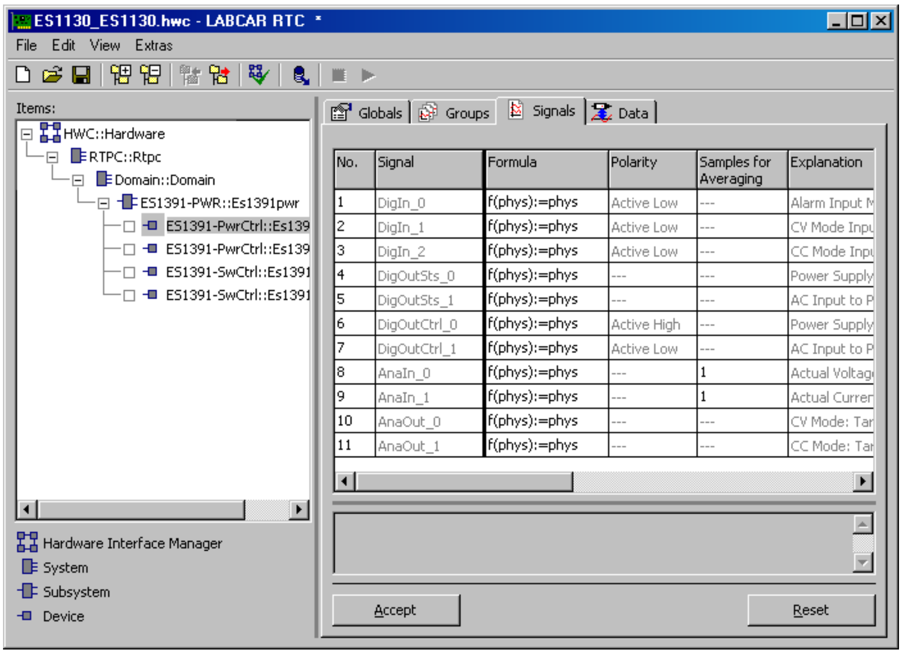


Fig. 15-6 The "Signals" Tab of the "ES1391-PwrCtrl" Device (for Remote Control of a Takasago Power Supply)

Tab. 15-10 lists the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Polarity	bool	Yes	Polarity of the digital input or output = 0: Low-active ≠ 0: High-active
Samples for Averaging	uint32	Yes	Analog input: Number of sample values for calculating the average value. Value range: 1 ... 100

* Data type which the RTIO driver uses internally for the parameter

Tab. 15-10 ES1391-PwrCtrl Device: Configuration Parameters of the "Signals" Tab

15.3 ES1391-SwCtrl Device

15.3.1 Globals (ES1391-SwCtrl Device)

The ES1391-SwCtrl device is used to control an ES1392 High Current Switch Board - it can, however, also be configured for tasks of digital input and output. Fig. 15-7 shows the "Globals" tab of an ES1391-SwCtrl device.

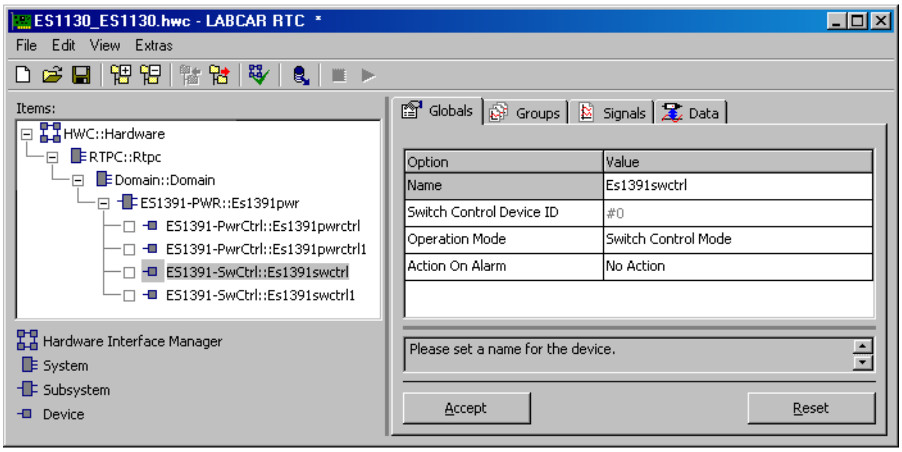


Fig. 15-7 The "Globals" Tab of the ES1391-SwCtrl Device

Switch Control Device ID

This list field allocates the ES1391-SwCtrl device to one of the two control units available on the ES1391.1 board.

The list field cannot be edited; its value is entered automatically. The option value "#0" denotes the hardware unit "SwCtrl0" and option value "#1" "SwCtrl1".

Operation Mode

This list field determines the operating mode of the ES1391-SwCtrl device and the relevant hardware unit. There are two modes available.

- The "Switch Control Mode" setting configures the ES1391-SwCtrl device for the control of ES1392 High Current Switch Boards.
- The "Digital IO Mode" setting configures the ES1391-SwCtrl device for digital input and output.

The list field can be edited during runtime and is set to "Switch Control Mode" by default.

Action On Alarm

This list field is where it is specified which actions are executed as a reaction to an active alarm signal of the ES1392 High Current Switch Board connected to the hardware unit. It can only be edited or is only of any significance if the ES1391-SwCtrl device is being used in "Switch Control Mode" to control the ES1392. One or both of the power supplies connected to the hardware units for remotely controlling the power supply can be disconnected as a reaction to an active alarm signal of the ES1392 High Current Switch Board.

The list field can be edited during runtime.

Tab. 15-11 lists the properties of the individual RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Switch Control Device ID	uint32	No	Allocation of a hardware unit for ES1392 control to the ES1391-SwCtrl device 0: Hardware unit "SwCtrl0" 1: Hardware unit "SwCtrl1" Cannot be edited, is set automatically.
Operation Mode	uint8	Yes	Operating mode 0: Digital IO Mode 1: Switch Control Mode "Switch Control Mode" is set by default.
Action On Alarm	uint8	Yes	Reaction to active ES1392 alarm signal 0: No reaction 1: Switch off power supply to "PwrCtrl0" port 2: Switch off power supply to "PwrCtrl1" port 3: Switch off both power supplies Can only be edited in "Switch Control Mode".
* Data type which the RTIO driver uses internally for the parameter			

Tab. 15-11 ES1391-SwCtrl Device: Configuration Parameters of the "Globals" Tab

15.3.2 Groups (ES1391-SwCtrl Device)

The ES1391-SwCtrl device has two signal groups (see Fig. 15-8).

The "DigIn" signal group is transferred from the ES1391.1 to the experimental target. It transports status information of the digital outputs of an ES1392 High Current Switch Board or, more generally speaking, status information of the digital inputs of the allocated hardware unit.

The "DigOut" signal group is transferred from the experimental target to the ES1391.1. It transports control information for the digital inputs of an ES1392 High Current Switch Board or, more generally speaking, for the digital outputs of the allocated hardware unit.

Tasks of the real-time operating system must be allocated to the signal groups. Normally, tasks with periodic activation are selected for all signal groups. The activation period depends on the dynamic behavior of the signals to be acquired and output as well as on the properties of the ES1391.1 firmware. The ES1391.1

firmware acquires the digital inputs at a rate of 1 millisecond and also updates its digital outputs at the same rate. This rate limits useful activation periods of the RTIO tasks to one millisecond.

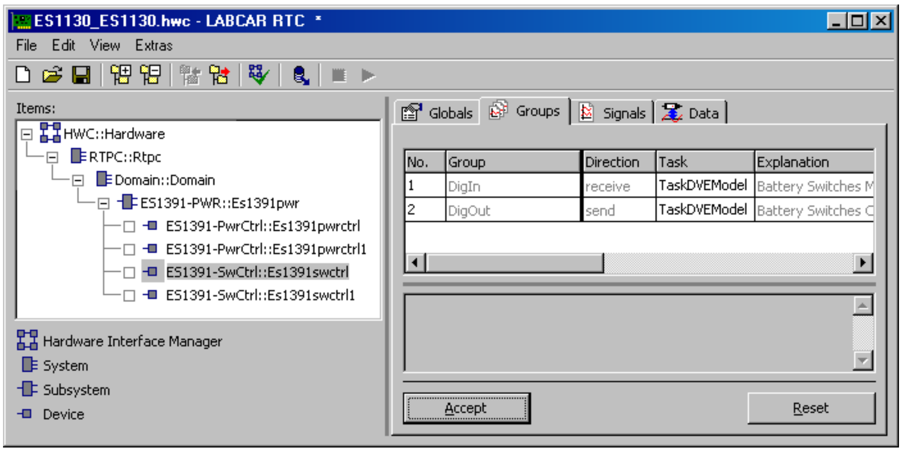


Fig. 15-8 The "Groups" Tab of the ES1391-SwCtrl Device

RTIO Signals of the "DigIn" Signal Group

The "DigIn" signal group has eight RTIO signals, "DigIn_0", "DigIn_1" and "DigOutSts_0" to "DigOutSts_5", which are of the data type "bool".

The "DigIn_x" signals (x = 0, 1) show whether the digital input x is active (value ≠ 0) or inactive (value = 0). The "DigOutSts_x" signals (x = 0, 1, ... 5) show whether the digital output x is active (value ≠ 0) or inactive (value = 0). The active level is determined by the polarity (high-active or low-active) of the input or output. The polarity is specified in the "Signals" tab. Tab. 15-12 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
DigIn_x (x = 0, 1) DigOutSts_x (x = 0, 1, ... 5)	bool	Acquiring the status of the digital inputs and outputs = 0: Input / output is inactive ≠ 0: Input / output is active
* Data type which the RTIO driver uses internally for the parameter		

Tab. 15-12 ES1391-SwCtrl Device: The RTIO Signals of the "DigIn" Signal Group

If the ES1391-SwCtrl device is in "Switch Control Mode" to control an ES1392 High Current Switch Board, the individual RTIO signals of the "DigIn" signal group have a predefined meaning, which is listed in Tab. 15-13.

RTIO Signal	Meaning	Signal Values
DigIn_0	Main relay input	= 0: Main relay open ≠ 0: Main relay closed
DigIn_1	Alarm input	= 0: Alarm inactive ≠ 0: Alarm active
DigOutSts_0 to DigOutSts_4	High Current Switch 0 to High Current Switch 4	= 0: Switch open ≠ 0: Switch closed
DigOutSts_5	Configuration of high current switch 0	= 0: Switch 0 switches to battery voltage ≠ 0: Switch 0 switches to battery ground

Tab. 15-13 Predefined Meaning of the RTIO Signals of the "DigIn" Signal Group (in "Switch Control Mode" to Control an ES1392 High Current Switch Board

RTIO Signals of the "DigOut" Signal Group

The "DigOut" signal group has six RTIO signals, "DigOutCtrl_0" to "DigOutCtrl_5", which are of the data type "bool". The digital outputs of a hardware unit are activated (value ≠ 0) and deactivated (value = 0) with these signals. The active level is determined by the polarity (high-active or low-active) of the outputs. The polarity of the outputs is specified in the "Signals" tab.

Tab. 15-14 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
DigOutCtrl_x (x=0, 1, ... 5)	bool	Control of the digital outputs 0: Deactivation of the output ≠ 0: Activation of the output
* Data type which the RTIO driver uses internally for the parameter		

Tab. 15-14 ES1391-SwCtrl Device: The RTIO Signals of the "DigOut" Signal Group

If the ES1391-SwCtrl device is in "Switch Control Mode" to control an ES1392 High Current Switch Board, the individual RTIO signals of the "DigOut" signal group have a predefined meaning, which is listed in Tab. 15-15.

RTIO Signal	Meaning	Signal Values
DigOutCtrl_0 to DigOutCtrl_4	High Current Switch 0 to High Current Switch 4	= 0: Open switch ≠ 0: Close switch
DigOutCtrl_5	Configuration of High Current Switch 0	= 0: Switch 0 switches to battery voltage ≠ 0: Switch 0 switches to battery ground

Tab. 15-15 Predefined Meaning of the RTIO Signals of the "DigOut" Signal Group (in "Switch Control Mode" to Control anES1392 High Current Switch Board)

15.3.3 Signals (ES1391-SwCtrl Device)

The digital inputs and outputs of a hardware unit are configured in the "Signals" tab. Fig. 15-9 shows the RTIO parameters of the "Signals" tab. All parameters can be edited can be edited online (i.e. during runtime of the model on the experimental target).

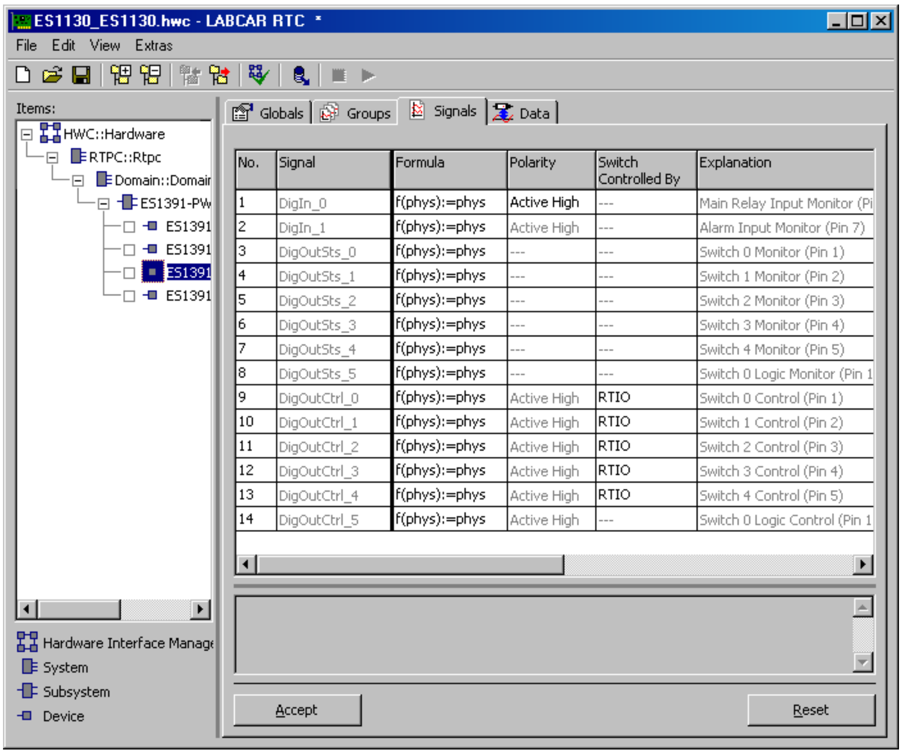


Fig. 15-9 The "Globals" Tab of the ES1391-SwCtrl Device (in "Switch Control Mode" to Control an ES1392 High Current Switch Board)

Polarity

The polarity (high-active or low-active) of a digital input or output is specified in this list field.

If the ES1391-SwCtrl device is being used in "Switch Control Mode" to control an ES1392 High Current Switch Board, the polarity values for the individual digital inputs and outputs are set automatically - in this case they cannot be edited by the user. The exception is the digital input 0 (the main relay input), the polarity of which can be edited.

If the ES1391-SwCtrl device is being used in "Digital IO Mode" for digital input/output, the polarity fields of all inputs and outputs can be edited.

Switch Controlled By

This list field is only relevant and can only be edited if the ES1391-SwCtrl device is being used to control an ES1392 High Current Switch Board (i.e. in "Switch Control Mode").

It specifies whether the allocated high-current switch of the ES1392 High Current Switch Board is powered on/off via RTIO ("RTIO" setting) or depending on the main relay status ("Main Relay" setting). If a high current switch is controlled by the main relay status, it is closed if the main relay is closed and opened, if the main relay is open.

If the ES1391-SwCtrl device is being used in "Digital IO Mode" for digital input/output, this list field is automatically set to "RTIO" for all digital outputs and cannot be edited.

Tab. 15-16 lists the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Polarity	bool	Yes	Polarity of the digital input or output = 0: Low-active ≠ 0: High-active
Switch Controlled By	bool	Yes	Control source for digital output = 0: RTIO ≠ 0: Main relay

* Data type which the RTIO driver uses internally for the parameter

Tab. 15-16 ES1391-SwCtrl Device: Configuration Parameters of the "Signals" Tab

16 ES1651.1 Carrier Board

This section describes the RTIO integration of the ES1651.1 Carrier Board which is used as a carrier board for up to four I/O modules in the LABCAR environment The board also provides a crankshaft angle clock bus as well as a connection to the CAN bus via two identical CAN controllers.

Structure of the ES1651.1 RTIO Tree

An ES1651.1 Carrier Board is integrated in the VMEbus chassis and connected directly to the experimental target there via the VMEbus. The hardware connection described is reflected in the RTIO tree (Fig. 16-1).

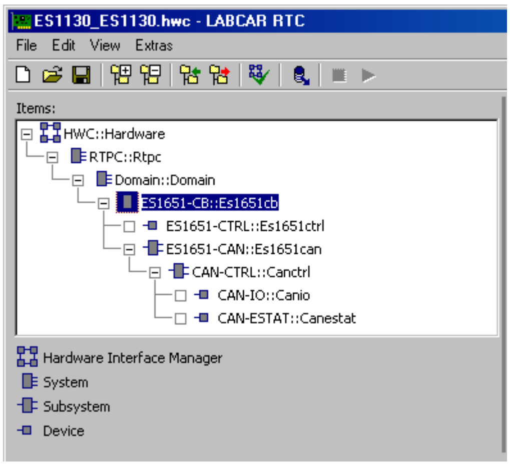


Fig. 16-1 RTIO Hardware Description with Integrated ES1651.1

The hardware of an ES1651.1 Carrier Board has a synchronization bus with four lines as well as a crankshaft angle clock bus – specially for testing engine ECUs – via which the I/O modules can be synchronized with a crankshaft angle clock for speed-synchronous signal acquisition and/or signal generation. The bus lines are configured and addressed using the ES1651-CTRL RTIO element. The CAN bus is configured and addressed using the "ES1651-CAN" RTIO element.

16.1 ES1651-CB Subsystem

16.1.1 Globals (ES1651-CB Subsystem)

Fig. 16-2 shows the "Globals" tab of an ES1651-CB subsystem.

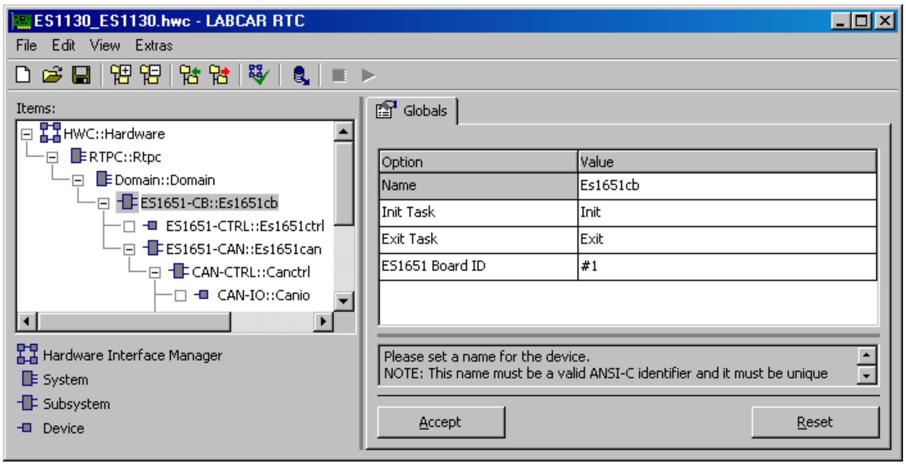


Fig. 16-2 The "Globals" Tab of the ES1651-CB Subsystem

ES1651 Board ID

This option field is used to identify the ES1651.1 Carrier Board. It establishes the assignment between the RTIO hardware description and the ES1651.1 in the VMEbus chassis for which this description is valid. The numbering of the ES1651.1 Carrier Boards in the VMEbus chassis takes place from left to right (starting with 1). Up to 20 ES1651.1s can be integrated per VMEbus chassis.

This RTIO parameter cannot be set during runtime.

Tab. 16-1 summarizes the properties of the "ES1651 Board ID" parameter.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
ES1651 Board ID	uint32	No	Identification of the ES1651.1. You can choose from #1 to #20.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-1 ES1651-CB Subsystem: Configuration Parameters of the "Globals" Tab

16.1.2 Hidden Option Fields

Further option fields (Fig. 16-3) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting "Show all Options" from the shortcut menu.

These option fields make it possible for ETAS service personnel to gain additional information on the ES1651.1 Carrier Board when localizing the cause of an error. They are not intended for the user and the default settings should therefore not be changed by the user.

Fig. 16-3 shows the "Globals" tab of an ES1651-CB subsystem with hidden options.

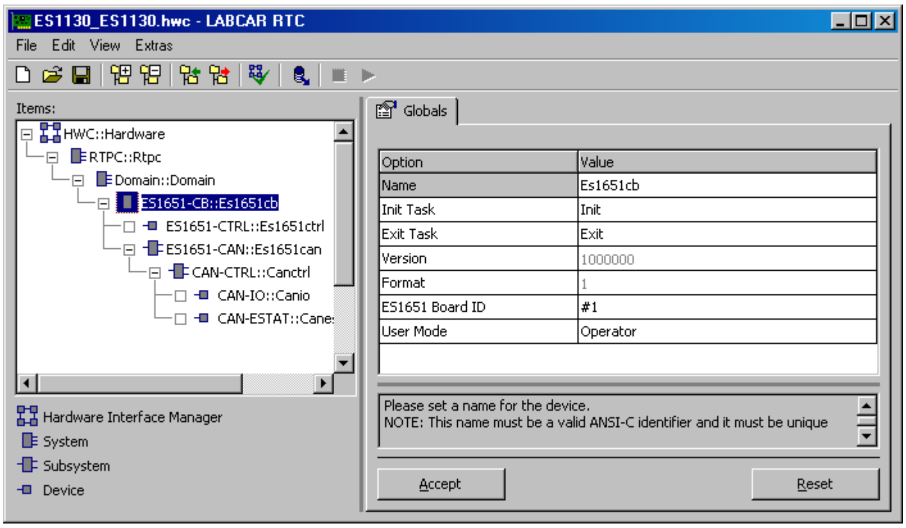


Fig. 16-3 The "Globals" Tab of the ES1651-CB Subsystem with Hidden Options

User Mode

Sets the RTIO user mode. Users should select "Operator" from the list field. "Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. The list field can be edited during runtime and is set to "Operator" by default.

Tab. 16-2 summarizes the properties of the options.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
User Mode	uint8	Yes	User mode 0: "Operator" (preset) 1: "Supervisor" activates additional options for localizing errors.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-2 ES1651-CB Subsystem: Options of the "Globals" Tab

16.2 ES1651-CTRL Device

16.2.1 Globals (ES1651-CTRL Device)

Fig. 16-4 shows the "Globals" tab of an ES1651-CTRL Device.

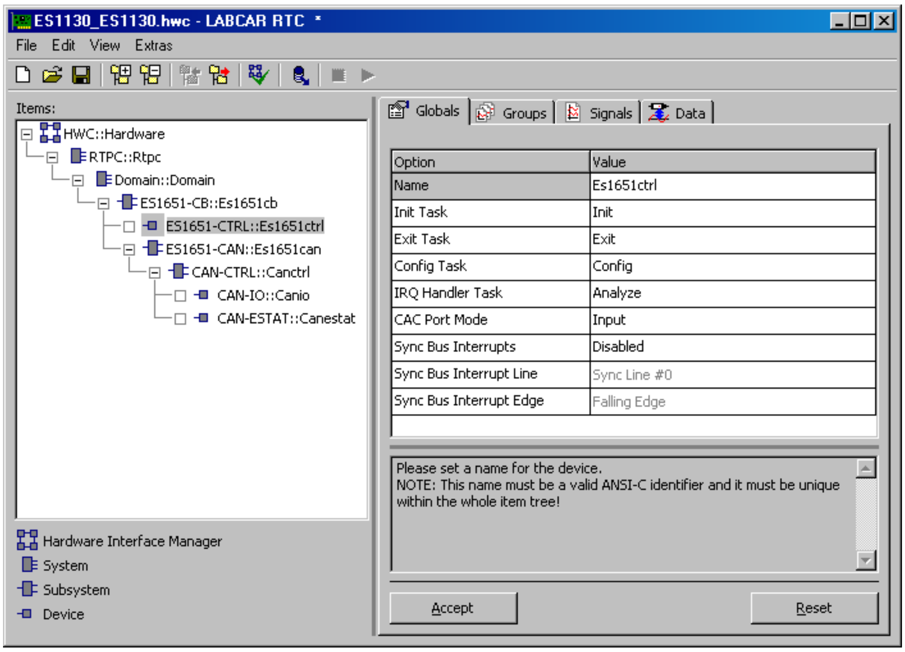


Fig. 16-4 The "Globals" Tab of the ES1651-CTRL Device

Analyze Task

An additional "Analyze Task" is required to support interrupts of the ES1651.1 Carrier Board. This task has to be created as a software task in the task list in the project editor – the number of possible task activations has to be between 2 and 255.

CAC Port Mode

This option field is used to configure the 9-pin Sub-D connector on the front panel of the ES1651.1 Carrier Board with reference to the crankshaft angle clock bus.

The connector can be configured in four different ways. "Isolated" isolates the connector from the crankshaft angle clock bus. "Output" makes it possible to supply external hardware with CAC signals of the ES1651.1. In the input mode "Input", crankshaft angle clock signals for carrier board hardware or for the I/O modules are made available by external sources. "Input ES1332 CAC" guarantees the processing of combined CAC synchronization and clock signals which can be generated by the ES1332.

Sync Bus Interrupts

This option field activates or deactivates the capability to generate VMEbus interrupts by transferring signals on the synchronization bus. "Enabled" means "interrupt generation is active" and "Disabled" means "interrupt generation is not active".

Sync Bus Interrupt Line

This option field is used to specify the interrupt source, i.e. which of the four sync lines is enabled to generate the VMEbus interrupt on signal transfer. This option can only be edited if "Sync Bus Interrupts" is "Enabled".

Sync Bus Interrupt Edge

This option field is used to select edge triggering for the generation of VMEbus interrupts. In this option, an interrupt is generated either on a falling or on a rising edge. This option can only be edited if "Sync Bus Interrupts" is "Enabled".

Tab. 16-3 summarizes the properties of the configuration parameters.

Parameter or Option Field	**Data Type***	**Can be Edited Online**	**Comment/Value Range**
CAC Port Mode	uint8	Yes	Configuration of the Sub-D connector on the front panel with reference to the crankshaft angle clock bus: 0: Connector is isolated from the crankshaft angle clock bus 1: Connector operates as an output which provides external hardware with CAC signals 2: Connector operates as an input and makes CAC signals from the carrier board hardware or from I/O modules available 3: Connector operates as an input and processes combined CAC synchronization and clock signals which can be generated by the ES1332
Sync Bus Interrupts	bool	Yes	Generation of VMEbus interrupts by transferring signals on the synchronization bus: False: deactivated True: activated
Sync Bus Interrupt Line	uint8	Yes/No	Source for VMEbus interrupt generation: 0: Sync line 0 1: Sync line 1 2: Sync line 2 3: Sync line 3
Sync Bus Interrupt Edge	uint8	Yes/No	Edge triggering for the generation of VMEbus interrupts: 0: Triggering with a falling edge 1: Triggering with a rising edge

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-3 ES1651-CB Device: Configuration Parameters of the "Globals" Tab

16.2.2 Hidden Option Fields

Further option fields (Fig. 16-5) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting "Show all Options" from the shortcut menu. These option fields make it possible for ETAS service personnel to gain additional information on the ES1651.1 Carrier Board when localizing the cause of an error.

They are not intended for the user and the default settings should therefore not be changed by the user.

Fig. 16-5 shows the "Globals" tab of an ES1651-CTRL device with hidden options.

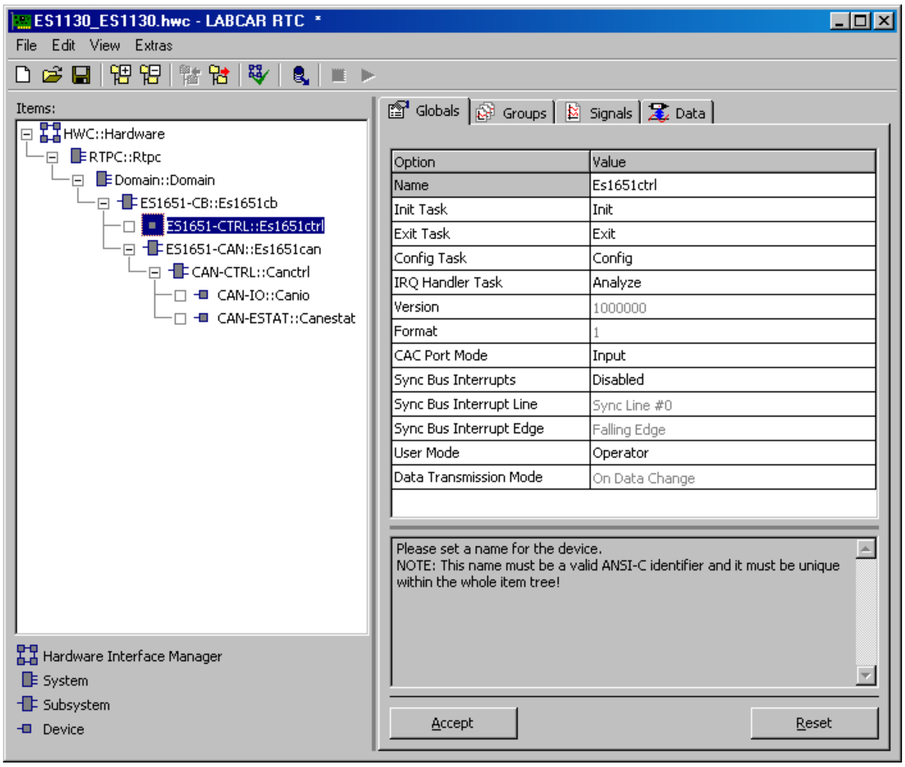


Fig. 16-5 The "Globals" Tab of the ES1651-CTRL Device with Hidden Options

User Mode

Sets the RTIO user mode. Users should select "Operator" from the list field. "Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. The list field can be edited during runtime and is set to "Operator" by default.

Data Transmission Mode

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies when real-time data is transmitted from the experimental target to the ES1651.1 Carrier Board.

If the "On Data Change" option is activated, signal group data is only transmitted if there has been a change. If the "Every Interval" option is activated, the real-time data is transmitted at every interval of the allocated real-time task.

The list field can be edited during runtime and is set to "On Data Change" by default.

Tab. 16-4 summarizes the properties of the options.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
User Mode	uint8	Yes	User mode 0: "Operator" is preset 1: "Supervisor" activates additional options for localizing errors
Data Transmission Mode	uint8	Yes	Transfer mode of the real-time data 0: "On Data Change" 1: "Every Interval" "On Data Change" is preset The option field can only be edited if "User Mode" = "Supervisor"

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-4 ES1651-CTRL Device: Options of the "Globals" Tab

16.2.3 Groups (ES1651-CTRL Device)

The ES1651-CTRL device has two signal groups which are transferred from the experimental target to the ES1651.1 Carrier Board or from the ES1651.1 to the experimental target (see Fig. 16-6).

The "SyncLvlCtrl" signal group transports level information for the sync lines controlled from to the ES1651.1.

The "SyncLvlMntr" signal group transports level information of the sync lines to the RTIO. This means that the individual level statuses of the sync lines can be displayed/monitored by the RTIO.

One task of the real-time operating system must be assigned to each signal group. A task with periodic activation is usually selected. The activation period depends on the dynamic behavior of the synchronization signals to be generated or monitored. Useful activation periods are limited downwards by runtimes of the firmware on the carrier boards to 1 millisecond.

Fig. 16-6 shows the "Groups" tab of an ES1651-CTRL device.

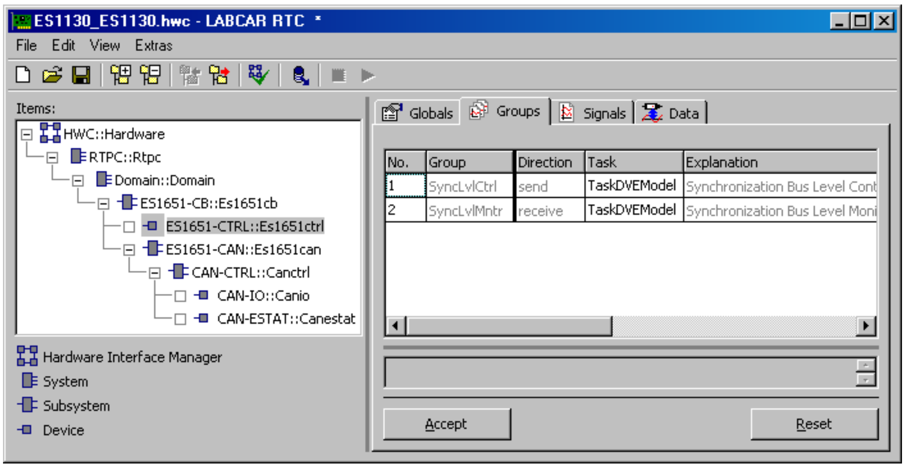


Fig. 16-6 The "Groups" Tab of the ES1651-CTRL Device

RTIO Signals of the "SyncLvlCtrl" Signal Group

The "SyncLvlCtrl" signal group has four RTIO signals, "SyncLvlCtrl_0" to "SyncLvlCtrl_3", which are of the data type "bool". If the sync line x (x = 0, 1, 2, 3) is configured to be controlled from the RTIO, the value of the RTIO signal "SyncLvlCtrl_x" defines the level of the line. "0" means low level; "1" means high level. If the sync line is not controlled by the RTIO, the signal is of no significance - the "Data" column of the sync lines, which are not controlled by the RTIO, is inactive.

Tab. 16-5 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
SyncLvlCtrl_0 to SyncLvlCtrl_3	bool	Yes	Level control on sync line x 0: Low level 1: High level
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-5 RTIO Signals of the "SyncLvlCtrl" Group

RTIO Signals of the "SyncLvlMntr" Signal Group

The "SyncLvlMntr" signal group has four RTIO signals, "SyncLvlMntr_0" to "SyncLvlMntr_3", which are of the data type "bool". It is used to monitor the current level of the relevant sync line.

"0" means low level; "1" means high level on the configured sync line. If a task with periodic activation is assigned to the signal group, the cyclical monitoring of all sync lines is permanently active.

Tab. 16-6 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
SyncLvlMntr_0 to SyncLvlMntr_3	bool	No	Level monitoring on sync line x 0: Low level 1: High level
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-6 RTIO Signals of the "SyncLvlMntr" Group

16.2.4 Signals (ES1651-CTRL Device)

In the "Signals" tab, the sync lines of an ES1651.1 Carrier Board are configured. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

Fig. 16-7 shows the "Signals" tab of an ES1651-CTRL device.

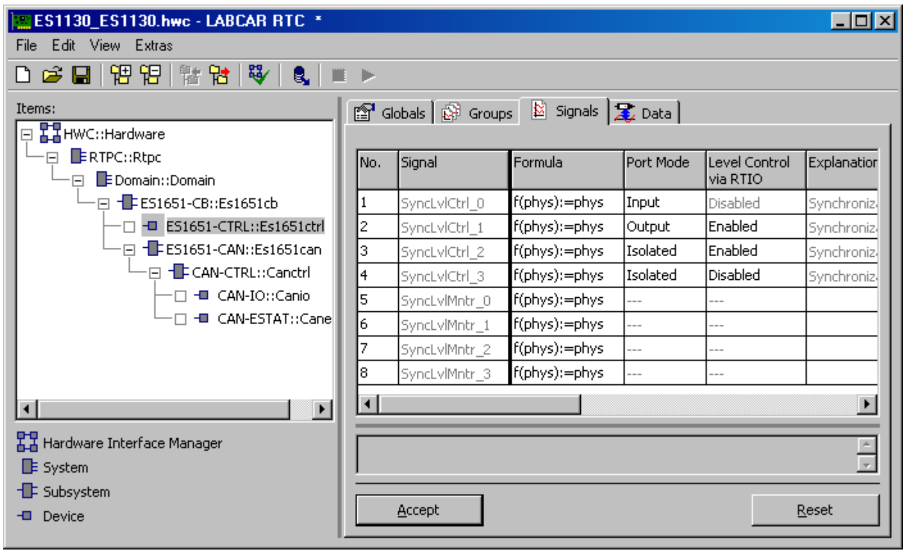


Fig. 16-7 The "Signals" Tab of the ES1651-CTRL Device

Port Mode

This option field is used to configure the sync lines with reference to the Sub-D connector on the front panel.

"Isolated" is used to isolate the sync line x (x = 0, 1, 2, 3) from the connector. "Output" configures the sync line as an output which means that synchronization signals can be made available to external hardware. Via the "Input" option, synchronization signals of the carrier board hardware or the I/O modules are made available.

Level Control via RTIO

If the sync line x (x = 0, 1, 2, 3) is configured with the "Port Mode" option "Isolated" or "Output", it is possible to make a setting in this option field. The level control of the sync line x (x = 0, 1, 2, 3) can now take place via the RTIO or the model. The relevant input field in the "Data" tab becomes editable accordingly. "0" means low level and "1" a high level on the sync line x (x = 0, 1, 2, 3).

Tab. 16-7 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Port Mode	uint8	Yes	Configures sync lines with reference to the Sub-D connector on the front panel: 0: "Isolated", sync line is isolated from the connector 1: "Output", sync signals are made available to external hardware 2: "Input", sync line is used as an input, signals are supplied
Level Control via RTIO	uint	Yes/No	Control of the level states of the sync lines using RTIO: 0: "Disabled", control is not via the RTIO 1: "Enabled", level is controlled via RTIO

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-7 RTIO Signals of the "Signals" Tab

16.2.5 Data (ES1651-CTRL Device)

Fig. 16-8 shows the "Data" tab of an ES1651-CTRL device.

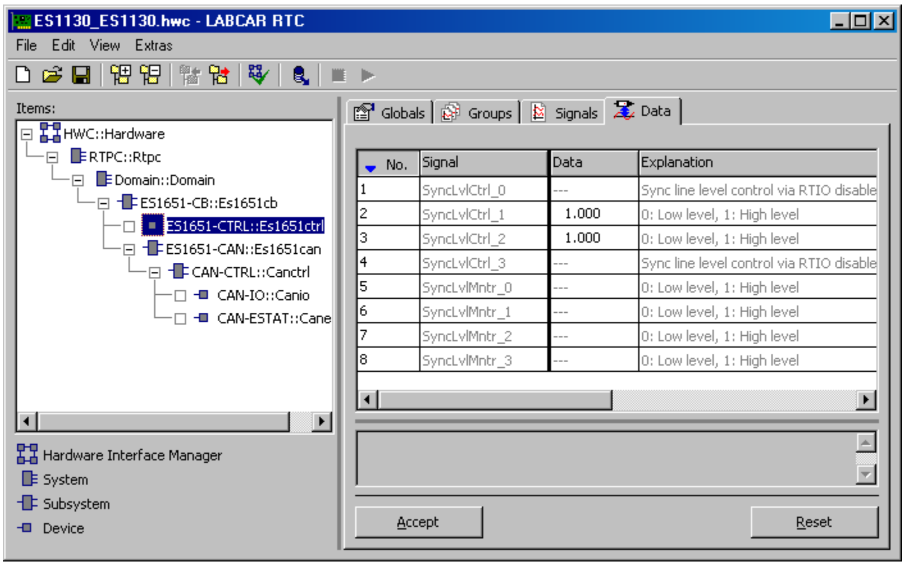


Fig. 16-8 The "Data" Tab of the ES1651-CTRL Device

You can see that the signals "SyncLvlCtrl_1" and "SyncLvlCtrl_2" are used to control the levels on the corresponding sync lines. Both levels were set to "high" active via the RTIO which is clearly shown by the level monitoring using the "SyncLvlMntr_1" and "SyncLvlMntr_2" signals.

16.3 ES1651-CAN Subsystem

16.3.1 Globals (ES1651-CAN Subsystem)

Fig. 16-9 shows the "Globals" tab of an ES1651-CAN subsystem.

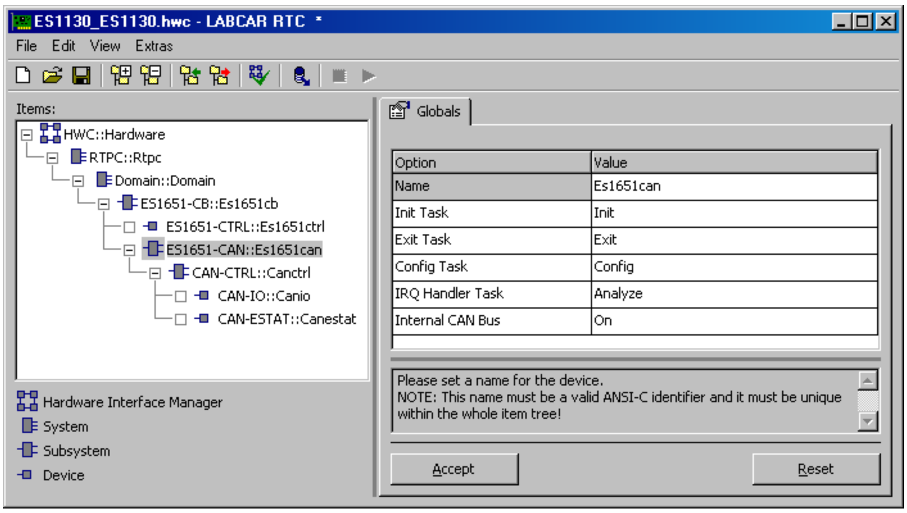


Fig. 16-9 The "Globals" Tab of the ES1651-CAN Subsystem

IRQ Handler Task

An additional task is required to support interrupts of the ES1651.1 Carrier Board. This task has to be created as a software task in the task list in the project editor. The number of possible task activations has to be between 2 and 255.

Internal CAN Bus

This option field is used to connect the two CAN controllers with each other internally. "Off" means the two CAN controllers are not connected and "On" indicates the hardware connection of the CAN controllers.

Tab. 16-8 summarizes the properties of the configuration parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Internal CAN Bus	bool	Yes	Configuration of the CAN controllers 0: "Off", CAN controller disconnected 1: "On", CAN controller connected "on board"
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-8 ES1651-CAN Parameters of the "Globals" Tab

16.4 CAN-CTRL Subsystem

16.4.1 Globals (CAN-CTRL Subsystem)

Fig. 16-10 shows the "Globals" tab of a CAN-CTRL subsystem.

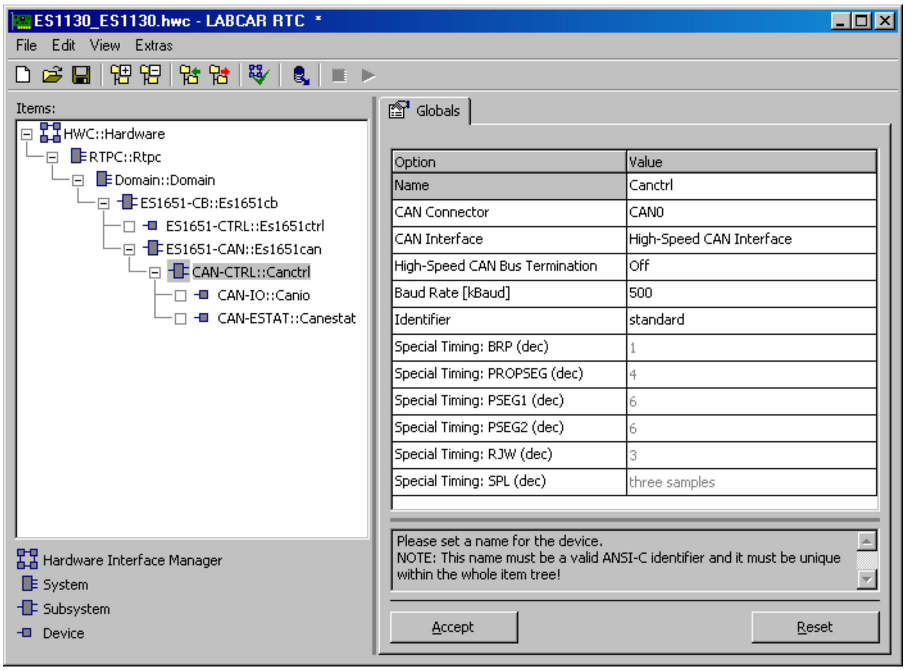


Fig. 16-10 The "Globals" Tab of the CAN-CTRL Subsystem

CAN Connector

This option field is used to assign the CAN-CTRL subsystem to a CAN controller or CAN connector on the front panel of the ES1651.1. Please note that a CAN-CTRL subsystem can only be assigned to a single CAN controller.

CAN Interface

This option field enables you to toggle between "High-Speed" and "Low-Speed" CAN functionality. The "High-Speed CAN Interface" option defines the CAN controller as a high-speed interface whilst "Fault-Tolerant CAN Interface" defines the CAN controller as a low-speed interface.

High-Speed CAN Bus

This option is used to activate a 120 Ohm terminating resistor for the high-speed CAN interface. This option is only active with the "High-Speed CAN Interface" configuration and can only be edited online. "On" means "termination active" and "Off" stands for "termination inactive".

Baud Rate [kBaud]

The transfer rate of the CAN controller is specified in this option field. The standard baud rates (1000, 500, 250, 125, 100, 50, 20, 10 kBaud) are available. Using the "Special Timing" setting, the low-level control of the CAN controller is activated with reference to bit timing and baud rate.

During the configuration of "Baud Rate", the "Special Timing" option fields are automatically configured with values with reference to the transfer rate set and transferred to the firmware.

Special Timing: BRP (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Baud Rate Prescaler" which determines the baud rate from the input clock of the CAN controller. The value range of this setting is 0 - 255.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: PROPSEG (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Propagation Segment Time" within the bit timing. The value range of this setting is 0 - 7.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: PSEG1 (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set "Phase Segment 1" within the bit timing. The value range of this setting is 0 - 7.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: PSEG2 (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set "Phase Segment 2" within the bit timing. The value range of this setting is 0 - 7.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: RJW (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Resynchronization Jump Width" within the bit timing. The value range of this setting is 0 - 3.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: SPL (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Sampling Mode" and determines how often the signal is sampled to determine the logical state. The possible settings are 1 (sampled once) or 3 (sampled three times).

For more information on this setting, refer to the CAN controller data sheet.

The formula for calculating the CAN bus baud rate is:

Baud Rate = 20 MHz / [(BRP + 1) x (4 + PROPSEG + TSEG1 + TSEG2)]

Identifier

You can select the length of the Identifier field of the CAN message in this line. With CAN messages, you can always choose between "standard" frames with 11-bit identifiers or "extended" frames with 29-bit identifiers.

Note

The setting of the length of the identifier must be the same per CAN controller.

Note

If the "standard" identifier is selected, only 11-bit values can be entered in the "Groups" tab of the "CAN-IO" RTIO element. If a larger value is entered, the "most significant bits" (MSB) are cut off. No warning is displayed.

Note

If the "extended" identifier is selected, only 29-bit values can be entered. If a larger value is entered, the "most significant bits" (MSB) are cut off. No warning is displayed.

Tab. 16-9 summarizes the properties of the configuration parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
CAN Connector	uint32	No	Assignment between the CAN-CTRL subsystem and CAN controller: 0: "CAN0", CAN controller 0 1: "CAN1", CAN controller 1
CAN Interface	uint8	Yes	Configuration of the CAN controller: 0: "Fault-Tolerant CAN Interface" 1: "High-Speed CAN Interface"
High-Speed CAN Bus	uint8	Yes	Activation of the terminating resistor in the case of high-speed CAN functionality: 0: "Off", termination inactive 1: "On", termination active
Identifier	uint8	No	Identifier length of the CAN message: 0: "standard", 11-bit length 1: "extended", 29-bit length
Special Timing: BPR (dec)	uint32	Yes/No	Determination of the baud rate from the input clock: Value range: 0 to 255
Special Timing: PROPSEG (dec)	uint8	Yes/No	"Propagation Segment Time" Value range: 0 to 7
Special Timing: PSEG1 (dec)	uint8	Yes/No	"Phase Segment 1" Value range: 0 to 7
Special Timing: PSEG2 (dec)	uint8	Yes/No	"Phase Segment 2" Value range: 0 to 7
Special Timing: RJW (dec)	uint8	Yes/No	"Resynchronization Jump Width" Value range: 0 to 3
Special Timing: SPL (dec)	uint8	Yes/No	Sampling rate of the signal: 0: "one sample", guarantees a higher transmission rate 1: "three samples", guarantees a better signal quality

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-9 Configuration Parameters of the "Globals" Tab

16.4.2 Globals (CAN-IO Device)

Fig. 16-11 shows the "Globals" tab of a CAN-IO device.

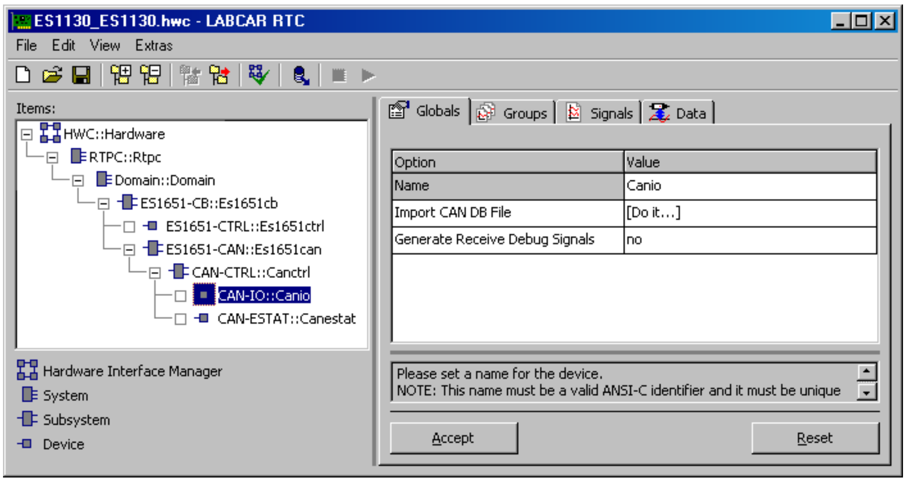


Fig. 16-11 The "Globals" Tab of the CAN-IO Device

Import CAN DB File

This option is used to read in a CAN database file which was created with the CANdb data management program made by the company Vector Informatik. This file can be used to create CAN messages and signals automatically if necessary.

The following dialog box opens when you press [Do it...]:

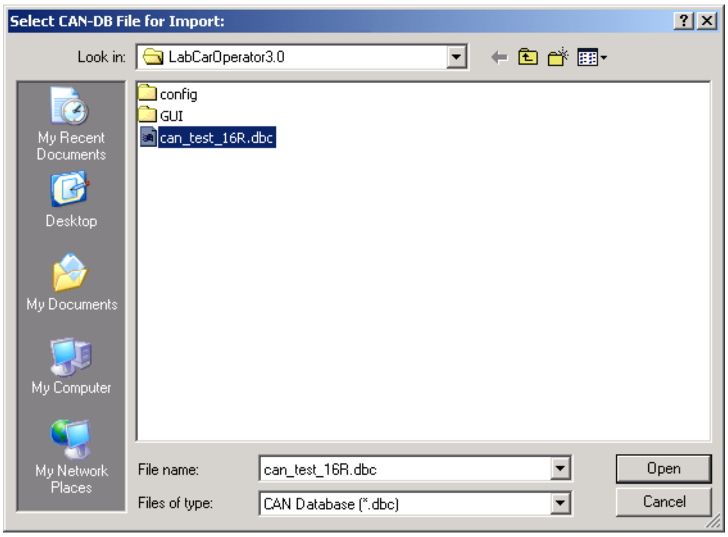


Fig. 16-12 CANdb Selection

The CAN DB file you want to import can be selected in this dialog box.

After pressing the **Open** button, the following dialog box opens in which you can specify the subsequent procedure:

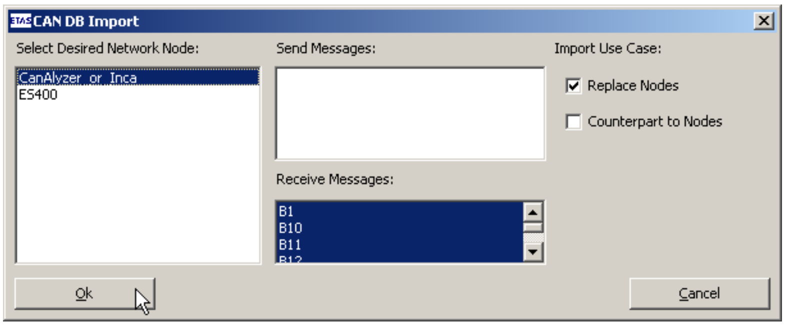


Fig. 16-13 CANdb Import Dialog

Several nodes of a CAN network are normally described in a CAN DB file. All existing nodes are shown in the left-hand list (Network Nodes). The two lists to the right of this (send messages/receive messages) list all the CAN messages defined for the node currently selected. The messages selected are used for the import. The "Import Use Case" can also be selected. The "Replace Nodes" option means that the CAN-IO device assumes the role of the network node, i.e. a send message of the node is also implemented as a send message of the CAN-IO device etc. The "Counterpart to Nodes" option means that the CAN-IO device is the counterpart to the network node, i.e. a send message of the node triggers a receive message etc.

After pressing the **OK** button, the data to be imported (CAN messages and signals) is checked together with the available signal groups and signals. The result of this check is then displayed in a dialog box:

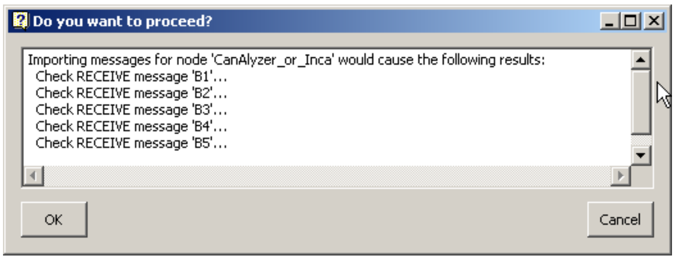


Fig. 16-14 CANdb Import Check

So far nothing has been modified in the available CAN-IO device. The procedure is not started until the **OK** button is pressed. The import procedure adds imported messages to the signal groups and also ensures the definition of the relevant signals.

Once the import procedure has been completed, the detailed protocol of the procedure is displayed in the "Monitor" window:

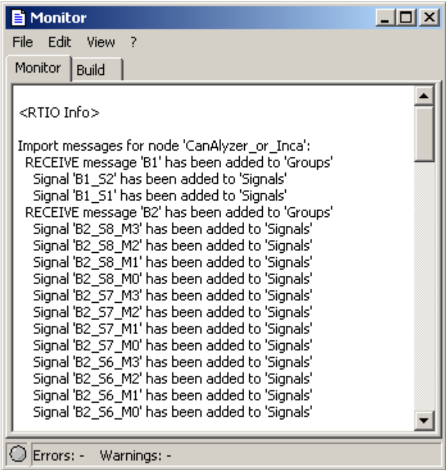


Fig. 16-15 CANdb Import Protocol

Note

When a CAN DB file is imported, only 29-bit ("extended" identifier) and 11-bit ("standard" identifier) identifiers are automatically added to the Identifier field. If the "standard" identifier was selected in the CAN-CTRL subsystem but the CAN DB file contains signals with 29-bit identifiers (ID > 231), the following takes place:

- *11 bits (bits [28...18]) are automatically written to the Identifier field. The other bits (MSB) are rejected.*
- *Warnings are shown in the monitor window. As only one identifier can be selected, there are conflicts with CAN DB files which contain both "standard" and "extended" identifiers.*

Generate Receive Debug Signals

If this option is activated (= yes), two additional signals are generated for every "receive" signal group.

- <GroupName>_Diag_dT
- <GroupName>_Diag_Rec

The "...dT" signal specifies the difference in seconds to the message received previously.

Note

This signal can be used when a normal message is received (IRQ = no) to monitor receipt. If, for example, the CANbus is interrupted, the value increases permanently in the raster of the receive task. For performance reasons, this value is not protected against overflow which occurs at approx. 300 s!

Unlike this, however, no real receipt monitoring can take place when a message is received with interrupt (IRQ = yes) as the value is calculated in the interrupt task. As there is no calculation if there is no receipt, the old value remains frozen. When an interrupt is received, there can be no distinction as to whether the receipt is interrupted or takes place absolutely regularly.

The "...Rec" signal is written as "true" each time a message is received. If this signal is mapped to a send-receive message and this is reset to "false" by the application each time it is read, it is very easy for the application to determine whether a message has been received between the processing cycles.

16.4.3 Groups (CAN-IO Device)

The CAN messages are specified in the form of signal groups in this tab.

Note

You can create new signal groups or CAN messages via the shortcut menu in the "Groups" tab.

Note

If the name of the signal group or signal, or the direction of the signal ("Direction") is changed, any message assigned may not be assigned automatically any more and may therefore have to be reassigned manually.

Fig. 16-16 shows the "Globals" tab of a CAN-IO device.

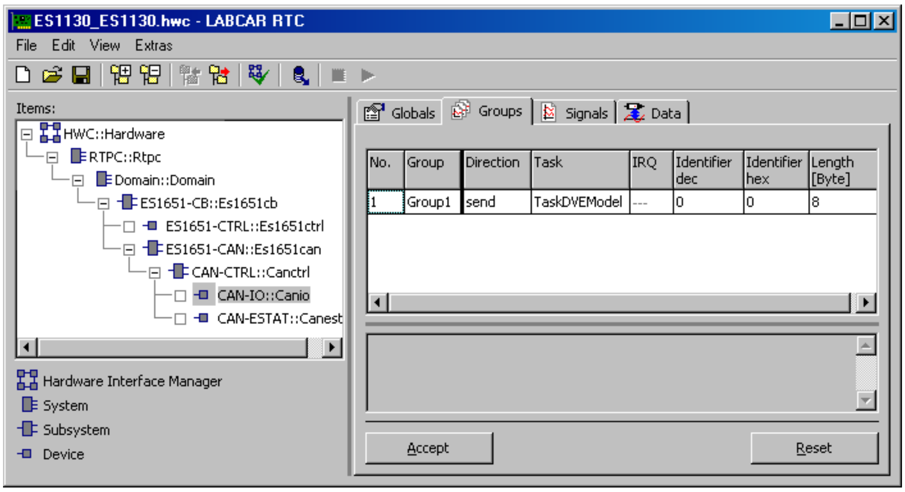


Fig. 16-16 The "Groups" Tab of the CAN-IO Device

16.4.4 The CAN Messages as Signal Groups

Direction

This is where you can determine the direction of the CAN message ("send" = send message, "receive" = receive message).

Task

This is where the task is specified in which the message is to be sent or received. If a receive message is to be received in interrupt mode, this setting is reset and locked.

IRQ

This option specifies whether the relevant receive message is to be received in interrupt mode or not. For "normal" receipt, the CAN message has to be "polled" within a task as often as it can be sent by the counterpart.

CAN messages which are not sent in any fixed raster or only occur sporadically are problematic in this mode. Interrupt receipt is ideal for these messages as it triggers message processing exactly when the message is received.

Identifier (dec / hex)

The message identifier has to be entered here. The value can vary in size depending on the selected identifier type setting in the superior CAN-CTRL subsystem ("standard" or "extended"):

Each signal group or CAN message has to have a different identifier.

- "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]

Determines the number of useful data bytes the relevant CAN message can transfer (1 to 8 bytes).

Activated Task

For "receive" CAN messages1 (signal groups), this is where you can specify a software task which is always activated when the relevant signal group is received and the receive task run. The task entered here can be used, for example, for post-processing. This column is masked out by default.

Prescaler

With a message received in interrupt mode, this is where you can set from which point exactly a VMEbus interrupt is triggered to transfer the data. The default setting "1" indicates that data transfer takes place with every received message. If this value is increased, for example, to "2", a VMEbus interrupt is only triggered with every second message received and then transfers the data of the message received last. This column is masked out by default.

Note

It might be useful to increase this value if the relevant message is received in very quick succession thus increasing the VMEbus interrupt too much.

Tab. 16-10 summarizes the properties of the configuration parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Direction	uint8	No	Transfer direction of the CAN message: 0x0: "send", sender 0x1: "receive", receiver 0x80: "irq", receiver using interrupt
Identifier	uint32	No	Identifier of the CAN message: "standard": 11-bit "extended": 29-bit
Length [Byte]	uint8	No	Specification of the useful data length of the CAN message: Value range: 1 to 8
Prescaler	uint16	No	Number of CAN messages for interrupt generation: Value range: 1 to 100

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-10 Configuration Parameters of the "Groups" Tab

16.4.5 Signals (CAN-IO Device)

The CAN messages are further specified in the "Signals" tab.

Note

New CAN signals can be created using the shortcut menu in the "Signals" tab.

Fig. 16-17 shows the "Signals" tab of the CAN-IO device.

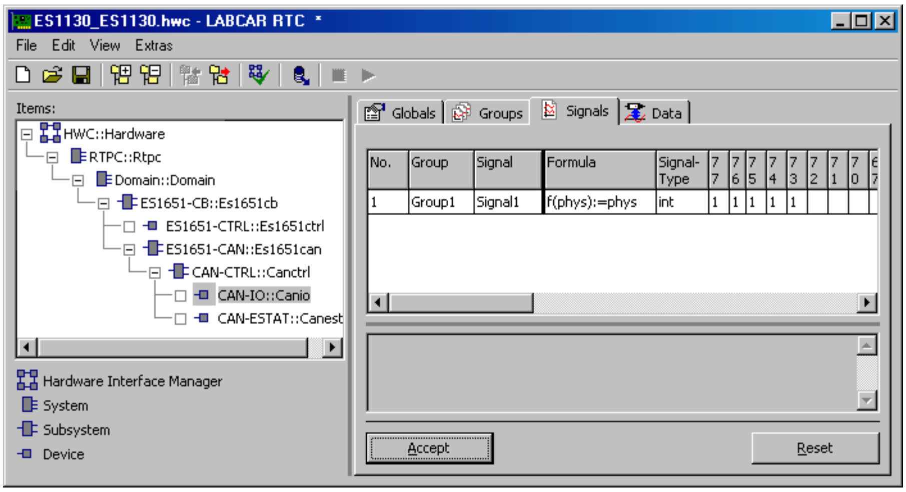


Fig. 16-17 The "Signals" Tab of the CAN-IO Device

Group

This is where a signal is assigned to the required signal group.

Signal Type

This is where the signal type is determined how the signal is transferred via the CAN bus.

Tab. 16-11 shows an overview of the possible signal types:

Signal Type	Data Type
int	Denotes a signed signal in the default complement to two data format (max. 32-bit)
(s) int	Denotes a signed signal in which the sign and then the absolute value of the signal are transferred as the most significant bit. If the sign bit is set, this is a negative integer (max. 32-bit)
uint	Denotes an unsigned signal (max. 32-bit)
bool	Denotes a Boolean signal. Only one single bit can be marked in the bit matrix.
real	Denotes a floating point value in the "Standard IEEE Float (4 Byte)" format. Only 32 bits can be marked in the bit matrix for this. Structure of the data type in acc. with IEEE: - Sign: 1-bit - Exponent: 8-bit - Fraction: 24-bit

Tab. 16-11 Signal Types

Bit Matrix

A CAN message can transfer up to 8 data bytes. In a bit matrix you can specify for each signal which bits this signal requires or occupies.

The structure of the columns is as follows:

7	7	...	0	0	Byte Number
7	6		1	0	Bit Number

Meaning of the Bit Fields:

- Empty field

  The relevant signal does not use the bit
- Occupied Field

  The signal requires this bit at this position
- "x" field

  The relevant bit is not available for data transfer because the signal has fewer useful data bytes (see the "Length" setting in the "Groups" tab).

Operating the Bit Fields:

The required bit cell can be selected using the cursor keys of the keyboard. You can "toggle" the cell between "unoccupied" (cell empty) and "1" (bit occupied) using the space bar.

Clicking with the mouse also "toggles" the relevant cell between "unoccupied" and "1". If you also hold down the <Alt> key while clicking, the value is incremented from "1" to "9" (block building).

You can select several bits at the same time as follows:

1. Select the first bit

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																															1

2. Select the second bit

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																							1								1

3. Hold down the <Shift> key and at the same time select the second bit again – the entire block between them is then selected.

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																							1	1	1	1	1	1	1	1	1

Data blocks with which virtually every transferred signal can be described can be built using the different numbers ("1111 2222..."). The numbers from which the data blocks are built have the significance that the block with the highest number ("2222") specifies the block which contains the most significant bits on transfer. The block with the smallest number ("1111") contains the least significant bits on transfer.

A signal can therefore be described with up to 9 bit blocks with the numbers available (1...9). As the representation of the bits is exactly the form in which an Intel signal is transferred, block building for the representation of signals in Motorola format is necessary as soon as the signal exceeds a length of 8 bits.

Examples of the Definition of Different Signals :

16 bit signal in Intel format:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Different description of the same signal:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1

12 bit signal in Intel format with a gap:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1												2	2	2	2									1	1	1	1	1	1	1	1

16 bit signal in Motorola format:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2

16.4.6 Data (CAN-IO Device)

Fig. 16-18 shows the "Data" tab of a CAN-IO device.

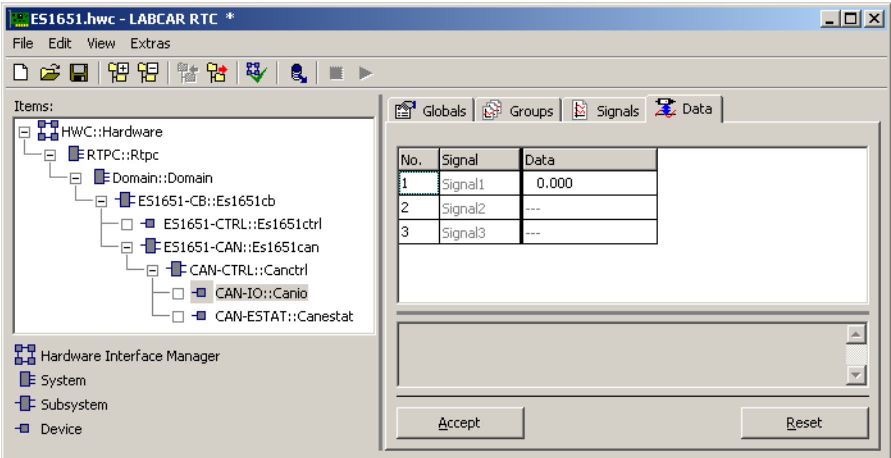


Fig. 16-18 The "Data" Tab of the CAN-IO Device

In the "Data" tab (Fig. 16-18 - "offline" mode), you can see that the signal "Signal1" is used to send a CAN message. In the "Data" field, the useful data can be occupied in "online" mode. In "offline" mode, the "Data" fields of the receive messages "Signal2" and "Signal3" are not active; in "online" mode, the received useful data of the relevant CAN messages are displayed there.

16.4.7 Globals (CAN-ESTAT Device)

Fig. 16-19 shows the "Globals" tab of a CAN-ESTAT RTIO element.

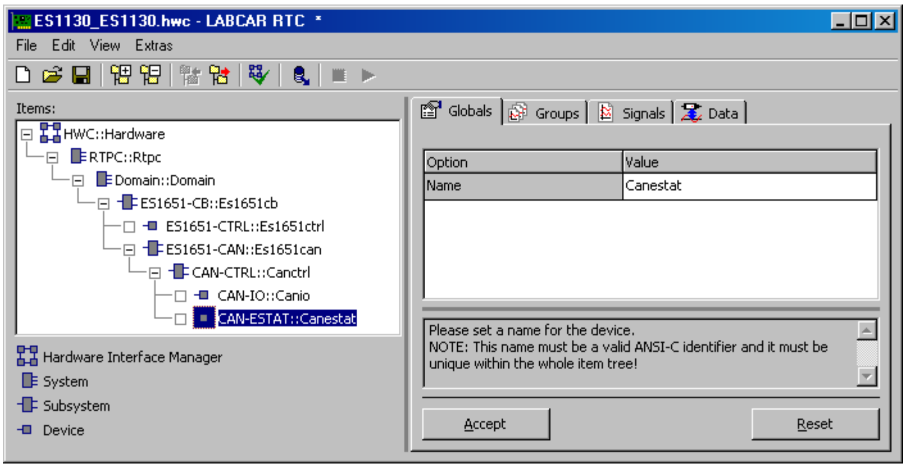


Fig. 16-19 The "Globals" Tab of the CAN-ESTAT Device

16.4.8 Groups (CAN-ESTAT Device)

The CAN-ESTAT device has a signal group which is transferred from the ES1651.1 Carrier Board to the experimental target (see Fig. 16-20). The "EstatIn" signal group transports status information and error statuses of the individual CAN controllers to the RTIO.

One task of the real-time operating system must be assigned to the signal group. A task with periodic activation is usually selected. The activation period depends on the dynamic behavior of the synchronization signals to be generated or monitored. Useful activation periods are limited downwards by runtimes of the firmware on the carrier boards to 500 milliseconds.

Fig. 16-20 shows the "Groups" tab of a CAN-ESTAT device.

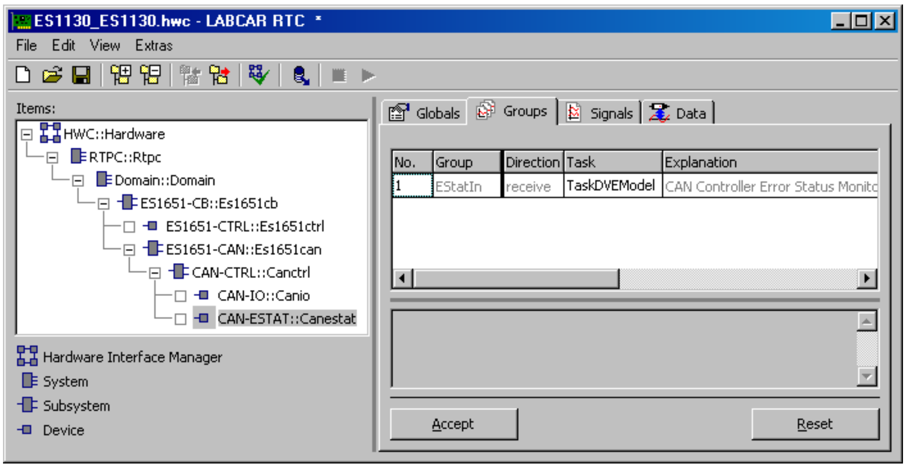


Fig. 16-20 The "Groups" Tab of the CAN-ESTAT Device

16.4.9 The Signals of the "EstatIn" Signal Group

The "EstatIn" signal group has eight RTIO signals, "EStatIn_0" to "EStatIn_7", which are of the data type "bool" (signals 0..5) or "uint8" (signals 6 and 7). No further configuration is necessary for the monitoring of the two CAN controllers. As soon as a task of the real-time operating system is assigned to the signal group, the state and error information are transferred to the RTIO in "online" mode.

Tab. 16-12 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
EstatIn_0	bool	No	TXWARN: Reflects the state of the send unit of the relevant CAN controller: 0: error counter < 96 1: error counter >= 96
EstatIn_1	bool	No	RXWARN: Reflects the state of the receive unit of the relevant CAN controller: 0: error counter < 96 1: error counter >= 96
EstatIn_2	bool	No	IDLE: reflects activities on the CAN bus: 0: Activities on the CAN bus 1: No activities on the CAN bus
EstatIn_3	bool	No	FCS: State of the CAN controller when an error occurs: 0: CAN controller is not in "error active" state 1: CAN controller is in "error active" state
EstatIn_4	bool	No	FCS: State of the CAN controller when an error occurs: 0: CAN controller is not in "error passive" state 1: CAN controller is in "error passive" state
EstatIn_5	bool	No	BOFFINT: This flag indicates the request of an interrupt when the CAN controller switches to "bus off" state: 0: No interrupt request 1: Interrupt request
EstatIn_6	uint8	No	RXECTR: Error counter of the receiver unit of the CAN controller
EstatIn_7	uint8	No	TXECTR: Error counter of the send unit of the CAN controller

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-12 RTIO Signals of the "Signals" Tab

16.4.10 Signals (CAN-ESTAT Device)

Fig. 16-21 shows the "Signals" tab of a CAN-ESTAT device.

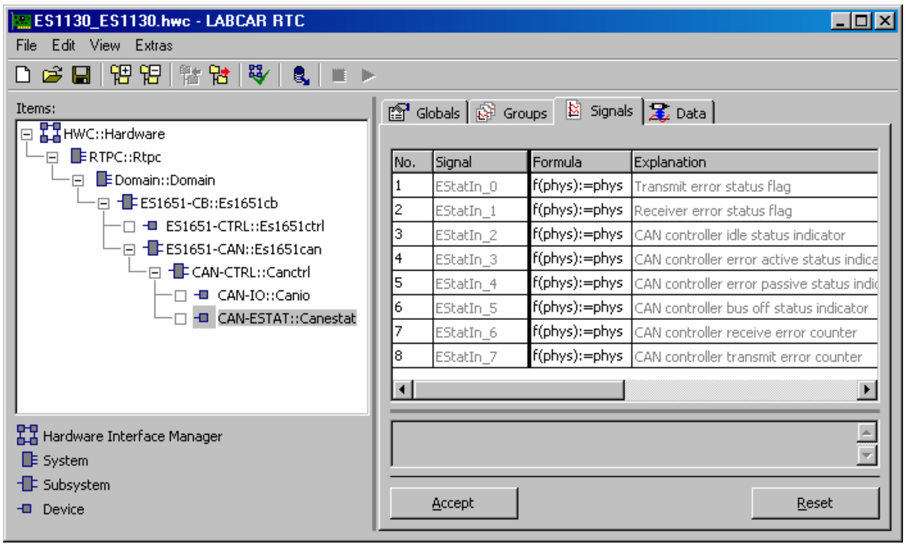


Fig. 16-21 The "Signals" Tab of the CAN-ESTAT Device

16.4.11 Data (CAN-ESTAT Device)

Fig. 16-22 shows the "Data" tab of a CAN-ESTAT device.

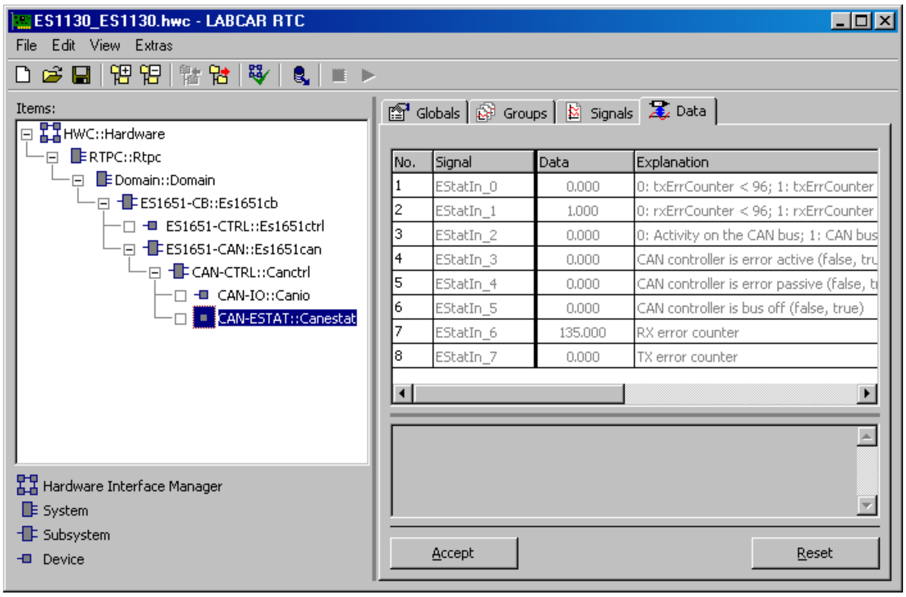


Fig. 16-22 The "Data" Tab of the CAN-ESTAT Device

The above figure ("online" mode) shows a CAN controller configured as a receiver in the case of an error. 135 receive errors were logged and the "RXWARN" flag was accordingly set to "true". No activities were registered on the CAN bus at the moment of recording.

16.5 PB1651ADC1

The PB1651ADC1 Module is used to acquire analog input signals such as engine temperature, accelerator position and other analog output values of the ECU.

A PB1651ADC1 Module is integrated in the VMEbus chassis on an ES1651.1 Carrier Board or in the VXI chassis on an ES4350 Carrier Board. The hardware connection described is reflected in the RTIO tree (Fig. 16-23). The PB1651ADC1 RTIO element of a PB1651ADC1 Module is allocated directly to the ES1651-CB or ES4350-CB RTIO element.

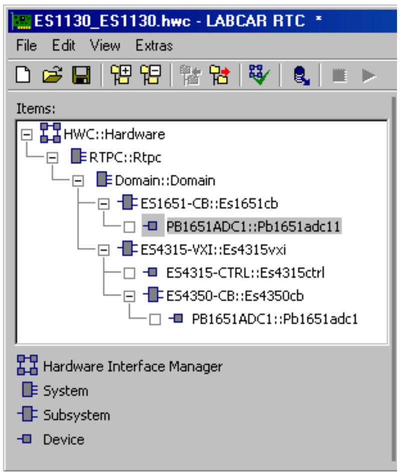


Fig. 16-23 RTIO Hardware Description with PB1651ADC1 Module

The PB1651ADC1 Module has 16 input channels to acquire analog input signals. These input channels are configured using the PB1651ADC1 RTIO element. There are two operation modes available for data acquisition. "Analog" provides the input values as a voltage value in the range 0 to 10 V, whereas "Comparator" compares voltages using a reference voltage to be specified.

16.5.1 Globals (PB1651ADC1 Device)

This section shows the global options of the PB1651ADC1 Device.

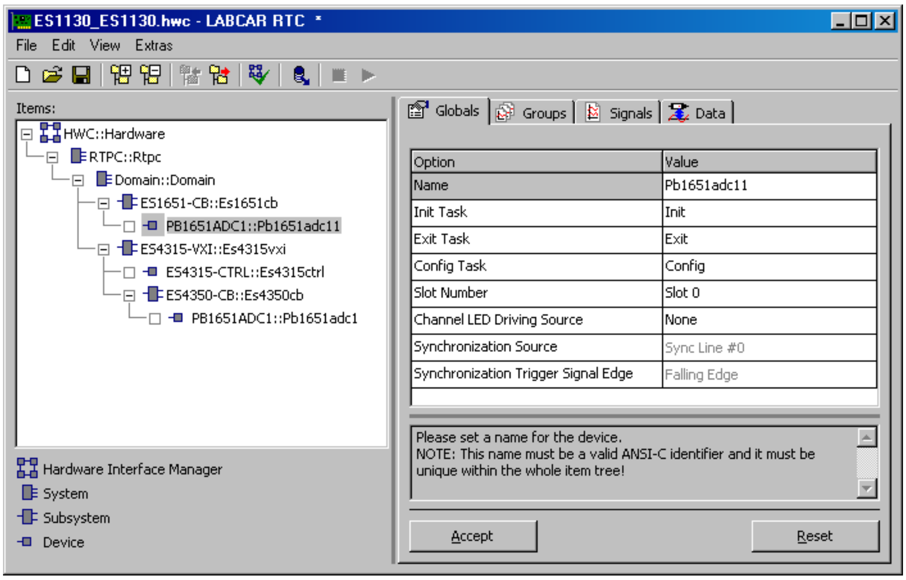


Fig. 16-24 The "Globals" Tab of the PB1651ADC1 Device

Slot Number

The number of the slot of an ES1651.1 Carrier Board or ES4350 Carrier Board holding the PB165ADC1 Module is entered in this list field. ES4350 Carrier Boards have six module slots the numbers of which are shown on the front panel. The numbering of the four slots of an ES1651.1 Carrier Board is shown in Fig. 16-25.

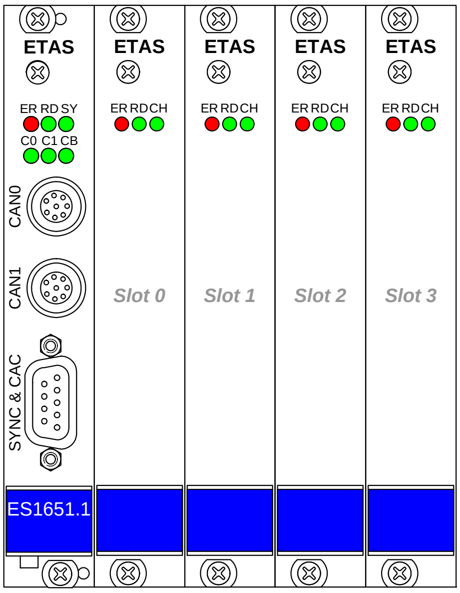


Fig. 16-25 Numbering of the Slots of an ES1651.1 Carrier Board

Channel LED Driving Source

This option field is where you specify the driving source of the "CH" LED on the PB1651ADC1 Module. If "RTIO" is set, the "CH" LED can be driven via the RTIO or via the simulation model.

If the "Comparator" operation mode is selected for an input measure value, the "CH" LED can be used to show the result of the comparison visually. If the input voltage is lower than the reference voltage set, the "CH" LED is off; if the input voltage is higher than the reference voltage, the LED is activated.

This function is only activated if the "Comparator" operation was activated for the relevant input channel. If this is not the case, an error message is issued (see Fig. 16-26).

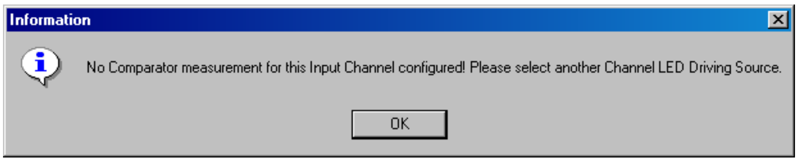


Fig. 16-26 Incorrect Configuration of the "CH" LED

Synchronization Source

This option is used to synchronize the analog input channels to an external event. The list field can only be edited if the "Hardware Triggered" option was selected for an input channel in the "Synchronization Mode" list field (in the "Signals" tab).

This is where the synchronization bus line is set, whose active edges trigger an update of the voltages on the channels of a PB1651ADC1 Module. The synchronization bus of an ES1651.1 Carrier Board has four lines; that of an ES4350 Carrier Board has six. The list field can be edited online.

Synchronization Trigger Signal Edge

This list field is where the active edge (rising or falling) is set, which triggers an update of the voltages on the channels of a PB1651ADC1 Module. As with the "Synchronization Source" list field, this list field can only be edited if if the "Hardware Triggered" option was selected for an input channel in the "Synchronization Mode" list field (in the "Signals" tab). The list field can be edited online.

Tab. 16-13 summarizes the properties of the configuration parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Slot Number	uint32	No	Slot number of the PB1651ADC1 Module on the carrier boards: ES1651: 0, 1, 2, or 3 ES4350: 0, 1, ... 5
Channel LED Driving Source	sint8	Yes	Driving source for "CH"-LED -2: "None" (no driving source) -1: "RTIO" ** 0: "Input Channel #0" ... 15: "Input Channel #15"
Synchronization Source	uint8	Yes/no	Synchronization line: 0: Sync line 0 1: Sync line 1 2: Sync line 2 3: Sync line 3 4: Sync line 4 *** 5: Sync line 5 ***
Synchronization Trigger Signal Edge	bool	Yes/no	Synchronization trigger signal edge: 0: Falling edge 1: Rising edge

* Data type which the RTIO driver uses internally for the parameter
** If the "CH" LED is driven via the RTIO, the LED is active when "1" is sent and inactive with "0"S
*** Sync lines 4 and 5 only with the ES4350 Carrier Board

Tab. 16-13 PB1651ADC1 Device: Configuration Parameters of the "Globals" Tab

Hidden Option Fields

Further option fields (Fig. 16-27) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting **Show all Options** from the shortcut menu. These option fields make it possible for ETAS service personnel to gain additional information on the PB1651ADC1 Module when localizing the cause of an error.

Note

They are not intended for the user and the default settings should therefore not be changed by the user.

In "Supervisor" mode, the data of the non-volatile memory (such as the calibration values of the module) is read out in the initialization phase of the RTIO driver and made visible to the user in the LABCAR-OPERATOR "Hardware Info" field.

Fig. 16-27 shows the "Globals" tab with the hidden options of a PB1651ADC1 device.

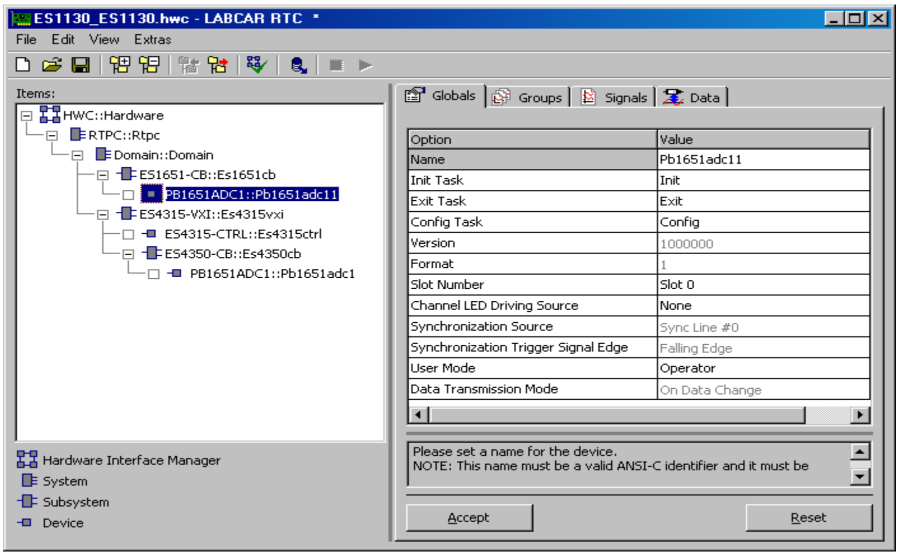


Fig. 16-27 The "Globals" Tab of the PB1651ADC1 Device with all Options Visible

User Mode

This option sets the RTIO user mode. Users should select "Operator" from the list field. "Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. The list field can be edited during runtime and is set to "Operator" by default.

Data Transmission Mode

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies when real-time data is transmitted from the experimental target to the PB1651ADC1 Module.

If the "On Data Change" option is activated, data of a signal group is only transmitted if there has been a change. If the "Every Interval" option is activated, the real-time data is transmitted at every interval of the allocated real-time task. The list field can be edited during runtime and is set to "On Data Change" by default.

Tab. 16-14 summarizes the properties of the options.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
User Mode	uint8	Yes	User mode 0: "Operator" is preset 1: "Supervisor" activates additional options for localizing errors
Data Transmission Mode	uint8	Yes	Transmission mode for real-time data ** 0: "On Data Change" 1: "Every Interval" "On Data Change" is preset.

* Data type which the RTIO driver uses internally for the parameter
** The option field can only be edited if "User Mode" = "Supervisor"

Tab. 16-14 ES1651ADC1 Device: Configuration Parameters of the "Globals" Tab

16.5.2 Groups (PB1651ADC1 Device)

The PB1651ADC1 device has two signal groups which are transmitted from the experimental target to the module via the carrier board or from the module to the experimental target via the carrier board. The "AnaIn" signal group transports 16-bit wide analog measure values from the 16 input channels from the module to the RTIO; the "LedDrv" signal group is available for driving the "CH" LED.

A task of the real-time operating system must be assigned to each signal group. A task with periodic activation is usually selected. The activation period depends on the dynamic behavior of the analog input signals to be measured or on the drive of the LED, e.g. from the simulation model. Useful activation periods are limited downwards by runtimes of the firmware on the carrier boards to 1 ms.

Fig. 16-28 shows the "Groups" tab of a PB1651ADC1 device.

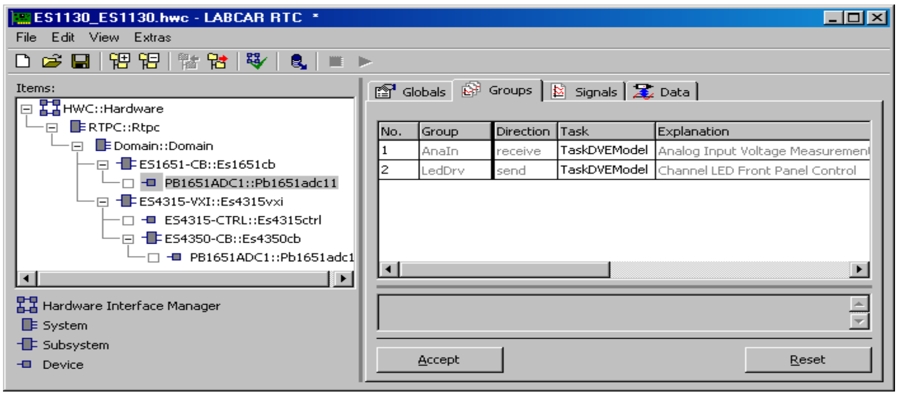


Fig. 16-28 The "Groups" Tab of the PB1651ADC Device

The RTIO Signals of the "AnaIn" Signal Group

The "AnaIn" signal group has 16 RTIO signals, "AnaIn_0" to "AnaIn_15", which are of the data type "real32". There are two measurement procedures available.

The "Analog" procedure provides analog voltage values from 0.0 to 10.0 Volt.

The "Comparator" measurement procedure compares the input voltage to a reference voltage to be specified by the user. If the input voltage is lower than the reference voltage, a "0" is issued; otherwise a "1" is issued.

The result of this measurement procedure can also be made visible via the "CH" LED. If an input value is configured with the "Comparator" procedure, this value can be used as a source for the "Channel LED Driving Source" option in the "Globals" tab.

Tab. 16-15 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
AnaIn_0 to AnaIn_15	real32	No	Data acquisition of the two measurement procedures: 0 -10 V: "Analog" measurement procedure 0 or 1: "Comparator" measurement procedure **
* Data type which the RTIO driver uses internally for the parameter ** If the "CH" LED drive is activated for a comparison operation, the LED is off when the input voltage is lower than the reference voltage; otherwise it is on			

Tab. 16-15 RTIO Signals of the "AnaIn" Signal Group

The RTIO Signals of the "LedDrv" Signal Group

The "LedDrv" signal group has one RTIO signal, "LedDrv"; it is of the data type "bool". It is used to drive the "CH" LED on the front panel of the PB1651ADC1 Module. "0" means the "LED inactive"; "1" means "LED active".

The signal can only be edited online if (in the "Globals" tab) the option "Channel LED Driving Source" is configured with the value "RTIO".

Tab. 16-16 summarizes the properties of the RTIO signal.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
LedDrv	bool	Yes/no	Drives the "CH" LED: 0: LED inactive 1: LED active
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-16 RTIO Signals of the "LedDrv" Signal Group

16.5.3 Signals (PB1651ADC1 Device)

The input channels of the PB1651ADC1 Module are configured in the "Signals" tab. All parameters of the "AnaIn" signal group can be edited online (i.e. during runtime of the model on the experimental target).

Fig. 16-29 shows the "Signals" tab of a PB1651ADC1 device.

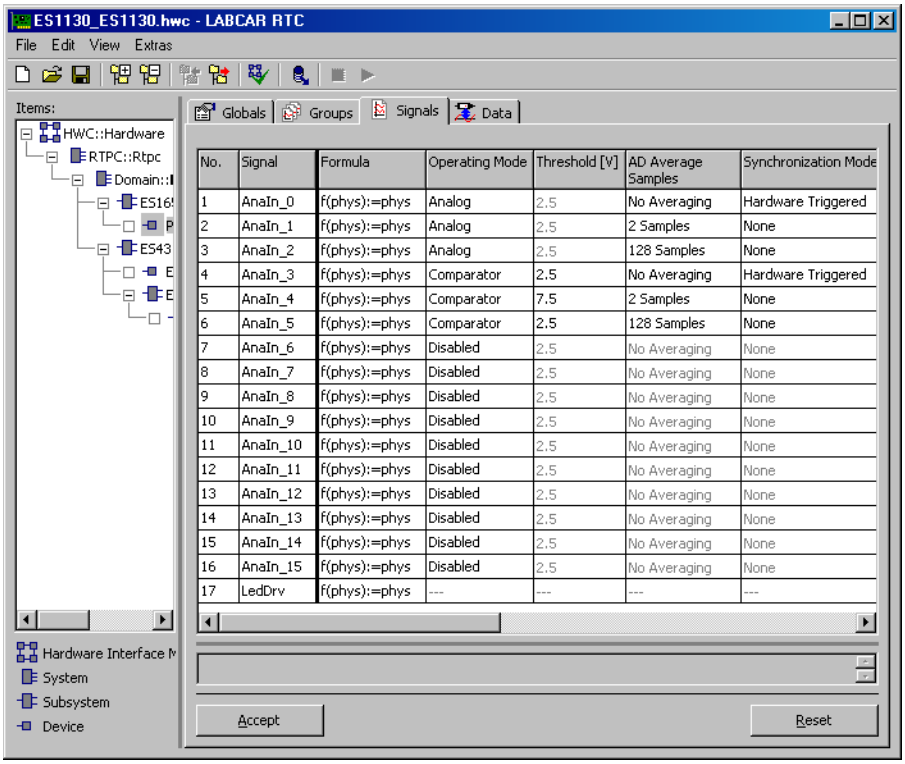


Fig. 16-29 The "Signals" Tab of the PB1651ADC1 Device

Operating Mode

Each of the 16 input channels of a PB1651ADC1 Module can be operated in three different modes which are set separately for each channel in the "Operating Mode" field:

- **Disabled**

  The input channel is deactivated.

- **Analog**

  The channel is configured for analog measurement acquisition. The relevant signal x (x = 0, 1, ...15) specifies the measured voltage in Volts.

- **Comparator**

  The channel is configured as a comparator. The relevant signal value x (x = 0, 1, ...15) is 0 if the input voltage of the channel is lower than a threshold voltage which can be set by the user; otherwise the signal value is 1.

Threshold [V]

The setting in the "Threshold [V]" field is only of any significance for the comparator mode, i.e. can be edited both "online" and "offline". This is where the threshold voltage for the comparator is specified.

AD Average Samples

This specifies how many sample values are used to determine an analog voltage value through averaging. Averaging can result in interference or noise suppression. The setting in the "AD Average Samples" field is of significance for all operating modes apart from "Disabled".

There are eight different configurations possible per channel:

- No Averaging (averaging is disabled)
- 2 Samples
- 4 Samples
- 8 Samples
- 16 Samples
- 32 Samples
- 64 Samples
- 128 Samples

Synchronization Mode

This list field determines when analog input signals from the PB1651ADC1 module are acquired by the hardware and transferred via the carrier board to the RTIO driver or to the RTIO.

If "None" is set in this list field, acquisition of input signals on the PB1651ADC1 Module for the relevant channel is activated immediately.

With the "Hardware Triggered" option, an update only takes place if an active edge is acquired on the synchronization bus line of an ES1651.1 Carrier Board or an ES4350 Carrier Board set in the "Synchronization Source" list field. In this way, the analog input can be synchronized to an external event.

The list field can be edited online (i.e. during runtime of the model on the experimental target) if the "Operating Mode" was configured with the options "Analog" or "Comparator".

Tab. 16-17 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Operating Mode	uint8	Yes	Measurement procedure for the input channels: 0: "Disabled": Channel deacti-vated 1: "Analog": Analog voltage acquisition 2: "Comparator": Comparator operation
Threshold	real32	Yes/no	Definition of the reference voltages from 0 to 10 Volts
AD Average Samples	uint8	Yes/no	Settings for averaging of each input channel: 0: "No Averaging": no averages are created 1: "2 Samples" 2: "4 Samples" 3: "8 Samples" 4: "16 Samples" 5: "32 Samples" 6: "64 Samples" 7: "128 Samples"
Synchronization Mode	uint8	Yes/no	Selection of the triggering for mea-surement acquisition: 0: "Hardware Triggered": Mea-surement acquisition is triggered at an external event (using a sync line) 1: "None": There is no external triggering

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-17 RTIO Signals of the "Signals" Tab

16.5.4 Data (PB1651ADC1 Device)

Fig. 16-30 shows the "Data" tab of a PB1651ADC1 device.

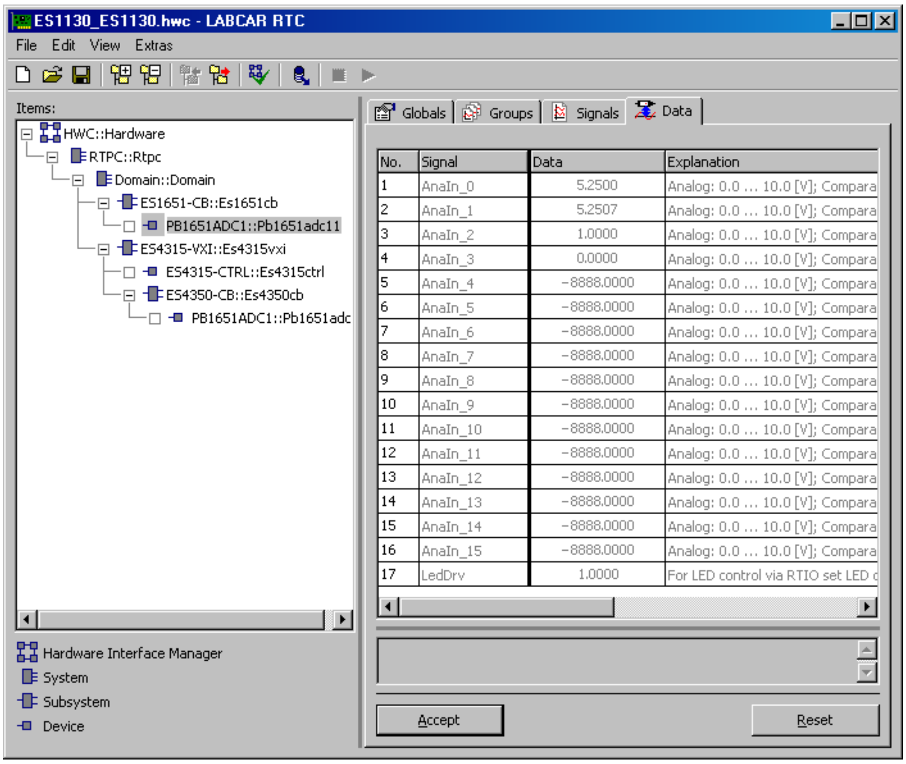


Fig. 16-30 The "Data" Tab of the PB1651ADC1 Device

The figure shows the "Data" tab in "online" mode. Measurement modes were selected for the first four input channels; the remaining channels are deactivated. Input channels 1 and 2 are in the measurement mode "Analog", i.e. analog input voltages are measured and displayed in the interval from 0 V to 10 V. The next two channels are operated in the measurement mode "Comparator", i.e. voltage is compared with a specific reference voltage. Input channel 3 shows a "1", i.e. the measured input voltage is higher than the specified reference voltage. The measure result from channel 4 is completely the opposite; the input signal is lower.

RTIO signal 17 is configured for the "CH" LED to be driven via the simulation model or an ASCET-SD message. The tab shows a "1", i.e. the "CH" LED on the front panel of the PB1651ADC1 Module is on.

16.6 PB1651PWM1

The PB1651PWM1 Module is used to issue and acquire pulse-width modulated signals. A typical example of how the module is used is, e.g., in LABCAR projects where it is used to simulate vehicle sensors and address actuators.

The PB1651PWM1 I/O Module is integrated in the RTIO-Editor by selecting the PB1651PWM1-RTIO element. This element can be allocated either to an ES1651.1 Carrier Board (VMEbus) or an ES4350 Carrier Board (VXIbus). Fig. 16-31 shows the connection of PB1651PWM1 Modules to both types of carrier board.

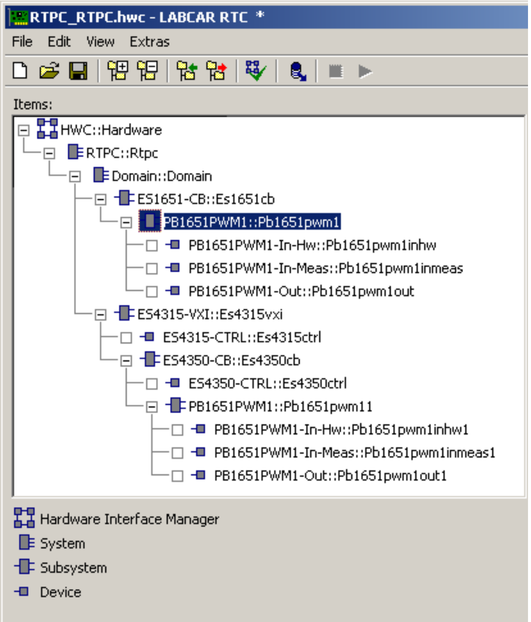


Fig. 16-31 RTIO Hardware Description with Integrated PB1651PWM1 Modules

The PB1651PWM1 I/O Module has 24 input channels for acquiring pulse-width modulated signals. These input channels are configured with the PB1651PWM1-In-Hw device.

In addition to this element, up to four PB1651PWM1-In-Meas devices and one PB1651PWM1-Out devices can be allocated to the PB1651PWM1-RTIO element.

Up to 24 measurements (frequency, pulse width, duty cycle etc) can be specified on the signals of the input channels with each of the PB1651PWM1-In-Meas devices. In total, the RTIO drivers and firmware of the PB1651PWM1 Module make 96 measurements possible.

The 16 channels of a PB1651PWM1 Module for issuing pulse-width modulated signals are configured and addressed using the PB1651PWM1-Out device.

Note

Before a PB1651PWM1-In-Meas device can be added, a PB1651PWM1-In-Hw device first has to be added to configure the input channels!

16.6.1 Globals (PB1651PWM1 Subsystem)

This section describes the global options of the PB1651PWM1 subsystem.

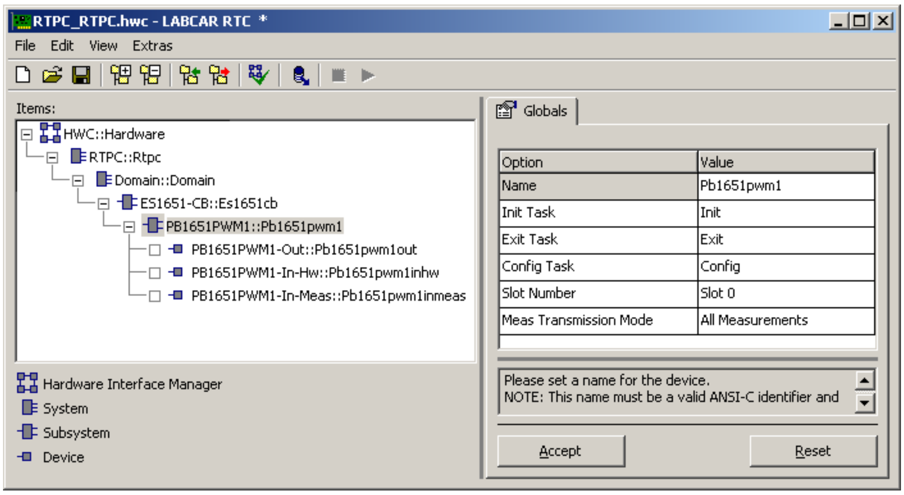


Fig. 16-32 The "Globals" Tab of the PB1651PWM1 Subsystem

Slot Number

The number of the slot of the ES1651.1 Carrier Board or ES4350 Carrier Board holding the PB1651PWM1 Module is entered in this list field. ES4350 Carrier Boards have six module slots the numbers of which are shown on the front panel. The numbering of the four slots of an ES1651.1 Carrier Board is shown in Fig. 16-33.

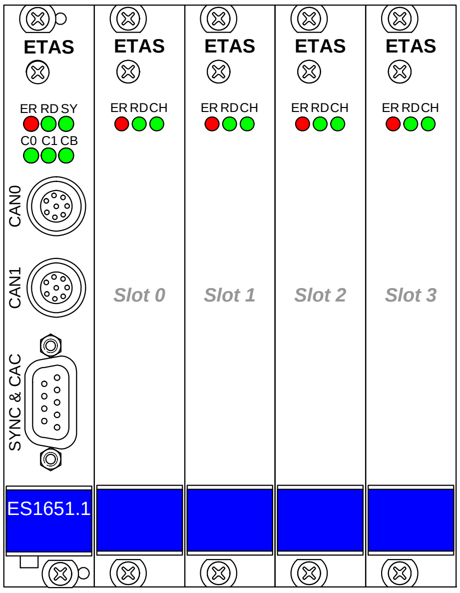


Fig. 16-33 Numbering of the Module Slots of an ES1651.1 Carrier Board

Meas Transmission Mode

This option is used to directly specify the data transfer between the I/O module and the experimental target. There are four different configurations available for signal measuring.

- All Measurements

  All measurements are transmitted to the experimental target. This means that all measurement procedures can be selected in the RTIO.

- Timestamp Measurements

  Only the timestamps are transmitted to the experimental target for this measurement procedure. This means that all measurement procedures apart from the "additive" ones can be selected in the RTIO.

- Additive Measurements

  Only the "additive" measurement procedures can be selected in this mode. There is no timeout detection available for this configuration.

- Level Measurements

  In this mode, only the signal levels of the input channels are transmitted to the experimental target. There is no timeout detection available for this configuration.

Note

The data transfer can only be configured before code is generated, i.e. this option cannot be edited online.

Note

"Timeout" detection is not available during runtime for the configurations "Additive Measurements" and "Level Measurements".

Tab. 16-18 summarizes the properties of the configuration parameters of the "Globals" tab.

Parameter or Option Field	Data-Type*	Can be Edited Online	Comment/Value Range
Slot Number	uint32	No	Slot number of the PB1651PWM1 Module on the carrier board ES1651: 0, 1, 2, or 3 ES4350: 0, 1, ... 5
Meas Transmis-sion Mode	uint8	No	Configuration of data transfer for signal measuring: 0: All measurements are trans-ferred 1: Only the timestamps are trans-ferred 2: Only the additive data is trans-ferred 3: Only the signal levels are trans-ferred

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-18 ES1651PWM1 Subsystem: Configuration Parameters of the "Glo-bals" Tab

Hidden Option Fields

Further option fields (see Fig. 16-34) can be made visible by clicking the "Glo-bals" tab with the right-hand mouse button and selecting **Show all Options** from the shortcut menu.

These option fields make it possible for ETAS service personnel to gain additional information on the PB1651PWM1 Module when localizing the cause of an error. They are not intended for the user and the default settings should therefore not be changed by the user.

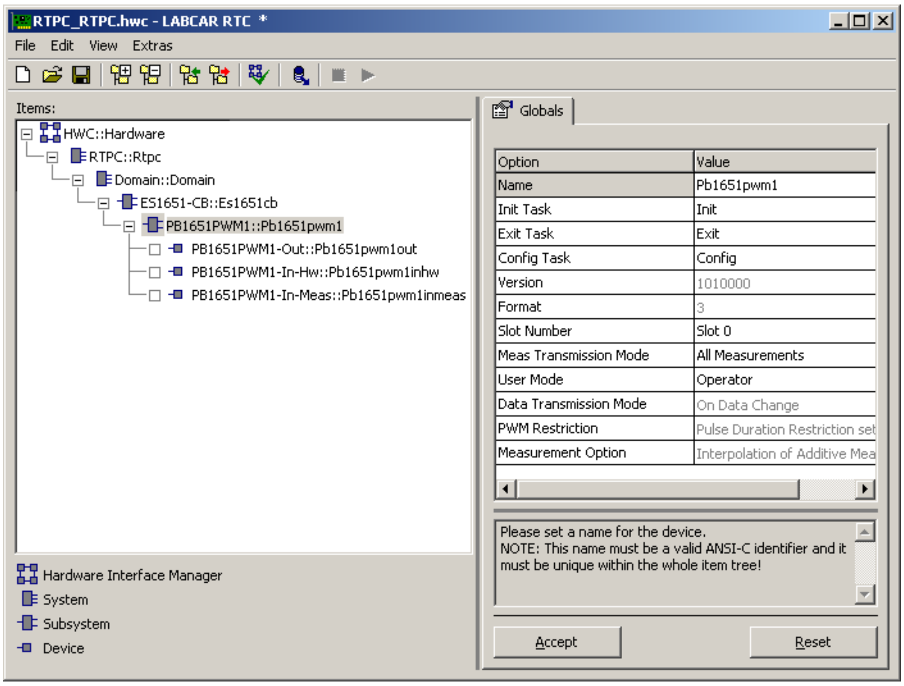


Fig. 16-34 The "Globals" Tab of the PB1651PWM1 Subsystem with all Options Visible

User Mode

This option sets the RTIO user mode. Users should select "Operator" from the list field. "Supervisor" user mode activates additional options for localizing errors and is only intended to be used by ETAS service personnel. The contents of the non-volatile memory of the PB1651PWM1 Module, for example, are output to the "Target Debug" window during initialization when "Supervisor" mode is active. The list field can be edited during runtime and is set to "Operator" by default.

Data Transmission Mode

This list field can only be edited if the "Supervisor" option is set in the "User Mode" list field. It specifies when real-time data is transmitted from the experimental target to the PB1651PWM1 Module. If the "On Data Change" option is activated, data of a signal group is only transmitted if there has been a change.

If the "Every Interval" option is activated, the real-time data is transmitted at every interval of the allocated real-time task. The list field can be edited during runtime and is set to "On Data Change" by default.

Tab. 16-19 summarizes the properties of the hidden RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data-Type	Can be Edited Online	Comment/Value Range
User Mode	uint	Yes	User mode 0: Operator (preset) 1: Supervisor (activates additional options for localizing errors)
Data Transmission Mode	uint	Yes	Transmission mode 0: On Data Change (preset) 1: Every Interval The option field can only be edited if "User Mode" = "Supervisor".

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-19 ES1651PWM1 Subsystem: Configuration Parameters of the "Globals" Tab

16.6.2 The PB1651PWM1-In-Hw Device

The PB1651PWM1-In-Hw device is used to configure the 24 input channels of the PWM Module and to configure and address the "CH" LED in the front panel of the module.

16.6.3 Globals (PB1651PWM1-In-Hw Device)

Fig. 16-35 shows the "Globals" tab of the PB1651PWM1-In-Hw device.

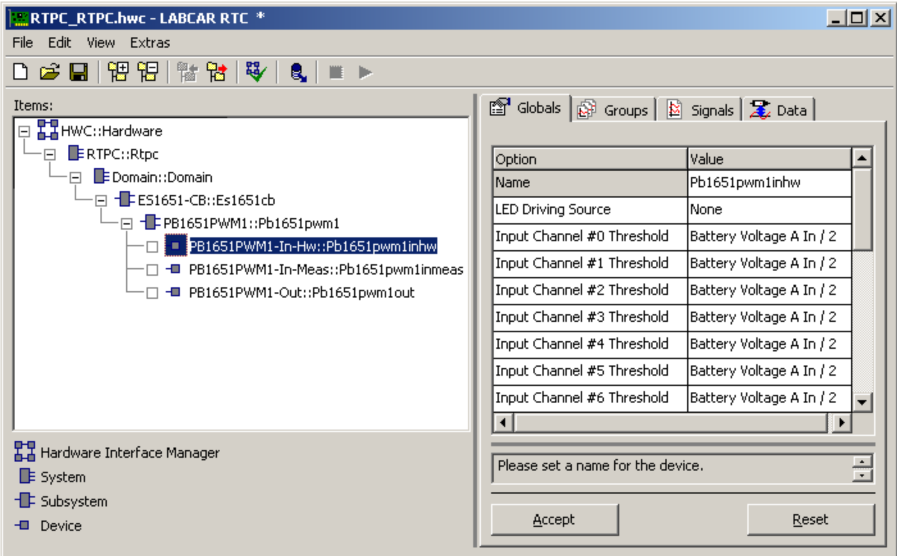


Fig. 16-35 The "Globals" Tab of the PB1651PWM1-In-Hw Device

LED Driving Source

This list field defines the driving source of the "CH" LED on the front panel of the PB1651PWM1 Module. One of the 24 input channels or "RTIO" can be set as driving source. If an input channel is set as the driving source, the LED lights up when the channel level is high and does not light up when the channel level is low.

If "RTIO" is set as the driving source, the LED can be powered on/off by the simulation model. The list field can be edited online (i.e. during runtime of the model on the experimental target).

Input Channel #x Threshold (x = 0, 1, ... 23)

The PWM signals pending at the inputs of the module are converted to TTL level by comparators in the input stages of the channels. The comparator threshold voltages of the input channels are set using these list fields. There are two voltages to choose from: "Battery Voltage A In/2" and "Battery Voltage B In/2".

The voltages "Battery Voltage A In" and "Battery Voltage B In" are supplied via the connector on the front panel of the PB1651PWM1 Module. Before they are routed to the comparators, they are divided by factor 2 with voltage divider circuits.

The list fields can be edited online (i.e. during runtime of the model on the experimental target).

Tab. 16-20 summarizes the properties of the configuration parameters of the "Globals" tab.

Parameter or Option Field	Data-Type*	Can be Edited Online	Comment/Value Range
LED Driving Source	sint8	Yes	Driving source of the "CH" LED -2: "None" (no driving source) -1: "RTIO" 0: "Input Channel #0" ... 23: "Input Channel #23"
Input Channel #x Threshold (x = 0, 1, ... 23)	uint8	Yes	Threshold voltage for input comparator 0: "Battery Voltage A In/2" 1: "Battery Voltage B In/2"

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-20 PB1651PWM1-In-Hw Device: Configuration Parameters of the "Globals" Tab

16.6.4 Groups (PB1651PWM1-In-Hw Device)

The PB1651PWM1-In-Hw device has one signal group which is transmitted from the experimental target to the PB1651PWM1 Module (see Fig. 16-36). It transports the on/off information for the "CH" LED on the front panel of the module. This information is, however, only evaluated by the hardware if "RTIO" is set in the "LED Driving Source" option field of the "Globals" tab.

A task of the real-time operating system must be allocated to the signal group. Usually, a task with periodic activation and an activation period greater than 100 milliseconds is selected.

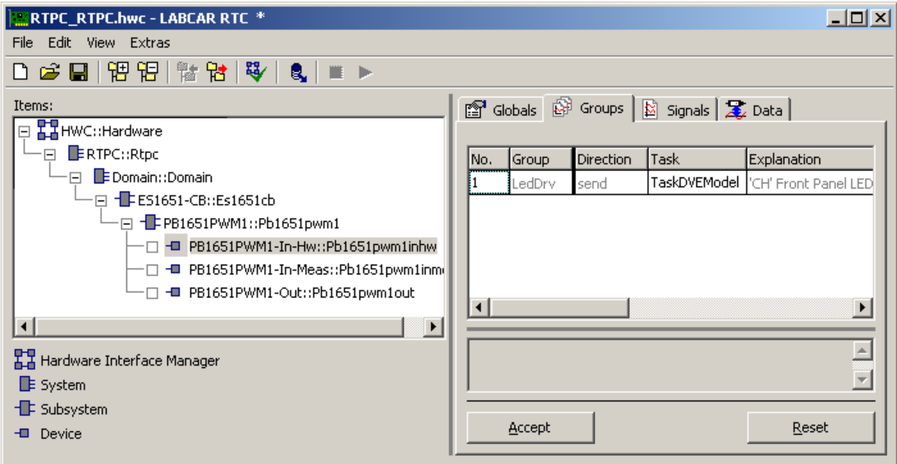


Fig. 16-36 The "Groups" Tab of the PB1651PWM1-In-Hw Device

The RTIO Signals of the "LedDrv" Signal Group

The signal group has one RTIO signal, "LedDrv"; it is of the data type "bool". It defines the switch status of the "CH" LED. The value 0 switches the LED off; a value not equal to 0 switches the LED on.

Tab. 16-21 summarizes the properties of the RTIO signal.

RTIO Signal	Data Type*	Comment/Value Range
LedDrv	bool	Driving signal of the "CH" LED 0: LED off ≠ 0: LED on
* Data type which the RTIO driver uses internally for the parameter		

Tab. 16-21 PB1651PWM1-In-Hw Device: The RTIO Signals of the "LedDrv" Signal Group

16.6.5 Signals (PB1651PWM1-In-Hw Device)

The PB1651PWM1-In-Hw device has no signal-specific options. This is why there are no settings to be made in the "Signals" tab (Fig. 16-37).

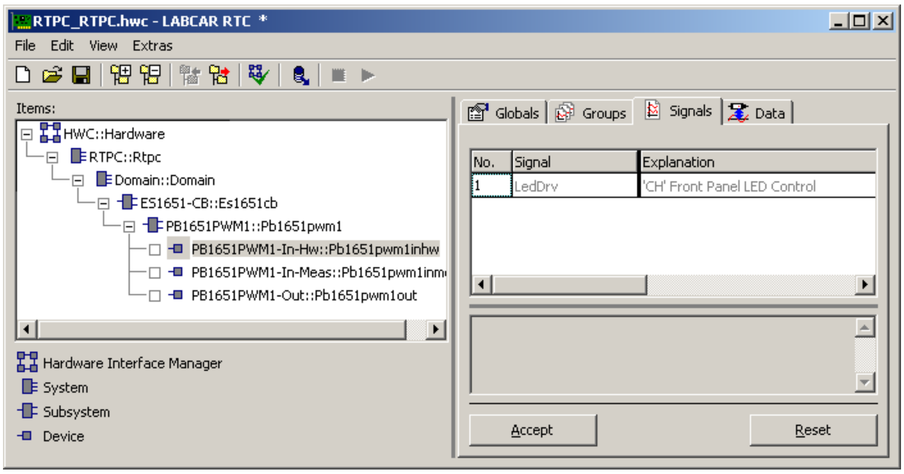


Fig. 16-37 The "Signals" Tab of the PB1651PWM1-In-Hw Device

16.6.6 The PB1651PWM1-In-Meas Device

The PB1651PWM1-In-Meas device is used to define measurements on the 24 input channels of the module. Up to four PB1651PWM1-In-Meas devices are supported per PB1651PWM1 Module. 24 measurements can be defined with every element, resulting in a possible total of 96 measurements per module.

16.6.7 Globals (PB1651PWM1-In-Meas Device)

No settings need to be made in the "Globals" tab (see Fig. 16-38). A number between 0 and 3 is allocated to the PB1651PWM1-In-Meas device by the system in the "Measurement Device ID" option field for identification purposes.

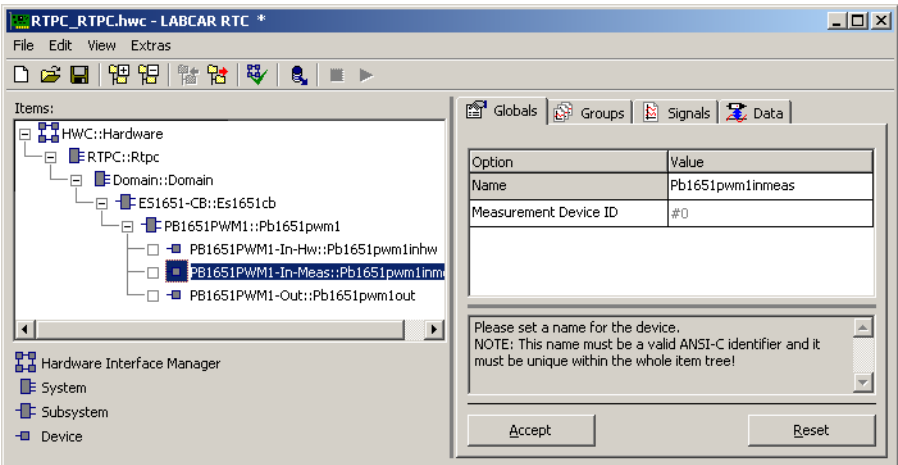


Fig. 16-38 The "Globals" Tab of the PB1651PWM1-In-Meas Device

16.6.8 Groups (PB1651PWM1-In-Meas Device)

The PB1651PWM1-In-Meas device has one signal group (see Fig. 16-39), which is transmitted from the PB1651PWM1 Module to the experimental target. It transports all measurement data, such as the actual measure values, update and timeout information.

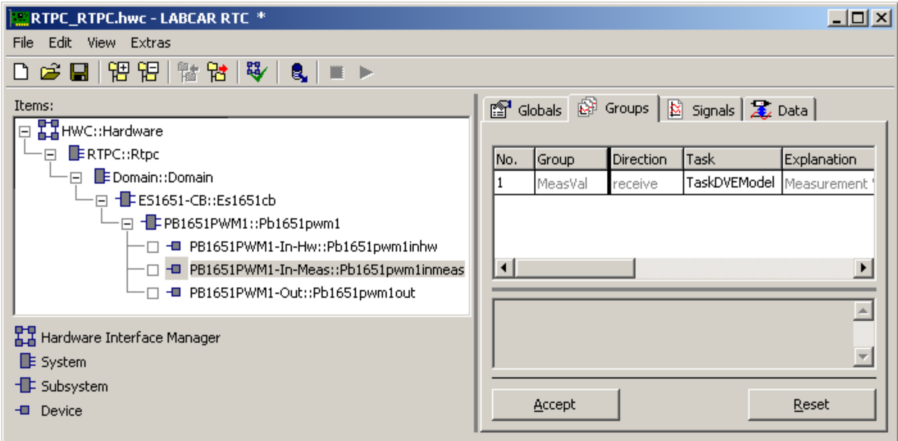


Fig. 16-39 The "Groups" Tab of the PB1651PWM1-In-Meas Device

A task of the real-time operating system must be allocated to the signal group. Usually, a task with periodic activation is selected. The activation period depends on the dynamic behavior and the period duration of the signals to be measured.

16.6.9 The RTIO Signals of the "MeasVal" Signal Group

The "MeasVal" signal group has 49 RTIO signals. The "TriggerBitField_0" signal is to be interpreted as a bit field – it has 24 bits. Update indicators for the 24 measurements are coded in this bit field, i.e. it displays which measure values were updated in the last activation of the read process for the "MeasVal" signal group. If a bit is set, the measure value of the relevant measurement has been redetermined. It does not matter whether the measure value was updated because of a timeout or a regular measure value calculation - in both cases, the indicator bit of the measure channel is set.

As data acquisition between the carrier board and the PWM Module is asynchronous to the task cycle time of data acquisition on the module, the RTIO signals "MeasCatch_0" and "MeasCatch_1" correspond to the cycle time of the value acquisition on the module.

The RTIO signals "MeasVal_0" to "MeasVal_23" contain the measure values of the 24 measurements. These are regularly acquired measure values or – in the case of a timeout – the timeout value intended here. If a measurement is not used, -8888.0 is allocated to the relevant measure value. The physical unit of the measure values depends on the measurement procedure:

- Time (additive) pulse duration measurements, period duration) is measured in microseconds
- Frequency is measured in Hertz
- All other measurements (duty cycles, pulse count, level measuring) have no dimension

The RTIO signals "Tout_0" to "Tout_23" contain the result of the timeout monitoring for the relevant measurement.

Note

When assigning the measure channels, make sure that the measure values of the 24 input channels are transferred in two blocks from the PWM Module to the carrier board (1st block: channels 0-11; 2nd block: channels 12-23).
If the number of channels used is smaller than 12, make sure that only channels within one block are used for performance reasons. This ensures that no unnecessary data is transmitted and no resources wasted.

Tab. 16-21 summarizes the properties of the RTIO signal.

RTIO Signal	Data Type*	Comment/Value Range
TriggerBitField	uint32	Bit field with update indicators for the 24 measurements Measurement 0: LSB (Least Significant Bit) Bit value 0: Measure value is unchanged Bit value 1: Measure value has been updated
MeasCatch_0 ... MeasCatch_1	real64	Cycle time of data acquisition on the PWM Module If the measurement is not used, -8888.0 is output as the value. Physical unit of the measure value: - Time is measured in microseconds
MeasVal_0 ... MeasVal_23	real64	Measure value If the measurement is not used, -8888.0 is output as the value. Physical unit of the measure value: - Time is measured in microseconds - Frequency is measured in Hertz
Tout_0 ... Tout_23	uint8	Result of timeout monitoring 0: No timeout 1: Timeout 2: Timeout monitoring is inactive

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-22 PB1651PWM1-In-Meas Device: The RTIO Signals of the "MeasVal" Signal Group

16.6.10 Signals (PB1651PWM1-In-Meas Device)

The 24 measurements of a PB1651PWM1-In-Meas device are defined in the "Signals" tab. Fig. 16-40 shows the option fields of the "Signals" tab. All option fields can be edited online (i.e. during runtime of the model on the experimental target).

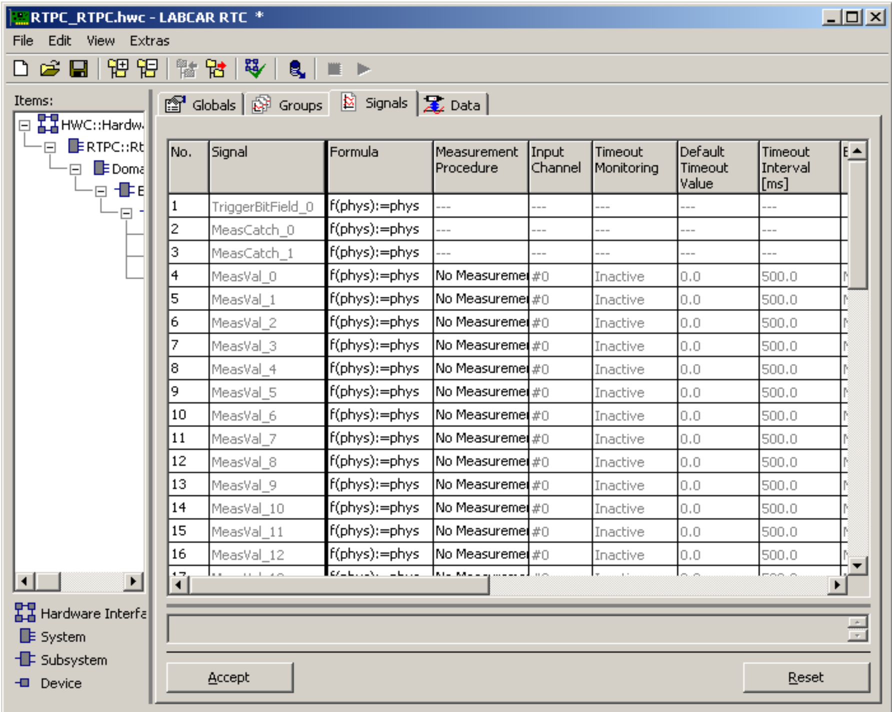


Fig. 16-40 The "Signals" Tab of the PB1651PWM1-In-Meas Device

Measurement Procedure

The measurement procedure is set in this list field. The section "The Measurement Procedures" on page 431 contains a detailed description as well as configuration guidelines on the individual procedures.

The notation --/-- in this list field means evaluation on rising edges whereas --\-- indicates evaluation on falling edges.

Note

To avoid unnecessary computing time, measurements which are not required should be deactivated, i.e. the option "No Measurement" is to be set in the list field.

Input Channel

The input channel of the module on which the measurement is to be carried out is set in this list field. The list field can only be edited if a measurement procedure has been selected for the measurement.

Timeout Monitoring

Definition of the timeout monitoring for the relevant measurement. The following settings are possible:

- "Inactive": No timeout monitoring.
- "Intvl Predef": Timeout monitoring in the interval defined in the "Timeout Interval [ms]" option field. The timeout measure value is the value set in the "Default Timeout Value" option field.
- "Intvl InpDep": Timeout monitoring in the interval defined in the "Timeout Interval [ms]" option field. The timeout measure value depends on the level of the input signal.

The list field can only be edited if a measurement procedure has been selected for the measurement.

Default Timeout Value

The value set in this numeric input field is output as the timeout measure value if "Intvl Predef" timeout monitoring mode has been set.

The input field can only be edited if a measurement procedure and the "Intvl Predef" timeout monitoring mode have been selected for the measurement.

Timeout Interval [ms]

Time interval in which the measurement is checked for a timeout. The interval is specified in milliseconds.

The input field can only be edited if a measurement procedure and one of the two timeout monitoring modes, "Intvl Predef" and "Intvl InpDep", have been selected for the measurement.

Parameter or Option Field	Data-Type*	Can be Edited Online	Comment/Value Range
Measurement Procedure	sint32	Yes	Definition of the measurement procedure. "Measurement Procedure" parameter value "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 L/H --\--" 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	Yes	Input channel on which the measurement is carried out. 0 ... 23: Channel number
Timeout Monitoring	uint8	Yes	Timeout monitoring. 0: "Inactive" 1: "Intvl Predef" 2: "Intvl InpDep"
Default Timeout Value	real32	Yes	Timeout measure value. Only relevant in "Intvl Predef" monitoring mode.
Timeout Interval [ms]	real32	Yes	Time interval (in milliseconds) in which the measurement is checked for timeout. Value range: ≥ 1.0 ms
* Data type which the RTIO driver uses internally for the parameter			

Tab. 16-23 PB1651PWM1-In-Meas Device: Configuration Parameters of the "Signals" Tab

16.6.11 The Measurement Procedures

Definitions

Active signal edge:

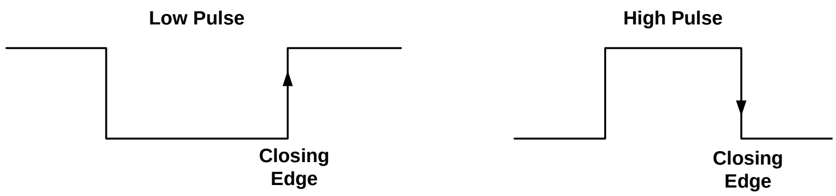
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.

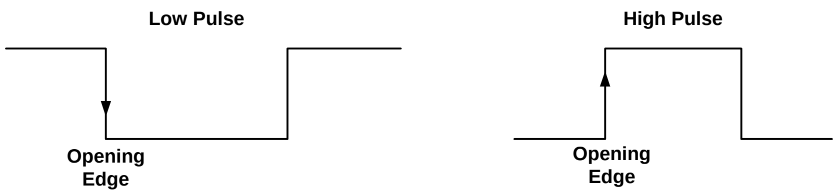


Inactive signal:

A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

Opening edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



The Basic Principle of Measure Value Calculation

The PB1651PWM1-RTIO driver calculates the measure values of the individual measurement procedures from a uniform set of data which it reads from the Dual-Ported RAM memory of the ES4350 Carrier Board or the ES1651.1 Carrier Board at the start of measure value calculation. The data set is periodically updated by the firmware of the carrier board; the RTIO driver accesses this data set asynchronously to this period.

The data set has the following structure:

- A 32-bit time stamp which specifies the time at which the PB1651PWM1 hardware acquired the data and stored it in the Dual-Ported RAM. From now on this time is referred to as the time of data acquisition.
- Four 32-bit data words per PWM input, which code the time stamp and the transfer direction of the last four edges preceding the time of data acquisition on the PWM input. The top bit (bit 32) of the relevant data word codes the transition type (1: rising, 0: falling). The 31 bits below contain the time stamp showing when the relevant edge occurred.
- One 32-bit data word per PWM input which specifies the cumulated high time of the PWM input since the initialization of the RTIO driver and the PB1651PWM1 hardware.
- One 32-bit data word per PWM input which specifies the cumulated low time of the PWM input since the initialization of the RTIO driver and the PB1651PWM1 hardware.
- One 16-bit data word per PWM input which specifies the number of rising edges on the PWM input since the initialization of the RTIO driver and the PB1651PWM1 hardware.
- One 16-bit data word per PWM input which specifies the number of falling edges on the PWM input since the initialization of the RTIO driver and the PB1651PWM1 hardware.

The time stamps have a resolution of 20 ns.

16.6.12 Pulse-Width Measurements

The measurement procedures

- High Time [µs]
- Low Time [µs]

measure the high and low times of the pulses of a PWM input. The pulse width of the pulse immediately preceding the last time of data acquisition is measured. Fig. 16-41 shows an example of a high-time measurement; the measured pulse is shown in bold print.

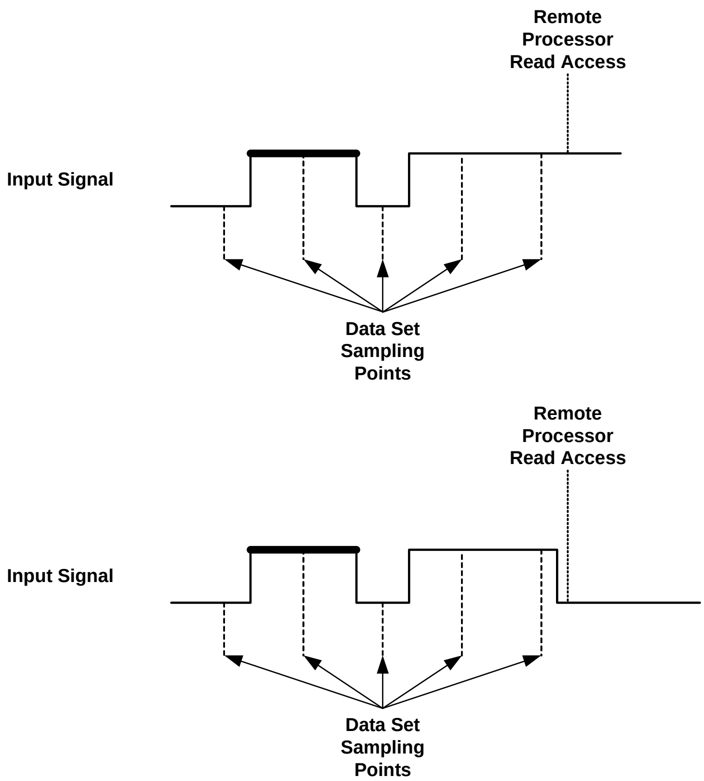


Fig. 16-41 High-Time Measurement (the Measured Pulse is shown in Bold Print)

If timeout monitoring has been activated for the measurement, a timeout is triggered at exactly the point when the time difference between the last time of data acquisition and the last closing edge is greater than the timeout period set.

The measure value which is output during a timeout is specified in Tab. 16-24.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	- If the signal level is inactive at the time of data acquisition, 0 is output as measure value (Fig. 16-42). - If the signal level is active at the time of data acquisition, the time difference between the time of data acquisition and the last opening edge of the pulse is output as measure value (Fig. 16-42).
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

Tab. 16-24 Pulse-Width Measurements: Timeout Measure Value with Different Procedures for Timeout Monitoring

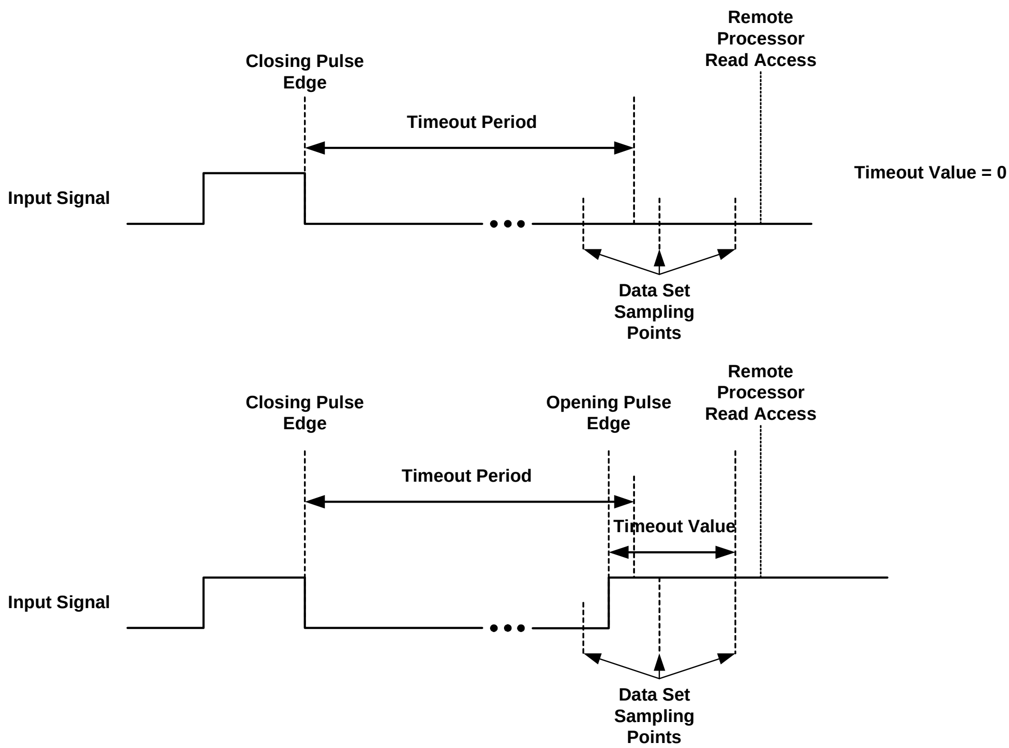


Fig. 16-42 Pulse-Width Measurements: Timeout Measure Value in "Intvl InpDep" Mode with the Example of a High-Time Measurement

16.6.13 Additive Pulse-Width Measurements

If the read process for the "MeasVal" signal group is activated again (periodically in special cases), the measurement procedures

- Additive High Time [µs]
- Additive Low Time [µs]

return the time in which the signal to be measured has assumed an overall active level between two consecutive activations of the read process.

Fig. 16-43 shows the measure value calculation using the example of an additive high-time measurement. The total of the line segments marked in bold print is the additive high time which is returned on the (n+1)th activation of the read process.

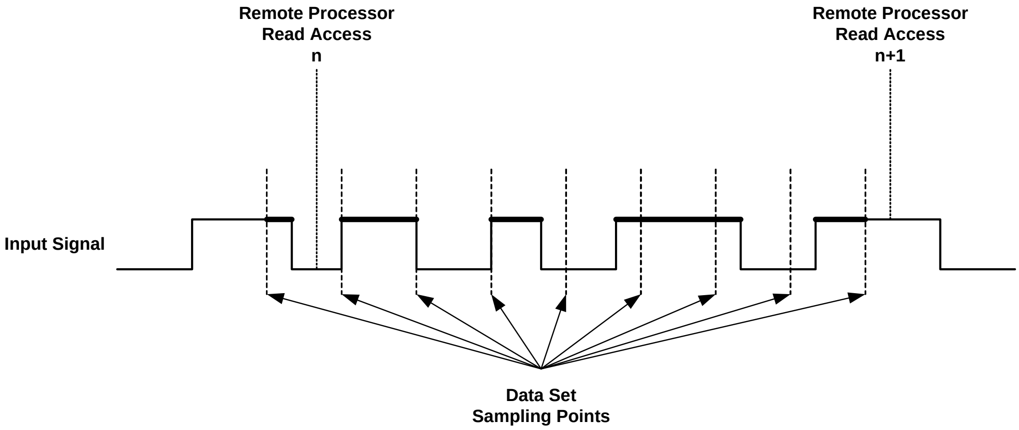


Fig. 16-43 Additive High-Time Measurement: Illustration of Measure Value Calculation

If timeout monitoring has been activated for the measurement, a timeout is detected at exactly the point when the time difference between the last time of data acquisition and the last edge of the signal (regardless of whether it is opening or closing) is greater than the timeout period set. In the case shown in Fig. 16-44, the timeout time expires between the time of data acquisition (a-1) and a. Shortly after the time of data acquisition (a+1), the RTIO driver reads data from the DPRAM of the carrier board and detects a timeout.

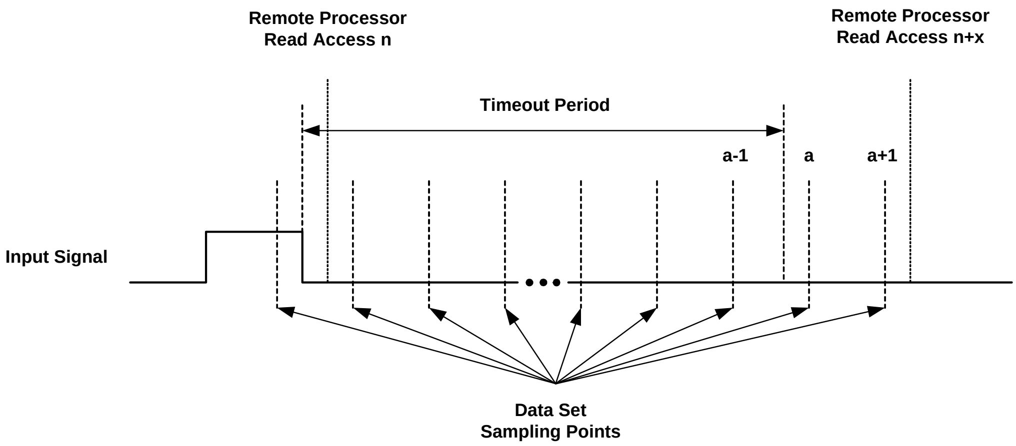


Fig. 16-44 Timeouts with Additive Pulse-Width Measurements

The measure value which is output during a timeout is specified in Tab. 16-25.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	The timeout measure value, as in a normal case without a timeout, equals the total of the active time of the signal between two activations of the read process. - This time is 0 if the signal level during this time was inactive. - This time is the same as the activation period if the signal level was active during this time.
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

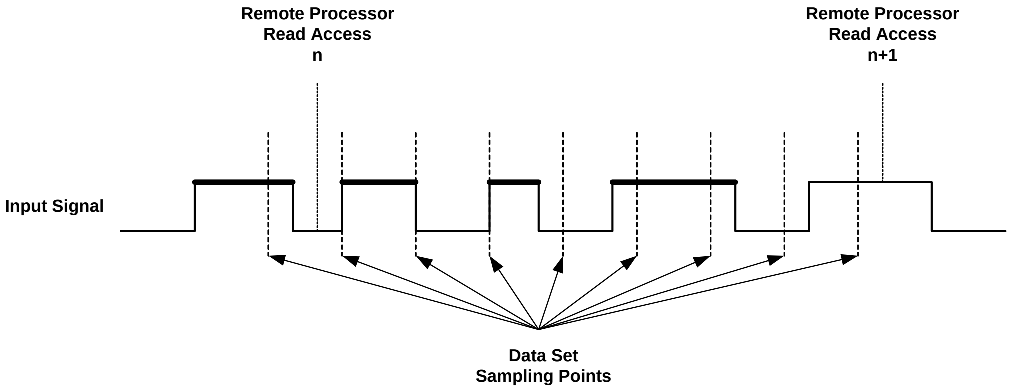
Tab. 16-25 Additive Pulse-Width Measurements: Timeout Measure Value with Different Procedures for Timeout Monitoring

16.6.14 Pulse and Edge Count

If the read process for the "MeasVal" signal group is activated again (periodically in special cases), the measurement procedures

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

return the number of active pulse or signal edges between two consecutive activations of the read process. Fig. 16-45 shows the method of pulse counting in detail using the example of a high-pulse count. All pulses whose closing edge is in the period between the activations of the read process are counted. If you want to specify this precisely, the period of time is defined by the times of data acquisition immediately preceding the activations. As far as the edge count is concerned, the evaluation period is exactly the same (Fig. 16-46).



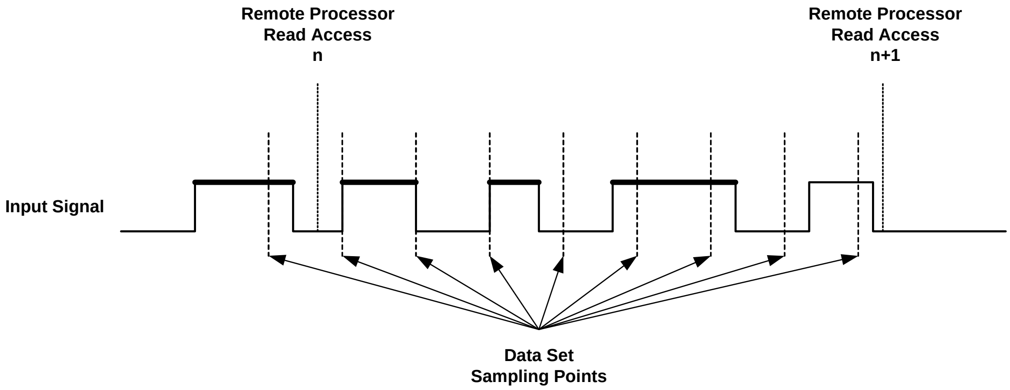


Fig. 16-45 High-Pulse Count: The Number of Pulses Marked in Bold Print is Returned as a Measure Value on the (n+1)th Time of Reading

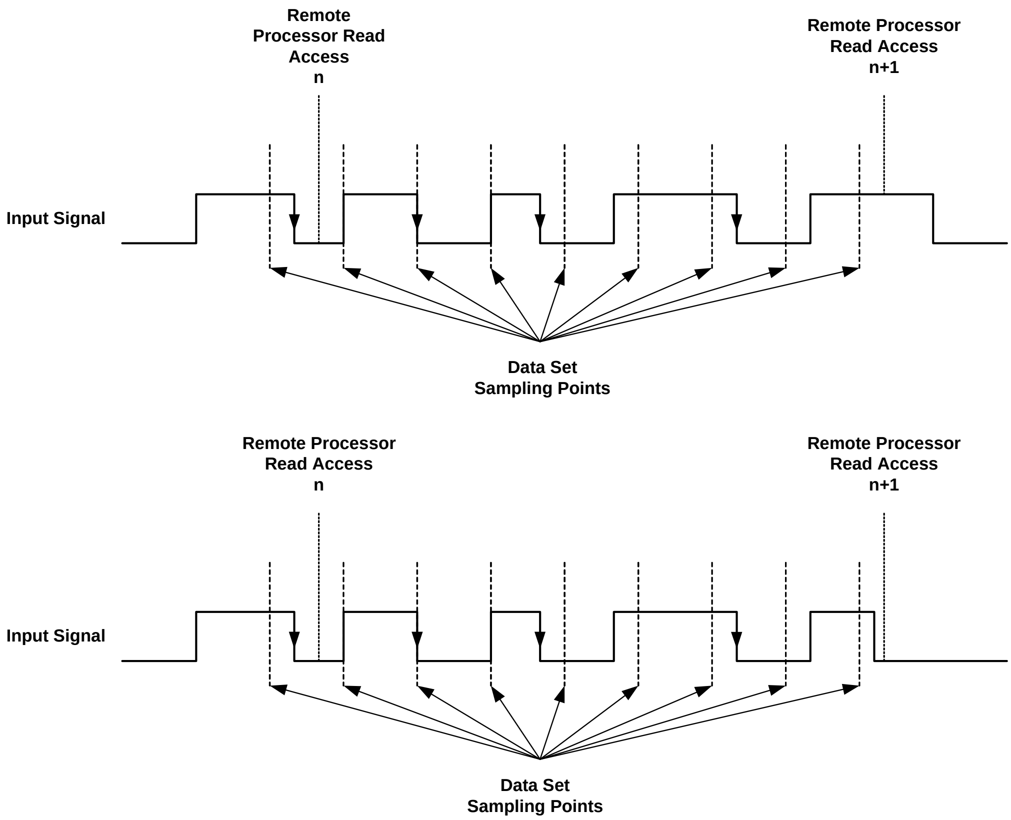


Fig. 16-46 Counting Falling Edges: The Number of Marked Falling Edges is returned as a Measure Value on the (n+1)th Time of Reading

If timeout monitoring has been activated for the measurement, a timeout is triggered at exactly the point when the time difference between the last [1] time of data acquisition and the last closing edge (pulse count) or active edge (edge count) of the signal is greater than the timeout period set.

The measure value which is output during a timeout is specified in Tab. 16-26.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	0 is output as timeout measure value.
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

Tab. 16-26 Pulse and Edge Count: Timeout Measure Value with Different Procedures for Timeout Monitoring

1. The time of data acquisition immediately preceding the RTIO driver's read access of the data set.

16.6.15 Frequency and Cycle Time Measuring

The measurement procedures

- "Cycle Time --/-- [µs]"
- "Cycle Time --\-- [µs]"
- "Frequency --/-- [Hz]"
- "Frequency --\-- [Hz]"

return the cycle time or the frequency of the signal at a PWM input measured on rising or falling edges. The active signal edge immediately preceding the last[1] time of data acquisition and its corresponding period are evaluated.

Fig. 16-47 shows the evaluated period using the example of a frequency or cycle measurement on rising edges.

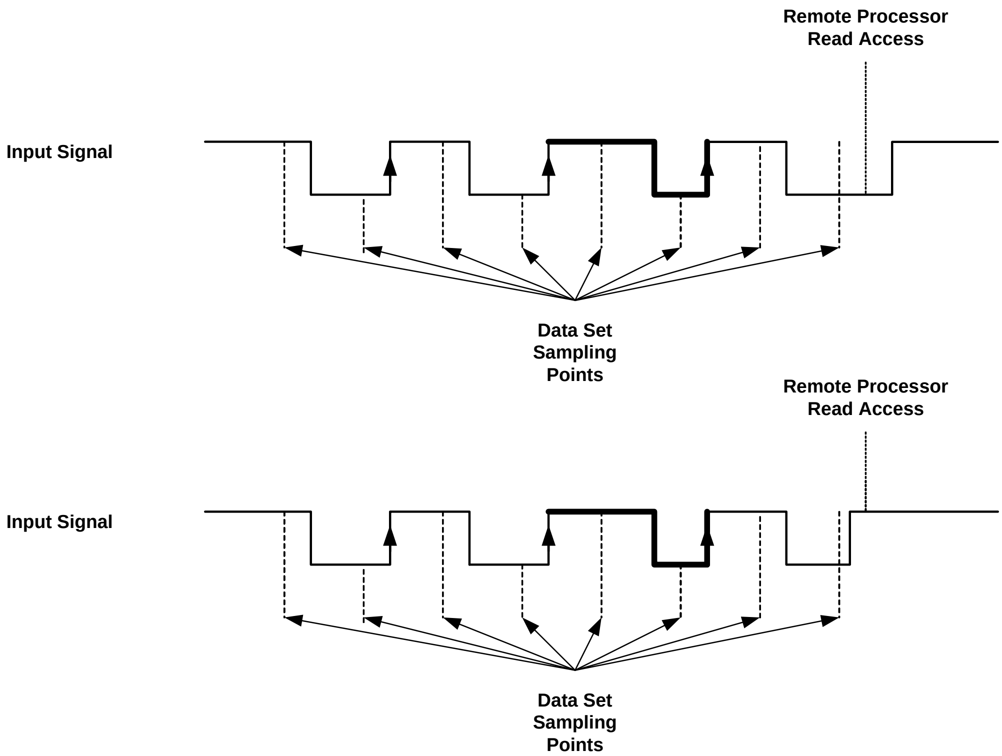


Fig. 16-47 Evaluated Period (in Bold) in a Frequency or Cycle Measurement on Rising Edges

1. The time of data acquisition immediately preceding the RTIO driver's read access of the data set.

If timeout monitoring has been activated for the measurement, a timeout is triggered at exactly the point when the time difference between the last[1] time of data acquisition and the last active signal edge is greater than the timeout period set (Fig. 16-48).

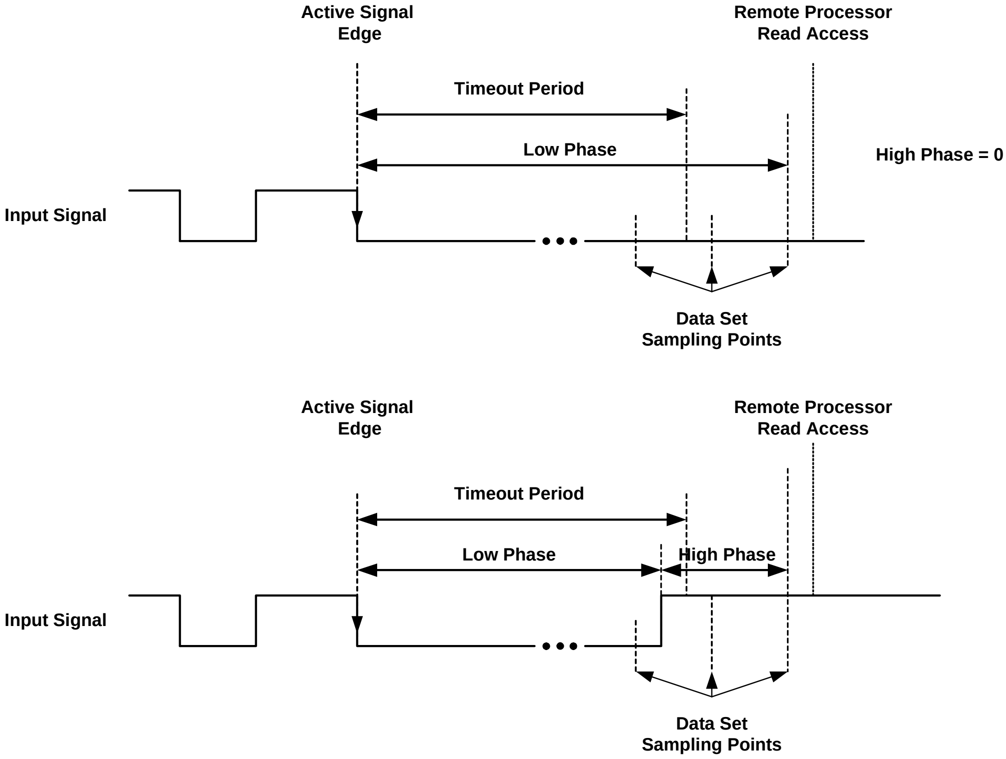


Fig. 16-48 Frequency and Cycle Time Measurements: Timeout Measure Value with "Intvl InpDep" Mode for Timeout Monitoring

The measure value which is output during a timeout is specified in Tab. 16-27.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	In cycle time measurements, the time difference between the last time of data acquisition and the last active signal edge is output. In frequency measurements, the corresponding reciprocal value.
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

Tab. 16-27 Frequency and Cycle Time Measurements: Timeout Measure Value with Different Procedures for Timeout Monitoring

[1] The time of data acquisition immediately preceding the RTIO driver's read access of the data set.

16.6.16 Duty Cycle Measurements

Duty cycles can be determined in various ways with the PB1651PWM1 Module. In duty cycle measurements, a distinction is made between measurements which set the high phase of a signal in relation to the low phase (or vice versa) and measurements which set the high phase (or the low phase) of a signal in relation to the cycle duration.

The first type of duty cycle measurement is referred to below as a duty cycle P/P; "P/P" means that two pulse durations are set in relation to each other. The second type of duty cycle measurement is referred to below as a duty cycle P/C; "P/C" means that a pulse duration is set in relation to the cycle time of the signal.

$$\text{Duty Cycle}^{P/P}: \quad \frac{L}{H}, \frac{H}{L} \qquad 0 \leq \text{Duty Cycle}^{P/P} \leq \infty$$

$$\text{Duty Cycle}^{P/C}: \quad \frac{L}{L+H}, \frac{H}{L+H} \qquad 0 \leq \text{Duty Cycle}^{P/C} \leq 1$$

The user determines the edge on which the measure value is calculated by selecting the measurement procedure. The following duty cycle measurements are available:

Type "P/P"

- "Duty Factor L/H --/--"
- "Duty Factor L/H --\--"
- "Duty Factor H/L --/--"
- "Duty Factor H/L --\--"

Type "P/C"

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

As far as the selection of the period of the input signal is concerned, on the basis of which the duty cycle calculation is executed, exactly the same as was described for frequency and cycle time measurements in section 16.6.15 on page 439 applies.

And as for timeout monitoring, what is written in section 16.6.15 on page 439 applies: A timeout is triggered at exactly the point when the time difference between the last[1] time of data acquisition and the last active signal edge is greater than the timeout period set.

[1] The time of data acquisition immediately preceding the RTIO driver's read access of the data set.

The measure value which is output during a timeout is specified in Tab. 16-28.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	In a timeout, the high phase and low phase are determined as shown in Fig. 16-49 in the case of duty cycle calculations on falling edges. The duty cycle is then calculated based on these values.
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

Tab. 16-28 Duty Cycle Measurements: Timeout Measure Value with Different Procedures for Timeout Monitoring

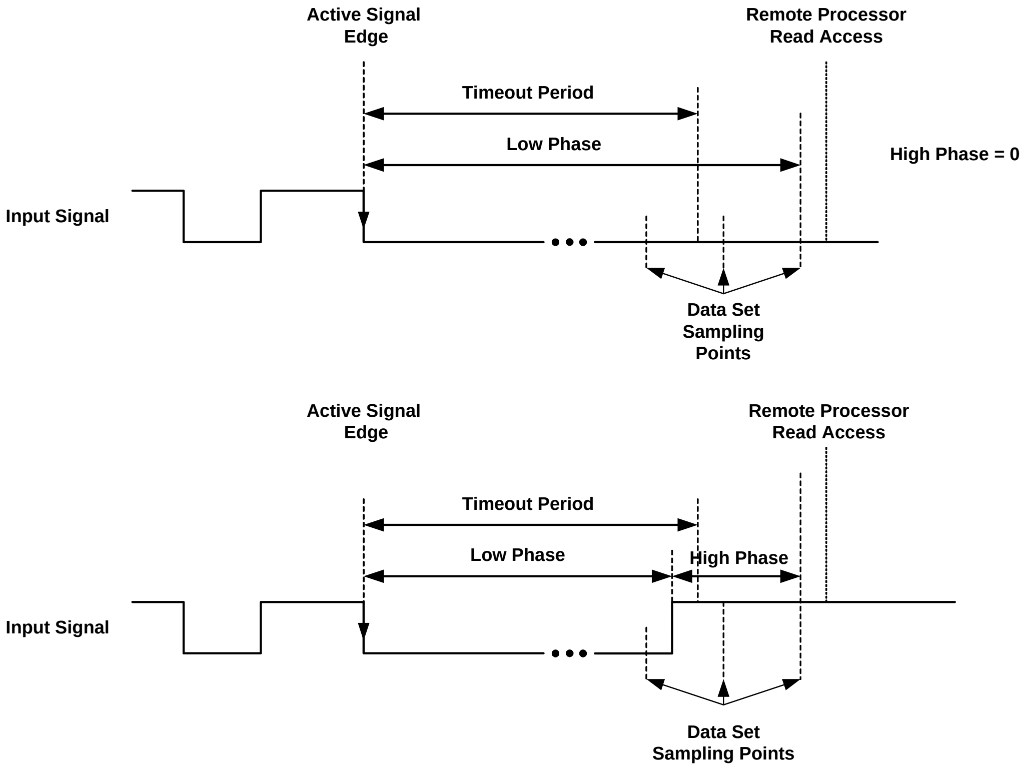


Fig. 16-49 Duty Cycle Measurements on Falling Edges: Determination of the High Phase and Low Phase of the Signal in a Timeout

16.6.17 Level Measurements

The measurement procedures

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

return the level of a PWM input in the form of active/inactive information. "0" means the signal is inactive; "1" means the signal is active.

If timeout monitoring has been activated for the measurement, a timeout is detected at exactly the point when the time difference between the last[1] time of dataacquisition and the last edge of the signal (regardless of whether it is opening or closing) is greater than the timeout period set.

The measure value which is output during a timeout is specified in Tab. 16-29.

Timeout Monitoring	Timeout Measure Value
"Intvl InpDep"	The timeout measure value is the same as the signal state on the last time of data acquisition, i.e.: - 0, if the signal level was inactive during the last timeof data acquisition. -1, if the signal level was active during the last timeof data acquisition.
"Intvl Predef"	The value set in the "Default Timeout Value" option field is output as measure value.

Tab. 16-29 Level Measurements: Timeout Measure Value with Different Procedures for Timeout Monitoring

1. The time of data acquisition immediately preceding the RTIO driver's read access to the data set.

16.6.18 The PB1651PWM1-Out Device

The PB1651PWM1-Out device is used to configure and address the 16 channels of a PB1651PWM1 Module for PWM output.

16.6.19 Globals (PB1651PWM1-Out Device)

No settings need to be made in the "Globals" tab.

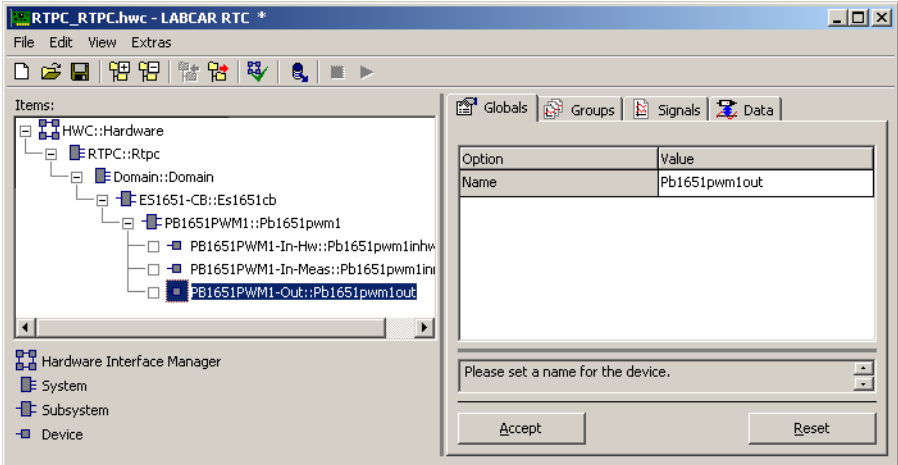


Fig. 16-50 The "Globals" Tab of the PB1651PWM1-Out Device

16.6.20 Groups (PB1651PWM1-Out Device)

The PB1651PWM1-Out device has two signal groups (see Fig. 16-51) which are transmitted to the PB1651PWM1 Module from the experimental target.

In addition to frequency, duty cycle and phase data for the individual PWM outputs, the "PwmOut" signal group transports driving signals for switching the PWM output on and off.

The signal group "Sync" transfers a synchronization signal, with which the acceptance of changed frequency and duty cycle data by the PB1651PWM1 hardware can be synchronized from the RTIO.

The signal group "ChnState" receives state signals which describe the current status of the output drivers of the corresponding output channels.

Tasks of the real-time operating system must be allocated to the signal groups. A task with periodic activation is selected for the "PwmOut" signal group. The activation period depends on the dynamic behavior of the signals to be output. A task with periodic activation or software activation is selected for the "Sync" signal group.

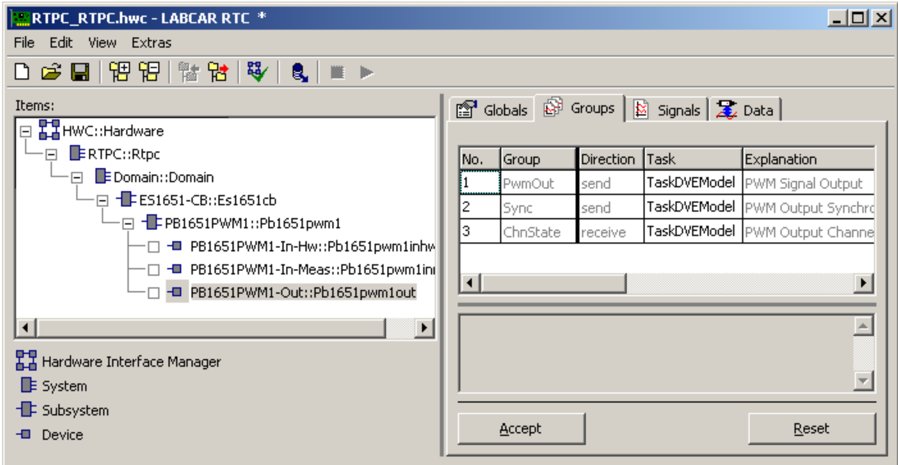


Fig. 16-51 The "Groups" Tab of the PB1651PWM1-Out Device

The RTIO Signals of the "PwmOut" Signal Group

The "PwmOut" signal group has 64 RTIO signals.

The relevant allocated PWM output can be switched between the state of the PWM signal output and high-impedance state using the signals "EnableChn_0" to "EnableChn_15". These signals can, for example, be used in error simulation to simulate a circuit failure with an actuator.

The concrete response of the PB1651PWM1 hardware and firmware with a deactivation of PWM output can be explained using Fig. 16-52:

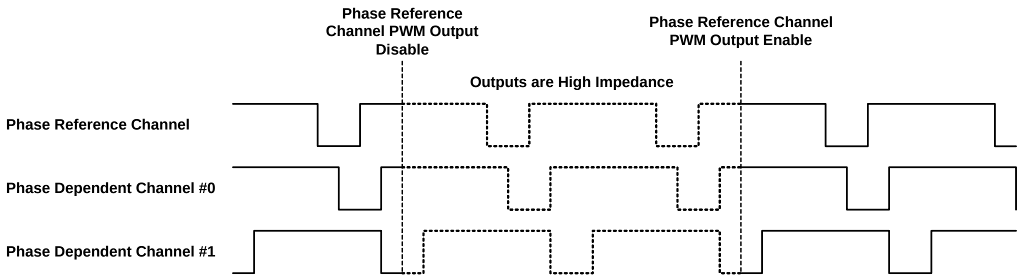


Fig. 16-52 Activation and Deactivation of the PWM Output of Channels which have a Defined Phase Relation with one another

If several channels have a defined phase relation with one another, a deactivation or activation of the PWM output of the phase reference channel will also result in the deactivation or activation of the PWM output of the phase-shifted channels. The "EnableChn_x" RTIO signals of the channels which are phase-shifted from the phase reference channel are ignored.

If the PWM output of a channel is reactivated after deactivation, PWM output is continued as if there had never been an interruption.

The "Frequency_0" to "Frequency_15" signals set the signal frequency of the allocated PWM output. The frequency range is 0 to 100 kHz and the frequency resolution is 0.01 Hz.

The "DutyCycle_0" to "DutyCycle_15" signals set the duty cycle of the allocated PWM output. Duty cycles are defined with values between 0.0 and 1.0. 0.0 corresponds to a duty cycle of 0%; 1.0 to a duty cycle of 100%.

PB1651PWM1 hardware and firmware make it possible to define the setting of phase shifts between output channels with the same PWM frequency. One of the PWM outputs has be to specified as a phase reference channel for this purpose. The phase relation of the remaining PWM outputs which have defined phase relations with the reference channel is set with the phase signals allocated to these outputs "Phase_x" (x = 0, 1, ... 15).

If the acceptance of values from signal generation is controlled on the hardware via the synchronization lines of the carrier board, the phase relation of the individual PWM outputs can also be changed later using the allocated phase signals.

Tab. 16-30 summarizes the properties of the RTIO signal.

RTIO Signal	Data Type*	Comment/Value Range
EnableChn_0 ... EnableChn_15	bool	Activation and deactivation of the PWM signal output. 0: Output is high-impedance; PWM signal output is inactive 1: PWM signal output is active. Is not important with PWM outputs with phase reference channel.
Frequency_0 ... Frequency_15	real32	Signal frequency - Frequency range: 0 to 100 kHz - Frequency resolution. 0.01 Hz - Is not important with PWM outputs with phase reference channel.
DutyCycle_0 ... DutyCycle_15	real32	Duty cycle; value range: 0.0 ... 1.0
Phase_0 ... Phase_15	real32	Phase relation; value range: -360.0° ... 360.0° Is not important with PWM outputs without phase reference channel.

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-30 PB1651PWM1-Out Device: The RTIO Signals of the "PwmOut" Signal Group

The RTIO Signals of the "Sync" Signal Group

The signal group "Sync" contains one RTIO signal, "SyncSgl".

PB1651PWM1 RTIO drivers and firmware make it possible to control when frequency and phase data transferred to the module by the RTIO or the simulation model is accepted by the hardware. The "Hardware Update" option in the "Signals" tab controls this process. If the "RTIO Controlled" option is set in this list field for one or more channels, the frequency and phase data transmitted to these channels is only accepted by the hardware when the module firmware detects a change of the "SyncSgl" RTIO signal from 0 to 1.

Tab. 16-21 summarizes the properties of the RTIO signal.

RTIO Signal	Data Type*	Comment/Value Range
SyncSgl	bool	A change from 0 to 1 means that suitably configured PWM outputs update changed frequency and phase data at their outputs.
* Data type which the RTIO driver uses internally for the parameter		

Tab. 16-31 PB1651PWM1-Out Device: The RTIO Signals of the "Sync" Signal Group

The RTIO Signals of the "ChnState" Signal Group

The "ChnState" signal group has 16 RTIO signals, "ChnState_0" to "ChnState_15". The signal group is used to independently acquire the current states of the individual output drivers for PWM signal generation.

Tab. 16-32 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
ChnState_0 ... ChnState_15	uint8	Every output channel has a 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver: - Bit0: Lowside - Bit1: Highside A - Bit2: Highside B - Bit3: Channel Level If one of the first three bits is set, an overcurrent has been detected in the current configuration of the output driver. The fourth bit describes the current signal level (high or low) of the relevant output channel.
* Data type which the RTIO driver uses internally for the parameter		

Tab. 16-32 PB1651PWM1-Out Device: The RTIO Signals of the Signal Group "ChnState"

16.6.21 Signals (PB1651PWM1-Out Device)

The 16 outputs of a PB1651PWM1 Module are configured in the "Signals" tab. Fig. 16-53 shows the option fields of the "Signals" tab. All option fields can be edited online (i.e. during runtime of the model on the experimental target).

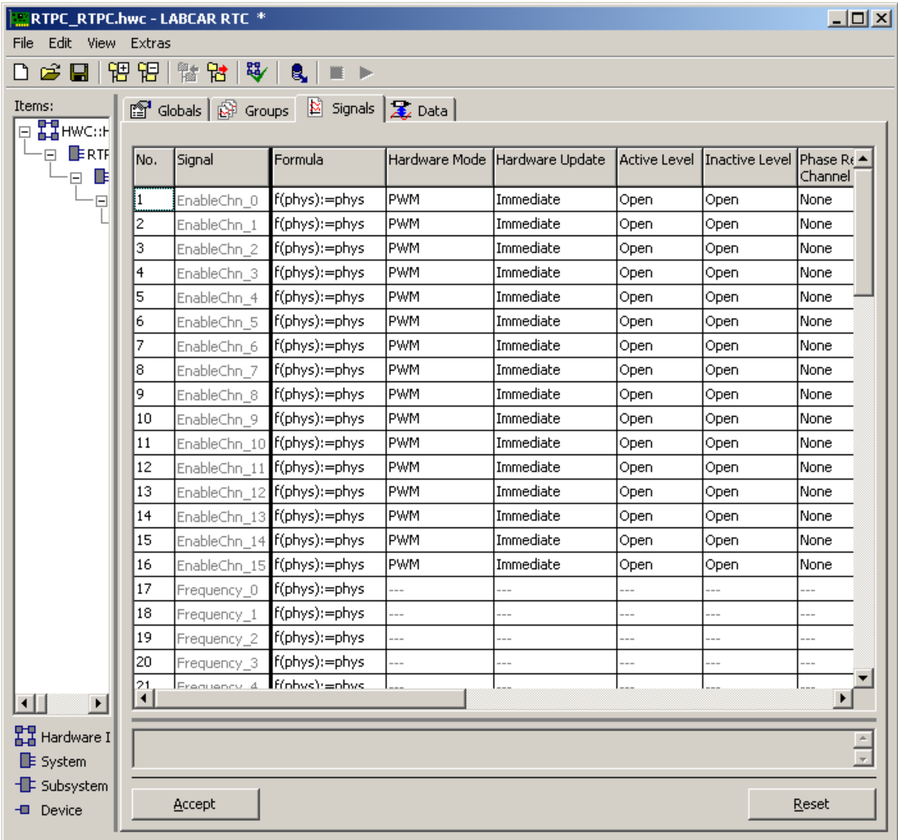


Fig. 16-53 The "Signals" Tab of the PB1651PWM1-Out Device

Hardware Mode

Every output channel can be operated in two different modes; "PWM" and "Digital Out".

- PWM

  If this option is active, PWM signals with a minimum pulse duration of 2 µs can be generated.

- Digital Out

  This configuration of the output channel generates signal levels. The frequency range is limited from 0 Hz to 100 Hz (with a resolution of 0.01 Hz). The signal levels are defined via the RTIO signals "DutyCycle_x" (x = 0, 1, … 15). Values less than 0.5 define a "Low" level and values greater than or equal to 0.5 generate a "High" level.

Note

In "Digital Out" mode, the other configuration options, "Hardware Update" and "Phase Reference Channel", cannot be changed. If output channels in "PWM" mode are related via the option "Phase Reference Channel", the basic state is generated when the "Digital Out" mode is selected, i.e. all relations with regard to "Phase Reference Channel" are deleted.

Hardware Update

This list field is used to set when changes made by the model or the user to the "Frequency_x" and "DutyCycle_x" RTIO signals (x = 0, 1, ... 15) are accepted by the PB1651PWM1 hardware.

The following settings are possible:

- **Immediate**

  Value changes are accepted immediately by the hardware.

- **RTIO Controlled**

  Value changes are only accepted when the module firmware has detected a change from 0 to 1 on the "SyncSgl" RTIO signal (see "The RTIO Signals of the "Sync" Signal Group" on page 446).

- **Cycle End**

  Value changes are only accepted when the current PWM period of the relevant PWM output is over.

- **Sync Line #x --\--**

  Value changes are only accepted on the next falling edge on synchronization line x.

- **Sync Line #x --/--**

  Value changes are only accepted on the next rising edge on synchronization line x.

The phase relation of the individual PWM outputs can be set in combination with the relevant phase signals. After one of the options named above has been configured, the relevant phase signals are also released.

Note

The settings in the "Hardware Update" list field only have an effect on the "Frequency_x" and "DutyCycle_x" RTIO signals. Value changes of the "EnableChn_x" RTIO signals (x = 0, 1, ... 15) are generally immediately accepted by the hardware. Value changes of an RTIO signal "Phase_x" (x = 0, 1, ... 15) are generally accepted once the current PWM period of the phase reference channel allocated to the PWM output x is concluded.

Note

This list field is deactivated for channels which have a defined phase relation with a reference channel – the settings of the phase reference channel in this field are accepted implicitly.

Active Level

The active state of a PWM signal is defined as follows: If a duty cycle of 60% is set for a PWM signal, it assumes active signal state for 60% of the period duration.

The active state of the PWM signal is set in this list field. The following options are available:

- **Open**

  The PWM output is high-impedance and can be set to a defined voltage level using an external pull-up- or pull-down resistor.
- **Ground**

  The PWM output is at ground.
- **V Batt A Out**

  The PWM output outputs battery voltage A.
- **V Batt B Out**

  The PWM output outputs battery voltage B.

Inactive Level

The inactive state of a PWM signal is defined as follows: If a duty cycle of 60% is set for a PWM signal, it assumes inactive signal state for 40% of the period duration.

The inactive state of the PWM signal is set in this list field. The following options are available:

- **Open**

  The PWM output is high-impedance and can be set to a defined voltage level using an external pull-up- or pull-down resistor.
- **Ground**

  The PWM output is at ground.

- **V Batt A Out**

  The PWM output outputs battery voltage A.
- **V Batt B Out**

  The PWM output outputs battery voltage B.

Phase Reference Channel

PB1651PWM1 hardware and firmware make it possible to define the setting of phase shifts between output channels with the same PWM frequency. One of the PWM outputs has be to specified as a phase reference channel for this purpose. The number of the reference channel is then to be entered in the "Phase Reference Channel" list field of the remaining outputs which have defined phase relations with this reference channel.

If a PWM output is configured so that it has a defined phase relation to a reference channel (a value not equal to "None" is set in the "Phase Reference Channel" list field), this has the following consequences:

- As the phase-shifted PWM output and phase reference channel must have the same PWM frequency, the RTIO signal "Frequency_x" of the phase-shifted PWM output is ignored by the firmware. The "Frequency_x" value of the phase reference channel is automatically accepted.
- The "Hardware Update" list field of the phase-shifted PWM output is deactivated. The setting of the phase reference channel is automatically assumed in this field for the phase-shifted PWM output.

  If the "Cycle End" option is set in the "Hardware Update" list field of the phase reference channel and therefore, as described above, automatically assigned in the "Hardware Update" list field of the phase-shifted PWM output, this also means the end of the PWM period of the phase reference channel and the phase-shifted PWM output.
- Chained phase relations are not allowed, i.e. if several PWM outputs have defined phase relations with one another, exactly one PWM output must be defined as the phase reference channel for these channels. This channel then has to be set in the "Phase Reference Channel" list field for the remaining phase-shifted outputs. The "None" option must be set for the phase reference channel in the "Phase Reference Channel" list field.
- The value of the RTIO signal "Phase_x" of a phase reference channel is ignored.

Tab. 16-33 summarizes the properties of the individual parameters.

Parameter or Option Field	Data-Type*	Can be Edited Online	Comment/Value Range
Hardware Mode	uint8	Yes	Type of signal generation PWM: 0 Digital Out: 1
Hardware Update	uint8	Yes	Time or event at which the PB1651PWM1 hardware accepts a changed frequency or a changed duty cycle at the allocated output "Time/event" parameter value "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	Yes	Active signal state "Open": 0 "Ground": 1 "V Batt A Out": 2 "V Batt B Out": 3
Inactive Level	uint8	Yes	Inactive signal state "Open": 0 "Ground": 1 "V Batt A Out": 2 "V Batt B Out": 3
Phase Reference Channel	sint8	Yes	Phase reference channel "None": -1 #0 ... #15: 0 ... 15

* Data type which the RTIO driver uses internally for the parameter

Tab. 16-33 PB1651PWM1-Out Device: Configuration Parameters of the "Signals" Tab

Note

*** The options "Sync Line #4 --/--", "Sync Line #5 --/--", "Sync Line #4 --\--" and "Sync Line #5 --\--" are only available if the module is inserted on an ES4350 Carrier Board*

17 ES4315-VXI - VME64x/VXI Adapter

The ES4315 VME64x/VXI adapter is the link between the VME64x and the VXI part of the ES4300 rack. The adapter is a fixed part of the ES4300 housing and is not accessible to the user. It is addressed by the VME as a slave board and is seen by the VXI as slot 0 controller. This means a VME master board can be used to control the VXI system.

An ES4300 VXI system is integrated into the RTIO Editor by adding an ES4315-VXI subsystem within an ES1130 system. Only one ES4315-VXI subsystem can be added to an ES1130 system as the ES1130 simulation system is physically inserted into the VME64x part of the ES4300 rack. All other VXI components, such as VXI boards, can be added below the ES4315-VXI subsystem. An RTIO driver is implemented for the ES4315-VXI subsystem: the driver is responsible for initializing the entire VXI system and administrating the VXI system resources.

17.1 ES4315-VXI Subsystem

17.1.1 Globals (ES4315-VXI Subsystem)

This section describes the specific options of the ES4315-VXI subsystem.

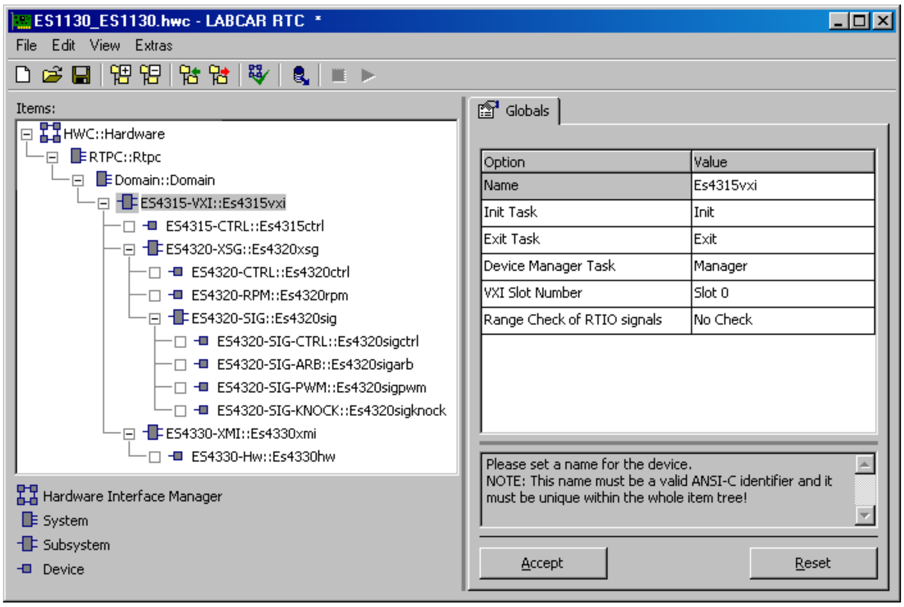


Fig. 17-1 The "Globals" Tab of the ES4315-VXI Subsystem

Device Manager Task

You can define a task which the driver’s Device Manager API function calls. This has no function, however, in this version.

VXI Slot Number

This option denotes the VXI slot position of the ES4315-VME64x/VXI adapter. The specification Slot 0 cannot be changed as the VME64/VXI adapter always corresponds to Slot 0 of the VXI system. The first usable slot in the VXI housing is therefore Slot 1.

Range Check of RTIO Signals

This option allows you to determine the extent of the RTIO signal value range check during runtime. The setting can be used by all VXI board drivers which are lower in the hierarchy. The following settings are possible. It is generally applied to all signals of both send and receive signal groups during code generation.

Setting	Description
No Check	No check takes place
Limit to Range	The signals are limited to the ranges defined by code generation.
Limit to Range and Notify to Monitor Window	The signals are limited to the ranges defined by code generation and an error message is issued in the Monitor Window if the range is exceeded. The error message contains the name of the RTIO signal and the RTIO process.

17.2 ES4315-CTRL Device

17.2.1 Globals (ES4315-CTRL Device)

This section describes the specific options of the ES4315-CTRL device.

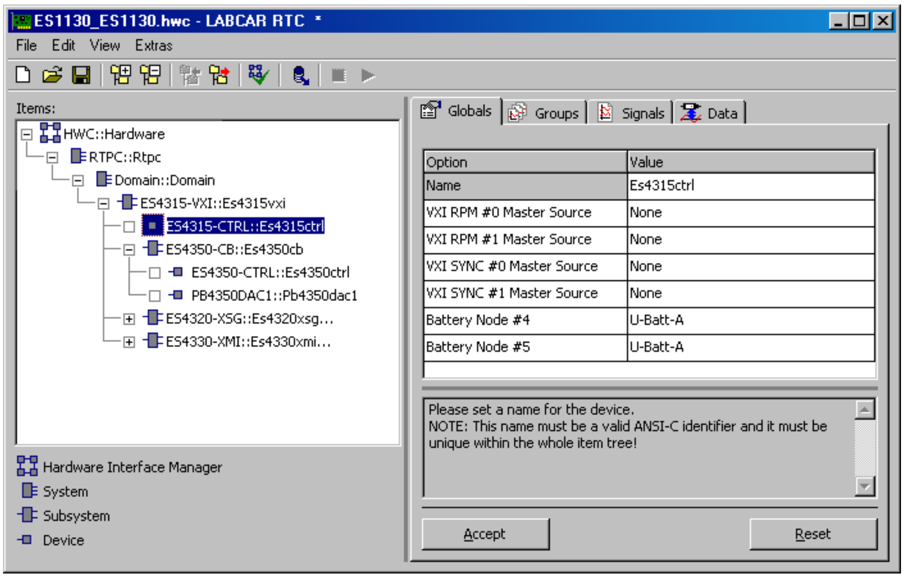


Fig. 17-2 The "Globals" Tab of the ES4315-CTRL Device

VXI RPM #0 Master Source, VXI RPM #1 Master Source

In LABCAR projects, it is often necessary to synchronize sequences. Projects for testing engine ECUs, for example, require precise angle synchronization between the generation of injection and ignition signals on the ECU and their being measured on test hardware (e.g. ES4330 boards). Engine ECU and test hardware are supplied with the same crankshaft angle clock (CAC) in LABCAR projects.

ES4300-based LABCAR systems provide two independent CAC buses on the ES4300 backplane, each with 3 lines. One bus line is used to synchronize to the zero crossing of the crankshaft angle, the second transfers the actual angle clock and the running direction of the engine is coded on the third.

Like every hardware bus, a CAC bus can only be driven by one source. In addition to CAC generators on ES4320 boards, ES4350 boards are also possible sources.

Receivers of CAC signals are:

- ES4330 boards which use the signals to measure speed-synchronous signals precisely
- ES4320 boards operated in Slave mode which use the signals to synchronize their signal generators to an external crankshaft angle clock
- ES4350 boards which transfer the crankshaft angle clock to the piggybacks

Use these list boxes to define the source of the CAC signals for the two buses on the backplane of the ES4300 Signal Box.

The list boxes are structured dynamically. If an ES4320 board is added to the hardware description, the four CAC generators (RPM #0 ... RPM #3) of this board are offered for selection. If the two external CAC ports SYNC0 and SYNC1 on the front panel of the ES4320 board are configured as inputs, they are also offered for selection (Ext RPM #0 and Ext RPM #1). For more detailed information on the configuration of the external CAC ports of an ES4320 Board, refer to the section "Sync #0, Sync #1 Configuration" on page 463.

Other possible sources, in addition to the CAC generators on ES4320 boards, are ES4350 boards. If an ES4350 board is added to the hardware description, its internal CAC buses (RPM #0 and RPM #1) can be switched to the backplane buses. At the time of writing, however, this possibility is limited as there are as yet no piggybacks for CAC generation on ES4350 boards.

VXI SYNC #0 Master Source, VXI SYNC #1 Master Source

In addition to the buses described to synchronize to the crankshaft angle clock, the ES4300 backplane has two other lines which are used to synchronize hardware units on different VXIbus boards of an ES4300 Signal Box. The list boxes "VXI SYNC #0 Master Source" and "VXI SYNC #1 Master Source" are used to specify the hardware units which drive these two lines. The list boxes are structured dynamically. If an ES4350 board is added to the RTIO hardware description, the six lines of the internal synchronization bus of this board are added to the selection. Each of these six synchronization lines of an ES4350 board can be switched to one or both backplane lines.

Battery Node Control Source

The battery node states are set in the "BnStates" signal group of the ES4315-CTRL device. You can specify whether each of the eight battery nodes is enabled or disabled.

Note

This option is therefore not normally visible to the user. The default setting "All ES43XX Boards" results in the battery node information being transferred to all boards by command.

Battery Node #4

This option is used to select which voltage (UBatt_A or UBatt_B) can be connected to battery node 4.

Battery Node #5

This option is used to select which voltage (UBatt_A or UBatt_B) can be connected to battery node 5.

Note

There are basically two different battery voltages, UBatt_A and UBatt_B. Battery nodes 0 to 3 can only be connected to UBatt_A, battery nodes 5 to 7 to UBatt_B. For battery nodes 4 and 5, you can choose between UBatt_A and UBatt_B. This LABCAR system setting cannot be changed during runtime and is communicated to the VXI system with these settings.

17.2.2 Groups (ES4315-CTRL Device)

The device implements a send signal group, "BnStates". You can use this to communicate the states of the battery nodes to the VXI system. Communication with the VXI system is not initiated by the driver until the status of at least one battery node has changed.

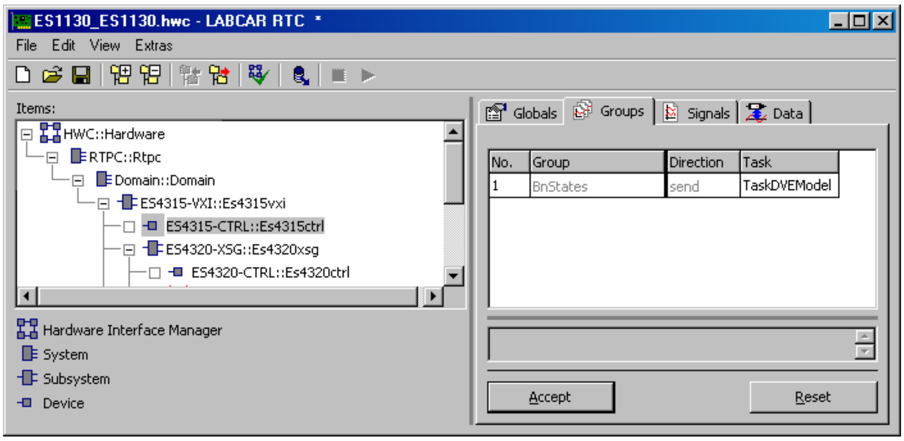


Fig. 17-3 The "Groups" Tab of the ES4315-CTRL Device

17.2.3 Signals (ES4315-CTRL Device)

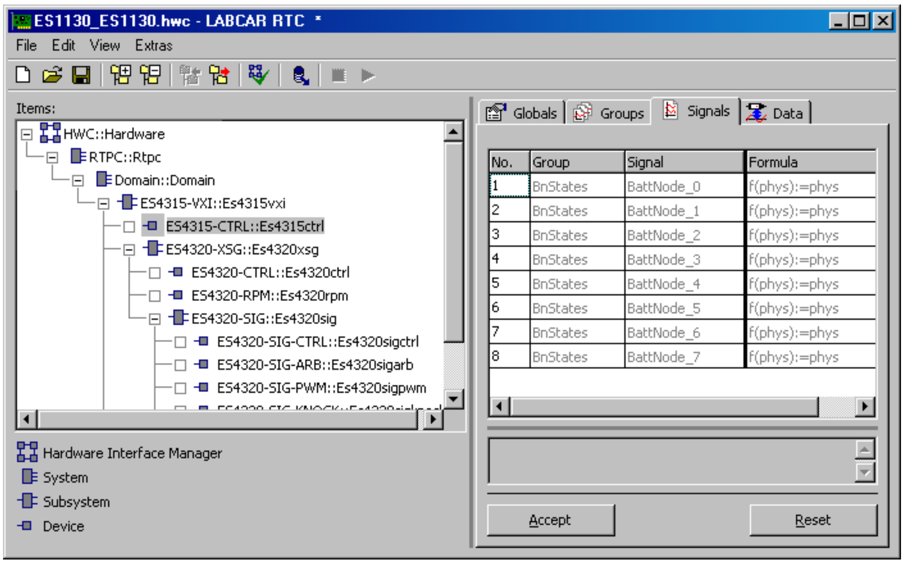


Fig. 17-4 The "Signals" Tab of the ES4315-CTRL Device

Signal	Direction	Description
BattNode_0 ... BattNode_7	send	Switch state of the battery node 0 ... 7

17.2.4 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
BattNode_0 ... BattNode_7	bool	false [0] = battery node disabled true [1] = battery node enabled

17.3 ES4320-XSG - PWM and Arbitrary Signal Generation

The ES4320-XSG board (eXtended Signal Generator board) makes it possible to generate PWM signals and freely programmable (arbitrary) signals with misfire functionality and knock signals. The board has twelve signal generators (channels) for this purpose. Each channel can be used as a PWM channel, as an arbitrary channel or as a knock channel.

Common Properties of the Channels:

- Galvanic isolation possible
- Two analog and several binary output modes possible
- Cutout relay to open the channel
- Settable pull-up current limitation for binary output modes
- Synchronous control of the channels in groups of three

Properties of the Arbitrary Channel

- Description of the signal trace with up to four signal banks
- Specification of an external reference voltage possible
- Combination with misfire generation possible

Properties of the PWM Channel

- Specification of frequency, duty cycle
- Specification of voltage values for low and high level possible

Properties of the Knock Channel

- Knock signal generation for up to 20 cylinders
- Specification of knock signal probability, noise properties and the knock envelope curve

A further unit of the board contains four independent speed (RPM) generators. These enable the generation of controllable angle signals with start mark and direction information. These RPM signals are necessary for addressing the arbitrary signal generators and acquiring angle-synchronous signals in the ES4330 VXI Signal Measurement Board (ES4330-XMI).

An ES4320 board is integrated by adding an ES4320-XSG subsystem to an ES4315-VXI subsystem in the RTIO Editor. Several ES4320 boards can be added to an ES4315-VXI subsystem. However, a maximum of 9 VXI boards can be added as each board occupies one VXI slot.

17.3.1 Globals (ES4320-XSG Subsystem)

This section describes the specific options of the ES4320-XSG subsystem.

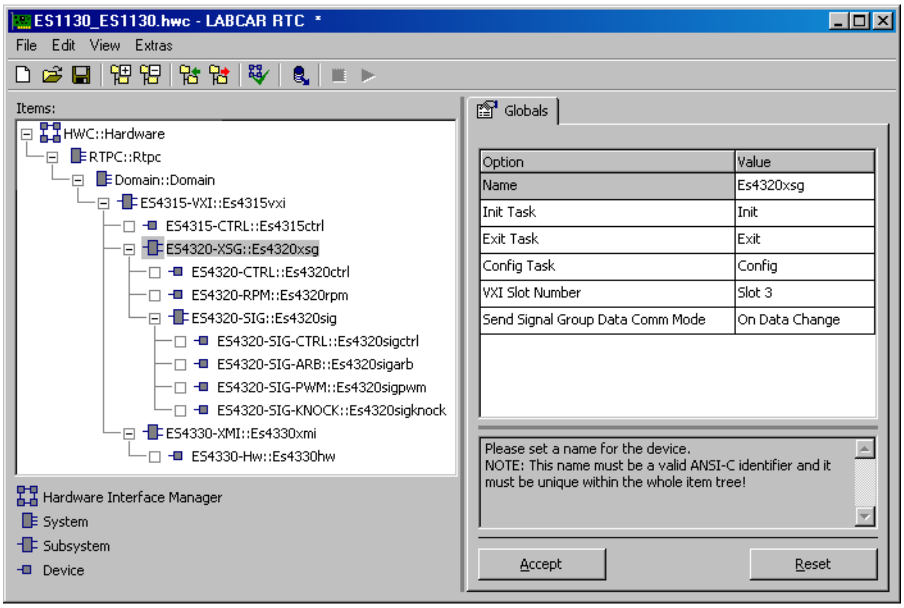


Fig. 17-5 The "Globals" Tab of the ES4320-XSG Subsystem

VXI Slot Number

This option is used to assign the ES4320 board to a VXI slot. The assignment of all the VXI boards to slots results in the actual arrangement of the boards in the ES4300 VXI housing. The entry selected must correspond to the actual arrangement in the ES4300 housing. If this is not the case, the RTIO driver is deactivated and the board described in the RTIO Editor cannot be used.

Send Signal Group Data Comm Mode

This option is used to select how the type of communication is handled by the ES1130 driver for signal groups which transfer data to the ES4320 board. The data can either be sent every time a task is activated ("Every Interval") or when a task is activated and signal group data is changed ("On Data Change"). Something as slight as a signal in the signal group having changed since the last successful communication constitutes a change.

If this option is set, it applies to all send signal groups of the relevant ES4320 board.

17.3.2 ES4320-CTRL Device

This section describes the specific options of the ES4320-CTRL device. This device contains signal groups and settings can be made which have an effect independently of individual RPM generators and SIG-ARB/SIG-PWM channels.

17.3.3 Globals (ES4320-CTRL Device)

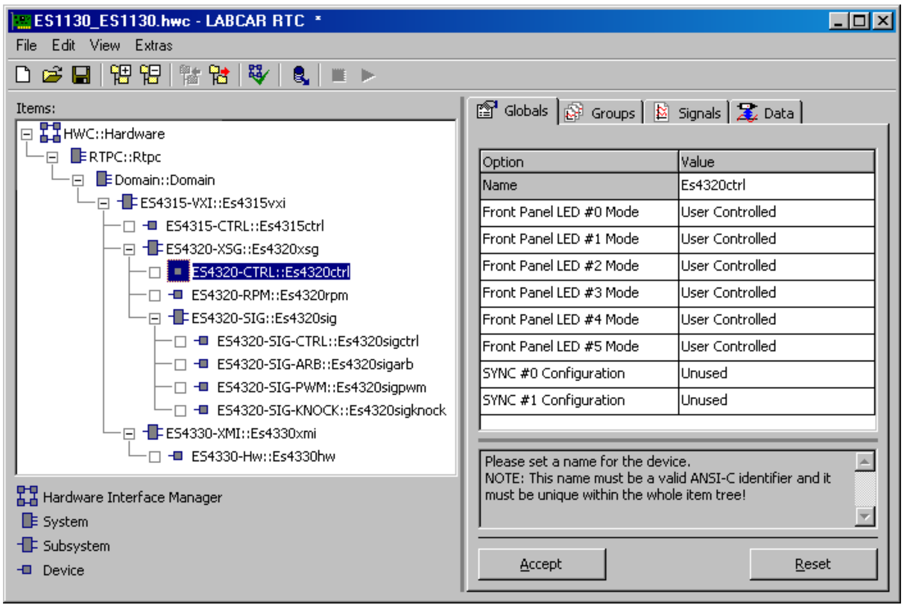


Fig. 17-6 The "Globals" Tab of the ES4320-CTRL Device

Front Panel LED #0 ... Front Panel LED #5 Mode

These options are used to specify the source controlling the relevant LEDs on the front panel of the ES4320 board. "User Controlled" makes it possible for the LEDs to be controlled by the model via the "StateLEDs" signal group in the same device. The LEDs can also be used to display the levels of the external disable inputs "Dis_0 ... Dis_3" and the triggers of different RPM trigger sources. As the triggers are very short, the LED changes its display state with every trigger event as opposed to the levels being shown.

Setting	Description
Disabled	The LED is not addressed.
User Controlled	The LED is controlled by specifications made by the user (corresponding signal LED_0 ... LED_5 of the signal group StateLEDs).
Dis_0 Level ... Dis_3 Level	The LED displays the level of the relevant external disable input (TTL high = LED on).

Setting	Description
RPM #0 Trg Toggle ... RPM #3 Trg Toggle	The LED changes its state with every trigger of the RPM signal of the selected RPM generator on this ES4320 board.
Sync #0 Trg Toggle ... Sync #1 Trg Toggle	The LED changes its state with every trigger of the selected Sync signal (i.e. the basic RPM signal is supplied by another system).
VXI-RPM #0 Trg Toggle ... VXI-RPM #1 Trg Toggle	The LED changes its state with every trigger of the RPM signal of the selected VXI-RPM signal (i.e. the basic RPM signal is generated by another ES4320 board in this VXI system).

Sync #0, Sync #1 Configuration

These options are used to determine the effect and direction of the SYNC0 and SYNC1 signals on the front panel. Each socket can either be used as an input or output for an RPM signal. When used as an output, an RPM signal generated on the board or one of the two VXI-RPM signals is issued. When used as an input, Sync #0 (Ext-RPM #0) or Sync #1 (Ext-RPM #1) can be provided with an external RPM signal and thus selected as a trigger or clock source for ARB channels. SYNC configuration settings cannot be changed during runtime.

Setting	Description
Disabled	SYNC is not used
Input	The external synchronization signal can be used to synchronize an RPM signal or as a direct synchronization signal of an arbitrary signal.
Output RPM #0	The signal of RPM generator 0 is connected to the "SYNC" connector on the front panel.
Output RPM #1	The signal of RPM generator 1 is connected to the "SYNC" connector on the front panel.
Output RPM #2	The signal of RPM generator 2 is connected to the "SYNC" connector on the front panel.
Output RPM #3	The signal of RPM generator 3 is connected to the "SYNC" connector on the front panel.
Output VXI-RPM #0	The RPM #0 signal of the VXI backplane is connected to the "SYNC" connector on the front panel.
Output VXI-RPM #1	The RPM #1 signal of the VXI backplane is connected to the "SYNC" connector on the front panel.

17.3.4 Groups (ES4320-CTRL Device)

The ES4320-CTRL device implements three signal groups which you can use for the following tasks.

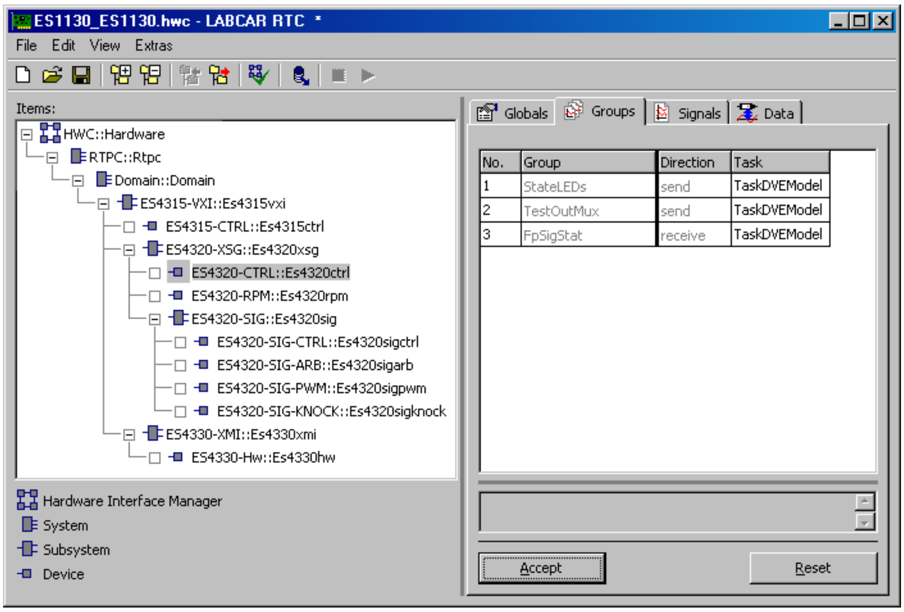


Fig. 17-7 The "Groups" Tab of the ES4320-CTRL Device

StateLEDs

This send signal group defines the state of one or more of the six LEDs on the front panel if the relevant LED is controlled by the model ("UserControlled" setting of the option "Front Panel LED #0 ... 5", refer to section 17.3.3 on page 462).

TestOutMux

This send signal group defines which signals are used as sources for the two analog and two TTL test outputs on the front panel.

FpSigStat

This receive signal group makes it possible to query the logical states of the trigger and disable inputs on the front panel.

17.3.5 Signals (ES4320-CTRL Device)

This section describes the signals of the ES4320-CTRL device.

Signal	Direction	Description
LED_0 ... LED_6	send	Specification of the states of the front panel LEDs in "UserDefined" mode
TestOutMuxAna_0 ("TST0")*, TestOutMuxAna_1 ("TST1")*	send	Selection of the sources for the two analog tests outputs on the front panel
TestOutMuxTTL_0 ("TST2")*, TestOutMuxTTL_1 ("TST3")*	send	Selection of the sources for the two TTL test outputs on the front panel
FpTrigger_0 ... FpTrigger_3	send	States of the trigger inputs on the front panel
ExtDisable_0 ... ExtDisable_3	send	States of the disable inputs on the front panel
* The relevant ES4320 Board outputs carrying these signals are specified in brackets		

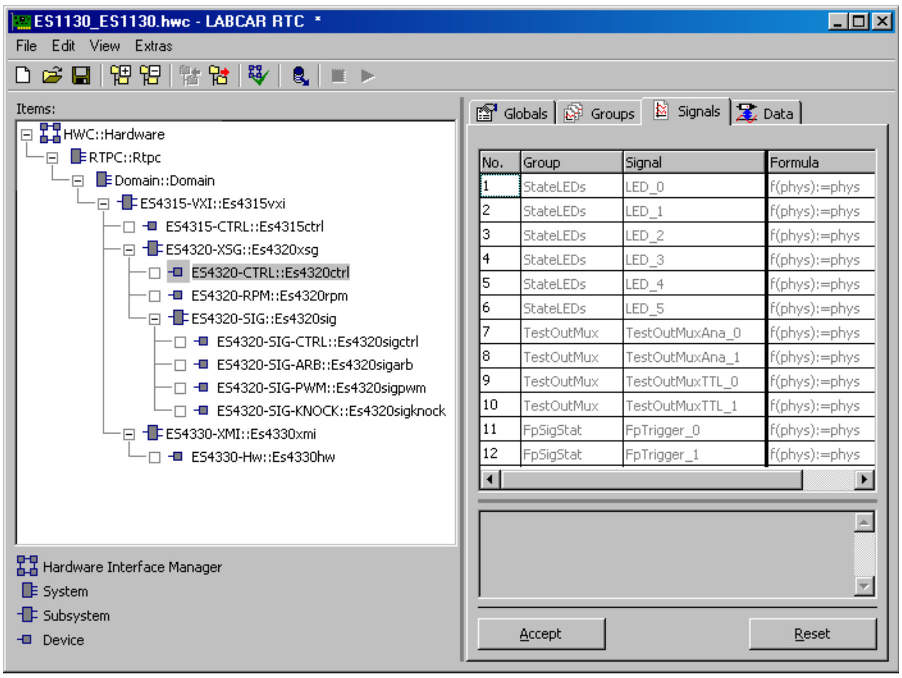


Fig. 17-8 The "Signals" Tab of the ES4320-CTRL Device

17.3.6 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
LED_0 ... LED_6	bool	false [0] = LED off true [1] = LED on
TestOutMuxAna_0 ("TST0")*, TestOutMuxAna_1 ("TST1")*	uint	0 ... 11 = analog signal of the PWM/ARB channel of ID 0 ... 11
TestOutMuxTTL_0 ("TST2")*, TestOutMuxTTL_1 ("TST3")*	uint	0, 1 = trigger from ExtRpm_0, ExtRpm_1 2, 3 = trigger from VxiRpm_0, VxiRpm_1 4, 5 = clock signal from ExtRpm_0, ExtRpm_1 6, 7 = clock signal from VxiRpm_0, VxiRpm_1 8 ... 11 = trigger signal from RPM generator #0 ... #3 12 ...15 = clock signal from the RPM generator #0 ... #3 of the board 16 ... 27 = local clock signal of the PWM/ARB channel of ID 0 ...11 32 ... 35 = additional triggers (Additional Trigger) 0 ... 3 40 ... 45 = internal test points
FpTrigger_0 ... FpTrigger_3	bool	false [0] = TTL low at the relevant input true [1] = TTL high at the relevant input
ExtDisable_0 ... ExtDisable_3	bool	false [0] = TTL low at the relevant input true [1] = TTL high at the relevant input
* The relevant ES4320 Board outputs carrying these signals are specified in brackets		

17.3.7 ES4320-RPM Device

This section describes the specific options of the ES4320-RPM device.

Signal groups are defined in this device and settings can be made which are necessary for the operation and configuration of the four RPM generators. Each of the four RPM generators is capable of generating an angle signal (RPM signal) consisting of clock cycle, direction and angle reference point. These RPM signals can now be used by an arbitrary channel on this board and/or can be made available to other boards (ES4320 or ES4330) in this VXI system via the VXI-RPM #0 and VXI-RPM #1 resources.

17.3.8 Globals (ES4320-RPM Device)

This tab allows you to make configuration settings which apply generally to all four RPM generators.

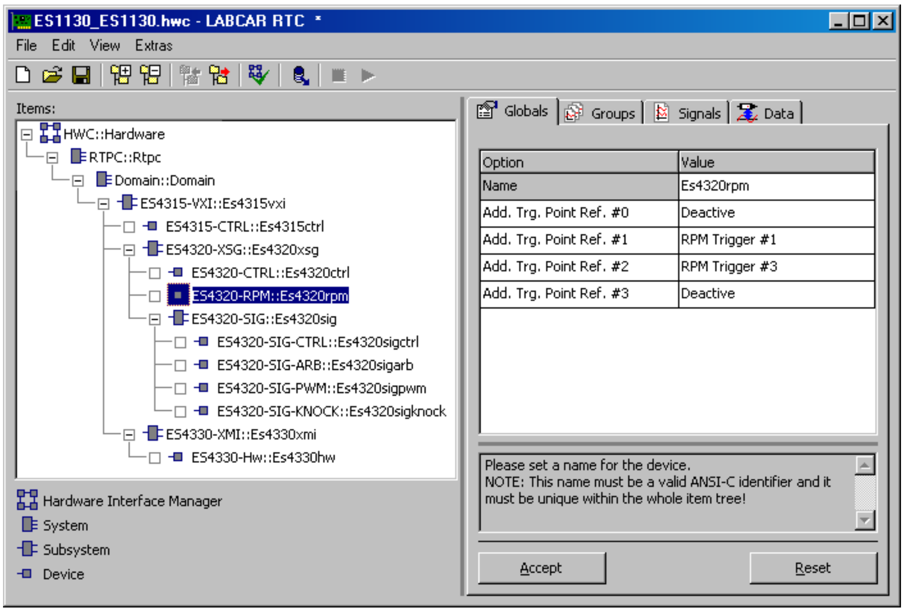


Fig. 17-9 The "Globals" Tab of the ES4320-RPM Device

Add. Trg. Point Ref. #0 ... Add Trg Point Ref. #3

This option is used to specify to which RPM generator the additional triggers refer. The additional trigger results from the zero reference point of the selected RPM generator and the phase shift defined for this additional trigger in the "AddTrgPhase" signal group.

Please note that the zero reference point of the RPM generator does not have to be the same as an individual RPM trigger as a phase shift can be specified between the zero reference point and the first trigger ("TrgPhase_<X>" signals in the "Ctrl_<X>" signal group of the RPM device).

Note

The generation of an additional trigger point is deactivated if a KNOCK-Ctrl item with the same ID is added (see "ES4320-KNOCK-Ctrl Device" on page 498)

17.3.9 Groups (ES4320-RPM Device)

The ES4320-RPM device implements signal groups for controlling and returning the angle position of the four RPM generators. This tab also contains the generator-specific configuration settings.

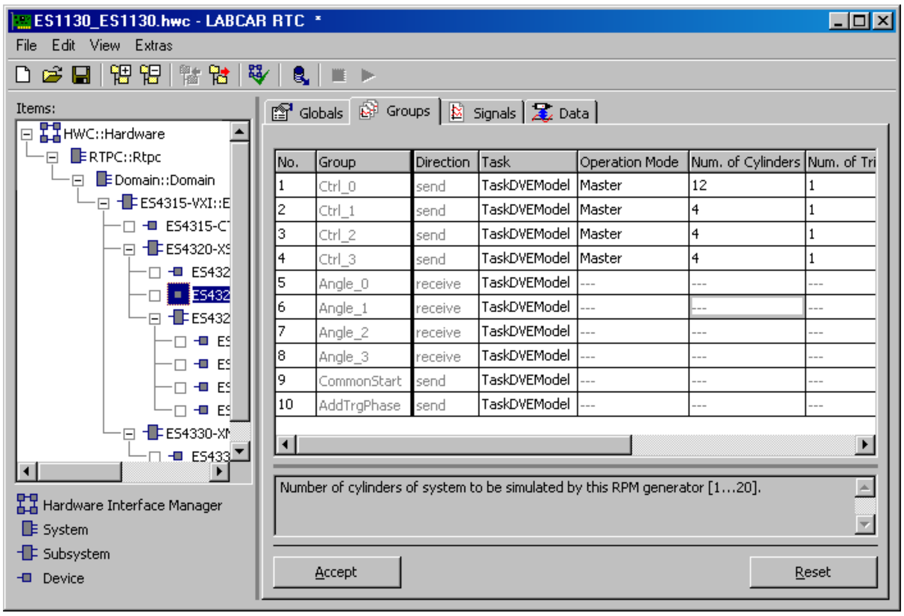


Fig. 17-10 The "Groups" Tab of the ES4320-RPM Device

Ctrl_0 ... Ctrl_3

The RPM generators #0 ... #3 are controlled with these send signal groups.

Angle_0 ... Angle_3

These receive signal groups provide the current angle information from the RPM generators #0 ... #3.

CommonStart

This send signal group makes it possible for several RPM generators to be started synchronously.

AddTrgPhase

This send signal group makes it possible to set the phase shift of the eight additional triggers. Setting options are assigned to the signal groups Ctrl_0 ... Ctrl_3. This means the configuration data assigned to the RPM generators #0 ... #3 can be set.

Task

This is where the activation time of the RPM generators between the host system and the ES4320 is defined.

Operation Mode

This option is used to define whether the relevant RPM generator is to work as a master generator or be synchronized by an external RPM signal.

Setting	Description
Master	The RPM generator generates the RPM signal itself.
Slave of RPM #0 RPM #1 RPM #2 RPM #3	The RPM generator takes the RPM signal of RPM generator #0 RPM generator #1 RPM generator #2 RPM generator #3
Slave of VXI-RPM #0 VXI-RPM #1	The RPM generator takes RPM signal #0 of the ES4300 VXI backplane RPM signal #1 of the ES4300 VXI backplane
Slave of Sync #0 Slave of Sync #1	The RPM generator takes the RPM signal which is supplied via the Sync0 connector on the front panel.* The RPM generator takes the RPM signal which is supplied via the Sync1 connector on the front panel.*

* For this purpose the "Sync #*n* Configuration" parameter in the ESXS4320-CTRL device has to be configured to input (see "Globals (ES4320-CTRL Device)" on page 462).

Num. of Cylinders

This parameter specifies the number of cylinders. This parameter is used for knock signal generation.

Num. of Triggers

This option is used to define the number of triggers (1...16) which are generated per period of the RPM signal (720 °CA). If there is more than one trigger, they are distributed equidistantly in terms of angle over the period of the RPM signal. Only one value can be set for "Num. of Triggers" whose whole-numbered multiple is "Period Length".

Period Length

This option is used to specify the number of clock cycles generated per period of the RPM signal. Only a whole-numbered multiple of "Num. of Triggers" can be set for "Period Length".

Disable Source

This option is used to select whether one of the four disable inputs Dis_0 ... Dis_3 is to be used on the front panel for the activation control of the RPM channel. The effect of the disable input on the RPM channel depends on the setting of the "Disable Mode" option.

Setting	Description
Deactive	The RPM generator is always active regardless of the state of the disable inputs.
Dis_0	Input Dis_0 controls the RPM generator
Dis_1	Input Dis_1 controls the RPM generator
Dis_2	Input Dis_2 controls the RPM generator
Dis_3	Input Dis_3 controls the RPM generator

Disable Mode

This option is used to determine what effect the specified disable input has on the RPM generator. The setting has no effect if "Deactive" is set for the "Disable Source" option.

Setting	Description
Cont. Low Active	The signal at the disable input is linked TTL low active with the clock cycle of the RPM generator. I.e. the RPM generator is stopped with TTL high level.
Cont. High Active	The signal at the disable input is linked TTL high active with the clock cycle of the RPM generator. I.e. the RPM generator is stopped with TTL low level.

Note

With an asynchronous start, the slave is not synchronized with the master until the next master trigger.

The following RPM parameter is ineffective in slave mode:

- EngSpeed_*n*

The following RPM parameters have to correspond to those of the master in slave mode:

- TrgEna_*n*
- StartAngle_*n*
- StartAngleEna_*n*

The following RPM parameters can be used freely in slave mode:

- StartMode_*n*
- StartFlag
- TrgPhase_*n*

Data Types and Value Ranges

Signal	Data Types	Possible Values
Num of Cylinders	uint8	2...20
Num of Triggers	uint8	1...16
Period Length	uint8	2...16384 (number of clock pulses per crank-shaft revolution)
Disable Source	uint8	See "Disable Source" on page 470
Disable Mode	uint8	See "Disable Mode" on page 470
* This setting is ineffective as soon as an RPM channel is configured as a slave. "Disable Source" is only possible at the master RPM.		

17.3.10 Signals (ES4320-RPM Device)

This section describes the signals of the ES4320-RPM device. The RPM generator-specific control signals are available individually for each RPM generator and are assigned the suffix _0, _1, _2 or _3.

Signal	Direction	Description
StartMode_0 ... StartMode_3	send	Defines whether the start of the RPM generator depends on the local or the common (global) start flag.
StartFlag_0 ... StartFlag_3	send	Local start flag for the relevant RPM channel.
EngSpeed_0 ... EngSpeed_3	send	Default value of angular speed of the RPM generator.
TrgPhase_0 ... TrgPhase_3	send	Default value for phase shift of the first trigger pulse.
StartAngle_0 ... StartAngle_3	send	Default value for start angle of the RPM generator. This is the angle value at which the generator can start or stop. The function is only effective if the relevant "StartAngleEna_<X>" flag is activated at the start or stop time. If this is not the case, starting or stopping respectively occurs when the angle is 0.0 °.
TrgEna_0 ... TrgEna_3	send	This signal means individual triggers of the RPM signal can be hidden. Bit 0 corresponds to the first trigger etc. The number of triggers per period is set by the "Num. of Triggers" option in the "Globals" tab.

Signal	Direction	Description
StartAngleEna_0 ... StartAngleEna_3	send	Flag for activating the defined start angle when starting / stopping the RPM generator #0 ... #3.
ActAngle_0 ... ActAngle_3	receive	Current angle position of the RPM generator #0 ... #3.
StartFlag	send	Global start flag for the synchronous start of several RPM generators.
Phase_0 ... Phase_7	send	Value of phase shift of the additional triggers in comparison to the base RPM trigger. The additional triggers can be used as trigger for the SIG-ARB channels.

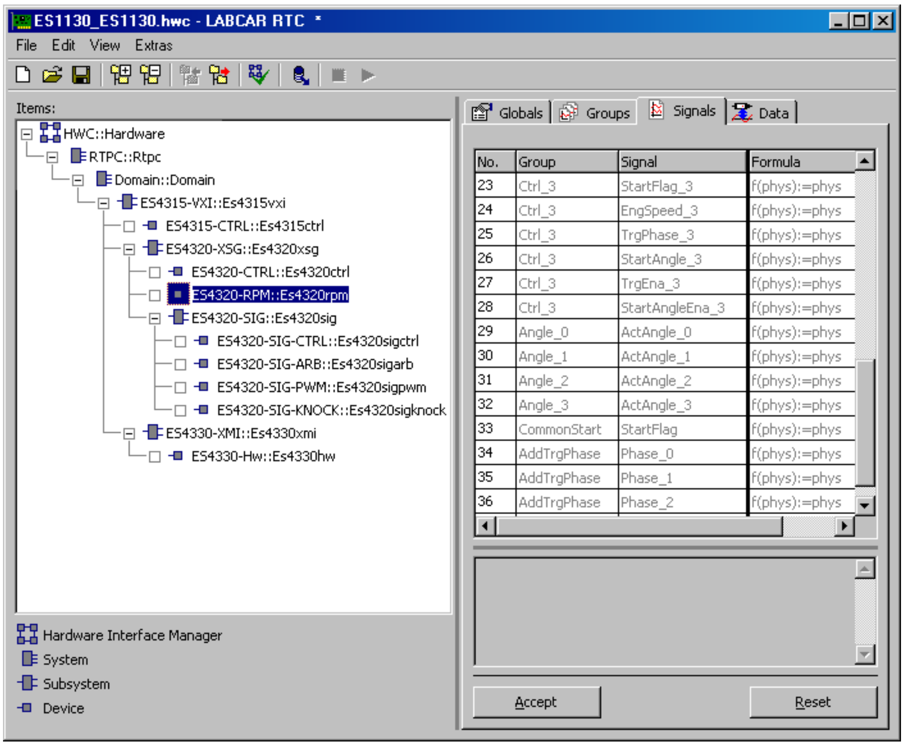


Fig. 17-11 The "Signals" Tab of the ES4320-RPM Device

17.3.11 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
StartMode_0 ... StartMode_3	uint8 / bool	0 [false]= local: start/stop is controlled by local start flag 1 [true] = global: start/stop is controlled by global start flag
StartFlag_0 ... StartFlag_3	uint8 / bool	0 [false] = RPM generator is stopped 1 [true] = RPM generator is started
EngSpeed_0 ... EngSpeed_3	real32	(0 ... 1.44 x10^8) / Period Length [1/min]
TrgPhase_0 ... TrgPhase_3	real32	0 ... 720.0° CA
StartAngle_0 ... StartAngle_3	real32	0 ... 720.0° CA
TrgEna_0 ... TrgEna_3	uint16	0x0001 ... 0xFFFF:
StartAngleEna_0 ... StartAngleEna_3	uint8 / bool	0 [false] = start/stop angle = 0.0° CA 1 [true] = set start/stop angle active
ActAngle_0 ... ActAngle_3	real32	0 ... 720.0° CA
StartFlag	uint8 / bool	0 [false] = relevant RPM generators are stopped 1 [true] = relevant RPM generators are started
Phase_0 ... Phase_7	real32	0 ... 720.0° CA

17.3.12 ES4320-SIG Subsystem

The ES4320-SIG subsystem has no setting possibilities. It is used as a container for items which describe the twelve output channels of the ES4320 board. Each of the twelve channels can either be used as a PWM or as an arbitrary channel. In the latter case, the values of the envelope curve are specified in a table. This means that various analog curve traces can be generated.

17.3.13 ES4320-SIG-CTRL Device

This section describes the signal groups of the ES4320-SIG-CTRL device. These signal groups can have an effect on all twelve channels regardless of whether these are configured as a PWM or arbitrary channel.

17.3.14 Groups (ES4320-SIG-CTRL Device)

The ES4320-SIG-CTRL device implements four signal groups which can be used for the synchronized start of several output channels. Each of the signal groups has an effect on three output channels. The three output channels must not, however, be configured with different functions, i.e., the PWM and arbitrary function must not be mixed within the group of the three. It is, however, possible to leave one or two channels unused, i.e. not configured.

If this requirement is not satisfied, the relevant "CommonStart" signal group has no effect.

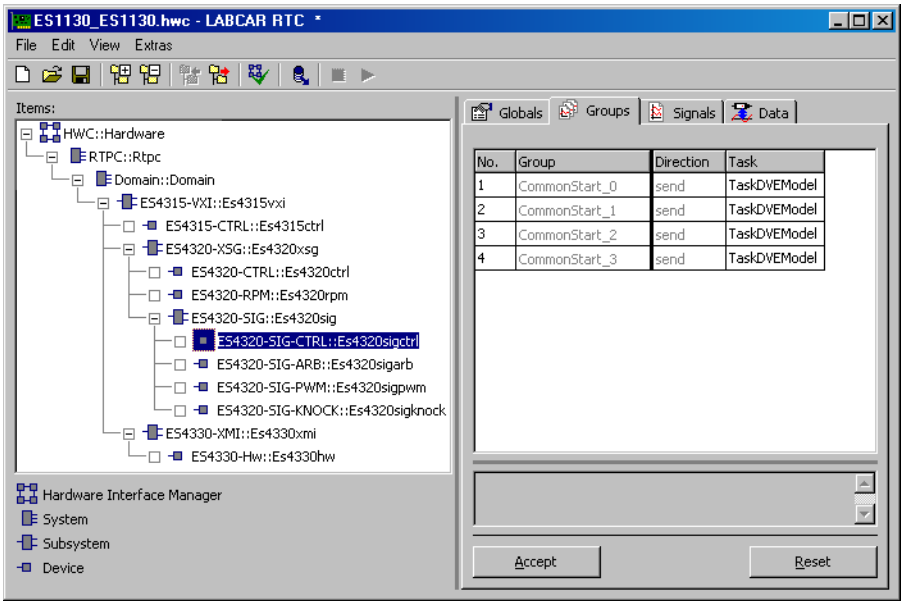


Fig. 17-12 The "Groups" Tab of the ES4320-SIG-CTRL Device

CommonStart_0

This send signal group is used for the synchronous start of the channels with the ID 0, 1, 2.

CommonStart_1

This send signal group is used for the synchronous start of the channels with the ID 3, 4, 5.

CommonStart_2

This send signal group is used for the synchronous start of the channels with the ID 6, 7, 8.

CommonStart_3

This send signal group is used for the synchronous start of the channels with the ID 9, 10, 11.

17.3.15 Signals (ES4320-SIG-CTRL Device)

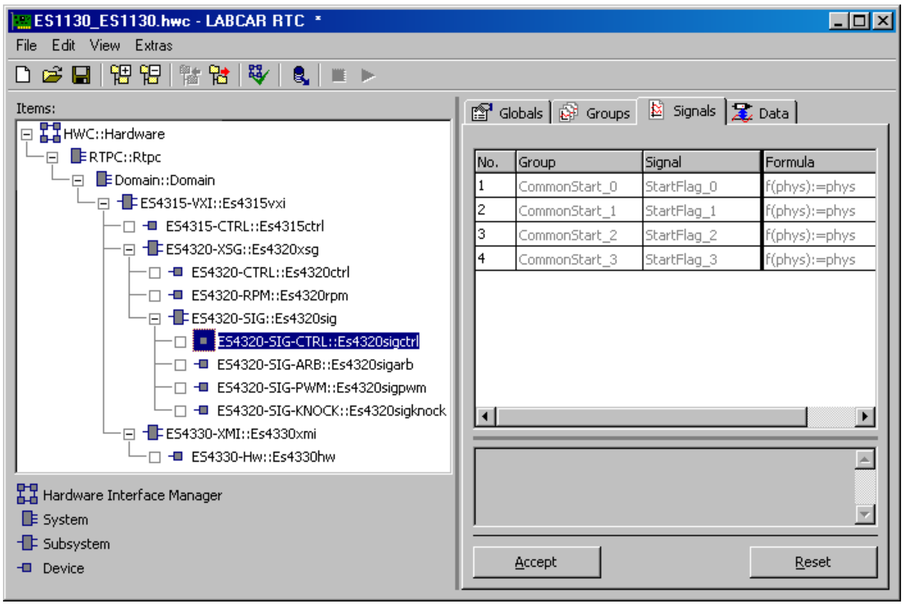


Fig. 17-13 The "Signals" Tab of the ES4320-SIG-CTRL Device

Signal	Direction	Description
StartFlag_0	send	Global start flag for synchronous start of the channels with the ID 0, 1, 2.
StartFlag_1	send	Global start flag for synchronous start of the channels with the ID 3, 4, 5.
StartFlag_2	send	Global start flag for synchronous start of the channels with the ID 6, 7, 8.
StartFlag_3	send	Global start flag for synchronous start of the channels with the ID 9, 10, 11.

17.3.16 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
StartFlag_0 ... StartFlag_3	uint / bool	0 [false] = relevant ARB or PWM channels are stopped 1 [true] = relevant ARB or PWM channels are started

17.3.17 Globals (ES4320-SIG-PWM Device)

This section describes the specific options and the signal group of the ES4320-SIG-PWM device. This device implements a PWM output using a hardware channel.

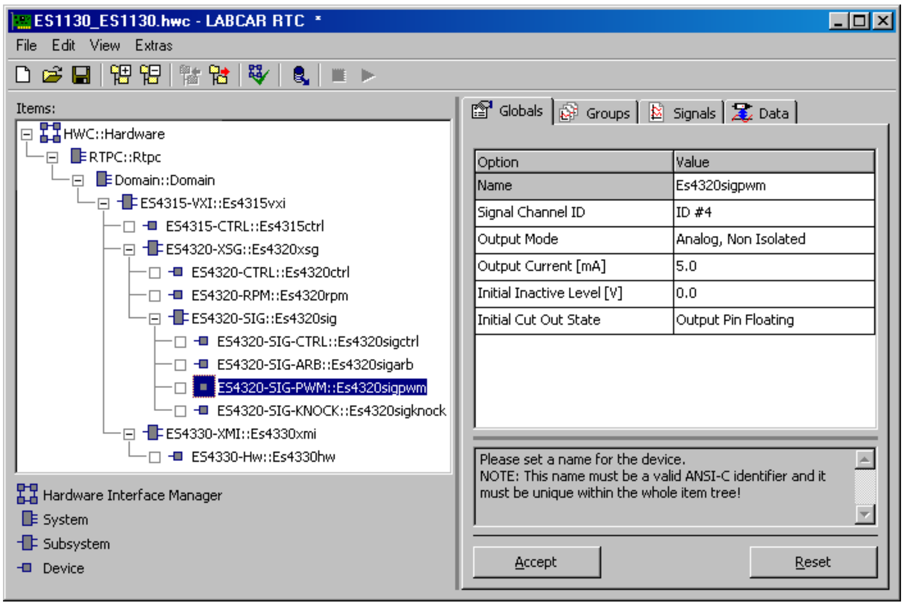


Fig. 17-14 The "Globals" Tab of the ES4320-SIG-PWM Device

The following configuration settings are possible in the "Globals" tab:

Signal Channel ID

This option is used to determine which of channels 0 ... 11 is to be used for the PWM function. A channel cannot be used more than once. The synchronous control of channels can only take place for channels 0, 1, 2 or 3, 4, 5 or 6, 7, 8 or 9, 10, 11.

Output Mode

This option is used to determine which output mode is to be used for the channel. There are two analog modes (with and without galvanic isolation). All other modes result in a binary output level whereby the binary value is formed using a Schmitt trigger with TTL input level. These are either pull-down (open collector) or pull-down/pull-up modes with reference to a battery voltage (battery nodes 0 ... 7 or UBatt A/UBatt B).

Output Current

This option is used to specify the value of the output current limitation in the range 5 ... 15 mA. This value denotes the maximum current in pull-up operation with binary modes (not open collector). The pull-down current is limited to 100 mA for the binary output modes.

Initial Inactive Level

This option is used to determine the initialization value of the low or inactive level of the output. This level is effective immediately after initialization before level values are specified using the "PWMOut" signal group. A state which corresponds to the threshold value of the Schmitt trigger results with binary output modes (see "Output Mode").

Initial Cut Out State

This option is used to specify a defined initialization state of the cutout relay. The state is set immediately after initialization before another value is specified for the relay by the "PWMOut" signal group.

17.3.18 Groups (ES4320-SIG-PWM Device)

The device implements a send signal group "PWMOut" to control the PWM channel.

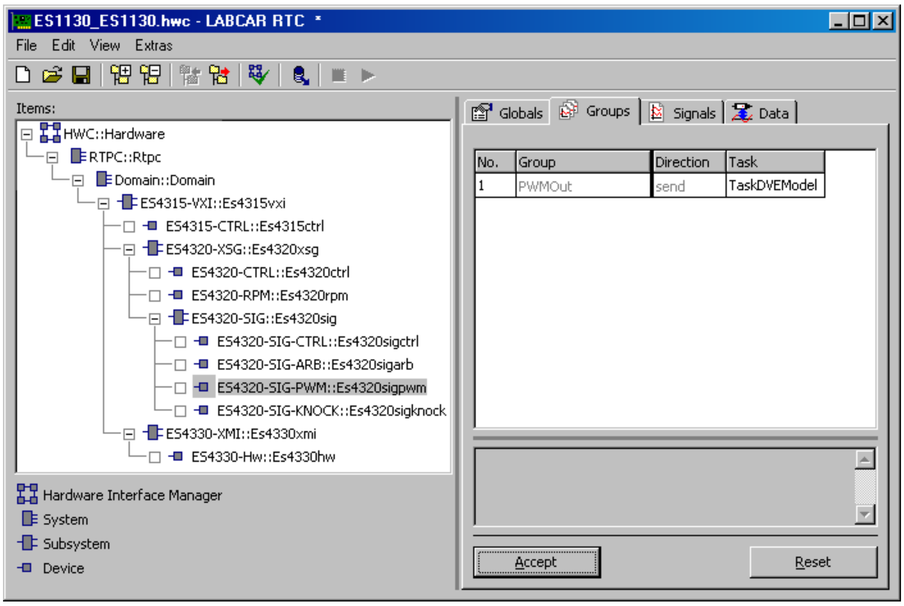


Fig. 17-15 The "Groups" Tab of the ES4320-SIG-PWM Device

17.3.19 Signals (ES4320-SIG-PWM Device)

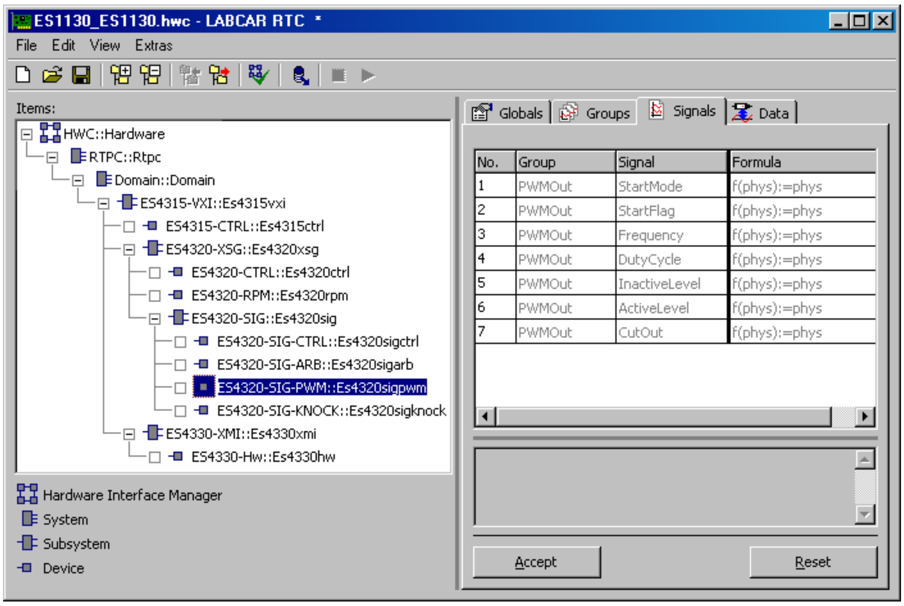


Fig. 17-16 The "Signals" Tab of the ES4320-SIG-PWM Device

Signal	Direction	Description
StartMode	send	Defines whether the start of the PWM channel depends on the local or a common (global) start flag
StartFlag	send	Local start flag for this PWM channel
Frequency	send	Frequency of the PWM signal
DutyCycle	send	Sampling rate of the PWM signal
InactiveLevel	send	Voltage value for the inactive (low) state of the PWM signal
ActiveLevel	send	Voltage value for the active (high) state of the PWM signal
CutOut	send	State of the cutout relay at the output

17.3.20 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
StartMode	uint8 / bool	0 [local] = PWM channel is controlled by the local start flag 1 [global] = PWM channel is controlled by the relevant global start flag
StartFlag	uint8 / bool	0 [false] = PWM channel is stopped 1 [true] = PWM channel is started
Frequency	real32	0.0 ...10000.0 Hz
DutyCycle	real32	0.0 ... 1.0
InactiveLevel	real32	0.0 ... 10.0 Volts
ActiveLevel	real32	0.0 ... 10.0 Volts
CutOut	uint8 / bool	0 [false] = cutout relay open 1 [true] = cutout relay closed

17.3.21 Globals (ES4320-SIG-ARB Device)

This section describes the specific options and the signal group of the ES4320-SIG-ARB device. This device makes it possible to generate an arbitrary output signal using a hardware channel.

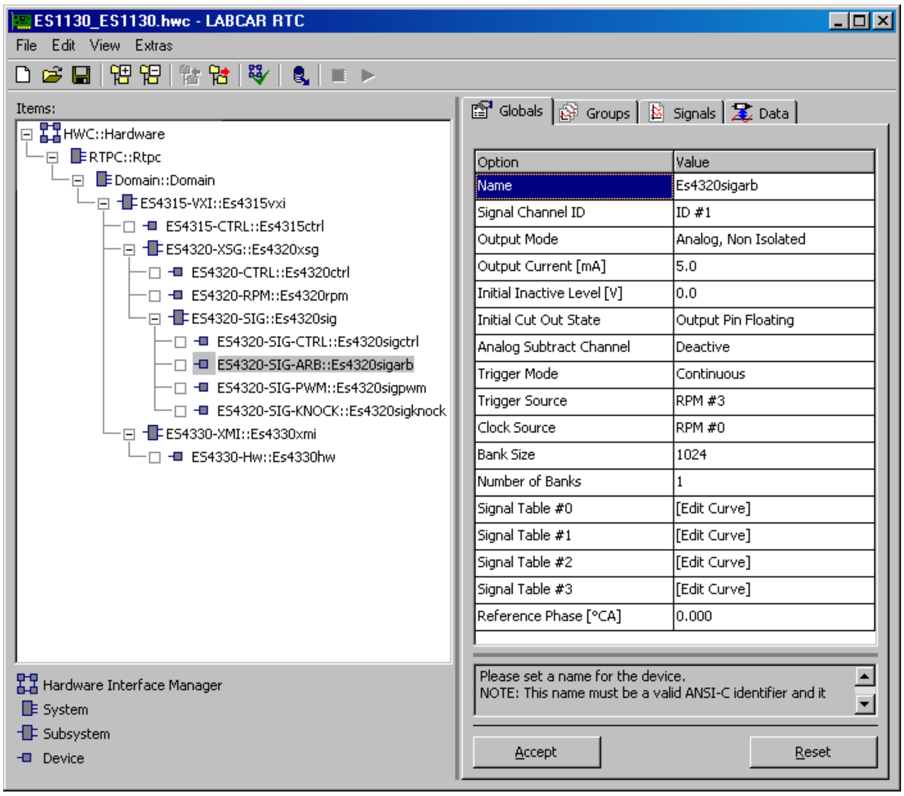


Fig. 17-17 The "Globals" Tab of the ES4320-SIG-ARB Device

The following configuration settings are possible in the "Globals" tab:

Signal Channel ID

This option is used to determine which of channels 0 ... 11 is to be used to generate the arbitrary output signal. A channel cannot be used more than once. A further limitation is that the synchronous control of channels can only take place for channels 0, 1, 2 or 3, 4, 5 or 6, 7, 8 or 9, 10, 11.

Output Mode

This option is used to determine which output mode is to be used for the channel.

There are two analog modes (with and without galvanic isolation). All other modes result in a binary output level whereby the binary value is formed using a Schmitt trigger with TTL input level. These are either pull-down (open collector) or pull-down/pull-up modes with reference to a battery voltage (battery nodes 0 ... 7 or UBatt A/UBatt B).

Setting	Description
Analog Non Isolated	Analog output driver without galvanic isolation. Specification of an external reference voltage possible.
Analog Isolated	Analog output driver, galvanic isolation
Open Collector	Open collector with pull-down current limitation of 100 mA
Pull-Up U-Batt-A Pull-Up U-Batt-B	Like Open Collector, but with pull-up to UBatt A or UBatt B with settable current limitation
Pull-Up Node #0 ... Pull-Up Node #7	Like Open Collector, but with pull-up to battery nodes #0 to #7 with settable current limitation

Output Current

This option is used to specify the value of the output current limitation in the range 5 ... 15 mA. This value denotes the maximum current in pull-up operation with binary modes (not open collector). The pull-down current is limited to 100 mA for the binary output modes.

Initial Inactive Level

This option is used to determine the initialization value of the low or inactive level of the output. This level is effective immediately after initialization before level values are specified using the "ArbOut" signal group. A state which corresponds to the threshold value of the Schmitt trigger results with binary output modes (see "Output Mode").

Initial Cut Out State

This option is used to specify a defined initialization state of the cutout relay. The state is set immediately after initialization before another value is specified for the relay by the "ArbOut" signal group.

Analog Subtract Channel

This option is used to determine whether a channel (or which one) is to be subtracted from the current channel. The voltage value of the selected channel is then subtracted on the analog side. If a binary output mode is set, the voltage difference results in a state at the output which corresponds to the threshold value of the Schmitt trigger. If the voltage value of another channel is not to be subtracted, the current channel or "Deactive" has to be set.

Trigger Mode

This option is used to determine the response of the arbitrary output signal when the signal bank contains fewer data samples than clock pulses made available between two RPM triggers. The content of the signal bank is then completely filled with discrete values before the next RPM trigger occurs.

If "Continuous" has been selected, the content of the signal bank is continuously issued cyclically after the last data sample. With "Wait for Trigger", however, the value of the last data sample remains at the output. Regardless of the previous state, the selected signal bank is filled with discrete values with the next trigger of the RPM signal.

Note

Please note that a phase shift of the arbitrary signal cannot be specified with Continuous trigger mode. The Reference Phase option and the PhaseRef signal of the ArbOut signal group in this device thus have no effect!

Trigger Source

This option is used to specify which signal source is to be used as a trigger. The trigger point is the rising edge of the selected signal. The following trigger sources are possible:

Setting	Description
RPM #0 ... RPM #3	The trigger signal of RPM generator 0 ... 3 of this ES4320 board is used.
Sync #0, Sync #1	The trigger signal of the RPM signal (SYNC0 or SYNC1) supplied on the front panel is used. This setting is only possible if SYNC0 or SYNC1 are configured as inputs.
VXI-RPM #0, VXI-RPM #1	The trigger signal of the RPM resource VXI RPM #0 or VXI RPM #1 available via VXI is used. These resources can be supplied by an RPM generator of an ES4320 board of the VXI system. The relevant setting is made in the ES4315-CTRL device.

Setting	Description
Add. Trigger #0 ... Add. Trigger #3	One of the additional triggers (Add. Trigger #0 ... Add. Trigger #3) defined in the ES4320-RPM device is used as the trigger source.
Front Panel Trigger #0 ... Front Panel Trigger #3	One of the four TRIG inputs of the front panel is used as a trigger signal. These signals must be supplied by an external source (high-active, TTL level).
SIG-Channel #0 ... SIG-Channel #11	The binary output value of a PWM or arbitrary channel of this ES4320 board is used as a trigger signal. Only one channel from the relevant group of six of the arbitrary channel can be selected. If, for example, this arbitrary channel uses the output channel ID #1, the output channel IDs #0, #2, #3, #4 or #5 can be used as trigger source.

Clock Source

This option is used to specify which signal source is to be used as clock cycle for filling the signal banks with discrete values. The following clock sources are available:

Setting	Description
RPM #0 ... RPM #3	The clock signal of the RPM generator 0 ... 3 is used.
Sync #0, Sync #1	The clock signal of the RPM signal (Sync #0 or Sync #1) supplied on the front panel is used. This setting is only possible if SYNC0 or SYNC1 are configured as inputs.
VXI-RPM #0, VXI-RPM #1	The trigger signal of the RPM resource VXI RPM #0 or VXI RPM #1 available via VXI is used. These resources can be supplied by an RPM generator of an ES4320 board of the VXI system. The relevant setting is made in the ES4315-CTRL device.
Local	The clock frequency defined by the "LocalClockFreq" signal of the "ArbOut" signal group in this device is used. No trigger is used. The value set using the "Trigger Source" option has no effect.
1 MHz	A fixed clock cycle of 1 MHz is used. No trigger is used. The value set using the "Trigger Source" option has no effect.

Bank Size

This option is used to define the number of data samples in an individual bank. This then corresponds to one period of the arbitrary signal. If the clock cycle and trigger are specified by an RPM generator, the value of "Bank Size" of the arbitrary channel must correspond to the value of "Period Length" of the RPM generator used.

Number of Banks

This option is used to specify the number of banks for this arbitrary channel. Basically one, two or four banks can be used - a maximum of 4 banks with a "Bank Size" of up to 8192, a maximum of 2 banks with a "Bank Size" of up to 16384.

But only one bank can be active at any one time, i.e. determine the voltage time curve of the output signal. The bank currently valid is selected with the "BankSelect" signal in the "ArbOut" send signal group of this item.

Signal Table #0 ... Signal Table #3

This option is used to define the data samples of the signal bank contents. A table editor opens when you click the input field "[Edit Curve]". You can now change the number of data samples and modify the data samples and data values. The X data samples must increase monotonously with a maximum of 1024 data samples being able to be defined. The absolute value range of the X or Z values is of no significance. When the signal banks to be output are calculated, the Z values are normalized to [0,1] and the X values are normalized to a period of the arbitrary signal. This means that only those data samples of the signal trace need to be specified in which the signal changes.

Reference Phase

This option is used to specify an additional phase shift between the RPM trigger and the signal trace of the selected signal bank. The total value of the phase shift results from the value set here and the value of the "PhaseRef" signal in the "ArbOut" signal group in this item. The value set here can be anywhere in the range -720.0 ° ... +720.0 °. This option is only effective with the trigger mode "Wait for Trigger". The value set here is of no significance with the trigger mode "Continuous".

17.3.22 Groups (ES4320-SIG-ARB Device)

The device implements a send signal group "ArbOut" to control the arbitrary channel.

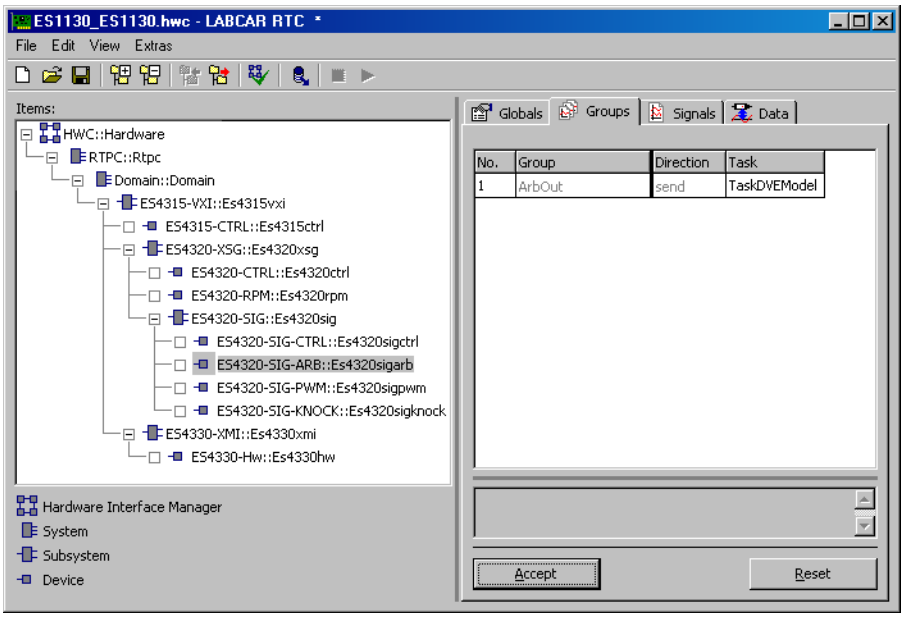


Fig. 17-18 The "Groups" Tab of the ES4320-SIG-ARB Device

17.3.23 Signals (ES4320-SIG-ARB Device)

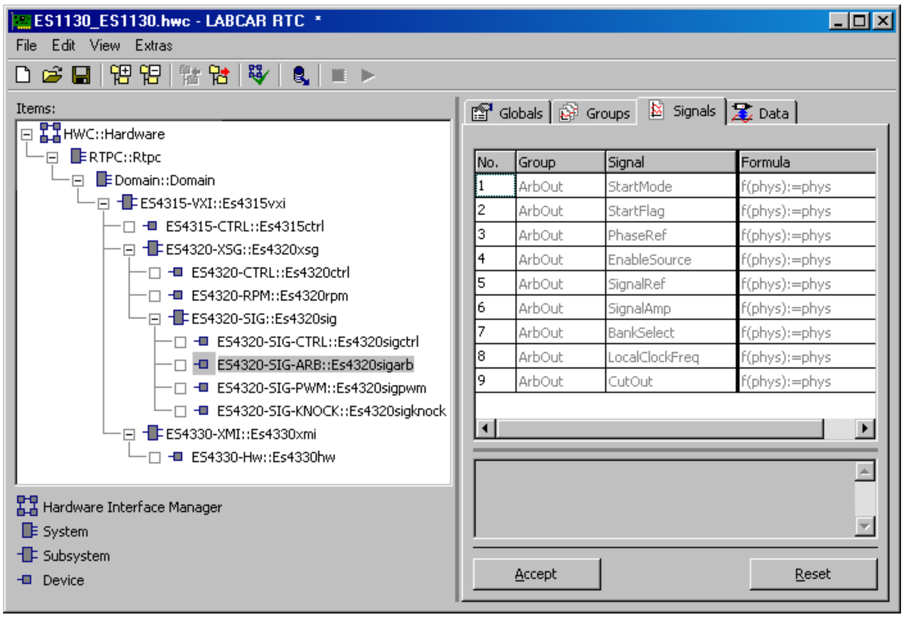


Fig. 17-19 The "Signals" Tab of the ES4320-SIG-ARB Device

Signal	Direction	Description
StartMode	send	Defines whether the start of the arbitrary channel depends on the local or a common (global) start flag.
StartFlag	send	Local start flag for this arbitrary channel
PhaseRef	send	Phase shift between RPM trigger and start of the signal table. Only effective with the trigger mode "Wait for Trigger". Any change to the variable has an immediate effect, i.e. in the calculation of the next data sample to be filled with discrete values.
EnableSource	send	This signal is used to control the clock cycle. The clock can be activated/deactivated unconditionally or it can be linked with the binary output value of another channel. This is, however, only possible with another channel from the relevant group of six (ID #0 ... ID #5 or. ID #6 ... ID #11).
SignalRef	send	Determines whether the internal default value "SignalAmp" or the externally applied reference voltage is used as reference voltage. The external reference voltage is always referenced to the analog ground of the board and is thus not potential free.

Signal	Direction	Description
SignalAmp	send	Default value for the internal reference voltage. This results in an output voltage range of 0 Volts up to this voltage value.
BankSelect	send	Selector which enables immediate toggling between the signal banks during running operation.
LocalClockFreq	send	Frequency of the local clock. This is used to fill the content of the signal bank with discrete values if "local" is set for the "Clock Source" option.
CutOut	send	State of the cutout relay at the output of the channel

17.3.24 Data Types and Value Ranges of the Signals

Signal	Data Types	Possible Values
StartMode	uint8/bool	0 [local] = arbitrary channel is controlled by local start flag 1 [global] = arbitrary channel is controlled by relevant global start flag
StartFlag	uint8/bool	0 [false] = arbitrary channel stopped 1 [true] = arbitrary channel started
PhaseRef	real32	-720.0 ° ... 720.0 ° CA
EnableSource	uint8	0 [off] = 1...6 [channel dependent] = 8 [unconditional] =
SignalRef	uint8/bool	0 [internal] = internal, settable reference voltage is used 1 [external] = external reference voltage is used
SignalAmp	real32	0 ... 10.0 Volts
BankSelect	uint8	0 = output of signal bank 0 1 = output of signal bank 1 2 = output of signal bank 2 3 = output of signal bank 3
LocalClockFreq	real32	0.0 ... 1.0 MHz
CutOut	uint8/bool	0 [false} = cutout relay open 1 [true] = cutout relay closed

17.3.25 ES4320-SIG-KNOCK Device

The knock generator is used to simulate signals from knock sensors. An uncontrolled combustion of the fuel-air mixture results in pressure oscillations that are captured by solid-borne sound sensors. The engine control unit can detect knocking within a certain frequency range depending on the amplitude of the oscillations.

Engine knocking can be generated separately for each cylinder, either randomly or according to a fixed pattern. The duration, signal shape, and amplitude modulation (envelope) of the knock signal can be freely defined.

Fig. 17-20 shows an example of a knock signal ("knock package"). It is usually a high frequent sine wave signal which amplitude is modulated by a slowly varying envelope signal.

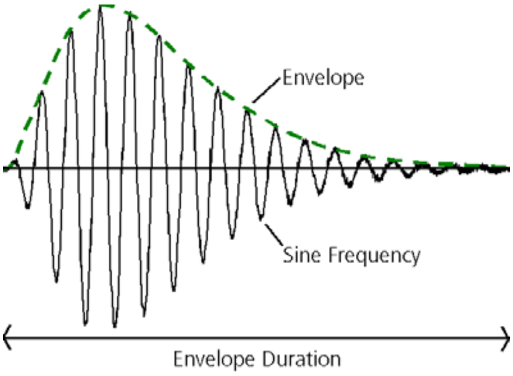


Fig. 17-20 Knock Signal (Solid Line) and Envelope (Dashed Line)

Note

The knock generator requires a specification of the number of available cylinders. This setting is done in the "Groups" tab of the ES4320-RPM Devices (refer to "Groups (ES4320-RPM Device)" on page 468).

17.3.26 Globals (ES4320-SIG-KNOCK Device)

This section describes the global options of the ES4320-SIG-KNOCK device.

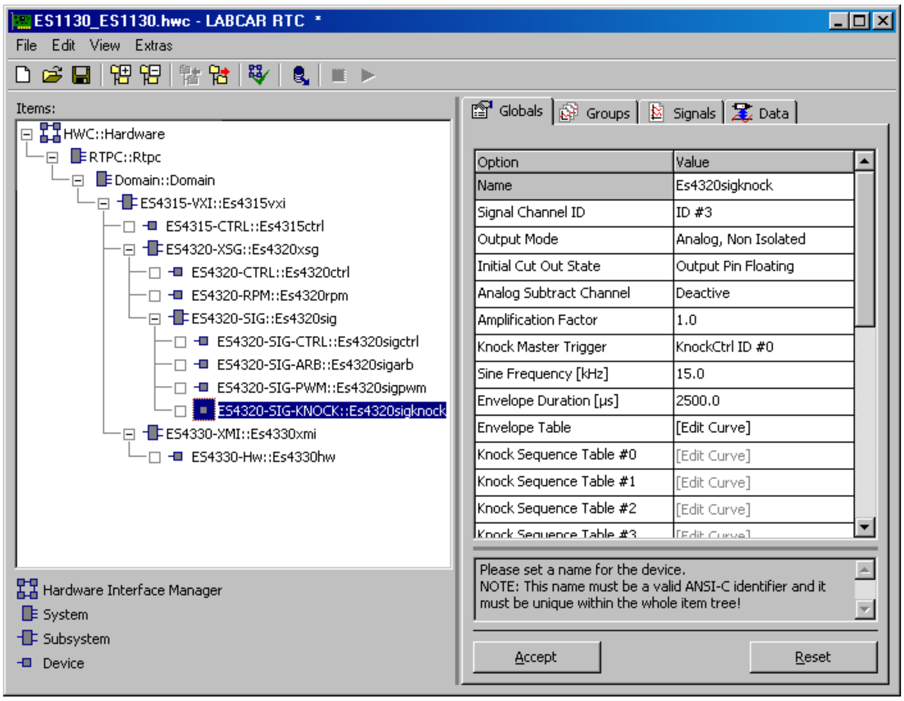


Fig. 17-21 The "Globals" Tab of the ES4320-SIG-KNOCK Device

Signal Channel ID

This option is used to determine which of channels 0 ... 11 is to be used to generate knock signals. A channel cannot be used more than once. A maximum of four channels of the ES4320 can be used for knock signal generation.

This parameter cannot be edited in runtime mode.

Output Mode

This option is used to determine which output mode is to be used for the channel. There are two analog modes (with and without galvanic isolation).

Initial Cut Out State

This option is used to specify a defined initialization state of the cutout relay. The state is set immediately after initialization before another value is specified for the relay by the "KnockOut" signal group.

Analog Subtract Channel

This option is used to determine whether a channel (or which one) is to be subtracted from the current knock generator channel. The voltage value of the selected channel is then subtracted on the analog side. If a binary output mode is set, the voltage difference results in a state at the output which corresponds to the threshold value of the Schmitt trigger. If the voltage value of another channel is not to be subtracted, the current channel or "Deactive" has to be set.

Amplification Factor

The low-amplitude knock signals are susceptible to interference during their transmission to the control unit. To avoid interference the signals are amplified on their way from the TS board knock generator output to the breakout box. An attenuator usually installed in the breakout box compensates the amplification so that knock signals with correct amplitudes are applied to the control unit. Hence the "Amplification Factor" parameter has to be set equal to the attenuation factor of the breakout box attenuator.

The "Amplification Factor" parameter can be edited in runtime mode.

Knock Master Trigger

This option is used to determine which knock master trigger is used to address the knock signal generator. The selection has to be made between KnockCtrl ID #0 ... #3.

The assignment between "KnockControl ID" and the RPM generator is defined in the ES4320-KNOCK-Ctrl item. All settings defined in the ES4320-RPM device (such as the number of cylinders or "Period Length") are used for knock signal generation. Depending on the number of cylinders, the relevant "Add. Trigger Point Ref." is used under "ES4320-RPM Device - Globals" (refer to page 467) to generate trigger points at the corresponding cylinder positions at which knock signal generation is activated. This is why the "Add. Trigger Point Ref." parameter is deactivated when using the RPM for knock signal generation (i.e. user settings are not possible).

Sine Frequency

This option is used to determine the basic sine frequency used for generating knock signals. This setting has an effect on all cylinders defined in the ES4320-SIG-KNOCK device.

The resulting knock signal results from the basic sinusoidal oscillation multiplied by the envelope duration and curve shape.

Envelope Duration [µs]

This option is used to determine how long the knock package is output.

A knock signal for one cylinder (see Fig. 17-20 on page 487) is defined by the envelope curve shape (defined by "Envelope Table"), the envelope duration (defined by "Envelope Duration") and the basic sine frequency of the knock signal (defined by "Sine Frequency").

Envelope Table

Clicking on the "Edit Curve" field opens the table editor for defining the envelope shape and duration of the knock signal. The x-axis specifies the time in µs. The x-axis value of the first point must be 0.0. The knock signal duration must be greater than or equal to the knock signal period. The z-axis specifies the normalized envelope shape, the z-values have to be in the range between 0.0 and 1.0.

The default envelope curve is:

$y = 25.0 * e^{2.0} * x^{2.0} * e^{(-10\,x)}$ where $25.0 * e^{2.0} = 184.7264025$

The "Envelope Table" can be edited in runtime mode.

Knock Sequence Table #0...#19

Clicking on the "Edit Curve" field opens the table editor to interactively specify the pattern for the occurrence of knock signals. The patterns are each 100 ignitions long. The x-axis has no meaning and is ignored. A z-value of 1 means that knocking is to be simulated for the current ignition, while 0 means that no knocking is to be simulated.

The "Knock Sequence" tables can be edited in runtime mode.

Tab. 17-1 summarizes the properties of the global knock generator RTIO parameters.

RTIO Parameter	Data Type	Editable in Run-time Mode	Comment
Signal Channel ID	uint32	No	Channel used for knock signal generation. Valid values: 0..11
Output Mode	uint8	Yes	Output mode for the channel: - analog without galv. isolation - analog with galv. isolation
Initial Cut Out State	uint8	Yes	Initialization state of the interrupt relay
Analog Subtract Channel	uint8	Yes	Definition of the channel to be subtracted from the current channel.
Amplification Factor	real32	Yes	Amplification factor to compensate the effect of an external attenuator. Default value: 1.0 (no external attenuator installed) Valid range: 0.1 ... 10.0
Knock Master Trigger	uint8	No	KnockCtrl ID #0 ... #3
Sine Frequency	real32	Yes	Frequency of the basic sinusoidal oscillation of the knock signal in kHz. Default value: 15 kHz Valid range: 1 kHz - 20 kHz
Envelope Duration [µs]	real32	Yes	Envelope duration of the knock signal Default value: 2500 µs Valid range: 2 µs - 12000 µs
Envelope Table	1-dim. Table	Yes	Normalized envelope curve for the knock signal. x-axis: no. of the breakpoint z-value range: 0.0 ... 1.0
Knock Sequence Table #0 ... Knock Sequence Table #19	1-dim. Table	Yes	Knock patterns. The patterns are each 100 ignitions long. z-value 1 means that knocking is to be simulated for the current ignition. z-value 0 means that no knocking is to be simulated. x-values are ignored.

Tab. 17-1 Global Parameters of the ES4320-SIG-KNOCK Device

17.3.27 Groups (ES4320-SIG-KNOCK Device)

This section describes the signal-group-specific options of the ES4320-SIG-KNOCK device.

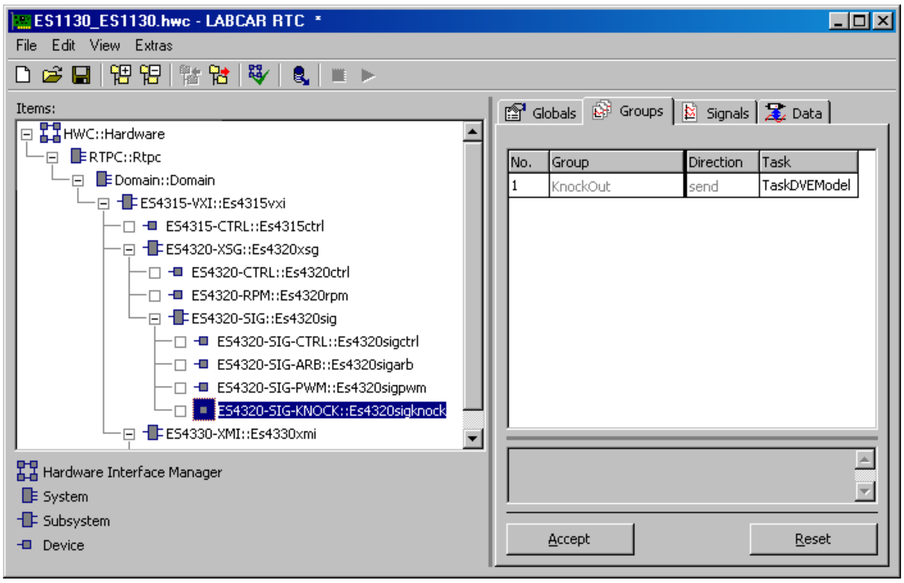


Fig. 17-22 The "Groups" Tab of the ES4320-SIG-KNOCK Device

KnockOut

KnockOut is the task in which knocking is output.

17.3.28 Signals (ES4320-SIG-KNOCK Device)

This section describes the signals with which the knock generator is controlled.

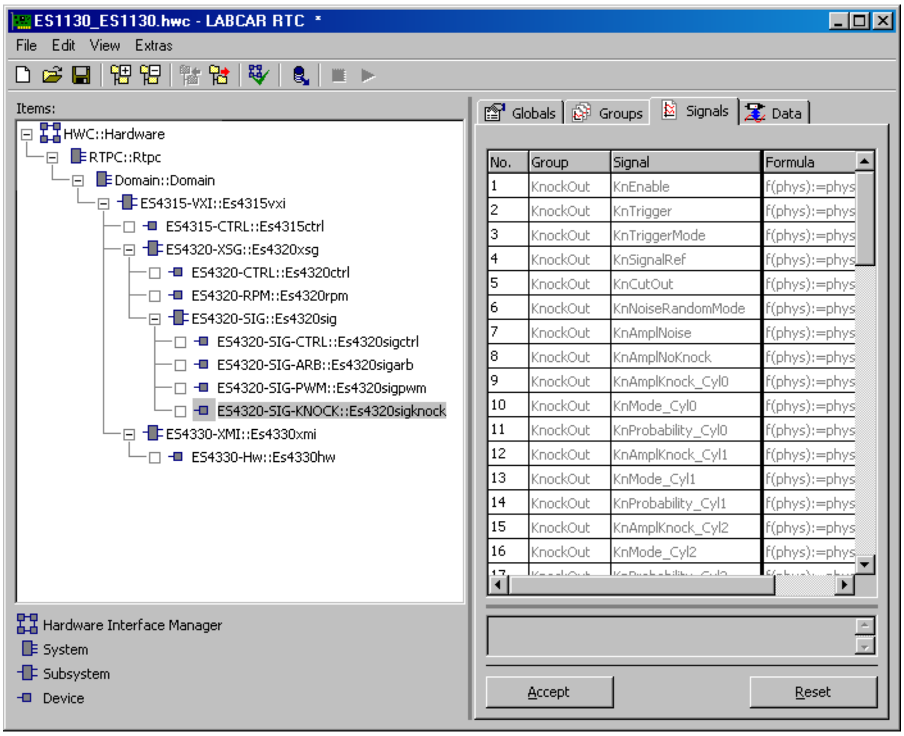


Fig. 17-23 The "Signals" Tab of the ES4320-SIG-KNOCK Device

KnEnable

The "KnEnable" RTIO signal is used to switch the knock generator on and off. A signal value of 0 switches the knock generator off; a signal value of 1 switches the knock generator on.

KnTrigger

A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during the time period covering the following 100 engine ignitions per cylinder.

The "KnTrigger" signal is only active when the knock generator is operated in trigger mode "Single Shot". The trigger mode is set using the "KnTriggerMode" RTIO signal (see page 493).

KnTriggerMode

The "KnTriggerMode" RTIO signal is used to control the knock generator trigger mode.

If the signal is set to 0 the knock generator operates in "Single Shot" trigger mode. A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during a time period covering the following 100 engine ignitions per cylinder. After that, knocking simulation is suspended until the next trigger occurs.

If the signal is set to 1, the knock generator operates in "Continuous" mode. In "Continuous" mode, knocking simulation is permanently active without trigger control. The "KnTrigger" RTIO signal is inactive in this trigger mode.

KnSignalRef

The "KnSignalRef" signal determines the source of the reference voltage of the D/A converter of the knock generator. If the signal value is set to 0, the internal 10 V reference voltage is selected. If the signal value is set to 1, the D/A converter reference voltage is supplied externally.

Note

If you use the internal reference voltage, the "KnAmplNoKnock", "KnAmplKnock_Cyln" and "KnAmplNoise" RTIO signal amplitudes refer to a reference voltage of 10 V. The signal values are limited to the range 0.0 ... 10.0/Amplification Factor.
If you use an external reference voltage, the "KnAmplNoKnock", "KnAmplKnock_Cyln" and "KnAmplNoise" RTIO signal amplitudes indicate a factor which is multiplied with the external reference voltage. The signal values are limited to the range 0.0 ... 1.0/Amplification Factor.
The external reference voltage always refers to the analog ground of the board and is therefore not potential-free!

KnCutOut

The "KnCutOut" signal is used to switch the knock generator output on and off. A signal value of 0 switches the channel output off; a signal value of 1 switches the channel output on.

KnNoiseRandomMode

The "KnNoiseRandom Mode" signal determines whether the modulation of the knock signals is generated individually for each cylinder (= 0) or as a pattern for all cylinders (= 1) during one revolution of the engine.

KnAmplNoise

Many control units interpret multiple occurrences of knock signals with the same amplitudes as sensor disturbances. To avoid this, the simulated knock signals can be modulated in terms of their amplitude using a random generator.

The "KnAmplNoise" RTIO signal sets the amplitude with which the knock signal is modulated randomly. This RTIO signal is used to simulate statistical fluctuations during combustion and thus of the knock sensor signals.

In mode "random", the knock signal is described by the following equation:

```
A(x) = (KnAmplKnock_Cyln - KnAmplNoKnock) * KnProbability
+ KnAmplNoKnock + KnAmplNoise * Random Value[0..1]
```

In mode "sequence":

```
A(x) = (KnAmplKnock_Cyln - KnAmplNoKnock) * KnockSequence
Table #n + KnAmplNoKnock + KnAmplNoise * Random
Value[0..1]
```

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnAmplNoKnock

Knock signal amplitude for controlled combustion. The "KnAmplNoKnock" RTIO signal sets the amplitude of the knock signal, if no knock signal is to be generated for the current ignition.

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnAmplKnock_Cyl0 ... KnAmplKnock_Cyl19

Knock signal amplitude for uncontrolled combustion. The "KnAmplKnock_Cyl*n*" RTIO signals set the amplitude of the knock signal of cylinder *n* if a knock signal is to be generated for the current ignition.

The knock amplitude is generated from the two signals in accordance with the knock probability "KnProbability_Cyl*n*".

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnMode_Cyl0 ... KnMode_Cyl19

The "KnMode_Cyl0" to "KnModeCyl19" RTIO signals specify the knock generator operating mode for the associated cylinder.

- 0 = disabled

  No knock signal generation for the relevant cylinder.
- 1 = sequence

  A knock signal is generated for the engine revolutions defined with 1 in "Knock Sequence Table #*n*".
- 2 = random

  A knock signal is generated with every engine revolution in accordance with the knock probability "KnProbability_Cyl*n*".

KnProbability_Cyl0 ... KnProbability_Cyl19

The "KnProbability_Cyl0" to "KnProbability_Cyl19" RTIO signals set the probability that uncontrolled combustion occurs during the current ignition. These RTIO signals are only active if the associated "KnMode" RTIO signal is set to "Random Mode".

The valid signal range is 0.0 to 1.0. A value of 0.0 means that uncontrolled combustion never occurs; a value of 1.0 means that uncontrolled combustion occurs during every ignition of the associated cylinder.

Note

The indices 0 to 19 appended to RTIO signals "KnMode_Cyl" and "KnProbability_Cyl" indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 17-2 summarizes all signals of the ES4320-SIG-KNOCK device.

RTIO Signal	Data Type	Editable in Runtime Mode	Comment
KnEnable	uint8 / bool	yes	Enable / disable knock generator. 0: disabled 1: enabled
KnTrigger	uint8 / bool	yes	Knock generator trigger signal. A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during the time period covering the following 100 engine ignitions per cylinder. Only active in trigger mode "single shot".
KnTriggerMode	uint8 / bool	yes	Knock generator trigger mode. 0: "single shot" trigger mode. 1: "continuous" mode.
KnSignalRef	uint8 / bool	yes	D/A converter reference voltage for knock generator output 0: internal 10 V reference voltage 1: external reference voltage
KnCutOut	uint8 / bool	yes	Switch on / off knock generator output channel. 0: output switched off 1: output switched on
KnNoiseRandomMode	uint8 / bool	yes	Type of knock signal modulation 0: for each cylinder individually 1: one pattern for all cylinders
KnAmplNoise	real32	yes	Amplitude (in volts) with which the knock signal is modulated randomly. Valid value range: Internal reference voltage: 0.0 ... 10.0/amplification factor External reference voltage: 0.0 ... 1.0/amplification factor
KnAmplNoKnock	real32	yes	Knock signal amplitude for controlled combustion. Valid value range: Internal reference voltage: 0.0 ... 10.0/amplification factor External reference voltage: 0.0 ... 1.0/amplification factor

RTIO Signal	Data Type	Editable in Runtime Mode	Comment
KnAmplKnock_Cyl0 ... KnAmplKnock_Cyl19	real32	yes	Knock signal amplitude for uncontrolled combustion. Valid value range: Internal reference voltage: 0.0 ... 10.0/amplification factor External reference voltage: 0.0 ... 1.0/amplification factor
KnMode_Cyl0 ... KnMode_Cyl19	uint8	yes	Knock generator operating mode for the associated cylinder. 0: Disabled 1: Pattern mode 2: Random mode
KnProbability_Cyl0 ... KnProbability_Cyl19	real32	yes	Probability that uncontrolled combustion occurs during the current ignition. Only active in knock mode "Random". Valid range: 0.0 (no knocking) ... 1.0 (always knocking)

Tab. 17-2 Signals of the ES4320-SIG-KNOCK Device

17.3.29 ES4320-KNOCK-Ctrl Device

To operate a knock generator, a ES4320-KNOCK-Ctrl device must be selected and configured for triggering purposes. Up to 4 independent Knock-Control units are available on the ES4320.

A trigger signal has to be generated for each cylinder of an engine, for which knocking is to be simulated, on the 720° CA level. The knock generator is supplied with the trigger signal by one of the "Additional Trigger" lines.

Note

The additional trigger "Add. Trigger #n" is always used for the trigger signal line with the "KnockCtrl ID" ID #n. The relevant additional trigger is then no longer available for other purposes once a KNOCK-Ctrl device with the relevant ID has been added.
The relevant entry is deactivated in the "Globals" tab of the ES4320-RPM device.

17.3.30 Globals (ES4320-KNOCK-Ctrl Device)

This section describes the global settings of the ES4320-KNOCK-Ctrl device.

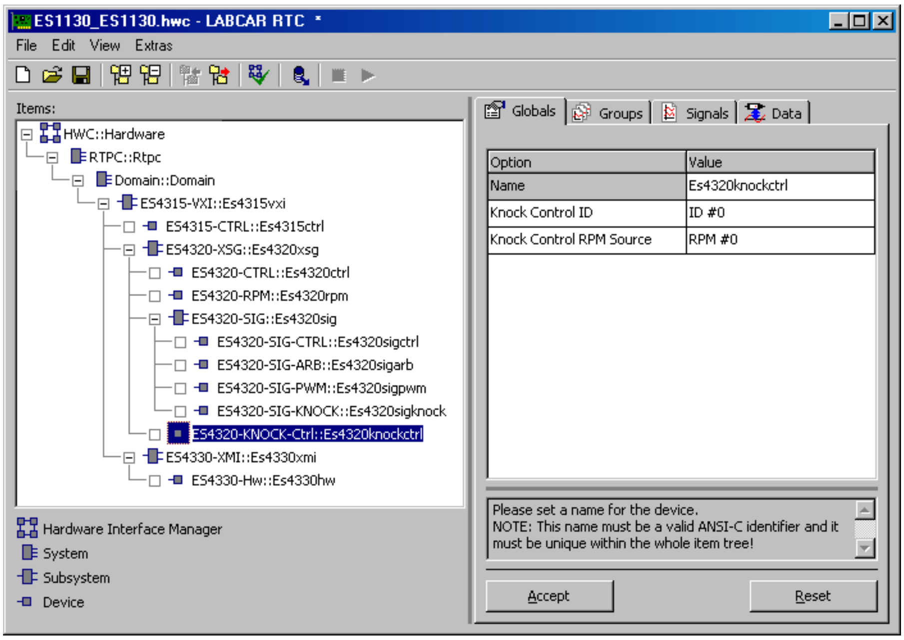


Fig. 17-24 The "Globals" Tab of the ES4320-KNOCK-Ctrl Device

Knock Control ID

This option is used to determine which ID the KNOCK-Ctrl device has. The reference to the RPM generator used is created using this ID with the "Knock Master Trigger" parameter of the ES4320-SIG-KNOCK device.

Possible settings are: ID #0 ... ID #3

This parameter cannot be edited in runtime mode.

Knock Control RPM Source

This option is used to determine which RPM generator is used for knock signal generation. This can be both the master and the slave.

Possible settings are: RPM #0 ... RPM #3

This parameter cannot be edited in runtime mode.

17.3.31 Groups (ES4320-KNOCK-Ctrl Device)

The ES4320-KNOCK-Ctrl device implements a "KnockCtrl" signal group which is used for real-time communication.

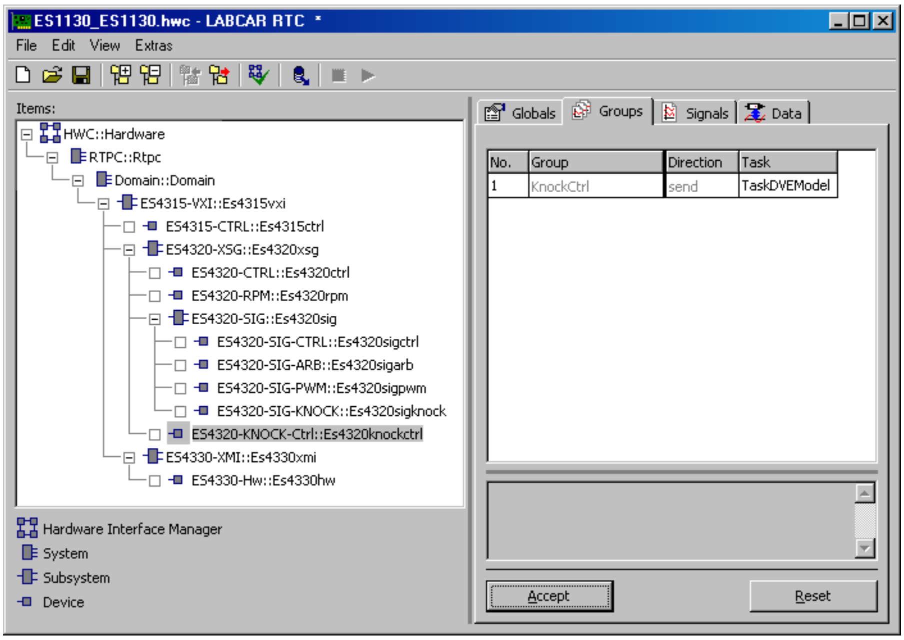


Fig. 17-25 The "Groups" Tab of the ES4320-KNOCK-Ctrl Device

17.3.32 Signals (ES4320-KNOCK-Ctrl Device)

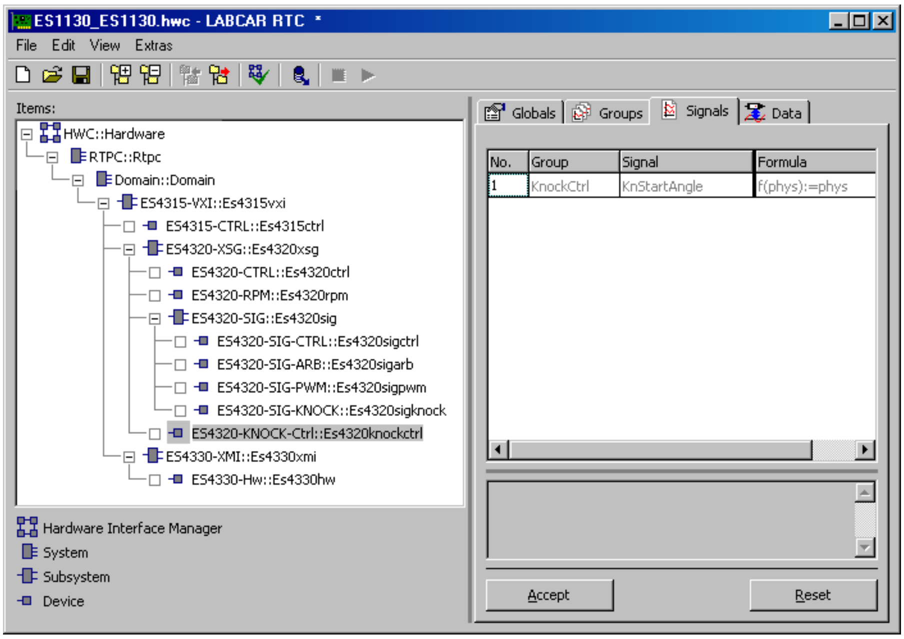


Fig. 17-26 The "Signals" Tab of the ES4320-KNOCK-Ctrl Device

KnStartAngle

This is the default value for the phase shift of knock signal generation.

Signal	Data Type	Possible Values
KnStartAngle	real32	-720° ... +720° CA

17.3.33 ES4320-Misfire Device

The misfire generator simulates a drop in engine speed after misfiring and is thus suitable for testing ECUs that recognize misfiring thanks to the drop in engine speed that follows.

Fig. 17-27 shows the connection between misfiring and a drop in engine speed.

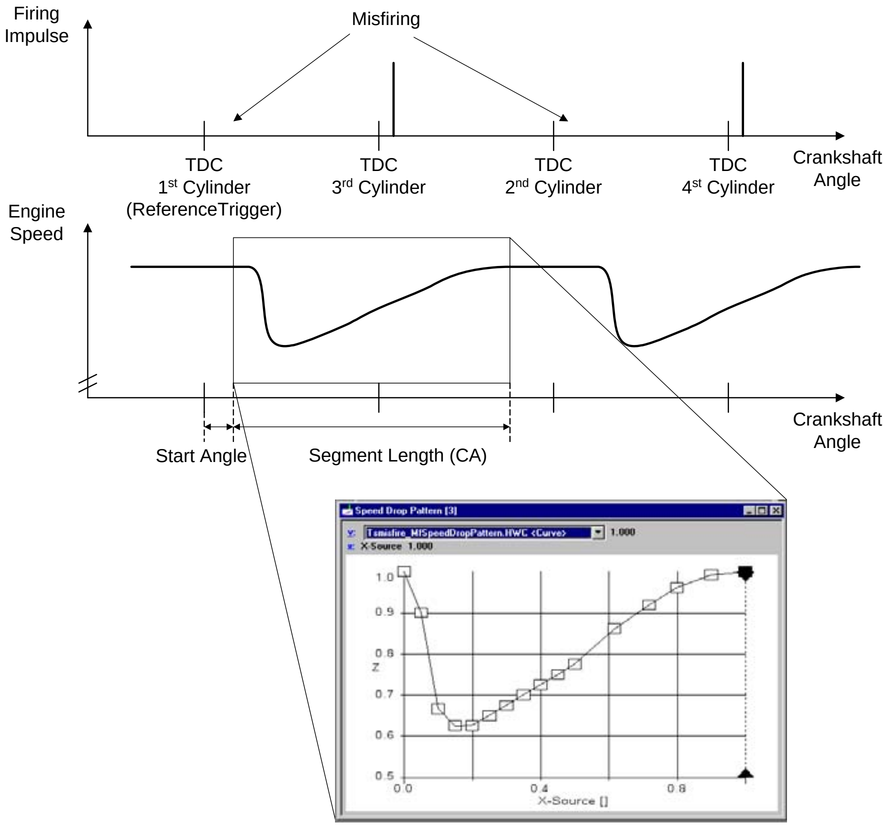


Fig. 17-27 Misfire and Drop in Engine Speed

Misfiring can be simulated for each individual cylinder for a maximum of twenty cylinders. Misfiring can be produced randomly with a probability that can be specified or according to a certain pattern specified separately for each cylinder. These patterns have a maximum length of 100 ignitions per cylinder and can be repeated as often as required.

It is also possible to specify a factor for each cylinder which specifies the drop in engine speed for each individual cylinder.

Note

The misfire generator requires a specification of the number of available cylinders. This setting is done in the "Groups" tab of the ES4320-RPM device (refer to "Groups (ES4320-RPM Device)" on page 468).

17.3.34 Globals (ES4320-Misfire Device)

This section describes the global options of the ES4320-Misfire device.

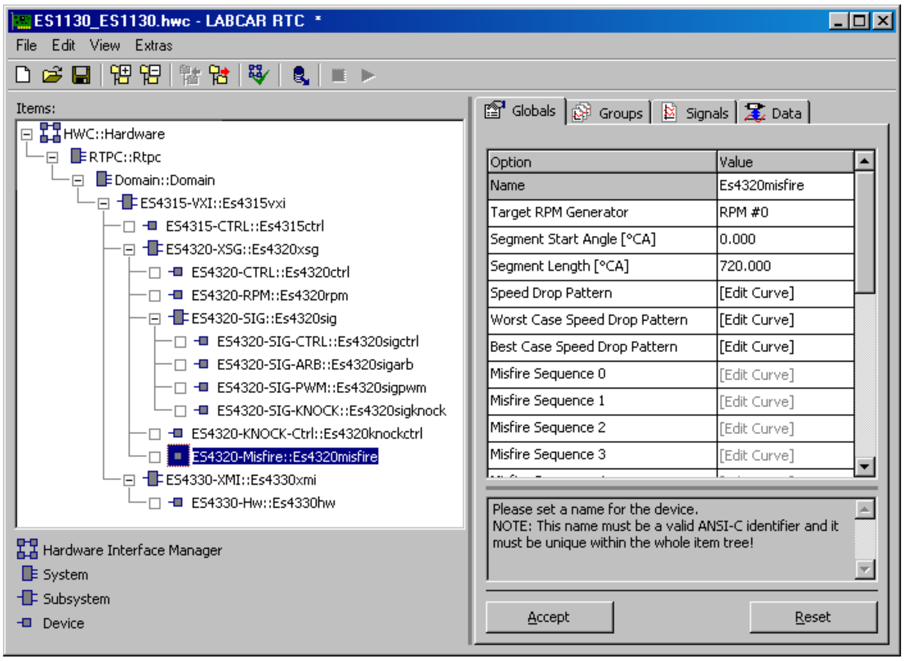


Fig. 17-28 The "Globals" Tab of the ES4320-Misfire Device

Target RPM Generator

The "Target RPM Generator" parameter is used to determine the RPM generator on which misfiring is to be simulated.

Segment Start Angle [°CA]

The "Segment Start Angle" parameter is used to define the angle between the top dead center of the cylinder and the start of the drop in engine speed due to misfiring (see Fig. 17-27 on page 501).

The "Segment Start Angle" parameter is defined in °CA and can be set to values between -720.0 °CA and 720.0 °CA. A positive start angle means that the start is below top dead center.

This parameter can be edited in runtime mode.

Segment Length [°CA]

The "Segment Length" parameter defines how long (in °CA) the drop in engine speed lasts (see Fig. 17-27 on page 501).

The "Segment Length" parameter is defined in °CA and can be set to values between 0 °CA and 720.0 °CA.

This parameter can be edited in runtime mode.

Speed Drop Pattern

Clicking on the "Edit Curve" field opens the table editor for defining the speed drop characteristic after misfiring.

The x-axis range does not have any meaning. The duration of the speed drop in °CA is defined using the "Segment Length" parameter (see Fig. 17-27 on page 501).

The z-axis defines the engine speed drop. z-values between 0.0 and 2.0 are allowed. A value of 0.5, for example, means that the engine speed is halved.

In general, engine speed during misfire simulation is determined by multiplying the engine speed prior to misfire with the factor for the engine speed drop (z-value).

The changes in engine speed caused by consecutive misfirings can overlap. For example, overlapping can occur if for a 4-cylinder engine the "Segment Length" > 180 °CA.

The "Speed Drop Pattern" table can be edited in runtime mode.

Worst Case Speed Drop/Best Case Speed Drop

This option is used to determine the speed drop characteristic after misfiring. The resulting characteristic is calculated using "Best Case Speed Drop" and "MfMode_Cyl*n*" = 3.

```
DropPatternValue = RandomValue * WorstCasePattern + (1
- RandomValue) * BestCasePattern
```

Misfire Sequence 0 ... Misfire Sequence 19

Clicking on the "Edit Curve" field opens the table editor to specify the pattern for the occurrence of misfiring. The patterns are each 100 ignitions long. The x-axis has no meaning and is ignored.

A z-value of 1 means that misfiring is to be simulated for the current ignition, while a z-value of 0 means that no misfiring is to be simulated for the current ignition.

The "Misfire Sequence *nn*" tables can be edited in runtime mode.

Note

The indices 0 to 19 appended to the "Misfire Sequence" RTIO parameters indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 17-3 summarizes the properties of the global parameters of the ES4320-Misfire device.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Segment Start Angle	real32	Yes	Specifies the angle between the top dead center (reference trigger) and the start of the drop in engine speed due to misfiring. Valid range: -720.0 °CA ... 720.0 °CA
Segment Length	real32	Yes	Specifies how long (in °CA) the drop in engine speed after misfiring lasts. Valid range: 0 °CA ... 720 °CA
Speed Drop Pattern	1-dim. Table	Yes	Engine speed drop characteristic after misfiring. z-value range: 0.0 ... 2.0 The x-values have no meaning.
Worst Case Speed Drop / Best Case Speed Drop	1-dim. Table	Yes	Speed drop characteristic after misfiring z-value range: 0.0 ... 2.0 The x-values have no meaning.
Misfire Sequence 0 ... Misfire Sequence 11	1-dim. Table	Yes	Misfiring patterns. The patterns are each 100 ignitions long. z-value = 1: misfiring is to be simulated for the current ignition. z-value = 0: misfiring is not to be simulated for the current ignition. The x-values have no meaning.

Tab. 17-3 Global RTIO Parameters of the ES4320-Misfire Device

17.3.35 Groups (ES4320-Misfire Device)

This section describes the signal-group-specific options of the ES4320-Misfire device.

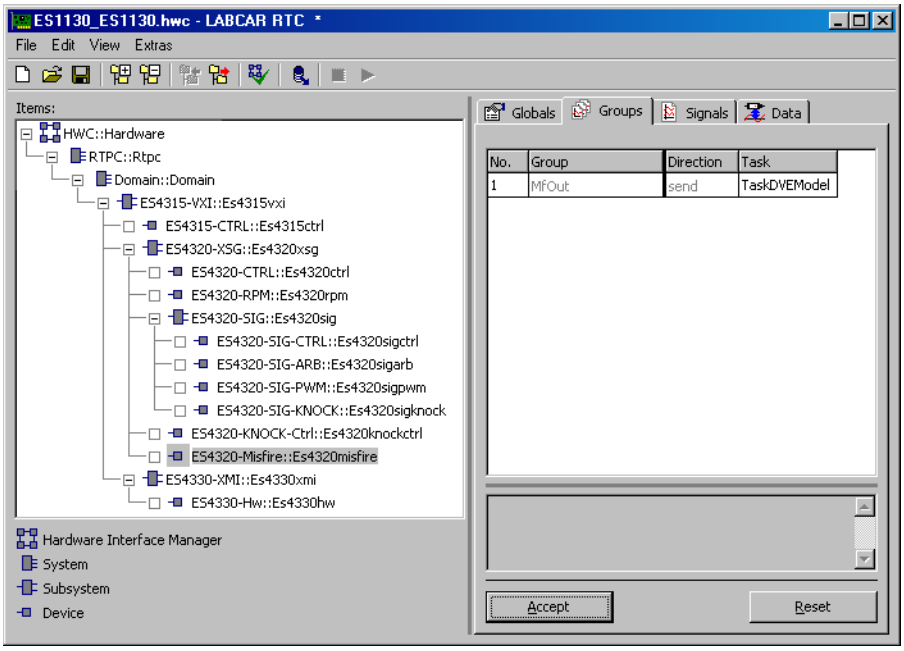


Fig. 17-29 The "Groups" Tab of the ES4320-Misfire Device

17.3.36 Signals (ES4320-Misfire Device)

The "Signals" tab lists the individual signals of the ES4320-Misfire device.

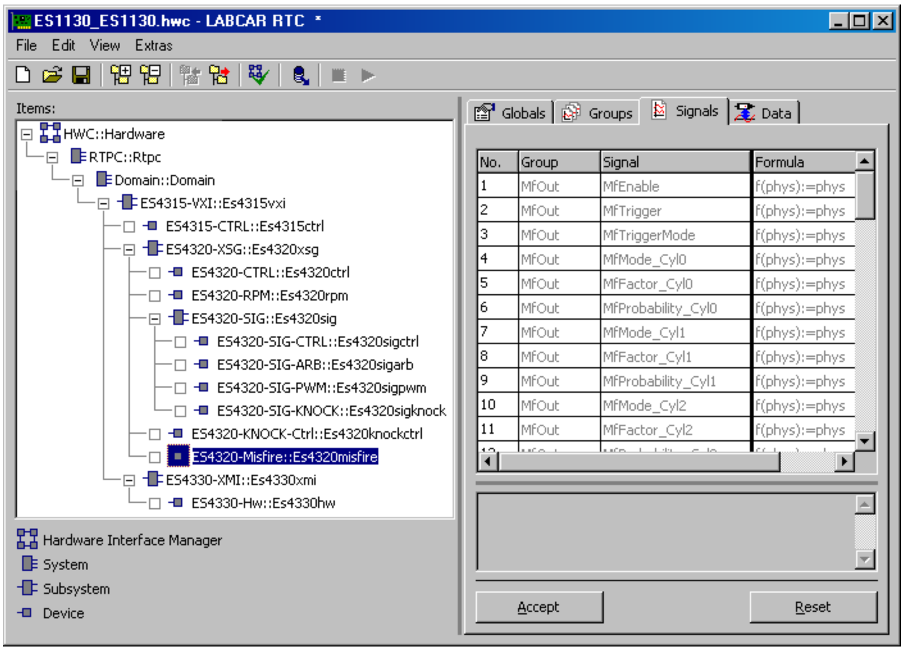


Fig. 17-30 The "Signals" Tab of the ES4320-Misfire Device

MfEnable

The "MfEnable" RTIO signal is used to switch the misfire generator on and off. A signal value of 0 switches the misfire generator off; a signal value of 1 switches the misfire generator on.

MfTrigger

A 0 → 1 transition of the "MfTrigger" signal triggers misfire simulation during the time period covering the following 100 engine ignitions (per cylinder).

The "MfTrigger" signal is only active when the misfire generator is operated in "Single Shot" trigger mode. The trigger mode is set using the "MfTriggerMode" signal.

MfTriggerMode

The "MfTriggerMode" RTIO signal is used to control the misfire generator trigger mode.

If the signal is set to 0, the misfire generator operates in "Single Shot" trigger mode.

A 0 → 1 transition of the 'MfTrigger' signal triggers misfire simulation during a time period covering the following 100 engine ignitions (per cylinder). After that, misfire simulation is suspended until the next trigger occurs.

If the signal is set to 1, the misfire generator operates in "Continuous" mode. In "Continuous" mode misfire simulation is permanently active. The "MfTrigger" signal is inactive in this trigger mode.

MfMode_Cyl0 ... MfMode_Cyl19

The "MfMode_Cyl0" to "MfMode_Cyl19" RTIO signals specify the misfire generator operating mode for the associated cylinder.

- Disabled Mode (MfMode_Cyl*n* = 0)

  Misfire generation is disabled for the associated cylinder.

- Pattern Mode (MfMode_Cyl*n* = 1)

  The associated "Misfire Sequence" table defines the misfiring pattern for the associated cylinder.

- Random Mode (MfMode_Cyl*n* = 2)

  A random generator controls whether misfiring is to be generated during the current ignition or not. The probability of whether misfiring occurs or not is set using the associated "MfProbability" signal.

- Worst Case Speed Drop/Best Case Speed Drop

  Misfired generation is controlled by the respective tables (see "Worst Case Speed Drop/Best Case Speed Drop" on page 503).

MfFactor_Cyl0 ... MfFactor_Cyl19

The "MfFactor_Cyl0" to "MfFactor_Cyl19" RTIO signals specify a correction factor for the engine speed drop after misfiring for the associated cylinder.

A factor of 0.7 means that only 70 % of the value defined for a drop in engine speed in the "Speed Drop Pattern" table is effective.

Valid value range: 0.0 ... 1.0.

MfProbability_Cyl0 ... MfProbabilit_Cyl19

The "MfProbability_Cyl0" to "MfProbability_Cyl19" RTIO signals set the probability that misfiring occurs during the current ignition.

These RTIO signals are only active if the associated "MfMode_Cyl*n*" signal is set to "Random Mode".

The valid value range is 0.0 to 1.0. A value of 0.0 means that misfiring never occurs; a value of 1.0 means that misfiring occurs during every ignition of the associated cylinder.

Note

The indices 0 to 19 appended to the "MfMode_Cyln", "MfFactor_Cyln" and "MfProbability_Cyln" RTIO signals indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 17-4 summarizes the properties of the RTIO signals of the ES4320-Misfire device.

RTIO Signal	Data Type	Comment
MfEnable	uint8	Enable / disable misfire generator 0: disable 1: enable
MfTrigger	uint8	Misfire generator trigger signal. A 0 → 1 transition triggers misfire simulation during the time period covering the following 100 engine ignitions (per cylinder). Only active in trigger mode "Single Shot"
MfTriggerMode	uint8	Misfire generator trigger mode 0: "Single Shot" trigger mode 1: "Continuous" trigger mode
MfMode_Cyl0 ... MfMode_Cyl19	uint8	Misfire generator operating mode for the associated cylinder 0: Disabled 1: Pattern mode 2: Random mode 3: Best/worst case mode
MfFactor_Cyl0 ... MfFactor_Cyl19	real32	Correction factor for speed drop after misfiring for the associated cylinder. Valid values: 0.0 ... 1.0
MfProbability_Cyl0... MfProbability_Cyl19	real32	Probability that misfiring occurs during the current ignition. Only active in "Random" misfire mode. Valid range: 0.0 (no misfiring) ... 1.0 (always misfiring)

Tab. 17-4 RTIO Signals of the ES4320-Misfire Device

17.4 ES4330-XMI - VXI Signal Measurement Board

Definitions

Active signal edge:

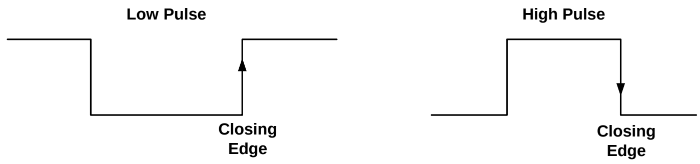
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.



Inactive signal:

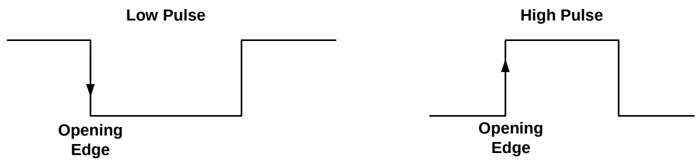
A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

LWL:

Stands for **L**ower **W**indow **L**imit and refers to the lower limit of the angle window in speed-synchronous measurements.

Opening edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



UWL:

Stands for **U**pper **W**indow **L**imit and refers to the upper limit of the angle window in speed-synchronous measurements.

17.4.1 Functional Description

Basic Principle of Measurement Value Determination

To ensure the greatest possible flexibility in the evaluation of pending input signals, the ES4330 board is based on the following basic principle:

First of all, each input signal is conditioned separately (analogously), then compared with a threshold value (which can be set individually for each signal using the software) to convert the analog input signal to digital 0/1 or inactive/active information. The binary level information of the signals obtained in this way at the 64 input channels is continuously checked for changes by a hardware circuit, i.e. "edges" are recognized on one or more input bits. If at least one input signal changes from active to inactive, the circuit saves the current statuses of the 64 inputs and calls the current values of two integrated counters which specify the current time and the current crankshaft angle. This information is transferred by the hardware to a FIFO buffer.

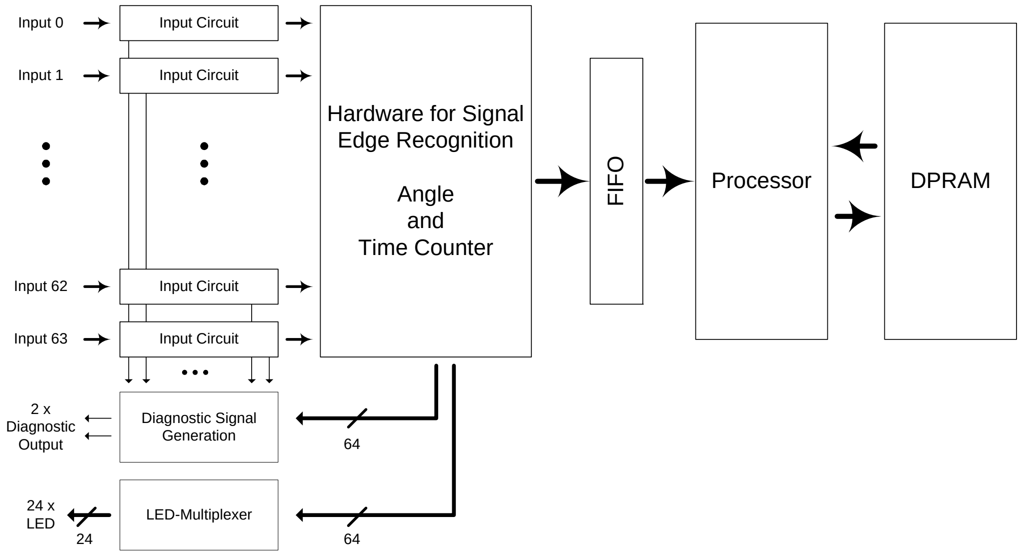


Fig. 17-31 Block Diagram of the ES4330 Board

The firmware downloaded to the processor of the ES4330 board can now read the generated values from the FIFO buffer and thus generate information on when and at what angle which input signals have changed and how. In turn, this information is used to generate all measurement values which are of interest to the user, such as injection times, ignition times and duty cycles.

This basic concept of hardware edge recognition with a post-connected software conditioning of the measurement values means that the usage factor of the processor depends directly on the number of recognized edges (i.e. the frequencies of the input signals). The advantage of this concept is that the hardware simply acquires all the changes and gives them an angle and time stamp. The evaluation, however, is executed by the software and is therefore relatively easy to change/adapt. The ES4330 firmware and RTIO driver offer a wide range of implemented measurement procedures (cf. page 532 ff.). The concept used also makes it possible to implement other measurement procedures by simply adapting the ES4330 firmware and the RTIO driver.

Input Circuit

The input circuit of every hardware channel on the ES4330 board already shown in Fig. 17-31 on page 509 is shown in more detail in Fig. 17-32. It is divided into two units which are explained below:

- realization of a "pull-up" / "pull-down" functionality and
- threshold comparison.

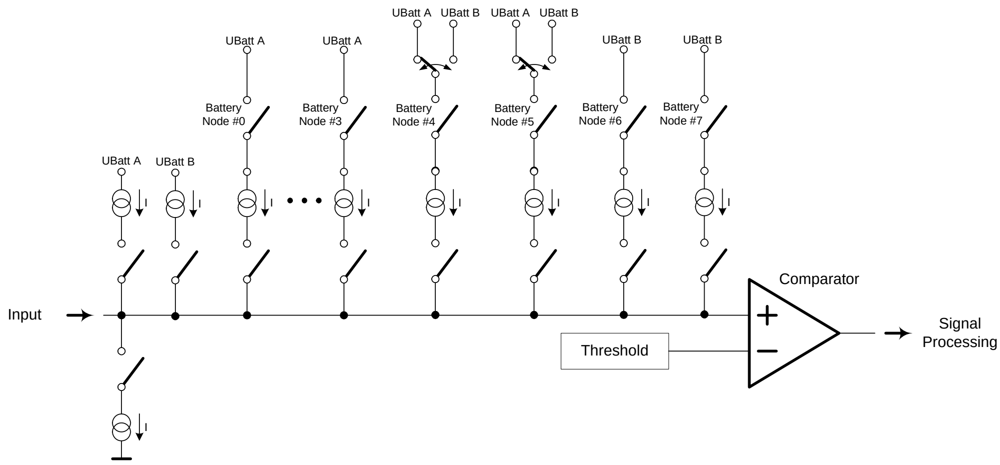


Fig. 17-32 Simplified Representation of the Input Circuit of an ES4330 Hardware Channel

Realization of "Pull-Up" / "Pull-Down" Functionality

A lot of ECU outputs are designed so that only the active levels of the signals are generated by the ECU itself, i.e. the outputs are simply switches against battery voltage or battery ground as shown in the following diagram (points 1 and 2). An example of this kind of circuit is the open collector outputs which in terms of their function form a switch against battery ground (2).

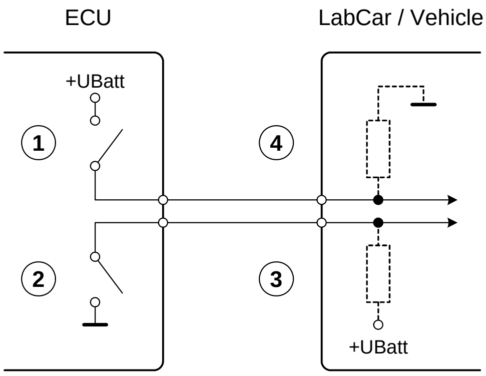


Fig. 17-33 Necessity of Pull-Up / Pull-Down Circuits

In a real vehicle, this kind of signal is used to address actuators which, for example, are connected directly to the battery voltage. In this kind of situation, switching the control signal against battery mass simply results in the circuit being closed.

If an ECU is being operated on LABCAR and if the actuators are replaced by the ES4330 board, you must observe the following two points:

- control signals, as described above, only have a well-defined level when active; when inactive they are undefined (from the point of view of the ECU). To be able to measure this kind of signal with the ES4330 board, the inactive status of the signal has to be specified by the ES4330 board.
- in an active state with a connected actuator, a current flows which is measured in several ECUs to enable error detection (if no current is flowing in spite of the active signal, the actuator must be defective). This flow of current also has to be enabled if an ES4330 is connected instead of an actuator.

The standard procedure for solving these problems would be to use pull-up and/or pull-down resistances, as shown in Fig. 17-33 (points 3 and 4). These would specify the voltage level of the inactive signal, and a current could flow via the resistances with a pending, active signal.

The disadvantage of this construction would be that the size of the flowing current would depend directly on the battery voltage ($I = U / R$); a simulation-controlled modification of the battery voltage could therefore have a direct effect on error detection in the ECU. This is why constant current sources are used on the ES4330 board as shown in Fig. 17-32 on page 510. These fulfil the same functionality as the corresponding pull-up / pull-down resistances but for various battery voltages it is not the resistance that is constant and the current variable but (as the name suggests) the current which is constant and the resulting (apparent) resistance which is variable.

Threshold Comparison

In addition to the realization of the pull-up and pull-down functionality, the circuit of an ES4330 input basically consists of a comparison with a threshold value defined by the user. This comparison leads to the analog input signal being converted to digital (0/1) information. The threshold value is defined in such a way that all voltages pending at the board input which are greater than this value, are interpreted as "1". All voltages under this threshold value are interpreted as "0".

The threshold value can be defined for each channel independently of all other channels. For each channel, it can be specified as being either absolute in the range 0 V to 60 V or relative to battery voltage A or B.

It is often the case in the automotive sector that relatively slow, i.e. low-frequency, signals are used which very often have large rise or fall times at the edges. If this kind of signal is compared to a specified threshold value, there is a danger – particularly if there is noise on the signal line – that the result of the comparison changes backwards and forwards from "1" to "0" a few times with a slow overstepping/undercutting of the threshold. To avoid these unwanted side-effects, the comparison circuits on the ES4330 board have a hysteresis, i.e. the threshold value for the change from "0" to "1" is slightly higher than the threshold value for the change from "1" to "0". The size of the hysteresis can be set via the software.

Measurement Value Acquisition

The ES4330 RTIO driver provides the user with up to 256 freely configurable measurement values (referred to below as software or measurement channels) with each individual measurement value being able to depend on each hardware channel. The assignment of hardware channels on the one hand and measurement channels on the other is thus not specified by the ES4330 RTIO driver, i.e. when using LABCAR for several projects it is not necessary to connect identical signals to the same hardware channels of the ES4330 board.

The 256 measurement channels give the user four times as many output values as the board has hardware channels. This makes it easy to determine several measurement values from just one input signal, such as frequency and duty cycle of a PWM signal. A double wiring of one and the same signal on different channels of the board is thus not necessary.

The following can be defined for each measurement channel (in addition to the base hardware channel) completely independently of all others:

- how the pending signal is to be evaluated
- when and how it should be checked for missing pulses (*timeout monitoring*) and
- whether the evaluation should take place continuously or only within a certain range of the crankshaft angle (*angle segmenting*).
- For some measurements, you can also determine the number pulse to be measured.

Timeout Monitoring

As described above, the ES4330 board is edge-controlled, i.e. the pending external signals are evaluated exclusively at their edges. This, however, means that the generated measurement values returned by the ES4330 RTIO driver are also only updated according to edges. A signal which has already been evaluated once by the board would thus continue to return the last valid measurement value generated, even if in the meantime it has completely dropped out, i.e. has no more edges.

The ES4330 RTIO driver provides timeout monitoring functionality to be able to correctly interpret signals which have completely or partially dropped out. A test can be carried out at certain intervals for every individual measurement value to see whether it has been updated by new edges at the input signal. This test can take place either in intervals specified by the user or at specific crankshaft angles specified by the user.

If this kind of check shows that no edges have been recognized by the hardware since the last check, the relevant measurement value can be modified in two different ways. It is either set to a specified timeout value or determined depending on the current level (high or low) of the input signal. This is particularly necessary for the correct evaluation of duty cycles (an inactive input results in the measurement values 0.0 or 1.0 – depending on the level of the input signal).

This makes a total of six different timeout monitoring possibilities:

- checking in time intervals of x ms: measurement value in a timeout predefined as value y.
- checking every 720° at x° crankshaft: measurement value in a timeout predefined as value y.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria occur: measurement value in a timeout predefined as value y.
- checking in time intervals of x ms: measurement value in a timeout depending on the level of the input signal.

- checking every 720° at x° crankshaft: measurement value in a timeout depending on the level of the input signal.
- checking every 720° at x_1° crankshaft and every x_2 ms. A timeout is determined if one or both timeout criteria occur: measurement value in a timeout depending on the level of the input signal.

17.4.2 The Structure of the ES4330 RTIO Tree

With the ES4300 Signal Box, ETAS developed a combined VMEbus/VXIbus system. The two bus systems are connected to each other via an ES4315 board which implements the protocols of the two buses. The ES4315 board is in Slot 0 of the VXIbus and in Slot 2 of the VMEbus. A VMEbus master (usually an ES1130 board) installed in the ES4300 also executes the tasks of a VXIbus system controller. This functionality is provided by the RTIO driver of the ES4315 board. The ES4330 board can be assembled in slots 1 to 9 of the VXIbus system.

Note

VXIbus slots are numbered consecutively starting with 0; VMEbus slots are numbered consecutively starting with 1.

The described hardware link of the VXIbus to the VMEbus is reflected in the RTIO tree (Fig. 17-34). The inclusion of the VXIbus part of an ES4300 Signal Box in an RTIO hardware description starts with the addition of an ES4315-VXI subsystem to an experimental target. The VXIbus boards in the VXIbus part of the ES4300 system must then be added to the ES4315-VXI subsystem.

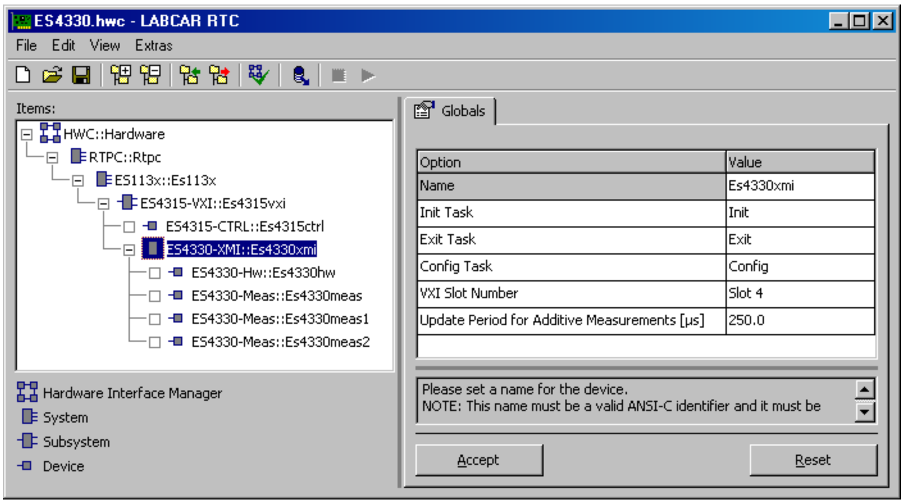


Fig. 17-34 RTIO Hardware Description with ES4315 VMEbus/VXIbus Adapter and ES4330 Board

The ES4330 board has 64 hardware channels for acquiring and conditioning analog and digital signals. The 64 hardware channels are configured using the ES4330-Hw device which is assigned to the ES4330-XMI subsystem. In addition to this item, up to 8 ES4330-Meas devices can be assigned to the ES4330-XMI subsystem. Up to 32 measurements can be specified with each of these items. This means that, in total, the RTIO driver and firmware of the ES4330 board provide 256 measurements or measurement channels.

Note

Before an ES4330-Meas device can be added, an ES4330-Hw device must first be added to configure the ES4330 hardware.

17.4.3 Global Settings - ES4330-XMI Subsystem

The ES4330-XMI Subsystem is used to set RTIO parameters which have a global effect, i.e. which have an effect on all ES4330-Meas devices.

17.4.4 Globals (ES4330-XMI Subsystem)

Fig. 17-35 shows the RTIO parameters of the "Globals" tab. Tab. 17-5 on page 515 lists the properties of the individual parameters.

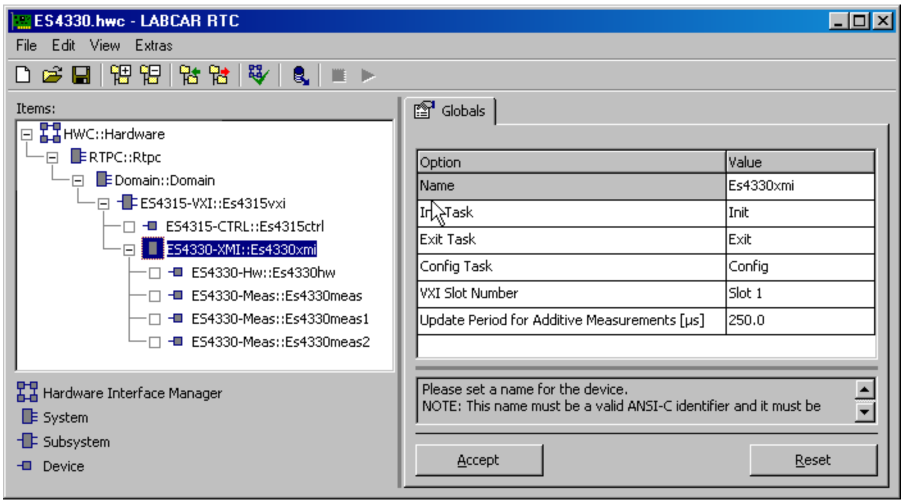


Fig. 17-35 The "Globals" Tab of the ES4330-XMI Subsystem

VXI Slot Number

Number of the VXIbus slot holding the ES4330 board. The visible VXIbus slots within the ES4300 Signal Box are numbered from left to right starting with 1. The ES4315 VXIbus / VMEbus adapter is permanently stored in the VXIbus slot 0 which is not visible to the user. The VXIbus slot number cannot be changed during runtime.

Update Period for Additive Measurements [µs]

This parameter is only of any importance to software and measurement channels which execute additive time measurements in asynchronous mode. It specifies the maximum intervals at which the additive time has to be updated. For more information, refer to "Functional Description: Asynchronous Additive Pulse-Width Measurements" on page 535. The parameter can be configured during runtime of the model.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
VXI Slot Number	uint32	No	VXIbus slot number You can choose from "Slot 1" to "Slot 9".
Update Period for Additive Measurements [µs]	real32	Yes	Measurement update rate for asynchronous additive measure-ments in microseconds. Minimum: 50 µs Maximum: 500 ms

* Data type which the RTIO driver uses internally for the parameter

Tab. 17-5 ES4330-XMI Subsystem: Configuration Parameters of the "Globals" Tab

17.4.5 Hardware Configuration - ES4330-Hw Device

The ES4330-Hw device is used to configure and control the ES4330 hardware. Settings which are independent of the individual hardware channels are executed in the "Globals" tab of this item. Channel-specific settings are made in the "Signals" tab.

17.4.6 Globals (ES4330-Hw Device)

Fig. 17-36 shows the RTIO parameters of the "Globals" tab. Tab. 17-6 on page 517 lists the properties of the individual parameters. The parameter can be edited online (i.e. during runtime of the model on the experimental target).

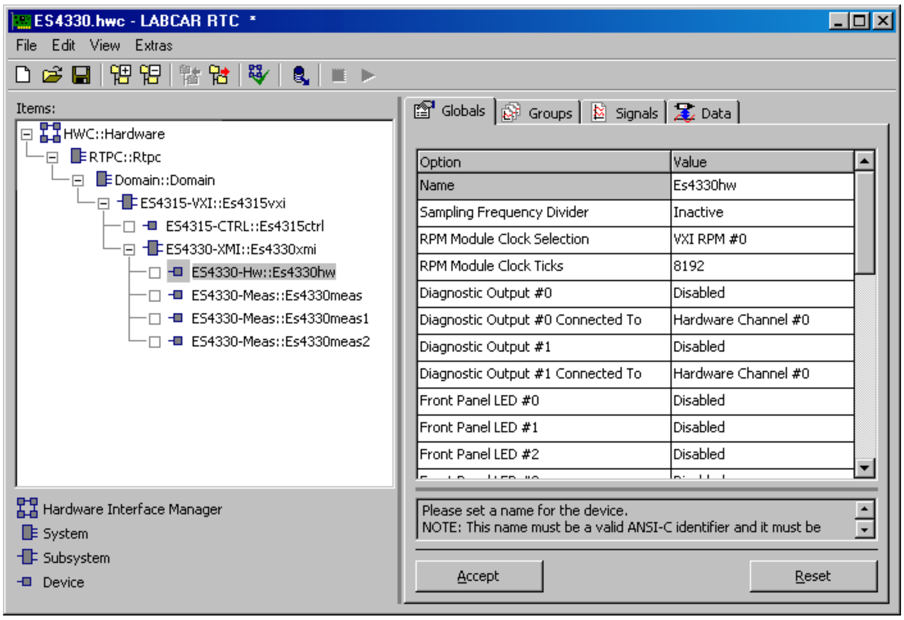


Fig. 17-36 The "Globals" Tab of the ES4330-Hw Device

Sampling Frequency Divider

The ES4330 samples the input signals at the hardware channels periodically. The maximum sampling rate is determined by the frequency of the internal clock, which is 4 MHz. The sampling frequency can be reduced with a frequency divider. The frequency divider is configured in the "Sampling Frequency Divider" option field of the "Globals" tab. Frequency division can be disabled or configured to dividers 2, 4, 8, 16 or 32. Please note that only input pulses which last longer than the sampling period can be acquired accurately.

RPM Module Clock Ticks - RPM Module Clock Selection

To execute speed-synchronous measurements, the ES4330 board requires information on the angle position and direction of rotation of the engine. This information is available to the ES4330 boards at the VXIbus backplane. On the VXIbus backplane, the ES4300 system provides two angle-clock lines each with three signals; a synchronization signal for the zero crossing of the crankshaft or camshaft angle, a signal in which the direction of rotation of the engine is coded and the actual clock signal in which the angle position of the engine is coded. The "RPM Module Clock Selection" option field specifies which of the two angle-clock lines ("VXI RPM #0" or "VXI RPM #1" option) provides the ES4330 with the angle clock. If "None" is set, the ES4330 board is not provided with an angle clock.

The numeric input field "RPM Module Clock Ticks" is used to define the angle resolution, i.e. the number of angle clocks per camshaft rotation (720° crankshaft). If, for example, 8192 is set, the ES4330 measurement acquisition works with an angle resolution of 720°CA / 8192 = 0.088°CA (°CA = degrees crankshaft).

Diagnostic Output #0, Diagnostic Output #0 Connected To, Diagnostic Output #1, Diagnostic Output #1 Connected To

The ES4330 board has two connectors on the front panel for test and diagnostic purposes. Two independent signals can, e.g., be connected directly to an oscilloscope via these connectors for monitoring and evaluation purposes. The input signals of all 64 hardware channels are available as test signals. The hardware channel is selected in the "Diagnostic Output #x Connected To" option field (x = 0, 1).

In addition to selecting the actual signal, the user can also determine from which point of the signal flow the signal is to be issued to the test output. It can be tapped as an analog signal immediately before the comparator of the hardware channel, whereby for technical reasons the test signal then has an output amplitude which is 1/6 of the amplitude of the input signal. The user can also tap the test signal immediately after the comparator of the hardware channel, i.e. as a TTL signal. The fact that you can tap the signal both before and after the comparator makes direct control of the set threshold value very easy. The signal type is selected in the "Diagnostic Output #x" option field (x = 0, 1); this option field also makes it possible to disable the test output.

Front Panel LED #0 ... Front Panel LED #23

There are 24 LEDs on the front panel of the ES4330 board. The signal level of a hardware channel can be displayed with each LED. If the signal has a low level, the LED is dark; if it has a high level, the LED is light. The definition of low and high level comes from the threshold voltage of the input comparator of the relevant hardware channel. The "Globals" tab has a "Front Panel LED #x" option field (x = 0, 1, ...23) for every LED with which the assigned hardware channel can be set or the LED deactivated.

Parameter or Option Field	**Data Type***	**Can be Edi-ted Online**	**Comment / Value Range**
Sampling Frequency Divider	uint8	Yes	Frequency divider for input signal sampling 0: no frequency division (4 MHz sampling) 1: division by 2 (2 MHz) 2: division by 4 (1 MHz) 3: division by 8 (500 kHz) 4: division by 16 (250 kHz) 5: division by 32 (125 kHz)
RPM Module Clock Ticks	uint32	Yes	Angle resolution Value range: 1 ... 65536
RPM Module Clock Selection	uint8	Yes	Source for angle clock 0: no angle clock 1: RPM module #0 2: RPM module #1
Diagnostic Out-put #0 Diagnostic Out-put #1	sint8	Yes	Activation and deactivation of the diagnostic outputs as well as selection of the signal type (analog or TTL) -1: deactivation of the diagnostic output 0: TTL signal output 1: analog signal output (amplitude is 1/6 of the amplitude of the input signal)
Diagnostic Out-put #0 Connected To Diagnostic Out-put #1 Connected To	uint8	Yes	Hardware channel which is connected to the diagnostic output. Value range: 0 ... 63
Front Panel LED #0 ... Front Panel LED #23	sint8	Yes	Hardware channel which is connected to the front panel LED or deactivation of the LED. -1: deactivation 0 ... 63: hardware channel

* Data type which the RTIO driver uses internally for the parameter

Tab. 17-6 ES4330-Hw Device: Configuration Parameters of the "Globals" Tab

17.4.7 Groups (ES4330-Hw Device)

The ES4330-Hw device has two signal groups (Fig. 17-37). The "Control" signal group is transferred from the experimental target to the ES4330 board. This signal group activates and deactivates edge recognition on the individual hardware channels of the board. The "Level" signal group is transferred in the other direction from the ES4330 board to the experimental target. This signal group contains the current level information of all activated hardware channels.

ERCOS^{EK} operating system tasks must be assigned to both signal groups. Normally, a task with "Type" "Alarm" and a relatively large activation period (e.g. 100 ms) is selected for the "Control" signal group as the activation/deactivation of hardware channels is not normally a highly dynamic procedure. If activation/deactivation of the hardware channels is only due to take place at model start/stop, it is sufficient to assign the "Control" signal group to the "Init" and "Exit" task of the model.

If the level information is evaluated in the simulation model, a task with "Type" "Alarm" is assigned to the "Level" signal group. The activation period depends on the dynamic behavior or period duration of the signals to be acquired.

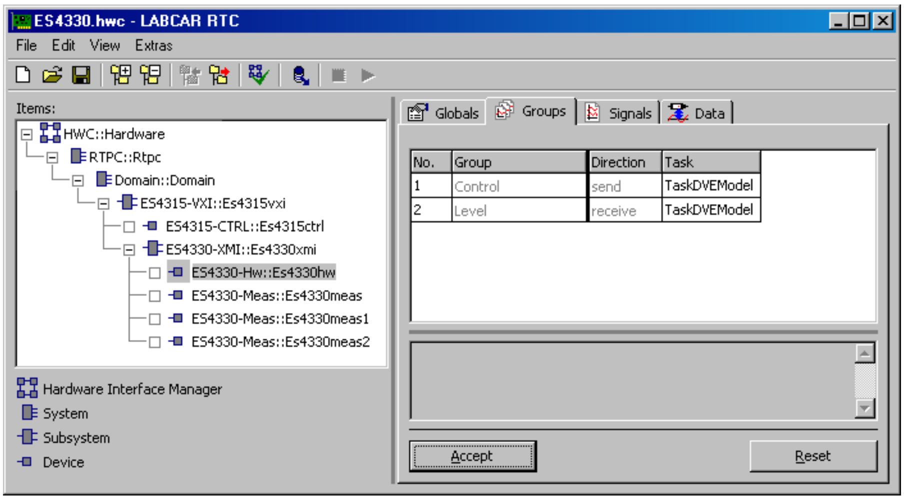


Fig. 17-37 The "Groups" Tab of the ES4330-Hw Device

17.4.8 RTIO Signals of the "Control" Signal Group

The "Control" signal group consists of 64 RTIO signals with which edge recognition on the 64 hardware channels can be activated or deactivated. Tab. 17-7 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
ChnEnable_0 ... ChnEnable_63	bool	Activation/deactivation of hardware channels 0 to 63 0: deactivation of edge recognition 1: activation of edge recognition
* Data type which the RTIO driver uses internally for the parameter		

Tab. 17-7 ES4330-Hw Device: The RTIO Signals of the "Control" Signal Group

Note

The "Control" signal group is only written to the DPRAM of the ES4330 board when the value of one of its RTIO signals has changed.

17.4.9 RTIO Signals of the "Level" Signal Group

The "Level" signal group consists of 2 RTIO signals which are to be interpreted as bit fields. The bit fields each have 32 bits. The "LvlBitField_0" bit field contains coded level information of hardware channels 0 to 31; the "LvlBitField_1" bit field of hardware channels 32 to 63. Tab. 17-8 lists the properties of the RTIO signals.

Note

Only level bits of hardware channels for which edge recognition is activated are valid in the "LvlBitField_0" and "LvlBitField_1" bit fields. For more information on activating and deactivating edge recognition on a hardware channel, refer to "RTIO Signals of the "Control" Signal Group" on page 519.

RTIO Signal	Data Type*	Comment / Value Range
LvlBitField_0	uint32	Bit field with level information of hardware channels 0 to 31. Channel 0: LSB (Least Significant Bit) Channel 31: MSB (Most Significant Bit) Bit value 0: low level Bit value 1: high level
LvlBitField_1	uint32	Bit field with level information of hardware channels 32 to 63. Channel 32: LSB Channel 63: MSB Bit value 0: low level Bit value 1: high level

* Data type which the RTIO driver uses internally for the parameter

Tab. 17-8 ES4330-Hw Device: The RTIO Signals of the "Level" Signal Group

17.4.10 Signals (ES4330-Hw Device)

The "Signals" tab is used to configure the 64 hardware channels of an ES4330 board. Fig. 17-38 shows the RTIO parameters of the "Signals" tab. Tab. 17-9 on page 523 lists the properties of the individual parameters. The parameter can be edited online (i.e. during runtime of the model on the experimental target).

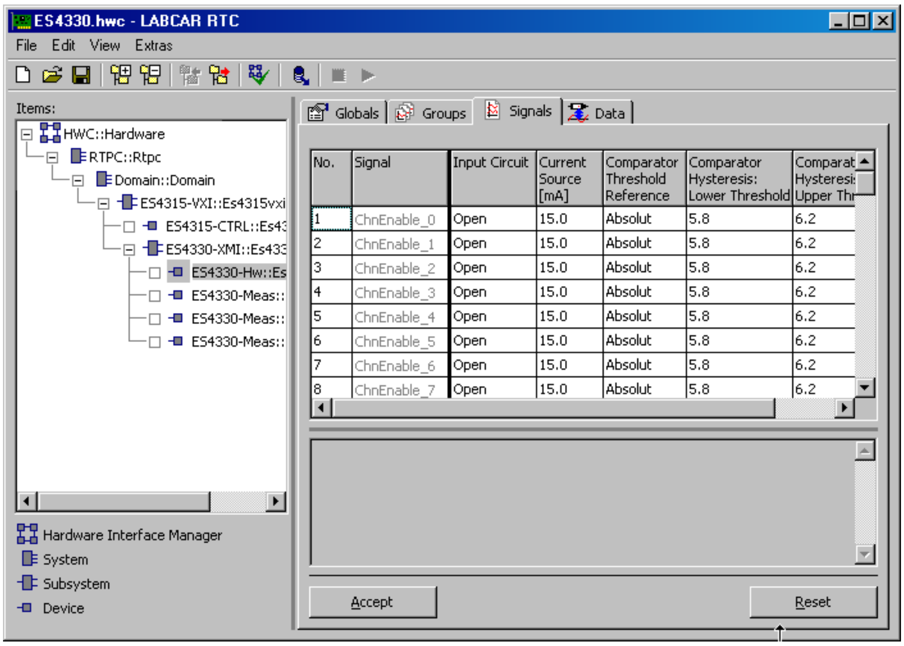


Fig. 17-38 The "Signals" Tab of the ES4330-Hw Device

Input Circuit

This option field is used to specify the input circuit of a hardware channel. The available options can be derived from the circuit diagram of a hardware channel (see Fig. 17-32 on page 510). In addition to the direct connection of the input signal to the input comparator ("Open" option), you can also connect a pull-up circuit to the battery voltages A or B or a pull-down circuit to battery ground. A pull-up circuit to one of the eight battery nodes of the ES4300 system is also possible. The connection of battery nodes 4 and 5 to battery voltage A or B shown in Fig. 17-32 on page 510 and the enabling/disabling of the individual battery nodes is not executed by the RTIO driver of the ES4330 board. These ES4300 system properties are configured and monitored by the RTIO driver of the ES4315 board.

Current Source [mA]

The current intensity of the constant current sources of a hardware channel is defined in the numeric input field "Current Source [mA]". The maximum admissible current intensity depends on the input circuit and the size of the battery voltage (see Tab. 17-9 on page 523).

Comparator Threshold Reference

The "Comparator Threshold Reference" option field is used to specify whether the threshold voltage of an input comparator is defined by specifying a voltage in Volts ("Absolute" option) or in relation to the battery voltages A or B.

Comparator Hysteresis: Lower Threshold - Comparator Hysteresis: Upper Threshold

The threshold voltage of the input comparators of a hardware channel has hysteresis behavior. The numeric input fields "Comparator Hysteresis: Lower Threshold" and "Comparator Hysteresis: Upper Threshold" are used to specify the comparator hysteresis and thus the threshold voltage.

If the "Absolute" option is selected in the "Comparator Threshold Reference" option field of the hardware channel, the numeric values in the "Comparator Hysteresis" fields are interpreted as voltages in Volts. The admissible value range is 0 V to 60 V.

If the option "Relative to UBatt_A" or "Relative to UBatt_B" is set in the "Comparator Threshold Reference" option field of the hardware channel, the comparator hystereses are obtained by multiplying the numeric values in the "Comparator Hysteresis" fields with the relevant battery voltage. The entries in the "Comparator Hysteresis" fields must be between 0 and 1 in this case.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Input Circuit	uint8	Yes	Specification of the input circuit 0: "Open". The input signal is connected directly to the input comparator. 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	Yes	Current intensity of the constant current source in mA. Minimum: 0.0 mA The maximum depends on the input circuit and the size of the battery voltage: 15 mA, if the input is connected to battery ground via a pull-down circuit. 15 mA with a pull-up circuit to a battery voltage > 30 V. 30 mA with a pull-up circuit to a battery voltage between 0 V and 30 V.
Comparator Threshold Reference	uint8	Yes	Comparator reference 1: "Relative to UBatt_A" 2: "Relative to UBatt_B" 3: "Absolute"
Comparator Hysteresis: Lower Threshold	real32	Yes	Lower comparator threshold Value range: 0.0 ... 1.0 with relative comparator reference 0.0 V ... 60 V with absolute comparator reference
Comparator Hysteresis: Upper Threshold	real32	Yes	Upper comparator threshold Value range: 0.0 ... 1.0 with relative comparator reference 0.0 V ... 60 V with absolute comparator reference
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-9 ES4330-Hw Device: Parameters of the "Signals" Tab

17.4.11 Measuring Configuration - ES4330-Meas Device

The ES4330-Meas device is used to specify and configure measurements. Each ES4330-Meas device has 32 software channels (referred to below as measurement channels), which can be connected freely to the 64 hardware channels of the ES4330 board. Up to 8 ES4330-Meas devices are supported per ES4330 board which means a total of 256 measurement channels can be configured.

17.4.12 Globals (ES4330-Meas Device)

No settings are to be made in the "Globals" tab (Fig. 17-39). In the "Measurement Device ID" option field, a number between 0 and 7 is assigned to the ES4330-Meas device by the system for identification purposes.

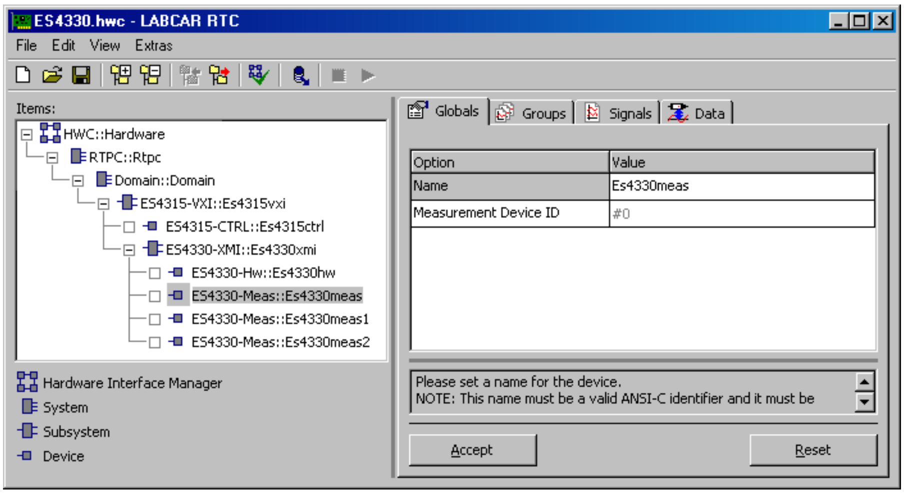


Fig. 17-39 The "Globals" Tab of the ES4330-Meas device

17.4.13 Groups (ES4330-Meas Device)

The ES4330-Meas device has a signal group (Fig. 17-40) which is transferred to the experimental target by the ES4330 board. It contains all measurement data such as measurement values, trigger and timeout information. An ERCOS^{EK} operating system task with "Type" "Alarm" must be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

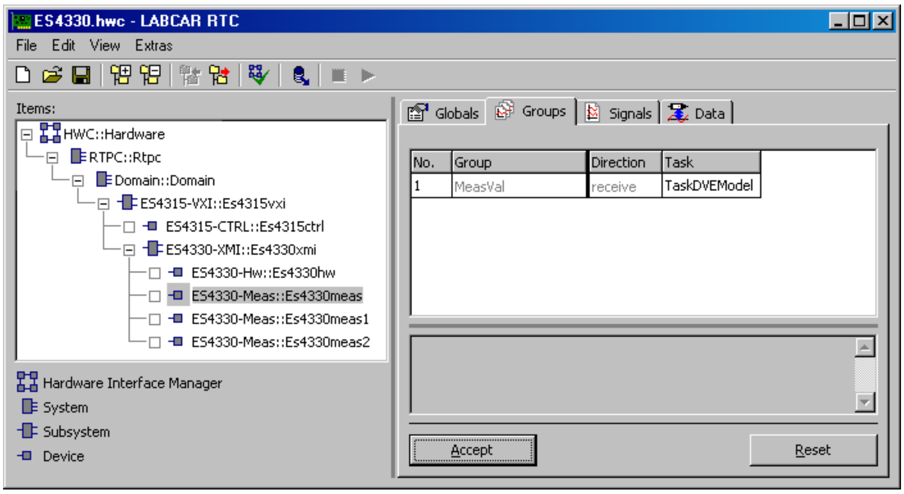


Fig. 17-40 The "Groups" Tab of the ES4330-Meas Device

17.4.14 RTIO Signals of the "MeasVal" Signal Group

The "MeasVal" group consists of 65 RTIO signals. The "TriggerBitField_0" signal must be interpreted as a bit field. It has 32 bits. The trigger or update data of the 32 measurement channels is coded in this bit field, i.e. it shows which measurement values have been updated since the last activation of the read process for the "MeasVal" signal group. If a bit is set, the measurement value of the relevant measurement channel has been newly determined. It is of no importance whether the measurement value was updated due to a timeout or a regular measurement value calculation; the update bit of the measurement channel is set in both cases.

The RTIO signals "MeasVal_0" to "MeasVal_31" contain the measurement values of the 32 measurement channels. These are either measurement values determined in the normal way or, in the case of a timeout, the timeout value intended for this case. If a measurement channel is not used, -8888.0 is assigned to the relevant measurement value. The physical unit of the measurement values depends on the measurement procedure:

- time measurements (time stamp, (additive) pulse duration measurements, period duration) are issued in microseconds
- frequency measurements are in hertz
- angle measurements and angle stamp are specified in degrees crankshaft (°CA)
- all other measurements (duty cycles, pulse count, stepper motor measurements, level measurements) are dimensionless

The RTIO signals "Tout_0" to "Tout_31" contain the results of the timeout monitoring for the relevant measurement channel.

Tab. 17-10 lists the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
TriggerBitField_0	uint32	Bit field with trigger information of the 32 measurement channels. Measurement channel 0: LSB Measurement channel 31: MSB Bit value 0: measurement value is unchanged Bit value 1: measurement value has been updated
MeasVal_0 ... MeasVal_31	real64	Measurement value If the measurement channel is not used, -8888.0 is issued as a value. Physical unit of the measurement value: Time measurements are in microseconds Frequency measurements are in hertz Angle measurements are in °CA
Tout_0 ... Tout_31	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive
* Data type which the RTIO driver uses internally for the parameter		

Tab. 17-10 ES4330-Meas Device: The RTIO Signals of the "MeasVal" Signal Group

17.4.15 Signals (ES4330-Meas Device)

The "Signals" tab is used to configure the 32 measurement channels of an ES4330-Meas device. Fig. 17-41 shows the RTIO parameters of the "Signals" tab. Tab. 17-11 on page 531 lists the properties of the individual parameters. The parameter can be edited online (i.e. during runtime of the model on the experimental target).

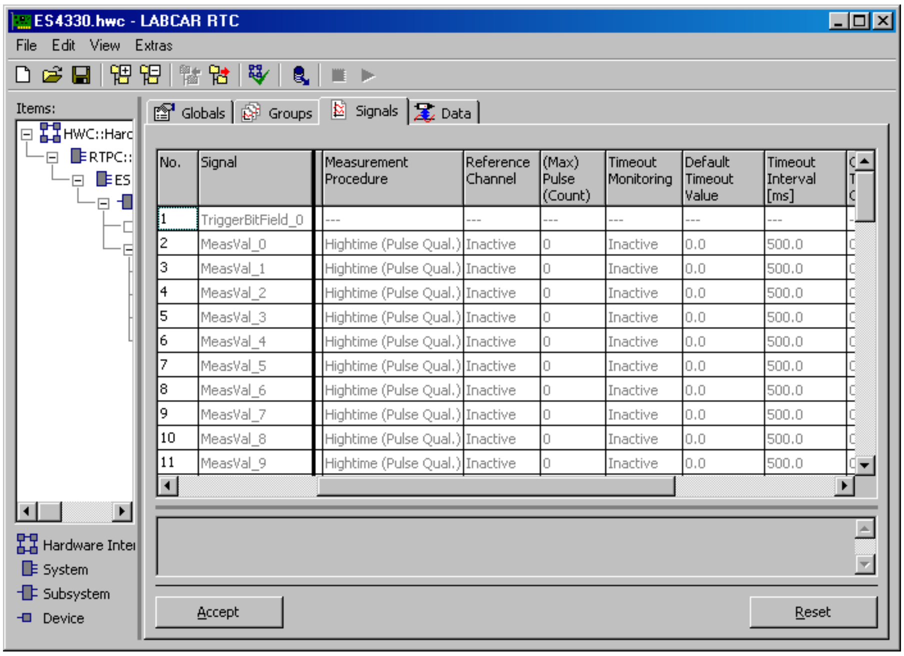


Fig. 17-41 The "Signals" Tab of the ES4330-Meas Device

Hardware Channel

This is where the hardware channel where measurement is due to take place is specified. If "Inactive" is selected, measuring does not take place, i.e. the measurement channel is deactivated.

Note

To avoid unnecessary computing time, measurement channels which are not required should be deactivated.

Measurement Mode

The "Measurement Mode" option field defines the mode in which measuring takes place. There are two options:

- "ESpeedSync": execution of speed-synchronous or angle-synchronous measurements. If speed-synchronous measurements are executed the ES4330 board must be supplied with an angle clock (angle position and direction of rotation of the engine) via the backplane of the ES4300 Signal Box. For more details on configuring the angle clock source, refer to "RPM Module Clock Ticks - RPM Module Clock Selection" on page 516.
- "Asynchronous": execution of asynchronous measurements.

Measurement Procedure

This is where the measurement procedure is selected. Sections 17.4.16 - 17.4.26 contain detailed descriptions and configuration guidelines for the individual procedures. The notation --/-- denotes evaluation at rising edges whereas --\-- denotes evaluation at falling edges.

Reference Channel

Some measurement procedures require a second hardware channel or measurement channel to be executed. The specification of this reference channel for measuring is made in "Reference Channel".

For the pulse-width measurements with an Enable or Validate signal described in "Measurement Procedure - Pulse-Width Measurements with Enable or Validate Signal" on page 537, the reference channel is a hardware channel, the channel at which the Enable or Validate signal is pending.

For the relative measurements between hardware channels described in "Measurement Procedure - Relative Measurements between Hardware Channels" on page 544 and for the measurements for position tracing with two-phase stepper motors described in "Measurement Procedure - Position Tracing with Two-Phase Stepper Motors" on page 555, the reference channel is a measurement channel.

The value in the "Reference Channel" list box is of no importance for all other measurement procedures.

(Max) Pulse (Count)

The meaning of this option field depends on the relevant measurement procedure. With pulse- and edge-selective measurements, the number of the pulse or edge to be measured is specified in this option field. With additive measurements and pulse counts, the maximum number of pulses to be taken into consideration is specified in this option field.

Timeout Monitoring

Definition of the timeout monitoring for the relevant measurement channel (see "Timeout Monitoring" on page 512). The following settings are possible:

- "Inactive": no timeout monitoring.
- "Intvl Predef": timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "Intvl InpDep": timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. The measurement value in a timeout depends on the level of the input signal.
- "CS Angle Predef": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "CS Angle InpDep": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field. The measurement value in a timeout depends on the level of the input signal.

- "Intvl & CS Angle Predef": timeout monitoring every 720° crankshaft rotation at the crankshaft angle specified in the "CS Angle Timeout Check Point" option field and timeout monitoring at the intervals defined in the "Timeout Interval [ms]" option field. A timeout is recognized when one or both timeout criteria are violated. The measurement value in a timeout is the value set in the "Default Timeout Value" option field.
- "Intvl & CS Angle InpDep": like "Intvl & CS Angle Predef", but the measurement value in a timeout depends on the level of the input signal.

Default Timeout Value

The value set in this numeric input field is issued as the measurement value in a timeout if one of the "Intvl Predef", "CS Angle Predef" or "Intvl & CS Angle Predef" timeout monitoring modes is set.

Timeout Interval [ms]

Time between two timeout checks in milliseconds. Only relevant for time-based timeout monitoring modes ("Intvl Predef", "Intvl InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Timeout Check Point

Crankshaft angle at which timeout checks are executed. Only relevant for angle-based timeout monitoring modes ("CS Angle Predef", "CS Angle InpDep", "Intvl & CS Angle Predef", "Intvl & CS Angle InpDep").

CS Angle Lower Limit

Lower limit of the angle window released for measurement in degrees crankshaft. Only relevant in the speed-synchronous measurement mode "ESpeed-Sync".

CS Angle Upper Limit

Upper limit of the angle window released for measurement in degrees crankshaft. Only relevant in the speed-synchronous measurement mode "ESpeed-Sync". The upper angle window limit is also used for additive time measurements and pulse counts to transfer the determined measurement value to the DPRAM of the ES4330 board.

CS Angle Reference

Angle reference point in degrees crankshaft to which all angle measurements refer. This parameter is of no importance in all other measurements.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Hardware Channel	sint8	Yes	Hardware channel on which measuring takes place, and deactivation of the measurement channel -1: deactivation of the measurement channel 0 ... 63: channel number
Measurement Mode	uint8	Yes	Measurement mode 0: "Asynchronous": asynchronous measurements 1: "ESpeedSync": speed- or angle-synchronous measurements
Measurement Procedure	uint32	Yes	Selection of measurement procedure. Measurement procedure; param. value "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 or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
			"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	Yes	Reference channel of the measurement -1: no reference channel 0 ... 63: hardware channels 0 to 63 64 ... 95: measurement channels 0 to 31

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
(Max) Pulse (Count)	uint32	Yes	With pulse- and edge-selective measurements: the number of the pulse or edge to be measured. With additive measurements and pulse counts: the maximum number of pulses to be taken into consideration. Value range: ≥ 0
Timeout Monitoring	uint8	Yes	Timeout monitoring. 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	Yes	Measurement value for timeout. Only relevant in "... Predef" modes for timeout monitoring.
Timeout Interval [ms]	real32	Yes	Time between two timeout checks in milliseconds. Value range: ≥ 1.0 ms Only relevant in "Intvl ..." modes for timeout monitoring.
CS Angle Timeout Check Point	real32	Yes	Crankshaft angle at which timeout checks are executed. Value range: -720 °CA to 720 °CA Only relevant in "CS Angle ..." modes for timeout monitoring.
CS Angle Lower Limit	real32	Yes	Lower limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Upper Limit	real32	Yes	Upper limit of the angle window released for measuring in degrees crankshaft. Value range: -720 °CA to 720 °CA Only relevant in the speed-synchronous measurement mode "ESpeedSync".
CS Angle Reference	real32	Yes	Angle reference point in degrees crankshaft to which all angle measurements refer. This parameter is of no importance in all other measurements. Value range: -720 °CA to 720 °CA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-11 ES4330-Meas Device: Configuration Parameters of the "Signals" Tab

17.4.16 Measurement Procedure - Pulse-Width Measurements

This section describes the properties of pulse-width measurements. A distinction is made between "pulse-qualified" and "edge-qualified" measurements. Each of the following measurements is described:

Pulse-Qualified Pulse-Width Measurements

- "Hightime (Pulse Qual.) [μs]"
- "Lowtime (Pulse Qual.) [μs]"
- "H-Time n-th Pulse (Pu Qual.) [μs]"
- "L-Time n-th Pulse (Pu Qual.) [μs]"

Edge-Qualified Pulse-Width Measurements

- "Hightime (Edge Qual.) [μs]"
- "Lowtime (Edge Qual.) [μs]"
- "H-Time n-th Pulse (Eg Qual.) [μs]"
- "L-Time n-th Pulse (Eg Qual.) [μs]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- pulse selection (only with "H-Time n-th Pulse ..." and "L-Time n-th Pulse ..."): "(Max) Pulse (Count)" defines the number of the pulse to be measured.
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode.
- timeout monitoring

Functional Description: Speed-Synchronous Pulse-Width Measurements

In *pulse-qualified measurements*, only those pulses are measured whose opening and closing edges are within one and the same angle window (valid pulse). Fig. 17-42 on page 533 illustrates how pulse-qualified measuring works using a low-active pulse-width measurement (measurement procedure "Lowtime (Pulse Qual.) [μs]").

In *edge-qualified measurements,* only those pulses are measured whose opening or closing edge is within a (any) angle window (valid pulse). Pulses which have both edges outside an angle window are not measured or taken into consideration. Fig. 17-43 on page 533 illustrates how edge-qualified measuring works using a low-active pulse-width measurement ("Lowtime (Edge Qual.) [μs]").

In *non-pulse-selective measurements* ("Hightime (Pulse Qual.) [μs]", "Lowtime (Pulse Qual.) [μs]", "Hightime (Edge Qual.) [μs]", "Lowtime (Edge Qual.) [μs]"), every valid pulse is measured; the measured pulse width is transferred to the DPRAM of the ES4330 board on the closing edge of the pulse.

In *pulse-selective measurements* ("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]"), the n-th valid pulse within an angle window is measured. The pulse number is defined with the "(Max) Pulse (Count)" parameter of the "Signals" tab. The measured pulse width is transferred on the closing edge of the n-th valid pulse.

Note

If 0 is set as pulse number for pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

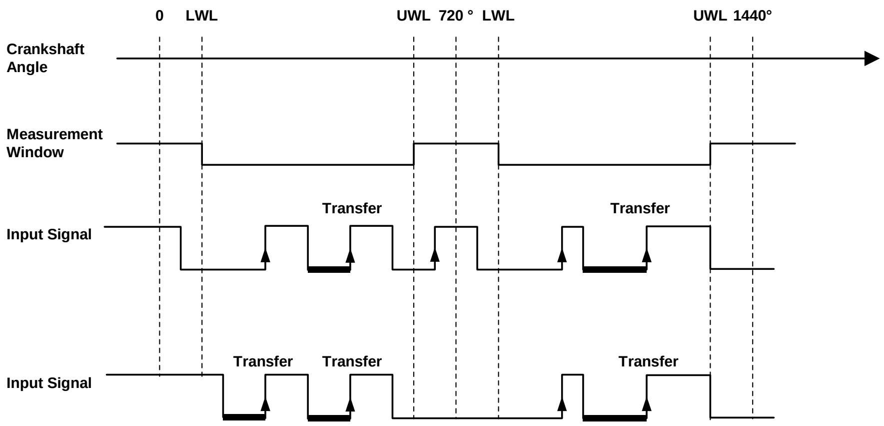


Fig. 17-42 Illustration of how a pulse-qualified low-active pulse-width measurement (measurement procedure "Lowtime (Pulse Qual.) [µs]") works. Measured pulses are shown by a thicker line.

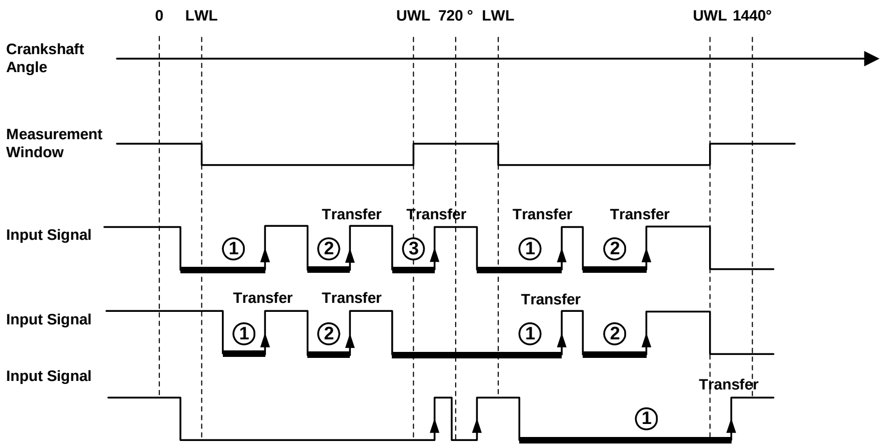


Fig. 17-43 Illustration of how an edge-qualified low-active pulse-width measurement (measurement procedure "Lowtime (Edge Qual.) [µs]") works. Measured pulses are shown by a thicker line. The assignment of the pulses to the angle windows and the counting of the pulses within an angle window is also shown.

Functional Description: Asynchronous Pulse-Width Measurements

With asynchronous pulse-width measurements, it is not possible or not necessary to make a distinction between *pulse-qualified* and *edge-qualified measurements* as there is no window functionality for selecting pulses. The pulse-qualified pulse-width measurements have exactly the same function in asynchronous mode as the edge-qualified measurements.

The *non-pulse-selective measurements* measure the pulse width of each pulse with an active level and transfer the measurement value to the DPRAM on the closing edge of the pulse.

The *pulse-selective pulse-width measurements* are unlikely to have any practical application in asynchronous mode. The pulse width of the n-th pulse with an active level is measured after the ES4330 board is initialized or configured. The measurement value is transferred to the DPRAM on the closing edge of the measured pulse.

Note

If 0 is set as pulse number for pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-12.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. If the signal level is active when the timeout check takes place, the length of time since the last opening edge of the signal is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-12 Pulse-width measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.17 Measurement Procedure - Additive Pulse-Width Measurements

This section describes the properties of additive pulse-width measurements. The functioning of the following measurements is described:

- "Additive Hightime [µs]"
- "Additive Lowtime [µs]"

Admissible measurement options are:

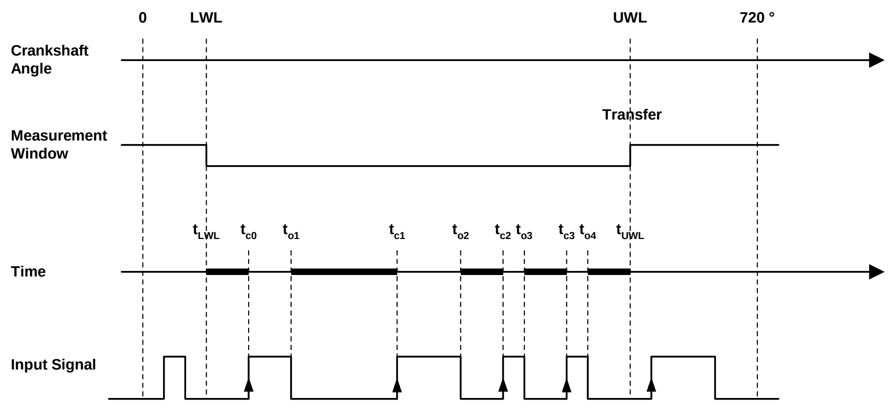
- "Measurement Mode": speed-synchronous or asynchronous measurement mode

- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description: Speed-Synchronous Additive Pulse-Width Measurements

The additive time results from the total of all time segments within an angle window in which the signal is active, regardless of whether the opening or closing edges of the pulses are inside or outside the angle window. Fig. 17-44 shows an example of measurement value calculation with an additive lowtime measurement.

The additive time is transferred to the DPRAM of the ES4330 board on the upper angle window limit.



$$t_{Additive} := (t_{c0} - t_{LWL}) + (t_{UWL} - t_{o4}) + \Sigma\,(t_{ci} - t_{oi}), \quad i = 1 \ldots 3$$

Fig. 17-44 Illustration of how a speed-synchronous additive lowtime measurement (measurement procedure "Additive Lowtime [µs]") works. The additive time is the total of the line segments shown in bold face.

Functional Description: Asynchronous Additive Pulse-Width Measurements

With asynchronous additive pulse-width measurements, the time in which the signal to be measured is active is totalled. The total in the DPRAM is updated at every closing pulse edge and at the latest after an updating period defined by the user. Fig. 17-45 shows when the additive time is updated using an additive lowtime measurement. The update period is set by the user in the "Update Period for Additive Measurements [µs]" parameter in the "Globals" tab of the ES4330-XMI subsystem. The totalled time is reset to 0 if the ES4330 board is initialized or configured.

Note

The additive time is acquired modulo 2^{52} µs, i.e. after (2^{52}-1) µs (corresponds to approx. 143 years), the additive time returns to 0. (2^{52}-1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

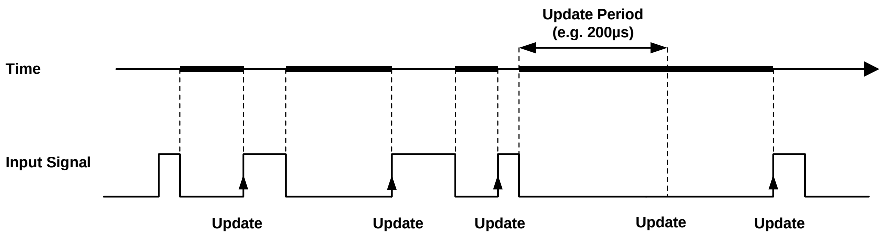


Fig. 17-45 Illustration of how an asynchronous additive lowtime measurement (measurement procedure "Additive Lowtime [µs]") works. The additive time results from the time segments shown in bold face. The times at which the total is updated in the DPRAM are also shown.

Timeout Monitoring

Tab. 17-13 lists the condition of when a timeout is initiated for different timeout monitoring procedures. The measurement value issued in a timeout is specified in Tab. 17-14.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "Intvl Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel *and* the signal has been inactive since the last timeout check. In speed-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"CS Angle InpDep" "CS Angle Predef"	A timeout is initiated if no edge (falling or rising) has been recognized on the hardware channel or input channel since the last timeout check. In speed-synchronous measurement mode, it is of no significance whether the edges occur inside or outside the angle window.
"Intvl & CS Angle InpDep" "Intvl & CS Angle Predef"	The timeout condition for this monitoring procedure results from an OR operation of the timeout condition of the "Intvl InpDep" and "Intvl Predef" procedures with the timeout condition of the "CS Angle InpDep" and "CS Angle Predef" procedures.

Tab. 17-13 Additive pulse-width measurements: timeout condition for different timeout monitoring procedures

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	Asynchronous measurement mode: The additive time determined up to the point where a check is made for a timeout is issued as measurement value in a timeout. Speed-synchronous measurement mode: 0 is issued as measurement value in a timeout.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-14 Additive pulse-width measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.18 Measurement Procedure - Pulse-Width Measurements with Enable or Validate Signal

In addition to the direct acquisition of the pulse widths of a signal which is independent of all other inputs, the ES4330 board also makes it possible to measure pulse widths of a signal dependent on a second input. This second input signal is interpreted as an Enable or Validate signal as shown in Fig. 17-46 on page 537 in the example of a validated lowtime measurement:

If, for example, the ES4330 board is configured in such a way that low times of an input signal A are to be measured whereby a second input signal B is to be used as a high-active Enable signal, only the pulse marked with 1 would be measured by the input signal in Fig. 17-46 on page 537. More generally: when using a second input signal as "Enable", only those pulses are measured which are *completely surrounded* by an active Enable pulse.

If, on the other hand, a further input signal C is used as a high-active Validate signal, only the pulse marked with 2 would be measured in the above example. More generally: when using a second input signal as "Validate", only those pulses are measured which *completely surround* an activate Validate pulse.

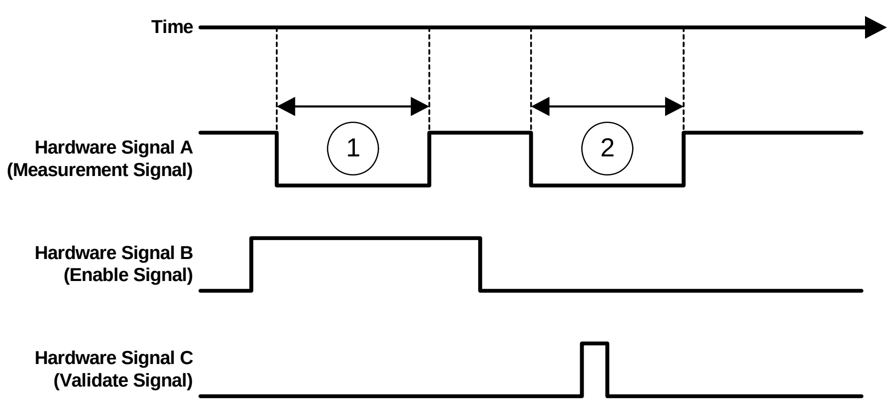


Fig. 17-46 Validated Time Measurements with the ES4330 Board

The ES4330 RTIO driver offers pulse-width measurements with an Enable or Validate option. The Enable and Validate pulses can be defined either as high-active or low-active. The functioning of the following measurements is described in this section:

- "Hightime using H-Enable [µs]"
- "Hightime using L-Enable [µs]"
- "Hightime using H-Validate [µs]"
- "Hightime using L-Validate [µs]"
- "Lowtime using H-Enable [µs]"
- "Lowtime using L-Enable [µs]"
- "Lowtime using H-Validate [µs]"
- "Lowtime using L-Validate [µs]"
- "H-Time n-th Pulse (H-Ena) [µs]"
- "H-Time n-th Pulse (L-Ena) [µs]"
- "H-Time n-th Pulse (H-Val) [µs]"
- "H-Time n-th Pulse (L-Val) [µs]"
- "L-Time n-th Pulse (H-Ena) [µs]"
- "L-Time n-th Pulse (L-Ena) [µs]"
- "L-Time n-th Pulse (H-Val) [µs]"
- "L-Time n-th Pulse (L-Val) [µs]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- "Reference Channel": hardware channel at which the Enable or Validate signal is pending
- pulse selection (only with "H-Time n-th Pulse ..." and "L-Time n-th Pulse ..."): "(Max) Pulse (Count)" defines the number of the pulse to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description: Speed-Synchronous Pulse-Width Measurements with Enable or Validate Signal

Fig. 17-47 illustrates how speed-synchronous pulse-width measurements with an Enable option work using a lowtime measurement with a high-active Enable signal. Only those pulses are measured whose opening and closing edges are within one and the same angle window and which are completely surrounded by an active Enable pulse (valid pulse).

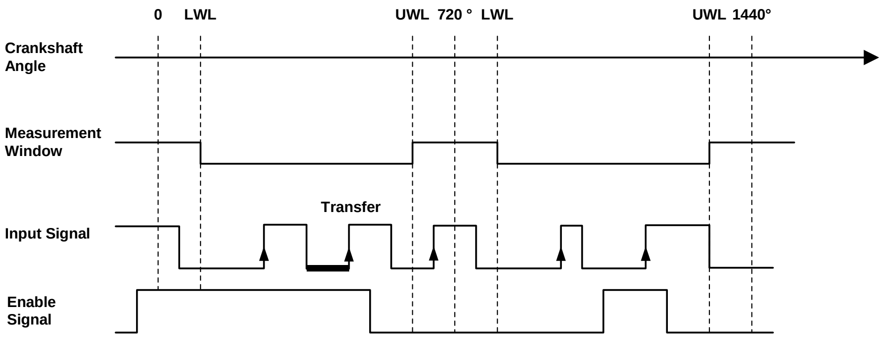


Fig. 17-47 Illustration of how a speed-synchronous pulse-width measurement with an Enable signal (measurement procedure "Lowtime using H-Enable [µs]") works. Measured pulses are shown by a thicker line.

Fig. 17-48 shows how speed-synchronous pulse-width measurements with a Validate option work using a lowtime measurement with a high-active Validate signal. Only those pulses are measured whose opening and closing edges are within one and the same angle window and which completely surround an active Validate pulse (valid pulse).

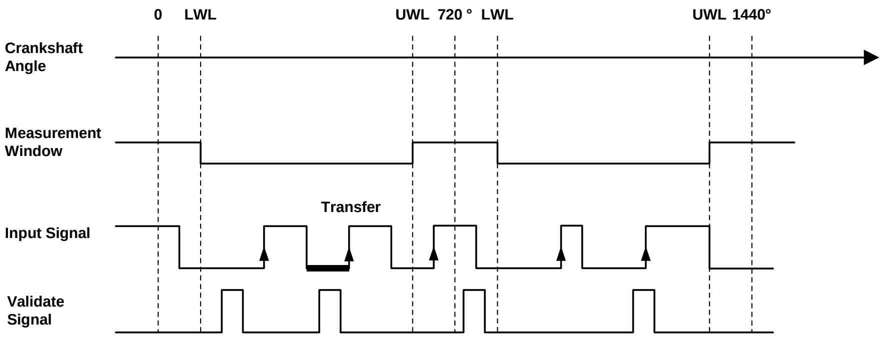


Fig. 17-48 Illustration of how a speed-synchronous pulse-width measurement with a Validate signal (measurement procedure "Lowtime using H-Validate [µs]") works. Measured pulses are shown by a thicker line.

With the *non-pulse-selective measurements,* each valid pulse is measured - the measured pulse width is transferred to the DPRAM of the ES4330 board on the closing edge of the pulse.

With the *pulse-selective measurements,* the n-th valid pulse within an angle window is measured. The pulse number is defined with the "(Max) Pulse (Count)" parameter of the "Signals" tab. The measured pulse width is transferred on the closing edge of the n-th valid pulse.

Note

If 0 is set as pulse number for pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Functional Description: Asynchronous Pulse-Width Measurements with an Enable or Validate Signal

The *non-pulse-selective measurements* measure the pulse width of each pulse with an active level validated by an Enable or Validate signal and transfer the measurement value to the DPRAM on the closing edge of the pulse.

The *pulse-selective pulse-width measurements* are unlikely to have any practical application in asynchronous mode. The pulse width of the n-th pulse with an active level validated by an Enable or Validate signal is measured after the ES4330 board is initialized or configured. The measurement value is transferred to the DPRAM on the closing edge of the measured pulse.

Note

If 0 is set as pulse number for pulse-selective measurements, these measurements behave like the relevant non-pulse-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-15.an

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. If the signal level is active when the timeout check takes place, the length of time since the last opening edge of the signal is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-15 Pulse-width measurements with an Enable or Validate signal: measurement value in a timeout with different timeout monitoring procedures

17.4.19 Measurement Procedure - Frequency and Cycle Time Measurements

The ES4330 board makes it possible to measure frequencies and cycle times at rising and falling signal edges. The user specifies the edge at which the measurement value is calculated by selecting a measurement procedure. The following measurement procedures are available:

- "Cycle Time --/-- [µs]"
- "Cycle Time --\-- [µs]"
- "Frequency --/-- [Hz]"
- "Frequency --\-- [Hz]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

In asynchronous measurement mode, measurement values are calculated at every active signal edge. In speed-synchronous (angle-segmented) measurement mode, measurement values are calculated at every active signal edge within an angle window. Fig. 17-49 shows the calculation of measurement values with angle-segmented frequency or cycle time measurement at rising signal edges.

Note

If the entire angle range of 720° CA is released in an angle-segmented cycle time or frequency measurement, this measurement behaves like the relevant measurement in asynchronous mode.

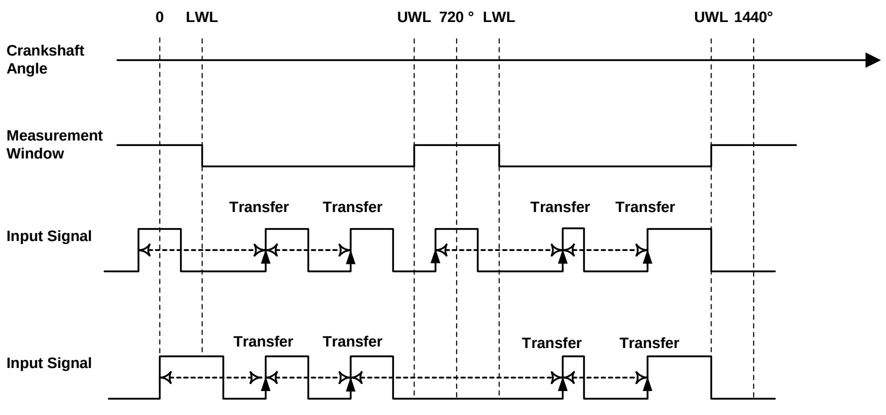


Fig. 17-49 Illustration of measurement value calculation in angle-segmented frequency or cycle time measurement at rising signal edges

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-16.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	*Cycle time measurements:* The length of time between the time the last active edge of the input signal occurred and the time at which a timeout check was executed is issued as measurement value (see Fig. 17-50). *Frequency measurements:* Measurement value calculation in a timeout takes place as for cycle time measurements but a resulting frequency is determined by inversion from the duration determined.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-16 Frequency and cycle time measurements: measurement value in a timeout with different timeout monitoring procedures

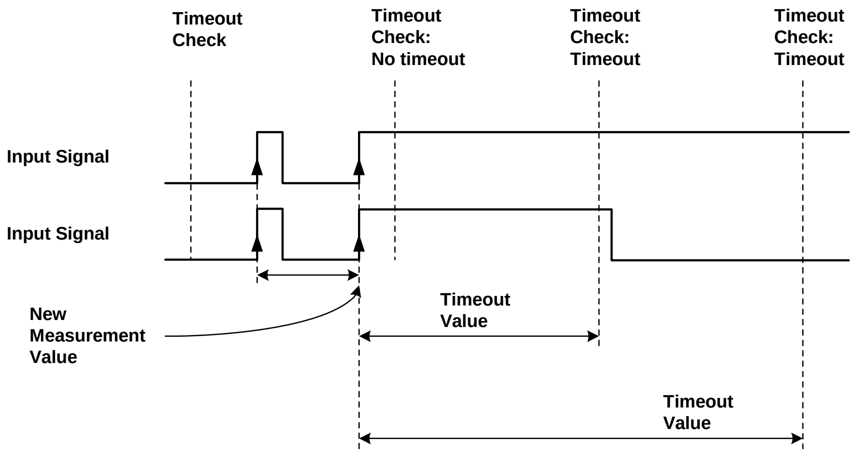


Fig. 17-50 Frequency and cycle time measurements at rising edges: illustration of measurement value calculation in a timeout (the measurement value is referred to in the figure as "Timeout Value") in the "... InpDep" timeout monitoring procedure

17.4.20 Measurement Procedure - Duty Cycle Measurements

Duty cycles (e.g. of PWM signals) can be determined in various ways with the ES4330 board. With duty cycle measurements, a distinction is made between measurements which set the high-phase of a signal in relation to the low-phase (or vice versa) and measurements which set the high-phase (or low-phase) of a signal in relation to the cycle duration. The first type of duty cycle measurement is referred to below as duty cycle $^{P/P}$; "P/P" shows that two pulse durations are set in relation to one another.

The second type of duty cycle measurement is referred to below as duty cycle $^{P/C}$; "P/C" shows that a pulse duration is set in relation to the cycle time of the signal.

$$\text{Duty Cycle}^{P/P}: \frac{L}{H}, \frac{H}{L} \qquad 0 \le \text{Duty Cycle}^{P/P} \le \infty$$

$$\text{Duty Cycle}^{P/C}: \frac{L}{L+H}, \frac{H}{L+H} \qquad 0 \le \text{Duty Cycle}^{P/C} \le 1$$

The user specifies the edge at which the measurement value is calculated by selecting the measurement procedure. The following "P/P" duty cycle measurements are available:

- "Duty Factor L/H --/--"
- "Duty Factor L/H --\--"
- "Duty Factor H/L --/--"
- "Duty Factor H/L --\--"

The following "P/C" duty cycle measurements are available:

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

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

As far as measurement value calculation in asynchronous and speed-synchronous (angle-segmented) measurement mode is concerned, exactly the same is true of duty cycle measurements as was described for frequency and cycle time measurements in section 17.4.19 on page 541.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-17 on page 544.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	$\text{Duty Cycle}^{P/P}: \frac{L}{H}, \frac{H}{L}$ • The measurement value is 0.0 if the signal level is "high" for L/H measurements or "low" for H/L measurements when the time-out check is carried out. • The measurement value is MAXREAL32 if the signal level is "low" with L/H measurements or "high" with H/L measurements when the timeout check is carried out. $\text{Duty Cycle}^{P/C}: \frac{L}{L+H}, \frac{H}{L+H}$ • The measurement value is 0.0 if the signal level is "high" with L/(L+H) measurements or "low" with H/(L+H) measurements when the timeout check is carried out. • The measurement value is 1.0 if the signal level is "low" with L/(L+H) measurements or "high" with H/(L+H) measurements when the timeout check is carried out.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-17 Duty cycle measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.21 Measurement Procedure - Relative Measurements between Hardware Channels

Functional Description

The measurement procedures described so far all had one thing in common: the measured times, frequencies, angles and duty cycles were defined by the signal edges of *one* hardware channel. This section describes measurement procedures in which events (edges) are measured on two hardware channels in relation to one another.

The measurement functionality is explained based on Fig. 17-51. The time from the opening edge of an Enable pulse to the n-th active edge on a reference signal is measured. Prerequisite for the determination of a measurement value is, however, that the n-th active edge is validated by the Enable pulse, i.e. the edge has to occur during the active phase of the Enable signal. The counting of the active edges of the reference signal is reset to 0 with the opening edge of the Enable pulse. The n parameter for edge selection is set in the "(Max) Pulse (Count)" option field in the "Signals" tab. Fig. 17-52 shows a signal trace during which there is no validation of the reference signal.

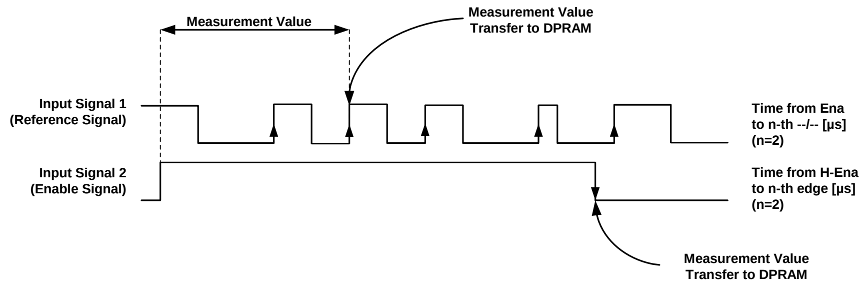


Fig. 17-51 How the measurement procedure for measuring events (edges) on different signals or hardware channels in relation to one another works. A measurement with a high-active Enable signal is shown. The RTIO parameter on edge selection ("(Max) Pulse (Count)" in the "Signals" tab) is greater than 0.

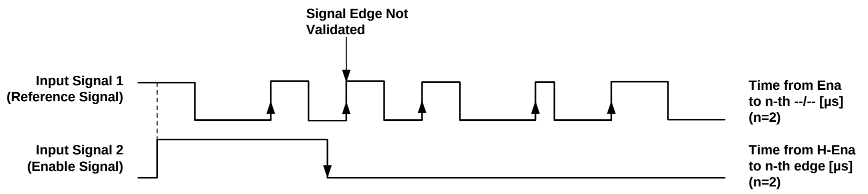


Fig. 17-52 Measuring of edges on different hardware channels: signal trace with high-active Enable signal and non-validated n-th active signal edge of the reference signal

Executing the described measurements requires the configuration of two measurement or software channels. The Enable signal must be assigned to a measurement channel. Depending on whether the Enable signal is low-active or high-active, one of the following two measurement procedures must be assigned to the measurement channel:

- "Time from L-Ena to n-th edge [µs]"
- "Time from H-Ena to n-th edge [µs]'"

Admissible measurement options are:

- "Reference Channel": refers to the measurement channel to which the reference signal is assigned
- timeout monitoring

The settings of all other measurement options are ignored.

The reference signal must be assigned to the other measurement channel. Depending on whether the active edge of the reference signal is rising or falling, one of the following two measurement procedures must be assigned to the measurement channel:

- "Time from Ena to n-th --/-- [µs]"
- "Time from Ena to n-th --\-- [µs]"

Admissible measurement options are:

- "Reference Channel": refers to the measurement channel to which the Enable signal is assigned
- edge selection: "(Max) Pulse (Count)" defines the number of the active edge of the reference signal to be measured

The settings of all other measurement options are ignored.

Note

Segmenting of angle windows is not supported. There is therefore no difference in the functionality of the measurements in asynchronous and in speed-synchronous measurement mode.

Basically both measurement channels of a measurement return the determined measurement value. The only difference is the time at which the measurement value is transferred to the DPRAM of the ES4330 board.

Measurement channels which are configured to the measurement procedure "Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]" transfer the measurement value when the n-th validated active edge occurs (see Fig. 17-51).

Measurement channels which are configured to the measurement procedure "Time from L-Ena to n-th edge [µs]" or "Time from H-Ena to n-th edge [µs]" transfer the measurement value when the closing edge of the Enable pulse occurs (see Fig. 17-51).

There is a special measurement response when the "(Max) Pulse (Count)" parameter for edge selection is set to 0 in the measurement procedure "Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]". In this case, the measurement procedures "Time from L-Ena to n-th edge [µs]" and "Time from H-Ena to n-th edge [µs]" return the time from the opening edge of the Enable pulse to the *last* validated active edge of the reference signal. The measurement value is transferred to the DPRAM *at the closing edge of the Enable pulse* (Fig. 17-53). In this case, the measurement procedures "Time from Ena to n-th --/-- [µs]" and "Time from Ena to n-th --\-- [µs]" return the time from the opening edge of the Enable pulse to *each* validated active edge of the reference signal. The measurement values are transferred to the DPRAM when the relevant validated active edge occurs (Fig. 17-53).

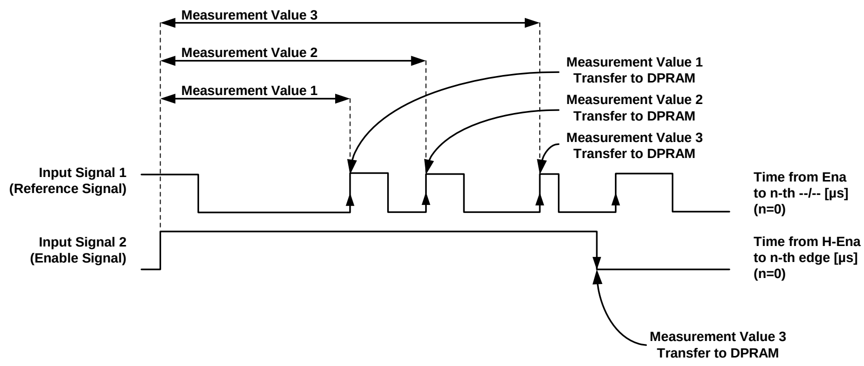


Fig. 17-53 How the measurement procedure for measuring events (edges) on different signals or hardware channels in relation to one another works. A measurement with a high-active Enable signal is shown. The RTIO parameter on edge selection ("(Max) Pulse (Count)" in the "Signals" tab) is equal to 0.

Tab. 17-18 lists configuration guidelines and measurement values of the measurement procedures for measuring edges on different hardware channels.

RTIO Parameters of the "Signals" Tab	Enable Signal	Reference Signal
Hardware Channel	Number of the hardware channel carrying the Enable signal.	Number of the hardware channel carrying the reference signal.
Measurement Mode	"Asynchronous" or "ESpeedSync". There is no difference between the modes.	"Asynchronous" or "ESpeedSync". There is no difference between the modes.
Measurement Procedure	"Time from L-Ena to n-th edge [µs]" or "Time from H-Ena to n-th edge [µs]" depending on whether the Enable signal is low- or high-active.	"Time from Ena to n-th --/-- [µs]" or "Time from Ena to n-th --\-- [µs]" depending on whether the active edge of the reference signal is rising or falling.
Reference Channel	Number of the measurement channel to which the reference signal has been assigned.	Number of the measurement channel to which the Enable signal has been assigned.
Measurement value with the setting (Max) Pulse (Count) = 0	The time from the opening edge of the Enable pulse to the *last* validated active edge of the reference signal is measured and transferred to the DPRAM *at the closing edge of the Enable pulse*.	The time from the opening edge of the Enable pulse to *each* validated active edge of the reference signal is measured and transferred to the DPRAM *when the relevant active edge occurs*.
Measurement value with the setting (Max) Pulse (Count) > 0	The time from the opening edge of the Enable pulse to the n-th validated active edge of the reference signal is measured and transferred to the DPRAM *at the closing edge of the Enable pulse*.	The time from the opening edge of the Enable pulse to the n-th validated active edge of the reference signal is measured and transferred to the DPRAM *when the n-th validated active edge occurs*.

Tab. 17-18 Configuration guidelines and measurement values of the measurement procedures for measuring edges on different hardware channels

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-19.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	If the signal level of the Enable signal is inactive when the timeout check takes place, 0 is issued as measurement value. If the signal level of the Enable signal is active when the timeout check takes place, 1 is issued as measurement value.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-19 Measurement procedure for measuring edges on different hardware channels: measurement value during timeout with different procedures of timeout monitoring

17.4.22 Measurement Procedure - Measuring Edges: Angle Stamp

Functional Description

The following measurement procedures measure the crankshaft angle of active signal edges on the assigned hardware channel in relation to the reference angle set.

- "Rising Edge --/-- [deg]"
- "Falling Edge --\-- [deg]"
- "Rising Edge of n-th Pulse [deg]"
- "Falling Edge of n-th Pulse [deg]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- edge selection (only with "Rising Edge of n-th Pulse [deg]" and "Falling Edge of n-th Pulse [deg]"): "(Max) Pulse (Count)" defines the number of edges to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- "CS Angle Reference": defines the reference angle of the measurement
- timeout monitoring

The crankshaft angle of an edge provided by the hardware is always in the range [0° CA, 720° CA[. In *speed-synchronous measurement mode*, the measured angle is mapped into the range [LWL, LWL + 720°CA[by the ES4330 firmware and then the difference is calculated with the reference angle. This difference is the measurement value provided by the measurement; it is positive when the mapped angle is smaller than the reference angle (cf. Fig. 17-54).

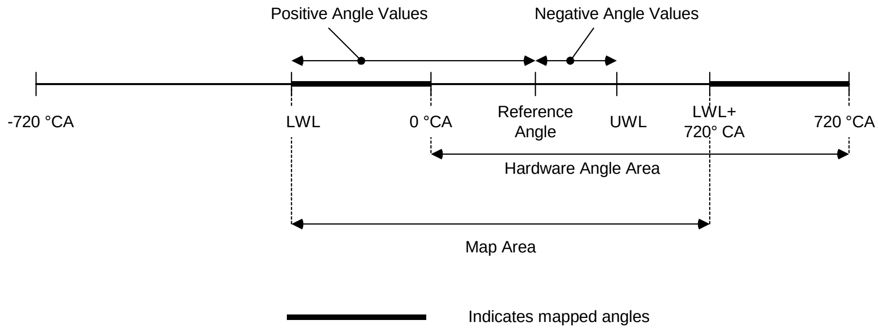


Fig. 17-54 How angle measurement works in speed-synchronous measurement mode

As the reference angle can be in the range [-720° CA, 720° CA], the measurement functions return angle values from the range]-1440 °CA, 1440 °CA] in speed-synchronous measurement mode. In *asynchronous measurement mode,* there is no mapping of the crankshaft angle provided by the hardware which means that the angle values are in the range]-1440 °CA, 720° CA] in this measurement mode.

In *speed-synchronous measurement mode,* the *non-edge-selective measurements* ("Rising Edge --/-- [deg]", "Falling Edge --\-- [deg]") acquire every active edge within one angle window. The *edge-selective measurements* ("Rising Edge of n-th Pulse [deg]", "Falling Edge of n-th Pulse [deg]") measure the n-th active edge within an angle window. When the lower angle window limit is undercut, the edge counter is reset to 0.

In *asynchronous measurement mode,* the *non-edge-selective measurements* acquire the crankshaft angle of every active edge. The *edge-selective measurements* are not useful in this measurement mode as the edge counter is not reset.

Note

If 0 is set as edge number for the edge-selective measurements in the "(Max) Pulse (Count)" option field, these measurements react like the relevant non-edge-selective measurements.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-20.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the low level for a measurement at rising edges and the high level for a measurement at falling edges. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the high level for a measurement at rising edges and the low level for a measurement at falling edges.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-20 Angle measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.23 Measurement Procedure - Measuring Edges: Time Stamp

Functional Description

The following measurement procedures acquire the time of active signal edges on the assigned hardware channel.

- "Time Stamp --/-- [µs]"
- "Time Stamp --\-- [µs]"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- "(Max) Pulse (Count)": defines the number of the edge to be measured
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

In *speed-synchronous measurement mode*, the time the n-th active signal edge occurs within an angle window is measured. If the lower angle window limit is undercut, the edge counter is reset to 0. If 0 is set as edge number in the "(Max) Pulse (Count)" option field, each active edge within an angle window is measured.

If in *asynchronous measurement mode,* the edge number is set as being 0 in the "(Max) Pulse (Count)" option field, the time each active signal edge occurs is measured. If the edge number is not equal to 0, the time of the n-th active edge is measured once after an initialization or a configuration step.

The times are specified in relation to the time of the last initialization or configuration of the ES4330 board. In other words: an initialization or configuration of the ES4330 board resets the clock to 0.

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-21.

Timeout Monitoring	**Measurement Value in a Timeout**
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The last measurement value (calculated before the timeout) is retained as measurement value in a timeout.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-21 Time stamp measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.24 Measurement Procedure - Pulse Count

This section describes the properties of the following pulse-count measurement procedures:

- "Number of Low-Pulses"
- "Number of High-Pulses"
- "Total Number of L-Pulses"
- "Total Number of H-Pulses"

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Functional Description

The "Number of Low-Pulses" and "Number of High-Pulses" measurement procedures acquire the number of valid active pulses within an angle window in *speed-synchronous measurement mode*. Valid pulses are pulses which are in the same angle window with both opening and closing edge. The total calculated is transferred to the DPRAM at the upper angle window limit. In speed-synchronous measurement mode, the "Total Number of L-Pulses" and "Total Number of

H-Pulses" measurement procedures acquire the total of all valid pulses since the last initialization or configuration of the ES4330 board. The total is updated at every upper angle window limit (Fig. 17-55).

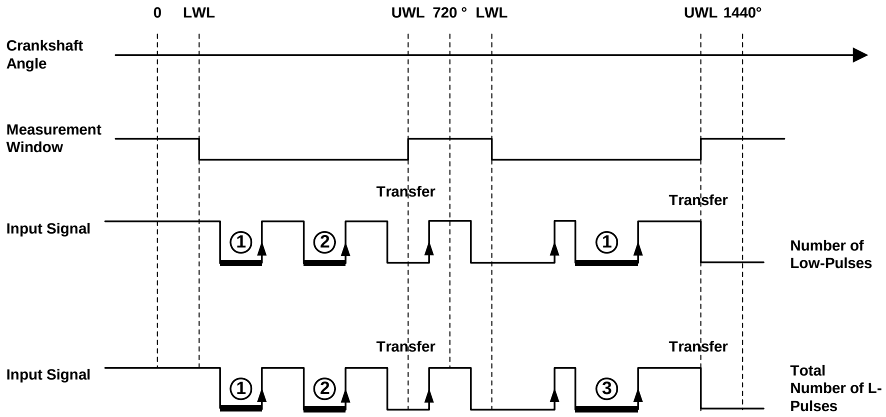


Fig. 17-55 Illustration of how speed-synchronous measurement procedures for pulse count (measurement procedures "Number of Low-Pulses" and "Total Number of L-Pulses") work. Pulses which have been counted are shown with thicker lines. The total is transferred to the DPRAM at the upper angle window limit.

In *asynchronous measurement mode* the "Number of ..." and "Total Number of ..." measurements have identical functions. The total number of active pulses since the last initialization or configuration of the ES4330 board is measured. The total is updated at each closing pulse edge.

Note

The pulse count takes place with modulo 2^{52}, i.e. after (2^{52}-1) pulses, the total returns to 0. (2^{52}-1) is the largest whole number of the type "double precision" which can be displayed with a resolution of 1.0.

Timeout Monitoring

A timeout is initiated if no valid active pulse has occurred since the last timeout check. In speed-synchronous measurement mode, a valid active pulse is an active pulse whose opening and closing edges are within one and the same angle window. In asynchronous measurement mode, every active pulse is a valid, active pulse. The measurement value issued in a timeout is specified in Tab. 17-22.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. An inactive signal level is the high level when low pulses are counted and the low level when high pulses are counted. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value. An active signal level is the low level when low pulses are counted and the high level when high pulses are counted.
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-22 Pulse count: measurement value in a timeout with different timeout monitoring procedures

17.4.25 Measurement Procedure - Level Measurement

Functional Description

The measurement procedures

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

are used to acquire the active level states on a hardware channel. An active level is signalized by the measurement value "one" (1); an inactive level is shown by the measurement value "zero" (0). In *asynchronous measurement mode,* the measurement value is updated at each edge of the input signal. In *speed-synchronous measurement mode*, updating only takes place at signal edges which occur within the angle window.

Admissible measurement options are:

- "Measurement Mode": speed-synchronous or asynchronous measurement mode
- angle window ("CS Angle Lower Limit" and "CS Angle Upper Limit") in speed-synchronous measurement mode
- timeout monitoring

Timeout Monitoring

A timeout is initiated if no measurement value has been able to be calculated or determined since the last timeout check. The measurement value issued in a timeout is specified in Tab. 17-23.

Timeout Monitoring	Measurement Value in a Timeout
"Intvl InpDep" "CS Angle InpDep" "Intvl & CS Angle InpDep"	The signal level when the timeout check takes place is issued as measurement value. - If the signal level is inactive when the timeout check takes place, 0 is issued as measurement value. - If the signal level is active when the timeout check takes place, 1 is issued as measurement value
"Intvl Predef" "CS Angle Predef" "Intvl & CS Angle Predef"	The value specified in the "Default Timeout Value" option field is issued as measurement value.

Tab. 17-23 Level measurements: measurement value in a timeout with different timeout monitoring procedures

17.4.26 Measurement Procedure - Position Tracing with Two-Phase Stepper Motors

Functional Description

The ES4330 firmware makes position tracing possible and thus effectively the acquisition of rotor position and speed of two-phase stepper motors. For this, two hardware channels and two software channels or measurement channels are required. At the two hardware channels, the input signals of the two phase windings of a two-phase stepper motor are applied. The 'Step Count (Step. Mot. Phase A)' measurement procedure must be assigned to one measurement channel and 'Step Count (Step. Mot. Phase B)' to the other. The "Hardware Channel" field of the measurement channels contains a reference to the number of the hardware channel at which the input signal for the relevant phase winding of the two-phase stepper motor is pending. In the "Reference Channel" field of a measurement channel, there must be a reference to the number of the other measurement channel involved in the measurement. Fig. 17-56 illustrates the procedure involved in configuration. The input signals of the phase windings are pending at hardware channels 2 and 3. The measurements are executed at measurement channels 0 and 1.

The configuration guidelines are compiled in Tab. 17-24.

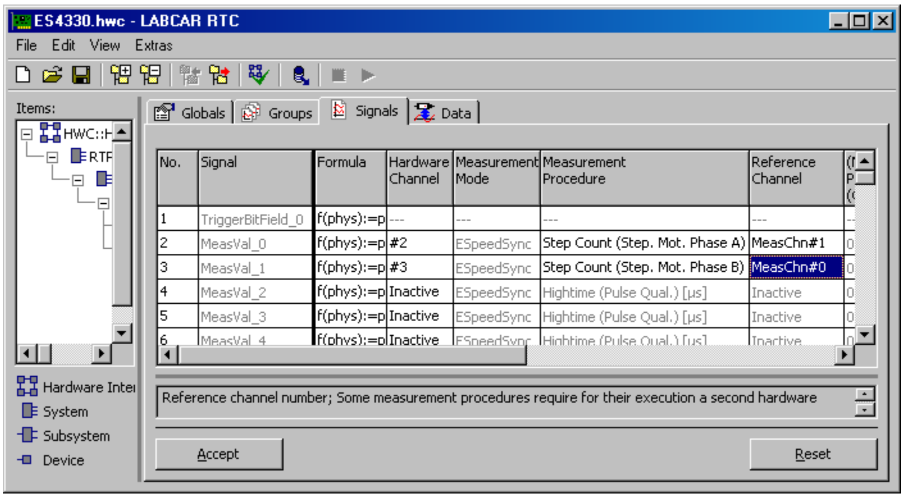


Fig. 17-56 Configuration of a measurement for position tracing with two-phase stepper motors

At each edge of the two-channel signal, a position counter is incremented or decremented (Fig. 17-57) and the value of the position counter in the DPRAM of the ES4330 board updated. The measurement channel to which the measurement procedure "Step Count (Step. Mot. Phase A)" is assigned, always returns a non-negative position value. In the measurement value of the measurement channel to which the measurement procedure "Step Count (Step. Mot. Phase B)" is assigned, the sign of the position value shows the direction of rotation of the motor (coded). An ascending position counter occurs when the signal at "Phase A" preceeds the signal at "Phase B"(Fig. 17-57).

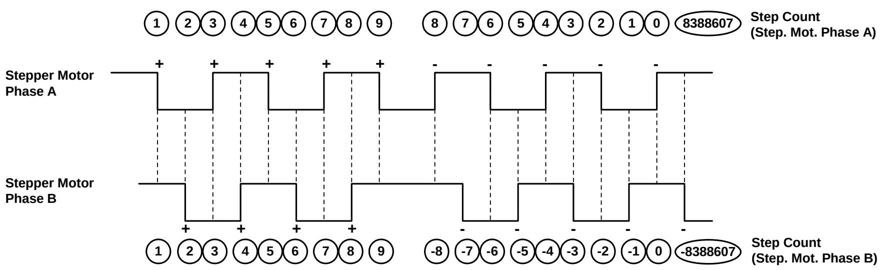


Fig. 17-57 Illustration of how position tracing with two-phase stepper motors works

Note

Position tracing follows modulo 2^{23}. Incrementing at a position status of (2^{23}-1) generates 0 as the value of the position counter. Decrementing at a position status of 0 generates a value of the position counter which in terms of its absolute value is equal to (2^{23}-1).

RTIO Parameters of the "Signals" Tab	Phase A	Phase B
Hardware Channel	Number of the hardware channel at which the input signal for phase winding A is pending.	Number of the hardware channel at which the input signal for phase winding B is pending.
Measurement Mode	Of no significance	Of no significance
Measurement Procedure	"Step Count (Step. Mot. Phase A)"	"Step Count (Step. Mot. Phase B)"
Reference Channel	Number of the measurement channel to which the Phase B signal was assigned.	Number of the measurement channel to which the Phase A signal was assigned.
Measurement Value	Current value of the position counter modulo 2^{23}.	Current value of the position counter modulo 2^{23}. The direction of rotation of the motor is coded in the sign.

Tab. 17-24 Configuration guidelines and measurement values of the measurement procedures for position tracing with two-phase stepper motors

Timeout Monitoring

The measurement procedures for position tracing with two-phase stepper motors offer no timeout monitoring.

17.5 ES4350 Carrier Board

This section describes the RTIO integration of the ES4350 Carrier Board which is used as a carrier board for I/O modules in the LABCAR environment.

17.5.1 The Structure of the ES4350-RTIO Tree

An ES4350 Carrier Board is installed in the VXIbus part of an ES4300 Signal Box. It is connected to the experimental target in the VMEbus part of the Signal Box via the ES4315 VME64x/VXI adapter of the Signal Box. The hardware connection described is reflected in the RTIO tree (Fig. 17-58). The ES4350-CB subsystem of an ES4350 board is assigned to the ES4315-VXI subsystem of an ES4315 board.

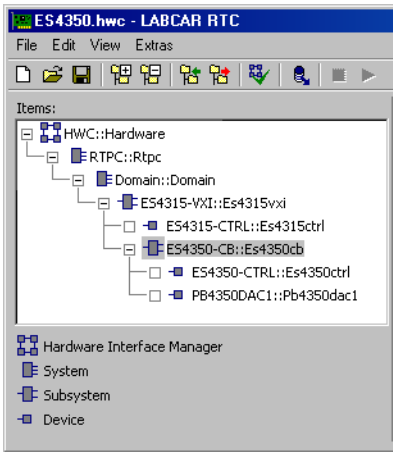


Fig. 17-58 RTIO Hardware Description with Connected ES4350 Carrier Board

ES4350 carrier board hardware has a synchronization bus with six lines as well as two identical buses with signals for synchronization to the crank angle especially for testing engine ECUs. The bus lines are configured and controlled using the ES4350-CTRL device.

17.5.2 Globals (ES4350-CB Subsystem)

Fig. 17-59 shows the RTIO parameters of the "Globals" tab.

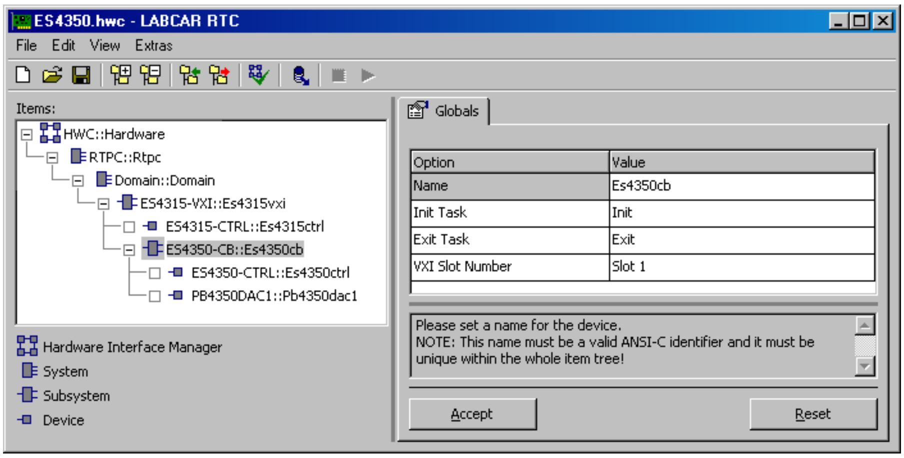


Fig. 17-59 The "Globals" Tab of the ES4350-CB Subsystem

VXI Slot Number

Number of the VXIbus slot holding the ES4350 board.

The visible VXIbus slots within the ES4300 Signal Box are numbered from left to right starting with 1. The ES4315 ES4315 VME64x/VXI adapter is permanently stored in the VXIbus slot 0 which is not visible to the user.

The parameter cannot be edited online. Tab. 17-25 summarizes the properties of the "VXI Slot Number" parameter.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
VXI Slot Number	uint32	No	VXIbus slot holding the ES4350 board.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-25 ES4350-CB Subsystem: Configuration Parameters of the "Globals" Tab

17.5.3 The ES4350-CTRL Device

The ES4350-CTRL device is used to configure and control ES4350 CAC (crankshaft angle clock) and synchronization buses.

Functional Description

In LABCAR projects, it is often necessary to synchronize sequences. Projects for testing engine ECUs, for example, require precise angle synchronization between the generation of injection and ignition signals on the ECU and their being measured on test hardware (e.g. ES4330 boards). Engine ECU and test hardware are supplied with the same crankshaft angle clock in LABCAR projects.

ES4300-based LABCAR systems provide two independent CAC buses on the ES4300 backplane, each with 3 lines (see Fig. 17-60 on page 560). One bus line is used to synchronize to the zero crossing of the crankshaft angle, the second transfers the actual angle clock and the running direction of the engine is coded on the third.

Like every hardware bus, a CAC bus can only be driven by one source. In addition to CAC generators on ES4320 boards, ES4350 boards are also possible sources. At the time of writing, however, this possibility is limited as there are as yet no piggybacks for CAC generation on ES4350 boards.

Receivers of CAC signals are:

- ES4330 boards which use the signals to measure speed-synchronous signals precisely
- ES4320 boards operated in Slave mode which use the signals to synchronize their signal generators to an external crankshaft angle clock
- ES4350 boards which transfer the crankshaft angle clock to the piggybacks

An ES4350 board has two internal CAC buses. Each of these buses can

- be driven by one of the two backplane CAC buses or
- by the board's FPGA
- or by one of the six piggybacks with the restriction that there is currently no suitable module (as has already been mentioned).

In addition to the CAC buses, the ES4300 backplane and ES4350 board also have other lines which are used to synchronize hardware units with each other. The synchronization bus of an ES4350 board, consisting of six lines, is routed to the FPGA of the board and to the six piggybacks. Unlike a CAC bus, whose lines form one unit, the lines of the synchronization bus are independent from one another. A piggyback can thus, e.g., drive one synchronization line and at the same time react to events on another synchronization line.

A synchronization bus line can only be driven by one source. Possible sources are the piggybacks as well as the software (Real-Time Execution Connector) which can control the levels on the synchronization lines of the ES4350 board by accessing the board's internal registers.

Each of the six synchronization lines of an ES4350 board can be switched to one of the two synchronization lines on the backplane of an ES4300 Signal Box. In this way, hardware and piggybacks can be synchronized on different ES4350 boards.

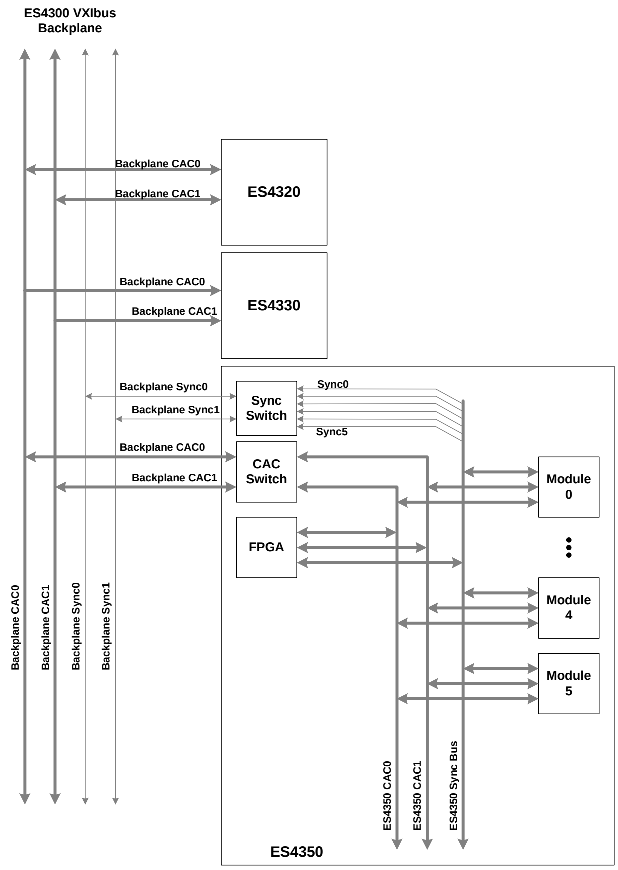


Fig. 17-60 ES4350 CAC Bus and Synchronization Bus

17.5.4 Globals (ES4350-CTRL Device)

The "Globals" tab of an ES4350-CTRL device is shown in Fig. 17-61.

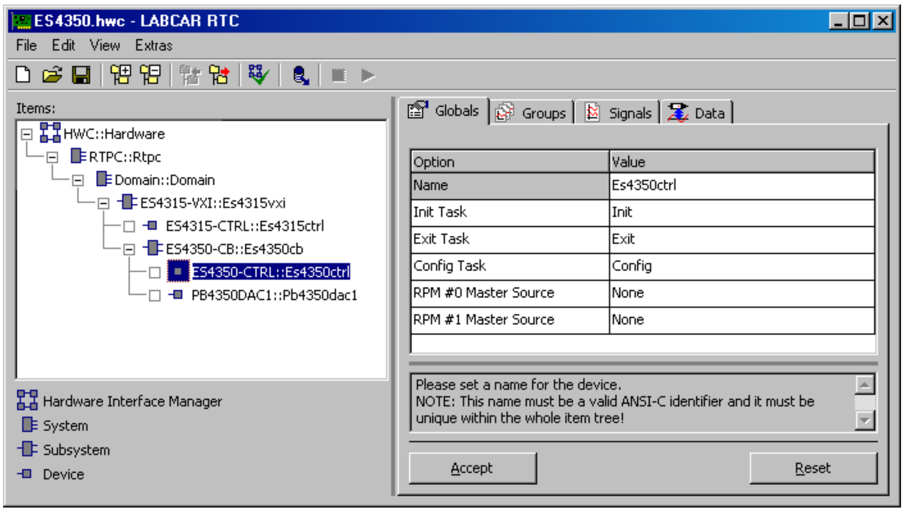


Fig. 17-61 The "Globals" Tab of the ES4350-CTRL Device

RPM #0 Master Source, RPM #1 Master Source

The source which drives the internal CAC bus 0 or 1 of an ES4350 board is specified in this list box. The list box is structured dynamically. Fixed options are:

- **None**

  The CAC bus is not driven.

- **VXI Backplane RPM #0**

  The CAC bus is driven by CAC bus 0 of the ES4300 backplane.

- **VXI Backplane RPM #1**

  The CAC bus is driven by CAC bus 1 of the ES4300 backplane.

If a piggyback that can generate CAC signals has been integrated, this piggyback is also made available in the list for selection.

The list box can be edited online (i.e. during runtime of the model on the experimental target).

Tab. 17-26 summarizes the properties of the individual RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
RPM #0 Master Source RPM #1 Master Source	uint8	Yes	Source for internal CAC line 0 and 1: 0: "None" 1: "VXI Backplane RPM #0" 2: "VXI Backplane RPM #1" 3: Reserved 4: Piggyback in Slot 0 5: Piggyback in Slot 1 6: Piggyback in Slot 2 7: Piggyback in Slot 3 8: Piggyback in Slot 4 9: Piggyback in Slot 5
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-26 ES4350-CTRL Device: Configuration Parameters of the "Globals" Tab

17.5.5 Groups (ES4350-CTRL Device)

The ES4350-CTRL device has one signal group which is transferred from the experimental target to the ES4350 board (see Fig. 17-62). It transports level information for the synchronization lines of an ES4350 board which are driven or controlled by the RTIO.

A real-time operating system task must be assigned to the signal group. Normally a task with periodic activation is selected. The activation period depends on the dynamic behavior of the synchronization signals to be generated. Useful activation periods are limited downwards to 1 ms by runtimes of the firmware on the carrier boards.

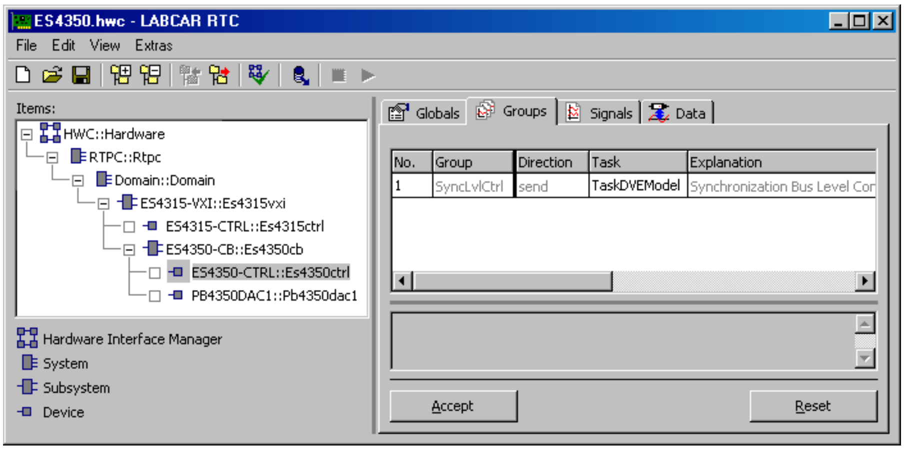


Fig. 17-62 The "Groups" Tab of the ES4350-CTRL Device

17.5.6 RTIO Signals of the "SyncLvlCtrl" Signal Group

The "SyncLvlCtrl" signal group consists of six RTIO signals, "SyncLine_0" to "SyncLine_5", which are of the data type "bool".

If synchronization line x (x = 0, 1, ... 5) is configured so that it can be driven or controlled by RTIO, the value of the RTIO signal "SyncLine_x" defines the level of the line. "0" indicates a low level and "1" a high level.

If synchronization line x is not controlled by RTIO, the signal has no meaning. This is also apparent from the "Data" tab of the RTIO item: the "Data" column of the synchronization lines which are not controlled by RTIO, is inactive. In Fig. 17-63, for example, the synchronization lines 0 to 2 are not controlled by RTIO; the levels of the synchronization lines 3 to 5, however, are controlled by RTIO.

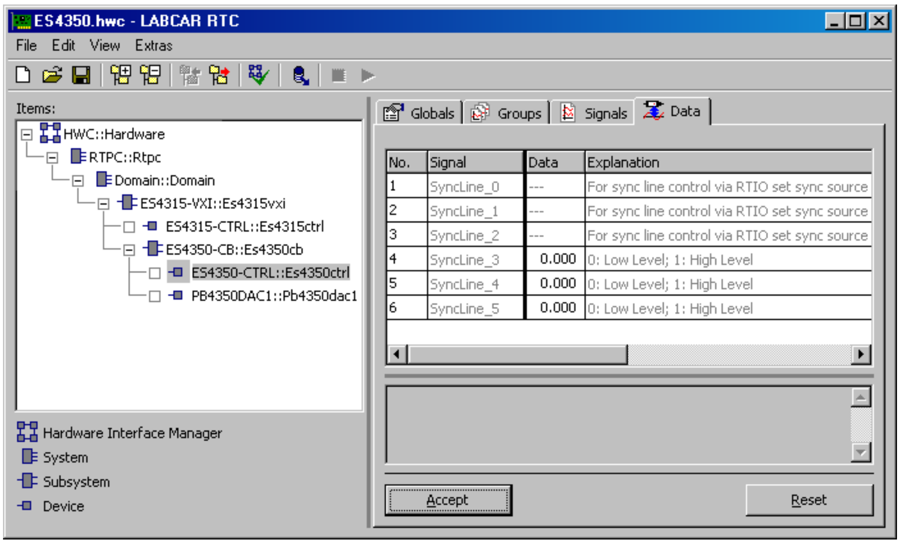


Fig. 17-63 The "Data" Tab of the ES4350-CTRL Device

Tab. 17-27 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment / Value Range
SyncLine_x (x=0, 1, ... 5)	bool	Level control on synchronization line x 0: Low level 1: High level
* Data type which the RTIO driver uses internally for the parameter		

Tab. 17-27 ES4350-CTRL Device: The RTIO Signals of the "SyncLvlCtrl" Signal Group

17.5.7 Signals (ES4350-CTRL Device)

The synchronization lines of an ES4350 board are configured in the "Signals" tab.

Fig. 17-64 shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

Synchronization Line Driving Source

The source which drives the synchronization line is specified in this list box. The list box is structured dynamically. Fixed options are:

- **None**

  The synchronization line is not driven.

- **VXI Backplane SYNC #0**

  The synchronization line is driven by synchronization line 0 of the ES4300 backplane.

- **VXI Backplane SYNC #1**

  The synchronization line is driven by synchronization line 1 of the ES4300 backplane.

- **RTIO**

  The level on the synchronization line is controlled by RTIO.

If an I/O module that can drive synchronization lines has been integrated, this module is also made available in the list for selection.

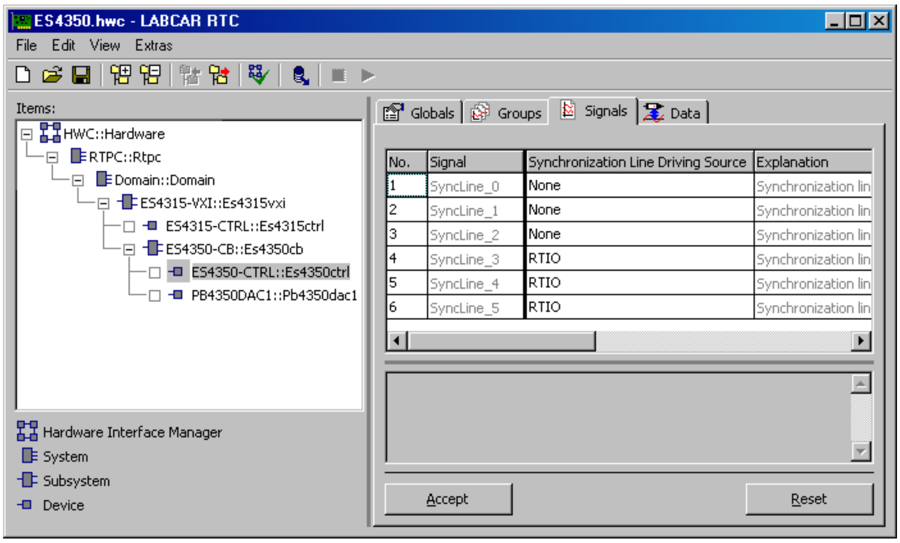


Fig. 17-64 The "Signals" Tab of the ES4350-CTRL Device

Tab. 17-28 summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Synchronization Line Driving Source	uint8	Yes	Source for synchronization line 0: None 1: VXI Backplane SYNC #0 2: VXI Backplane SYNC #1 3: RTIO 4: Piggyback in Slot 0 5: Piggyback in Slot 1 6: Piggyback in Slot 2 7: Piggyback in Slot 3 8: Piggyback in Slot 4 9: Piggyback in Slot 5
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-28 ES4350-CTRL Device: Configuration Parameters of the "Signals" Tab

17.6 The PB4350DAC1 I/O Module

This section describes the RTIO integration of the PB4350DAC1 module.

The PB4350DAC1 module is generally suited for tasks requiring the output of unipolar, analog voltages. The module is typically used for simulating vehicle sensors in LABCAR projects.

17.6.1 The Structure of the PB4350DAC1-RTIO Tree

The PB4350DAC1 module is integrated by selecting the PB4350DAC1 device in the RTIO editor. This item can either be assigned to an ES1651 Carrier Board or an ES4350 Carrier Board. Fig. 17-65 shows the connection of PB4350DAC1 modules to both types of carrier board.

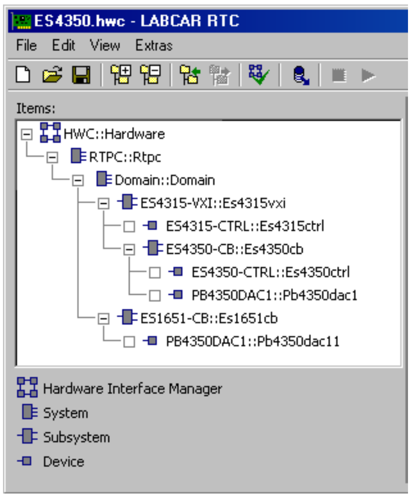


Fig. 17-65 RTIO Hardware Description with Integrated PB4350DAC1 Modules

17.6.2 Globals (PB4350DAC1 Device)

Fig. 17-66 shows the RTIO parameters of the "Globals" tab.

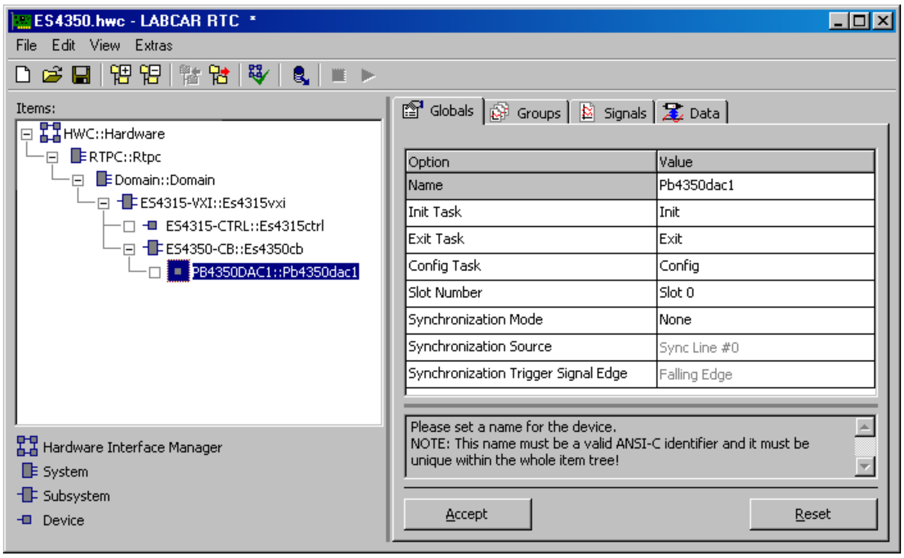


Fig. 17-66 The "Globals" Tab of the PB4350DAC1 Device

Slot Number

The number of the slot of the ES1651 Carrier Board or the ES4350 Carrier Board holding the PB4350DAC1 module is entered in this list box. ES4350 carrier boards have six module slots whose numbers are printed on the front panel.

The numbering of the four slots of an ES1651 carrier board is shown in Fig. 17-67.

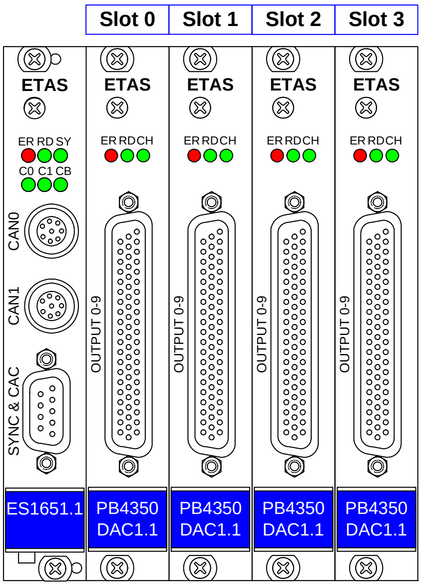


Fig. 17-67 Numbering of the Module Slots of an ES1651 Carrier Board

Synchronization Mode

This list box is where you specify when the analog output values transferred by the PC software (Real-Time Execution Connector) or by the RTIO driver to the PB4350DAC1 module are accepted by the hardware of the PB4350DAC1 module.

If "None" is set in this list box, the analog output values transferred by the PC software are immediately updated at the outputs of the PB4350DAC1 module.

With the "Hardware Triggered" option, updating does not take place until an active edge has been acquired on the line of the synchronization bus of an ES1651 Carrier Board or an ES4350 Carrier Board defined in the "Synchronization Source" list box. In this way, the analog output can be synchronized with an external event.

The list box can be edited online (i.e. during runtime of the model on the experimental target).

Synchronization Source

This is where the synchronization bus line is specified whose active edges initiate an update of the voltages at the channels of a PB4350DAC1 module.

This list box can only be edited if the "Hardware Triggered" option (synchronization of the analog output channels with an external event) has been set in the "Synchronization Mode" list box (see above).

The synchronization bus of an ES1651 carrier board has four lines; that of an ES4350 Carrier Board has six.

The list box can be edited online (i.e. during runtime of the model on the experimental target).

Synchronization Trigger Signal Edge

The active edge ("Rising Edge" or "Falling Edge") which initiates an update of the voltages at the channels of a PB4350DAC1 module is set in this list box.

Like the "Synchronization Source" list box, this list box can only be edited if the "Hardware Triggered" option is set in the "Synchronization Mode" list box.

The list box can be edited online (i.e. during runtime of the model on the experimental target).

Tab. 17-29 summarizes the properties of the configuration parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Slot Number	uint32	No	Slot number of the module on the board. ES1651: 0, 1, 2, or 3 ES4350: 0, 1, ... 5
Synchronization Mode	uint8	Yes	Synchronization mode 1: No synchronization 0: Hardware Triggered
Synchronization Source	uint8	Yes	Synchronization line 0: synchronization line 0 1: synchronization line 1 2: synchronization line 2 3: synchronization line 3 4: synchronization line 4 5: synchronization line 5 (4 and 5 only with ES4350 carrier boards!)
Synchronization Trigger Signal Edge	bool	Yes	Synchronization trigger edge 0: Falling edge 1: Rising edge

* Data type which the RTIO driver uses internally for the parameter

Tab. 17-29 PB4350DAC1 Device: Configuration Parameters of the "Globals" Tab

Hidden Option Fields

If the "Globals" tab is clicked with the right-hand mouse key and the "Show all Options" option is selected in the context menu, other option boxes become visible (see Fig. 17-68).

These option boxes make it possible for the ETAS service team to get additional information on the PB4350DAC1 module when troubleshooting.

Note

These option boxes are not intended for users - the predefined settings should therefore not be changed by the user!

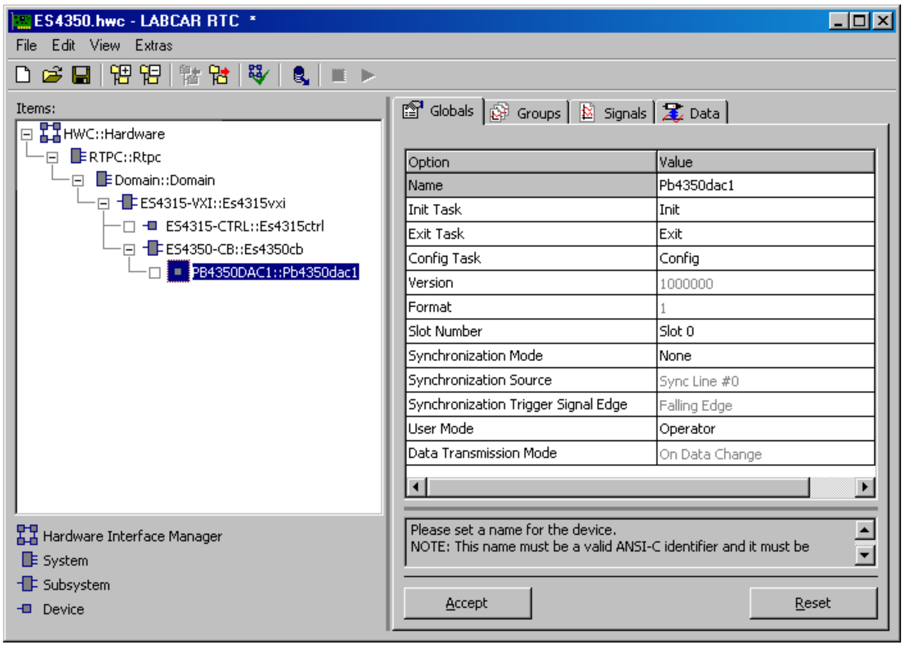


Fig. 17-68 The "Globals" Tab of the PB4350DAC1 Device (with all Option Boxes)

User Mode

Sets the RTIO user mode.

Users should select the option "Operator" in this list box. The user mode "Supervisor" activates additional options for troubleshooting and is only intended for the ETAS service team. When the "Supervisor" mode is activated, the contents of the non-volatile data memory of the PB4350DAC1 module are also displayed in the "Target Debug" window during initialization.

The list box can be edited during runtime and has the predefined setting "Operator".

Data Transmission Mode

This list box can only be edited if the option "Supervisor" is set in the "User Mode" list box. It determines when real-time data is transferred from the experimental target to the PB4350DAC1 module.

If the "On Data Change" option is activated, the data of a signal group is only transmitted if data has actually been changed. If the "Every Interval" option is activated, the real-time data is transmitted at every interval of the allocated real-time task. The list box can be edited during runtime and is preset to "On Data Change".

Tab. 17-30 summarizes the properties of the hidden RTIO parameters of the "Globals" tab.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
User Mode	uint8	Yes	User mode 0: Operator 1: Supervisor "Operator" is preset. "Supervisor" activates additional options for trouble shooting.
Data Transmission Mode	uint8	Yes	Transmission mode of real-time data 0: On Data Change 1: Every Interval "On Data Change" is preset. Option field can only be edited if "User Mode" = "Supervisor"

* Data type which the RTIO driver uses internally for the parameter

Tab. 17-30 PB4350DAC1 Device: Hidden Configuration Parameters of the "Globals" Tab

17.6.3 Groups (PB4350DAC Device)

The PB4350DAC1 device has two signal groups (see Fig. 17-69) which are both transferred by the experimental target to the PB4350DAC1 module.

The "AnaOut" signal group transports the voltage values for the ten analog outputs of a module.

The PB4350DAC1-RTIO driver provides the signal group "CutOff" for error simulation (e.g. to simulate a short in the line of a vehicle sensor to the ECU). This signal group makes it possible to switch the analog outputs of a module between the normal mode of analog voltage output and the error of a high-impedance output.

Real-time operating system tasks must be assigned to the signal groups. A task with periodic activation is normally selected for the "AnaOut" signal group for analog signal output. The activation period depends on the dynamic behavior of the signals to be output. Useful activation periods are limited downwards to 500 µs by runtimes of the firmware on the carrier boards as well as on the

PB4350DAC1 module. Tasks with software activation or periodic tasks with slow activation (larger than 10 ms) are usually selected for the signal group "CutOff" for error simulation.

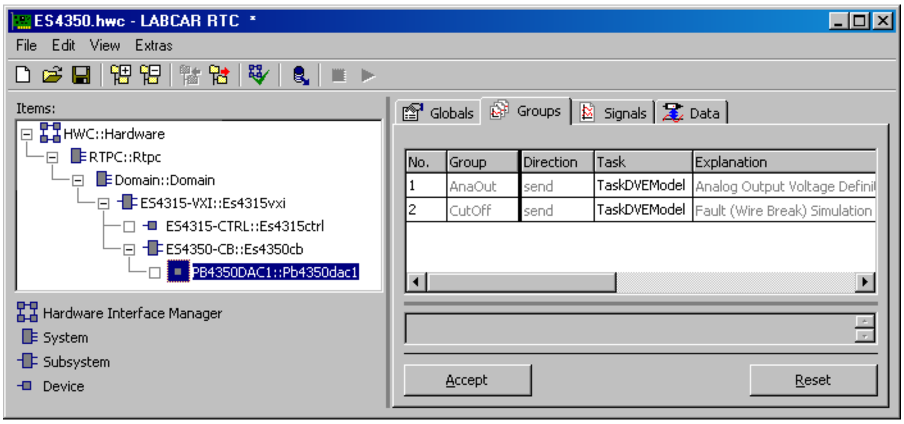


Fig. 17-69 The "Groups" Tab of the PB4350DAC1 Device

17.6.4 RTIO Signals of the "AnaOut" Signal Group

The "AnaOut" signal group consists of ten RTIO signals ("AnaOut_0" to "AnaOut_9") which are of the data type "real32". They define the voltages at the analog outputs of the PB4350DAC1 module.

The value range of the RTIO signals extends from 0.0 V to 10.0 V if the allocated DA output is operated with an internal voltage reference, or from 0.0 to 1.0, if the allocated DA output is operated with an external voltage reference.

Tab. 17-31 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Comment / Value Range
AnaOut_0 to AnaOut_9	real32	Analog output 0 to 9: Internal voltage reference: Output voltage in Volts (0.0 V to 10.0 V) External voltage reference: output voltage in percent of the external reference voltage Value range: 0.0 (corresponds to 0 %) to 1.0 (corresponds to 100 %)
* Data type which the RTIO driver uses internally for the parameter		

Tab. 17-31 PB4350DAC1 Device: The RTIO Signals of the "AnaOut" Signal Group

17.6.5 RTIO Signals of the "CutOff" Signal Group

The "CutOff" signal group consists of ten RTIO signals ("CutOff_0" to "CutOff_9") which are of the data type "bool". They define whether the allocated DA output is operated in the mode for analog voltage output or in the mode for error simulation.

Note

To ensure the switch commands are implemented, the relays for interrupting the output signals should be switched every 50 ms at most.

Tab. 17-32 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Comment / Value Range
CutOff_0 to CutOff_9	bool	Operating mode for analog output 0 to 9: 0: analog voltage output (preset) 1: error simulation
* Data type which the RTIO driver uses internally for the parameter		

Tab. 17-32 PB4350DAC1 Device: The RTIO Signals of the "CutOff" Signal Group

17.6.6 Signals (PB4350DAC Device)

The DA outputs of a PB4350DAC1 module are configured in the "Signals" tab.

Fig. 17-70 shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

Reference

The voltage reference (internal 10 Volt voltage reference or external voltage reference) of a DA output is specified in this list box.

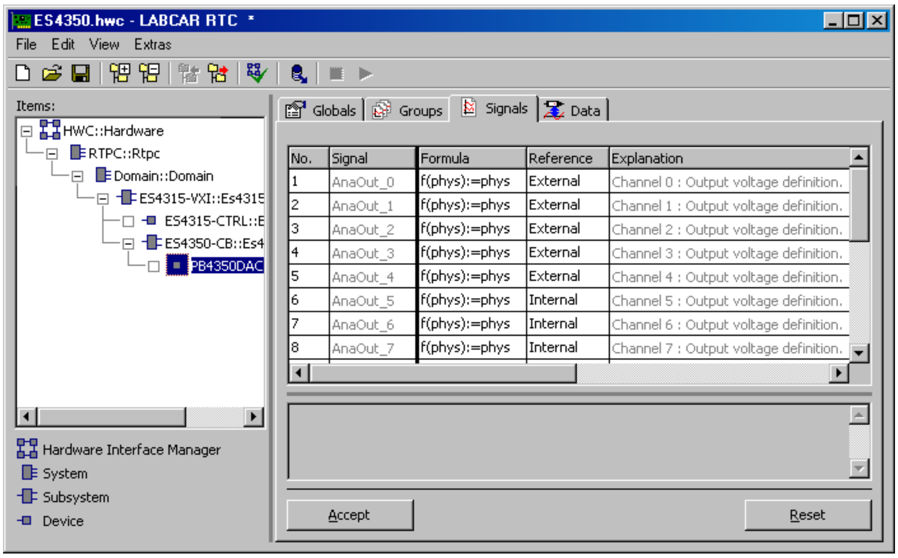


Fig. 17-70 The "Signals" Tab of the PB4350DAC1 Device

Tab. 17-33 summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment / Value Range
Reference	bool	Yes	Voltage reference of the DA output 0: External reference voltage 1: Internal 10 V reference voltage (preset)
* Data type which the RTIO driver uses internally for the parameter			

Tab. 17-33 PB4350DAC1 Device: Configuration Parameters of the "Signals" Tab

18 ES4408 System

The ES4408.1 Load Chassis and the associated boards create a system for simulating electric loads. The various boards make it possible to simulate various loads to test the behavior of ECUs in an HIL system. These particularly include pull-up/pull-down loads (up to 150 mA) and the simulation of electromagnetic injectors and piezo injectors.

The ES4408CON.1 Communication Interface is required for communication with the boards via the Ethernet interface in the ES4408.1 Load Chassis.

The Structure of the ES4408 RTIO Tree

The ES4408.1 Load Chassis is incorporated in the RTIO Editor by selecting the ES4408-Load-Chassis subsystem. This subsystem is allocated to an RTPC experimental target.

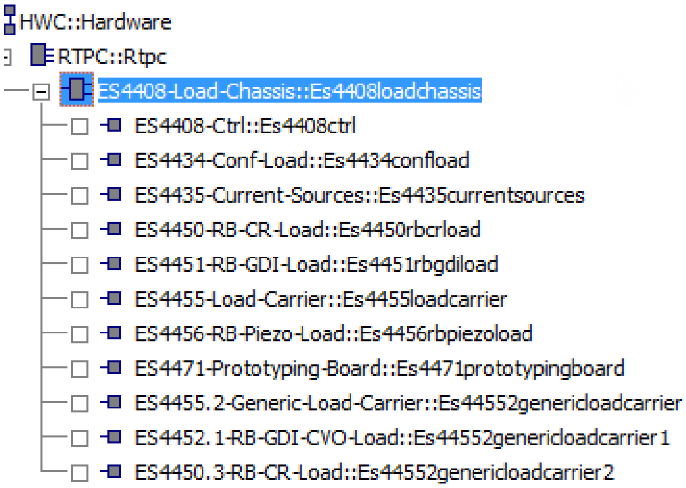


Fig. 18-1 RTIO Hardware Description with Integrated ES4408.1 Load Chassis and other Boards in it

The instances of the integrated boards follow immediately as RTIO devices.

18.1 ES4408-Load-Chassis Subsystem

18.1.1 Globals (ES4408-Load-Chassis Subsystem)

All settings which refer to the ES4408.1 Load Chassis and the boards in it are made in the "Globals" tab.

The following figure shows the "Globals" tab of the ES4408-Load-Chassis subsystem after selecting **View → Show All**.

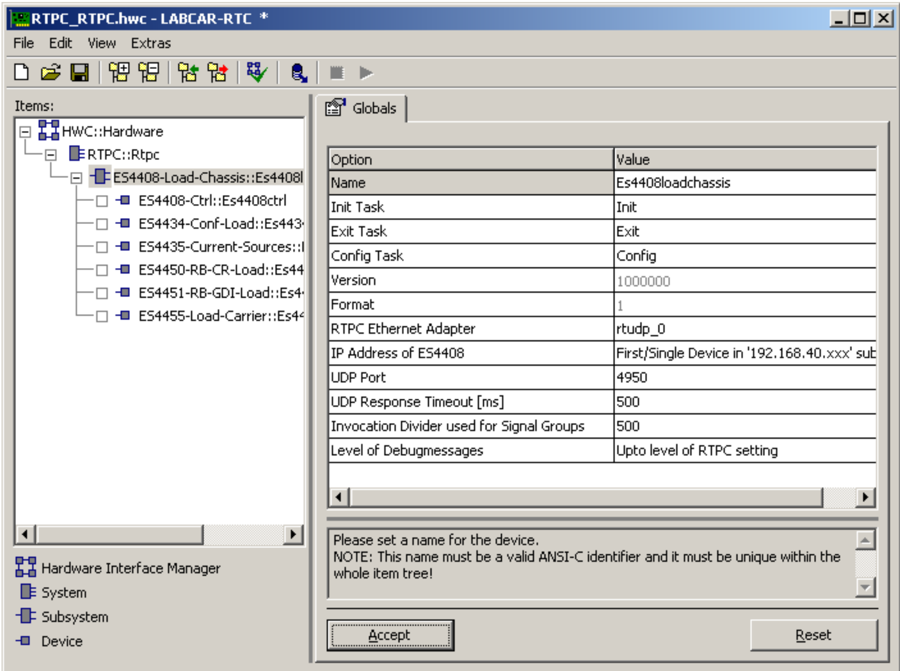


Fig. 18-2 The "Globals" Tab of the ES4408-Load-Chassis Subsystem

The following parameters can be set in this tab.

Note

*The options shown with a * are only visible after selecting **View → Show All**.*

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Init Task, Exit Task, Config Task

In these option fields, you can select the tasks in which the generated Init, Exit and Config processes are to be included.

RTPC Ethernet Adapter

This is where the logical Ethernet adapter of the real-time PC is selected which is to be used for communicating with the ES4408.1 Load Chassis via rtudp (Real-Time UDP). For an ES4408.1 Load Chassis a "nonrtudp" port is also possible.

IP Address of ES4408

This is where you assign an IP address to the ES4408.1 Load Chassis from a range between 192.168.40.50 and 192.168.40.70.

In addition, the first or second device found in the network is easy to specify (determined by a broadcast in the subnetwork).

Note

*The options shown with a * are only displayed after selecting **View** → **Show All**.*

*UDP Port**

The UDP port of the Ethernet adapter to be used for communication (default 4950).

*UDP Response Timeout [ms]**

Timeout for a UDP response package from the ES4408.1 Load Chassis (default 500).

*Invocation Divider used for Signal Groups**

The "Invocation Divider..." setting makes it possible to keep the number of queries at the ES4408.1 Load Chassis to a manageable level.

The processes of a signal group are normally included in one single fast simulation task; the queries at the ES4408.1, however, typically refer to status information and do not have to be dealt with under real-time conditions.

A reasonable time span when querying such information is 1 - 5 seconds. UDP queries in signal groups are implemented as "non-blocking" and are thus also processed over several task calls.

*Level of Debug Messages**

The "Level of Debug Messages" option makes it possible to set the level of error messages which are independent from that of the real-time PC or the rest of the system.

"SCPI Debugging" can also be enabled – in this operating mode, every SCPI command returns a response, regardless of whether an error has occurred or not.

The possible settings are explained in the following table:

Setting	Meaning
Upto Level of RTPC setting	The level set in LABCAR-RTPC is used
Upto LOG_ERR	Only emergencies, criticals, alerts and errors are logged
Upto LOG_WARNING	Like LOG_ERROR plus warnings
Upto LOG_NOTICE	Like LOG_WARNING plus notices
Upto LOG_INFO	Like LOG_NOTICE plus information
Upto LOG_DEBUG	All available information is recorded
…+ SCPI DEB ON	Can also be selected in addition to the above options: "SCPI Debugging" enabled

Tab. 18-1 Possible Settings for "Level of Debug Messages"

The following table summarizes the properties of the individual parameters.

Note

*The options shown with a * are only visible after selecting **View** → **Show All**.*

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	String	No	Name of subsystem
Init Task, Exit Task, Config Task	Selection	No	Tasks for the Init, Exit and Config processes
RTPC Ethernet Adapter	int	No	The Ethernet adapter of the real-time PC for communicating with the ES4408.1 Load Chassis. Value range: rtudp_0...rtudp_3, nonr-tudp
IP Address of ES4408	int	No	The IP address assigned to the ES4408.1 Load Chassis Value range: 'First/Single Device' |'Second Device'| 192.168.40.x , x = 50...70
UDP Port*	int	No	The UDP port of the Ethernet adapter (default: 4950) Value range: 1000…65535
UDP Response Timeout [ms]*	int	No	Timeout for communication with the ES4408.1 Load Chassis (default: 500) Value range: 10…10000 ms
Invocation Divider used for Signal Groups*	int	No	Communication of a signal group only takes place with every n-th call of the task (default: 500) Value range: 10…10000
Level of Debugmessages*	int	Yes	Settings for the scope of debug messages (see Tab. 18-1 on page 576)

Tab. 18-2 ES4408-Load-Chassis Subsystem: Configuration Parameters of the "Globals" Tab

18.2 ES4408-Ctrl Device

The ES4408-Ctrl device is used to set globally valid parameters of the ES4408CON.1 Communication Interface.

18.2.1 Globals (ES4408-Ctrl Device)

The following figure shows the "Globals" tab of the ES4408-Ctrl device.

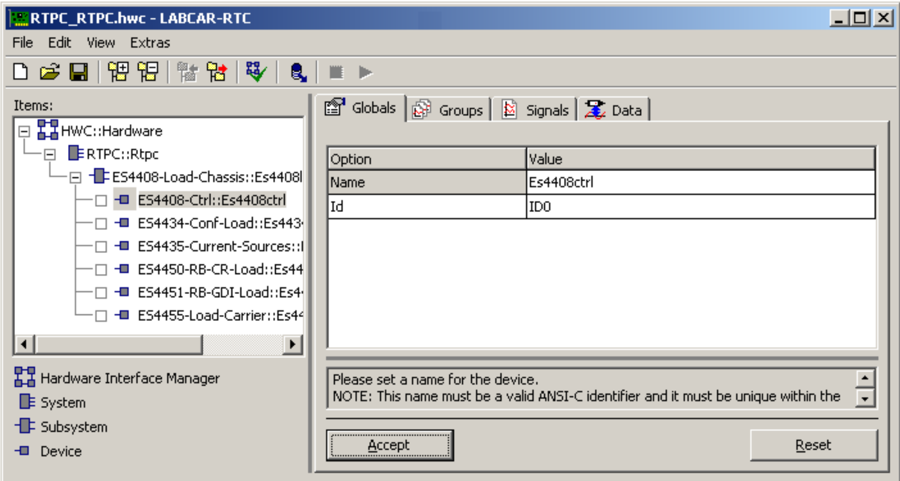


Fig. 18-3 The "Globals" Tab of the ES4408-Ctrl Device

This tab contains the following parameters:

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Id

Identifier of the ES4408CON.1 Communication Interface

18.2.2 Data (ES4408-Ctrl Device)

In the "Data" tab of the ES4408-Ctrl device, status information on the supply voltages, temperature and battery voltage are output.

The following figure shows the "Data" tab of the ES4408-Ctrl device.

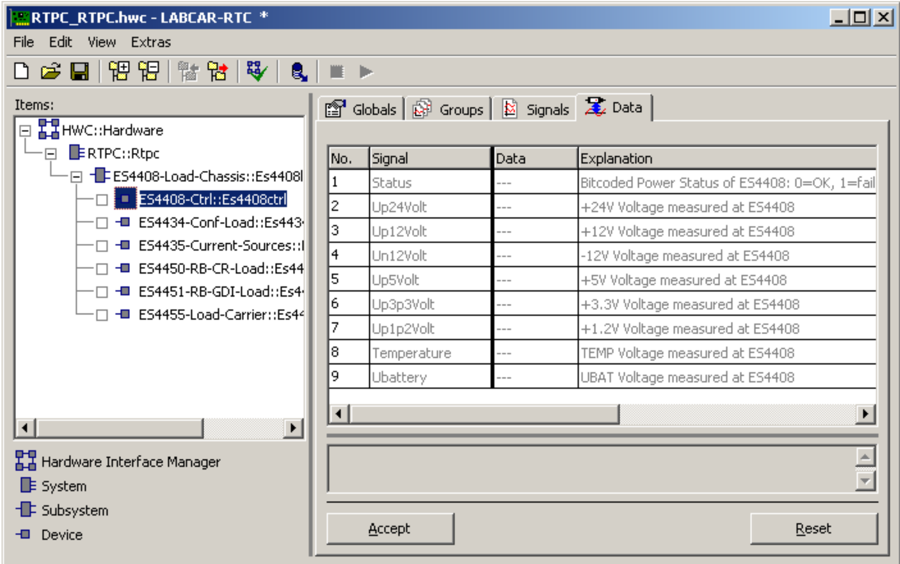


Fig. 18-4 The "Data" Tab of the ES4408-Ctrl Device

The following signal data is displayed.

Status

The general status of the power supply of the ES4408.1 Load Chassis.

A set bit 0, 1, 2, 3 or 4 means the failure of the voltages +12 V, -12 V, +5 V, +3.3 V or +1.2 V respectively.

Up24V, Up12V, Un12V, Up5V, Up3p3V, Up1p2V

This is where the supply voltage values measured during operation are shown.

Temperature

The value measured by the temperature sensor in the ES4408.1 Load Chassis.

Ubattery

Current value of the battery voltage supplied to the ES4408.1 Load Chassis.

The following table summarizes the properties of the RTIO signals.

RTIO Signal	Data Type	Comment
Status	uint8	If bit 0, 1, 2, 3 or 4 is set, this signifies an error in the corresponding supply voltages +12 V, -12 V, +5 V, +3.3 V or +1.2 V
Up24V	real32	Measure value for +24 V supply voltage
Up12V	real32	Measure value for +12 V supply voltage
Un12V	real32	Measure value for -12 V supply voltage
Up5V	real32	Measure value for +5 V supply voltage
Up3p3V	real32	Measure value for +3.3 V supply voltage
Up1p2V	real32	Measure value for +1.2 V supply voltage
Temperature	real32	Measure value for voltage at temperature sensor
Ubattery	real32	Measure value for battery voltage
Module Present	uint8	= 0: The board assigned to this RTIO item is not available = 1: The board assigned to this RTIO item is available and active

Fig. 18-5 ES4408-Ctrl Device: Signals of the "Data" Tab

18.3 ES4434-Conf-Load Device

The ES4434.1 Configurable Load Board is configured on the board so that no settings are necessary in the software.

18.3.1 Globals (ES4434-Conf-Load Device)

The following figure shows the "Globals" tab of the ES4434-Conf-Load device.

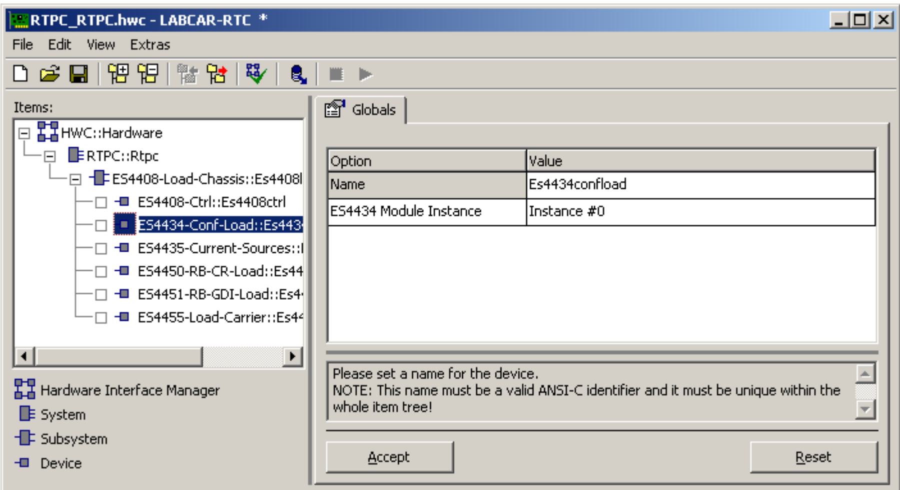


Fig. 18-6 The "Globals" Tab of the ES4434-Conf-Load Device

This tab contains the following parameters:

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

ES4434 Module Instance

The internal names of this instance of an ES4434.1 Configurable Load Board (Instance #0 ... Instance #6).

18.4 ES4435-Current-Sources Device

Unlike the ES4434.1 Configurable Load Board, the ES4435.1 Current Source Load Board is configured and controlled virtually entirely by the LABCAR-RTC.

18.4.1 Globals (ES4435-Current-Sources Device)

The following figure shows the "Globals" tab of the ES4435-Current-Sources device.

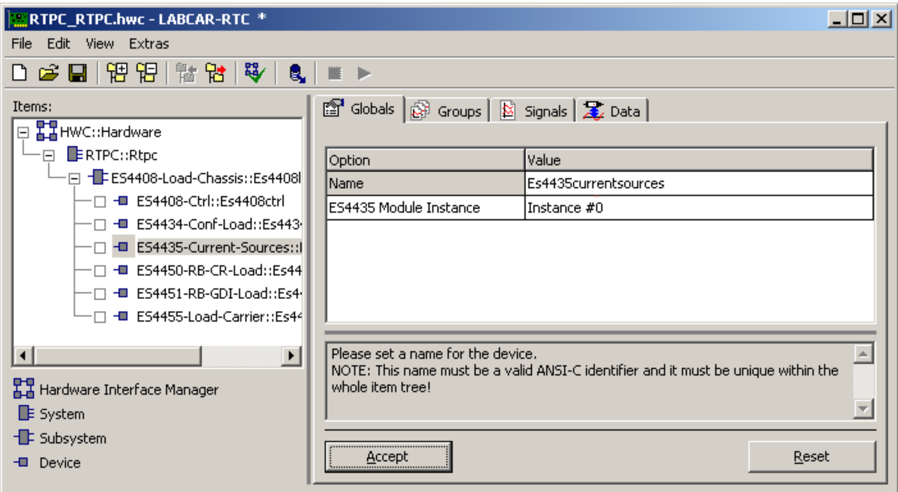


Fig. 18-7 The "Globals" Tab of the ES4435-Current-Sources Device

The following parameters can be set in this tab:

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

ES4435 Module Instance

The internal names of this instance of an ES4435.1 Current Source Load Board (Instance #0 ... Instance #6).

18.4.2 Signals (ES4435-Current-Sources Device)

The following figure shows the "Signals" tab of the ES4435-Current-Sources device.

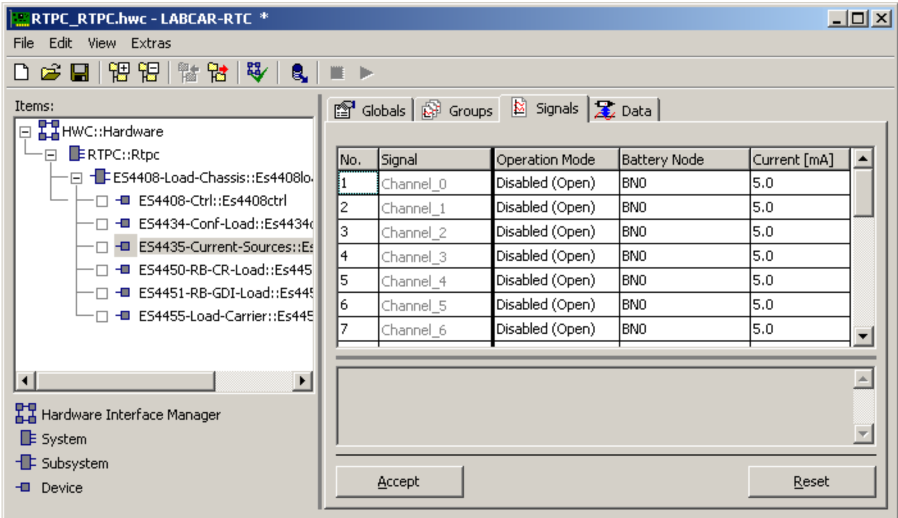


Fig. 18-8 The "Signals" Tab of the ES4435-Current-Sources Device

The following parameters can be set for each of the 24 channels in this tab:

Operation Mode

This is where the current sources can be configured for pull-up or pull-down operation or disabled (i.e. configured neither against a battery node nor a battery ground).

Battery Node

This is where the battery voltage can be selected to which the current source is to be connected. Select from the battery nodes 0..5, permanent plus ("PERM") and no voltage ("NONE").

Current

This is where the current is set which the current source should supply (5..150 mA).

18.4.3 Data (ES4435-Current-Sources Device)

Status information on every channel is output in the "Data" tab of the ES4435-Current-Sources device.

The following figure shows the "Data" tab of the ES4435-Current-Sources device.

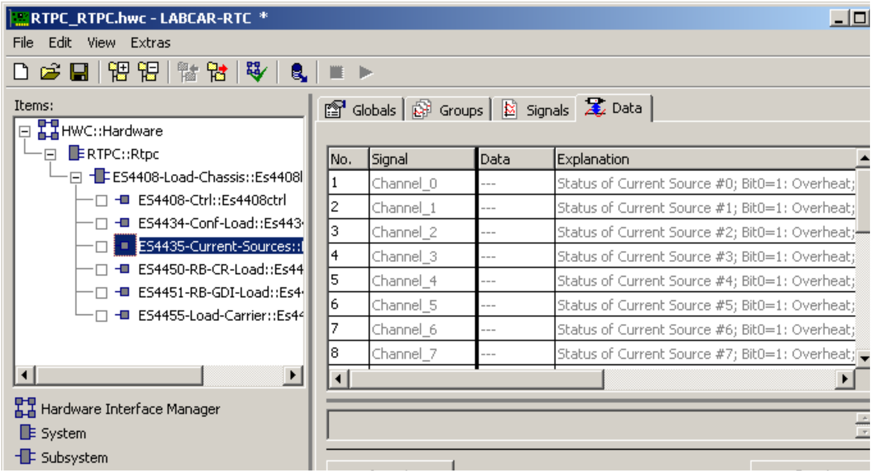


Fig. 18-9 The "Data" Tab of the ES4435-Current-Sources Device

The following table contains information on the meaning of the status data:

RTIO Signal	Data Type	Comment
Channel_xx [xx= 0…23]	uint8	Bit 0 = 1: Overtemperature Bit 1 = 1: Power over 3 W (current is reduced)
Module Present	uint8	= 0: The board assigned to this RTIO item is not available = 1: The board assigned to this RTIO item is available and active

Tab. 18-3 ES4435-Current-Sources Device: Signals of the "Data" Tab

18.5 ES4450-RB-CR-Load Device

The ES4450-RB-CR-Load device represents an ES4450.2 Load Board for 4 RB CRS Injectors installed in the ES4408.1 Load Chassis.

Note

The RTIO configuration of an ES4451.3 Load Board for 4 RB GDI Injectors or an ES4455.1 Load Board is identical to the configuration of the ES4450.2 Load Board for 4 RB CRS Injectors described here and is thus not described separately.

18.5.1 Globals (ES4450-RB-CR-Load Device)

The following figure shows the "Globals" tab of the ES4450-RB-CR-Load device.

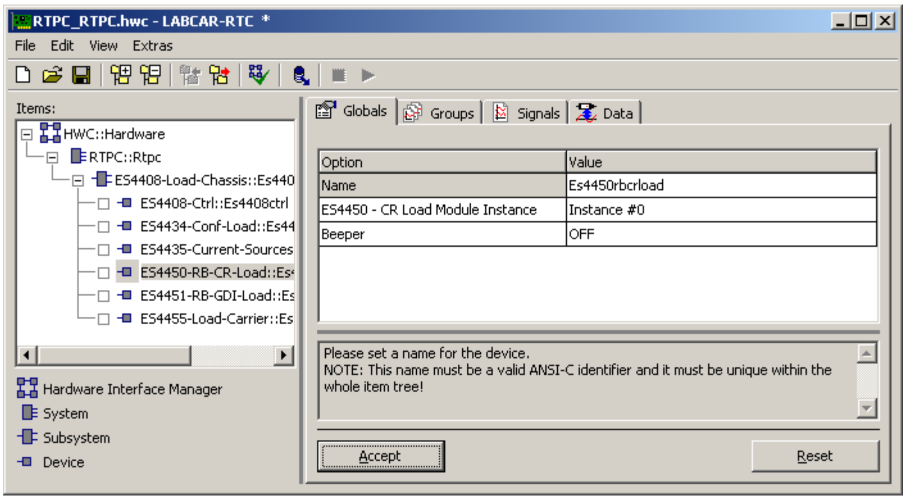


Fig. 18-10 The "Globals" Tab of the ES4450-RB-CR-Load Device

This tab contains the following parameters:

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

ES4450 - CR Load Module Instance

The internal names of this instance of an ES4450.2 Load Board for 4 RB CRS Injectors (Instance #0...Instance #2).

Beeper

Use this option to enable/disable the function for the acoustic signal (when exceeding thresholds which can be set in the "Signals" tab).

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Beeper	uint8	Yes	0 = Off, 1 = On

Tab. 18-4 ES4450-RB-CR-Load Device: Configuration Parameter of the "Globals" Tab

18.5.2 Signals (ES4450-RB-CR-Load Device)

The thresholds for the LED display on the front panel are set in the "Signals" tab of the ES4450-RB-CR-Load device.

The following figure shows the "Signals" tab of the ES4450-RB-CR-Load device.

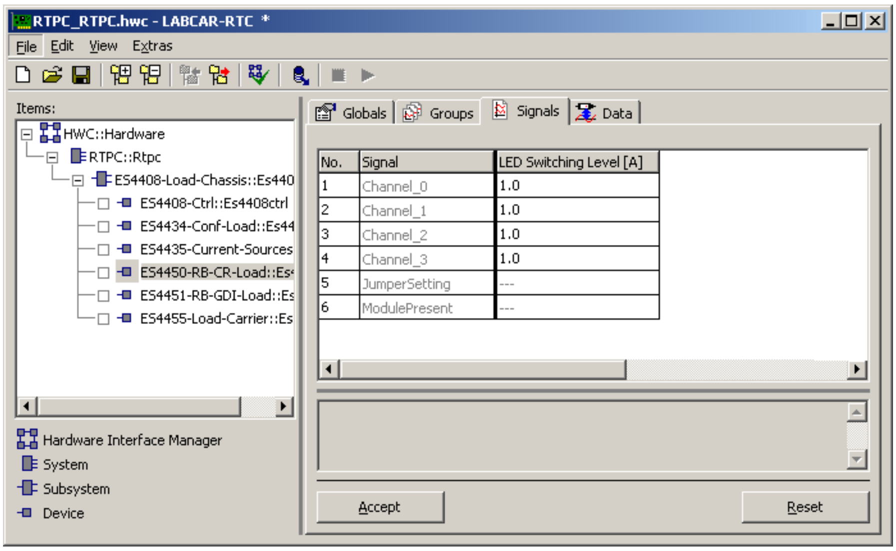


Fig. 18-11 The "Signals" Tab of the ES4450-RB-CR-Load Device

Channel_n (n = 0...3)

This is where you can set the current value. If this is exceeded, the corresponding LED on the front panel lights up.

18.5.3 Data (ES4450-RB-CR-Load Device)

Status information on every channel is output in the "Data" tab of the ES4450-RB-CR-Load device.

The following figure shows the "Data" tab of the ES4450-RB-CR-Load device.

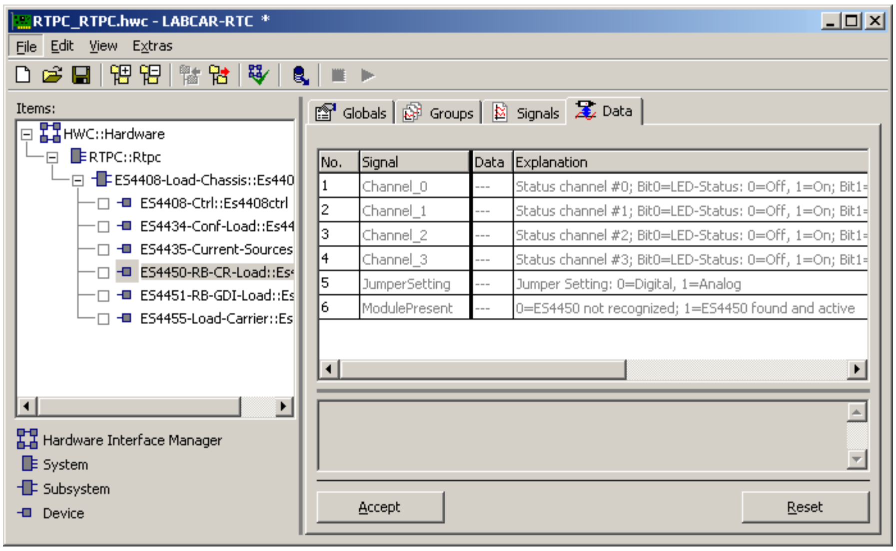


Fig. 18-12 The "Data" Tab of the ES4450-RB-CR-Load Device

The following table contains information on the meaning of the status data:

RTIO Signal	Data Type	Comment
Channel_x [x = 0…3]	uint8	Bit 0 = LED status: 0 = LED on, 1 = LED off Bit 1 = Hysteresis status: 0 = low, 1 = high)
Jumper Setting	uint8	Position of the jumper strip JP200: 0 = Current is output digitally 1 = Current is output analog
Module Present	uint8	= 0: The board assigned to this RTIO item is not available = 1: The board assigned to this RTIO item is available and active

Tab. 18-5 ES4450-RB-CR-Load Device: Signals of the "Data" Tab

18.6 ES4456-RB-Piezo-Load Device

The ES4456-RB-Piezo-Load device represents an ES4456.1 Load Board for 8 RB Piezo Injectors.

18.6.1 Globals (ES4456-RB-Piezo-Load Device)

The following figure shows the "Globals" tab of the ES4456-RB-Piezo-Load device.

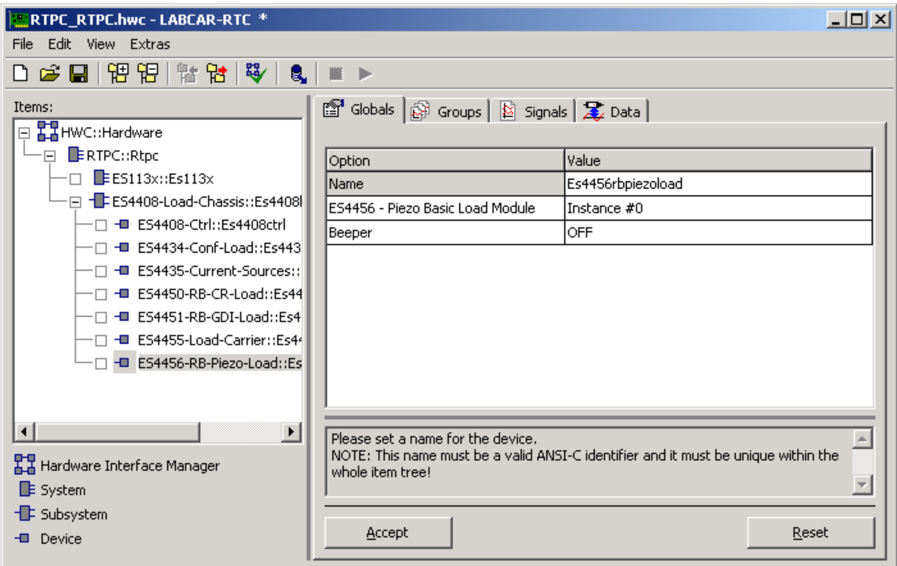


Fig. 18-13 The "Globals" Tab of the ES4456-RB-Piezo-Load Device

This tab contains the following parameters:

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

ES4456 - Piezo Basic Load Module

The internal names of this instance of an ES4456.1 Load Board for 8 RB Piezo Injectors (Instance #0...Instance #2).

Beeper

Use this option to enable/disable the function for the acoustic signal (when exceeding thresholds which can be set in the "Signals" tab).

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Beeper	uint8	Yes	0 = Off, 1 = On

Tab. 18-6 ES4456-RB-Piezo-Load Device: Configuration Parameter of the "Globals" Tab

18.6.2 Signals (ES4456-RB-Piezo-Load Device)

The thresholds for the LED display on the front panel are set in the "Signals" tab of the ES4456-RB-Piezo-Load device.

The following figure shows the "Signals" tab of the ES4456-RB-Piezo-Load device.

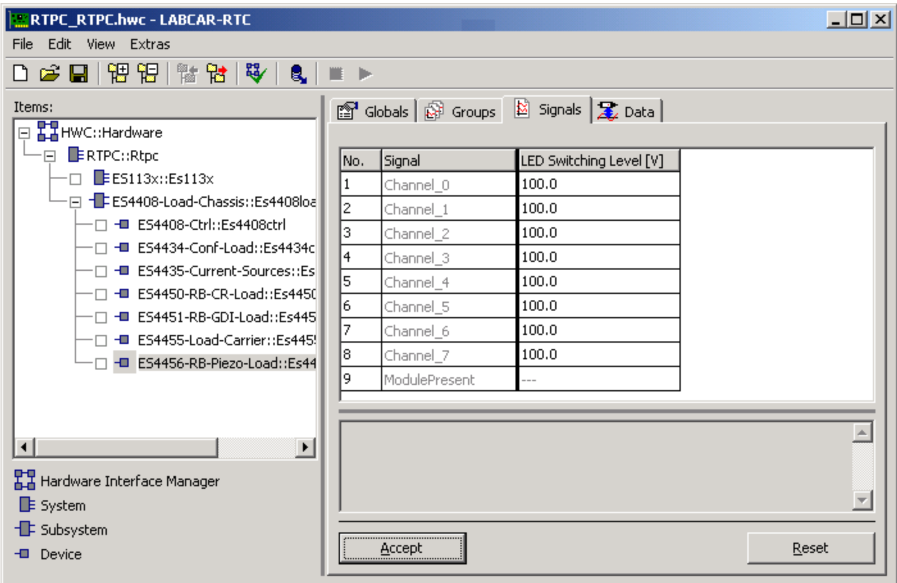


Fig. 18-14 The "Signals" Tab of the ES4456-RB-Piezo-Load Device

Channel_n (n = 0...7)

This is where you can set the voltage value. If this is exceeded, the corresponding LED on the front panel lights up (0...250 V).

18.6.3 Data (ES4456-RB-Piezo-Load Device)

Status information on every channel is output in the "Data" tab of the ES4456-RB-Piezo-Load device.

The following figure shows the "Data" tab of the ES4456-RB-Piezo-Load device.

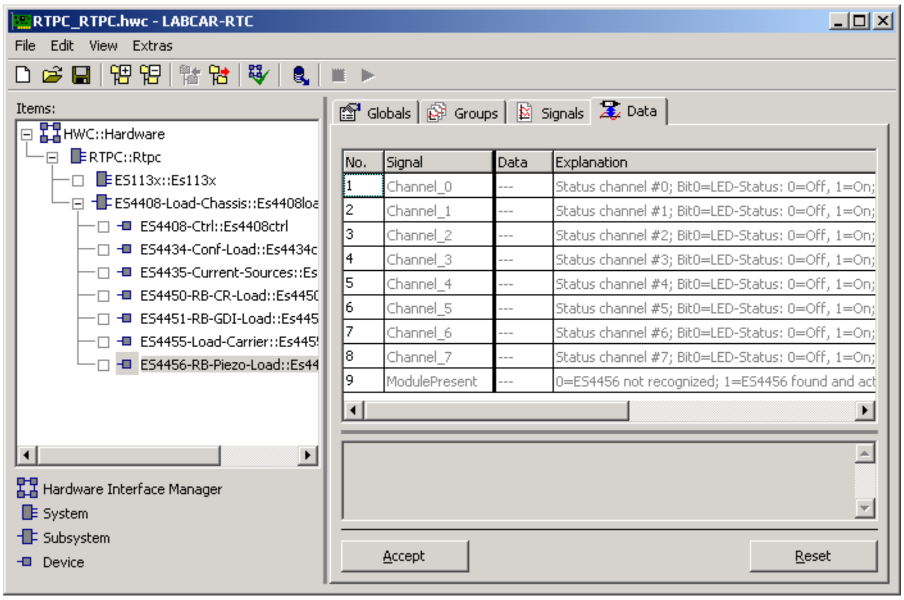


Fig. 18-15 The "Data" Tab of the ES4456-RB-Piezo-Load Device

The following table contains information on the meaning of the status data:

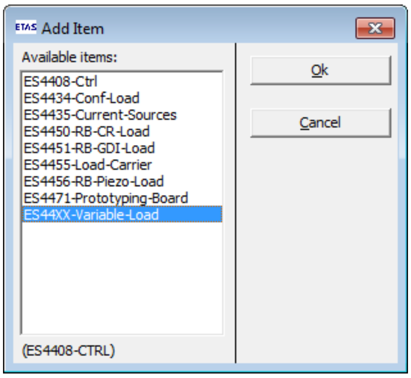
RTIO Signal	Data Type	Comment
Channel_x [x = 0…7]	uint8	Bit 0 = LED status: 0 = LED on, 1 = LED off Bit 1 = overload detection: 0 = no, 1 = yes)
Module Present	uint8	= 0: The board assigned to this RTIO item is not available = 1: The board assigned to this RTIO item is available and active

Tab. 18-7 ES4456-RB-Piezo-Load Device: Signals of the "Data" Tab

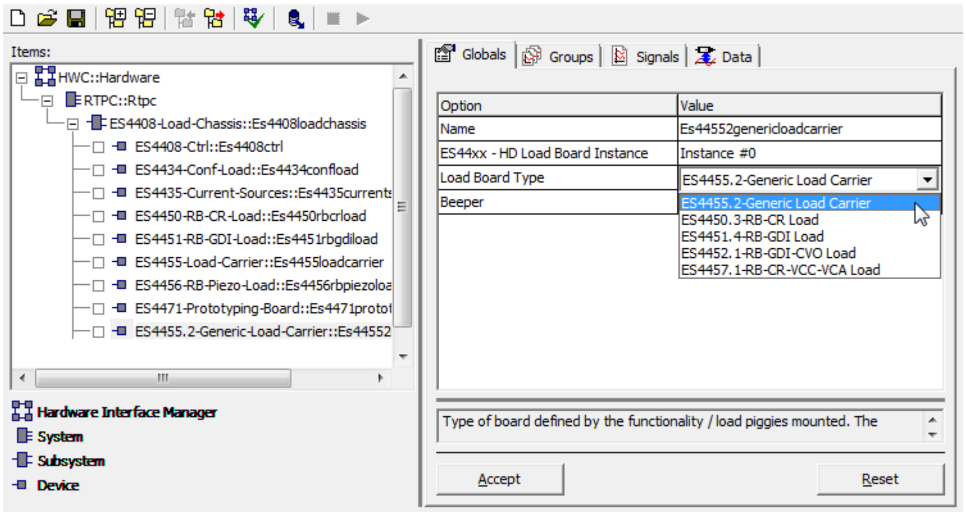
18.7 ES44XX-Variable-Load

The ES44XX Variable-Load boards are represented with a single, generic RTIO item. You can switch between the different board types later using the parameter "Load Board Type" in the "Globals" tab (see "Load Board Type" on page 591).

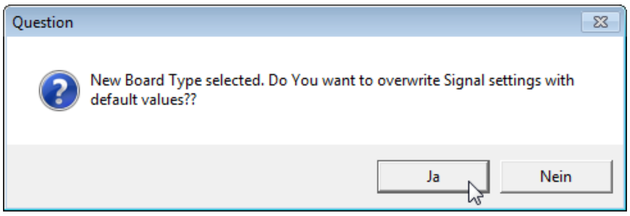
- To add a Variable-Load board to the RTIO tree, select an ES4408-Load-Chassis item and select **Add Item** from the context menu.
- You can now select "ES44XX-Variable-Load" and add the item. When it is added, its display name changes to "ES4455.2-Generic-Load-Carrier" - this is the generic board variant of the variable load.



- To select a different board variant, do so by selecting "Load Board Type".



- Specify whether you want to keep the parameter set in the "Signals" tab or whether you want to overwrite it with the board-specific default values.



- To make these changes, click **Accept**.

18.7.1 Globals (ES44XX-Variable-Load)

The "Globals" tab is identical for all boards of the type "Variable Load".

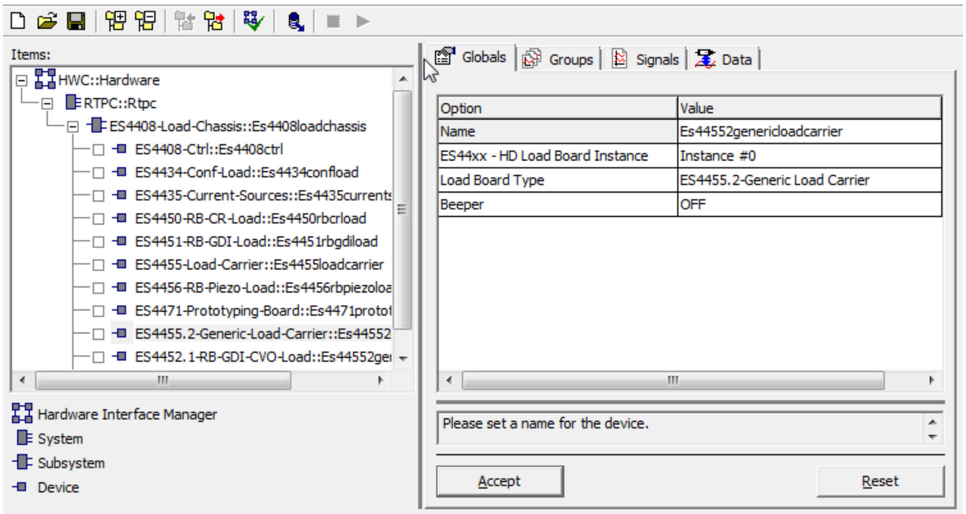


Fig. 18-16 The "Globals" Tab of the ES44XX-Variable-Load Device

Name

The C-Name of the item – this is used as a prefix for all global identifiers. It may be useful to change it accordingly for better clarity (presetting: "Es44552genericloadcarrier").

ES44xx – HD Load Board Instance

The board instance (Instance #0 … Instance #2). A maximum of three "Variable Load" boards of whatever type can be added.

Note

The instances are not verified during configuration, i.e. several boards of the same type can be configured with the same instance. But verification does take place when the consistency is checked (when the configuration is saved). Please note that boards of different generations but with the same name (e.g. ES4455.1 and ES4455.2) are numbered together.

Load Board Type

The type of load board – an "SCPI Board Name" is derived from this and used for communication with the ES4408_CON. The board-specific functionalities are encapsulated in the ES4408_CON firmware and in the HDC of the load board.

Display Name	SCPI Board Name	Function
ES4455.2-Generic Load Carrier	ES4455_2	Generic board carrier without specific loads
ES4450.3-RB-CR Load	ES4450_3	Diesel loads
ES4451.4-RB-GDI Load	ES4451_4	Gasoline loads
ES4452.1-RB-GDI-CVO Load	ES4452	Gasoline loads with CVO
ES4457.1-RB-CR-VCC-VCA Load	ES4457	Diesel loads with VCC/VCA

Tab. 18-8 Possible Settings for "Load Board Type"

Beeper

Enabling/disabling the signal generator of the loads.

Note

A signal is only emitted if at least one of the loads is addressed.

Furthermore, there are hidden setting parameters which can be made visible with the context menu item **Show All**. These are used as control values for the SCPI communication of the "Control" signal group and should not normally be changed.

18.7.2 Groups (ES44XX-Variable-Load)

Four signal groups are made available, three of which are "Receive" signal groups.

Name of Signal Group	Direction	Function
Status	Receive	Level of the "I_Dig_Out" signals and status of the load and the switches
Control	Send	Real-time control signals of the loads
Measure	Receive	Current values of load current and voltage
Presence	Receive	Presence/Driver Ready status

Tab. 18-9 Signal Groups of the ES44XX-Variable-Load Device

Together with all other "Receive" signal groups of the ES4408 subsystem, the "Receive" signal groups are supplied sequentially in accordance with the settings in the "Globals" tab.

The "Send" signal group, on the other hand, is processed asynchronously to the "Receive" signal groups, i.e. SCPI command sequences can be communicated to an ES4408_COM at greater speed.

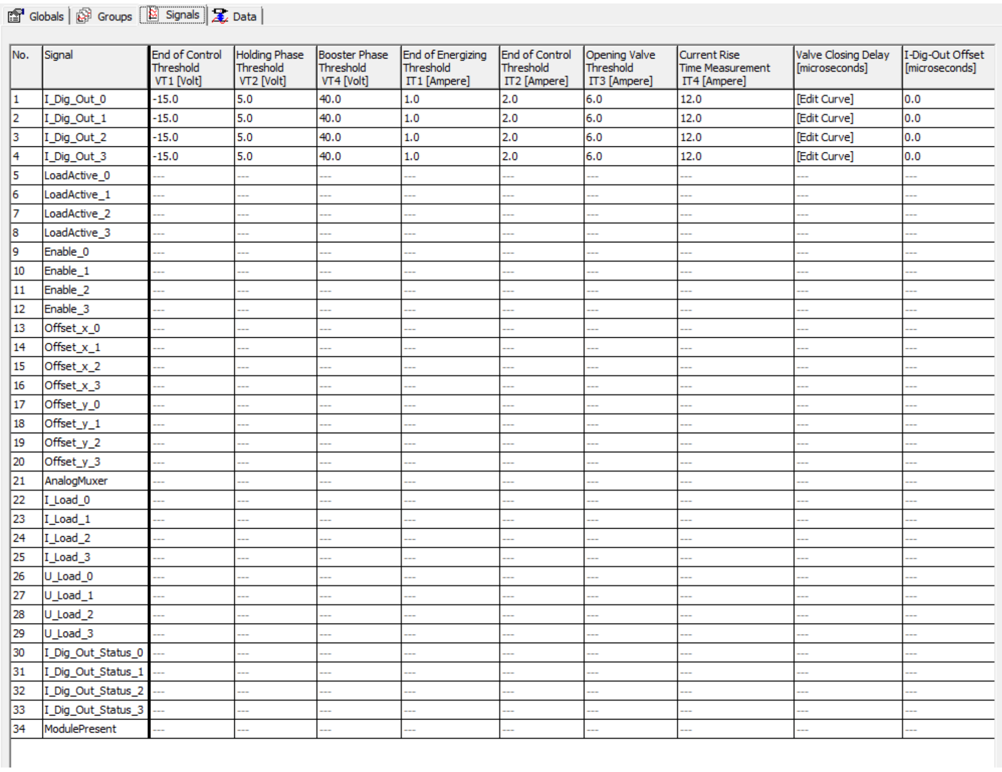
However, there can be no response from the SCPI command sequence of the "Control" signal group as this can influence the assignment of the SCPI responses to the "Receive" signal groups. This is why the sending of SCPI sequences of the "Control" signal group is prevented with debug level "SCPI DEBUG ON".

Generally, an SCPI sequence is only generated when there is a change to the input signals it is based on, i.e. there is no SCPI communication if there is no change. The maximum frequency of SCPI communication of the "Control" signal group is limited by a hold-back time which prevents the SCPI channel from being overloaded. The value is specified in the "Globals" tab by the "Control Guard Time" parameter which is not normally visible.

18.7.3 Signals (ES44XX-Variable-Load)

In the "Signals" tab, the settings of the four load channels can be made using the signals "I_Dig_Out_0"..."I_Dig_Out_3".

An item is defined by a group of setting parameters and a waveform – depending on the type of load board selected, only the setting options required for this load board type are visible.


Globals | Groups | Signals | Data

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	---	---	---	---	---	---	---	---	---

Fig. 18-17 The "Signals" Tab of the ES44XX-Variable-Load Device

Parameter	ES4455.2	ES4450.3	ES4451.4	ES4452	ES4457
VT1 - End of Control Threshold [±128.0 V]	Visible -15.0	Visible -15.0	Visible -15.0	Visible -15.0	Visible -15.0
VT2 - Holding Phase Threshold [±128.0 V]	Visible 5.0	Visible 5.0	Visible 5.0	Visible 5.0	Visible 5.0
VT3 - Reserved [±128 V]	Invisible	Invisible	Invisible	Invisible	Invisible
VT4 - Booster Phase Threshold [±128.0 V]	Visible 40.0	Visible 40.0	Visible 40.0	Visible 40.0	Visible 40.0
IT1 - End of Energizing Threshold [±32.0 A]	Invisible 1.0	Invisible 1.0	Invisible 1.0	Visible 1.0	Visible 1.0
IT2 - End of Control Threshold [±32.0 A]	Visible 2.0	Visible 2.0	Visible 2.0	Visible 2.0	Visible 2.0
IT3 - Opening Valve Threshold [±32.0 A]	Visible 6.0	Visible 6.0	Visible 6.0	Visible 6.0	Visible 6.0
IT4 - Current Rise Time measurement [±32.0 A]	Visible 40.0	Visible 40.0	Visible 40.0	Visible 40.0	Visible 40.0
Valve Closing Delay t_close = f(_tcontrol) [µs]	Invisible Default table	Invisible Default table	Invisible Default table	Visible Default table	Visible Default table
I-Dig-Out-Offset [µs]	Visible 0.0	Visible 0.0	Visible 0.0	Visible 0.0	Visible 0.0

Tab. 18-10 The Parameters of the "Signals" Tab, their Visibility and Default Values

The significance of the thresholds VT1...4 and IT1...IT4 is described in section 18.8.1 on page 597.

Valve Closing Delay [microseconds]

This setting is only of any significance for CVO.

In the FPGA of the load board, an additional correction value is calculated for the injection time. The input value is the trigger time of the valve in µs, the output value is the correction value of what is being injected that both lengthens the "I_Dig_Out" output signal and delays the generation of the CVO dip.

The table describes up to 32 points on this curve that is loaded into the ES4408 during configuration time.

I-Dig-Out-Offset [microseconds]

This setting option delays the opening edge of the injection signal, i.e. the injection time is reduced. The parameter can be used as a correction value. Unlike the correction of what is being injected, only a static correction value can be set here.

18.7.4 Data (ES44XX-Variable-Load Device)

In the "Data" tab, level and status information of the receive signal groups is shown and input values for the "Control" signal group are entered.

Globals | Groups | Signals | Data

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 microsecs]
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 microsecs]
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 microsecs]
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 microsecs]
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 microsecs]
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 microsecs]
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 microsecs]
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 microsecs]
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]

Fig. 18-18 The "Data" Tab with Displayed Signal Group

I_Dig_Out_0…I_Dig_Out_3

These signals deliver the level of the "I_Dig_Out" signal of the relevant channel.

Value range: 0 = Low, 1 = High

LoadActive_0… LoadActive_3

With these control signals, the load of the relevant channel can be disconnected via a relay.

Value range: 0 = Load disconnected, 1 = Load connected (default)

Enable_0…Enable_3

With these control signals, specific functions can be controlled with the load board types ES4452.1 (CVO) and ES4457.1 (VCC/VCA).

- In the case of the CVO load board type, the simulation of the discontinuity of the voltage signal when the valve is closed (dip) can be enabled/disabled with these signals.

  Value range: 0 = No dip, 1 = Dip is generated
- In the case of the VCC/VCA load board type, these signals can be used to toggle between legacy operation (injector CRI1.16), VCC/VCA mode and VCC/VCA mode with a delay.

  Value range:
  – 0 = Legacy
  – 1 = VCC/VCA on
  – 2 = VCC/VCA with additional delay

Offset_x_0…Offset_x_3, Offset_y_0…Offset_y_3

These signals are used to influence the time when a dip occurs with a CVO load board type.

The "Offset_x" signals are added to the trigger duration before interpolation in the "Valve Closing Delay" table. The "Offset_y" values, however, are added to the resulting y value of the table. The resulting time value is the delay time after trigger end for generating the dip.

All offset signals are time values with a value range of ±500 µs – values outside this range are limited accordingly.

AnalogMuxer

This signal controls which of the load currents or load voltages is switched to the analog output. The relevant value range is scaled to the ±10 V range of the analog output.

Currents: -32 A…+32 A → -10 V…+10 V

Voltages: -128 V…+128 V → -10 V…+10 V

I_Load_0…I_Load_3, U_Load_0…U_Load_3

These signals deliver the voltage or current values just measured at the or through the loads respectively. The measurements are not synchronized.

Currents have a max. value range of ±32 A, voltages of ±128 V.

I_Dig_Out_Status_0… I_Dig_Out_Status_3

These signals make status and error information of the load channels available. The signals are bit-coded, whereby Bit = 0 means "no error" and Bit = 1 "error".

- Bit 0

  Error signal of the relevant "I_Dig_Out" output, e.g. in the case of a short of this output. The error bit is pending until the value expected is read at the output.
- Bits 1, 2, 3

  Overload signals of switches 1, 2, 3. These error bits are derived from the overcurrent detection of the FETs. The error bits are not reset until the experiment is paused or re-initialized.
- Bit 4

  Disabling of the channel because of too great a power loss. The temperature is balanced for every channel due to the power loss in the HDC. If the value is too high, the load is disconnected until an uncritical value is attained.

ModulePresent

This signal indicates whether the selected load board instance is available and usable.

Value range:

- -1 = No data available
- 0 = No usable board
- 1 = Board available and usable

18.8 Injector Simulation with ES4452.1 (Gasoline) and ES4457.1 (Diesel)

The products ES4452.1 (gasoline) and ES4457.1 (diesel) make it possible to simulate "intelligent/active" injectors or new control algorithms for highly precise fuel dosing in ECUs.

The products ES4451.4 (gasoline) and ES4450.3 (diesel) are of a passive nature and simulate the electronic load of an injector. The predecessor of the ES4451.4 is the ES4451.3 (based on the ES4455.1 Load Board), that of the ES4450.3 is the ES4450.2 (based on the ES4455.1 Load Board).

18.8.1 HDEV5 Injector Simulation (CVO) with ES4452.1

The ES4452.1 simulates HDEV5 injectors and enables the stimulation of CVO control algorithms. CVO stands for "Controlled Valve Operation" and uses physical effects or features in electric signals in injector control for the precise determination of the actual injector closing time.

An "I_Dig_Out" signal is low-active and normally represents the valve opening time. The "I_Dig_Out" signal of the relevant injector simulation can be added to an angle-synchronous time measuring of a HIL system (e.g. ES5340.2-ICE) to communicate the actual injection time to a combustion engine model.

The occurrence of the falling edge of the "I_Dig_Out" signal can either be caused by changing the current/voltage threshold (see Tab. 18-11 on page 601) or via the parameter "I_Dig_Out_Offset".

Current and Voltage Thresholds

To obtain the required function, various current and voltage thresholds have to be configured. These thresholds are described below using typical current/voltage curves for CVO and VCC as is their influence on the generation of the "I_Dig_Out" signals.

Injector control can roughly be divided into four phases (see Fig. 18-19 on page 598):

1. Booster Phase
2. Pickup Phase
3. Holding Phase
4. Switching-off Phase

These four phases are determined in various internal combinations via threshold setting or triggering.

Timetable

It is important to configure a timetable in LABCAR-RTC for the CVO function of the simulated injector. The timetable should at least represent the dependence of the occurrence of the feature (tab) on the injector control duration (ti).

The injector control duration is calculated using the time difference between "exceeding VT4" and the last "undercutting of the VT1 threshold" with the appropriate current thresholds. The resulting duration or start/begin of the "I_Dig_Out" signal can be corrected by setting thresholds, "I_Dig_Out_Offset_Time", the timetable and the x/y time offset. Other influencing factors may be the quality of the fuel, rail pressure etc – ask your injector supplier for the relevant curves.

Fig. 18-19 shows an example: an "unrealistic" curve (possibly relevant for SW tests) (axis unit in microseconds). Using real-time parameters (x/y time offset), the position of the curve during runtime can be influenced via the model. This is attained with the simple addition of the timetable values to the x/y time offset values.

We recommend using the default values of the current/voltage thresholds (see Tab. 18-11 on page 601).

The amplitude of the feature cannot be directly manipulated and depends on the energy injected into the injector simulation. The amplitude of the feature is all the greater, the earlier the feature is to be generated after high-speed quenching (last VT2 event).

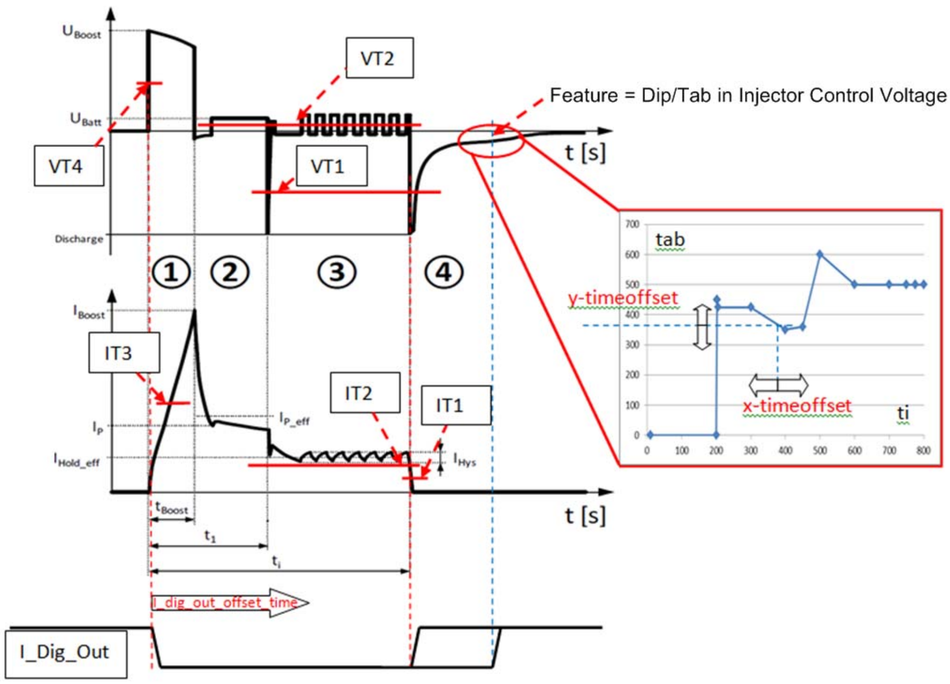


Fig. 18-19 Typical HDEV5 CVO Injector Current/Injector Voltage Flow with Feature and "I_Dig_Out" Signal

18.8.2 CRI2-x Injector Simulation (VCC/VCA) with ES4457.1

The ES4457.1 simulates CRI2-x injectors and enables the stimulation of VCC/VCA control algorithms. VCC/VCA stands for "Valve Closing Control" or "Valve Closing Adjustment" respectively and uses physical effects or features in electric signals in injector control for the precise determination of actual injector opening times.

A typical current flow of a CRI2-x injector is shown in Fig. 18-20. The definable current thresholds are shown in red and can be modified in the "Signals" tab. The preset values and their functions are described in Tab. 18-11 on page 601.

The occurrence of the rising edge of "I_Dig_out" can also be influenced by setting current/threshold values. In VCC mode, "I_Dig_Out" will experience its rising edge when the feature occurs (blue dashed line). In what is referred to as Legacy Mode (red dashed line), a rising edge of the "I_Dig_Out" signal is generated when current threshold IT2 is undercut.

Feature generation in VCC injector simulation is realized with a static delay and can only be influenced in one step – VCC must be disabled via RTIO for this purpose.

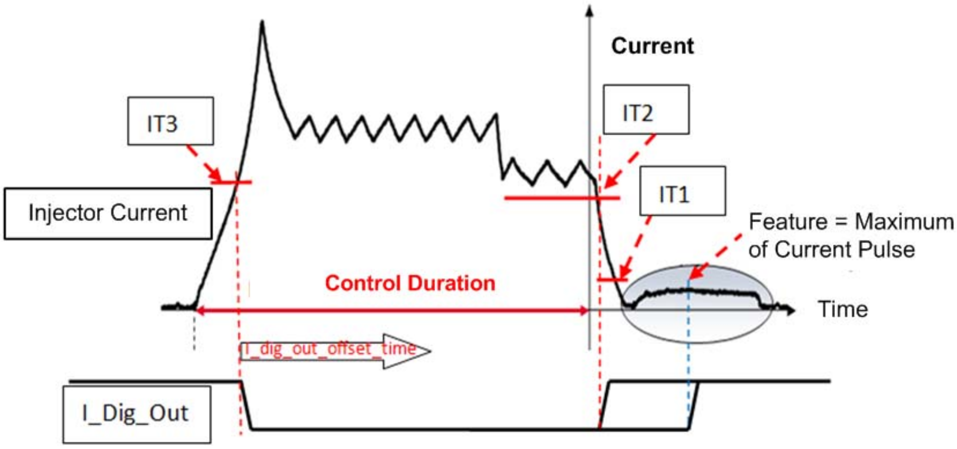


Fig. 18-20 Typical CRI2-x Injector Current Flow with Feature (Gray Oval) and "I_Dig_Out" Signal

18.8.3 Passive Injector Simulation

The I_Dig_out generation of the passive injector simulation can be manipulated as described in Tab. 18-11 on page 601.

Threshold	**ES4455.2**	**ES4452.1 (CVO gasoline)**	**ES4457.1 (VCC diesel)**	**ES4451.4 (gasoline)**	**ES4450.3 (diesel)**
Injector current threshold IT1 (internal)	No function	Internal use (default value: 1 A)	Internal use (default value: 1 A)	Internal use (default value: 1 A)	Internal use (default value: 1 A)
Injector current threshold IT2 (internal)	No function	Internal use (default value: 2 A). Must be significantly smaller than the minimal value of the hold current ripple.	Control end; rising edge of I_Dig_Out_x if VCC disabled; I_Dig_Out_x extension to current maximum of feature; Must be significantly smaller than the minimal value of the hold current ripple (default value: 1 A).	Control end; rising edge of I_Dig_Out_x	Control end; rising edge of I_Dig_Out_x
Injector current threshold IT3	No function	No function	No function	Control start; falling edge of I_Dig_Out_x	Control start; falling edge of I_Dig_Out_x
Injector current threshold IT4 (internal)	No function	Internal use (default value: 12 A)	Internal use (default value: 12 A)	No function	No function

Threshold	ES4455.2	ES4452.1 (CVO gasoline)	ES4457.1 (VCC diesel)	ES4451.4 (gasoline)	ES4450.3 (diesel)
Injector voltage threshold VT1	No function	Control end/pickup phase detection (default value: -15 V); Rising edge of I_Dig_Out_x (if CVO disabled); Start tab generation depending on ti = time between VT4 and VT1 occurrence if IT1 is underrun or timetable value	Pickup phase detection (default value: -15 V)	No function	No function
Injector voltage threshold VT2	No function	Detection of the PWM signals in the hold phase (default value: 5 V)	Detection of the PWM signals in the hold phase (default value: 5 V)	No function	No function
Injector voltage threshold VT4	No function	Control start; falling edge of I_Dig_Out_x	Start of trigger; falling edge of I_Dig_Out_x	No function	No function

Tab. 18-11 Thresholds and their Function/Influence on "I_Dig_Out" and/or Feature Generation

19 ES5300.1-A Housing

The ES5300.1-A Housing is a system housing for creating a Hardware-in-the-Loop test system. Digital and analog interfaces to the ECU in the form of PCI Express, SPI or I^2C based boards can be installed in the ES5300.1-A Housing.

19.1 ES5300-Chassis System

This RTIO item represents an ES5300.1-A Housing and enables the hierarchical representation of the plug-in boards.

The following are intended to be installed in an ES5300.1-A Housing:

- ETAS and third-party PCIe plug-in boards in PC form factor by means of an ES5370.1 adapter
- PCIe plug-in boards in ES5300 form factor *
- SPI plug-in boards in ES5300 form factor
- ES443x load boards by means of ES5371.1 adapters
- ES445x load boards by means of ES5372.1 adapters *

* currently no support available.

However, the RTIO system can only represent the ETAS I/O boards. Please observe the following differences and rules when handling the ETAS boards:

- The instancing of the PCIe boards applies to the entire system whereby boards of the same type are numbered in ascending order, starting with "ES5300-Chassis" instance 0, slot 0.
- The arrangement of the PCIe boards with PC form factor in an ES5300-Chassis System is for reasons of clarity only. They can also be arranged directly under the "RTPC" RTIO item. The board is selected exclusively using the instance number.
- SPI boards of the same type, on the other hand, are numbered in ascending order within each "ES5300-Chassis" RTIO item, starting with slot 0 and instance number 0.
- The adapters required for the ES44xx load boards are transparent on the RTIO side and are thus not represented.

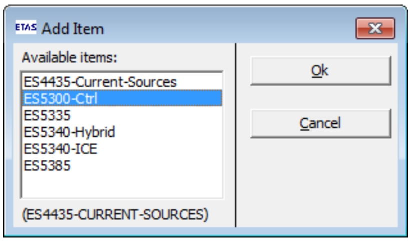


Fig. 19-1 The RTIO Items Available under an ES5300-Chassis

19.1.1 Globals (ES5300-Chassis System)

The instance number of the ES5300-Chassis RTIO item can be changed in its "Globals" tab. When several ES5300 systems are used, instance numbers are assigned according to the topology of the PCIe and SYNC bus.

The instance number is required by the RTIO drivers of the SPI boards and by the driver assigned to the ES5300-Ctrl Subsystem.

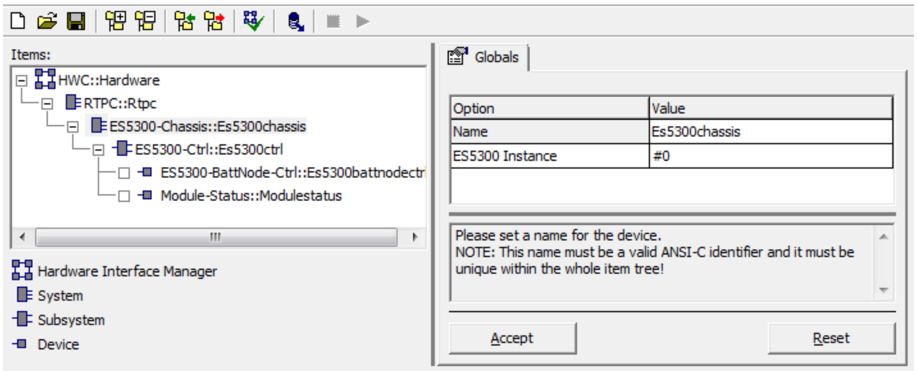


Fig. 19-2 The "Globals" Tab of the ES5300-Chassis System

Note

If only one ES5300.1-A Housing is used, "ES5300 Instance" should have the value "0" – for an additional ES5300.1-B Housing "ES5300 Instance" must be set to "1"!

19.2 ES5300-Ctrl Subsystem

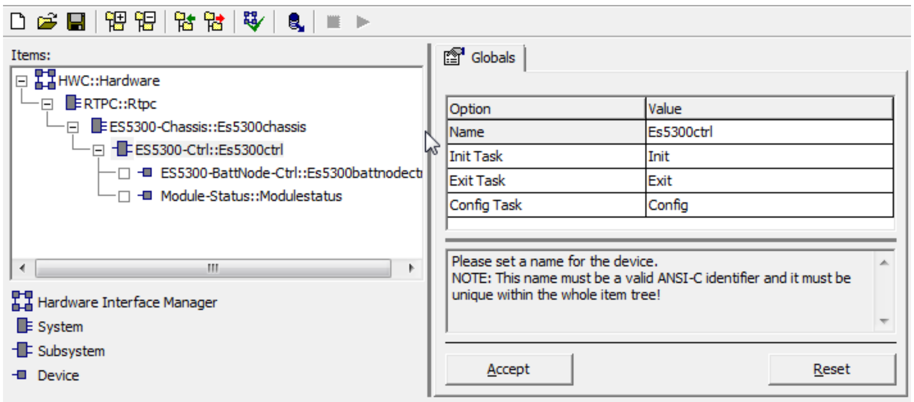


Fig. 19-3 The "Globals" Tab of the ES5300-Ctrl Subsystem

19.3 ES5300-BattNode-Ctrl Device

The ES5300-BattNode-Ctrl Device represents the driver level of the ES5300.1-A Housing – the driver makes it possible to control the battery node signals.

19.3.1 Globals (ES5300-BattNode-Ctrl)

The device name can be changed in the "Globals" tab. It is used as a prefix for all global identifiers in C code generation.

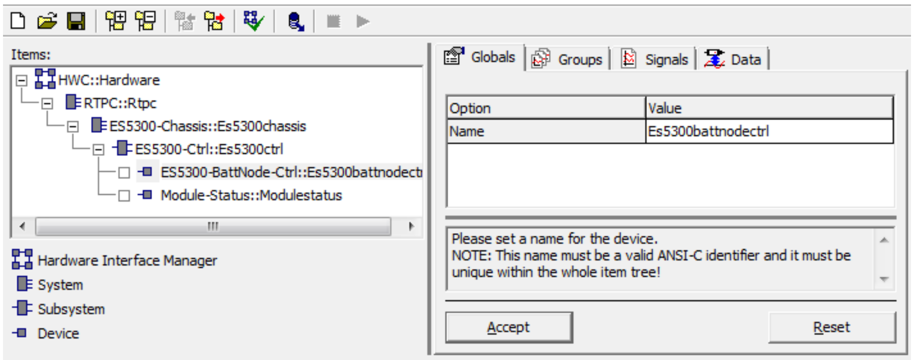


Fig. 19-4 The "Globals" Tab of the ES5300-BattNode-Ctrl Device

19.3.2 Groups (ES5300-Batt-Node)

The ES5300-Batt-Node Device implements a send signal group "BattNodeCtrl".

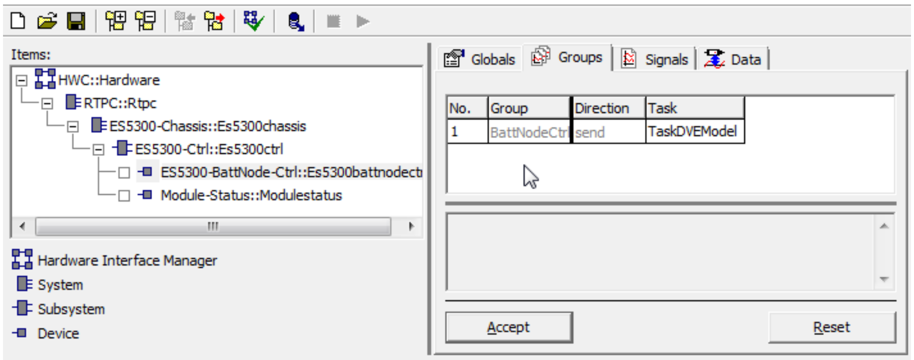


Fig. 19-5 The "Groups" Tab of the ES5300-BattNode-Ctrl Device

19.3.3 Data (ES5300-Batt-Node)

There are six signals available for controlling the signals routed to the backplane for battery node control.

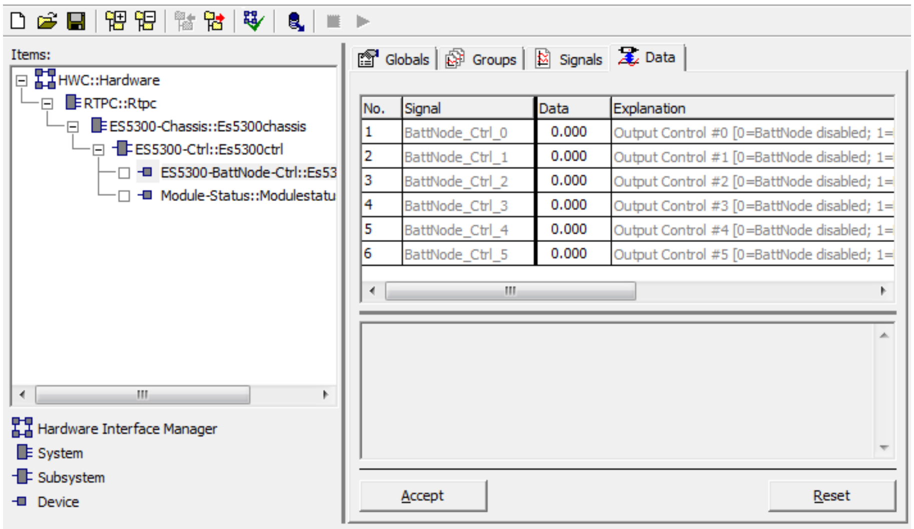


Fig. 19-6 The "Data" Tab of the ES5300-BattNode-Ctrl Device

20 ES5321.1 PWM I/O Board

The ES5321.1 PWM I/O Board is used to issue and acquire pulse-width modulated signals. A typical example of how the ES5321.1 is used is, e.g., in LABCAR projects where it is used to simulate vehicle sensors and address actuators.

The ES5321.1 PWM I/O Board is integrated in the RTIO-Editor by selecting the ES5321 subsystem.

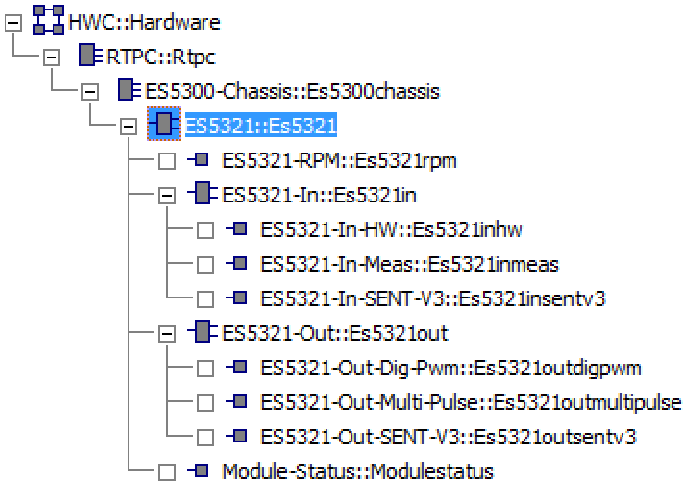


Fig. 20-1 RTIO Hardware Description with ES5321.1 PWM I/O Board

The ES5321.1 PWM I/O Board has 48 input channels (four galvanically isolated groups, each with twelve channels) for acquiring pulse-width modulated signals – these input channels are configured using the ES5321-In-HW subsystem.

In addition to this subsystem, up to four ES5321-In-SENT and eight ES5321-In-Meas subsystems can be assigned to the ES5321-RTIO-Element.

The up to four ES5321-Out subsystems can each consist of one ES5321-Out-Dig-Pwm, up to four ES5321-Out-Multi-Pulse devices and up to twelve ES5321-Out-SENT devices.

Measure Functions of the ES5321.1 PWM I/O Board

The inputs of the ES5321.1 have the following measure functions:

- Level (Active High)
- Level (Active Low)
- High Time [µs]
- Low Time [µs] (yes)
- Additive High Time [µs] *
- Additive Low Time [µs] *
- Cycle Time --/-- [µs]
- Frequency --/-- [Hz]
- Duty Cycle L/(L+H) --/--
- Duty Cycle H/(L+H) --/--
- Cycle Time --\-- [µs]
- Frequency --\-- [Hz]
- Duty Cycle L/(L+H) --\--
- Duty Cycle H/(L+H) --\--

20.1 ES5321 Subsystem

20.1.1 Globals (ES5321 Subsystem)

This section describes the global options of the ES5321 subsystem.

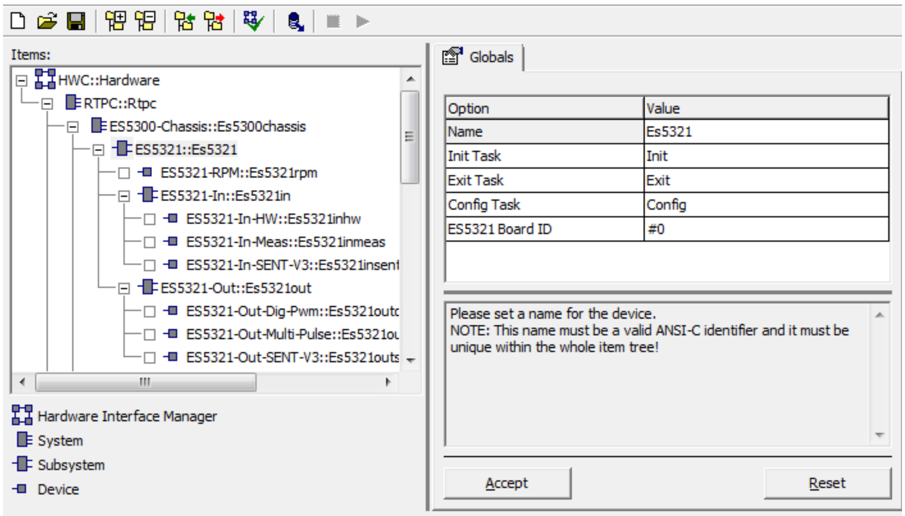


Fig. 20-2 The "Globals" Tab of the ES5321 Subsystem

ES5321 Board ID

This option field is used to identify the ES5321.1 PWM I/O Board.

In the RTIO-Editor, up to ten ES5321.1 PWM I/O Boards can be integrated per chassis.

This RTIO parameter cannot be edited during runtime of the model on the experimental target.

20.2 ES5321-RPM Device

The ES5321-RPM device is used to configure and control the ES5321-RPM unit.

20.2.1 Globals (ES5321-RPM)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

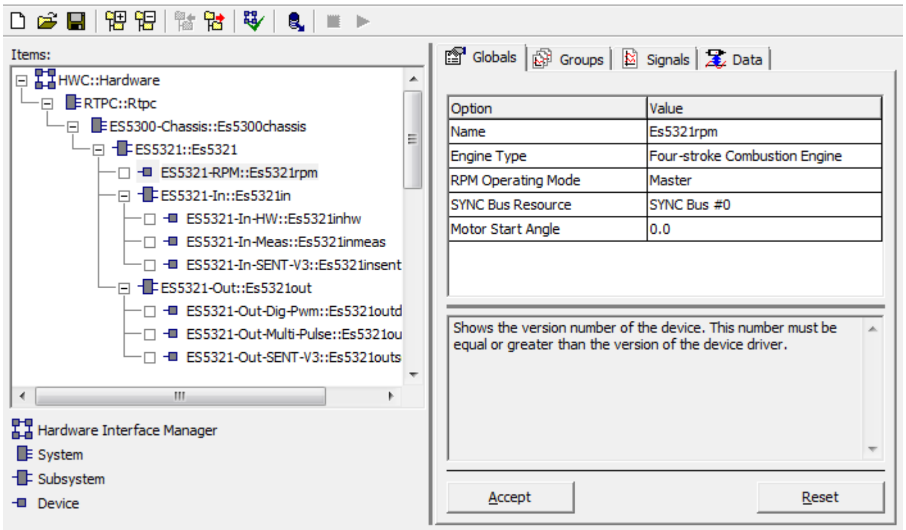


Fig. 20-3 The "Globals" Tab of the ES5321-RPM Device

Engine Type

Selection of the engine type.

RPM Operating Mode

Selection of master or slave mode.

Note

The options "Engine Type" und "RPM Operating Mode" are only visible when ***View → Show All*** *is selected!*

SYNC Bus Resource

The angle clock signal is transported between the boards via synchronization bus lines. This configuration parameter is used to select the bus line via which the angle clock is read in in slave mode or output in master mode.

If the RPM unit is configured to master mode, there is an additional option "None" – in this case an angle clock is generated and used internally on the board, but not output on one of the two synchronization buses.

Note

Please note that there can only be one angle clock master on each synchronization bus!

Motor Start Angle

The angle at which the engine starts after model initialization.

The following table summarizes the properties of the individual configuration parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Engine Type	int	No	0: Four-stroke combustion engine 1: Electric motor 2: Two-stroke combustion engine
RPM Operating Mode	int	Yes	0: Slave 1: Master
SYNC Bus Resource	int	Yes	0: SYNC Bus #0 1: SYNC Bus #1 255: None (only in master mode)
Motor Start Angle	float	Yes	0.0 °RA...360.0 °RA: E-Motor 0.0 °CA...360.0 °CA: Two-stroke engine 0.0 °CA...720.0 °CA: Four-stroke engine

* Data type which the RTIO driver uses internally for the parameter

Tab. 20-1 The "Globals" Tab of the ES5321-RPM Device

Note

Please note that the angle limits (720 ° or 360 °) vary depending on the "Engine Type" selected here. This can have a corresponding effect on the value range of other parameters!

20.2.2 Groups (ES5321-RPM)

The ES5321-RPM device has an "Outputs" signal group for controlling the RPM unit and an "Inputs" signal group for reading the current status information of the RPM unit.

20.2.3 Signals (ES5321-RPM)

The signals made available in the "Signals" tab for controlling and monitoring the status of the RPM unit depend on the type of engine selected. Fig. 20-4 on page 612 shows the RTIO signals for the engine type "combustion engine", Fig. 20-5 on page 612 shows the RTIO signals for the engine type "electric engine". Tab. 20-2 on page 612 and Tab. 20-3 on page 613 lists the signal properties.

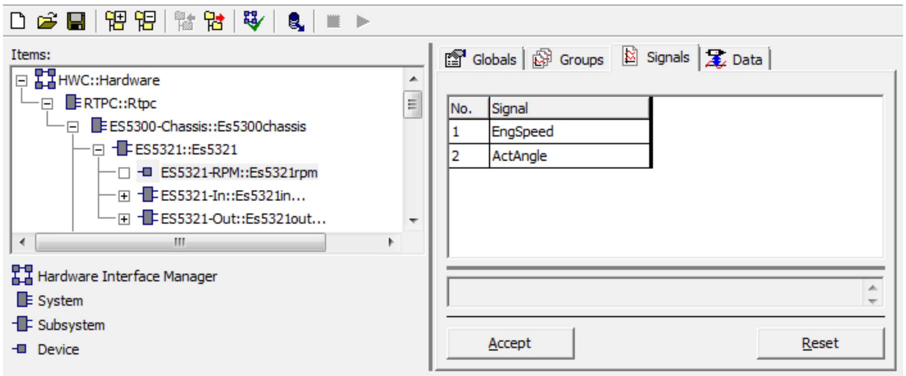


Fig. 20-4 The "Signals" Tab of the ES5321-RPM Device (Combustion Engine)

Signal Name	Data Type	Notes
EngSpeed	float	Engine speed in revolutions per minute Value range: Four-stroke engine: -60000.0 rpm…+60000.0 rpm resolution: 0.001 rpm Two-stroke engine: -30000.0 rpm…+30000.0 rpm resolution: 0.0005 rpm An engine rotating backwards can be simulated by entering a negative speed.
ActAngle	float	Current degree crankshaft angle in °CA Value range: 0.0 °CA … 720.0 °CA (or 360 °CA) Resolution: 0.011 °CA (or 0.0055 °CA)

Tab. 20-2 ES5321-RPM Device: RTIO Signals

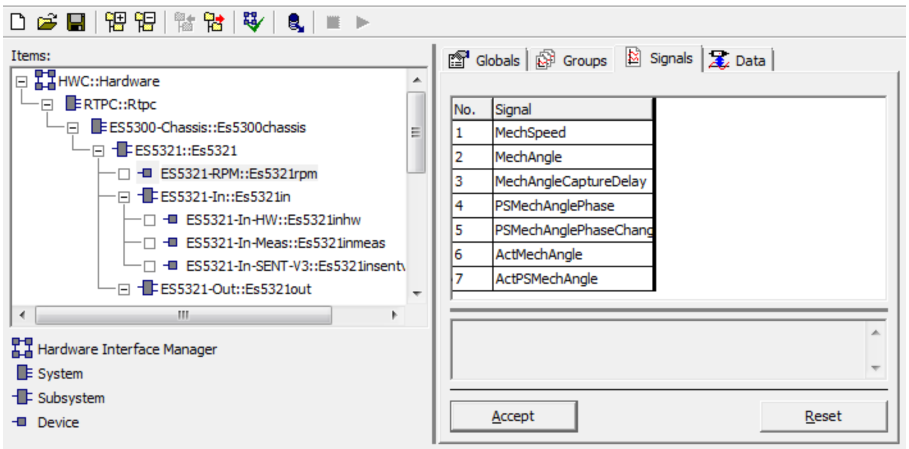


Fig. 20-5 The "Signals" Tab of the ES5321-RPM Device (Electric Motor)

Signal Name	Data Type	Notes
MechSpeed	float	Engine speed in revolutions per minute Value range: -30000.0 rpm … +30000.0 rpm Resolution: 0.001 rpm An engine rotating backwards can be simulated by entering a negative speed.
MechAngle	float	Target angle specification in degrees Value range: 0.0 °RA … 360 °RA Resolution: 0.0055 °RA This signal is only of any importance when the RPM unit is operated as "Master with Angle Adaption".
MechAngle-CaptureDelay	float	This signal defines the delay (in µs) until the hardware accepts the target values for RPM and angle (with reference to the trigger event). Maximum value: 1000.0 µs This signal is only of any importance when the RPM unit is operated as "Master with Angle Adaption".
PSMechAngle-Phase	float	Offset of the phase-shifted mechanical angle in relation to the rotor angle Value range: -360.0 °RA … 360.0 °RA Resolution: 0.0055 °RA
PSMechAngle-PhaseChange-Speed	float	Change speed of the phase-shifted mechanical angle in relation to the rotor angle after a "PSMechAngle" jump (in degrees per second) Maximum change speed: 1000.0 °RA/s If 0 is entered here, target phase shift is set immediately.
ActMechAngle	float	Current rotor angle in degrees Value range 0.0 °RA … 360 °RA Resolution: 0.0055 °RA
ActPSMechAngle	float	Current value of the angle phase-shifted with respect to the rotor angle in degrees Value range 0.0 °RA … 360 °RA Resolution: 0.0055 °RA

Tab. 20-3 ES5321-RPM Device: RTIO Signals (Electric Motor)

20.3 ES5321-In Subsystem

20.3.1 Globals (ES5321-In Subsystem)

The ES5321-In subsystem is used to configure the measure inputs of the ES5321.1 PWM I/O Board.

Fig. 20-6 shows the "Globals" tab of the ES5321-In Subsystem.

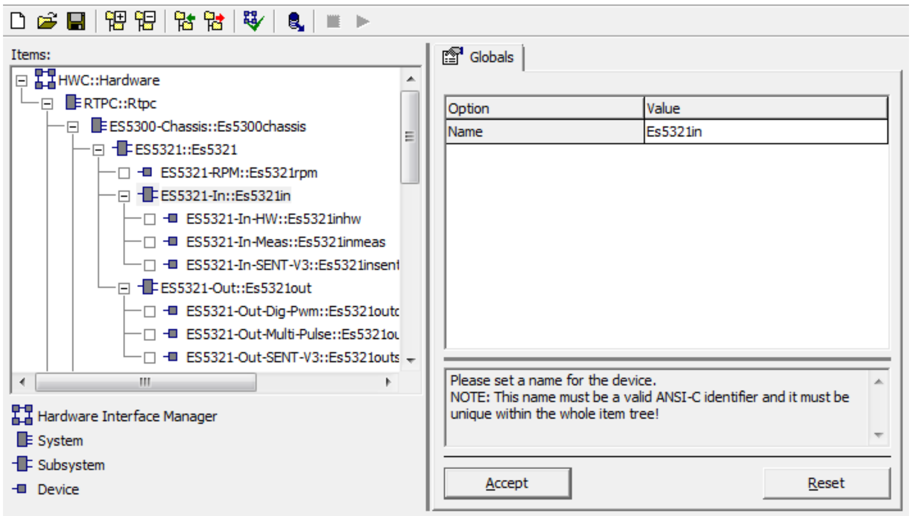


Fig. 20-6 The "Globals" Tab of the ES5321-In Subsystem

20.4 ES5321-In-HW Device

20.4.1 Globals (ES5321-In-HW Device)

The ES5321-In-HW device is used to activate channels and to set the threshold levels of the input comparators.

The following figure shows the RTIO parameters of the "Globals" tab.

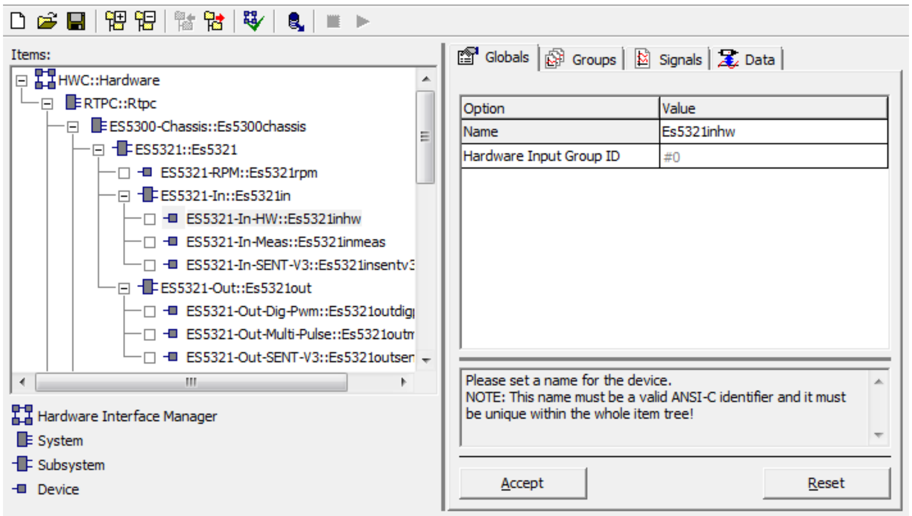


Fig. 20-7 The "Globals" Tab of the ES5321-In-HW Device

Hardware Input Group

One of the four galvanically isolated input groups (0...3).

20.4.2 Groups (ES5321-In-HW Device)

The ES5321-In-HW device has a send signal group "Control" for controlling the board and a receive signal group "Level" for reading in the input values.

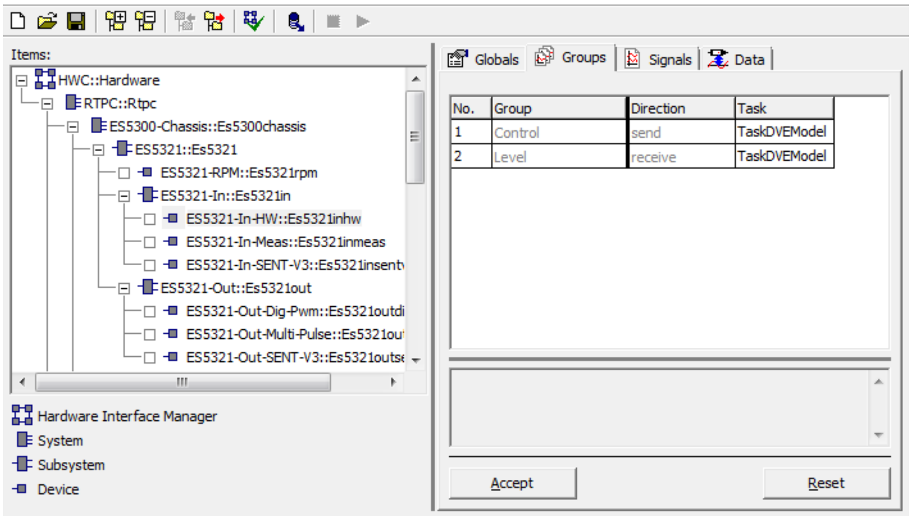


Fig. 20-8 The "Groups" Tab of the ES5321-In-HW Device

20.4.3 Signals (ES5321-In-HW Device)

The following figure shows the signals of the ES5321-In-HW Device.

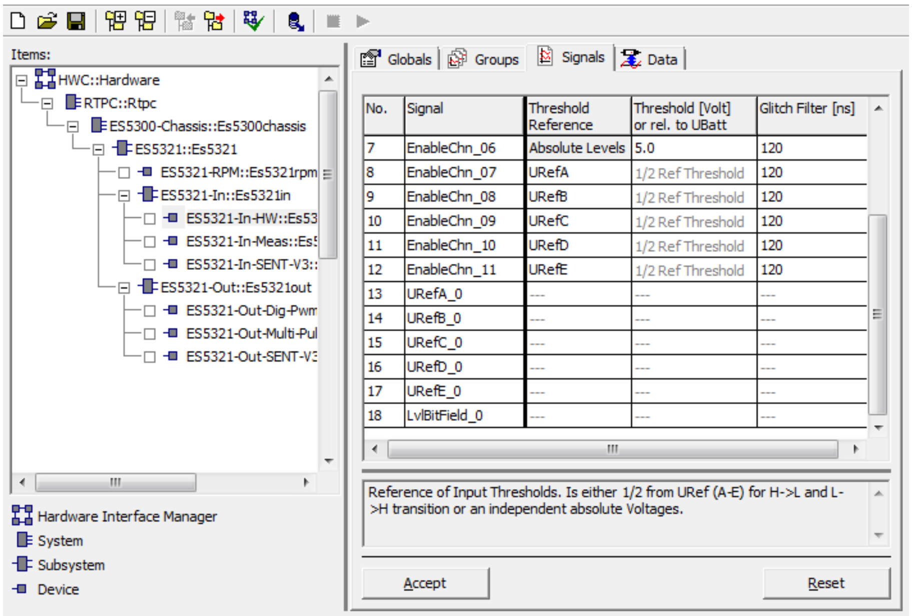


Fig. 20-9 The "Signals" Tab of the ES5321-In-HW Device

Signal Name	Data Type	Notes
EnableChn_xx	bool	0: Input is disabled 1: Input is enabled (default)
URefA...URefE	real32	Reference voltages [1...19 V]
LvlBitField_0	unint32	Bit field containing the level information of the inputs

Tab. 20-4 ES5321-In-HW Device: RTIO Signals

These signals have the following parameters:

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Threshold Reference	uint8	Yes	Reference of input threshold. Thresholds are either relative to URef or absolute voltage values.
Threshold or relative to UBatt	real32	Yes	Depending on setting for Threshold Reference. Only editable when absolute voltage threshold is set up.
Glitch Filter	uint16	Yes	Values less than 120 will be mapped to 0.0. Values between 120 and 5000 will be rounded to the nearest value that is dividable by 8. Default: 120 ns (Resolution 8 ns)
* Data type which the RTIO driver uses internally for the parameter			

Tab. 20-5 ES5321-In-HW Device: Configuration Parameter of the "Signals" tab

20.5 ES5321-In-Meas Device

20.5.1 Globals (ES5321-In-Meas Device)

The ES5321-In-Meas device is used to configure the measure functions on the 24 input channels.

No settings need to be made in the "Globals" tab.

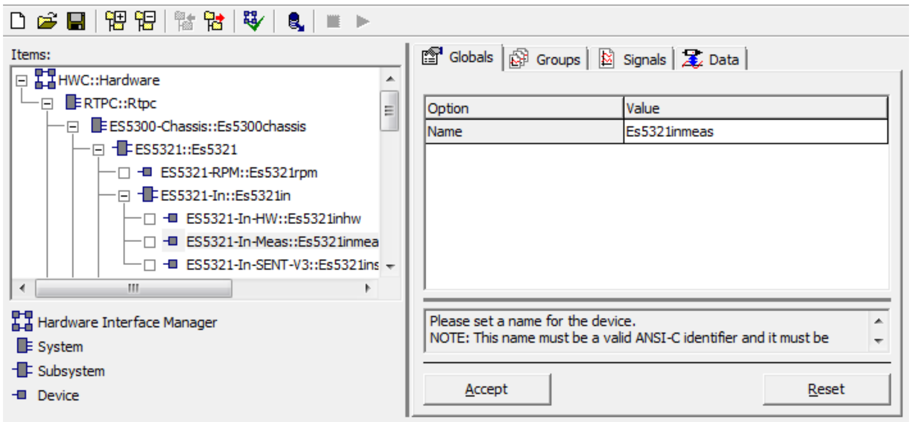


Fig. 20-10 The "Globals" Tab of the ES5321-In-Meas Device

20.5.2 Signals (ES5321-In-Meas Device)

In the "Signals" tab, the 192 measurements (in the case of eight ES5321-In-Meas devices) of an ES5321-In-Meas device are defined which are divided into blocks of 24 channels each.

All option fields can be edited online (i.e. during runtime of the model on the experimental target).

The following figure shows the signals of the "Signals" tab.

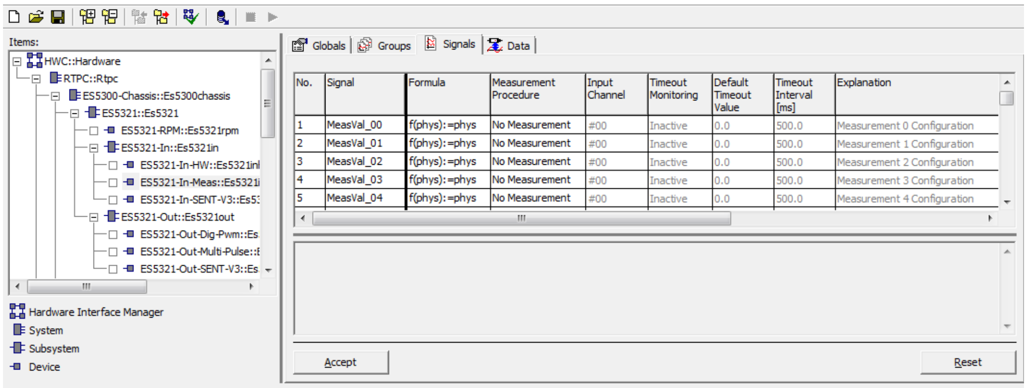


Fig. 20-11 The "Signals" Tab of the ES5321-In-Meas Device

Signal Name	Data Type	Notes
MeasVal_*n* (n = 0..23)	real32	The 24 measurements
Tout_*n* (n = 0..23)	real32	Timeout configuration 0: no timeout 1: timeout 2: not used

Tab. 20-6 The Signal of the ES5321-In-Meas Device

These signals have the following parameters:

Measurement Procedure

The measurement procedure is set in this list field. Refer to the section "The Measurement Procedures" on page 620 for a detailed description and configuration guidelines on the individual procedures.

Input Channel

This is where the input channel of the ES5321.1 to be used for measuring is set.

Timeout Monitoring

Definition of timeout monitoring for the relevant measurement. The following settings are possible:

- "Inactive": no timeout monitoring.
- "Intvl Predef": timeout monitoring in the intervals defined in the "Timeout Interval [ms]" option field. The timeout measure value is the value set in the "Default Timeout Value" option field.
- "Intvl InpDep": timeout monitoring in the intervals defined in the "Timeout Interval [ms]" option field. The timeout measure value depends on the level of the input signal.

Default Timeout Value

The value set in this numeric input field is issued as the timeout measure value if "Intvl Predef" timeout monitoring mode has been set.

The timeout condition is fulfilled if one of the four evaluated timestamps is outside the timeout value set.

Timeout Interval [ms]

Interval for monitoring a timeout. This value can only be edited in the timeout operating modes "Intvl Predef" and "Intvl InpDef".

- Default value: 500.0 [ms]
- Setting range: 0.1 … 10000.0 [ms]

Measure Methods and Timeout Handling

If the "Intvl InpDep" option was selected in "Timeout Monitoring", the following values are issued – depending on the measurement procedure set.

Measure Method	Value (Input Level Dependent)
No Measurement	-
High Time [µs]	0
Low Time [µs]	0
Additive High Time [µs] (free-running)	0
Additive Low Time [µs] (free-running)	0
Additive High Time [µs] (delta)	0
Additive Low Time [µs] (delta)	0
Cycle Time --/-- [µs] Cycle Time --\-- [µs]	Time passed since last valid timestamp
Frequency --/-- [Hz] Frequency --\-- [Hz]	Reciprocal of time passed since last timestamp
Duty Cycle L/(L+H) --/-- Duty Cycle L/(L+H) --\--	1: input level is low 0: input level is high
Duty Cycle H/(L+H) --/-- Duty Cycle H/(L+H) --\--	1: input level is high 0: input level is low
Level (Active High)	0
Level (Active Low)	0

20.6 The Measurement Procedures

Definitions

Active signal edge:

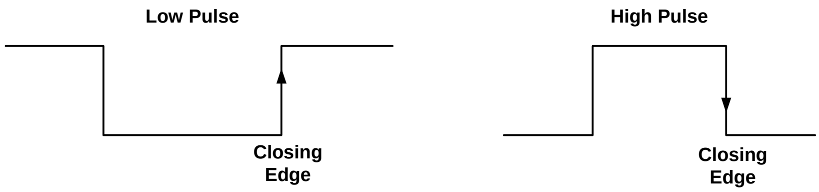
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal:

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing edge:

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.

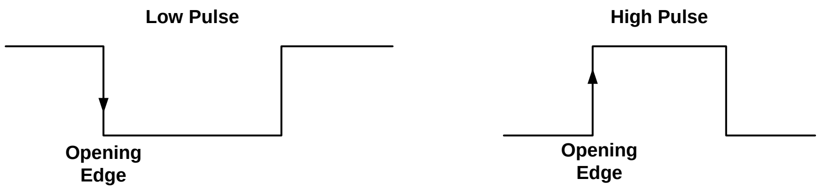


Inactive signal:

A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

Opening edge:

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



The Basic Principle of Measure Value Calculation

The measure data (timestamp of detected edges) is transmitted by DMA (Direct Memory Access) from the ES5321.1 PWM I/O Board to the Real-Time PC where it is evaluated.

The evaluation takes place in the RTIO driver which then forwards the measure value and the timeout value to the GUI.

The measurements have a resolution of 8 ns.

20.6.1 Pulse-Width Measurements

The measurement procedures

- High Time [µs]
- Low Time [µs]

measure the high and low times of the pulses of a PWM input. The pulse width of the pulse immediately preceding the last time of data acquisition is measured. Fig. 20-12 shows an example of a high-time measurement; the measured pulse is shown in bold print.

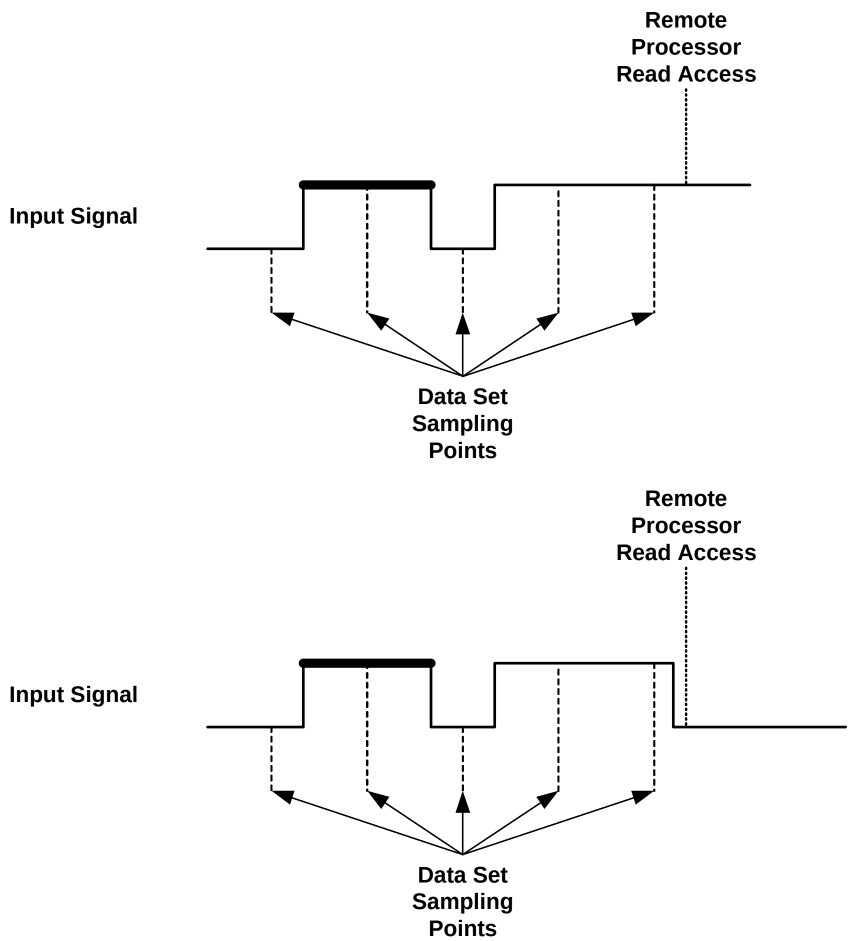


Fig. 20-12 High-Time Measurement (the Measured Pulse is shown in Bold Print)

A timeout occurs when at least one of the four edges is older than the current timestamp minus the timeout time.

Timeout handling is the same for all measurement procedures.

20.6.2 Additive Pulse-Width Measurements

If the read process for the "MeasVal" signal group is activated again (periodically in special cases), the measurement procedures

- Additive High Time [µs]
- Additive Low Time [µs]

return the time in which the signal to be measured has assumed an active level between two consecutive activations of the read process.

Fig. 20-13 shows the measure value calculation using the example of an additive high-time measurement. The total of the line segments marked in bold print is the additive high time which is returned on the (n+1)th activation of the read process.

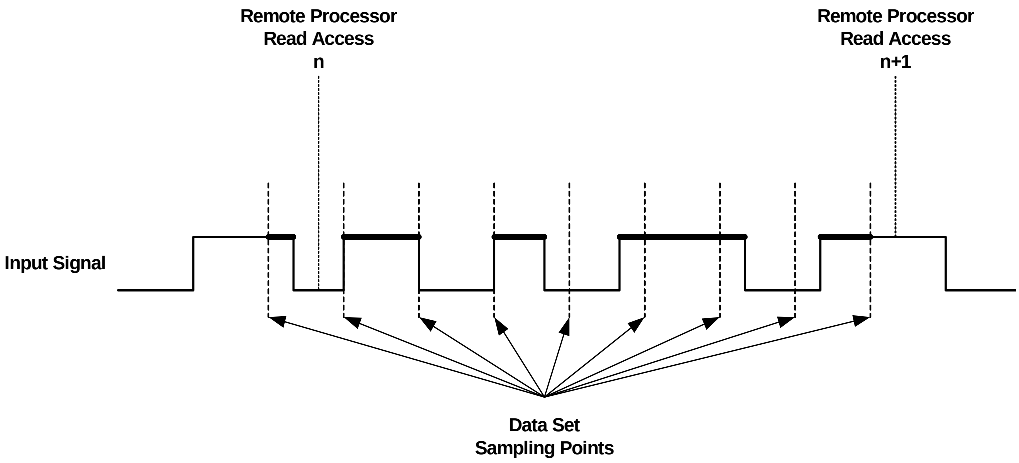


Fig. 20-13 Additive High-Time Measurement: Illustration of Measure Value Calculation

A timeout occurs when at least one of the four edges is older than the current timestamp minus the timeout time.

"Free Running" Mode and "Delta" Mode

There are two variants of the two measuring methods "Additive High Time" and "Additive Low Time":

- Free running

  This method returns the "Additive High/Low Time" since the start of simulation. The time is measured in "Clock Ticks" to every 8 ns and stored as a 64-bit integer.

- Delta

  This method returns the "Additive High/Low Time" between two calls of the task running the measuring function.

Note

*As the hardware uses a 32-bit counter to measure the additive times, the time between two calls of the function cannot be more than 34.38 s (2^{32} * 8 ns). Otherwise there is no guarantee that the measured data ("delta" and "free-running") is correct!*

20.6.3 Frequency and Cycle Time Measuring

The measurement procedures

- Cycle Time
- Frequency [Hz]

return the cycle time or the frequency of the signal at a PWM input measured on rising or falling edges. The active signal edge immediately preceding the last[1] time of data acquisition and its corresponding period are evaluated.

Fig. 20-14 shows the evaluated period using the example of a frequency or cycle measurement on rising edges.

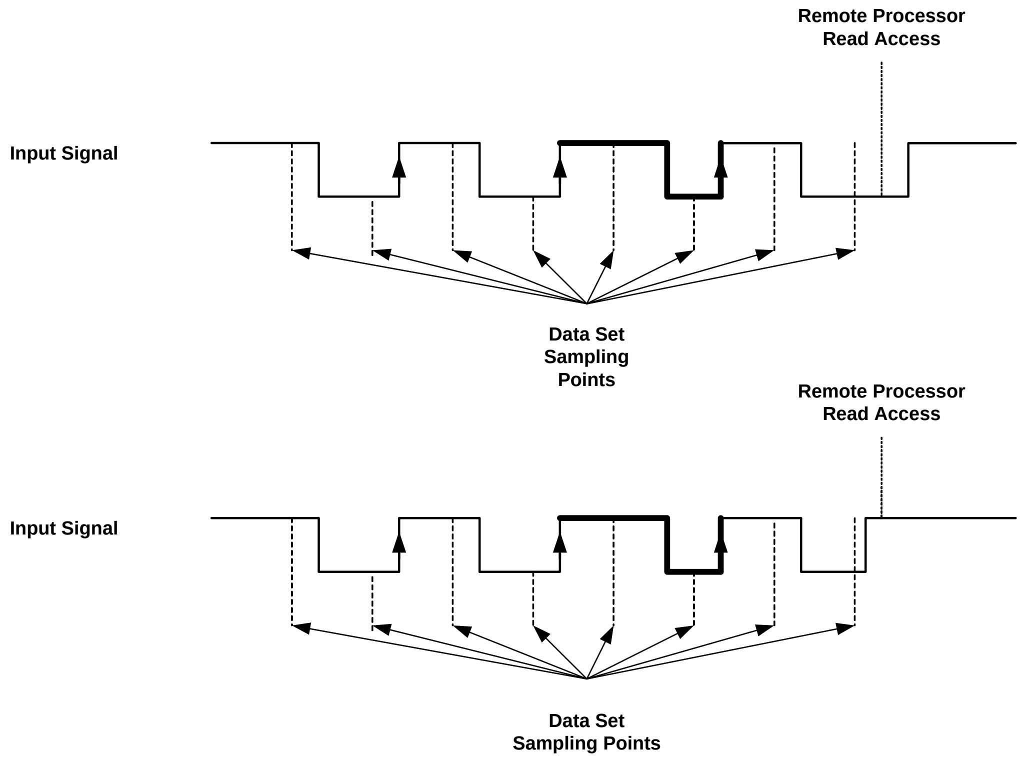


Fig. 20-14 Evaluated Period (in Bold) in a Frequency or Cycle Measurement on Rising Edges

20.6.4 Duty Cycle Measurements

The following duty cycle measurements are available with the ES5321.1 PWM I/O Board:

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

In these measurements, a pulse duration is set in relation to the signal cycle time. The user specifies the edge on which the measure value is calculated when selecting the measurement procedure.

1. The time of data acquisition immediately preceding the RTIO driver's read access of the data set

As far as the selection of the period of the input signal is concerned, on the basis of which the duty cycle calculation is executed, exactly the same as was described for frequency and cycle time measurements in section 20.6.3 on page 623 applies.

Note

*In order to detect a **duty cycle of 0 or 1** the measurement must be combined with timeout monitoring "Intvl InpDep":*

- *For PWM signals having a **constant frequency** and a variable duty cycle, "Timeout Interval [ms]" must be set to a time greater than or equal to the cycle time of the PWM input signal.*
- *For PWM signals having a **variable frequency**, "Timeout Interval [ms]" must be set to a time greater than or equal to the maximum cycle time of the PWM input signal.*

At the end of the timeout interval the measurement is set to 0 or 1 according to the measurement method used and the input level.

*If the **frequency of the PWM input signal is measured simultaneously**, the timeout measurement methods "Intvl Predef" and "Intvl InpDep" can be used in order to set the measurement to the desired value:*

- *Timeout monitoring "Intvl Predef" returns a predefined frequency.*
- *Timeout monitoring "Intvl InpDep" returns the reciprocal value of the time having passed since the last valid time stamp.*

20.6.5 Level Measurements

The measurement procedures

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

return the level of a PWM input in the form of active/inactive information. "0" means the signal is inactive; "1" means the signal is active.

20.7 ES5321-In-SENT Device

20.7.1 Globals (ES5321-In-SENT Device)

This RTIO element makes it possible to receive signals in accordance with the SENT data protocol – up to four ES5321-In-SENT devices can be added.

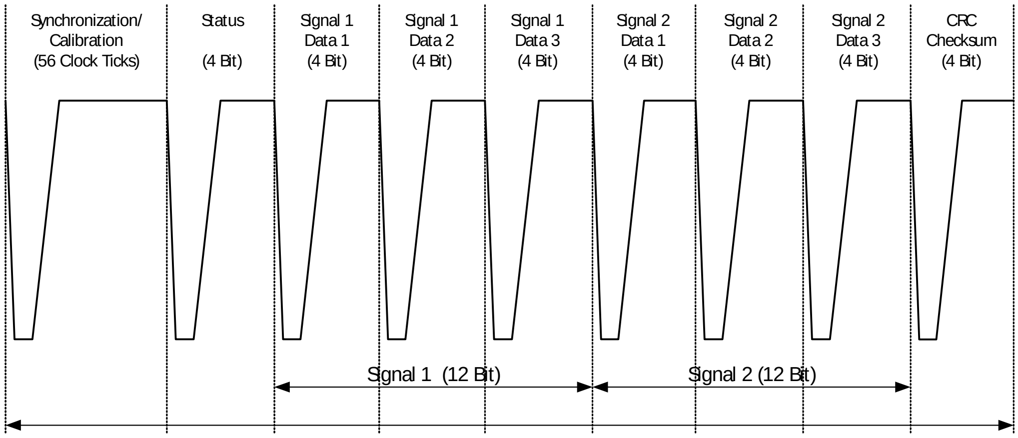


Fig. 20-15 Format of a Signal in acc. with the SENT Data Protocol

Serial Messages Format

Bit 2 and bit 3 of the status nibble are used for the Serial Message Channel (SMC).

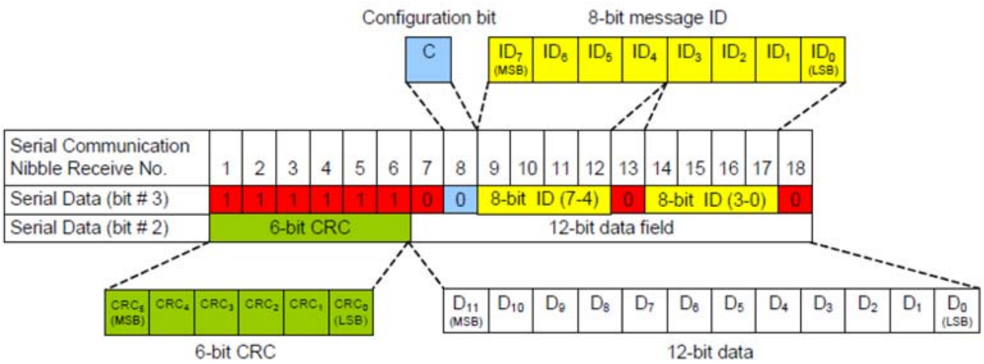


Fig. 20-16 Enhanced Serial Message Format with 12-Bit Data Field und 8-Bit Message ID

For 16 or 18 frames, the data in bit 2 and bit 3 of the status nibble is collected and stored in the memory of the ES5321.1.

The evaluation of the data is done in LABCAR-RTPC using a special C code module. This module is available as open source code and can be found on the LABCAR-OPERATOR installation medium in the `\MISC` folder.

The following figure shows the "Globals" tab of the ES5321-In-SENT device.

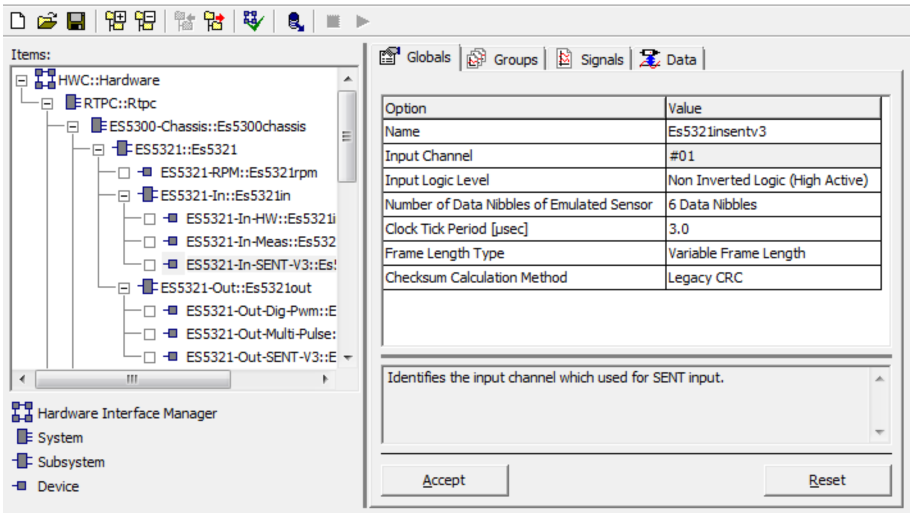


Fig. 20-17 The "Globals" Tab of the ES5321-In-SENT Device

Input Channel

Any input channel can be used – the level and threshold settings of this channel in the ES5321-In-HW device will be used. Please ensure that they correspond to the level specification of the SENT transmitter (typically: inactive = GND, active = 5 Volt).

Input Logic Level

This setting determines whether the level of the SENT signal is output inverted or non-inverted.

- Non inverted Logic (High Active)

  Fig. 20-18 shows a signal with non-inverted logic.

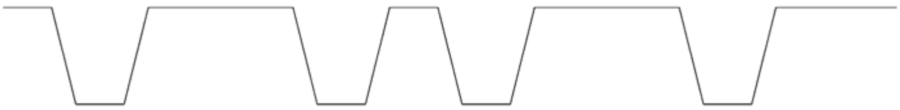


Fig. 20-18 Non-Inverted Logic

- Inverted Logic (Low Active)

  Fig. 20-19 shows a signal with inverted logic.

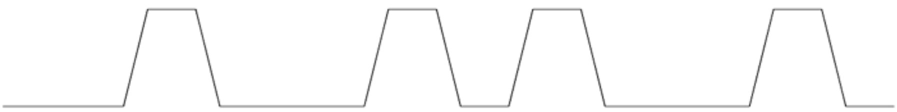


Fig. 20-19 Inverted Logic

Number of Data Nibbles of Emulated Sensor

This parameter is used to set the number of data nibbles of the emulated sensor.

Clock Tick Period [µsec]

Length (in µs) of a clock tick – the length of the "calibration/synchronization pulse" and of the "nibble pulses" are derived from this value.

Frame Length Type

The transmission mode of the SENT frames.

- Variable Frame Length

  A SENT frame is expected to follow the preceding frame immediately – there is no "pause" pulse.
- Constant Frame Length

  A "pause" pulse is expected in between two SENT frames. The length of the "pause" pulse is such that the whole frame (including "pause") is of constant length.

Checksum Calculation Method

The calculation method for the 4-bit checksum in the ES5321.1.

- None

  Checksum calculation is disabled.
- Recommended CRC

  Checksum calculation according to SENT specification Jan 2010.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.
- Legacy CRC

  Checksum calculation according to SENT specification April 2007 and Feb 2008.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.

20.7.2 Groups (ES5321-In-SENT Device)

The ES5321-In-SENT device implements a receive signal group "SentIn". This returns the uninterpreted data of the last SENT data word received in entirety and status information.

The interpretation of the data depends on the protocol type of the sender.

20.7.3 Signals (ES5321-In-SENT Device)

The following figure shows the "Signals" tab of the ES5321-In-SENT device.

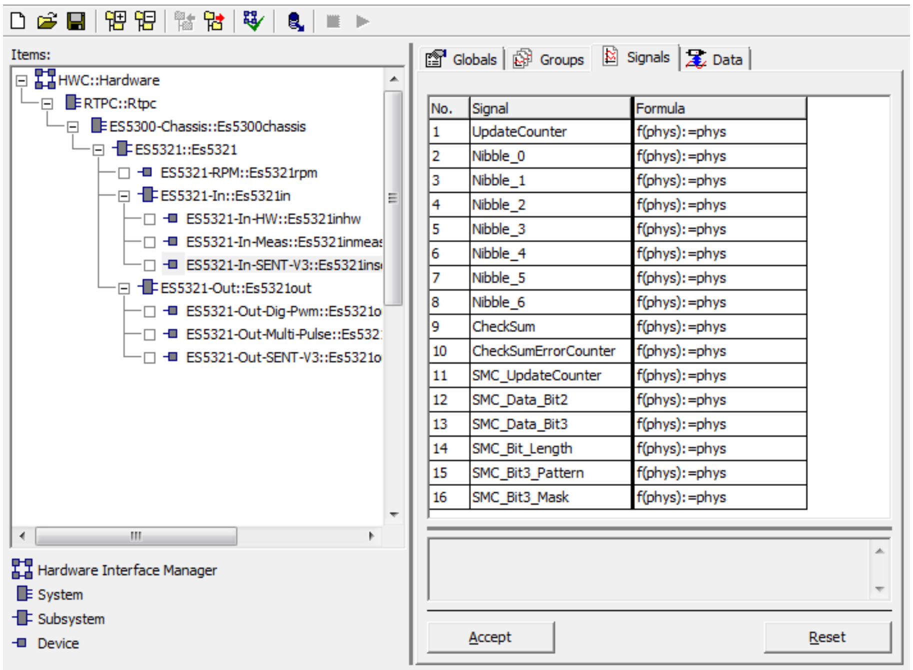


Fig. 20-20 The "Signals" Tab of the ES5321-In-SENT Device

Signals of the receive signal group "SentIn":

UpdateCounter

Counter which increases when new data are available – the value range is 0…255.

Nibble_0 ... Nibble_6

Values of the 7 nibbles (1 status nibble and 2x3 data nibbles) of the last SENT data word received – the value range is 0…15.

CheckSum

Checksum transmitted by the SENT frame.

CheckSumErrorCounter

Counter which increases when a SENT message was completely received in the ES5321.1 but the checksum was wrong.

Signals of the receive signal group "SentInSmc":

SMC_UpdateCounter

8-bit counter which increases when the ES5321.1 has received a complete SMC message and is ready for evaluation at the signals "SMC_Data_Bit2" and "SMC_Data_Bit3".

SMC_Data_Bit2

Data word containing the bit-2 data stream received (see SAE J2716 version Jan 2010, Ch.5.2.4). The higher bit positions contain data which was received later.

SMC_Data_Bit3

Data word containing the bit-3 data stream received (see SAE J2716 version Jan 2010, Ch.5.2.4). The higher bit positions contain data which was received later.

Signals of the send signal group "SentInSmcCtrl":

SMC_Bit_Length

Input signal controlling the SMC receiving unit of the ES5321.1.

"0" disables the reception, a value > 0 (… 31) enables it and defines the number of bits for a SMC cycle. According to the SENT specification, valid values are "16 " (short serial message) or "18 "(enhanced serial message).

SMC_Bit3_Pattern

Input signal controlling the SMC receiving unit of the ES5321.1.

It defines a bit pattern for bit 3 checking the validity of the SMC message.

A message is recognised as valid, when

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

I.e., "SMC_Bit3_Pattern" defines the value of the relevant bits.

SMC_Bit3_Mask

Input signal controlling the SMC receiving unit of the ES5321.1.

It defines a bit mask for bit 3 checking the validity of the SMC message.

A message is recognized as valid, when

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

I.e., "SMC_Bit3_Mask" marks the relevant bit positions with a "1".

The ES5321.1 and the RTIO only provides generic functionality for the reception of SMC messages. Control, decoding of the data and verification of checksums have to take place outside the RTIO.

A C code module template "SentInSMC_Control" with basic functionality is provided. For that purpose all SMC_* signals of the RTIO are connected to the corresponding ports of the C code module.

However, for a specific application the C code module must be adapted by

- defining "SMC format mode" according to the SENT specification
- converting the data values of the required Msg-IDs into physical values
- providing these values at the output ports

According to SAE J2716 Jan 2010 the following values are defined:

- 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)

The ES5321-Out subsystem is used to configure and address the 32 output channels at which digital signals, pulse-width modulated signals, pulse sequences and SENT signals can be issued.

Furthermore, this is where you can define which level is to be "inactive" and which one "active".

Four ES5321-Out-Dig-Pwm, four ES5321-Out-Multi-Pulse and 16 ES5321-Out-SENT devices can be added under this subsystem.

No settings need to be made in the "Globals" tab.

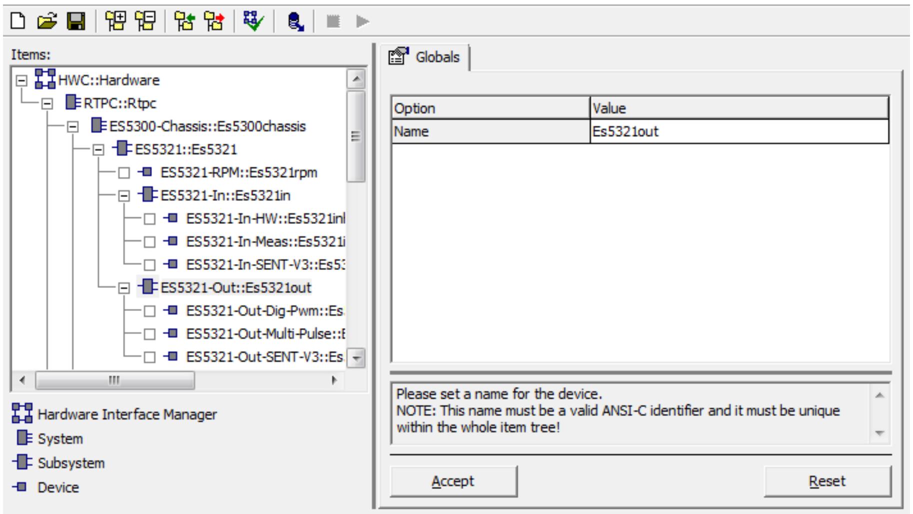


Fig. 20-21 The "Globals" Tab of the ES5321-Out Subsystem

20.9 ES5321-Out-Dig-Pwm Device

In a hardware configuration, up to four ES5321-Out-Dig-PWM subsystems can be inserted (one subsystem for each galvanically isolated output group). Each of these items controls eight output channels so that a total of 32 outputs can be configured.

20.9.1 Globals (ES5321-Out-Dig-Pwm Device)

The ES5321-Out-Dig-Pwm device is used to output PWM signals. The following figure shows the RTIO parameters of the "Globals" tab.

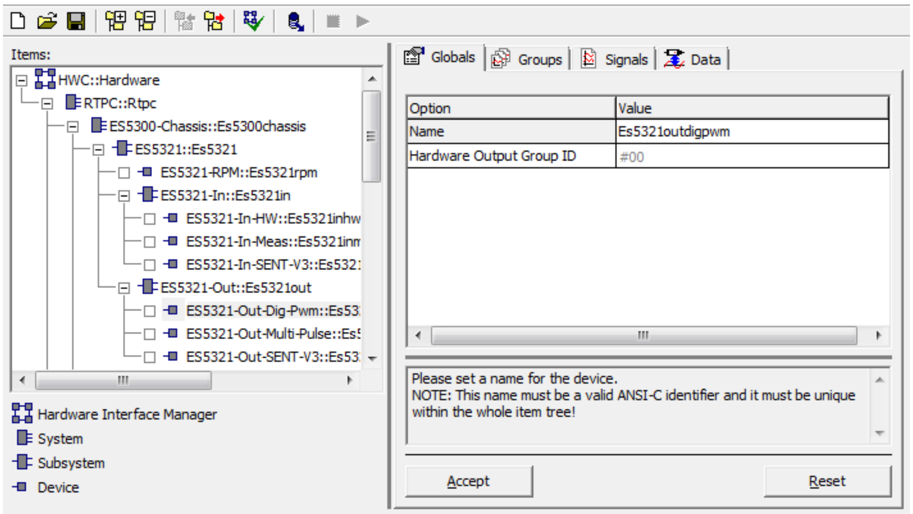


Fig. 20-22 The "Globals" Tab of the ES5321-Out-Dig-Pwm Device

Hardware Input Group

One of the four galvanically isolated input groups (0...3).

20.9.2 Signals (ES5321-Out-Dig-Pwm Device)

Fig. 20-23 shows the "Signals" tab of the ES5321-Out-Dig-Pwm device.

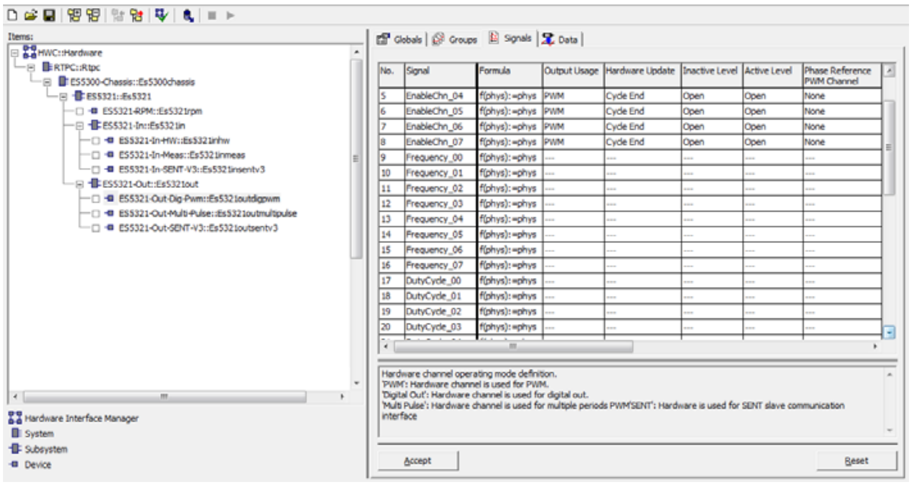


Fig. 20-23 The "Signals" Tab of the ES5321-Out-Dig-Pwm Device

Signals of send signal group "PwmOut":

Signal Name	Data Type	Notes
EnableChn_xx	bool	0: Input is not used (default) 1: Input is used No significance if a phase relationship to another channel has been established (see "Phase Reference PWM Channel" on page 634)
Frequency_xx	real32	PWM frequency [0.0 Hz...100000.0 Hz] No significance if a phase relationship to another channel has been established - in this case, the frequency of the master channel is used.
DutyCycle_xx	real32	PWM duty cycle [0.0...1.0]
Phase_xx	real32	If there is a phase relationship to another channel, the value of the phase shift can be set here [-360.0°...+360 °]
Ground_Shift_GDN	bool	Activates/deactivates a ground shift for GND
Ground_Shift_UBattA	bool	Activates/deactivates a ground shift for UBatt A
Ground_Shift_UBattB	bool	Activates/deactivates a ground shift for UBatt B

Tab. 20-7 The Signals of the ES5321-Out-Dig-Pwm Device - Signal Group "PwmOut"

Signals of receive signal group "ChnState":

Signal Name	Data Type	Notes
ChnState_xx	uint8	Each output channels has a 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver: Bit 0: Output level (0 = inactive, 1 = active) Bit 1: Lowside FET overcurrent (1 = active) Bit 2: Highside FET overcurrent (1 = active) Bit 3: Error condition detected (1 = active)

Tab. 20-8 The Signals of the ES5321-Out-Dig-Pwm Device - Signal Group "ChnState"

You can define the following options for these signals:

Output Usage

In this view, you can choose between PWM and Digital Out.

If an ES5321-Out-Multi-Pulse or an ES5321-Out-SENT device has already been added, the outputs used for this are grayed out.

- PWM

  Generation of a pulse-width modulated signal – described by frequency, duty cycle and phase relation to another PWM output
- Digital Out

  Signal levels are generated in this configuration of the output channel

Hardware Update

This list field is used to set when value changes made by the model or the user to the "Frequency_x" and "DutyCycle_x" RTIO signals (x = 0...7) are accepted by the ES5321.1 hardware.

- Immediate

  A new value is immediately visible at the output, i.e. the current pulse is cancelled and started with new values.
- Cycle End

  A new value only becomes visible at the output when the complete pulse is ended.

Inactive Level

The inactive state of the PWM signal is set in this list field. The inactive state of a PWM signal is defined as follows: if a duty cycle of 60% is set for a PWM signal, it assumes inactive signal state for 40% of the period duration.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Active Level

The active state of the PWM signal is set in this list field.

The active state of a PWM signal is defined as follows: if a duty cycle of 60% is set for a PWM signal, it assumes active signal state for 60% of the period duration.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level using an external pull-up- or pull-down resistor.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Note

The simultaneous use of battery voltage A and battery voltage B as active/inactive level is not permissible!

Phase Reference PWM Channel

The ES5321.1 makes it possible to define the setting of phase shifts between output channels with the same PWM frequency. One of the PWM outputs must be specified as a phase reference channel for this purpose. T

he number of the reference channel is then to be entered in the "Phase Reference PWM Channel" list field of the remaining outputs which have defined phase relations with this reference channel.

20.10 ES5321-Out-Multi-Pulse Device

20.10.1 Globals (ES5321-Out-Multi-Pulse Device)

A multi-pulse sequence can consist of up to eight pulses with each one being defined by a frequency and a duty cycle.

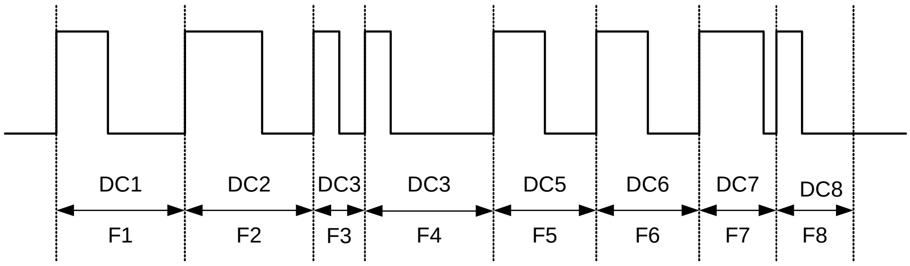


Fig. 20-24 A Sequence of Eight Pulses (Fx = Frequency of Pulse x, DCx = Duty Cycle of Pulse x)

The following figure shows the "Globals" tab of the ES5321-Multi-Pulse device.

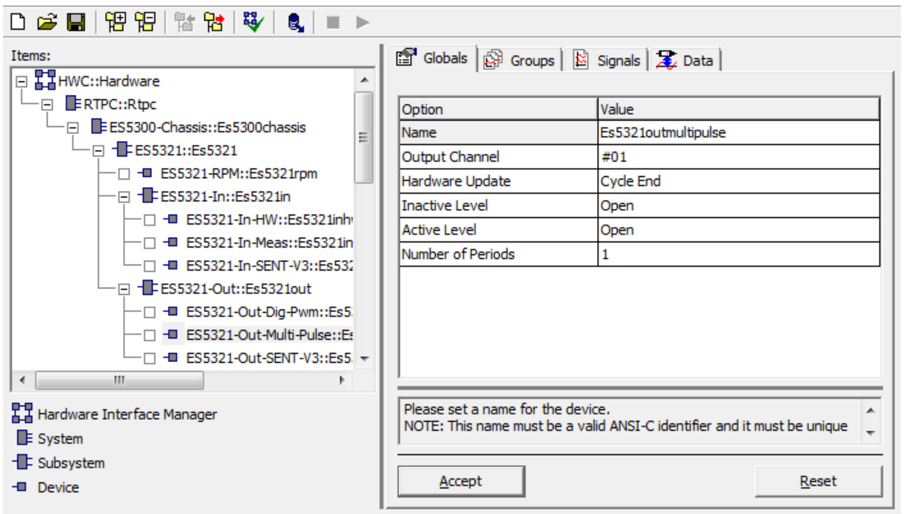


Fig. 20-25 The "Globals" Tab of the ES5321-Multi-Pulse Device

The meaning of the available options is described below.

Output Channel

The hardware channel of the ES5321.1 to be used as a multi-pulse output.

Hardware Update

This list field is used to set when changes made by the model or the user to the pulse package are accepted by the ES5321.1 hardware.

- Immediate

  A new value is immediately visible at the output, i.e. the current pulse is cancelled and started with new values.

- Cycle End

  A new value only becomes visible at the output when the complete pulse is ended.

- RTIO Controlled

  A new value becomes visible at the output after a change from "0 → 1" of the "SyncSgl" signal.

Inactive Level

The inactive state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.

- Ground

  The PWM output is at ground.

- V Batt A Out

  The PWM output outputs battery voltage A.

- V Batt B Out

  The PWM output outputs battery voltage B.

Active Level

The active state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance and can be set to a defined voltage level via an external pull-up- or pull-down resistor.

- Ground

  The PWM output is at ground.

- V Batt A Out

  The PWM output outputs battery voltage A.

- V Batt B Out

  The PWM output outputs battery voltage B.

Note

The simultaneous use of battery voltage A and battery voltage B as active/inactive level is not permissible!

Number of Periods

Number of pulses of the sequence (1..8)

20.10.2 Signals (ES5321-Out-Multi-Pulse Device)

The following figure shows the signals of the "Signals" tab

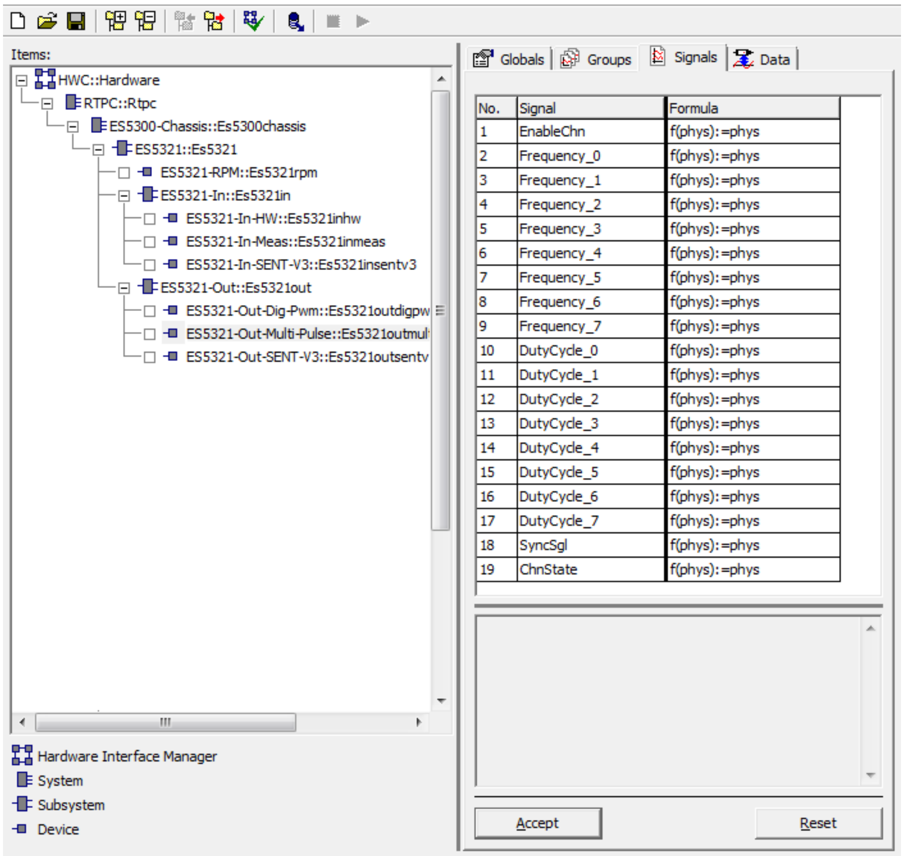


Fig. 20-26 The "Signals" Tab of the ES5321-Multi-Pulse Device

The "Signals" tab contains the following signals:

Signals of send signal group "MultiPulseOut ":

Signal Name	Data Type	Notes
EnableChn	bool	0: Input is not used (default) 1: Input is used
Frequency_xx	real32	PWM frequency [0.0 Hz...100000.0 Hz]
DutyCycle_xx	real32	PWM duty cycle [0.0...1.0]
SyncSgl	bool	If the "Hardware Update" option (see "Hardware Update" on page 635) is set to "RTIO Controlled", the pulse package data transmitted to this channel is only accepted by the hardware after a transition from 0 to 1 has been detected on the "SyncSgl" signal.

Tab. 20-9 The Signals of the ES5321-Multi-Pulse Device - Signal Group "PwmOut"

Signals of receive signal group "ChnState":

Signal Name	Data Type	Notes
ChnState	uint8	Each output channels has a 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver: Bit 0: Output level (0 = inactive, 1 = active) Bit 1: Lowside FET overcurrent (1 = active) Bit 2: Highside FET overcurrent (1 = active) Bit 3: Error condition detected (1 = active)

Tab. 20-10 The Signals of the ES5321-Multi-Pulse Device - Signal Group "ChnState"

20.11 ES5321-Out-SENT Device

20.11.1 Globals (ES5321-Out-SENT Device)

The following figure shows the "Globals" tab of the ES5321-Out-SENT device.

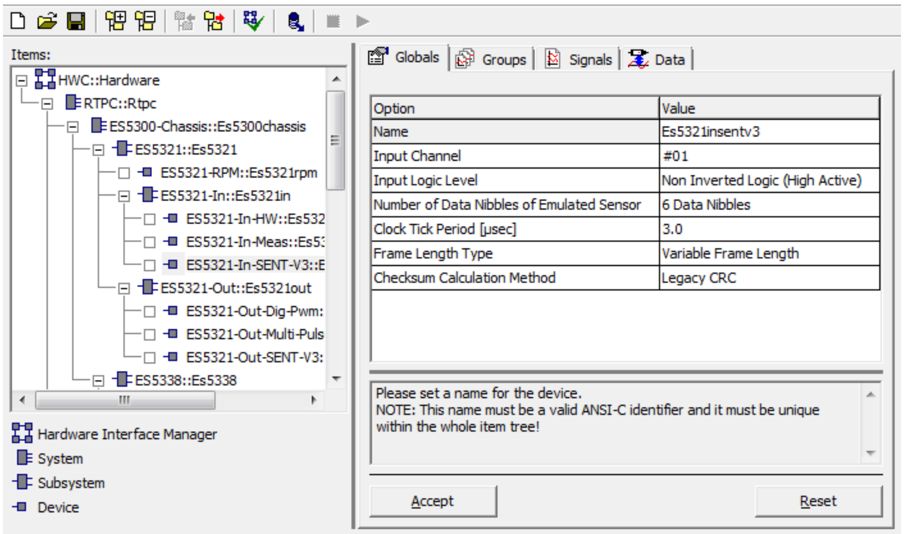


Fig. 20-27 The "Globals" Tab of the ES5321-Out-SENT Device

The following options are set in this tab:

Output Channel

Selection of the hardware channel which is to be used as the SENT transmitter.

A maximum of 16 ES5321-Out-SENT devices can be configured.

Inactive Level

The inactive state of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance.

- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Note

For a non-inverted output signal, select "Ground" for the inactive level.

Active Level

The active state (of the signal is set in this list field.

The following options are available:

- Open

  The PWM output is high-impedance.
- Ground

  The PWM output is at ground.
- V Batt A Out

  The PWM output outputs battery voltage A.
- V Batt B Out

  The PWM output outputs battery voltage B.

Note

For a non-inverted output signal, select "V Batt A" or "V Batt B" for the active level.

Note

The simultaneous use of battery voltage A and battery voltage B as active/inactive level is not permissible!

Protocol Type of Emulated Sensor

Selection of the protocol used in acc. with the SENT specification.

The following options are available:

- Without Rolling Counter

  The values of "Nibble_4" and "Nibble_5" (as shown in the "Data" tab) are considered.
- With Rolling Counter Nibble 4/5

  An 8-bit counter value transmitted with nibble 4 and nibble_5. The values of "Nibble_4" and "Nibble_5" (as shown in the "Data" tab) are not considered.
- *n* Data Nibbles

  The number of data nibbles (as defined in the protocol)

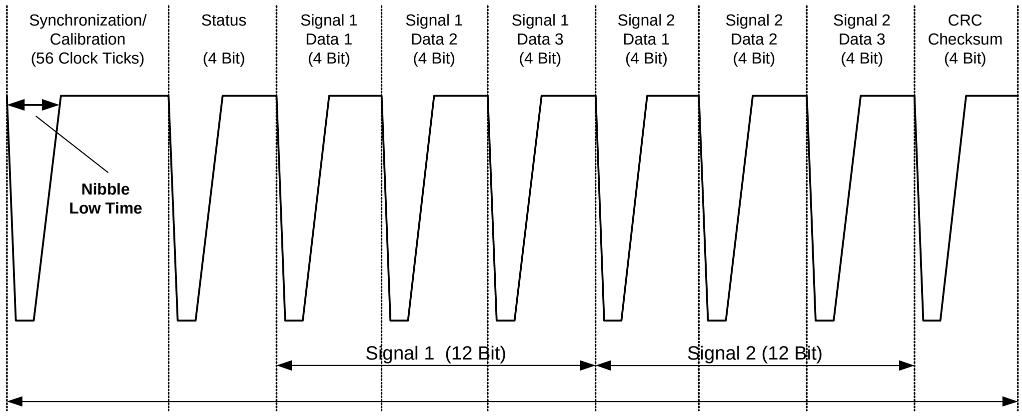
Clock Tick Period [µsec]

Length of a "clock tick", which is the smallest unit of time in the SENT protocol (1...100 µs, resolution 8 ns).

The smallest possible length with the ES5321-Out-SENT device is 1,504 µs.

Nibble Low Pulse [Ticks]

Number of ticks for the low pulse of a nibble. According to the SENT specification Jan 2010 this value should be > 4.



Frame Length Type

Transmission mode of the SENT frames.

- Variable Frame Length

  A SENT frame is expected to follow the preceding frame immediately – there is no "pause" pulse.

- Constant Frame Length

  A "pause" pulse is expected in between two SENT frames. The length of the "pause" pulse is such that the whole frame (including "pause") is of constant length.

Total Frame Lenght [Ticks]

Setting of the frame length, if "Frame Length Type" is set to "Constant Frame Length".

Checksum Calculation Method

The calculation method for the 4-bit checksum in the ES5321.1.

- Legacy CRC (0) *

  Checksum calculation according to SENT specification April 2007 and Feb 2008.

- Recommended CRC (1) *

  Checksum calculation according to SENT specification Jan 2010.

  If the calculated checksum differs from the transmitted checksum, "CheckSumErrorCounter" is incremented and the value is discarded.

- CRC = 0x0 (2) *

  The value of the CRC nibble is always "0x0".
- CRC = 0xF (3) *

  The value of the CRC nibble is always "0xF".
- CRC Mode 0…3 by Signal

  With the signal "Checksum_Mode_Data" (see "Checksum_Mode_Data" on page 642) one of the calculation methods 0...3 can be selected (any other value yields CRC = 0).
- CRC Data 0…15 by Signal

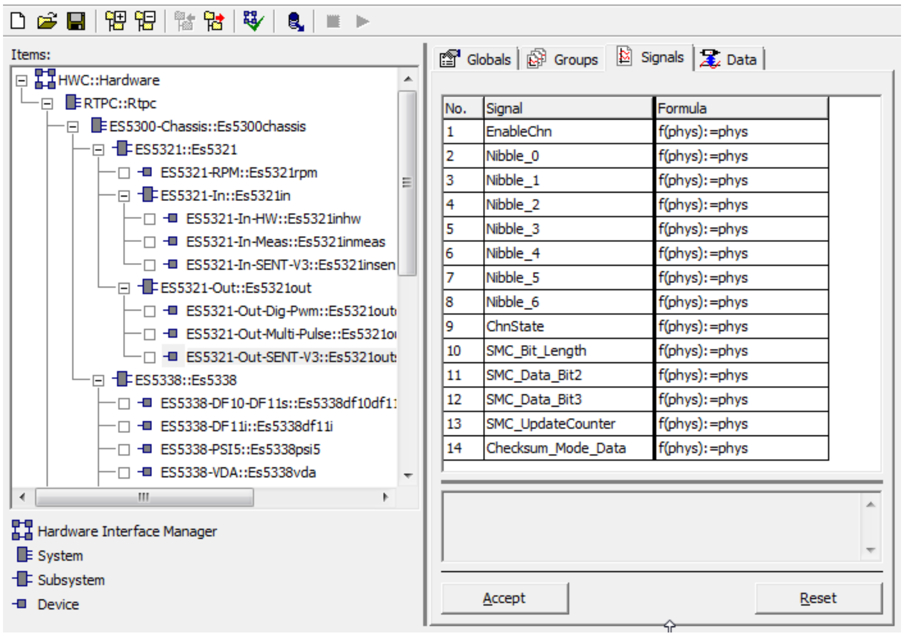
  With the signal "Checksum_Mode_Data" the CRC nibble can be set explicitely.

Note

** The number in brackets indicates the mode number for the "CRC Mode 0…3 by Signal" setting.*

20.11.2 Signals (ES5321-Out-SENT Device)

The following figure shows the "Signals" tab of the ES5321-Out-SENT device.



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

Fig. 20-28 The "Signals" Tab of the ES5321-Out-SENT Device

The "Signals" tab contains the following signals:

Signals of send signal group "SENTOut":

EnableChn

Enables or disables the hardware channel used for the SENT output

- = 0: output is high-impedance, signal output is inactive
- = 1: signal output is active

Nibble_0..Nibble_6

The value of the individual nibbles.

Checksum_Mode_Data

If "Checksum Calculation Method" (see page 640) was set to "CRC Mode 0…3 by Signal", one of the calculation methods 0...3 can be set here.

If "Checksum Calculation Method" was set to "CRC Data 0...15 by Signal", the CRC nibble can be set here.

Signal of receive signal group "ChnState":

ChnState

The 4-bit error state coding of the used hardware channel. The individual bits describe different settings and the resulting states of the output driver:

- Bit 0: Output Level (0 = inactive, 1 = active)
- Bit 1: Lowside Switch Overcurrent (1 = active)
- Bit 2: Highside Switch Overcurrent (1 = active)
- Bit 3: Error Condition Detected (1 = active)

Signals of send signal group "SentOutSmc":

SMC_Bit_Length

Input signal controlling the SMC sending unit of the ES5321.1.

"0" disables SMC sending, i.e., bit 2 and bit 3 of nibble 0 are transmitted. A value > 0 (… 31) activates the overlay of bit 2 and bit 3 of the respective data words. The value defines the number of bits for a SMC cycle. According to the SENT specification, valid values are "16" (short serial message) or "18" (enhanced serial message).

SMC_Data_Bit2

Input signal carrying the data word for bit 2 (see SAE J2716 version Jan 2010, Ch.5.2.4).

SMC_Data_Bit3

Input signal carrying the data word for bit 3 (see SAE J2716 version Jan 2010, Ch.5.2.4).

Signal of receive signal group "SentOutSmcState":

SMC_UpdateCounter

8-bit counter which increases when an SMC message was transmitted completely.

Additionally this signalizes that the ES5321.1 has started sending a new SMC message with the current values of SMC Data_Bit2 and SMC_Data_Bit3. Thus the counter can be used for synchronization purposes for the cyclic output of a list of message IDs.

The values of SMC_Data_Bit2 and SMC_Data_Bit3 are transferred at that moment, when a new short serial message is output – after data transfer SMC_Data_Bit2 and SMC_Data_Bit3 should be filled with new data. This again is communicated to the model by incrementing the value of "SMC_UpdateCounter".

The ES5321.1 and the RTIO only provides generic functionality for the reception of SMC messages. Control, decoding of the data and verification of checksums have to take place outside the RTIO.

A C code module template "SentOutSMC_Control" with basic functionality is provided. For that purpose all SMC_* signals of the RTIO are connected to the corresponding ports of the C code module.

However, for a specific application the C code module must be adapted by

- defining "SMC format mode"
- defining "SMC_Sequence" (cyclic output of a series of Msg_IDs
- providing the physical data values at the input ports
- converting the data values and assigning values to "DataField[MsgId]"

Signal Name	Data Type	Notes
EnableChn	bool	0: Input is not used (default) 1: Input is used
Nibble_xx	uint8	Nibble value
ChnState	uint8	Each output channels has a 4-bit state coding. The individual bits describe different settings and the resulting states of the output driver: Bit 0: Output level (0 = inactive, 1 = active) Bit 1: Lowside FET overcurrent (1 = active) Bit 2: Highside FET overcurrent (1 = active) Bit 3: Error condition detected (1 = active)
SMC_UpdateCounter	uint8	Counter for serial messages [0..15]
SMC_Bit_Length	uint8	Bit length of smessage [0..31], 0 = SMC inactive
SMC_Data_Bit2	uint32	Input signal containing the data word for Bit 2
SMC_Data_Bit3	uint32	Input signal containing the data word for Bit 3
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 The Signals of the ES5321-Out-SENT Device

21 ES5338.1 Carrier Board for Wheel Speed Sensor Simulation

With the ES5338.1 Carrier Board for Wheel Speed Sensor Simulation, the following types of sensors can be simulated:

- Active digital sensors with a current interface with two current levels (type DF10)
- Active digital sensors with a current interface with two current levels and additional information (type DF11i)
- Active digital sensors with a current interface with three current levels and additional information (type VDA)
- PSI5 sensor signals on the basis of PSI5 standards V1.3 and V2.1

Furthermore it is possible to directly address the current source via the simulation model ("Direct Current Output").

The Structure of the ES5338.1 RTIO Tree

In the RTIO Editor, the ES5338.1 Carrier Board for Wheel Speed Sensor Simulation is integrated by selecting an ES5338 subsystem

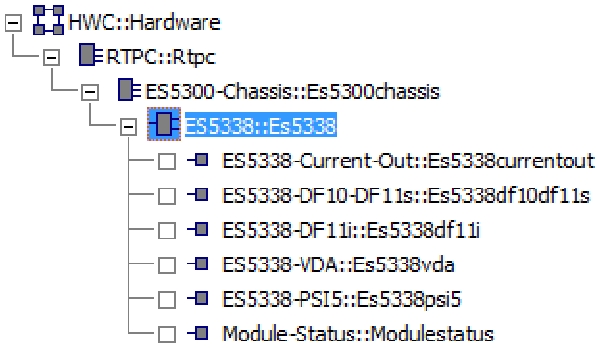


Fig. 21-1 RTIO Hardware Description with Integrated ES5338.1

An ES5300-Chassis system can contain up to ten ES5338 subsystems, of which in turn each one can accommodate six modules for sensor simulation.

21.1 ES5338 Subsystem

21.1.1 Globals (ES5338 Subsystem)

The ES5338 subsystem is used to set globally valid parameters, i.e. which have an effect on all ES5338.1 RTIO elements.

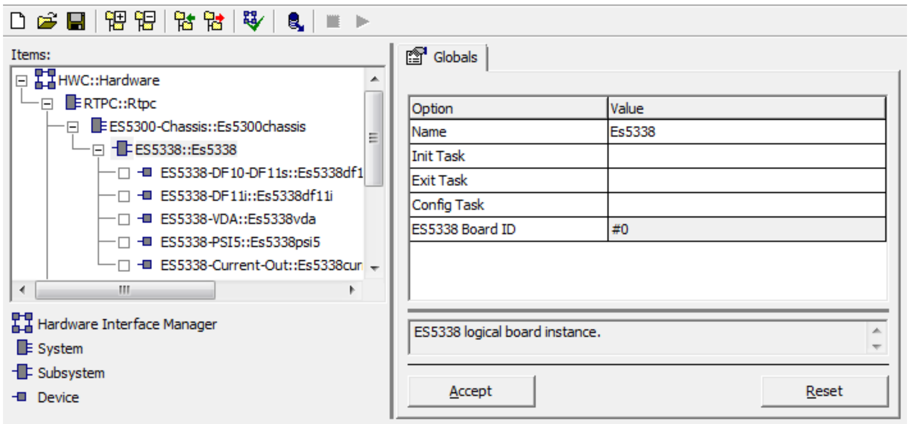


Fig. 21-2 The "Globals" Tab of the ES5338 Subsystem

Init Task, Exit Task, Config Task

The tasks for start-up, shut-down and configuration of the board.

ES5338 Board ID

This option field is used to identify the ES5338.1 – up to ten ES5338.1s can be integrated per ES5300 in the RTIO editor.

21.2 ES5338-Current-Out Device

With the ES5338-Current-Out device it is possible to directly address the current source via the simulation model.

21.2.1 Globals (ES5338-Current-Out Device)

The following figure shows the parameters of the "Globals" tab.

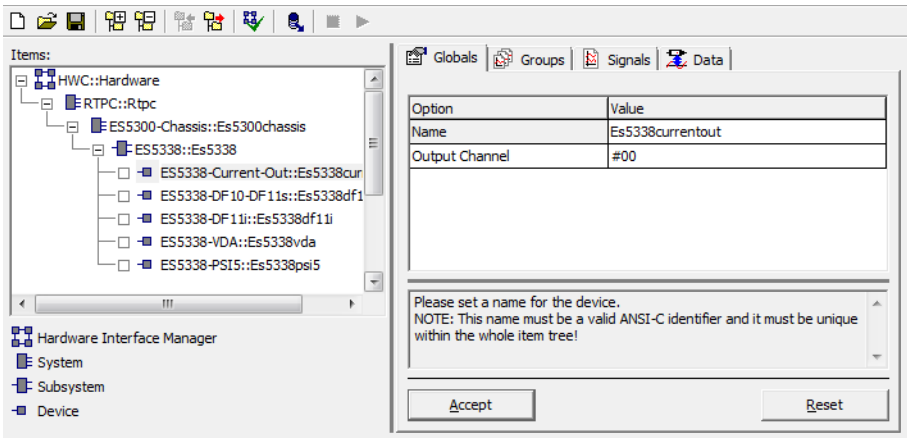


Fig. 21-3 The "Globals" Tab of the ES5338-Current-Out Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Output Channel

With this option you can specify the output to which the simulated sensor is to be switched (see "Output Channel" on page 649).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Output Channel	uint8	No	Output used [0...5]

* Data type which the RTIO driver uses internally for the parameter

Tab. 21-1 ES5338-Current-Out Device: Configuration Parameters of the "Globals" Tab

21.2.2 Signals (ES5338-Current-Out Device)

The ES5338-Current-Out device has an send signal group containing the following signals:

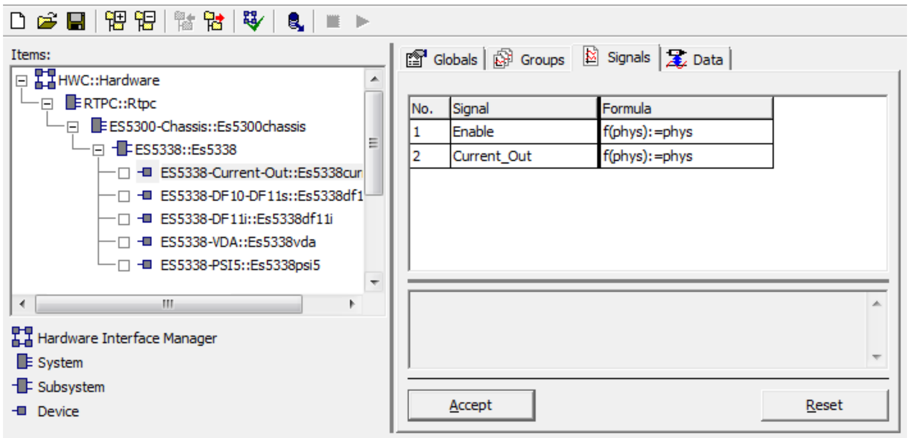


Fig. 21-4 The "Signals" Tab of the ES5338-Current-Out Device

Enable

This signal activates or deactivates the current source.

Current_Out

The output current.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Current_Out	real32	Yes	Current [0.0...120.0 mA]

* Data type which the RTIO driver uses internally for the parameter

Tab. 21-2 ES5338-Current-Out Device: Signals

21.3 Wheel Sensor Simulation – Common Settings

The RTIO elements for representing the simulated sensors have an identical structure and only differ in the specific features of the different sensors.

Output Channel

Every sensor can be assigned to an output in the "Analog Output" option field in the "Globals" tab.

Operation Mode

Use the "Operation Mode" option field to define whether the wheel speed should be specified via velocity, number of teeth, wheel radius or directly via the tooth frequency of the sensor.

Number of Teeth, Teeth Drop Pattern

Apart from the number of teeth (or pole pairs) per revolution, you can define for each tooth whether it should be output regularly or as a fault. With all defective teeth, the voltage or current value defined in the option field is output throughout the tooth. The effect of tooth faults depends on the "ToothFaultEnable" signal.

Tooth faults are defined via a proprietary input dialog whereby individual tooth numbers or tooth ranges separated by commas can be entered. Ranges are defined in the form

```
<first tooth> + <-> + <last tooth>
```

Numbering starts with 0, i.e. for a wheel with 48 teeth/pole pairs the range is 0 – 47.

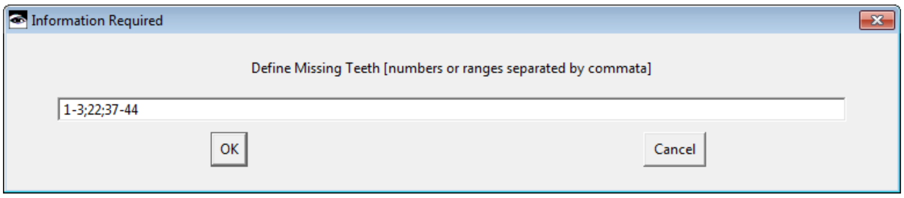


Fig. 21-5 Input Dialog for Defining Missing Teeth

21.3.1 Signals Common to all Sensors

Enable

This signal activates or deactivates the sensor. When inactive, the relevant output is high-impedance. When activated, the sensor is restarted and the output switched to the voltage (with type "DF10") or current output (all other types of sensor). When the sensor is restarted, you start with tooth 0.

Note

Please note that the output is switched using relays which is why certain switching and/or bounce times occur.

Frequency

This signal is used to directly specify the frequency of the sensor (only effective in the operation mode "Frequency", see "Operation Mode" on page 649).

With sensors which detect the direction of rotation, a negative value changes the direction of rotation (positive value: forward or normal direction of rotation, negative value: backwards). The sign is ignored by sensors which cannot detect the direction of rotation.

WheelSpeed, WheelRadius

This signal is used to specify wheel speed and wheel radius. In the operation mode "Wheelspeed", the frequency is calculated as follows from these values together with the number of teeth/pole pairs.

$$f\,[Hz] = \frac{v\,[km/h] \cdot z}{r\,[m] \cdot 2\pi \cdot 3{,}6}$$

In the operation mode "Frequency" the signals have no effect. The same is true for a negative value of wheel speed as with "Frequency", i.e. a change in the direction of rotation or rotating backwards is simulated for sensors which can detect the direction of rotation.

ToothFaultEnable

With this signal, you can enable (= 1) or disable (= 0) the settings for tooth faults made in the "Globals" tab. For every tooth or every pole pair you can determine whether an output should take place regularly or as a fault. A specified current level is output for all defective teeth of a sensor.

Accepting Signal Changes at the Outputs

Changes to frequency or wheel speed become effective when the register is written at the subsequent pole change (half tooth) at the relevant output.

Protocol information (type "DF11i" and "VDA") is not synchronized with tooth or pole changes. The values available when the register is accessed are used.

The enabling/disabling of tooth fault simulation with the "ToothFaultEnable" signal is not synchronized with a change of tooth, pole or revolution either. This could result in incomplete output of protocol or tooth information with an unfavorable switch time.

21.3.2 Globals (ES5338-DF10-DF11s Device)

With the ES5338-DF10-DF11s device you can simulate a "DF10" or "DF11s" wheel sensor.

The following figure shows the parameters of the "Globals" tab.

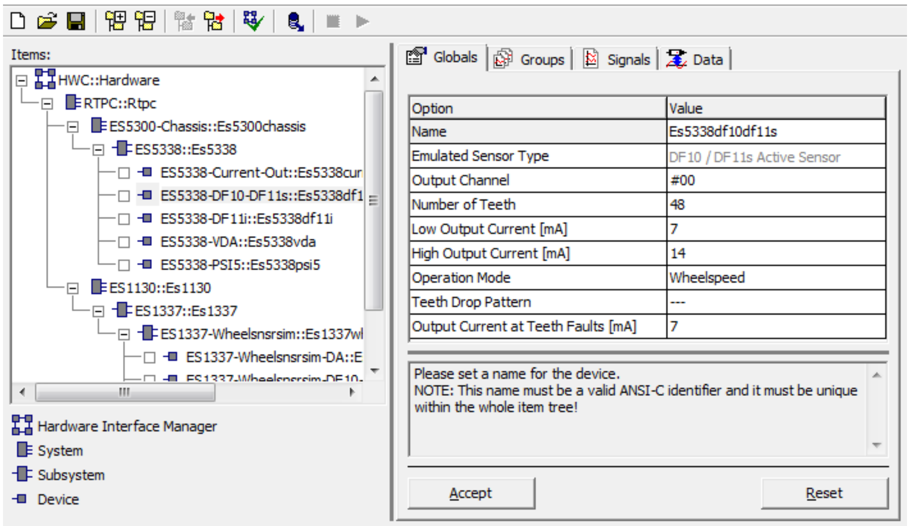


Fig. 21-6 The "Globals" Tab of the ES5338-DF10-DF11s Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Output Channel

With this option you can specify the output to which the simulated sensor is to be switched (see "Output Channel" on page 649).

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High Output Current

With this parameter you can specify the upper current value.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed (see "Operation Mode" on page 649).

Teeth Drop Pattern

With this option you can define a teeth drop pattern (see "Number of Teeth, Teeth Drop Pattern" on page 649).

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Output Channel	uint8	No	Output used [0...5]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simulated as defective
Output Current at Teeth Fault	uint8	Yes	Current value when there is a tooth fault [0...35 mA], default: 7 mA

* Data type which the RTIO driver uses internally for the parameter

Tab. 21-3 ES5338-DF10-DF11s Device: Configuration Parameters of the "Globals" Tab

21.3.3 Groups (ES5338-DF10-DF11s Device)

The ES5338-DF10-DF11s device has an "Output" signal group with which sensor simulation is controlled.

21.3.4 Signals (ES5338-DF10-DF11s Device)

The signals for controlling sensor simulation are listed in the "Signals" tab.

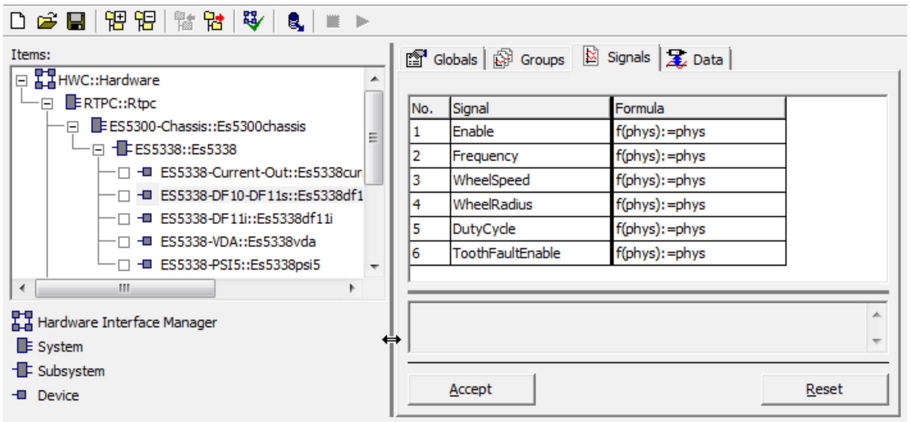


Fig. 21-7 The "Signals" Tab of the ES5338-DF10-DF11s Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-20…20 kHz] ("Frequency" mode only)
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h] ("WheelSpeed" mode only)
WheelRadius	real32	Yes	Wheel radius [0.02…2 m] ("WheelSpeed" mode only)
DutyCycle	real32	Yes	Duty cycle [0.05…0.95] ("WheelSpeed" mode only)
ToothFaultEnable	bool	Yes	Tooth fault disabling [0, 1]

* Data type which the RTIO driver uses internally for the parameter

Tab. 21-4 ES5338-DF10-DF11s Device: Signals

Note

The resulting frequency is limited to 20 kHz regardless of the operation mode – when exceeded, a corresponding error message is generated.

21.3.5 Globals (ES5338-DF11i Device)

With the ES5338-DF11i device you can simulate a "DF11i" wheel sensor with one of the four sensor outputs.

The following figure shows the parameters of the "Globals" tab.

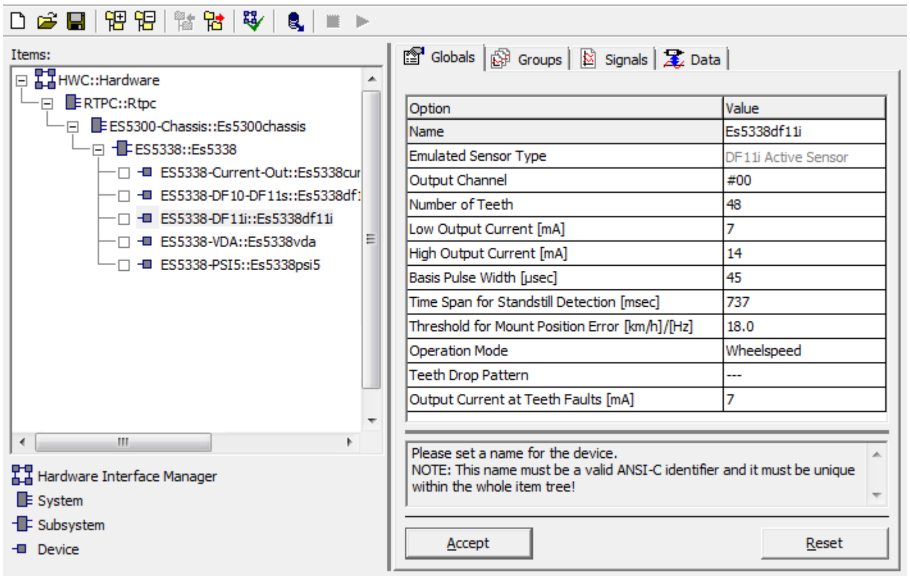


Fig. 21-8 The "Globals" Tab of the ES5338-DF11i Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Output Channel

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High Output Current

With this parameter you can specify the upper current value.

Basic Pulse Width

With this parameter you can specify the basic pulse width. Other pulse widths are derived from this value, e.g.

- Air gap limit
- Direction of rotation left
- Direction of rotation right
- Direction of rotation left and mount position
- Direction of rotation right and mount position

Time Span for Standstill Detection

With this parameter you can define the time after which the fact that the wheel is not moving is detected and the standstill pulse is output. Standstill monitoring is reset with a tooth pulse or standstill pulse, i.e. on standstill, standstill pulses are output in these intervals.

Threshold for Mount Position Error

With this option, you can define the threshold after which a position error should be output via wheel speed (operation mode "Wheelspeed") or frequency (operation mode "Frequency"). If the wheel speed or frequency is below this value, the error signal is set.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

In this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Output Channel	uint8	No	Output used [0..5]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Basic Pulse Width	uint8	Yes	Basic pulse width [37...53 µs], default: 45 µs
Time Span for Standstill Detec-tion	uint8	Yes	Time span for the detection of condition "wheel stopped" [100 ms...1000 ms], default: 737 ms
Threshold for Mount Position Error	real32	Yes	Threshold for standstill behavior [-100.0…100.0 km/h] or [-500.0…500.0 Hz]
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1 = Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simu-lated as defective
Output Current at Teeth Fault	uint8	Yes	Current value when there is a tooth fault [0…35 mA], default: 7 mA
* Data type which the RTIO driver uses internally for the parameter			

Tab. 21-5 ES5338-DF11i Device: Configuration Parameters of the "Globals" Tab

21.3.6 Groups (ES5338-DF11i Device)

The ES5338-DF11i device has an "Output" signal group with which sensor simulation is controlled.

21.3.7 Signals (ES5338-DF11i Device)

The signals for controlling sensor simulation are listed in the "Signals" tab.

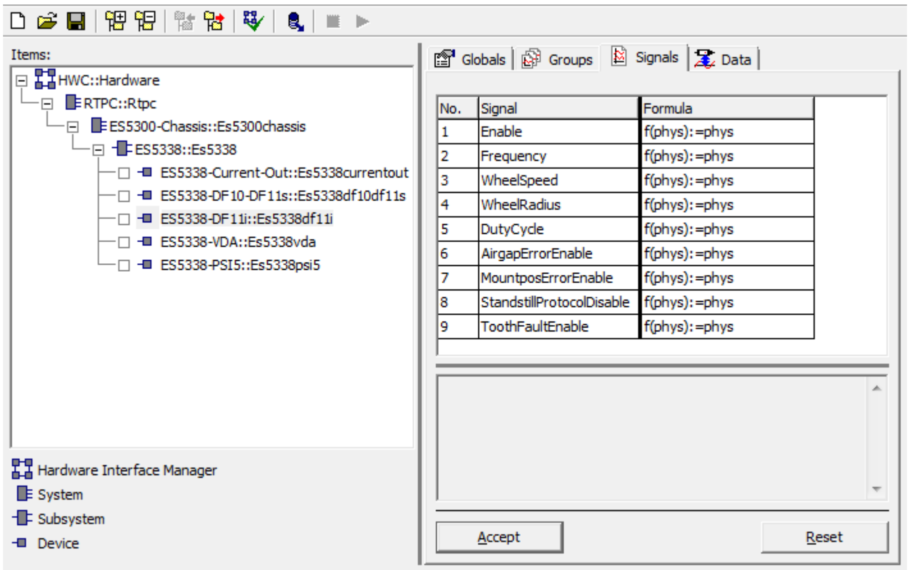


Fig. 21-9 The "Signals" Tab of the ES5338-DF11i Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

AirgapErrorEnable

Use this signal to enable/disable the "air gap error" condition.

MountposErrorEnable

Use this signal to enable/disable the "mount position error" condition.

StandstillProtocolDisable

Use this signal to enable/disable the behavior defined for this sensor on standstill.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern in the "Globals" tab.

The following table summarizes the properties of the individual signals.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1]
Frequency	real32	Yes	Frequency [-20…20 kHz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02...2 m]
DutyCycle	real32	Yes	Duty cycle [0.05…0.95]
AirgapErrorEnable	real32	Yes	Output "Air gap error" condition [0.0 = off, 1.0 = active]
MountposError-Enable	real32	Yes	Activation of position error [0.0 = off, 1.0 = active]
Standstill-ProtocolDisable	real32	Yes	Activation of standstill protocol [0.0 = off, 1.0 = active]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0 = off, 1 = active]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 21-6 ES5338-DF11i Device: Signals

Note

The resulting frequency is limited to 20 kHz regardless of the operation mode – when exceeded, a corresponding error message is generated.

21.3.8 Globals (ES5338-VDA Device)

With the ES5338-VDA device you can simulate a "VDA" wheel sensor.

The following figure shows the parameters of the "Globals" tab.

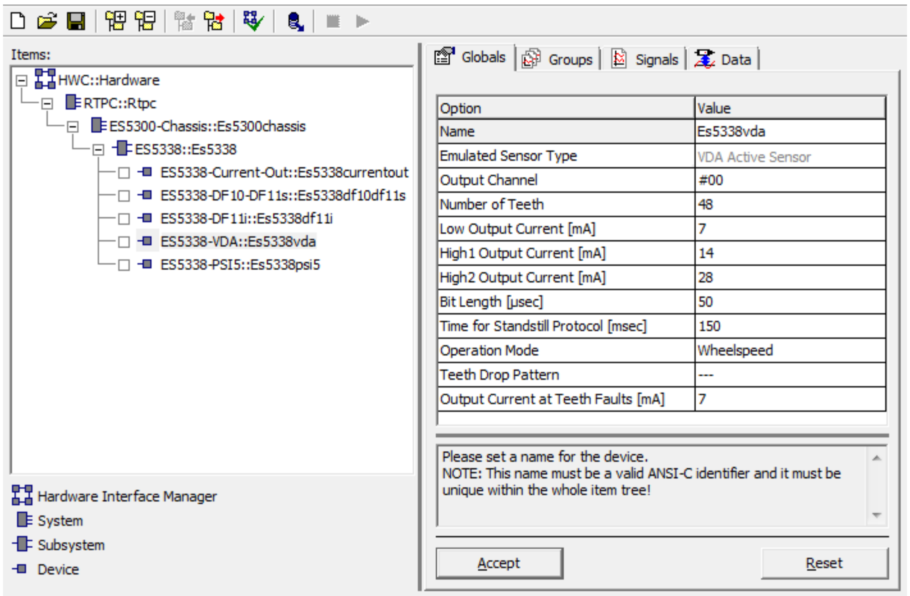


Fig. 21-10 The "Globals" Tab of the ES5338-VDA Device

Name

A unique name for this device.

Note

As this name is used for variable and parameter names, blanks and special characters are not permissible!

Output Channel

With this option you can specify the output to which the simulated sensor is to be switched.

Number of Teeth

With this parameter, you can specify the number of teeth or pole pairs.

Low Output Current

With this parameter you can specify the lower current value.

High1 Output Current

With this parameter you can specify the medium current value.

High2 Output Current

With this parameter you can specify the upper current value.

Bit Length

With this parameter you can specify the basic pulse width. Other pulse widths are derived from this value, for example

- Speed pulse
- Data pulse
- Initial bit

Time for Standstill Protocol

With this parameter you can define the time after which the fact that the wheel is not moving is detected and the standstill pulse is output. Standstill monitoring is reset with a tooth pulse or standstill pulse, i.e. on standstill, standstill pulses are output in these intervals.

Operation Mode

With this option you can define the operation mode for specifying the wheel speed.

Teeth Drop Pattern

In this option you can define a teeth drop pattern.

Output Current at Teeth Faults

With this parameter you can define which current value should be output with a tooth defined as being defective.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	**Data Type***	**Can be edited online**	**Comment/Value Range**
Name	Identi-fier	No	User-specific name
Output Channel	uint8	No	Output used [0...5]
Number of Teeth	uint8	Yes	Number of teeth/pole pairs [1…128], default: 48
Low Output Current	uint8	Yes	Lower current value [0…35 mA], default: 7 mA
High Output Current	uint8	Yes	Upper current value [0…35 mA], default: 14 mA
Bit Length	uint8	Yes	Basic pulse width [20…80 µs], default: 50 µs
Time for Standstill Protocol	uint16	Yes	Time span for the detection of condition "wheel stopped" [50 ms...255 ms], default: 150 ms
Operation Mode	uint8	Yes	Operation mode for specifying speed change [0 = Wheelspeed | 1= Frequency]
Teeth Drop Pattern	Curve	Yes	List of the teeth/pole pairs or tooth/pole pair ranges to be simulated as defective, default: No tooth fault
Output Current at Teeth Faults	uint8	Yes	Current value when there is a tooth fault [0…35 mA], default: 7mA

* Data type which the RTIO driver uses internally for the parameter

Tab. 21-7 ES5338-VDA Device: Configuration Parameters of the "Globals" Tab

21.3.9 Groups (ES5338-VDA Device)

The ES5338-VDA device has an "Output" signal group with which sensor simulation is controlled.

21.3.10 Signals (ES5338-VDA Device)

The signals for controlling sensor simulation are listed in the "Signals" tab.

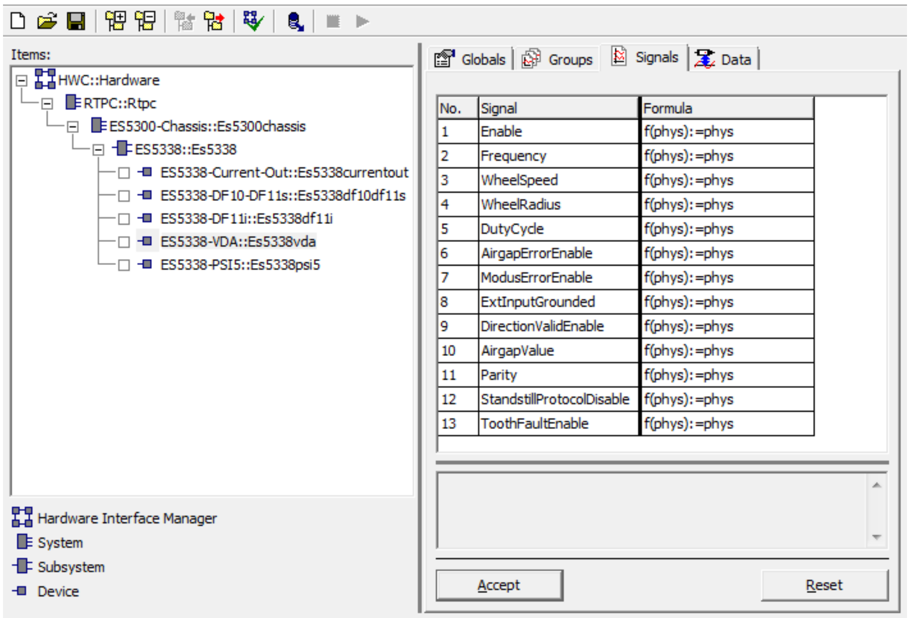


Fig. 21-11 The "Signals" Tab of the ES5338-VDA Device

Enable

Use this signal to enable/disable the output.

Frequency

Use this signal to directly specify the tooth frequency (with operation mode "Frequency").

WheelSpeed, WheelRadius

Use these signals to specify the wheel speed/wheel radius (with operation mode "Wheelspeed").

DutyCycle

Use this signal to define the duty cycle of the two poles of a tooth. A duty cycle of 0.5 results in a symmetrical output signal.

AirgapErrorEnable

Use this signal to enable/disable the "AirgapError" condition (protocol bit 0).

ModusErrorEnable

Use this signal to enable/disable the "ModusError" condition (protocol bit 1).

ExtInputGrounded

Use this signal to influence the corresponding protocol bit 2. A zero indicates "External input open".

DirectionValidEnable

Use this signal to influence the corresponding protocol bit 3. A zero indicates that the direction of rotation transferred is valid.

AirgapValue

Use this signal to influence the value of the corresponding protocol bits 5..7 . The air gap value is in a range between 0 and 7.

Parity

Use this signal to influence the parity creation (protocol bit 8). A value of zero corresponds to even parity.

The meaning of the protocol bits is summarized in the following table.

Bit	Meaning	Coding
0	Error bit air gap limit	0 = correct 1 = air gap limit
1	Can be assigned freely	
2	Can be assigned freely	
3	Validity of direction of rotation	0 = valid 1 = invalid
4	Direction of rotation – is calculated from the sign of "WheelSpeed" or "Frequency"	0 = positive 1 = negative
5	Can be assigned freely	
6	Can be assigned freely	
7	Can be assigned freely	
8	Parity bit	Is set to 0 or 1 to retain even parity (incl. the parity bit itself).

Tab. 21-8 Meaning of the Nine Protocol Bits

StandstillProtocolDisable

Use this signal to influence the standstill protocol behavior of the sensor.

ToothFaultEnable

Use this signal to enable/disable the tooth fault pattern defined in the "Globals" tab.

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Sensor enabling [0, 1], default: 1
Frequency	real32	Yes	Frequency [-20…20 kHz]
WheelSpeed	real32	Yes	Wheel speed [-1000.0…1000.0 km/h]
WheelRadius	real32	Yes	Wheel radius [0.02...2 m], default: 0.32 ,
DutyCycle	real32	Yes	Duty cycle [0.05…0.95], default: 0.5
AirgapErrorEnable	real32	Yes	Activation of "air gap error" [0.0 = off, 1.0 = active]
ModusErrorEnable	real32	Yes	Activation of "modus error" [0.0 = off, 1.0 = active]
ExtInputGrounded	real32	Yes	Level of ext. input [0.0 = open, 1.0 = ground]
DirectionValidEnable	real32	Yes	Indicates direction of rotation valid [0.0 = direction of rotation valid, 1.0 = direction of rotation invalid]
AirgapValue	real32	Yes	Air gap value [0…7]
Parity	real32	Yes	Setting parity [0.0 = even, 1.0 = odd]
Standstill-ProtocolDisable	real32	Yes	Activation of standstill protocol [0.0 = active, 1.0 = inactive]
ToothFaultEnable	bool	Yes	Tooth fault disabling [0 = off, 1 = active]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 21-9 ES5338-VDA Device: Signals

Note

The resulting frequency is limited to 20 kHz regardless of the operation mode – when exceeded, a corresponding error message is generated.

21.4 ES5338-PSI5 Device

The ES5338-PSI5 device is used to configure the output of the PSI5 signals.

21.4.1 Globals (ES5338-PSI5 Device)

In the "Globals" tab, the following settings can be made:

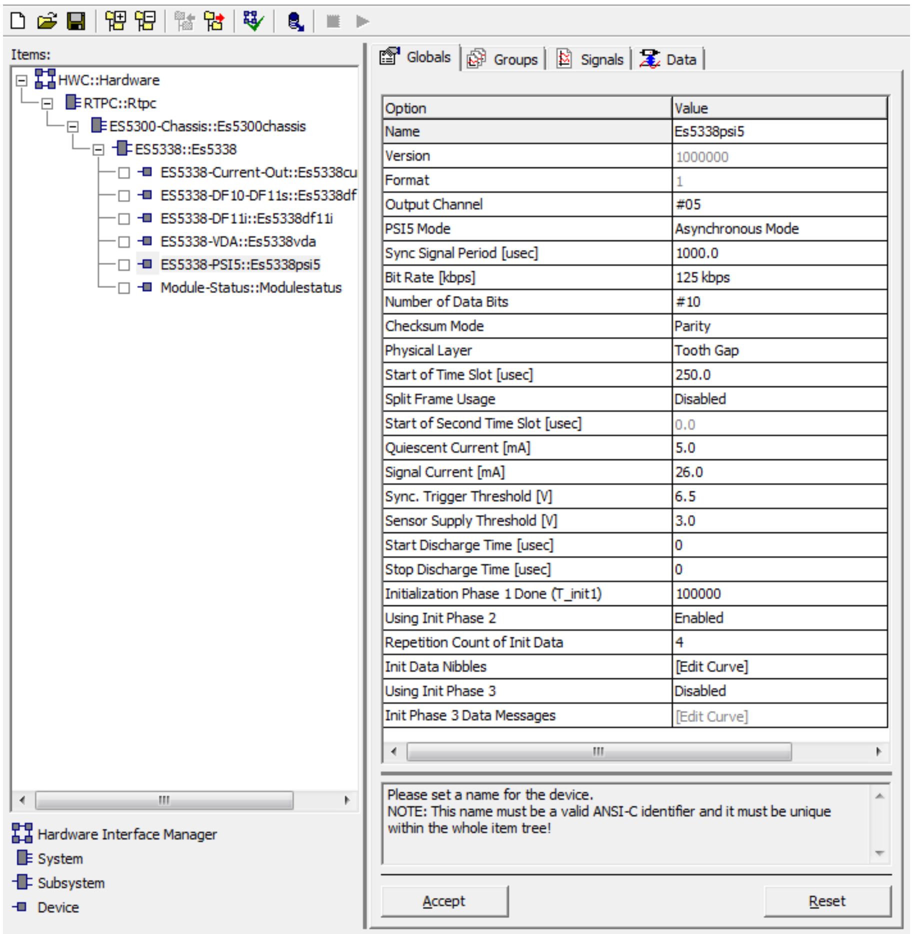


Fig. 21-12 The "Globals" Tab of the ES5338-PSI5 Device

Output Channel

With this option you can specify the output to which the PSI5 signal is to be switched.

PSI5 Mode

The communication mode of the sensor:

- Asynchronous Mode
- Sync. Parallel Bus Mode
- Sync. Universal Bus Mode

Sync Signal Period [µsec]

Defines the maximum period length of the Sync signal (0…6553 µs, resolution 0.1 µs).

Bit Rate [kpbs]

Bit rate of the sensor (125 kbps, 189 kbps)

Number of Data Bits

Defines the number of data bits of the PSI5 sensor (10...28).

Checksum Mode

Defines the method for calculating the checksum (Parity, CRC).

Physical Layer

Defines the communication method between sensor and control units on the physical layer (Tooth Gap, Pulse Width).

Start of Time Slot [µsec]

Defines the time (after the Sync pulse has been received) from which the sensor simulation sends the PSI5 data.

Split Frame Usage

Enables/disables "Split Frame" mode.

The reference data is divided into two data packages: For example, 16 bits of reference data can be split into 2 x 8 bits and sent in frames with 10 data bits.

Start of Second Time Slot [µsec]

Defines the time (after the Sync pulse has been received) from which the PSI5 data of the second frame is sent.

Quiescent Current [mA]

Defines the quiescent current or the normal current consumption of a PSI5 sensor (= the logical "0" in sensor ECU communication).

Signal Current [mA]

The increased current consumption when a logical "1" is represented in sensor ECU communication.

Sync Trigger Threshold [V]

Defines the voltage value from which the hardware recognizes a voltage pulse from the ECU as such (see Fig. 21-13 on page 668).

Sensor Supply Threshold [V]

Defines a threshold from which the ES5338.1 accepts the supply voltage as valid. Range: 0...60 V, typical value: supply voltage/2 (see Fig. 21-13).

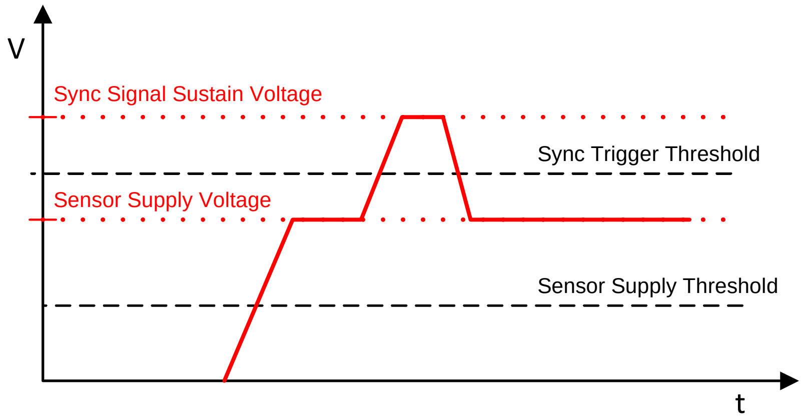


Fig. 21-13 Typical Signal Trace with Threshold Voltages

Start Discharge Time

Defines the starting time of an increased current consumption of the PSI5 sensor.

Only visible if **View → Show All** was selected.

Stop Discharge Time

Defines the end of an increased current consumption of the PSI5 sensor.

Only visible if **View → Show All** was selected.

Initialization Phase 1 Done (T_init1)

Defines the time at which initialization phase 1 is concluded and the sensor can start sending Init Phase 2 data nibbles.

Using Init Phase 2

Enables/disables the "Init Phase 2" feature.

If this parameter is disabled, the "Repetition Count" and "Init Data Nibbles" parameters are grayed out.

Repetition Count of Init Data

Defines how often the Init Phase data nibbles are sent.

Init Data Nibbles

Table for storing the Init Phase data nibbles. The number of entries in this table corresponds to the number of Init Phase data nibbles.

Using Init Phase 3

Defines whether an "Init Phase 3" is to be used or not.

Init Phase 3 Data Messages

At the end of Init Phase 2, a sensor returns either "OK" (in the form of special data packages) or an error code. If an Init Phase 3 is used, the data can be defined with it.
The following table summarizes the configuration parameters:

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Name	Identi-fier	No	User-specific name
Output Channel	uint8	No	The output used
PSI5 Mode	uint8	Yes	Communication mode: - Asynchronous Mode - Sync. Parallel Bus Mode - Sync. Universal Bus Mode Default: Asynchronous Mode
Sync Signal Period	real32	Yes	Maximum period duration of the Sync signal [0…6553 µs], Resolution: 0.1 µs Default: 1000.0 µs
Bit Rate [kpbs]	uint8	Yes	Bit rate of the sensor [125 kbps, 189 kbps] Default: 125 kbps
Number of Data Bits	uint8	Yes	Number of data bits [10...28], Default: 10 bits
Checksum Mode	uint8	Yes	Calculation of the checksum [CRC, Parity], Default: Parity
Physical Layer	uint8	Yes	Communication with the ECU on the physical layer [Tooth Gap, Pulse Width] Default: Tooth Gap
Start of Time Slot	real32	Yes	Start of time slot [0…6553 µs], Resolution: 0.1 µs Default: 250.0 µs
Split Frame Usage	uint8	Yes	Use of "Split Frame" mode [Enable, Disable] Default: Disable
Start of Second Time Slot	real32	Yes	Start of second time slot [0…6553 µs], Resolution: 0.1 µs Default: 0.0 µs
Quiescent Current	real32	Yes	Quiescent current of the sensor [0… mA] (no upper limit) Default: 10 mA

Parameter or Option Field	**Data Type***	**Can be edited online**	**Comment/Value Range**
Signal Current	real32	Yes	Current consumption when dis-playing a logical "1" [0… mA] (no upper limit) Default: 26 mA
Sync Trigger Threshold	real32	Yes	Threshold for pulse detection [0…60 V] Default: 3.5 V
Sensor Supply Threshold	real32	Yes	Threshold for supply voltage [0…60 V] Default: 2.0 V
Start Discharge Time *	real32	Yes	Start of increased current consumption [0… µsec] (no upper limit) Default: 0 µsec
Stop Discharge Time *	real32	Yes	End of increased current consumption [0… µsec] (no upper limit) Default: 0 µsec
Initialization Phase 1 Done (T_init1)	real32	Yes	End of "Initialization Phase 1" [0… µsec] (no upper limit) Default: 100000 µsec
Using Init Phase 2	uint8	Yes	Enables/disables Initialization Phase 2 of the PSI5 sensor [Enable / Disable] Default: Enabled
Repetition Count of Init Data	real32	Yes	Repetitions "Sending initializa-tion data" [0…255] Default: 4
Init Data Nibbles	Table	Yes	Are entered as curves – every point represents a data nibble.
Using Init Phase 3	uint8	Yes	Enables/disables Initialization Phase 3 of the PSI5 sensor [Enable / Disable] Default: Disabled
Init Phase 3 Data Messages	Table	Yes	Contains information on Init Phase 3 data messages
The options shown with a * are only visible after selecting **View → Show All**			

Tab. 21-10 ES5338-PSI5 Device: Configuration Parameters of the "Globals" Tab

21.4.2 Signals (ES5338-PSI5 Device)

The ES5338-PSI5 device has the following signals:

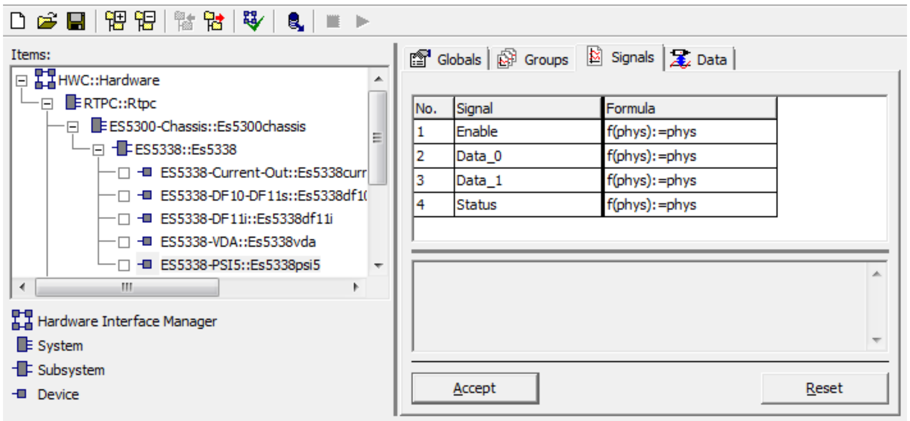


Fig. 21-14 The "Signals" Tab of the ES5338-PSI5 Device

Signals of the "Send" signal group :

Parameter or Option Field	Data Type*	Can be edited online	Comment/Value Range
Enable	bool	Yes	Activates/deactivates the output, [0,1] default: 1 (= output active)
Data_0	real64	Yes	Data register 0 of the PSI5 sensor
Data_1	real64	Yes	Data register 1 of the PSI5 sensor ("Split Frame" mode only)
* Data type which the RTIO driver uses internally for the parameter			

Signals of the "Receive" signal group:

Parameter or Option Field	Data Type*	Comment/Value Range
Status	real64	Error and status bits of the PSI5 sensor. **Status Bits Bit 0....Bit 2**: 000**000**: Sensor unpowererd / reset 000**001**: Sensor in "Init Phase 1" 000**010**: Sensor in "Init Phase 2" 000**011**: Sensor in "Init Phase 3" 000**100**: Sensor running **Error Bits Bit 4...Bit 5**: **01**0000: Overvoltage detected **10**0000: Overload detected, output current reduced.
* Data type which the RTIO driver uses internally for the parameter		

22 ES5340.1/.2 Electric Drive Simulation Board

The ES5340.1/.2 Electric Drive Simulation Board with a PCI express interface is used to test ECUs for inverters/electric motors at signal level and can be used as a multi I/O board for testing in other domains as power train or battery management.

For more details on functions and electric drive simulation, please refer to the ES5340.1/.2 Electric Drive Simulation Board User's Guide.

Structure of the RTIO Hierarchy

The structure of the RTIO hierarchy in the RTC Editor as well as the configuration and real-time data of the individual items are described in more detail below.

To create a new LABCAR Software hierarchy in the LABCAR RTC Editor, add an ES5340-Hybrid subsystem at RTPC system level. This RTIO element is used as a container for the description of a master and also, if required, a slave board and does not offer any other settings.

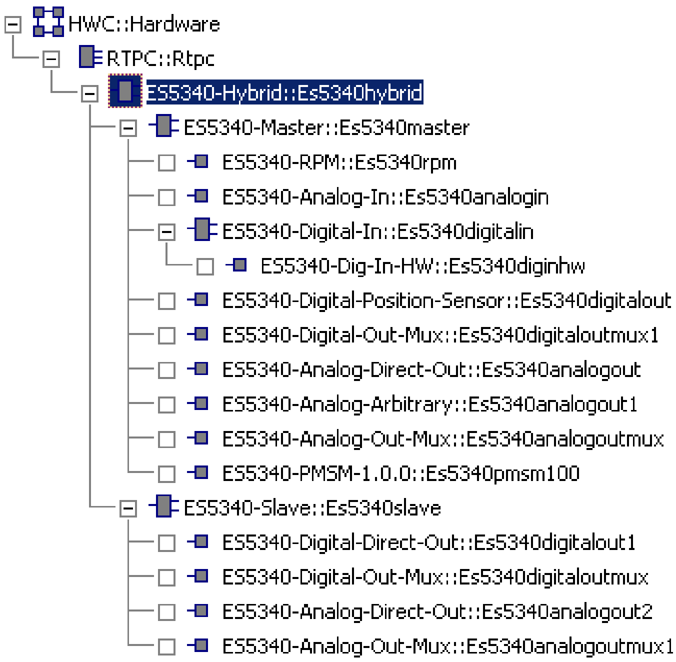


Fig. 22-1 Complete ES5340-Hybrid RTIO Tree with Master and Slave Board

Electric Drive Simulation Master Board (ES5340.1/.2-M)

This board offers the entire function scope:

I/O:

- 8 analog outputs (incl. simulation of analog position sensor signals)
- 8 digital outputs (incl. simulation of digital position sensor signals)
- 20 digital inputs
- 4 analog inputs

Model support:

- FPGA model available
- RTPC model possible

To be able to use the master board, add an LABCAR Software-Master subsystem (see "ES5340-Master" on page 676) in the LABCAR RTC Editor at LABCAR Software-Hybrid subsystem level.

Electric Drive Simulation Slave Board (ES5340.1/.2-S)

In comparison to the master board, the slave board has a reduced function scope:

I/O:

- 6 analog outputs (without simulation of analog position sensor signals)
- 6 digital outputs (without simulation of digital position sensor signals)

The source of the output signals are the "Analog-/Digital-Direct-Out" outputs of the slave board and especially the values of an FPGA model of the ES5340.1/.2-M.

To be able to use the slave board, add an ES5340-Slave subsystem (see "ES5340-Slave" on page 740) in the LABCAR RTC Editor at LABCAR Software-Hybrid subsystem level.

Signal Generators in the Project and in the RTIO Hardware Driver

The following table contains an overview of the number of available signal generators depending on the ES5340/FPGA variant:

ES5340/ FPGA model	Number of analog + digital signal generators	Number of PWM generators	Number of analog + digital multiwave signal generators
Electric Drive Simulation (without FPGA model) – FPGAs before V1.0.12	8 + 8	0	0
Electric Drive Simulation (without FPGA model) – FPGA V1.0.12 after 9.3.2011	8 + 8	8	0
PMSM E-Motor V1.0	4 + 4	0	0
IM E-Motor V1.0	4 + 4	0	0
FlexibleFPGA V1.0	8 + 8	0	0
FlexibleFPGA V2.0	n + m (n, m = 0…8)	m (m = 0…8)	0
ICE (Internal Combustion Engine)	8 + 8 (ES1335-compatible)	8	4 + 4

Tab. 22-1 ES5340M-FPGA Variants and Available Signal Generators

LABCAR-RTC can only check the total maximum possible number of signal generators – the ES5340 driver then checks which and how many generators are actually available in the current FPGA.

If a LABCAR-OPERATOR project requires a specific resource which is not available in the FPGA, the following generic error message is output for all variants of the ES5340 hardware (the example shows the message for PWM generators):

```
[RTIO] 'ES5340-Master' Inst: 0 ('/dev/es5340m_0', handle=0):
Unsuitable Board Firmware. Resource 'PwmGen #0' used by 'PWM
Generator' is not implemented in HDC!
```

The strings `'...'` differ accordingly for arbitrary generators.

22.1 ES5340-Master

The ES5340-Master subsystem is used to configure an ES5340-Master board.

22.1.1 Globals (ES5340-Master Subsystem)

This is where you make global settings for the ES5340-Master board.

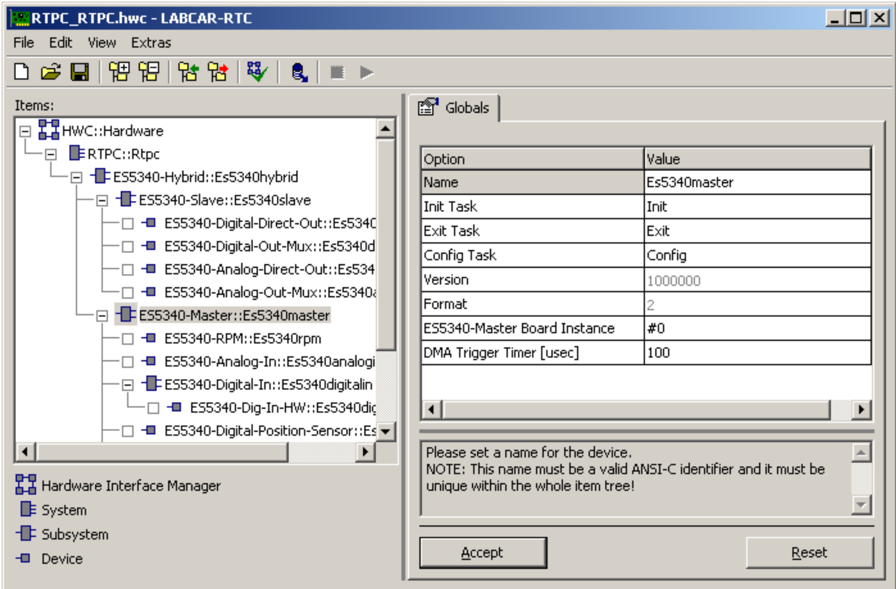


Fig. 22-2 The "Globals" Tab of the ES5340-Master Subsystem ("Show All" Option)

ES5340-Master Board Instance

This option field is used to specify the board instance assigned to this RTIO subsystem. Numbering starts at 0 and the boards of any one type are numbered (as with the VMEbus system) in ascending order from left to right[1].

For this purpose, however, the internal angle clock bus (SYNC interface) must be connected via all boards (see User's Guide) – the board instance cannot be changed during the experiment.

DMA Trigger Time [µsec]

This field is only displayed if "Show All" was selected in the shortcut menu (right mouse button).

This is where you can select the interval for triggering two asynchronous DMA transfers – the data measured is written directly to the PC main memory by the LABCAR Software-Master board.

This concerns the last four time stamps of all 20 digital inputs, the current measure values of the four analog values, the RTIO input values of the RPM unit and when using an FPGA model its RTIO input values. This data can be changed during simulation by reconfiguring.

- Default value: 100 µs
- Setting range: 5 µs … 1 s

1. Only applies if the boards are connected via the SYNC-Bus lines. Otherwise, numbering takes place either by chance or depends on the hardware.

22.2 ES5340-RPM – RPM Unit

The angle-based synchronization between LABCAR boards takes place via an angle clock signal consisting of three components (Fig. 22-3). The "ac_{SYNC}" signal displays a zero passage of the crankshaft angle in a combustion engine or of the rotor angle in an electric motor, "ac_{CLK}" contains the actual angle clock, "ac_{DoR}" specifies the direction of rotation of the engine.

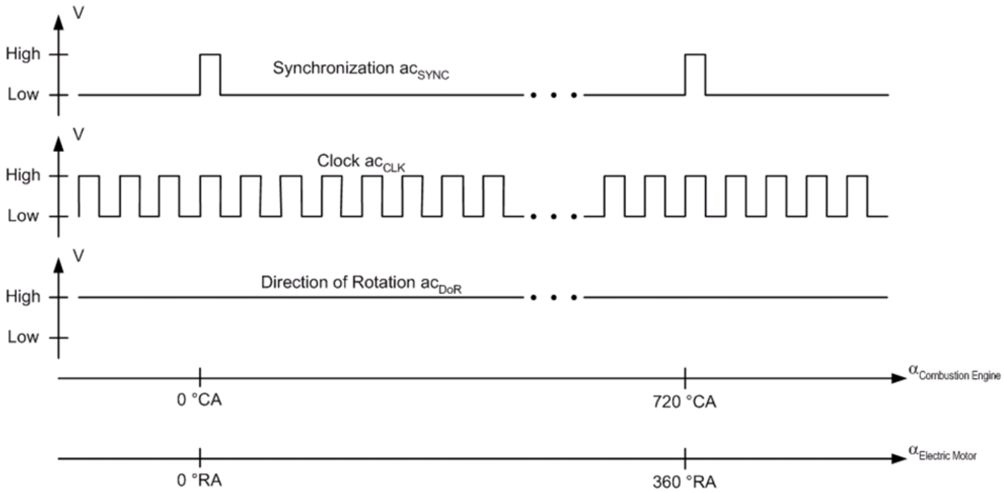


Fig. 22-3 The Three Components of the Angle Clock Signal

The clock signal offers a resolution δ_α of 16 bits per 720 °CA of the combustion engine or 360 °RA of the electric motor.

$$\delta_{\alpha,\ CombustionEngine} = \frac{720\,°CA}{2^{16}} = 0{,}011°CA$$

Equ. 22-1

$$\delta_{\alpha,\ EMotor} = \frac{360\,°RA}{2^{16}} = 0{,}011°RA$$

Equ. 22-2

The RPM unit of the LABCAR Software can be operated in slave and master mode. In slave mode, the RPM unit is synchronized to an angle clock generated by another board in the system. In master mode, the RPM unit generates the angle clock for other boards in the system. Two buses are available for transferring the angle clock between the boards. There can only be one master at a bus.

The control and status signals of the RPM unit depend on the engine type simulated. When combustion engines are simulated, the engine speed n (in revolutions per minute) is the input value of the RPM unit. It is only relevant if the RPM unit is configured for master mode. The current engine position α (in degrees crankshaft) is supplied as output value.

Signal	**Direction**	**Description**
n	Input	Engine speed in revolutions per minute
α	Output	Engine angle in degrees crankshaft

Tab. 22-2 Control and Status Signals of the RPM Unit in the Simulation of Combustion Engines

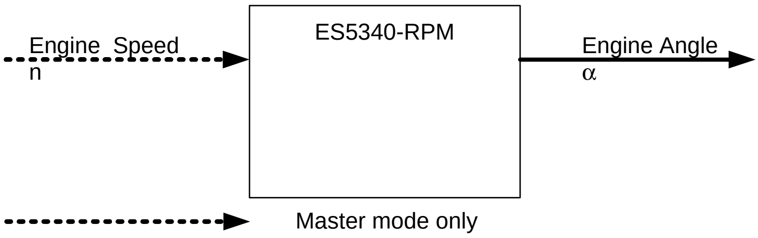


Fig. 22-4 Control and Status Signals of the RPM Unit in the Simulation of Combustion Engines

The input and output values in the simulation of electric motors are listed in Tab. 22-3 and shown in Fig. 22-5. In addition to the current engine position α, the value of an angle phase-shifted with respect to the engine angle α_{PS} is output. The phase $\Delta\phi$ relative to the engine angle as well as the phase change speed $\Omega_{\Delta\phi}$ are controlled by relevant inputs of the RPM unit.

Signal	**Direction**	**Description**
n	Input	Engine speed in revolutions per minute
$\Delta\phi$	Input	Phase of the phase-shifted mechanical angle in °RA
$\Omega_{\Delta\phi}$	Input	Change speed in °RA per second of the phase-shifted mechanical angle when the target value changes
α	Output	Rotor position or mechanical angle in °RA
α_{PS}	Output	Phase-shifted mechanical angle in °RA

Tab. 22-3 Control and Status Signals of the RPM Unit in the Simulation of Electric Motors in Simple Master Mode

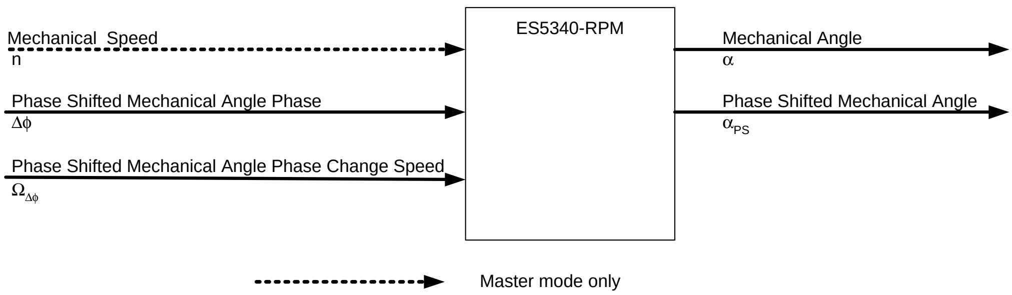


Fig. 22-5 Control and Status Signals of the RPM Unit in the Simulation of Electric Motors in Simple Master Mode

In addition to simple master mode, the RPM unit offers a further operating mode for the simulation of electric motors – the "master mode with angle adaption". The functions of this operating mode are explained below, based on Fig. 22-6 on page 680.

This figure shows the basic procedure in an ECU test using a LABCAR. At the beginning of a simulation cycle, the LABCAR hardware reads the measure data into the simulation model. This data is used to calculate the vehicle model and, once the calculation is completed, the computed control values are output to the LABCAR hardware.

Unlike simple master mode, in which only engine speed is added as a control value of the RPM unit, the model calculates an additional angle value and transfers it to the RPM unit in "master mode with angle adaption".

This angle value ($\alpha_T(t_1)$, $\alpha_T(t_2)$, $\alpha_T(t_3)$ etc.) represents the angle the engine should have at the beginning of the next simulation cycle (times t_1, t_2, t_3 etc.).

In reality, the actual engine angle can, or rather will, deviate from this target value (shown as $\alpha_{HW}(t_1)$). This is why the RPM unit calculates an RPM additive internally that offsets this angle difference by the start of the next simulation cycle (t_2 in this case):

$$n_{HW} = n_T + \frac{\alpha_T - \alpha_{HW}}{6\Delta t}$$

Equ. 22-3

$$\Delta n = \frac{\alpha_T - \alpha_{HW}}{6\Delta t}$$

Equ. 22-4

with n in revolutions per minute, α in degrees rotor angle, Δt in seconds.

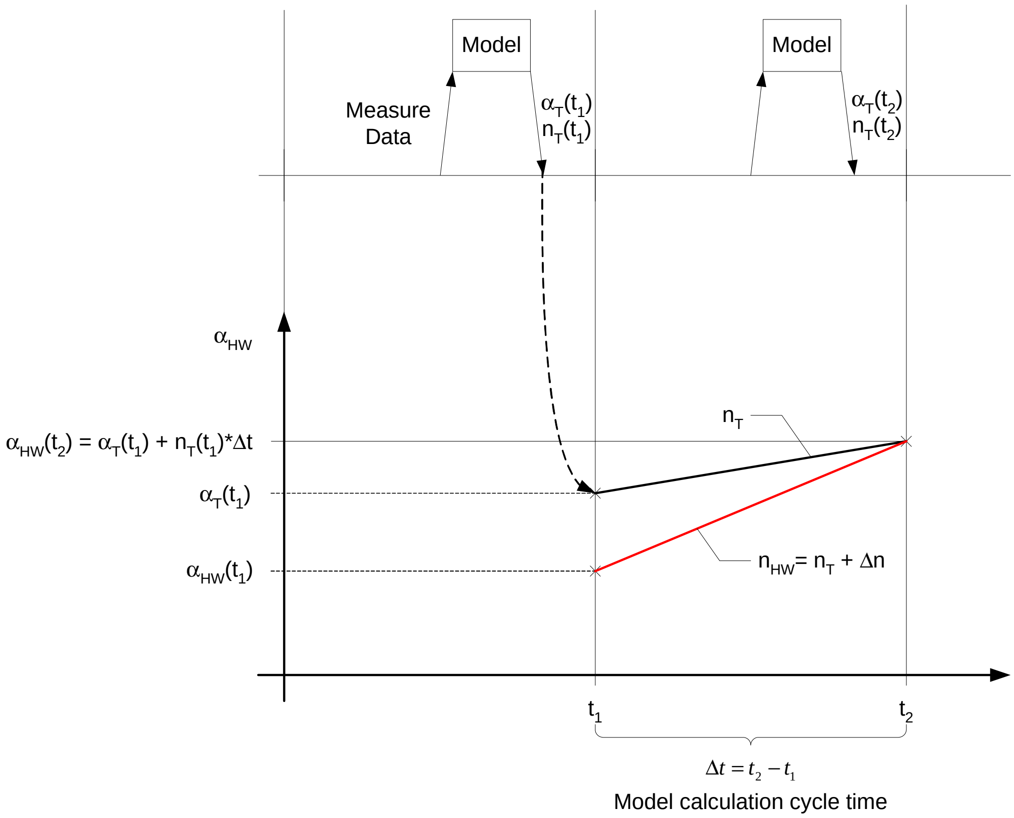


Fig. 22-6 Master Mode with Angle Adaption

The input and output values in the simulation of electric motors in "master mode with angle adaption" are listed in Tab. 22-4 on page 681 and shown in Fig. 22-7 on page 681.

Using the control value Δt_{CD}, a delay can be set by the hardware (in relation to the relevant start of the simulation cycle) for the acceptance of the target values n_T and α_T. In other words, the hardware does not accept the target values n_T and α_T at t_v but at $t_v + \Delta t_{CD}$.

Signal	Direction	Description
n_T	Input	Target engine speed in revolutions per minute
α_T	Input	Target value of the mechanical angle at the start of the next simulation cycle in °RA
Δt_{CD}	Input	Delay for the acceptance of target engine speed and target angle
$\Delta\phi$	Input	Phase of the phase-shifted mechanical angle in °RA
$\Omega_{\Delta\phi}$	Input	Change speed in °RA per second of the phase-shifted mechanical angle when the target value changes
α	Output	Rotor position or mechanical angle in °RA
α_{PS}	Output	Phase-shifted mechanical angle in °RA

Tab. 22-4 Control and Status Signals of the RPM Unit in the Simulation of Electric Motors in "Master Mode with Angle Adaption"

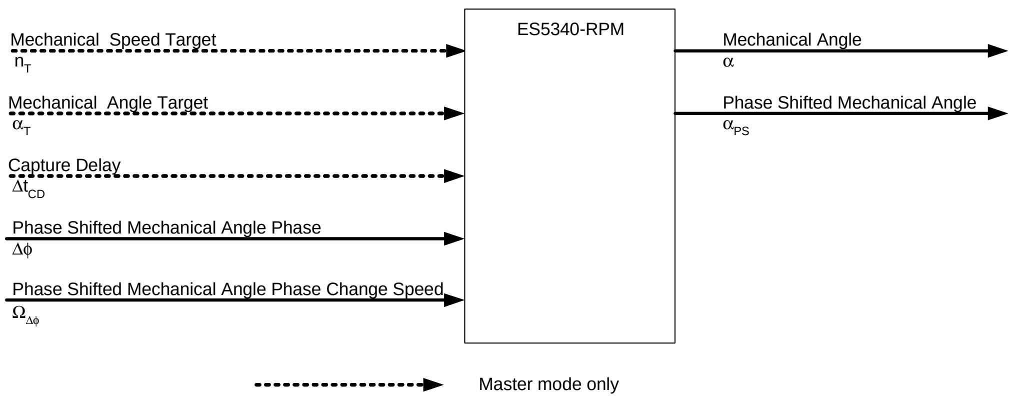


Fig. 22-7 Control and Status Signals of the RPM Unit in the Simulation of Electric Motors in "Master Mode with Angle Adaption"

22.2.1 ES5340-RPM – RPM Unit

The ES5340-RPM device is used to configure and control the ES5340-RPM unit.

22.2.2 Globals (ES5340-RPM)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

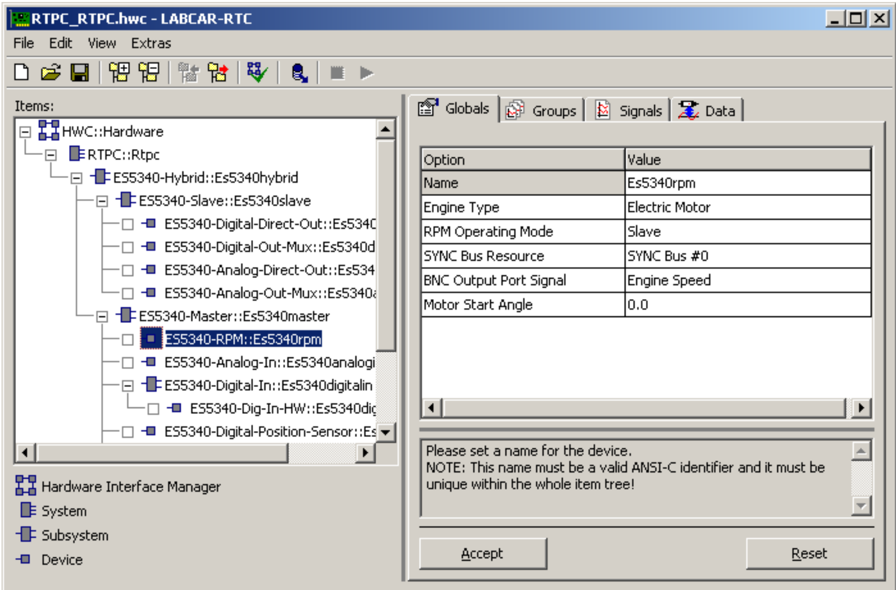


Fig. 22-8 The "Globals" Tab of the ES5340-RPM Device

Engine Type

Selection of the engine type.

RPM Operating Mode

Selection of master or slave mode. In the case of the electric motor, there is a choice of simple master mode and "master mode with angle adaption".

SYNC Bus Resource

The angle clock signal is transported between the boards via synchronization bus lines. This configuration parameter is used to select the bus line via which the angle clock is read in in slave mode or output in master mode.

If the RPM unit is configured to master mode, there is an additional option "None" – in this case an angle clock is generated and used internally on the board, but not output on one of the two synchronization buses.

Note

Please note that there can only be one angle clock master on each synchronization bus!

BNC Output Port Signal

This is where the output signal of the BNC port on the front panel of the ES5340.1/.2 Electric Drive Simulation Board is defined. The signals shown in Fig. 22-9 are available for selection.

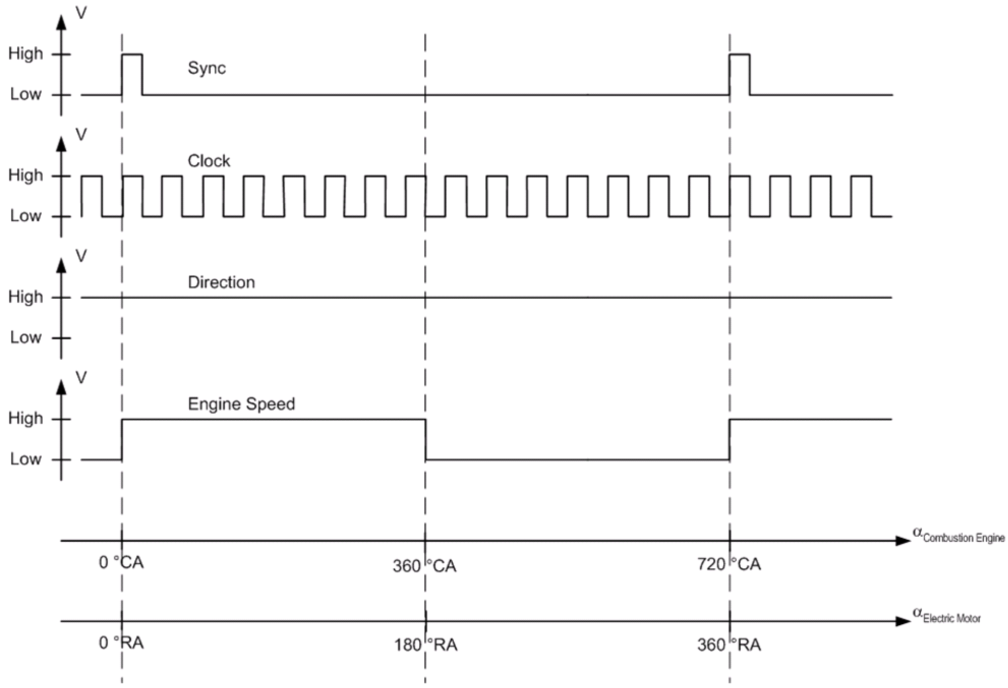


Fig. 22-9 Output Signal at the BNC Port

Motor Start Angle

The angle at which the engine starts after model initialization.

The following table summarizes the properties of the individual configuration parameters.

Parameter	Data Type	Can be edited online	Notes
Engine Type	int	No	0: Combustion engine 1: Electric motor
RPM Operating Mode	int	Yes	0: Slave 1: Master 2: Master with Angle Adaption
SYNC Bus Resource	int	Yes	0: SYNC Bus #0 1: SYNC Bus #1 255: None (only in master mode)
BNC Output Port Signal	int	Yes	0: Sync 1: Clock 2: Direction 3: Engine Speed
Motor Start Angle	float	Yes	0.0 °RA…360.0 °RA: Electric motor 0.0 °CA…720.0 °CA: Combustion engine

Tab. 22-5 The "Globals" Tab of the ES5340-RPM Device

22.2.3 Groups (ES5340-RPM)

The ES5340-RPM device has an "Outputs" signal group for controlling the RPM unit and an "Inputs" signal group for reading the current status information of the RPM unit.

Note

Please note the following restriction when assigning a task of the real-time operating system to the "Outputs" signal group:
If the configuration parameter "RPM Operating Mode" is set to "Master with Angle Adaption" and a "ES5340-Dig-In-Inverter-Meas" device is ***simultaneously*** *integrated to execute inverter measurements, the same task must be assigned to both the "Meas" signal group of the "ES5340-Dig-In-Inverter-Meas" device and the "Outputs" signal group of the "ES5340-RPM" device.*

22.2.4 Data (ES5340-RPM)

The signals made available in the "Data" tab for controlling and monitoring the status of the RPM unit depend on the type of engine selected. Fig. 22-10 on page 684 shows the RTIO signals for the engine type "Combustion Engine" – Tab. 22-6 on page 685 lists the signal properties.

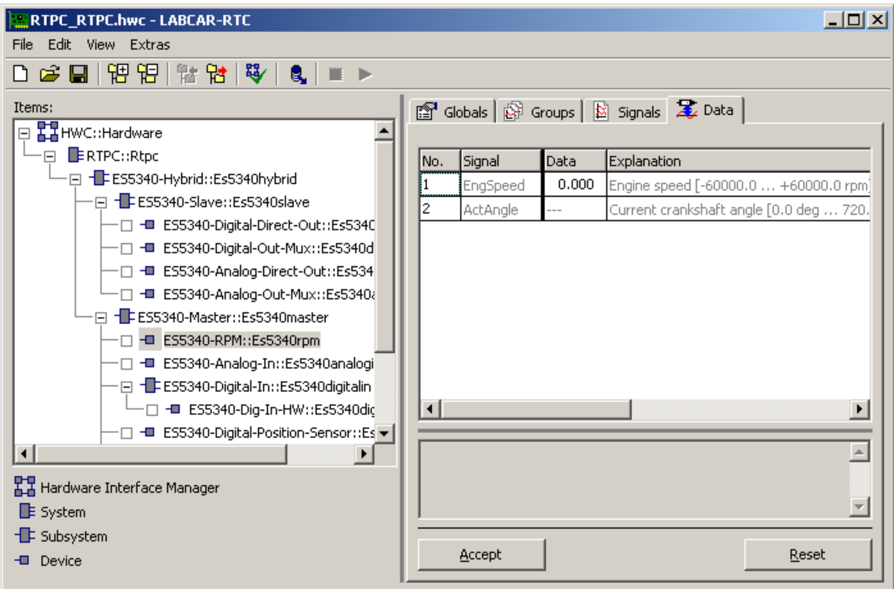


Fig. 22-10 The "Data" Tab of the ES5340-RPM Device (Combustion Engine)

Signal Name	Data Type	Notes
EngSpeed	float	Engine speed in revolutions per minute Value range: -60000.0 rpm … +60000.0 rpm Resolution: 0.001 rpm An engine rotating backwards can be simulated by entering a negative speed.
ActAngle	float	Current degree crankshaft angle in °CA Value range: 0.0 °CA … 720.0 °CA Resolution: 0.011 °CA

Tab. 22-6 ES5340-RPM Device: RTIO Signals (Combustion Engine)

The RTIO signals for the engine type "Electric Motor" are shown in Fig. 22-11 and explained briefly in Tab. 22-7 on page 686.

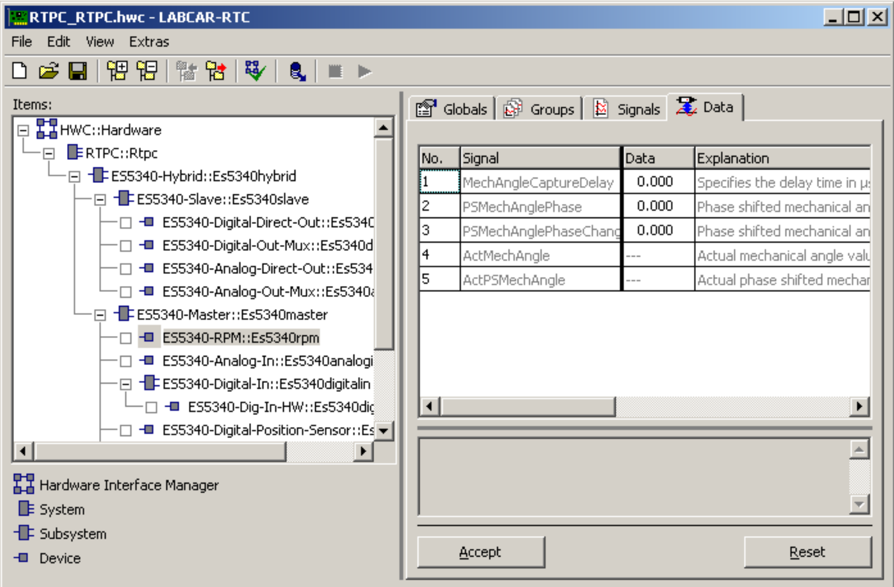


Fig. 22-11 The "Data" Tab of the ES5340-RPM Device (Electric Motor)

Signal Name	**Data Type**	**Notes**
MechSpeed	float	Engine speed in revolutions per minute Value range: -30000.0 rpm … +30000.0 rpm Resolution: 0.001 rpm An engine rotating backwards can be simulated by entering a negative speed.
MechAngle	float	Target angle specification in degrees Value range: 0.0 °RA … 360 °RA Resolution: 0.0055 °RA This signal is only of any importance when the RPM unit is operated as "Master with Angle Adaption".
MechAngle-CaptureDelay	float	This signal defines the delay (in µs) until the hardware accepts the target values for RPM and angle (with reference to the trigger event). Maximum value: 1000.0 µs This signal is only of any importance when the RPM unit is operated as "Master with Angle Adaption".
PSMechAngle-Phase	float	Offset of the phase-shifted mechanical angle in relation to the rotor angle Value range: -360.0 °RA … 360.0 °RA Resolution: 0.0055 °RA
PSMechAngle-PhaseChangeSpeed	float	Change speed of the phase-shifted mechanical angle in relation to the rotor angle after a "PSMechAngle" jump (in degrees per second) Maximum change speed: 1000.0 °RA/s If 0 is entered here, target phase shift is set immediately.
ActMechAngle	float	Current rotor angle in degrees Value range 0.0 °RA … 360 °RA Resolution: 0.0055 °RA
ActPSMechAngle	float	Current value of the angle phase-shifted with respect to the rotor angle in degrees Value range 0.0 °RA … 360 °RA Resolution: 0.0055 °RA

Tab. 22-7 ES5340-RPM Device: RTIO Signals (Electric Motor)

22.3 ES5340-Analog-In – Analog Inputs

The ES5340.1/.2 Electric Drive Simulation Board has four analog inputs with an electrical strength of ±60 V – each of these inputs is routed to a channel of a 4 channel AD converter.

In each case, two AD converters have a voltage range of 0 to 5 V or 0 to 40 V. Each channel of an AD converter converts at a rate of 500000 samples per second.

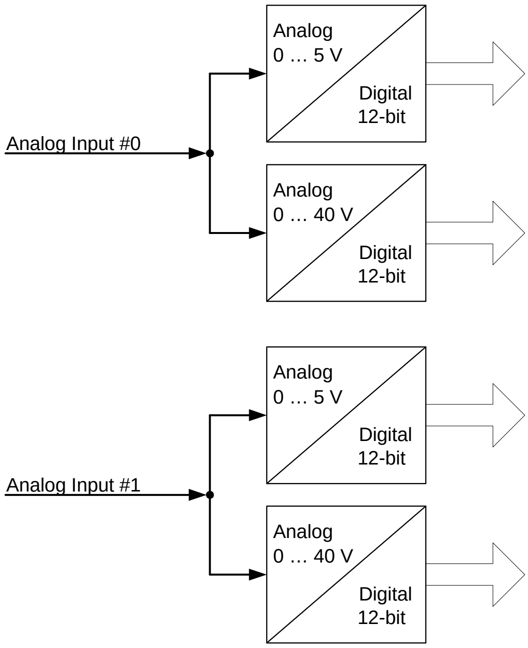


Fig. 22-12 Schematic of the Analog Unit

22.3.1 ES5340-Analog-In – Analog Inputs

The "ES5340-Analog-In" device is used to configure the AD converters and acquire the analog voltages applied.

22.3.2 Signals (ES5340-Analog-In)

The average value of acquired AD values can be calculated for each channel of the analog converter in the "Signals" tab.

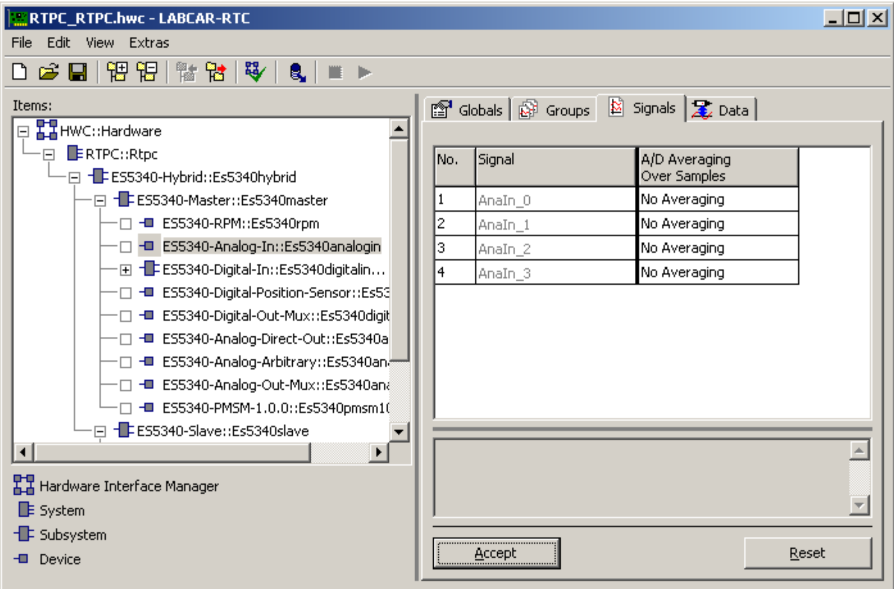


Fig. 22-13 The "Signals" Tab of the ES5340-Analog-In Device

Parameter	Data Type	Can be edited online	Notes
A/D Averaging over Samples	int	Yes	0: No averaging 1: Averaging over 2 samples 2: Averaging over 4 samples 3: Averaging over 8 samples 4: Averaging over 16 samples 5: Averaging over 32 samples 6: Averaging over 64 samples 7: Averaging over 128 samples

Tab. 22-8 ES5340-Analog-In Device: Configuration Parameters of the "Signals" Tab

22.3.3 Data (ES5340-Analog-In)

The "Data" tab shows the acquired (and optionally averaged) voltage values in Volt.

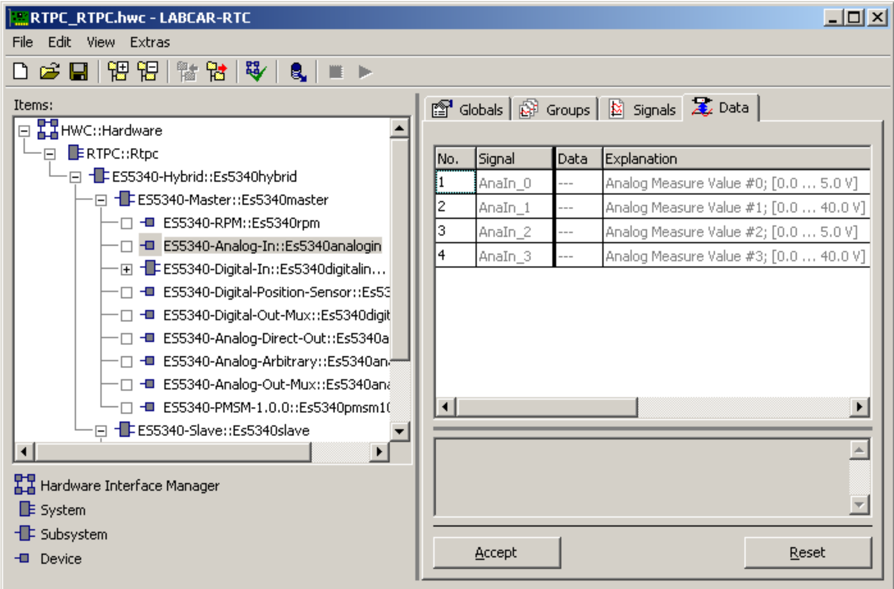


Fig. 22-14 The "Data" Tab of the ES5340-Analog-In Device

Signal Name	Data Type	Notes
AnaIn_0	float	Analog input 0 (measures in the range 0.0…5.0 V)
AnaIn_1	float	Analog input 1 (measures in the range 0.0…40.0 V)
AnaIn_2	float	Analog input 2 (measures in the range 0.0…5.0 V)
AnaIn_3	float	Analog input 3 (measures in the range 0.0…40.0 V)

Tab. 22-9 ES5340-Analog-In Device: Signals of the "Data" Tab

22.4 ES5340-Digital-In – Digital Inputs

This RTIO element is only used for creating hierarchies. No other settings can be made.

22.4.1 ES5340-Dig-In-HW – Configuration of the Digital Inputs

This is where you can configure how the 20 digital inputs are to be used. The inputs can either be used for the general acquisition of time-variable signals (e.g. PWM signals) or alternatively for special electric motor measurement functions.

22.4.2 Globals (ES5340-Dig-In-Hw)

This is where you can assign special inverter functions to the hardware input channels and specify the thresholds for the inputs.

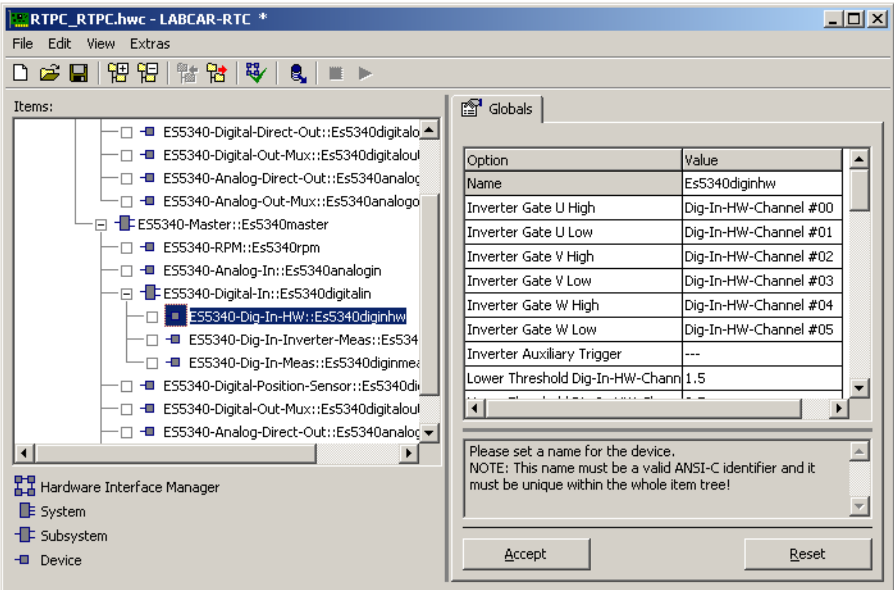


Fig. 22-15 The "Globals" Tab of the ES5340-Dig-In-Hw Device

Inverter Gate {U, V, W} {High, Low}

These option fields are used to assign the inverter control signals to hardware input channels. The channels connected to the relevant outputs of the hybrid ECU must be set. This assignment is required for inverter signal measurement by the ES5340-Dig-In-Inverter-Meas device or by the FPGA model of the electric motor simulation (ES5340-PMSM device).

- Default values: Dig-In-HW channels 0…5

Inverter Auxiliary Trigger

This option field is used to assign the inverter auxiliary trigger signal to a hardware input channel. The auxiliary signal must generate a rising or a falling edge either in the middle or at the beginning of the PWM phase.

This setting is only required for the corresponding operating modes of the ES5340-Dig-In-Inverter-Meas device.

- Default value: --- (no assignment)

Lower Threshold Dig-In HW Channel #<xx>

These option fields describe the voltage value at input #<xx> which, when undercut, triggers a logic transition from High → Low.

- Default value: 1.5 Volt
- Setting range: 0.0 … 10.0 Volt

Upper Threshold Dig-In HW Channel #<xx>

These option fields describe the voltage value at input #<xx> which, when exceeded, triggers a logic transition from Low → High.

- Default value: 3.5 Volt
- Setting range: 0.0 … 10.0 Volt

If a higher value is entered for the lower threshold of a channel then for the higher threshold the two thresholds are automatically swapped.

22.4.3 ES5340-Dig-In-Meas – Non-Synchronized Time/Frequency Measurements

This RTIO device is used to measure time, frequency and duty cycle at periodic signals or to acquire the binary value of an input. A maximum of ten measurement channels can be configured in one RTIO device and a maximum of four of these RTIO devices can be integrated into an ES5340-Digital-In subsystem. In total, there are a maximum of 40 software channels available.

The individual measurements are independent of each other and are also basically independent of any other use of the inputs (inverter signal measurement).

The Basic Principle of Measure Value Acquisition and Calculation

Measure values are acquired on the ES5340.1/.2 Electric Drive Simulation Board – the 20 input signals are digitized in accordance with the thresholds specified (in the ES5340-Dig-In-HW device).

A data word is generated for any change of edge in a hardware channel. This consists of a 31-bit time stamp of a local counter and one bit for edge information. The time stamp has a resolution of 8 ns (125 MHz clocking) so a time of max. 17.28 s can be shown. An edge bit of "1" means a positive edge, i.e. the level is then "1".

The measure values are calculated in the driver of the real-time PC. For this purpose, the data of the last four edge changes are transferred to the RAM of the real-time PC for every hardware channel using cyclical DMA transfer. With a sufficiently low DMA transfer cycle time (default value: 100 µs), all the information for calculating the measurement procedures with low latency is available in the driver.

If there have not been four edge changes since the beginning of data acquisition on a hardware channel, the same data word is generated for all four edges at the start of data acquisition. This consists of the current time stamp and edge information indicating the current level.

These simple coding rules make it possible to distinguish valid data from invalid data:

- The current level can also be determined without an edge change.
- Invalid edge data has an edge bit that has not changed since the previous data.

The measurement procedures for returning the level thus always return valid data. In all other measurement procedures, there have to be sufficient edges available which depends on the procedure and on the current state.

If no measure value can be calculated, -8888.0 is returned as value (providing there is no timeout or monitoring).

22.4.4 The Measurement Procedures

Below you will find a description of the possible procedures for measuring digital signals.

Note

Generally speaking, the measured times between two flank events must not exceed the mapping area of the 31-bit counter (17.28 s). If this happens the counter overflows resulting in incorrect measurements.

Frequency and Cycle Time Measurement

The measurement procedures

- Cycle Time --/-- [µs]
- Cycle Time --\-- [µs]
- Frequency --/-- [Hz]
- Frequency --\-- [Hz]

provide the cycle time or frequency of the signal at the relevant input measured between two rising edges (--/--) or two falling edges (--\--).

High/Low Time Measurement

The measurement procedures

- High Time [µs]
- Low Time [µs]

provide the time between the last rising and the following falling edge (High Time) or the last falling and rising edge (Low Time).

Duty Cycle Measurements

The measurement procedures

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

provide the ratio of the low or high time, with reference to the cycle time of rising to rising edge (--/--) or falling to falling edge (--\--).

Level Measurement

The measurement procedures

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

provide the level of a PWM input in the form of active/inactive information. "0" means the signal is inactive, "1" means the signal is active.

What Happens if a Timeout Occurs

Monitoring of timeouts can be set for every measurement channel. When monitoring is activated, the default return value and the monitoring interval can be changed. A timeout is reached when the difference between the current time of a DMA transfer and the time of the last edge change of the channel exceeds the specified interval time. The maximum interval time is 10 s. The last measure value is transferred up to this time (or -8888.0 if there is no valid value).

The measure value depends on the operating mode when a timeout occurs. There are basically three operating modes for timeout monitoring:

- Inactive

  No monitoring – the last measured value (or -8888.0 if there is no value) is returned. The "Tout_<xx>" signal always has the value "2 = not used".

- Intvl Predef

  If a timeout occurs, the "Tout_<xx>" signal is set to "1 = Timeout" and the predefined value returned.

- Intvl InpDep

  The "Tout_<xx>" signal is set to "1 = Timeout" and a value dependent on the measurement procedure and possibly the input level is returned (see Tab. 22-10).

Measurement Procedures	Value (Input Level Dependent)
No Measurement	-8888.0
High Time [µs] Low Time [µs]	0 0
Cycle Time --/-- [µs] Cycle Time --\-- [µs]	Time since the last edge [µs] Time since the last edge [µs]
Frequency --/-- [Hz] Frequency --\-- [Hz]	Reciprocal value of the time since the last edge [Hz] Reciprocal value of the time since the last edge [Hz]
Duty Cycle L/(L+H) --/-- Duty Cycle L/(L+H) --\-- Duty Cycle H/(L+H) --/-- Duty Cycle H/(L+H) --\--	1.0 with a low level, 0.0 with a high level at the input 1.0 with a low level, 0.0 with a high level at the input 0.0 with a low level, 1.0 with a high level at the input 0.0 with a low level, 1.0 with a high level at the input
Level (Active High) Level (Active Low)	0 0

Tab. 22-10 Return Values in a Timeout ("Intvl InpDep" Operating Mode)

For those values transferred when frequency and period duration are measured, the overflow of the 31-bit wide counter is taken into consideration, i.e. these can be as small or as large as required (frequency or time) as part of the 64-bit floating-point display.

The timeout is over when the 32-bit data word of the latest edge (time stamp + edge information) changes in comparison to the one transferred at the start of the timeout. This allows timeouts to be displayed that exceed the overflow value of the 31-bit counter (17.28 s). The "Tout_<xx>" signal is set to the value "0 = no Timeout".

22.4.5 Groups (ES5340-Dig-In-Meas)

This is where you can assign the signal group a task – all measurement channels of this signal group are calculated in this task.

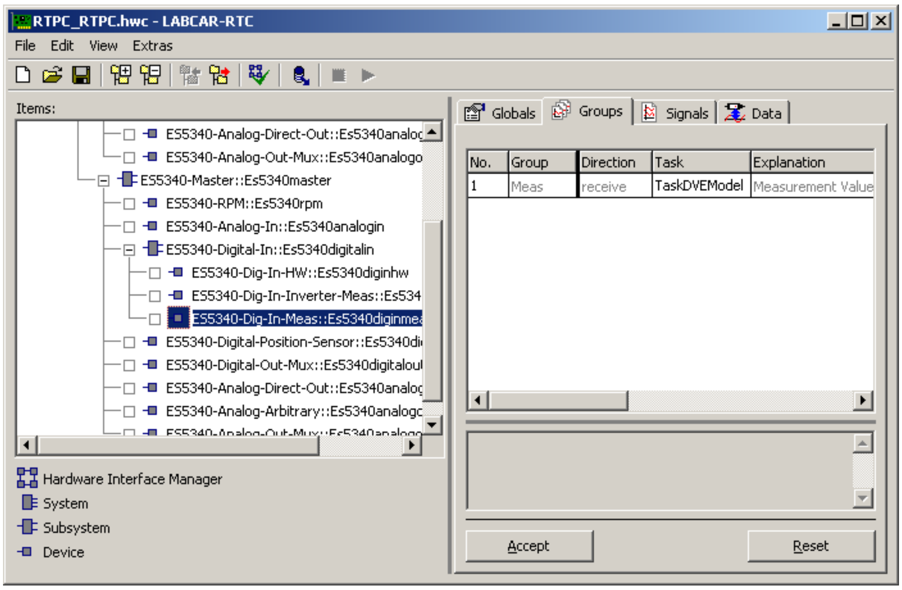


Fig. 22-16 The "Groups" Tab of the ES5340-Dig-In-Meas Device

22.4.6 Signals (ES5340-Dig-In-Meas)

This is where you can configure the properties of the measurement channels. The configuration parameters apply to all "MeasVal_<xx>" measure values. All parameters can be configured online.

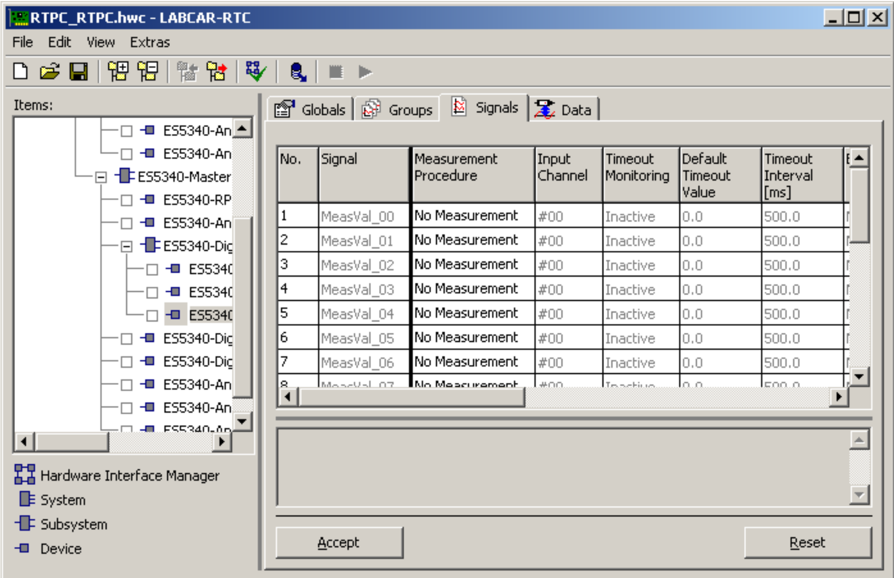


Fig. 22-17 The "Signals" Tab of the ES5340-Dig-In-Meas Device

Measurement Procedure

The measurement procedure (see "The Measurement Procedures" on page 692). If "No Measurement" is activated, no value is calculated and -8888.0 is transferred.

- Default value: No Measurement

Input Channel

The hardware input channel to which the measurement procedure refers.

- Default value: #00
- Setting range: #00, #01, ..., #19

Timeout Monitoring

The operating mode for monitoring a timeout (see "What Happens if a Timeout Occurs" on page 693).

- Default value: Inactive

Default Timeout Value

Return value for the timeout operating mode "Intvl InpDep". This value can only be edited if this timeout operating mode is set for the measurement channel.

- Default value: 0.0
- Setting range: 64-bit floating-point number

Timeout Interval [ms]

Interval for monitoring a timeout. This value can only be edited in the timeout operating modes "Intvl Predef" and "Intvl InpDef".

- Default value: 500.0 [ms]
- Setting range: 0.1 … 10000.0 [ms]

22.4.7 Data (ES5340-Dig-In-Meas Device)

This is where the measure results and the corresponding timeout flags are displayed.

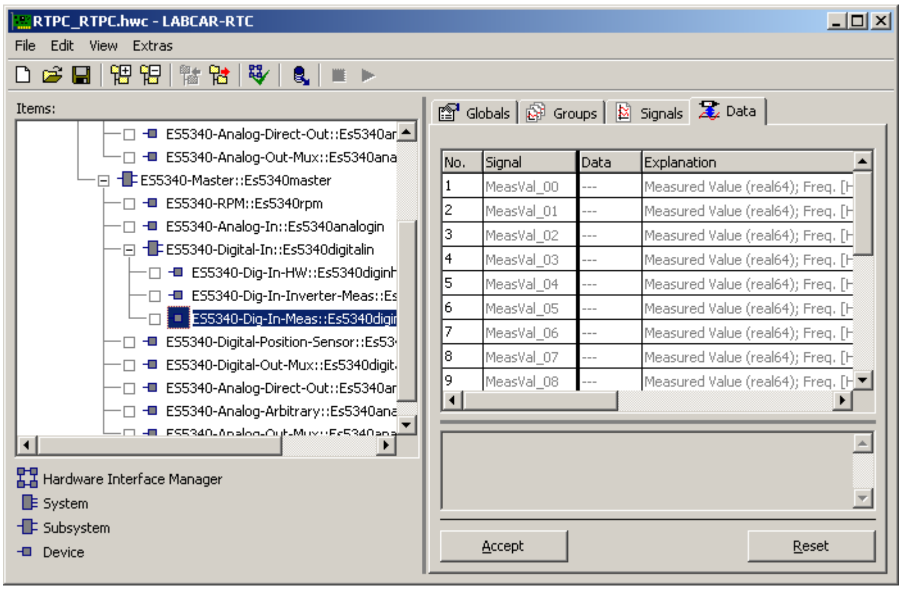


Fig. 22-18 The "Data" Tab of the ES5340-Dig-In-Meas Device

RTIO Signal	Data Type	Signal Group	Description
MeasVal_00 - MeasVal_09	real64	Meas	Measure result, depending on the measurement procedure time [µs], frequency [Hz], ratio, -8888.0 for invalid measure value or timeout default value
Tout_00 - Tout_09	real64	Meas	Indicates specification or state of timeout 0: No timeout, monitoring active if timeout occurs 1: Timeout, monitoring active if timeout occurs 2: No monitoring if timeout

Tab. 22-11 The RTIO Signals of the Signal Group "Meas"

22.4.8 ES5340-Dig-In-Inverter-Meas – Measuring Inverter Control Signals

This RTIO device is used to measure the control signals for a permanent magnet synchronous machine and activate a task with low latency when new measure data is available. The actual measuring takes place in the LABCAR Software. The measure values are transferred to the RAM of the real-time PC via DMA transfer. A task is activated when a software scheduler queries a counter in the transferred measure data.

For this purpose a "Trigger" task must have been created in the "OS-Configuration" section of LABCAR-IP. The "Exclusive Core Usage" option also has to be activated and this task has to be linked exclusively with one of the existing CPU cores.

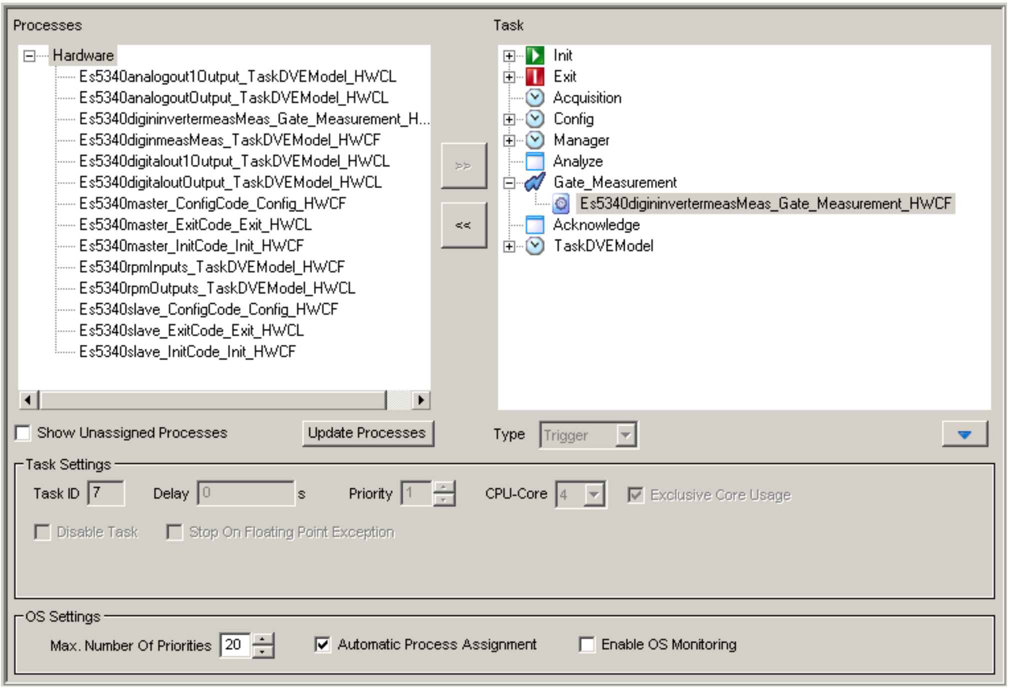


Fig. 22-19 Setting the Trigger Task in the OS Configuration in LABCAR-IP

The task is linked to the inverter measurement by assigning the "Meas" signal group in the "Groups" tab.

This is the only way to achieve the low latency required – for this purpose, you need a real-time PC with a processor with at least three physical cores.

The inverter control signals are assigned to hardware input channels in the ES5340-Dig-In-HW device. If this device is not available, an error message is generated in the RTIO Editor when the hardware configuration is checked and no code is generated.

22.4.9 Globals (ES5340-Dig-In-Inverter-Meas)

This is where you can make settings on Trigger mode and the PWM period.

Trigger Mode

This is where the type of triggering is specified. This takes place either via clock recovery from the control signals or using an auxiliary trigger.

The auxiliary trigger modes need the auxiliary trigger signal to be assigned to a hardware input in the ES5340-Dig-In-HW device. If this is not the case, an error message is issued and it is not possible to set a trigger mode using an auxiliary trigger.

Nominal PWM Period of Center Aligned PWM [usec]

This is where the target value of the period duration of the PWM signal is determined. This is required for calculating the duty cycles so they can take place in the middle of a PWM impulse.

- Default value: 100.0 µs
- Setting range: 0...500.0 µs

Nominal PWM Period Deviation of Center Aligned PWM [%]

This option describes the max. permissible deviation of the PWM period from the target value (in percent). This setting can only be seen and changed when "Show All" has been activated.

- Default value: 5%
- Setting range: 0%…100%

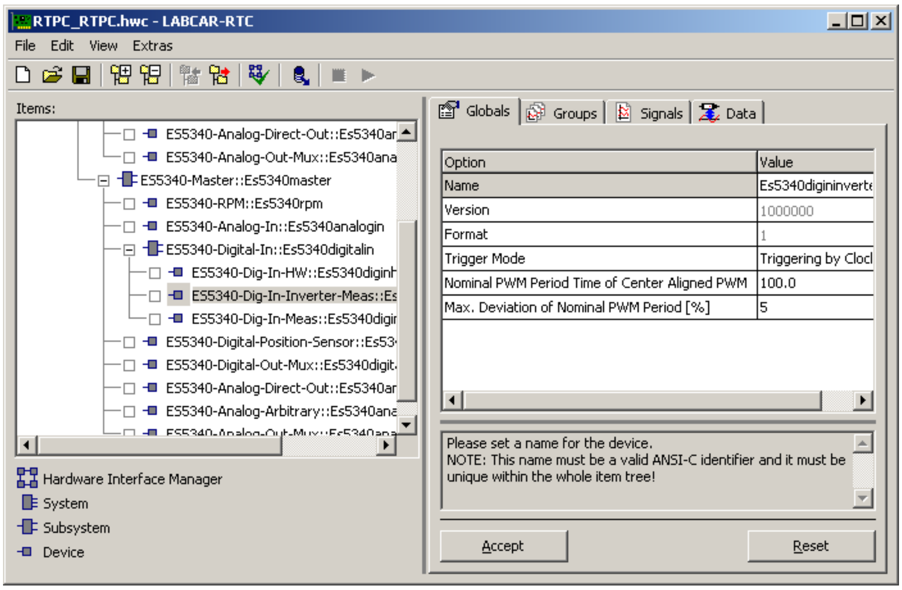


Fig. 22-20 The "Globals" Tab of the ES5340-Dig-In-Inverter-Meas Device ("Show All" View Option)

22.4.10 Groups (ES5340-Dig-In-Inverter-Meas)

This is where you can assign tasks for inverter measuring ("Meas" signal group) and possibly a further control signal group ("Ctrl"). In inverter measuring, this is also the task that is triggered by the new measure data or a new DMA transfer of the center-aligned PWM measuring.

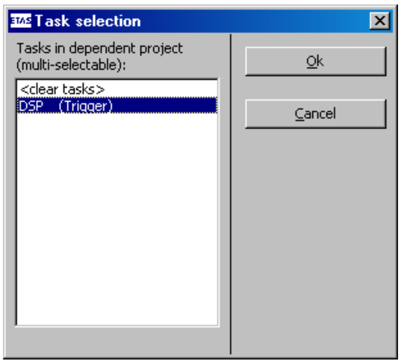


Fig. 22-21 Selection Box for "Trigger" Task

Only "Trigger" tasks with the attribute "Exclusive Core Usage" are available for selection.

If there are no tasks available for selection, proceed as follows:

To create a "Trigger" task

- Save any changes you have made in the LAB-CAR-RTC Editor and close the editor.
- Toggle to the "OS Configuration" tab in LAB-CAR-IP.
- Add a new task or change an existing one as follows:
  - Type: "Trigger"
  - Extended tasks settings: "Exclusive Core Usage" (with the specification of a CPU core)
- Save the changes in LABCAR-IP and then re-open the LABCAR-RTC Editor.
- Assign the new or changed task to the signal group.

A send signal group "Ctrl" is created for the trigger modes with auxiliary trigger. This is used to set additional delay times for the six inverter control signals (Gate U,V,W / High, Low) and the auxiliary trigger signal.

22.4.11 Signals (ES5340-Dig-In-Inverter-Meas)

This is where you can make settings for the trigger mode "Triggering by Clock Recovery from PWM Signals".

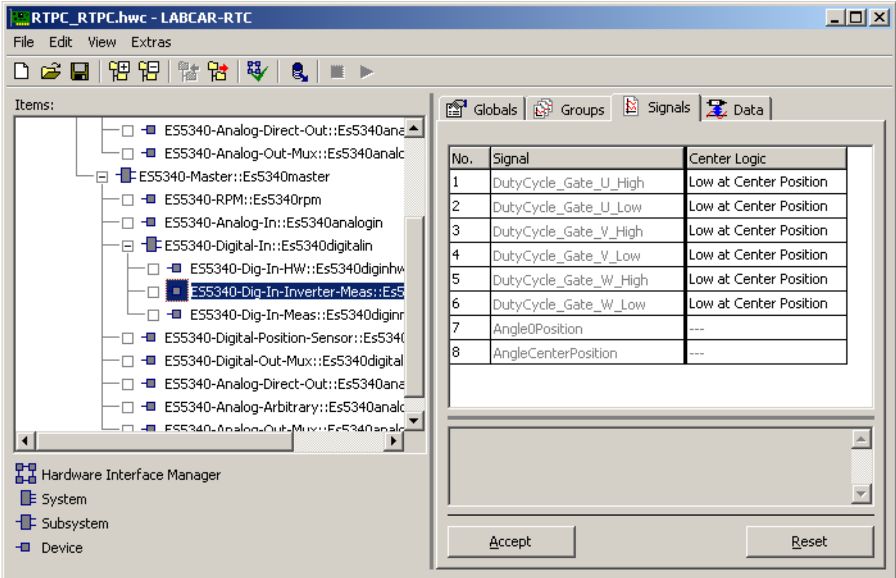


Fig. 22-22 The "Signals" Tab of the ES5340-Dig-In-Inv-Meas Device

Center Logic

These settings determine the polarity of the six inverter control signals by specifying a level in the center of the relevant PWM control impulse. The parameters are of no importance for trigger modes with an auxiliary trigger signal and are not visible.

- Default value: "Low at Center Position"
- Setting range: "Low at Center Position", "High at Center Position"

22.4.12 Data (ES5340-Dig-In-Inverter-Meas)

This is where the measure result of the duty cycles of the inverter control signals and some angle positions are displayed. In the trigger modes with an auxiliary trigger signal, the signals for specifying the delays of the inverter control signals are also available as output values.

RTIO Signal	Data Type	Signal Group	Description
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	Measure result of the duty cycles of the inverter control signals (Duty Cycle H/(H+L)) as ratio [0.0 … 1.0]
Angle0Position	real64	Meas	Starting angle position [0.0 °RA … 360.0 °RA]
AngleCenterPosition	real64	Meas	Center angle position [0.0 °RA … 360.0 °RA]
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	

* derived from 16-bit registers
** mapped to 11-bit registers

Tab. 22-12 The RTIO Signals of the Signal Groups "Meas" and "Ctrl"

22.5 ES5340-Digital-Out – Digital Outputs

The RTIO element ES5340-Digital-Out can be used in four different operating modes.

Mode	Description
Digital Direct Out	Digital output (see "ES5340-Digital-Direct-Out" on page 702)
Digital Position Sensor	Simulation of a position sensor of the type "Digital Position Sensor" (see "ES5340-Digital-Position-Sensor" on page 705)
Digital Arbitrary	Output of a freely configurable digital waveform (see "ES5340-Digital-Arbitrary" on page 709)
PWM Output	Output of PWM signals (see "ES5340-PWM-Output" on page 712)

Tab. 22-13 Operating Modes of the ES5340-Digital-Out RTIO Element

Note

For more information on the operating modes, refer to the ES5340.1/.2 Electric Drive Simulation Board User's Guide.

The configuration of the RTIO element is different for every output mode – the appearance of the ES5340-Digital-Out RTIO element also changes accordingly.

The LABCAR Software uses a two-step procedure when outputting digital values:

1. The output values are configured via the ES5340-Digital-Out RTIO element. You can configure more signals here than there are output channels on the LABCAR Software (max. 8 channels in "Digital-Direct-Out" mode and 8 channels for the other digital output modes)
2. The values actually output are selected via the ES5340-Digital-Out-Mux RTIO element (see "ES5340-Digital-Out-Mux – Digital Output Multiplexer" on page 714).

22.5.1 ES5340-Digital-Direct-Out

This section describes the configuration of the ES5340-Digital-Out RTIO element with the output mode "Digital Direct Out". One to eight digital outputs can be driven directly in this operating mode.

22.5.2 Globals (ES5340-Digital-Direct-Out)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

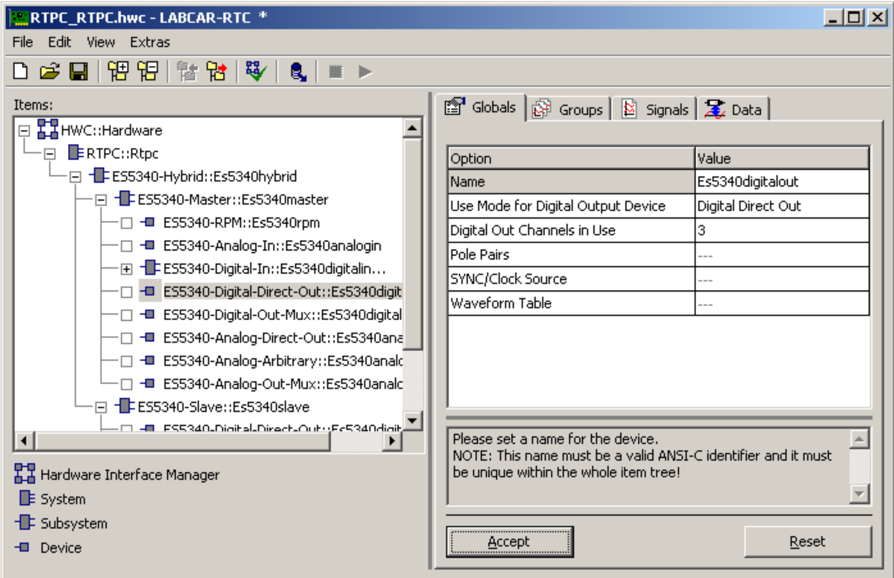


Fig. 22-23 The "Globals" Tab of the ES5340-Digital-Direct-Out RTIO Element

Use Mode for Digital Output Device

Selection of the output mode.

Digital Out Channels in Use

Number of digital output channels used (max. 8).

22.5.3 Groups (ES5340-Digital-Direct-Out)

The ES5340 Digital-Direct-Out-RTIO element has a signal group called "Outputs", for controlling the relevant signal generator.

22.5.4 Signals (ES5340-Digital-Direct-Out)

No further settings are necessary here.

22.5.5 Data (ES5340-Digital-Direct-Out)

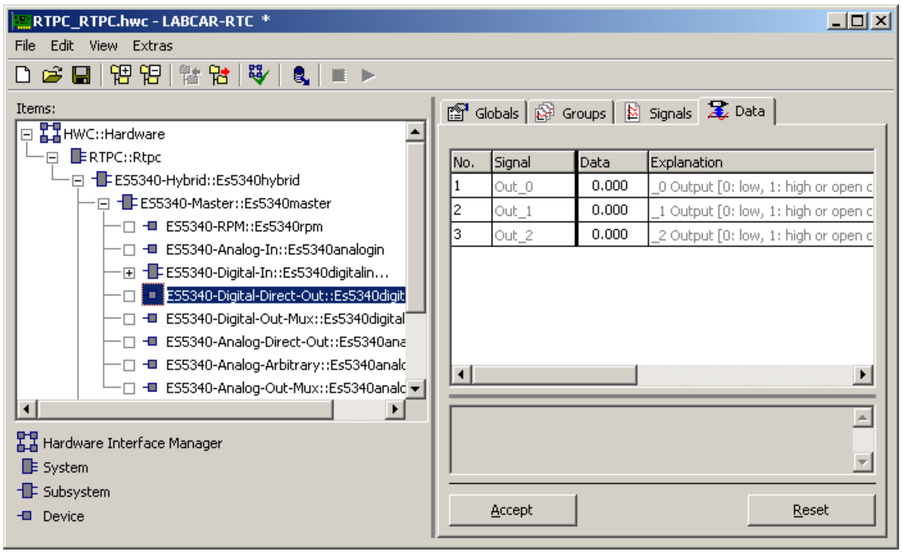


Fig. 22-24 The "Data" Tab of the ES5340-Digital-Direct-Out RTIO Element

Signal Name (n = 0...7)	**Data Type***	**Notes**
OutValue_*n*	uint8	Digital output value 0: low; 1: high or open collector
*Data type which the RTIO driver uses internally for the parameter		

Tab. 22-14 ES5340-Digital-Direct-Out: Signals of the "Data" Tab

22.5.6 ES5340-Digital-Position-Sensor

The operating mode "Digital Position Sensor" makes it possible to simulate a digital position sensor and internally assigns three arbitrary signals generators.

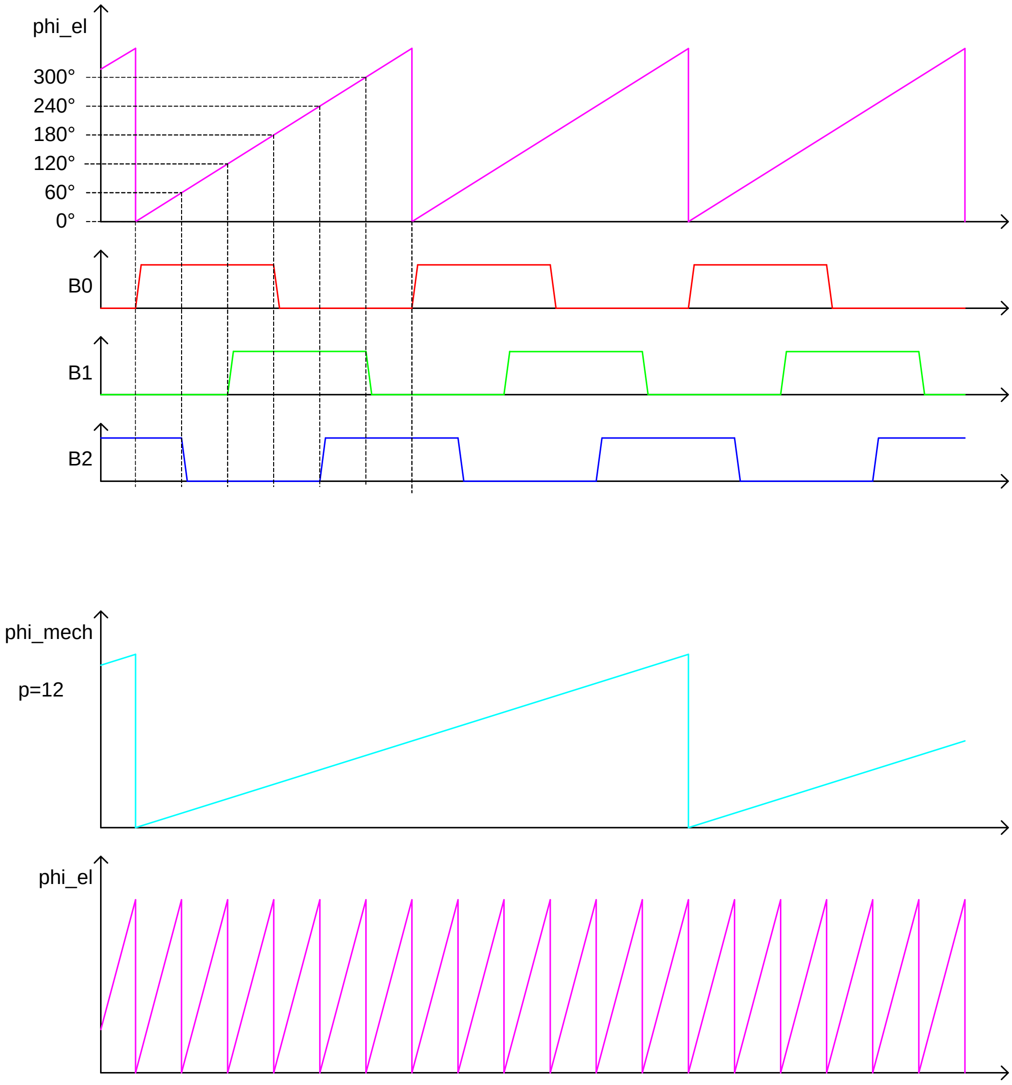


Fig. 22-25 Digital Position Sensor – Signals

22.5.7 Globals (ES5340-Digital-Position-Sensor)

The following figure shows the configuration parameters of the "Globals" tab.

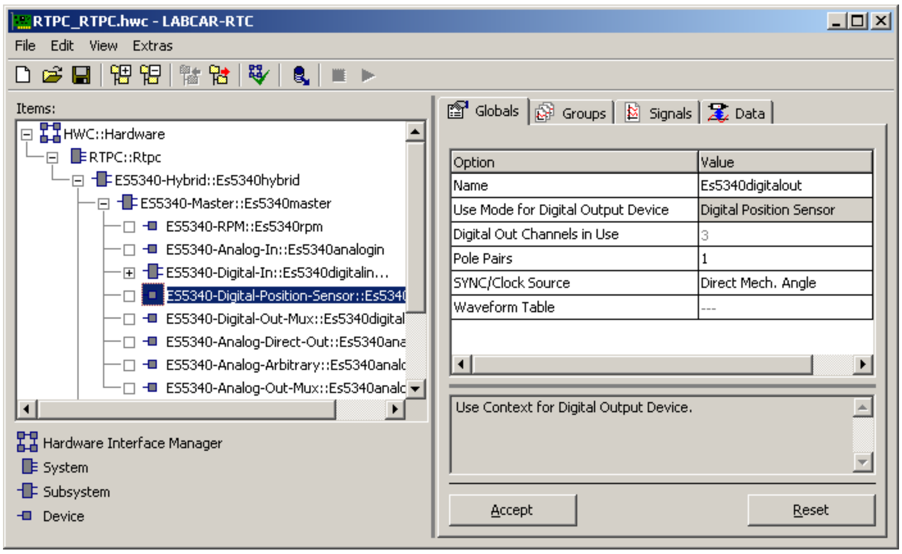


Fig. 22-26 The "Globals" Tab of the ES5340-Digital-Position-Sensor RTIO Element

Use Mode for Digital Output Device

Selection of the output mode.

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The digital signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal). The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case:

- The **local phase shift** is defined in the "Data" tab by the signals "PhaseRefB0", "PhaseRefB1" and "PhaseRefB2".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.

- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

22.5.8 Groups (ES5340-Digital-Position-Sensor)

The ES5340-Digital-Position-Sensor RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.5.9 Signals (ES5340-Digital-Position-Sensor)

The following figure shows the signals of the ES5340-Digital-Position-Sensor RTIO element.

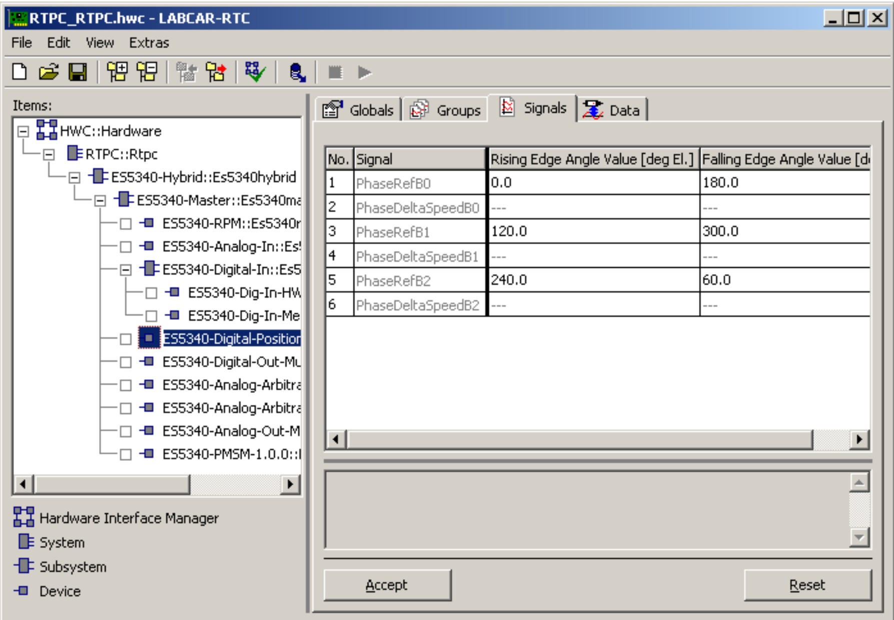


Fig. 22-27 The "Signals" Tab of the ES5340-Digital-Position-Sensor RTIO Element

Rising Edge Angle Value [deg El.]

The angle value for the rising edge of the digital signal – when the item is inserted, the default values from Fig. 22-27 are set.

Falling Edge Angle Value [deg El.]

The angle value for the falling edge of the digital signal – when the item is inserted, the default values from Fig. 22-27 are set.

22.5.10 Data (ES5340-Digital-Position-Sensor)

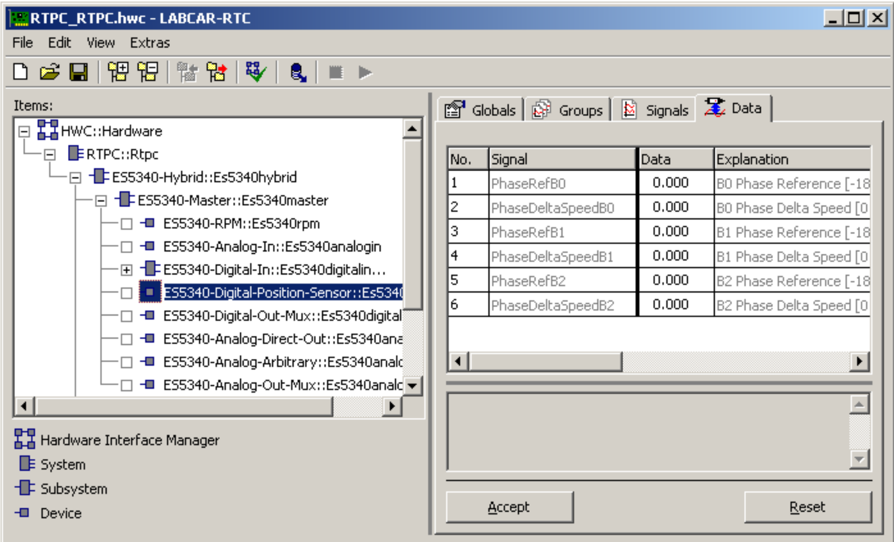


Fig. 22-28 The "Data" Tab of the ES5340-Digital-Position-Sensor RTIO Element

PhaseRefBn (n = 0...2)

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedBn"). This is why the signal "PhaseDeltaSpeedBn" must have a value > 0 for most applications.

PhaseDeltaSpeedBn (n = 0...2)

The phase change speed is controlled by the signal "PhaseDeltaSpeedB*n*" – a change of the signal "PhaseRefBx" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000 deg mech/s.

Signal Name	Data Type*	Notes
PhaseRefB0	sint16	B0 phase reference [-180.0…180.0]
PhaseDeltaSpeedB0	uint16	B0 phase change speed [0.0…1000.0 deg mech/s]
PhaseRefB1	sint16	B1 phase reference [-180.0…180.0]
PhaseDeltaSpeedB1	uint16	B1 phase change speed [0.0…1000.0 deg mech/s]
PhaseRefB2	sint16	B2 phase reference [-180.0…180.0]
PhaseDeltaSpeedB2	uint16	B2 phase change speed [0.0…1000.0 deg mech/s]

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-15 ES5340-Digital-Position-Sensor: Signals of the "Data" Tab

22.5.11 ES5340-Digital-Arbitrary

In the operating mode "Digital Arbitrary", an arbitrary signal generator is used to generate a digital waveform (using the RPM generator).

22.5.12 Globals (ES5340-Digital-Arbitrary)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

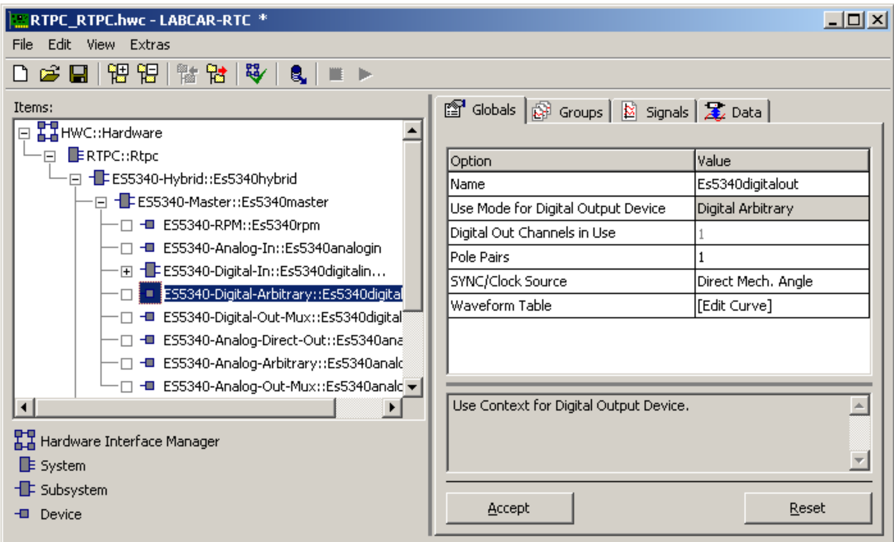


Fig. 22-29 The "Globals" Tab of the ES5340-Digital-Arbitrary RTIO Element

Use Mode for Digital Output Device

Selection of the output mode.

Digital Out Channels in Use

In the operating mode "Digital Arbitrary", this setting is of no importance as only one arbitrary signal generator is used per RTIO element.

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The digital signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal). The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case:

- The **local phase shift** is defined in the "Data" tab by the signal "PhaseValue".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.
- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

Waveform Table

This is where the signal form of the output signal can be specified using a table. The z values of the table are scaled within the interval -1.0 … 1.0, the available x values are scaled within one cycle of the RPM generator. The interpolated analog z values are converted to binary values (threshold: 0.1 (uint16-output value: 0xCCC), i.e. < 0.1 → low, > = 0.1 → high).

22.5.13 Groups (ES5340-Digital-Arbitrary)

The ES5340-Digital-Arbitrary RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.5.14 Signals (ES5340-Digital-Arbitrary)

No further settings are necessary here.

22.5.15 Data (ES5340-Digital-Arbitrary)

The following figure shows the "Data" tab.

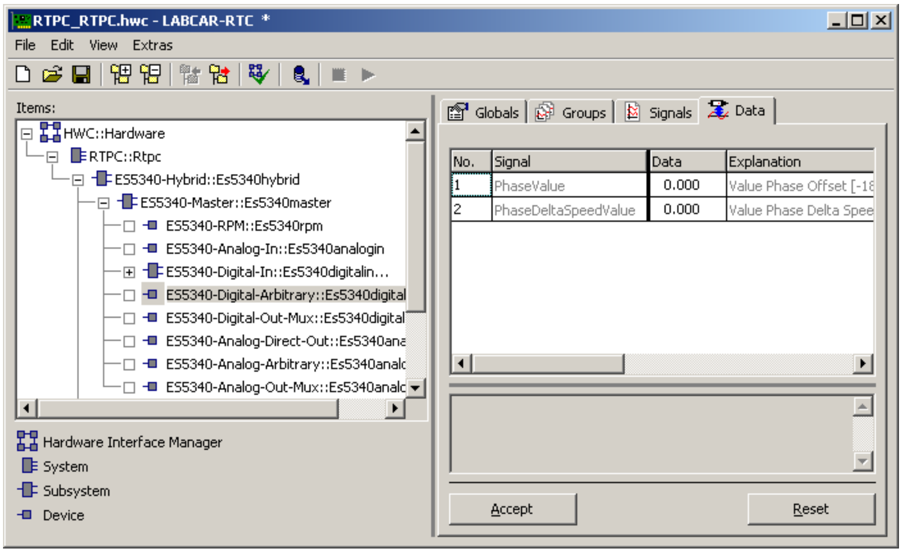


Fig. 22-30 The "Data" Tab of the ES5340-Digital-Arbitrary RTIO Element

PhaseValue

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedValue"). This is why the signal "PhaseDeltaSpeed-Value" must have a value > 0 for most applications.

PhaseDeltaSpeedValue

The phase change speed is controlled by the signal "PhaseDeltaSpeed" – a change of the signal "PhaseValue" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000 deg mech/s.

Signal Name	Data Type*	Notes
PhaseValue	sint16	Local phase shift [-180.0…180.0 deg El. Angle]
PhaseDeltaSpeedValue	uint16	Phase change speed [0.0…1000.0 deg mech/s]

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-16 ES5340-Digital-Arbitrary: Signals of the "Data" Tab

22.5.16 ES5340-PWM-Output

In the operating mode "PWM Output", up to eight digital outputs can be configured for PWM output ("active-high" only). Frequency and duty cycle are specified separately for each PWM channel. The new data is accepted at the end of each relevant PWM cycle.

22.5.17 Globals (ES5340-PWM-Output)

The following figure shows the configuration parameters of the "Globals" tab.

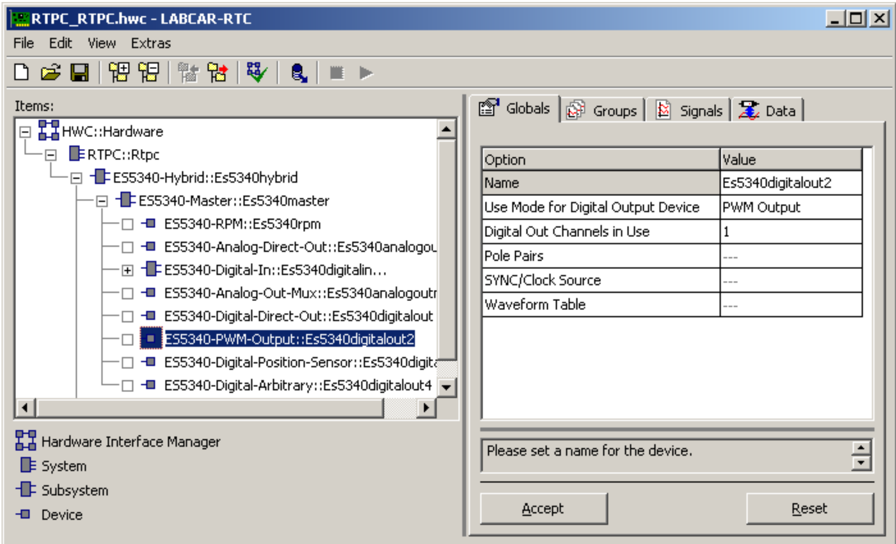


Fig. 22-31 The "Globals" Tab of the ES5340-PWM-Output Element

Use Mode for Digital Output Device

This is where the output mode is selected.

Digital Out Channels in Use

Number of digital output channels used (max. 8).

22.5.18 Groups (ES5340-PWM-Output)

The ES5340-PWM-Output has a signal group called "Outputs" for controlling the relevant signal generator(s).

22.5.19 Signals (ES5340-PWM-Output)

No further settings are necessary here.

22.5.20 Data (ES5340-PWM-Output)

The following figure shows the "Data" tab.

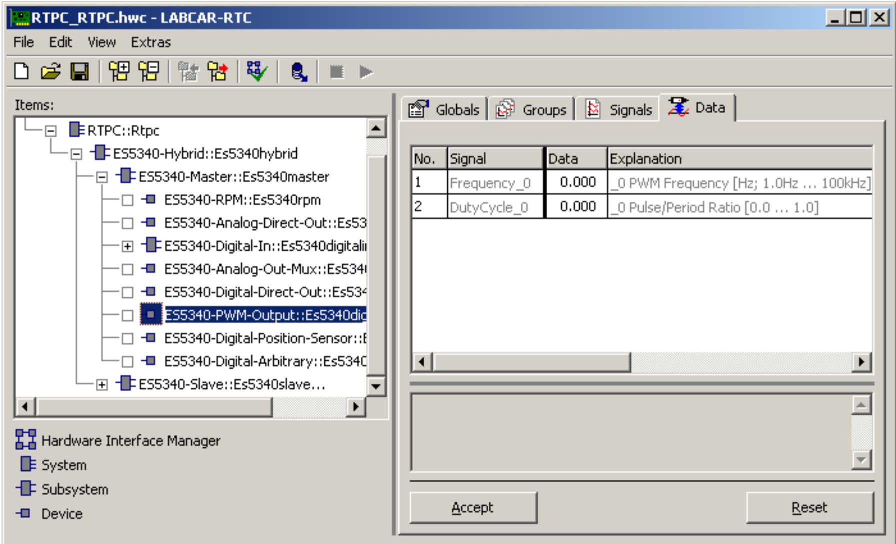


Fig. 22-32 The "Data" Tab of the ES5340-PWM-Output-RTIO Element

Frequency_n

The frequency of the PWM signal at output n (n = 0...7).

DutyCycle_n

The duty cycle of the PWM signal at output n (n = 0...7).

Signal Name (n = 0...7)	**Data Type***	**Notes**
Frequency_*n*	uint32	Frequency of the PWM signal (1.0Hz…100kHz)
DutyCycle_*n*	uint32	Duty cycle (0.0…1.0)

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-17 ES5340-PWM-Output: Signals of the "Data" Tab

22.6 ES5340-Digital-Out-Mux – Digital Output Multiplexer

In the RTIO element "ES5340-Digital-Out-Mux", signals configured in the RTIO element "ES5340-Digital-Out" are assigned to the digital signal outputs that are actually present on the ES5340.1/.2 Electric Drive Simulation Board.

22.6.1 Signals (ES5340-Digital-Out-Mux)

The eight (LABCAR Software-Master) or six (LABCAR Software-Slave) digital outputs are configured in the "Signals" tab.

The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

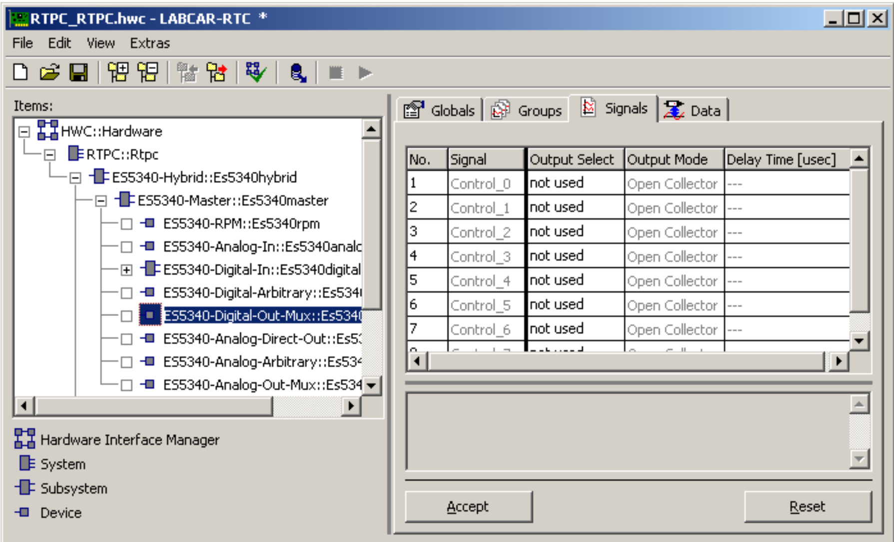


Fig. 22-33 The "Signals" Tab of the ES5340-Digital-Out-Mux RTIO Element

Output Select

The option field "Output Select" is used to select the internal signal to be applied to the particular output.

A list of all internal signal sources is available for selection:

- Output values of all Digital-Out RTIO elements
- Digital measure values of the inverter gate control
- Digital output values of a FPGA model (if there is one)

If an FPGA model provides digital output values, these can also be output using a connected LABCAR Software-Slave board. This requires the two boards to be connected to each other via Gigabit Link.

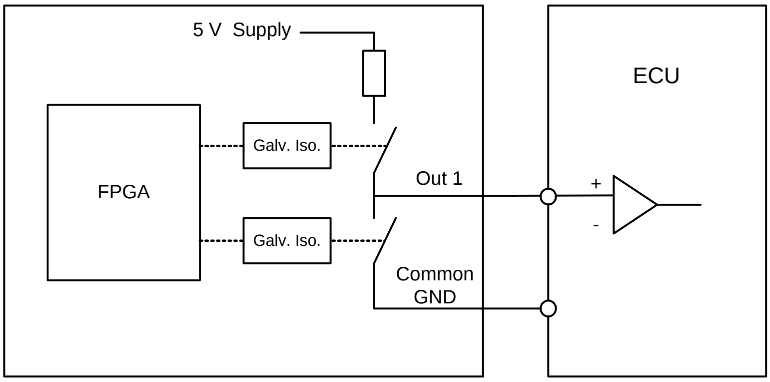
Note

In the case of the ES5340.1/.2 Electric Drive Simulation Board, digital measure values of the inverter gate can be output directly or with a delay (this is necessary, for example, for the diagnostics of some ECUs). The ES5340-Dig-In-HW RTIO element must be in the RTIO tree for this purpose.

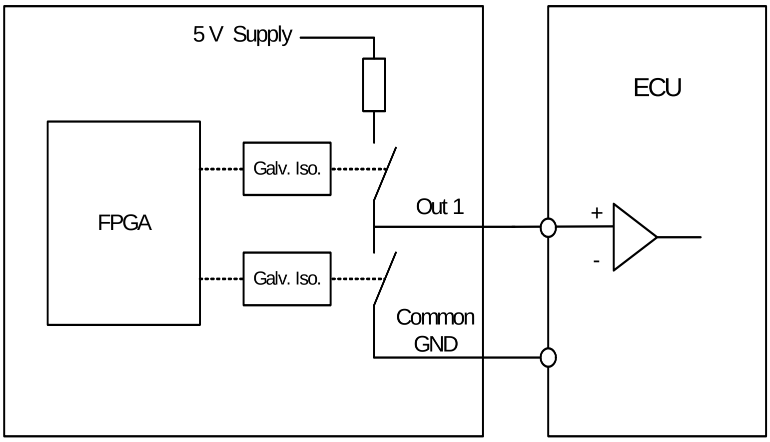
Output Mode

There are two different output modes to choose from:

- Pull-Up to +5V
  An internal pull-up resistor is switched to 5 V



- Open Collector
  An external pull-up resistor is expected here.



Delay Time

The delay with which the inverter gate measure signals are output again. Values from 0.008 µs through 5 µs are possible.

22.6.2 Data (ES5340-Digital-Out-Mux)

In this signal group, eight RTIO signals that control the closing of the output relay of each channel are processed. It is still possible to apply the outputs to GND or +5 V permanently for error simulation.

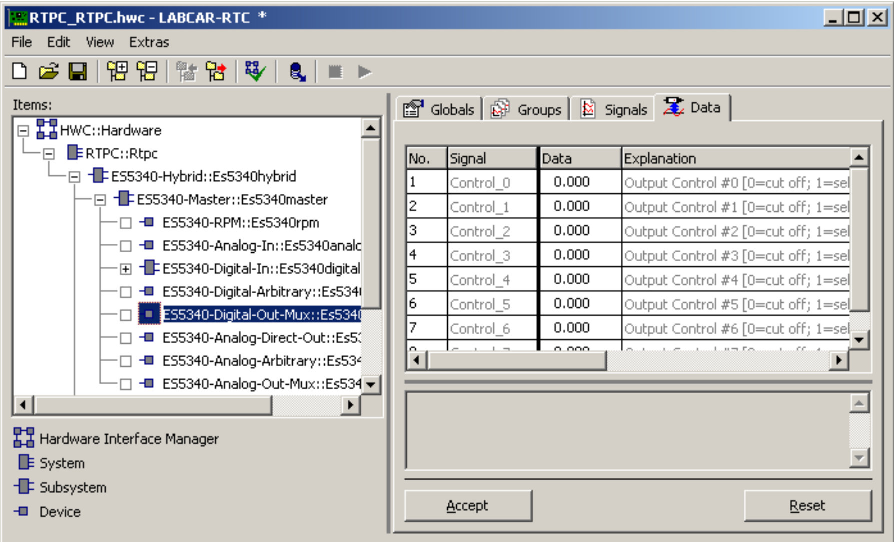


Fig. 22-34 The "Data" Tab of the ES5340-Digital-Out-Mux RTIO Element

Signal Name (n = 0...7)	Data Type*	Notes
Control_*n*	uint8	0: Output disconnected 1: Output connected 2: Shorted to GND 3: Short to +5 V
*Data type which the RTIO driver uses internally for the parameter		

Tab. 22-18 ES5340-Digital-Arbitrary: Signals of the "Data" Tab

22.7 ES5340-Analog-Out – Analog Outputs

The ES5340-Analog-Out RTIO element can be used in five different operating modes.

Mode	Description
Analog Direct Out	Output of a predefined voltage value
Sine Extrapolated	Output of a sine-shaped voltage trace
Sine Encoder	Simulation of a "sine encoder" position sensor
Resolver	Simulation of a "resolver" position sensor
Analog Arbitrary	Output of a freely configurable waveform

Tab. 22-19 Operating Modes of the ES5340-Analog-Out RTIO Element

Note

For more information on the operating modes, refer to the ES5340.1/.2 Electric Drive Simulation Board User's Guide.

The configuration of the RTIO element is different for every output mode – the appearance of the ES5340-Analog-Out RTIO element also changes accordingly.

The LABCAR Software uses a two-step procedure when outputting analog values:

1. The output values are configured via the ES5340-Analog-Out RTIO element. You can configure more signals here than there are output channels on the LABCAR Software: 8 channels of the "Analog Direct Out" type and 8 channels of the "Sine Extrapolated", "Sine Encoder", "Resolver", or "Analog Arbitrary" type.
2. The values actually output are selected via the ES5340-Analog-Out-Mux RTIO element (see "ES5340-Analog-Out-Mux – Analog Output Multiplexer" on page 737).

22.7.1 ES5340-Analog-Direct-Out

In the operating mode "Analog Direct Out", you can use one to eight analog outputs for the immediate output of an analog value.

22.7.2 Globals (ES5340-Analog-Direct-Out)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

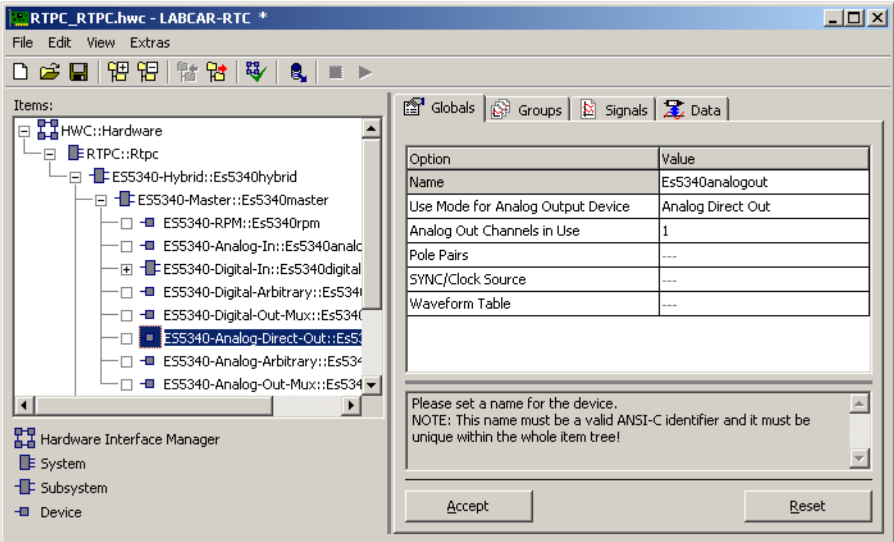


Fig. 22-35 The "Globals" Tab of the ES5340-Analog-Direct-Out RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Analog Out Channels in Use

Number of analog output channels used (max. 8).

22.7.3 Groups (ES5340-Analog-Direct-Out)

The ES5340 Analog-Direct-Out RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.7.4 Signals (ES5340-Analog-Direct-Out)

No further settings are necessary here.

22.7.5 Data (ES5340-Analog-Direct-Out)

The following figure shows the "Data" tab.

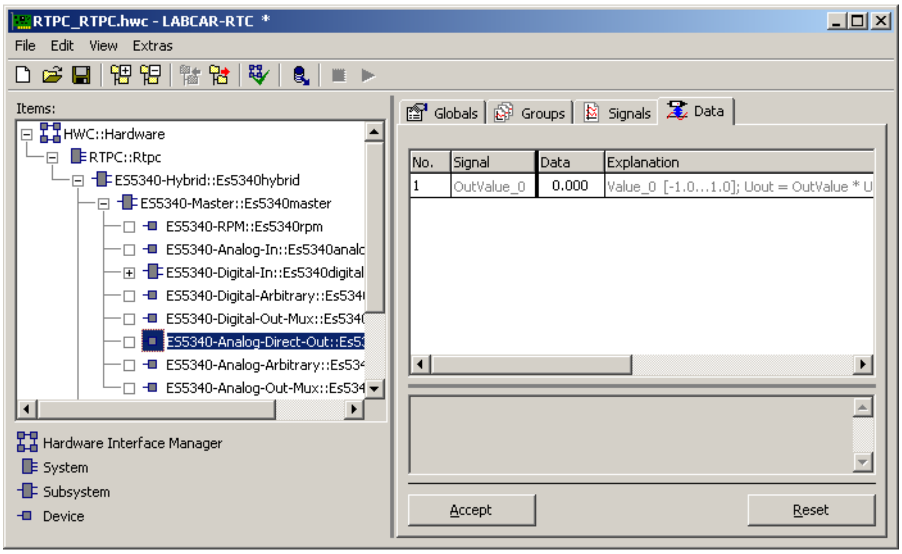


Fig. 22-36 The "Data" Tab of the ES5340-Analog-Direct-Out RTIO Element

OutValue_x

Relative amplitude of the output – is multiplied by the reference voltage specified in the analog multiplexer.

Signal Name (n = 0...5)	Data Type	Notes
OutValue_*n*	sint16	Analog output value in the range -1.0 … 1.0; Uout = OutValue * Uref

Tab. 22-20 ES5340-Analog-Direct-Out: Signals of the "Data" Tab

22.7.6 ES5340-Sine-Extrapolated

This section describes the configuration of the ES5340-Analog-Out RTIO element in the operating mode "Sine Extrapolated". This is used to output one or more sine signals which can be set independently of one another (in terms of phase and amplitude).

22.7.7 Globals (ES5340-Sine-Extrapolated)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

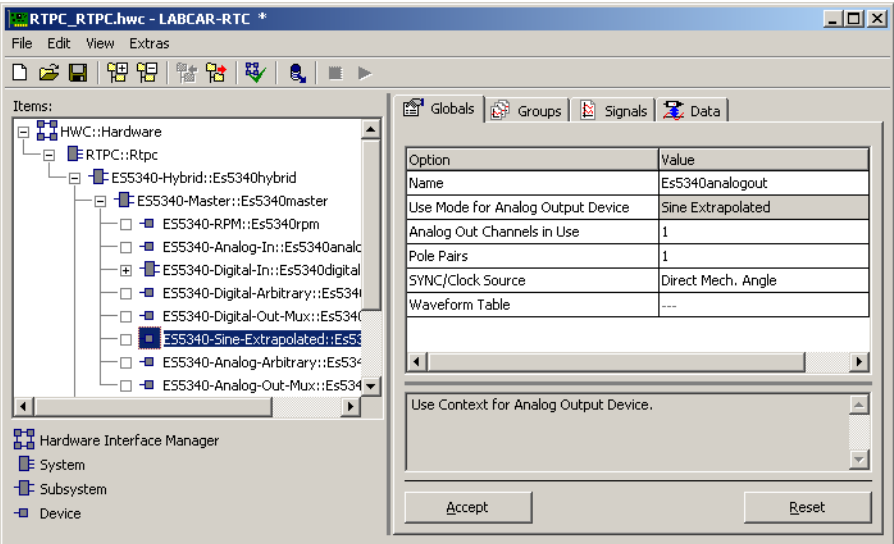


Fig. 22-37 The "Globals" Tab of the ES5340-Sine-Extrapolated RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Analog Out Channels in Use

Number of arbitrary signal generators used (max. 8). The control signals are generated in accordance with the number of channels.

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The analog signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal). The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case.

- The **local phase shift** is defined in the "Data" tab by the signal "PhaseValue_*n*".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.

- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

22.7.8 Groups (ES5340-Sine-Extrapolated)

The ES5340-Sine-Extrapolated RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.7.9 Signals (ES5340-Sine-Extrapolated)

No further settings are necessary here.

22.7.10 Data (ES5340-Sine-Extrapolated)

The following figure shows the "Data" tab.

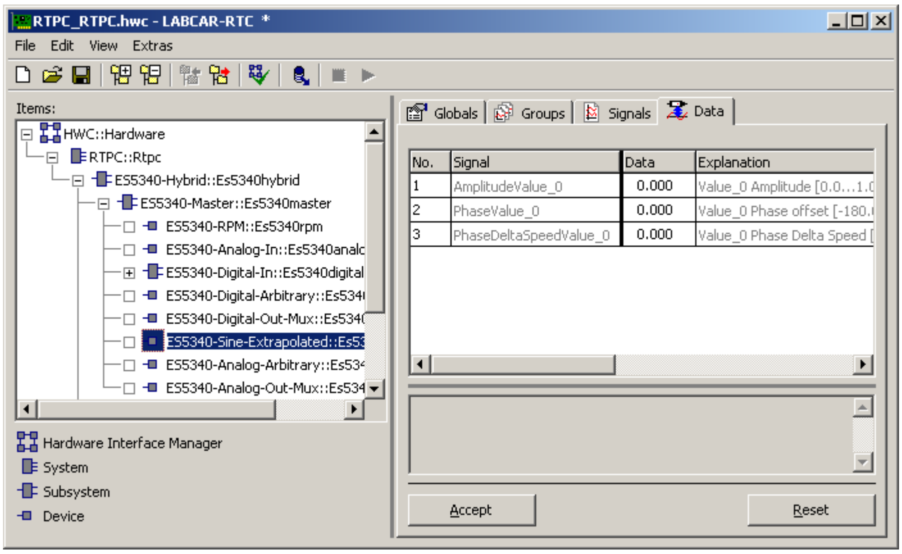


Fig. 22-38 The "Data" Tab of the ES5340-Sine-Extrapolated RTIO Element

AmplitudeValue_n (n = 0...2)

The output amplitude of the signal generator is controlled by the signal "AmplitudeValue_*n*" (value range: 0.0...1.0). The actual voltage signal is only created in the output multiplexer from this signal and the relevant reference voltage.

PhaseValue_n (n = 0...2)

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedValue_n"). This is why the signal "PhaseDeltaSpeedValue_n" must have a value greater than 0 for most applications.

PhaseDeltaSpeedValue_n (n = 0...2)

The phase change speed is controlled by the signal "PhaseDeltaSpeedValue_*n*" – a change of the signal "PhaseValue_*n*" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

Signal Name	Data Type*	Notes
AmplitudeValue_*n*	uint16	Signal amplitude of the output signal [0.0…1.0]
PhaseValue_*n*	sint16	Phase shift [-180.0…180.0 deg mech]
PhaseDeltaSpeedValue_*n*	uint16	Phase change speed [0.0…1000.0 deg mech/s]
*Data type which the RTIO driver uses internally for the parameter		

Tab. 22-21 ES5340-Sine-Extrapolated: Signals of the "Data" Tab

22.7.11 ES5340-Sine-Encoder

This section describes the configuration of the ES5340-Analog-Out RTIO element in the operating mode "sine encoder".

In this operating mode, two arbitrary signal generators are used to generate sine signals that are phase-shifted 90° to each other (sine, cosine signal). These signals are output directly and are used to simulate the relevant sensor.

The following figures show signal traces with the parameters:

- Pole Pairs: 8
- AmplitudeSine: 0.25
- OffsetSine: 0.5
- PhaseSine: -90.0°
- AmplitudeCosine: 0.25
- OffsetCosine: 0.5
- PhaseCosine: -90.0°

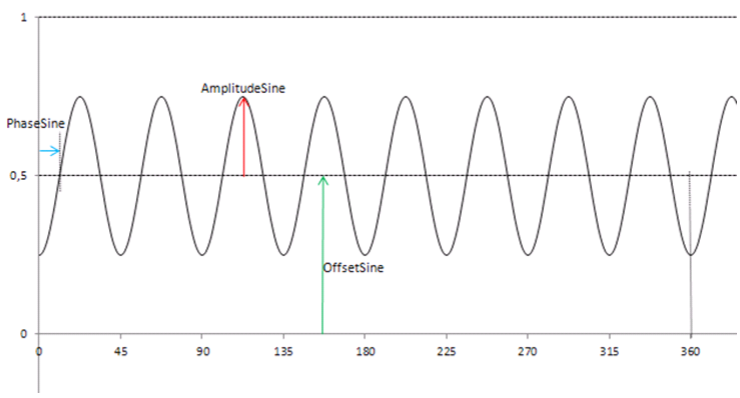


Fig. 22-39 Sine-Encoder: Sine Channel

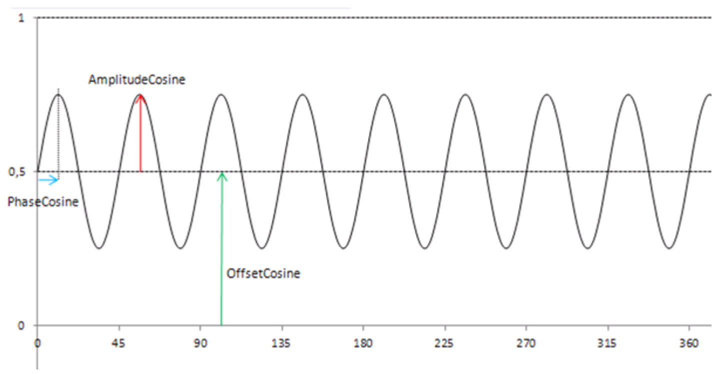


Fig. 22-40 Sine-Encoder: Cosine Channel

22.7.12 Globals (ES5340-Sine-Encoder)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

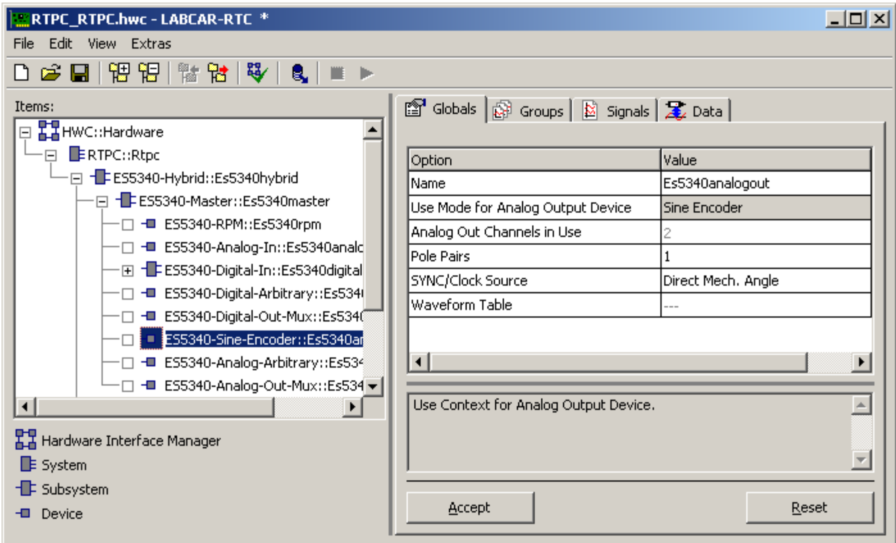


Fig. 22-41 The "Globals" Tab of the ES5340-Sine-Encoder RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The analog signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal).

The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case:

- The **local phase shift** is defined in the "Data" tab by the signals "PhaseSine" and "PhaseCosine".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.
- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

22.7.13 Groups (ES5340-Sine-Encoder)

The ES5340-Sine-Encoder RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.7.14 Signals (ES5340-Sine-Encoder)

No further settings are necessary here.

22.7.15 Data (ES5340-Sine-Encoder)

The following figure shows the "Data" tab.

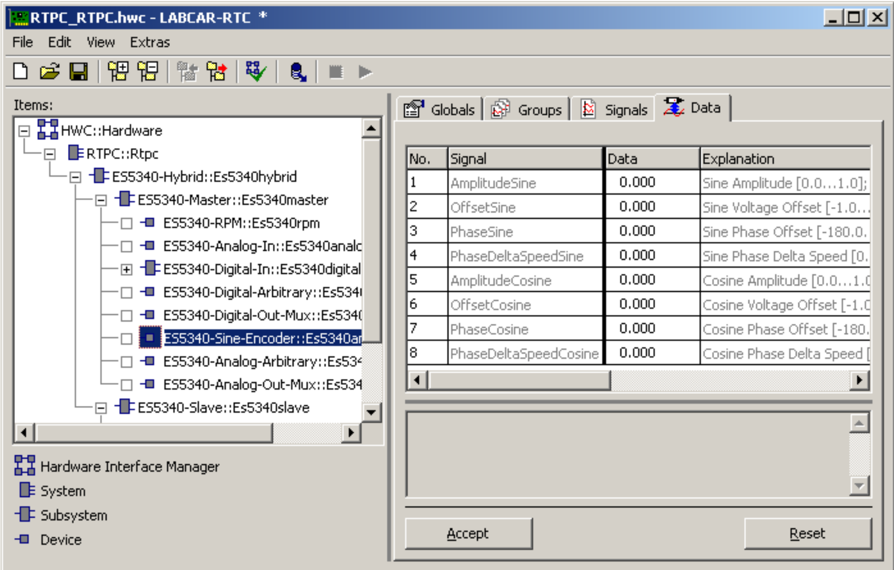


Fig. 22-42 The "Data" Tab of the ES5340-Sine-Encoder RTIO Element

AmplitudeSine

The output amplitude of the signal generator is controlled by the signal "AmplitudeSine" (value range: 0.0...1.0). The actual voltage signal is only created in the output multiplexer from this signal and the relevant reference voltage.

OffsetSine

The offset of the signal generator is controlled by the signal "OffsetSine" (value range: -1.0…1.0.) The output value is offset by the value of "OffsetSine".

The actual voltage signal is not created until the output multiplexer. If the sum of "OffsetSine" and "AmplitudeSine" exceeds the value range -1.0 through +1.0, the output value is limited to this value range.

PhaseSine

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedSine"). This is why the signal "PhaseDeltaSpeedSine" must have a value greater than 0 for most applications.

PhaseDeltaSpeedSine

The phase change speed is controlled by the signal "PhaseDeltaSpeedSine" – a change of the signal "PhaseSine" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

AmplitudeCosine

The output amplitude of the signal generator is controlled by the signal "AmplitudeCosine" (value range: 0.0...1.0). The actual voltage signal is created from this signal and the relevant reference voltage in the output multiplexer.

OffsetCosine

The offset value of the signal generator is controlled by the signal "OffsetSine" – the values range from -1.0 … 1.0. The output value is offset by the value of OffsetSine.

Please note that a voltage signal is not created until the output multiplexer from the internal signal with the value range -1.0 through +1.0. If the sum of "OffsetCosine" and "AmplitudeCosine" exceeds the value range -1.0 through +1.0, the output value is limited to this value range.

PhaseCosine

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedCosine"). This is why the signal "PhaseDeltaSpeedCosine" must have a value greater than 0 for most applications.

PhaseDeltaSpeedCosine

The phase change speed is controlled by the signal "PhaseDeltaSpeedCosine" – a change of the signal "PhaseCosine" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

Signal Name	Data Type*	Notes
AmplitudeSine	uint16	Signal amplitude of the output signal [0.0 … 1.0]
OffsetSine	sint16	Offset value for the output signal [-1.0 … 1.0]
PhaseSine	sint16	Phase shift [-180.0 … 180.0 deg mech]
PhaseDelta-SpeedSine	uint16	Phase change speed [0.0…1000.0 deg mech/s]
Amplitude-Cosine	uint16	Signal amplitude of the output signal [0.0 … 1.0]
OffsetCosine	sint16	Offset value for the output signal [-1.0 … 1.0]
PhaseCosine	sint16	Phase shift [-180.0 … 180.0 deg mech]
PhaseDelta-SpeedCosine	uint16	Phase change speed [0.0…1000.0 deg mech/s]

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-22 ES5340-Sine-Encoder: Signals of the "Data" Tab

22.7.16 ES5340-Resolver

This section describes the configuration of the ES5340-Resolver RTIO element in the operating mode "resolver".

This operating mode uses two arbitrary signal generators to generate sine signals that are phase-shifted 90° to each other (sine, cosine signal). These signals are the envelope of a higher-frequency signal that is generated using a hardware multiplier. This combination is used to simulate a "resolver" position sensor.

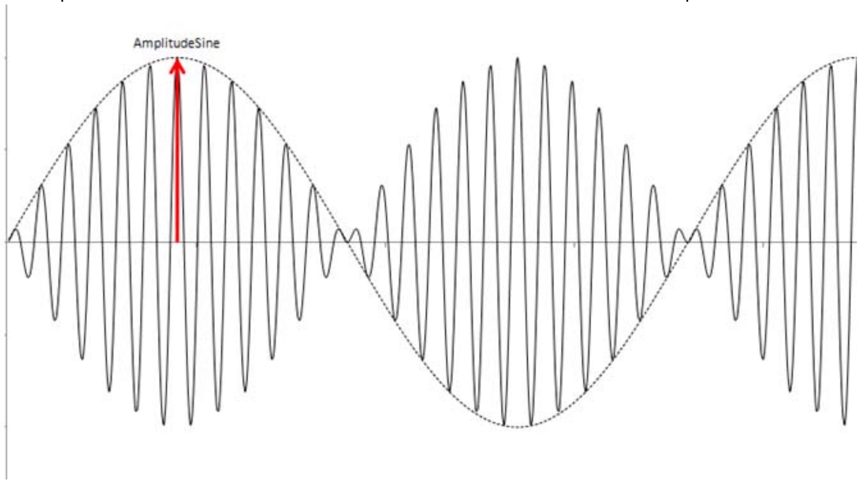


Fig. 22-43 Resolver: Sine Channel

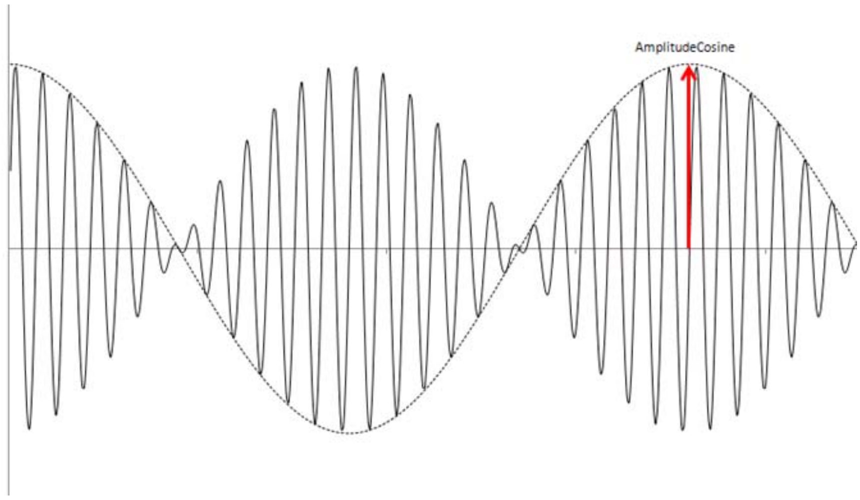


Fig. 22-44 Resolver: Cosine Channel

22.7.17 Globals (ES5340-Resolver)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

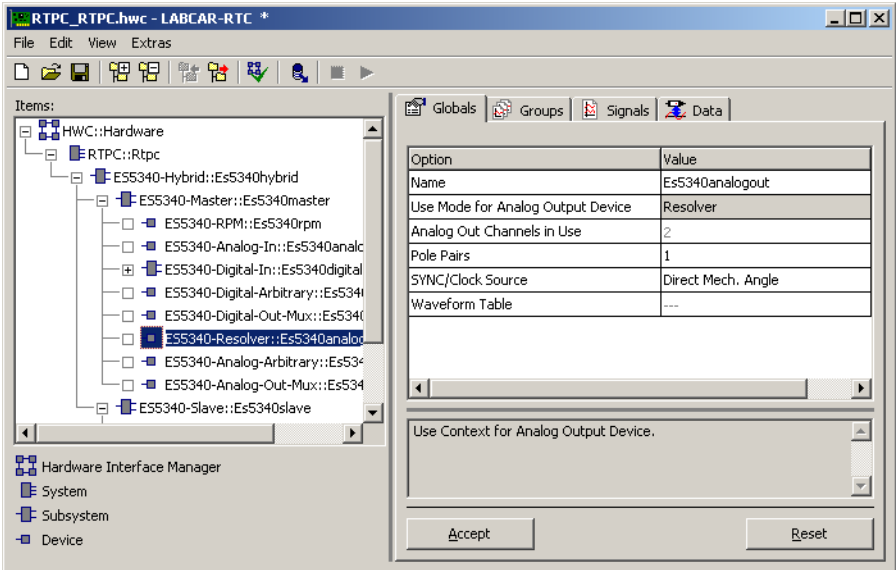


Fig. 22-45 The "Globals" Tab of the ES5340-Resolver RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Analog Out Channels in Use

This setting is of no importance in the output mode "Resolver" (internally, two arbitrary signal generators with consecutive IDs are used).

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The analog signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal). The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case:

- The **local phase shift** is defined in the "Data" tab by the signals "PhaseSine" and "PhaseCosine".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.
- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

22.7.18 Groups (ES5340-Resolver)

The ES5340-Resolver RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.7.19 Signals (ES5340-Resolver)

No further settings are necessary here.

22.7.20 Data (ES5340-Resolver)

The following figure shows the "Data" tab.

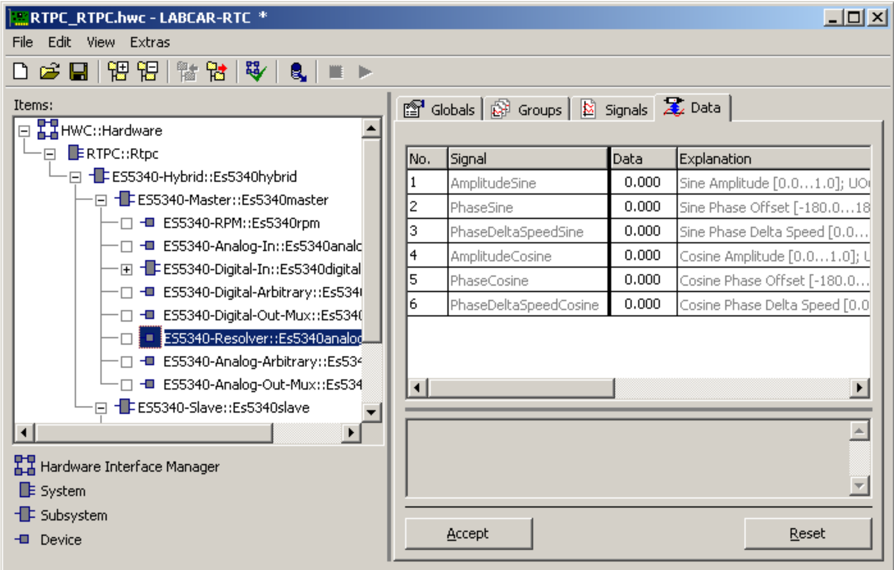


Fig. 22-46 The "Data" Tab of the ES5340-Resolver RTIO Element

AmplitudeSine

The output amplitude of the signal generator is controlled by the signal "AmplitudeSine" (value range: 0.0...1.0). The actual voltage signal is only created in the output multiplexer from this signal and the relevant reference voltage.

PhaseSine

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedSine"). This is why the signal "PhaseDeltaSpeedSine" must have a value greater than 0 for most applications.

PhaseDeltaSpeedSine

The phase change speed is controlled by the signal "PhaseDeltaSpeedSine" – a change of the signal "PhaseSine" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

AmplitudeCosine

The output amplitude of the signal generator is controlled by the signal "AmplitudeCosine" (value range: 0.0...1.0). The actual voltage signal is only created in the output multiplexer from this signal and the relevant reference voltage.

PhaseCosine

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed at a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedCosine"). This is why the signal "PhaseDeltaSpeedCosine" must have a value greater than 0 for most applications.

PhaseDeltaSpeedCosine

The phase change speed is controlled by the signal "PhaseDeltaSpeedCosine" – a change of the signal "PhaseCosine" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

Signal Name	Data Type*	Notes
AmplitudeSine	uint16	Signal amplitude of the output signal [0.0 … 1.0]
PhaseSine	sint16	Phase shift [-180.0 … 180.0 deg mech]
PhaseDelta-SpeedSine	uint16	Phase change speed [0.0…1000.0 deg mech/s]
Amplitude-Cosine	uint16	Signal amplitude of the output signal [0.0 … 1.0]
PhaseCosine	sint16	Phase shift [-180.0 … 180.0 deg mech]
PhaseDelta-SpeedCosine	uint16	Phase change speed [0.0…1000.0 deg mech/s]

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-23 ES5340-Resolver: Signals of the "Data" Tab

22.7.21 ES5340-Analog-Arbitrary

This section describes the configuration of the ES5340-Analog-Arbitrary RTIO element in the operating mode "Analog Arbitrary". In this operating mode, an arbitrary signal generator is used to generate a waveform (using the RPM generator).

22.7.22 Globals (ES5340-Analog-Arbitrary)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

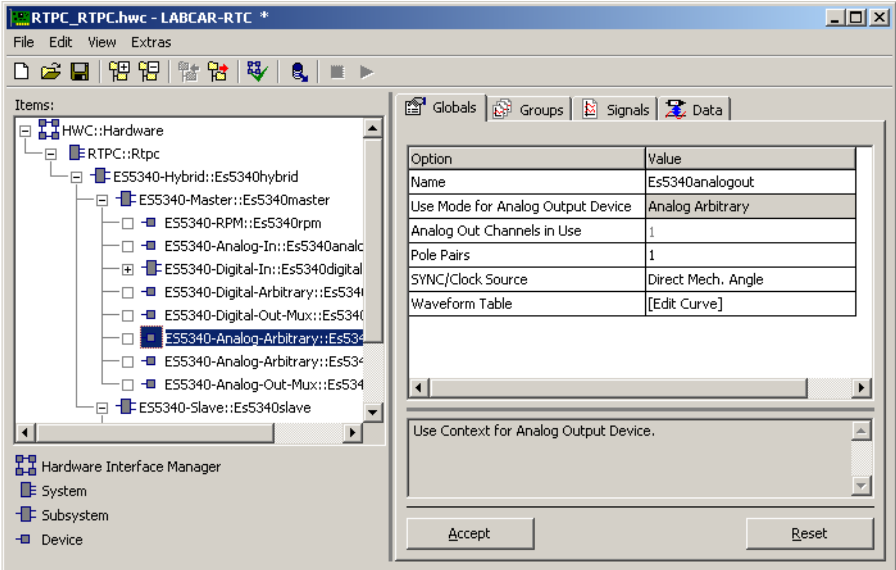


Fig. 22-47 The "Globals" Tab of the ES5340-Analog-Arbitrary RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Analog Out Channels in Use

This setting is of no importance for the output mode "Analog Arbitrary".

Pole Pairs

Number of pole pairs (max. 64).

Note

A number n of pole pairs results in an increase of the output frequency by the factor n in comparison to the basic frequency predefined by the RPM generator.

SYNC/Clock Source

The analog signals are output depending on the mechanical angle generated by the LABCAR Software and made available to the LABCAR model via the RTIO element "ES5340-RPM" ("ActMechAngle" signal). The ES5340.1/.2 Electric Drive Simulation Board makes it possible to shift the phase of the mechanical angle.

There is a local phase shift and a global phase shift in each case:

- The **local phase shift** is defined in the "Data" tab by the signal "PhaseValue".
- The **global phase shift** is defined in the RTIO element "ES5340-RPM" (in the "Data" tab) by the signal "PSMechAnglePhase".

Note

*The local phase shift refers to the phase of the generated signal – the frequency of this signal is "motor speed * number of pole pairs".*

With the concept of global phase shift, it is possible to shift the phase of a group of analog or digital outputs synchronously. There are two possible settings:

- **Direct Mech. Angle**

  Only the value of the local phase shift is used to calculate the phase.

- **Phase Shifted Mech. Angle**

  The values of both local and global phase shift are used to calculate the phase.

Waveform Table

This is where the signal form of the output signal can be specified in the form of a table. The z values of the table are scaled within the interval -1.0 … 1.0, the available x values within one cycle of the RPM generator.

22.7.23 Groups (ES5340-Analog-Arbitrary)

The ES5340-Analog-Arbitrary RTIO element has a signal group called "Outputs" for controlling the relevant signal generator.

22.7.24 Signals (ES5340-Analog-Arbitrary)

No further settings are necessary here.

22.7.25 Data (ES5340-Analog-Arbitrary)

The following figure shows the "Data" tab.

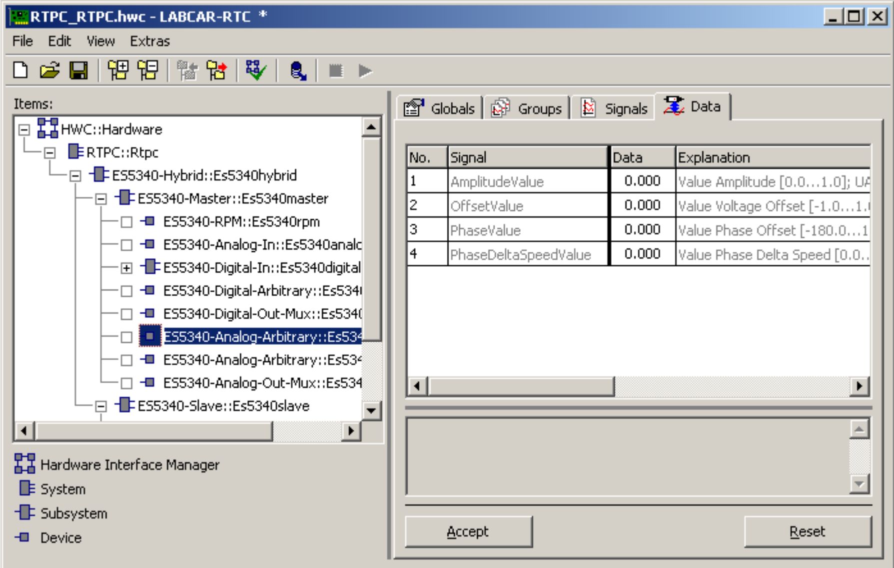


Fig. 22-48 The "Data" Tab of the ES5340-Analog-Arbitrary RTIO Element

AmplitudeValue

The output amplitude of the signal generator is controlled by the signal "AmplitudeValue" (value range: 0.0...1.0). The actual voltage signal is created from this signal and the relevant reference voltage in the output multiplexer.

OffsetValue

The offset of the signal generator is controlled by the signal "OffsetValue" (value range: -1.0…1.0). The output value is offset by the value of "OffsetValue".

The actual voltage signal is not created until the output multiplexer. If the sum of "OffsetValue" and "AmplitudeValue" exceeds the value range -1.0 through +1.0, the output value is limited to this value range.

PhaseValue

This signal describes a local phase shift with which a model-controlled phase shift can be executed during runtime.

Note

The phase shift is executed with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeedValue"). This is why the signal "PhaseDeltaSpeedValue" must have a value greater than 0 for most applications.

PhaseDeltaSpeedValue

The phase change speed is controlled by the signal "PhaseDeltaSpeedValue" – a change of the signal "PhaseValue" does not have an infinitely fast effect, but takes place at this speed.

The values range from 0 (the phase change is erratic) to 1000° mech/s.

Signal Name	**Data Type***	**Notes**
AmplitudeValue	uint16	Signal amplitude of the output signal [0.0…1.0]
OffsetValue	sint16	Offset for the output signal [-1.0…1.0]
PhaseValue	sint16	Phase shift [-180.0…180.0 deg mech]
PhaseDeltaSpeedValue	uint16	Phase change speed [0.0…1000.0 deg mech/s]

*Data type which the RTIO driver uses internally for the parameter

Tab. 22-24 ES5340-Analog-Arbitrary: Signals of the "Data" Tab

22.8 ES5340-Analog-Out-Mux – Analog Output Multiplexer

In the RTIO element "ES5340-Analog-Out-Mux", signals configured in the RTIO element "ES5340-Analog-Out" are assigned to the analog signal outputs that actually exist in the ES5340.1/.2 Electric Drive Simulation Board.

22.8.1 Signals (ES5340-Analog-Out-Mux)

The eight (ES5340-Master) or six (ES5340-Slave) analog outputs are configured in the "Signals" tab. The properties of the analog output as well as the source of the output value can be determined here.

The following figure shows the RTIO parameters of the "Signals" tab – the properties of the individual parameters are listed in Tab. 22-25 on page 739. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

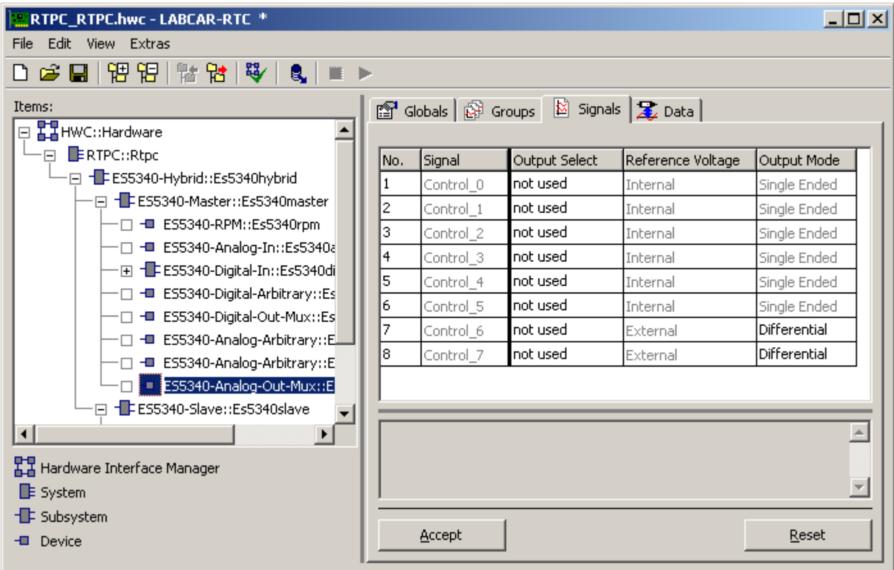


Fig. 22-49 The "Signals" Tab of the ES5340-Analog-Out-Mux RTIO Element

Output Select

The "Output Select" option field is used to select the internal signal that is to be applied at the relevant output. A list of all internal signal sources is available for selection. The list includes the possible output values of all Analog-Out RTIO elements and possibly analog output values of an FPGA model.

The number of possible output values of an Analog-Out RTIO element depends on its assignment of the arbitrary generators and direct output channels.

If an FPGA model provides analog output values, these can also be output using a connected LABCAR Software-Slave board. This requires the two boards to be connected to each other via Gigabit Link.

Note

Only channels 6 and 7 can be used to generate the resolver signals as this requires special signal conditioning on the LABCAR Software.

Reference Voltage

This option field is used to select the reference voltage for the D/A converter of the relevant output channel. Possible values are "internal" (for an internal reference voltage of 10 V) and "external" (for an externally pending reference voltage).

Output Mode

This parameter can only be modified for channels 6 and 7. You can choose between a single ended and a differential output signal.

Note

Choose "differential" to ensure resolver signal generation works correctly.

The following combinations are permissible for the two parameters:

- Reference Voltage: Internal / Output Mode: Single Ended
- Reference Voltage: External / Output Mode: Single Ended
- Reference Voltage: External / Output Mode: Differential

The Output Mode for channels 6 and 7 is always identical.

22.8.2 Data (ES5340-Analog-Out-Mux)

In this signal group, eight RTIO signals that control the closing of the output relay of each channel are processed. It is still possible to apply the outputs to GND or Uref permanently for error simulation.

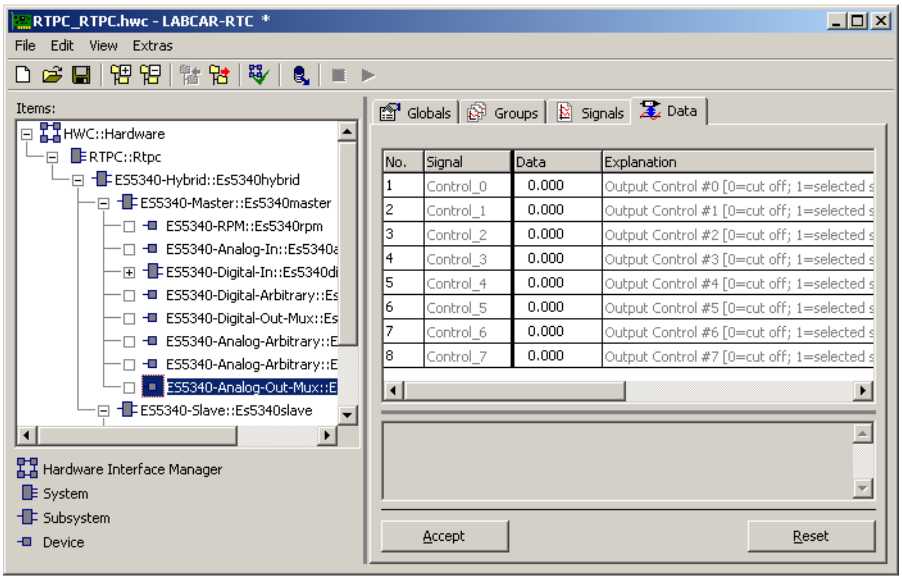


Fig. 22-50 The "Data" Tab of the ES5340-Analog-Out-Mux RTIO Element

Signal Name (n = 0...7)	Data Type*	Notes
Control_*n*	uint8	0 = cut off 1 = selected source 2 = connected with GND 3 = connected with URef
*Data type which the RTIO driver uses internally for the parameter		

Tab. 22-25 ES5340-Analog-Out-Mux: Signals of the "Data" Tab

Note

If the Output Mode is set to "Differential", a direct voltage cannot be output.

If at the same time "Reference Voltage" is set to "Internal", 0 V is output in case 3 (shorted to Uref). If, however, "Reference Voltage" is set to "External", the AC voltage at the external reference input ("Excitation P/N") is output with full amplitude in case 3 (shorted to Uref).

22.9 ES5340-Slave

The ES5340-Slave subsystem is used to configure an ES5340-Slave board.

22.10 ES5340-Digital-Out (Slave)

With the slave board, digital signals can only be output in the operating mode "Digital Direct Out" with a max. of six channels.

For a description of the ES5340-Digital-Direct-Out RTIO element, refer to the section "ES5340-Digital-Direct-Out" on page 702.

22.11 ES5340-Digital-Out-Mux (Slave)

In the RTIO element "ES5340-Digital-Out-Mux", the signals to be issued at the six digital signal outputs of the slave board are selected.

The following signals can be selected:

- The signals of the type "Digital Direct Out" of the slave board
- The digital output values of a model of the master board

For more information on configuring signal outputs, refer to the section "Signals (ES5340-Digital-Out-Mux)" on page 714.

22.12 ES5340-Analog-Out (Slave)

With the slave board, analog signals can only be output in the operating mode "Analog Direct Out" with a max. of six channels.

For a description of the ES5340-Analog-Direct-Out RTIO element, refer to the section "ES5340-Analog-Direct-Out" on page 718.

22.13 ES5340-Analog-Out-Mux (Slave)

In the RTIO element "ES5340-Analog-Out-Mux", the signals to be issued at the six analog signal outputs of the slave board are selected.

The following signals can be selected:

- The signals of the type "Analog Direct Out" of the slave board
- The analog output values of a model of the master board

For more information on configuring signal outputs, refer to the section "Signals (ES5340-Analog-Out-Mux)" on page 737.

22.14 ES5340-PMSM-1.0.0 – PMSM-FPGA Model

The ES5340-PMSM-1.0.0 RTIO element is used to configure the Inverter, PMSM and Mechanics model implemented in the FPGA of the LABCAR Software.

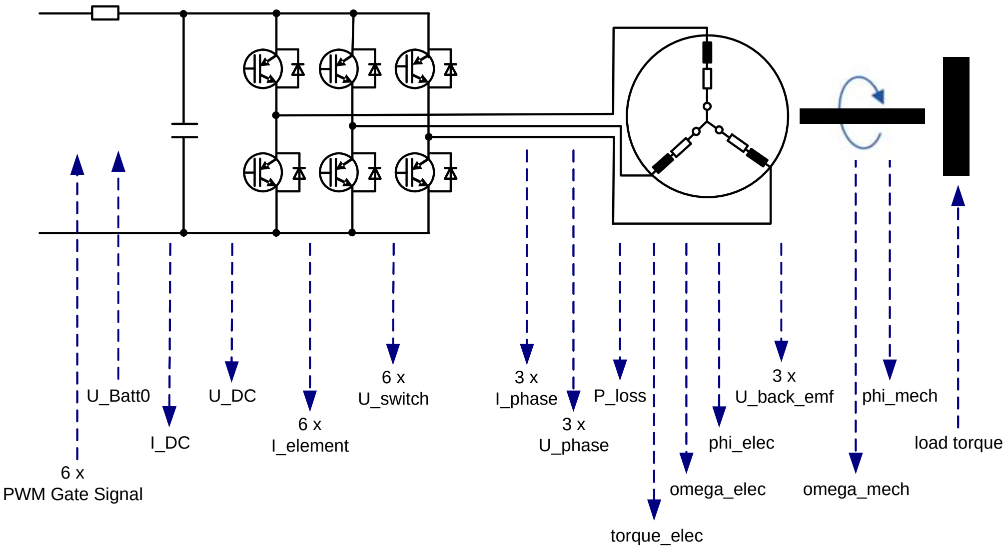


Fig. 22-51 Block Diagram of the ES5340-PMSM-1.0.0 Model

22.14.1 Globals (ES5340-PMSM-1.0.0)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

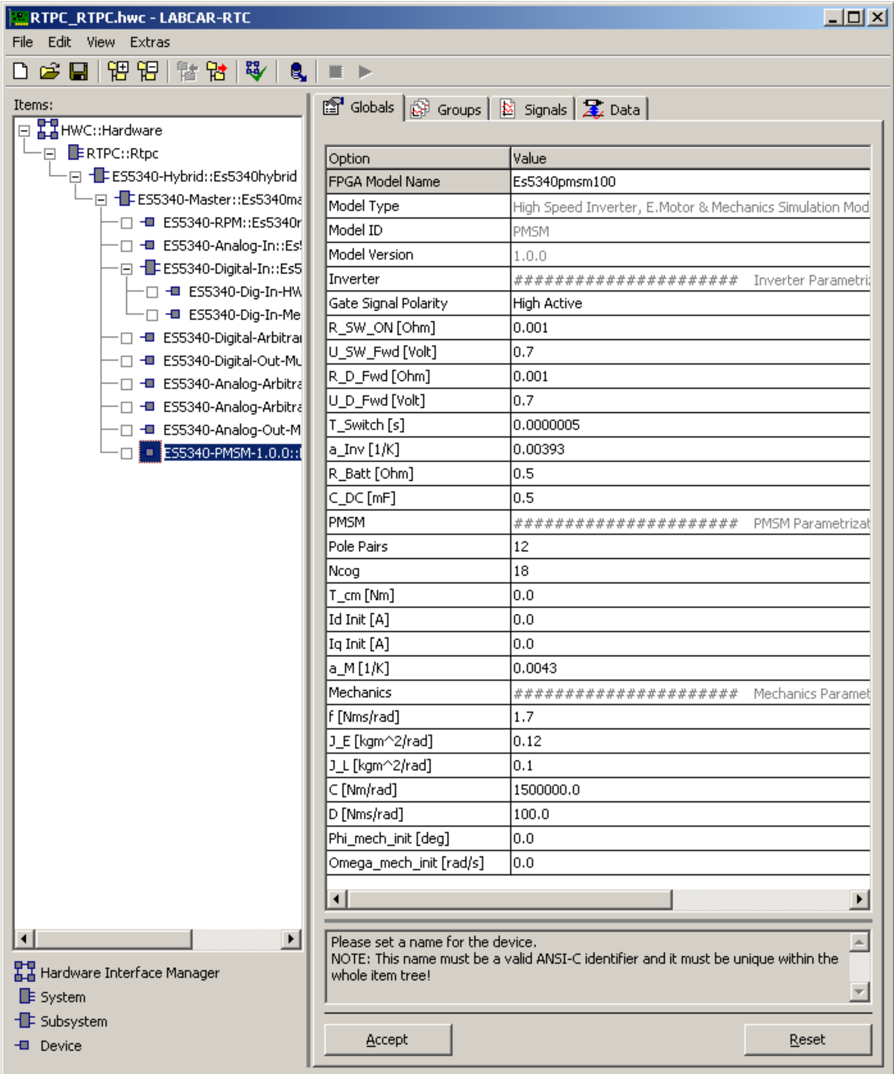


Fig. 22-52 The "Globals" Tab of the ES5340-PMSM-1.0.0 RTIO Element

The "Globals" tab is divided into four sections:

- General Information
- Inverter Model
- PMSM Model
- Mechanics Model

22.14.2 General Information

FPGA Model Name

Name of the model in the RTIO.

Model Type

Display of the model type.

Model ID

Displays the model ID.

Model Version

Displays the model version.

22.14.3 Inverter Model

Gate Signal Polarity

This setting is used to determine the polarity of the inverter gate signals applied to 6 (of 20) digital inputs.

R_SW_ON

Resistance value of a conductive inverter switch at 20 °C.

U_SW_FWD

Voltage in forward direction via a conductive inverter switch.

R_D_FWD

Resistance value of the free-wheeling diode of a conductive inverter switch at 20°C.

U_D_FWD

Voltage in forward direction via a diode with a conductive inverter switch.

T_Switch

Switch time of an inverter switch.

a_Inv

Temperature coefficient of the ohmic values of the inverter. This coefficient makes it possible to represent temperature dependency for R_SW_ON and R_D_SW as follows:

```
R_SW_ON(Theta_Inv) =
R_SW_On(20 °C) + (1 + a_Inv * (Theta_Inv - 20 °C) )
```

```
R_D_FWD(Theta_Inv) =
R_D_FWD(20 °C) + ( 1 + a_Inv * (Theta_Inv - 20 °C) )
```

R_Batt

Internal resistance of the battery. No temperature dependency is simulated for this variable.

If this value is set to 0, the entire RC circuit (comprising R_Batt and C_DC) is disabled regardless of the value of the variable C_DC.

If the RC circuit is disabled, the following is true:

```
I_Batt = I_DC
```

```
U_DC = U_Batt0
```

C_DC

Capacity of the DC voltage link.

If this value is set to 0, the entire RC circuit (comprising R_Batt and DC) is disabled regardless of the value of the variable R_Batt.

```
I_Batt = I_DC
U_DC = U_Batt0
```

22.14.4 PMSM Model

Pole Pairs

The number of pole pairs of the permanent magnet synchronous machine.

Ncog

The number of cogging positions of the permanent magnet synchronous machine.

T_cm

The peak value of the cogging torque of the permanent magnet synchronous machine.

Id_Init

d component of the stator current initialization value in the d-q coordinate system that is fixed with regard to the rotor.

Iq_Init

q component of the stator current initialization value in the d-q coordinate system that is fixed with regard to the rotor.

a_M

Temperature coefficient of the ohmic values of the permanent magnet synchronous machine. This coefficient makes it possible to represent temperature dependency for the coil resistance R_d and/or R_q as follows:

```
R_d(Theta_M) = R_d(20 °C) + ( 1 + a_M * (Theta_M - 20 °C))
R_q(Theta_M) = R_q(20 °C) + ( 1 + a_M * (Theta_M - 20 °C))
```

R_d(Theta_M) and R_q(Theta_M) are the variables that have an effect in the model. These are calculated in accordance with the above formulae depending on the resistance values at 20 °C (R_d(20 °C) and R_q(20 °C), on the current machine temperature Theta_M and on the temperature coefficient a_M (see Tab. 22-29 on page 753).

22.14.5 Mechanics Model

The Mechanics model simulates a two-mass system comprising a PMSM machine and load.

The inertia of these two masses is defined by the variables J_E (PMSM machine) and J_L (load or drivetrain) – the friction of the overall system is set via the parameter f.

The two masses are connected to each other via a spring-and-damper element (spring constant C, damping D).

f

Friction of the mechanical overall system.

J_E

Mass inertia of the machine.

J_L

Mass inertia of the load or drivetrain.

C

Torsional stiffness or spring constant of the connecting element between two masses.

D

Damping of the connecting element between two masses.

Phi_mech_init

Initialization value of the mechanical position angle of the electric machine/rotor.

Omega_mech_init

Initialization value of the mechanical angle speed of the electric motor/rotor.

The following table lists the configuration parameters of the model with data type, value range and physical unit.

Parameter	Data Type	Can be edited online	Notes
Gate Signal Polarity	int	Yes	High Active Low Active
R_SW_ON	float	Yes	0.0 .. 0.05 Ω
U_SW_FWD	float	Yes	0.0 .. 3.0 V
R_D_FWD	float	Yes	0.0 .. 0.05 Ω
U_D_FWD	float	Yes	0.0 .. 3.0 V
T_Switch	float	Yes	0.0000002 .. 0.000001 s
a_Inv	float	Yes	0.0 .. 0.01 1/K
R_Batt	float	Yes	0.01 .. 10.0 Ω
C_DC	float	Yes	0.01 .. 10.0 mF
Pole Pairs	float	Yes	1 .. 64
Ncog	float	Yes	1 .. 50
T_cm	float	Yes	0.0 .. 100.0 Nm
Id_Init	float	Yes	0.0 .. 1500.0 A
Iq_Init	float	Yes	0.0 .. 1500.0 A
a_M	float	Yes	0.0 .. 0.01 1/K
f	float	Yes	0.0 .. 10.0 Nms
J_E	float	Yes	0.0 .. 1.0 kgm^2
J_L	float	Yes	0.0 .. 10.0 kgm^2
C	float	Yes	0.0 .. 10000000.0 Nm/rad
D	float	Yes	0.0 .. 1000.0 Nms/rad
Phi_mech_init	float	Yes	0.0 .. 360.0 °
Omega_mech_init	float	Yes	-833.333 .. +833.333 rad/s

Tab. 22-26 ES5340-PMSM-1.0.0 RTIO Element: Configuration Parameters of the "Globals" Tab

22.14.6 Groups (ES5340-PMSM-1.0.0)

The ES5340-PMSM-1.0.0 RTIO element has a signal group called "Outputs" for controlling the engine model and a signal group called "Inputs" for reading the current engine model status information and simulation values.

22.14.7 Signals (ES5340-PMSM-1.0.0)

In the "Signals" tab, you can specify which value range is mapped to which voltage area for every analog model output value when this value is selected for analog output in the analog output multiplexer.

The physical value "Model Value (min)" is always mapped to the voltage value "DAC Voltage (min)" and the physical value "Model Value (max)" to the voltage value "DAC Voltage (max)".

The Min/Max values for "DAC Voltage" also limit the output voltage – voltage values outside this range cannot be set. If the physical value range of a simulation value is exceeded, the Min/Max values for "DAC Voltage" represent an absolute limit.

Note

If, however, error state 3 (shorted to Uref) is activated in the analog output multiplexer, this restriction has no effect! This means that the voltage defined by the error is applied at the output and could exceed the above-mentioned restriction!

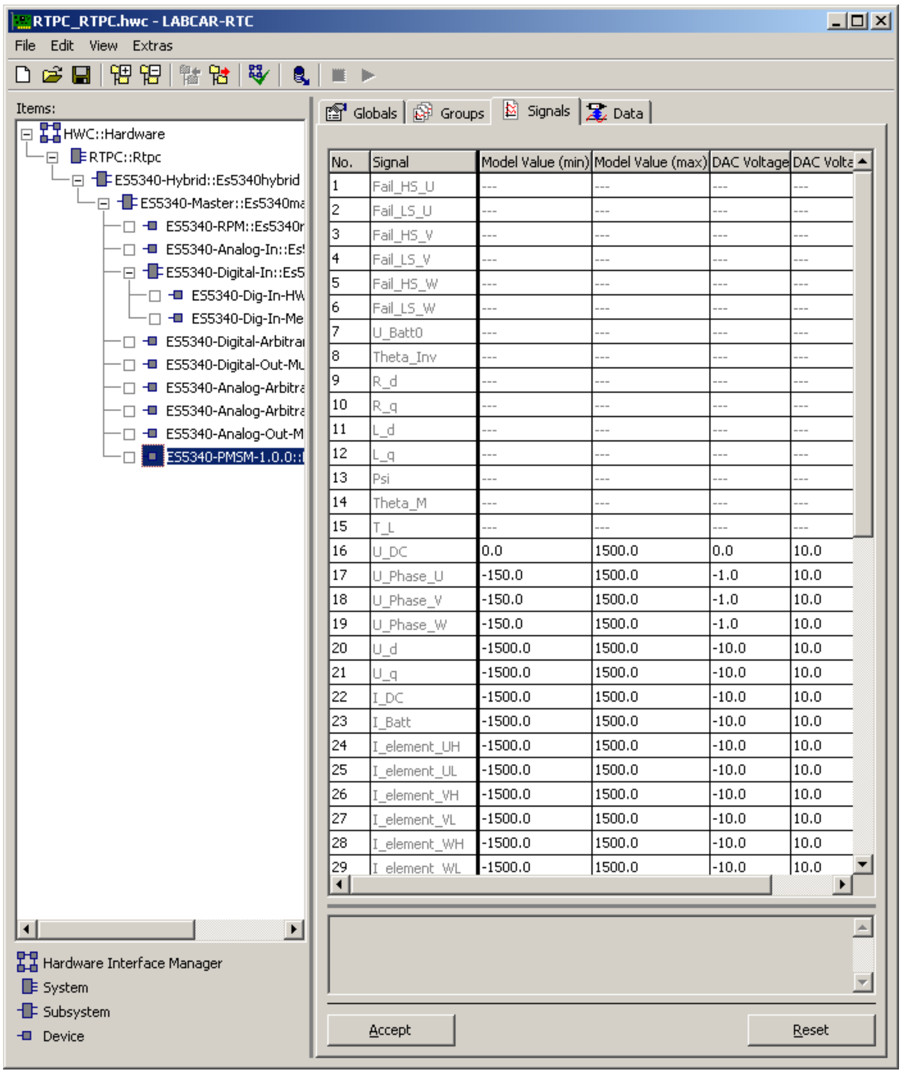


Fig. 22-53 The "Signals" Tab of the ES5340-PMSM-1.0.0 RTIO Element (Part 1)

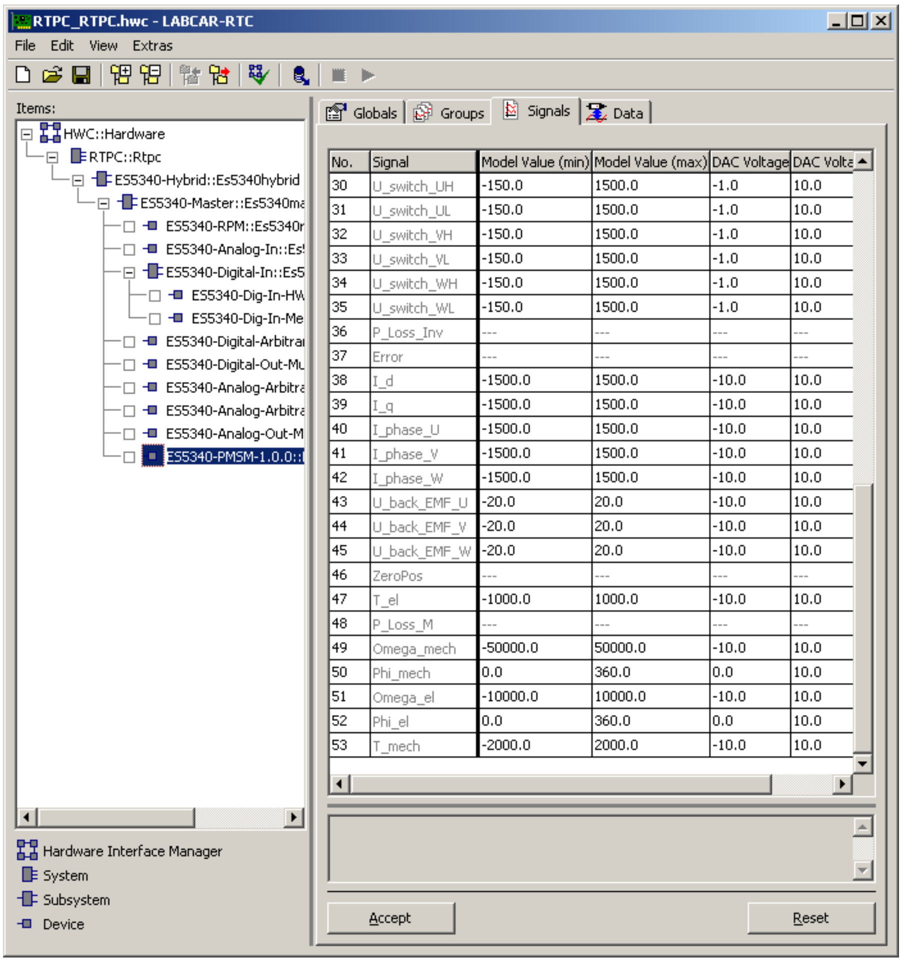


Fig. 22-54 The "Signals" Tab of the ES5340-PMSM-1.0.0 RTIO Element (Part 2)

The following table shows the configuration parameters of the "Signals" tab with data type and value range.

Parameter	Data Type	Can be edited online	Notes
Model Value (min)	float	No	Value range depending on the model variable
Model Value (max)	float	No	Value range depending on the model variable
DAC Voltage (min)	float	No	Value range: -10.0 V .. +10.0 V
DAC Voltage (max)	float	No	Value range: -10.0 V .. +10.0 V

Tab. 22-27 ES5340-PMSM-1.0.0 RTIO Element: Configuration Parameters of the "Signals" Tab

The following table contains the value ranges of the scaling parameters of analog model variables to analog output voltage ranges:

Parameter	Model Value (min)	Model Value (max)	DAC Volt-age (min)	DAC Volt-age (max)	Meaning
U_DC	0	1500	-10	10	Voltage at the intermediate circuit capacitor [V]
U_Phase_x*	-150	1500	-10	10	Phase voltage of phase x with respect to the ground potential of the intermedi-ate circuit [V]
U_d	-1500	1500	-10	10	d component of the volt-ages in the d-q coordinate system that is fixed with regard to the rotor [V]
U_q	-1500	1500	-10	10	q component of the volt-ages in the d-q coordinate system that is fixed with regard to the rotor [V]
I_DC	-1500	1500	-10	10	Current into the inverter [A]
I_Batt	-1500	1500	-10	10	Current through the simu-lated battery internal resis-tance [A]
I_element_xH*, I_element_xL*	-1500	1500	-10	10	Element current through the high-side or low-side path of phase x [A]
U_switch_xH*, U_switch_xL*	-150	1500	-10	10	Voltage over the high-side or low-side switch of phase x [A]
I_d	-1500	1500	-10	10	d component of the cur-rents in the d-q coordinate system that is fixed with regard to the rotor [A]
I_q	-1500	1500	-10	10	q component of the cur-rents in the d-q coordinate system that is fixed with regard to the rotor [A]
I_phase_x*	-1500	1500	-10	10	Phase current in phase x [A]
U_back_EMF_x*	-1500	1500	-10	10	Induced voltage phase x [U]
T_el	-1000	1000	-10	10	Electrically generated torque [Nm]
Omega_mech	-50000	50000	-10	10	Mechanical angle speed [rad/s]
Phi_mech	0	360	-10	10	Mechanical angle, rotor angle [°]
Omega_el	-50000	50000	-10	10	Electric angle speed [rad/s]

Parameter	Model Value (min)	Model Value (max)	DAC Volt-age (min)	DAC Volt-age (max)	Meaning
Phi_el	0	360	-10	10	Electric angle [°]
T_mech	-2000	2000	-10	10	Actual effective mechanical torque [Nm]

*x = U,V,W

Tab. 22-28 ES5340-PMSM-1.0.0 RTIO Element: Value Ranges for Signals

Note

DAC Voltage (min) must always be less than DAC Voltage (max).

22.14.8 Data (ES5340-PMSM-1.0.0)

The following figure shows the "Data" tab.

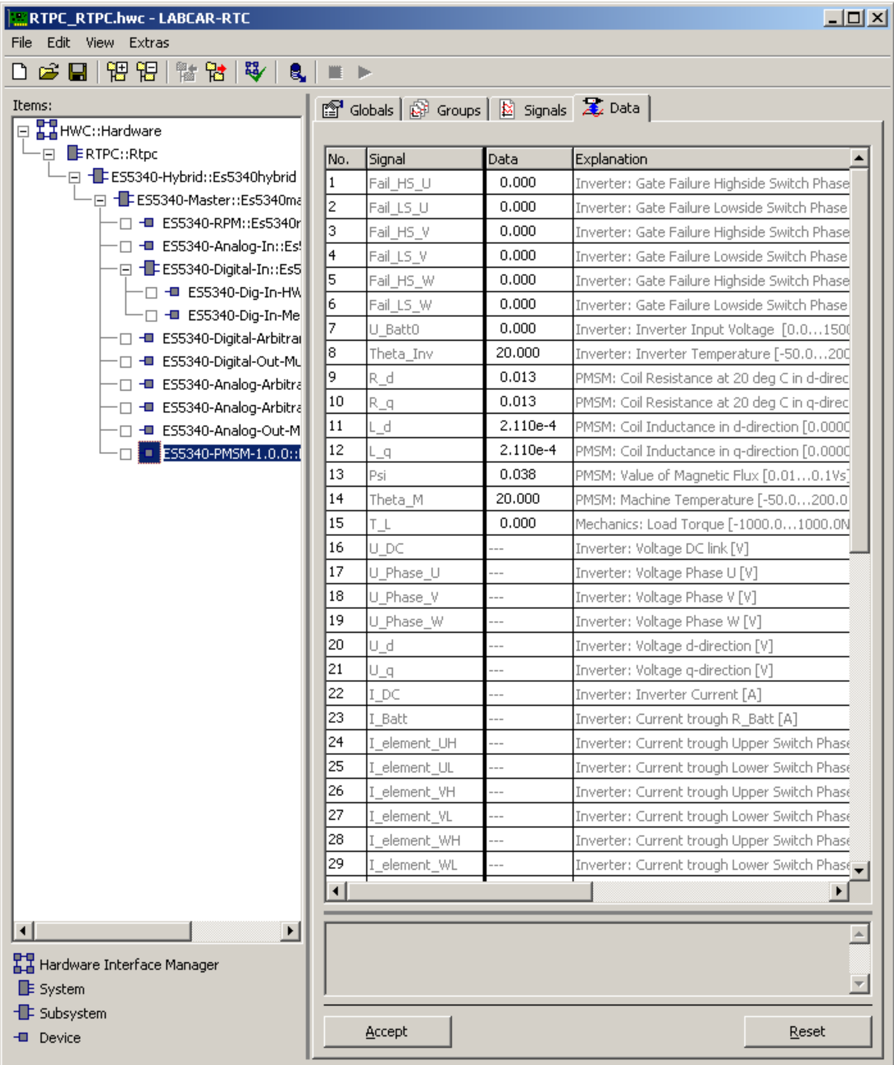


Fig. 22-55 The "Data" Tab of the ES5340-PMSM-1.0.0 RTIO Element (Part 1)

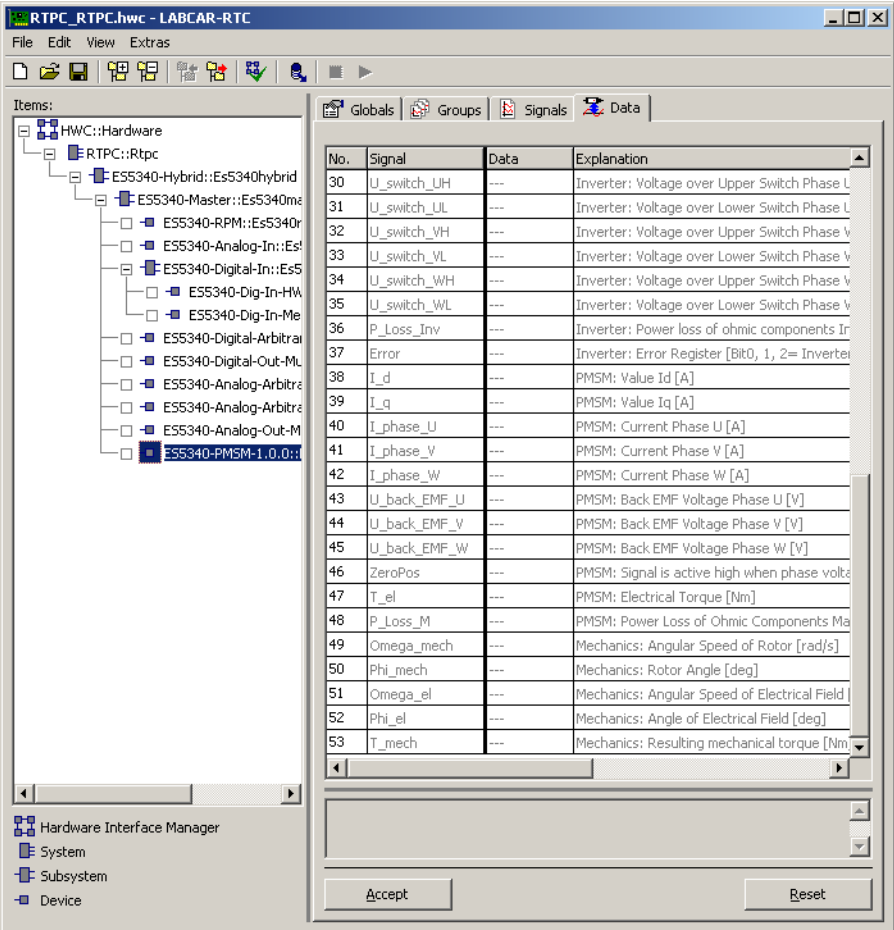


Fig. 22-56 The "Data" Tab of the ES5340-PMSM-1.0.0 RTIO Element (Part 2)

The following table shows the signals of the "Data" tab.

Signal Name	Data Type	Direction	Notes
Fail_HS_x*	float	Input	Inverter: Gate error high-side switch phase x* 0 = normal function 1 = switch permanently off 2 = switch permanently on
Fail_LS_x*	float	Input	Inverter: Gate error low-side switch phase x* 0 = normal function 1 = switch permanently off 2 = switch permanently on
U_Batt0	float	Input	DC voltage link input voltage/ battery voltage Value range: 0.0 .. 1500.0 V
Theta_Inv	float	Input	Inverter temperature Value range: -50.0 .. 200.0 °C
R_d	float	Input	d component of the stator coil resistance at 20 °C Value range: 0.01 .. 1.0 Ω
R_q	float	Input	q component of the stator coil resistance at 20 °C Value range: 0.01 .. 1.0 Ω
L_d	float	Input	d component of the stator coil inductance Value range: 0.00008 .. 0.001 H
L_q	float	Input	q component of the stator coil inductance Value range: 0.00008 .. 0.001 H
Psi	float	Input	Magnetic flow caused by permanent magnet of the machine Value range: 0.01 .. 0.1 Vs
Theta_M	float	Input	Temperature in the machine (for calculating the heat dependency of ohmic values) Value range: -50.0 .. 200.0 °C
T_L	float	Input	Load torque Positive value: Torque works in a negative direction of rotation, brake torque Negative value: Torque works in a positive direction of rotation, accelerating torque Value range: -1000.0 .. 1000.0 Nm
U_DC	float	Output	Voltage at the DC voltage link capacitor [V]
U_Phase_x*	float	Output	Phase voltage of phase x with respect to the ground potential of the DC voltage link [V]
U_d	float	Output	d component of the voltages in the d-q coordinate system that is fixed with regard to the rotor [V]
U_q	float	Output	q component of the voltages in the d-q coordinate system that is fixed with regard to the rotor [V]
I_DC	float	Output	Current into the inverter [A]
I_Batt	float	Output	Current through the simulated battery internal resistance [A]

Signal Name	Data Type	Direction	Notes
I_element_xH*, I_element_xL*	float	Output	Element current through the high-side or low-side path of phase x [A]
U_switch_xH*, U_switch_xL*	float	Output	Voltage over the high-side or low-side switch of phase x [A]
P_Loss_Inv	float	Output	Power loss of the ohmic components of the inverter [W]
Error	float	Output	Bit vector in floating-point representation: Bit 0: Inverter short high-side/low-side phase U Bit 1: Inverter short high-side/low-side phase V Bit 2: Inverter short high-side/low-side phase W
I_d	float	Output	d component of the currents in the d-q coordinate system that is fixed with regard to the rotor [A]
I_q	float	Output	q component of the currents in the d-q coordinate system that is fixed with regard to the rotor [A]
I_phase_x*	float	Output	Phase current in phase x [A]
U_back_EMF_x*	float	Output	Induced voltage in phase x [V]
ZeroPos	float	Output	1, if U_back_EMF_U >= U_back_EMF_V 0, if U_back_EMF_U < U_back_EMF_V
T_el	float	Output	Electrically generated torque [Nm]
P_Loss_M	float	Output	Power loss of the ohmic components of the machine [W]
Omega_mech	float	Output	Mechanical angle speed [rad/s]
Phi_mech	float	Output	Mechanical angle, rotor angle [°]
Omega_el	float	Output	Electric angle speed [rad/s]
Phi_el	float	Output	Electric angle [°]
T_mech	float	Output	Actual effective mechanical torque [Nm]
*x = U,V,W			

Tab. 22-29 ES5340-PMSM-1.0.0 RTIO Element: Signals of the "Data" Tab

22.15 ES5340-IM-1.0.0 – IM-FPGA Model

The ES5340-IM-1.0.0 RTIO element is used to configure the inverter, IM and mechanics model implemented in the FPGA of the LABCAR Software.

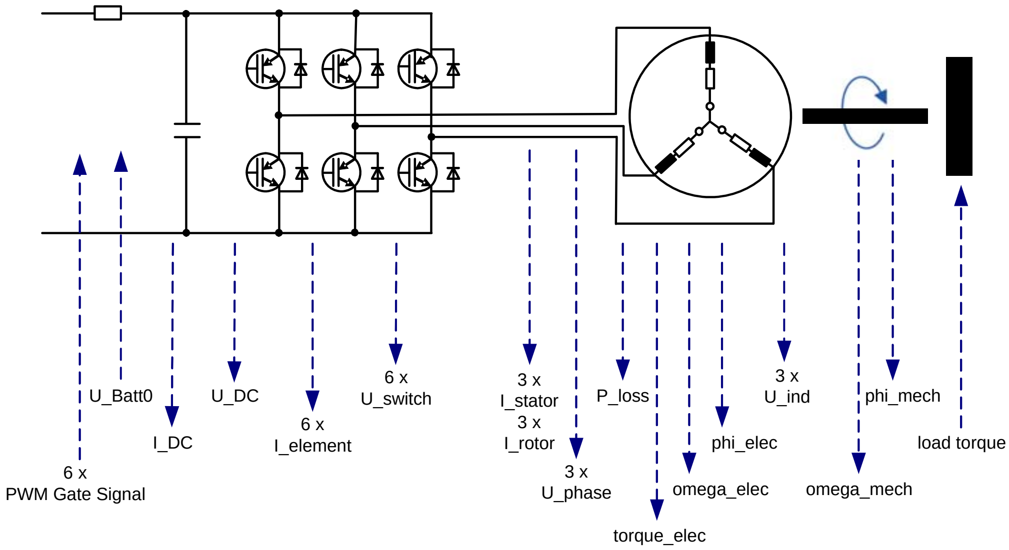
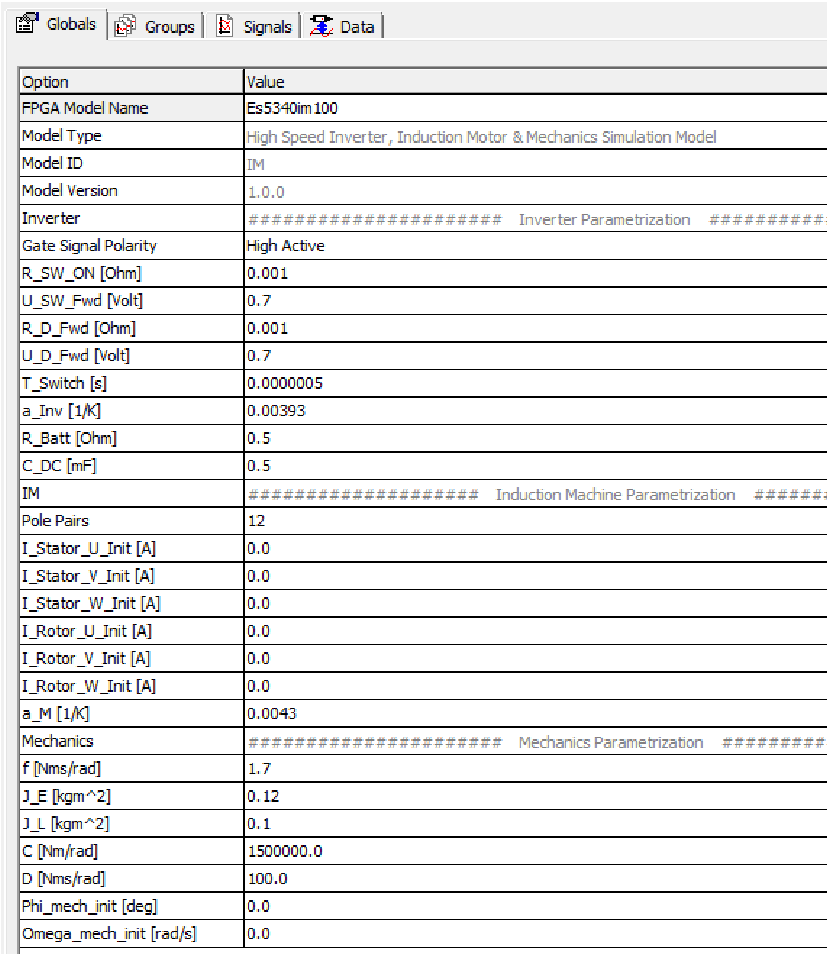


Fig. 22-57 Block Diagram of the ES5340-IM-1.0.0 Model

22.15.1 Globals (ES5340-IM-1.0.0)

The following figure shows the RTIO configuration parameters of the "Globals" tab.



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^2]	0.12
J_L [kgm^2]	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

Fig. 22-58 The "Globals" Tab of the ES5340-IM-1.0.0 RTIO Element

The "Globals" tab is divided into four sections:

- General Information
- Inverter Model
- IM Model
- Mechanics Model

22.15.2 General Information

FPGA Model Name

Name of the model in the RTIO.

Model Type

Display of the model type.

Model ID

Displays the model ID.

Model Version

Displays the model version.

22.15.3 Inverter Model

Gate Signal Polarity

This setting is used to determine the polarity of the inverter gate signals applied to 6 (of 20) digital inputs.

R_SW_ON

Resistance value of a conductive inverter switch at 20 °C.

U_SW_FWD

Voltage drop in forward direction via a conductive inverter switch.

R_D_FWD

Resistance value of the free-wheeling diode of a conductive inverter switch at 20°C.

U_D_FWD

Voltage in forward direction via a diode with a conductive inverter switch.

T_Switch

Switch time of an inverter switch.

a_Inv

Temperature coefficient of the ohmic values of the inverter. This coefficient makes it possible to represent temperature dependency for R_SW_ON and R_D_SW as follows:

```
R_SW_ON(Theta_Inv) =
R_SW_On(20 °C) + (1 + a_Inv * (Theta_Inv - 20 °C) )

R_D_FWD(Theta_Inv) =
R_D_FWD(20 °C) + ( 1 + a_Inv * (Theta_Inv - 20 °C) )
```

R_Batt

Internal resistance of the battery. No temperature dependency is simulated for this variable.

If this value is set to 0, the software sets a minimal value = 0.01. Then the entire RC circuit (comprising R_Batt and C_DC) is disabled regardless of the value of the variable C_DC.

If the RC circuit is disabled, the following is true:

```
I_Batt = I_DC

U_DC = U_Batt0
```

C_DC

Capacity of the DC voltage link.

If this value is set to 0, the software sets a minimal value = 0.01. Then the entire RC circuit (comprising R_Batt and DC) is disabled regardless of the value of the variable R_Batt.

```
I_Batt = I_DC
U_DC = U_Batt0
```

22.15.4 IM Model

Pole Pairs

The number of pole pairs of the induction machine.

T_cm

The peak value of the cogging torque of the induction machine.

I_Stator_X_Init (X = U,V,W)

Initialization of stator current of phase X.

I_Rotor_X_Init (X = U,V,W)

Initialization of rotor current of phase X.

a_M

Temperature coefficient of the ohmic values of the induction machine.

These temperature coefficients are used to implement a temperature dependency for the calculation of the coil resistances R_s1 (stator) and R_s2 (rotor) as follows:

```
R_s1(Theta_M) = R_s1(20 °C) + ( 1 + a_M * (Theta_M - 20 °C))
R_s2(Theta_M) = R_s2(20 °C) + ( 1 + a_M * (Theta_M - 20 °C))
```

R_s1(Theta_M) and R_s2(Theta_M) are the variables that have an effect in the model. These are calculated in accordance with the above formulae depending on the resistance values at 20 °C (R_s1(20 °C) and R_s2(20 °C, R_s1 and R_s2 in the "Signals" tab), on the current machine temperature Theta_M and on the temperature coefficient a_M (see Tab. 22-33 on page 766).

22.15.5 Mechanics Model

The Mechanics model simulates a two-mass system comprising the rotor of the induction machine and the load.

The inertia of these two masses is defined by the variables J_E (induction machine) and J_L (load or drivetrain) – the friction of the overall system is set via the parameter f.

The two masses are connected to each other via a spring-and-damper element (spring constant C, damping D).

f

Friction of the mechanical overall system.

J_E

Mass inertia of the rotor of the induction machine.

J_L

Mass inertia of the load or drivetrain.

C

Torsional stiffness or spring constant of the connecting element between two masses.

D

Damping of the connecting element between two masses.

Phi_mech_init

Initialization value of the mechanical position angle of the electric machine/rotor.

Omega_mech_init

Initialization value of the mechanical angle speed of the electric motor/rotor.

The following table lists the configuration parameters of the model with data type, value range and physical unit.

Parameter	Data Type	Can be edited online	Notes
Gate Signal Polarity	int	Yes	High Active Low Active
R_SW_ON	float	Yes	0.0 .. 0.05 Ω
U_SW_FWD	float	Yes	0.0 .. 3.0 V
R_D_FWD	float	Yes	0.0 .. 0.05 Ω
U_D_FWD	float	Yes	0.0 .. 3.0 V
T_Switch	float	Yes	0.0000002 .. 0.000001 s
a_Inv	float	Yes	0.0 .. 0.01 1/K
R_Batt	float	Yes	0.01 .. 10.0 Ω
C_DC	float	Yes	0.01 .. 10.0 mF
Pole Pairs	float	Yes	1 .. 16
I_Stator_X_Init (X = U,V,W)	float	Yes	-1500.0 .. 1500.0 A
I_Rotor_X_Init (X = U,V,W)	float	Yes	-1500.0 .. 1500.0 A
a_M	float	Yes	0.0 .. 0.01 1/K
f	float	Yes	0.0 .. 10.0 Nms
J_E	float	Yes	0.0 .. 1.0 kgm^2
J_L	float	Yes	0.0 .. 10.0 kgm^2
C	float	Yes	0.0 .. 10000000.0 Nm/rad
D	float	Yes	0.0 .. 1000.0 Nms/rad
Phi_mech_init	float	Yes	0.0 .. 360.0 °
Omega_mech_init	float	Yes	-833.333 .. +833.333 rad/s

Tab. 22-30 ES5340-IM-1.0.0 RTIO Element: Configuration Parameters of the "Globals" Tab

22.15.6 Groups (ES5340-IM-1.0.0)

The ES5340-IM-1.0.0 RTIO element has a signal group called "Outputs" for controlling the engine model and a signal group called "Inputs" for reading the current engine model status information and simulation values.

22.15.7 Signals (ES5340-IM-1.0.0)

In the "Signals" tab, you can specify which value range is mapped to which voltage area for every analog model output value when this value is selected for analog output in the analog output multiplexer.

The physical value "Model Value (min)" is always mapped to the voltage value "DAC Voltage (min)" and the physical value "Model Value (max)" to the voltage value "DAC Voltage (max)".

The Min/Max values for "DAC Voltage" also limit the output voltage – voltage values outside this range cannot be set. If the physical value range of a simulation value is exceeded, the Min/Max values for "DAC Voltage" represent an absolute limit.

Note

If, however, error state 3 (shorted to Uref) is activated in the analog output multiplexer, this restriction has no effect! This means that the voltage defined by the error is applied at the output and could exceed the above-mentioned restriction!

Globals | Groups | Signals | Data

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

Fig. 22-59 The "Signals" Tab of the ES5340-IM-1.0.0 RTIO Element

The following table shows the configuration parameters of the "Signals" tab with data type and value range.

Parameter	Data Type	Can be edited online	Notes
Model Value (min)	float	No	Value range depending on the model variable
Model Value (max)	float	No	Value range depending on the model variable
DAC Voltage (min)	float	No	Value range: -10.0 V .. +10.0 V
DAC Voltage (max)	float	No	Value range: -10.0 V .. +10.0 V

Tab. 22-31 ES5340-IM-1.0.0 RTIO Element: Configuration Parameters of the "Signals" Tab

The following table contains the value ranges of the scaling parameters of analog model variables to analog output voltage ranges:

Parameter	Model Value (min)	Model Value (max)	DAC Voltage (min)	DAC Voltage (max)	Meaning
I_Batt	-1500	1500	-10	10	Current through the simulated battery internal resistance [A]
U_DC	0	1500	-10	10	Voltage at the intermediate circuit capacitor [V]
I_DC	-1500	1500	-10	10	Current into the inverter [A]
I_element_xH*, I_element_xL*	-1500	1500	-10	10	Element current through the high-side or low-side path of phase x [A]
U_switch_xH*, U_switch_xL*	-150	1500	-10	10	Voltage over the high-side or low-side switch of phase x [A]
U_phase_x	-150	1500	-10	10	Phase voltage of phase x with reference to ground potential of intermediate circuit [V]
Error	–	–	-10	10	Latched overflow bit from error register
I_stator_x*	-1500	1500	-10	10	Current in stator coil phase x [A]
I_rotor_x*	-1500	1500	-10	10	Current in rotor coil phase x [A]
I_phase_x*	-1500	1500	-10	10	Phase current in phase x [A]
U_ind_x*	-1500	1500	-10	10	Induced voltage in phase x [U]
T_el	-1000	1000	-10	10	Electrically generated torque [Nm]
Omega_mech	-50000	50000	-10	10	Mechanical angle speed [rad/s]
Phi_mech	0	360	-10	10	Mechanical angle, rotor angle [°]
Omega_el	-50000	50000	-10	10	Electric angle speed [rad/s]
Phi_el	0	360	-10	10	Electric angle [°]
T_mech	-2000	2000	-10	10	Actual effective mechanical torque [Nm]
*x = U,V,W					

Tab. 22-32 ES5340-IM-1.0.0 RTIO Element: Value Ranges for Signals

Note

DAC Voltage (min) must always be less than DAC Voltage (max).

22.15.8 Data (ES5340-IM-1.0.0)

The following figure shows the "Data" tab.

Globals | Groups | Signals | 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]

Fig. 22-60 The "Data" Tab of the ES5340-IM-1.0.0 RTIO Element

The following table shows the signals of the "Data" tab.

Signal Name	Data Type	Direction	Notes
Fail_HS_x*	float	Input	Inverter: Gate error high-side switch phase x* 0 = normal function 1 = switch permanently off 2 = switch permanently on
Fail_LS_x*	float	Input	Inverter: Gate error low-side switch phase x* 0 = normal function 1 = switch permanently off 2 = switch permanently on
U_Batt0	float	Input	DC voltage link input voltage/ battery voltage Value range: 0.0 .. 1500.0 V
Theta_Inv	float	Input	Inverter temperature Value range: -50.0 .. 200.0 °C
R_s1	float	Input	Stator coil resistance at 20 °C Value range: 0.01 .. 1.0 Ω
R_s2	float	Input	Rotor coil resistance at 20 °C Value range: 0.01 .. 1.0 Ω
L_m	float	Input	Effective inductance Value range: 0.00008 .. 0.001 H
L_s1	float	Input	Stator coil inductance Value range: 0.00008 .. 0.001 H
L_s2	float	Input	Rotor coil inductance Value range: 0.00008 .. 0.001 H
Theta_M	float	Input	Temperature in the machine (for calculating the heat dependency of ohmic values) Value range: -50.0 .. 200.0 °C
T_L	float	Input	Load torque Positive value: Torque works in a negative direction of rotation, brake torque Negative value: Torque works in a positive direction of rotation, accelerating torque Value range: -1000.0 .. 1000.0 Nm
U_DC	float	Output	Voltage at the DC voltage link capacitor [V]
I_DC	float	Output	Current into the inverter [A]
I_element_xH*, I_element_xL*	float	Output	Element current through the high-side or low-side path of phase x [A]
U_switch_xH*, U_switch_xL*	float	Output	Voltage over the high-side or low-side switch of phase x [A]
U_Phase_x*	float	Output	Phase voltage of phase x with respect to the ground potential of the DC voltage link [V]
P_Loss_Inv	float	Output	Power loss of the ohmic components of the inverter [W]
I_Batt	float	Output	Current through the simulated battery internal resistance [A]

Signal Name	Data Type	Direction	Notes
Error	float	Output	Bit vector in floating-point representation: Bit 0: Inverter short high-side/low-side phase U Bit 1: Inverter short high-side/low-side phase V Bit 2: Inverter short high-side/low-side phase W Bit 3: Overflow in 52-bit fixed-point value (IM model) Bit 4: Overflow in 52-bit fixed-point value (mechanical model)
I_stator_x*	float	Output	Current in stator coil phase x [A]
I_rotor_x*	float	Output	Current in rotor coil phase x [A]
U_ind_x*	float	Output	Induced voltage in phase x [V]
T_el	float	Output	Electrically generated torque [Nm]
P_Loss_M	float	Output	Power loss of the ohmic components of the machine [W]
Omega_mech	float	Output	Mechanical angle speed [rad/s]
Phi_mech	float	Output	Mechanical angle, rotor angle [°]
Omega_el	float	Output	Electric angle speed [rad/s]
Phi_el	float	Output	Electric angle [°]
T_mech	float	Output	Actual effective mechanical torque [Nm]
*x = U,V,W			

Tab. 22-33 ES5340-IM-1.0.0 RTIO Element: Signals of the "Data" Tab

23 ES5340.2 Internal Combustion Engine Application

This chapter describes the hardware configuration if the ES5340.2 Internal Combustion Engine Application (short: LABCAR Software). For more details on the functions of the board please refer to the ES5340.2 Internal Combustion Engine Application User's Guide.

Structure of the RTIO Hierarchy

The structure of the RTIO hierarchy in the RTC Editor as well as the configuration and real-time data of the individual items are described in more detail below.

To create a new LABCAR Software hierarchy in the LABCAR RTC Editor, add an ES5340-ICE subsystem at RTPC system level. This RTIO element is used as a container for the description of a master and also, if required, a slave board as well as the subsystems for signal measurement and signal generation.

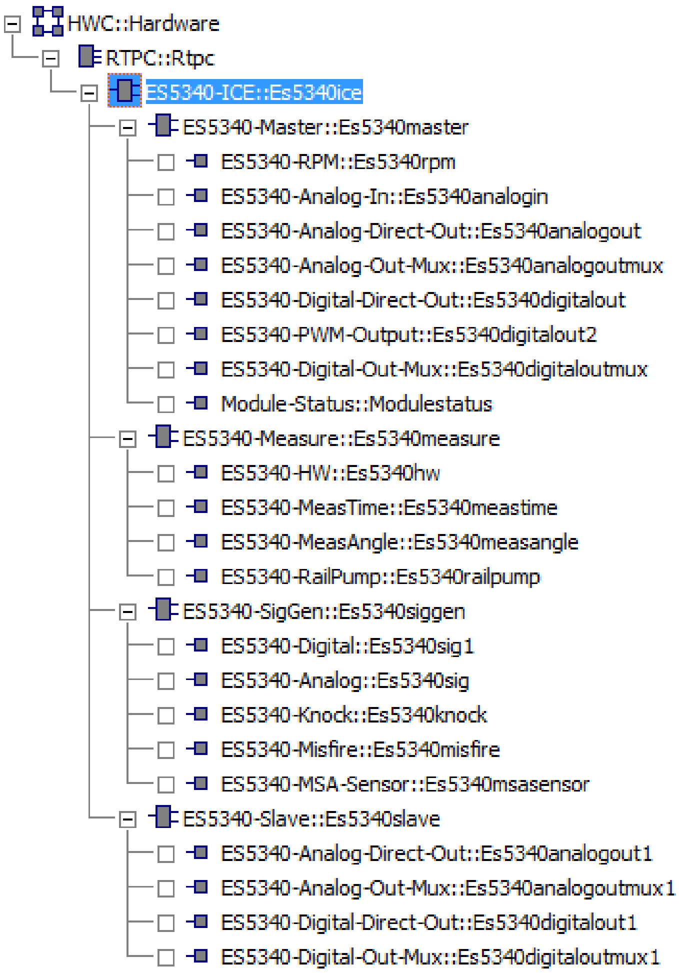


Fig. 23-1 Complete ES5340-ICE RTIO Tree

23.1 ES5340-Master

The ES5340-Master subsystem contains the RTIO elements used with the ES5340-Hybrid and their settings.

23.1.1 Globals (ES5340-Master Subsystem)

This is where you make global settings for the ES5340-Master board.

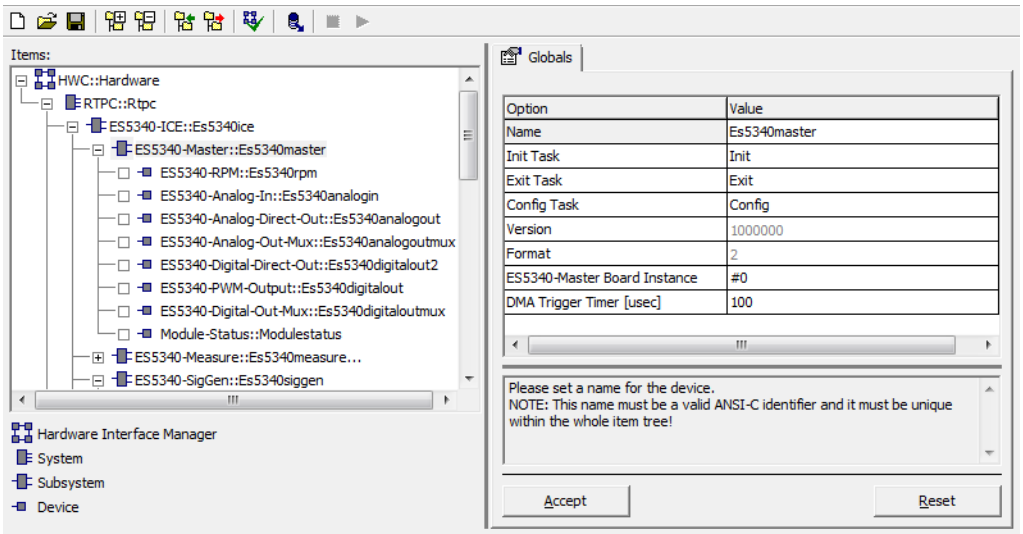


Fig. 23-2 The "Globals" Tab of the ES5340-Master Subsystem ("Show All" Option)

ES5340-Master Board Instance

This option field is used to specify the board instance assigned to this RTIO subsystem. Numbering starts at 0 and the boards of any one type are numbered (as with the VMEbus system) in ascending order from left to right[1].

For this purpose, however, the internal angle clock bus (SYNC interface) must be connected via all boards (see User's Guide) – the board instance cannot be changed during the experiment.

DMA Trigger Time [µsec]

This field is only displayed if "Show All" was selected in the shortcut menu (right mouse button).

This is where you can select the interval for triggering two asynchronous DMA transfers – the data measured is written directly to the PC main memory by the LABCAR Software-Master board.

This concerns the last four time stamps of all 20 digital inputs, the current measure values of the four analog values, the RTIO input values of the RPM unit and status information for knocking.

This data can be changed during simulation by reconfiguring.

- Default value: 100 µs
- Setting range: 5 µs … 1 s

1. Only applies if the boards are connected via the SYNC-Bus lines. Otherwise, numbering takes place either by chance or depends on the hardware.

23.2 ES5340-RPM – RPM Unit

The angle-based synchronization between LABCAR boards takes place via an angle clock signal consisting of three components (Fig. 23-3). The "ac_{SYNC}" signal displays a zero passage of the crankshaft angle in a combustion engine, "ac_{CLK}" contains the actual angle clock, "ac_{DoR}" specifies the direction of rotation of the engine.

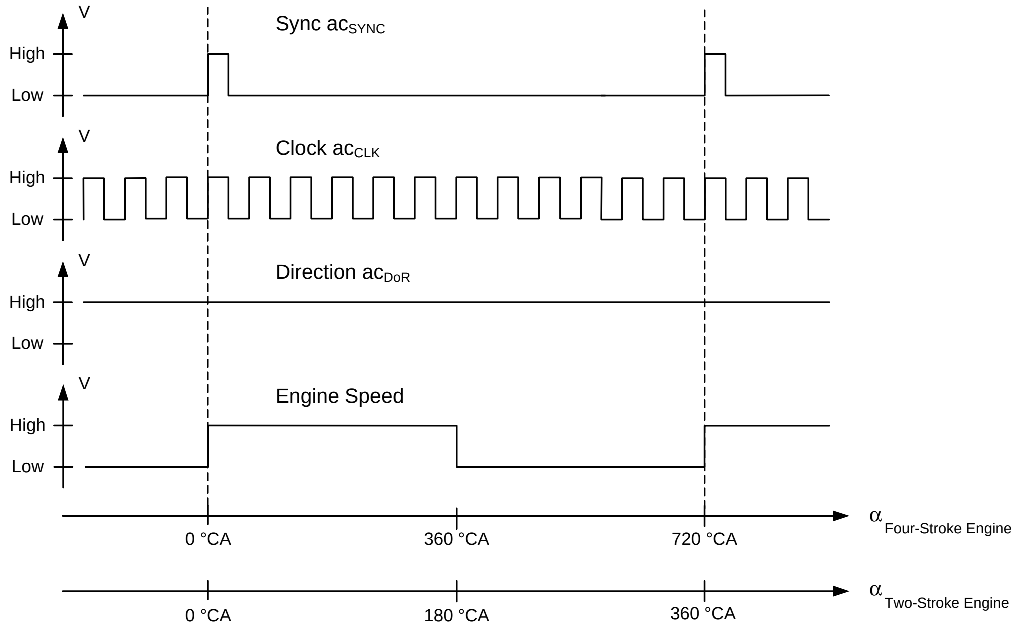


Fig. 23-3 The Three Components of the Angle Clock Signal

The clock signal offers a resolution δ_{α} of 16 bits per 720 °CA of the four-stroke engine or 360 °CA of the two-stroke engine.

The RPM unit of the LABCAR Software can be operated in slave and master mode. In slave mode, the RPM unit is synchronized to an angle clock generated by another board in the system. In master mode, the RPM unit generates the angle clock for other boards in the system. Two buses are available for transferring the angle clock between the boards. There can only be one master at a bus.

23.2.1 TES5340-RPM – RPM Unit

The ES5340-RPM device is used to configure and control the ES5340-RPM unit.

23.2.2 Globals (ES5340-RPM)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

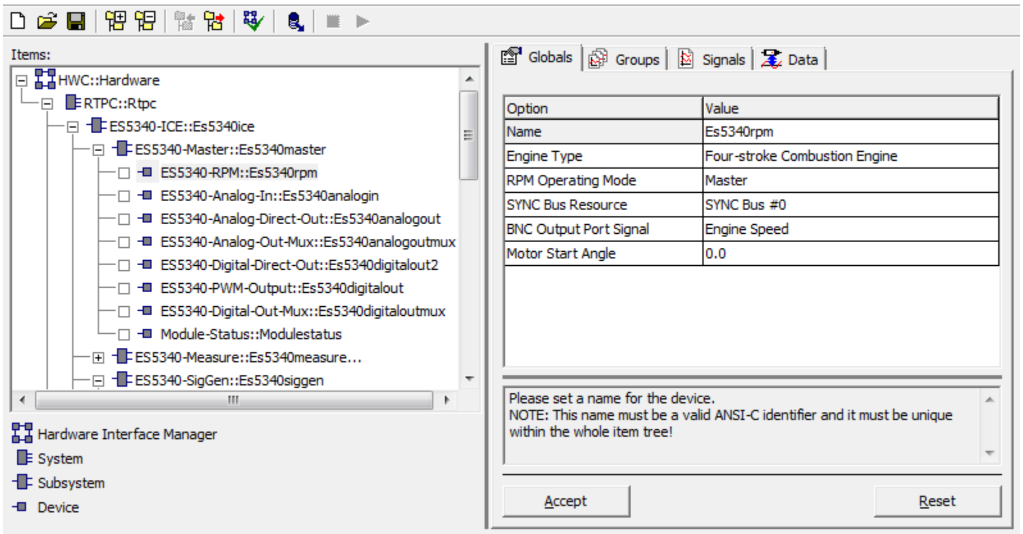


Fig. 23-4 The "Globals" Tab of the ES5340-RPM Device

Engine Type

Selection of the engine type.

RPM Operating Mode

Selection of master or slave mode.

SYNC Bus Resource

The angle clock signal is transported between the boards via synchronization bus lines. This configuration parameter is used to select the bus line via which the angle clock is read in in slave mode or output in master mode.

If the RPM unit is configured to master mode, there is an additional option "None" – in this case an angle clock is generated and used internally on the board, but not output on one of the two synchronization buses.

Note

Please note that there can only be one angle clock master on each synchronization bus!

BNC Output Port Signal

This is where the output signal of the BNC port on the front panel of the LABCAR Software is defined. The signals shown in Fig. 23-3 on page 769 are available for selection.

Motor Start Angle

The angle at which the engine starts after model initialization.

The following table summarizes the properties of the individual configuration parameters.

Parameter	Data Type	Can be edited online	Notes
Engine Type	int	No	0: Four-stroke combustion engine 1: Two-stroke combustion engine
RPM Operating Mode	int	Yes	0: Slave 1: Master
SYNC Bus Resource	int	Yes	0: SYNC Bus #0 1: SYNC Bus #1 255: None (only in master mode)
BNC Output Port Signal	int	Yes	0: Sync 1: Clock 2: Direction 3: Engine Speed
Motor Start Angle	float	Yes	0.0 °CA…360.0 °CA: Two-stroke engine 0.0 °CA…720.0 °CA: Four-stroke engine

Tab. 23-1 The "Globals" Tab of the ES5340-RPM Device

Note

Please note that the angle limits (720 ° or 360 °) vary depending on the "Engine Type" selected here. This can have a corresponding effect on the value range of other parameters!

23.2.3 Groups (ES5340-RPM)

The ES5340-RPM device has an "Outputs" signal group for controlling the RPM unit and an "Inputs" signal group for reading the current status information of the RPM unit.

23.2.4 Data (ES5340-RPM)

The signals made available in the "Data" tab for controlling and monitoring the status of the RPM unit depend on the type of engine selected. Fig. 23-5 on page 772 shows the RTIO signals for the engine type "four-stroke engine" – Tab. 23-2 on page 772 lists the signal properties.

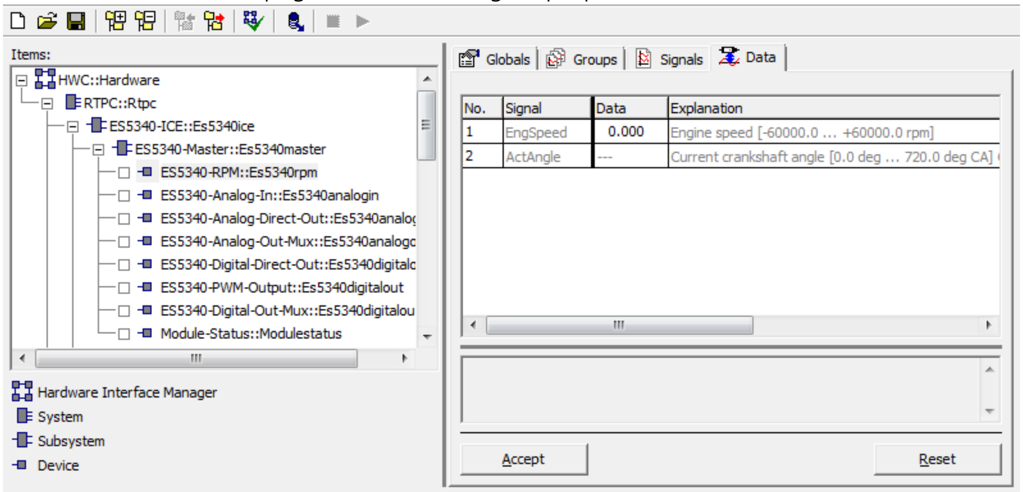


Fig. 23-5 The "Data" Tab of the ES5340-RPM Device (Four-Stroke Engine)

Signal Name	Data Type	Notes
EngSpeed	float	Engine speed in revolutions per minute Value range: Four-stroke engine: -60000.0 rpm … +60000.0 rpm resolution: 0.001 rpm Two-stroke engine: -30000.0 rpm … +30000.0 rpm resolution: 0.0005 rpm An engine rotating backwards can be simulated by entering a negative speed.
ActAngle	float	Current degree crankshaft angle in °CA Value range: 0.0 °CA … 720.0 °CA (or 360 °CA) Resolution: 0.011 °CA (or 0.0055 °CA)

Tab. 23-2 ES5340-RPM Device: RTIO Signals

23.3 ES5340-Analog-In – Analog Inputs

The LABCAR Software has four analog inputs with an electrical strength of ±60 V – each of these inputs is routed to a channel of an AD converter

Two AD converters have a voltage range of 0 to 5 V or 0 to 40 V. Each channel of an AD converter converts at a rate of 500 ksamples per second.

23.3.1 ES5340-Analog-In – Analog Inputs

The "ES5340-Analog-In" device is used to configure the AD converters and acquire the analog voltages applied.

23.3.2 Signals (ES5340-Analog-In)

The average value of acquired AD values can be calculated for each channel of the analog converter in the "Signals" tab.

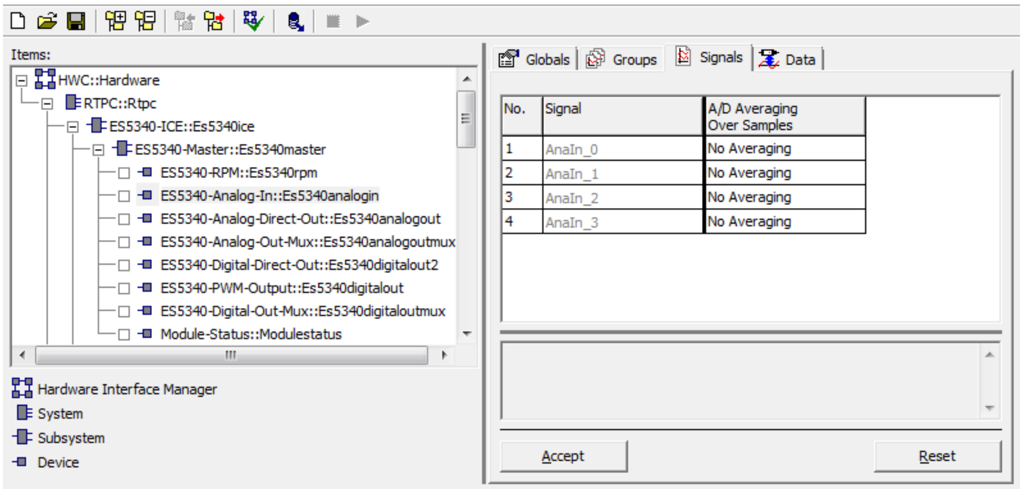


Fig. 23-6 The "Signals" Tab of the ES5340-Analog-In Device

Parameter	Data Type	Can be edited online	Notes
A/D Averaging over Samples	int	Yes	0: No averaging 1: Averaging over 2 samples 2: Averaging over 4 samples 3: Averaging over 8 samples 4: Averaging over 16 samples 5: Averaging over 32 samples 6: Averaging over 64 samples 7: Averaging over 128 samples

Tab. 23-3 ES5340-Analog-In Device: Configuration Parameters of the "Signals" Tab

23.3.3 Data (ES5340-Analog-In)

The "Data" tab shows the current voltage values in Volt.

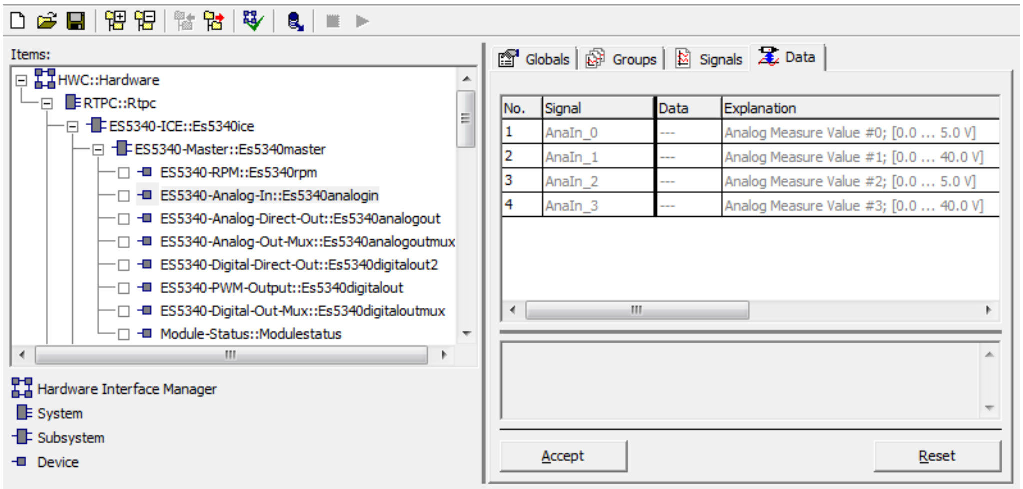


Fig. 23-7 The "Data" Tab of the ES5340-Analog-In Device

Signal Name	Data Type	Notes
AnaIn_0	float	Analog input 0 (measures in the range 0.0…5.0 V)
AnaIn_1	float	Analog input 1 (measures in the range 0.0…40.0 V)
AnaIn_2	float	Analog input 2 (measures in the range 0.0…5.0 V)
AnaIn_3	float	Analog input 3 (measures in the range 0.0…40.0 V)

Tab. 23-4 ES5340-Analog-In Device: Signals of the "Data" Tab

23.4 ES5340-Digital-Out – Digital Outputs

The RTIO element ES5340-Digital-Out can be used in two different operating modes.

Mode	Description
Digital Direct Out	Digital output (see "ES5340-Digital-Direct-Out" on page 775)
PWM Output	Output of PWM signals (see "ES5340-PWM-Output" on page 777)

Tab. 23-5 Operating Modes of the ES5340-Digital-Out RTIO Element

The configuration of the RTIO element is different for every output mode – the appearance of the ES5340-Digital-Out RTIO element also changes accordingly.

The LABCAR Software uses a two-step procedure when outputting digital values:

1. The output values are configured via the ES5340-Digital-Out RTIO element. You can configure more signals here than there are output channels on the LABCAR Software (max. 8 channels in "Digital-Direct-Out" mode and 8 channels for the other digital output modes)
2. The values actually output are selected via the ES5340-Digital-Out-Mux RTIO element (see "ES5340-Digital-Out-Mux – Digital Output Multiplexer" on page 779).

23.4.1 ES5340-Digital-Direct-Out

This section describes the configuration of the ES5340-Digital-Out RTIO element with the output mode "Digital Direct Out". One to eight digital outputs can be driven directly in this operating mode.

23.4.2 Globals (ES5340-Digital-Direct-Out)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

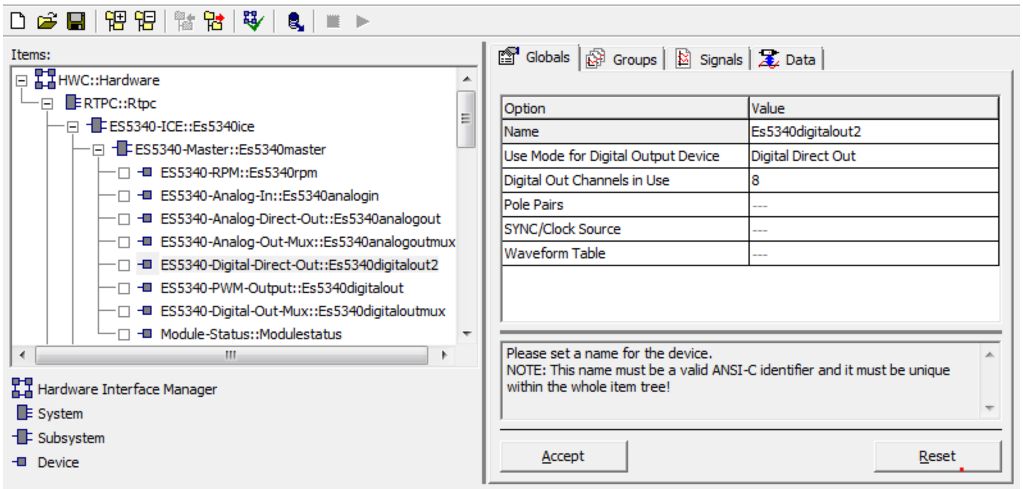


Fig. 23-8 The "Globals" Tab of the ES5340-Digital-Direct-Out RTIO Element

Use Mode for Digital Output Device

Selection of the output mode.

Digital Out Channels in Use

Number of the configurable digital output channels (max. 8).

23.4.3 Groups (ES5340-Digital-Direct-Out)

The ES5340 Digital-Direct-Out-RTIO element has a signal group called "Outputs", for controlling the relevant signal generator.

23.4.4 Signals (ES5340-Digital-Direct-Out)

No further settings are necessary here.

23.4.5 Data (ES5340-Digital-Direct-Out)

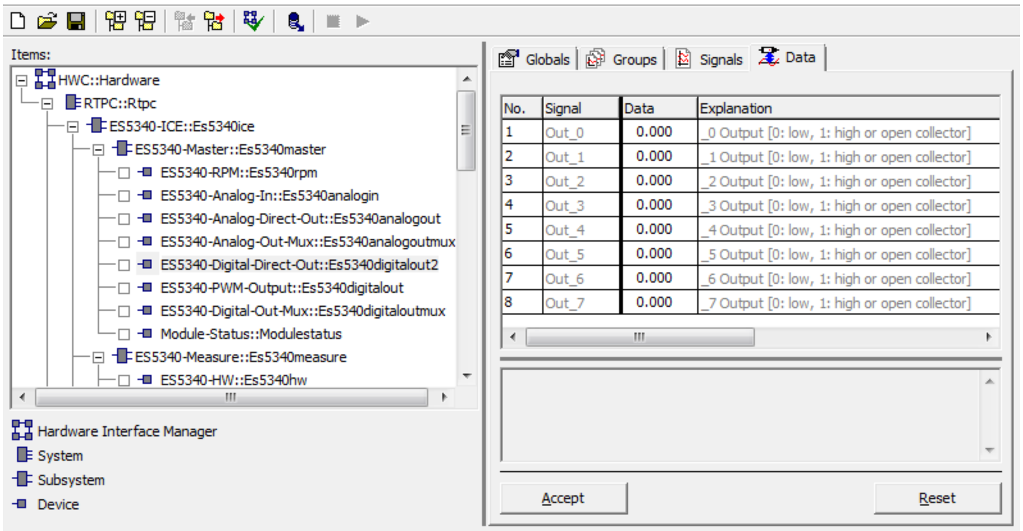


Fig. 23-9 The "Data" Tab of the ES5340-Digital-Direct-Out RTIO Element

Signal Name (n = 0...7)	Data Type*	Notes
Out_*n*	uint8	Digital output value 0: low; 1: high or open collector
*Data type which the RTIO driver uses internally for the parameter		

Tab. 23-6 ES5340-Digital-Direct-Out: Signals of the "Data" Tab

23.4.6 ES5340-PWM-Output

In the operating mode "PWM Output", up to eight digital outputs can be configured for PWM output ("active-high" only). Frequency and duty cycle are specified separately for each PWM channel. The new data is accepted at the end of each relevant PWM cycle.

Note

If the frequency set is less than 1 Hz , new values are accepted immediately!

23.4.7 Globals (ES5340-PWM-Output)

The following figure shows the configuration parameters of the "Globals" tab.

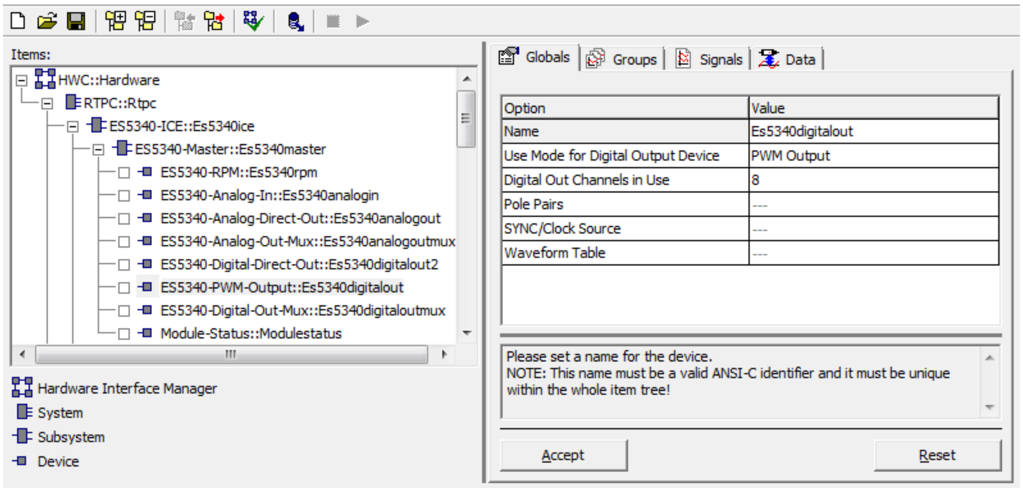


Fig. 23-10 The "Globals" Tab of the ES5340-PWM-Output Element

Use Mode for Digital Output Device

This is where the output mode is selected.

Digital Out Channels in Use

Number of configurable digital output channels (max. 8).

23.4.8 Groups (ES5340-PWM-Output)

The ES5340-PWM-Output has a signal group called "Outputs" for controlling the relevant signal generator(s).

23.4.9 Signals (ES5340-PWM-Output)

No further settings are necessary here.

23.4.10 Data (ES5340-PWM-Output)

The following figure shows the "Data" tab.

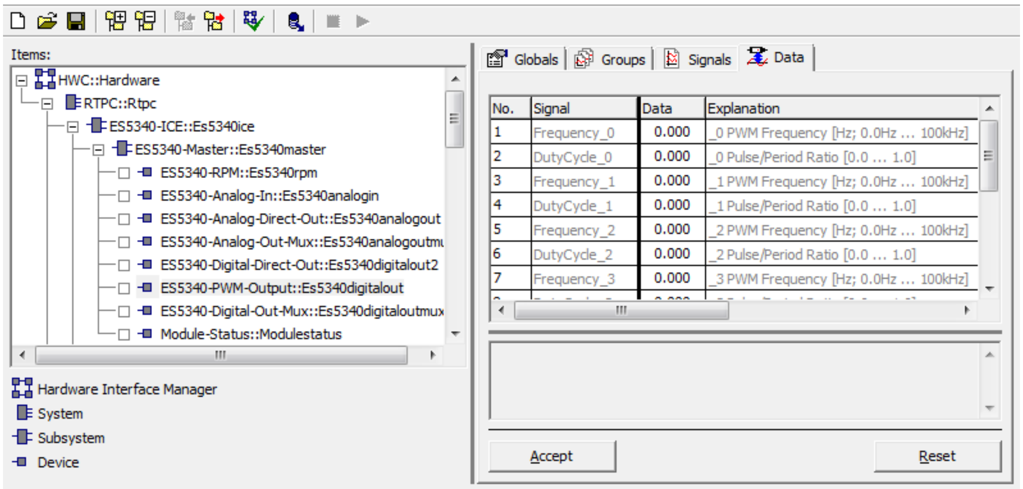


Fig. 23-11 The "Data" Tab of the ES5340-PWM-Output-RTIO Element

Frequency_n

The frequency of the signal from PWM generator n (n = 0...7).

DutyCycle_n

The duty cycle of the PWM signal at output n (n = 0...7).

Signal Name (n = 0...7)	Data Type*	Notes
Frequency_*n*	uint32	Frequency of the PWM signal (0 Hz…100 kHz)
DutyCycle_*n*	uint32	Duty cycle (0.0…1.0)

*Data type which the RTIO driver uses internally for the parameter

Tab. 23-7 ES5340-PWM-Output: Signals of the "Data" Tab

23.5 ES5340-Digital-Out-Mux – Digital Output Multiplexer

In the RTIO element "ES5340-Digital-Out-Mux", signals (configured in the RTIO elements "ES5340-Digital-Out", "ES5340-MSA-Sensor" and "ES5340-Analog" of the "ES5340-Siggen" subsystems") are assigned to the digital signal outputs that are actually present on the LABCAR Software.

23.5.1 Signals (ES5340-Digital-Out-Mux)

The digital outputs are configured in the "Signals" tab.

The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

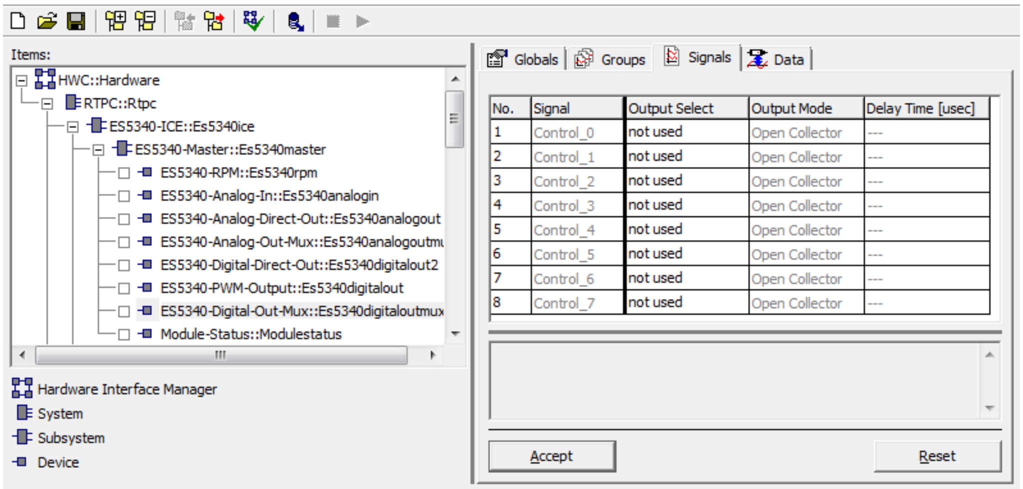


Fig. 23-12 The "Signals" Tab of the ES5340-Digital-Out-Mux RTIO Element

Output Select

The option field "Output Select" is used to select the internal signal to be applied to the particular output.

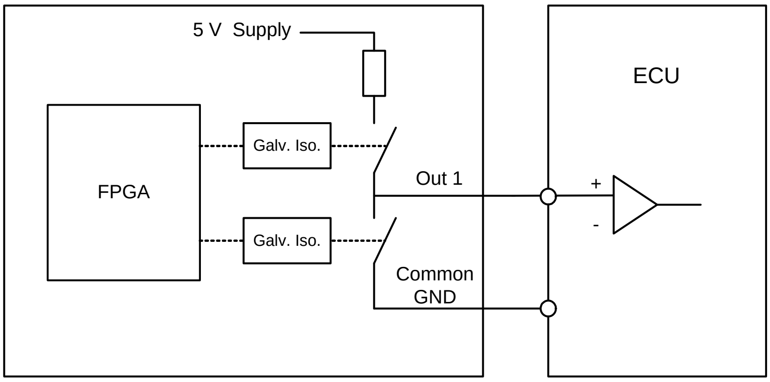
A list of all internal signal sources is available for selection:

- Output values of all Digital-Out RTIO elements (ES5340-Digital-Direct-Out and ES5340-PWM-Output)
- Digital signals of the arbitrary signal generators (see "ES5340-SigGen Subsystem" on page 820)
- MSA Sensor signal

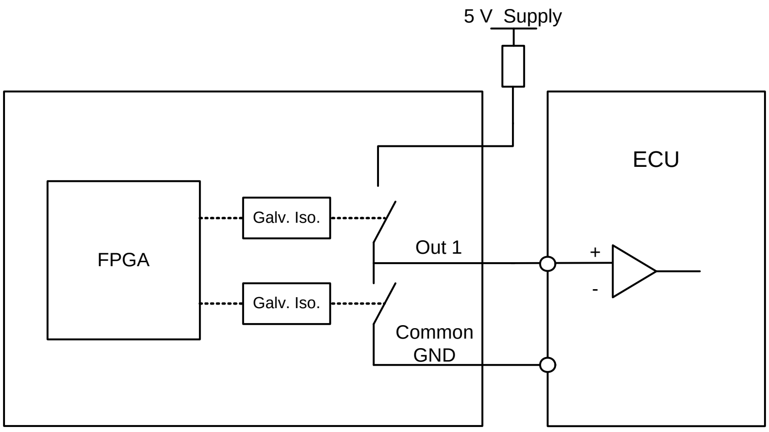
Output Mode

There are two different output modes to choose from:

- Pull-Up to +5V
  An internal pull-up resistor is switched to 5 V



- Open Collector
  An external pull-up resistor is expected here.



23.5.2 Data (ES5340-Digital-Out-Mux)

In this signal group, eight RTIO signals that control the closing of the output relay of each channel are processed. It is still possible to apply the outputs to GND or +5 V permanently for error simulation.

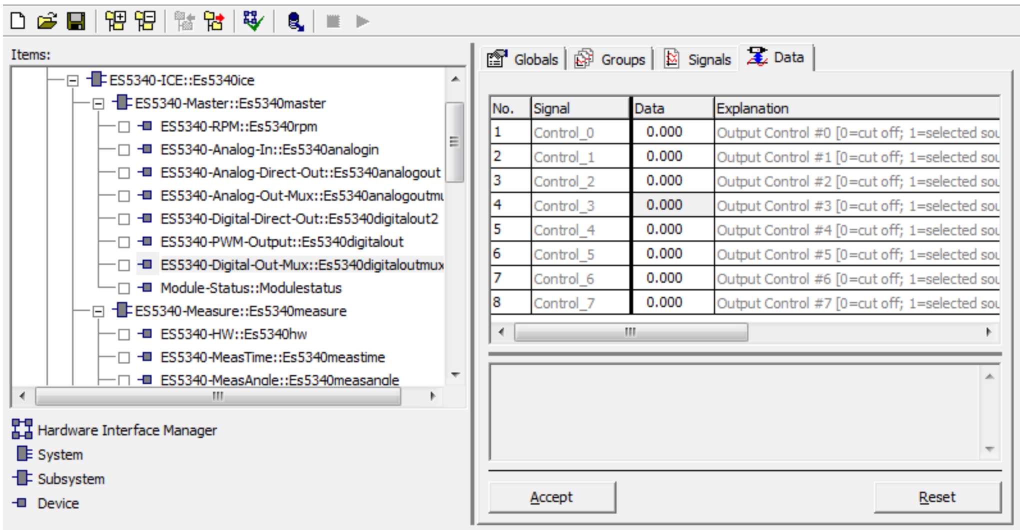


Fig. 23-13 The "Data" Tab of the ES5340-Digital-Out-Mux RTIO Element

Signal Name (n = 0...7)	Data Type*	Notes
Control_*n*	uint8	0: Output disconnected 1: Output connected 2: Shorted to GND 3: Short to +5 V
*Data type which the RTIO driver uses internally for the parameter		

Tab. 23-8 ES5340-Digital-Out-Mux: Signals of the "Data" Tab

23.6 ES5340-Analog-Out – Analog Outputs

The ES5340-Analog-Out RTIO element can be used in the operating mode "Analog Direct Out" (output of a predefined voltage value).

The LABCAR Software uses a two-step procedure when outputting analog values:

1. The output values are configured via the ES5340-Analog-Out RTIO element.
2. The values actually output are selected via the ES5340-Analog-Out-Mux RTIO element (see "ES5340-Analog-Out-Mux – Analog Output Multiplexer" on page 784).

23.6.1 ES5340-Analog-Direct-Out

In the operating mode "Analog Direct Out", you can configure one to eight analog outputs for the immediate output of an analog value.

23.6.2 Globals (ES5340-Analog-Direct-Out)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

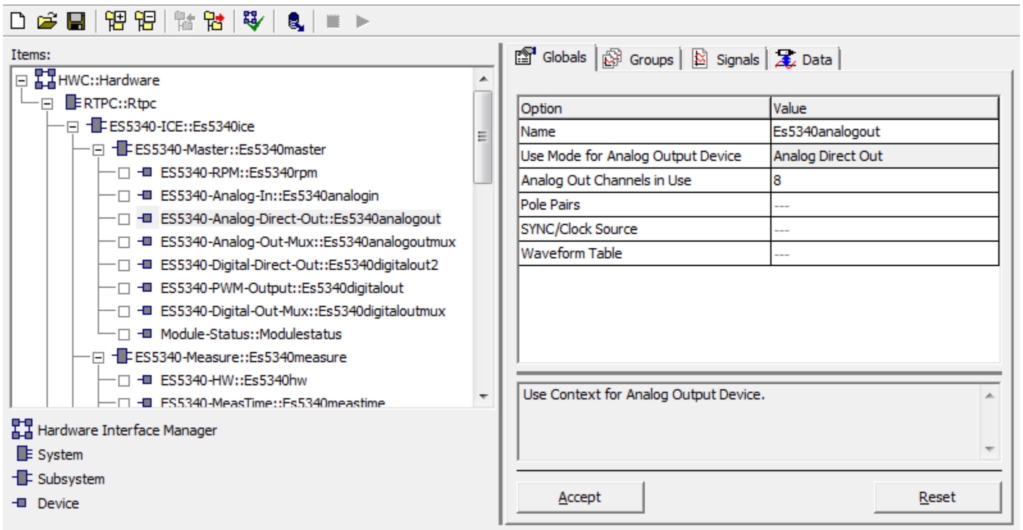


Fig. 23-14 The "Globals" Tab of the ES5340-Analog-Direct-Out RTIO Element

Use Mode for Analog Output Device

Selection of the output mode.

Analog Out Channels in Use

Number of configurable analog output channels (max. 8).

23.6.3 Groups (ES5340-Analog-Direct-Out)

The ES5340 Analog-Direct-Out RTIO element has a signal group called "Outputs" for controlling the output of analog signals.

23.6.4 Signals (ES5340-Analog-Direct-Out)

No further settings are necessary here.

23.6.5 Data (ES5340-Analog-Direct-Out)

The following figure shows the "Data" tab.

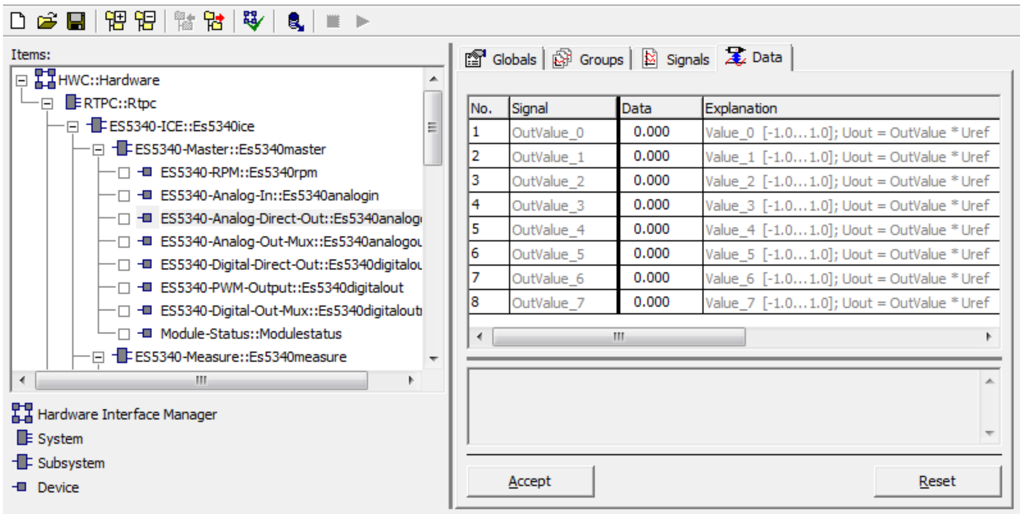


Fig. 23-15 The "Data" Tab of the ES5340-Analog-Direct-Out RTIO Element

OutValue_x

Relative amplitude of the output – is multiplied by the reference voltage specified in the analog multiplexer.

Signal Name (n = 0...7)	Data Type	Notes
OutValue_*n*	sint16	Analog output value in the range -1.0 … 1.0; Uout = OutValue * Uref

Tab. 23-9 ES5340-Analog-Direct-Out: Signals of the "Data" Tab

23.7 ES5340-Analog-Out-Mux – Analog Output Multiplexer

In the RTIO element "ES5340-Analog-Out-Mux", signals (configured in the RTIO element "ES5340-Analog-Out" and in the sources of the "ES5340SigGen" subsystem – see below) are assigned to the analog signal outputs that actually exist in the LABCAR Software.

23.7.1 Signals (ES5340-Analog-Out-Mux)

The eight (ES5340-Master) or six (ES5340-Slave) analog outputs are configured in the "Signals" tab. The properties of the analog output as well as the source of the output value can be determined here.

The following figure shows the RTIO parameters of the "Signals" tab – the properties of the individual parameters are listed in Tab. 23-10 on page 785. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

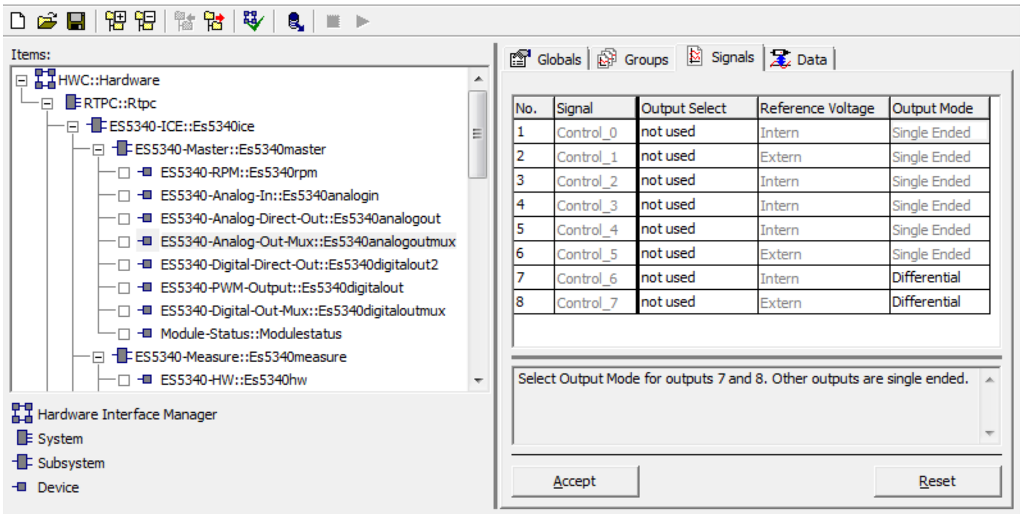


Fig. 23-16 The "Signals" Tab of the ES5340-Analog-Out-Mux RTIO Element

Output Select

The "Output Select" option field is used to select the internal signal that is to be applied at the relevant output. A list of all internal signal sources is available for selection:

- Signals from ES5340-Analog-Direct-Out
- Signals from ES5340-Analog (arbitrary signal generators)
- Signals from ES5340-Knock (knock generators)

Reference Voltage

This option field is used to select the reference voltage for the D/A converter of the relevant output channel. Possible values are "internal" (for an internal reference voltage of 10 V) and "external" (for an externally pending reference voltage of 12 V).

Output Mode

This parameter can only be modified for channels 6 and 7. You can choose between a single ended and a differential output signal. The Output Mode for channels 6 and 7 is always identical.

23.7.2 Data (ES5340-Analog-Out-Mux)

In this signal group, eight RTIO signals that control the closing of the output relay of each channel are processed. It is still possible to apply the outputs to GND or Uref permanently for error simulation.

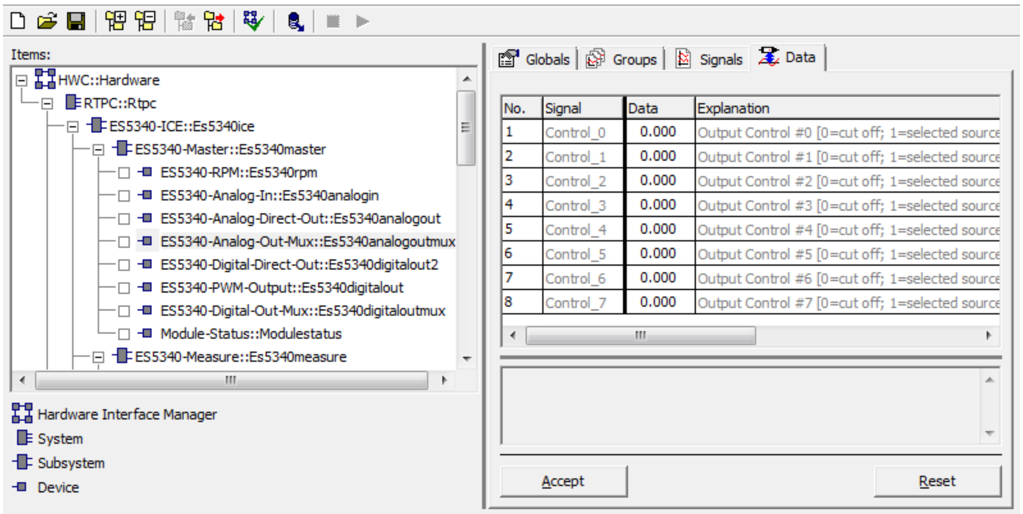


Fig. 23-17 The "Data" Tab of the ES5340-Analog-Out-Mux RTIO Element

Signal Name (n = 0...7)	Data Type*	Notes
Control_*n*	uint8	0: Output disconnected 1: Output connected 2: Shorted to GND 3: Short to +5 V
*Data type which the RTIO driver uses internally for the parameter		

Tab. 23-10 ES5340-Analog-Out-Mux: Signals of the "Data" Tab

Note

If the Output Mode is set to "Differential", a direct voltage cannot be output.

If at the same time "Reference Voltage" is set to "Internal", 0 V is output in case 3 (shorted to Uref). If, however, "Reference Voltage" is set to "External", the AC voltage at the external reference input ("Excitation P/N") is output with full amplitude in case 3 (shorted to Uref).

23.8 ES5340-Slave

The ES5340-Slave subsystem is used to configure an ES5340-Slave board.

23.9 ES5340-Digital-Out (Slave)

With the slave board, digital signals can only be output in the operating mode "Digital Direct Out" with a max. of six channels.

For a description of the ES5340-Digital-Direct-Out RTIO element, refer to the section "ES5340-Digital-Direct-Out" on page 775.

23.10 ES5340-Digital-Out-Mux (Slave)

In the RTIO element "ES5340-Digital-Out-Mux", the signals to be issued at the six digital signal outputs of the slave board are selected.

The following signals can be selected:

- The signals of the type "Digital Direct Out" of the slave board
- The digital output values of a model of the master board

For more information on configuring signal outputs, refer to the section "Signals (ES5340-Digital-Out-Mux)" on page 779.

23.11 ES5340-Analog-Out (Slave)

With the slave board, analog signals can only be output in the operating mode "Analog Direct Out" with a max. of six channels.

For a description of the ES5340-Analog-Direct-Out RTIO element, refer to the section "ES5340-Analog-Direct-Out" on page 782.

23.12 ES5340-Analog-Out-Mux (Slave)

In the RTIO element "ES5340-Analog-Out-Mux", the signals to be issued at the six analog signal outputs of the slave board are selected.

Signals of the type "Analog Direct Out" of the slave board can be selected.

For more information on configuring signal outputs, refer to the section "Signals (ES5340-Analog-Out-Mux)" on page 784.

23.13 ES5340-Measure Subsystem

Terms

Active signal edge

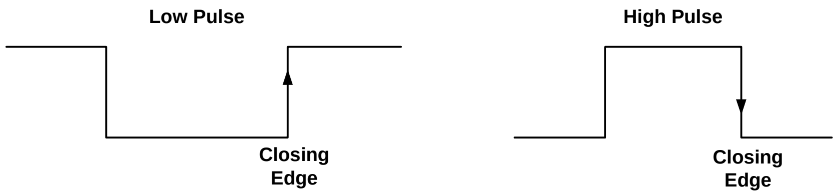
The active signal edge is the rising edge in measurements which are executed at rising edges or the falling edge in measurements which are executed at falling edges.

Active signal

A signal is referred to as being active if its level is low during lowtime measurements or high during hightime measurements.

Closing pulse edge

The rising edge is the closing edge of a low pulse. The falling edge is the closing edge of a high pulse.



Inactive signal

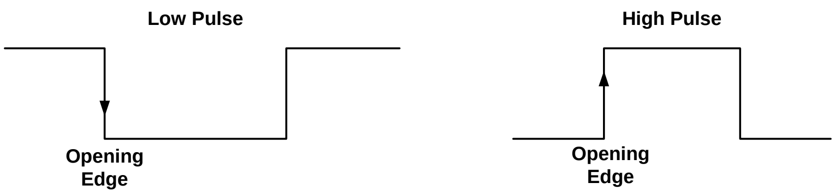
A signal is referred to as being inactive if its level is high during lowtime measurements or low during hightime measurements.

LWL

Stands for Lower Window Limit and refers to the lower limit of the angle window in angle-synchronous measurements.

Opening pulse edge

The falling edge is the opening edge of a low pulse. The rising edge is the opening edge of a high pulse.



UWL

Stands for Upper Window Limit and refers to the upper limit of the angle window in angle-synchronous measurements.

23.13.1 The Basic Principle of Measure Value Calculation

Comparator Levels

For further data acquisition, it is necessary to extract binary level information from the measured input voltages. Every input voltage is routed to a comparator level with hysteresis behavior for this purpose.

The user can configure the threshold voltages of the comparators in different ways:

- Absolute levels
- Lower threshold = 1/3 URefX (model values) (X = A...E)
  Upper threshold = 2/3 URefX (model values) (X = A...E)
- Lower threshold = 1/3 Analog-In *n*
  Upper threshold = 2/3 Analog-In *n*

Pulse-Width and Angle Measurements

The level information of every channel is constantly being checked for changes, i.e. signal edges are detected. When an edge occurs, the current time and current crankshaft angle are stored in an FPGA-internal memory. The processor firmware of the LABCAR Software can now read the acquired values from the memory and thus generate information on when and at which angle which input signals have changed.

All measure values which are of interest to the user can then be generated from this information, values such as injection times, ignition times and duty cycles.

This basic concept of hardware-technical edge detection with subsequent software editing of the measure values has the advantage that the firmware and RTIO driver simply have to be adjusted to implement further measurement procedures (in addition to the ones already implemented).

Asynchronous Measurements

With asynchronous measurements, the relevant measure value (e.g. frequency, duty cycle or hightime) is calculated on the basis of the most recent edge entries available in the memory.

Time-synchronous timeout monitoring can be activated in these measurements. If no edge is detected on the relevant hardware channel within a period defined by the user, a timeout is detected.

Angle-Synchronous Measurements

Angle windows which are specified by a lower angle window limit (LWL) in CA° and an upper angle window limit (UWL) in °CA are characteristic for angle-synchronous measurements.

The user can define up to three angle windows per hardware channel which can overlap but whose size must not exceed 720 °CA. The user defines measurements of the following kind:

- hightime of the n-th pulse in the angle window
- angle of the n-th falling edge in the angle window
- additive lowtime in the angle window

The calculation of the measure values takes place whenever the crankshaft angle has exceeded the upper angle window limit. Timeout monitoring can also be activated for angle-synchronous measurements. If, at the upper window limit, it is discovered that the measure value cannot be calculated, e.g. if the hightime of the fourth pulse is to be measured in the angle window but only three pulses occurred, a timeout is detected.

Timeouts

A timeout is indicated by the setting of a timeout flag (with activated timeout monitoring). At the same time, the valid measure value is defined by timeout monitoring.

There are two possibilities: Either a measure value specified by the user is transferred to the simulation model or a value defined by the relevant measure method. The user decides which of the two values is transferred whereby the second procedure (a value defined by the measure method) is only available with a few measurement procedures.

23.13.2 Globals (ES5340-Measure Subsystem)

The following figure shows the RTIO parameters of the "Globals" tab.

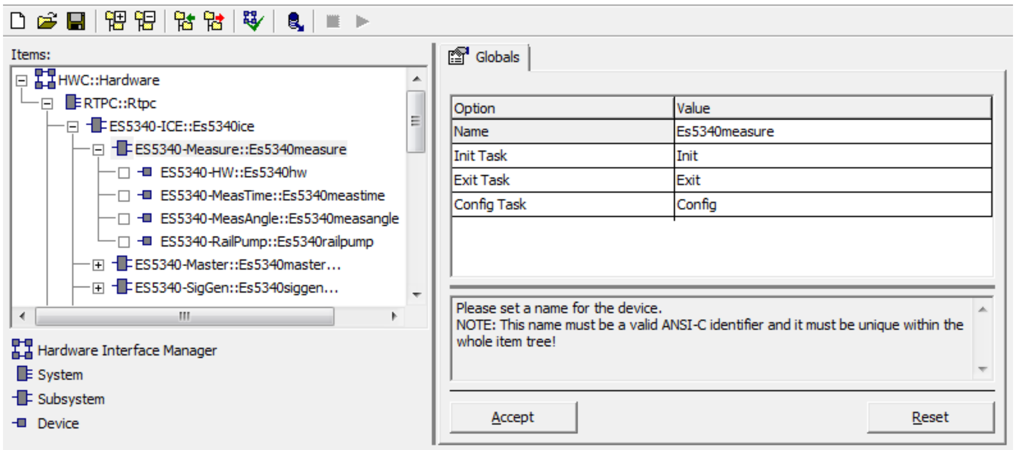


Fig. 23-18 The "Globals" Tab of the ES5340-Measure Subsystem

23.14 ES5340-HW Device

The ES5340-HW device is used to configure and address the LABCAR Software.

23.14.1 Groups (ES5340-HW Device)

The ES5340-HW device has two signal groups.

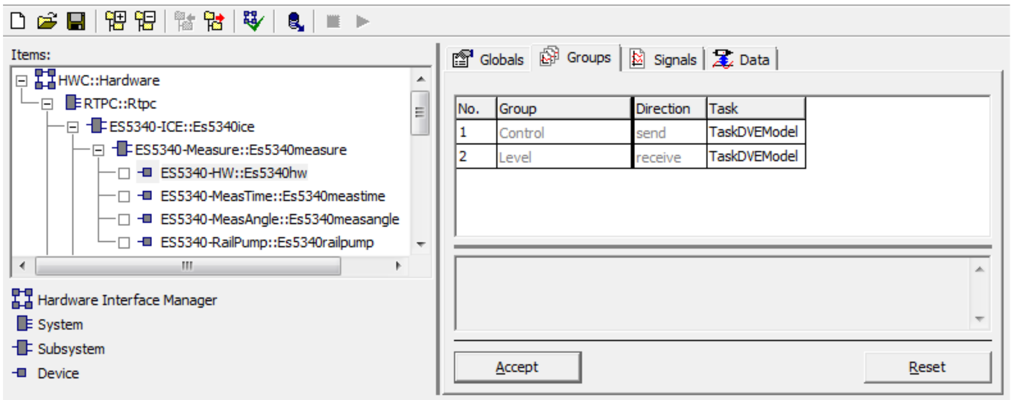


Fig. 23-19 The "Groups" Tab of the ES5340-HW Device

The "Control" signal group is transferred from the experimental target to the LABCAR Software. This signal group is used to enable/disable measure value calculation on the individual hardware channels of the board.

The "Level" signal group is transferred in the opposite direction, from the LABCAR Software to the experimental target. This signal group contains the current level information of all hardware channels.

Operating system tasks are to be assigned to the signal groups. Usually a task with periodic activation and a relatively large activation period (e.g. 100 ms) is selected for the "Control" signal group as the enabling and disabling of the hardware channels is not usually a highly dynamic procedure. If the hardware channels are only to be enabled and disabled when the model is started or stopped, it is sufficient to assign the "Control" signal group to the Init task and the Exit task of the model.

If level information is also evaluated in the simulation model, a task with periodic activation is assigned to the "Level" signal group. The activation period depends on the dynamic behavior or the period duration of the signals to be acquired.

RTIO Signals of the "Control" Signal Group

The "Control" signal group consists of 20 RTIO signals with which measure value calculation on the 20 hardware channels can be enabled or disabled.

RTIO Signals of the "Level" Signal Group

The "Level" signal group consists of one RTIO signal, "LvlBitField_0", which is to be interpreted as a bit field. The bit field is a 20-bit field; the level information of hardware channels 0 to 19 is coded in it.

23.14.2 Signals (ES5340-HW Device)

The 20 hardware channels of an LABCAR Software are configured in the "Signals" tab. Fig. 23-20 shows the RTIO parameters of the "Signals" tab.

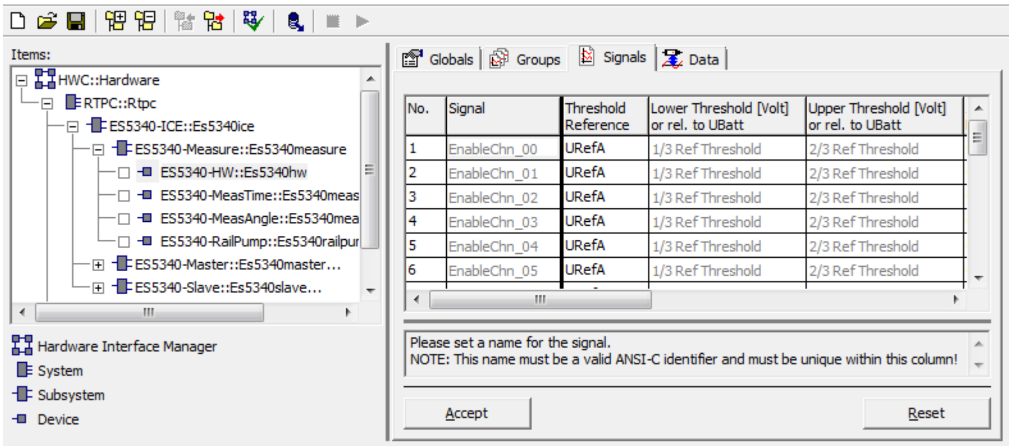


Fig. 23-20 The "Signals" Tab of the ES5340-HW Device

Tab. 23-11 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
EnableChn_00 ... EnableChn_19	bool	Enabling/disabling measure value calculation on hardware channels 0 to 19 0: Disable 1: Enable
UrefA...URefE	float	Definition of the five voltages for the comparator stage. Value range: 0...10 V
LvlBitField_0	uint32	Bit field with level information of hardware channels 0 to 19: Channel 0: bit 0, LSB (Least Significant Bit) Channel 19: bit 19 Bit value 0: low level Bit value 1: high level
* Data type which the RTIO driver uses internally for the parameter		

Tab. 23-11 ES5340-HW Device: RTIO Signals

The properties of the individual parameters are described below. All parameters can be edited during runtime of the model on the experimental target.

Threshold Reference

The "Threshold Reference" option field is used to define the reference voltages of the comparators. The threshold voltages of the comparators can be configured in three different ways:

- Absolute levels

  The comparator thresholds are defined explicitly in volts. The lower comparator threshold is defined in the numeric input field "Lower Threshold [Volt] or rel. to UBatt", the upper comparator threshold is defined in the numeric input field "Upper Threshold [Volt] or rel. to UBatt".
- URef*X* (X = A...E)

  In this case the lower threshold is 1/3 of battery voltage URefX and the upper threshold is 2/3 of battery voltage URefX.
- Analog-In *n* (n = 0..3)

  In this case the lower threshold is 1/3 of Analog_In *n* and the upper threshold is 2/3 of Analog_In *n*.

Lower Threshold [Volt] or rel. to UBatt

Upper Threshold [Volt] or rel. to UBatt

These two numeric input fields can only be edited if "Absolute Levels" has been selected in the "Threshold Reference" field. The input comparators of a hardware channel exhibit hysteresis behavior. The input fields define the upper and lower threshold of the comparator hysteresis in volts.

Angle Window #x Lower Limit [°CA] (x = 0, 1, 2)

Angle Window #x Upper Limit [°CA] (x = 0, 1, 2)

To execute angle-synchronous measurements, three angle windows can be defined on every hardware channel. "Angle Window #x Lower Limit [°CA]" defines each lower angle window limit in °CA, "Angle Window #x Upper Limit [°CA]" defines each upper angle window limit in °CA.

The limits can be set in the range [-720.0 °CA, 720 °CA[(four-stroke engine) and [-360.0 °CA, 360 °CA[(two-stroke engine) with the limitations that an angle window must not be greater than 720 °CA (and 360 °CA repectively) and the lower angle window limit must be smaller than the upper angle window limit.

23.15 ES5340-MeasTime Device

23.15.1 Non-Synchronized Time/Frequency Measurements

This RTIO device is used to measure time, frequency and duty cycle at periodic signals or to acquire the binary value of an input. A maximum of ten measurement channels can be configured in one RTIO device and a maximum of four of these RTIO devices can be integrated into an ES5340-Digital-In subsystem. In total, there are a maximum of 40 software channels available.

The individual measurements are independent of each other and are also basically independent of any other use of the inputs (snchronous signal measurement).

The Basic Principle of Measure Value Acquisition and Calculation

Measure values are acquired on the LABCAR Software – the 20 input signals are digitized in accordance with the thresholds specified (in the ES5340-HW device).

A data word is generated for any change of edge in a hardware channel. This consists of a 31-bit time stamp of a local counter and one bit for edge information. The time stamp has a resolution of 8 ns (125 MHz clocking) so a time of max. 17.28 s can be shown. An edge bit of "1" means a positive edge, i.e. the level is then "1".

The measure values are calculated in the driver of the real-time PC. For this purpose, the data of the last four edge changes are transferred to the RAM of the real-time PC for every hardware channel using cyclical DMA transfer. With a sufficiently low DMA transfer cycle time (default value: 100 µs), all the information for calculating the measurement procedures with low latency is available in the driver.

If there have not been four edge changes since the beginning of data acquisition on a hardware channel, the same data word is generated for all four edges at the start of data acquisition. This consists of the current time stamp and edge information indicating the current level.

These simple coding rules make it possible to distinguish valid data from invalid data:

- The current level can also be determined without an edge change.
- Invalid edge data has an edge bit that has not changed since the previous data.

The measurement procedures for returning the level thus always return valid data. In all other measurement procedures, there have to be sufficient edges available which depends on the procedure and on the current state.

If no measure value can be calculated, -8888.0 is returned as value (providing there is no timeout or monitoring).

23.15.2 The Measurement Procedures

Below you will find a description of the possible procedures for measuring digital signals.

Note

Generally speaking, the measured times between two flank events must not exceed the mapping area of the 31-bit counter (17.28 s). If this happens the counter overflows resulting in incorrect measurements.

Frequency and Cycle Time Measurement

The measurement procedures

- Cycle Time --/-- [µs]
- Cycle Time --\-- [µs]
- Frequency --/-- [Hz]
- Frequency --\-- [Hz]

provide the cycle time or frequency of the signal at the relevant input measured between two rising edges (--/--) or two falling edges (--\--).

High/Low Time Measurement

The measurement procedures

- High Time [µs]
- Low Time [µs]

provide the time between the last rising and the following falling edge (High Time) or the last falling and rising edge (Low Time).

Duty Cycle Measurements

The measurement procedures

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

provide the ratio of the low or high time, with reference to the cycle time of rising to rising edge (--/--) or falling to falling edge (--\--).

Level Measurement

The measurement procedures

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

provide the level of a PWM input in the form of active/inactive information. "0" means the signal is inactive, "1" means the signal is active.

What Happens if a Timeout Occurs

Monitoring of timeouts can be set for every measurement channel. When monitoring is activated, the default return value and the monitoring interval can be changed. A timeout is reached when the difference between the current time of a DMA transfer and the time of the last edge change of the channel exceeds the specified interval time. The maximum interval time is 10 s. The last measure value is transferred up to this time (or -8888.0 if there is no valid value).

The measure value depends on the operating mode when a timeout occurs. There are basically three operating modes for timeout monitoring:

- Inactive

  No monitoring – the last measured value (or -8888.0 if there is no value) is returned. The "Tout_<xx>" signal always has the value "2 = not used".

- Intvl Predef

  If a timeout occurs, the "Tout_<xx>" signal is set to "1 = Timeout" and the predefined value returned.

- Intvl InpDep

  The "Tout_<xx>" signal is set to "1 = Timeout" and a value dependent on the measurement procedure and possibly the input level is returned (see Tab. 23-12).

Measurement Procedures	Value (Input Level Dependent)
No Measurement	-8888.0
High Time [µs] Low Time [µs]	0 0
Cycle Time --/-- [µs] Cycle Time --\-- [µs]	Time since the last edge [µs] Time since the last edge [µs]
Frequency --/-- [Hz] Frequency --\-- [Hz]	Reciprocal value of the time since the last edge [Hz] Reciprocal value of the time since the last edge [Hz]
Duty Cycle L/(L+H) --/-- Duty Cycle L/(L+H) --\-- Duty Cycle H/(L+H) --/-- Duty Cycle H/(L+H) --\--	1.0 with a low level, 0.0 with a high level at the input 1.0 with a low level, 0.0 with a high level at the input 0.0 with a low level, 1.0 with a high level at the input 0.0 with a low level, 1.0 with a high level at the input
Level (Active High) Level (Active Low)	0 0

Tab. 23-12 Return Values in a Timeout ("Intvl InpDep" Operating Mode)

For those values transferred when frequency and period duration are measured, the overflow of the 31-bit wide counter is taken into consideration, i.e. these can be as small or as large as required (frequency or time) as part of the 64-bit floating-point display.

The timeout is over when the 32-bit data word of the latest edge (time stamp + edge information) changes in comparison to the one transferred at the start of the timeout. This allows timeouts to be displayed that exceed the overflow value of the 31-bit counter (17.28 s). The "Tout_<xx>" signal is set to the value "0 = no Timeout".

23.15.3 Globals (ES5340-MeasTime Device)

The ES5340-MeasTime device is used to specify and configure time-based measurements. Every ES5340-MeasTime device offers 10 measurements which can be applied freely to the 20 hardware channels of the LABCAR Software. Up to four ES5340-MeasTime devices are supported per LABCAR Software so that a total of 40 measurements can be configured.

No settings are to be made in the "Globals" tab.

23.15.4 Groups (ES5340-MeasTime Device)

The ES5340-MeasTime device has one signal group (Fig. 23-21) which is transferred from the LABCAR Software to the experimental target. It contains all measure data such as measure values, trigger and timeout information.

An operating system task with periodic activation is to be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

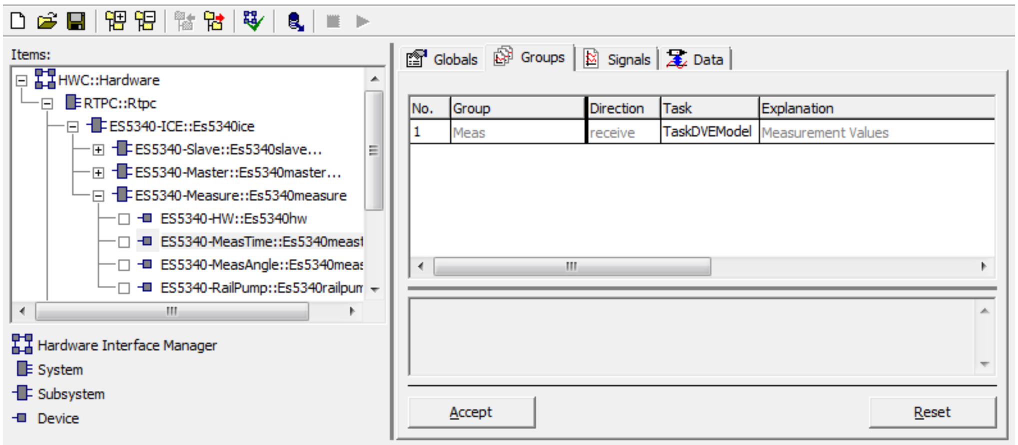


Fig. 23-21 The "Groups" Tab of the ES5340-MeasTime Device

The RTIO Signals of the "Meas" Group

The "Meas" group consists of 20 RTIO signals.

The RTIO signals "MeasVal_00" to "MeasVal_09" contain the measure values of the 10 measurements. These are either measure values which have been determined in the normal way or (in the case of a timeout) the timeout value intended for this case.

If a measurement is not used, 8888.0 is assigned to the relevant measure value. The physical unit of the measure values depends on the measurement procedure:

- time measurements (timestamp, (additive) pulse-width measurements, period durations) are specified in microseconds
- frequency measurements are in hertz
- voltage measurements are in volts
- integral evaluation in V*µs
- all other measurements (duty cycles, pulse count) are dimensionless

The RTIO signals "Tout_00" to "Tout_09" contain the results of the timeout monitoring for the relevant measure value.

Tab. 23-13 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
MeasVal_00 ... MeasVal_09	real64	Measure value. If the measurement is not used, -8888.0 is issued as a value. Physical unit of the measure value: - time measurements are in µs - frequency measurements are in Hz - angle measurements are in °CA - voltage measurements are in V
Tout_00 ... Tout_09	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-13 ES5340-MeasTime Device: the RTIO Signals of the "Meas" Signal Group

23.15.5 Signals (ES5340-MeasTime Device)

The 32 measurements of an ES5340-MeasTime device are configured in the "Signals" tab. Fig. 23-22 shows the RTIO parameters of the "Signals" tab – Tab. 23-14 on page 799 summarizes the properties of the individual parameters.

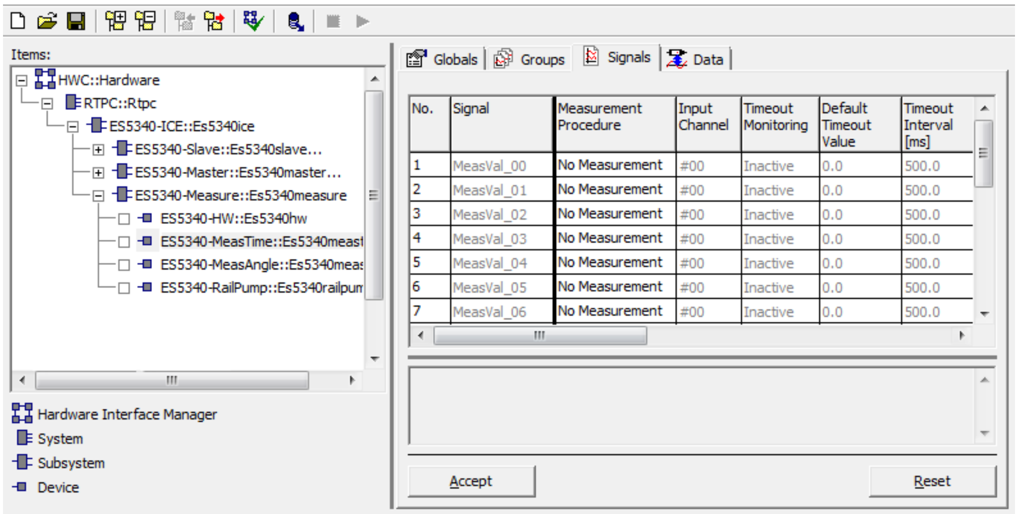


Fig. 23-22 The "Signals" Tab of the ES5340-MeasTime Device

All parameters can be edited during runtime of the model on the experimental target.

Measurement Procedure

This is where the measurement procedure is selected. The section 23.15.5 on page 797 contains detailed descriptions and configuration guidelines for the individual procedures. The notation --/-- denotes evaluation at rising edges whereas --\-- denotes evaluation at falling edges.

If "No Measurement" is set in the list field, no measurement takes place.

Note

To avoid unnecessary computing time, measurements which are not required should be deactivated.

Input Channel

This is where you specify the hardware channel on which measurement is to take place.

Timeout Monitoring

With asynchronous measurements, timeout monitoring takes place in periodic intervals. The time can be set separately for each hardware channel. A timeout is determined when no edge is determined on the hardware channel relevant to the measurement during the timeout period. Every edge on the hardware channel resets timeout monitoring.

In the "Timeout Monitoring" list field, timeout monitoring can be deactivated ("Inactive" option) or activated ("Intvl Predef" and "Intvl InpDep" options). The difference between the options "Intvl Predef" and "Intvl InpDep" is the measure value which, in the case of a timeout, is transferred to the model. If the "Intvl Predef" option is set, the value specified in the numeric input field "Default Timeout Value" is transferred. With the "Intvl InpDep" option, the value returned depends on the measurement procedure, although this option is actually only available with specific measurement procedures.

Default Timeout Value

This numeric input field can only be edited if "Intvl Predef" has been set in the "Timeout Monitoring" field. In this case, the input field defines the measure value which is transferred to the model in the case of a timeout.

Timeout Interval

This is used to determine the timeout interval – the value range is between 0.1 ms and 10000 ms.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Measurement Procedure	uint32	Yes	Measurement procedure: "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	Yes	Hardware channel on which the measurement is executed. 00 ... 19: channel number
Timeout Monitoring	uint8	Yes	Timeout monitoring. 0: "Inactive" 1: "Intvl Predef" 2: "Intvl InpDep"
Default Timeout Value	real32	Yes	Measure value in the case of a time-out. Only relevant in the "Intvl Predef" mode for timeout monitoring.
Timeout Interval	float	Yes	0.1...10000 ms
* Data type which the RTIO driver uses internally for the parameter			

Tab. 23-14 ES5340-MeasTime Device: Configuration Parameters of the "Signals" Tab

23.16 ES5340-MeasAngle Device

23.16.1 Pulse-Width Measurements

This section describes the properties of pulse-width measurements. The functioning of the following measurements is described:

- Angular: H-Time n-th Pulse (Pu Qual.) [µs]
- Angular: L-Time n-th Pulse (Pu Qual.) [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the pulse to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef" (the "Intvl InpDep" option is not supported)
- Default Timeout Value

Functional Description

Pulse-width measurements are pulse-qualified measurements, i.e. only those pulses are measured and counted whose opening and closing edge are within one and the same angle window (valid pulse). The pulse-width of the n-th valid pulse in the selected angle window is determined. The measure value is calculated at the upper angle window limit.

Fig. 23-23 shows how this works using the example of a low-active pulse-width measurement.

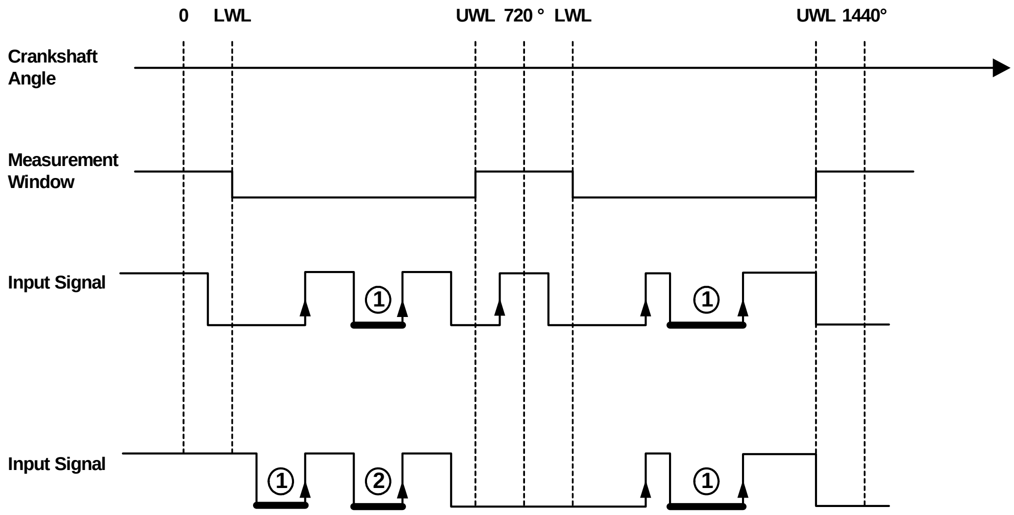


Fig. 23-23 Representation of how pulse-qualified, low-active pulse-width measurement (Angular: L-Time n-th Pulse (Pu Qual.) [µs] measurement procedure) works. Measured pulses or pulses which are counted are indicated by the bold lines.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the lowtime of the fifth pulse is to be measured, but only four pulses occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

23.16.2 Pulse-Width Measurements with Validate Signal

In the case of a pulse-width measurement with a Validate signal, a pulse on the measure channel is only measured or counted if it completely surrounds a pulse on the second channel (referred to below as the Validate channel). The Enable and Validate pulses can be defined either as high-active or low-active.

The following measurements are possible:

- Angular: H-Time n-th Pulse (H-Valid.) [µs]
- Angular: H-Time n-th Pulse (L-Valid.) [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the measure channel
- Pulse Number

  Defines the number of the pulse to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place

- Timeout Monitoring
  Either "Inactive" or "Intvl Predef" ("Intvl InpDep" is not supported)
- Default Timeout Value

Note

The number of the Validate channel is predefined. It is shown in the "Reference Channel" list field. The hardware channels 0 and 1 can make a pair whereby 0 can be selected as a measure channel, channel 1 is then the Validate channel. Channel 1 can, however, also be selected as measure channel; in this case, channel 0 would be the Validate channel. The same is true for each of the channel pairs (2, 3), (4, 5), (6, 7), (8, 9), (10, 11), (12, 13), (14, 15), (16, 17), (18, 19).

Functional Description

Fig. 23-24 shows how angle-synchronous pulse-width measurements with Validate option work using a hightime measurement with a high-active Validate signal. Only those pulses are taken into consideration whose opening and closing edge are within one and the same angle window, and which completely surround an active Validate pulse ("validated pulse").

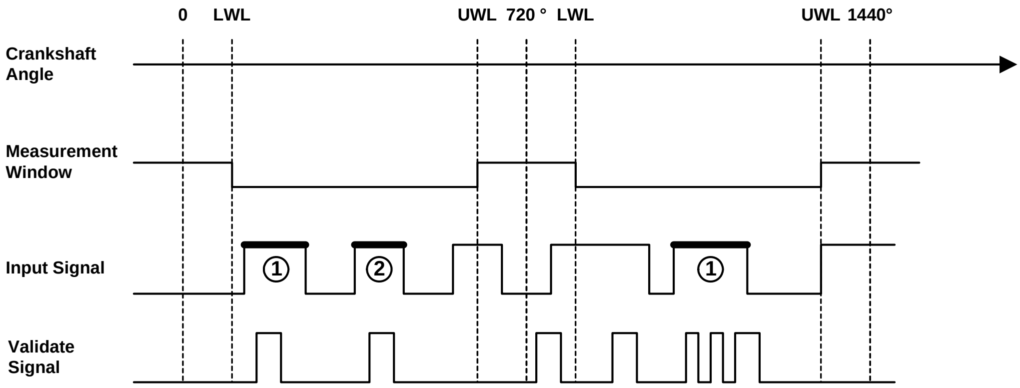


Fig. 23-24 Representation of how an angle-synchronous pulse-width measurement works with a Validate signal ("Angular: H-Time n-th Pulse (H-Valid) [µs]" measurement procedure). Validated pulses are indicated by bold lines.

The n-th validated pulse within an angle window is measured. The pulse number is defined in the "Pulse Number" list field of the "Signals" tab.

The measured pulse widths are transferred to the model at the upper angle window limit.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the lowtime of the fifth validated or enabled pulse is to be measured, but only four such pulses occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

23.16.3 Additive Pulse-Width Measurements

This section summarizes the properties of additive pulse-width measurements. The functioning of the following measurements is described:

- Angular: Additive Hightime [µs]
- Angular: Additive Lowtime [µs]

Admissible measurement options are:

- Hardware Channel

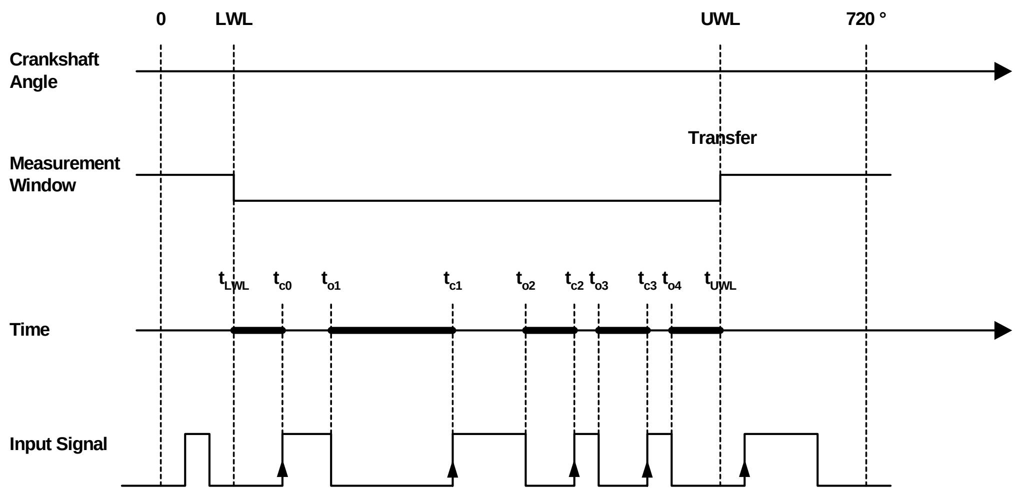
  Defines the hardware channel
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  All three options (Inactive, Intvl InpDep and Intvl Predef) are supported.
- Default Timeout Value

Functional Description

The additive time is the total of all time segments within an angle window in which the signal is active, regardless of whether the opening or closing edges of the pulses are inside or outside the angle window. Fig. 23-25 shows an example of measurement value calculation with an additive lowtime measurement. The additive time is transferred to the model at the upper angle window limit.



$$t_{Additive} := (t_{c0}-t_{LWL}) + (t_{UWL}-t_{o4}) + \Sigma\,(t_{ci} - t_{oi}), \quad i = 1 \ldots 3$$

Fig. 23-25 Representation of the functioning of an angle-synchronous additive lowtime measurement (Angular: Additive Lowtime [µs] measurement procedure). The additive time consists of the total of the bold line segments.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected when no edge occurred within the angle window.

Tab. 23-15 shows a list of measure values which are transferred to the model if a timeout occurs when "Intvl InpDep" is selected in the "Timeout Monitoring" list field.

Measurement Procedure	Signal Level in the Angle Window	Measure Value in a Timeout
Angular: Additive Hightime [µs]	low	0
	high	Time difference: time of upper angle window limit – time of lower angle window limit ($t_{UWL} - t_{LWL}$)
Angular: Additive Lowtime [µs]	low	Time difference: time of upper angle window limit – time of lower angle window limit ($t_{UWL} - t_{LWL}$)
	high	0

Tab. 23-15 Additive Pulse-Width Measurements: Measure Value in a Timeout when "Intvl InpDep" is set in the "Timeout Monitoring" List Field

23.16.4 Measuring Edges: Angle Stamp

This section describes the acquisition of angle stamps of edges. The functioning of the following measurements is described:

- Angular: Rising Edge of n-th Pulse [°CA]
- Angular: Falling Edge of n-th Pulse [°CA]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Pulse Number

  Defines the number of the edge to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- CS Angle Reference

  Reference angle
- Timeout Monitoring

  Either Inactive or Intvl Predef; "Intvl InpDep" is not supported
- Default Timeout Value

Functional Description

The measurements determine the angle of the n-th rising or falling edge within an angle window. The edge counter is reset to 0 when the lower angle window limit is exceeded.

The crankshaft angle of an edge delivered by the hardware is generally in the range [0 °CA, 720 °CA[. The measured angle is mapped by the LABCAR Software firmware in the range [LWL, LWL + 720 °CA[and the difference is then calculated using the reference angle. This difference is the measure value provided by measurement; it is positive when the angle in question is less than the reference angle (cf. Fig. 23-26 on page 805).

An example: let's assume an angle window of [-480 °CA, 120 °CA] has been defined. The reference angle should be around -80 °CA. In a first case, the measured angle should be around 60 °CA. No mapping is necessary for angle values between 0 °CA and (-480 °CA + 720 °CA = 240 °CA); the measure value supplied is (-80 °CA - 60 °CA = -140 °CA).

In a second case, the measured angle should be around 540 °CA. This angle is greater than 240 °CA and is mapped so that it is in the angle window. 720 °CA are thus subtracted from the angle value (540 °CA - 720 °CA = -180 °CA). The measure value is then (-80 °CA + 180 °CA = 100 °CA).

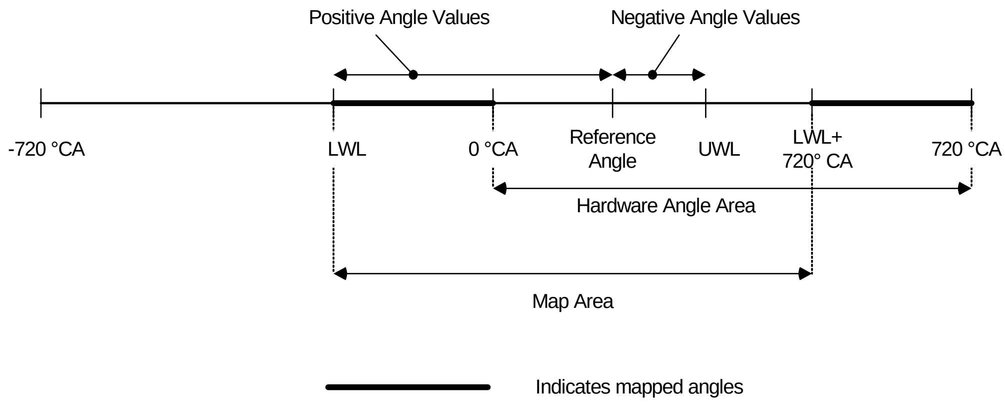


Fig. 23-26 How Angle Measurement Works

As the reference angle can be in the range [-720 °CA, 720 °CA], the measure functions return angle values from the range]-1440 °CA, 1440 °CA].

Note

The resolution of the measured angles is specified by the angle clock generator.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. If the measure value could not be calculated because, for example, the angle of the fifth rising edge is to be measured, but only four such edges occurred, the default timeout value defined by the user is transferred to the model as measure value. "Intvl InpDep" is not supported.

23.16.5 Measuring Edges: Time Stamps

This section describes the acquisition of time stamps of edges. The functioning of the following measurements is described:

- Angular: Time Stamp of n-th Rising Edge [µs]
- Angular: Time Stamp of n-th Falling Edge [µs]

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel

- Pulse Number

  Defines the number of the edge to be measured
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place.
- Timeout Monitoring

  Either "Inactive", "Intvl Predef" or "Intvl InpDep"
- Default Timeout Value

Functional Description

The measurements determine the time stamp of the n-th rising or falling edge within an angle window. The edge counter is reset to 0 if the lower angle window limit is exceeded.

Note

The time stamp returned has no connection to other time stamps in the LABCAR system. It is a purely internal LABCAR Software time stamp with a resolution of 8 ns.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected if the measure value could not be calculated because, for example, the time stamp of the fifth rising edge is to be measured, but only four such edges occurred.

If "Intvl InpDep" is set in the "Timeout Monitoring" list field, the time stamp determined in the last camshaft revolution to have no timeout is transferred as measure value to the model. In other words: the old time stamp is retained.

If "Intvl Predef" is set in the "Timeout Monitoring" list field, the default timeout value defined by the user is transferred to the model as measure value.

23.16.6 Pulse Counting

This section describes the properties of the following measurement procedures for pulse counting:

- Angular: Number of High-Pulses
- Angular: Number of Low-Pulses

Admissible measurement options are:

- Hardware Channel

  Defines the hardware channel
- Angle Window Number

  Defines the number of the angle window in which measuring is to take place
- Timeout Monitoring

  Either "Inactive" or "Intvl Predef"
- Default Timeout Value

Functional Description

The measurement procedures determine the number of valid active pulses within an angle window. Valid pulses are pulses which are in the same window with both an opening and a closing edge.

The total determined is transferred to the model at the upper angle window limit. (Fig. 23-27).

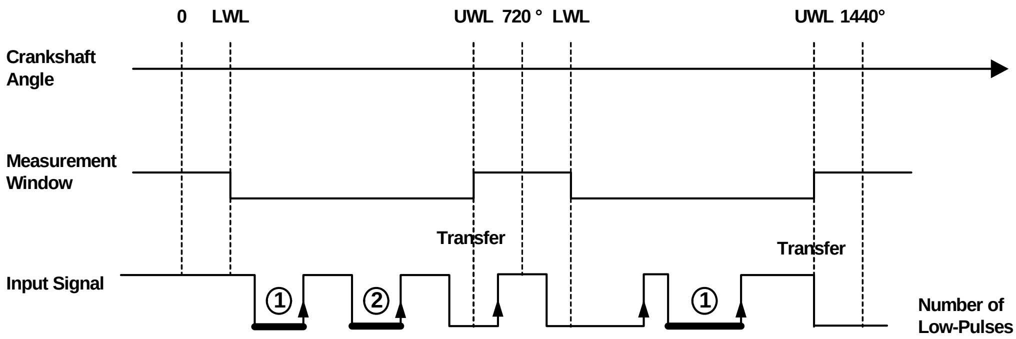


Fig. 23-27 Representation of how angle-synchronous measurement procedure for pulse count (Angular: Number of Low-Pulses Measurement Procedure) takes place. Pulses which are counted are indicated by bold lines.

Timeout Monitoring

With active timeout monitoring, monitoring takes place every 720 °CA at the upper angle window limit. A timeout is detected when no edge occurred in the angle window.

If "Intvl Predef" is set in the "Timeout Monitoring" list field, the default timeout value defined by the user is transferred to the model as measure value.

23.16.7 Globals (ES5340-MeasAngle Device)

The ES5340-MeasAngle device is used to specify and configure angle-based (synchronous) measurements. Every ES5340-MeasAngle device offers 32 measurements which can be applied freely to the 20 hardware channels of the LABCAR Software. Up to 9 ES5340-MeasAngle devices are supported per LABCAR Software so that a total of 288 measurements can be configured.

No settings are to be made in the "Globals" tab.

23.16.8 Groups (ES5340-MeasAngle Device)

The ES5340-MeasAngle device has one signal group which is transferred from the LABCAR Software to the experimental target. It contains all measure data such as measure values, trigger and timeout information.

An operating system task with periodic activation is to be assigned to this signal group. The activation period depends on the dynamic behavior and period duration of the signals to be measured.

The RTIO Signals of the "MeasVal" Group

The "MeasVal" group consists of 65 RTIO signals. The "TriggerBitField_00" signal must be interpreted as a bit field. It consists of 32 bits. The trigger or update data of the 32 measurements is coded in this bit field, i.e. it shows which measure values have been updated for the "MeasVal" signal group since the read process was last activated. If a bit is set, the measure value of the relevant measurement has been newly determined. It does not matter whether the measure value was updated as part of a timeout or due to a regular measure value calculation; the update bit of the measurement is set in both cases.

The RTIO signals "MeasVal_00" to "MeasVal_31" contain the measure values of the 32 measurements. These are either measure values which have been determined in the normal way or (in the case of a timeout) the timeout value intended for this case.

If a measurement is not used, 8888.0 is assigned to the relevant measure value. The physical unit of the measure values depends on the measurement procedure:

- time measurements (timestamp, (additive) pulse-width measurements, period durations) are specified in microseconds
- angle measurements and angle stamps are specified in degrees crankshaft (°CA)
- all other measurements (duty cycles, pulse count) are dimensionless

The RTIO signals "Tout_00" to "Tout_31" contain the results of the timeout monitoring for the relevant measure value.

Tab. 23-16 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
TriggerBitField_00	uint32	Bit field with update information of the 32 measurements. Measurement 0: LSB Measurement 31: MSB Bit value 0: measure value unchanged Bit value 1: measure value updated
MeasVal_00 ... MeasVal_31	real64	Measure value. If the measurement is not used, -8888.0 is issued as a value. Physical unit of the measure value: - time measurements are in µs - frequency measurements are in Hz - angle measurements are in °CA - voltage measurements are in V - integral evaluations are in V*µs
Tout_00 ... Tout_31	uint8	Result of timeout monitoring 0: no timeout 1: timeout 2: timeout monitoring is inactive
* Data type which the RTIO driver uses internally for the parameter		

Tab. 23-16 ES5340-MeasAngle Device: the RTIO Signals of the "MeasVal" Signal Group

23.16.9 Signals (ES5340-MeasAngle Device)

The 32 measurements of an ES5340-MeasAngle device are configured in the "Signals" tab. Fig. 23-28 shows the RTIO parameters of the "Signals" tab – Tab. 23-17 on page 811 summarizes the properties of the individual parameters.

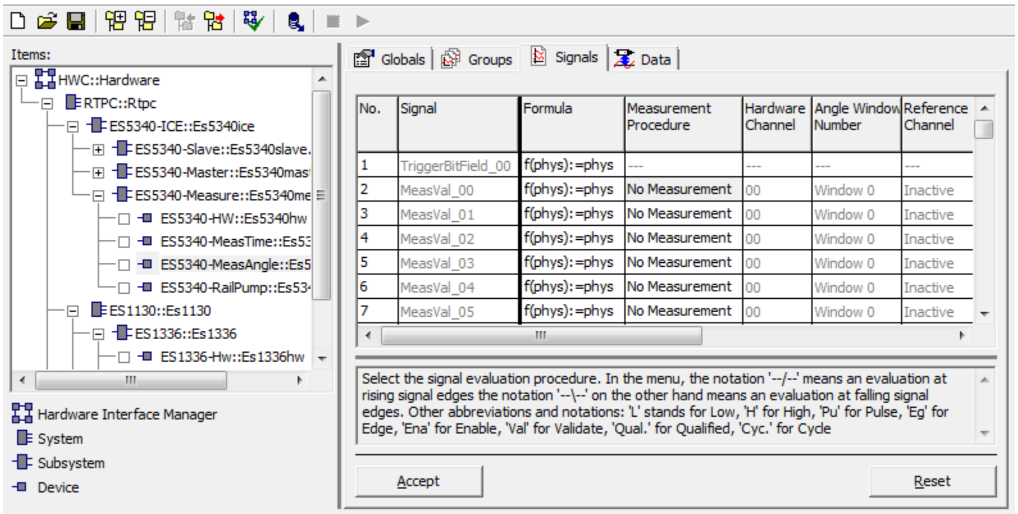


Fig. 23-28 The "Signals" Tab of the ES5340-MeasAngle Device

All parameters can be edited during runtime of the model on the experimental target.

Measurement Procedure

This is where the measurement procedure is selected. The sections on page 800ff contain detailed descriptions and configuration guidelines for the individual procedures. The notation --/-- denotes evaluation at rising edges whereas --\-- denotes evaluation at falling edges.

If "No Measurement" is set in the list field, no measurement takes place.

Note

To avoid unnecessary computing time, measurements which are not required should be deactivated.

Hardware Channel

This is where you specify the hardware channel on which measurement is to take place.

Angle Window Number

This list field can only be edited if an angle-synchronous measurement procedure has been selected in the "Measurement Procedure" list field. The angle window is defined within which the measurements are to be executed. Three angle windows are available; these are the angle windows of the assigned hardware channel (for more information see "Signals (ES5340-HW Device)" on page 791).

Reference Channel

The pulse-width measurements with "Validate" signal described in section 23.16.2 on page 801 require a second hardware channel for the purposes of execution: the channel at which the Enable or Validate signal is pending. The "Reference Channel" field displays the hardware channel at which (when this kind of measurement is selected) the Enable or Validate signal is to be connected.

The field is a display field only, which means it cannot be edited and is of no importance for all other measurements.

Pulse Number

This numeric input field can only be edited if an angle-synchronous measurement procedure has been selected in the "Measurement Procedure" field. With pulse- and edge-selective measurements, the number of the pulse or edge to be measured is specified in this field.

This field is of no importance for additive measurements and pulse counts, and is therefore disabled.

CS Angle Reference

The angle reference in degrees crankshaft to which all angle measurements refer. This parameter is of no importance to all other measurements and is thus disabled.

Timeout Monitoring

With angle-synchronous measurements, timeout monitoring takes place every 720 °CA, at the upper angle window limit in each case. A timeout is detected when no measure value could be calculated. For example, a timeout is detected when the hightime of the fifth pulse in the angle window is to be measured, but only four pulses occurred.

In the "Timeout Monitoring" list field, timeout monitoring can be deactivated ("Inactive" option) or activated ("Intvl Predef" and "Intvl InpDep" options). The difference between the options "Intvl Predef" and "Intvl InpDep" is the measure value which, in the case of a timeout, is transferred to the model. If the "Intvl Predef" option is set, the value specified in the numeric input field "Default Timeout Value" is transferred. With the "Intvl InpDep" option, the value returned depends on the measurement procedure, although this option is actually only available with specific measurement procedures. For more details, refer to the sections starting on page 800.

Default Timeout Value

This numeric input field can only be edited if "Intvl Predef" has been set in the "Timeout Monitoring" field. In this case, the input field defines the measure value which is transferred to the model in the case of a timeout.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Measurement Procedure	uint32	Yes	Measurement procedure: "No Measurement" (0) "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	Yes	Hardware channel on which the measurement is executed. 0 ... 19: channel number
Angle Window Number	uint8	Yes	Angle window selection (value range: 0, 1, 2) Only relevant / editable with angle-synchronous measurements.
Reference Channel	-	No	Reference channel for measurements with Enable and Validate signal. Display field only; cannot be edited.
Pulse Number	uint16	Yes	With pulse- and edge-selective measurements: the number of the pulse or edge to be measured (value range: 1, 2, ... 32)
CS Angle Reference	real64	Yes	Angle reference in degrees crankshaft to which all angle measurements refer (Value range: [-720.0 °CA, 720.0 °CA[)
Timeout Monitoring	uint8	Yes	Timeout monitoring. 0: "Inactive" 1: "Intvl Predef" 2: "Intvl InpDep"
Default Timeout Value	real32	Yes	Measure value in the case of a timeout. Only relevant in the "Intvl Predef" mode for timeout monitoring.

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-17 ES5340-MeasAngle Device: Configuration Parameters of the "Signals" Tab

23.17 ES5340-RailPump Device

23.17.1 Globals (ES5340-RailPump Device)

For each ES5340-RailPump device, four measure channels of the same measurement procedure, but with different angle windows, are available. Two ES5340-RailPump devices can be added, i.e. resources are available for a total of eight measure channels.

The following figure shows the "Globals" tab of the ES5340-RailPump device.

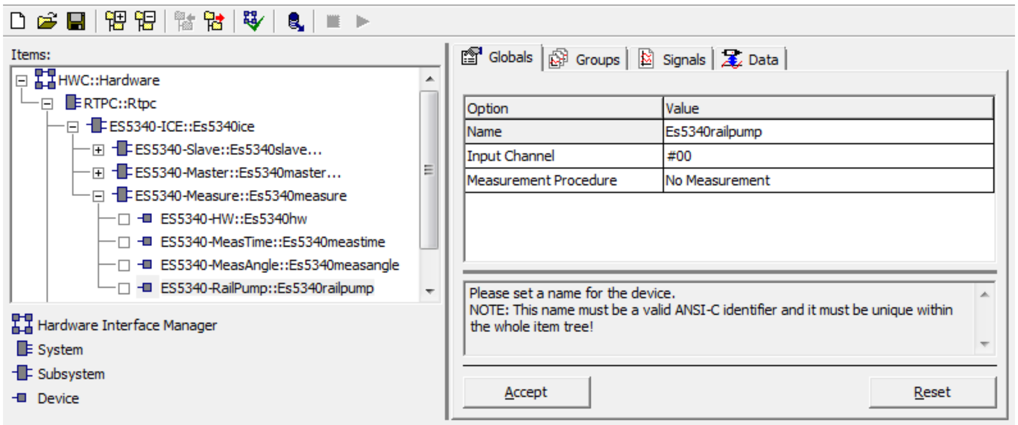


Fig. 23-29 The "Globals" Tab of the ES5340-RailPump Device

The following parameters can be set in this tab:

Input Channel

The input channel to be used is selected here – the level and threshold value settings of the ES5340-HW device are used. All four software measure channels of this ES5340-RailPump device refer to this hardware channel.

Measurement Procedure

This is where the measurement procedure used is selected for all four software measure channels of the RTIO element – in total, the following four measure methods are available:

- 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]

The angle value of the first or last rising or falling edge (respectively) which has occurred within an angle window (defined by LWL and UWL, see Fig. 23-30 on page 813) is measured in each case. The measure value results from a reference angle value minus this angle. Edges before the reference angle return a positive result, edges after the reference angle return a negative result.

If no corresponding edge occurs in the angle window, no result is transferred and a timeout is detected.

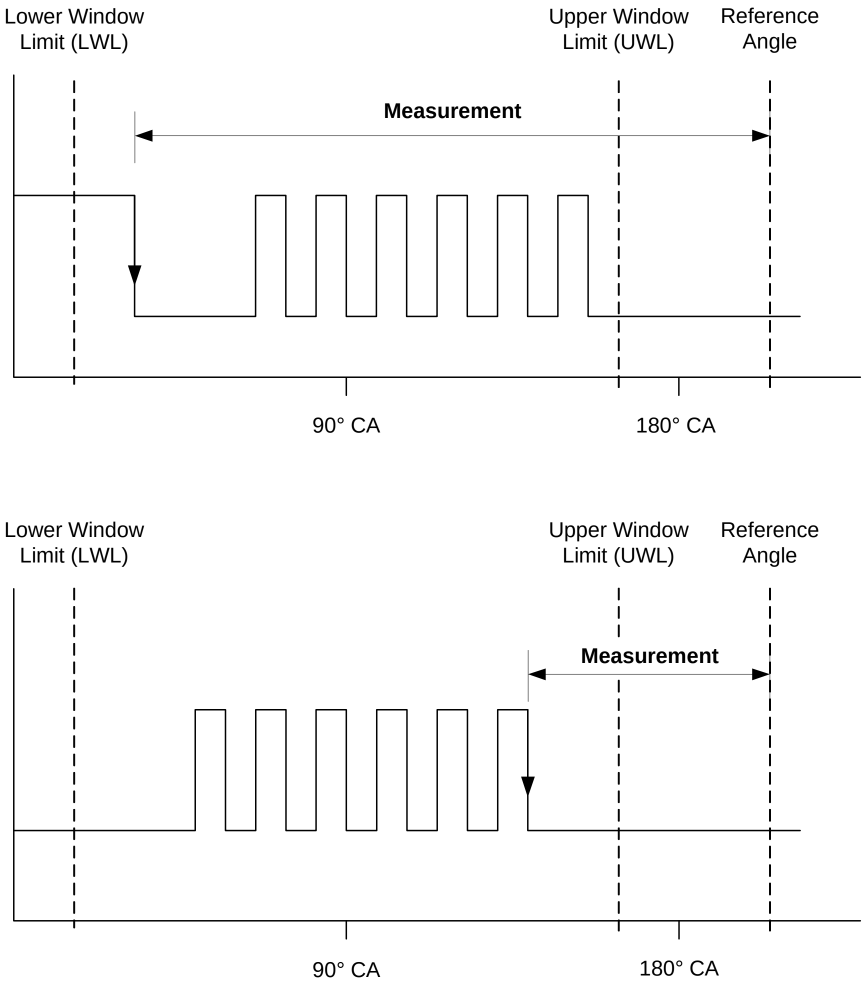


Fig. 23-30 Measure Values with "Angle of First Falling Edge of Pulse Sequence" (top) and "Angle of Last Falling Edge of Pulse Sequence" (bottom)

23.17.2 Signals (ES5340-RailPump Device)

The ES5340-RailPump device implements a "MeasVal" receive signal group.

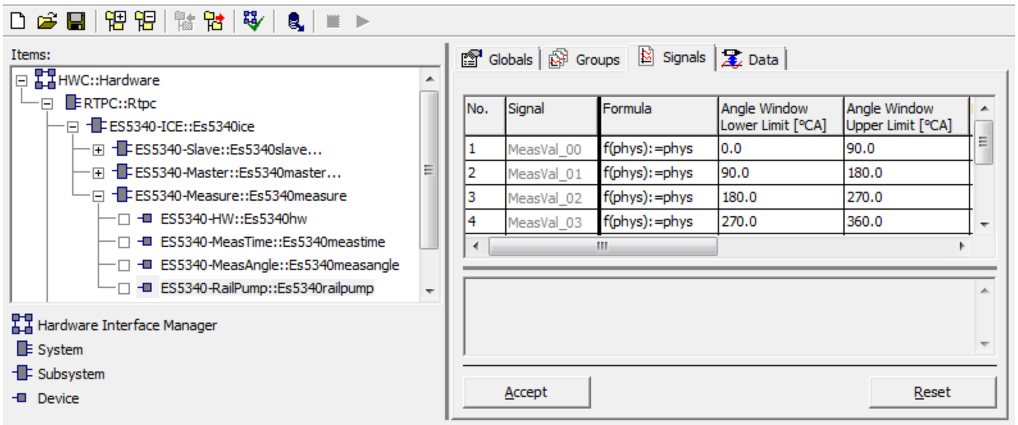


Fig. 23-31 The "Signals" Tab of the ES5340-RailPump Device

The following signals are created:

MeasVal_xx [xx = 00…03]

The result values for software measure channels 0…3 (value range -720.0…720.0 °CA, -8888.0 if channel not used)

Tout_xx [xx= 00…03]

Timeout signals for the software measure channels 0…3

- 0 = no timeout
- 1 = timeout detected
- 2 = timeout not configured
- 4 = angle interface not synchronous

Offset Angle

Angle window and reference angle can be shifted by this angle with respect to the crankshaft angle.

In the "Signals" tab, the following settings can be made for each individual measure channel.

Angle Window Lower Limit

The starting angle of the measure window (corresponds to "LWL" in Fig. 23-30 on page 813).

Angle Window Upper Limit

The concluding angle of the window (corresponds to "UWL" in Fig. 23-30 on page 813)

Reference Angle

The reference angle for calculating the measure value.

Timeout Monitoring

The operating mode for a timeout.

- Inactive
  No timeout detection; the previous value is transferred
- Intvl Predef
  The substitute value for a timeout is transferred

Default Timeout Value

The substitute value for a timeout.

23.18 Arbitrary Signal Generators

There are eight analog and eight digital signal generators available on the LABCAR Software. Each of the signal generators can play back one of the 16 waveforms. A central RPM generator and one variable clock generator per signal generator (maximum frequency: 1 MHz) are available as clock sources.

One individual basic phase as well as an additional phase shift can be controlled per signal generator. The speed at which a change of the phase shift takes effect can be defined.

When using the variable clock generator, the frequency of the clock generator, the trigger mode (single shot, continuous) and a trigger signal can be specified.

The amplitude of the internal output signal of the signal generator can be varied between 0.0 and 1.0.

23.18.1 RPM Generator

The LABCAR Software has a central speed generator (RPM generator) which outputs an engine-speed-specific clock signal. This clock signal can be used by the signal generators to read out and output the waveforms. The maximum speed is 30000 rpm, the resolution in around 0.1 rpm. The speed signal itself can be modulated using a misfire generator.

For measuring purposes, the speed signal can be applied to the BNC port on the front panel of the LABCAR Software.

Angular Resolution

The angular resolution is 65536 points per cycle. With a typical four-stroke engine with a period of 720 °CA, this corresponds to an angular resolution of approx. 0.01 °CA.

23.18.2 Waveform Pool for Signal Generators

There are 16 waveforms available which can be used by the arbitrary signal generators. The user can decribe the waveforms with tables. The signal trace in the table is written to the relevant waveform using an interpolation procedure.

Waveform resolution:

The maximum resolution of a waveform is determined by the maximum possible number of 65536 data points. Here too, the resolution can be reduced to 16 points in powers of two; please note that the resolution (1/(number of data points)) of a waveform must be smaller than or equal to the angular resolution. Normally the resolution of a waveform should correspond to the angular resolution.

The waveforms are read out and output by the signal generators. Either the central RPM generator can act as clock source or a variable frequency generator (maximum frequency: 1 MHz) in the signal generator is used.

Waveform resolution smaller than angular resolution:

If a high-frequency signal is to be output via the signal generator (using the variable frequency generator), it might be necessary to keep the resolution of one waveform smaller than the angular resolution.

The following example illustrates the procedure:

If a sinusoidal signal of 40 kHz is to be output, the signal table describes a single sine period. The angular resolution is 65536 points. Due to a maximum frequency of the variable clock generator of 1 MHz, the maximum signal frequency for the sinusoidal signal is 1 MHz/65536 = 15.25 Hz which, of course, is considerably less than the desired 40 kHz. By reducing the waveform resolution to, for example, 16 data points, the sinusoidal signal is stored several times in succession (in fact 65536/16 = 4096-fold) in the waveform with 65536 data points. This results in a total maximum frequency for the sinusoidal signal of 1 MHz/16 = 62.5 kHz, which is above the desired frequency of 40 kHz. Due to a corresponding reduction of the variable clock rate to 640 kHz, the desired sinusoidal signal can be generated with 40 kHz.

The example shows that due to a reduction in the waveform resolution in comparison to the angular resolution, the waveform resolution is not really reduced. The signal of the signal table is simply written to the waveform several times in succession and the "visible" resolution thus reduced.

23.18.3 Knock Signal Generator

The knocking which occurs with a combustion engine can be simulated by the knock signal generator. A knock signal consists of individual knock packages. A knock package itself consists of a sinusoidal oscillation with selectable frequency and an envelope curve which modulates the sinusoidal oscillation with a duration which can be defined.

The following figure shows an individual knock package. A sine half wave is used as an envelope curve.

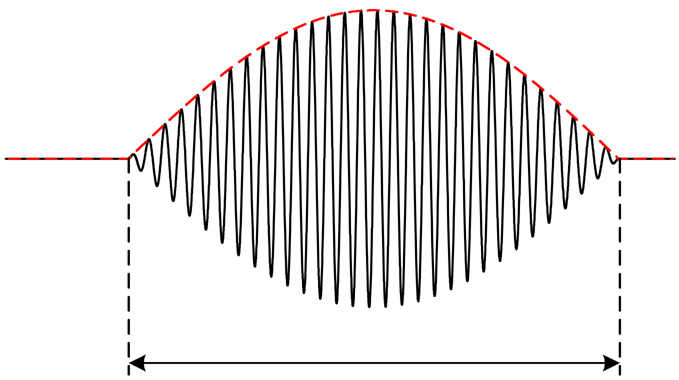


Fig. 23-32 A Knock Package

Non-knocking combustion also generates noises which are acquired by a real structure-borne noise knock sensor. A distinction is made between correct and knocking combustion via the control of the amplitude of the knock signal.

In addition, there is also a stochastic variation of the amplitude of a knock package. This is used for the simulation of variations in the knock signals which occur in real operation.

A certain amount of noise also exists if no knock package is being output. This basic noise is required for example to be able to get through the initial diagnostics of the sensor. Modern ECUs treat inputs without noise as faulty or not present.

The angular position (in °CA) of a knock signal as well as the occurrence of the knock event can now be controlled individually for each cylinder using a probability value or sequence tables (see "Sequence Tables" on page 818).

The knock signal generator has four internal outputs. You can select which cylinders serve the relevant output. In multi-cylinder vehicles, it is important that individual knock packages can overlay each other.

Note

A maximum of four waveforms can overlay each other!

23.18.4 Misfire Control

A relevant control mechanism is available on the LABCAR Software to simulate misfiring; this results in a modulation of the speed of the RPM generator in a specific angle range. It is possible to modify the speed in relation to the specified speed of the RPM generator (reduce/increase by the factor 0.01 to 2.0). When simulating misfiring, the speed is normally reduced in comparison to the defined speed.

The start effect angle of speed modulation can be defined for each individual cylinder. The effect of speed modulation can be controlled for each cylinder using a probability value or sequence tables (see "Sequence Tables" on page 818).

Speed modulation can be defined via four modulation profiles which represent the course of modulation over a complete period of 720 °CA (or 360 ° for two-stroke engines). 1.0 represents a non-existent modulation; 0.01 reduces the speed to 1% of the specified speed; 2.0 doubles the specified speed. One of the four available modulation profiles can be selected individually per cylinder.

23.18.5 Sequence Tables

Sequence tables are used with the misfire generator and the knock signal generator: these make it possible for the user to describe complex knock and misfiring sequences.

A table with a maximum of 100 data points is used for this purpose. Once the sequence has been started, the sequence proceeds one data point per period. In the case of misfiring, a value greater than 0.5 at the relevant data point means that misfiring occurs in this period. With the knock signal generator, this value in the table can also be used to define the intensity with which the knock sensor perceives the knock signal (close cylinder: high value, distant cylinder: low value).

After 100 data points, the sequence is either started from the beginning again, or play-back is terminated and has to be restarted via the relevant trigger signal.

It is possible to specify one individual sequence per cylinder. There is, however, one common sequence ("Common Sequence") both with the misfire generator and the knock generator which all cylinders can access. This facilitates the fast setting of sequences which are to be used for several cylinders.

23.18.6 MSA Sensor

Signal generators are also used to simulate crankshaft sensors which can detect the direction of rotation (MSA sensors). A tooth pulse has no fixed angle width but a fixed pulse duration. Moreover, the output signal is predefined as being a low-active open collector signal.

If an MSA sensor RTIO element is used, (potential) tooth center information is calculated for all waveform traces during configuration and stored in the waveform pool. However, not all waveforms are necessarily suitable for this algorithm; when an unsuitable waveform is selected, an error message is issued.

23.19 ES5340-SigGen Subsystem

23.19.1 Globals (ES5340-SigGen Subsystem)

The ES5340-SigGen Subsystem is used to set RTIO parameters which are effective globally, i.e. which have an effect on all elements of the ES5340-SigGen Subsystem.

The following figure shows the RTIO parameters of the "Globals" tab.

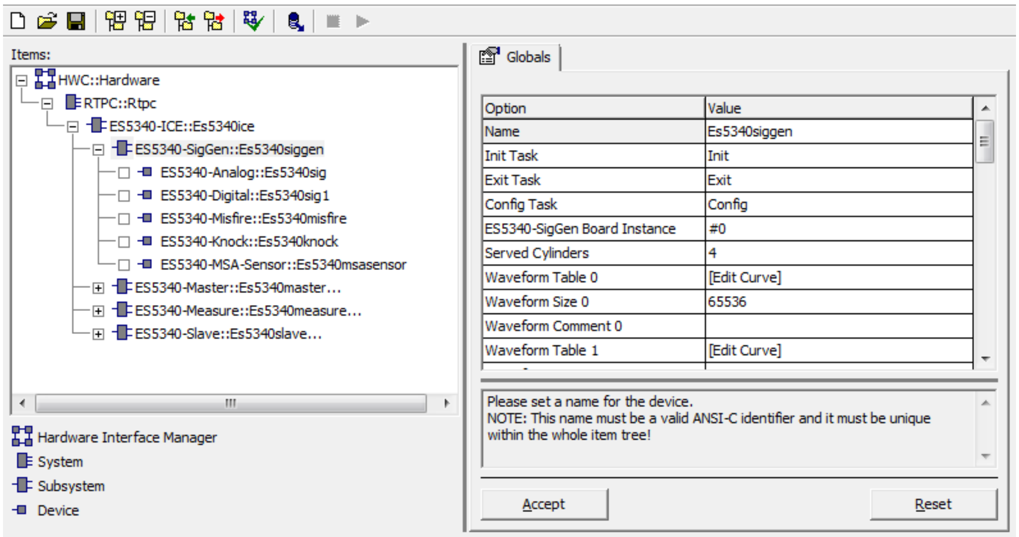


Fig. 23-33 The "Globals" Tab of the ES5340-SigGen Subsystem

Served Cylinders

This is where the number of cylinders supported by the knock signal generator and with misfire control is set. Possible values are 1 ... 12.

Waveform Table 0…15

This is where the signal shapes of the 16 waveforms can be specified.

Waveform Size 0…15

The number of data points of the 16 waveforms can be specified using the corresponding options fields (see "Waveform resolution" on page 816 and "Waveform resolution smaller than angular resolution" on page 816).

Waveform Comment 0…15

An individual comment can be specified to obtain a simple overview of how the 16 waveforms are used.

Note

The comment field does not permit blanks or special characters.

MSA Sensors: Logic of Input Waveform

This parameter determines whether a signal with negative logic (neg. pulse corresponds to "tooth") or positive logic (pos. pulse corresponds to "tooth") is to be interpreted when the tooth centers (for an MSA sensor) are calculated. This setting applies to all twelve waveforms.

The parameter can be edited online; however, when a change occurs, all waveforms must be recalculated and written to the board in the waveform pool.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	**Data Type***	**Can be Edited Online**	**Comment/Value Range**
Served Cylinders	uint32	Yes	Supported cylinders for knocking and misfiring Value range: 1 … 12
Resolution	uint32	Yes	Angular resolution Value range: 16 … 65536 in powers of two
Waveform Table 0…15	Table	Yes	The relevant waveform as table Value range: -1.0 … +1.0
Waveform Size 0…15	uint32	Yes	Number of data points of the relevant waveform Value range: 16 … 65536 in powers of two
Waveform Comment 0…15	Identi-fier	No	Comment on the relevant wave-form
MSA Sensors: Logic of Input Waveform	bool	Yes	Interpretation of wave form for MSA sensor Value range: 0 = FALSE (neg. logic) |, 1 = TRUE (pos. logic)

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-18 ES5340-SigGen Subsystem: Configuration Parameters of the "Globals" Tab

23.20 ES5340-SigGen Device

The ES5340-Analog device is used to configure and control one of the arbitrary signal generators on the LABCAR Software.

23.20.1 Globals (ES5340-Analog Device)

The ES5340-Analog device is used to configure and control one of the six arbitrary signal generators on the LABCAR Software.

The following figure shows the RTIO parameters of the "Globals" tab.

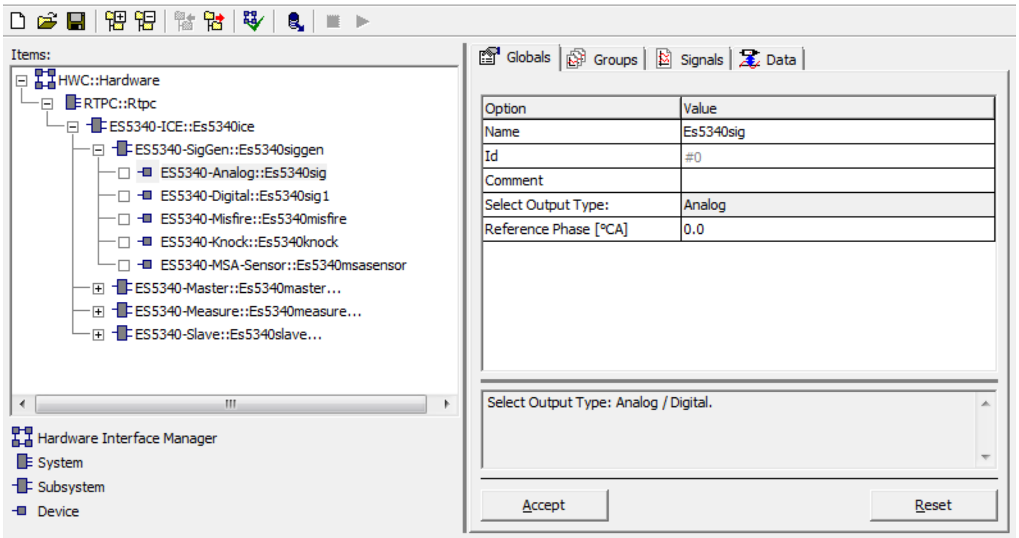


Fig. 23-34 The "Globals" Tab of the ES5340-Analog Device

Comment

A comment can be made for this signal generator.

Note

The comment field does not permit blanks or special characters.

Select Output Type

Analog or digital.

Reference Phase

This parameter describes the basic phase shift of the signal generator. The actual phase is determined by adding this value to the "PhaseRef" signal (see "PhaseRef" on page 824).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Comment	Identi-fier	No	User-specific comment
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range -720...720 °CA

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-19 ES5340-Analog Device: Configuration Parameters of the "Globals" Tab

23.20.2 Groups (ES5340-Analog Device)

The ES5340-Analog device has one signal group which is used to control the signal generator.

23.20.3 Data (ES5340-Analog Device)

The "Data" tab lists the signals for controlling the signal generator.

The following figure shows the signals in the "Data" tab:

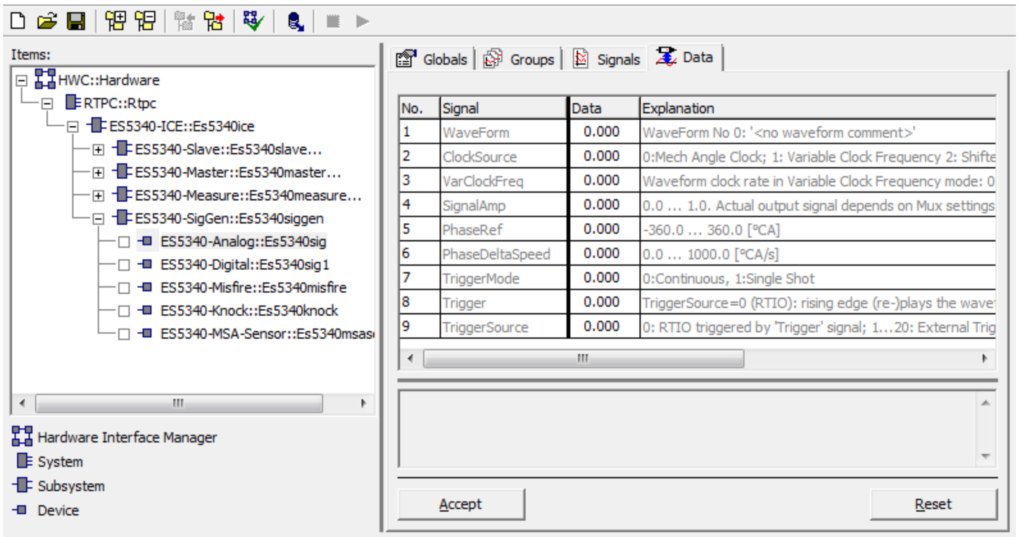


Fig. 23-35 The "Data" Tab of the ES5340-Analog Device

WaveForm

The "WaveForm" signal is used to select the waveform to be read out and output by the signal generator. The values 0 to 15 are possible. These correspond to the central waveforms in the ES5340-SigGen Subsystem.

ClockSource

The "ClockSource" signal is used to determine the clock source for the signal generator. You can choose between the central speed generator (RPM generator) and a variable clock generator. The value 0 corresponds to the central speed generator; the value 1 to the variable clock generator and the value 2 corresponds to the central speed generator with an offset ("Shifted Mech Angle Clock").

VarClockFreq

The "VarClockFreq" signal controls the frequency of the variable clock generator. The signal is only effective if the "ClockSource" signal is 1 (= variable clock generator). The maximum frequency of the variable clock generator is 1 MHz.

SignalAmp

The output amplitude of the signal generator is controlled via the "SignalAmp" signal. The values range from 0.0 to 1.0.

Please note that a voltage signal is not created until an internal signal is within the value range -1.0 to +1.0.

PhaseRef

This signal describes a phase shift which is added to the global parameter "ReferencePhase". This makes a model-controlled phase shift during runtime possible. Please note that the phase shift is takes place with a finite phase change speed (controlled by the parameter "PhaseDeltaSpeed"). This is why the "PhaseDeltaSpeed" signal must have a value greater than 0 for most applications.

PhaseDeltaSpeed

The phase change speed is controlled by the "PhaseDeltaSpeed" signal. A change of the value of the "PhaseRef" signal does not have an infinitely fast effect; rather the phase is calibrated with this speed. The values range from 0 (no calibration of the phase) to 1000 °CA/s.

TriggerMode

Note

The "TriggerMode" signal is only of importance if the variable clock generator is selected as clock source for the signal generator (see "ClockSource" on page 823).

The variable clock generator can work continuously ("TriggerMode" is 0) or it stops once a complete waveform has been played back once ("Single Shot") ("TriggerMode" is 1).

A phase shift is also taken into consideration during operation with the variable clock generator. This controls the starting point of play-back.

The trigger signal "Trigger" results in a start or new start.

Please note that the complete waveform is always played back over the total number of all data points. If the number of data points of the waveform is selected as being smaller than the angular resolution, the required output signal is stored several times in succession in the waveform pool. The required output signal is then played back several times when play-back takes place in "Single Shot" mode (see "Waveform resolution smaller than angular resolution" on page 816).

Trigger

Note

The "TriggerMode" signal is only of importance if the variable clock generator is selected as clock source for the signal generator (see "ClockSource" on page 823).

The "Trigger" signal enables you to start the variable clock generator. An edge of 0 → 1 on the "Trigger" signal starts the reading out and outputting of the waveform with the variable clock generator. In all cases, a corresponding edge leads to reading out and outputting being restarted even if the previous playback has not been completed or is in continuous mode.

TriggerSource

Note

The "TriggerMode" signal is only of importance if the variable clock generator is selected as clock source for the signal generator (see "ClockSource" on page 823).

This signal can be used to configure the triggering of the signal generators via software (LABCAR-RTC) or by one of the digital inputs 1 to 20 – a rising edge on the selected input triggers the clocking out of the waveform.

When using external triggers, "TriggerSource" must be set to 1 – at the same time, the value of "VarClockFreq" must not be zero.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
WaveForm	uint8	Yes	Selection of the waveform 0 - 11
ClockSource	uint8	Yes	Clock source of the signal generator 0: Mech Angle Clock 1: Variable Clock Frequency 2: Shifted Mech Angle Clock
VarClockFreq	real32	Yes	Frequency of the variable signal generator in Hertz. 0.0 - 1000000.0 [Hz]
SignalAmp	uint16	Yes	Signal amplitude of the output signal 0…65535 corresponds to 0.0…1.0
PhaseRef	sint16	Yes	Additive phase shift -32768…32767 corresponds to -360…+360 [°CA]
PhaseDeltaSpeed	uint16	Yes	Phase change speed 0 … 1000 [°CA/s]
TriggerMode	uint8	Yes	Trigger mode 0: Continuous 1: Single-Shot
Trigger	uint8	Yes	Trigger signal Edge of 0 → 1 starts play-back
TriggerSource	uint8	Yes	Trigger source Value range: 0...6 0 = LABCAR-RTC (default) 1...20 = Ext. trigger signal
* Data type which the RTIO driver uses internally for the parameter			

Tab. 23-20 ES5340-Analog Device: Signals of the "Data" Tab

23.21 ES5340-Misfire Device

The ES5340-Misfire device is used to configure and control the misfire control on the LABCAR Software.

23.21.1 Globals (ES5340-Misfire Device)

The following figure shows the RTIO parameters of the "Globals" tab.

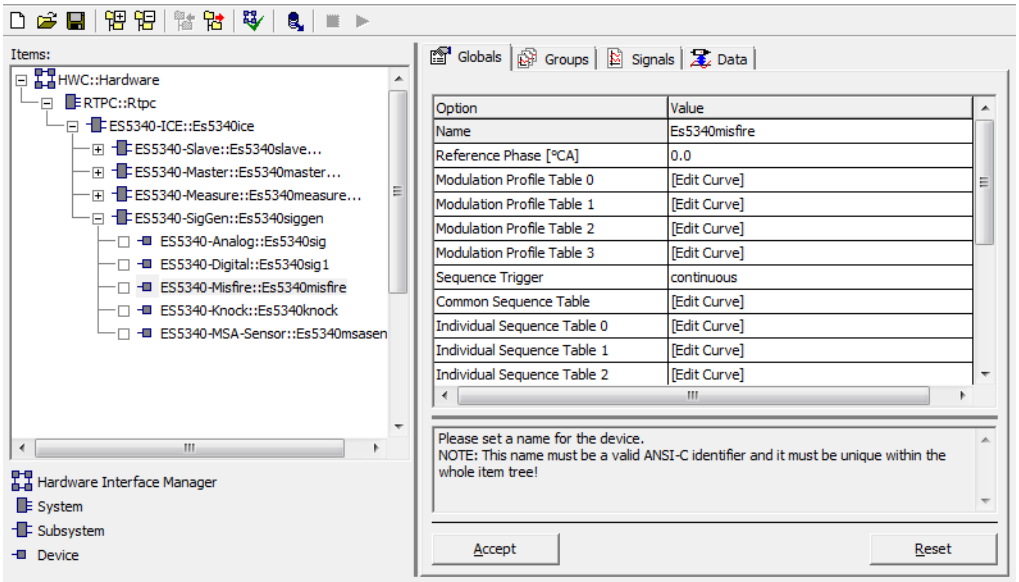


Fig. 23-36 The "Globals" Tab of the ES5340-Misfire Device

Reference Phase

This parameter describes the basic phase shift of the misfire control. The actual cylinder-specific phase is determined by adding this value to the relevant parameter "Start Angle" from the "Signals" tab.

Modulation Profile Table 0 … 3

Four modulation profiles are available which define a modulation of the predefined speed over a specific angle area (see "Misfire Control" on page 818). The modulation profiles can be specified using the table editor. In a modulation curve the speed modulation is plotted against the crankshaft angle. A value of 1 at an angle represents the unchanged acceptance of the predefined speed. A value less than 1 (to 0.01) lowers the speed. A value greater than 1 represents an increase in speed.

The first data point value of the signal table corresponds to the relevant cylinder-specific start angle in °CA. The signal table describes a complete period of 720 °CA.

The speed modulation of several cylinders can overlap – the individual cylinder-related modulation values are multiplied with each other.

Sequence Trigger

The activation of the speed modulation can be controlled for each cylinder using sequence tables (see "Sequence Tables" on page 818). This parameter is used to decide whether the sequence table should be played back continuously ("continuous") or only once ("single shot"). Play-back is started with the signal "Sqnc_Trigger" (see "Sqnc_Trigger" on page 831).

Common Sequence Table, Individual Sequence Table 0 … 11

There are 12 cylinder-individual sequence tables for controlling speed modulation (see "Sequence Tables" on page 818). There is also a "Common Sequence Table" which makes it possible to use one sequence on several cylinders.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range 0 … 720 °CA
Modulation Profile 0…3	table	Yes	The data points are mapped to a complete period (720 °CA). The values are within the range 0.01 to 2.0
Sequence Trigger	uint8	Yes	Play-back mode of the sequence tables: Continuous:0 Single Shot: 1
Common Sequence Table Individual Sequence Table 0…11	table	Yes	Sequence tables

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-21 ES5340-Misfire Device: Configuration Parameters of the "Globals" Tab

23.21.2 Groups (ES5340-Misfire Device)

The ES5340-Misfire device has a signal group in which the misfire control signals are processed.

23.21.3 Signals (ES5340-Misfire Device)

Misfire control is configured in the "Signals" tab. The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

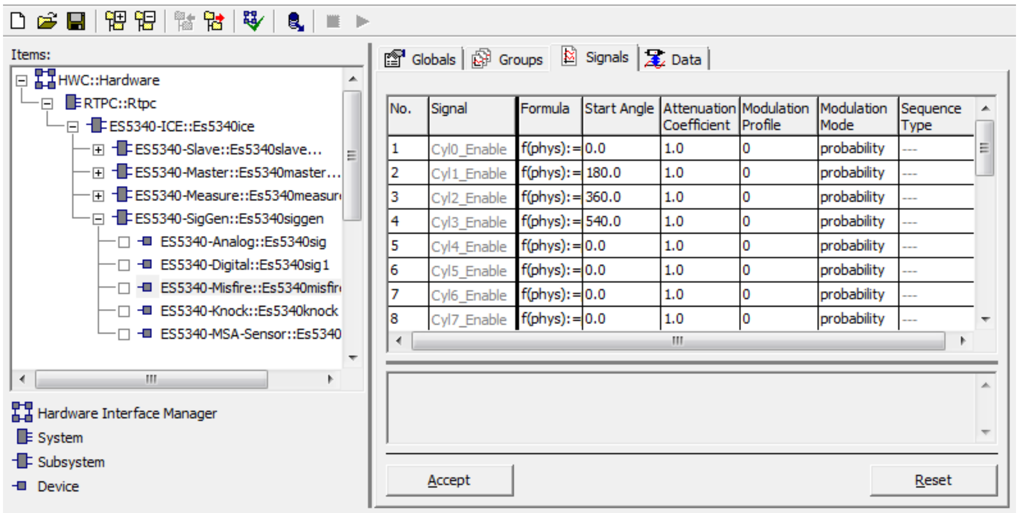


Fig. 23-37 The "Signals" Tab of the ES5340-Misfire Device

Start Angle

The option field "Start Angle" is used to determine the start angle of the misfire modulation profile for the relevant cylinder. The "Reference Phase" parameter from the "Globals" tab (see "Reference Phase" on page 827) is added to this angle.

Attenuation Coefficient

This parameter is used to reduce the degree of speed modulation for each individual cylinder. A value of 1.0 means that the speed modulation described in the selected modulation profile is applied completely. A value less than 1 reduces the effect in the relevant proportion; a value of 0.0 ultimately leads to no speed modulation being executed for the cylinder.

Modulation Profile

One of the four available modulation profiles from the "Globals" tab can be selected per cylinder.

Modulation Mode

The existence of misfiring in a period can be controlled either via a probability function or using sequence tables. You can toggle between these two operating modes using this parameter.

Sequence Type

When selecting "Modulation Mode = sequence", you can specify here whether the common sequence table ("common") or the cylinder-specific sequence table ("individual") is to be used (see "Sequence Tables" on page 818).

Parameter or Option Field	**Data Type***	**Can be Edited Online**	**Comment/Value Range**
Start Angle	real32	Yes	Start angle of the speed modulation 0.0 … 720.0 °CA
Attenuation Coefficient	real32	Yes	Proportional effect of the modulation profile 0.0 … 1.0
Modulation Profile	uint8	Yes	Selection of the modulation profile for the cylinder 0 … 3
Modulation Mode	uint8	Yes	Probability or sequence table for misfire control
Sequence Type	uint8	Yes	Selection: Common or cylinder-individual sequence.
* Data type which the RTIO driver uses internally for the parameter			

Tab. 23-22 ES5340-Misfire Device: Configuration Parameters of the "Signals" Tab

23.21.4 Data (ES5340-Misfire Device)

The "Data" tab shows the RTIO signals which are available for controlling the misfire unit.

Note

Please ensure that the "Served Cylinders" parameter (see "Served Cylinders" on page 820) from the upstream ES5340-SigGen Subsystem has been set correctly as this value is used when processing the signals!

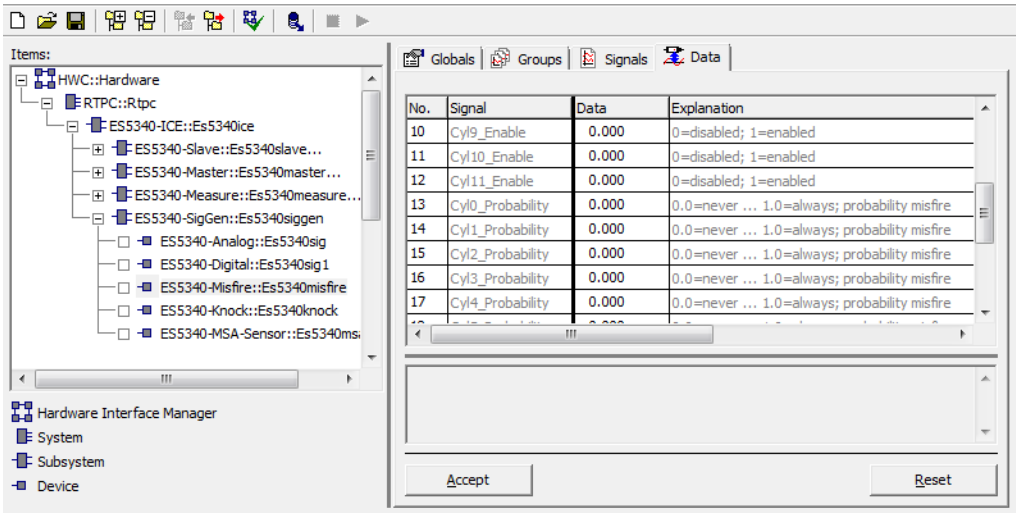


Fig. 23-38 The "Data" Tab of the ES5340-Misfire Device

Cyl0_Enable … Cyl11_Enable

These twelve signals activate the speed modulation for the relevant cylinder.

Cyl0_Probability … Cyl11_Probability

These signals are used to control the probability of the occurrence of misfiring for a cylinder. The signal is only taken into consideration if the relevant cylinder has been configured for using the probability function ("Modulation Mode" on page 829).

Sqnc_Trigger

This signal represents the trigger with which the sequence table play-back is started. A rising edge of 0 → 1 on this signal starts play-back. A sequence which has not run completely is interrupted by a rising edge on the trigger signal and restarted (re-trigger).

The following table summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Cyl0_Enable … Cyl11_Enable	uint8	Enabling/disabling the outputs 0: Disabling the output 1: Enabling the output
Cyl0_Probability … Cyl11_Probability	uint16	Probability of misfiring Value range: 0 to 10000 corresponds to 0 …1.0
Sqnc_Trigger	uint8	Starting or restarting sequence table play-back An edge 0 → 1 is used as trigger
* Data type which the RTIO driver uses internally for the parameter		

Tab. 23-23 ES5340-Misfire Device: The RTIO Signals of the "Data" Tab

23.22 ES5340-Knock Device

The ES5340-Knock device is used to configure and control the knock signal generator on the LABCAR Software.

23.22.1 Globals (ES5340-Knock Device)

The following figure shows the RTIO parameters of the "Globals" tab.

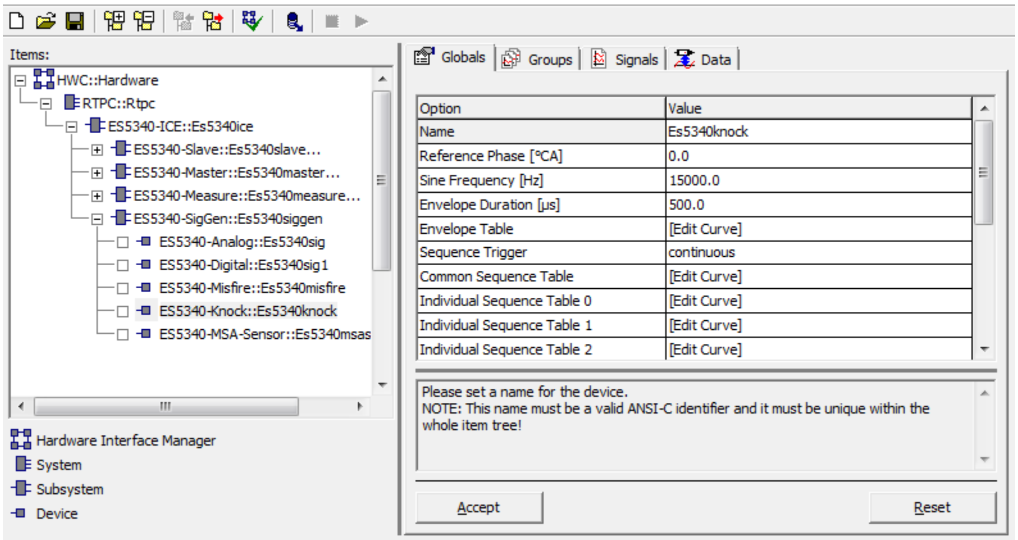


Fig. 23-39 The "Globals" Tab of the ES5340-Knock Device

Reference Phase

This parameter describes the basic phase shift of the knock signal generator. The actual cylinder-specific phase is determined by adding this value to the relevant "Base Angle" parameter from the "Signals" tab (see "Base Angle" on page 835) and the "Cyln_AngleOffset" signal (see "Cyl0_AngleOffset ... Cyl11_AngleOffset" on page 838).

Sine Frequency

This parameter describes the frequency of the sinusoidal signal of which the knock signal consists. This sinusoidal signal is modulated with an envelope curve.

Envelope Duration

This parameter describes the duration of a knock signal in µs.

Envelope Table

The form of the envelope curve is described by this table. The length of the envelope curve is defined by the parameter "Envelope Duration" (see "Knock Signal Generator" on page 817).

Sequence Trigger

With this parameter you can choose between "single shot" and the continuous triggering of the sequence tables.

Common Sequence Table, Individual Sequence Table 0 … 11

There are 12 cylinder-individual sequence tables available for controlling the generation of knock packages. There is also a "Common Sequence Table" which makes it possible to use one sequence on several cylinders.

Cylinder to Sensor Mapping Table 0 … 3

The knock signal generator has four outputs. Each of the four "Cylinder to Sensor Mapping" allocation tables defines which cylinders the generated knock signals are issued from. This makes it possible to output a cylinder on several internal outputs by selecting the relevant cylinder several times in the allocation table.

The tables each consist of up to 12 data points which represent the cylinders 0 to 11. A value of 1.0 at a data point corresponds to the allocation of the relevant cylinder to the relevant output. A value of 0.0 shows that the cylinder is not allocated to the output. Values between 0.0 and 1.0 denote the amplitude of the occurring knocking signal.

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Reference Phase	real32	Yes	Basic phase shift in °CA. Value range: 0 … 720 °CA
Sine Frequency	real32	Yes	Sine frequency of the knock packages in the range 1000 - 20000 Hz
Envelope Duration	real32	Yes	Duration of the envelope curve in µs. Value range: 50-12000 µs
Envelope Table	table	Yes	Envelope curve signal
Sequence Trigger	uint8	Yes	Selection of the trigger mode for the sequence tables. 0: Continuous 1: Single Shot
Common Sequence Table	table	Yes	Sequence table which can be used for all cylinders together
Individual Sequence Table 0 … 11	table	Yes	Cylinder-individual sequence tables
Cylinder to Sensor Mapping Table 0 … 3	table	Yes	Tables in which the relevant cylinders can be allocated to an output. The data points correspond to the 12 cylinders – a value greater than 0.5 means that the relevant cylinder issues the knock signal at the output.

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-24 ES5340-Knock Device: Configuration Parameters of the "Globals" Tab

23.22.2 Groups (ES5340-Knock Device)

The ES5340-Knock device has a signal group in which the signals of the misfire control are processed.

23.22.3 Signals (ES5340-Knock Device)

The knock generator is configured in the "Signals" tab. The following figure shows the RTIO parameters of the "Signals" tab. All parameters can be edited online (i.e. during runtime of the model on the experimental target).

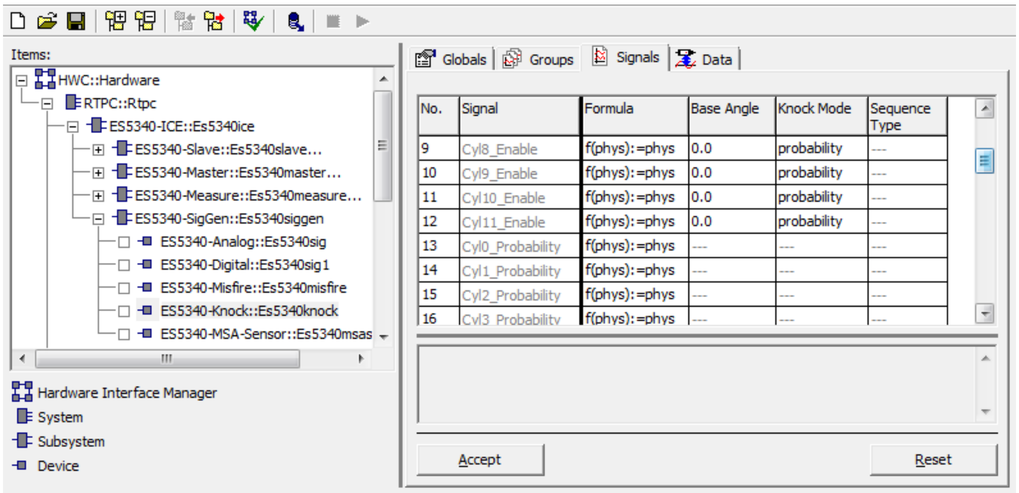


Fig. 23-40 The "Signals" Tab of the ES5340-Knock Device

Base Angle

The option field "Base Angle" is used together with the parameter "Reference Phase" from the "Globals" tab as the base angle for controlling the start angle of a knock package. The cylinder-specific signal "Cylx_AngleOffset" is added to this start angle enabling modification during runtime. Please note that the signal "Cylx_AngleOffset" can only be varied over an angle range of ±127 °CA.

Knock Mode

The occurrence of knock events in a period of 720 °CA (or 360 ° with two-stroke engines) can be controlled either via a probability function or using sequence tables. You can toggle between these two operating modes via this parameter.

Sequence Type

When selecting "Modulation Mode = sequence", you can specify here whether the common sequence table ("common") or the cylinder-individual sequence table ("individual") is to be used.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Base Angle	real32	Yes	Base output angle for knock packages 0.0 … 720.0 °CA
Knock Mode	uint8	Yes	Probability or sequence table for outputting knock packages
Sequence Type	uint8	Yes	Selection: Common or cylinder-individual sequence
* Data type which the RTIO driver uses internally for the parameter			

Tab. 23-25 ES5340-Knock Device: Configuration Parameters of the "Signals" Tab

23.22.4 Data (ES5340-Knock Device)

The existing signal group shows RTIO signals which are available for controlling the knock signal generator.

Note

Please ensure that the "Served Cylinders" parameter in the upstream ES5340-SigGen Subsystem ("Served Cylinders" on page 820) has been set correctly as this value is used when processing the signals!

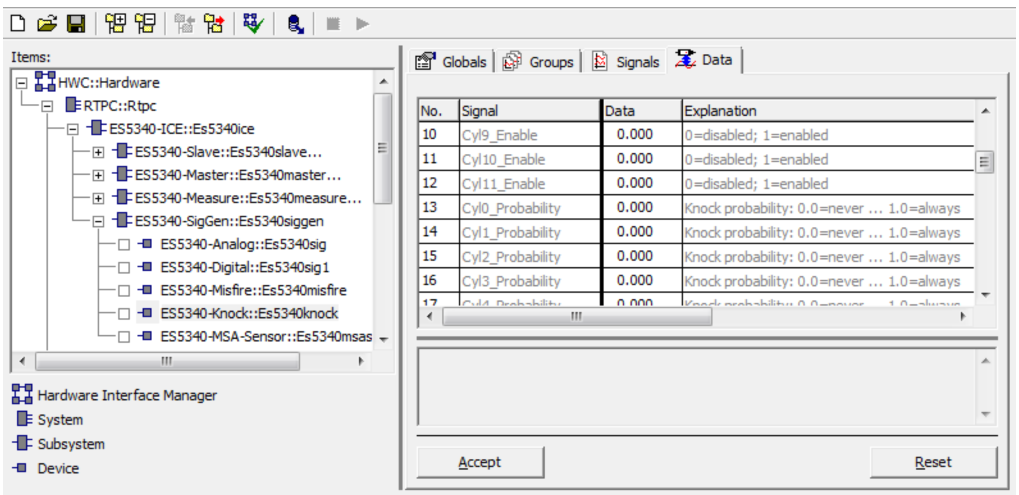


Fig. 23-41 ES5340-Knock Device: Configuration Parameters of the "Data" Tab

Cyl0_Enable … Cyl11_Enable

These twelve signals activate the generation of knock packages for uncontrolled combustion for the relevant cylinders.

Cyl0_Probability … Cyl11_Probability

These signals are used to control the probability of the occurrence of knocking combustion for a cylinder. The signal is only taken into consideration if the relevant cylinder has been configured for using the probability function.

Sqnc_Trigger

This signal represents the trigger with which sequence table play-back can be started. A rising edge of 0 → 1 on this signal starts play-back. A sequence which has not run completely is interrupted by a rising edge on the trigger signal and restarted (re-trigger).

NoiseAmpl

This signal controls a stochastic variation of the amplitudes of the knock packages. There is no variation of the amplitude with a value of 0.0; with a value of 0.1, the amplitude can vary by 10% of the maximum output voltage.

This variation has an effect on the output of all knock packages (knocking and non-knocking).

NoKnockAmpl

Knock packages can still be output in combustion without knocking. The amplitudes of these knock packages are described with this signal. Normally, the amplitude of "NoKnockAmpl" is defined as being smaller than the amplitudes in which knocking occurs.

Cyl0_KnockAmpl … Cyl11_KnockAmpl

The cylinder-specific amplitudes for knocking combustion can be output using these 12 signals.

Cyl0_AngleOffset … Cyl11_AngleOffset

The output angle of the knock packages can be varied with this signal for each individual cylinder by ±127 °CA in intervals of 1 °CA. The total of "Reference Phase" ("Globals" tab) and the relevant "Base Angle" ("Signals" tab) is used as base angle.

RTIO Signal	Data Type*	Comment/Value Range
Cyl0_Enable … Cyl11_Enable	uint8	Control of the knock signals for the individual cylinders 0: Knock packages for knocking possible 1: Knock packages for knocking not possible
Cyl0_Probability ... Cyl11_Probability	uint16	0.0 to 1.0
Sqnc_Trigger	uint8	Starting or restarting sequence table playback An edge 0 → 1 is used as trigger
NoiseAmpl	uint16	0.0 to 1.0
NoKnockAmpl	uint16	0.0 to 1.0
Cyl0_KnockAmpl … Cyl11_KnockAmpl	uint16	0.0 to 1.0
Cyl0_AngleOffset … Cyl11_AngleOffset	sint8	Variation of the knock angle Value range -127 to +127 (resolution 1 °CA)
* Data type which the RTIO driver uses internally for the parameter		

Tab. 23-26 ES5340-Knock Device: The RTIO Signals of the "Data" Tab

23.23 ES5340-MSA-Sensor Device

The ES5340-MSA-Sensor device is used to simulate an MSA crankshaft sensor which detects the direction of rotation.

23.23.1 Basic Functioning

The MSA sensor simulation can use the same waveform traces of the tooth signal as the (Hall) sensors used so far. However, to represent an MSA sensor, the angle position of the center of each tooth of the crankshaft gear must be known. A fixed-length pulse is output for each tooth at the tooth center – in the real sensor, the tooth centers are determined electronically.

In the configuration phase, these angle positions are calculated in accordance with the trace of the waveform and stored in the waveform pool. As the waveform can be selected in real time from one of the 16 predefined waveforms using the "Waveform" signal, all waveforms have to undergo this procedure.

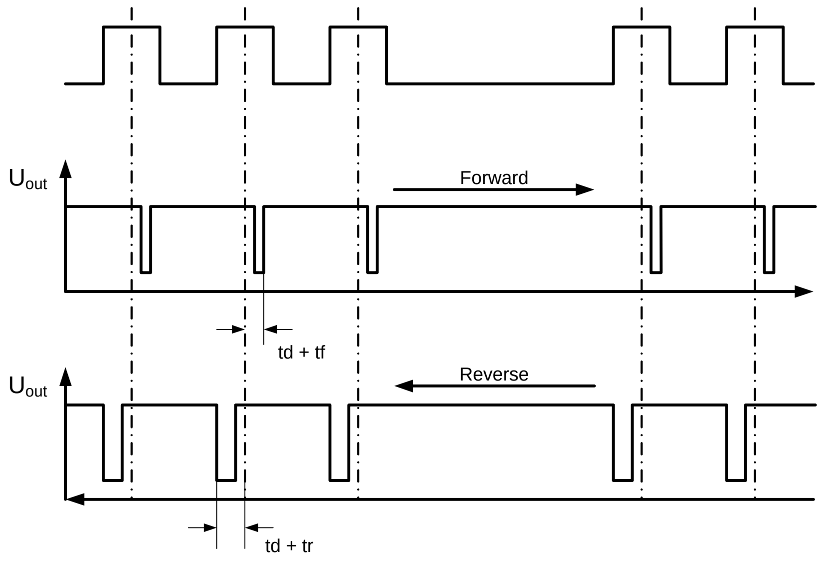


Fig. 23-42 Crankshaft Teeth and Sensor Signals with an MSA Sensor

23.23.2 Tooth Center Calculation

First of all the positions of the negative and positive zero crossings are determined for all waveform traces which have already been normed to a value between -1.0 and 1.0.

In positive logic, a tooth is an area between one positive and the following negative zero crossing (see Fig. 23-43 on page 840). The tooth center is accordingly the arithmetic center of the two positions (and accordingly vice versa with negative logic):

Let np1, np2 be the (positive) zero crossings from bottom to top and nn1, nn2 the (negative) zero crossings from top to bottom. This results in the tooth center positions m1, m2 being m1 = (np1 + nn1) / 2 and m2 = (np2 + nn2) / 2.

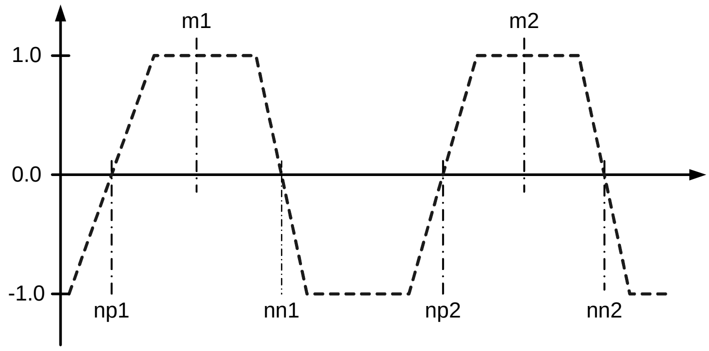


Fig. 23-43 Center Position Calculation (with a not quite symmetrical tooth trace and positive logic)

The logic is set for all curve forms together in the "Globals" tab of the ES5340-SigGen device (see "MSA Sensors: Logic of Input Waveform" on page 821).

23.23.3 Globals (ES5340-MSA-Sensor Device)

The ES5340-MSA-Sensor device is used to configure and address one of the signal generators on the LABCAR Software. The following figure shows the RTIO parameters of the "Globals" tab.

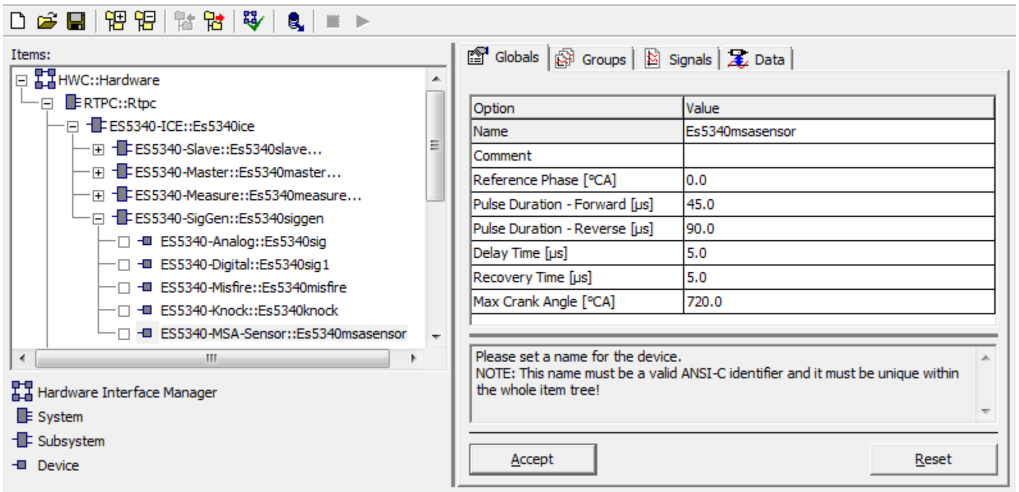


Fig. 23-44 The "Globals" Tab of the ES5340-MSA-Sensor Device

The meaning of the options is described below.

Comment

Any comment for this signal generator.

Note

The comment field does not permit blanks or special characters.

Reference Phase [° CA]

This parameter describes the reference phase shift of the signal generator. The actual phase is determined by adding this value to the signal "PhaseRef".

Pulse Duration - Forward [µs]

This parameter defines the pulse duration of a tooth with a positive direction of rotation (forwards) (tf in Fig. 23-42 on page 839).

Pulse Duration - Reverse [µs]

This parameter defines the pulse duration of a tooth with a negative direction of rotation (backwards) (tr in Fig. 23-42 on page 839).

Delay Time [µs]

This parameter defines the length of the delay time (td in Fig. 23-42 on page 839).

Recovery Time [µs]

This parameter defines the length of a forced pulse (in certain situations when the direction of rotation is reversed).

The following table summarizes the properties of the individual parameters.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
Comment	Identi-fier	No	User-specific comment
Reference Phase	real32	Yes	Ref. phase shift in ° CA, Value range 0…720 °CA
Pulse Duration - Forward	real32	Yes	Pulse duration tf with a forwards rotation, value range 10.0…1000.0 µs
Pulse Duration - Reverse	real32	Yes	Pulse duration tr with a reverse rotation, value range 10.0…1000.0 µs
Delay Time	real32	Yes	Delay time td in acc. with sensor specification value range 1.0…100.0 µs
Recovery Time	real32	Yes	Pulse duration tv in acc. with sensor specification, value range 1.0...100.0 µs

* Data type which the RTIO driver uses internally for the parameter

Tab. 23-27 ES5340-MSA-Sensor: Configuration Parameters of the "Globals" Tab

23.23.4 Groups (ES5340-MSA-Sensor Device)

The ES5340-MSA-Sensor device has one signal group which is used to control the MSA generator.

23.23.5 Data (ES5340-MSA-Sensor Device)

The "Data" tab lists the signals for controlling the signal generator.

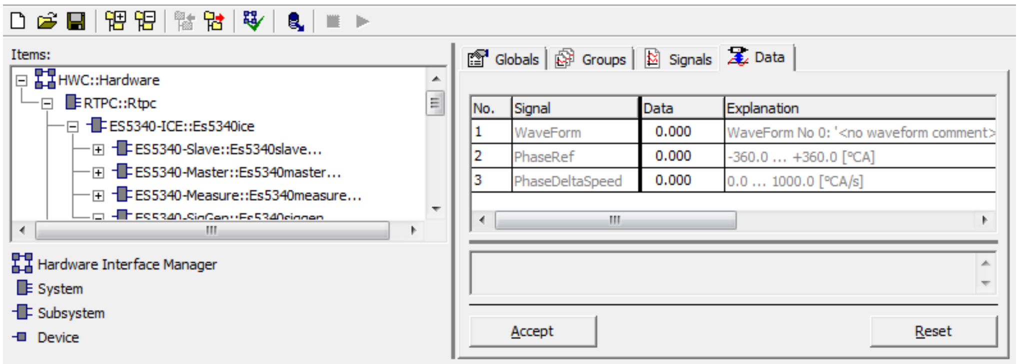


Fig. 23-45 The "Data" Tab of the ES5340-MSA-Sensor Device

The meaning of the signals is described below.

WaveForm

The waveform to be read out and output by the signal generator is selected via the signal "WaveForm". Values from 0 to 15 are possible, corresponding to the central waveform pools in the ES5340-SigGen device.

PhaseRef

This signal describes a phase shift which is added to the global parameter "ReferencePhase". This enables the generation of a model-controlled phase shift during runtime.

Please note that a phase shift is changed with a finite speed (controlled via the "PhaseDeltaSpeed" parameter) which is why the "PhaseDeltaSpeed" signal must have a value > 0 for most applications.

PhaseDeltaSpeed

The change speed of the phase shift is controlled by the signal "PhaseDeltaSpeed" as a change of the value of "PhaseRef" can only take place with finite speed. The value range is from 0 (i.e. no adjustment of the phase) to 1000° CA/s.

Parameter or Option Field	Data Type*	Can be Edited Online	Comment/Value Range
WaveForm	uint8	Yes	Selection of waveform 0-11
PhaseRef	sint16	Yes	Additive phase shift, value range: 32768…32767 (corresponds to -360…+360 [° CA])
PhaseDeltaSpeed	uint16	Yes	Speed of phase change 0…1000 [° CA/s]
* Data type which the RTIO driver uses internally for the parameter			

Tab. 23-28 ES5340-MSA-Sensor: Signals of the "Data" Tab

23.23.6 Troubleshooting

An error message about unsuitable waveform traces is only output when an unsuitable waveform is actually used as the basis of the output of an MSA sensor.

The error message will have the following format:

```
"Waveform <x> not suitable for MSA sensor implemented
by SigGen #<y> because <error> No signal generated"
```

`<x>` represents the number of the unsuitable waveform 0-15 and `#<y>` the number of the signal generator 0…7 used.

The following table contains a description of the error messages `<error>`:

Error Message <error>	Description
…too many zero crossings (> 2048)…	Too many zero crossings were detected to generate a sensible output for an MSA sensor.
…number of positive and negative zero crossings do not match…	A varying number of positive and negative zero crossings was determined. Curve shape is not suitable for representing an MSA sensor.
…too few zero crossings (< 2)…	Too few zero crossings were detected to represent a sensible output for an MSA sensor.
…minimum distance between some tooth centers < 0.4 °CA…	At least two of the calculated tooth center positions are too close together to be processed correctly by the output unit. A minimum distance of around 0.4 °CA is necessary.

Tab. 23-29 List of Possible Error Messages with Unsuitable Waveforms

24 ES5350.1 Analog Board

The ES5350.1 Analog Board is used to generate and measure analog signals. In the RTIO Editor, the ES5350.1 Analog Board is integrated by selecting the ES5350 Subsystem.

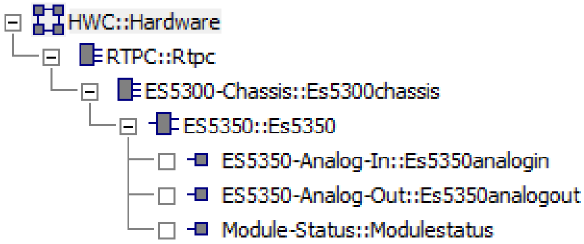


Fig. 24-1 RTIO Hardware Description with ES5350.1 Analog Board

24.1 ES5350 Subsystem

24.1.1 Globals (ES5350 Subsystem)

This section describes the global settings of the ES5350 Subsystem.

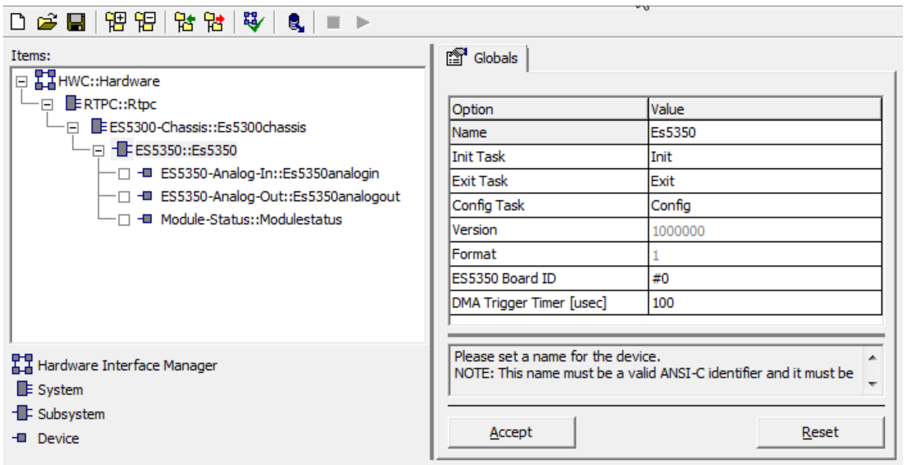


Fig. 24-2 The "Globals" Tab of the ES5350 Subsystem

ES5350 Board ID

In the RTIO Editor up to 16 ES5350.1 boards can be integrated per ES5300 chassis. This option is used to identify the ES5350.1 Analog Board – numbered from left (slot 0) to right.

This RTIO parameter cannot be edited during runtime of the model on the experimental target.

The following option becomes visible if the option **View → Show All** was selected in the editor menu:

DMA Trigger Timer [µsec]

This is where the interval for triggering two asynchronous DMA transmissions can be selected – the current voltages at the ten analog inputs are written directly to the PC main memory.

- Default value: 100 µs
- Setting range: 5 µs … 1 s

24.2 ES5350-Analog-In Device

24.2.1 Globals (ES5350-Analog-In Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

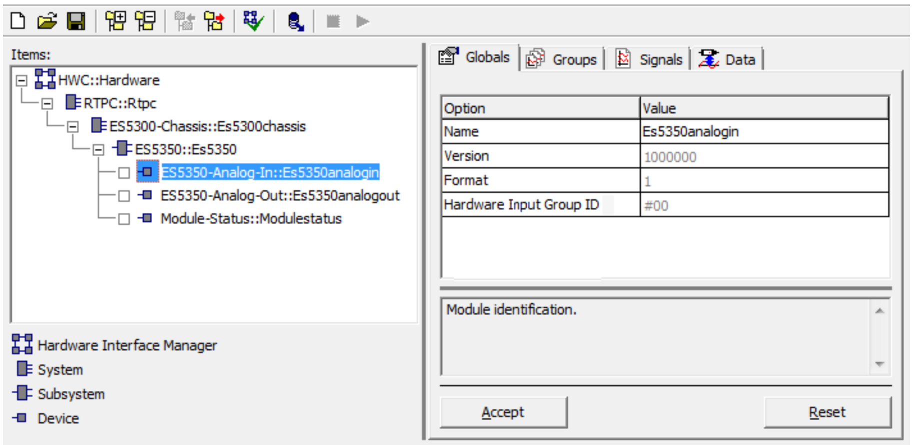


Fig. 24-3 The "Globals" Tab of the ES5350-Analog-In Device

24.2.2 Groups (ES5350-Analog-In Device)

The ES5350-Analog-In Device has one Receive group "AnaIn" for the measure values of the analog inputs.

24.2.3 Signals (ES5350-Analog-In Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

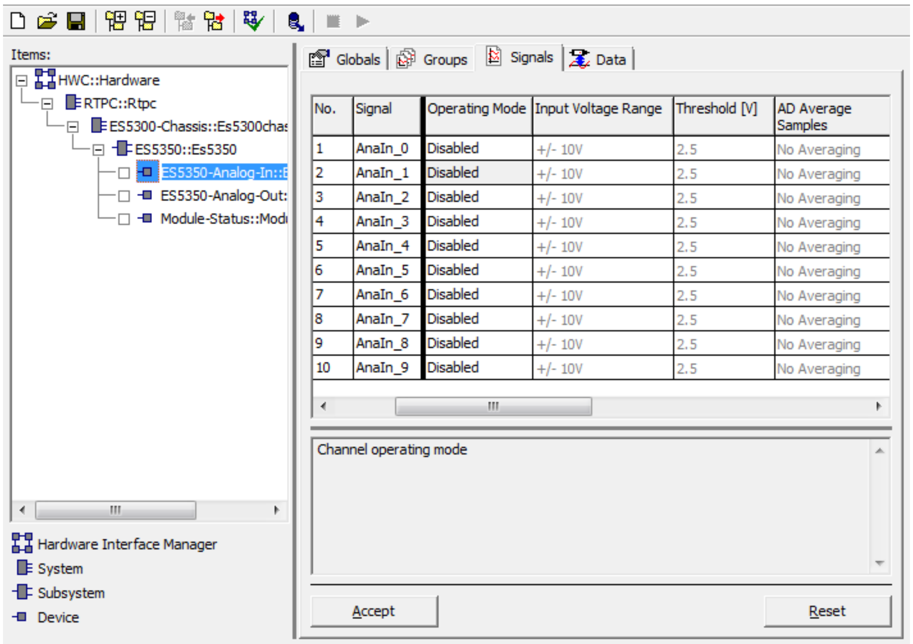


Fig. 24-4 The "Signals" Tab of the ES5350-Analog-In Device

Signal Name	Notes
AnaIn_x (x = 0...9)	Analog voltage at input #x

Tab. 24-1 The Signals of the ES5350-Analog-In Device

The following parameters are available for the signals:

Parameter or Option Field	Can be edited online	Comment/Value Range
Operating Mode	Yes	- **Disabled** Input is disabled - **Analog** Voltage measurement/value range see "Input Voltage Range" - **Comparator** Input works as comparator [0.0, 1.0] with hysteresis (values see "Input Voltage Range")
Input Voltage Range	Yes	Input voltage range (hysteresis) - [-1 V...+1 V] (±0.05 V) - [-10 V...+10 V] (±0.5 V) - [-60 V...+60 V] (±0.5 V)
Threshold [V]	Yes	Voltage threshold of the comparator
AD Average Samples	Yes	Number of samples for averaging

* Data type which the RTIO driver uses internally for the parameter

Tab. 24-2 ES5350-Analog-In Device: Configuration Parameters of the "Signals" Tab

24.3 ES5350-Analog-Out Device

Up to five Analog-Out Devices can be added under one ES5350 Subsystem.

24.3.1 Globals (ES5350-Analog-Out Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

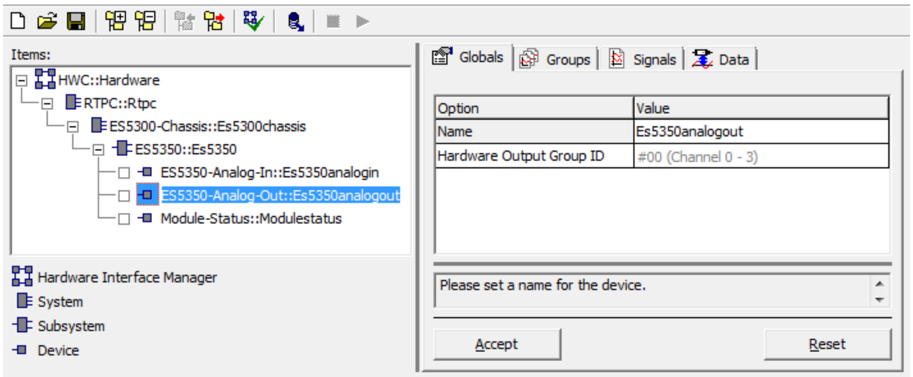


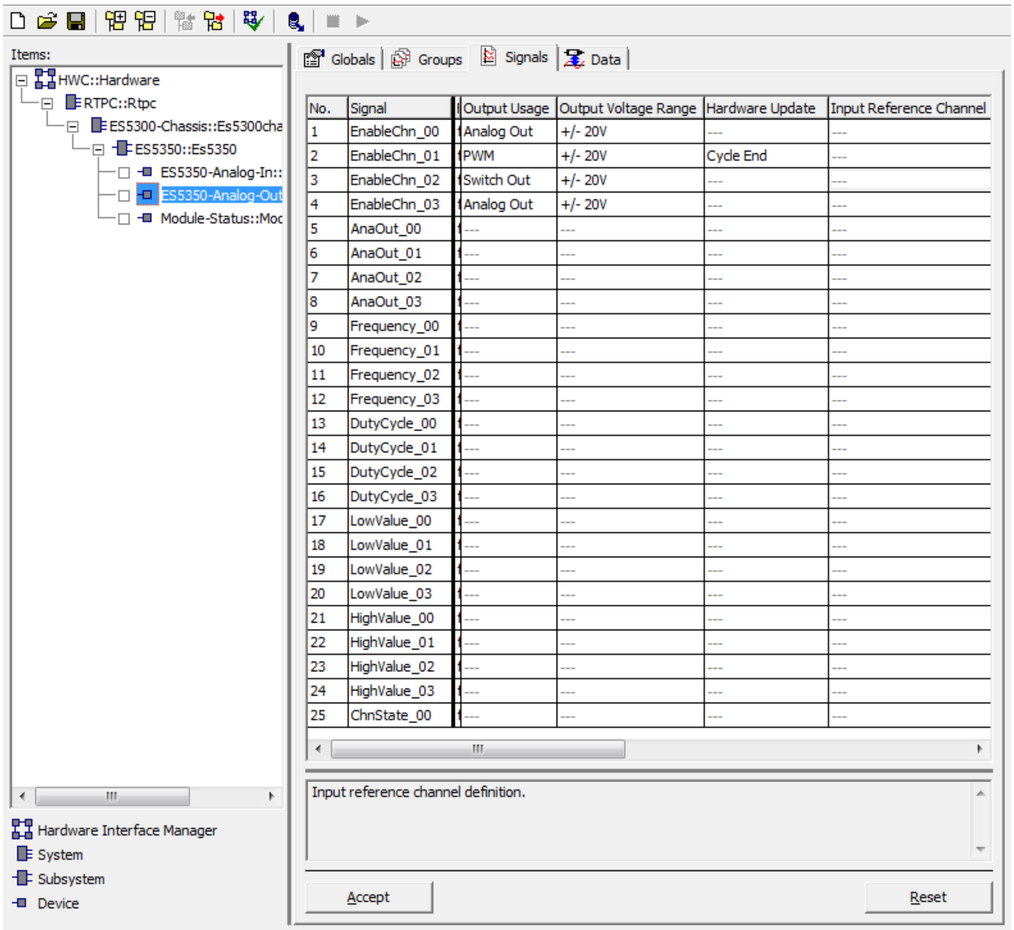
Fig. 24-5 The "Globals" Tab of the ES5350-Analog-Out Device

24.3.2 Groups (ES5350-Analog-Out Device)

The ES5350-Analog-Out Device has one Send group "AnaOut" for the analog output signals and one Receive group "ChnState" for status information.

24.3.3 Signals (ES5350-Analog-Out Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.



No.	Signal	Output Usage	Output Voltage Range	Hardware Update	Input Reference Channel
1	EnableChn_00	Analog Out	+/- 20V	---	---
2	EnableChn_01	PWM	+/- 20V	Cycle End	---
3	EnableChn_02	Switch Out	+/- 20V	---	---
4	EnableChn_03	Analog Out	+/- 20V	---	---
5	AnaOut_00	---	---	---	---
6	AnaOut_01	---	---	---	---
7	AnaOut_02	---	---	---	---
8	AnaOut_03	---	---	---	---
9	Frequency_00	---	---	---	---
10	Frequency_01	---	---	---	---
11	Frequency_02	---	---	---	---
12	Frequency_03	---	---	---	---
13	DutyCycle_00	---	---	---	---
14	DutyCycle_01	---	---	---	---
15	DutyCycle_02	---	---	---	---
16	DutyCycle_03	---	---	---	---
17	LowValue_00	---	---	---	---
18	LowValue_01	---	---	---	---
19	LowValue_02	---	---	---	---
20	LowValue_03	---	---	---	---
21	HighValue_00	---	---	---	---
22	HighValue_01	---	---	---	---
23	HighValue_02	---	---	---	---
24	HighValue_03	---	---	---	---
25	ChnState_00	---	---	---	---

Fig. 24-6 The "Signals" Tab of the ES5350-Analog-Out Device

Signal Name	Notes
EnableChn_xx	0.0 = channel disabled, otherwise channel enabled
AnaOut_xx (x = 00...03)	Depending on the setting for parameter "Output Usage" in Tab. 24-4 on page 851: - Analog Out: output voltage [-20.0..+20.0 V] or (if "Output Voltage Range" = "Input Dep") -1.0..1.0 - PWM: no significance - Switch Out: 0.0 (LowValue_xx)..1.0 (HighValue_xx)
Frequency_xx	PWM frequency Value range: 0.0..1000.0 Hz
DutyCycle_xx	Duty cycle of the PWM signal Value range: 0.0...1.0
LowValue_xx	Switch Out: If ±20 V was selected with "Output Voltage Range" (see Tab. 24-4 on page 851): Low value of the output voltage [V]. If "Output Voltage Range" = "Input Dep": -1.0...1.0 of the input value.
HighValue_xx	Switch Out: If ±20 V was selected with "Output Voltage Range" (see Tab. 24-4 on page 851): High value of the output voltage [V]. If "Output Voltage Range" = "Input Dep": -1.0...1.0 of the input value.
ChnState_xx	Overtemperature: Bits 0 - 3 (channel-selective, Bit 0 corresponds to ch. 0 etc.) 0 = No overtemperature -1 = Overtemperature Overload: Bit 4 (is valid for the entire output group) 0 = No overload -1 = Overload

Tab. 24-3 The Signals of the ES5350-Analog-Out Device

The following parameters are available for the signals:

Parameter or Option Field	Can be edited online	Comment/Value Range
Output Usage	Yes	Operating mode of the channel: - **Analog Out** Channel issues analog signals - **PWM** Channel issues PWM signals - **Switch Out** Output voltage toggles between "LowValue_xx" and "HighValue xx"
Output Voltage Range	Yes	Voltage range: - **[-20...+20 V]** - **Input Dep**endent - requires definition of an "Input Reference Channel" (see below)
Hardware Update	Yes	Defines when (with PWM signals) changes of frequency or duty cycle are accepted. - **Immediate** Changes are accepted immediately. - **Cycle End** Changes are not accepted until after the end of the current period.
Input Reference Channel	Yes	Reference input channel (#0..#9) if the value "Input Dep" was selected with "Output Voltage Range".

* Data type which the RTIO driver uses internally for the parameter

Tab. 24-4 ES5350-Analog-Out Device: Configuration Parameters of the "Signals" Tab

24.4 Module-Status Device

See "Module Status" on page 971.

25 ES5385.1 Carrier Board for Resistor Cascade

The ES5385.1 Carrier Board for Resistor Cascade is used in the LABCAR environment as a resistor board to simulate both temperature and oxygen sensors and is integrated into the ES5300.1 Housing.

25.1 ES5385 Device

25.1.1 Globals (ES5385 Device)

The "Globals" tab of an ES5385 device is shown in Fig. 25-1.

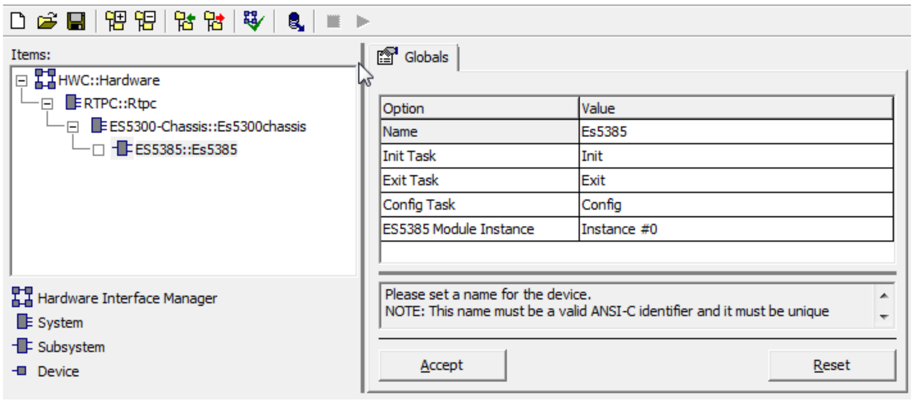


Fig. 25-1 The "Globals" Tab of the ES5385 Device

Name

The C name of the item – used as prefix for all global identifiers.

ES5385 Module Instance

The module instance.

25.2 ES5385-ResistorCascade Device

25.2.1 Globals (ES5385-ResistorCascade Device)

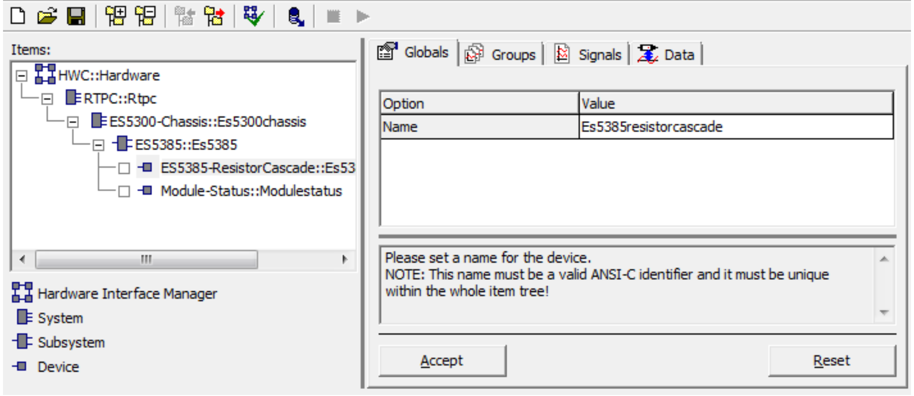


Fig. 25-2 The "Globals" Tab of the ES5385-ResistorCascade Device

Name

The C name of the item – used as prefix for all global identifiers.

25.2.2 Groups (ES5385-ResistorCascade Device)

The ES5385-ResistorCascade device has three signal groups which are transmitted from the experimental target to the ES5385.1 Carrier Board for Resistor Cascade or from the ES5385.1 to the experimental target (see Fig. 25-3 on page 854).

- The **signal group "CasCtrl"** transports the resistance values of the cascades specified by the user and prepared by the RTIO to the ES5385.1. The resistance values specified by the user are accumulated to obtain the resistance value actually possible by taking calibration and monotony into consideration using an approximation method. Then it is transmitted to the hardware.
- The **signal group "CasMntr"** displays the real resistance values created by the RTIO low-level driver for the user. The activation period of this signal group should correspond to the activation period of the signal group "CasCtrl".
- The **signal group "CasStat"** transports current status information of the independent resistor cascades to the RTIO. This means that the individual error states of the hardware can be displayed/monitored via the RTIO.

One task of the real-time operating system must be assigned to each signal group. A task with periodic activation is usually selected. The activation period depends on the dynamic behavior of the resistance values to be generated or the status information to be monitored.

Fig. 25-3 shows the "Groups" tab of an ES5385.1 ResistorCascade device.

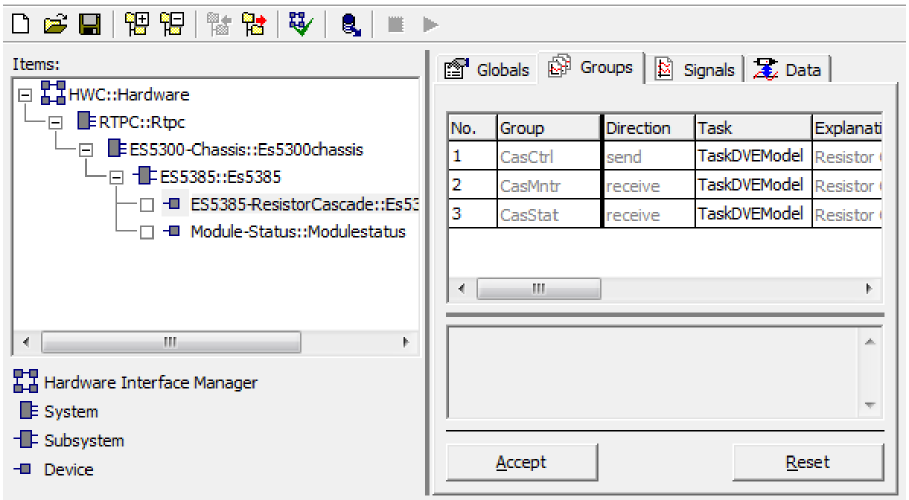


Fig. 25-3 The "Groups" Tab of the ES5385-ResistorCascade Device

RTIO Signals of the "CasCtrl" Signal Group

The "CasCtrl" signal group has 12 RTIO signals, "Ctrl_R_0" to "Ctrl_R_11", which are of the data type "uint32". If the resistor cascade x (x = 0...11) is configured so that it is controlled by the RTIO, the value of the RTIO signal "Ctrl_R_x" defines the target resistance value of the corresponding cascade specified by the user in Ohm.

If the resistor cascade is not controlled by the RTIO, the signal is irrelevant.

RTIO Signal	Data Type*	Comment/Value Range
Ctrl_R_0 to Ctrl_R_11	uint32	Resistance value defined by the user
* Data type which the RTIO driver uses internally for the parameter		

Tab. 25-1 ES5385-ResistorCascade Device: The RTIO Signals of the "CasCtrl" Signal Group

RTIO Signals of the "CasMntr" Signal Group

The "CasMntr" signal group has 12 RTIO signals, "Mntr_R_0" to "Mntr_R_11", which are of the data type "real32". It is used to display the actual resistance value of a cascade transmitted to the hardware. Every resistance value defined by the user must be created using an approximation method taking the calibration values and monotony into consideration.

Tab. 25-2 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Mntr_R_0 to Mntr_R_11	real32	The actual resistance value transmitted to the hardware
* Data type which the RTIO driver uses internally for the parameter		

Tab. 25-2 ES5385-ResistorCascade Device: The RTIO Signals of the "CasMntr" Signal Group

RTIO Signals of the "CasStat" Signal Group

The "CasStat" signal group has 12 RTIO signals, "Stat_C_0" to "Stat_C_11", which are of the data type "uint16". It is used to display the current status of the hardware channels.

Tab. 25-3 summarizes the properties of the RTIO signals.

RTIO Signal	Data Type*	Comment/Value Range
Stat_C_0 to Stat_C_11	uint16	Status of the resistor cascade 0: no error 1: overcurrent was detected
* Data type which the RTIO driver uses internally for the parameter		

Tab. 25-3 ES5385-ResistorCascade Device: The RTIO Signals of the "CasStat" Signal Group

25.2.3 Signals (ES5385-ResistorCascade Device)

In the "Signals" tab, the resistor cascades of an ES5385.1 board are configured.

Fig. 25-4 shows the RTIO parameters of the "Signals" tab. All parameters can be edited during runtime of the RTIO driver on the experimental target.

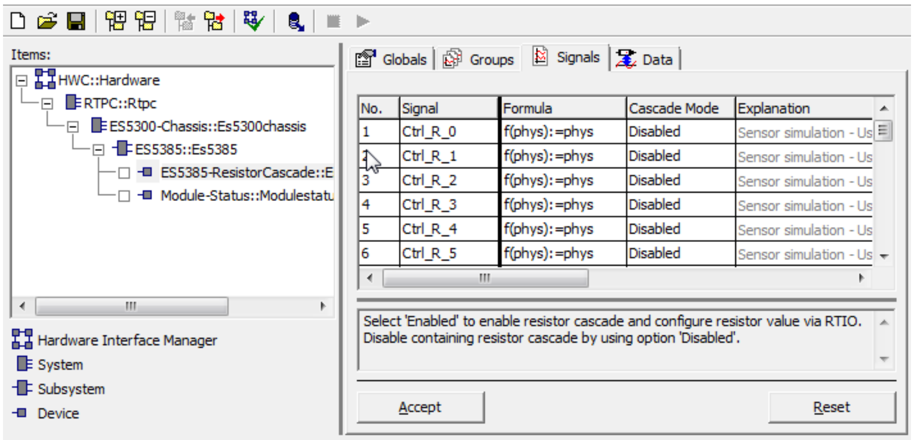


Fig. 25-4 The "Signals" Tab of the ES5385-ResistorCascade Device

Cascade Mode

This option field is used to configure the resistor cascade. "Disabled" deactivates the cascade x (x = 0...11), i.e. all relays are open. No resistance values can be entered and transmitted to the hardware in this state.

"Enabled" activates the cascade x (x = 0...110), i.e. resistance values can only be transported to the hardware via the RTIO in this state.

Tab. 25-4 summarizes the properties of the RTIO signals.

Parameter or Option Field	Data Type*	Can be Edited in Runtime Mode	Comment/Value Range
Cascade Mode	uint8	Yes	Configures the resistor cascades: 0: "Disabled", cascade is deactivated 1: "Enabled", cascade is activated
* Data type which the RTIO driver uses internally for the parameter			

Tab. 25-4 ES5385-ResistorCascade Device: Configuration Parameter of the "Signals" Tab

26 ES5392.1 High Current Switch Board

The ES5392.1 High Current Switch Board is used to control an external power supply and switch battery nodes.

In the RTIO Editor, the ES5392.1 High Current Switch Board is integrated by selecting the ES5392 subsystem.

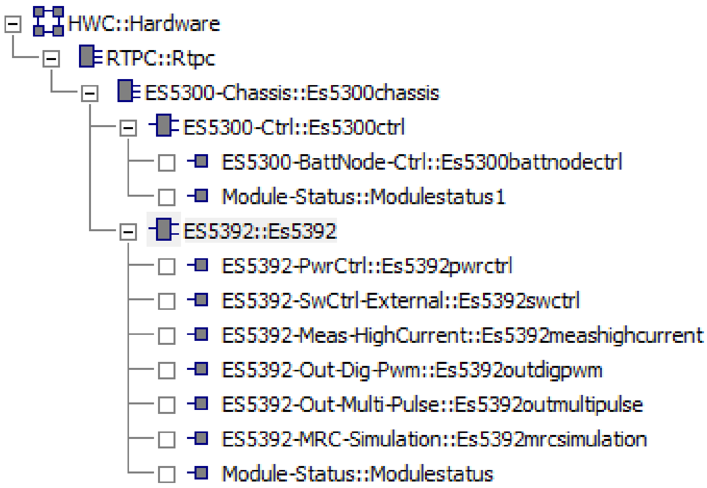


Fig. 26-1 RTIO Hardware Description with ES5392.1 High Current Switch Board

26.1 ES5392 Subsystem

26.1.1 Globals (ES5392 Subsystem)

This section describes the global settings of the ES5392 subsystem.

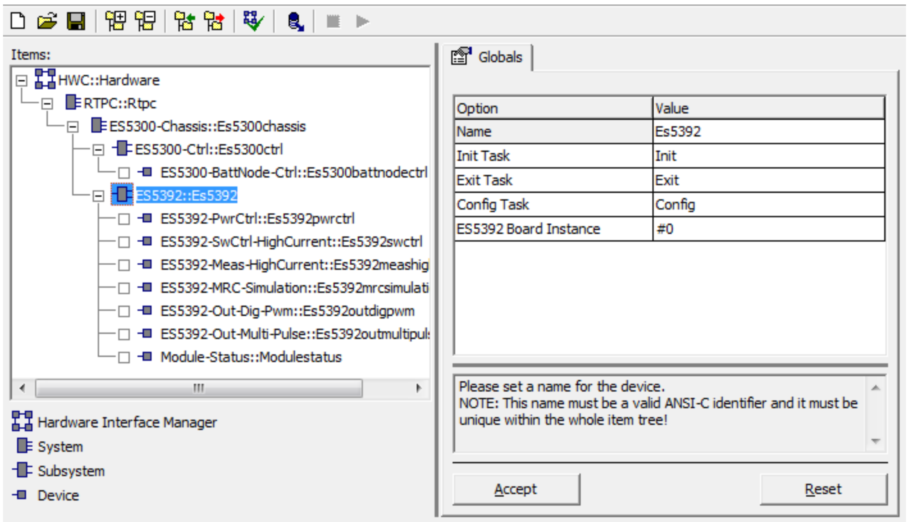


Fig. 26-2 The "Globals" Tab of the ES5392 Subsystem

ES5392 Board Instance

In the RTIO Editor up to four ES5392.1 boards can be integrated per ES5300 chassis. This option is used to identify the ES5392.1 High Current Switch Board – numbered from left (slot 0) to right.

This RTIO parameter cannot be edited during runtime of the model on the experimental target.

26.2 ES5392-PwrCtrl Device

26.2.1 Globals (ES5392-PwrCtrl Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

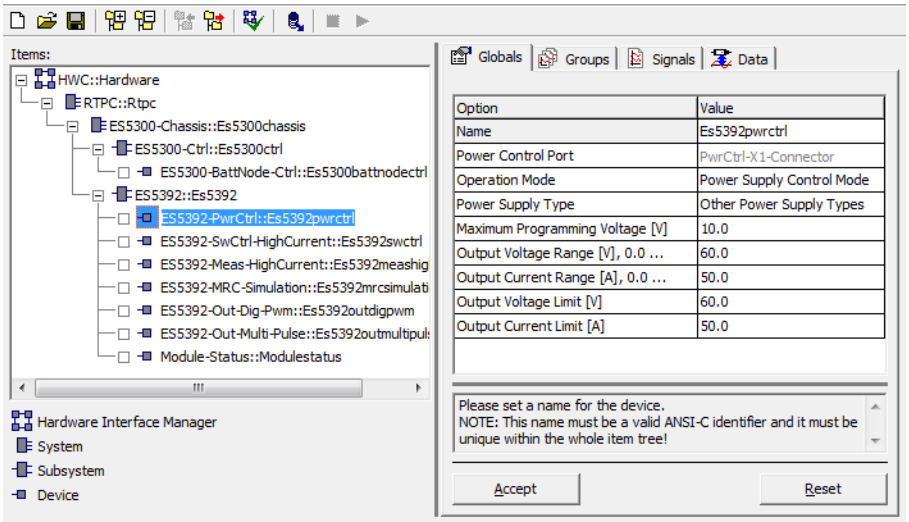


Fig. 26-3 The "Globals" Tab of the ES5392-PwrCtrl Device

Power Control Port

The ES5392.1 High Current Switch Board has a "PwrCtrl" (X1) port on the front panel.

Operation Mode

Operation mode of the power supply control unit:

- **Analog & Digital IO Mode**

  For general digital/analog I/O operations
- **Power Supply Control Mode**

  Control of a power supply control unit which can be selected under "Power Supply Type".

Power Supply Type

Selection of the power supply control unit used.

If you are not using any of the predefined devices, select "Other Power Supply Types". You can then adapt the following parameters accordingly.

Maximum Programming Voltage [V]

Maximum programming voltage in V.

Note

Programming voltages > 10 V cannot be generated with the ES5392.1!

Output Voltage Range [V]

Maximum output voltage in V. This voltage is output when the programming voltage has the maximum value and the power supply control unit is working in "Constant Voltage" (CV) mode.

Output Current Range [A]

Maximum output current in A. This current is output when the programming voltage has the maximum value and the power supply control unit is working in "Constant Current" (CC) mode.

Output Voltage Limit [V]

Limitation of the output voltage - this value is not exceeded even if a higher output voltage was programmed.

Output Current Limit [A]

Limitation of the output current - this value is not exceeded even if a higher output current was programmed.

26.2.2 Groups (ES5392-PwrCtrl Device)

The ES5392-PwrCtrl Device has four groups for each of the analog or digital inputs or outputs.

26.2.3 Signals (ES5392-PwrCtrl Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

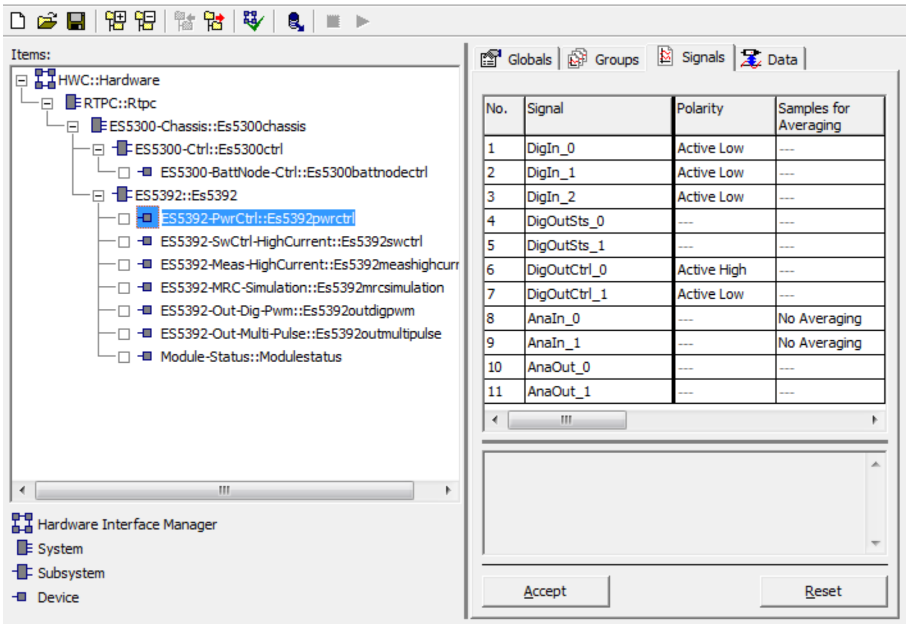


Fig. 26-4 The "Signals" Tab of the ES5392-PwrCtrl Device

Signal Name	Notes
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: Current voltage (X1 Pin 4 / AGND = Pin 3) 1: Current current (X1 Pin 9 / AGND = Pin 14)
AnaOut_x	0: CV Mode: Target voltage, CC Mode: Voltage limit (X1 Pin 5 / AGND = Pin 13) 1: CC Mode: Target current, CV Mode: Current limit (X1 Pin 10 / AGND = Pin 15)

Tab. 26-1 The Signals of the ES5392-PwrCtrl Device

The following parameters are available for the signals:

Parameter or Option Field	Can be edited online	Comment/Value Range
Polarity	Yes	Polarity of the signals
Samples for Averaging	Yes	Number of samples for averaging

Tab. 26-2 ES5392-PwrCtrl Device: Configuration Parameters of the "Signals" Tab

26.3 ES5392-SwCtrl-External Device

26.3.1 Globals (ES5392-SwCtrl-External Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

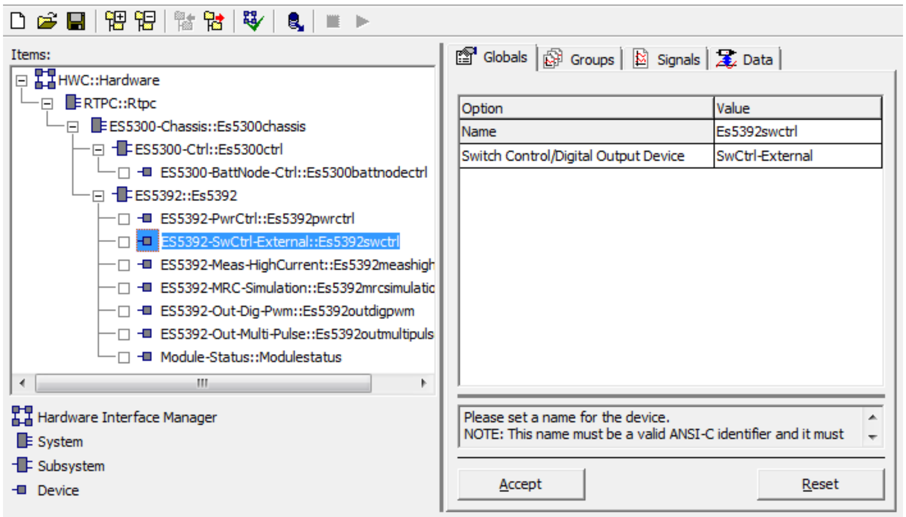


Fig. 26-5 The "Globals" Tab of the ES5392-SwCtrl-External Device

Switch Control/Digital Output Device

Specification of the output (Switch Control/Digital Output).

The following configurations are possible:

- **SwCtrl-External**

  Signals are made available at the "SwrCtrl" (X2) port (see "Signals (ES5392-SwCtrl-External Device)" on page 862).

- **SwCtrl-DigOut**

  Further signals are made available at the X4 port (see "Signals (ES5392-SwCtrl-DigOut)" on page 863).

- **SwCtrl-HighCurrent**

  For controlling the high-current switches – the switched battery voltages are made available at the X4 port
  (see "Signals (ES5392-SwCtrl-HighCurrent)" on page 864).

26.3.2 Groups (ES5392-SwCtrl-External Device)

The ES5392-SwCtrl-External Device has two groups for control and status signals.

26.3.3 Signals (ES5392-SwCtrl-External Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

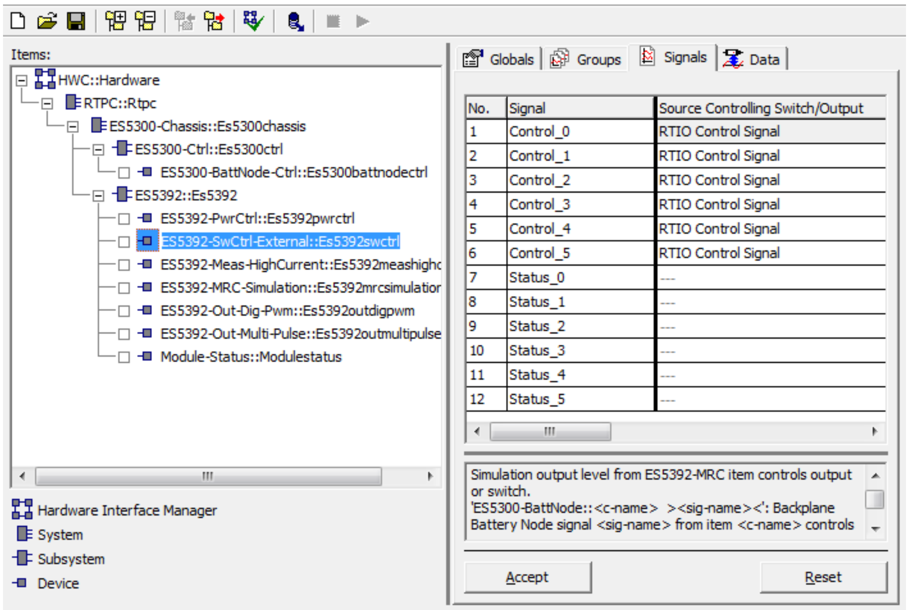


Fig. 26-6 The "Signals" Tab of the ES5392-SwCtrl-External Device

Signal Name	Notes
Control_x (x = 0...5)	Digital output 0: 0 V 1: 5 V
Status_x (x = 0...5)	Status of the switch or output #x 0: No error 1: Error

Tab. 26-3 The Signals of the ES5392-SwCtrl-External Device

The following parameters are available for controlling the outputs:

Source Controlling Switch/Output

- **RTIO Control Signal**

  Output/switch is controlled by the RTIO signal of this channel.

- **ES5392-MRC-Simulation::<c-name> ><sig-name><**

  Output/switch is controlled by the output level of the main relay simulation (see "Globals (ES5392-MRC-Simulation Device)" on page 868).

- **ES5300-BattNode::<c-name> ><sig-name><**

  Output/switch is controlled by the battery node signal <sig-name> of the item <c-name> (see "ES5300-BattNode-Ctrl Device" on page 605).

- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=High/On**
  Output/switch is controlled by the PWM channel <x> (positive polarity).
- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=Low/Off**
  Output/switch is controlled by the PWM channel <x> (negative polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=High/On**
  Output/switch is controlled by the multi-pulse channel <x> (positive polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=Low/Off**
  Output/switch is controlled by the multi-pulse channel <x> (negative polarity).

26.3.4 Signals (ES5392-SwCtrl-DigOut)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

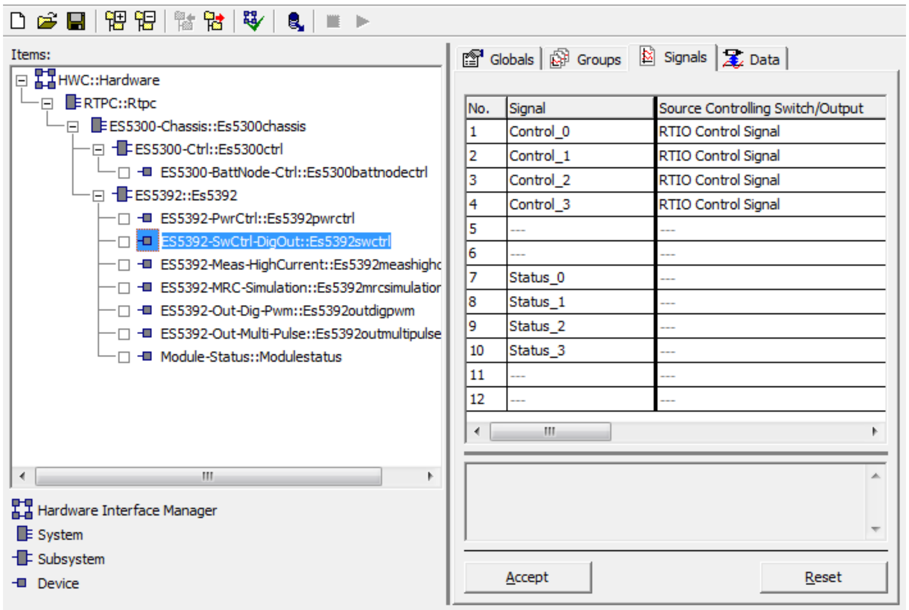


Fig. 26-7 The "Signals" Tab of the ES5392-SwCtrl-DigOut Device

Signal Name	Notes
Control_x (x = 0...3)	Digital output 0: 0 V 1: 5 V
Status_x (x = 0...3)	Status of the switch or output #x 0: No error 1: Error

Tab. 26-4 The Signals of the ES5392-SwCtrl-DigOut Device

The following parameters are available for controlling the outputs:

Source Controlling Switch/Output

- **RTIO Control Signal**

  Output/switch is controlled by the RTIO signal of this channel.
- **ES5392-MRC-Simulation::<c-name> ><sig-name><**

  Output/switch is controlled by the output level of the main relay simulation (see "Globals (ES5392-MRC-Simulation Device)" on page 868).
- **ES5300-BattNode::<c-name> ><sig-name><**

  Output/switch is controlled by the battery node signal <sig-name> of the item <c-name> (see "ES5300-BattNode-Ctrl Device" on page 605).
- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=High/On**

  Output/switch is controlled by the PWM channel <x> (positive polarity).
- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=Low/Off**

  Output/switch is controlled by the PWM channel <x> (negative polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=High/On**

  Output/switch is controlled by the multi-pulse channel <x> (positive polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=Low/Off**

  Output/switch is controlled by the multi-pulse channel <x> (negative polarity).

26.3.5 Signals (ES5392-SwCtrl-HighCurrent)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

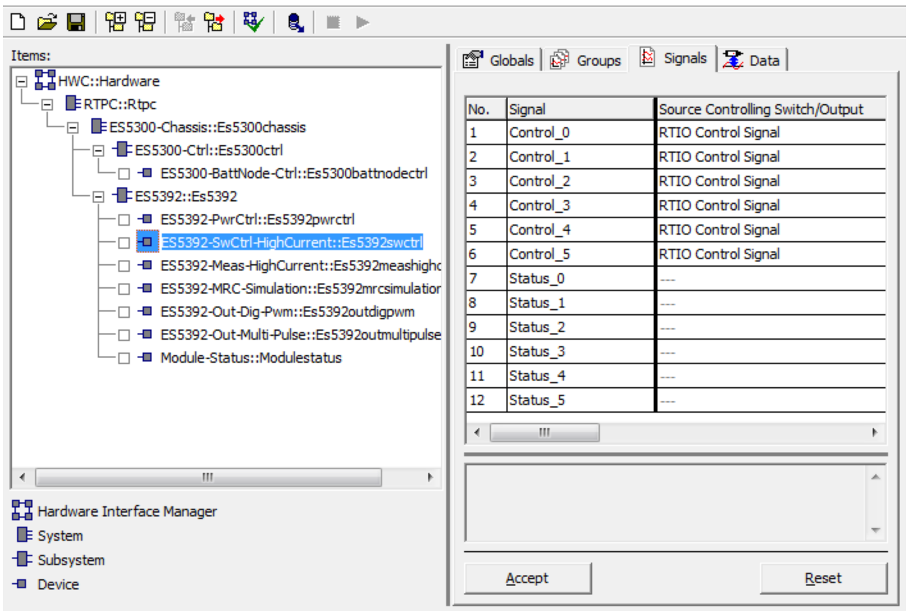


Fig. 26-8 The "Signals" Tab of the ES5392-SwCtrl-DigOut Device

Signal Name	Notes
Control_x (x = 0...5)	Digital output 0: 0 V 1: 5 V
Status_x (x = 0...5)	Status of the switch or output #x 0: No error 1: Error

Tab. 26-5 The Signals of the ES5392-SwCtrl-DigOut Device

The following parameters are available for controlling the outputs:

Source Controlling Switch/Output

- **RTIO Control Signal**

  Output/switch is controlled by the RTIO signal of this channel.
- **ES5392-MRC-Simulation::<c-name> ><sig-name><**

  Output/switch is controlled by the output level of the main relay simulation (see "Globals (ES5392-MRC-Simulation Device)" on page 868).
- **ES5300-BattNode::<c-name> ><sig-name><**

  Output/switch is controlled by the battery node signal <sig-name> of the item <c-name> (see "ES5300-BattNode-Ctrl Device" on page 605).
- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=High/On**

  Output/switch is controlled by the PWM channel <x> (positive polarity).
- **ES5392-Out-Dig-Pwm::<c-name> Channel #<x>: Active=Low/Off**

  Output/switch is controlled by the PWM channel <x> (negative polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=High/On**

  Output/switch is controlled by the multi-pulse channel <x> (positive polarity).
- **ES592-Out-Multi-Pulse::<c-name>: Active=Low/Off**

  Output/switch is controlled by the multi-pulse channel <x> (negative polarity).

Cut-Off Limit [A]

The value of the trip current of the self-healing fuse.

Once the temperature and current have regained acceptable values, the fuse switches back on again.

26.4 ES5392-Meas-HighCurrent Device

26.4.1 Globals (ES5392-Meas-HighCurrent Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

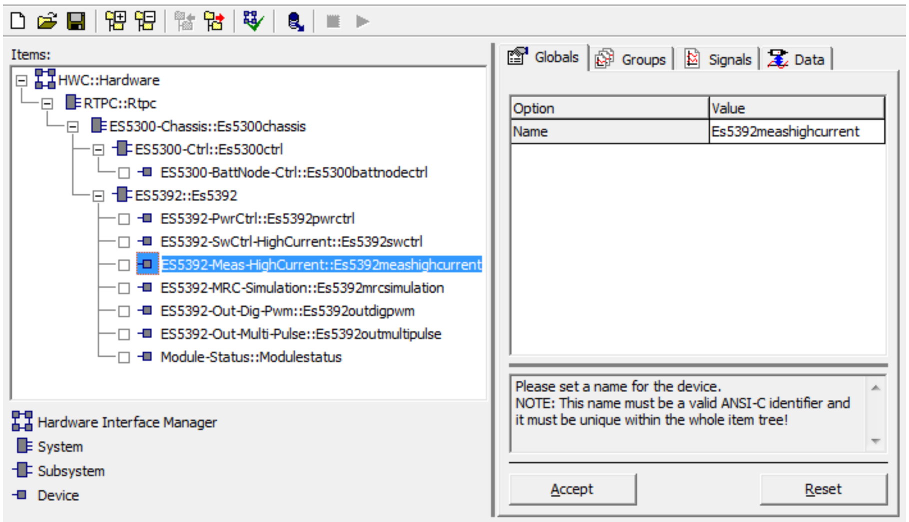


Fig. 26-9 The "Globals" Tab of the ES5392-Meas-HighCurrent Device

26.4.2 Groups (ES5392-Meas-HighCurrent Device)

The ES5392-Meas-HighCurrent Device has one Receive signal group "CurrentIn".

26.4.3 Signals (ES5392-Meas-HighCurrent Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

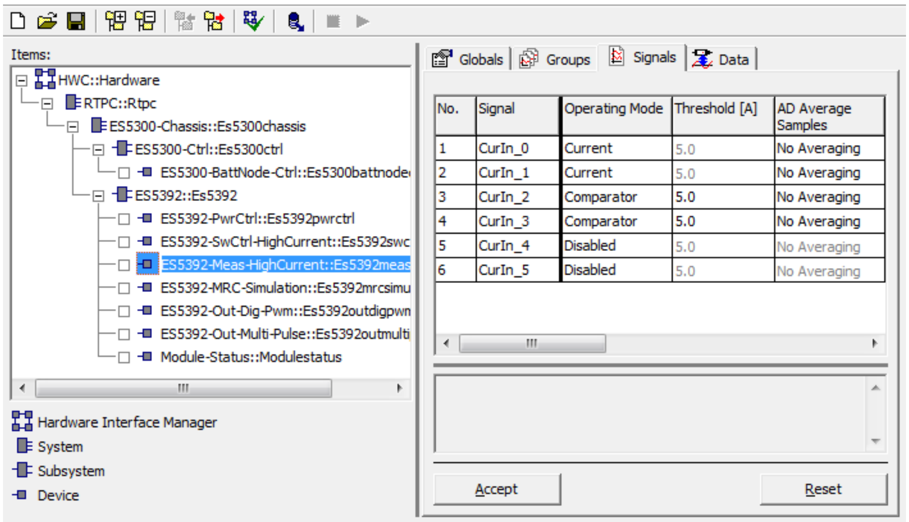


Fig. 26-10 The "Signals" Tab of the ES5392-Meas-HighCurrent Device

Signal Name	Notes
CurIn_x	Configuration of current measuring for channel x (x = 0...5) Value range: -15.0 A...+15.0 A

Tab. 26-6 The Signals of the ES5392-Meas-HighCurrent Device

The following parameters are available for the signals:

Parameter or Option Field	Can be edited online	Comment/Value Range
Operating Mode	Yes	- Disabled: CurIn_x =: 0.0 - Current: CurIn_x contains the measured current - Comparator: CurIn_x = 1.0 (true), if the threshold set was exceeded
Threshold [A]	Yes	Comparator threshold If the measured current is equal to or greater than this value, the signal is set to "true" (1.0)
AD Average Samples	Yes	Number of samples for averaging

Tab. 26-7 ES5392-Meas-HighCurrent Device: Configuration Parameters of the "Signals" Tab

26.5 ES5392-MRC-Simulation Device

26.5.1 Globals (ES5392-MRC-Simulation Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

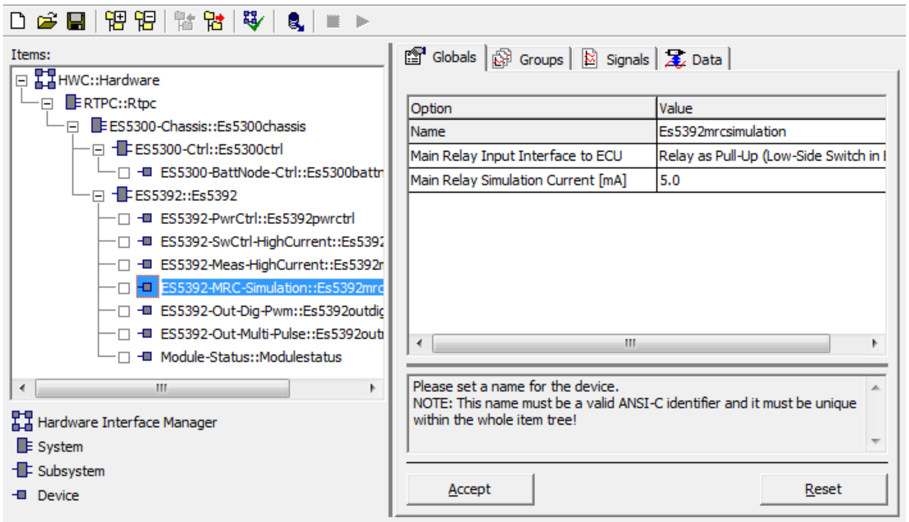


Fig. 26-11 The "Globals" Tab of the ES5392-MRC-Simulation Device

Main Relay Input Interface to ECU

Enables the simulation of the "Main Relay" for the ECU.

Possible settings:

- Disabled
- Relay as Pull-Up (to +UBatt)
- Relay as Pull-Down (to GND)

Main Relay Simulation Current [mA]

Current (in mA) of the current source

26.5.2 Groups (ES5392-MRC-Simulation Device)

The ES5392-MRC-Simulation Device has one Receive signal group "Status" for level and error information.

26.5.3 Signals (ES5392-MRC-Simulation Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

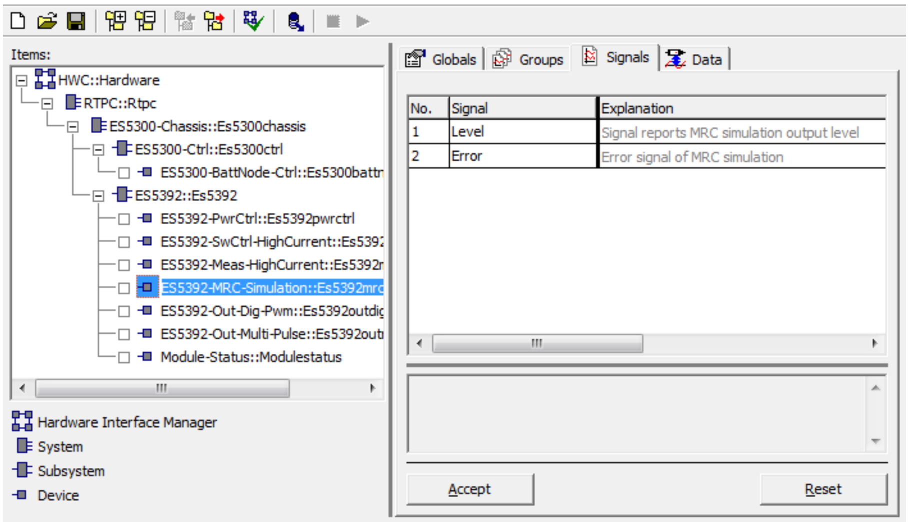


Fig. 26-12 The "Signals" Tab of the ES5392-MRC-Simulation Device

Signal Name	Notes
Level	Status of the output of the MRC simulation: 0 = Off 1 = On
Error	Error status of the MRC output 0 = Ok 1 = Error

Tab. 26-8 The Signals of the ES5392-MRC-Simulation Device

26.6 ES5392-Out-Dig-Pwm Device

26.6.1 Globals (ES5392-Out-Dig-Pwm Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

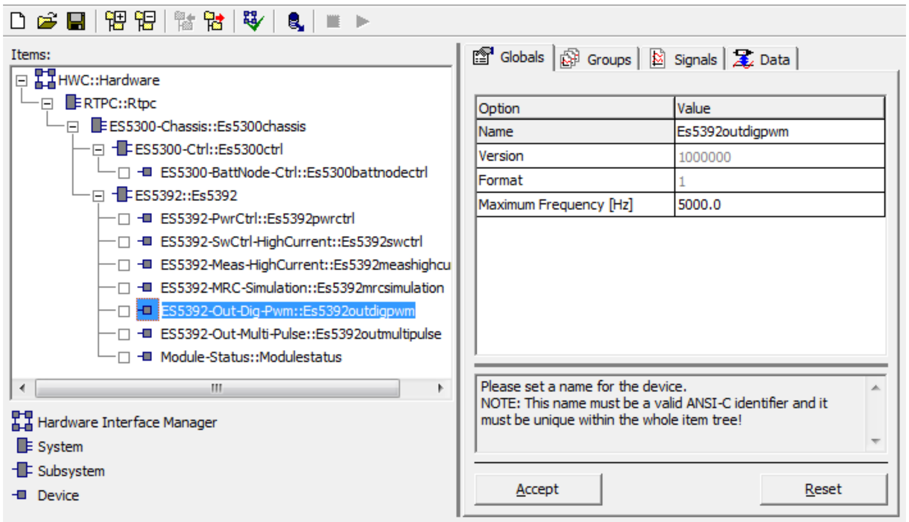


Fig. 26-13 The "Globals" Tab of the ES5392-Out-Dig-Pwm Device

The following option is visible when the option **View → Show All** was selected in the editor menu:

Maximum Frequency [Hz]

Maximum settable frequency (in Hz) for all PWM units of this device.

26.6.2 Groups (ES5392-Out-Dig-Pwm Device)

The ES5392-Out-Dig-Pwm Device has one Send group "Control" for the control signals of the PWM channels.

26.6.3 Signals (ES5392-Out-Dig-Pwm Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

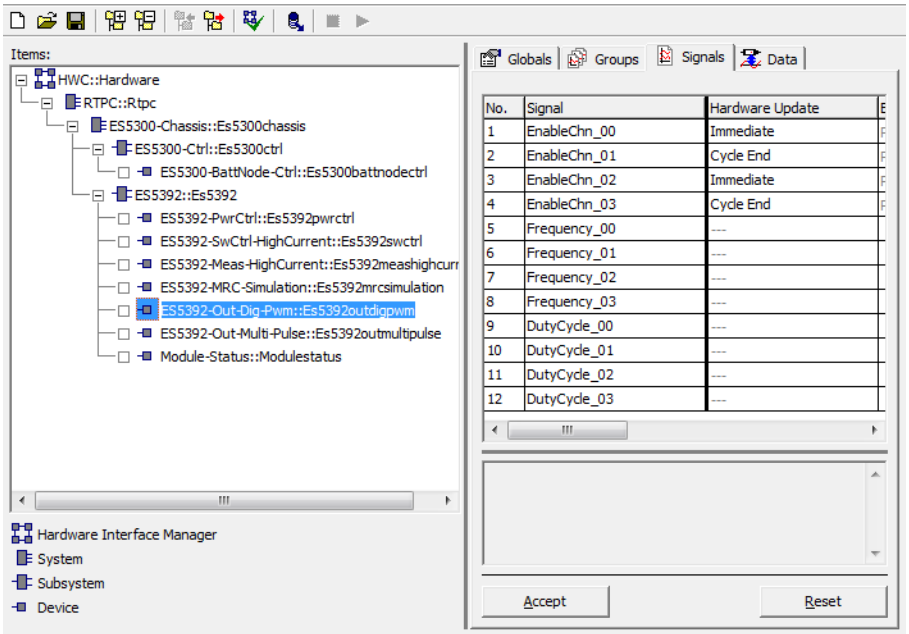


Fig. 26-14 The "Signals" Tab of the ES5392-Out-Dig-Pwm Device

Signal Name	Notes
EnableChn_xx	Enabling of the PWM channel 'xx': 0 = Disabled 1 = Enabled
Frequency	Frequency of the PWM signal in Hz, Value range: 4.547×10^{-5}...5000.0
DutyCycle	Duty cycle of the PWM signal. Value range: 0.0...1.0

Tab. 26-9 The Signals of the ES5392-Out-Dig-Pwm Device

Hardware Update

For the "EnableChn" signal, there is the option "Hardware Update" which is used to define when changes of frequency or duty cycle are accepted.

- **Immediate**

  Changes are accepted immediately.
- **Cycle End**

  Changes are not accepted until the end of the current cycle.

26.7 ES5392-Out-Multi-Pulse Device

26.7.1 Globals (ES5392-Out-Multi-Pulse Device)

The following figure shows the RTIO configuration parameters of the "Globals" tab.

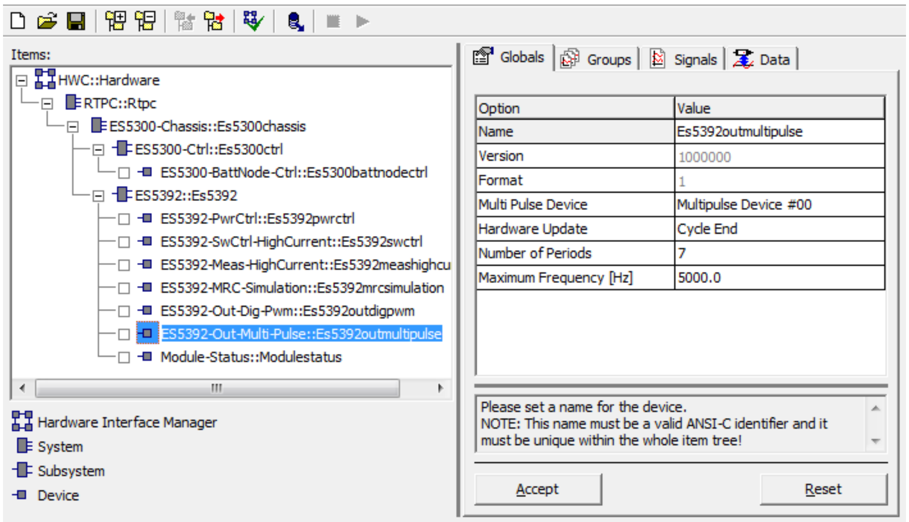


Fig. 26-15 The "Globals" Tab of the ES5392-Out-Multi-Pulse Device

Multi Pulse Device

Defines the output channel for the multi-pulse signal (0...3).

Hardware Update

This defines when changes of frequency or duty cycle are accepted:

- **RTIO Controlled**

  Changes are accepted at the rising edges of the "SyncSgl" signal.
- **Immediate**

  Changes are accepted immediately.
- **Cycle End**

  Changes are not accepted until the end of the current cycle.

Number of Periods

Number of periods (1...8, each defined by frequency and duty cycle) of a multi-pulse cycle.

Maximum Frequency [Hz]

Maximum settable frequency (in Hz) for all PWM units of this device.

Only visible if the option **View → Show All** was selected in the editor menu.

26.7.2 Groups (ES5392-Out-Multi-Pulse Device)

The ES5392-Out-Multi-Pulse Device has one Send group "Control" for controlling the multi-pulse outputs.

26.7.3 Signals (ES5392-Out-Multi-Pulse Device)

The following figure shows the RTIO configuration parameters of the "Signals" tab.

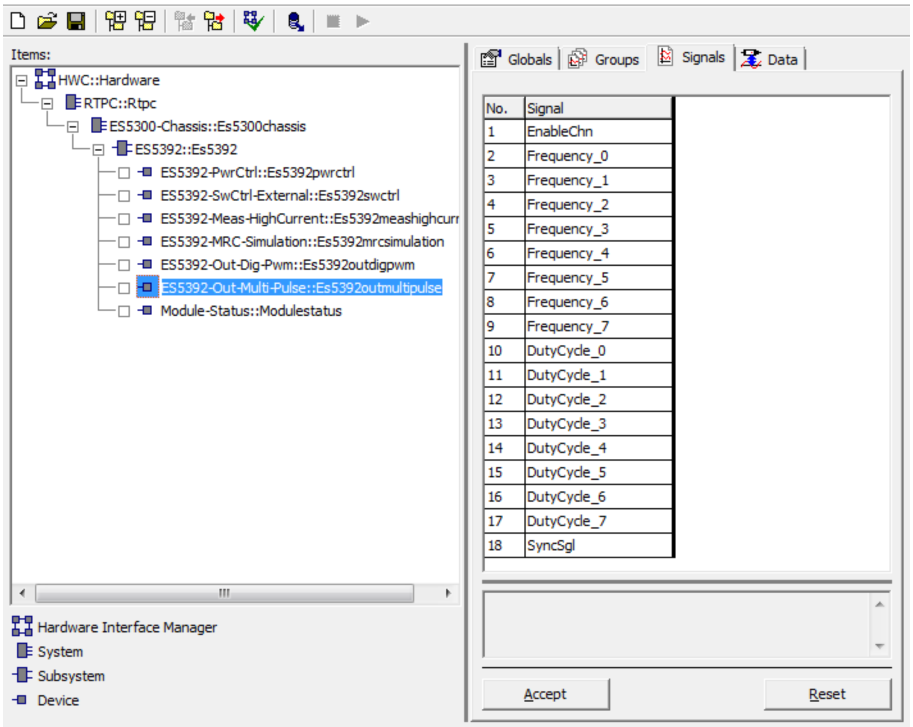


Fig. 26-16 The "Signals" Tab of the ES5392-Out-Multi-Pulse Device

Signal Name	Notes
EnableChn	Enabling of the multi-pulse channel: 0 = Disabled 1 = Enabled
Frequency_x [x = 0...7]	Frequency of the signal in Hz Value range: 1.164×10^{-2}....5000.0
DutyCycle_x [x = 0...7]	Duty cycle of the PWM signal Value range: 0.0...1.0
SyncSgl	Synchronization signal for RTIO-controlled triggers with 0 → 1 transition (see "Hardware Update" on page 872).

26.8 Module Status Device

See "Module Status" on page 971.

27 IXXAT iPCI-I XC16/PCI CAN Interface Board

This document describes the RTIO integration of the PCI CAN board "iPC-I XC16/PCI" made by the company IXXAT for the ETAS RTPC. The board is referred to as the "IXXAT-XC16-CAN" in the RTIO and in this documentation.

The board has two CAN controllers which can be used independently and which - when equipped accordingly - can be switched between "high-speed" and "low-speed" CAN mode via software.

Structure of the IXXAT-XC16-CAN RTIO Tree

An IXXAT-XC16-CAN Board is inserted into a free PCI slot of an real-time PC and connected directly to the "RTPC"experimental target there via the PCI bus. The hardware connection described is reflected in the RTIO tree (Fig. 27-1). The "IXXAT-XC16-CAN" RTIO element of an IXXAT-XC16-CAN Board is assigned directly to the "RTPC" experimental target.

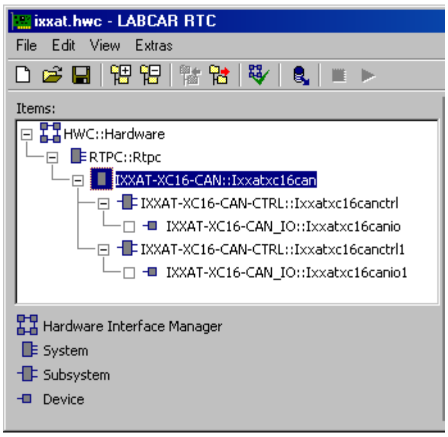


Fig. 27-1 RTIO Hardware Description with Integrated IXXAT-XC16-CAN Board

Note

Fig. 27-1 shows the connection of the IXXAT-XC16-CAN to an RTPC experimental target; connection to an ES1130 experimental target is not possible.

27.1 IXXAT-XC16-CAN Subsystem

27.1.1 Globals (IXXAT-XC16-CAN Subsystem)

Fig. 27-2 shows the "Globals" tab of an IXXAT-XC16-CAN subsystem.

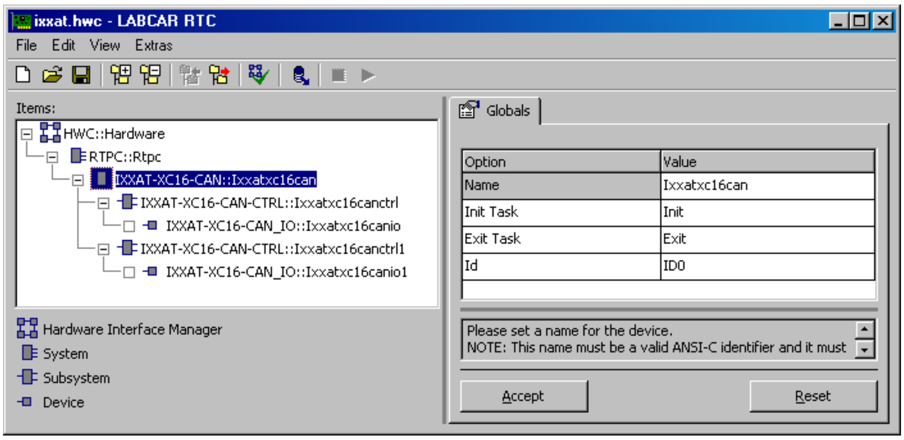


Fig. 27-2 The "Globals" Tab of the IXXAT-XC16-CAN Subsystem

Id

This option field is used to identify the IXXAT-XC16-CAN Board. It establishes the assignment between the RTIO hardware description and the IXXAT-XC16-CAN Board in the PCI bus for which this description is valid. The numbering of the IXXAT-XC16-CAN Boards in the real-time PC depends on the PC used and is not standard; the numbering starts with 0. Up to four IXXAT-XC16-CAN Boards can be integrated per Lreal-time PC.

This RTIO parameter cannot be set during runtime.

27.1.2 Hidden Option Fields

Further option fields (Fig. 27-3) can be made visible by clicking the "Globals" tab with the right-hand mouse button and selecting "Show all Options" from the shortcut menu. These option fields make it possible for ETAS service personnel to gain additional information on the IXXAT-XC16-CAN Board when localizing the cause of an error. They are not intended for the user and the default settings should therefore not be changed by the user.

Fig. 27-3 shows the "Globals" tab of an IXXAT-XC16-CAN subsystem with hidden options.

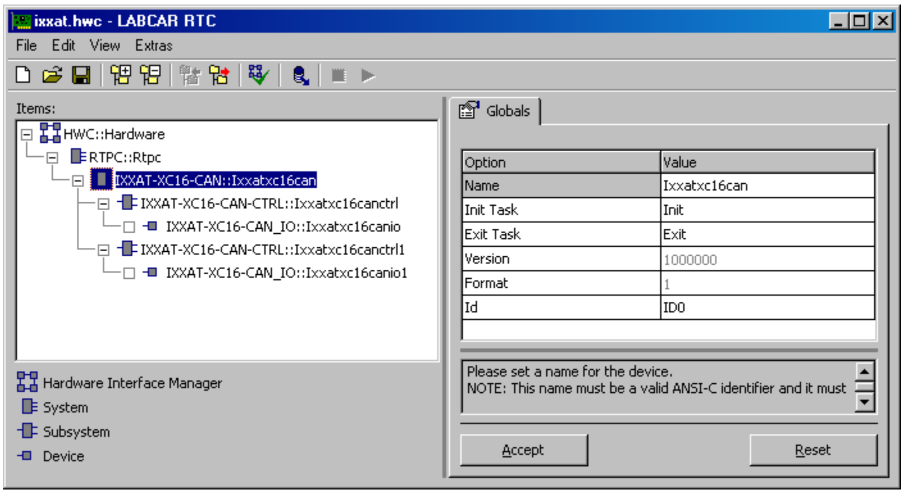


Fig. 27-3 The "Globals" Tab of the IXXAT-XC16-CAN Subsystem with Hidden Options

27.2 IXXAT-XC16-CAN-CTRL Subsystem

27.2.1 Globals (IXXAT-XC16-CAN-CTRL Subsystem)

Note

Changes to the settings of the IXXAT-XC16-CAN Board are not possible in online mode.

Fig. 27-4 shows the "Globals" tab of an IXXAT-XC16-CAN-CTRL subsystem.

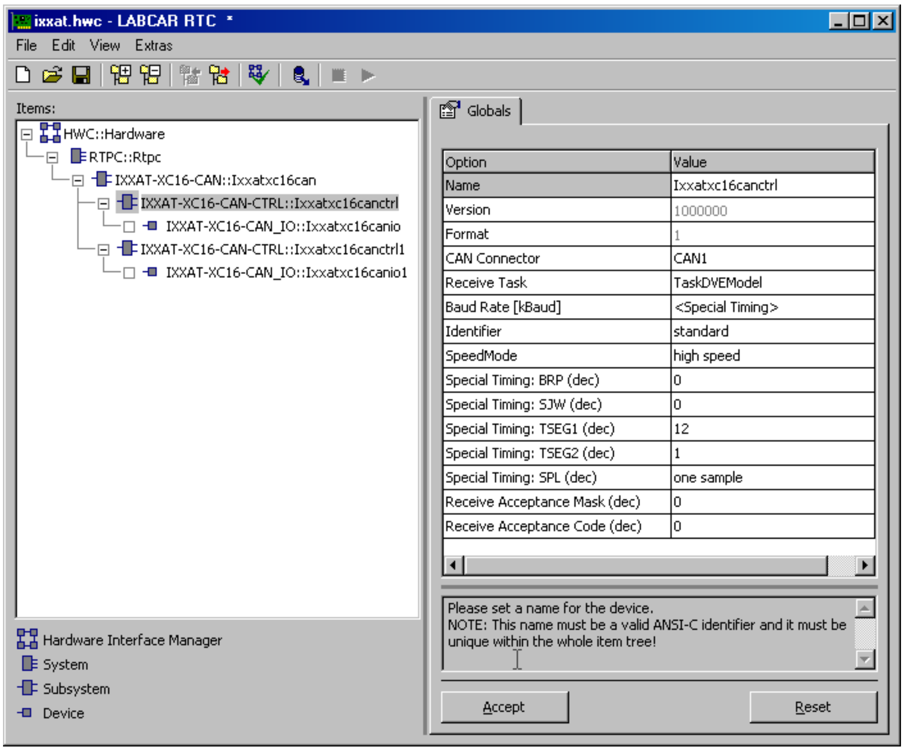


Fig. 27-4 The "Globals" Tab of an IXXAT-XC16-CAN-CTRL Subsystem

CAN Connector

This option field is used to assign the IXXAT-XC16-CAN-CTRL subsystem to a CAN controller or CAN connector on the slot sheet of the IXXAT-XC16-CAN. Please note that an IXXAT-XC16-CAN-CTRL subsystem can only be assigned to a single CAN controller. Possible values are "CAN1" and "CAN2".

Receive Task

With the IXXAT-XC16-CAN, all CAN receive messages are processed in one central function and then distributed to the individual signals. This option is used to select the task in which this central function runs. This is why it is not possible to select a task for receive messages in the IXXAT-XC16-CAN_IO device of the RTIO.

Baud Rate [kBaud]

The transfer rate of the CAN controller is specified in this option field. The standard baud rates (1000, 500, 250, 125, 100, 50, 20, 10 kBaud) are available. Using the "Special Timing" setting, the low-level control of the CAN controller is activated with reference to bit timing and baud rate.

During the configuration of "Baud Rate", the "Special Timing" option fields are automatically configured with values with reference to the transfer rate set and transferred to the firmware.

Identifier

You can select the length of the Identifier field of the CAN message in this line. With CAN messages, you can always choose between "standard" frames with 11-bit identifiers or "extended" frames with 29-bit identifiers.

Note

The setting of the length of the identifier must be the same per CAN controller.

Note

If the "standard" identifier is selected, only 11-bit values can be entered in the "Groups" tab of the IXXAT-XC16-CAN_IO device. If a larger value is entered, the "most significant bits" (MSB) are cut off. No warning is displayed.

Note

If the "extended" identifier is selected, only 29-bit values can be entered. If a larger value is entered, the "most significant bits" (MSB) are cut off. No warning is displayed.

SpeedMode

This option field is used to switch between "high-speed" and "low-speed" CAN functionality. The option "high speed" defines the CAN controller as a "high-speed" interface whereas "low speed" defines it as a "fault tolerant low-speed" interface.

Special Timing: BRP (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Baud Rate Prescaler" which determines the baud rate from the input clock of the CAN controller. The value range of this setting is 0 - 255.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: SJW (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Resynchronization Jump Width" within the bit timing. The value range of this setting is 0 - 3.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: TSEG1 (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set "Phase Segment 1" within the bit timing. The value range of this setting is 0 - 7.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: TSEG2 (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set "Phase Segment 2" within the bit timing. The value range of this setting is 0 - 7.

For more information on this setting, refer to the CAN controller data sheet.

Special Timing: SPL (dec)

This option can only be edited when the "Baud Rate [kBaud]" option is set to "Special Timing". This parameter is used to set the "Sampling Mode" and defines how often the signal is sampled to determine the logical state. The possible settings are "one sample" (sampled once) or "three samples" (sampled three times).

For more information on this setting, refer to the CAN controller data sheet.

The formula for calculating the CAN bus baud rate is:

Baud Rate = 8 MHz / [(BRP + 1) x (3 + TSEG1 + TSEG2)]

Receive Acceptance Mask (dec)

For information on this setting, refer to the CAN controller data sheet.

Receive Acceptance Code (dec)

For information on this setting, refer to the CAN controller data sheet.

27.3 IXXAT-XC16-CAN_IO Device

27.3.1 Globals (IXXAT-XC16-CAN_IO Device)

Fig. 27-5 shows the "Globals" tab of an IXXAT-XC16-CAN_IO device.

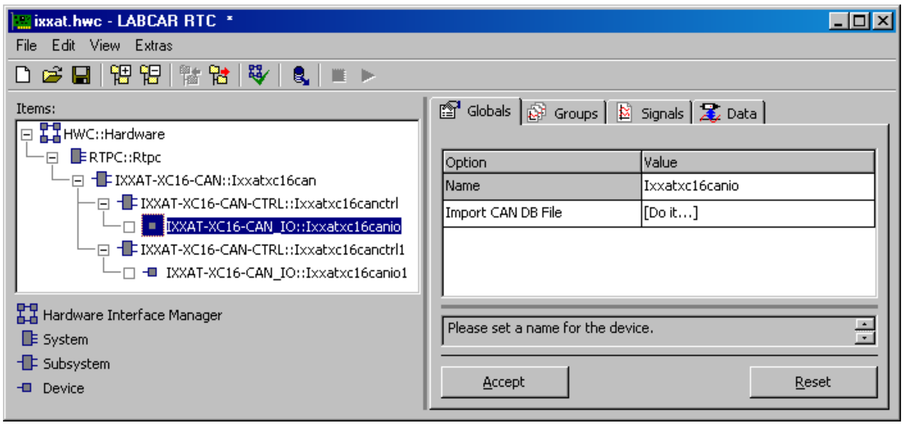


Fig. 27-5 The "Globals" Tab of the IXXAT-XC16-CAN_IO Device

Import CAN DB File

This option is used to read in a CAN database file which was created with the CANdb data management program made by the company Vector Informatik. This file can be used to create CAN messages and signals automatically if necessary.

The following dialog box opens when you press [Do it...]:

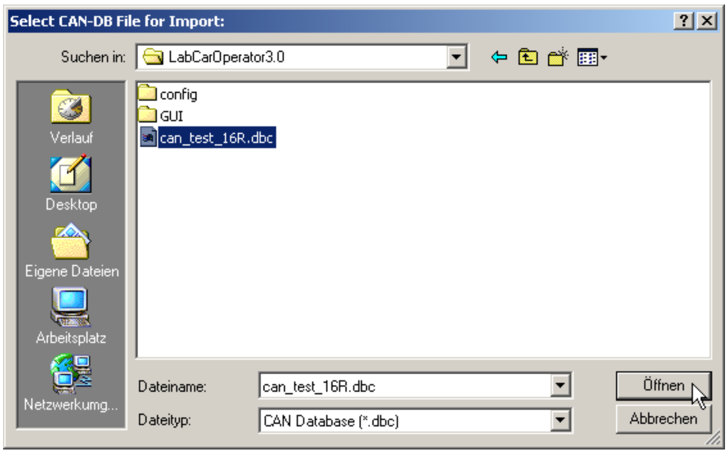


Fig. 27-6 CANdb Selection

The CAN DB file you want to import can be selected in this dialog box.

After pressing the **Open** button, the following dialog box opens in which you can specify the subsequent procedure.

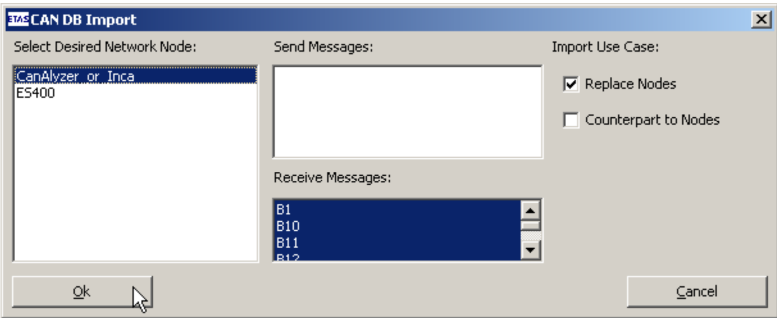


Fig. 27-7 CANdb Import Dialog

Several nodes of a CAN network are normally described in a CAN DB file. All existing nodes are shown in the left-hand list (Network Nodes). The two lists to the right of this (Send Messages/Receive Messages) list all the CAN messages defined for the node currently selected. The messages selected are used for the import. The "Import Use Case" can also be selected.

The "Replace Nodes" option means that the IXXAT-XC16-CAN_IO device assumes the role of the network mode, i.e. a send message of the node is also implemented as a send message of the IXXAT-XC16-CAN_IO device etc. The "Counterpart to Nodes" option means that the IXXAT-XC16-CAN_IO device is the counterpart to the network node, i.e. a send message of the node triggers a receive message etc.

After pressing the **OK** button, the data to be imported (CAN messages and signals) is checked together with the available signal groups and signals. The result of this check is then displayed in a dialog box.

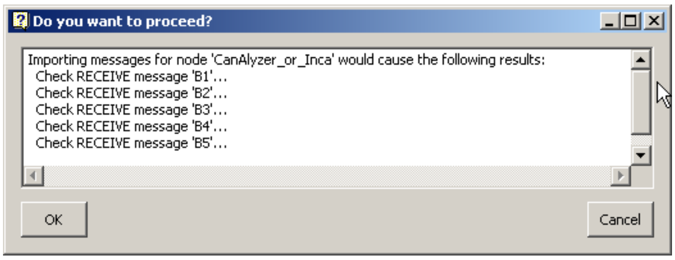


Fig. 27-8 CANdb Import Check

So far nothing has been modified in the available IXXAT-XC16-CAN_IO device. The import procedure is not started until the **OK** button is pressed. The import procedure adds imported messages to the signal groups and also ensures the definition of the relevant signals.

Once the import procedure has been completed, the detailed protocol of the procedure is displayed in the "Monitor" window.

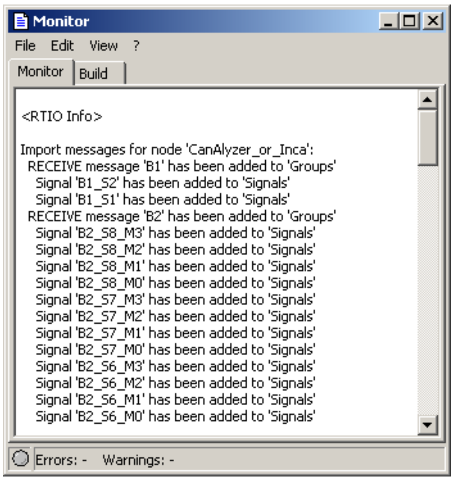


Fig. 27-9 CANdb Import Protocol

Note

When a CAN DB file is imported, only 29-bit ("extended" identifier) and 11-bit ("standard" identifier) identifiers are automatically added to the Identifier field. If the "standard" identifier was selected in the CAN-CTRL subsystem but the CAN DB file contains signals with 29-bit identifiers (ID > 231), the following takes place:

- *11 bits (bits [28...18]) are automatically written to the Identifier field. The other bits (MSB) are rejected.*
- *Warnings are shown in the monitor window. As only one identifier can be selected, there are conflicts with CAN DB files which contain both "standard" and "extended" identifiers.*

27.3.2 Groups (IXXAT-XC16-CAN_IO Device)

The CAN messages are specified in the form of signal groups in this tab.

Note

You can create new signal groups or CAN messages via the shortcut menu in the "Groups" tab.

Note

If the name of the signal group or signal, or the direction of the signal ("Direction") is changed, any message assigned may not be assigned automatically any more and may therefore have to be reassigned manually.

Fig. 27-10 shows the "Groups" tab of an IXXAT-XC16-CAN_IO device.

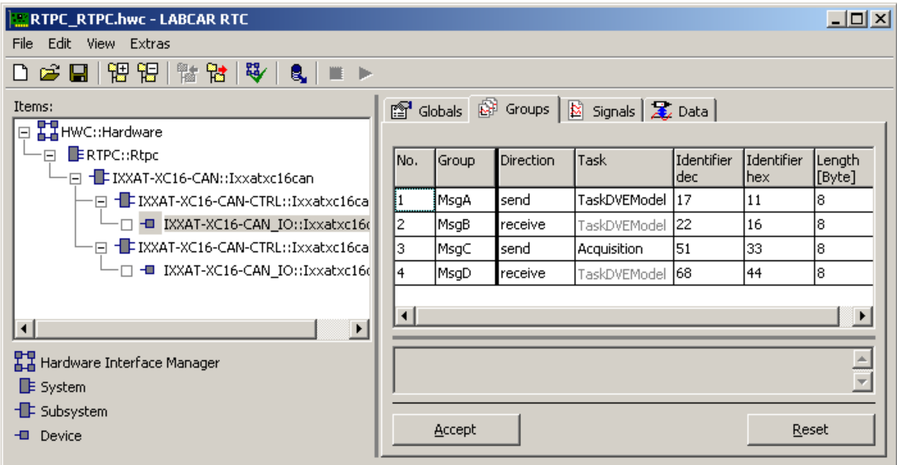


Fig. 27-10 The "Groups" Tab of the IXXAT-XC16-CAN_IO Device

27.3.3 The CAN Messages as Signal Groups

Direction

This is where you can determine the direction of the CAN message ("send" = send message, "receive" = receive message).

Task

For send messages, this is where the task is specified in which the message is to be sent.

All receive messages are processed in one common task; an individual setting is not possible here. The common task for the receive messages is shown here in gray; it can be selected in the IXXAT-XC16-CAN-CTRL subsystem.

Identifier (dec / hex)

The message identifier has to be entered here. The value can vary in size depending on the selected identifier type setting in the superior IXXAT-XC16-CAN-CTRL subsystem ("standard" or "extended"):

Each signal group or CAN message has to have a different identifier:

- "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]

Determines the number of useful data bytes the relevant CAN message can transfer (1 to 8 bytes).

27.3.4 Signals (IXXAT-XC16-CAN_IO Device)

The CAN messages are specified further in the "Signals" tab.

Note

New CAN signals can be created using the shortcut menu in the "Signals" tab.

Fig. 27-11 shows the "Signals" tab of the IXXAT-XC16-CAN_IO device.

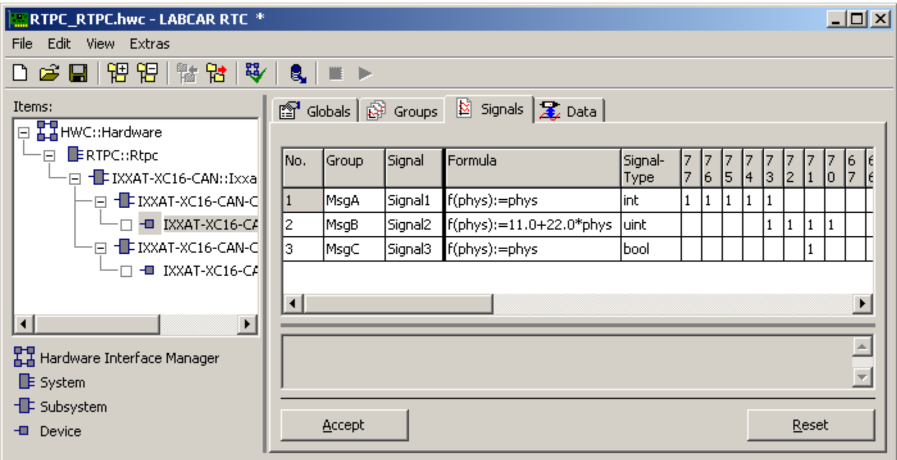


Fig. 27-11 The "Signals" Tab of the IXXAT-XC16-CAN_IO Device

Group

This is where a signal is assigned to the required signal group.

Signal Type

This is where the signal type is determined how the signal is transferred via the CAN bus.

Tab. 27-1 shows an overview of the possible signal types:

Signal Type	Data Type
int	Denotes a signed signal in the default complement to two data format (max. 32-bit)
(s) int	Denotes a signed signal in which the sign and then the absolute value of the signal are transferred as the most significant bit. If the sign bit is set, this is a negative integer (max. 32-bit)
uint	Denotes an unsigned signal (max. 32-bit)
bool	Denotes a Boolean signal. Only one single bit can be marked in the bit matrix.
real	Denotes a floating-point value in the "Standard IEEE Float (4 Byte)" format. Only 32 bits can be marked in the bit matrix for this. Structure of the data type in acc. with IEEE: - Sign: 1-bit - Exponent: 8-bit - Fraction: 24-bit

Tab. 27-1 Signal Types

Bit Matrix

A CAN message can transfer up to 8 data bytes. In a bit matrix you can specify for each signal which bits this signal requires or occupies.

The structure of the columns is as follows:

7	7	...	0	0	Byte number
7	6		1	0	Bit number

Meaning of the Bit Fields:

- Empty field

  The relevant signal does not use the bit
- Occupied field

  The signal requires this bit at this position
- "x" field

  The relevant bit is not available for data transfer because the signal has fewer useful data bytes (see the "Length" setting in the "Groups" tab).

Operating the Bit Fields:

The required bit cell can be selected using the cursor keys of the keyboard. You can "toggle" the cell between "unoccupied" (cell empty) and "1" (bit occupied) by clicking with the mouse. If you also hold down the <Alt> key while clicking, the value is incremented from "1" to "9" (block building).

You can select several bits at the same time as follows:

1. Select the first bit

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																															1

2. Select the second bit

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																								1							1

3. Hold down the <Shift> key and at the same time select the second bit again - the entire block between them is then selected.

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																								1	1	1	1	1	1	1	1

Data blocks with which virtually every transferred signal can be described can be built using the different numbers ("1111 2222..."). The numbers from which the data blocks are built have the significance that the block with the highest number ("2222") specifies the block which contains the most significant bits on transfer. The block with the smallest number ("1111") contains the least significant bits on transfer.

A signal can therefore be described with up to 9 bit blocks with the numbers available (1...9). As the representation of the bits is exactly the form in which an Intel signal is transferred, block building for the representation of signals in Motorola format is necessary as soon as the signal exceeds a length of 8 bits.

Examples of the Definition of Different Signals:

16-bit signal in Intel format:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Different description of the same signal:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1

12-bit signal in Intel format with gap:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1												2	2	2	2									1	1	1	1	1	1	1	1

16-bit signal in Motorola format:

No.	Group	Signal	36	35	34	33	32	31	30	27	26	25	24	23	22	21	20	17	16	15	14	13	12	11	10	07	06	05	04	03	02	01	00
1	Group1	Signal1																1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2

27.3.5 Data (IXXAT-XC16-CAN_IO Device)

Fig. 27-12 shows the "Data" tab of an IXXAT-XC16-CAN_IO device.

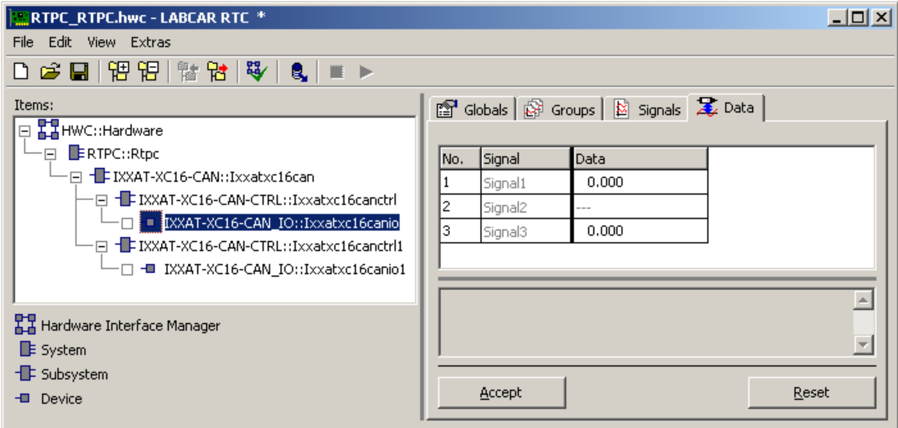


Fig. 27-12 The "Data" Tab of the IXXAT-XC16-CAN_IO Device

In the "Data" tab (Fig. 27-12 - "offline" mode), you can see that the signal "Signal1" is used to send a CAN message. In the "Data" field, the useful data can be occupied in "online" mode. In "offline" mode, the "Data" fields of the receive messages "Signal2" and "Signal3" are not active; in "online" mode, the received useful data of the relevant CAN messages is displayed there.

28 The RTIO Package for Transputer-Based Hardware

There are two hardware realizations for linking transputer-based LabCar2 hardware to LABCAR-OPERATOR.

One of these is the ES4205 system. The system consists of a VMEbus chassis, built into the LabCar2 signal box and providing five VMEbus slots, and a board implementing the connection between the parallel VMEbus and the serial transputer link bus of the LabCar2 hardware.

ES4205 VME64x Integration - ES1206 Serial Interface Board (T8VME)

The ES4205 board consists of an ES1206 Serial Interface Board (T8VME), plugged into the VMEbus part of the chassis, and a piggyback module plugged onto the ES1206 that is connected to the transputer link system. Both bus systems are connected using firmware that runs on the transputer of the ES1206. The ES4205 system allows a LabCar2 signal box to be connected to LABCAR-OPERATOR.

ES1206 - ES1381 - ES4206

The second hardware realization is used if more than one LabCar2 signal box is to be addressed. On the VMEbus side, the hardware consists of an ES1206 Serial Interface Board (T8VME) with an ES1381 Serial Interface Piggyback. A special transputer link cable that routes 5 transputer links connects the ES1381 to an ES4206 Serial Interface Board in the LabCar2 signal box. The ES1206/ES1381 board is built into a standard VMEbus chassis, such as the ES4100 system. A separate ES1206/ES1381 board must be used for each LabCar2 signal box to be connected. The bus systems are connected, as with the ES4205 system, using firmware that runs on the transputer of the ES1206.

The RTIO Package includes all drives required to control the boards of a LabCar2 signal or load box with a LABCAR-OPERATOR system.

Contents of this Section

This section describes the options of the RTIO Editor for the ES4205 and the ES1206/ES1381 boards, as well as the transputer-based LabCar2 hardware. For information about the features and functionality of each board, please refer to the relevant hardware documentation. The following boards are discussed below:

This chapter focuses exclusively on option settings that are specific to these boards. Options that have the same significance for all boards are described in "Configuration Tabs" on page 40.

28.1 ES4205 and ES1206/ES1381 - Bus Link

In the RTIO Editor, the description or the connection of LabCar2 hardware takes place by allocating an ES4205 or ES1206-ES1381-ES4206 subsystem to an ES1130 system.

28.1.1 Globals (ES4205 or ES1206-ES1381-ES4206 Subsystem)

This section describes the global options of the item.

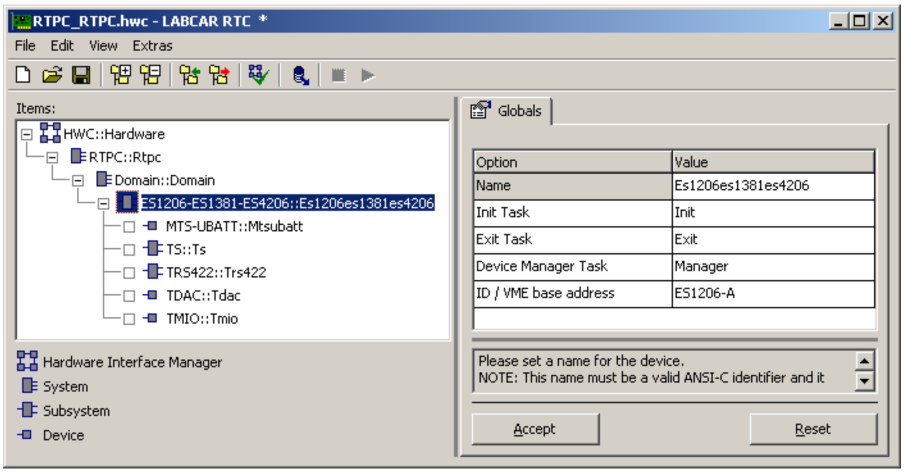


Fig. 28-1 The "Globals" Tab of the ES4205 / ES1206-ES1381-ES4206 Subsystem

Device Manager Task

This is where a task is entered in which the system services of the ES4205 or ES1206 are processed.

These system services consist (in terms of the display) of error messages, warnings or information messages which are sent by the LabCar2 boards to the host PC and are intended for the user. These messages are first written to a buffer on the ES4205 or ES1206. The messages are read from this buffer by the ES1130 in the Device Manager task and sent to the host PC. To ensure the buffer on the ES4205 or ES1206 does not overflow, which would result in messages being lost, the activation time of the Device Manager Task should be smaller than or equal to 10 ms.

ID / VME base address

The VMEbus base address of the ES1206 boards is specified in this field. This base address is coded in the EPLDs of these boards and can only be changed by ETAS. Please note that this field is disabled for ES4205 boards as these generally only support one VMEbus base address.

28.2 MTS-UBATT/UBATT-POW - Battery Control

The combination of MTS-UBATT board and UBATT-POW board fulfills primarily three objectives in LABCAR:

- carrier unit for processor modules

  In contrast to the previous software generation, the calculation of model parts on the MTS-UBATT board is not supported any longer by LABCAR-OPERATOR. The entire model is now calculated on the ES1130 processor board or on a Real-Time PC.

- control of a power supply unit connected externally with LABCAR for battery voltage emulation
- emulation of "battery nodes", i.e. circuit breakers, for battery voltage supply to the ECU, LABCAR I/O hardware and loads

The two boards form one combined unit, although they are distributed across two boards for circuitry-design reasons. Only one MTS-UBATT board and one UBATT-POW board can be used in LABCAR.

Apart from this the two boards communicate with each other via an interface specially developed for LABCAR; therefore, the UBATT-POW board can only be controlled by the MTS-UBATT board. It is not possible to use the UBATT-POW board without the MTS-UBATT board. This is why, since all options are set via the MTS-UBATT board, there is no RTIO item for the UBATT-POW board.

28.2.1 Globals (MTS-UBATT Device)

This section describes the global options of the MTS-UBATT device.

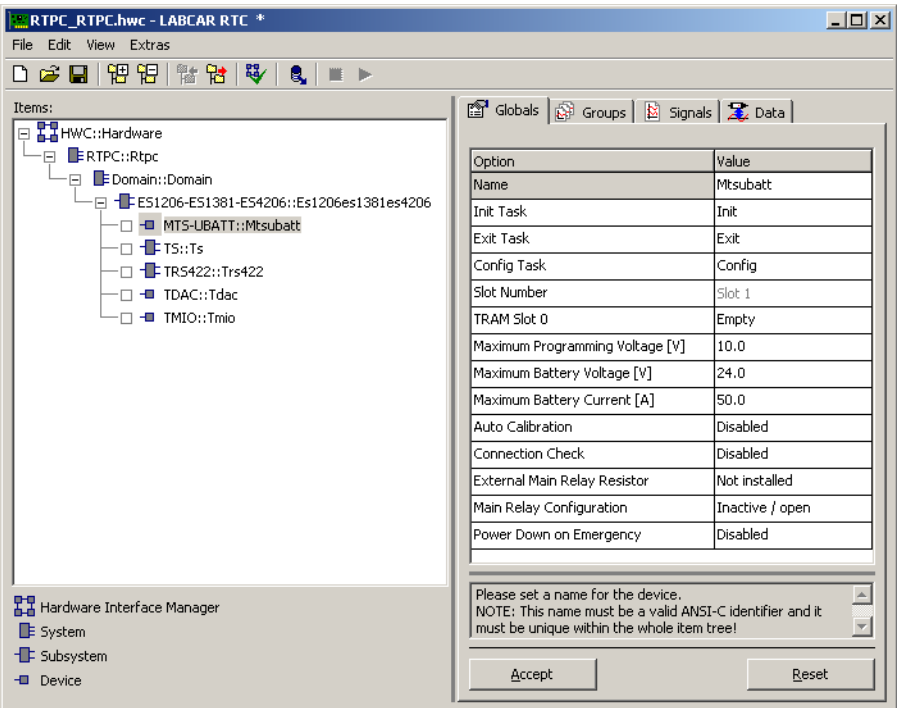


Fig. 28-2 The "Globals" Tab of the MTS-UBATT Device

Slot Number

Enter the slot number of the LabCar2 signal box which holds the MTS-UBATT board. The slots are numbered from left to right (starting with 1). An MTS-UBATT board can only be inserted in the first slot of a LabCar2 signal box.

TRAM Slot 0

The MTS-UBATT board has slots for up to eight TRAM modules (modules which consists of a transputer and the corresponding main memory). Specify here whether slot 0 is equipped with a TRAM module ("TRAM Module") or not ("Empty").

This information is required by the boot algorithm of the LabCar2 system. This boot algorithm loads a special program, referred to as the router, which forwards messages in the serial LabCar2 system to transputers which have no I/O functionality. This is why any TRAM modules installed in slot 0 of the MTS-UBATT board have to be specified. TRAM modules in slots 1- 7 of the MTS-UBATT board do not have to be specified as the boot algorithm bridges these slots by setting a link connection.

Maximum Programming Voltage [V]

In the LABCAR sector, a power supply unit is used to emulate the battery. These power supply units are virtually all available in versions that can be addressed electronically. Analog interfaces are usually used for this purpose. An externally applied "control voltage" can then be used, for example, to set the output voltage of the power supply unit.

Enter the maximum control voltage of the external power supply unit here (in volts). The model controls the external power supply unit by setting a target voltage and a maximum current. The analog interface of the power supply units usually permits only one relative set value (e.g. a control voltage between 0 V and 10 V of the linear control of the output voltage to values between 0 V and the maximum voltage of the unit).

To be able to convert the absolute values into relative ones, the maximum output voltage, maximum output current, and maximum control voltage of the external power supply unit need to be entered.

Maximum Battery Voltage [V]

Maximum output voltage of the external power supply unit in volts.

Please refer to "Maximum Programming Voltage [V]" on page 892.

Maximum Battery Current [A]

Maximum output current of the external power supply unit in amps.

Please refer to "Maximum Programming Voltage [V]" on page 892.

Auto Calibration

The MTS-UBATT board features an auto-calibration mechanism for the analog interfaces to and from the external power supply unit. Each board, when delivered, is already properly calibrated. However, should you discover an inaccuracy in the setting of the external power supply unit, a new auto-calibration can be achieved by setting this option to "Enabled". The actual calibration is performed the next time the board is initialized.

Connection Check

The MTS-UBATT board has a special serial high-speed interface for communicating with the UBATT-POW board. By default, the connection between the MTS-UBATT board and the UBATT-POW board is checked for proper operation each time the two boards communicate with each other. If an error is detected, the model generates a corresponding error message. To avoid this in a LABCAR which has no UBATT-POW board installed, the error detection mechanism can be suppressed by setting this option to "Disabled".

External Main Relay Resistor

The UBATT-POW board gives the user the possibility of evaluating an existing ECU signal for addressing a "main relay" directly in the hardware and to incorporate this in the enabling/disabling of the individual battery nodes.

The board has a special input for this purpose which, controlled by software, can be assigned with a pull-up or pull-down circuit to define the level of the inactive signal state.

The pull-up or pull-down circuit on the MTS-UBATT board is a stabilized current regulator with a nominal current of 160 mA. To avoid the connected ECU being overloaded, it is possible to realize the pull-up or pull-down function with an external connection.

This option defines against which battery node the external resistance of the main relay circuit is connected. If you are using the internal pull-up or pull-down circuit, select "Not installed".

Main Relay Configuration

This is where you can specify the connection of the main relay input of the UBATT-POW board. If you select "External pull down" or "Onboard pull down", the main relay is considered active when its signal level is above half the battery voltage. If you select "External pull up" or "Onboard pull up", the main relay is considered active when its signal level is below half the battery voltage.

Select "Onboard pull down" or "Onboard pull up" to choose the internal stabilized current regulator for the main relay circuit. For the "External pull down" or "External pull up" selections, you need to insert an external connection to specify the level at the output of the ECU.

Selecting "Inactive / open" disables main relay detection, i.e. it is always evaluated as inactive.

Power Down on Emergency

This field determines how the software reacts to an active "Emergency" signal. If this option is set to "Disabled", an active *Emergency* signal results in all battery node switches being opened. The value of the battery voltage, however, remains unchanged. If this option is set to *Enabled*, the battery voltage is also set to 0, i.e. the LABCAR energy supply is switched off. The "Emergency" signal is an input signal of the board's RTIO driver.

28.2.2 Groups (MTS-UBATT Device)

This section describes the signal-group-specific options of the MTS-UBATT device.

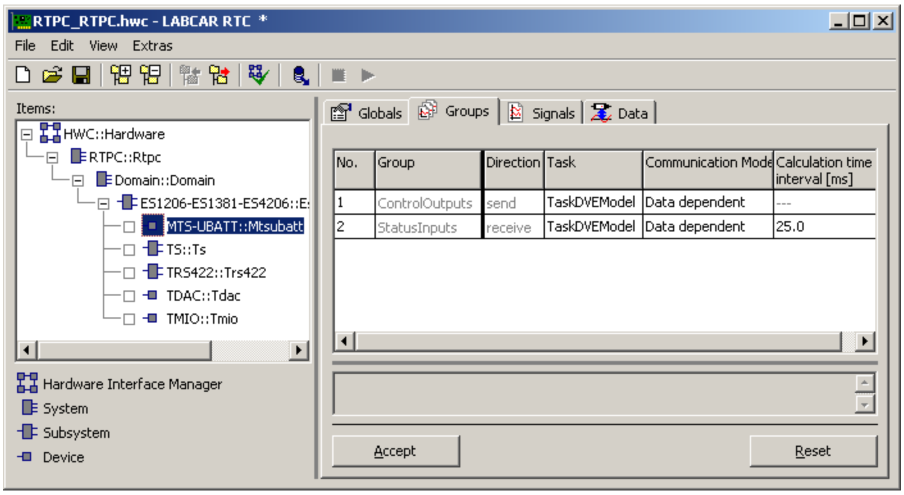


Fig. 28-3 The "Groups" Tab of the MTS-UBATT Device

There are two signal groups for the MTS-UBATT board:

- ControlOutputs

  ControlOutputs are controlled by the model and modify the settings of the MTS-UBATT board. These signals are used to switch between the battery nodes and to set the battery voltage and battery current.

- StatutsInputs

  StatutsInputs contain information about the hardware status.

  They are used to capture the battery voltage and current as well as the switching and error statuses of the battery nodes and the main relay.

Communication Mode

This field specifies when a data transfer takes place between the simulation target and the MTS-UBATT board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

Calculation time interval [ms]

A process runs on the on-board transputer of the MTS-UBATT board that determines the status information of the battery nodes and the main relay and also measures the battery voltage and current. The time interval, in which this process is calculated or activated, is specified in this field.

28.2.3 Signals (MTS-UBATT Device)

This section describes the signals of the MTS-UBATT board and their special options.

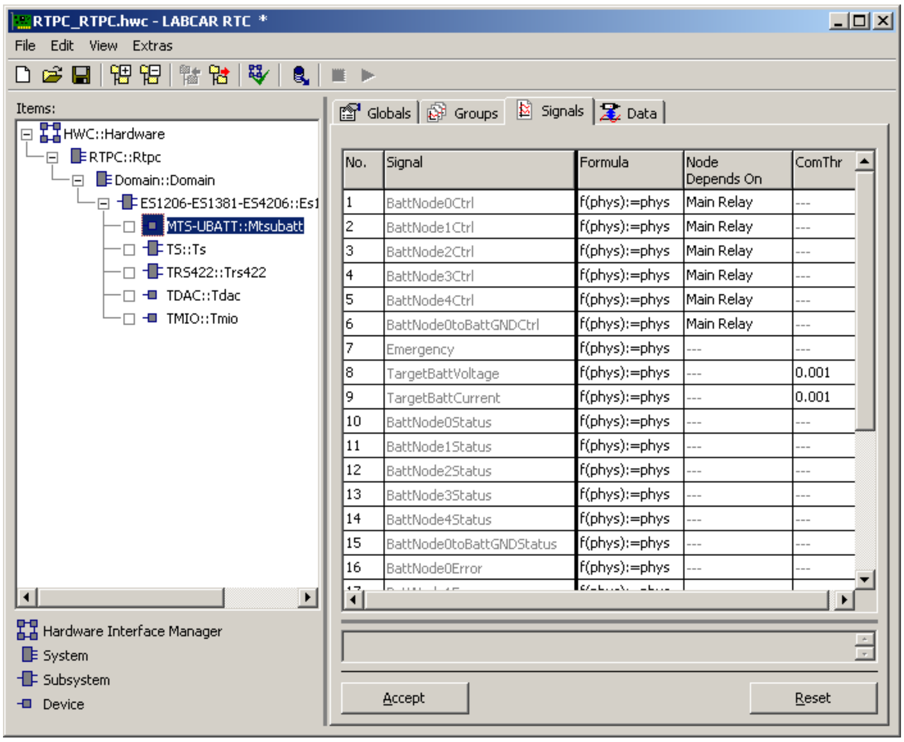


Fig. 28-4 The "Signals" Tab of the MTS-UBATT Device

The data types and the meaning of the various values of each signal are described in the section "Data Types and Value Ranges" on page 897.

Signal	Direction	Description
BattNodenCtrl	send	Switches the battery node *n* to battery voltage. If a node `n` is configured so that it does **not** depend on the status of the main relay, *BattNodenCtrl* specifies whether it should be switched (≠ 0) or opened (0). If, on the other hand, the node depends on the main relay status, *BattNodenCtrl* is of no significance.
BattNode0toBattGNDCtrl	send	Switches the battery node 0 to battery ground. If the battery node 0 is controlled by the main relay status, *BattNode0toBattGNDCtrl* is of no significance.
Emergency	send	Signal for emergency stop of all battery nodes and disabling of the main relay. If "Power Down on Emergency" is "enabled", the battery voltage is also set to 0.
TargetBattVoltage	send	Target battery voltage
TargetBattCurrent	send	Maximum target current

Signal	Direction	Description
BattNodenStatus	receive	Switching status of respective battery node
BattNode0to-BattGNDStatus	receive	Indicates whether battery node 0 is connected to ground.
BattNodenError	receive	The signal provides the model with status information on the battery node switch *n*. If the signal returns a value ≠ 0, node *n* has been disabled due to an error status (such as a short). If there is no load, the signal value is also ≠ 0, but the node is *not* disabled.
BattNode0to-BattGNDError	receive	Provides the model with status information on the battery node switch 0 to ground (please compare with *BattNodenError*).
MainRelayStatus	receive	Status of the main relay
MainRelayError	receive	Indicates whether there has been an error at the main relay.
ActualBattVoltage	receive	Actual battery voltage in volts
ActualBattCurrent	receive	Actual battery current in amps

Node Depends On

This option defines whether the respective battery node is controlled by the model ("Input") or whether it depends on the state of the main relay ("Main Relay").

ComThr

This option is only of any significance when "Data dependent" has been selected in the "Communication Mode" field of the relevant signal group. In this case, "ComThr" specifies how much the relevant signal has had to change with reference to its maximum value for a data transfer to take place between the simulation target and the MTS-UBATT board. If, for example, a voltage signal has a maximum value of 10 V and a value of 0.01 is specified for the communication threshold, `ComThr`, the voltage signal has to change 1%, i.e. 100 mV, for a data transfer to take place.

28.2.4 Data Types and Value Ranges

This section describes the possible data types and the meaning of the various values of each signal.

Signal	Data Types	Possible Values
BattNode*n*Ctrl	int	≠ 0 : battery node *n* is connected to battery voltage 0 : battery node *n* is disconnected from battery voltage
BattNode0to-BattGNDCtrl	int	≠ 0 : battery node 0 is connected to battery ground 0 : battery node 0 is disconnected from battery ground
Emergency	int	≠ 0 : emergency mode is enabled 0 : emergency mode is disabled
TargetBattVoltage	real32	Target battery voltage in volts
TargetBattCurrent	real32	Maximum target current in amps
BattNode*n*Status	any integer data type	≠ 0 : battery node *n* is connected to battery voltage 0: battery node *n* is disconnected from battery voltage
BattNode0to-BattGNDStatus	any integer data type	≠ 0: battery node 0 is connected to battery ground 0: battery node 0 is disconnected from battery ground
BattNode*n*Error	any integer data type	≠ 0: error in battery node *n* 0: no error in battery node *n*
BattNode0to-BattGNDError	any integer data type	≠ 0: error at switch of battery node 0 to ground 0: no error
MainRelayStatus	any integer data type	0: main relay is inactive ≠ 0: main relay is active
MainRelayError	any integer data type	≠ 0: error in main relay 0: no error in main relay
ActualBattVoltage	real32, real64	Actual battery voltage in volts
ActualBattCurrent	real32, real64	Actual battery current in amps

28.3 TS-PWM and Arbitrary Signal Generation

The TS board (Transputer Signal Generator Board) is a LABCAR stimulation board that is used to generate PWM and freely programmable signals with a sampling rate smaller than or equal to 1 MHz.

The board contains two groups of signal generators:

- freely programmable signal generators (arbitrary signal generators)

  A total of three arbitrary signal generators are available to the user. Every signal generator contains two channels that are output in sync with each other via the same control logic.

- pulse-width modulated signal generators (PWM signal generation)

  The second generator group contains four PWM generators. The signal traces for this signal type are determined with a settable period and a duty cycle. The PWM signals are generated asynchronously to the arbitrary signals.

The functionality of the board is supplemented by a central speed module. This module is used to generate speed-synchronous signals and thus forms the "heart" of the stimulation of engine ECUs. The speed module is used to control the arbitrary signal generators and for digital signal capture with the TMIO board.

The TS board is used to generate the following vehicle signals:

- generation of speed-synchronous signals with arbitrary signal generators
  - speed signal (is used by the engine ECU to measure engine speed and the crankshaft angle)
  - camshaft signal (the camshaft signal is required by the ECU for cylinder detection)
  - knock signal (the knock signal is used for the simulation of the knock sensor)
- generation of pulse-width modulated signals with PWM signal generators
  - wheel speed signal (simulation of vehicle speed)
  - pedal sensor signal (simulation of accelerator, clutch, brake pedal)

28.3.1 Globals (TS Subsystem)

This section describes the global options of the TS subsystem.

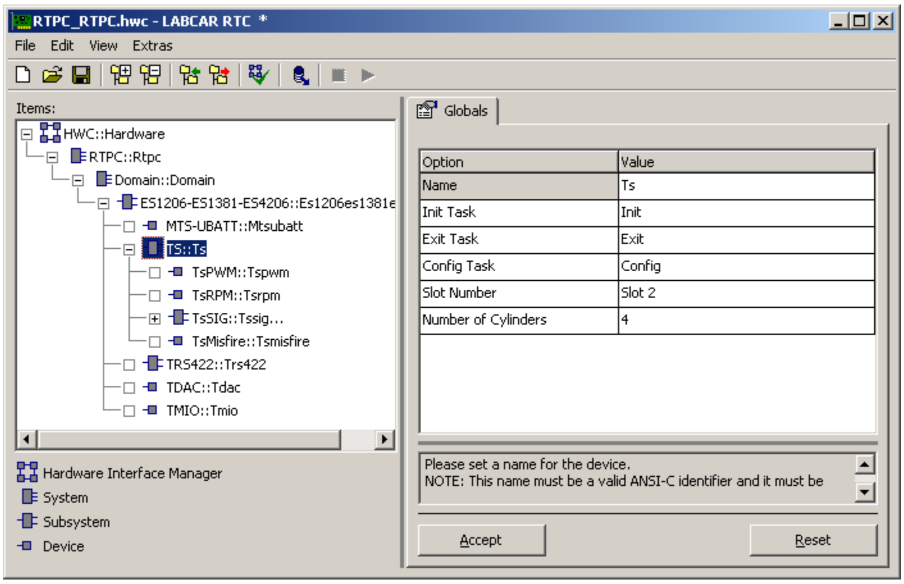


Fig. 28-5 The "Globals" Tab of the TS Subsystem

There are three separate RTIO items each for PWM signal generation, arbitrary signal generation and speed signal generation. These can be assigned to the TS item as subitems and have their own global options.

Slot Number

Enter the slot of the LabCar2 signal that holds the TS board. The slots are numbered from left to right starting with one.

Number of Cylinders

Both the misfire and the knock generator require a specification of the number of available cylinders.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Number of Cylinders	uint8	yes	The misfire generator supports 1 to 12 cylinders. The knock generator supports 1 to 8 cylinders.

28.3.2 Globals (TsPWM Device)

This section describes the global options of the TsPWM device.

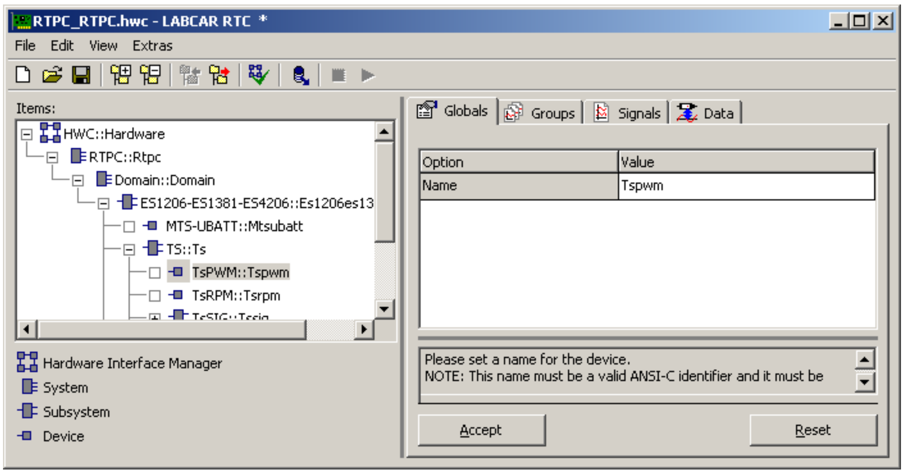


Fig. 28-6 The "Globals" Tab of the TsPWM Device

The TsPWM device is used to configure the four PWM signal generators of the TS board. The configuration is individual for each generator and can be modified at runtime from within the model.

28.3.3 Groups (TsPWM Device)

This section describes the signal-group-specific options of the TsPWM device.

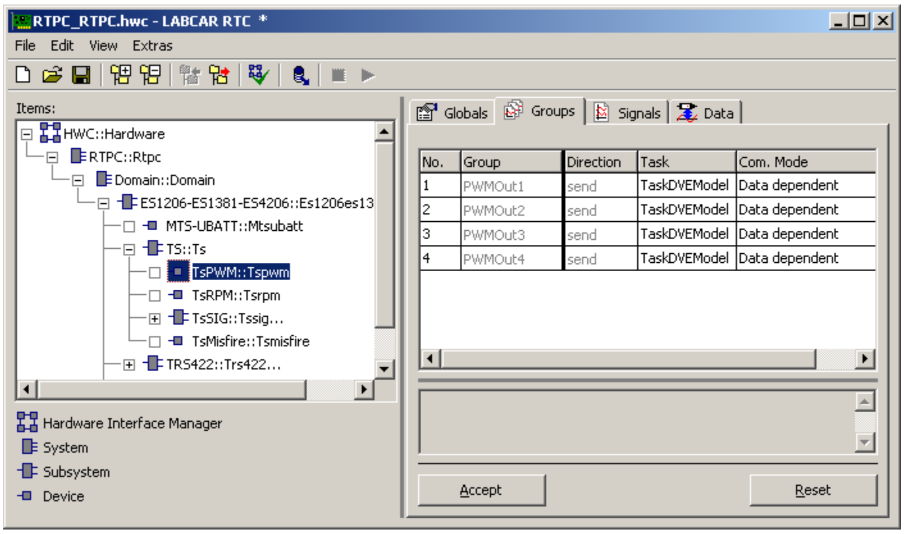


Fig. 28-7 The "Groups" Tab of the TsPWM Device

The TsPWM device has four signal groups which each control the options of one of the four PWM generators. All signals are assigned the *send* direction, which means that they are transferred from the simulation target to the TS board.

Com. Mode

This field specifies when a data transfer takes place between the simulation target and the TS board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

28.3.4 Signals (TsPWM Device)

This section describes the signals of the TsPWM device and their options.

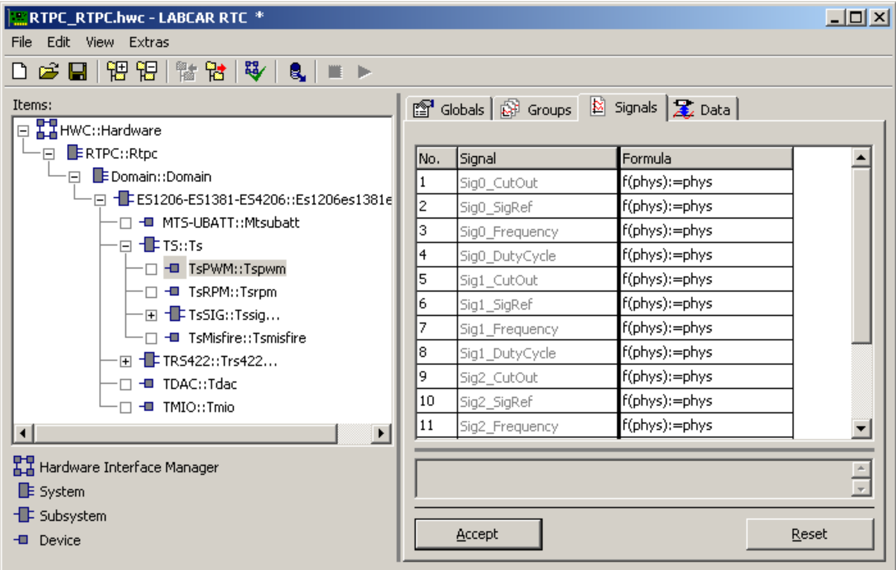


Fig. 28-8 The "Signals" Tab of the TsPWM Device

Each PWM generator can be configured by modifying the following four parameters.

Sign_Frequency

Specifies the frequency of the PWM signal *n*.

Sign_DutyCycle

Specifies the duty cycle of the PWM signal *n*. The duty cycle is the high level portion within the period duration of the PWM signal. The duty cycle is output as a factor of the period, e.g. a factor of 0.5 corresponds to half the period duration.

Sign_CutOut

Activates/deactivates the PWM signal *n*.

Sign_SigRef

Specifies the output stage of the PWM signal. You choose between an Open-Collector end stage and an end stage with an internal 2 kΩ collector resistance to battery node 4. If the internal end stage is selected, battery node 4 has to be activated for PWM output. Each PWM signal has its own input for external ground which has to be connected for proper operation of the PWM generator.

28.3.5 Data Types and Value Ranges

This section describes the possible data types for each signal and the meaning of the various signal values.

Signal	Data Types	Possible Values
Sign_Frequency	float	0.0 to 10000.0 [Hz]
Sign_DutyCycle	float	0.0 to 1.0
Sign_CutOut	bool	false = PWM output off true = PWM output on
Sign_SigRef	bool	false = end stage with internal 2 kΩ pull-up resistance to battery voltage true = open collector end stage

28.3.6 TsRPM Device

The speed generator is configured with the TsRPM device of the TS board. The speed generator is used to stimulate (clock and trigger) the arbitrary signal generators. It also provides the clock signal, the trigger signal and the crankshaft angle.

Clock Signal (Speed Generator Output)

A memory bank with 8192 data samples is filled with clock discrete values for one engine period. An engine period of a four-stroke engine includes one camshaft revolution or two crankshaft revolutions. Since the engine speed specifies the crankshaft frequency, the clock signal has the (8192/2)-fold frequency of the engine speed (n):

$$f_{Clock} = n_{Engine} \cdot \frac{MemoryBankSize}{2}$$

Equ. 28-1 General Calculation of the Speed Clock

If the memory bank size is specified as 8192 as in Equ. 28-1 and the clock frequency is converted to hertz, the following relation is obtained:

$$\frac{f_{Clock}}{\mathrm{Hz}} = 68,2667 \cdot \frac{n_{Engine}}{rpm}$$

Equ. 28-2 Speed Clock for 8K Data Samples, TS Board

The maximum clock frequency is 1.2 MHz, i.e. the maximum speed is 17578 rpm.

Note

If the memory bank size is reduced (e.g. 2K or 4K), the maximum engine speed is increased. Reducing the memory bank size is, however, detrimental for the angle resolution.
If you want to change the memory bank size, please contact your LABCAR user consultant.

Crankshaft Angle

The angle resolution α is generally as follows in accordance with the memory bank size:

$$\alpha = \frac{720° \text{ CA}}{MemoryBankSize}$$

Equ. 28-3 Angle Resolution α in accordance with the Memory Bank Size (° CA = ° Crankshaft)

This results in the following for 8K data samples:

$$\alpha = \frac{720° \text{ Kw}}{8192} = 0,0879° \text{ Kw}$$

Equ. 28-4 Angle Resolution with 8K Data Samples

The crankshaft angle Γ result from the data sample n filled with clock discrete values and the angle resolution α:

$$\Gamma = n \cdot \alpha$$

Equ. 28-5 Crankshaft Angle Γ

Trigger Signal (Speed Generator Output)

Every 8192 data samples, a trigger signal is generated by the speed generator with the pulse duration of a clock period. This trigger pulse results in the new start of an engine period.

The trigger signal has the following frequency:

$$f_{Trig} = \frac{f_{Clock}}{MemoryBankSize}$$

Equ. 28-6 Trigger Frequency for Speed-Synchronous Signals, TS Board

The trigger signal is made available to the LABCAR user via a BNC jack (TFO) on the TS board front panel for external trigger purposes. All speed-synchronous processes within LABCAR are synchronized with the "master trigger".

28.3.7 Globals (TsRPM Device)

This section describes the global options of the TsRPM device and their options.

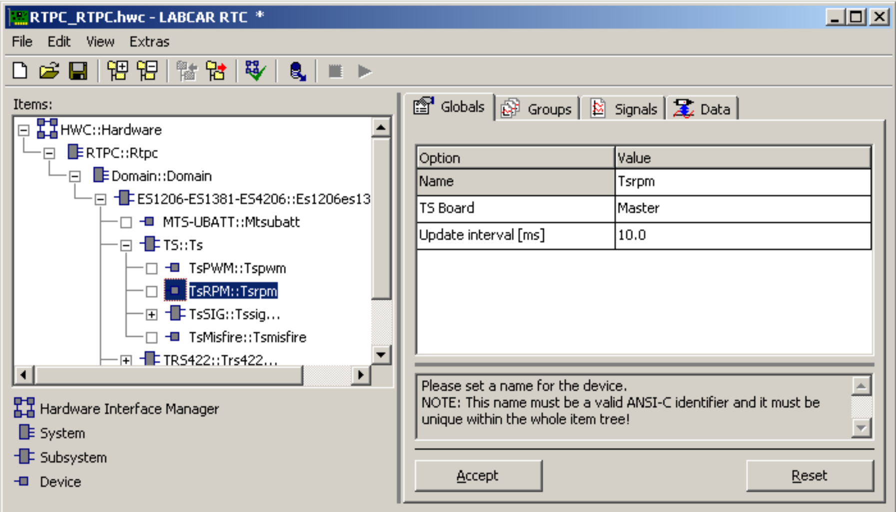


Fig. 28-9 The "Globals" Tab of the TsRPM Device

TS Board

Specifies whether the TS board is used as a master or slave. If more than one TS board is used in a signal box, one always has to be defined as the master. The master board specifies the speed signal while all other TS boards are synchronized with this signal via the backplane.

Update Interval [ms]

The interval in milliseconds at which the current crankshaft angle is fed back to the model.

28.3.8 Groups (TsRPM Device)

This section describes the signal-group-specific options of the TsRPM device.

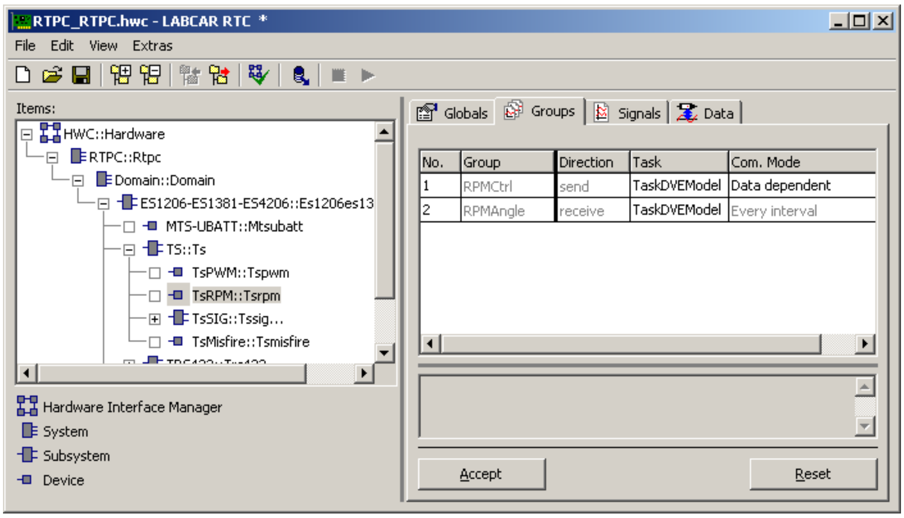


Fig. 28-10 The "Groups" Tab of the TsRPM Device

The TsRPM device has two signal groups:

- RPMCtrl

  Contains all signals used for the configuration of the speed generator. This group should be assigned to the task that calculates the current speed in the model.

- RPMAngle

  Contains the signal that returns the crankshaft angle to the model. This group’s interval should match the "Update Interval [ms]" parameter to avoid unnecessary data transmissions.

Com. Mode

This field specifies when a data transfer takes place between the simulation target and the TS board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

28.3.9 Signals (TsRPM Device)

This section describes the signals of the TsRPM device and their options.

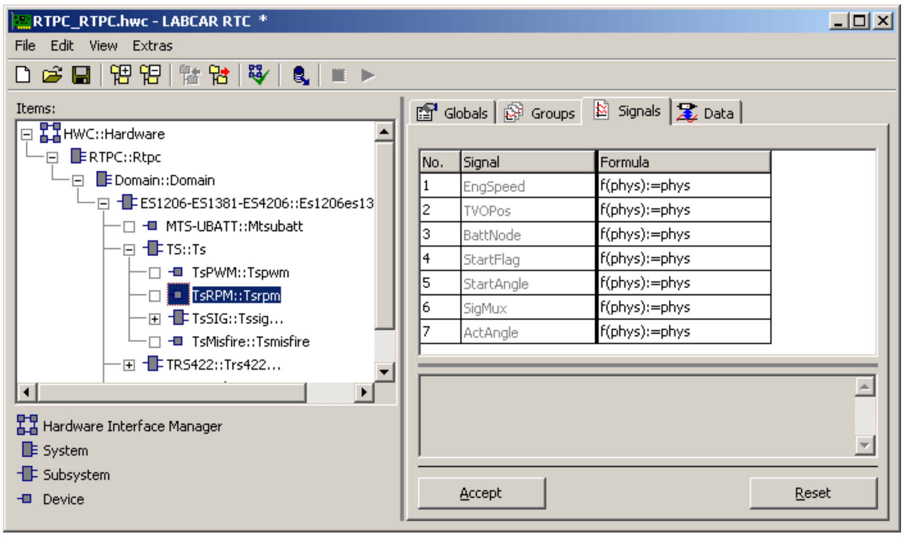


Fig. 28-11 The "Signals" Tab of the TsRPM Device

All signals of the TsRPM device, apart from *ActAngle*, are used to configure the board.

Signal	Direction	Description
EngSpeed	send	Target speed specified by model in revolutions per minute.
TVOPos	send	The offset in degrees for trigger signal, TVO, relative to trigger signal TFO. The maximum offset is one engine period (720° CA).
BattNode	send	Switches the "engine" on or off.
StartFlag	send	Switches "starting off" of a defined crankshaft angle at engine shutdown on or off.
StartAngle	send	Defines a crankshaft angle in degrees that is reached when the engine is shut-down, if this option was enabled with the *StartFlag* signal.
SigMux	send	Defines the channel issued at the signal multiplexer output, TSTO. This can be any channel of the TS board.
ActAngle	receive	The current crankshaft angle in degrees returned to the model.

28.3.10 Data Types and Value Ranges

This section describes the possible data types and value ranges for each signal and the meaning of the various signal values.

Signal	Data Types	Possible Values
EngSpeed	float	0 to 17578 (maximum speed)
TVOPos	float	0 to 720
BattNode	bool	true = engine on false = engine off
StartFlag	bool	true = "StartAngle" function activated false = "StartAngle" function not activated
StartAngle	float	0 to 720
SigMux	int	0 to 15, significance see Tab. 28-1 on page 907
ActAngle	float	0 to 720

The *SigMux* signal can take the following values:

0	Signal 1 (signal generator 1)
1	Signal 2 (signal generator 1)
2	Signal 1 (signal generator 2)
3	Signal 2 (signal generator 2)
4	Signal 1 (signal generator 3)
5	Signal 2 (signal generator 3)
6	PWM signal 1
7	PWM signal 2
8	PWM signal 3
9	PWM signal 4
10	Reference signal 1 (signal generator 1)
11	Reference signal 2 (signal generator 1)
12	Reference signal 1 (signal generator 2)
13	Reference signal 2 (signal generator 2)
14	Reference signal 1 (signal generator 3)
15	Reference signal 2 (signal generator 3)

Tab. 28-1 Values of the "SigMux" Signal

28.3.11 Globals (TsSIG Subsystem)

This section describes the global options of the TsSIG item.

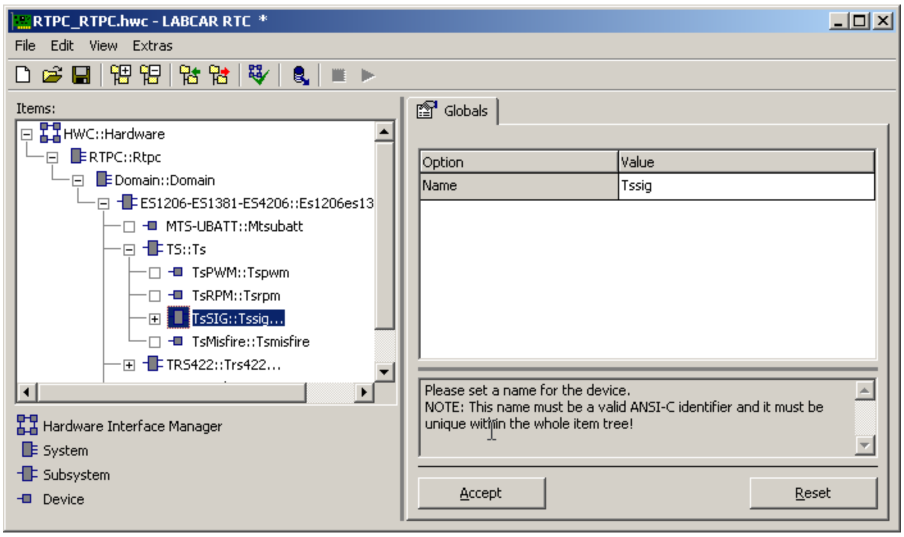


Fig. 28-12 The "Globals" Tab of the TsSIG Subsystem

The TsSIG subsystem contains up to three TsSIGGen devices. Each of these devices is responsible for configuring one of the three arbitrary signal generators of the TS board.

28.3.12 Globals (TsSIGGen Device)

This section describes the global options of the TsSIGGEN device.

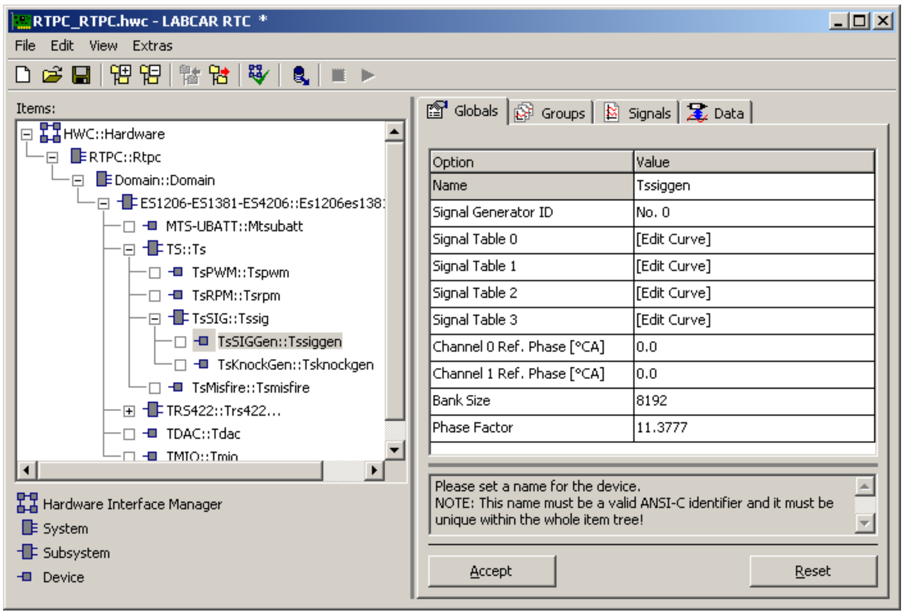


Fig. 28-13 The "Globals" Tab of the TsSIGGEN Device

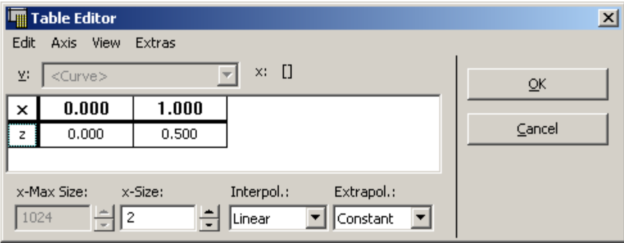
The "Globals" tab of the TsSIGGen device contains the configuration options for one each of the three arbitrary signal generators of the TS board. They can be configured independently of each other, each in its own device.

Signal Generator ID

This option is used to select one of the three signal generators on the TS board.

Signal Table 0-3

This option can be used to define the data samples for each signal bank. There are four signal banks for each signal generator. The Table Editor is opened by clicking "[Edit Curve]".



Now you can define any number of data samples (a maximum of 8192). Only ascending values are allowed on the x-axis. The value range is scaled so that it can be mapped to the range 0 to 8191. The additional data samples are determined by linear interpolation. This means only the data samples necessary for the signal trace need to be specified. Click OK to complete data-sample entry.

If the arbitrary signal generators are triggered by the speed signal generator, the value that is stored at the corresponding data sample of the selected memory bank of the signal generator is output at each increment of the counter. This ensures that the crankshaft signal is in sync with the signal generators.

Note

Changes to the signal traces are only active with the next trigger signal in the signal generators to guarantee a defined, crankshaft-synchronous switching of the signal form!

Channel 0 Ref. Phase

Defines a phase shift between 0 and 720° for channel 0 of the signal generator.

Channel 1 Ref. Phase

Defines a phase shift between 0 and 720° for channel 1 of the signal generator.

Bank Size

Defines the size of the memory bank of the signal generator. The preset value of 8192 should not be changed.

Phase Factor

Defines the number of cycle periods per degree of the crankshaft angle. "Phase Factor" multiplied by 720 gives the size of the memory bank. The preset value of 11.38 should not be changed.

28.3.13 Groups (TsSIGGen Device)

This section describes the signal-group-specific options of the TsSIGGEN device.

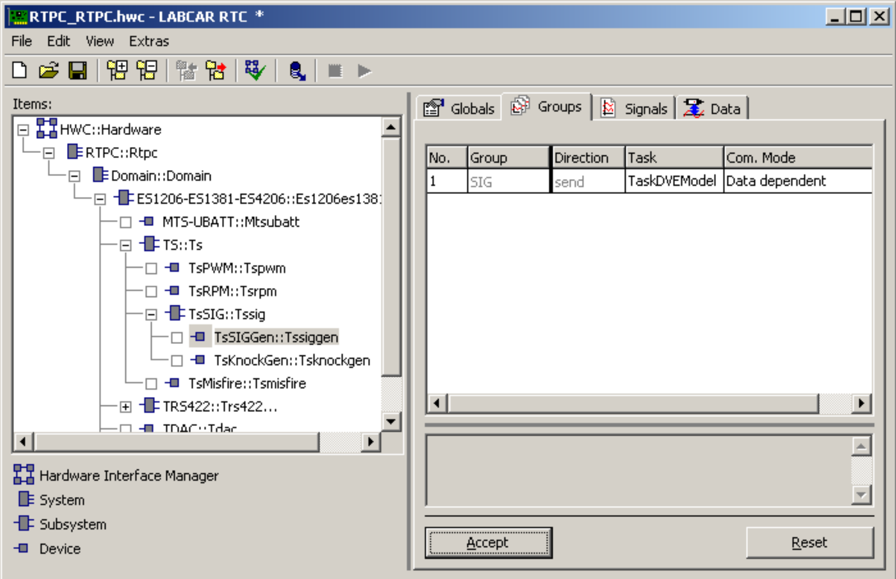


Fig. 28-14 The "Groups" Tab of the TsSIGGEN Device

There is only one "SIG" signal group for the TsSIGGen device. Its direction is *send*, i.e. it is transferred from the simulation target to the TS board. All signals of the signal group are used to configure the generator.

Com. Mode

This field specifies when a data transfer takes place between the simulation target and the TS board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

28.3.14 Signals (TsSIGGen Device)

This section describes the signals used to configure the arbitrary signal generator.

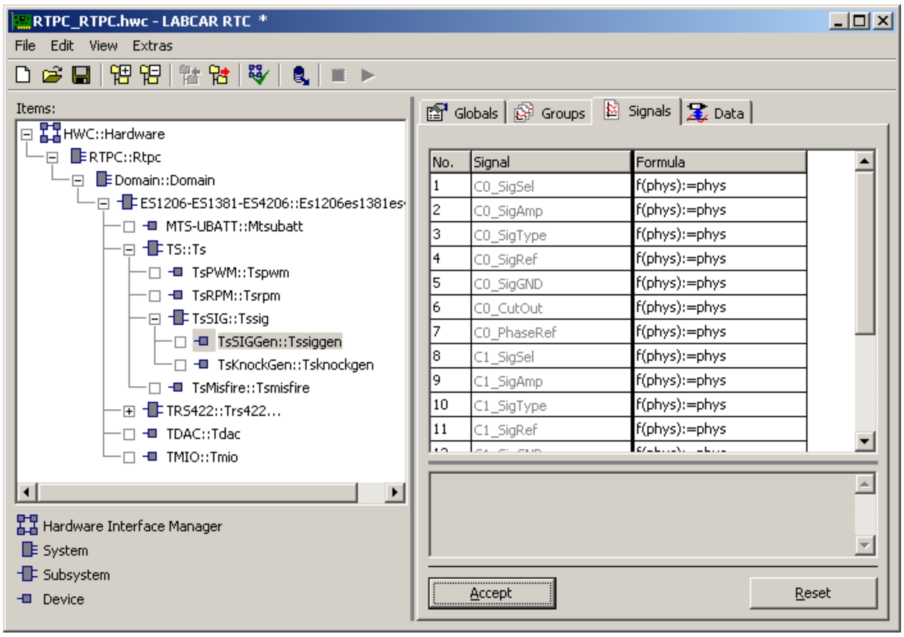


Fig. 28-15 The "Signals" Tab of the TsSIGGEN Device

The TsSIGGen device is configured with the following signals:

Signal	Direc-tion	Description
C0_SigSel	send	Defines which signal bank is output at channel 0
C0_SigAmp	send	Defines the amplification factor for the analog output of channel 0
C0_SigType	send	Defines the active output stage
C0_SigRef	send	Toggling between internal and external reference voltage for channel 0
C0_SigGND	send	Selection between battery and external ground for channel 0
C0_CutOut	send	Switches the signal generator on or off
C0_PhaseRef	send	Phase shift for channel 0. This phase shift is added to the phase shift defined under *Globals*; but, unlike the latter, it can be modified at runtime.
C1_...	send	For channel 1, the same signals are available as for channel 0.
TrigSource	send	Defines the trigger source for the signal generator. Both the speed generator and various external sources can be used as trigger.

Signal	Direc-tion	Description
ClkSource	send	Defines the clock source for the signal generator. The clock source can be the speed generator, an internal clock source, a 1 MHz clock source and various external sources.
LocalClkFreq	send	If the internal clock source has been selected, this signal sets its frequency.
TrgMode	send	Specifies the trigger mode. • Wait function (wait for trigger): Waits for the next trigger once the last data point has been filled with discrete values. The level of the last data point is issued during this time. • Continuous function: In this setting, the trigger has a reset function. Signal output is restarted with every trigger. If there is no subsequent trigger, signal output is restarted once the last data point is reached.

Data Types and Value Ranges

This section describes the possible data types and value ranges for each signal and the meaning of the various signal values. The same signals are available for both channel 0 (C0_...) and channel 1 (C1_...). The "TrigSource", "ClckSource" and "LocalClckFreq" are valid for both channels.

Signal	Data Type	Possible Values
C0_SigSel	int	0 to 3, corresponds to signal bank 0 to 3
C0_SigAmp	float	0 to 10 volts with internal reference voltage 0 to 1 with external reference voltage
C0_SigType	bool	true = analog false = digital (open collector switch)
C0_SigRef	bool	true = external reference voltage false = internal reference voltage
C0_SigGND	bool	true = battery ground false = external ground
C0_CutOut	bool	true = channel activated false = channel set to high impedance
C0_PhaseRef	float	0 to 720°
C1_...	As for channel 0	As for channel 0
TrigSource	int	0 to 12, see Tab. 28-2 on page 913

Signal	Data Type	Possible Values
ClckSource	int	0 to 4, see Tab. 28-3 on page 913
LocalClckFreq	int	0 Hz to 780 kHz
TrgMode	int	= 0: continuous function ≠ 0: wait for trigger function

The settings for the "TrigSource" option have the following meaning:

0	Speed generator of the master board
1	External trigger source 0
2	External trigger source 1
3	External trigger source 2
4	External trigger source 3
5 - 9	Not used
10	No trigger source
11	Not used
12	OR operation of the external trigger signals

Tab. 28-2 Settings of the "TrigSource" Option

The settings for the "ClckSource" option have the following meaning:

0	1 Mhz
1	Local clock source
2	External clock source 1
3	External clock source 2
4	Speed generator of the master board

Tab. 28-3 Settings of the "ClckSource" Option

28.3.15 TsKnockGen Device

The knock generator is used to simulate signals from knock sensors. An uncontrolled combustion of the fuel-air mixture results in pressure oscillations that are captured by solid-borne sound sensors. The engine control unit can detect knocking within a certain frequency range depending on the amplitude of the oscillations.

Engine knocking can be generated separately for each cylinder, either randomly or according to a fixed pattern. The duration, signal shape, and amplitude modulation (envelope) of the knock signal can be freely defined.

Fig. 28-16 shows an example of a knock signal ("knock package"). It is usually a high frequent sine wave signal which amplitude is modulated by a slowly varying envelope signal.

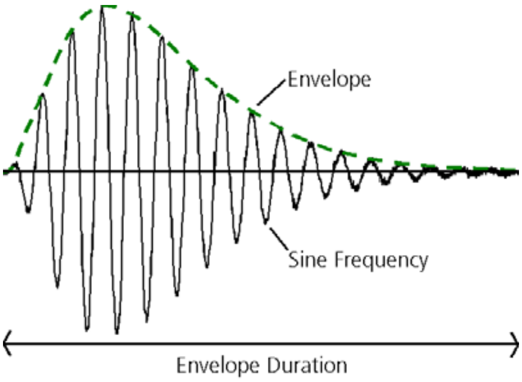


Fig. 28-16 Knock Signal (Solid Line) and Envelope (Dashed Line)

Note

The knock generator requires a specification of the number of available cylinders. This setting is done in the "Globals" tab of the TS subsystem (refer to section "Number of Cylinders" on page 899)

28.3.16 Globals (TsKnockGen Device)

This section describes the global options of the TsKnockGen Device.

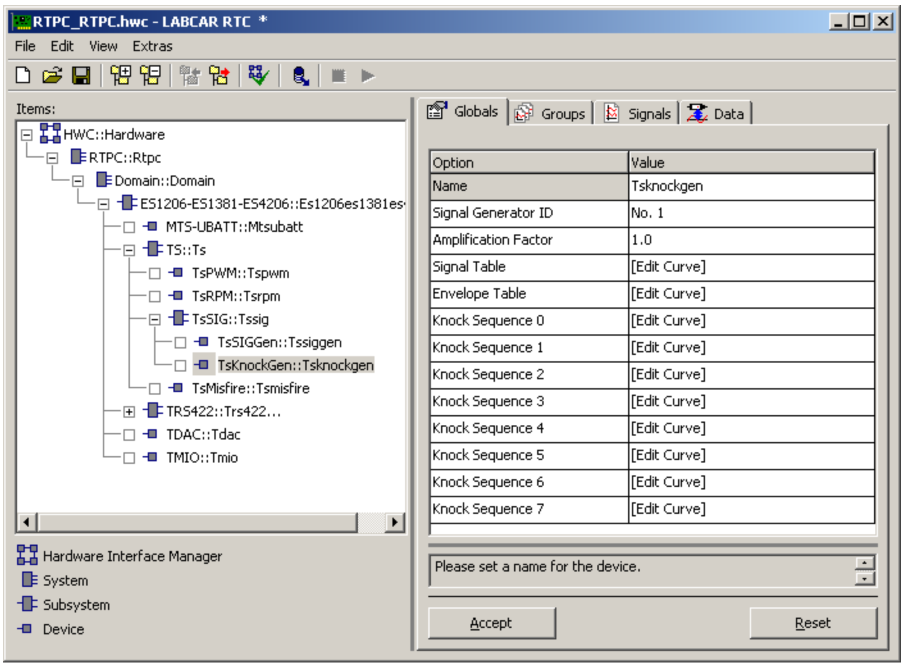


Fig. 28-17 The "Globals" Tab of the TsKnockGen Device

Signal Generator ID

The "Signal Generator ID" identifies which of the three TS board signal generators is used for knock signal generation.

This parameter cannot be edited in runtime mode.

Amplification Factor

The low-amplitude knock signals are susceptible to interference during their transmission to the control unit. To avoid interference the signals are amplified on their way from the TS board knock generator output to the breakout box. An attenuator usually installed in the breakout box compensates the amplification so that knock signals with correct amplitudes are applied to the control unit. Hence the "Amplification Factor" parameter has to be set equal to the attenuation factor of the breakout box attenuator.

The "Amplification Factor" parameter can be edited in runtime mode.

Signal Table

Clicking on the "Edit Curve" field opens the table editor for defining the signal shape and time period for the knock signal. The x-axis specifies the time in µs. The x-axis value of the first point must be 0.0. The minimum knock signal period is 3 µs. The z-axis specifies the normalized knock signal shape, the z-values have to be between -1.0 and 1.0.

The "Signal Table" can be edited in runtime mode.

Envelope Table

Clicking on the "Edit Curve" field opens the table editor for defining the envelope shape and duration of the knock signal. The x-axis specifies the time in µs. The x-axis value of the first point must be 0.0. The knock signal duration must be greater than or equal to the knock signal period. The z-axis specifies the normalized envelope shape, the z-values have to be in the range between 0.0 and 1.0.

The "Envelope Table" can be edited in runtime mode.

Knock Sequence 0 ... Knock Sequence 7

Clicking on the "Edit Curve" field opens the table editor to interactively specify the pattern for the occurrence of knock signals. The patterns are each 100 ignitions long. The x-axis has no meaning and is ignored. A z-value of 1 means that knocking is to be simulated for the current ignition, while 0 means that no knocking is to be simulated.

The "Knock Sequence" tables can be edited in runtime mode.

Tab. 28-4 summarizes the properties of the global knock generator RTIO parameters.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Signal Generator ID	uint32	No	TS board signal generator used for knock signal generation. Valid values: 1, 2, 3
Amplification Factor	real32	Yes	Amplification factor to compensate the effect of an external attenuator. Default value: 1.0 (no external attenuator installed) Valid range: 0.1 ... 10.0
Signal Table	1-dim. Table	Yes	Normalized knock signal shape. x-axis: time in µs z-axis: -1.0 ... 1.0
Envelope Table	1-dim. Table	Yes	Normalized envelope curve for the knock signal. x-axis: time in µs z-value range: 0.0 ... 1.0
Knock Sequence 0 ... Knock Sequence 7	1-dim. Table	Yes	Knock patterns. The patterns are each 100 ignitions long. z-value 1 means that knocking is to be simulated for the current ignition. z-value 0 means that no knocking is to be simulated. x-values are ignored.

Tab. 28-4 Global Parameters of the TsKnockGen Device

28.3.17 Groups (TsKnockGen Device)

This section describes the signal-group-specific options of the TsKnockGen Device.

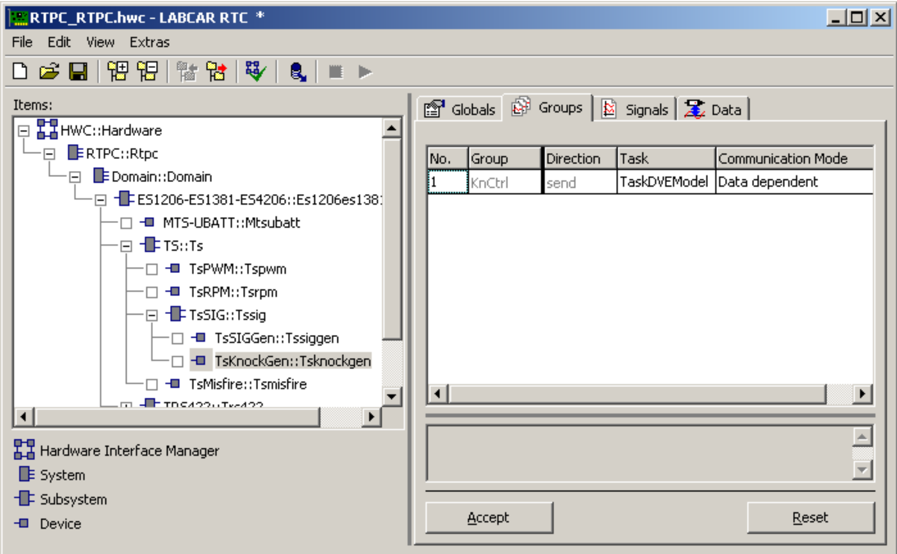


Fig. 28-18 The "Groups" Tab of the TsKnockGen Device

Communication Mode

Signal group communication mode.

Option "Every interval": The signal group is sent to the TS board every interval (task period).

Option "Data dependent": The signal group is sent to the TS board if at least one signal value has changed. The option "Data dependent" should be the preferred choice.

This parameter can be edited in runtime mode.

Tab. 28-5 summarizes the properties of the signal-group-specific RTIO parameters.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Communication Mode	uint8	Yes	Signal group communication mode 0: every interval (task period) 1: data dependent

Tab. 28-5 Signal-Group-Specific Parameters of the TsKnockGen Device

28.3.18 Signals (TsKnockGen Device)

This section describes the signals with which the knock generator is configured.

C0_SigRef, C1_SigRef

The "C0_SigRef" and "C1_SigRef" signals determine the source of the reference voltage of the D/A converter for knock generator channel 0 and 2 respectively. If the signal value is set to 0, the internal 10 V reference voltage is selected. If the signal value is set to 1, the D/A converter reference voltage is supplied externally.

Note

If you use the internal reference voltage, the "KnAmplSubLimit", "KnAmplOverLimit" and "KnNoiseAmpl" RTIO signal amplitudes refer to a reference voltage of 10 V. The signal values are limited to the range 0.0 ... 10.0/Amplification Factor.
If you use an external reference voltage, the "KnAmplSubLimit", "KnAmplOverLimit" and "KnNoiseAmpl" RTIO signal amplitudes indicate a factor which is multiplied with the external reference voltage. The signal values are limited to the range 0.0 ... 1.0/Amplification Factor.

C0_SigGND, C1_SigGND

The "C0_SigGND" and "C1_SigGND" signals define the reference potential for knock generator output channels 1 and 2 respectively. If the signal value is set to 0, the internal LABCAR ground is selected. If the signal value is set to 1, the reference potential is supplied externally.

C0_CutOut, C1_CutOut

The "C0_CutOut" and "C1_CutOut" signals are used to switch the respective knock generator output on and off. A signal value of 0 switches the channel output off; a signal value of 1 switches the channel output on.

KnEnable

The "KnEnable" RTIO signal is used to switch the knock generator on and off. A signal value of 0 switches the knock generator off; a signal value of 1 switches the knock generator on.

KnTrigger

A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during the time period covering the following 100 engine ignitions per cylinder.

The "KnTrigger" signal is only active when the knock generator is operated in trigger mode "Single Shot". The trigger mode is set using the "KnTriggerMode" RTIO signal (see page 918).

KnTriggerMode

The "KnTriggerMode" RTIO signal is used to control the knock generator trigger mode.

If the signal is set to 0 the knock generator operates in "Single Shot" trigger mode. A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during a time period covering the following 100 engine ignitions per cylinder. After that, knocking simulation is suspended until the next trigger occurs.

If the signal is set to 1, the knock generator operates in "Continuous" mode. In "Continuous" mode, knocking simulation is permanently active without trigger control. The "KnTrigger" RTIO signal is inactive in this trigger mode.

KnAmplSubLimit

Knock signal amplitude for controlled combustion. The "KnAmplSubLimit" RTIO signal sets the amplitude of the knock signal, if no knock signal is to be generated for the current ignition.

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnAmplOverLimit

Knock signal amplitude for uncontrolled combustion. The "KnAmplOverLimit" RTIO signal sets the amplitude of the knock signal, if a knock signal is to be generated for the current ignition.

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnNoiseAmpl

Many control units interpret multiple occurrences of knock signals with the same amplitudes as sensor disturbances. To avoid this, the simulated knock signals can be modulated in terms of their amplitude using a random generator.

The "KnNoiseAmpl" RTIO signal sets the amplitude with which the knock signal is modulated randomly. This RTIO signal is used to simulate statistical fluctuations during combustion and thus of the knock sensor signals.

Valid amplitude range:

Internal reference voltage selected: 0.0 ... 10.0/Amplification Factor.

External reference voltage selected: 0.0 ... 1.0/Amplification Factor

KnStartAngle

The "KnStartAngle" RTIO signal sets the start of the knock signal relative to the top dead center of the respective cylinder. The relative position of the top dead center is specified in degrees crankshaft angle (°CA). A positive sign indicates the start of the knock signal before the top dead center, a negative sign means after the top dead center.

Valid crankshaft angle range is -720 °CA ... 720 °CA.

KnMode0 ... KnMode7

The "KnMode0" to "KnMode7" RTIO signals specify the knock generator operating mode for the associated cylinder.

Disabled Mode: Knock signal generation is disabled for the associated cylinder, if the RTIO signal is set to 0.

Pattern Mode: The associated "Knock Sequence" table defines whether a knock signal is to be generated during the current ignition or not if the RTIO signal is set to 1.

Random Mode: A random generator controls whether a knock signal is to be generated during the current ignition or not if the RTIO signal is set to 2. The probability of knock generation is set using the associated "KnProbability" RTIO signal.

KnProbability0 ... KnProbability7

The "KnProbability0" to "KnProbability7" RTIO signals set the probability that uncontrolled combustion occurs during the current ignition. These RTIO signals are only active if the associated "KnMode" RTIO signal is set to "Random Mode".

The valid signal range is 0.0 to 1.0. A value of 0.0 means that uncontrolled combustion never occurs; a value of 1.0 means that uncontrolled combustion occurs during every ignition of the associated cylinder.

Note

The indices 0 to 7 appended to RTIO signals "KnMode" and "KnProbability" indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 28-6 summarizes all signals of the TsKnockGen device.

RTIO Signal	Data Type	Comment
C0_SigRef C1_SigRef	uint8	D/A converter reference voltage for knock generator output channel 0 and output channel 1. 0: internal 10 V reference voltage 1: external reference voltage
C0_SigGND C1_SigGND	uint8	Reference potential for knock generator output channel 0 and output channel 1. 0: internal LABCAR ground 1: external ground
C0_CutOut C1_CutOut	uint8	Switch on / off knock generator output channel 0 and output channel 1. 0: output switched off 1: output switched on
KnEnable	uint8	Enable / disable knock generator. 0: disabled 1: enabled
KnTrigger	uint8	Knock generator trigger signal. A 0 → 1 transition of the "KnTrigger" RTIO signal triggers knocking simulation during the time period covering the following 100 engine ignitions per cylinder. Only active in trigger mode "Single Shot".
KnTriggerMode	uint8	Knock generator trigger mode. 0: "Single Shot" trigger mode. 1: "Continuous" mode.

RTIO Signal	Data Type	Comment
KnAmplSubLimit	real32	Knock signal amplitude for controlled combustion. Valid value range: Internal reference voltage: 0.0 ... 10.0/Amplification Factor External reference voltage: 0.0 ... 1.0/Amplification Factor
KnAmplOverLimit	real32	Knock signal amplitude for uncontrolled combustion. Valid value range: Internal reference voltage: 0.0 ... 10.0/Amplification Factor External reference voltage: 0.0 ... 1.0/Amplification Factor
KnNoiseAmpl	real32	Amplitude (in volts) with which the knock signal is modulated randomly. Valid value range: Internal reference voltage: 0.0 ... 10.0/Amplification Factor External reference voltage: 0.0 ... 1.0/Amplification Factor
KnStartAngle	real32	Start of the knock signal relative to the top dead center of the respective cylinder. A positive sign indicates the start of the knock signal before the top dead center, a negative sign means after the top dead center. Valid crankshaft angle range: -720 °CA ... 720 °CA.
KnMode0 ... KnMode7	real32	Knock generator operating mode for the associated cylinder. 0: Disabled 1: Pattern mode 2: Random mode
KnProbability0 ... KnProbability7	real32	Probability that uncontrolled combustion occurs during the current ignition. Only active in knock mode "Random". Valid range: 0.0 (no knocking) ... 1.0 (always knocking)

Tab. 28-6 Signals of the TsKnockGen Device

28.3.19 TsMisfire Device

The misfire generator simulates a drop in engine speed after misfiring and is thus suitable for testing ECUs that recognize misfiring thanks to the drop in engine speed that follows.

Fig. 28-19 shows the connection between misfiring and a drop in engine speed.

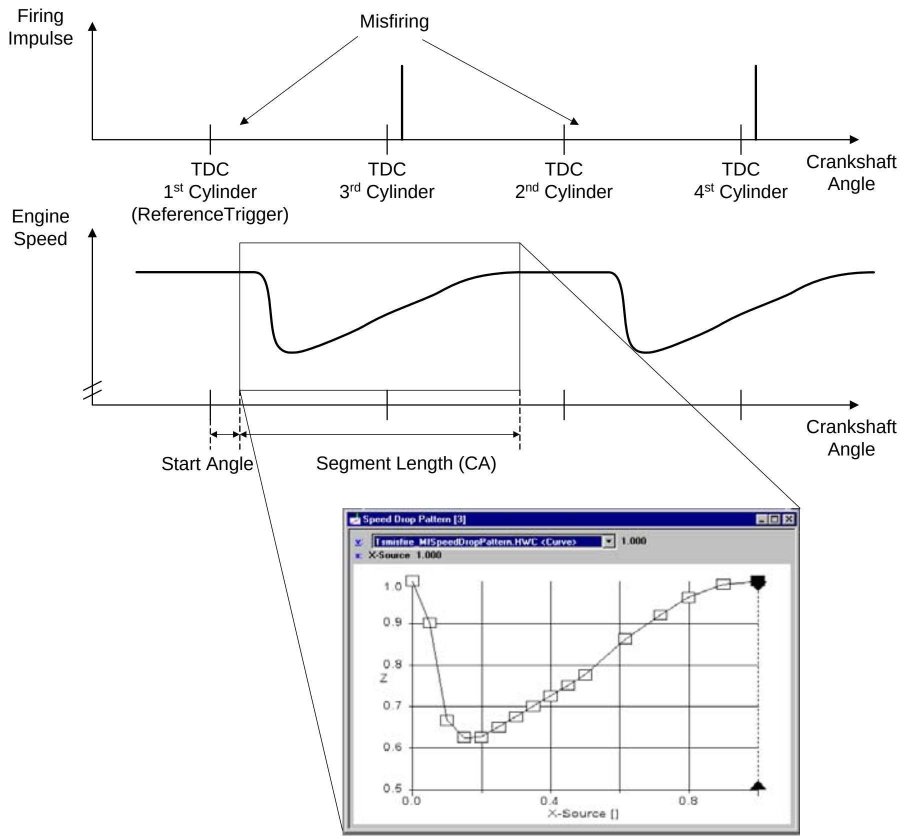


Fig. 28-19 Misfire and Drop in Engine Speed

Misfiring can be simulated for each individual cylinder for a maximum of twelve cylinders. Misfiring can be produced randomly with a probability that can be specified or according to a certain pattern specified separately for each cylinder. These patterns have a maximum length of 100 ignitions per cylinder and can be repeated as often as required.

It is also possible to specify an individual factor for each cylinder which specifies the drop in engine speed for each individual cylinder.

Note

The misfire generator requires a specification of the number of available cylinders. This setting is done in the "Globals" tab of the TS subsystem (refer to section "Number of Cylinders" on page 899)

28.3.20 Globals (TsMisfire Device)

This section describes the global options of the TsMisfire Device.

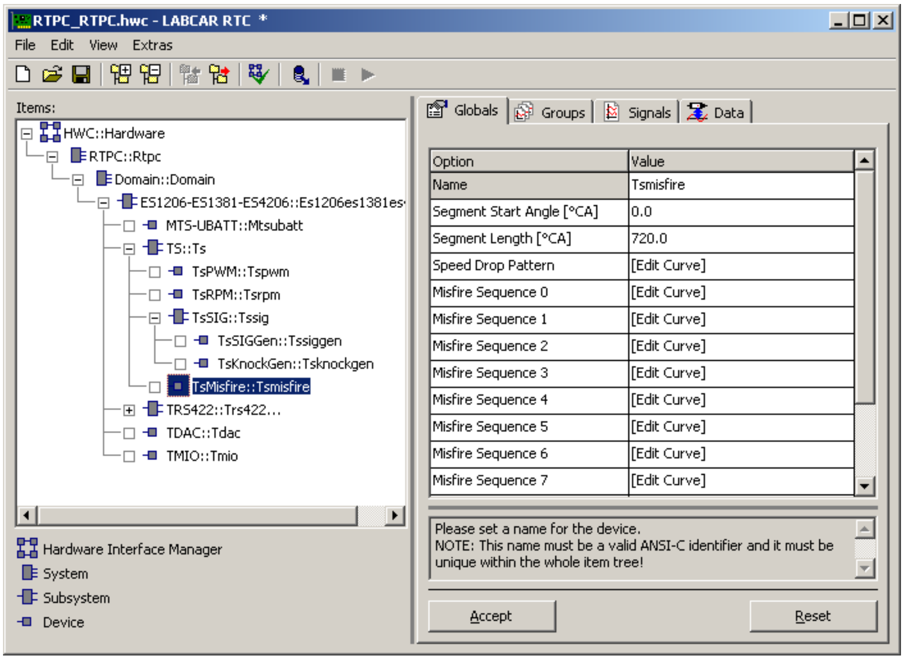


Fig. 28-20 The "Globals" Tab of the TsMisfire Device

Segment Start Angle [°CA]

The "Segment Start Angle" parameter is used to define the angle between the top dead center of the cylinder and the start of the drop in engine speed due to misfiring (see Fig. 28-19 on page 922).

The "Segment Start Angle" parameter is defined in °CA and can be set to values between -720.0 °CA and 720.0 °CA.

This parameter can be edited in runtime mode.

Segment Length [°CA]

The "Segment Length" parameter defines how long (in °CA) the drop in engine speed lasts (see Fig. 28-19 on page 922).

The "Segment Length" parameter is defined in °CA and can be set to values between 0 °CA and 720.0 °CA.

This parameter can be edited in runtime mode.

Speed Drop Pattern

Clicking on the "Edit Curve" field opens the table editor for defining the speed drop characteristic after misfiring.

The x-axis range does not have any meaning. The duration of the speed drop in °CA is defined using the "Segment Length" parameter (see Fig. 28-19 on page 922).

The z-axis defines the engine speed drop. z-values between 0.0 and 2.0 are allowed. A value of 0.5, for example, means that the engine speed is halved.

In general, engine speed during misfire simulation is determined by multiplying the engine speed prior to misfire with the factor for the engine speed drop (z-value).

The changes in engine speed caused by consecutive misfirings can overlap. For example, overlapping can occur if for a 4-cylinder engine the "Segment Length" > 180 °CA.

The "Speed Drop Pattern" table can be edited in runtime mode.

Misfire Sequence 0 ... Misfire Sequence 11:

Clicking on the "Edit Curve" field opens the table editor to specify the pattern for the occurrence of misfiring. The patterns are each 100 ignitions long. The x-axis has no meaning and is ignored.

A z-value of 1 means that misfiring is to be simulated for the current ignition, while a z-value of 0 means that no misfiring is to be simulated for the current ignition.

The "Misfire Sequence nn" tables can be edited in runtime mode.

Note

The indices 0 to 11 appended to the "Misfire Sequence" RTIO parameters indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 28-7 summarizes the properties of the global parameters of the TsMisfire device.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Segment Start Angle	real32	Yes	Specifies the angle between the top dead center (reference trigger) and the start of the drop in engine speed due to misfiring. Valid range: -720.0 °CA ... 720.0 °CA
Segment Length	real32	Yes	Specifies how long (in °CA) the drop in engine speed after misfiring lasts. Valid range: 0 °CA ... 720 °CA
Speed Drop Pattern	1-dim. Table	Yes	Engine speed drop characteristic after misfiring. z-value range: 0.0 ... 2.0 The x-values have no meaning.
Misfire Sequence 0 ... Misfire Sequence 11	1-dim. Table	Yes	Misfiring patterns. The patterns are each 100 ignitions long. z-value = 1: misfiring is to be simulated for the current ignition. z-value = 0: misfiring is not to be simulated for the current ignition. The x-values have no meaning.

Tab. 28-7 Global RTIO Parameters of the TsMisfire Device

28.3.21 Groups (TsMisfire Device)

This section describes the signal-group-specific options of the TsMisfire Device.

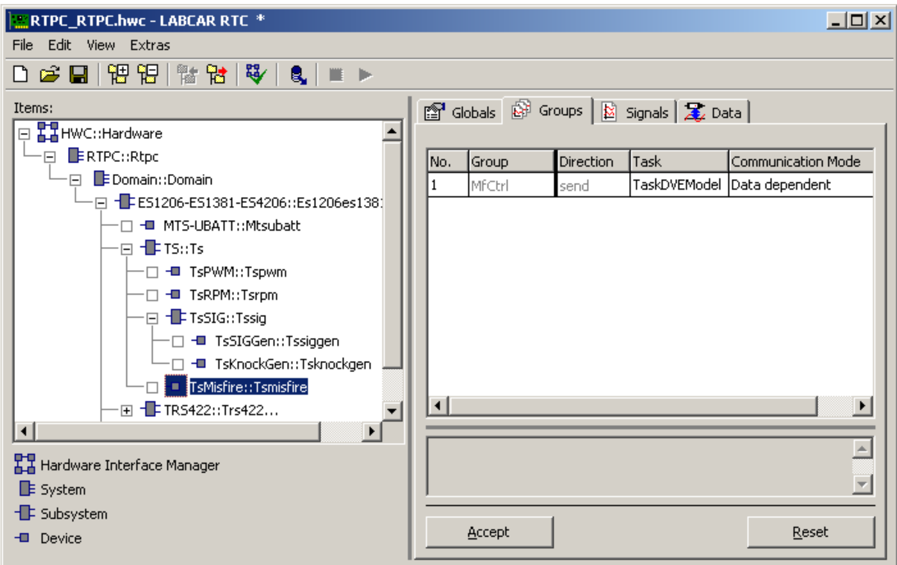


Fig. 28-21 The "Groups" Tab of the TsMisfire Device

Communication Mode

Signal group communication mode.

Option "Every interval": The signal group is sent to the TS board every interval (task period).

Option "Data dependent": The signal group is sent to the TS board if at least one signal value has changed. The option "Data dependent" should be the preferred choice.

This parameter can be edited in runtime mode.

Tab. 28-8 summarizes the properties of the signal-group-specific RTIO parameters.

RTIO Parameter	Data Type	Editable in Runtime Mode	Comment
Communication Mode	uint8	Yes	Signal group communication mode 0: every interval (task period) 1: data dependent

Tab. 28-8 Signal-Group-Specific Parameters of the TsMisfire Device

28.3.22 Signals (TsMisfire Device)

The "Signals" tab lists the individual signals and the signal group belonging to each signal.

MfEnable

The "MfEnable" RTIO signal is used to switch the misfire generator on and off. A signal value of 0 switches the misfire generator off; a signal value of 1 switches the misfire generator on.

MfTrigger

A 0 → 1 transition of the "MfTrigger" signal triggers misfire simulation during the time period covering the following 100 engine ignitions (per cylinder).

The "MfTrigger" signal is only active when the misfire generator is operated in "Single Shot" trigger mode. The trigger mode is set using the "MfTriggerMode" signal.

MfTriggerMode

The "MfTriggerMode" RTIO signal is used to control the misfire generator trigger mode.

If the signal is set to 0, the misfire generator operates in "Single Shot" trigger mode.

A 0 → 1 transition of the 'MfTrigger' signal triggers misfire simulation during a time period covering the following 100 engine ignitions (per cylinder). After that, misfire simulation is suspended until the next trigger occurs.

If the signal is set to 1, the misfire generator operates in "Continuous" mode. In "Continuous" mode misfire simulation is permanently active. The "MfTrigger" signal is inactive in this trigger mode.

MfMode0 ... MfMode11

The "MfMode0" to "MfMode11" RTIO signals specify the misfire generator operating mode for the associated cylinder.

Disabled Mode (MfMode*nn* = 0): Misfire generation is disabled for the associated cylinder.

Pattern Mode (MfMode*nn* = 1): The associated "Misfire Sequence" table defines the misfiring pattern for the associated cylinder.

Random Mode (MfMode*nn* = 2): A random generator controls whether misfiring is to be generated during the current ignition or not. The probability of whether misfiring occurs or not is set using the associated "MfProbability" signal.

MfFactor0 ... MfFactor11

The "MfFactor0" to "MfFactor11" RTIO signals specify a correction factor for the engine speed drop after misfiring for the associated cylinder.

A factor of 0.7 means that only 70 % of the value defined for a drop in engine speed in the "Speed Drop Pattern" table is effective.

Valid value range: 0.0 ... 1.0.

MfProbability0 ... MfProbability11

The "MfProbability0" to "MfProbability11" RTIO signals set the probability that misfiring occurs during the current ignition.

These RTIO signals are only active if the associated "MfMode" signal is set to "Random Mode".

The valid value range is 0.0 to 1.0. A value of 0.0 means that misfiring never occurs; a value of 1.0 means that misfiring occurs during every ignition of the associated cylinder.

Note

The indices 0 to 11 appended to the "MfMode", "MfFactor" and "MfProbability" RTIO signals indicate the ignition sequence of the engine. The ignition sequence is not necessarily the same as the cylinder sequence.

Tab. 28-9 summarizes the properties of the RTIO signals of the TsMisfire Device.

RTIO Signal	Data Type	Comment
MfEnable	uint8	Enable / disable misfire generator 0: disable 1: enable
MfTrigger	uint8	Misfire generator trigger signal. A 0 → 1 transition triggers misfire simulation during the time period covering the following 100 engine ignitions (per cylinder). Only active in trigger mode "Single Shot"
MfTriggerMode	uint8	Misfire generator trigger mode 0: "Single Shot" trigger mode 1: "Continuous" trigger mode
MfMode0 ... MfMode11	uint8	Misfire generator operating mode for the associated cylinder 0: Disabled 1: Pattern mode 2: Random mode
MfFactor0 ... MfFactor11	real32	Correction factor for speed drop after misfiring for the associated cylinder. Valid values: 0.0 ... 1.0
MfProbability0... MfProbability11	real32	Probability that misfiring occurs during the current ignition. Only active in "Random" misfire mode. Valid range: 0.0 (no misfiring) ... 1.0 (always misfiring)

Tab. 28-9 RTIO Signals of the TsMisfire Device

28.4 TMIO - Digital Signal Acquisition and I/O

The TMIO board (Transputer Measuring and I/O board) primarily fulfills two basic functions in the LABCAR environment:

- acquisition and evaluation of digital ECU signals
- generation of switch signals (as ECU inputs)

A total of 30 hardware channels (0-29) are made available to the user that can all be used for signal acquisition. Some of these channels (0-15) also make it possible to generate signals, i.e. these channels can *either* be used to acquire signals *or* to stimulate ECU inputs.

The TMIO boards thus make the following possible:

- acquisition and evaluation of speed-synchronous signals, e.g. injection signals, ignition signals
- acquisition and evaluation of time-synchronous signals (usually PWM), e.g. exhaust-gas recirculation, canister purge valve
- acquisition and evaluation of static (switch) signals, e.g. fan control, fault indicator
- generation of static (switch) signals, e.g. air-conditioning switch, brake switch

28.4.1 Basic Principle of the Measuring Signal Determination

To attain as great a flexibility as possible when evaluating pending input signals, the TMIO board is based on the following basic principle:

First of all each input signal is prepared separately (analogously), then compared with a threshold value (that can be set in the software for every signal individually) to realize the conversion of the analog input signal into digital 0/1 or inactive/active information. The 32 bits obtained in this way (30 channels plus two internal diagnostic signals) are constantly checked by a hardware switch for changes, i.e. "edges" on one or more input bits are recognized. If at least one input signal changes from "active" to "inactive" or vice versa, the switching saves the current statuses of the 32 inputs and calls the current values of two integrated counters which specify the current time and the current crankshaft angle. This information is then transferred to a ring buffer using DMA (Direct Memory Access), i.e. controlled by hardware.

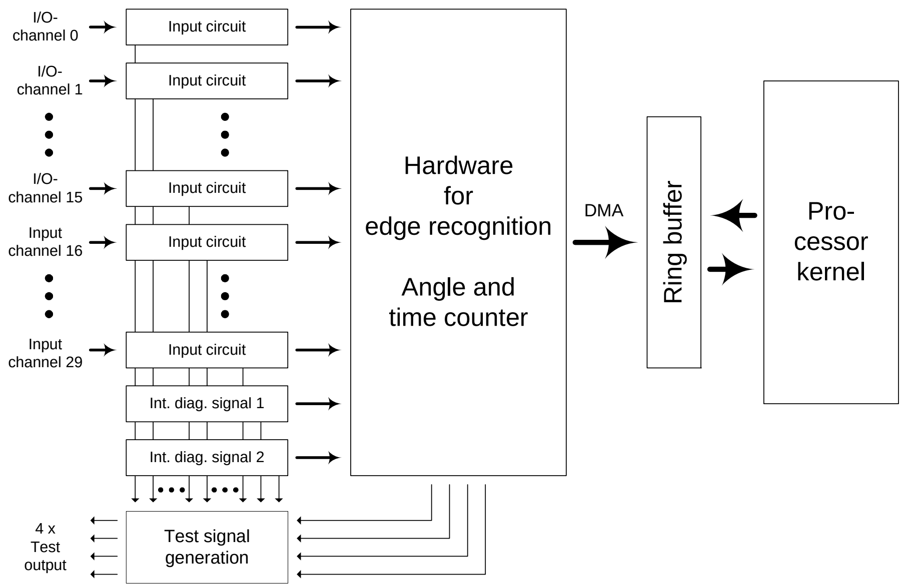


Fig. 28-22 TMIO Board Block Circuit Diagram

Note

The angle counter of the TIMO board is controlled by two signals that are generated by the TS board. The availability of a TS board is absolutely necessary for the determination of crankshaft-angle-dependent measuring signals.

The hardware of the TMIO plug-in board also acquires pulses, whose pulse width is smaller than the set resolution. Such short pulses, referred to as "spikes", can lead to the wrong interpretation with some evaluation procedures (for example of the pulse counting etc.). Pulses that are shorter than the resolution set are thus *not* taken into consideration by the evaluation software.

The firmware downloaded onto the transputer of the TMIO board can now read the generated values from the ring buffer and thus create the information, when and at what angle which input signals have changed and how. This information is used to generate all measuring signals that are of interest to the user, e.g. injection times, ignition times or duty cycles.

This basic concept of a hardware edge recognition with a secondary software preparation of the measuring signals means that the use of the integrated processor kernel depends directly on the number of recognized edges (i.e. the frequencies of the input signals). The integrated processor kernel is *only* intended to measure and evaluate input signals in the LABCAR environment. In addition, the computing performance of the processor is high enough to be able to determine the necessary measuring signals quickly enough even when using all available input signals without causing any real-time violations.

The advantage of this concept is obvious: the hardware measures changes and assigns them "angle" and "time stamps". The (sometimes very project-specific) evaluation takes place by software and is thus relatively easy to change or adapt. The firmware offers different forms of measuring signal determination (cf. the section "Signal Evaluation" on page 933). The concept used on the TMIO board makes easy adaptation possible: for the realization of new types of measurement, it is sufficient to adapt the TMIO firmware and update the RTIO integration of the board.

28.4.2 Measuring Signal Acquisition

The RTIO integration of the TMIO board makes a total of 64 freely configurable measuring signals available to the user, whereby each individual measuring signal can depend on any hardware channel. The allocation of hardware channels on the one hand and resulting measuring signals on the other is thus not defined by the TMIO board, i.e. when using a LABCAR for several projects, it is not absolutely necessary to always connect signals that have the same meaning to the same hardware channels of the TMIO board.

The generated 64 measuring signals provide the user with more than twice as many output variables as the board has I/O channels. This means that it is no problem to determine several measuring signals from a single input signal, such as for example frequency and duty cycle of a PWM signal. A double wiring of one and the same signal on different channels of the board is thus not necessary.

The following can be defined for each of these measuring signals (in addition to the base hardware channel) completely independently of all other measuring signals:

- how the pending signal is to be evaluated
- when and how it should be checked for missing pulses (*timeout recognition*) and
- whether the evaluation should be carried out permanently or only within a certain area of the crankshaft angle (*angle segmenting*).

For some measurements, you can continue to determine which pulse should be measured.

Note on the Number of Measure Channels

The TMIO board has memory for 75 measure channels, of which a maximum of 64 can be used/configured by the user. Please consult the explanation below as to why it is a maximum of 64 measure channels and not exactly 64.

Some measure procedures require a second measure channel internally for execution purposes. This measure channel is configured by the TMIO firmware and thus can no longer be used/configured by the user. If the number of measure channels with measure procedures requiring 2 channels exceeds 11, the 64 measure channels are no longer all available to the user.

Generally the number of measure channels the user can use is:

n = 64 - (number of measure channels with measure procedures requiring 2 channels - 11).

n_{max} = 64

The following measure procedures require a second internal measure channel:

- Hightime using H-Enable [µs]
- Hightime using L-Enable [µs]
- Hightime using H-Validate [µs]
- Hightime using L-Validate [µs]
- Lowtime using H-Enable [µs]
- Lowtime using L-Enable [µs]
- Lowtime using H-Validate [µs]
- Lowtime using L-Validate [µs]
- H-Time n-th Pulse (H-Ena) [µs]
- H-Time n-th Pulse (L-Ena) [µs]
- H-Time n-th Pulse (H-Val) [µs]
- H-Time n-th Pulse (L-Val) [µs]
- L-Time n-th Pulse (H-Ena) [µs]
- L-Time n-th Pulse (L-Ena) [µs]
- L-Time n-th Pulse (H-Val) [µs]
- L-Time n-th Pulse (L-Val) [µs]
- Time to first --/-- (H-En) [µs]
- Time to first --\-- (H-En) [µs]
- Time to first --/-- (L-En) [µs]
- Time to first --\-- (L-En) [µs]
- Time to last --/-- (H-En) [µs]
- Time to last --\-- (H-En) [µs]
- Time to last --/-- (L-En) [µs]
- Time to last --\-- (L-En) [µs]
- Step Count (Stepper Motor)

28.4.3 Signal Evaluation

The RTIO integration of the TMIO board supports a number of different possibilities of signal evaluation:

Level Acquisition

The TMIO board makes it possible to measure levels to acquire (static) switch signals. A measuring signal of the TMIO board configured accordingly depends directly on the 0/1 result of the threshold comparison for the dedicated hardware channel:

- acquisition of *active high* signals

  If the threshold value comparison returns a "0" as result, the measuring signal is 0.0, otherwise it is 1.0.

- acquisition of active *low signals*

  Unlike the evaluation of active high signals, the measuring signal is 1.0 if the comparison returns "0". The comparison result "1" results in the measuring signal 0.0.

Level Acquisition:	
Measuring:	"active high level" / "active low level"
Value range:	0.0; 1.0
Resolution:	
Calculation:	0.0 / 1.0, if input < threshold 1.0 / 0.0, if input > threshold
Calculation time:	Every rising and every falling edge
Updating:	In the next signal group time step

Measuring Angles

It is possible to measure angles both at rising and falling edges of the relevant input signal. A reference point can be defined for every measuring signal; the measuring signal is a result of the difference between this reference point and the current angle (cf. following figure).

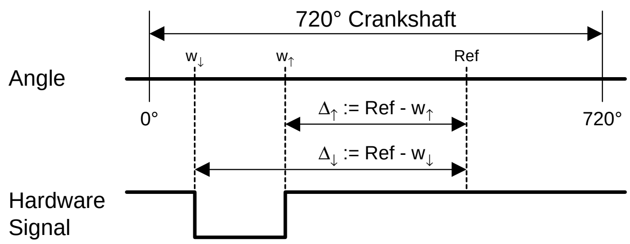


Fig. 28-23 Measuring Angles with the TMIO Board

Measuring angles is useful for example in the acquisition of ignition and injection angles. The reference point must be determined by the user in such a way that it corresponds to the top dead center (TDC) of the relevant cylinder. The firmware then returns "angle before TDC" as a positive value and "angle after TDC" as a negative one.

For measuring angles you can continue to determine which pulse should be measured within an angle segment.

Measuring Angles:	
Measuring:	"Rising edge" / "Falling edge"
Value range:	–720.0° … +720.0°
Resolution:	(720.0° / 8192) = 0.088°
Calculation:	Ref – w↑ / Ref – w↓
Calculation time:	↑ / ↓
Selection condition:	Angle segment nth pulse
Updating:	In the next signal group time step

High and Low Times

The TMIO board offers both a high-time and a low-time measurement to determine pulse durations as shown in the following figure.

Measuring high and low times can be qualified both via a specified angle segment as well as via the number of the pulse to be measured.

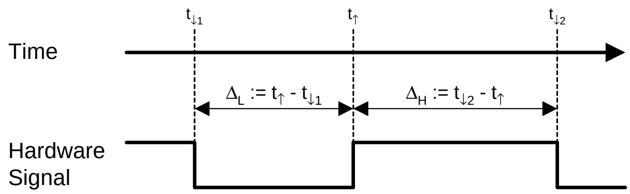


Fig. 28-24 Measuring Time with the TMIO Board

Measuring Time:	
Measurement:	"Low time" / "High time"
Value range:	0.0 µs … +∞
Resolution:	1.0 µs
Calculation:	t↑ – t↓ / t↓ – t↑
Calculation time:	↑ / ↓
Selection condition:	Angle segment nth pulse
Updating:	In the next signal group time step

Additive Times and Number of Pulses

In addition to the measuring of individual pulse widths, the TMIO board also makes it possible to determine "additive" times (Fig. 28-25 „Additive Time Measuring with the TMIO Board" shows additive low–time measuring). All high or low pulses are added together within a predefined angle segment (cf. the section "Angle Segmenting" on page 940) up to the highest pulse number defined by the user. A special form of this measuring is "pulse counting" whereby the times are not totaled but only the pulses that actually occurred.

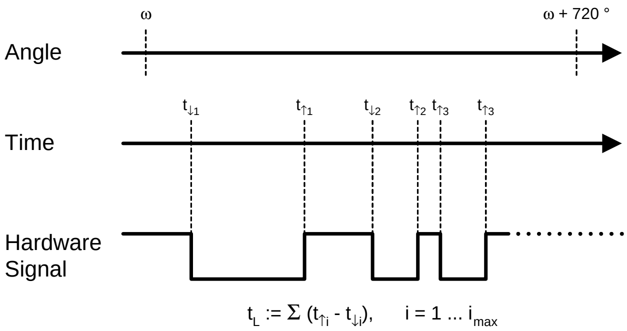


Fig. 28-25 Additive Time Measuring with the TMIO Board

Note

Using additive measurements within a released angle segment (cf. the section "Angle Segmenting" on page 940) it cannot be predicted whether further pulses will occur within the same segment. So it is only possible to update the measuring signal at the end of the segment with these measurements!

Additive Time Measuring:	
Measuring:	"Additive low time" / "Additive high time"
Value range:	0.0 µs … +∞
Resolution:	1.0 µs
Calculation:	Σ(t↑ – t↓) / Σ(t↓ – t↑)
Calculation time:	↑ / ↓
Selection condition:	Angle segment
Updating:	In the first signal group time step that follows the upper limit of the angle segment used

Pulse Counting:	
Measuring:	"Number of low pulses" / "Number of high pulses"
Value range:	0 … +∞
Resolution:	—
Calculation:	Number (↓↑) / number (↑↓)
Calculation time:	↑ / ↓
Selection condition:	Angle segment Continuous addition
Updating:	In the first signal group time step that follows the upper limit of the angle segment used

Difference: "Additive L/H-Time" and "Additive L/H-Time (extended)"

The difference between the two measurement modes is shown in Fig. 28-26.

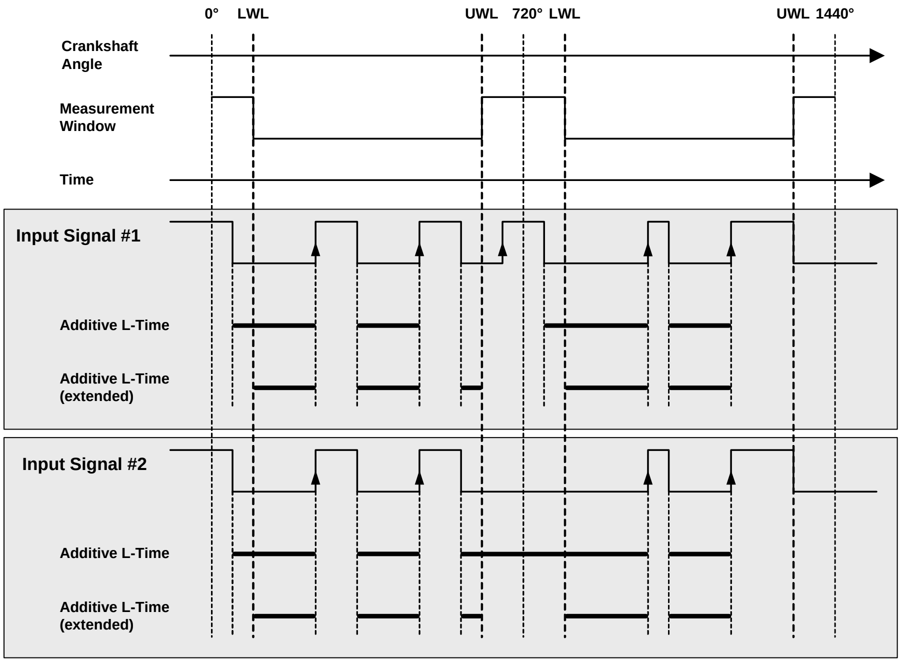


Fig. 28-26 Additive L-Time and Additive L-Time (extended)

"Additive low time" measures all low times the closing edges of which are within an angle window (regardless of when the low time began) whereas "additive low time (extended)" only adds the low times within the angular window (regardless of the position of the edges).

Frequencies and Cycle Times

The TMIO board makes it possible to measure frequencies and cycle times as shown in the following figure. The edge at which the measuring signal is to be determined can be specified as required by the user.

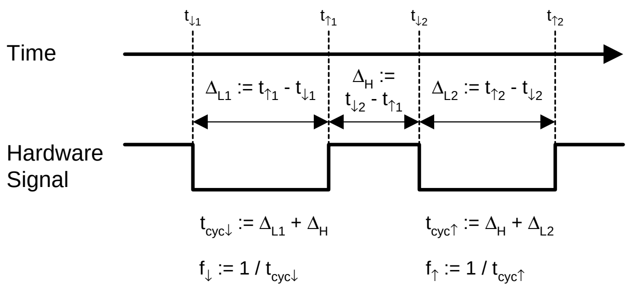


Fig. 28-27 Frequency Measuring with the TMIO Board

Cycle Time Measuring:	
Measuring:	"Cycle time↑" / "cycle time ↓"
Value range:	0.0 µs … +∞
Resolution:	1.0 µs
Calculation:	(t↑ – t↓) + (t↓ – t↑) / (t↓ – t↑) + (t↑ – t↓)

Cycle Time Measuring:	
Calculation time:	↑ / ↓
Selection condition:	Angle segment
Updating:	In the next signal group time step

Frequency Measuring:	
Measuring:	"Frequency ↑" / "frequency ↓"
Value range:	0.0 Hz ... +∞
Resolution:	Depending on cycle time determination
Calculation:	*1 / ((t↑ – t↓) + (t↓ – t↑)) / 1 / ((t↓ – t↑) + (t↑ – t↓))*
Calculation time:	↑ / ↓
Selection condition:	Angle segment
Updating:	In the next signal group time step

Duty Cycles

Duty cycles (e.g. of PWM signals) can be determined in different ways with the TMIO board, as shown in the following figure. With these measurements, the user can specify any edge at which the evaluation is to be executed.

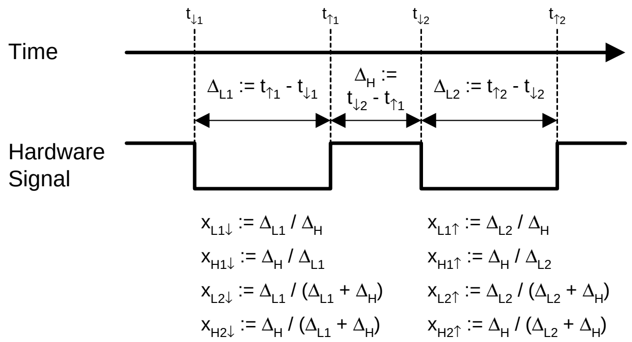


Fig. 28-28 Duty Cycle Determination with the TMIO Board

Determining the Duty Cycle (1):	
Measuring:	"Ratio L/H ↑" / "Ratio H/L ↑" / "Ratio L/H ↓" / "Ratio H/L ↓"
Value range:	0.0 ... +∞
Resolution:	Depends on the high or low time determination
Calculation:	*(t↓ – t↑) / (t↑ – t↓) / (t↑ – t↓) / (t↓ – t↑)* *(t↓ – t↑) / (t↑ – t↓) / (t↑ – t↓) / (t↓ – t↑)*
Calculation time:	↑ / ↑ ↓ / ↓
Selection condition:	Angle segment
Updating:	In the next signal group time step

Determining the Duty Cycle (2):	
Measuring:	"Ratio L/(L+H) ↑" / "Ratio H/(L+H) ↑" / "Ratio L/(L+H) ↓" / "Ratio H/(L+H) ↓"
Value range:	0.0 ... 1.0
Resolution:	Depends on the high or low time determination
Calculation:	$(t_{\downarrow} - t_{\uparrow}) / (t_{\uparrow} - t_{\uparrow})$ / $(t_{\uparrow} - t_{\downarrow}) / (t_{\uparrow} - t_{\uparrow})$ $(t_{\downarrow} - t_{\uparrow}) / (t_{\downarrow} - t_{\downarrow})$ / $(t_{\uparrow} - t_{\downarrow}) / (t_{\downarrow} - t_{\downarrow})$
Calculation time:	↑ / ↑ ↓ / ↓
Selection condition:	Angle segment
Updating:	In the next signal group time step

High and Low Times with External Signal Validation

In addition to these direct evaluations of a signal independent of all other inputs, the TMIO board also makes it possible to determine high or low times of a signal depending on a second input (can be specified by the user). This second input signal is interpreted as an "enable" or "validate" signal, as shown in the following figure in the example of validated low-time measuring:

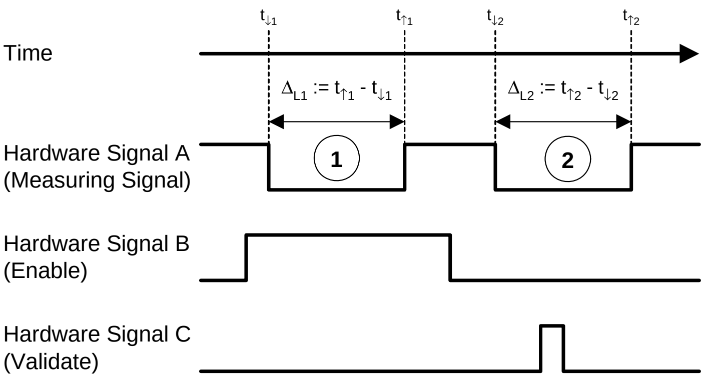


Fig. 28-29 Validated Time Measuring with the TMIO Board

If the TMIO board is configured in such a way that low times of an input signal A are to be measured, whereby a second input signal B is to be used as an "active high" enable signal, only the pulse marked 1 would be measured from the input signal in the above figure. More generally: when using a second input signal as "enable", only those pulses are measured which are **completely surrounded** by an active enable pulse.

If, on the other hand, a further input signal C is used as an "active high" validate signal, only the pulse marked 2 would be measured in the above example. More generally: when using a second input signal as "validate", only those pulses are measured which **completely surround** an active validate pulse.

The TMIO board makes it possible to use both enable and validate signals for both high and low time measurements. These can either be defined as "active high" or "active low":

Validated Time Measuring (1)	
Measuring:	"Low time with high or low enable" / "Low time with high or low validate"
Value range:	0.0 µs ... +∞
Resolution:	1.0 µs
Calculation:	t↓ – t↑, if surrounded by an active enable pulse / t↓ – t↑, if active validate pulse surrounded
Calculation time:	↑
Selection condition:	Angle segment Enable signal Validate signal nth pulse
Updating:	In the next signal group time step

Validated Time Measuring (2):	
Measuring:	"High time with high or low enable" / "High time with high or low validate"
Value range:	0.0 µs ... +∞
Resolution:	1.0 µs
Calculation:	t↑ – t↓, if surrounded by an active enable pulse / t↑ – t↓, if active validate pulse surrounded
Calculation time:	↓
Selection condition:	Angle segment Enable signal Validate signal nth pulse
Updating:	In the next signal group time step

28.4.4 Timeout Recognition

As already explained in the section "Basic Principle of the Measuring Signal Determination" on page 929, the TMIO board is controlled by the edges, i.e. the evaluation of the pending external signals takes place exclusively at their edges. This results in the generated measuring signals provided by the TMIO board also only being updated according to the edges. A signal that has already been evaluated by the board once, would then still "generate" the valid measuring signal last generated, even if it has in the meantime for example completely expired, i.e. has no more edges.

To be able to correctly interpret signals that have completely or partially expired, the TMIO board offers the possibility of timeout recognition. Each individual measuring signal generated by the firmware can be tested in certain intervals as to whether it was generated by new edges at the input signal. This test can take place in time intervals specified by the user or at current crankshaft angles specified by the user.

If it is determined in this kind of check that no edge has been detected by the hardware since the last check, the relevant measuring signal can be modified in two different ways. It is either set to a predefined timeout value or is determined according to the current status of the input signal. This is particularly necessary for the correct evaluation of duty cycles (an inactive input results here in the measuring signal 0.0 or 1.0 – depending on the input signal).

In total there are four possibilities of timeout recognition:

- checking in time intervals of `x` ms: measuring signal is predefined to `y` if a timeout occurs
- checking every 720° at `x`° crankshaft: measuring signal is predefined to `y` if a timeout occurs
- checking in time intervals of `x` ms: measuring signal depends on the status of the input signal ("low" or "high") if a timeout occurs
- checking every 720° with `x`° crankshaft: measuring signal depends on the state of the input signal ("low" or "high") if a timeout occurs

With the setting "Every 720 deg CA", a standing engine would not lead to a timeout signal because the crankshaft angle does not change. To help out with this problem, speeds smaller than 10 (1/min) are interpreted as an engine that is not moving and the relevant measuring signals are set to "Timeout".

28.4.5 Angle Segmenting

Speed-synchronous signals often have to be evaluated differently for different angle segments (and independently of one another) (e.g.: ignition distribution). This is taken into account on the TMIO board by the fact that an angle segment (in which the evaluation of the signal has to take place) can be defined for every measurement specified. All edges outside this segment are ignored in the calculation of the measuring signal.

The angle segment that can be defined separately can be anywhere within a 720° period for every measuring signal, i.e. segments that exceed the 0° limit can easily be set. A signal as shown in the following figure can thus very easily be "divided" into the four specified measuring signals. For this purpose, it would only be necessary to reference to the same hardware channel in the configuration of four different measuring signals and to specify the upper and lower limits specified in the figure of the angle segment to be released in each case.

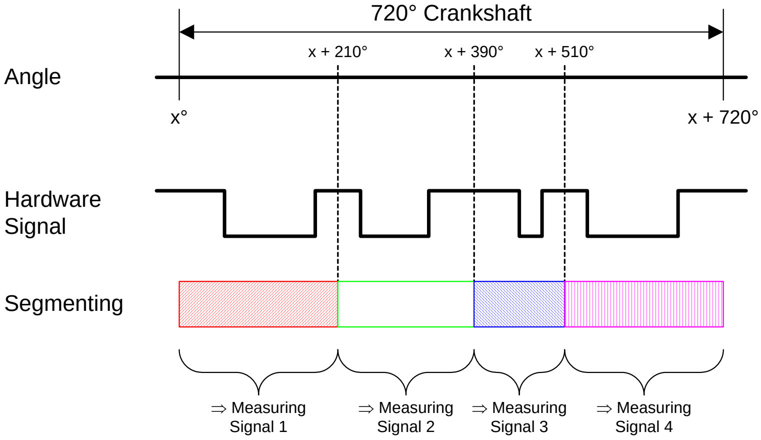


Fig. 28-30 Angle Segmenting with the TMIO Board

28.4.6 Globals (TMIO Device)

This section describes the global options of the TMIO device.

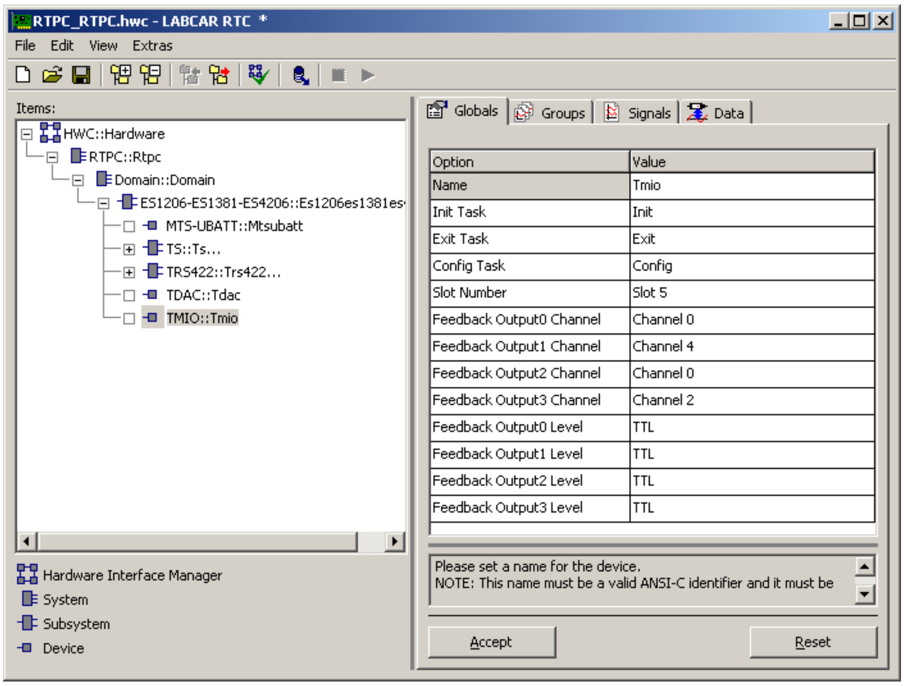


Fig. 28-31 The "Globals" Tab of the TMIO Device

Slot Number

Enter the slot of the LabCar2 signal box that holds the TMIO board. The slots are numbered from left to right starting with one.

Feedback Outputn Channel

This board has a total of four BNC jacks on the front panel for direct support of a search for possible causes of an error of the TMIO board. These enable four independent test signals to be directly connected, for example to an oscilloscope, for observation and evaluation purposes.

Each of these test signals gives the user the opportunity to issue one of 16 input signals of the board using a *multiplexer*. The compilation of these 16 possible "signal sources" is different for every test output.

Feedback Outputn Level

In addition to the selection of the actual signal, the user can also specify from which point of the signal flow the signal should be issued to the test jack. It can be tapped as an analog signal immediately before the threshold comparison whereby the test signal then, for technical reasons, has an output amplitude of 40% of the amplitude of the input signal.

Alternatively the user can tap the test signal immediately after the threshold comparison, i.e. as a TTL signal. The fact that the signal can be tapped both before and after the comparison means that direct control of the threshold set is very easy.

28.4.7 Groups (TMIO Device)

This section describes the signal-group-specific options of the TMIO device.

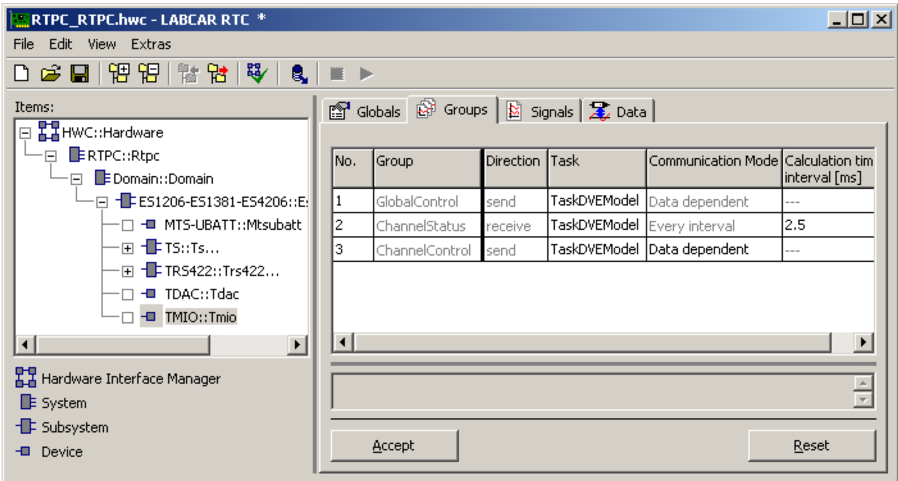


Fig. 28-32 The "Groups" Tab of the TMIO Device

The TMIO board has three signal groups.

- GlobalControl

  The "GlobalControl" signal group contains the crankshaft angle signal and the *Enable* signal with which the entire edge detection and edge evaluation of the TMIO board can be enabled and disabled. The direction of these signals is *send*, i.e. they are transferred from the simulation model to the TMIO board.

- ChannelStatus

  The "ChannelStatus" signal group contains all measuring and status signals of the board. The direction of these signals is *receive*, i.e. they are transferred from the TMIO board to the simulation model.

- ChannelControl

  The "ChannelControl" signal group contains the control signals used for the configuration of the trigger thresholds for the threshold comparisons, the switch signals for stimulating ECU inputs and the control signals for enabling and disabling hardware channels. The direction of these signals is *send*.

Communication Mode

This field specifies when a data transfer takes place between the simulation target and the TMIO board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

Calculation time interval [ms]

A process runs on the on-board transputer of the TMIO board which periodically executes data acquisition and status determination of the hardware channels and transfers these values to the simulation target in accordance with the option selected in the "Communication Mode" field: either always or only with changed values. This field specifies the time interval in which this process is calculated or activated.

28.4.8 Signals (TMIO Device)

This section describes the signals of the TMIO board and the configuration options of each signal.

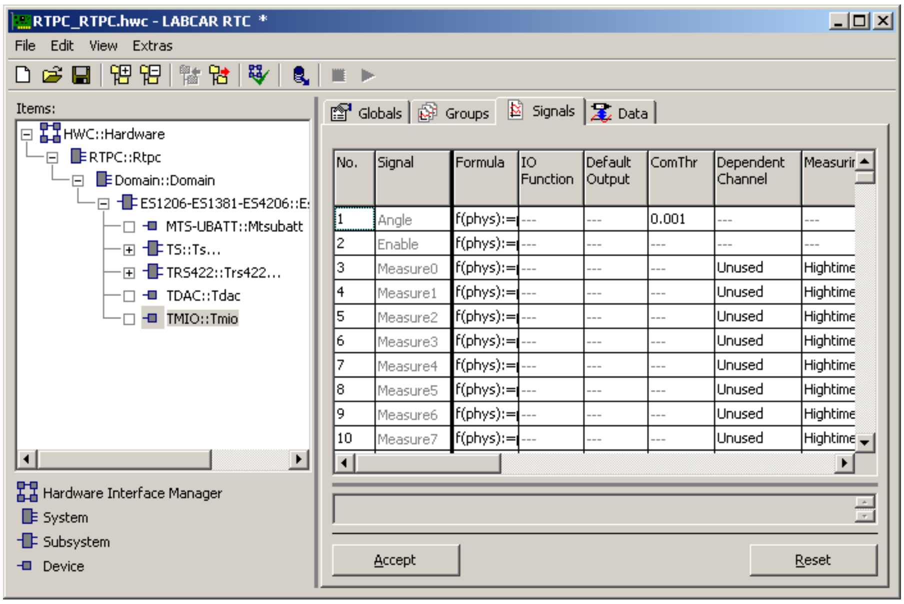


Fig. 28-33 The "Signals" Tab of the TMIO Device

The TMIO board has the following signals:

Signal	Direc-tion	Description
Angle	send	The TMIO board receives the current crankshaft angle via this signal. This information is absolutely necessary for angle segmenting and for timeout detection! The signal must be connected with the crankshaft-angle output signal of the TS board.
Enable	send	This signal is used to activate (true) or deactivate (false) the entire edge detection and evaluation of the TMIO board. A deactivation of the edge evaluation is to be recommended if the signal box is not supplied with the battery voltage U_{BATT}. In this situation the internal threshold comparison for all channels whose threshold is set relative to the battery voltage, will not work because the voltage is not available. An evaluation of the hardware signals concerned would therefore lead to errors in the simulation. The LABCAR signal box is supplied with battery voltage via battery node 4 of the UBATT-POW board. This is why, under normal circumstances, the *Enable* signal should always be connected to the signal for controlling battery node 4 of the MTS-UBATT board.
Measure0-63	receive	Measured values of the TMIO board
Timeout0-63	receive	The *Timeout* signals correspond directly to the *Measure* signals. For every *Measure i* signal, the *Timeout i* signal reports whether a timeout was detected during the measurement or not.
Trigger0	receive	*Trigger0* is a 32-bit value which contains information on the 32 measured values *Measure0* to *Measure31*. Each bit specifies whether the relevant measured value was updated in the last simulation step (the bit is set) or not (the bit is not set). Bit position 0 belongs to measured value *Measure0* and bit position 31 belongs to measured value *Measure31*.
Trigger1	receive	The same as *Trigger0*, but for the 32 measured values *Measure32* to *Measure63*. Bit position 0 belongs to *Measure32* and bit position 31 to measured value *Measure63*.

Signal	Direction	Description
ActiveLevel	receive	*ActiveLevel* is a 32-bit value which indicates the status of all 32 hardware channels at each stimulation step. If the threshold comparison for channel i (i = 0 ... 31) returns a "0", bit i is set from *ActiveLevel* to 0, otherwise to 1.
CurrentLimit	receive	*CurrentLimit* is a 16-bit value which indicates whether the current is limited to 1 A for the switches of I/O channels 0 to 15 at each stimulation step. If bit i (i = 0 ... 15) is set to 1, the current for I/O channel i is limited; if the bit is set to 0, there is no limitation.
OverCurrentP/N	receive	If one of these signals has the value "true", all channels of the TMIO board have been separated from the battery voltage because a total current exceeding 10 A has been detected on the board.
TrgThrSgl0-31	send	The signals *TrgThrSgl0* to *TrgThrSgl31* specify the thresholds for the threshold comparisons of the 32 hardware channels. The value set for a channel is, however, only taken into consideration if "Signal" is selected in the "Trigger Threshold Source" configuration field of the signal.
DisableHwChn0-31	send	Control signals for enabling or disabling hardware channels. A value not equal to 0 disables the relevant hardware channel. If measurements are carried out on a disabled channel, a timeout is detected if this is a measurement with timeout check.
OutputSetting0-15	send	For every I/O channel i of the TMIO board used as a switch or output, the setting of this switch is determined using the value of the *OutputSettingi* signal during simulation.

28.4.9 Data Types and Value Ranges

This section describes the possible data types and value ranges for each signal and the meaning of the various signal values of the TMIO board.

Signal	Data Type	Possible Values
Angle	float	0 to 720° crankshaft angle
Enable	bool	true = edge detection and evaluation is activated false = the entire edge detection and evaluation of the board is deactivated
Measure0-63	float	Depends on the type of measurement

Signal	Data Type	Possible Values
Timeout0-63	uint	0: **No timeout** detected when determining the relevant measured value. 1: **Timeout** detected when determining the relevant measured value. 2: The relevant measured value is **not used** at the moment (in accordance with the parameterization of the board).
Trigger0	uint32	
Trigger1	uint32	
ActiveLevel	uint32	
CurrentLimit	uint16	
OverCurrentP/N	bool	true = emergency stop enabled false = emergency stop disabled
TrgThrSgl0-31	float	Channel 0 - 7: 0.0 ... 1.0 (*U_{BATT}) Channel 8-15: 0.0 V ... 35.0 V Channel 16-23: 0.0 ... 1.0 (*U_{BATT}) Channel 24-29: 0.0 V... 35.0 V Channel Diag1 & Diag2: 0.0 V... 35.0 V
DisableHwChn0-31	bool	true = disabling of the relevant hardware channel false = enabling of the relevant hardware channel
OutputSetting0-15	uint	0 - the channel is high-impedance 1 - pull-up connection to battery voltage 2 - pull-down connection to battery ground 3 - direct connection to battery voltage 4 - direct connection to battery ground

28.4.10 Signal Configuration Parameters

IO Function

Defines the function and input connection of the 32 hardware channels. The following options are possible:

- Ipt

  The channel is used as an input and has neither a pull-up nor a pull-down connection.
- Ipt + PullUp

  The channel is used as an input and has a pull-up connection.
- Ipt + PullDown

  The channel is used as an input and has a pull-down connection.
- Output

  The channel is used as an output. This option is only available with hardware channels 0 to 15.

Default Output

This list box enables you to specify the standard setting after initialization for channels which are to be used as an output. The same options are available as for the *OutputSetting* signal which can change the output connection at runtime from the model. This setting is only relevant if the channel is used as an output.

ComThr

This setting is only of any significance if "Data dependent" was selected in the "Communication Mode" field of the relevant signal group. In this case "ComThr" specifies how much the relevant signal has to have changed in reference to its maximum value for a data transfer to take place between the simulation target and the TMIO board. If, for example, there is an angle signal with the maximum value of 720° and if a value of 0.01 is specified for the communication threshold, "ComThr", the angle signal has to change by 1%, i.e. 7.2°, for data to be transferred.

Dependent Channel

This list box enables you to specify which hardware channel (0-29, diag 1 or diag 2) the measured value specification should apply to. If the measured signal *Measurei* is not required in the model, select "Unused" (to avoid unnecessary processing time).

Measuring Procedure

Specifies which of the measurements described in the section "Signal Evaluation" on page 933 are to be executed. The notation in the list box has the following significance: `--/--` an evaluation of rising (↑) edges; `--\--` on the other hand means an evaluation of falling (↓) edges.

Measurement Method	Explanation
High/low time	High or low times of the signal within the specified angle segment
High/low time of nth pulse	High or low time of the nth pulse within the specified angle segment
Additive high/low time	Total of the high or low times of the signal within the specified angle segment up to the maximum allowed number of pulses
High/low time using H/L Enable	Total of the high or low times of the signal within the specified angle segment taking the Enable signal into account
High/low time using H/L Validate	Total of the high or low times of the signal within the specified angle segment taking the Validate signal into account
High/low time of the nth pulse using H/L Enable	High or low time of the nth pulse within the specified angle segment taking the Enable signal into account
High/low time of the nth pulse using H/L Validate	High or low time of the nth pulse within the specified angle segment taking the Validate signal into account

Measurement Method	Explanation
Frequency	Frequency of the rising or falling edges within the specified angle segment
Cycle time	Cycle time of the rising or falling edges within the specified angle segment
Duty factor	Duty cycle of low to high time or high to low time from the rising or falling edge within the specified angle segment
Duty cycle	Duty cycle of low time to period duration or high time to period duration from the rising or falling edge within the specified angle segment
Rising/falling edge	Angle measurement of the rising or falling edge of the nth pulse within the specified angle segment
Rising/falling edge of nth pulse	Angle measurement of the rising or falling edge of the nth pulse within the specified angle segment
Number of high/low pulses	Count of high or low pulses within the specified angle segment
Total number of high/ low pulses	Count of the high or low pulses since the last initial-ization, the last parameter configuration or the last timeout
Level (active high/low)	Threshold comparison with the voltage defined as "Trigger Level"

Enable Validate Channel

The "Enable Validate Channel" field is only of any significance for certain measurements. The Enable or Validate channel is specified in this field for validated time measurements, but is of no relevance for other measurements. For more details on special measurements please refer to the section "Signal Evaluation" on page 933.

Timeout Check

This setting specifies the type of timeout check for the relevant measured value. The following settings are possible:

- Inactive

  No timeout check.

- Intvl Predef

  Performs the timeout check at an interval defined in the "Timeout Interval [ms]" field. If a timeout is detected, the value defined in "Default Timeout Value" is issued.

- Intvl InpDep

  Performs the timeout check at an interval defined in the "Timeout Interval [ms]" field. The measured value output at timeout depends on the current state of the input signal. If, for example, a PWM signal is constantly active, 1.0 is issued; if it is constantly inactive, 0.0 is issued.

- CS Angle Predef

  Performs the timeout check every 720° crankshaft revolutions at the point defined by the "CS Angle Check Point" option. If a timeout is detected, the value defined in "Default Timeout Value" is issued.

- CS Angle InpDep

  Performs the timeout check every 720° crankshaft revolutions at the point defined by the "CS Angle Check Point" option. The measured value output at timeout depends on the current state of the input signal. If, for example, a PWM signal is constantly active, 1.0 is issued; if it is constantly inactive, 0.0 is issued.

Max Pulse Count

The "Max Pulse Count" is only of any relevance for certain measurements. "Max Pulse Count" defines the maximum number of pulses to be taken into account for additive measurement or pulse counts.

This is where you enter the number of the pulse when measuring a certain pulse within an interval (nth pulse).

Default Timeout Value

Default measured value that is output with certain settings in the `Timeout Check` field if a timeout occurs.

Timeout Interval [ms]

This field specifies the interval between timeout checks. This field is only of any significance if "Intvl predef" or "Intvl InpDep" are selected in the "Timeout Check" field.

CS Angle Check Point

This field specifies the crankshaft angle at which the timeout check is performed. This field is only of any significance if the options "CS Angle predef" or "CS Angle InpDep" are selected in the "Timeout Check" field.

CS Angle Lower Limit

This field specifies the lower limit of the angle segment free for measuring (cf. section 28.4.5 on page 940). This field is only of any significance for measurements with angle segmenting.

CS Angle Upper Limit

This field specifies the upper limit of the angle segment free for measuring (cf. section 28.4.5 on page 940). This upper segment limit is also used in additive time measurements and pulse counts for transferring the determined end value to the model (cf. section "Signal Evaluation" on page 933).

CS Angle Reference

This field defines the angle reference to which all angle measurements refer. This value is of no significance to all other measurements.

Trigger Threshold Source

There are two ways of defining the trigger thresholds for the threshold comparisons on the individual hardware channels. The trigger thresholds can be controlled in real time from the model or can be defined by the user with a configuration parameter in the RTIO user interface.

This field specifies whether the trigger threshold of the relevant hardware channel is controlled by the model ("Signal") or whether it is defined in the RTIO user interface ("Parameter").

Trigger Threshold Parameter

This field specifies the trigger threshold for the threshold comparison of the input signal pending at the channel if "Parameter" is selected in the "Trigger Threshold Source" field of the relevant hardware channel. Otherwise this field is of no significance.

28.5 TDAC - Analog Signal Acquisition and Generation

The TDAC board (Transputer D/A Converter Board) allows the generation and acquisition of analog voltage values and traces. It includes:

- 14 D/A converter channels
- 8 A/D converter channels
- 3 relays for free connections as determined by the user.

The A/D and D/A channels can directly input or output analog values or they can be used as comparators. The D/A channels can be operated both with internal and external reference voltage.

In the LABCAR environment, the TDAC board allows, for example, the generation of the engine temperature signal and the acquisition of the throttle angle signal.

28.5.1 Globals (TDAC Device)

This section describes the global options of the TDAC device.

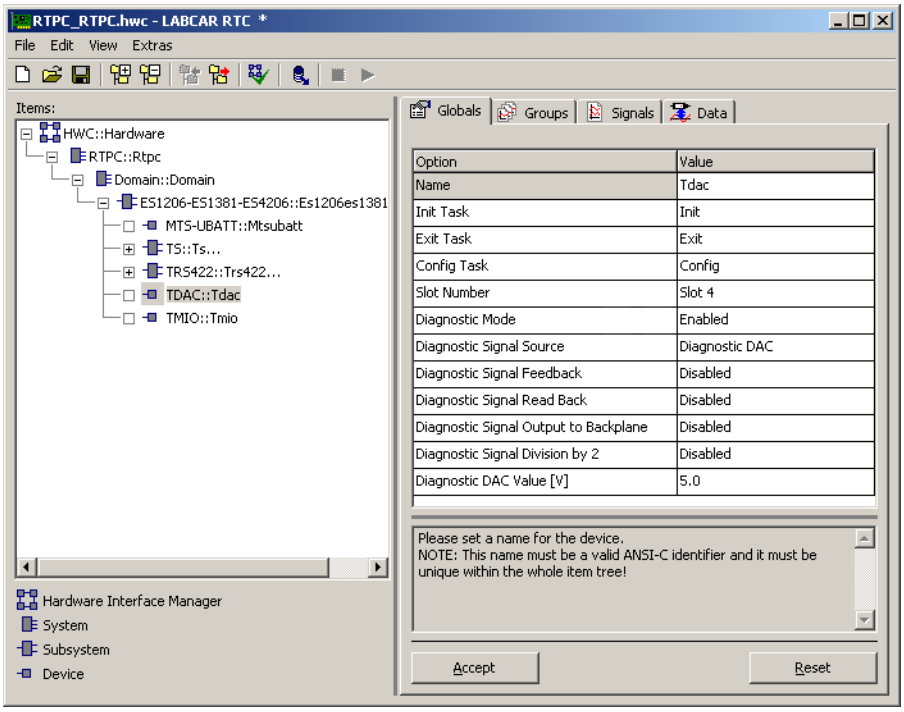


Fig. 28-34 The "Globals" Tab of the TDAC Device

Slot Number

Enter the slot of the LabCar2 signal box that holds the TDAC board. The slots are numbered from left to right starting with one.

Diagnostic Mode

For diagnostic purposes, the TDAC board has a fifteenth D/A converter. This is connected to different "signal receivers" via a multiplexer, to which **all** other D/A channels are connected:

- the output signal of the multiplexer is output at a separate jack specially intended for diagnostic purposes on the front panel of the board. This means that any D/A converter output can, for example, be connected directly to an oscilloscope.
- the diagnostic signal can also be issued on the backplane of the signal box using a switch so that, for example, it can be evaluated with a TS board or a second TDAC board.
- the diagnostic signal can also be switched directly (or halved by the amplitude) to the A/D converter using a switch. This enables the board to test itself to a certain degree.

The "enabled" setting activates generation of the internal diagnostic signal of the board.

Diagnostic Signal Source

Defines the output signal of the diagnostic multiplexer and thus the source of the diagnostic signal. Any D/A channel ("DAC Channel 0-13"), the additional diagnostic D/A converter ("Diagnostic DAC") or the board-internal ground ("GND") can be selected as source. If the diagnostic D/A converter is selected, its output voltage must be specified with the "Diagnostic DAC Value [V]" option.

Diagnostic Signal Feedback

If this field is set to "enabled", the output signal of the diagnostic multiplexer of the TDAC board is rerouted to the system for further processing (output on the backplane or read back via the A/D converter) and issued simultaneously on the front panel jack. If the field is set to "disabled", the output signal of the diagnostic multiplexer is not available to the system.

Diagnostic Signal Read Back

If this field is set to "enabled", the diagnostic signal is routed to the A/D converter and measured. Please note that while the diagnostic signal is being routed to the A/D converter, the 8 analog inputs cannot be measured!

By default this setting is set to "disabled". In this case, the analog signals of the 8 inputs are routed to the A/D converter.

Diagnostic Signal Output to Backplane

If the option is set to "enabled", the diagnostic signal is switched to the backplane with an activated diagnostic mode.

Diagnostic Signal Division by 2

If the option is set to "enabled", the measured value of the diagnostic signal is divided by 2.

Diagnostic DAC Value [V]

Defines the output voltage of the diagnostic D/A converter.

28.5.2 Groups (TDAC Device)

This section describes the signal-group-specific options of the TDAC device.

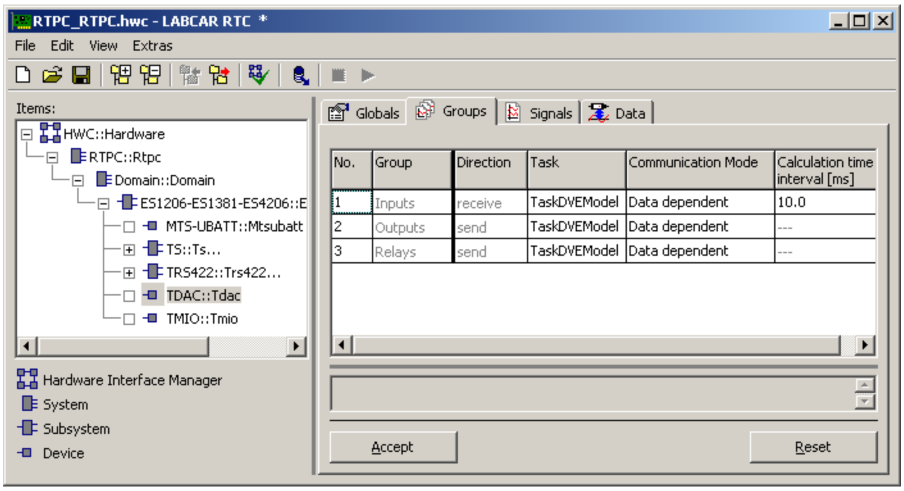


Fig. 28-35 The "Groups" Tab of the TDAC Device

The TDAC board has three signal groups for real-time communication:

- Inputs

  The "Inputs" group comprises the 8 A/D signals and the diagnostic signal of the board. The values of these signals are measured by the TDAC board and transferred to the simulation target. The transfer direction of the signals is therefore *receive*.

- Outputs

  The "Outputs" group comprises the 14 D/A signals of the board. These signals are set in the simulation model and transferred to the TDAC board. The transfer direction of the signals is therefore *send*.

- Relays

  The "Relays" group includes the three relays that are available to the user. These signals, too, have the *send* direction.

Communication Mode

This field specifies when a data transfer takes place between the simulation target and the TDAC board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

Calculation time interval [ms]

A process runs on the on-board transputer of the TDAC board which periodically determines the values of the 8 A/D converters and the diagnostic signal and transfers these values to the simulation target in accordance with the option selected in the "Communication Mode" field: either always or only with changed values. This field specifies the time interval in which this process is calculated or activated.

28.5.3 Signals (TDAC Device)

This section describes the signals of the TDAC board and their configuration parameters. Please see the section below for more information on the value ranges and the meaning of each value.

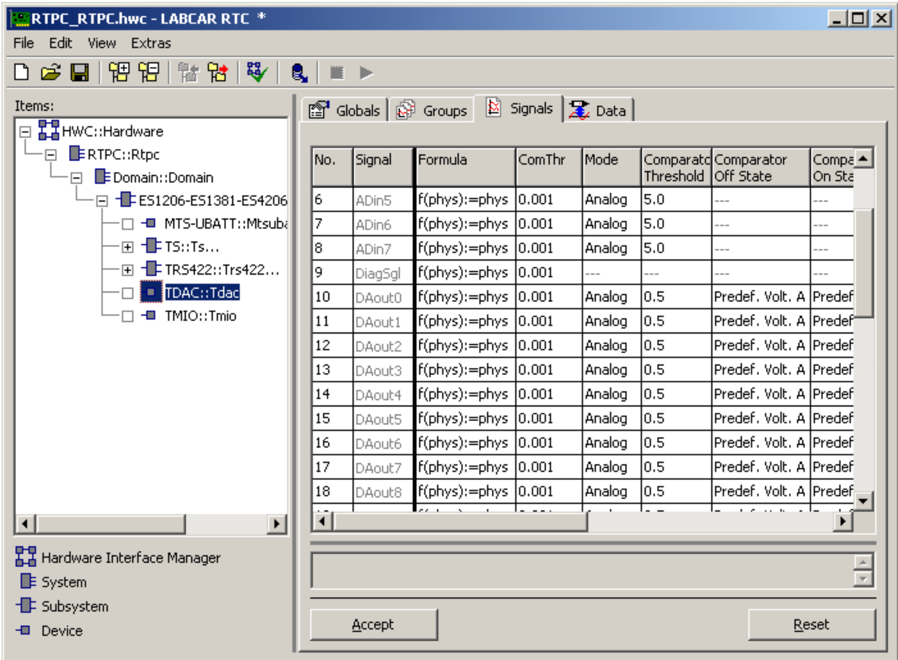


Fig. 28-36 The "Signals" Tab of the TDAC Device

The TDAC board has the following signals:

Signal	Direc-tion	Data Types	Description
ADin*n*	receive	real32, real64	Signal of the A/D converter input *n*
DiagSgl	receive	real32, real64	Diagnostic signal
DAout*n*	send	real32	Signal of the D/A converter output *n*
DACtrl*n*	send	uint8	Error simulation signal for D/A con-verter output *n*
Rel*n*	send	real32	Signal for relay output *n*

ComThr

This option is only of any significance when "Data dependent" has been selected in the "Communication Mode" field of the relevant signal group. In this case, "ComThr" specifies how much the relevant signal has had to change in reference to its maximum value for a data transfer to take place between the simulation target and the TDAC board. If, for example, a voltage signal has a maximum value of 10 V and a value of 0.01 is specified for the communication threshold, `ComThr`, the voltage signal has to change 1%, i.e. 100 mV, for a data transfer to take place.

Mode

This option is used to select between the three operating modes of a D/A output channel or an A/D input channel:

- Analog

  The "Analog" mode is used for issuing analog voltage traces on a D/A output channel or for measuring analog voltage traces on an A/D input channel.

- Comparator

  In "Comparator" mode, the channel works like a comparator. With D/A output channels, the *DAOutn* signal (n = 0 ... 13) of a D/A output channel is compared to a threshold defined in the "Comparator Threshold" field. If the value of the *DAOutn* signal is smaller than the threshold, the channel state defined in the "Comparator Off State" field is set. If the value of the *DAOutn* signal is greater than or equal to the threshold, the channel state defined in the "Comparator On State" field is set. With A/D input channels, the analog voltage pending at the channel input is compared to the threshold voltage defined in the "Comparator Threshold" field. If the voltage is smaller than the threshold voltage, the input signal *ADinn* (n = 0 ... 7) returns the value 0.0; if the voltage is greater than or equal to the threshold voltage, 1.0 is returned. Relay output channels can only be operated in comparator mode. If the value of the control signal *Reln* (n = 0, 1, 2) is smaller than the threshold, the channel state defined in the "Comparator Off State" field is set. If the value of the *Reln* signal is greater than or equal to the threshold, the channel state defined in the "Comparator On State" field is set.

- Disabled

  In "Disabled" mode, a D/A output channel is set to high impedance: an A/D input channel is completely disabled. This disabling is particularly useful if a hardware input of the TDAC board is not wired as the open, analog input could otherwise corrupt the measured values of the connected channels.

Comparator Threshold

Threshold value for the Comparator mode. The A/D input channels allow values between 0 and 10 V; the D/A output channels and relays allow relative values between 0 and 1.

Comparator Off State

Only relevant for D/A and relay output channels in Comparator mode. The possible settings for the D/A channels are:

- High Impedance

  The channel is set to high impedance.

- Predef. Volt. A

  The voltage set in the "DA Predef. Voltage A [V]" field is issued when the Comparator is in "offstate" mode.

- +Ubatt

  The channel is connected to battery voltage when the Comparator is in "offstate" mode.
- -Ubatt

  The channel is connected to battery ground when the Comparator is in "offstate" mode.

Possible relay states:

- Open

  The relay state "Open" corresponds to the Comparator state "offstate".
- Closed

  The relay state "Closed" corresponds to the Comparator state "offstate".

Comparator On State

Only relevant for D/A channels and relays in Comparator mode. The possible settings for D/A channels are:

- High Impedance

  The channel is set to high impedance.
- Predef. Volt. B

  The voltage set in the "DA Predef. Voltage B [V]" field is issued when the Comparator is in "onstate" mode.
- +Ubatt

  The channel is connected to battery voltage when the Comparator is in "onstate" mode.
- -Ubatt

  The channel is connected to battery ground when the Comparator is in "onstate" mode.

Possible relay states:

- Open

  The relay state "Open" corresponds to the Comparator state "onstate".
- Closed

  The relay state "Closed" corresponds to the Comparator state "onstate".

AD Samples

Only relevant for A/D input channels in Analog mode. This setting defines how many measured values of the analog input signal are used to created an average value that is used as an output value. The averaging of the measured values is used for noise and interference suppression. Permissible values are between 1 and 100. If 1 is selected as the value for the "AD Samples" option, there is no averaging. Please note that every sampling requires approx. 10 µs. High values for averaging on several channels will have a negative effect on the processing time.

DA Reference

For each of the 14 D/A output channels of the TDAC board, you can toggle between an internal reference voltage of 10 V and an external reference voltage specified by the user. This setting is only relevant for D/A output channels in Analog mode. The internal reference voltage is used for D/A output channels in Comparator mode.

DA Scale

The setting for the scaling factor `DA Scale` is only relevant for D/A output channels in Analog mode. If the channel is operated with the *internal* reference voltage, the significance of the output signal `DAout`*n* (n = 0, 1 ... 13) can be influenced by the value specified in this field. The output voltage of the D/A channel is specified for internal reference in accordance with the following equation:

```
voltage [V] := DAOutn * DA Scale
```

If, for example, "DA Scale" is set to 1, *DAoutn* specifies the output voltage directly in volts: the block input is therefore *absolute* in this case. If, on the other hand, 10 is specified for "DA Scale", the relevant value for *DAoutn* is somewhere between 0.0 and 1.0 (for voltages between 0 V and 10 V): the block input is therefore *relative*.

DA Predef. Voltage A [V]

Output value for the "offstate" state in Comparator mode.

DA Predef. Voltage B [V]

Output value for the "onstate" state in Comparator mode.

28.5.4 Value Ranges

This section describes the permissible data types data types and value ranges for each signal in the different operating modes.

ADin

The A/D channels can be operated in three modes:

- in "Disabled" mode, -1 is output. -1 is also output if the "Diagnostic Signal Read Back" field is set to "Enabled" in the `Globals` tab
- in "Analog" mode, the value range is 0 to 10 volts
- in "Comparator" mode, 0 is returned, if the measured voltage is below the threshold. If it is greater than or equal to the threshold, 1 is returned

DiagSgl

When the values of the 8 A/D converter inputs are read back (the "Diagnostic Signal Read Back" field is set to "Disabled" in the "Globals" tab), -1 is returned; otherwise the value range is from 0 to 10 volts.

DAout

In "Analog" channel mode, the value range of the DAout signals depends on whether the internal reverence voltage or an external one is used for the relevant D/A output channel.

- if the internal reference voltage is used, the following is true for the value range of the DAout signals
  `0 <= DAout <= 10.0 V / DA Scale`
  "DA Scale" is thus the scaling factor set in the "Signals" tab for the channel.
- if the external reference voltage is used, the output voltage of the D/A channel is calculated in accordance with the following equation:
  `voltage [V] := DAout * external reference voltage`
  The values for the DAout signals must therefore be between 0.0 and 1.0.

In the "Comparator" channel mode, the values for the DAout signals should be between 0.0 and 1.0.

In the "Disabled" channel mode, the values of the DAout signals are of no significance.

DACtrl

By setting the *DACtrl* signal, various error conditions can be simulated on the respective D/A channel. Depending on the value assigned to the signal, an error condition is simulated at the relevant output channel. The signal can take the following values:

Value	Description
0	Normal operation, no error simulation. The output signal results from the value of the DAout signal (depending on the channel configuration).
1	Sensor drop is simulated; the channel output is switched to high impedance
2	Short circuit against battery voltage is simulated
3	Short circuit against battery ground is simulated

During error simulation, i.e. if the DACtrl signal has the values 1, 2 or 3, the DAout signal of the respective channel is of no significance.

Rel

The values of the relay control signals should always be between 0.0 and 1.0.

28.6 TRS422

With the TRS422 board, it is possible to connect a maximum of 2 transputer modules (TRAMs), used as function modules in the LABCAR environment, with up to three different optional LABCAR components (VME extension, load box, console etc.). The symmetric standard interface, RS422A, is used as an interface. The LABCAR-OPERATOR Software supports the connection of one or more load boxes of these possibilities made possible by the hardware. The software support of a VME extension is no longer necessary in the VMEbus-based LABCAR system.

Transputer links of the TRAM modules, which can be inserted in both the TRAM slots, are routed to Sub-D connectors or jacks on the front panel of the TRS422 board to connect the load boxes. A connection to the T8VME board of a load box can thus be established using a serial transputer link cable. This T8VME board for connecting a load box will be referred to as the T8IO board to better distinguish it from T8VME boards which are used for the implementation of the parallel VMEbus on the serial transputer link bus.

28.6.1 Globals (TRS422 Subsystem)

This section describes the global options of the TRS422 board.

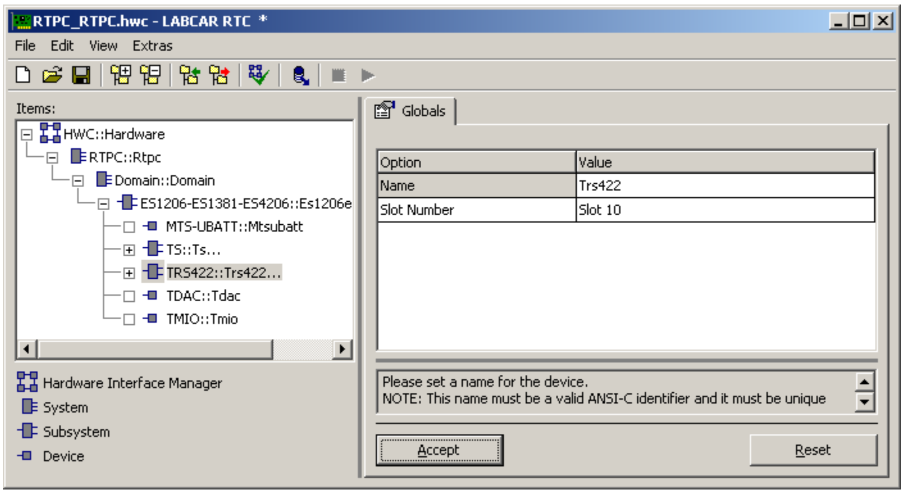


Fig. 28-37 The "Globals" Tab of the TRS422 Subsystem

Slot Number

Enter the slot of the LabCar2 signal box that holds the TRS422 board. The slots are numbered from left to right starting with one.

28.7 T8Module Subsystem

Up to two T8Module subsystems can be allocated to the TRS422 subsystem. These correspond to the piggyback modules physically inserted into the slots of the TRS422 board. It is of no significance whether these modules are TRAM or TCAN modules.

Load boxes which are connected using transputer links of the T8 modules, are allocated to the T8 modules in the hierarchy tree of the RTIO Editor as is explained below in the description of the T8IO subsystem.

28.7.1 Globals (T8Module Subsystem)

This section describes the global options of the T8Module subsystem.

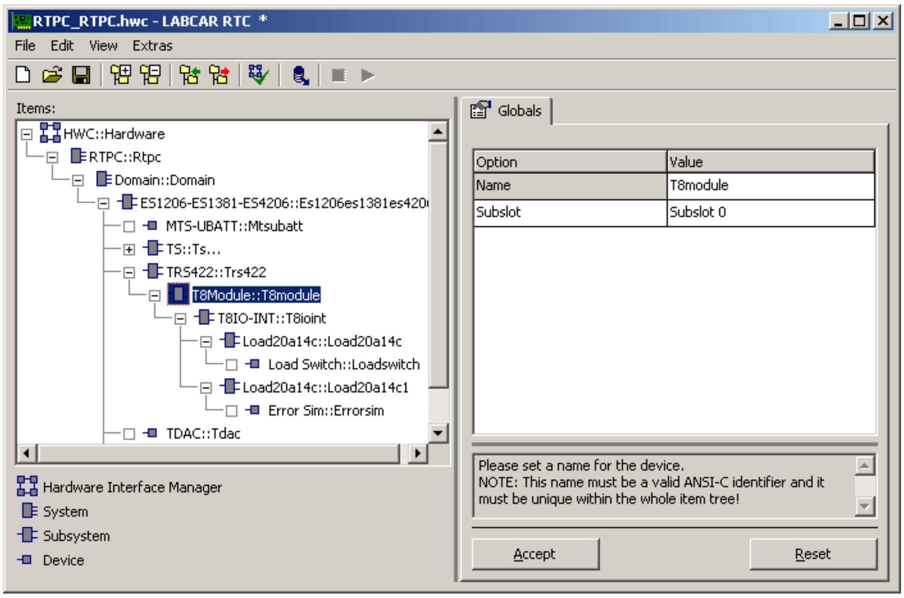


Fig. 28-38 The "Globals" Tab of the T8Module Subsystem

Subslot

Enter the slot number of the slot of the TRS422 board that holds the TRAM or TCAN module.

28.8 T8IO

The T8IO board is used to connect a load box with a VME backplane to a transputer-based signal box. The boards of the load box are located in the hierarchy tree of the RTIO Editor below the T8IO board and are controlled by it. In contrast to the previous software generation, the calculation of model parts on the T8IO board is no longer supported by LABCAR-OPERATOR.

28.8.1 Globals (T8IO Subsystem)

This section describes the global options of the T8IO subsystem.

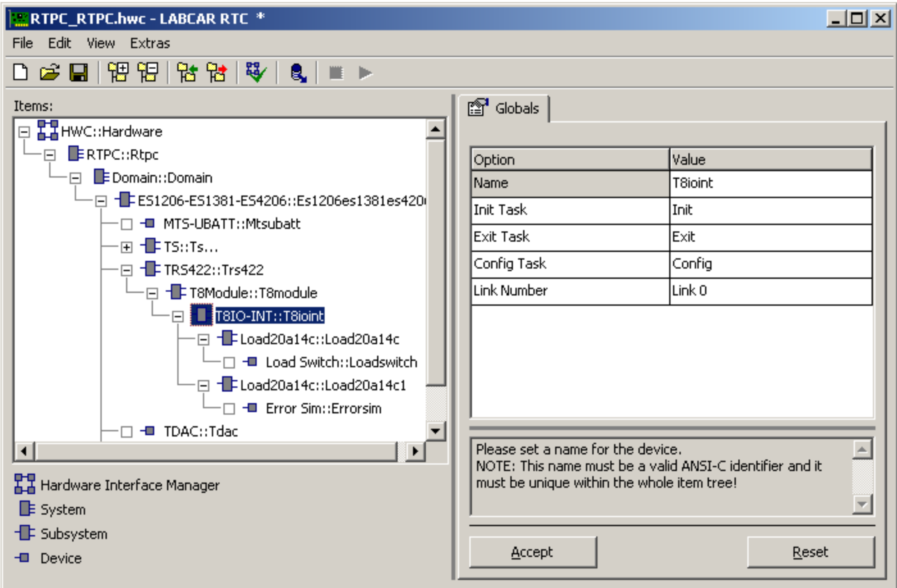


Fig. 28-39 The "Globals" Tab of the T8IO Subsystem

The Init Exit and Config tasks for the T8IO board are also valid for all load boards connected.

Link Number

Specify here via which link of the relevant T8 module the T8IO board is connected.

28.9 LOAD20A14C High-Current Load-Switching Board

The Load20A14C High-Current Load-Switching Board can be used in two alternative operating modes:

- as a toggle switch of ECU connections between different loads
- as a short-circuit simulator for ECU connections

In the first operating mode, original loads or simulated loads can be connected to control unit ports by software. 14 channels are available, each of which can be used to switch a control unit port between two load connections.

The second operating mode makes it possible to generate short-circuits of the control unit ports to ground or supply voltage. The load ports can be connected with a short-circuit bus for this purpose. If the short-circuit bus is connected to ground or the supply voltage, a current sensor measures the current flow. A current-proportional voltage signal is passed to the control port on the front panel.

In both operating modes, a status evaluation circuit compares the voltage for the control unit or load connection of each channel with two fixed threshold values. The result of the comparison is displayed by LEDs on the front panel and is available for processing via a status register.

One port of each control unit can be connected to one connection of the control port by software in order to measure the voltage. At this location, for example, the voltage trace of the control unit port can be monitored with an oscilloscope.

The Load20A14C board can be used either as a switch board or for short-circuit simulation. The board is configured by ETAS for the relevant application. There is a device in the RTIO Editor for each of the two operating modes. For each instance of a Load20A14C board, the RTIO Editor can be used to create only one of the two devices, and this must be appropriate for the operating mode set on the board.

28.9.1 Globals (Load20A14C Subsystem)

This section describes the global options of the Load20A14C subsystem.

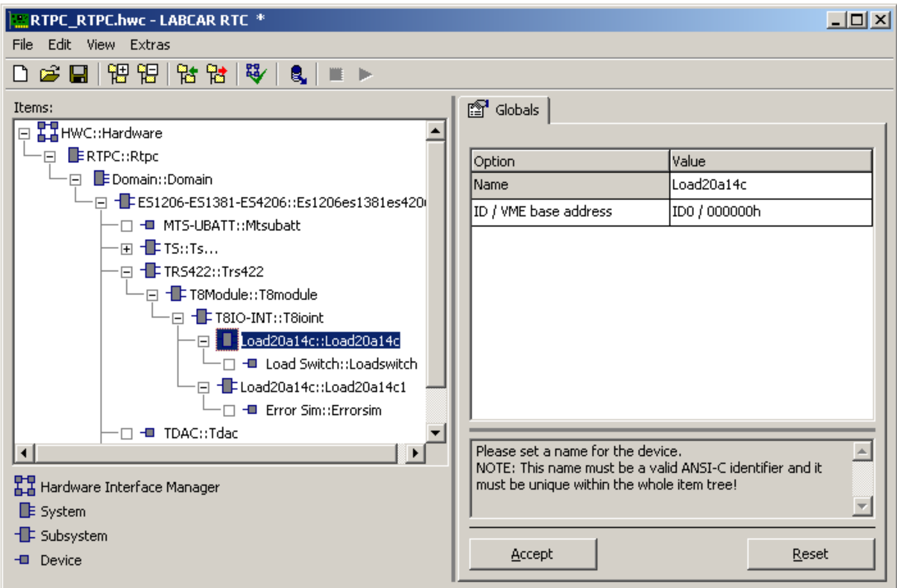


Fig. 28-40 The "Globals" Tab of theLoad20A14C Subsystem

ID / VME base address

Defines the VME base address of the board.

28.9.2 Globals (ErrorSim Device)

This section describes the global options of the ErrorSim device.

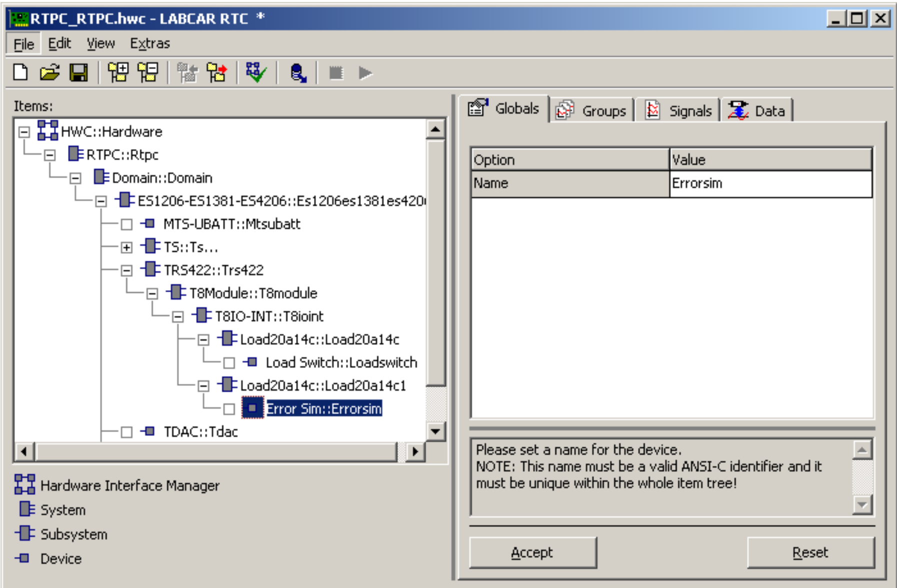


Fig. 28-41 The "Globals" Tab of the ErrorSim Device

The ErrorSim Device is used to control the Load20A14C board in *Short-Circuit Simulation* mode. One control unit port can be switched to the battery voltage or the LABCAR ground by the software. The current is limited with a fuse. The status of the fuse is available to the simulation model as an input signal.

28.9.3 Groups (ErrorSim Device)

This section describes the signal-group-specific options of the ErrorSim device.

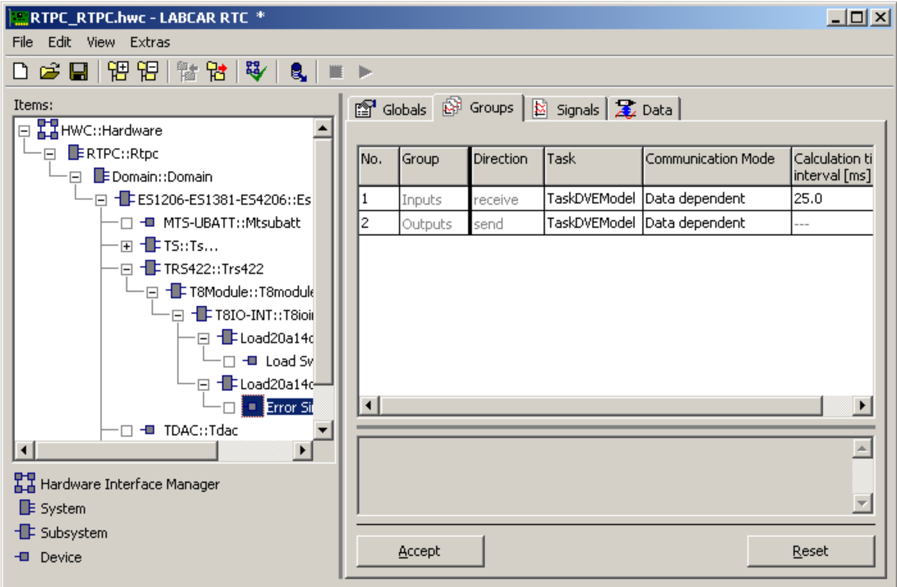


Fig. 28-42 The "Groups" Tab of the ErrorSim Device

The ErrorSim device has the following signal groups:

Signal Group	Direction	Description
Inputs	receive	Display of the status of fuse and the results of the threshold comparisons on the 14 status channels.
Outputs	send	Selection of the measuring channel, the ECU channel for short-circuit simulation and the type of short-circuit simulation.

Communication Mode

This field specifies when a data transfer takes place between the simulation target and the Load20A14C board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

Calculation time interval [ms]

A process runs on the on-board transputer of the T8IO board which periodically determines the status of the fuse and the results of the threshold comparisons on the 14 status channels and transfers these values to the simulation target in accordance with the option selected in the "Communication Mode" field: either always or only with changed values. This field specifies the time interval in which this process is calculated or activated.

28.9.4 Signals (ErrorSim Device)

This section describes the signals of the ErrorSim device and their options.

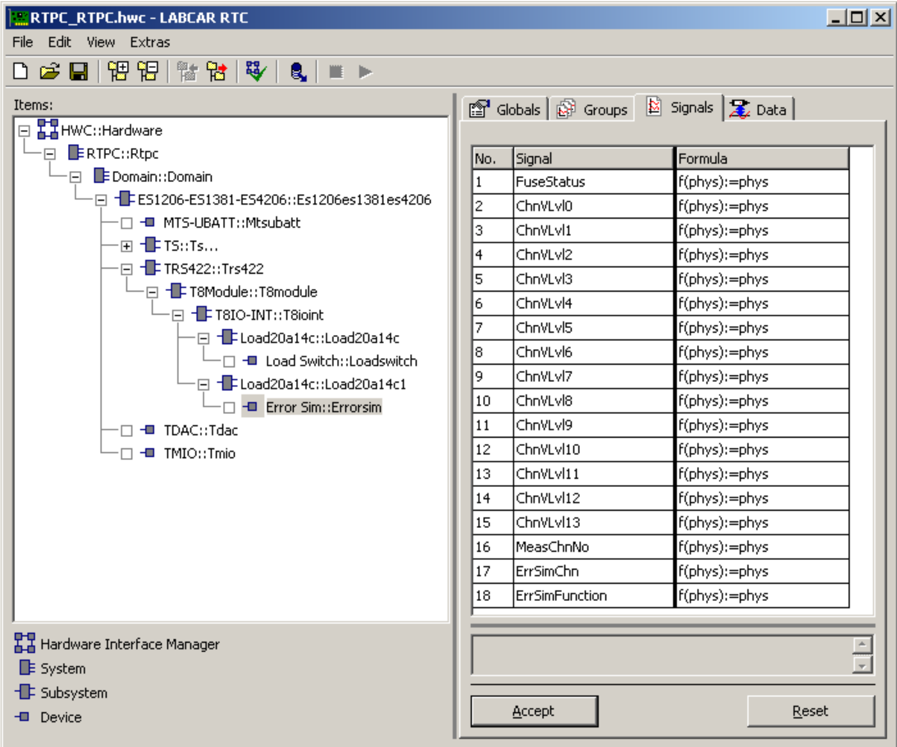


Fig. 28-43 The "Signals" Tab of the ErrorSim Device

The ErrorSim device has the following signals:

Signal	Direction	Description
FuseStatus	receive	Indicates the status of the fuse on the board
MeasChnNo	send	Selects the measure channel to be switched to a connection of the control port on the front panel of the Load20A14C board
ChnVLvl0-13	receive	These signals show the results of the threshold comparisons on the 14 status channels.
ErrSimChn	send	Selects the channel in which a short circuit or error will be simulated
ErrSimFunc-tion	send	Specifies the type of short-circuit or error simula-tion of the selected channel

28.9.5 Data Types and Value Ranges

This section describes the possible data types and value ranges of each signal of the ErrorSim device and the meaning of the various signal values.

Signal	Data Type	Description
FuseStatus	bool, int	0 or false: fuse is ok ≠ 0 or true: fuse is burned out
MeasChnNo	int	0 -13: corresponds to the number of the measure channel Other values: no measure channel is switched to the control port on the front panel
ChnVLvl0-13	int	0: $V_{ECUpin} < 2.5$ V 1: $V_{ECUpin} > 3.0$ V 2: 2.5 V <= V_{ECUpin} <= 3.0 V -> channel is not connected $V_{Loadpin}$ = voltage at the pin of connector "LOAD A"
ErrSimChn	int	0 -13: corresponds to the number of the error simulation channel Other values: no channel for error simulation is selected -> error simulation is disabled
ErrSimFunction	int	1: short-circuit to battery voltage 2: short-circuit to battery ground Other values: signal interruption

28.9.6 Globals (LoadSwitch Device)

This section describes the global options of the LoadSwitch device.

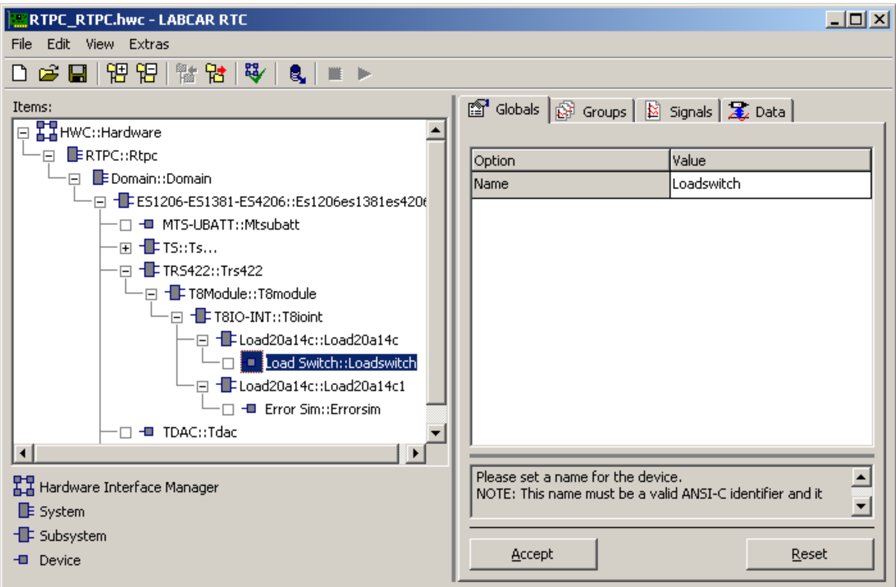


Fig. 28-44 The "Globals" Tab of the LoadSwitch Device

The LoadSwitch device is used to control the Load20A14C board in *Load-Switching* mode. An ECU port can be switched between two load ports with each channel of the board by software. The load connections are referred to as Load A and Load B.

28.9.7 Groups (LoadSwitch Device)

This section describes the signal-group-specific options of the LoadSwitch device.

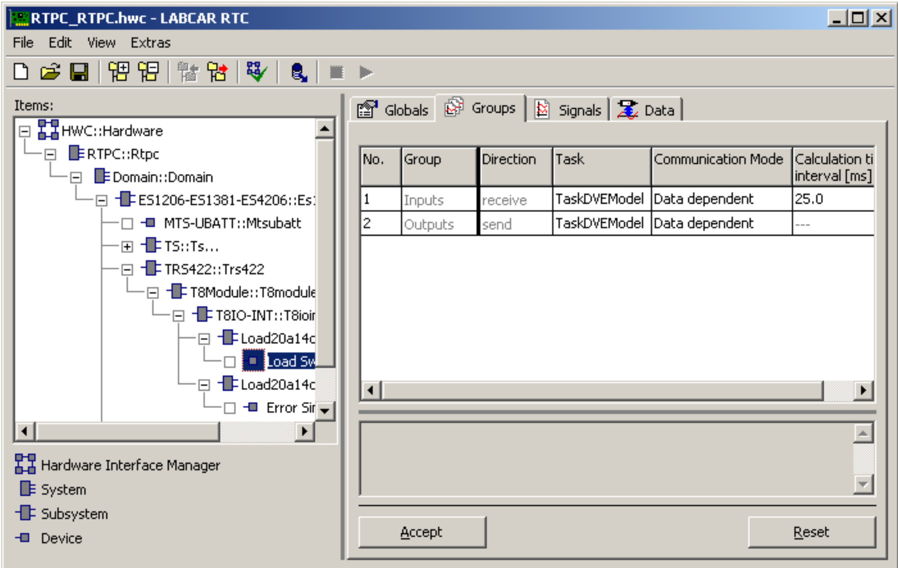


Fig. 28-45 The "Groups" Tab of the LoadSwitch Device

The LoadSwitch device has the following signal groups:

Signal Group	Direction	Description
Inputs	receive	Display of the status of the fuse and the results of the threshold comparisons on the 14 status channels
Outputs	send	Selection of the measure channel and control of load switching

Communication Mode

This field specifies when a data transfer takes place between the simulation target and the Load20A14C board. If the "Every interval" option is selected, the data transfer takes place every program step. If the "Data dependent" option is selected, the data is only transferred if it has changed. The "Data dependent" option is recommended to minimize data transfer on the serial transputer link bus.

Calculation time interval [ms]

A process runs on the on-board transputer of the T8IO board which periodically determines the status of the fuse and the results of the threshold comparisons on the 14 status channels and transfers these values to the simulation target in accordance with the option selected in the "Communication Mode" field: either always or only with changed values. This field specifies the time interval in which this process is calculated or activated.

28.9.8 Signals (LoadSwitch Device)

This section describes the signals of the LoadSwitch device and their options.

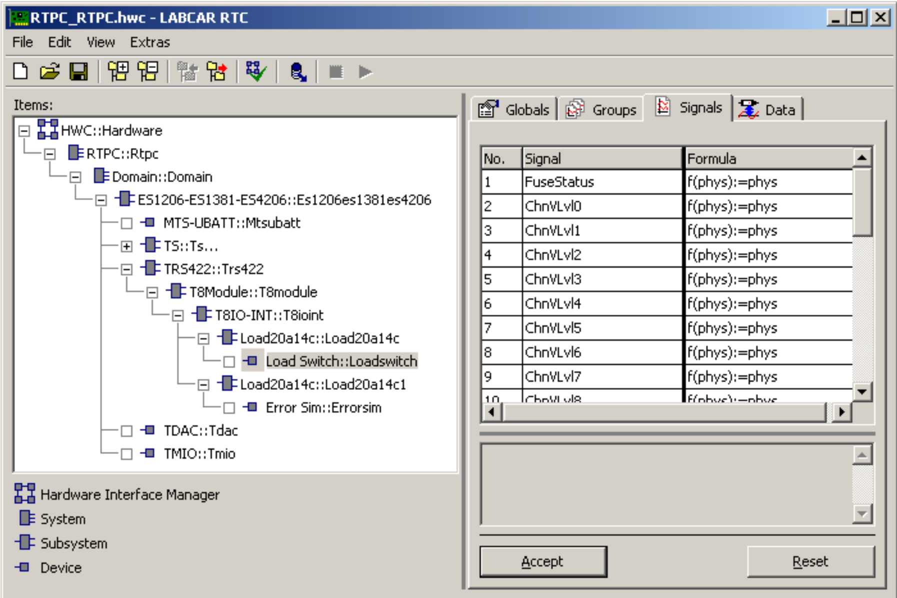


Fig. 28-46 The "Signals" Tab of the LoadSwitch Device

The LoadSwitch device has the following signals:

Signal	Direction	Description
FuseStatus	receive	Indicates the status of the fuse on the board
MeasChnNo	send	Selects the measure channel to be switched to a connection of the control port on the front panel of the Load20A14C board
ChnVLvl0-13	receive	These signals show the results of the threshold comparisons on the 14 status channels.
LdSwitchCtrl0-13	send	Specifies which load input the relevant switch channel should be switched to

28.9.9 Data Types and Value Ranges

This section describes the possible data types and value ranges of each signal of the LoadSwitch device and the meaning of the various signal values.

Signal	Data Type	Description
FuseStatus	bool, int	0 or false: fuse is ok ≠ 0 or true: fuse is burned out
MeasChnlNo	int	0 -13: corresponds to the number of the measure channel Other values: no measure channel is switched to the control port on the front panel
ChnVLvl0-13	int	0: V_{ECUpin} < 2.5 V 1: V_{ECUpin} > 3.0 V 2: 2.5 V <= V_{ECUpin} <= 3.0 V -> channel is not connected V_{ECUpin} = voltage at the ECU pin
LdSwitchCtrl0-13	int	0 - switches to Load A 1 - switches to Load B

29 Module Status

The Module-Status device provides information on the availability and the status of the module and of the hardware driver as well as serial and model number of the board.

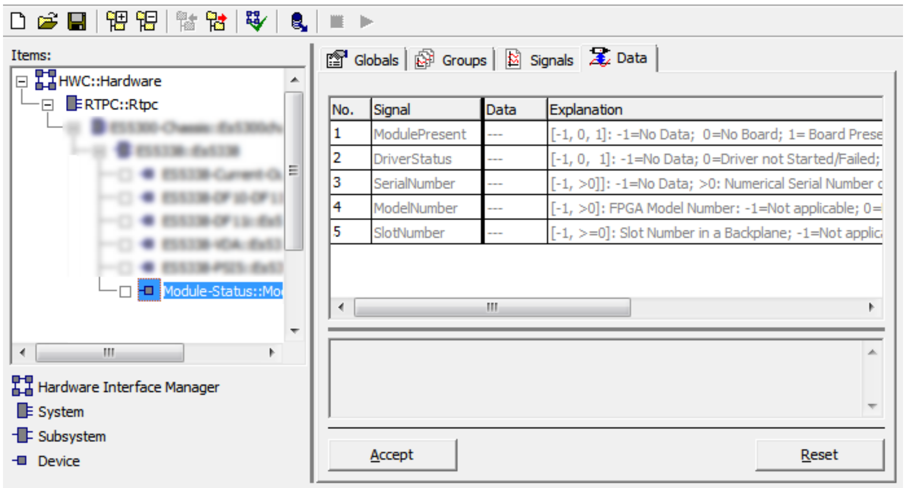


Fig. 29-1 The "Data" Tab of the Module-Status RTIO Element

Module Present

This signal contains information on the availability of the board.

DriverStatus

This signal contains information on the status of the hardware driver.

SerialNumber

The serial number of the board (if available).

ModelNumber

The model number of the board (if available).

SlotNumber

Slot number in specific backplane (if availiable).

Signal Name	Notes
ModulePresent	-1: No data available 0: Board not available 1: Board available
DriverStatus	-1: No data available 0: Driver not started 1: Driver started
SerialNumber	-1: No data available > 0: Serial number
ModelNumber	-1: No data available > 0: Model number
SlotNumber	-1: No data available > 0: Slot number

Tab. 29-1 Module-Status: Signals of the "Data" Tab

30 ETAS Contact Addresses

ETAS HQ

ETAS GmbH

Borsigstraße 14	Phone:	+49 711 3423-0
70469 Stuttgart	Fax:	+49 711 3423-2106
Germany	WWW:	www.etas.com

ETAS Subsidiaries and Technical Support

For details of your local sales office as well as your local technical support team and product hotlines, take a look at the ETAS website:

ETAS subsidiaries	WWW:	www.etas.com/en/contact.php
ETAS technical support	WWW:	www.etas.com/en/hotlines.php

Index

F