Technical Note

# Infineon AURIX TC2xx: HSM programming

May 2023

This Technical Note describes a procedure of programming an HSM application through winIDEA on Infineon AURIX TC2xx devices.

Tool requirements:
- ✓ winIDEA 9.21.0 or newer
- ✓ BlueBox iC5700, iC5000
- ✓ Infineon AGBT Active Probe or Infineon DAP/DAPE Active Probe or 10-pin 1.27 mm Infineon DAP2 Wide Debug Adapter or 22-pin ERF8 DAP2 Debug Adapter

The HSM (Hardware Security Module) is an optional module available on selected AURIX devices. Through winIDEA you can program an HSM application and configure the UCB (User Configuration Block) which is required to enable the HSM on your AURIX device.

*HSM application programming and UCB configuration requires caution because a misconfiguration can potentially lock your chip.*

*It is recommended to use Image checker during UCB or HSM code programming. Please note that a separate license is needed for Image checking.*

## winIDEA Configuration

The proper procedure to program your HSM application can be summarized in two steps:
1. Program the TriCore application and the HSM code.
2. Enable the HSM in the UCB.

### Program the TriCore application and the HSM code

1. Prevent chip locking by misconfiguration via *Hardware / CPU Options / Debugging / Image checker / Reject programming*.

2. Verify that you have selected the program files for the regular TriCore application(s) as well as the program file for the HSM via *Debug / Configure session / SoC / Program Files*.

3. Your download files must contain at least one valid boot mode header and valid HSM code before you proceed to enabling the HSM. If you want to see what is loaded into the target memory before you actually flash any code to your target, you should activate *Tools / Demo mode*. While in this mode, perform a *Debug / Download (Ctrl+F3)* and use the *Load Map* option via *Debug / Show Load Map* to inspect the contents of the BMHD locations (BMHD0 - BHMD3).

4. Disable *Demo mode* if the BMHD and HSM code are populated correctly and perform a Download to program the code onto your target.

5. Before proceeding, check the validity of the programmed boot mode header by making sure that the CPU0 boots correctly.
   a. Disable presenting the CPU0 program counter by setting *Hardware / CPU Options / Cores / CPU0 / Preset PC after stopped in init* to *Do not preset*.
   b. Reset the CPU via *Debug / Reset* and observe its reset vector.

*Enabling the HSM without having a valid boot mode header programmed can permanently lock your chip!*

## Enable the HSM in the UCB

1. Enable writing to the UCB via *Hardware / Options / Programming* by checking *Infineon TC2xx_UCB_16kB*.

*The amount of writes to the UCB is limited. For the exact number of writes, refer to your AURIX device reference manual. Uncheck the UCB memory device once the UCB is programmed.*

2. Establish a new session via *Debug / Load Symbols Only*.

3. Open the UCB plugin window via *View / [TC2xx] Aurix / UCB*. Double click on the **UCB_HSMCOTP** register. Change the values of the following registers:

- **PROCONHSMCOTP_x**: 0x01 or 0x781; Select the value and use it for all these registers.
- **CONFIRMATION_x**: UNLOCKED (0x43211234)

*Recommended values for **PROCONHSMCOTP_x** during development are: 0x01 or 0x781. Other bits of this register should only be changed with extreme care, as you can lock the HSM flash, prevent start-up or disable debug access. Setting **CONFIRMATION_x** to CONFIRMED makes the whole UCB sector OTP protected from programming.*

After you click OK, the BlueBox programs **UCB_HSMCOTP**.

In case of any security violation, programming is rejected. This completes the configuration. After the power-on reset, the newly entered UCBs are read by the CPU and the HSM is started accordingly.

## More resources in winIDEA Help

- UCB plugin
- Image checker
- Demo mode
- Load Map
- UCB programming