CC2480 Software Examples

User's Guide

Table of contents

1 Introduction

This document describes software examples for the CC2480 ZigBee Network Processor, a Smart Transceiver. It also describes the necessary hardware and software to run the examples, and how to get started. A comprehensive description of the hardware Z-Accel is found in the ez430 – RF2480 Demo Kit User's Guide. There is also a ez430-RF2480 Quick Start Guide which gives a short tutorial on how to get started with the kit. These documents are found on the Z-Accel Demonstration Kit website [3].

Section 3 of this document describes necessary prerequisites and how to get started with the code examples. Section 4 describes how to run each of the application examples. The software library that the code examples are built upon is described in section 5. The latter section also gives an API reference and describes the functionality of the software library.

2 Abbreviations

API	-	Application Programming Interface
HAL	-	Hardware Abstraction Layer
IO	-	Input/Output
RPC	-	Remote Procedure Call
SAPI	-	Simple API
SPI	-	Serial Peripheral Bus
TB	-	Target Board
VCP	-	Virtual Com Port

3 Using the software

This section describes the necessary hardware and software, and how to get started with the application examples for CC2480.

3.1 Prerequisites

To successfully download and run the software described in this document, the following material is needed:

- 3 x CC2480 Target Boards
- 2 x ez430 Battery Board
- 1 x ez430 USB Emulator board
- IAR Embedded Workbench for MSP430 (Either Full version or Kickstart version can be used)
- 4 AAA batteries

A free, code size limited edition of IAR Embedded Workbench (IAR Kickstart) is available from the IAR Systems website (www.iar.com/ew430) or from the TI MSP430 homepage www.ti.com/msp430.

3.2 Getting started

The following sections describe hardware and software setup, how to program the board and how to run example code from the IAR debugger. A description of how to operate each software example is found in section 4 of this guide.

3.2.1 Set up Hardware and Software

Follow these steps to configure the hardware and software needed:

1. Install IAR Embedded Workbench for MSP430
2. Download the CC2480 Software Examples zip file (SWRU169.zip) from the kit website [3] and unzip this file.
3. Attach the ez430 – RF2480 to a USB port
4. If you are running Windows and using the USB emulator tool for the first time, you will be asked to install some drivers for the tool. For Windows XP, they are located in $IAR_INSTALL_DIR$\430\drivers\TIUSBFET\WinXP.

3.2.2 Program the board with IAR

IAR always reverts to “simulator” instead of “debugger” in the menu shown below when setting up a project for the first time. Check this the first time the project is setup

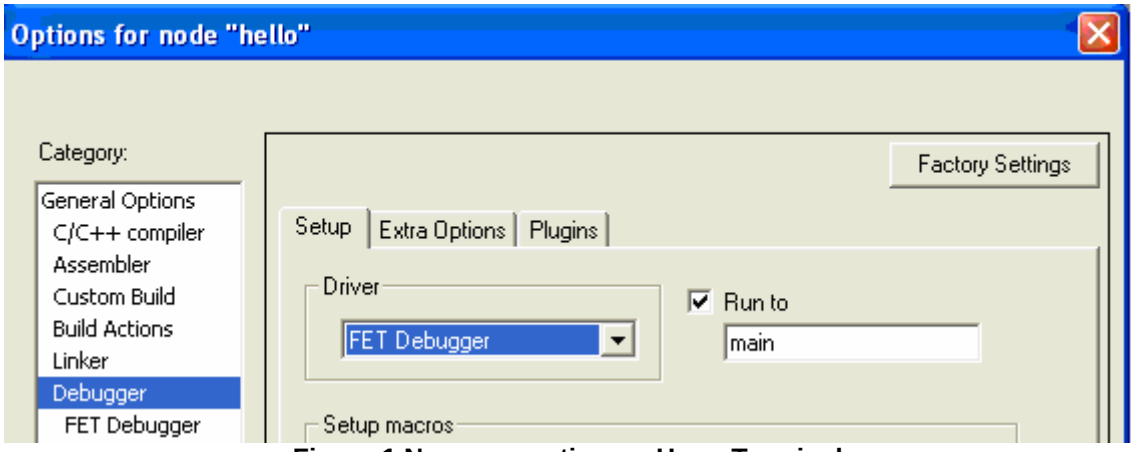


Figure 1 New connection on HyperTerminal

1. Open IAR Embedded Workbench
2. Open the workspace file **CC2480_SW_examples_src.eww** with IAR. This file is found in the folder **iar\ez430_rf2480** under the folder where the CC2480 Software Examples was unzipped.
3. Each application example has its own project tabs in the IAR workspace viewer. Select the project to be compiled in the workspace viewer of IAR. See section 4 for a description of the application examples.
4. Select **Project->Rebuild All**. This will perform a full rebuild on the selected project.
5. Select **Project->Debug.** IAR will now establish a connection with the target MCU, download the application and program the MSP430. The debugger will be started, halting the target at main().
6. Start the application by selecting **Debug -> Go**.
7. The board can be reset by selecting **Debug -> Reset**.
8. The debugger can be stopped by selecting **Debug -> Stop Debugging**.
9. The unit can now be operated independently from the debugger by disconnecting the FET tool and using the AAA batteries as power source. Cycle power with the jumper switch on the MSP430 Battery Board.
10. Repeat the steps 6 to 8 to program additional boards.

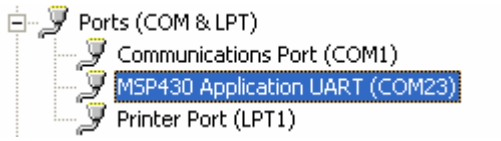
3.2.3 Peripherals used by the example code

The application examples utilize various peripherals on the CC2480 target board i.e. buttons, LEDs, Light Sensor and VCP. Please refer to Z-Accel TB User’s Guide for a description of these peripherals. This document is found on the Z-Accel website [3].

3.2.4 Configuring HyperTerminal

Some of the application examples outputs to the serial port. The PC HyperTerminal application can be used to display this output. Perform the following steps to configure the HyperTerminal:

1. Start Control Panel (Windows Key + Break) **Hardware** -> **Device Manager** -> **Ports** to verify the VCP for eZ430-RF2480 connection to select the correct COMx port for HyperTerminal



2. Start the HyperTerminal application. On Windows XP this is found under the **Start menu -> All Programs -> Accessories -> Communications -> HyperTerminal**
3. If the new connection window pops up (shown in Figure) enter a name of the new connection and click OK.

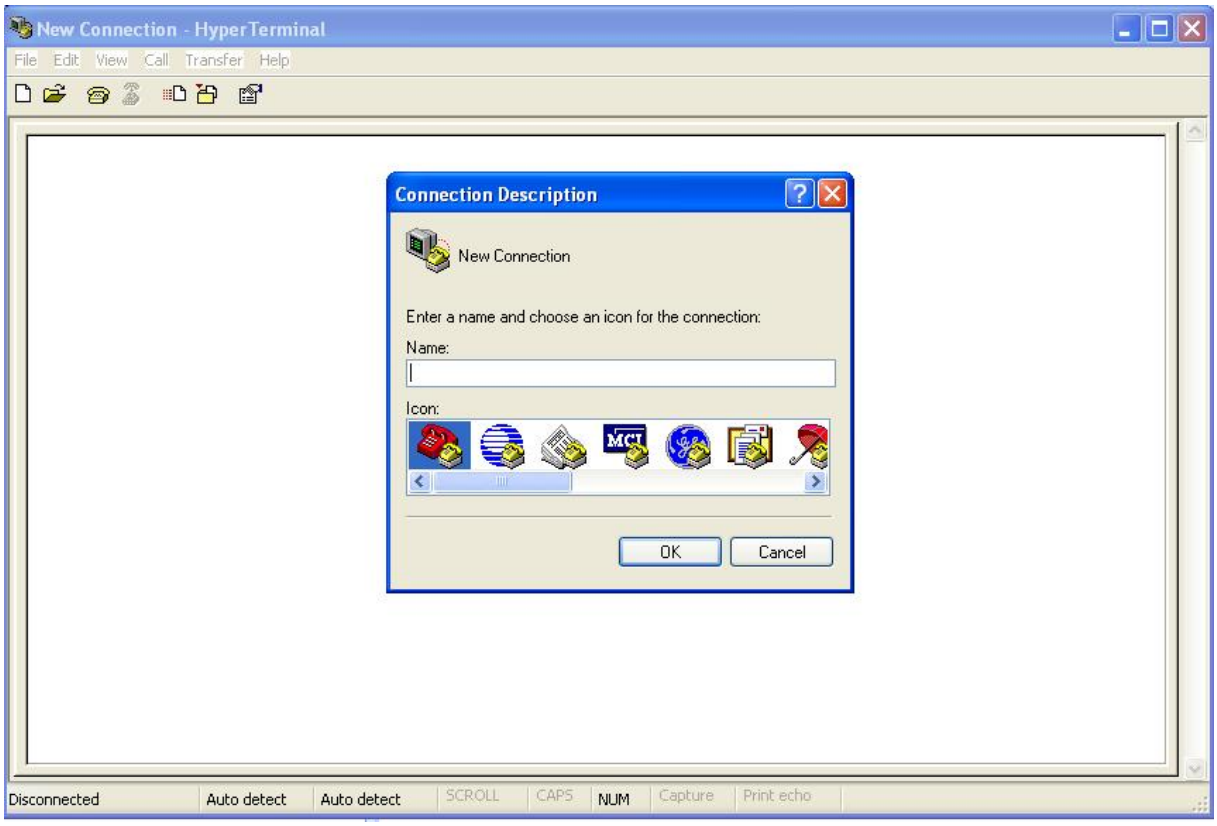


Figure 2 New connection on HyperTerminal

4. Choose the correct COM port on your PC.

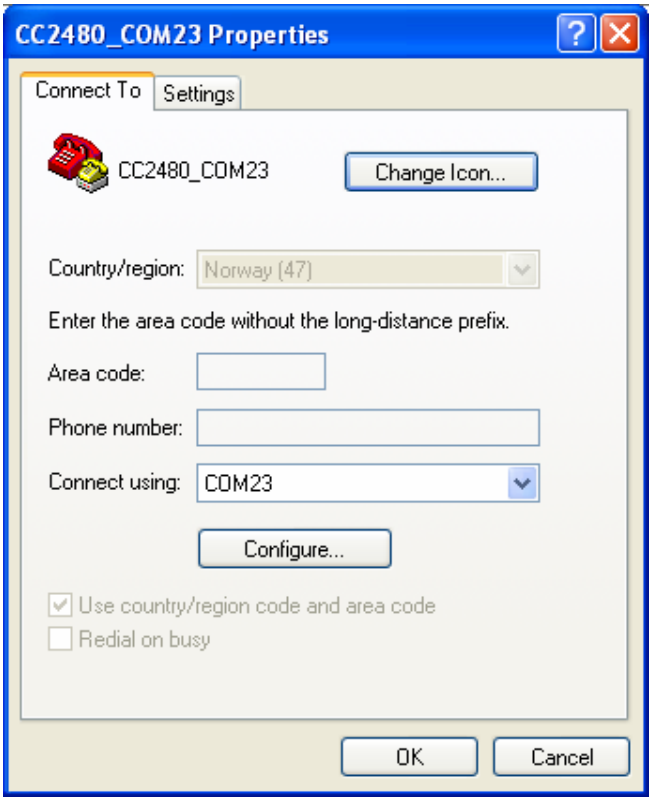


Figure 3 Choose correct COM port

5. Set COM port settings as shown in Figure ; Baudrate 9600 bits per second, 8 data bits, No parity, 1 stop bit and Flow Control None

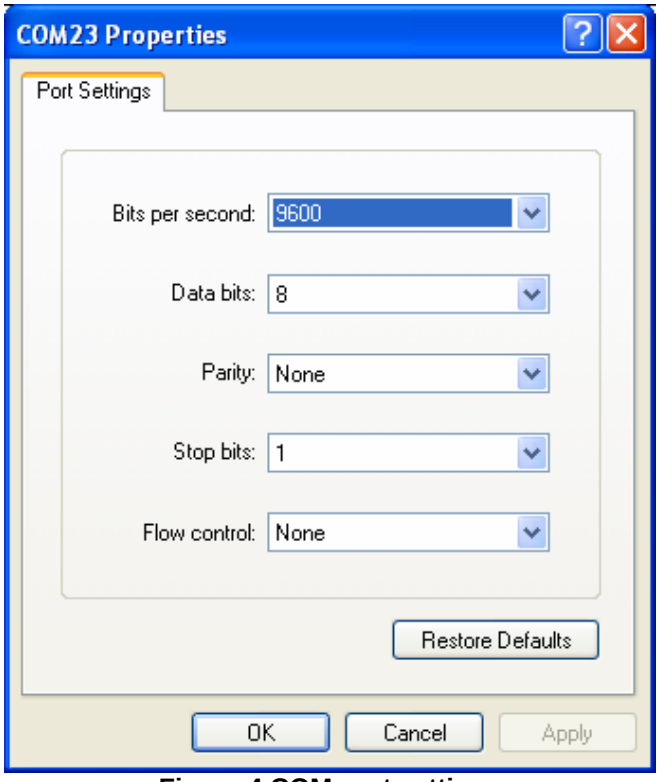


Figure 4 COM port settings

4 Application Examples

The following application examples are included in this software package:

"reset"	**"**System Reset" example. This application writes the CC2480 System Reset Indication to the serial port.
"hello"	**"**Hello world" example. This application writes the CC2480 Chip ID and version number to the serial port.
"RPC"	"RPC" example. This application shows the basic 4 RPC commands available for the MSP430 to communicate with CC2480 by SPI transport
"SAPI"	Simple Application Programmer Interface test application using the configuration SAPI function calls. [3].

Details about how to run the different application examples can be found in the following sections. Section 3.2 describes how to program the applications on the target.

4.1 Reset application

This application example requires 1 node with serial connection to the PC. Please refer to section 3.2 for how to program the application examples. Select the project 'reset' in the IAR workspace viewer.

The "reset" application can be used as a simple test to check that the hardware is working and is properly set up. This application will write the CC2480 System Reset Indication to the serial port.

Follow the steps below to run the application. These steps assume that the board is programmed with the "Reset" application:

1. Start the HyperTerminal application on the PC and configure it according to section 3.2.4.
2. Connect the ez430 –RF2480 to a USB port
3. Board is Power up through the USB port automatically.
4. The LED D1 (RED) is set ON to indicate that the board is powered up. A message displaying the application name and chip type can also be seen on the HyperTerminal.
5. Press Button 1 on the CC2480 TB; The system reset indication* is then displayed on the HyperTerminal window.

**The 'h' seen after the numbers indicates the number is a hexadecimal value.*

4.2 RPC Application

This application example requires 1 node with serial connection to the PC. Please refer to section 3.2 for how to program the application examples. Select the project 'rpc' in the IAR workspace viewer.

This application shows the usage of the four CC2480 command types. The indication and response from CC2480 will be written out on the serial port.

After programming the board with the 'spi' example, follow these steps to run the application:

1. Start the HyperTerminal application on the PC and configure it according to section 3.2.4.
2. Connect the ez430 – RF2480 to a USB port
3. Board is Power up through the USB port automatically.
4. The LED D1 (RED) is set ON to indicate that the board is powered up. A message displaying the application name and chip type can also be seen on the HyperTerminal.
5. Press Button 1 on the CC2480 TB; A Hard reset will be performed and reason will be displayed* on the HyperTerminal window.

6. Press Button 1 again on the CC2480 TB; A Soft reset will be performed and the LED D2 (GREEN) is set on to indicate success.
7. Press Button 1 again on the CC2480 TB; A SYS_VERSION command will be performed, the LED D2 (GREEN) is cleared, and the system reset response* is then displayed on the HyperTerminal window.

**The 'h' seen after the numbers indicates the number is a hexadecimal value.*

4.3 Hello application

This application example requires 1 node with serial connection to the PC. Please refer to section 3.2 for how to program the application examples. Select the project 'hello' in the IAR workspace viewer.

The "hello" application can be used as a simple test to check that the hardware is working and is properly set up. This application will write the CC2480 Product ID and IEEE address to the serial port.

Follow the steps below to run the application. These steps assume that the board is programmed with the "Hello" application:

1. Start the HyperTerminal application on the PC and configure it according to section 3.2.4.
2. Connect the ez430 – RF2480 to a USB port
3. Board is Power up through the USB port automatically
4. The LED D1 (RED) is set ON to indicate that the board is powered up. A message displaying the application name and chip type can also be seen on the HyperTerminal.
5. Press Button 1 on the CC2480 TB; The Product Id* and IEEE address* are then displayed on the HyperTerminal window.

**The 'h' seen after the numbers indicates the number is a hexadecimal value.*

4.4 SAPI application

This application example requires 1 node with serial connection to the PC. Please refer to section 3.2 for how to program the application examples. Select the project 'sapi' in the IAR workspace viewer.

The "SAPI" application can be used as a simple test to check that the configuration interface of CC2480 using the SAPI commands, "zb_WriteConfiguration" and zb_ReadConfiguration. This application will write the CC2480 write the response messages to the serial port.

Follow the steps below to run the application. These steps assume that the board is programmed with the "sampleapi" application:

6. Start the HyperTerminal application on the PC and configure it according to section 3.2.4.
7. Connect the ez430 – RF2480 to a USB port
8. Board is Power up through the USB port automatically
9. The LED D1 (RED) is set ON to indicate that the board is powered up. A message displaying the application name and chip type can also be seen on the HyperTerminal.
10. Press Button 1 on the CC2480 TB; The write configuration response is then displayed on the HyperTerminal window.
11. Press Button 1 again to see the read configuration response
12. Press Button 1 again to see the loopback response of a configuration change.

**The 'h' seen after the numbers indicates the number is a hexadecimal value.*

5 Software Library Reference

This section describes the software libraries the application examples are built upon.

5.1 Software architecture

The design of the software in this package is based on the layered architecture as depicted in Figure below.

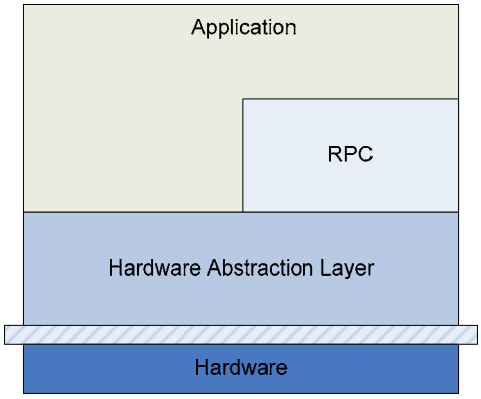


Figure 5 SW architecture

The software implementation consists of the following modules:

- **Application** layer. This software package contains several applications examples with access to Basic RF and HAL.
- **RPC** This layer offers a simple protocol for the RPC transport transmission and reception on the SPI interface link.
- **Hardware Abstraction Layer**. Contains functionality for access to the radio and onboard peripherals modules like LED, UART, Light Sensor, buttons, timers etc.

The most simple application examples are built directly on top of HAL, while the other examples are built on top of the RPC protocol. A detailed description of the RPC protocol is found in the CC2480 documentation

5.1.1 Software folder structure

The software and documentation in this package is organized in the folder structure shown in Figure . The documentation is found in the *docs* folder. Workspace, and project files are found in the folder *iar/ez430_rf2480* folder. Source code for the different applications can be found in the folder *source/Apps*. The *Components* folder includes source code for the different components used by the applications. The HAL source code components are found under the *Components* folder.

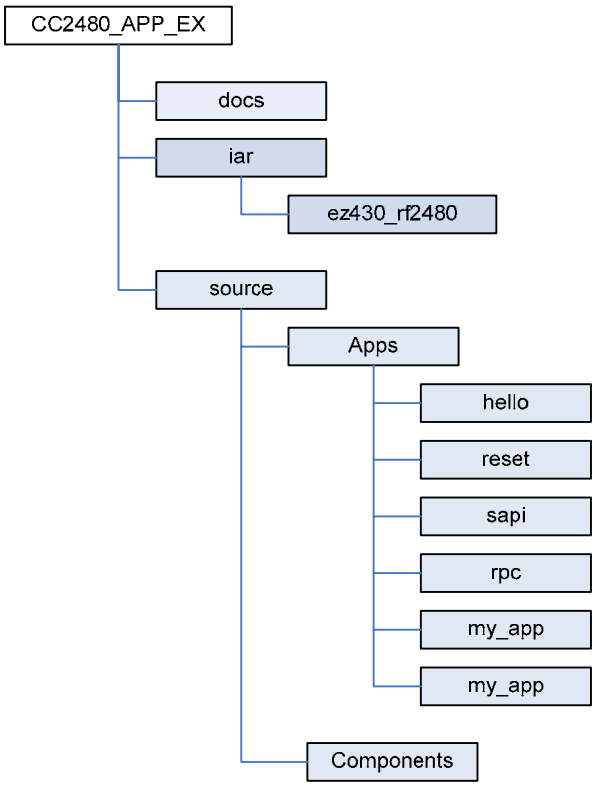


Figure 6 Software folder structure

References

[1] **CC2480 Interface Specification**

[2] **CC2480 datasheet**

[3] **ez430-RF2480 website**

Document History

Revision	Date	Description/Changes
-	2008-04-30	Initial release