

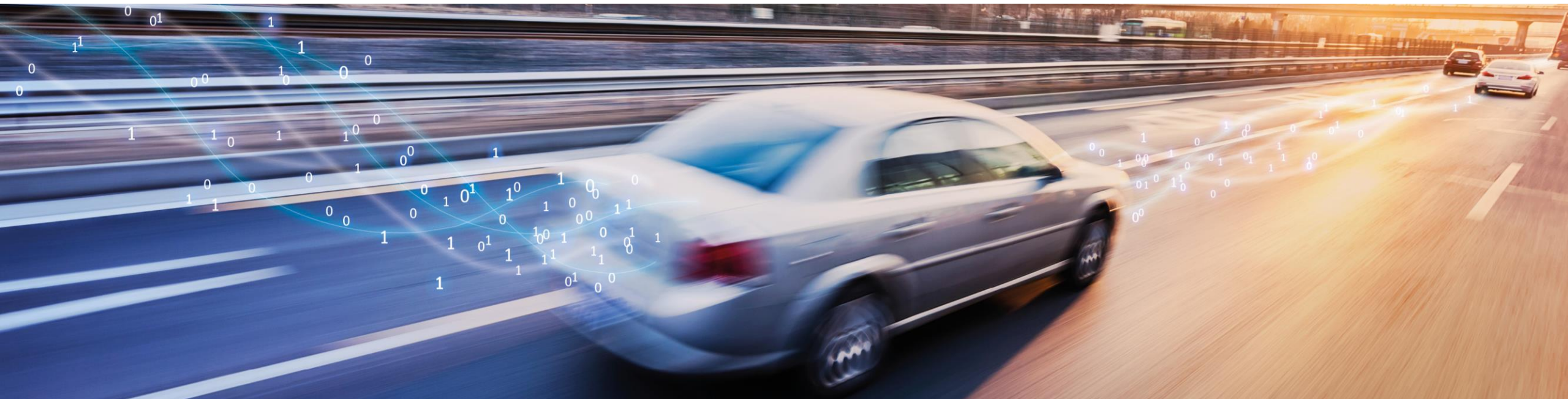
ASAM AE MCD-1 XCP SW-Debug V1.1

Release Presentation

Dr.-Ing. Ralf König

20.07.2021

Principal Software Engineer, Vector Informatik GmbH



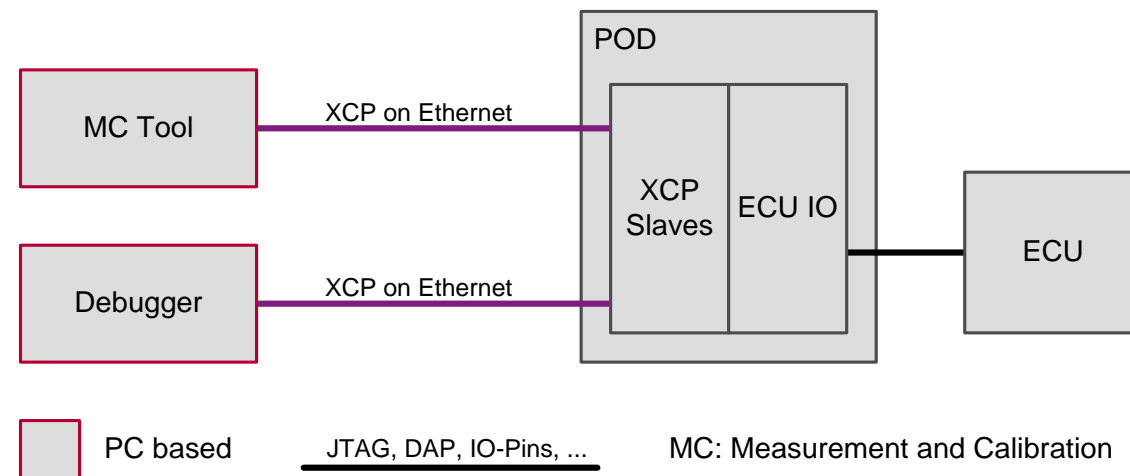
Agenda

- 1 Introduction**
- 2 Motivation for New Release**
- 3 New Features**
- 4 Other Changes**
- 5 Backward-Compatibility**
- 6 Relation to Other Standards**
- 7 Deliverables**

Introduction

Software debugging as well as measurement data acquisition and calibration are essential techniques used during all stages of electronic control unit (ECU) development.

In the past, these techniques typically have been used apart. Version 1.0 of the Software Debugging over XCP standard defines methodologies as a solution to this problem. Using technologies described in the standard enables debuggers and thus end users to perform debugging tasks on the ECU, either in an exclusive way or in parallel to a MC tool. The below shows such a scenario in combination with a Plug on Device (POD). The standard has meanwhile been adopted by debugger and POD vendors and has successfully been established in the field.



Motivation

Since the release of version 1.0 of the standard the number of Arm based ECUs is steadily increasing. However, support for Arm based microcontrollers has not been considered at that time. Taking the importance of Arm based ECUs and their support into account demands for extension of the Software Debugging over XCP standard.

Furthermore, based on the experiences gained from the initial version of the standard the need was identified to improve the Target Resource Identifier handling. Version 1.0 of the standard did not foresee a generic mechanism for disclosing the XCP slave specific Target Resource Identifier mapping to an XCP master. However, this is required to improve interoperability and to simplify support of target devices.

In some use cases, the ECU flash mechanisms specified in the XCP standard cannot be used. Rather than that, low level access to debug interfaces or other dedicated interfaces for ECU flashing is required. The low level access mechanism defined in version 1.0 of the standard cannot be used for flashing Renesas RH850 targets using the Serial Communication Interface. A solution therefore is required.

New Features (1)

Disclosing of the Target Resource Identifier (TRI) to the XCP master

- Chapter 3.6: added new introductory chapter dedicated to TRI
- Chapter 4.2.21
 - new command with which the XCP master can request the TRI mapping from the XCP slave
 - description of general methodology
 - definition of requirements for backward compatibility to version 1.0
 - added command specific error handling in chapter 5.2
- Appendix: C. Explanation of Target Resource Identifier
 - complete revision of chapter to reflect the new TRI methodology
 - added Arm Devices
- Chapter 7.10: XCP sequence examples for obtaining TRI mapping from XCP slave
- Tool: examples for generating and compatibility testing the Target Resource Identifier descriptor table

New Features (2)

Serial Wire Debug (SWD) Support for Arm targets

- Extension of dialect parameter table (Table 14) by SWD
- Appendix B.2: mapping between the Low Level Telegram (LLT) command and SWD packets
- Chapter 7.8: XCP sequence example for SWD over LLT

Renesas RH850 U2A Serial Communication Interface (SCI) Support

- Extension of dialect parameter table (Table 14) by SCI
- Chapter 4.1.22: new Low Level Byte Telegram (LLBT) command providing low level access mechanisms to byte-oriented debug interfaces
- Appendix B.3: mapping between the LLBT command and SCI
- Chapter 7.11: XCP sequence example for SCI over LLBT

Other Changes

New ASAM Word template

- Update of specification document to comply to the new ASAM Word template

Rearrangement of chapters

- Interchange of chapter 3.4 and chapter 3.5

Backward Compatibility

ASAM MCD-1 XCP SW-Debug V1.1 is compatible to ASAM MCD-1 XCP SW-Debug V1.0.

Relation to Other Standards

Dependencies on ASAM MCD-1 XCP V1.5

- XCP-resource Debugging (DBG)
- Subcommand space for *Software Debugging over XCP*
- XCP-event space for *Software Debugging over XCP*
- XCP-error space for *Software Debugging over XCP*
- Seed & Key mechanism for XCP-resource DBG

Deliverables

Documents

- XCP Associated Standard: ASAM_AE_MCD-1-XCP_AS_SW-DBG-over-XCP_V1-1-0.pdf

Tools

- Visual Studio project and examples for generating and compatibility testing the Target Resource Identifier descriptor table

```
TargetResourceIdentifier.zip
|  soc400-example_tri_descriptor_blob.bin
|  soc600-example_tri_descriptor_blob.bin
|  soc600-example_tri_descriptor_blob.hex
|
|---xcpdbg_tri_desc
|  makefile
|  readme.txt
|  tri_desc.cpp
|  tri_desc.h
|  xcpdbg_tri_desc.sln
|  xcpdbg_tri_desc.vcxproj
|  xcpdbg_tri_desc.vcxproj.filters
|
|---script
|  decode.bat
|  example1.bat
|  example1.bin
|  example1.txt
|  example2.bin
|  examples.bat
```