

XCP – a powerful protocol with multiple applications

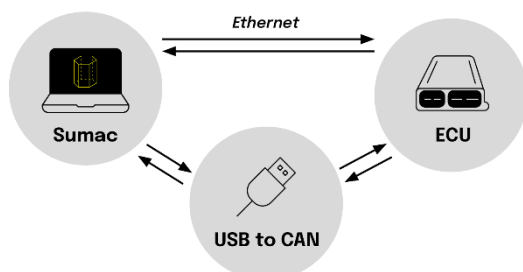
Originally designed for the automotive industry, XCP (Universal Measurement and Calibration Protocol) is a protocol with the primary purpose of adjusting internal parameters and acquiring the current values of internal variables of an ECU. Over time, it has gained popularity in other industries such as aerospace, transportation, and medical industries. Furthermore, its growth and adaptation are continuing.

XCP is used by OEMs, Tier one, Tier 2 etc.

With scalable design, low resource consumption and high performance, XCP fulfills all the requirements of a universal measurement and calibration protocol. XCP standardizes the calibration and data acquisition process and hence reduces the variety of calibration systems and avoids the need to create specific ECU implementations for specific application tasks.

The X in XCP

The first letter X in XCP expresses the fact that the protocol is designed with a clear separation of the core and transport layers for a variety of bus systems, which provides the flexibility developers and engineers need to work with different systems. The standard consists of a base standard, which describes memory-oriented protocol services without direct dependencies on specific bus systems. Several associate standards contain the transport layer definitions for CAN, FlexRay, Ethernet (UDP/IP and TCP/IP), serial links (SPI and SCI) and USB. This principle is illustrated in the figure below. The measurement program running on the PC, [Sumac](#) in this case, can communicate with the ECU via several different physical transport layers.



Read and write access at runtime

XCP enables read and write access to variables and calibration parameters of microcontroller systems at runtime. Variables and entire datasets can be

acquired or stimulated synchronous to events triggered by timers or operating conditions. At any time, you can have a full view of internal processes and acquire detailed information on how the algorithm is running. In addition, XCP also supports programming of flash memory.

Single master multi slave concept

XCP is as a single master multi slave concept. The master is a measurement and calibration tool. Usually, the master is a computer application or a data logger. Slaves are integrated in the embedded system. A master can be connected in parallel with many slaves at the same time, while a slave has only one master at a time.

Fast and reliable data transfer

One key advantage of XCP is its ability to provide fast and reliable data transfer between a host computer and an embedded system. This allows developers and engineers to easily perform calibration, measurement, and debugging.

“In addition to its speed and reliability, XCP offers a range of features that make it a flexible and versatile protocol.”

For example, XCP allows for dynamic DAQ (data acquisition), which enables engineers to easily change the data being collected, based on changes in the system. This is particularly useful for systems that are constantly evolving or where conditions are constantly changing.

Efficient tool for calibration, measurement, and visualization

Udoka Electronics has developed an intuitive, lightweight, and powerful tool – [Sumac](#) – for calibration, measurement and visualization of embedded control systems or ECUs. It works either offline or in real-time via XCP communication or DBC-based CAN measurement. It is the only software on the market today supporting Linux, and compatible with commercial off-the-shelf USB-to-CAN and CAN FD dongles from ETAS, Kvaser, Peak, Vector & Lawicel. With flexible license models Sumac is also cost effective.

When developing the software and designing Sumac’s intuitive and UX-friendly interface, much effort went in to making it pedagogic and functional, with no unnecessary features or functions. This makes Sumac fast and easy to use, ideal during all stages of software development, calibration, testing, and validation.

A2L files

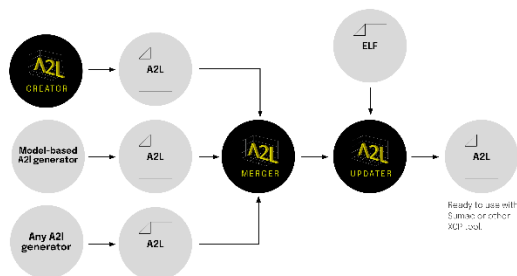
To be able to set up data acquisition and perform calibration an XCP master needs information about the software running in the ECU. An A2L-file is a structured ASCII text file that contains measurement, calibration, and event definitions used with XCP for data acquisition, stimulation, and calibration.

No need for recompiling or reprogramming to change measurement scope

The A2L files provides access to specific parameters and variables without the need to have hardcoded access in the ECU application software. In other words, the ECU contains only a generic XCP-protocol slave stack, which responds to memory access requests from the calibration system. This makes the software package to be integrated in the ECU very small and efficient. Different calibration and measurement tasks can be performed by different configurations of the calibration system without recompiling and reprogramming the ECU application code.

Automating A2L creation and management

[Acam](#) is a software developed by Udoka Electronics, that automates all steps of A2L creation and management in a software build, maximizing efficiency and quality by minimizing the need for human actions. A typical workflow is illustrated in the figure below.



Acam is designed to be integrated in build environments and C&I/CD systems to achieve a fully automated workflow. It's compatible with Windows, Mac OS and Linux.

Creating A2L files

An A2L file can be created in several ways:

- Generated by a code generator such as Embedded Coder, TargetLink or ASCET
- Hand-written
- Created by Acam Creator

"Acam Creator offers a unique algorithm which automatically create A2L files from C/C++ source code"

Merging A2L files

Regardless of how they are created, the A2L files describing the components must be merged into a single A2L file before they can be used with a measurement and calibration tool such as [Sumac](#), ATI Vision, Vector CANape, ETAS INCA etc.

Update from latest software build

[Acam](#) also updates your A2L with ECU addresses from the latest software build. The A2L file which describes the software must be updated with the memory address of each signal and calibration parameter whenever the software is rebuilt. [Acam](#) smoothly performs this task for plain variables, struct fields, class members, arrays and more.

Smooth filtering of your A2L file

To protect intellectual property or prevent accidental calibration changes it is often a good idea to remove some measurements, characteristics, groups, and functions before sending the software and A2L to customers or partners. Acam accomplishes this task by filtering the A2L file. It offers a rich set of expressions to select which items shall be kept or removed. The A2L-file contains all the information necessary to access and correctly interpret the data that is transmitted via the XCP protocol.

Automating test, calibration, and data acquisition for ECUs

[Pass](#) is a python interface for calibration and data acquisition of embedded control systems via XCP. Developed by Udoka Electronics, Pass is ideal to automate tasks such as calibration, measurement, test, and validation. In a nutshell – everything which can be done manually with Sumac can be automated with Pass. It is straightforward to integrate Pass in existing test frameworks such as for example Robot Framework.

In line with the evolution

As new technologies and applications emerge, Udoka Electronics continues to develop intuitive solutions. Please feel free to [contact us](#) for further input on how our products can become an inherent part of your development toolchain to facilitate and speed up your processes.