

# Integrating an XCP driver in ASCET build environment

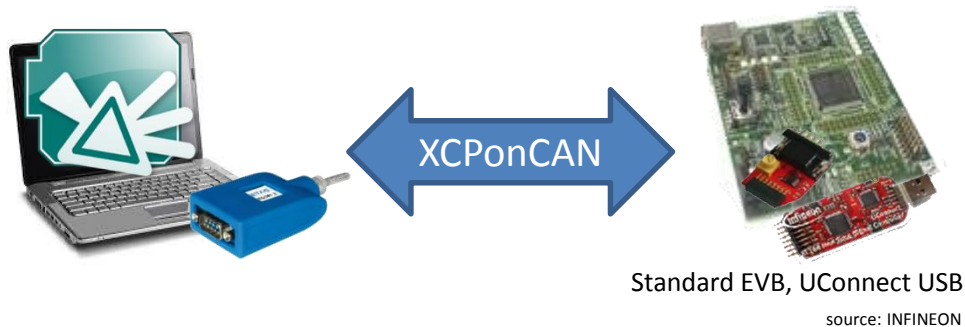
## Setting up real time hardware with ASCET (II)

### Do you know this challenge?

In the last paper "Integrating evalboard hardware in ASCET build environment – Setting up real time hardware with ASCET (I)" we showed how to integrate hardware specific initialization and drivers (CAN) as external c code into the ASCET build environment, which allows to attach the EVB to a CAN bus like an ECU. However, in doing so, the software running on the target can only be accessed via debugger. Instead, we want to have easy measurement access using INCA.

### Our Solution – integration of an XCP stack into the ASCET-SE build environment

To enable the use of the EVB within a typical automotive environment using standard tools like INCA, we now also integrate the ETAS XCP ECU software into the ASCET build system. This will add the XCP protocol on top of the CAN stack and, together with the automatically generated A2L file, allows XCPonCAN based measurement on the target board using INCA.



### Technical Details

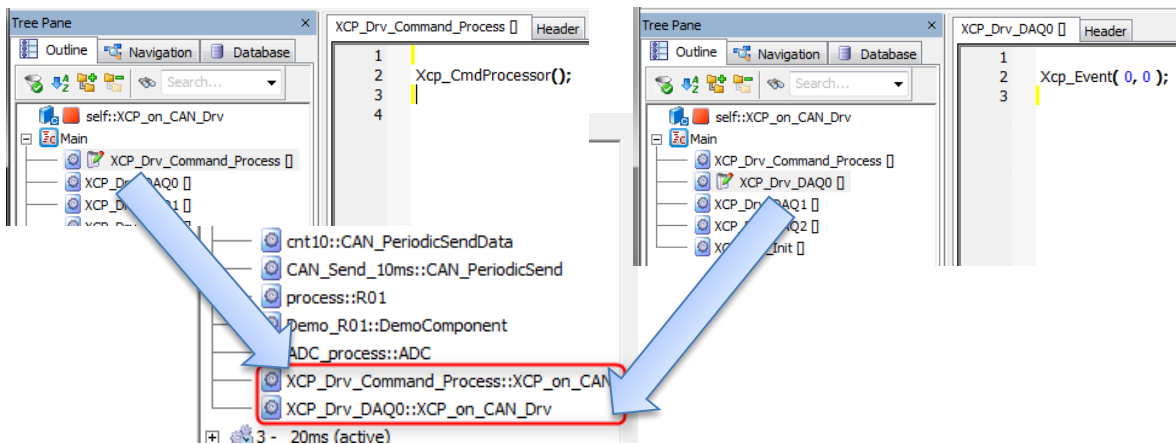
The ETAS XCP ECU software provides a configurable XCP stack with interfaces to CAN or Ethernet and also creates A2L templates for XCP based on the configuration. For configuration details, please see the documentation of the ETAS XCP ECU software which can be found on the [ETAS Website](#).

The generated code is integrated into the ASCET build environment as C code in the same way as the board specific CAN drivers. The CAN drivers are extended to provide additional CAN identifiers for XCP commands and DAQs.

The real time OS in the ASCET project is also configured for the XCP command processor and the DAQs. This means that the XCP command processor must be called in a faster raster, and the DAQ processors in timer rasters that match their sample times. To do this, the driver calls are mapped to processes in an ASCET C code module and assigned to the OS tasks.

# Integrating an XCP driver in ASCET build environment

## Setting up real time hardware with ASCET (II)



The A2L fragments generated by the ETAS XCP ECU software generator are added to the ASCET A2L generation. For this, we are changing

a) the *aml\_template.a2l* to load AML files for XCP in :

```

/begin A2ML
    /include XCP_v1_0_definitions.aml"
    block "IF_DATA" taggedunion if_data { /include XCP_v1_0.aml"};
/end A2ML

```

b) The *if\_data\_template.a2l* to load the IF\_DATA XCP section generated by the ETAS XCP ECU software:

```

/include "Example_XE162/XCPonCAN/Config/if_data_xcp_session0.a2l"

```

The ASCET build environment will build and compile the code to a hex file and will also generate an A2L file containing the XCP interface description and the information for the application software variables. INCA can import the two files as ECU project and can be set up to measure variables on the target EVB. The CAN on the EVB can be accessed using the USB2CAN Interface ES581 or any of the ES59x devices.

### Next Steps

1. Flash the UConnect with the hex file and
2. Import the hex and a2l file generated by the ASCET build into INCA, set up an XCPonCAN workspace and start measurement experiment

### Tools used

- INCA: 7.2.0
- ASCET-MD/SE XCV2 v6.4.1, ETAS XCP ECU software v1.0.5, Perl
- Hardware: ES581, UConnect XE162N (Infineon) or similar

### Your ETAS Contact

Markus Gebhardt; +49 711 3423 - 2278  
[markus.gebhardt@etas.com](mailto:markus.gebhardt@etas.com)