Current Consumption for a Polling Receiver

By Siri Johnsrud and Hung Li

Keywords

- *Wake On Radio*
- *Sleep Timer*
- *CC1100*
- *CC1101*
- *CC1100E*
- *CC1110Fx*
- *CC1111Fx*
- *CC2500*
- *CC2510Fx*
- *CC2511Fx*

1 Introduction

The purpose of this design note is to show how to optimize the current consumption for a polling receiver based on some of the digital features of the CC1100 [1], CC1101 [2], CC1100E [11] and CC2500 [3], such as WOR and RSSI. The design note will also show how a polling receiver can be implemented using the CC1110/CC1111Fx [4] and CC2510/CC2511Fx [5]. This design note will assume that the reader is familiar with the WOR functionality of the CC1100, CC1100E, CC1101 and CC2500. Details on WOR can be found in AN047 [6]. The current consumption plots shown throughout this document are obtained from measurements done on the CC2500 and the CC2510Fx, but will look similar for the CC1100, CC1100E, CC1101, and the CC1110Fx (the RX current consumption will be different). Be aware that there are only done a couple of measurements on each data rate. This means that the average current consumption given in this document is not necessarily typical values. The values are simply given to provide a better understanding on how different parameters affect the current consumption. Project collateral discussed in this design note can be downloaded from the following URL: http://www.ti.com/lit/zip/SWRA207.

Table of Contents

2 Abbreviations

HS	High Speed
PM{n}	Power Mode n
RSSI	Received Signal Strength Indicator
SoC	System on Chip (in this document, used for CC1110Fx/CC1111Fx and CC2510Fx/CC2511Fx)
WOR	Wake On Radio
XOSC	Crystal Oscillator

3 System Overview

This design note will use the following scenario as a case study to show different ways of implementing a polling receiver:

A receiver enters RX mode once every second to listen for an incoming packet. When the receiver is not listening for packets, it saves power by entering SLEEP state (CC2500) or PM2 (CC2510Fx).

A transmitter is transmitting the same packet repeatedly for a little more than 1 second to make sure that a packet is available when the receiver is in RX mode.

Figure 1 illustrates how the transmitter and receiver are configured to operate.

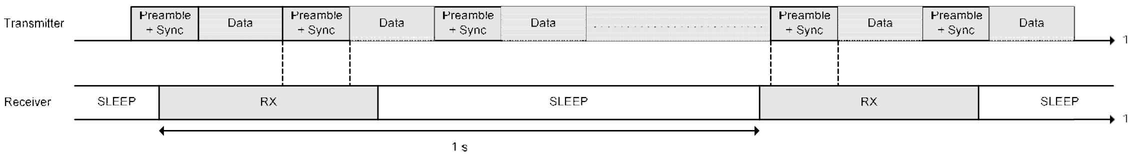


Figure 1. System Overview

3.1 RX Period

The RX period is given by the RX termination timer and is in the data sheets ([1], [2], [3], [4], and [5]) referred to as the RX timeout. Assume that the packet is 18 bytes long and have the following format:

- 4 bytes preamble
- 4 bytes sync word
- 10 data bytes (including optional length byte, address byte, payload, and CRC bytes)

To guarantee that the receiver is in RX mode long enough to receive a packet, the RX timeout should be minimum the duration of 26 bytes (18 bytes (a complete packet) + 8 bytes (preamble + sync)). The reason is that if the radio wakes up just too late detect a sync word (as for the first RX period in Figure 1) it should be able to detect the subsequent sync word. Note that if the transmitter is not able to transmit packets back-to-back, the delay between packets will come in addition to the duration of the 26 bytes.

Two different data rates are used for example purposes; 10 kBaud and 250 kBaud. Minimum RX timeout for the different data rates are shown in Table 1.

Data Rate [kBaud]	Minimum RX Timeout [ms]	Closest RX Timeout Available [ms]
10	$26 \cdot 8 \cdot (1/10) = 20.8$	31.25
250	$26 \cdot 8 \cdot (1/250) = 0.832$	1.95

Table 1. Minimum RX Timeout

3.2 Limitations

Be aware that this design note only focuses on the current consumption in the cases where no data is being received (the radio returns to SLEEP state/PM2 after the RX timeout is reached or RX is terminated due to no carrier sense). If data is being received, the current consumption will increase, but how much it increases is very application dependant. The following factors will have a say on the average current consumption:

- How often a packet is being received (once every second, hour, day?)
- How long the packets are and when in the RX period they are being received (a long packet received at the end of the RX period will force the radio to spend additional time in RX mode, hence increasing the current consumption)

- The current consumption of the MCU both in low power mode (while waiting for a packet) and in active mode (while processing the packet) for the CC2500 case, and the method of reception used on the CC2510Fx (DMA, interrupt, polling)
- Processing time for the incoming packets
- Data rate (only two different data rates are used in this design note)

4 Test Cases

12 different test cases are discussed throughout this document; 6 for the CC2500 and 6 for the CC2510Fx. Table 2 lists the different test cases.

Test Case #	CC2500	Test Case #	CC2510Fx
1	The RX strobe is issued **346.2 µs**[1] after exiting SLEEP state. The PLL is **calibrated** when going from IDLE to RX. The radio stays in RX for **31.25 ms**[1]	7	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **calibrated** when going from IDLE to RX. The radio stays in RX for **31.25 ms**[1]
2	The RX strobe is issued **173.1 µs**[1] after exiting SLEEP state The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **31.25 ms**[1]	8	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **31.25 ms**[1]
3	The RX strobe is issued **173.1 µs**[1] after exiting SLEEP state. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **~246 µs**[3]	9	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **~246 µs**[3]
4	The RX strobe is issued **346.2 µs**[1] after exiting SLEEP state. The PLL is **calibrated** when going from IDLE to RX. The radio stays in RX for **1.95 ms**[1]	10	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **calibrated** when going from IDLE to RX. The radio stays in RX for **1.95 ms**[1]
5	The RX strobe is issued **173.1 µs**[1] after exiting SLEEP state. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **1.95 ms**[1]	11	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **1.95 ms**[1]
6	The RX strobe is issued **173.1 µs**[1] after exiting SLEEP state. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **~89 µs**[3]	12	The RX strobe is issued **350 µs**[2] after exiting PM2. The PLL is **not calibrated** when going from IDLE to RX. The radio stays in RX for **~133 µs**[3]

Table 2. Test Cases

[1] See Table 3
[2] PM2 to Active: ~100 µs, HS XOSC start-up time: ~250 µs
[3] See DN505 [8]

5 CC2500

5.1 Radio Configuration for CC2500

For both data rates, preferred settings (optimized for sensitivity) from SmartRF™ Studio [7], version 6.9.2.0, are used, with some modifications; `PKTLEN` = 0x3D (to avoid overflow of the RXFIFO) and `IOCFG2` = 0x2E. Three other registers are modified for the different test cases, and these registers are `MCSM0`, `WORCTRL`, and `MCSM2` (see Table 3 for details).

Register	Register Field	Comment	
MCSM0	FS_AUTOCAL	01	The PLL is calibrated every time before the radio enters RX mode.
		00	The PLL is not being calibrated (manual calibration must be performed at a given time interval depending on the environment the system operates in)
	PO_TIMEOUT	00	CHP_RDYn is asserted 2.4 µs after the crystal is stable (CHP_RDYn is asserted 150 µs + 2.4 µs = 152.4 µs after the chip exit SLEEP state → EVENT1 should be 001 or higher)
		10	CHP_RDYn is asserted 155.1 µs after the crystal is stable (CHP_RDYn is asserted 150 µs + 155.1 µs = 305.1 µs after the chip exit SLEEP state → EVENT1 should be 011 or higher)
WORCTRL	EVENT1	001	t_{Event1} = 173.1 µs
		011	t_{Event1} = 346.2 µs
MCSM2	RX_TIME_RSSI	0	Radio will go back to SLEEP when the RX timeout expires
		1	Radio will go back to SLEEP if the RSSI level is below a certain threshold. Please see DN505 [8] for details on how to estimate this time for the different register settings.
	RX_TIME	010	Duty cycle = 3.125% and hence RX timeout = 31.25 ms (used when the data rate is 10 kBaud, since minimum RX timeout is 20.8 ms (see Section 3)).
		110	Duty cycle = 0.195% and hence RX timeout = 1.95 ms (used when the data rate is 250 kBaud, since minimum RX timeout is 832 µs (see Section 3)).

Table 3. Registers Modified for the Different Test Cases (1 -6)

5.1.1 Register Settings for the Different CC2500 Test Cases (Test Case 1 - 6)

Register settings for the different test cases are shown in Table 4.

Data Rate [kBaud]	Test Case #	Register	Reg. Setting	Comment
10	1	MCSM0	0x18	FS_AUTOCAL = 01 PO_TIMEOUT = 10
		WORCTRL	0x38	EVENT1 = 011
		MCSM2	0x02	RX_TIME_RSSI = 0 RX_TIME = 010
	2	MCSM0	0x00	FS_AUTOCAL = 00 PO_TIMEOUT = 00
		WORCTRL	0x18	EVENT1 = 001
		MCSM2	0x02	RX_TIME_RSSI = 0 RX_TIME = 010
	3	MCSM0	0x00	FS_AUTOCAL = 00 PO_TIMEOUT = 00
		WORCTRL	0x18	EVENT1 = 001
		MCSM2	0x12	RX_TIME_RSSI = 1 RX_TIME = 010

250	4	`MCSM0`	0x18	`FS_AUTOCAL` = 01 `PO_TIMEOUT` = 10
		`WORCTRL`	0x38	`EVENT1` = 011
		`MCSM2`	0x06	`RX_TIME_RSSI` = 0 `RX_TIME` = 110
	5	`MCSM0`	0x00	`FS_AUTOCAL` = 00 `PO_TIMEOUT` = 00
		`WORCTRL`	0x18	`EVENT1` = 001
		`MCSM2`	0x06	`RX_TIME_RSSI` = 0 `RX_TIME` = 110
	6	`MCSM0`	0x00	`FS_AUTOCAL` = 00 `PO_TIMEOUT` = 00
		`WORCTRL`	0x18	`EVENT1` = 001
		`MCSM2`	0x16	`RX_TIME_RSSI` = 1 `RX_TIME` = 110

Table 4. Register Settings Test Case 1 – 6

See Table 2 for a more detailed explanation of the different test cases.

5.2 Measurements

The following pseudo code shows the program flow when measuring the current consumption. Code for handling incoming packets and entering low power mode on the MCU is not implemented and must be taken care of by the application.

```
void main (void)
{
    // Init MCU
    // Reset Radio
    // Configure Radio
    // Calibrate the Radio
    // Enter WOR mode
    // Wait for a packet to be received
}
```

The average current consumption is given by Equation 1.

$$Current_{AVG}[us] = \frac{t_{Active}[us] \cdot Current_{Active}[uA] + \left(t_{polling}[us] - t_{Active}[us]\right) \cdot Current_{SLEEP}[uA]}{t_{polling}[us]}$$

Equation 1. Average Current Consumption

where:

t_{Active}: The time the CC2500/CC2510Fx is not in SLEEP state/PM2
$Current_{Active}$: Average current consumption in the period where the radio is not in SLEEP state/PM2 mode
$t_{Polling}$: The polling period t_{Event0} (1 s in this case)
$Current_{SLEEP}$: Current consumption when the CC2500/CC2510Fx is in SLEEP state/PM2 (these numbers are taken from the data sheets [3] and [5])

From the oscilloscope the XY waveform pairs were input to Excel for analysis. 2000 XY pairs where analyzed, given a resolution of 25 µs (X-axis) in test case 1 - 3 and 2.5 µs in test case 4 - 6.

5.2.1 Test Case 1

Figure 2 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state, while Figure 3 zooms in on the start of this period.

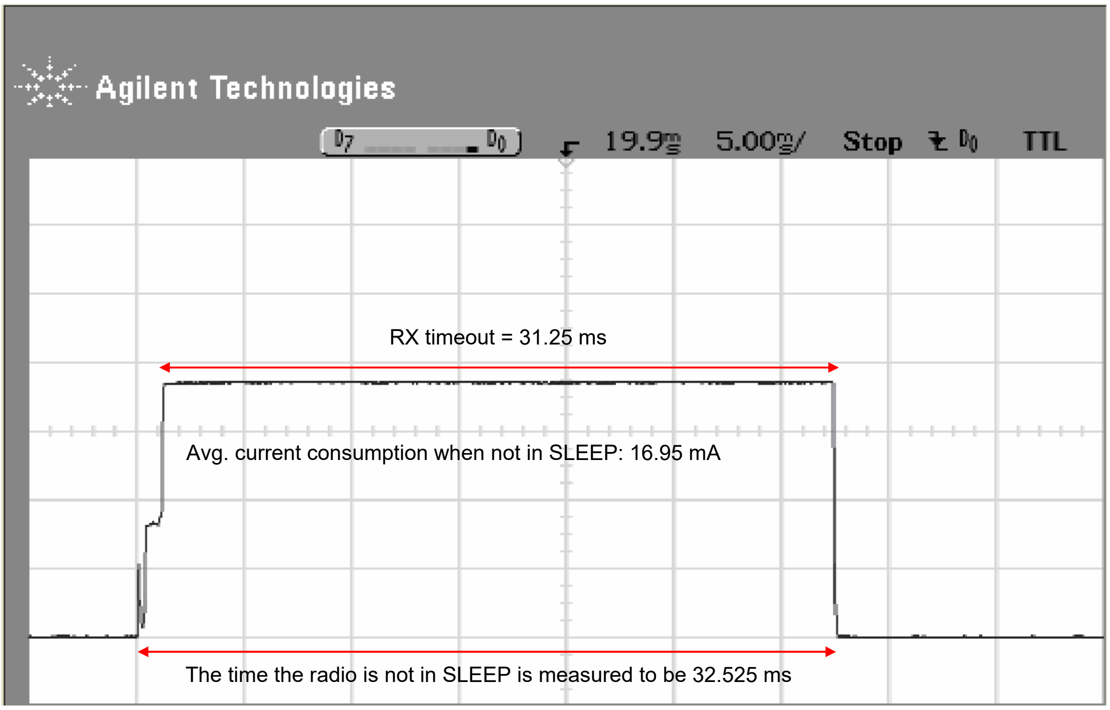


Figure 2. Current Consumption Test Case 1

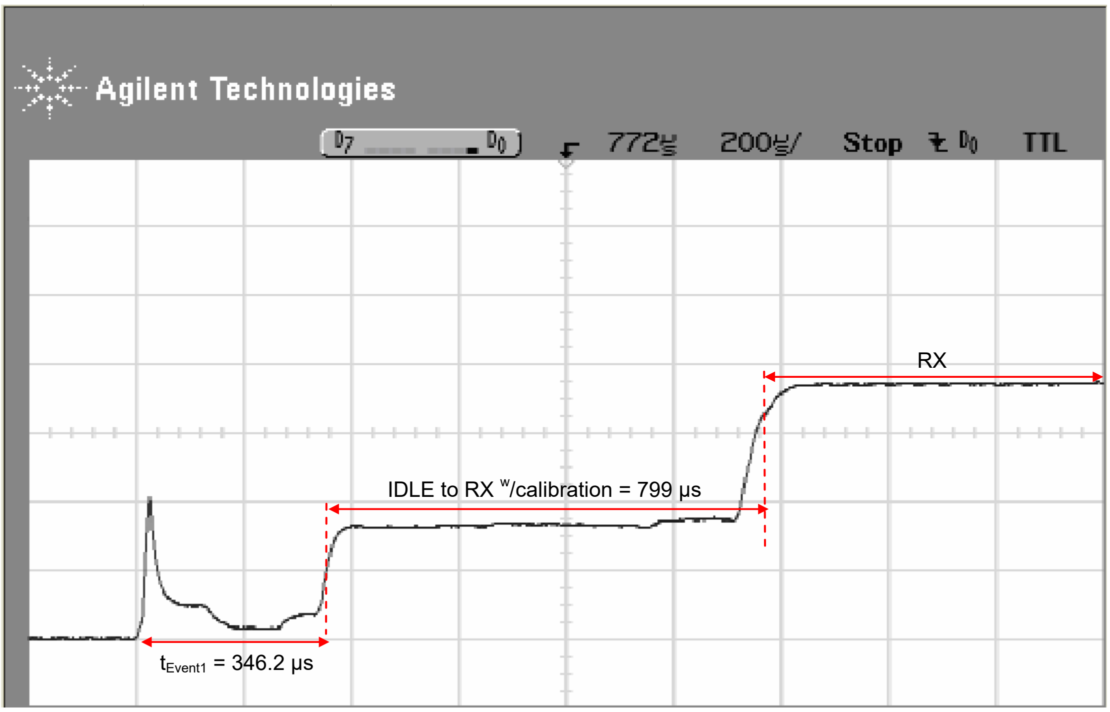


Figure 3. Current Consumption Test Case 1 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{32525 \cdot 16950 + \left(1000000 - 32525\right) \cdot 0.9}{1000000} = 552.2\ uA$$

Using the Excel sheet (see Figure 4), the current consumption was estimated to be 548.4 µA, giving an error in the estimation of about 0.69%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	2		RX	17300	uA
8		RX timeout in RCOSC periods	1083		Sleep state	0,5	uA
9		RX timeout period	0,031240	s	Extra WOR current	0,4	uA
10		RX duty cycle	3,124 %				
11							
12		**CHANBW_E**	1		Timeout setting from Register block		
13		**CHANBW_M**	3		4		
14		Channel filter bandwidth	232	kHz	6		
15		dec_tick frequency	0,464	MHz	8		
16					12		
17		Modulation type	FSK		16		
18					24		
19		**DRATE_E**	8		32		
20		**DRATE_M**	147		48		
21		Data rate	9,993	kbps			
22		datarate_pulse frequency	0,080	MHz			
23							
24		**WAIT_TIME**	1				
25		Waiting time at startup / after gain change	34	us			
26		**FILTER_LENGTH**	1				
27		Averaging length from channel filter (not ASK)	34	us			
28							
29		Delay from RX start to filter ready signal from demod.	125	us			
30							
31		RX response time to first valid carrier sense	263	us			
32							
33		Probability of carrier sense	100 %				
34		Average RX time	31 240	us	312	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	3				
44		tEvent1	346	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	5 100	196	1500	0,294	
53	0,000346	IDLE to RX, with calibration	20 768	799	7400	5,911	
54	0,001145	RX	812 250	31 240	17300	540,459	
55	0,032385	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,032385	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216	
57	0,032385	SLEEP state, RC oscillator running		967 615	0,9	0,871	
58	1	Average current consumption				548,382	

Figure 4. Estimated Current Consumption for Test Case 1

5.2.2 Test Case 2

Figure 5 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state, while Figure 6 zooms in on the start of this period.

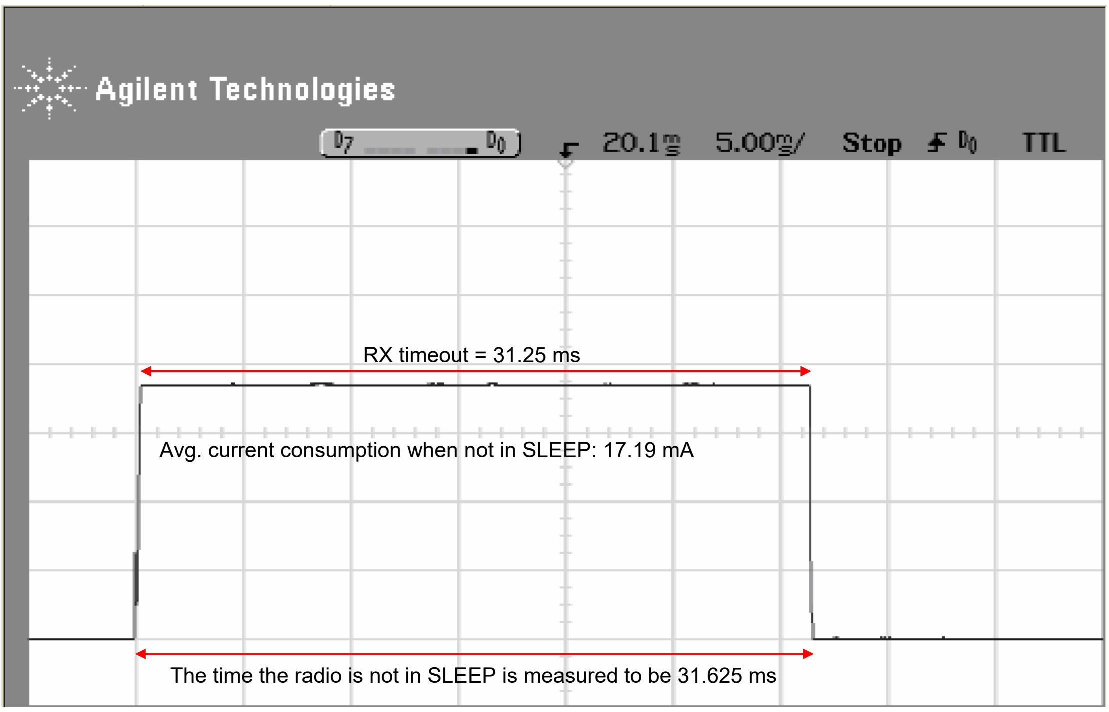


Figure 5. Current Consumption Test Case 2

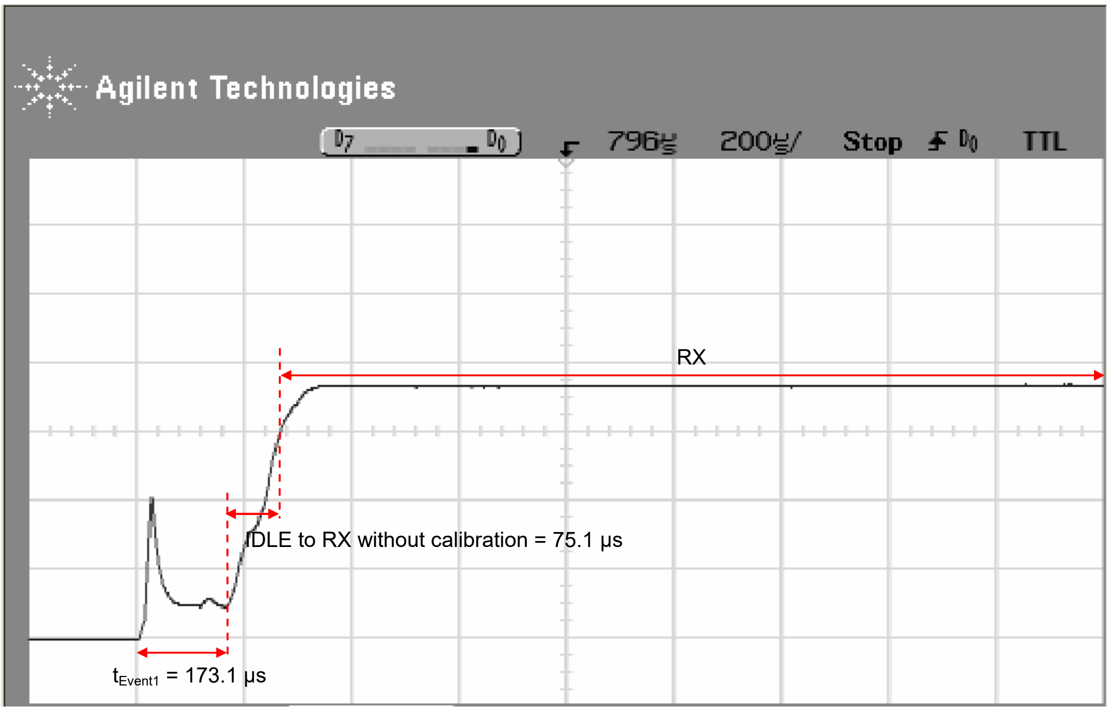


Figure 6. Current Consumption Test Case 2 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{31625 \cdot 17190 + (1000000 - 31625) \cdot 0.9}{1000000} = 544.5\ uA$$

Using the Excel sheet (see Figure 7), the current consumption was estimated to be 542.8 µA, giving an error in the estimation of about 0.31%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	2		RX	17300	uA
8		RX timeout in RCOSC periods	1083		Sleep state	0,5	uA
9		RX timeout period	0,031240	s	Extra WOR current	0,4	uA
10		RX duty cycle	3,124 %				
11							
12		**CHANBW_E**	1		Timeout setting from Register block		
13		**CHANBW_M**	3		4		
14		Channel filter bandwidth	232	kHz	6		
15		dec_tick frequency	0,464	MHz	8		
16					12		
17		Modulation type	FSK		16		
18					24		
19		**DRATE_E**	8		32		
20		**DRATE_M**	147		48		
21		Data rate	9,993	kbps			
22		datarate_pulse frequency	0,080	MHz			
23							
24		**WAIT_TIME**	1				
25		Waiting time at startup / after gain change	34	us			
26		**FILTER_LENGTH**	1				
27		Averaging length from channel filter (not ASK)	34	us			
28							
29		Delay from RX start to filter ready signal from demod.	125	us			
30							
31		RX response time to first valid carrier sense	263	us			
32							
33		Probability of carrier sense	100 %				
34		Average RX time	31 240	us	312	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	1				
44		tEvent1	173	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time [s]	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1500	0,035	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	812 250	31 240	17300	540,459	
55	0,031489	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,031489	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216	
57	0,031489	SLEEP state, RC oscillator running		968 511	0,9	0,872	
58	1	Average current consumption				542,768	

Figure 7. Estimated Current Consumption for Test Case 2

5.2.3 Test Case 3

Figure 8 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state, while Figure 9 zooms in on the start of this period.

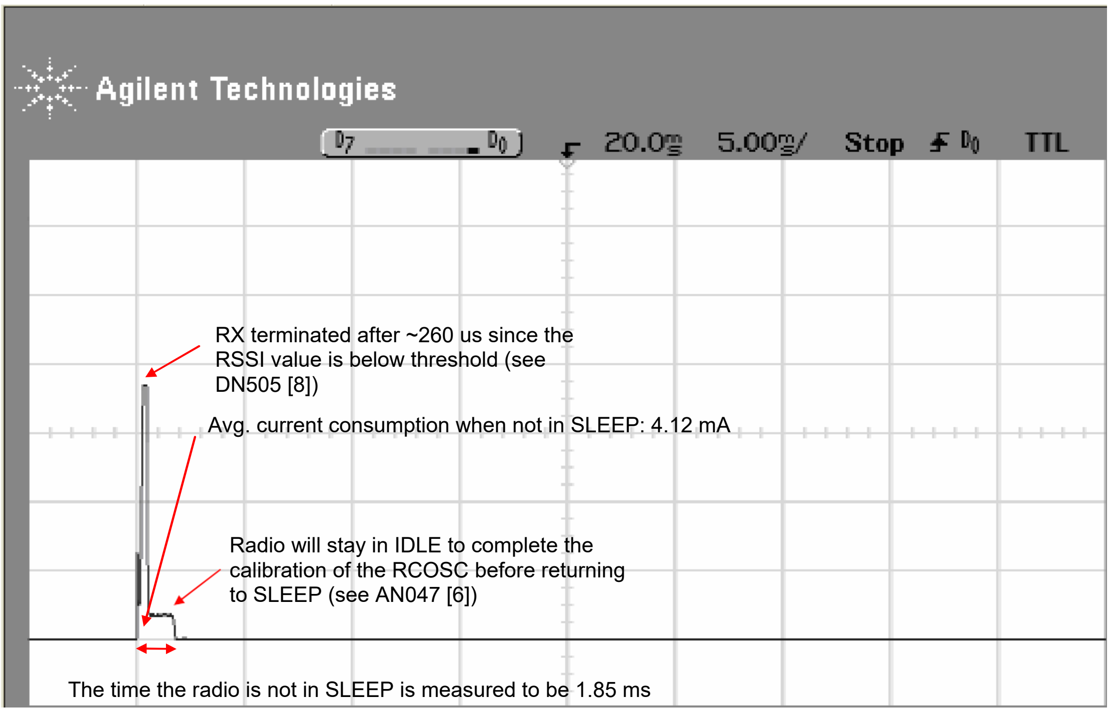


Figure 8. Current Consumption Test Case 3

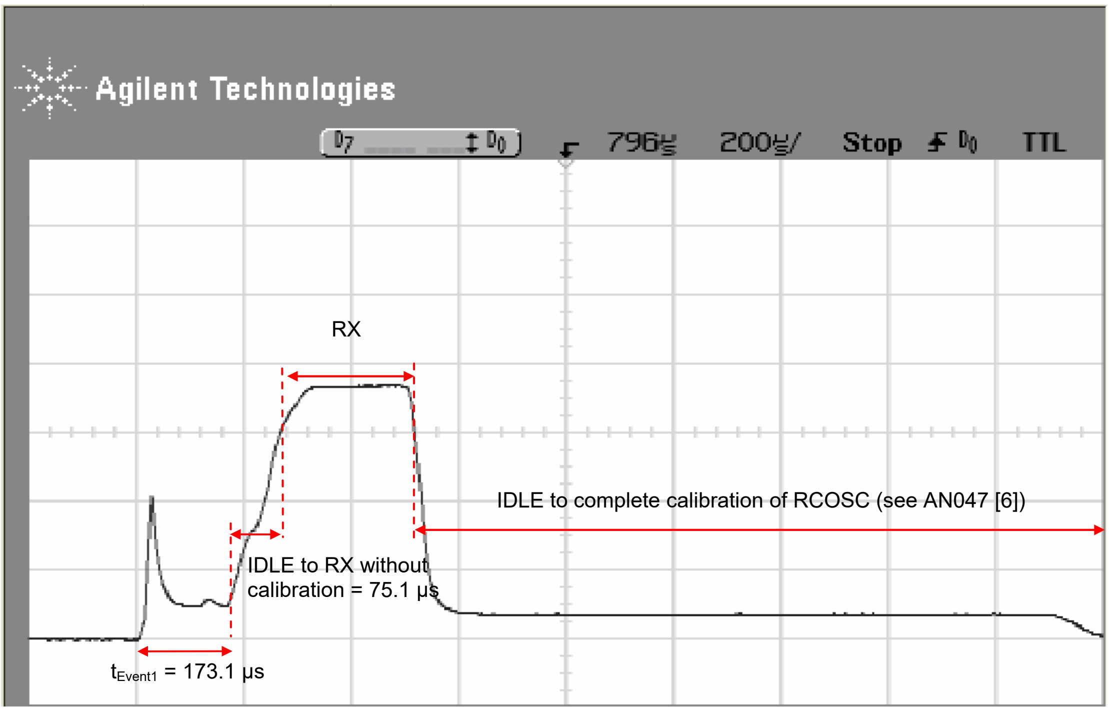


Figure 9. Current Consumption Test Case 3 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{4120 \cdot 1850 + (1000000 - 1850) \cdot 0.9}{1000000} = 8.5\ uA$$

Using the Excel sheet (see Figure 10), the current consumption was estimated to be 9.2 µA, giving an error in the estimation of about −7.61%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	2		RX	17300	uA
8		RX timeout in RCOSC periods	1083		Sleep state	0,5	uA
9		RX timeout period	0,031240	s	Extra WOR current	0,4	uA
10		RX duty cycle	3,124 %				
11							
12		**CHANBW_E**	1		Timeout setting from Register block		
13		**CHANBW_M**	3		4		
14		Channel filter bandwidth	232	kHz	6		
15		dec_tick frequency	0,464	MHz	8		
16					12		
17		Modulation type	FSK		16		
18					24		
19		**DRATE_E**	8		32		
20		**DRATE_M**	147		48		
21		Data rate	9,993	kbps			
22		datarate_pulse frequency	0,080	MHz			
23							
24		**WAIT_TIME**	1				
25		Waiting time at startup / after gain change	34	us			
26		**FILTER_LENGTH**	1				
27		Averaging length from channel filter (not ASK)	34	us			
28							
29		Delay from RX start to filter ready signal from demod.	125	us			
30							
31		RX response time to first valid carrier sense	263	us			
32							
33		Probability of carrier sense	0 %				
34		Average RX time	263	us	3	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	1				
44		tEvent1	173	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1500	0,035	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	6 836	263	17300	4,549	
55	0,000511	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,000511	Finishing RC oscillator recalibration before entering SLEEP	43 209	1 662	1500	2,493	
57	0,000511	SLEEP state, RC oscillator running		999 489	0,9	0,900	
58	1	Average current consumption				9,162	

Figure 10. Estimated Current Consumption for Test Case 3

The error in the estimation is mainly due to the time it takes to calibrate the RCOSC. The Excel sheet sets the calibration time to be 2 ms while it was measured to be 1.625 ms in test case 3. Changing this number (C46 changed to 1625), the estimated current consumption is 8.6 µA (see Figure 11), giving an error in the estimation of about −1.16%.

	A	B	C	D	E	F	G
45							
46		Average RC oscillator recalibration time	1 625	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1500	0,035	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	6 836	263	17300	4,549	
55	0,000511	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,000511	Finishing RC oscillator recalibration before entering SLEEP	33 459	1 287	1500	1,930	
57	0,000511	SLEEP state, RC oscillator running		999 489	0,9	0,900	
58	1	Average current consumption				8,600	

Figure 11. Estimated Current Consumption for Test Case 3 (C46 changed to 1625)

5.2.4 Comparing Test Case 1, 2, and 3

Figure 12 and Figure 13 shows the three test cases 1 - 3 in the same plot. Test case 2 has lower current consumption than test case 1 due to lack of calibration of the PLL, while test case 3 has much lower current consumption than the other two due to significantly reduced time spent in RX.

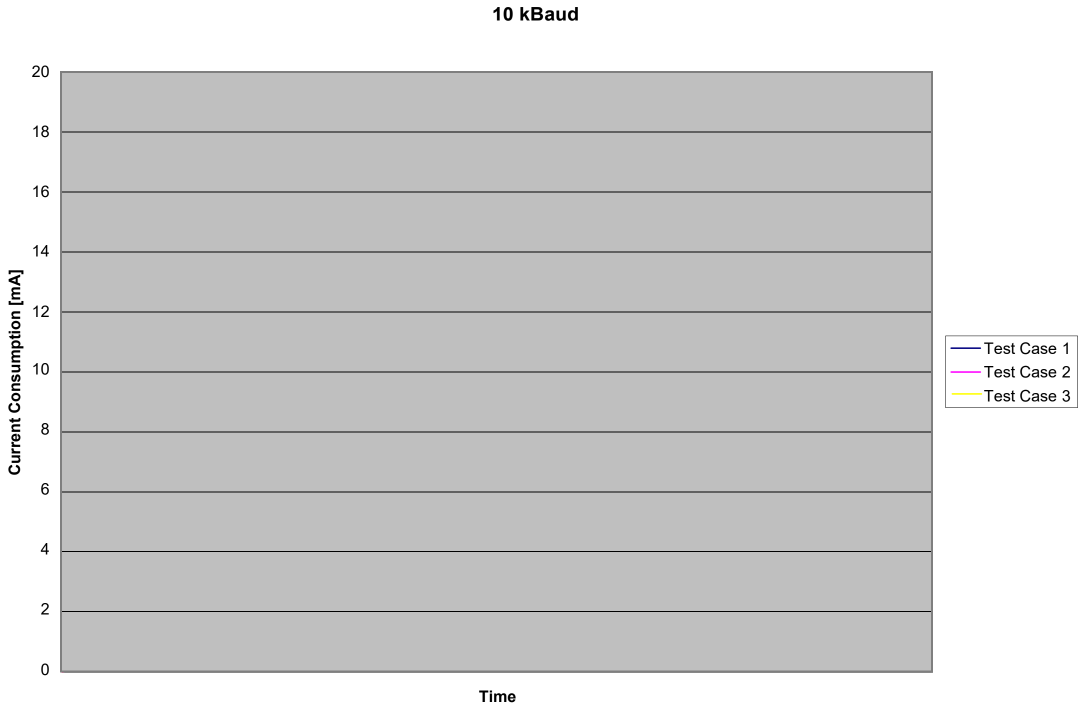


Figure 12. Test Case 1 - 3

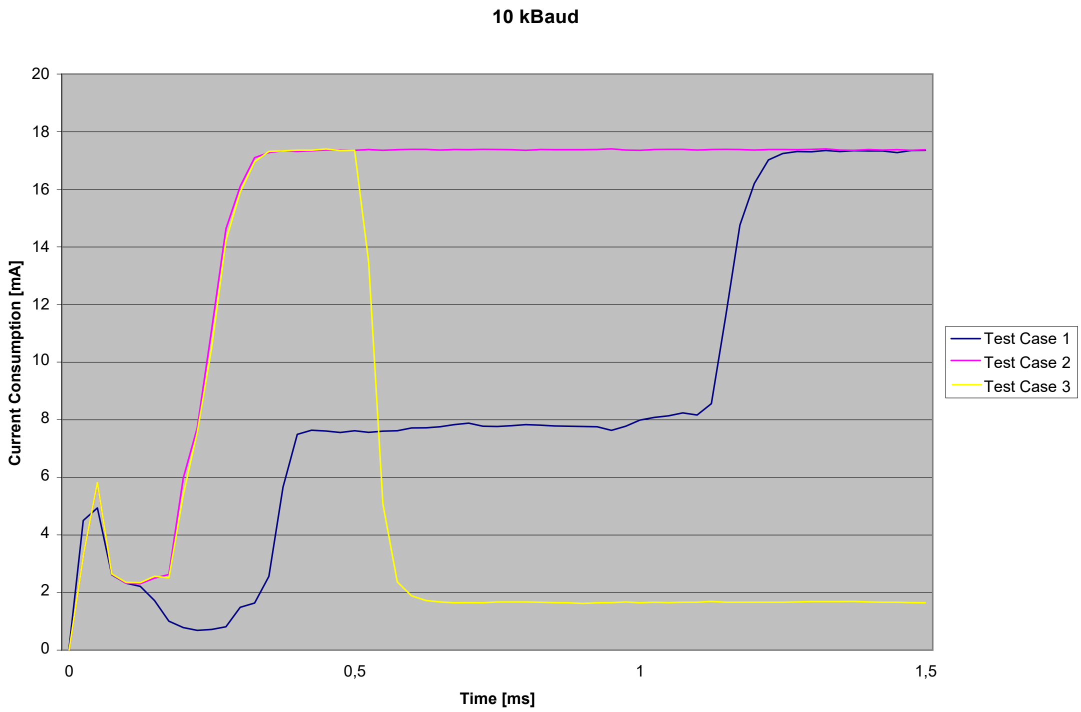


Figure 13. Test Case 1 - 3 (start-up sequence)

5.2.5 Test Case 4

Figure 14 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state.

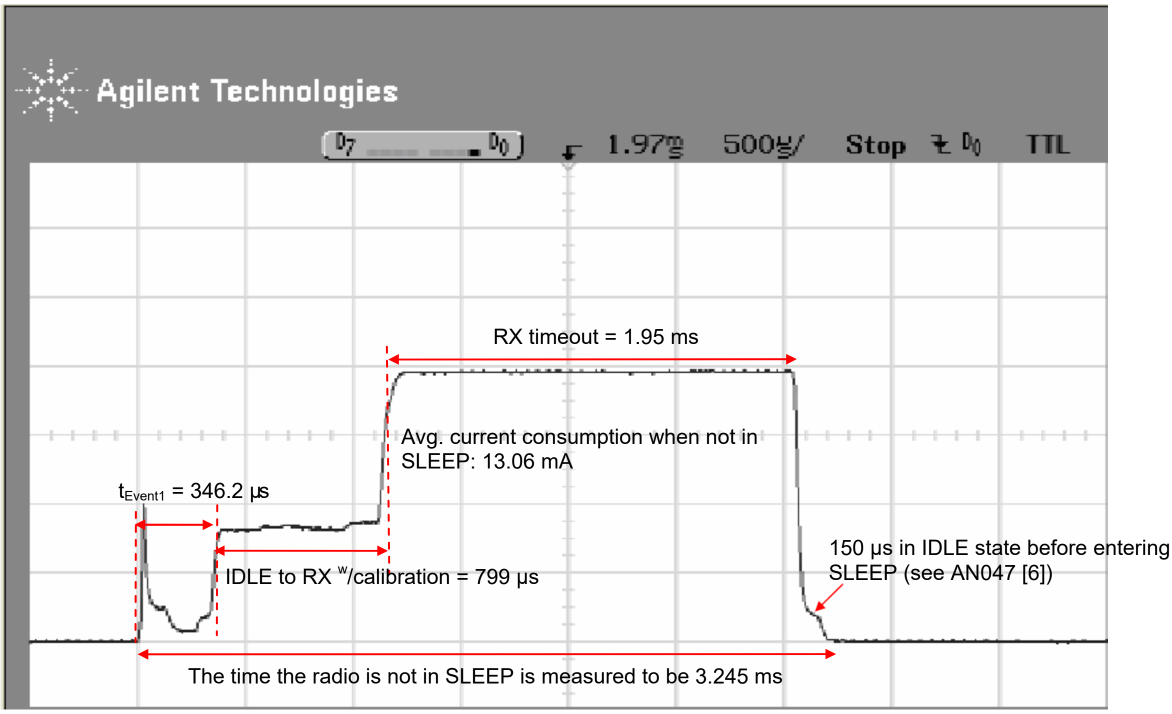


Figure 14. Current Consumption Test Case 4

This gives an average current consumption of

$$Current_{AVG} = \frac{3245 \cdot 13060 + (1000000 - 3245) \cdot 0.9}{1000000} = 43.3\ uA$$

Using the Excel sheet (see Figure 15), the current consumption was estimated to be 44.3 µA, giving an error in the estimation of about −2.26%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		WOR_RES	0		FS calibration	7400	uA
6		EVENT0	34666,66667		FS settling /FSTXON	7400	uA
7		RX_WAIT	6		RX	18800	uA
8		RX timeout in RCOSC periods	67		Sleep state	0,5	uA
9		RX timeout period	0,001933	s	Extra WOR current	0,4	uA
10		RX duty cycle	0,193 %				
11							
12		CHANBW_E	0		Timeout setting from Register block		
13		CHANBW_M	2		4		
14		Channel filter bandwidth	542	kHz	6		
15		dec_tick frequency	1,083	MHz	8		
16					12		
17		Modulation type	MSK		16		
18					24		
19		DRATE_E	13		32		
20		DRATE_M	59		48		
21		Data rate	249,939	kbps			
22		datarate_pulse frequency	2,000	MHz			
23							
24		WAIT_TIME	3				
25		Waiting time at startup / after gain change	30	us			
26		FILTER_LENGTH	0				
27		Averaging length from channel filter (not ASK)	7	us			
28							
29		Delay from RX start to filter ready signal from demod.	18	us			
30							
31		RX response time to first valid carrier sense	92	us			
32							
33		Probability of carrier sense	100 %				
34		Average RX time	1 933	us	483	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		EVENT1	3				
44		tEvent1	346	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	5 100	196	1500	0,294	
53	0,000346	IDLE to RX, with calibration	20 768	799	7400	5,911	
54	0,001145	RX	50 250	1 933	18800	36,335	
55	0,003078	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,003078	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216	
57	0,003078	SLEEP state, RC oscillator running		996 922	0,9	0,897	
58	1	Average current consumption				44,284	

Figure 15. Estimated Current Consumption for Test Case 4

The error in the estimation is mainly due to the RX current consumption. The Excel sheet sets this to be 18.8 mA, while it was measured to be 18.4 mA in test case 4. If this current is changed (F7 changed to 18400), the estimated current consumption is
43.5 µA (see Figure 16), giving an error in the estimation of about −0.46%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		WOR_RES	0		FS calibration	7400	uA
6		EVENT0	34666,66667		FS settling /FSTXON	7400	uA
7		RX_TIME	6		RX	18400	uA
8		RX timeout in RCOSC periods	67		Sleep state	0,5	uA
9		RX timeout period	0,001933	s	Extra WOR current	0,4	uA

.
.
.

	A	B	C	D	E	F
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180
51	-	Startup of XOSC	3 900	150	3000	0,450
52	0,000150	Waiting for EVENT1	5 100	196	1500	0,294
53	0,000346	IDLE to RX, with calibration	20 768	799	7400	5,911
54	0,001145	RX	50 250	1 933	18400	35,562
55	0,003078	RX to IDLE, no calibration	2	0	7400	0,001
56	0,003078	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216
57	0,003078	SLEEP state, RC oscillator running		996 922	0,9	0,897
58	1	Average current consumption				43,511

Figure 16. Estimated Current Consumption for Test Case 4 (F7 changed to 18400)

5.2.6 Test Case 5

Figure 17 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state.

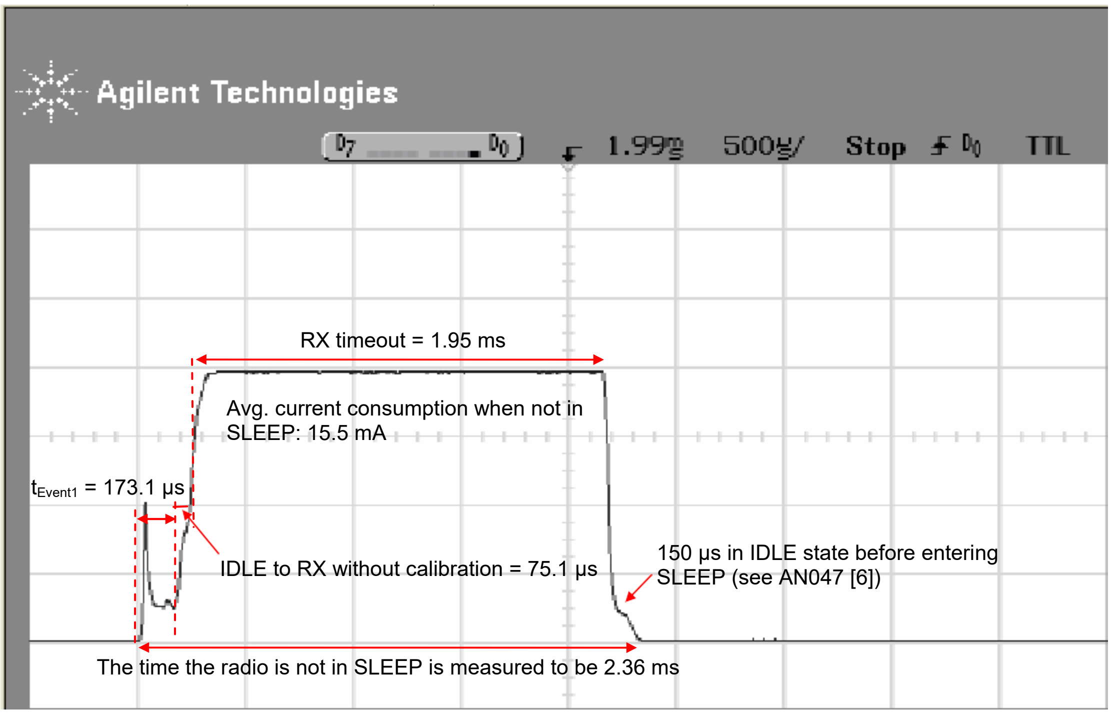


Figure 17. Current Consumption Test Case 5

This gives an average current consumption of

$$Current_{AVG} = \frac{2360 \cdot 15500 + (1000000 - 2360) \cdot 0.9}{1000000} = 37.5\ uA$$

Using the Excel sheet (see Figure 18), the current consumption was estimated to be 38.7 µA, given an error in the estimation of about −3.1%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	6		RX	18800	uA
8		RX timeout in RCOSC periods	67		Sleep state	0,5	uA
9		RX timeout period	0,001933	s	Extra WOR current	0,4	uA
10		RX duty cycle	0,193 %				
11							
12		**CHANBW_E**	0		Timeout setting from Register block		
13		**CHANBW_M**	2		4		
14		Channel filter bandwidth	542	kHz	6		
15		dec_tick frequency	1,083	MHz	8		
16					12		
17		Modulation type	MSK		16		
18					24		
19		**DRATE_E**	13		32		
20		**DRATE_M**	59		48		
21		Data rate	249,939	kbps			
22		datarate_pulse frequency	2,000	MHz			
23							
24		**WAIT_TIME**	3				
25		Waiting time at startup / after gain change	30	us			
26		**FILTER_LENGTH**	0				
27		Averaging length from channel filter (not ASK)	7	us			
28							
29		Delay from RX start to filter ready signal from demod.	18	us			
30							
31		RX response time to first valid carrier sense	92	us			
32							
33		Probability of carrier sense	100 %				
34		Average RX time	1 933	us	483	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	1				
44		tEvent1	173	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1500	0,035	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	50 250	1 933	18800	36,335	
55	0,002181	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,002181	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216	
57	0,002181	SLEEP state, RC oscillator running		997 819	0,9	0,898	
58	1	Average current consumption				38,670	

Figure 18. Estimated Current Consumption for Test Case 5

Also in this case, the error in the estimation is mainly due to the RX current consumption (see Section 5.2.5). Setting it to 18.4 mA instead of 18.8 mA, the estimated current consumption will be 37.9 µA (see Figure 19), giving an error in the estimation of about −1.06%.

49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180
51	-	Startup of XOSC	3 900	150	3000	0,450
52	0,000150	Waiting for EVENT1	600	23	1500	0,035
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556
54	0,000248	RX	50 250	1 933	18400	35,562
55	0,002181	RX to IDLE, no calibration	2	0	7400	0,001
56	0,002181	Finishing RC oscillator recalibration before entering SLEEP	3 750	144	1500	0,216
57	0,002181	SLEEP state, RC oscillator running		997 819	0,9	0,898
58	1	Average current consumption				37,897

Figure 19. Estimated Current Consumption for Test Case 5 (F7 changed to 18400)

5.2.7 Test Case 6

Figure 20 shows the current consumption for the complete period of time when the CC2500 is not in SLEEP state.

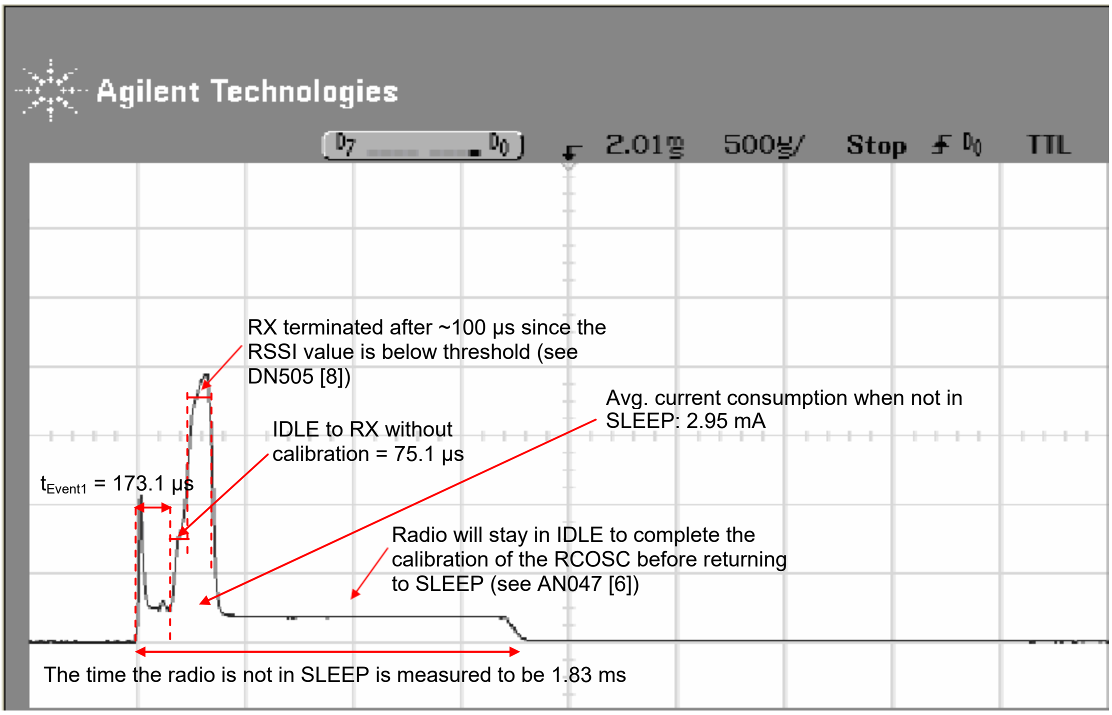


Figure 20. Current Consumption Test Case 6

This gives an average current consumption of

$$Current_{AVG} = \frac{1830 \cdot 2950 + (1000000 - 1830) \cdot 0.9}{1000000} = 6.3\ uA$$

Using the Excel sheet (see Figure 21), the current consumption was estimated to be 6.6 µA, given an error in the estimation of about −4.55%.

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1500	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	6		RX	18800	uA
8		RX timeout in RCOSC periods	67		Sleep state	0,5	uA
9		RX timeout period	0,001933	s	Extra WOR current	0,4	uA
10		RX duty cycle	0,193 %				
11							
12		**CHANBW_E**	0		Timeout setting from Register block		
13		**CHANBW_M**	2		4		
14		Channel filter bandwidth	542	kHz	6		
15		dec_tick frequency	1,083	MHz	8		
16					12		
17		Modulation type	MSK		16		
18					24		
19		**DRATE_E**	13		32		
20		**DRATE_M**	59		48		
21		Data rate	249,939	kbps			
22		datarate_pulse frequency	2,000	MHz			
23							
24		**WAIT_TIME**	3				
25		Waiting time at startup / after gain change	30	us			
26		**FILTER_LENGTH**	0				
27		Averaging length from channel filter (not ASK)	7	us			
28							
29		Delay from RX start to filter ready signal from demod.	18	us			
30							
31		RX response time to first valid carrier sense	92	us			
32							
33		Probability of carrier sense	0 %				
34		Average RX time	92	us	23	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	1				
44		tEvent1	173	us			
45							
46		Average RC oscillator recalibration time	2 000	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1500	0,035	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	2 400	92	18800	1,735	
55	0,000341	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,000341	Finishing RC oscillator recalibration before entering SLEEP	47 645	1 833	1500	2,749	
57	0,000341	SLEEP state, RC oscillator running		999 659	0,9	0,900	
58	1	Average current consumption				6,605	

Figure 21. Estimated Current Consumption for Test Case 6

The error in the estimation is mainly due to the time it takes to calibrate the RCOSC in addition to the IDLE current consumption. The Excel sheet sets the calibration time to be 2 ms and the IDLE current consumption to 1.5 mA while it was measured to be 1.625 ms and 1.7 mA respectively. Changing these numbers (C46 changed to 1600 and F4 changed to 1700, in addition to changing the RX current consumption as in Test Case 4 and 5), the estimated current consumption is 6.3 µs (see Figure 22).

	A	B	C	D	E	F	G
1							
2		Crystal frequency	26	MHz	Current consumption:		
3		Polling period	1	s	XOSC startup current	3000	uA
4		Total event 0 count	34666,66667		IDLE	1700	uA
5		**WOR_RES**	0		FS calibration	7400	uA
6		**EVENT0**	34666,66667		FS settling /FSTXON	7400	uA
7		**RX_WAIT**	6		RX	18400	uA
8		RX timeout in RCOSC periods	67		Sleep state	0,5	uA
9		RX timeout period	0,001933	s	Extra WOR current	0,4	uA
10		RX duty cycle	0,193 %				
11							
12		**CHANBW_E**	0		Timeout setting from Register block		
13		**CHANBW_M**	2		4		
14		Channel filter bandwidth	542	kHz	6		
15		dec_tick frequency	1,083	MHz	8		
16					12		
17		Modulation type	MSK		16		
18					24		
19		**DRATE_E**	13		32		
20		**DRATE_M**	59		48		
21		Data rate	249,939	kbps			
22		datarate_pulse frequency	2,000	MHz			
23							
24		**WAIT_TIME**	3				
25		Waiting time at startup / after gain change	30	us			
26		**FILTER_LENGTH**	0				
27		Averaging length from channel filter (not ASK)	7	us			
28							
29		Delay from RX start to filter ready signal from demod.	18	us			
30							
31		RX response time to first valid carrier sense	92	us			
32							
33		Probability of carrier sense	0 %				
34		Average RX time	92	us	23	bit periods	
35							
36		RX to IDLE, no calibration	2	cycles			
37		RX to IDLE, including calibration	18 817	cycles			
38		Average cycles RX to IDLE, one of four calibration	4 706	cycles			
39							
40		IDLE to RX, no calibration	1 953				
41		IDLE to RX, with calibration	20 768				
42							
43		**EVENT1**	1				
44		tEvent1	173	us			
45							
46		Average RC oscillator recalibration time	1625	us			
47		RC oscillator recalibration abort time	3 750	cycles	144	us	
48							
49	Time (s)	Description	Cycles	Time [us]	Current [uA]	Contribution [uA]	
50	-	Charge 100nF decoupling to 1.8V	-	-		0,180	
51	-	Startup of XOSC	3 900	150	3000	0,450	
52	0,000150	Waiting for EVENT1	600	23	1700	0,039	
53	0,000173	IDLE to RX, no calibration	1 953	75	7400	0,556	
54	0,000248	RX	2 400	92	18400	1,698	
55	0,000341	RX to IDLE, no calibration	2	0	7400	0,001	
56	0,000341	Finishing RC oscillator recalibration before entering SLEEP	37 895	1 458	1700	2,478	
57	0,000341	SLEEP state, RC oscillator running		999 659	0,9	0,900	
58	1	Average current consumption				6,302	

Figure 22. Estimated Current Consumption for Test Case 6 (F7 changed to 18400, F4 changed to 1700, and C46 changed to 1625)

5.2.8 Comparing Test Case 4, 5, and 6

Figure 23 shows the three test cases 4 - 6 in the same plot. The result is the same as discussed in Section 5.2.4 (test case 4 corresponds to test case 1, 5 to 2, and 6 to 3 respectively).

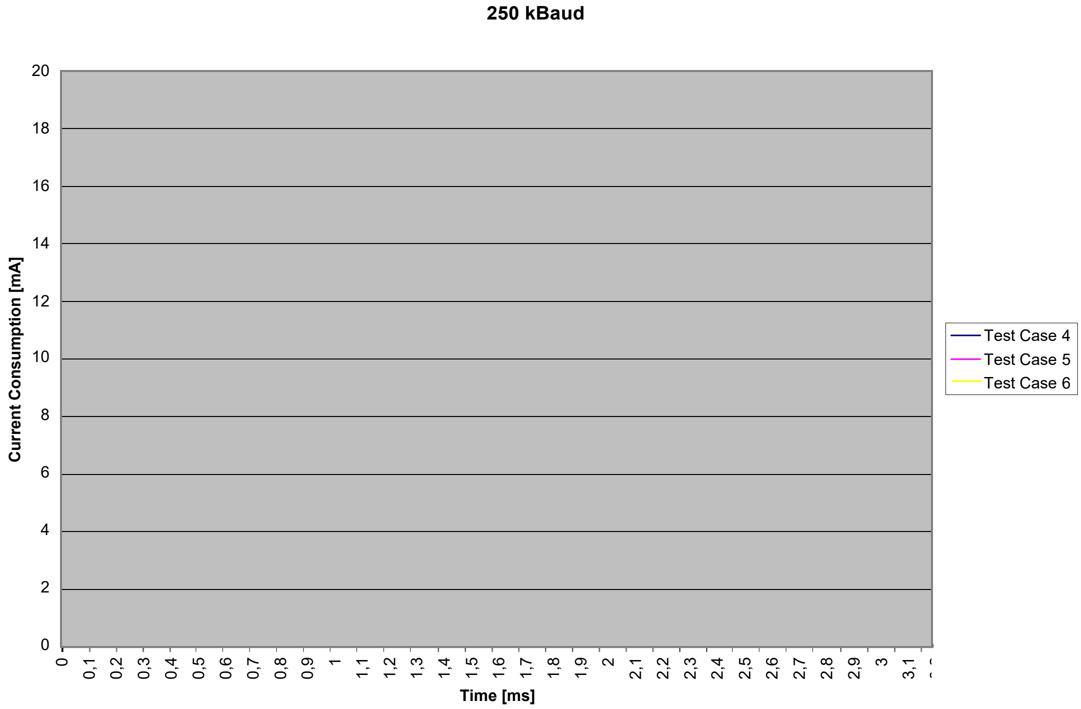


Figure 23. Test Case 4 - 6

6 CC2510

6.1 Configuring the CC2510

To be able to reduce the power consumption on the CC2510, the system clock speed should be reduced from its max value, f_{XOSC} (26 MHz). Be aware that reducing the system clock speed not only reduces the current consumption, but also increases the transition times between different radio states (see DN110 [10] for more details).

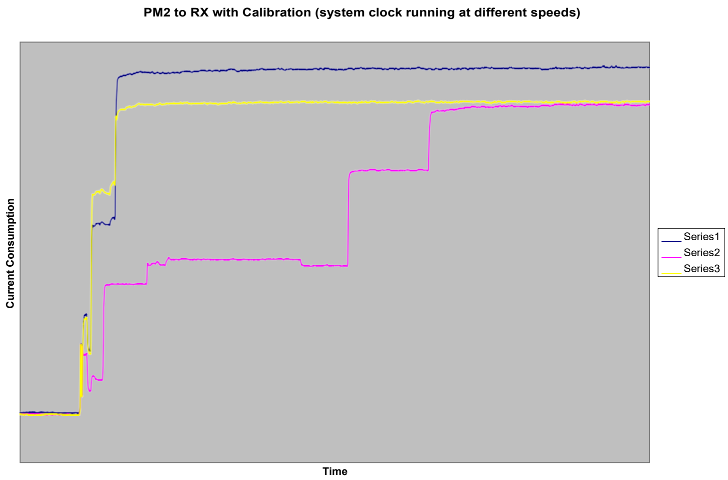


Figure 24. PM2 to RX with Calibration (system clock running at different speeds)

Figure 24 shows the current consumption when waking up from PM2 and entering RX state. The program flow for the three different series is shown in Table 5.

Series	Description
1	Wake up from PM2 and turn on the HSXOSC (system clock speed set to 26 MHz)
	Strobe RX
	Enter PM0 (CPU is halted)
2	Wake up from PM2 and turn on the HSXOSC (system clock speed set to 203.125 kHz)
	Strobe RX
	Enter PM0 (CPU is halted)
3	Wake up from PM2 and turn on the HSXOSC (system clock speed set to 26 MHz)
	Strobe RX
	Wait for radio to enter RX state (`MARCSTATE` = 0x0D)
	Change system clock speed to 203.125 kHz
	Enter PM0

Table 5. Pseudo Code for Test Code running when generating the Current Consumption Plots shown in Figure 24

From Figure 24 one can see that even if the RX current consumption is reduced when the system clock speed is 203.125 kHz (series2) compared to when it is 26 MHz (series1), the transition time is about 10 times as long, and hence the average current consumption will increase instead of decrease when reducing the system clock speed. A solution will be to let the system clock run at 26 MHz until RX state is reached and the turn it down to 203.125 kHz (series3). Even if both series1 and series3 have the system clock speed set to 26 MHz during the transition to RX state, series3 will have higher current consumption during the transition, since the CPU must be in active mode to poll `MARCSTATE` instead of being halted (PM0), like is the case with series1.

Figure 25 shows how the currents consumption decreases and the transition time increases with decreasing system clock speed.

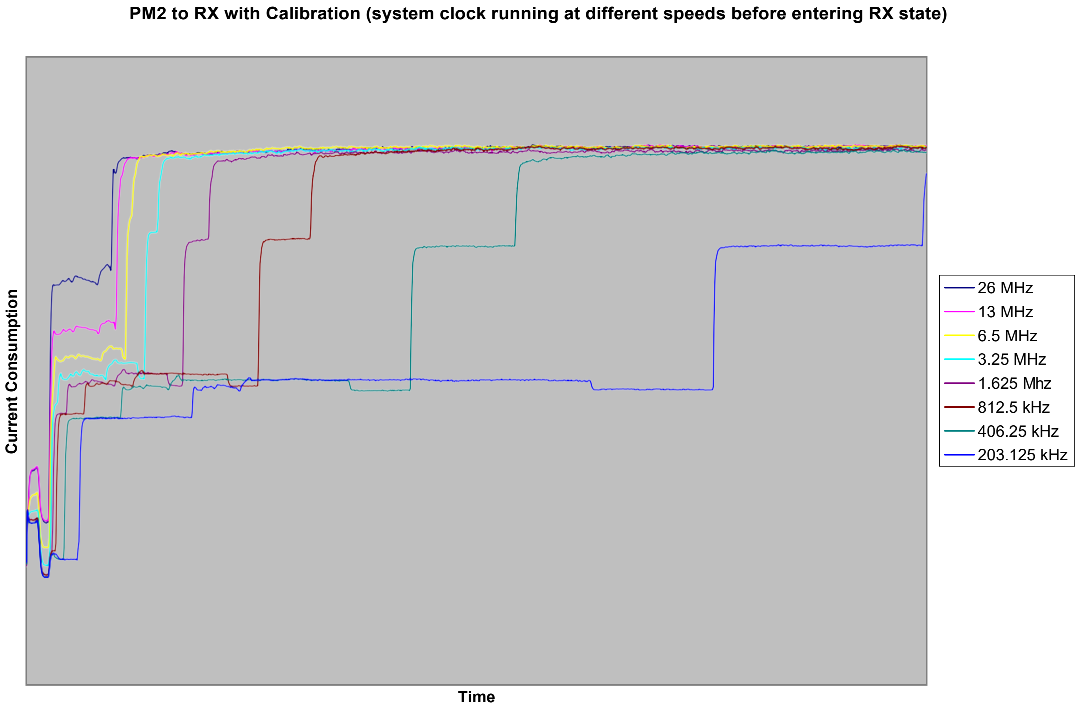


Figure 25. PM2 to RX with Calibration (different system clock speeds used before entering RX mode)

Since the RX time is constant, as is the polling interval, the system with the longest transition time will be the one that spends the least time in PM2. Analyzing the above current plot in Excel shows that running the system clock at 13 MHz until RX state is reached and then switch to 203.125 kHz will be the best solution when trying to reduce the current consumption as much as possible (running the system clock at 26 MHz gives a slightly higher current consumption). However, it is not necessarily the case that running the system clock at 13 MHz and then reduce it to 203.125 kHz will always give the lowest current consumption. Figure 26 shows the case where no calibration is performed when going from IDLE to RX and RX state is terminated if the RSSI is below a certain threshold. In this particular case, running the system clock at 26 MHz before RX mode is entered gives a lower current consumption than if the system clock runs at 13 MHz.

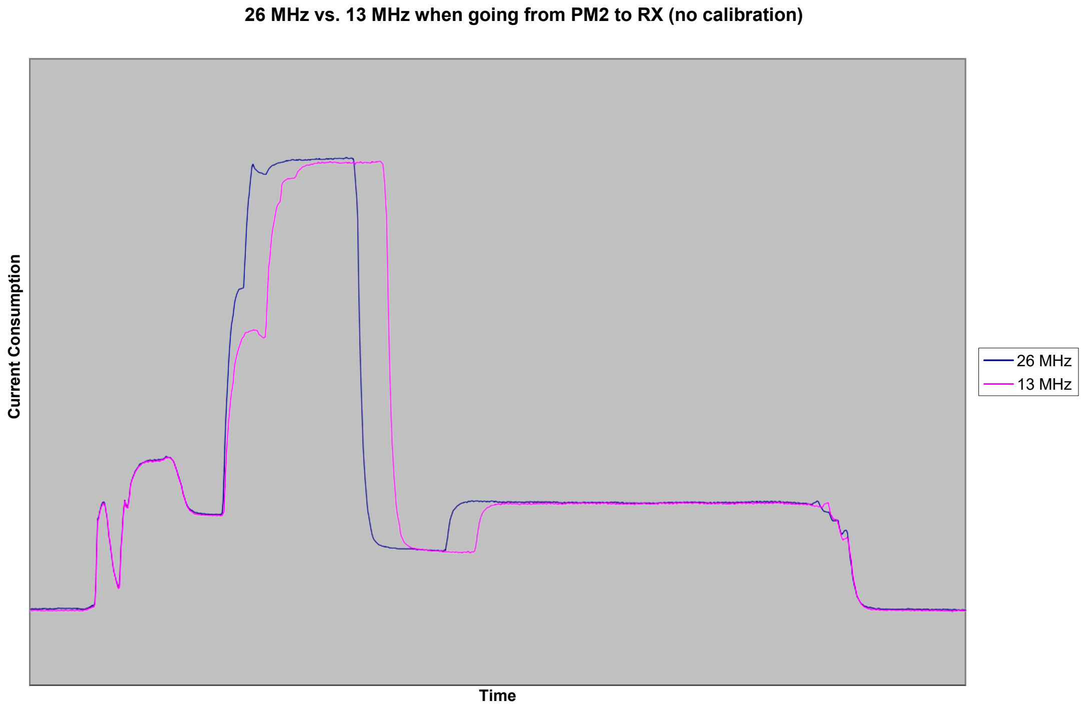


Figure 26. 26 MHz vs. 13 MHz when going from PM2 to RX without Calibration

6.2 Radio Configuration for CC2510

For both data rates, preferred settings (optimized for sensitivity) from SmartRF™ Studio [7] , version 6.9.2.0, are used. Two other registers are modified for the different test cases, and these registers are `MCSM0` and `MCSM2` (see Table 6 for details).

Register	Register Field	Comment	
`MCSM0`	`FS_AUTOCAL`	01	The PLL is calibrated every time before the radio enters RX mode
		00	The PLL is not being calibrated (manual calibration must be performed at a given time interval depending on the environment the system operates in)
`MCSM2`	`RX_TIME_RSSI`	0	Radio will go back to IDLE when the RX timeout expires
		1	Radio will go back to IDLE if the RSSI level is below a certain threshold. Please see DN505 [8] for details on how to estimate this time for the different register settings
	`RX_TIME`	010	Duty cycle = 3.125% and hence RX timeout = 31.25 ms (used when the data rate is 10 kBaud, since minimum RX timeout is 20.8 ms (see Section 3)).
		110	Duty cycle = 0.195% and hence RX timeout = 1.95 ms (used when the data rate is 250 kBaud, since minimum RX timeout is 832 µs (see Section 3)).

Table 6. Registers Modified for the Different Test Cases (7 - 12)

6.2.1 Register Settings for the Different CC2510 Test Cases (Test Case 7 - 12)

Register settings for the different test cases are shown in Table 7.

Data Rate [kBaud]	Test Case #	Register	Reg. Setting	Comment
10	7	MCSM0	0x14	FS_AUTOCAL = 01
		MCSM2	0x02	RX_TIME_RSSI = 0 RX_TIME = 010
	8	MCSM0	0x04	FS_AUTOCAL = 00
		MCSM2	0x02	RX_TIME_RSSI = 0 RX_TIME = 010
	9	MCSM0	0x04	FS_AUTOCAL = 00
		MCSM2	0x12	RX_TIME_RSSI = 1 RX_TIME = 010
250	10	MCSM0	0x14	FS_AUTOCAL = 01
		MCSM2	0x06	RX_TIME_RSSI = 0 RX_TIME = 110
	11	MCSM0	0x04	FS_AUTOCAL = 00
		MCSM2	0x06	RX_TIME_RSSI = 0 RX_TIME = 110
	12	MCSM0	0x04	FS_AUTOCAL = 00
		MCSM2	0x16	RX_TIME_RSSI = 1 RX_TIME = 110

Table 7. Register Settings Test Case 7 - 12

In test case 7 and 10 the system clock will run at 13 MHz until the radio is in RX state and will then be switched to 203.125 kHz and 1.625 MHz[4] respectively. In test case 8, 9, 11, and 12 (when FS_AUTOCAL=00) the system clock will run at 26 MHz until the radio reaches RX state and then be switched to 203.125 kHz and 1.625 MHz[4] respectively. See Table 2 for a more detailed explanation of the different test cases.

[4] When using a data rate of 10 kBaud (test case 7 - 9) the minimum system clock speed is 203.125 kHz. When using 250 kBaud (test case 10 - 12), the minimum system clock speed is 1.625 MHz. Please see the SoC data sheets ([4] and [5]) for more details.

6.3 Measurements

The following pseudo code shows the program flow when measuring the current consumption. Code for handling incoming packets is not implemented and must be taken care of by the application.

```
void main (void)
{
    // Configure Radio
    // Calibrate the Radio
    // Turn on the HSXOSC and turn off the HSRCOSC (System clock running at
    // 13 MHz for test case 7 and 10 and at 26 MHz for test case 8, 9, 11, and 12)
    while (TRUE)
    {
        // Enable RX timeout interrupt
        // Strobe RX
        // Wait for radio to enter RX state
        // Set system clock to run at 203.125 kHz (test case 7, 8, and 9) or 1.625
        // MHz (test case 10, 11, and 12)
        // Enter PM0 (continue from next line after an RX timeout interrupt occur)
        // Disable RX timeout interrupt and enable Sleep timer interrupt
        // Enter PM2 (this must be done according to DN106 [9])
    }
}
//---------------------------------------------------------------------------------
#pragma vector=RF_VECTOR
__interrupt void rf_IRQ(void)
{
    // Clear interrupt flag
    // Turn on the HSRCOSC and turn off the HSXOSC (System clock running at
    // 13 MHz)
}
//---------------------------------------------------------------------------------
#pragma vector=ST_VECTOR
__interrupt void st_IRQ(void)
{
    // Enable Flash Cache
    // Clear interrupt flags
    // Turn on the HSXOSC and turn off the HSRCOSC (System clock running at
    // 13 MHz for test case 7 and 10 and at 26 MHz for test case 8, 9, 11, and 12)
    // Disable Sleep timer interrupt
    // Clear the MODE bits
}
```

From the oscilloscope the XY waveform pairs were input to Excel for analysis. 2000 XY pairs where analyzed, given a resolution of 25 µs (X-axis) in test case 7 - 9 and 2.5 µs in test case 10 - 12.

6.3.1 Test Case 7

Figure 27 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2, while Figure 28 zooms in on the start of this period.

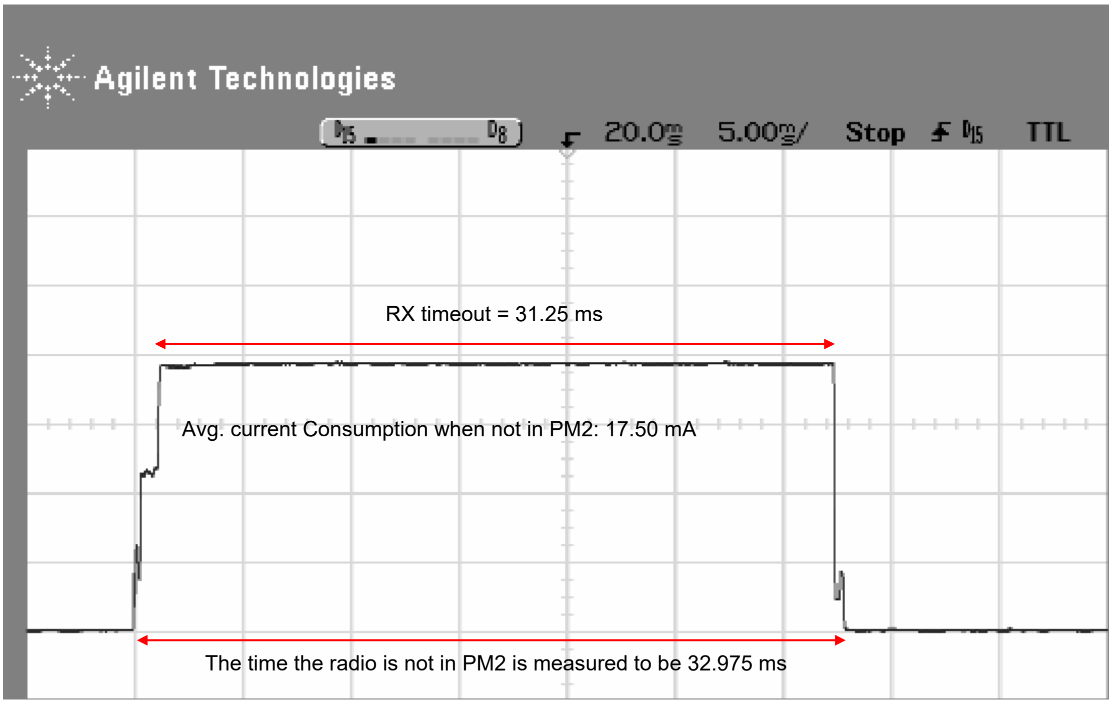


Figure 27. Current Consumption Test Case 7

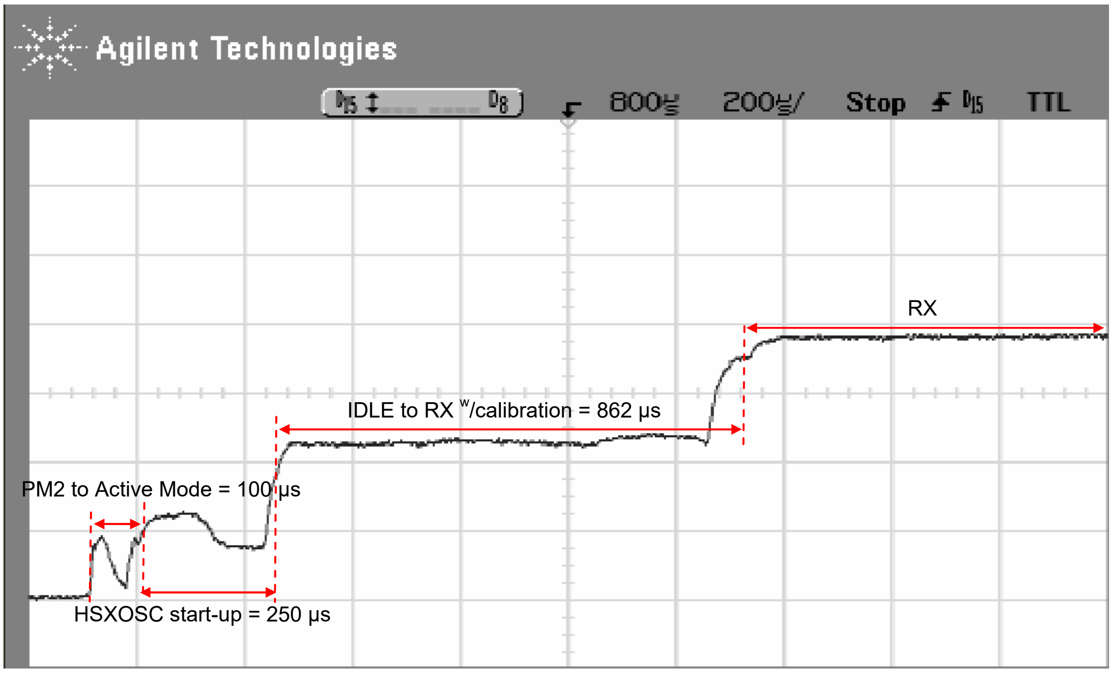


Figure 28. Current Consumption Test Case 7 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{32975 \cdot 17500 + (1000000 - 32975) \cdot 0.5}{1000000} = 577.5\ uA$$

6.3.2 Test Case 8

Figure 29 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2, while Figure 30 zooms in on the start of this period.

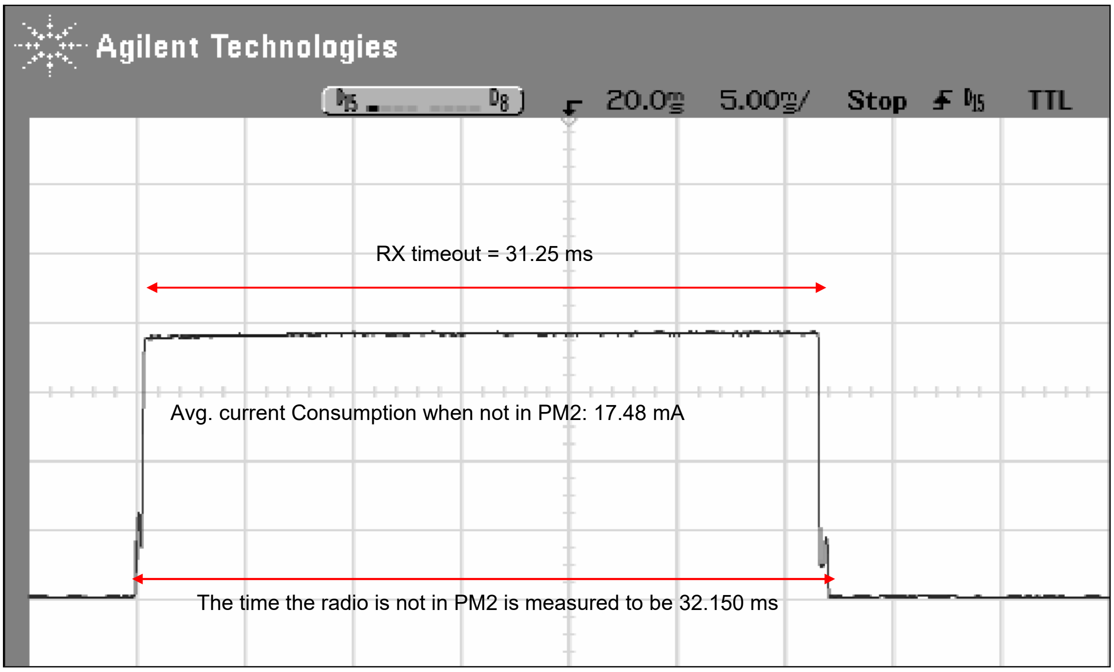


Figure 29. Current Consumption Test Case 8

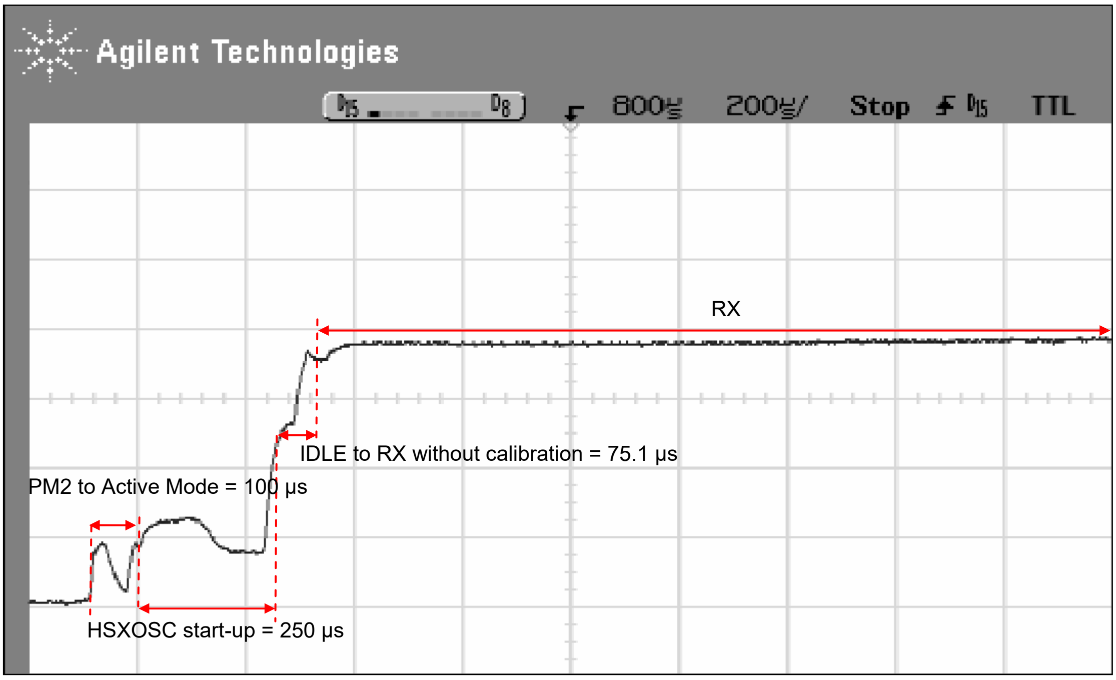


Figure 30. Current Consumption Test Case 8 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{32150 \cdot 17480 + (1000000 - 32150) \cdot 0.5}{1000000} = 562.5\ uA$$

6.3.3 Test Case 9

Figure 31 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2, while Figure 32 zooms in on the start of this period.

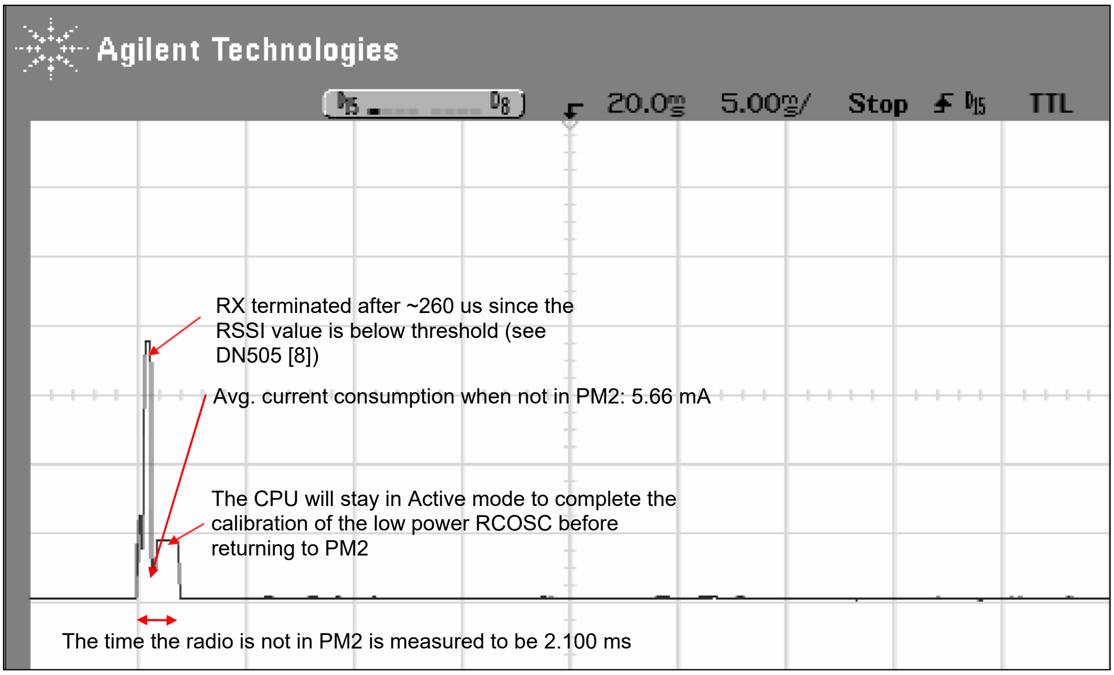


Figure 31. Current Consumption Test Case 9

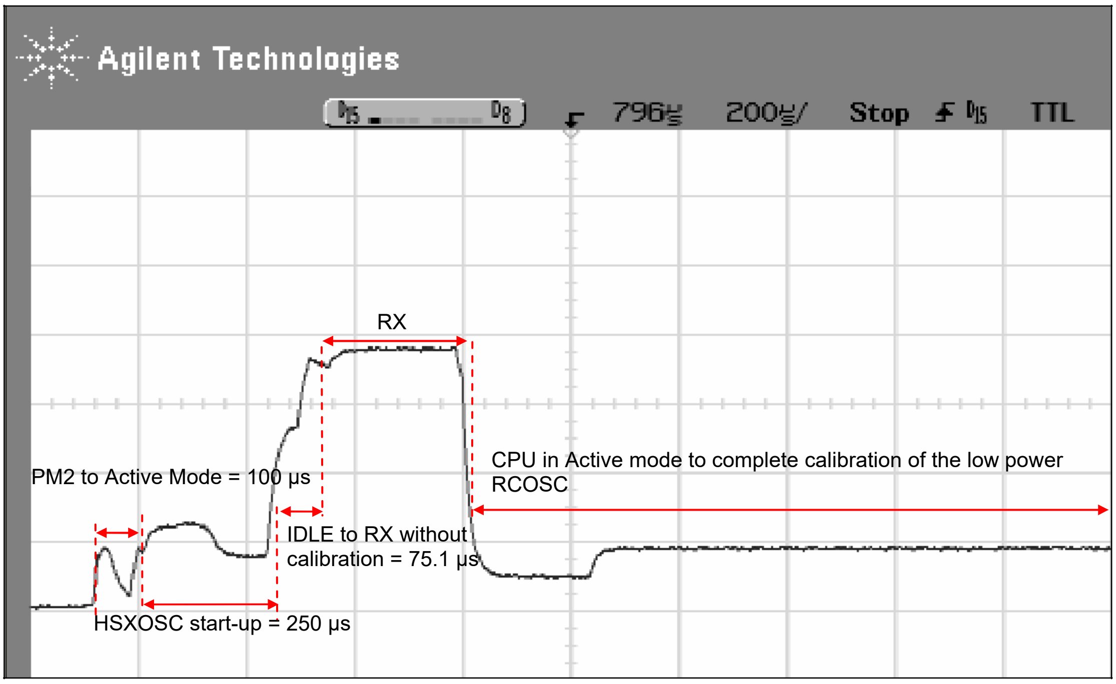


Figure 32. Current Consumption Test Case 9 (start-up sequence)

This gives an average current consumption of

$$Current_{AVG} = \frac{2100 \cdot 5660 + (1000000 - 2100) \cdot 0.5}{1000000} = 12.4\ uA$$

6.3.4 Comparing Test Case 7, 8, and 9

Figure 33 and Figure 34 shows the three test cases 7 - 9 in the same plot. Test case 8 has lower current consumption than test case 7 due to lack of calibration of the PLL, while test case 9 has much lower current consumption than the other two due to significantly reduced time spent in RX.

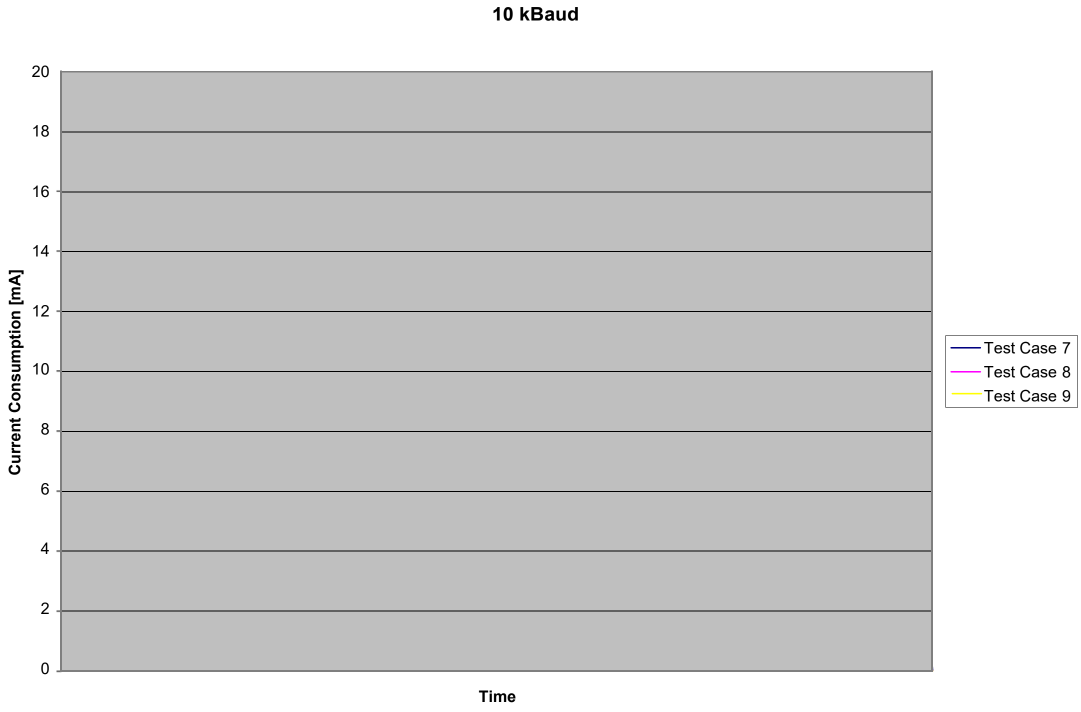


Figure 33. Test Case 7 - 9

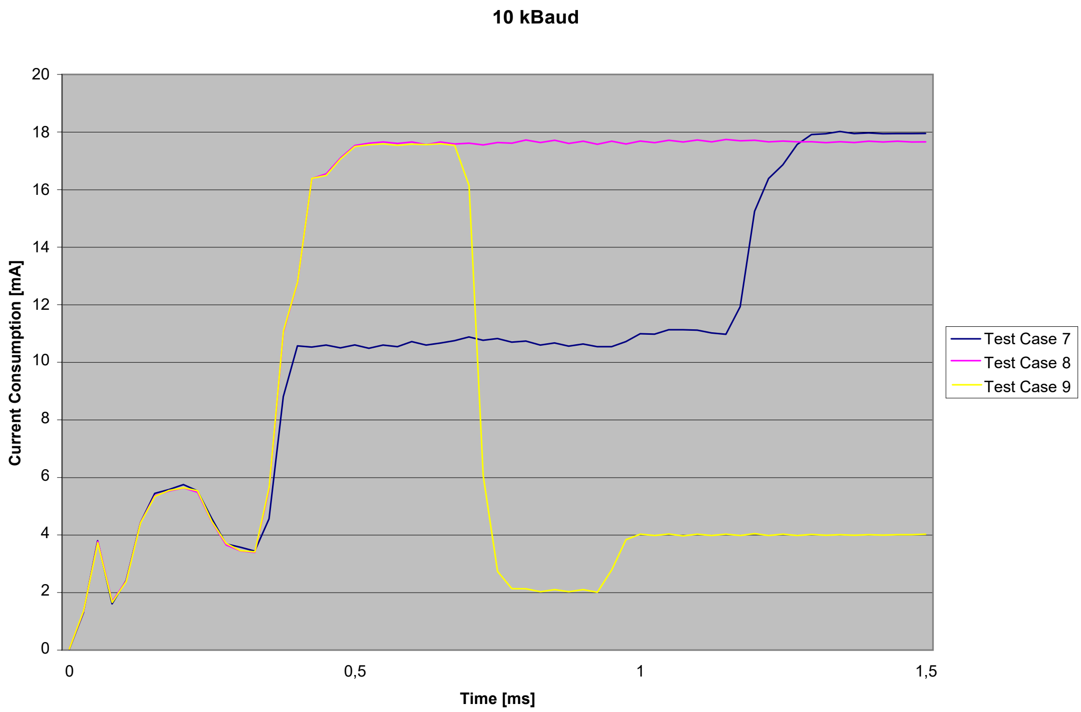


Figure 34. Test Case 7 - 9 (start-up sequence)

6.3.5 Test Case 10

Figure 35 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2.

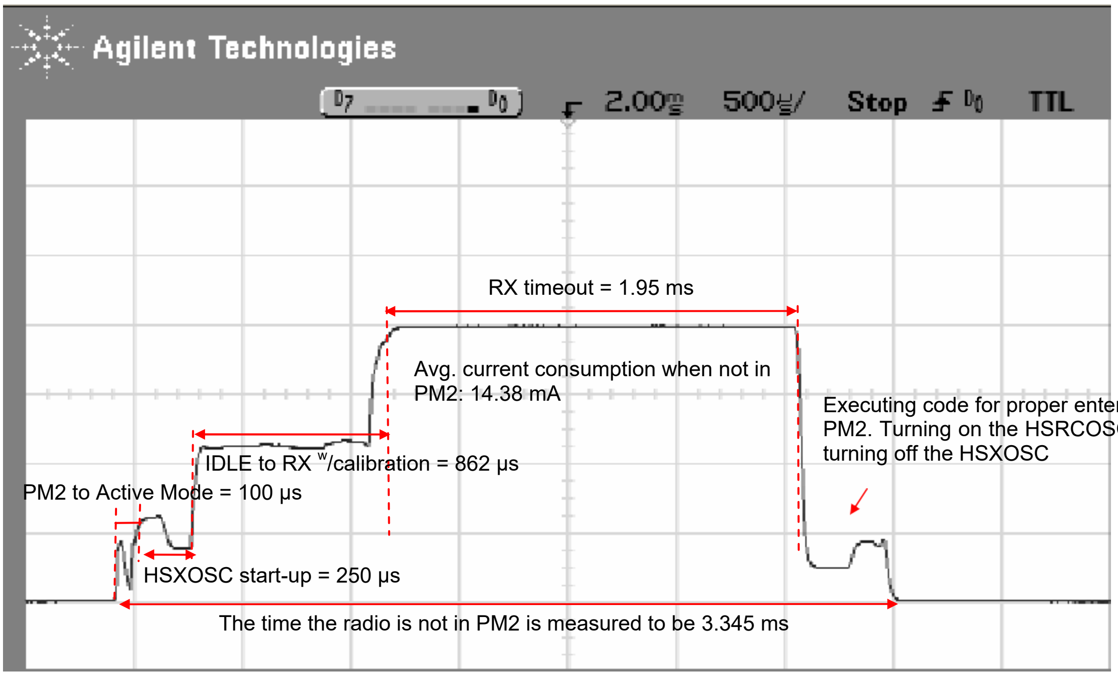


Figure 35. Current Consumption Test Case 10

This gives an average current consumption of

$$Current_{AVG} = \frac{3345 \cdot 14380 + (1000000 - 3345) \cdot 0.5}{1000000} = 48.6\ uA$$

6.3.6 Test Case 11

Figure 36 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2.

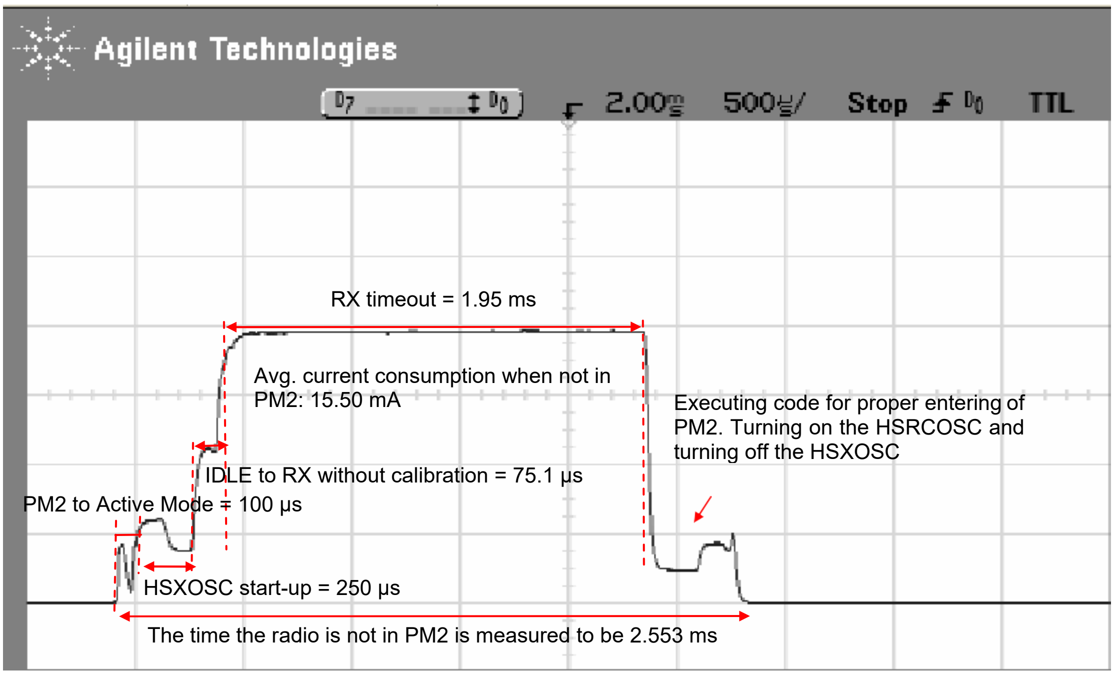


Figure 36. Current Consumption Test Case 11

This gives an average current consumption of

$$Current_{AVG} = \frac{2553 \cdot 15500 + (1000000 - 2553) \cdot 0.5}{1000000} = 40.1uA$$

6.3.7 Test Case 12

Figure 37 shows the current consumption for the complete period of time when the CC2510Fx is not in PM2.

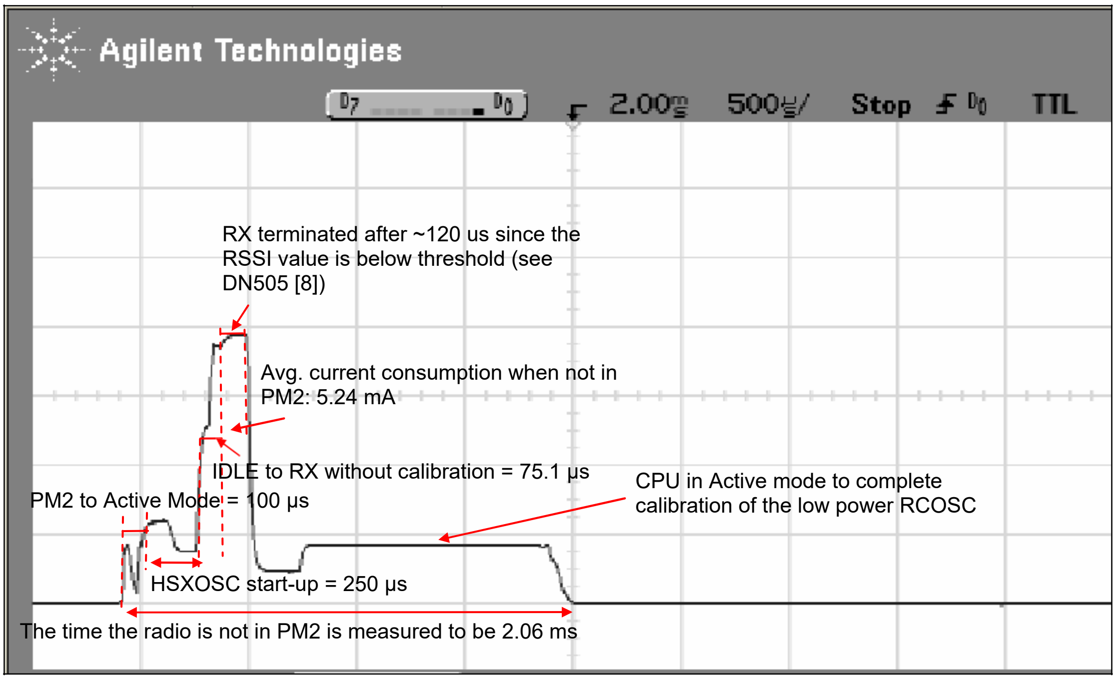


Figure 37. Current Consumption Test Case 12

This gives an average current consumption of

$$Current_{AVG} = \frac{2060 \cdot 5240 + (1000000 - 2060) \cdot 0.5}{1000000} = 11.3\ uA$$

6.3.8 Comparing Test Case 10, 11, and 12

Figure 38 shows the three test cases 10 - 12 in the same plot. The result is the same as discussed in Section 6.3.4 (test case 10 corresponds to test case 7, 11 to 8, and 12 to 9 respectively).

Figure 38. Test Case 10 - 12

7 CC2500 vs. CC2510

Table 8 shows a comparison between two different solutions for implementing a polling receiver. One solution is using a transceiver in WOR mode (in this case the CC2500) and the other solution is to use the Sleep Timer of a SoC (in this case CC2510Fx). As we can see, the current consumption seems a little bit higher on the SoC solution, but have in mind that for the transceiver solution, the current consumption of the MCU will be in addition to the numbers shown in Table 8. Please also note that the numbers represented here are not typical numbers as only a few measurements have been done.

	CC2500		CC2510Fx	
Data Rate [kBaud]	**Test Case**	**Avg. Current Consumption [µA]**	**Test Case**	**Avg. Current Consumption [µA]**
10	1	552.2	7	577.5
	2	544.5	8	562.5
	3	8.5	9	12.4
250	4	43.3	10	48.6
	5	37.7	11	40.1
	6	6.3	12	11.3

Table 8. CC2500 vs. CC2510Fx

Figure 39 and Figure 40 shows the different CC2500 test cases (1 - 6) compared to the CC2510Fx test cases (7 - 12).

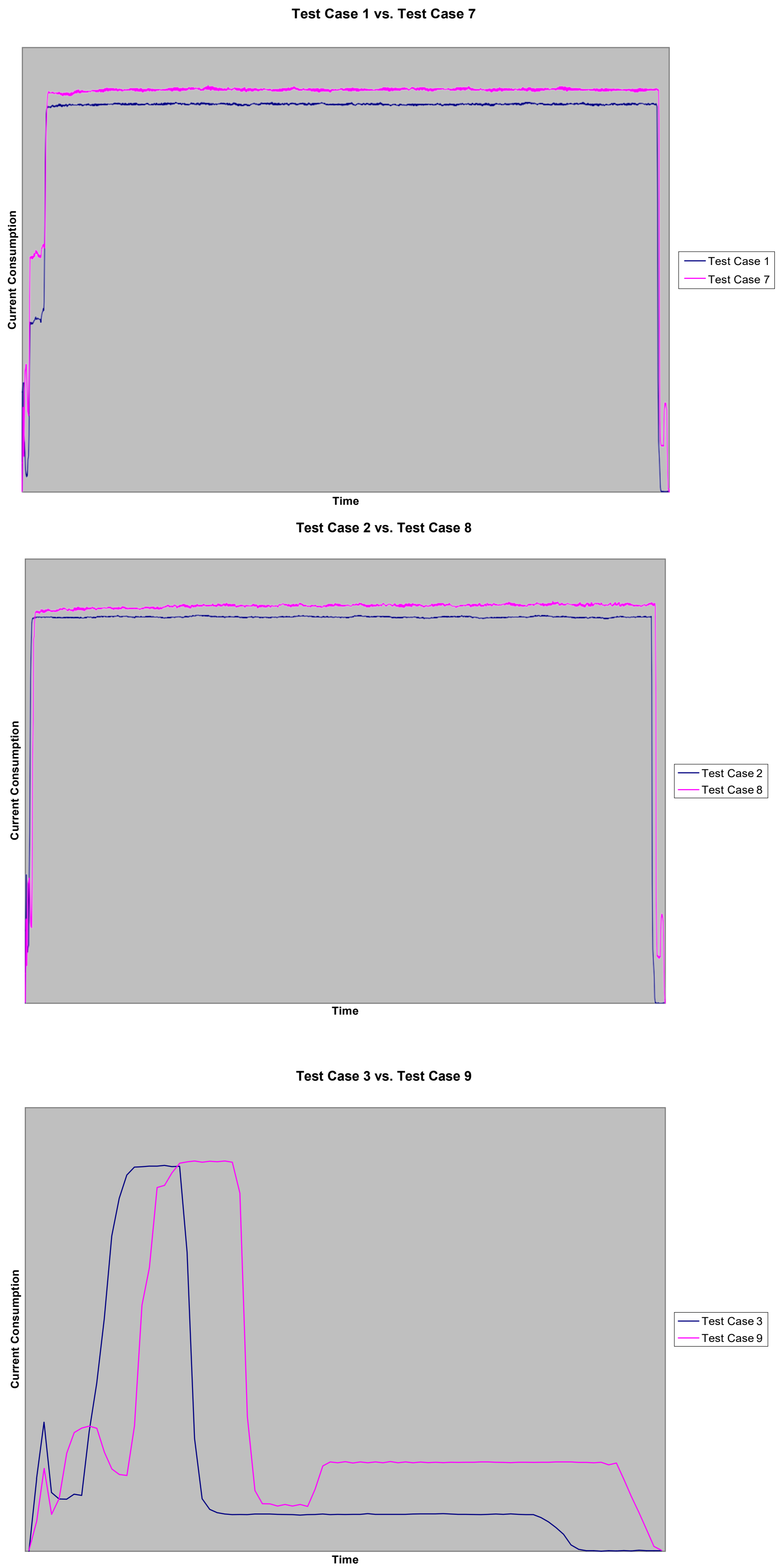
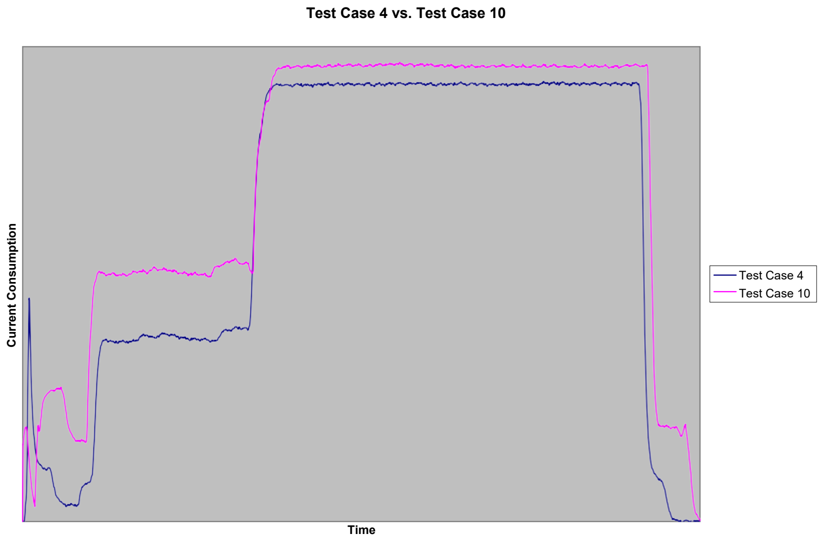
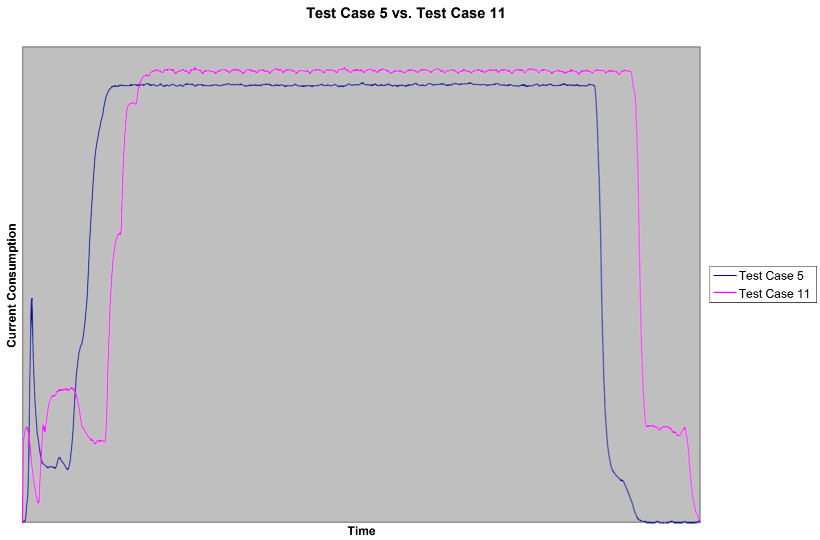


Figure 39. CC2500 vs. CC2510Fx (10 kBaud)





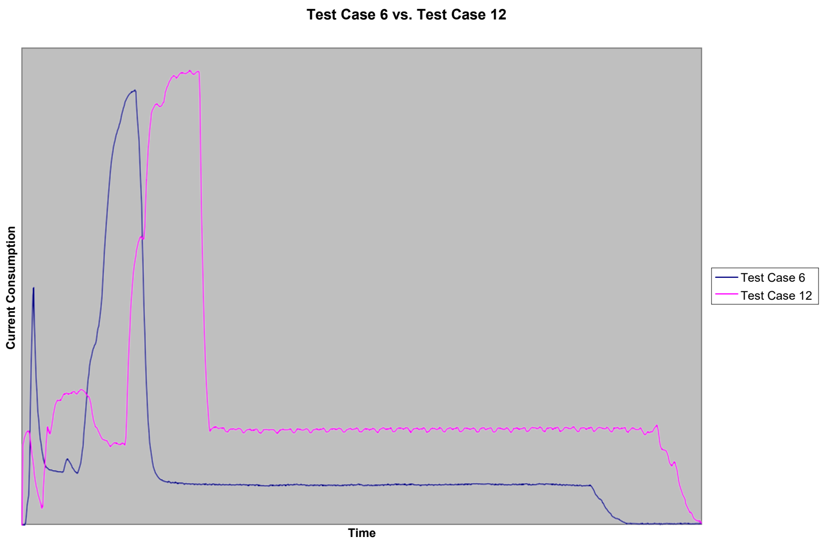


Figure 40. CC2500 vs. CC2510Fx (250 kBaud)

8 Conclusion

There are several things that play a role when optimizing for current consumption, and even if this design note can not show what is the best solution for all possible applications, it has hopefully provided some useful hints on what to take into consideration when optimizing a polling receiver with respect to current consumption.

Below is a short summary of the different things one should have in mind:

- A transceiver solution might or might not be better than the SoC solution, depending on the current consumption of the MCU
- Calibration of the PLL does not have to be done every time the radio should enter RX mode
- RX termination based on RSSI will reduce the current consumption significantly, as will a higher data rate (less time in RX)
- The system clock speed on the SoCs does not only affect the current consumption but also the state transition times
- Halting the CPU (PM0) will reduce the current consumption compared to when the SoC is in Active mode

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

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

[6] Application Note AN047 CC1100/CC2500 - Wake-On-Radio (swra126.pdf)

[7] SmartRF™ Studio (swrc046.zip)

[8] DN505 RSSI Interpretation and Timing (swra114.pdf)

[9] DN106 Power Modes in CC111xFx, CC243x, and CC251xFx (swra162.pdf)

[10] DN110 State Transition Times on CC111xFx and CC251xFx(swra191.pdf)

[11] CC1100E Single-Chip Low Cost Low Power RF-Transceiver, Data sheet (cc1100E.pdf)

General Information

9.1 Document History

Revision	Date	Description/Changes
SWRA207A	2009.03.12	Added CC1100E
SWRA207	2008.07.11	Initial release.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2019, Texas Instruments Incorporated