EXAMPLE NAME: CRC_SW_EXAMPLE_XMC45

OVERVIEW: This example uses the SW CRC APP to check the integrity of an incoming data frame. The incoming frame is a 9-bytes frame (8bytes data + 1byte CRC checksum).

DESCRIPTION:

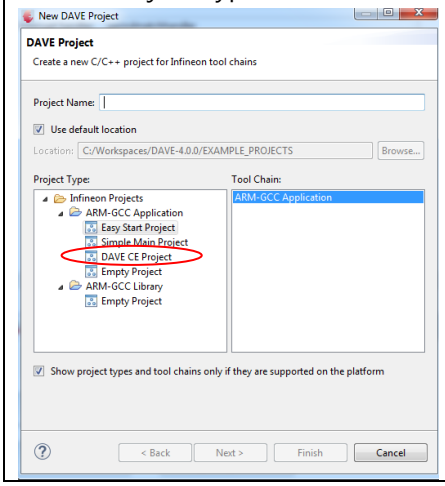
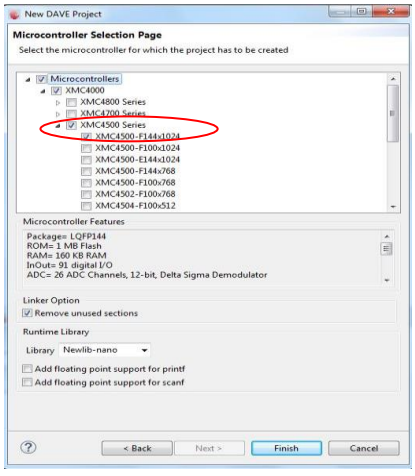
- CRC_SW APP is used to calculate a CRC checksum based on a pre-determined polynomial. No algorithm control is selected for the CRC calculation.
- UART APP is used to receive the incoming data frame. The incoming frame is a 9-bytes frame (8bytes data + 1byte CRC checksum). The 9th byte is the checksum byte.
- SYSTIMER APP is used to generate a period SW timer of 0.5 seconds interval. It toggles the LED if checksum is matching else LED is turned off.
- DIGITAL_IO APP is used for LED toggling.
- Main program polls for a 9bytes data frame. The incoming frame (8 bytes data) is used to calculate the CRC checksum. The result is used to compare against the checksum byte of the data frame. If result is matching, the LED toggles every 0.5 seconds, else LED is turned off.

REQUIRMENTS:

Boards Required: XMC4500 CPU Board (Order Nr: KIT_XMC45_EE1_002)

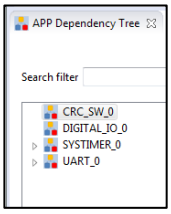
HOW TO CREATE THE PROJECT:

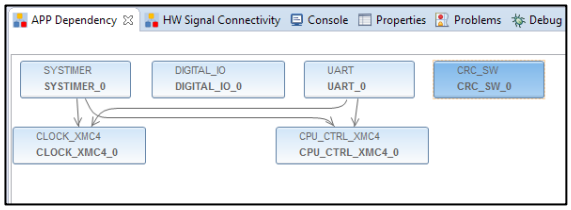
1. Open the DAVE CE and use  "Add IDE New Project Wizard" on the toolbar to add a new DAVE Project.

Enter Project Name Select "Project Type ➔ DAVE CE Project"	Select Microcontroller ➔XMC4500 Series ➔ XMC4500-F144X1024
	

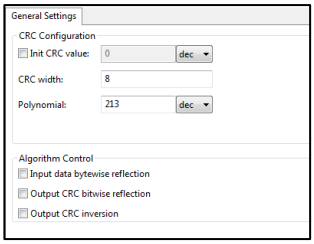
2. Use the  "Add New APP" in the toolbar to add APPs listed. Configure the App instances with the following configurations.

Views: App Dependency Tree and App Dependency



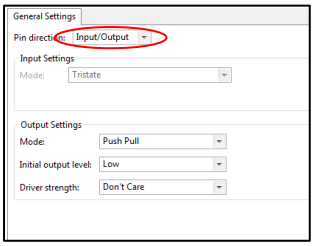


CRC_SW_0:

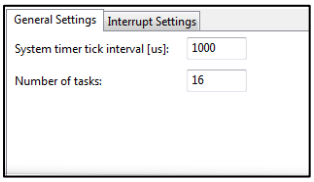


⇨ No change to tab; Default settings taken.

DIGITAL_IO_0:

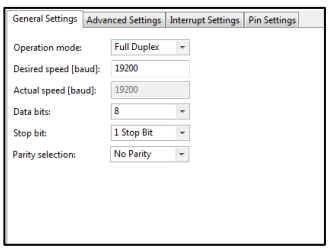


SYSTIMER_0:



⇨ No change to tab; Default settings taken.

UART_0:

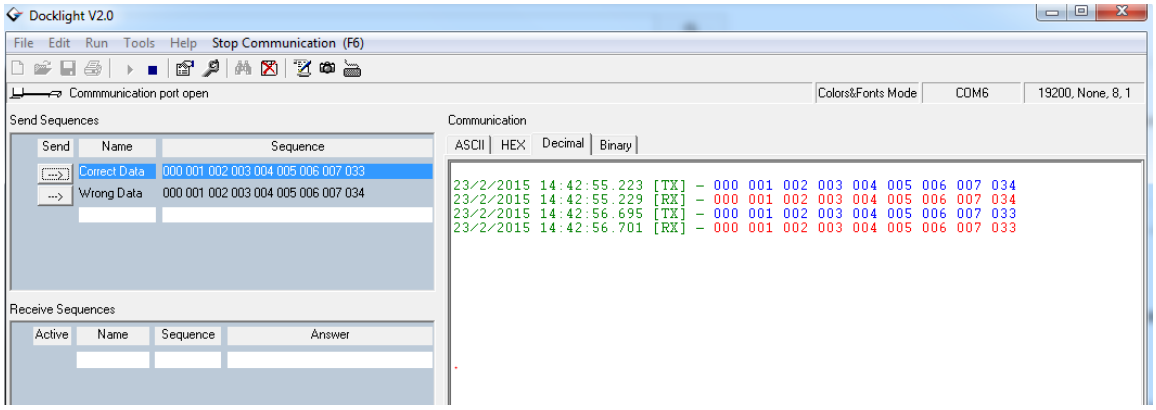


⇨ No change to tab; Default settings taken.

3. Use the  "Manual Pin Allocator" found in the toolbar, configure the output pin for the LED and UART pins.

App Instance Name	Resource	Port-Pin	Pin Number
◢ DIGITAL_IO_0			
	pin	(P3.9)	#12
◢ UART_0			
	Receive Pin	(P1.4)	#108
	Transmit Pin	(P1.5)	#107

4. Generate the code for configurations made and change main.c accordingly.
5. Build and download to the microcontroller.
6. Using a terminal program (e.g. Docklight, MTTY) to send the data to the microcontroller.



Frame Correct Data: 000 001 002 003 004 005 006 007 **033**

Frame Wrong Data: 000 001 002 003 004 005 006 007 **034**

**checksum is highlighted*

HOW TO TEST:

Download and run the demo in the uC.

OBSERVATIONS:

- When a "Frame Correct Data" is sent, LED toggles at 0.5 seconds interval .
- When a "Frame Wrong Data" is sent, LED turns off.

HINTS WHEN MIGRATING EXAMPLE TO OTHER DEVICES IN THE SAME XMC FAMILY:

No additional information is required.