6021Z

Technical description

This product is a thermostat with wireless control. It is a transceiver operating at 2.4GHz ISM band. The Thermostat is powered by 2 × "AAA", 2 × "C" and output 6VDC adaptor. The Wall Unit is powered by 2 × AAA and output 24VAC adaptor.
The 2.4GHz RF transceiver function is provided by EM250 chips (MD1/ MM1). The RX architecture of the chip is based on a low-IF, superheterodyne receiver. It utilizes differential signal paths to minimize noise interference. The input RF signal is mixed down to the IF frequency of 4MHz by I and Q mixers. The output of the mixers is filtered and combined prior to being sampled by a 12Msps ADC. The EM250 transmitter utilizes both analog circuitry and digital logic to produce the O-QPSK modulated signal. The area-efficient TX architecture directly modulates the spread symbols prior to transmission. The differential signal paths increase noise immunity and provide a common interface for the external balun.

For the Wall Unit, the received data will be processed by RF IC (MD1) and the sampled data will transfer the relay driver IC (U3) that will control the relays ON/ OFF. At the same time, the RC IC (MD1) will transmit a confirmation signal to the corresponding Thermostat.
For the Thermostat, the thermo-sensor (TC1) will monitor the temperature. When the target temperature is reached, the MCU (U1) will send the command with RF IC (MM1) and get back confirmation that the command was received and implemented by the corresponding Wall Unit.

The external components of the RF path consist of a Balun circuit (composed of L1, L2, L3, C1, C2, C3), which is a 50/200 Ohm conversion which stabilizes the load observed by the internal PA across the 2.4GHz ISM band, a harmonic filter circuit (composed of L4, C6, C7), and an antenna.

GOLDEN POWER MANUFACTURING LTD.

Model: 6021Z
The Channel frequency of 802.15.4 (ZigBee) as below:

Channel	**Frequency (MHz)**
11	2405
12	2410
13	2415
14	2420
15	2425
16	2430
17	2435
18	2440
19	2445
20	2450
21	2455
22	2460
23	2465
24	2470
25	2475
26	2480

ember

EM250

Single-Chip ZigBee/802.15.4 Solution

Integrated 2.4GHz, IEEE 802.15.4-compliant transceiver:

- Robust RX filtering allows co-existence with IEEE 802.11g and Bluetooth devices
- - 99dBm RX sensitivity (1% PER, 20byte packet)
- + 3dBm nominal output power
- Increased radio performance mode (boost mode) gives - 100dBm sensitivity and + 5dBm transmit power
- Integrated VCO and loop filter

Integrated IEEE 802.15.4 PHY and lower MAC with DMA

Integrated hardware support for Packet Trace Inter-face for InSight Development Environment

Provides integrated RC oscillator for low power operation

Supports optional 32.768kHz crystal oscillator for higher accuracy needs

16-bit XAP2b microprocessor

Integrated memory:

- 128kB of Flash
- 5kB of SRAM

Configurable memory protection scheme

Two sleep modes:

- Processor idle
- Deep sleep—1.0μA (1.5μA with optional 32.768kHz oscillator enabled)

Seventeen GPIO pins with alternate functions

Two Serial Controllers with DMA

- SC1: I^2C master, SPI master + UART
- SC2: I^2C master, SPI master/slave

Two 16-bit general-purpose timers; one 16-bit sleep timer

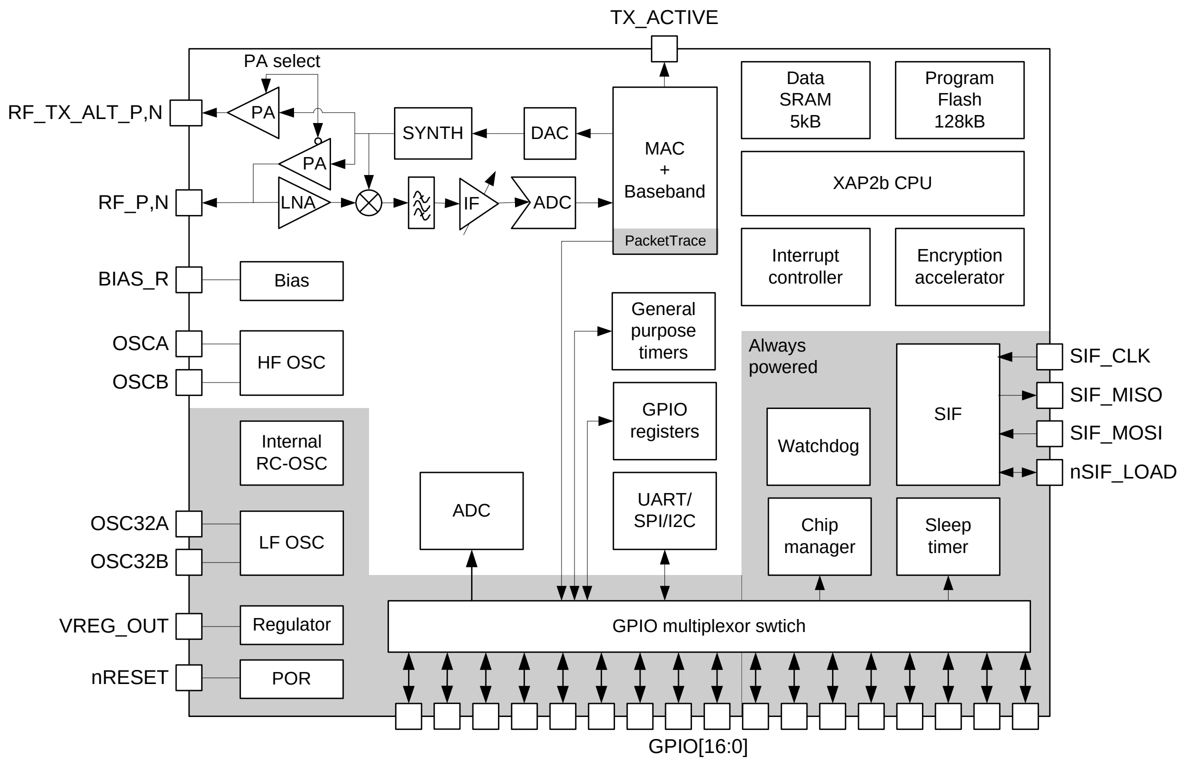
Watchdog timer and power-on-reset circuitry

Non-intrusive debug interface (SIF)

Integrated AES encryption accelerator

Integrated ADC module first-order, sigma-delta converter with 12-bit resolution

Integrated 1.8V voltage regulator



Ember Corporation
343 Congress Street
Boston MA 02210 USA
+1 617.951.0200
www.ember.com

120-0082-000M
June 29, 2007

General Description

The EM250 is a single-chip solution that integrates a 2.4GHz, IEEE 802.15.4-compliant transceiver with a 16-bit XAP2b microprocessor. It contains integrated Flash and RAM memory and peripherals of use to designers of ZigBee-based applications.

The transceiver utilizes an efficient architecture that exceeds the dynamic range requirements imposed by the IEEE 802.15.4-2003 standard by over 15dB. The integrated receive channel filtering allows for co-existence with other communication standards in the 2.4GHz spectrum such as IEEE 802.11g and Bluetooth. The integrated regulator, VCO, loop filter, and power amplifier keep the external component count low. An optional high performance radio mode (boost mode) is software selectable to boost dynamic range by a further 3dB.

The XAP2b microprocessor is a power-optimized core integrated in the EM250. It supports two different modes of operation—System Mode and Application Mode. The EmberZNet stack runs in System Mode with full access to all areas of the chip. Application code runs in Application Mode with limited access to the EM250 resources; this allows for the scheduling of events by the application developer while preventing modification of restricted areas of memory and registers. This architecture results in increased stability and reliability of deployed solutions.

The EM250 has 128kB of embedded Flash memory and 5kB of integrated RAM for data and program storage. The EM250 software stack employs an effective wear-leveling algorithm in order to optimize the lifetime of the embedded Flash.

To maintain the strict timing requirements imposed by ZigBee and the IEEE 802.15.4-2003 standard, the EM250 integrates a number of MAC functions into the hardware. The MAC hardware handles automatic ACK transmission and reception, automatic backoff delay, and clear channel assessment for transmission, as well as automatic filtering of received packets. In addition, the EM250 allows for true MAC level debugging by integrating the Packet Trace Interface.

To support user-defined applications, a number of peripherals such as GPIO, UART, SPI, I^2C, ADC, and general-purpose timers are integrated. Also, an integrated voltage regulator, power-on-reset circuitry, sleep timer, and low-power sleep modes are available. The deep sleep mode draws less than 1µA, allowing products to achieve long battery life.

Finally, the EM250 utilizes the non-intrusive SIF module for powerful software debugging and programming of the XAP2b microcontroller.

Target applications for the EM250 include:

- *Building automation and control*
- *Home automation and control*
- *Home entertainment control*
- *Asset tracking*

The EM250 is purchased with EmberZNet, the Ember ZigBee-compliant software stack, providing a ZigBee profile-ready, platform-compliant solution. This technical datasheet details the EM250 features available to customers using it with the EmberZNet stack.

Contents

1 Pin Assignment

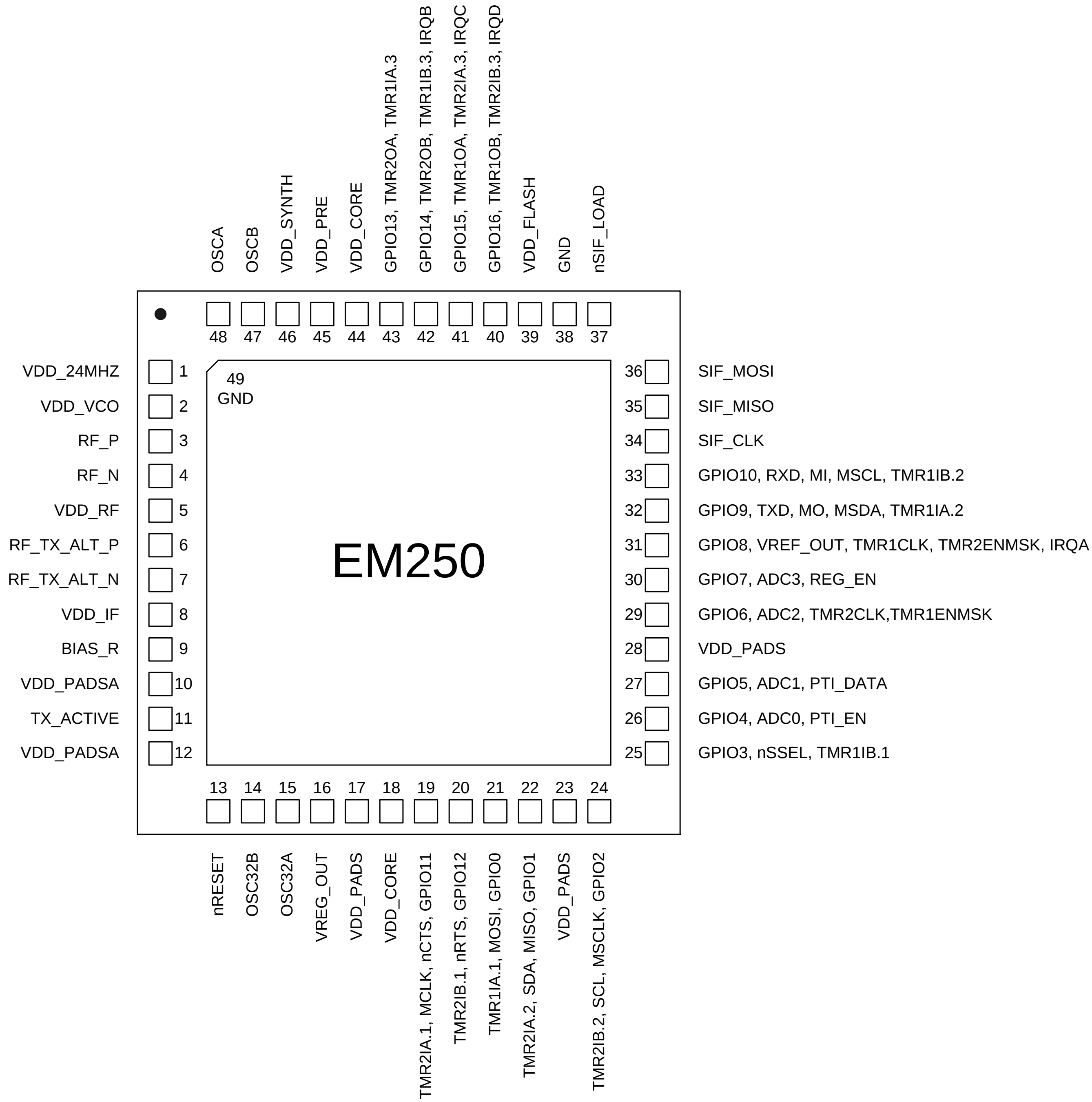


Figure 1. EM250 Pin Assignment

Refer to Table 17 and Table 18 for selecting alternate pin functions.

Table 1. Pin Descriptions

Pin #	Signal	Direction	Description
1	VDD_24MHZ	Power	1.8V high-frequency oscillator supply
2	VDD_VCO	Power	1.8V VCO supply
3	RF_P	I/O	Differential (with RF_N) receiver input/transmitter output
4	RF_N	I/O	Differential (with RF_P) receiver input/transmitter output
5	VDD_RF	Power	1.8V RF supply (LNA and PA)
6	RF_TX_ALT_P	O	Differential (with RF_TX_ALT_N) transmitter output (optional)
7	RF_TX_ALT_N	O	Differential (with RF_TX_ALT_P) transmitter output (optional)
8	VDD_IF	Power	1.8V IF supply (mixers and filters)
9	BIAS_R	I	Bias setting resistor
10	VDD_PADSA	Power	Analog pad supply (1.8V)
11	TX_ACTIVE	O	Logic-level control for external RX/TX switch The EM250 baseband controls TX_ACTIVE and drives it high (1.8V) when in TX mode. (Refer to Table 6 and section 4.2.2.)
12	VDD_PADSA	Power	Analog pad supply (1.8V)
13	nRESET	I	Active low chip reset (internal pull-up)
14	OSC32B	I/O	32.768kHz crystal oscillator. This pin should be left open when using external clock on OSC32A or when using the internal RC Oscillator
15	OSC32A	I/O	32.768kHz crystal oscillator or digital clock input. This pin can be left open when using the internal RC Oscillator.
16	VREG_OUT	Power	Regulator output (1.8V)
17	VDD_PADS	Power	Pads supply (2.1-3.6V)
18	VDD_CORE	Power	1.8V digital core supply
19	GPIO11	I/O	Digital I/O Enable GPIO11 with `GPIO_CFG[7:4]`
	nCTS	I	UART CTS handshake of Serial Controller SC1 Enable SC1-4A with `GPIO_CFG[7:4]`, select UART with `SC1_MODE`
	MCLK	O	SPI master clock of Serial Controller SC1 Enable SC1-3M with `GPIO_CFG[7:4]`, select SPI with `SC1_MODE`, enable master with `SC1_SPICFG[4]`
	TMR2IA.1	I	Capture Input A of Timer 2 Enable CAP2-0 with `GPIO_CFG[7:4]`
20	GPIO12	I/O	Digital I/O Enable GPIO12 with `GPIO_CFG[7:4]`
	nRTS	O	UART RTS handshake of Serial Controller SC1 Enable SC1-4A with `GPIO_CFG[7:4]`, select UART with `SC1_MODE`
	TMR2IB.1	I	Capture Input B for Timer 2 Enable CAP2-0 with `GPIO_CFG[7:4]`

Pin #	Signal	Direction	Description
21	GPIO0	I/O	Digital I/O Enable GPIO0 with GPIO_CFG[7:4]
	MOSI	O	SPI master data out of Serial Controller SC2 Enable SC2-3M with GPIO_CFG[7:4], select SPI with SC2_MODE, enable master with SC2_SPICFG[4]
	MOSI	I	SPI slave data in of Serial Controller SC2 Enable SC2-4S with GPIO_CFG[7:4], select SPI with SC2_MODE, enable slave with SC2_SPICFG[4]
	TMR1IA.1	I	Capture Input A of Timer 1 Enable CAP1-0 with GPIO_CFG[7:4]
22	GPIO1	I/O	Digital I/O Enable GPIO1 with GPIO_CFG[7:4]
	MISO	I	SPI master data in of Serial Controller SC2 Enable SC2-3M with GPIO_CFG[7:4], select SPI with SC2_MODE, enable master with SC2_SPICFG[4]
	MISO	O	SPI slave data out of Serial Controller SC2 Enable SC2-4S with GPIO_CFG[7:4], select SPI with SC2_MODE, enable slave with SC2_SPICFG[4]
	SDA	I/O	I^2C data of Serial Controller SC2 Enable SC2-2 with GPIO_CFG[7:4], select I^2C with SC2_MODE
	TMR2IA.2	I	Capture Input A of Timer 2 Enable CAP2-1 with GPIO_CFG[7:4]
23	VDD_PADS	Power	Pads supply (2.1-3.6V)
24	GPIO2	I/O	Digital I/O Enable GPIO2 with GPIO_CFG[7:4]
	MSCLK	O	SPI master clock of Serial Controller SC2 Enable SC2-3M with GPIO_CFG[7:4], select SPI with SC2_MODE, enable master with SC2_SPICFG[4]
	MSCLK	I	SPI slave clock of Serial Controller SC2 Enable SC2-4S with GPIO_CFG[7:4], select SPI with SC2_MODE, enable slave with SC2_SPICFG[4]
	SCL	I/O	I^2C clock of Serial Controller SC2 Enable SC2-2 with GPIO_CFG[7:4], select I^2C with SC2_MODE
	TMR2IB.2	I	Capture Input B of Timer 2 Enable CAP2-1 with GPIO_CFG[7:4]
25	GPIO3	I/O	Digital I/O Enable GPIO3 with GPIO_CFG[7:4]
	nSSEL	I	SPI slave select of Serial Controller SC2 Enable SC2-4S with GPIO_CFG[7:4], select SPI with SC2_MODE, enable slave with SC2_SPICFG[4]
	TMR1IB.1	I	Capture Input B of Timer 1 Enable CAP1-0 with GPIO_CFG[7:4]

Pin #	Signal	Direction	Description
26	GPIO4	I/O	Digital I/O Enable GPIO4 with GPIO_CFG[12] and GPIO_CFG[8]
	ADC0	Analog	ADC Input 0 Enable ADC0 with GPIO_CFG[12] and GPIO_CFG[8]
	PTI_EN	O	Frame signal of Packet Trace Interface (PTI) Enable PTI with GPIO_CFG[12]
27	GPIO5	I/O	Digital I/O Enable GPIO5 with GPIO_CFG[12] and GPIO_CFG[9]
	ADC1	Analog	ADC Input 1 Enable ADC1 with GPIO_CFG[12] and GPIO_CFG[9]
	PTI_DATA	O	Data signal of Packet Trace Interface (PTI) Enable PTI with GPIO_CFG[12]
28	VDD_PADS	Power	Pads supply (2.1-3.6V)
29	GPIO6	I/O	Digital I/O Enable GPIO6 with GPIO_CFG[10]
	ADC2	Analog	ADC Input 2 Enable ADC2 with GPIO_CFG[10]
	TMR2CLK	I	External clock input of Timer 2
	TMR1ENMSK	I	External enable mask of Timer 1
30	GPIO7	I/O	Digital I/O Enable GPIO7 with GPIO_CFG[13] and GPIO_CFG[11]
	ADC3	Analog	ADC Input 3 Enable ADC3 with GPIO_CFG[13] and GPIO_CFG[11]
	REG_EN	O	External regulator open collector output Enable REG_EN with GPIO_CFG[13]
31	GPIO8	I/O	Digital I/O Enable GPIO8 with GPIO_CFG[14]
	VREF_OUT	Analog	ADC reference output Enable VREF_OUT with GPIO_CFG[14]
	TMR1CLK	I	External clock input of Timer 1
	TMR2ENMSK	I	External enable mask of Timer 2
	IRQA	I	External interrupt source A

Pin #	Signal	Direction	Description
32	GPIO9	I/O	Digital I/O Enable GPIO9 with GPIO_CFG[7:4]
	TXD	O	UART transmit data of Serial Controller SC1 Enable SC1-4A or SC1-2 with GPIO_CFG[7:4], select UART with SC1_MODE
	MO	O	SPI master data out of Serial Controller SC1 Enable SC1-3M with GPIO_CFG[7:4], select SPI with SC1_MODE, enable master with SC1_SPICFG[4]
	MSDA	I/O	I^2C data of Serial Controller SC1 Enable SC1-2 with GPIO_CFG[7:4], select I^2C with SC1_MODE
	TMR1IA.2	I	Capture Input A of Timer 1 Enable CAP1-1 or CAP1-1h with GPIO_CFG[7:4]
33	GPIO10	I/O	Digital I/O Enable GPIO10 with GPIO_CFG[7:4]
	RXD	I	UART receive data of Serial Controller SC1 Enable SC1-4A or SC1-2 with GPIO_CFG[7:4], select UART with SC1_MODE
	MI	I	SPI master data in of Serial Controller SC1 Enable SC1-3M with GPIO_CFG[7:4], select SPI with SC1_MODE, enable master with SC1_SPICFG[4]
	MSCL	I/O	I^2C clock of Serial Controller SC1 Enable SC1-2 with GPIO_CFG[7:4], select I^2C with SC1_MODE
	TMR1IB.2	I	Capture Input B of Timer 1 Enable CAP1-1 with GPIO_CFG[7:4]
34	SIF_CLK	I	Serial interface, clock (internal pull-down)
35	SIF_MISO	O	Serial interface, master in/slave out
36	SIF_MOSI	I	Serial interface, master out/slave in
37	nSIF_LOAD	I/O	Serial interface, load strobe (open-collector with internal pull-up)
38	GND	Power	Ground supply
39	VDD_FLASH	Power	1.8V Flash memory supply
40	GPIO16	I/O	Digital I/O Enable GPIO16 with GPIO_CFG[3]
	TMR1OB	O	Waveform Output B of Timer 1 Enable TMR1OB with GPIO_CFG[3]
	TMR2IB.3	I	Capture Input B of Timer 2 Enable CAP2-2 with GPIO_CFG[7:4]
	IRQD	I	External interrupt source D
41	GPIO15	I/O	Digital I/O Enable GPIO15 with GPIO_CFG[2]
	TMR1OA	O	Waveform Output A of Timer 1 Enable TMR1OA with GPIO_CFG[2]
	TMR2IA.3	I	Capture Input A of Timer 2 Enable CAP2-2 with GPIO_CFG[7:4]
	IRQC	I	External interrupt source C

Pin #	Signal	Direction	Description
42	GPIO14	I/O	Digital I/O Enable GPIO14 with GPIO_CFG[1]
	TMR2OB	O	Waveform Output B of Timer 2 Enable TMR2OB with GPIO_CFG[1]
	TMR1IB.3	I	Capture Input B of Timer 1 Enable CAP1-2 with *GPIO_CFG[7:4]*
	IRQB	I	External interrupt source B
43	GPIO13	I/O	Digital I/O Enable GPIO13 with GPIO_CFG[0]
	TMR2OA	O	Waveform Output A of Timer 2 Enable TMR2OA with GPIO_CFG[0]
	TMR1IA.3	I	Capture Input A of Timer 1 Enable CAP1-2 or CAP1-2h with GPIO_CFG[7:4]
44	VDD_CORE	Power	1.8V digital core supply
45	VDD_PRE	Power	1.8V prescaler supply
46	VDD_SYNTH	Power	1.8V synthesizer supply
47	OSCB	I/O	24MHz crystal oscillator or left open when using external clock input on OSCA
48	OSCA	I/O	24MHz crystal oscillator or external clock input
49	GND	Ground	Ground supply pad in the bottom center of the package forms Pin 49 (see the *EM250 Reference Design* for PCB considerations)

2 Top-Level Functional Description

Figure 2 shows a detailed block diagram of the EM250.

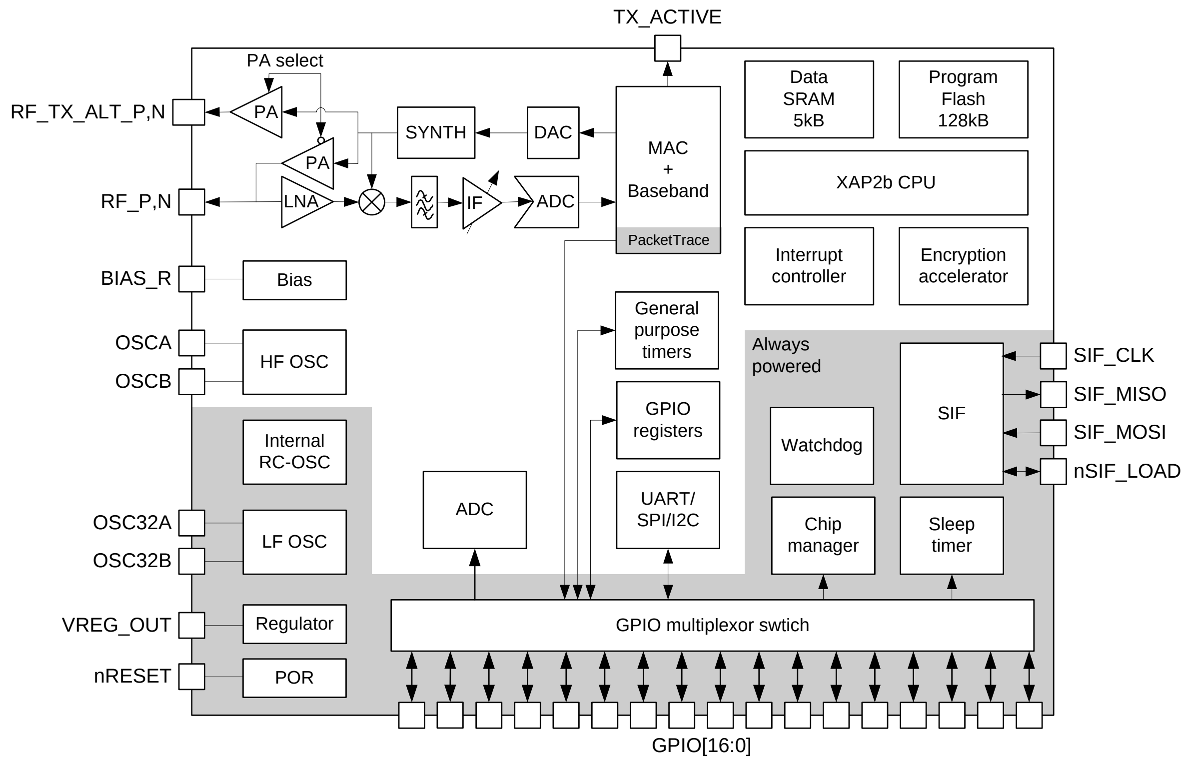


Figure 2. EM250 Block Diagram

The radio receiver is a low-IF, super-heterodyne receiver. It utilizes differential signal paths to minimize noise interference, and its architecture has been chosen to optimize co-existence with other devices within the 2.4GHz band (namely, IEEE 802.11g and Bluetooth). After amplification and mixing, the signal is filtered and combined prior to being sampled by an ADC.

The digital receiver implements a coherent demodulator to generate a chip stream for the hardware-based MAC. In addition, the digital receiver contains the analog radio calibration routines and control of the gain within the receiver path.

The radio transmitter utilizes an efficient architecture in which the data stream directly modulates the VCO. An integrated PA boosts the output power. The calibration of the TX path as well as the output power is controlled by digital logic. If the EM250 is to be used with an external PA, the TX_ACTIVE signal should be used to control the timing of the external switching logic.

The integrated 4.8 GHz VCO and loop filter minimize off-chip circuitry. Only a 24MHz crystal with its loading capacitors is required to properly establish the PLL reference signal.

The MAC interfaces the data memory to the RX and TX baseband modules. The MAC provides hardware-based IEEE 802.15.4 packet-level filtering. It supplies an accurate symbol time base that minimizes the synchronization effort of the software stack and meets the protocol timing requirements. In addition, it provides timer and synchronization assistance for the IEEE 802.15.4 CSMA-CA algorithm.

The EM250 integrates hardware support for a Packet Trace module, which allows robust packet-based debug. This element is a critical component of InSight Desktop, the Ember software IDE, providing advanced network debug capability when coupled with the InSight Adapter.

The EM250 integrates a 16-bit XAP2b microprocessor developed by Cambridge Consultants Ltd. This power-efficient, industry-proven core provides the appropriate level of processing power to meet the needs of ZigBee applications. In addition, 128kB of Flash and 5kB of SRAM comprise the program and data memory elements, respectively. The EM250 employs a configurable memory protection scheme usually found on larger microcontrollers. In addition, the SIF module provides a non-intrusive programming and debug interface allowing for real-time application debugging.

The EM250 contains 17 GPIO pins shared with other peripheral (or alternate) functions. Flexible routing within the EM250 lets external devices utilize the alternate functions on a variety of different GPIOs. The integrated Serial Controller SC1 can be configured for SPI (master-only), I2C (master-only), or UART functionality, and the Serial Controller SC2 can be configured for SPI (master or slave) or I2C (master-only) operation.

The EM250 has an ADC integrated which can sample analog signals from four GPIO pins single-ended or differentially. In addition, the unregulated voltage supply VDD_PADS, regulated supply VDD_PADSA, voltage reference VREF, and GND can be sampled. The integrated voltage reference VREF for the ADC can be made available to external circuitry.

The integrated voltage regulator generates a regulated 1.8V reference voltage from an unregulated supply voltage. This voltage is decoupled and routed externally to supply the 1.8V to the core logic. In addition, an integrated POR module allows for the proper cold start of the EM250.

The EM250 contains one high-frequency (24MHz) crystal oscillator and, for low-power operation, a second low-frequency oscillator (either an internal 10kHz RC oscillator or an external 32.768kHz crystal oscillator).

The EM250 contains two power domains. The always-powered High Voltage Supply is used for powering the GPIO pads and critical chip functions. The rest of the chip is powered by a regulated Low Voltage Supply which can be disabled during deep sleep to reduce the power consumption.

3 Electrical Characteristics

3.1 Absolute Maximum Ratings

Table 2 lists the absolute maximum ratings for the EM250.

Table 2. Absolute Maximum Ratings

Parameter	Test Conditions	Min.	Max.	Unit
Regulator voltage (VDD_PADS)		- 0.3	3.6	V
Core voltage (VDD_24MHZ, VDD_VCO, VDD_RF, VDD_IF, VDD_PADSA, VDD_FLASH, VDD_PRE, VDD_SYNTH, VDD_CORE)		- 0.3	2.0	V
Voltage on RF_P,N; RF_TX_ALT_P,N		- 0.3	3.6	V
Voltage on any GPIO[16:0], SIF_CLK, SIF_MISO, SIF_MOSI, nSIF_LOAD, OSC32A, OSC32B, nRESET, VREG_OUT		- 0.3	VDD_PADS+ 0.3	V
Voltage on TX_ACTIVE, BIAS_R, OSCA, OSCB		- 0.3	VDD_CORE+ 0.3	V
Storage temperature		- 40	+ 140	°C

3.2 Recommended Operating Conditions

Table 3 lists the rated operating conditions of the EM250.

Table 3. Operating Conditions

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Regulator input voltage (VDD_PADS)		2.1		3.6	V
Core input voltage (VDD_24MHZ, VDD_VCO, VDD_RF, VDD_IF, VDD_PADSA, VDD_FLASH, VDD_PRE, VDD_SYNTH, VDD_CORE)		1.7	1.8	1.9	V
Temperature range		- 40		+ 85	°C

3.3 Environmental Characteristics

Table 4 lists the environmental characteristics of the EM250.

Table 4. Environmental Characteristics

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
ESD (human body model)	On any Pin	- 2		+ 2	kV
ESD (charged device model)	Non-RF Pins	- 400		+ 400	V
ESD (charged device model)	RF Pins	- 225		+ 225	V
Moisture Sensitivity Level (MSL)			MSL3		

3.4 DC Electrical Characteristics

Table 5 lists the DC electrical characteristics of the EM250.

Note: Current Measurements were collected using the EmberZNet software stack Version 3.0.1.

Table 5. DC Characteristics

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Regulator input voltage (VDD_PADS)		2.1		3.6	V
Power supply range (VDD_CORE)	Regulator output or external input	1.7	1.8	1.9	V
Deep Sleep Current					
Quiescent current, including internal RC oscillator	At 25° C.			1.0	µA
Quiescent current, including 32.768kHz oscillator	At 25° C.			1.5	µA
RX Current					
Radio receiver, MAC, and baseband (boost mode)			30.0		mA
Radio receiver, MAC, and baseband			28.0		mA
CPU, RAM, and Flash memory	At 25° C and 1.8V core		8.0		mA
Total RX current (= $I_{\text{Radio receiver, MAC and baseband, CPU}}$+ $I_{\text{RAM, and Flash memory}}$)	At 25° C, VDD_PADS=3.0V		36.0		mA
TX Current					
Radio transmitter, MAC, and baseband (boost mode)	At max. TX power (+ 5dBm typical)		34.0		mA
Radio transmitter, MAC, and baseband	At max. TX power (+ 3dBm typical)		28.0		mA
	At 0 dBm typical		24.0		mA
	At min. TX power (-32dBm typical)		19.0		mA
CPU, RAM, and Flash memory	At 25° C, VDD_PADS=3.0V		8.0		mA
Total TX current (= $I_{\text{Radio transmitter, MAC and baseband, CPU}}$ + $I_{\text{RAM, and Flash memory}}$)	At 25° C and 1.8V core; max. power out		36.0		mA

Table 6 contains the digital I/O specifications for the EM250. The digital I/O power (named VDD_PADS) comes from three dedicated pins (Pins 17, 23, and 28). The voltage applied to these pins sets the I/O voltage.

Table 6. Digital I/O Specifications

Parameter	Name	Min.	Typ.	Max.	Unit
Voltage supply	VDD_PADS	2.1		3.6	V
Input voltage for logic 0	V_{IL}	0		0.2 x VDD_PADS	V
Input voltage for logic 1	V_{IH}	0.8 x VDD_PADS		VDD_PADS	V
Input current for logic 0	I_{IL}			- 0.5	µA
Input current for logic 1	I_{IH}			0.5	µA
Input pull-up resistor value	R_{IPU}		30		kΩ
Input pull-down resistor value	R_{IPD}		30		kΩ
Output voltage for logic 0	V_{OL}	0		0.18 x VDD_PADS	V

Parameter	Name	Min.	Typ.	Max.	Unit
Output voltage for logic 1	V_{OH}	0.82 x VDD_PADS		VDD_PADS	V
Output source current (standard current pad)	I_{OHS}			4	mA
Output sink current (standard current pad)	I_{OLS}			4	mA
Output source current high current pad: GPIO[16:13]	I_{OHH}			8	mA
Output sink current high current pad: GPIO[16:13]	I_{OLH}			8	mA
Total output current (for I/O Pads)	$I_{OH} + I_{OL}$			40	mA
Input voltage threshold for OSC32A		0.2 x VDD_PADS		0.8 x VDD_PADS	V
Input voltage threshold for OSCA		0.2 x VDD_CORE		0.8 x VDD_CORE	V
Output voltage level (TX_ACTIVE)		0.18 x VDD_CORE		0.82 x VDD_CORE	V
Output source current (TX_ACTIVE)				1	mA

3.5 RF Electrical Characteristics

3.5.1 Receive

Table 7 lists the key parameters of the integrated IEEE 802.15.4 receiver on the EM250.

Note: Receive Measurements were collected with Ember's EM250 Lattice Balun Reference Design (Version B1) at 2440MHz and using the EmberZNet software stack Version 3.0.1. The Typical number indicates one standard deviation above the mean, measured at room temperature (25C). The Min and Max numbers were measured over process corners at room temperature.

Table 7. Receive Characteristics

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Frequency range		2400		2500	MHz
Sensitivity (boost mode)	1% PER, 20byte packet defined by IEEE 802.15.4		-100	-95	dBm
Sensitivity	1% PER, 20byte packet defined by IEEE 802.15.4		-99	-94	dBm
High-side adjacent channel rejection	IEEE 802.15.4 signal at - 82dBm		35		dB
Low-side adjacent channel rejection	IEEE 802.15.4 signal at - 82dBm		35		dB
2nd high-side adjacent channel rejection	IEEE 802.15.4 signal at - 82dBm		43		dB
2nd low-side adjacent channel rejection	IEEE 802.15.4 signal at - 82dBm		43		dB
Channel rejection for all other channels	IEEE 802.15.4 signal at - 82dBm		40		dB
802.11g rejection centered at + 12MHz or - 13MHz	IEEE 802.15.4 signal at - 82dBm		40		dB
Maximum input signal level for correct operation (low gain)		0			dBm
Image suppression			30		dB

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Co-channel rejection	IEEE 802.15.4 signal at - 82dBm		- 6		dBc
Relative frequency error (2x40 ppm required by IEEE 802.15.4)		- 120		+ 120	ppm
Relative timing error (2x40 ppm required by IEEE 802.15.4)		- 120		+ 120	ppm
Linear RSSI range		40			dB
RSSI Range		-90		-30	dB

3.5.2 Transmit

Table 8 lists the key parameters of the integrated IEEE 802.15.4 transmitter on the EM250.

Note: Transmit Measurements were collected with Ember's EM250 Lattice Balun Reference Design (Version B1) at 2440MHz and using the EmberZNet software stack Version 3.0.1. The Typical number indicates one standard deviation below the mean, measured at room temperature (25C). The Min and Max numbers were measured over process corners at room temperature.

Table 8. Transmit Characteristics

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Maximum output power (boost mode)	At highest power setting		5		dBm
Maximum output power	At highest power setting	0	3		dBm
Minimum output power	At lowest power setting		- 32		dBm
Error vector magnitude	As defined by IEEE 802.15.4, which sets a 35% maximum		5	15	%
Carrier frequency error		- 40		+ 40	ppm
Load impedance			200+j90		Ω
PSD mask relative	3.5MHz away	- 20			dB
PSD mask absolute	3.5MHz away	- 30			dBm

3.5.3 Synthesizer

Table 9 lists the key parameters of the integrated synthesizer on the EM250.

Table 9. Synthesizer Characteristics

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Frequency range		2400		2500	MHz
Frequency resolution			11.7		kHz
Lock time	From off, with correct VCO DAC setting			100	µs
Relock time	Channel change or RX/TX turnaround (IEEE 802.15.4 defines 192µs turnaround time)			100	µs
Phase noise at 100kHz			- 71		dBc/Hz

Phase noise at 1MHz			- 91		dBc/Hz
Phase noise at 4MHz			- 103		dBc/Hz
Phase noise at 10MHz			- 111		dBc/Hz

4 Functional Description—System Modules

The EM250 contains a dual-thread mode of operation—System Mode and Application Mode—to guarantee microcontroller bandwidth to the application developer and protect the developer from errant software access.

During System Mode, all areas including the RF Transceiver, MAC, Packet Trace Interface, Sleep Timer, Power Management Module, Watchdog Timer, and Power on Reset Module are accessible.

Since the EM250 comes with a license to EmberZNet, the Ember ZigBee-compliant software stack, these areas are not available to the application developer in Application Mode. The following brief description of these modules provides the necessary background on the operation of the EM250. For more information, contact www.ember.com/support.

4.1 Receive (RX) Path

The EM250 RX path spans the analog and digital domains. The RX architecture is based on a low-IF, superheterodyne receiver. It utilizes differential signal paths to minimize noise interference. The input RF signal is mixed down to the IF frequency of 4MHz by I and Q mixers. The output of the mixers is filtered and combined prior to being sampled by a 12Msps ADC. The RX filtering within the RX path has been designed to optimize the co-existence of the EM250 with other 2.4GHz transceivers, such as the IEEE 802.11g and Bluetooth.

4.1.1 RX Baseband

The EM250 RX baseband (within the digital domain) implements a coherent demodulator for optimal performance. The baseband demodulates the O-QPSK signal at the chip level and synchronizes with the IEEE 802.15.4-2003 preamble. Once a packet preamble is detected, it de-spreads the demodulated data into 4-bit symbols. These symbols are buffered and passed to the hardware-based MAC module for filtering.

In addition, the RX baseband provides the calibration and control interface to the analog RX modules, including the LNA, RX Baseband Filter, and modulation modules. The EmberZNet software includes calibration algorithms which use this interface to reduce the effects of process and temperature variation.

4.1.2 RSSI and CCA

The EM250 calculates the RSSI over an 8-symbol period as well as at the end of a received packet. It utilizes the RX gain settings and the output level of the ADC within its algorithm. The linear range of RSSI is specified to be 40dB over all temperatures. At room temperature, the linear range is approximately 60dB (-90 dBm to -30dBm).

The EM250 RX baseband provides support for the IEEE 802.15.4-2003 required CCA methods summarized in Table 10. Modes 1, 2, and 3 are defined by the 802.15.4-2003 standard; Mode 0 is a proprietary mode.

Table 10. CCA Mode Behavior

CCA Mode	Mode Behavior
0	Clear channel reports busy medium if either carrier sense OR RSSI exceeds their thresholds.
1	Clear channel reports busy medium if RSSI exceeds its threshold.
2	Clear channel reports busy medium if carrier sense exceeds its threshold.
3	Clear channel reports busy medium if both RSSI AND carrier sense exceed their thresholds.

The EmberZNet Software Stack sets the CCA Mode, and it is not configurable by the Application Layer. For software versions beginning with EmberZNet 2.5.4, CCA Mode 1 is used, and a busy channel is reported if the RSSI exceeds its threshold. For software versions prior to 2.5.4, the CCA Mode was set to 0.

At RX input powers higher than –25dBm, there is some compression in the receive chain where the gain is not properly adjusted. In the worst case, this has resulted in packet loss of up to 0.1%. This packet loss can be seen in range testing measurements when nodes are closely positioned and transmitting at high power or when receiving from test equipment. There is no damage to the EM250 from this problem. This issue will rarely occur in the field as ZigBee Nodes will be spaced far enough apart. If nodes are close enough for it to occur in the field, the MAC and networking software treat the packet as not having been received and therefore the MAC level and network level retries resolve the problem without upper level application being notified.

4.2 Transmit (TX) Path

The EM250 transmitter utilizes both analog circuitry and digital logic to produce the O-QPSK modulated signal. The area-efficient TX architecture directly modulates the spread symbols prior to transmission. The differential signal paths increase noise immunity and provide a common interface for the external balun.

4.2.1 TX Baseband

The EM250 TX baseband (within the digital domain) performs the spreading of the 4-bit symbol into its IEEE 802.15.4-2003-defined 32-chip I and Q sequence. In addition, it provides the interface for software to perform the calibration of the TX module in order to reduce process, temperature, and voltage variations.

4.2.2 TX_ACTIVE Signal

Even though the EM250 provides an output power suitable for most ZigBee applications, some applications will require an external power amplifier (PA). Due to the timing requirements of IEEE 802.15.4-2003, the EM250 provides a signal, TX_ACTIVE, to be used for external PA power management and RF Switching logic. When in TX, the TX Baseband drives TX_ACTIVE high (as described in Table 6). When in RX, the TX_ACTIVE signal is low. If an external PA is not required, then the TX_ACTIVE signal should be connected to GND through a 100k Ohm resistor, as shown in the application circuit in Figure 16.

The TX_ACTIVE signal can only source 1mA of current, and it is based upon the 1.8V signal swing. If the PA Control logic requires greater current or voltage potential, then TX_ACTIVE should be buffered externally to the EM250.

4.3 Integrated MAC Module

The EM250 integrates critical portions of the IEEE 802.15.4-2003 MAC requirements in hardware. This allows the microcontroller to provide greater bandwidth to application and network operations. In addition, the hardware acts as a first-line filter for non-intended packets. The EM250 MAC utilizes a DMA interface to RAM memory to further reduce the overall microcontroller interaction when transmitting or receiving packets.

When a packet is ready for transmission, the software configures the TX MAC DMA by indicating the packet buffer RAM location. The MAC waits for the backoff period, then transitions the baseband to TX mode and performs channel assessment. When the channel is clear, the MAC reads data from the RAM buffer, calculates the CRC, and provides 4-bit symbols to the baseband. When the final byte has been read and sent to the baseband, the CRC remainder is read and transmitted.

The MAC resides in RX mode most of the time, and different format and address filters keep non-intended packets from using excessive RAM buffers, as well as preventing the CPU from being interrupted. When the reception of a packet begins, the MAC reads 4-bit symbols from the baseband and calculates the CRC. It assembles the received data for storage in a RAM buffer. A RX MAC DMA provides direct access to the RAM memory. Once the packet has been received, additional data is appended to the end of the packet in the RAM buffer space. The appended data provides statistical information on the packet for the software stack.

The primary features of the MAC are:

- CRC generation, appending, and checking
- Hardware timers and interrupts to achieve the MAC symbol timing

- Automatic preamble, and SFD pre-pended to a TX packet
- Address recognition and packet filtering on received packets
- Automatic acknowledgement transmission
- Automatic transmission of packets from memory
- Automatic transmission after backoff time if channel is clear (CCA)
- Automatic acknowledgement checking
- Time stamping of received and transmitted messages
- Attaching packet information to received packets (LQI, RSSI, gain, time stamp, and packet status)
- IEEE 802.15.4 timing and slotted/unslotted timing

4.4 Packet Trace Interface (PTI)

The EM250 integrates a true PHY-level PTI for effective network-level debugging. This two-signal interface monitors all the PHY TX and RX packets (in a non-intrusive manner) between the MAC and baseband modules. It is an asynchronous 500kbps interface and cannot be used to inject packets into the PHY/MAC interface. The two signals from the EM250 are the frame signal (PTI_EN) and the data signal (PTI_DATA). The PTI is supported by InSight Desktop.

4.5 XAP2b Microprocessor

The EM250 integrates the XAP2b microprocessor developed by Cambridge Consultants Ltd., making it a true system-on-a-chip solution. The XAP2b is a 16-bit Harvard architecture processor with separate program and data address spaces. The word width is 16 bits for both the program and data sides. Data-side addresses are always specified in bytes, though they can be accessed as either bytes or words, while program-side addresses are always specified and accessed as words. The data-side address bus is effectively 15 bits wide, allowing for an address space of 32kB; the program-side address bus is 16 bits wide, addressing 64k words.

The standard XAP2 microprocessor and accompanying software tools have been enhanced to create the XAP2b microprocessor used in the EM250. The XAP2b adds data-side byte addressing support to the XAP2 by utilizing the 15^{th} bit of the data-side address bus to indicate byte or word accesses. This allows for more productive usage of RAM, optimized code, and a more familiar architecture for Ember customers when compared to the standard XAP2.

The XAP2b clock speed is 12MHz. When used with the EmberZNet stack, code is loaded into Flash memory over the air or by a serial link using a built-in bootloader in a reserved area of the Flash. Alternatively, code may be loaded via the SIF interface with the assistance of RAM-based utility routines also loaded via SIF.

The XAP2b in the EM250 has also been enhanced to support two separate protection levels. The EmberZNet stack runs in System Mode, which allows full, unrestricted access to all areas of the chip, while application code runs in Application Mode. When running in Application Mode, writing to certain areas of memory and registers is restricted to prevent common software bugs from interfering with the operation of the EmberZNet stack. These errant writes are captured and details are reported to the developer to assist in tracking down and fixing these issues.

4.6 Embedded Memory

As shown in Figure 3, the program side of the address space contains mappings to both integrated Flash and RAM blocks.

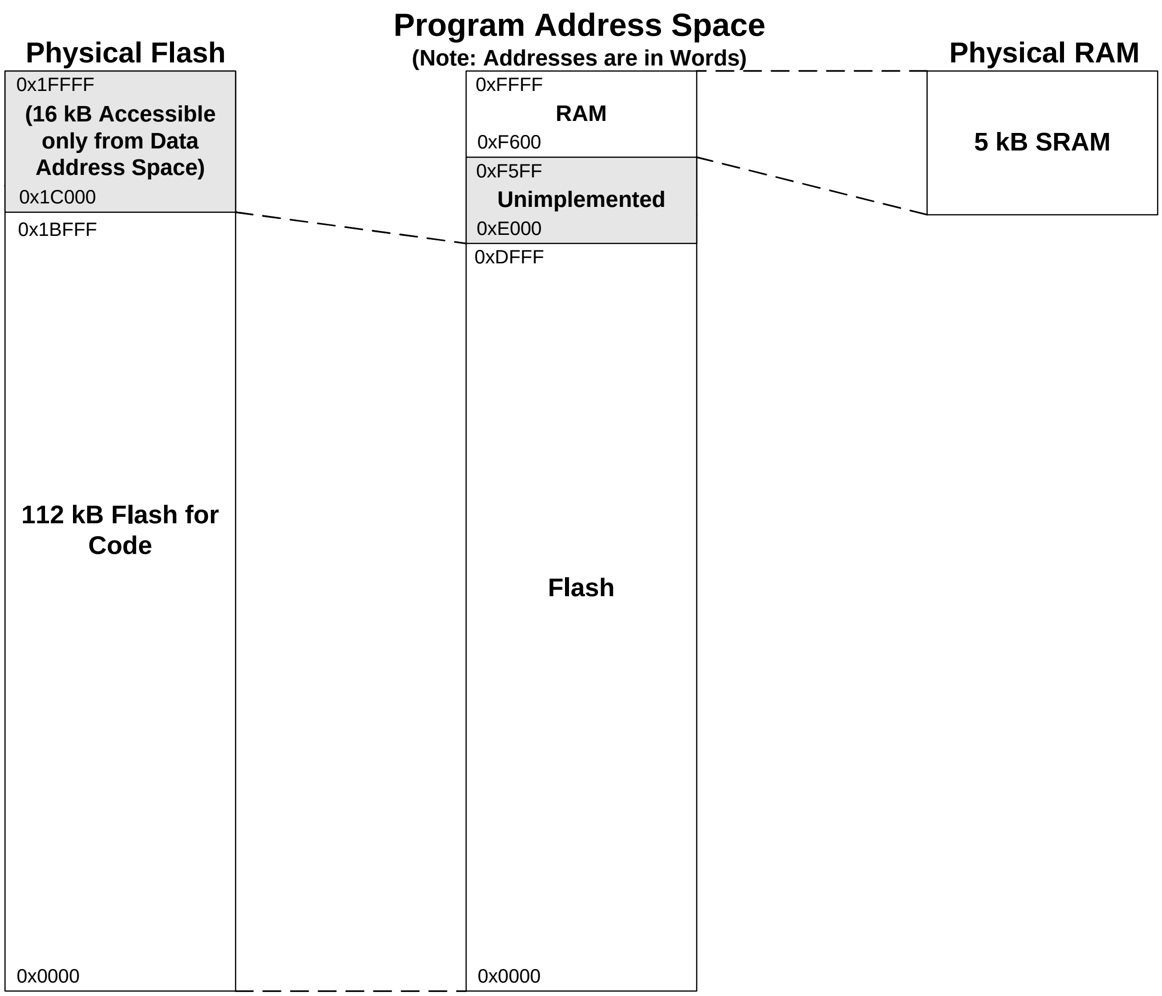


Figure 3. Program Address Space

The data side of the address space contains mappings to the same Flash and RAM blocks, as well as registers and a separate Flash information area, as shown in Figure 4.

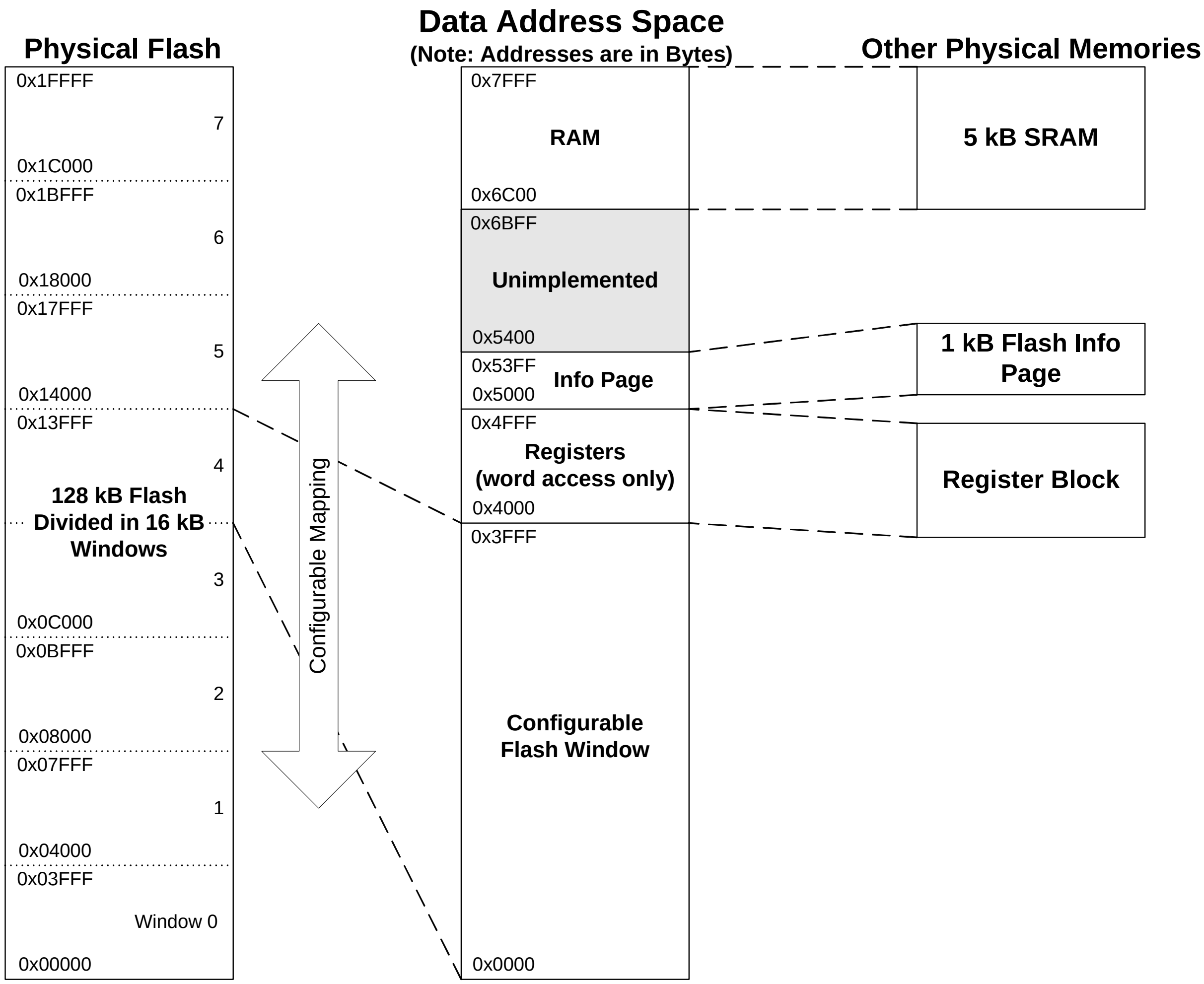


Figure 4. Data Address Space

4.6.1 Flash Memory

The EM250 integrates 128kB of Flash memory. The Flash cell has been qualified for a data retention time of >100 years at room temperature. Each Flash page size is 1024 bytes and is rated to have a guaranteed 1,000 write/erase cycles.

The Flash memory has mappings to both the program and data side address spaces. On the program side, the first 112kB of the Flash memory are mapped to the corresponding first 56k word addresses to allow for code storage, as shown in Figure 3.

On the program side, the Flash is always read as whole words. On the data side, the Flash memory is divided into eight 16kB sections, which can be separately mapped into a Flash window for the storage of constant data and the Simulated EEPROM. As shown in Figure 4, the Flash window corresponds to the first 16kB of the data-side address space. On the data side, the Flash may be read as bytes, but can only be written to one word at a time using utility routines in the EmberZNet stack and HAL.

4.6.2 Simulated EEPROM

The Ember stack reserves a section of Flash memory to provide Simulated EEPROM storage area for stack and customer tokens. Therefore, the EM250 utilizes 8kB of upper Flash storage. This section of Flash is only accessible when mapped to the Flash window in the data-side address space. Because the Flash cells are qualified for up to 1,000 write cycles, the Simulated EEPROM implements an effective wear-leveling algorithm which effectively extends the number of write cycles for individual tokens.

4.6.3 Flash Information Area (FIA)

The EM250 also includes a separate 1024-byte FIA that can be used for storage of data during manufacturing, including serial numbers and calibration values. This area is mapped to the data side of the address space, starting at address 0x5000. While this area can be read as individual bytes, it can only be written to one word at a time, and may only be erased as a whole. Programming of this special Flash page can only be enabled using the SIF interface to prevent accidental corruption or erasure. The EmberZNet stack reserves a small portion of this space for its own use, but the rest is available to the application.

4.6.4 RAM

The EM250 integrates 5kB of SRAM. Like the Flash memory, this RAM is also mapped to both the program and data-side address spaces. On the program side, the RAM is mapped to the top 2.5k words of the program address space. The program-side mapping of the RAM is used for code when writing to or erasing the Flash memory. On the data side, the RAM is also mapped to the top of the address space, occupying the last 5kB, as shown in Figure 3 and Figure 4.

Additionally, the EM250 supports a protection mechanism to prevent application code from overwriting system data stored in the RAM. To enable this, the RAM is segmented into 32-byte sections, each with a configurable bit that allows or denies write access when the EM250 is running in Application Mode. Read access is always allowed to the entire RAM, and full access is always allowed when the EM250 is running in System Mode. The EmberZNet stack intelligently manages this protection mechanism to assist in tracking down many common application errors.

4.6.5 Registers

Table 40 provides a short description of all application-accessible registers within the EM250. Complete descriptions are provided at the end of each applicable Functional Description section. The registers are mapped to the data-side address space starting at address 0x4000. These registers allow for the control and configuration of the various peripherals and modules. The registers may only be accessed as whole word quantities; attempts to access them as bytes may result in undefined behavior. There are additional registers used by the EmberZNet stack when the EM250 is running in System Mode, allowing for control of the MAC, baseband, and other internal modules. These system registers are protected from being modified when the EM250 is running in Application Mode.

4.7 Encryption Accelerator

The EM250 contains a hardware AES encryption engine that is attached to the CPU using a memory-mapped interface. NIST-based CCM, CCM*, CBC-MAC, and CTR modes are implemented in hardware. These modes are described in the IEEE 802.15.4-2003 specification, with the exception of CCM*, which is described in the ZigBee Security Services Specification 1.0. The EmberZNet stack implements a security API for applications that require security at the application level.

4.8 nRESET Signal

When the asynchronous external reset signal, nRESET (Pin 13), is driven low for a time greater than 200ns, the EM250 resets to its default state. An integrated glitch filter prevents noise from causing an inadvertent reset to occur. If the EM250 is to be placed in a noisy environment, an external LC Filter or supervisory reset circuit is recommended to guarantee the integrity of the reset signal.

4.9 Reset Detection

The EM250 contains multiple reset sources. The reset event is logged into the reset source register, which lets the CPU determine the cause of the last reset. The following reset causes are detected:

- Power-on-Reset
- Watchdog
- PC rollover
- Software reset
- Core Power Dip

4.10 Power-on-Reset (POR)

Each voltage domain (1.8V Digital Core Supply VDD_CORE and Pads Supply VDD_PADS) has a power-on-reset (POR) cell.

The VDD_PADS POR cell holds the always-powered high-voltage domain in reset until the following conditions have been met:

- The high-voltage Pads Supply VDD_PADS voltage rises above a threshold.
- The internal RC clock starts and generates three clock pulses.
- The 1.8V POR cell holds the main digital core in reset until the regulator output voltage rises above a threshold.

Additionally, the digital domain counts 1,024 clock edges on the 24MHz crystal before releasing the reset to the main digital core.

Table 11 lists the features of the EM250 POR circuitry.

Table 11. POR Specifications

Parameter	Min.	Typ.	Max.	Unit
VDD_PADS POR release	1.0	1.2	1.4	V
VDD_PADS POR assert	0.5	0.6	0.7	V
1.8V POR release	1.35	1.5	1.65	V
1.8V POR hysteresis	0.08	0.1	0.12	V

4.11 Clock Sources

The EM250 integrates three oscillators: a high-frequency 24MHz crystal oscillator, an optional low-frequency 32.768kHz crystal oscillator, and a low-frequency internal 10kHz RC oscillator.

4.11.1 High-Frequency Crystal Oscillator

The integrated high-frequency crystal oscillator requires an external 24MHz crystal with an accuracy of ±40ppm. Based upon the application Bill of Materials and current consumption requirements, the external crystal can cover a range of ESR requirements. For a lower ESR, the cost of the crystal increases but the over-all current consumption decreases. Likewise, for higher ESR, the cost decreases but the current consumption increases. Therefore, the designer can choose a crystal to fit the needs of the application.

Table 12 lists the specifications for the high-frequency crystal.

Table 12. High-Frequency Crystal Specifications

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Frequency			24		MHz
Duty cycle		40		60	%
Phase noise from 1kHz to 100kHz				- 120	dBc/Hz
Accuracy	Initial, temperature, and aging	- 40		+ 40	Ppm
Crystal ESR	Load capacitance of 10pF			100	Ω
Crystal ESR	Load capacitance of 18pF			60	Ω
Start-up time to stable clock (max. bias)				1	Ms
Start-up time to stable clock (optimum bias)				2	Ms
Current consumption	Good crystal: 20Ω ESR, 10pF load		0.2	0.3	mA
Current consumption	Worst-case crystals (60Ω, 18pF or 100Ω, 10pF)			0.5	mA
Current consumption	At maximum bias			1	mA

4.11.2 Low-Frequency Oscillator

The optional low-frequency crystal source for the EM250 is a 32.768kHz crystal. Table 13 lists the requirements for the low-frequency crystal. The low-frequency crystal may be used for applications that require greater accuracy than can be provided by the internal RC oscillator. When using the internal RC Oscillator, the pins OSC32A and OSC32B can be left open (or not connected). If the designer would like to implement the low frequency clock source with an external digital logic source, then the OSC32A pin should be connected to the clock source with OSC32B left open.

The crystal oscillator has been designed to accept any standard watch crystal with an ESR of 100 kΩ (max). In order to keep the low frequency oscillator from being overdriven by the 32.768kHz crystal, Ember recommends the PCB designer asymmetrically load the capacitor with 18pF on OSC32A and 27pf on OSC32B. For more information on this design recommendation, please review document *120-5026-000_Designing with an EM250.*

Table 13. Low-Frequency Crystal Specifications

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Frequency			32.768		kHz
Accuracy	Initial, temperature, and aging	- 100		+ 100	Ppm
Load capacitance (18pF on OSC32A and 27pF on OSC32B)			12.5		pF
Crystal ESR				100	kΩ
Start-up time				1	S
Current consumption	At 25°C, VDD_PADS=3.0V		0.6		µA

4.11.3 Internal RC Oscillator

The EM250 has a low-power, low-frequency RC oscillator that runs all the time. Its nominal frequency is 10kHz.

The RC oscillator has a coarse analog trim control, which is first adjusted to get the frequency as close to 10kHz as possible. This raw clock is used by the chip management block. It is also divided down to 1kHz using a variable divider to allow software to accurately calibrate it. This calibrated clock is available to the sleep timer.

Timekeeping accuracy depends on temperature fluctuations the chip is exposed to, power supply impedance, and the calibration interval, but in general it will be better than 150ppm (including crystal error of 40ppm). If this tolerance is accurate enough for the application, then there is no need to use an external 32.768kHz crystal oscillator. By removing the 32.768kHz oscillator, the external component count further decreases as does the Bill of Material cost. Note, if the 32.768kHz crystal is not needed, then OSC32A and OSC32B pins should be left open or not connected.

Table 14 lists the specifications of the RC oscillator.

Table 14. RC Oscillator Specifications

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
Frequency			10		kHz
Analog trim steps			1		kHz
Frequency variation with supply	For a voltage drop from 3.6V to 3.1V or 2.6V to 2.1V		0.75	1.5	%

4.12 Random Number Generator

The EM250 allows for the generation of random numbers by exposing a randomly generated bit from the RX ADC. Analog noise current is passed through the RX path, sampled by the receive ADC, and stored in a register. The value contained in this register could be used to seed a software-generated random number. The EmberZNet stack utilizes these random numbers to seed the Random MAC Backoff and Encryption Key Generators.

4.13 Watchdog Timer

The EM250 contains a watchdog timer clocked from the internal oscillator. The watchdog is disabled by default, but can be enabled or disabled by software.

If the timer reaches its time-out value of approximately 2 seconds, it will generate a reset signal to the chip.

When software is running properly, the application can periodically restart this timer to prevent the reset signal from being generated.

The watchdog will generate a low watermark interrupt in advance of actually resetting the chip. This low watermark interrupt occurs approximately 1.75 seconds after the timer has been restarted. This interrupt can be used to assist during application debug.

4.14 Sleep Timer

The 16-bit sleep timer is contained in the always-powered digital block. It has the following features:

- Two output compare registers, with interrupts
- Only Compare A Interrupt generates Wake signal

- Further clock divider of 2^N, for N = 0 to 10

The clock source for the sleep timer can be either the 32.768 kHz clock or the calibrated 1kHz clock (see Table 15). After choosing the clock source, the frequency is slowed down with a 2^N prescaler to generate the final timer clock (see Table 16). Legal values for *N* are 0 to 10. The slowest rate the sleep timer counter wraps is $2^{16} * 2^{10}$ / 1kHz ≈ 67109 sec. ≈ about 1118.48 min. ≈ 18.6 hrs.

Table 15. Sleep Timer Clock Source Selection

CLK_SEL	Clock Source
0	Calibrated 1kHz clock
1	32.768kHz clock

Table 16. Sleep Timer Clock Source Prescaling

CLK_DIV[3:0]	Clock Source Prescale Factor
N = 0..10	2^N
N = 11..15	2^{10}

The EmberZNet software allows the application to define the clock source and prescaler value. Therefore, a programmable sleep/wake duty cycle can be configured according to the application requirements.

4.15 Power Management

The EM250 supports three different power modes: processor ACTIVE, processor IDLE, and DEEP SLEEP.

The IDLE power mode stops code execution of the XAP2b until any interrupt occurs or an external SIF wakeup command is seen. All peripherals of the EM250 including the radio continue to operate normally.

The DEEP SLEEP power mode powers off most of the EM250 but leaves the critical chip functions, such as the GPIO pads and RAM powered by the High Voltage Supply (VDD_PADS). The EM250 can be woken by configuring the sleep timer to generate an interrupt after a period of time, using an external interrupt, or with the SIF interface. Activity on a serial interface may also be configured to wake the EM250, though actual reception of data is not re-enabled until the EM250 has finished waking up. Depending on the speed of the serial data, it is possible to finish waking up in the middle of a byte. Care must be taken to reset the serial interface between bytes and discard any garbage data before the rest. Another condition for wakeup is general activity on GPIO pins. The GPIO activity monitoring is described in section 5.1.

When in DEEP SLEEP, the internal regulator is disabled and VREG_OUT is turned off. All GPIO output signals are maintained in a frozen state. Additionally, the state of all registers in the powered-down low-voltage domain of the EM250 is lost. Register settings for application peripherals should be preserved by the application as desired. The operation of DEEP SLEEP is controlled by EmberZNet APIs which automatically preserve the state of necessary system peripherals. The internal XAP2b CPU registers are automatically saved and restored to RAM by hardware when entering and leaving the DEEP SLEEP mode, allowing code execution to continue from where it left off. The event that caused the wakeup and any additional events that occurred while waking up are reported to the application via the EmberZNet APIs. Upon waking from DEEP SLEEP, the internal regulator is re-enabled.