EXAMPLE NAME: CAN _EXAMPLE_XMC45

OVERVIEW: This example shows transmission and reception of message objects between two CAN nodes of the MultiCAN peripheral in Loop-Back Mode.

DESCRIPTION:
CAN_NODE APP (Two instances) configures two nodes of the MultiCAN peripheral to send and receive messages between themselves using the internal CAN bus of the XMC4500 device (LoopBack).

Request_Node is used to transmit a data frame in Loop-Back Mode with "User defined Message Id" (0x255). This is then expecting to receive a data frame with the "User defined Message Id" (0x355).

React_Node is used to receive the frame with the ID0x255 and on receiving (Receive Interrupt) the data frame, message with ID 0x355 is transmitted in Loop-Back Mode.

INTERRUPT APP (Two instances) configures the different NVIC nodes connected to the respective CAN_NODE.

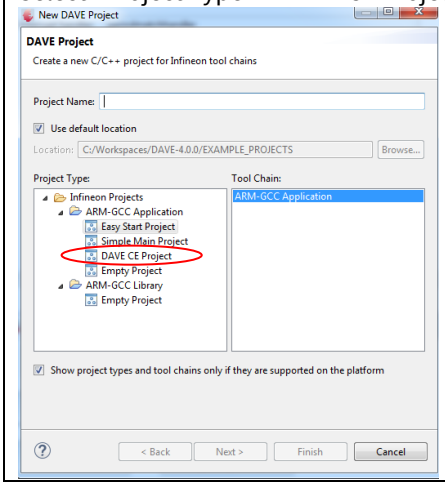
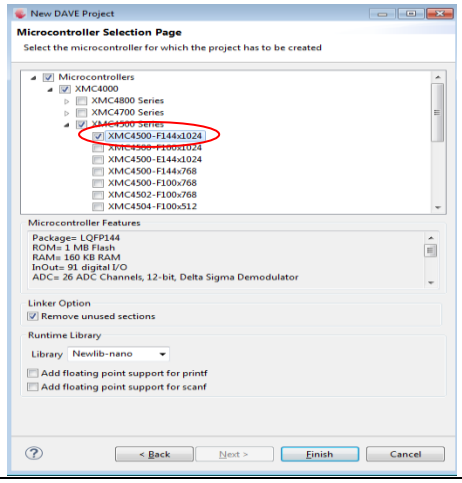
DIGITAL_IO APP is used to show the correct transmission of the Message Objects.

REQUIRMENTS:

Boards Required: XMC4500 CPU Board

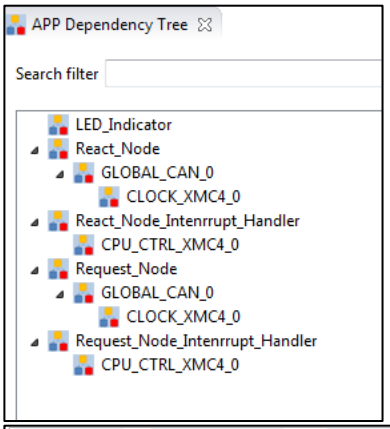
HOW TO CREATE THE PROJECT:

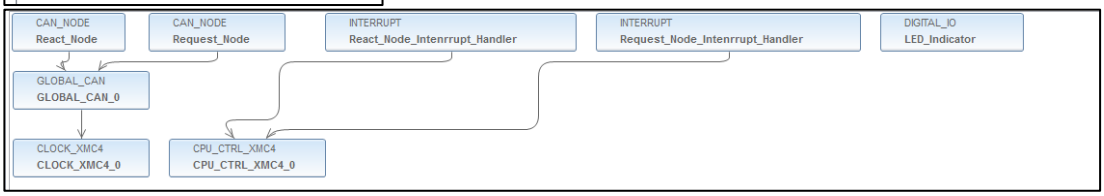
1. Open the DAVE CE and use  "Add IDE New Project Wizard" on the toolbar to add a new DAVE Project.

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

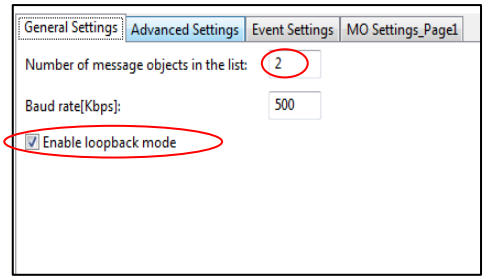
2. Use the "Add New APP" in the toolbar to add 1 instance of CAN_NODE. Configure the APP instance with the following configurations.

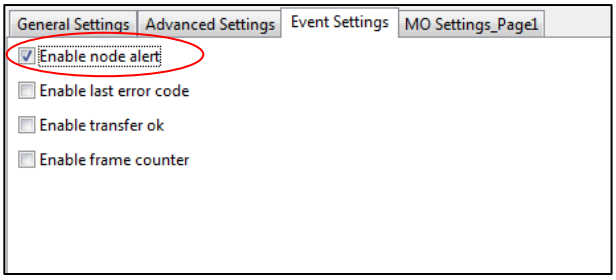
Views: APP Dependency Tree and APP Dependency

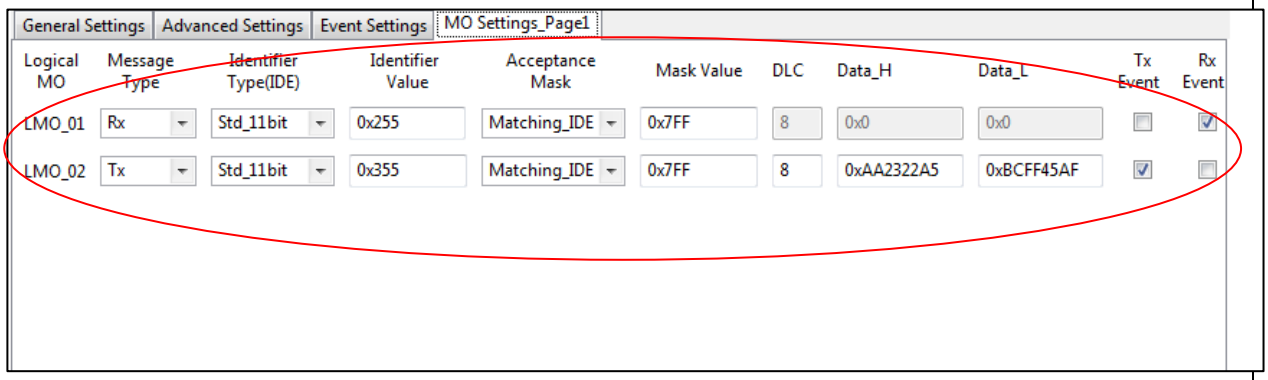




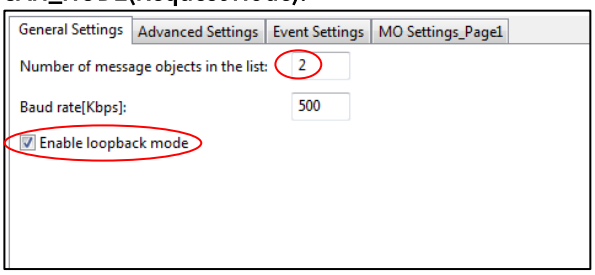
CAN_NODE(React Node):

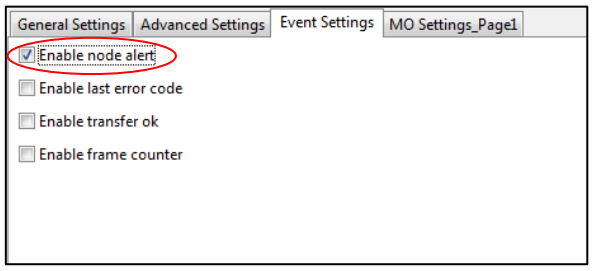


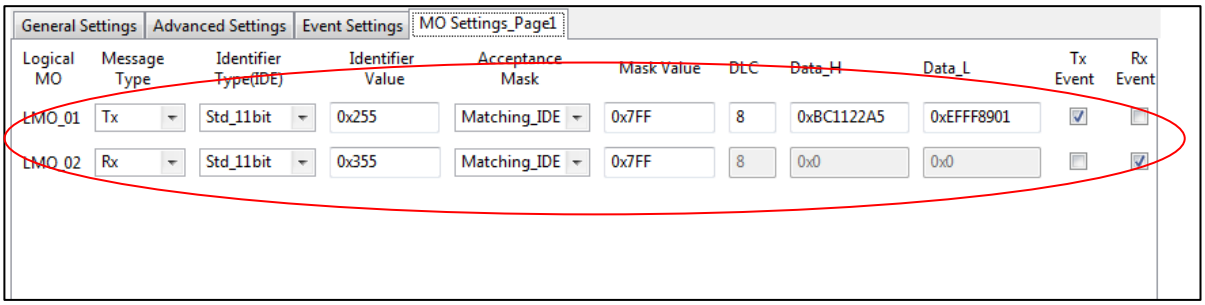




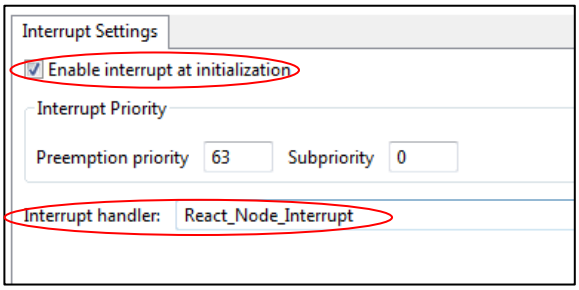
CAN_NODE(Request Node):



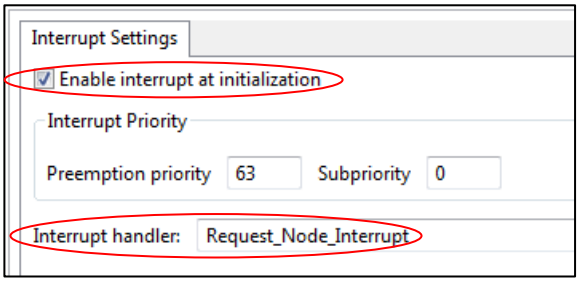




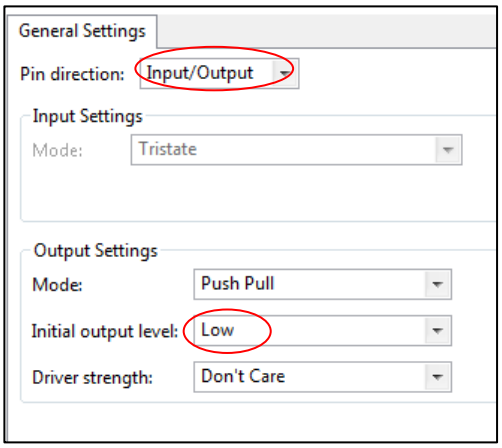
INTERRUPT (React_Node_Interrupt_Handler):



INTERRUPT (Request_Node_Interrupt_Handler):

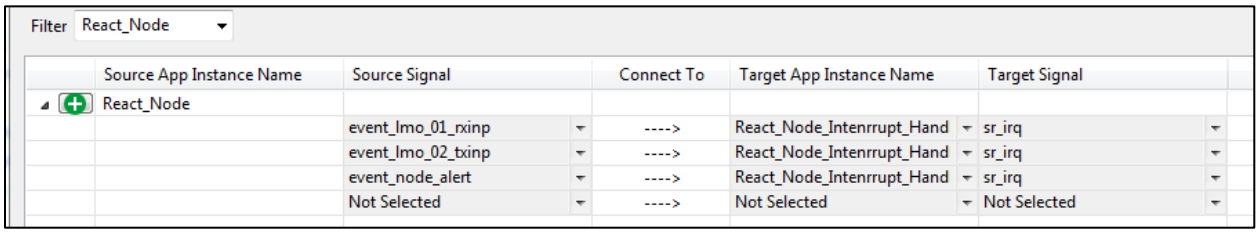


DIGITAL_IO(LED_Indicator):

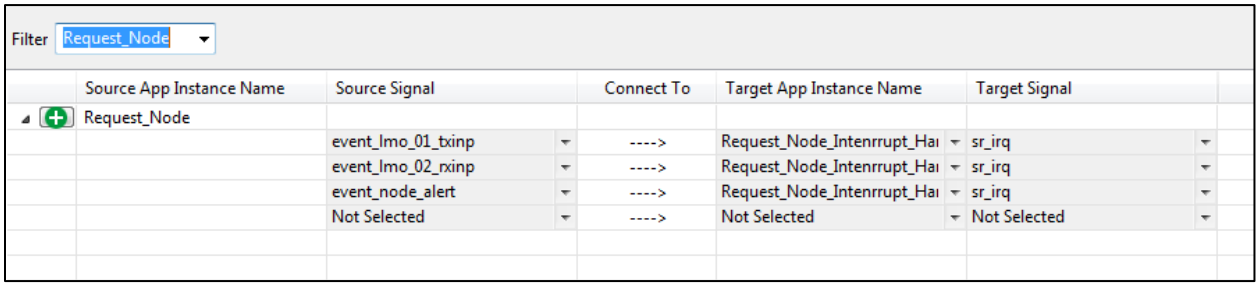


3. Use the "Hardware Signal Connections" found at the bottom right of the APP UI window, to configure the signal connections for the respective APPs.

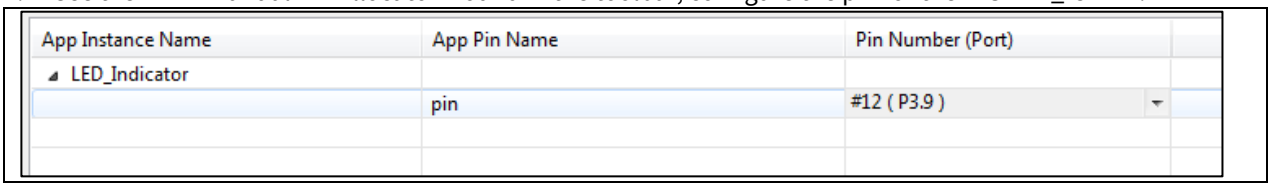
CAN_NODE(React Node):



CAN_NODE(Request Node):



4. Use the  "Manual Pin Allocator" found in the toolbar, configure the pin for the DIGITAL_IO APP.



5. Generate the code for the configurations made and change main.c accordingly.

6. Build and download to the microcontroller.

HOW TO TEST:

Download and run the demo in the microcontroller.
LED connected to Pin 3.9 will switch on when Request Node receives message from the React Node .