Measurement & Calibration for µP-based vehicle computers

Tilman Glötzner, May 2024

Introduction

Currently, the automotive industry undergoes fundamental changes such as electrification, connectivity and automation of driving functions. This has profound implications on the infrastructure needed. It drives all OEMs to overhaul their existing electric and electronic vehicle architectures or to develop the next generation to what is to become the "Software Defined Vehicle (SDV)".

Rise in Complexity

Not only autonomous driving functions but also new applications spanning across the classical domains of powertrain, chassis and body control will require more computing power. In addition, cross-domain functions that are heavily interdependent with other functions are being consolidated and migrated from µController based to µProcessor based systems. Running them together on a vehicle computer rather than on dedicated controllers helps to manage the complexity. An example would be a navigation function for an electric vehicle that needs to take the charge and thermal state of the batteries, drive profile, availability and power of charging stations among other information into account. This will further fuel the push towards the deployment of µP-based vehicle computers in cars to deal with the ever-increasing complexity of Software Defined Vehicles and their Digital Twins.

Mixed Networks of collaborating µPs and µCs

To reduce the need for space and to allow for multipurpose hardware that offers more flexibility and with it, shortens the time-to market, functions are already today realized as much as possible in (embedded) software. This trend will intensify to the point where functions may run either completely or partially on vehicle computers rather than a dedicated controller/processor.

Standards like AUTOSAR support this by providing frameworks that allow the deployment of functions across several computing platforms, whether they are based on µCs, µPs or combinations of both in SoCs.

Software Deployment after SOP

Already today, there are new business models emerging. Rather than providing the full set of functionalities with the produced car, the OEMs intend to provide extra functions on request of the customer thereby generating an additional revenue stream.

Moreover, the capability to release software into cars in the field, allows the OEMs to react quickly to market or legal changes, to evolve the functions in the cars over time and get early feedback from the user base, or to provide bug fixes.

µPs together with corresponding operating systems and technologies such as OTA (updates over the air) are particularly well suited to enable post-SOP software deployments without the need for the customer to visit a workshop. Unlike µCs, µPs allow partial updates, i.e. an application or software component rather than a complete firmware image. This not only reduces downtime but also supports the introduction of service-oriented software architectures in cars.

Changes in Development Model

Traditionally, the automotive industry developed software along the V-cycle that is well suited for embedded controller functions. With the move of the automotive industry towards service-oriented architectures and agile development methods also development technologies and methodologies such as object orientation, encapsulation, test driven development and continuous development and deployment are increasingly applied. At the same time, technologies that proved to be beneficial in the past, will be retained. Though the functions and associated data sets might be different, monitoring and optimizing data sets directly in the control unit during test drives, either virtually or on the road, remain an efficient approach to improve the overall performance of a car. The future challenge will be to combine elements of both worlds, i.e. the monolithic systems with preconfigured communication matrices and service-oriented architectures, to speed up the development cycles and at the same time, to master the rising complexity. Excellent tool support, the application of standards and agile development models will prove to be vital along this journey.

What are the implications on the requirements of a measurement and calibration system?

Support of AUTOSAR Adaptive for µPs

As µPs running AUTOSAR Adaptive Platform are increasingly deployed in the vehicles, the instrumentation needs to support µP as well as µC measurement and calibration at the same time and integrated into a common tooling.

Support of Distributed Functions

The instrumentation needs to be able to measure and calibrate data from functions deployed across several computing platforms and additionally offer a function centric view of the acquired data.

Support of interactive measurement and calibration as well as data logging

The use case of a calibration engineer visualizing and tuning parameters in a test car will still be widespread. In addition, there will be more developers wanting to record ECU/application internal data while the car is being operated and to analyze the results at a later point in time. Data logging but also filtering data, i.e. data reduction, triggering and on board pre-processing will become more important.

Measurement and calibration needs to support existing roles along the development process

Even though the software development methodology changes, measurement & calibration will remain a worthy, efficient and powerful methodology for functional debugging and optimizing the software along the development cycle. Also, today's users of the measurement & calibration instrumentation will continue to benefit from it, i.e. software developers, verification & validation engineers, integrators, and calibration engineers.

Measurement and Calibration of global and local objects within an application

To enable efficient and error-free implementations, developers need to understand the behavior/state of functions as a whole even if they are deployed across several ECUs. Not only monitoring of the external behavior at the APIs is required but also the examination of internal behavior and states of an application needs to be possible. The possibility to change the values of variables within an application, i.e. to calibrate, allows to conveniently trigger functions on demand whose activation might be difficult or even dangerous to reproduce during a real driving situation on a test track. This implies that read access (measurement) as well as write access (calibration) to global and local variables in methods or procedures of an application shall be supported. As local variables are typically stored on the stack and their lifetime limited to the duration of the method or procedure being executed, precautions have to be taken to not attempt access to objects no longer being available.

Time Synchronization

Today, vehicle computers across a network of ECUs share a common time base. To be able to visualize processes with high time resolution across several ECUs, the measurement instrumentation has to provide a common high-precision time base. The time base may be independent or synchronized with the time base of the vehicle network. Protocols such as IEEE802.1as are well suited for this.

Security

More and more functions are distributed over several nodes that communicate over the vehicle network. These functions may partially run outside the vehicle as cloud applications which requires the vehicle to be connected to the internet. Compared to a function that is completely deployed in one controller, distributed functions connected over potentially insecure vehicle busses are an easy target for attacks. An online connection to the internet opens up new modes of attack. Security breaches may have implications on system safety. To ensure the integrity of the overall system, the instrumentation will also need to address security aspects.

Small footprint of measurement & calibration instrumentation

Even though µP based vehicle computers provide more resources, these are to be used for new functions providing even more value to the customers. As in the past, the instrumentation for measurement and calibration needs to maintain a small footprint in terms of memory and computational resource consumption.

Solution Approaches for µP based Measurement & Calibration

As for embedded µC based control units' interactive measurement and calibration tool solutions are already available and widely used, there is a clear need of similar approaches for µP based control units. While monitoring the data exchange between applications via services can be accommodated with the existing infrastructure provided by the communication management functional cluster, measurement & calibration requires dedicated instrumentation.

In principle, there are 3 approaches to the instrumentation for µP based controllers.

First Approach

Figure 1 depicts the first approach: Measurement and Calibration via SOME/IP. SOME/IP is often used for inter-controller applications within the vehicle and is therefore typically available. The measurement & calibration system is physically attached to the automotive ethernet network as additional node. On protocol level, it behaves like an additional controller that subscribes to services. SOME/IP can work with both, TCP or UDP. Even though the resource footprint of UDP is smaller, customers typically prefer a reliable transmission protocol with guaranteed delivery for their measurement and calibration systems and hence employ TCP.

Advantages:

- The infrastructure is readily available and standardized as part of AUTOSAR Adaptive.
- As long as the user has full access to the source code as well as the build chain, its implementation is relatively easy. This would for example apply to a software developer in the early stages of an ECU development project.

Disadvantages:

- The set of signals to be measured or calibrated is not dynamically reconfigurable.
- The user needs to implement a service as part of the application to be instrumented to make variables available for measurement or calibration. This requires control over the source code of the application, the configuration of the Communication Management Functional Cluster as well as the build and deployment chain. For changes of the measurement configuration, the complete cycle of compiling, linking, building, and deploying typically needs to be performed.
- To access local variables within the application, the user needs to implement a mechanism that deals with variables leaving their scope and therewith becoming temporarily inaccessible for measurement or calibration.
- SOME/IP does not specify any security mechanisms, such as encryption, authentication and authorization. Hence, this has to be addressed separately. This may prove to be complex and prone to misconfigurations.

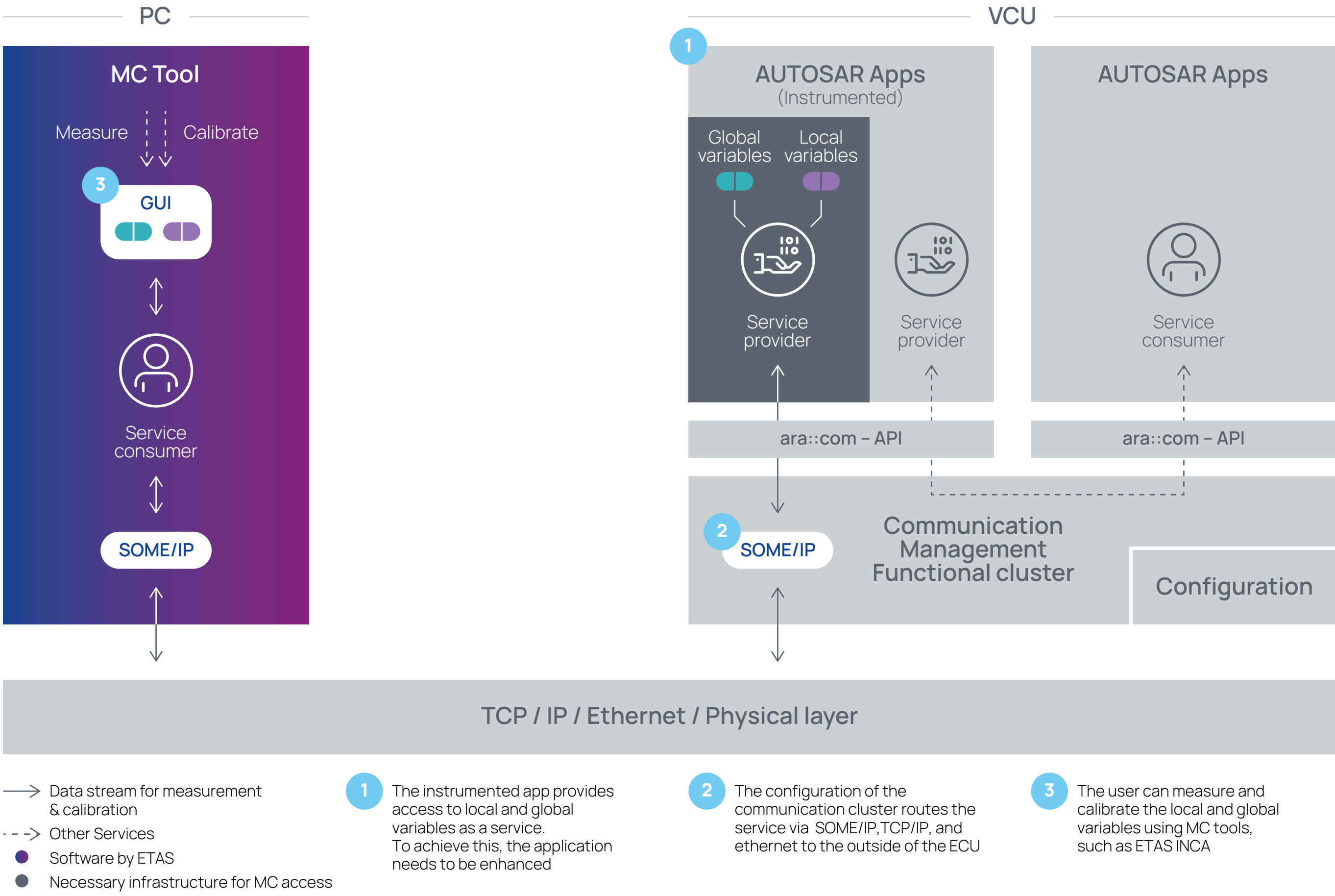

Fig. 1: Measurement and Calibration via SOME/IP.

Second Approach

Figure 2 depicts the next possible approach: Integration of XCP slaves

An XCP library is added to each application requiring access for measurement and calibration of application internal variables.

Advantages:

- XCP is a mature standard that is supported by most measurement & calibration tools. It can hence be directly connected to most tools.
- Changing the variable set to be measured or calibrated can be handled via XCP. A new build of the AUTOSAR App is not necessary.

Disadvantages:

- Access to application local variables requires the XCP implementation to take care of their scope.
- As each application requires to establish its own XCP connection, a library requiring a considerable footprint needs to be integrated in each application. This drains the resources of the underlying µP platform in terms of memory demand – in particular if TCP is used for lossless transport. In sum, this approach scales poorly with number of applications to be instrumented.
- Security mechanisms may only need to be implemented once on application level as part of the library. As each XCP stack maintains its own connection, each connection needs to be configured individually.

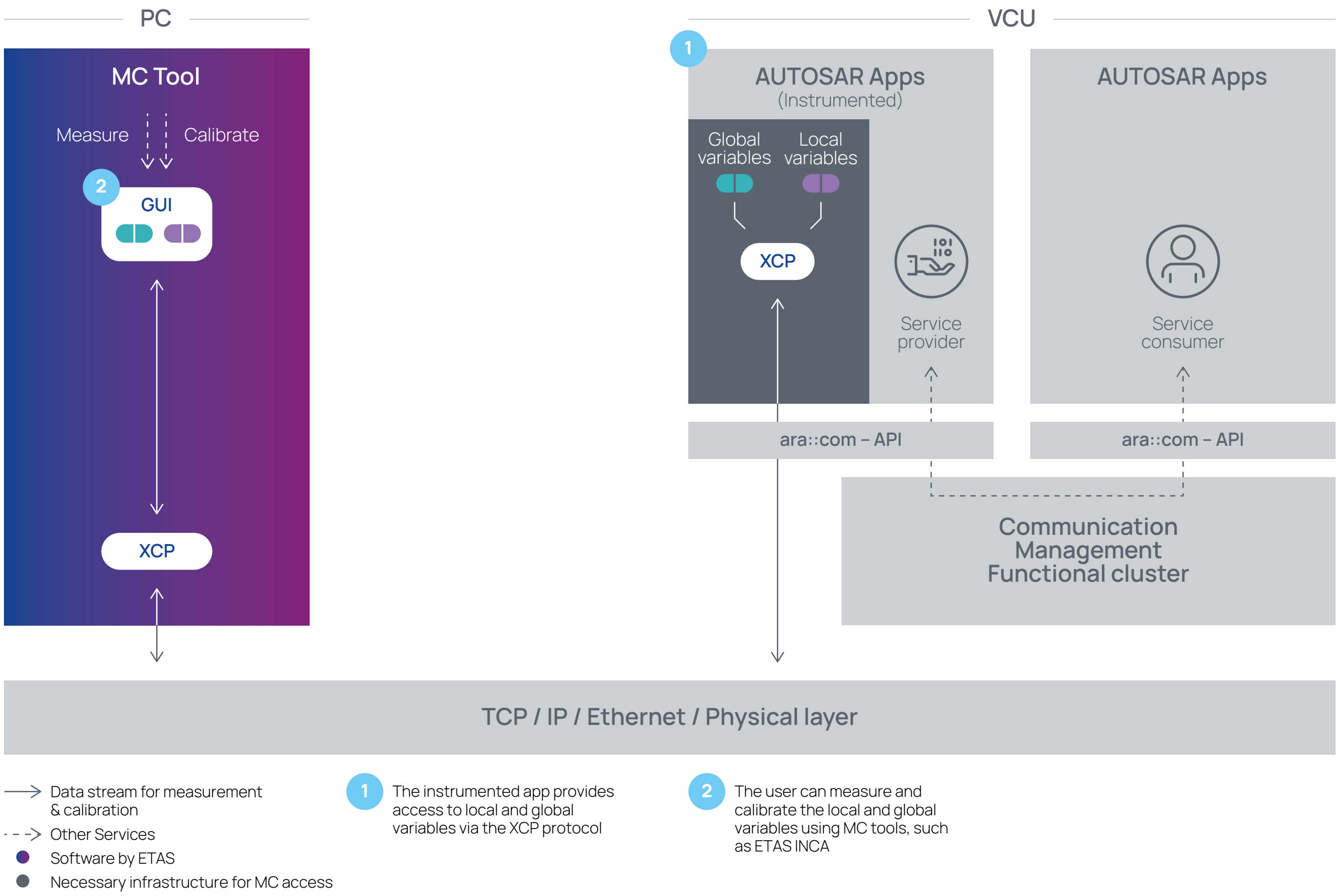

Fig. 2: Integration of XCP slaves

Third Approach

Figure 3 depicts the last possible approach: ETAS Measurement & Calibration Gateway

The gateway application concentrates on the measured data coming from the applications and packs it into one TCP connection destined for the MCD tool. For calibration data, the gateway application serves as a distributor that delivers the data coming from the MCD tool to the application. XCP works address based. The gateway application also ensures that these addresses are mapped into the local address space of the application. The actual write or read access to a variable is facilitated by a small library provided by ETAS that is linked to the application.

Advantages:

- XCP is a mature standard that is supported by most measurement & calibration tools, i.e. the existing tooling can be reused in the AUTOSAR ecosystem.
- The ETAS Measuremet & Calibration Gateway application only needs to be integrated once.
- The middleware requires no reconfiguration. Changing measurement configurations are handled via the gateway application.
- Access to application local variables is possible. The ETAS library instrumenting the application also treats objects that have a lifetime shorter than the application such as local variables in methods or procedures.
- Optimized resource footprint with good scaling properties. Only one instance of the resource heavy XCP stack is needed in the gateway application to serve multiple instrumented applications. The applications themselves require the integration of only a small library.
- Security mechanisms and their configuration can be addressed centrally as part of the gateway application.

Disadvantages:

- As with the second approach, "Integration of XCP slaves", the gateway approach also requires the integration of an ETAS library into each app to be instrumented. However, in comparison, this library is small and only requires limited resources.

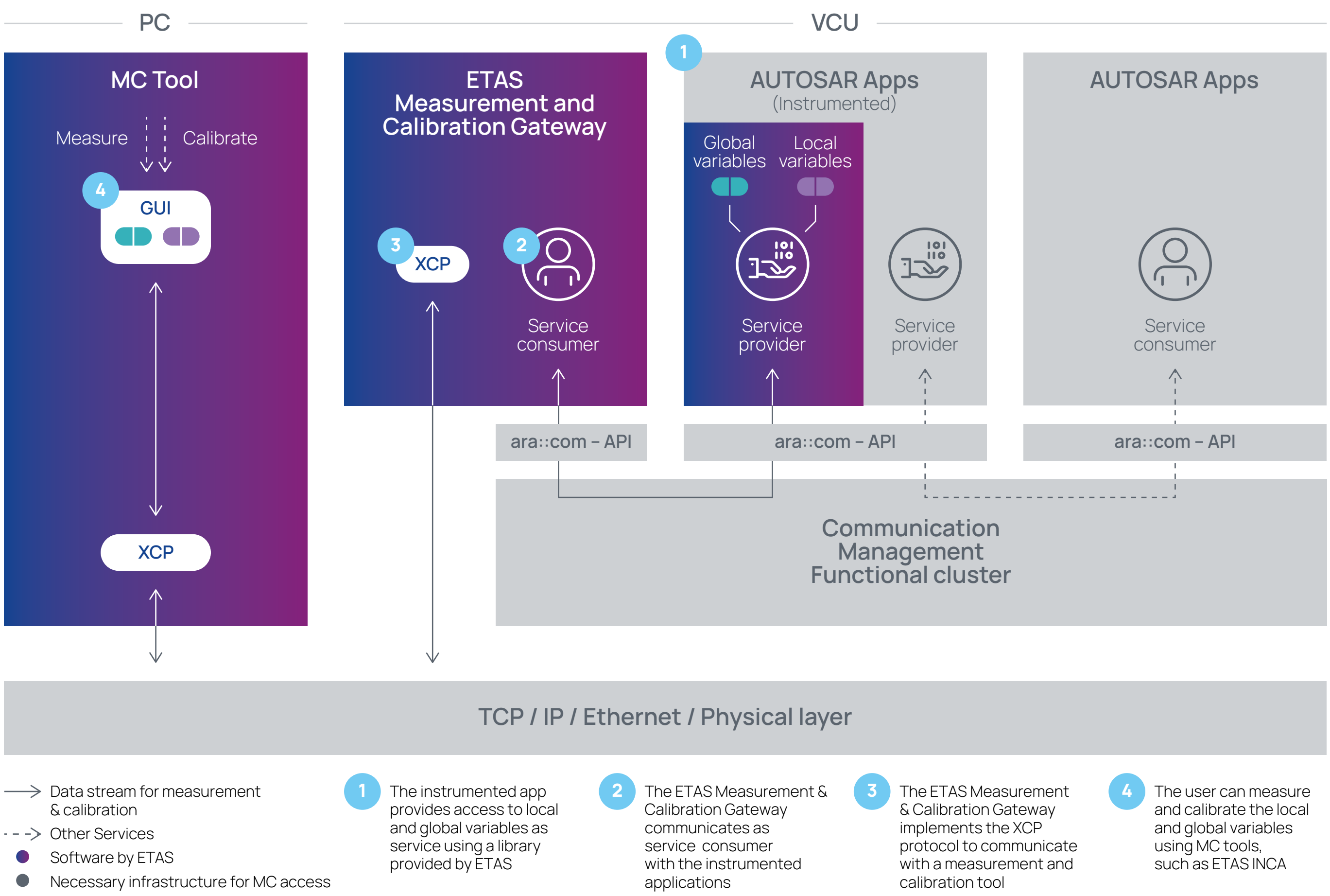

Fig. 3: XCP Gateway

Conclusion

With the increased application of μPs in the classical domains such as power train or chassis, in-vehicle measurement and parameter modification will still be needed as consistent working model. Measurement and Calibration as methodology remains a valuable asset to efficiently optimize and validate functions.

ETAS will support the users in 2 ways:

For users that have access to the application source and the build system and at the same time, only few variables to measure or calibrate, ETAS will enhance its measurement & calibration tool INCA to support SOME/IP by an add-on. This represents the approach described in solution (1).

Solution (2), i.e. the integration of an XCP slave, and solution (3), the ETAS Measurement & Calibration Gateway, provide an identical set of functionalities with technically different approaches. Both allow for the efficient handling of many labels. In addition, both solutions allow to seamlessly reuse existing tool chains established for ECUs (μCs) and apply it to VCUs (μPs).

The ETAS Measurement & Calibration Gateway approach only needs relatively few resources of a μP in terms of absolute computing power, memory and bandwidth as well as in comparison to solution (2). ETAS hence chose to implement and offer the Measurement & Calibration Gateway to measurement & calibration users with higher demands. To further increase the performance by using the available resources more efficiently, ETAS developed a protocol, GCF, to bundle several XCP and their underlying TCP connections into one.

With these solutions, ETAS accompanies its customer in the transition to VCUs and provides necessary functionalities known from μC based ECUs such as measurement and calibration also for VCUs while keeping the impact onto the VCU as low as possible and maintaining a high degree of comfort and performance.

About ETAS

Founded in 1994, ETAS GmbH is a wholly owned subsidiary of Robert Bosch GmbH with a presence in twelve countries in Europe, North and South America, as well as Asia. The ETAS portfolio includes basic vehicle software, middleware, development tools, and holistic cybersecurity solutions for the implementation of software-defined vehicles. With our solutions and services, we empower vehicle manufacturers and suppliers to develop, operate, and secure their products with utmost efficiency.

For more information about our solution please contact us via email to **sales.de@etas.com** or fill out our **contact form**.

ETAS GmbH
Borsigstraße 24, 70469 Stuttgart, Germany
T +49 711 3423-0, info@etas.com