

# AN1200.07 Application Note

## *RKE Demo with Encrypted Rolling Code*

Developed by **Leon Li**  
Embedded Software Engineer  
Semtech Semiconductor (Shenzhen) Co.,Ltd.  
Lli@semtech.com

## Table of Contents

Table of Contents.....	2
List of Figures .....	4
List of Tables.....	4
Legal Notice.....	5
1. Introduction.....	6
2. Quick Demo Guide .....	7
2.1. Transmitter Board.....	7
2.2. Receiver Board.....	7
2.3. Binding Procedure .....	7
2.4. Erasing the Receiver .....	7
3. Hardware Solution .....	8
3.1. Microcontroller Choice .....	8
3.2. Transmitter Design .....	9
3.2.1. Block Diagram .....	9
3.2.2. Pin Assignment.....	9
3.2.3. Physical Design .....	9
3.3. Receiver Design .....	10
3.3.1. Block Diagram .....	10
3.3.2. Pin Assignment.....	10
3.3.3. Physical Design .....	10
4. Software Solution.....	11
4.1. RF Protocol.....	11
4.1.1. Fixed Code vs. Rolling Code .....	11
4.1.2. RF Packet Format .....	11
4.1.3. RF Settings.....	14
4.2. Encryption Engine .....	14
4.2.1. Description .....	14
4.2.2. Encryption / Decryption Block Diagram .....	14
4.2.3. Encryption / Decryption Source Code.....	15
4.2.4. Encryption / Decryption Efficiency Improvement .....	15
4.2.5. Cipher Key Generation .....	15
4.3. Receiver's Operation Window .....	16
4.3.1. Payload Validation.....	16
4.3.2. Offset Calculation .....	16
4.3.3. Operation Windows .....	16
4.3.4. Vendor-Configurable Window Setting.....	17
4.3.5. Operations Window Source Code .....	17
4.4. Tx Firmware.....	17
4.4.1. Flowchart.....	18
4.4.2. Interrupts .....	18
4.4.3. EEPROM Map.....	19
4.5. Rx Firmware .....	20
4.5.1. Flowchart.....	20
4.5.2. Interrupts .....	20
4.5.3. EEPROM map.....	20

ADVANCED COMMUNICATIONS & SENSING		APPLICATION NOTE
4.6.	Binding Procedure .....	21
4.7.	Rx Output Types.....	22
4.8.	Power Management.....	23
5.	Appendixes .....	24
5.1.	Microchip References .....	24
5.2.	Encryption / Decryption Source Code.....	24
5.3.	Operation Window Code.....	24
5.4.	Tx Board Schematics.....	25
5.5.	Rx Board Schematics .....	26
5.6.	Bills Of Materials.....	27
5.7.	Boards Pictures .....	29
5.8.	Tx 433MHz PCB Antenna Reference Design .....	29
5.9.	Boards Layout .....	30

## List of Figures

Figure 1: RKE System .....	8
Figure 2: Tx Block Diagram .....	9
Figure 3: Rx Block Diagram .....	10
Figure 4: RF Packet Structure.....	11
Figure 5: Encryption System .....	14
Figure 6: Decryption System.....	14
Figure 7: Operation Windows Description.....	16
Figure 8: Tx Main Loop Flowchart.....	18
Figure 9: Rx Main Loop Flowchart .....	20
Figure 11: Tx Board Schematics.....	25
Figure 12: Rx Board Schematics .....	26
Figure 13: Rx and Tx Boards Pictures .....	29
Figure 14: 433 MHz Helical Antenna .....	29
Figure 15: Rx Board Layout .....	30
Figure 16: Tx Board Layout .....	30

## List of Tables

Table 1: MCU Choice.....	8
Table 2: Tx MCU Pin Assignment .....	9
Table 3: Rx MCU Pin Assignment.....	10
Table 4: Button Packet Payload - All-Encrypted, Binding .....	12
Table 5: Button Packet Payload - Plain-Index, Binding.....	12
Table 6: Bind Packet Payload - All-Encrypted, Binding.....	13
Table 7: Bind Packet Payload - Plain-Index, Binding .....	13
Table 8: RF Settings .....	14
Table 9: Window Slots Build Conditionals.....	17
Table 10: Tx EEPROM Map.....	19
Table 11: Rx EEPROM Map .....	21
Table 12: SX1211/12 Power Specification .....	23
Table 13: PIC16F631 Power Specification .....	23
Table 14: Tx Board BOM .....	27
Table 15: Rx Board BOM.....	28

## Legal Notice

THE SOFTWARE IS BASED ON THIRD-PARTY SOURCE CODE AVAILABLE FROM <http://www.microchip.com/keelog/> UPON EXECUTION OF A LICENSE AGREEMENT. ACCORDINGLY, SEMTECH MAKES NO REPRESENTATION OR WARRANTY WITH RESPECT TO THE SOFTWARE OR DOCUMENTATION AND YOU SHALL RECEIVE THE LICENSES AND RIGHTS GRANTED HEREUNDER "AS IS". SEMTECH DOES NOT REPRESENT OR WARRANT THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS OR IS FREE FROM ERRORS, THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR FREE OR THAT ANY ERRORS IN THE SOFTWARE CAN BE REMEDIATED. ANY IMPLIED WARRANTIES, INCLUDING ANY WARRANTY OF MERCHANTABILITY, TITLE, FITNESS FOR A PARTICULAR PURPOSE AND INFRINGEMENT OF THIRD PARTY RIGHTS ARE EXPRESSLY EXCLUDED. ANY IMPLIED INDEMNIFICATIONS, INCLUDING ANY INDEMNIFICATION BASED ON INFRINGEMENT OF THIRD PARTY'S RIGHTS ARE EXPRESSLY EXCLUDED. THE SOFTWARE LICENCE DOES NOT APPLY TO OPEN SOURCE. OPEN SOURCE IS LICENSED SEPARATELY PURSUANT TO THE TERMS PROVIDED IN THIS ATTACHMENT. YOU AGREE TO BE BOUND BY THE TERMS OF THE ATTACHED OPEN SOURCE LICENCE WHEN YOU AGREE TO THE TERMS OF THE SOFTWARE LICENSE.

PIC is a trademark of Microchip Technology Incorporated in the USA and other countries.

## 1. Introduction

A remote keyless system (RKS) is a system designed to remotely permit or deny access to premises or automobiles.

In the case of automobiles an RKS performs the functions of a standard car key without physical contact; power door systems can be locked or unlocked from several feet away or even from inside a building.

A remote keyless system can include both a remote keyless entry system (RKE) and a remote keyless ignition system (RKI).

Remote keyless systems first began appearing as an option on several American Motors vehicles in 1983. Most RKE systems operate at 315MHz in North America and Japan, and 433.92 MHz in Europe. Modern systems implement encryption to prevent car thieves from intercepting and spoofing the signal.

The functions of a remote keyless entry system are contained on a key fob or built into the ignition key handle itself. Buttons are dedicated to locking or unlocking the doors and opening the trunk (or, on sport utility vehicles and station wagons, unlock/open the rear tailgate). Some cars will also close any open windows and roof when remotely locking the car. Some remote keyless fobs also feature a red panic button which activates the car alarm as a standard feature.

Some car engines with remote keyless ignition systems can be started by the push of a button on the key fob.

For offices or residences, the system can also be coupled with the security system, garage door opener or remotely activated lighting devices.

## 2. Quick Demo Guide

The demo kit running the software described in this application note is very easy to use. It allows the user to quickly evaluate the solution provided by Semtech.

### 2.1. Transmitter Board

4 push buttons are named SW1, SW2, SW3 and SW4. Each of them corresponds to a unique Button Payload, and they activate the matching LEDs D1 to D4 on the Receiver (Rx) board.

### 2.2. Receiver Board

4 LEDs are labeled D1 to D4. They are activated by the corresponding keys on the Transmitter (Tx) board. The LEDs can be used in Interlock, Pulse or Toggle mode, please see section 4.7 for details.

### 2.3. Binding Procedure

To Bind a Tx board with a Rx board:

- Hold the “Bind” key on the Rx until the “Bind” LED lights on
- Within 10 seconds, press SW1 and SW4 together on the Tx board
- Binding success is acknowledged if the Bind LED blinks 3 times
- The Bind LED will simply turn off if binding fails

Refer to section 4.6 for details on the Binding procedure. Note that the “Binding” procedure is also known as “learning” in the usual Microchip nomenclature.

### 2.4. Erasing the Receiver

To erase all Tx information in a Rx board, hold the “Bind” key for more than 10 seconds. The “bind” LED will blink 10 times if the procedure goes through. A more detailed explanation is provided in section 4.6.

### 3. Hardware Solution

The solution described in this technical note is based on Semtech's transmitter SX1230 (Tx side) and transceivers SX1211 or SX1212 (Rx side). These combinations of products make the platform suitable for the 868MHz, 915MHz or 433MHz ISM bands.

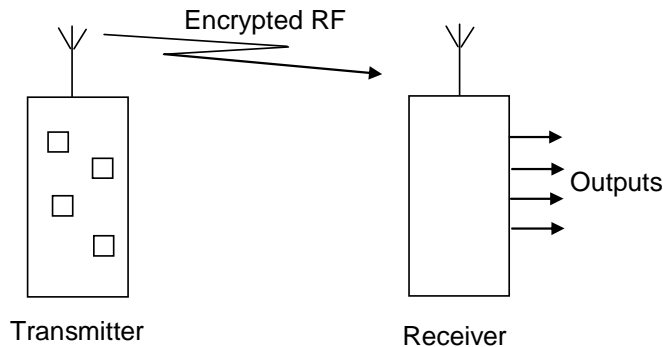


Figure 1: RKE System

Details on the SX1211, SX1212 and SX1230 chips can be downloaded from the Semtech website, at the following addresses:

- <http://www.semtech.com/products/sx1211/>
- <http://www.semtech.com/products/sx1212/>
- <http://www.semtech.com/products/sx1230/>

#### 3.1. Microcontroller Choice

The following Microchip devices have been selected to run the RKE demo:

Table 1: MCU Choice

Device	Program memory	Data memory		I/O	10-bit A/D (ch)	Comparators	Timers 8/16 bits	SSP (SPI)
	Flash (words)	SRAM (bytes)	EEPROM (bytes)					
PIC16F631	1024	64	128	18	-	2	1/1	No
PIC16F677	2047	128	256	18	12	2	1/1	Yes

Note: PIC16F677 and PIC16F631 are compatible.

Detailed specification of those products can be downloaded from the Microchip website: [www.microchip.com](http://www.microchip.com)

## 3.2. Transmitter Design

### 3.2.1. Block Diagram

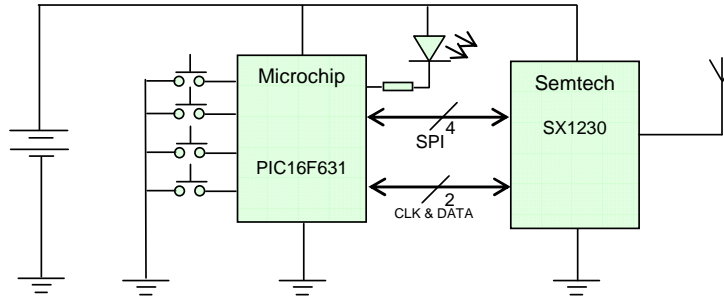


Figure 2: Tx Block Diagram

### 3.2.2. Pin Assignment

Table 2: Tx MCU Pin Assignment

Pin number	Pin name	Pin function	Type	Description
1	VDD	VDD	Power	VDD
2	RA5	MISO	I	SPI bus: Master In Slave Out
3	RA4	DCLK	I	Clock for RF packet bit shifting provided by the Tx RF chip
4	/MCLR (RA3)	Reset	I	Power on Reset
5	RC5	MOSI	O	SPI bus Master Out Slave In
6	RC4	SCK	I	SPI bus clock
7	RC3	NC	-	Not used
8	RC6	DATA	O	RF data bits to the Tx RF chip
9	RC7	NSS	O	SPI chip select
10	RB7	SW1	I	Remote control button
11	RB6	SW2	I	Remote control button
12	RB5	SW3	I	Remote control button
13	RB4	SW4	I	Remote control button
14	RC2	NC	-	Not used
15	RC1	NC	-	Not used
16	RC0	NC	-	Not used
17	RA2	LED	O	Sent LED
18	RA1	NC	-	Not used
19	RA0	NC	-	Not used
20	VSS	VSS	Ground	GND

### 3.2.3. Physical Design

All design files for the Tx design are available in the Appendix:

- Schematics, Bill Of Materials
- Layout
- Module Picture

### 3.3. Receiver Design

#### 3.3.1. Block Diagram

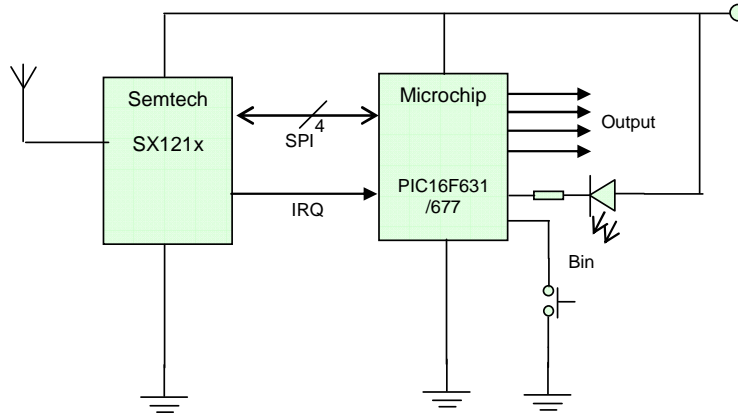


Figure 3: Rx Block Diagram

#### 3.3.2. Pin Assignment

Table 3: Rx MCU Pin Assignment

Pin number	Pin name	Pin function	Type	Description
1	VDD	VDD	Power	VDD
2	RA5	MISO	I	SPI bus Master In Slave Out
3	RA4	IICDAT	O	Ext. EEPROM IIC data*
4	RA3(/MCLR)	BIND	I	Bind button
5	RC5	MOSI	O	SPI bus Master Out Slave In
6	RC4	SCK	O	SPI bus clock
7	RC3	NC	-	Not used
8	RC6	NSS_DATA	O	Select DATA SPI registers
9	RC7	NSS_CFG	O	Select CONFIG SPI registers
10	RB7	Output 1	O	Decoded output for Tx button 1
11	RB6	Output 2	O	Decoded output for Tx button 2
12	RB5	Output 3	O	Decoded output for Tx button 3
13	RB4	Output 4	O	Decoded output for Tx button 4
14	RC2	IICWP	O	Ext. EEPROM write protect*
15	RC1	IICSCL	O	Ext. EEPROM IIC clock*
16	RC0	BIND_LED	O	Bind LED
17	INT(RA2)	IRQ_0	I	RF chip "Payload_ready" IRQ
18	RA1	NC	-	Not used
19	RA0	NC	-	Not used
20	VSS	VSS	Ground	GND

(\*: For "Plain-Index, Binding" protocol)

#### 3.3.3. Physical Design

All design files for the Rx design are available in the Appendix.

## 4. Software Solution

### 4.1. RF Protocol

This section gives details on the structure of the RF packet. It refers to the encryption method, thoroughly detailed in section 4.2.

#### 4.1.1. Fixed Code vs. Rolling Code

In early generations of RKE systems, “Fixed Code” was used for packet encryption. Usually a fixed 8-bit address was used to identify and bind a single pair of transmitter and receiver. This means that the exact same message was being exchanged between the Tx and the Rx every time

This RF packet can very simply be intercepted, analyzed and repeated to pass the authentication process at Rx and open the vehicle. This hacking technique is commonly referred to as “relay attack”, or “replay attack”. It would also be quick and easy, once the packet structure is identified, to sweep all of the 256 addresses and get access to the vehicle.

A “Rolling Code” mechanism can solve the above shortcoming of “Fixed Code”. In a rolling code system, the RF packet not only contains the Device ID, button status and checksum, but also embeds a Synchronize Counter, incremented by 1 on every transmission. The receiver will check all the information contained in the received RF packet. Not only must the Device ID and checksum be correct, but the Synchronize Counter value must also be correct. In practice, it must be strictly greater than the previous recorded value (within a pre-defined range, defined in the Operation Window, refer to section 4.3 for details). A Receiver with Rolling Code will be insensitive to replay attack, as it will discard two incoming packets with identical Synchronize Counter values.

In addition to this feature, the Payload is fully encrypted to prevent potential hackers to analyze the packets, and deduce what the next packet may be.

#### 4.1.2. RF Packet Format

The RF packet is built as follows:

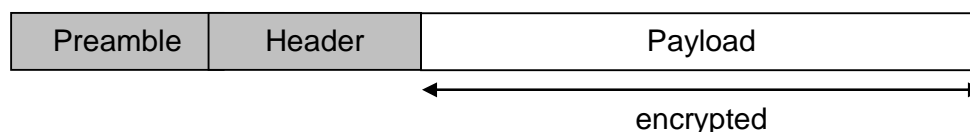


Figure 4: RF Packet Structure

Note: Data is sent MSB first.

##### ➤ Preamble

The preamble is a series of “01010101” with 50% duty cycle. The current implementation uses 3 bytes of preamble.

➤ Header

The header is treated as a Network ID. The Receiver will accept the payload if the header meets a pre-defined pattern. This sorting operation is performed by the Sync Word recognition block of the SX1211/12. The current implementation uses 4 bytes of header, with error tolerance set to 0 bit.

➤ Button Packet Payload

The Transmitter will send the Button Packet payload, encrypted, when one of the buttons is pressed. The current RKE implementation has two options: “All-Encrypted, Binding”, or “Plain-Index, Binding”. Preamble and Header are the same in the two implementations, but the Button Packet and Bind Packet payloads differ:

The “All-Encrypted, Binding” protocol supports up to 16 Tx units bound to a single Receiver with the PIC16F631, or 32 Tx units when the MCU is a PIC16F677.

The “Plain-Index, Binding” protocol supports up to 800 Tx units bound to a single Receiver, regardless of the MCU. An external EEPROM is however required to store the Tx information for this protocol.

- All-Encrypted, Binding  
There is a total of 64 bits (8 bytes) in the Button Packet. They are divided into two 4-byte blocks, individually encrypted.

**Table 4: Button Packet Payload - All-Encrypted, Binding**

Block	Byte Offset	Name	Description
1	0	XOR_CHKSM	XOR checksum of bytes 1 to 7
	1	Button	Button status. 0 = pressed
	2	Device_ID[0]	Device ID [0] is the LSB
	3	Device_ID[1]	
4	Device_ID[2]		
5	Device_ID[3]		
0	6	Sync_CounterLSB	Synchronize Counter
	7	Sync_CounterMSB	

- Plain-Index, Binding  
There is a total of 80 bits (10 bytes) in the Button Packet. The first 8 bytes are divided into 2 blocks for individual encryption. The last 2 bytes are Index values, which are not encrypted. Index value is used to specify which EEPROM location should be used to store the information related to this specific Transmitter, in the Receiver it is bound with.

**Table 5: Button Packet Payload - Plain-Index, Binding**

Block	Byte Offset	Name	Description
1	0	XOR_CHKSM	XOR checksum of bytes 1 to 9

ADVANCED COMMUNICATIONS & SENSING APPLICATION NOTE			
	1	Button	Button status. 0 = pressed
	2	Device_ID[0]	Device ID [0] is the LSB
	3	Device_ID[1]	
4	Device_ID[2]		
5	Device_ID[3]		
0	6	Sync_CounterLSB	Synchronize Counter
	7	Sync_CounterMSB	
N/A	8	Index_LSB	Index value LSB
	9	Index_MSB	Index value MSB

➤ Bind Packet Payload

When Button1 and Button4 are pressed simultaneously, the Tx sends a Bind Packet. Bind Packets are never encrypted. They are sent in a low power mode (-18dBm) to reduce the chances of being intercepted.

- All-Encrypted, Binding  
There is a total of 64 bits (8 bytes) in the Bind Packet. Since it is not encrypted, the Bind Packet doesn't need to be divided into blocks.

**Table 6: Bind Packet Payload - All-Encrypted, Binding**

Byte Offset	Name	Description
0	XOR_CHKSM	XOR checksum of bytes 1 to 7
1	Reserved	Must be 0x00
2	Device_ID[0]	Device ID [0] is the LSB
3	Device_ID[1]	
4	Device_ID[2]	
5	Device_ID[3]	
6	Sync_CounterLSB	Current value of the Synchronize Counter
7	Sync_CounterMSB	

- Plain-Index, Binding  
There is a total of 80 bits (10 bytes) in the Bind Packet. Since it is not encrypted, Bind Packet doesn't need to be divided into blocks.

**Table 7: Bind Packet Payload - Plain-Index, Binding**

Byte Offset	Name	Description
0	XOR_CHKSM	XOR checksum of bytes 1 to 9
1	Reserved	Must be 0x00
2	Device_ID[0]	Device ID [0] is the LSB
3	Device_ID[1]	
4	Device_ID[2]	
5	Device_ID[3]	
6	Sync_CounterLSB	Current value of the Synchronize Counter
7	Sync_CounterMSB	
8	Index_LSB	Index value LSB
9	Index_MSB	Index value MSB

### 4.1.3. RF Settings

The following table summarizes the basic RF settings programmed on the SX1211, SX1212 or SX1230.

Table 8: RF Settings

Item	Value	Unit	Description
-	FSK	-	RF Modulation mode
$f_{RF}$	434 868 915	MHz	Center frequency SX1211 addresses 868 and 915 MHz SX1212 addresses 434 MHz (same PCB)
-	+13	dBm	Tx RF output power
-	-18	dBm	Bind packet RF power
BRF	25	kbps	Bit Rate
FDA	50	kHz	Frequency Deviation
-	3	Bytes	Packet Preamble (0x555555)
Sync_size	4	Bytes	Packet Header (Sync Word) size. ("ROLL" → 0x52,0x4F,0x4C,0x4C)
Sync_tol	0	bit	Number of errors tolerated in the Sync Word

## 4.2. Encryption Engine

### 4.2.1. Description

The RF payload in this RKE system is encrypted for security reasons. The encryption algorithm we use can encrypt a 32-bit plain text with a 64-bit cipher key. It outputs is a 32-bit cipher text. This method is a proprietary hardware-dedicated NLFSR-based block cipher.

The encoder encrypts a 0-filled 32-bit block with cipher to produce a 32-bit "hopping code". This encoder guarantees that at least 50% of the bits are modified, when a single bit is changed in the plain text being encrypted.

### 4.2.2. Encryption / Decryption Block Diagram

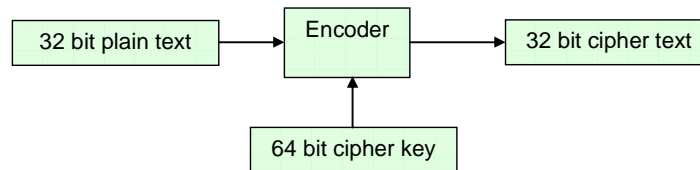


Figure 5: Encryption System

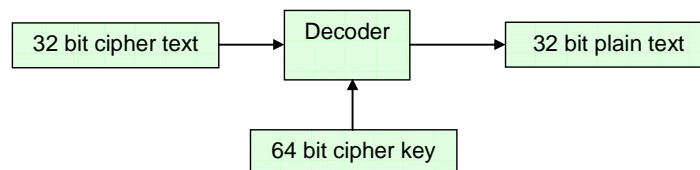


Figure 6: Decryption System

### 4.2.3. Encryption / Decryption Source Code

The source code for the encryption and the decryption engine implemented in the Semtech RKE reference design can be downloaded from the Microchip website. The user should accept the Microchip license agreement.

Note that the encoder / decoder source code is only distributed in PIC assembly.

### 4.2.4. Encryption / Decryption Efficiency Improvement

Since most C compilers do not support 64-bit data type natively, the union “\_U64” is used as a 64-bit data type. The routine “\_U8 bit64 (\_U64 x, \_U8 n)” handles the 64-bit data shift operation. This standard C code may not be the most efficient in terms of code space and speed. For code size or speed critical applications, assembly code dedicated to the specific host architecture can be developed.

### 4.2.5. Cipher Key Generation

The Cipher key, used to encrypt data, is assigned as follows:

**Cipher Key (high 32 bits) = Device ID (32 bits) XOR Factory ID (high 32 bits)**  
**Cipher Key (low 32 bits) = Device ID (32 bits) XOR Factory ID (low 32 bits)**

Where:

- Factory ID: 8-byte value. Each vendor has its individual Factory ID, used in all its Rx and Tx systems. The Factory ID is stored in the memory of the microcontroller, and is never sent over the air. It is not recommended to disclose the Factory ID to unrelated parties.
- Device ID: 4-byte value. Each Transmitter has its individual Device ID, Assigned in the factory when the processor is being programmed. The Device ID will be sent in plain text format when sending a Bind Packet, and encrypted in the Cipher when sending a Button Packet. The device ID is stored in the EEPROM of the Transmitter device. Receivers don't have any Device ID. They “learn” the Device ID of the Transmitters they are bound with during the Biding process.
- Cipher Key: 8-byte value. On the transmitter side, the Encoder encrypts the Button Packet using the cipher key. On the Receiver side, the Decoder decrypts the Button Packet using the cipher key.

Note: Usually, a blank EEPROM will be filled with 0x00 or 0xFF bytes. For security reasons, the Device IDs 0x00000000 and 0xFFFFFFFF are defined as illegal and will be rejected by a Receiver.

### 4.3. Receiver's Operation Window

#### 4.3.1. Payload Validation

When the Receiver detects a Packet with a valid Network ID (aka Header, set to the ASCII string "ROLL" or 0x524F4C4C in the current implementation), the Payload will be stored in the receiver FIFO. The microcontroller on the Receiver will be informed by an IRQ, and will retrieve the FIFO contents through the SPI bus. More verification is performed:

- 4-byte Device ID
- 1-byte XOR checksum
- 2-byte Synchronize Counter

The Payload will be accepted only if all of the aforementioned validations pass.

#### 4.3.2. Offset Calculation

The Synchronization Counter value is extracted from the Payload and the Offset is calculated as follows:

$$\text{offset} = \text{Synchronize Counter from Tx(16 bit)} - \text{Synchronized Counter stored in Rx(16 bit)}$$

(Note: all parameters are unsigned)

Depending on the Offset value, three different Operation Windows are defined:

#### 4.3.3. Operation Windows

The Synchronization Counter is coded over 2 Bytes, defining up to 64k slots in the Operation Window:

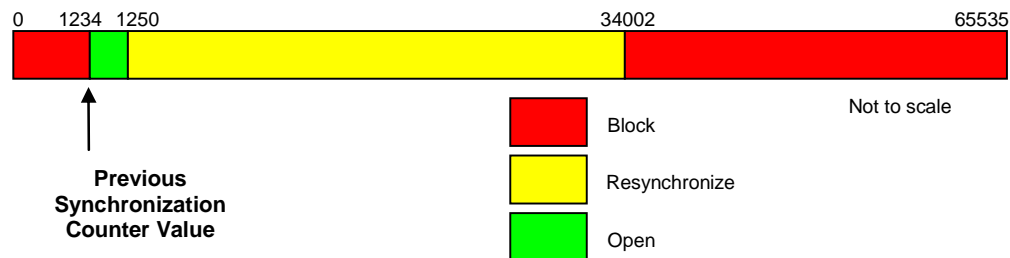


Figure 7: Operation Windows Description

- Open Window:  $1 \leq \text{Offset} \leq 16$

The Receiver will update its previous counter value with the new one, and execute the action specified by the button pressed. The RKE system is meant to work under this scenario most of the time.

- Re-Synchronize Window:  $17 \leq \text{Offset} \leq 32K$

The Receiver does not execute the action requested in the Payload, but records this new Synchronize Counter value. If a new valid packet (Network

**ADVANCED COMMUNICATIONS & SENSING APPLICATION NOTE**

ID is correct) is received within a 5-second period, with the same Button information and the exact next Synchronization Counter value, the Rx will synchronize its old counter value with the new one, and execute the action as specified by the buttons pressed.

This scenario happens when the user somehow hits a Tx button more than 16 times in a row while the RF communication is not possible (out of range, interference...). When the packet resumes going through, 2 clicks of the same button will make the whole system work again.

Obviously, the Re-Synchronize procedure is transparent to the user. He will naturally press the same key one more time if there is no action on the Rx side.

- Block Window:  $32K < \text{Offset} \leq 64K$

The Receiver does not execute the requested action. The way to recover a Tx falling into Block Window is to use another Tx or a mechanical key to get access to the Rx unit, and run a binding procedure as described in section 4.6.

**4.3.4. Vendor-Configurable Window Setting**

The recommend values for the Open Window and Re-Synchronize Window are 16 slots and (32k-16) slots respectively. However these parameters can be changed according to the vendor requirement. There are 2 build conditionals to control the size of Open Window and Re-Synchronize Window:

**Table 9: Window Slots Build Conditionals**

Build conditional	Description
OPEN_WINDOW_SLOTS	$\text{offset} \in [1, \text{OPEN\_WINDOW\_SLOTS}]$
RE_SYNC_WINDOW_SLOTS	$\text{offset} \in (\text{OPEN\_WINDOW\_SLOTS}, \text{RE\_SYNC\_WINDOW\_SLOTS}]$

**4.3.5. Operations Window Source Code**

The source code defining the Synchronization Window Operation is given in the Appendix.

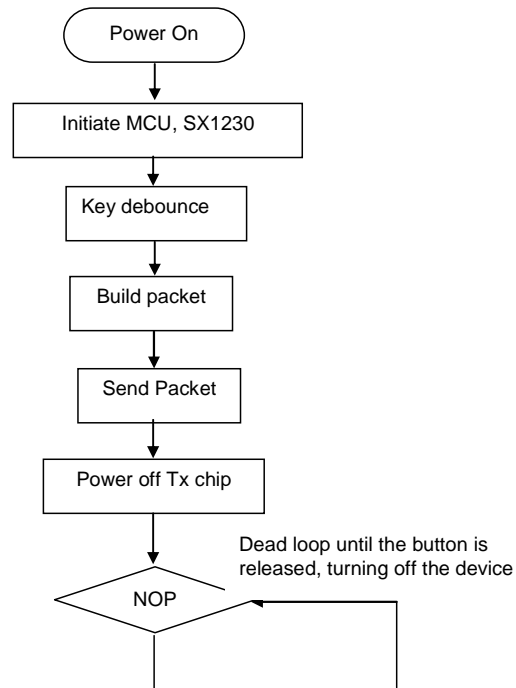
Note:

The Synchronize Counter value is sensitive information. Its value is stored in the EEPROM in both the Transmitter and the Receiver, so as to avoid its loss during a power cut (intermittent supply or battery change). A corrupted Synchronize Counter value causes the loss of synchronization between Tx and Rx, and requires a Binding procedure to recover.

**4.4. Tx Firmware**

To reduce power consumption, Tx is only powered when a key is pressed. Upon button press, Tx will get power and run its routine from the Power-on-Reset state.

#### 4.4.1. Flowchart



**Figure 8: Tx Main Loop Flowchart**

There are 4 tasks in the main loop. A variable recording the current status will determine which task will be executed. The major functions of each task are:

- **Initiate MCU, SX1230:**
  - Initiates the PIC16F631 including clock, watch dog, timer, I/O settings, etc...
  - Initiates the Tx chip SX1230 for all parameters including data mode, bit rate, frequency, deviation settings etc...
  - Also initiates global variables which are used in the Tx firmware.
- **Key Debounce:**  
The debounce time is set to 40ms. If a key is continuously pressed during this period, this key press is confirmed.
- **Build Packet:**  
After a key press is confirmed, the firmware will build the RF packet according to the RF protocol. It includes the encryption process.
- **Send Packet:**  
The MCU will control the RF chip to send the RF packet out. "Send packet" routine is called.

#### 4.4.2. Interrupts

No interrupt is used in the Tx firmware.

### 4.4.3. EEPROM Map

Table 10: Tx EEPROM Map

Byte Offset	Name	Description
0	EE_SIGNATURE	Signature symbol → 'S' (0x51 = 0101_0011B)
1	EE_DEVICE_ID_0	Device ID[0] -- LSB
2	EE_DEVICE_ID_1	Device ID[1]
3	EE_DEVICE_ID_2	Device ID[2]
4	EE_DEVICE_ID_3	Device ID[3] -- MSB
5	EE_SYNC_COUNTER_LSB	Current Synchronize Counter LSB
6	EE_SYNC_COUNTER_MSB	Current Synchronize Counter MSB
7	EE_INDEX_LSB*	Index value LSB
8	EE_INDEX_MSB*	Index value MSB

\*: For "Plain-Index, Binding" protocol. Reserved for "All-Encrypted, Binding" protocol.

## 4.5. Rx Firmware

### 4.5.1. Flowchart

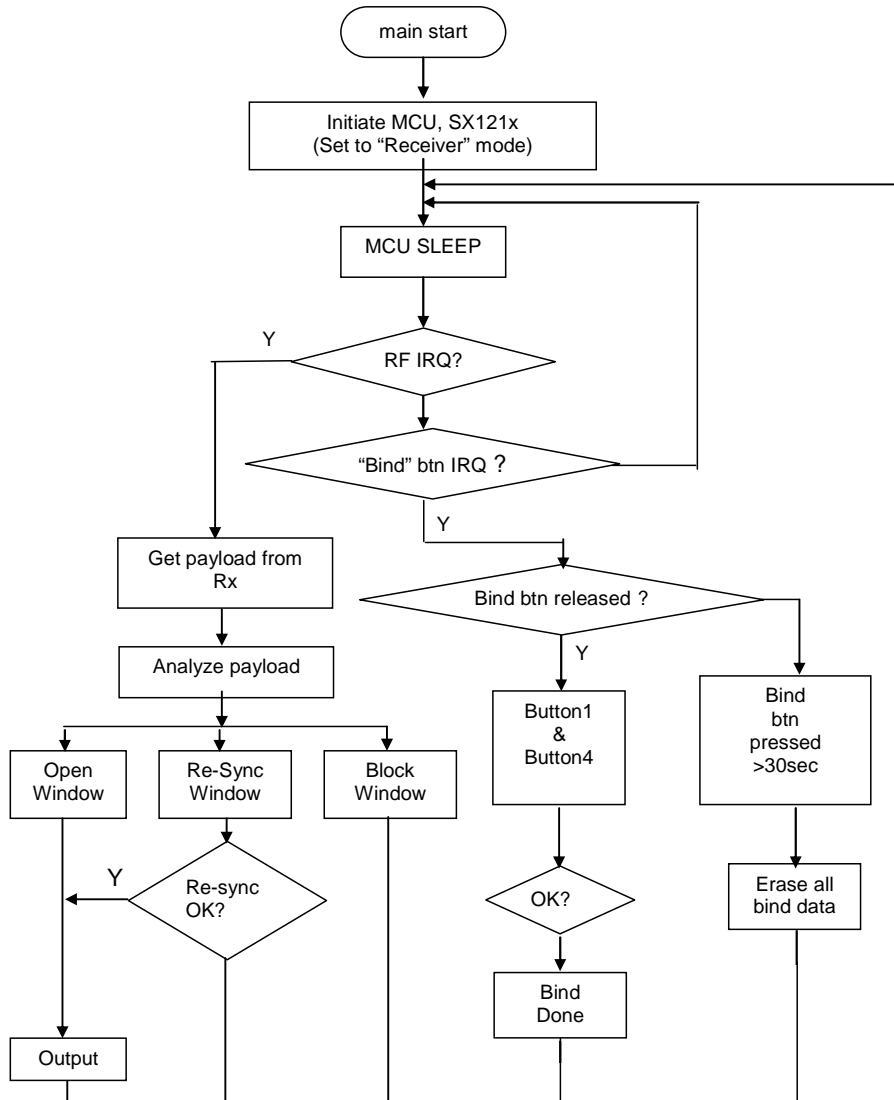


Figure 9: Rx Main Loop Flowchart

### 4.5.2. Interrupts

- External interrupt for "Payload ready" signal from RF chip: asserted when a correct Network ID is detected in the RF chip.
- Port change interrupt for "Bind" button

### 4.5.3. EEPROM map

Table 11: Rx EEPROM Map

Byte Offset	Device No	Name
0	0	Device ID[0] LSB
1	0	Device ID[1]
2	0	Device ID[2]
3	0	Device ID[3] MSB
4	0	Sync Counter LSB
5	0	Sync Counter MSB
6	0	Index value LSB*
7	0	Index value MSB*
8	1	Device ID[0] LSB
9	1	Device ID[1]
10	1	Device ID[2]
11	1	Device ID[3] MSB
12	1	Sync Counter LSB
13	1	Sync Counter MSB
14	1	Index value LSB*
15	1	Index value MSB*
...	...	...
48	6	Device ID[0]
49	6	Device ID[1]
50	6	Device ID[2]
51	6	Device ID[3] MSB
52	6	Sync Counter LSB
53	6	Sync Counter MSB
54	6	Index value LSB*
55	6	Index value MSB*
...	...	...

\*: For “Plain-Index, Binding” protocol. Reserved for “All-Encrypted, Binding” protocol.

#### 4.6. Binding Procedure

Only a Tx and Rx with the same Factory ID can work together since Factory ID is one of the parameters to determine cipher key (due to limitations in the RF protocol, the Rx will show that the binding succeeded even if the Factory ID was different. But this pair of Tx and Rx will not be able to exchange Button Packets, as the cipher keys used for encryption and decryption will not match). To make a pair of Transmitter and Receiver work together, or “Bind” them, a “Bind” procedure is requested. One Rx can record up to 5 Tx. It is not recommended to bind a Tx to multiple Receivers, to avoid loss of synchronization.

To bind two devices, proceed as follows:

- Press the “Bind” key on the Rx. The “Bind” LED will turn on.

- Press “Button1” & “Button4” simultaneously on the Tx within 10 seconds.
- If binding succeeds, the Rx will record the Tx Device ID and current Synchronize Counter value, and blink the “Bind” LED 3 times.
- If binding fails, the Rx will just turn off the “Bind” LED and exit the binding procedure.

**Notes:**

- With the “All-Encrypted, Binding” protocol, the Rx can remember up to 16 Tx IDs using a PIC16F631 and 32 Tx IDs using a PIC16F677. If one more Tx is bound to a Rx already in full capacity, the earliest Tx which was bound will be kicked out from the Rx memory.
- Re-binding a Tx to the Rx to which it is already bound with resynchronizes the Synchronize Counter value. No Tx will be kicked out.
- With the “Plain-Index, Binding” protocol, the Rx can support up to 800 Tx. Attempting to Bind a Tx with an existing Index value in the Rx memory map, will replace the former Tx and update the Synchronize Counter value.
- To erase all Tx units bound with a Receiver:
  - Press and hold the “Bind” button on the Rx for 10 seconds
  - The Rx will erase all bound Tx information in its memory
  - The “Bind” LED will blink 10 times

#### **4.7. Rx Output Types**

Two different actions can be performed by the Rx board when a Button packet is accepted. They are accessible in the software with build conditionals:

- Pulse:  
Upon key press on the Tx, the corresponding output on the Rx will turn on for a pre-defined amount of time.
- Flip-flop:  
The Rx output will toggle each time the corresponding key is pressed on the Tx. The ON/OFF states of all outputs are stored in RAM, so they will be set to ALL\_OFF if Rx power intermittent.

## 4.8. Power Management

- Line powered

If the Rx is powered by line power, it's not necessary to implement special power management schemes. The RF chip is always running in "receiver" mode. The Rx firmware structure detailed above is based on this assumption.

- Battery powered

If the Rx is powered by battery, a special power management is required to guarantee a long battery life. The very fast turn-on times and low power modes of the SX1211/12 are well adapted for advanced power management techniques.

The following tables summarize the power consumption figure of the chips being used:

Conditions: T = 25°C, VDD = 3V

**Table 12: SX1211/12 Power Specification**

Symbol	Description	Typ	Max	Unit
IDDSL	Supply current in sleep mode	0.1	2	μ A
IDDST	Supply current in standby mode CLKOUT disabled	65	80	μ A
IDDR	Supply current in receiver mode	3	3.5	mA

**Table 13: PIC16F631 Power Specification**

Description	Typ	Max	Unit
Supply current in SLEEP mode	0.2	1.5	μ A
Supply current in working mode	700	950	μ A

## 5. Appendixes

### 5.1. Microchip References

More information on the encryption algorithm can be found on AN742, AN744 and TB003, available on the Microchip website.

Please refer to <http://www.microchip.com/keeloq/>

### 5.2. Encryption / Decryption Source Code

The PIC® assembly code for the encoder and decoder can be downloaded at <http://www.microchip.com/keeloq/>, after acceptance of the license agreement.

### 5.3. Operation Window Code

```
// Vendor configurable of Rolling Code decode Operation Windows range
#define OPEN_WINDOW_SLOTS    16
#define RE_SYNC_WINDOW_SLOTS 32768

#define OPEN_WINDOW    0
#define RE_SYNC_WINDOW 1
#define BLOCK_WINDOW   2

_U8 ValidateSyncCounter (_U16 value_in_Rx, _U16 value_from_Tx) {
    _U16 offset;

    // calculates offset
    offset = value_from_Tx - value_in_Rx; // all vars are unsigned

    // check Window range
    if(offset >= 1 && offset <= OPEN_WINDOW_SLOTS) {
        return OPEN_WINDOW;
    }
    if(offset > OPEN_WINDOW_SLOTS && offset <= RE_SYNC_WINDOW_SLOTS) {
        return RE_SYNC_WINDOW;
    }
    return BLOCK_WINDOW;
}
```

### 5.4. Tx Board Schematics

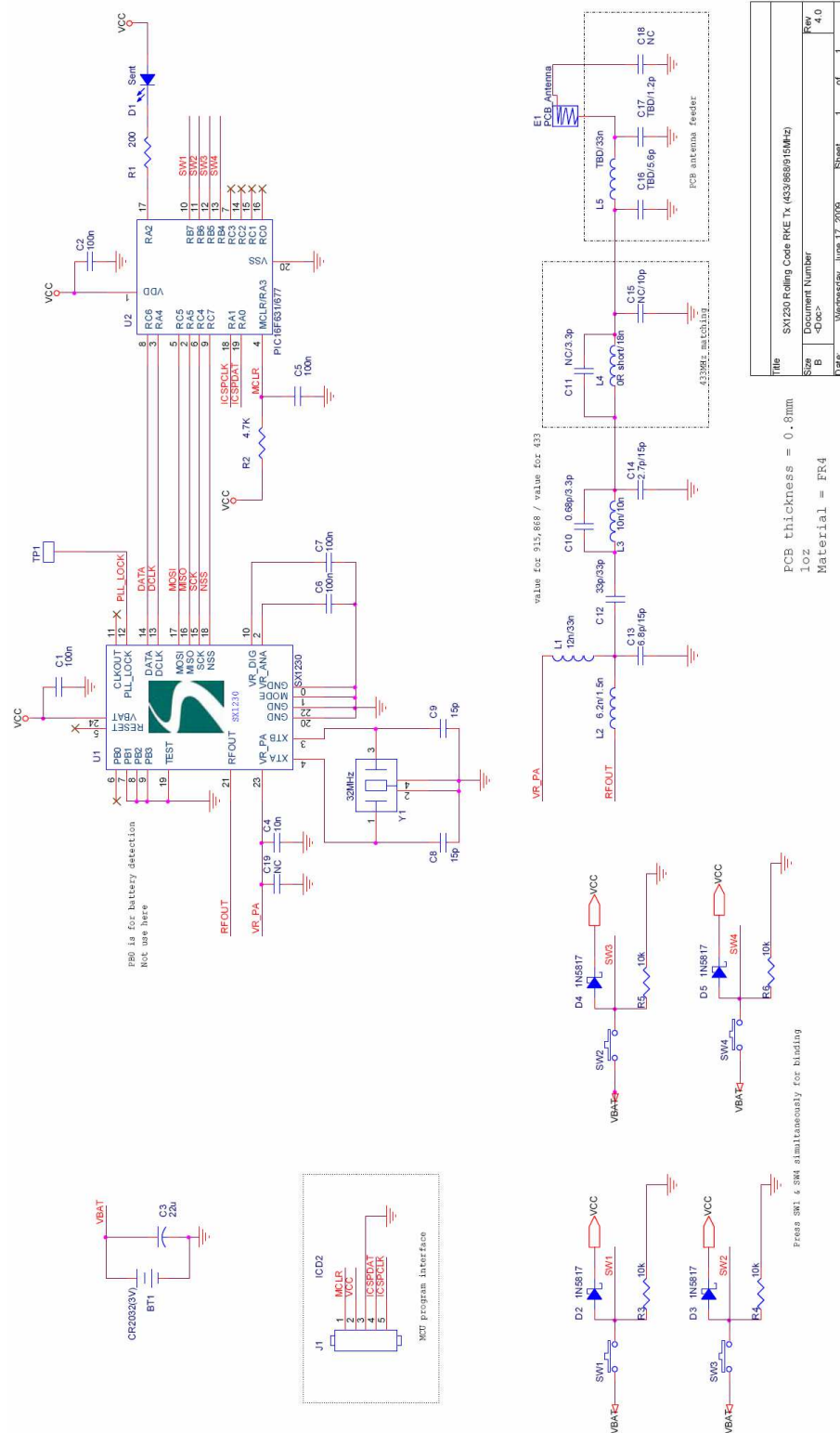


Figure 10: Tx Board Schematics



## 5.6. Bills Of Materials

Table 14: Tx Board BOM

Ref	Qty	Value		Tol +/-	Techno	Footprint	Comment	Manufacturer
		868/915MHz	433MHz					
BT1	1	CR2032 battery						
C1,C2	2	100n		10%	X5R	0402	IC VDD decoupling GRM155R61A104KA01D	Murata
C3	1	22u		10%	X7R	1210	Battery VDD decoupling GRM32ER61C226KE20L	Murata
C4	1	10n		5%	NPO	0805	GRM21BR60J226ME39L	Murata
C5	1	100n		10%	X5R	0402	matching GRM155R71H103KA88D	Murata
C6,C7	2	100n		5%	X5R	0402	MCU reset GRM155R61A104KA01D	Murata
C8,C9	2	15p		5%	NPO	0402	VR decoupling GRM155R61A104KA01D	Murata
C10	1	0.68p	3.3p	5%	NPO	0402	XTAL OSC GRM1555CH150JZ01D	Murata
C11	1	NC	3.3p	5%	NPO	0402	matching GRM1555CH1HR68CZ01D GRM1555CH1HR33CZ01D	Murata
C12	1	33p		5%	NPO	0402	matching GRM1555CH1HR33CZ01D	Murata
C13	1	6.8p	15p	5%	NPO	0402	matching GRM1555CH1H6R8JZ01D GRM1555CH150JZ01D	Murata
C14	1	2.7p	15p	5%	NPO	0402	matching GRM1555CH1H2R7JZ01D GRM1555CH150JZ01D	Murata
C15	1	NC	10p	5%	NPO	0402	matching GRM1555CH100JZ01D	Murata
C16	1	NC(TBD)	5.6p	5%	NPO	0402	matching (50V) GRM1555CH1H5R6DZ01D	Murata
C17	1	NC(TBD)	1.2p	5%	NPO	0402	matching (50V) GRM1555CH1HR2CZ01D	Murata
C18	1	NC(TBD)	NC(TBD)			0402	matching	
C19	1	NC				0402	matching	
D1	1	Sent LED				0603	Red	
D2,D3,D4,D5	4	1N5817			Schottky diode	0805		
E1	1	433MHz				PCB antenna	PCB antenna design for 433MHz	
L1	1	12n	33n	5%	Wire Wound	0402	PA choke LQW15AN12NG00D LQW15AN33NG00D 0402CS-12NX_LW 0402CS-33NX_LW	Murata CoilCraft
L2	1	6.2n	1.5n	5%	Wire Wound	0402	matching LQW15AN6N2B00D LQW15AN1N5B00D 0402CS-6N2X_LW 0402CS-1N2X_LW*	Murata CoilCraft
L3	1	10n		5%	Wire Wound	0402	matching LQW15AN10NG00D 0402CS-10NX-LW	Murata CoilCraft
L4	1	0R	18n	5%	Wire Wound	0402	matching LQW15AN18NG00D 0402CS-18NX-LW	Murata CoilCraft
L5	1	0R	33n	5%	Wire Wound	0402	matching LQW15AN33NG00D 0402CS-33NX_LW	Murata CoilCraft
R1	1	200R		5%		0402	sent LED load	
R2	1	4.7K		1%		0402	reset	
R3,R4,R5,R6	4	10K		5%		0402	button pull down	
Sw1,Sw2,Sw3,Sw4	4	SPST push button				4.8x3 SMD	buttons	
U1	1	SX1230				MLPQ-24(4x4)	Transmit IC	Semtech
U2	1	PIC16F631H677				SSOP20	MCU	Microchip
Y1	1	32MHz		15ppm	AT-out, fundamental	2.5x2.0	XTL581100-S291-102	Siward

Table 15: Rx Board BOM

Ref	Qty	Value			Tol +/-	Techno	Footprint	Comment	Manufacturer
		868MHz	915MHz	433MHz					
BT1	2	AA battery							
C1	1	1u			15%	X5R	0402	VDD decoupling GRM155R60J105KE19D	Murata
C2	1	1u	1u	100n	15%	X5R	0402	Top regulator decoupling GRM155R61A104KA01D	Murata
C3	1	220n	220n	100n	10%	X5R	0402	Digital regulator decoupling GRM155R61A224KE19D GRM155R61A104KA01D	Murata
C4	1	22p	22p	10p	5%	NPO	0402	DC block and L2 adjust GRM155C1H220JZ01D GRM155C1H100JZ01D	Murata
C5	1	47p			5%	NPO	0402	DC block GRM155C1H470JZ01D	Murata
C6	1	1.8p	1.8p	NC	0.25p	NPO	0402	Matching GRM155C1H1R8C201D	Murata
C7	1	NC	NC	6.8p	5%	NPO	0402	Matching GRM155C1H6R8JZ01D	Murata
C8	1	NC	NC	47p	5%	NPO	0402	PA regulator decoupling GRM155C1H470JZ01D	Murata
C9	1	47n	47n	1n	10%	X7R	0402	PA regulator decoupling GRM155R71E473KA88D GRM155R71H102KA01D	Murata
C10	1	680p			5%	NPO	0402	Loop filter GRM155C1H681JA01D	Murata
C11	1	100n			10%	X7R	0402	VCO regulator decoupling GRM155R61A104KA01D	Murata
C12	1	10n			10%	X7R	0402	Loop filter GRM155R71H103KA88D	Murata
C15	1	1u			10%	X7R	0402	VDD decoupling GRM155R60J105KE19D	Murata
C16	1	100n			10%	X7R	0402	VDD decoupling GRM155R61A104KA01D	Murata
D1,D2,D3,D4	4	Output LED					0603	Red	
D5	1	Binding LED					0603	Red	
J1	1	SMA					SMB_V-RJ45	Antenna socket	
L1	1	100n	100n	18n	5%	Wire Wound	0402	PA choke LQW15ANR10J00D LQW15AN18NG00D 0402CS-R10X-LW	Murata CoilCraft
L2	1	8.2n	8.2n	27n	5%	Wire Wound	0402	Matching LQW15AN8N2G00D LQG15HS27N02D 0402CS-8N2X-LW	Murata CoilCraft
L4,L5	2	8.2n	6.8n	19n	0.2nH	Wire Wound	0402	VCO tank inductor LQW15AN8N2G00D LQW15AN6N8G00D LQW15AN19NG00D 0402CS-8N2X_LW 0402CS-6N8X_LW	Murata CoilCraft
R1	1	100K			5%		0402	SX121x DATA pin pull up	
R2	1	1R	1R	0R	1%		0402	PA regulator	
R3	1	6.8K			1%		0402	Loop filter	
R4,R5,R6,R7	4	1K			5%		0402	output LED load	
R8	1	1K			5%		0402	Bind LED load	
R9	1	10K			5%		0402	Bind button pull up	
SW1	1	SPST switch					9x3.5 DIP	Power switch	
SW2	1	SPST push button					6x6 SMD	bind button	
U1	1	SX1211	SX1211	SX1212			TQFN-32	receiver IC	Semtech
U2	1	869MHz	915MHz	434MHz			3.8x3.8	SAW filter // SAFCH434MAM0T00	Murata
U4	1	PIC16F631#677					SSOP20	MCU	Microchip
U5	1	24C256					SOIC8	EEPROM (optional for 800 Tx)	Microchip
Y1	1	12.8MHz			15ppm	AT-cut, fundamental	5.0x3.2	XTL541200-S291-008	Siwad

### 5.7. Boards Pictures

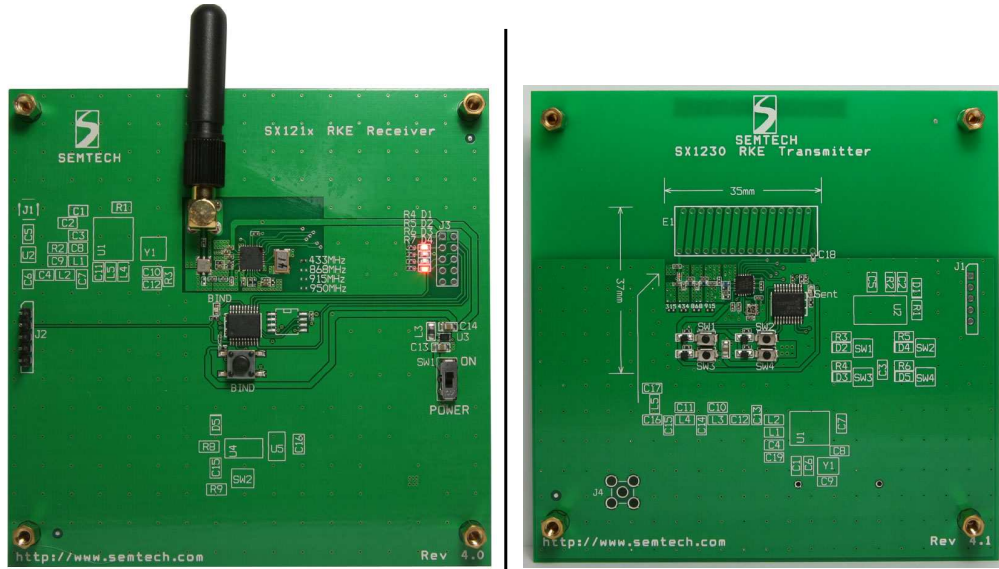


Figure 12: Rx and Tx Boards Pictures

### 5.8. Tx 433MHz PCB Antenna Reference Design

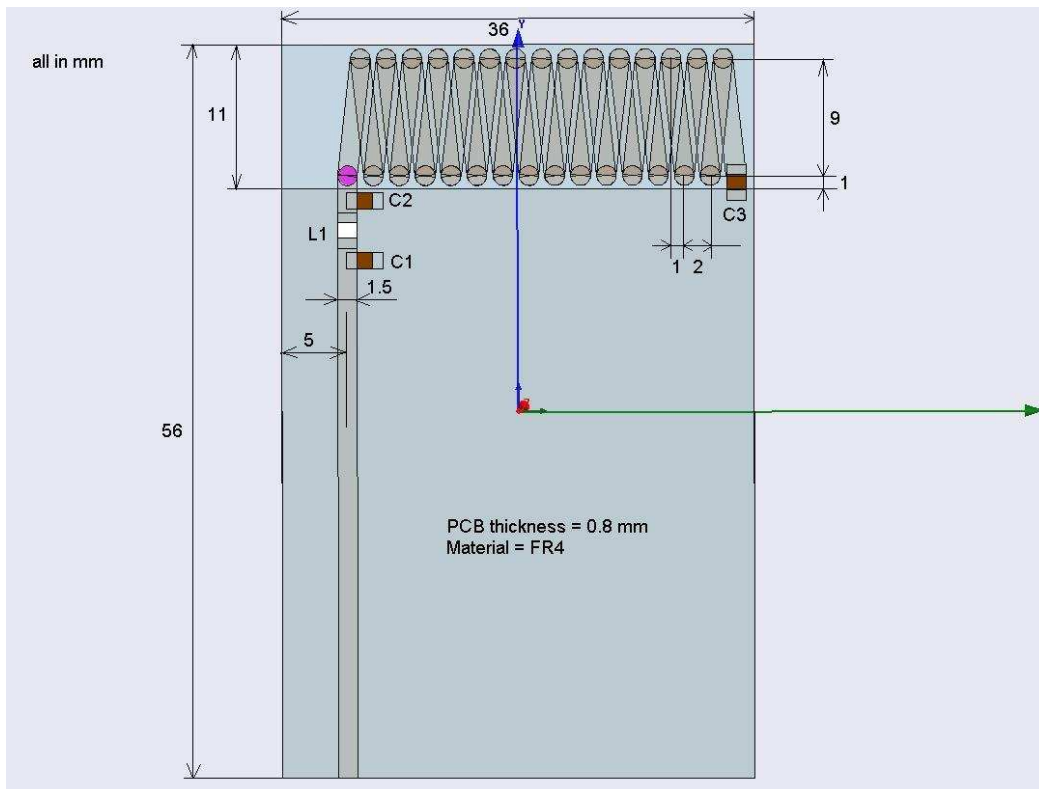


Figure 13: 433 MHz Helical Antenna

### 5.9. Boards Layout

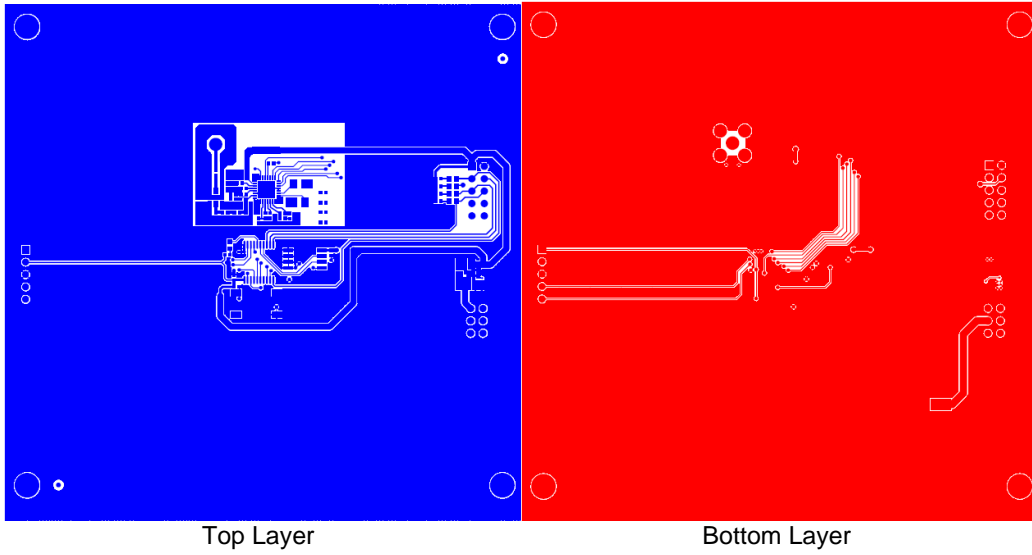


Figure 14: Rx Board Layout

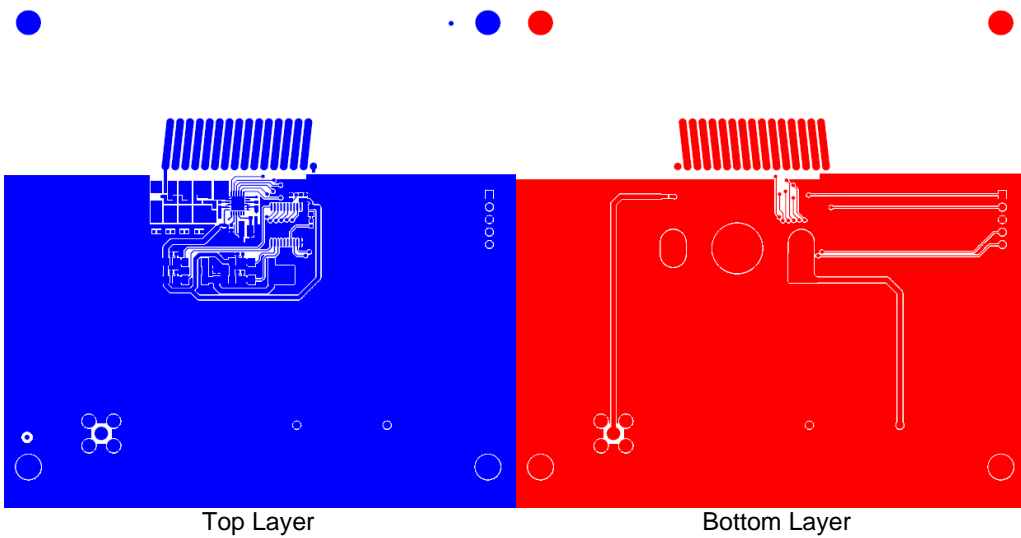


Figure 15: Tx Board Layout

© Semtech 2010

All rights reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent or other industrial or intellectual property rights. Semtech assumes no responsibility or liability whatsoever for any failure or unexpected operation resulting from misuse, neglect improper installation, repair or improper handling or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified range.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

#### Contact Information

Taiwan	Tel: 886-2-2748-3380 Fax: 886-2-2748-3390	Switzerland	Tel: 41-32-729-4000 Fax: 41-32-729-4001
Korea	Tel: 82-2-527-4377 Fax: 82-2-527-4376	United Kingdom	Tel: 44-1794-527-600 Fax: 44-1794-527-601
Shanghai	Tel: 86-21-6391-0830 Fax: 86-21-6391-0831	France	Tel: 33-(0)169-28-22-00 Fax: 33-(0)169-28-12-98
Japan	Tel: 81-3-6408-0950 Fax: 81-3-6408-0951	Germany	Tel: 49-(0)8161-140-123 Fax: 49-(0)8161-140-124