

Open Interface for ECUs

By
Burkhard Triess,
ETAS

XETKs speak XCP-on-Ethernet

Among the typical tasks characterizing the development and calibration of automotive ECUs are the acquisition of measurement data, the setting of values of parameters, characteristics, and maps for open-loop, closed-loop, and diagnostic functions, along with ECU flash memory programming.

In situations where controls of any kind depend on fast responses, measurement data must be acquired at high rates. In many cases, the modification of the values of characteristic quantities must occur as concurrently as possible. Flash programming should not only take place as rapidly as possible but should also still be possible after an ECU crash. With respect to the bypass method deployed in many cases in the development of new vehicle functions, these are implemented on an experimentation system. For a Function-in-the-Loop test, the behavior of individual ECU functions is validated by means of a plant simulation. In both cases, the experimentation or testing system must be interconnected with the ECU in real time. Accordingly, the performance of development and calibration tasks calls for a sufficiently powerful ECU interface.

At first glance it would seem appropriate to handle the communication with the development ECU through the same serial interface – the production interface – that will be deployed with the production ECU. The downside of this practice are the limited data transfer rates characteristic of serial interfaces: Firstly, the driver software requires computing power for the serial communications, and secondly, the bandwidth provided by this interface type is at best moderate. An inherent benefit of production interfaces is the availability of standardized protocols, such as XCP for ECU calibration.

Ethernet as an interface for automotive ECUs

As an alternative to production interfaces, the deployment of Ethernet as an ECU interface provides development with a variety of benefits: Ethernet is a globally distributed standard with proven long-term stability. Virtually all of today's PCs and notebooks are equipped with an Ethernet port. Data transfer occurs at high speeds of 100 Mbit/s or 1 Gbit/s and uses economical Ethernet cables. These can be up to 100 meters in length, which is ideal for deployment onboard a vehicle or on the test bench. The data transfer is extremely robust. On the hardware side, the Ethernet standard provides for electrical isolation and includes mechanisms for both error detection and troubleshooting.

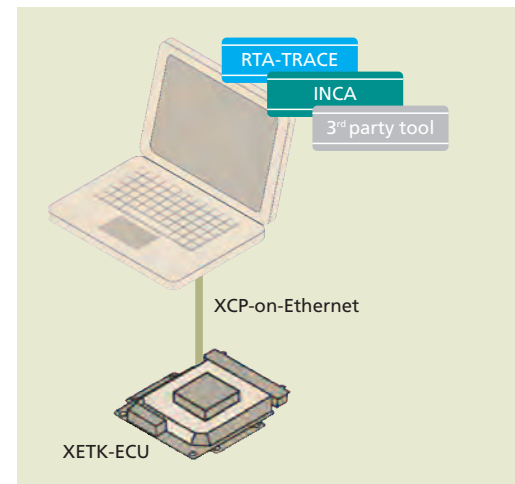


Figure 1:
Several software
applications on
a host PC access an
XETK-equipped
ECU.

There are two possible approaches to setting up a direct Ethernet connection between an ECU and an off-the-shelf Windows® PC, with the use of standard protocols such as TCP/IP or UDP/IP. The conventional approach – used in conjunction with the ETK (memory emulator test probe) – deploys an economical compact interface hardware that is integrated in the ECU. The ETK is connected to the PC by means of an interface module, which provides for TCP/IP and UDP protocol handling. The direct PC connection of the new approach – implemented in conjunction with the XETK – involves somewhat more sophisticated hardware, which is installed onboard the ECU.

XETKs speak XCP-on-Ethernet

Using the capabilities of the FPGA system-on-a-chip technology, it is today possible, without the need to change the geometry, to implement the XETK's Ethernet interface in tight quarters beside the parallel or serial ETK microcontroller interfaces. XETKs use the XCP-on-Ethernet standard protocol for data transfer. Analogous to XCP-on-CAN, XCP-on-Ethernet comprises an open ECU interface.

The XCP-on-Ethernet interface of the XETK is as efficient as the ETK interface, which makes it considerably more powerful than CAN. As is the case with the TCP/IP and UDP carrier protocols, the XCP protocol is run by the XETK. The ECU's microcontroller is not burdened in any way by the XETK's communications with the PC or a rapid prototyping hardware setup. In contrast to both CAN and ETK, the XETK dispenses with the additional PC interface.

Fully ETK compatible

As regards the ECU, both ETK and XETK are entirely identical, both in terms of the interface handling ECU hardware and software, and of the physical dimensions. This makes the new XETK interface fully compatible with the established ETK. ECUs therefore interact in an identical fashion with either ETK or XETK. Both interfaces, i.e., ETK and XETK, are equally supported by the INCA, INTECRIO, and ASCET development tools.

The deployment of the XETK obviates the expenditure for the PC interface. The ETK approach, on the other hand, offers a cost advantage with respect to both system ECU and ECU interface.

New applications

The XETK is supported by the RTA-TRACE and LABCAR-RTPC tools. RTA-TRACE allows the accurate analysis and statistical evaluation of individual OS tasks, both in terms of sequence and interrupt patterns, and of the dynamic memory and computing time allocation.

The new Release V4.1 of the LABCAR-RTPC application facilitates the direct connection of an XETK-equipped ECU to a PC-based Driver-Vehicle-Environment (DVE) simulation.

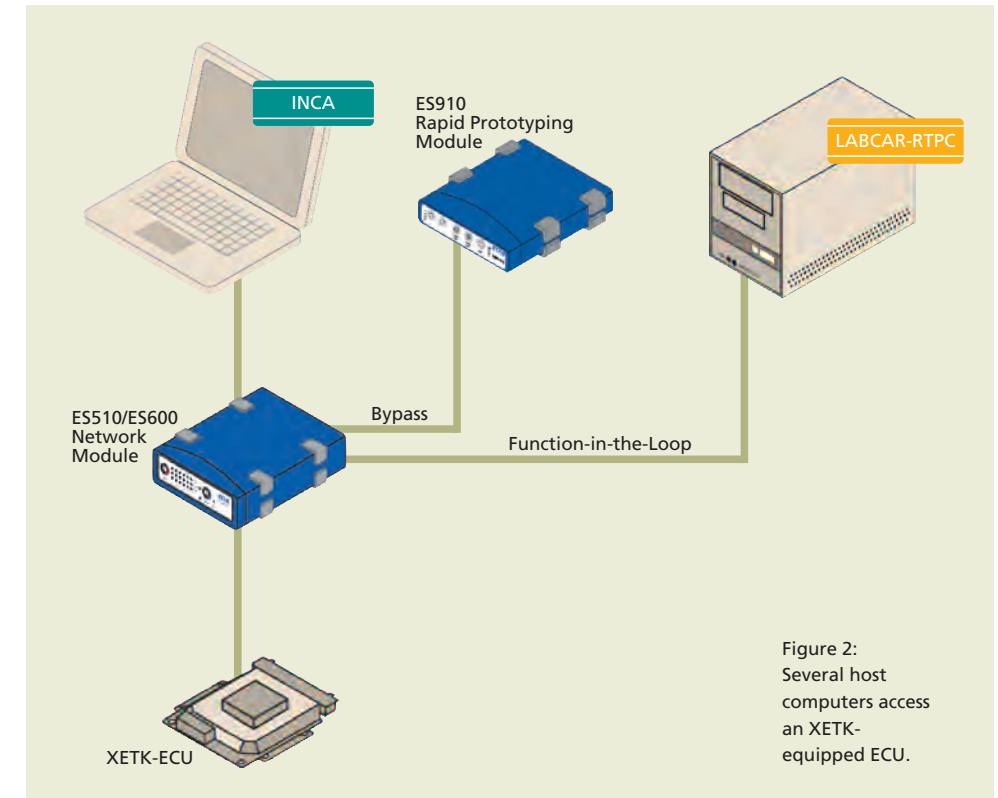
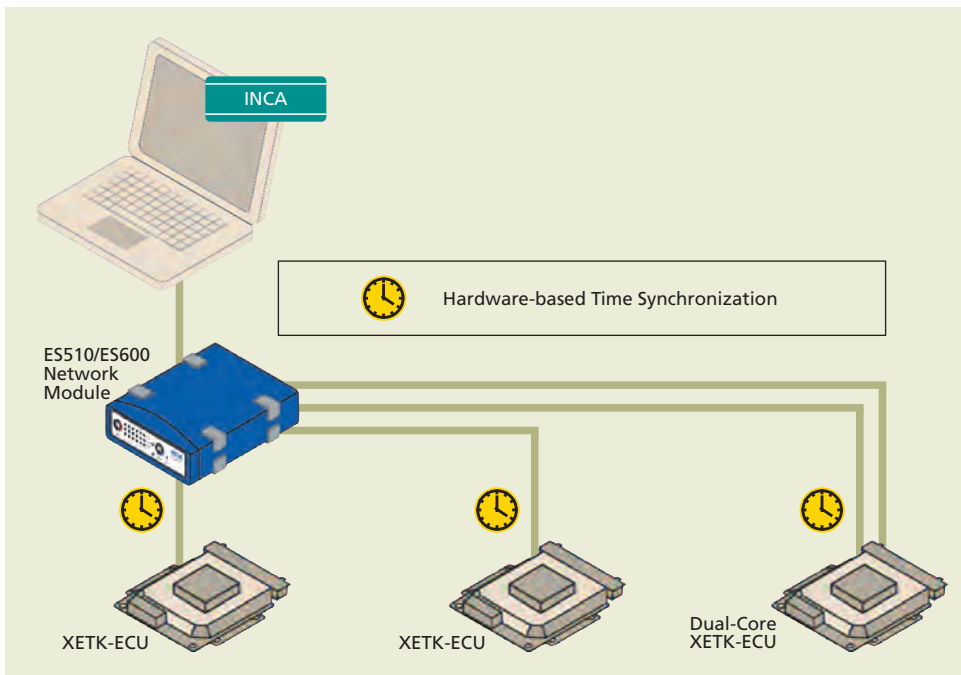


Figure 2:
Several host
computers access
an XETK-
equipped ECU.

This is accomplished by setting up a software link between the real-time simulation and the application functions of the ECU. On the one hand, the resulting "Function-in-the-Loop" setup dispenses with the signal conditioning hardware required for connecting the ECU in conventional testing systems. On the other hand, compared with a computer-based, classical Software-in-the-Loop test, this technology offers the advantage of testing the ECU logic on the target platform.

The XETK is capable of simultaneously handling up to four XCP communication channels. Not only does this facilitate concurrent access by several PC-based software applications to an XETK-equipped ECU. For example, the parallel operation of INCA and RTA-TRACE, makes it possible to check whether an implausible ECU signal measured with INCA is the result of a software fault (Figure 1). On the other hand, an XETK-equipped ECU can be connected to several computers by means of a network module. For example, a new function being implemented on a rapid prototyping module could be connected to an XETK-ECU through a bypass. It could then be calibrated in INCA with the use of an environment that is simulated by LABCAR-RTPC (Figure 2).

Conversely, several XETK data lines can be connected to a host computer through a network module (Figure 3). This option is of interest in the calibration of open-loop and closed-loop control functions, whenever a system is either distributed across several ECUs or implemented by an ECU equipped with several processors. In such cases, the XETK data of the various ECUs can be reconciled by means of a hardware-based time synchronization process.



Summary

XCP-on-Ethernet is a standardized, high-performance data transfer protocol deployed in the development of automotive electronics. Making use of this standard, the XETK interface for electronic control units facilitates a direct ECU-to-PC connection. Thanks to the XCP-on-Ethernet standard protocol, XETK-equipped ECUs can be easily integrated with existing tool-scapes. Paired with the proven Ethernet topology, the XETK's arbitration function facilitates new multiple-host applications featuring networked ECUs and/or ECUs equipped with multiple processors.

Figure 3:
Systems containing several ECUs, or multi-processor ECUs are connected to a host application.