C-nRF52-SKY66112 BLE Long Range Reference Design

Demonstration Software

User Guide

V1.0

February 6, 2017

Table of Contents

Introduction

notWired.co has created a Long Range (or Range Extension) BLE reference design based on Nordic Semiconductor BLE SOCs and Skyworks Solutions Front End Modules (FEM). The solution is based on the Arduino platform and allows customers to easily port previously developed code while extending the range of their solution.

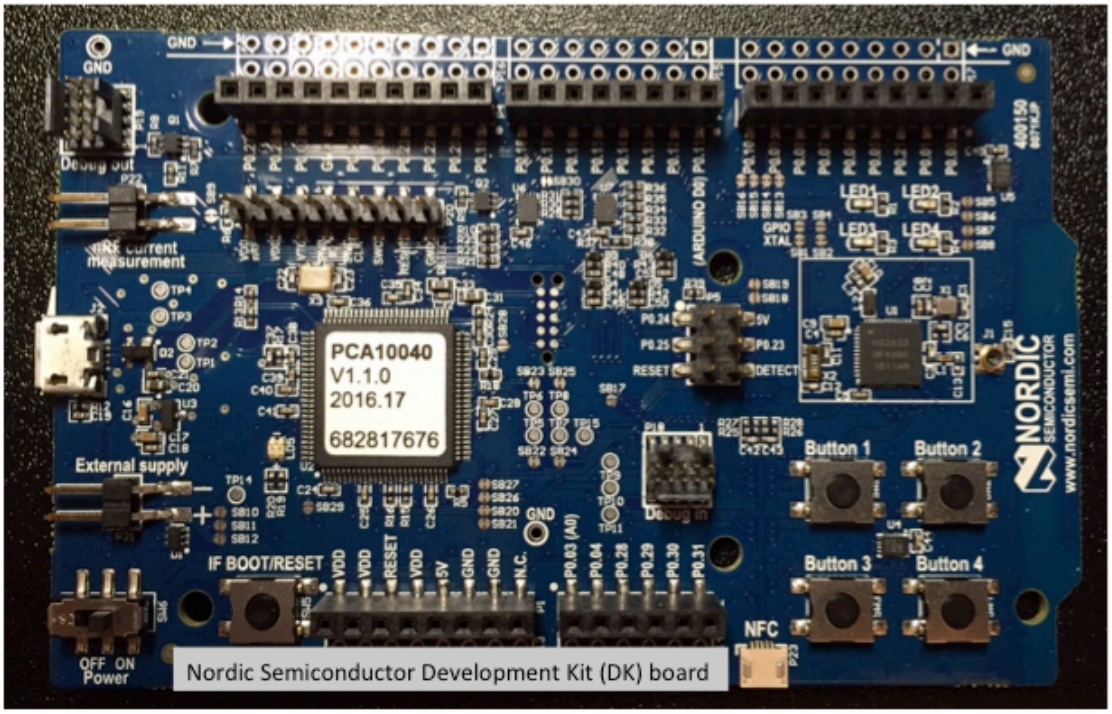
The design is easily connectable to the Nordic Development Kit, separately available from Nordic Semiconductor. Utilizing Skyworks Solutions state of the art Front End Modules provides for elegant solutions for customers looking to reach BLE nodes previously thought unreachable.

What You Will Need

To program and evaluate the Notwired.co BLE reference design, the user will need access to the following hardware and software.

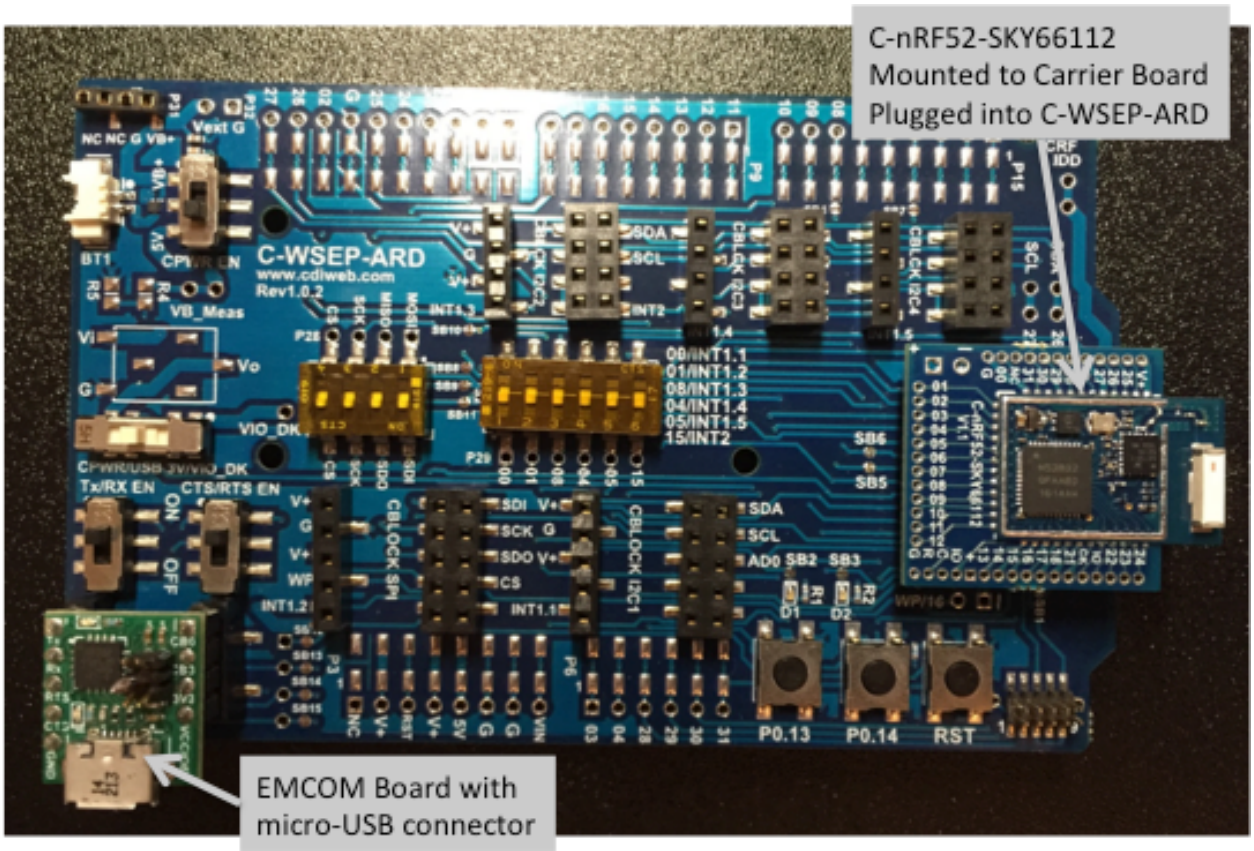
Required Hardware:

- (1) Nordic Development Kit (DK) board. This board will be used to program the target devices. This board is available from Nordic Semiconductor.



Nordic Semiconductor Development Kit (DK) board

- (2) Notwired.co C-WSEP-ARD evaluation boards with included EMCOM boards for USB-to-serial conversion
- (2) Notwired.co C-NRF52-SKY66112 reference design boards, which will be plugged into the mating headers on the C-WSEP-ARD boards.



- (2) Personal Computers that can run a RS232 terminal emulation program and have an open USB port. Laptop computers provide convenience when range testing the modules, due to their portability.
- (2) USB to Micro-USB interface cables

Required Software:

- Nordic Semiconductor "nRF Go" application, which will be used to program the target devices.
- Terminal emulation software for RS232-to-USB terminal emulation. For PC based computers, the program "Tera Term VT" is a good example. For MacOS based computers, the program "SerialTools.app" is a good example.
- Hex file containing the Nordic Semiconductor BLE SoftDevice. Version 3.1.0 has been tested with the Notwired.co application software.
- Hex file containing the Notwired.co "Central" application code.

- Hex file containing the Notwired.co “Peripheral” application code.
- Note that these three hex files may be downloaded together from the Notwired.co website.

Optional Software:
If the user plans to modify the source code files, the following software/files will be required.

- Keil Compiler (Keil uVision5)
- Notwired.co .zip archive containing the source code files.

WARNING: The Notwired.co source code .zip file in a large download, approximately 110 MB. It contains all of the Nordic Semiconductor SDK files in addition to the Notwired.co source code files. Be sure to install this code into a separate directory if you are already using the Nordic SDK, at it will over-write existing files.

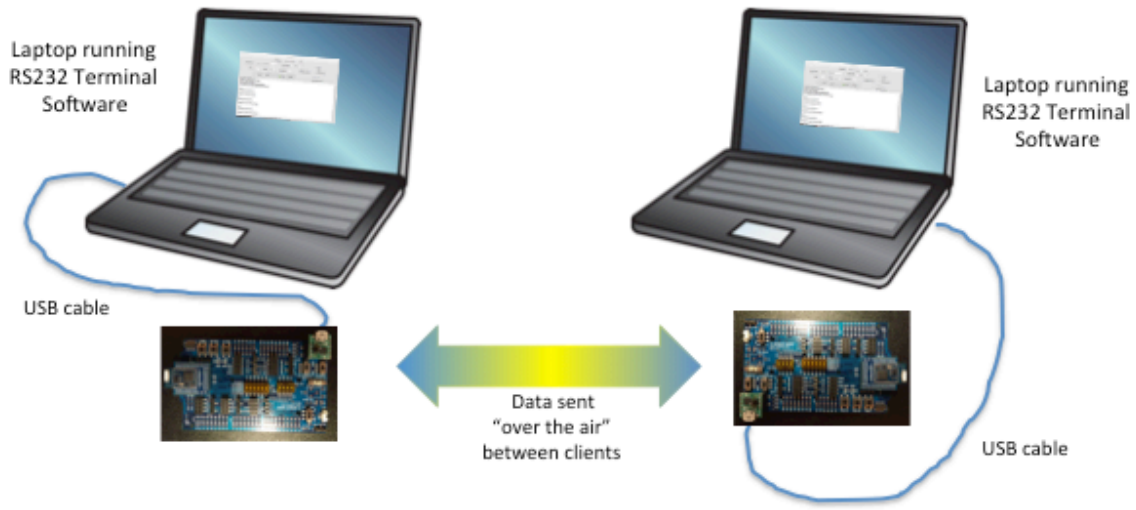
Disclaimer

The provided software files, including both hex files and source code files, are provided “as-is”, and without warranty. The software is provided only to help the user exercise the C-nRF52-SKY66112 hardware for evaluation purposes. Notwired.co assumes no obligation to update or maintain this software.

Test Setup

The basic test setup for range testing the C-nRF52-SKY66112 module is shown below. Laptop computers are shown, but any computer that has a USB port and can run RS232 Terminal emulation software will work.

Note that the two devices under test (C-nRF52-SKY66112) need to run different software clients, on a "central" client and one a "peripheral" client. The "central" client serves as a "bus master" for the BLE protocol. The software provided by Notwired.co includes hex files and source code for both the "central" and "peripheral" client.



The basic test method will be to establish a link between the clients at relatively short range, then confirm the link is working by sending text string messages to each other by typing a test message onto one RS232 terminal emulator and pressing "return". This will trigger the message to be sent "over the air" to the other unit, and in turn appear in the RS232 terminal emulation window on the PC connected to that unit. In this manner, messages can be sent back-and-forth, which confirms that the BLE link is working in both directions.

Gradually, the devices under test can be moved further and further apart, while sending messages back and forth to confirm link integrity. Eventually, the link will fail, and the distance between the units can be measured to determine the effective link range.

The link range will vary as the environment changes, and also by changing the mode of operation of the client modules. Different modes of operation will be outlined in the sections that follow. Environmental changes, such as walls, trees, hills, and similar obstructions will impact the range.

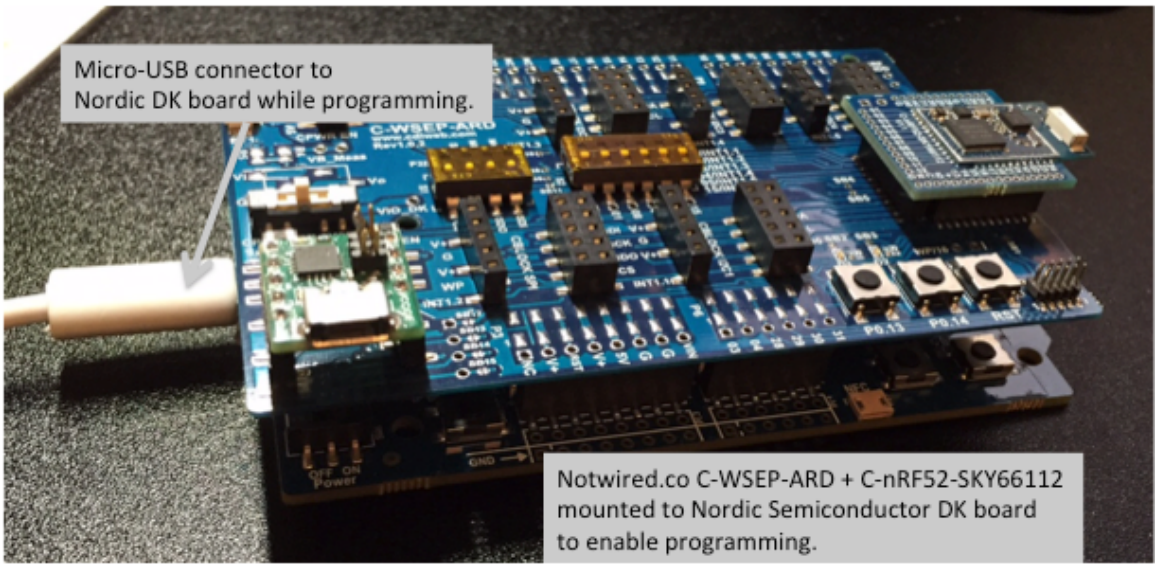
Using this method, Notwired.co has shown a BLE link can be achieved at a range of approximately 800 meters, when the SKY66112 FEM is in "High Power" transmit mode (more about modes later in this document). This range test was done over a large body of water, where interference and obstructions are minimized. It should not be taken as a "specification", but rather as an example of the range it may be possible to achieve in good conditions.

Programming Instructions

Programming each of the target devices follows the same process. The only difference is that the application code that must be loaded is different for the "central" client and the "peripheral" client. The hex files are named in such a way to make it easy to distinguish one from another.

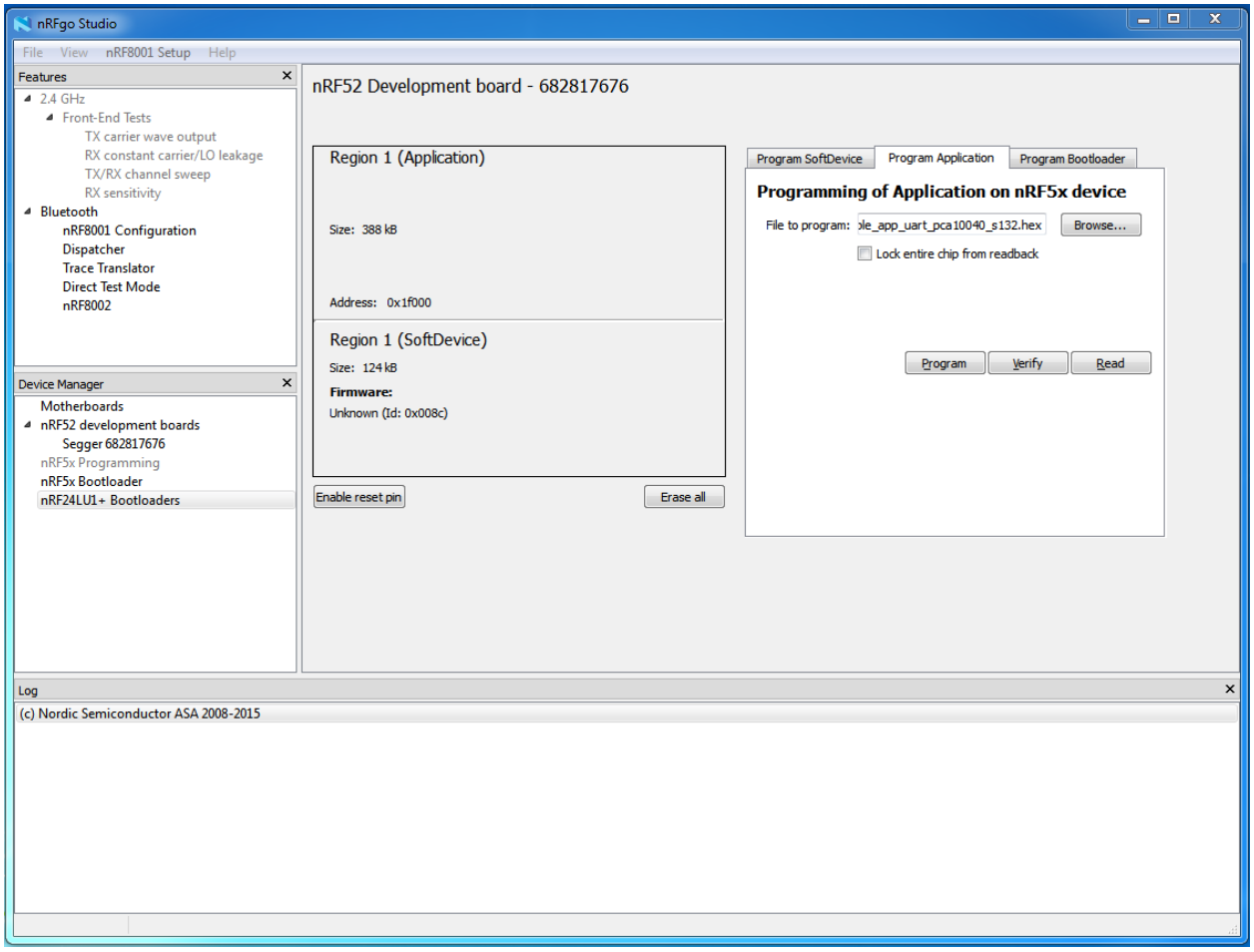
Programming steps:

1. Connect the C-WSEP-ARD board that contains the client C-nRF52-SKY66112 module to be programmed to the Nordic DK board. Connect the micro-USB connector on the Nordic DK board to the host PC, which is running the "nRFgo Studio" software. See image below.



2. Launch the "nRFgo Studio" software.
3. Along the left side of the nRFgo Studio software interface, in the lower pane labeled as "Device Manager", choose the Segger 682817676 (this number will

match the number on your Nordic nRF52 DK board) under “nRF52 development boards”. Your window should appear as shown below.



4. Press the “Erase all” button and wait for a confirmation in the Log window to show that the target device has been erased.

5. Choose the “Program SoftDevice” tab from the left side of the main window. Use the “Browse” button to navigate to the SoftDevice hex files (filename is: s132_nrf52_3.1.0_softdevice.hex). Then click on the “Program” button. After a short period of time, the Log window will indicate that the SoftDevice has been programmed correctly.

6. Now, choose the "Program Application" tab in the main window. Use the "Browse" button to select the correct hex file for either the central or the peripheral application. The hex file names are:

   - ble_app_uart_C_NRF52_Sky66112_central.hex
   - ble_app_uart_C_NRF52_Sky66112_peripheral.hex

   Click the "Program" button and wait for the confirmation to show in the Log window.

Assuming all has gone well, you have now programmed your device. You should disconnect the micro-USB from the Nordic DK board, and separate the C-WSEP-ARD board from the Nordic DK board before proceeding to the next step.

Remember, you must program one device with the "central" code and one with the "peripheral" code to allow them to connect to one another.

Using the Software

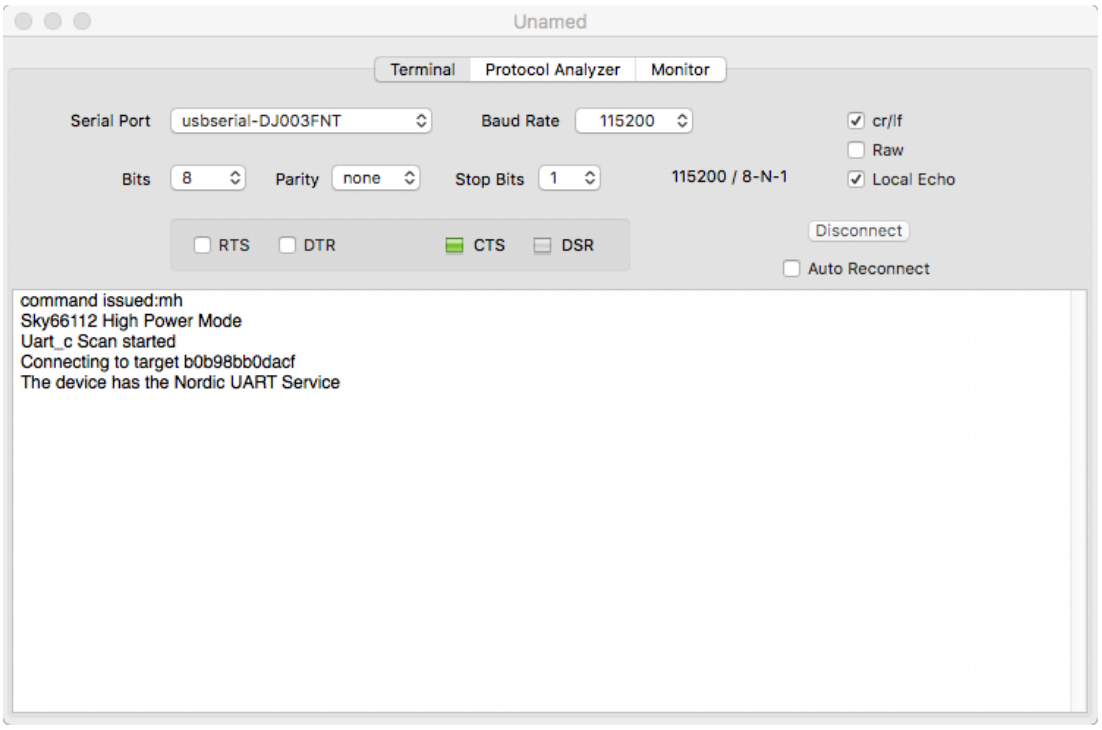
Configure the RS232 Terminal emulation program with the following parameters:

Serial Communication Settings:
Baud Rate = 115200 bps
Bits = 8
Party = none
Stop bits = 1
Flow Control is not important, hardware or "off" should both work

Terminal Settings:
Append a carriage return and line feed (CR+LF) to each message
Local Echo enabled

Once these settings are correct, connect the micro-USB connector from the EMCOMM board mounted to the C-WSEP-ARD board to the host PC. Using the Reset pushbutton on the C-WSEP-ARD (button is labeled "RST" on the board), reset the processor while monitoring the serial port.

For the "Central" device, you should see a message as below.



Here is a breakdown of the text sent by the device under test to the serial terminal upon reset:

“command issued:mh” indicates that a command has been received by the code, and the command was “:mh” (More about commands, below.)

“Sky66112 High Power Mode” indicates that the Skyworks FEM has been configured in high-power transmit / LNA receive mode. (More about this, below.)

“Uart_c Scan started” means that the device has started scanning for a device to pair with.

“Connecting to target b0b98bb0dacf” means that the device is scanning for a peripheral device loaded with a specific “personality”.

“The device has the Nordic UART Service” means that a peripheral device has been found and a connection has been established.

Obviously, this last line will not appear if you do not have a board with peripheral client software running. If you do have a peripheral device running, and this line does not appear, try resetting the peripheral device.

A “peripheral” client device will have a similar set of UART messages up to the “Sky66112 High Power Mode” message. After that, you will see a “UART Start!” message. No confirmation of a connection is provided on the peripheral software.

Once a connection has been made, the user should be able to type a test message into one terminal window, press <Return>, and the test message should be transmitted to the other unit, appearing in the target terminal emulation window.

Supported Commands

The SKY66112 FEM can be placed into several modes of operation. From the C-nRF52-SKY66112 documentation:

Four lines control the state of the Skyworks SKY66112-FEM. See the table below for more information. By manipulating these lines, the user has a wide range of control of the FEM, allowing for dynamic operation to maximize battery life while maintaining a stable RF connection.

Schematic Net Name	nRF SoC Pin	SKY66112-11 Pin	Description
P0.20/CTX	P0.20	CTX	Active high, used to set Transmit mode on
P0.19/CRX	P0.19	CRX	Active high, used to set Receive mode on
P0.06/CPS	P0.06	CPS	Active high, used to set TX or RX bypass mode on
P0.07/CHL	P0.07	CHL	Active high, used to set TX Power mode (Hi/Lo)

A more complete description of the functionality is provided in the following truth table:

Mode Control Logic Table

Mode of Operation	CPS	CRX	CTX	CHL
Receive LNA Mode	0	1	0	x
Transmit High-Power Mode	0	0	1	1
Transmit Low-Power Mode	0	0	1	0
Receive Bypass Mode	1	1	0	x
Transmit Bypass Mode	1	0	1	x
Sleep Mode	x	0	0	x

Note: "x" = don't care

For a fully detailed description of the function of various control lines as well as modes of operation, please refer to the SKY66112-11 datasheet.

The CTX and CRX pins are controlled automatically, in real-time by the BLE SoftDevice, but it is still possible to set three different modes of operation for the FEM using UART commands.

The supported modes of operation include:

Bypass Mode: Both TX and RX signals are connected through the FEM without the use of either a PA or LNA unit. The overall wireless link range is reduced in this mode, but power consumption is extremely low, as both the PA and LNA are "turned off".

TX Low Power Mode: In this mode, the PA has about 2 dB less gain, but the saturated output power of the PA remains about 20-21 dBm. This mode may be useful if the user finds that they are over-driving the SKY66112 FEM and causing interference via non-linearity as a result. Overall wireless link range may be reduced, and FEM power consumption will be reduced. Refer to the Skyworks datasheet for more information.

TX High Power Mode: This is the "default" mode for both the "central" and "peripheral" client. It will yield the longest wireless link range, at the cost of highest power consumption.

To change modes, the user can issue the following commands to the UART:

:mb = place the FEM in bypass mode

:ml = place the FEM in TX Low Power mode

:mh = place the FEM in TX High Power mode

Both the central and peripheral client will acknowledge receipt of a command with the response "command issued:mx", where x = b, l, or h respectively. The UART response will also indicate that the FEM has been placed into the appropriate mode.

Note that the UART command is also sent over-the-air to the paired device, so it will appear on the other PC RS232 terminal screen. This does not mean that the command has been issued to the paired device. Instead, the command must be issued to each device at the respective RF232 terminal UART interface.

Examples of the commands are responses are shown below: