Si4362-C

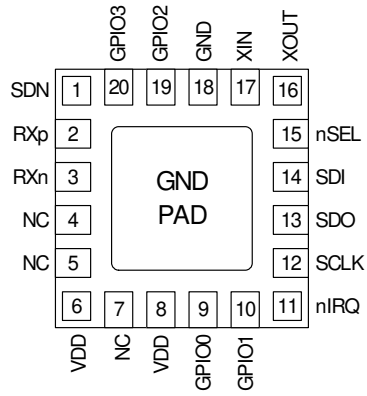
HIGH-PERFORMANCE, LOW-CURRENT RECEIVER

Features

- Frequency range = 142–1050 MHz
- Receive sensitivity = –129 dBm
- Modulation
  - (G)FSK, 4(G)FSK, (G)MSK
  - OOK
- Low active power consumption
  - 10/13 mA RX
- Ultra low current powerdown modes
  - 30 nA shutdown, 40 nA standby
- Preamble Sense Mode
  - 6 mA average Rx current at 1.2 kbs
- Fast preamble detection
  - 1 byte preamble detection
- Data rate = 100 bps to 1 Mbps
- Fast wake and hop times
- Power supply = 1.8 to 3.8 V
- Excellent selectivity performance
  - 69 dB adjacent channel
  - 79 dB blocking at 1 MHz
- Antenna diversity and T/R switch control
- Highly configurable packet handler
- RX 129 byte FIFO
- Auto frequency control (AFC)
- Automatic gain control (AGC)
- Low BOM
- Low battery detector
- Temperature sensor
- 20-Pin QFN package
- IEEE 802.15.4g and WMBus compliant
- Suitable FCC Part 90 Mask D, FCC part 15.247,15,231, 15,249, ARIB T-108, T-96, T-67,RCR STD-30, China regulatory, ETSI EN 300 220, ETSI Category 1 Operation



Pin Assignments



Patents pending

Applications

- Smart metering (802.15.4g and MBus)
- Remote control
- Home security and alarm
- Telemetry
- Garage and gate openers
- Remote keyless entry
- Home automation
- Industrial control
- Sensor networks
- Health monitors
- Electronic shelf labels

Description

Silicon Labs Si4362 devices are high-performance, low-current receivers covering the sub-GHz frequency bands from 142 to 1050 MHz. The radios are part of the EZRadioPRO® family, which includes a complete line of transmitters, receivers, and transceivers covering a wide range of applications. All parts offer outstanding sensitivity of –129 dBm while achieving extremely low active and standby current consumption. The 69 dB adjacent channel selectivity with 12.5 kHz channel spacing ensures robust receive operation in harsh RF conditions, which is particularly important for narrowband operation. RX current of 10 mA coupled with extremely low standby current and fast wake times ensure extended battery life in the most demanding applications.

Copyright © 2014 by Silicon Laboratories

This information applies to a product under development. Its characteristics and specifications are subject to change without notice.

Functional Block Diagram

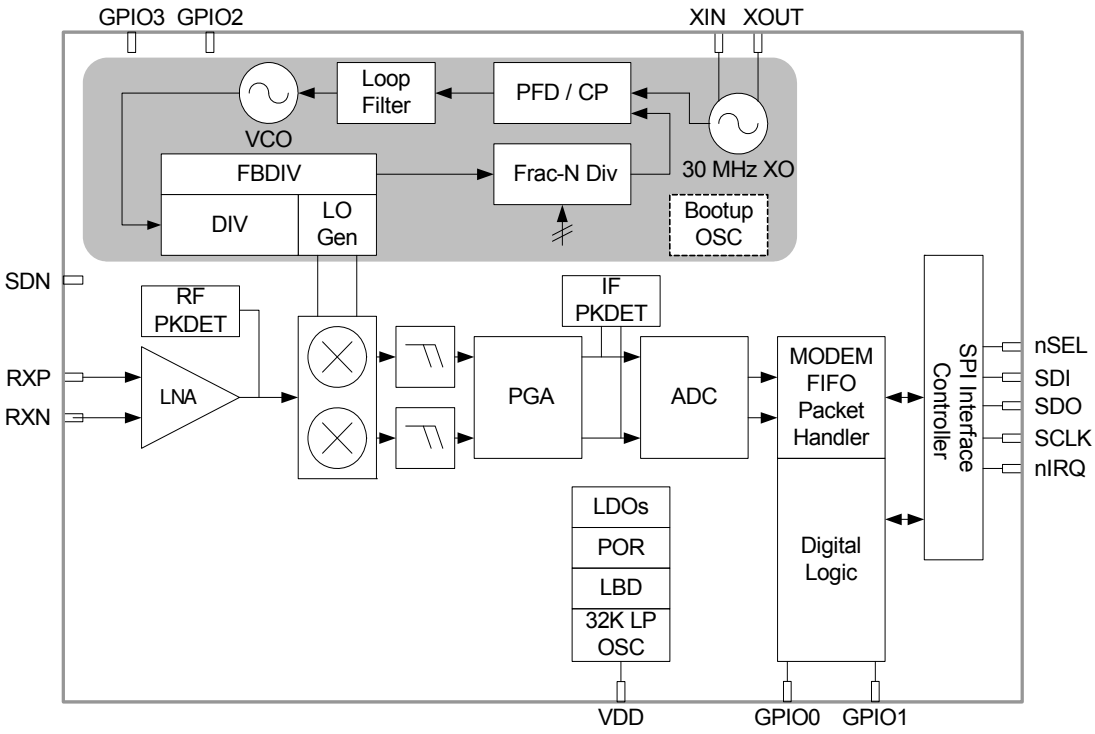


TABLE OF CONTENTS

Section **Page**

1. Electrical Specifications 4
2. Functional Description 12
3. Controller Interface 13
3.1. Serial Peripheral Interface (SPI) 13
3.2. Fast Response Registers 15
3.3. Operating Modes and Timing 15
3.4. Application Programming Interface (API) 18
3.5. Interrupts 18
3.6. GPIO 19
4. Modulation and Hardware Configuration Options 20
4.1. Hardware Configuration Options 20
4.2. Preamble Length 21
5. Internal Functional Blocks 23
5.1. RX Chain 23
5.2. RX Modem 24
5.3. Synthesizer 26
5.4. Crystal Oscillator 28
6. Data Handling and Packet Handler 29
6.1. RX FIFOs 29
6.2. Packet Handler 30
7. RX Modem Configuration 31
8. Auxiliary Blocks 32
8.1. Wake-up Timer and 32 kHz Clock Source 32
8.2. Low Duty Cycle Mode (Auto RX Wake-Up) 33
8.3. Temperature, Battery Voltage, and Auxiliary ADC 33
8.4. Low Battery Detector 33
8.5. Antenna Diversity 34
8.6. Preamble Sense Mode 34
9. Wireless MBUS support 35
10. ETSI EN300 220 Category 1 35
11. Pin Descriptions: Si4362-C 36
12. Ordering Information 38
13. Package Outline: Si4362 39
14. PCB Land Pattern: Si4362 41
15. Top Marking 43
15.1. Si4362 Top Marking 43
15.2. Top Marking Explanation 43
Contact Information 44

1. Electrical Specifications

Table 1. DC Characteristics*

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Supply Voltage Range	V_{DD}		1.8	3.3	3.8	V
Power Saving Modes	$I_{Shutdown}$	RC Oscillator, Main Digital Regulator, and Low Power Digital Regulator OFF	—	30	—	nA
	$I_{Standby}$	Register values maintained and RC oscillator/WUT OFF	—	40	—	nA
	$I_{SleepRC}$	RC Oscillator/WUT ON and all register values maintained, and all other blocks OFF	—	740	—	nA
	$I_{SleepXO}$	Sleep current using an external 32 kHz crystal.	—	1.7	—	µA
	$I_{Sensor-LBD}$	Low battery detector ON, register values maintained, and all other blocks OFF	—	1	—	µA
	I_{Ready}	Crystal Oscillator and Main Digital Regulator ON, all other blocks OFF	—	1.8	—	mA
Preamble Sense Mode Current	I_{psm}	Duty cycling during preamble search, 1.2 kbps, 4 byte preamble	—	6	—	mA
TUNE Mode Current	I_{Tune_RX}	RX Tune, High Performance Mode	—	7.6	—	mA
RX Mode Current	I_{RXH}	High Performance Mode (measured at 915 MHz and 40 kbps)	—	13.7	—	mA
	I_{RXL}	Low Power Mode (measured at 315 MHz and 40 kbps)	—	10.9	—	mA

***Note:** All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at VDD = 3.3 V and 25 °C unless otherwise stated.

Table 2. Synthesizer AC Electrical Characteristics*

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Synthesizer Frequency Range (Si4362)	F_{SYN}		142	—	175	MHz
			284	—	350	MHz
			350	—	525	MHz
			850	—	1050	MHz
Synthesizer Frequency Resolution	$F_{RES-960}$	850–1050 MHz	—	28.6	—	Hz
	$F_{RES-525}$	420–525 MHz	—	14.3	—	Hz
	$F_{RES-420}$	350–420 MHz	—	11.4	—	Hz
	$F_{RES-350}$	284–350 MHz	—	9.5	—	Hz
	$F_{RES-175}$	142–175 MHz	—	4.7	—	Hz
Synthesizer Settling Time	t_{LOCK}	Measured from exiting Ready mode with XOSC running to any frequency. Including VCO Calibration.	—	50	—	µs

***Note:** All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at VDD = 3.3 V and 25 °C unless otherwise stated.

Table 3. Receiver AC Electrical Characteristics[1,2]

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
RX Frequency Range	F_{RX}		850	—	1050	MHz
			350	—	525	MHz
			284	—	350	MHz
			142	—	175	MHz
RX Sensitivity 169 MHz[3]	$P_{RX_0.5}$	(BER < 0.1%) (500 bps, GFSK, BT = 0.5, Δf = ±250Hz)	—	–129	—	dBm
	P_{RX_40}	(BER < 0.1%) (40 kbps, GFSK, BT = 0.5, Δf = ±20 kHz)	—	–110	—	dBm
	P_{RX_100}	(BER < 0.1%) (100 kbps, GFSK, BT = 0.5, Δf = ±50 kHz)	—	–106	—	dBm
	P_{RX_500}	(BER < 0.1%) (500 kbps, GFSK, BT = 0.5, Δf = ±250 kHz)	—	–98	—	dBm
	$P_{RX_9.6}$	(PER 1%) (9.6 kbps, 4GFSK, BT = 0.5, Δf = ±2.4 kHz)	—	–110	—	dBm
	P_{RX_1M}	(PER 1%) (1 Mbps, 4GFSK, BT = 0.5, inner deviation = 83.3 kHz)	—	–89	—	dBm
	P_{RX_OOK}	(BER < 0.1%, 4.8 kbps, 350 kHz BW, OOK, PN15 data)	—	–110	—	dBm
		(BER < 0.1%, 40 kbps, 350 kHz BW, OOK, PN15 data)	—	–103	—	dBm
		(BER < 0.1%, 120 kbps, 350 kHz BW, OOK, PN15 data)	—	–97	—	dBm

Notes:

1. All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at V_{DD} = 3.3 V and 25 °C unless otherwise stated.
2. For PER tests, 48 preamble symbols, 4 byte sync word, 10 byte payload and CRC-32 was used.
3. Measured over 50000 bits using PN9 data sequence and data and clock on GPIOs. Sensitivity is expected to be better if reading data from packet handler FIFO especially at higher data rates.

Table 3. Receiver AC Electrical Characteristics[1,2] (Continued)

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
RX Sensitivity 915/868 MHz[3]	$P_{RX_0.5}$	(BER < 0.1%) (500 bps, GFSK, BT = 0.5, Δf = ±250Hz)	—	–127	—	dBm
	P_{RX_40}	(BER < 0.1%) (40 kbps, GFSK, BT = 0.5, Δf = ±20 kHz)	—	–109	—	dBm
	P_{RX_100}	(BER < 0.1%) (100 kbps, GFSK, BT = 0.5, Δf = ±50 kHz)	—	–104	—	dBm
	P_{RX_500}	(BER < 0.1%) (500 kbps, GFSK, BT = 0.5, Δf = ±250 kHz)	—	–97	—	dBm
	$P_{RX_9.6}$	(PER 1%) (9.6 kbps, 4GFSK, BT = 0.5, Δf = ±2.4 kHz)	—	–109	—	dBm
	P_{RX_1M}	(PER 1%) (1 Mbps, 4GFSK, BT = 0.5, inner deviation = 83.3 kHz)	—	–88	—	dBm
	P_{RX_OOK}	(BER < 0.1%, 4.8 kbps, 350 kHz BW, OOK, PN15 data)	—	–108	—	dBm
		(BER < 0.1%, 40 kbps, 350 kHz BW, OOK, PN15 data)	—	–101	—	dBm
		(BER < 0.1%, 120 kbps, 350 kHz BW, OOK, PN15 data)	—	–96	—	dBm
RX Channel Bandwidth	BW		1.1	—	850	kHz
RSSI Resolution	RES_{RSSI}	Valid from –110 dBm to –90 dBm	—	±0.5	—	dB
±1-Ch Offset Selectivity, 169 MHz[3]	$C/I_{1\text{-}CH}$	Desired Ref Signal 3 dB above sensitivity, BER < 0.1%. Interferer is CW, and desired is modulated with 2.4 kbps ΔF = 1.2 kHz GFSK with BT = 0.5, RX channel BW = 4.8 kHz, channel spacing = 12.5 kHz	—	–69	—	dB
±1-Ch Offset Selectivity, 450 MHz[3]	$C/I_{1\text{-}CH}$		—	–60	—	dB
±1-Ch Offset Selectivity, 868 / 915 MHz[3]	$C/I_{1\text{-}CH}$		—	–55	—	dB
Blocking 1 MHz Offset	$1M_{BLOCK}$	Desired Ref Signal 3 dB above sensitivity, BER = 0.1%. Interferer is CW, and desired is modulated with 2.4 kbps, ΔF = 1.2 kHz GFSK with BT = 0.5, RX channel BW = 4.8 kHz	—	–79	—	dB
Blocking 8 MHz Offset	$8M_{BLOCK}$		—	–86	—	dB

Notes:

1. All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at V_{DD} = 3.3 V and 25 °C unless otherwise stated.
2. For PER tests, 48 preamble symbols, 4 byte sync word, 10 byte payload and CRC-32 was used.
3. Measured over 50000 bits using PN9 data sequence and data and clock on GPIOs. Sensitivity is expected to be better if reading data from packet handler FIFO especially at higher data rates.

Table 3. Receiver AC Electrical Characteristics[1,2] (Continued)

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Image Rejection (IF = 468.75 kHz)	Im_{REJ}	No image rejection calibration. Rejection at the image frequency. RF = 460 MHz	—	40	—	dB
		With image rejection calibration in Si4362. Rejection at the image frequency. RF = 460 MHz	—	55	—	dB
		No image rejection calibration. Rejection at the image frequency. RF = 915 MHz	—	45	—	dB
		With image rejection calibration in Si4362. Rejection at the image frequency. RF = 915 MHz	—	52	—	dB
		No image rejection calibration. Rejection at the image frequency. RF = 169 MHz	—	45	—	dB
		With image rejection calibration in Si4362. Rejection at the image frequency. RF = 169 MHz	—	60	—	dB

Notes:

1. All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at V_{DD} = 3.3 V and 25 °C unless otherwise stated.
2. For PER tests, 48 preamble symbols, 4 byte sync word, 10 byte payload and CRC-32 was used.
3. Measured over 50000 bits using PN9 data sequence and data and clock on GPIOs. Sensitivity is expected to be better if reading data from packet handler FIFO especially at higher data rates.

Table 4. Auxiliary Block Specifications[1]

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Temperature Sensor Sensitivity[1]	TS_S		—	4.5	—	ADC Codes/ °C
Low Battery Detector Resolution	LBD_{RES}		—	50	—	mV
Microcontroller Clock Output Frequency Range[2]	F_{MC}	Configurable to Fxtal or Fxtal divided by 2, 3, 7.5, 10, 15, or 30 where Fxtal is the reference XTAL frequency. In addition, 32.768 kHz is also supported.	32.768K	—	Fxtal	Hz
Temperature Sensor Conversion[1]	$TEMP_{CT}$	Programmable setting	—	3	—	ms
XTAL Range[3]	$XTAL_{Range}$		25	—	32	MHz
30 MHz XTAL Start-Up Time	t_{30M}	Using XTAL and board layout in reference design. Start-up time will vary with XTAL type and board layout.	—	300	—	µs
30 MHz XTAL Cap Resolution[1]	$30M_{RES}$		—	70	—	fF
32 kHz XTAL Start-Up Time[1]	t_{32k}		—	2	—	sec
32 kHz Accuracy using Internal RC Oscillator[1]	$32KRC_{RES}$		—	2500	—	ppm
POR Reset Time	t_{POR}		—	—	6	ms

Notes:
1. All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at V_{DD} = 3.3 V and 25 °C unless otherwise stated.
2. Microcontroller clock frequency tested in production at 1 MHz, 30 MHz, 32 MHz, and 32.768 kHz. Other frequencies tested in bench characterization.
3. XTAL Range tested in production using an external clock source (similar to using a TCXO).

Table 5. Digital IO Specifications (GPIO_x, SCLK, SDO, SDI, nSEL, nIRQ, SDN)[1]

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Rise Time[2,3]	T_{RISE}	0.1 x V_{DD} to 0.9 x V_{DD}, C_L = 10 pF, DRV<1:0> = LL	—	2.3	—	ns
Fall Time[3,4]	T_{FALL}	0.9 x V_{DD} to 0.1 x V_{DD}, C_L = 10 pF, DRV<1:0> = LL	—	2	—	ns
Input Capacitance	C_{IN}		—	2	—	pF
Logic High Level Input Voltage	V_{IH}		V_{DD} x 0.7	—	—	V
Logic Low Level Input Voltage	V_{IL}		—	—	V_{DD} x 0.3	V
Input Current	I_{IN}	0<V_{IN}< V_{DD}	–1	—	1	µA
Input Current If Pullup is Activated	I_{INP}	V_{IL} = 0 V	1	—	4	µA
Drive Strength for Output Low Level	I_{OmaxLL}	DRV[1:0] = LL[3]	—	6.66	—	mA
	I_{OmaxLH}	DRV[1:0] = LH[3]	—	5.03	—	mA
	I_{OmaxHL}	DRV[1:0] = HL[3]	—	3.16	—	mA
	I_{OmaxHH}	DRV[1:0] = HH[3]	—	1.13	—	mA
Drive Strength for Output High Level	I_{OmaxLL}	DRV[1:0] = LL[3]	—	5.75	—	mA
	I_{OmaxLH}	DRV[1:0] = LH[3]	—	4.37	—	mA
	I_{OmaxHL}	DRV[1:0] = HL[3]	—	2.73	—	mA
	I_{OmaxHH}	DRV[1:0] = HH[3]	—	0.96	—	mA
Drive Strength for Output High Level for GPIO0	I_{OmaxLL}	DRV[1:0] = LL[3]	—	2.53	—	mA
	I_{OmaxLH}	DRV[1:0] = LH[3]	—	2.21	—	mA
	I_{OmaxHL}	DRV[1:0] = HL[3]	—	1.70	—	mA
	I_{OmaxHH}	DRV[1:0] = HH[3]	—	0.80	—	mA
Logic High Level Output Voltage	V_{OH}	DRV[1:0] = HL	V_{DD} x 0.8	—	—	V
Logic Low Level Output Voltage	V_{OL}	DRV[1:0] = HL	—	—	V_{DD} x 0.2	V

Notes:

1. All minimum and maximum values are guaranteed across the recommended operating conditions of supply voltage and from –40 to +85 °C unless otherwise stated. All typical values apply at VDD = 3.3 V and 25 °C unless otherwise stated.
2. 6.7 ns is typical for GPIO0 rise time.
3. Assuming VDD = 3.3 V, drive strength is specified at Voh (min) = 2.64 V and Vol(max) = 0.66 V at room temperature.
4. 2.4 ns is typical for GPIO0 fall time.

Table 6. Thermal Characteristics

Parameter	Symbol	Value	Unit
Operating Ambient Temperature Range	T_A	–40 to +85	°C
Thermal Impedance Junction to Ambient	θ_{JA}	25	°C/W
Junction Temperature Maximum Value	T_j	+105	°C
Storage Temperature Range	T_{STG}	–55 to +150	°C

Table 7. Absolute Maximum Ratings

Parameter	Value	Unit
V_{DD} to GND	–0.3, +3.8	V
Voltage on Digital Inputs	–0.3, V_{DD} + 0.3	V
Voltage on Analog Inputs	–0.7, V_{DD} + 0.3	V
RX Input Power	+10	dBm

Note: Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. These are stress ratings only and functional operation of the device at or beyond these ratings in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Power Amplifier may be damaged if switched on without proper load or termination connected. Caution: ESD sensitive device.

2. Functional Description

The Si4362 is a high performance, low current, wireless ISM receiver that covers major sub-GHz bands. The wide operating voltage range of 1.8–3.8 V and low current consumption make the Si4362 an ideal solution for battery powered applications. The device uses a single-conversion mixer to downconvert the 2/4-level FSK/GFSK or OOK modulated receive signal to a low IF frequency. Following a programmable gain amplifier (PGA) the signal is converted to the digital domain by a high performance ΔΣ ADC allowing filtering, demodulation, slicing, and packet handling to be performed in the built-in DSP increasing the receiver's performance and flexibility versus analog based architectures. The demodulated signal is output to the system MCU through a programmable GPIO or via the standard SPI bus by reading the 64-byte RX FIFO.

A single high precision local oscillator (LO) is used for receive mode. The LO is generated by an integrated VCO and ΔΣ Fractional-N PLL synthesizer. The synthesizer is designed to support configurable data rates from 100 bps to 1 Mbps. The Si4362 operates in the frequency bands of 142–175, 283–350, 350–525, and 850–1050 MHz with a maximum frequency accuracy step size of 28.6 Hz.

The Si4362 supports frequency hopping and antenna diversity switch control to extend the link range and improve performance. Built-in antenna diversity and support for frequency hopping can be used to further extend range and enhance performance. Antenna diversity is completely integrated into the Si4362 and can improve the system link budget by 8–10 dB, resulting in substantial range increases under adverse environmental conditions. A highly configurable packet handler allows for autonomous encoding/decoding of nearly any packet structure. Additional system features, such as an automatic wake-up timer, low battery detector, 64 byte RX FIFOs, and preamble detection, reduce overall current consumption and allows for the use of lower-cost system MCUs. An integrated temperature sensor, power-on-reset (POR), and GPIOs further reduce overall system cost and size. The Si4362 is designed to work with an MCU, crystal, and a few passive components to create a very low-cost system.

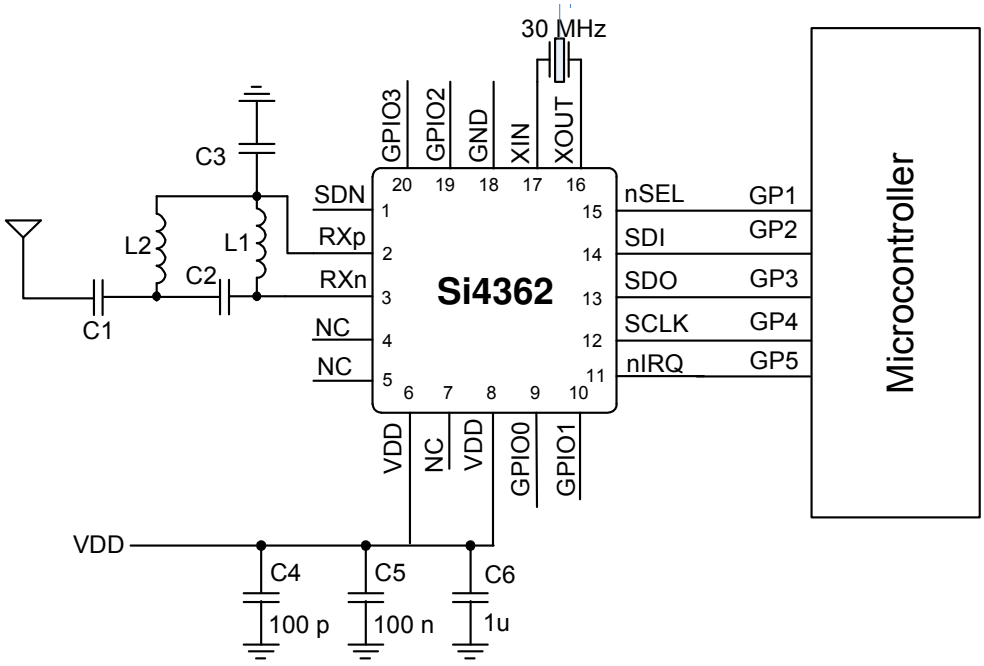


Figure 1. Si4362 Application Example

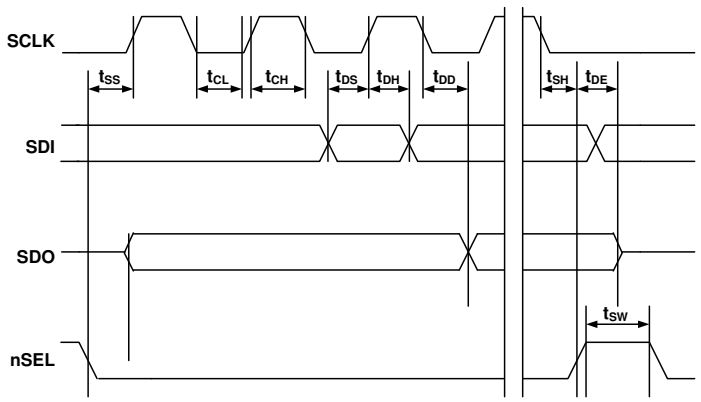
3. Controller Interface

3.1. Serial Peripheral Interface (SPI)

The Si4362 communicates with the host MCU over a standard 4-wire serial peripheral interface (SPI): SCLK, SDI, SDO, and nSEL. The SPI interface is designed to operate at a maximum of 10 MHz. The SPI timing parameters are demonstrated in Table 8. The host MCU writes data over the SDI pin and can read data from the device on the SDO output pin. Figure 2 demonstrates an SPI write command. The nSEL pin should go low to initiate the SPI command. The first byte of SDI data will be one of the firmware commands followed by n bytes of parameter data which will be variable depending on the specific command. The rising edges of SCLK should be aligned with the center of the SDI data.

Table 8. Serial Interface Timing Parameters

Symbol	Parameter	Min (ns)	Max (ns)	Diagram
t_{CH}	Clock high time	40		
t_{CL}	Clock low time	40		
t_{DS}	Data setup time	20		
t_{DH}	Data hold time	20		
t_{DD}	Output data delay time		43	
t_{DE}	Output disable time		45	
t_{SS}	Select setup time	20		
t_{SH}	Select hold time	50		
t_{SW}	Select high period	80		



***Note:** CL = 10 pF; VDD = 1.8 V; SDO Drive strength setting = 10.

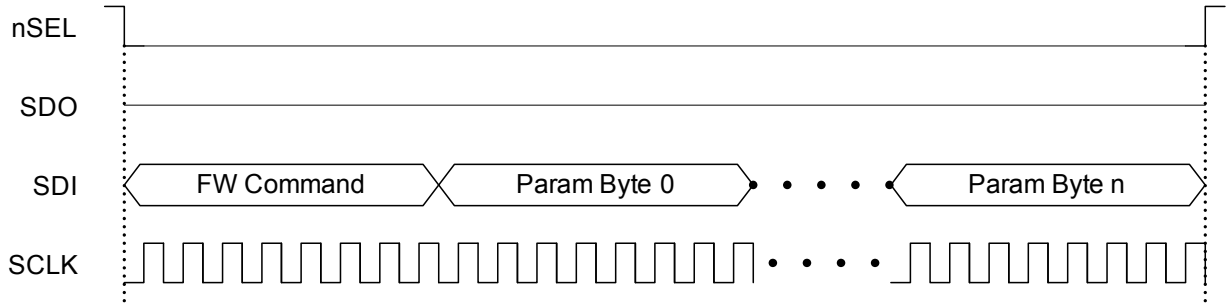
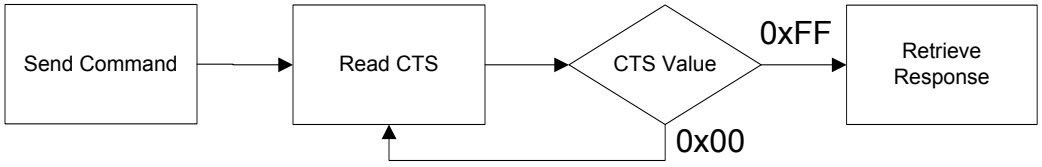


Figure 2. SPI Write Command

The Si4362 contains an internal MCU that controls all the internal functions of the radio. For SPI read commands a typical MCU flow of checking clear-to-send (CTS) is used to make sure the internal MCU has executed the command and prepared the data to be output over the SDO pin. Figure 3 demonstrates the general flow of an SPI read command. Once the CTS value reads FFh then the read data is ready to be clocked out to the host MCU. The typical time for a valid FFh CTS reading is 20 µs. Figure 4 demonstrates the remaining read cycle after CTS is set to FFh. The internal MCU will clock out the SDO data on the negative edge so the host MCU should process the SDO data on the rising edge of SCLK.

Firmware Flow



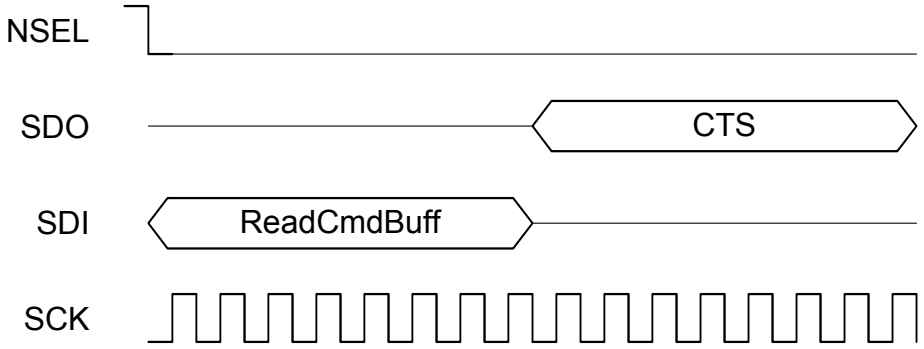


Figure 3. SPI Read Command—Check CTS Value

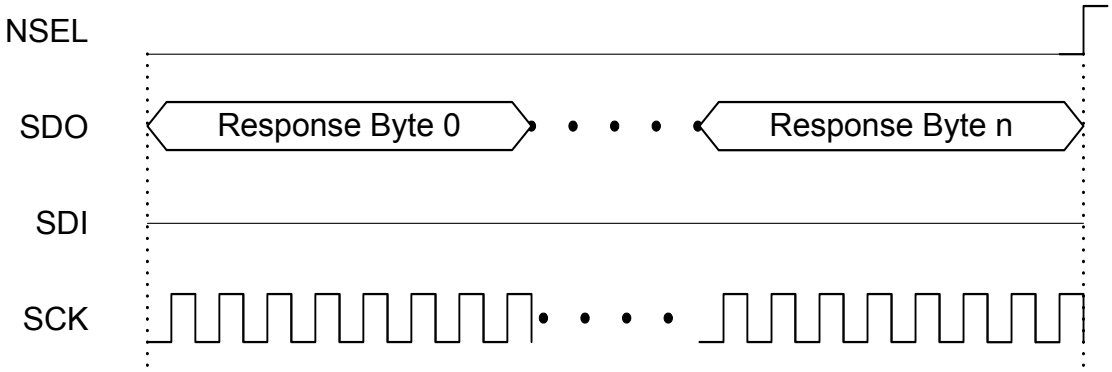


Figure 4. SPI Read Command—Clock Out Read Data

3.2. Fast Response Registers

The fast response registers are registers that can be read immediately without the requirement to monitor and check CTS. There are four fast response registers that can be programmed for a specific function. The fast response registers can be read through API commands, 0x50 for Fast Response A, 0x51 for Fast Response B, 0x53 for Fast Response C, and 0x57 for Fast Response D. The fast response registers can be configured by the "FRR_CTL_X_MODE" properties.

The fast response registers may be read in a burst fashion. After the initial 16 clock cycles, each additional eight clock cycles will clock out the contents of the next fast response register in a circular fashion. The value of the FRRs will not be updated unless NSEL is toggled.

3.3. Operating Modes and Timing

The primary states of the Si4362 are shown in Figure 5. The shutdown state completely shuts down the radio to minimize current consumption. Standby/Sleep, SPI Active, Ready, and RX tune are available to optimize the current consumption and response time to RX for a given application. API commands START_RX, and CHANGE_STATE control the operating state with the exception of shutdown which is controlled by SDN, pin 1. Table 9 shows each of the operating modes with the time required to reach RX mode as well as the current consumption of each mode. The times in Table 9 are measured from the rising edge of nSEL until the chip is in the desired state. Note that these times are indicative of state transition timing but are not guaranteed and should only be used as a reference data point. An automatic sequencer will put the chip into RX from any state. It is not necessary to manually step through the states. To simplify the diagram it is not shown but any of the lower power states can be returned to automatically after RX.

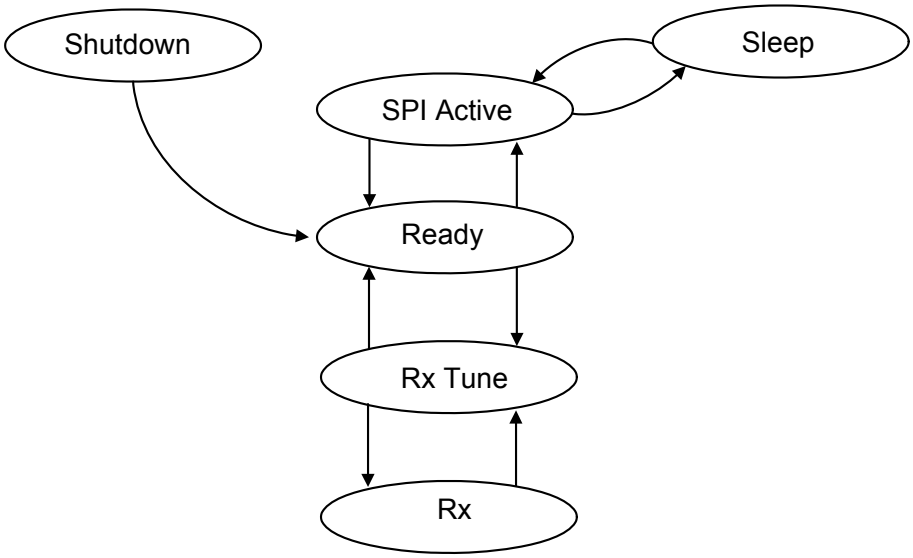


Figure 5. State Machine Diagram

Table 9. Operating State Response Time and Current Consumption*

State/Mode	Response Time to RX	Current in State/Mode
Shutdown State	15 ms	30 nA
Standby State Sleep State SPI Active State Ready State RX Tune State	440 µs 440 µs 340 µs 100 µs 60 µs	40 nA 740 nA 1.35 mA 1.8 mA 7.6 mA
RX State	75 µs	10.9 or 13.7 mA

Figure 6 shows the POR timing and voltage requirements. The power consumption (battery life) depends on the duty cycle of the application or how often the part is in either Rx state. In most applications the utilization of the standby state will be most advantageous for battery life but for very low duty cycle applications shutdown will have an advantage. For the fastest timing the next state can be selected in the START_RX API command to minimize SPI transactions and internal MCU processing.

3.3.1. Power on Reset (POR)

A power on reset (POR) sequence is used to boot the device up from a fully off or shutdown state. To execute this process, VDD must ramp within 1ms and must remain applied to the device for at least 10 ms. If VDD is removed, then it must stay below 0.15 V for at least 10 ms before being applied again. See Figure 6 and Table 10 for details.

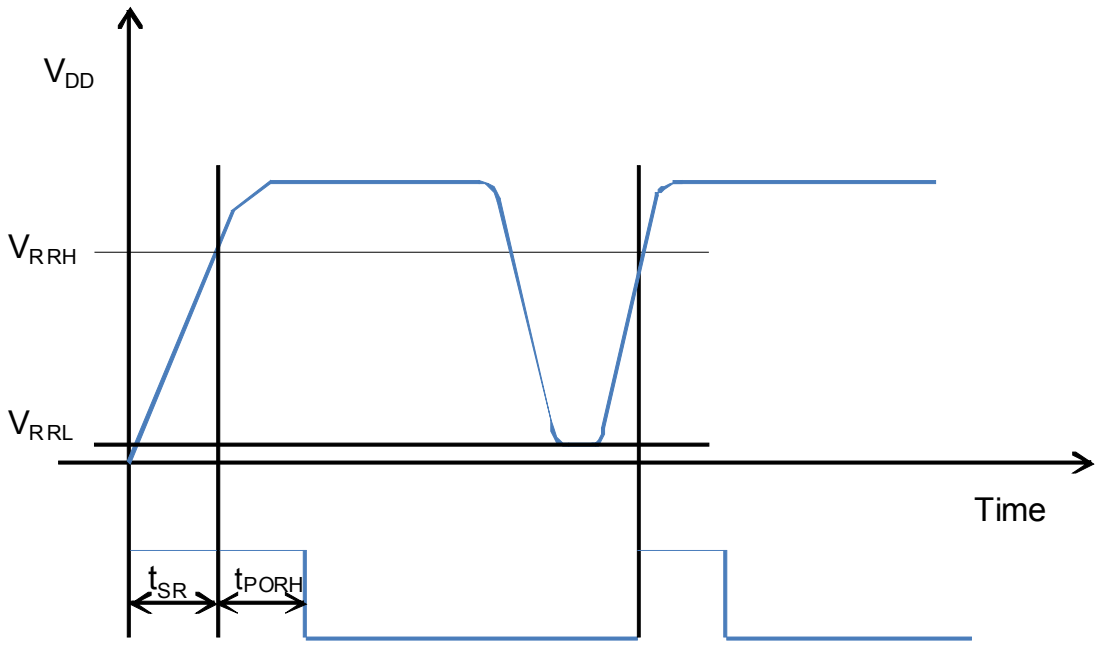


Figure 6. POR Timing Diagram

Table 10. POR Timing

Variable	Description	Min	Typ	Max	Units
t_{PORH}	High time for VDD to fully settle POR circuit.	10			ms
t_{PORL}	Low time for VDD to enable POR.	10			ms
V_{RRH}	Voltage for successful POR	90% x V_{DD}			V
V_{RRL}	Starting Voltage for successful POR	0		150	mV
t_{SR}	Slew rate of VDD for successful POR			1	ms

3.3.2. Shutdown State

The shutdown state is the lowest current consumption state of the device with nominally less than 30 nA of current consumption. The shutdown state may be entered by driving the SDN pin (Pin 1) high. The SDN pin should be held low in all states except the shutdown state. In the shutdown state, the contents of the registers are lost and there is no SPI access. When coming out of the shutdown state a power on reset (POR) will be initiated along with the internal calibrations. After the POR the POWER_UP command is required to initialize the radio. The SDN pin needs to be held high for at least 10 µs before driving low again so that internal capacitors can discharge. Not holding the SDN high for this period of time may cause the POR to be missed and the device to boot up incorrectly. If POR timing and voltage requirements cannot be met, it is highly recommended that SDN be controlled using the host processor rather than tying it to GND on the board.

3.3.3. Standby State

Standby state has the lowest current consumption with the exception of shutdown but has much faster response time to RX mode. In most cases standby should be used as the low power state. In this state the register values are maintained with all other blocks disabled. The SPI is accessible during this mode but any SPI event, including FIFO R/W, will enable an internal boot oscillator and automatically move the part to SPI active state. After an SPI event the host will need to re-command the device back to standby through the “Change State” API command to achieve the 40 nA current consumption. If an interrupt has occurred (i.e., the nIRQ pin = 0) the interrupt registers must be read to achieve the minimum current consumption of this mode.

3.3.4. Sleep State

Sleep state is the same as standby state but the wake-up-timer and a 32 kHz clock source are enabled. The source of the 32 kHz clock can either be an internal 32 kHz RC oscillator which is periodically calibrated or a 32 kHz oscillator using an external XTAL.The SPI is accessible during this mode but an SPI event will enable an internal boot oscillator and automatically move the part to SPI active mode. After an SPI event the host will need to re-command the device back to sleep. If an interrupt has occurred (i.e., the nIRQ pin = 0) the interrupt registers must be read to achieve the minimum current consumption of this mode.

3.3.5. SPI Active State

In SPI active state the SPI and a boot up oscillator are enabled. After SPI transactions during either standby or sleep the device will not automatically return to these states. A “Change State” API command will be required to return to either the standby or sleep modes.

3.3.6. Ready State

Ready state is designed to give a fast transition time to RX state with reasonable current consumption. In this mode the Crystal oscillator remains enabled reducing the time required to switch to RX mode by eliminating the crystal start-up time.

3.3.7. RX State

The RX state may be entered from any of the other states by using the “Start RX” or “Change State” API command. A built-in sequencer takes care of all the actions required to transition between states. The following sequence of events will occur automatically to get the chip into RX mode when going from standby to RX state:

1. Enable the digital LDO and the analog LDOs.
2. Start up crystal oscillator and wait until ready (controlled by an internal timer).
3. Enable PLL.
4. Calibrate VCO
5. Wait until PLL settles to required receive frequency (controlled by an internal timer).
6. Enable receiver circuits: LNA, mixers, and ADC.
7. Enable receive mode in the digital modem.

Depending on the configuration of the radio, all or some of the following functions will be performed automatically by the digital modem: AGC, AFC (optional), update status registers, bit synchronization, packet handling (optional) including sync word, header check, and CRC. The next state after RX may be defined in the “Start RX” API command. The START_RX commands and timing will be equivalent to the timing shown in Figure 7.

3.4. Application Programming Interface (API)

An application programming interface (API), which the host MCU will communicate with, is embedded inside the device. The API is divided into two sections, commands and properties. The commands are used to control the chip and retrieve its status. The properties are general configurations which will change infrequently. The API descriptions can be found on the Silicon Labs web site.

3.5. Interrupts

The Si4362 is capable of generating an interrupt signal when certain events occur. The chip notifies the microcontroller that an interrupt event has occurred by setting the nIRQ output pin LOW = 0. This interrupt signal will be generated when any one (or more) of the interrupt events (corresponding to the Interrupt Status bits) occur. The nIRQ pin will remain low until the microcontroller reads the Interrupt Status Registers. The nIRQ output signal will then be reset until the next change in status is detected.

The interrupts sources are grouped into three groups: packet handler, chip status, and modem. The individual interrupts in these groups can be enabled/disabled in the interrupt property registers. An interrupt must be enabled for it to trigger an event on the nIRQ pin. The interrupt group and the individual interrupts must be enabled.

Once an interrupt event occurs and the nIRQ pin is low there are two ways to read and clear the interrupts. All of the interrupts may be read and cleared in the “GET_INT_STATUS” API command. By default all interrupts will be cleared once read. If only specific interrupts want to be read in the fastest possible method the individual interrupt groups (Packet Handler, Chip Status, Modem) may be read and cleared by the “GET_MODEM_STATUS”, “GET_PH_STATUS” (packet handler), and “GET_CHIP_STATUS” API commands.

The instantaneous status of a specific function maybe read if the specific interrupt is enabled or disabled. The status results are provided after the interrupts and can be read with the same commands as the interrupts. The status bits will give the current state of the function whether the interrupt is enabled or not.

The fast response registers can also give information about the interrupt groups but reading the fast response registers will not clear the interrupt and reset the nIRQ pin.

3.6. GPIO

Four general purpose IO pins are available to utilize in the application. The GPIO are configured by the GPIO_PIN_CFG command in address 13h. For a complete list of the GPIO options please see the API guide. GPIO pins 0 and 1 should be used for active signals such as data or clock. GPIO pins 2 and 3 have more susceptibility to generating spurious in the synthesizer than pins 0 and 1. The drive strength of the GPIOs can be adjusted with the GEN_CONFIG parameter in the GPIO_PIN_CFG command. By default the drive strength is set to minimum. The default configuration for the GPIOs and the state during SDN is shown below in Table 11. The state of the IO during shutdown is also shown in Table 11. As indicated previously in Table 5, GPIO 0 has lower drive strength than the other GPIOs.

Table 11. GPIOs

Pin	SDN State	POR Default
GPIO0	0	POR
GPIO1	0	CTS
GPIO2	0	POR
GPIO3	0	POR
nIRQ	resistive VDD pull-up	nIRQ
SDO	resistive VDD pull-up	SDO
SDI	High Z	SDI
SCLK	High Z	SCLK
NSEL	High Z	NSEL

4. Modulation and Hardware Configuration Options

The Si4362 supports different modulation options and can be used in various configurations to tailor the device to any specific application or legacy system for drop in replacement. The modulation and configuration options are set in API property, MODEM_MOD_TYPE. Refer to the API documentation for details on modem-related properties.

4.1. Hardware Configuration Options

There are different receive demodulator options to optimize the performance and mutually-exclusive options for how the RX data is transferred from the host MCU to the RF device.

4.1.1. Receive Demodulator Options

There are multiple demodulators integrated into the device to optimize the performance for different applications, modulation formats, and packet structures. The calculator built into WDS will choose the optimal demodulator based on the input criteria.

4.1.1.1. Synchronous Demodulator

The synchronous demodulator's internal frequency error estimator acquires the frequency error based on a 101010 preamble structure. The bit clock recovery circuit locks to the incoming data stream within four transactions of a “10” or “01” bit stream. The synchronous demodulator gives optimal performance for 2- or 4-level FSK or GFSK modulation that has a modulation index less than 2.

4.1.1.2. Asynchronous Demodulator

The asynchronous demodulator should be used OOK modulation and for FSK/GFSK under one or more of the following conditions:

- Modulation index $\geq$ 2
- Non-standard preamble (not 1010101... pattern)

When the modulation index exceeds 2, the asynchronous demodulator has better sensitivity compared to the synchronous demodulator. An internal deglitch circuit provides a glitch-free data output and a data clock signal to simplify the interface to the host. There is no requirement to perform deglitching in the host MCU. The asynchronous demodulator will typically be utilized for legacy systems and will have many performance benefits over devices used in legacy designs. Unlike the Si4432/31 solution for non-standard packet structures, there is no requirement to perform deglitching on the data in the host MCU. Glitch-free data is output from the Si4362, and a sample clock for the asynchronous data can also be supplied to the host MCU; so, oversampling or bit clock recovery is not required by the host MCU. There are multiple detector options in the asynchronous demodulator block, which will be selected based upon the options entered into the WDS calculator. The asynchronous demodulator's internal frequency error estimator is able to acquire the frequency error based on any preamble structure.

4.1.2. RX Data Interface With MCU

There are two different options for transferring the data from the RF device to the host MCU. FIFO mode uses the SPI interface to transfer the data, while direct mode transfers the data in real time over GPIO.

4.1.2.1. FIFO Mode

In FIFO mode, the receive data are stored in integrated FIFO register memory. The RX FIFO is accessed by writing command 77h followed by the number of clock cycles of data the host would like to read out of the RX FIFO. The RX data will be clocked out onto the SDO pin.

In RX mode, the Packet Handler must be enabled to allow storage of received data bytes into RX FIFO memory. The Packet Handler is required to detect the Sync Word, and proper detection of the Sync Word is required to determine the start of the Payload. All bytes after the Sync Word are stored in RX FIFO memory except the CRC checksum and (optionally) the variable packet length byte(s). When the FIFO is being used in RX mode, all of the received data may still be observed directly (in realtime) by properly programming a GPIO pin as the RXDATA output pin; this can be quite useful during application development.

When in FIFO mode, the chip will automatically exit the RX State when PACKET_RX interrupt occurs. The chip will return to the IDLE state programmed in the argument of the “START RX” API command, RXVALID_STATE[3:0].

4.1.2.2. FIFO Direct Mode (Infinite Receive)

In some applications, there is a need to receive extremely long packets (greater than 40 kB) while relying on preamble and sync word detection from the on-chip packet handler. In these cases, the packet length is unknown, and the device will load the bits after the sync word into the RX FIFO forever. Other features, such as Data Whitening, CRC, Manchester, etc., are supported in this mode, but CRC calculation is not because the end of packet is unknown to the device. The RX data and clock are also available on GPIO pins. The host MCU will need to reset the packet handler by issuing a START_RX to begin searching for a new packet.

4.1.2.3. Direct Mode

For legacy systems that perform packet handling within the host MCU or other baseband chip, it may not be desirable to use the FIFO. For this scenario, a Direct mode is provided, which bypasses the FIFOs entirely. In RX Direct mode, the RX Data and RX Clock can be programmed for direct (real-time) output to GPIO pins. The microcontroller may then process the RX data without using the FIFO or packet handler functions of the RFIC.

4.2. Preamble Length

4.2.1. Digital Signal Arrival Detector (DSA)

Traditional preamble detection requires 20 bits to detect preamble. This device introduces a new approach to signal detection that can detect a preamble pattern in as little as one byte. If AFC is enabled a preamble length of two bytes is sufficient to reliably detect signal arrival and settle a one shot AFC. The impact of this is significant for low-power solutions as it reduces the amount of time the receiver has to stay active to detect the preamble. This feature is used with Preamble Sense Mode (see "8.6. Preamble Sense Mode" on page 34) and the latest WMBus N modes as well as with features, such as frequency hopping, which may use signal arrival as a condition to hop. The traditional preamble detector is also available to maintain backward compatibility. Note that the DSA is using the RSSI jump detector. When used for collision detection, the RSSI jump detector may need to be reconfigured after preamble detection. Refer to the API documentation for details on how to configure the device to use the signal arrival detector.

4.2.2. Traditional Preamble Detection

Optimal performance of the chip is obtained by qualifying reception of a valid Preamble pattern prior to continuing with reception of the remainder of the packet (e.g., Sync Word and Payload). Reception of the Preamble is considered valid when a minimum number of consecutive bits of 101010... pattern have been received; the required threshold for preamble detection is specified by the RX_THRESH[6:0] field in the PREAMBLE_CONFIG_STD_1 property. The appropriate value of the detection threshold depends upon the system application and typically trades off speed of acquisition against the probability of false detection. If the detection threshold is set too low, the chip may readily detect the short pattern within noise; the chip then proceeds to attempt to detect the remainder of the non-existent packet, with the result that the arrival of an actual valid packet may be missed. If the detection threshold is set too high, the required number of transmitted Preamble bits must be increased accordingly, leading to longer packet lengths and shorter battery life. A preamble detection threshold value of 20 bits is suitable for most applications. The total length of the transmitted Preamble field must be at least equal to the receive preamble detection threshold, plus an additional number of bits to allow for acquisition of bit timing and settling of the AFC algorithm. The recommended preamble detection thresholds and preamble lengths for a variety of operational modes are listed in Table 12.

Configuration of the preamble detection threshold in the RX_THRESH[6:0] field is only required for reception of a standard Preamble pattern (i.e., 101010... pattern). Reception of a repetitive but non-standard Preamble pattern is also supported in the chip but is configured through the PREAMBLE_CONFIG_NSTD and PREAMBLE_PATTERN properties.

Table 12. Recommended Preamble Length

Mode	AFC	Antenna Diversity	Preamble Type	Recommended Preamble Length	Recommended Preamble Detection Threshold
(G)FSK	Disabled	Disabled	Standard	4 Bytes	20 bits
(G)FSK	Enabled	Disabled	Standard	5 Bytes	20 bits
(G)FSK	Disabled	Disabled	Non-standard	2 Bytes	0 bits
(G)FSK	Enabled		Non-standard	Not Supported	
(G)FSK	Disabled	Enabled	Standard	7 Bytes	24 bits
(G)FSK	Enabled	Enabled	Standard	8 Bytes	24 bits
4(G)FSK	Disabled	Disabled	Standard	40 symbols	16 symbols
4(G)FSK	Enabled	Disabled	Standard	48 symbols	16 symbols
4(G)FSK			Non-standard	Not Supported	
OOK	Disabled	Disabled	Standard	4 Bytes	20 bits
OOK	Disabled	Disabled	Non-standard	2 Bytes	0 bits
OOK	Enabled			Not Supported	

Notes:

1. The recommended preamble length and preamble detection thresholds listed above are to achieve 0% PER. They may be shortened when occasional packet errors are tolerable.
2. All recommended preamble lengths and detection thresholds include AGC and BCR settling times.
3. “Standard” preamble type should be set for an alternating data sequence at the max data rate (…10101010…)
4. “Non-standard” preamble type can be set for any preamble type including …10101010...
5. When preamble detection threshold = 0, sync word needs to be 3 Bytes to avoid false syncs. When only a 2 Byte sync word is available the sync word detection can be extended by including the last preamble Byte into the RX sync word setting.

5. Internal Functional Blocks

The following sections provide an overview to the key internal blocks and features.

5.1. RX Chain

The internal low-noise amplifier (LNA) is designed to be a wide-band LNA that can be matched with three external discrete components to cover any common range of frequencies in the sub-GHz band. The LNA has extremely low noise to suppress the noise of the following stages and achieve optimal sensitivity; so, no external gain or front-end modules are necessary. The LNA has gain control, which is controlled by the internal automatic gain control (AGC) algorithm. The LNA is followed by an I-Q mixer, filter, programmable gain amplifier (PGA), and ADC. The I-Q mixers downconvert the signal to an intermediate frequency. The PGA then boosts the gain to be within dynamic range of the ADC. The ADC rejects out-of-band blockers and converts the signal to the digital domain where filtering, demodulation, and processing is performed. Peak detectors are integrated at the output of the LNA and PGA for use in the AGC algorithm.

5.1.1. RX Chain Architecture

It is possible to operate the RX chain in different architecture configurations: fixed-IF, zero-IF, scaled-IF, and modulated IF. There are trade-offs between the architectures in terms of sensitivity, selectivity, and image rejection. Fixed-IF is the default configuration and is recommended for most applications. With 35 dB native image rejection and autonomous image calibration to achieve 55 dB, the fixed-IF solution gives the best performance for most applications. Fixed-IF obtains the best sensitivity, but it has the effect of degraded selectivity at the image frequency. An autonomous image rejection calibration is included in the Si4362 and described in more detail in "5.2.3. Image Rejection and Calibration" on page 25. For fixed-IF and zero-IF, the sensitivity is degraded for data rates less than 100 kbps or bandwidths less than 200 kHz. The reduction in sensitivity is caused by increased flicker noise as dc is approached. The benefit of zero-IF is that there is no image frequency; so, there is no degradation in the selectivity curve, but it has the worst sensitivity. Modulated IF is useful for OOK if image elimination is required similar to Zero-IF. Scaled-IF is a trade-off between fixed-IF and zero-IF. In the scaled-IF architecture, the image frequency is placed or hidden in the adjacent channel where it only slightly degrades the typical adjacent channel selectivity. The scaled-IF approach has better sensitivity than zero-IF but still some degradation in selectivity due to the image. In scaled-IF mode, the image frequency is directly proportional to the channel bandwidth selected. Figure 7 demonstrates the trade-off in sensitivity between the different architecture options.

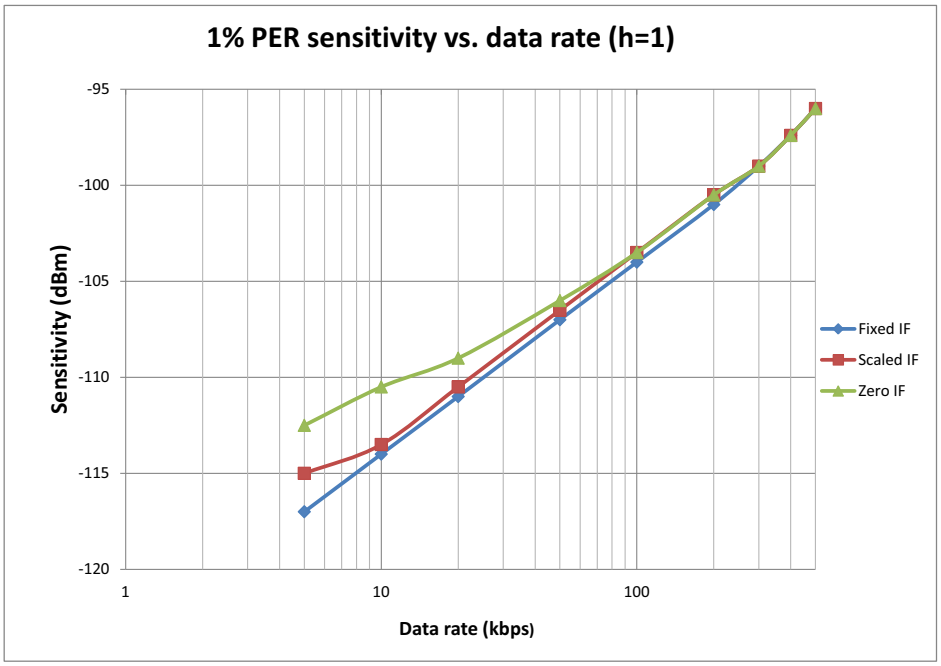


Figure 7. RX Architecture vs. Data Rate

5.2. RX Modem

Using high-performance ADCs allows channel filtering, image rejection, and demodulation to be performed in the digital domain, which allows for flexibility in optimizing the device for particular applications. The digital modem performs the following functions:

- Channel selection filter
- RX demodulation
- Automatic Gain Control (AGC)
- Preamble detection
- Invalid preamble detection
- Radio signal strength indicator (RSSI)
- Automatic frequency compensation (AFC)
- Image Rejection Calibration
- Packet handling
- Cyclic redundancy check (CRC)

The digital channel filter and demodulator are optimized for ultra-low-power consumption and are highly configurable. Supported modulation types are GFSK, FSK, 4GFSK, 4FSK, GMSK, and OOK. The channel filter can be configured to support bandwidths ranging from 850 down to 1.1 kHz. A large variety of data rates are supported ranging from 100 bps up to 1 Mbps. The configurable preamble detector is used with the synchronous demodulator to improve the reliability of the sync-word detection. Preamble detection can be skipped using only sync detection, which is a valuable feature in certain applications. The received signal strength indicator (RSSI) provides a measure of the signal strength received on the tuned channel. The resolution of the RSSI is 0.5 dB. This high-resolution RSSI enables accurate channel power measurements for clear channel assessment (CCA), carrier sense (CS), and listen before talk (LBT) functionality. A comprehensive programmable packet handler is integrated to create a variety of communication topologies ranging from peer-to-peer networks to mesh networks. The extensive programmability of the packet header allows for advanced packet filtering, which, in turn enables a mix of broadcast, group, and point-to-point communication. A wireless communication channel can be corrupted by noise and interference, so it is important to know if the received data is free of errors. A cyclic redundancy check (CRC) is used to detect the presence of erroneous bits in each packet. A CRC is computed and appended at the end of each transmitted packet and verified by the receiver to confirm that no errors have occurred. The packet handler and CRC can significantly reduce the load on the system microcontroller allowing for a simpler and cheaper microcontroller.

5.2.1. Automatic Gain Control (AGC)

The AGC algorithm is implemented digitally using an advanced control loop optimized for fast response time. The AGC occurs within a single bit or in less than 2 µs. Peak detectors at the output of the LNA and PGA allow for optimal adjustment of the LNA gain and PGA gain to optimize IM3, selectivity, and sensitivity performance.

5.2.2. Auto Frequency Correction (AFC)

Frequency mistuning caused by crystal inaccuracies can be compensated for by enabling the digital automatic frequency control (AFC) in receive mode. There are two types of integrated frequency compensation: modem frequency compensation, and AFC by adjusting the PLL frequency. With AFC disabled, the modem compensation can correct for frequency offsets up to ±0.25 times the IF bandwidth. When the AFC is enabled, the received signal will be centered in the pass-band of the IF filter, providing optimal sensitivity and selectivity over a wider range of frequency offsets up to ±0.35 times the IF bandwidth. When AFC is enabled, the preamble length needs to be long enough to settle the AFC. As shown in Table 12 on page 22, an additional byte of preamble is typically required to settle the AFC.

5.2.3. Image Rejection and Calibration

Since the receiver utilizes a low-IF architecture, the selectivity will be affected by the image frequency. The IF frequency is 468.75 kHz (Fxtal/64), and the image frequency will be at 937.5 kHz below the RF frequency. The native image rejection of the Si4362 is 40 dB. Image rejection calibration is available in the Si4362 to improve the image rejection to more than 55 dB. The calibration is initiated with the IRCAL API command. The calibration uses an internal signal source, so no external signal generator is required. The initial calibration takes 250 ms, and periodic re-calibration takes 100 ms. Re-calibration should be initiated when the temperature has changed more than 30 °C.

5.2.4. Received Signal Strength Indicator

The received signal strength indicator (RSSI) is an estimate of the signal strength in the channel to which the receiver is tuned. The RSSI measurement is done after the channel filter, so it is only a measurement of the in-band signal power (desired or undesired). There are two methods for reading the RSSI value and several different options for configuring the returned RSSI value. The fastest method for reading the RSSI is to configure one of the four fast response registers (FRR) to return a latched RSSI value. The latched RSSI value is measured once per packet and is latched at a configurable amount of time after RX mode is entered. The fast response registers can be read in 16 SPI clock cycles with no requirement to wait for CTS. The RSSI value may also be read out of the GET_MODEM_STATUS command. In this command, both the current RSSI and the latched RSSI are available. The current RSSI value represents the signal strength at the instant in time the GET_MODEM_STATUS command is processed and may be read multiple times per packet. Reading the RSSI in the GET_MODEM_STATUS command takes longer than reading the RSSI out of the fast response register. After the initial command, it takes 33 µs for CTS to be set and then the four or five bytes of SPI clock cycles to read out the respective current or latched RSSI values.

The RSSI configuration options are set in the MODEM_RSSI_CONTROL API property. The latched RSSI value may be latched and stored based on the following events: preamble detection, sync detection, or a configurable number of bit times measured after the start of RX mode (minimum of 4 bit times). The requirement for a minimum of four bit times is determined by the processing delay and settling through the modem and digital channel filter. In MODEM_RSSI_CONTROL, the RSSI may be defined to update every bit period or to be averaged and updated every four bit periods. If RSSI averaging over four bits is enabled, the latched RSSI value will be delayed to a minimum of seven bits after the start of RX mode to allow for the averaging. The latched RSSI values are cleared when entering RX mode so they may be read after the packet is received or after dropping back to standby mode. If the RSSI value has been cleared by the start of RX but not yet latched, a value of 0 will be returned if it is attempted to be read.

The RSSI value read by the API may be translated into dBm by the following linear equation:

RF_Input_Level_dBm = (RSSI_value / 2) – MODEM_RSSI_COMP – 70

The MODEM_RSSI_COMP property provides for fine adjustment of the relationship between the actual RF input level (in dBm) and the returned RSSI value. That is, adjustment of this property allows the user to shift the RSSI vs RF Input Power curve up and down. This may be desirable to compensate for differences in front-end insertion loss between multiple designs (e.g., due to the presence of a SAW preselection filter, or an RF switch). A value of MODEM_RSSI_COMP = 0x40 = 64d is appropriate for most applications.

Clear channel assessment (CCA) or RSSI threshold detection is also available. An RSSI threshold may be set in the MODEM_RSSI_THRESH API property. If the Current RSSI value is above this threshold, an interrupt or GPIO may notify the host. Both the latched version and asynchronous version of this threshold are available on any of the GPIOs. Automatic fast hopping based on RSSI is available. See “5.3.1.2. Automatic RX Hopping and Hop Table”.

5.3. Synthesizer

An integrated Sigma Delta (ΣΔ) Fractional-N PLL synthesizer capable of operating over the bands from 142–175, 283–350, 420–525, and 850–1050 MHz for the Si4362. Using a ΣΔ synthesizer has many advantages; it provides flexibility in choosing data rate, deviation, channel frequency, and channel spacing. The nominal reference frequency to the PLL is 30 MHz, but any XTAL frequency from 25 to 32 MHz may be used. The modem configuration calculator in WDS will automatically account for the XTAL frequency being used. The PLL utilizes a differential LC VCO with integrated on-chip inductors. The output of the VCO is followed by a configurable divider, which will divide the signal down to the desired output frequency band.

5.3.1. Synthesizer Frequency Control

The frequency is set by changing the integer and fractional settings to the synthesizer. The WDS calculator will automatically provide these settings, but the synthesizer equation is shown below for convenience. The APIs for setting the frequency are FREQ_CONTROL_INTE, FREQ_CONTROL_FRAC2, FREQ_CONTROL_FRAC1, and FREQ_CONTROL_FRAC0.

$$\text{RF_channel} = \left(\text{fc_inte} + \frac{\text{fc_frac}}{2^{19}}\right) \times \frac{2 \times \text{freq_xo}}{\text{outdiv}} (\text{Hz})$$

Note: The fc_frac/2^{19} value in the above formula has to be a number between 1 and 2.

Table 13. Output Divider (Outdiv) Values for the Si4362

Outdiv	Lower (MHz)	Upper (MHz)
24	142	175
12	284	350
10	350	420
8	420	525
4	850	1050

5.3.1.1. EZ Frequency Programming

In applications that utilize multiple frequencies or channels, it may not be desirable to write four API registers each time a frequency change is required. EZ frequency programming is provided so that only a single register write (channel number) is required to change frequency. A base frequency is first set by first programming the integer and fractional components of the synthesizer. This base frequency will correspond to channel 0. Next, a channel step size is programmed into the FREQ_CONTROL_CHANNEL_STEP_SIZE_1 and FREQ_CONTROL_CHANNEL_STEP_SIZE_0 API registers. The resulting frequency will be:

$$\text{RF Frequency} = \text{Base Frerquency} + \text{Channel} \times \text{Stepsize}$$

The second argument of the START_RX is CHANNEL, which sets the channel number for EZ frequency programming. For example, if the channel step size is set to 1 MHz, the base frequency is set to 900 MHz with the INTE and FRAC API registers, and a CHANNEL number of 5 is programmed during the START_RX command, the resulting frequency will be 905 MHz. If no CHANNEL argument is written as part of the START_RX command, it will default to the previous value. The initial value of CHANNEL is 0; so, if no CHANNEL value is written, it will result in the programmed base frequency.

5.3.1.2. Automatic RX Hopping and Hop Table

The receiver supports an automatic hopping feature that can be fully configured through the API. This is intended for RX hopping where the device has to hop from channel to channel and look for packets. Once the device is put into the RX state, it automatically starts hopping through the hop table if the feature is enabled.

The hop table can hold up to 64 entries and is maintained in firmware. Each entry is a channel number; so, the hop table can hold up to 64 channels. The number of entries in the table is set by RX HOP TABLE_SIZE API. The specified channels correspond to the EZ frequency programming method for programming the frequency. The receiver starts at the base channel and hops in sequence from the top of the hop table to the bottom. The table will

wrap around to the base channel once it reaches the end of the table. An entry of 0xFF in the table indicates that the entry should be skipped. The device will hop to the next non 0xFF entry.

There are three conditions that can be used to determine whether to continue hopping or to stay on a particular channel. These conditions are:

- RSSI threshold
- Preamble timeout (invalid preamble pattern)
- Sync word timeout (invalid or no sync word detected after preamble)

These conditions can be used individually, or they can be enabled all together by configuring the RX_HOP_CONTROL API. However, the firmware will make a decision on whether or not to hop based on the first condition that is met.

The RSSI that is monitored is the current RSSI value. This is compared to the threshold, and, if it is above the threshold value, it will stay on the channel. If the RSSI is below the threshold, it will continue hopping. There is no averaging of RSSI done during the automatic hopping from channel to channel. Since the preamble timeout and the sync word timeout are features that require packet handling, the RSSI threshold is the only condition that can be used if the user is in “direct” or “RAW” mode where packet handling features are not used.

Note that the RSSI threshold is not an absolute RSSI value; instead, it is a relative value and should be verified on the bench to find an optimal threshold for the application.

The turnaround time from RX to RX on a different channel using this method is 115 µs. The time spent in receive mode will be determined by the configuration of the hop conditions. Manual RX hopping will have the fastest turn-around time but will require more overhead and management by the host MCU.

The following are example steps for using Auto Hop:

1. Set the base frequency (inte + frac) and channel step size.
2. Define the number of entries in the hop table (RX_HOP_TABLE_SIZE).
3. Write the channels to the hop table (RX_HOP_TABLE_ENTRY_n)
4. Configure the hop condition and enable auto hopping- RSSI, preamble, or sync (RX_HOP_CONTROL).
5. Set preamble and sync parameters if enabled.
6. Program the RSSI threshold property in the modem using “MODEM_RSSI_THRESH”.
7. Set the preamble threshold using “PREAMBLE_CONFIG_STD_1”.
8. Program the preamble timeout property using “PREAMBLE_CONFIG_STD_2”.
9. Set the sync detection parameters if enabled.
10. If needed, use “GPIO_PIN_CFG” to configure a GPIO to toggle on hop and hop table wrap.
11. Use the “START_RX” API with channel number set to the first valid entry in the hop table (i.e., the first non 0xFF entry).
12. Device should now be in auto hop mode.

5.3.1.3. Manual RX Hopping

The RX_HOP command provides the fastest method for hopping from RX to RX but it requires more overhead and management by the host MCU. Using the RX_HOP command, the turn-around time is 75 µs. The timing is faster with this method than Start_RX or RX hopping because one of the calculations required for the synthesizer calibrations is offloaded to the host and must be calculated/stored by the host, VCO_CNT0. For information about using fast manual hopping, contact customer support.

5.4. Crystal Oscillator

The Si4362 includes an integrated crystal oscillator with a fast start-up time of less than 250 µs. The design is differential with the required crystal load capacitance integrated on-chip to minimize the number of external components. By default, all that is required off-chip is the crystal. The default crystal is 30 MHz, but the circuit is designed to handle any XTAL from 25 to 32 MHz. If a crystal different than 30 MHz is used, the POWER_UP API boot command must be modified. The WDS calculator crystal frequency field must also be changed to reflect the frequency being used. The crystal load capacitance can be digitally programmed to accommodate crystals with various load capacitance requirements and to adjust the frequency of the crystal oscillator. The tuning of the crystal load capacitance is programmed through the GLOBAL_XO_TUNE API property. The total internal capacitance is 11 pF and is adjustable in 127 steps (70 fF/step). The crystal frequency adjustment can be used to compensate for crystal production tolerances. The frequency offset characteristics of the capacitor bank are demonstrated in Figure 8.

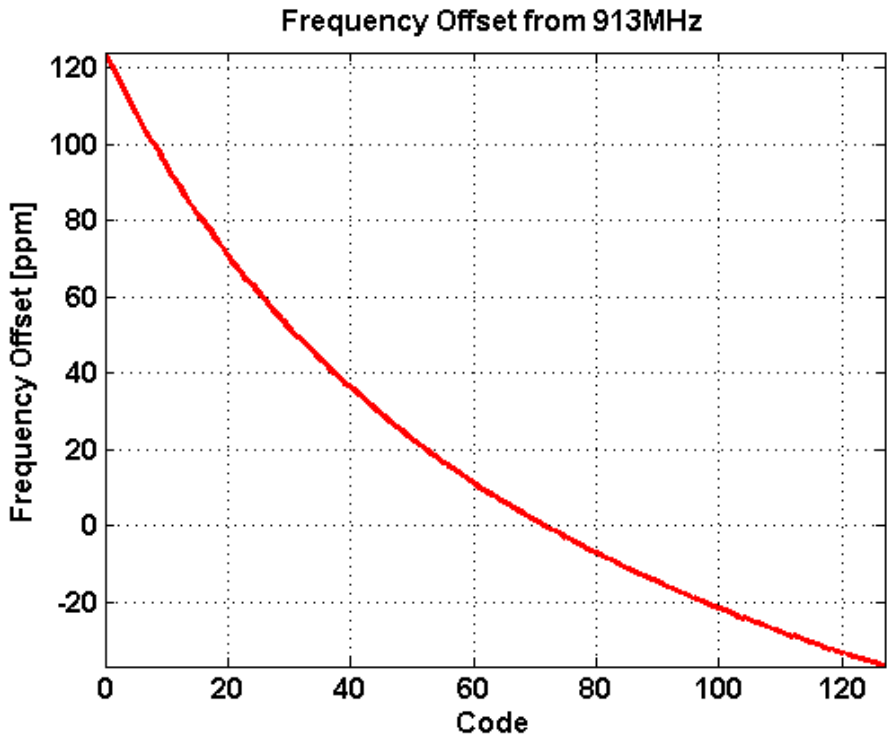


Figure 8. Capacitor Bank Frequency Offset Characteristics

Utilizing the on-chip temperature sensor and suitable control software, the temperature dependency of the crystal can be canceled.

A TCXO or external signal source can easily be used in place of a conventional XTAL and should be connected to the XIN pin. It is recommended that the incoming clock signal have a peak-to-peak swing in the range of 600 mV to 1.4 V and ac-coupled to the XIN pin. If the peak-to-peak swing of the TCXO exceeds 1.4 V peak-to-peak, then dc coupling to the XIN pin should be used. The maximum allowed swing on XIN is 1.8 V peak-to-peak. The XO capacitor bank should be set to 0 whenever an external drive is used on the XIN pin. In addition, the POWER_UP command should be invoked with the TCXO option whenever the external drive is used.

6. Data Handling and Packet Handler

6.1. RX FIFOs

The FIFO size is up to 129 bytes. Reading from command Register 77h reads data from the RX FIFO. The RX FIFO has one programmable threshold, which is programmed by setting the “RX_FIFO_FULL” property. When the incoming RX data crosses the Almost Full Threshold, an interrupt will be generated to the microcontroller via the nIRQ pin. The microcontroller will then need to read the data from the RX FIFO. The RX Almost Full Threshold indication implies that the host can read at least the threshold number of bytes from the RX FIFO at that time. The RX FIFO may be cleared or reset with the “FIFO_RESET” command.

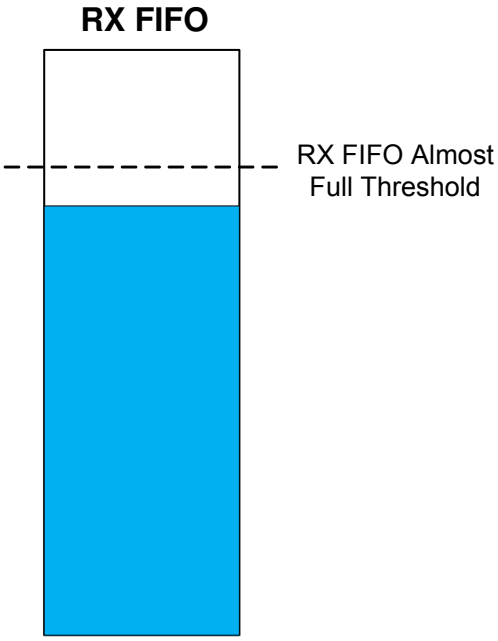


Figure 9. RX FIFO

6.2. Packet Handler

When using the FIFOs, automatic packet handling may be enabled for RX mode. The usual fields for network communication, such as preamble, synchronization word, headers, packet length, and CRC, can be configured to be automatically added to the data payload. Automatically adding these fields to the data payload and automatically checking them in RX mode greatly reduces the amount of communication between the microcontroller and the Si4362. It also greatly reduces the required computational power of the microcontroller. The general packet structure is shown in Figure 10. Any or all of the fields can be enabled and checked by the internal packet handler.

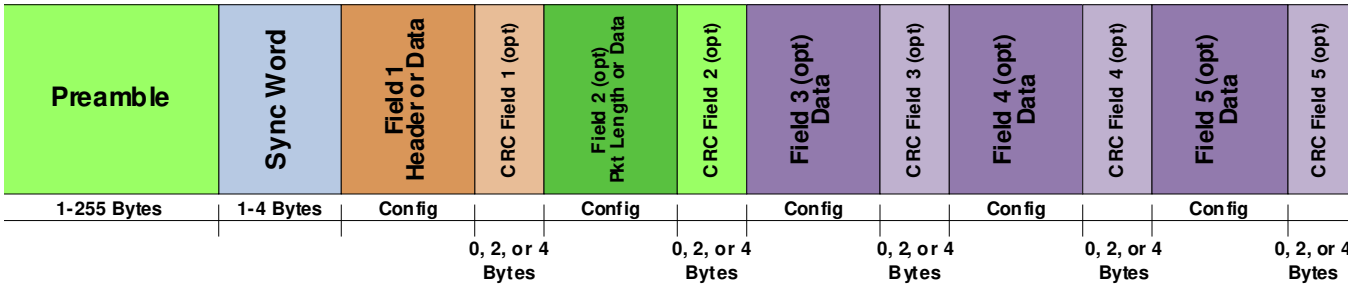


Figure 10. Packet Handler Structure

The fields are highly programmable and can be used to check any kind of pattern in a packet structure. The general functions of the packet handler include the following:

- Detection/validation of Preamble quality in RX mode (PREAMBLE_VALID signal)
- Detection of Sync word in RX mode (SYNC_OK signal)
- Detection of valid packets in RX mode (PKT_VALID signal)
- Detection of CRC errors in RX mode (CRC_ERR signal)
- Data de-whitening and/or Manchester decoding (if enabled) in RX mode
- Match/Header checking in RX mode
- Storage of Data Field bytes into FIFO memory in RX mode

For details on how to configure the packet handler, see “AN626: Packet Handler Operation for Si446x RFICs”.

7. RX Modem Configuration

The Si4362 can easily be configured for different data rate, deviation, frequency, etc. by using the Radio Configuration Application (RCA) GUI which is part of the Wireless Development Suite (WDS) program.

8. Auxiliary Blocks

8.1. Wake-up Timer and 32 kHz Clock Source

The chip contains an integrated wake-up timer that can be used to periodically wake the chip from sleep mode. The wake-up timer runs from either the internal 32 kHz RC Oscillator, or from an external 32 kHz XTAL.

The wake-up timer can be configured to run when in sleep mode. If WUT_EN = 1 in the GLOBAL_WUT_CONFIG property, prior to entering sleep mode, the wake-up timer will count for a time specified defined by the GLOBAL_WUT_R and GLOBAL_WUT_M properties. At the expiration of this period, an interrupt will be generated on the nIRQ pin if this interrupt is enabled in the INT_CTL_CHIP_ENABLE property. The microcontroller will then need to verify the interrupt by reading the chip interrupt status either via GET_INT_STATUS or a fast response register. The formula for calculating the Wake-Up Period is as follows:

$$WUT = WUT_M \times \frac{4 \times 2^{WUT_R}}{32.768}[ms]$$

The RC oscillator frequency will change with temperature; so, a periodic recalibration is required. The RC oscillator is automatically calibrated during the POWER_UP command and exits from the Shutdown state. To enable the recalibration feature, CAL_EN must be set in the GLOBAL_WUT_CONFIG property, and the desired calibration period should be selected via WUT_CAL_PERIOD[2:0] in the same API property. During the calibration, the 32 kHz RC oscillator frequency is compared to the 30 MHz XTAL and then adjusted accordingly. The calibration needs to start the 30 MHz XTAL, which increases the average current consumption; so, a longer CAL_PERIOD results in a lower average current consumption. The 32 kHz XTAL accuracy is comprised of both the XTAL parameters and the internal circuit. The XTAL accuracy can be defined as the XTAL initial error + XTAL aging + XTAL temperature drift + detuning from the internal oscillator circuit. The error caused by the internal circuit is typically less than 10 ppm.

8.2. Low Duty Cycle Mode (Auto RX Wake-Up)

The low duty cycle (LDC) mode is implemented to automatically wake-up the receiver to check if a valid signal is available. It allows low average current polling operation by the Si4362 for which the wake-up timer (WUT) is used. RX LDC operation must be set via the GLOBAL_WUT_CONFIG property when setting up the WUT. The LDC wake-up period is determined by the following formula:

$$LDC = WUT_LDC \times \frac{4 \times 2^{WUT_R}}{32.768}[ms]$$

where the WUT_LDC parameter can be set by the GLOBAL_WUT_LDC property. The WUT period must be set in conjunction with the LDC mode duration; for the relevant API properties, see the wake-up timer (WUT) section.

The basic operation of RX LDC mode is shown in Figure 11. The receiver periodically wakes itself up to work on RX_STATE during LDC mode duration. If a valid preamble is not detected, a receive error is detected, or an entire packet is not received, the receiver returns to the WUT state (i.e., ready or sleep) at the end of LDC mode duration and remains in that mode until the beginning of the next wake-up period. If a valid preamble or sync word is detected, the receiver delays the LDC mode duration to receive the entire packet. If a packet is not received during two LDC mode durations, the receiver returns to the WUT state at the last LDC mode duration until the beginning of the next wake-up period.

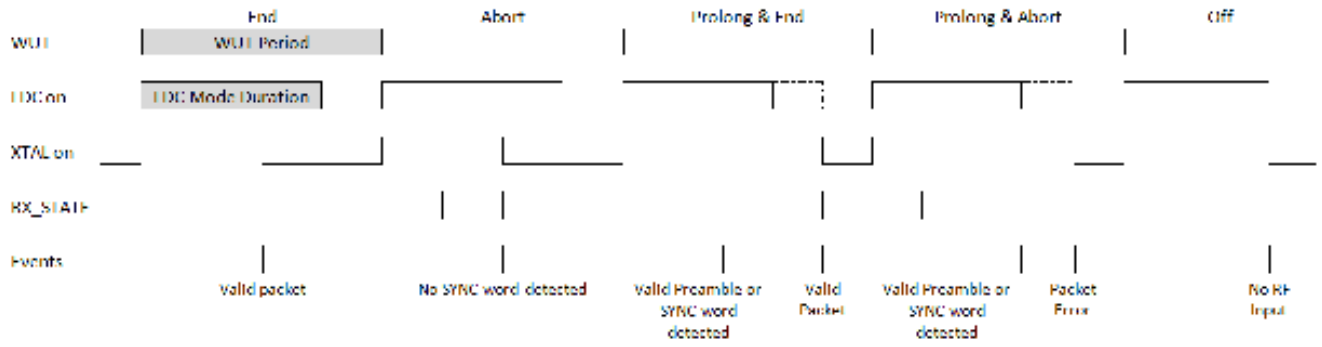


Figure 11. Low Duty Cycle Mode for RX

8.3. Temperature, Battery Voltage, and Auxiliary ADC

The Si4362 contains an integrated auxiliary ADC for measuring internal battery voltage, an internal temperature sensor, or an external component over a GPIO. The ADC utilizes a SAR architecture and achieves 11-bit resolution. The Effective Number of Bits (ENOB) is 9 bits. When measuring external components, the input voltage range is 1 V, and the conversion rate is between 300 Hz to 2.44 kHz. The ADC value is read by first sending the GET_ADC_READING command and enabling the inputs that are desired to be read: GPIO, battery, or temp. The temperature sensor accuracy at 25 °C is typically ±2 °C.

8.4. Low Battery Detector

The low battery detector (LBD) is enabled and utilized as part of the wake-up-timer (WUT). The LBD function is not available unless the WUT is enabled, but the host MCU can manually check the battery voltage anytime with the auxiliary ADC. The LBD function is enabled in the GLOBAL_WUT_CONFIG API property. The battery voltage will be compared against the threshold each time the WUT expires. The threshold for the LBD function is set in GLOBAL_LOW_BATT_THRESH. The threshold steps are in increments of 50 mV, ranging from a minimum of 1.5 V up to 3.05 V. The accuracy of the LBD is ±3%. The LBD notification can be configured as an interrupt on the nIRQ pin or enabled as a direct function on one of the GPIOs.

8.5. Antenna Diversity

To mitigate the problem of frequency-selective fading due to multipath propagation, some transceiver systems use a scheme known as antenna diversity. In this scheme, two antennas are used. Each time the receiver enters RX mode the receive signal strength from each antenna is evaluated. This evaluation process takes place during the preamble portion of the packet. The antenna with the strongest received signal is then used for the remainder of that RX packet. This chip fully supports antenna diversity with an integrated antenna diversity control algorithm. The required signals needed to control an external SPDT RF switch (such as a PIN diode or GaAs switch) are available on the GPIO pins. The operation of these GPIO signals is programmable to allow for different antenna diversity architectures and configurations. The antdiv[2:0] bits are found in the MODEM_ANT_DIV_CONTROL API property descriptions and enable the antenna diversity mode. The GPIO pins are capable of sourcing up to 5 mA of current; so, it may be used directly to forward-bias a PIN diode if desired. The antenna diversity algorithm will automatically toggle back and forth between the antennas until the packet starts to arrive. The recommended preamble length for optimal antenna selection is 8 bytes.

8.6. Preamble Sense Mode

This mode of operation is suitable for extremely low power applications where power consumption is important. The preamble sense mode (PSM) takes advantage of the Digital Signal Arrival detector (DSA), which can detect a preamble within eight bit times with no sensitivity degradation. This fast detection of an incoming signal can be combined with duty cycling of the receiver during the time the device is searching or sniffing for packets over the air. The average receive current is lowered significantly when using this mode. In applications where the timing of the incoming signal is unknown, the amount of power savings is primarily dependent on the data rate and preamble length as the Rx inactive time is determined by these factors. In applications where the sleep time is fixed and the timing of the incoming signal is known, the average current also depends on the sleep time. The PSM mode is similar to the low duty cycle mode but has the benefit of faster signal detection and autonomous duty cycling of the receiver to achieve even lower average receive currents. This mode can be used with the low power mode (LP) which has an active RX current of 10 mA or with the high-performance (HP) mode which has an active RX current of 13 mA.

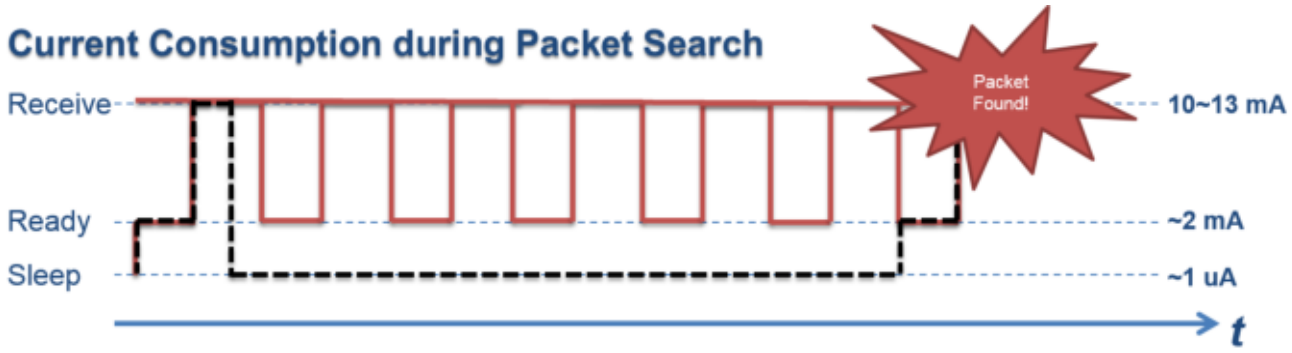


Figure 12. Preamble Sense Mode

Table 14. Data Rates*

	Data Rate				
	1.2 kbps	**9.6 kbps**	**50 kbps**	**100 kbps**	
PM length = 4 bytes	6.48	6.84	8.44	10.43	mA
PM length = 8 bytes	3.83	3.96	4.57	5.33	mA
***Note:** Typical values. Active RX current is 13 mA.					

9. Wireless MBUS support

Wireless MBus is a widely accepted standard for smart meter communication in Europe. The radio supports all WMBus modes per the latest draft specification of the EN13757-4 standard. This includes a much wider deviation error tolerance of ±30% and frequency error tolerance of ±4 kHz, short preamble support (16-bit preamble for 2 and 4 level FSK modes), 3-of-6 encoding and decoding and 169 MHz N modes including N2g.

In addition, Silicon Labs has a production ready WMBus stack available at no additional cost which supports all modes and runs on the EFM32 (32-bit ARM) family of energy friendly microcontrollers. This stack and complete documentation including PHY configuration and test results are available for download from the EZRadioPRO page on the Silicon Labs web site.

10. ETSI EN300 220 Category 1

The radio is capable of supporting ETSI Category 1 applications (social alarms, healthcare applications, etc.) in the 169 MHz and 868 MHz bands. Blocking performance is improved at the 2 MHz and 10 MHz offsets allowing for additional margin from the regulatory limits. The radio complies with ACS limits at the 25 kHz offset in both, 169 MHz and 868 MHz bands. In the 169 MHz band, there is no need for an external SAW filter for 2 MHz and 10 MHz blocking resulting in a lower system cost. In the 868 MHz band, an external SAW filter is still required to meet the Cat 1 blocking limits. An RF Pico board is available for evaluation specifically for ETSI Cat 1 applications.

Test conditions for ETSI Cat 1 specifications are different from the typical conditions and are stated below.

Data Rate: 3 kbps

Deviation: 2 kHz

Modulation: 2 GFSK

IF mode: Fixed and/or Scaled IF

RX bandwidth: 13 kHz

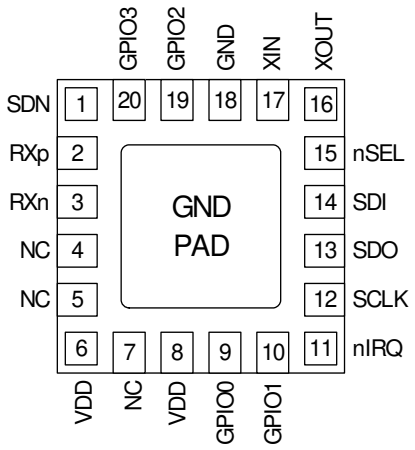
BER target: 0.1%

Blocker signal: CW

	ETSI Cat 1 limits	169 MHz band (no SAW)	868 MHz band (no SAW)
±25 kHz ACS	54 dB	62 dB	58 dB
±2 MHz blocking	84 dB	88 dB	76 dB
±10 MHz blocking	84 dB	90 dB	82 dB
RX sensitivity	–107 dB	–108 dB	–108 dB
Spurious response	35 dB	40 dB	40 dB

For further details on configuring the radio for ETSI Cat 1 applications, refer to the application notes available on the Silicon Labs web site.

11. Pin Descriptions: Si4362-C



Pin	Pin Name	I/0	Description
1	SDN	I	**Shutdown Input Pin**. 0–VDD V digital input. SDN should be = 0 in all modes except Shutdown mode. When SDN = 1, the chip will be completely shut down, and the contents of the registers will be lost.
2	RXp	I	**Differential RF Input Pins of the LNA.** See application schematic for example matching network.
3	RXn	I	
4	NC		No Connect. Not connected internally to any circuitry.
5	NC		It is recommended to connect this pin to GND per the reference design schematic. Not connected internally to any circuitry.
6	VDD	VDD	**+1.8 to +3.8 V Supply Voltage Input to Internal Regulators.** The recommended VDD supply voltage is +3.3 V.
7	NC		No Connect. Not connected internally to any circuitry.
8	VDD	VDD	**+1.8 to +3.8 V Supply Voltage Input to Internal Regulators.** The recommended VDD supply voltage is +3.3 V.
9	GPIO0	I/O	**General Purpose Digital I/O.** May be configured through the registers to perform various functions including: Microcontroller Clock Output, FIFO status, POR, Wake-Up timer, Low Battery Detect, AntDiversity control, etc.
10	GPIO1	I/O	
11	nIRQ	O	**General Microcontroller Interrupt Status Output.** When the Si4362 exhibits any one of the interrupt events, the nIRQ pin will be set low = 0. The Microcontroller can then determine the state of the interrupt by reading the interrupt status. No external resistor pull-up is required, but it may be desirable if multiple interrupt lines are connected.

Pin	Pin Name	I/0	Description
12	SCLK	I	**Serial Clock Input.** 0–VDD V digital input. This pin provides the serial data clock function for the 4-line serial data bus. Data is clocked into the Si4362 on positive edge transitions.
13	SDO	O	**0–VDD V Digital Output.** Provides a serial readback function of the internal control registers.
14	SDI	I	**Serial Data Input.** 0–VDD V digital input. This pin provides the serial data stream for the 4-line serial data bus.
15	nSEL	I	**Serial Interface Select Input.** 0–VDD V digital input. This pin provides the Select/Enable function for the 4-line serial data bus.
16	XOUT	O	**Crystal Oscillator Output.** Connect to an external 25 to 32 MHz crystal, or leave floating when driving with an external source on XIN.
17	XIN	I	**Crystal Oscillator Input.** Connect to an external 25 to 32 MHz crystal, or connect to an external source.
18	GND	GND	When using a XTAL, leave floating per the reference design schematic. When using a TCXO, connect to TCXO GND which should be separate from the board's reference ground plane.
19	GPIO2	I/O	**General Purpose Digital I/O.** May be configured through the registers to perform various functions, including Microcontroller Clock Output, FIFO status, POR, Wake-Up timer, Low Battery Detect, AntDiversity control, etc.
20	GPIO3	I/O	
PKG	PADDLE_GND	GND	The exposed metal pad on the bottom of the Si4362 supplies the RF and circuit ground(s) for the entire chip. It is very important that a good solder connection is made between this exposed metal pad and the ground plane of the PCB underlying the Si4362.

12. Ordering Information

Part Number*	Description	Package Type	Operating Temperature
Si4362-C2A-GM	ISM EZRadioPRO Receiver	QFN-20 Pb-free	–40 to 85 °C

***Note:** Add an “(R)” at the end of the device part number to denote tape and reel option.

13. Package Outline: Si4362

Figure 13 illustrates the package details for the Si4362. Table 15 lists the values for the dimensions shown in the illustration.

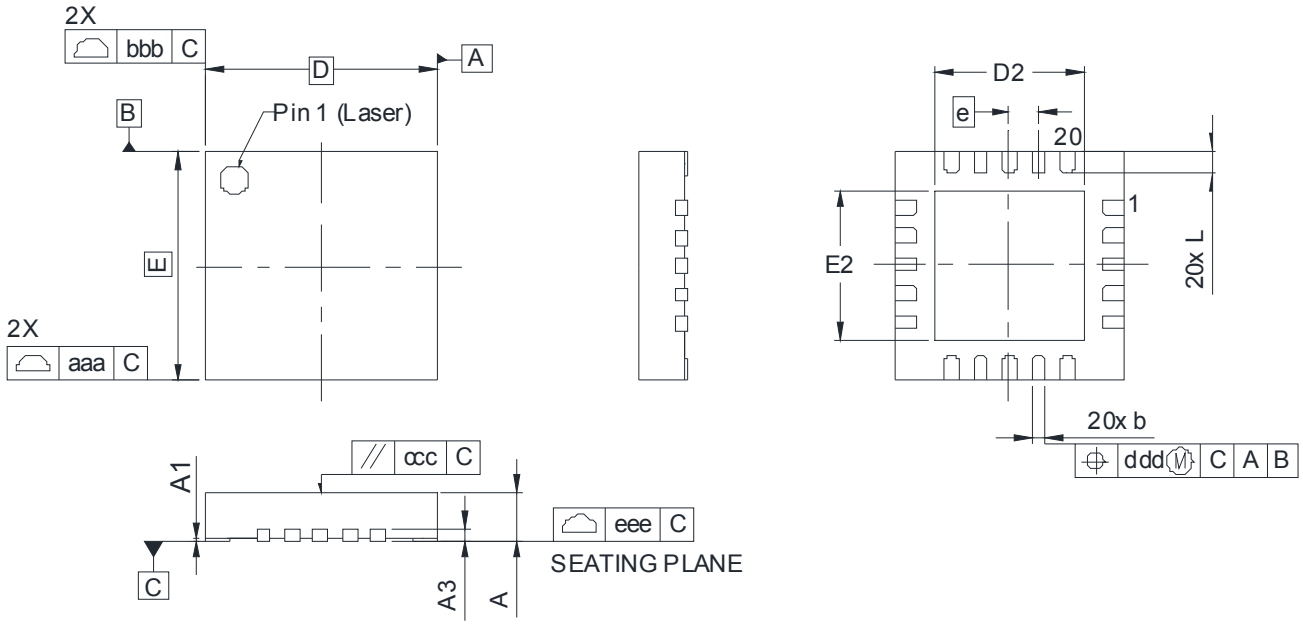


Figure 13. 20-Pin Quad Flat No-Lead (QFN)

Table 15. Package Dimensions

Dimension	Min	Nom	Max
A	0.80	0.85	0.90
A1	0.00	0.02	0.05
A3		0.20 REF	
b	0.18	0.25	0.30
D		4.00 BSC	
D2	2.45	2.60	2.75
e		0.50 BSC	
E		4.00 BSC	
E2	2.45	2.60	2.75
L	0.30	0.40	0.50
aaa		0.15	
bbb		0.15	
ccc		0.10	
ddd		0.10	
eee		0.08	

Notes:

1. All dimensions are shown in millimeters (mm) unless otherwise noted.
2. Dimensioning and tolerancing per ANSI Y14.5M-1994.
3. This drawing conforms to the JEDEC Solid State Outline MO-220, Variation VGGD-8.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

14. PCB Land Pattern: Si4362

Figure 14 illustrates the PCB land pattern details for the Si4362. Table 16 lists the values for the dimensions shown in the illustration.

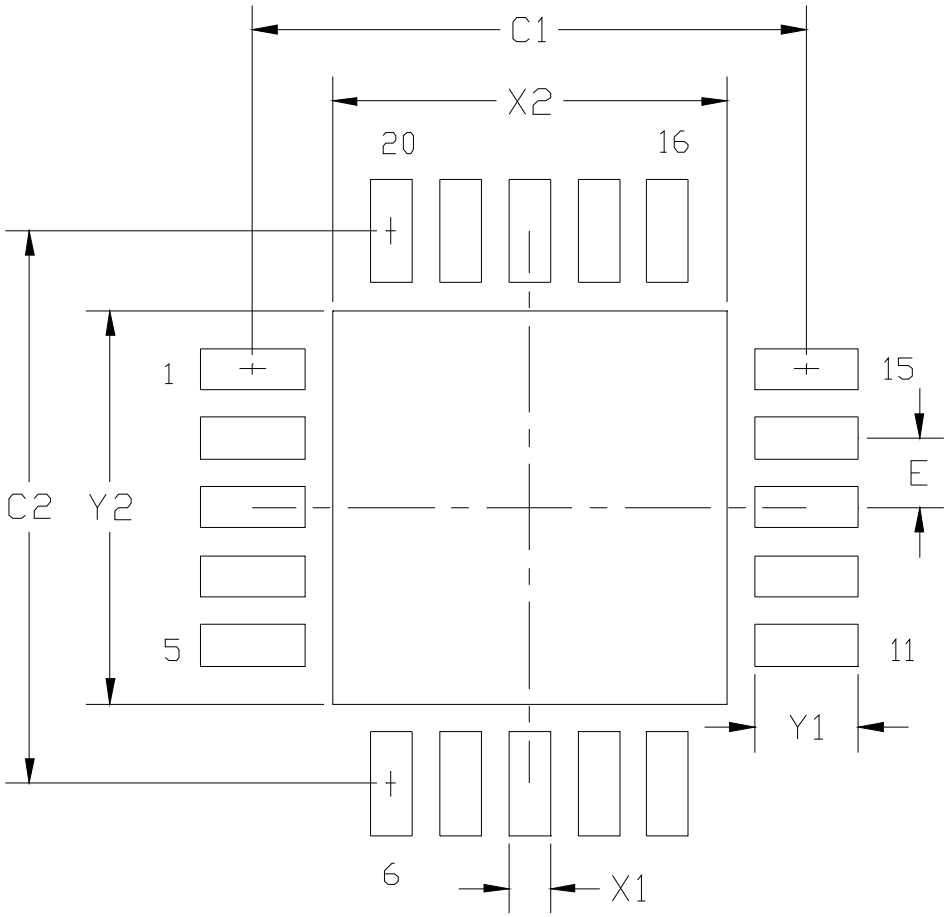


Figure 14. PCB Land Pattern

Table 16. PCB Land Pattern Dimensions

Symbol	Millimeters	
	Min	Max
C1	3.90	4.00
C2	3.90	4.00
E	0.50 REF	
X1	0.20	0.30
X2	2.55	2.65
Y1	0.65	0.75
Y2	2.55	2.65

Notes:

General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This land pattern design is based on IPC-7351 guidelines.

Solder Mask Design

3. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 µm minimum, all the way around the pad.

Stencil Design

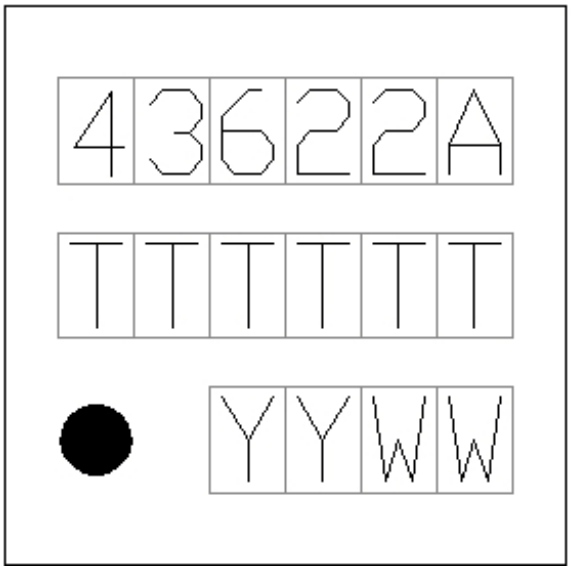
4. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
5. The stencil thickness should be 0.125 mm (5 mils).
6. The ratio of stencil aperture to land pad size should be 1:1 for the perimeter pads.
7. A 2x2 array of 1.10 x 1.10 mm openings on 1.30 mm pitch should be used for the center ground pad.

Card Assembly

8. A No-Clean, Type-3 solder paste is recommended.
9. The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for small body components.

15. Top Marking

15.1. Si4362 Top Marking



15.2. Top Marking Explanation

Mark Method	YAG Laser	
Line 1 Marking	Part Number	43622A = Si4362 Rev 2A[1]
Line 2 Marking	TTTTTT = Internal Code	Internal tracking code.[2]
Line 3 Marking	YY = Year WW = Workweek	Assigned by the Assembly House. Corresponds to the last significant digit of the year and workweek of the mold date.

Notes:
1. The first letter after the part number is part of the ROM revision. The last letter indicates the firmware revision.
2. The first letter of this line is part of the ROM revision.



Simplicity Studio

One-click access to MCU tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

www.silabs.com/simplicity

MCU Portfolio
www.silabs.com/mcu

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

Disclaimer
Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information
Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOmodem ®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.

SILICON LABS®

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com