RemoTI™ Coexistence Testing

Keywords

- *RemoTI™*
- *RF4CE*
- *Remote Control*
- *CC2530*
- *2.4GHz*
- *Interference*
- *WiFi*
- *802.11n*

1 Introduction

The ZigBee RF4CE standard protocol [1] was designed to enable RF-based remote control applications based on the IEEE 802.15.4 radio technology in the 2.4-GHz band. One of the main concerns for any wireless system intended to operate in this band is the potential interference from sources like WiFi networks, microwave ovens, cordless phones and bluetooth devices.

This application note describes the features in RF4CE technology that enable it to perform well even in the presence of strong interference. Actual testing is performed using the RemoTI development kit and sample application.

Note that a variety of factors will influence the actual performance. The final measurements and calculations should be performed on customer hardware and expected environment.

Table of Contents

2 Abbreviations

RF4CE	Radio Frequency for Consumer Electronics
RemoTI	Texas Instruments implementation of the RF4CE protocol stack
WiFi	Wireless LAN products based on 802.11 standards
CSMA-CA	Carrier Sense Multiple Access – Collision Avoidance
ACK	Acknowledgement
EM	Evaluation module
MAC	Medium Access Control
RC	Remote Controller
RF	Radio Frequency
RX	Receive
STB	Set Top Box
TX	Transmit

3 Coexistence features

The performance of a RF4CE system in the presence of interference depends on both the radio device performance as well as the protocol stack features.

3.1 RF4CE channels

The IEEE 802.15.4 standard has defined 16 available channels in the 2.4GHz frequency band. These are numbered 11 through 26 and the center frequency of each channel is Fc = 2405 + 5 (k – 11) MHz, for k = 11, 12, , 26

From among these channels, RF4CE has selected 3 channels (numbers 15, 20 and 25) for operation. These channels were selected to minimize interference from WiFi networks. See Figure 1 below for the channel plan of WiFi and 802.15.4. Note that the RF4CE channels fall between the gaps of the most popular WiFi channels.

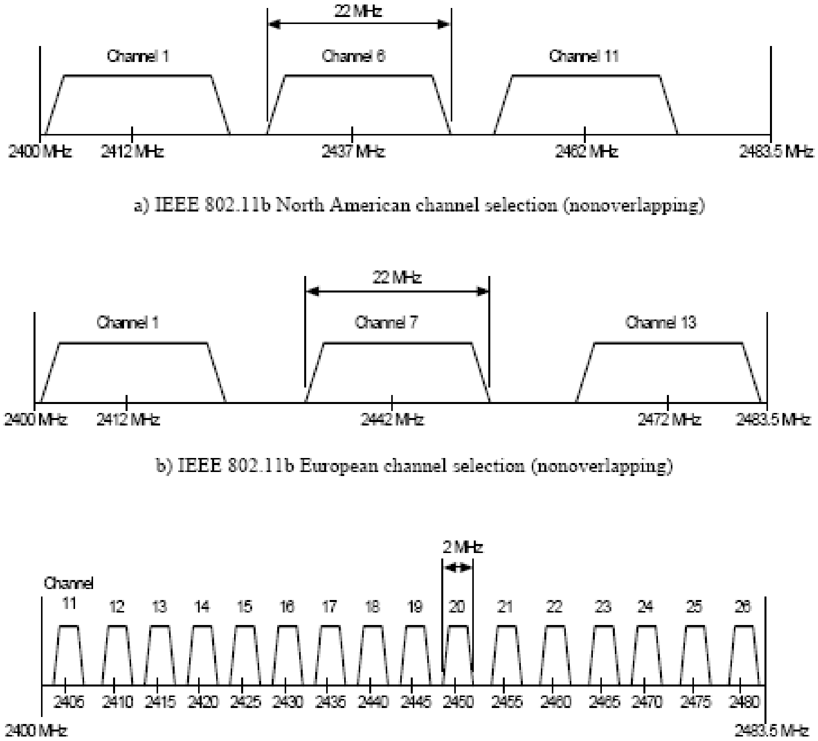


Figure 1: 2.4-Ghz channels (source: IEEE 802.15.4 standard)

3.2 Radio performance

The performance of an 802.15.4 radio in the presence of interference is dependent on the following factors

- Transmit power
- Adjacent channel rejection - This is a measure of the receiver filtering of unwanted signals at +/- 5MHz from the wanted signal
- Alternate channel rejection - This is a measure of the receiver filtering of unwanted signals at +/- 10MHz from the wanted signal

The CC2530 radio has best-in-class performance in all these criteria with a transmit power of +4.5dbm, adjacent channel rejection of 49dB and alternate channel rejection of 57dB [3].

The effect of this is that a CC2530 radio will filter out interference signals that are close to the operational channel much more effectively than other radios.

3.3 MAC features

The 802.15.4 MAC layer combats the effects of interference through the CSMA-CA channel access mechanism and through the use of acknowledgements and retransmissions.

In the CSMA-CA algorithm, the radio first checks the channel to ensure that the energy level is below a threshold. If not, a random backoff delay is performed and the energy detection check is done again. This process continues until the channel is detected to be idle (or until a maximum number of backoffs is performed) and then the packet transmission occurs. This feature ensures that a 15.4 transmitter will attempt to find the times when the channel is idle and transmit at that time rather than overlap its transmissions with other interfering transmissions.

Additionally, the sending device can request an acknowledgement from the receiving device upon reception of the packet. If the sender does not immediately receive the acknowledgement packet, it will retransmit the packet for upto a maximum number of times.

3.4 RF4CE Frequency Agility

The RF4CE network layer has a frequency agility feature that ensures that the Target devices will continuously monitor interference conditions on the current channel and migrate to a different channel if necessary. This is intended to ensure that the devices will always operate on the best possible channel from the three available channels. For details on the frequency agility implementation in the RemoTI software stack, refer to [4].

4 Test setup

The RF4CE performance is measured using the test application provided with the sample application included in the RemoTI-CC2530DK kit. The test mode feature within this application is used [5]. The RC will transmit a fixed number of packets to the Target device and measure the success rate and latency for each packet.

4.1 Physical setup

The interference is generated via an 802.11n network. An 802.11n NIC (network interface card) equipped laptop is associated with an AP (access point). The WiFi network is operating with the 40-MHz channel bonding option on channels 7-11. This is the worst case interference source for RF4CE as it interference strongly with one RF4CE channel (number 20) and also bleeds interference into the other two channels. See Figure 2 below for the actual spectrum utilization as captured via the Wi-Spy analyzer.

Note: *The WiFi equipment used in test was capable of being configured to force the use the 40-MHz. This is done to maximize the interference over the RF4CE channels.*

However, the 802.11n specification is being updated to prevent the use of the 40-Mhz option in the 2.4-GHz band if the access point detects any other wireless signals in that channel. If the access point is using the 40-Mhz option and detects other wireless signals, it will be required to fallback to the 20-Mhz band option.

Also note that major manufacturers including Intel (Centrino chipset) and Apple have not supported the 40-Mhz channel option in the 2.4-Ghz band at all from the beginning. Linksys (Cisco) access points have also already implemented the “fallback” operation

The iPerf tool [6] is used to generate UDP traffic from the laptop to the AP at rates of 6Mbps and 15Mbps.

The distance between the WiFi transmitter and receiver is 4 meters. The controller and target devices are placed in various locations as depicted in Figure 3 below. The RF4CE devices are placed either 20cm from the WiFi device or in the center depending on the scenario.

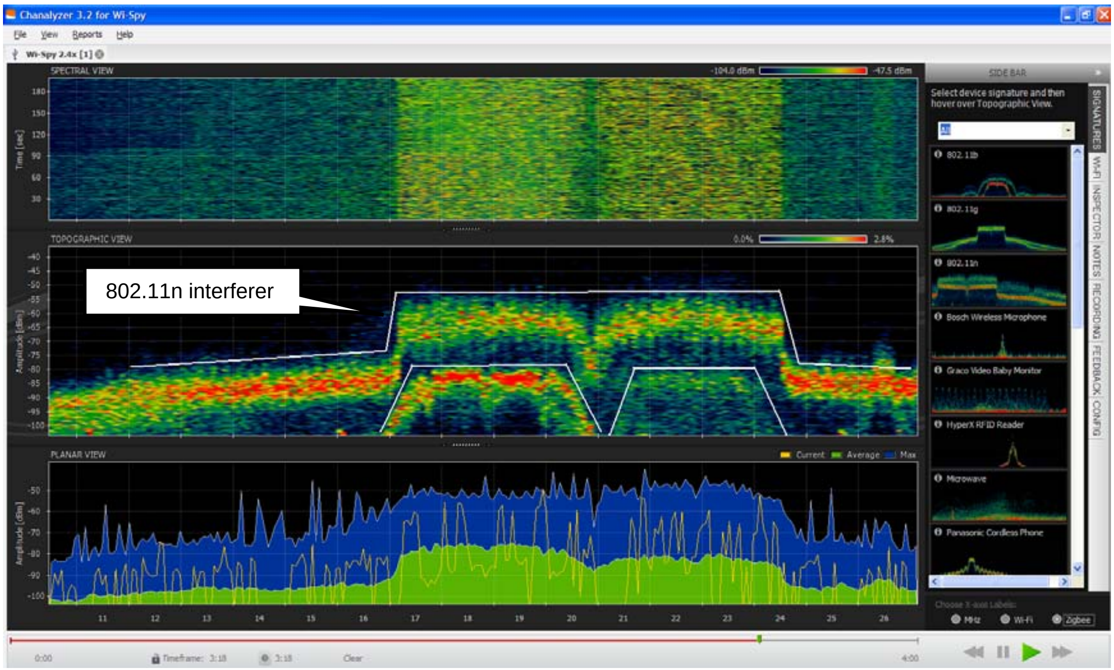
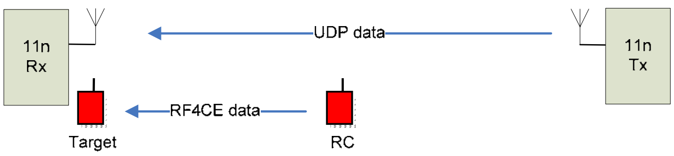
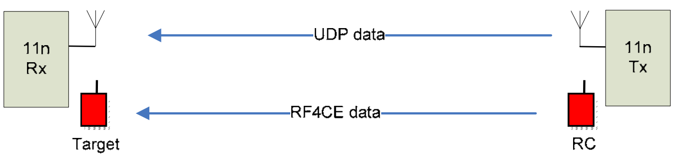


Figure 2: Test interference spectrum

<u>Scenario 1</u>



<u>Scenario 2</u>



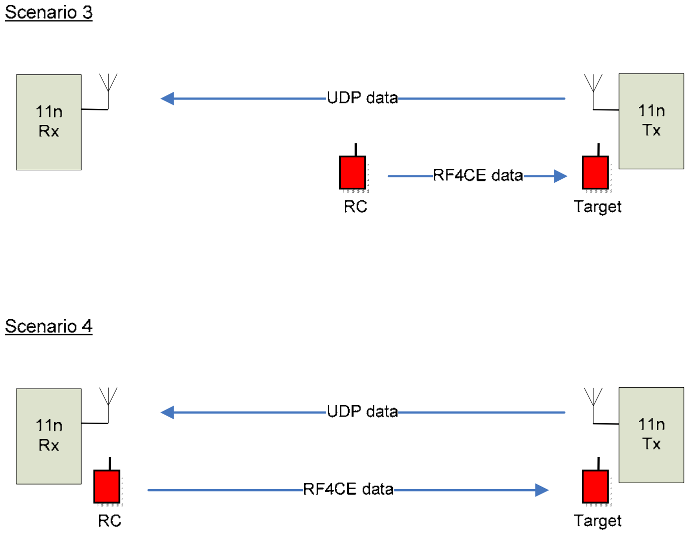


Figure 3: Physical test setup scenarios

4.2 Results

The results are listed in the Table 1 below. From the figures, it is clear the RemoTI solution performs reliably in the presence of strong interference and the only effect of the interference is an increase in packet latency.

In normal conditions, the maximum (99 percentile) packet latency is under 10 milliseconds. In moderate interference conditions this increases to about 20ms, and in very strong interference conditions, it is still within 50ms.

Scenario 1

Latency (ms)	Packets received	
	6mbps	***15mbps***
0 – 10	87.2	78.9
10 – 20	99.8	94.3
20 – 30	100.0	97.9
30 – 40	100.0	98.9
40 – 50	100.0	99.6
50 – 60	100.0	99.8
60 – 70	100.0	99.9
70 – 80	100.0	100.0
80 - 90	100.0	100.0
90 – 100	100.0	100.0

Scenario 2

Latency (ms)	Packets received	
	6mpbs	***15mbps***
0 – 10	81.9	81.9
10 – 20	99.1	95.7
20 – 30	99.7	98.0
30 – 40	99.9	98.9
40 – 50	100.0	99.5

50 – 60	100.0	99.7
60 – 70	100.0	99.8
70 – 80	100.0	99.9
80 - 90	100.0	100.0
90 – 100	100.0	100.0

Scenario 3

Latency (ms)	Packets received	
	6mbps	***15mbps***
0 – 10	91.4	88.6
10 – 20	99.6	98.5
20 – 30	99.8	99.6
30 – 40	99.9	99.9
40 – 50	99.9	99.9
50 – 60	99.9	100.0
60 – 70	100.0	100.0
70 – 80	100.0	100.0
80 - 90	100.0	100.0
90 – 100	100.0	100.0

Scenario 4

Latency (ms)	Packets received	
	6mbps	***15mbps***
0 – 10	83.7	55.2
10 – 20	99.3	79.4
20 – 30	99.8	90.0
30 – 40	99.9	94.0
40 – 50	100.0	96.3
50 – 60	100.0	97.9
60 – 70	100.0	98.8
70 – 80	100.0	99.4
80 - 90	100.0	99.6
90 – 100	100.0	99.8
100 – 150	100.0	100.0

Table 1: Results

4.2.1 Low latency applications

For most remote control applications, maximum packet latency of about 100 - 200 milliseconds is acceptable. However, certain type of applications (like game controllers, for example) require very low latency, typically less than 20 milliseconds. For these applications, the above results indicate that operation in high interference conditions may result in sub-optimal performance.

The increased packet latency is primarily caused by the fact the device performs a channel energy detection and only transmits if it is idle. If not, it will do a random backoff and then try again. In the presence of strong interference, the channel is frequently busy and so the device has to wait longer before it can transmit.

To improve the packet latency for such applications, it is possible to adjust certain MAC and network layer parameters from their default values so that the transmitting device can be more aggressive in terms of finding the channel idle. The relevant parameters that affect the latency are *macMinBE*, *macMaxBE* and *nwkMaxFirstAttemptCSMABackoffs*.

The *macMinBE* and *macMaxBE* parameters determine the amount of random backoff that the device performs each time it determines the channel to be busy during a transmission attempt. For a given packet transmission attempt, the backoff delay is initially set to a random value between 0 and ($2^{macMinBE}$ - 1) in units of 320 microseconds. Every time the channel is detected to be busy for that packet transmission attempt, the range of the backoff delay is doubled upto a maximum of ($2^{macMaxBE}$ - 1). The default values for these parameters are 3 and 5 respectively. These MAC parameters can be modified via the following function

```
extern uint8 MAC_MlmeSetReq(uint8 pibAttribute, void *pValue);
```

The `pibAttribute` identifiers are defined in the MAC specification and are equal to 0x4F for the *macMinBE* attribute and 0x57 for the *macMaxBE* attribute. For the low-latency experiment, these parameter values were reduced to values of 0 and 3 respectively.
The *nwkMaxFirstAttemptCSMABackoffs* parameter determines the maximum number of times the device will perform channel sensing for each packet transmission attempt. Since the device is now more aggressive in performing energy detection, the number of attempts should be increased to allow for more failures.
This parameter is an RTI state attribute and can be modified through the `RTI_WriteItem()` function. The parameter can also be modified at run-time using the RCN interface as shown in example below

```
uint8 value = 5;
RCN_NlmeSetReq(RCN_NIB_NWK_MAX_FIRST_ATTEMPT_CSMA_BACKOFFS,0,&value );
```

The results with the low-latency settings are shown in Table 2 below for test scenario 4 with interference of 12 Mbps. Note that the parameter values can be further modified based on customer application requirements and testing.

Latency (ms)	Packets received	
	Default parameters	***Low-latency settings***
0 – 10	73.1	88.3
10 – 20	90.2	95.8
20 – 30	94.8	98.1
30 – 40	97.1	99.1
40 – 50	98.5	99.5
50 – 60	99.1	99.8
60 – 70	99.5	99.9
70 – 80	99.7	99.9
80 - 90	99.8	100.0
90 – 100	99.9	100.0

Table 2: Low latency results

5 Conclusion

The RemoTI protocol stack and the CC2530 radio combine to give a high level of performance in interference conditions. The reliability of the packet transmissions is not affected. Instead, the only effect of interference is the increased packet latency. The increased latency is well within the requirements for remote control applications. However, for those applications that need very low latency, certain parameters can be modified to achieve low latency even in the presence of strong interference.

References

[1] ZigBee RF4CE Specification v1.00 (ZigBee Alliance document 094945r00)

[2] IEEE 802.15.4-2006, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). *http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf*

[3] CC2530 Datasheet, Texas Instruments, *www.ti.com/lit/gpn/cc2530*

[4] RemoTI Developer's Guide, Texas Instruments, *www.ti.com/lit/swru198*

[5] RemoTI Sample Application User's Guide, Texas Instruments (*included in RemoTI software package*)

[6] Jperf network traffic generator tool. *http://code.google.com/p/xjperf/*

6 General Information

6.1 Document History

Revision	Date	Description/Changes
SWRA285	2009.05.12	Initial release.