

Simple Media Access Controller (SMAC) for the HCS08 Demonstration Applications

User's Guide

Document Number: SMAC08DAUG
Rev. 0.0
05/2011

How to Reach Us:**Home Page:**

www.freescale.com

E-mail:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:

Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-521-6274 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

IAR Systems, IAR Embedded Workbench, C-SPY, visualSTATE, From Idea to Target, IAR KickStart Kit, IAR PowerPac, IAR YellowSuite, and IAR are trademarks or registered trademarks owned by IAR Systems AB.

Copyright © 2007 IAR Systems AB.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2005, 2006, 2007, 2008, 2009, 2010, 2011. All rights reserved.

Contents

About This Book

Audience	iii
Organization	iii
Revision History	iv
Definitions, Acronyms, and Abbreviations	iv
References	v

Chapter 1 Introducing the SMAC

1.1 Supported Hardware	1-1
1.2 SMAC Software	1-1
1.3 Generating an Application in BeeKit	1-2
1.4 Loading Applications Into a Board Using the BDM	1-5
1.5 IDE Considerations	1-7
1.6 UART/USB Virtual Com Port Setup	1-7

Chapter 2 Wireless UART

2.1 Generating the Wireless UART Application	2-1
2.2 Compile and Load the Wireless UART Application	2-1
2.3 Starting the Wireless UART Application	2-1
2.4 Entering My Address ID, PAN ID and Destination Address ID	2-2
2.5 Over The Air Package Structure	2-7
2.6 Hardware Interface	2-7
2.6.1 Hardware Usage (Board by Board)	2-7
2.6.2 Hardware Interface Description	2-8

Chapter 3 Connectivity Test

3.1 Generating the Connectivity Test Application	3-2
3.2 Compile and Load the Connectivity Test Application	3-2
3.3 Starting the Connectivity Test (Serial Interface)	3-2
3.4 Connectivity Test Applications	3-3
3.4.1 Test Mode (Continuous Transmit or Receive)	3-4
3.4.2 Spectrum Analyzer Captures	3-5
3.5 Packet Error Rate (PER) Test	3-7
3.5.1 Starting the PER Test	3-7
3.5.2 PER Operation	3-9
3.6 Range Test	3-9
3.7 Continuous Energy Detect (ED)	3-10
3.8 Continuous Scan	3-10
3.9 Crystal Adjustment	3-11

3.10	Radio Register Edit	3-11
3.11	Manual User Interface Mode	3-14
3.11.1	Manual Interface Test States	3-14

Chapter 4 Generic Demonstration Application

4.1	Generating the Generic Application	4-1
4.2	Compile and Load the Generic Application	4-1
4.3	Starting the Generic Application Demonstration	4-1

Chapter 5 Simple ZigBee Test Client (Simple ZTC)

5.1	Generating the Simple ZTC Application	5-1
5.2	Compile and Load the Simple ZTC Application	5-1
5.3	Starting the Simple ZTC Application	5-1

Chapter 6 Repeater Demonstration

6.1	Generating the Repeater Application	6-1
6.2	Compile and Load the Repeater Application	6-1
6.3	Starting the Repeater Application	6-2
6.3.1	Menu Function Overview	6-2
6.3.2	Changing the Operating Channel	6-3
6.3.3	LED Functionality (Channel Selection)	6-3
6.3.4	Default Values	6-3
6.4	Using the Repeater Application	6-4

Chapter 7 Range Demonstration

7.1	Generating the Range Demonstration Application	7-2
7.2	Compile and Load the Range Demonstration Application	7-2
7.3	Range Demonstration Setup	7-3
7.4	Range Test Calibration	7-4
7.5	Range Demonstration Operation	7-6
7.5.1	PER Test	7-6
7.5.2	LQI Test	7-7
7.5.3	LQI Manual Mode	7-8

Chapter 8 Accelerometer Demonstration

8.1	Generating the Accelerometer Demonstration Application	8-1
8.2	Compile and Load the Accelerometer Demonstration Application	8-1

8.3	Local Node and Remote Node Board Setup	8-1
8.3.1	Board One Setup (Local Node)	8-1
8.3.2	Board Two Setup (Remote Node)	8-2
8.4	Accelerometer Demonstration Application Overview	8-2
8.5	Triax GUI Application Setup	8-3
8.6	Running the Triax GUI	8-4
8.6.1	Calibration Procedure	8-5
8.7	Verifying Operation	8-5

About This Book

This guide provides a detailed description of the Simple Media Access Controller (SMAC) for the HCS08 demonstration applications. The demonstration applications are delivered as ANSI C source code and are included in the SMAC HCS08 Codebase. The Codebase is incorporated into the Freescale BeeKit Wireless Connectivity Toolkit. See the *BeeKit Wireless Connectivity Toolkit User's Guide* (BKWCTKUG) and the BeeKit on-line help for more information.

This guide explains how to run various SMAC for the HCS08 demonstration applications and provides information that allows users to more easily take advantage of these demonstration applications or modify the applications to fit their specific needs.

For more details about the MC1320x, MC1321x and MC1323x devices, see the appropriate Reference Manuals and/or Data Sheets.

For more details about the SMAC for the HCS08, see the *SMAC for the HCS08 Reference Manual* (SMAC08RM)

Audience

This document is intended for application developers generating SMAC for the HCS08 demonstration applications from the Freescale BeeKit Wireless Connectivity Toolkit or modifying these applications as a starting point for proprietary applications.

Organization

This document is organized into the following chapters:

- Chapter 1 **Introducing the SMAC** — This chapter shows how to generate and load SMAC demonstration applications using BeeKit.
- Chapter 2 **Wireless UART** — Describes how the Wireless UART application allows the Freescale IEEE 802.15.4 family of boards to communicate at typematic rates from one board to another over the air.
- Chapter 3 **Connectivity Test** — Provides an easy way to test the RF performance of the transceiver for basic transmitter and receiver tests. It includes test mode (continuous tx, modulated, unmodulated, etc.); Packet Error Rate and Range (LQI measurements).
- Chapter 4 **Low Power Bell Application** — Details how the Low Power receiver application reports a change of state on the LCD on the development board when it receives a new packet from the Low Power transmit application.
- Chapter 5 **Generic Application** — Describes how this application can be used as a template to develop proprietary applications. This application provides all the initialization settings required to start the transceiver and the commonly used peripherals such as the UART and KBIs.

- Chapter 6 **Simple ZigBee Test Client (Simple ZTC)** — Describes how the Simple ZTC allows users to test the SMAC primitives by sending special data frames through the Serial/USB port.
- Chapter 7 **Repeater** — Details how this application repeats all the messages received in a specific channel in order to extend the range of a Point-to-Point link.
- Chapter 8 **Range Demonstration** — Describes how the Range Demonstration application allows users to determine the maximum development board RF range.
- Chapter 9 **Accelerometer Demonstration** — Details how the Accelerometer Demonstration application uses the Freescale X, Y, and Z axes accelerometers.

Revision History

The following table summarizes revisions to this document since the previous release (Rev. 0.0).

Revision History

Location	Revision
Entire document	First public release.

Definitions, Acronyms, and Abbreviations

The following list defines the acronyms and abbreviations used in this document.

AES	Advanced Encryption Standard
API	Application Program Interface
ASM	Advanced Security Module
BDM	Background Debug Module
BDM debugger	A debugger using the BDM interface for communication with the MCU. An example is the P&E BDM Multilink debugger for HCS08.
CBC	Cipher Block Chaining.
CBC-MAC	Cipher Block Chaining Message Authentication Code.
CCM	Counter with CBC-MAC
CTR	Counter.
dBm	Decibels referred to one milliwatt.
ED	Energy Detect.
EN	End Node - Evaluation Board.
EVB	Evaluation Boards.
EVK	Evaluation Kit
EWV	Embedded Workbench IDE work space file
GUI	Graphical User Interface
IDE	Integrated Development Environment

LP	Low Power
MAC	Media Access Control
MAC	Message Authentication Code
MCU	MicroController Unit
NCB	Network Coordinator Board
NVM	None-Volatile Memory
OTA	Over-the-air
OTAP	Over-the-air Programming
PC	Personal Computer
PCB	Printed Circuit Board
PiP	Platform in Package
RCM	1323x Remote Control Module
REB	1323x Remote Extender Board
RX	Reception
S19	S - Record. 'S19' is the file extension used for the Freescale binary image format. The S19 file encapsulates the binary image as a list of ASCII records. Each record contains a length -, address -, data - and checksum field. The 16 bit address field allows a memory space for up to 64 KB. The S19 can be generated with CodeWarrior IDE and is the product from the linking process. S19 does not contain additional information to a debugger (where to look for source files).
Safe Mode Boot	The Embedded Bootloader boots up using safe default system values.
SMAC	Simple Media Access Control.
SN	Sensor Node Evaluation Board.
STDB	Socket Test Development Board.
TX	Transmission.

References

The following sources were referenced to produce this book:

[1] Freescale 802.15.4 MAC/PHY Software Reference Manual (802154MPSRM)

Chapter 1

Introducing the SMAC

The Freescale Simple Media Access Control (SMAC) is incorporated into the Freescale BeeKit Wireless Connectivity Toolkit. BeeKit generates the appropriate project files and project folder structure. Everything in a project is in source code, and BeeKit generates a single target per project. Every BeeKit generated project runs transparently with CodeWarrior 10.1 Special Edition.

To use any of the existing applications available in SMAC, users must first generate the applications as projects in a BeeKit solution.

For more information about BeeKit, BeeKit Projects and Solutions, refer to the *BeeKit Wireless Connectivity Toolkit User's Guide* (BKWCTKUG) and the BeeKit on-line help.

1.1 Supported Hardware

This version of SMAC contains targets for the following Freescale hardware:

- 1320x-QE128
- 1321x-SRB
- 1321x-NCB
- 1323x-RCM
- 1323x-REM

The example provided in this guide uses the Remote Control Motherboard (RCM) as the primary hardware target. The Remote Control Motherboard (RCM) has a Modular Reference Board (MRB) plugged into it. The Remote Extender Module (REM) has a Modular Reference Board (MRB) plugged into it. For more information about these boards see the *1323x Development Hardware Reference Manual* (1323xDHRM).

1.2 SMAC Software

Download the latest version of the SMAC which is part of BeeKit, from Freescale. The SMAC Codebase release notes provide a list of the currently available SMAC demonstration applications.

CAUTION

Ensure that CodeWarrior and its plug-ins are completely up to date. Before proceeding, follow the CodeWarrior update process as described in the Freescale *BeeKit Wireless Connectivity ToolKit Quick Start Guide* (BKWCTKUG).

1.3 Generating an Application in BeeKit

This section shows how to generate an SMAC demonstration application using BeeKit. Because all of the SMAC demonstration applications are generated from BeeKit, this section is used as a reference where needed for several tasks required by some but not all of the SMAC demonstration applications. The exceptions are noted as required.

1. Launch the BeeKit GUI and the BeeKit main window appears as shown in [Figure 1-1](#).

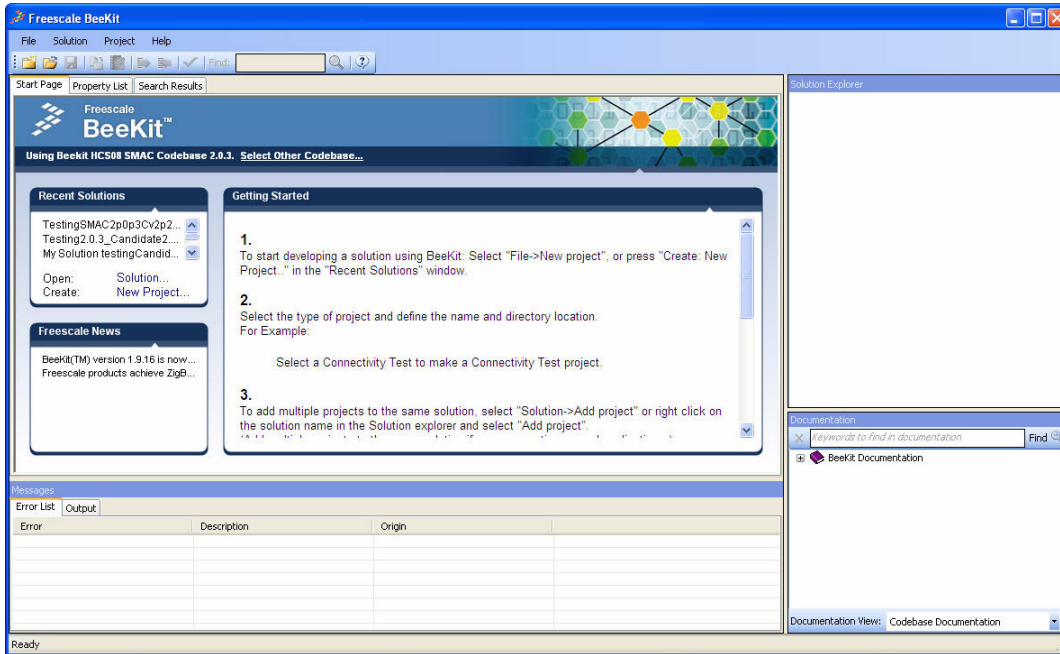


Figure 1-1. BeeKit Graphical User Interface

2. Click on "Select other codebase" to change codebase as shown in [Figure 1-2](#)

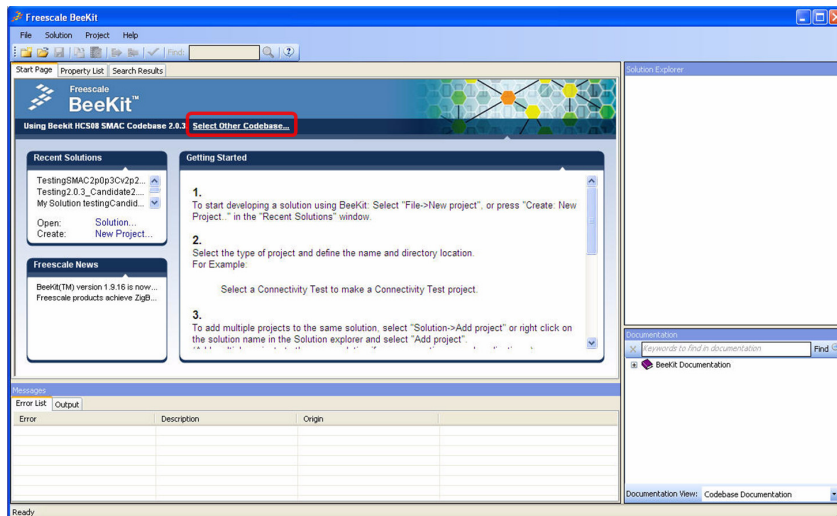


Figure 1-2. Select Other Codebase

The Set Active Codebase window appears as shown in [Figure 1-3](#).

3. Select the appropriate SMAC Codebase as shown in [Figure 1-3](#).

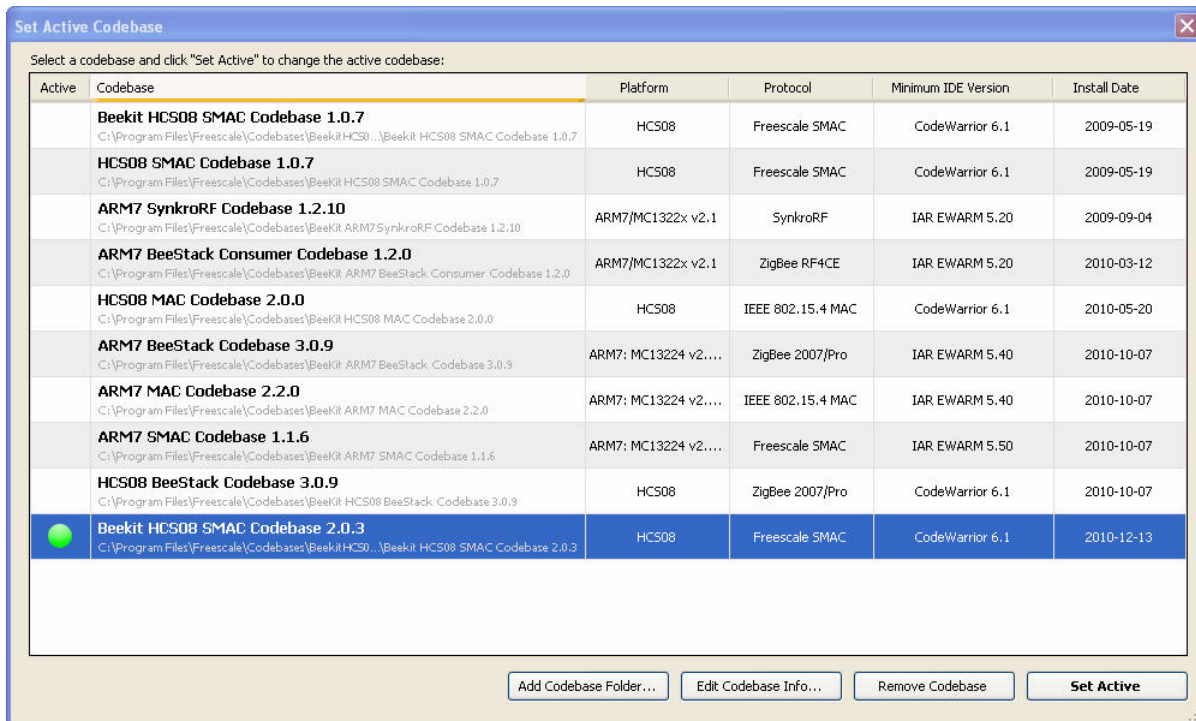


Figure 1-3. selecting SMAC Codebase

In BeeKit, a Codebase is a set of files and rules that permit users to generate usable applications.

4. Create a new project by selecting File->New Project in BeeKit. Then select the project template to use from those available as shown in [Figure 1-4](#).

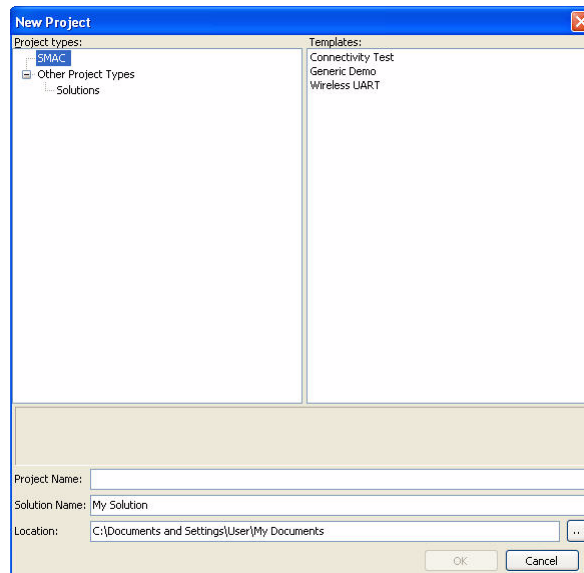


Figure 1-4. Creating a new project

If users create a new project and there is not an existing solution, then users, by creating the project, are also creating a solution. Users must specify a name for the solution.

5. If the demonstration application requires it, add other projects to the current solution as shown in [Figure 1-5](#).

If the demonstration application requires different evaluation boards, create a new project with the proper configuration for each evaluation board.

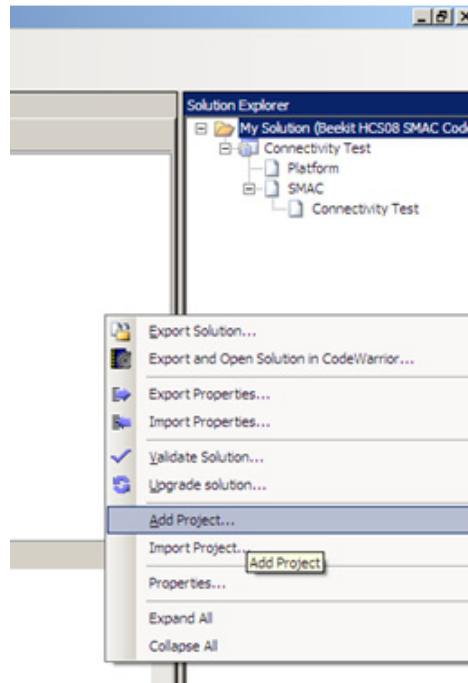


Figure 1-5. Adding a New Project to an Existing Solution

6. Modify the project properties in BeeKit as shown in [Figure 1-6](#). Use the BeeKit online help for assistance.

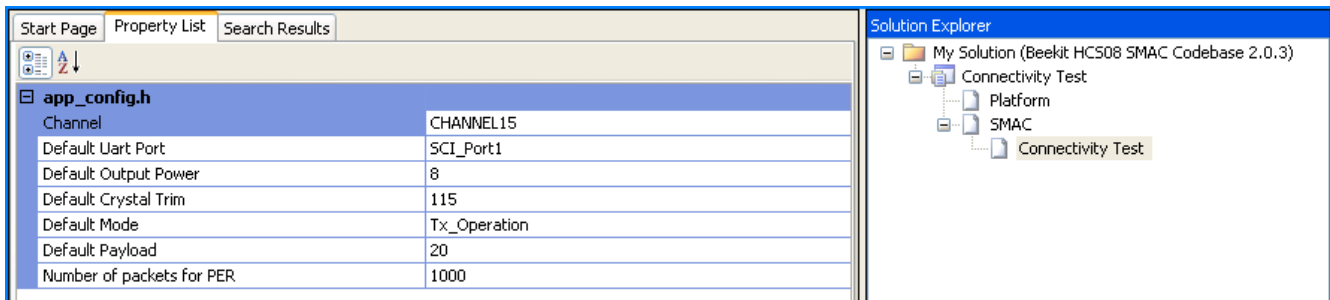


Figure 1-6. Software Component Properties in BeeKit

7. Validate the solution as shown in [Figure 1-7](#).

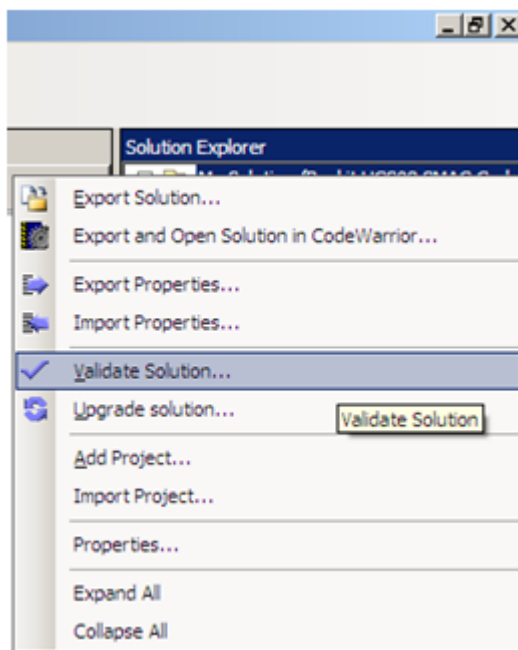


Figure 1-7. Validating a Solution

If the system detects any errors, a brief description of the error appears in the Messages section located in the lower left corner of the BeeKit Main window, under the Errors tab as shown in [Figure 1-1](#).

8. After validating the solution, users must import the solution into CodeWarrior as described in the Freescale *BeeKit Wireless Connectivity ToolKit Quick Start Guide* (BKWCTKUG) then return to [Section 1.4, “Loading Applications Into a Board Using the BDM”](#) to build and load the application to a board.

1.4 Loading Applications Into a Board Using the BDM

After the solution is imported into CodeWarrior as described in Freescale *BeeKit Wireless Connectivity ToolKit Quick Start Guide* (BKWCTKUG), users can now build and load the application to a board. This example uses the Connectivity Test, but also works with all currently available SMAC applications.

1. From the CodeWarrior main window, select the previously created project and click on the “Build” button as shown in [Figure 1-8](#).

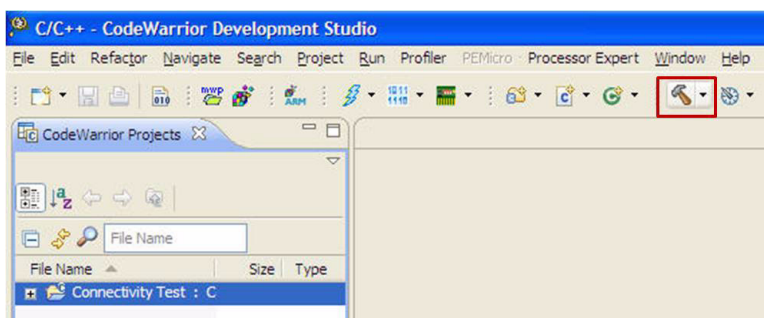


Figure 1-8. Build Button

If required, the IDE builds the target and provides information messages, reports errors, and displays warnings as needed.

2. If no errors are encountered from the build process, connect the BDM to the target. The blue LED on the P&E Multilink turns on if the connection to the PC is functioning. The yellow LED on the P&E Multilink turns on if the cable is properly connected to the target.
3. Click the Debug/Run button as shown in the top left corner of [Figure 1-9](#).

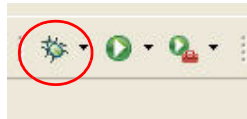


Figure 1-9. Debug Button

If the connection to the USB BDM debugger is in place and the polarity to the BDM interface is correct, the debugger tries to download the binary to the target.

After the FLASH memory is programmed, the Default/Main debugger window appears as shown in [Figure 1-10](#).

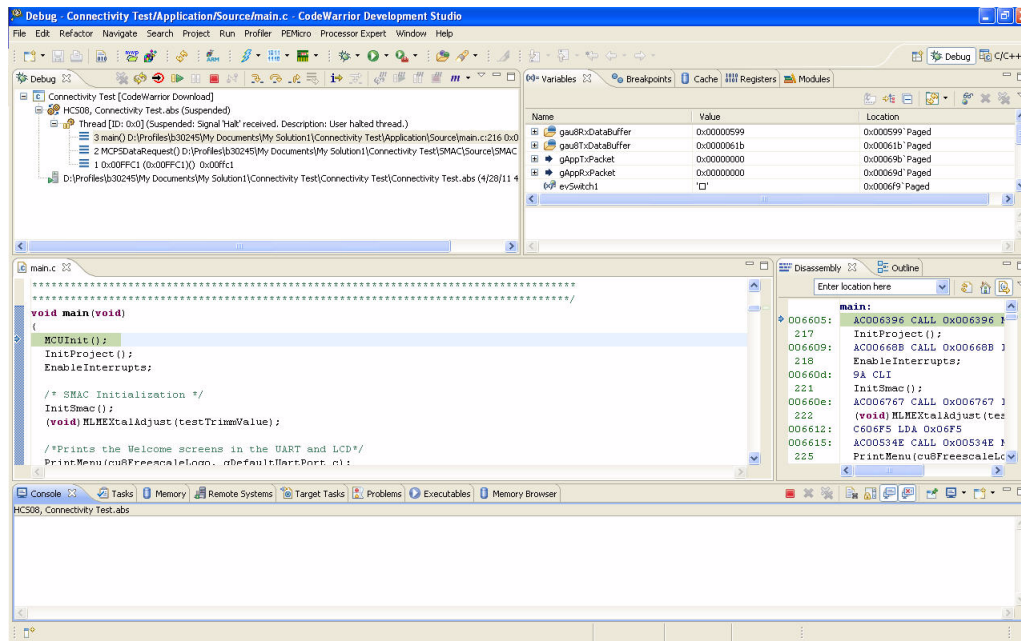


Figure 1-10. Debugger Window

4. To run the program, users have two options:
 - a) Disconnect the BDM cable from the board and push the Reset button on the board once.
 - b) Leave the BDM cable connected and click on the Run/Resume button as shown in [Figure 1-11](#).



Figure 1-11. Run/Resume button.

The loading process described in this section applies to all currently available SMAC applications.

1.5 IDE Considerations

The SMAC does not support the device initialization feature of CodeWarrior.

1.6 UART/USB Virtual Com Port Setup

This section describes how to set up the UART/USB virtual Com port for the Connectivity, Wireless UART and the rest of the demonstration applications. [Figure 1-12](#) shows the default SMAC RS-232 settings.

1. In the PC terminal program, set the correct baud rate, data bits, parity, COM port, and flow control as shown in [Figure 1-12](#).

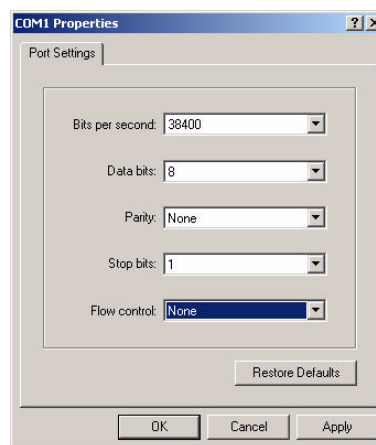


Figure 1-12. Default SMAC RS-232 Setting

Users are now ready to reset the boards and start the demonstration applications.

2. In the PC terminal program, set the properties in the optional settings as shown in [Figure 1-13](#).

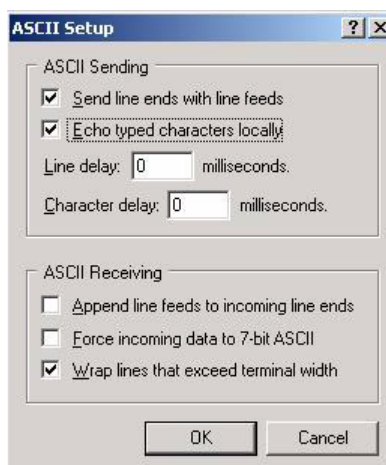


Figure 1-13. Additional Terminal Program Settings

3. Connect the boards to a USB cable, the USB driver is required and do not use a USB hub. The driver is located in the following directory:

```
<INSTALL DIR>\Freescale\Drivers
```

The BeeKit CD image is available from the Freescale ZigBee web site at www.freescale.com/802154.

Chapter 2

Wireless UART

The Wireless UART demonstration application uses the Freescale IEEE 802.15.4 family of boards to communicate at typematic rates over the air from one board to another. This allows users to implement a UART-to-Wireless communication bridge.

The following is a simplified example and the code as it exists cannot be used as a cable replacement. If cable replacement is the goal, then queues, buffers, and other constructs must be added to increase the reliability and efficiency of this demonstration.

The Wireless UART application now includes PAN ID and Address ID filtering. Only boards with the same PAN ID will be able to communicate between themselves. Users can choose the Destination Address ID for each message. The application implements acknowledge messages over the air (OTA). However, this feature is optional and can be disabled at compile time.

2.1 Generating the Wireless UART Application

The Wireless UART demonstration requires at least two boards to perform Over-the-Air (OTA) communication. At least two projects are needed to correctly run this demonstration.

Generate the application with BeeKit and download to each board using the steps as shown in [Chapter 1](#), “[Introducing the SMAC](#)”.

2.2 Compile and Load the Wireless UART Application

To compile and load a demonstration application into the evaluation boards, see [Section 1.4](#), “[Loading Applications Into a Board Using the BDM](#)”.

2.3 Starting the Wireless UART Application

1. Configure the Com Port as shown in [Section 1.6](#), “[UART/USB Virtual Com Port Setup](#)”.
2. Connect each board to a USB cable and a USB port (one PC for each board) and ensure they are enabled as shown in [Section 1.6](#), “[UART/USB Virtual Com Port Setup](#)”.
3. Press the Reset button once on both boards. The terminal displays text as shown in [Figure 2-1](#).

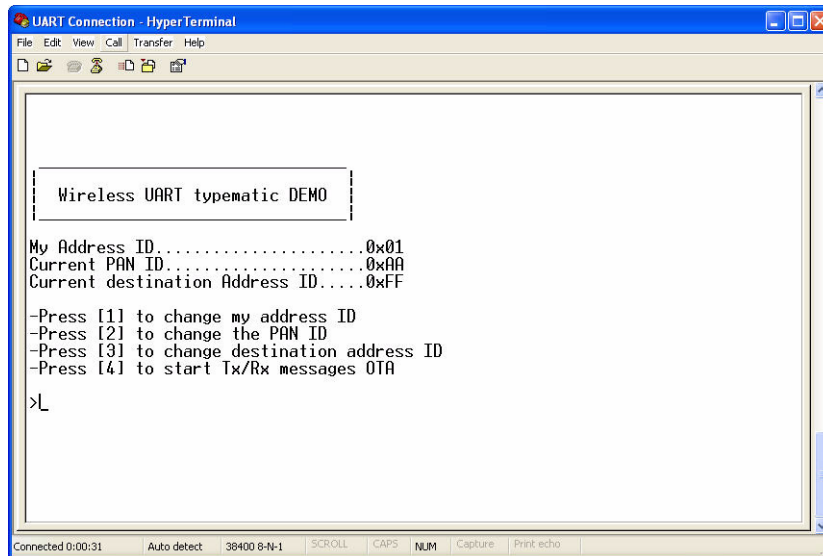


Figure 2-1. Wireless UART Reset UART Start Screen

2.4 Entering My Address ID, PAN ID and Destination Address ID

The My Address ID field is defined at compile time and filters the incoming messages. If the incoming message is a broadcast message (destination address ID = 0xFF) or if it has its destination address equal to My Address ID, the message is processed as a received message by the application. Otherwise, the message is ignored.

The My Address ID, PAN ID and Destination Address ID may be changed by following the directions from the main menu. The default values for these three fields are selected at compile time.

1. To change the My Address ID value, press "1" from the main menu screen and the My Address ID string appears as shown in [Figure 2-2](#).

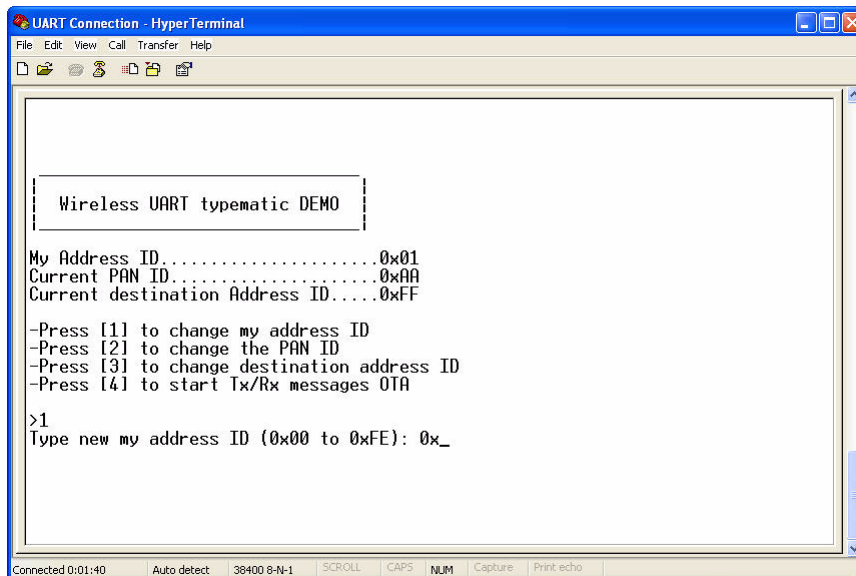


Figure 2-2. My Address ID String

- To change the PAN ID value, press "2" from the main menu screen and the PAN ID string appears as shown in Figure 2-3.

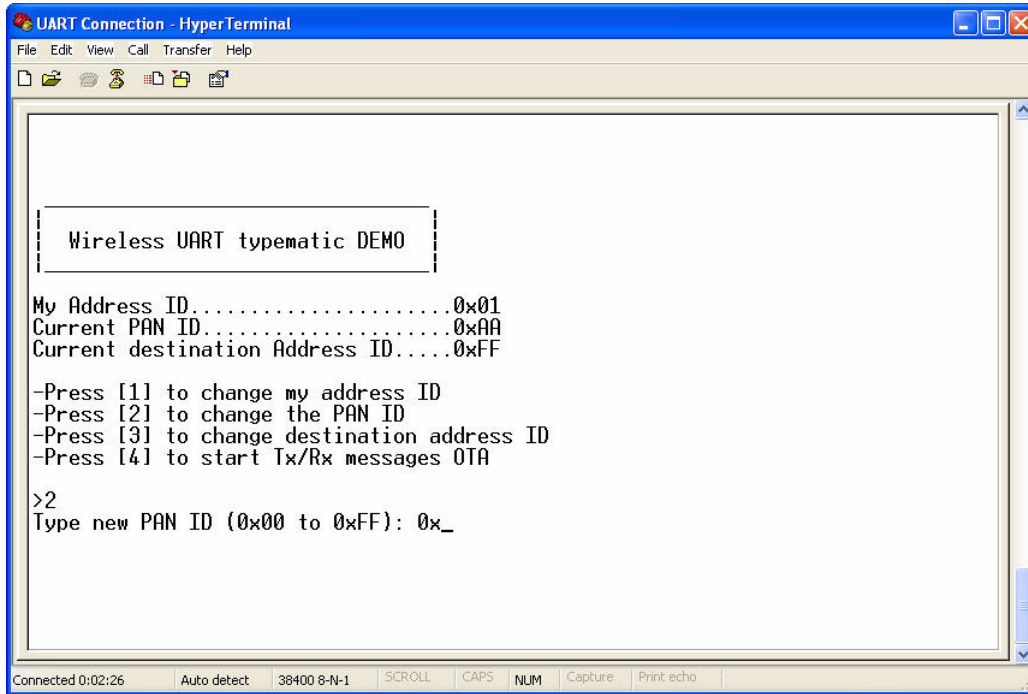


Figure 2-3. PAN ID String

- To change the Destination Address ID, press "3" from the main menu and the Address ID string appears as shown in Figure 2-4.

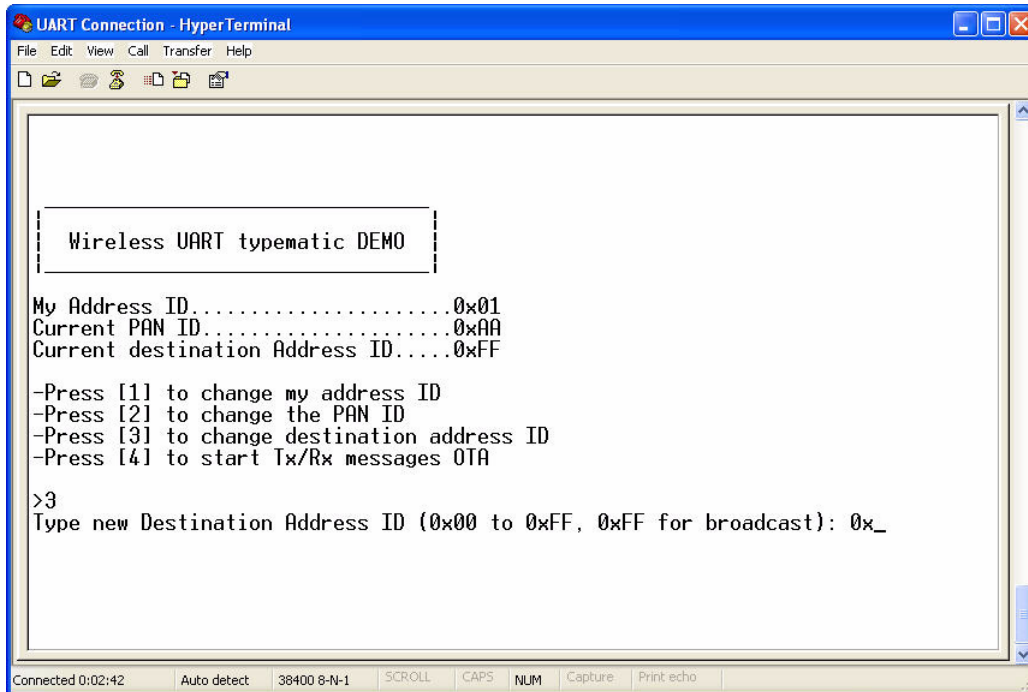


Figure 2-4. Destination Address ID String

The Hyper Terminal screen is not cleared when these modes are requested. The screen only clears when the main menu is refreshed.

Users can choose values from 00 to FF in these modes. If an invalid character is entered, the Invalid command/value string appear as shown in [Figure 2-5](#).

The My Address ID, PAN ID or Destination Address ID screen is refreshed below the Invalid value string, so that users can enter another My Address ID, PAN ID or Destination Address ID.

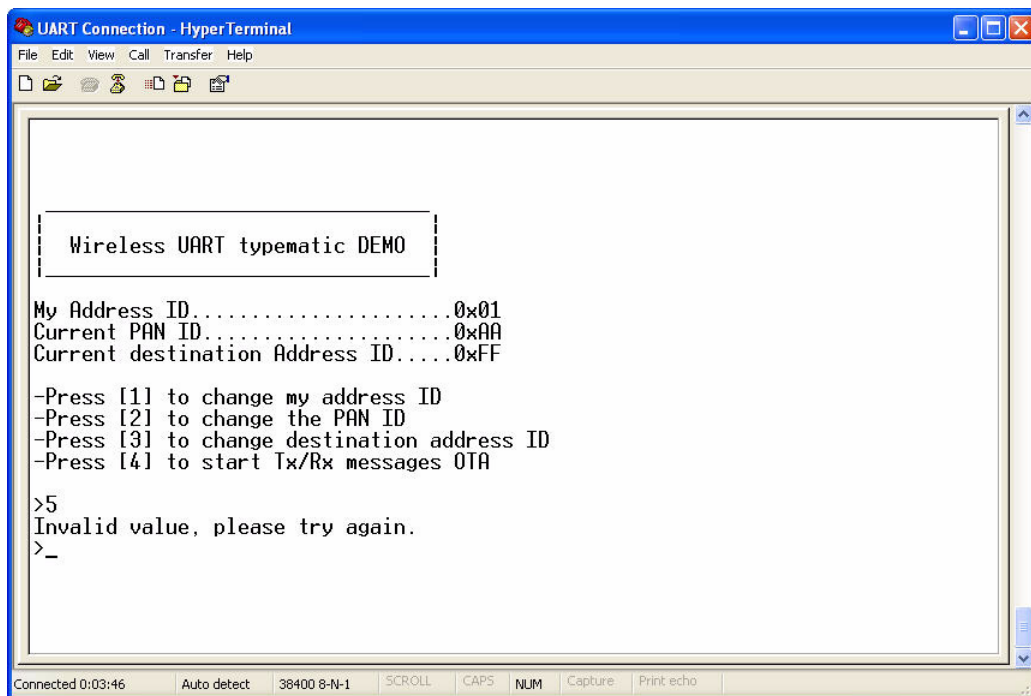


Figure 2-5. Wireless UART Invalid Command/Value String

NOTE

Only when a valid value (00 to FF) is entered for either the My Address ID, PAN ID or Destination Address ID values will users be able to return to the main menu.

4. After a valid value is entered, and users are ready to start sending or receiving messages over the air, press option "4" from the main menu. The screen is cleared and the "typing a message" screen appears as shown in [Figure 2-6](#). This window allows users to send the data that was typed into one Hyper Terminal windows and see all the messages received displayed in the other Hyper Terminal window.

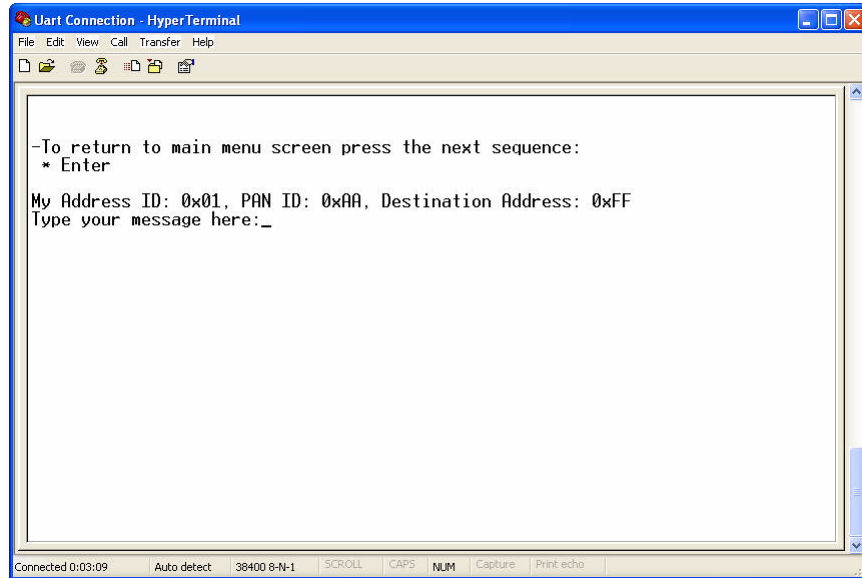


Figure 2-6. Wireless UART Typed Message Screen

5. To return to the main menu to change the My Address ID, PAN ID or Destination Address ID, enter the following character sequence:
* Enter
6. Once both boards are reset, connectivity is established and a valid My Address ID, PAN ID or Destination ID have been entered, type some characters in the HyperTerminal Session (Board 1) window ([Figure 2-7](#)) and the typed message appears on the other PC's HyperTerminal Session (Board 2) window ([Figure 2-8](#)).

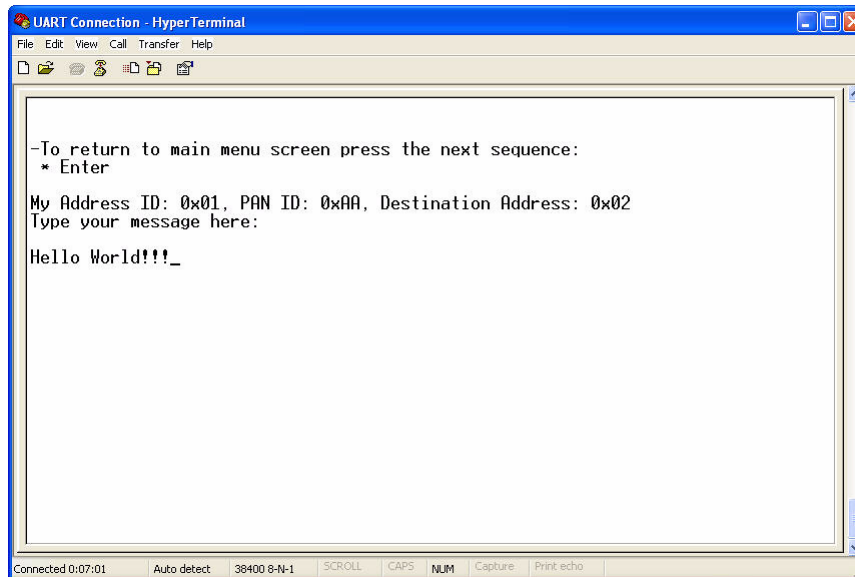


Figure 2-7. Wireless UART Board 1 (Output to Board 2)

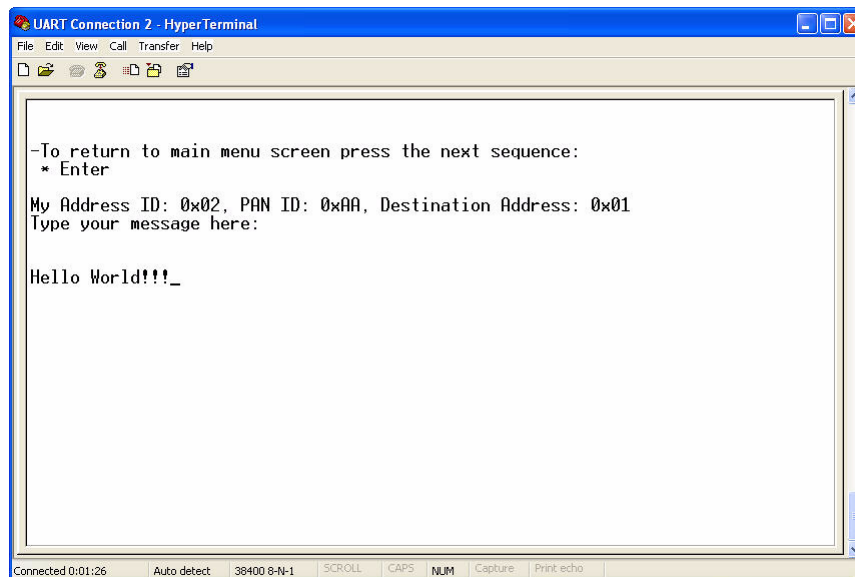


Figure 2-8. Wireless UART Board 2 (Input from Board 1)

Because the Wireless UART demonstration is a two-way communication protocol, the boards retry their packets up to three times if an acknowledgement is not received. As already stated, this application highlights a very basic SMAC wireless UART implementation and because it is a basic demonstration application, it is not intended for large file transfers.

NOTE

Users can disable the configuration mode so that the application defaults directly to the typing message window (Figure 2-6) with the default values for My Address ID, PAN ID and Destination Address ID already loaded. This option is included for users who do not need to change the ID values.

2.5 Over The Air Package Structure

This section describes the Over the Air data frame format that the Wireless UART application implements for the PAN ID, Address ID and the My Address ID.

Table 2-1. Over the Air Data Frame Format

Field	PAN ID	Destination Address ID	My Address ID	Message Payload
Length	1 byte	1 byte	1 byte	Up to 120 bytes
Value	0x00 – 0xFF	0x00 – 0xFF	0x00 – 0xFF	
Description	Personal Area Network ID.	Address ID of the node that must receive the package (0xFF for broadcast)	Address ID of the local node that will transmit the package.	

2.6 Hardware Interface

The Wireless UART software uses the built-in hardware on the target boards (LEDs, Switches, Touch pads, and Keyboard) to perform Wireless UART application functionality.

2.6.1 Hardware Usage (Board by Board)

- MC1323x-RCM
 - Four LEDs
 - Touch pad
 - Two keys from keyboard
- MC1323x-REM
 - Four LEDs
 - Two keys from keyboard
- MC1321x-NCB
 - Four LEDs
 - Two switches
- MC1321x-SRB
 - Four LEDs
 - Two switches
- 1320x-QE128-EVB
 - Four LEDs
 - Two switches

2.6.2 Hardware Interface Description

The Hardware Interface allows users to navigate Wireless UART application functionality using the switches, LEDs, buttons, touch pad, etc. of the board under test.

2.6.2.1 Switch Functionality

The switches or keys from keyboard, increase or decrease the channel number used for receiving and transmitting packages over the air.

- SW1 increases the channel number
- SW2 decreases the channel number

2.6.2.2 Touch Pad

The touch pad interface is only available on the 1323x RCM. The touch pad increases and decreases the channel number. Users flick the touch pad interface to change the channels as shown in [Figure 2-9](#).

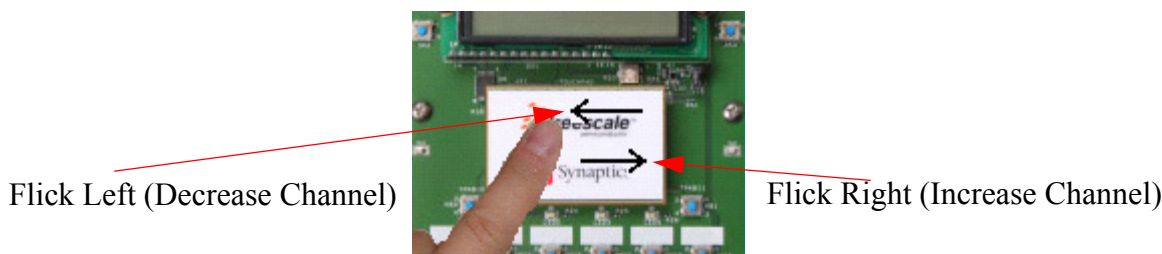


Figure 2-9. Wireless UART Touch Pad Operation

2.6.2.3 LED Functionality (Channel Selection)

- LED1 turns ON momentarily every time a package is transmitted or received.
- By pressing SW1 or SW2, the LEDs display the current channel number (using binary) as shown in [Figure 2-10](#). The LED display is persistent for few seconds to allow users to see which channel is currently selected. Channels range from 11 to 26 and only a partial list is shown in [Figure 2-10](#).

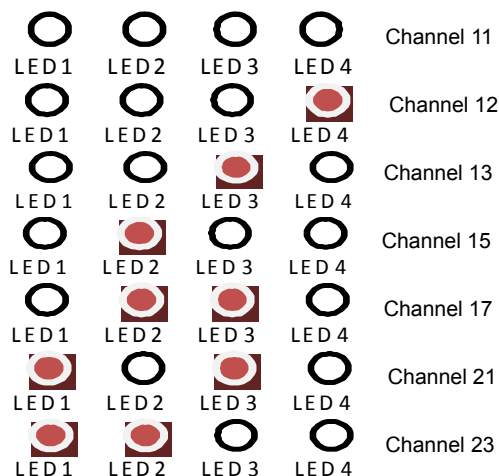


Figure 2-10. Binary Channel LED Display (partial)

Chapter 3

Connectivity Test

The connectivity test evaluates the basic connectivity between two transceivers. Both the transmitter and receiver are supported under the same Connectivity Test application. The Connectivity Test has two interfaces, a serial interface that works through a PC terminal program and a manual user interface that uses the buttons, switches, etc., on the boards to perform the functions. The Connectivity Test supports the following functionality:

- Works for receive or transmit modes
 - Configure the transceiver in a specific test mode to test transceiver RF performance. The transmit tests are as follows:
 - IDLE
 - Pseudorandom Binary Sequence (PRBS) Pulse transmit
 - Modulated transmit
 - Unmodulated transmit
 - Packet Error Rate (PER) transmit
 - Range Test transmit
- The receive tests are as follows:
- IDLE
 - Continuous receive
 - Packet Error Rate (PER) receive
 - Range Test receive
 - Energy Detect scan
- Set the 802.15.4 operating channel
 - Set output power
 - Adjust crystal trim values
 - Adjust Range and PER tests packet length
 - Radio Register Edit (Allows users to modify direct and indirect radio and MCU access registers during run time.)

Two interfaces are implemented to use this application: a manual interface that uses the push buttons on the boards and a terminal based serial interface that uses menus to navigate application functionality.

Only one interface can be selected at a time. Both interfaces can not been active at the same time. The interface type is selected when starting the application. Pressing “enter” in the Hyperterminal enables the serial interface. Pressing SW1 on the board under test enables the manual interface.

3.1 Generating the Connectivity Test Application

The Connectivity Test demonstration requires at least two boards to perform Over-the-Air (OTA) communication. At least two projects are needed to correctly run this demonstration:

- Connectivity Test for RCM
- Connectivity Test for REM

Generate the application with BeeKit and download to each board using the steps as shown in [Chapter 1, “Introducing the SMAC”](#).

3.2 Compile and Load the Connectivity Test Application

To compile and load a demonstration application into the evaluation boards, see [Section 1.4, “Loading Applications Into a Board Using the BDM”](#).

3.3 Starting the Connectivity Test (Serial Interface)

To start the connectivity test (serial interface), perform the following tasks:

1. Configure the Com Port as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
2. Connect each board to a USB cable and a USB port (one PC for each board) and ensure they are enabled as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
3. Press the Reset button once on both boards. The terminal displays text as shown in [Figure 3-1](#).

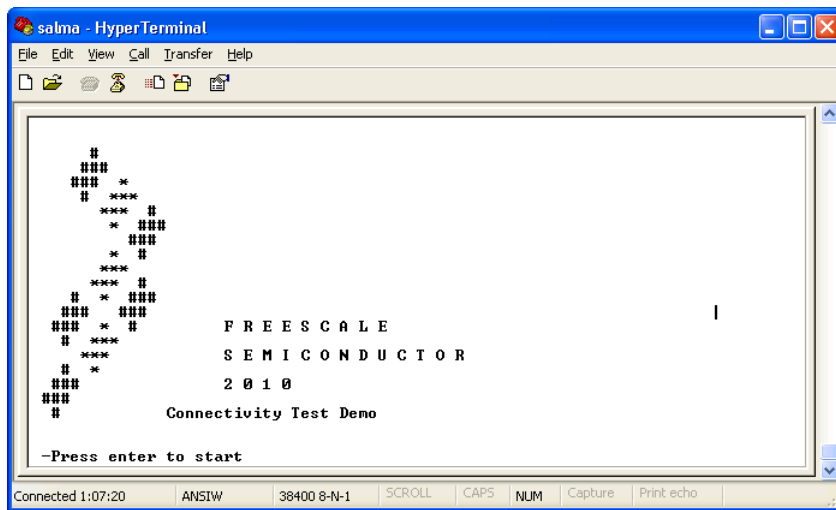


Figure 3-1. Starting the Connectivity Test

1. Press “Enter” to start the application.

The terminal appears as shown in [Figure 3-2](#). The first list details key shortcuts available at any screen, then the main menu appears. The last line shows the current configuration.

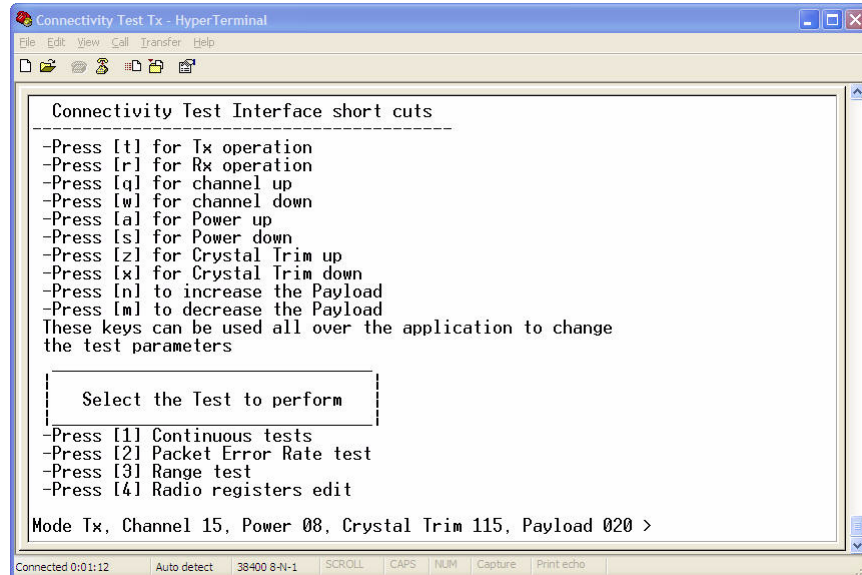


Figure 3-2. Connectivity Test Main Menu

2. Use the shortcut keys as shown in [Figure 3-2](#) to configure the role (TX or RX), channel, power level, crystal trim and payload length as needed.

NOTE

One board must be used as the transmitter and the other board used as the receiver. This is required for the PER and Range Tests, but not for the continuous mode tests.

3. Select the type of test to perform by pressing “1”, “2”, “3” or “4”.

3.4 Connectivity Test Applications

The Connectivity Test application combines the following applications:

- Test Mode (Continuous transmit or receive)
- Packet Error Rate (PER)
- Range testing
- Radio Register Edit

The following sections describe each of these tests.

3.4.1 Test Mode (Continuous Transmit or Receive)

Test Mode allows users to test the RF performance of the transceiver for basic transmitter and receiver functionality. The Test Mode application is a collection of modes consisting of the following tests:

- Idle
- Transmission 9th Order Binary Polynomial (PRBS9)
- Modulated Transmit
- Unmodulated Transmit
- Receive
- Energy Detect
- Scan

3.4.1.1 Selecting the Continuous Transmit/Receive Mode

To select a test mode, press the appropriate key as shown in Figure 3-3. Figure 3-3 shows options for both transmit and receive modes.

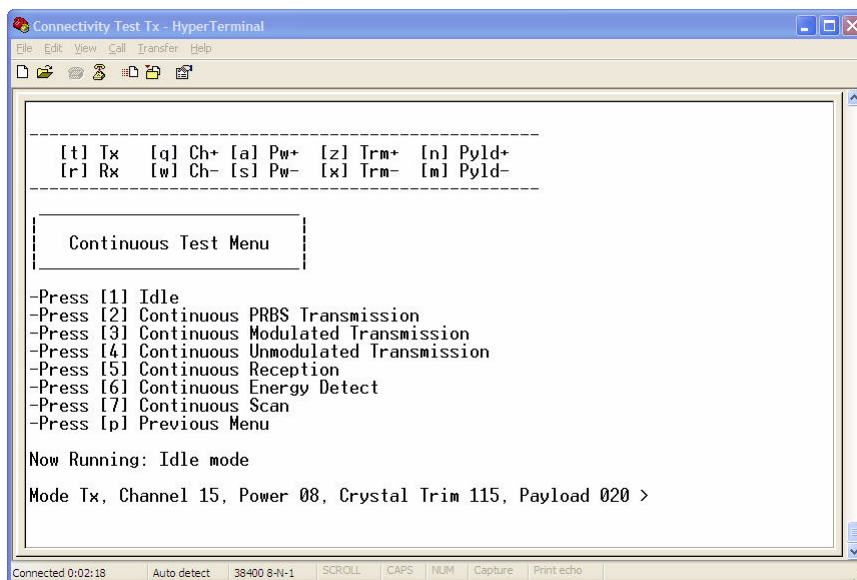


Figure 3-3. Continuous Transmit/Receive Test Mode Menu

3.4.2 Spectrum Analyzer Captures

The measurements shown in this section are example illustrations of the Test Modes and were captured from a MC1323-REM using the F-antenna. These measurements are radiated measurement and not conducted measurements.

3.4.2.1 Idle Mode

The Idle Mode captures the noise floor as shown in Figure 3-4.

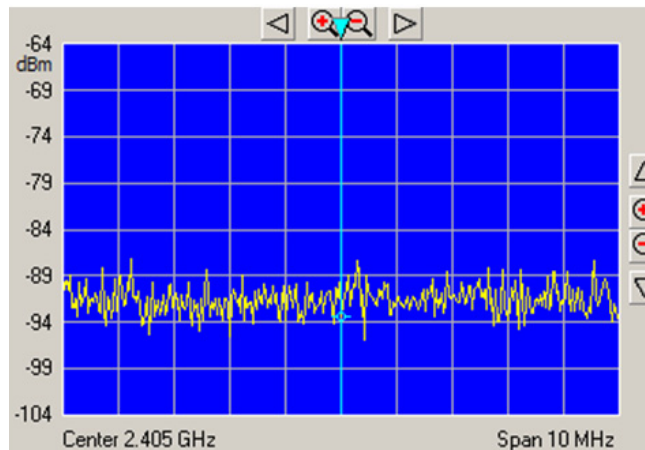


Figure 3-4. Idle Mode

3.4.2.2 PRBS9Transmit Mode

The PRBS9 Transmit Mode captures the 9th Order Binary Polynomial as shown in Figure 3-5.

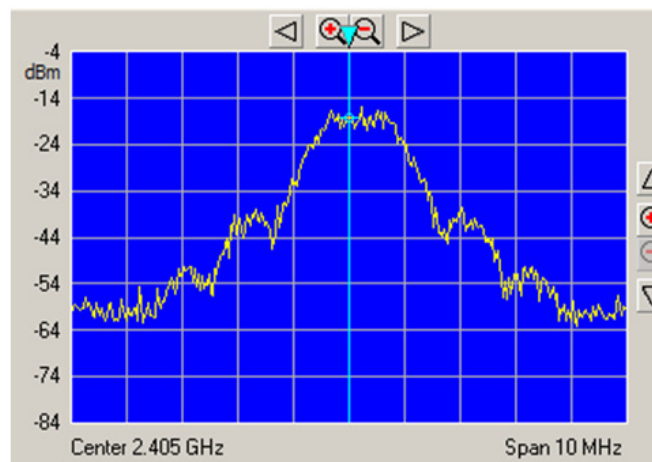


Figure 3-5. PRBS9 TX Mode

3.4.2.3 Receive Mode

Figure 3-6 shows the Receive Mode. Because the radio is in receive mode, there is little or no energy captured in the selected channel band.

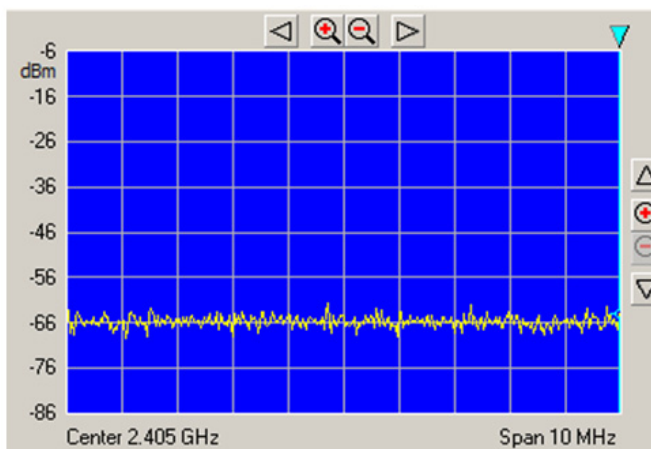


Figure 3-6. Receive Mode

3.4.2.4 Transmit with Modulation

Figure 3-7 shows transmit mode with modulation.

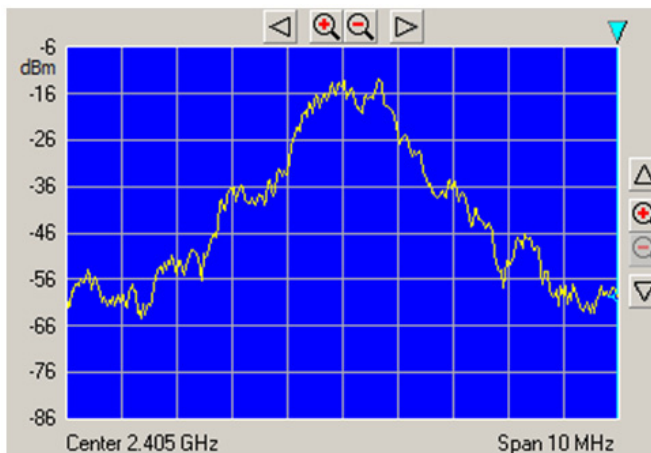


Figure 3-7. Transmit With Modulation

3.4.2.5 Transmit without Modulation (CW Mode)

Figure 3-8 shows Continuous Wave (CW) Mode.

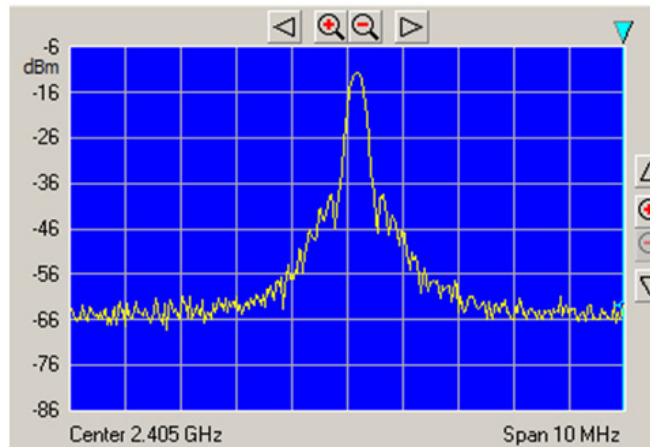


Figure 3-8. Transmit Without Modulation (CW Mode)

3.5 Packet Error Rate (PER) Test

The Packet Error Rate (PER) test is a unidirectional test that sends a number of packets from a transmitter to a receiver all on a single channel. The packet length is configurable (default length is 20 bytes of data with 2 bytes of SMAC). The results of the packets received are displayed at the end of the transmit cycle on the receive side.

3.5.1 Starting the PER Test

Ensure that both the Connectivity Test boards are on the same channel, one configured to receive and the other configured to transmit. The PER-TX board and the PER-RX board are asynchronous to each other which means that the PER-TX board does not know when the PER-RX board is on. Users must place the PER-RX board into Listen Mode prior to setting the PER-TX board into Transmit Packet Mode. If this is not done correctly, the reported packet error rate may be incorrect.

Select the Packet Error Rate option on both the Connectivity Test TX and Connectivity Test RX boards, and follow the instructions as shown in the menus in [Figure 3-9](#) (PER Transmit) and [Figure 3-10](#) (PER Receive).

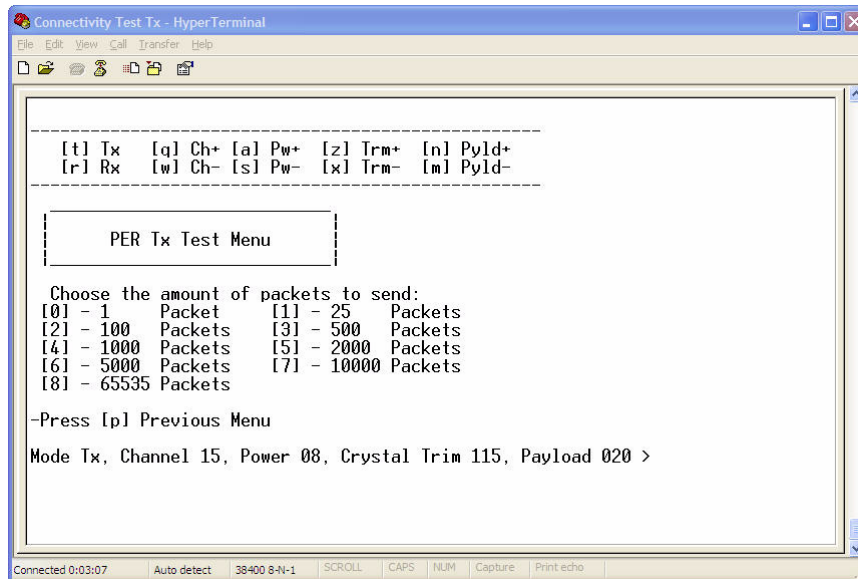


Figure 3-9. Packet Error Rate Test (PER Transmit)

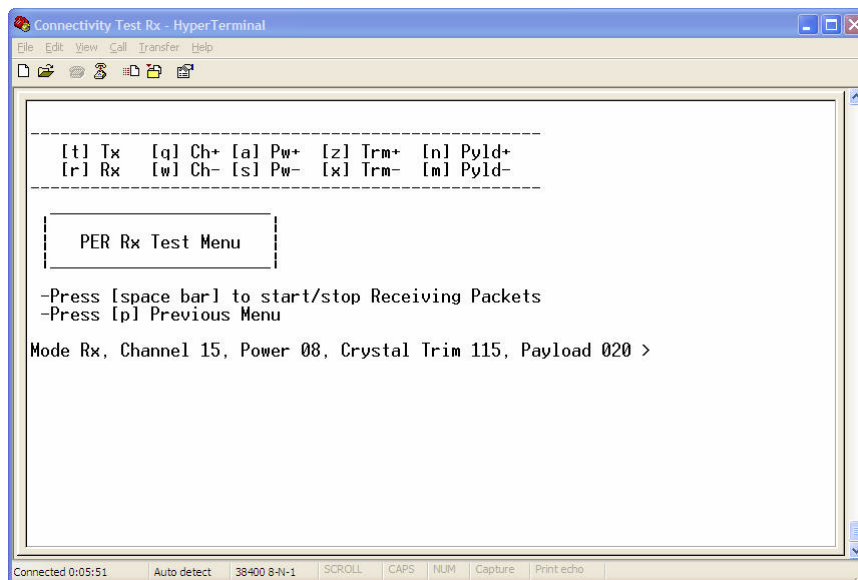


Figure 3-10. Packet Error Rate (PER Receive)

Set the the appropriate role (Transmitter or Receiver) using the shortcuts and select “2” on the main menu to enter the PER TX and PER RX applications.

To start the test press the space bar to start the PER RX application. On the PER TX application, configure the payload size by using “n” to increase and “m” to decrease the payload. The default payload value is 20. Select option 0 to 8 from the menu to define the total number of packets to transmit.

3.5.2 PER Operation

If a packet is received, the application displays “Packet nnn” through the UART connection where ‘nnn’ is the amount of received packets.

These values are shown in [Figure 3-11](#). When the test ends, the total packets received versus the total packets sent is also displayed on the PER RX application.

```

Connectivity Test - HyperTerminal
File Edit View Call Transfer Help
Packet 410
Packet 411
Packet 412
Packet 413
Packet 414
Packet 415
Packet 416
Packet 417
Packet 418
Packet 419
Packet 420
Packet 421
Packet 422
Packet 423
Packet 424
Packet 425
Packet 426
Packet 427
Packet 428
Packet 429
Packet 430
Packet 431
Packet 432
-
Connected 0:03:30 Auto detect 38400 8-N-1 SCROLL CAPS NUM Capture Print echo
    
```

Figure 3-11. Output During PER Test

3.6 Range Test

The Range Test uses the Link Quality Indicator (LQI) to monitor the variation of distance between the transmitter and the receiver during a normal transmission. This test uses two devices, one configured as the transmitter and the other as the receiver.

During the Range Test, the transmitter sends a message and the receiver calculates the LQI of the received packet as shown in [Figure 3-12](#). This process is repeated until the user stops the test.

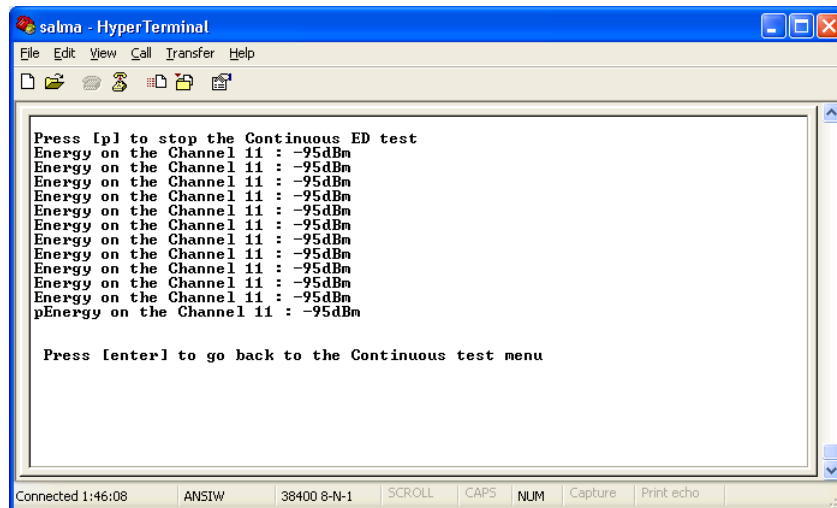
```

salmita - HyperTerminal
File Edit View Call Transfer Help
Range Test Rx Running
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
LQI = -27
Range Test Rx Stopped
Average LQI -27
Press [enter] to go back to the Range Rx test menu_
Connected 0:09:36 ANSIW 38400 8-N-1 SCROLL CAPS NUM Capture Print echo
    
```

Figure 3-12. Range Test

3.7 Continuous Energy Detect (ED)

The Connectivity Test supports an ED operation which monitors and reports the RF energy level detected on one selected channel. The ED displays this value in dBms as shown on [Figure 3-13](#).



```

salma - HyperTerminal
File Edit View Call Transfer Help

Press [p] to stop the Continuous ED test
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
Energy on the Channel 11 : -95dBm
pEnergy on the Channel 11 : -95dBm

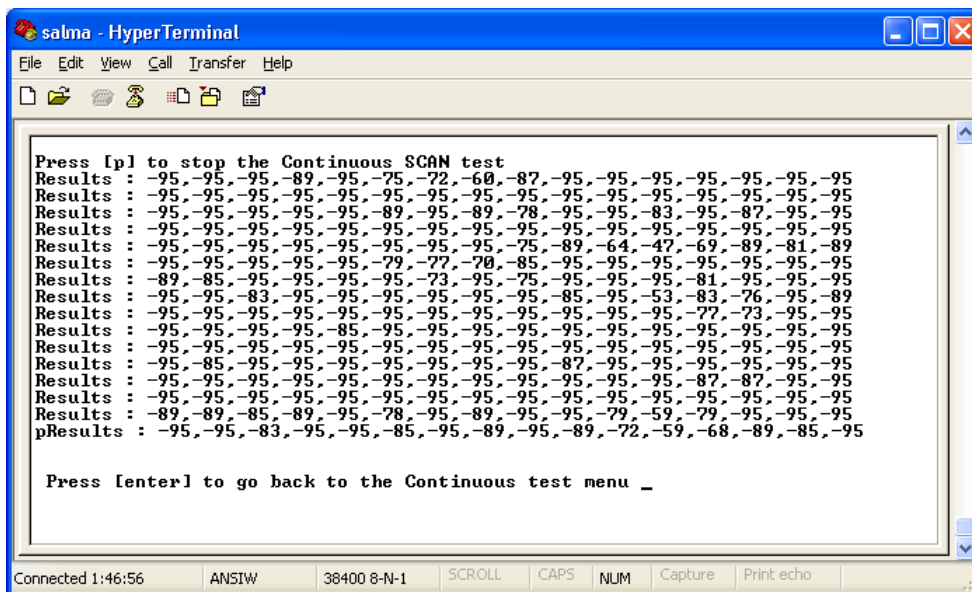
Press [enter] to go back to the Continuous test menu

Connected 1:46:08 ANSIW 38400 8-N-1 SCROLL CAPS NUM Capture Print echo
    
```

Figure 3-13. Energy Detect Report

3.8 Continuous Scan

The Connectivity Test supports an Continuous scan operation which monitors and reports the RF energy level detected on each of the 16 channels. The Continuous scan displays this value in dBms as shown on [Figure 3-14](#).



```

salma - HyperTerminal
File Edit View Call Transfer Help

Press [p] to stop the Continuous SCAN test
Results : -95,-95,-95,-89,-95,-75,-72,-60,-87,-95,-95,-95,-95,-95,-95,-95
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95
Results : -95,-95,-95,-95,-95,-89,-95,-89,-78,-95,-95,-83,-95,-87,-95,-95
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95
Results : -95,-95,-95,-95,-95,-79,-77,-70,-85,-95,-95,-95,-95,-95,-95,-95
Results : -89,-85,-95,-95,-95,-95,-73,-95,-75,-95,-95,-95,-81,-95,-95,-95
Results : -95,-95,-83,-95,-95,-95,-95,-95,-95,-85,-95,-53,-83,-76,-95,-89
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-77,-73,-95,-95
Results : -95,-95,-95,-95,-85,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95
Results : -95,-85,-95,-95,-95,-95,-95,-95,-95,-87,-95,-95,-95,-95,-95,-95
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-87,-87,-95,-95
Results : -95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95,-95
Results : -89,-89,-85,-89,-95,-78,-95,-89,-95,-95,-79,-59,-79,-95,-95,-95
pResults : -95,-95,-83,-95,-95,-85,-95,-89,-95,-89,-72,-59,-68,-89,-85,-95

Press [enter] to go back to the Continuous test menu _

Connected 1:46:56 ANSIW 38400 8-N-1 SCROLL CAPS NUM Capture Print echo
    
```

Figure 3-14. Continuous Scan Report

3.9 Crystal Adjustment

The Connectivity Test application includes the functionality to adjust the on board trim capacitance for the MC1323x, 32 MHz crystal reference oscillator. This crystal trim value can be increased or decreased using the “z” and “x” shortcuts. The trim capacitance can be changed to center the oscillator frequency to the target 32 MHz which minimizes the error frequency (in ppm). For more information on tuning the oscillator, see the *MC1323x Reference Manual*.

The reference frequency can be directly monitored using either of the following GPIO pins:

- 32M_OUT pin
- 32K_OUT pin.

3.10 Radio Register Edit

The Radio Register Edit application allows users to modify radio and MCU direct and indirect access registers during run time. When users select option four from the Connectivity Test Main Window (Figure 3-2), the following screen appears.

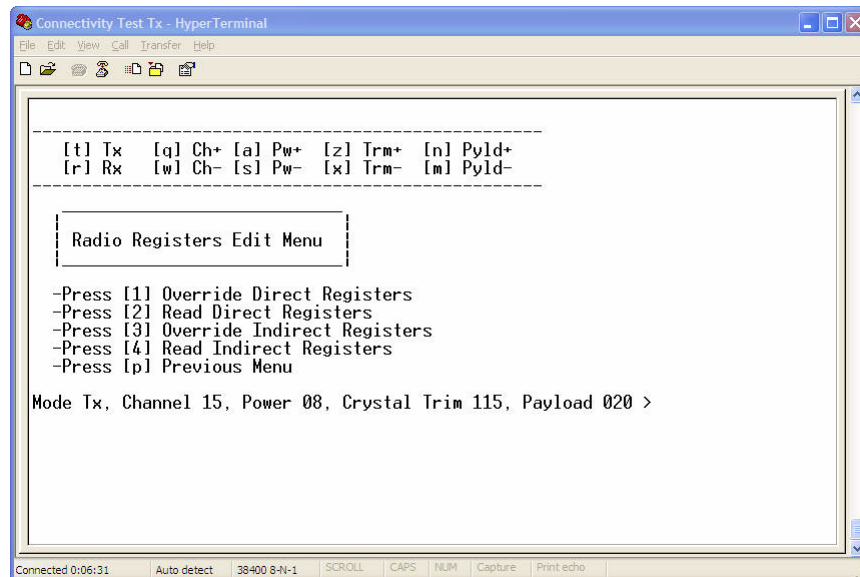


Figure 3-15. Radio Register Edit Main

To perform the register read and write functions, follow the on screen menus as they appear in the Edit Registers windows. Select option [1] and the following screen appears.

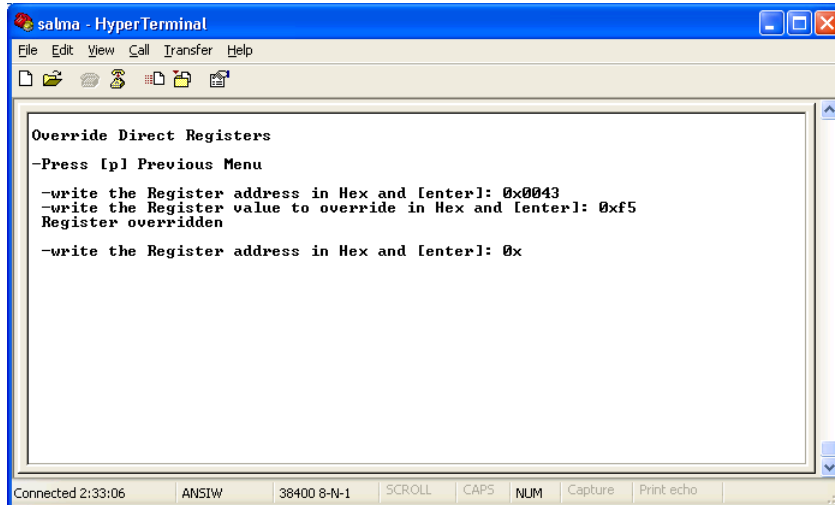


Figure 3-16. Radio Registers Edit Override Direct Registers

Select option [2] and the following screen appears.

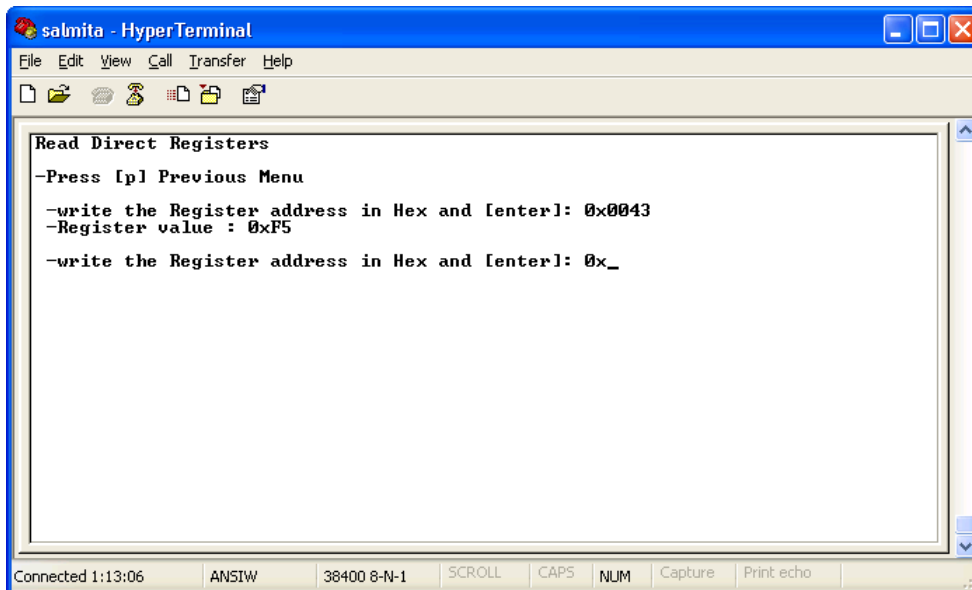


Figure 3-17. Radio Registers Edit Read Direct Registers

Select option [3] and the following screen appears.

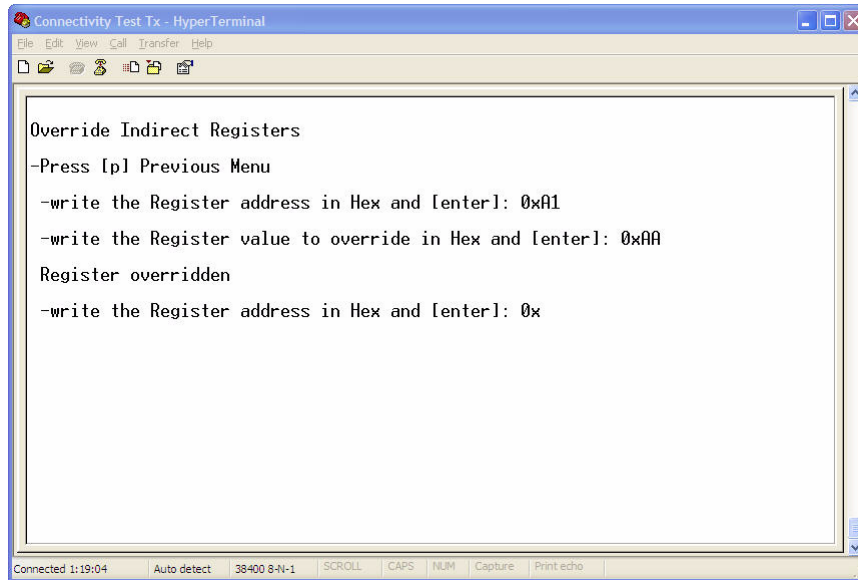


Figure 3-18. Radio Registers Edit Override Indirect Registers

Select option [4] and the following screen appears.

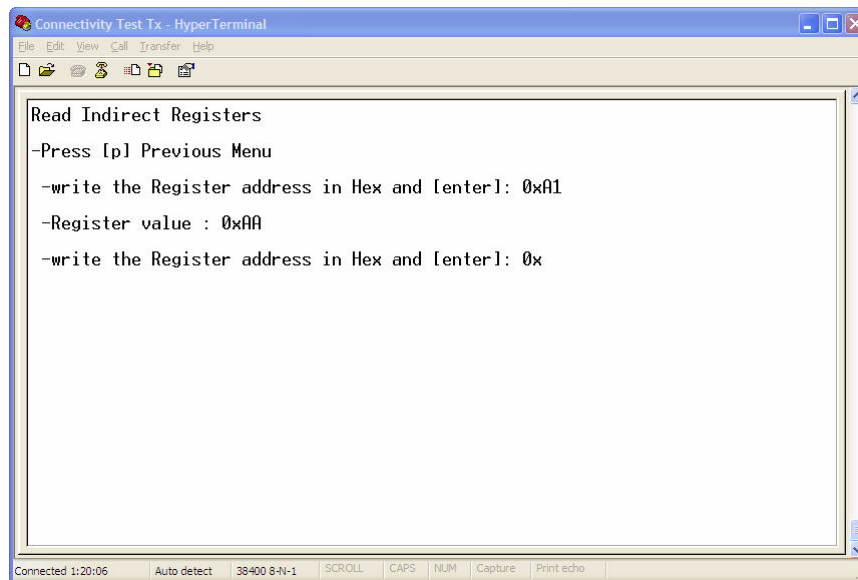


Figure 3-19. Radio Registers Edit Read Indirect Registers

NOTE

When using the write register function, be aware that some registers are read-only registers.

3.11 Manual User Interface Mode

After the boards are programmed and connected, select the manual user interface by pressing SW1 on the target board. The software uses the built in hardware in the target board (LEDs, LCDs, Switches, Touch pads, Buzzers and Keyboard) to provide feedback to users about application functionality.

3.11.1 Manual Interface Test States

The Manual User Interface allows users to navigate Connectivity Test functionality using the switches, buttons, touch pads, etc. of the board under test. [Figure 3-20](#) shows the primary test state flow.

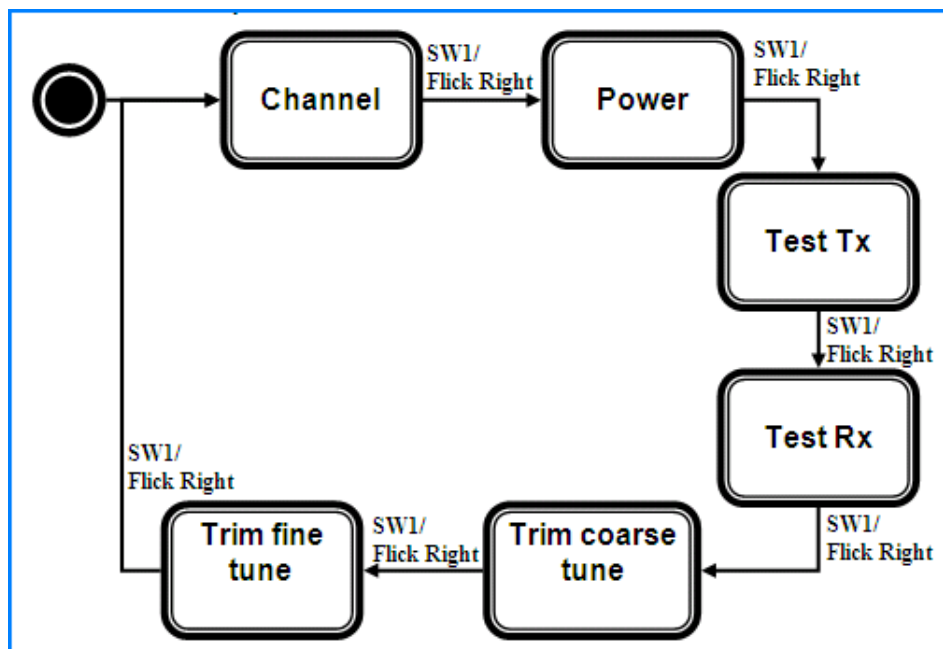


Figure 3-20. Manual User interface Primary Test State Flow

Using SW1 on the boards or the touchpad on the 1323x-RCM, users can step through the main test states of the Connectivity Test in the following order:

1. Change Channel (See [Section 3.11.1.6, “Change Channels”](#) for channel values.)
2. Change Power (See [Section 3.11.1.7, “Change Power”](#) for power values.)
3. Transmit Tests (See [Section 3.11.1.8, “Transmit Tests”](#) for transmit test options.)
4. Receive Tests (See [Section 3.11.1.9, “Receive Tests”](#) for receive test options.)
5. Trim Coarse Tune (See [Section 3.11.1.10, “Trim Coarse Tune”](#) for coarse tune values.)
6. Trim Fine Tune (See [Section 3.11.1.11, “Trim Fine Tune”](#) for fine tune values.)

NOTE

Users cannot step backwards through these options. They can only be cycled through in order. Current status is displayed using either the LCD ([Section 3.11.1.5, “LCD display”](#)) or the LEDs ([Section 3.11.1.3, “LED Functionality”](#)).

3.11.1.1 Switch Functionality

The buttons perform the functions as follows:

- SW1 Selects the primary test state
- SW2 increases a value within that test state
- SW3 decreases a value within that test state
- SW4 starts the selected test

3.11.1.2 Touch pad Functionality

The MC1323x-RCM board has a built-in touch pad. The touch pad performs functions using board gestures as follows:

- Flick right selects an application mode (forward)
- Flick up increases a value
- Flick down decreases a value
- A single tap starts a test

3.11.1.3 LED Functionality

Four LEDs on the boards indicate the states and/or results of Connectivity Test functions. When the application starts, all the LEDs toggle on and off to indicate the user that the application is ready to run.

Once SW1 is pressed, the LEDs represent the selected demonstration mode as follows:

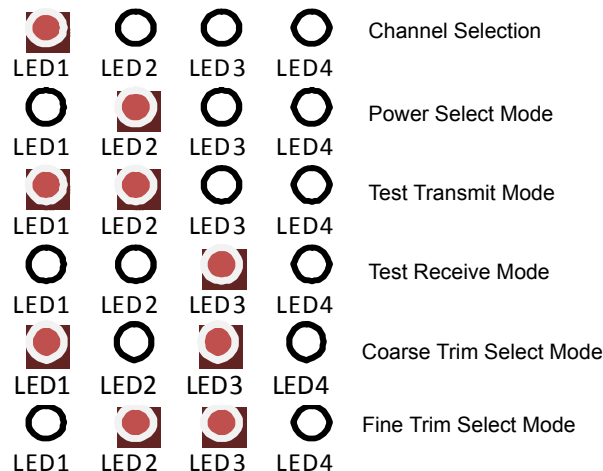


Figure 3-21. Connectivity Test LED Display

When SW2/SW3 are pressed to select a different value, the new value is shown by the LEDs in sequence, using binary code as the switches are pushed. These values are as follows:

1. First switch push, all the LEDs flash for a half second On and a half second Off.
2. Next switch push, the new value is displayed for two seconds.
3. Next switch push, all the LEDs again flash for a half second On and a half second Off.
4. The last switch push, the LEDs return to the demonstration mode.

When running tests, the LEDs have different meanings as follows:

- For the Continuous Transmit Test, LED4 toggles to show that a Continuous Transmit Test is running.
- For the Continuous Receive Test, LED4 toggles only when a packet is received.
- For the Continuous Energy Detect Test, all the LEDs show the Proportional Energy in the channel in binary.
- For the Continuous Scan Test, the LEDs show the index of the best channel in binary.
- At the end of the PER test, the LEDs show the proportion of received packets. (All LEDs lit, all packets received.)
- For the Range Test, the LEDs show the proportion of Link Quality. (All LEDs lit, excellent link Quality.)

3.11.1.4 Buzzer

If the board in use has a buzzer, the buzzer will sound to indicate the finish of PER test in manual mode.

3.11.1.5 LCD display

When the Connectivity Test Application starts, the LCD displays a start screen as shown in [Figure 3-22](#).



Figure 3-22. Connectivity Test, Start

Once SW1 is pressed the LCD displays the selected mode and the mode state or value as shown in [Figure 3-23](#).

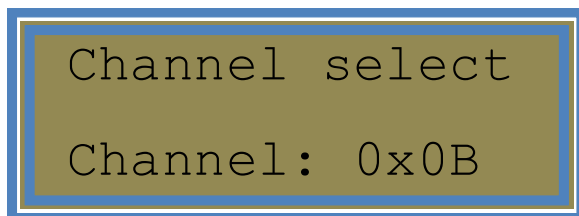


Figure 3-23. Connectivity Test, Mode and Value Selection

When running the PER test, the LCD displays the Cumulative number of packets received, Cumulative number of packets NOT received and Average LQI for all received packet. As shown in [Figure 3-24](#), these values are:

- 111 (average LQI)
- 22222 (cumulative number of packets received)
- 33333 (cumulative number of packets NOT received)



Figure 3-24. Connectivity Test, PER Running

When running the Range Test, the LCD displays the instant LQI and the Average LQI for all received packet as shown in [Figure 3-25](#).

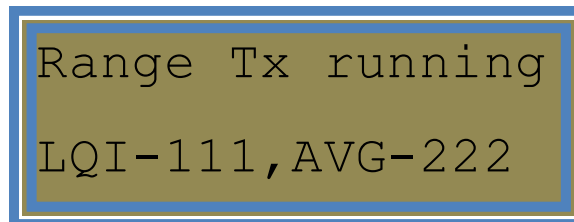


Figure 3-25. Connectivity Test, Range Test Running

3.11.1.6 Change Channels

After selecting the Change Channel state using SW1 or the touchpad, move through the channels by pressing SW2 to select the next highest channel and SW3 to select the next lowest channel. The default channel is 15 and the channel values range from 11 to 26.

3.11.1.7 Change Power

After selecting the Change Power state using SW1 or the touchpad, move through the power settings by pressing SW2 to select the next highest power value and SW3 to select the next lowest power values. The default power value is 8 and the power value ranges from 0 to 14.

3.11.1.8 Transmit Tests

After selecting the Transmit Test State using SW1 or the touchpad, move through the transmit tests by pressing SW2 to move forward and SW3 to move backwards. The Transmit Tests are as follows:

- PER Tx
- Range Tx
- Continuous Idle
- Continuous PRBS9 transmission
- Continuous modulated transmission
- Continuous unmodulated transmission

To start a test, press SW4 and wait for the results.

3.11.1.9 Receive Tests

After selecting the Receive Test State using SW1 or the touchpad, move through the receive tests by pressing SW2 to move forward and SW3 to move backwards. The Receive Tests are as follows:

- PER Rx
- Range Rx
- Continuous Reception
- Continuous Scan
- Continuous Energy Detect

To start a test, press SW4 and wait for the results.

3.11.1.10 Trim Coarse Tune

After selecting the Trim Coarse Tune state using SW1 or the touchpad, move through the coarse trim settings by pressing SW2 to select the next highest trim value and SW3 to select the next lowest trim value. The default trim value is 126 and the trim value ranges from 0 to 255. The Trim Coarse Tune state steps through the values 16 steps at a time.

3.11.1.11 Trim Fine Tune

After selecting the Trim Fine Tune state using SW1 or the touchpad, move through the fine trim settings by pressing SW2 to select the next highest trim value and SW3 to select the next lowest trim value. The default trim value is 126 and the trim value ranges from 0 to 255. The Trim Fine Tune state steps through the values 1 step at a time.

Chapter 4

Generic Demonstration Application

The SMAC codebase includes a Generic Application Demonstration that can be used as a template to develop proprietary applications. This application shows all the initialization required to start the radio and the commonly used peripherals such as UART, LEDs and KBIs.

4.1 Generating the Generic Application

Generate the application with BeeKit and download to each board using the steps as shown in [Chapter 1, “Introducing the SMAC”](#).

4.2 Compile and Load the Generic Application

To compile and load a demonstration application into the evaluation boards, see [Section 1.4, “Loading Applications Into a Board Using the BDM”](#).

4.3 Starting the Generic Application Demonstration

1. Configure the Com Port as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
2. Connect a board to a USB cable and a USB port and ensure it is enabled as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
3. Start the serial communication software and reset the board. The following text appears:

```
Generic Demonstration Application
```

From this point, users can use this empty project to start building their own application.



Chapter 5

Simple ZigBee Test Client (Simple ZTC)

The SMAC for S08 Simple ZigBee Test Client (Simple ZTC) application allows users to test the SMAC primitives by sending special data frames through the Serial/USB port. The Simple ZTC application requires two boards and the appropriate Python scripts to communicate with the boards or a serial port packet generator.

5.1 Generating the Simple ZTC Application

Generate the application with BeeKit and download to each board using the steps as shown in [Chapter 1, “Introducing the SMAC”](#).

5.2 Compile and Load the Simple ZTC Application

To compile and load a demonstration application into the evaluation boards, see [Section 1.4, “Loading Applications Into a Board Using the BDM”](#).

1. Download the Simple ZTC application to the board that will be used as the tester.
2. Download the WirelessUART application to a board that will be used as the subject of the test.

5.3 Starting the Simple ZTC Application

1. Configure the Com Port as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
2. Connect the boards to USB cables and USB ports and ensure they are enabled as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
3. Connect the board that is running the WirelessUART application to the serial port of the PC and press that board’s reset button. A menu appears on the PC display as shown in [Figure 5-1](#).

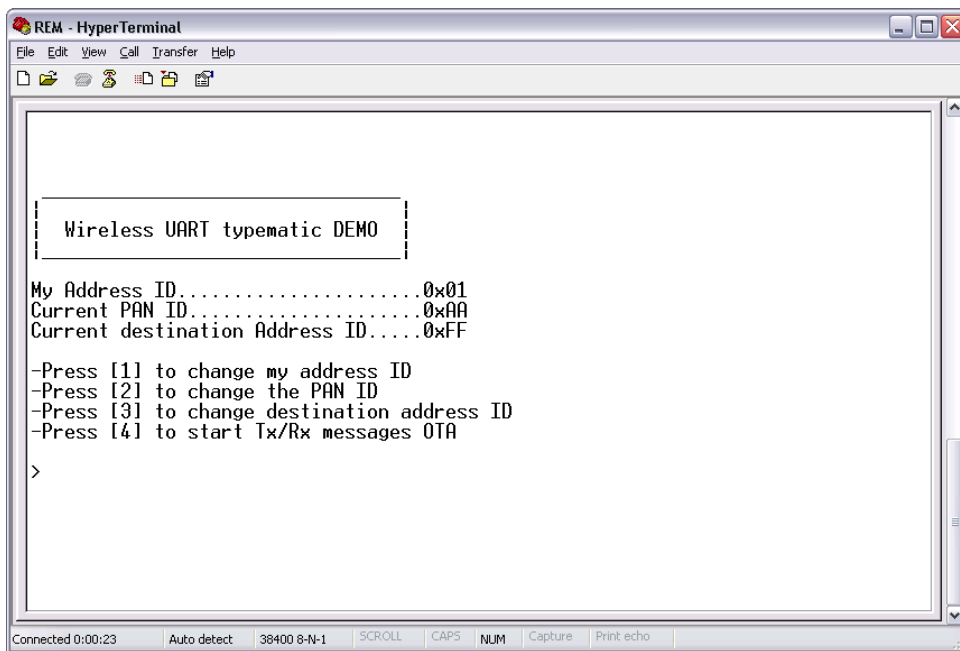


Figure 5-1. Wireless UART Application

4. Start receiving packets over the air by choosing Option 4 using the PC Keyboard.
5. Launch Freescale Test Tool 12 as follows:

Start -> Programs -> Freescale Beekit -> Test Tool 12 -> Test Tool 12

See the Freescale *Test Tool User's Guide* for more details about Test Tool.

6. The Test Tool main window appears. Configure a serial connection by reviewing which ports are active as displayed on the left side of the Test Tool main window. Configure a port by clicking on the Settings button located next to the port name as shown in Figure 5-2.

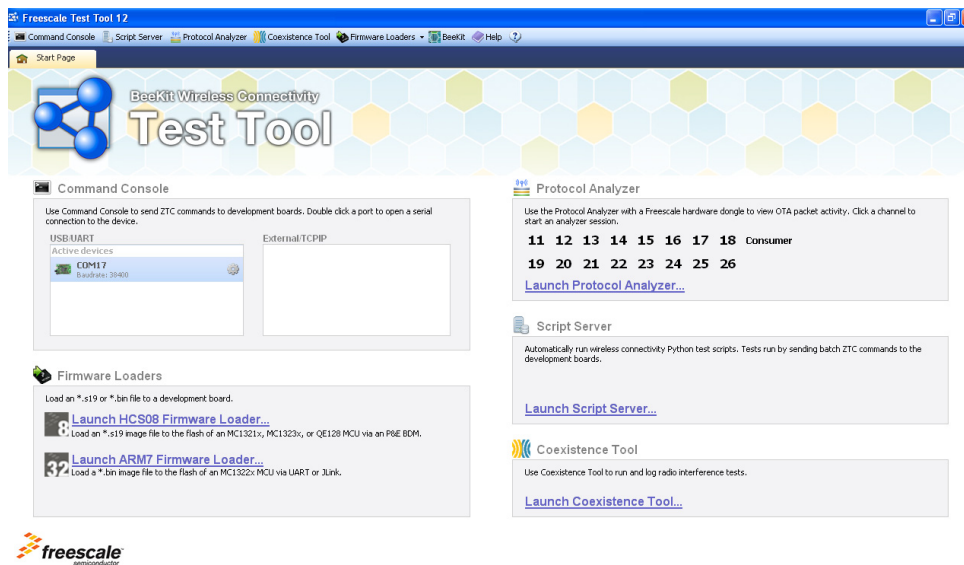


Figure 5-2. Test Tool Communication Settings

7. Set the baud rate at 38400 and check that all the settings are set as shown in [Figure 5-3](#).

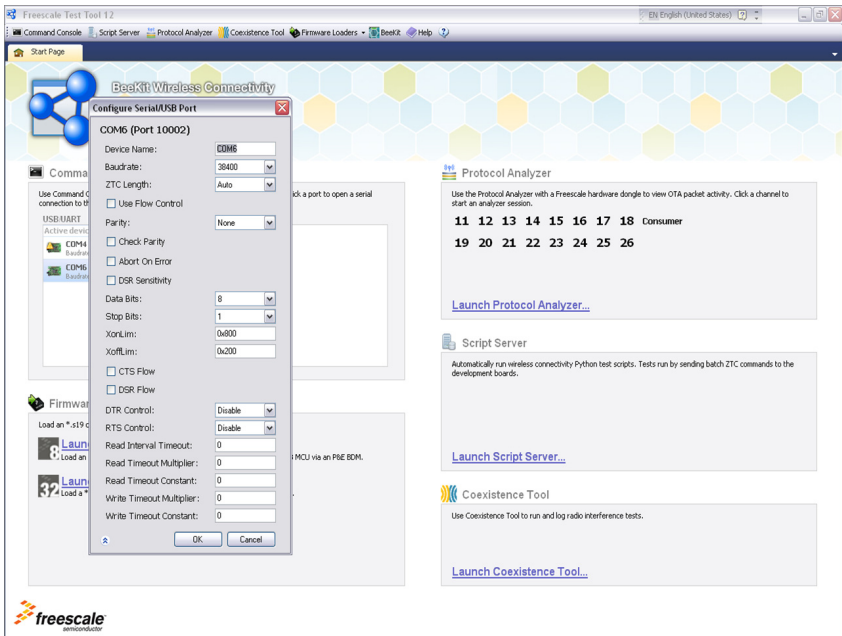


Figure 5-3. List of Devices Window

- 8. Click the OK button.
- 9. As shown in [Figure 5-4](#), open the Script Server application by clicking on the Launch Script Server button. The Script Server window appears as shown in [Figure 5-5](#). See the *Freescale Test Tool User's Guide* for more details about Script Server.

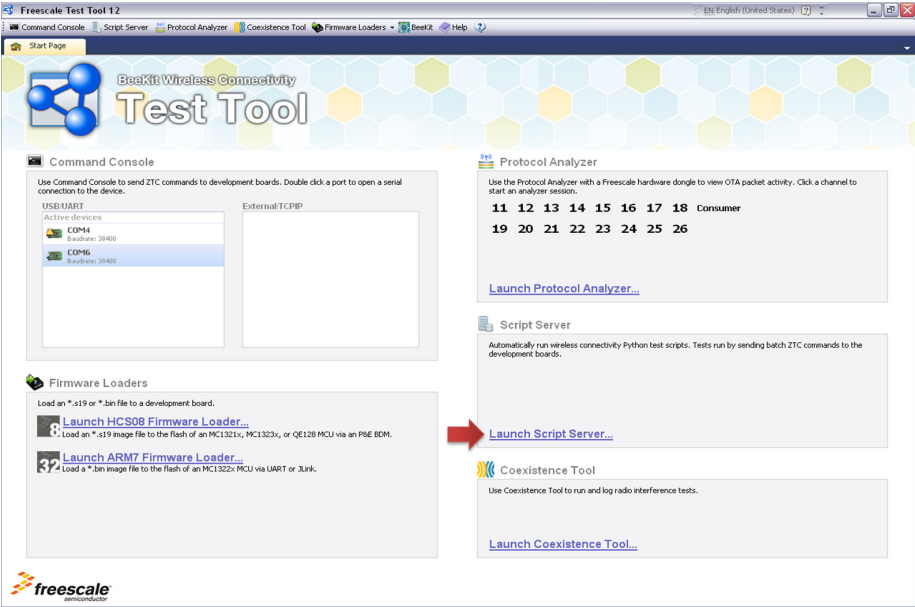


Figure 5-4. Test Tool Adding a Python (py) script

10. Load the Sending Letters To The Wireless Uart.py Python script, which is located in the following directory and shown in Figure 5-5.

~\Program Files\Freescale\Test Tool 12\Scripts\SMAC\Demo Scripts\

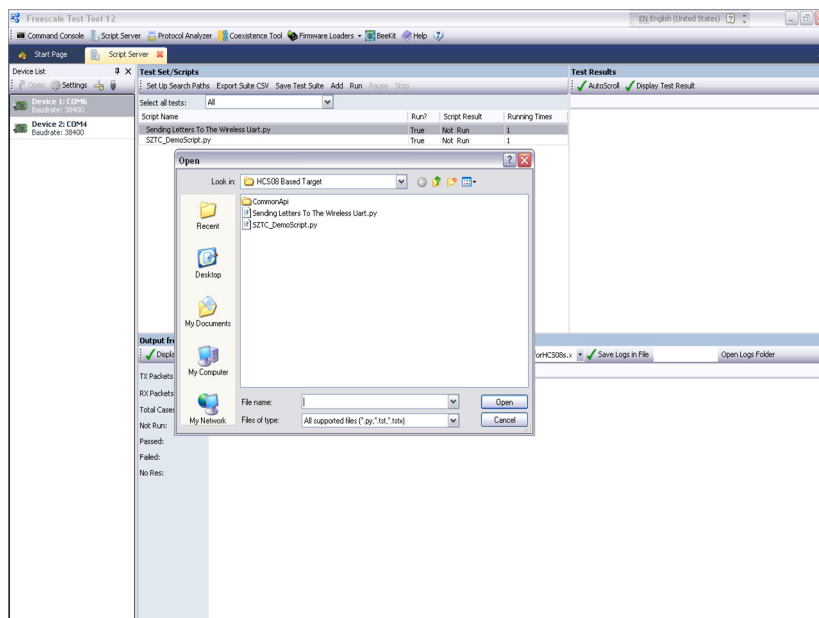


Figure 5-5. Loading a Script

11. Select a script, then click on the Open button to return to the Script Server window.
12. Ensure that the desired script is highlighted, then click the Run button.
13. On the PC running the Serial Terminal application and is connected to the board running the WirelessUART application, the expected end results are shown in Figure 5-6. If this is not the end result, try running the WirelessUART application on the Simple ZTC again.

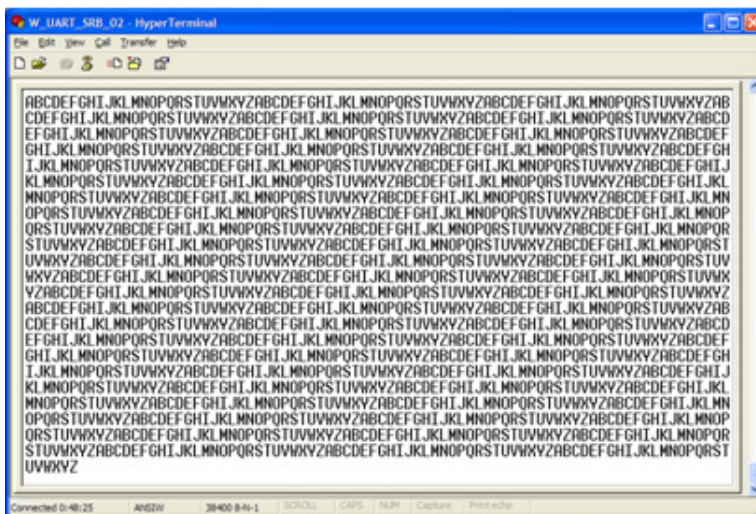


Figure 5-6. Result at the Wireless UART Application

Chapter 6

Repeater Demonstration

The Repeater application repeats all the messages received in a specific channel to extend the range of a Point-to-Point link as shown in [Figure 6-1](#). The Repeater application is configured through the UART/USB virtual com port so the application can determine the working channel, the output power, and other statistics about packets received, dropped and repeated.

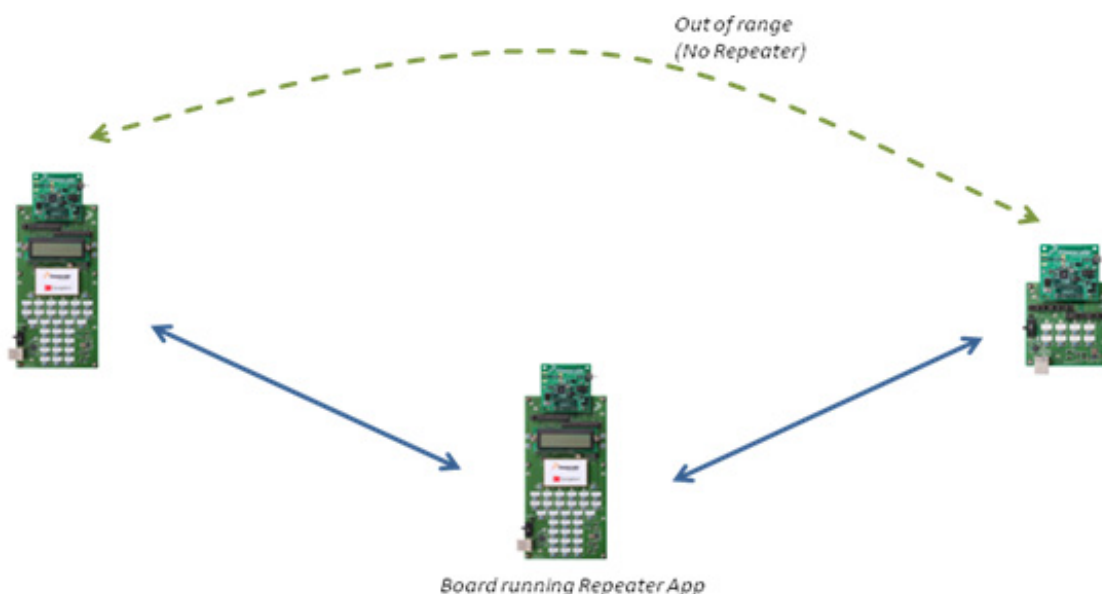


Figure 6-1. Repeater Basic Functionality

NOTE

The boards used in [Figure 6-1](#) are just an example. Any of the boards listed in [Section 1.1, “Supported Hardware”](#) can be used as a repeater.

6.1 Generating the Repeater Application

Generate the application with BeeKit and download to each board using the steps as shown in [Section 1.3, “Generating an Application in BeeKit”](#).

6.2 Compile and Load the Repeater Application

To compile and load the demonstration application into the evaluation boards, see [Section 1.4, “Loading Applications Into a Board Using the BDM”](#).

6.3 Starting the Repeater Application

1. Configure the Com Port as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
2. Connect the board to a USB cable and a USB port and ensure it is enabled as shown in [Section 1.6, “UART/USB Virtual Com Port Setup”](#).
3. Press the Reset button once on the board. The terminal displays text as shown in [Figure 6-2](#).

```

Repeater Demo - Hyper Terminal
File Edit View Call Transfer Help
-----
Repeater Application Menu
-----
-Press [s] to show statistics
-Press [0] to reset statistics
-Press [r] to repeat menu
-Press [m] to switch Repeater mode (Normal - Dummy)
-Press [p] to change output power
-Press [d] to change delay length
-Press [c] to switch channel
-Press [z] to EXECUTE

Type an option and then press Enter

You can also change channel using Switch1, Switch2 and Touchpad if available.

Current Device's configuration is:
App Mode      Channel Power Level      Delay
-----
NORMAL        15      MAX POWER      00001 ms
Ready>
-----
Connected 0:00:27  Auto detect  38400 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

Figure 6-2. Repeater Start Window

6.3.1 Menu Function Overview

The Repeater menu displays the following eight options:

- [s] Shows packet statistics (received, transmitted, bad, dropped)
- [0] Resets packet statistics
- [r] Stops the selected option and returns to the menu screen as shown in [Figure 6-2](#)
- [m] Selects the repeater mode (Normal or Dummy)
- [p] Changes the RF power setting
- [d] Changes the retransmit delay length (Normal repeater mode only)
- [c] Cycles through the RF channels (lowest to highest)
- [z] Runs the selected repeater application option

6.3.2 Changing the Operating Channel

Users can change the operating channel by performing one of the following:

- Press SW 1 to change the channel up one
- Press SW 2 to change the channel down one
- Touchpad swipe right gesture to change the channel up one
- Touchpad swipe left gesture to change the channel down one

6.3.3 LED Functionality (Channel Selection)

- By pressing SW1 or SW2, the LEDs display the current channel number (using binary) as shown in [Figure 6-3](#). The LED display is persistent for few seconds to allow users to see which channel is currently selected. Channels range from 11 to 26 and only a partial list is shown in [Figure 6-3](#).

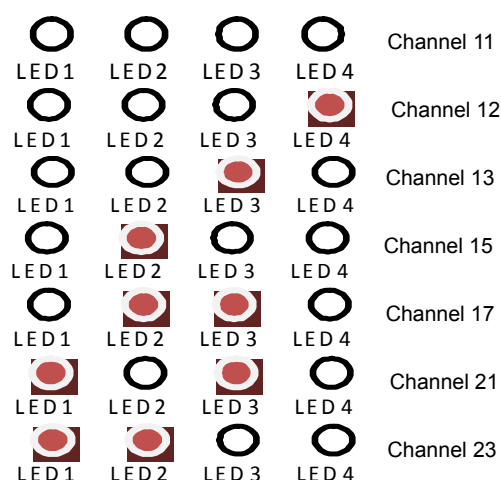


Figure 6-3. Binary Channel LED Display (partial)

6.3.4 Default Values

Default values are as follows:

- Channel 15
- Max output power
- Retransmit delay length 1 ms
- Security disabled (Can only be changed in the Repeater Application source code.)
- 38400 bps (Can only be changed in the Repeater Application source code.)
- Low power mode disabled (Can only be changed in the Repeater Application source code.)
- OTAP supported (Can only be changed in the Repeater Application source code.)

6.4 Using the Repeater Application

To select a repeater application option, type the character that corresponds to the option from the menu as shown in [Figure 6-2](#). The options are NOT case sensitive. To run the selected option, enter “z” and to exit the option and return to the repeater menu as shown in [Figure 6-2](#), type “r”.

1. Start the repeater application as shown in [Section 6.3](#), “Starting the Repeater Application”.
2. Type “s”, then Enter, to show repeater statistics as shown in [Figure 6-4](#).

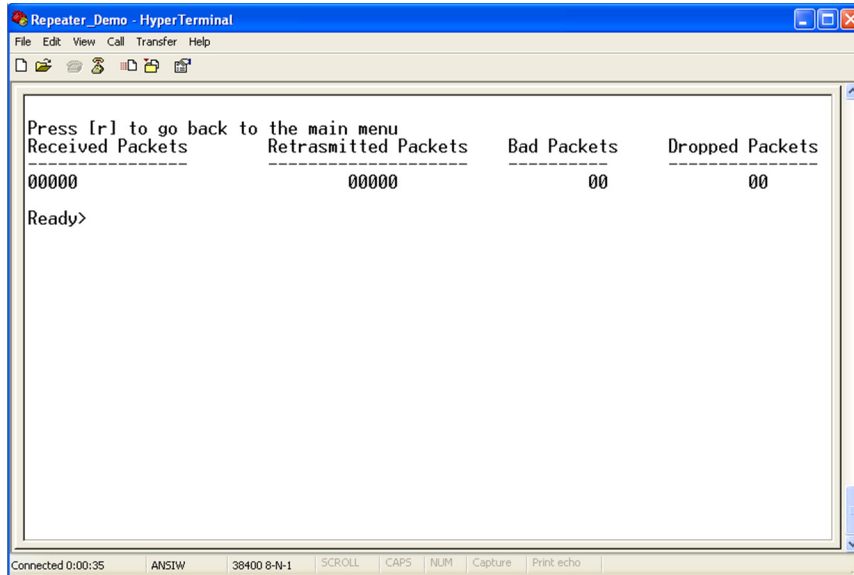


Figure 6-4. Repeater Statistics

If an LCD is available on the board under test, the statistics are displayed as shown in [Figure 6-5](#).

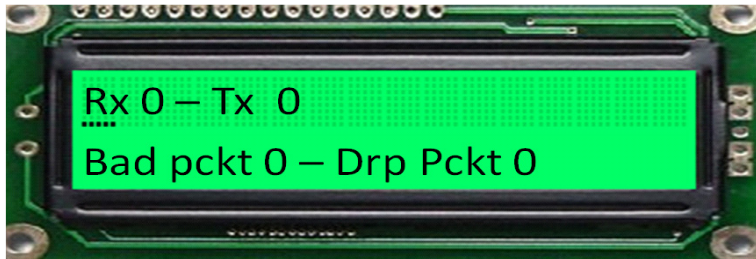


Figure 6-5. Repeater Statistics on LCD

The statistics values are defined as follows:

Received	The number of frames received.
Retransmitted	The number of frames successfully received and retransmitted.
Bad	The number of corrupted packets received (the SMAC status of the packet received is not rxSuccessStatus_c). Bad packets are discarded and not retransmitted.
Dropped	The number of frames received but not retransmitted. This is only used in Normal Mode.

3. Type “0” (zero), then Enter, to reset statistics.
4. Type “m”, then Enter, to switch between the Dummy or Normal repeater operational modes.
In Normal Mode, the application receives a message and a wait “Delay Length“ (option d) to receive other messages before retransmitting. If a second message is received, the original received packet is dropped and not retransmitted. To run the repeater in Normal Mode on channel 11, perform the following tasks on the board used as the repeater.
 - a) Start a serial terminal as shown in [Section 6.3, “Starting the Repeater Application”](#).
 - b) Reset the board that has the repeater application in its FLASH.
 - c) Press the “c“ key as needed until the Channel is set to 11.
 - d) Press the “m” key as needed until the repeater mode is set to NORMAL.
 - e) Press the “z” key to start the repeater.
 - f) Configure and load two other boards with the Wireless UART Demo set to Channel 11. (see [Section Chapter 2, “Wireless UART”](#).)
 - g) Start the Wireless UART application on both boards.
 - h) Keep the boards running the Wireless UART application within RF range of each other. The repeater will not retransmit the packets because both boards are within range of each other.
 - i) Now place the boards running the Wireless UART application out of RF range of each other, but each within range of the repeater board. The repeater retransmits the messages sent and thus increases the range of the Wireless UART application.
 - j) To stop the repeater, press “r” at the terminal interface being used by the repeater application.

In Dummy Mode, anything received by the repeater is retransmitted as soon as possible. To run a dummy repeater on channel 11, perform the following tasks on the board used as the repeater.

- a) Start a serial terminal as shown in [Section 6.3, “Starting the Repeater Application”](#).
- b) Reset a board that has the repeater application in its FLASH.
- c) Press the “c“ key as needed until the Channel is set to 11.
- d) Press the “m” key as needed until the repeater mode is set to DUMMY.
- e) Press the “z” key to start the repeater.
- f) Configure and load two other boards with the Wireless UART Demo set to Channel 11. (see [Section Chapter 2, “Wireless UART”](#).)
- g) Start the Wireless UART application on both boards.
- h) Every character transmitted from on Wireless UART board to the other shows its echo on the repeater terminal because the message is immediately being retransmitted by the repeater.
- i) To stop the repeater, press “r” at the terminal interface used by the repeater application.

- Type “p” to change the transceiver output power. As shown in [Figure 6-6](#), the power increases each time “p” is typed until the maximum value is reached. (0 to 14) When the maximum value (14) is reached, it resets to the minimum value (0) and begins increasing again. See the appropriate device reference manual for details about the power settings.

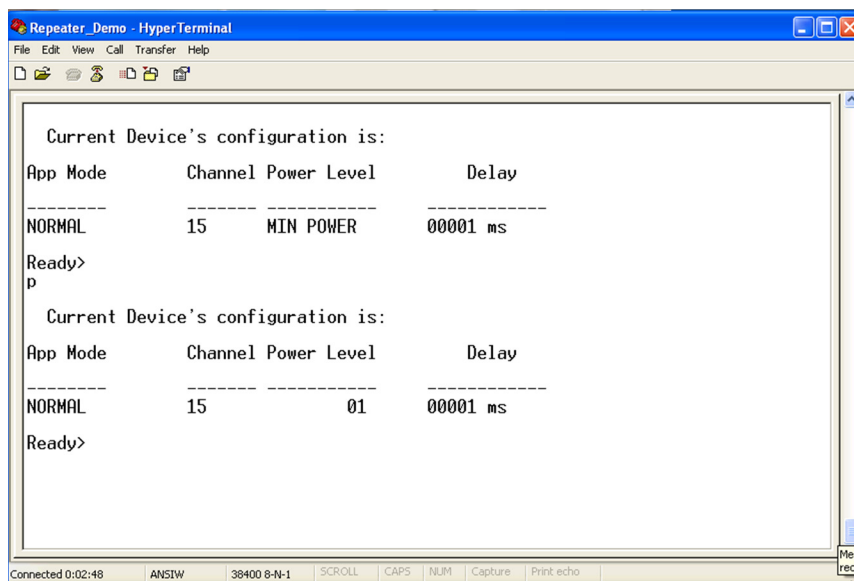


Figure 6-6. Change Output Power

- Type “d” to change the “Delay Length” value as shown in [Figure 6-7](#). The range is 00001 ms to 65535 ms and users must enter all five digits.

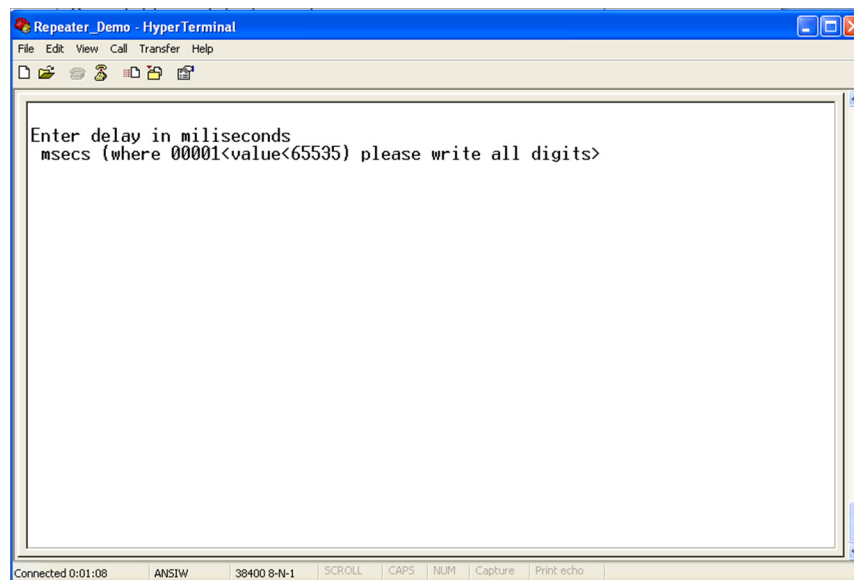


Figure 6-7. Change Delay Length

- Type “c” to change the RF channel (11 to 26). The channel change appears as shown in [Figure 6-8](#)

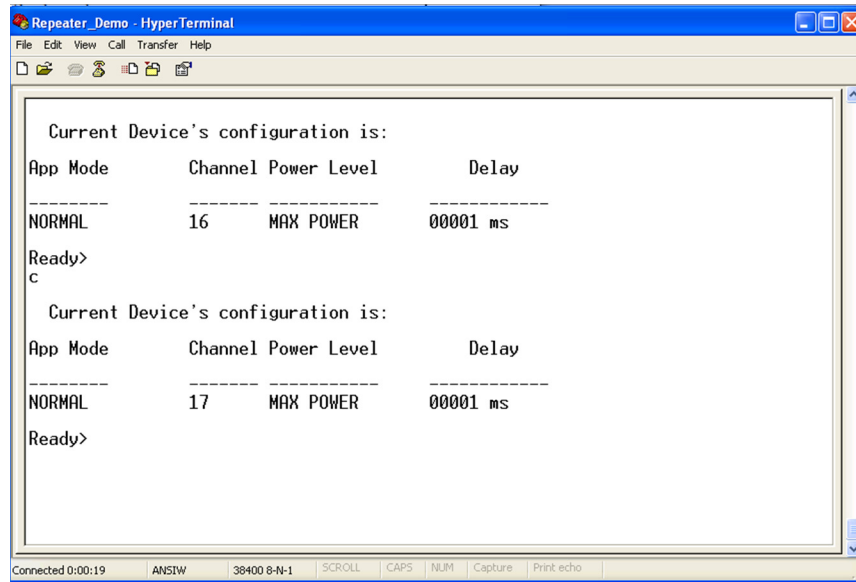


Figure 6-8. Switching Channels

Chapter 7

Range Demonstration

The Range Demonstration application allows users to determine the maximum RF range between two development boards. One board is connected to a PC (Local Node) and the other board (Remote Node) is placed at varying distances from the Local Node. As shown in [Figure 7-1](#), this example uses a 1323x-REM as a Local Node and another 1323x-REM as a Remote Node. Both REMs have a 1323x-MRB plugged into them. See [Section 1.1, “Supported Hardware”](#), for a list of supported hardware.



Figure 7-1. Range Demonstration Basic Operation

The Range Demonstration application can be set up with any combination of boards as Local Nodes and Remote Nodes as long as the Local Node has a serial connection. (Local Node connected to the PC through a UART connection.) Users can manually select the Remote Node or Local Node from the BeeKit Platform Editor, or in the `app_config.h` file. See the *Simple Media Access Controller (SMAC) for the HCS08 Reference Manual* for more information about file contents and packet structure.

To see which Freescale boards are supported, see [Section 1.1, “Supported Hardware”](#).

The distance between the Local and Remote Nodes is displayed in the Range Demonstration GUI as a scale. (The scale is longer when the distance is greater between the nodes and shorter when the distance is shorter between the nodes). In the Range Demonstration GUI, the Local Node appears as a static image and the Remote Node as a variable image that moves on the GUI to represent distance. The distance is an interpretation of the Link Quality Indication (LQI) value being reported by the Local Node from the Remote Node. The Remote Node displays the relative signal strength of packets coming from the Local Node using the LEDs on the Remote Node board. The Range Demonstration application does not display actual distance. The distance is represented proportionally based on the LQI. The LQI value may be influenced by the antennas polarization and the area where the test is being performed. Freescale recommends that the polarization of the antennas remains similar during the test and that it is done with the antennas in line of sight.

7.1 Generating the Range Demonstration Application

1. Generate the application with BeeKit and download to each board using the steps as shown in [Section 1.3, “Generating an Application in BeeKit”](#).
2. Ensure that Remote_Node is selected for one board and Local_Node for another board from the “Device Type” option in BeeKit as shown in [Figure 7-2](#).

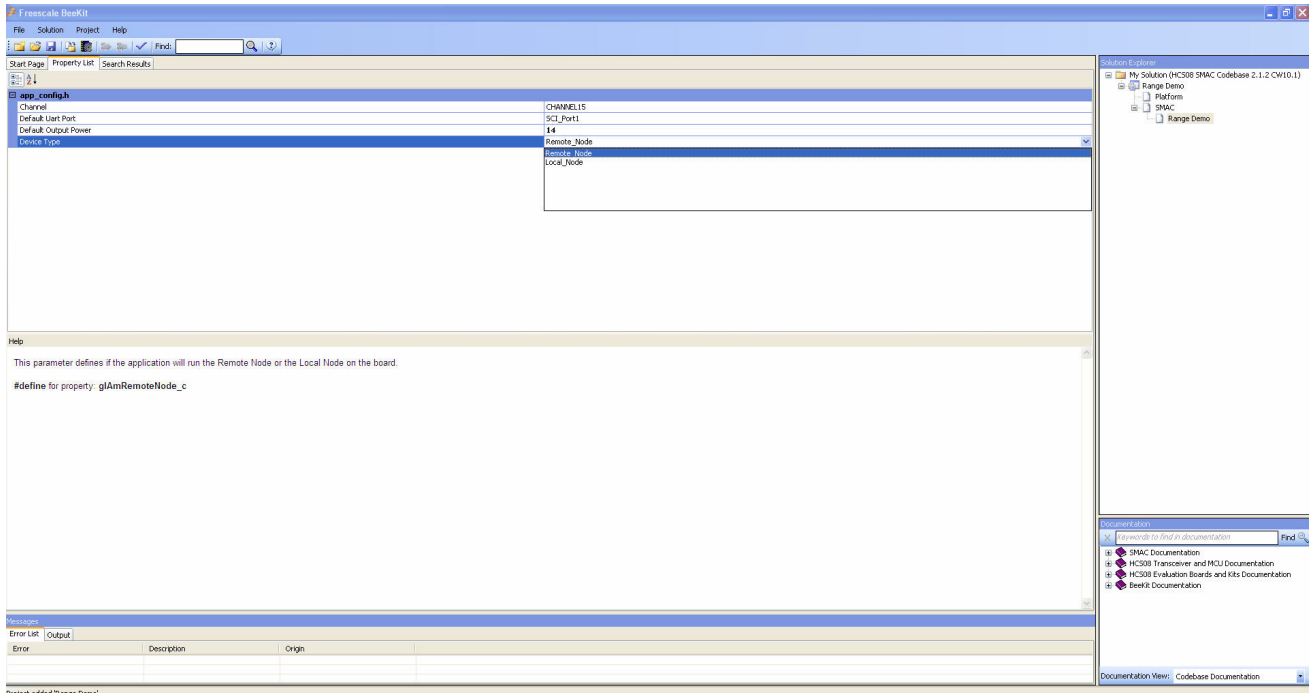


Figure 7-2. Node Selection in BeeKit

7.2 Compile and Load the Range Demonstration Application

Compile and load demonstration applications into evaluation boards as shown in [Section 1.4, “Loading Applications Into a Board Using the BDM”](#).

7.3 Range Demonstration Setup

1. Connect the board being used as the Local Node to the PC and power it on.
2. Power on the Remote Node.
3. Start the Range Demonstration application from Windows by clicking on Start -> Programs -> Freescale BeeKit -> Range Demo -> Range Test GUI. The Range Demonstration main window appears as shown in Figure 7-3.

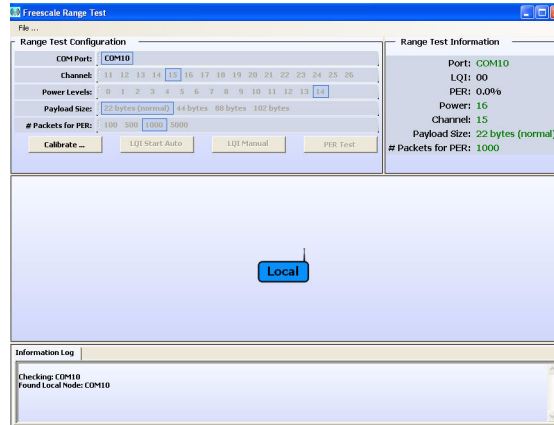


Figure 7-3. Range Demonstration Main Window

The Range Demonstration application automatically detects the COM port to which the Local Node is connected. The port is displayed in the “Range Test Information” display area located in the top right corner of the Range Demonstration main window and the Local Node icon appears in the main display area as shown in Figure 7-3.

If the application does not find a device attached to a port, the text, “No Local Node Found” appears instead of the port number as shown in Figure 7-4.

NOTE

If the Range Demo does not find the Local Node, check the USB cable connection from the PC to the Local Node, cycle the power on the Local Node and restart the Range Demo application.

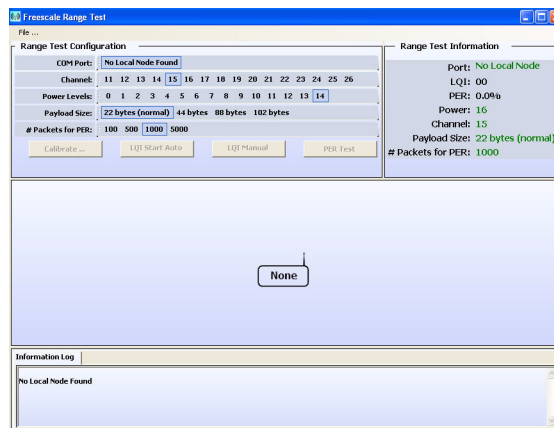


Figure 7-4. Unable to Find Local Node

7.4 Range Test Calibration

1. To begin the LQI test, place the boards used as the Local and Remote nodes close to each other so the application can calibrate. The boards should be one next to each other for calibration as shown in [Figure 7-5](#).

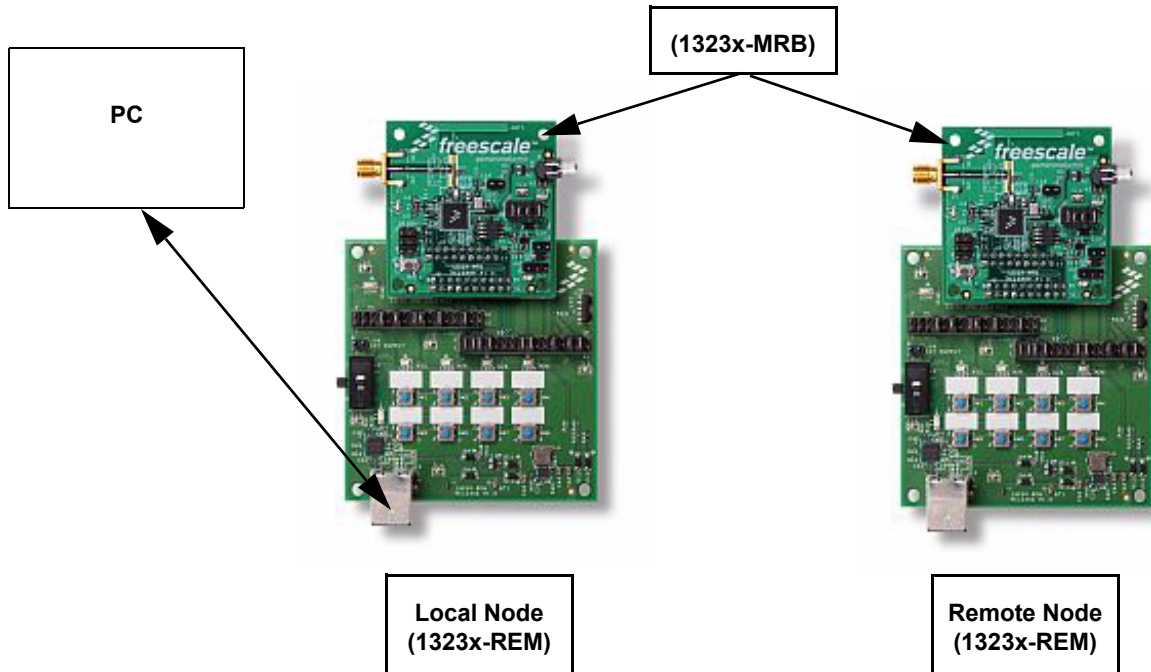


Figure 7-5. Example Board Orientation and Placement for Calibration

NOTE

The LQI value is affected by the characteristics of the antenna. For example, the F-antenna on the boards is not perfectly omnidirectional, so the LQI will vary depending on how the boards are placed next to each other, even if the antennas are at the same distance.

For calibration, the Remote Node sends packets to the Local Node. The Local Node sums up the LQI of all the packets it receives and divides it by the total number of packets sent (average). Then a packet is sent by the Local Node to the Remote Node that contains the calculated LQI.

2. After the local and remote node boards are programmed, powered on and set at the distance as shown in [Figure 7-5](#), click on the “Calibrate...” button in the Range Demonstration application main window and wait for the calibration to complete.

The text, “Calibration complete: success” appears in the Information Log and the Local and Remote nodes appear as shown in [Figure 7-6](#).

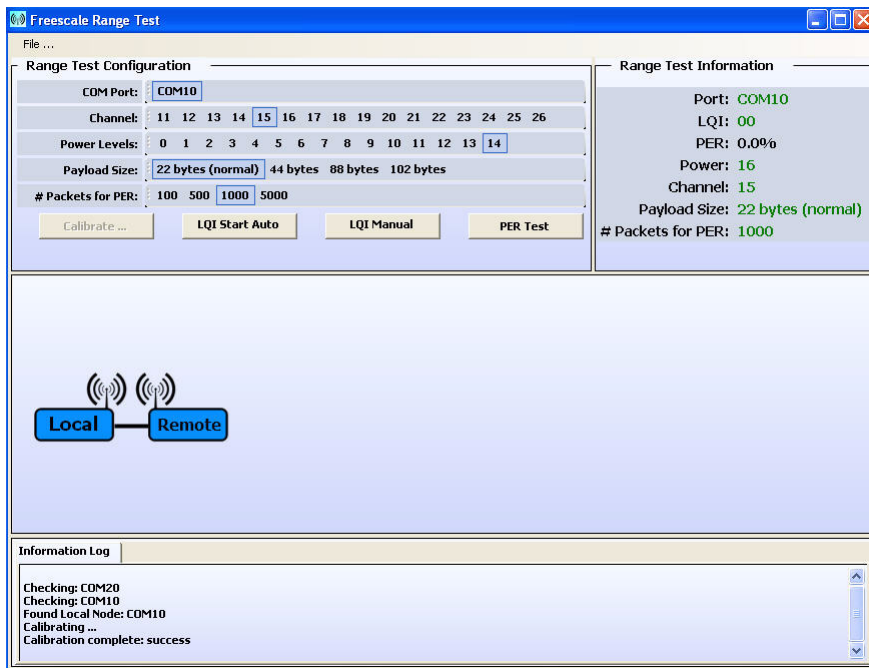


Figure 7-6. Calibration Success

3. If a calibration time out occurs, follow the instructions as they appear in the information log tab as shown in Figure 7-7.

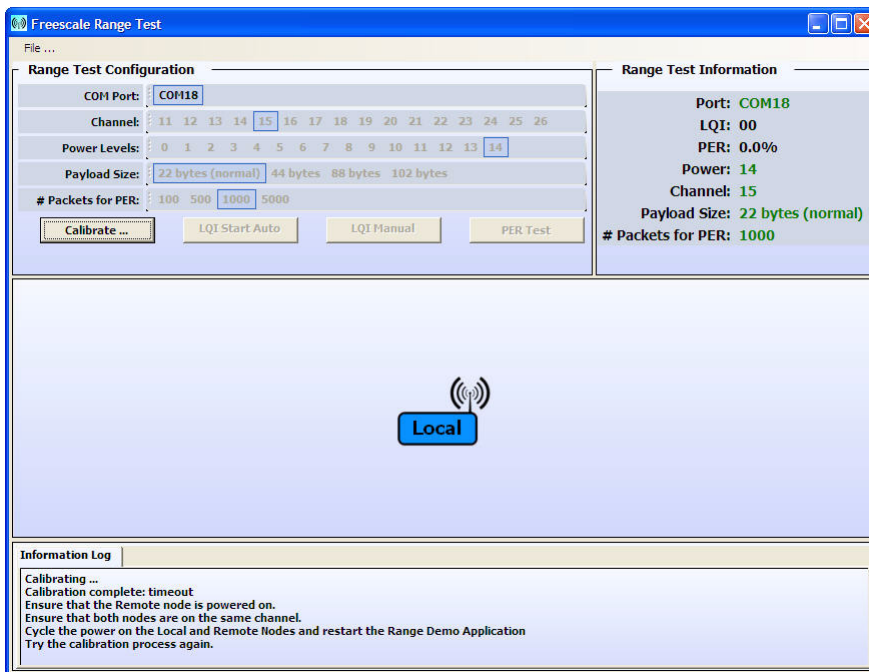


Figure 7-7. Calibration Time Out

7.5 Range Demonstration Operation

Once calibration is complete, users can run the Range Demonstration in one of two tests, Link Quality Indicator (LQI) or Packet Error Rate (PER). The LQI Test can run in either auto or manual modes and the PER test is a “run once” test.

7.5.1 PER Test

1. Clicking the “PER test” button runs the PER test one time only. Click the “PER test” button and the test sends the number of packets as displayed in the “# Packets for PER” field of the Range Demonstration main window as shown in [Figure 7-8](#). A “Please Wait” image shows the PER Test is already running.



Figure 7-8. PER Test Running

When the PER Test completes, the application displays if the PER test passed or failed. If more than 1% of the packets were lost, the PER test considers it a failed test as shown in [Figure 7-9](#).



Figure 7-9. PER Test Failed

If less than 1% of the packets were lost, the PER test considers it a passed test as shown in [Figure 7-9](#).



Figure 7-10. PER Test Passed

NOTE

All the options as they appear in the Range Demonstration main window (Baud Rate, Channel, Power Level Payload Size and # Packets for PER) can be changed, but only when the tests are NOT running.

7.5.2 LQI Test

1. To start the LQI test in auto mode, click on “LQI Start Auto” button. In this mode, the local and remote nodes exchange information and the LQI value is continuously reported to the PC as shown in [Figure 7-11](#).

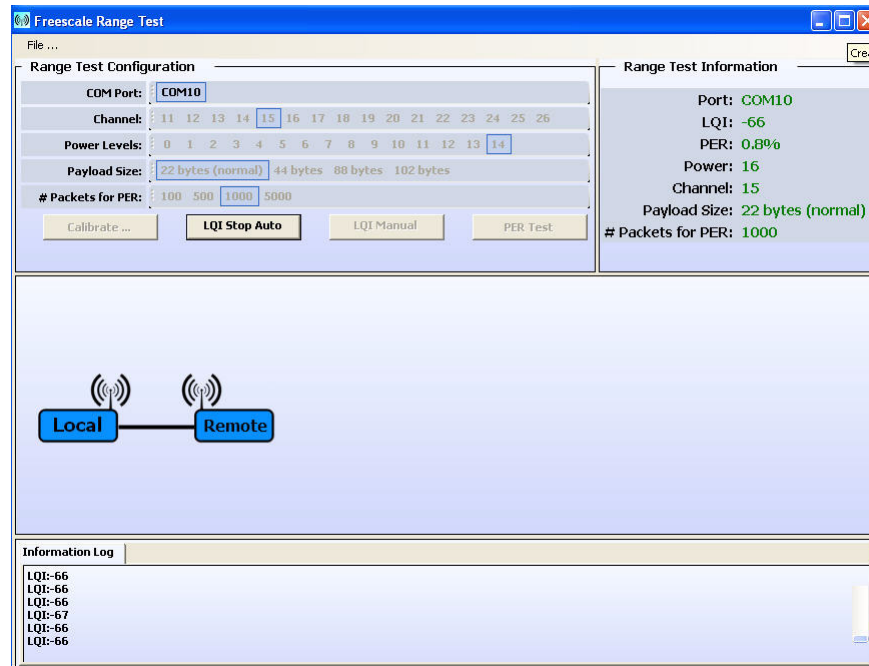


Figure 7-11. LQI Test (Auto Mode) Running

2. To stop the LQI test, click on the “LQI Stop Auto” button.
The LQI value is also automatically refreshed and displayed on the board LEDs. The LEDs show the Link Quality of the signal between the two nodes. So, when the nodes are close to each other and thus the LQI is better, more LEDs turn on and as the boards are placed further apart, only one LED is turned on to represent the greater distance and a poor LQI as shown in [Figure 7-12](#).

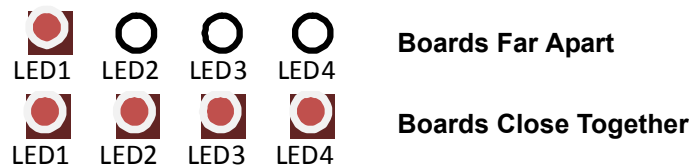


Figure 7-12. LQI Test (Auto Mode) Board LEDs

7.5.3 LQI Manual Mode

1. Press the “LQI Manual” button on the Range Demonstration main window to run the LQI test once. Each time the “LQI Manual” button is clicked, the test is run and the information is updated as shown in Figure 7-13.

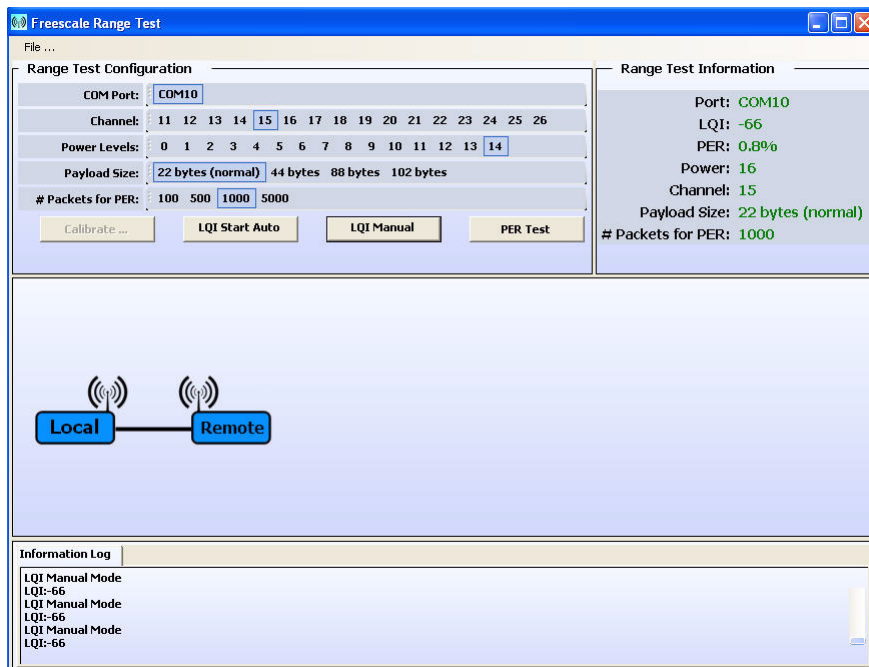


Figure 7-13. LQI Test (Manual Mode) Running

Chapter 8

Accelerometer Demonstration

The Accelerometer Demonstration application shows various uses of the Freescale X, Y, and Z axes accelerometers located on-board the 1321x-SRB and 1323x-RCM development boards.

Any combination of boards currently supported in the SMAC Codebase can be employed for this application, but the Remote Node board must be either a 1321x-SRB or 1323x-RCM because these are the only boards that have the on-board accelerometer. [Figure 8-1](#) shows the basic system setup.

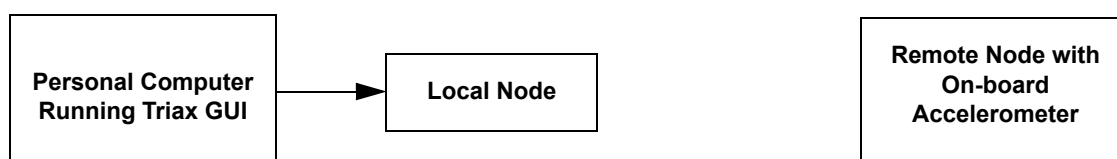


Figure 8-1. Simplified Configuration

8.1 Generating the Accelerometer Demonstration Application

Generate the application with BeeKit and download to each board using the steps as shown in [Section 1.3](#), “Generating an Application in BeeKit”.

8.2 Compile and Load the Accelerometer Demonstration Application

To compile and load a demonstration application into the evaluation boards, see [Section 1.4](#), “Loading Applications Into a Board Using the BDM”.

8.3 Local Node and Remote Node Board Setup

To setup both boards for operation (Local Node and Remote Node), perform the following tasks.

8.3.1 Board One Setup (Local Node)

1. Connect the board to be used as the Local Node to the PC using an appropriate USB or serial cable and open the Triax GUI Application located by choosing:
Start -> All Programs -> Freescale Beekit -> Triax.
2. Power on the Local Node by setting the Power Switch to the ON position.
3. Press the Reset Button once. The LEDs will blink twice and if a buzzer is available, it will make a short sound. The application is now running and in receive mode.

8.3.2 Board Two Setup (Remote Node)

1. Ensure that the 1323x-RCM or 1321x-SRB is powered by a USB connection, batteries, or an appropriate DC power supply. Power on the Remote Node by setting the Power Switch to the ON position.
2. Press the Reset Button once. The LEDs will blink twice and if a buzzer is available on the board, it will make a short sound. The application is now running and in receive mode.
3. Press button SW1 once and LED3 momentarily flashes to indicate that the Accelerometer Mode (Application Two) is running.
4. The Accelerometer application sends a ping packet to the Local Node at a preset interval to report its status and information concerning the X, Y and Z axes. This information is displayed on the Remote Node by a flashing LED2.
5. When the Remote Node is moved it toggles its LED1 and sends a packet to the Local Node with updated axis information.
6. The Local Node toggles its LED1 to indicate that the Remote Node was moved.

8.4 Accelerometer Demonstration Application Overview

There are six (6) available applications within the Accelerometer Demonstration:

1. Local Node Application (Application One) — This application runs on the Local Node and communicates with the PC and the Triax GUI when appropriate. The Local Node application is used in conjunction with every application running on the Remote Node. There is no need to change the application on the Local Node.
2. Accelerometer Application (Application Two) — This application runs on the Remote Node and communicates with the Local Node. The Local Node then communicates with the Triax GUI.
3. Free Fall (Application Three) — This standalone application runs on the Remote Node and does not need the Triax GUI to run. This application detects when the Z axis accelerometer is measuring approximately 0g's. If a free fall is detected and if a buzzer is available on either board, then the buzzer is activated on both the Local and Remote Node.
4. Shock Detection (Application Four) — This standalone application runs on the Remote Node and does not need the Triax GUI to run. If this application detects movement in any of the X, Y or Z axes, it flashes LED1 and if a buzzer is available on either board, then the buzzer is activated on both the Local and Remote Node as follows:
 - Once if there is a change in the X axis.
 - Twice if there is a change in the Y axis.
 - Three times if there is a change in the Z axis.
5. Anti-Theft (Application Five) — This standalone application runs on the Remote Node and does not need the Triax GUI to run. If the Remote Node is tilted more than 15 degrees from the default center position, and if a buzzer is available on either board, then the buzzer is activated on both the Local and Remote Node.
6. Battery Save (Application Six) — This standalone application runs on the Remote Node and does not need the Triax GUI to run. This application shows how an accelerometer can be used as a power saving component on a battery operated board. When the Remote Node is not moved for at least

five (5) seconds, the board simulates powering down by turning off its buzzer. Once the Remote Node detects movement, it simulates a power up and if a buzzer is available on either board, then the buzzer is activated on both the Local and Remote Node.

To switch between applications, press SW1 on the board. The LEDs momentarily indicate in binary, the application selected as shown in [Figure 8-2](#).

NOTE

If a board does not have an accelerometer on-board, the application will not allow users to change from Application One.

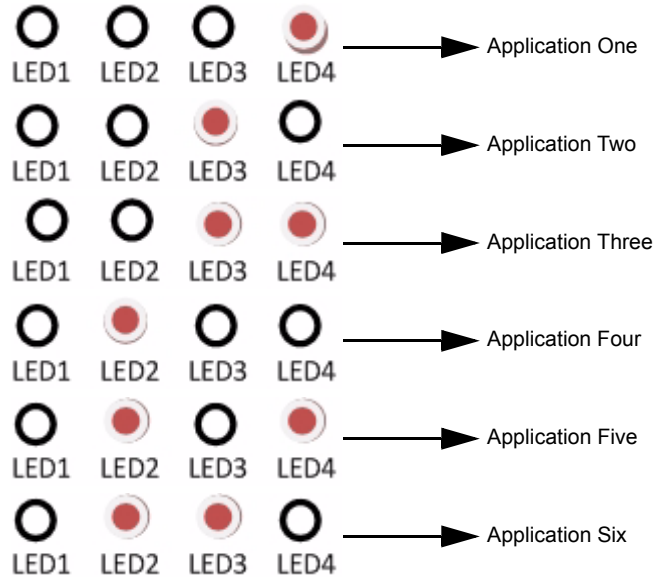


Figure 8-2. Binary Display of Selected Application

8.5 Triax GUI Application Setup

1. By default, the Triax GUI is installed with BeeKit. However, if for any reason the Triax GUI was not installed, navigate to and run the `Triax.exe` file located in the default BeeKit installation directory.
2. From Windows, select Start->Programs->Freescale BeeKit->Triax.
3. The Triax GUI is set up to run on COM ports 1-10. To check which COM port is being used by the USB, perform the following tasks:
 - a) Open the System Properties window by clicking on Start->Settings->Control Panel->System.
 - b) Select the Hardware tab, and click the Device Manager button. The Device Manager window appears as shown in [Figure 8-3](#).
 - c) Scroll to the “Ports” label and expand the tree by clicking the “+” sign to show the COM ports in the system.
 - d) The COM Port selected by the system is titled `USB Serial Port (COM24)` as shown in [Figure 8-3](#).

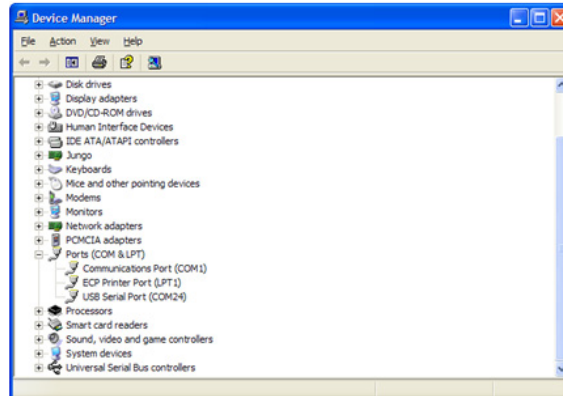


Figure 8-3. COM Port Determination in Device Manager

4. If the COM port chosen for `Freescale Zigbee/802.15.4 MAC COM Device` is not a port within Ports 1-10, then perform the following tasks:
 - a) Double click on `Freescale Zigbee/802.15.4 MAC COM Device` in the Device Manager window and the Properties window will appear.
 - b) Select the Port Settings tab and then click the Advanced button.
 - c) Go to the Com Port Number drop down menu and select a COM port between 1-10 that is not in use.

8.6 Running the Triax GUI

After port selection, the Triax GUI main window appears as shown in [Figure 8-4](#).



Figure 8-4. Triax GUI Main Window

When the accelerometer is moved, the Remote Node sends data to the Local Node to update its data. Data is only sent when the changes are within system tolerance.

The receiver receives these packets and updates its cached value of the accelerometer data. The PC periodically sends a command string through the USB (or serial port) to the Local Node that requests the accelerometer data. This data is interpreted by the Triax GUI to enable several applications and demonstrations.

8.6.1 Calibration Procedure

To improve accelerometer application accuracy and ensure that the X and Y axes are “zeroed out” as much as possible, perform the following tasks.

1. Ensure that the Triax GUI is running.
 1. Press the Reset button once on the Local Node. (Once reset, only the receive application runs.)
 2. Press the Reset button once on the Remote Node.
 3. Place the Remote Node on a flat surface (component side up).
 4. Press SW1 once on the Remote Node to select the Accelerometer Mode (Application Two). LED3 momentarily lights to indicate that the Accelerometer Mode is running.
 5. Open the “XYZ” view from Triax GUI.
 6. If the data points for the X, Y, and Z axes shown on the Triax GUI are not near the origin of the graph (axis point 0,0) perform the following steps.
 - a) Press SW4 on the Remote Node once. While calibration is being performed, all LEDs are lit.
 - b) Verify that the Remote Node is calibrated by reviewing the X, Y, and Z axes on the Triax GUI. The squares on the display should be close to axis point 0,0.
 - c) If the calibration is still incorrect, repeat the calibration procedure until the data points are at or as close as possible to axis point 0,0.

8.7 Verifying Operation

1. The Remote Node routinely sends a ping packet to the Local Node as indicated by LED2. The Remote Node quickly blinks LED2 indicating that the ping packet was sent. When the Local Node receives the ping packet it toggles LED2.
2. To check if the Remote Node is working, move the Remote Node board. LED1 on the Local Node will blink to indicate that the Remote Node was moved.
3. When using the default application (Application 1), data that represents movement of the Remote Node is sent to the Local Node. When the Local Node receives this data it LED1 toggles.
4. Select the Raw Data applet from the Triax GUI main window. The Raw Data applet shows the X, Y, and Z axes as reported by the Remote Node. When the Remote Node is laying flat (component side up), the raw values should read approximately 0g for both the X and Y axes. However, the Z axis should read approximately 1g. As shown in [Figure 8-5](#), when the Remote Node is moved, the values of the X, Y, and Z axes are updated on the Raw Data applet.

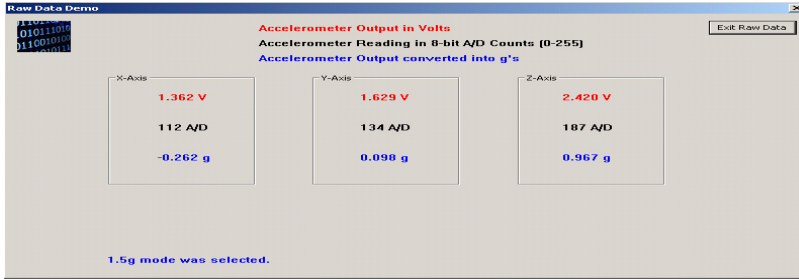


Figure 8-5. Raw Data Demonstration

NOTE

The 1321x-SRB and 1323x-RCM boards contain different accelerometer devices, so the displayed data and behavior of the Accelerometer Demonstration may vary from one platform to another.