Lab 3 - Advanced Embedded Systems - Spring 2015

Wireless communications using the YRDKRX63N and an XBee

Learning Objectives

This lab will allow the students to establish a communication between two Renesas RX63N boards using an XBee communications module. Your goal is to discover the secret "cypher key".

General Information

1. Learn the serial communication concepts of an XBee module.
2. Create a new workspace following all the steps needed.
3. Build the project and load it onto both the RX63N board. Run the program and observe the operation.

Pre-lab Activity

To perform this lab you will need the following materials:

1. One Renesas RX63N Evaluation Boards (YRDKRX63N)
2. PC with the Renesas HEW integrated Development Environment loaded.
3. One XBee module for communications with another XBEE module.

Some other important things to do before the lab:

1. Properly solder your XBee breakout board.
2. Obtain jumper wires with headers to avoid soldering to the XBee breakout board and the Renesas Evaluation Board.
3. ***Be sure that you DO NOT solder anything directly to the XBee***.

Laboratory Assignment Overview

In this lab the students will have to connect their RX63N board to another board using an XBee module. The XBee module is interfaced via USART (SCI for the RX63N) configured with 8 bits of data, 1 stop bit, and no parity at 9600 baud. Each module will be pre-configured to connect to a coordinator set up inside of the EPIC building using a pre-determined PAN ID. The Interfacing of the boards is to be done in such a way that when one character of data is sent from your XBee to the coordinator (16-bit destination address 0), a single "coded character" is returned from the other board. You need to determine the "cipher code" (how the character is modified) (Hint: The cipher code is a single operation with a single constant value).

Requirements:

Req. 1. The code generated must be written in C for the RX63N Evaluation Board.

Req. 2. The code must be well commented and easy to follow.

Req. 3. A single data byte payload should be sent from your board to another board using an XBee module.

Req. 4. For every message sent to the other board, you will receive a single message with a single data byte in return.

Req. 5. You data byte payload should be a value between 0x20 and 0x7E (Character ‘space’ to ‘~’).

Req. 6. Show the entered code on the top row of the LCD. Before any switch is pressed, show the character 'space' (0x20).

Req. 7. All characters from 0x20 (‘space’) to 0x7F (‘~’) are used.

Req. 8. When SW1 of the board is pressed, the displayed character "increments" by one, i.e. 'A' increases to 'B'.

Req. 9. When SW2 of the board is pressed, the displayed character "decrements" by one, i.e. 'Z' increases to 'Y'.

Req. 10. The incrementing of the character will roll over from '~' to 'space'.

Req. 11. The decrementing of the character will roll over from 'space' to '~'.

Req. 12. When SW3 of the board is pressed, the current character should be sent via the XBee module to the other board.

Req. 13. When the character code is sent from a board, it should NOT disappear from the top row of the LCD.

Req. 14. When the returned character code is received from the other board, it should be displayed on the second line of the LCD.

Req. 15. Packets sent through the XBee should be constructed using the “Transmit Request” API Type. (See XBee Series 2 documentation for details on the Transmit Request API type)

To Submit:

- All C files used during the creation of the product. Code must be commented enough for the TA to understand what each line is doing.
- A zip file containing the files specified above.
- Your lab check-off sheet at the demonstration.

Embedded Systems Lab Demonstration Validation Sheet

This sheet should be modified by the student to reflect the current lab assignment being demonstrated

Lab Number:	Lab 3 – Xbee Transmission
Team Members	Team Member 1 : Team Member 2 :
Date:	

Lab Requirements

Obtain a list of the Lab requirements from the end of the lab handout and type them here, perform a self-review and indicate with an X if you met each requirement or not.

REQ Number	Objective	TA Review
1	UART is properly configured for 9600 baud, no parity, 1 stop bit on the RX63N	
2	XBee Transmission packets are properly created and transmitted.	
3	XBee Reception packets are properly received.	
4	The correct “cypher code” is determined.	