RSSI Interpretation and Timing

By Siri Namtvedt

Keywords

- *CC1100*
- *CC1101*
- *CC1110*
- *CC1111*
- *CC2500*
- *CC2510*
- *CC2511*
- *RSSI*

1 Introduction

The Received Signal Strength Indicator is a measure of the RF power input to the transceiver. The RSSI value is based on the gain setting in the RX chain and the measured signal level in the channel. In RX mode, the RSSI value can be read continuously from the `RSSI` status register until the demodulator detects a sync word (when sync word detection is enabled). At that point the RSSI readout value is frozen until the next time the chip enters the RX state. If sync word detection is disabled, the `RSSI` register will be continuously updated.

Table of Contents

2 Abbreviations

AGC	Automatic Gain Control
BW	Band Width
CS	Carrier Sense
RSSI	Received Signal Strength Indicator
WOR	Wake on Radio

3 RSSI Register Interpretation

The RSSI value read from the `RSSI` status register is a 2's complement number and is in dBm with ½ dB resolution. The following procedure can be used to convert the RSSI reading to an absolute power level (RSSI_dBm).

Assume that the RSSI offset is 72 dB and that 11111010_b is read from the `RSSI` register.

If 11111010_b is interpret as a 2's complement number it has the value -6.

This means that the RSSI value, before taking the offset into account, is -3 dBm. Subtracting the RSSI offset, the RSSI value in dBm ends up at -3 dBm - 72 dB = -75 dBm.

```
UINT8 rssi_dec;
INT16 rssi_dBm;
UINT8 rssi_offset = 72;

rssi_dec = halSpiReadStatus(CCxxx0_RSSI);

if (rssi_dec >= 128)
    rssi_dBm = (INT16)((INT16)( rssi_dec - 256) / 2) - rssi_offset;
else
    rssi_dBm = (rssi_dec / 2) - rssi_offset;
```

4 RSSI Offset

The RSSI offset will vary with register settings, and hence different data rates (using different AGC settings) will have different RSSI offset. Table 1 shows the RSSI offset for some data rates for CC1100, CC1101, CC1110/CC1111, CC2500, and CC2510/CC2511.

Data Rate [kBaud]	**CC1100**		**CC1101**		**CC1110/ CC1111**			**CC2500**	**CC2510/ CC2511**
	RSSI Offset [dB]		RSSI Offset [dB]		RSSI Offset [dB]			RSSI Offset [dB]	RSSI Offset [dB]
	433 [MHz]	868 [MHz]	433 [MHz]	868 [MHz]	315 [MHz]	433 [MHz]	868 [MHz]		
1.2	75	74	74	74	74	75	73	-	-
2.4	-	-	-	-	-	-	-	71	74
10	-	-	-	-	-	-	-	69	74
38.4	75	74	74	74	73	74	73	-	-
250	79	78	74	74	74	73	77	72	71
500	79	77	74	74	-	-	-	72	72

Table 1. RSSI Offset

When using data rates other than the ones in Table 1, it is possible to find the RSSI offset by applying a known input power and then read out the RSSI value.

5 RSSI Update Rate

The RSSI update rate, f_{RSSI} (Equation 1), depends on the receiver filter bandwidth ($BW_{channel}$, Equation 2) and filter length (see the `AGCCTRL0` register in the data sheet ([1], [2], [3], [7], and [8])).

$$f_{RSSI} = \frac{2 \cdot BW_{channel}}{8 \cdot 2^{FILTER\_LENGTH}}$$

Equation 1. RSSI Update Rate

$$BW_{channel} = \frac{f_{xosc}}{8 \cdot (4 + CHANBW\_M) \cdot 2^{CHANBW\_E}}$$

Equation 2. Receiver Filter Bandwidth

The time it takes from the chip is in RX mode to a valid RSSI value is present in the `RSSI` register can be estimated as shown in Equation 3 (only valid for weak signals; ~-70 dBm or lower):

$$RSSI\ Response\,Time = \frac{8}{BW_{channel}} \cdot \left(\frac{\text{WAIT\_TIME}}{2} + 1 + 2^{FILTER\_LENGTH} \right) + MAX\left[\frac{20}{BW_{channel}}, \frac{10}{(8 \cdot \text{DataRate})} \right]$$

Equation 3. RSSI Response Time

It is important to notice that this equation only gives an indication of the time required in RX before a valid RSSI value can be obtained (and it is only valid for weak signals). Adding a safety margin is therefore necessary in a real life application.

Example (CC2500):

Data Rate = 10 kBaud

`AGCCTRL0` = 0x91 ⇒ `AGCCTRL0.WAIT_TIME` = 1
`AGCCTRL0.FILTER_LENGTH` = 1

`MDMCFG4` = 0x78 ⇒ `MDMCFG4.CHANBW_M` = 3
`MDMCFG4.CHANBW_E` = 1

$$BW_{channel} = \frac{f_{xosc}}{8 \cdot (4 + CHANBW\_M) \cdot 2^{CHANBW\_E}} = \frac{26 \cdot 10^6}{8 \cdot (4+3) \cdot 2^1} = \frac{26 \cdot 10^6}{112} = 232142.9\,[Hz]$$

$$RSSI\ Response\,Time = \frac{8}{232142.9} \cdot \left(\frac{1}{2} + 1 + 2^1 \right) + MAX\left[\frac{20}{232142.9}, \frac{10}{(8 \cdot 10000)} \right] = 246\,[us]$$

The equation for estimating the RSSI response time can for example be used when estimating the time the radio will be in RX when using WOR and direct RX termination based on RSSI measurements (`MCSM2.RX_TIME_RSSI` = 1). Using this feature, RX mode will be terminated if the RSSI level is below a programmable threshold. For the CC1100, the following test where done:

A CC1100, operating at 76.8 kBaud and $BW_{channel}$ = 232 kHz, was put in WOR mode with direct RX termination (This test cannot be performed on CC1110/CC1111 and CC2510/CC2511 since these devises do not support WOR). For different values of `AGCCTRL0.WAIT_TIME` and `AGCCTRL0.FILTER_LENGTH`, the time spent in RX (before terminating due to no CS) was measured and compared to the estimated RSSI response time. The results can be seen in Table 2.

FILTER_LENGTH	WAIT_TIME	Estimated RSSI Response Time [us]	Measured time spent in RX before termination [us]
0	0	155	152
0	1	172	170
0	2	190	186
0	3	207	204
1	0	190	189
1	1	207	204
1	2	224	222
1	3	241	238
2	0	258	257
2	1	276	276
2	2	293	294
2	3	310	316

Table 2. Estimated vs. Measured RSSI Response Time (76.8 kBaud)

For stronger signals, it will take longer time before one can read a valid RSSI value since the AGC requires longer time to set the gain. Assume an application where one wants to put the radio in RX for a minimum amount of time before reading the `RSSI` register. One of the challenges is that the time taken before a valid RSSI value can be read varies with signal strength, and the signal strength is unknown. In this case one should do a set of measurements where a known strong signal is input to the receiver, and the `RSSI` register is read continuously. One should then measure the time from the radio enters RX state until a valid RSSI value is available. Figure 1, Figure 2, and Figure 3 shows 3 different signals strengths (-50 dBm, -70 dBm, and -86 dBm) at 3 different data rates (2.4 kBaud, 10 kBaud, and 250 kBaud) and the time it takes before a valid value can be read from the `RSSI` register. CC2500 was used for this test and register settings were obtained from SmartRF® Studio.

Pseudo code for test firmware:

1. Strobe RX
2. Wait until the radio is in RX mode
3. Start reading the RSSI register repeatedly and store the values in a buffer (for the plots in Figure 1, Figure 2, and Figure 3, the `RSSI` register was read 250 times, with 5.96 us between readings)
4. From the RSSI register values, calculate the actual RSSI value and plot the RSSI values vs. time

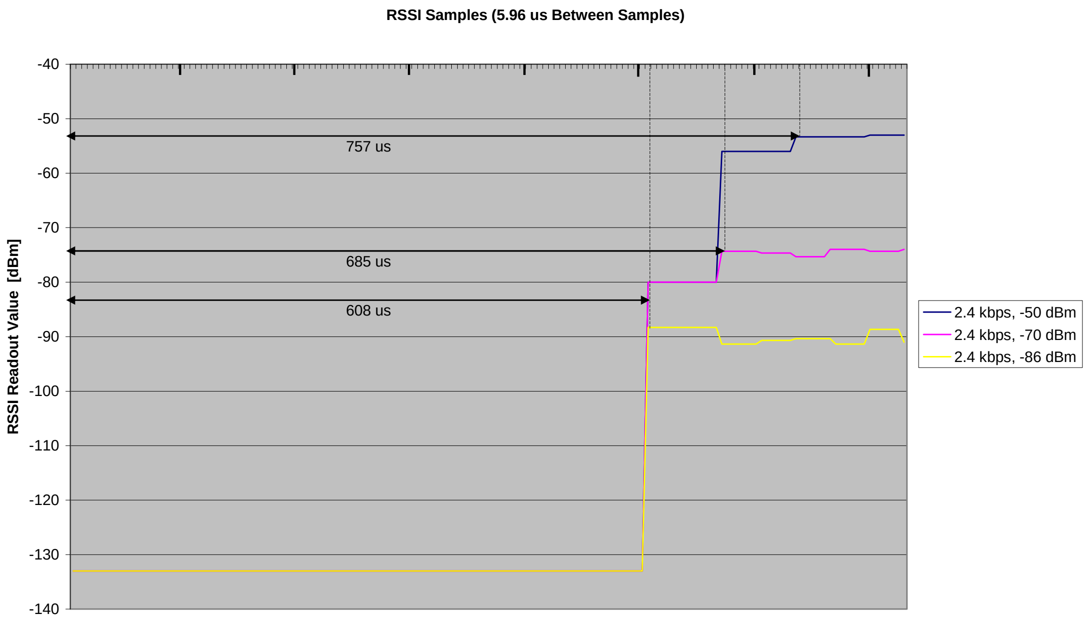

Figure 1. RSSI Value vs. Time (2.4 kBaud)

$$RSSI\ Response\ Time = \frac{8}{203125}\cdot\left(\frac{1}{2}+1+2^{1}\right)+MAX\left[\frac{20}{203125},\frac{10}{(8\cdot 2400)}\right]=658.7\,[us]$$

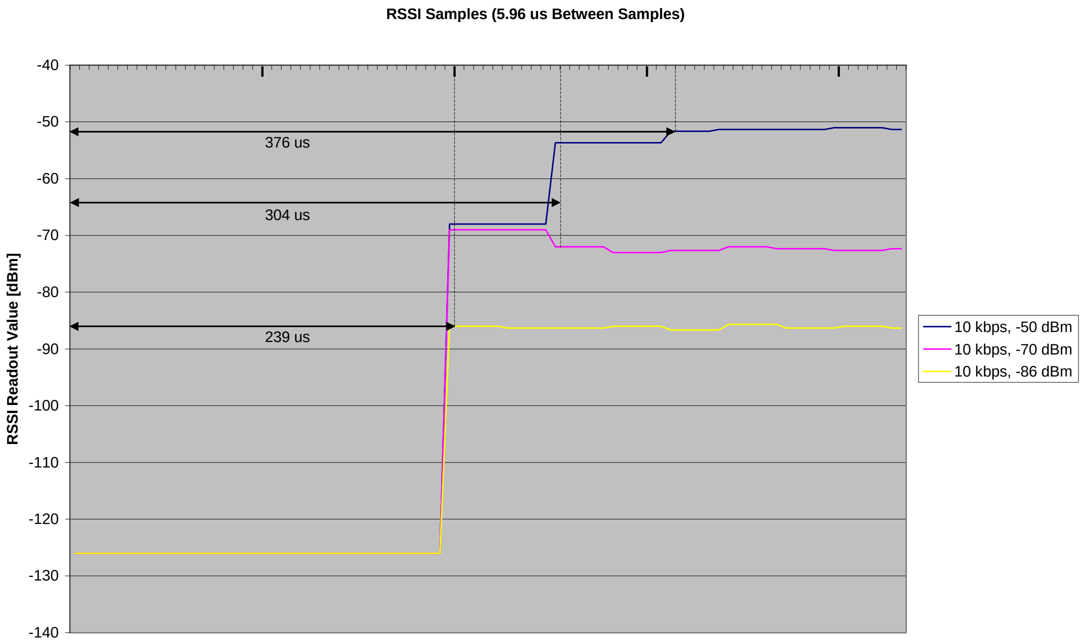

Figure 2. RSSI Value vs. Time (10 kBaud)

$$RSSI\ Response\ Time = \frac{8}{232142.857}\cdot\left(\frac{1}{2}+1+2^{1}\right)+MAX\left[\frac{20}{232142.857},\frac{10}{(8\cdot 10000)}\right]=245.6\,[us]$$

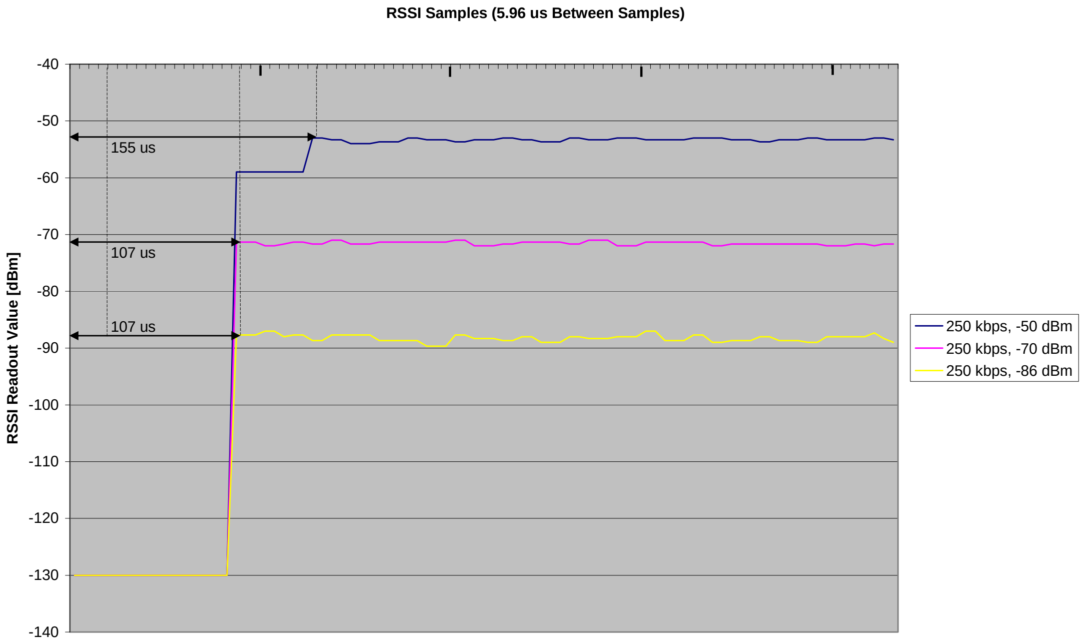

Figure 3. RSSI Value vs. Time (250 kBaud)

$$RSSI\ Response\,Time = \frac{8}{541666.667}\cdot\left(\frac{3}{2}+1+2^{1}\right)+MAX\left[\frac{20}{541666.667},\frac{10}{(8\cdot 250000)}\right]=103.4\,[us]$$

6 Improving RSSI Readout Accuracy

It is important to be aware that in a real system, the output power from the transmitter and variations in the transmission path will contributes to the RSSI inaccuracy. However, there are certain steps one can do, to improve the RSSI readout accuracy

1) Calculate an average RSSI value based on several readouts
2) Follow workaround for the SPI read synchronization issue (see [4], [5], and [6]). This is only necessary on the CC1100/CC1101/CC2500, not the SoCs.

7 RSSI Threshold

Table 3 shows typical RSSI value at CS threshold when the data rate is 250 kBaud, AGCCTRL2.MAGN_TARGET = 3, and AGCCTRL1.CARRIER_SENSE_ABS_THR = 0.

		AGCCTRL2.MAX_DVGA_GAIN[1:0]			
		00	01	10	11
AGCCTRL2.MAX_LNA_GAIN[2:0]	000	-96	-90	-84	-78.5
	001	-94.5	-89	-83	-77.5
	010	-92.5	-87	-81	-75
	011	-91	-85	-78.5	-73
	100	-87.5	-82	-76	-70
	101	-85	-79.5	-73.5	-67.5
	110	-83	-76.5	-70.5	-65
	111	-78	-72	-66	-60

Table 3. Typical RSSI Value in dBm at CS Threshold

Assume MAX_LNA_GAIN = 4 and MAX_DVGA_GAIN = 1. This means that when MAGN_TARGET = 3 and CARRIER_SENSE_ABS_THR = 0, the RSSI value at CS threshold is -82 dBm.

AGCCTRL2.MAGN_TARGET[2:0]		AGCCTRL1.CARRIER_SENSE_ABS_THR[3:0]	
0	24 dBm	-8	Absolute carrier sense threshold disabled
1	27 dBm	-7	7 dB below MAGN_TARGET setting
2	30 dBm	..	. . .
3	33 dBm	-1	1 dB below MAGN_TARGET setting
4	36 dBm	0	At MAGN_TARGET setting
5	38 dBm	1	1 dB above MAGN_TARGET setting
6	40 dBm	..	. . .
7	42 dBm	7	7 dB above MAGN_TARGET setting

Table 4. MAGN_TARGET and CARRIER_SENSE_ABS_THR Settings

Changing MAGN_TARGET to 7 and CARRIER_SENSE_ABS_THR to 4, will change the CS threshold from -82 dBm to -69 dBm (-82 + (42 - 33) + (4 - 0) = -69 [dBm]).

8 References

[1] CC1100 Single-Chip Low Cost Low Power RF-Transceiver, Data sheet (cc1100.pdf)

[2] CC1101 Single-Chip Low Cost Low Power RF-Transceiver, Data sheet (cc1101.pdf)

[3] CC2500 Single-Chip Low Cost Low Power RF-Transceiver, Data sheet (cc2500.pdf)

[4] CC1100 Errata Notes (swrz012.pdf)

[5] CC1101 Errata Notes (swrz020.pdf)

[6] CC2500 Errata Notes (swrz002.pdf)

[7] CC1110Fx/CC1111Fx Low-Power Sub-1 GHz RF System-on-Chip (SoC) with MCU, Memory, Transceiver, and USB Controller (cc1110f32.pdf)

[8] CC2510Fx/CC2511Fx Low-Power SoC (System-on-Chip) with MCU, Memory, 2.4 GHz RF Transceiver, and USB Controller (cc2510f32.pdf)

9 General Information

9.1 Document History

Revision	Date	Description/Changes
SWRA114B	2007.10.22	Removed logo from header. Added info about CC1101 and CC1111. Added info to Table 1.
SWRA114A	2007.01.12	Corrected error in code example. Added chapter 7. RSSI Threshold
SWRA114	2006.10.23	Initial release.

IMPORTANT NOTICE