

Technical Description  
 Remote Controller  
 Model: KC-2616

Component	Part Number	Description
U2	SN8P1707	Microprocessor
XC9110	XC9110C/D	Voltage Converter
U1	HT7133	Voltage Regulator
U102	HJM1611	Voltage Regulator
U3	CC2500	2.4GHz Modulator
Oscillator	Y1	12MHz Crystal
CON1	MAR105P	RF Connector
Oscillator	OSC1 Y1	26MHz Crystal
Filter	L131,L121,C132,C131, C121,C122	2.4GHz LC Filter/ Matching network

1. Crystal (Y1): 12MHz crystal supply the CPU base frequency.
2. Crystal (OSC1 Y1): 26MHz crystal supply the RF modulator base frequency.
3. RF modulator (CC2500) : Receive and transmit the control data.
4. CPU (MCS2750) : deal with the button single and receive data, communicate with RF modulator, manager the whole system.
5. The CPU and RF modulator is not ROM, so SPI face add ROM place apply soft. When power on, The CPU remove form ROM apply the soft to CPU RAM.
6. Key and stick: input the control data to CPU.
7. LDO (XC9110C/D) : make the voltage is 3.3v, give the CPU supply the stable voltage.
8. DC/DC SET UP : set up two batteries voltage to 3.3v, give the CPU supply the stable voltage.
9. Charging circuit: charging with rechargeable battery throughout USB port 5V.
10. PCB antenna: the 50 OHM PCB antenna make the RF single transmit and receive.
11. CE: control the DC/DC set up circuit. Set up circuit is not working when charging circuit is on.
12. LC filter: consisted of the L131, L121, C132, C131, C121, C122.

## **CC2500**

# Single Chip Low Cost Low Power RF Transceiver

### Applications

- 2400-2483.5 MHz ISM/SRD band systems
- Consumer Electronics
- Wireless game controllers
- Wireless audio
- Wireless keyboard and mouse

### Product Description

The **CC2500** is a low cost true single chip 2.4 GHz transceiver designed for very low power wireless applications. The circuit is intended for the ISM (Industrial, Scientific and Medical) and SRD (Short Range Device) frequency band at 2400-2483.5 MHz.

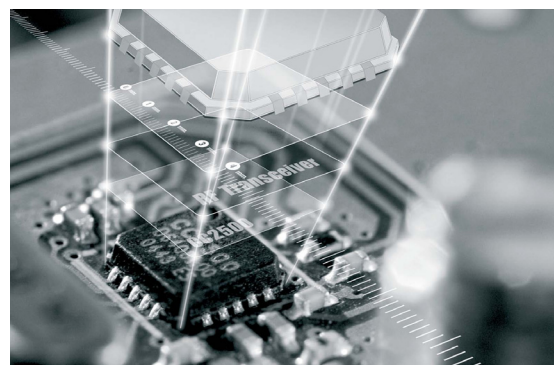
The RF transceiver is integrated with a highly configurable baseband modem. The modem supports various modulation formats and has a configurable data rate up to 500 kbps. The communication range can be increased by enabling a Forward Error Correction option, which is integrated in the modem.

**CC2500** provides extensive hardware support for packet handling, data buffering, burst transmissions, clear channel assessment, link quality indication and wake-on-radio.

The main operating parameters and the 64-byte transmit/receive FIFOs of **CC2500** can be

controlled via an SPI interface. In a typical system, the **CC2500** will be used together with a microcontroller and a few additional passive components.

**CC2500** is based on Chipcon's SmartRF<sup>®</sup>04 technology in 0.18  $\mu$ m CMOS.



### Key Features

- Small size (QLP 4x4 mm package, 20 pins)
- True single chip 2.4 GHz RF transceiver
- Frequency range: 2400-2483.5 MHz
- High sensitivity (-101 dBm at 10 kbps, 1% packet error rate)
- Programmable data rate up to 500 kbps
- Low current consumption (13.3 mA in RX, 250 kbps, input 30 dB above sensitivity limit)
- Programmable output power up to +1 dBm
- Excellent receiver selectivity and blocking performance
- Very few external components: Completely on-chip frequency synthesizer, no external filters or RF switch needed
- Programmable baseband modem
- Ideal for multi-channel operation
- Configurable packet handling hardware
- Suitable for frequency hopping systems due to a fast settling frequency synthesizer
- Optional Forward Error Correction with interleaving
- Separate 64-byte RX and TX data FIFOs
- Efficient SPI interface: All registers can be programmed with one "burst" transfer
- Digital RSSI output
- Suited for systems compliant with EN 300 328 and EN 300 440 class 2 (Europe), CFR47 Part 15 (US), and ARIB STD-T66 (Japan)
- Wake-on-radio functionality for automatic low-power RX polling
- Many powerful digital features allow a high-performance RF system to be made using an inexpensive microcontroller
- Integrated analog temperature sensor
- Lead-free "green" package

## Features (continued from front page)

- Flexible support for packet oriented systems: On chip support for sync word detection, address check, flexible packet length and automatic CRC handling.
- Programmable channel filter bandwidth
- 2-FSK, GFSK and MSK supported
- OOK supported
- Automatic Frequency Compensation can be used to align the frequency synthesizer to received centre frequency
- Optional automatic whitening and de-whitening of data
- Support for asynchronous transparent receive/transmit mode for backwards compatibility with existing radio communication protocols
- Programmable Carrier Sense indicator
- Programmable Preamble Quality Indicator for detecting preambles and improved protection against sync word detection in random noise
- Support for automatic Clear Channel Assessment (CCA) before transmitting (for listen-before-talk systems)
- Support for per-package Link Quality Indication

## Abbreviations

Abbreviations used in this data sheet are described below.

2-FSK	Binary Frequency Shift Keying	MSK	Minimum Shift Keying
ADC	Analog to Digital Converter	NA	Not Applicable
AFC	Automatic Frequency Offset Compensation	PA	Power Amplifier
AGC	Automatic Gain Control	PCB	Printed Circuit Board
AMR	Automatic Meter Reading	PD	Power Down
ASK	Amplitude Shift Keying	PER	Packet Error Rate
BER	Bit Error Rate	PLL	Phase Locked Loop
CCA	Clear Channel Assessment	POR	Power-on Reset
CRC	Cyclic Redundancy Check	PQI	Preamble Quality Indicator
CS	Carrier Sense	PQT	Preamble Quality Threshold
DC	Direct Current	RCOSC	RC Oscillator
EIRP	Equivalent Isotropic Radiated Power	RF	Radio Frequency
ESR	Equivalent Series Resistance	RSSI	Received Signal Strength Indicator
FEC	Forward Error Correction	RX	Receive, Receive Mode
FIFO	First-In-First-Out	SAW	Surface Acoustic Wave
FHSS	Frequency Hopping Spread Spectrum	SNR	Signal to Noise Ratio
FSK	Frequency Shift Keying	SPI	Serial Peripheral Interface
GFSK	Gaussian shaped Frequency Shift Keying	TBD	To Be Defined
IF	Intermediate Frequency	TX	Transmit, Transmit Mode
LBT	Listen Before Transmit	VCO	Voltage Controlled Oscillator
LNA	Low Noise Amplifier	WOR	Wake on Radio, Low power polling
LO	Local Oscillator	XOSC	Crystal Oscillator
LQI	Link Quality Indicator	XTAL	Crystal
MCU	Microcontroller Unit		

## Table Of Contents

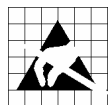
<b>APPLICATIONS</b> .....	<b>1</b>
<b>PRODUCT DESCRIPTION</b> .....	<b>1</b>
<b>KEY FEATURES</b> .....	<b>1</b>
<b>FEATURES (CONTINUED FROM FRONT PAGE)</b> .....	<b>2</b>
<b>ABBREVIATIONS</b> .....	<b>2</b>
<b>TABLE OF CONTENTS</b> .....	<b>3</b>
<b>1 ABSOLUTE MAXIMUM RATINGS</b> .....	<b>6</b>
<b>2 OPERATING CONDITIONS</b> .....	<b>6</b>
<b>3 GENERAL CHARACTERISTICS</b> .....	<b>6</b>
<b>4 ELECTRICAL SPECIFICATIONS</b> .....	<b>7</b>
4.1 CURRENT CONSUMPTION .....	7
4.2 RF RECEIVE SECTION .....	8
4.3 RF TRANSMIT SECTION .....	10
4.4 CRYSTAL OSCILLATOR .....	10
4.5 LOW POWER RC OSCILLATOR .....	11
4.6 FREQUENCY SYNTHESIZER CHARACTERISTICS .....	11
4.7 ANALOG TEMPERATURE SENSOR .....	12
4.8 DC CHARACTERISTICS .....	12
4.9 POWER-ON RESET .....	12
<b>5 PIN CONFIGURATION</b> .....	<b>13</b>
<b>6 CIRCUIT DESCRIPTION</b> .....	<b>15</b>
<b>7 APPLICATION CIRCUIT</b> .....	<b>15</b>
<b>8 CONFIGURATION OVERVIEW</b> .....	<b>17</b>
<b>9 CONFIGURATION SOFTWARE</b> .....	<b>18</b>
<b>10 4-WIRE SERIAL CONFIGURATION AND DATA INTERFACE</b> .....	<b>19</b>
10.1 CHIP STATUS BYTE .....	20
10.2 REGISTER ACCESS .....	21
10.3 COMMAND STROBES .....	22
10.4 FIFO ACCESS .....	22
10.5 PATABLE ACCESS.....	22
<b>11 MICROCONTROLLER INTERFACE AND PIN CONFIGURATION</b> .....	<b>23</b>
11.1 CONFIGURATION INTERFACE .....	23
11.2 GENERAL CONTROL AND STATUS PINS .....	23
11.3 OPTIONAL RADIO CONTROL FEATURE .....	23
<b>12 DATA RATE PROGRAMMING</b> .....	<b>24</b>
<b>13 RECEIVER CHANNEL FILTER BANDWIDTH</b> .....	<b>24</b>
<b>14 DEMODULATOR, SYMBOL SYNCHRONIZER AND DATA DECISION</b> .....	<b>25</b>
14.1 FREQUENCY OFFSET COMPENSATION.....	25
14.2 BIT SYNCHRONIZATION.....	25
14.3 BYTE SYNCHRONIZATION.....	25
<b>15 PACKET HANDLING HARDWARE SUPPORT</b> .....	<b>25</b>
15.1 DATA WHITENING .....	26
15.2 PACKET FORMAT .....	26
15.3 PACKET FILTERING IN RECEIVE MODE .....	27
15.4 CRC CHECK .....	28
15.5 PACKET HANDLING IN TRANSMIT MODE.....	28
15.6 PACKET HANDLING IN RECEIVE MODE .....	28
<b>16 MODULATION FORMATS</b> .....	<b>29</b>
16.1 FREQUENCY SHIFT KEYING .....	29
16.2 MINIMUM SHIFT KEYING.....	29
16.3 AMPLITUDE MODULATION .....	29

<b>17</b>	<b>RECEIVED SIGNAL QUALIFIERS AND LINK QUALITY INFORMATION .....</b>	<b>30</b>
17.1	SYNC WORD QUALIFIER.....	30
17.2	PREAMBLE QUALITY THRESHOLD (PQT) .....	30
17.3	RSSI.....	30
17.4	CARRIER SENSE (CS).....	31
17.5	CLEAR CHANNEL ASSESSMENT (CCA) .....	32
17.6	LINK QUALITY INDICATOR (LQI) .....	32
<b>18</b>	<b>FORWARD ERROR CORRECTION WITH INTERLEAVING .....</b>	<b>33</b>
18.1	FORWARD ERROR CORRECTION (FEC).....	33
18.2	INTERLEAVING .....	33
<b>19</b>	<b>RADIO CONTROL.....</b>	<b>34</b>
19.1	POWER-ON START-UP SEQUENCE .....	35
19.2	CRYSTAL CONTROL.....	35
19.3	VOLTAGE REGULATOR CONTROL.....	35
19.4	ACTIVE MODES .....	35
19.5	WAKE ON RADIO (WOR).....	36
19.6	TIMING .....	37
19.7	RX TERMINATION TIMER .....	37
<b>20</b>	<b>DATA FIFO .....</b>	<b>38</b>
<b>21</b>	<b>FREQUENCY PROGRAMMING.....</b>	<b>39</b>
<b>22</b>	<b>VCO .....</b>	<b>40</b>
22.1	VCO AND PLL SELF-CALIBRATION .....	40
<b>23</b>	<b>VOLTAGE REGULATORS .....</b>	<b>40</b>
<b>24</b>	<b>OUTPUT POWER PROGRAMMING .....</b>	<b>40</b>
<b>25</b>	<b>SELECTIVITY GRAPHS .....</b>	<b>42</b>
<b>26</b>	<b>CRYSTAL OSCILLATOR.....</b>	<b>44</b>
26.1	REFERENCE SIGNAL .....	44
<b>27</b>	<b>EXTERNAL RF MATCH .....</b>	<b>45</b>
<b>28</b>	<b>GENERAL PURPOSE / TEST OUTPUT CONTROL PINS .....</b>	<b>45</b>
<b>29</b>	<b>ASYNCHRONOUS AND SYNCHRONOUS SERIAL OPERATION .....</b>	<b>47</b>
29.1	ASYNCHRONOUS OPERATION.....	47
29.2	SYNCHRONOUS SERIAL OPERATION .....	47
<b>30</b>	<b>SYSTEM CONSIDERATIONS AND GUIDELINES .....</b>	<b>47</b>
30.1	SRD REGULATIONS.....	47
30.2	FREQUENCY HOPPING AND MULTI-CHANNEL SYSTEMS.....	47
30.3	DATA BURST TRANSMISSIONS.....	48
30.4	CONTINUOUS TRANSMISSIONS .....	48
30.5	CRYSTAL DRIFT COMPENSATION .....	48
30.6	SPECTRUM EFFICIENT MODULATION.....	48
30.7	LOW COST SYSTEMS .....	49
30.8	BATTERY OPERATED SYSTEMS .....	49
30.9	INCREASING OUTPUT POWER .....	49
<b>31</b>	<b>CONFIGURATION REGISTERS.....</b>	<b>49</b>
31.1	CONFIGURATION REGISTER DETAILS – REGISTERS WITH PRESERVED VALUES IN SLEEP STATE .....	54
31.2	CONFIGURATION REGISTER DETAILS – REGISTERS THAT LOSE PROGRAMMING IN SLEEP STATE .....	69
31.3	STATUS REGISTER DETAILS.....	70
<b>32</b>	<b>PACKAGE DESCRIPTION (QLP 20).....</b>	<b>73</b>
32.1	RECOMMENDED PCB LAYOUT FOR PACKAGE (QLP 20).....	74
32.2	PACKAGE THERMAL PROPERTIES .....	74
32.3	SOLDERING INFORMATION.....	74
32.4	TRAY SPECIFICATION.....	74
32.5	CARRIER TAPE AND REEL SPECIFICATION .....	75
<b>33</b>	<b>ORDERING INFORMATION.....</b>	<b>75</b>
<b>34</b>	<b>GENERAL INFORMATION.....</b>	<b>75</b>

34.1	DOCUMENT HISTORY .....	75
34.2	PRODUCT STATUS DEFINITIONS .....	75
34.3	DISCLAIMER .....	76
34.4	TRADEMARKS.....	76
34.5	LIFE SUPPORT POLICY .....	76
<b>35</b>	<b>ADDRESS INFORMATION.....</b>	<b>77</b>

## 1 Absolute Maximum Ratings

Under no circumstances must the absolute maximum ratings given in Table 1 be violated. Stress exceeding one or more of the limiting values may cause permanent damage to the device.



**Caution!** ESD sensitive device.  
Precaution should be used when handling the device in order to prevent permanent damage.

Parameter	Min	Max	Units	Condition
Supply voltage	-0.3	3.6	V	All supply pins must have the same voltage
Voltage on any digital pin	-0.3	VDD+0.3 max 3.6	V	
Voltage on the pins RF_P, RF_N and DCOUPL	-0.3	2.0	V	
Voltage ramp-up rate		120	kV/ $\mu$ s	
Input RF level		+10	dBm	
Storage temperature range	-50	150	$^{\circ}$ C	
Solder reflow temperature		260	$^{\circ}$ C	T = 10 s

**Table 1: Absolute Maximum Ratings**

## 2 Operating Conditions

The operating conditions for **CC2500** are listed Table 2 in below.

Parameter	Min	Max	Unit	Condition
Operating temperature	-40	85	$^{\circ}$ C	
Operating supply voltage	1.8	3.6	V	All supply pins must have the same voltage

**Table 2: Operating Conditions**

## 3 General Characteristics

Parameter	Min	Typ	Max	Unit	Condition/Note
Frequency range	2400		2483.5	MHz	
Data rate	1.2		500	kbps	Modulation formats supported: (Shaped) MSK (also known as differential offset QPSK) up to 500 kbps 2-FSK up to 500 kbps GFSK and OOK (up to 250 kbps) Optional Manchester encoding (halves the data rate).

**Table 3: General Characteristics**

## 4 Electrical Specifications

### 4.1 Current Consumption

Tc = 25°C, VDD = 3.0 V if nothing else stated. All measurements were performed using the CC2500EM reference design.

Parameter	Min	Typ	Max	Unit	Condition
Current consumption in power down modes		400		nA	Voltage regulator to digital part off, register values retained (SLEEP state)
		900		nA	Voltage regulator to digital part off, register values retained, low-power RC oscillator running (SLEEP state with WOR enabled)
		92		μA	Voltage regulator to digital part off, register values retained, XOSC running (SLEEP state with <code>MCSM0.OSC_FORCE_ON</code> set)
		157		μA	Voltage regulator to digital part on, all other modules in power down (XOFF state)
Current consumption		1.4		μA	Automatic RX polling once each second, using low-power RC oscillator, with 460 kHz filter bandwidth and 250 kbps data rate, PLL calibration every 4 <sup>th</sup> wakeup. Average current with signal in channel <i>below</i> carrier sense level.
		17		μA	Same as above, but with signal in channel <i>above</i> carrier sense level, 1.9 ms RX timeout, and no preamble/sync word found.
		0.9		μA	Automatic RX polling every 15 <sup>th</sup> second, using low-power RC oscillator, with 460 kHz filter bandwidth and 250 kbps data rate, PLL calibration every 4 <sup>th</sup> wakeup. Average current with signal in channel below carrier sense level.
		37		μA	Same as above, but with signal in channel <i>above</i> carrier sense level, 14 ms RX timeout, and no preamble/sync word found.
		1.5		mA	Only voltage regulator to digital part and crystal oscillator running (IDLE state)
		7.4		mA	Only the frequency synthesizer running (after going from IDLE until reaching RX or TX states, and frequency calibration states)
Current consumption, RX states		15.3		mA	Receive mode, 2.4 kbps, input at sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		12.8		mA	Receive mode, 2.4 kbps, input 30 dB above sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		15.4		mA	Receive mode, 10 kbps, input at sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		12.9		mA	Receive mode, 10 kbps, input 30 dB above sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		18.8		mA	Receive mode, 250 kbps, input at sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 0</code>
		15.7		mA	Receive mode, 250 kbps, input 30 dB above sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 0</code>
		16.6		mA	Receive mode, 250 kbps reduced current, input at sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		13.3		mA	Receive mode, 250 kbps reduced current, input 30 dB above sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 1</code>
		19.6		mA	Receive mode, 500 kbps, input at sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 0</code>
		17.0		mA	Receive mode, 500 kbps, input 30 dB above sensitivity limit, <code>MDMCFG2.DEM_DCFILT_OFF = 0</code>
Current consumption, TX states		11.1		mA	Transmit mode, -12 dBm output power
		15.1		mA	Transmit mode, -6 dBm output power



Parameter	Min	Typ	Max	Unit	Condition
		21.2		mA	Transmit mode, 0 dBm output power
		21.5		mA	Transmit mode, 1.5 dBm output power

**Table 4: Current Consumption**

#### 4.2 RF Receive Section

Tc = 25°C, VDD = 3.0 V if nothing else stated. All measurements were performed using the CC2500EM reference design.

Parameter	Min	Typ	Max	Unit	Condition/Note
Digital channel filter bandwidth	58		812	kHz	User programmable. The bandwidth limits are proportional to crystal frequency (given values assume a 26.0 MHz crystal).
<b>2.4 kbps data rate, reduced current, MDMCFG2.DEM_DCFILT_OFF = 1</b> (2-FSK, 1% packet error rate, 20 bytes packet length, 203 kHz digital channel filter bandwidth)					
Receiver sensitivity		-104		dBm	The sensitivity can be improved to typically -106 dBm by setting MDMCFG2.DEM_DCFILT_OFF = 0. The typical current consumption is in this case 17.0 mA at sensitivity limit.
Saturation		-13		dBm	
Adjacent channel rejection		23		dB	Desired channel 3 dB above the sensitivity limit. 250 kHz channel spacing
Alternate channel rejection		31		dB	Desired channel 3 dB above the sensitivity limit. 250 kHz channel spacing
					See Figure 17 for plot of selectivity versus frequency offset
<b>10 kbps data rate, reduced current, MDMCFG2.DEM_DCFILT_OFF = 1</b> (2-FSK, 1% packet error rate, 20 bytes packet length, 232 kHz digital channel filter bandwidth)					
Receiver sensitivity		-99		dBm	The sensitivity can be improved to typically -101 dBm by setting MDMCFG2.DEM_DCFILT_OFF = 0. The typical current consumption is in this case 17.3 mA at sensitivity limit.
Saturation		-9		dBm	
Adjacent channel rejection		18		dB	Desired channel 3 dB above the sensitivity limit. 250 kHz channel spacing
Alternate channel rejection		25		dB	Desired channel 3 dB above the sensitivity limit. 250 kHz channel spacing
					See Figure 18 for plot of selectivity versus frequency offset
<b>250 kbps data rate, MDMCFG2.DEM_DCFILT_OFF = 0</b> (MSK, 1% packet error rate, 20 bytes packet length, 540 kHz digital channel filter bandwidth)					
Receiver sensitivity		-89		dBm	
Saturation		-13		dBm	
Adjacent channel rejection		21		dB	Desired channel 3 dB above the sensitivity limit. 750 kHz channel spacing
Alternate channel rejection		30		dB	Desired channel 3 dB above the sensitivity limit. 750 kHz channel spacing
					See Figure 19 for plot of selectivity versus frequency offset

Parameter	Min	Typ	Max	Unit	Condition/Note
<b>250 kbps data rate, reduced current</b> , MDMCFG2.DEM_DCFILT_OFF = 1 (MSK, 1% packet error rate, 20 bytes packet length, 540 kHz digital channel filter bandwidth)					
Receiver sensitivity		-87		dBm	
Saturation		-13		dBm	
Adjacent channel rejection		21		dB	Desired channel 3 dB above the sensitivity limit. 750 kHz channel spacing
Alternate channel rejection		30		dB	Desired channel 3 dB above the sensitivity limit. 750 kHz channel spacing
					See Figure 20 for plot of selectivity versus frequency offset
<b>500 kbps data rate</b> , MDMCFG2.DEM_DCFILT_OFF = 0 (MSK, 1% packet error rate, 20 bytes packet length, 812 kHz digital channel filter bandwidth)					
Receiver sensitivity		-82		dBm	
Saturation		-18		dBm	
Adjacent channel rejection		14		dB	Desired channel 3 dB above the sensitivity limit. 1 MHz channel spacing
Alternate channel rejection		25		dB	Desired channel 3 dB above the sensitivity limit. 1 MHz channel spacing
					See Figure 21 for plot of selectivity versus frequency offset
<b>General</b>					
Selectivity at 10 MHz offset		47		dB	Desired channel at -80 dBm. Compliant with ETSI EN 300 440 class 2 receiver requirements.
Selectivity at 20 MHz offset		52		dB	Desired channel at -80 dBm. Compliant with ETSI EN 300 440 class 2 receiver requirements.
Selectivity at 50 MHz offset		54		dB	Desired channel at -80 dBm. Compliant with ETSI EN 300 440 class 2 receiver requirements.
Spurious emissions 25 MHz – 1 GHz			-57	dBm	
Above 1 GHz			-47	dBm	

**Table 5: RF Receive Parameters**

### 4.3 RF Transmit Section

Tc = 25°C, VDD = 3.0 V, 0 dBm if nothing else stated. All measurements were performed using the CC2500EM reference design.

Parameter	Min	Typ	Max	Unit	Condition/Note
Differential load impedance		80 + j74		Ω	Differential impedance as seen from the RF-port (RF_P and RF_N) towards the antenna. Follow the CC2500EM reference design available from Chipcon's website.
Output power, highest setting		+1		dBm	Output power is programmable and is available across the entire frequency band Delivered to a 50 Ω single-ended load via Chipcon reference design RF matching network.
Output power, lowest setting		-30		dBm	Output power is programmable and is available across the entire frequency band Delivered to a 50 Ω single-ended load via Chipcon reference design RF matching network.
Spurious emissions 25 MHz – 1 GHz 47-74, 87.5-118, 174-230, 470-862 MHz 1800-1900 MHz At 2-RF and 3-RF Otherwise above 1 GHz			-36 -54 -47 -41 -30	dBm dBm dBm dBm dBm	Restricted band in Europe Restricted bands in USA

**Table 6: RF Transmit Parameters**

### 4.4 Crystal Oscillator

Tc = 25°C, VDD = 3.0 V if nothing else stated.

Parameter	Min	Typ	Max	Unit	Condition/Note
Crystal frequency	26	26	27	MHz	
Tolerance		±40		ppm	This is the total tolerance including a) initial tolerance, b) crystal loading, c) aging and d) temperature dependence. The acceptable crystal tolerance depends on RF frequency and channel spacing / bandwidth.
ESR			100	Ω	
Start-up time		300		μs	Measured on Chipcon's CC2500EM reference design.

**Table 7: Crystal Oscillator Parameters**

#### 4.5 Low Power RC Oscillator

Typical performance is for Tc = 25°C @ VDD = 3.0 V if nothing else is stated. The values in the table are simulated results and will be updated in later versions of the data sheet.

Parameter	Min	Typ	Max	Unit	Condition/Note
Calibrated frequency	34.6	34.7	36	kHz	Calibrated RC Oscillator frequency is XTAL frequency divided by 750
Frequency accuracy after calibration			+0.3 -10	%	
Temperature coefficient		+0.4		% / °C	Frequency drift when temperature changes after calibration
Supply voltage coefficient		+3		% / V	Frequency drift when supply voltage changes after calibration
Initial calibration time		2		ms	When the RC Oscillator is enabled, calibration is continuously done in the background as long as the crystal oscillator is running.
Wake-up period	58e-6		59650	Seconds	Programmable, dependent on XTAL frequency

**Table 8: RC Oscillator Parameters**

#### 4.6 Frequency Synthesizer Characteristics

Tc = 25°C, VDD = 3.0 V if nothing else stated. All measurements were performed using the CC2500EM reference design.

Parameter	Min	Typ	Max	Unit	Condition/Note
Programmed frequency resolution	397	$F_{XOSC}/2^{16}$	412	Hz	26-27 MHz crystal.
Synthesizer frequency tolerance		±40		ppm	Given by crystal used. Required accuracy (including temperature and aging) depends on frequency band and channel bandwidth / spacing.
RF carrier phase noise		-78		dBc/Hz	@ 50 kHz offset from carrier
RF carrier phase noise		-78		dBc/Hz	@ 100 kHz offset from carrier
RF carrier phase noise		-81		dBc/Hz	@ 200 kHz offset from carrier
RF carrier phase noise		-90		dBc/Hz	@ 500 kHz offset from carrier
RF carrier phase noise		-100		dBc/Hz	@ 1 MHz offset from carrier
RF carrier phase noise		-108		dBc/Hz	@ 2 MHz offset from carrier
RF carrier phase noise		-116		dBc/Hz	@ 5 MHz offset from carrier
RF carrier phase noise		-127		dBc/Hz	@ 10 MHz offset from carrier
PLL turn-on / hop time		90		µs	Time from leaving the IDLE state until arriving in the RX, FSTXON or TX state, when not performing calibration. Crystal oscillator running.
PLL RX/TX and TX/RX settling time		10		µs	Settling time for the 1xIF frequency step from RX to TX, and vice versa.
PLL calibration time	0.69	18739 0.72	0.72	XOSC cycles ms	Calibration can be initiated manually, or automatically before entering or after leaving RX/TX. Min/typ/max time is for 27/26/26 MHz crystal frequency.

**Table 9: Frequency Synthesizer Parameters**

#### 4.7 Analog Temperature Sensor

The characteristics of the analog temperature sensor are listed in Table 10 below. Note that it is necessary to write 0xBF to the PTEST register to use the analog temperature sensor in the IDLE state.

Parameter	Min	Typ	Max	Unit	Condition/Note
Output voltage at -40°C		0.660		V	
Output voltage at 0°C		0.755		V	
Output voltage at +40°C		0.859		V	
Output voltage at +80°C		0.958		V	
Output voltage at +120°C		1.056		V	
Temperature coefficient		2.54		mV/°C	Fitted from -20°C to +80°C
Error in calculated temperature, calibrated		0		°C	From -20°C to +80°C when using 2.54 mV / °C, after 1-point calibration at room temperature
Current consumption increase when enabled		0.3		mA	

**Table 10: Analog Temperature Sensor Parameters**

#### 4.8 DC Characteristics

The DC Characteristics of **CC2500** are listed in Table 11 below.

Tc = 25°C if nothing else stated.

Digital Inputs/Outputs	Min	Max	Unit	Condition
Logic "0" input voltage	0	0.7	V	
Logic "1" input voltage	VDD-0.7	VDD	V	
Logic "0" output voltage	0	0.5	V	For up to 4 mA output current
Logic "1" output voltage	VDD-0.3	VDD	V	For up to 4 mA output current
Logic "0" input current	NA	-1	μA	Input equals 0V
Logic "1" input current	NA	1	μA	Input equals VDD

**Table 11: DC Characteristics**

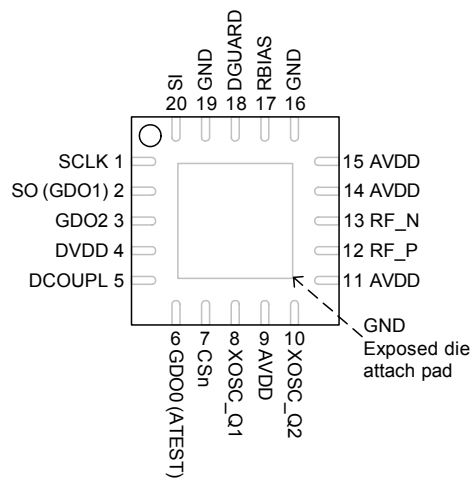
#### 4.9 Power-On Reset

When the power supply complies with the requirements in Table 12 below, proper Power-On-Reset functionality is guaranteed. Otherwise, the chip should be assumed to have unknown state until transmitting an SRES strobe over the SPI interface. See Section 19.1 on page 35 for further details.

Parameter	Min	Typ	Max	Unit	Condition/Note
Power ramp-up time			5	ms	From 0 V until reaching 1.8 V
Power off time	1			ms	Minimum time between power-on and power-off.

**Table 12: Power-On Reset Requirements**

## 5 Pin Configuration



**Figure 1: Pinout top view**

Note: The exposed die attach pad **must** be connected to a solid ground plane as this is the main ground connection for the chip.

Pin #	Pin name	Pin type	Description
1	SCLK	Digital Input	Serial configuration interface, clock input
2	SO (GDO1)	Digital Output	Serial configuration interface, data output. Optional general output pin when CS <sub>n</sub> is high
3	GDO2	Digital Output	Digital output pin for general use: <ul style="list-style-type: none"> <li>• Test signals</li> <li>• FIFO status signals</li> <li>• Clear Channel Indicator</li> <li>• Clock output, down-divided from XOSC</li> <li>• Serial output RX data</li> </ul>
4	DVDD	Power (Digital)	1.8 - 3.6 V digital power supply for digital I/O's and for the digital core voltage regulator
5	DCOUPPL	Power (Digital)	1.6 - 2.0 V digital power supply output for decoupling. NOTE: This pin is intended for use with the <b>CC2500</b> only. It can not be used to provide supply voltage to other devices.
6	GDO0 (ATEST)	Digital I/O	Digital output pin for general use: <ul style="list-style-type: none"> <li>• Test signals</li> <li>• FIFO status signals</li> <li>• Clear Channel Indicator</li> <li>• Clock output, down-divided from XOSC</li> <li>• Serial output RX data</li> <li>• Serial input TX data</li> </ul> Also used as analog test I/O for prototype/production testing
7	CS <sub>n</sub>	Digital Input	Serial configuration interface, chip select
8	XOSC_Q1	Analog I/O	Crystal oscillator pin 1, or external clock input
9	AVDD	Power (Analog)	1.8 - 3.6 V analog power supply connection
10	XOSC_Q2	Analog I/O	Crystal oscillator pin 2
11	AVDD	Power (Analog)	1.8 - 3.6 V analog power supply connection
12	RF_P	RF I/O	Positive RF input signal to LNA in receive mode Positive RF output signal from PA in transmit mode
13	RF_N	RF I/O	Negative RF input signal to LNA in receive mode Negative RF output signal from PA in transmit mode
14	AVDD	Power (Analog)	1.8 - 3.6 V analog power supply connection
15	AVDD	Power (Analog)	1.8 - 3.6 V analog power supply connection
16	GND	Ground (Analog)	Analog ground connection
17	RBIAS	Analog I/O	External bias resistor for reference current
18	DGUARD	Power (Digital)	Power supply connection for digital noise isolation
19	GND	Ground (Digital)	Ground connection for digital noise isolation
20	SI	Digital Input	Serial configuration interface, data input

**Table 13: Pinout overview**

## 6 Circuit Description

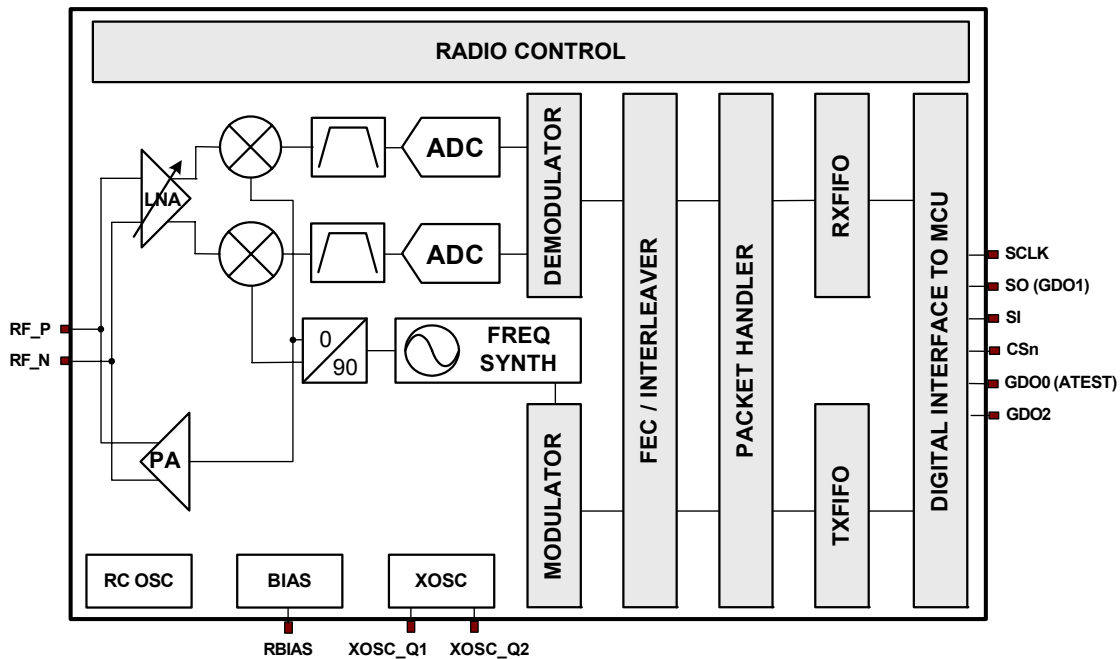


Figure 2: **CC2500** simplified block diagram

A simplified block diagram of **CC2500** is shown in Figure 2.

**CC2500** features a low-IF receiver. The received RF signal is amplified by the low-noise amplifier (LNA) and down-converted in quadrature (I and Q) to the intermediate frequency (IF). At IF, the I/Q signals are digitised by the ADCs. Automatic gain control (AGC), fine channel filtering, demodulation bit/packet synchronization is performed digitally.

The transmitter part of **CC2500** is based on direct synthesis of the RF frequency.

The frequency synthesizer includes a completely on-chip LC VCO and a 90 degrees

phase shifter for generating the I and Q LO signals to the down-conversion mixers in receive mode.

A crystal is to be connected to XOSC\_Q1 and XOSC\_Q2. The crystal oscillator generates the reference frequency for the synthesizer, as well as clocks for the ADC and the digital part.

A 4-wire SPI serial interface is used for configuration and data buffer access.

The digital baseband includes support for channel configuration, packet handling and data buffering.

## 7 Application Circuit

Only a few external components are required for using the **CC2500**. The recommended application circuit is shown in Figure 3. The external components are described in Table 14, and typical values are given in Table 15. Note that the PCB antenna alternative indicated in Figure 3 is preliminary and subject to changes. Performance for the PCB antenna alternative will be included in future revisions of this data sheet.

### Bias resistor

The bias resistor R171 is used to set an accurate bias current.

### Balun and RF matching

C122, C132, L121 and L131 form a balun that converts the differential RF signal on **CC2500** to a single-ended RF signal (C121 and C131 are also needed for DC blocking). Together



with an appropriate LC network, the balun components also transform the impedance to match a 50 Ω antenna (or cable). Component values for the RF balun and LC network are easily found using the SmartRF® Studio software. Suggested values are listed in Table 15.

**Crystal**

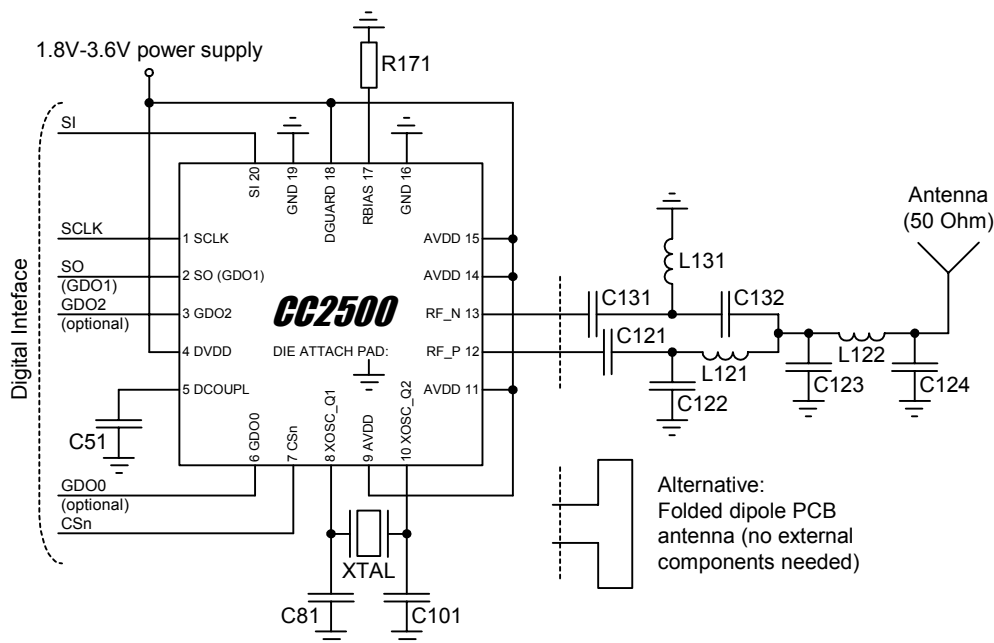
The crystal oscillator uses an external crystal with two loading capacitors (C81 and C101). See Section 26 on page 44 for details.

**Power supply decoupling**

The power supply must be properly decoupled close to the supply pins. Note that decoupling capacitors are not shown in the application circuit. The placement and the size of the decoupling capacitors are very important to achieve the optimum performance. Chipcon provides a reference design that should be followed closely.

Component	Description
C51	Decoupling capacitor for on-chip voltage regulator to digital part
C81/C101	Crystal loading capacitors, see Section 26 on page 44 for details
C121/C131	RF balun DC blocking capacitors
C122/C132	RF balun/matching capacitors
C123/C124	RF LC filter/matching capacitors
L121/L131	RF balun/matching inductors (inexpensive multi-layer type)
L122	RF LC filter inductor (inexpensive multi-layer type)
R171	Resistor for internal bias current reference
XTAL	26-27 MHz crystal, see Section 26 on page 44 for details

**Table 14: Overview of external components (excluding supply decoupling capacitors)**



**Figure 3: Typical application and evaluation circuit (excluding supply decoupling capacitors)**

Component	Value
C51	100 nF ±10%, 0402 X5R
C81	27 pF ±5%, 0402 NP0
C101	27 pF ±5%, 0402 NP0
C121	100 pF ±5%, 0402 NP0
C122	1.0 pF ±0.25 pF, 0402 NP0
C123	1.8 pF ±0.25 pF, 0402 NP0
C124	1.5 pF ±0.25 pF, 0402 NP0
C131	100 pF ±5%, 0402 NP0
C132	1.0 pF ±0.25 pF, 0402 NP0
L121	1.2 nH ±0.3 nH, 0402 monolithic, Murata LQG-15 series
L122	1.2 nH ±0.3 nH, 0402 monolithic, Murata LQG-15 series
L131	1.2 nH ±0.3 nH, 0402 monolithic, Murata LQG-15 series
R171	56 kΩ ±1%, 0402
XTAL	26.0 MHz surface mount crystal

**Table 15: Bill Of Materials for the application circuit**

In the CC2500EM reference design LQG-15 series inductors from Murata have been used. Measurements have been performed with

multi-layer inductors from other manufacturers (e.g. Würth) and the measurement results were the same as when using the Murata part.

## 8 Configuration Overview

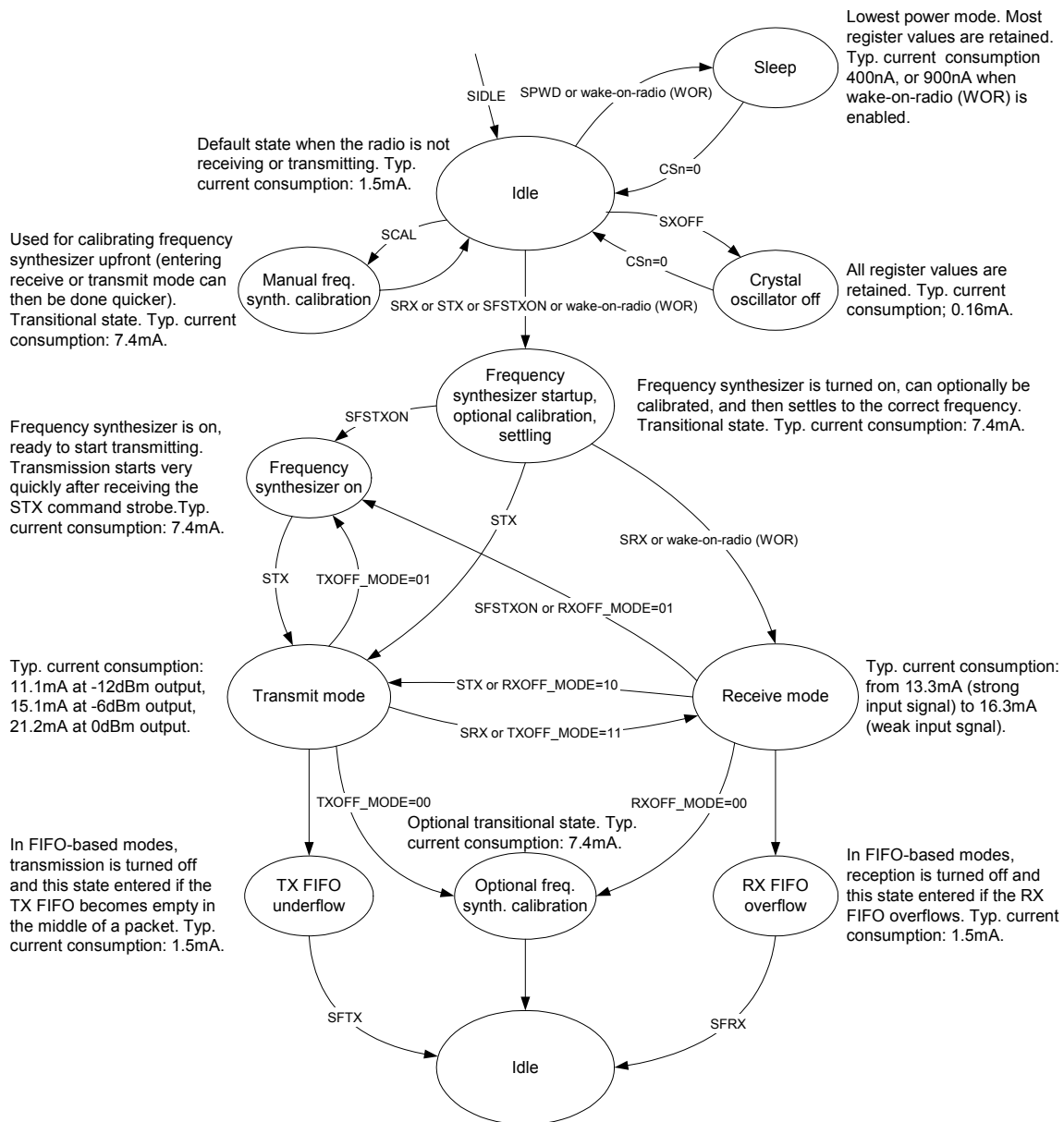
**CC2500** can be configured to achieve optimum performance for many different applications. Configuration is done using the SPI interface. The following key parameters can be programmed:

- Power-down / power up mode
- Crystal oscillator power-up / power-down
- Receive / transmit mode
- RF channel selection
- Data rate
- Modulation format
- RX channel filter bandwidth
- RF output power
- Data buffering with separate 64-byte receive and transmit FIFOs

- Packet radio hardware support
- Forward Error Correction with interleaving
- Data Whitening
- Wake-On-Radio (WOR)

Details of each configuration register can be found in Section 31, starting on page 49.

Figure 4 shows a simplified state diagram that explains the main **CC2500** states, together with typical usage and current consumption. For detailed information on controlling the **CC2500** state machine, and a complete state diagram, see Section 19, starting on page 34.



**Figure 4: Simplified state diagram, with typical usage and current consumption at 250 kbps data rate and MDMCFG2.DEM\_DCFILT\_OFF = 1 (reduced current)**

## 9 Configuration Software

**CC2500** can be configured using the SmartRF® Studio software, available for download from <http://www.chipcon.com>. The SmartRF® Studio software is highly recommended for obtaining

optimum register settings, and for evaluating performance and functionality. A screenshot of the SmartRF® Studio user interface for **CC2500** is shown in Figure 5.

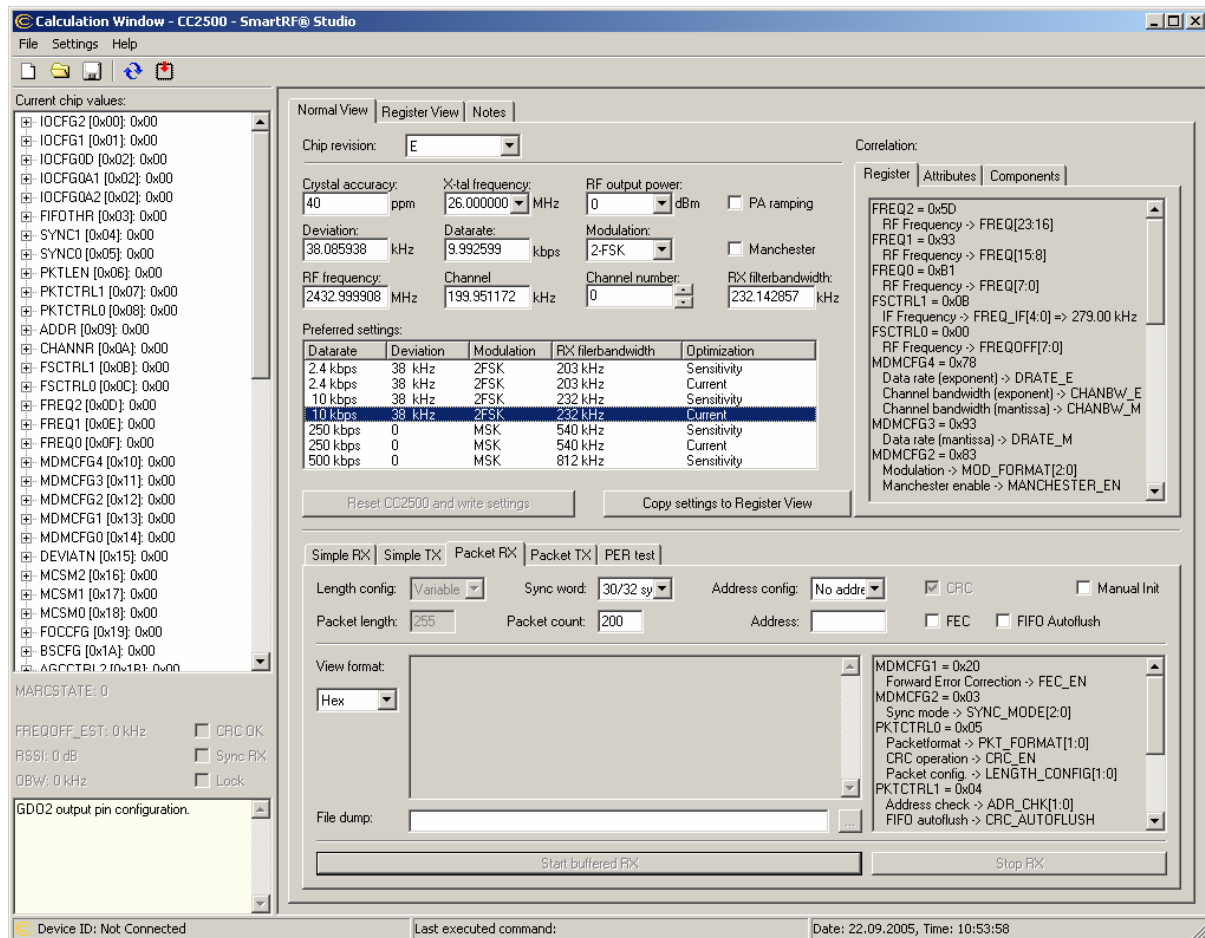


Figure 5: SmartRF® Studio user interface

## 10 4-wire Serial Configuration and Data Interface

**CC2500** is configured via a simple 4-wire SPI-compatible interface (SI, SO, SCLK and CS<sub>n</sub>) where **CC2500** is the slave. This interface is also used to read and write buffered data. All address and data transfer on the SPI interface is done most significant bit first.

All transactions on the SPI interface start with a header byte containing a read/write bit, a burst access bit and a 6-bit address.

During address and data transfer, the CS<sub>n</sub> pin (Chip Select, active low) must be kept low. If CS<sub>n</sub> goes high during the access, the transfer

will be cancelled. The timing for the address and data transfer on the SPI interface is shown in Figure 6 with reference to Table 16.

When CS<sub>n</sub> goes low, the MCU must wait until **CC2500** SO pin goes low before starting to transfer the header byte. This indicates that the voltage regulator has stabilized and the crystal is running. Unless the chip was in the SLEEP or XOFF states, the SO pin will always go low immediately after taking CS<sub>n</sub> low.

Figure 7 gives a brief overview of different register access types possible.

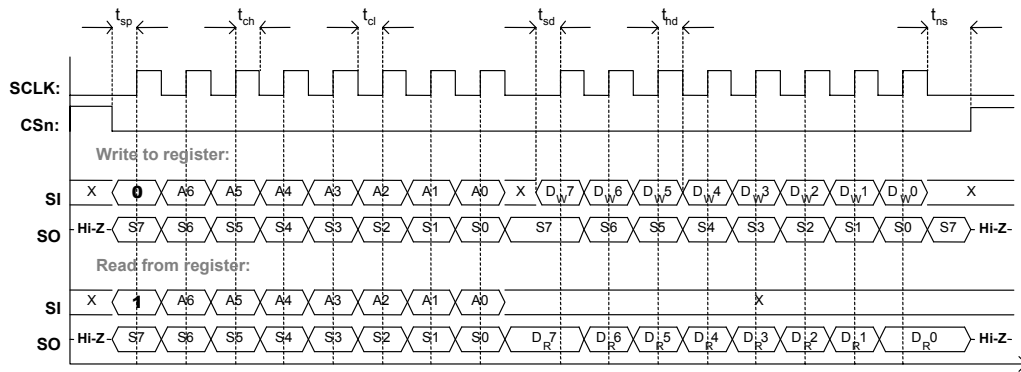


Figure 6: Configuration registers write and read operations (A6 is the “burst” bit)

Parameter	Description	Min	Max	Units
$F_{SCLK}$	SCLK frequency	0	10	MHz
$t_{sp,pd}$	CSn low to positive edge on SCLK, in power-down mode	TBD	-	$\mu$ s
$t_{sp}$	CSn low to positive edge on SCLK, in active mode	TBD	-	ns
$t_{ch}$	Clock high	50	-	ns
$t_{cl}$	Clock low	50	-	ns
$t_{rise}$	Clock rise time	-	TBD	ns
$t_{fall}$	Clock fall time	-	TBD	ns
$t_{sd}$	Setup data to positive edge on SCLK	TBD	-	ns
$t_{hd}$	Hold data after positive edge on SCLK	TBD	-	ns
$t_{ns}$	Negative edge on SCLK to CSn high.	TBD	-	ns

Table 16: SPI interface timing requirements

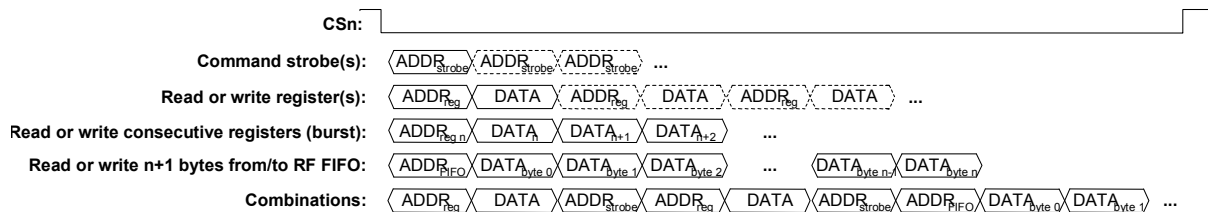


Figure 7: Register access types

### 10.1 Chip Status Byte

When the header byte, data byte or command strobe is sent on the SPI interface, the chip status byte is sent by the **CC2500** on the SO pin. The status byte contains key status signals, useful for the MCU. The first bit, s7, is

the CHIP\_RDYn signal; this signal must go low before the first positive edge of SCLK. The CHIP\_RDYn signal indicates that the crystal is running and the regulated digital supply voltage is stable.

Bits 6, 5 and 4 comprise the `STATE` value. This value reflects the state of the chip. The XOSC and power to the digital core is on in the IDLE state, but all other modules are in power down. The frequency and channel configuration should only be updated when the chip is in this state. The RX state will be active when the chip is in receive mode. Likewise, TX is active when the chip is transmitting.

The last four bits (3:0) in the status byte contains `FIFO_BYTES_AVAILABLE`. For read operations, the `FIFO_BYTES_AVAILABLE` field contains the number of bytes available for reading from the RX FIFO. For write operations, the `FIFO_BYTES_AVAILABLE` field contains the number of bytes free for writing into the TX FIFO. When `FIFO_BYTES_AVAILABLE=15`, 15 or more bytes are available/free.

Table 17 gives a status byte summary.

Bits	Name	Description																											
7	CHIP_RDYn	Stays high until power and crystal have stabilized. Should always be low when using the SPI interface.																											
6:4	STATE[2:0]	Indicates the current main state machine mode <table border="1"> <thead> <tr> <th>Value</th> <th>State</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>IDLE</td> <td>Idle state (Also reported for some transitional states instead of SETTLING or CALIBRATE)</td> </tr> <tr> <td>001</td> <td>RX</td> <td>Receive mode</td> </tr> <tr> <td>010</td> <td>TX</td> <td>Transmit mode</td> </tr> <tr> <td>011</td> <td>FSTXON</td> <td>Frequency synthesizer is on, ready to start transmitting</td> </tr> <tr> <td>100</td> <td>CALIBRATE</td> <td>Frequency synthesizer calibration is running</td> </tr> <tr> <td>101</td> <td>SETTLING</td> <td>PLL is settling</td> </tr> <tr> <td>110</td> <td>RXFIFO_OVERFLOW</td> <td>RX FIFO has overflowed. Read out any useful data, then flush the FIFO with <code>SFRX</code></td> </tr> <tr> <td>111</td> <td>TXFIFO_UNDERFLOW</td> <td>TX FIFO has underflowed. Acknowledge with <code>SFTX</code></td> </tr> </tbody> </table>	Value	State	Description	000	IDLE	Idle state (Also reported for some transitional states instead of SETTLING or CALIBRATE)	001	RX	Receive mode	010	TX	Transmit mode	011	FSTXON	Frequency synthesizer is on, ready to start transmitting	100	CALIBRATE	Frequency synthesizer calibration is running	101	SETTLING	PLL is settling	110	RXFIFO_OVERFLOW	RX FIFO has overflowed. Read out any useful data, then flush the FIFO with <code>SFRX</code>	111	TXFIFO_UNDERFLOW	TX FIFO has underflowed. Acknowledge with <code>SFTX</code>
Value	State	Description																											
000	IDLE	Idle state (Also reported for some transitional states instead of SETTLING or CALIBRATE)																											
001	RX	Receive mode																											
010	TX	Transmit mode																											
011	FSTXON	Frequency synthesizer is on, ready to start transmitting																											
100	CALIBRATE	Frequency synthesizer calibration is running																											
101	SETTLING	PLL is settling																											
110	RXFIFO_OVERFLOW	RX FIFO has overflowed. Read out any useful data, then flush the FIFO with <code>SFRX</code>																											
111	TXFIFO_UNDERFLOW	TX FIFO has underflowed. Acknowledge with <code>SFTX</code>																											
3:0	FIFO_BYTES_AVAILABLE[3:0]	The number of bytes available in the RX FIFO or free bytes in the TX FIFO (depends on the read/write-bit). If <code>FIFO_BYTES_AVAILABLE=15</code> , there are 15 or more bytes in RX FIFO or 49 or less bytes in the TX FIFO.																											

**Table 17: Status byte summary**

## 10.2 Register Access

The configuration registers of the **CC2500** are located on SPI addresses from `0x00` to `0x2F`. Table 35 on page 51 lists all configuration registers. The detailed description of each register is found in Section 31.1, starting on page 54. All configuration registers can be both written to and read. The read/write bit controls if the register should be written to or read. When writing to registers, the status byte is sent on the `SO` pin each time a header byte or data byte is transmitted on the `SI` pin. When reading from registers, the status byte is sent on the `SO` pin each time a header byte is transmitted on the `SI` pin.

Registers with consecutive addresses can be accessed in an efficient way by setting the burst bit in the address header. The address sets the start address in an internal address counter. This counter is incremented by one each new byte (every 8 clock pulses). The burst access is either a read or a write access and must be terminated by setting `CSn` high.

For register addresses in the range `0x30-0x3D`, the "burst" bit is used to select between status registers and command strobes (see below). The status registers can only be read. Burst read is not available for status registers, so they must be read one at a time.

### 10.3 Command Strobes

Command Strobes may be viewed as single byte instructions to **CC2500**. By addressing a Command Strobe register, internal sequences will be started. These commands are used to disable the crystal oscillator, enable receive mode, enable wake-on-radio etc. The 14 command strobes are listed in Table 34 on page 50.

The command strobe registers are accessed in the same way as for a register write operation, but no data is transferred. That is, only the R/W bit (set to 0), burst access (set to 0) and the six address bits (in the range 0x30 through 0x3D) are written. A command strobe may be followed by any other SPI access without pulling  $CS_n$  high. The command strobes are executed immediately, with the exception of the *SPWD* and the *SXOFF* strobes that are executed when  $CS_n$  goes high.

When writing command strobes, the status byte is sent on the *SO* pin.

### 10.4 FIFO Access

The 64-byte TX FIFO and the 64-byte RX FIFO are accessed through the 0x3F address. When the read/write bit is zero, the TX FIFO is accessed, and the RX FIFO is accessed when the read/write bit is one.

The TX FIFO is write-only, while the RX FIFO is read-only.

The burst bit is used to determine if FIFO access is single byte or a burst access. The single byte access method expects address with burst bit set to zero and one data byte. After the data byte a new address is expected; hence,  $CS_n$  can remain low. The burst access method expects one address byte and then consecutive data bytes until terminating the access by setting  $CS_n$  high.

The following header bytes access the FIFOs:

- 0x3F: Single byte access to TX FIFO
- 0x7F: Burst access to TX FIFO
- 0xBF: Single byte access to RX FIFO
- 0xFF: Burst access to RX FIFO

When writing to the TX FIFO, the status byte (see Section 10.1) is output for each new data byte on *SO*, as shown in Figure 6. This status byte can be used to detect TX FIFO underflow

while writing data to the TX FIFO. Note that the status byte contains the number of bytes free *before* writing the byte in progress to the TX FIFO. When the last byte that fits in the TX FIFO is transmitted to the *SI* pin, the status byte received concurrently on the *SO* pin will indicate that one byte is free in the TX FIFO.

The transmit FIFO may be flushed by issuing a *SFTX* command strobe. Similarly, a *SFRX* command strobe will flush the receive FIFO. A *SFTX* or *SFRX* command strobe can only be issued in the *IDLE*, *TXFIFO\_UNDERFLOW* or *RXFIFO\_OVERFLOW* state. Both FIFOs are flushed when going to the *SLEEP* state.

### 10.5 PATABLE Access

The 0x3E address is used to access the *PATABLE*, which is used for selecting PA power control settings. The SPI expects up to eight data bytes after receiving the address. By programming the *PATABLE*, controlled PA power ramp-up and ramp-down can be achieved. See Section 0 on page 40 for output power programming details.

The *PATABLE* is an 8-byte table that defines the PA control settings to use for each of the eight PA power values (selected by the 3-bit value *FREND0.PA\_POWER*). The table is written and read from the lowest setting (0) to the highest (7), one byte at a time. An index counter is used to control the access to the table. This counter is incremented each time a byte is read or written to the table, and set to the lowest index when  $CS_n$  is high. When the highest value is reached the counter restarts at zero.

The access to the *PATABLE* is either single byte or burst access depending on the burst bit. When using burst access the index counter will count up; when reaching 7 the counter will restart at 0. The read/write bit controls whether the access is a write access (*R/W*=0) or a read access (*R/W*=1).

If one byte is written to the *PATABLE* and this value is to be read out then  $CS_n$  must be set high before the read access in order to set the index counter back to zero.

Note that the content of the *PATABLE* is lost when entering the *SLEEP* state, except for the first byte (index 0).

## 11 Microcontroller Interface and Pin Configuration

In a typical system, **CC2500** will interface to a microcontroller. This microcontroller must be able to:

- Program **CC2500** into different modes,
- Read and write buffered data
- Read back status information via the 4-wire SPI-bus configuration interface (SI, SO, SCLK and CSn).

### 11.1 Configuration Interface

The microcontroller uses four I/O pins for the SPI configuration interface (SI, SO, SCLK and CSn). The SPI is described in Section 10 on page 19.

### 11.2 General Control and Status Pins

The **CC2500** has two dedicated configurable pins and one shared pin that can output internal status information useful for control software. These pins can be used to generate interrupts on the MCU. See Section 28 on page 45 for more details on the signals that can be programmed. The dedicated pins are called GDO0 and GDO2. The shared pin is the SO pin in the SPI interface. The default setting for GDO1/SO is 3-state output. By selecting any other of the programming options the GDO1/SO pin will become a generic pin. When CSn is low, the pin will always function as a normal SO pin.

In the synchronous and asynchronous serial modes, the GDO0 pin is used as a serial TX data input pin while in transmit mode.

The GDO0 pin can also be used for an on-chip analog temperature sensor. By measuring the voltage on the GDO0 pin with an external ADC, the temperature can be calculated. Specifications for the temperature sensor are found in Section 4.7 on page 12.

The temperature sensor output is only available when the frequency synthesizer is enabled (e.g. the MANCAL, FSTXON, RX and TX states). It is necessary to write 0xBF to the PTEST register to use the analog temperature sensor in the IDLE state. Before leaving the

IDLE state, the PTEST register should be restored to its default value (0x7F).

### 11.3 Optional Radio Control Feature

The **CC2500** has an optional way of controlling the radio, by reusing SI, SCLK and CSn from the SPI interface. This feature allows for a simple three-pin control of the major states of the radio: SLEEP, IDLE, RX and TX.

This optional functionality is enabled with the MCSM0.PIN\_CTRL\_EN configuration bit.

State changes are commanded as follows: When CSn is high the SI and SCLK is set to the desired state according to Table 18. When CSn goes low the state of SI and SCLK is latched and a command strobe is generated internally according to the control coding. It is only possible to change state with this functionality. That means that for instance RX will not be restarted if SI and SCLK are set to RX and CSn toggles. When CSn is low the SI and SCLK has normal SPI functionality.

All pin control command strobes are executed immediately, except the SPWD strobe, which is delayed until CSn goes high.

CSn	SCLK	SI	Function
1	X	X	Chip unaffected by SCLK/SI
↓	0	0	Generates SPWD strobe
↓	0	1	Generates STX strobe
↓	1	0	Generates SIDLE strobe
↓	1	1	Generates SRX strobe
0	SPI mode	SPI mode	SPI mode (wakes up into IDLE if in SLEEP/XOFF)

Table 18: Optional pin control coding



## 12 Data Rate Programming

The data rate used when transmitting, or the data rate expected in receive is programmed by the `MDMCFG3.DRATE_M` and the `MDMCFG4.DRATE_E` configuration registers. The data rate is given by the formula below. As the formula shows, the programmed data rate depends on the crystal frequency.

$$R_{DATA} = \frac{(256 + DRATE\_M) \cdot 2^{DRATE\_E}}{2^{28}} \cdot f_{XOSC}$$

The following approach can be used to find suitable values for a given data rate:

$$DRATE\_E = \left\lceil \log_2 \left( \frac{R_{DATA} \cdot 2^{20}}{f_{XOSC}} \right) \right\rceil$$

$$DRATE\_M = \frac{R_{DATA} \cdot 2^{28}}{f_{XOSC} \cdot 2^{DRATE\_E}} - 256$$

If `DRATE_M` is rounded to the nearest integer and becomes 256, increment `DRATE_E` and use `DRATE_M=0`.

The data rate can be set from 1.2 kbps to 500 kbps with the minimum step size of:

Data rate start	Typical data rate	Data rate stop	Data rate step size
0.8 kbps	1.2/2.4 kbps	3.17 kbps	0.0062 kbps
3.17 kbps	4.8 kbps	6.35 kbps	0.0124 kbps
6.35 kbps	9.6 kbps	12.7 kbps	0.0248 kbps
12.7 kbps	19.6 kbps	25.4 kbps	0.0496 kbps
25.4 kbps	38.4 kbps	50.8 kbps	0.0992 kbps
50.8 kbps	76.8 kbps	101.6 kbps	0.1984 kbps
101.6 kbps	153.6 kbps	203.1 kbps	0.3967 kbps
203.1 kbps	250 kbps	406.3 kbps	0.7935 kbps
406.3 kbps	500 kbps	500 kbps	1.5869 kbps

**Table 19: Data rate step size**

## 13 Receiver Channel Filter Bandwidth

In order to meet different channel width requirements, the receiver channel filter is programmable. The `MDMCFG4.CHANBW_E` and `MDMCFG4.CHANBW_M` configuration registers control the receiver channel filter bandwidth, which scales with the crystal oscillator frequency. The following formula gives the relation between the register settings and the channel filter bandwidth:

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

The **CC2500** supports the following channel filter bandwidths:

MDMCFG4. CHANBW_M	MDMCFG4.CHANBW_E			
	00	01	10	11
00	812	406	203	102
01	650	325	162	81
10	541	270	135	68
11	464	232	116	58

**Table 20: Channel filter bandwidths [kHz] (assuming a 26 MHz crystal)**

Above 300 kHz bandwidth, however, the sensitivity and blocking performance may be somewhat degraded. For best performance, the channel filter bandwidth should be selected so that the signal bandwidth occupies at most 80% of the channel filter bandwidth. The channel centre tolerance due to crystal accuracy should also be subtracted from the signal bandwidth. The following example illustrates this:

With the channel filter bandwidth set to 600 kHz, the signal should stay within 80% of 600

kHz, which is 480 kHz. Assuming 2.44 GHz frequency and  $\pm 20$  ppm frequency uncertainty for both the transmitting device and the receiving device, the total frequency uncertainty is  $\pm 40$  ppm of 2.44 GHz, which is

$\pm 98$  kHz. If the whole transmitted signal bandwidth is to be received within 480 kHz, the transmitted signal bandwidth should be maximum 480 kHz–298 kHz, which is 284 kHz.

## 14 Demodulator, Symbol Synchronizer and Data Decision

**CC2500** contains an advanced and highly configurable demodulator. Channel filtering and frequency offset compensation is performed digitally. To generate the RSSI level (see Section 17.3 for more information) the signal level in the channel is estimated. Data filtering is also included for enhanced performance.

### 14.1 Frequency Offset Compensation

When using FSK, GFSK or MSK modulation, the demodulator will compensate for the offset between the transmitter and receiver frequency, within certain limits, by estimating the centre of the received data. This value is available in the `FREQEST` status register. Writing the value from `FREQEST` into `FSCTRL0.FREQOFF` the frequency synthesizer is automatically adjusted according to the estimated frequency offset.

Note that frequency offset compensation is not supported for OOK modulation.

### 14.2 Bit Synchronization

The bit synchronization algorithm extracts the clock from the incoming symbols. The algorithm requires that the expected data rate is programmed as described in Section 12 on page 24. Re-synchronization is performed continuously to adjust for error in the incoming symbol rate.

### 14.3 Byte Synchronization

Byte synchronization is achieved by a continuous sync word search. The sync word is a 16 or 32 bit configurable field that is automatically inserted at the start of the packet by the modulator in transmit mode. The demodulator uses this field to find the byte boundaries in the stream of bits. The sync word will also function as a system identifier, since only packets with the correct predefined sync word will be received. The sync word detector correlates against the user-configured 16-bit sync word. The correlation threshold can be set to 15/16 bits match or 16/16 bits match. The sync word can be further qualified using the preamble quality indicator mechanism described below and/or a carrier sense condition. The sync word is programmed with `SYNC1` and `SYNC0`.

In order to make false detections of sync words less likely, a mechanism called preamble quality indication (PQI) can be used to qualify the sync word. A threshold value for the preamble quality must be exceeded in order for a detected sync word to be accepted. See Section 17.2 on page 30 for more details.

## 15 Packet Handling Hardware Support

The **CC2500** has built-in hardware support for packet oriented radio protocols.

In transmit mode, the packet handler will add the following elements to the packet stored in the TX FIFO:

- A programmable number of preamble bytes. 4 preamble bytes is recommended.

- A two byte Synchronization Word. Can be duplicated to give a 4-byte sync word. (Recommended).
- Optionally whiten the data with a PN9 sequence.
- Optionally Interleave and Forward Error Code the data.
- Optionally compute and add a CRC checksum over the data field.

In receive mode, the packet handling support will de-construct the data packet:

- Preamble detection.
- Sync word detection.
- Optional one byte address check.
- Optionally compute and check CRC.
- Optionally append two status bytes (see Table 21 and Table 22) with RSSI value, Link Quality Indication and CRC status.

Bit	Field name	Description
7:0	RSSI	RSSI value

**Table 21: Received packet status byte 1 (first byte appended after the data)**

Bit	Field name	Description
7	CRC_OK	1: CRC for received data OK (or CRC disabled) 0: CRC error in received data
6:0	LQI	The Link Quality Indicator estimates how easily a received signal can be demodulated

**Table 22: Received packet status byte 2 (second byte appended after the data)**

Note that register fields that control the packet handling features should only be altered when **CC2500** is in the IDLE state.

### 15.1 Data Whitening

From a radio perspective, the ideal over the air data are random and DC free. This results in the smoothest power distribution over the occupied bandwidth. This also gives the regulation loops in the receiver uniform operation conditions (no data dependencies).

Real world data often contain long sequences of zeros and ones. Performance can then be improved by whitening the data before transmitting, and de-whitening in the receiver. With **CC2500**, this can be done automatically by setting `PKTCTRL0.WHITE_DATA=1`. All data, except the preamble and the sync word, are then XOR-ed with a 9-bit pseudo-random (PN9) sequence before being transmitted. At the receiver end, the data are XOR-ed with the same pseudo-random sequence. This way, the whitening is reversed, and the original data appear in the receiver.

Setting `PKTCTRL0.WHITE_DATA=1` is recommended for all uses, except when over-the-air compatibility with other systems is needed.

### 15.2 Packet Format

The format of the data packet can be configured and consists of the following items:

- Preamble
- Synchronization word
- Length byte or constant programmable packet length
- Optional address byte
- Payload
- Optional 2 byte CRC

The preamble pattern is an alternating sequence of ones and zeros (01010101...). The minimum length of the preamble is programmable. When enabling TX, the modulator will start transmitting the preamble. When the programmed number of preamble bytes has been transmitted, the modulator will send the sync word and then data from the TX FIFO if data is available. If the TX FIFO is empty, the modulator will continue to send preamble bytes until the first byte is written to the TX FIFO. The modulator will then send the sync word and then the data bytes. The number of preamble bytes is programmed with the `MDMCFG1.NUM_PREAMBLE` value.

The synchronization word is a two-byte value set in the `SYNC1` and `SYNC0` registers. The sync word provides byte synchronization of the incoming packet. A one-byte sync word can be emulated by setting the `SYNC1` value to the preamble pattern. It is also possible to emulate a 32 bit sync word by using `MDMCFG2.SYNC_MODE=3` or 7. The sync word will then be repeated twice.

**CC2500** supports both fixed packet length protocols and variable packet length protocols. Variable or fixed packet length mode can be used for packet up to 255 bytes. For longer packets, infinite packet length mode must be used.

Fixed packet length mode is selected by setting `PKTCTRL0.LENGTH_CONFIG=0`. The desired packet length is set by the `PKTLEN` register. The packet length is defined as the payload data, excluding the length byte and the optional automatic CRC. In variable length mode, `PKTCTRL0.LENGTH_CONFIG=1`, the packet length is configured by the first byte after the sync word. The `PKTLEN` register is

used to set the maximum packet length allowed in RX. Any packet received with a length byte with a value greater than `PKTLEN` will be discarded.

With `PKTCTRL0.LENGTH_CONFIG=2`, the packet length is set to infinite and transmission and reception will continue until turned off manually. The infinite mode can be turned off while a packet is being transmitted or received. As described in the next section, this can be used to support packet formats with different length configuration than natively supported by **CC2500**.

### 15.2.1 Arbitrary Length Field Configuration

The fixed length field can be reprogrammed during receive and transmit. This opens the possibility to have a different length field configuration than supported for variable length packets. At the start of reception, the packet length is set to a large value. The MCU reads out enough bytes to interpret the length field in the packet. Then the `PKTLEN` value is set according to this value. The end of packet will occur when the byte counter in the packet handler is equal to the `PKTLEN` register. Thus, the MCU must be able to program the correct length, before the internal counter reaches the packet length.

By utilizing the infinite packet length option, arbitrary packet length is available. At the start of the packet, the infinite mode must be active. When less than 256 bytes remains of the packet, the MCU sets the `PKTLEN` register to  $\text{mod}(\text{length}, 256)$ , disables infinite packet length and activates fixed length packets. When the internal byte counter reaches the `PKTLEN` value, the transmission or reception ends. Automatic CRC appending/checking can be used (by setting `PKTCTRL0.CRC_EN` to 1).

When for example a 454-byte packet is to be transmitted, the MCU does the following:

- Set `PKTCTRL0.LENGTH_CONFIG=2` (10).
- Pre-program the `PKTLEN` register to  $\text{mod}(454,256)=198$ .
- Transmit at least 198 bytes, for example by filling the 64-byte TX FIFO four times (256 bytes transmitted).
- Set `PKTCTRL0.LENGTH_CONFIG=0` (00).
- The transmission ends when the packet counter reaches 198. A total of  $256+198=454$  bytes are transmitted.

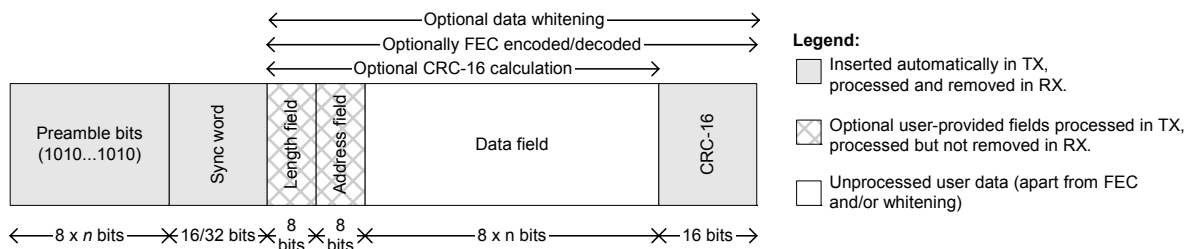


Figure 8: Packet format

### 15.3 Packet Filtering in Receive Mode

**CC2500** supports three different packet-filtering criteria: address filtering, maximum length filtering and CRC filtering.

#### 15.3.1 Address Filtering

Setting `PKTCTRL1.ADR_CHK` to any other value than zero enables the packet address filter. The packet handler engine will compare the destination address byte in the packet with the programmed node address in the `ADDR` register and the 0x00 broadcast address when `PKTCTRL1.ADR_CHK=10` or both 0x00 and

0xFF broadcast addresses when `PKTCTRL1.ADR_CHK=11`. If the received address matches a valid address, the packet is received and written into the RX FIFO. If the address match fails, the packet is discarded and receive mode restarted (regardless of the `MCSM1.RXOFF_MODE` setting).

#### 15.3.2 Maximum Length Filtering

In the variable packet length mode the `PKTLEN.PACKET_LENGTH` register value is used to set the maximum allowed packet length. If the received length byte has a larger value than this, the packet is discarded and

receive mode restarted (regardless of the `MCSM1.RXOFF_MODE` setting).

### 15.3.3 CRC Filtering

The filtering of a packet when CRC check fails is enabled with `PKTCTRL1.CRC_AUTOFLUSH`. The CRC auto flush function will flush the *entire* RX FIFO if the CRC check fails. After auto flushing the RX FIFO, the next state depends on the `MCSM1.RXOFF_MODE` setting.

When using the auto flush function, the maximum packet length is 63 bytes in variable packet length mode and 64 bytes in fixed packet length mode. Note that the maximum allowed packet length is reduced by two bytes when `PKTCTRL1.APPEND_STATUS` is enabled, to make room in the RX FIFO for the two status bytes appended at the end of the packet. Since the entire RX FIFO is flushed when the CRC check fails, the previously received packet must be read out of the FIFO before receiving the current packet. The MCU must not read from the current packet until the CRC has been checked as OK.

## 15.4 CRC Check

It is possible to read back the CRC status in 2 different ways:

1) Set `PKTCTRL1.APPEND_STATUS=1` and read the `CRC_OK` flag in the MSB of the second byte appended to the RX FIFO after the packet data. This requires double buffering of the packet, i.e. the entire packet content of the RX FIFO must be completely read out before it is possible to check whether the CRC indication is OK or not.

2) To avoid reading the whole RX FIFO, another solution is to use the `PKTCTRL1.CRC_AUTOFLUSH` feature. If this feature is enabled, the entire RX FIFO will be flushed if the CRC check fails. If `GDOx_CFG=0x06` the `GDOx` pin will be asserted when a sync word is found. The `GDOx` pin will be de-asserted at the end of the packet. When the latter occurs the MCU should read the number of bytes in the RX FIFO from the `RXBYTES.NUM_RXBYTES` status register. If `RXBYTES.NUM_RXBYTES=0` the CRC check failed and the FIFO was flushed. If `RXBYTES.NUM_RXBYTES>0` the CRC check was OK and data can be read out of the FIFO.

## 15.5 Packet Handling in Transmit Mode

The payload that is to be transmitted must be written into the TX FIFO. The first byte written must be the length byte when variable packet length is enabled. The length byte has a value equal to the payload of the packet (including the optional address byte). If fixed packet length is enabled, then the first byte written to the TX FIFO is interpreted as the destination address, if this feature is enabled in the device that receives the packet.

The modulator will first send the programmed number of preamble bytes. If data is available in the TX FIFO, the modulator will send the two-byte (optionally 4-byte) sync word and then the payload in the TX FIFO. If CRC is enabled, the checksum is calculated over all the data pulled from the TX FIFO and the result is sent as two extra bytes at the end of the payload data.

If whitening is enabled, the length byte, payload data and the two CRC bytes will be whitened. This is done before the optional FEC/Interleaver stage. Whitening is enabled by setting `PKTCTRL0.WHITE_DATA=1`.

If FEC/Interleaving is enabled, the length byte, payload data and the two CRC bytes will be scrambled by the interleaver, and FEC encoded before being modulated.

## 15.6 Packet Handling in Receive Mode

In receive mode, the demodulator and packet handler will search for a valid preamble and the sync word. When found, the demodulator has obtained both bit and byte synchronism and will receive the first payload byte.

If FEC/Interleaving is enabled, the FEC decoder will start to decode the first payload byte. The interleaver will de-scramble the bits before any other processing is done to the data.

If whitening is enabled, the data will be de-whitened at this stage.

When variable packet length is enabled, the first byte is the length byte. The packet handler stores this value as the packet length and receives the number of bytes indicated by the length byte. If fixed packet length is used, the packet handler will accept the programmed number of bytes.

Next, the packet handler optionally checks the address and only continues the reception if the address matches. If automatic CRC check is

enabled, the packet handler computes CRC and matches it with the appended CRC checksum.

At the end of the payload, the packet handler will optionally write two extra packet status bytes that contain CRC status, link quality indication and RSSI value.

## 16 Modulation Formats

**CC2500** supports amplitude, frequency and phase shift modulation formats. The desired modulation format is set in the `MDMCFG2.MOD_FORMAT` register.

Optionally, the data stream can be Manchester coded by the modulator and decoded by the demodulator. This option is enabled by setting `MDMCFG2.MANCHESTER_EN=1`. Manchester encoding is not supported at the same time as using the FEC/Interleaver option. Manchester coding can be used with the 2-ary modulation formats (2-FSK, GFSK, OOK and MSK).

### 16.1 Frequency Shift Keying

2-FSK can optionally be shaped by a Gaussian filter with `BT=1`, producing a GFSK modulated signal.

The frequency deviation is programmed with the `DEVIATION_M` and `DEVIATION_E` values in the `DEVIATN` register. The value has an exponent/mantissa form, and the resultant deviation is given by:

$$f_{dev} = \frac{f_{xosc}}{2^{17}} \cdot (8 + DEVIATION\_M) \cdot 2^{DEVIATION\_E}$$

The symbol encoding is shown in Table 23.

Format	Symbol	Coding
2-FSK/GFSK	'0'	- Deviation
	'1'	+ Deviation

**Table 23: Symbol encoding for FSK modulation**

### 16.2 Minimum Shift Keying

When using MSK<sup>1</sup>, the complete transmission (preamble, sync word and payload) will be MSK modulated.

Phase shifts are performed with a constant transition time. This means that the rate of change for the 180-degree transition is twice that of the 90-degree transition.

The fraction of a symbol period used to change the phase can be modified with the `DEVIATN.DEVIATION_M` setting. This is equivalent to changing the shaping of the symbol. Setting `DEVIATN.DEVIATION_M=7` will generate a standard shaped MSK signal.

The MSK modulation format implemented in **CC2500** inverts the sync word and data compared to e.g. signal generators.

### 16.3 Amplitude Modulation

The supported amplitude modulation On-Off Keying (OOK) simply turns on or off the PA to modulate 1 and 0 respectively.

<sup>1</sup> Identical to offset QPSK with half-sine shaping (data coding may differ)

## 17 Received Signal Qualifiers and Link Quality Information

**CC2500** has several qualifiers that can be used to increase the likelihood that a valid sync word is detected.

### 17.1 Sync Word Qualifier

If sync word detection in RX is enabled in register `MDMCFG2` the **CC2500** will not start filling the RX FIFO and perform the packet filtering described in Section 15.3 before a valid sync word has been detected. The sync word qualifier mode is set by `MDMCFG2.SYNC_MODE` and is summarized in Table 24. Carrier sense in Table 24 is described in Section 17.4.

MDMCFG2. SYNC_MODE	Sync word qualifier mode
000	No preamble/sync
001	15/16 sync word bits detected
010	16/16 sync word bits detected
011	30/32 sync word bits detected
100	No preamble/sync, carrier sense above threshold
101	15/16 + carrier sense above threshold
110	16/16 + carrier sense above threshold
111	30/32 + carrier sense above threshold

**Table 24: Sync word qualifier mode**

### 17.2 Preamble Quality Threshold (PQT)

The Preamble Quality Threshold (PQT) sync-word qualifier adds the requirement that the received sync word must be preceded with a preamble with a quality above a programmed threshold.

Another use of the preamble quality threshold is as a qualifier for the optional RX termination timer. See Section 19.7 on page 37 for details.

The preamble quality estimator increases an internal counter by one each time a bit is received that is different from the previous bit, and decreases the counter by 4 each time a bit is received that is the same as the last bit. The counter saturates at 0 and 31. The threshold is configured with the register field

`PKTCTRL1.PQT`. A threshold of  $4 \cdot PQT$  for this counter is used to gate sync word detection. By setting the value to zero, the preamble quality qualifier of the sync word is disabled.

A “Preamble Quality Reached” flag can also be observed on one of the GDO pins and in the `status` register bit `PKTSTATUS.PQT_REACHED`. This flag asserts when the received signal exceeds the PQT.

### 17.3 RSSI

The RSSI value is an estimate of the signal level in the current channel. This value is based on the current 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. The RSSI value is in dB with  $\frac{1}{2}$ dB resolution.

If `PKTCTRL1.APPEND_STATUS` is enabled, a snapshot of the RSSI during the first 8 bytes of the packet is automatically added to the end of each received packet.

The RSSI value read from the RSSI status register is a 2’s complement number. The following procedure can be used to convert the RSSI reading to an absolute power level (RSSI\_dBm).

- 1) Read the RSSI status register
- 2) Convert the reading from a hexadecimal number to a decimal number (RSSI\_dec)
- 3) If  $RSSI\_dec \geq 128$  then  $RSSI\_dBm = (RSSI\_dec - 256)/2 - RSSI\_offset$
- 4) Else if  $RSSI\_dec < 128$  then  $RSSI\_dBm = (RSSI\_dec)/2 - RSSI\_offset$

Table 25 gives typical values for the `RSSI_offset`.

Figure 9 shows typical plots of RSSI reading as a function of input power level for different data rates.

Data rate	RSSI_offset (decimal)
2.4 kbps	71
10 kbps	69
250 kbps	72
500 kbps	72

Table 25: Typical RSSI\_offset values

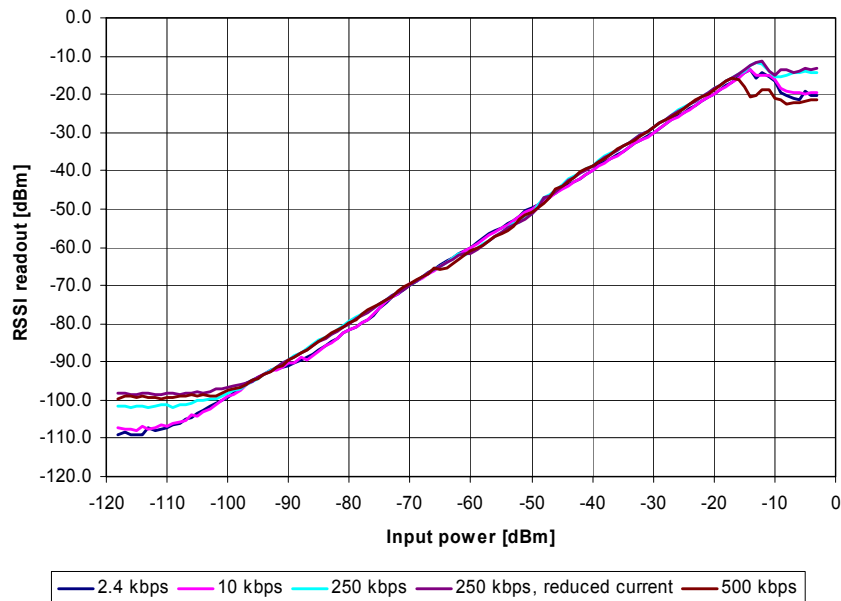


Figure 9: Typical RSSI value vs. input power level for some typical data rates

### 17.4 Carrier Sense (CS)

The Carrier Sense flag is used as a sync word qualifier and for CCA. The CS flag can be set based on two conditions, which can be individually adjusted:

- CS is asserted when the RSSI is above a programmable absolute threshold, and de-asserted when RSSI is below the same threshold (with hysteresis).
- CS is asserted when the RSSI has increased with a programmable number of dB from one RSSI sample to the next, and de-asserted when RSSI has decreased with the same number of dB. This setting is not dependent on the absolute signal level and is thus useful to detect signals in environments with a time varying noise floor.

Carrier Sense (CS) can be used as a sync word qualifier that requires the signal level to be higher than the threshold for a sync word

search to be performed. The signal can also be observed on one of the GDO pins and in the status register bit `PKTSTATUS.CS`.

Other uses of Carrier Sense include the TX-If-CCA function (see Section 17.5 on page 32) and the optional fast RX termination (see Section 19.7 on page 37).

CS can be used to avoid interference from e.g. WLAN.

#### 17.4.1 CS Absolute Threshold

The absolute threshold related to the RSSI value is given by:

$$THR_{RSSI} = MAGN\_TARGET + CARRIER\_SENSE\_ABS\_THR - GAIN_{MAX}$$

The maximum possible gain can be reduced using the `AGCCTRL2.MAX_LNA_GAIN` and `AGCCTRL2.MAX_DVGA_GAIN` register fields. `CARRIER_SENSE_ABS_THR` is programmable in 1 dB steps from -7 dB to +7dB. Table 26 and Table 27 show the RSSI readout values



at the CS threshold at 2.4 kbps and 250 kbps data rate respectively. The default `CARRIER_SENSE_ABS_THR = 0` (0 dB) and `MAGN_TARGET = 3` (33 dB) have been used.

		MAX_DVGA_GAIN[1:0]			
		00	01	10	11
MAX_LNA_GAIN[2:0]	000	-99	-93	-87	-81.5
	001	-97	-90.5	-85	-78.5
	010	-93.5	-87	-82	-76
	011	-91.5	-86	-80	-74
	100	-90.5	-84	-78	-72.5
	101	-88	-82.5	-76	-70
	110	-84.5	-78.5	-73	-67
	111	-82.5	-76	-70	-64

**Table 26: Typical RSSI value in dBm at CS threshold with default MAGN\_TARGET at 2.4 kbps**

		MAX_DVGA_GAIN[1:0]			
		00	01	10	11
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 27: Typical RSSI value in dBm at CS threshold with default MAGN\_TARGET at 250 kbps**

If the threshold is to be set high, e.g. only signals with good strength are wanted, the threshold should be adjusted upwards by first reducing the `MAX_LNA_GAIN` value and then the `MAX_DVGA_GAIN` value. This will reduce power consumption in the receiver front end, since the highest gain settings are avoided.

The `MAGN_TARGET` setting is a compromise between blocker tolerance/selectivity and sensitivity. The value sets the desired signal

level in the channel into the demodulator. Increasing this value reduces the headroom for blockers, and therefore close-in selectivity.

#### 17.4.2 CS relative threshold

The relative threshold detects sudden changes in the measured signal level. This setting is not dependent on the absolute signal level and is thus useful to detect signals in environments with a time varying noise floor. The register field `AGCCTRL1.CARRIER_SENSE_REL_THR` is used to enable/disable relative CS, and to select threshold of 6 dB, 10 dB or 14 dB RSSI change

### 17.5 Clear Channel Assessment (CCA)

The Clear Channel Assessment is used to indicate if the current channel is free or busy. The current CCA state is viewable on any of the GDO pins.

`MCSM1.CCA_MODE` selects the mode to use when determining CCA.

When the `STX` or `SFSTXON` command strobe is given while **CC2500** is in the RX state, the TX state is only entered if the clear channel requirements are fulfilled. The chip will otherwise remain in RX. This feature is called TX if CCA.

Four CCA requirements can be programmed:

- Always (CCA disabled, always goes to TX)
- If RSSI is below threshold
- Unless currently receiving a packet
- Both the above (RSSI below threshold and not currently receiving a packet)

### 17.6 Link Quality Indicator (LQI)

The Link Quality Indicator is a metric of the current quality of the received signal. If `PKTCTRL1.APPEND_STATUS` is enabled, the value is automatically appended to the end of each received packet. The value can also be read from the `LQI` status register. The LQI is calculated over the 64 symbols following the sync word (first 8 packet bytes for 2-ary modulation). LQI is best used as a relative measurement of the link quality, since the value is dependent on the modulation format.

## 18 Forward Error Correction with Interleaving

### 18.1 Forward Error Correction (FEC)

**CC2500** has built in support for Forward Error Correction (FEC). To enable this option, set `MDMCFG1.FEC_EN` to 1. FEC is employed on the data field and CRC word in order to reduce the gross bit error rate when operating near the sensitivity limit. Redundancy is added to the transmitted data in such a way that the receiver can restore the original data in the presence of some bit errors.

The use of FEC allows correct reception at a lower SNR, thus extending communication range. Alternatively, for a given SNR, using FEC decreases the bit error rate (BER). As the packet error rate (PER) is related to BER by:

$$PER = 1 - (1 - BER)^{packet\_length}$$

a lower BER can be used to allow longer packets, or a higher percentage of packets of a given length, to be transmitted successfully. Finally, in realistic ISM radio environments, transient and time-varying phenomena will produce occasional errors even in otherwise good reception conditions. FEC will mask such errors and, combined with interleaving of the coded data, even correct relatively long periods of faulty reception (burst errors).

The FEC scheme adopted for **CC2500** is convolutional coding, in which  $n$  bits are generated based on  $k$  input bits and the  $m$  most recent input bits, forming a code stream able to withstand a certain number of bit errors between each coding state (the  $m$ -bit window).

The convolutional coder is a rate 1/2 code with a constraint length of  $m=4$ . The coder codes one input bit and produces two output bits; hence, the effective data rate is halved.

### 18.2 Interleaving

Data received through radio channels will often experience burst errors due to interference and time-varying signal strengths. In order to increase the robustness to errors spanning multiple bits, interleaving is used when FEC is enabled. After de-interleaving, a continuous span of errors in the received stream will become single errors spread apart.

**CC2500** employs matrix interleaving, which is illustrated in Figure 10. The on-chip interleaving and de-interleaving buffers are 4 x 4 matrices. In the transmitter, the data bits are written into the rows of the matrix, whereas the bit sequence to be transmitted is read from the columns of the matrix and fed to the rate 1/2 convolutional coder. Conversely, in the receiver, the received symbols are written into the columns of the matrix, whereas the data passed onto the convolutional decoder is read from the rows of the matrix.

When FEC and interleaving is used, the amount of data transmitted over the air must be a multiple of the size of the interleaver buffer (two bytes). In addition, at least one extra byte is required for trellis termination. The packet control hardware therefore automatically inserts one or two extra bytes at the end of the packet, so that the total length of the data to be interleaved is an even number. Note that these extra bytes are invisible to the user, as they are removed before the received packet enters the RX FIFO.

Due to the implementation of the FEC and interleaver, the data to be interleaved must be at least two bytes. One byte long fixed length packets without CRC is therefore not supported when FEC/interleaving is enabled.

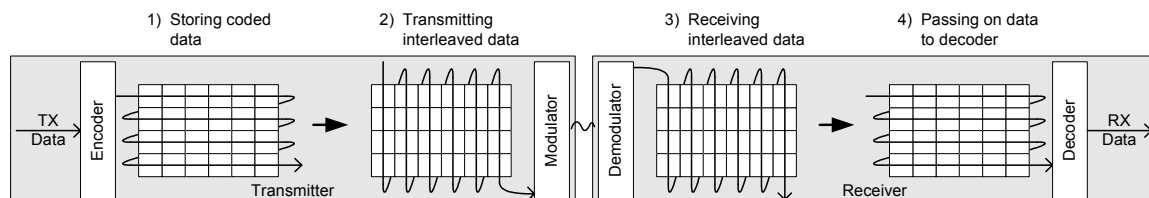
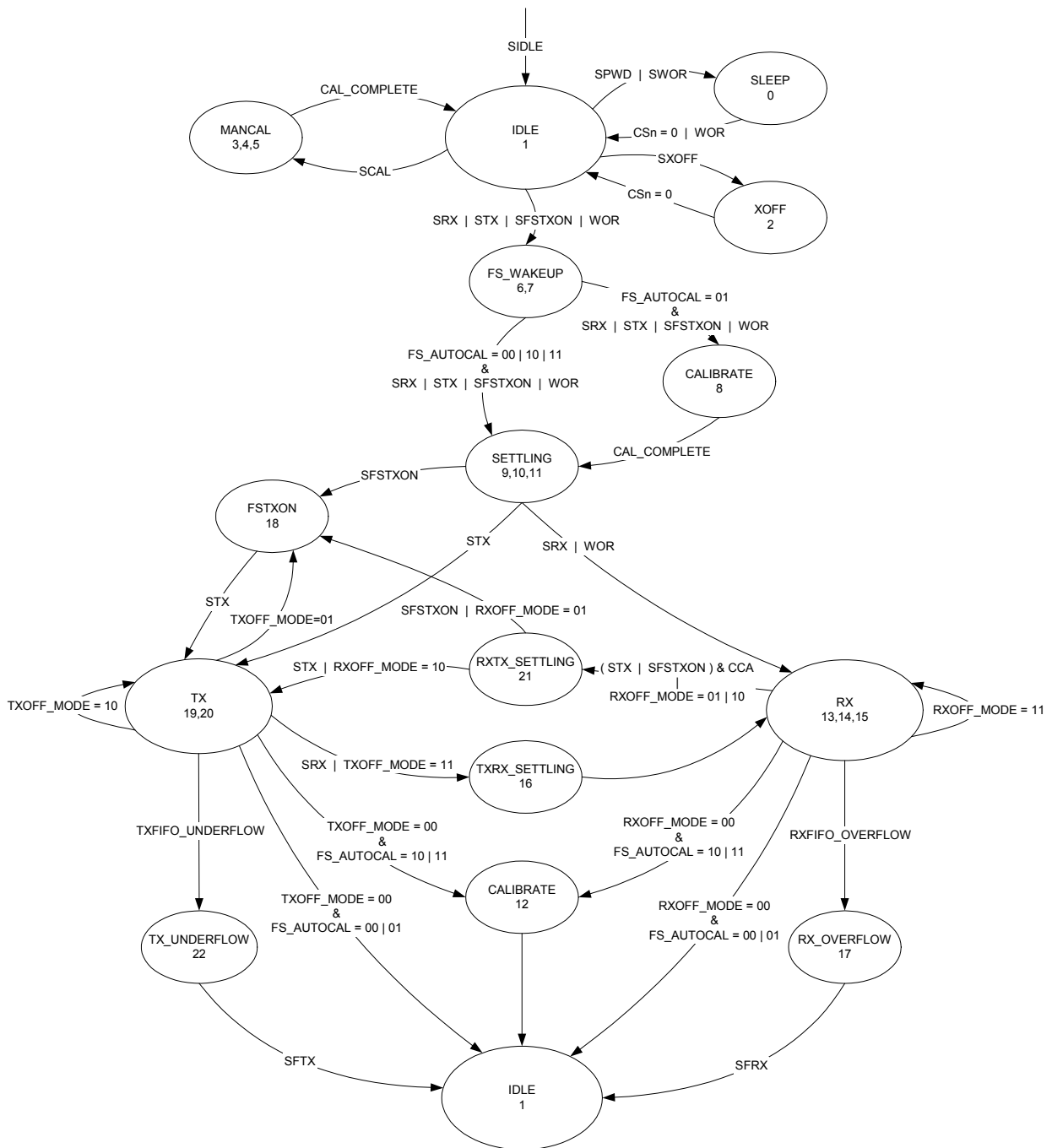


Figure 10: General principle of matrix interleaving

**19 Radio Control**



**Figure 11: Complete radio control state diagram**

**CC2500** has a built-in state machine that is used to switch between different operation states (modes). The change of state is done either by using command strobes or by internal events such as TX FIFO underflow.

A simplified state diagram, together with typical usage and current consumption, is

shown in Figure 4 on page 13. The complete radio control state diagram is shown in Figure 11. The numbers refer to the state number readable in the `MARCSSTATE` status register. This register is primarily for test purposes.

### 19.1 Power-On Start-Up Sequence

When the power supply is turned on, the system must be reset. One of the following two sequences must be followed: Automatic power-on reset (POR) or manual reset.

#### 19.1.1 Automatic POR

A power-on reset circuit is included in the **CC2500**. The minimum requirements stated in Section 4.9 must be followed for the power-on reset to function properly. The internal power-up sequence is completed when `CHIP_RDYn` goes low. `CHIP_RDYn` is observed on the `SO` pin after `CSn` is pulled low. See Section 10.1 for more details on `CHIP_RDYn`.

#### 19.1.2 Manual Reset

The other global reset possibility on **CC2500** is the `SRES` command strobe. By issuing this strobe, all internal registers and states are set to the default, idle state. The manual power-up sequence is as follows (see Figure 12):

- Set `SCLK=1` and `SI=0`, to avoid potential problems with pin control mode (see Section 11.3 on page 23).
- Strobe `CSn` low / high.
- Hold `CSn` high for at least 40  $\mu$ s.
- Pull `CSn` low and wait for `SO` to go low (`CHIP_RDYn`).
- Issue the `SRES` strobe on the `SI` line.
- When `SO` goes low again, reset is complete and the chip is in the IDLE state.

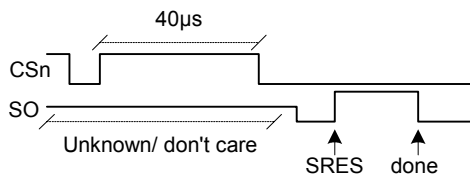


Figure 12: Power-on reset with `SRES`

### 19.2 Crystal Control

The crystal oscillator (XOSC) is either automatically controlled or always on, if `MCSM0.XOSC_FORCE_ON` is set.

In the automatic mode, the XOSC will be turned off if the `SXOFF` or `SPWD` command strobes are issued; the state machine then goes to `XOFF` or `SLEEP` respectively. This can only be done from the IDLE state. The

XOSC will be turned off when `CSn` is released (goes high). The XOSC will be automatically turned on again when `CSn` goes low. The state machine will then go to the IDLE state. The `SO` pin on the SPI interface must be zero before the SPI interface is ready to be used; as described in Section 19 on page 20.

If the XOSC is forced on, the crystal will always stay on even in the SLEEP state.

Crystal oscillator start-up time depends on crystal ESR and load capacitances. The electrical specification for the crystal oscillator can be found in Section 4.4 on page 10.

### 19.3 Voltage Regulator Control

The voltage regulator to the digital core is controlled by the radio controller. When the chip enters the SLEEP state, which is the state with the lowest current consumption, this regulator is disabled. This occurs after `CSn` is released when a `SPWD` command strobe has been sent on the SPI interface. The chip is now in the SLEEP state. Setting `CSn` low again will turn on the regulator and crystal oscillator and make the chip enter the IDLE state.

When wake on radio is enabled, the WOR module will control the voltage regulator as described in Section 19.5.

### 19.4 Active Modes

**CC2500** has two active modes: receive and transmit. These modes are activated directly by the MCU by using the `SRX` and `STX` command strobes, or automatically by Wake on Radio.

The frequency synthesizer must be calibrated regularly. **CC2500** has one manual calibration option (using the `SCAL` strobe), and three automatic calibration options, controlled by the `MCSM0.FS_AUTOCAL` setting:

- Calibrate when going from IDLE to either RX or TX (or FSTXON)
- Calibrate when going from either RX or TX to IDLE
- Calibrate every fourth time when going from either RX or TX to IDLE

The calibration takes a constant number of XOSC cycles (see Table 28 for timing details).

When RX is activated, the chip will remain in receive mode until a packet is successfully

received or the RX termination timer expires (see Section 19.7). Note: the probability that a false sync word is detected can be reduced by using PQT, CS, maximum sync word length and sync word qualifier mode as describe in Section 17. After a packet is successfully received the radio controller will then go to the state indicated by the `MCSM1.RXOFF_MODE` setting. The possible destinations are:

- IDLE
- FSTXON: Frequency synthesizer on and ready at the TX frequency. Activate TX with `STX`.
- TX: Start sending preambles
- RX: Start search for a new packet

Similarly, when TX is active the chip will remain in the TX state until the current packet has been successfully transmitted. Then the state will change as indicated by the `MCSM1.TXOFF_MODE` setting. The possible destinations are the same as for RX.

The MCU can manually change the state from RX to TX and vice versa by using the command strobes. If the radio controller is currently in transmit and the `SRX` strobe is used, the current transmission will be ended and the transition to RX will be done.

If the radio controller is in RX when the `STX` or `SFSTXON` command strobes are used, the “TX if clear channel” function will be used. If the channel is not clear, the chip will remain in RX. The `MCSM1.CCA_MODE` setting controls the conditions for clear channel assessment. See Section 17.5 on page 32 for details.

The `SIDLE` command strobe can always be used to force the radio controller to go to the IDLE state.

### 19.5 Wake On Radio (WOR)

The optional Wake on Radio (WOR) functionality enables **CC2500** to periodically wake up from deep sleep and listen for incoming packets without MCU interaction.

When WOR is enabled, the **CC2500** will go to the SLEEP state when `CSn` is released after the `SWOR` command strobe has been sent on the SPI interface. The RC oscillator must be enabled before the WOR strobe can be used, as it is the clock source for the WOR timer. The on-chip timer will get **CC2500** back into the

IDLE state when the timer expires. After a programmable time in RX, the chip goes back to SLEEP, unless a packet is received. See Section 19.7 for details on how the timeout works.

**CC2500** can be set up to signal the MCU that a packet has been received by using the GDO pins. If a packet is received, the `MCSM1.RXOFF_MODE` will determine the behaviour at the end of the received packet. When the MCU has read the packet, it can put the chip back into SLEEP with the `SWOR` strobe from the IDLE state. The FIFO will lose its contents in the SLEEP state.

The WOR timer has two events, Event 0 and Event 1. In the SLEEP state with WOR activated, reaching Event 0 will turn the digital regulator and start the crystal oscillator. Event 1 follows Event 0 after a programmed timeout.

The time between two consecutive Event 0 is programmed with a mantissa value given by `WOREVT1.EVENT0` and `WOREVT0.EVENT0`, and an exponent value set by `WORCTRL.WOR_RES`. The equation is:

$$t_{Event0} = \frac{750}{f_{XOSC}} \cdot EVENT0 \cdot 2^{5 \cdot WOR\_RES}$$

The Event 1 timeout is programmed with `WORCTRL.EVENT1`. Figure 13 shows the timing relationship between Event 0 timeout and Event 1 timeout.

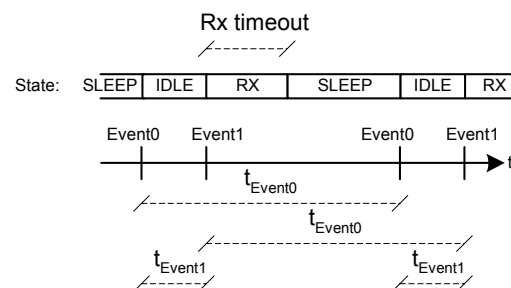


Figure 13: Event 0 and Event 1 relationship

#### 19.5.1 RC Oscillator and Timing

The frequency of the low-power RC oscillator used for the WOR functionality varies with temperature and supply voltage. In order to

keep the frequency as accurate as possible, the RC oscillator will be calibrated whenever possible, which is when the XOSC is running and the chip is not in the SLEEP state. When the power and XOSC is enabled, the clock used by the WOR timer is a divided XOSC clock. When the chip goes to the sleep state, the RC oscillator will use the last valid calibration result. The frequency of the RC oscillator is locked to the main crystal frequency divided by 750.

### 19.6 Timing

The radio controller controls most timing in **CC2500**, such as synthesizer calibration, PLL lock and RT/TX turnaround times. Timing from IDLE to RX and IDLE to TX is constant, dependent on the auto calibration setting. RX/TX and TX/RX turnaround times are constant. The calibration time is constant 18739 clock periods. Table 28 shows timing in crystal clock cycles for key state transitions.

Power on time and XOSC start-up times are variable, but within the limits stated in Table 7. Note that in a frequency hopping spread spectrum or a multi-channel protocol the calibration time can be reduced from 721  $\mu$ s to approximately 150  $\mu$ s. This is explained in Section 30.2.

Description	XOSC periods	26 MHz crystal
Idle to RX, no calibration	2298	88.4 $\mu$ s
Idle to RX, with calibration	~21037	809 $\mu$ s
Idle to TX/FSTXON, no calibration	2298	88.4 $\mu$ s
Idle to TX/FSTXON, with calibration	~21037	809 $\mu$ s
TX to RX switch	560	21.5 $\mu$ s
RX to TX switch	250	9.6 $\mu$ s
RX or TX to IDLE, no calibration	2	0.1 $\mu$ s
RX or TX to IDLE, with calibration	~18739	721 $\mu$ s
Manual calibration	~18739	721 $\mu$ s

**Table 28: State transition timing**

### 19.7 RX Termination Timer

**CC2500** has optional functions for automatic termination of RX after a programmable time.

The main use for this functionality is wake-on-radio (WOR), but it may be useful for other applications. The termination timer starts when in RX state. The timeout is programmable with the `MCSM2.RX_TIME` setting. When the timer expires, the radio controller will check the condition for staying in RX; if the condition is not met, RX will terminate. After the timeout, the condition will be checked continuously.

The programmable conditions are:

- `MCSM2.RX_TIME_QUAL=0`: Continue receive if sync word has been found
- `MCSM2.RX_TIME_QUAL=1`: Continue receive if sync word has been found or preamble quality is above threshold (PQT)

If the system can expect the transmission to have started when enabling the receiver, the `MCSM2.RX_TIME_RSSI` function can be used. The radio controller will then terminate RX if the first valid carrier sense sample indicates no carrier (RSSI below threshold). See Section 17.4 on page 31 for details on Carrier Sense.

For OOK modulation, lack of carrier sense is only considered valid after eight symbol periods. Thus, the `MCSM2.RX_TIME_RSSI` function can be used in OOK mode when the distance between "1" symbols is 8 or less.

If RX terminates due to no carrier sense when the `MCSM2.RX_TIME_RSSI` function is used, or if no sync word was found when using the `MCSM2.RX_TIME` timeout function, the chip will always go back to IDLE if WOR is disabled and back to SLEEP if WOR is enabled. Otherwise, the `MCSM1.RXOFF_MODE` setting determines the state to go to when RX ends.

Note that in wake-on-radio (WOR) mode, the WOR state is cleared in the latter case. This means that the chip will not automatically go back to SLEEP again, even if e.g. the address field in the packet did not match. It is therefore recommended to always wake up the microcontroller on sync word detection when using WOR mode. This can be done by selecting output signal 6 (see Table 33 on page 46) on one of the programmable GDO output pins, and programming the microcontroller to wake up on an edge-triggered interrupt from this GDO pin.

## 20 Data FIFO

The **CC2500** contains two 64 byte FIFOs, one for received data and one for data to be transmitted. The SPI interface is used to read from the RX FIFO and write to the TX FIFO. Section 10.4 contains details on the SPI FIFO access. The FIFO controller will detect overflow in the RX FIFO and underflow in the TX FIFO.

When writing to the TX FIFO it is the responsibility of the MCU to avoid TX FIFO overflow. A TX FIFO overflow will result in an error in the TX FIFO content.

Likewise, when reading the RX FIFO the MCU must avoid reading the RX FIFO past its empty value, since an RX FIFO underflow will result in an error in the data read out of the RX FIFO.

The chip status byte that is available on the **SO** pin while transferring the SPI address contains the fill grade of the RX FIFO if the address is a read operation and the fill grade of the TX FIFO if the address is a write operation. Section 10.1 on page 20 contains more details on this.

The number of bytes in the RX FIFO and TX FIFO can also be read from the status registers `RXBYTES.NUM_RXBYTES` and `TXBYTES.NUM_TXBYTES` respectively. If receiving data while reading the last byte in the RX FIFO, the RX FIFO pointer is not updated, resulting in a duplication of the last byte read.

To avoid this problem one should never empty the RX FIFO before the last byte of the packet is received. The following software fix can be used:

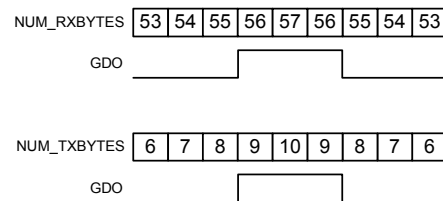
1. Read `RXBYTES.NUM_RXBYTES`
2. If `RXBYTES.NUM_RXBYTES < packet length`, read `RXBYTES.NUM_RXBYTES-1` bytes from the FIFO
3. Repeat until `RXBYTES.NUM_RXBYTES = number of remaining bytes of the packet`
4. Read the remaining bytes from the FIFO

The 4-bit `FIFOTHRR.FIFO_THR` setting is used to program threshold points in the FIFOs. Table 29 lists the 16 `FIFO_THR` settings and

the corresponding thresholds for the RX and TX FIFOs. The threshold value is coded in opposite directions for the RX FIFO and TX FIFO. This gives equal margin to the overflow and underflow conditions when the threshold is reached.

A flag will assert when the number of bytes in the FIFO is equal to or higher than the programmed threshold. The flag is used to generate the FIFO status signals that can be viewed on the GDO pins (see Section 28 on page 45).

Figure 15 shows the number of bytes in both the RX FIFO and TX FIFO when the threshold flag toggles, in the case of `FIFO_THR=13`. Figure 14 shows the flag as the respective FIFO is filled above the threshold, and then drained below.



**Figure 14: `FIFO_THR=13` vs. number of bytes in FIFO (`GDOx_CFG=0x00` in Rx and `GDOx_CFG=0x02` in Tx)**

<code>FIFO_THR</code>	Bytes in TX FIFO	Bytes in RX FIFO
0 (0000)	61	4
1 (0001)	57	8
2 (0010)	53	12
3 (0011)	49	16
4 (0100)	45	20
5 (0101)	41	24
6 (0110)	37	28
7 (0111)	33	32
8 (1000)	29	36
9 (1001)	25	40
10 (1010)	21	44
11 (1011)	17	48
12 (1100)	13	52
13 (1101)	9	56
14 (1110)	5	60
15 (1111)	1	64

**Table 29: `FIFO_THR` settings and the corresponding FIFO thresholds**

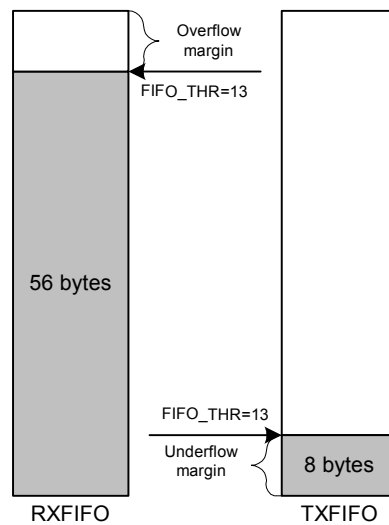


Figure 15: Example of FIFOs at threshold

## 21 Frequency Programming

The frequency programming in **CC2500** is designed to minimize the programming needed in a channel-oriented system.

To set up a system with channel numbers, the desired channel spacing is programmed with the `MDMCFG0.CHANSPC_M` and `MDMCFG1.CHANSPC_E` registers. The channel spacing registers are mantissa and exponent respectively.

The base or start frequency is set by the 24 bit frequency word located in the `FREQ2`, `FREQ1` and `FREQ0` registers. This word will typically be set to the centre of the lowest channel frequency that is to be used.

The desired channel number is programmed with the 8-bit channel number register, `CHANNR.CHAN`, which is multiplied by the channel offset. The resultant carrier frequency is given by:

$$f_{carrier} = \frac{f_{XOSC}}{2^{16}} \cdot (FREQ + CHAN \cdot ((256 + CHANSPC\_M) \cdot 2^{CHANSPC\_E-2}))$$

With a 26 MHz crystal the maximum channel spacing is 405 kHz. To get e.g. 1 MHz channel spacing one solution is to use 333 kHz channel spacing and select each third channel in `CHANNR.CHAN`.

The preferred IF frequency is programmed with the `FSCTRL1.FREQ_IF` register. The IF frequency is given by:

$$f_{IF} = \frac{f_{XOSC}}{2^{10}} \cdot FREQ\_IF$$

Note that the SmartRF® Studio software automatically calculates the optimum `FSCTRL1.FREQ_IF` register setting based on channel spacing and channel filter bandwidth.

If any frequency programming register is altered when the frequency synthesizer is running, the synthesizer may give an undesired response. Hence, the frequency programming should only be updated when the radio is in the IDLE state.



## 22 VCO

The VCO is completely integrated on-chip.

### 22.1 VCO and PLL Self-Calibration

The VCO characteristics will vary with temperature and supply voltage changes, as well as the desired operating frequency. In order to ensure reliable operation, **CC2500** includes frequency synthesizer self-calibration circuitry. This calibration should be done regularly, and must be performed after turning on power and before using a new frequency (or channel). The number of XOSC cycles for completing the PLL calibration is given in Table 28 on page 37.

The calibration can be initiated automatically or manually. The synthesizer can be automatically calibrated each time the synthesizer is turned on, or each time the synthesizer is turned off. This is configured with the `MCSM0.FS_AUTOCAL` register setting. In manual mode, the calibration is initiated when the `SCAL` command strobe is activated in the IDLE mode.

Note that the calibration values are maintained in sleep mode, so the calibration is still valid after waking up from sleep mode (unless supply voltage or temperature has changed significantly).

## 23 Voltage Regulators

**CC2500** contains several on-chip linear voltage regulators, which generate the supply voltage needed by low-voltage modules. These voltage regulators are invisible to the user, and can be viewed as integral parts of the various modules. The user must however make sure that the absolute maximum ratings and required pin voltages in Table 1 and Table 13 are not exceeded. The voltage regulator for the digital core requires one external decoupling capacitor.

Setting the `CSn` pin low turns on the voltage regulator to the digital core and starts the crystal oscillator. The `SO` pin on the SPI interface must go low before using the serial interface (setup time is given in Table 16).

If the chip is programmed to enter power-down mode, (`SPWD` strobe issued), the power will be turned off after `CSn` goes high. The power and crystal oscillator will be turned on again when `CSn` goes low.

The voltage regulator output should only be used for driving the **CC2500**.

## 24 Output Power Programming

The RF output power level from the device has two levels of programmability, as illustrated in Figure 16. Firstly, the special `PATABLE` register can hold up to eight user selected output power settings. Secondly, the 3-bit `FREND0.PA_POWER` value selects the `PATABLE` entry to use. This two-level functionality provides flexible PA power ramp up and ramp down at the start and end of transmission. All the PA power settings in the `PATABLE` from index 0 up to the `FREND0.PA_POWER` value are used.

The power ramping at the start and at the end of a packet can be turned off by setting `FREND0.PA_POWER` to zero and then program the desired output power to index zero in the `PATABLE`.

Table 31 contains recommended `PATABLE` settings for various output levels and frequency bands. See Section 10.5 on page 22 for `PATABLE` programming details.

`PATABLE` must be programmed in burst mode if you want to write to other entries than `PATABLE[0]`.

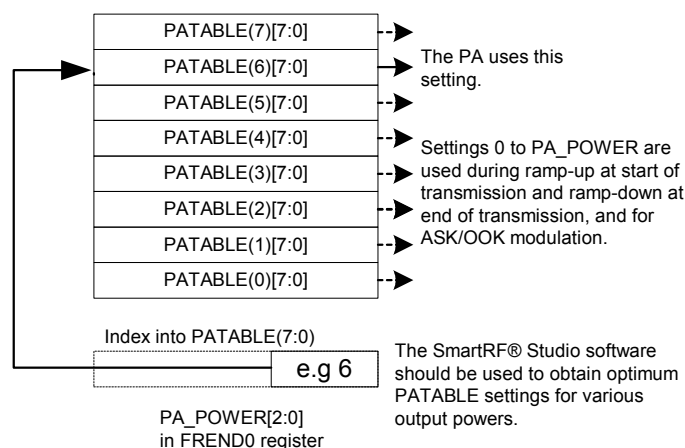


Figure 16: PA\_POWER and PATABLE

Default power setting	Output power, typical [dBm]	Current consumption, typical [mA]
0xC6	-11.8	11.1

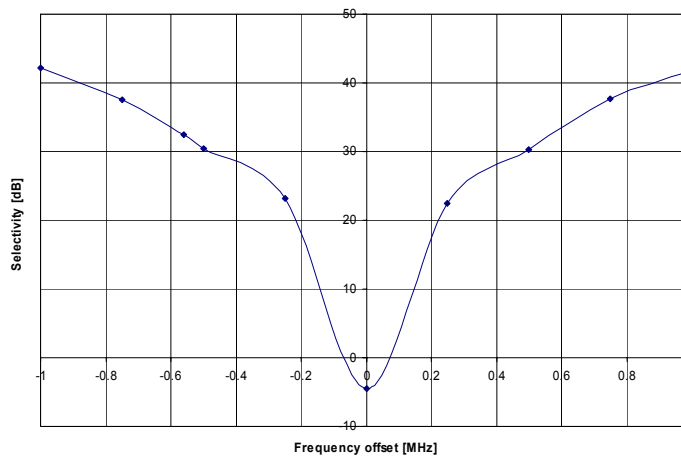
Table 30: Output power and current consumption for default PATABLE setting

Output power, typical, +25°C, 3.0 V [dBm]	PATABLE value	Current consumption, typical [mA]
(-55 or less)	0x00	8.4
-30	0x50	9.9
-28	0x44	9.7
-26	0xC0	10.2
-24	0x84	10.1
-22	0x81	10.0
-20	0x46	10.1
-18	0x93	11.7
-16	0x55	10.8
-14	0x8D	12.2
-12	0xC5	11.1
-10	0x97	12.2
-8	0x6E	14.1
-6	0x7F	15.1
-4	0xA9	16.2
-2	0xBB	17.7
0	0xFE	21.2
1.5	0xFF	21.5

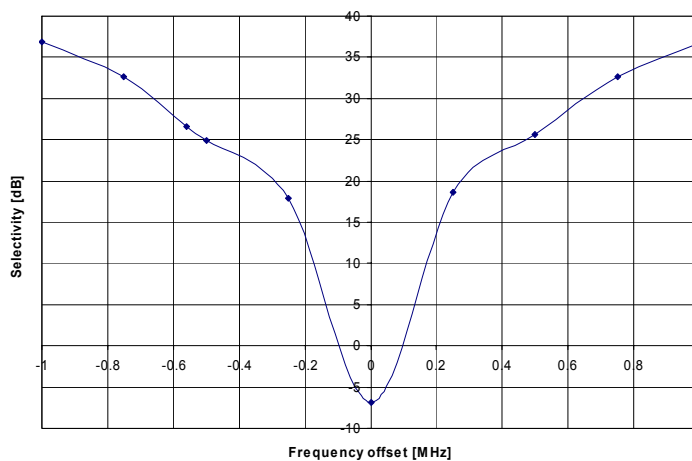
Table 31: Optimum PATABLE settings for various output power levels (subject to changes)

## 25 Selectivity Graphs

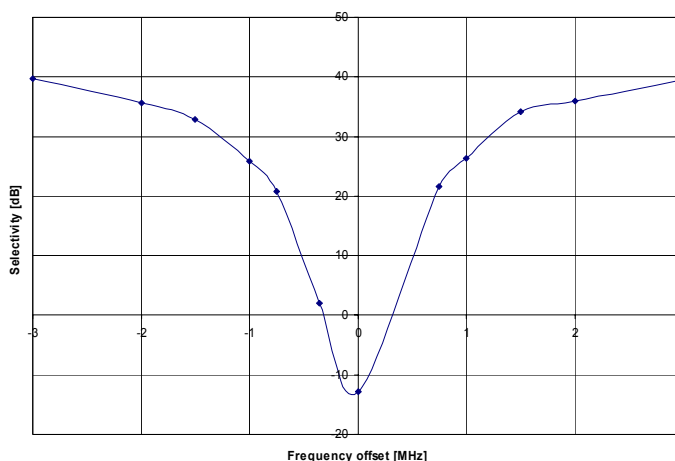
Figure 17 to Figure 21 show the typical selectivity performance (adjacent and alternate rejection).



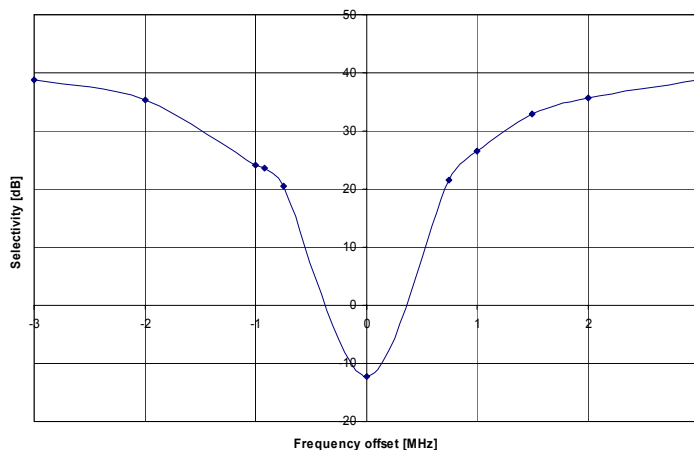
**Figure 17: Typical selectivity at 2.4 kbps. IF frequency is 273.9 kHz.**  
**MDMCFG2.DEM\_DCFILT\_OFF = 1**



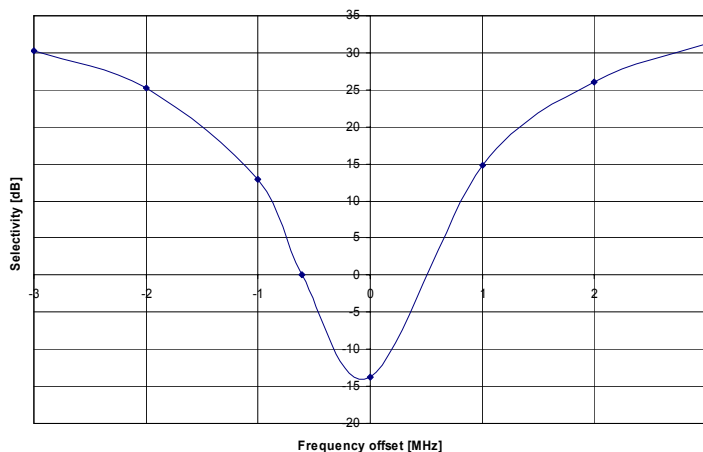
**Figure 18: Typical selectivity at 10 kbps. IF frequency is 273.9 kHz.**  
**MDMCFG2.DEM\_DCFILT\_OFF = 1**



**Figure 19: Typical selectivity at 250 kbps. IF frequency is 177.7 kHz.  
MDMCFG2.DEM\_DCFILT\_OFF = 0**



**Figure 20: Typical selectivity at 250 kbps. IF frequency is 457 kHz.  
MDMCFG2.DEM\_DCFILT\_OFF = 1**



**Figure 21: Typical selectivity at 500 kbps. IF frequency is 307.4 kHz.  
MDMCFG2.DEM\_DCFILT\_OFF = 0**

## 26 Crystal Oscillator

A crystal in the frequency range 26-27 MHz must be connected between the XOSC\_Q1 and XOSC\_Q2 pins. The oscillator is designed for parallel mode operation of the crystal. In addition, loading capacitors (C81 and C101) for the crystal are required. The loading capacitor values depend on the total load capacitance,  $C_L$ , specified for the crystal. The total load capacitance seen between the crystal terminals should equal  $C_L$  for the crystal to oscillate at the specified frequency.

$$C_L = \frac{1}{\frac{1}{C_{81}} + \frac{1}{C_{101}}} + C_{parasitic}$$

The parasitic capacitance is constituted by pin input capacitance and PCB stray capacitance. Total parasitic capacitance is typically 2.5 pF.

The crystal oscillator circuit is shown in Figure 22. Typical component values for different values of  $C_L$  are given in Table 32.

The crystal oscillator is amplitude regulated. This means that a high current is used to start up the oscillations. When the amplitude builds up, the current is reduced to what is necessary to maintain approximately 0.4Vpp signal swing. This ensures a fast start-up, and keeps the drive level to a minimum. The ESR of the crystal should be within the specification in order to ensure a reliable start-up (see Section 4.4 on page 10).

The initial tolerance, temperature drift, aging and load pulling should be carefully specified in order to meet the required frequency accuracy in a certain application. By specifying the *total* expected frequency accuracy in SmartRF® Studio together with data rate and frequency deviation, the software calculates the total bandwidth and compares this to the chosen receiver channel filter bandwidth. The software reports any contradictions, and a more accurate crystal is recommended if required.

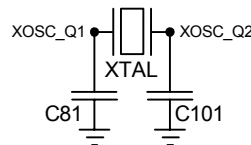


Figure 22: Crystal oscillator circuit

Component	$C_L = 10 \text{ pF}$	$C_L = 13 \text{ pF}$	$C_L = 16 \text{ pF}$
C81	15 pF	22 pF	27 pF
C101	15 pF	22 pF	27 pF

Table 32: Crystal oscillator component values

### 26.1 Reference Signal

The chip can alternatively be operated with a reference signal from 26 to 27 MHz instead of a crystal. This input clock can either be a full-swing digital signal (0 V to VDD) or a sine wave of maximum 1 V peak-peak amplitude. The reference signal must be connected to the

XOSC\_Q1 input. The sine wave must be connected to XOSC\_Q1 using a serial capacitor. The XOSC\_Q2 line must be left unconnected. C81 and C101 can be omitted when using a reference signal.

## 27 External RF Match

The balanced RF input and output of **CC2500** share two common pins and are designed for a simple, low-cost matching and balun network on the printed circuit board. The receive- and transmit switching at the **CC2500** front-end is controlled by a dedicated on-chip function, eliminating the need for an external RX/TX-switch.

A few passive external components combined with the internal RX/TX switch/termination circuitry ensures match in both RX and TX mode.

Although **CC2500** has a balanced RF input/output, the chip can be connected to a single-ended antenna with few external low cost capacitors and inductors.

The passive matching/filtering network connected to **CC2500** should have the following differential impedance as seen from the RF-port (RF\_P and RF\_N) towards the antenna:

$$Z_{out} = 80 + j74 \Omega$$

## 28 General Purpose / Test Output Control Pins

The three digital output pins GDO0, GDO1 and GDO2 are general control pins configured with IOCFG0.GDO0\_CFG, IOCFG1.GDO1\_CFG and IOCFG2.GDO3\_CFG respectively. Table 33 shows the different signals that can be monitored on the GDO pins. These signals can be used as an interrupt to the MCU. GDO1 is the same pin as the SO pin on the SPI interface, thus the output programmed on this pin will only be valid when CS<sub>n</sub> is high. The default value for GDO1 is 3-stated, which is useful when the SPI interface is shared with other devices.

The default value for GDO0 is a 135-141 kHz clock output (XOSC frequency divided by 192). Since the XOSC is turned on at power-on-reset, this can be used to clock the MCU in systems with only one crystal. When the MCU is up and running, it can change the clock frequency by writing to IOCFG0.GDO0\_CFG.

An on-chip analog temperature sensor is enabled by writing the value 128 (0x80h) to the IOCFG0.GDO0\_CFG register. The voltage on the GDO0 pin is then proportional to temperature. See Section 4.7 on page 12 for temperature sensor specifications.

GDO0_CFG[5:0] GDO1_CFG[5:0] GDO2_CFG[5:0]	Description
0 (0x00)	Associated to the RX FIFO: Asserts when RX FIFO is filled above RXFIFO_THR. De-asserts when RX FIFO is drained below RXFIFO_THR.
1 (0x01)	Associated to the RX FIFO: Asserts when RX FIFO is filled above RXFIFO_THR or the end of packet is reached. De-asserts when RX FIFO is empty.
2 (0x02)	Associated to the TX FIFO: Asserts when the TX FIFO is filled above TXFIFO_THR. De-asserts when the TX FIFO is below TXFIFO_THR.
3 (0x03)	Associated to the TX FIFO: Asserts when TX FIFO is full. De-asserts when the TX FIFO is drained below TXFIFO_THR.
4 (0x04)	Asserts when the RX FIFO has overflowed. De-asserts when the FIFO has been flushed.
5 (0x05)	Asserts when the TX FIFO has underflowed. De-asserts when the FIFO is flushed.
6 (0x06)	Asserts when sync word has been sent / received, and de-asserts at the end of the packet. In RX, the pin will de-assert when the optional address check fails or the RX FIFO overflows. In TX the pin will de-assert if the TX FIFO underflows.
7 (0x07)	Reserved
8 (0x08)	Preamble Quality Reached. Asserts when the PQI is above the programmed PQT value.
9 (0x09)	Clear channel assessment. High when RSSI level is below threshold (dependent on the current CCA_MODE setting)
10 (0x0A)	Lock detector output
11 (0x0B)	Serial Clock. Synchronous to the data in synchronous serial mode. Data is set up on the falling edge and is read on the rising edge of SERIAL_CLK when GDOx_INV=0.
12 (0x0C)	Serial Synchronous Data Output (DO). Used for synchronous serial mode. The MCU must read DO on the rising edge of SERIAL_CLK when GDOx_INV=0. Data is set up on the falling edge by <b>CC2500</b> .
13 (0x0D)	Serial transparent Data Output. Used for asynchronous serial mode.
14 (0x0E)	Carrier sense. High if RSSI level is above threshold.
15 (0x0F)	Reserved
16 (0x10)	Reserved – used for test.
17 (0x11)	Reserved – used for test.
18 (0x12)	Reserved – used for test.
19 (0x13)	Reserved – used for test.
20 (0x14)	Reserved – used for test.
21 (0x15)	Reserved – used for test.
22 (0x16)	RX_HARD_DATA[1]. Can be used together with RX_SYMBOL_TICK for alternative serial RX output.
23 (0x17)	RX_HARD_DATA[0]. Can be used together with RX_SYMBOL_TICK for alternative serial RX output.
24 (0x18)	Reserved – used for test.
25 (0x19)	Reserved – used for test.
26 (0x1A)	Reserved – used for test.
27 (0x1B)	PA_PD. PA is enabled when 0, in power-down when 1. Can be used to control external PA or RX/TX switch.
28 (0x1C)	LNA_PD. LNA is enabled when 0, in power-down when 1. Can be used to control external LNA or RX/TX switch.
29 (0x1D)	RX_SYMBOL_TICK. Can be used together with RX_HARD_DATA for alternative serial RX output.
30 (0x1E)	Reserved – used for test.
31 (0x1F)	Reserved – used for test.
32 (0x20)	Reserved – used for test.
33 (0x21)	Reserved – used for test.
34 (0x22)	Reserved – used for test.
35 (0x23)	Reserved – used for test.
36 (0x24)	Reserved – used for test.
37 (0x25)	Reserved – used for test.
38 (0x26)	Reserved – used for test.
39 (0x27)	Reserved – used for test.
40 (0x28)	Reserved – used for test.
41 (0x29)	CHIP_RDY
42 (0x2A)	Reserved – used for test.
43 (0x2B)	XOSC_STABLE
44 (0x2C)	Reserved – used for test.
45 (0x2D)	GDO0_Z_EN_N. When this output is 0, GDO0 is configured as input (for serial TX data).
46 (0x2E)	High impedance (3-state)
47 (0x2F)	HW to 0 (HW1 achieved with _INV signal)
48 (0x30)	CLK_XOSC/1
49 (0x31)	CLK_XOSC/1.5
50 (0x32)	CLK_XOSC/2
51 (0x33)	CLK_XOSC/3
52 (0x34)	CLK_XOSC/4
53 (0x35)	CLK_XOSC/6
54 (0x36)	CLK_XOSC/8
55 (0x37)	CLK_XOSC/12
56 (0x38)	CLK_XOSC/16
57 (0x39)	CLK_XOSC/24
58 (0x3A)	CLK_XOSC/32
59 (0x3B)	CLK_XOSC/48
60 (0x3C)	CLK_XOSC/64
61 (0x3D)	CLK_XOSC/96
62 (0x3E)	CLK_XOSC/128
63 (0x3F)	CLK_XOSC/192

**Table 33: GDO signal selection**

## 29 Asynchronous and Synchronous Serial Operation

Several features and modes of operation have been included in the **CC2500** to provide backward compatibility with previous Chipcon products and other existing RF communication systems. For new systems, it is recommended to use the built-in packet handling features, as they can give more robust communication, significantly offload the microcontroller and simplify software development.

### 29.1 Asynchronous operation

For backward compatibility with systems already using the asynchronous data transfer from other Chipcon products, asynchronous transfer is also included in **CC2500**. When asynchronous transfer is enabled, several of the support mechanisms for the MCU that are included in **CC2500** will be disabled, such as packet handling hardware, buffering in the FIFO and so on. The asynchronous transfer mode does not allow the use of the data whitener, interleaver and FEC.

Only 2-FSK, GFSK and OOK are supported for asynchronous transfer.

Setting `PKTCTRL0.PKT_FORMAT` to 3 enables asynchronous transparent (serial) mode.

In TX, the `GDO0` pin is used for data input (TX data). Data output can be `GDO0`, `GDO1` or `GDO2`.

The MCU must control start and stop of transmit and receive with the `STX`, `SRX` and `SIDLE` strobes.

The **CC2500** modulator samples the level of the asynchronous input 8 times faster than the programmed data rate. The timing requirement for the asynchronous stream is that the error in the bit period must be less than one eighth of the programmed data rate.

### 29.2 Synchronous serial operation

Setting `PKTCTRL0.PKT_FORMAT` to 1 enables synchronous serial operation mode. In the synchronous serial operation mode, data is transferred on a two wire serial interface. The **CC2500** provides a clock that is used to set up new data on the data input line or sample data on the data output line. Data input (TX data) is the `GDO0` pin. This pin will automatically be configured as an input when TX is active. The data output pin can be any of the `GDO` pins; this is set by the `IOCFG0.GDO0_CFG`, `IOCFG1.GDO1_CFG` and `IOCFG2.GDO2_CFG` fields.

Preamble and sync word insertion/detection may or may not be active, dependent on the sync mode set by the `MDMCFG2.SYNC_MODE`. If preamble and sync word is disabled, all other packet handler features and FEC should also be disabled. The MCU must then handle preamble and sync word insertion and detection in software. If preamble and sync word insertion/detection is left on, all packet handling features and FEC can be used. The **CC2500** will insert and detect the preamble and sync word and the MCU will only provide/get the data payload. This is equivalent to the recommended FIFO operation mode.

## 30 System considerations and Guidelines

### 30.1 SRD Regulations

International regulations and national laws regulate the use of radio receivers and transmitters. Short Range Devices (SRDs) for license free operation are allowed to operate in the 2.45 GHz bands worldwide. The most important regulations are EN 300 440 and EN 300 328 (Europe), FCC CFR47 part 15.247 and 15.249 (USA), and ARIB STD-T66 (Japan). A summary of the most important aspects of these regulations can be found in Application Note *AN032 SRD regulations for license-free transceiver operation in the 2.4*

*GHz band*, available from the Chipcon website.

Please note that compliance with regulations is dependent on complete system performance. It is the customer's responsibility to ensure that the system complies with regulations.

### 30.2 Frequency Hopping and Multi-Channel Systems

The 2.400 – 2.4835 GHz band is shared by many systems both in industrial, office and



home environments. It is therefore recommended to use frequency hopping spread spectrum (FHSS) or a multi-channel protocol because the frequency diversity makes the system more robust with respect to interference from other systems operating in the same frequency band. FHSS also combats multipath fading.

**CC2500** is highly suited for FHSS or multi-channel systems due to its agile frequency synthesizer and effective communication interface. Using the packet handling support and data buffering is also beneficial in such systems as these features will significantly offload the host controller.

Charge pump current, VCO current and VCO capacitance array calibration data is required for each frequency when implementing frequency hopping for **CC2500**. There are 3 ways of obtaining the calibration data from the chip:

1) Frequency hopping with calibration for each hop. The PLL calibration time is approximately 720  $\mu$ s.

2) Fast frequency hopping without calibration for each hop can be done by calibrating each frequency at startup and saving the resulting `FSCAL3`, `FSCAL2` and `FSCAL1` register values in MCU memory. Between each frequency hop, the calibration process can then be replaced by writing the `FSCAL3`, `FSCAL2` and `FSCAL1` register values corresponding to the next RF frequency. The PLL turn on time is approximately 90  $\mu$ s.

3) Run calibration on a single frequency at startup. Next write 0hex to `FSCAL3[5:4]` to disable the charge pump calibration. After writing to `FSCAL3[5:4]` strobe `SRX` (or `STX`) with `MCSM0.FS_AUTOCAL = 1` for each new frequency hop. That is, VCO current and VCO capacitance calibration is done but not charge pump current calibration. When charge pump current calibration is disabled the calibration time is reduced from approximately 720  $\mu$ s to approximately 150  $\mu$ s.

There is a trade off between blanking time and memory space needed for storing calibration data in non-volatile memory. Solution 2) above gives the shortest blanking interval, but requires more memory space to store calibration values. Solution 3) gives approximately 570  $\mu$ s smaller blanking interval than solution 1).

### 30.3 Data Burst Transmissions

The high maximum data rate of **CC2500** opens up for burst transmissions. A low average data rate link (say 10 kbps), can be realized using a higher over-the-air data rate. Buffering the data and transmitting in bursts at high data rate (say 500 kbps) will reduce the time in active mode, and hence also reduce the average current consumption significantly. Reducing the time in active mode will reduce the likelihood of collisions with other systems, e.g. WLAN.

### 30.4 Continuous Transmissions

In data streaming applications the **CC2500** opens up for continuous transmissions at 500 kbps effective data rate. As the modulation is done with an I/Q up-converter with LO I/Q-signals coming from a closed loop PLL, there is no limitation in the length of a transmission. (Open loop modulation used in some transceivers often prevents this kind of continuous data streaming and reduces the effective data rate.)

### 30.5 Crystal Drift Compensation

The **CC2500** has a very fine frequency resolution (see Table 9). This feature can be used to compensate for frequency offset and drift.

The frequency offset between an 'external' transmitter and the receiver is measured in the **CC2500** and can be read back from the `FREQEST` status register as described in Section 14.1. The measured frequency offset can be used to calibrate the frequency using the 'external' transmitter as the reference. That is, the received signal of the device will match the receiver's channel filter better. In the same way the centre frequency of the transmitted signal will match the 'external' transmitter's signal.

### 30.6 Spectrum Efficient Modulation

**CC2500** also has the possibility to use Gaussian shaped FSK (GFSK). This spectrum-shaping feature improves adjacent channel power (ACP) and occupied bandwidth. In 'true' FSK systems with abrupt frequency shifting, the spectrum is inherently broad. By making the frequency shift 'softer', the spectrum can be made significantly narrower. Thus, higher data rates can be

transmitted in the same bandwidth using GFSK.

### 30.7 Low Cost Systems

As the **CC2500** provides 500 kbps multi-channel performance without any external filters, a very low cost system can be made.

A differential antenna will eliminate the need for a balun, and the DC biasing can be achieved in the antenna topology, see Figure 3.

A HC-49 type SMD crystal is used in the CC2500EM reference design. Note that the crystal package strongly influences the price. In a size constrained PCB design a smaller, but more expensive, crystal may be used.

### 30.8 Battery Operated Systems

In low power applications, the SLEEP state with the crystal oscillator core switched off should be used when the **CC2500** is not active. It is possible to leave the crystal oscillator core running in the SLEEP state if start-up time is critical.

The WOR functionality should be used in low power applications.

### 30.9 Increasing Output Power

In some applications it may be necessary to extend the link range. Adding an external power amplifier is the most effective way of doing this.

The power amplifier should be inserted between the antenna and the balun, and two T/R switches are needed to disconnect the PA in RX mode. See Figure 23.

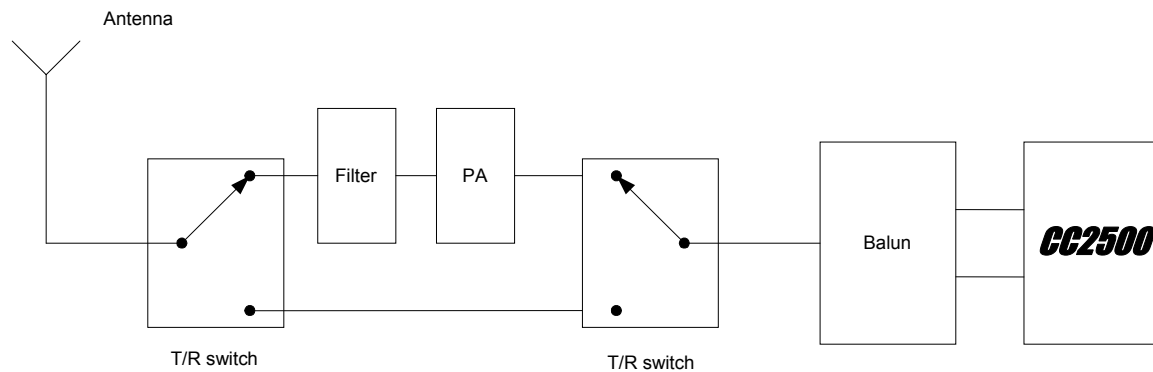


Figure 23. Block diagram of **CC2500** usage with external power amplifier

## 31 Configuration Registers

The configuration of **CC2500** is done by programming 8-bit registers. The configuration data based on selected system parameters are most easily found by using the SmartRF® Studio software. Complete descriptions of the registers are given in the following tables. After chip reset, all the registers have default values as shown in the tables.

There are 14 Command Strobe Registers, listed in Table 34. Accessing these registers will initiate the change of an internal state or mode. There are 47 normal 8-bit Configuration Registers, listed in Table 35. Many of these

registers are for test purposes only, and need not be written for normal operation of **CC2500**.

There are also 12 Status registers, which are listed in Table 36. These registers, which are read-only, contain information about the status of **CC2500**.

The two FIFOs are accessed through one 8-bit register. Write operations write to the TX FIFO, while read operations read from the RX FIFO.

During the address transfer and while writing to a register or the TX FIFO, a status byte is

returned. This status byte is described in Table 17 on page 21.

Table 37 summarizes the SPI address space. The address to use is given by adding the base address to the left and the burst and

read/write bits on the top. Note that the burst bit has different meaning for base addresses above and below 0x2F.

Address	Strobe Name	Description
0x30	SRES	Reset chip.
0x31	SFSTXON	Enable and calibrate frequency synthesizer (if <code>MCSM0.FS_AUTOCAL=1</code> ). If in RX (with CCA): Go to a wait state where only the synthesizer is running (for quick RX / TX turnaround).
0x32	SXOFF	Turn off crystal oscillator.
0x33	SCAL	Calibrate frequency synthesizer and turn it off (enables quick start). SCAL can be strobed in IDLE state without setting manual calibration mode ( <code>MCSM0.FS_AUTOCAL=0</code> )
0x34	SRX	Enable RX. Perform calibration first if coming from IDLE and <code>MCSM0.FS_AUTOCAL=1</code> .
0x35	STX	In IDLE state: Enable TX. Perform calibration first if <code>MCSM0.FS_AUTOCAL=1</code> . If in RX state and CCA is enabled: Only go to TX if channel is clear.
0x36	SIDLE	Exit RX / TX, turn off frequency synthesizer and exit Wake-On-Radio mode if applicable.
0x38	SWOR	Start automatic RX polling sequence (Wake-on-Radio) as described in Section 19.5.
0x39	SPWD	Enter power down mode when <code>CSn</code> goes high.
0x3A	SFRX	Flush the RX FIFO buffer. Only issue in IDLE, <code>TXFIFO_UNDERFLOW</code> or <code>RXFIFO_OVERFLOW</code> states.
0x3B	SFTX	Flush the TX FIFO buffer. Only issue in IDLE, <code>TXFIFO_UNDERFLOW</code> or <code>RXFIFO_OVERFLOW</code> states.
0x3C	SWORRST	Reset real time clock.
0x3D	SNOP	No operation. May be used to pad strobe commands to two bytes for simpler software.

**Table 34: Command Strobes**

Address	Register	Description	Preserved in SLEEP state	Details on page number
0x00	IOCFG2	GDO2 output pin configuration	Yes	54
0x01	IOCFG1	GDO1 output pin configuration	Yes	54
0x02	IOCFG0	GDO0 output pin configuration	Yes	54
0x03	FIFOTHR	RX FIFO and TX FIFO thresholds	Yes	55
0x04	SYNC1	Sync word, high byte	Yes	55
0x05	SYNC0	Sync word, low byte	Yes	55
0x06	PKTLEN	Packet length	Yes	55
0x07	PKTCTRL1	Packet automation control	Yes	56
0x08	PKTCTRL0	Packet automation control	Yes	57
0x09	ADDR	Device address	Yes	57
0x0A	CHANNR	Channel number	Yes	57
0x0B	FSCTRL1	Frequency synthesizer control	Yes	58
0x0C	FSCTRL0	Frequency synthesizer control	Yes	58
0x0D	FREQ2	Frequency control word, high byte	Yes	58
0x0E	FREQ1	Frequency control word, middle byte	Yes	58
0x0F	FREQ0	Frequency control word, low byte	Yes	58
0x10	MDMCFG4	Modem configuration	Yes	59
0x11	MDMCFG3	Modem configuration	Yes	59
0x12	MDMCFG2	Modem configuration	Yes	60
0x13	MDMCFG1	Modem configuration	Yes	61
0x14	MDMCFG0	Modem configuration	Yes	61
0x15	DEVIATN	Modem deviation setting	Yes	62
0x16	MCSM2	Main Radio Control State Machine configuration	Yes	62
0x17	MCSM1	Main Radio Control State Machine configuration	Yes	63
0x18	MCSM0	Main Radio Control State Machine configuration	Yes	64
0x19	FOCCFG	Frequency Offset Compensation configuration	Yes	64
0x1A	BSCFG	Bit Synchronization configuration	Yes	64
0x1B	AGCTRL2	AGC control	Yes	65
0x1C	AGCTRL1	AGC control	Yes	66
0x1D	AGCTRL0	AGC control	Yes	66
0x1E	WOREVT1	High byte Event 0 timeout	Yes	66
0x1F	WOREVT0	Low byte Event 0 timeout	Yes	67
0x20	WORCTRL	Wake On Radio control	Yes	67
0x21	FREND1	Front end RX configuration	Yes	67
0x22	FREND0	Front end TX configuration	Yes	68
0x23	FSCAL3	Frequency synthesizer calibration	Yes	68
0x24	FSCAL2	Frequency synthesizer calibration	Yes	68
0x25	FSCAL1	Frequency synthesizer calibration	Yes	68
0x26	FSCAL0	Frequency synthesizer calibration	Yes	69
0x27	RCCTRL1	RC oscillator configuration	Yes	69
0x28	RCCTRL0	RC oscillator configuration	Yes	69
0x29	FSTEST	Frequency synthesizer calibration control	No	69
0x2A	PTEST	Production test	No	69
0x2B	AGCTEST	AGC test	No	69
0x2C	TEST2	Various test settings	No	70
0x2D	TEST1	Various test settings	No	70
0x2E	TEST0	Various test settings	No	70

**Table 35: Configuration Registers Overview**

Address	Register	Description	Details on page number
0x30 (0xF0)	PARTNUM	Part number for <b>CC2500</b>	70
0x31 (0xF1)	VERSION	Current version number	70
0x32 (0xF2)	FREQEST	Frequency Offset Estimate	70
0x33 (0xF3)	LQI	Demodulator estimate for Link Quality	71
0x34 (0xF4)	RSSI	Received signal strength indication	71
0x35 (0xF5)	MARCSTATE	Control state machine state	71
0x36 (0xF6)	WORTIME1	High byte of WOR timer	72
0x37 (0xF7)	WORTIME0	Low byte of WOR timer	72
0x38 (0xF8)	PKTSTATUS	Current GDOx status and packet status	72
0x39 (0xF9)	VCO_VC_DAC	Current setting from PLL calibration module	72
0x3A (0xFA)	TXBYTES	Underflow and number of bytes in the TX FIFO	72
0x3B (0xFB)	RXBYTES	Overflow and number of bytes in the RX FIFO	72

**Table 36: Status Registers Overview**

	Write		Read		
	Single byte	Burst	Single byte	Burst	
	+0x00	+0x40	+0x80	+0xC0	
0x00					IOCFG2
0x01					IOCFG1
0x02					IOCFG0
0x03					FIFOTHR
0x04					SYNC1
0x05					SYNC0
0x06					PKTLEN
0x07					PKTCTRL1
0x08					PKTCTRL0
0x09					ADDR
0x0A					CHANNR
0x0B					FSCTRL1
0x0C					FSCTRL0
0x0D					FREQ2
0x0E					FREQ1
0x0F					FREQ0
0x10					MDMCFG4
0x11					MDMCFG3
0x12					MDMCFG2
0x13					MDMCFG1
0x14					MDMCFG0
0x15					DEVIATN
0x16					MCSM2
0x17					MCSM1
0x18					MCSM0
0x19					FOCCFG
0x1A					BSCFG
0x1B					AGCCTRL2
0x1C					AGCCTRL1
0x1D					AGCCTRL0
0x1E					WOREVT1
0x1F					WOREVT0
0x20					WORCTRL
0x21					FREND1
0x22					FREND0
0x23					FSCAL3
0x24					FSCAL2
0x25					FSCAL1
0x26					FSCAL0
0x27					RCCTRL1
0x28					RCCTRL0
0x29					FSTEST
0x2A					PTEST
0x2B					AGCTEST
0x2C					TEST2
0x2D					TEST1
0x2E					TEST0
0x2F					
0x30	SRES		SRES	PARTNUM	
0x31	SFSTXON		SFSTXON	VERSION	
0x32	SXOFF		SXOFF	FREQUST	
0x33	SCAL		SCAL	LQI	
0x34	SRX		SRX	RSSI	
0x35	STX		STX	MARCSTATE	
0x36	SIDLE		SIDLE	WORTIME1	
0x37	SAFC		SAFC	WORTIME0	
0x38	SWOR		SWOR	PKTSTATUS	
0x39	SPWD		SPWD	VCO VC DAC	
0x3A	SFRX		SFRX	TXBYTES	
0x3B	SFTX		SFTX	RXBYTES	
0x3C	SWORRST		SWORRST		
0x3D	SNOP		SNOP		
0x3E	PATABLE	PATABLE	PATABLE	PATABLE	
0x3F	TX FIFO	TX FIFO	RX FIFO	RX FIFO	

R/W configuration registers, burst access possible

Command Strobes, Status registers (read only) and multi byte registers

Table 37: SPI Address Space

### 31.1 Configuration Register Details – Registers with preserved values in sleep state

#### 0x00: IOCFG2 – GDO2 output pin configuration

Bit	Field Name	Reset	R/W	Description
7	Reserved		R0	
6	GDO2_INV	0	R/W	Invert output, i.e. select active low (1) / high (0)
5:0	GDO2_CFG[5:0]	41 (0x29)	R/W	Default is CHIP_RDY (see Table 33 on page 46). Should be set to 3-state for lowest power down current.

#### 0x01: IOCFG1 – GDO1 output pin configuration

Bit	Field Name	Reset	R/W	Description
7	GDO_DS	0	R/W	Set high (1) or low (0) output drive strength on the GDO pins.
6	GDO1_INV	0	R/W	Invert output, i.e. select active low (1) / high (0)
5:0	GDO1_CFG[5:0]	46 (0x2E)	R/W	Default is 3-state (see Table 33 on page 46)

#### 0x02: IOCFG0 – GDO0 output pin configuration

Bit	Field Name	Reset	R/W	Description
7	TEMP_SENSOR_ENABLE	0	R/W	Enable analog temperature sensor. Write 0 in all other register bits when using temperature sensor.
6	GDO0_INV	0	R/W	Invert output, i.e. select active low (1) / high (0)
5:0	GDO0_CFG[5:0]	63 (0x3F)	R/W	Default is CLK_XOSC/192 (see Table 33 on page 46). Should be set to 3-state for lowest power down current.

**0x03: FIFOTHR – RX FIFO and TX FIFO thresholds**

Bit	Field Name	Reset	R/W	Description																																																			
7:3	Reserved	0	R/W	Write 0 for compatibility with possible future extensions																																																			
3:0	FIFO_THR[3:0]	7 (0111)	R/W	Set the threshold for the TX FIFO and RX FIFO. The threshold is exceeded when the number of bytes in the FIFO is equal to or higher than the threshold value. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Setting</th> <th>Bytes in TX FIFO</th> <th>Bytes in RX FIFO</th> </tr> </thead> <tbody> <tr><td>0 (0000)</td><td>61</td><td>4</td></tr> <tr><td>1 (0001)</td><td>57</td><td>8</td></tr> <tr><td>2 (0010)</td><td>53</td><td>12</td></tr> <tr><td>3 (0011)</td><td>49</td><td>16</td></tr> <tr><td>4 (0100)</td><td>45</td><td>20</td></tr> <tr><td>5 (0101)</td><td>41</td><td>24</td></tr> <tr><td>6 (0110)</td><td>37</td><td>28</td></tr> <tr><td>7 (0111)</td><td>33</td><td>32</td></tr> <tr><td>8 (1000)</td><td>29</td><td>36</td></tr> <tr><td>9 (1001)</td><td>25</td><td>40</td></tr> <tr><td>10 (1010)</td><td>21</td><td>44</td></tr> <tr><td>11 (1011)</td><td>17</td><td>48</td></tr> <tr><td>12 (1100)</td><td>13</td><td>52</td></tr> <tr><td>13 (1101)</td><td>9</td><td>56</td></tr> <tr><td>14 (1110)</td><td>5</td><td>60</td></tr> <tr><td>15 (1111)</td><td>1</td><td>64</td></tr> </tbody> </table>	Setting	Bytes in TX FIFO	Bytes in RX FIFO	0 (0000)	61	4	1 (0001)	57	8	2 (0010)	53	12	3 (0011)	49	16	4 (0100)	45	20	5 (0101)	41	24	6 (0110)	37	28	7 (0111)	33	32	8 (1000)	29	36	9 (1001)	25	40	10 (1010)	21	44	11 (1011)	17	48	12 (1100)	13	52	13 (1101)	9	56	14 (1110)	5	60	15 (1111)	1	64
Setting	Bytes in TX FIFO	Bytes in RX FIFO																																																					
0 (0000)	61	4																																																					
1 (0001)	57	8																																																					
2 (0010)	53	12																																																					
3 (0011)	49	16																																																					
4 (0100)	45	20																																																					
5 (0101)	41	24																																																					
6 (0110)	37	28																																																					
7 (0111)	33	32																																																					
8 (1000)	29	36																																																					
9 (1001)	25	40																																																					
10 (1010)	21	44																																																					
11 (1011)	17	48																																																					
12 (1100)	13	52																																																					
13 (1101)	9	56																																																					
14 (1110)	5	60																																																					
15 (1111)	1	64																																																					

**0x04: SYNC1 – Sync word, high byte**

Bit	Field Name	Reset	R/W	Description
7:0	SYNC[15:8]	211 (0xD3)	R/W	8 MSB of 16-bit sync word

**0x05: SYNC0 – Sync word, low byte**

Bit	Field Name	Reset	R/W	Description
7:0	SYNC[7:0]	145 (0x91)	R/W	8 LSB of 16-bit sync word

**0x06: PKTLEN – Packet length**

Bit	Field Name	Reset	R/W	Description
7:0	PACKET_LENGTH	255 (0xFF)	R/W	Indicates the packet length when fixed length packets are enabled. If variable length packets are used, this value indicates the maximum length packets allowed.



**0x07: PKTCTRL1 – Packet automation control**

Bit	Field Name	Reset	R/W	Description										
7:5	PQT[2:0]	0 (000)	R/W	<p>Preamble quality estimator threshold. The preamble quality estimator increases an internal counter by one each time a bit is received that is different from the previous bit, and decreases the counter by 4 each time a bit is received that is the same as the last bit. The counter saturates at 0 and 31.</p> <p>A threshold of 4-PQT for this counter is used to gate sync word detection. When PQT=0 a sync word is always accepted.</p>										
4	Reserved	0	R/W											
3	CRC_AUTOFLUSH	0	R/W	Enable automatic flush of RX FIFO when CRC is not OK. This requires that only one packet is in the RX FIFO and that packet length is limited to the RX FIFO size.										
2	APPEND_STATUS	1	R/W	When enabled, two status bytes will be appended to the payload of the packet. The status bytes contain RSSI and LQI values, as well as the CRC OK flag.										
1:0	ADR_CHK[1:0]	0 (00)	R/W	<p>Controls address check configuration of received packages.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Address check configuration</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>No address check</td> </tr> <tr> <td>1 (01)</td> <td>Address check, no broadcast</td> </tr> <tr> <td>2 (10)</td> <td>Address check, 0 (0x00) broadcast</td> </tr> <tr> <td>3 (11)</td> <td>Address check, 0 (0x00) and 255 (0xFF) broadcast</td> </tr> </tbody> </table>	Setting	Address check configuration	0 (00)	No address check	1 (01)	Address check, no broadcast	2 (10)	Address check, 0 (0x00) broadcast	3 (11)	Address check, 0 (0x00) and 255 (0xFF) broadcast
Setting	Address check configuration													
0 (00)	No address check													
1 (01)	Address check, no broadcast													
2 (10)	Address check, 0 (0x00) broadcast													
3 (11)	Address check, 0 (0x00) and 255 (0xFF) broadcast													

**0x08: PKTCTRL0 – Packet automation control**

Bit	Field Name	Reset	R/W	Description										
7	Reserved		R0											
6	WHITE_DATA	1	R/W	Turn data whitening on / off 0: Whitening off 1: Whitening on										
5:4	PKT_FORMAT[1:0]	0 (00)	R/W	Format of RX and TX data <table border="1"> <thead> <tr> <th>Setting</th> <th>Packet format</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>Normal mode, use FIFOs for RX and TX</td> </tr> <tr> <td>1 (01)</td> <td>Serial Synchronous mode, used for backwards compatibility. Data in on GDO0</td> </tr> <tr> <td>2 (10)</td> <td>Random TX mode; sends random data using PN9 generator. Used for test. Works as normal mode, setting 0 (00), in RX.</td> </tr> <tr> <td>3 (11)</td> <td>Asynchronous transparent mode. Data in on GDO0 and Data out on either of the GDO pins</td> </tr> </tbody> </table>	Setting	Packet format	0 (00)	Normal mode, use FIFOs for RX and TX	1 (01)	Serial Synchronous mode, used for backwards compatibility. Data in on GDO0	2 (10)	Random TX mode; sends random data using PN9 generator. Used for test. Works as normal mode, setting 0 (00), in RX.	3 (11)	Asynchronous transparent mode. Data in on GDO0 and Data out on either of the GDO pins
Setting	Packet format													
0 (00)	Normal mode, use FIFOs for RX and TX													
1 (01)	Serial Synchronous mode, used for backwards compatibility. Data in on GDO0													
2 (10)	Random TX mode; sends random data using PN9 generator. Used for test. Works as normal mode, setting 0 (00), in RX.													
3 (11)	Asynchronous transparent mode. Data in on GDO0 and Data out on either of the GDO pins													
3	CC2400_EN	0	R/W	Enable CC2400 support. Use same CRC implementation as CC2400.										
2	CRC_EN	1	R/W	1: CRC calculation in TX and CRC check in RX enabled 0: CRC disabled for TX and RX										
1:0	LENGTH_CONFIG[1:0]	1 (01)	R/W	Configure the packet length <table border="1"> <thead> <tr> <th>Setting</th> <th>Packet length configuration</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>Fixed length packets, length configured in PKTLEN register</td> </tr> <tr> <td>1 (01)</td> <td>Variable length packets, packet length configured by the first byte after sync word</td> </tr> <tr> <td>2 (10)</td> <td>Enable infinite length packets</td> </tr> <tr> <td>3 (11)</td> <td>Reserved</td> </tr> </tbody> </table>	Setting	Packet length configuration	0 (00)	Fixed length packets, length configured in PKTLEN register	1 (01)	Variable length packets, packet length configured by the first byte after sync word	2 (10)	Enable infinite length packets	3 (11)	Reserved
Setting	Packet length configuration													
0 (00)	Fixed length packets, length configured in PKTLEN register													
1 (01)	Variable length packets, packet length configured by the first byte after sync word													
2 (10)	Enable infinite length packets													
3 (11)	Reserved													

**0x09: ADDR – Device address**

Bit	Field Name	Reset	R/W	Description
7:0	DEVICE_ADDR[7:0]	0 (0x00)	R/W	Address used for packet filtration. Optional broadcast addresses are 0 (0x00) and 255 (0xFF).

**0x0A: CHANNR – Channel number**

Bit	Field Name	Reset	R/W	Description
7:0	CHAN[7:0]	0 (0x00)	R/W	The 8-bit unsigned channel number, which is multiplied by the channel spacing setting and added to the base frequency.

**0x0B: FSCTRL1 – Frequency synthesizer control**

Bit	Field Name	Reset	R/W	Description
7:5	Reserved		R0	
4:0	FREQ_IF[4:0]	10 (0x0F)	R/W	<p>The desired IF frequency to employ in RX. Subtracted from FS base frequency in RX and controls the digital complex mixer in the demodulator.</p> $f_{IF} = \frac{f_{XOSC}}{2^{10}} \cdot FREQ\_IF$ <p>The default value gives an IF frequency of 254 kHz, assuming a 26.0 MHz crystal.</p>

**0x0C: FSCTRL0 – Frequency synthesizer control**

Bit	Field Name	Reset	R/W	Description
7:0	FREQOFF[7:0]	0 (0x00)	R/W	<p>Frequency offset added to the base frequency before being used by the FS. (2-complement).</p> <p>Resolution is <math>F_{XTAL}/2^{14}</math> (1.59 - 1.65 kHz); range is <math>\pm 202</math> kHz to <math>\pm 210</math> kHz, dependent of XTAL frequency.</p>

**0x0D: FREQ2 – Frequency control word, high byte**

Bit	Field Name	Reset	R/W	Description															
7:6	FREQ[23:22]	1 (01)	R	FREQ[23:22] is always binary 01 (the FREQ2 register is in the range 85 to 95 with 26-27 MHz crystal)															
5:0	FREQ[21:16]	30 (0x1E)	R/W	<p>FREQ[23:0] is the base frequency for the frequency synthesiser in increments of <math>F_{XOSC}/2^{16}</math>.</p> $f_{carrier} = \frac{f_{XOSC}}{2^{16}} \cdot FREQ[23:0]$ <p>The default frequency word gives a base frequency of 2464 MHz, assuming a 26.0 MHz crystal. With the default channel spacing settings, the following FREQ2 values and channel numbers can be used:</p> <table border="1"> <thead> <tr> <th>FREQ2</th> <th>Base frequency</th> <th>Frequency range (CHAN numbers)</th> </tr> </thead> <tbody> <tr> <td>91 (0x5B)</td> <td>2386 MHz</td> <td>2400.2-2437 MHz (71-255)</td> </tr> <tr> <td>92 (0x5C)</td> <td>2412 MHz</td> <td>2412-2463 MHz (0-255)</td> </tr> <tr> <td>93 (0x5D)</td> <td>2438 MHz</td> <td>2431-2483.4 MHz (0-227)</td> </tr> <tr> <td>94 (0x5E)</td> <td>2464 MHz</td> <td>2464-2483.4 MHz (0-97)</td> </tr> </tbody> </table>	FREQ2	Base frequency	Frequency range (CHAN numbers)	91 (0x5B)	2386 MHz	2400.2-2437 MHz (71-255)	92 (0x5C)	2412 MHz	2412-2463 MHz (0-255)	93 (0x5D)	2438 MHz	2431-2483.4 MHz (0-227)	94 (0x5E)	2464 MHz	2464-2483.4 MHz (0-97)
FREQ2	Base frequency	Frequency range (CHAN numbers)																	
91 (0x5B)	2386 MHz	2400.2-2437 MHz (71-255)																	
92 (0x5C)	2412 MHz	2412-2463 MHz (0-255)																	
93 (0x5D)	2438 MHz	2431-2483.4 MHz (0-227)																	
94 (0x5E)	2464 MHz	2464-2483.4 MHz (0-97)																	

**0x0E: FREQ1 – Frequency control word, middle byte**

Bit	Field Name	Reset	R/W	Description
7:0	FREQ[15:8]	196 (0xC4)	R/W	Ref. FREQ2 register

**0x0F: FREQ0 – Frequency control word, low byte**

Bit	Field Name	Reset	R/W	Description
7:0	FREQ[7:0]	236 (0xEC)	R/W	Ref. FREQ2 register

**0x10: MDMCFG4 – Modem configuration**

Bit	Field Name	Reset	R/W	Description
7:6	CHANBW_E[1:0]	2 (10)	R/W	
5:4	CHANBW_M[1:0]	0 (00)	R/W	<p>Sets the decimation ratio for the delta-sigma ADC input stream and thus the channel bandwidth.</p> $BW_{channel} = \frac{f_{XOSC}}{8 \cdot (4 + CHANBW\_M) \cdot 2^{CHANBW\_E}}$ <p>The default values give 203 kHz channel filter bandwidth, assuming a 26.0 MHz crystal.</p>
3:0	DRATE_E[3:0]	12 (1100)	R/W	The exponent of the user specified symbol rate

**0x11: MDMCFG3 – Modem configuration**

Bit	Field Name	Reset	R/W	Description
7:0	DRATE_M[7:0]	34 (0x22)	R/W	<p>The mantissa of the user specified symbol rate. The symbol rate is configured using an unsigned, floating-point number with 9-bit mantissa and 4-bit exponent. The 9<sup>th</sup> bit is a hidden '1'. The resulting data rate is:</p> $R_{DATA} = \frac{(256 + DRATE\_M) \cdot 2^{DRATE\_E}}{2^{28}} \cdot f_{XOSC}$ <p>The default values give a data rate of 115.051 kbps (closest setting to 115.2 kbps), assuming a 26.0 MHz crystal.</p>

**0x12: MDMCFG2 – Modem configuration**

Bit	Field Name	Reset	R/W	Description																		
7	DEM_DCFILT_OFF	0	R/W	Disable digital DC blocking filter before demodulator. 0 = Enable (better sensitivity for data rates ≤ 250 kbps) 1 = Disable (reduced current)  The recommended IF frequency changes when the DC blocking is disabled.																		
6:4	MOD_FORMAT[2:0]	0 (000)	R/W	The modulation format of the radio signal <table border="1"> <thead> <tr> <th>Setting</th> <th>Modulation format</th> </tr> </thead> <tbody> <tr> <td>0 (000)</td> <td>2-FSK</td> </tr> <tr> <td>1 (001)</td> <td>GFSK</td> </tr> <tr> <td>2 (010)</td> <td>-</td> </tr> <tr> <td>3 (011)</td> <td>OOK</td> </tr> <tr> <td>4 (100)</td> <td>-</td> </tr> <tr> <td>5 (101)</td> <td>-</td> </tr> <tr> <td>6 (110)</td> <td>-</td> </tr> <tr> <td>7 (111)</td> <td>MSK</td> </tr> </tbody> </table>	Setting	Modulation format	0 (000)	2-FSK	1 (001)	GFSK	2 (010)	-	3 (011)	OOK	4 (100)	-	5 (101)	-	6 (110)	-	7 (111)	MSK
Setting	Modulation format																					
0 (000)	2-FSK																					
1 (001)	GFSK																					
2 (010)	-																					
3 (011)	OOK																					
4 (100)	-																					
5 (101)	-																					
6 (110)	-																					
7 (111)	MSK																					
3	MANCHESTER_EN	0	R/W	Enables Manchester encoding/decoding. 0 = Disable 1 = Enable																		
2:0	SYNC_MODE[2:0]	2 (010)	R/W	Combined sync-word qualifier mode.  The values 0 (000) and 4 (100) disables sync word transmission in TX and sync word detection in RX.  The values 1 (001), 2 (001), 5 (101) and 6 (110) enables 16-bit sync word transmission in TX and 16-bits sync word detection in RX. Only 15 of 16 bits need to match in RX when using setting 1 (001) or 5 (101). The values 3 (011) and 7 (111) enables repeated sync word transmission in RX and 32-bits sync word detection in RX (only 30 of 32 bits need to match). <table border="1"> <thead> <tr> <th>Setting</th> <th>Sync-word qualifier mode</th> </tr> </thead> <tbody> <tr> <td>0 (000)</td> <td>No preamble/sync</td> </tr> <tr> <td>1 (001)</td> <td>15/16 sync word bits detected</td> </tr> <tr> <td>2 (010)</td> <td>16/16 sync word bits detected</td> </tr> <tr> <td>3 (011)</td> <td>30/32 sync word bits detected</td> </tr> <tr> <td>4 (100)</td> <td>No preamble/sync, carrier-sense above threshold</td> </tr> <tr> <td>5 (101)</td> <td>15/16 + carrier-sense above threshold</td> </tr> <tr> <td>6 (110)</td> <td>16/16 + carrier-sense above threshold</td> </tr> <tr> <td>7 (111)</td> <td>30/32 + carrier-sense above threshold</td> </tr> </tbody> </table>	Setting	Sync-word qualifier mode	0 (000)	No preamble/sync	1 (001)	15/16 sync word bits detected	2 (010)	16/16 sync word bits detected	3 (011)	30/32 sync word bits detected	4 (100)	No preamble/sync, carrier-sense above threshold	5 (101)	15/16 + carrier-sense above threshold	6 (110)	16/16 + carrier-sense above threshold	7 (111)	30/32 + carrier-sense above threshold
Setting	Sync-word qualifier mode																					
0 (000)	No preamble/sync																					
1 (001)	15/16 sync word bits detected																					
2 (010)	16/16 sync word bits detected																					
3 (011)	30/32 sync word bits detected																					
4 (100)	No preamble/sync, carrier-sense above threshold																					
5 (101)	15/16 + carrier-sense above threshold																					
6 (110)	16/16 + carrier-sense above threshold																					
7 (111)	30/32 + carrier-sense above threshold																					

**0x13: MDMCFG1 – Modem configuration**

Bit	Field Name	Reset	R/W	Description																		
7	FEC_EN	0	R/W	Enable Forward Error Correction (FEC) with interleaving for packet payload 0 = Disable 1 = Enable																		
6:4	NUM_PREAMBLE[2:0]	2 (010)	R/W	Sets the minimum number of preamble bytes to be transmitted <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Setting</th> <th>Number of preamble bytes</th> </tr> </thead> <tbody> <tr><td>0 (000)</td><td>2</td></tr> <tr><td>1 (001)</td><td>3</td></tr> <tr><td>2 (010)</td><td>4</td></tr> <tr><td>3 (011)</td><td>6</td></tr> <tr><td>4 (100)</td><td>8</td></tr> <tr><td>5 (101)</td><td>12</td></tr> <tr><td>6 (110)</td><td>16</td></tr> <tr><td>7 (111)</td><td>24</td></tr> </tbody> </table>	Setting	Number of preamble bytes	0 (000)	2	1 (001)	3	2 (010)	4	3 (011)	6	4 (100)	8	5 (101)	12	6 (110)	16	7 (111)	24
Setting	Number of preamble bytes																					
0 (000)	2																					
1 (001)	3																					
2 (010)	4																					
3 (011)	6																					
4 (100)	8																					
5 (101)	12																					
6 (110)	16																					
7 (111)	24																					
3:2	Reserved		R0																			
1:0	CHANSPC_E[1:0]	2 (10)	R/W	2 bit exponent of channel spacing																		

**0x14: MDMCFG0 – Modem configuration**

Bit	Field Name	Reset	R/W	Description
7:0	CHANSPC_M[7:0]	248 (0xF8)	R/W	8-bit mantissa of channel spacing (initial 1 assumed). The channel spacing is multiplied by the channel number CHAN and added to the base frequency. It is unsigned and has the format:  $\Delta f_{CHANNEL} = \frac{f_{XOSC}}{2^{18}} \cdot (256 + CHANSPC\_M) \cdot 2^{CHANSPC\_E} \cdot CHAN$ The default values give 199.951 kHz channel spacing (the closest setting to 200 kHz), assuming 26.0 MHz crystal frequency.

0x15: DEVIATN – Modem deviation setting

Bit	Field Name	Reset	R/W	Description
7	Reserved		R0	
6:4	DEVIATION_E[2:0]	4 (100)	R/W	Deviation exponent
3	Reserved		R0	
2:0	DEVIATION_M[2:0]	7 (111)	R/W	<p>When MSK modulation is enabled: Sets fraction of symbol period used for phase change.</p> <p>When FSK modulation is enabled: Deviation mantissa, interpreted as a 4-bit value with MSB implicit 1. The resulting FSK deviation is given by:</p> $f_{dev} = \frac{f_{xosc}}{2^{17}} \cdot (8 + DEVIATION\_M) \cdot 2^{DEVIATION\_E}$ <p>The default values give ±47.607 kHz deviation, assuming 26.0 MHz crystal frequency.</p>

0x16: MCSM2 – Main Radio Control State Machine configuration

Bit	Field Name	Reset	R/W	Description																											
7:5	Reserved		R0	Reserved																											
4	RX_TIME_RSSI	0	R/W	Direct RX termination based on RSSI measurement (carrier sense). For OOK modulation, RX times out if there is no carrier sense in the first 8 symbol periods.																											
3	RX_TIME_QUAL	0	R/W	When the RX_TIME timer expires the chip stays in RX mode if sync word is found when RX_TIME_QUAL=0, or either sync word is found or PQT is set when RX_TIME_QUAL=1.																											
2:0	RX_TIME[2:0]	7 (111)	R/W	<p>Timeout for sync word search in RX. The timeout is relative to the programmed EVENT0 timeout, which means that the duty cycle can be set in wake-on-radio (WOR) mode. The RX timeout is scaled by 1 bit less than the EVENT0 timeout with respect to the WORCTRL.WOR_RES setting, as very long timeouts probably also will use very low RX duty cycles.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>RX timeout</th> <th>Duty cycle, WOR</th> </tr> </thead> <tbody> <tr> <td>0 (000)</td> <td><math>T_{EVENT0} / 2^{(3+WOR\_RES)}</math></td> <td>12.5% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>1 (001)</td> <td><math>T_{EVENT0} / 2^{(4+WOR\_RES)}</math></td> <td>6.25% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>2 (010)</td> <td><math>T_{EVENT0} / 2^{(5+WOR\_RES)}</math></td> <td>3.125% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>3 (011)</td> <td><math>T_{EVENT0} / 2^{(6+WOR\_RES)}</math></td> <td>1.563% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>4 (100)</td> <td><math>T_{EVENT0} / 2^{(7+WOR\_RES)}</math></td> <td>0.781% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>5 (101)</td> <td><math>T_{EVENT0} / 2^{(8+WOR\_RES)}</math></td> <td>0.391% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>6 (110)</td> <td><math>T_{EVENT0} / 2^{(9+WOR\_RES)}</math></td> <td>0.195% / <math>2^{WOR\_RES}</math></td> </tr> <tr> <td>7 (111)</td> <td>Until end of packet</td> <td>N/A (no timeout)</td> </tr> </tbody> </table> <p>Note that the RC oscillator must be enabled in order to use setting 0-6, because the timeout counts RC oscillator periods. WOR mode does not need to be enabled. The timeout counter resolution is limited: With RX_TIME=0, the timeout count is given by the 13 MSBs of EVENT0, decreasing to the 7 MSBs of EVENT0 with RX_TIME=6.</p>	Setting	RX timeout	Duty cycle, WOR	0 (000)	$T_{EVENT0} / 2^{(3+WOR\_RES)}$	12.5% / $2^{WOR\_RES}$	1 (001)	$T_{EVENT0} / 2^{(4+WOR\_RES)}$	6.25% / $2^{WOR\_RES}$	2 (010)	$T_{EVENT0} / 2^{(5+WOR\_RES)}$	3.125% / $2^{WOR\_RES}$	3 (011)	$T_{EVENT0} / 2^{(6+WOR\_RES)}$	1.563% / $2^{WOR\_RES}$	4 (100)	$T_{EVENT0} / 2^{(7+WOR\_RES)}$	0.781% / $2^{WOR\_RES}$	5 (101)	$T_{EVENT0} / 2^{(8+WOR\_RES)}$	0.391% / $2^{WOR\_RES}$	6 (110)	$T_{EVENT0} / 2^{(9+WOR\_RES)}$	0.195% / $2^{WOR\_RES}$	7 (111)	Until end of packet	N/A (no timeout)
Setting	RX timeout	Duty cycle, WOR																													
0 (000)	$T_{EVENT0} / 2^{(3+WOR\_RES)}$	12.5% / $2^{WOR\_RES}$																													
1 (001)	$T_{EVENT0} / 2^{(4+WOR\_RES)}$	6.25% / $2^{WOR\_RES}$																													
2 (010)	$T_{EVENT0} / 2^{(5+WOR\_RES)}$	3.125% / $2^{WOR\_RES}$																													
3 (011)	$T_{EVENT0} / 2^{(6+WOR\_RES)}$	1.563% / $2^{WOR\_RES}$																													
4 (100)	$T_{EVENT0} / 2^{(7+WOR\_RES)}$	0.781% / $2^{WOR\_RES}$																													
5 (101)	$T_{EVENT0} / 2^{(8+WOR\_RES)}$	0.391% / $2^{WOR\_RES}$																													
6 (110)	$T_{EVENT0} / 2^{(9+WOR\_RES)}$	0.195% / $2^{WOR\_RES}$																													
7 (111)	Until end of packet	N/A (no timeout)																													

**0x17: MCSM1 – Main Radio Control State Machine configuration**

Bit	Field Name	Reset	R/W	Description										
7:6	Reserved		R0											
5:4	CCA_MODE[1:0]	3 (11)	R/W	<p>Selects CCA_MODE; Reflected in CCA signal</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Clear channel indication</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>Always</td> </tr> <tr> <td>1 (01)</td> <td>If RSSI below threshold</td> </tr> <tr> <td>2 (10)</td> <td>Unless currently receiving a packet</td> </tr> <tr> <td>3 (11)</td> <td>If RSSI below threshold unless currently receiving a packet</td> </tr> </tbody> </table>	Setting	Clear channel indication	0 (00)	Always	1 (01)	If RSSI below threshold	2 (10)	Unless currently receiving a packet	3 (11)	If RSSI below threshold unless currently receiving a packet
Setting	Clear channel indication													
0 (00)	Always													
1 (01)	If RSSI below threshold													
2 (10)	Unless currently receiving a packet													
3 (11)	If RSSI below threshold unless currently receiving a packet													
3:2	RXOFF_MODE[1:0]	0 (00)	R/W	<p>Select what should happen when a packet has been received</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Next state after finishing packet reception</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>IDLE</td> </tr> <tr> <td>1 (01)</td> <td>FSTXON</td> </tr> <tr> <td>2 (10)</td> <td>TX</td> </tr> <tr> <td>3 (11)</td> <td>Stay in RX</td> </tr> </tbody> </table> <p>It is not possible to set RXOFF_MODE to be TX or FSTXON and at the same time use CCA.</p>	Setting	Next state after finishing packet reception	0 (00)	IDLE	1 (01)	FSTXON	2 (10)	TX	3 (11)	Stay in RX
Setting	Next state after finishing packet reception													
0 (00)	IDLE													
1 (01)	FSTXON													
2 (10)	TX													
3 (11)	Stay in RX													
1:0	TXOFF_MODE[1:0]	0 (00)	R/W	<p>Select what should happen when a packet has been sent (TX)</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Next state after finishing packet transmission</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>IDLE</td> </tr> <tr> <td>1 (01)</td> <td>FSTXON</td> </tr> <tr> <td>2 (10)</td> <td>Stay in TX (start sending preamble)</td> </tr> <tr> <td>3 (11)</td> <td>RX</td> </tr> </tbody> </table>	Setting	Next state after finishing packet transmission	0 (00)	IDLE	1 (01)	FSTXON	2 (10)	Stay in TX (start sending preamble)	3 (11)	RX
Setting	Next state after finishing packet transmission													
0 (00)	IDLE													
1 (01)	FSTXON													
2 (10)	Stay in TX (start sending preamble)													
3 (11)	RX													



**0x18: MCSM0 – Main Radio Control State Machine configuration**

Bit	Field Name	Reset	R/W	Description															
7:6	Reserved		R0																
5:4	FS_AUTOCAL[1:0]	0 (00)	R/W	<p>Automatically calibrate when going to RX or TX, or back to IDLE</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>When to perform automatic calibration</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>Never (manually calibrate using SCAL strobe)</td> </tr> <tr> <td>1 (01)</td> <td>When going from IDLE to RX or TX (or FSTXON)</td> </tr> <tr> <td>2 (10)</td> <td>When going from RX or TX back to IDLE</td> </tr> <tr> <td>3 (11)</td> <td>Every 4<sup>th</sup> time when going from RX or TX to IDLE</td> </tr> </tbody> </table> <p>In some automatic wake-on-radio (WOR) applications, using setting 3 (11) can significantly reduce current consumption.</p>	Setting	When to perform automatic calibration	0 (00)	Never (manually calibrate using SCAL strobe)	1 (01)	When going from IDLE to RX or TX (or FSTXON)	2 (10)	When going from RX or TX back to IDLE	3 (11)	Every 4 <sup>th</sup> time when going from RX or TX to IDLE					
Setting	When to perform automatic calibration																		
0 (00)	Never (manually calibrate using SCAL strobe)																		
1 (01)	When going from IDLE to RX or TX (or FSTXON)																		
2 (10)	When going from RX or TX back to IDLE																		
3 (11)	Every 4 <sup>th</sup> time when going from RX or TX to IDLE																		
3:2	PO_TIMEOUT	1 (01)	R/W	<p>Programs the number of times the six-bit ripple counter must expire before CHP_RDY_N goes low. Values other than 0 (00) are most useful when the XOSC is left on during power-down.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Expire count</th> <th>Timeout after XOSC start</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>1</td> <td>Approx. 2.3 – 2.7 <math>\mu</math>s</td> </tr> <tr> <td>1 (01)</td> <td>16</td> <td>Approx. 37 – 43 <math>\mu</math>s</td> </tr> <tr> <td>2 (10)</td> <td>64</td> <td>Approx. 146 – 171 <math>\mu</math>s</td> </tr> <tr> <td>3 (11)</td> <td>256</td> <td>Approx. 585 – 683 <math>\mu</math>s</td> </tr> </tbody> </table> <p>Exact timeout depends on crystal frequency.</p>	Setting	Expire count	Timeout after XOSC start	0 (00)	1	Approx. 2.3 – 2.7 $\mu$ s	1 (01)	16	Approx. 37 – 43 $\mu$ s	2 (10)	64	Approx. 146 – 171 $\mu$ s	3 (11)	256	Approx. 585 – 683 $\mu$ s
Setting	Expire count	Timeout after XOSC start																	
0 (00)	1	Approx. 2.3 – 2.7 $\mu$ s																	
1 (01)	16	Approx. 37 – 43 $\mu$ s																	
2 (10)	64	Approx. 146 – 171 $\mu$ s																	
3 (11)	256	Approx. 585 – 683 $\mu$ s																	
1	PIN_CTRL_EN	0	R/W	Enables the pin radio control option															
0	XOSC_FORCE_ON	0	R/W	Force the XOSC to stay on in the SLEEP state.															

**0x19: FOCCFG – Frequency Offset Compensation configuration**

Bit	Field Name	Reset	R/W	Description
7:6	Reserved		R0	
5:0	FOCCFG[5:0]	54 (0x36)	R/W	Frequency offset compensation configuration. The value to use in this register is given by the SmartRF <sup>®</sup> Studio software.

**0x1A: BSCFG – Bit Synchronization configuration**

Bit	Field Name	Reset	R/W	Description
7:0	BSCFG[7:0]	108 (0x6C)	R/W	Bit Synchronization configuration. The value to use in this register is given by the SmartRF <sup>®</sup> Studio software.

**0x1B: AGCCTRL2 – AGC control**

Bit	Field Name	Reset	R/W	Description																		
7:6	MAX_DVGA_GAIN[1:0]	0 (00)	R/W	<p>Reduces the maximum allowable DVGA gain.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Allowable DVGA settings</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>All gain settings can be used</td> </tr> <tr> <td>1 (01)</td> <td>The highest gain setting can not be used</td> </tr> <tr> <td>2 (10)</td> <td>The 2 highest gain settings can not be used</td> </tr> <tr> <td>3 (11)</td> <td>The 3 highest gain settings can not be used</td> </tr> </tbody> </table>	Setting	Allowable DVGA settings	0 (00)	All gain settings can be used	1 (01)	The highest gain setting can not be used	2 (10)	The 2 highest gain settings can not be used	3 (11)	The 3 highest gain settings can not be used								
Setting	Allowable DVGA settings																					
0 (00)	All gain settings can be used																					
1 (01)	The highest gain setting can not be used																					
2 (10)	The 2 highest gain settings can not be used																					
3 (11)	The 3 highest gain settings can not be used																					
5:3	MAX_LNA_GAIN[2:0]	0 (000)	R/W	<p>Sets the maximum allowable LNA + LNA 2 gain relative to the maximum possible gain.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Maximum allowable LNA + LNA 2 gain</th> </tr> </thead> <tbody> <tr> <td>0 (000)</td> <td>Maximum possible LNA + LNA 2 gain</td> </tr> <tr> <td>1 (001)</td> <td>Approx. 2.6 dB below maximum possible gain</td> </tr> <tr> <td>2 (010)</td> <td>Approx. 6.1 dB below maximum possible gain</td> </tr> <tr> <td>3 (011)</td> <td>Approx. 7.4 dB below maximum possible gain</td> </tr> <tr> <td>4 (100)</td> <td>Approx. 9.2 dB below maximum possible gain</td> </tr> <tr> <td>5 (101)</td> <td>Approx. 11.5 dB below maximum possible gain</td> </tr> <tr> <td>6 (110)</td> <td>Approx. 14.6 dB below maximum possible gain</td> </tr> <tr> <td>7 (111)</td> <td>Approx. 17.1 dB below maximum possible gain</td> </tr> </tbody> </table>	Setting	Maximum allowable LNA + LNA 2 gain	0 (000)	Maximum possible LNA + LNA 2 gain	1 (001)	Approx. 2.6 dB below maximum possible gain	2 (010)	Approx. 6.1 dB below maximum possible gain	3 (011)	Approx. 7.4 dB below maximum possible gain	4 (100)	Approx. 9.2 dB below maximum possible gain	5 (101)	Approx. 11.5 dB below maximum possible gain	6 (110)	Approx. 14.6 dB below maximum possible gain	7 (111)	Approx. 17.1 dB below maximum possible gain
Setting	Maximum allowable LNA + LNA 2 gain																					
0 (000)	Maximum possible LNA + LNA 2 gain																					
1 (001)	Approx. 2.6 dB below maximum possible gain																					
2 (010)	Approx. 6.1 dB below maximum possible gain																					
3 (011)	Approx. 7.4 dB below maximum possible gain																					
4 (100)	Approx. 9.2 dB below maximum possible gain																					
5 (101)	Approx. 11.5 dB below maximum possible gain																					
6 (110)	Approx. 14.6 dB below maximum possible gain																					
7 (111)	Approx. 17.1 dB below maximum possible gain																					
2:0	MAGN_TARGET[2:0]	3 (011)	R/W	<p>These bits set the target value for the averaged amplitude from the digital channel filter (1 LSB = 0 dB).</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Target amplitude from channel filter</th> </tr> </thead> <tbody> <tr> <td>0 (000)</td> <td>24 dB</td> </tr> <tr> <td>1 (001)</td> <td>27 dB</td> </tr> <tr> <td>2 (010)</td> <td>30 dB</td> </tr> <tr> <td>3 (011)</td> <td>33 dB</td> </tr> <tr> <td>4 (100)</td> <td>36 dB</td> </tr> <tr> <td>5 (101)</td> <td>38 dB</td> </tr> <tr> <td>6 (110)</td> <td>40 dB</td> </tr> <tr> <td>7 (111)</td> <td>42 dB</td> </tr> </tbody> </table>	Setting	Target amplitude from channel filter	0 (000)	24 dB	1 (001)	27 dB	2 (010)	30 dB	3 (011)	33 dB	4 (100)	36 dB	5 (101)	38 dB	6 (110)	40 dB	7 (111)	42 dB
Setting	Target amplitude from channel filter																					
0 (000)	24 dB																					
1 (001)	27 dB																					
2 (010)	30 dB																					
3 (011)	33 dB																					
4 (100)	36 dB																					
5 (101)	38 dB																					
6 (110)	40 dB																					
7 (111)	42 dB																					

**0x1C: AGCCTRL1 – AGC control**

Bit	Field Name	Reset	R/W	Description																		
7	Reserved		R0																			
6	AGC_LNA_PRIORITY	1	R/W	Selects between two different strategies for LNA and LNA 2 gain adjustment. When 1, the LNA gain is decreased first. When 0, the LNA 2 gain is decreased to minimum before decreasing LNA gain.																		
5:4	CARRIER_SENSE_REL_THR[1:0]	0 (00)	R/W	<p>Sets the relative change threshold for asserting carrier sense</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Carrier sense relative threshold</th> </tr> </thead> <tbody> <tr> <td>0 (00)</td> <td>Relative carrier sense threshold disabled</td> </tr> <tr> <td>1 (01)</td> <td>6 dB increase in RSSI value</td> </tr> <tr> <td>2 (10)</td> <td>10 dB increase in RSSI value</td> </tr> <tr> <td>3 (11)</td> <td>14 dB increase in RSSI value</td> </tr> </tbody> </table>	Setting	Carrier sense relative threshold	0 (00)	Relative carrier sense threshold disabled	1 (01)	6 dB increase in RSSI value	2 (10)	10 dB increase in RSSI value	3 (11)	14 dB increase in RSSI value								
Setting	Carrier sense relative threshold																					
0 (00)	Relative carrier sense threshold disabled																					
1 (01)	6 dB increase in RSSI value																					
2 (10)	10 dB increase in RSSI value																					
3 (11)	14 dB increase in RSSI value																					
3:0	CARRIER_SENSE_ABS_THR[3:0]	0 (0000)	R/W	<p>Sets the absolute RSSI threshold for asserting carrier sense. The 2-complement signed threshold is programmed in steps of 1 dB and is relative to the MAGN_TARGET setting.</p> <table border="1"> <thead> <tr> <th>Setting</th> <th>Carrier sense absolute threshold (Equal to channel filter amplitude when AGC has not decreased gain)</th> </tr> </thead> <tbody> <tr> <td>-8 (1000)</td> <td>Absolute carrier sense threshold disabled</td> </tr> <tr> <td>-7 (1001)</td> <td>7 dB below MAGN_TARGET setting</td> </tr> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>-1 (1111)</td> <td>1 dB below MAGN_TARGET setting</td> </tr> <tr> <td>0 (0000)</td> <td>At MAGN_TARGET setting</td> </tr> <tr> <td>1 (0001)</td> <td>1 dB above MAGN_TARGET setting</td> </tr> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>7 (0111)</td> <td>7 dB above MAGN_TARGET setting</td> </tr> </tbody> </table>	Setting	Carrier sense absolute threshold (Equal to channel filter amplitude when AGC has not decreased gain)	-8 (1000)	Absolute carrier sense threshold disabled	-7 (1001)	7 dB below MAGN_TARGET setting	...	...	-1 (1111)	1 dB below MAGN_TARGET setting	0 (0000)	At MAGN_TARGET setting	1 (0001)	1 dB above MAGN_TARGET setting	...	...	7 (0111)	7 dB above MAGN_TARGET setting
Setting	Carrier sense absolute threshold (Equal to channel filter amplitude when AGC has not decreased gain)																					
-8 (1000)	Absolute carrier sense threshold disabled																					
-7 (1001)	7 dB below MAGN_TARGET setting																					
...	...																					
-1 (1111)	1 dB below MAGN_TARGET setting																					
0 (0000)	At MAGN_TARGET setting																					
1 (0001)	1 dB above MAGN_TARGET setting																					
...	...																					
7 (0111)	7 dB above MAGN_TARGET setting																					

**0x1D: AGCCTRL0 – AGC control**

Bit	Field Name	Reset	R/W	Description
7:0	AGCCTRL0[7:0]	145 (0x91)	R/W	AGC control register. The value to use in this register is given by the SmartRF® Studio software.

**0x1E: WOREVT1 – High byte Event0 timeout**

Bit	Field Name	Reset	R/W	Description
7:0	EVENT0[15:8]	135 (0x87)	R/W	<p>High byte of Event 0 timeout register</p> $t_{Event0} = \frac{750}{f_{XOSC}} \cdot EVENT0 \cdot 2^{5-WOR\_RES}$

**0x1F: WOREVT0 – Low byte Event0 timeout**

Bit	Field Name	Reset	R/W	Description
7:0	EVENT0[7:0]	107 (0x6B)	R/W	Low byte of Event 0 timeout register. The default Event 0 value gives 1.0s timeout, assuming a 26.0 MHz crystal.

**0x20: WORCTRL – Wake On Radio control**

Bit	Field Name	Reset	R/W	Description																											
7	RC_PD	1	R/W	Power down signal to RC oscillator. When written to 0, automatic initial calibration will be performed																											
6:4	EVENT1[2:0]	7 (111)	R/W	Timeout setting from register block. Decoded to Event 1 timeout. RC oscillator clock frequency equals $F_{XOSC}/750$ , which is 34.7-36 kHz, depending on crystal frequency. The table below lists the number of clock periods after Event 0 before Event 1 times out. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Setting</th> <th>WOR_AUTOSYNC=0</th> <th>WOR_AUTOSYNC=1</th> </tr> </thead> <tbody> <tr><td>0 (000)</td><td>4 (0.111 – 0.115 ms)</td><td>16 (0.44 – 0.46 ms)</td></tr> <tr><td>1 (001)</td><td>6 (0.167 – 0.173 ms)</td><td>24 (0.67 – 0.69 ms)</td></tr> <tr><td>2 (010)</td><td>8 (0.222 – 0.230 ms)</td><td>32 (0.89 – 0.92 ms)</td></tr> <tr><td>3 (011)</td><td>12 (0.333 – 0.346 ms)</td><td>48 (1.33 – 1.38 ms)</td></tr> <tr><td>4 (100)</td><td>16 (0.444 – 0.462 ms)</td><td>64 (1.78 – 1.85 ms)</td></tr> <tr><td>5 (101)</td><td>24 (0.667 – 0.692 ms)</td><td>96 (2.67 – 2.77 ms)</td></tr> <tr><td>6 (110)</td><td>32 (0.889 – 0.923 ms)</td><td>128 (3.56 – 3.69 ms)</td></tr> <tr><td>7 (111)</td><td>48 (1.333 – 1.385 ms)</td><td>192 (5.33 – 5.54 ms)</td></tr> </tbody> </table>	Setting	WOR_AUTOSYNC=0	WOR_AUTOSYNC=1	0 (000)	4 (0.111 – 0.115 ms)	16 (0.44 – 0.46 ms)	1 (001)	6 (0.167 – 0.173 ms)	24 (0.67 – 0.69 ms)	2 (010)	8 (0.222 – 0.230 ms)	32 (0.89 – 0.92 ms)	3 (011)	12 (0.333 – 0.346 ms)	48 (1.33 – 1.38 ms)	4 (100)	16 (0.444 – 0.462 ms)	64 (1.78 – 1.85 ms)	5 (101)	24 (0.667 – 0.692 ms)	96 (2.67 – 2.77 ms)	6 (110)	32 (0.889 – 0.923 ms)	128 (3.56 – 3.69 ms)	7 (111)	48 (1.333 – 1.385 ms)	192 (5.33 – 5.54 ms)
Setting	WOR_AUTOSYNC=0	WOR_AUTOSYNC=1																													
0 (000)	4 (0.111 – 0.115 ms)	16 (0.44 – 0.46 ms)																													
1 (001)	6 (0.167 – 0.173 ms)	24 (0.67 – 0.69 ms)																													
2 (010)	8 (0.222 – 0.230 ms)	32 (0.89 – 0.92 ms)																													
3 (011)	12 (0.333 – 0.346 ms)	48 (1.33 – 1.38 ms)																													
4 (100)	16 (0.444 – 0.462 ms)	64 (1.78 – 1.85 ms)																													
5 (101)	24 (0.667 – 0.692 ms)	96 (2.67 – 2.77 ms)																													
6 (110)	32 (0.889 – 0.923 ms)	128 (3.56 – 3.69 ms)																													
7 (111)	48 (1.333 – 1.385 ms)	192 (5.33 – 5.54 ms)																													
3	RC_CAL	1	R/W	Enables (1) or disables (0) the RC oscillator calibration. Included for debug/test purposes only.																											
2	Reserved		R0																												
1:0	WOR_RES	0 (00)	R/W	Controls the Event 0 resolution and maximum timeout of the WOR module: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Setting</th> <th>Resolution (1 LSB)</th> <th>Max timeout</th> </tr> </thead> <tbody> <tr><td>0 (00)</td><td>1 period (28 – 29 <math>\mu</math>s)</td><td>1.8 – 1.9 seconds</td></tr> <tr><td>1 (01)</td><td><math>2^5</math> periods (0.89 – 0.92 ms)</td><td>58 – 61 seconds</td></tr> <tr><td>2 (10)</td><td><math>2^{10}</math> periods (28 – 30 ms)</td><td>31 – 32 minutes</td></tr> <tr><td>3 (11)</td><td><math>2^{15}</math> periods (0.91 – 0.94 s)</td><td>16.5 – 17.2 hours</td></tr> </tbody> </table> <p>Adjusting the resolution does not affect the resolution of the WOR time readout registers <code>WORTIME1/WORTIME0</code>.</p>	Setting	Resolution (1 LSB)	Max timeout	0 (00)	1 period (28 – 29 $\mu$ s)	1.8 – 1.9 seconds	1 (01)	$2^5$ periods (0.89 – 0.92 ms)	58 – 61 seconds	2 (10)	$2^{10}$ periods (28 – 30 ms)	31 – 32 minutes	3 (11)	$2^{15}$ periods (0.91 – 0.94 s)	16.5 – 17.2 hours												
Setting	Resolution (1 LSB)	Max timeout																													
0 (00)	1 period (28 – 29 $\mu$ s)	1.8 – 1.9 seconds																													
1 (01)	$2^5$ periods (0.89 – 0.92 ms)	58 – 61 seconds																													
2 (10)	$2^{10}$ periods (28 – 30 ms)	31 – 32 minutes																													
3 (11)	$2^{15}$ periods (0.91 – 0.94 s)	16.5 – 17.2 hours																													

**0x21: FREND1 – Front end RX configuration**

Bit	Field Name	Reset	R/W	Description
7:0	FREND1[7:0]	166 (0xA6)	R/W	Front end RX configuration. The value to use in this register is given by the SmartRF® Studio software.

**0x22: FRENDO – Front end TX configuration**

Bit	Field Name	Reset	R/W	Description
7:6	Reserved		R0	
5:4	LODIV_BUF_CURRENT_TX[1:0]	1 (01)	R/W	Adjusts current TX LO buffer (input to PA). The value to use in this field is given by the SmartRF® Studio software.
3	Reserved		R0	
2:0	PA_POWER[2:0]	0 (000)	R/W	Selects PA power setting. This value is an index to the PATABLE, which can be programmed with up to 8 different PA settings. The PATABLE settings from index '0' to the PA_POWER value are used for power ramp-up/ramp-down at the start/end of transmission in all TX modulation formats.

**0x23: FSCAL3 – Frequency synthesizer calibration**

Bit	Field Name	Reset	R/W	Description
7:6	FSCAL3[7:6]	2 (10)	R/W	Frequency synthesizer calibration configuration. The value to write in this register before calibration is given by the SmartRF® Studio software.
5:4	CHP_CURR_CAL_EN[1:0]	2 (10)	R/W	Disable charge pump calibration stage when 0
3:0	FSCAL3[3:0]	9 (1001)	R/W	Frequency synthesizer calibration result register. Fast frequency hopping without calibration for each hop can be done by calibrating upfront for each frequency and saving the resulting FSCAL3, FSCAL2 and FSCAL1 register values. Between each frequency hop, calibration can be replaced by writing the FSCAL3, FSCAL2 and FSCAL1 register values corresponding to the next RF frequency.

**0x24: FSCAL2 – Frequency synthesizer calibration**

Bit	Field Name	Reset	R/W	Description
7:6	Reserved		R0	
5:0	FSCAL2[5:0]	10 (0x0A)	R/W	Frequency synthesizer calibration result register. Fast frequency hopping without calibration for each hop can be done by calibrating upfront for each frequency and saving the resulting FSCAL3, FSCAL2 and FSCAL1 register values. Between each frequency hop, calibration can be replaced by writing the FSCAL3, FSCAL2 and FSCAL1 register values corresponding to the next RF frequency.

**0x25: FSCAL1 – Frequency synthesizer calibration**

Bit	Field Name	Reset	R/W	Description
7:6	Reserved		R0	
5:0	FSCAL1[5:0]	32 (0x20)	R/W	Frequency synthesizer calibration result register. Fast frequency hopping without calibration for each hop can be done by calibrating upfront for each frequency and saving the resulting FSCAL3, FSCAL2 and FSCAL1 register values. Between each frequency hop, calibration can be replaced by writing the FSCAL3, FSCAL2 and FSCAL1 register values corresponding to the next RF frequency.

**0x26: FSCAL0 – Frequency synthesizer calibration**

Bit	Field Name	Reset	R/W	Description
7	Reserved		R0	
4:0	FSCAL0[6:0]	13 (0x0D)	R/W	Frequency synthesizer calibration control. The value to use in this register is given by the SmartRF® Studio software.

**0x27: RCCTRL1 – RC oscillator configuration**

Bit	Field Name	Reset	R/W	Description
7	Reserved	0	R0	
6:0	RCCTRL1[6:0]	65 (0x41)	R/W	RC oscillator configuration. Do not write to this register.

**0x28: RCCTRL0 – RC oscillator configuration**

Bit	Field Name	Reset	R/W	Description
7	Reserved	0	R0	
6:0	RCCTRL0[6:0]	0 (0x00)	R/W	RC oscillator configuration. Do not write to this register.

**31.2 Configuration Register Details – Registers that lose programming in sleep state**

**0x29: FSTEST – Frequency synthesizer calibration control**

Bit	Field Name	Reset	R/W	Description
7:0	FSTEST[7:0]	87 (0x59)	R/W	For test only. Do not write to this register.

**0x2A: PTEST – Production test**

Bit	Field Name	Reset	R/W	Description
7	PTEST[7:0]	127 (0x7F)	R/W	Writing 0xBF to this register makes the on-chip temperature sensor available in the IDLE state. The default 0x7F value should then be written back before leaving the IDLE state. Other use of this register is for test only.

**0x2B: AGCTEST – AGC test**

Bit	Field Name	Reset	R/W	Description
7:0	AGCTEST[7:0]	63 (0x3F)	R/W	For test only. Do not write to this register.

**0x2C: TEST2 – Various test settings**

Bit	Field Name	Reset	R/W	Description
7:0	TEST2[7:0]	152 (0x88)	R/W	The value to use in this register is given by the SmartRF® Studio software.

**0x2D: TEST1 – Various test settings**

Bit	Field Name	Reset	R/W	Description
7:0	TEST1[7:0]	49 (0x31)	R/W	The value to use in this register is given by the SmartRF® Studio software.

**0x2E: TEST0 – Various test settings**

Bit	Field Name	Reset	R/W	Description
7:0	TEST0[7:0]	11 (0x0B)	R/W	The value to use in this register is given by the SmartRF® Studio software.

**31.3 Status register details**

**0x30 (0xF0): PARTNUM – Chip ID**

Bit	Field Name	Reset	R/W	Description
7:0	PARTNUM[7:0]	128 (0x80)	R	Chip part number

**0x31 (0xF1): VERSION – Chip ID**

Bit	Field Name	Reset	R/W	Description
7:0	VERSION[7:0]	3 (0x03)	R	Chip version number.

**0x32 (0xF2): FREQEST – Frequency Offset Estimate from demodulator**

Bit	Field Name	Reset	R/W	Description
7:0	FREQOFF_EST		R	The estimated frequency offset (two's complement) of the carrier. Resolution is $F_{XTAL}/2^{14}$ (1.59 - 1.65 kHz); range is $\pm 202$ kHz to $\pm 210$ kHz, dependent of XTAL frequency.  Frequency offset compensation is only supported for FSK and MSK modulation. This register will read 0 when using OOK modulation.

**0x33 (0xF3): LQI – Demodulator estimate for Link Quality**

Bit	Field Name	Reset	R/W	Description
7	Reserved			
6:0	LQI_EST[6:0]		R	The Link Quality Indicator estimates how easily a received signal can be demodulated. Calculated over the 64 symbols following the sync word (first 8 packet bytes for 2-ary modulation).

**0x34 (0xF4): RSSI – Received signal strength indication**

Bit	Field Name	Reset	R/W	Description
7:0	RSSI		R	Received signal strength indicator

**0x35 (0xF5): MARCSTATE – Main Radio Control State Machine state**

Bit	Field Name	Reset	R/W	Description																																																																								
7:5	Reserved		R0																																																																									
4:0	MARC_STATE[4:0]		R	<p>Main Radio Control FSM State</p> <table border="1"> <thead> <tr> <th>Value</th> <th>State name</th> <th>State (Figure 11, page 34)</th> </tr> </thead> <tbody> <tr><td>0 (0x00)</td><td>SLEEP</td><td>SLEEP</td></tr> <tr><td>1 (0x01)</td><td>IDLE</td><td>IDLE</td></tr> <tr><td>2 (0x02)</td><td>XOFF</td><td>XOFF</td></tr> <tr><td>3 (0x03)</td><td>VCOON_MC</td><td>MANCAL</td></tr> <tr><td>4 (0x04)</td><td>REGON_MC</td><td>MANCAL</td></tr> <tr><td>5 (0x05)</td><td>MANCAL</td><td>MANCAL</td></tr> <tr><td>6 (0x06)</td><td>VCOON</td><td>FS_WAKEUP</td></tr> <tr><td>7 (0x07)</td><td>REGON</td><td>FS_WAKEUP</td></tr> <tr><td>8 (0x08)</td><td>STARTCAL</td><td>CALIBRATE</td></tr> <tr><td>9 (0x09)</td><td>BWBOOST</td><td>SETTLING</td></tr> <tr><td>10 (0x0A)</td><td>FS_LOCK</td><td>SETTLING</td></tr> <tr><td>11 (0x0B)</td><td>IFADCON</td><td>SETTLING</td></tr> <tr><td>12 (0x0C)</td><td>ENDCAL</td><td>CALIBRATE</td></tr> <tr><td>13 (0x0D)</td><td>RX</td><td>RX</td></tr> <tr><td>14 (0x0E)</td><td>RX_END</td><td>RX</td></tr> <tr><td>15 (0x0F)</td><td>RX_RST</td><td>RX</td></tr> <tr><td>16 (0x10)</td><td>TXRX_SWITCH</td><td>TXRX_SETTLING</td></tr> <tr><td>17 (0x11)</td><td>RX_OVERFLOW</td><td>RX_OVERFLOW</td></tr> <tr><td>18 (0x12)</td><td>FSTXON</td><td>FSTXON</td></tr> <tr><td>19 (0x13)</td><td>TX</td><td>TX</td></tr> <tr><td>20 (0x14)</td><td>TX_END</td><td>TX</td></tr> <tr><td>21 (0x15)</td><td>RXTX_SWITCH</td><td>RXTX_SETTLING</td></tr> <tr><td>22 (0x16)</td><td>TX_UNDERFLOW</td><td>TX_UNDERFLOW</td></tr> </tbody> </table> <p>Note: it is not possible to read back the SLEEP or XOFF state numbers because setting CS<sub>N</sub> low will make the chip enter the IDLE mode from the SLEEP or XOFF states.</p>	Value	State name	State (Figure 11, page 34)	0 (0x00)	SLEEP	SLEEP	1 (0x01)	IDLE	IDLE	2 (0x02)	XOFF	XOFF	3 (0x03)	VCOON_MC	MANCAL	4 (0x04)	REGON_MC	MANCAL	5 (0x05)	MANCAL	MANCAL	6 (0x06)	VCOON	FS_WAKEUP	7 (0x07)	REGON	FS_WAKEUP	8 (0x08)	STARTCAL	CALIBRATE	9 (0x09)	BWBOOST	SETTLING	10 (0x0A)	FS_LOCK	SETTLING	11 (0x0B)	IFADCON	SETTLING	12 (0x0C)	ENDCAL	CALIBRATE	13 (0x0D)	RX	RX	14 (0x0E)	RX_END	RX	15 (0x0F)	RX_RST	RX	16 (0x10)	TXRX_SWITCH	TXRX_SETTLING	17 (0x11)	RX_OVERFLOW	RX_OVERFLOW	18 (0x12)	FSTXON	FSTXON	19 (0x13)	TX	TX	20 (0x14)	TX_END	TX	21 (0x15)	RXTX_SWITCH	RXTX_SETTLING	22 (0x16)	TX_UNDERFLOW	TX_UNDERFLOW
Value	State name	State (Figure 11, page 34)																																																																										
0 (0x00)	SLEEP	SLEEP																																																																										
1 (0x01)	IDLE	IDLE																																																																										
2 (0x02)	XOFF	XOFF																																																																										
3 (0x03)	VCOON_MC	MANCAL																																																																										
4 (0x04)	REGON_MC	MANCAL																																																																										
5 (0x05)	MANCAL	MANCAL																																																																										
6 (0x06)	VCOON	FS_WAKEUP																																																																										
7 (0x07)	REGON	FS_WAKEUP																																																																										
8 (0x08)	STARTCAL	CALIBRATE																																																																										
9 (0x09)	BWBOOST	SETTLING																																																																										
10 (0x0A)	FS_LOCK	SETTLING																																																																										
11 (0x0B)	IFADCON	SETTLING																																																																										
12 (0x0C)	ENDCAL	CALIBRATE																																																																										
13 (0x0D)	RX	RX																																																																										
14 (0x0E)	RX_END	RX																																																																										
15 (0x0F)	RX_RST	RX																																																																										
16 (0x10)	TXRX_SWITCH	TXRX_SETTLING																																																																										
17 (0x11)	RX_OVERFLOW	RX_OVERFLOW																																																																										
18 (0x12)	FSTXON	FSTXON																																																																										
19 (0x13)	TX	TX																																																																										
20 (0x14)	TX_END	TX																																																																										
21 (0x15)	RXTX_SWITCH	RXTX_SETTLING																																																																										
22 (0x16)	TX_UNDERFLOW	TX_UNDERFLOW																																																																										



**0x36 (0xF6): WORTIME1 – High byte of WOR time**

Bit	Field Name	Reset	R/W	Description
7:0	TIME[15:8]		R	High byte of timer value in WOR module

**0x37 (0xF7): WORTIME0 – Low byte of WOR time**

Bit	Field Name	Reset	R/W	Description
7:0	TIME[7:0]		R	Low byte of timer value in WOR module

**0x38 (0xF8): PKTSTATUS – Current GDOx status and packet status**

Bit	Field Name	Reset	R/W	Description
7	Reserved			
6	CS		R	Carrier sense
5	PQT_REACHED		R	Preamble Quality reached
4	CCA		R	Clear channel assessment
3	SFD		R	Sync word found
2	GDO2		R	Current GDO2 value. Note: the reading gives the non-inverted value irrespective what IOCFG2.GDO2_INV is programmed to.
1	Reserved			
0	GDO0		R	Current GDO0 value. Note: the reading gives the non-inverted value irrespective what IOCFG0.GDO0_INV is programmed to.

**0x39 (0xF9): VCO\_VC\_DAC – Current setting from PLL calibration module**

Bit	Field Name	Reset	R/W	Description
7:0	VCO_VC_DAC[7:0]		R	Status register for test only.

**0x3A (0xFA): TXBYTES – Underflow and number of bytes**

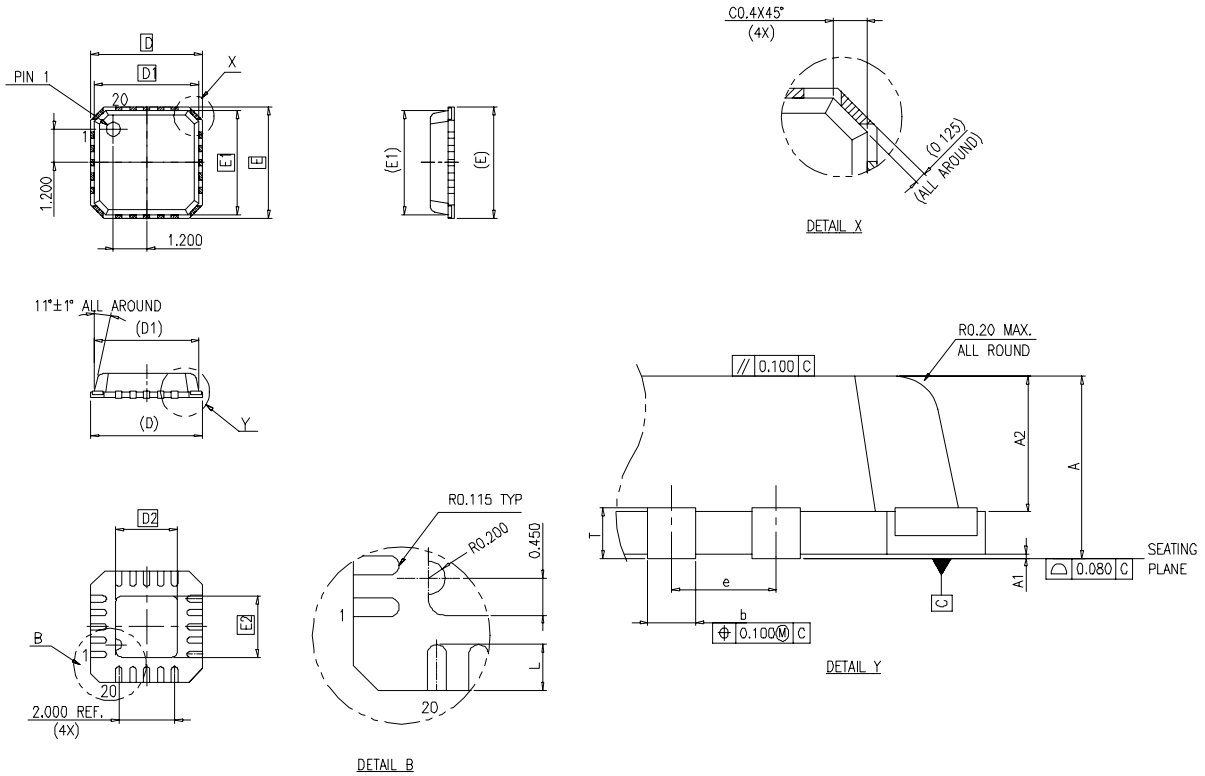
Bit	Field Name	Reset	R/W	Description
7	TXFIFO_UNDERFLOW		R	
6:0	NUM_TXBYTES		R	Number of bytes in TX FIFO

**0x3B (0xFB): RXBYTES – Overflow and number of bytes**

Bit	Field Name	Reset	R/W	Description
7	RXFIFO_OVERFLOW		R	
6:0	NUM_RXBYTES		R	Number of bytes in RX FIFO

### 32 Package Description (QLP 20)

All dimensions are in millimetres, angles in degrees. NOTE: The **CC2500** is available in RoHS lead-free package only.

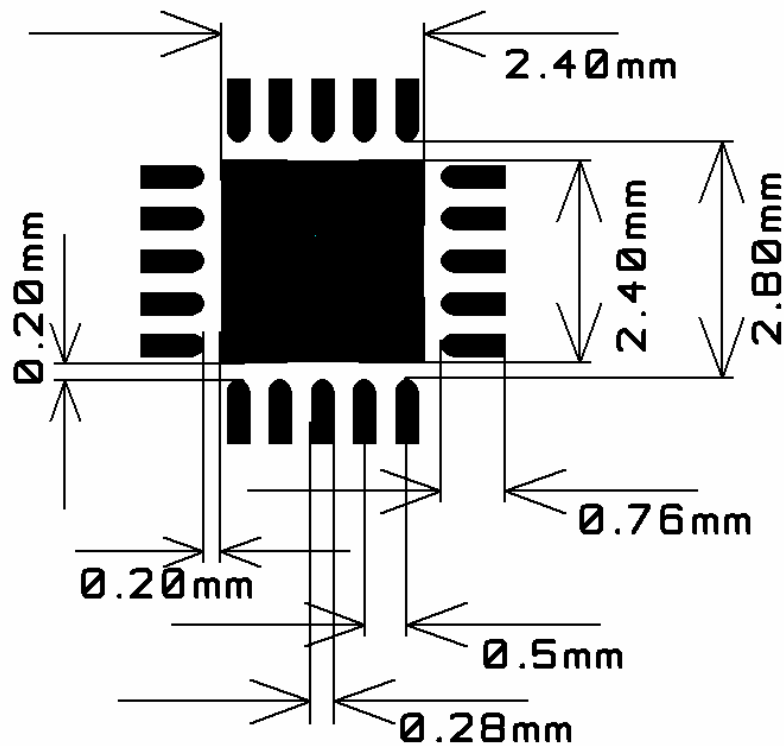


**Figure 24: Package dimensions drawing**

Package type		A	A1	A2	D	D1	D2	E	E1	E2	L	T	b	e
QLP 20 (4x4)	Min	0.75	0.005	0.55	3.90	3.65		3.90	3.65		0.45	0.190	0.18	
	Typ.	0.85	0.025	0.65	4.00	3.75	2.40	4.00	3.75	2.40	0.55		0.23	0.50
	Max	0.95	0.045	0.75	4.10	3.85		4.10	3.85		0.65	0.245	0.30	

**Table 38: Package dimensions**

**32.1 Recommended PCB layout for package (QLP 20)**



**Figure 25: Recommended PCB layout for QLP 20 package**

Note: The figure is an illustration only and not to scale. There are five 14 mil diameter via holes distributed symmetrically in the ground pad under the package. See also the **CC2500** EM reference design.

**32.2 Package thermal properties**

Thermal resistance	
Air velocity [m/s]	0
Rth,j-a [K/W]	40.4

**Table 39: Thermal properties of QLP 20 package**

**32.3 Soldering information**

The recommendations for lead-free reflow in IPC/JEDEC J-STD-020C should be followed.

**32.4 Tray specification**

**CC2500** can be delivered in standard QLP 4x4mm shipping trays.

Tray Specification				
Package	Tray Width	Tray Height	Tray Length	Units per Tray
QLP 20	135.9 mm	7.62 mm	322.6 mm	490

**Table 40: Tray specification**

### 32.5 Carrier tape and reel specification

Carrier tape and reel is in accordance with EIA Specification 481.

Tape and Reel Specification					
Package	Tape Width	Component Pitch	Hole Pitch	Reel Diameter	Units per Reel
QLP 20	12 mm	8 mm	4 mm	13 inches	2500

**Table 41: Carrier tape and reel specification**

## 33 Ordering Information

Ordering part number	Description	Minimum Order Quantity (MOQ)
1167	<b>CC2500</b> - RTY1 QLP20 RoHS Pb-free 490/tray	490 (tray)
1190	<b>CC2500</b> - RTR1 QLP20 RoHS Pb-free 2500/T&R	2500 (tape and reel)
1192	<b>CC2500</b> SK Sample kit 5pcs.	1
10069	<b>CC2500</b> <b>CC2550</b> DK Development Kit	1

**Table 42: Ordering information**

## 34 General Information

### 34.1 Document History

Revision	Date	Description/Changes
1.1	2005-10-20	MDMCFG2[7] used. 26-27 MHz crystal range. Chapter 15: description of the 2 optional append bytes. Added matching information. Added information about using a reference signal instead of a crystal. CRC can only be checked by append bytes or CRC_AUTOFLUSH. Added equation for calculating RSSI in dBm. Selectivity performance graphs added.
1.0	2005-01-24	First preliminary release.

**Table 43: Document history**

### 34.2 Product Status Definitions

Data Sheet Identification	Product Status	Definition
Advance Information	Planned or Under Development	This data sheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	Engineering Samples and First Production	This data sheet contains preliminary data, and supplementary data will be published at a later date. Chipcon reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.
No Identification Noted	Full Production	This data sheet contains the final specifications. Chipcon reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.
Obsolete	Not In Production	This data sheet contains specifications on a product that has been discontinued by Chipcon. The data sheet is printed for reference information only.

**Table 44: Product status definitions**

### 34.3 Disclaimer

Chipcon AS believes the information contained herein is correct and accurate at the time of this printing. However, Chipcon AS reserves the right to make changes to this product without notice. Chipcon AS does not assume any responsibility for the use of the described product; neither does it convey any license under its patent rights, or the rights of others. The latest updates are available at the Chipcon website or by contacting Chipcon directly.

As far as possible, major changes of product specifications and functionality, will be stated in product specific Errata Notes published at the Chipcon website. Customers are encouraged to sign up to the Developers Newsletter for the most recent updates on products and support tools.

When a product is discontinued this will be done according to Chipcon's procedure for obsolete products as described in Chipcon's Quality Manual. This includes informing about last-time-buy options. The Quality Manual can be downloaded from Chipcon's website.

Compliance with regulations is dependent on complete system performance. It is the customer's responsibility to ensure that the system complies with regulations.

### 34.4 Trademarks

*SmartRF*<sup>®</sup> is a registered trademark of Chipcon AS. SmartRF<sup>®</sup> is Chipcon's RF technology platform with RF library cells, modules and design expertise. Based on SmartRF<sup>®</sup> technology Chipcon develops standard component RF circuits as well as full custom ASICs based on customer requirements and this technology.

All other trademarks, registered trademarks and product names are the sole property of their respective owners.

### 34.5 Life Support Policy

This Chipcon product is not designed for use in life support appliances, devices, or other systems where malfunction can reasonably be expected to result in significant personal injury to the user, or as a critical component in any life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness. Chipcon AS customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Chipcon AS for any damages resulting from any improper use or sale.