DESCRIPTION

The EM9305 is a tiny, ultra-low power and high-performance Bluetooth® LE chip.

Its highly flexible architecture allows it to act as a companion IC to any ASIC or MCU-based product, or as a complete System-on-Chip (SoC). Its flexibility also allows it to be used with other protocols. Custom applications can be executed on an efficient 32-bit ARC processor from a 512kB flash using a cache and DMA to optimize power. DSP and floating-point units can be exploited to implement advanced audio and tracking algorithms. Low-power serial interfaces (SPI, I2C, I2S, and UART) can be used to talk with external sensors, memory, display, or touch drivers. In addition, EM9305 can directly be connected to a USB port of a PC.

Included in the software is a fully featured Bluetooth® LE link layer, Host Controller Interface (HCI), stack, profiles and services, and firmware over-the-air (FOTA) updating capability. All Bluetooth® Core 5.4 features are available on the EM9305, including LE 2M and LE Coded communication, Direction Finding for localization applications. Isochronous channels are also available for LE Audio applications.

The EM9305 includes a sophisticated on-chip power management system supporting 1.5V or 3V batteries, and it can be directly powered from an attached 5V USB port. An inductor-less mode for 1.5V batteries can help further reduce the BOM. All 64kB of RAM memory can be kept in retention during sleep, or in 4kB increments to optimize leakage. A stable, low-power sleep oscillator (RC or crystal based) minimizes power consumption while in a connected state or RTC operation. Current consumption is minimized for all modes of application utilizing an efficient scheduler and memory manager.

The EM9305 features a state-of-the-art 2.4GHz transceiver: a low-power receiver with excellent sensitivity/selectivity, and a programmable transmitter up to +10dBm for optimized output power and current consumption.

The PCB footprint and cost is minimized with a very low external component count and several package options. The circuit is offered in a WLCSP23 wafer level chip-scale package, a plastic QFN-28 package, and bare die/wafer form. The device and reference design are available over the industrial temperature range.

BLUETOOTH® LE SOC AND COMPANION CHIP WITH FLASH

KEY FEATURES

I Energy efficient, industry standard, ARC EM7D, 32-bit MCU with DMA and Cache for minimum power consumption
· DSP and FPU for signal processing
· 64kB ROM
· 512kB flash memory for multi-protocol and applications
· 64kB data/instruction RAM, all retainable starting from 4kB
· 48MHz MCU and memory-read speed
· USB (only on QFN/die), UART, I2C, I2S/TDM and SPI interfaces
· ADC and on-chip temperature indicator
· 12 GPIO on QFN, 10 GPIO on WLCSP

I Qualified against Bluetooth® Core 5.4
· LE 2M and LE Coded support
· Direction Finding support
· Isochronous channels for LE Audio applications
· SPI and UART HCI Transport Layers
· Up to 4 simultaneous connections

I Security Features
· True Random Number Generator
· AES-128 Hardware Encryption/Decryption Engine
· Key Generation (ECC-P256)
· Secure Key Containers
· Secure Firmware Over-the-Air Updating (FOTA)
· Secure lifecycle management

I Sophisticated Power Management System
· Digital step up/down DCDC operation supporting 1.1-3.6V
· Inductor-less voltage multiplier mode for minimal BOM
· Direct power from 5V USB port (only on QFN/die)
· Low frequency RC or 32kHz crystal oscillator time base

I Low Current Consumption at 3V
· 3.1mA typical receiver current
· 3.4mA typical transmitter current at 0dBm
· 390nA Bluetooth® LE sleep mode with XTAL (4 kB RAM retention)
· 200nA in deep sleep mode (no RAM retention)
· 7.5nA in disable mode

I High Performance RF
· -94/-97/-103dBm RX sensitivity for 2Mbps/1Mbps/125kbps modes and 37-byte payload
· -33dBm to +10dBm transmitter output power range

I Low Component Count and Cost:
· Single ended antenna pin with matching network
· Minimal DC component network
· 48MHz XTAL, and optional 32kHz XTAL
· QFN-28 (4x4mm), WLCSP23 (1.8x1.8mm), bare-die/wafer

I Operating temperature: -40°C to +85°C

APPLICATIONS

Bluetooth® LE applications such as:
- Beacons
- Positioning
- Logistic tracking
- Wearables and Sports Equipment
- Healthcare Monitoring
- Remote Sensing
- Motion and Tracking Devices
- Home Automation
- Light Control Applications
- Smart cities
- Wireless Mice and Keyboards
- Alarms and Security System
- Toys

Enabled by the mobile phone as access point to the Internet

CUSTOMER SUPPORT

- Hardware and PCB design
- Firmware development, software development kits
- FCC/CE certification support
- Forum for hardware and software support

TYPICAL APPLICATION DIAGRAMS

The EM9305 is the ideal Bluetooth® LE companion for any MCU or ASIC application, as illustrated in Figure 0-1. It may be easily connected with:
- standard 3V MCU's within the very rich catalog of any microprocessor vendor,
- standard 1.5V MCU's such as ultra-low-power watch microprocessors from EM Microelectronic,
- any custom sensor processing ASIC for customers requiring a simple add-on function.

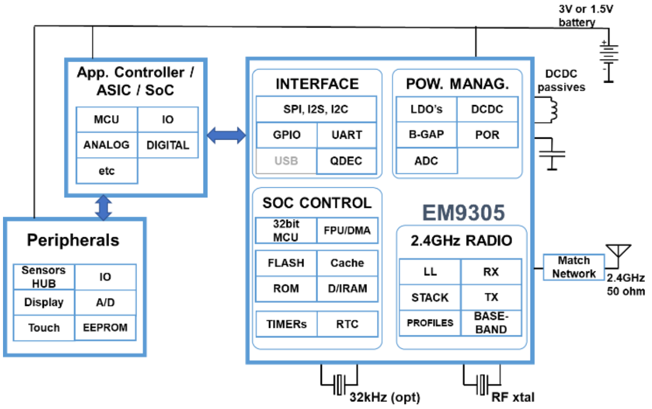


Figure 0-1: Typical application schematic – Bluetooth® LE Companion IC to any MCU or ASIC

The EM9305 is a flexible solution, which may also be used without an external MCU for simple applications such as beacons. As illustrated in Figure 0-2, it may be directly connected to external digital sensors, which may exploit the EM9305's internal 32-bit processing capability, or with sensor hubs such as from EM's SENtral platform.

Other applications using A/D converters, display and/or touch interfaces can also be implemented.

The EM9305 can be supplied from a 3V battery (e.g. Lithium coin-cell). In such case, the DCDC converter is put into step-down configuration. The EM9305 can also be supplied from a 1.5V battery (e.g. Alkaline, Silver-Oxide, or Zinc-Air single cells). In such case, the DCDC converter is put into step-up configuration.

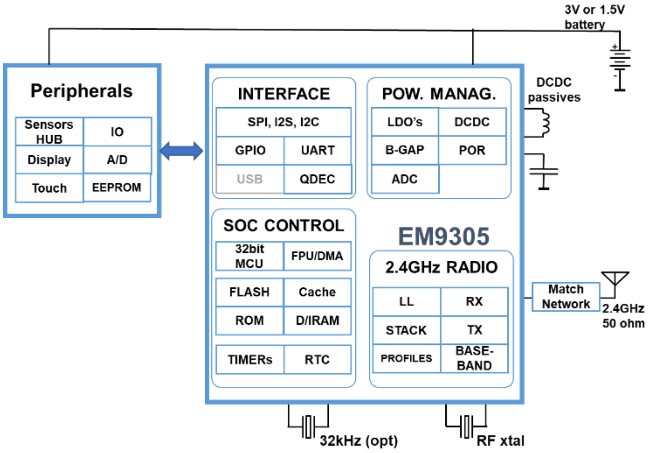


Figure 0-2: Typical application schematic – EM9305 in SOC mode connected with peripherals

Finally, configurations are also possible without the DCDC converter (no inductor required), for systems already with power management, or for systems that require the lowest bill of materials. An example of a configuration not requiring the inductor is the USB powered application shown in Figure 0-3.

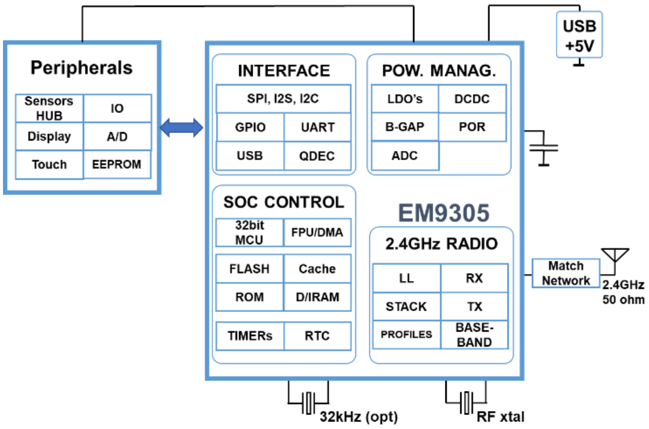


Figure 0-3: Typical application schematic – EM9305 in SOC mode powered from USB

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. PRODUCT OVERVIEW

1.1. INTRODUCTION

The EM9305 is a universal 2.4 GHz RF system on chip with the following target applications:

- **Bluetooth® LE Controller Mode**
  - Host connect to controller via Host Controller Interface (HCI)
  - HCI implemented via SPI or UART transport layer
  - Link layer implemented in flash
  - Up to 4 simultaneous connections supported (Contact EM Microelectronic for specific request)
  - Long packet lengths (payload up to 255 bytes) supported
- **Bluetooth® LE Application Mode**
  - Application hosting
    - Low energy applications such as proximity or sensor beacons using Bluetooth® LE are easily implemented
    - Connections to digital peripherals through SPI, I2C, I2S/TDM, UART, and GPIOs allow for data collection, storage, or display for example.
  - Software development platform
    - Application Program Interface (API) available for all levels of firmware
    - Full featured development tools (Metaware/MetawareLite)
    - Development and debugging using JTAG or cJTAG interface
    - Programs can be developed in flash
    - Programs stored in flash will execute from RAM or flash (through a cache system)
  - All Controller Mode and Companion Mode features supported
- **USB transport Mode**

A block diagram of the EM9305 is shown in Figure 1-1 with the power management configured in DCDC Step-Down Configuration. The main blocks are the Digital Processing Block (DPR), the RF block, the Power Management Logic (PML), the Power Management Blocks, and the General Purpose Input Output blocks (GPIO).

The chip architecture is briefly described in Section 1.2. The GPIO configuration is described in Section 1.3.

The package pins are described in Section 3.

A more detailed description of the digital processing block can be found in Section 5. A 32-bit CPU efficiently controls the movement of data between the RF modem, memory, and the digital interfaces. Digital interfaces include UART, SPI, I2C and I2S, which are mapped to GPIO as required by the application. Other peripherals include an interrupt manager, timers, and encryption engines.

An advanced power management system is described in Section 6. Power consumption and battery life are optimized in all conditions. Most common 1.5V and 3.0V primary battery cell technologies are directly supported including Lithium, Alkaline, Zinc-Air and Silver Oxide. For 1.5V batteries, the on-chip DCDC converter steps up the voltage to the required internal levels. An inductor-less voltage multiplier also allows for lower BOM. For 3.0V batteries, the on-chip DCDC converter steps the voltage down internally for efficient power consumption. Very few external components are necessary for the DCDC converter operation; however, it is also possible to operate with an internal LDO to minimize component count even further.

The RF modem is described in Section 7. A very sensitive RF front-end achieves a -97dBm typical sensitivity for 1Mbps operation with 37-byte payloads, while dissipating very low current. The efficient transmitter has programmable RF power levels from -57dBm up to +10dBm are possible. A connection to a 50-ohm antenna needs to be done with an appropriate matching circuit. Fast mode transition times and extremely low sleep current (225nA) enable very low energy application implementations.

Section 8 is describing all the peripherals and interfaces of the EM9305.

The security features, the security concept and the product life cycle are described in Section 9.

The embedded firmware is described in Section 10 and Section 11. The firmware is partitioned between flash and ROM to optimize for future proofing. A scheduler is used to manage priorities, and memory manager is used to optimize power consumption. The state can be retained in retention memory selectable from a minimum of 4kB to optimize features versus memory leakage. The flash memory is used to store trimming, unique identification numbers, protocol, stack, profiles, patches, and applications. The firmware can be updated or "patched" through SPI, UART, or over-the-air (FOTA). Patches can be loaded into and executed from flash. A jump-table system allows for managing updates.

The reference schematics are described in Section 12 with required and optional external components.

Ordering information and packaging options are described in Sections 13 and 14. Several types are offered for various application constraints. A wafer chip-scale package (WLCSP23) is offered for a minimum PCB footprint 1.8mm x 1.8mm; a QFN-28 package compatible with standard PCB technology is offered with additional GPIOs; and bare die in wafer format is offered for i.e. chip-on-board (COB) applications. Other packages are available upon request.

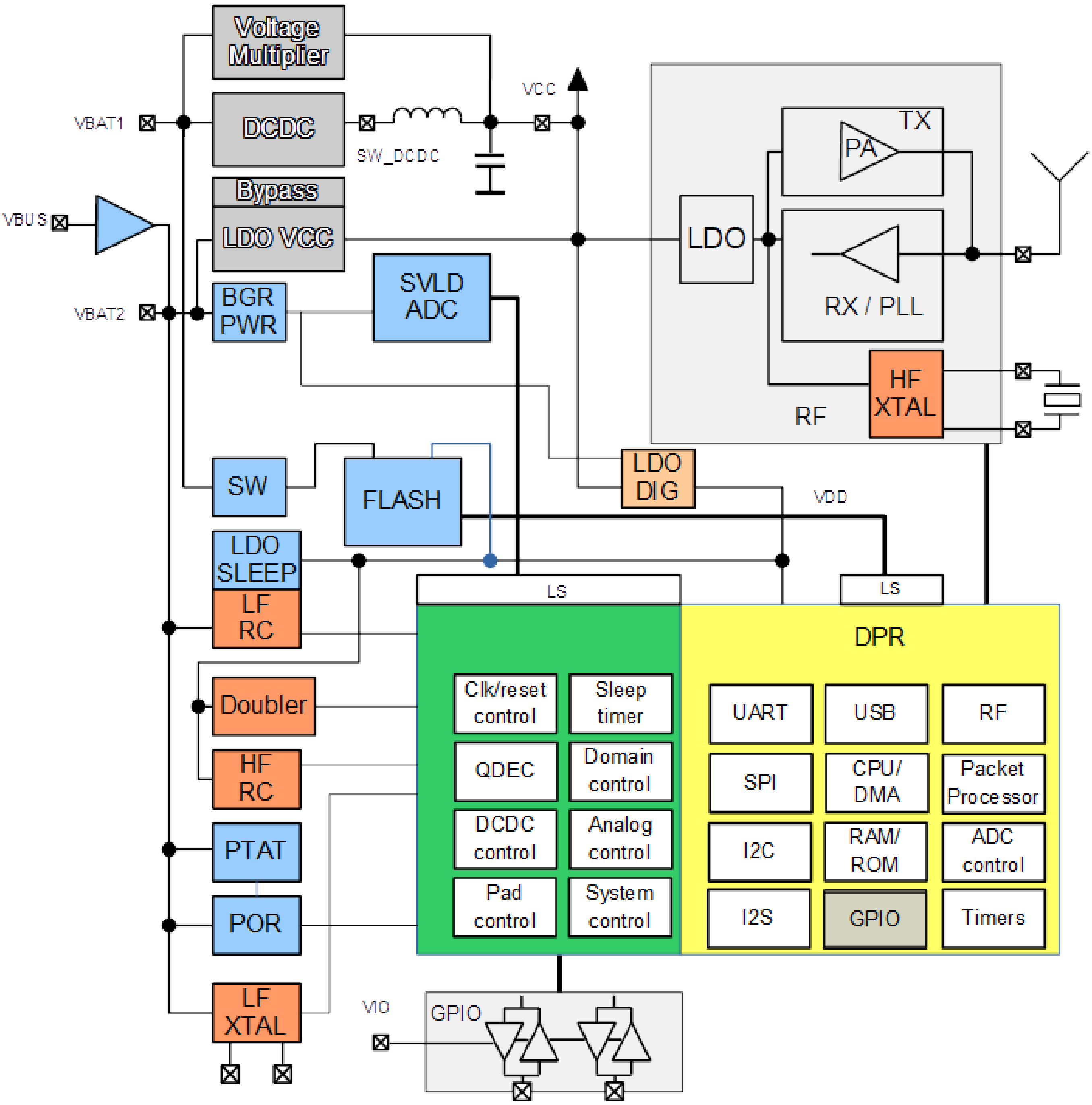


Figure 1-1: Block Diagram configured in Step-Down

1.2. CHIP ARCHITECTURE

1.2.1. MCU AND DIGITAL ARCHITECTURE

In Figure 1-2, the hardware architecture is shown. A 32-bit MCU efficiently controls the movement of data between the RF modem, memory, and the digital interfaces. The 32-bit MCU includes a digital signal-processing unit (DSP) and a floating-point unit (FPU) for efficient implementation of signal processing algorithms. A CRC coprocessor is also included for efficient verification of program memory. Memories are included for the following functions:

- ROM (64kB) – used mainly for the boot
- RAM (64kB) – used for application development and for data
  - All 64kB are retainable starting from a minimum of 4kB
  - All 64kB can be used for data (DRAM)
  - Up to 36kB can be used for instructions (IRAM) if not used for data
- Flash (512kB) – used for data, protocol, stack, profiles, and applications
  - The architecture includes also an additional 32kB as information area. This area is used for trimming as well.

- Cache (2kB) – used to optimize code access in flash

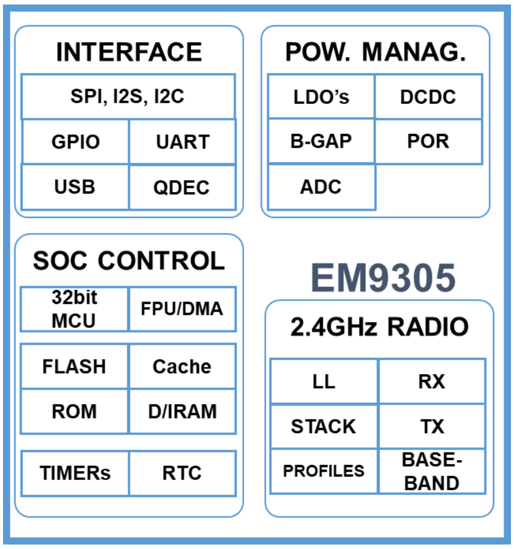


Figure 1-2: Hardware Architecture

The memory architecture is divided into several different power domains for power consumption optimizations. When a memory is not being used, it can be switched off to reduce current consumption. During Bluetooth® connected sleep mode the entire MCU subsystem can be shut off and only the power management system and required state retention memories (if any) need to be powered. The power management system will properly wake up the MCU subsystem when it is needed. The RF modem is turned on and off as needed in order to minimize energy consumption.

Digital interfaces including UART, SPI master/slave, I2S master/slave and I2C master are mapped to GPIO as required by the application. When the chip is used as a peripheral to a host application, then the SPI slave and UART can be used for communication. The standard Host-Controller Interface (HCI) is implemented for communicating with the link layer. When the chip is used as an application host, then the SPI master or I2C master can be used to talk with most standard digital peripherals. Other peripherals include an interrupt manager and two timers for low power implementations, and an AES-128 encryption engine for security implementation. A NIST compliant true random number generator is also included for key generation.

1.2.2. SOFTWARE ARCHITECTURE

The software architecture is described in detail in Section 10 and Section 11. The firmware is implemented in a power efficient manner using a basic scheduler and memory manager. The EM9305 implements a Bluetooth® Core 5.4 compliant link layer at the bottom of the stack and accessed through the standard HCI interface. Peripherals are accessed through hardware drivers. Bluetooth® HCI commands are implemented and additionally some vendor specific commands are implemented. The link layer is designed to optimize power consumption in each role. The CPU is normally halted and is only activated when a task needs to be accomplished. When sleeping, states and connection information are properly stored in the retention memory and all other memories and peripherals are turned off. A low power timer is used to properly wake up the system. The EM9305 is certified by the Bluetooth® SIG as Bluetooth® Core 5.4 Controller Subsystem.

The Bluetooth® LE stack is accessed through an Application Programming Interface (API) in application mode. The stack includes the L2CAP, Security Manager, ATT, GAP, and GATT. Standard Bluetooth® profiles such as Proximity and Find-Me are also included, as well as proprietary data-exchange and Firmware-Over-the-Air updating procedures to help manage data and program transfer.

The EM9305 can be customized using the on-chip flash memory. Specific Bluetooth® profiles can be loaded depending on the application. Simple applications can be implemented including sensor interfaces. Package configuration, production trim parameters and unique identification numbers can also be stored here.

1.2.3. RF TRANSCEIVER OVERVIEW

The RF transceiver exceeds the specifications and requirements of the Bluetooth® Core 5.4 specification.

The main features of the RF transceiver are the following:

- Ultra-low power: The peak current in receive mode is 3.1mA and in transmit mode is 3.4mA at 0dBm and 3.0V in DCDC Step-Down Configuration at room temperature.

- Excellent RF performance: including -97dBm sensitivity for 1Mbps operation with 37-byte payload and a programmable output power range from -33dBm to +10dBm
- Low voltage: Operation from 3.6V down to 1.1V
- Very high degree of integration: small footprint with few external components

The RF transceiver block diagram is shown in Figure 1-3.

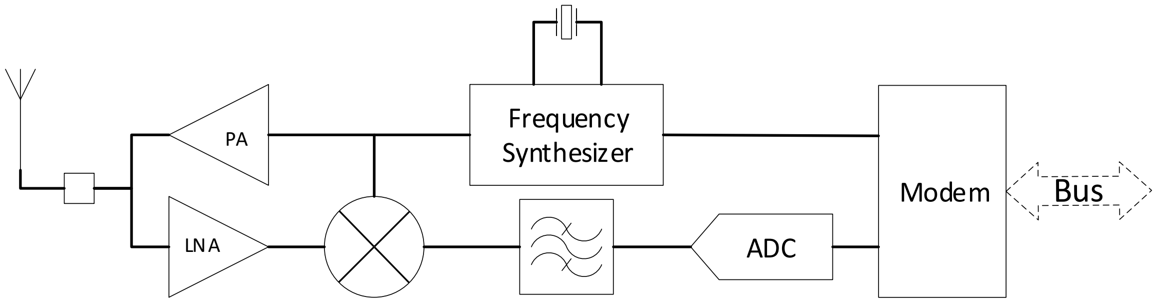


Figure 1-3: RF Transceiver Architecture (Simplified View)

The RF transceiver is divided into an analog/RF part and a digital part. The analog/RF part consists of the RX chain from LNA to ADC, the TX chain from DAC to the PA and the synthesizer subsystem that generates the LO frequency. The digital block consists of the modulator, demodulator, all calibration engines for radio optimization, timing sequencer for TX and RX and the register interface.

The Receiver chain has a low-IF architecture. The Transmitter has a dual point modulation architecture. The Synthesizer is a Frac-N PLL to generate the required LO frequency. All the blocks are optimized for lower current consumption. The RX and TX chains have multiple pre-burst calibrations that help achieve superior performance. The timing sequencer controls all pre-burst calibrations.

The RF transceiver is detailed in Section 7.

1.2.4. POWER MANAGEMENT OVERVIEW

An advanced power management system is implemented on the EM9305. Key low-power circuits include a configurable and highly efficient DCDC converter, low noise bandgap references, low dropout regulators (LDOs), a high frequency RC oscillator for efficient MCU operation, and a high accuracy, low frequency, RC oscillator for sleep mode control. A sophisticated digital control system optimizes power consumption and battery life in all conditions. Several power management configurations are possible. Four of those are described here: DCDC Step-Up, DCDC Step-Down, Voltage multiplier and USB. The configuration is automatically detected by EM9305 from the PCB.

The most common application configurations are DCDC Step-Up and DCDC Step-Down. Depending on the battery voltage, these configurations allow for higher efficiency. Note that in the WLCSP package, VBAT2 and VIO are internally shorted together.

In Figure 1-4, DCDC Step-Up Configuration, a typical arrangement is shown for connecting a 1.5V battery (1 Alkaline, Zinc Air or Silver Oxide cell, for example) to the EM9305 and an external 1.5V MCU.

The battery is applied to VCC, which is the main supply for the IC including the RF portion, and VBAT2, which powers key analog circuits in the power management. Using the SW pin, a coil and capacitor, the DCDC converter steps up the battery voltage to generate 1.7V on the VBAT1 pin to supply the flash. During sleep mode, the DCDC operation is off.

In Figure 1-5, DCDC Step-Down Configuration, a typical arrangement is shown for connecting a 3V battery (1 Lithium or 2 Alkaline cells, for example) to the EM9305 and an external 3V MCU.

The battery is applied to VBAT1, which powers the DCDC converter, and VBAT2, which powers key analog circuits in the power management. Using the SW pin, a coil and capacitor, the DCDC converter efficiently steps down the battery voltage to generate 1.15V on the VCC pin. This is the main supply voltage for the rest of the IC during normal operation. During sleep mode, the DCDC operation is off and an optional charging circuit is used to maintain VCC.

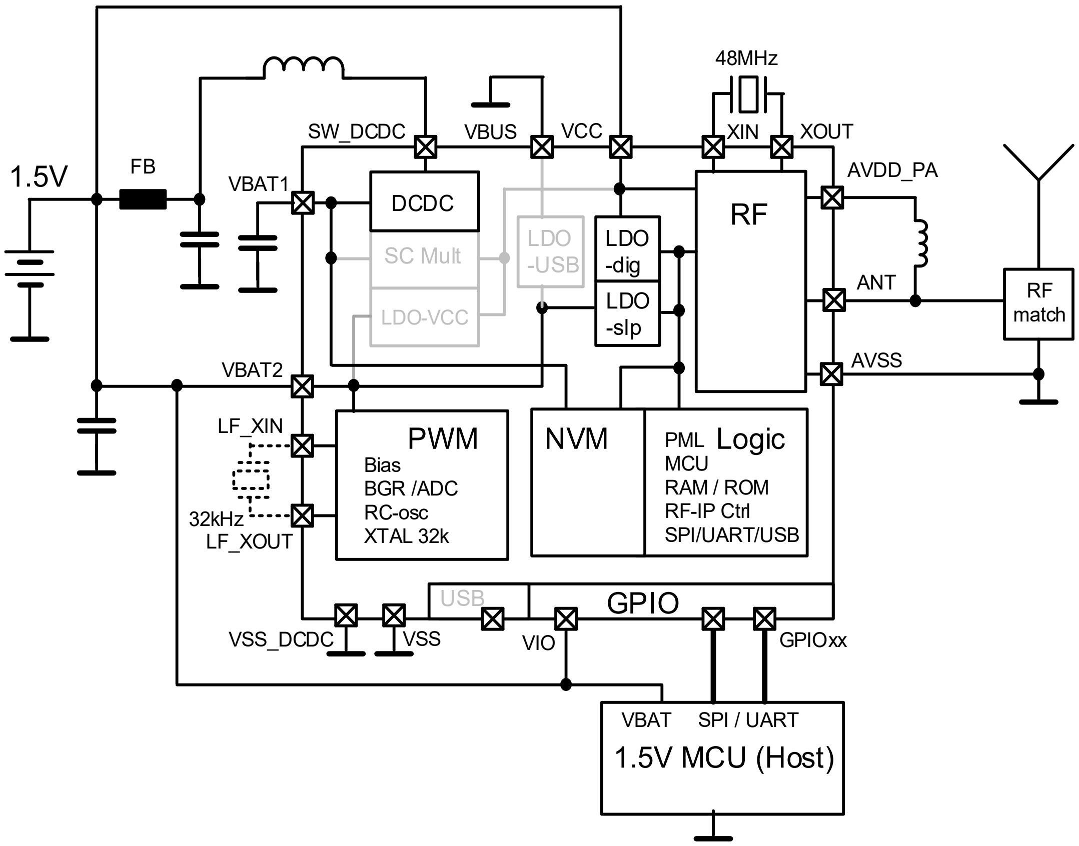


Figure 1-4: DCDC Step-Up Configuration

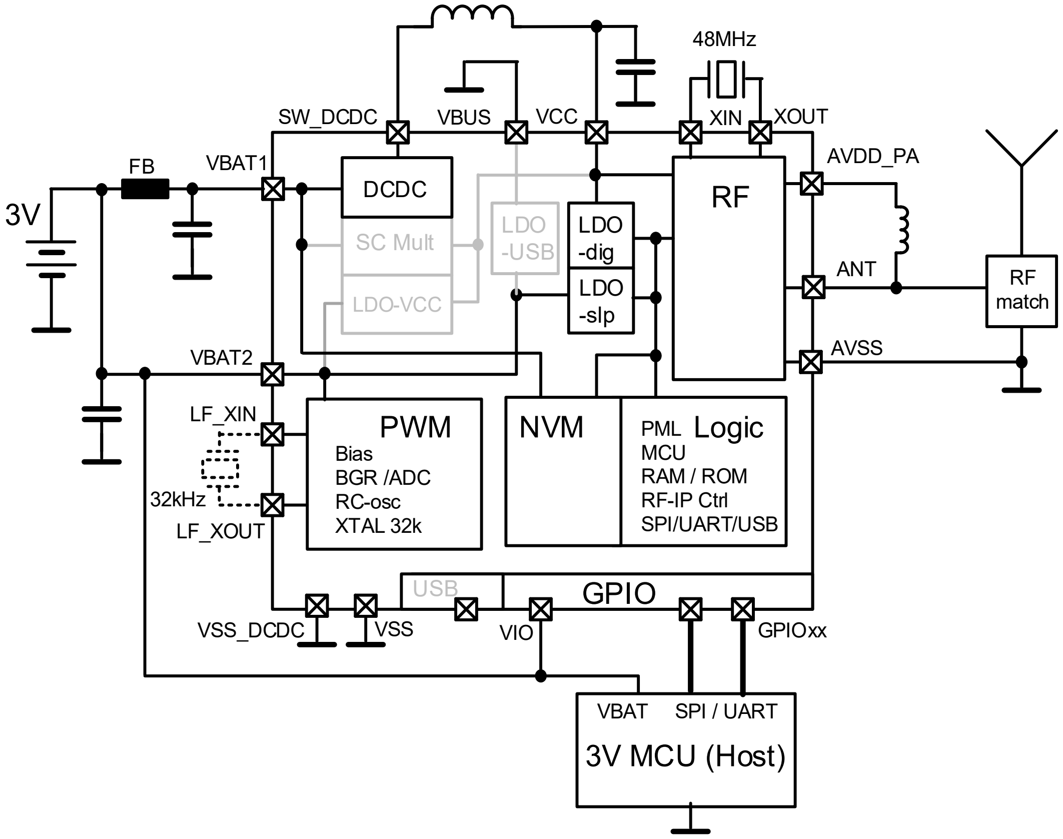


Figure 1-5: DCDC Step-Down Configuration

Other configurations are possible in order to minimize the number of external components.

For example, the Figure 1-6, Voltage Multiplier Configuration, shows a typical arrangement for connecting a 1.5V battery. VBAT1 voltage is generated by the capacitor multiplier from VBAT2. The first capacitor is internal. The multiplier is enabled/disabled by SW according to NVM usage. This configuration is slightly less efficient than the step-up configuration.

A similar configuration for higher battery voltage is possible (Direct Power configuration). Please contact EM Microelectronic for more information.

An additional configuration is shown in Figure 1-7, where the chip can directly be powered by a USB port applied on VBUS. An internal LDO is then used to generate 1.15V on the VCC pin. The DCDC converter is not used in this arrangement and is turned off. The external 1Ω resistor forms a supply voltage filter together with the capacitor. This will limit the supply slew-rate. Without it and with a fast-rising USB supply, transient overvoltage can occur inside the EM9305. In case the USB supply slew-rate is proven to be lower than 50V/µs, then the resistor can be dropped.

Other configurations could be possible, but please review with EM Microelectronic before proceeding.

A summary of some configuration options and voltage ranges are shown in Table 1-1 (including USB described below). The external supply is coming from the VBAT2 or VBUS in the case of USB mode.

Table 1-1: Power Management Configuration Options

CONFIGURATION	VBUS	VBAT1	VBAT2	VCC
DCDC Step-up	GND	1.9V from internal DCDC	1.1V – 1.9V	VBAT2
Voltage multiplier	GND	1.9V from voltage multiplier	1.1V – 1.9V	VBAT2
		>1.9V from voltage multiplier	1.9V – 2.5V	
DCDC Step-down	GND	VBAT2	1.9V – 3.6V	Internal voltage from DCDC
Direct Power (DCDC off)	GND	VBAT2	1.9V – 3.6V	Internal voltage from LDO
USB[1]	4.4V – 5.25V	VBAT2	Internal buffer	Internal voltage from LDO

If VIO is not directly connected to VBAT1/VBAT2, the condition VIO≥VBAT2 must be fulfilled.

See section 4.2 for the minimal battery supply voltage for RF operations.

The EM9305 can directly be used with the battery types listed in Table 1-2, for example. Most common 1.5V and 3.0V primary battery cell technologies are directly supported including Lithium, Alkaline, Zinc-Air and Silver Oxide. Other battery types are also supported, for example carbon-printed batteries, but additional decoupling capacitors may be necessary to supply the peak current without the battery level decreasing below the minimum voltage.

The USB supply range shall be maintained between the values indicated in Table 1-1.

Table 1-2: Typical Battery Types

TYPE	NOMINAL VOLTAGE (V)	MINIMUM VOLTAGE (V)	CAPACITY (mAh)
CR1225/CR2032 Li/MnO_2	3.0	2.0	48/225
LR44/AAA Alkaline (Zn/MnO_2)	1.5	0.9	200/1100
Zinc Air ZA675	1.4	1.1	650
Silver Oxide (Zn/Ag_2O) 357	1.55	1.2	190

[1] Only on QFN/die.

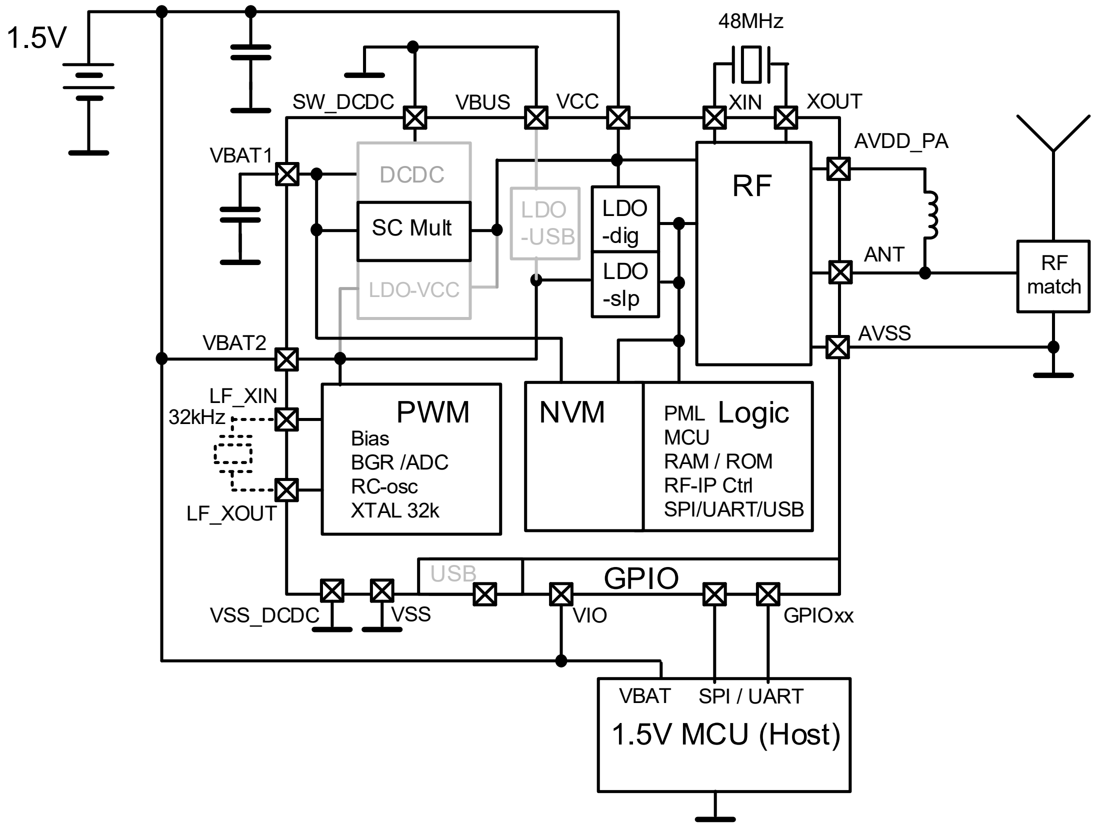


Figure 1-6: Voltage Multiplier Configuration

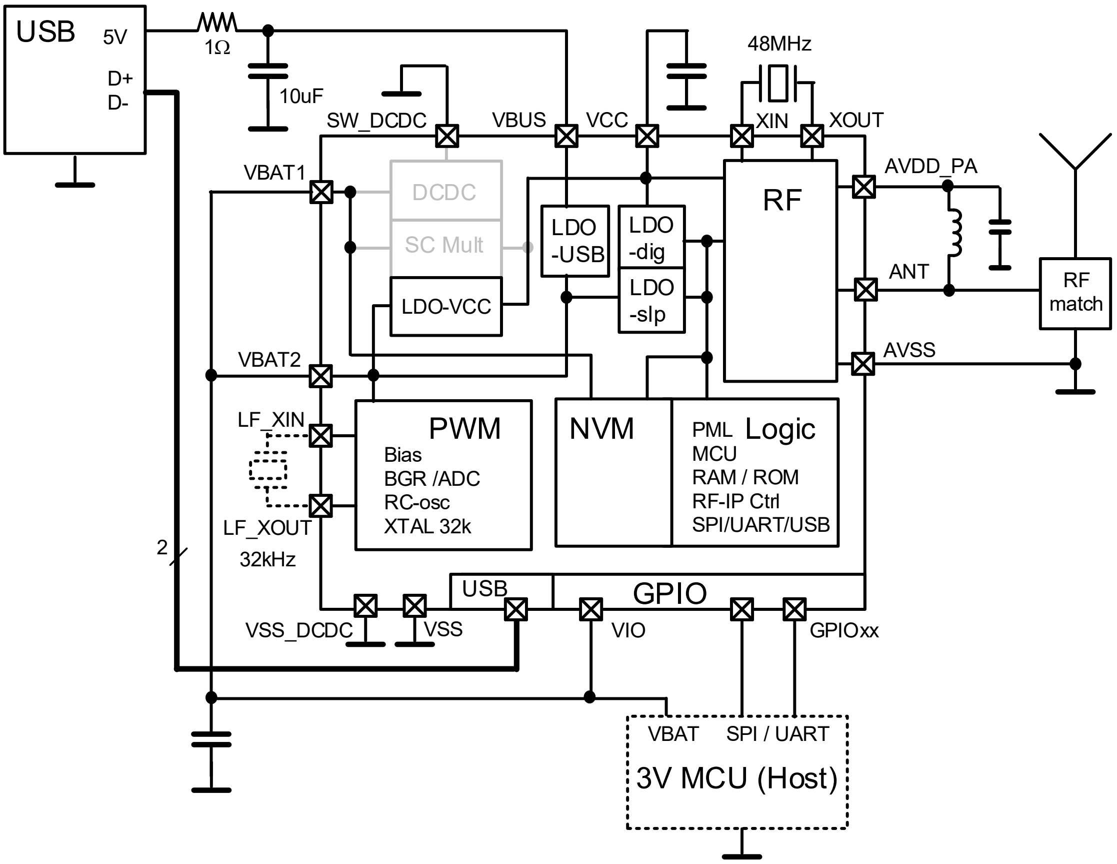


Figure 1-7: USB configuration

1.2.5. OPERATING MODES

The chip has several modes of operation including several active, standby and sleep modes. These modes are described in detail in Section 6.4. Power consumption is optimized in each of these modes. The lowest power mode while maintaining an active connection dissipates 320nA with 4kB RAM in retention state if a 32kHz crystal is used, or 585nA if the internal sleep RC oscillator is used. Additionally, a deep sleep mode is provided with typical current consumption of 200nA, and a chip disable mode is provided with typical current consumption below 10nA.

Note that special circuitry is added to keep peak currents to the battery typically less than 12.5mA when transmit power is set to 0dBm or lower.

1.3. GPIO CONFIGURATION

The EM9305 has the digital serial interfaces listed in Table 1-3, and a complete function list shown in Table 8-1. These interfaces are available through General Purpose I/O pins (GPIO). These include SPI slave and a UART for an HCI application interface, test interface and for debug. An SPI master and I2C master are provided for interfacing to external memory or digital sensors for beacon type applications, for example. A JTAG interface is provided for debug. An I2S interface is also provided for audio streaming applications.

Table 1-3: Digital Interfaces

INTERFACE	PADS	COMMENT
HCI/ SPI Slave 24 MHz @ VBAT>=1.9V 16 MHz @ VBAT < 1.9V 5 wires, 8-bit Flow control on dedicated pad	Programmable	HCI application default interface
UART, max 1.84Mbps	Programmable	HCI test interface, debug interface
SPI Master 24 MHz @ VBAT>=1.9V 16 MHz @ VBAT < 1.9V 3 or 4 wires, 8-bit	Programmable	
I2C Master Standard (up to 100kbps) Fast (up to 400kbps)	Programmable	Filters and delay lines out of pads
I2S/TDM	Programmable	
JTAG, cJTAG	GPIO8, GPIO9, GPIO10, GPIO11	Conventional 4-pin and compact 2-pin interfaces supported
USB 2.0, 12Mbps (Full Speed)	GPIO6, GPIO7	Only on QFN package or die

1.4. RELATED DOCUMENTS

The EM9305 was designed to comply with the following Bluetooth® specifications published by the *Bluetooth® Special Interest Group (SIG)* on http://www.bluetooth.org:

- *Bluetooth® Core Specification, Version 5.4, Bluetooth® SIG, 31.01.2023*
- *Bluetooth® RF PHY Test Specification, p19, Bluetooth® SIG, 07.02.2023*
- *Bluetooth® Link Layer Test Specification, p21, Bluetooth® SIG, 07.02.2023*
- *Bluetooth® Host Controller Interface (HCI), p32, Bluetooth® SIG, 07.02.2023*

Customers are, however, required to test the compliance of their final systems incorporating or embedding the EM9305 with these or other standards as they may apply and to obtain all necessary licenses and authorizations.

Other standards have also been taken into account in order to have a more generic multi-protocol support. Contact EM Microelectronic for more information.

2. HANDLING PROCEDURES

This device has built-in protection against high static voltages or electric fields; however, anti-static precautions must be taken as for any other CMOS component.

Unless otherwise specified, proper operation can only occur when all terminal voltages are kept within the voltage range. Unused inputs must always be tied to a defined logic voltage level unless otherwise specified.

3. PIN DESCRIPTION

The pins of the EM9305 are described in Table 3-1. For the QFN-28 package, the 28 pins are also shown in Figure 3-1. For the WLCSP-23, 23 pins of the 5x5 array are used, as shown in Figure 3-2. The die information can be found in Section 14.3.

Analog pins include voltage supply pins, pins for crystal oscillators, and an antenna pin for the RF. Power is supplied through VBAT1, VBAT2, VIO, and VCC depending on the power management configuration described in Section 1.2.4. The ground is connected to the various ground pins. Impedance of these connections should be minimized for low noise performance. Ideally, they should be connected directly to a ground plane on the PCB using multiple vias where possible. Two pins are provided for the 48MHz crystal (XIN, XOUT) and two are provided for the optional 32kHz crystal (LF_XIN, LF_XOUT). If there is already a 32kHz crystal in the system, the LF_XIN pin can be used to receive a reference signal. The RF antenna is connected to the ANT pin.

Digital pins include general-purpose I/O pins (GPIO) and a dedicated input pin for chip enable. There are 12 GPIO pins available on the QFN-28 package and in die form. There are 10 GPIO available on the WLCSP-23. Configuration of the GPIO is described in Section 8.2. The chip enable pin (EN) is provided to achieve the lowest possible power consumption (less than 10nA) of the device. The chip is not operational when this pin is low, and then it is initialized when this pin is raised high. VIO supplies the GPIO pins. VBUS is used only for the USB supply mode.

Table 3-1: Pin-out Description

QFN-28 Pin #	WLCSP-23 Pin #	Name	Type	Description
0		VSS_DIE		Die attach pad, connect to VSS on PCB
1	A5	ANT	RF	RF single ended antenna
2[2]	B5	AVSS	Supply	Analog ground
3	C5	XIN	XTAL	48MHz Xtal
4	C4	XOUT	XTAL	48MHz Xtal
5	C3	GPIO11	Digital	Logic input/output
6	C2	GPIO10	Digital	Logic input/output
7	D4	GPIO9	Digital	Logic input/output
8	D3	GPIO8	Digital	Logic input/output
9		GPIO7	Digital	Logic input/output – USB D+
10		GPIO6	Digital	Logic input/output – USB D-
11		VIO	Supply	GPIO voltage level
12	E3	GPIO4	Digital	Logic input/output
13	E4	GPIO1	Digital	Logic input/output
14	E2	GPIO2	Digital	Logic input/output
15	E5	GPIO3	Digital	Logic input/output
16	E1	GPIO0	Digital	Logic input/output
17	D2	GPIO5	Digital	Logic input/output – ADC – PTM
18	D1	VBAT1	Supply	Battery voltage for DCDC, configuration detection
19	C1	SW_DCDC	DCDC	Coil; configuration detection
20		VSS_DCDC	DCDC	Ground of DCDC switches
21		VBUS	Supply	USB supply
22	B1	VSS	Supply	Logic ground
23	A1	VCC	Supply	Analog supply
24	A2	VBAT2	Supply	External battery voltage
25	B2	LF_XOUT	XTAL	32kHz Xtal
26	B3	LF_XIN	XTAL	32kHz Xtal
27	B4	EN	Digital	Chip enable
28	A3	AVDD_PA	Supply	RF Supply

[2] The pin 2 is physically not connected inside the QFN package, but it shall be connected to the ground of the die attach pad on the PCB.

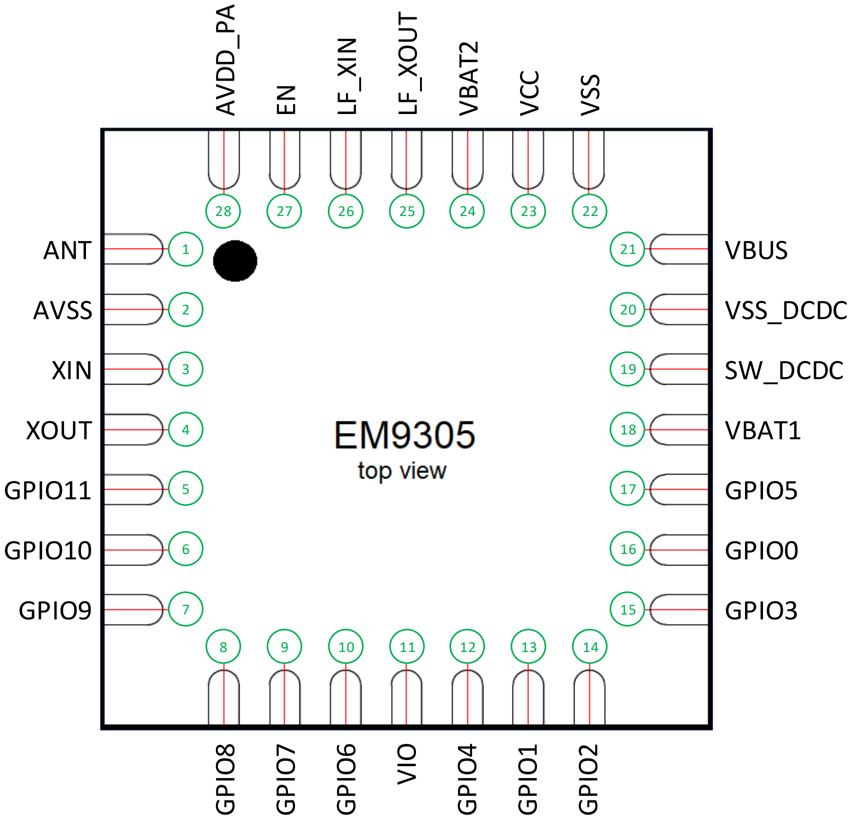


Figure 3-1: Pin location on QFN package (top/PCB view)[3]

Mechanical drawings of the QFN package are reported in Section 14.1.

[3] The pin 2 “AVSS” is physically not connected inside the package, but it shall be connected to the ground of the die attach pad on the PCB.

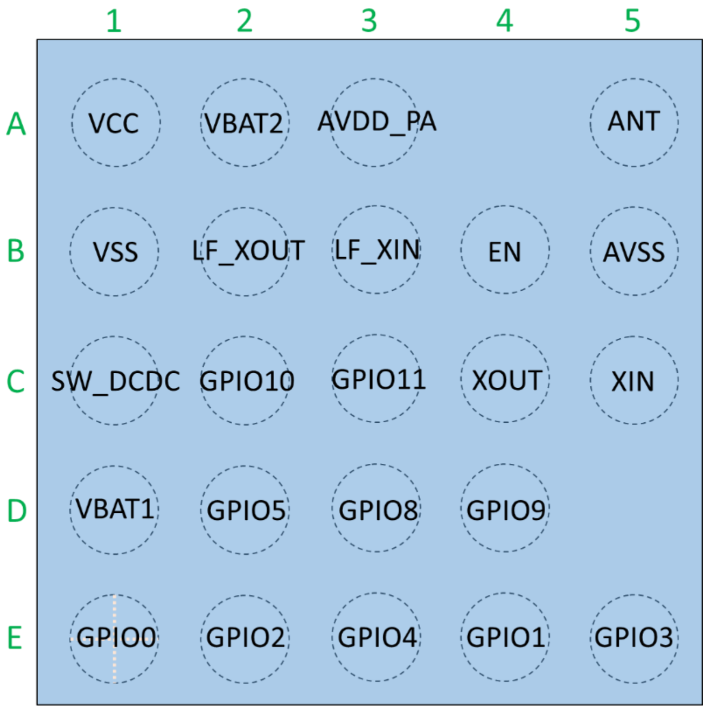


Figure 3-2: Ball location on WLCSP package (top/PCB view). Ball E1 is referenced as (0, 0) in Table 3-2

Table 3-2: Ball position (top/PCB view) with reference to Figure 3-2

Pin number	Pin name	Ball location	
		X (µm)	Y (µm)
A1	VCC	0	1400
A2	VBAT2	350	1400
A3	AVDD_PA	700	1400
A4			
A5	ANT	1400	1400
B1	VSS	0	1050
B2	LF_XOUT	350	1050
B3	LF_XIN	700	1050
B4	EN	1050	1050
B5	AVSS	1400	1050
C1	SW_DCDC	0	700
C2	GPIO10	350	700
C3	GPIO11	700	700
C4	XOUT	1050	700
C5	XIN	1400	700
D1	VBAT1	0	350
D2	GPIO5	350	350
D3	GPIO8	700	350
D4	GPIO9	1050	350
D5			
E1	GPIO0	0	0
E2	GPIO2	350	0
E3	GPIO4	700	0
E4	GPIO1	1050	0
E5	GPIO3	1400	0

Mechanical drawings of the WLCSP package are reported in Section 14.2.

4. ELECTRICAL SPECIFICATIONS

4.1. ABSOLUTE MAXIMUM RATINGS

Table 4-1 summarizes the absolute maximum ratings for the EM9305.

Table 4-1: Absolute Maximum Ratings

PARAMETER	SYMBOL	VALUE		UNIT
		MIN	MAX	
Voltage at any ground pin	V_{GND}	-0.2	0.2	V
Battery voltage 1	V_{BAT1}	-0.2	3.9	V
Battery voltage 2 (external supply)	V_{BAT2}	-0.2	3.9	V
Analog supply voltage	V_{CC}	-0.2	2.7	V
USB V_{BUS} pin voltage	V_{BUS}	-0.2	5.7	V
Voltage on USB DP/DM pin (GPIO6-7) only when in USB mode	V_{DP}/V_{DM}	-1.0	4.6	V
Voltage at any remaining pin	V_{PIN}	-0.2	3.9	V
RF input power (at the antenna with +6dBm matching network)	P_{IN}	–	17	dBm
Storage temperature	T_{ST}	-50	150	°C
HBM electrostatic discharge referred to ground – according to JEDEC JS-001	V_{HBM}	-2000	2000	V
CDM electrostatic discharge – according to JEDEC JS-002	V_{CDMQFN}	-750	750	V
Maximum soldering conditions	As per JEDEC J-STD-020 standard			

Stresses above these listed maximum ratings may cause permanent damage to the device. Exposure beyond specified operating conditions may affect device reliability or cause malfunction.

4.2. GENERAL OPERATING CONDITIONS

Table 4-2 shows the general operating conditions for the EM9305. While the chip is operational down to 1.1V, the output power levels are only guaranteed at the supply voltages specified in the table.

Table 4-2: General Operating Conditions

PARAMETER	SYMBOL	VALUE		UNIT
		MIN	MAX	
Operating temperature range	T_{OP}	-40	85	°C
Battery voltage 1 (for DCDC)	V_{BAT1}	1.7	3.6	V
Battery voltage 2 (external supply) for 0dBm TX	V_{BAT2_0}	1.1	3.6	V
Battery voltage 2 (external supply) for 3dBm TX	V_{BAT2_3}	1.25	3.6	V
Battery voltage 2 (external supply) for 6dBm TX	V_{BAT2_6}	1.73	3.6	V
Battery voltage 2 (external supply) for 8dBm TX	V_{BAT2_8}	2.55	3.6	V
Battery voltage 2 (external supply) for 10dBm TX	V_{BAT2_10}	2.55	3.6	V
Analog supply voltage	V_{CC}	1.1	1.9	V
USB supply voltage	V_{BUS}	4.4	5.25	V

All DC voltages are referred to the absolute voltage at the pin VSS.

The output power will decrease automatically if supply voltage drops below the minimum value.

The battery voltage is always connected to VBAT2, and alternatively to VBAT1 and VCC; see Figure 1-4 to Figure 1-7.

4.3. ELECTRICAL CHARACTERISTICS

Unless otherwise specified:

- All DC voltages are referred to the absolute voltage at the pin VSS.
- Typical values are measured at 25°C; minimal and maximal values are measured from -40°C to +85°C.

4.3.1. DC CHARACTERISTICS

Table 4-3 shows the DC characteristics in DCDC Step-Down Configuration with 3.0V applied to VBAT2, no flash usage, and GPIO not toggling. The values are compared with the DC characteristics in Voltage Multiplier Configuration with VBAT2 = 1.5V.

Table 4-3: Typical DC current, DCDC Step-Down and Voltage Multiplier Configurations at 48MHz CPU speed

	PARAMETER	STEP DOWN V_{BAT2} = 3.0V	V MULT V_{BAT2} = 1.5V	UNIT
CPU in Halt	RX 1Mbps mode	3.1	6.7	mA
	RX 2Mbps mode	3.3	7.2	mA
	TX mode, -33 dBm	1.6	3.4	mA
	TX mode, -24 dBm	1.7	3.6	mA
	TX mode, -15 dBm	1.9	4.0	mA
	TX mode, -12 dBm	2.0	4.3	mA
	TX mode, -9 dBm	2.2	4.7	mA
	TX mode, -6 dBm	2.5	5.4	mA
	TX mode, -3 dBm	2.9	6.3	mA
	TX mode, 0 dBm	3.4	7.4	mA
	TX mode, 3 dBm	4.4	9.1	mA
	TX mode, 4 dBm	5.3	10.0	mA
	TX mode, 5 dBm	6.4	10.8	mA
	TX mode, 6 dBm	7.8	N.A.	mA
	TX mode, 8 dBm	14.5	N.A.	mA
	TX mode, 10 dBm	17.6	N.A.	mA
	Active RC	250	510	µA
	Active XTAL	420	890	µA
	Coremark test running in NVM	1.08	2.33	mA
Sleep mode currents	Sleep mode, LF RC, no retention	530		nA
	Sleep mode, LF XTAL, no retention	310		nA
	Deep Sleep mode, LF RC, no retention	200		nA
	Deep Sleep mode, LF XTAL, no retention	310		nA
	4kB RAM in retention	80		nA
	8kB RAM in retention	110		nA
	12kB RAM in retention	145		nA
	28kB RAM in retention	260		nA
	44kB RAM in retention	415		nA
	60kB RAM in retention	550		nA
	64kB RAM in retention	650		nA
	Battery in-rush current at boot	11.5		mA
	Chip disable	7.5		nA

Sleep currents are measured on VBAT2 and with the sleep timer off.

From 8dBm and above, the current consumption is in low drop out mode. For 8dBm output power, DCDC mode can still be used to reduce the current consumption.

Typical currents in Direct Power Configuration or DCDC Step-Up are similar to those in Voltage Multiplier Configuration shown in Table 4-3.

When supplied by USB with V_{BAT2} = 5V, the chip disable current is 125 µA. This background current has to be added to the values shown in Table 4-3 to get the DC characteristics for an application supplied by USB.

4.3.2. RF CHARACTERISTICS

All the RF parameters are measured using the reference design presented in Section 12.1. Parameters, measuring conditions and device configurations are compliant with the documents listed in Section 1.4.

The supported data rates are listed in Table 4-4.

Table 4-4: Supported data rates

PARAMETER	SYMBOL	COMMENTS	MIN	TYP	MAX	UNIT
Data rate	R_{BT1} R_{BT2} R_{LR1} R_{LR2}	Bluetooth® LE 1M PHY Bluetooth® LE 2M PHY Bluetooth® LE Coded at 125kb/s (S=8) Bluetooth® LE Coded at 500kb/s (S=2)		1000 2000 125 500		kbps kbps kbps kbps

The general specifications for the high frequency crystal oscillator are shown in Table 4-5.

Table 4-5: High Frequency Crystal Oscillator Specifications

PARAMETER	SYMBOL	COMMENTS	MIN	TYP	MAX	UNIT
Crystal frequency	f_{XTAL}	Fundamental		48		MHz
Crystal deviation	df_0/f_0	Including frequency tolerance, stability over temperature, aging, and total tolerances of external capacitances			±50	ppm
Supported Xtal parameters	ESR_{XTAL} CL_{XTAL}	Equivalent series resistance Differential equivalent load capacitance	 8	 10	60 11	Ω pF

The tuning capacitance range for the high frequency (HF) XTAL is from 5pF to 13.5pF. There are six trimming bits for the on-chip capacitance with 64 steps.

The general transmitter characteristics are shown in Table 4-6.

Table 4-6: General transmitter characteristics

PARAMETER	SYMBOL	COMMENTS	MIN	TYP	MAX	UNIT
Minimum output power	P_{min}			-28		dBm
Maximum output power	P_{max}	Power setting = 10		10		dBm
Output power	P_0 P_3 P_4 P_5 P_6 P_8 P_{10}	Power setting = 0, $V_{BAT2} \geq$ 1.1V Power setting = 3, $V_{BAT2} \geq$ 1.25V Power setting = 4, $V_{BAT2} \geq$ 1.35V Power setting = 5, $V_{BAT2} \geq$ 1.5V Power setting = 6, $V_{BAT2} \geq$ 1.73V Power setting = 8, $V_{BAT2} \geq$ 2.55V Power setting = 10, $V_{BAT2} \geq$ 2.55V		0 3 4 5 6 8 10		dBm dBm dBm dBm dBm dBm dBm
Out-of-band emission in 2nd harmonic	P_{TX_H2}	Power setting = 6, step-down configuration		-45		dBm
Out-of-band emission in 3rd harmonic	P_{TX_H3}	Power setting = 6, step-down configuration		-48		dBm
Out-of-band emission in 4th harmonic	P_{TX_H4}	Power setting = 6, step-down configuration		-51		dBm
In-band spurious emission	$P_{out(fc+foff)}$	1Mbps, $\|f_{off}\|$ = 2MHz 1Mbps, $\|f_{off}\|$ = 3MHz 2Mbps, $\|f_{off}\|$ = 4MHz 2Mbps, $\|f_{off}\|$ = 5MHz		-50 -60 -55 -60	-	dBm dBm dBm dBm
Spurious emissions		f=30MHz – 88MHz f=88MHz – 1GHz f=1GHz – 12.75GHz		-70 -68 -45		dBm dBm dBm
Carrier frequency	f_{RF}		2402		2480	MHz
Deviation from the channel center frequency	Δf_c		-150		150	kHz
Frequency drift for any packet length	Δf_{c_pkt}		-50		50	kHz
Drift rate	$\Delta f_c/\Delta T$				400	Hz/µs
Modulated frequency deviation	Δf_{mod}	2Mbps 1Mbps and lower		±500 ±250		kHz

Receiver characteristics are shown in Table 4-7.

Table 4-7: General receiver characteristics

PARAMETER	SYMBOL	COMMENTS	MIN	TYP	MAX	UNIT
Sensitivity		2Mbps, 37-byte payload, BER = 1^{e-3} 2Mbps, 255-byte payload, BER = 1^{e-3} 1Mbps, 37-byte payload, BER = 1^{e-3} 1Mbps, 255-byte payload, BER = 1^{e-3} 500kbps, 37-byte payload, BER = 1^{e-3} 500kbps, 255-byte payload, BER = 1^{e-3} 125kbps, 37-byte payload, BER = 1^{e-3} 125kbps, 255-byte payload, BER = 1^{e-3}		-94 -93 -97 -96 -100 -99 -103 -102		dBm dBm dBm dBm dBm dBm dBm dBm
Maximum input power					6.5	dBm
In-band blocking (-67dBm desired signal)	C/I_0 $C/I_{\pm1}$ C/I_{+2} C/I_{-2} C/I_{+3} C/I_{-3} $C/I_{4\text{-}10}$ $C/I_{\geq10}$	Co-channel interference (i.e.0MHz) Adjacent ±1MHz interference Adjacent +2MHz interference Adjacent -2MHz interference (image) Adjacent +3MHz interference Adjacent -3MHz interference (next to image) Adjacent ≥±4MHz and ≤±10MHz interference Adjacent ≥±10MHz interference		9 -2 -45 -25 -48 -40 -45 -50		dB dB dB dB dB dB dB dB
Out-of-band blocking (-67dBm desired signal)		30MHz – 2000MHz 2003MHz – 2399MHz (excepted for f_{RF}-96MHz) 2484MHz – 2997MHz (excepted for f_{RF}+96MHz) 3000MHz – 12.75GHz	-24 -29 -29 -24	-3 -15 -10 --3		dBm dBm dBm dBm
Intermodulation @1Mbps		$f_{RX}=2*f_1-f_2$ and f_2-f_1=±3MHz / ±4MHz /±5MHz		-27		dBm
Intermodulation @2Mbps		$f_{RX}=2*f_1-f_2$ and f_2-f_1=±6MHz / ±8MHz /±10MHz		-28		dBm

An RSSI circuit with 96 dB nominal range is available with 1.5dB of accuracy after calibration and suitable for use in some applications. However, the accuracy should be thoroughly evaluated in the final application to ensure it meets the desired needs. Typically, the RSSI value should be ignored in most applications.

4.3.3. DIGITAL INTERFACES CHARACTERISTICS

The GPIO pin characteristics are described in Table 4-8. Functions implemented are digital input, push-pull output, selectable pull-down/pull-up resistors, 3.3V standard protections against VSS and VIO. More details on the functions are described in Section 8.

Table 4-8: GPIO pin characteristics

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Schmidt trigger hysteresis	V_{hys}	GPIO 0-5, 8-11 GPIO 6, 7 (USB)	0.03*VIO 0.01*VIO			V V
Input low level	V_{il}	VIO≥1.9V VIO<1.9V			0.3*VIO 0.15*VIO	V V
Input high level	V_{ih}	VIO≥1.9V VIO<1.9V	0.7*VIO 0.85*VIO			V V
Output voltage low Low drive	$V_{ol\_low\_ld}$	IOUT=3mA, VIO=1.9V, for GPIO 6, 7 (USB) IOUT=3mA, VIO=1.9V, for GPIO 0-5, 8-11 IOUT=0.8mA, VIO=1.1V, for GPIO 6, 7 (USB) IOUT=0.8mA, VIO=1.1V, for GPIO 0-5, 8-11			0.5 0.4 0.25 0.21	V V V
Output voltage high Low drive	$V_{oh\_high\_ld}$	IOUT=3mA, VIO=1.9V, for GPIO 0-5, 8-11 IOUT=3mA, VIO=1.9V, for GPIO 6, 7 (USB) IOUT=0.8mA, VIO=1.1V	1.5 1.3 0.84			V V
Output voltage low High drive	$V_{ol\_low\_hd}$	IOUT=5mA, VIO=1.9V IOUT=2mA, VIO=1.1V			0.4 0.21	V V
Output voltage high High drive	$V_{oh\_high\_hd}$	IOUT=5mA, VIO=1.9V IOUT=2mA, VIO=1.1V	1.5 0.84			V V
Pull resistor, except GPIO5[4]	R_{pull}		70	100	130	kΩ
Leakage at High-Z	I_{leak}	VIO = 1.5V VIO = 3.0V		74 190		pA pA

Output parameters for I2C mode are shown in Table 4-9. For parameters not shown here, those on Table 4-8 apply.

Table 4-9: Output parameters for I2C mode

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Output voltage low	V_{ol_low}	IOUT=3mA, VIO=1.9V IOUT=2mA, VIO=1.1V			0.4 0.21	V V
Pull resistor on GPIO5	R_{pull}		8	12	18	kΩ

4.4. TIMING CHARACTERISTICS

The general specifications for the low frequency crystal oscillator are shown in Table 4-10.

Table 4-10: Low Frequency Crystal Oscillator Specifications

PARAMETER	SYMBOL	COMMENTS	MIN	TYP	MAX	UNIT
Crystal frequency	f_{LFXTAL}	Fundamental		32768		kHz
Crystal deviation	$df_0/f_{0LFXTAL}$	Including frequency tolerance, stability over temperature, aging, and total tolerances of external capacitances	-300[5]	-20		ppm
Typical supported Xtal parameters	ESR_{LFXTAL} CL_{LFXTAL}	Equivalent series resistance Differential equivalent load capacitance		55 6	100	kΩ pF

The typical start-up time for each power configuration is listed in Table 4-11. Start-up time is defined as ENABLE pin high until RDY SPI pin high, when the device is ready to receive an HCI command. The times listed assume a SPI transport layer is used. The times assume the components listed in Table 12-1 are used.

[4] Pull-up resistor on GPIO5 is 12kΩ. It is permanent and not switchable by I2C selection signal.

[5] Valid for crystal specified in Table 12-1. Much better precision can be obtained using the on-chip calibration system.

When a 32kHz crystal is used, a cold start counter runs for 500ms. The sleep mode will be available after this delay in order to ensure precise timing. If an external signal is used, the cold start duration is 1ms for a sine wave and it is skipped for a square wave.

Table 4-11: Start-up characteristics

CONFIGURATION	COMMENTS	MIN	TYP	MAX	UNIT
Step-Down, Direct Power	V_{BAT2} ≥ 3.0V, ramp=0.3V/µs, LF RC		3.0		ms
Step-Up, Voltage multiplier	V_{BAT2} ≥ 1.5V, ramp=0.3V/µs, LF RC, 1ms precharge		3.9		ms
Any	32kHz crystal or external signal used		1.2		s

Timing characteristics from sleep to active modes are listed in Table 4-12. The transition is started by a wake-up event (from any pad or from the sleep timer) and finishes when the device is ready to receive an HCI command (wake-up flags in the registers). The times listed assume only trimming information is stored in flash and no other patches are stored in flash. The times assume the components listed in Table 12-4 are used.

Table 4-12: Timing Characteristics when changing mode

FROM MODE	TO MODE	COMMENTS	MIN	TYP	MAX	UNIT
Sleep	Active RC	Step-Up, Voltage multiplier Wake-up activated by HCI command To HCI ready signal VCC charging during Sleep is supposed to be charged		1340		µs
Sleep	Active RC	Step-Down, Direct Power Wake-up activated by HCI command To HCI ready signal VCC charging during Sleep is supposed to be charged		315		
Active RC	Active XTAL	Depends on Q of HF XTAL XTAL start-up time is dominant		280		µs

The RF controller has the ability to manage some communication timing shown in Table 4-13. See Section 7.2 for more details.

Table 4-13: Timing Management of the RF controller

PARAMETER		MIN	TYP	MAX	UNIT
RX-TX switching	Range	0		1023.75	µs
	Granularity		0.25		µs
Packet processor delay after radio start	Range	0		1023.75	µs
	Granularity		0.25		µs
Timeout for RX operation	Range	0		4 194 303.75	µs
	Granularity		0.25		µs
Delay to control external PA/LNA	Range	0		15.75	µs
	Granularity		0.25		µs

4.4.1. DIGITAL INTERFACES TIMING PARAMETERS

For I2C interface, the additional timing parameters as shown in Table 4-14 apply. For other parameters, refer to the I2C bus specification Rev.6.

Table 4-14: I2C timing parameters

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Output delay, falling edge	t_{ODEL_F}	Output load 25pF Output load 150pF Output load 400pF			500 520 540	ns ns ns
Rise time of SDA and SCL signals	t_r	Standard mode 100kHz Fast mode 400kHz			1000 300	ns ns
Fall time of SDA and SCL signals	t_f	Standard mode 100kHz Fast mode 400kHz			300 300	ns ns
Capacitive load for each bus line	C_b	Standard mode 100kHz Fast mode 400kHz			400 400	pF pF

The timing specifications for the SPI are shown in Table 4-15 and the definition is shown in Figure 4-1 for Slave and Figure 4-2 for Master.

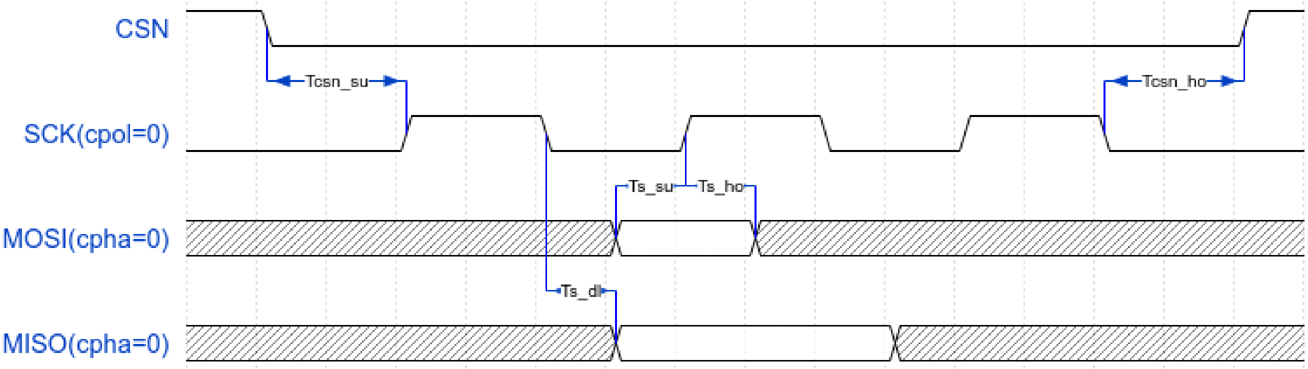


Figure 4-1: SPI Slave timing definition

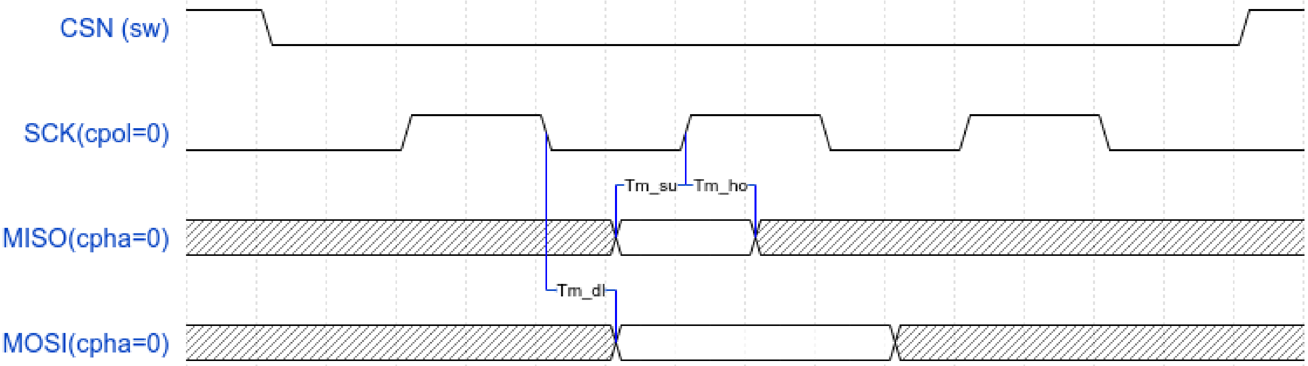


Figure 4-2: SPI Master timing definition

Table 4-15: SPI timing specifications (load conditions: 15pF)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Clock frequency Slave [6]	t_frq_s	Low drive, VIO ≥ 1.9V		16	24	MHz
		Low drive, VIO < 1.9V		8	16	MHz
		High drive, VIO ≥ 1.9V		16	24	MHz
		High drive, VIO < 1.9V		8	16	MHz
Clock frequency Master [7]	t_frq_m	Low drive, VIO ≥ 1.9V		12	24	MHz
		Low drive, VIO < 1.9V		8	16	MHz
		High drive, VIO ≥ 1.9V		12	24	MHz
		High drive, VIO < 1.9V		8	16	MHz
Slave input setup time	T_{S_SU}	Low drive, VIO ≥ 1.9V	10			ns
		Low drive, VIO < 1.9V	10			ns
		High drive, VIO ≥ 1.9V	10			ns
		High drive, VIO < 1.9V	10			ns
Slave input hold time	T_{S_HO}	Low drive, VIO ≥ 1.9V	10			ns
		Low drive, VIO < 1.9V	10			ns
		High drive, VIO ≥ 1.9V	10			ns
		High drive, VIO < 1.9V	10			ns
Slave output delay	T_{S_DL}	Low drive, VIO ≥ 1.9V	13			ns
		Low drive, VIO < 1.9V	35			ns
		High drive, VIO ≥ 1.9V	10			ns
		High drive, VIO < 1.9V	29			ns
Master input setup time	T_{M_SU}	Low drive, VIO ≥ 1.9V	31			ns
		Low drive, VIO < 1.9V	57			ns
		High drive, VIO ≥ 1.9V	30			ns
		High drive, VIO < 1.9V	52			ns
Master input hold time	T_{M_HO}	Low drive, VIO ≥ 1.9V	0			ns
		Low drive, VIO < 1.9V	0			ns
		High drive, VIO ≥ 1.9V	0			ns
		High drive, VIO < 1.9V	0			ns
Master output delay	T_{M_DL}	Low drive, VIO ≥ 1.9V	5		1	ns
		Low drive, VIO < 1.9V	5		1	ns
		High drive, VIO ≥ 1.9V	5		1	ns
		High drive, VIO < 1.9V	5		1	ns
CSN setup time	T_{CSN_SU}	Low drive, VIO ≥ 1.9V	100			ns
		Low drive, VIO < 1.9V	100			ns
		High drive, VIO ≥ 1.9V	100			ns
		High drive, VIO < 1.9V	100			ns
CSN hold time	T_{CSN_HO}	Low drive, VIO ≥ 1.9V	100			ns
		Low drive, VIO < 1.9V	100			ns
		High drive, VIO ≥ 1.9V	100			ns
		High drive, VIO < 1.9V	100			ns

[6] Typical values are obtained with the standard SPI protocol, while max values are obtained with the modified protocol (see Section 8.7.1).

[7] Typical values are obtained with the standard SPI protocol, while max values are obtained with the modified protocol (see Section 8.7.1).

4.5. NVM CHARACTERISTICS

The NVM memory is split into two areas, which are accessible using the same physical memory interface. The main area has a size of 512kB, and the info area has a size of 32kB. Both have page size of 8kB.

Table 4-16: NVM characteristics

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Supply voltage	V_{NVM}		1.65		3.6	V
Endurance			10’000			cycles
Data Retention		At 125°C	10			Years
Program Time	T_{prog}	32-bit word	8		16	µs
Page Erase Time	T_{erase}	Erase one full page	8		20	ms
Mass Erase Time	T_{erase}	Erase all the user part	8		20	ms
Program Current	I_{prog}				3	mA
Page Erase Current	I_{erase}				2	mA
Mass Erase Current	I_{erase}				2	mA

5. DIGITAL PROCESSING (DPR)

The digital processing capability is described in this section, including subsections on the CPU, memory and controllers.

The digital process block (DPR) in Figure 5-1 is designed as a CPU based system with closed coupled instruction and data memories and peripherals connected as slaves to AHB-Lite bus. All data and control transactions are done by register access, either directly or by CPU DMA control. The RF block is connected to two CPU peripherals: RF Controller and Packet Processor. The architecture is described in this Section 5 for the digital blocks around the CPU, in Section 7 for those concerning the radio, in Section 8 for the peripherals and in Section 9 for the security.

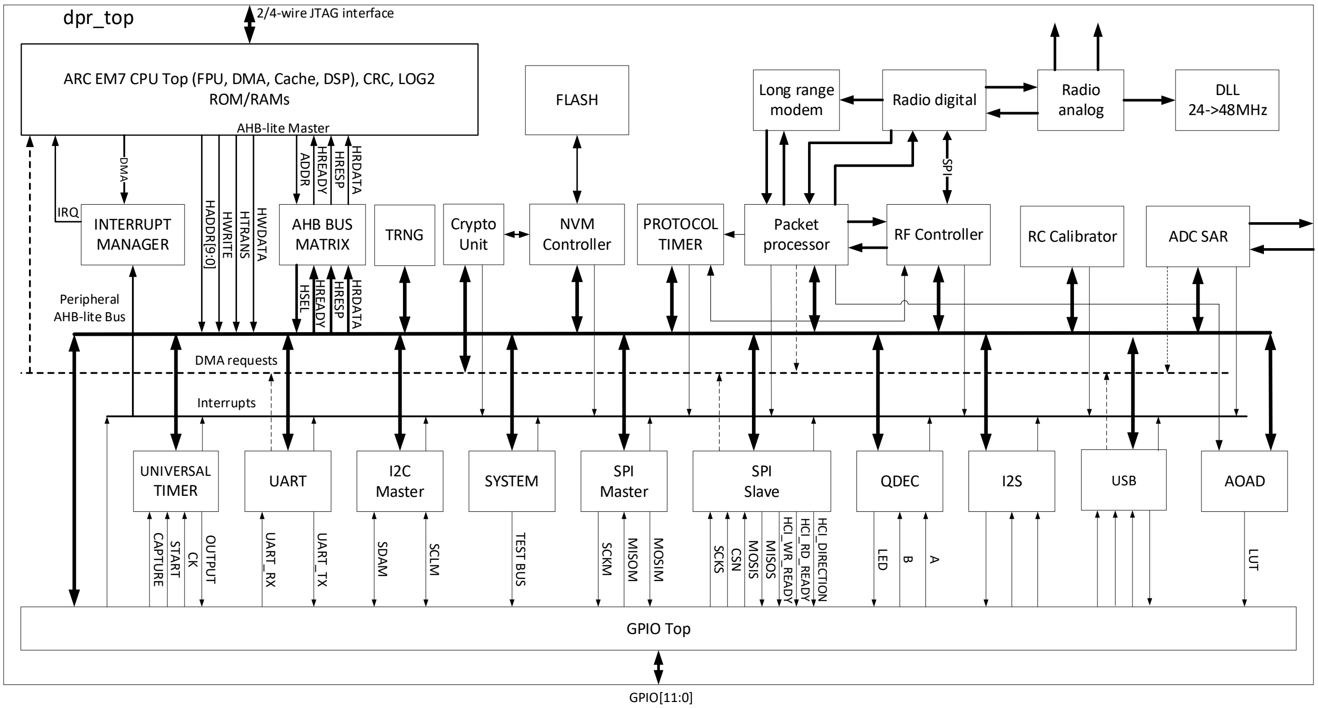


Figure 5-1: Block diagram of the Digital Processing.

5.1. CPU

The CPU is a 32-bit ARC EM7D v5.0.37 by Synopsys including a highly configurable DSP optimized for applications, which requires low-power consumption and high performance.

The integrated floating-point unit (FPU) offers hardware single-precision and double-precision math support. The micro-DMA engine allows system resources and peripherals to access the memories independent of the processor. The core has support for instruction cache.

The hardware architecture is fully pipelined to allow for single-cycle issues where possible. Power-saving features are available on all data paths and intermediate registers. The block diagram of the top-level CPU is shown in Figure 5-2.

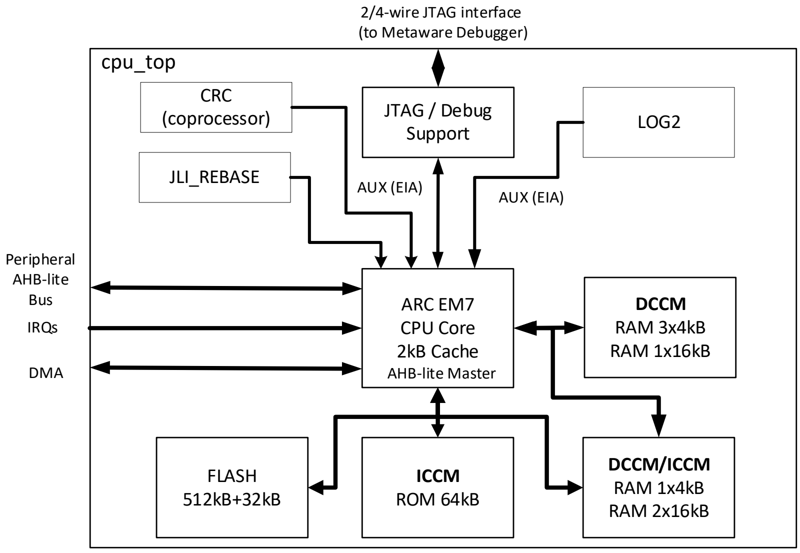


Figure 5-2: Top-level CPU block diagram

Effective use of the sleep instruction minimizes power consumption. The CPU is awakened on an interrupt, quickly executes the required functions with a 48MHz clock, and returns to sleep. A hardware interrupt handler is implemented with several interrupt levels in order to define high and low priority functions.

There are four additional co-processors extensions with dedicated CPU instructions. These extensions permit to accelerate the calculation of CRC values with bit swap. The log2() function and the JLI table rebasing are done with single cycle instruction.

The ARC processor is supported with a commercially available MetaWare toolkit (compiler, linker, debugger, etc). An integrated development environment (IDE) and software development kit (SDK) customized for the EM9305 implementation, can easily be used to link to functions embedded in ROM, implement patches, implement code in flash, and execute code from flash. A JTAG and cJTAG debugging interface are provided in the QFN package format. The WLCSP package supports only cJTAG. The mode is detected automatically.

5.1.1. DIGITAL SIGNAL PROCESSING (DSP)

The DSP can be used to minimize the CPU load when executing specific algorithms such as audio processing.

The DSP includes:

- support for fractional numbers
- high accuracy rounding of number conversion
- saturation for arithmetic operations that doesn’t allow modulo-wrapping
- accumulators
- fractional and integer vector operation on 16-bit data types
- single-cycle complex numbers multipliers
- FFT butterfly acceleration instructions

The DSP extends the baseline EM7D with a DSP pipeline that encapsulates all DSP-related instructions.

5.1.2. FLOATING POINT UNIT (FPU)

The ARC also includes a Floating-Point Unit (FPU) compliant with the IEEE 754-2008 specification:

- Full support for quiet and signaling NaN, infinity and sub-normals
- Support for all IEEE specified rounding modes
- Support for the following single-precision and double-precision operations:
  - add
  - subtract
  - multiply
- Support for the following single-precision operations:
  - floating-point format conversions
  - fast multiplication
  - square root
  - divide

All FPU operations are supported by the Metaware compiler for ARC EM7D.

5.1.3. DIRECT MEMORY ACCESS (DMA)

The micro-DMA controller provides a fast and efficient low energy way for copying large blocks of data around memory, offloading this potentially heavy-duty work from the main EM7D processor. The DMA controller is tightly coupled to the processor core interfaces to achieve low latency and reduce energy consumption during transfers.

The CPU contains a DMA controller with 8 channels, 6 descriptors in AUX, and 2 descriptors in DCCM. There is one set (*done, error*) of DMA interrupts common to all channels. The following peripherals can trigger DMA transfer: SPI Slave, UART, USB, I2S/TDM, Packet processor and ADC. All the operations are concurrent with the CPU.

5.1.4. CACHE

A 2kB instruction cache completes the architecture of the CPU.

5.2. MEMORY

Code memory is split to ROM, RAM and flash and use is optimized for power consumption. In sleep mode, ROM can be shut off without losing its contents while RAM will lose its contents if not in retention and needs to be reloaded. Therefore, all critical functions for Bluetooth® LE controller and host are implemented in the flash.

The patching system is based on using instruction index tables and dedicated CPU instructions (JLI, etc.). All functions which are supposed to be patched must use index table call. The table is stored in ROM if no patch is present. The index table is created in RAM during boot sequence if any patch is found. It is also rebuilt when waking from sleep mode if IRAM0 is not specified for retention.

Up to 64kB of RAM is provided for data. Retention memory can be set from a minimum of 4kB up to the full area of the data RAM. The retention memory is kept active during sleep mode at the expense of additional leakage current. Data in non-retention memory is lost during sleep mode.

512kB + 32kB NVM flash are available for instructions or data. Trimming, configuration data, Bluetooth® profiles and services, application and code patches can be installed into flash during manufacturing or at a later time in the field using the over-the-air firmware (FOTA) updating mechanism. Flash execution is supported by the CPU cache. The NVM contains a separate 32kB area used for storing info data, trimming data, MAC address or key containers.

The EM9305 memories are described in Table 5-1 and the architecture is shown in Figure 5-3.

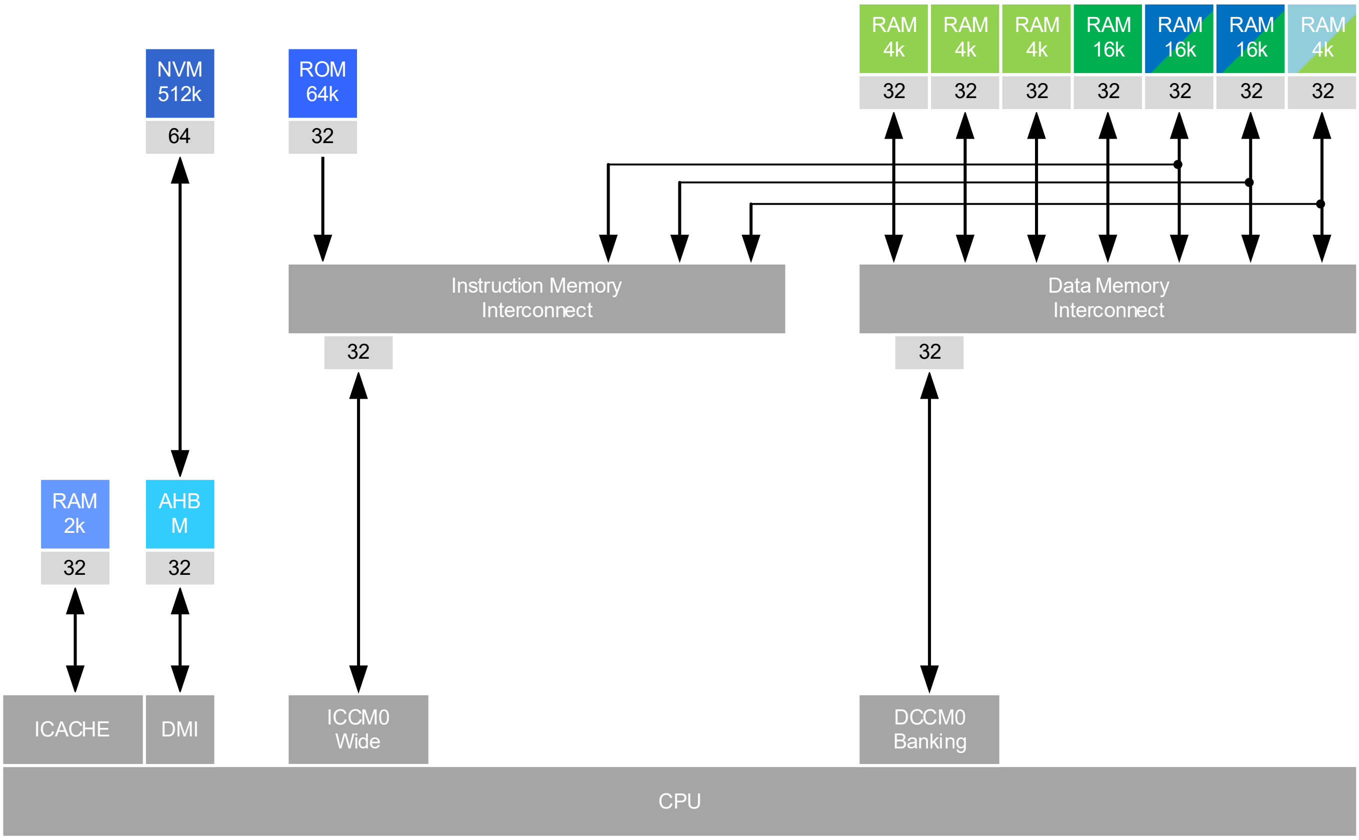


Figure 5-3: Memory architecture and interconnection

Table 5-1: Memory map

NAME	TYPE	SIZE	USAGE	ADDRESS RANGE
IROM	ROM	64kB	CPU program, boot	0x100000 0x10FFFF
IRAM1	SRAM	16kB	CPU instructions – remapped DRAM6	0x180000 0x183FFF
IRAM2	SRAM	16KB	CPU instructions – remapped DRAM5	0x184000 0x187FFF
IRAM0	SRAM	4kB	CPU instructions or JLI table – remapped DRAM0	0x1FF000 0x1FFFFF
FLASH	FLASH	512kB	CPU program, LL, Host, profiles, application, CPU data	0x300000 0x37FFFF
Flash info	FLASH	32kB	Trimming, configuration, key containers	0x400000 0x407FFF
DRAM0	SRAM	4kB	CPU data, switchable, retention storage	0x800000 0x800FFF
DRAM1	SRAM	4kB	CPU data, retention storage	0x801000 0x801FFF
DRAM2	SRAM	4kB	CPU data, retention storage	0x802000 0x802FFF
DRAM3	SRAM	4kB	CPU data, retention storage	0x803000 0x803FFF
DRAM4	SRAM	16kB	CPU data, retention storage	0x804000 0x807FFF
DRAM5	SRAM	16kB	CPU data, switchable, retention storage	0x808000 0x80BFFF
DRAM6	SRAM	16kB	CPU data, switchable, retention storage	0x80C000 0x80FFFF
PML	AHB peripheral	1kB	Power Management, sleep timers and locking bits	0xF00400 0xF007FF
IRQ MANAGER	AHB peripheral	1kB	System Interrupt Manager	0xF00800 0xF00BFF
RNG	AHB peripheral	1kB	Random Number Generator	0xF00C00 0xF00FFF
ADC	AHB peripheral	1kB	Analog-to-Digital Converter	0xF01000 0xF013FF
UNIVERSAL TIMER	AHB peripheral	1kB	Universal timer	0xF01400 0xF017FF
PROTOCOL TIMER	AHB peripheral	1kB	Timer for protocol support	0xF01800 0xF01BFF
RC CALIBRATION	AHB peripheral	1kB	LF RC oscillator	0xF01C00 0xF01FFF
UART	AHB peripheral	1kB	UART peripheral	0xF02000 0xF023FF
I2C MASTER	AHB peripheral	1kB	I2C peripheral	0xF02400 0xF027FF
NVM CONTROLLER	AHB peripheral	1kB	Flash controller	0xF02800 0xF02BFF
SPI MASTER	AHB peripheral	1kB	SPI Master	0xF02C00 0xF02FFF
SPI SLAVE	AHB peripheral	1kB	SPI Slave	0xF03000 0xF033FF
GPIO	AHB peripheral	1kB	GPIO configuration	0xF03400 0xF037FF
SYSTEM	AHB peripheral	1kB	System clock, memory configuration, DMA, USB transport	0xF03800 0xF03BFF
QDEC	AHB peripheral	1kB	QDEC configuration	0xF03C00 0xF03FFF
PACKET PROCESSOR	AHB peripheral	1kB	Packet configuration	0xF04000 0xF043FF
RF CONTROLLER	AHB peripheral	1kB	Configuration and control of the radio	0xF04400 0xF047FF
RADIO RF	AHB peripheral	1kB	RF and protocol settings of the radio	0xF04800 0xF04BFF
DIRECTION FINDING	AHB peripheral	1kB	AOA and AOD switching matrix	0xF04C00 0xF04FFF
CRYPTO UNIT	AHB peripheral	1kB	Crypto unit and key containers	0xF05000 0xF053FF
I2S	AHB peripheral	1kB	I2S	0xF05400 0xF057FF
USB	AHB peripheral	4kB	USB Device	0xF80000 0xF80FFF
USB FIFO	AHB peripheral	128kB	USB Data FIFO Direct Access	0xF81000 0xFA0FFF

5.3. NVM CONTROLLER

The 512kB flash memory has a page size of 8kB. The IO bus width is 64bits. The power management is fully under software control.

The NVM controller (or flash controller) is connected to memory AHB-Lite bus with a cache of ARC for read operation as shown in Figure 5-4. It has the following features:

- AHB error is asserted, and all control signals are set inactive if brown-out detector is asserted
- Converts 64-bit bus to 32-bit bus
- Makes flash Page Erase, Mass Erase and Full Mass Erase operation
- Makes flash Write operation with optional burst with 1 up to 8 32-bits
- Access right of erase and write, and page granularity are implemented by lock bits
- Implements register control of test signals
- Makes flash test mass read operation

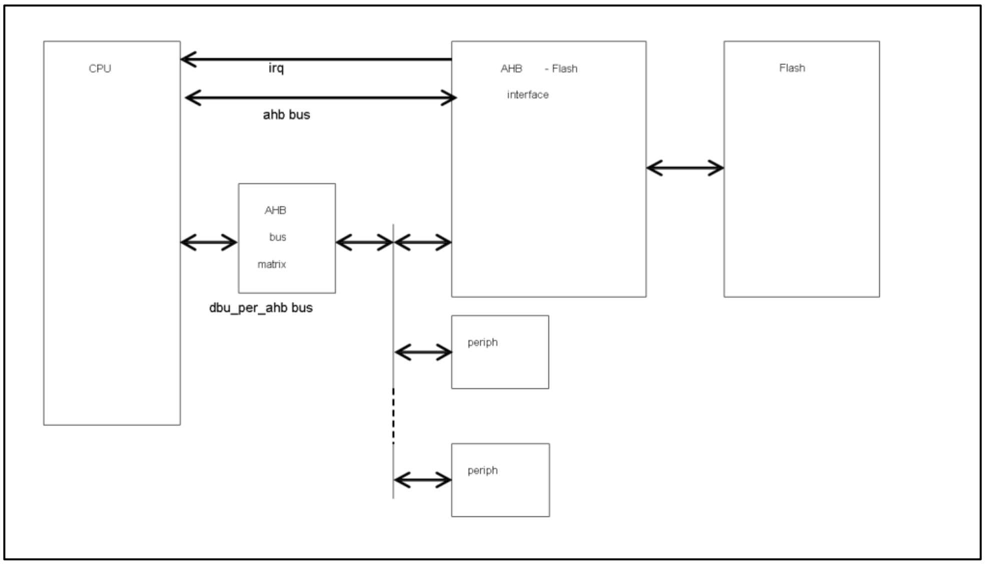


Figure 5-4: AHB-flash interface

If brownout is detected during read operation, brownout is ignored. If brownout is detected during Program/Erase operation, all control signals are set to inactive value, operation interrupted, and an error is reported. Recovery time is guaranteed by SW.

Flash read operation is performed by memory bus. All other operations are performed by SW using peripheral bus.

5.4. CLOCK CONTROLLER

The clock controller input is always 48MHz and it is responsible to generate 48MHz clocks to each block independently according to clock request from given block. It also generates 48MHz clock for some peripherals independent on the frequency mode.

The clock structure is described in Section 6.6.

5.5. APPLICATIONS

The DPR can be configured as a Bluetooth® controller or host, and software can be installed in flash.

5.5.1. BLUETOOTH® CONTROLLER

When configured as a Bluetooth® controller, the GPIO is configured as an SPI slave as the HCI transport layer by default. Alternatively, the GPIO can be configured as a UART or USB to be used for the HCI transport layer. In this configuration, the EM9305 responds to all valid HCI commands on the activated transport layer and generates the proper HCI events. Internal Bluetooth® Host stack is not used. This is the default mode.

5.5.2. SOFTWARE DEVELOPMENT

When configured for software development, software can be developed with the ARC EM7D Metaware IDE and downloaded into flash. A software development kit (SDK) can be obtained from EM Microelectronic with supporting functions, API, and examples. A hardware development kit (DVK) can be obtained from EM Microelectronic with the proper package version and a configurable PCB.

5.5.3. SOFTWARE INSTALLATION

After proper software development and verification, the object code can be loaded into flash using tools supplied by EM Microelectronic. Object code examples include link-layer or stack patches, user defined functions (vendor specific HCI commands, for example), Bluetooth® profiles, or a small Bluetooth® application such as a sensor beacon.

6. POWER MANAGEMENT AND CLOCK STRUCTURE

The power management is described in this section, including subsections on the power supply domains, the DCDC converter function, voltage multiplier, power supply monitoring, chip enable, and the reset and clock structures. The Power Management Logic (PML) manages all these blocks.

This section describes the EM9305 power management. The DCDC converter is a single-output, step-up/down converter with a simple bang-bang digital regulation running on a 48MHz clock. Alternately, for step-up operations a voltage multiplier can be used without a requirement for an external inductor. In both cases the output voltage is compared by dynamic programmable comparators, forming a supply voltage level detector (SVLD) circuit and the regulation is digitally controlled.

In sleep mode, the DCDC converter (in step-down) is off but the output capacitor on VCC can optionally be kept charged.

6.1. SUPPLY DOMAINS

The power management can have several different configurations:

- In the step-down configuration, the battery is connected to pads VBAT1 and VBAT2; the DCDC output is on the VCC pad. For TX power +8/10dBm, LDO_VCC is set to the required level.
- In the step-up configuration, the battery is connected to pads VCC and VBAT2; the DCDC output is on the VBAT1 pad.
- In the Direct Power configuration, the battery is connected to pads VBAT1 and VBAT2. The DCDC switching output is connected to ground and VCC is connected to the 2.2µF decoupling capacitor. There is a dedicated internal LDO between VBAT2 and VCC, which creates the VCC voltage level of 1.15V.
- In the Voltage Multiplier configuration, the battery is connected to pads VCC and VBAT2. The switching output is connected to ground, and a voltage is produced by an internal voltage multiplier, with a level of 1.9V output on the VBAT1 pad.

In the Direct Power configuration, the device can be supplied either to the VBAT1 and VBAT2 pins (max 3.6V) or from VBUS pin (max 5.25V, USB supply).

The supply domains in the device are described in Table 6-1.

Table 6-1: Main supply domains

SUPPLY	CONFIGURATION	RANGE [V]	DESCRIPTION
VBAT2	Step-down Direct Power (DCDC off)	1.9 – 3.6	External supply
	Step-up	1.1 – 1.9	External supply
	Voltage Multiplier	1.1 – 2.5	External supply
	USB	3.3	Output of LDO_USB
VBAT1	Step-down Direct Power (DCDC off)	1.9 – 3.6	External voltage VBAT2
	Step-up	1.9 ±30mV	DCDC output in step-up mode,
	Voltage Multiplier		Output of Voltage multiplier
	USB	3.3	External voltage VBAT2
VCC	Step-down Direct Power (DCDC off)	1.15 – 2.5	DCDC output in step-down mode * 2.5V ±25mV from output of LDO_VCC when P_8/P_{10} output power
	Step-up	1.1 – 1.9	External supply
	Voltage Multiplier	1.1 – 2.5	External supply
	USB	1.15 – 2.5	Output of LDO_VCC
VBUS	All	GND	
	USB	4.4 – 5.25	Voltage converted to 3.3V and connected to VBAT2 net

6.2. LOGIC POWER DOMAINS

The logic power domains are briefly described in Table 6-2 and shown in Figure 6-1. The logic is mainly powered by one of two internal low voltage supply, derived from VBAT2 or from VCC.

Table 6-2: Logic power domains

DOMAIN	BLOCKS	CONTROL	DESCRIPTION
VDD_P	PML	Always ON	System controller, reset controller, clock controller, pad controller, DC/DC control
VDD0	QDEC	By PML	Mouse roller controller
VDD1	DPR, IROM	By PML	The complete CPU subsystem with all peripherals
VDD2	DRAM0/IRAM0	By PML	Data/Instruction memory, retention, 4kB
VDD3	DRAM1	By PML	Data memory, retention, 4kB
VDD4	DRAM2	By PML	Data memory, retention, 4kB
VDD5	DRAM3	By PML	Data memory, retention, 4kB
VDD6	DRAM4	By PML	Data memory, retention, 16kB
VDD7	DRAM5/IRAM2	By PML	Data/Instruction memory, retention, 16kB
VDD8	DRAM6/IRAM1	By PML	Data/Instruction memory, retention, 16kB
VDD9	FLASH	By CPU	Flash VDD voltage; must be enabled when flash VIO voltage is present

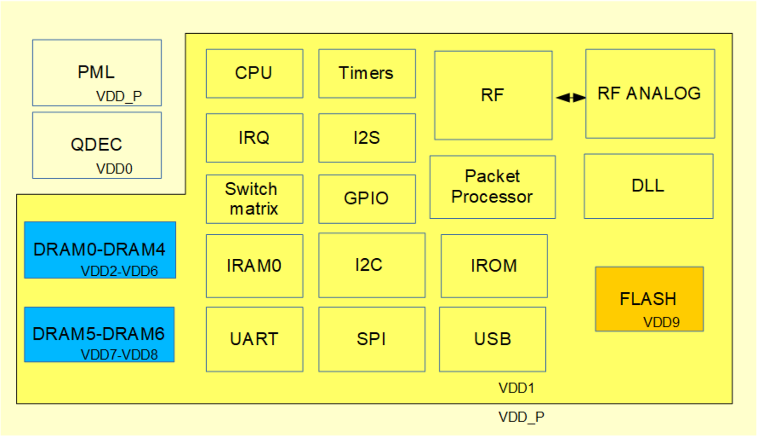


Figure 6-1: Logic Power Domains

6.3. SUPPLY MONITORING

The power management configuration is automatically detected by the power management logic and setup appropriately. This is accomplished with the use of an internal supply voltage level detector (SVLD) that can be applied to the following supply domains: VBAT1, VCC, and SW_DCDC.

A digital power meter is available for an estimation of the current consumption.

6.3.1. SUPPLY VOLTAGE LEVEL DETECTOR (SVLD)

The SVLD supply domains are described in Table 6-3.

There are 4 dynamic comparators for supply domains: VBAT1, VCC and SW_DCDC. One more is used for flash brownout detector.

Comparator thresholds are programmable for each comparator. The bandgap is used as a reference. The clock frequency is 500kHz without DCDC and programmable (default 2MHz) with DCDC.

Table 6-3: SVLD supply domains

SUPPLY	RANGE	STEP	AUTOMODE	NORMAL OPERATION
VBAT1	1.7V 1.9 – 3.4V	100mV 4 bits	Not used	Used for battery voltage monitoring Step-down/Direct Power: battery monitoring Step-up: DCDC control loop Voltage Multiplier: Switched cap output
VCC	0.95 – 1.7V	50mV 4 bits	Reference set to VBAT2-50mV	Step-up/Voltage Multiplier: battery monitoring Step-down: DCDC control loop Direct Power. LDO voltage monitor
SW_DCDC	NA	NA	Measured signal: SW_DCDC Reference: 100mV	Not used
BROWN-OUT	1.62V	NA	Not used	Flash brown-out detector Measured signal: VBAT1_NVM Reference: 1.65V

VBAT1 is used to monitor the battery voltage in DCDC Step-Down or Direct Power (DCDC off) Configurations. In DCDC Step-Up Configuration, the DCDC control loop is monitored. VBAT1 comparator has one fixed level 1.7V selected with high priority. Other programmable levels are possible between 1.9 to 3.4V.

VCC is used to monitor the battery voltage in DCDC Step-Up and Voltage Multiplier. In DCDC Step-Down Configuration, the DCDC control loop is monitored, and it should start with a pre-charged VCC capacitor. VCC level is fixed by PML to 0.95V in Direct Power (DCDC off) mode and during power-up phase in all modes. The reason is to not delay power-up for batteries close to 1.1V. The system supervisory is guaranteed by power-check.

6.3.2. DIGITAL CURRENT METER

In Step-down mode, the average battery current can be estimated from number of active DCDC cycles in a given time period. There are two dedicated 12-bit counters implemented. One is “free running” which measures the frame period (frame counter), and one counts number of active DCDC clocks inside the frame period (active counter).

The completion of a measurement is signaled by an IRQ and the counter value is then available in a register. The maximal measurement period for 4MHz DCDC operation is 1.024ms.

6.4. OPERATING MODES

The operating modes are designed to optimize the power consumption during operation. The device can be in one of the power modes shown on the Table 6-4.

When the CPU is active, two modes are possible. Active RC Mode is used with a high frequency (HF) RC oscillator for fast turn-on and turn-off performance. This can be used to service the peripherals, for example. When the RF is required, Active XTAL Mode is used with a high-accuracy crystal oscillator for channel frequency precision required by the RF standards. The crystal oscillator takes longer to turn on than the RC oscillator and more energy is consumed when in use.

In Sleep Mode and Deep Sleep Mode, the sleep timer clock (device timing reference) can be connected either to the digitally calibrated low frequency (LF) RC oscillator or to the low-frequency crystal (LF XTAL) oscillator. If the LF crystal oscillator is used, the LF RC oscillator is switched to a relaxed setting with less supply current and less accuracy. LF RC runs all the time since it is used for the power management logic. It can run either in high power mode (HP) either in low power mode (LP). The LF crystal option brings higher clock accuracy (about 10 times while periodically calibrated), which requires a shorter RF window in time for TDMA functions, hence lower average current. However, it requires an external crystal component.

Chip Disable Mode is provided as the lowest power mode possible with the battery voltage still applied to the IC, but all functions are disabled. In this mode, digital outputs are put to a Hi-Z condition and the logic states are not maintained. The firmware is also reset.

Table 6-4: Power modes

MODE	VDD OPERATION	CLOCK CPU	CLOCK PML	DESCRIPTION
Active RC	LDO and DCDC active	HF RC 48MHz	LF RC HP 500kHz	CPU enabled, logic power domains controlled by CPU
Active XTAL	LDO and DCDC active	HF XTAL 48MHz	LF RC HP 500kHz	CPU enabled, logic power domains controlled by CPU RF controlled by CPU
Sleep RC	Sleep mode of power management	None	LF RC HP 250kHz	CPU powered-down, VCC optionally charged (on by default) Selected DRAM in retention mode Optionally QDEC active
Sleep XTAL	Sleep mode of power management	None	LF RC LP 100kHz	CPU powered-down, VCC optionally charged (on by default) Selected DRAM in retention mode Ultra-low power POR and PTAT
Deep Sleep RC	Sleep mode low-power of power management	None	LF RC LP 100kHz	CPU powered-down, VCC not charged QDEC cannot be used Selected DRAM in retention mode Ultra-low power POR and PTAT
Deep Sleep XTAL	Sleep mode low-power of power management	None	LF RC LP 100kHz	CPU powered-down, VCC not charged QDEC cannot be used Selected DRAM in retention mode Ultra-low power POR and PTAT
Chip disable	None	None	None	A special mode when device is not powered, ENABLE = 0

The diagram in Figure 6-2 shows the transition between the modes.

When the chip is powered from the OFF Mode, enabled from Chip Disable Mode, or woken from Sleep Mode or Deep Sleep Mode, it enters the Active RC Mode. Active XTAL Mode is then entered if the radio is to be used to send or receive data or the HF crystal is required for other reasons. When no CPU tasks are pending, the CPU is halted. If there is enough time before the next pending CPU task Sleep Mode is entered.

Alternatively, Deep Sleep Mode can be entered by issuing the appropriate command. The chip can be woken up by the sleep timer (scheduled by the link-layer) or from pad activity such as an HCI command and enters active RC Mode as previously described.

Chip Disable Mode is entered at any time by setting the ENABLE pin to logic 0.

While using the chip in USB transport configuration, the sleep and deep sleep modes are not allowed.

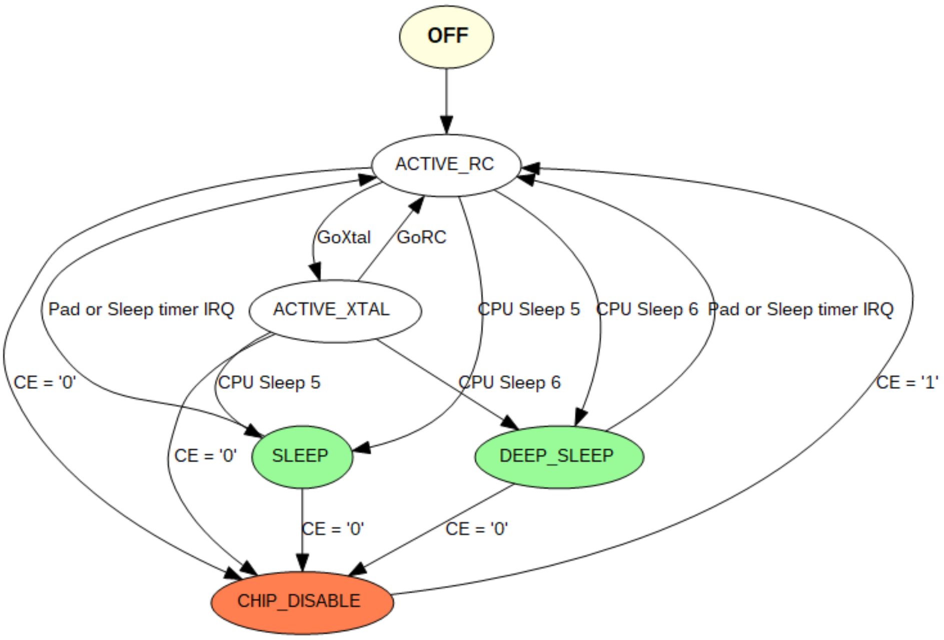


Figure 6-2: Power modes transition diagram

The transition requirements from sleep modes to active modes are shown in Table 4-12.

A transition starts with a wake-up event (QDEC, pad or Sleep timer) and finishes when the device is ready to receive HCI commands.

The device supports three HCI transport layers – UART (with flow control), SPI Slave (with flow control) and USB. Sleep modes cannot be entered if USB transport is selected as HCI transport layer. The SPI HCI interface is powered down in Sleep modes. Sleep mode in 4-wire UART (with flow control) is possible using one additional GPIO for wake-up.

6.5. RESET STRUCTURE

When the battery is inserted, a power-on-reset is generated, and the entire logic is reset. Additionally, the DPR can be reset in four different ways shown in Table 6-5.

Table 6-5: Reset sources

TYPE	CONDITION	WHAT TO RESET	DESCRIPTION
VDD power check	VDD < 0.83V	The whole logic	Static comparator
Brown-out	VBAT1 < 1.62V	DPR and RF logic	Dynamic comparator
Watchdog	CPU watchdog	DPR and RF logic	ARC request processed in PML
Software reset	**SWReset** register	DPR and RF logic	Writing a sequence 0xCAFEABAB, 0xCDCDBEEF to the register **PmlSWRst**

Every reset source has a flag register, which is set by reset event and cleared by CPU. The power domain controller must ensure proper reset of blocks after power enable.

The watchdog is done by the CPU watchdog. After watchdog event or software reset, the device stays in the current active mode (RC or XTAL).

6.6. CLOCK STRUCTURE

The device has the following main clock sources as shown in Table 6-6. By default, the EM9305 is configured to use the internal RC oscillator (lf_rc_clk) as the sleep clock source with frequency calibration enabled.

Table 6-6: Main system clock sources

NAME	OSCILLATOR, FREQUENCY	ACCURACY	DESCRIPTION
lf_rc_clk	RC, 500kHz	±10% after trimming Bluetooth® 250ppm SCA setting possible in most applications/environments. Bluetooth® 500ppm SCA setting possible in all applications/environments. Contact EM Microelectronic for details.	PML system clock when Bluetooth® sleep timer needed (on during power up)
lf_rc_lp_clk	RC, 500kHz	±30% at 25°C Drift: 2.5kHz/°C	PML system clock when Bluetooth® Sleep timer not needed or running on LF XTAL 500kHz in Active mode, it goes at 100kHz in Sleep mode
hf_rc_clk	RC, 48MHz	±2% at 25°C	Used as system clock
clk_dig_act clk_dcm_act	XTAL, 24MHz (XTAL / 2)	20ppm with XTAL from Table 12-1	Output of RF block
hf_xtal_clk	DLL, 48MHz	as XTAL	Output of DLL block multiplying XTAL clock 24MHz to 48MHz with 50:50 duty cycle
lf_xtal_clk	LF XTAL, 32.768kHz	50ppm after calibration	Used for Sleep timer
clk_spi	SPI slave clock, 16MHz	NA	SPI (HCI) slave clock

6.6.1. LF XTAL MODES

The low-frequency crystal (LF XTAL) block can operate in 3 different modes or disabled, as described in Table 6-7.

In the first mode, a crystal is connected between LF_XIN and LF_XOUT pads. In the second mode, a full swing logic signal is applied to LF_XIN and the internal amplifier is turned off. In the third mode, an analog sine wave is applied to LF_XIN and the internal low-noise amplifier is used to amplify the signal to the internal logic level. If LF_XIN or LF_XOUT are not used, they shall be left floating.

Table 6-7: LF XTAL modes

LFXtalEn	DESCRIPTION
00	LF XTAL disabled
01	LF XTAL enabled, XTAL connected to LF_XIN and LF_XOUT
10	LF XTAL disabled, 12pF load capacitor disconnected, external full swing, square wave clock signal connected to LF_XIN
11	LF XTAL amplifier enabled, 12pF load capacitor disconnected, external sine wave clock signal connected to LF_XIN

6.6.2. LF RC CLOCK CALIBRATION

Bluetooth® LE requires a minimum of +/-500ppm timing precision for a connected state. The actual communication timing precision is calculated based on the precision of both master and slave devices. Better timing precisions down to 250ppm are possible with

EM9305 and allow for longer sleep time in between communication slots, resulting in lower average power consumption, especially for longer connection intervals.

Before using the low frequency (LF) RC clock for protocol timing, it must be calibrated against the high-frequency crystal (HF XTAL). Typically, the HF XTAL has better than ±50 ppm versus offset, temperature and aging, but this should be confirmed with the specification of the actual crystal used. Frequency trimming of the HF XTAL may be required for each application since the PCB layout parasitic varies between designs. The EM9305 comes trimmed with the frequency centered based on the reference design. The accuracy achieved depends on the length of the calibration, the frequency of the calibration and the expected maximum temperature gradient of the application. The length and frequency of calibration can be adjusted in the configuration of the EM9305.

The calibration system consists of two timers. First timer running on LF clock, second timer running on HF XTAL clock. Calibration is started manually by writing to register. Once calibration is finished, IRQ is generated.

The calibration runs until the timer running on LF clock reaches predefined target value (number of LF clock periods, which are used for measurement).

The RC calibration system has the following features:

- HF XTAL clock as a reference clock (48MHz).
- Fixed calibration interval (fixed number of reference clock period).
- Variable calibration interval (fixed number of measured clock period).
- Two independent counters for LF and HF clock as shown in Table 6-8.

Table 6-8: Counters used for RC calibration

COUNTER	CLOCK SOURCE	FREQUENCY	MAXIMUM TIMING COUNT
LF – 14-bit	Sleep timer LF RC LF XTAL	32kHz – 65kHz 500kHz 32kHz	512ms – 252ms 32.78ms 512ms
HF – 24-bit	HF XTAL	48MHz	350ms

The general approach (shown in Figure 6-3) is to count the number of LF clock periods in interval timed by HF XTAL clock. Then recalculation between LF and HF clock is:

$$N_{HF} = N_{LF} \cdot \frac{f_{HF}}{f_{LF}} = N_{LF} \cdot \frac{M_{HF}}{M_{LF}}$$

M_{HF} and M_{LF} are the number of HF clock periods and LF clock periods respectively in the reference interval. Two approaches for measurement can be used. Either using constant number of LF clock periods (M_{LF}) or using constant number of HF clock periods (M_{HF}).

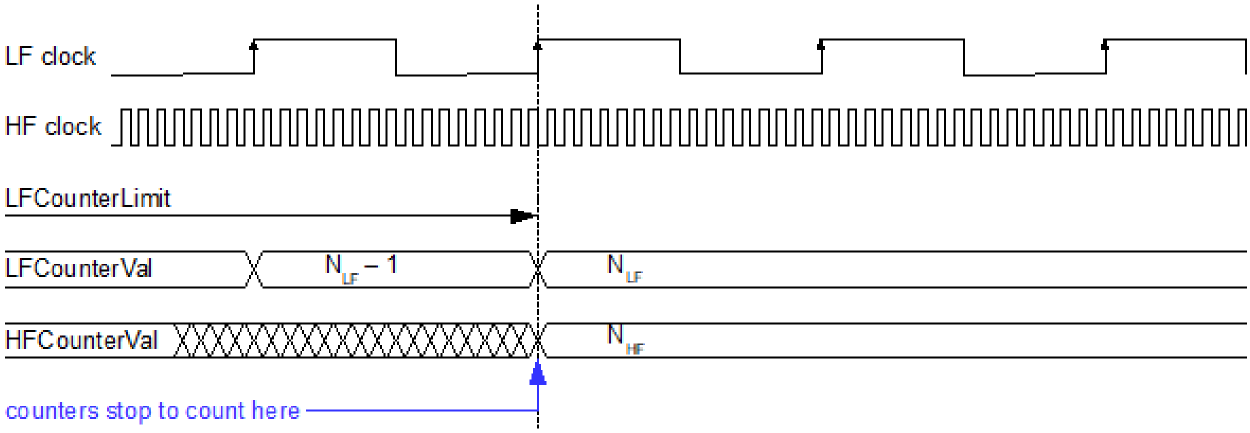


Figure 6-3: LF Clock Calibration

6.6.3. LF XTAL CLOCK CALIBRATION

If a low-frequency crystal is used, the oscillator frequency accuracy can also be improved with calibration. A typical low-cost crystal is centered at room temperature, but has a quadratic temperature dependency and it can drift to -200ppm at both -40°C and 85°C. These calibration parameters can also be adjusted in the configuration of the EM9305.

Thus, also the LF XTAL oscillator must be calibrated by the same system as the LF RC oscillator.

Contact EM Microelectronic for further guidance in setting the low-frequency clock calibration parameters for your application.

7. RADIO

The radio is made by low-IF transceiver (see Section 7.1) and has a significant portion implemented in digital including the modem, the RF controller, the packet processing, the antenna switching unit, and encryption, which is interfaced to the DPR CPU bus and also to the GPIOs. A block diagram is shown in Figure 7-1 and a description is given in the next sections.

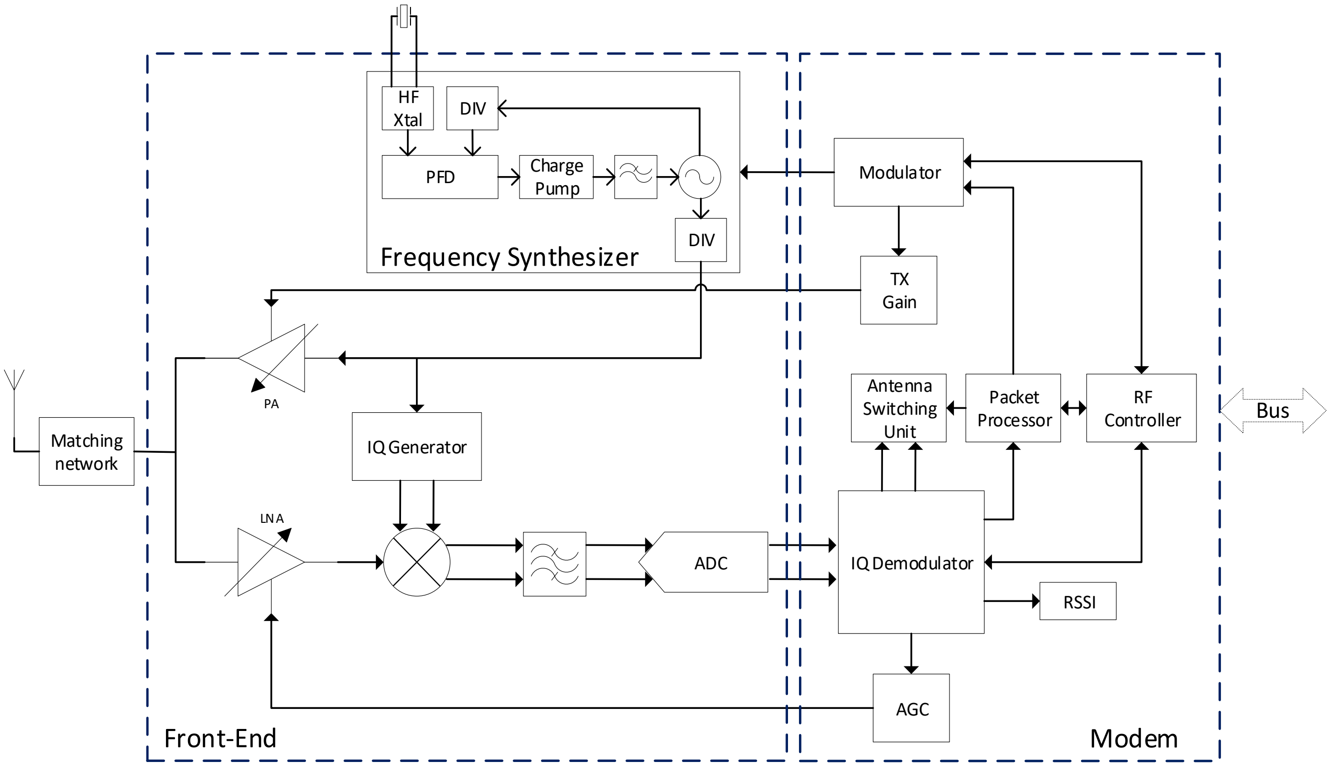


Figure 7-1: Block diagram of the transceiver, including the front-end and the modem

The RF controller (described in Section 7.2) has the task to configure and control the radio, and to interface with the packet processor to start/stop packet sending/receiving.

The data rate supported includes those for Bluetooth® LE 2M with 2Mbps, 1Mbps and support for Bluetooth® LE Coded low data rate of 500kbps and 125kbps. The packet processor (described in Section 7.3) manages all the packet formats included in the Bluetooth® Core 5.4.

The modem allows access to the IQ sampling for direction finding functionalities (see Section 7.4) with the integrated antenna-switching unit for Direction Finding.

The modem includes automatic packet handling with preamble & sync, CRC, separate 32-bytes RX FIFO and 16-bytes TX FIFO, support of long packets, early signaling of incoming packet, integrated CCM-AES encryption, and supporting multiple simultaneous connections.

7.1. 2.4GHz FRONT-END

The RF block operates at 2.4GHz and includes a transmitter (TX) with a programmable power amplifier (PA), a receiver (RX), and a phase-locked loop (PLL) frequency synthesizer. The PLL includes a high frequency crystal oscillator (HF XTAL), which requires an external 48MHz quartz crystal.

The 2.4GHz front-end is based on a low-IF architecture and comprises the following building blocks:

- Single-ended RF port with off-chip matching network and harmonic filter
- High gain, low power, LNA and mixer with gain control
- Low-IF receiver with programmable channel filter BW, and ADC
- Power Amplifier with programmable output power range
- Fully integrated frequency synthesis with fast settling time to generate the required LO frequency
- Large range RSSI of 96dB with 1.5dB step accuracy
- 48MHz XTAL reference with finely trimmable internal loading capacitor
- Fully integrated FSK-based modem, with programmable pulse shape, data rate and modulation index
- On-chip calibration for filters, TX gain, frequency
- RX/TX chains pre-burst calibrations
- Modem including baseband with link layer functionalities

7.2. RF CONTROLLER

The RF controller has 4 main interfaces which functions are described in Table 7-1 and illustrated in Figure 7-2.

Table 7-1: RF controller interfaces

INTERFACE TO	FUNCTION
Radio	Radio configuration – SPI interface to radio IP. Radio control – control radio basic operation like start TX/RX mode, stop radio, read RSSI value.
CPU subsystem	AHB Lite slave interface to the block register map. CPU interrupt signals.
Packet processor	Control packet processor to start radio data (bit-stream) encoding/decoding. Get status from packet processor when packet is sent/received or other packet related events.
Protocol timer	Protocol timer generates HW triggers to start radio operation at precise time.

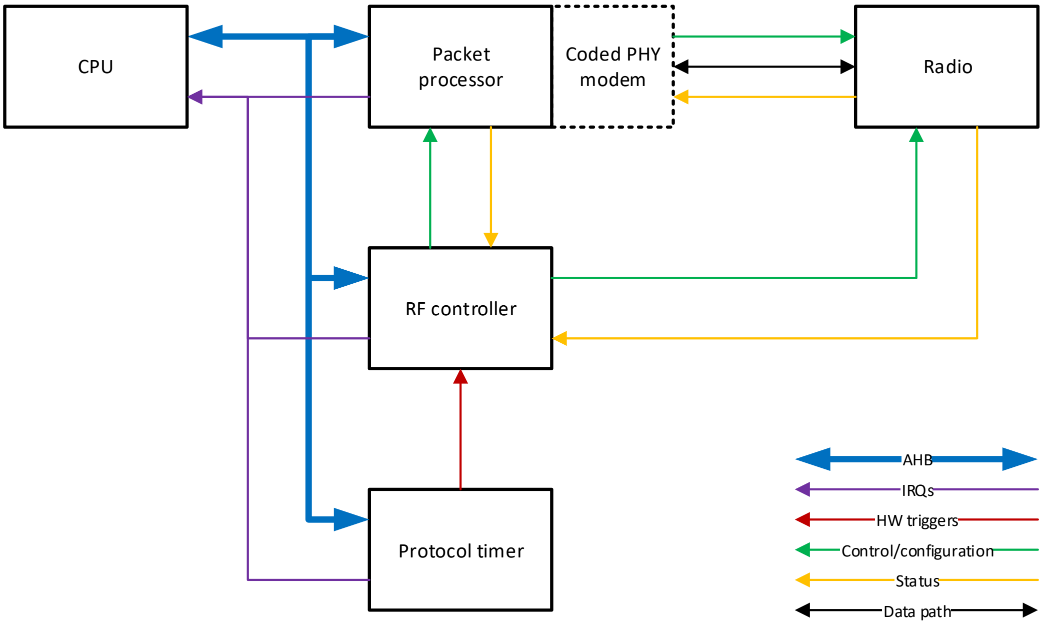


Figure 7-2: Principal connection between the RF controller and surrounding blocks

The RF controller has the following features:

- AHB-SPI bridge to access radio registers for configuration
- Direct control of radio to start/stop given RF operation – i.e. RX or TX
- HW triggers to start radio operation – i.e. RX or TX
  - HW triggers enable/disable
  - SW trigger also possible
- Status indicating state of radio subsystem
  - Radio not running
  - RX running
  - TX running
  - Radio switching from RX to TX mode
  - Radio switching from TX to RX mode
- Interrupts indicating operation status
  - RX sync word received
  - RX timeout
  - RX done (packet received)
  - TX done (packet sent)
- Capturing RSSI values during packet reception
  - RSSI + AGC capture upon sync word detection

The following features are tightly coupled with the RF controller and have the range and granularity shown in Table 4-13:

- Automatic switching between RX and TX mode according to defined timing
  - Independent timing for RX→TX and TX→RX

    - Independent enable for RX→TX and TX→RX
- Configurable timing to start radio in given mode and to start packet processor in given mode with defined delay with respect to radio start (packet processor is started with defined and configurable delay from radio start)
    - Independent delay for RX and TX operation
- Timeout for RX operation
    - If the RX packet (sync word) is not received during timeout, RX operation is stopped
- External PA/LNA control

The signals of External LNA and PA (rf_ext_pa_en) are created according to the timing diagrams below (Figure 7-3). The interval a ←→ b or c ←→ d is set by registers with a granularity of 0.25µs, as specified in Table 4-13.

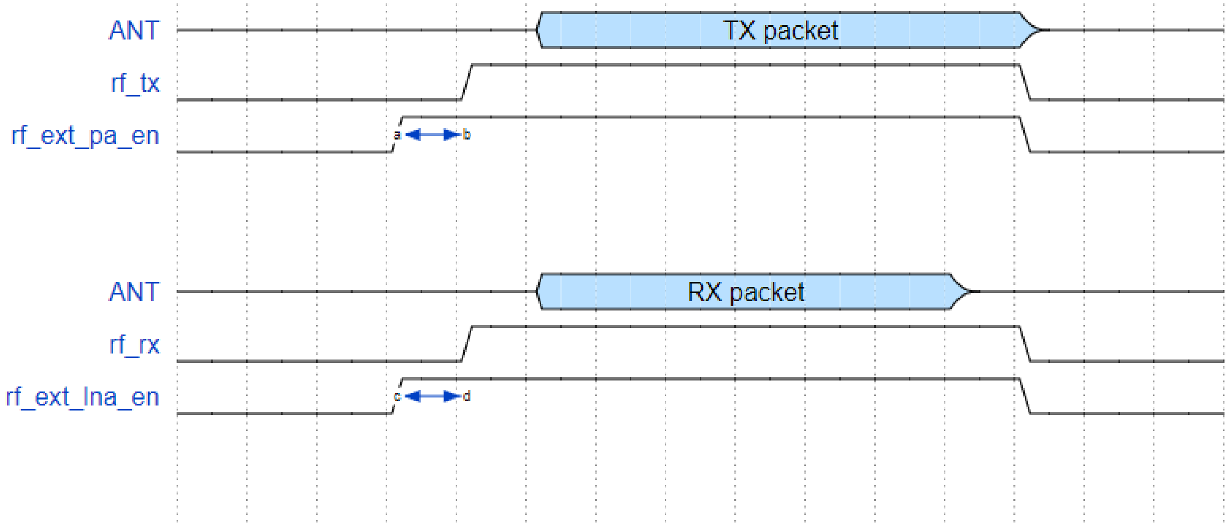


Figure 7-3: External RF control timing

7.3. PACKET PROCESSOR

The packet processor has the following generic features:

- Preamble insertion with configurable length.
- Sync word insertion/detection with configurable length.
- Fixed and variable packet lengths.
- Automatic CRC generation and checking with fully configurable polynomial of up to 32 bits, configurable CRC length, configurable range (Sync word, PDU), CRC calculation before/after whitening.
- Automatic whitening/dewhitening with fully configurable polynomial of up to 32 bits. Configurable whitening range (PDU, CRC), configurable whitening bitstream order for PDU and CRC.
- FIFO for RX and TX data with 32/8-bit (optionally 16 bit) access from CPU.
- Packet encryption (AES-CCM).
- HW and SW triggers to start RX/TX operations.
- 1Mbit/s and 2Mbit/s support for Bluetooth® LE and Bluetooth® LE 2M.
- Asymmetric Bluetooth® PHY support (any PHY combination of 1Mbps, 2Mbps, S=2, S=8 for RX and TX mode).
- 125kbit/s (S=8) and 500kbit/s (S=2) support for Bluetooth® LE Coded.

The generic packet format supported by the packet processor is shown in Figure 7-4.

Preamble	**Syncword**	**PDU**	**CRC**	**CTE**
0-16 Bytes	0-5 Bytes	0-258 Bytes	0-4 Bytes	16-160 µs

Figure 7-4: Generic packet format

The most typical packet formats are described in the next sections. More details can be found in the core or standard documents (see Section 1.4). Other packet formats are possible, also with different encryption modes. Please contact EM Microelectronic for further information about the support of other protocols.

7.3.1. BLUETOOTH® LE AND BLUETOOTH® LE 2M PACKET FORMAT

The packet processor supports the Bluetooth® LE and Bluetooth® LE 2M packet formats shown in Figure 7-5 with configurable length of preamble, sync word, PDU, and CRC. The supported data rates are 1Mbit/s and 2Mbit/s.

Preamble 1-2 Bytes	Access Ad. 4 Bytes	PDU 2-258 Bytes	CRC 3 Bytes	CTE 16-160 µs

Figure 7-5: Packet format for Bluetooth® LE and Bluetooth® LE 2M

The Constant Tone Extension (CTE) field is optional and can be Angle-Of-Arrival (AOA) type or Angle-Of-Departure (AOD) type.

7.3.2. BLUETOOTH® LE CODED PACKET FORMAT

The Bluetooth® LE Coded packet format (Figure 7-6) has two different coding schemes depending on if the data rate is 500kbit/s (S=2) or 125kbit/s (S=8).

Preamble 80 µs	Access Ad. 256 µs	CI 16 µs	TERM1 24 µs	PDU (N Bytes * 8 * S) µs	CRC (24 * S) µs	TERM2 (3 * S) µs

Figure 7-6: Packet format for Bluetooth® LE Coded

The preamble is not coded. The access address, the coding indicator (CI) and the first terminator (TERM1) are always coded using S=8 coding scheme. The PDU, CRC and the second terminator (TERM2) are coded according to the value of CI, using S=2 or S=8 coding scheme.

7.4. DIRECTION FINDING

The principle of the Direction Finding (either Angle-Of-Arrival, or Angle-Of-Departure) is that a transmitter transmits CTE (Constant Tone Extension). During this time, either a transmitter switches signal to different antennas (AOD – see Figure 7-7) or a receiver receives the CTE on different antennas (AOA –see Figure 7-8).

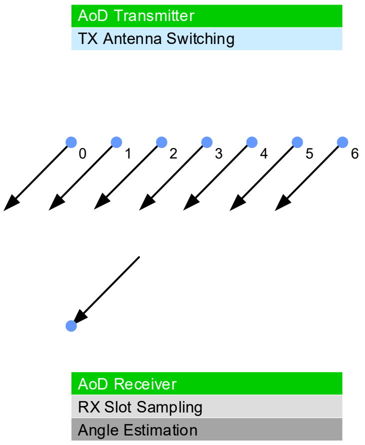


Figure 7-7: Angle-Of-Departure (AOD) system representation

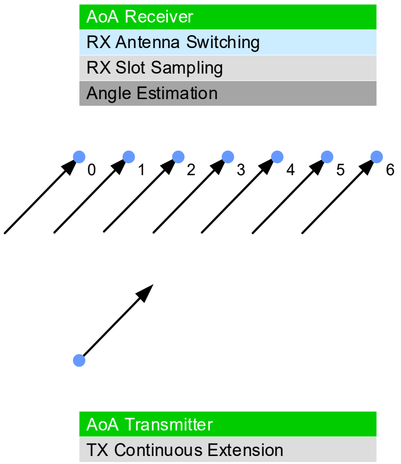


Figure 7-8: Angle-Of-Arrival (AOA) system representation

A localization of departure respectively arrival can be done by analyzing signals from different antennas.

The EM9305 supports this feature by controlling an external antenna switch through GPIOs pins. The switch control is implemented internally into the EM9305 with a switching matrix, with the signal flow being in transmit mode for AOD (Figure 7-9) or in receive mode in AOA (Figure 7-10).

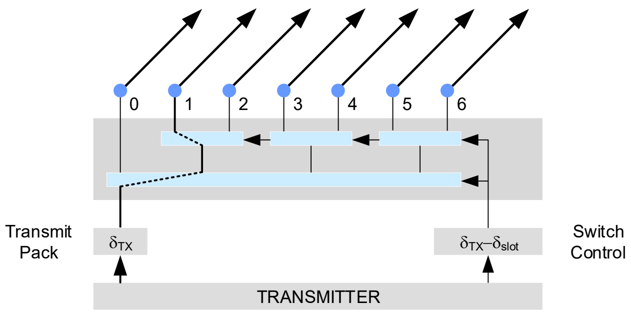


Figure 7-9: AOD TX antenna switching

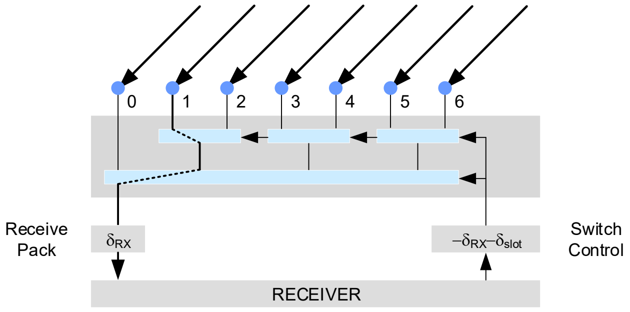


Figure 7-10: AOA RX antenna switching

The Constant Tone Extension period can be 1µs or 2µs. Based on the length of CTE duration there can be up to 74 different antennas switched during CTE plus a default antenna 0, which is enabled before CTE. In the case of AOD, the antenna switching is done on the transmitter, and a slot sampling is necessary on the receiver, as shown in Figure 7-11. For AOA, switching and sampling is done on the receiving side, and the transmitter performs a simple continuous transmission, as shown in Figure 7-12.

Slot			1		2		3		4		...	73		74	
AoD TX 1	Guard Period	Reference Period	Switch	TX	Switch	TX	Switch	TX	Switch	TX	...	Switch	TX	Switch	TX
Time [us]	4	8	1	1	1	1	1	1	1	1	...	1	1	1	1

Slot			1		2		3		4		...	73		74	
AoD RX 1	Guard Period	Reference Period	Switch	Sample	Switch	Sample	Switch	Sample	Switch	Sample	...	Switch	Sample	Switch	Sample
Time [us]	4	8	1	1	1	1	1	1	1	1	...	1	1	1	1

Figure 7-11: TX CTE antenna switching packet format for AOD with 1µs period

Slot			
AoA TX 1	Continuous Transmission		
Time [us]	4	8	4-148

Slot			1		2		3		4		...	73		74	
AoA RX 1	Guard Period	Reference Period	Switch	Sample	Switch	Sample	Switch	Sample	Switch	Sample	...	Switch	Sample	Switch	Sample
Time [us]	4	8	1	1	1	1	1	1	1	1	...	1	1	1	1

Figure 7-12: RX CTE antenna switching packet format for AOA with 1µs period

The block diagram of the switching matrix is shown in Figure 7-13. It consists of a Look-Up Table with 75 entries mapped into the registers with 8 bits control words. The counter with 1µs/2µs period is controlled by the packet processor and it indicates the start and end of CTE. The multiplexer selects one control word to output based on the counter value. An appropriate mapping is then done. The output of the switching matrix is a 12-bit signal each corresponding to one GPIO bit (GPIO0-11). It is possible to assign any LUT bit to any of these 12 outputs.

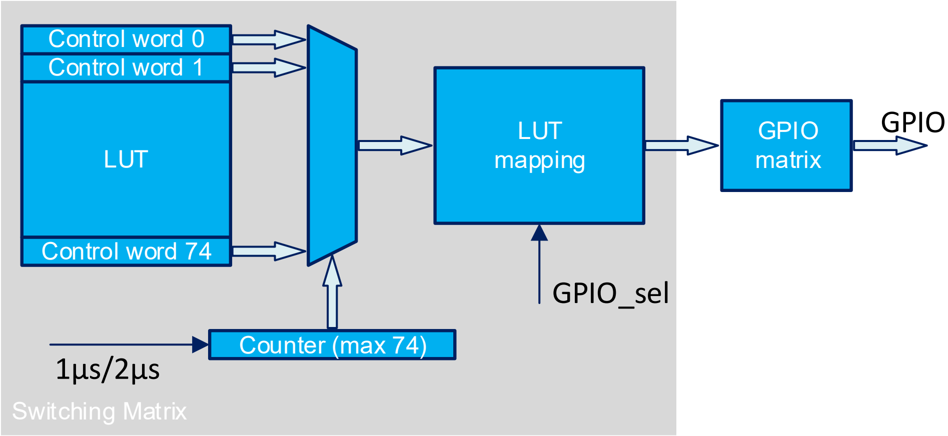


Figure 7-13: Switching matrix block diagram

The following features of the packet processor are used for direction finding applications:

- HW triggers for antennae switching (switching itself done externally).
- Partial packet decoding for CTE info extraction.
- Programmable timers for CTE HW trigger generation and I/Q sampling (granularity better than 125ns).
- I/Q sampling based on received CTE info (in Bluetooth® LE packets) and storing into FIFO.

8. PERIPHERALS

Peripherals include ADC, GPIO, I2C master, I2S/TDM interface, JTAG and cJTAG, PWM, QDEC, SPI master and slave, Temperature indicator, Timers, UART and USB.

The peripherals are listed in Table 8-1. The peripherals can be mapped to several different pins in order to optimize the PCB layout when connecting it to an external MCU, sensor, or other devices to avoid crossed wires when routing the board.

Table 8-1: Peripherals Functions

BLOCK	PADS	COMMENT	SECTION
ADC	Analog input	Analog input to ADC	8.1
CTE	LUT0 to LUT7	Control signal to external CTE device for antenna switching	7.4
GPIO	GPIO0 to GPIO11	General IO function input, output, pull down, pull up	8.2
I2C Master	SCK, SDA	Programmable mapping	8.3
I2S/TDM	SDI, SDO, SCK, MCK, FSYNC	Programmable mapping	8.4
JTAG (4-wires)	TMS, TDI, TDO, TCK	JTAG 1149.1 interface, ARC debugging port	8.5.1
cJTAG (2-wires)	TMSC, TCKC	Compact JTAG 1149.7 interface, ARC debugging port	8.5.2
QDEC	A, B, LED	Programmable mapping	8.6
PML	Mode, Clock	Active or Sleep mode indication. Programmable clock output.	6.4
RF Activity	RX_ON, TX_ON	Output, SW controlled, RF activity monitor	7.2
RF External	PA_Enable, LNA_Enable	For switching external PA/LNA on during TX/RX	7.2
RF IQ	IQ_CLK, IQ_SEL, IQ0 to IQ7	Digital quadrature signals	
SPI Master	CSN, SCK, MISO, MOSI	Programmable mapping	8.7
SPI Slave	CSN, SCK, MISO, MOSI, RDY	Programmable mapping	8.8
UART (1-2-4 wires)	RX, TX, nCTS, nRTS	Programmable mapping	8.11
USB	D+, D-	Fixed mapping to GPIO6 and GPIO7, only on QFN/die	8.12
Timer Start/Stop	Start/Stop	HW start/stop of Universal timer	8.10.1
Timer Capture	Capture	HW capture of Universal timer	8.10.1
Timer Clock Input	Clock Input	External clock of Universal timer	8.10.1
Timer PWM Output	PWM CH0 to CH3	Output of Universal Timer	8.10.1

8.1. ADC

The ADC 9-bit SAR is based on a resistive ladder. The output is digitally corrected by software.

The ADC is connected to GPIO5 via two different inputs: signal or battery. The signal path can use the sample-and-hold function. The battery path is a direct connection to an internal resistive divider.

The sampling rates are shown in Table 8-2 where specifications of the ADC are summarized.

The possible measurements ranges and input sources are shown in Table 8-3.

Table 8-2: ADC specifications

PARAMETER	COMMENTS	MIN	TYP	MAX	UNIT
Resolution			9		bits
INL	After calibration in defined measurement range		±2		LSB
DNL	After calibration in defined measurement range	-2		2	LSB
Offset	After calibration in defined measurement range	-10		10	LSB
Sampling frequency	S&H enabled	45		80	kS/s
	S&H disabled	100		120	kS/s

Table 8-3: ADC sources and measurement ranges

ADC SOURCE	ADC RANGE [V]	HIGH INPUT IMPEDANCE	S&H SELECTABLE	VBAT2 [V]
GPIO5 `(0x0)`	0.020 – 0.5 `(0x0)` 0.030 – 1.0 `(0x1)` 0.035 – 2.0 `(0x2)`	yes	yes	1.1 – 3.6 1.1 – 3.6 1.1 – 3.6 with VBAT2>GPIO5
VCC `(0x1)`	0.96 – 1.86	NA	no	1.1 – 3.6
VBAT1 `(0x2)`	1.82 – 3.5	NA	no	1.8 – 3.6
GPIO5 `(0x3)`	0.96 – 1.86 `(0x0)` 1.82 – 3.5 `(0x1)`	3MΩ	no	1.1 – 3.6 with VBAT2>GPIO5 1.1 – 3.6 with VBAT2>GPIO5

Default calibration is done for ADC source 0x0, range 0x2 and for 9 bits format.

When the source is GPIO5 signal input voltage with sample-and-hold, there is an input buffer with high input impedance. In this case, there is the possibility to internally down-scale the range over 3 programmable maximal levels as shown in Table 8-3. This will allow having a fine precision for low amplitude signals.

The ADC is not used as a permanent supply supervisory system. For this purpose, the SVLD is used (see Section 6.3.1).

The different application examples with different sources are shown in Figure 8-1, Figure 8-2, Figure 8-3 and Figure 8-4.

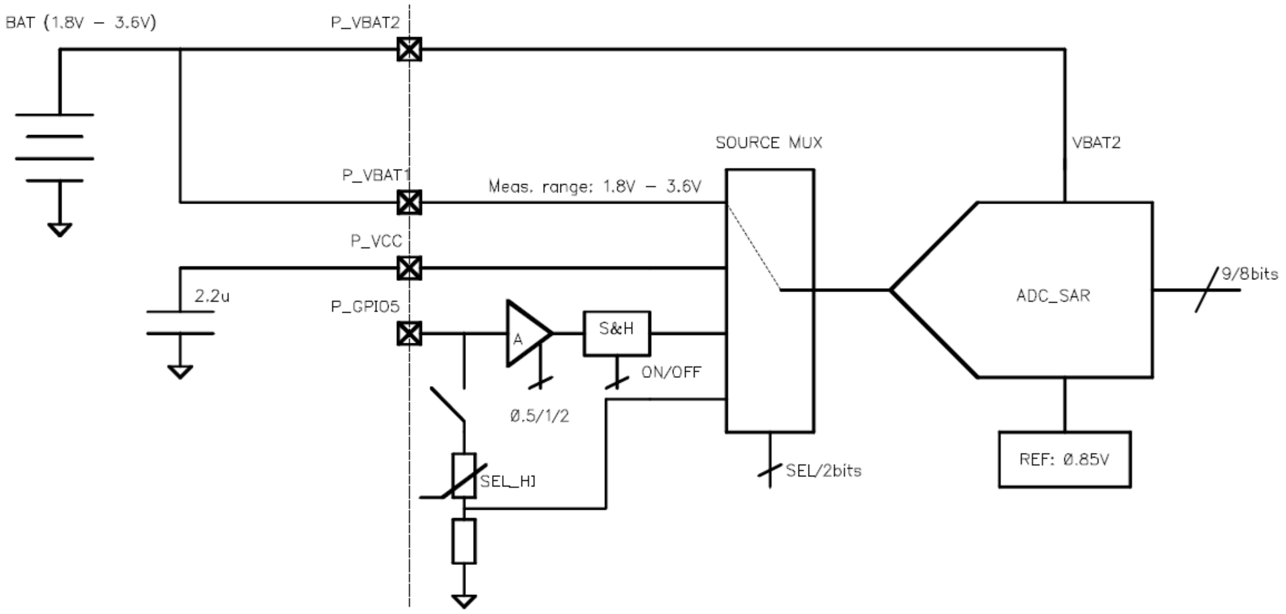


Figure 8-1: ADC application example: VBAT1 measurement

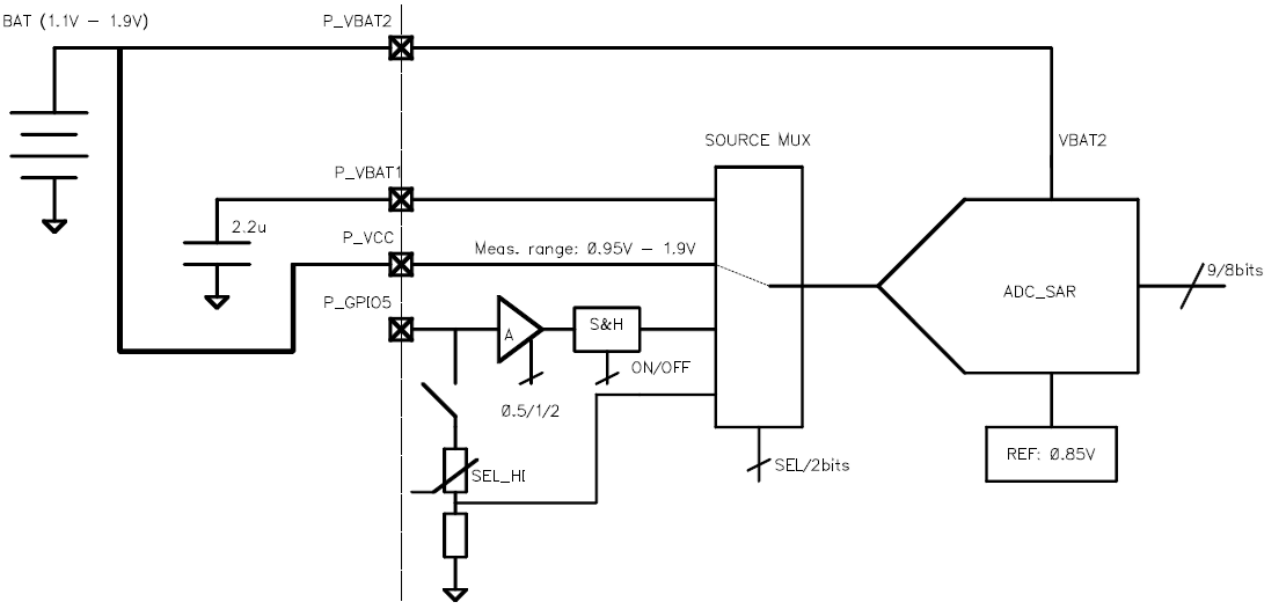


Figure 8-2: ADC application example: VCC measurement

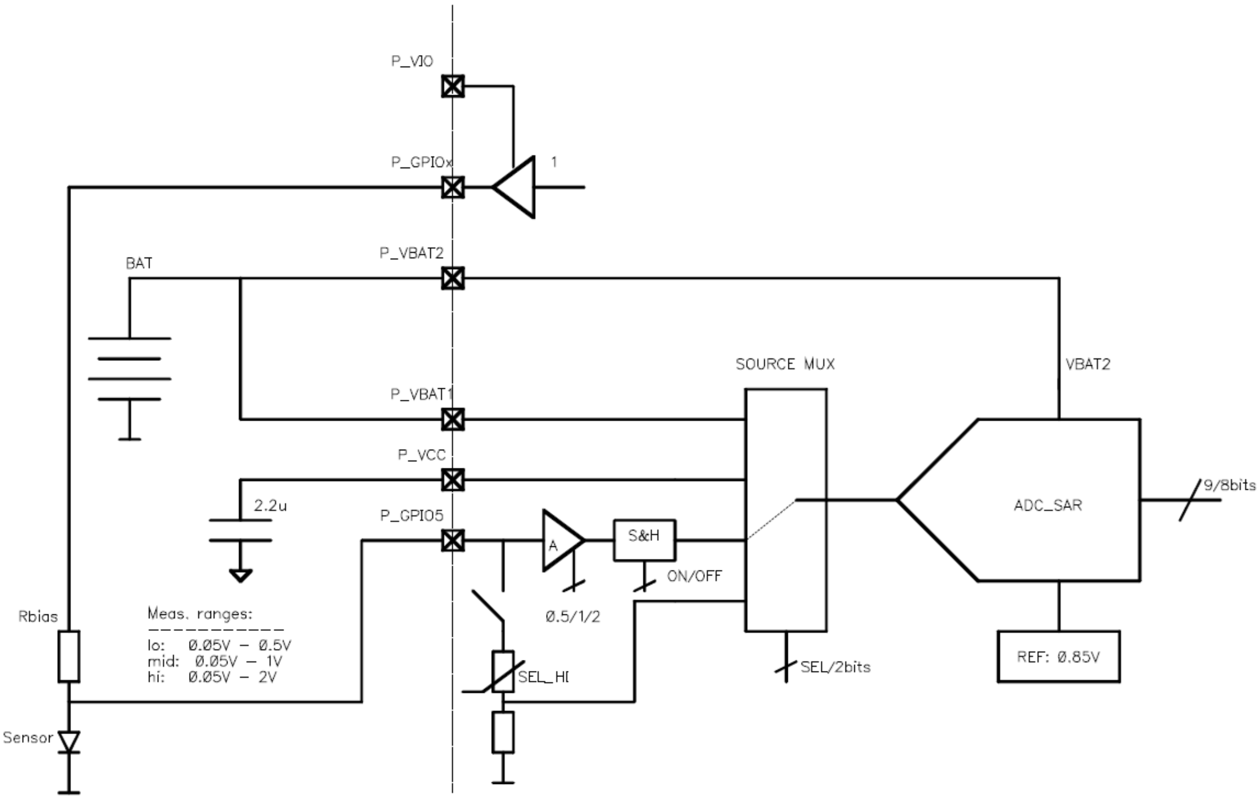


Figure 8-3: ADC application example: GPIO5 measurement with sample and hold

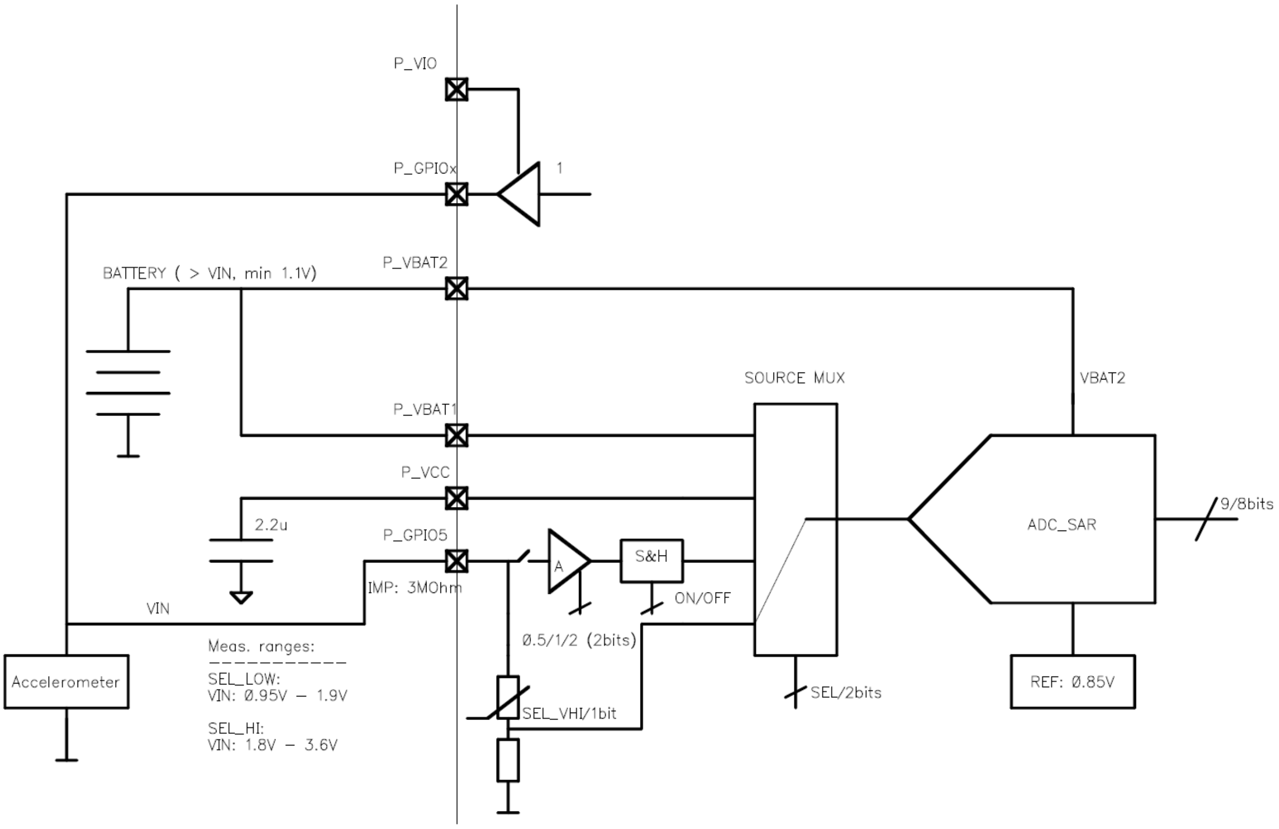


Figure 8-4: ADC application example: GPIO5 measurement without sample and hold

8.2. GPIO

The GPIO are standard input/output structures with high-drive, pull-up or pull-down capability. The block diagram is shown in Figure 8-5. Level shifters are provided between VIO and the internal supply voltage. An optional input debouncer is clocked by the LF RC oscillator divided by a selectable value between 16 and 2048. Any GPIO can interrupt the CPU on any input value and are maskable.

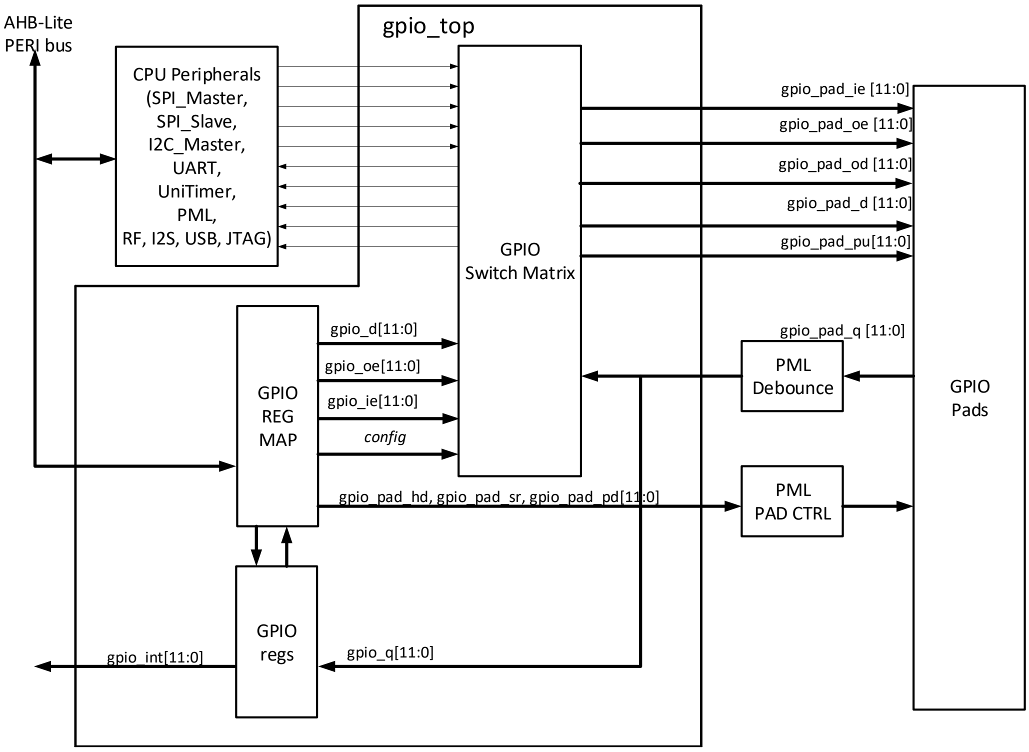


Figure 8-5: GPIO block diagram

The purpose of the GPIO Switch Matrix is to select between general I/O functionality and a peripheral function for a given GPIO pin. The matrix controls pad data signals, pad input enable and pad output enable. Debouncer, pull-down, pull-up is done only by GPIO register setting.

There are two matrices, one for input, and one for output connection. One GPIO pad can be connected as input and output at the same time, which creates a possibility of an internal loopback.

Each GPIO pin contains the following features:

- Direction selection input/output
- standard and high drive capability (only for GPIO1-GPIO2-GPIO3 and GPIO6-GPIO7)
- input debouncer; clock is LF RC, selectable division from /16 to /2048, enabled/disabled by GPIO
- Tri-state output with selectable pull-up or pull-down control
- Interrupts generated by events on GPIO (masking implemented as part of Interrupt Manager)
- Selectable polarity of GPIO interrupt.

Electrical characteristics of the GPIO pins are reported in Table 4-8.

A GPIO interrupt is edge sensitive for an active device. Depending on how it is configured, the interruption is triggered either on a leading edge or on a falling edge of the GPIO input signal. Thus, GPIO input signals are synchronized to the clock domain which means that whenever a GPIO state change happens, the related interruption is generated to be synchronized with the leading edge of the clock.

The Figure 8-6 depicts the interruption generation based on a GPIO state change. As can be seen, there is a delay between the moment when the GPIO state changes and the next leading edge of the clock at which the interruption will be generated.

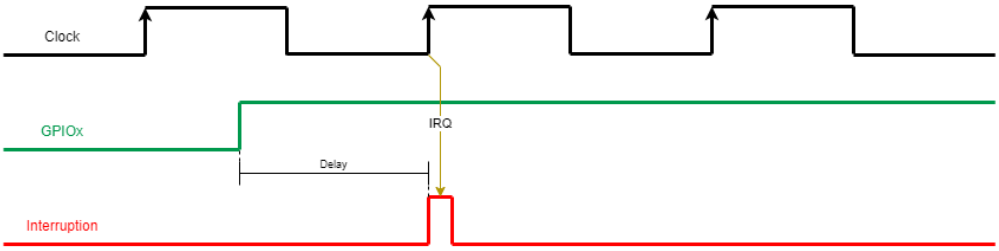


Figure 8-6: GPIO interrupts synchronization

However, the way the device is woken up when in sleep mode behaves differently. The GPIO pads have configurable wake-up circuitry which is level sensitive. They can be configured to wake-up the device on either a low level or on a high level.

When a GPIO level switches to the "wake-up" level, the device will become active. During the startup process, this trigger is translated to a leading edge (or a falling edge) in order to trigger an interrupt that will be caught by the running software.

The Figure 8-7 depicts this behavior.

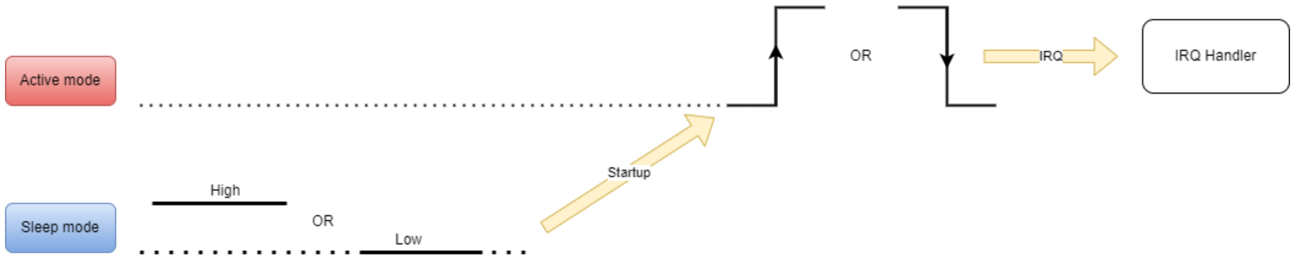


Figure 8-7: GPIO wake-up

This means that if the device is configured to be woken up with a high level on one specific GPIO, the software shall ensure that this level is low before switching to sleep mode.

In case this rule is not observed, i.e. the GPIO level already has the "wake-up" level before switching to sleep mode, the device will immediately wake-up from low power after it has switched to sleep mode. Seen from outside of the device, it looks like the device does not switch to sleep and stays in active mode. However, it is not the case, and the device actually switches to sleep mode.

In all cases, the CPU restarts executing the ROM code startup and an interruption is triggered which is the regular behavior. Of course, in order to process this interruption, the software must have already registered a dedicated interruption handler.

After HW reset, all GPIO are in disable state (input blocked, output in Hi-Z, no pull resistors). This avoids any unwanted interactions during power-up. The exception is GPIO8, which makes pull-down, pull-up, pull-down sequence to indicate POR.

The default configuration upon software boot is determined by whether we enter Configuration mode or jump into user applications. If we jump into the user application, the ROM does not configure any GPIO. All GPIO registers are set to 0, allowing the user to configure their own GPIO settings. However, if we enter Configuration mode, the ROM configures the GPIO for SPI communication, with CSN, SCK, and MOSI enabled as inputs, and MISO and RDY enabled as outputs. There are no pull-up or pull-down resistors configured in either case.

The default configuration after SW boot in Configuration mode is shown in Table 8-4.

The default hardware GPIO configuration is applied briefly during start-up and after chip reset until the new configuration is read from flash and applied to the GPIO.

GPIO1-GPIO2-GPIO3 and GPIO6-GPIO7 have a programmable output buffer drive strength to allow fast SPI and USB.

Table 8-4: GPIO status after SW boot in Configuration Mode: SPI Slave (green cells) – GPIO (purple cells)

GPIO #											
0	1	2	3	4	5	6 *	7 *	8	9	10	11
CSN-S	SCK-S	MISO-S	MOSI-S	RDY	GPIO5	GPIO6	GPIO7	GPIO8	GPIO9	GPIO10	GPIO11

* Not available on WLCSP package version.

The GPIO configuration options are detailed in Table 8-5. The peripherals can typically be mapped to several possible GPIO. If a GPIO can be configured as an input, it is marked with an “I” at the appropriate column and the function is shown in the corresponding row. Likewise, if a GPIO can be configured as an output it is marked with an “O”, and if it can be both an input and output it is marked as “IO”.

Table 8-5: GPIO configuration

PERIPHERAL	FUNCTION	GPIO #											
		0	1	2	3	4	5	6 *	7 *	8	9	10	11
ADC	IN						I						
CTE	LUT0 – LUT7	O	O	O	O	O	O	O	O	O	O	O	O
GPIO	GPIO0 – GPIO11	IO	IO	IO	IO	IO	IO	IO	IO	IO	IO	IO	IO
I2C Master	SCK	IO				IO					IO	IO	
	SDA		IO				IO			IO		IO	IO
I2S/TDM	SDATA_I	I	I	I	I	I				I	I	I	I
	SDATA_O	O	O	O	O	O				O	O	O	O
	SCK	IO	IO	IO	IO	IO				IO	IO	IO	IO
	MCK	O	O	O	O	O				O	O	O	O
	FSYNC	IO	IO	IO	IO	IO				IO	IO	IO	IO
JTAG (4-wires)	TMS												I
	TDI									I			
	TDO										O		
	TCK											I	
cJTAG (2-wires)	TMSC												IO
	TCKC											I	
QDEC	A	I	I	I	I	I	I	I	I	I	I	I	I
	B	I	I	I	I	I	I	I	I	I	I	I	I
	LED	O	O	O	O	O	O	O	O	O	O	O	O
PML	Mode	O	O	O	O	O	O	O	O	O	O	O	O
	Clock	O	O	O	O	O	O	O	O	O	O	O	O
PTM (test)	Clock						I						
RF	RX_ON	O	O	O	O	O	O	O	O	O	O	O	O
	TX_ON	O	O	O	O	O	O	O	O	O	O	O	O
	PA_Enable	O	O	O	O	O	O	O	O	O	O	O	O
	LNA_Enable	O	O	O	O	O	O	O	O	O	O	O	O
	IQ_CLK			O									
	IQ_SEL				O								
	IQ0 – IQ7					O	O	O	O	O	O	O	O
SPI Master	CSN [8]	O	O	O	O	O	O	O	O	O	O	O	O
	SCK		O			O					O	O	
	MISO			I	I						I	I	
	MOSI			O	O					O		O	O
SPI Slave	CSN	I				I					I		I
	SCK		I							I			
	MISO			O	O						O	O	
	MOSI			I	I						I	I	
	RDY		O			O	O	O	O				
UART (1-wire)	RX/TX	IO	IO										
UART (2-wires or 4-wires)	RX	I	I	I	I	I		I					
	TX	O	O	O	O				O	O			
	nCTS		I	I	I	I				I	I	I	
	nRTS	O	O	O	O				O			O	
USB	D+								IO				
	D-							IO					
Timer2	Start/Stop	I	I	I	I	I	I	I	I	I	I	I	I
	Capture	I	I	I	I	I	I	I	I	I	I	I	I
	Clock	I	I	I	I	I	I	I	I	I	I	I	I
	PWM CH0	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH1	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH2	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH3	O	O	O	O	O	O	O	O	O	O	O	O
Timer3	Start/Stop	I	I	I	I	I	I	I	I	I	I	I	I
	Capture	I	I	I	I	I	I	I	I	I	I	I	I
	Clock	I	I	I	I	I	I	I	I	I	I	I	I
	PWM CH0	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH1	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH2	O	O	O	O	O	O	O	O	O	O	O	O
	PWM CH3	O	O	O	O	O	O	O	O	O	O	O	O

* Not available on WLCSP package version.

Even with a reduced number of GPIO with respect to the QFN version, the WLCSP package allows the simultaneous use of different interfaces. To help the user, some of the possible combinations are shown from Table 8-6 to Table 8-12. Other combinations are possible.

[8] CSN is not part of the SPI Master block. The user must use GPIO function for this purpose.

Table 8-6: SPI Slave with high drive capability (green cells) – SPI Master (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
CSN-S	SCK-S	MISO-S	MOSI-S	RDY		CSN-M	SCK-M	MISO-M	MOSI-M

Table 8-7: SPI Master with high drive capability (green cells) – SPI Slave (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
CSN-M	SCK-M	MISO-M	MOSI-M	RDY		SCK-S	MOSI-S	MISO-S	CSN-S

Table 8-8: SPI Master with high drive capability (green cells) – UART (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
CSN-M	SCK-M	MISO-M	MOSI-M	RX		TX	nCTS	nRTS	

Table 8-9: SPI Slave with high drive capability (green cells) – UART (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
RX	SCK-S	MISO-S	MOSI-S	RDY		TX	nCTS	nRTS	

Table 8-10: I2S/TDM (green cells) – I2C Master (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
SDI	SDO	SCK$_{I2S}$	MCK	FSYNC				SCK$_{I2C}$	SDA

Table 8-11: SPI Master with high drive capability (green cells) – I2S/TDM (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
CSN-M	SCK-M	MISO-M	MOSI-M	SDI		FSYNC	MCK	SCK	SDO

Table 8-12: SPI Slave with high drive capability (green cells) – I2S/TDM (purple cells) combination on WLCSP device

GPIO #									
0	1	2	3	4	5	8	9	10	11
CSN-S	SCK-S	MISO-S	MOSI-S	SDI	RDY	FSYNC	MCK	SCK	SDO

8.3. I2C MASTER

The I2C Master interface supports the following features:

- Master mode only.
- Supported speeds:
  - Standard mode (up to 100 kbit/s).
  - Fast mode (up to 400 kbit/s).
  - Timing is configurable by register.
- Clock stretching
- Addressing modes:
  - 7-bit device addressing mode.
- I2C enable/disable.
- 16 bytes long RX buffer for reception and 16 bytes long TX buffer for transmission.
- I2C transactions:
  - Write:
    1. start bit,
    2. device address,
    3. memory/register address to write,
    4. data to write (N bytes),
    5. stop bit
  - Simple read:
    1. start bit,
    2. device address,
    3. data to read (N bytes),
    4. stop bit
  - Combined read:
    1. start bit,
    2. device address,
    3. memory/register address to read,
    4. repeated start bit,
    5. device address,
    6. data to read (N bytes),
    7. stop bit
  - I2C Device address in separate register.
  - Optional memory/register address to read/write in a separate register, 1 byte long.
  - Number of bytes to send/receive is 0-16. Sending/receiving started by writing to control register, stop condition can be omitted allowing sending the next byte sequence in the same I2C transaction.
- Number of bytes sent/received in last transaction.
- Stopping current transaction (after finishing current byte).
- Status flags:
  - I2C transaction status (busy flag)
  - Start condition detected
  - Stop condition detected
  - No ACK detected
  - Slave address + read/write sent
  - Clock is currently stretching by slave
  - Pause status (end of bytes transfer, waiting for the next transfer; no Stop condition sent)
- IRQs:
  - End of sequence

The electrical characteristics for GPIO pins capable of an I2C interface are shown in Table 4-8 and Table 4-9. Based on the load and required speed, an external pull-up might be needed to lower the internal pull-up of the chip.

The timing characteristics are according to I2C bus specification Rev.6 and otherwise they are specified as in Table 4-14.

8.4. I2S/TDM

The I2S/TDM interface is mainly dedicated to audio applications. It is a synchronous interface, which can be configured as master or slave and supports the following features:

- Master and Slave mode
- Frame length programmable for the following runtime configurable formats:
  - TDM
  - I2S
  - Left-Justified
  - Right-Justified
  - TDM in I2S
- Full-duplex operation

- Up to 2 phases per frame, with up to 128 channels per phase
- Programmable data delay
- Channel order loss reporting
- Audio sample FIFO with:
    - Pre-synthesis configurable depth
    - Status and number of samples reporting
    - Runtime configurable threshold reporting
    - Programmable MSB or LSB first serial audio data transfer
- Programmable audio sample size: 8, 12, 16, 20, 24 or 32 bits
- Programmable sample rate: 8, 11.025, 16, 22.050, 24, 32, 44.100, 48, 88.200, 96 or 192 kHz

The architecture of the serial audio interface is shown in Figure 8-8.

The different frame formats are shown from Figure 8-9 to Figure 8-13.

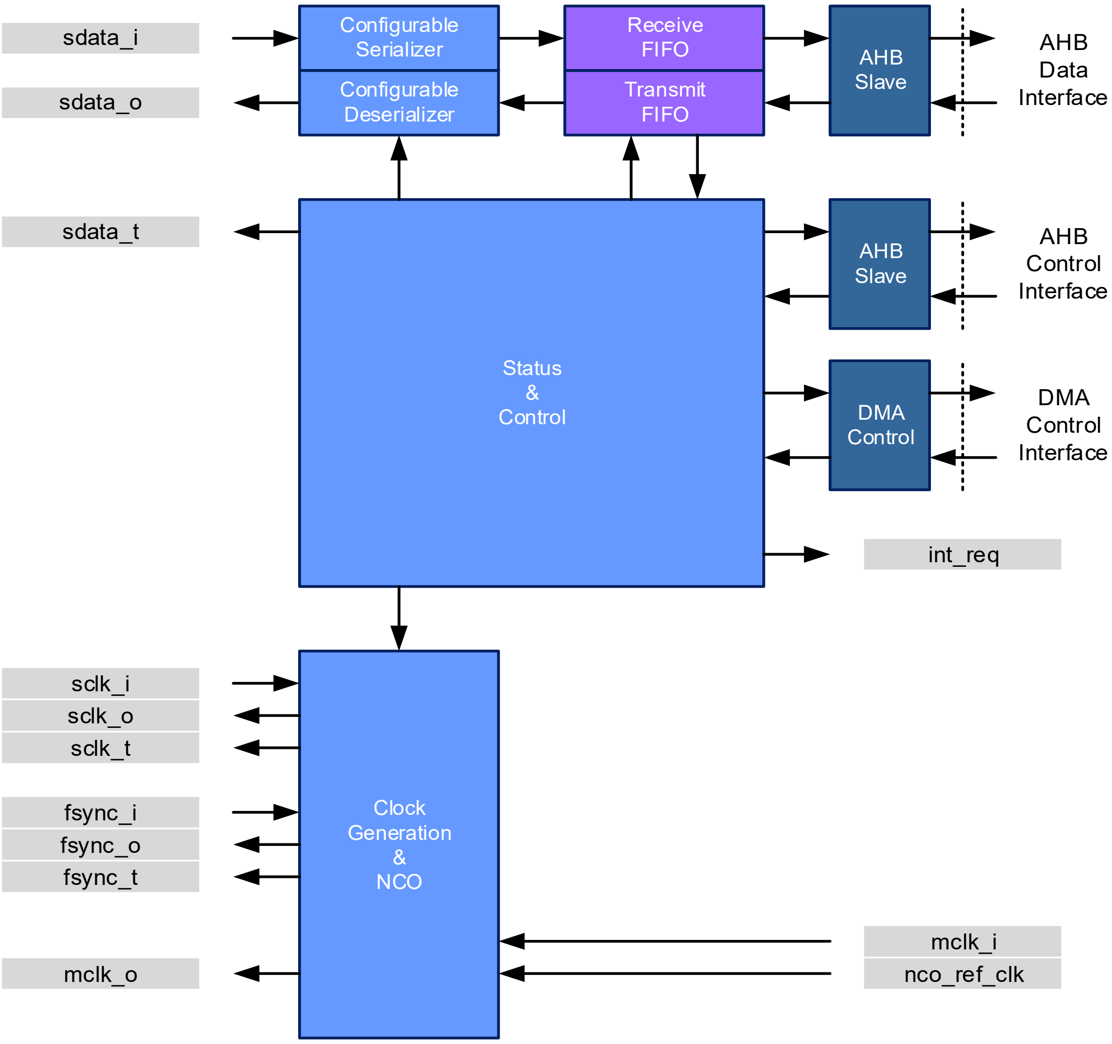


Figure 8-8: Architecture of the serial audio interface I2S/TDM

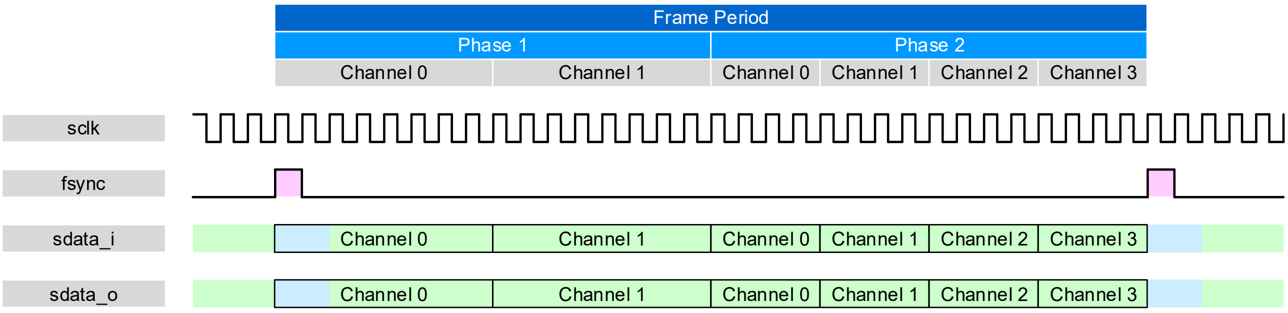


Figure 8-9: TDM frame format

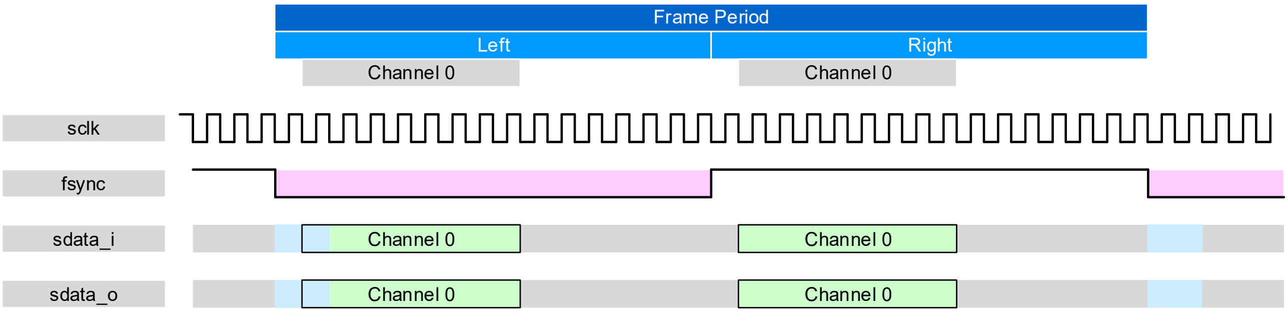


Figure 8-10: I2S frame format

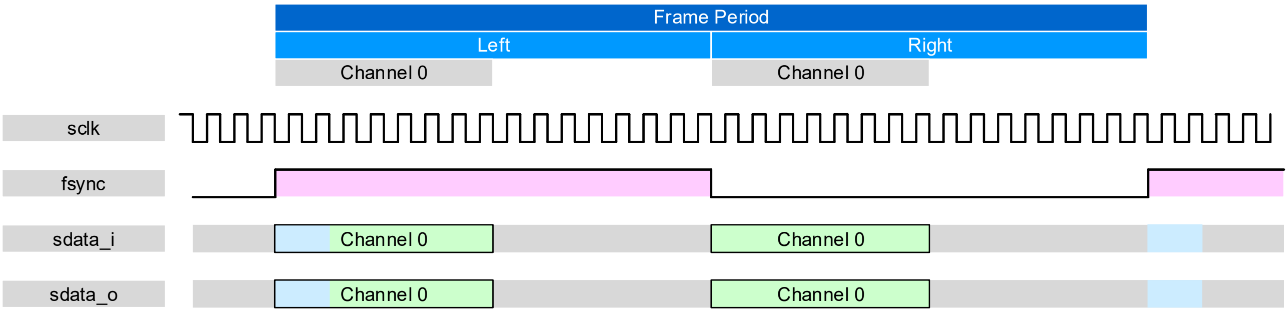


Figure 8-11: Left-Justified frame format

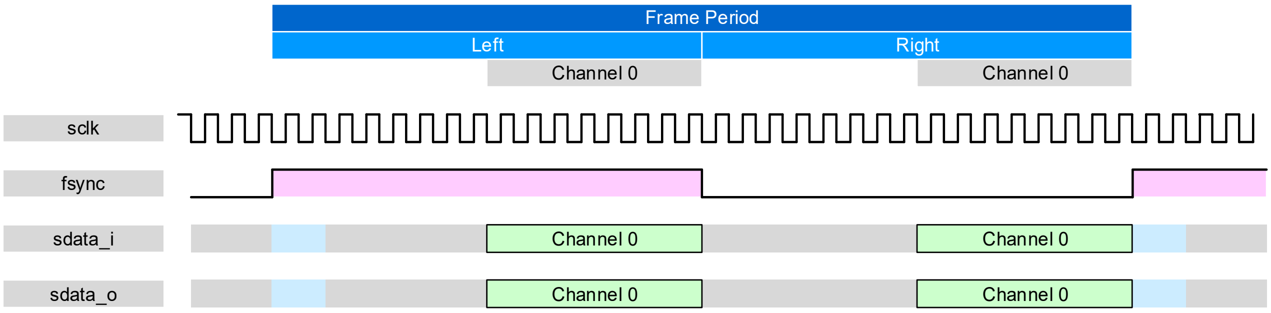


Figure 8-12: Right-Justified frame format

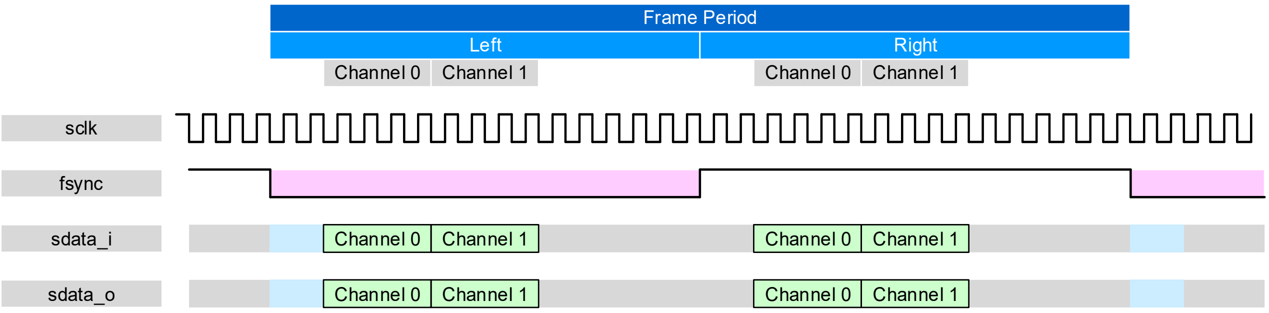


Figure 8-13: TDM in I2S frame format

8.5. JTAG

EM9305 has support for the JTAG interface in order to debug and test prior to release the final application. The 4-wire/2-wire configuration is automatically selected by enable registers. The JTAG port is also known as TAP (Test Access Port).

Maximum JTAG clock speed is 6MHz.

8.5.1. JTAG (IEEE 1149.1)

JTAG is specified by the standards IEEE 1149.1 and the interface is shown in Figure 8-14. This interface allows multiple devices to be operated with a single programmer using a daisy-chain connection.

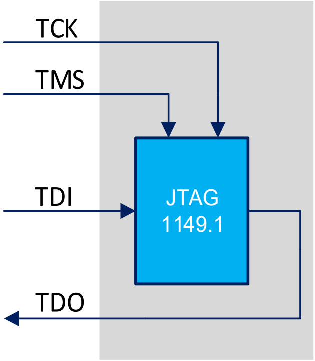


Figure 8-14: JTAG (IEEE 1149.1) interface

8.5.2. cJTAG (IEEE 1149.7)

The Compact JTAG (cJTAG) also known as 2-wire JTAG includes a reduced pin count interface that manipulates the regular 4-wire JTAG state machine with only TCKC and TMSC signals to transmit/receive instructions and data.

The Compact JTAG is an adapter added as a separate component in front of JTAG (IEEE 1149.1) ports of the processor (Figure 8-15). Multiple devices can still be operated using a single programmer using a star connection topology.

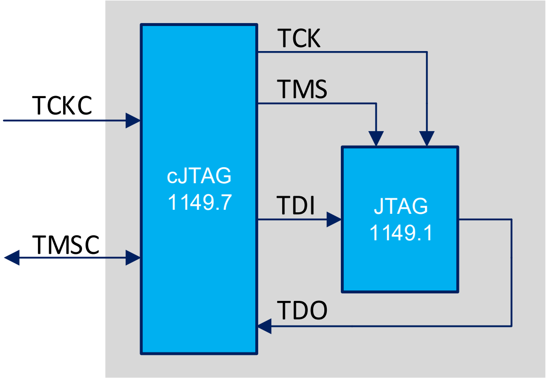


Figure 8-15: cJTAG (IEEE 1149.7) interface

8.6. QDEC

The quadrature decoder (QDEC) is a mouse roller movement detector suitable for mechanical and optical sensors. The concept of the rotary decoder is shown in Figure 8-16.

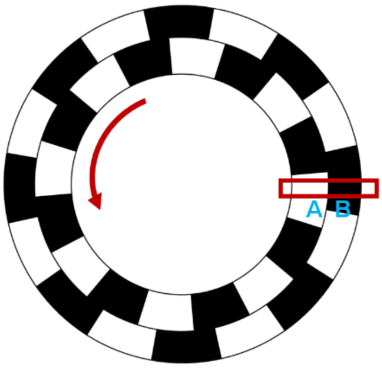


Figure 8-16: Rotary decoder with corresponding logic states for A and B signals

The QDEC is implemented as a separated logic block, capable of running in Active and Sleep mode. It contains also LED output control. It uses a 500kHz clock, so it is not compatible with Deep Sleep mode.

The QDEC (Figure 8-17) consists of:

- Decoding logic, which computes the direction of motion according to inputs from sensor.
- Debouncers: each channel (A, B) has separate debouncer. This debouncer has same debounce time as sampling time.
- LED driver for optical sensor, which is synchronous with sampling period.
- clk_ctrl: in active mode, QDEC is clocked with synchronous clock from AHB. In sleep mode, the PML clock is used.

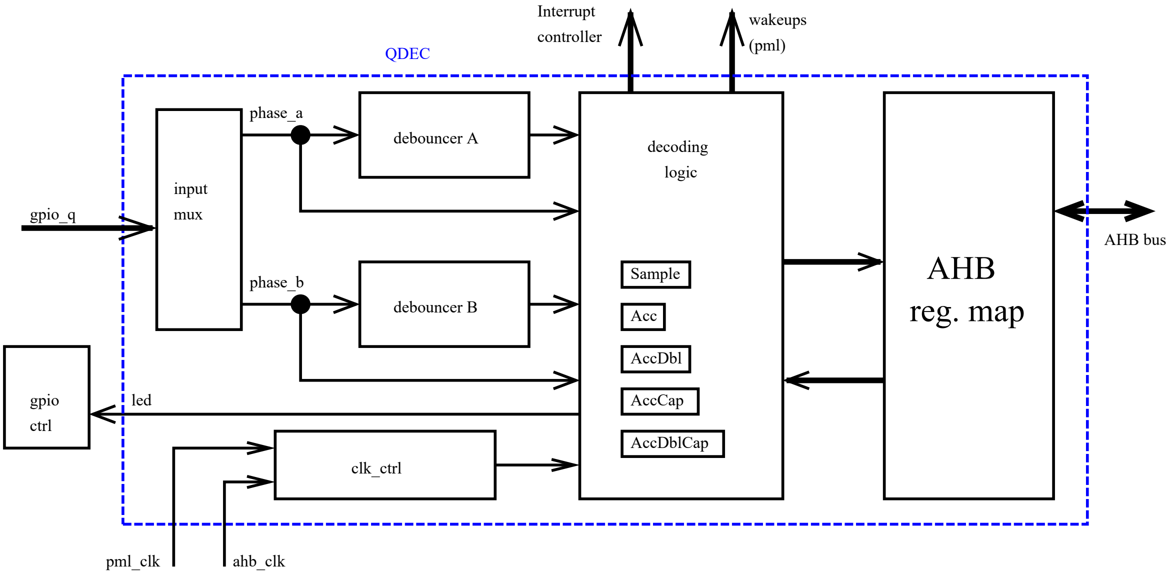


Figure 8-17: QDEC block diagram

The direction of the movement (QDecSample) is computed according to current and previous state of inputs A, B (Table 8-13). The inputs are sampled with a period that can be selected from those shown in Table 8-14.

Table 8-13: Movement detection depending on A, B signals

Prev A	Prev B	Curr A	Curr B	QDecSample	DESCRITPION
0	0	0	0	0	No movement
0	0	0	1	+1	Movement in positive direction
0	0	1	0	-1	Movement in negative direction
0	0	1	1	+2	Error (double movement)
0	1	0	0	-1	Movement in negative direction
0	1	0	1	0	No movement
0	1	1	0	+2	Error (double movement)
0	1	1	1	+1	Movement in positive direction
1	0	0	0	+1	Movement in positive direction
1	0	0	1	+2	Error (double movement)
1	0	1	0	0	No movement
1	0	1	1	-1	Movement in negative direction
1	1	0	0	+2	Error (double movement)
1	1	0	1	-1	Movement in negative direction
1	1	1	0	+1	Movement in positive direction
1	1	1	1	0	No movement

Table 8-14: QDEC sampling periods

QDecSamplePer	PERIOD OF SAMPLING
0	128µs
1	256µs
2	512µs
3	1024µs
4	2048µs
5	4096µs
6	8192µs
7	16384µs
8	32768µs
9	65536µs
10-15	131072µs

If a single movement is detected, a report is generated. The report period can be selected accordingly to Table 8-15.

Table 8-15: QDEC report period values

QDecReportPer	SAMPLES/REPORT
0	1
1	5
2	10
3	20
4	40
5	80
6	160
7	320

The QDEC LED driver is synchronous to the sampling period. It can be activated permanently or a selectable time before the sample is finished, as in Figure 8-18.

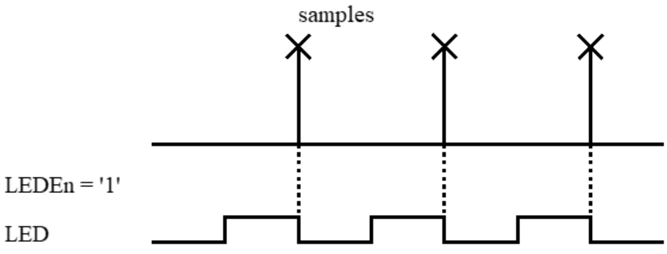


Figure 8-18: QDEC LED enable before sample

8.7. SPI MASTER

The SPI master block supports the following features:

- 4 wire SPI interface (SCK, MISO, MOSI, CSN).
- 3 wire SPI interface (SCK, MOSI as shared MISO/MOSI, CSN).
- Chip select (CSN) to peripherals are generated through the GPIO by software.
- Full duplex communication 4 wire SPI.
- Half duplex communication 3 wire SPI.
- Configurable clock speed: 24MHz, 16MHz, 12MHz, 8MHz, 6MHz, 3MHz, 1.5MHz, 0.75MHz.
- Motorola compliant, all 4 SPI clock polarity/phase configurations supported (CPOL = 0,1; CPHA = 0,1).
  - CPHA is forced at 0 for 16MHz.
- SPI enable/disable.
- 16 bytes receive and 16 bytes transmit buffer.
- 4 bytes receive and 4 bytes transmit command buffer.
- The number of bytes sent/received can be 1-16.
- The SPI bit-order can be configured (MSB or LSB first).
- SPI transaction is started by writing into control register.
- One shot mode: number of bytes to send/receive is configured in control register, together with SPI transaction start register.
- SPI transactions can be stopped or aborted by writing into the control register.
- An interrupt is generated at the end of SPI transaction.
- SPI status register has the following flags:
  - busy flag – SPI transaction is running
  - stop flag – pending request to SPI transaction stop
  - byte count – progress of current SPI transaction

The SPI modes that are supported are shown in Figure 8-19.

Note that the SPI Master must set a pull resistor to MISO until RDY=1 or tolerate a High-Z state on MISO.

The SPI Master timing signals are shown in Figure 4-2. The SPI timing specifications are shown in Table 4-15.

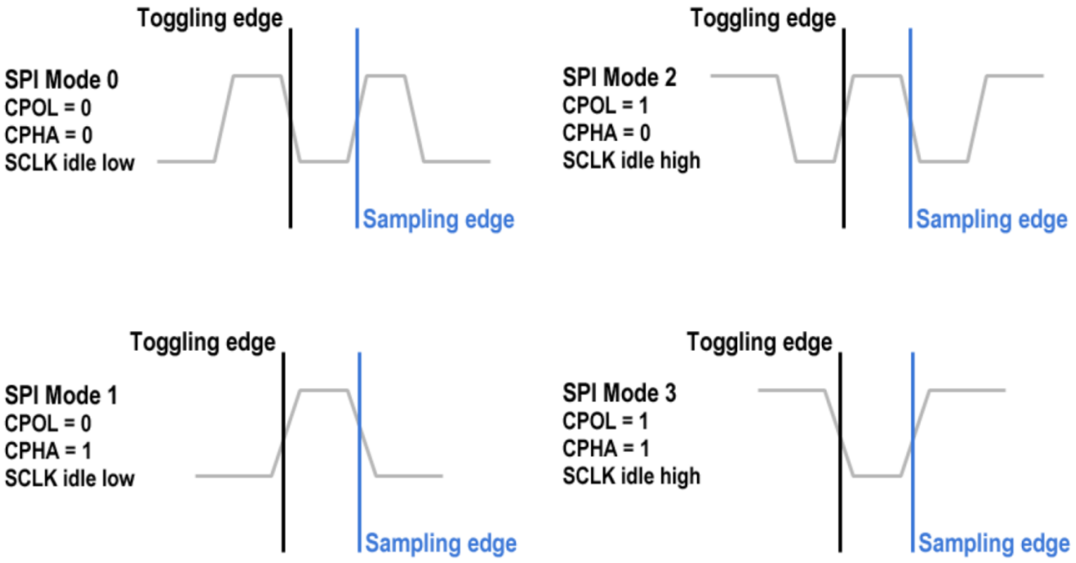


Figure 8-19: SPI modes

8.7.1. SPI MODIFIED PROTOCOL

To achieve the required SPI maximum frequency, the SPI sampling edge and toggling edge are modified. By default, the GPIOs used for SPI are in low drive mode.

On the Master, the modification means that MISO input is sampled on the other edge than the protocol requires (the edge where data are changed). See blue arrow on Figure 8-20.

This means that the whole path SCK => Slave => MISO is limited by the whole clock period, removing also the influence of the duty cycle.

On the Slave, the modification means that MISO output is changed on the other edge than protocol requires (the edge where data are sampled). See black arrow on Figure 8-20.

This means that the internal slave delay is shorted by the clock half period.

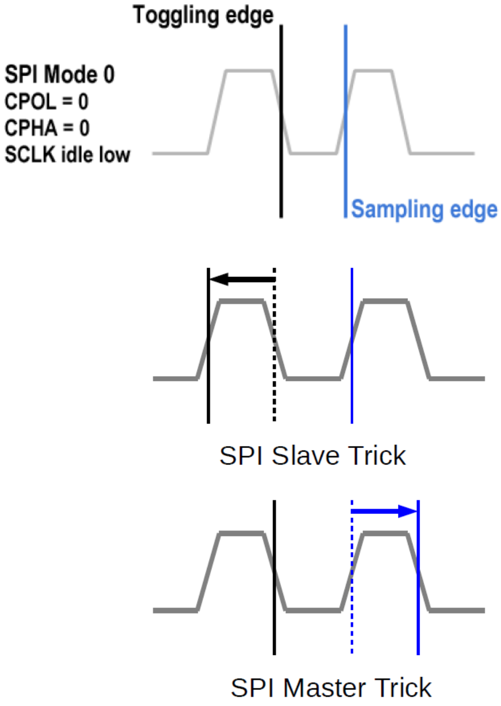


Figure 8-20: SPI trick to increase maximum speed

8.8. SPI SLAVE

The SPI slave block supports the following features:

- 4 wire SPI interface (SCK, CSN, MISO, MOSI) with flow control (RDY signal)
- Half duplex communication. Direction (write/read) is determined by a control byte
- Supported speed up to 16MHz.
- Motorola compliant, clock polarity CPOL = 0 (clock is inactive low), clock phase CPHA = 0 (data is valid on clock rising edge).
- All 4 SPI clock polarity/phase configurations.
- SPI enable.
- Flow control enable/disable.
- 16 bytes long RX FIFO for reception and 16 bytes long TX FIFO for transmission.
- CPU reads/writes (via AHB Lite bus) 1 byte, or 2 bytes, or 3 bytes, or 4 bytes from/to FIFOs by different register address
- FIFO status is available for CPU:
  - TX FIFO empty.
  - TX FIFO limit (less bytes in TX FIFO than user defined limit).
  - RX FIFO limit (more bytes in RX FIFO than user defined limit).
  - RX FIFO full.
  - Number of bytes in each FIFO.
- FIFO flush to remove all content. Each FIFO independently.
- IRQs:
  - Only three interrupt vectors. One group for TX interrupts, one group for RX interrupts, and one group for error interrupts:
  - TX interrupts:
    - Byte sent
    - TX FIFO empty
    - TX FIFO limit
  - RX interrupts:
    - Byte received
    - RX FIFO full
    - RX FIFO limit
  - Error interrupts:

  - TX FIFO underflow
  - RX FIFO overflow
  - Configurable bit order (LSB first or MSB first).
- Multi byte transactions (without de-asserting CSN between bytes)

The SPI Slave timing signals are shown in Figure 4-1. The SPI timing specifications are shown in Table 4-15.

8.8.1. SPI HALF-DUPLEX OPERATION

Half-duplex mode is the default one for HCI communication.

Each SPI transaction starts by activating CSN. After setting CSN to '0', RDY goes to '0' to indicate that SPI interface is not ready for transfer. Once RDY is at '1' it indicates that the SPI interface is ready for transaction and header bytes can be sent.

The control byte (CTRL) sent on MOSI defines the type of transaction (read or write). The second byte on MOSI is dummy to align with read status data from the Slave and reads 0x00.

The status byte STS1 on MISO indicates the general status of device. The status byte STS2 on MISO indicates buffer capacity and it depends on read or write transaction. In case of write transaction, STS2 contains the number of bytes that can be written into Slave RX buffer. In case of read transaction, STS2 contains the number of bytes that can be read from Slave TX buffer.

After receiving header bytes (STS1 and STS2), the SPI Master knows how many bytes can be transferred via SPI in the current transaction (read or write) and the rest of SPI transaction is standard.

An SPI transaction can be terminated by the SPI Master at any time. The SPI Master can send only a header in order to get the status of the RX/TX buffer and then stop the SPI transaction.

The type of transaction (read or write) in half duplex mode is defined by CTRL, the 1st byte sent by the Master:

- CTRL = 0x81 for a read transaction.
- CTRL = 0x42 for a write transaction.

The STS1 byte contains the status of the SPI slave.

- STS1 = 0xC0 when slave is ready.

The STS2 byte contains the maximum number of bytes that can be written into RX FIFO without RX FIFO overflow during a write transaction, or the maximum number of bytes that can be read from TX FIFO without TX FIFO underflow during a read transaction.

8.8.2. SPI FULL-DUPLEX OPERATION

SPI slave supports also standard full-duplex mode. Full-duplex mode is not intended for HCI, as HCI requires flow control (RDY).

Each SPI transaction starts by activating CSN. After setting CSN to '0', RDY goes to '0' to indicate that SPI interface is not ready for transfer. Once RDY is at '1' it indicates that the SPI interface is ready for transaction and header bytes can be sent. The control byte (CTRL) sent on MOSI defines the type of transaction (read or write). The second byte on MOSI is 0x00 and it is there only to read the status data from the Slave. The status byte STS1 on MISO indicates the general status of device, the status byte STS2 on MISO indicates buffer status and it depends on transaction type. In case of write transaction, STS2 contains the number of bytes that can be written into Slave RX buffer. In case of read transaction, STS2 contains the number of bytes that can be read from Slave TX buffer.

After receiving header bytes (STS1 and STS2), the SPI Master knows how many bytes can be transferred via SPI in the current transaction (read or write) and the rest of SPI transaction is standard.

8.8.3. SPI FLOW CONTROL

SPI flow control is available in half-duplex mode used for HCI over SPI. Generally, flow control is there to address the following needs:

- Control transition from Sleep mode to Active mode (external Host must wait until EM9305 indicates readiness for communication)
- Indication of RX buffer not being full = indication that Host can send additional data
- Indication of TX buffer not being empty = indication that Host can receive additional data

8.8.3.1. FLOW CONTROL USING RDY

RDY signal has the following meaning depending on SPI transaction phase:

1. Data ready (when CSN = '1')

- RDY at '1' indicates SPI Slave has some data to send.
- RDY at '0' indicates SPI Slave has no data to send.

2. SPI ready (between CSN falling edge and end of first header byte)

- RDY at '1' indicates that SPI Slave is ready and SPI transaction can start.
- RDY at '0' indicates that SPI Slave is not ready and SPI transaction cannot start. SPI master has to wait until RDY is at '1'.

3. Buffer ready (between end of first header byte and CSN rising edge)

- RDY at '1' indicates that buffer is ready and byte can be written/read
- RDY at '0' indicates that buffer is not ready and byte cannot be written/read. SPI master has to wait until RDY is at '1'.

After asserting CSN and before sending first byte, the SPI Master checks if RDY is at '1'. This check shall be done at least T_RDY (100ns) after asserting CSN. If RDY is at '1', SPI transaction can start. If RDY is at '0', SPI master has to wait until RDY is at '1'.

In this case, of flow control, RDY is used as online information indicating SPI buffer status (buffer ready). SPI Master has to check RDY after each transmitted byte. If RDY is at '1' SPI Master can transmit another byte. If RDY is at'0,' SPI Master may not transmit another byte and SPI Master has to wait until RDY goes to '1'. Once RDY is at '1', SPI Master can continue in transmitting additional byte(s). This approach allows SPI Master to send/receive unlimited number of bytes in single SPI transaction thanks to RDY providing updated status of SPI buffer after each transmitted byte.

Example of write transaction using RDY as a flow control during transaction is shown in Figure 8-21.

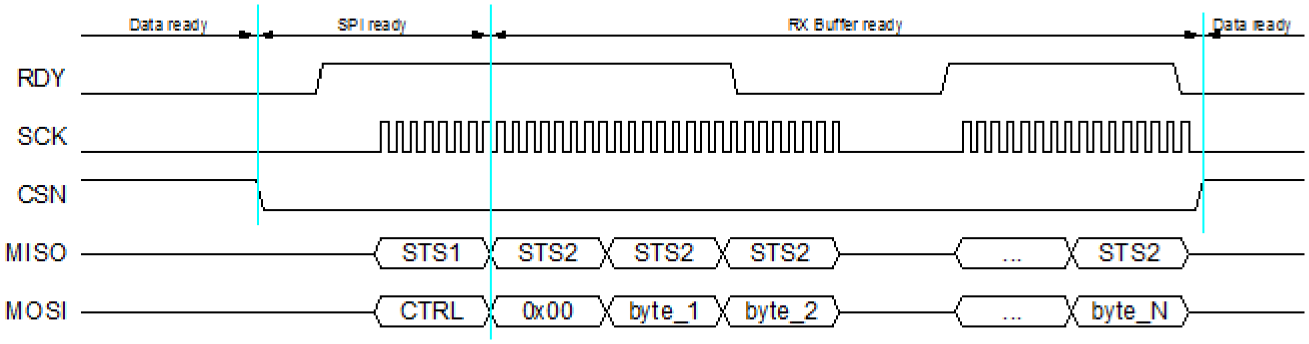


Figure 8-21: SPI Slave write transaction with active flow control using RDY

In this approach, RDY has to be checked after each transmitted byte, which may insert gaps into SPI transaction. Due to that fact, using DMA on SPI Master side might be difficult or even impossible unless DMA can be triggered by RDY pin.

8.8.3.2. FLOW CONTROL USING STATUS BYTE STS2

In this case, SPI Master can ignore RDY during SPI transaction and instead of that use only status byte STS2 to determine maximum length of SPI transaction. Once the maximum length of SPI transaction is known (from STS2), SPI Master can configure DMA to realize the rest of SPI transaction. Using DMA would lead to smooth and fast SPI transaction. After transmitting the given number of bytes (less than or equal to maximum length determined from STS2) SPI transaction is finished by de-asserting CSN. A new SPI transaction starts by asserting CSN and reading status bytes to determine maximum length of this new transaction.

Examples of transaction ignoring RDY are shown in Figure 8-22and in Figure 8-23.

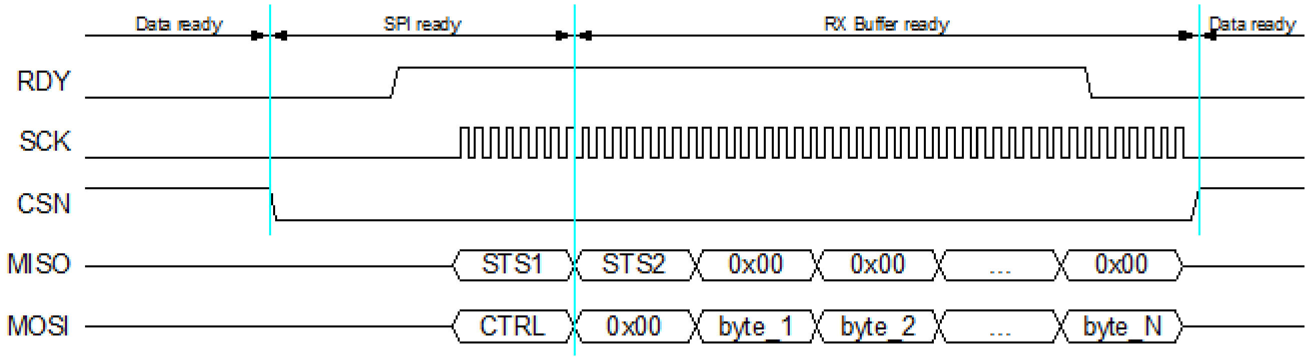


Figure 8-22: SPI control, slave ready for transactions

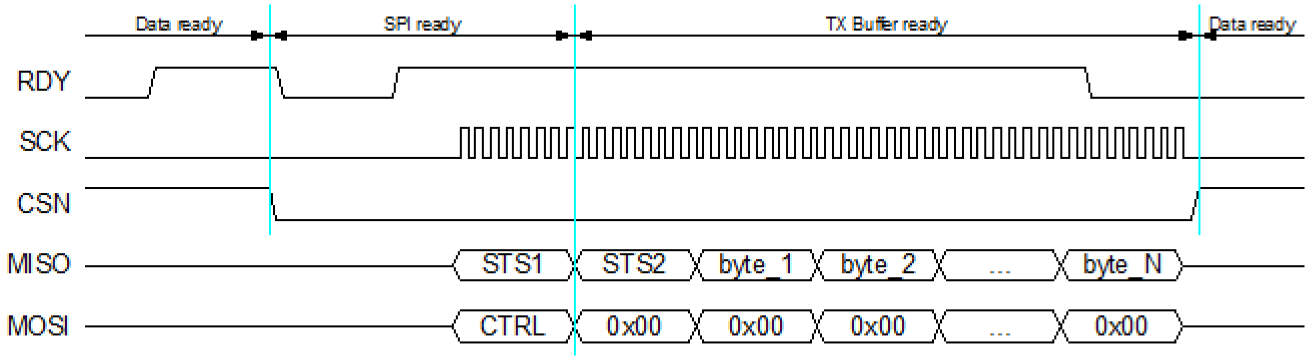


Figure 8-23: SPI control, data ready in RX buffer

8.9. TEMPERATURE INDICATOR

A temperature indicator is embedded into the EM9305 and it is based on the frequency variation of the Low-Frequency RC low-power (LF RC LP) oscillator.

The aim of this sensor is to give an indication of the temperature. This value can be used in order to know if calibration is needed or to have a rough evaluation of the ambient temperature. The real temperature depends on where the EM9305 is placed on the PCB, the actual current consumption and the surrounding environment. With a 2-points calibration, an accuracy of ±5°C can be reached. Depending on the temperature range of the application, the accuracy can be improved. Please contact EM Microelectronic for more information.

8.10. TIMERS

The EM9305 has three types of timers in the system.

The universal timer can be used by the application. The sleep timer is dedicated to the sleep function, which controls when the CPU is woken up. The protocol timer is used by the link layer for higher-speed protocol related timing.

8.10.1. UNIVERSAL TIMER

There are two universal timers (identified by Timer 2 and Timer 3) which can be used by the application. They have the following features:

- 32-bit up counter, selectable auto-reload
- clock source: system clock, GPIO
- 7-bit pre-scaler
- SW start/stop
- HW start/stop
- input capture on HW events (GPIO)
- input capture on SW event
- limit value
- 4 channels output compare value (PWM)
- output to GPIO
- interrupt on limit value, compare and input capture
- synchronization with the protocol timer when the CPU wakes-up

The universal timer configuration may be changed only when the universal timer is disabled.

8.10.2. SLEEP TIMER

The sleep timer is a 32-bit timer driven by the 32kHz crystal oscillator or low-frequency RC oscillator and is dedicated to the sleep function, which controls when the CPU is woken up.

The sleep timer has the following features:

- Dedicated clock (either LF XTAL, or LF RC/12, or LF RC LP 100kHz)
- Automatic Start/Stop in Deep Sleep mode if no LF XTAL is available
- 32-bit timer with additional 16-bit overflow counter. Support of auto-reload, SW start/stop, SW clear
- 8 HW capture events; Protocol timer start/stop; 6 from GPIO
- 4 capture 32-bit values registers with selectable event
- 4 output 32-bit compare values with IRQ and wake event generation

8.10.3. PROTOCOL TIMER

The protocol timer is a 32-bit timer used by the link layer for higher-speed protocol related timing.

The protocol timer has the following features:

- Running at 24MHz clock maximum speed. By default, running at 1MHz using the divider.
- 32-bit timer (32bit@24MHz: 0-178,96sec)
- Optional 8 bits divider
- Start/stop synchronized with sleep timer clock
- Timer value modification during run: value from register is added to the current timer value on request
- Timer preload to specific value (used when switching from Sleep mode to Active mode)
- 8 output compare channels
- 4 input capture channels
- Input capture by SW and HW
- Interrupt generated to CPU whenever interval compare register matches timer value.
- Individual interrupt signals for each interval register
- Software version of event outputs
- Capture events: RX packet sync word detected, RX packet received, TX packet sent, GPIO

There is a tight cooperation between the protocol timer and the sleep timer. The protocol timer is used during RF operation only (when HF XTAL clock is available). The sleep timer is used during sleep period to maintain timing even if main logic including protocol timer is switched off (see Figure 8-24).

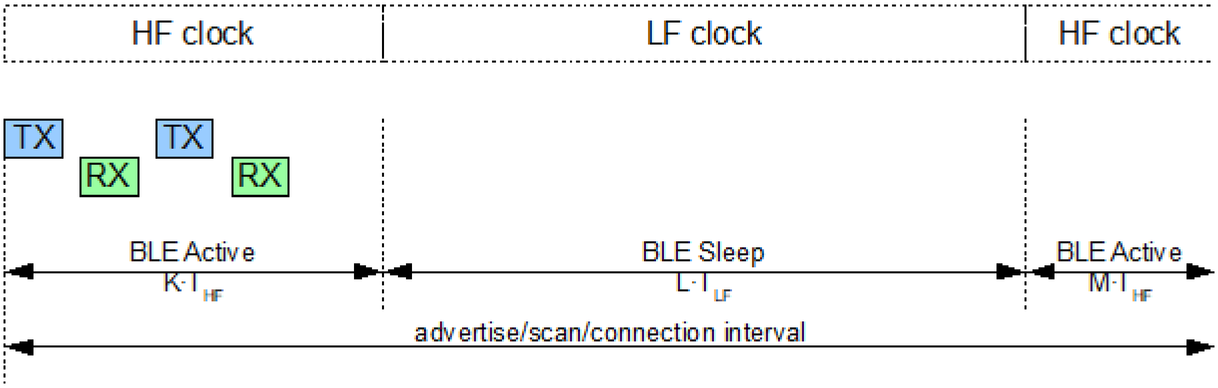


Figure 8-24: Protocol timer and sleep timer cooperation

The following sequence is performed when switching from Bluetooth® LE Active to Bluetooth® LE Sleep:

1. The protocol timer is stopped. This stopping is done synchronously with sleep timer clock.
2. The current protocol timer value (PT_val) and the current sleep timer value (ST_val1) are stored to RAMs.
3. The sleep timer compare value is configured to wake-up the chip after a given time.
4. The main logic is powered down excluding some RAM instances.

The following sequence is performed when switching from Bluetooth® LE Sleep to Bluetooth® LE Active:

1. The sleep timer generates IRQ/event related to the compare value. This IRQ/event is proceeded in PML, which ensures power up of main logic. Main logic is then initialized (reset is performed and content of some blocks is restored).
2. The protocol timer is preloaded to the value PT_val.
3. The protocol timer is started (it continues in counting from PT_val). This starting is done synchronously with sleep timer clock. At the same time sleep timer captures its current value (ST_val2).
4. The length of sleep time is calculated (ST_val2 – ST_val1)*T_{LF}. This value is then recalculated in the protocol timer clock and loaded to protocol timer to compensate the current value. After compensation, the protocol timer will have the value: PT_val + (ST_val2 – ST_val1)*(f_{HF}/f_{LF}).

During Bluetooth® LE Active part, the protocol timer is also used to time particular phases. The compare value is used to time RX window for which RX packet can be received. After receiving RX packet, the compare value is reconfigured by the RF controller to time T_{IFS} after which TX packet is sent.

The timer value recalculation between sleep and protocol timer is done in the CPU. Before recalculation, both frequencies have to be known. In the case of LF and HF XTAL, the frequencies are known. In the case of LF RC oscillator, the frequency has to be first measured – it is described in section 6.6.2.

The protocol timer and the sleep timer are also shared among all active Bluetooth® LE roles. In case of several slave connections, the scheduling algorithm (Link Layer SW) decides which connection/role to be handled first and timers are set accordingly to manage this first request. Once the timing of the given connection is managed, the timers are reconfigured to handle the next connection.

Some timing intervals like Connection Supervision Timeout or Control Procedure Timeout are managed by SW timers.

8.11. UART

The UART block contains a standard buffered UART with RX/TX data FIFO. It supports the following features:

- 2-wire interface without flow control (TX, RX).
- 4-wire interface with flow control (TX, RX, CTS, RTS).
- Full duplex communication.
- 1 start bit, 8 data bits, 1 stop bit.
- Even/odd parity mode or without parity.
- HW generation and checking of parity.
- Configurable speed. Supported speeds: 9600Bd, 14400Bd, 19200Bd, 28800Bd, 38400Bd, 57600Bd, 76800Bd, 115200Bd (default), 230400Bd, 460800Bd, 921600Bd, 1843200Bd.
    - Configurable divider
    - Configurable oversampling factor in range from 4 to 16
- Flow control (CTS, RTS signals) enable/disable.
- Separate TX/RX enable.
- 16 bytes long RX FIFO for reception and 16 bytes long TX FIFO for transmission.
- CPU reads/writes (via AHB Lite bus) 1 byte, or 2 bytes, or 3 bytes, or 4 bytes from/to FIFOs by different register address
- FIFO status is available for CPU:
    - TX FIFO empty.
    - TX FIFO limit (less bytes in TX FIFO than user defined limit).
    - RX FIFO limit (more bytes in RX FIFO than user defined limit).
    - RX FIFO full.
    - Number of bytes in each FIFO.
    - FIFO flush to remove all content. Each FIFO independently.
- IRQs:
- Only three interrupt vectors. One group for TX interrupts, one group for RX interrupts, and one group for error interrupts:
    - TX interrupts:
        - Byte sent
        - TX FIFO empty
        - TX FIFO limit
    - RX interrupts:
        - Byte received
        - RX FIFO full
    - RX FIFO limit
    - Error interrupts:
        - TX FIFO underflow
        - RX FIFO overflow
        - Frame error detected (wrong stop bit)
        - Parity error detected
- Configurable bit order (LSB first or MSB first)

8.11.1. UART 1-WIRE

The EM9305 supports also 1-wire UART. This is done by enabling Open Drain capability on the GPIO and assigning UART RX and TX to the same pad.

8.12. USB

The USB interface is available on QFN or die packages. It consists of the USB controller and the USB PHY (as shown in Figure 8-25). To use the USB transport mode, it is necessary to be in Active XTAL mode, because of XTAL clock precision needed.

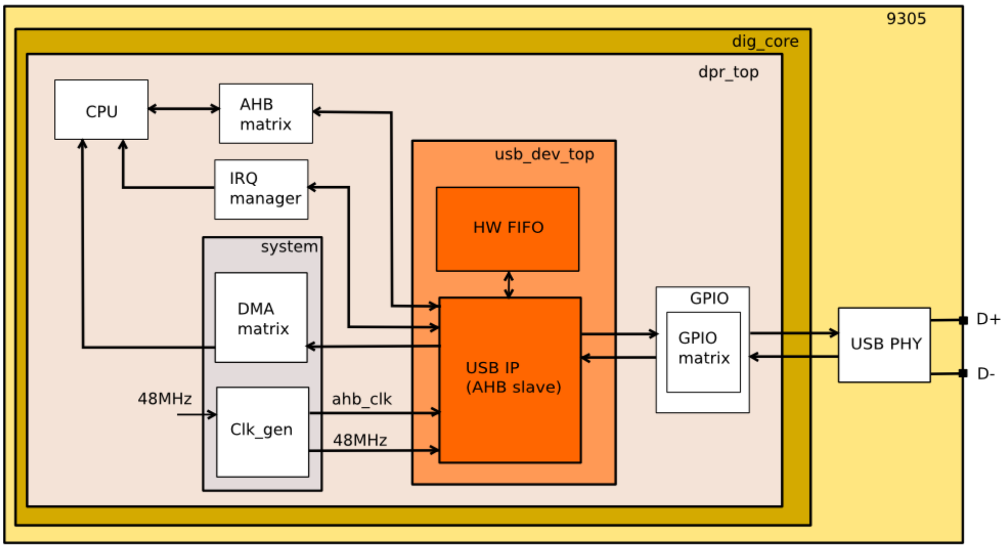


Figure 8-25: USB system level integration

Once the USB transport function is enabled, it activates the clock for the USB controller, and it connects the controller signals to the USB PHY.

As USB is operating at 48MHz, VDD voltage must be set to 1V value before USB transport function is enabled.

No Sleep mode may be activated if the USB transport function is enabled as the USB controller is not supplied in Sleep modes.

The USB transport supports the following configuration:

- Device only, non-OTG, external DMA
- USB 2.0 12Mbps Full Speed mode only
- 3 device Endpoints (1 periodic), 1 control Endpoint, all Device Endpoints are IN/OUT
- Packet counter width 4, queue depth 4
- No power opt features (Hibernation, LPM, Extended Hibernation)

9. SECURITY

The security features are implemented in a combination of digital hardware and software functions. A hardware based true random number generator is implemented which complies with the NIST 800-90A standard. Packet encryption and decryption are implemented in hardware with an AES-128 core embedded in-line with the RF packet processor block, which allows the operation to be performed on the fly. A second AES-128 block is included for non-real-time operation. The ECC P-256 function is implemented in software for key generations. Finally, several features such as the JTAG or some access to the NVM can be locked along the product life cycle.

9.1. CRYPTOGRAPHIC SERVICES

The EM9305 offers several cryptographic services like hardware blocks (AES-128, TRNG) and like software libraries.

9.1.1. AES

The goal of the Advanced Encryption Standard (AES) block is to encrypt (or decrypt) a block message of 128 bits (called plaintext for encryption, resp. ciphertext for decryption) using a private key of 128 bits. The result is an encrypted (or decrypted) block of 128 bits, called ciphertext for encryption (resp. plaintext for decryption). The original message and the private key used are transmitted and processed by the AES core itself using direct parallel inputs, connected to the rest of the logic of the chip. The ciphertext (resp. plaintext) response is disclosed when the processing is finished and released via a parallel output of 128-bit bus.

Two AES-128 blocks are included in EM9305. The first one is only dedicated to the radio, and it enables full AES-CCM authenticated encryption mode support. A second AES-128 encryption/decryption core is available for generic usage.

These blocks are designed according to FIPS-197.

9.1.2. RNG

The EM9305 embeds a Random Number Generator (RNG) compliant with FIPS 140-1, FIPS 140-2 and NIST SP 800-22.

To be compatible with the Bluetooth® Core specifications, a Pseudo-Random Number Generator, also called post-treatment, is mandatory. This post-treatment block needs to be FIPS and NIST proven. The RNG block provides a certain number of bits (seed) to this PRNG block, depending on the entropy of these bits (see Figure 9-1). Typically, the PRNG block needs between 256 to 640 bits from the RNG. Here, the RNG block is used as a source of entropy. This source is used as an input of a pseudo-random number generator function, chosen among all the algorithms detailed in FIPS PUB 140-2.

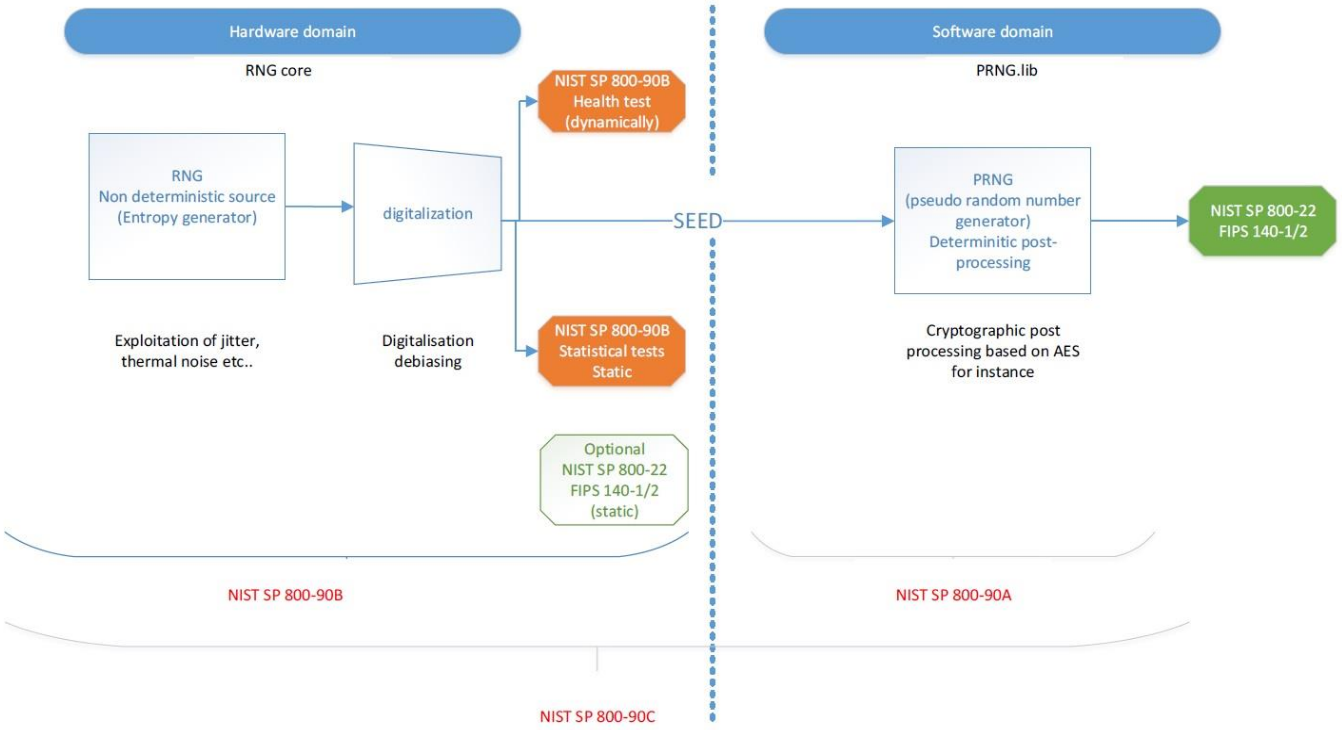

Hardware domain
RNG core
NIST SP 800-90B Health test (dynamically)
RNG Non deterministic source (Entropy generator)
digitalization
SEED
NIST SP 800-90B Statistical tests Static
Exploitation of jitter, thermal noise etc..
Digitalisation debiasing
Optional NIST SP 800-22 FIPS 140-1/2 (static)
Software domain
PRNG.lib
PRNG (pseudo random number generator) Determinitic post-processing
NIST SP 800-22 FIPS 140-1/2
Cryptographic post processing based on AES for instance
NIST SP 800-90B
NIST SP 800-90A
NIST SP 800-90C

Figure 9-1: Overall structure of a RNG and the associated standards

9.1.3. SOFTWARE LIBRARIES

A generic AES (Advanced Encryption Standard (AES) (FIPS PUB 197)) library is available. The three key sizes (128-bit, 192-bit and 256-bit) are supported in both encryption and decryption. Regarding the AES-128 algorithm, either a full software implementation or the hardware block is called.

The following AES libraries are available as part of the SDK:

- AES-CCM (NIST Special Publication 800-38C)
- AES-GCM (NIST Special Publication 800-38D)
- AES-CMAC (NIST Special Publication 800-38B)

The Elliptic-Curve Diffie-Hellman (NIST Special Publication 800-56A Revision 3: ECDH) library based on P-256 curve is part of the Bluetooth® pairing process.

The Elliptic-Curve Digital Signature Algorithm (Digital Signature Standard (DSS) (FIPS 186-4): ECDSA) verification library also based on P-256 curve is mainly used during over-the-air update process.

All standard hash algorithms (Secure Hash Standard (SHS) (FIPS PUB 180-4): SHA-1, SHA-224, SHA-256, SHA-384, SHA-512) are available as software libraries.

The Pseudo-Random Number Generator (PRNG) library is built on both the True Random Number Generator (TRNG) and a post-processing algorithm (National Institute of Standards and Technology (NIST) Special Publication 800-90A Revision 1). That combination is NIST SP 800-22 compliant.

9.2. SECURE KEY CONTAINERS

The secure key containers are located in a dedicated 8kB page of the information area of the NVM. The page is logically divided into 256 keys, where only the first 8 keys can be individually protected against writing. The whole page can also be locked against both erase and write operations. The keys are not readable by the CPU in order to prevent access to any malicious or vulnerable software (see Figure 9-2). It is possible to invalidate any specific key, so that the key is not anymore executable.

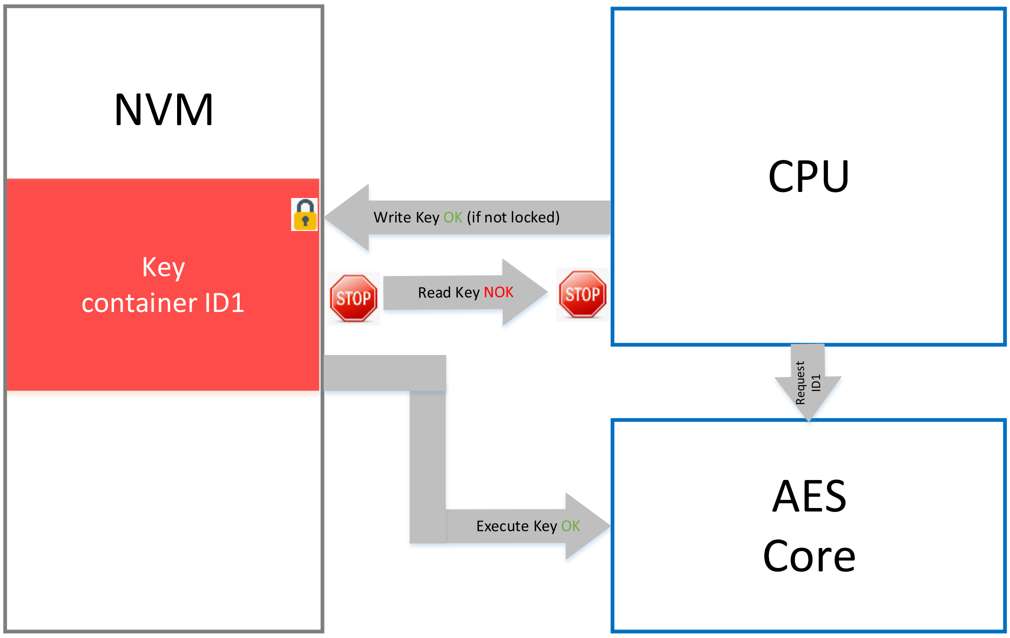


Figure 9-2: Process to write and read in the key container

9.2.1. EIGHT FIRST KEY CONTAINERS

The first eight key containers can be individually locked against writing. This mechanism prevents the key from overwriting, thus prevents it from setting the key to all zeros.

The first eight keys can be written at different stages of the product life cycle. For instance, one can write a set of keys at production and another one at personalization (see Figure 9-3).

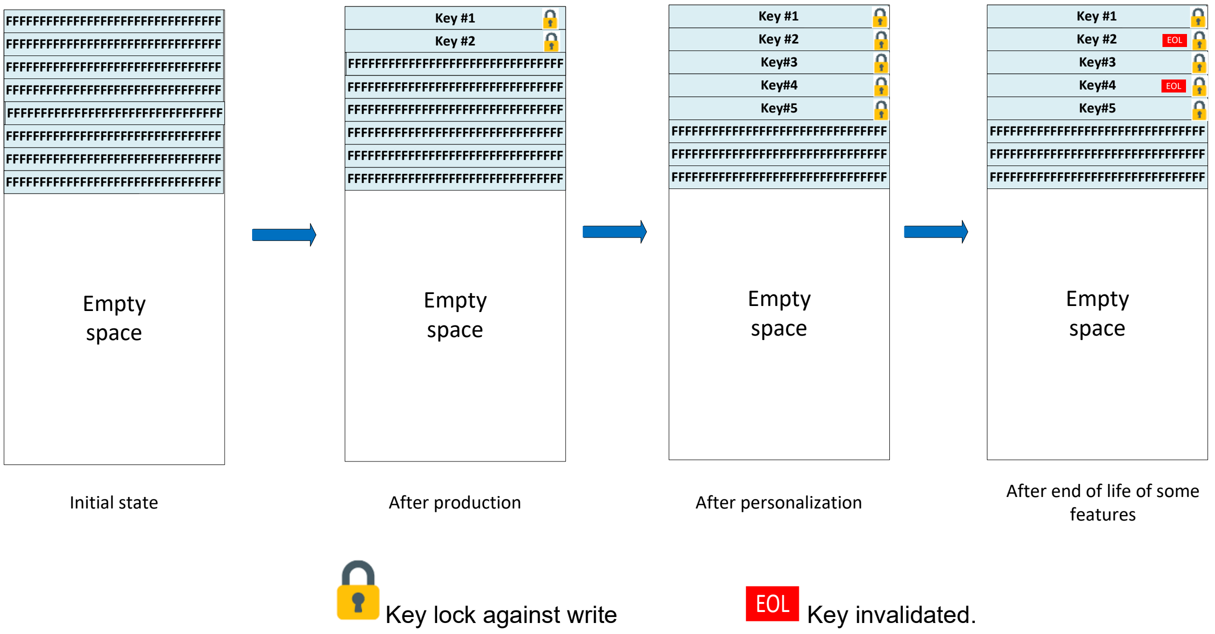


Figure 9-3: 8 first keys life cycle

9.2.2. OTHER KEY CONTAINERS

The other key containers cannot be individually locked against writing. If such a lock is desired, it means that all the keys needed have to be written before locking the full NVM page against writing. This is the only difference between the eight first key containers and the others.

A use case for these key containers would be the storage of session keys.

9.3. AUTHENTICATION SCHEME

The AES-based two-pass authentication (Figure 9-4) is compliant with [ISO/IEC 9798-2:2019(E), 7.2.3 Mechanism UNI.CR – Two-pass authentication]. This scheme is based on symmetric cryptography. It means that a secret key has to be stored in the chip. A key container is dedicated to this key.

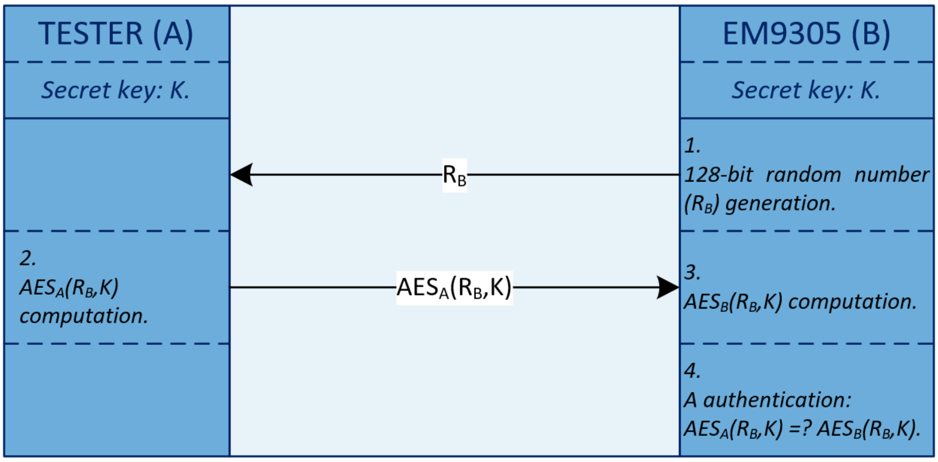

TESTER (A)
Secret key: K.
EM9305 (B)
Secret key: K.
1. 128-bit random number (R_B) generation.
R_B
2. $AES_A(R_B,K)$ computation.
$AES_A(R_B,K)$
3. $AES_B(R_B,K)$ computation.
4. A authentication: $AES_A(R_B,K)$ =? $AES_B(R_B,K)$.

Figure 9-4: AES-based two-pass authentication

10. SOFTWARE ARCHITECTURE

The Figure 10-1 shows the software static architecture made of various software blocks provided in the SDK deliveries. Each block can be seen as an independent piece of software that can be integrated in the final end-user application depending on the needs.

The bottom of the picture shows blocks with black colored background. These blocks describe the software drivers that provide a hardware abstraction layer hiding the hardware complexity and register interface underneath.

Green blocks are middleware software blocks, otherwise called libraries. They provide higher level much complex functionalities on which the end user application can rely to achieve a set of specific features. For example, if the end user application needs to have a timer functionality, then it can directly rely on the timer library that will provide a timer implementation. Therefore, the end user can get rid of any timer hardware related considerations since they are abstracted below this timer library.

Other blocks in grey and orange color background are protocol stacks. They are provided as well, and the end user application can embed them if they are needed. Note that the Bluetooth® LE stack is split into three functional blocks. This is due to the fact that depending on the end user needs, not all three blocks are needed. Only the Bluetooth® LE controller block is mandatory for implementing the Bluetooth® LE protocol in case the device is used as a controller. If more Bluetooth® LE functionalities are needed, then the two other blocks can be embedded as well. If the device is used as a full system on chip, then the Bluetooth® LE host layer is included. And if the device needs to be configured with a specific Bluetooth® LE behavior, then the profiles block is needed.

Finally, the blue background blocks represent the available functional modes. These modes can be seen as a specific way of the device to behave providing the fact that on the field, the application mode is the default one. Other modes are for specific operations like the firmware update over the air that requires a specific application able to receive a new firmware image over the Bluetooth® LE link and to program it into the device's non-volatile memory.

Some blocks are drawn with a purple bold border. This indicates that such software blocks (drivers and/or libraries) are provided in the SDK and in the device's ROM. Such functions located in ROM can be directly called by the end user application from which they can be referred to. Consequently, the end user application does not have any need to embed them.

Some blocks have interdependencies meaning that if one block is embedded or used in the end user application, then one or more other blocks are required as well. For example, the EM System block brings the capability to receive and to interpret HCI commands. For this block to be able to receive its commands, it needs the so-called transport manager which comes along with a transport layer, and either the SPI or the UART driver for the low-level communication.

All these software blocks are provided in the SDK and available for the end user application. Other modes are also accessible to the end user except for the production test mode which is EM only.

More technical documentation is available in the SDK's documentation.

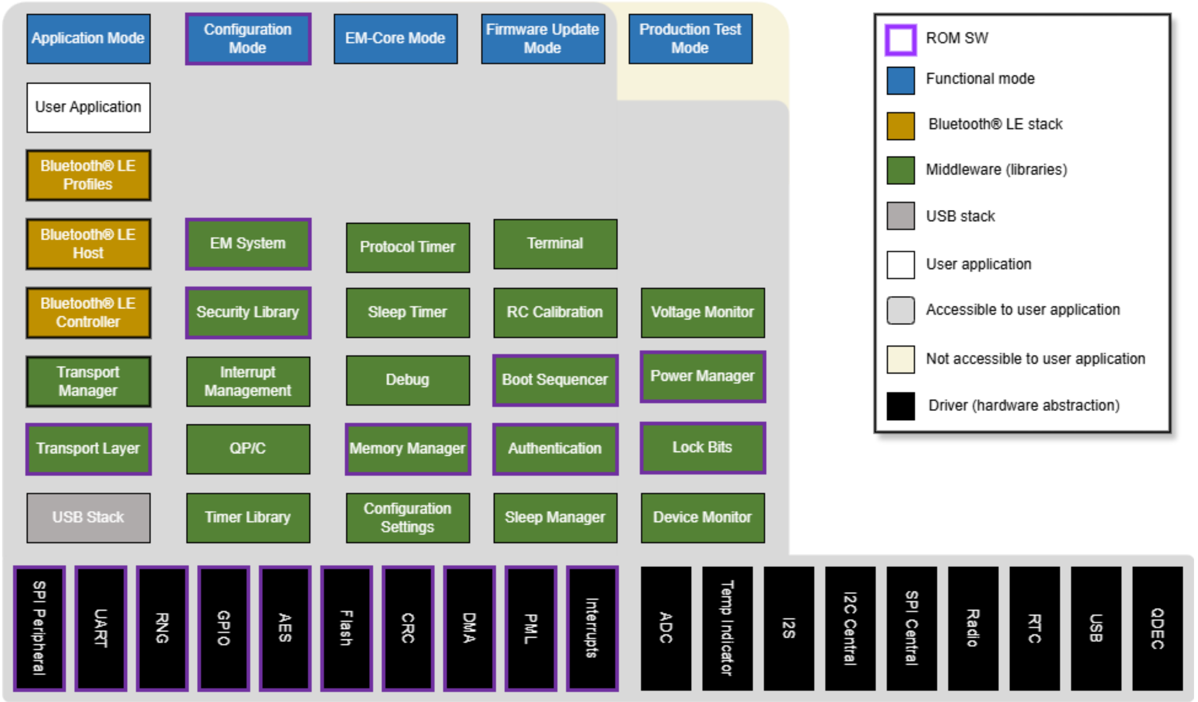


Figure 10-1: Software architecture

10.1. HARDWARE PLATFORM

The core firmware executes on the Synopsys DesignWare ARC EM7D processor (ARC CPU) operating at 48MHz. The ARC CPU supports two types of closely coupled memories:

- Instruction Closely Coupled Memory (ICCM). ICCM is accessible for instruction fetch as well as through data load/store operation.
- Data Closely Coupled Memory (DCCM). The DCCM is accessed by the processor core through data load/store operation only.

Although the ARC CPU architecture has physically separate instruction and data paths, the instruction and data memories, and the peripheral addresses are mapped to the same address space.

The peripheral address space provides memory mapped access to peripherals and peripherals are accessible via dedicated peripheral 32-bit AHB-Lite bus.

The auxiliary address space is a separate address space, where internal ARC CPU peripheries are mapped and accessible using dedicated register load/store instructions.

NVM memory is connected to ARC CPU using dedicated 32-bit AHB-Lite bus and allows instruction fetch as well as data load operation. To speed up executing and fetching instruction from NVM memory ARC CPU is equipped with pre-fetch logic and dedicated instruction cache memory of 2KB.

10.1.1. MEMORY ORGANIZATION

The memory address space is mapped to IROM, IRAM, NVM, DRAM, and the peripheral register address space.

10.1.1.1. INSTRUCTION ADDRESS SPACE

The instruction address space is mapped into the lower half of addressable space. The Instruction Closely Coupled Memory (ICCM) consists of 4 memory blocks: IROM, IRAM1, IRAM2, and IRAM0. The NVM memory is not part of ICCM and it is an instruction memory connected to a dedicated AHB-Lite bus. Furthermore, NVM memory is split into two blocks – Main area and Info area.

The NVM and IRAM memories can be used for data storage. Access to data in IRAM0/1/2 is accomplished using the load and store instructions as is done with the DRAM0/1/2. However, the data access time of IRAM0/1/2 might be longer than the access time of DRAM0/1/2. This is due to IRAM0/1/2 residing on the same physical bus as IROM, so instructions and data access are interleaved decreasing system performance when fetching instruction and data from ICCM.

Access to data in NVM memory is also slower since multiple clock cycles are needed to load data from NVM memory and on the top data access might interfere with instruction fetch further prolonging time needed to load data from NVM memory.

IRAM0 memory is assumed to be used as the JLI table.

Figure 10-2 shows the full RAM address space of the device.

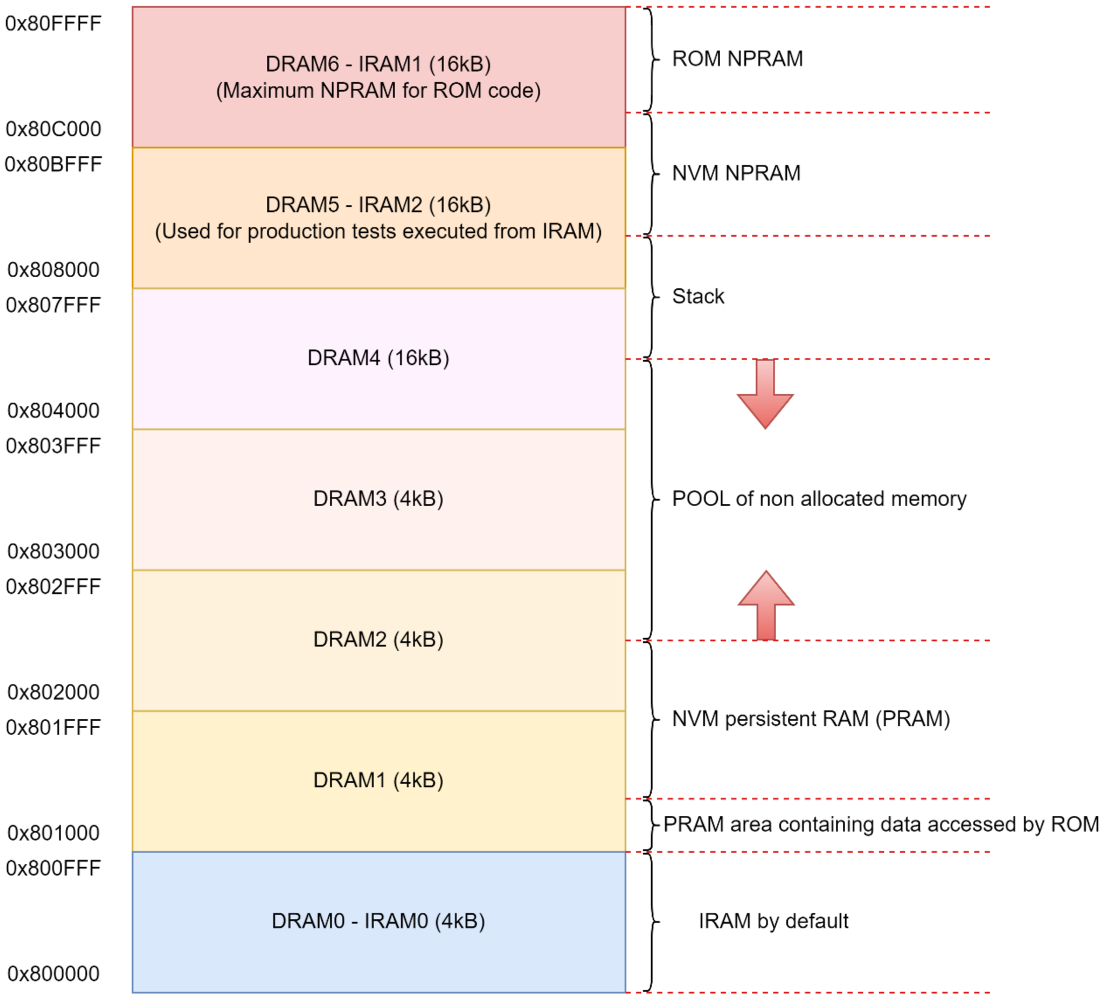


Figure 10-2: RAM address space

Once the reset is released, ARC CPU starts to execute code from address 0x100000, which corresponds to first address in IROM.

10.1.1.2. DATA ADDRESS SPACE

The dedicated data memory address space is mapped to DRAM0, DRAM1, DRAM2, DRAM3, DRAM4, DRAM5, and DRAM6. The DRAMs reside at the beginning of the upper half of the address space. All DRAM memories can be configured as retention or non-retention memory. The memory manager enables persistence for DRAM memories as required to support the requested memory allocation.

It shall be noted that by default, the first RAM block at address 0x800000 is configured as an instruction RAM.

Figure 10-3 shows the data address space of the device.

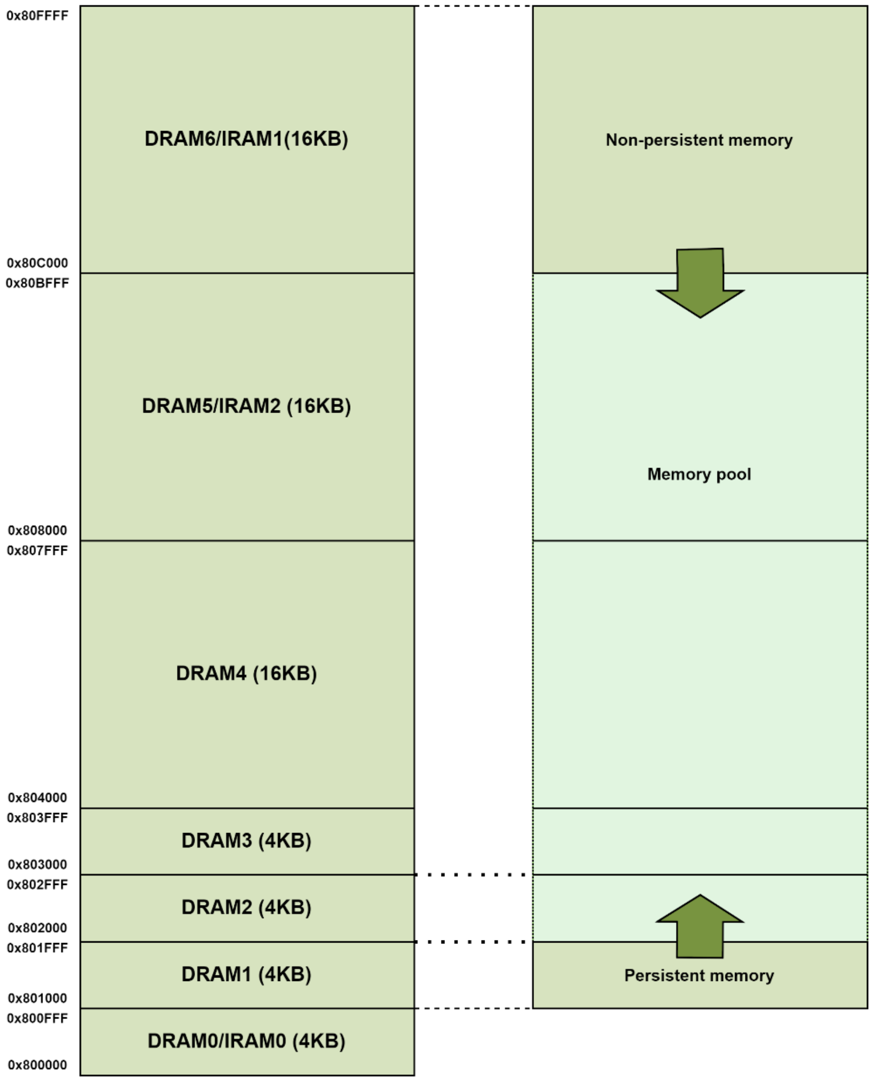


Figure 10-3: Data address space and persistence strategy

10.1.1.3. PERIPHERAL ADDRESS SPACE

The peripheral address space is located at the end of the data address space. The peripheral address space provides access to the memory mapped peripheral registers connected to the AHB-Lite bus.

The peripheral memory map is defined in Table 5-1.

10.1.2. NVM MEMORY

The NVM memory is split into two areas, which are accessible using the same physical memory interface. These two areas (Main area and Info area) are mapped into instruction address space as show in Figure 10-2. The NVM memory characteristics are summarized in Table 10-1.

Table 10-1: NVM memory main characteristics

NVM main area	
NVM Size	512kB
Page Size	8kB
Number of pages	64
NVM information area	
Information Area Size	32kB
Information Page Size	8kB
Number of pages	4
NVM timing	
NVM erase time	8-16 ms to erase the entire NVM
Page erase time	8-16 ms

10.2. MODES OF OPERATION

The EM9305 embedded software or firmware provides an embedded application framework to support Bluetooth® LE connectivity and communication when coupled with a link layer and Bluetooth® LE stack. The core firmware presents a hardware abstraction layer (HAL) to support a fully integrated stack, and an application framework to add optional user applications. The HAL, application framework, and other associated functionality is referred to as the core firmware, which excludes the stack.

The core firmware supports two basic modes of operation as described in the following sections.

10.2.1. CONFIGURATION MODE

The Configuration Mode (CM) is a mode of operation that does not apply configuration options from the NVM memory and does not execute any application code from NVM memory. It provides a reduced function set, which allows configuring device and programming device NVM memory including also functionality to safely invalidate patches in the NVM memory. This is particularly useful to regain access to the EM9305 when an errant user application has eliminated other means of interacting with the EM9305.

10.2.2. APPLICATION MODE

The application mode is a mode of operation where the user application is stored in device NVM memory and executed from NVM memory. Depending on user application (content of NVM memory), there might be various types of application mode.

10.2.2.1. SOC MODE

The user application and the full featured Bluetooth® LE stack host and controller reside in the NVM memory.

10.2.2.2. CONTROLLER MODE

Controller mode is used with an external host where the user application and the Bluetooth® LE host layers of the stack reside in the external processor or host controller while the Bluetooth® LE link layer is stored in the EM9305 NCM memory. Interaction with the EM9305 occurs through HCI.

10.2.2.3. OTA MODE

OTA mode allows the EM9305 to be programmed using OTA with functionality present in the NVM memory. OTA mode might be even part of SoC mode or Controller mode.

10.3. MEMORY USAGE BY FIRMWARE

10.3.1. IROM

IROM contains vector table (reset vector, exception vectors, and interrupt vectors) located at the beginning of IROM. IROM is the first place where ARC CPU starts to execute code (reset vector) hence; IROM contains first stage of boot sequence (ROM boot sequence). SW in IROM also implements the Configuration mode (see Section 10.2.1) which requires the following modules/drivers for its function:

- CPU core boot and initialization
- ROM boot sequence
- Drivers
  - IRQ driver
  - PML driver (lightweight version)
  - NVM driver
  - RNG driver
  - AES/Crypto Unit driver
  - Security libraries
  - Memory utilities
  - SPI Slave driver (non-DMA version)
  - Transport layer
  - EM Transport Manager
  - EM System
- ROM application
  - EM System application

Additional modules/drivers can be put into IROM depending on available space.

10.3.2. NVM INFO AREA

NVM Info area contains four pages. Each page is assigned for different purposes shown in Table 10-2

Table 10-2: NVM Info area organization

NVM INFO PAGE	PURPOSE
Page 3	EM data. This page is used by EM to store: • EM production information (wafer and lot information, device ID, MAC address) • EM trimming data • EM hardware lock bits • EM authentication public keys This page is locked and in read only mode.
Page 2	User data. This page is used to store User related date: • User flags • User trimming data • User hardware lock bits • User authentication public keys
Page 1	Reserved for future use.
Page 0	Key container. CPU does not have access to this page.

Detailed description of NVM Info area pages 3 and 2 are shown in Table 10-3 and Table 10-4 respectively.

Table 10-3: NVM info area page 3 - EM configuration data

ADDRESS HEX	SIZE [B]	DESCRIPTION
0x8000	0	
0x7FC0	64	**Reserved**
0x7F80	64	Test history
0x7F10	112	**Reserved**
0x7F08	8	EM authentication required flag
0x7F00	8	NVM programmed flag (page3 programmed)
0x7E08	248	**Reserved**
0x7DC8	64	EM Authentication Public Key (Diversified Key)
0x7D88	64	EM Patch Signature Public Key (fixed Key)
0x7D84	4	CRC
0x7D80	4	Length
0x7D38	72	**Reserved**
0x7D30	8	RegPmlLockBits
0x7D28	8	RegNvmKcLockKey
0x7D20	8	RegNvmLockMaster
0x7D18	8	RegNvmLockInfo
0x7D10	8	RegNvmLockMain1
0x7D08	8	RegNvmLockMain0
0x7D04	4	CRC
0x7D00	4	Length
0x7CB0	80	**Reserved**
0x7CA8	8	REG_RF_XO_DEBUG_DIG
0x7CA0	8	REG_RF_XO_SEQ_DIG
0x7C98	8	RegNvmTime
0x7C90	8	RegNvmRedunCfg
0x7C88	8	RegPmlTrim
0x7C84	4	CRC
0x7C80	4	Length
0x7C4C	52	Product Information
0x7C48	4	Product Information Structure Version
0x7C44	4	CRC
0x7C40	4	Length
0x7C10	48	**Reserved**
0x7C08	8	MAC address
0x7C04	4	CRC
0x7C00	4	Length
0x7BC4	60	**Reserved**
0x7B88	60	ADC Calibration Data
0x7B84	4	CRC
0x7B80	4	Length
0x6000	7168	**Reserved**

Table 10-4: NVM info area page 2 - User configuration data

ADDRESS HEX	SIZE [B]	DESCRIPTION
0x6000	0	
0x5F10	240	**Reserved**
0x5F08	8	USER authentication required flag
0x5F00	8	NVM read lock
0x5DC8	312	**Reserved**
0x5D88	64	USER Patch Signature Public Key (fixed key)
0x5D84	4	CRC
0x5D80	4	Length
0x5D38	72	**Reserved**
0x5D30	8	RegPmlLockBits
0x5D28	8	RegNvmKcLockKey
0x5D20	8	RegNvmLockMaster
0x5D18	8	RegNvmLockInfo
0x5D10	8	RegNvmLockMain1
0x5D08	8	RegNvmLockMain0
0x5D04	4	CRC
0x5D00	4	Length
0x5C98	104	**Reserved**
0x5C90	8	REG_RF_XO_DEBUG_DIG
0x5C88	8	REG_RF_XO_SEQ_DIG
0x5C84	4	CRC
0x5C80	4	Length
0x5C10	112	**Reserved**
0x5C08	8	MAC address
0x5C04	4	CRC
0x5C00	4	Length
0x5BFC	4	PML Configuration
0x5BF8	4	CRC
0x5BF4	4	Length
0x4000	7156	**Reserved**

10.3.3. NVM MAIN AREA

NVM Main area contains 64 pages where page 0 is dedicated as an entry point for any NVM application. NVM application starts in page 0 containing the 1st part of NVM boot sequence. NVM application continues in code section located in pages 1-63 containing 2nd part of NVM boot sequence and also NVM application itself. NVM Main area also contains a vector table, which is relocated from IROM during boot and dedicated for NVM application. NVM Main area organization is depicted in Figure 10-4.

Depending on NVM application, various modules/drivers are included in NVM code. For example:

- NVM boot sequence
- Drivers (all drivers except those which must run from ROM – e.g. NVM part of PML, security)
  - IRQ manager
  - GPIO driver
  - UART driver (DMA version)
  - SPI Slave driver (DMA version)
  - Transport layer
  - Radio driver
  - Protocol Timer driver
  - Sleep Timer driver
  - RC calibration driver
  - PML driver (full version)
- TSOS/RTOS
  - QPC
- NVM application
  - E.g. Bluetooth® Link Layer of full Bluetooth® stack

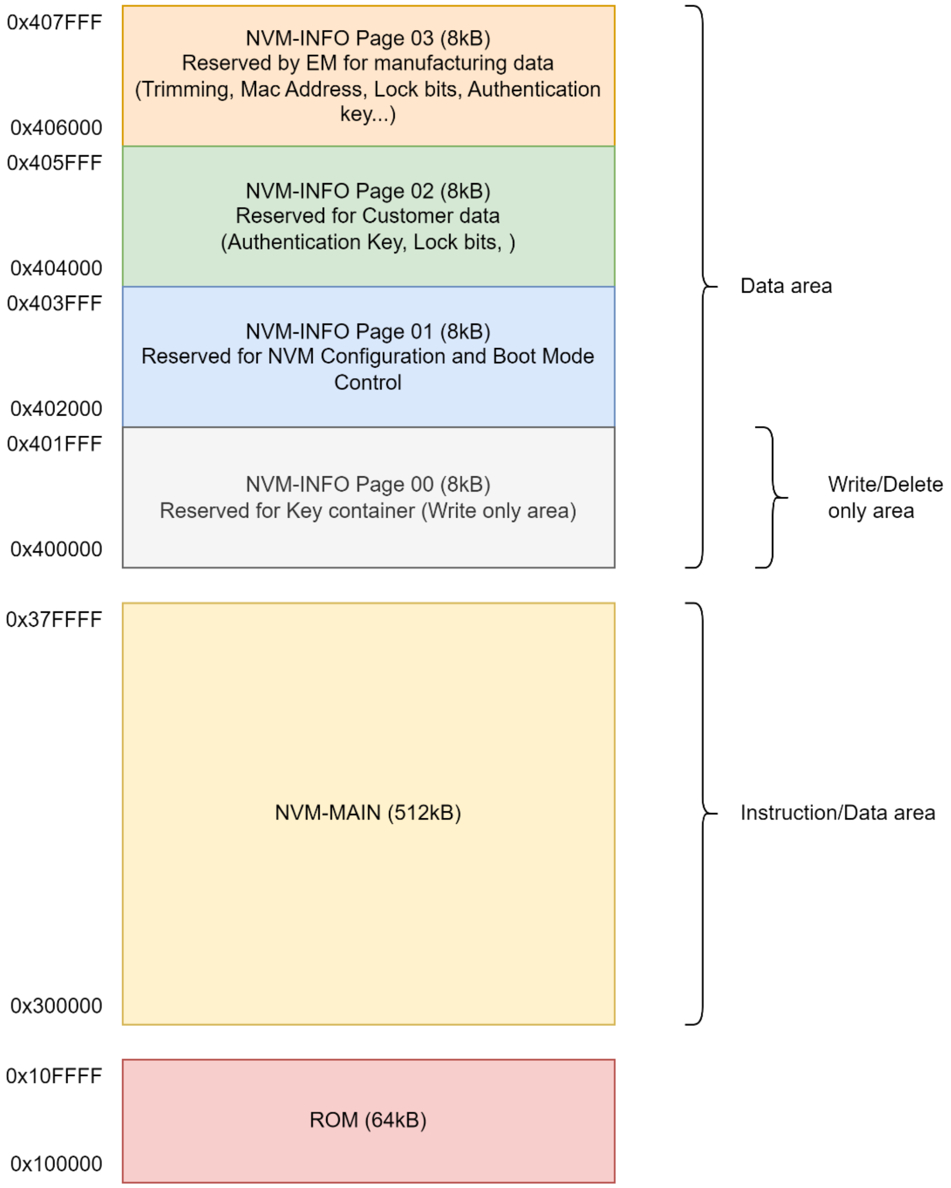


Figure 10-4: Simplified ROM and NVM Main area layout

10.4. BOOT SEQUENCE

Boot sequence consists of various stages stored in ROM and in NVM. Depending on the mode of operation (Configuration mode or Application mode), various stages of boot sequence are executed. Boot sequence consists of following stages:

1. First stage (ROM boot sequence)
2. Second stage (NVM boot sequence)
   a. 1st part of NVM boot sequence (NVM entry point)
   b. 2nd part of NVM boot sequence (Code section in NVM memory)

In Configuration mode, only the first stage boot sequence (ROM boot sequence) is executed. In Application mode, also the second stage boot sequence (NVM boot sequence) is executed.

10.4.1. ROM BOOT SEQUENCE

The security of the boot is ensured primarily by the fact that this first stage boot is in ROM (see Figure 10-6). It ensures no manipulation of this piece of the code. JTAG is disabled during boot.

ROM boot sequence contains these steps:

- CPU core and memory initialization

- NVM power-up sequence and checking if chip is blank
- Loading EM trimming values. If EM trimming data integrity is corrupted, EM trimming data is not applied. Device continues in boot sequence with default EM trimming values stored in HW registers (registers reset values).
- Checking GPIO5 toggling (request to enter Configuration mode). GPIO5 toggling is checked only after POR.
- Configuration mode path
  - Configuration mode initialization by initializing needed drivers (GPIO, SPI slave, transport layer)
- Application mode path
  - Loading EM lock bits. If EM lock bits data integrity is corrupted, EM lock bits are not applied. Device stops executing boot sequence and switches to Halt state (it prevents to boot into application mode without applying EM lock bits).
  - Loading User trimming. If User trimming data integrity is corrupted, User trimming data is not applied and device continues in boot sequence.
  - Loading User lock bits. If User lock bits data integrity is corrupted User lock bits are not applied. Device stops executing boot sequence and switch to Halt state (it prevents to boot into application mode without applying User lock bits).
  - Checking entry point in NVM memory and jumping to NVM memory

Trimming data and lock bits data are applied each time boot sequence is executed – i.e. after any reset and also after resume from Sleep mode.

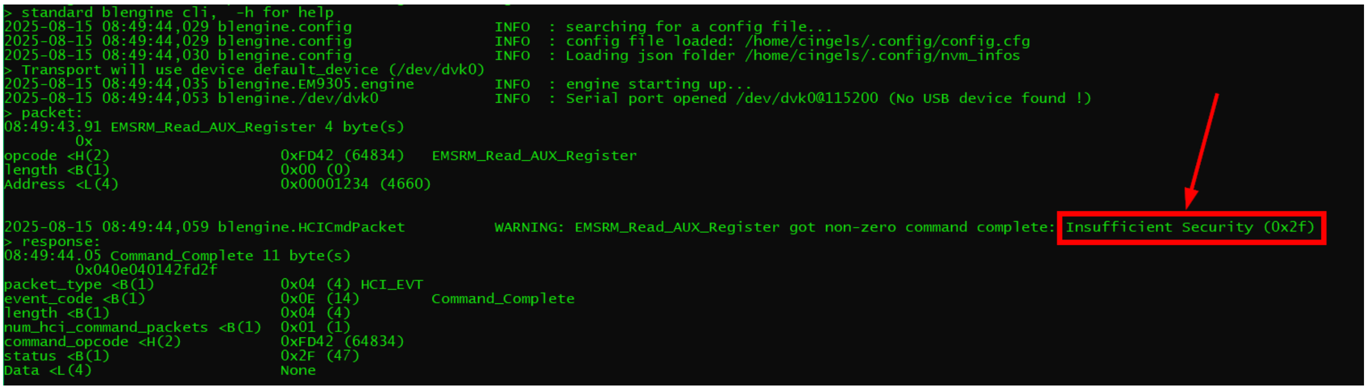
REQUIREMENT

The user should ensure that EM authentication is enabled. To do that, the following sequence is required:

1. Launch the command 'EMSRM_Read_AUX_Register' (0xFD42). This command is used to get the content of a CPU register.

```
$ blengine -port=/dev/COMXX send emsrm_read_aux_register address=0x1234
```

If the EM authentication is active, then this command shall be rejected with the "Insufficient Security" error message as it is shown in the Figure 10-5.

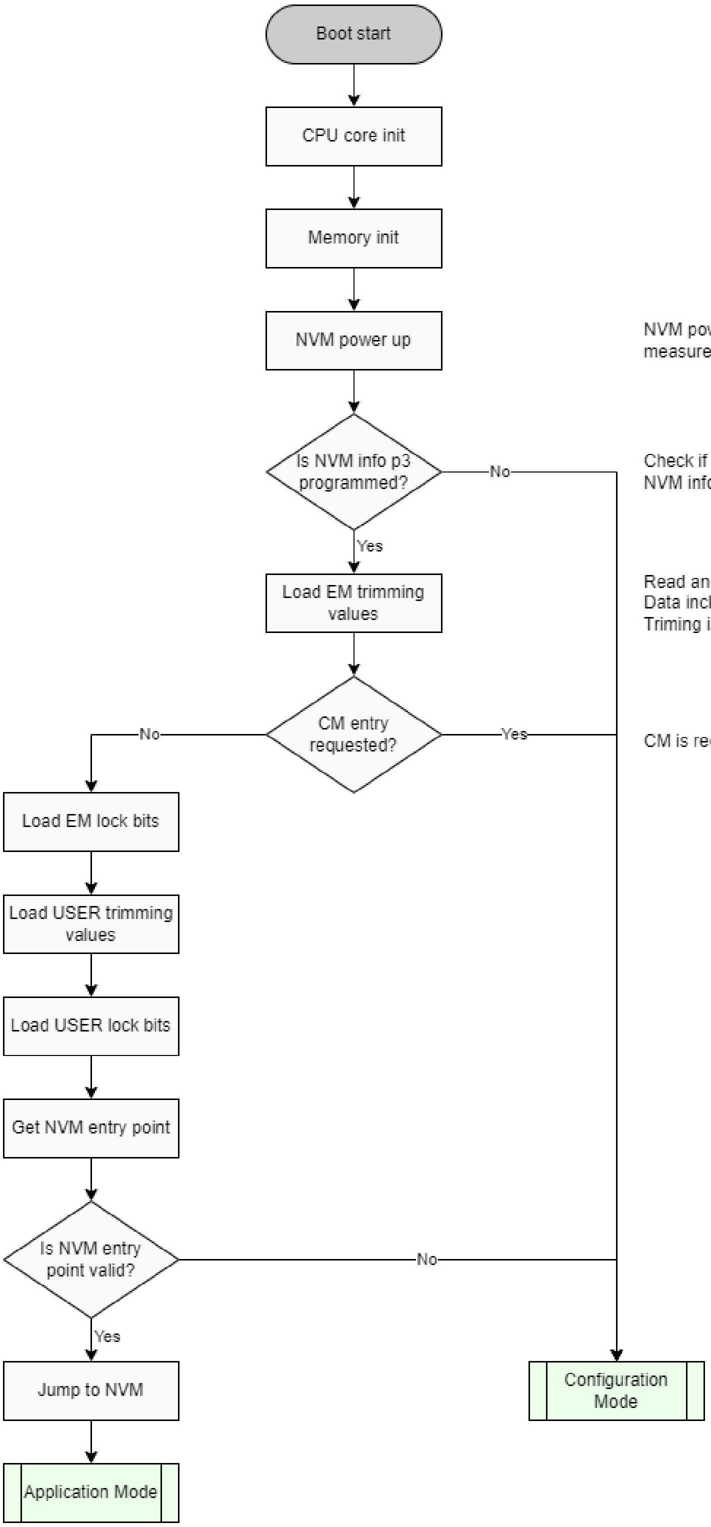


```
> standard blengine cli,  -h for help
2025-08-15 08:49:44,029 blengine.config               INFO  : searching for a config file...
2025-08-15 08:49:44,029 blengine.config               INFO  : config file loaded: /home/cingels/.config/config.cfg
2025-08-15 08:49:44,030 blengine.config               INFO  : Loading json folder /home/cingels/.config/nvm_infos
> Transport will use device default_device (/dev/dvk0)
2025-08-15 08:49:44,035 blengine.EM9305.engine        INFO  : engine starting up...
2025-08-15 08:49:44,053 blengine./dev/dvk0            INFO  : Serial port opened /dev/dvk0@115200 (No USB device found !)
> packet:
08:49:43.91 EMSRM_Read_AUX_Register 4 byte(s)
        0x
opcode <H(2)                   0xFD42 (64834)   EMSRM_Read_AUX_Register
length <B(1)                   0x00 (0)
Address <L(4)                  0x00001234 (4660)


2025-08-15 08:49:44,059 blengine.HCICmdPacket         WARNING: EMSRM_Read_AUX_Register got non-zero command complete: Insufficient Security (0x2f)
> response:
08:49:44.05 Command_Complete 11 byte(s)
        0x040e040142fd2f
packet_type <B(1)              0x04 (4) HCI_EVT
event_code <B(1)               0x0E (14)        Command_Complete
length <B(1)                   0x04 (4)
num_hci_command_packets <B(1)  0x01 (1)
command_opcode <H(2)           0xFD42 (64834)
status <B(1)                   0x2F (47)
Data <L(4)                     None
```

Figure 10-5: Read AUX register command rejected for devices with authentication is active

2. Verify that the returned status is '0x2F – Insufficient Authentication'. This status indicates that EM authentication is enabled.



NVM power up with default parameters no voltage measurement and DCDC config based on voltage.

Check if NVM info page 3 is programmed - check if NVM info page 3 contains programming flag/pattern.

Read and apply EM trimming values stored in NVM info page 3.
Data includes trimming and NVM redundancy page mapping.
Triming is applied here for both - application and CM mode.

CM is requested if GPIO5 is toggling at ~30kHz.

CM should be as simple as possible since patching will not be possible (security reasons, NVM erase operations) or very limited.
By default, it initializes the system authentication level to NONE.

Figure 10-6: ROM boot sequence

10.4.2. NVM BOOT SEQUENCE

Second stage boot sequence (NVM boot sequence) is executed only in Application mode. Main goal of NVM boot sequence is to properly set and configure device for given application after reset or after resume from Sleep mode. Depending if device starts after releasing reset or after resuming from Sleep mode various steps should be executed. Typically, NVM boot sequence should execute these steps:

- IVT relocation from IROM to NVM memory
- Memory retention configuration
- Start after reset path
    - Memory initialization
    - Modules/driver initialization
    - QPC initialization and starting QPC tasks
- Resume from Sleep mode path
    - Clock source switching (HF RC/HF XTAL oscillator)
    - Modules/driver configuration restore
    - QPC resume

10.5. CONFIGURATION MODE (ROM SW)

Configuration Mode (CM) is a mode of operation that does not require any SW in NVM memory, and it resides fully in IROM memory. Main purpose of CM is to provide functionality to program user application into NVM memory and verify/validate NVM memory content after programming.

Once NVM memory programming is done CM can be used to secure chip by applying lock bits to lock device function and/or NVM memory regions against write or erase.

10.5.1. CM ENTERING

CM is entered if GPIO5 is toggling at 30kHz rate during the boot process. The CM request detection is based on edge detection of a signal in the range 25kHz – 50kHz. When CM is entered, all interrupts are disabled except those required for CM operations. The TSOS is not active, but the hardware modules used in CM are initialized.

CM is accessible from external host using SPI transport layer.

Once CM is entered, it is signalized by an EM System event sent to the external host.

10.5.2. CM OPERATION

Once CM is entered, it performs auto-authentication and then it is ready to receive and process EM system commands sent by external host. EM System command processing is done in simple loop where processing of new command is started only when processing of previous command is completed. External host can send only one command at the time and next command can be sent only once host receives event to previous command.

Section 11 describes all EM System commands and events and which of them are available in CM.

CM executes commands in order they are received. If there is no pending command and there are no other pending tasks CM puts device in sleep mode (CPU sleep).

10.5.3. CM SECURITY

To protect sensitive content of the device (mainly NVM memory content), CM provides security features to allow accessing that sensitive content only after successful authentication. There are 3 authentication levels available:

1. NONE (lowest level)
2. USER
3. EM (highest level)

EM System commands not accessing sensitive information can be executed without any authentication (authentication level = NONE) while EM System commands which access sensitive information needs authentication (authentication level = USER or EM). List of available commands and required authentication level is listed in the Section 11.

Upon entering CM auto-authentication is performed which sets authentication level to highest possible one based on flags stored in NVM – flags indicating if EM authentication is required and/or User authentication is required.

If during command processing a higher authentication level is needed, the command processing fails and returns error code to host. Host then can issue command to authenticate to given required level and if successful, host can repeat previous command again.

User authentication uses AES with key stored in Key Container located in NVM Info page 0. Additional information about the authentication scheme can be found in Section 9.3.

10.6. APPLICATION FRAMEWORK LAYER

The application framework provides functional blocks to support the stack and user applications. The application framework consists of the Real-Time Embedded Framework (RTEF), interrupt handler, timer management, and utility functionality such as the firmware patch manager, test mode manager, and standard C runtime libraries.

10.6.1. REAL-TIME EMBEDDED FRAMEWORK

The Real-Time Embedded Framework (RTEF) is the task management software framework QP/C ported to the ARC processor with minor customizations and optimizations. RTEF is a run-to-completion, pre-emptive, cooperative operating system that is optimized for speed with a minimal ROM footprint in NVM. It contains a pre-emption kernel with up to 16 levels of priority. Each priority must run a single task that may consist of 1 or more finite state machines (FSM). These tasks must be defined before RTEF is started.

Pre-emption

RTEF uses two forms of pre-emption: synchronous and asynchronous. Pre-emption occurs when an event is posted from an ISR or within the currently running task.

Synchronous Pre-emption

Synchronous pre-emption occurs when a task involves an event that is handled by a higher priority task. In this case, the event will be added to the event queue and the task scheduler is immediately called to run the higher priority task. Once all higher priority tasks are complete, execution resumes in the original task.

Asynchronous Pre-emption

Asynchronous pre-emption occurs when the interrupt handler issues an event handled by a higher priority task. In this case, the event will be added to the event queue and the task scheduler will run after all interrupts are handled that are higher priority than the tasks. For this, a context switch is required to save the state of the previous task, and another context switch is required to restore the previous task after all higher priority tasks are complete.

Context Switching

Context switching is achieved with software interrupts. These correspond to the priority levels for the RTEF. When asynchronous pre-emption occurs (even in the case where the CPU is sleeping and no tasks are in the run queue), the software interrupt for the corresponding priority of the event that will be received by a task is triggered. Remember that asynchronous pre-emption occurs when a hardware ISR issues an event. Each software interrupt is given a unique priority that has no overlap with any other interrupt. All software interrupts are lower priority than all other interrupts so that execution may be paused while a hardware event is processed.

Note that when a synchronous pre-emption occurs, the corresponding software interrupt for the new priority is not triggered. The RTEF will simply note the new priority and prevent any hardware ISR from triggering the higher priority (but even higher priorities may still be triggered asynchronously) until the priority drops again. In this case, the event will be processed (again in the lower priority software interrupt) after the task currently running at that priority is completed. This produces the minimal amount of software interrupts and context switches.

The other method of handling context switching is to trigger a single software interrupt (again at the lowest priority) that will execute the scheduler and run the new task upon exit. This is achieved by modifying the stack pointer and return address. Once the task is complete, a second interrupt is triggered to switch back to the previous task. At this time, the context restoring interrupt will remove itself from the stack so that when the interrupt returns, the task returned to is the previous pre-empted task. Unfortunately, this requires time to be consumed saving the state of the CPU for the context restoring interrupt, only for it to be popped off the stack immediately upon entry.

Since the RTEF only contains a maximum of 8 priority levels and the two interrupt (ARM) approach requires more time to context switch, 8 software interrupts were used instead. This produces less context switching code and allows for a simpler and assembly free solution. An added benefit of the assembly free solution is that any architecture changes (like another register that must be saved) on the ARC will not require a change of the context switching code.

10.6.2. INTERRUPT HANDLER

The interrupt handler provides a default interrupt handler for all interrupt vectors. The default interrupt handler for a specific interrupt vector is overwritten to provide specific functionality to handle to the specific interrupt.

10.6.3. MEMORY MANAGER

The memory manager provides functionality to dynamically allocate memory from a memory pool composed of the optionally retention DRAM memories (see Figure 10-3). The persistence option is only enabled once a retention memory allocation is made in that memory. Once the persistence option is enabled for a DRAM, it is not disabled during operation. Functionality is provided to allocate either retention or non-retention memory with or without 32-bit alignment enforced. To eliminate the complexity and non-determinism of garbage collection, de-allocating or freeing memory is not supported.

During memory manager initialization, it determines how much persistent memory is needed (from build process) and it sets corresponding DRAM blocks into retention mode to fulfil initial memory requirements. Additional memory is allocated dynamically as needed in application.

10.6.4. NVM MEMORY DRIVER AND MANAGEMENT

The EM9305 embedded software supports the following items and features pertaining to the EM9305 Flash.

- Software interface to the underlying NVM memory by means of an NVM software driver.
- Reading, writing, page erasing, information area erase, and full Flash erase.
- User data storage.
- EM data and manufacturing data storage.
- Lock bits.

The Flash memory layout depends on the memory configuration. The minimalistic NVM layout content is represented by the Figure 10-7. It contains the following software:

- The NVM bootloader responsible of starting up any other firmware including the end user application
- The end user application

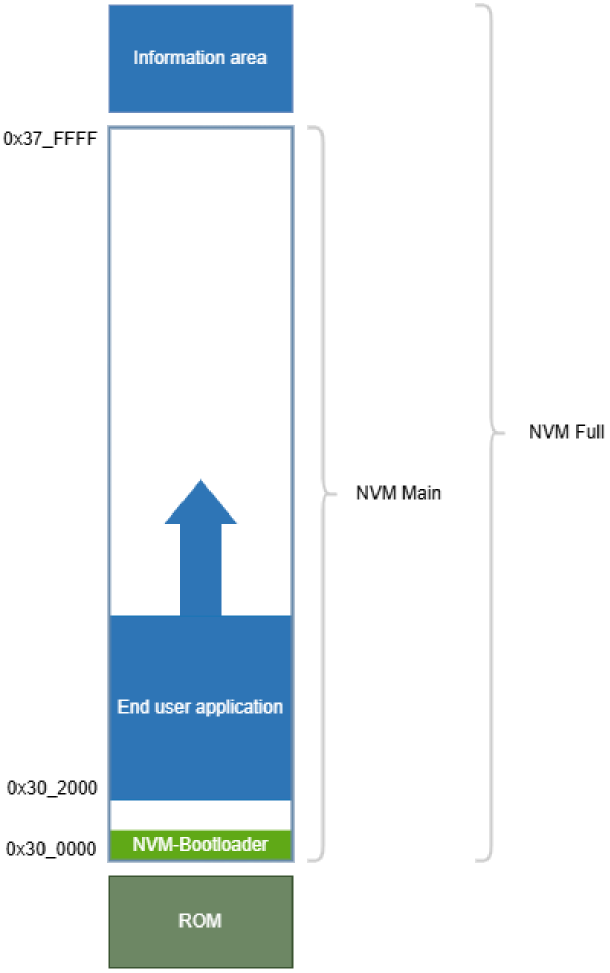


Figure 10-7: Minimalistic NVM layout content

As shown on Figure 10-7, the size of the end user can grow toward the top of the memory. It embeds all the drivers, libraries and protocol stacks (including the Bluetooth® LE stack) it needs for its own proper behavior.

The Figure 10-8 shows the content of the NVM memory in case a firmware update over the air is needed. A dedicated piece of software available in the SDK is programmed at the end of the NVM, on a flash block address boundary. The usually reserved address is `0x360000`. The execution of this firmware is triggered by the NVM Bootloader based on configuration flags that are controlled by the end user application.

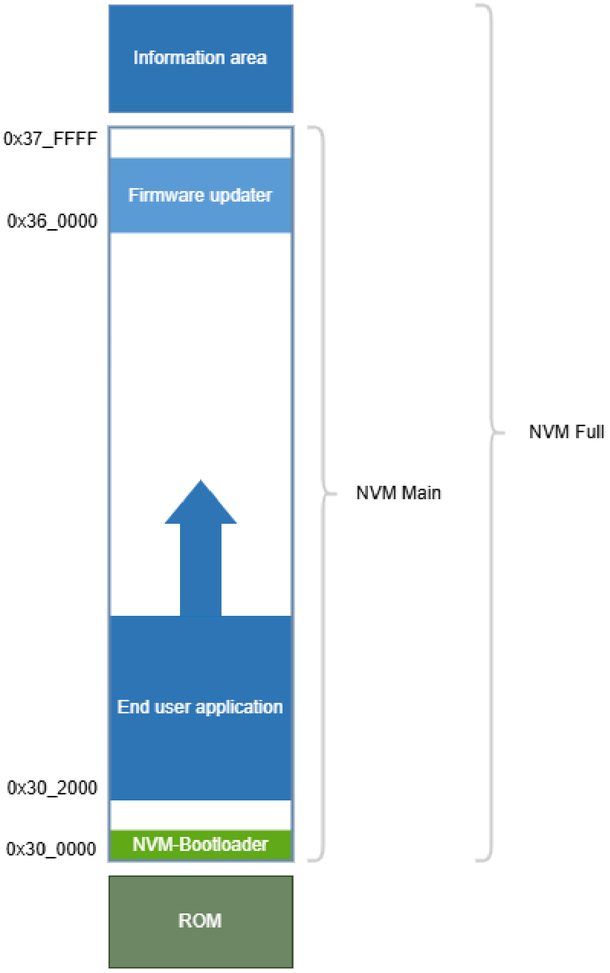


Figure 10-8: NVM layout for firmware update over the air

The Figure 10-9 shows another NVM layout content when the end user application relies on the presence of the EM-Core firmware. In this case, the EM-Core starts at the beginning address of the first free block after the NVM Bootloader (usually `0x302000`). The end user application starts then at the beginning address of the first available block after the end of EM-Core.

Still based on configuration flags that can be driven by the end user application, the NVM Bootloader can be instructed to start either EM-Core, or the end user application or the firmware updater if it is part of the memory.

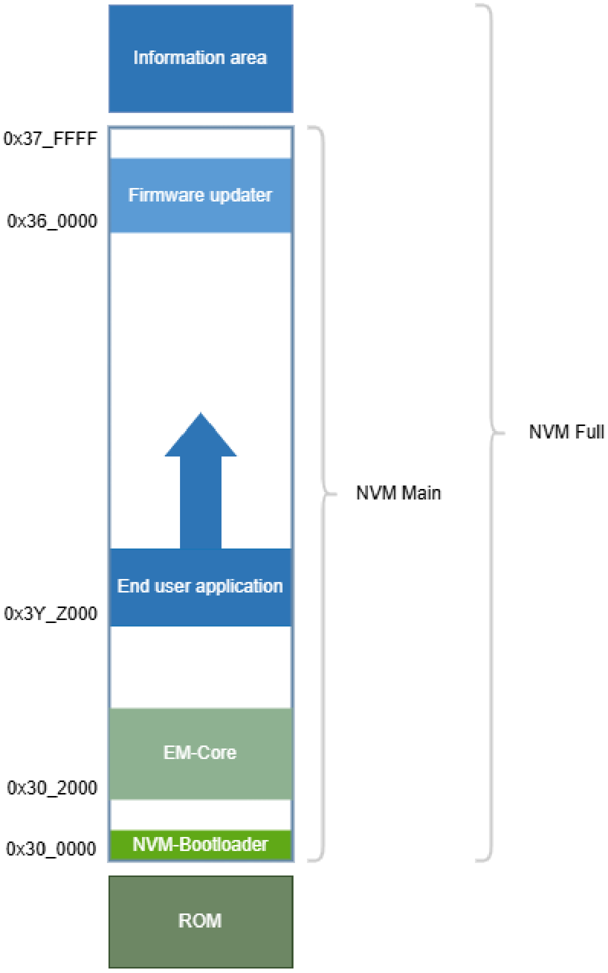


Figure 10-9: NVM layout content with EM-Core

10.6.5. EM SYSTEM

EM System (EMS) layer is module that is used for handling and processing EM System commands. The EM System layer has its own transport manager layer and defines a set of EM System commands and events (Vendor Specific HCI Bluetooth® LE commands and events) as described in Section 11. These EM System commands and events are fully compliant with the packets format defined in the Bluetooth® Core Specification. Figure 10-10 shows a simplified view of the overall system architecture using EM System layer.

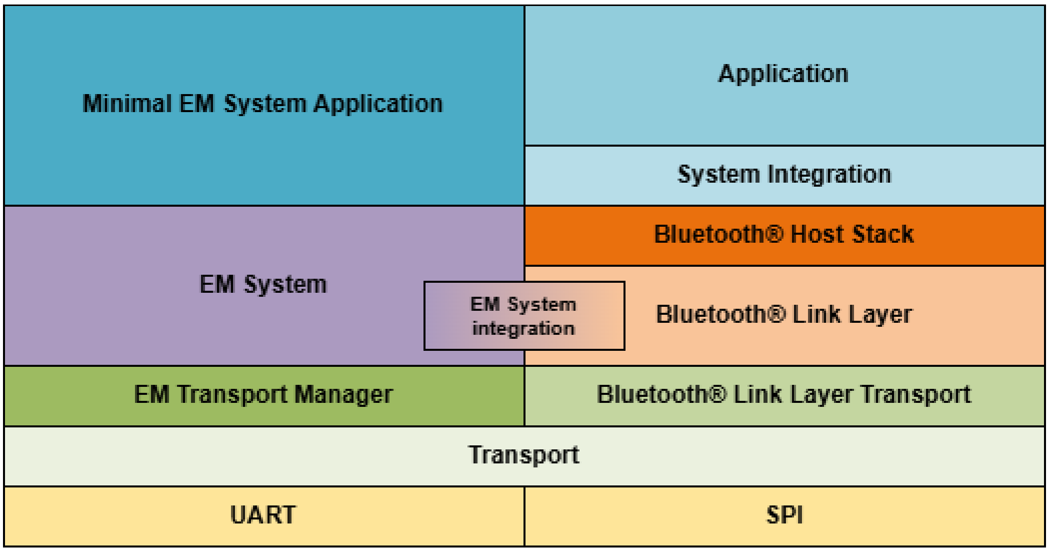


Figure 10-10: Architecture using EM System layer

Simplified description of all the layers:

- **UART**: Low level UART driver.
- **SPI**: Low level SPI driver.
- **Transport**: Abstraction layer for UART or SPI physical transports.
- **EM Transport Manager**: Simple transport manager containing one RX buffer, one TX buffer, and a generic HCI commands parser.
- **EM System**: Contains all the EM System commands parsers.
- **Minimal EM System Application**: Minimal simple application that manages the system without the need of a RTOS.
- **Bluetooth® Link Layer Transport**: Full transport manager containing a generic Bluetooth® packets parser and one or more RX/TX packets buffers.
- **Bluetooth® Link Layer**: Bluetooth® Link Layer according to Bluetooth® Core Specification.
- **EM System integration**: Allows the EM System commands received through the Link Layer to be parsed by the EM System commands parsers.
- **Bluetooth® Host Stack**: Bluetooth® Host Stack according to Bluetooth® Core Specification.
- **System integration**: Bluetooth® Link Layer and Host Stack system integration layer.
- **Application**: SoC Bluetooth® application.

10.6.5.1. CONFIGURATION MODE (MINIMAL EM SYSTEM CONFIGURATION)

The minimal configuration is the one used in Configuration Mode (Figure 10-11). In this configuration, only a limited set of EM System commands and events are available – details are in Section 11. No Bluetooth® features are available.

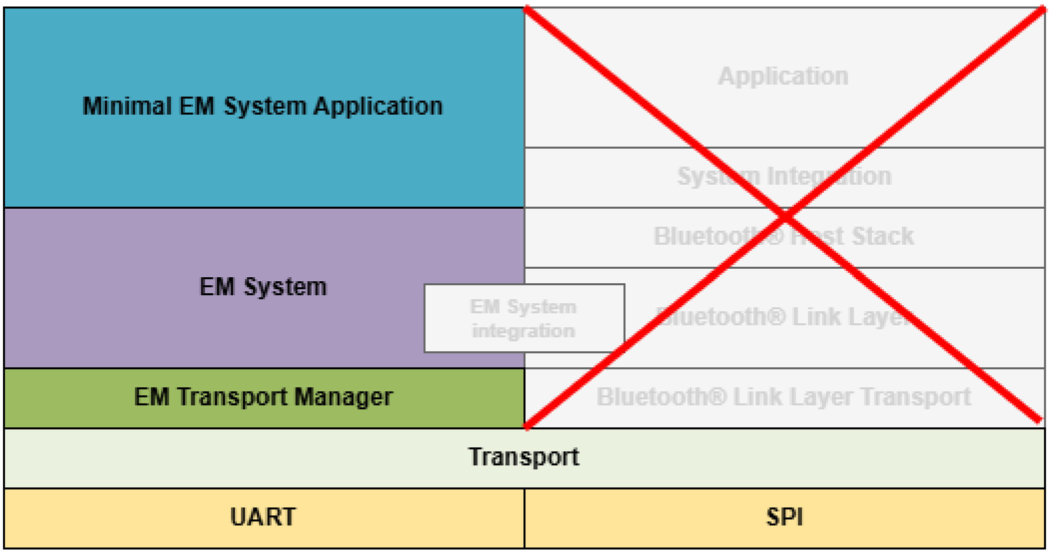


Figure 10-11: Architecture using EM System layer in Configuration Mode.

10.6.5.2. APPLICATION MODE (BLUETOOTH® SYSTEM CONFIGURATION)

In Application mode (with Bluetooth® Link Layer SW in NVM), the EM Transport Manager and the Minimal EM System Application are not used. Those layers are replaced by Bluetooth® Link Layer Transport layer, Bluetooth® Link Layer and optionally by other upper layers (Figure 10-12). The EM System commands are still available through the EM System integration layer.

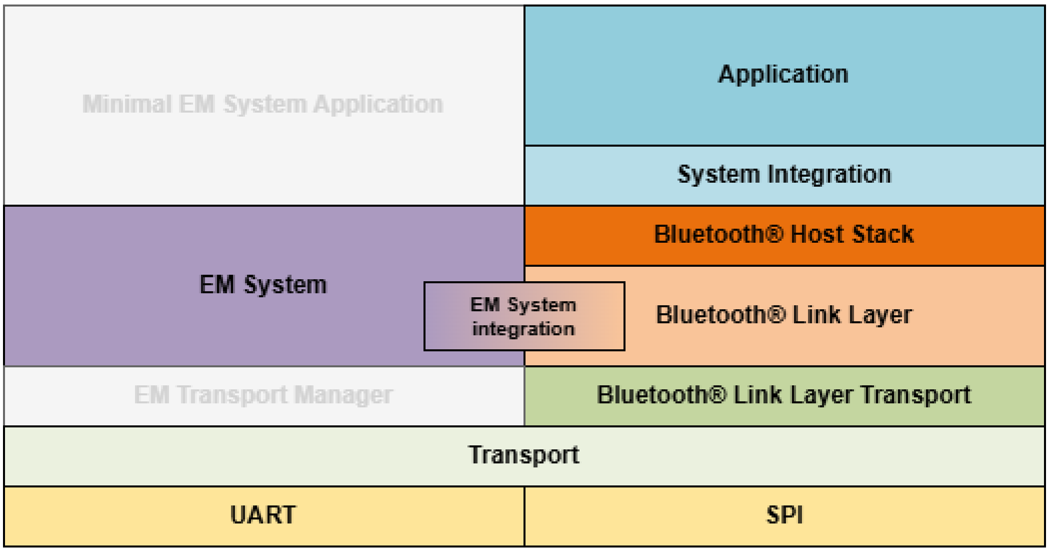


Figure 10-12: Architecture using EM System layer in Application Mode (with Bluetooth® Link Layer SW).

10.6.5.3. TRANSPORT PROTOCOL LAYER

There are two different transport layers in the system:

- **EM Transport Manager**: It is only used in the minimal configuration, and it supports only the EM System commands. The legacy Bluetooth® HCI commands are not supported. Moreover, all other legacy Bluetooth® packet types (ACL, Synchronous Data, and Isochronous Data) are invalid when this transport is active.
- **Bluetooth® Link Layer Transport**: This transport is more complete and is compliant with all the Bluetooth® packet types described in the Bluetooth® Core Specification.

Both of these transports use the legacy Bluetooth® packets format and a Command-Response-Event mechanism as described in the Bluetooth® Core Specification. When a Command is issued by the Host, a Response is transmitted by the Target device. When the Target needs to send information to the Host in an autonomous manner, an Event may be used.

10.6.6. EM TRANSPORT MANAGER

The EM Transport Manager Layer can work without the need of a RTOS/TSOS. This layer has one RX buffer to be able to receive Command Packets and to forward them to the EM System Layer. It also has one TX buffer and provides some APIs to send Event Packets.

10.6.6.1. TASKS PROCESSING

The packets processing is not executed directly in the transport interrupt handlers. Instead of that, a function is called from the EM Transport Manager Layer to notify the scheduler (part of the Minimal EM System Application) that there are some tasks to process. The scheduler then calls the process function of the EM Transport Manager Layer to execute the internal pending tasks outside of the ISR context.

10.6.6.2. SYNCHRONIZATION LOST

A loss of transport synchronization is detected when an invalid packet identifier is received. It means either the length of the previous command was wrong or there is a transport issue.

After a loss of synchronization, the EM Transport Manager layer re-synchronizes on the next “HCI Reset” command sent by the Host. All other received bytes are ignored. This procedure is detailed in the Bluetooth® Core Specification.

10.7. HARDWARE ABSTRACTION LAYER

The HAL provides hardware abstraction of the EM9305 hardware platform in the form of a boot sequencer, power manager, and hardware drivers.

10.7.1. BOOT SEQUENCER

The boot sequencer is the functional block that executes after the following conditions:

- Transition from powered off to powered on (boot state) – the value of all memory is undefined.
- Reset (reset state) – the value of all memory is retained from prior to the reset
- Exit from sleep mode (resume state) – retention memory is retained from prior to sleep, but all non-retention memory is unknown.

10.7.2. SLEEP MANAGER

The sleep manager primarily determines if a sleep mode can be entered. The firmware identifies three sleep modes, CPU sleep, sleep, and deep sleep. In CPU sleep mode, the CPU is placed in sleep mode with the other system components power on. In sleep mode, the CPU is placed in sleep mode, and many of the system components are powered off. Deep sleep mode is executed by the core firmware unless specifically commanded through a vendor specific HCI command.

The following sequence of events are executed prior to entering sleep mode.

1. TSOS enters the idle state.
2. TSOS calls the power manager’s sleep function.
3. The sleep function determines if all conditions have been met to enter sleep mode as defined below.
4. The store functions of all driver modules are called to store the configuration data.
5. Retrieve and store the next protocol timer transaction from the linker layer to determine if the radio should be powered on when resuming from sleep.
6. The sleep function executes the sleep sequence as defined in the Figure 6-2 of this datasheet.

All of the following requirements must be met prior to entering sleep mode:

- TSOS has entered the idle state
- No UART transactions are active or pending
- No SPI transactions are active or pending
- No USB transactions are active or pending
- Maximum sleep time reported by the link layer (link layer API) is greater than the minimum allowed sleep time plus overhead. The minimum allowed sleep time is defined as 3ms plus overhead and derived by the sum of the following time periods:
    - Time to store the configuration
    - Time to restore the configuration
    - Time to start HF XTAL
    - Additional margin

When in sleep mode, the JTAG function of the device is powered off, which will result in the debugger disconnecting from the device. The embedded software provides a configuration bit that forces the embedded software to substitute CPU sleep for sleep. Since CPU sleep does not power off the JTAG function, the debugger will remain connected for software debugging.

10.7.3. VOLTAGE MONITOR

The voltage monitor is responsible for providing power level status to the core firmware and link layer to limit the use of or disable the power-level sensitive hardware components. The voltage monitor is executed after resuming from sleep mode and when commanded by the voltage monitor IRQ. It is also executed when starting the RF power domain, although only if the previous result is older than 0.5ms. An IRQ from voltage monitor is triggered whenever the voltage drops below a defined limit. The voltage monitor provides functions to identify the following conditions:

- Voltage level sufficient to support NVM operations
- Voltage level sufficient to support Bluetooth® LE radio operations

If the supply voltage drops below a specific level, the voltage monitor issues a system notification.

10.7.4. DEVICE DRIVERS

The device drivers provide a layer of abstraction of the underlying hardware generally through memory-mapped registers. All interaction with a hardware component is accomplished through the device driver. Table 10-5 lists all required hardware drivers and the hardware component the driver manages.

Table 10-5: Device drivers list

FIRMWARE DRIVER	DESCRIPTION
AES	Advanced Encryption Standard (AES-128) for Bluetooth® encryption/decryption, authentication and user purposes
DMA	DMA driver to allow faster and/or automated transfer between memory and periphery (e.g. radio FIFO to memory).
GPIO	General Purpose IO driver to control and read GPIO state and GPIO configuration
I2C Master	I2C master interface controller
I2S	I2S interface controller
Interrupt Handler	Interrupt controller
NVM	Read / Write handling of non-volatile memory (NVM)
Power Control	Power Management Logic and control
Protocol Timer	Protocol timer driver used for precise protocol timing (e.g. Bluetooth® protocol)
Radio	Radio driver allowing to configure radio and send/receive packets.
RC Calibration	Calibration system for the RC calibration system
RNG	Random Number Generator used for Bluetooth® authentication and encryption
Sleep Timer	Sleep timer managing timing in Sleep and Deep Sleep mode
SPI Master	SPI master interface controller
SPI Slave	SPI slave interface
UART	UART interface controller
Universal Timer	General purpose timer active in the active power mode
USB	USB 2.0 driver handling up to 3 endpoints
Voltage Monitor	Voltage monitoring

11. EM SYSTEM COMMANDS AND EVENTS

This section includes EM System Commands and Events (vendor specific command and events) and also vendor specific error code. The commands that are not included in the ROM are available only through the EM-Core.

11.1. PACKET DESCRIPTION

11.1.1. PACKET INDICATORS

A packet is always starting with a packet indicator to determine which kind of packet it is. The Table 11-1 shows the different kinds of packets and the associated packet indicators.

Table 11-1: HCI Packet Indicators

HCI PACKET TYPE	HCI PACKET INDICATOR
HCI Command packet	0x01
HCI ACL Data packet	0x02
HCI Synchronous Data packet	0x03
HCI Event packet	0x04
HCI ISO Data packet	0x05

The EM Transport only supports receiving Command Packets (0x01) and sending Event Packets (0x04). All other kinds of packets are invalid when the EM Transport is active.

11.1.2. PACKET FORMATS

The packet formats are compliant with the Bluetooth® Core Specification. The following Figure 11-1 and Figure 11-2 show the different packet formats. The detailed description of the parameters is out of the scope of this document but they are available in the Bluetooth® Core Specification.

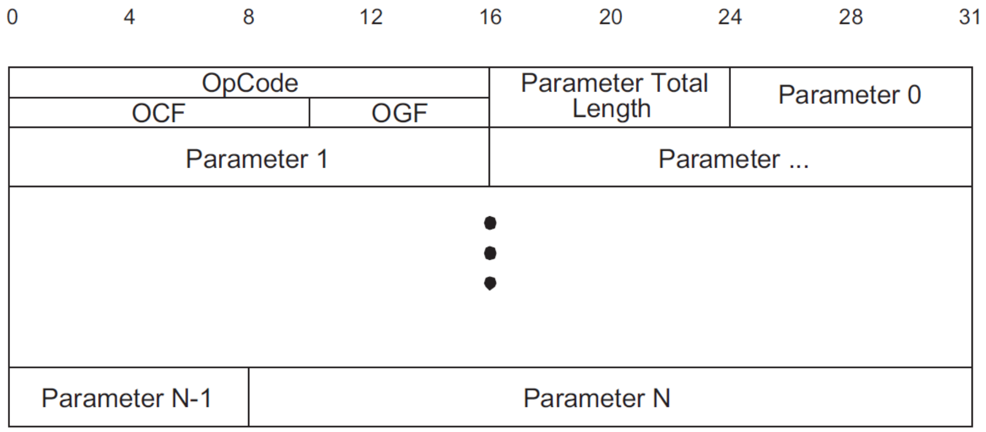


Figure 11-1: HCI command packet format

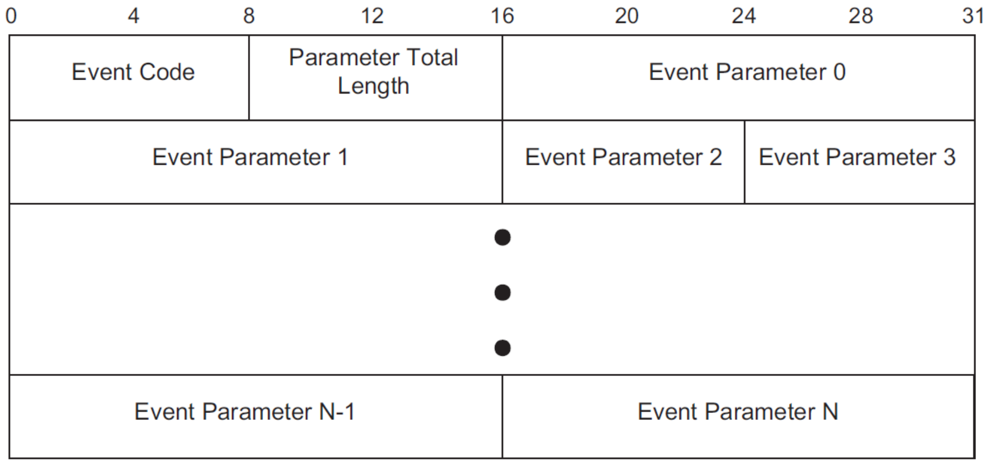


Figure 11-2: HCI event packet format

11.1.3. OPCODES FORMAT

The opcode of all the HCI commands is split into two parts as defined in the Bluetooth® Core Specification: The Opcode Group Field (OGF) and the Opcode Command Field (OCF).

All the EM System Commands use the Vendor Specific Group (OGF = 0x3F). To allow the EM System Commands to be classified in different groups, the OCF is split into two parts: the EM Opcode Group Field (EMOGF) and the EM Opcode Command Field (EMOCF).

The Table 11-2 shows the length of each part of the opcode:

Table 11-2: Opcode format

OPCODE (16-bits)		
OGF (6-bits)	OCF (10-bits)	
OGF (6-bits)	EMOGF (4-bits)	EMOCF (6-bits)

11.2. EM SYSTEM COMMANDS

The EM System Layer is able to parse all the vendor specific HCI commands (also called EM System (EMS) Commands), execute the required actions, and send responses and events to the host.

11.2.1. PROTECTED EM SYSTEM (EMS) COMMANDS

When no one is authenticated, the protected commands are locked. The system supports two levels of authentication (USER and EM). All the EM System Commands are available in Configuration Mode until the “Authentication Required” flags are written in the NVM (auto-authentication).

The set of protected commands can be unlocked using the EMS Security (EMSS) Commands. The unlock mechanism uses an AES authentication for the user.

11.2.2. BACKWARD COMPATIBILTY

The commands shown in Table 11-3 are backward compatible with the previous products. It means that both old and new opcodes are valid for each of them, and the parameters are the same. The detailed description of the commands in the following sections shows only the new opcodes as the old ones are deprecated and will be removed in the future. The old opcodes shall not be used in new tools.

Table 11-3: EM System Commands Backward Compatible

OLD COMMAND	OLD OPCODE	NEW COMMAND	NEW OPCODE
Set Public Address	**0xFC02**	EMSG Set Public Address	0xFC43
Set UART Baud Rate	**0xFC07**	EMSG Set UART Baud Rate	0xFC44
Transmitter Test	**0xFC11**	EMSRC Transmitter Test	0xFCC1
Transmitter Test End	**0xFC12**	EMSRC Transmitter Test End	0xFCC2
Read At Address	**0xFC20**	EMSMM Read At Address	0xFD01
Read Continue	**0xFC21**	EMSMM Read Continue	0xFD02
Write At Address	**0xFC22**	EMSMM Write At Address	0xFD03
Write Continue	**0xFC23**	EMSMM Write Continue	0xFD04
Set Power Mode	**0xFC24**	EMSG Set Power Mode	0xFC48
Set RF Activity	**0xFC25**	EMSRC Set RF Activity	0xFCC3
Set RF Power	**0xFC26**	EMSRC Set RF Power	0xFCC4
Write Patch Start	**0xFC27**	EMSPM Write Patch Start	0xFD81
Write Patch Continue	**0xFC28**	EMSPM Write Patch Continue	0xFD82
Write Patch Abort	**0xFC29**	EMSPM Write Patch Abort	0xFD83
Set Clock Source	**0xFC2A**	EMSG Set Clock Source	0xFC45
Get Memory Usage	**0xFC2C**	EMSG Get Memory Usage	0xFC47
Set Sleep Options	**0xFC2D**	EMSG Set Sleep Options	0xFC49
SVLD Measurement	**0xFC2E**	EMSG SVLD Measurement	0xFC4A
CPU Reset	**0xFC32**	EMSG CPU Reset	0xFC42
Calculate CRC32	**0xFC33**	EMSG Calculate CRC32	0xFC4E
Patch Query	**0xFC34**	EMSPM Patch Query	0xFD84

11.2.3. EMS GENERAL (EMSG) COMMANDS

The overview of the EMSG commands is shown in Table 11-4.

Table 11-4: EM System General (EMSG) commands overview

COMMAND	OPCODE	IN ROM (CM)	AUTHENTICATION
EMSG Read Product Information	**0xFC01**	YES	-
EMSG Read Supported Features	**0xFC41**	-	-
EMSG CPU Reset	**0xFC42**	YES	-
EMSG Set Public Address	**0xFC43**	-	-
EMSG Set UART Baud Rate	**0xFC44**	-	-
EMSG Set Clock Source	**0xFC45**	YES	USER
EMSG Set HF Clock Frequency	**0xFC46**	YES	USER
EMSG Get Memory Usage	**0xFC47**	-	-
EMSG Set Power Mode	**0xFC48**	YES	USER
EMSG Set Sleep Options	**0xFC49**	-	-
EMSG SVLD Measurement	**0xFC4A**	-	-
EMSG Execute JLI Function	**0xFC4B**	YES	EM
EMSG Execute Function	**0xFC4C**	YES	EM
EMSG Jump To Function	**0xFC4D**	YES	EM
EMSG Calculate CRC32	**0xFC4E**	YES	USER
EMSG Enter Configuration Mode	**0xFC4F**	-	-
EMSG Leave Configuration Mode	**0xFC50**	YES	-

11.2.3.1. EMSG Read Product Information

Return some information about the device (RAW data).

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC01
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Device Information	X	Device information (RAW data).	

A Command Complete Event is generated containing the device information.

11.2.3.2. EMSG Read Supported Features

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC41
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

11.2.3.3. EMSG CPU Reset

Execute a CPU reset. The chip will reboot in the current mode (Active or Configuration).

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC42
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated prior to issuing the CPU reset operation.

11.2.3.4. EMSG Set Public Address

Set the device public address.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC43
Public Address	6	Device Public Address.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated prior to issuing the CPU reset operation.

11.2.3.5. EMSG Set UART Baud Rate

Set the UART baud rate to the specified value.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC44
Baud Rate	1	Baud rate selection. 0x00 = 1200 Baud 0x01 = 2400 Baud 0x02 = 4800 Baud 0x03 = 9600 Baud 0x04 = 14400 Baud 0x05 = 19200 Baud 0x06 = 28800 Baud 0x07 = 38400 Baud 0x08 = 57600 Baud 0x09 = 76800 Baud 0x0A = 115200 Baud 0x0B = 230400 Baud 0x0C = 460800 Baud 0x0D = 921600 Baud 0x0E = 1843200 Baud 0x0F – 0xFF Reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the success reception of this command. After issuing the Command Complete Event, the UART baud rate will be changed.

11.2.3.6. EMSG Set Clock Source

Set the high frequency clock source.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC45
Clock Source	1	Clock source selection. 0x00 = High Frequency RC 0x01 = High Frequency Crystal 0x02 – 0xFF Reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the success reception of this command.

11.2.3.7. EMSG Set HF Clock Frequency

Set the high frequency clock frequency.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC46
Clock Source	1	Clock source selection. 0x00 = Reserved 0x01 = 48MHz 0x02 – 0xFF Reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the success reception of this command.

11.2.3.8. EMSG Get Memory Usage

Get memory usage statistics.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC47
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
RAM Pool Size	4	Total size of the memory pool in bytes.	
PRAM Used	4	Retention memory used in bytes.	
NPRAM Used	4	Non-retention memory used in bytes.	
PRAM Reserved	4	Size in bytes of memory reserved as retention memory.	

Memory reserved as retention memory can be used for non-retention data, but the data will be maintained in sleep mode.

A Command Complete Event is generated containing the memory usage statistics.

11.2.3.9. EMSG Set Power Mode

Force the system to go in the specified power mode.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC48
Power Mode	1	Power Mode selection. 0x00 = Active mode 0x01 = CPU Sleep mode 0x02 = Sleep mode 0x03 = Deep Sleep mode 0x04 – 0xFF reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after setting the sleep option.

11.2.3.10. EMSG Set Sleep Options

Enable or disable sleep mode.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC49
Sleep Options	1	Sleep Options Settings. 0x00 = Disable automatic sleep mode. 0x01 = Enable automatic sleep mode. 0x02 – 0xFF reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after setting the sleep option.

11.2.3.11. EMSG SVLD Measurements

Get the Power Configuration and the Supply Voltage Level Detector (SVLD) value corresponding to the SVLD comparator level. The source of the SVLD value is VBAT1 in DCDC Step-Down or Direct Power (DCDC off) Configurations, and VCC in DCDC Step-Up or Voltage Multiplier Configurations.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4A
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Power Configuration	1	Power Management Setups. 0x00 = DCDC Step-down configuration. 0x01 = Direct Power (DCDC off) configuration. 0x02 = DCDC Step-up configuration. 0x03 = Voltage Multiplier configuration. 0x04 – 0xFF Reserved	
SVLD Value	1	SVLD comparator value.	

A Command Complete Event is generated containing the Power Configuration and the SVLD Value.

11.2.3.12. EMSG Execute JLI Function

Execute a function through the JLI table.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4B
JLI Entry	2	JLI Entry Number.	
Number of Arguments	1	Number of arguments of the function (maximum = 4).	
Argument 1	4	First argument of the function.	
Argument 2	4	Second argument of the function.	
Argument 3	4	Third argument of the function.	
Argument 4	4	Fourth argument of the function.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Return Value	4	Return value of the function.	

A Command Complete Event is generated containing the return value of the function.

11.2.3.13. EMSG Execute Function

Execute a function using its address.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4C
Function Address	4	Address of the function.	
Number of Arguments	1	Number of arguments of the function (maximum = 4).	
Argument 1	4	First argument of the function.	
Argument 2	4	Second argument of the function.	
Argument 3	4	Third argument of the function.	
Argument 4	4	Fourth argument of the function.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Return Value	4	Return value of the function.	

A Command Status Event is generated after the reception of the command.

If the command is valid, a Command Complete Event is generated after the completion of the called function containing the return value of this one.

11.2.3.14. EMSG Jump To Function

Jump to a function using its address. Do not wait until the command returns.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4D
Function Address	4	Address of the function.	
Number of Arguments	1	Number of arguments of the function (maximum = 4).	
Argument 1	4	First argument of the function.	
Argument 2	4	Second argument of the function.	
Argument 3	4	Third argument of the function.	
Argument 4	4	Fourth argument of the function.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated containing the return value of the function.

11.2.3.15. EMSG Calculate CRC32

Calculate the 32-bit CRC for the specified continuous address range.

The used polynomial is Ethernet CRC: x32+x26+x23+x22+x16+x12+x11+x10+x8+x7+x5+x4+x2+x+1.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4E
Start Address	4	Starting address (included in the calculation).	
End Address	4	Ending address (excluded from the calculation).	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
CRC32	4	Calculated 32-bit CRC value.	

Addresses values outside of the physical memory address space may result in unexpected behavior of the embedded software.

A Command Complete Event is generated containing the calculated 32-bit CRC value.

11.2.3.16. EMSG Enter Configuration Mode

Enter Configuration Mode. A CPU reset operation is performed.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC4F
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated prior to issuing the CPU reset operation.

11.2.3.17. EMSG Leave Configuration Mode

Leave Configuration Mode. A CPU reset operation is performed.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC50
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated prior to issuing the CPU reset operation.

11.2.4. EMS SECURITY (EMSS) COMMANDS

The overview of the EMSS commands is shown in Table 11-5.

Table 11-5: EM System Security (EMSS) commands overview

COMMAND	OPCODE	IN ROM (CM)	AUTHENTICATION
EMSS EM Get Challenge	**0xFC81**	YES	-
EMSS EM Authenticate	**0xFC82**	YES	-
EMSS USER Write Private Key	**0xFC83**	YES	USER
EMSS USER Get Challenge	**0xFC84**	YES	-
EMSS USER Authenticate	**0xFC85**	YES	-
EMSS EM Read Public Key	**0xFC86**	YES	-

11.2.4.1. EMSS Get Challenge

Send a Commitment and get a Verifier Challenge for EM authentication.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC81
Commitment X	32	Commitment – X part	
Commitment Y	32	Commitment – Y part	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Verifier Challenge	32	Verifier Challenge	

A Command Complete Event is generated containing a Verifier Challenge.

11.2.4.2. EMSS EM Authenticate

Authenticate (EM level) by sending the calculated signature. This command can be called only once by generated Verifier Challenge.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC82
Signature	32	Signature	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.4.3. EMSS USER Write Private Key

Write the AES Private Key for USER authentication in the Key Container #0.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC83
Private Key	16	Private Key	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.4.4. EMSS USER Get Challenge

Get a Verifier Challenge for USER authentication.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC84
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Verifier Challenge	16	Verifier Challenge	

A Command Complete Event is generated containing a Verifier Challenge.

11.2.4.5. EMSS USER AUTHENTICATE

Authenticate (USER level) by sending the calculated signature. This command can be called only once by generated Verifier Challenge.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC85
Signature	16	Signature	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.4.6. EMSS EM Read Public Key

Read the EM Authentication Public Key.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFC86
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Public Key X	32	EM Authentication Public Key – X part	
Public Key Y	32	EM Authentication Public Key - Y part	

A Command Complete Event is generated containing the EM Authentication Public Key.

11.2.5. EMS RADIO CONTROL (EMSRC) COMMANDS

The overview of the EMSRC commands is shown in Table 11-6.

Table 11-6: EM System Radio Control (EMSRC) commands overview

COMMAND	OPCODE	IN ROM (CM)	AUTHENTICATION
EMSRC Transmitter Test	**0xFCC1**	-	-
EMSRC Transmitter Test V2	**0xFCCA**	-	-
EMSRC Transmitter Test V3	**0xFCCB**	-	-
EMSRC Transmitter Test End	**0xFCC2**	-	-
EMSRC Set RF Activity	**0xFCC3**	-	-
EMSRC Set RF Power	**0xFCC4**	-	-
EMSRC Receiver Test	**0xFCC5**	-	-
EMSRC Receiver Test V2	**0xFCCC**	-	-
EMSRC Receiver Test V3	**0xFCCD**	-	-
EMSRC Receiver Test End	**0xFCC6**	-	-

11.2.5.1. EMSRC Transmitter Test

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC1
TX test mode	1	TX test mode (PM = Packet Mode, CM = Continuously modulated signal): 0x00 = PM (sending packets as in Bluetooth® LE DTM mode). Not a full implementation of DTM. Please use the standard HCI command for DTM if doing more than just testing the radio can send packetized data. 0x01 = CM - PRBS9 sequence 0x02 = CM - PRBS15 sequence 0x03 = CM - '0000000000000000' sequence 0x04 = CM - '1111111111111111' sequence 0x05 = CM - '0101010101010101' sequence 0x06 = CM - '0011001100110011' sequence 0x07 = CM - '0000111100001111' sequence 0x08 = CM - '0000000011111111' sequence All other values Reserved for future use.	
TX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
Test Data Length	1	Length in bytes of payload data in each packet	
Packet Payload	1	Packet payload type: 0x00 = PRBS9 sequence 0x01 = Repeated '11110000' sequence 0x02 = Repeated '10101010' sequence 0x03 = PRBS15 0x04 = Repeated '11111111' sequence 0x05 = Repeated '00000000' sequence 0x06 = Repeated '00001111' sequence 0x07 = Repeated '01010101' sequence All other values Reserved for future use	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding All other values Reserved for future use.	
Transmit Power Level	1	Transmit power level Range: -28 to +20 Units: dBm	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Transmit Power Level Max	1	Maximum transmit power level available. Units: dBm	
Transmit Power Level Actual	1	Actual transmit power level used. Units: dBm	

Parameters Test Data Length and Packet Payload are used only in packet mode. In continuous modulation mode, those two parameters are not used and are ignored.

11.2.5.2. EMSRC Transmitter Test V2

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCCA
TX test mode	1	TX test mode (PM = Packet Mode, CM = Continuously modulated signal): 0x00 = PM (sending packets as in Bluetooth® LE DTM mode). Not a full implementation of DTM. Please use the standard HCI command for DTM if doing more than just testing the radio can send packetized data. 0x01 = CM - PRBS9 sequence 0x02 = CM - PRBS15 sequence 0x03 = CM - '0000000000000000' sequence 0x04 = CM - '1111111111111111' sequence 0x05 = CM - '0101010101010101' sequence 0x06 = CM - '0011001100110011' sequence 0x07 = CM - '0000111100001111' sequence 0x08 = CM - '0000000011111111' sequence All other values Reserved for future use.	
TX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
Test Data Length	1	Length in bytes of payload data in each packet	
Packet Payload	1	Packet payload type: 0x00 = PRBS9 sequence 0x01 = Repeated '11110000' sequence 0x02 = Repeated '10101010' sequence 0x03 = PRBS15 0x04 = Repeated '11111111' sequence 0x05 = Repeated '00000000' sequence 0x06 = Repeated '00001111' sequence 0x07 = Repeated '01010101' sequence All other values Reserved for future use	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding All other values Reserved for future use.	
Transmit Power Level	1	Transmit power level Range: -127 to +20 Units: dBm	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Transmit Power Level Max	1	Maximum transmit power level available. Units: dBm	
Transmit Power Level Actual	1	Actual transmit power level used. Units: dBm	

Parameters Test Data Length and Packet Payload are used only in packet mode. In continuous modulation mode, those two parameters are not used and are ignored.

11.2.5.3. EMSRC Transmitter Test V3

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCCB
TX test mode	1	TX test mode (PM = Packet Mode, CM = Continuously modulated signal): 0x00 = PM (sending packets as in Bluetooth® LE DTM mode). Not a full implementation of DTM. Please use the standard HCI command for DTM if doing more than just testing the radio can send packetized data. 0x01 = CM - PRBS9 sequence 0x02 = CM - PRBS15 sequence 0x03 = CM - '0000000000000000' sequence 0x04 = CM - '1111111111111111' sequence 0x05 = CM - '0101010101010101' sequence 0x06 = CM - '0011001100110011' sequence 0x07 = CM - '0000111100001111' sequence 0x08 = CM - '0000000011111111' sequence All other values Reserved for future use.	
TX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
Test Data Length	1	Length in bytes of payload data in each packet	
Packet Payload	1	Packet payload type: 0x00 = PRBS9 sequence 0x01 = Repeated '11110000' sequence 0x02 = Repeated '10101010' sequence 0x03 = PRBS15 0x04 = Repeated '11111111' sequence 0x05 = Repeated '00000000' sequence 0x06 = Repeated '00001111' sequence 0x07 = Repeated '01010101' sequence All other values Reserved for future use	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding Note: If using a Constant Tone Extension (CTE), coded PHY is not supported. All other values Reserved for future use.	
Transmit Power Level	1	Transmit power level Range: -127 to +20 Units: dBm	
CTE Length	1	Length of Constant Tone Extension 0x00 = Do not transmit a Constant Tone Extension 0x02 – 0x14 = Length of the Constant Tone Extension in 8 microsecond units. All other values Reserved for future use.	
CTE Type	1	Type of Constant Tone Extension 0x00 = AoA Constant Tone Extension 0x01 = AoD Constant Tone Extension with 1 microsecond slots 0x02 = AoD Constant Tone Extension with 2 microsecond slots All other values Reserved for future use.	
Switching Pattern Length	1	0x02 to 0x05 = The number of Antenna IDs in the pattern All other values Reserved for future use.	
Antenna IDs	Switching Pattern Length x 1	0xXX = Antenna ID in the pattern	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Transmit Power Level Max	1	Maximum transmit power level available. Units: dBm	
Transmit Power Level Actual	1	Actual transmit power level used. Units: dBm	

Parameters Test Data Length and Packet Payload are used only in packet mode. In continuous modulation mode, those two parameters are not used and are ignored.

11.2.5.4. EMSRC Transmitter Test End

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC2
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Number of Packets	2	Number of packets transmitted.	

11.2.5.5. EMSRC Set RF Activity

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC3
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

11.2.5.6. EMSRC Set RF Power

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC4
Transmit Power Level	1	Transmit power level: Range: -127 to +20 Unites: dBm	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Transmit Power Level Max	1	Maximum transmit power level available. Units: dBm	
Transmit Power Level Actual	1	Actual transmit power level used. Units: dBm	

11.2.5.7. EMSRC Receiver Test

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC5
RX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding All other values Reserved for future use.	
Syncword	4	Sync word (Access Address) used in receiver test. Note: Sync word for Bluetooth® LE RF test mode is 0x71764129.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

11.2.5.8. EMSRC Receiver Test V2

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCCC
RX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding All other values Reserved for future use.	
Syncword	4	Sync word (Access Address) used in receiver test. Note: Sync word for Bluetooth® LE RF test mode is 0x71764129.	
Modulation Index	1	0x00 = Assume transmitter will have a standard modulation index 0x01 = Assume transmitter will have a stable modulation index All other values Reserved for future use.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

11.2.5.9. EMSRC Receiver Test V3

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCCD
RX Channel	1	RF channel N = (F-2402) / 2 Range: 0x00 to 0x27 Frequency Range: 2402 MHz to 2480 MHz	
PHY	1	PHY mode: 0x01 = Transmitter set to use the LE 1M PHY 0x02 = Transmitter set to use the LE 2M PHY 0x03 = Transmitter set to use the LE Coded PHY with S=8 data coding 0x04 = Transmitter set to use the LE Coded PHY with S=2 data coding All other values Reserved for future use.	
Syncword	4	Sync word (Access Address) used in receiver test. Note: Sync word for Bluetooth® LE RF test mode is 0x71764129.	
Modulation Index	1	0x00 = Assume transmitter will have a standard modulation index 0x01 = Assume transmitter will have a stable modulation index All other values Reserved for future use.	
Expected CTE Length	1	Length of Constant Tone Extension 0x00 = No Constant Tone Extension expected 0x02 – 0x14 = Expected length of the Constant Tone Extension in 8 microsecond units. All other values Reserved for future use.	
Expected CTE Type	1	Type of Constant Tone Extension 0x00 = Expect AoA Constant Tone Extension 0x01 = Expect AoD Constant Tone Extension with 1 microsecond slots 0x02 = Expect AoD Constant Tone Extension with 2 microsecond slots All other values Reserved for future use.	
Slot Durations	1	0x01 = Switching and sampling slots are 1 microsecond each 0x02 = Switching and sampling slots are 2 microseconds each All other values Reserved for future use.	
Switching Pattern Length	1	0x02 to 0x05 = The number of Antenna IDs in the pattern All other values Reserved for future use.	
Antenna IDs	Switching Pattern Length x 1	0xXX = Antenna ID in the pattern	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

11.2.5.10. EMSRC Receiver Test End

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFCC6
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Number of Packets	2	Number of packets received.	
Average RSSI	2	Average RSSI of all received packets.	

To compute the average Received Signal Strength Indicator (RSSI) in the Receive Test End command, we need to convert the RSSI value from hexadecimal to a negative signed integer and divide it by 10. This will give us the average RSSI value in dBm.

11.2.6. EMS MEMORY MANAGEMENT (EMSMM) COMMANDS

The overview of the EMSMM commands is shown in Table 11-7.

Table 11-7: EM System Memory Management (EMSMM) commands overview

COMMAND	OPCODE	IN ROM (CM)	AUTHENTICATION
EMSMM Read At Address	**0xFD01**	YES	USER
EMSMM Read Continue	**0xFD02**	YES	USER
EMSMM Write At Address	**0xFD03**	YES	USER
EMSMM Write Continue	**0xFD04**	YES	USER
EMSMM NVM Erase Full	**0xFD05**	YES	EM
EMSMM NVM Erase Main	**0xFD06**	YES	USER
EMSMM NVM Erase Page	**0xFD07**	YES	USER
EMSMM NVM Get Lock Page	**0xFD08**	YES	USER
EMSMM NVM Lock Page	**0xFD09**	YES	USER
EMSMM NVM Power Control	**0xFD0A**	YES	EM
EMSMM Write At Address Without Response	**0xFD0B**	YES	USER
EMSMM Write Continue Without Response	**0xFD0C**	YES	USER

11.2.6.1. EMSMM Read At Address

Read one or more bytes at the given address. The maximum number of bytes to read is inferred from the HCI Event packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD01
Start Address	4	Address to start reading.	
Data Length	1	Number of bytes to read.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Data	Data Length	Data read from the specified address.	

A Command Complete Event is generated after the completion of this command.

11.2.6.2. EMSMM Read Continue

Read one or more bytes beginning from where the previous read command ended. This allows for continuous reading without supplying the address. The maximum number of bytes to read is inferred from the HCI Event packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD02
Data Length	1	Number of bytes to read.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Data	Data Length	Data read.	

A Command Complete Event is generated after the completion of this command.

11.2.6.3. EMSMM Write At Address

Write one or more bytes at the given address. The maximum number of bytes to write is inferred from the HCI Command packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD03
Start Address	4	Address to start writing.	
Data	Data Length	Data to write to the specified address.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.4. EMSMM Write Continue

Write one or more bytes beginning from where the previous write command ended. This allows for continuous writing without supplying the address. The maximum number of bytes to write is inferred from the HCI Command packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD04
Data	Data Length	Data to write.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.5. EMSMM NVM Erase Full

Perform an NVM Full Erase operation. It erases both main and information areas.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD05
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.6. EMSMM NVM Erase Main

Perform an NVM Main Erase operation. It erases only the main area.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD06
None	0	This command contains no parameters.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.7. EMSMM NVM Erase Page

Perform an NVM Page Erase operation. It erases the page given in parameter.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD07
Area	1	Area selection. 0x00 = Main area 0x01 = Information area 0x02 – 0xFF Reserved	
Page	1	Page number in the given area.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.8. EMSMM NVM Get Lock Page

Get the lock status of a page in NVM.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD08
Area	1	Area selection. 0x00 = Main area 0x01 = Information area 0x02 – 0xFF Reserved	
Page	1	Page number in the given area.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Lock	1	Lock status of the page. 0x00 = Page unlocked 0x01 = Page locked 0x02 – 0xFF Reserved	

A Command Complete Event is generated after the completion of this command.

11.2.6.9. EMSMM NVM Lock Page

Lock a page in NVM. This command only temporary locks a page. A power on reset (POR) restores the original behavior (page not locked).

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD09
Area	1	Area selection. 0x00 = Main area 0x01 = Information area 0x02 – 0xFF Reserved	
Page	1	Page number in the given area.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.10. EMSMM NVM Power Control

Power up or power down the NVM.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD0A
Action	1	Action selection. 0x00 = Power Down the NVM 0x01 = Power Up the NVM 0x02 – 0xFF Reserved	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.6.11. EMSMM Write At Address Without Response

Write one or more bytes at the given address. The maximum number of bytes to write is inferred from the HCI Command packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD0B
Start Address	4	Address to start writing.	
Data	Data Length	Data to write to the specified address.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command only in case of an error. If the status is SUCCESS (0x00) no event is sent.

11.2.6.12. EMSMM Write Continue Without Response

Write one or more bytes beginning from where the previous write command ended. This allows for continuous writing without supplying the address. The maximum number of bytes to write is inferred from the HCI Command packet size.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD0C
Data	Data Length	Data to write.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command only in case of an error. If the status is SUCCESS (0x00) no event is sent.

11.2.7. EMS REGISTER MANAGEMENT (EMSRM) COMMANDS

The overview of the EMSRM commands is shown in Table 11-8.

Table 11-8: EM System Register Management (EMSRM) commands overview

COMMAND	OPCODE	IN ROM (CM)	AUTHENTICATION
EMSRM Write AUX Register	**0xFD41**	YES	EM
EMSRM Read AUX Register	**0xFD42**	YES	EM

11.2.7.1. EMSRM Write AUX Register

Write a register in the AUX register space.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD41
Address	4	Address of the AUX register.	
Data	4	Data to write to the specified address.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	

A Command Complete Event is generated after the completion of this command.

11.2.7.2. EMSRM Read AUX Register

Read a register in the AUX register space.

PARAMETER	SIZE (BYTES)	DESCRIPTION	OPCODE: 0xFD42
Address	4	Address of the AUX register.	
RETURN PARAMATERS			
Status	1	Standard Bluetooth® Error Code.	
Data	4	Data read from the specified address.	

A Command Complete Event is generated after the completion of this command.

11.3. EM SYSTEM EVENTS

The Table 11-9 lists the EM system events (vendor specific events) that are supported. All EM System Events use Event code dedicated for vendor events (Event Code = 0xFF). To support multiple vendor events, Subevent Code is defined in Event Parameter 0 in event packet – see Figure 11-2. Detailed information on EM System Events can be found in the SDK documentation. Please contact EM Microelectronic for up-to-date SDK release.

Table 11-9: EM System Events overview

COMMAND	EVENT CODE	SUBEVENT CODE	IN ROM (CM)
EMS Active State Entered	0xFF	0x01	-
EMS Configuration Mode Entered	0xFF	0x03	YES
EMS HAL Notification	0xFF	0x04	-

11.3.1. EMS ACTIVE STATE ENTERED

Reports that the active state was entered. This event is sent after one of the following conditions:

- Any hardware reset including POR after link layer is enabled
- Link layer is enabled after exiting from Production Test Mode (PTM)
- Active state is entered after resuming from sleep following the successful execution of the EMSG Set Power Mode HCI command.
- Active state is entered after the host activates the Wakeup pin when the UART is enabled without flow control.

PARAMETER	SIZE (BYTES)	DESCRIPTION	EVENT CODE: 0xFF
Subevent Code	1	0x01	

11.3.2. EMS CONFIGURATION MODE ENTERED

Reports that Configuration Mode was entered and device is ready for communication.

PARAMETER	SIZE (BYTES)	DESCRIPTION	EVENT CODE: 0xFF
Subevent Code	1	0x03	

11.3.3. EMS HAL NOTIFICATION

Reports on the HAL notification. Notifications provide information or warnings such as indicating a hardware error is pending.

PARAMETER	SIZE (BYTES)	DESCRIPTION	EVENT CODE: 0xFF
Subevent Code	1	0x04	
Notification Event	1	0x00 = No event 0x01 = NVM disabled due to voltage level 0x02 = RF power reduced due to voltage level 0x03 = RF power increased due to restored voltage level 0x04 = RF disabled due to voltage level 0x05 = Power voltage level critical 0x06 = PLL lock lost 0x07 = Memory manager returned a null pointer 0xFF = Unknown error	

11.4. HARDWARE ERROR EVENT CODES

The hardware error event is a standard Bluetooth® event where error codes are vendor defined. The following table defines these vendor specific hardware error codes.

PARAMETER	SIZE (BYTES)	DESCRIPTION	
Error Code	1	0x00 = No error 0x01 = HCI synchronization lost 0x02 = Reserved 0x03 = Reserved 0x04 = RF system error 0x05 = CPU reset (watchdog) 0x06 = CPU reset (bus error) 0x07 = Crystal oscillator start-up error 0x08 = CRC error in NVM 0x80 = Device not programmed 0xFF = Unknown error	

12. TYPICAL APPLICATION

Several reference designs are described in this section.

The recommended components list and their size is shown in Table 12-1.

Table 12-1: Recommended components list and their size

Designator	Value	0201	0402	0603	Other
Cvio	100nF	-	GRM155R 62A104KE14D	-	-
Cvbat2/Crc	1µF	-	GRM155R 61C105MA12D	-	-
Cvcc/Cvbat1	2.2µF	-	GRM155R 61C225KE11D	-	-
Ldcdc	4.7µH (*2.2 µH)	-	(*MLZ1005M2R2W T000)	LQM18PN4R7 MFRL	-
FB	3µH	-	BLM15HG 601SN1D	-	-
Cvbus	10µF	-	-	GRM188R 61C106MAALD	-
Rvbus	1Ω	-	RC0402FR-071RL	-	-
Ldc	See Table 12-2 and Table 12-3	-	LQG15HS1N5 B02D	-	-
Cdc1	See Table 12-2 and Table 12-3	Type GRM0335C	-	-	-
Cm1	See Table 12-2 and Table 12-3	Type GRM0335C	-	-	-
Cm2	See Table 12-2 and Table 12-3	Type GRM0335C	-	-	-
Cm3	See Table 12-2 and Table 12-3	Type GRM0335C	-	-	-
Lm1	See Table 12-2 and Table 12-3	Type LQP03TN	-	-	-
Lm2	See Table 12-2 and Table 12-3	Type LQP03TN	-	-	-
Cc1	220pF	GRM0335C 1H221JA01D	-	-	-
X32k (Optional)	32.768kHz	-	-	-	Micro Crystal CM7V-T1A
X48M	48MHz	-	-	-	Hosonic ETSB48E007502E

The size of the matching network will depend on the maximum TX output power of the application. A good matching network depends as well on the PCB design. An initial selection can be made looking at the Table 12-2, where the components for the QFN reference design (see Section 12.1) are listed. Table 12-3 shows an initial selection for the WLCSP reference design.

Table 12-2: Components selection for the matching network on QFN reference design

	Ldc	Cdc1	Cm1	Lm1	Cm2	Lm2	Cm3	Cc1
Up to 0dBm	1.5nH	18pF		2.4nH	2pF			*
+6dBm	1.5nH	18pF	2.0pF	2.4nH	2pF			*
+10dBm	1.5nH	39pF	1.5pF	2.4nH	2pF	2.2nH	1.0pF	*

Table 12-3: Components selection for the matching network on WLCSP reference design

	Ldc	Cdc1	Cm1	Lm1	Cm2	Lm2	Cm3	Cc1
+6dBm	2.0nH	18pF	1.6pF	2.4nH	2.4pF			*

Cc1 (220pF) is required if the antenna is referenced to ground. It is not required if the antenna is referenced to the supply.

Other optimized reference designs are available and other components (size and value) can be used. An example is Ldcdc = 2.2 µH that can reduce the PCB footprint. Please contact EM Microelectronic for more information.

12.1. REFERENCE DESIGNS

The reference designs for QFN and WLCSP follow the schematic shown in Figure 12-1. The needed components for each power configuration are shown in Table 12-4. Underfill is recommended for the WLCSP assembly.

A resistor on the Enable line is recommended when using the WLCSP package. It is not needed when using the QFN package.

Contact EM Microelectronic to get detailed information about schematic and layout guidelines.

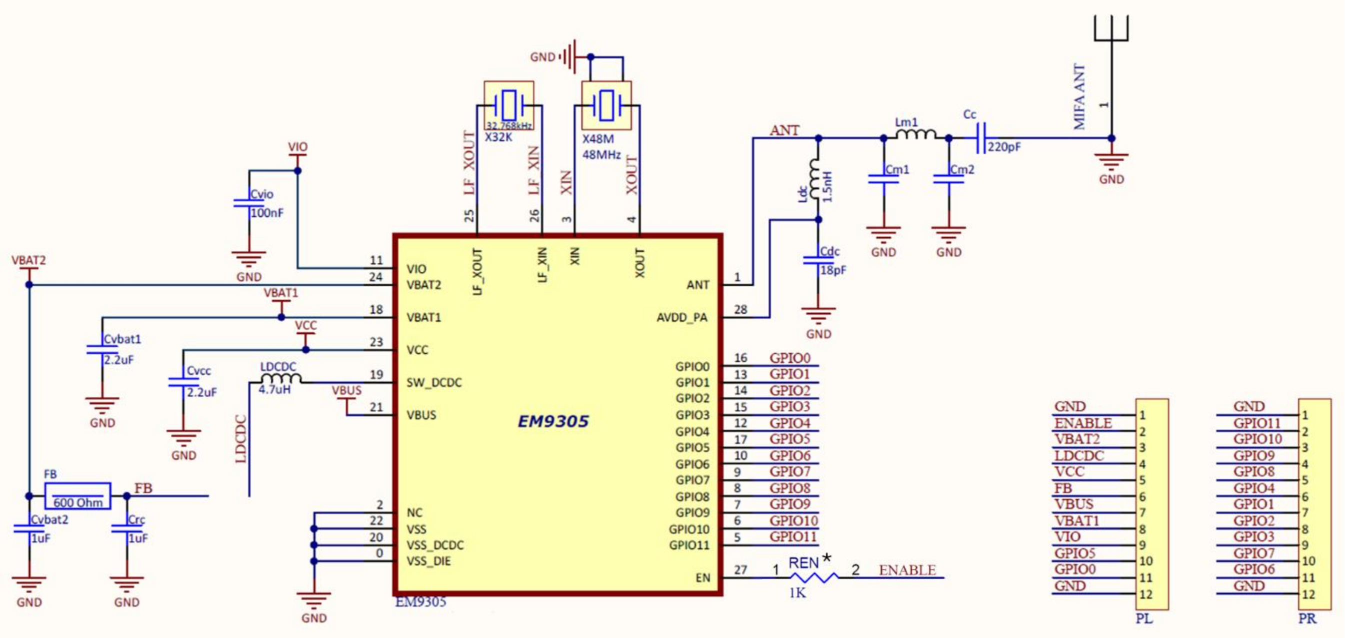


Figure 12-1: General Reference Design Schematic

Table 12-4: Power management components for the different configurations

QFN	CVIO	CVBAT2	CVBAT1	CVCC	CRC	LDCDC	FB
Step-Down	X	X		X	X	X	X
Step Up	X	X	X		X	X	X
Voltage multiplier	X	X	X	X			

12.2. DEVELOPMENT KIT

An EM9305 DVK (see Figure 12-2) is available for the development and as an evaluation platform for the EM9305 Bluetooth® LE Radio.

The key features of the DVK include:

- Reference Design Sockets with configurable power mode selection
- Optional Level Shifters to test with IO signals other than 3.3V
- 4 power supply options for VIO & VBAT2
- Expansion header to pair custom sensors to the EM9305

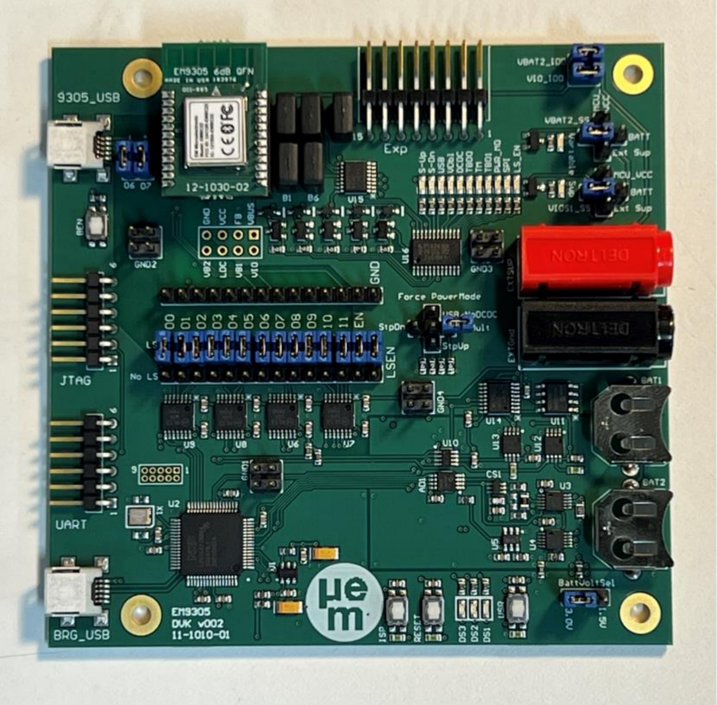


Figure 12-2: EM9305 Development Kit

Please contact EM Microelectronic for an application note and for more information.

13. ORDERING INFORMATION

The EM9305 is available in the following version:

Table 13-1: EM9305 versions and ordering information

TYPE	DESCRIPTION	PACKAGING	DELIVERY FORM - UNITS	PART NUMBER
Integrated Circuit	**Standard version**	QFN-28	Tape on reel – 2’500	EM9305*B*-S*yyy*-LF28B
	Bluetooth® LE SoC	WLCSP23	Tape on reel – 13’000	EM9305*B*-S*yyy*-CS23B
	With +6dBm output power	Tested wafer	Wafer – 18’500 (est.)	EM9305*B*-S*yyy*-WW31
	High output power version	QFN-28	Tape on reel – 2’500	EM9305B-H*yyy*-LF28B
	Bluetooth® LE SoC	WLCSP23	Tape on reel – 13’000	EM9305B-H*yyy*-CS23B
	With +10dBm output power	Tested wafer	Wafer – 18’500 (est.)	EM9305B-H*yyy*-WW31
	USB version	QFN-28	Tape on reel – 2’500	EM9305B-U*yyy*-LF28B
	Bluetooth® LE SoC			
	With +10dBm and USB capabilities	Tested wafer	Wafer – 18’500 (est.)	EM9305B-U*yyy*-WW31
Development kit	**Fully featured development kit**	Board	PCB – 1	EMDVK9305SOC
Reference design	**QFN with antenna**	Board	PCB – 1	EMREF9305QFN

The Minimum Order Quantity (MOQ) for standard products is 10’000 units for QFN-28, and 13’000 units for WLCSP23. For dedicated products, the MOQ corresponds to 1 wafer (18’500 units) and is subject to approval by EM Microelectronic.

The EM9305 is available with a modular SDK that can include the full set of Bluetooth® Core 5.4 features (Direction Finding, Isochronous Channels). The EM9305 comes with technical support and the possibility of software IC upgrade in the future.

The EM9305 is certified by the Bluetooth® SIG against Bluetooth® Core 5.4.

The access to the Bluetooth® LE stack is done through an Application Programming Interface (API) in application mode.

The version of the IC identified by “yyy” will be last hardware version available and communicated by EM Microelectronic.

For specific software features or other delivery formats, please contact EM Microelectronic representative.

14. PACKAGING INFORMATION

14.1. QFN PACKAGE INFORMATION

EM9305 is available in a QFN-28 4mm x 4mm package. The QFN-28 package mechanical drawing is shown in Figure 14-1. For the pin assignment, refer to Table 3-1.

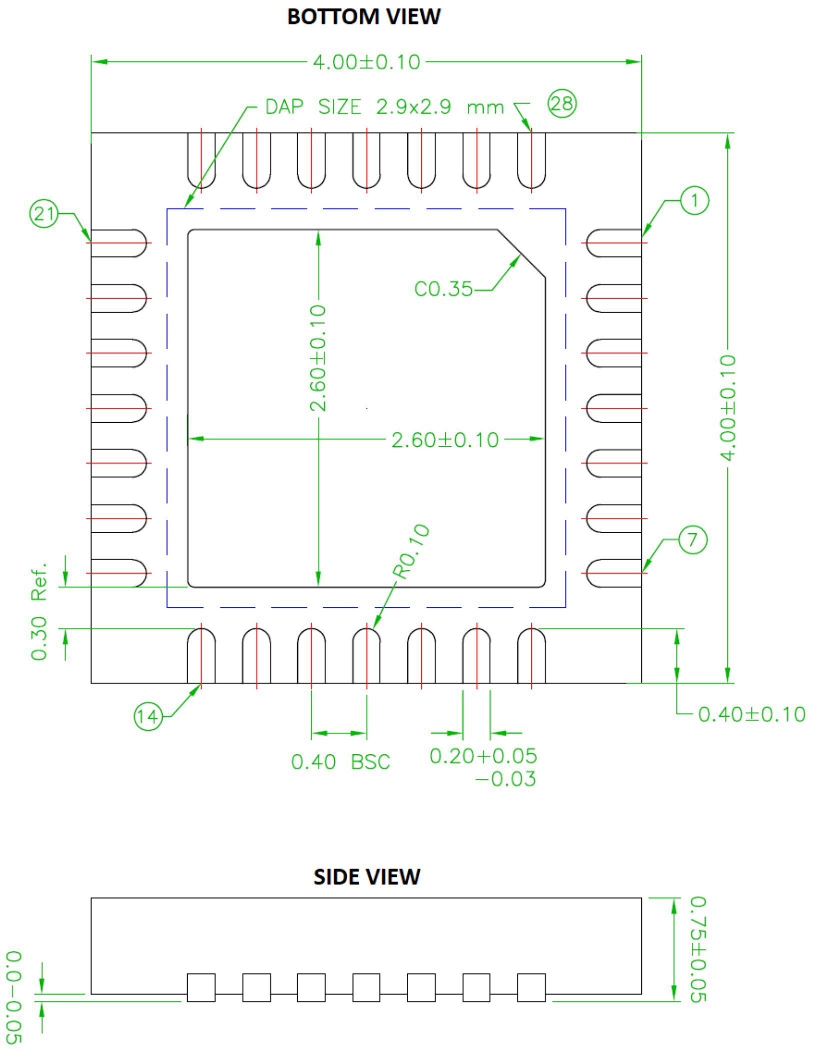


Figure 14-1: QFN-28 Mechanical Drawing (All dimensions are in millimeters)

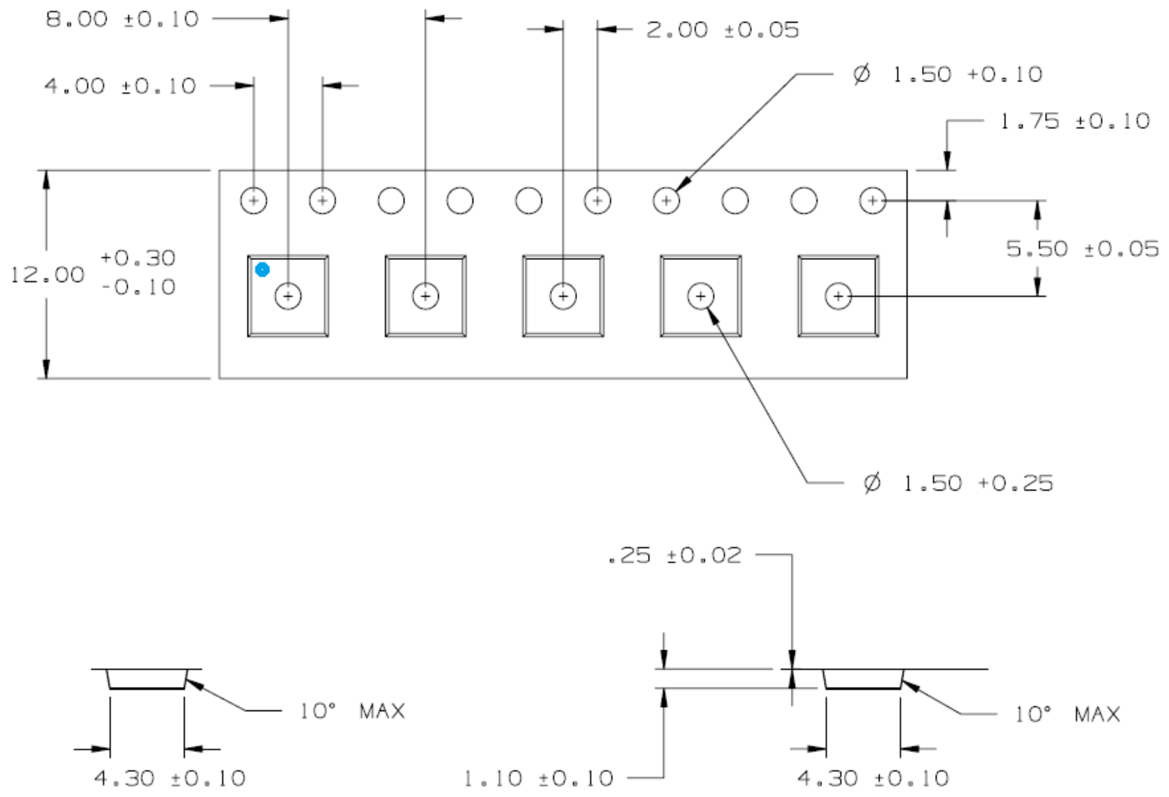


Figure 14-2 Tape & Reel diagram. Blue dot shows pin 1 orientation on QFN tape (All dimensions are in millimeters)

14.1.1. QFN PACKAGE MARKING

The EM9305 markings for the QFN package are shown in Table 14-1

Line A indicates the product number. Code B1-3 indicates the product version, code B4 indicates last digit of year of assembly, and code B5 indicates package information. Additional marking in line C is used for lot traceability.

Table 14-1: Package marking information for QFN device

	1	2	3	4	5	6
A	E	M	9	3	0	5
B	1	0	5	3	X	
C	*	*	*	*	*	*

The example in Table 14-1 shows an EM9305, version 105, produced in 2023 (2) in a QFN package.

14.2. WLCSP PACKAGE INFORMATION

The EM9305 is available as a Wafer Level Chip Scale Package (WLCSP23). The package has 23 balls in a 1.8mm x 1.8mm, 5x5 ball array with 0.35mm pitch. The mechanical drawing is summarized in Figure 14-3.

When using the WLCSP package, particular care has to be taken since this is an extremely small package where the four sides of the IC are exposed. For more information on the unwanted photodiode effect, you can request the application note [600003] to EM Microelectronics.

For shelf life, please refer to EM Microelectronic document “Shelf life of EM IC products (490008)”.

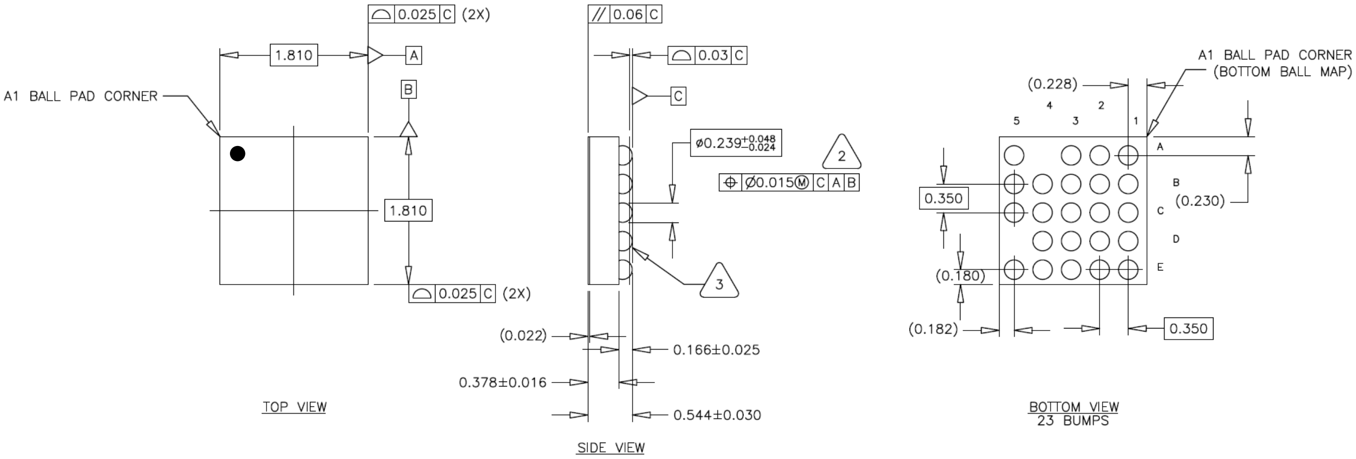
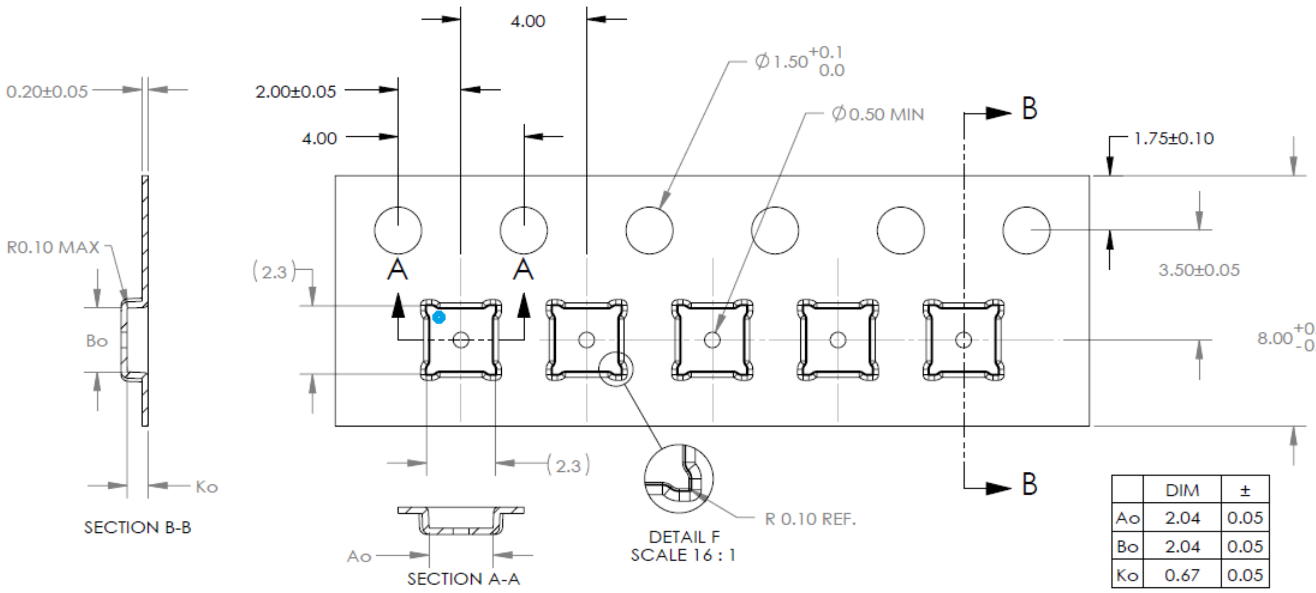


Figure 14-3: WLCSP23 Mechanical Drawing (All dimensions are in millimeters)

WLCSP ball pad corner A1 in Figure 14-3 is marked on the flat backside of the die (top view) so that that the orientation of the component can be easily determined before and after assembly. This dot has been designed so that it can be detected by standard vision systems.



	DIM	±
Ao	2.04	0.05
Bo	2.04	0.05
Ko	0.67	0.05

Figure 14-4 Tape & Reel diagram. Blue dot shows pin A1 orientation on WLCSP tape (All dimensions are in millimeters)

14.2.1. WLCSP Package Marking

The EM9305 markings for the WLCSP package are shown in Table 14-2.

Line A indicates the product number. Code B4-6 indicates the product version. Additional marking in line C is used for lot traceability.

Table 14-2: Package marking information for WLCSP device

	1	2	3	4	5	6
A	E	M	9	3	0	5
B		*	*	2	0	7
C			*	*	*	*

14.2.2. WLCSP Package Reflow

Figure 14-5 shows the typical Temperature / Time Reflow profile for lead-free solder (SnAgCu) with recommended parameter values. The specification details are in Table 14-3. The optimum profile may depend on many factors such as the oven type, the solder type, the temperature difference across the board, the oven temperature / thermocouples tolerance etc. and must be fine-tuned to establish a robust process.

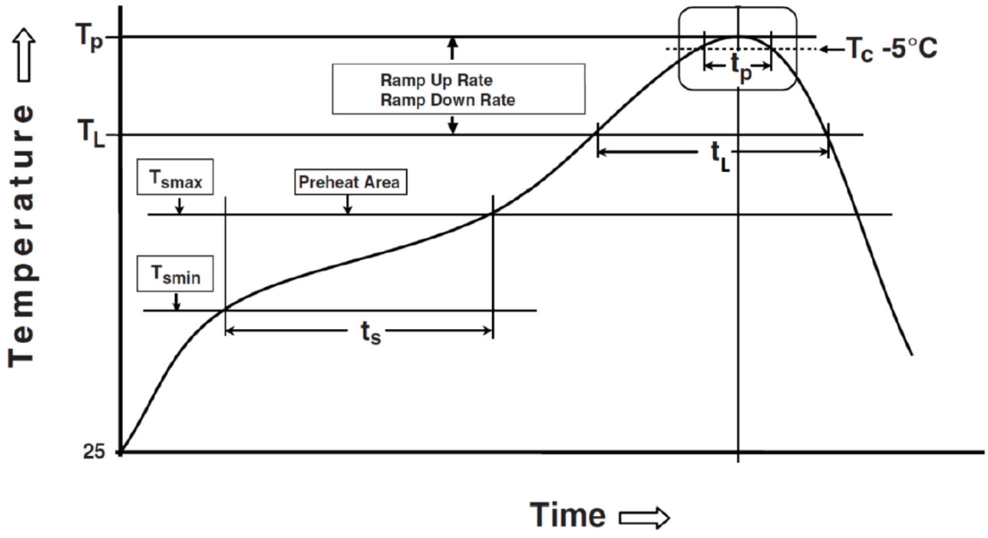


Figure 14-5: WLCSP Temperature vs Time Reflow Profile

Table 14-3: WLCSP Reflow Specifications

PROFILE FEATURE (PREHEAT/SOAK)	Pb-FREE ASSEMBLY	UNIT
Temperature Min (T_{smin})	150	°C
Temperature Max (T_{smax})	180	°C
Time (t_s) from T_{smin} to T_{smax}	60-180	s
Ramp-up rate from T_L to T_p	3	°C/s
Liquidous temperature (T_L)	220	°C
Time (t_L) maintained above T_L	30-90	s
Peak package body temperature (T_p)	260	°C
Time (t_p) within 5 °C of the specified Peak package body temperature (T_p)	10-20	s
Ramp-down rate from T_p to T_L (max)	6	°C/s

14.3. DIE INFORMATION

The EM9305 is available as a Wafer Level Chip die with pad location diagram shown in Figure 14-3. The pad location is shown in Table 14-4, where the origin (0; 0) is the intersection of the outer left pad X coordinate with the lowest pad Y coordinate.

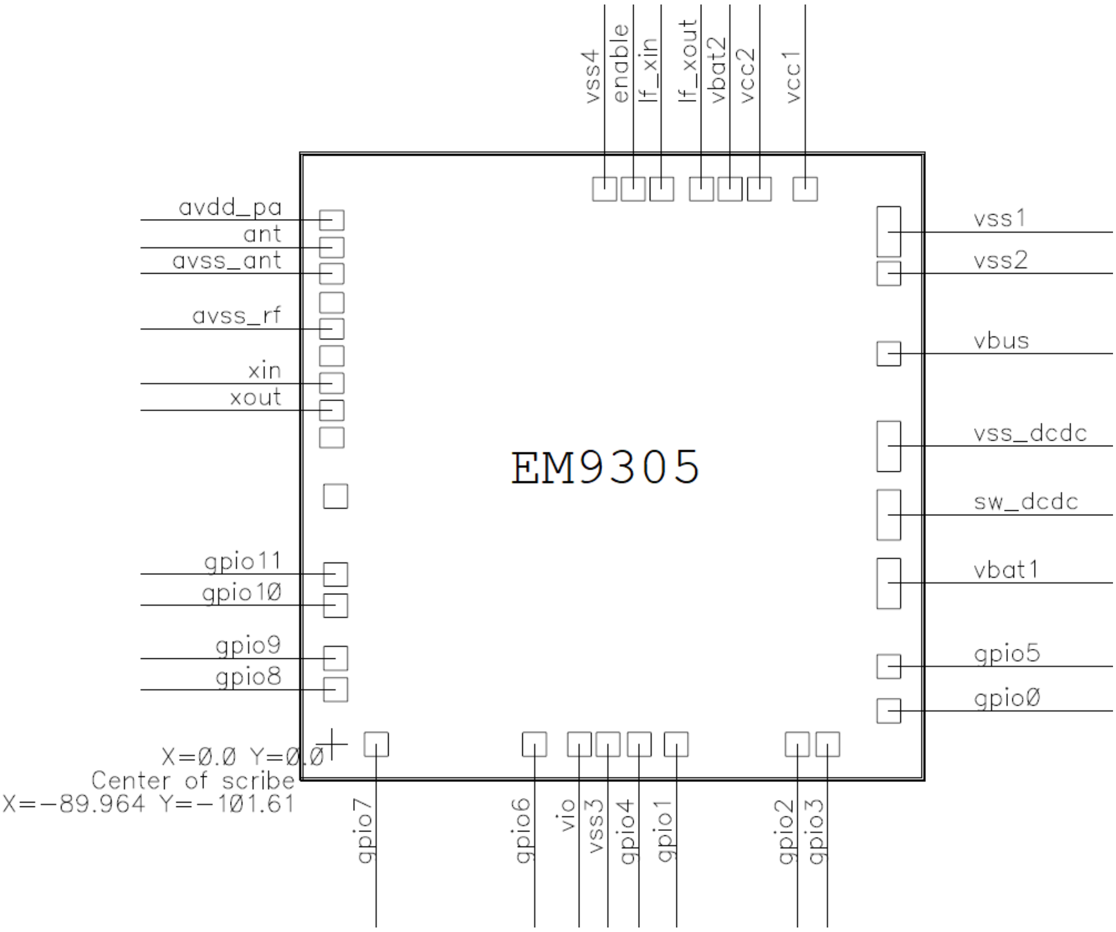


Figure 14-6: Pad location diagram

Table 14-4: Pad position (top view)

Pad number	Pad name	Pad location X (µm)	Y (µm)	Pad opening size width (µm)	length (µm)
1	GPIO8	11.6	161.6	68.0	68.0
2	GPIO9	11.6	253.8	68.0	68.0
3	GPIO10	11.6	409.0	68.0	68.0
4	GPIO11	11.6	501.2	68.0	68.0
5	dnc				
6	dnc				
7	XOUT	0.0	984.2	66.6	55.8
8	XIN	0.0	1064.3	66.6	55.8
9	dnc				
10	AVSS_RF	0.0	1224.5	66.6	55.8
11	dnc				
12	AVSS_ANT	0.0	1387.4	66.6	55.8
13	ANT	0.0	1464.8	66.6	55.8
14	AVDD_PA	0.0	1544.9	66.6	55.8
15	VSS4	807.8	1637.2	68.0	68.0
16	EN	892.2	1637.2	68.0	68.0
17	LF_XIN	976.6	1637.2	68.0	68.0
18	LF_XOUT	1094.9	1637.2	68.0	68.0
19	VBAT2	1179.3	1637.2	68.0	68.0
20	VCC2	1265.5	1637.2	68.0	68.0
21	VCC1	1401.6	1637.2	68.0	68.0
22	VSS1	1648.8	1511.2	68.0	146.0
23	VSS2	1648.8	1387.8	68.0	68.0
24	VBUS	1648.8	1151.2	68.0	68.0
25	VSS_DCDC	1648.8	878.5	68.0	146.0
26	SW_DCDC	1648.8	676.6	68.0	146.0
27	VBAT1	1648.8	474.6	68.0	146.0
28	GPIO5	1648.8	229.5	68.0	68.0
29	GPIO0	1648.8	98.5	68.0	68.0
30	GPIO3	1468.0	0.0	68.0	68.0
31	GPIO2	1378.0	0.0	68.0	68.0
32	GPIO1	1019.7	0.0	68.0	68.0
33	GPIO4	910.2	0.0	68.0	68.0
34	VSS3	817.5	0.0	68.0	68.0
35	VIO	733.1	0.0	68.0	68.0
36	GPIO6	600.3	0.0	68.0	68.0
37	GPIO7	132.2	0.0	68.0	68.0

15. DATASHEET VERSION HISTORY

This section summarizes all the updates done in the various distributed or published datasheets.

15.1. Version 4.5.3 (August 20th, 2025)

- Throughout the datasheet
    - Corrected wrong cross-references.
    - Following the Communication Guidelines for Supported Bluetooth® Functionality.
    - Updated minimum TX output power from -28dBm to -33dBm.
    - Removed wording “secure” near the word “boot” to avoid confusion.
- Table 1-1
    - Updated VBAT1 of Voltage Multiplier when VBAT2 is between 1.9V and 2.5V.
- Section 1.2.4
    - Removed requirement of minimum supply voltage ramp-up in the automatic configuration detection.
- Section 4.3.1
    - Detailed measurement of sleep current.
- Section 6.1
    - Voltage multiplier configuration updated for VBAT1 level from 1.7V to 1.9V.
- Table 6-1
    - VCC changed as an external supply for Step-up and Voltage Multiplier.
- Section 10 and Figure 10-1
    - Refactored diagrams related to software static block architecture.
- Figure 10-7, Figure 10-8 and Figure 10-9
    - Added the diagrams and removed former figure showing JLI and OTA.
- Section 10.2.2
    - Clarified storage of the Bluetooth® LE host and controller in various modes.
- Section 10.4.1
    - Replaced command EMSMM_NVM_ERASE_FULL by EMSRM_READ_AUX_REGISTER for authentication checking. Added command line and result screenshot.
- Section 10.7.2
    - Minimum sleep time reduced from 15ms to 3ms.
- Table 12-1
    - Added mention that 32kHz crystal is optional.
- Section 13
    - Added Minimum Order Quantity.
- Section 14.2 and Figure 14-3
    - Added dot description for A1 WLCSP ball identification.

15.2. Version 4.5.2 (October 29th, 2024)

- Section 1.1
    - Updated number of simultaneous connections.
- Table 1-1
    - Upgraded voltage multiplier supply range up to 2.5V.
- Table 5-1
    - Corrected remapping of IRAM1 and IRAM2.
- Table 6-1
    - Updated supply range for Voltage Multiplier configuration.
- Table 6-6
    - Added 250ppm Bluetooth® Sleep Clock Accuracy (SCA).
- Section 8.2
    - Detailed description of GPIO wake-up mechanism including Figure 8-6 and Figure 8-7 to provide a visual explanation of the wake-up process.
- Section 8.10.3
    - Added default protocol timer clock frequency.
- Figure 10-1
    - Updated software architecture.
- Figure 10-2 and Figure 10-3
    - Corrected address of DRAM5 and DRAM6
- Section 12
    - Simplified the description of the reference designs.
- Section 14.2
    - Added reference to shelf life.

15.3. Version 4.5.1 (December 18th, 2023)

- Throughout the datasheet
  - Removed “Confidential” status from Datasheet.
  - Update to Bluetooth® Core 5.4.
- Section 1.4
  - Updated Bluetooth® specification documents.
- Section 11.2.5.10
  - Added description and updated size of Average RSSI.
- Section 14.2
  - Added shelf life.

15.4. Version 4.5.0 (September 28th, 2023)

- Throughout the datasheet
  - Removed yellow highlights coming from version 4.4.1.
- Table 1-3
  - Removed I2C Fast+ 1MHz mode
- Table 4-3
  - Updated RX and TX currents.
  - Removed details about TX @-57dBm (no PA mode).
  - Updated current consumption in sleep, deep sleep and retention.
- Table 4-11
  - Updated start-up time with the inclusion of the default 1ms precharge time.
- Table 4-12
  - Updated mode transition time.
- Table 6-6
  - Specified conditions with defined XTAL component.
- Section 8.1
  - Minor changes in the wording description.
- Section 8.2
  - Updated description of the GPIO status at software boot.
- Table 10-2
  - Added information about that info page 3 is locked.
- Section 10.4.2
  - Removed erroneous auto-authentication in NVM boot sequence.
- Section 11.2.5
  - Added v2 and v3 Transmitter and Receiver Test commands.
- Table 12-3
  - New table: WLCSP matching components.
- Table 13-1
  - Changed reel size of WLCSP to 13’000.
- Section 16
  - Removed Errata Sheet section.

15.5. Version 4.4.3 (September 28th, 2023)

- Throughout the datasheet
  - Removed yellow highlights coming from version 4.4.1.

15.6. Version 4.4.2 (September 6th, 2023)

- Table 6-6
  - Added accuracy of LF RC clock after calibration.

15.7. Version 4.4.1 (June 30th, 2023)

- Throughout the datasheet
  - Highlighted values/information that might change in the next version of the datasheet.
  - In particular, see Table 1-1, Table 4-3, Table 6-7, Table 12-1.
- Section 8.1
  - Updated ADC details for sources and measurement ranges.
- Section 10.4.1
  - Added a description of the authentication procedure.

- Table 12-1
  - Added 2.2µH reference component.

15.8. Version 4.4.0 (February 23rd, 2023)

- Throughout the datasheet
  - Specifically mentioned USB availability only on QFN/die versions.
  - Removed the wording “programming” for JTAG interface.
- Table 4-3, Table 4-6
  - Added 4dBm and 5dBm modes.
  - Updated consumption of 8dBm and 10dBm modes.
- Table 6-1
  - Update of the voltage multiplier output tolerance.
- Section 8.9
  - Update of the temperature accuracy.
- Figure 10-2
  - Update of the RAM layout representation.
- Figure 10-3
  - Update of the data RAM space with persistence strategy.
- Figure 10-4
  - Update of the NVM layout representation.
- Section 11.2.4.4
  - Corrected size of Verifier Challenge.
- Section 11.3
  - Added reference to SDK documentation for updated EM System Events.
- Table 12-1
  - Update of Vbat1 capacitor value.

15.9. Version 4.3.0 (November 4th, 2022)

- Table 4-3
  - Update of the current consumption for high TX output power modes.
- Table 4-7
  - Update of the out-of-band blocking performances.
- Table 8-2
  - Added requirements on GPIO5 source.
- Section 12.1
  - Added underfill recommendation for WLCSP.

15.10. Version 4.2.0 (August 31st, 2022)

- Table 4-3
  - Added Voltage Multiplier configuration.
  - Added RX consumption in 1 and 2 Mbps modes.
- Section 4.3.1
  - Update of the description for the case of USB supply (below Table 4-3).
- Table 4-5
  - Removed shunt capacitor parameter.
  - Update of the minimum differential equivalent load capacitance.
- Section 5.4
  - Removed the figure of the clock controller architecture.
- Section 8.7
  - Update of the phase behavior for 16MHz.
- Section 8.8.2
  - Renamed the section as “SPI full-duplex operation”.
- Section 8.8.3
  - New section: description of the SPI flow control.
- Table 10-3, Table 10-4
  - Update of the structure of the table to identify the memory area covered by CRC field.
  - Update with new content.
- Section 11.2.6
  - New command: EMSMM Write At Address Without Response (Section 11.2.6.11).

  - New command: EMSMM Write Continue Without Response (Section 11.2.6.12).
- Table 12-2
  - Update of the +10dBm matching network.
- Section 15
  - New section: datasheet version history.
- Section 16
  - Errata sheet: Update of the remaining upgrades of the chip.

15.11. Version 4.1.0 (April 20th, 2022)

- Throughout the datasheet
  - Removed “Preliminary” status from Datasheet.
  - Update to Bluetooth® Core 5.3.
  - Update of the parameter values reflect measurements and not simulations.
- Section 1.4
  - Update of related documents to reflect the update of the Bluetooth® specifications.
- Table 4-3
  - Update with additional TX modes.
- Table 4-4
  - Update of the table name containing now only data rates.
- Table 4-5
  - Update of the Differential equivalent load capacitance for Typical supported Xtal parameters.
- Table 4-6
  - Update of the spurious emissions.
- Table 4-7
  - Update in the Sensitivity and simplification of intermodulation values.
- Table 4-8
  - Update of GPIO characteristics with detailed conditions for USB pins.
- Table 4-12
  - Update of timing going from Active RC to Active XTAL.
- Table 4-14
  - Removed I2C Fast+ 1MHz mode.
- Table 4-15
  - Update of SPI Slave output delay and Master output delay.
- Table 4-16
  - Update of the NVM currents.
- Figure 5-3
  - Update of the diagram to identify the RAM blocks that are used either for data either for instructions.
- Section 6.3.2
  - Renamed the section as “Digital Current Meter”.
- Table 6-6
  - Update of lf_rc_lp_clk clock source with better description.
- Section 8.1
  - Update of the ADC description, including figures for ADC application examples.
- Section 8.3
  - Removed I2C Fast+ 1Mbit/s mode.
- Section 8.5
  - Added maximum JTAG clock speed.
- Section 9
  - Update of the introductory security paragraph.
- Section 9.2
  - Update of the Secure Key containers with a detailed description.
- Section 10.4.1
  - Update of ROM Boot Sequence description.
- Section 10.5.1
  - Update of the signal range for Configuration mode detection.
- Table 11-3
  - Update of the backward compatible EM System commands.
- Section 11.2.3.15
  - Added CRC polynomial description.
- Table 11-5
  - Added EM System Security command to read public key.
- Section 11.2.4.6
  - New section: description of the added command to read public key.
- Section 11.2.5.1

  - Update of the EMSRC Transmitter Test returned parameters.
- Section 11.2.5.4
  - Added description of the command EMSRC Set RF Power.
- Table 12-1
  - Update in recommended capacitor values for Cvbat2/Cvbat1/Crc.
- Figures 12-4, 12-6, 12-8
  - Update of the WLCSP reference design layout.
- Section 13
  - Update of ordering information description.
- Table 14-4
  - Added pad opening size.
- Section 15
  - New section: errata sheet showing the roadmap of future chip upgrades.

15.12. Version 3.2.0 (October 27th, 2021)

- Section 1.2.4
  - Added note about supply configuration in WLCSP.
- Figure 3-2
  - Updated mechanical view of WLCSP.
- Table 3-2
  - New table: ball position coordinates for WLCSP.
- Table 4-3
  - Update of the current consumption in TX, RX, sleep modes.
  - Added Coremark test current consumption.
- Table 4-6
  - Update of the minimum voltage for 3dBm power setting.
- Table 4-7
  - Update of sensitivity values.
- Table 4-8
  - Update of hysteresis and high drive voltages.
- Section 4.4.1
  - Removed table of GPIO timing characteristics.
- Table 4-16
  - Update of the NVM minimum supply voltage.
- Section 8.10
  - Update in the description of the universal timer and the protocol timer.
- Section 14.2
  - Removed table of ball position coordinates (moved as table 3-2).
- Figure 14-3
  - Added ball transparency on WLCSP mechanical drawings.

15.13. Version 3.1.3 (June 25th, 2021)

- Section 1.2.4
  - Added note about 1Ω resistor used in USB configuration.
- Table 3-1 and Figure 3-1
  - Added note about AVSS pin 2 in QFN package.
- Table 4-1
  - Added maximum voltage rating for USB DP/DM pin.
- Table 4-3
  - Update of the table for 48MHz CPU speed only.
- Table 4-7
  - Update of sensitivity values.
- Table 4-12
  - Update of timing values.
- Table 4-14
  - Removed input delays.
- Section 4.5
  - New section: NVM characteristics.
- Section 5.2
  - Update of the description.
- Table 5-1

  - Update of the table with additional AHB peripherals.
- Table 6-6
  - Update of the accuracy for lf_rc_clk and lf_rc_lp_clk.
- Table 8-2
  - Update of the specification for use cases with or without S&H.
- Section 8.7
  - Update of the SPI Master clock speeds.
- Section 8.8.1
  - Renamed the section as “SPI half-duplex operation”.
- Section 8.8.2
  - Renamed the section as “SPI full-duplex with flow control operation”.
- Section 8.8.2.1
  - Added a more detailed description about the RDY signal.
- Section 9
  - Updated and reorganized completely the “Security” section with 3 main sub-sections. New description and new figures.
- Section 10
  - Updated and reorganized completely the “Software architecture” section. New description and new figures.
- Section 11
  - Updated and reorganized completely the “EM System Command and Events” section. New description and new figures.
- Section 12
  - Update of the section with new layout figures and addition of dedicated WLCSP reference designs for the different power modes.

15.14. Version 3.0.2 (August 18th, 2020)

First distributed version.

The Bluetooth® word mark and logos are registered trademarks owned by the Bluetooth® SIG, Inc.

EM Microelectronic-Marin SA ("EM") makes no warranties for the use of EM products, other than those expressly contained in EM's applicable General Terms of Sale, located at http://www.emmicroelectronic.com. EM assumes no responsibility for any errors which may have crept into this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein.

No licenses to patents or other intellectual property rights of EM are granted in connection with the sale of EM products, neither expressly nor implicitly.

In respect of the intended use of EM products by customer, customer is solely responsible for observing existing patents and other intellectual property rights of third parties and for obtaining, as the case may be, the necessary licenses.

Important note: The use of EM products as components in medical devices and/or medical applications, including but not limited to, safety and life supporting systems, where malfunction of such EM products might result in damage to and/or injury or death of persons is expressly prohibited, as EM products are neither destined nor qualified for use as components in such medical devices and/or medical applications. The prohibited use of EM products in such medical devices and/or medical applications is exclusively at the risk of the customer.