Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

THIS SPEC IS OBSOLETE

Spec No: 001-79455

Spec Title: AN79455 - GETTING STARTED WITH WIRELESSUSB(TM) LP RADIO AND ENCORE(TM) V LV

Replaced by: NONE

AN79455

Getting Started with WirelessUSB™ LP Radio and enCoRe™ V LV

Author: Dikshak Pandya
Associated Project: Yes
Associated Part Family: CY3660 DVK, CY7C604XX, CYRF6936
Software Version: PSoC® Designer™ 5.2 SP1 or higher
Related Application Notes: For a complete list of the application notes, click here.

AN79455 demonstrates the use of the Cypress WirelessUSB™-LP 2.4 GHz Radio (CYRF6936) with an enCoRe™ V LV microcontroller device (CY7C604xx). It explains how to use PSoC® Designer™ 5.2 software to configure the ADC, LCD, and SPI User Modules with an enCoRe V LV device. Attached code examples demonstrate how to use the WirelessUSB™-LP Radio driver APIs and PSoC library APIs for the wireless data exchange application.

Contents

1 Introduction

In this application note, we use a CY3660 Development Kit (DVK) to demonstrate wireless data exchange between two enCoRe™ V LV devices each using the WirelessUSB™-LP Radio. The CY3660 DVK includes two enCoRe V LV development boards, each with a WirelessUSB-LP radio. With the PSoC® Designer™ projects attached to this application, we configure one board for data transmission and the second board for data reception.

The enCoRe V LV (Low Voltage) is a next generation low-power microcontroller (MCU) for wireless peripheral applications. The enCoRe V LV family offers a 10-bit ADC, three 16-bit timers, and up to 32 KB of flash memory to provide a highly integrated and cost-effective solution for wireless Human Interface Device (HID) applications. It is the ideal choice for sensor based low wireless applications and low power wireless peripheral applications.

The CYRF6936 WirelessUSB-LP radio is a second generation of the Cypress WirelessUSB Radio family. The radio chip contains a 2.4 GHz radio transceiver with following key features:

- 2.4 GHz Direct Sequence Spread Spectrum transceiver
- 21 mA operating current
- Transmit power up to +4 dBm
- Receive sensitivity up to –97 dBm

- Sleep current less than 1 µA
- Packet data buffering
- Packet framer
- Up to 4 MHz of Serial Peripheral Interface (SPI) for data transfer and device configuration
- Auto Transaction Sequencer (ATS) - no MCU intervention
- CRC (16 bit)
- Auto ACK
- Power Management Unit (PMU) for MCU/Sensor
- Fast Startup and Fast Channel Changes
- AutoRate™ - dynamic data rate reception
- Receive Signal Strength Indication (RSSI)

For more information, use the CYRF6936 Technical Reference Manual.

Note: WirelessUSB™-LP radio is Cypress proprietary radio technology different from WUSB protocol by USB Org.

2 WirelessUSB Resources

Cypress provides a wealth of data at www.cypress.com to help you to select the right WirelessUSB device for your design, and quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the wireless webpage.

- **Overview**: Wireless Roadmap , Modules Roadmap , Wireless Portfolio
- **Product Selectors**: Wireless Product selector
- **Datasheets**: Describe and provide electrical specifications for various device families. You can access the datasheet of all wireless products here.
- **Application Notes and Code Examples**:

  Cover a broad range of topics, from basic to advanced level. Many of the application notes include code examples. You can access the complete list of wireless AN here.
- **Technical Reference Manuals (TRM)**:

  Provide detailed descriptions of the architecture and registers in each WirelessUSB device family.

  You can access the complete list of wireless products TRM here.
- **Development Kits**:

  You can access the complete list of wireless kits and reference designs here.

Cypress offers ARM Cortex-M0 based, single-chip Bluetooth Low Energy (BLE) or Bluetooth Smart solutions. You can learn more about Cypress BLE devices here.

Cypress also has a similar radio CyFi™ Low-Power RF, which is targeted for star network implementation. You can find more details about CyFi here.

2.1 PSoC Designer

PSoC Designer is the revolutionary Integrated Design Environment (IDE) that you can use to customize PSoC to meet your specific application requirements. PSoC Designer software accelerates system bring-up and time-to-market. Develop your applications using a library of pre-characterized analog and digital peripherals in a drag-and-drop design environment. Then, customize your design leveraging the dynamically generated API libraries of code. Finally, debug and test your designs with the integrated debug environment including in-circuit emulation and standard software debug features.

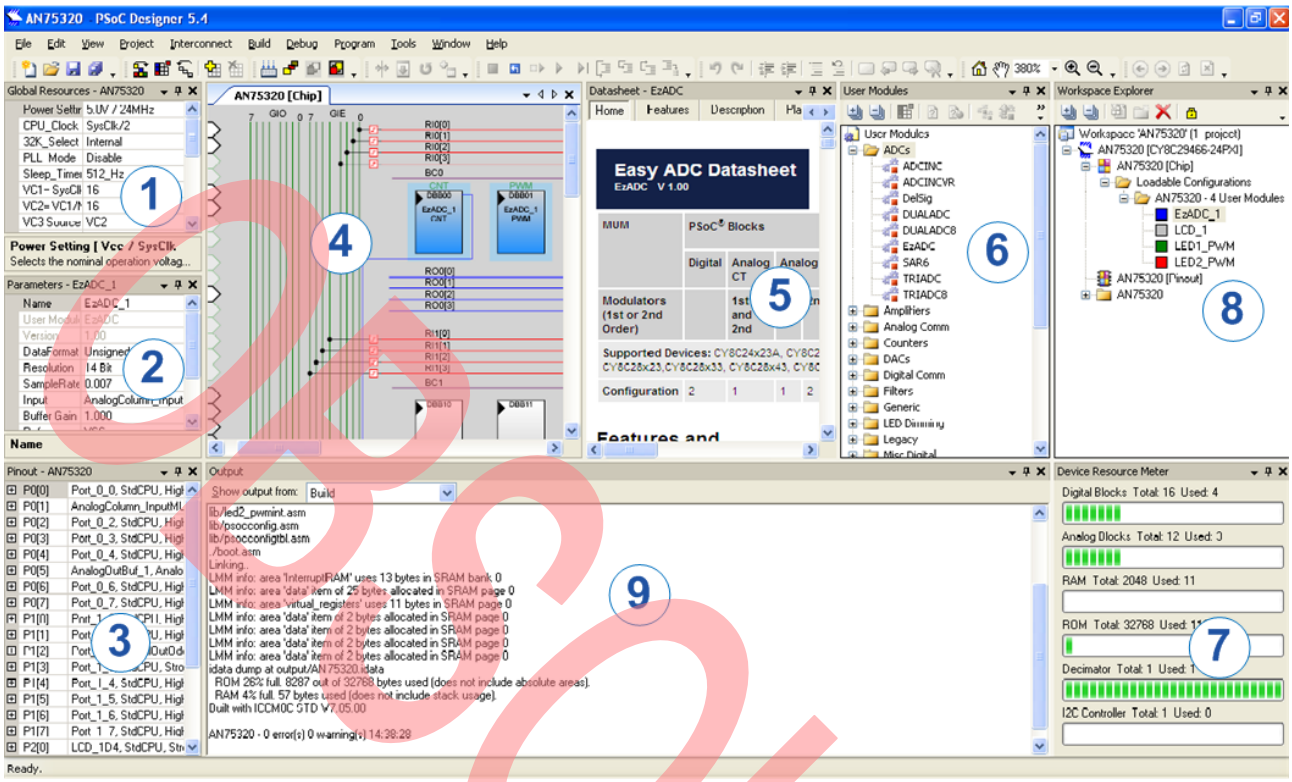
- Application Editor GUI for device and User Module configuration and dynamic reconfiguration
- Extensive User Module Catalog
- Integrated source code editor (C and Assembly)
- Free C compiler with no size restrictions or time limits
- Built-in Debugger
- Integrated Circuit Emulation (ICE)
- Built-in Support for Communication Interfaces:
  - Hardware and software I2C slaves and masters
  - Low/Full-speed USB 2.0
  - Up to 4 full-duplex UARTs, SPI master and slave, and Wireless

Figure 1 shows PSoC Designer windows. **Note:** This is not the default view.

1. **Global Resources –** all device hardware settings.
2. **Parameters –** the parameters of the currently selected User Modules.
3. **Pinout –** information related to device pins.
4. **Chip-Level Editor –** a diagram of the resources available on the selected chip.
5. **Datasheet –** the datasheet for the currently selected UM
6. **User Modules –** all available User Modules for the selected device.
7. **Device Resource Meter –** device resource usage for the current project configuration.
8. **Workspace –** a tree level diagram of files associated with the project.
9. **Output –** output from project build and debug operations.

Note: For detailed information on PSoC Designer, go to **PSoC® Designer** > **Help** > **Documentation** > **Designer Specific Documents** > **IDE User Guide**.

Figure 1. PSoC Designer Layout



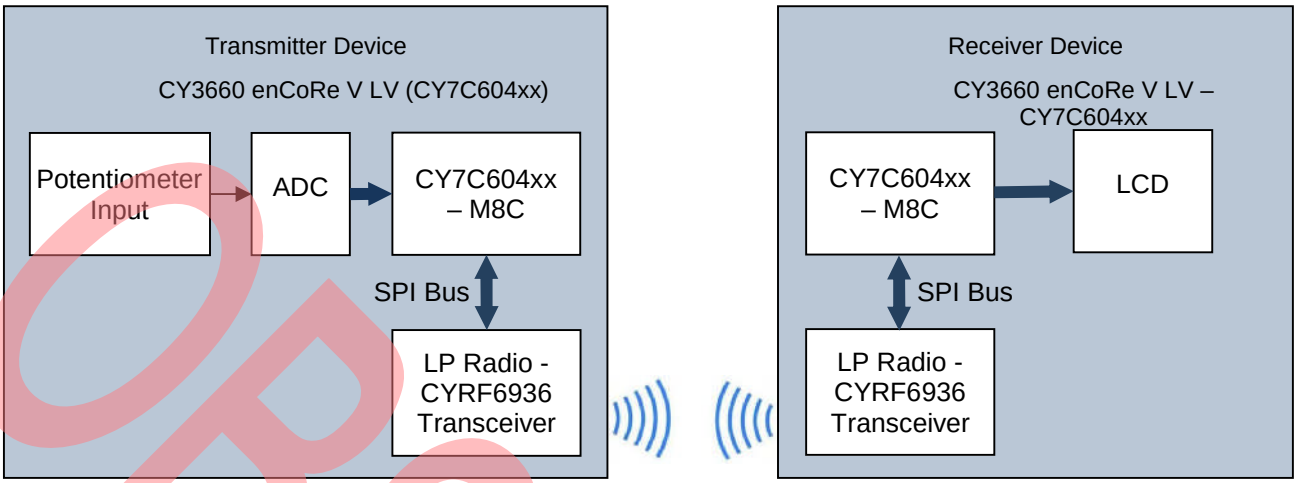
3 System Overview

In this project, we configure one board as the “transmitter device.” It uses the 10-bit ADC to continuously sample the onboard potentiometer input data and then periodically transmit sampled sensor data. The other board is the “receiver device.” The WirelessUSB-LP radio on the second board continuously listens for WirelessUSB-LP data packets over fixed channel of 2.4 GHz and then displays the received data on an LCD. The transmitter device uses the Incremental Analog to Digital Converter (ADC) and Serial Peripheral Interface (SPI) Master User Modules, while the receiver device uses the Liquid Crystal Display (LCD) and SPI Master User Modules.

Application firmware for both the transmitter and receiver devices uses the WirelessUSB-LP radio driver to configure and control radio operation.

The code example in this application note has two PSoC Designer projects:

Figure 2. System Level Block Diagram



- "Transmitter Device" project – Demonstrates wireless sensor data transmission using WirelessUSB™-LP radio with enCoRe V LV.
- "Receiver Device" project – Demonstrates wireless sensor data reception using WirelessUSB™-LP radio with enCoRe V LV.

Figure 2 provides a system level block diagram of the two projects used in this application note.

4 Transmitter Device

4.1 Project Name

"Transmitter Device" - Data Transmission Using WirelessUSB™-LP Radio and enCoRe V LV

4.2 Overview

We use PSoC Designer 5.2 to design and develop the enCoRe microcontroller applications. In this application project, the enCoRe V LV microcontroller uses the ADC User Module to sample onboard potentiometer data and configures the WirelessUSB™-LP radio to transmit the sampled data over a 2.4 GHz fixed frequency channel. Application firmware continuously transmits the sampled ADC data and indicates successful or unsuccessful data transmission status by a green LED, red LED blink respectively.

4.3 Hardware Setup

4.3.1 Requirements

- CY3660 DVK components
  - enCoRe V LV development board
  - 2.4 GHz wireless module using Cypress's WirelessUSB-LP radio
  - PSoC MiniProg
  - USB A/Mini-B Cable

4.3.2 Board Setup

- Jumper settings
  - Place a jumper on J1 and J12
  - Place a jumper between VDD and VREG on J2
  - Place a jumper J5 to select the fixed 3.3 V VREG
- Wire up P0[1] pin on connector P7 with VR pin on Connector P10 to configure Potentiometer input.

- Wire up P0[2], P0[0] pin on connector P7 with LED1, LED3 pin on Connector P10 as successful and unsuccessful transmission indications respectively.
- Mount the WirelessUSB-LP radio module on the LP radio connector P2 of the enCoRe V board.
- Board setup for programming the flash
  - Connect PSoC MiniProg to the 5-pin ISSP header J7 on the enCoRe V board.
  - Connect the USB cable to the PSoC MiniProg and to the PC.

Refer CY3660 kit user guide Appendix section for enCoRe V and LP radio schematic and board layout information.

Table 1. Pin Level Configuration

Pin	Name	Select	Drive	Interrupt
P0[1]	Pot_input	StdCPU	Open drain low	DisableInt
P0[2]	LED_indication	StdCPU	Strong	DisableInt
P0[0]	LED_indication	StdCPU	Strong	DisableInt
P1[5]	SPIM_Radio_MISO	SPIM MISO	Pull up	DisableInt
P1[3]	SPIM_Radio_SCK	SPIM CLK	Strong	DisableInt
P1[1]	SPIM_Radio_MISO	SPIM MOSI	Strong	DisableInt
P1[6]	LP_IRQ	StdCPU	Open drain low	DisableInt
P1[7]	LP_nSS	StdCPU	Strong	DisableInt

4.4 Software Setup

4.4.1 Tools Required

- PSoC Designer (version 5.2 or higher)
- PSoC Programmer (version 3.15.1 or higher)

CY3660 DVK CD includes PSoC Designer and PSoC Programmer software. Please install it, unless already installed.

4.4.2 Configure Transmitter Project

1. From the PC desktop, launch PSoC Designer 5.2. Open the **"Transmitter Device"** project. It is configured for enCoRe V LV MCU CY7C604xx.
2. From PSoC Designer, select **View** > **Chip Editor**.
3. Verify the User Modules parameter settings, pin connections and their placement as per the instructions in Table 1 and the User Module and Global Resources Settings section.
4. Select **Build** > **Generate Configuration Files** from PSoC Designer menu tab. PSoC Designer generates the configuration files for this project based on this action.
5. Select **Build** > **Build Transmitter Device** Project from PSoC Designer menu tab. You can also press [F7]. PSoC Designer builds the project.

4.4.3 User Modules

Table 2 lists the user modules required for the Transmitter Project:

Table 2. User Modules for Transmitter Project

User Module	Placement
SPI Master	In the fixed single I2C/SPI block
ADCINC	In the Systems Performance Controller (SPC) block (block does not appear in the Chip Editor)

4.4.4 User Module and Global Resources Settings

The following tables list the user module and global resource settings required for the Transmitter project:

Table 3. SPI User Module

Parameter	Value	Comments
Clock	SysClk/16	The SPI clock is SysClk/16. SysClk source is IMO of 24 MHz.
MISO	P1[5]	The MISO of the SPI block connects to enCoRe V via P1[5] pin.
MOSI	P1[1]	The MOSI of the SPI block connects to enCoRe V via P1[1] pin.
SCLK	P1[3]	The SCLK of the SPI block connects to the enCoRe V via P1[3].
Interrupt Mode	TxRegEmpty	The “TxRegEmpty” option causes the generation of an interrupt as soon as the data transfers from the Data register to the Shift register.
ClockSync	Sync to SysClk	The clock synchronizes with the system clock.
InvertMISO	Normal	Not applicable in this application

Table 4. ADC User Module

Parameter	Value	Comments
DataFormat	Unsigned	This selection determines the data format of the return result. It is configured as an unsigned entity.
Resolution	10-bit	This selection determines the size of returned result. It is configured for 10-bits data size.
DataClock	12 MHz	This selection determines the sample rate. This is configured to 12 MHz.
ClockPhase	Normal	The selection of the clock phase is used to synchronize between the PSoC blocks. It is configured as normal.
PosInput	Pot_input	The main input to the ADC connects to on board potentiometer through P0[1].
NegInput		This is kept blank as negative input is not used.
NegInputGain	Disconnected	Not applicable in this application
PulseWidth	1	Not applicable in this application
PWMOutput	None	Not applicable in this application

Operating voltage of the CYRF6936 device is from 1.8 to 3.6 V. You must restrict the SPI between the MCU and the radio to this voltage range. The rest of the GPIO pins are by default configured in High-Z state.

Table 5. Global Resource Settings for Transmitter Project

Parameter	Value	Comments
Power Settings	3.3 V/24 MHz	Selects 3.3-V operation
IMO Setting	24 MHz	Selects internal IMO clock to 24 MHz
CPU_Clock	12 MHz (SysClk/2)	Sets the CPU clock to 12 MHz
System Clock Source	Internal	Sets system clock source to internal clock source

5 Receiver Device

5.1 Project Name

“Receiver Device” - Data Reception Using WirelessUSB-LP Radio and enCoRe V LV

5.2 Overview

We use PSoC Designer 5.2 to design and develop the enCoRe microcontroller applications. In this application project, the firmware configures WirelessUSB-LP radio for a receive operation. The application continuously listens for the WirelessUSB data packets over a fixed frequency channel of 2.4 GHz ISM band, processes the received data packets, and then displays the received data on an LCD module.

5.3 Hardware Setup

5.3.1 Requirements

- CY3660 DVK components:
  - enCoRe V LV development board
  - 2.4-GHz wireless module using Cypress's WirelessUSB™-LP radio
  - PSoC MiniProg
  - 3.3-V LCD Module
  - USB A/Mini-B Cable

5.3.2 Board Setup

- Jumper settings
  - Place a jumper on J1 and J12.
  - Place a jumper between VDD and VREG on J2.
  - Place a jumper J5 to select the fixed 3.3 V VREG.
- Mount a 3.3-V LCD on connector P3 of the enCoRe V board.
- Mount the WirelessUSB-LP radio module on the LP radio connector P2 of enCoRe V board.
- Board setup for programming the flash:
  - Connect PSoC MiniProg to the 5-pin ISSP header J7 on the enCoRe V board.
  - Connect the USB cable to the PSoC MiniProg and to the PC.

Please refer CY3660 kit user guide Appendix section for enCoRe V and LP radio schematic and board layout information.

Table 6. Pin Level Configuration of Receiver Project

Pin	Name	Select	Drive	Interrupt
P1[5]	SPIM_Radio_MISO	SPIM MISO	Pull up	DisableInt
P1[3]	SPIM_Radio_SCK	SPIM CLK	Strong	DisableInt
P1[1]	SPIM_Radio_MISO	SPIM MOSI	Strong	DisableInt
P1[6]	LP_IRQ	StdCPU	Open drain low	DisableInt
P1[7]	LP_Nss	StdCPU	Strong	DisableInt

5.4 Software Setup

5.4.1 Tools Required

- PSoC Designer (version 5.2 SP1 or higher)
- PSoC Programmer (version 3.15.1 or higher)

CY3660 DVK CD includes PSoC Designer and PSoC Programmer software. Please install it, unless already installed.

5.4.2 Configure Receiver Project

1. From the PC desktop, launch PSoC Designer 5.2. Open the **"Receiver Device"** project.

It is configured for enCoRe V LV MCU CY7C604xx.

2. From PSoC Designer, select **View** > **Chip Editor**.
3. Verify the User Modules parameter settings, pin connections and their placement as per the instructions in Table 6 and User Module and Global Resources section.
4. Select **Build** > **Generate Configuration Files**.

   PSoC Designer generates the configuration files for this project.
5. Select **Build** > **Build "Receiver Device" Project**.

   You can also press [F7]. PSoC Designer builds the project.

5.4.3 User Modules

Table 7 lists the user modules required for the Receiver Project:

Table 7. User Modules for Receiver Project

User Module	Placement
SPI Master	In the fixed single I^2C/SPI block
LCD	Only uses seven I/O pins of one port and does not use any digital or analog blocks

5.4.4 User Module and Global Resources Settings

The following tables list the user module and global resource settings required for the Receiver project.

Table 8. SPI User Module

Parameter	Value	Comments
Clock	SysClk/16	The SPI clock is SysClk/16. The SysClk source is IMO of 24 MHz.
MISO	P1[5]	The MISO of the SPI block connects to the enCoRe V via the P1[5] pin.
MOSI	P1[1]	The MOSI of the SPI block connects to the enCoRe V via the P1[1] pin.
SClk	P1[3]	The SCLK of the SPI block connects to the enCoRe V via the P1[3].
Interrupt Mode	TXRegEmpty	The "TxRegEmpty" option causes the generation of an interrupt when the data transfers from the Data register to the Shift register.
ClockSync	Sync to SysClk	The clock is synchronized with the system clock.
InvertMISO	Normal	Not applicable in this application

The operating voltage of the CYRF6936 device is from 1.8 to 3.6 V. You must restrict the SPI between the MCU and the radio to this voltage range. The rest of the GPIO pins are by default configured in HI-Z state.

Table 9. LCD User Module

Parameter	Value	Comments
LCDPort	Port_3	Port 3 of the MCU interfaces with the LCD.
BarGraph	Disable	Not applicable in this application

Table 10. Global Resources for Receiver Project

Parameter	Value	Comments
Power Settings	3.3 V/24 MHz	Selects 3.3 V operation
IMO Setting	24 MHz	Selects internal IMO clock to 24 MHz

CPU_Clock	12 MHz (SysClk/2)	Sets the CPU clock to 12 MHz
System Clock Source	Internal	Sets system clock source to internal clock source

6 Operation

1. Prepare one enCoRe V LV board as the Transmitter Device and other as the Receiver Device by following corresponding PSoC Designer project Hardware Setup sections as described in this application note.
2. Mount the WirelessUSB-LP Radio to both enCoRe V LV boards.
3. Follow section 3.2 of CY3660 DVK guide to use PSoC Programmer to program the firmware on enCoRe V boards.
4. Program the Transmitter Device with corresponding **Transmitter Device** project binary.
5. Program the Receiver Device with corresponding **Receiver Device** project binary.
6. Power on both the enCoRe V LV boards.
7. Vary the potentiometer knob on the transmitter device. Use the LCD display of the receiver enCoRe V LV board to validate the value.

   The device uses 10-bit ADC to sample the potentiometer value. As a result, the output displayed on the receiver LCD should vary from 0 to 1024 based on the potentiometer setting on the transmitter device.

   The photos in Figure 3 and Figure 4 display the transmitter and receiver devices in operation.

Figure 3. Transmitter Device in Operation

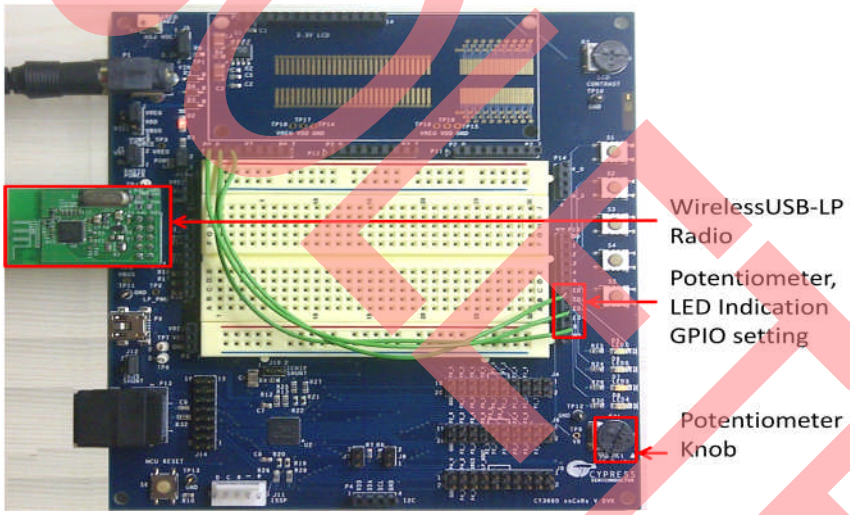
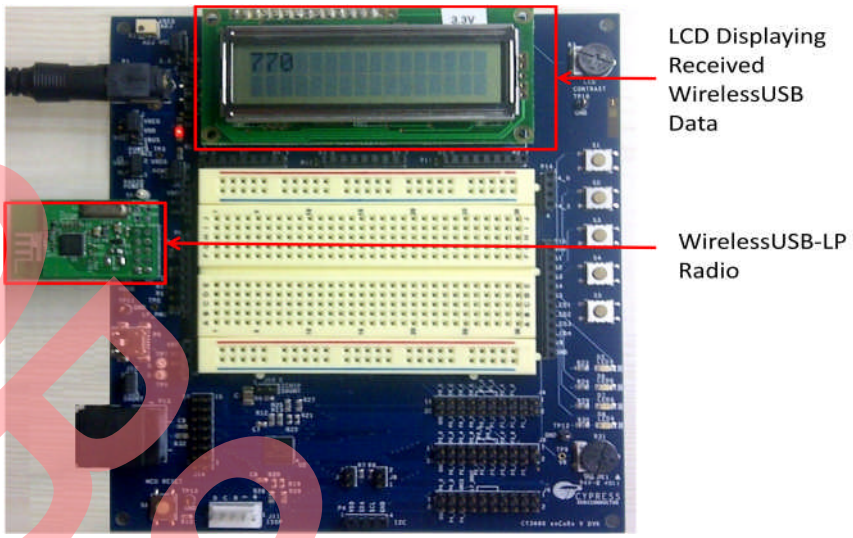


Figure 4. Receiver Device in Operation



7 Additional Resources

7.1 Terms and Abbreviations

- WirelessUSB-LP: A Cypress 2.4-GHz proprietary low-power radio technology
- enCoRe V LV: Enhanced component reduction 8-bit low-voltage MCU based on the PSoC 1 device family.
- ADC: Analog to digital converter
- SPI: Serial peripheral interface
- LCD: Liquid crystal display

7.2 Related Documentation

Related reference documents available from Cypress website are following:

- CY3660 Kit Guide
- WirelessUSB-LP Radio TRM
- WirelessUSB-LP Radio Datasheet
- WirelessUSB-LP Radio Driver
- CY7C604xx datasheet

7.3 Related Application Notes

- AN48610 - Design and Layout Guidelines for Matching Network and Antenna for WirelessUSB-LP Family
- AN15257 - Guidelines for Evaluating System Performance of Cypress WirelessUSB RF Products
- AN4003 - WirelessUSB 2-Way HID Systems
- AN43353 - Using enCoRe™ V 16-Bit Timer Modules as PWMs

8 Appendix A: Sample Code for Transmitter Device

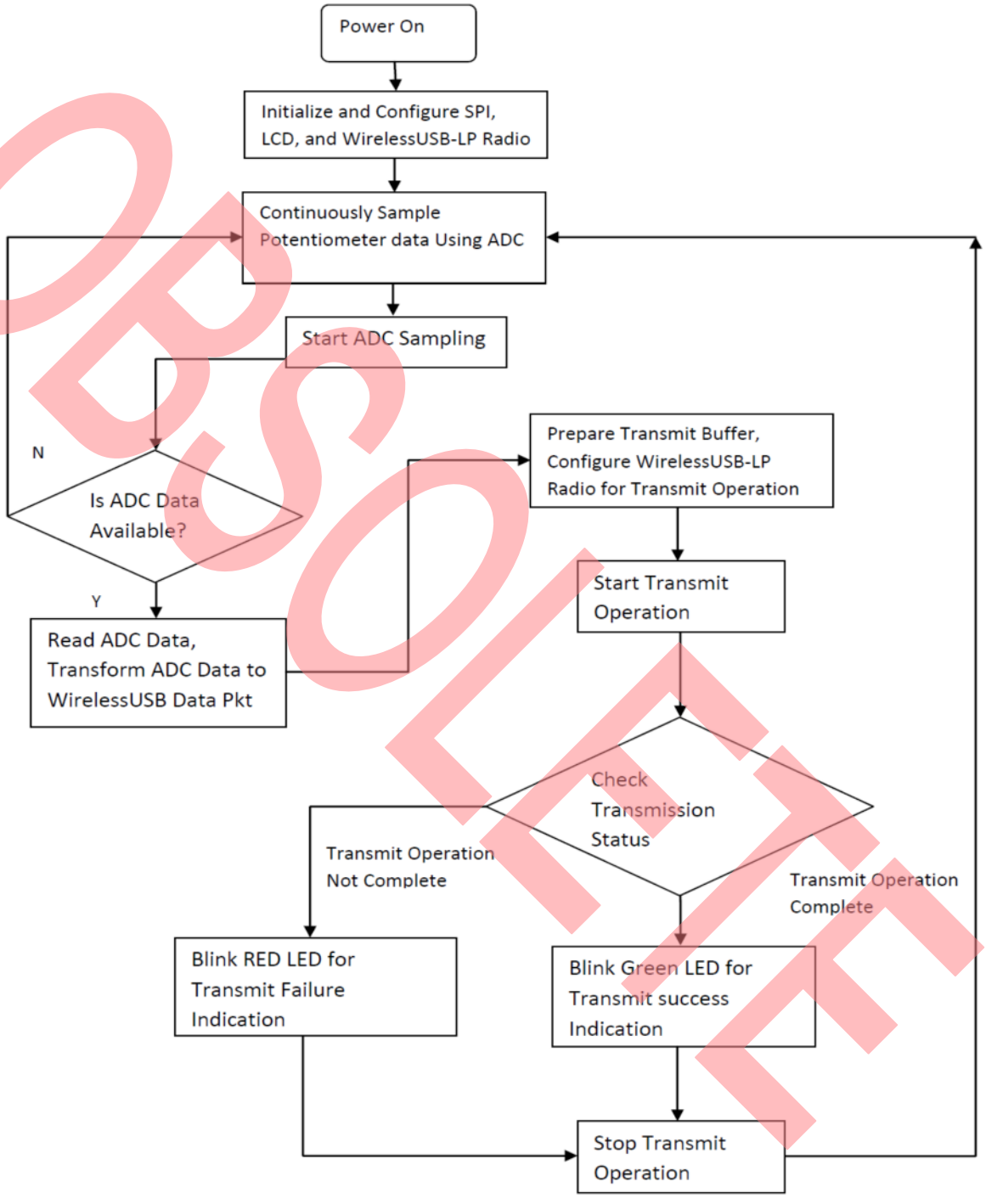
```
{
```

```
    // 1 .Initialize hardware
    // 2. SPI configuration for Radio interface
    HardwareInit();

    // 1. Initialise LP Radio
    // 2. Set the receive channel frequency
    // 3. Set the PN Code for the Start of Packet
    // 4. Set transmit packet length
    // 5. Register the transmit buffer
    InitialiseRadio();

    // Turn on global interrupts
    M8C_EnableGInt;

    // Application Logic
    while(TRUE)
    {
        // Apply power to the SC Block
        ADCINC_Start(ADCINC_INPUT_ANALOG_BUS);

        // Get sample from the ADC
        ADCINC_GetSamples(0);

        // Loop until ADC value ready to be read
        while(ADCINC_fIsDataAvailable() == 0);

        // Clear ADC flag and get data
        ADCValue = ADCINC_wGetData();

        // Stop the ADC
        ADCINC_Stop();

        // Convert the integer to ascii
        itoa(&TxPayloadBuffer[0],ADCValue,10);

        // Start the blocking transmit
        LpRadioState = RadioBlockingTransmit(2, PAYLOAD_LENGTH);

        // Check if Transmission is successfully complete
        if( (LpRadioState & (RADIO_ERROR | RADIO_COMPLETE)) == RADIO_COMPLETE)
        {}
        else
        {}

        // End the Radio transmission
        RadioEndTransmit();
    }
}
```

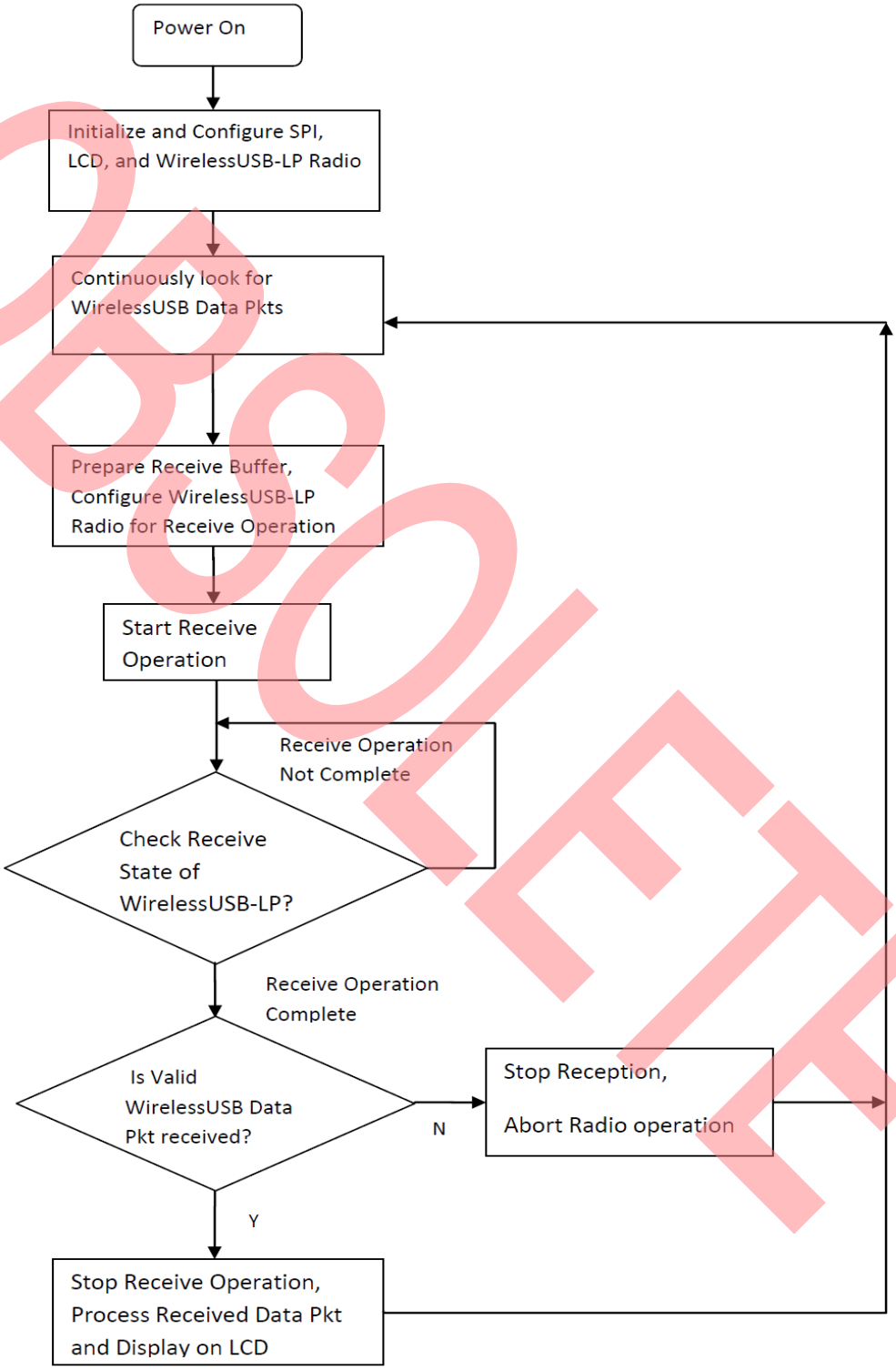
9 Appendix B: Sample Code for Receiver Device

```
{
    //1. Initialise hardware
    //2. SPI RF interface configuration
    //3. LCD module init/start
    HardwareInit();
    // 1. Initialise LP Radio
    // 2. Set the receive channel frequency
    // 3. Change the PN Code for the Start of Packet
    // 4. Set Receive packet length
    // 5. Set the receive buffer

    InitialiseRadio();

    // Turn on interrupts
    M8C_EnableGInt ;

    while(TRUE) // Application Logic
    {
        // Defining a variable to store the recieved payload size
        RADIO_LENGTH ReceivedPayloadSize;

        // Defining a variable to store the receive state of type RADIO_STATE
        RADIO_STATE  RxState;

        // Start the receiver
        RadioStartReceive();

        while(TRUE) // Receive Continuously
        {
            // Get the state of the receiver
            RxState = RadioGetReceiveState();

              // Check for Receive operation status
      if (RxState & RADIO_COMPLETE)
            {
                    // Receive Success
                if (!(RxState & RADIO_ERROR))
                {
                    //End reception. Returns the payload size
                    ReceivedPayloadSize = RadioEndReceive();

                    //Process the received data packets, display on LCD
                    ProcessReceivedPacket(…);
                 }
                 else{}      // Error in Pkt Reception, Abort & Terminate Rx
                 break; //Break the loop
             }
         }
     }
}
```

Appendix C: Functional Flow for Transmitter Device



Appendix D: Functional Flow for Receiver Device



Document History

Document Title: AN79455 - Getting Started with WirelessUSB™ LP Radio and enCoRe™ V LV

Document Number: 001-79455

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	3721638	DKSH	08/27/2012	New application note for Getting Started with WirelessUSB™-LP and enCoRe™ V LV
*A	3751055	DKSH	09/21/2012	Updated document as per review feedback.
*B	4809656	ANKC	07/08/2015	Updated in new template Added Resources section Sunset review
*C	5857304	ANKC	08/18/2017	Obsoleting the spec

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc
Memory	cypress.com/go/memory
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/usb
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Technical Support

cypress.com/go/support

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

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