Infineon AURIX: DAP over CAN (DXCPL) Configuration

Technical Note

September 2023

This Technical Note describes how to configure DAP over CAN Physical Layer (DXCPL) for TC2xx, TC3xx and TC4xx microcontrollers.

Requirements:

- ✓ winIDEA 9.21.81 or newer
- ✓ BlueBox iC5700
- ✓ DAP over CAN Physical Layer (DXCPL) Converter, Infineon DAP/DAPE Active Probe II or 10-pin 1.27mm Infineon DAP2 Wide Debug Adapter

DAP over CAN Physical Layer Converter (DXCPL) enables debugging via the regular CAN pins of the ECU connector without opening its housing. This connection is electrically robust due to the nature of the CAN bus. The DXCPL Converter translates Infineon AURIX™ SPD (Single Pin DAP) encoded DAP messages to the CAN bus physical layer. Hardware solution supports DXCPL on Infineon AURIX™ TC2xx, TC3xx and TC4xx devices.

Target system requirements

TC4xx

There are several requirements that need to be satisfied for the DXCPL to be operational.

- CAN01 (P14.0 and P14.1) or CAN00 (P33.12 and P33.13) pins have to be connected to the CAN connector
- External CAN transceiver has to be used
- Hardware is designed in such a way that the TRST pin is **high** at the time of the PORST pin release; Note that the debug tool doesn't have access to the PORST pin.
- Correct mode of operation is ensured in the UCB_CS_DBG_*.TRI_DXCCTR
- DXCPL is not disabled by user application by setting the SYSCON.DDC or TRI_DXCCTR.DXCPLDIS bit

TC3xx

- CAN1 (P14.0 and P14.1) or CAN0 (P33.12 and P33.13) pins have to be connected to the CAN connector
- External CAN transceiver
- TRST pin is **low** at the time of the PORST pin release. Note that the debug tool doesn't have access to the PORST pin
- Correct mode of operation is ensured in the UCB_DBG_*.DMU_HF_PROCONDBG.TIC bitfield
- DXCPL is not disabled via your application by setting the SYSCON.DATM bit

TC2xx

- CAN1 pins (P14.0 and P14.1) have to be connected to the CAN connector
- External CAN transceiver
- TRST pin is **low** at the time of the PORST pin release
- DXCPL is not disabled by Start-up Software (SSW) by setting the PROCONWOPp.DATM bit
- DXCPL is not disabled via your application by setting the SYSCON.DATM bit

DXCPL is not intended to be used on live systems since it forces all non-DXCPL capable nodes into a disabled state.

The bit STSTAT.SPDEN can be checked via your application to check if DXCPL is active and possibly ignore any CAN related errors or alarms.

Configuration

1. Configure a new workspace via *File / Workspace / New Workspace.*

2. Select DXCPL as Debug channel Mode in *Hardware / CPU Options / SoC.*

Connecting to the device at power-on

This is the recommended way to gain debug access to the device. It is mandatory to use when either:

- Debug password is used in *CPU Options / SoC*
- The MCU does not have a valid code programmed

> *TC3xx, TC4xx: This mode of operation is only available when CAN1 is reserved for DXCPL. With the alternative pin set, this mode is not possible.*
>
> *TC2xx: When this mode is used, it is mandatory for the CAN transceiver to be enabled at the time of PORST release.*

1. With the target device not powered, Execute *Debug / Reset.*

Debug Status will be SoC ATTACHING .

2. Apply power to the device.

The MCU will enter the Debug Status STOP and the debug code can be debugged.

Connecting to the already powered device

In this mode of operation, you can use winIDEA as you normally would, but with certain restrictions:

- Debug right after Power-On-Reset is not possible
- The MCU needs to have a valid code programmed
- Debug password should not be used in *CPU Options / SoC*

More resources:

- DAP over CAN User Manual
- CPU Options Setup