PSoC 4 CAN Basic Example Project

1.0

Features

- Configures Transmit and Receive mailboxes in Basic CAN Mode

General Description

This example project demonstrates how to configure the CAN component to transmit and receive messages over the CAN bus in the Basic CAN mode.

This is only one part of the CAN example project. Use this example along with CAN_Full_P4_Example for complete demonstration.

Development kit configuration

This example project is designed to be executed on CY8CKIT-044 from Cypress Semiconductor. A full description of the kit, along with more example programs and ordering information, can be found at http://www.cypress.com/go/cy8ckit-044.

Also, the example project requires CY8CKIT-017 CAN/LIN Expansion Board kit. A full description of the kit, along with more example programs and ordering information, can be found at http://www.cypress.com/?rID=40215.

To use a CY8CKIT-017 CAN/LIN Expansion Board kit with a CY8CKIT-044/CY8CKIT-046 kit the following connection has to be done:

1. Connect the GND and V5_0 pins of the CY8CKIT-017 kit to appropriate GND and V5.0 pins on header J1 of CY8CKIT-044/CY8CKIT-046 kit.
2. Connect the C_RX and C_TX pins of the CY8CKIT-017 kit to pin P0[0] and pin P0[1] appropriately on header J2 of CY8CKIT-044 kit or on J18 header of CY8CKIT-046 kit.

The CY8CKIT-017 CAN/LIN Expansion Board kit should be used with the following installed jumpers: JP2 (CAN Termination Resistor), CANEXTPWR and JP6 set to Vdd-V5_0 (default setting for 5V operation). Any jumper on the board not mentioned above should have no jumper installed.

1. Build the project and program the hex file into the target device.
2. Set 3.3V position on header J9.
3. Connect the precise external clock source (with accuracy ±0.2%) to P0[6].
4. Connect the voltage source to pin P2[0].
5. Power cycle the device and observe the results using a USB-UART bridge.

6. A CAN – USB analyzer can be used to analyze the data traffic.
7. An oscilloscope can be used to verify the PWM out on port 0[2].

The project requires configuration settings changes to run on other kits from Cypress Semiconductor. Table 1 is the list of the supported kits. To switch from CY8CKIT-044 to any other kit, change the project's device with the help of Device Selector called from the project's context menu.

Table 1. Development Kits vs Parts

Development Kit	Device
CY8CKIT-044	CY8C4245AXI-483
CY8CKIT-046	CY8C4248BZI-L489

The pin assignments for the supported kits are in Table 2.

Table 2. Pin Assignment

Pin Name	Development Kit	
	CY8CKIT-044	CY8CKIT-046
RX	P0[0]	P0[0]
TX	P0[1]	P0[1]
UART:rx	P7[0]	P3[0]
UART:tx	P7[1]	P3[1]
V1	P2[0]	P2[0]
PWM_OUT	P0[2]	P0[2]
Switch1	P0[7]	P0[7]

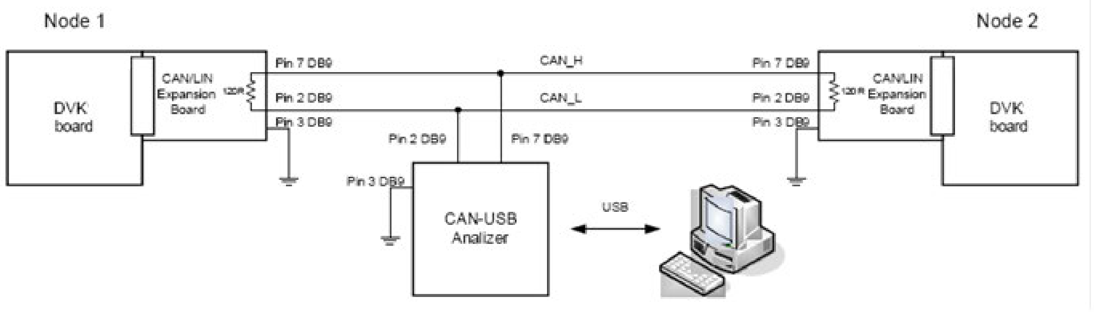
Project Configuration

The example project consists of the CAN, ADC_SAR, TCPWM (PWM mode), SCB (UART mode), and Interrupt components.

CAN received and transmitted data, the ADC output, and PWM pulse width can be verified using a USB-UART bridge. To ensure proper functioning of the examples projects, you should create a mini network from at least two CAN nodes as shown in Figure 1. The network as shown in Figure 1 also includes the CAN-USB analyzer to analyze the data traffic.

Note that P0[6] is shared pin for an external clock source and the RGB LED for the CY8CKIT-044 kit. If you want disable the LED then please remove R16 resistor.

Figure 1. Test CAN Network Topology



Project Description

This example illustrates how to transmit and receive messages using the CAN component.

In this project, the CAN component is configured to transmit two messages over the CAN bus:

Message 1: Status of Switch1 - Sent whenever there is a change in the status.

Message 2: ADC data - Sent every 100ms.

The component is also configured to receive data which is used to set a pulse width of the PWM used in the project. Both transmitted and received data can be verified using a USB-UART bridge.

Every 100ms ADC data measures by the node and sends over the CAN bus to the remote node.

At button connected to Switch1 press – Message 1 (with status of Switch1) sends to remote node and sends over the UART state of Switch1. After that remote node increments value of PWM pulse width by ten and sends back updated PWM pulse width value. The node set new PWM pulse width value and sends over the UART. Please note that PWM pulse width isn't being sent before first button connected to Switch1 press.

All transmitted and received data sends over the UART for both nodes.

Expected Results

Program the device with the project and observe the ADC data, PWM pulse width and state of Switch1 (pressed or released), using a USB-UART bridge. PWM out can be verified on port 0[2].

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com