

FiltSure Device State — Technical Overview & Current Issues

Device Capabilities

The FiltSure sensor node is designed to collect and transmit a variety of environmental and operational data critical for HVAC monitoring and general sensing applications. The device measures and reports:

- Pressure
- Humidity
- Temperature
- RFID scans
- Wind speed
- Battery level

Data transmission is supported over Wi-Fi when available, with LoRa (RFM96) as a fallback path to ensure resiliency in environments with poor or absent Wi-Fi coverage.

Hardware & Architecture

- Main MCU: ESP32-C6 (primary control, Wi-Fi/LoRa stack)
- Sensors:
 - Bosch BME280 (Pressure/Humidity/Temperature)
 - MFRC522 (RFID)
 - Micro-turbine based wind speed sensor
- Communications:
 - Wi-Fi uplink
 - LoRa (SX1276/77/78/79 family – RFM96) point-to-point link
- Power: USB umbilical for bench testing or 3V D-cell in field tests
- Bus Topology: All peripherals share a single SPI bus with separate chip select (CS) lines.

Datasheets and references:

- LoRa (SX1276–79):

https://cdn-shop.adafruit.com/product-files/3179/sx1276_77_78_79.pdf

- MFRC522 RFID:

<https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>

- Bosch BME280 PHT:

<https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme280-ds002.pdf>

- TI TPS61023 Power Regulator:

<https://www.ti.com/lit/ds/symlink/tps61023.pdf?ts=1756468678481>

- Espressif ESP32-C6:

https://www.espressif.com/sites/default/files/documentation/esp32-c6_datasheet_en.pdf

- ESP32-C6 DevKitM-1:

<https://docs.espressif.com/projects/esp-dev-kits/en/latest/esp32c6/esp32-c6-devkitm-1/index.html>

Additional project documentation: <https://github.com/ExecTio9/FiltSureDev>

Current Challenges

1. Multi-Device SPI Bus Conflicts

- Individually, each SPI device (RFID, BME280, LoRa) operates correctly.
- Issue: When multiple peripherals are active simultaneously, communication with the ESP32-C6 becomes unreliable.
- Workaround: Devices have been separated onto multiple boards for independent testing.

2. RFID Instability

- MFRC522 demonstrates inconsistent scanning.
- After some runtime, the module fails entirely, returning blank packets to the server until reset.

3. LoRa Inconsistency

- The RFM96 (SX127x family) initially showed promising results in point-to-point tests.
- However, performance degraded unexpectedly after a few days (with no major environmental changes).
- Symptoms:
 - Packets received only intermittently

- Sessions that previously sustained hundreds of packets now stall after only a handful
- This inconsistency has been the main blocker to stable fallback communications.

4. Power Considerations

- Current prototypes are powered either by USB or a 3V D-cell.
- Early tests suggest the regulator chain is functional, but no deep characterization of power draw vs. stability has been done yet.

5. Software Maturity

- Codebase largely functional (multiple peer reviews by software and EE students).
- Minor updates are required to ensure LoRa compatibility and improve multi-device coordination over SPI.

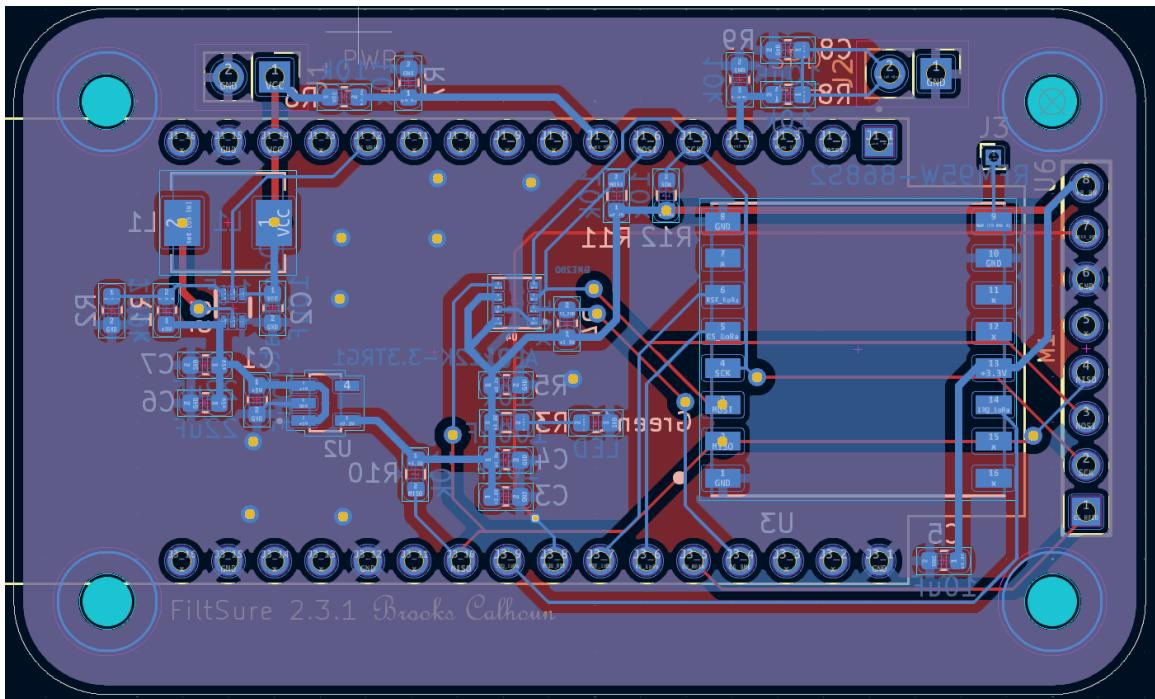
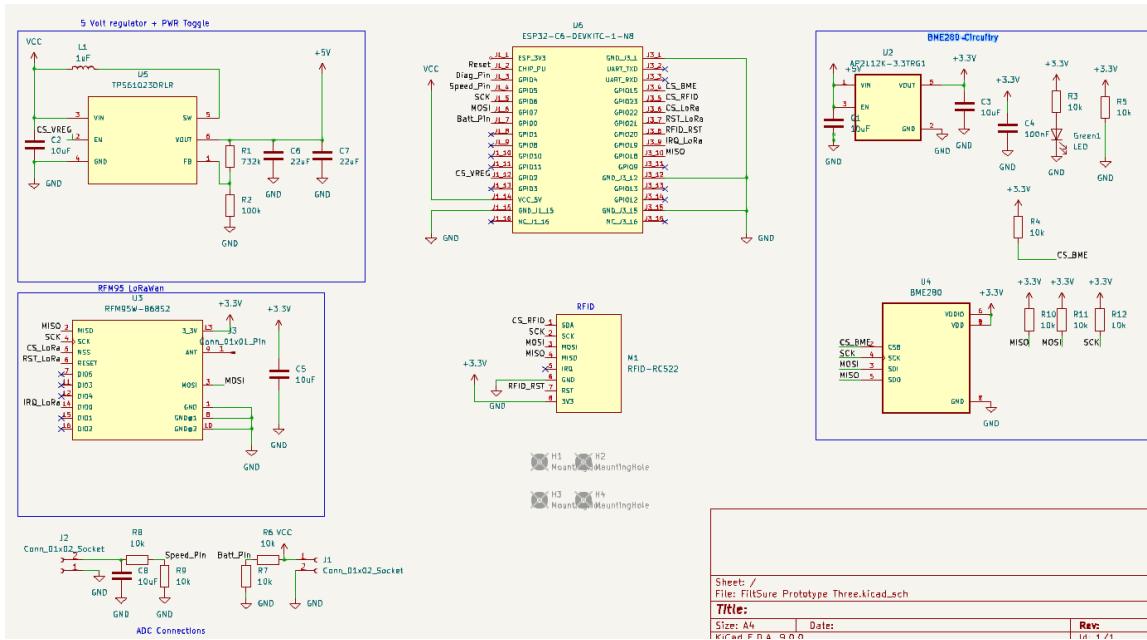
Summary

The FiltSure device is feature-complete in terms of sensing and dual-path communications (Wi-Fi + LoRa).

The main blockers to a stable field deployment are:

1. SPI contention issues with multiple peripherals
2. RFID (MFRC522) instability during extended operation
3. LoRa (RFM96) reliability degradation after initial success

Once these are resolved, the platform will be capable of consistently collecting and transmitting environmental and operational data over redundant communication paths, fulfilling its role as a robust monitoring node.



Past This point are the Datasheets linked in the upper pages

ESP32-C6 Series

Datasheet Version 1.3

Ultra-low-power SoC with RISC-V single-core microprocessor

2.4 GHz Wi-Fi 6 (802.11ax), Bluetooth® 5 (LE), Zigbee and Thread (802.15.4)

Optional flash in the chip's package

30 or 22 GPIOs, rich set of peripherals

QFN40 (5×5 mm) or QFN32 (5×5 mm) package

Including:

ESP32-C6

ESP32-C6FH4

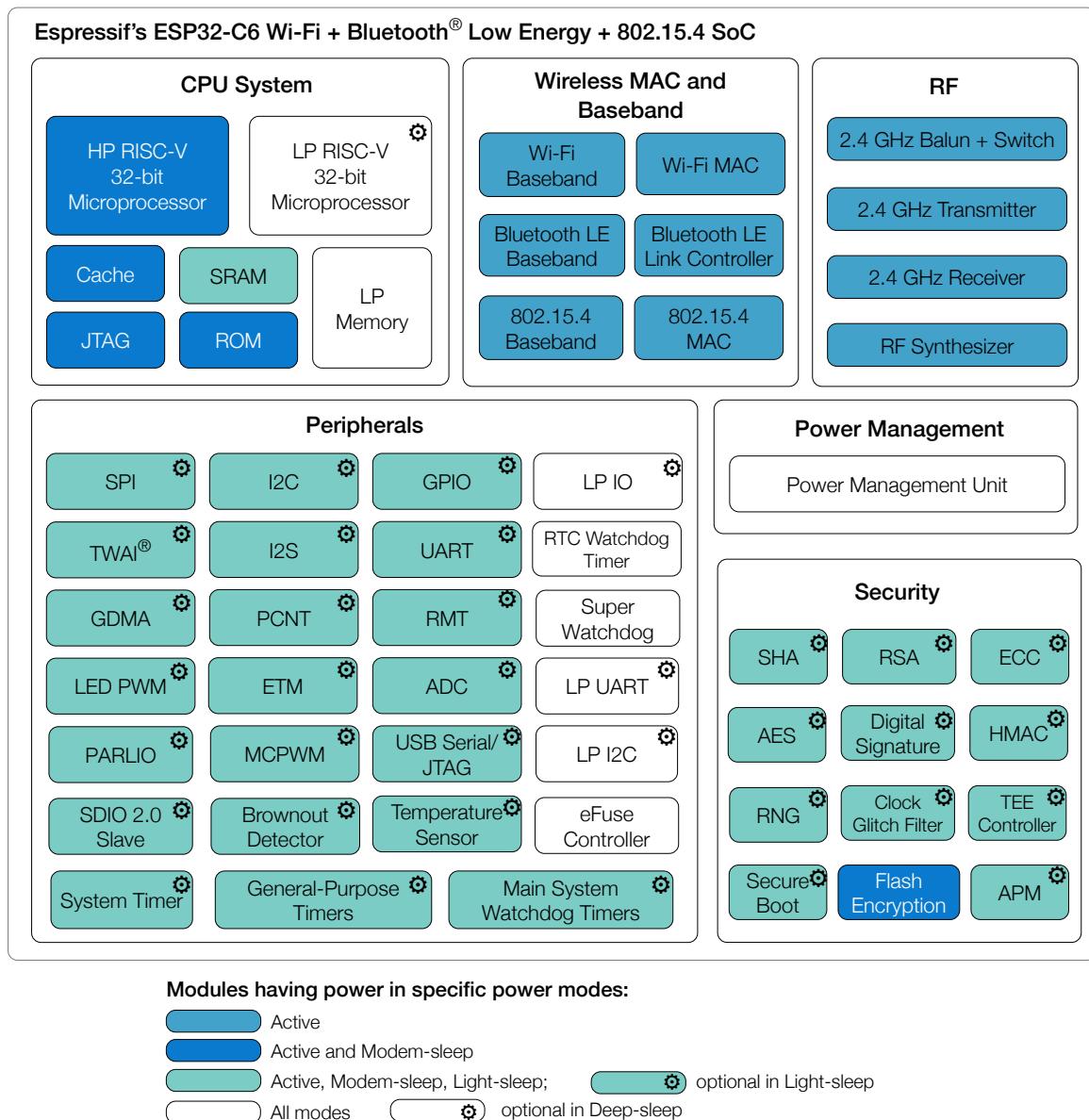
ESP32-C6FH8



Product Overview

The ESP32-C6 SoC (System on Chip) supports Wi-Fi 6 in 2.4 GHz band, Bluetooth 5, Zigbee 3.0 and Thread 1.3. It consists of a high-performance (HP) 32-bit RISC-V processor, a low-power (LP) 32-bit RISC-V processor, wireless baseband and MAC (Wi-Fi, Bluetooth LE, and 802.15.4), RF module, and numerous [peripherals](#). Wi-Fi, Bluetooth and 802.15.4 coexist with each other and share the same antenna.

The functional block diagram of the SoC is shown below.



ESP32-C6 Functional Block Diagram

For more information on power consumption, see Section [4.1.3.7 Power Management Unit](#).

Features

Wi-Fi

- 1T1R in 2.4 GHz band
- Operating frequency: 2412 ~ 2484 MHz
- IEEE 802.11ax-compliant
 - 20 MHz-only non-AP mode
 - MCS0 ~MCS9
 - Uplink and downlink OFDMA, especially suitable for simultaneous connections in high-density environments
 - Downlink MU-MIMO (multi-user, multiple input, multiple output) to increase network capacity
 - Beamformee that improves signal quality
 - Channel quality indication (CQI)
 - DCM (dual carrier modulation) to improve link robustness
 - Spatial reuse to maximize parallel transmissions
 - Target wake time (TWT) that optimizes power saving mechanisms
- Fully compatible with IEEE 802.11b/g/n protocol
 - 20 MHz and 40 MHz bandwidth
 - Data rate up to 150 Mbps
 - Wi-Fi Multimedia (WMM)
 - TX/RX A-MPDU, TX/RX A-MSDU
 - Immediate Block ACK
 - Fragmentation and defragmentation
 - Transmit opportunity (TXOP)
 - Automatic Beacon monitoring (hardware TSF)
 - Four virtual Wi-Fi interfaces
 - Simultaneous support for Infrastructure BSS in Station mode, SoftAP mode, Station + SoftAP mode, and promiscuous mode

Note that when ESP32-C6 scans in Station mode, the SoftAP channel will change along with the Station channel

- Antenna diversity
- 802.11mc FTM

Bluetooth®

- Bluetooth LE: Bluetooth 5.3 certified
- Bluetooth mesh
- High power mode (20 dBm)
- Speed: 125 Kbps, 500 Kbps, 1 Mbps, 2 Mbps
- Advertising extensions
- Multiple advertisement sets
- Channel selection algorithm #2
- LE power control
- Internal co-existence mechanism between Wi-Fi and Bluetooth to share the same antenna

IEEE 802.15.4

- Compliant with IEEE 802.15.4-2015 protocol
- OQPSK PHY in 2.4 GHz band
- Data rate: 250 Kbps
- Thread 1.3
- Zigbee 3.0

CPU and Memory

- HP RISC-V processor:
 - Clock speed: up to 160 MHz
 - Four stage pipeline
 - CoreMark® score: 496.66 CoreMark; 3.10 CoreMark/MHz (160 MHz)
- LP RISC-V processor:
 - Clock speed: up to 20 MHz
 - Two stage pipeline

- L1 cache: 32 KB
- ROM: 320 KB
- HP SRAM: 512 KB
- LP SRAM: 16 KB
- Supported SPI protocols: SPI, Dual SPI, Quad SPI, QPI interfaces that allow connection to flash and other SPI devices off the chip's package
- Flash controller with cache is supported
- Flash in-Circuit Programming (ICP) is supported

Advanced Peripheral Interfaces

- 30 GPIOs (QFN40), or 22 GPIOs (QFN32)
 - 5 strapping GPIOs
 - 6 GPIOs needed for [off-package flash](#)
- Analog interfaces:
 - 12-bit SAR ADC, up to 7 channels
 - Temperature sensor
- Digital interfaces:
 - Two UARTs
 - Low-power (LP) UART
 - Two SPI ports for communication with flash
 - General purpose SPI port
 - I2C
 - Low-power (LP) I2C
 - I2S
 - Pulse count controller
 - USB Serial/JTAG controller
 - Two TWAI® controllers, compatible with ISO 11898-1 (CAN Specification 2.0)
 - SDIO slave controller
 - LED PWM controller, up to 6 channels
 - Motor Control PWM (MCPWM)
 - Remote control peripheral (TX/RX)
 - Parallel IO interface (PARLIO)

- General DMA controller, with 3 transmit channels and 3 receive channels
- Event task matrix (ETM)
- Timers:
 - 52-bit system timer
 - Two 54-bit general-purpose timers
 - Three digital watchdog timers
 - Analog watchdog timer

Power Management

- Fine-resolution power control through a selection of clock frequency, duty cycle, Wi-Fi operating modes, and individual power control of internal components
- Four power modes designed for typical scenarios: Active, Modem-sleep, Light-sleep, Deep-sleep
- Power consumption in Deep-sleep mode is $7 \mu\text{A}$
- Low-power (LP) memory remains powered on in Deep-sleep mode

Security

- Secure boot - permission control on accessing internal and external memory
- Flash encryption - memory encryption and decryption
- 4096-bit OTP, up to 1792 bits for users
- Trusted execution environment (TEE) controller and access permission management (APM)
- Cryptographic hardware acceleration:
 - AES-128/256 (FIPS PUB 197)
 - ECC
 - HMAC
 - RSA
 - SHA (FIPS PUB 180-4)
 - Digital signature

- External Memory Encryption and Decryption (XTS_AES)
- Random Number Generator (RNG)

- Up to +21 dBm of power for an 802.11b transmission
- Up to +19.5 dBm of power for an 802.11ax transmission
- Up to -106 dBm of sensitivity for Bluetooth LE receiver (125 Kbps)

RF Module

- Antenna switches, RF balun, power amplifier, low-noise receive amplifier

Applications

With low power consumption, ESP32-C6 is an ideal choice for IoT devices in the following areas:

- Smart Home
- Industrial Automation
- Health Care
- Consumer Electronics
- Smart Agriculture
- POS Machines
- Service Robot
- Audio Devices
- Generic Low-power IoT Sensor Hubs
- Generic Low-power IoT Data Loggers

Note:

Check the link or the QR code to make sure that you use the latest version of this document:
https://www.espressif.com/documentation/esp32-c6_datasheet_en.pdf



Contents

Product Overview

| | |
|--------------|---|
| Features | 3 |
| Applications | 5 |

1 ESP32-C6 Series Comparison

| | |
|------------------|----|
| 1.1 Nomenclature | 12 |
| 1.2 Comparison | 12 |

2 Pins

| | |
|---|----|
| 2.1 Pin Layout | 13 |
| 2.2 Pin Overview | 15 |
| 2.3 IO Pins | 18 |
| 2.3.1 IO MUX Pin Functions | 18 |
| 2.3.2 LP IO MUX Functions | 21 |
| 2.3.3 Analog Functions | 22 |
| 2.3.4 Restrictions for GPIOs and LP GPIOs | 23 |
| 2.3.5 Peripheral Pin Assignment | 24 |
| 2.4 Analog Pins | 27 |
| 2.5 Power Supply | 28 |
| 2.5.1 Power Pins | 28 |
| 2.5.2 Power Scheme | 28 |
| 2.5.3 Chip Power-up and Reset | 29 |
| 2.6 Pin Mapping Between Chip and Flash | 31 |

3 Boot Configurations

| | |
|--|----|
| 3.1 Chip Boot Mode Control | 33 |
| 3.2 SDIO Sampling and Driving Clock Edge Control | 33 |
| 3.3 ROM Messages Printing Control | 34 |
| 3.4 JTAG Signal Source Control | 35 |

4 Functional Description

| | |
|---------------------------------|----|
| 4.1 System | 36 |
| 4.1.1 Microprocessor and Master | 36 |
| 4.1.1.1 High-Performance CPU | 36 |
| 4.1.1.2 RISC-V Trace Encoder | 36 |
| 4.1.1.3 Low-Power CPU | 37 |

| | | |
|----------|---|----|
| 4.1.1.4 | GDMA Controller | 37 |
| 4.1.2 | Memory Organization | 38 |
| 4.1.2.1 | Internal Memory | 39 |
| 4.1.2.2 | External Memory | 39 |
| 4.1.2.3 | eFuse Controller | 40 |
| 4.1.3 | System Components | 40 |
| 4.1.3.1 | IO MUX and GPIO Matrix | 40 |
| 4.1.3.2 | Reset | 40 |
| 4.1.3.3 | Clock | 41 |
| 4.1.3.4 | Interrupt Matrix | 41 |
| 4.1.3.5 | Event Task Matrix | 42 |
| 4.1.3.6 | System Timer | 42 |
| 4.1.3.7 | Power Management Unit | 43 |
| 4.1.3.8 | Timer Group | 43 |
| 4.1.3.9 | Watchdog Timers | 44 |
| 4.1.3.10 | Permission Control | 44 |
| 4.1.3.11 | System Registers | 44 |
| 4.1.3.12 | Debug Assistant | 45 |
| 4.1.4 | Cryptography and Security Component | 45 |
| 4.1.4.1 | AES Accelerator | 45 |
| 4.1.4.2 | ECC Accelerator | 46 |
| 4.1.4.3 | HMAC Accelerator | 46 |
| 4.1.4.4 | RSA Accelerator | 46 |
| 4.1.4.5 | SHA Accelerator | 47 |
| 4.1.4.6 | Digital Signature | 47 |
| 4.1.4.7 | External Memory Encryption and Decryption | 47 |
| 4.1.4.8 | Random Number Generator | 48 |
| 4.2 | Peripherals | 49 |
| 4.2.1 | Connectivity Interface | 49 |
| 4.2.1.1 | UART Controller | 49 |
| 4.2.1.2 | SPI Controller | 49 |
| 4.2.1.3 | I2C Controller | 50 |
| 4.2.1.4 | I2S Controller | 51 |
| 4.2.1.5 | Pulse Count Controller | 51 |
| 4.2.1.6 | USB Serial/JTAG Controller | 52 |
| 4.2.1.7 | Two-wire Automotive Interface | 52 |
| 4.2.1.8 | SDIO Slave Controller | 53 |
| 4.2.1.9 | LED PWM Controller | 54 |
| 4.2.1.10 | Motor Control PWM | 54 |
| 4.2.1.11 | Remote Control Peripheral | 55 |
| 4.2.1.12 | Parallel IO Controller | 56 |
| 4.2.2 | Analog Signal Processing | 56 |
| 4.2.2.1 | SAR ADC | 56 |
| 4.2.2.2 | Temperature Sensor | 57 |
| 4.3 | Wireless Communication | 58 |
| 4.3.1 | Radio | 58 |

| | | |
|--|--|----|
| 4.3.1.1 | 2.4 GHz Receiver | 58 |
| 4.3.1.2 | 2.4 GHz Transmitter | 58 |
| 4.3.1.3 | Clock Generator | 58 |
| 4.3.2 | Wi-Fi | 59 |
| 4.3.2.1 | Wi-Fi Radio and Baseband | 59 |
| 4.3.2.2 | Wi-Fi MAC | 59 |
| 4.3.2.3 | Networking Features | 61 |
| 4.3.3 | Bluetooth LE | 61 |
| 4.3.3.1 | Bluetooth LE PHY | 61 |
| 4.3.3.2 | Bluetooth LE Link Controller | 61 |
| 4.3.4 | 802.15.4 | 62 |
| 4.3.4.1 | 802.15.4 PHY | 62 |
| 4.3.4.2 | 802.15.4 MAC | 62 |
| 5 | Electrical Characteristics | 63 |
| 5.1 | Absolute Maximum Ratings | 63 |
| 5.2 | Recommended Operating Conditions | 63 |
| 5.3 | VDD_SPI Output Characteristics | 64 |
| 5.4 | DC Characteristics (3.3 V, 25 °C) | 64 |
| 5.5 | ADC Characteristics | 64 |
| 5.6 | Current Consumption Characteristics | 65 |
| 5.6.1 | Current Consumption in Active Mode | 65 |
| 5.6.2 | Current Consumption in Other Modes | 66 |
| 5.7 | Reliability | 67 |
| 6 | RF Characteristics | 68 |
| 6.1 | Wi-Fi Radio | 68 |
| 6.1.1 | Wi-Fi RF Transmitter (TX) Characteristics | 68 |
| 6.1.2 | Wi-Fi RF Receiver (RX) Characteristics | 69 |
| 6.2 | Bluetooth 5 (LE) Radio | 71 |
| 6.2.1 | Bluetooth LE RF Transmitter (TX) Characteristics | 71 |
| 6.2.2 | Bluetooth LE RF Receiver (RX) Characteristics | 73 |
| 6.3 | 802.15.4 Radio | 75 |
| 6.3.1 | 802.15.4 RF Transmitter (TX) Characteristics | 75 |
| 6.3.2 | 802.15.4 RF Receiver (RX) Characteristics | 75 |
| 7 | Packaging | 76 |
| Appendix A – ESP32-C6 Consolidated Pin Overview | | 77 |
| Glossary | | 79 |
| Related Documentation and Resources | | 80 |
| Revision History | | 81 |

List of Tables

| | | |
|------|--|----|
| 1-1 | ESP32-C6 Series Comparison | 12 |
| 2-1 | QFN40 Pin Overview | 15 |
| 2-2 | QFN32 Pin Overview | 16 |
| 2-3 | Peripheral Signals Routed via IO MUX | 18 |
| 2-4 | QFN40 IO MUX Pin Functions | 19 |
| 2-5 | QFN32 IO MUX Pin Functions | 20 |
| 2-6 | LP Peripheral Signals Routed via LP IO MUX | 21 |
| 2-7 | LP IO MUX Functions | 21 |
| 2-8 | Analog Signals Routed to Analog Functions | 22 |
| 2-9 | Analog Functions | 22 |
| 2-10 | QFN40 Peripheral Pin Assignment | 25 |
| 2-11 | QFN32 Peripheral Pin Assignment | 26 |
| 2-12 | Analog Pins | 27 |
| 2-13 | Power Pins | 28 |
| 2-14 | Voltage Regulators | 28 |
| 2-15 | Description of Timing Parameters for Power-up and Reset | 30 |
| 2-16 | Pin Mapping Between QFN40 Chip and Flash | 31 |
| 3-1 | Default Configuration of Strapping Pins | 32 |
| 3-2 | Description of Timing Parameters for the Strapping Pins | 33 |
| 3-3 | Chip Boot Mode Control | 33 |
| 3-4 | SDIO Input Sampling Edge/Output Driving Edge Control | 34 |
| 3-5 | UARTO ROM Message Printing Control | 34 |
| 3-6 | USB Serial/JTAG ROM Message Printing Control | 34 |
| 3-7 | JTAG Signal Source Control | 35 |
| 5-1 | Absolute Maximum Ratings | 63 |
| 5-2 | Recommended Power Characteristics | 63 |
| 5-3 | VDD_SPI Internal and Output Characteristics | 64 |
| 5-4 | DC Characteristics (3.3 V, 25 °C) | 64 |
| 5-5 | ADC Characteristics | 65 |
| 5-6 | ADC Calibration Results | 65 |
| 5-7 | Current Consumption for Wi-Fi (2.4 GHz) in Active Mode | 65 |
| 5-8 | Current Consumption for Bluetooth LE in Active Mode | 66 |
| 5-9 | Current Consumption for 802.15.4 in Active Mode | 66 |
| 5-10 | Current Consumption in Modem-sleep Mode | 66 |
| 5-11 | Current Consumption in Low-Power Modes | 67 |
| 5-12 | Reliability Qualifications | 67 |
| 6-1 | Wi-Fi RF Characteristics | 68 |
| 6-2 | TX Power with Spectral Mask and EVM Meeting 802.11 Standards | 68 |
| 6-3 | TX EVM Test ¹ | 68 |
| 6-4 | RX Sensitivity | 69 |
| 6-5 | Maximum RX Level | 70 |
| 6-6 | RX Adjacent Channel Rejection | 70 |
| 6-7 | Bluetooth LE RF Characteristics | 71 |

| | | |
|------|---|----|
| 6-8 | Bluetooth LE - Transmitter Characteristics - 1 Mbps | 71 |
| 6-9 | Bluetooth LE - Transmitter Characteristics - 2 Mbps | 71 |
| 6-10 | Bluetooth LE - Transmitter Characteristics - 125 Kbps | 72 |
| 6-11 | Bluetooth LE - Transmitter Characteristics - 500 Kbps | 72 |
| 6-12 | Bluetooth LE - Receiver Characteristics - 1 Mbps | 73 |
| 6-13 | Bluetooth LE - Receiver Characteristics - 2 Mbps | 73 |
| 6-14 | Bluetooth LE - Receiver Characteristics - 125 Kbps | 74 |
| 6-15 | Bluetooth LE - Receiver Characteristics - 500 Kbps | 74 |
| 6-16 | 802.15.4 RF Characteristics | 75 |
| 6-17 | 802.15.4 Transmitter Characteristics - 250 Kbps | 75 |
| 6-18 | 802.15.4 Receiver Characteristics - 250 Kbps | 75 |
| 7-1 | QFN40 Pin Overview | 77 |
| 7-2 | QFN32 Pin Overview | 78 |

List of Figures

| | | |
|-----|---|----|
| 1-1 | ESP32-C6 Series Nomenclature | 12 |
| 2-1 | ESP32-C6 Pin Layout (QFN40, Top View) | 13 |
| 2-2 | ESP32-C6 Pin Layout (QFN32, Top View) | 14 |
| 2-3 | ESP32-C6 Power Scheme | 29 |
| 2-4 | Visualization of Timing Parameters for Power-up and Reset | 29 |
| 3-1 | Visualization of Timing Parameters for the Strapping Pins | 33 |
| 4-1 | Address Mapping Structure | 38 |
| 7-1 | QFN40 (5×5 mm) Package | 76 |
| 7-2 | QFN32 (5×5 mm) Package | 76 |

1 ESP32-C6 Series Comparison

1.1 Nomenclature

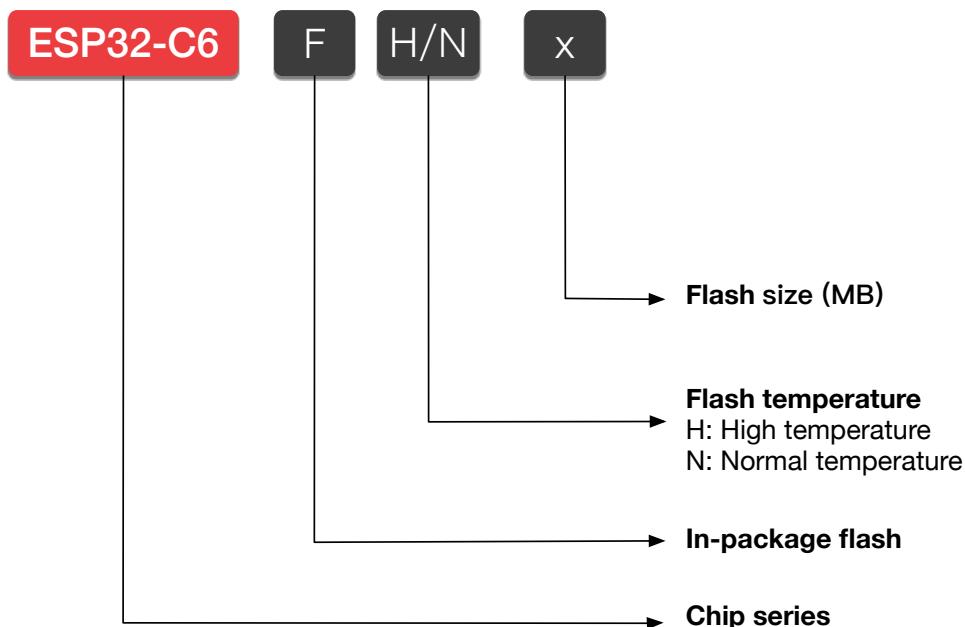


Figure 1-1. ESP32-C6 Series Nomenclature

1.2 Comparison

Table 1-1. ESP32-C6 Series Comparison

| Ordering Code ¹ | In-Package Flash | Ambient Temp. ² | Package |
|----------------------------|---------------------------------|----------------------------|----------------|
| ESP32-C6 | — ³ | −40 ~ 105 °C | QFN40 (5×5 mm) |
| ESP32-C6FH4 | 4 MB (Quad SPI) ^{4, 5} | −40 ~ 105 °C | QFN32 (5×5 mm) |
| ESP32-C6FH8 | 8 MB (Quad SPI) ^{4, 5} | −40 ~ 105 °C | QFN32 (5×5 mm) |

¹ For details on chip marking and packing, see Section [7 Packaging](#).

² Ambient temperature specifies the recommended temperature range of the environment immediately outside an Espressif chip.

³ Can connect a flash outside the chip package. For details, see Section [4.1.2.2 External Memory](#).

⁴ For details about SPI modes, see Section [2.6 Pin Mapping Between Chip and Flash](#).

⁵ For information about in-package flash, see also Section [4.1.2.1 Internal Memory](#). By default, the SPI flash on the chip operates at a maximum clock frequency of 80 MHz and does not support the auto suspend feature. If you have a requirement for a higher flash clock frequency of 120 MHz or if you need the flash auto suspend feature, please contact us.

2 Pins

2.1 Pin Layout

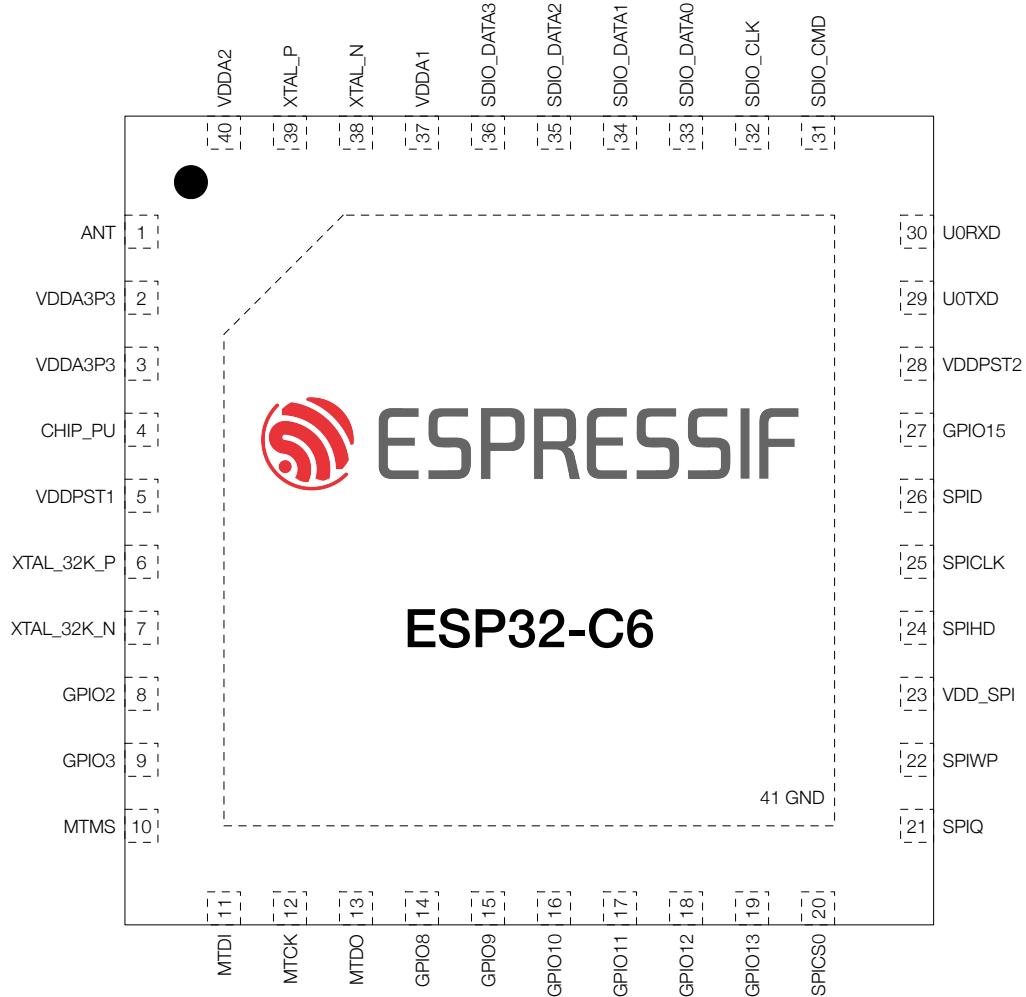


Figure 2-1. ESP32-C6 Pin Layout (QFN40, Top View)

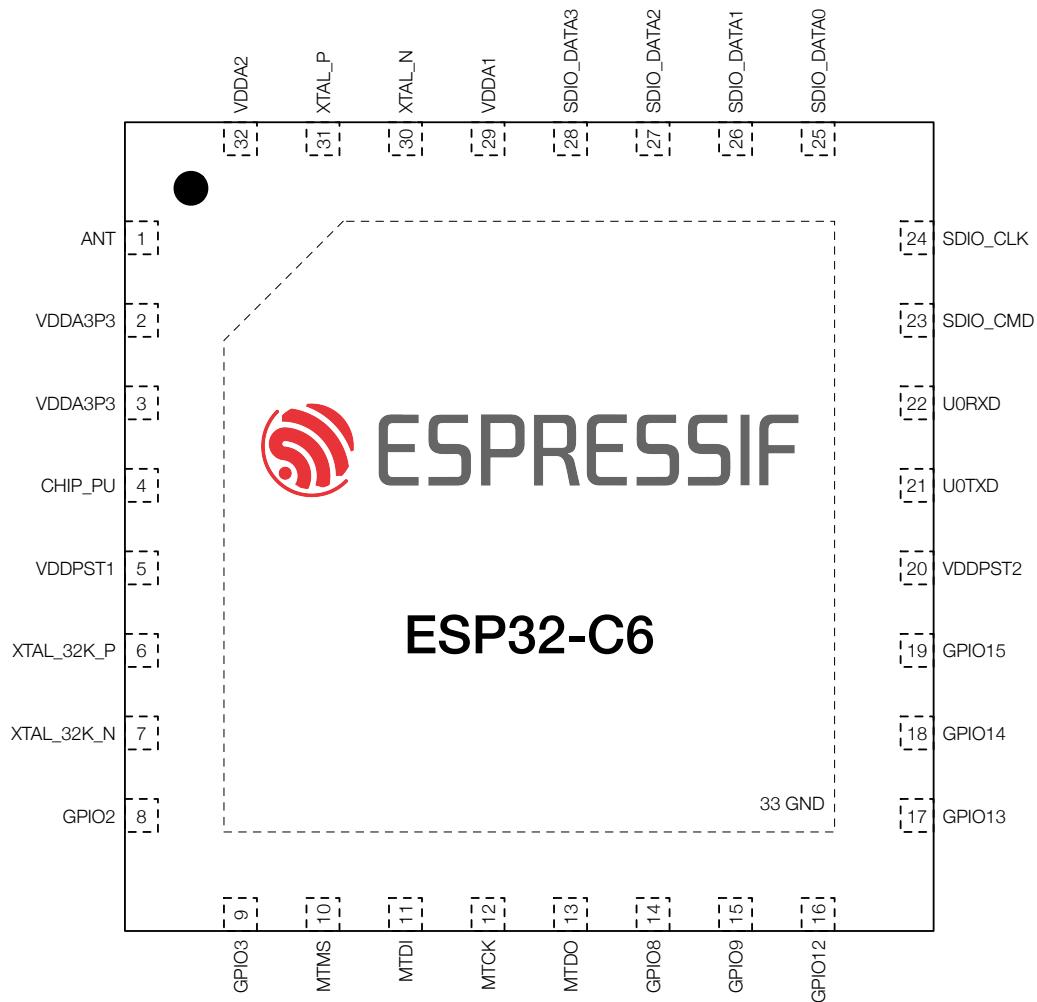


Figure 2-2. ESP32-C6 Pin Layout (QFN32, Top View)

2.2 Pin Overview

The ESP32-C6 chip integrates multiple peripherals that require communication with the outside world. To keep the chip package size reasonably small, the number of available pins has to be limited. So the only way to route all the incoming and outgoing signals is through pin multiplexing. Pin muxing is controlled via software programmable registers (see [ESP32-C6 Technical Reference Manual](#) > Chapter *IO MUX and GPIO Matrix*).

All in all, the ESP32-C6 chip has the following types of pins:

- **IO pins** with the following predefined sets of functions to choose from:
 - Each IO pin has predefined **IO MUX functions** – see Table [2-4 QFN40 IO MUX Pin Functions](#) or Table [2-5 QFN32 IO MUX Pin Functions](#)
 - Some IO pins have predefined **LP IO MUX functions** – see Table [2-7 LP IO MUX Functions](#)
 - Some IO pins have predefined **analog functions** – see Table [2-9 Analog Functions](#)

Predefined functions means that each IO pin has a set of direct connections to certain signals from on-chip peripherals. During run-time, the user can configure which peripheral signal from a predefined set to connect to a certain pin at a certain time via memory mapped registers (see the TRM).

- **Analog pins** that have exclusively-dedicated **analog functions** – see Table [2-12 Analog Pins](#)
- **Power pins** that supply power to the chip components and non-power pins – see Table [2-13 Power Pins](#)

Table [2-1 QFN40 Pin Overview](#) or Table [2-2 QFN32 Pin Overview](#) gives an overview of all the pins. For more information, see the respective sections for each pin type below, or [Appendix A – ESP32-C6 Consolidated Pin Overview](#).

Table 2-1. QFN40 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power ^{2, 3} | Pin Settings ⁴⁻⁶ | | Pin Function Sets ¹ | | |
|---------|------------|----------|-------------------------------------|-----------------------------|----------------------|--------------------------------|-----------|--------|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 1 | ANT | Analog | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | |
| 4 | CHIP_PU | Analog | VDDPST1 | | | | | |
| 5 | VDDPST1 | Power | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | IO MUX | LP IO MUX | Analog |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | IO MUX | LP IO MUX | Analog |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 10 | MTMS | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 11 | MTDI | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU ⁵ | IO MUX | LP IO MUX | Analog |
| 13 | MTDO | IO | VDDPST1 | | IE | IO MUX | LP IO MUX | |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | IO MUX | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | IO MUX | | |
| 16 | GPIO10 | IO | VDDPST2 | | IE | IO MUX | | |
| 17 | GPIO11 | IO | VDDPST2 | | IE | IO MUX | | |

Cont'd on next page

Table 2-1 – cont'd from previous page

| Pin No. | Pin Name | Pin Type | Pin Providing Power ^{2, 3} | Pin Settings ⁴⁻⁶ | | Pin Function Sets ¹ | | |
|---------|------------|----------|-------------------------------------|-----------------------------|------------------|--------------------------------|-----------|--------|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 18 | GPIO12 | IO | VDDPST2 | | IE | IO MUX | | Analog |
| 19 | GPIO13 | IO | VDDPST2 | USB_PU | IE, USB_PU | IO MUX | | Analog |
| 20 | SPICSO | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 21 | SPIQ | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 22 | SPIWP | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 23 | VDD_SPI | Power/IO | — | | | IO MUX | | Analog |
| 24 | SPIHD | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 25 | SPICLK | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 26 | SPIID | IO | VDD_SPI | WPU | IE, WPU | IO MUX | | |
| 27 | GPIO15 | IO | VDDPST2 | IE | IE | IO MUX | | |
| 28 | VDDPST2 | Power | | | | | | |
| 29 | UOTXD | IO | VDDPST2 | | WPU ⁶ | IO MUX | | |
| 30 | UORXD | IO | VDDPST2 | | IE, WPU | IO MUX | | |
| 31 | SDIO_CMD | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 32 | SDIO_CLK | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 33 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 34 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 35 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 36 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 37 | VDDA1 | Power | | | | | | |
| 38 | XTAL_N | Analog | | | | | | |
| 39 | XTAL_P | Analog | | | | | | |
| 40 | VDDA2 | Power | | | | | | |
| 41 | GND | Power | | | | | | |

Table 2-2. QFN32 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power ^{2, 3} | Pin Settings ⁴⁻⁶ | | Pin Function Sets ¹ | | |
|---------|------------|----------|-------------------------------------|-----------------------------|----------------------|--------------------------------|-----------|--------|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 1 | ANT | Analog | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | |
| 4 | CHIP_PU | Analog | VDDPST1 | | | | | |
| 5 | VDDPST1 | Power | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | IO MUX | LP IO MUX | Analog |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | IO MUX | LP IO MUX | Analog |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 10 | MTMS | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 11 | MTDI | IO | VDDPST1 | IE | IE | IO MUX | LP IO MUX | Analog |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU ⁵ | IO MUX | LP IO MUX | Analog |
| 13 | MTDO | IO | VDDPST1 | | IE | IO MUX | LP IO MUX | |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | IO MUX | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | IO MUX | | |

Cont'd on next page

Table 2-2 – cont'd from previous page

| Pin No. | Pin Name | Pin Type | Pin Providing Power ^{2, 3} | Pin Settings ⁴⁻⁶ | | Pin Function Sets ¹ | | |
|---------|------------|----------|-------------------------------------|-----------------------------|------------------|--------------------------------|-----------|--------|
| | | | | At Reset | After Reset | IO MUX | LP IO MUX | Analog |
| 16 | GPIO12 | IO | VDDPST2 | | IE | IO MUX | | Analog |
| 17 | GPIO13 | IO | VDDPST2 | USB_PU | IE, USB_PU | IO MUX | | Analog |
| 18 | GPIO14 | IO | VDDPST2 | | IE | IO MUX | | |
| 19 | GPIO15 | IO | VDDPST2 | IE | IE | IO MUX | | |
| 20 | VDDPST2 | Power | | | | | | |
| 21 | UOTXD | IO | VDDPST2 | | WPU ⁶ | IO MUX | | |
| 22 | UORXD | IO | VDDPST2 | | IE, WPU | IO MUX | | |
| 23 | SDIO_CMD | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 24 | SDIO_CLK | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 25 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 26 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 27 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 28 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | IO MUX | | |
| 29 | VDDA1 | Power | | | | | | |
| 30 | XTAL_N | Analog | | | | | | |
| 31 | XTAL_P | Analog | | | | | | |
| 32 | VDDA2 | Power | | | | | | |
| 33 | GND | Power | | | | | | |

1. **Bold** marks the pin function set in which a pin has its default function in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

2. In column **Pin Providing Power**, regarding pins powered by VDD_SPI:

- Power actually comes from the internal power rail supplying power to VDD_SPI. For details, see Section 2.5.2 *Power Scheme*.

3. Except for GPIO12 and GPIO13 whose default drive strength is 40 mA, the default drive strength for all the other pins is 20 mA.

4. Column **Pin Settings** shows predefined settings at reset and after reset with the following abbreviations:

- IE – input enabled
- WPU – internal weak pull-up resistor enabled
- WPD – internal weak pull-down resistor enabled
- USB_PU – USB pull-up resistor enabled
 - By default, the USB function is enabled for USB pins (i.e., GPIO12 and GPIO13), and the pin pull-up is decided by the USB pull-up resistor. The USB pull-up resistor is controlled by `USB_SERIAL_JTAG_DP/DM_PULLUP` and the pull-up value is controlled by `USB_SERIAL_JTAG_PULLUP_VALUE`. For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *USB Serial/JTAG Controller*.
 - When the USB function is disabled, USB pins are used as regular GPIOs. At reset, GPIO13's internal weak pull-up resistor is disabled by default. After reset, GPIO13's internal weak pull-up resistor is enabled by default. A pin's internal weak pull-up and pull-down resistors are configurable by `IO_MUX_FUN_WPU/WPD`.

5. Depends on the value of `EFUSE_DIS_PAD_JTAG`

- 0 - default value. Input enabled, and internal weak pull-up resistor enabled (IE & WPU)
- 1 - input enabled (IE)

6. Output enabled

2.3 IO Pins

2.3.1 IO MUX Pin Functions

The IO MUX allows multiple input/output signals to be connected to a single input/output pin. Each IO pin of ESP32-C6 can be connected to one of the three signals (IO MUX functions, i.e. F0-F2), as listed in Table 2-4 [QFN40 IO MUX Pin Functions](#) and Table 2-5 [QFN32 IO MUX Pin Functions](#).

Among the three sets of signals:

- Some are routed via the GPIO Matrix ([GPIO0, GPIO1, etc.](#)), which incorporates internal signal routing circuitry for mapping signals programmatically. It gives the pin access to almost any peripheral signals. However, the flexibility of programmatic mapping comes at a cost as it might affect the latency of routed signals. For details about connecting to peripheral signals via GPIO Matrix, see [ESP32-C6 Technical Reference Manual](#) > Chapter *IO MUX and GPIO Matrix*.
- Some are directly routed from certain peripherals ([UOTXD, MTCK, etc.](#)), including UART0, JTAG, SPI0/1, SPI2, and SDIO - see Table 2-3 [Peripheral Signals Routed via IO MUX](#).

Table 2-3. Peripheral Signals Routed via IO MUX

| Pin Function | Signal | Description |
|--|--|---|
| UOTXD UORXD | Transmit data Receive data | UART0 interface |
| MTCK MTDO MTDI MTMS | Test clock Test Data Out Test Data In Test Mode Select | JTAG interface for debugging |
| SPIQ SPID SPIHD SPIWP SPICLK SPICSO | Data out Data in Hold Write protect Clock Chip select | 3.3 V SPI0/1 interface for connection to in-package or off-package flash via the SPI bus. It supports 1-, 2-, 4-line SPI modes. See also Section 2.6 Pin Mapping Between Chip and Flash |
| FSPIQ FSPID FSPIHD FSPIWP FSPICLK FSPICS... | Data out Data in Hold Write protect Clock Chip select | SPI2 interface for fast SPI connection. It supports 1-, 2-, 4-line SPI modes |
| SDIO_CMD SDIO_CLK SDIO_DATA... | Command Clock Data | SDIO interface |

Table 2-4 [QFN40 IO MUX Pin Functions](#) or Table 2-5 [QFN32 IO MUX Pin Functions](#) shows the IO MUX functions of IO pins.

Table 2-4. QFN40 IO MUX Pin Functions

| Pin No. | IO MUX / GPIO Name ² | IO MUX Function ^{1, 2, 3} | | | | | |
|---------|---------------------------------|------------------------------------|-------------------|--------|-------|---------|--------|
| | | F0 | Type ³ | F1 | Type | F2 | Type |
| 6 | GPIO0 | GPIO0 | I/O/T | GPIO0 | I/O/T | | |
| 7 | GPIO1 | GPIO1 | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | GPIO2 | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | GPIO3 | I/O/T | GPIO3 | I/O/T | | |
| 10 | GPIO4 | MTMS | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | GPIO5 | MTDI | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | GPIO6 | MTCK | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | GPIO7 | MTDO | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | GPIO8 | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | GPIO9 | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO10 | GPIO10 | I/O/T | GPIO10 | I/O/T | | |
| 17 | GPIO11 | GPIO11 | I/O/T | GPIO11 | I/O/T | | |
| 18 | GPIO12 | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 19 | GPIO13 | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 20 | GPIO24 | SPICSO | O/T | GPIO24 | I/O/T | | |
| 21 | GPIO25 | SPIQ | I1/O/T | GPIO25 | I/O/T | | |
| 22 | GPIO26 | SPIWP | I1/O/T | GPIO26 | I/O/T | | |
| 23 | GPIO27 | GPIO27 | I/O/T | GPIO27 | I/O/T | | |
| 24 | GPIO28 | SPIHD | I1/O/T | GPIO28 | I/O/T | | |
| 25 | GPIO29 | SPICLK | O/T | GPIO29 | I/O/T | | |
| 26 | GPIO30 | SPID | I1/O/T | GPIO30 | I/O/T | | |
| 27 | GPIO15 | GPIO15 | I/O/T | GPIO15 | I/O/T | | |
| 29 | GPIO16 | UOTXD | O | GPIO16 | I/O/T | FSPICSO | I1/O/T |
| 30 | GPIO17 | UORXD | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 31 | GPIO18 | SDIO_CMD | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 32 | GPIO19 | SDIO_CLK | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 33 | GPIO20 | SDIO_DATA0 | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 34 | GPIO21 | SDIO_DATA1 | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 35 | GPIO22 | SDIO_DATA2 | I1/O/T | GPIO22 | I/O/T | | |
| 36 | GPIO23 | SDIO_DATA3 | I1/O/T | GPIO23 | I/O/T | | |

¹ **Bold** marks the default pin functions in the default boot mode. See Section 3.1 *Chip Boot Mode Control*.

² Regarding **highlighted** cells, see Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*.

³ Each IO MUX function (F_n , $n = 0 \sim 2$) is associated with a type. The description of type is as follows:

- I – input. O – output. T – high impedance.
- I1 – input; if the pin is assigned a function other than F_n , the input signal of F_n is always 1.
- IO – input; if the pin is assigned a function other than F_n , the input signal of F_n is always 0.

Table 2-5. QFN32 IO MUX Pin Functions

| Pin No. | IO MUX / GPIO Name ² | IO MUX Function ^{1, 2, 3} | | | | | |
|---------|---------------------------------|------------------------------------|-------------------|--------|-------|---------|--------|
| | | F0 | Type ³ | F1 | Type | F2 | Type |
| 6 | GPIO0 | GPIO0 | I/O/T | GPIO0 | I/O/T | | |
| 7 | GPIO1 | GPIO1 | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | GPIO2 | I/O/T | GPIO2 | I/O/T | FSPIQ | I1/O/T |
| 9 | GPIO3 | GPIO3 | I/O/T | GPIO3 | I/O/T | | |
| 10 | GPIO4 | MTMS | I1 | GPIO4 | I/O/T | FSPIHD | I1/O/T |
| 11 | GPIO5 | MTDI | I1 | GPIO5 | I/O/T | FSPIWP | I1/O/T |
| 12 | GPIO6 | MTCK | I1 | GPIO6 | I/O/T | FSPICLK | I1/O/T |
| 13 | GPIO7 | MTDO | O/T | GPIO7 | I/O/T | FSPID | I1/O/T |
| 14 | GPIO8 | GPIO8 | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | GPIO9 | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO12 | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 17 | GPIO13 | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 18 | GPIO14 | GPIO14 | I/O/T | GPIO14 | I/O/T | | |
| 19 | GPIO15 | GPIO15 | I/O/T | GPIO15 | I/O/T | | |
| 21 | GPIO16 | UOTXD | O | GPIO16 | I/O/T | FSPICSO | I1/O/T |
| 22 | GPIO17 | UORXD | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 23 | GPIO18 | SDIO_CMD | I1/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 24 | GPIO19 | SDIO_CLK | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 25 | GPIO20 | SDIO_DATA0 | I1/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 26 | GPIO21 | SDIO_DATA1 | I1/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 27 | GPIO22 | SDIO_DATA2 | I1/O/T | GPIO22 | I/O/T | | |
| 28 | GPIO23 | SDIO_DATA3 | I1/O/T | GPIO23 | I/O/T | | |

¹ **Bold** marks the default pin functions in the default boot mode. See Section [3.1 Chip Boot Mode Control](#).

² Regarding **highlighted** cells, see Section [2.3.4 Restrictions for GPIOs and LP GPIOs](#).

³ Each IO MUX function (F_n , $n = 0 \sim 2$) is associated with a type. The description of type is as follows:

- I – input. O – output. T – high impedance.
- I1 – input; if the pin is assigned a function other than F_n , the input signal of F_n is always 1.
- IO – input; if the pin is assigned a function other than F_n , the input signal of F_n is always 0.

2.3.2 LP IO MUX Functions

When the chip is in Deep-sleep mode, the IO MUX described in Section [2.3.1 IO MUX Pin Functions](#) will not work. That is where the LP IO MUX comes in. It allows multiple input/output signals to be a single input/output pin in Deep-sleep mode, as the pin is connected to the LP system and powered by VDDPST1.

LP IO pins can be assigned to **LP functions**. They can

- Either work as LP GPIOs (**LP_GPIO0**, **LP_GPIO1**, etc.), connected to the LP CPU
- Or connect to LP peripheral signals (**LP_I2C_SDA**, **LP_I2C_SCL**, etc.) - see Table [2-6 LP Peripheral Signals Routed via LP IO MUX](#)

Table 2-6. LP Peripheral Signals Routed via LP IO MUX

| Pin Function | Signal | Description |
|--------------|---------------------|-------------------|
| LP_I2C_SDA | Serial data | |
| LP_I2C_SCL | Serial clock | LP I2C interface |
| LP_UART_RXD | Receive | |
| LP_UART_TXD | Transmit | |
| LP_UART_RTSN | Request to send | |
| LP_UART_CTSN | Clear to send | LP UART interface |
| LP_UART_DTRN | Data set ready | |
| LP_UART_DSRN | Data terminal ready | |

Table [2-7 LP IO MUX Functions](#) shows the LP functions of LP IO pins.

Table 2-7. LP IO MUX Functions

| Pin No. | LP IO Name ^{1, 2, 3} | LP IO MUX Function | |
|---------|-------------------------------|--------------------|--------------|
| | | F0 | F1 |
| 6 | LP_GPIO0 | LP_GPIO0 | LP_UART_DTRN |
| 7 | LP_GPIO1 | LP_GPIO1 | LP_UART_DSRN |
| 8 | LP_GPIO2 | LP_GPIO2 | LP_UART_RTSN |
| 9 | LP_GPIO3 | LP_GPIO3 | LP_UART_CTSN |
| 10 | LP_GPIO4 | LP_GPIO4 | LP_UART_RXD |
| 11 | LP_GPIO5 | LP_GPIO5 | LP_UART_TXD |
| 12 | LP_GPIO6 | LP_GPIO6 | LP_I2C_SDA |
| 13 | LP_GPIO7 | LP_GPIO7 | LP_I2C_SCL |

¹ **Bold** marks the default pin functions in the default boot mode. See Section [3.1 Chip Boot Mode Control](#).

² This column lists the LP GPIO names, since LP functions are configured with LP GPIO registers that use LP GPIO numbering.

³ Regarding **highlighted** cells, see Section [2.3.4 Restrictions for GPIOs and LP GPIOs](#).

2.3.3 Analog Functions

Some IO pins also have **analog functions**, for analog peripherals (such as ADC) in any power mode. Internal analog signals are routed to these analog functions, see Table [2-8 Analog Signals Routed to Analog Functions](#).

Table 2-8. Analog Signals Routed to Analog Functions

| Pin Function | Signal | Description |
|------------------|-------------------------|---|
| ADC1_CH... | ADC1 channel ... signal | ADC1 interface |
| XTAL_32K_N | Negative clock signal | 32 kHz external clock input/output |
| XTAL_32K_P | Positive clock signal | connected to ESP32-C6's crystal or oscillator |
| | | |
| USB_D- USB_D+ | Data - Data + | USB Serial/JTAG function |

[Table 2-9 Analog Functions](#) shows the analog functions of IO pins.

Table 2-9. Analog Functions

| QFN40 Pin No. | QFN32 Pin No. | Analog IO Name ^{1, 2} | Analog Function ² | |
|------------------|------------------|-----------------------------------|------------------------------|----------|
| | | | F0 | F1 |
| 6 | 6 | GPIO0 | XTAL_32K_P | ADC1_CH0 |
| 7 | 7 | GPIO1 | XTAL_32K_N | ADC1_CH1 |
| 8 | 8 | GPIO2 | | ADC1_CH2 |
| 9 | 9 | GPIO3 | | ADC1_CH3 |
| 10 | 10 | GPIO4 | | ADC1_CH4 |
| 11 | 11 | GPIO5 | | ADC1_CH5 |
| 12 | 12 | GPIO6 | | ADC1_CH6 |
| 18 | 16 | GPIO12 | USB_D- | |
| 19 | 17 | GPIO13 | USB_D+ | |
| 23 | — | GPIO27 | VDD_SPI | |

¹ **Bold** marks the default pin functions in the default boot mode.

See Section [3.1 Chip Boot Mode Control](#).

² Regarding **highlighted** cells, see Section [2.3.4 Restrictions for GPIOs and LP GPIOs](#).

2.3.4 Restrictions for GPIOs and LP GPIOs

All IO pins of ESP32-C6 have GPIO and some have LP GPIO pin functions. However, the IO pins are multiplexed and can be configured for different purposes based on the requirements. Some IOs have restrictions for usage. It is essential to consider the multiplexed nature and the limitations when using these IO pins.

In tables of this section, some pin functions are highlighted in red or yellow. They are important pin functions, and the IO pins with these functions should be used with caution as GPIO / GPIO :

- **IO Pins** – allocated for communication with flash and NOT recommended for other uses. For details, see Section [2.6 Pin Mapping Between Chip and Flash](#).
- **IO Pins** – have one of the following important functions:
 - **Strapping pins** – need to be at certain logic levels at startup. See Section [3 Boot Configurations](#).

Note:
Strapping pins are highlighted by pin name, instead of pin functions.
 - **USB_D+/-** – by default, connected to the USB Serial/JTAG Controller. To function as GPIOs, these pins need to be reconfigured.
 - **JTAG interface** – often used for debugging. See Table [2-4 QFN40 IO MUX Pin Functions](#) or Table [2-5 QFN32 IO MUX Pin Functions](#). To free these pins up, the pin functions USB_D+/- of the USB Serial/JTAG Controller can be used instead. See also Section [3.4 JTAG Signal Source Control](#).
 - **UART0 interface** – often used for debugging. See Table [2-4 QFN40 IO MUX Pin Functions](#) or Table [2-5 QFN32 IO MUX Pin Functions](#).
 - **VDD_SPI** – the power supply pin for off-package flash by default, and can only be used as a GPIO pin if the flash is powered by an external power supply.

For more information about assigning pins, please see Section [2.3.5 Peripheral Pin Assignment](#) and Appendix A – ESP32-C6 Consolidated Pin Overview.

2.3.5 Peripheral Pin Assignment

Table 2-10 *QFN40 Peripheral Pin Assignment* and Table 2-11 *QFN32 Peripheral Pin Assignment* highlight which pins can be assigned to each peripheral interface according to the following priorities:

- Priority 1 (P1) : Fixed pins connected directly to peripheral signals via IO MUX or RTC IO MUX.
- Any GPIO pins mapping to peripheral signals via GPIO Matrix, can be priority 2, 3, or 4:
 - Priority 2 (P2) : GPIO pins can be freely used without restrictions.
 - Priority 3 (P3) : GPIO pins should be used with caution, as they have one of the following important functions, as described in Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*:
 - * GPIO4, GPIO5, GPIO8, GPIO9, GPIO15 : Strapping pins.
 - * GPIO12, GPIO13 : USB Serial/JTAG interface.
 - * GPIO4, GPIO5, GPIO6, GPIO7 : JTAG interface.
 - * GPIO16, GPIO17 : UARTO interface.
 - * GPIO27 : The VDD_SPI pin. The power supply pin for off-package flash by default, and can only be reconfigured as a GPIO pin if the flash is powered by an external power supply.
 - Priority 4 (P4) : GPIO pins already allocated or not recommended for use, as described in Section 2.3.4 *Restrictions for GPIOs and LP GPIOs*:
 - * GPIO24, GPIO25, GPIO26, GPIO28, GPIO29, GPIO30 : SPI0/1 interface recommended for the off-package flash.

If a peripheral interface does not have priority 1 pins (e.g., UART1), it can be assigned to any GPIO pins from priority 2 to priority 4.

If a peripheral interface does not have priority 2 to 4 pins (e.g., USB Serial/JTAG), it means it can be assigned only to priority 1 pins.

Note:

- For details about which peripheral signals are connected to IO MUX or LP IO MUX pins, please refer to Section 2.3.1 *IO MUX Pin Functions* or Section 2-7 *LP IO MUX Functions*.
- For details about which peripheral signals can be assigned to GPIO pins, please refer to [ESP32-C6 Technical Reference Manual](#) > Chapter IO MUX and GPIO Matrix > Section Peripheral Signal List.

Table 2-10. QFN40 Peripheral Pin Assignment

| Pin No. | Pin Name | USB Serial/JTAG | JTAG | SDIO Slave | LP UART | LP I2C | ADC | UART0 ¹ | SPI0/1 ¹ | SPI2 ¹ | UART1 | I2C | I2S | PCNT | TWAI | LED PWM | MCPWM | RMT | PARLIO | |
|---------|------------|-----------------|------|-----------------|-------------------|---------------|---------------|--------------------|---------------------|-------------------|-------------|-------------|---------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1 | ANT | | | | | | | | | | | | | | | | | | | |
| 2 | VDDA3P3 | | | | | | | | | | | | | | | | | | | |
| 3 | VDDA3P3 | | | | | | | | | | | | | | | | | | | |
| 4 | CHIP_PU | | | | | | | | | | | | | | | | | | | |
| 5 | VDDPST1 | | | | | | | | | | | | | | | | | | | |
| 6 | XTAL_32K_P | | | | LP_UART_DTRN (P1) | | ADC1_CH0 (P1) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | |
| 7 | XTAL_32K_N | | | | LP_UART_DSRN (P1) | | ADC1_CH1 (P1) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | |
| 8 | GPIO2 | | | | LP_UART_RTSN (P1) | | ADC1_CH2 (P1) | GPIO2 (P2) | GPIO2 (P2) | FPIQ (P1) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | |
| 9 | GPIO3 | | | | LP_UART_CTSN (P1) | | ADC1_CH3 (P1) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | |
| 10 | MTMS | MTMS (P1) | | | LP_UART_RXD (P1) | | ADC1_CH4 (P1) | GPIO4 (P3) | GPIO4 (P3) | FPIHD (P1) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | |
| 11 | MTDI | MTDI (P1) | | | LP_UART_TXD (P1) | | ADC1_CH5 (P1) | GPIO5 (P3) | GPIO5 (P3) | FPIWP (P1) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | |
| 12 | MTCK | MTCK (P1) | | | LP_I2C_SDA (P1) | ADC1_CH6 (P1) | GPIO6 (P3) | GPIO6 (P3) | FPICLK (P1) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | |
| 13 | MTDO | MTDO (P1) | | | LP_I2C_SCL (P1) | | GPIO7 (P3) | GPIO7 (P3) | FSPID (P1) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | |
| 14 | GPIO8 | | | | | | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | | |
| 15 | GPIO9 | | | | | | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | | |
| 16 | GPIO10 | | | | | | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | GPIO10 (P2) | | |
| 17 | GPIO11 | | | | | | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | GPIO11 (P2) | | |
| 18 | GPIO12 | USB_D- (P1) | | | | | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | | |
| 19 | GPIO13 | USB_D+ (P1) | | | | | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | | |
| 20 | SPICS0 | | | | | | GPIO24 (P4) | SPICS0 (P1) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | GPIO24 (P4) | | |
| 21 | SPIQ | | | | | | GPIO25 (P4) | SPIQ (P1) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | GPIO25 (P4) | | |
| 22 | SPIWP | | | | | | GPIO26 (P4) | SPIWP (P1) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | GPIO26 (P4) | | |
| 23 | VDD_SPI | | | | | | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | GPIO27 (P3) | | |
| 24 | SPIHD | | | | | | GPIO28 (P4) | SPIHD (P1) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | GPIO28 (P4) | | |
| 25 | SPICLK | | | | | | GPIO29 (P4) | SPICLK (P1) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | GPIO29 (P4) | | |
| 26 | SPID | | | | | | GPIO30 (P4) | SPID (P1) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | GPIO30 (P4) | | |
| 27 | GPIO15 | | | | | | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | | |
| 28 | VDDPST2 | | | | | | | | | | | | | | | | | | | |
| 29 | UOTXD | | | | | | UOTXD (P1) | GPIO16 (P3) | FSPICSO (P1) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | | |
| 30 | UORXD | | | | | | UORXD (P1) | GPIO17 (P3) | FSPICSI1 (P1) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | | |
| 31 | SDIO_CMD | | | SDIO_CMD (P1) | | | | | | | GPIO18 (P2) | GPIO18 (P2) | FSPICSI2 (P1) | GPIO18 (P2) |
| 32 | SDIO_CLK | | | SDIO_CLK (P1) | | | | | | | GPIO19 (P2) | GPIO19 (P2) | FSPICSI3 (P1) | GPIO19 (P2) |
| 33 | SDIO_DATA0 | | | SDIO_DATA0 (P1) | | | | | | | GPIO20 (P2) | GPIO20 (P2) | FSPICSI4 (P1) | GPIO20 (P2) |
| 34 | SDIO_DATA1 | | | SDIO_DATA1 (P1) | | | | | | | GPIO21 (P2) | GPIO21 (P2) | FSPICSI5 (P1) | GPIO21 (P2) |
| 35 | SDIO_DATA2 | | | SDIO_DATA2 (P1) | | | | | | | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) |
| 36 | SDIO_DATA3 | | | SDIO_DATA3 (P1) | | | | | | | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) |
| 37 | VDDA1 | | | | | | | | | | | | | | | | | | | |
| 38 | XTAL_N | | | | | | | | | | | | | | | | | | | |
| 39 | XTAL_P | | | | | | | | | | | | | | | | | | | |
| 40 | VDDA2 | | | | | | | | | | | | | | | | | | | |
| 41 | GND | | | | | | | | | | | | | | | | | | | |

¹ Signals of UART0, SPI0/1, and SPI2 interfaces can be mapped to any GPIO pins through the GPIO Matrix, regardless of whether they are directly routed to fixed pins via IO MUX.

Table 2-11. QFN32 Peripheral Pin Assignment

| Pin No. | Pin Name | USB Serial/JTAG | JTAG | SDIO Slave | LP UART | LP I2C | ADC | UART0 ¹ | SPI2 ¹ | UART1 | I2C | I2S | PCNT | TWAI | LED PWM | MCPWM | RMT | PARLIO | |
|---------|------------|-----------------|------|-----------------|-------------------|-----------------|---------------|--------------------|-------------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1 | ANT | | | | | | | | | | | | | | | | | | |
| 2 | VDDA3P3 | | | | | | | | | | | | | | | | | | |
| 3 | VDDA3P3 | | | | | | | | | | | | | | | | | | |
| 4 | CHIP_PU | | | | | | | | | | | | | | | | | | |
| 5 | VDDPST1 | | | | | | | | | | | | | | | | | | |
| 6 | XTAL_32K_P | | | | LP_UART_DTRN (P1) | | ADC1_CH0 (P1) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | GPIO0 (P2) | |
| 7 | XTAL_32K_N | | | | LP_UART_DSRN (P1) | | ADC1_CH1 (P1) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | GPIO1 (P2) | |
| 8 | GPIO2 | | | | LP_UART_RTSN (P1) | | ADC1_CH2 (P1) | GPIO2 (P2) | FSPIQ (P1) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | GPIO2 (P2) | |
| 9 | GPIO3 | | | | LP_UART_CTSN (P1) | | ADC1_CH3 (P1) | GPIO3 (P2) | GPIO1 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | GPIO3 (P2) | |
| 10 | MTMS | | | MTMS (P1) | LP_UART_RXD (P1) | | ADC1_CH4 (P1) | GPIO4 (P3) | FSPID (P1) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | GPIO4 (P3) | |
| 11 | MTDI | | | MTDI (P1) | LP_UART_TXD (P1) | | ADC1_CH5 (P1) | GPIO5 (P3) | FSPIDWP (P1) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | GPIO5 (P3) | |
| 12 | MTCK | | | MTCK (P1) | | LP_I2C_SDA (P1) | ADC1_CH6 (P1) | GPIO6 (P3) | FSPICLK (P1) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | GPIO6 (P3) | |
| 13 | MTDO | | | MTDO (P1) | | LP_I2C_SCL (P1) | | | GPIO7 (P3) | FSPID (P1) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | GPIO7 (P3) | |
| 14 | GPIO8 | | | | | | | | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | GPIO8 (P3) | |
| 15 | GPIO9 | | | | | | | | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | GPIO9 (P3) | |
| 16 | GPIO12 | USB_D- (P1) | | | | | | | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | GPIO12 (P3) | |
| 17 | GPIO13 | USB_D+ (P1) | | | | | | | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | GPIO13 (P3) | |
| 18 | GPIO14 | | | | | | | | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | GPIO14 (P2) | |
| 19 | GPIO15 | | | | | | | | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | GPIO15 (P3) | |
| 20 | VDDPST2 | | | | | | | | | | | | | | | | | | |
| 21 | UOTXD | | | | | | | | UOTXD (P1) | FSPICSO (P1) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | GPIO16 (P3) | |
| 22 | UORXD | | | | | | | | UORXD (P1) | FSPICSI (P1) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | GPIO17 (P3) | |
| 23 | SDIO_CMD | | | SDIO_CMD (P1) | | | | | | GPIO18 (P2) | FSPICS2 (P1) | GPIO18 (P2) |
| 24 | SDIO_CLK | | | SDIO_CLK (P1) | | | | | | GPIO19 (P2) | FSPICS3 (P1) | GPIO19 (P2) |
| 25 | SDIO_DATA0 | | | SDIO_DATA0 (P1) | | | | | | GPIO20 (P2) | FSPICS4 (P1) | GPIO20 (P2) |
| 26 | SDIO_DATA1 | | | SDIO_DATA1 (P1) | | | | | | GPIO21 (P2) | FSPICS5 (P1) | GPIO21 (P2) |
| 27 | SDIO_DATA2 | | | SDIO_DATA2 (P1) | | | | | | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) | GPIO22 (P2) |
| 28 | SDIO_DATA3 | | | SDIO_DATA3 (P1) | | | | | | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) | GPIO23 (P2) |
| 29 | VDDA1 | | | | | | | | | | | | | | | | | | |
| 30 | XTAL_N | | | | | | | | | | | | | | | | | | |
| 31 | XTAL_P | | | | | | | | | | | | | | | | | | |
| 32 | VDDA2 | | | | | | | | | | | | | | | | | | |
| 33 | GND | | | | | | | | | | | | | | | | | | |

¹ Signals of UART0 and SPI2 interfaces can be mapped to any GPIO pins through the GPIO Matrix, regardless of whether they are directly routed to fixed pins via IO MUX.² SPI0/1 interface connected to the in-package flash is not available on QFN32 chips.

2.4 Analog Pins

Table 2-12. Analog Pins

| QFN40 Pin No. | QFN32 Pin No. | Pin Name | Pin Type | Pin Function |
|------------------|------------------|-------------|-------------|---|
| 1 | 1 | ANT | I/O | RF input and output |
| 4 | 4 | CHIP_PU | — | High: on, enables the chip (powered up). Low: off, disables the chip (powered down). Note: Do not leave the CHIP_PU pin floating. |
| 38 | 30 | XTAL_N | — | External clock input/output connected to chip's crystal or oscillator. |
| 39 | 31 | XTAL_P | — | P/N means differential clock positive/negative. |

2.5 Power Supply

2.5.1 Power Pins

The chip is powered via the power pins described in Table 2-13 *Power Pins*.

Table 2-13. Power Pins

| QFN40 Pin No. | QFN32 Pin No. | Pin Name | Direction | Power Supply ^{1,2} Power Domain / Other | IO Pins ⁴ |
|---------------|---------------|----------------------|-----------|---|----------------------|
| 2 | 2 | VDDA3P3 | Input | Analog power domain | |
| 3 | 3 | VDDA3P3 | Input | Analog power domain | |
| 5 | 5 | VDDPST1 | Input | LP digital and part of analog pin power domains | LP IO ³ |
| 23 | — | VDD_SPI ⁴ | Input | In-package flash (backup power line) | |
| | | | Output | In-package flash and off-package flash | |
| 28 | 20 | VDDPST2 | Input | HP digital power domain | HP IO |
| 37 | 29 | VDDA1 | Input | Analog power domain | |
| 40 | 32 | VDDA2 | Input | Analog power domain | |
| 41 | 33 | GND | — | External ground connection | |

¹ See in conjunction with Section 2.5.2 *Power Scheme*.

² For recommended and maximum voltage and current, see Section 5.1 *Absolute Maximum Ratings* and Section 5.2 *Recommended Operating Conditions*.

³ LP IO pins are those powered by VDDPST1 and so on, as shown in Figure 2-3 *ESP32-C6 Power Scheme*. See also Table 2-1 *QFN40 Pin Overview* or Table 2-2 *QFN32 Pin Overview* > Column *Pin Providing Power*.

⁴ To configure VDD_SPI as input or output, see *ESP32-C6 Technical Reference Manual* > Chapter *Low-power Management*.

2.5.2 Power Scheme

The power scheme is shown in Figure 2-3 *ESP32-C6 Power Scheme*.

The components on the chip are powered via voltage regulators.

Table 2-14. Voltage Regulators

| Voltage Regulator | Output | Power Supply |
|-------------------|--------|-----------------|
| HP | 1.1 V | HP power domain |
| LP | 1.1 V | LP power domain |

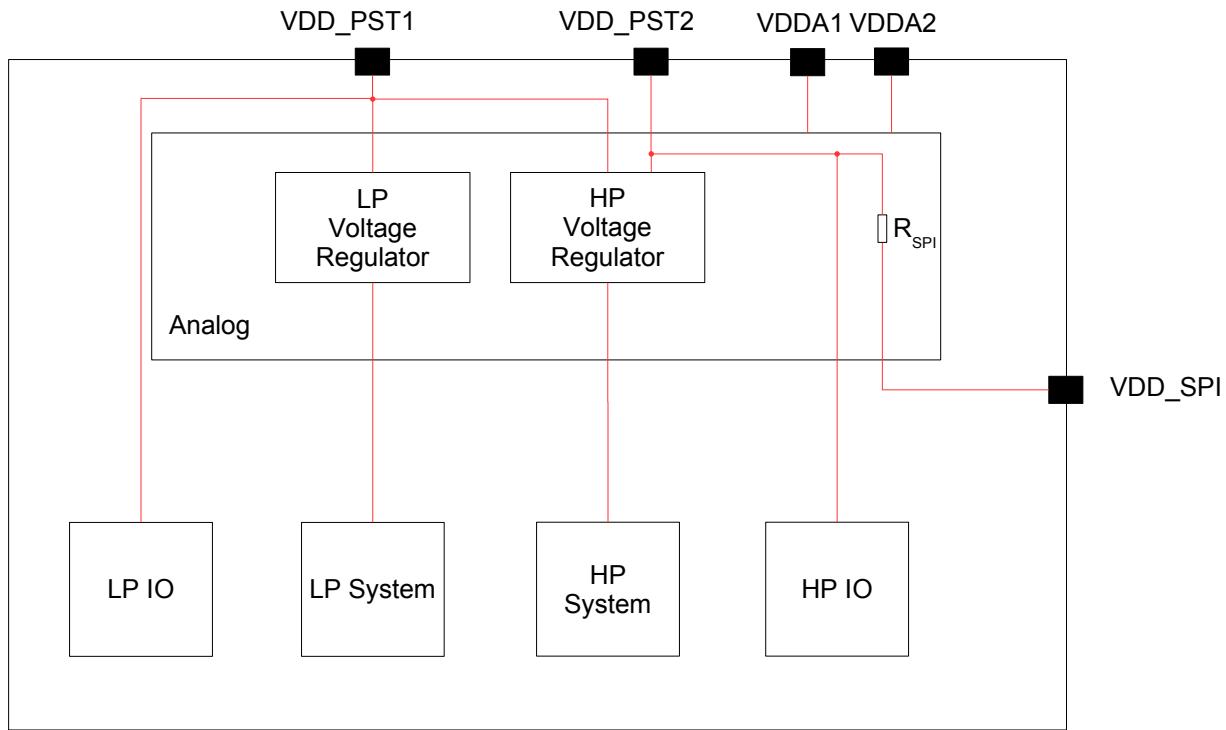


Figure 2-3. ESP32-C6 Power Scheme

2.5.3 Chip Power-up and Reset

Once the power is supplied to the chip, its power rails need a short time to stabilize. After that, CHIP_PU – the pin used for power-up and reset – is pulled high to activate the chip. For information on CHIP_PU as well as power-up and reset timing, see Figure 2-4 and Table 2-15.

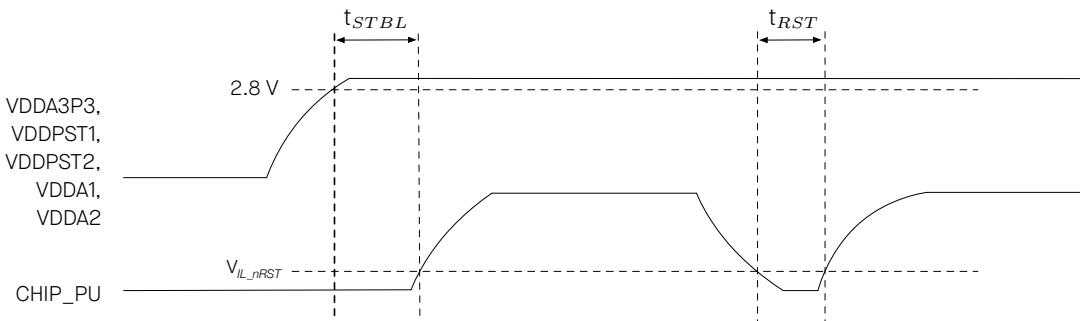


Figure 2-4. Visualization of Timing Parameters for Power-up and Reset

Table 2-15. Description of Timing Parameters for Power-up and Reset

| Parameter | Description | Min (μ s) |
|------------|--|----------------|
| t_{STBL} | Time reserved for the power rails of VDDA3P3, VDDPST1, VD-DPST2, VDDA1 and VDDA2 to stabilize before the CHIP_PU pin is pulled high to activate the chip | 50 |
| t_{RST} | Time reserved for CHIP_PU to stay below V_{IL_nRST} to reset the chip (see Table 5-4) | 50 |

2.6 Pin Mapping Between Chip and Flash

Table 2-16 lists the pin mapping between the chip and off-package flash for all SPI modes.

For chip variants with [in-package flash](#) (namely variants in QFN32 package, see [Table 1-1 ESP32-C6 Series Comparison](#)), the pins allocated for communication with in-package flash are not routed out, but you can take Table 2-16 as a reference.

For more information on SPI controllers, see also Section [4.2.1.2 SPI Controller](#).

Notice:

It is not recommended to use the pins connected to flash for any other purposes.

Table 2-16. Pin Mapping Between QFN40 Chip and Flash

| QFN40 Pin No. | Pin Name | Single SPI Flash | Dual SPI Flash | Quad SPI / QPI Flash |
|------------------|----------|---------------------|-------------------|-------------------------|
| 25 | SPICLK | CLK | CLK | CLK |
| 20 | SPICSO | CS# | CS# | CS# |
| 26 | SPIID | MOSI | SIO0 | SIO0 |
| 21 | SPIQ | MISO | SIO1 | SIO1 |
| 22 | SPIWP | WP# | | SIO2 |
| 24 | SPIHD | HOLD# | | SIO3 |

¹ SIO: Serial Data Input and Output

3 Boot Configurations

The chip allows for configuring the following boot parameters through [strapping pins](#) and [eFuse parameters](#) at power-up or a hardware reset, without microcontroller interaction.

- **Chip boot mode**
 - Strapping pin: GPIO8 and GPIO9
- **SDIO Sampling and Driving Clock Edge**
 - Strapping pin: MTMS and MTDI
- **ROM message printing**
 - Strapping pin: GPIO8
 - eFuse parameter: EFUSE_UART_PRINT_CONTROL and EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT
- **JTAG signal source**
 - Strapping pin: GPIO15
 - eFuse parameter: EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG, and EFUSE_JTAG_SEL_ENABLE

The default values of all the above eFuse parameters are 0, which means that they are not burnt. Given that eFuse is one-time programmable, once programmed to 1, it can never be reverted to 0. For how to program eFuse parameters, please refer to [ESP32-C6 Technical Reference Manual](#) > Chapter eFuse Controller.

The default values of the strapping pins, namely the logic levels, are determined by pins' internal weak pull-up/pull-down resistors at reset if the pins are not connected to any circuit, or connected to an external high-impedance circuit.

Table 3-1. Default Configuration of Strapping Pins

| Strapping Pin | Default Configuration | Bit Value |
|---------------|-----------------------|-----------|
| MTMS | Floating | – |
| MTDI | Floating | – |
| GPIO8 | Floating | – |
| GPIO9 | Weak pull-up | 1 |
| GPIO15 | Floating | – |

To change the bit values, the strapping pins should be connected to external pull-down/pull-up resistances. If the ESP32-C6 is used as a device by a host MCU, the strapping pin voltage levels can also be controlled by the host MCU.

All strapping pins have latches. At chip power-on reset and brownout reset, the latches sample the bit values of their respective strapping pins and store them until the chip is powered down or shut down. The states of latches cannot be changed in any other way. It makes the strapping pin values available during the entire chip operation, and the pins are freed up to be used as regular IO pins after reset.

The timing of signals connected to the strapping pins should adhere to the *setup time* and *hold time* specifications in Table 3-2 and Figure 3-1.

Table 3-2. Description of Timing Parameters for the Strapping Pins

| Parameter | Description | Min (ms) |
|-----------|--|----------|
| t_{SU} | Setup time is the time reserved for the power rails to stabilize before the CHIP_PU pin is pulled high to activate the chip. | 0 |
| t_H | Hold time is the time reserved for the chip to read the strapping pin values after CHIP_PU is already high and before these pins start operating as regular IO pins. | 3 |

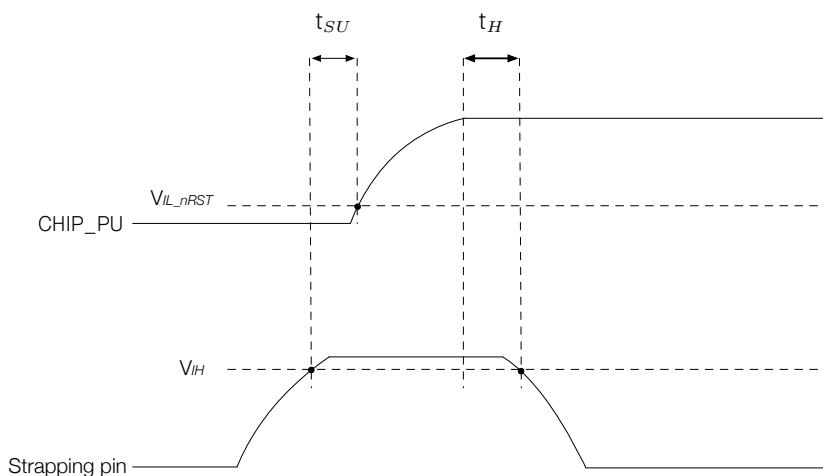


Figure 3-1. Visualization of Timing Parameters for the Strapping Pins

3.1 Chip Boot Mode Control

GPIO8 and GPIO9 control the boot mode after the reset is released. See Table 3-3 *Chip Boot Mode Control*.

Table 3-3. Chip Boot Mode Control

| Boot Mode | GPIO8 | GPIO9 |
|---------------------------------------|-----------|-------|
| SPI boot mode | Any value | 1 |
| Joint download boot mode ² | 1 | 0 |

¹ **Bold** marks the default value and configuration.

² Joint Download Boot mode supports the following download methods:

- USB-Serial-JTAG Download Boot
- UART Download Boot
- SDIO Download Boot

3.2 SDIO Sampling and Driving Clock Edge Control

The strapping pin MTMS and MTDI can be used to decide on which clock edge to sample signals and drive output lines. See Table 3-4 *SDIO Input Sampling Edge/Output Driving Edge Control*.

Table 3-4. SDIO Input Sampling Edge/Output Driving Edge Control

| Edge behavior | MTMS | MTDI |
|--|------|------|
| Falling edge sampling, falling edge output | 0 | 0 |
| Falling edge sampling, rising edge output | 0 | 1 |
| Rising edge sampling, falling edge output | 1 | 0 |
| Rising edge sampling, rising edge output | 1 | 1 |

¹ MTMS and MTDI are floating by default, so above are not default configurations.

3.3 ROM Messages Printing Control

During the boot process, ROM message printing is enabled if LP_AON_STORE4_REG[0] is 0 (default), and disabled if LP_AON_STORE4_REG[0] is 1. When ROM message printing is enabled, the messages can be printed to:

- **(Default) UARTO and USB Serial/JTAG controller**
- USB Serial/JTAG controller
- UARTO

EFUSE_UART_PRINT_CONTROL and GPIO8 control ROM messages printing to **UARTO** as shown in Table 3-5 *UARTO ROM Message Printing Control*.

Table 3-5. UARTO ROM Message Printing Control

| UARTO ROM Code Printing | EFUSE_UART_PRINT_CONTROL | GPIO8 |
|-------------------------|--------------------------|---------|
| Enabled | 0 | Ignored |
| | 1 | 0 |
| | 2 | 1 |
| Disabled | 1 | 1 |
| | 2 | 0 |
| | 3 | Ignored |

¹ **Bold** marks the default value and configuration.

EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT controls the printing to **USB Serial/JTAG controller** as shown in Table 3-6 *USB Serial/JTAG ROM Message Printing Control*.

Table 3-6. USB Serial/JTAG ROM Message Printing Control

| USB Serial/JTAG ROM Code Printing | EFUSE_DIS_USB_SERIAL_JTAG ² | EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT |
|-----------------------------------|--|-------------------------------------|
| Enabled | 0 | 0 |
| Disabled | 0 | 1 |
| | 1 | Ignored |

¹ **Bold** marks the default value and configuration.

² EFUSE_DIS_USB_SERIAL_JTAG controls whether to disable USB Serial/JTAG.

3.4 JTAG Signal Source Control

The strapping pin GPIO15 can be used to control the source of JTAG signals during the early boot process. This pin does not have any internal pull resistors and the strapping value must be controlled by the external circuit that cannot be in a high impedance state.

As Table [3-7 JTAG Signal Source Control](#) shows, GPIO15 is used in combination with EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG and EFUSE_JTAG_SEL_ENABLE.

Table 3-7. JTAG Signal Source Control

| JTAG Signal Source | EFUSE_DIS_PAD_JTAG | EFUSE_DIS_USB_JTAG | EFUSE_JTAG_SEL_ENABLE | GPIO15 |
|----------------------------|--------------------|--------------------|-----------------------|---------|
| USB Serial/JTAG Controller | 0 | 0 | 0 | Ignored |
| | 0 | 0 | 1 | 1 |
| | 1 | 0 | Ignored | Ignored |
| JTAG pins ² | 0 | 0 | 1 | 0 |
| | 0 | 1 | Ignored | Ignored |
| JTAG is disabled | 1 | 1 | Ignored | Ignored |

¹ **Bold** marks the default value and configuration.

² JTAG pins refer to MTDI, MTCK, MTMS, and MTDO.

4 Functional Description

4.1 System

This section describes the core of the chip's operation, covering its microprocessor, memory organization, system components, and security features.

4.1.1 Microprocessor and Master

This subsection describes the core processing units within the chip and their capabilities.

4.1.1.1 High-Performance CPU

The ESP-RISC-V CPU (HP CPU) is a high-performance 32-bit core based on the RISC-V instruction set architecture (ISA) comprising base integer (I), multiplication/division (M), atomic (A) and compressed (C) standard extensions.

Feature List

- Four-stage pipeline that supports an operating clock frequency up to 160 MHz
- [RV32IMAC ISA](#) (instruction set architecture)
- Compatible with RISC-V ISA Manual Volume I: Unprivileged ISA Version 2.2 and RISC-V ISA Manual, Volume II: Privileged Architecture, Version 1.10
- Zero wait cycle access to on-chip SRAM and Cache for program and data access over IRAM/DRAM interface
- Branch target buffer (BTB) with static branch prediction
- User (U) mode support along with interrupt delegation
- Interrupt controller with up to 28 external vectored interrupts for both M and U modes with 16 programmable priority and threshold levels
- Core local interrupts (CLINT) dedicated for each privilege mode
- Debug module (DM) compliant with the specification RISC-V External Debug Support Version 0.13 with external debugger support over an industry-standard JTAG/USB port
- Support for instruction trace, see Section [4.1.1.2 RISC-V Trace Encoder](#)
- Hardware trigger compliant to the specification RISC-V External Debug Support Version 0.13 with up to 4 breakpoints/watchpoints
- Physical memory protection (PMP) and attributes (PMA) for up to 16 configurable regions

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *High-Performance CPU*.

4.1.1.2 RISC-V Trace Encoder

The RISC-V Trace Encoder in the ESP32-C6 chip provides a way to capture detailed trace information from the High-Performance CPU's execution, enabling deeper analysis and optimization of the system. It connects to

the HP CPU's instruction trace interface and compresses the information into smaller packets, which are then stored in internal SRAM.

Feature List

- Compatible with RISC-V Processor Trace Version 1.0
- Synchronization packets sent every few clock cycles or packets
- Zero bytes as anchor tags to identify boundaries between data packets
- Configurable memory writing mode: loop mode or non-loop mode
- Trace lost status to indicate packet loss
- Automatic restart after packet loss

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *RISC-V Trace Encoder (TRACE)*.

4.1.1.3 Low-Power CPU

The ESP32-C6 Low-Power CPU (LP CPU) is a 32-bit processor based on the RISC-V ISA comprising integer (I), multiplication/division (M), atomic (A), and compressed (C) standard extensions. It is designed for ultra-low power consumption and is capable of staying powered on during Deep-sleep mode when the HP CPU is powered down.

Feature List

- Two-stage pipeline that supports a clock frequency of up to 20 MHz
- [RV32IMAC ISA](#) (instruction set architecture)
- 19 vector interrupts
- Debug module compliant with RISC-V External Debug Support Version 0.13 with external debugger support over an industry-standard JTAG/USB port
- Hardware trigger compliant with RISC-V External Debug Support Version 0.13 with up to 2 breakpoints/watchpoints
- 32-bit AHB system bus for peripheral and memory access
- Core performance metric events
- Able to wake up the HP CPU and send an interrupt to it
- Access to HP memory and LP memory
- Access to the entire peripheral address space

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Low-Power CPU*.

4.1.1.4 GDMA Controller

The GDMA Controller is a General Direct Memory Access (GDMA) controller that allows peripheral-to-memory, memory-to-peripheral, and memory-to-memory data transfer with the CPU's intervention. The GDMA has six

independent channels, three transmit and three receive. These channels are shared by peripherals with the GDMA feature, such as SPI2, UHCI (UART0/UART1), I2S, AES, SHA, ADC, and PARLIO.

Feature List

- Programmable length of data to be transferred in bytes
- Linked list of descriptors for efficient data transfer management
- INCR burst transfer when accessing internal RAM for improved performance
- Access to an address space of up to 384 KB in internal RAM
- Software-configurable selection of peripheral requesting service
- Fixed-priority and round-robin channel arbitration schemes for managing bandwidth
- Support for Event Task Matrix

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter GDMA Controller (DMA).

4.1.2 Memory Organization

This subsection describes the memory arrangement to explain how data is stored, accessed, and managed for efficient operation.

Figure 4-1 illustrates the address mapping structure of ESP32-C6.

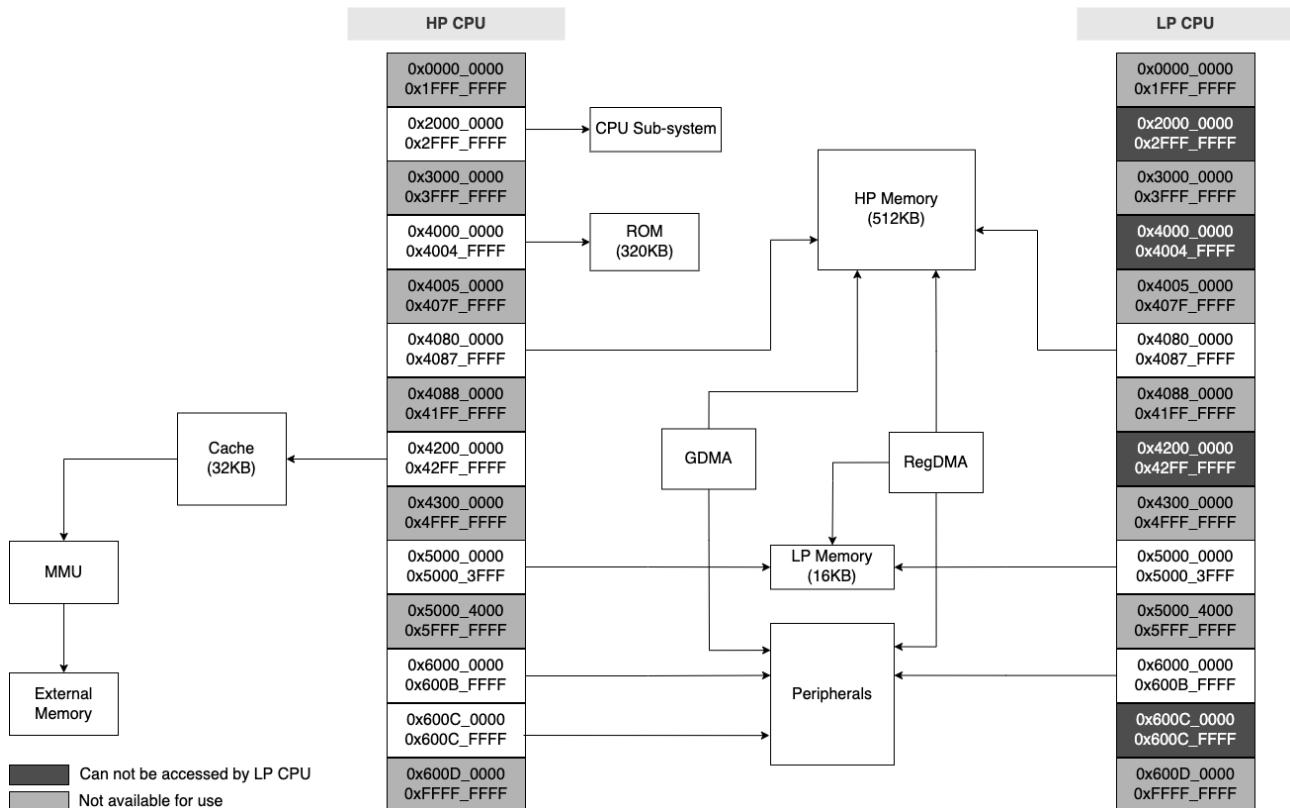


Figure 4-1. Address Mapping Structure

4.1.2.1 Internal Memory

The internal memory of ESP32-C6 refers to the memory integrated on the chip die or in the chip package, including ROM, SRAM, eFuse, and flash.

Feature List

- 320 KB of ROM for booting and core functions
- 512 KB of high-performance SRAM (HP SRAM) for data and instructions
- 16 KB of low-power SRAM (LP SRAM) that can be accessed by HP CPU or LP CPU. It can retain data in Deep-sleep mode
- 4096-bit eFuse memory, with 1792 bits available for users. See also Section [4.1.2.3 eFuse Controller](#)
 - See flash size in Chapter [1 ESP32-C6 Series Comparison](#)
 - More than 100,000 program/erase cycles
 - More than 20 years of data retention time
 - Clock frequency up to 80 MHz by default
- In-package flash
 -
 -
 -
 -

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter System and Memory.

4.1.2.2 External Memory

ESP32-C6 allows connection to memories outside the chip's package via the SPI, Dual SPI, Quad SPI, and QPI interfaces.

Feature List

- Support connection to [off-package flash](#) of 16 MB at most
 - Support hardware encryption/decryption based on XTS-AES
 - Up to 16 MB of CPU instruction memory space can map into flash as individual blocks of 64 KB. 32-bit fetch is supported
 - Up to 16 MB of CPU data memory space can map into flash as individual blocks of 64 KB. 8-bit, 16-bit and 32-bit reads are supported
- External memory accessed via a 32 KB read-only cache
 - Four-way set associative
 - 32-byte cache block
 - Critical word first and early restart

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter System and Memory.

4.1.2.3 eFuse Controller

The eFuse memory is a one-time programmable memory that stores parameters and user data, and the eFuse controller of ESP32-C6 is used to program and read this eFuse memory.

Feature List

- Configure write protection for some blocks
- Configure read protection for some blocks
- Various hardware encoding schemes against data corruption

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter eFuse Controller.

4.1.3 System Components

This subsection describes the essential components that contribute to the overall functionality and control of the system.

4.1.3.1 IO MUX and GPIO Matrix

The IO MUX and GPIO Matrix in the ESP32-C6 chip provide flexible routing of peripheral input and output signals to the GPIO pins. These peripherals enhance the functionality and performance of the chip by allowing the configuration of I/O, support for multiplexing, and signal synchronization for peripheral inputs.

Feature List

- 30 or 22 GPIO pins for general-purpose I/O or connection to internal peripheral signals
- GPIO matrix:
 - Routing 85 peripheral input and 93 output signals to any GPIO pin
 - Signal synchronization for peripheral inputs based on IO MUX operating clock
 - GPIO Filter hardware for input signal filtering
 - Glitch Filter hardware for second time filtering on input signal
 - Sigma delta modulated (SDM) output
- IO MUX for directly connecting certain digital signals (SPI, JTAG, UART) to pins
- LP IO MUX for controlling eight LP GPIO pins (GPIO0 ~ GPIO7) used by peripherals in the LP system
- Support for Event Task Matrix

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter IO MUX and GPIO Matrix.

4.1.3.2 Reset

The ESP32-C6 chip provides four types of reset that occur at different levels, namely CPU Reset, Core Reset, System Reset, and Chip Reset. Except for Chip Reset, all reset types preserve the data stored in internal memory.

Feature List

- Four types of reset:
 - CPU Reset – Resets the CPU core
 - Core Reset – Resets the whole digital system except for the LP system
 - System reset – Resets the whole digital system, including the LP system
 - Chip reset – Resets the whole chip
- Reset trigger:
 - Directly by hardware
 - Via software by configuring the corresponding registers of the CPU
- Support for retrieving reset cause

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Reset and Clock.

4.1.3.3 Clock

The ESP32-C6 chip has clocks sourced from oscillators, RC circuits, and PLL circuits, which are then processed by dividers or selectors. The clocks can be classified into high speed clocks for devices working at higher frequencies and slow speed clocks for low-power systems and some peripherals.

Feature List

- High speed clocks for HP system
 - 40 MHz external crystal clock

Note:
The chip cannot operate without the external crystal clock.

- 480 MHz internal PLL clock
- Slow speed clocks for LP system and some peripherals working in low-power mode
 - 32 kHz external crystal clock
 - Internal fast RC oscillator with adjustable frequency (17.5 MHz by default)
 - 136 kHz Internal slow RC oscillator
 - External slow clock input through XTAL_32K_P (32 kHz by default)

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Reset and Clock.

4.1.3.4 Interrupt Matrix

The Interrupt Matrix in the ESP32-C6 chip routes interrupt requests generated by various peripherals to CPU interrupts.

Feature List

- 77 peripheral interrupt sources accepted as input
- 31 CPU peripheral interrupts generated to CPU as output
- Current interrupt status query of peripheral interrupt sources
- Multiple interrupt sources mapping to a single CPU interrupt (i.e., shared interrupts)

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Interrupt Matrix*.

4.1.3.5 Event Task Matrix

The Event Task Matrix (ETM) allows events from any specified peripheral to be mapped to tasks of any specified peripheral, enabling peripherals to execute specified tasks without CPU intervention. Peripherals supporting ETM include GPIO, LED PWM, general-purpose timers, RTC Timer, system timer, MCPWM, temperature sensor, ADC, I2S, LP CPU, GDMA, and PMU.

Feature List

- 50 channels that can be enabled and configured independently
- Receive 124 events from multiple peripherals
- Generate 130 tasks for multiple peripherals

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Event Task Matrix*.

4.1.3.6 System Timer

The System Timer (SYSTIMER) in the ESP32-C6 chip is a 52-bit timer that can be used to generate tick interrupts for the operating system or as a general timer to generate periodic or one-time interrupts.

Feature List

- Two 52-bit counters and three 52-bit comparators
- 52-bit alarm values and 26-bit alarm periods
- Two modes to generate alarms: target mode and period mode
- Three comparators generating three independent interrupts based on configured alarm value or alarm period
- Ability to load back sleep time recorded by RTC timer via software after Deep-sleep or Light-sleep
- Counters can be stalled if the CPU is stalled or in OCD mode
- Real-time alarm events

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *System Timer*.

4.1.3.7 Power Management Unit

The ESP32-C6 has an advanced Power Management Unit (PMU). It can be flexibly configured to power up different power domains of the chip to achieve the best balance between chip performance, power consumption, and wakeup latency.

The integrated LP CPU allow the ESP32-C6 to operate in Deep-sleep mode with most of the power domains turned off, thus achieving extremely low-power consumption.

Configuring the PMU is a complex procedure. To simplify power management for typical scenarios, there are the following **predefined power modes** that power up different combinations of power domains:

- **Active mode** – The HP CPU, RF circuits, and all peripherals are on. The chip can process data, receive, transmit, and listen.
- **Modem-sleep mode** – The HP CPU is on, but the clock frequency can be reduced. The wireless connections can be configured to remain active as RF circuits are periodically switched on when required.
- **Light-sleep mode** – The HP CPU stops running, and can be optionally powered on. The LP peripherals, as well as the LP CPU can be woken up periodically by the timer. The chip can be woken up via all wake up mechanisms: MAC, SDIO host, RTC timer, or external interrupts. Wireless connections can remain active. Some groups of digital peripherals can be optionally powered off.
- **Deep-sleep mode** – Only the LP system is powered on. Wireless connection data is stored in LP memory.

For power consumption in different power modes, see Section [5.6 Current Consumption Characteristics](#).

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Low-Power Management.

4.1.3.8 Timer Group

The Timer Group (TIMG) in the ESP32-C6 chip can be used to precisely time an interval, trigger an interrupt after a particular interval (periodically and aperiodically), or act as a hardware clock. ESP32-C6 has two timer groups, each consisting of one general-purpose timer and one Main System Watchdog Timer.

Feature List

- 16-bit prescaler
- 54-bit auto-reload-capable up-down counter
- Able to read real-time value of the time-base counter
- Halt, resume, and disable the time-base counter
- Programmable alarm generation
- Timer value reload (auto-reload at an alarm or a software-controlled instant reload)
- RTC slow clock frequency calculation
- Real-time alarm events
- Level interrupt generation

- Support for several ETM tasks and events

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Timer Group (TIMG)*.

4.1.3.9 Watchdog Timers

The Watchdog Timers (WDT) in ESP32-C6 are used to detect and recover from malfunctions. The chip contains three digital watchdog timers: one in each of the two timer groups (MWDT) and one in the RTC Module (RWDT). Additionally, there is one analog watchdog timer called the Super watchdog (SWD) that helps prevent the system from operating in a sub-optimal state.

Feature List

- Digital watchdog timers:
 - Four stages, each with a separately programmable timeout value and timeout action
 - Timeout actions: Interrupt, CPU reset, core reset, system reset (RWDT only)
 - Flash boot protection under SPI Boot mode at stage 0
 - Write protection that makes WDT register read only unless unlocked
 - 32-bit timeout counter
- Analog watchdog timer:
 - Timeout period slightly less than one second
 - Timeout actions: Interrupt, system reset

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Watchdog Timers*.

4.1.3.10 Permission Control

The Permission Control module in ESP32-C6 is responsible for managing access permissions to memory and peripheral registers. It consists of two parts: PMP (Physical Memory Protection) and APM (Access Permission Management).

Feature List

- Access permission management for ROM, HP memory, HP peripheral, LP memory, and LP peripheral address spaces
- APM supports each master (such as DMA) to select one of the four security modes
- Access permission configuration for up to 16 address ranges
- Interrupt function and exception information record

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *Permission Control (PMS)*.

4.1.3.11 System Registers

The System Registers in the ESP32-C6 chip are used to configure various auxiliary chip features.

Feature List

- Control External memory encryption and decryption
- Control HP core/LP core debugging
- Control Bus timeout protection

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter System Registers (HP_SYSREG).

4.1.3.12 Debug Assistant

The Debug Assistant provides a set of functions to help locate bugs and issues during software debugging. It offers various monitoring capabilities and logging features to assist in identifying and resolving software errors efficiently.

Feature List

- Read/write monitoring: Monitor whether the HP CPU bus reads from or writes to a specified memory address space
- Stack pointer (SP) monitoring: Prevent stack overflow or erroneous push/pop operations violation will trigger an interrupt.
- Program counter (PC) logging: Record PC value. The developer can get the last PC value at the most recent HP CPU reset
- Bus access logging: Record information about bus access when the HP CPU, LP CPU, or DMA writes a specified value

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Debug Assistant (ASSIST_DEBUG).

4.1.4 Cryptography and Security Component

This subsection describes the security features incorporated into the chip, which safeguard data and operations.

4.1.4.1 AES Accelerator

ESP32-C6 integrates an Advanced Encryption Standard (AES) accelerator, which is a hardware device that speeds up computation using AES algorithm significantly, compared to AES algorithms implemented solely in software. The AES accelerator integrated in ESP32-C6 has two working modes, which are Typical AES and DMA-AES.

Feature List

- Typical AES working mode
 - AES-128/AES-256 encryption and decryption
- DMA-AES working mode
 - AES-128/AES-256 encryption and decryption
 - Block cipher mode

- * ECB (Electronic Codebook)
 - * CBC (Cipher Block Chaining)
 - * OFB (Output Feedback)
 - * CTR (Counter)
 - * CFB8 (8-bit Cipher Feedback)
 - * CFB128 (128-bit Cipher Feedback)
- Interrupt on completion of computation

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter AES Accelerator (AES).

4.1.4.2 ECC Accelerator

The ECC Accelerator accelerates calculations based on the Elliptic Curve Cryptography (ECC) algorithm and ECC-derived algorithms like ECDSA, which offers the advantages of smaller public keys compared to RSA cryptography with equivalent security.

Feature List

- Supports two different elliptic curves (P-192 and P-256)
- Six working modes that supports Base Point Verification, Base Point Multiplication, Jacobian Point Verification, and Jacobian Point Multiplication

For details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter ECC Accelerator (ECC).

4.1.4.3 HMAC Accelerator

The HMAC Accelerator (HMAC) module is designed to compute Message Authentication Codes (MACs) using the SHA-256 Hash algorithm and keys as described in RFC 2104. It provides hardware support for HMAC computations, significantly reducing software complexity and improving performance.

Feature List

- Standard HMAC-SHA-256 algorithm
- HMAC-SHA-256 calculation based on key in eFuse
 - Whose result cannot be accessed by software in downstream mode for high security
 - Whose result can be accessed by software in upstream mode
- Generates required keys for the Digital Signature Algorithm (DSA) peripheral in downstream mode
- Re-enables soft-disabled JTAG in downstream mode

For details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter HMAC Accelerator.

4.1.4.4 RSA Accelerator

The RSA accelerator provides hardware support for high-precision computation used in various RSA asymmetric cipher algorithms, significantly improving their run time and reducing their software complexity.

Compared with RSA algorithms implemented solely in software, this hardware accelerator can speed up RSA algorithms significantly.

Feature List

- Large-number modular exponentiation with two optional acceleration options, operands width up to 3072 bits
- Large-number modular multiplication, operands width up to 3072 bits
- Large-number multiplication, operands width up to 1536 bits
- Operands of different widths
- Interrupt on completion of computation

For details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter RSA Accelerator.

4.1.4.5 SHA Accelerator

The SHA Accelerator (SHA) is a hardware device that significantly speeds up the SHA algorithm compared to software-only implementations.

Feature List

- Support for multiple SHA algorithms: SHA-1, SHA-224, and SHA-256
- Two working modes: Typical SHA based on CPU and DMA-SHA based on DMA

For more details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter SHA Accelerator (SHA).

4.1.4.6 Digital Signature

The Digital Signature (DS) module in the ESP32-C6 chip generates message signatures based on RSA with hardware acceleration.

Feature List

- RSA digital signatures with key length up to 3072 bits
- Encrypted private key data, only decryptable by DS module
- SHA-256 digest to protect private key data against tampering by an attacker

For more details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter Digital Signature (DS).

4.1.4.7 External Memory Encryption and Decryption

The External Memory Encryption and Decryption (XTS_AES) module in the ESP32-C6 chip provides security for users' application code and data stored in the external memory (flash).

Feature List

- General XTS-AES algorithm, compliant with IEEE Std 1619-2007

- Software-based manual encryption
- High-speed auto decryption without software's participation
- Encryption and decryption functions jointly enabled/disabled by registers configuration, eFuse parameters, and boot mode
- Configurable Anti-DPA

For more details, see the [ESP32-C6 Technical Reference Manual](#) > Chapter *External Memory Encryption and Decryption (XTS_AES)*.

4.1.4.8 Random Number Generator

The Random Number Generator (RNG) in the ESP32-C6 is a true random number generator that generates 32-bit random numbers for cryptographic operations from a physical process.

Feature List

- RNG entropy source
 - Thermal noise from high-speed ADC or SAR ADC
 - An asynchronous clock mismatch

For more details about the Random Number Generator, refer to the [ESP32-C6 Technical Reference Manual](#) > Chapter *Random Number Generator (RNG)*.

4.2 Peripherals

This section describes the chip's peripheral capabilities, covering connectivity interfaces and on-chip sensors that extend its functionality.

4.2.1 Connectivity Interface

This subsection describes the connectivity interfaces on the chip that enable communication and interaction with external devices and networks.

4.2.1.1 UART Controller

The UART Controller in the ESP32-C6 chip facilitates the transmission and reception of asynchronous serial data between the chip and external UART devices. It consists of two UARTs in the main system, and one low-power LP UART.

Feature List

- Programmable baud rates up to 5 MBaud
- RAM shared by TX FIFOs and RX FIFOs
- Support for various lengths of data bits and stop bits
- Parity bit support
- Special character AT_CMD detection
- RS485 protocol support (not supported by LP UART)
- IrDA protocol support (not supported by LP UART)
- High-speed data communication using GDMA (not supported by LP UART)
- Receive timeout feature
- UART as the wake-up source
- Software and hardware flow control

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *UART Controller (UART, LP_UART)*.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.2 SPI Controller

ESP32-C6 has the following SPI interfaces:

- **SPI0** used by ESP32-C6's cache and GDMA to access in-package or off-package flash
- **SPI1** used by the CPU to access in-package or off-package flash
- **SPI2** is a general-purpose SPI controller with access to general-purpose DMA channels

SPI0 and SPI1 are reserved for system use, and only SPI2 is available for users.

Features of SPIO and SPI1

- Supports Single SPI, Dual SPI, Quad SPI (QPI) modes
- Data transmission is in bytes

Features of SPI2

- Supports operation as a master or slave
- Support for GDMA
- Supports Single SPI, Dual SPI, Quad SPI (QPI) modes
- Configurable clock polarity (CPOL) and phase (CPHA)
- Configurable clock frequency
- Data transmission is in bytes
- Configurable read and write data bit order: most-significant bit (MSB) first, or least-significant bit (LSB) first
- As a master
 - Supports 2-line full-duplex communication with clock frequency up to 80 MHz
 - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 80 MHz
 - Provides six FSPICS... pins for connection with six independent SPI slaves
 - Configurable CS setup time and hold time
- As a slave
 - Supports 2-line full-duplex communication with clock frequency up to 40 MHz
 - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 40 MHz

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter SPI Controller (SPI).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.3 I2C Controller

The I2C Controller supports communication between the master and slave devices using the I2C bus.

Feature List

- Two I2C controllers: one in the main system and one in the low-power system
- Communication with multiple external devices
- Master and slave modes for I2C, and master mode only for LP I2C
- Standard mode (100 Kbit/s) and fast mode (400 Kbit/s)
- SCL clock stretching in slave mode

- Programmable digital noise filtering
- Support for 7-bit and 10-bit addressing, as well as dual address mode

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter I2C Controller (I2C).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.4 I2S Controller

The I2S Controller in the ESP32-C6 chip provides a flexible communication interface for streaming digital data in multimedia applications, particularly digital audio applications.

Feature List

- Master mode and slave mode
- Full-duplex and half-duplex communications
- Separate TX and RX units that can work independently or simultaneously
- A variety of audio standards supported:
 - TDM Philips standard
 - TDM MSB alignment standard
 - TDM PCM standard
 - PDM standard
- PCM-to-PDM TX interface
- Configurable high-precision BCK clock, with frequency up to 40 MHz
 - Sampling frequencies can be 8 kHz, 16 kHz, 32 kHz, 44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz, 128 kHz, 192 kHz, etc.
- 8-/16-/24-/32-bit data communication
- Direct Memory Access (DMA)
- A-law and μ-law compression/decompression algorithms for improved signal-to-quantization noise ratio
- Flexible data format control

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter I2S Controller (I2S).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.5 Pulse Count Controller

The Pulse Count Controller (PCNT) is designed to count input pulses by tracking rising and falling edges of the input pulse signal.

Feature List

- Four independent pulse counters with two channels each
- Counter modes: increment, decrement, or disable
- Glitch filtering for input pulse signals and control signals
- Selection between counting on rising or falling edges of the input pulse signal

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Pulse Count Controller.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.6 USB Serial/JTAG Controller

The USB Serial/JTAG controller in the ESP32-C6 chip provides an integrated solution for communicating to the chip over a standard USB CDC-ACM serial port as well as a convenient method for JTAG debugging. It eliminates the need for external chips or JTAG adapters, saving space and reducing cost.

Feature List

- USB 2.0 full speed compliant, capable of up to 12 Mbit/s transfer speed (Note that this controller does not support the faster 480 Mbit/s high-speed transfer mode)
- CDC-ACM virtual serial port and JTAG adapter functionality
- CDC-ACM:
 - CDC-ACM adherent serial port emulation (plug-and-play on most modern OSes)
 - Host controllable chip reset and entry into download mode
- JTAG adapter functionality:
 - Fast communication with CPU debugging core using a compact representation of JTAG instructions
- Support for reprogramming of attached flash memory through the ROM startup code
- Internal PHY

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter USB Serial/JTAG Controller (USB_SERIAL_JTAG).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.7 Two-wire Automotive Interface

The Two-wire Automotive Interface (TWAI[®]) is a multi-master, multi-cast communication protocol designed for automotive applications. The TWAI controller facilitates the communication based on this protocol.

Feature List

- Compatible with ISO 11898-1 protocol (CAN Specification 2.0)
- Standard frame format (11-bit ID) and extended frame format (29-bit ID)
- Bit rates from 1 Kbit/s to 1 Mbit/s
- Multiple modes of operation: Normal, Listen Only, and Self-Test (no acknowledgment required)
- Special transmissions: Single-shot and Self Reception
- Acceptance filter (single and dual filter modes)
- Error detection and handling: error counters, configurable error warning limit, error code capture, arbitration lost capture, automatic transceiver standby

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Two-wire Automotive Interface.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.8 SDIO Slave Controller

The SDIO Slave Controller in the ESP32-C6 chip provides hardware support for the Secure Digital Input/Output (SDIO) device interface. It allows an SDIO host to access the ESP32-C6 via an SDIO bus protocol.

Feature List

- Compatible with SD Physical Layer Specification V2.00 and SDIO V2.00 specifications
- Support for SPI, 1-bit SDIO, and 4-bit SDIO transfer modes
- Clock range of 0 ~ 50 MHz
- Configurable sample and drive clock edge
- Integrated and SDIO-accessible registers for information interaction
- Support for SDIO interrupt mechanism
- Automatic padding data and discarding the padded data on the SDIO bus
- Block size up to 512 bytes
- Interrupt vector between the host and slave for bidirectional interrupt
- Support DMA for data transfer
- Support for wake-up from sleep when connection is retained

For more details about the SDIO Slave Controller, refer to the [ESP32-C6 Technical Reference Manual](#) > Chapter SDIO Slave Controller (SDIO).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.9 LED PWM Controller

The LED PWM Controller (LEDC) is designed to generate PWM signals for LED control.

Feature List

- Six independent PWM generators
- Maximum PWM duty cycle resolution of 20 bits
- Four independent timers with 20-bit counters, configurable fractional clock dividers and counter overflow values
- Adjustable phase of PWM signal output
- PWM duty cycle dithering
- Automatic duty cycle fading
 - Linear duty cycle fading — only one duty cycle range
 - Gamma curve fading — up to 16 duty cycle ranges for each PWM generator, with independently configured fading direction (increase or decrease), fading amount, number of fades, and fading frequency
- PWM signal output in low-power mode (Light-sleep mode)
- Event generation and task response achieved by the Event Task Matrix (ETM)

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter *LED PWM Controller*.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.10 Motor Control PWM

The Motor Control Pulse Width Modulator (MCPWM) is designed for driving digital motors and smart light. The MCPWM is divided into five main modules: PWM timers, PWM operators, Capture module, Fault Detection module, and Event Task Matrix (ETM) module.

Feature List

- Three PWM timers for precise timing and frequency control
 - Every PWM timer has a dedicated 8-bit clock prescaler
 - The 16-bit counter in the PWM timer can work in count-up mode, count-down mode, or count-up-down mode
 - Hardware or software synchronization to trigger a reload on the PWM timer or the prescaler's restart, with selectable hardware synchronization source
- Three PWM operators for generating waveform pairs
 - Six PWM outputs to operate in several topologies
 - Configurable dead time on rising and falling edges; each set up independently

- Modulating of PWM output by high-frequency carrier signals, useful when gate drivers are insulated with a transformer
- Capture module for hardware-based signal processing
 - Speed measurement of rotating machinery
 - Measurement of elapsed time between position sensor pulses
 - Period and duty cycle measurement of pulse train signals
 - Decoding current or voltage amplitude derived from duty-cycle-encoded signals of current/voltage sensors
 - Three individual capture channels, each of which with a 32-bit time-stamp register
 - Selection of edge polarity and prescaling of input capture signals
 - The capture timer can sync with a PWM timer or external signals
- Fault Detection module
 - Programmable fault handling in both cycle-by-cycle mode and one-shot mode
 - A fault condition can force the PWM output to either high or low logic levels
- Event generation and task response achieved by the Event Task Matrix (ETM)

For details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Motor Control PWM (MCPWM).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.11 Remote Control Peripheral

The Remote Control Peripheral (RMT) controls the transmission and reception of infrared remote control signals.

Feature List

- Four channels for sending and receiving infrared remote control signals
- Independent transmission and reception capabilities for each channel
- Support for Normal TX/RX mode, Wrap TX/RX mode, Continuous TX mode
- Modulation on TX pulses and Demodulation on RX pulses
- RX filtering for improved signal reception
- Ability to transmit data simultaneously on multiple channels
- Clock divider counter, state machine, and receiver for each RX channel
- Default allocation of RAM blocks to channels based on channel number
- RAM containing 16-bit entries with “level” and “period” fields

For more details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Remote Control Peripheral (RMT).

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.1.12 Parallel IO Controller

The Parallel IO Controller (PARLIO) in the ESP32-C6 chip enables data transfer between external devices and internal memory on a parallel bus through GDMA. It consists of a transmitter (TX unit) and a receiver (RX unit), making it a versatile interface for connecting various peripherals.

Feature List

- 1/2/4/8/16-bit configurable data bus width
- Half-duplex communication with 16-bit data bus width and full-duplex communication with 8-bit data bus width
- Bit reordering in 1/2/4-bit data bus width mode
- RX unit supports 15 receive modes categorized into three major categories: Level Enable mode, Pulse Enable mode, and Software Enable mode
- TX unit can generate a valid signal aligned with TX

For more details, see [ESP32-C6 Technical Reference Manual](#) > Chapter Parallel IO Controller.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.2 Analog Signal Processing

This subsection describes components on the chip that sense and process real-world data.

4.2.2.1 SAR ADC

ESP32-C6 integrates a Successive Approximation Analog-to-Digital Converter (SAR ADC) to convert analog signals into digital representations.

Feature List

- 12-bit sampling resolution
- Analog voltage sampling from up to seven pins
- Attenuation of input signals for voltage conversion
- Software-triggered one-time sampling
- Timer-triggered multi-channel scanning
- DMA continuous conversion for seamless data transfer
- Two filters with configurable filter coefficient
- Threshold monitoring which helps to trigger an interrupt

- Support for Event Task Matrix

For more details, see [ESP32-C6 Technical Reference Manual](#) > Chapter On-Chip Sensors and Analog Signal Processing.

Pin Assignment

For details, see Section [2.3.5 Peripheral Pin Assignment](#).

4.2.2.2 Temperature Sensor

The Temperature Sensor in the ESP32-C6 chip allows for real-time monitoring of temperature changes inside the chip.

Feature List

- Measurement range: $-40^{\circ}\text{C} \sim 125^{\circ}\text{C}$
- Software triggering, wherein the data can be read continuously once triggered
- Hardware automatic triggering and temperature monitoring
- Configurable temperature offset based on the environment to improve the accuracy
- Adjustable measurement range
- Two automatic monitoring wake-up modes: absolute value mode and incremental value mode
- Support for Event Task Matrix

For more details, see [ESP32-C6 Technical Reference Manual](#) > Chapter On-Chip Sensors and Analog Signal Processing.

4.3 Wireless Communication

This section describes the chip's wireless communication capabilities, spanning radio technology, Wi-Fi, Bluetooth, and 802.15.4.

4.3.1 Radio

This subsection describes the fundamental radio technology embedded in the chip that facilitates wireless communication and data exchange. The ESP32-C6 radio consists of the following blocks:

- 2.4 GHz receiver
- 2.4 GHz transmitter
- bias and regulators
- balun and transmit-receive switch
- clock generator

4.3.1.1 2.4 GHz Receiver

The 2.4 GHz receiver demodulates the 2.4 GHz RF signal to quadrature baseband signals and converts them to the digital domain with two high-resolution, high-speed ADCs. To adapt to varying signal channel conditions, ESP32-C6 integrates RF filters, Automatic Gain Control (AGC), DC offset cancelation circuits, and baseband filters.

4.3.1.2 2.4 GHz Transmitter

The 2.4 GHz transmitter modulates the quadrature baseband signals to the 2.4 GHz RF signal, and drives the antenna with a high-powered CMOS power amplifier. The use of digital calibration further improves the linearity of the power amplifier.

Additional calibrations are integrated to cancel any radio imperfections, such as:

- carrier leakage
- I/Q amplitude/phase matching
- baseband nonlinearities
- RF nonlinearities
- antenna matching

These built-in calibration routines reduce the cost, time, and specialized equipment required for product testing.

4.3.1.3 Clock Generator

The clock generator produces quadrature clock signals of 2.4 GHz for both the receiver and the transmitter. All components of the clock generator are integrated into the chip, including inductors, varactors, filters, regulators and dividers.

The clock generator has built-in calibration and self-test circuits. Quadrature clock phases and phase noise are optimized on chip with patented calibration algorithms which ensure the best performance of the receiver and the transmitter.

4.3.2 Wi-Fi

This subsection describes the chip's Wi-Fi capabilities, which facilitate wireless communication at a high data rate.

4.3.2.1 Wi-Fi Radio and Baseband

The ESP32-C6 Wi-Fi radio and baseband support the following features:

- compliant with IEEE 802.11b/g/n/ax
- 1T1R in 2.4 GHz band
- 802.11ax
 - 20 MHz-only non-AP mode
 - MCS0 ~MCS9
 - uplink and downlink OFDMA
 - downlink MU-MIMO (multi-user, multiple input, multiple output)
 - longer OFDM symbol, with 0.8, 1.6, 3.2 μ s guard interval
 - DCM (dual carrier modulation), up to 16-QAM
 - single-user/multi-user beamformee
 - channel quality indication (CQI)
 - RX STBC (single spatial stream)
- 802.11b/g/n
 - MCS0 ~MCS7 that supports 20 MHz and 40 MHz bandwidth
 - MCS32
 - data rate up to 150 Mbps
 - 0.4 μ s guard interval
- adjustable transmitting power
- antenna diversity

ESP32-C6 supports antenna diversity with an external RF switch. This switch is controlled by one or more GPIOs, and used to select the best antenna to minimize the effects of channel imperfections.

4.3.2.2 Wi-Fi MAC

ESP32-C6 implements the full IEEE 802.11 b/g/n/ax Wi-Fi MAC protocol. It supports the Basic Service Set (BSS) STA and SoftAP operations under the Distributed Control Function (DCF). Power management is handled automatically with minimal host interaction to minimize the active duty period.

The ESP32-C6 Wi-Fi MAC applies the following low-level protocol functions automatically:

- 4 × virtual Wi-Fi interfaces
- infrastructure BSS in Station mode, SoftAP mode, Station + SoftAP mode, and promiscuous mode
- RTS protection, CTS protection, Immediate Block ACK
- fragmentation and defragmentation
- TX/RX A-MPDU, TX/RX A-MSDU
- transmit opportunity (TXOP)
- Wi-Fi multimedia (WMM)
- GCMP, CCMP, TKIP, WAPI, WEP, BIP, WPA2-PSK/WPA2-Enterprise, and WPA3-PSK/WPA3-Enterprise
- automatic beacon monitoring (hardware TSF)
- 802.11mc FTM

Note:

This feature is not supported in some chip revisions. See [ESP32-C6 Series SoC Errata](#).

- 802.11ax supports:
 - target wake time (TWT) requester
 - multiple BSSIDs
 - triggered response scheduling
 - uplink power headroom
 - operating mode
 - buffer status report
 - Multi-user Request-to-Send (MU-RTS), Multi-user Block ACK Request (MU-BAR), and Multi-STA Block ACK (M-BA) frame
 - intra-PPDU power saving mechanism
 - two network allocation vectors (NAV)
 - BSS coloring
 - spatial reuse
 - uplink power headroom
 - operating mode control
 - buffer status report
 - TXOP duration RTS threshold
 - UL-OFDMA random access (UORA)

4.3.2.3 Networking Features

Espressif provides libraries for TCP/IP networking, ESP-WIFI-MESH networking, and other networking protocols over Wi-Fi. TLS 1.0, 1.1 and 1.2 is also supported.

4.3.3 Bluetooth LE

This subsection describes the chip's Bluetooth capabilities, which facilitate wireless communication for low-power, short-range applications. ESP32-C6 includes a Bluetooth Low Energy subsystem that integrates a hardware link controller, an RF/modem block and a feature-rich software protocol stack. It supports the core features of Bluetooth 5 and Bluetooth mesh.

4.3.3.1 Bluetooth LE PHY

Bluetooth Low Energy PHY in ESP32-C6 supports:

- 1 Mbps PHY
- 2 Mbps PHY for higher data rates
- coded PHY for longer range (125 Kbps and 500 Kbps)
- HW listen before talk (LBT)

4.3.3.2 Bluetooth LE Link Controller

Bluetooth Low Energy Link Controller in ESP32-C6 supports:

- LE advertising extensions, to enhance broadcasting capacity and broadcast more intelligent data
- multiple advertisement sets
- simultaneous advertising and scanning
- multiple connections in simultaneous central and peripheral roles
- adaptive frequency hopping and channel assessment
- LE channel selection algorithm #2
- LE power control
- connection parameter update
- high duty cycle non-connectable advertising
- LE privacy 1.2
- LE data packet length extension
- link layer extended scanner filter policies
- low duty cycle directed advertising
- link layer encryption
- LE Ping

4.3.4 802.15.4

This subsection describes the chip's compatibility with the 802.15.4 standard, which facilitates wireless communication for low-power, short-range applications. ESP32-C6 includes an IEEE Standard 802.15.4 subsystem that integrates PHY and MAC layer. It supports various software stacks including Thread, Zigbee, Matter, HomeKit, MQTT and so on.

4.3.4.1 802.15.4 PHY

ESP32-C6's 802.15.4 PHY supports:

- O-QPSK PHY in 2.4 GHz
- 250 Kbps data rate
- RSSI and LQI supported

4.3.4.2 802.15.4 MAC

ESP32-C6 supports most key features defined in [IEEE Standard 802.15.4-2015](#), including:

- CSMA/CA
- active scan and energy detect
- HW frame filter
- HW auto acknowledge
- HW auto frame pending
- coordinated sampled listening (CSL)

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stresses above those listed in Table 5-1 *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only and normal operation of the device at these or any other conditions beyond those indicated in Section 5.2 *Recommended Operating Conditions* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Table 5-1. Absolute Maximum Ratings

| Parameter | Description | Min | Max | Unit |
|-------------------------------|------------------------------|------|------|------|
| Input power pins ¹ | Allowed input voltage | -0.3 | 3.6 | V |
| I_{output} ² | Cumulative IO output current | — | 1000 | mA |
| T_{STORE} | Storage temperature | -40 | 150 | °C |

¹ For more information on input power pins, see Section 2.5.1 *Power Pins*.

² The product proved to be fully functional after all its IO pins were pulled high while being connected to ground for 24 consecutive hours at ambient temperature of 25 °C.

5.2 Recommended Operating Conditions

Table 5-2. Recommended Power Characteristics

| Parameter ¹ | Description | Min | Typ | Max | Unit |
|-------------------------|---------------------------|-----|-----|-----|------|
| VDDA1, VDDA2, VDDA3P3 | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| VDDPST1 | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| VDD_SPI (as input) | — | 3.0 | 3.3 | 3.6 | V |
| VDDPST2 ^{2, 3} | Recommended input voltage | 3.0 | 3.3 | 3.6 | V |
| I_{VDD} | Cumulative input current | 0.5 | — | — | A |
| T_A | Ambient temperature | -40 | — | 105 | °C |

¹ See in conjunction with Section 2.5 *Power Supply*.

² If VDDPST2 is used to power VDD_SPI (see Section 2.5.2 *Power Scheme*), the voltage drop on R_{SPI} should be accounted for. See also Section 5.3 *VDD_SPI Output Characteristics*.

³ If writing to eFuses, the voltage on VDDPST2 should not exceed 3.3 V as the circuits responsible for burning eFuses are sensitive to higher voltages.

5.3 VDD_SPI Output Characteristics

Table 5-3. VDD_SPI Internal and Output Characteristics

| Parameter | Description ¹ | Typ | Unit |
|-----------|---|-----|----------|
| R_{SPI} | VDD_SPI powered by VDDPST2 via R_{SPI} for 3.3 V flash ² | 3 | Ω |

¹ See in conjunction with Section 2.5.2 Power Scheme.

² VDD3P3_RTC must be more than $VDD_{flash_min} + I_{flash_max} * R_{SPI}$; where

- VDD_{flash_min} – minimum operating voltage of flash
- I_{flash_max} – maximum operating current of flash

5.4 DC Characteristics (3.3 V, 25 °C)

Table 5-4. DC Characteristics (3.3 V, 25 °C)

| Parameter | Description | Min | Typ | Max | Unit |
|----------------|---|---------------------|-----|---------------------|------------|
| C_{IN} | Pin capacitance | — | 2 | — | pF |
| V_{IH} | High-level input voltage | $0.75 \times VDD^1$ | — | $VDD^1 + 0.3$ | V |
| V_{IL} | Low-level input voltage | -0.3 | — | $0.25 \times VDD^1$ | V |
| I_{IH} | High-level input current | — | — | 50 | nA |
| I_{IL} | Low-level input current | — | — | 50 | nA |
| V_{OH}^2 | High-level output voltage | $0.8 \times VDD^1$ | — | — | V |
| V_{OL}^2 | Low-level output voltage | — | — | $0.1 \times VDD^1$ | V |
| I_{OH} | High-level source current ($VDD^1 = 3.3$ V, $V_{OH} \geq 2.64$ V, PAD_DRIVER = 3) | — | 40 | — | mA |
| I_{OL} | Low-level sink current ($VDD^1 = 3.3$ V, $V_{OL} = 0.495$ V, PAD_DRIVER = 3) | — | 28 | — | mA |
| R_{PU} | Internal weak pull-up resistor | — | 45 | — | k Ω |
| R_{PD} | Internal weak pull-down resistor | — | 45 | — | k Ω |
| V_{IH_nRST} | Chip reset release voltage CHIP_PU voltage is within the specified range) | $0.75 \times VDD^1$ | — | $VDD^1 + 0.3$ | V |
| V_{IL_nRST} | Chip reset voltage (CHIP_PU voltage is within the specified range) | -0.3 | — | $0.25 \times VDD^1$ | V |

¹ VDD – voltage from a power pin of a respective power domain.

² V_{OH} and V_{OL} are measured using high-impedance load.

5.5 ADC Characteristics

The measurements in this section are taken with an external 100 nF capacitor connected to the ADC, using DC signals as input, and at an ambient temperature of 25 °C with disabled Wi-Fi.

Table 5-5. ADC Characteristics

| Symbol | Min | Max | Unit |
|--|-----|-----|-------------------|
| DNL (Differential nonlinearity) ¹ | -8 | 12 | LSB |
| INL (Integral nonlinearity) | -10 | 10 | LSB |
| Sampling rate | — | 100 | kSPS ² |

¹ To get better DNL results, you can sample multiple times and apply a filter, or calculate the average value.

² kSPS means kilo samples-per-second.

The calibrated ADC results after hardware calibration and [software calibration](#) are shown in Table 5-6. For higher accuracy, you may implement your own calibration methods.

Table 5-6. ADC Calibration Results

| Parameter | Description | Min | Max | Unit |
|-------------|---|-----|-----|------|
| Total error | ATTENO, effective measurement range of 0 ~ 1000 | -12 | 12 | mV |
| | ATTEN1, effective measurement range of 0 ~ 1300 | -12 | 12 | mV |
| | ATTEN2, effective measurement range of 0 ~ 1900 | -23 | 23 | mV |
| | ATTEN3, effective measurement range of 0 ~ 3300 | -40 | 40 | mV |

Note:

The above ADC measurement range and accuracy are applicable to chips manufactured on and after the Date Code **212023** on shielding cases, or assembled on and after the D/C 1 and D/C 2 **2321** on bar-code labels. For chips manufactured or assembled earlier than these date codes, please ask [our sales team](#) to provide the actual range and accuracy according to batch.

For details of Date Code and D/C, please refer to [Espressif Chip Packaging Information](#).

5.6 Current Consumption Characteristics

5.6.1 Current Consumption in Active Mode

The current consumption measurements are taken with a 3.3 V supply at 25 °C ambient temperature.

TX current consumption is rated at a 100% duty cycle.

RX current consumption is rated when the peripherals are disabled and the CPU idle.

Table 5-7. Current Consumption for Wi-Fi (2.4 GHz) in Active Mode

| Work Mode | RF Condition | Description | Peak (mA) |
|---------------------|--------------|-----------------------------------|-----------|
| Active (RF working) | TX | 802.11b, 1 Mbps, DSSS @ 21.0 dBm | 354 |
| | | 802.11g, 54 Mbps, OFDM @ 19.5 dBm | 300 |
| | | 802.11n, HT20, MCS7 @ 18.5 dBm | 280 |
| | | 802.11n, HT40, MCS7 @ 18.0 dBm | 268 |
| | | 802.11ax, MCS9, @ 16.5 dBm | 252 |

Cont'd on next page

Table 5-7 – cont'd from previous page

| Work Mode | RF Condition | Description | Peak (mA) |
|-----------|--------------|-------------------|-----------|
| | RX | 802.11b/g/n, HT20 | 78 |
| | | 802.11n, HT40 | 82 |
| | | 802.11ax, HE20 | 78 |

Table 5-8. Current Consumption for Bluetooth LE in Active Mode

| Work Mode | RF Condition | Description | Peak (mA) |
|---------------------|--------------|--------------------------|-----------|
| Active (RF working) | TX | Bluetooth LE @ 20.0 dBm | 315 |
| | | Bluetooth LE @ 9.0 dBm | 190 |
| | | Bluetooth LE @ 0 dBm | 130 |
| | | Bluetooth LE @ -15.0 dBm | 94 |
| | RX | Bluetooth LE | 71 |

Table 5-9. Current Consumption for 802.15.4 in Active Mode

| Work Mode | RF Condition | Description | Peak (mA) |
|---------------------|--------------|----------------------|-----------|
| Active (RF working) | TX | 802.15.4 @ 20.0 dBm | 305 |
| | | 802.15.4 @ 12.0 dBm | 187 |
| | | 802.15.4 @ 0 dBm | 119 |
| | | 802.15.4 @ -15.0 dBm | 92 |
| | RX | 802.15.4 | 74 |

5.6.2 Current Consumption in Other Modes

Table 5-10. Current Consumption in Modem-sleep Mode

| Mode | CPU Frequency (MHz) | Description | Typ (mA) | |
|----------------------------|---------------------|----------------|---------------------------------|---|
| | | | All Peripherals Clocks Disabled | All Peripherals Clocks Enabled ¹ |
| Modem-sleep ^{2,3} | 160 | CPU is running | 27 | 38 |
| | | CPU is idle | 17 | 28 |
| | 80 | CPU is running | 19 | 30 |
| | | CPU is idle | 14 | 25 |

¹ In practice, the current consumption might be different depending on which peripherals are enabled.

² In Modem-sleep mode, Wi-Fi is clock gated.

³ In Modem-sleep mode, the consumption might be higher when accessing flash.

Table 5-11. Current Consumption in Low-Power Modes

| Mode | Description | Typ (μ A) |
|-------------|---|----------------|
| Light-sleep | CPU and wireless communication modules are powered down, peripheral clocks are disabled, and all GPIOs are high-impedance | 180 |
| | CPU, wireless communication modules and peripherals are powered down, and all GPIOs are high-impedance | 35 |
| Deep-sleep | RTC timer and LP memory are powered on | 7 |
| Power off | CHIP_PU is set to low level, the chip is powered off | 1 |

5.7 Reliability

Table 5-12. Reliability Qualifications

| Test Item | Test Conditions | Test Standard |
|--|---|-----------------------------------|
| HTOL (High Temperature Operating Life) | 125 °C, 1000 hours | JESD22-A108 |
| ESD (Electro-Static Discharge Sensitivity) | HBM (Human Body Mode) ¹ \pm 2000 V | JS-001 |
| | CDM (Charge Device Mode) ² \pm 1000 V | JS-002 |
| Latch up | Current trigger \pm 200 mA | JESD78 |
| | Voltage trigger $1.5 \times VDD_{max}$ | |
| Preconditioning | Bake 24 hours @125 °C Moisture soak (level 3: 192 hours @30 °C, 60% RH) IR reflow solder: 260 + 0 °C, 20 seconds, three times | J-STD-020, JESD47, JESD22-A113 |
| TCT (Temperature Cycling Test) | -65 °C / 150 °C, 500 cycles | JESD22-A104 |
| uHAST (Highly Accelerated Stress Test, unbiased) | 130 °C, 85% RH, 96 hours | JESD22-A118 |
| HTSL (High Temperature Storage Life) | 150 °C, 1000 hours | JESD22-A103 |
| LTS (Low Temperature Storage Life) | -40 °C, 1000 hours | JESD22-A119 |

¹ JEDEC document JEP155 states that 500 V HBM allows safe manufacturing with a standard ESD control process.

² JEDEC document JEP157 states that 250 V CDM allows safe manufacturing with a standard ESD control process.

6 RF Characteristics

This section contains tables with RF characteristics of the Espressif product.

The RF data is measured at the antenna port, where RF cable is connected, including the front-end loss. The front-end circuit is a 0Ω resistor.

Devices should operate in the center frequency range allocated by regional regulatory authorities. The target center frequency range and the target transmit power are configurable by software. See [ESP RF Test Tool and Test Guide](#) for instructions.

Unless otherwise stated, the RF tests are conducted with a 3.3 V ($\pm 5\%$) supply at 25 °C ambient temperature.

6.1 Wi-Fi Radio

Table 6-1. Wi-Fi RF Characteristics

| Name | Description |
|---|---------------------|
| Center frequency range of operating channel | 2412 ~ 2484 MHz |
| Wi-Fi wireless standard | IEEE 802.11b/g/n/ax |

6.1.1 Wi-Fi RF Transmitter (TX) Characteristics

Table 6-2. TX Power with Spectral Mask and EVM Meeting 802.11 Standards

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|------------------------|-----------|-----------|-----------|
| 802.11b, 1 Mbps, DSSS | — | 21.0 | — |
| 802.11b, 11 Mbps, CCK | — | 21.0 | — |
| 802.11g, 6 Mbps, OFDM | — | 20.5 | — |
| 802.11g, 54 Mbps, OFDM | — | 19.5 | — |
| 802.11n, HT20, MCS0 | — | 19.5 | — |
| 802.11n, HT20, MCS7 | — | 18.5 | — |
| 802.11n, HT40, MCS0 | — | 19.0 | — |
| 802.11n, HT40, MCS7 | — | 18.0 | — |
| 802.11ax, HE20, MCS0 | — | 19.5 | — |
| 802.11ax, HE20, MCS9 | — | 16.5 | — |

Table 6-3. TX EVM Test¹

| Rate | Min (dB) | Typ (dB) | Limit (dB) |
|-----------------------|----------|----------|------------|
| 802.11b, 1 Mbps, DSSS | — | -25.5 | -10.0 |
| 802.11b, 11 Mbps, CCK | — | -25.5 | -10.0 |

Cont'd on next page

Table 6-3 – cont'd from previous page

| Rate | Min (dB) | Typ (dB) | Limit (dB) |
|------------------------|-------------|-------------|---------------|
| 802.11g, 6 Mbps, OFDM | — | -26.5 | -5.0 |
| 802.11g, 54 Mbps, OFDM | — | -29.0 | -25.0 |
| 802.11n, HT20, MCS0 | — | -29.0 | -5.0 |
| 802.11n, HT20, MCS7 | — | -30.0 | -27.0 |
| 802.11n, HT40, MCS0 | — | -28.5 | -5.0 |
| 802.11n, HT40, MCS7 | — | -29.5 | -27.0 |
| 802.11ax, HE20, MCS0 | — | -29.0 | -5.0 |
| 802.11ax, HE20, MCS9 | — | -34.0 | -32.0 |

¹ EVM is measured at the corresponding typical TX power provided in Table 6-2 *TX Power with Spectral Mask and EVM Meeting 802.11 Standards* above.

6.1.2 Wi-Fi RF Receiver (RX) Characteristics

For RX tests, the PER (packet error rate) limit is 8% for 802.11b, and 10% for 802.11g/n/ax.

Table 6-4. RX Sensitivity

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|------------------------|--------------|--------------|--------------|
| 802.11b, 1 Mbps, DSSS | — | -99.2 | — |
| 802.11b, 2 Mbps, DSSS | — | -96.8 | — |
| 802.11b, 5.5 Mbps, CCK | — | -93.8 | — |
| 802.11b, 11 Mbps, CCK | — | -90.0 | — |
| 802.11g, 6 Mbps, OFDM | — | -94.0 | — |
| 802.11g, 9 Mbps, OFDM | — | -93.2 | — |
| 802.11g, 12 Mbps, OFDM | — | -92.6 | — |
| 802.11g, 18 Mbps, OFDM | — | -90.0 | — |
| 802.11g, 24 Mbps, OFDM | — | -86.8 | — |
| 802.11g, 36 Mbps, OFDM | — | -83.2 | — |
| 802.11g, 48 Mbps, OFDM | — | -79.0 | — |
| 802.11g, 54 Mbps, OFDM | — | -77.6 | — |
| 802.11n, HT20, MCS0 | — | -93.6 | — |
| 802.11n, HT20, MCS1 | — | -92.4 | — |
| 802.11n, HT20, MCS2 | — | -89.6 | — |
| 802.11n, HT20, MCS3 | — | -86.2 | — |
| 802.11n, HT20, MCS4 | — | -82.8 | — |
| 802.11n, HT20, MCS5 | — | -78.8 | — |
| 802.11n, HT20, MCS6 | — | -77.2 | — |
| 802.11n, HT20, MCS7 | — | -75.6 | — |
| 802.11n, HT40, MCS0 | — | -91.0 | — |
| 802.11n, HT40, MCS1 | — | -90.0 | — |

Cont'd on next page

Table 6-4 – cont'd from previous page

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|----------------------|--------------|--------------|--------------|
| 802.11n, HT40, MCS2 | — | -87.4 | — |
| 802.11n, HT40, MCS3 | — | -83.8 | — |
| 802.11n, HT40, MCS4 | — | -80.8 | — |
| 802.11n, HT40, MCS5 | — | -76.6 | — |
| 802.11n, HT40, MCS6 | — | -75.0 | — |
| 802.11n, HT40, MCS7 | — | -73.4 | — |
| 802.11ax, HE20, MCS0 | — | -93.8 | — |
| 802.11ax, HE20, MCS1 | — | -91.2 | — |
| 802.11ax, HE20, MCS2 | — | -88.4 | — |
| 802.11ax, HE20, MCS3 | — | -85.6 | — |
| 802.11ax, HE20, MCS4 | — | -82.2 | — |
| 802.11ax, HE20, MCS5 | — | -78.4 | — |
| 802.11ax, HE20, MCS6 | — | -76.6 | — |
| 802.11ax, HE20, MCS7 | — | -74.8 | — |
| 802.11ax, HE20, MCS8 | — | -71.0 | — |
| 802.11ax, HE20, MCS9 | — | -69.0 | — |

Table 6-5. Maximum RX Level

| Rate | Min (dBm) | Typ (dBm) | Max (dBm) |
|------------------------|--------------|--------------|--------------|
| 802.11b, 1 Mbps, DSSS | — | 5 | — |
| 802.11b, 11 Mbps, CCK | — | 5 | — |
| 802.11g, 6 Mbps, OFDM | — | 5 | — |
| 802.11g, 54 Mbps, OFDM | — | 0 | — |
| 802.11n, HT20, MCS0 | — | 5 | — |
| 802.11n, HT20, MCS7 | — | 0 | — |
| 802.11n, HT40, MCS0 | — | 5 | — |
| 802.11n, HT40, MCS7 | — | 0 | — |
| 802.11ax, HE20, MCS0 | — | 5 | — |
| 802.11ax, HE20, MCS9 | — | 0 | — |

Table 6-6. RX Adjacent Channel Rejection

| Rate | Min (dB) | Typ (dB) | Max (dB) |
|------------------------|-------------|-------------|-------------|
| 802.11b, 1 Mbps, DSSS | — | 38 | — |
| 802.11b, 11 Mbps, CCK | — | 38 | — |
| 802.11g, 6 Mbps, OFDM | — | 31 | — |
| 802.11g, 54 Mbps, OFDM | — | 20 | — |

Cont'd on next page

Table 6-6 – cont'd from previous page

| Rate | Min (dB) | Typ (dB) | Max (dB) |
|----------------------|-------------|-------------|-------------|
| 802.11n, HT20, MCS0 | — | 31 | — |
| 802.11n, HT20, MCS7 | — | 16 | — |
| 802.11n, HT40, MCS0 | — | 28 | — |
| 802.11n, HT40, MCS7 | — | 10 | — |
| 802.11ax, HE20, MCS0 | — | 25 | — |
| 802.11ax, HE20, MCS9 | — | 2 | — |

6.2 Bluetooth 5 (LE) Radio

Table 6-7. Bluetooth LE RF Characteristics

| Name | Description |
|---|------------------|
| Center frequency range of operating channel | 2402 ~ 2480 MHz |
| RF transmit power range | -15.0 ~ 20.0 dBm |

6.2.1 Bluetooth LE RF Transmitter (TX) Characteristics

Table 6-8. Bluetooth LE - Transmitter Characteristics - 1 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|------------------------------------|---|-----|-------|-----|------|
| Carrier frequency offset and drift | Max. $ f_n _{n=0, 1, 2, 3, \dots k}$ | — | 1.3 | — | kHz |
| | Max. $ f_0 - f_n _{n=2, 3, 4, \dots k}$ | — | 1.5 | — | kHz |
| | Max. $ f_n - f_{n-5} _{n=6, 7, 8, \dots k}$ | — | 0.9 | — | kHz |
| | $ f_1 - f_0 $ | — | 0.6 | — | kHz |
| Modulation characteristics | $\Delta F1_{avg}$ | — | 249.9 | — | kHz |
| | Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$) | — | 212.1 | — | kHz |
| | $\Delta F2_{avg}/\Delta F1_{avg}$ | — | 0.88 | — | — |
| In-band emissions | ± 2 MHz offset | — | -29 | — | dBm |
| | ± 3 MHz offset | — | -36 | — | dBm |
| | $> \pm 3$ MHz offset | — | -39 | — | dBm |

Table 6-9. Bluetooth LE - Transmitter Characteristics - 2 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|------------------------------------|---|-----|-----|-----|------|
| Carrier frequency offset and drift | Max. $ f_n _{n=0, 1, 2, 3, \dots k}$ | — | 2.2 | — | kHz |
| | Max. $ f_0 - f_n _{n=2, 3, 4, \dots k}$ | — | 1.1 | — | kHz |
| | Max. $ f_n - f_{n-5} _{n=6, 7, 8, \dots k}$ | — | 1.1 | — | kHz |
| | $ f_1 - f_0 $ | — | 0.5 | — | kHz |

Cont'd on next page

Table 6-9 – cont'd from previous page

| Parameter | Description | Min | Typ | Max | Unit |
|----------------------------|---|-----|-------|-----|------|
| Modulation characteristics | $\Delta F1_{avg}$ | — | 499.4 | — | kHz |
| | Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$) | — | 443.5 | — | kHz |
| | $\Delta F2_{avg}/\Delta F1_{avg}$ | — | 0.95 | — | — |
| In-band emissions | ± 4 MHz offset | — | -40 | — | dBm |
| | ± 5 MHz offset | — | -41 | — | dBm |
| | $> \pm 5$ MHz offset | — | -42 | — | dBm |

Table 6-10. Bluetooth LE - Transmitter Characteristics - 125 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|------------------------------------|---|-----|-------|-----|------|
| Carrier frequency offset and drift | Max. $ f_n _{n=0, 1, 2, 3, \dots k}$ | — | 0.7 | — | kHz |
| | Max. $ f_0 - f_n _{n=1, 2, 3, \dots k}$ | — | 0.3 | — | kHz |
| | $ f_0 - f_3 $ | — | 0.1 | — | kHz |
| | Max. $ f_n - f_{n-3} _{n=7, 8, 9, \dots k}$ | — | 0.4 | — | kHz |
| Modulation characteristics | $\Delta F1_{avg}$ | — | 250.0 | — | kHz |
| | Min. $\Delta F1_{max}$ (for at least 99.9% of all $\Delta F1_{max}$) | — | 238.0 | — | kHz |
| In-band emissions | ± 2 MHz offset | — | -29 | — | dBm |
| | ± 3 MHz offset | — | -36 | — | dBm |
| | $> \pm 3$ MHz offset | — | -39 | — | dBm |

Table 6-11. Bluetooth LE - Transmitter Characteristics - 500 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|------------------------------------|---|-----|-------|-----|------|
| Carrier frequency offset and drift | Max. $ f_n _{n=0, 1, 2, 3, \dots k}$ | — | 0.5 | — | kHz |
| | Max. $ f_0 - f_n _{n=1, 2, 3, \dots k}$ | — | 0.3 | — | kHz |
| | $ f_0 - f_3 $ | — | 0.1 | — | kHz |
| | Max. $ f_n - f_{n-3} _{n=7, 8, 9, \dots k}$ | — | 0.4 | — | kHz |
| Modulation characteristics | $\Delta F2_{avg}$ | — | 230.7 | — | kHz |
| | Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$) | — | 217.6 | — | kHz |
| In-band emissions | ± 2 MHz offset | — | -28 | — | dBm |
| | ± 3 MHz offset | — | -36 | — | dBm |
| | $> \pm 3$ MHz offset | — | -39 | — | dBm |

6.2.2 Bluetooth LE RF Receiver (RX) Characteristics

Table 6-12. Bluetooth LE - Receiver Characteristics - 1 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|--|--|-----|-------|-----|------|
| Sensitivity @30.8% PER | — | — | -98.5 | — | dBm |
| Maximum received signal @30.8% PER | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel $F = F_0$ MHz | — | 7 | — | dB |
| | $F = F_0 + 1$ MHz | — | 4 | — | dB |
| | $F = F_0 - 1$ MHz | — | 3 | — | dB |
| | $F = F_0 + 2$ MHz | — | -21 | — | dB |
| | $F = F_0 - 2$ MHz | — | -22 | — | dB |
| | $F = F_0 + 3$ MHz | — | -28 | — | dB |
| | $F = F_0 - 3$ MHz | — | -36 | — | dB |
| | $F \geq F_0 + 4$ MHz | — | -27 | — | dB |
| | $F \leq F_0 - 4$ MHz | — | -36 | — | dB |
| | Image frequency | — | -26 | — | dB |
| Out-of-band blocking performance | Adjacent channel to image frequency $F = F_{image} + 1$ MHz | — | -29 | — | dB |
| | $F = F_{image} - 1$ MHz | — | -28 | — | dB |
| Intermodulation | 30 MHz ~ 2000 MHz | — | -16 | — | dBm |
| | 2003 MHz ~ 2399 MHz | — | -24 | — | dBm |
| | 2484 MHz ~ 2997 MHz | — | -16 | — | dBm |
| | 3000 MHz ~ 12.75 GHz | — | -1 | — | dBm |

Table 6-13. Bluetooth LE - Receiver Characteristics - 2 Mbps

| Parameter | Description | Min | Typ | Max | Unit |
|--|--|-----|-------|-----|------|
| Sensitivity @30.8% PER | — | — | -95.5 | — | dBm |
| Maximum received signal @30.8% PER | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel $F = F_0$ MHz | — | 8 | — | dB |
| | $F = F_0 + 2$ MHz | — | 3 | — | dB |
| | $F = F_0 - 2$ MHz | — | 2 | — | dB |
| | $F = F_0 + 4$ MHz | — | -23 | — | dB |
| | $F = F_0 - 4$ MHz | — | -25 | — | dB |
| | $F = F_0 + 6$ MHz | — | -31 | — | dB |
| | $F = F_0 - 6$ MHz | — | -35 | — | dB |
| | $F \geq F_0 + 8$ MHz | — | -36 | — | dB |
| | $F \leq F_0 - 8$ MHz | — | -36 | — | dB |
| | Image frequency | — | -23 | — | dB |
| Out-of-band blocking performance | Adjacent channel to image frequency $F = F_{image} + 2$ MHz | — | -30 | — | dB |
| | $F = F_{image} - 2$ MHz | — | 3 | — | dB |
| Intermodulation | 30 MHz ~ 2000 MHz | — | -18 | — | dBm |
| | 2003 MHz ~ 2399 MHz | — | -28 | — | dBm |
| | 2484 MHz ~ 2997 MHz | — | -16 | — | dBm |

Cont'd on next page

Table 6-13 – cont'd from previous page

| Parameter | Description | Min | Typ | Max | Unit |
|-----------------|----------------------|-----|-----|-----|------|
| | 3000 MHz ~ 12.75 GHz | — | -1 | — | dBm |
| Intermodulation | — | — | -29 | — | dBm |

Table 6-14. Bluetooth LE - Receiver Characteristics - 125 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|--|-------------------------------------|--------------------------------|--------|-----|------|
| Sensitivity @30.8% PER | — | — | -106.0 | — | dBm |
| Maximum received signal @30.8% PER | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel | F = F ₀ MHz | — | 2 | — |
| | Adjacent channel | F = F ₀ + 1 MHz | — | -1 | — |
| | | F = F ₀ - 1 MHz | — | -3 | — |
| | | F = F ₀ + 2 MHz | — | -31 | — |
| | | F = F ₀ - 2 MHz | — | -27 | — |
| | | F = F ₀ + 3 MHz | — | -33 | — |
| | | F = F ₀ - 3 MHz | — | -42 | — |
| | | F ≥ F ₀ + 4 MHz | — | -31 | — |
| | | F ≤ F ₀ - 4 MHz | — | -48 | — |
| | Image frequency | — | — | -31 | — |
| | Adjacent channel to image frequency | F = F _{image} + 1 MHz | — | -36 | — |
| | | F = F _{image} - 1 MHz | — | -33 | — |

Table 6-15. Bluetooth LE - Receiver Characteristics - 500 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|--|-------------------------------------|--------------------------------|--------|-----|------|
| Sensitivity @30.8% PER | — | — | -102.0 | — | dBm |
| Maximum received signal @30.8% PER | — | — | 8 | — | dBm |
| C/I and receiver selectivity performance | Co-channel | F = F ₀ MHz | — | 4 | — |
| | Adjacent channel | F = F ₀ + 1 MHz | — | 1 | — |
| | | F = F ₀ - 1 MHz | — | -1 | — |
| | | F = F ₀ + 2 MHz | — | -23 | — |
| | | F = F ₀ - 2 MHz | — | -24 | — |
| | | F = F ₀ + 3 MHz | — | -33 | — |
| | | F = F ₀ - 3 MHz | — | -41 | — |
| | | F ≥ F ₀ + 4 MHz | — | -31 | — |
| | | F ≤ F ₀ - 4 MHz | — | -41 | — |
| | Image frequency | — | — | -30 | — |
| | Adjacent channel to image frequency | F = F _{image} + 1 MHz | — | -35 | — |
| | | F = F _{image} - 1 MHz | — | -27 | — |

6.3 802.15.4 Radio

Table 6-16. 802.15.4 RF Characteristics

| Name | Description |
|---|-----------------|
| Center frequency range of operating channel | 2405 ~ 2480 MHz |

¹ Zigbee in the 2.4 GHz range supports 16 channels at 5 MHz spacing from channel 11 to channel 26.

6.3.1 802.15.4 RF Transmitter (TX) Characteristics

Table 6-17. 802.15.4 Transmitter Characteristics - 250 Kbps

| Parameter | Min | Typ | Max | Unit |
|-------------------------|-------|-------|------|------|
| RF transmit power range | -15.0 | — | 20.0 | dBm |
| EVM | — | 13.0% | — | — |

6.3.2 802.15.4 RF Receiver (RX) Characteristics

Table 6-18. 802.15.4 Receiver Characteristics - 250 Kbps

| Parameter | Description | Min | Typ | Max | Unit |
|---------------------------------|-------------------|-----------------------------|--------|-----|------|
| Sensitivity @1% PER | — | — | -104.0 | — | dBm |
| Maximum received signal @1% PER | — | — | 8 | — | dBm |
| Relative jamming level | Adjacent channel | F = F ₀ + 5 MHz | — | 27 | dB |
| | | F = F ₀ - 5 MHz | — | 32 | dB |
| | Alternate channel | F = F ₀ + 10 MHz | — | 47 | dB |
| | | F = F ₀ - 10 MHz | — | 50 | dB |

7 Packaging

- For information about tape, reel, and chip marking, please refer to [Espressif Chip Packaging Information](#).
- The pins of the chip are numbered in anti-clockwise order starting from Pin 1 in the top view. For pin numbers and pin names, see also Figure 2-1 *ESP32-C6 Pin Layout (QFN40, Top View)* and Figure 2-2 *ESP32-C6 Pin Layout (QFN32, Top View)*.
- The recommended land pattern [source file \(asc\)](#) is available for download. You can import the file with software such as PADS and Altium Designer.

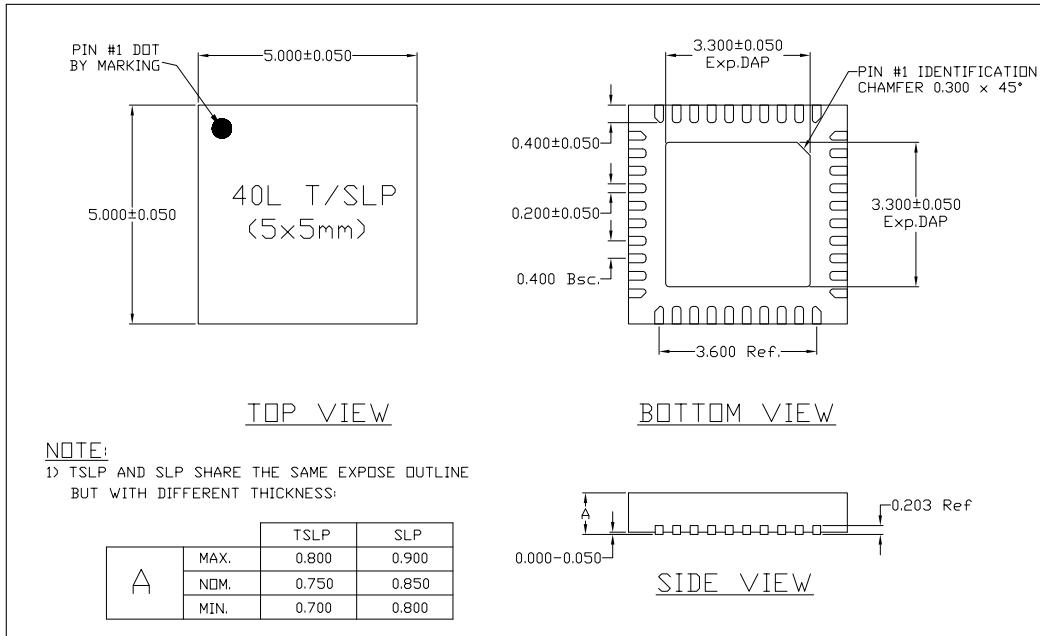


Figure 7-1. QFN40 (5x5 mm) Package

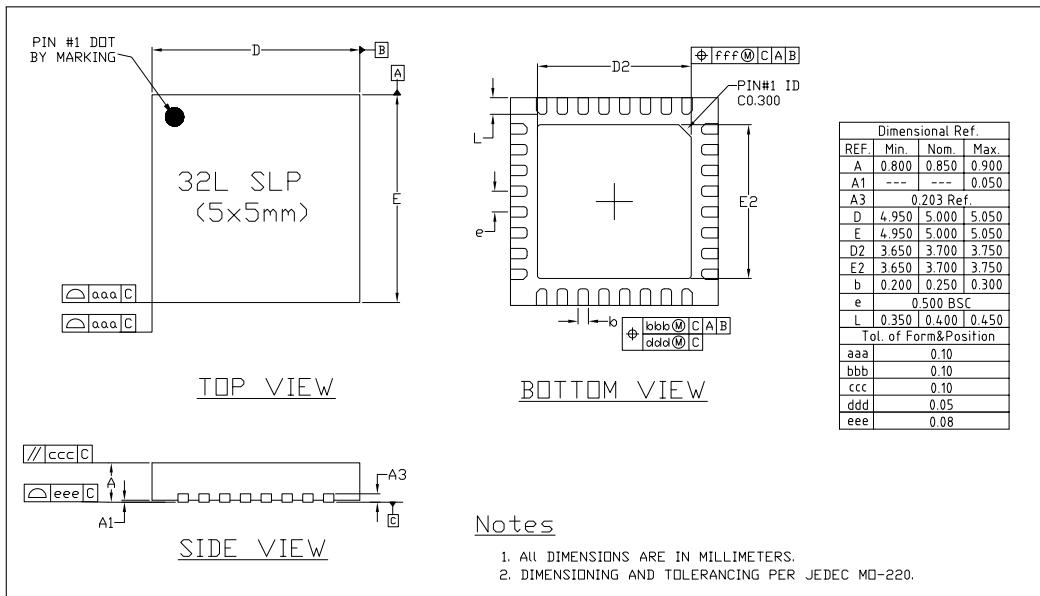


Figure 7-2. QFN32 (5x5 mm) Package

Appendix A – ESP32-C6 Consolidated Pin Overview

Table 7-1. QFN40 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power | Pin Settings | | Analog Function | | LP IO MUX Function | | IO MUX Function | | | |
|---------|------------|----------|---------------------|--------------|-------------|-----------------|----------|--------------------|--------------|-----------------|--------|--------|-------|
| | | | | At Reset | After Reset | 0 | 1 | 0 | 1 | 0 | Type | 1 | Type |
| 1 | ANT | Analog | | | | | | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | | | | | | |
| 4 | CHIP_PU | Analog | | | | | | | | | | | |
| 5 | VDDPST1 | Power | | | | | | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | XTAL_32K_P | ADC1_CH0 | LP_GPIO0 | LP_UART_DTRN | GPIO0 | I/O/T | GPIO0 | I/O/T |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | XTAL_32K_N | ADC1_CH1 | LP_GPIO1 | LP_UART_DSRN | GPIO1 | I/O/T | GPIO1 | I/O/T |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | | ADC1_CH2 | LP_GPIO2 | LP_UART_RTSN | GPIO2 | I/O/T | GPIO2 | I/O/T |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | | ADC1_CH3 | LP_GPIO3 | LP_UART_CTSN | GPIO3 | I/O/T | GPIO3 | I/O/T |
| 10 | MTMS | IO | VDDPST1 | IE | IE | | ADC1_CH4 | LP_GPIO4 | LP_UART_RXD | MTMS | I1 | GPIO4 | I/O/T |
| 11 | MTDI | IO | VDDPST1 | IE | IE | | ADC1_CH5 | LP_GPIO5 | LP_UART_TXD | MTDI | I1 | GPIO5 | I/O/T |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU | | ADC1_CH6 | LP_GPIO6 | LP_I2C_SDA | MTCK | I1 | GPIO6 | I/O/T |
| 13 | MTDO | IO | VDDPST1 | | IE | | | LP_GPIO7 | LP_I2C_SCL | MTDO | O/T | GPIO7 | I/O/T |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | | | | | GPIO8 | I/O/T | GPIO8 | I/O/T |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | | | | | GPIO9 | I/O/T | GPIO9 | I/O/T |
| 16 | GPIO10 | IO | VDDPST2 | | IE | | | | | GPIO10 | I/O/T | GPIO10 | I/O/T |
| 17 | GPIO11 | IO | VDDPST2 | | IE | | | | | GPIO11 | I/O/T | GPIO11 | I/O/T |
| 18 | GPIO12 | IO | VDDPST2 | | IE | USB_D- | | | | GPIO12 | I/O/T | GPIO12 | I/O/T |
| 19 | GPIO13 | IO | VDDPST2 | | IE, WPU | USB_D+ | | | | GPIO13 | I/O/T | GPIO13 | I/O/T |
| 20 | SPICSO | IO | VDD_SPI | WPU | IE, WPU | | | | | SPICSO | O/T | GPIO24 | I/O/T |
| 21 | SPIQ | IO | VDD_SPI | WPU | IE, WPU | | | | | SPIQ | I/O/T | GPIO25 | I/O/T |
| 22 | SPIWP | IO | VDD_SPI | WPU | IE, WPU | | | | | SPIWP | I/O/T | GPIO26 | I/O/T |
| 23 | VDD_SPI | Power/IO | — | | | VDD_SPI | | | | GPIO27 | I/O/T | GPIO27 | I/O/T |
| 24 | SPIHD | IO | VDD_SPI | WPU | IE, WPU | | | | | SPIHD | I/O/T | GPIO28 | I/O/T |
| 25 | SPICLK | IO | VDD_SPI | WPU | IE, WPU | | | | | SPICLK | O/T | GPIO29 | I/O/T |
| 26 | SPID | IO | VDD_SPI | WPU | IE, WPU | | | | | SPID | I1/O/T | GPIO30 | I/O/T |
| 27 | GPIO15 | IO | VDDPST2 | IE | IE | | | | | GPIO15 | I/O/T | GPIO15 | I/O/T |
| 28 | VDDPST2 | Power | | | | | | | | | | | |
| 29 | UOTXD | IO | VDDPST2 | | WPU | | | | | UOTXD | O | GPIO16 | I/O/T |
| 30 | UORXD | IO | VDDPST2 | | IE, WPU | | | | | UORXD | I1 | GPIO17 | I/O/T |
| 31 | SDIO_CMD | IO | VDDPST2 | WPU | IE | | | | | SDIO_CMD | I1/O/T | GPIO18 | I/O/T |
| 32 | SDIO_CLK | IO | VDDPST2 | WPU | IE | | | | | SDIO_CLK | I1 | GPIO19 | I/O/T |
| 33 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA0 | I1/O/T | GPIO20 | I/O/T |
| 34 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA1 | I1/O/T | GPIO21 | I/O/T |
| 35 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA2 | I1/O/T | GPIO22 | I/O/T |
| 36 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA3 | I1/O/T | GPIO23 | I/O/T |
| 37 | VDDA1 | Power | | | | | | | | | | | |
| 38 | XTAL_N | Analog | | | | | | | | | | | |
| 39 | XTAL_P | Analog | | | | | | | | | | | |
| 40 | VDDA2 | Power | | | | | | | | | | | |
| 41 | GND | Power | | | | | | | | | | | |

* For details, see Section 2 Pins. Regarding highlighted cells, see Section 2.3.4 Restrictions for GPIOs and LP GPIOs.

Table 7-2. QFN32 Pin Overview

| Pin No. | Pin Name | Pin Type | Pin Providing Power | Pin Settings | | Analog Function | | LP IO MUX Function | | IO MUX Function | | | | | |
|---------|------------|----------|---------------------|--------------|-------------|-----------------|----------|--------------------|--------------|-----------------|-------|--------|-------|---------|-------|
| | | | | At Reset | After Reset | 0 | 1 | 0 | 1 | 0 | Type | 1 | Type | 2 | Type |
| 1 | ANT | Analog | | | | | | | | | | | | | |
| 2 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 3 | VDDA3P3 | Power | | | | | | | | | | | | | |
| 4 | CHIP_PU | Analog | | | | | | | | | | | | | |
| 5 | VDDPST1 | Power | | | | | | | | | | | | | |
| 6 | XTAL_32K_P | IO | VDDPST1 | | | XTAL_32K_P | ADC1_CH0 | LP_GPIO0 | LP_UART_DTRN | GPIO0 | I/O/T | GPIO0 | I/O/T | | |
| 7 | XTAL_32K_N | IO | VDDPST1 | | | XTAL_32K_N | ADC1_CH1 | LP_GPIO1 | LP_UART_DSRN | GPIO1 | I/O/T | GPIO1 | I/O/T | | |
| 8 | GPIO2 | IO | VDDPST1 | IE | IE | | ADC1_CH2 | LP_GPIO2 | LP_UART_RTSN | GPIO2 | I/O/T | GPIO2 | I/O/T | FSPIQ | I/O/T |
| 9 | GPIO3 | IO | VDDPST1 | IE | IE | | ADC1_CH3 | LP_GPIO3 | LP_UART_CTSN | GPIO3 | I/O/T | GPIO3 | I/O/T | | |
| 10 | MTMS | IO | VDDPST1 | IE | IE | | ADC1_CH4 | LP_GPIO4 | LP_UART_RXD | MTMS | I1 | GPIO4 | I/O/T | FSPIHD | I/O/T |
| 11 | MTDI | IO | VDDPST1 | IE | IE | | ADC1_CH5 | LP_GPIO5 | LP_UART_TXD | MTDI | I1 | GPIO5 | I/O/T | FSPIWP | I/O/T |
| 12 | MTCK | IO | VDDPST1 | | IE, WPU | | ADC1_CH6 | LP_GPIO6 | LP_I2C_SDA | MTCK | I1 | GPIO6 | I/O/T | FSPICLK | I/O/T |
| 13 | MTDO | IO | VDDPST1 | | IE | | | LP_GPIO7 | LP_I2C_SCL | MTDO | O/T | GPIO7 | I/O/T | FSPID | I/O/T |
| 14 | GPIO8 | IO | VDDPST2 | IE | IE | | | | | GPIO8 | I/O/T | GPIO8 | I/O/T | | |
| 15 | GPIO9 | IO | VDDPST2 | IE, WPU | IE, WPU | | | | | GPIO9 | I/O/T | GPIO9 | I/O/T | | |
| 16 | GPIO12 | IO | VDDPST2 | | IE | USB_D- | | | | GPIO12 | I/O/T | GPIO12 | I/O/T | | |
| 17 | GPIO13 | IO | VDDPST2 | | IE, WPU | USB_D+ | | | | GPIO13 | I/O/T | GPIO13 | I/O/T | | |
| 18 | GPIO14 | IO | VDDPST2 | | IE | | | | | GPIO14 | I/O/T | GPIO14 | I/O/T | | |
| 19 | GPIO15 | IO | VDDPST2 | IE | IE | | | | | GPIO15 | I/O/T | GPIO15 | I/O/T | | |
| 20 | VDDPST2 | Power | | | | | | | | | | | | | |
| 21 | UOTXD | IO | VDDPST2 | | WPU | | | | | UOTXD | O | GPIO16 | I/O/T | FSPICS0 | I/O/T |
| 22 | UORXD | IO | VDDPST2 | | IE, WPU | | | | | UORXD | I1 | GPIO17 | I/O/T | FSPICS1 | O/T |
| 23 | SDIO_CMD | IO | VDDPST2 | WPU | IE | | | | | SDIO_CMD | I/O/T | GPIO18 | I/O/T | FSPICS2 | O/T |
| 24 | SDIO_CLK | IO | VDDPST2 | WPU | IE | | | | | SDIO_CLK | I1 | GPIO19 | I/O/T | FSPICS3 | O/T |
| 25 | SDIO_DATA0 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA0 | I/O/T | GPIO20 | I/O/T | FSPICS4 | O/T |
| 26 | SDIO_DATA1 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA1 | I/O/T | GPIO21 | I/O/T | FSPICS5 | O/T |
| 27 | SDIO_DATA2 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA2 | I/O/T | GPIO22 | I/O/T | | |
| 28 | SDIO_DATA3 | IO | VDDPST2 | WPU | IE | | | | | SDIO_DATA3 | I/O/T | GPIO23 | I/O/T | | |
| 29 | VDDA1 | Power | | | | | | | | | | | | | |
| 30 | XTAL_N | Analog | | | | | | | | | | | | | |
| 31 | XTAL_P | Analog | | | | | | | | | | | | | |
| 32 | VDDA2 | Power | | | | | | | | | | | | | |
| 33 | GND | Power | | | | | | | | | | | | | |

* For details, see Section 2 Pins. Regarding highlighted cells, see Section 2.3.4 Restrictions for GPIOs and LP GPIOs.

Glossary

module

A self-contained unit integrated within the chip to extend its capabilities, such as cryptographic modules, RF modules [2](#)

peripheral

A hardware component or subsystem within the chip to interface with the outside world [2](#)

off-package flash

Flash external to the chip's package [4](#), [31](#), [39](#)

in-package flash

Flash integrated directly into the chip's package, and external to the chip die [31](#)

strapping pin

A type of GPIO pin used to configure certain operational settings during the chip's power-up, and can be reconfigured as normal GPIO after the chip's reset [32](#)

eFuse parameter

A parameter stored in an electrically programmable fuse (eFuse) memory within a chip. The parameter can be set by programming EFUSE_PGM_DATA*n*_REG registers, and read by reading a register field named after the parameter [32](#)

SPI boot mode

A boot mode in which users load and execute the existing code from SPI flash [33](#)

joint download boot mode

A boot mode in which users can download code into flash via the UART or other interfaces (see Table [3-3 Chip Boot Mode Control](#) > Note), and load and execute the downloaded code from the flash or SRAM [33](#)

eFuse

A one-time programmable (OTP) memory which stores system and user parameters, such as MAC address, chip revision number, flash encryption key, etc. Value 0 indicates the default state, and value 1 indicates the eFuse has been programmed [39](#)

Related Documentation and Resources

Related Documentation

- [ESP32-C6 Technical Reference Manual](#) – Detailed information on how to use the ESP32-C6 memory and peripherals.
- [ESP32-C6 Hardware Design Guidelines](#) – Guidelines on how to integrate the ESP32-C6 into your hardware product.
- Certificates
<https://espressif.com/en/support/documents/certificates>
- *ESP32-C6 Product/Process Change Notifications (PCN)*
<https://espressif.com/en/support/documents/pcns?keys=ESP32-C6>
- *Documentation Updates and Update Notification Subscription*
<https://espressif.com/en/support/download/documents>

Developer Zone

- [ESP-IDF Programming Guide for ESP32-C6](#) – Extensive documentation for the ESP-IDF development framework.
- *ESP-IDF* and other development frameworks on GitHub.
<https://github.com/espressif>
- *ESP32 BBS Forum* – Engineer-to-Engineer (E2E) Community for Espressif products where you can post questions, share knowledge, explore ideas, and help solve problems with fellow engineers.
<https://esp32.com/>
- *The ESP Journal* – Best Practices, Articles, and Notes from Espressif folks.
<https://blog.espressif.com/>
- See the tabs *SDKs and Demos*, *Apps*, *Tools*, *AT Firmware*.
<https://espressif.com/en/support/download/sdks-demos>

Products

- *ESP32-C6 Series SoCs* – Browse through all ESP32-C6 SoCs.
<https://espressif.com/en/products/socs?id=ESP32-C6>
- *ESP32-C6 Series Modules* – Browse through all ESP32-C6-based modules.
<https://espressif.com/en/products/modules?id=ESP32-C6>
- *ESP32-C6 Series DevKits* – Browse through all ESP32-C6-based devkits.
<https://espressif.com/en/products/devkits?id=ESP32-C6>
- *ESP Product Selector* – Find an Espressif hardware product suitable for your needs by comparing or applying filters.
<https://products.espressif.com/#/product-selector?language=en>

Contact Us

- See the tabs *Sales Questions*, *Technical Enquiries*, *Circuit Schematic & PCB Design Review*, *Get Samples* (Online stores), *Become Our Supplier*, *Comments & Suggestions*.
<https://espressif.com/en/contact-us/sales-questions>

Revision History

| Date | Version | Release notes |
|------------|---------|---|
| 2025-03-21 | v1.3 | <ul style="list-style-type: none"> Updated CPU CoreMark® score in Section Product Overview Added Section 2.3.5 Peripheral Pin Assignment According to AR2024-011, removed "32 kHz internal slow RC oscillator" in Section 4.1.3.3 Clock |
| 2024-08-23 | v1.2 | Added the ESP32-C6FH8 variant |
| 2024-05-10 | v1.1 | <ul style="list-style-type: none"> Updated CPU CoreMark® score in Section Product Overview Added flash erase cycles, retention time, maximum clock frequency in Section 4.1.2.1 Internal Memory Updated cumulative IO output current in Table 5-1 Absolute Maximum Ratings Updated the value of R_{SPI} in Table 5-3 VDD_SPI Internal and Output Characteristics Added links and descriptions of PCB land pattern in Section 7 Packaging Added Section Glossary Improved the formatting, structure, and wording in the following sections: <ul style="list-style-type: none"> Section 2 Pins Section 3 Boot Configurations (used to be named as "Strapping Pins") Section 4 Functional Description Other minor updates |
| 2023-07-25 | v1.0 | <ul style="list-style-type: none"> Added descriptions of USB_PU in Table 2-4 QFN40 IO MUX Pin Functions and Table 2-5 QFN32 IO MUX Pin Functions, note 4 Updated Section 3.3 ROM Messages Printing Control Added Section 5.5 ADC Characteristics Updated the measurement conditions in Table 5-8 Current Consumption for Bluetooth LE in Active Mode and Table 5-9 Current Consumption for 802.15.4 in Active Mode from -24.0 dBm to -15.0 dBm, and the corresponding peak values Added Section 5.7 Reliability Updated the minimum value of RF transmit power range to -15.0 dBm in Table 6-7 Bluetooth LE RF Characteristics and Table 6-17 802.15.4 Transmitter Characteristics - 250 Kbps Updated Related Documentation and Resources Other minor changes |

Cont'd on next page

Cont'd from previous page

| Date | Version | Release notes |
|------------|---------|---------------------|
| 2023-01-16 | v0.5 | Preliminary release |



Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice.

ALL THIRD PARTY'S INFORMATION IN THIS DOCUMENT IS PROVIDED AS IS WITH NO WARRANTIES TO ITS AUTHENTICITY AND ACCURACY.

NO WARRANTY IS PROVIDED TO THIS DOCUMENT FOR ITS MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, NOR DOES ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member logo is a trademark of the Wi-Fi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

Copyright © 2025 Espressif Systems (Shanghai) Co., Ltd. All rights reserved.

www.espressif.com



BOSCH

BME280

Combined humidity and pressure sensor



BME280 – Data sheet

| | |
|-------------------------|---|
| Document revision | 1.24 |
| Document release date | February 2024 |
| Document number | BST-BME280-DS001-24 |
| Sales Part Number (SPN) | 0 273 141 185 |
| Notes | Data and descriptions in this document are subject to change without notice. Product photos and pictures are for illustration purposes only and may differ from the real product appearance |

BME280

Digital humidity, pressure and temperature sensor

Key features

- | | |
|---|---|
| • Package | 2.5 mm x 2.5 mm x 0.93 mm metal lid LGA |
| • Digital interface | I ² C (up to 3.4 MHz) and SPI (3 and 4 wire, up to 10 MHz) |
| • Supply voltage | V _{DD} main supply voltage range: 1.71 V to 3.6 V |
| • Current consumption | V _{DDIO} interface voltage range: 1.2 V to 3.6 V 1.8 µA @ 1 Hz humidity and temperature 2.8 µA @ 1 Hz pressure and temperature 3.6 µA @ 1 Hz humidity, pressure and temperature 0.1 µA in sleep mode |
| • Operating range | -40...+85 °C, 0...100 % rel. humidity, 300...1100 hPa |
| • Humidity sensor and pressure sensor can be independently enabled / disabled | |
| • Register and performance compatible to Bosch Sensortec BMP280 digital pressure sensor | |
| • RoHS compliant, halogen-free, MSL1 | |

Key parameters for humidity sensor

- | | |
|-----------------------------------|------------------------|
| • Response time ($\tau_{63\%}$) | 1 s |
| • Accuracy tolerance | ±3 % relative humidity |
| • Hysteresis | ±1% relative humidity |

Key parameters for pressure sensor

- | | |
|----------------------------------|--|
| • RMS Noise | 0.2 Pa, equiv. to 1.7 cm |
| • Offset temperature coefficient | ±1.5 Pa/K, equiv. to ±12.6 cm at 1 °C temperature change |

Typical application

- Context awareness, e.g. skin detection, room change detection
- Fitness monitoring / well-being
 - Warning regarding dryness or high temperatures
 - Measurement of volume and air flow
- Home automation control
 - control heating, venting, air conditioning (HVAC)
- Internet of things
- GPS enhancement (e.g. time-to-first-fix improvement, dead reckoning, slope detection)
- Indoor navigation (change of floor detection, elevator detection)
- Outdoor navigation, leisure and sports applications
- Weather forecast
- Vertical velocity indication (rise/sink speed)

Target devices

- Handsets such as mobile phones, tablet PCs, GPS devices
- Navigation systems
- Gaming, e.g flying toys
- Camera (DSC, video)
- Home weather stations
- Flying toys
- Watches

General Description

The BME280 is a combined digital humidity, pressure and temperature sensor based on proven sensing principles. The sensor module is housed in an extremely compact metal-lid LGA package with a footprint of only $2.5 \times 2.5 \text{ mm}^2$ with a height of 0.93 mm. Its small dimensions and its low power consumption allow the implementation in battery driven devices such as handsets, GPS modules or watches. The BME280 is register and performance compatible to the Bosch Sensortec BMP280 digital pressure sensor (see chapter 5.2 for details).

The BME280 achieves high performance in all applications requiring humidity and pressure measurement. These emerging applications of home automation control, in-door navigation, fitness as well as GPS refinement require a high accuracy and a low TCO at the same time.

The humidity sensor provides an extremely fast response time for fast context awareness applications and high overall accuracy over a wide temperature range.

The pressure sensor is an absolute barometric pressure sensor with extremely high accuracy and resolution and drastically lower noise than the Bosch Sensortec BMP180.

The integrated temperature sensor has been optimized for lowest noise and highest resolution. Its output is used for temperature compensation of the pressure and humidity sensors and can also be used for estimation of the ambient temperature.

The sensor provides both SPI and I²C interfaces and can be supplied using 1.71 to 3.6 V for the sensor supply V_{DD} and 1.2 to 3.6 V for the interface supply V_{DDIO} . Measurements can be triggered by the host or performed in regular intervals. When the sensor is disabled, current consumption drops to 0.1 μA .

BME280 can be operated in three power modes (see chapter 3.3):

- sleep mode
- normal mode
- forced mode

In order to tailor data rate, noise, response time and current consumption to the needs of the user, a variety of oversampling modes, filter modes and data rates can be selected.

Please contact your regional Bosch Sensortec partner for more information about software packages.

Index of Contents

| | |
|--|-----------|
| 1. Specification | 8 |
| 1.1 General electrical specification..... | 8 |
| 1.2 Humidity parameter specification | 9 |
| 1.3 Pressure sensor specification..... | 10 |
| 1.4 Temperature sensor specification..... | 11 |
| 2. Absolute maximum ratings | 13 |
| 3. Functional description | 14 |
| 3.1 Block diagram | 14 |
| 3.2 Power management | 14 |
| 3.3 Sensor modes..... | 14 |
| 3.3.1 Sensor mode transitions | 15 |
| 3.3.2 Sleep mode..... | 15 |
| 3.3.3 Forced mode..... | 15 |
| 3.3.4 Normal mode | 16 |
| 3.4 Measurement flow | 17 |
| 3.4.1 Humidity measurement..... | 17 |
| 3.4.2 Pressure measurement | 17 |
| 3.4.3 Temperature measurement | 17 |
| 3.4.4 IIR filter..... | 18 |
| 3.5 Recommended modes of operation | 19 |
| 3.5.1 Weather monitoring | 19 |
| 3.5.2 Humidity sensing | 19 |
| 3.5.3 Indoor navigation | 20 |
| 3.5.4 Gaming | 20 |
| 3.6 Noise..... | 21 |
| 4. Data readout..... | 23 |
| 4.1 Data register shadowing..... | 23 |

| | |
|--|-----------|
| 4.2 Output compensation | 23 |
| 4.2.1 Computational requirements | 23 |
| 4.2.2 Trimming parameter readout | 24 |
| 4.2.3 Compensation formulas | 25 |
| 5. Global memory map and register description..... | 26 |
| 5.1 General remarks | 26 |
| 5.2 Register compatibility to BMP280..... | 26 |
| 5.3 Memory map..... | 26 |
| 5.4 Register description..... | 27 |
| 5.4.1 Register 0xD0 “ <i>id</i> ”..... | 27 |
| 5.4.2 Register 0xE0 “ <i>reset</i> ”..... | 27 |
| 5.4.3 Register 0xF2 “ <i>ctrl_hum</i> ” | 27 |
| 5.4.4 Register 0xF3 “ <i>status</i> ” | 28 |
| 5.4.5 Register 0xF4 “ <i>ctrl_meas</i> ” | 28 |
| 5.4.6 Register 0xF5 “ <i>config</i> ” | 30 |
| 5.4.7 Register 0xF7...0xF9 “ <i>press</i> ” (_ <i>msb</i> , _ <i>lsb</i> , _ <i>xlsb</i>) | 30 |
| 5.4.8 Register 0xFA...0xFC “ <i>temp</i> ” (_ <i>msb</i> , _ <i>lsb</i> , _ <i>xlsb</i>) | 31 |
| 5.4.9 Register 0xFD...0xFE “ <i>hum</i> ” (_ <i>msb</i> , _ <i>lsb</i>) | 31 |
| 6. Digital interfaces | 32 |
| 6.1 Interface selection | 32 |
| 6.2 I ² C Interface | 32 |
| 6.2.1 I ² C write..... | 33 |
| 6.2.2 I ² C read | 33 |
| 6.3 SPI interface | 34 |
| 6.3.1 SPI write..... | 34 |
| 6.3.2 SPI read | 35 |
| 6.4 Interface parameter specification | 35 |
| 6.4.1 General interface parameters..... | 35 |
| 6.4.2 I ² C timings..... | 35 |
| 6.4.3 SPI timings..... | 36 |

| | |
|---|-----------|
| 7. Pin-out and connection diagram | 38 |
| 7.1 Pin-out | 38 |
| 7.2 Connection diagram I ² C..... | 39 |
| 7.3 Connection diagram 4-wire SPI..... | 40 |
| 7.4 Connection diagram 3-wire SPI..... | 41 |
| 7.5 Package dimensions | 42 |
| 7.6 Landing pattern recommendation..... | 43 |
| 7.7 Marking..... | 44 |
| 7.7.1 Mass production devices | 44 |
| 7.7.2 Engineering samples | 44 |
| 7.8 Soldering guidelines and reconditioning recommendations..... | 45 |
| 7.9 Reconditioning Procedure | 46 |
| 7.10 Tape and reel specification..... | 46 |
| 7.10.1 Dimensions | 46 |
| 7.10.2 Orientation within the reel | 47 |
| 7.11 Mounting and assembly recommendations..... | 48 |
| 7.12 Environmental safety | 48 |
| 7.12.1 RoHS | 48 |
| 7.12.2 Halogen content..... | 48 |
| 7.12.3 Internal package structure | 48 |
| 8. Appendix A: Alternative compensation formulas | 49 |
| 8.1 Compensation formulas in double precision floating point..... | 49 |
| 8.2 Pressure compensation in 32 bit fixed point..... | 50 |
| 9. Appendix B: Measurement time and current calculation..... | 51 |
| 9.1 Measurement time | 51 |
| 9.2 Measurement rate in forced mode | 51 |
| 9.3 Measurement rate in normal mode..... | 51 |
| 9.4 Response time using IIR filter..... | 52 |

| | |
|--|-----------|
| 9.5 Current consumption | 52 |
| 10. Self test..... | 53 |
| 10.1 Self-test flow | 53 |
| 10.2 Function return codes..... | 54 |
| 10.3 Usage | 55 |
| 10.3.1 File and function pointer integration | 55 |
| 10.3.2 Function call..... | 55 |
| 10.3.3 Test time and interface requirements | 55 |
| 10.4 Function explanation | 56 |
| 10.4.1 Communication test..... | 56 |
| 10.4.2 Bond wire test..... | 56 |
| 10.4.3 Measurement plausibility test | 56 |
| 10.5 Sample read, write and delay function | 57 |
| 11. Legal disclaimer | 58 |
| 11.1 Engineering samples | 58 |
| 11.2 Product use..... | 58 |
| 11.3 Application examples and hints..... | 58 |
| 12. Document history and modification..... | 59 |

1. Specification

If not stated otherwise,

- All values are valid over the full voltage range
- All minimum/maximum values are given for the full accuracy temperature range
- Minimum/maximum values of drifts, offsets and temperature coefficients are $\pm 3\sigma$ values over lifetime
- Typical values of drifts, offsets and temperature coefficients are $\pm 1\sigma$ values over lifetime
- Typical values of currents and state machine timings are determined at 25 °C
- Minimum/maximum values of currents are determined using corner lots over complete temperature range
- Minimum/maximum values of state machine timings are determined using corner lots over 0...+65 °C temperature range

The specification tables are split into humidity, pressure, and temperature part of BME280.

1.1 General electrical specification

Table 1: Electrical parameter specification

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|--|-----------------------|--|------|-----|----------|------------|
| Supply Voltage Internal Domains | V _{DD} | ripple max. 50 mVpp | 1.71 | 1.8 | 3.6 | V |
| Supply Voltage I/O Domain | V _{DDIO} | | 1.2 | 1.8 | 3.6 | V |
| Sleep current | I _{DDSL} | | | 0.1 | 0.3 | µA |
| Standby current (inactive period of normal mode) | I _{DDSB} | | | 0.2 | 0.5 | µA |
| Current during humidity measurement | I _{DDH} | Max value at 85 °C | | 340 | | µA |
| Current during pressure measurement | I _{DDP} | Max value at -40 °C | | 714 | | µA |
| Current during temperature measurement | I _{DDT} | Max value at 85 °C | | 350 | | µA |
| Start-up time | t _{startup} | Time to first communication after both V _{DD} > 1.58 V and V _{DDIO} > 0.65 V | | | 2 | ms |
| Power supply rejection ratio (DC) | PSRR | full V _{DD} range | | | ±0.01 ±5 | %RH/V Pa/V |
| Standby time accuracy | Δt _{standby} | | | ±5 | ±25 | % |

1.2 Humidity parameter specification

Table 2: Humidity parameter specification

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|---|-------------------|--|-----|-------|-----|--------------|
| Operating range ¹ | R_H | For temperatures < 0 °C and > 60 °C see Figure 1 | -40 | 25 | 85 | °C |
| | | | 0 | | 100 | %RH |
| Supply current | $I_{DD,H}$ | 1 Hz forced mode, humidity and temperature | | 1.8 | 2.8 | µA |
| Absolute accuracy tolerance | A_H | 20...80 %RH, 25 °C, including hysteresis | | ±3 | | %RH |
| Hysteresis ² | H_H | 10→90→10 %RH, 25 °C | | ±1 | | %RH |
| Nonlinearity ³ | NL_H | 10→90 %RH, 25 °C | | 1 | | %RH |
| Response time to complete 63% of step ⁴ | $\tau_{63\%}$ | 90→0 or 0→90 %RH, 25°C | | 1 | | s |
| Resolution | R_H | | | 0.008 | | %RH |
| Noise in humidity (RMS) | N_H | Highest oversampling, see chapter 3.6 | | 0.02 | | %RH |
| Long term stability | ΔH_{stab} | 10...90 %RH, 25 °C | | 0.5 | | %RH/ year |

¹ When exceeding the operating range (e.g. for soldering), humidity sensing performance is temporarily degraded and reconditioning is recommended as described in section 7.8. Operating range only for non-condensing environment.

² For hysteresis measurement the sequence 10→30→50→70→90→70→50→30→10 %RH is used. The hysteresis is defined as the difference between measurements of the humidity up / down branch and the averaged curve of both branches

³ Non-linear contributions to the sensor data are corrected during the calculation of the relative humidity by the compensation formulas described in section 4.2.3.

⁴ The air-flow in direction to the vent-hole of the device has to be dimensioned in a way that a sufficient air exchange inside to outside will be possible. To observe effects on the response time-scale of the device an air-flow velocity of approx. 1 m/s is needed.

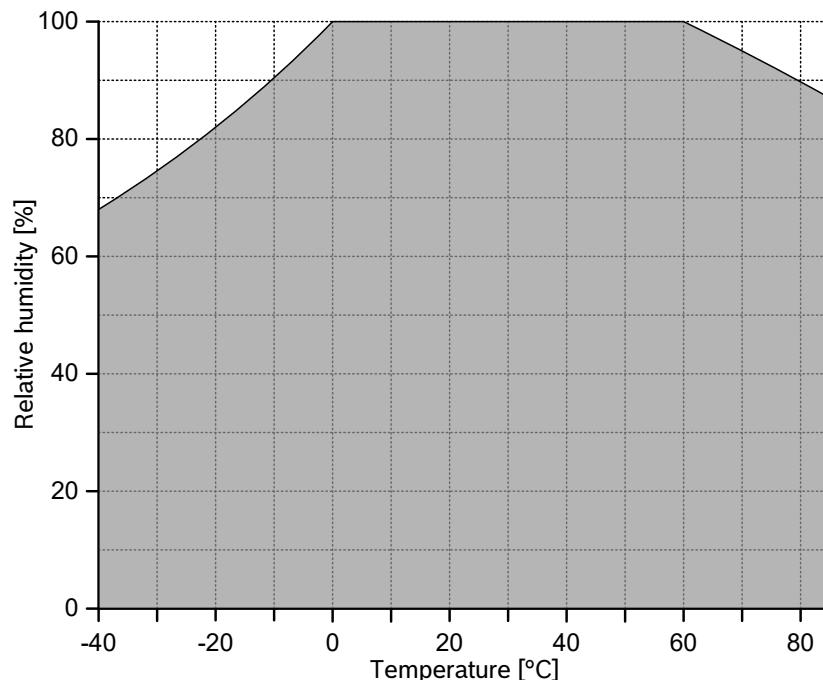


Figure 1: humidity sensor operating range

1.3 Pressure sensor specification

Table 3: Pressure parameter specification

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|-----------------------------|-------------|--|-----|-----|------|------|
| Operating temperature range | T_A | operational | -40 | 25 | +85 | °C |
| | | full accuracy | 0 | | +65 | |
| Operating pressure range | P | full accuracy | 300 | | 1100 | hPa |
| Supply current | $I_{DD,LP}$ | 1 Hz forced mode, pressure and temperature, lowest power | | 2.8 | 4.2 | µA |

| | | | | | | |
|--|-------------------------------|---|------|-------|------|------|
| Temperature coefficient of offset ⁵ | TCOP | 25...65 °C, 900 hPa | | ±1.5 | | Pa/K |
| | | | | ±12.6 | | cm/K |
| Absolute accuracy pressure | A ^P _{ext} | 300...1100 hPa -20...0 °C | | ±1.7 | | hPa |
| | A _{P,full} | 300...1100 hPa 0...65 °C | | ±1.0 | | hPa |
| | A ^P | 1100...1250 hPa 25...40 °C | | ±1.5 | | hPa |
| Relative accuracy pressure $V_{DD} = 3.3V$ | A _{rel} | 700...900hPa 25...40 °C | | ±0.12 | | hPa |
| Resolution of pressure output data | R _P | Highest oversampling | | 0.18 | | Pa |
| Noise in pressure | N _{P,fullBW} | Full bandwidth, highest oversampling See chapter 3.6 | | 1.3 | | Pa |
| | | | | 11 | | cm |
| | N _{P,filtered} | Reduced bandwidth, highest oversampling See chapter 3.6 | | 0.2 | | Pa |
| | | | | 1.7 | | cm |
| Solder drift | | Minimum solder height 50µm | -0.5 | | +2.0 | hPa |
| Long term stability ⁶ | ΔP _{stab} | per year | | ±1.0 | | hPa |
| Possible sampling rate | f _{sample_P} | Lowest oversampling, see chapter 9.2 | 157 | 182 | | Hz |

1.4 Temperature sensor specification

Table 4: Temperature parameter specification

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|-----------------|-------------------|--|-----|------|-----|------|
| Operating range | T | Operational | -40 | 25 | 85 | °C |
| | | Full accuracy | 0 | | 65 | °C |
| Supply current | I _{DD,T} | 1 Hz forced mode, temperature measurement only | | 1.0 | | µA |
| | | | | | | |
| | A _{T,25} | 25 °C | | ±0.5 | | °C |

⁵ When changing temperature by e.g. 10 °C at constant pressure / altitude, the measured pressure / altitude will change by (10 × TCOP).

⁶ Long term stability is specified in the full accuracy operating pressure range 0 ... 65 °C

| | | | | | | |
|---|---------------------------------|-----------------------|--|-------|--|----|
| Absolute accuracy temperature ⁷ | A _{T,full} | 0...65 °C | | ±0.5 | | °C |
| | A _{T,ext} ⁸ | -20 0 °C | | ±1.25 | | °C |
| | A _{T,ext} ⁹ | -40 ... -20 °C | | ±1.5 | | °C |
| Output resolution | R _T | API output resolution | | 0.01 | | °C |
| RMS noise | N _T | Lowest oversampling | | 0.005 | | °C |

⁷ Temperature measured by the internal temperature sensor. This temperature value depends on the PCB temperature, sensor element self-heating and ambient temperature and is typically above ambient temperature.

⁸ Target values & not guaranteed

⁹ Target values & not guaranteed

2. Absolute maximum ratings

The absolute maximum ratings are determined over complete temperature range using corner lots. The values are provided in Table 5.

Table 5: Absolute maximum ratings

| Parameter | Condition | Min | Max | Unit |
|------------------------------|-----------------------------|------|------------------|------|
| Voltage at any supply pin | V_{DD} and V_{DDIO} pin | -0.3 | 4.25 | V |
| Voltage at any interface pin | | -0.3 | $V_{DDIO} + 0.3$ | V |
| Storage temperature | $\leq 65\%$ RH | -45 | +85 | °C |
| Pressure | | 0 | 20 000 | hPa |
| ESD | HBM, at any pin | | ± 2 | kV |
| | CDM | | ± 500 | V |
| | Machine model | | ± 200 | V |

3. Functional description

3.1 Block diagram

Figure 2 shows a simplified block diagram of the BME280:

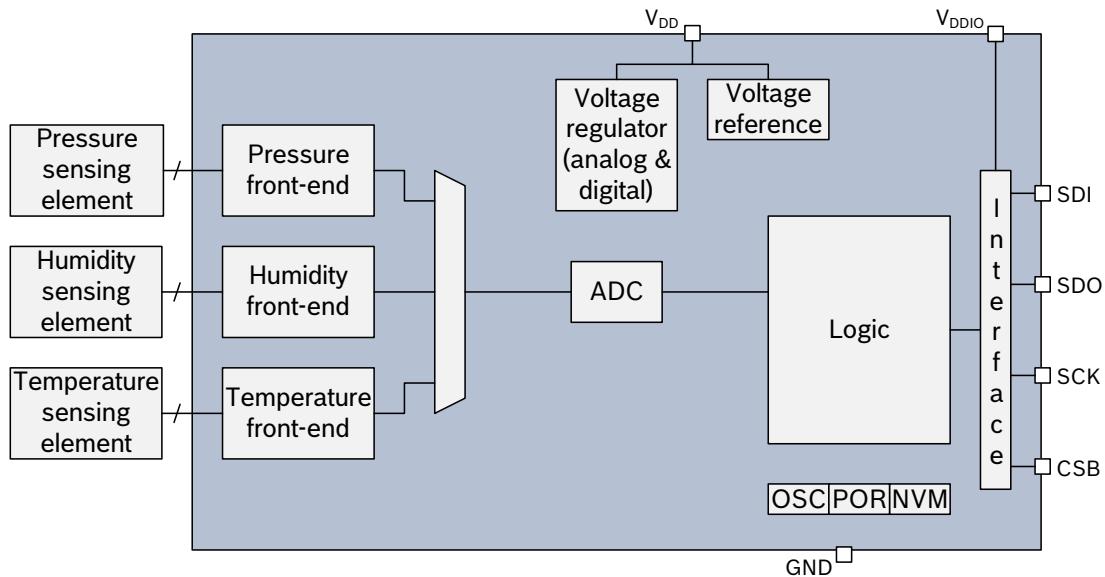


Figure 2: Block diagram of BME280

3.2 Power management

The BME280 has two distinct power supply pins

- V_{DD} is the main power supply for all internal analog and digital functional blocks
- V_{DDIO} is a separate power supply pin used for the supply of the digital interface

A power-on reset (POR) generator is built in; it resets the logic part and the register values after both V_{DD} and V_{DDIO} reach their minimum levels. There are no limitations on slope and sequence of raising the V_{DD} and V_{DDIO} levels. After powering up, the sensor settles in sleep mode (described in chapter 3.3.2).

It is prohibited to keep any interface pin (SDI, SDO, SCK or CSB) at a logical high level when V_{DDIO} is switched off. Such a configuration can permanently damage the device due an excessive current flow through the ESD protection diodes.

If V_{DDIO} is supplied, but V_{DD} is not, the interface pins are kept at a high-Z level. The bus can therefore already be used freely before the BME280 V_{DD} supply is established.

Resetting the sensor is possible by cycling V_{DD} level or by writing a soft reset command. Cycling the V_{DDIO} level will not cause a reset.

3.3 Sensor modes

The BME280 offers three sensor modes: sleep mode, forced mode and normal mode. These can be selected using the *mode[1:0]* setting (see chapter 5.4.5). The available modes are:

- Sleep mode: no operation, all registers accessible, lowest power, selected after startup
- Forced mode: perform one measurement, store results and return to sleep mode
- Normal mode: perpetual cycling of measurements and inactive periods.

The modes will be explained in detail in chapters 3.3.2 (sleep mode), 3.3.3 (forced mode) and 3.3.4 (normal mode).

3.3.1 Sensor mode transitions

The supported mode transitions are shown in Figure 3. If the device is currently performing a measurement, execution of mode switching commands is delayed until the end of the currently running measurement period. Further mode change commands or other write commands to the register `ctrl_hum` are ignored until the mode change command has been executed. Mode transitions other than the ones shown below are tested for stability but do not represent recommended use of the device.

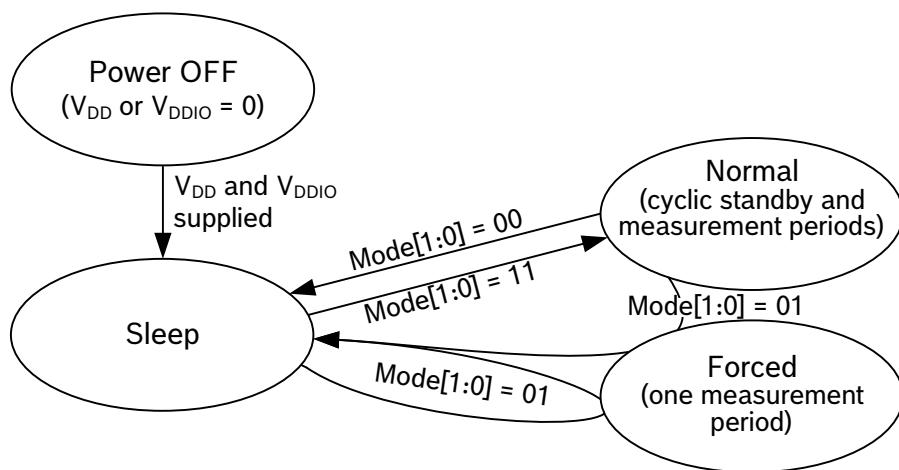


Figure 3: Sensor mode transition diagram

3.3.2 Sleep mode

Sleep mode is entered by default after power on reset. In sleep mode, no measurements are performed and power consumption (I_{DDSM}) is at a minimum. All registers are accessible; Chip-ID and compensation coefficients can be read. There are no special restrictions on interface timings.

3.3.3 Forced mode

In forced mode, a single measurement is performed in accordance to the selected measurement and filter options. When the measurement is finished, the sensor returns to sleep mode and the measurement results can be obtained from the data registers. For a next measurement, forced mode needs to be selected again. This is similar to BMP180 operation. Using forced mode is recommended for applications which require low sampling rate or host-based synchronization. The timing diagram is shown below.

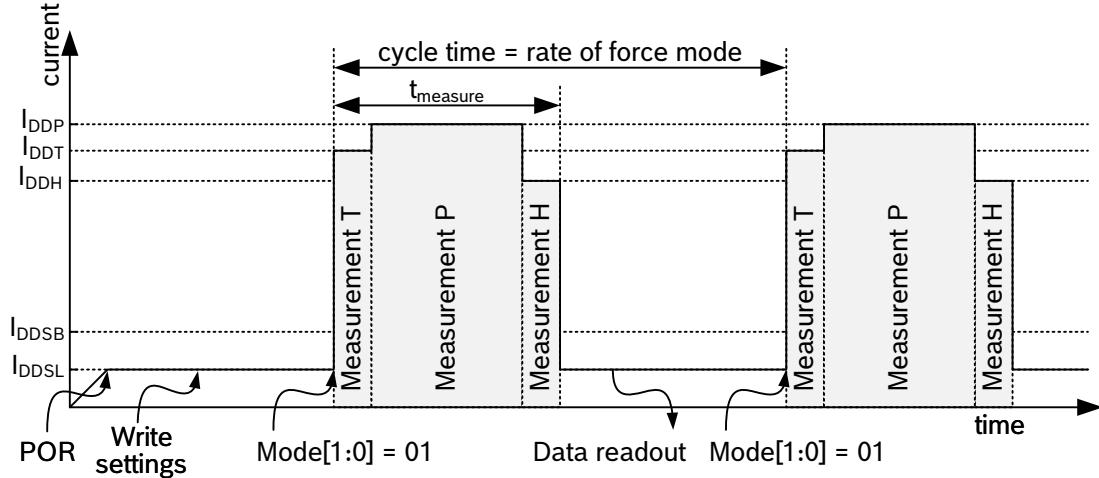


Figure 4: Forced mode timing diagram

3.3.4 Normal mode

Normal mode comprises an automated perpetual cycling between an (active) measurement period and an (inactive) standby period.

The measurements are performed in accordance to the selected measurement and filter options. The standby time is determined by the setting $t_{sb}[2:0]$ and can be set to between 0.5 and 1000 ms according to Table 27.

The total cycle time depends on the sum of the active time (see chapter 9) and standby time $t_{standby}$. The current in the standby period (I_{DDSB}) is slightly higher than in sleep mode. After setting the measurement and filter options and enabling normal mode, the last measurement results can always be obtained at the data registers without the need of further write accesses.

Using normal mode is recommended when using the IIR filter. This is useful for applications in which short-term disturbances (e.g. blowing into the sensor) should be filtered. The timing diagram is shown below:

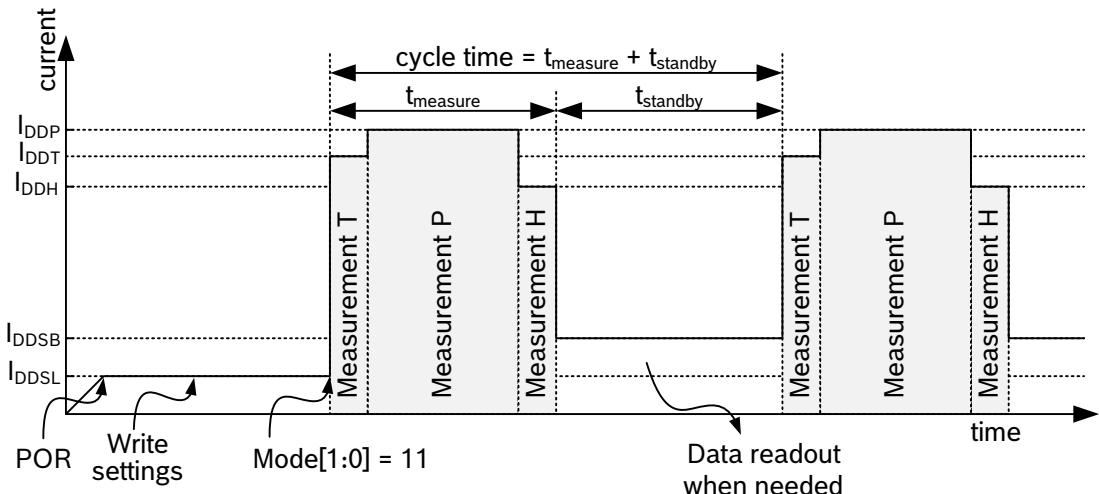


Figure 5: Normal mode timing diagram

3.4 Measurement flow

The BME280 measurement period consists of a temperature, pressure and humidity measurement with selectable oversampling. After the measurement period, the pressure and temperature data can be passed through an optional IIR filter, which removes short-term fluctuations in pressure (e.g. caused by slamming a door). For humidity, such a filter is not needed and has not been implemented. The flow is depicted in the diagram below.

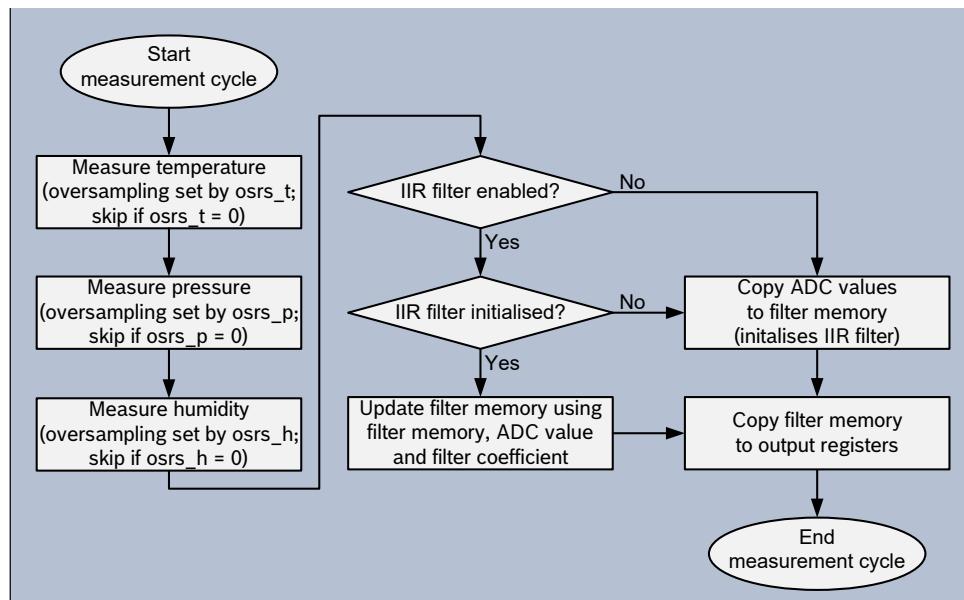


Figure 6: BME280 measurement cycle

The individual blocks of the diagram above will be detailed in the following subchapters.

3.4.1 Humidity measurement

The humidity measurement can be enabled or skipped. When enabled, several oversampling options exist. The humidity measurement is controlled by the `osrs_h[2:0]` setting, which is detailed in chapter 5.4.3. For the humidity measurement, oversampling is possible to reduce the noise. The resolution of the humidity measurement is fixed at 16 bit ADC output.

3.4.2 Pressure measurement

Pressure measurement can be enabled or skipped. When enabled, several oversampling options exist. The pressure measurement is controlled by the `osrs_p[2:0]` setting which is detailed in chapter 5.4.5. For the pressure measurement, oversampling is possible to reduce the noise. The resolution of the pressure data depends on the IIR filter (see chapter 3.4.4) and the oversampling setting (see chapter 5.4.5):

- When the IIR filter is enabled, the pressure resolution is 20 bit.
- When the IIR filter is disabled, the pressure resolution is $16 + (\text{osrs}_p - 1)$ bit, e.g. 18 bit when `osrs_p` is set to '3'.

3.4.3 Temperature measurement

Temperature measurement can be enabled or skipped. Skipping the measurement could be useful to measure pressure extremely rapidly. When enabled, several oversampling options exist. The temperature measurement is controlled by the `osrs_t[2:0]` setting which is detailed in chapter 5.4.5. For the temperature measurement, oversampling is possible to reduce the noise.

The resolution of the temperature data depends on the IIR filter (see chapter 3.4.4) and the oversampling setting (see chapter 5.4.5):

- When the IIR filter is enabled, the temperature resolution is 20 bit.

- When the IIR filter is disabled, the temperature resolution is $16 + (\text{osrs_t} - 1)$ bit, e.g. 18 bit when osrs_t is set to '3'.

3.4.4 IIR filter

The humidity value inside the sensor does not fluctuate rapidly and does not require low pass filtering. However, the environmental pressure is subject to many short-term changes, caused e.g. by slamming of a door or window, or wind blowing into the sensor. To suppress these disturbances in the output data without causing additional interface traffic and processor work load, the BME280 features an internal IIR filter. It effectively reduces the bandwidth of the temperature and pressure output signals¹⁰ and increases the resolution of the pressure and temperature output data to 20 bit. The output of a next measurement step is filtered using the following formula:

$$\text{data_filtered} = \frac{\text{data_filtered_old} \cdot (\text{filter_coefficient} - 1) + \text{data_ADC}}{\text{filter_coefficient}}$$

`Data_filtered_old` is the data coming from the current filter memory, and `data_ADC` is the data coming from current ADC acquisition. `Data_filtered` is the new value of filter memory and the value that will be sent to the output registers.

The IIR filter can be configured to different filter coefficients, which slows down the response to the sensor inputs. Note that the response time with enabled IIR filter depends on the number of samples generated, which means that the data output rate must be known to calculate the actual response time. For register configuration, please refer to Table 28. A sample response time calculation is shown in chapter 9.4.

Table 6: *filter* settings

| Filter coefficient | Samples to reach $\geq 75\%$ of step response |
|--------------------|---|
| Filter off | 1 |
| 2 | 2 |
| 4 | 5 |
| 8 | 11 |
| 16 | 22 |

In order to find a suitable setting for *filter*, please consult chapter 3.5.

When writing to the register *filter*, the filter is reset. The next ADC values will pass through the filter unchanged and become the initial memory values for the filter. If temperature or pressure measurements are skipped, the corresponding filter memory will be kept unchanged even though the output registers are set to 0x80000. When the previously skipped measurement is re-enabled, the output will be filtered using the filter memory from the last time when the measurement was not skipped. If this is not desired, please write to the *filter* register in order to re-initialize the filter.

¹⁰ Since the BME280 does not sample continuously, filtering can suffer from signals with a frequency higher than the sampling rate of the sensor. E.g. environmental fluctuations caused by windows being opened and closed might have a frequency <5 Hz. Consequently, a sampling rate of ODR = 10 Hz is sufficient to obey the Nyquist theorem.

The step response (e.g. response to sudden change in height) of the different filter settings is displayed in Figure 7.

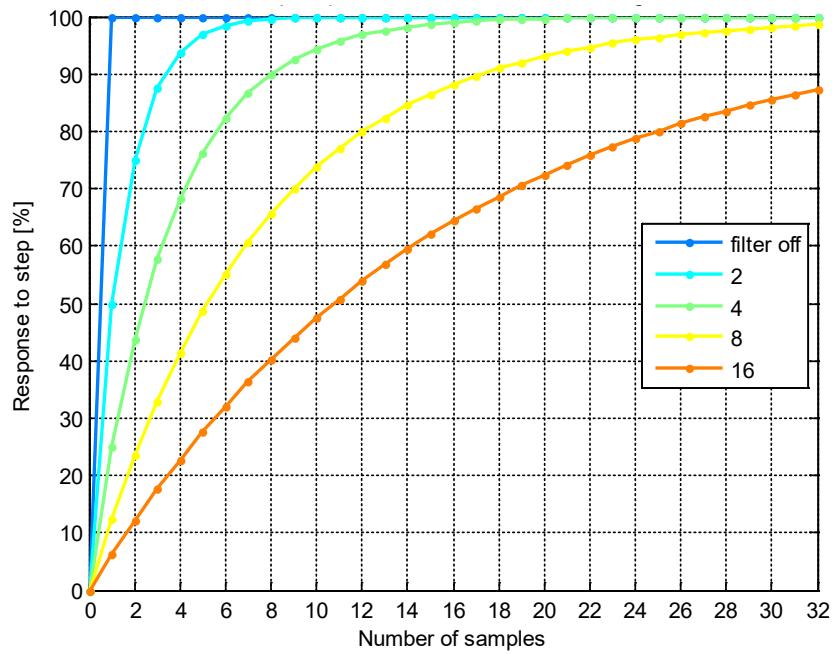


Figure 7: Step response at different IIR filter settings

3.5 Recommended modes of operation

The different oversampling options, filter settings and sensor modes result in a large number of possible settings. In this chapter, a number of settings recommended for various scenarios are presented.

3.5.1 Weather monitoring

Description: Only a very low data rate is needed. Power consumption is minimal. Noise of pressure values is of no concern. Humidity, pressure and temperature are monitored.

Table 7: Settings and performance for weather monitoring

| Suggested settings for weather monitoring | |
|---|--|
| Sensor mode | forced mode, 1 sample / minute |
| Oversampling settings | pressure ×1, temperature ×1, humidity ×1 |
| IIR filter settings | filter off |
| Performance for suggested settings | |
| Current consumption | 0.16 µA |
| RMS Noise | 3.3 Pa / 30 cm, 0.07 %RH |
| Data output rate | 1/60 Hz |

3.5.2 Humidity sensing

Description: A low data rate is needed. Power consumption is minimal. Forced mode is used to minimize power consumption and to synchronize readout, but using normal mode would also be possible.

Table 8: Settings and performance for humidity sensing

| Suggested settings for weather monitoring | |
|--|--|
| Sensor mode | forced mode, 1 sample / second |
| Oversampling settings | pressure ×0, temperature ×1, humidity ×1 |
| IIR filter settings | filter off |
| Performance for suggested settings | |
| Current consumption | 2.9 µA |
| RMS Noise | 0.07 %RH |
| Data output rate | 1 Hz |

3.5.3 Indoor navigation

Lowest possible altitude noise is needed. A very low bandwidth is preferred. Increased power consumption is tolerated. Humidity is measured to help detect room changes. This setting is suggested for the Android settings ‘SENSOR_DELAY_NORMAL’ and ‘SENSOR_DELAY_UI’.

Table 9: Settings and performance for indoor navigation

| Suggested settings for indoor navigation | |
|---|--|
| Sensor mode | normal mode, $t_{\text{standby}} = 0.5 \text{ ms}$ |
| Oversampling settings | pressure ×16, temperature ×2, humidity ×1 |
| IIR filter settings | filter coefficient 16 |
| Performance for suggested settings | |
| Current consumption | 633 µA |
| RMS Noise | 0.2 Pa / 1.7 cm |
| Data output rate | 25Hz |
| Filter bandwidth | 0.53 Hz |
| Response time (75%) | 0.9 s |

3.5.4 Gaming

Low altitude noise is needed. The required bandwidth is ~2 Hz in order to respond quickly to altitude changes (e.g. be able to dodge a flying monster in a game). Increased power consumption is tolerated. Humidity sensor is disabled. This setting is suggested for the Android settings ‘SENSOR_DELAY_GAMING’ and ‘SENSOR_DELAY_FASTEST’.

Table 10: Settings and performance for gaming

| Suggested settings for gaming | |
|---|--|
| Sensor mode | normal mode, $t_{\text{standby}} = 0.5 \text{ ms}$ |
| Oversampling settings | pressure $\times 4$, temperature $\times 1$, humidity $\times 0$ |
| IIR filter settings | filter coefficient 16 |
| Performance for suggested settings | |
| Current consumption | 581 μA |
| RMS Noise | 0.3 Pa / 2.5 cm |
| Data output rate | 83 Hz |
| Filter bandwidth | 1.75 Hz |
| Response time (75%) | 0.3 s |

3.6 Noise

The noise depends on the oversampling and, for pressure and temperature, on the filter setting used. The stated values were determined in a controlled environment and are based on the average standard deviation of 32 consecutive measurement points taken at highest sampling speed. This is needed in order to exclude long term drifts from the noise measurement. The noise depends both on humidity/pressure oversampling and temperature oversampling, since the temperature value is used for humidity/pressure temperature compensation. The oversampling combinations use below results in an optimal power to noise ratio.

Table 11: Noise and current for humidity

| Humidity / temperature oversampling setting | Typical RMS noise in humidity [%RH] at 25 °C | Typ. current [μA] at 1 Hz forced mode, 25 °C, humidity and temperature measurement, incl. I_{DDSM} |
|---|--|--|
| $\times 1 / \times 1$ | 0.07 | 1.8 |
| $\times 2 / \times 1$ | 0.05 | 2.5 |
| $\times 4 / \times 1$ | 0.04 | 3.8 |
| $\times 8 / \times 1$ | 0.03 | 6.5 |
| $\times 16 / \times 1$ | 0.02 | 11.7 |

Table 12: Noise and current for pressure

| Pressure / temperature oversampling setting | Typical RMS noise in pressure [Pa] at 25 °C | | | | | Typ. current [μ A] at 1 Hz forced mode, 25 °C, pressure and temperature measurement, incl. I_{DDSM} | |
|--|---|-----|-----|-----|-----|---|--|
| | IIR filter coefficient | | | | | | |
| | off | 2 | 4 | 8 | 16 | | |
| ×1 / ×1 | 3.3 | 1.9 | 1.2 | 0.9 | 0.4 | 2.8 | |
| ×2 / ×1 | 2.6 | 1.5 | 1.0 | 0.6 | 0.4 | 4.2 | |
| ×4 / ×1 | 2.1 | 1.2 | 0.8 | 0.5 | 0.3 | 7.1 | |
| ×8 / ×1 | 1.6 | 1.0 | 0.6 | 0.4 | 0.2 | 12.8 | |
| ×16 / ×2 | 1.3 | 0.8 | 0.5 | 0.4 | 0.2 | 24.9 | |

Table 13: Temperature dependence of pressure noise

| RMS noise at different temperatures | |
|-------------------------------------|--|
| Temperature | Typical change in noise compared to 25 °C |
| -10 °C | +25 % |
| 25 °C | ±0 % |
| 75 °C | -5 % |

Table 14: Noise in temperature

| Temperature oversampling setting | Typical RMS noise in temperature [°C] at 25 °C |
|-------------------------------------|---|
| ×1 | 0.005 |
| ×2 | 0.004 |
| ×4 | 0.003 |
| ×8 | 0.003 |
| ×16 | 0.002 |

4. Data readout

To read out data after a conversion, it is strongly recommended to use a burst read and not address every register individually. This will prevent a possible mix-up of bytes belonging to different measurements and reduce interface traffic. Note that in I²C mode, even when pressure was not measured, reading the unused registers is faster than reading temperature and humidity data separately.

Data readout is done by starting a burst read from 0xF7 to 0xFC (temperature and pressure) or from 0xF7 to 0xFE (temperature, pressure and humidity). The data are read out in an unsigned 20-bit format both for pressure and for temperature and in an unsigned 16-bit format for humidity. It is strongly recommended to use the BME280 API, available from Bosch Sensortec, for readout and compensation. For details on memory map and interfaces, please consult chapters 5 and 6 respectively.

After the uncompensated values for pressure, temperature and humidity ‘ut’, ‘up’ and ‘uh’ have been read, the actual humidity, pressure and temperature needs to be calculated using the compensation parameters stored in the device. The procedure is elaborated in chapter 4.2.

4.1 Data register shadowing

In normal mode, the timing of measurements is not necessarily synchronized to the readout by the user. This means that new measurement results may become available while the user is reading the results from the previous measurement. In this case, shadowing is performed in order to guarantee data consistency. Shadowing will only work if all data registers are read in a single burst read.

Therefore, the user must use burst reads if he does not synchronize data readout with the measurement cycle. Using several independent read commands may result in inconsistent data.

If a new measurement is finished and the data registers are still being read, the new measurement results are transferred into shadow data registers. The content of shadow registers is transferred into data registers as soon as the user ends the burst read, even if not all data registers were read.

The end of the burst read is marked by the rising edge of CSB pin in SPI case or by the recognition of a stop condition in I²C case. After the end of the burst read, all user data registers are updated at once.

4.2 Output compensation

The BME280 output consists of the ADC output values. However, each sensing element behaves differently. Therefore, the actual pressure and temperature must be calculated using a set of calibration parameters. In this chapter, the method to read out the trimming values will be given. The recommended calculation uses fixed point arithmetic and is given in chapter 4.2.3.

In high-level languages like MatlabTM or LabVIEWTM, fixed-point code may not be well supported. In this case the floating-point code in appendix 8.1 can be used as an alternative.

For 8-bit micro controllers, the variable size may be limited. In this case a simplified 32 bit integer code with reduced accuracy is given in appendix 8.2.

4.2.1 Computational requirements

In the table below an overview is given for the number of clock cycles needed for compensation on a 32 bit Cortex-M3 micro controller with GCC optimization level -O2. This controller does not feature a floating point unit, thus all floating-point calculations are emulated. Floating point is only recommended for PC application, where an FPU is present and these calculations are performed drastically faster.

Table 15: Computational requirements for compensation formulas

| Compensation of | Number of clocks (ARM Cortex-M3) | | |
|-----------------|----------------------------------|----------------|---------------------|
| | 32 bit integer | 64 bit integer | Double precision |
| Humidity | ~83 | – | ~2900 ¹¹ |
| Temperature | ~46 | – | ~2400 ¹¹ |
| Pressure | ~112 ¹² | ~1400 | ~5400 ¹¹ |

4.2.2 Trimming parameter readout

The trimming parameters are programmed into the devices' non-volatile memory (NVM) during production and cannot be altered by the customer. Each compensation word is a 16-bit signed or unsigned integer value stored in two's complement. As the memory is organized into 8-bit words, two words must always be combined in order to represent the compensation word. The 8-bit registers are named calib00...calib41 and are stored at memory addresses 0x88...0xA1 and 0xE1...0xE7. The corresponding compensation words are named dig_T# for temperature compensation related values, dig_P# for pressure related values and dig_H# for humidity related values. The mapping is seen in Table 16.

Table 16: Compensation parameter storage, naming and data type

| Register Address | Register content | Data type |
|------------------|-----------------------|----------------|
| 0x88 / 0x89 | dig_T1 [7:0] / [15:8] | unsigned short |
| 0x8A / 0x8B | dig_T2 [7:0] / [15:8] | signed short |
| 0x8C / 0x8D | dig_T3 [7:0] / [15:8] | signed short |
| 0x8E / 0x8F | dig_P1 [7:0] / [15:8] | unsigned short |
| 0x90 / 0x91 | dig_P2 [7:0] / [15:8] | signed short |
| 0x92 / 0x93 | dig_P3 [7:0] / [15:8] | signed short |
| 0x94 / 0x95 | dig_P4 [7:0] / [15:8] | signed short |
| 0x96 / 0x97 | dig_P5 [7:0] / [15:8] | signed short |
| 0x98 / 0x99 | dig_P6 [7:0] / [15:8] | signed short |
| 0x9A / 0x9B | dig_P7 [7:0] / [15:8] | signed short |
| 0x9C / 0x9D | dig_P8 [7:0] / [15:8] | signed short |
| 0x9E / 0x9F | dig_P9 [7:0] / [15:8] | signed short |
| 0xA1 | dig_H1 [7:0] | unsigned char |
| 0xE1 / 0xE2 | dig_H2 [7:0] / [15:8] | signed short |
| 0xE3 | dig_H3 [7:0] | unsigned char |
| 0xE4 / 0xE5[3:0] | dig_H4 [11:4] / [3:0] | signed short |
| 0xE5[7:4] / 0xE6 | dig_H5 [3:0] / [11:4] | signed short |

¹¹ Use only recommended for high-level programming languages like Matlab™ or LabVIEW™¹² Use only recommended for 8-bit micro controllers

0xE7

dig_H6

signed char

4.2.3 Compensation formulas

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer. Humidity is expected to be received in 16 bit format, positive, stored in a 32 bit signed integer.

The variable `t_fine` (signed 32 bit) carries a fine resolution temperature value over to the pressure and humidity compensation formula and could be implemented as a global variable.

The data type “`BME280_S32_t`” should define a 32 bit signed integer variable type and can usually be defined as “`long signed int`”.

The data type “`BME280_U32_t`” should define a 32 bit unsigned integer variable type and can usually be defined as “`long unsigned int`”.

For best possible calculation accuracy in pressure, 64 bit integer support is needed. If this is not possible on your platform, please see appendix 8.2 for a 32 bit alternative.

The data type “`BME280_S64_t`” should define a 64 bit signed integer variable type, which on most supporting platforms can be defined as “`long long signed int`”. The revision of the code is rev.1.1.

```

// Returns temperature in DegC, resolution is 0.01 DegC. Output value of "5123" equals 51.23
DegC.
// t_fine carries fine temperature as global value
BME280_S32_t t_fine;
BME280_S32_t BME280_compensate_T_int32(BME280_S32_t adc_T)
{
    BME280_S32_t var1, var2, T;
    var1 = (((adc_T>>3) - ((BME280_S32_t)dig_T1<<1)) * ((BME280_S32_t)dig_T2)) >> 11;
    var2 = (((((adc_T>>4) - ((BME280_S32_t)dig_T1)) * ((adc_T>>4) - ((BME280_S32_t)dig_T1)))>> 12) *
        ((BME280_S32_t)dig_T3)) >> 14;
    t_fine = var1 + var2;
    T = (t_fine * 5 + 128) >> 8;
    return T;
}

// Returns pressure in Pa as unsigned 32 bit integer in Q24.8 format (24 integer bits and 8
// fractional bits).
// Output value of "24674867" represents 24674867/256 = 96386.2 Pa = 963.862 hPa
BME280_U32_t BME280_compensate_P_int64(BME280_S32_t adc_P)
{
    BME280_S64_t var1, var2, p;
    var1 = ((BME280_S64_t)t_fine) - 128000;
    var2 = var1 * var1 * (BME280_S64_t)dig_P6;
    var2 = var2 + ((var1*(BME280_S64_t)dig_P5)<<17);
    var2 = var2 + (((BME280_S64_t)dig_P4)<<35);
    var1 = ((var1 * var1 * (BME280_S64_t)dig_P3)>>8) + ((var1 * (BME280_S64_t)dig_P2)<<12);
    var1 = (((((BME280_S64_t)1)<<47)+var1))*((BME280_S64_t)dig_P1)>>33;
    if (var1 == 0)
    {
        return 0; // avoid exception caused by division by zero
    }
    p = 1048576-adc_P;
    p = (((p<<31)-var2)*3125)/var1;
    var1 = (((BME280_S64_t)dig_P9) * (p>>13) * (p>>13)) >> 25;
    var2 = (((BME280_S64_t)dig_P8) * p) >> 19;
    p = ((p + var1 + var2) >> 8) + (((BME280_S64_t)dig_P7)<<4);
    return (BME280_U32_t)p;
}

// Returns humidity in %RH as unsigned 32 bit integer in Q22.10 format (22 integer and 10
// fractional bits).
// Output value of "47445" represents 47445/1024 = 46.333 %RH
BME280_U32_t bme280_compensate_H_int32(BME280_S32_t adc_H)
{
    BME280_S32_t v_x1_u32r;
    v_x1_u32r = (t_fine - ((BME280_S32_t)76800));

```

```

v_x1_u32r = (((((adc_H << 14) - ((BME280_S32_t)dig_H4) << 20) - (((BME280_S32_t)dig_H5) *
v_x1_u32r)) + ((BME280_S32_t)16384)) >> 15) * (((((v_x1_u32r * ((BME280_S32_t)dig_H6)) >> 10) +
((BME280_S32_t)32768)) >> 10) + ((BME280_S32_t)2097152)) * ((BME280_S32_t)dig_H2) +
8192) >> 14));
v_x1_u32r = (v_x1_u32r - (((v_x1_u32r >> 15) * (v_x1_u32r >> 15)) >> 7) *
((BME280_S32_t)dig_H1)) >> 4));
v_x1_u32r = (v_x1_u32r < 0 ? 0 : v_x1_u32r);
v_x1_u32r = (v_x1_u32r > 419430400 ? 419430400 : v_x1_u32r);
return (BME280_U32_t)(v_x1_u32r>>12);
}

```

5. Global memory map and register description

5.1 General remarks

The entire communication with the device is performed by reading from and writing to registers. Registers have a width of 8 bits. There are several registers which are reserved; they should not be written to and no specific value is guaranteed when they are read. For details on the interface, consult chapter 6.

5.2 Register compatibility to BMP280

The BME280 is downward register compatible to the BMP280, which means that the pressure and temperature control and readout is identical to BMP280. However, the following exceptions have to be considered:

Table 17: Register incompatibilities between BMP280 and BME280

| Register | Bits | Content | BMP280 | BME280 |
|---------------------|------|---------|---|--|
| 0xD0 “id” | 7:0 | chip_id | Read value is 0x56 / 0x57 (samples) 0x58 (mass production) | Read value is 0x60 |
| 0xF5 “config” | 7:5 | t_sb | ‘110’: 2000 ms ‘111’: 4000 ms | ‘110’: 10 ms ‘111’: 20 ms |
| 0xF7...0xF9 “press” | 19:0 | press | Resolution (16...20 bit) depends only on osrs_p | Without filter, resolution depends on osrs_p; when using filter, resolution is always 20 bit |
| 0xFA...0xFC “temp” | 19:0 | temp | Resolution (16...20 bit) only depends on osrs_t | Without filter, resolution depends on osrs_t; when using filter, resolution is always 20 bit |

5.3 Memory map

The memory map is given in Table 18 below. Reserved registers are not shown.

Table 18: Memory map

| Register Name | Address | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | Reset state |
|------------------|-------------|-------------|-----------------|------|------------------|------|-------------|--------------|------|-------------|
| hum_lsb | 0xFE | | | | hum_lsb<7:0> | | | | | 0x00 |
| hum_msb | 0xFD | | | | hum_msb<7:0> | | | | | 0x80 |
| temp_xlsb | 0xFC | | temp_xlsb<7:4> | | 0 | 0 | 0 | 0 | | 0x00 |
| temp_lsb | 0xFB | | | | temp_lsb<7:0> | | | | | 0x00 |
| temp_msb | 0xFA | | | | temp_msb<7:0> | | | | | 0x80 |
| press_xlsb | 0xF9 | | press_xlsb<7:4> | | 0 | 0 | 0 | 0 | | 0x00 |
| press_lsb | 0xF8 | | | | press_lsb<7:0> | | | | | 0x00 |
| press_msb | 0xF7 | | | | press_msb<7:0> | | | | | 0x80 |
| config | 0xF5 | t_sb[2:0] | | | filter[2:0] | | | spi3w_en[0] | | 0x00 |
| ctrl_meas | 0xF4 | osrs_t[2:0] | | | osrs_p[2:0] | | mode[1:0] | | | 0x00 |
| status | 0xF3 | | | | measuring[0] | | | im_update[0] | | 0x00 |
| ctrl_hum | 0xF2 | | | | | | osrs_h[2:0] | | | 0x00 |
| calib26..calib41 | 0xE1...0xF0 | | | | calibration data | | | | | individual |
| reset | 0xE0 | | | | reset[7:0] | | | | | 0x00 |
| id | 0xD0 | | | | chip_id[7:0] | | | | | 0x60 |
| calib00..calib25 | 0x88...0xA1 | | | | calibration data | | | | | individual |

| Registers: | Reserved registers | Calibration data | Control registers | Data registers | Status registers | Chip ID | Reset |
|------------|--------------------|------------------|-------------------|----------------|------------------|-----------|------------|
| Type: | do not change | read only | read / write | read only | read only | read only | write only |

5.4 Register description

5.4.1 Register 0xD0 “id”

The “id” register contains the chip identification number chip_id[7:0], which is 0x60. This number can be read as soon as the device finished the power-on-reset.

5.4.2 Register 0xE0 “reset”

The “reset” register contains the soft reset word reset[7:0]. If the value 0xB6 is written to the register, the device is reset using the complete power-on-reset procedure. Writing other values than 0xB6 has no effect. The readout value is always 0x00.

5.4.3 Register 0xF2 “ctrl_hum”

The “ctrl_hum” register sets the humidity data acquisition options of the device. **Changes to this register only become effective after a write operation to “ctrl_meas”.**

Table 19: Register 0xF2 “ctrl_hum”

| Register 0xF2 “ctrl_hum” | Name | Description |
|-----------------------------|-------------|--|
| Bit 2, 1, 0 | osrs_h[2:0] | Controls oversampling of humidity data. See Table 20 for settings and chapter 3.4.1 for details. |

Table 20: register settings osrs_h

| osrs_h[2:0] | Humidity oversampling |
|-------------|--------------------------------|
| 000 | Skipped (output set to 0x8000) |
| 001 | oversampling ×1 |
| 010 | oversampling ×2 |
| 011 | oversampling ×4 |
| 100 | oversampling ×8 |
| 101, others | oversampling ×16 |

5.4.4 Register 0xF3 “status”

The “status” register contains two bits which indicate the status of the device.

Table 21: Register 0xF3 “status”

| Register 0xF3 “status” | Name | Description |
|---------------------------|--------------|---|
| Bit 3 | measuring[0] | Automatically set to ‘1’ whenever a conversion is running and back to ‘0’ when the results have been transferred to the data registers. |
| Bit 0 | im_update[0] | Automatically set to ‘1’ when the NVM data are being copied to image registers and back to ‘0’ when the copying is done. The data are copied at power-on-reset and before every conversion. |

5.4.5 Register 0xF4 “ctrl_meas”

The “ctrl_meas” register sets the pressure and temperature data acquisition options of the device. The register needs to be written after changing “ctrl_hum” for the changes to become effective.

Table 22: Register 0xF4 “ctrl_meas”

| Register 0xF4 “ctrl_meas” | Name | Description |
|------------------------------|-------------|---|
| Bit 7, 6, 5 | osrs_t[2:0] | Controls oversampling of temperature data. See Table 24 for settings and chapter 3.4.3 for details. |
| Bit 4, 3, 2 | osrs_p[2:0] | Controls oversampling of pressure data. See Table 23 for settings and chapter 3.4.2 for details. |
| Bit 1, 0 | mode[1:0] | Controls the sensor mode of the device. See Table 25 for settings and chapter 3.3 for details. |

Table 23: register settings osrs_p

| osrs_p[2:0] | Pressure oversampling |
|-------------|---------------------------------|
| 000 | Skipped (output set to 0x80000) |
| 001 | oversampling ×1 |
| 010 | oversampling ×2 |
| 011 | oversampling ×4 |
| 100 | oversampling ×8 |
| 101, others | oversampling ×16 |

Table 24: register settings osrs_t

| osrs_t[2:0] | Temperature oversampling |
|-------------|---------------------------------|
| 000 | Skipped (output set to 0x80000) |
| 001 | oversampling ×1 |
| 010 | oversampling ×2 |
| 011 | oversampling ×4 |
| 100 | oversampling ×8 |
| 101, others | oversampling ×16 |

Table 25: register settings mode

| mode[1:0] | Mode |
|-----------|-------------|
| 00 | Sleep mode |
| 01 and 10 | Forced mode |
| 11 | Normal mode |

5.4.6 Register 0xF5 “config”

The “config” register sets the rate, filter and interface options of the device. Writes to the “config” register in normal mode may be ignored. In sleep mode writes are not ignored.

Table 26: Register 0xF5 “config”

| Register 0xF5 “config” | Name | Description |
|---------------------------|-------------|--|
| Bit 7, 6, 5 | t_sb[2:0] | Controls inactive duration t_{standby} in normal mode. See Table 27 for settings and chapter 3.3.4 for details. |
| Bit 4, 3, 2 | filter[2:0] | Controls the time constant of the IIR filter. See Table 27 for settings and chapter 3.4.4 for details. |
| Bit 0 | spi3w_en[0] | Enables 3-wire SPI interface when set to ‘1’. See chapter 6.3 for details. |

Table 27: t_{sb} settings

| $t_{\text{sb}}[2:0]$ | $t_{\text{standby}} [\text{ms}]$ |
|----------------------|----------------------------------|
| 000 | 0.5 |
| 001 | 62.5 |
| 010 | 125 |
| 011 | 250 |
| 100 | 500 |
| 101 | 1000 |
| 110 | 10 |
| 111 | 20 |

Table 28: filter settings

| filter[2:0] | Filter coefficient |
|-------------|--------------------|
| 000 | Filter off |
| 001 | 2 |
| 010 | 4 |
| 011 | 8 |
| 100, others | 16 |

5.4.7 Register 0xF7...0xF9 “press” (_msb, _lsb, _xlsb)

The “press” register contains the raw pressure measurement output data up[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 29: Register 0xF7 ... 0xF9 “press”

| Register 0xF7...0xF9 “press” | Name | Description |
|---------------------------------|-----------------|--|
| 0xF7 | press_msb[7:0] | Contains the MSB part up[19:12] of the raw pressure measurement output data. |
| 0xF8 | press_lsb[7:0] | Contains the LSB part up[11:4] of the raw pressure measurement output data. |
| 0xF9 (bit 7, 6, 5, 4) | press_xlsb[3:0] | Contains the XLSB part up[3:0] of the raw pressure measurement output data. Contents depend on temperature resolution. |

5.4.8 Register 0xFA...0xFC “temp” (_msb, _lsb, _xlsb)

The “temp” register contains the raw temperature measurement output data ut[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 30: Register 0xFA ... 0xFC “temp”

| Register 0xFA...0xFC “temp” | Name | Description |
|--------------------------------|----------------|--|
| 0xFA | temp_msb[7:0] | Contains the MSB part ut[19:12] of the raw temperature measurement output data. |
| 0xFB | temp_lsb[7:0] | Contains the LSB part ut[11:4] of the raw temperature measurement output data. |
| 0xFC (bit 7, 6, 5, 4) | temp_xlsb[3:0] | Contains the XLSB part ut[3:0] of the raw temperature measurement output data. Contents depend on pressure resolution. |

5.4.9 Register 0xFD...0xFE “hum” (_msb, _lsb)

The “temp” register contains the raw temperature measurement output data ut[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 31: Register 0xFD ... 0xFE “hum”

| Register 0xFD...0xFE “hum” | Name | Description |
|-------------------------------|---------------|---|
| 0xFD | hum_msb[7:0] | Contains the MSB part uh[15:8] of the raw humidity measurement output data. |
| 0xFE | temp_lsb[7:0] | Contains the LSB part uh[7:0] of the raw humidity measurement output data. |

6. Digital interfaces

The BME280 supports the I²C and SPI digital interfaces; it acts as a slave for both protocols. The I²C interface supports the Standard, Fast and High Speed modes. The SPI interface supports both SPI mode '00' (CPOL = CPHA = '0') and mode '11' (CPOL = CPHA = '1') in 4-wire and 3-wire configuration.

The following transactions are supported:

- Single byte write
- multiple byte write (using pairs of register addresses and register data)
- single byte read
- multiple byte read (using a single register address which is auto-incremented)

6.1 Interface selection

Interface selection is done automatically based on CSB (chip select) status. If CSB is connected to V_{DDIO}, the I²C interface is active. If CSB is pulled down, the SPI interface is activated. After CSB has been pulled down once (regardless of whether any clock cycle occurred), the I²C interface is disabled until the next power-on-reset. This is done in order to avoid inadvertently decoding SPI traffic to another slave as I²C data. Since the device startup is deferred until both V_{DD} and V_{DDIO} are established, there is no risk of incorrect protocol detection because of the power-up sequence used. However, if I²C is to be used and CSB is not directly connected to V_{DDIO} but is instead connected to a programmable pin, it must be ensured that this pin already outputs the V_{DDIO} level during power-on-reset of the device. If this is not the case, the device will be locked in SPI mode and not respond to I²C commands.

6.2 I²C Interface

The I²C slave interface is compatible with Philips I²C Specification version 2.1. For detailed timings, please review Table 33. All modes (standard, fast, high speed) are supported. SDA and SCL are not pure open-drain. Both pads contain ESD protection diodes to V_{DDIO} and GND. As the device does not perform clock stretching, the SCL structure is a high-Z input without drain capability.

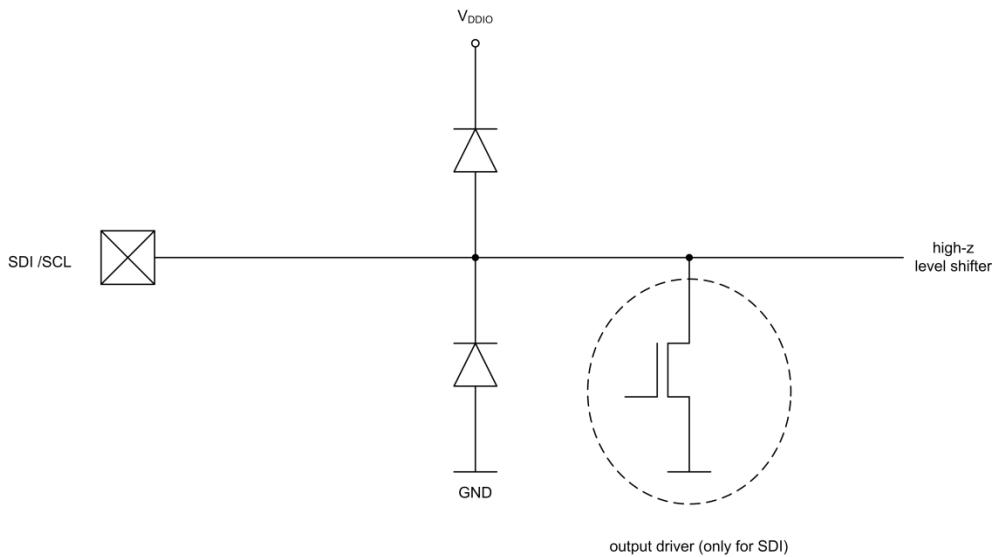


Figure 8: SDI/SCK ESD drawing

The 7-bit device address is 111011x. The 6 MSB bits are fixed. The last bit is changeable by SDO value and can be changed during operation. Connecting SDO to GND results in slave address 1110110 (0x76); connecting it to V_{DDIO} results in slave address 1110111 (0x77), which is the same as

BMP280's I²C address. The SDO pin cannot be left floating; if left floating, the I²C address will be undefined.

The I²C interface uses the following pins:

- SCK: serial clock (SCL)
- SDI: data (SDA)
- SDO: Slave address LSB (GND = '0', V_{DDIO} = '1')

CSB must be connected to V_{DDIO} to select I²C interface. SDI is bi-directional with open drain to GND: it must be externally connected to V_{DDIO} via a pull up resistor. Refer to chapter 7 for connection instructions.

The following abbreviations will be used in the I²C protocol figures:

- S Start
- P Stop
- ACKS Acknowledge by slave
- ACKM Acknowledge by master
- NACKM Not acknowledge by master

6.2.1 I²C write

Writing is done by sending the slave address in write mode (RW = '0'), resulting in slave address 111011X0 ('X' is determined by state of SDO pin. Then the master sends pairs of register addresses and register data. The transaction is ended by a stop condition. This is depicted in Figure 9.

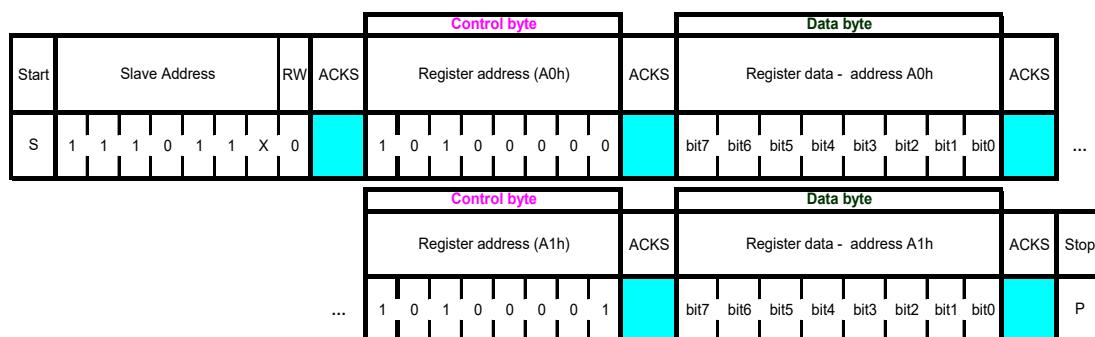


Figure 9: I²C multiple byte write (not auto-incremented)

6.2.2 I²C read

To be able to read registers, first the register address must be sent in write mode (slave address 111011X0). Then either a stop or a repeated start condition must be generated. After this the slave is addressed in read mode (RW = '1') at address 111011X1, after which the slave sends out data from auto-incremented register addresses until a NOACKM and stop condition occurs. This is depicted in Figure 10, where register 0xF6 and 0xF7 are read.

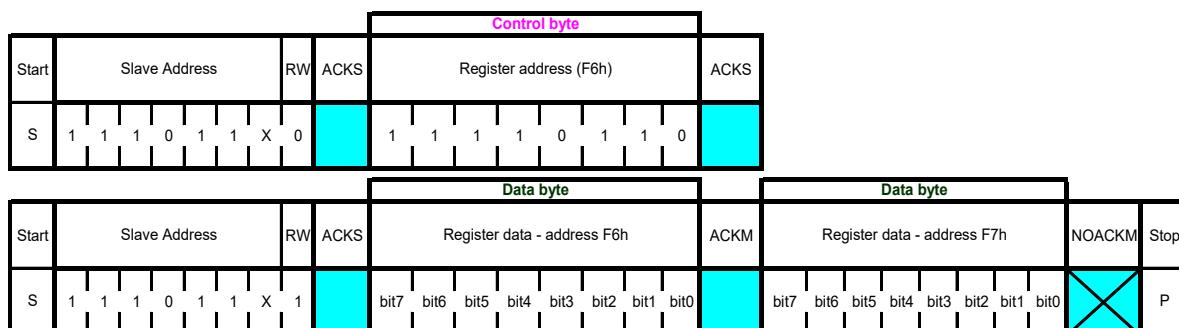


Figure 10: I²C multiple byte read

6.3 SPI interface

The SPI interface is compatible with SPI mode ‘00’ (CPOL = CPHA = ‘0’) and mode ‘11’ (CPOL = CPHA = ‘1’). The automatic selection between mode ‘00’ and ‘11’ is determined by the value of SCK after the CSB falling edge.

The SPI interface has two modes: 4-wire and 3-wire. The protocol is the same for both. The 3-wire mode is selected by setting ‘1’ to the register spi3w_en. The pad SDI is used as a data pad in 3-wire mode.

The SPI interface uses the following pins:

- CSB: chip select, active low
- SCK: serial clock
- SDI: serial data input; data input/output in 3-wire mode
- SDO: serial data output; hi-Z in 3-wire mode

Refer to chapter 7 for connection instructions.

CSB is active low and has an integrated pull-up resistor. Data on SDI is latched by the device at SCK rising edge and SDO is changed at SCK falling edge. Communication starts when CSB goes to low and stops when CSB goes to high; during these transitions on CSB, SCK must be stable. The SPI protocol is shown in Figure 11. For timing details, please review Table 34.

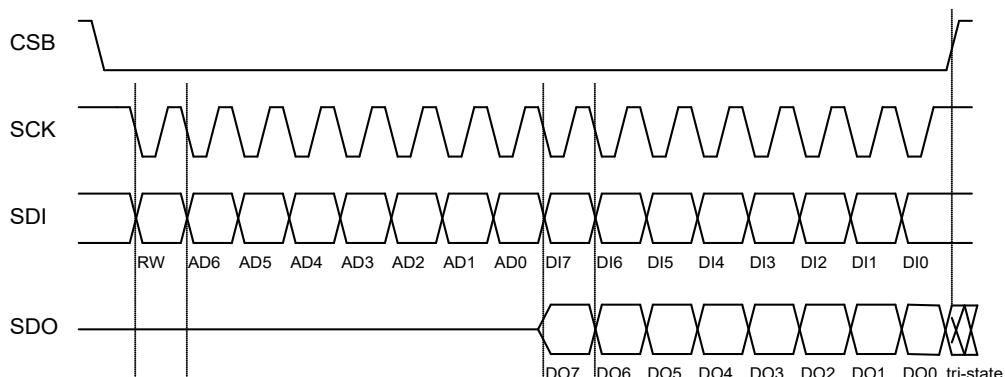


Figure 11: SPI protocol (shown for mode ‘11’ in 4-wire configuration)

In SPI mode, only 7 bits of the register addresses are used; the MSB of register address is not used and replaced by a read/write bit (RW = ‘0’ for write and RW = ‘1’ for read).

Example: address 0xF7 is accessed by using SPI register address 0x77. For write access, the byte 0x77 is transferred, for read access, the byte 0xF7 is transferred.

6.3.1 SPI write

Writing is done by lowering CSB and sending pairs control bytes and register data. The control bytes consist of the SPI register address (= full register address without bit 7) and the write command (bit7 = RW = ‘0’). Several pairs can be written without raising CSB. The transaction is ended by a raising CSB. The SPI write protocol is depicted in Figure 12.

| | | Control byte | | | | | | | | Data byte | | | | | | | | | | | | | | | | | | | | | | | |
|---------|----|------------------------|---|---|---|---|---|---|-----------------------------|-----------|------|------|------|------|------|------|------------------------|---|---|---|---|---|---|-----------------------------|------|------|------|------|------|------|------|------|---------|
| Start | RW | Register address (F4h) | | | | | | | Data register - address F4h | | | | | | | RW | Register address (F5h) | | | | | | | Data register - address F5h | | | | | | | Stop | | |
| CSB = 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | CSB = 1 |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Figure 12: SPI multiple byte write (not auto-incremented)

6.3.2 SPI read

Reading is done by lowering CSB and first sending one control byte. The control bytes consist of the SPI register address (= full register address without bit 7) and the read command (bit 7 = RW = '1'). After writing the control byte, data is sent out of the SDO pin (SDI in 3-wire mode); the register address is automatically incremented. The SPI read protocol is depicted in Figure 13.

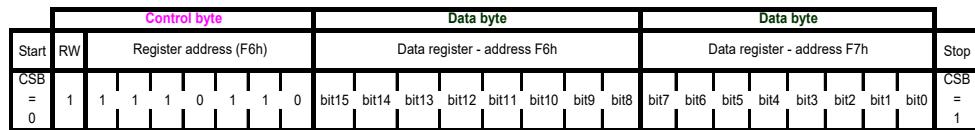


Figure 13: SPI multiple byte read

6.4 Interface parameter specification

6.4.1 General interface parameters

The general interface parameters are given in Table 32 below.

Table 32: interface parameters

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|-------------------------------------|-------------------------|--|-----|-----|-----|------------------------|
| Input low level | V _{il_si} | V _{DDIO} =1.2 V to 3. 6V | | | 20 | %V _{DDI} o |
| Input high level | V _{ih_si} | V _{DDIO} =1.2 V to 3.6 V | 80 | | | %V _{DDI} o |
| Output low level I ² C | V _{ol_SDI} | V _{DDIO} =1.62 V, I _{ol} =3 mA | | | 20 | %V _{DDI} o |
| Output low level I ² C | V _{ol_SDI_1.2} | V _{DDIO} =1.20 V, I _{ol} =3 mA | | | 23 | %V _{DDI} o |
| Output low level SPI | V _{ol_SDO} | V _{DDIO} =1.62 V, I _{ol} =1 mA | | | 20 | %V _{DDI} o |
| Output low level SPI | V _{ol_SDO_1.2} | V _{DDIO} =1.20 V, I _{ol} =1 mA | | | 23 | %V _{DDI} o |
| Output high level | V _{oh} | V _{DDIO} =1.62 V, I _{oh} =1 mA (SDO, SDI) | 80 | | | %V _{DDI} o |
| Output high level | V _{oh_1.2} | V _{DDIO} =1.20 V, I _{oh} =1 mA (SDO, SDI) | 60 | | | %V _{DDI} o |
| Pull-up resistor | R _{pull} | Internal CSB pull-up resistance to V _{DDIO} | 70 | 120 | 190 | kΩ |
| I ² C bus load capacitor | C _b | On SDI and SCK | | | 400 | pF |

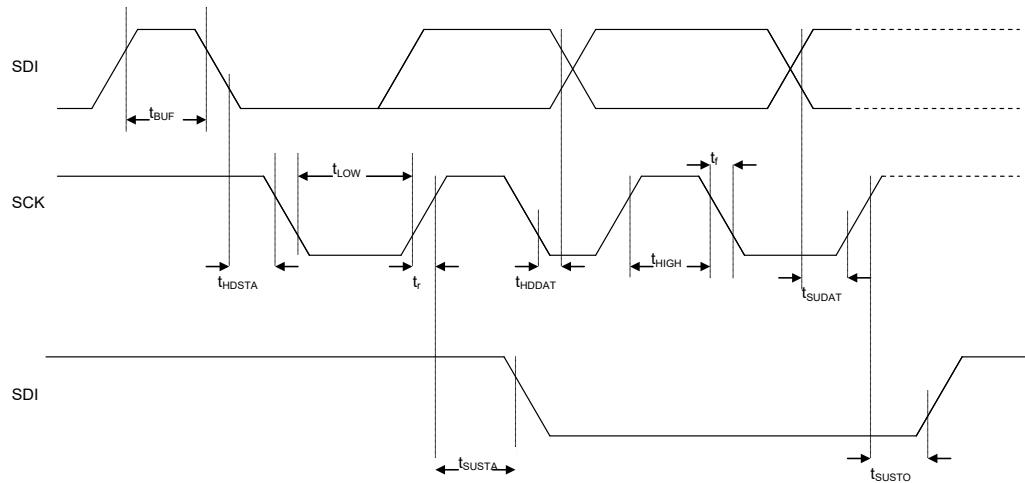
6.4.2 I²C timings

For I²C timings, the following abbreviations are used:

- “S&F mode” = standard and fast mode
- “HS mode” = high speed mode
- C_b = bus capacitance on SDA line

All other naming refers to I²C specification 2.1 (January 2000).

The I²C timing diagram is in Figure 14. The corresponding values are given in Table 33.

Figure 14: I²C timing diagramTable 33: I²C timings

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|----------------|--------------|---|-----------|-----|-----|----------|
| SDI setup time | $t_{SU;DAT}$ | S&F Mode HS mode | 160 30 | | | ns ns |
| SDI hold time | $t_{HD;DAT}$ | S&F Mode, $C_b \leq 100 \text{ pF}$ | 80 | | | ns |
| | | S&F Mode, $C_b \leq 400 \text{ pF}$ | 90 | | | ns |
| | | HS mode, $C_b \leq 100 \text{ pF}$ | 18 | | 115 | ns |
| | | HS mode, $C_b \leq 400 \text{ pF}$ | 24 | | 150 | ns |
| SCK low pulse | t_{LOW} | HS mode, $C_b \leq 100 \text{ pF}$ $V_{DDIO} = 1.62 \text{ V}$ | 160 | | | ns |
| SCK low pulse | t_{LOW} | HS mode, $C_b \leq 100 \text{ pF}$ $V_{DDIO} = 1.2 \text{ V}$ | 210 | | | ns |

The above-mentioned I²C specific timings correspond to the following internal added delays:

- Input delay between SDI and SCK inputs: SDI is more delayed than SCK by typically 100 ns in Standard and Fast Modes and by typically 20 ns in High Speed Mode.
- Output delay from SCK falling edge to SDI output propagation is typically 140 ns in Standard and Fast Modes and typically 70 ns in High Speed Mode.

6.4.3 SPI timings

The SPI timing diagram is in Figure 15, while the corresponding values are given in Table 34. All timings apply both to 4- and 3-wire SPI.

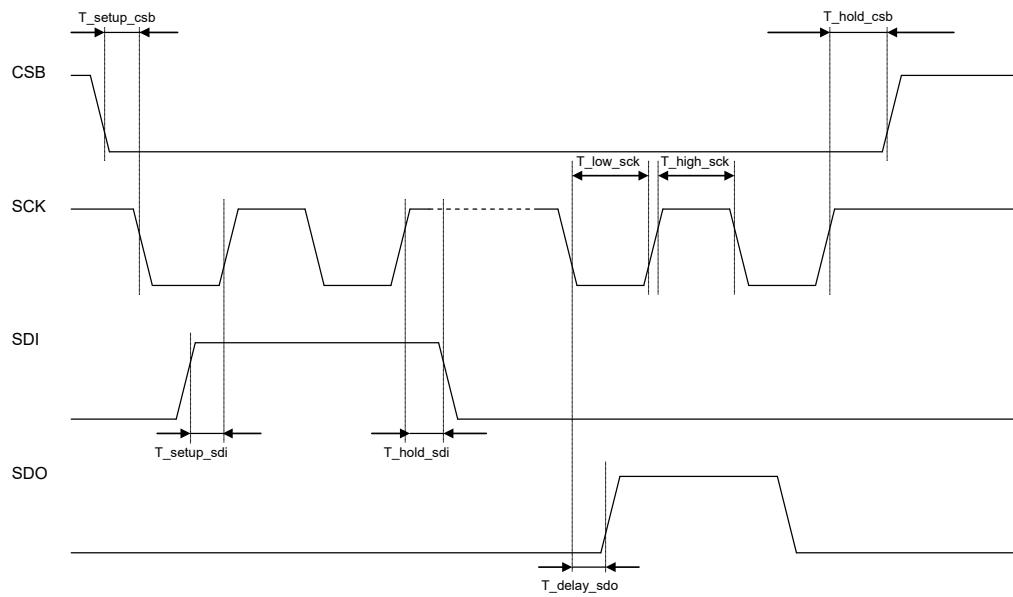


Figure 15: SPI timing diagram

Table 34: SPI timings

| Parameter | Symbol | Condition | Min | Typ | Max | Unit |
|---------------------------|-------------------------|---|-----|-----|-----|------|
| SPI clock input frequency | F_{spi} | | 0 | | 10 | MHz |
| SCK low pulse | $T_{\text{low_sck}}$ | | 20 | | | ns |
| SCK high pulse | $T_{\text{high_sck}}$ | | 20 | | | ns |
| SDI setup time | $T_{\text{setup_sdi}}$ | | 20 | | | ns |
| SDI hold time | $T_{\text{hold_sdi}}$ | | 20 | | | ns |
| SDO output delay | $T_{\text{delay_sdo}}$ | 25 pF load, $V_{\text{DDIO}}=1.6 \text{ V min}$ | | | 30 | ns |
| SDO output delay | $T_{\text{delay_sdo}}$ | 25 pF load, $V_{\text{DDIO}}=1.2 \text{ V min}$ | | | 40 | ns |
| CSB setup time | $T_{\text{setup_csb}}$ | | 20 | | | ns |
| CSB hold time | $T_{\text{hold_csb}}$ | | 20 | | | ns |

7. Pin-out and connection diagram

7.1 Pin-out

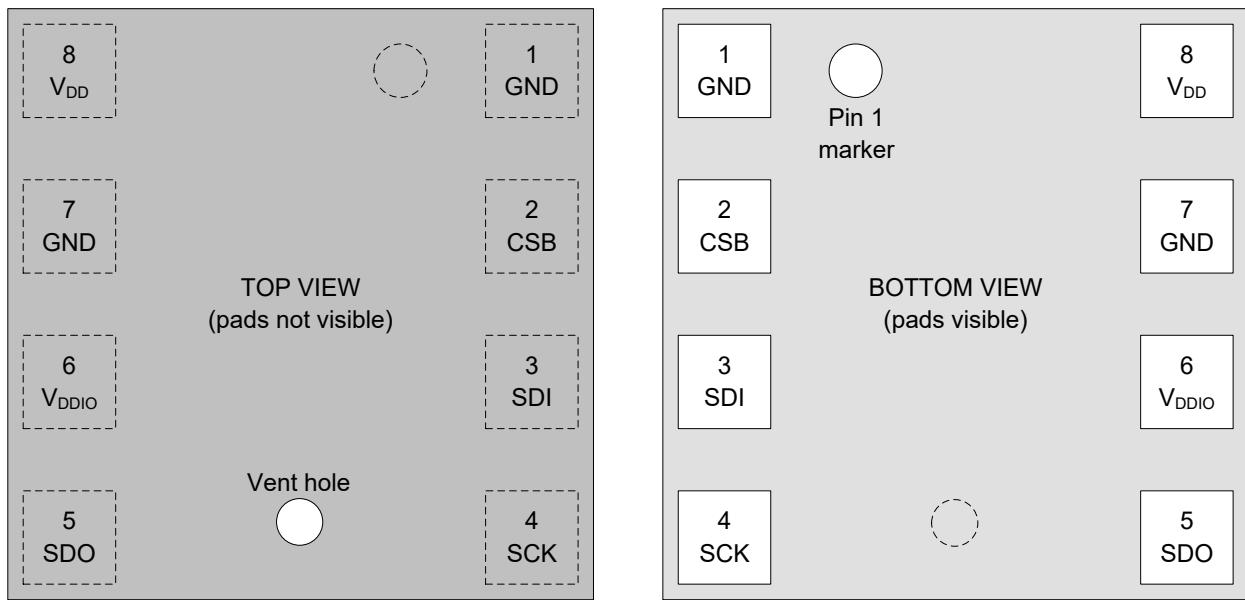


Figure 16: Pin-out top and bottom view

Note: The pin numbering of BME280 is performed in the untypical clockwise direction when seen in top view and counter-clockwise when seen in bottom view.

Table 35: Pin description

| Pin | Name | I/O Type | Description | Connect to | | |
|-----|-------------------|----------|----------------------------|-------------------|---------|-------------------------|
| | | | | SPI 4W | SPI 3W | I ² C |
| 1 | GND | Supply | Ground | GND | | |
| 2 | CSB | In | Chip select | CSB | CSB | V _{DDIO} |
| 3 | SDI | In/Out | Serial data input | SDI | SDI/SDO | SDA |
| 4 | SCK | In | Serial clock input | SCK | SCK | SCL |
| 5 | SDO | In/Out | Serial data output | SDO | DNC | GND for default address |
| 6 | V _{DDIO} | Supply | Digital / Interface supply | V _{DDIO} | | |
| 7 | GND | Supply | Ground | GND | | |
| 8 | V _{DD} | Supply | Analog supply | V _{DD} | | |

7.2 Connection diagram I²C

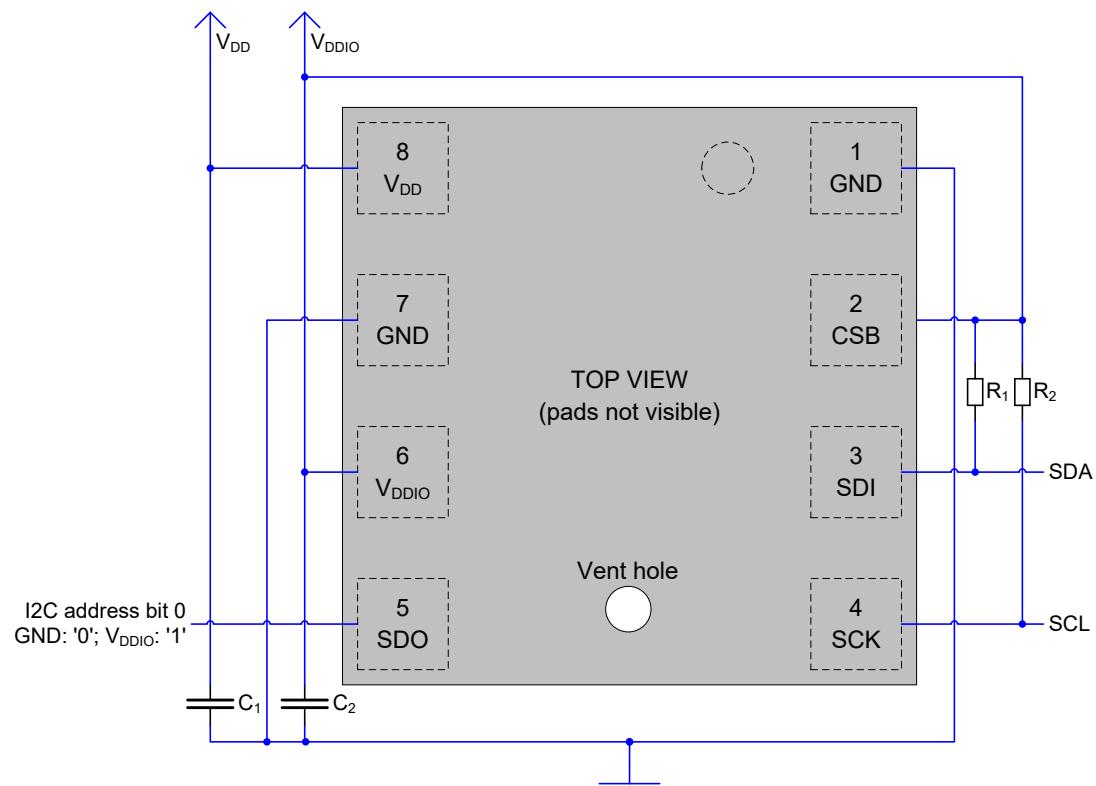


Figure 17: I²C connection diagram

Notes:

- The recommended value for C₁, C₂ is 100 nF
- The value for the pull-up resistors R₁, R₂ should be based on the interface timing and the bus load; a normal value is 4.7 kΩ
- A direct connection between CSB and V_{DDIO} is required

7.3 Connection diagram 4-wire SPI

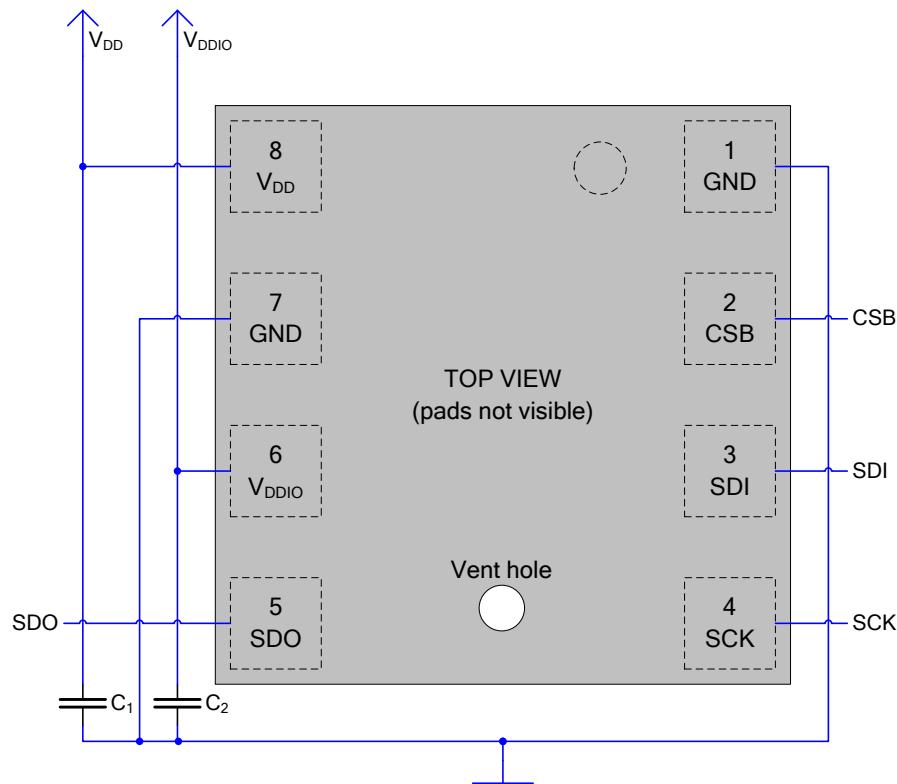


Figure 18: 4-wire SPI connection diagram

Note: The recommended value for C₁, C₂ is 100 nF

7.4 Connection diagram 3-wire SPI

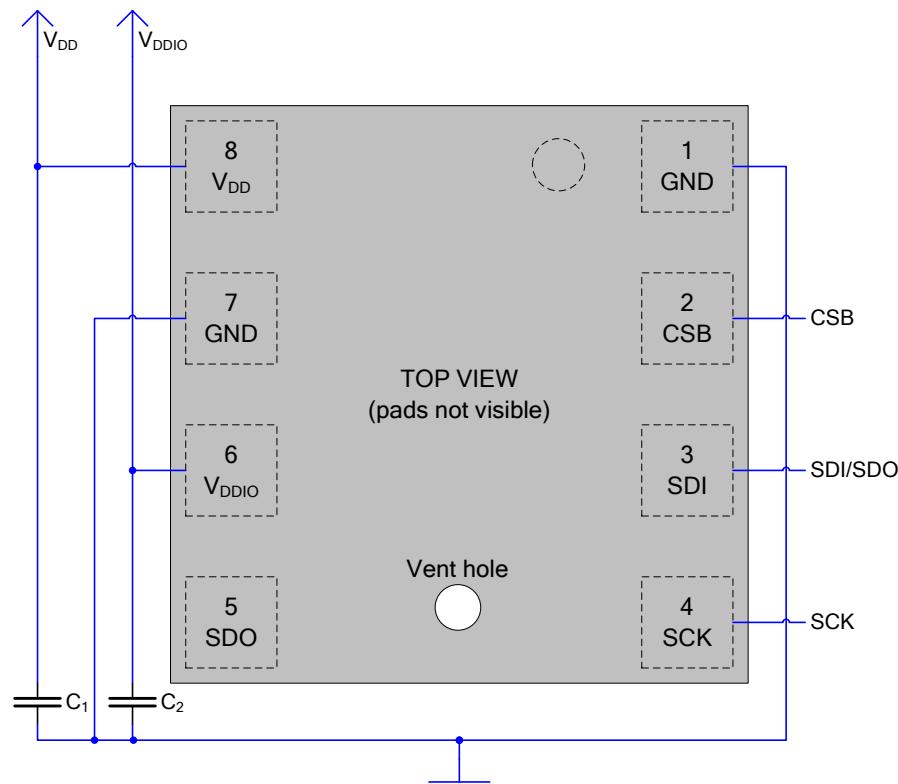


Figure 19: 3-wire SPI connection diagram

Note: The recommended value for C₁, C₂ is 100 nF

7.5 Package dimensions

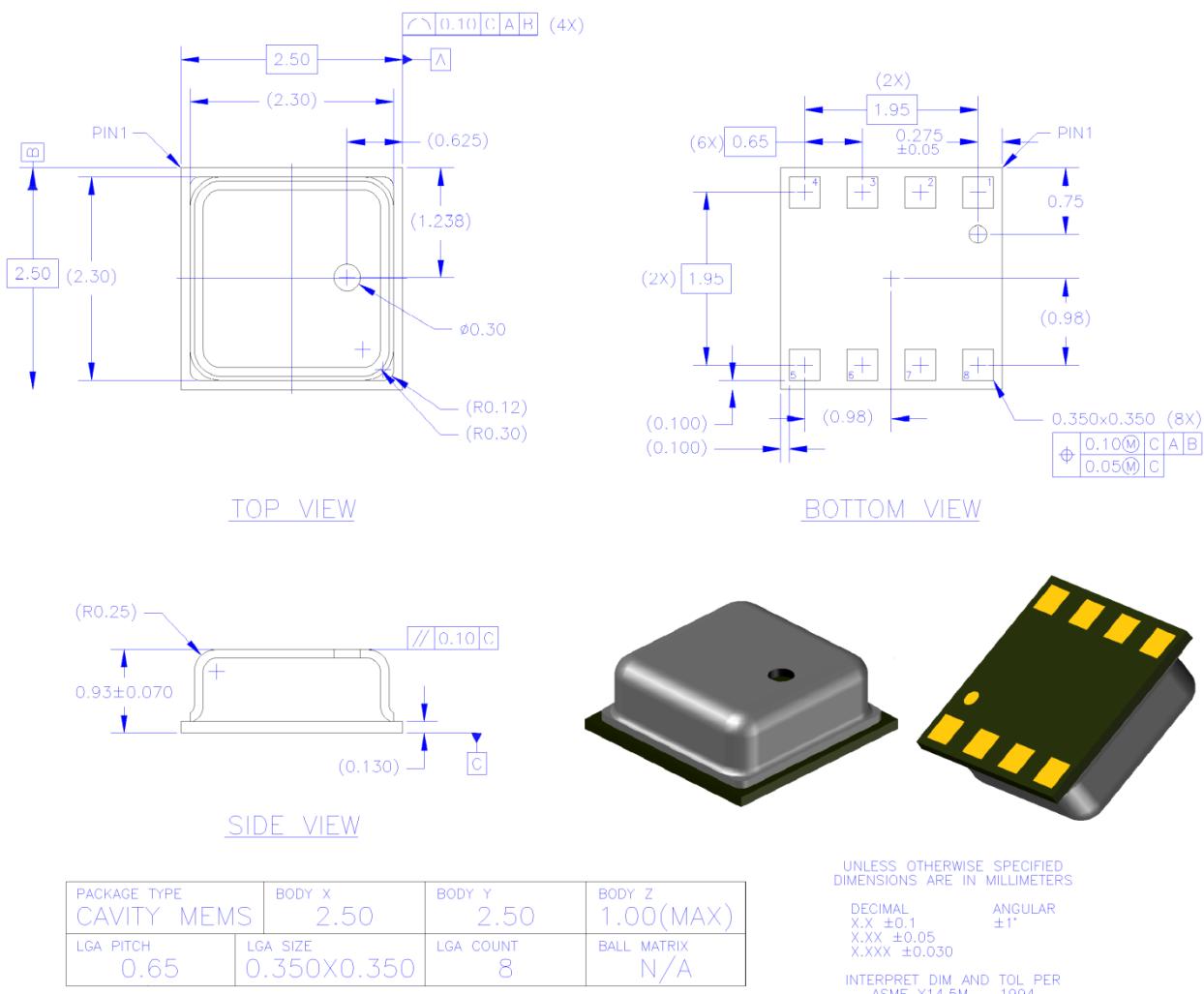


Figure 20: Package dimensions for top, bottom and side view

7.6 Landing pattern recommendation

For the design of the landing pattern, the following dimensioning is recommended:

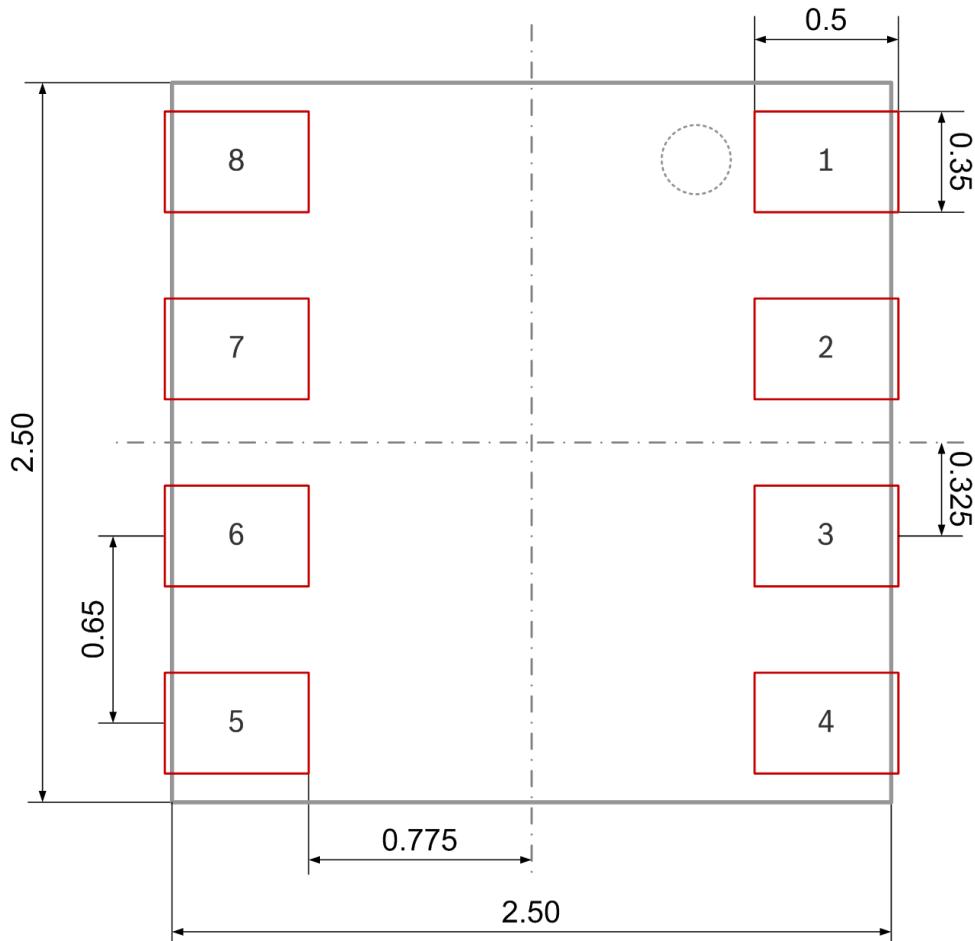


Figure 21: Recommended landing pattern (top view)

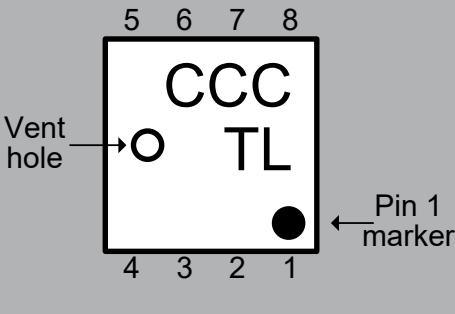
Note: red areas demarcate exposed PCB metal pads.

- In case of a solder mask defined (SMD) PCB process, the land dimensions should be defined by solder mask openings. The underlying metal pads are larger than these openings.
- In case of a non solder mask defined (NSMD) PCB process, the land dimensions should be defined in the metal layer. The mask openings are larger than these metal pads.

7.7 Marking

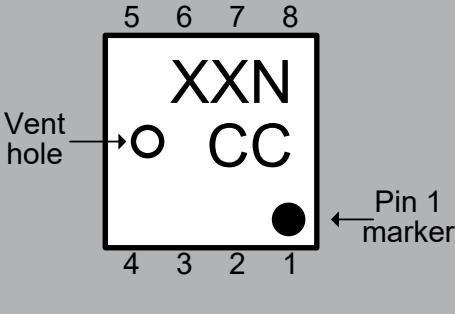
7.7.1 Mass production devices

Table 36: Marking of mass production parts

| Marking | Symbol | Description |
|---|--------|--|
|  | CCC | <u>Lot counter</u> : 3 alphanumeric digits, variable to generate mass production trace-code |
| | T | <u>Product number</u> : 1 alphanumeric digit, fixed to identify product type, T = "U" "U" is associated with the product BME280 (part number 0 273 141 185) |
| | L | <u>Sub-contractor ID</u> : 1 alphanumeric digit, variable to identify sub-contractor (L = "P") |

7.7.2 Engineering samples

Table 37: Marking of engineering samples

| Marking | Symbol | Description |
|---|--------|---|
|  | XX | <u>Sample ID</u> : 2 alphanumeric digits, variable to generate trace-code |
| | N | <u>Eng. Sample ID</u> : 1 alphanumeric digit, fixed to identify engineering sample, N = "*" or "e" or "E" |
| | CC | <u>Counter ID</u> : 2 alphanumeric digits, variable to generate trace-code |

7.8 Soldering guidelines and reconditioning recommendations

The moisture sensitivity level of the BME280 sensors corresponds to JEDEC Level 1, see also:

- IPC/JEDEC J-STD-020C “Joint Industry Standard: Moisture/Reflow Sensitivity Classification for non-hermetic Solid State Surface Mount Devices”
- IPC/JEDEC J-STD-033A “Joint Industry Standard: Handling, Packing, Shipping and Use of Moisture/Reflow Sensitive Surface Mount Devices”.

The sensor fulfils the lead-free soldering requirements of the above-mentioned IPC/JEDEC standard, i.e. reflow soldering with a peak temperature up to 260°C. The minimum height of the solder after reflow shall be at least 50µm. This is required for good mechanical decoupling between the sensor device and the printed circuit board (PCB).

| Profile Feature | | Pb-Free Assembly |
|---|--|------------------|
| Average Ramp-Up Rate ($T_{S_{\max}}$ to T_p) | | 3° C/second max. |
| Preheat | | |
| - Temperature Min ($T_{S_{\min}}$) | | 150 °C |
| - Temperature Max ($T_{S_{\max}}$) | | 200 °C |
| - Time ($t_{S_{\min}}$ to $t_{S_{\max}}$) | | 60-180 seconds |
| Time maintained above: | | |
| - Temperature (T_L) | | 217 °C |
| - Time (t_L) | | 60-150 seconds |
| Peak/Classification Temperature (T_p) | | 260 °C |
| Time within 5 °C of actual Peak Temperature (t_p) | | 20-40 seconds |
| Ramp-Down Rate | | 6 °C/second max. |
| Time 25 °C to Peak Temperature | | 8 minutes max. |

Note 1: All temperatures refer to topside of the package, measured on the package body surface.

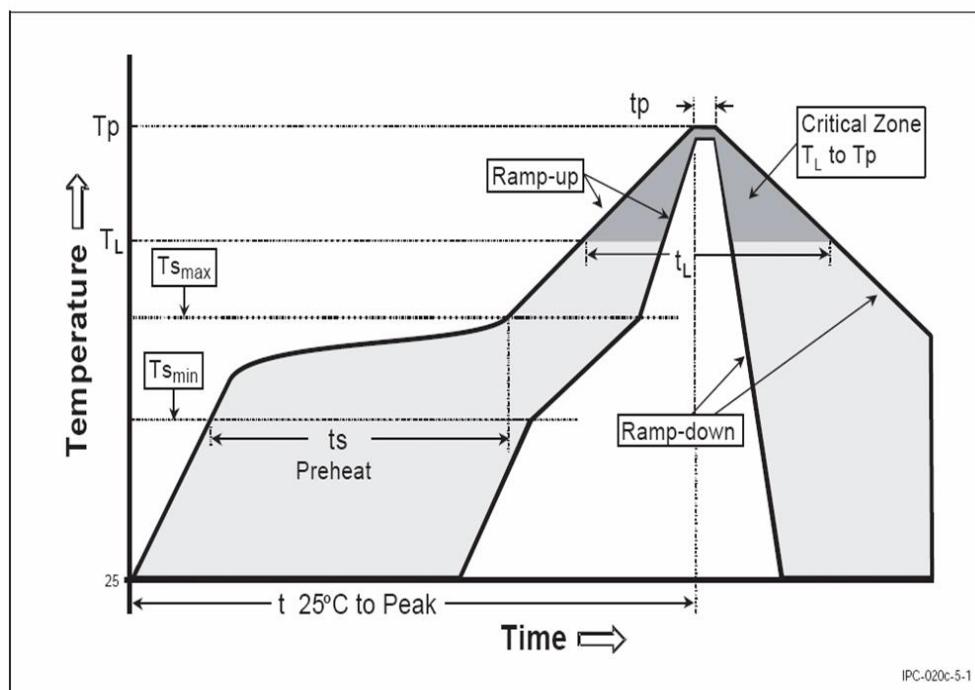


Figure 22: Soldering profile

7.9 Reconditioning Procedure

After exposing the device to operating conditions, which exceed the limits specified in section 1.2, e.g. after reflow, the humidity sensor may possess an additional offset. Therefore the following reconditioning procedure is mandatory to restore the calibration state:

1. Dry-Baking: 120 °C at <5% rH for 2 h
2. Re-Hydration: 70 °C at 75% rH for 6 h

or alternatively

1. Dry-Baking: 120 °C at <5% rH for 2 h
2. Re-Hydration: 25 °C at 75% rH for 24 h

or alternatively after solder reflow only

1. Do not perform Dry-Baking
2. Ambient Re-Hydration: ~25 °C at >40% rH for >5d

7.10 Tape and reel specification

7.10.1 Dimensions

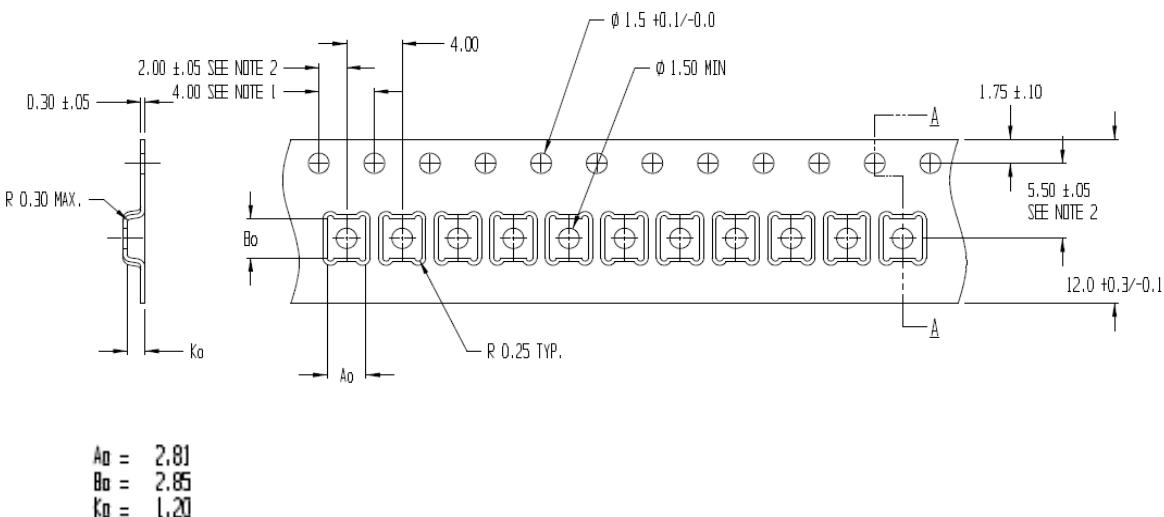


Figure 23: Tape and Reel dimensions

Quantity per reel: 10 kpcs.

7.10.2 Orientation within the reel

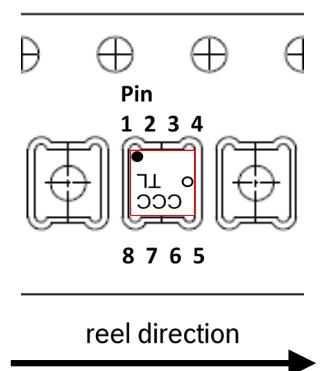


Figure 24: Orientation within tape

7.11 Mounting and assembly recommendations

In order to achieve the specified performance for your design, the following recommendations and the "Handling, soldering & mounting instructions BME280" should be taken into consideration when mounting a pressure sensor on a printed-circuit board (PCB):

- The clearance above the metal lid shall be 0.1mm at minimum.
- For the device housing appropriate venting needs to be provided in case the ambient pressure shall be measured.
- Liquids shall not come into direct contact with the device.
- During operation the sensor chip is sensitive to light, which can influence the accuracy of the measurement (photo-current of silicon). The position of the vent hole minimizes the light exposure of the sensor chip. Nevertheless, Bosch Sensortec recommends avoiding the exposure of BME280 to strong light sources.
- Soldering may not be done using vapor phase processes since the sensor will be damaged by the liquids used in these processes.

7.12 Environmental safety

7.12.1 RoHS

The BME280 sensor meets the requirements of the EC restriction of hazardous substances (RoHS) directive, see also:

RoHS-Directive 2011/65/EU and its amendments, including the amendment 2015/863/EU on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

7.12.2 Halogen content

The BME280 is halogen-free. For more details on the analysis results please contact your Bosch Sensortec representative.

7.12.3 Internal package structure

Within the scope of Bosch Sensortec's ambition to improve its products and secure the mass product supply, Bosch Sensortec qualifies additional sources (e.g. 2nd source) for the package of the BME280.

While Bosch Sensortec took care that all of the technical packages parameters are described above are 100% identical for all sources, there can be differences in the chemical content and the internal structural between the different package sources.

However, as secured by the extensive product qualification process of Bosch Sensortec, this has no impact to the usage or to the quality of the BME280 product.

8. Appendix A: Alternative compensation formulas

8.1 Compensation formulas in double precision floating point

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer. Humidity is expected to be received in 16 bit format, positive, stored in a 32 bit signed integer.

The variable `t_fine` (signed 32 bit) carries a fine resolution temperature value over to the pressure compensation formula and could be implemented as a global variable.

The data type “`BME280_S32_t`” should define a 32 bit signed integer variable type and could usually be defined as “long signed int”. The revision of the code is rev. 1.1 (pressure and temperature) and rev. 1.0 (humidity).

Compensating the measurement value with double precision gives the best possible accuracy but is only recommended for PC applications.

```
// Returns temperature in DegC, double precision. Output value of "51.23" equals 51.23 DegC.
// t_fine carries fine temperature as global value
BME280_S32_t t_fine;
double BME280_compensate_T_double(BME280_S32_t adc_T)
{
    double var1, var2, T;
    var1 = (((double)adc_T)/16384.0 - ((double)dig_T1)/1024.0) * ((double)dig_T2);
    var2 = (((((double)adc_T)/131072.0 - ((double)dig_T1)/8192.0) *
        ((double)adc_T)/131072.0 - ((double)dig_T1)/8192.0)) * ((double)dig_T3);
    t_fine = (BME280_S32_t)(var1 + var2);
    T = (var1 + var2) / 5120.0;
    return T;
}
// Returns pressure in Pa as double. Output value of "96386.2" equals 96386.2 Pa = 963.862 hPa
double BME280_compensate_P_double(BME280_S32_t adc_P)
{
    double var1, var2, p;
    var1 = ((double)t_fine/2.0) - 64000.0;
    var2 = var1 * var1 * ((double)dig_P6) / 32768.0;
    var2 = var2 + var1 * ((double)dig_P5) * 2.0;
    var2 = (var2/4.0)+((double)dig_P4) * 65536.0;
    var1 = (((double)dig_P3) * var1 * var1 / 524288.0 + ((double)dig_P2) * var1) / 524288.0;
    var1 = (1.0 + var1 / 32768.0)*(double)dig_P1;
    if (var1 == 0.0)
    {
        return 0; // avoid exception caused by division by zero
    }
    p = 1048576.0 - (double)adc_P;
    p = (p - (var2 / 4096.0)) * 6250.0 / var1;
    var1 = ((double)dig_P9) * p * p / 2147483648.0;
    var2 = p * ((double)dig_P8) / 32768.0;
    p = p + (var1 + var2 + ((double)dig_P7)) / 16.0;
    return p;
}
// Returns humidity in %rH as as double. Output value of "46.332" represents
// 46.332 %rH
double bme280_compensate_H_double(BME280_S32_t adc_H);
{
    double var_H;

    var_H = ((double)t_fine) - 76800.0;
    var_H = (adc_H - (((double)dig_H4) * 64.0 + ((double)dig_H5) / 16384.0 *
        var_H)) * (((double)dig_H2) / 65536.0 * (1.0 + ((double)dig_H6) /
        67108864.0 * var_H *
        (1.0 + ((double)dig_H3) / 67108864.0 * var_H)));
    var_H = var_H * (1.0 - ((double)dig_H1) * var_H / 524288.0);

    if (var_H > 100.0)
        var_H = 100.0;
    else if (var_H < 0.0)
        var_H = 0.0;
    return var_H;
}
```

8.2 Pressure compensation in 32 bit fixed point

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer.

The variable `t_fine` (signed 32 bit) carries a fine resolution temperature value over to the pressure compensation formula and could be implemented as a global variable.

The data type “`BME280_S32_t`” should define a 32 bit signed integer variable type and can usually be defined as “long signed int”.

The data type “`BME280_U32_t`” should define a 32 bit unsigned integer variable type and can usually be defined as “long unsigned int”.

Compensating the pressure value with 32 bit integer has an accuracy of typically 1 Pa (1-sigma). At high filter levels this adds a significant amount of noise to the output values and reduces their resolution.

```
// Returns temperature in DegC, resolution is 0.01 DegC. Output value of "5123" equals 51.23
DegC.
// t_fine carries fine temperature as global value
BME280_S32_t t_fine;
BME280_S32_t BME280_compensate_T_int32(BME280_S32_t adc_T)
{
    BME280_S32_t var1, var2, T;
    var1 = (((adc_T>>3) - ((BME280_S32_t)dig_T1<<1))) * ((BME280_S32_t)dig_T2)) >> 11;
    var2 = (((((adc_T>>4) - ((BME280_S32_t)dig_T1)) * ((adc_T>>4) - ((BME280_S32_t)dig_T1)))>> 12) *
        ((BME280_S32_t)dig_T3)) >> 14;
    t_fine = var1 + var2;
    T = (t_fine * 5 + 128) >> 8;
    return T;
}

// Returns pressure in Pa as unsigned 32 bit integer. Output value of "96386" equals 96386 Pa
= 963.86 hPa
BME280_U32_t BME280_compensate_P_int32(BME280_S32_t adc_P)
{
    BME280_S32_t var1, var2;
    BME280_U32_t p;
    var1 = (((BME280_S32_t)t_fine)>>1) - (BME280_S32_t)64000;
    var2 = (((var1>>2) * (var1>>2)) >> 11 ) * ((BME280_S32_t)dig_P6);
    var2 = var2 + ((var1*((BME280_S32_t)dig_P5))<<1);
    var2 = (var2>>2)+((BME280_S32_t)dig_P4)<<16);
    var1 = (((dig_P3 * ((var1>>2) * (var1>>2)) >> 13 )) >> 3) + (((BME280_S32_t)dig_P2) *
var1)>>1)>>18;
    var1 = (((32768+var1))*((BME280_S32_t)dig_P1))>>15;
    if (var1 == 0)
    {
        return 0; // avoid exception caused by division by zero
    }
    p = (((BME280_U32_t) (((BME280_S32_t)1048576)-adc_P)-(var2>>12)))*3125;
    if (p < 0x80000000)
    {
        p = (p << 1) / ((BME280_U32_t)var1);
    }
    else
    {
        p = (p / (BME280_U32_t)var1) * 2;
    }
    var1 = (((BME280_S32_t)dig_P9) * ((BME280_S32_t)((p>>3) * (p>>3))>>13))>>12;
    var2 = (((BME280_S32_t)(p>>2)) * ((BME280_S32_t)dig_P8))>>13;
    p = (BME280_U32_t)((BME280_S32_t)p + ((var1 + var2 + dig_P7) >> 4));
    return p;
}
```

9. Appendix B: Measurement time and current calculation

In this chapter, formulas are given to calculate measurement rate, filter bandwidth and current consumption in different settings.

9.1 Measurement time

The active measurement time depends on the selected values for humidity, temperature and pressure oversampling and can be calculated in milliseconds using the formulas below.

$$t_{measure,typ} = 1 + [2 \cdot T_{oversampling}]_{osrs_t \neq 0} + [2 \cdot P_{oversampling} + 0.5]_{osrs_p \neq 0} \\ + [2 \cdot H_{oversampling} + 0.5]_{osrs_h \neq 0}$$

$$t_{measure,max} = 1.25 + [2.3 \cdot T_{oversampling}]_{osrs_t \neq 0} + [2.3 \cdot P_{oversampling} + 0.575]_{osrs_p \neq 0} \\ + [2.3 \cdot H_{oversampling} + 0.575]_{osrs_h \neq 0}$$

For example, using temperature oversampling $\times 1$, pressure oversampling $\times 4$ and no humidity measurement, the measurement time is:

$$t_{measure,typ} = 1 + [2 \cdot 1] + [2 \cdot 4 + 0.5] + [0] = 11.5 \text{ ms}$$

$$t_{measure,max} = 1.25 + [2.3 \cdot 1] + [2.3 \cdot 4 + 0.575] + [0] = 13.325 \text{ ms}$$

9.2 Measurement rate in forced mode

In forced mode, the measurement rate depends on the rate at which it is forced by the master. The highest possible frequency in Hz can be calculated as:

$$ODR_{max,forced} = \frac{1000}{t_{measure}}$$

If measurements are forced faster than they can be executed, the data rate saturates at the attainable data rate. For the example above with 11.5 ms measurement time, the typically achievable output data rate would be:

$$ODR_{max,forced} = \frac{1000}{11.5} = 87 \text{ Hz}$$

9.3 Measurement rate in normal mode

The measurement rate in normal mode depends on the measurement time and the standby time and can be calculated in Hz using the following formula:

$$ODR_{normal_mode} = \frac{1000}{t_{measure} + t_{standby}}$$

The accuracy of $t_{standby}$ is described in the specification parameter $\Delta t_{standby}$. For the example above with 11.5 ms measurement time, setting normal mode with a standby time of 62.5 ms would result in a data rate of:

$$ODR_{normal_mode} = \frac{1000}{11.5 + 62.5} = 13.51 \text{ Hz}$$

9.4 Response time using IIR filter

When using the IIR filter, the response time of the sensor depends on the selected filter coefficient and the data rate used. It can be calculated using the following formula:

$$t_{response, 75\%} = \frac{1000 \cdot n_{samples, 75\%}}{ODR}$$

For the example above with a data rate of 13.51 Hz, the user could select a filter coefficient of 8.

According to Table 6, the number of samples needed to reach 75% of a step response using this filter setting is 11. The response time with filter is therefore:

$$t_{response, 75\%} = \frac{1000 \cdot 11}{13.51} = 814 \text{ ms}$$

9.5 Current consumption

The current consumption depends on the selected oversampling settings, the measurement rate and the sensor mode, but not on the IIR filter setting. It can be calculated as:

$$\begin{aligned} I_{DD,forced} &= I_{DDSL} \cdot (1 - t_{measure} \cdot ODR) + \frac{ODR}{1000} \\ &\quad \cdot (205 + I_{DDT} \cdot [2 \cdot T_oversampling]_{osrs_t \neq 0} + I_{DDP} \cdot [2 \cdot P_oversampling + 0.5]_{osrs_p \neq 0} \\ &\quad + I_{DDH} \cdot [2 \cdot H_oversampling + 0.5]_{osrs_h \neq 0}) \end{aligned}$$

$$\begin{aligned} I_{DD,normal} &= I_{DDSB} \cdot (1 - t_{measure} \cdot ODR) + \frac{ODR}{1000} \\ &\quad \cdot (205 + I_{DDT} \cdot [2 \cdot T_oversampling]_{osrs_t \neq 0} + I_{DDP} \cdot [2 \cdot P_oversampling + 0.5]_{osrs_p \neq 0} \\ &\quad + I_{DDH} \cdot [2 \cdot H_oversampling + 0.5]_{osrs_h \neq 0}) \end{aligned}$$

Note that the only difference between forced and normal mode current consumption is that the current for the inactive time is either I_{DDSL} or I_{DDSB} . For the example above, the current would be

$$\begin{aligned} I_{DD,normal} &= 0.2 \cdot (1 - 0.0115 \cdot 13.51) + \frac{13.51}{1000} (205 + 350 \cdot [2 \cdot 1] + 714 \cdot [2 \cdot 4 + 0.5] + [0]) \\ &= 0.2 \cdot (0.845) + \frac{13.51}{1000} (205 + 700 + 6069 + 0) \\ &= 0.2 + 94.2 = 94.4 \mu\text{A} \end{aligned}$$

10. Self test

The following chapter provides an explanation to the self-test code for the Bosch Sensortec BME280. The code itself refers to the API (Application Programming Interface) of the sensor, which can be obtained from Bosch Sensortec and is also included in this release package.

10.1 Self-test flow

The self-test starts by performing a soft reset of the device. After this, Chip-ID and trimming data are read and verified. Then temperature and pressure are measured and compared against customisable plausibility limits. A flow chart is given below.

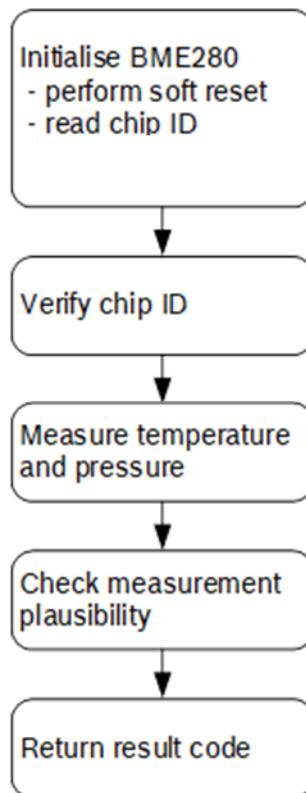


Figure 25: Self-test flow chart

10.2 Function return codes

A list of the possible function return codes can be found below.

- 0 Sensor OK
- 10 Communication error or wrong device found
- 20 Trimming data out of bound
- 30 Temperature bond wire failure or MEMS defect
- 31 Pressure bond wire failure or MEMS defect
- 40 Implausible temperature (default limits: 0...40°C)
- 41 Implausible pressure (default limits: 900...1100 hPa)
- 42 Implausible humidity (default limits: 20...80 %rH)

Error testing is done in ascending error code sequence. This means that if e.g. a trimming data error is detected (code 20), the temperature plausibility (code 40) is not checked anymore. Instead, error code 20 is returned and no others tests are performed.

10.3 Usage

10.3.1 File and function pointer integration

- ▶ Include bme280.c in your programming environment and add the path to the compiler.
- ▶ Include bme280_selftest.c in your programming environment and add the path to the compiler.
- ▶ Modify the lines with read/write function pointer to match your system. Sample functions are given in chapter 10.5:

```
bme280.bus_read    = BME280_I2C_bus_read; // must be defined by customer
bme280.bus_write   = BME280_I2C_bus_write; // must be defined by customer
bme280.delay_msec  = BME280_delay_msec; // must be defined by customer
```

- ▶ If necessary, adapt the measurement plausibility limits in bme280_selftest.h. The default limits are 0...40°C for temperature and 900...1100 hPa for pressure measurement.
- ▶ If you are using I²C communication with the address 0x77 (SDO pin high), then change the BME280.h line

```
#define BME280_I2C_ADDRESS      BME280_I2C_ADDRESS1
      into
#define BME280_I2C_ADDRESS      BME280_I2C_ADDRESS2
```

10.3.2 Function call

Call the self test function using:

```
unsigned char testresult;
testresult = bme280_selftest();
```

A test result of 0 indicates no error. The other return codes are detailed in chapter 10.2.

10.3.3 Test time and interface requirements

The self test uses a total wait time of 9 milliseconds. Of this, 2 milliseconds are used as wait time for soft reset and 7 milliseconds are used as wait time for conversion. The soft reset is performed in order to erase any possible old settings and could be omitted if the sensor is known to be in an untouched state after power on.

In the self test function, 4 write commands and 6 read commands are issued. In total, 4 bytes are written and 34 bytes are read. Assuming burst read is used, the following time duration can be expected for communication including overhead:

- ▶ 6.0 ms for I²C at 100 kHz
- ▶ 1.5 ms for I²C at 400 kHz
- ▶ 0.5 ms for SPI at 1 MHz

Assuming a 400 kHz I²C interface with burst reads, the total function run time therefore equals 10.5 milliseconds.

10.4 Function explanation

10.4.1 Communication test

This function attempts to read the Chip ID. If it is correct, a functioning communication is assumed. Note that the write function functionality is not explicitly tested.

10.4.2 Bond wire test

A pressure and temperature measurement is performed and uncompensated pressure and temperature values are read out. If the measurement results are clipped to the respective minimum or maximum ADC values, this is usually caused by defective bond wires. However, a defective sensing element could also cause this test to fail.

Please note that some combinations of bond wire or sensing element defects do not result in clipping of the measurement value and will therefore not be detected with this test. These cases can be detected by the plausibility test instead.

10.4.3 Measurement plausibility test

The pressure and temperature values read out previously are compensated using the read out compensation parameters. The compensated temperature and pressure is compared against plausibility limits set in bme280_selftest.h, which must be set to match the customer production environment. Please use the the plausibility limits as described in chapter 3.

10.5 Sample read, write and delay function

Below some samples read, write and delay functions are given. These are platform dependant and should only give an idea of how the functions could look.

```
signed char BME280_I2C_bus_read(unsigned char device_addr, unsigned char
    reg_addr, unsigned char *reg_data, unsigned char cnt)
{
    int iError=0;
    unsigned char array[I2C_BUFFER_LEN];
    unsigned char stringpos;
    array[0] = reg_addr;
    iError = I2C_write_read_string(I2C0, device_addr, array, array, 1, cnt);
    for(stringpos=0;stringpos<cnt;stringpos++)
    {
        *(reg_data + stringpos) = array[stringpos];
    }
    return (signed char)iError;
}

signed char BME280_I2C_bus_write(unsigned char device_addr, unsigned char
    reg_addr, unsigned char *reg_data, unsigned char cnt)
{
    int iError=0;
    unsigned char array[I2C_BUFFER_LEN];
    unsigned char stringpos;
    array[0] = reg_addr;
    for(stringpos=0;stringpos<cnt;stringpos++)
    {
        array[stringpos+1] = *(reg_data + stringpos);
    }
    iError = I2C_write_string(I2C0, device_addr, array, cnt+1);
    return (signed char)iError;
}

void BME280_delay_msec(BME280_U16_t msec) //delay in milliseconds
{
    BME280_U32_t counter;
    for (counter = 0; counter/2000 < msec; counter++); // 2000 counts = 1
msec
}
```

11. Legal disclaimer

11.1 Engineering samples

Engineering Samples are marked with an asterisk (*), (E) or (e). Samples may vary from the valid technical specifications of the product series contained in this data sheet. They are therefore not intended or fit for resale to third parties or for use in end products. Their sole purpose is internal client testing. The testing of an engineering sample may in no way replace the testing of a product series. Bosch Sensortec assumes no liability for the use of engineering samples. The Purchaser shall indemnify Bosch Sensortec from all claims arising from the use of engineering samples.

11.2 Product use

Bosch Sensortec products are developed for the consumer goods industry. They may only be used within the parameters of this product data sheet. They are not fit for use in life-sustaining or safety-critical systems. Safety-critical systems are those for which a malfunction is expected to lead to bodily harm, death or severe property damage. In addition, they shall not be used directly or indirectly for military purposes (including but not limited to nuclear, chemical or biological proliferation of weapons or development of missile technology), nuclear power, deep sea or space applications (including but not limited to satellite technology).

Bosch Sensortec products are released on the basis of the legal and normative requirements relevant to the Bosch Sensortec product for use in the following geographical target market: BE, BG, DK, DE, EE, FI, FR, GR, IE, IT, HR, LV, LT, LU, MT, NL, AT, PL, PT, RO, SE, SK, SI, ES, CZ, HU, CY, US, CN, JP, KR, TW. If you need further information or have further requirements, please contact your local sales contact.

The resale and/or use of Bosch Sensortec products are at the purchaser's own risk and his own responsibility. The examination of fitness for the intended use is the sole responsibility of the purchaser.

The purchaser shall indemnify Bosch Sensortec from all third party claims arising from any product use not covered by the parameters of this product data sheet or not approved by Bosch Sensortec and reimburse Bosch Sensortec for all costs in connection with such claims.

The purchaser accepts the responsibility to monitor the market for the purchased products, particularly with regard to product safety, and to inform Bosch Sensortec without delay of all safety-critical incidents.

11.3 Application examples and hints

With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Bosch Sensortec hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights or copyrights of any third party. The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. They are provided for illustrative purposes only and no evaluation regarding infringement of intellectual property rights or copyrights or regarding functionality, performance or error has been made.

12. Document history and modification

| Rev. No | Page | Description of modification/changes | Date |
|---------|--------------|--|------------|
| 0.1 | | Document creation | 2012-11-06 |
| 1.0 | | Final datasheet | 2014-11-12 |
| 1.1 | 48 | Updated RoHS directive to 2011/65/EU effective 8 June 2011 | 2015-05-07 |
| 1.2 | 2, 3 | Adjusted target devices, applications | 2015-10-15 |
| 1.4 | | Minor corrections | 2018-01-17 |
| 1.5 | | Template update | 2018-09-17 |
| 1.8 | Chapter 10 | Updated legal disclaimer | 2020-03-12 |
| 1.9 | Chapter 10 | Update disclaimer | 2020-11-23 |
| 1.10 | Chapter 10 | New added chapter self-test disclaimer now 11 | 2021-03-18 |
| 1.21 | Chapter 1.4 | New typical accuracy | 2021-07-22 |
| 1.22 | Chapter 7.12 | Update of ROHS directive | 2021-10-22 |
| 1.23 | 1.4 | Final Temperature spec | 2022-01-22 |
| 1.24 | Chapter 2 | Remove condensation | 2024-02-20 |

Bosch Sensortec GmbH
Gerhard-Kindler-Straße 9
72770 Reutlingen / Germany

contact@bosch-sensortec.com
www.bosch-sensortec.com

Modifications reserved
Document number: BST-BME280-DS001-24
Revision_1.24_022024



MFRC522

Standard performance MIFARE and NTAG frontend

Rev. 3.9 — 27 April 2016
112139

Product data sheet
COMPANY PUBLIC

1. Introduction

This document describes the functionality and electrical specifications of the contactless reader/writer MFRC522.

Remark: The MFRC522 supports all variants of the MIFARE Mini, MIFARE 1K, MIFARE 4K, MIFARE Ultralight, MIFARE DESFire EV1 and MIFARE Plus RF identification protocols. To aid readability throughout this data sheet, the MIFARE Mini, MIFARE 1K, MIFARE 4K, MIFARE Ultralight, MIFARE DESFire EV1 and MIFARE Plus products and protocols have the generic name MIFARE.

1.1 Differences between version 1.0 and 2.0

The MFRC522 is available in two versions:

- MFRC52201HN1, hereafter referred to version 1.0 and
- MFRC52202HN1, hereafter referred to version 2.0.

The MFRC522 version 2.0 is fully compatible to version 1.0 and offers in addition the following features and improvements:

- Increased stability of the reader IC in rough conditions
- An additional timer prescaler, see [Section 8.5](#).
- A corrected CRC handling when RX Multiple is set to 1

This data sheet version covers both versions of the MFRC522 and describes the differences between the versions if applicable.

2. General description

The MFRC522 is a highly integrated reader/writer IC for contactless communication at 13.56 MHz. The MFRC522 reader supports ISO/IEC 14443 A/MIFARE and NTAG.

The MFRC522's internal transmitter is able to drive a reader/writer antenna designed to communicate with ISO/IEC 14443 A/MIFARE cards and transponders without additional active circuitry. The receiver module provides a robust and efficient implementation for demodulating and decoding signals from ISO/IEC 14443 A/MIFARE compatible cards and transponders. The digital module manages the complete ISO/IEC 14443 A framing and error detection (parity and CRC) functionality.

The MFRC522 supports MF1xxS20, MF1xxS70 and MF1xxS50 products. The MFRC522 supports contactless communication and uses MIFARE higher transfer speeds up to 848 kBd in both directions.



The following host interfaces are provided:

- Serial Peripheral Interface (SPI)
- Serial UART (similar to RS232 with voltage levels dependant on pin voltage supply)
- I²C-bus interface

3. Features and benefits

- Highly integrated analog circuitry to demodulate and decode responses
- Buffered output drivers for connecting an antenna with the minimum number of external components
- Supports ISO/IEC 14443 A/MIFARE and NTAG
- Typical operating distance in Read/Write mode up to 50 mm depending on the antenna size and tuning
- Supports MF1xxS20, MF1xxS70 and MF1xxS50 encryption in Read/Write mode
- Supports ISO/IEC 14443 A higher transfer speed communication up to 848 kBd
- Supports MFIN/MFOUT
- Additional internal power supply to the smart card IC connected via MFIN/MFOUT
- Supported host interfaces
 - ◆ SPI up to 10 Mbit/s
 - ◆ I²C-bus interface up to 400 kBd in Fast mode, up to 3400 kBd in High-speed mode
 - ◆ RS232 Serial UART up to 1228.8 kBd, with voltage levels dependant on pin voltage supply
- FIFO buffer handles 64 byte send and receive
- Flexible interrupt modes
- Hard reset with low power function
- Power-down by software mode
- Programmable timer
- Internal oscillator for connection to 27.12 MHz quartz crystal
- 2.5 V to 3.3 V power supply
- CRC coprocessor
- Programmable I/O pins
- Internal self-test

4. Quick reference data

Table 1. Quick reference data

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|----------------|------------------------|---|---|-----|-----|-----|------|
| V_{DDA} | analog supply voltage | $V_{DD(PVDD)} \leq V_{DDA} = V_{DDD} = V_{DD(TVDD)}$; $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | [1] [2] | 2.5 | 3.3 | 3.6 | V |
| V_{DDD} | digital supply voltage | | | 2.5 | 3.3 | 3.6 | V |
| $V_{DD(TVDD)}$ | TVDD supply voltage | | | 2.5 | 3.3 | 3.6 | V |
| $V_{DD(PVDD)}$ | PVDD supply voltage | | [3] | 1.6 | 1.8 | 3.6 | V |
| $V_{DD(SVDD)}$ | SVDD supply voltage | $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | | 1.6 | - | 3.6 | V |

Table 1. Quick reference data ...*continued*

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|----------------|------------------------|--|-----------|-----|-----|-----|--------------------|
| I_{pd} | power-down current | $V_{DDA} = V_{DDD} = V_{DD(TVDD)} = V_{DD(PVDD)} = 3\text{ V}$ | | | | | |
| | | hard power-down; pin NRSTPD set LOW | [4] | - | - | 5 | μA |
| | | soft power-down; RF level detector on | [4] | - | - | 10 | μA |
| I_{DDD} | digital supply current | pin DVDD; $V_{DDD} = 3\text{ V}$ | | - | 6.5 | 9 | mA |
| I_{DDA} | analog supply current | pin AVDD; $V_{DDA} = 3\text{ V}$, CommandReg register's RcvOff bit = 0 | | - | 7 | 10 | mA |
| | | pin AVDD; receiver switched off; $V_{DDA} = 3\text{ V}$, CommandReg register's RcvOff bit = 1 | | - | 3 | 5 | mA |
| $I_{DD(PVDD)}$ | PVDD supply current | pin PVDD | [5] | - | - | 40 | mA |
| $I_{DD(TVDD)}$ | TVDD supply current | pin TVDD; continuous wave | [6][7][8] | - | 60 | 100 | mA |
| T_{amb} | ambient temperature | HVQFN32 | | -25 | - | +85 | $^{\circ}\text{C}$ |

[1] Supply voltages below 3 V reduce the performance in, for example, the achievable operating distance.

[2] V_{DDA} , V_{DDD} and $V_{DD(TVDD)}$ must always be the same voltage.

[3] $V_{DD(PVDD)}$ must always be the same or lower voltage than V_{DDD} .

[4] I_{pd} is the total current for all supplies.

[5] $I_{DD(PVDD)}$ depends on the overall load at the digital pins.

[6] $I_{DD(TVDD)}$ depends on $V_{DD(TVDD)}$ and the external circuit connected to pins TX1 and TX2.

[7] During typical circuit operation, the overall current is below 100 mA.

[8] Typical value using a complementary driver configuration and an antenna matched to $40\ \Omega$ between pins TX1 and TX2 at 13.56 MHz.

5. Ordering information

Table 2. Ordering information

| Type number | Package | | | Version |
|------------------------|---------|--|--|----------|
| | Name | Description | | |
| MFRC52201HN1/TRAYB[1] | HVQFN32 | plastic thermal enhanced very thin quad flat package; no leads; 32 terminal; body $5 \times 5 \times 0.85\text{ mm}$ | | SOT617-1 |
| MFRC52201HN1/TRAYBM[2] | HVQFN32 | plastic thermal enhanced very thin quad flat package; no leads; 32 terminal; body $5 \times 5 \times 0.85\text{ mm}$ | | SOT617-1 |
| MFRC52202HN1/TRAYB[1] | HVQFN32 | plastic thermal enhanced very thin quad flat package; no leads; 32 terminal; body $5 \times 5 \times 0.85\text{ mm}$ | | SOT617-1 |
| MFRC52202HN1/TRAYBM[2] | HVQFN32 | plastic thermal enhanced very thin quad flat package; no leads; 32 terminal; body $5 \times 5 \times 0.85\text{ mm}$ | | SOT617-1 |

[1] Delivered in one tray.

[2] Delivered in five trays.

6. Block diagram

The analog interface handles the modulation and demodulation of the analog signals.

The contactless UART manages the protocol requirements for the communication protocols in cooperation with the host. The FIFO buffer ensures fast and convenient data transfer to and from the host and the contactless UART and vice versa.

Various host interfaces are implemented to meet different customer requirements.

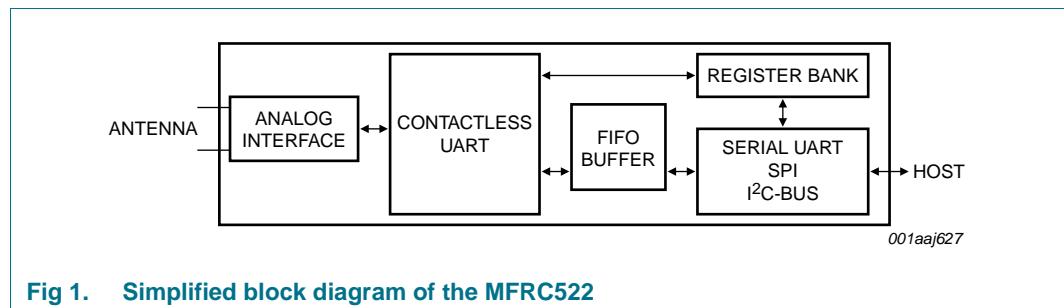


Fig 1. Simplified block diagram of the MFRC522

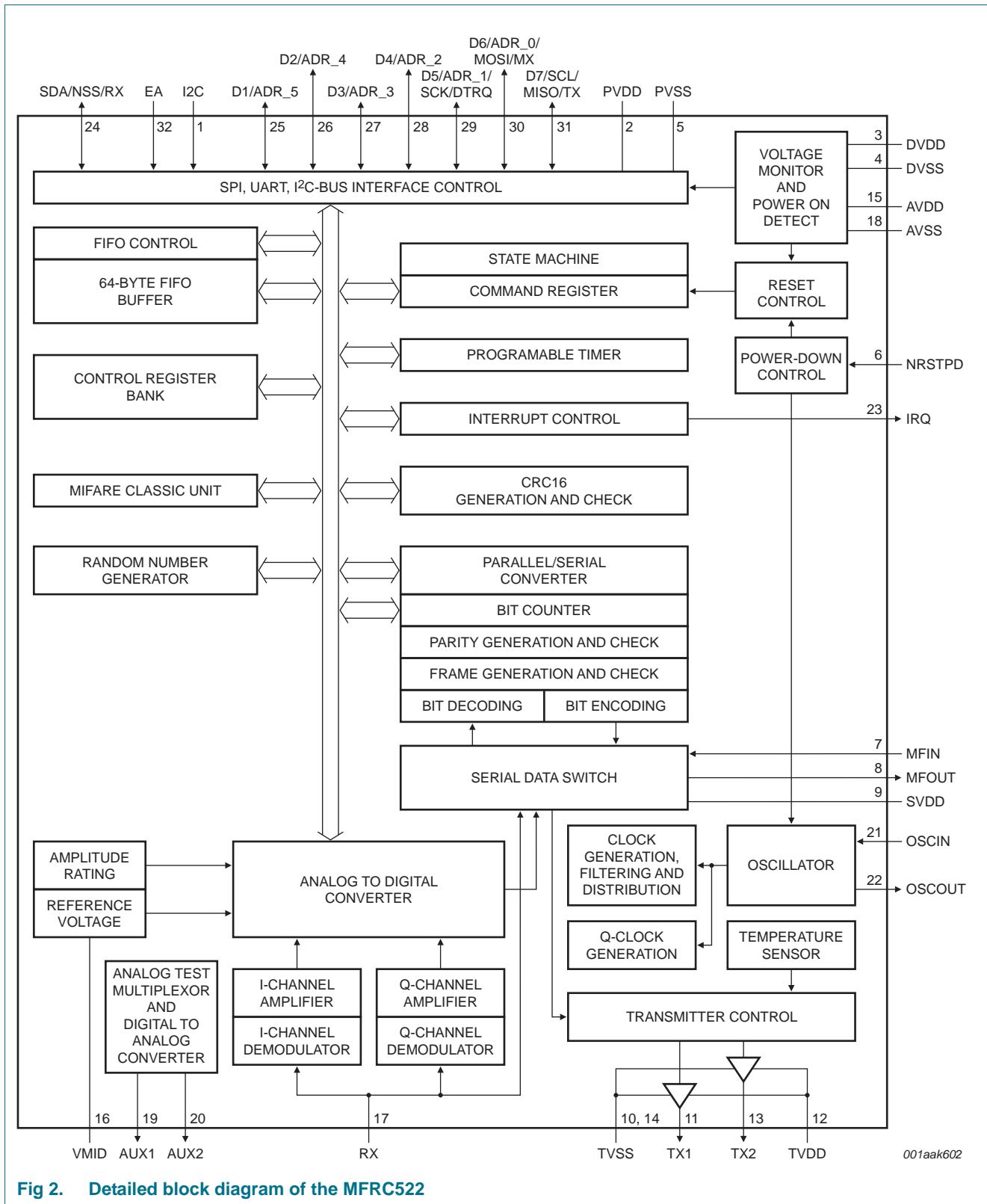


Fig 2. Detailed block diagram of the MFRC522

7. Pinning information

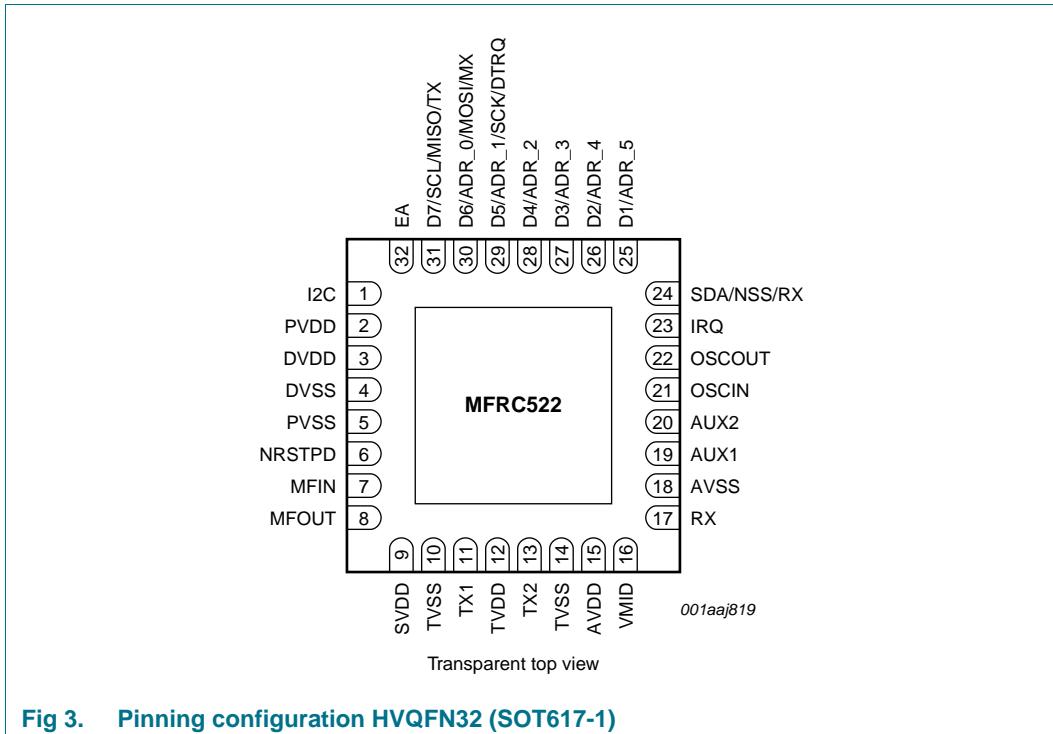


Fig 3. Pinning configuration HVQFN32 (SOT617-1)

7.1 Pin description

Table 3. Pin description

| Pin | Symbol | Type ^[1] | Description |
|-----|--------|---------------------|---|
| 1 | I2C | I | I ² C-bus enable input ^[2] |
| 2 | PVDD | P | pin power supply |
| 3 | DVDD | P | digital power supply |
| 4 | DVSS | G | digital ground ^[3] |
| 5 | PVSS | G | pin power supply ground |
| 6 | NRSTPD | I | reset and power-down input: power-down: enabled when LOW; internal current sinks are switched off, the oscillator is inhibited and the input pins are disconnected from the outside world reset: enabled by a positive edge |
| 7 | MFIN | I | MIFARE signal input |
| 8 | MFOUT | O | MIFARE signal output |
| 9 | SVDD | P | MFIN and MFOUT pin power supply |
| 10 | TVSS | G | transmitter output stage 1 ground |
| 11 | TX1 | O | transmitter 1 modulated 13.56 MHz energy carrier output |
| 12 | TVDD | P | transmitter power supply: supplies the output stage of transmitters 1 and 2 |
| 13 | TX2 | O | transmitter 2 modulated 13.56 MHz energy carrier output |
| 14 | TVSS | G | transmitter output stage 2 ground |
| 15 | AVDD | P | analog power supply |

Table 3. Pin description ...continued

| Pin | Symbol | Type ^[1] | Description |
|-----|--------|---------------------|---|
| 16 | VMID | P | internal reference voltage |
| 17 | RX | I | RF signal input |
| 18 | AVSS | G | analog ground |
| 19 | AUX1 | O | auxiliary outputs for test purposes |
| 20 | AUX2 | O | auxiliary outputs for test purposes |
| 21 | OSCIN | I | crystal oscillator inverting amplifier input; also the input for an externally generated clock ($f_{clk} = 27.12$ MHz) |
| 22 | OSCOUT | O | crystal oscillator inverting amplifier output |
| 23 | IRQ | O | interrupt request output: indicates an interrupt event |
| 24 | SDA | I/O | I ² C-bus serial data line input/output ^[2] |
| | NSS | I | SPI signal input ^[2] |
| | RX | I | UART address input ^[2] |
| 25 | D1 | I/O | test port ^[2] |
| | ADR_5 | I/O | I ² C-bus address 5 input ^[2] |
| 26 | D2 | I/O | test port |
| | ADR_4 | I | I ² C-bus address 4 input ^[2] |
| 27 | D3 | I/O | test port |
| | ADR_3 | I | I ² C-bus address 3 input ^[2] |
| 28 | D4 | I/O | test port |
| | ADR_2 | I | I ² C-bus address 2 input ^[2] |
| 29 | D5 | I/O | test port |
| | ADR_1 | I | I ² C-bus address 1 input ^[2] |
| | SCK | I | SPI serial clock input ^[2] |
| | DTRQ | O | UART request to send output to microcontroller ^[2] |
| 30 | D6 | I/O | test port |
| | ADR_0 | I | I ² C-bus address 0 input ^[2] |
| | MOSI | I/O | SPI master out, slave in ^[2] |
| | MX | O | UART output to microcontroller ^[2] |
| 31 | D7 | I/O | test port |
| | SCL | I/O | I ² C-bus clock input/output ^[2] |
| | MISO | I/O | SPI master in, slave out ^[2] |
| | TX | O | UART data output to microcontroller ^[2] |
| 32 | EA | I | external address input for coding I ² C-bus address ^[2] |

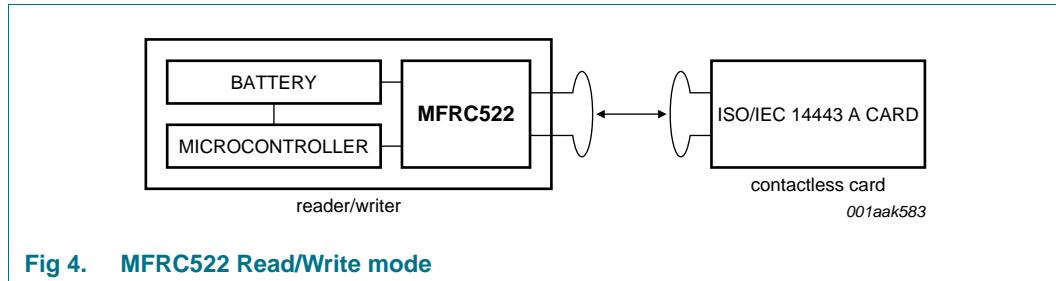
[1] Pin types: I = Input, O = Output, I/O = Input/Output, P = Power and G = Ground.

[2] The pin functionality of these pins is explained in [Section 8.1 “Digital interfaces”](#).

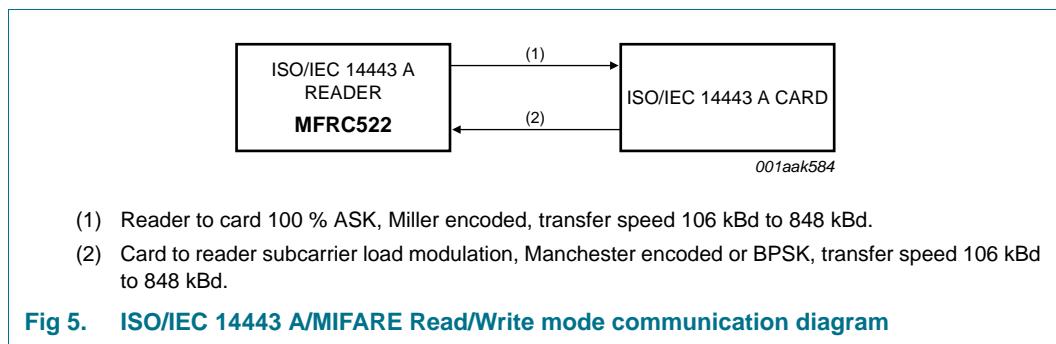
[3] Connection of heatsink pad on package bottom side is not necessary. Optional connection to pin DVSS is possible.

8. Functional description

The MFRC522 transmission module supports the Read/Write mode for ISO/IEC 14443 A/MIFARE using various transfer speeds and modulation protocols.



The physical level communication is shown in [Figure 5](#).



The physical parameters are described in [Table 4](#).

Table 4. Communication overview for ISO/IEC 14443 A/MIFARE reader/writer

| Communication direction | Signal type | Transfer speed | | | |
|---|------------------------|----------------------------|----------------------------|----------------------------|----------------------------|
| | | 106 kBd | 212 kBd | 424 kBd | 848 kBd |
| Reader to card (send data from the MFRC522 to a card) | reader side modulation | 100 % ASK | 100 % ASK | 100 % ASK | 100 % ASK |
| | bit encoding | modified Miller encoding | modified Miller encoding | modified Miller encoding | modified Miller encoding |
| | bit length | 128 (13.56 µs) | 64 (13.56 µs) | 32 (13.56 µs) | 16 (13.56 µs) |
| Card to reader (MFRC522 receives data from a card) | card side modulation | subcarrier load modulation | subcarrier load modulation | subcarrier load modulation | subcarrier load modulation |
| | subcarrier frequency | 13.56 MHz / 16 |
| | bit encoding | Manchester encoding | BPSK | BPSK | BPSK |

The MFRC522's contactless UART and dedicated external host must manage the complete ISO/IEC 14443 A/MIFARE protocol. [Figure 6](#) shows the data coding and framing according to ISO/IEC 14443 A/MIFARE.

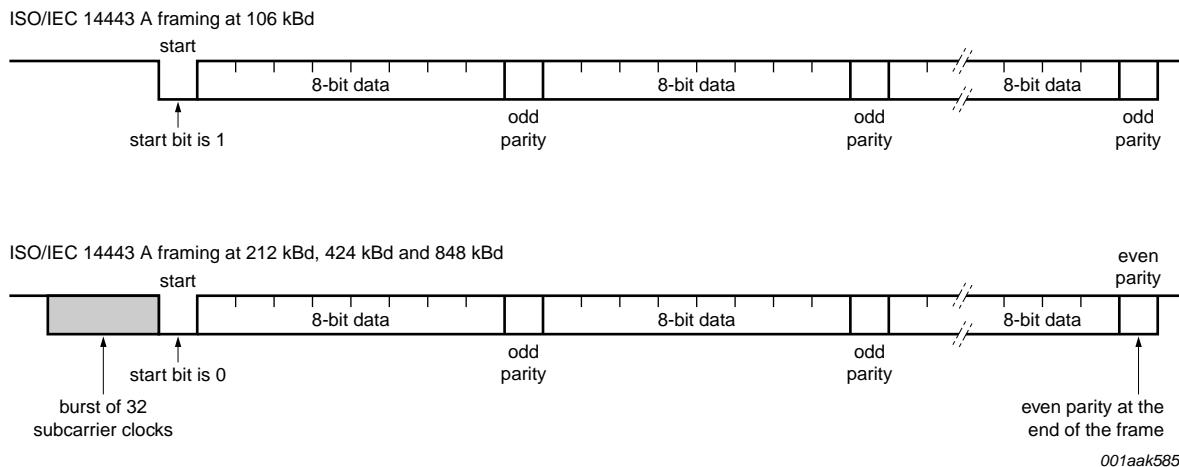


Fig 6. Data coding and framing according to ISO/IEC 14443 A

The internal CRC coprocessor calculates the CRC value based on ISO/IEC 14443 A part 3 and handles parity generation internally according to the transfer speed. Automatic parity generation can be switched off using the MfRxReg register's ParityDisable bit.

8.1 Digital interfaces

8.1.1 Automatic microcontroller interface detection

The MFRC522 supports direct interfacing of hosts using SPI, I²C-bus or serial UART interfaces. The MFRC522 resets its interface and checks the current host interface type automatically after performing a power-on or hard reset. The MFRC522 identifies the host interface by sensing the logic levels on the control pins after the reset phase. This is done using a combination of fixed pin connections. [Table 5](#) shows the different connection configurations.

Table 5. Connection protocol for detecting different interface types

| Pin | Interface type | | |
|------------------|----------------|--------------|----------------------------|
| | UART (input) | SPI (output) | I ² C-bus (I/O) |
| SDA | RX | NSS | SDA |
| I ² C | 0 | 0 | 1 |
| EA | 0 | 1 | EA |
| D7 | TX | MISO | SCL |
| D6 | MX | MOSI | ADR_0 |
| D5 | DTRQ | SCK | ADR_1 |
| D4 | - | - | ADR_2 |
| D3 | - | - | ADR_3 |
| D2 | - | - | ADR_4 |
| D1 | - | - | ADR_5 |

8.1.2 Serial Peripheral Interface

A serial peripheral interface (SPI compatible) is supported to enable high-speed communication to the host. The interface can handle data speeds up to 10 Mbit/s. When communicating with a host, the MFRC522 acts as a slave, receiving data from the external host for register settings, sending and receiving data relevant for RF interface communication.

An interface compatible with SPI enables high-speed serial communication between the MFRC522 and a microcontroller. The implemented interface is in accordance with the SPI standard.

The timing specification is given in [Section 14.1 on page 78](#).

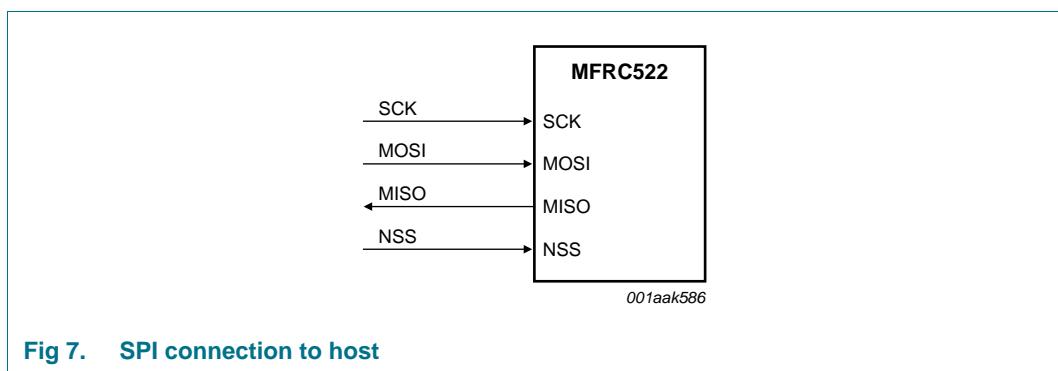


Fig 7. SPI connection to host

The MFRC522 acts as a slave during SPI communication. The SPI clock signal SCK must be generated by the master. Data communication from the master to the slave uses the MOSI line. The MISO line is used to send data from the MFRC522 to the master.

Data bytes on both MOSI and MISO lines are sent with the MSB first. Data on both MOSI and MISO lines must be stable on the rising edge of the clock and can be changed on the falling edge. Data is provided by the MFRC522 on the falling clock edge and is stable during the rising clock edge.

8.1.2.1 SPI read data

Reading data using SPI requires the byte order shown in [Table 6](#) to be used. It is possible to read out up to n-data bytes.

The first byte sent defines both the mode and the address.

Table 6. MOSI and MISO byte order

| Line | Byte 0 | Byte 1 | Byte 2 | To | Byte n | Byte n + 1 |
|------|------------------|-----------|-----------|-----|------------|------------|
| MOSI | address 0 | address 1 | address 2 | ... | address n | 00 |
| MISO | X ^[1] | data 0 | data 1 | ... | data n – 1 | data n |

[1] X = Do not care.

Remark: The MSB must be sent first.

8.1.2.2 SPI write data

To write data to the MFRC522 using SPI requires the byte order shown in [Table 7](#). It is possible to write up to n data bytes by only sending one address byte.

The first send byte defines both the mode and the address byte.

Table 7. MOSI and MISO byte order

| Line | Byte 0 | Byte 1 | Byte 2 | To | Byte n | Byte n + 1 |
|------|------------------|------------------|------------------|-----|------------------|------------------|
| MOSI | address 0 | data 0 | data 1 | ... | data n – 1 | data n |
| MISO | X ^[1] | X ^[1] | X ^[1] | ... | X ^[1] | X ^[1] |

[1] X = Do not care.

Remark: The MSB must be sent first.

8.1.2.3 SPI address byte

The address byte must meet the following format.

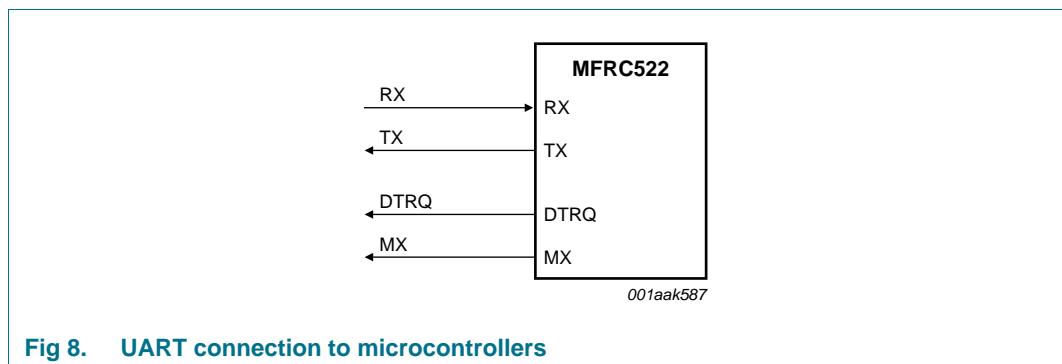
The MSB of the first byte defines the mode used. To read data from the MFRC522 the MSB is set to logic 1. To write data to the MFRC522 the MSB must be set to logic 0. Bits 6 to 1 define the address and the LSB is set to logic 0.

Table 8. Address byte 0 register; address MOSI

| 7 (MSB) | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
|-----------------------|---|---|---|---|---|---|---------|
| 1 = read 0 = write | | | | | | | 0 |

8.1.3 UART interface

8.1.3.1 Connection to a host



Remark: Signals DTRQ and MX can be disabled by clearing TestPinEnReg register's RS232LineEn bit.

8.1.3.2 Selectable UART transfer speeds

The internal UART interface is compatible with an RS232 serial interface.

The default transfer speed is 9.6 kBd. To change the transfer speed, the host controller must write a value for the new transfer speed to the SerialSpeedReg register. Bits BR_T0[2:0] and BR_T1[4:0] define the factors for setting the transfer speed in the SerialSpeedReg register.

The BR_T0[2:0] and BR_T1[4:0] settings are described in [Table 9](#). Examples of different transfer speeds and the relevant register settings are given in [Table 10](#).

Table 9. BR_T0 and BR_T1 settings

| BR_Tn | Bit 0 | Bit 1 | Bit 2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 |
|--------------|---------|----------|----------|----------|----------|----------|----------|----------|
| BR_T0 factor | 1 | 1 | 2 | 4 | 8 | 16 | 32 | 64 |
| BR_T1 range | 1 to 32 | 33 to 64 |

Table 10. Selectable UART transfer speeds

| Transfer speed (kBd) | SerialSpeedReg value | | Transfer speed accuracy (%) ^[1] |
|----------------------|----------------------|-------------|--|
| | Decimal | Hexadecimal | |
| 7.2 | 250 | FAh | -0.25 |
| 9.6 | 235 | EBh | 0.32 |
| 14.4 | 218 | DAh | -0.25 |
| 19.2 | 203 | CBh | 0.32 |
| 38.4 | 171 | ABh | 0.32 |
| 57.6 | 154 | 9Ah | -0.25 |
| 115.2 | 122 | 7Ah | -0.25 |
| 128 | 116 | 74h | -0.06 |
| 230.4 | 90 | 5Ah | -0.25 |
| 460.8 | 58 | 3Ah | -0.25 |
| 921.6 | 28 | 1Ch | 1.45 |
| 1228.8 | 21 | 15h | 0.32 |

[1] The resulting transfer speed error is less than 1.5 % for all described transfer speeds.

The selectable transfer speeds shown in [Table 10](#) are calculated according to the following equations:

If BR_T0[2:0] = 0:

$$\text{transfer speed} = \frac{27.12 \times 10^6}{(BR_T0 + 1)} \quad (1)$$

If BR_T0[2:0] > 0:

$$\text{transfer speed} = \left(\frac{27.12 \times 10^6}{(BR_T1 + 33)} \right) \cdot 2^{(BR_T0 - 1)} \quad (2)$$

Remark: Transfer speeds above 1228.8 kBd are not supported.

8.1.3.3 UART framing

Table 11. UART framing

| Bit | Length | Value |
|-------|--------|-------|
| Start | 1-bit | 0 |
| Data | 8 bits | data |
| Stop | 1-bit | 1 |

Remark: The LSB for data and address bytes must be sent first. No parity bit is used during transmission.

Read data: To read data using the UART interface, the flow shown in [Table 12](#) must be used. The first byte sent defines both the mode and the address.

Table 12. Read data byte order

| Pin | Byte 0 | Byte 1 |
|-------------|---------|--------|
| RX (pin 24) | address | - |
| TX (pin 31) | - | data 0 |

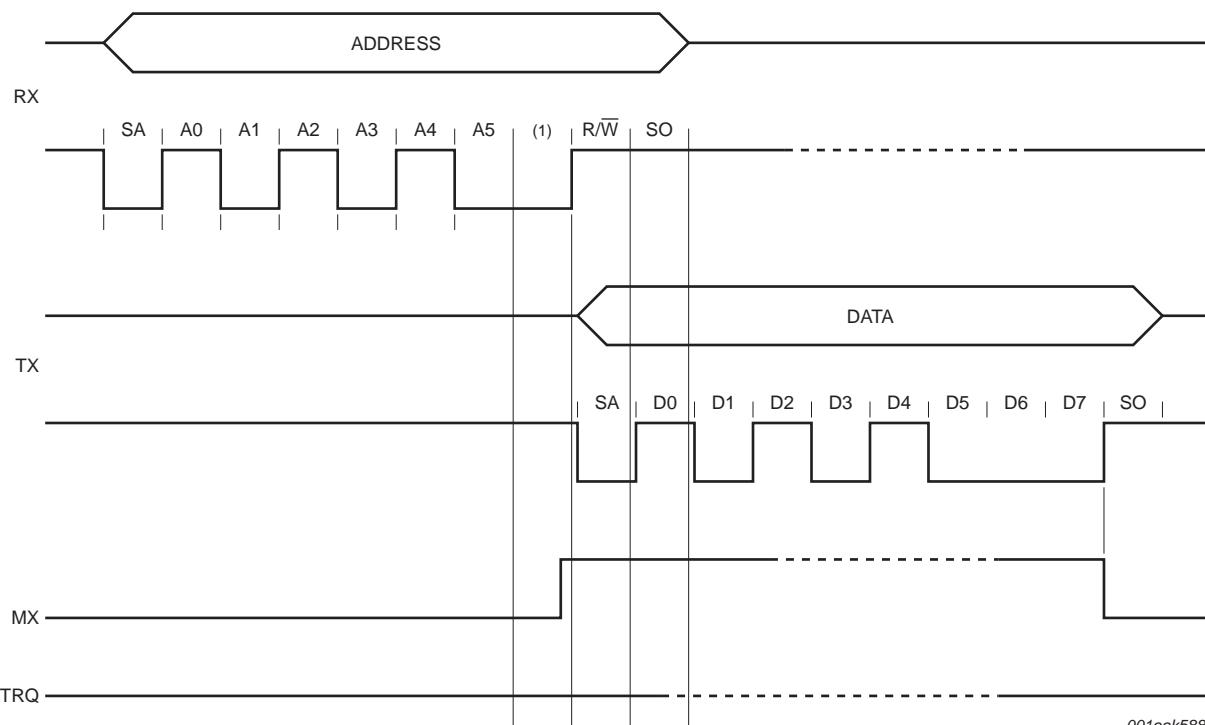


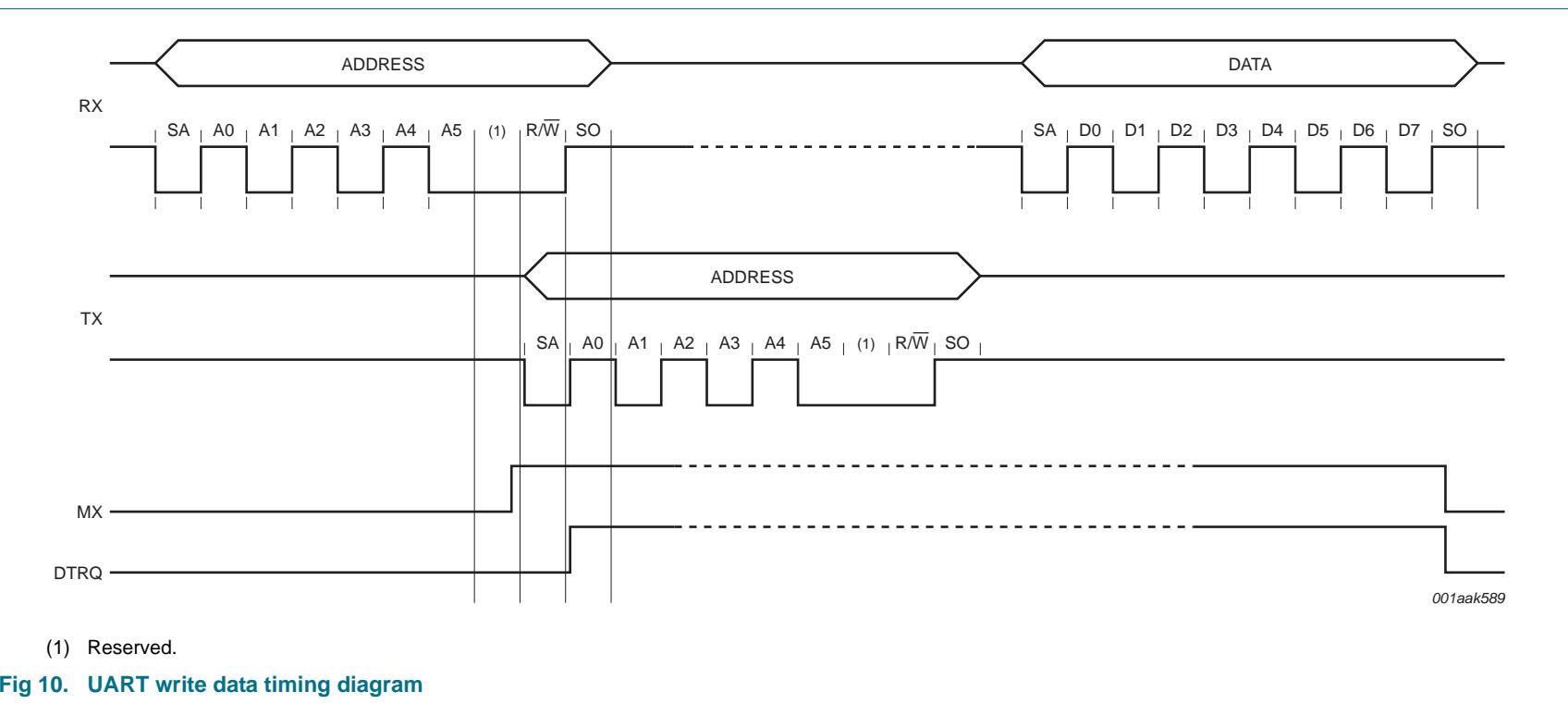
Fig 9. UART read data timing diagram

Write data: To write data to the MFRC522 using the UART interface, the structure shown in [Table 13](#) must be used.

The first byte sent defines both the mode and the address.

Table 13. Write data byte order

| Pin | Byte 0 | Byte 1 |
|-------------|-----------|-----------|
| RX (pin 24) | address 0 | data 0 |
| TX (pin 31) | - | address 0 |



(1) Reserved.

Fig 10. UART write data timing diagram

Remark: The data byte can be sent directly after the address byte on pin RX.

Address byte: The address byte has to meet the following format:

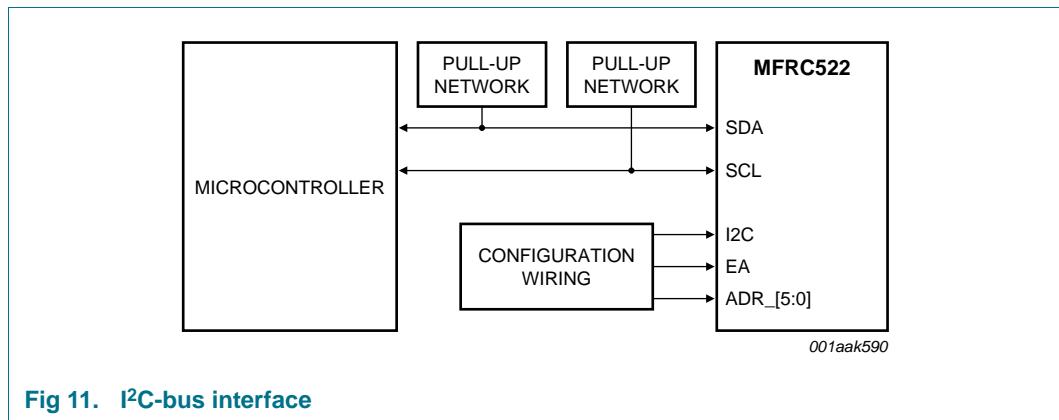
The MSB of the first byte sets the mode used. To read data from the MFRC522, the MSB is set to logic 1. To write data to the MFRC522 the MSB is set to logic 0. Bit 6 is reserved for future use, and bits 5 to 0 define the address; see [Table 14](#).

Table 14. Address byte 0 register; address MOSI

| 7 (MSB) | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
|-----------------------|----------|---------|---|---|---|---|---------|
| 1 = read 0 = write | reserved | address | | | | | |

8.1.4 I²C-bus interface

An I²C-bus (Inter-IC) interface is supported to enable a low-cost, low pin count serial bus interface to the host. The I²C-bus interface is implemented according to NXP Semiconductors' *I²C-bus interface specification, rev. 2.1, January 2000*. The interface can only act in Slave mode. Therefore the MFRC522 does not implement clock generation or access arbitration.

**Fig 11.** I²C-bus interface

The MFRC522 can act either as a slave receiver or slave transmitter in Standard mode, Fast mode and High-speed mode.

SDA is a bidirectional line connected to a positive supply voltage using a current source or a pull-up resistor. Both SDA and SCL lines are set HIGH when data is not transmitted. The MFRC522 has a 3-state output stage to perform the wired-AND function. Data on the I²C-bus can be transferred at data rates of up to 100 kBd in Standard mode, up to 400 kBd in Fast mode or up to 3.4 Mbit/s in High-speed mode.

If the I²C-bus interface is selected, spike suppression is activated on lines SCL and SDA as defined in the I²C-bus interface specification.

See [Table 155 on page 79](#) for timing requirements.

8.1.4.1 Data validity

Data on the SDA line must be stable during the HIGH clock period. The HIGH or LOW state of the data line must only change when the clock signal on SCL is LOW.

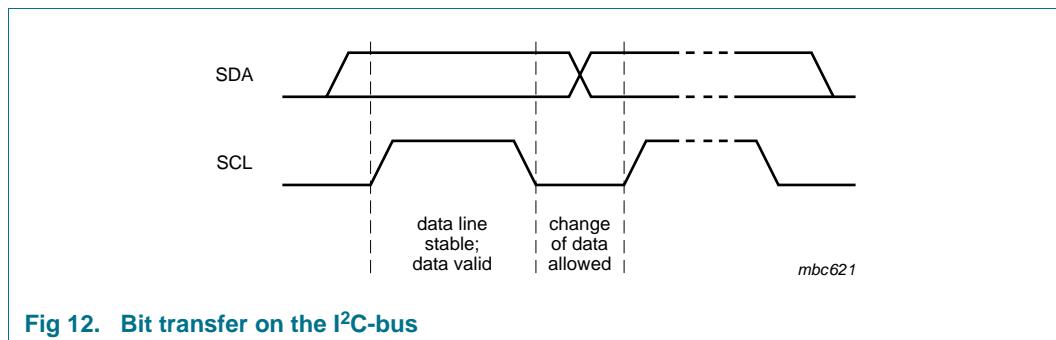


Fig 12. Bit transfer on the I²C-bus

8.1.4.2 START and STOP conditions

To manage the data transfer on the I²C-bus, unique START (S) and STOP (P) conditions are defined.

- A START condition is defined with a HIGH-to-LOW transition on the SDA line while SCL is HIGH.
- A STOP condition is defined with a LOW-to-HIGH transition on the SDA line while SCL is HIGH.

The I²C-bus master always generates the START and STOP conditions. The bus is busy after the START condition. The bus is free again a certain time after the STOP condition.

The bus stays busy if a repeated START (Sr) is generated instead of a STOP condition. The START (S) and repeated START (Sr) conditions are functionally identical. Therefore, S is used as a generic term to represent both the START (S) and repeated START (Sr) conditions.

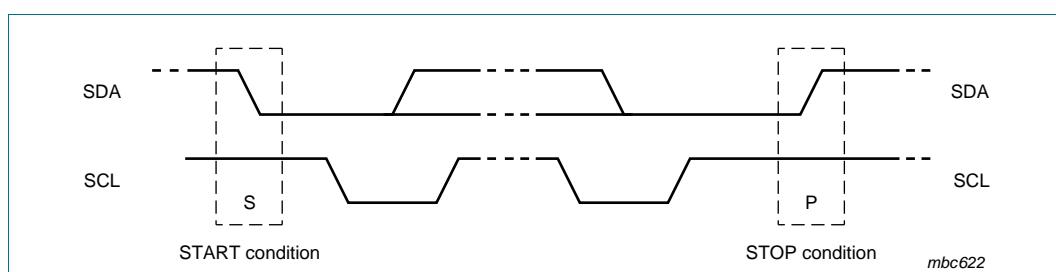


Fig 13. START and STOP conditions

8.1.4.3 Byte format

Each byte must be followed by an acknowledge bit. Data is transferred with the MSB first; see [Figure 16](#). The number of transmitted bytes during one data transfer is unrestricted but must meet the read/write cycle format.

8.1.4.4 Acknowledge

An acknowledge must be sent at the end of one data byte. The acknowledge-related clock pulse is generated by the master. The transmitter of data, either master or slave, releases the SDA line (HIGH) during the acknowledge clock pulse. The receiver pulls down the SDA line during the acknowledge clock pulse so that it remains stable LOW during the HIGH period of this clock pulse.

The master can then generate either a STOP (P) condition to stop the transfer or a repeated START (Sr) condition to start a new transfer.

A master-receiver indicates the end of data to the slave-transmitter by not generating an acknowledge on the last byte that was clocked out by the slave. The slave-transmitter releases the data line to allow the master to generate a STOP (P) or repeated START (Sr) condition.

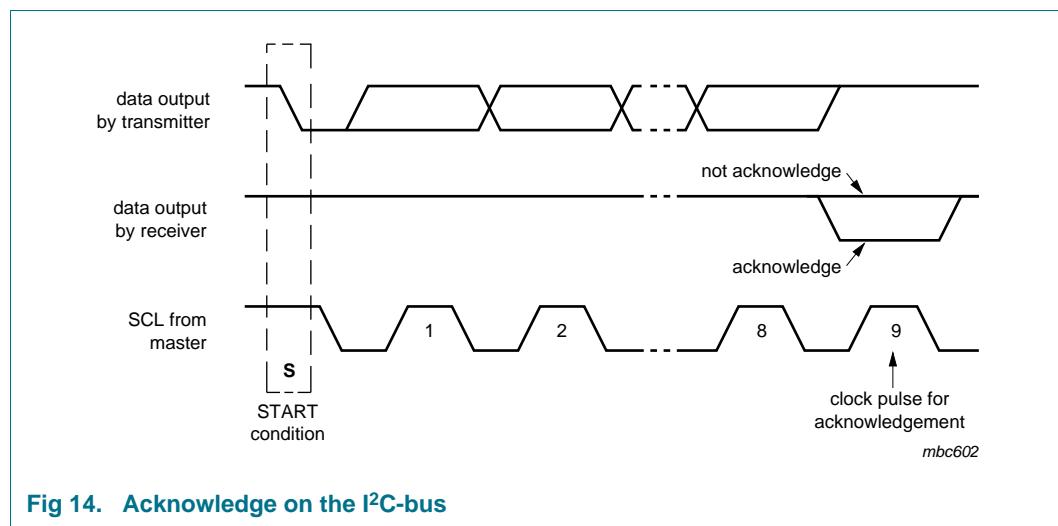


Fig 14. Acknowledge on the I²C-bus

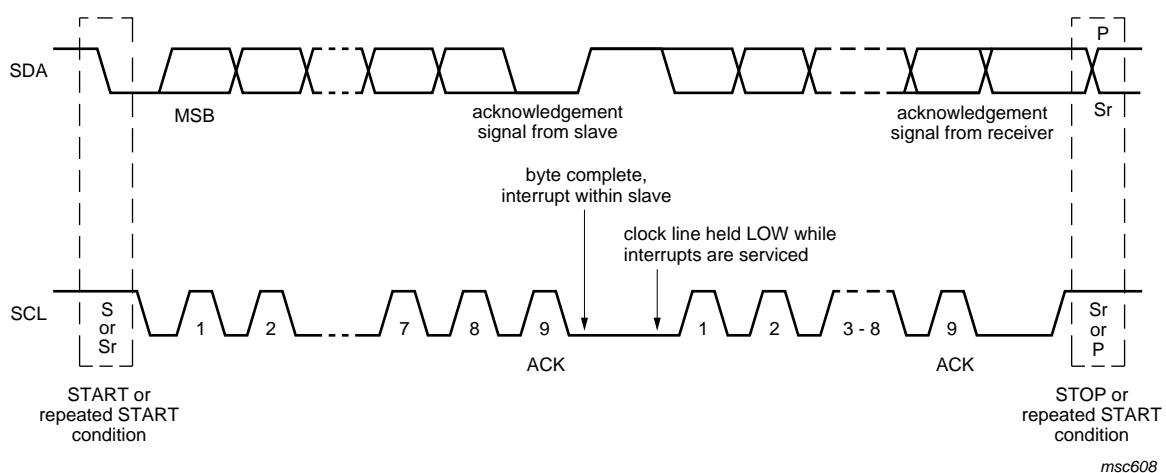


Fig 15. Data transfer on the I²C-bus

8.1.4.5 7-Bit addressing

During the I²C-bus address procedure, the first byte after the START condition is used to determine which slave will be selected by the master.

Several address numbers are reserved. During device configuration, the designer must ensure that collisions with these reserved addresses cannot occur. Check the *I²C-bus specification* for a complete list of reserved addresses.

The I²C-bus address specification is dependent on the definition of pin EA. Immediately after releasing pin NRSTPD or after a power-on reset, the device defines the I²C-bus address according to pin EA.

If pin EA is set LOW, the upper 4 bits of the device bus address are reserved by NXP Semiconductors and set to 0101b for all MFRC522 devices. The remaining 3 bits (ADR_0, ADR_1, ADR_2) of the slave address can be freely configured by the customer to prevent collisions with other I²C-bus devices.

If pin EA is set HIGH, ADR_0 to ADR_5 can be completely specified at the external pins according to [Table 5 on page 9](#). ADR_6 is always set to logic 0.

In both modes, the external address coding is latched immediately after releasing the reset condition. Further changes at the used pins are not taken into consideration. Depending on the external wiring, the I²C-bus address pins can be used for test signal outputs.

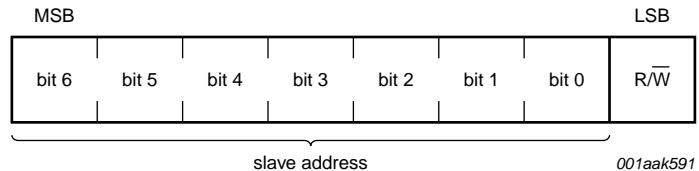


Fig 16. First byte following the START procedure

8.1.4.6 Register write access

To write data from the host controller using the I²C-bus to a specific register in the MFRC522 the following frame format must be used.

- The first byte of a frame indicates the device address according to the I²C-bus rules.
- The second byte indicates the register address followed by up to n-data bytes.

In one frame all data bytes are written to the same register address. This enables fast FIFO buffer access. The Read/Write (R/\overline{W}) bit is set to logic 0.

8.1.4.7 Register read access

To read out data from a specific register address in the MFRC522, the host controller must use the following procedure:

- Firstly, a write access to the specific register address must be performed as indicated in the frame that follows
- The first byte of a frame indicates the device address according to the I²C-bus rules
- The second byte indicates the register address. No data bytes are added
- The Read/Write bit is 0

After the write access, read access can start. The host sends the device address of the MFRC522. In response, the MFRC522 sends the content of the read access register. In one frame all data bytes can be read from the same register address. This enables fast FIFO buffer access or register polling.

The Read/Write (R/W) bit is set to logic 1.

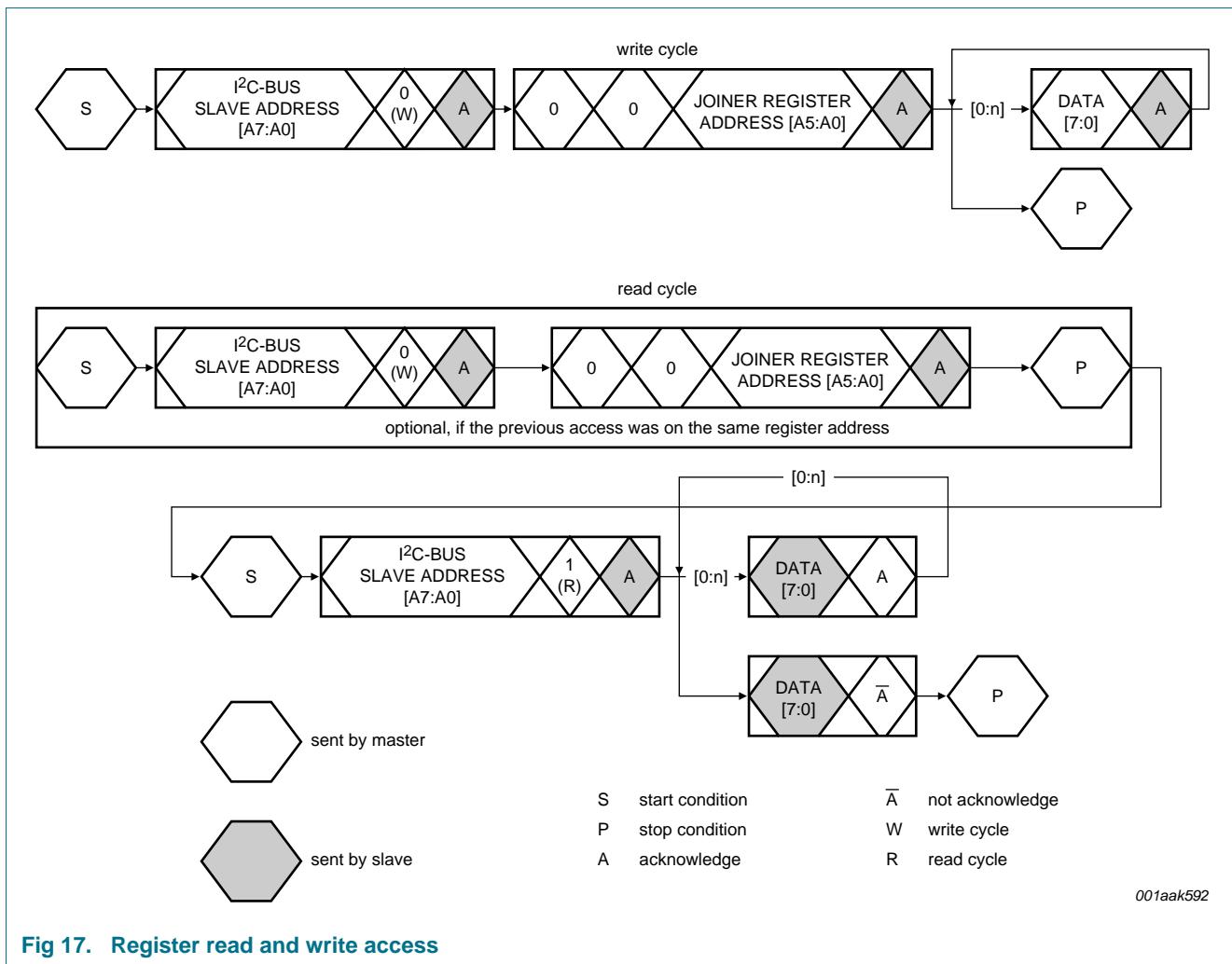


Fig 17. Register read and write access

8.1.4.8 High-speed mode

In High-speed mode (HS mode), the device can transfer information at data rates of up to 3.4 Mbit/s, while remaining fully downward-compatible with Fast or Standard mode (F/S mode) for bidirectional communication in a mixed-speed bus system.

8.1.4.9 High-speed transfer

To achieve data rates of up to 3.4 Mbit/s the following improvements have been made to I²C-bus operation.

- The inputs of the device in HS mode incorporate spike suppression, a Schmitt trigger on the SDA and SCL inputs and different timing constants when compared to F/S mode
- The output buffers of the device in HS mode incorporate slope control of the falling edges of the SDA and SCL signals with different fall times compared to F/S mode

8.1.4.10 Serial data transfer format in HS mode

The HS mode serial data transfer format meets the Standard mode I²C-bus specification. HS mode can only start after all of the following conditions (all of which are in F/S mode):

1. START condition (S)
2. 8-bit master code (00001XXXb)
3. Not-acknowledge bit (\bar{A})

When HS mode starts, the active master sends a repeated START condition (Sr) followed by a 7-bit slave address with a R/W bit address and receives an acknowledge bit (A) from the selected MFRC522.

Data transfer continues in HS mode after the next repeated START (Sr), only switching back to F/S mode after a STOP condition (P). To reduce the overhead of the master code, a master links a number of HS mode transfers, separated by repeated START conditions (Sr).

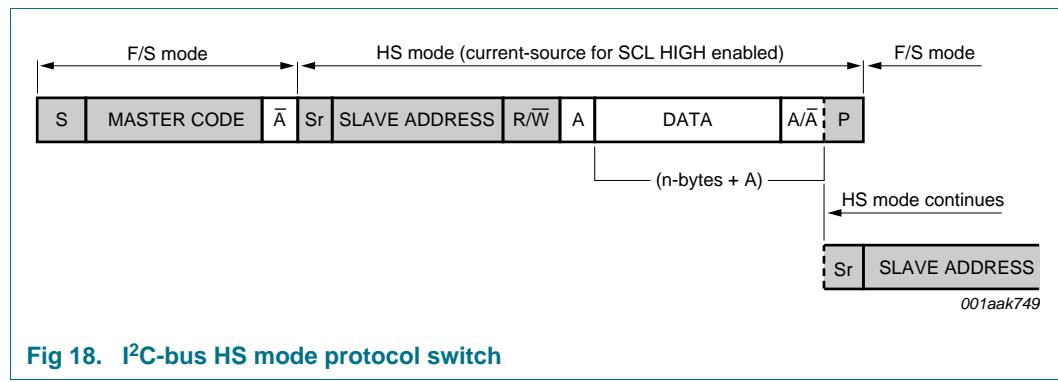
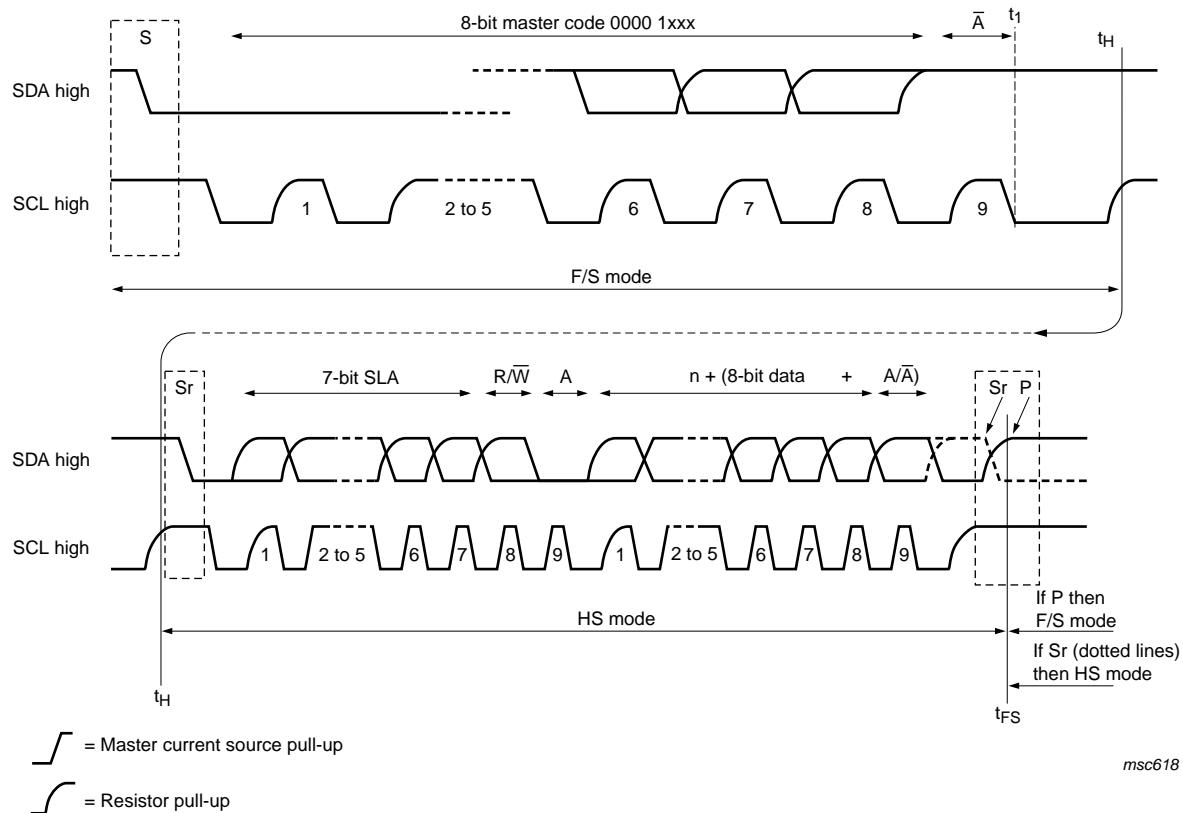


Fig 18. I²C-bus HS mode protocol switch

Fig 19. I²C-bus HS mode protocol frame

8.1.4.11 Switching between F/S mode and HS mode

After reset and initialization, the MFRC522 is in Fast mode (which is in effect F/S mode as Fast mode is downward-compatible with Standard mode). The connected MFRC522 recognizes the “S 00001XXX A” sequence and switches its internal circuitry from the Fast mode setting to the HS mode setting.

The following actions are taken:

1. Adapt the SDA and SCL input filters according to the spike suppression requirement in HS mode.
2. Adapt the slope control of the SDA output stages.

It is possible for system configurations that do not have other I²C-bus devices involved in the communication to switch to HS mode permanently. This is implemented by setting Status2Reg register's I²CForceHS bit to logic 1. In permanent HS mode, the master code is not required to be sent. This is not defined in the specification and must only be used when no other devices are connected on the bus. In addition, spikes on the I²C-bus lines must be avoided because of the reduced spike suppression.

8.1.4.12 MFRC522 at lower speed modes

MFRC522 is fully downward-compatible and can be connected to an F/S mode I²C-bus system. The device stays in F/S mode and communicates at F/S mode speeds because a master code is not transmitted in this configuration.

8.2 Analog interface and contactless UART

8.2.1 General

The integrated contactless UART supports the external host online with framing and error checking of the protocol requirements up to 848 kBd. An external circuit can be connected to the communication interface pins MFIN and MFOUT to modulate and demodulate the data.

The contactless UART handles the protocol requirements for the communication protocols in cooperation with the host. Protocol handling generates bit and byte-oriented framing. In addition, it handles error detection such as parity and CRC, based on the various supported contactless communication protocols.

Remark: The size and tuning of the antenna and the power supply voltage have an important impact on the achievable operating distance.

8.2.2 TX p-driver

The signal on pins TX1 and TX2 is the 13.56 MHz energy carrier modulated by an envelope signal. It can be used to drive an antenna directly using a few passive components for matching and filtering; see [Section 15 on page 81](#). The signal on pins TX1 and TX2 can be configured using the TxControlReg register; see [Section 9.3.2.5 on page 50](#).

The modulation index can be set by adjusting the impedance of the drivers. The impedance of the p-driver can be configured using registers CWGsPReg and ModGsPReg. The impedance of the n-driver can be configured using the GsNReg register. The modulation index also depends on the antenna design and tuning.

The TxModeReg and TxSelReg registers control the data rate and framing during transmission and the antenna driver setting to support the different requirements at the different modes and transfer speeds.

Table 15. Register and bit settings controlling the signal on pin TX1

| Bit Tx1RFEn | Bit Force 100ASK | Bit InvTx1RFOn | Bit InvTx1RFOff | Envelope | Pin TX1 | GSPMos | GSNMos | Remarks |
|-------------|------------------|----------------|-----------------|----------|---------|--------|--------|--|
| 0 | X[1] | X[1] | X[1] | X[1] | X[1] | X[1] | X[1] | not specified if RF is switched off |
| 1 | 0 | 0 | X[1] | 0 | RF | pMod | nMod | 100 % ASK: pin TX1 pulled to logic 0, independent of the InvTx1RFOff bit |
| | | | | | RF | pCW | nCW | |
| | 0 | 1 | X[1] | 0 | RF | pMod | nMod | |
| | | | | | RF | pCW | nCW | |
| | 1 | 1 | X[1] | 0 | 0 | pMod | nMod | |
| | | | | | RF_n | pCW | nCW | |

[1] X = Do not care.

Table 16. Register and bit settings controlling the signal on pin TX2

| Bit Tx1RFEn | Bit Force 100ASK | Bit Tx2CW | Bit InvTx2RFOOn | Bit InvTx2RFOff | Envelope | Pin TX2 | GSPMos | GSMos | Remarks |
|-------------|------------------|-----------|-----------------|-----------------|----------|---------|--------|-------|--|
| 0 | X[1] | X[1] | X[1] | X[1] | X[1] | X[1] | X[1] | X[1] | not specified if RF is switched off |
| 1 | 0 | 0 | 0 | X[1] | 0 | RF | pMod | nMod | - |
| | | | | | | RF | pCW | nCW | |
| | | | 1 | X[1] | 0 | RF_n | pMod | nMod | |
| | | | | | | RF_n | pCW | nCW | |
| | | | 1 | 0 | X[1] | RF | pCW | nCW | conductance always CW for the Tx2CW bit |
| | | | | | | RF_n | pCW | nCW | |
| | 1 | 0 | 0 | X[1] | 0 | 0 | 0 | pMod | nMod |
| | | | | | | RF | pCW | nCW | 100 % ASK: pin TX2 pulled to logic 0 (independent of the InvTx2RFOOn/InvTx2RFOff bits) |
| | | | 1 | X[1] | 0 | 0 | 0 | pMod | nMod |
| | | | | | | RF_n | pCW | nCW | |
| | | | 1 | X[1] | X[1] | RF | pCW | nCW | |
| | | | | | | RF_n | pCW | nCW | |

[1] X = Do not care.

The following abbreviations have been used in [Table 15](#) and [Table 16](#):

- RF: 13.56 MHz clock derived from 27.12 MHz quartz crystal oscillator divided by 2
- RF_n: inverted 13.56 MHz clock
- GSPMos: conductance, configuration of the PMOS array
- GSMMos: conductance, configuration of the NMOS array
- pCW: PMOS conductance value for continuous wave defined by the CWGsPReg register
- pMod: PMOS conductance value for modulation defined by the ModGsPReg register
- nCW: NMOS conductance value for continuous wave defined by the GsNReg register's CWGsN[3:0] bits
- nMod: NMOS conductance value for modulation defined by the GsNReg register's ModGsN[3:0] bits
- X = do not care.

Remark: If only one driver is switched on, the values for CWGsPReg, ModGsPReg and GsNReg registers are used for both drivers.

8.2.3 Serial data switch

Two main blocks are implemented in the MFRC522. The digital block comprises the state machines, encoder/decoder logic. The analog block comprises the modulator and antenna drivers, the receiver and amplifiers. It is possible for the interface between these two blocks to be configured so that the interfacing signals are routed to pins MFIN and MFOUT.

This topology allows the analog block of the MFRC522 to be connected to the digital block of another device.

The serial signal switch is controlled by the TxSelReg and RxSelReg registers.

[Figure 20](#) shows the serial data switch for p-driver TX1 and TX2.

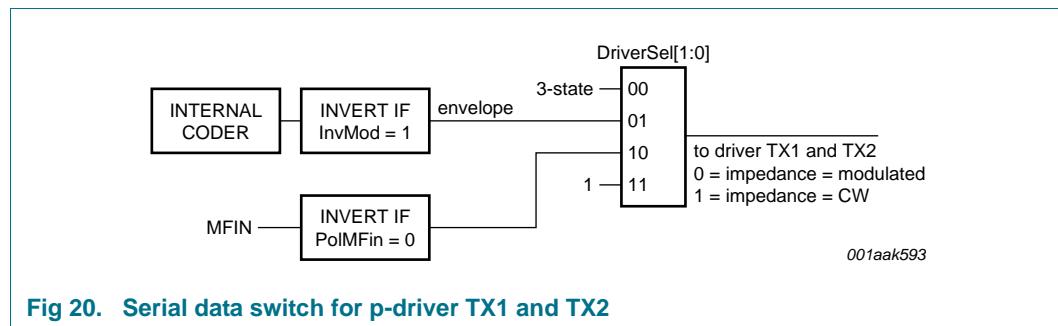


Fig 20. Serial data switch for p-driver TX1 and TX2

8.2.4 MFIN and MFOUT interface support

The MFRC522 is divided into a digital circuit block and an analog circuit block. The digital block contains state machines, encoder and decoder logic and so on. The analog block contains the modulator and antenna drivers, receiver and amplifiers. The interface between these two blocks can be configured so that the interfacing signals can be routed to pins MFIN and MFOUT; see [Figure 21 on page 28](#). This configuration is implemented using TxSelReg register's MFOutSel[3:0] and DriverSel[1:0] bits and RxSelReg register's UARTSel[1:0] bits.

This topology allows some parts of the analog block to be connected to the digital block of another device.

Switch MFOutSel in the TxSelReg register can be used to measure MIFARE and ISO/IEC14443 A related signals. This is especially important during the design-in phase or for test purposes as it enables checking of the transmitted and received data.

The most important use of pins MFIN and MFOUT is found in the active antenna concept. An external active antenna circuit can be connected to the MFRC522's digital block. Switch MFOutSel must be configured so that the internal Miller encoded signal is sent to pin MFOUT (MFOutSel = 100b). UARTSel[1:0] must be configured to receive a Manchester signal with subcarrier from pin MFIN (UARTSel[1:0] = 01).

It is possible to connect a passive antenna to pins TX1, TX2 and RX (using the appropriate filter and matching circuit) and an active antenna to pins MFOUT and MFIN at the same time. In this configuration, two RF circuits can be driven (one after another) by a single host processor.

Remark: Pins MFIN and MFOUT have a dedicated supply on pin SVDD with the ground on pin PVSS. If pin MFIN is not used it must be connected to either pin SVDD or pin PVSS. If pin SVDD is not used it must be connected to either pin DVDD, pin PVDD or any other voltage supply pin.

001aak594

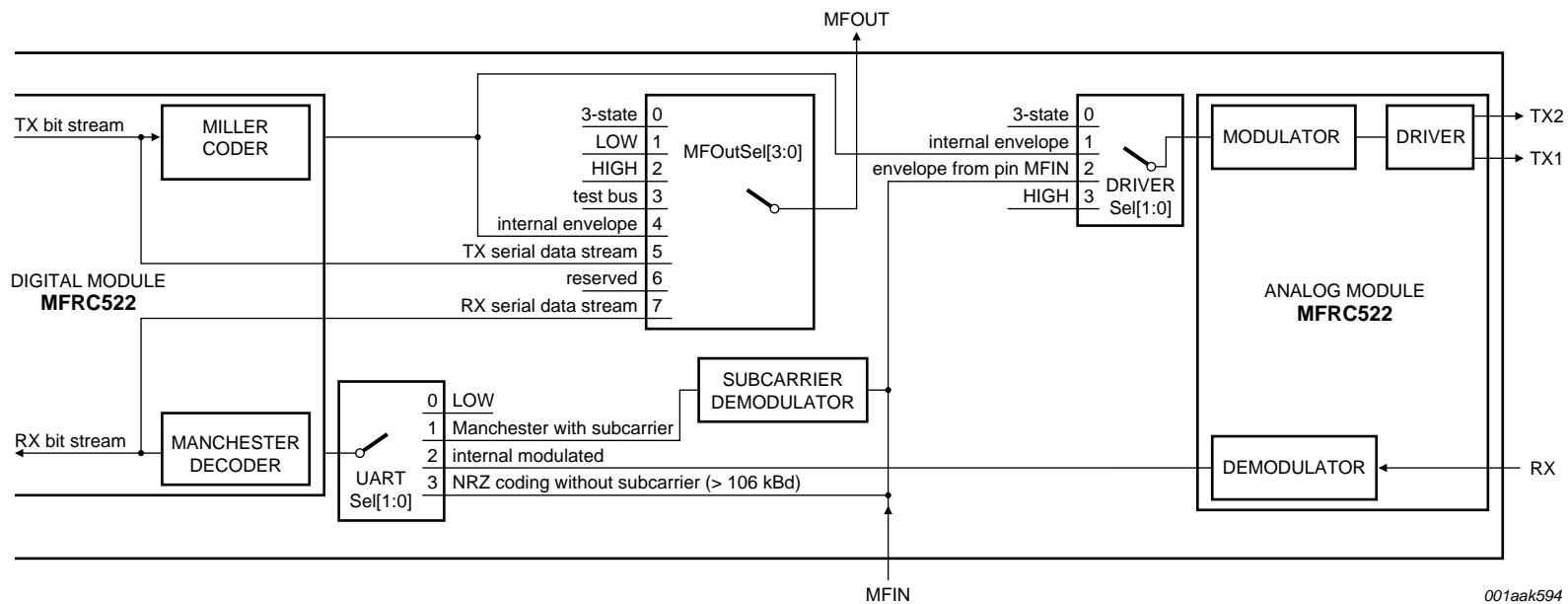


Fig 21. Overview of MFIN and MFOUT signal routing

8.2.5 CRC coprocessor

The following CRC coprocessor parameters can be configured:

- The CRC preset value can be either 0000h, 6363h, A671h or FFFFh depending on the ModeReg register's CRCPreset[1:0] bits setting
- The CRC polynomial for the 16-bit CRC is fixed to $x^{16} + x^{12} + x^5 + 1$
- The CRCResultReg register indicates the result of the CRC calculation. This register is split into two 8-bit registers representing the higher and lower bytes.
- The ModeReg register's MSBFFirst bit indicates that data will be loaded with the MSB first.

Table 17. CRC coprocessor parameters

| Parameter | Value |
|---------------------|---|
| CRC register length | 16-bit CRC |
| CRC algorithm | algorithm according to ISO/IEC 14443 A and ITU-T |
| CRC preset value | 0000h, 6363h, A671h or FFFFh depending on the setting of the ModeReg register's CRCPreset[1:0] bits |

8.3 FIFO buffer

An 8×64 bit FIFO buffer is used in the MFRC522. It buffers the input and output data stream between the host and the MFRC522's internal state machine. This makes it possible to manage data streams up to 64 bytes long without the need to take timing constraints into account.

8.3.1 Accessing the FIFO buffer

The FIFO buffer input and output data bus is connected to the FIFODataReg register. Writing to this register stores one byte in the FIFO buffer and increments the internal FIFO buffer write pointer. Reading from this register shows the FIFO buffer contents stored in the FIFO buffer read pointer and decrements the FIFO buffer read pointer. The distance between the write and read pointer can be obtained by reading the FIFOLevelReg register.

When the microcontroller starts a command, the MFRC522 can, while the command is in progress, access the FIFO buffer according to that command. Only one FIFO buffer has been implemented which can be used for input and output. The microcontroller must ensure that there are not any unintentional FIFO buffer accesses.

8.3.2 Controlling the FIFO buffer

The FIFO buffer pointers can be reset by setting FIFOLevelReg register's FlushBuffer bit to logic 1. Consequently, the FIFOLevel[6:0] bits are all set to logic 0 and the ErrorReg register's BufferOvfl bit is cleared. The bytes stored in the FIFO buffer are no longer accessible allowing the FIFO buffer to be filled with another 64 bytes.

8.3.3 FIFO buffer status information

The host can get the following FIFO buffer status information:

- Number of bytes stored in the FIFO buffer: FIFOLevelReg register's FIFOLevel[6:0]
- FIFO buffer almost full warning: Status1Reg register's HiAlert bit

- FIFO buffer almost empty warning: Status1Reg register's LoAlert bit
- FIFO buffer overflow warning: ErrorReg register's BufferOvfl bit. The BufferOvfl bit can only be cleared by setting the FIFOLevelReg register's FlushBuffer bit.

The MFRC522 can generate an interrupt signal when:

- ComIEnReg register's LoAlertIEn bit is set to logic 1. It activates pin IRQ when Status1Reg register's LoAlert bit changes to logic 1.
- ComIEnReg register's HiAlertIEn bit is set to logic 1. It activates pin IRQ when Status1Reg register's HiAlert bit changes to logic 1.

If the maximum number of WaterLevel bytes (as set in the WaterLevelReg register) or less are stored in the FIFO buffer, the HiAlert bit is set to logic 1. It is generated according to [Equation 3](#):

$$HiAlert = (64 - FIFOLength) \leq WaterLevel \quad (3)$$

If the number of WaterLevel bytes (as set in the WaterLevelReg register) or less are stored in the FIFO buffer, the LoAlert bit is set to logic 1. It is generated according to [Equation 4](#):

$$LoAlert = FIFOLength \leq WaterLevel \quad (4)$$

8.4 Interrupt request system

The MFRC522 indicates certain events by setting the Status1Reg register's IRq bit and, if activated, by pin IRQ. The signal on pin IRQ can be used to interrupt the host using its interrupt handling capabilities. This allows the implementation of efficient host software.

8.4.1 Interrupt sources overview

[Table 18](#) shows the available interrupt bits, the corresponding source and the condition for its activation. The ComIrqReg register's TimerIRq interrupt bit indicates an interrupt set by the timer unit which is set when the timer decrements from 1 to 0.

The ComIrqReg register's TxIRq bit indicates that the transmitter has finished. If the state changes from sending data to transmitting the end of the frame pattern, the transmitter unit automatically sets the interrupt bit. The CRC coprocessor sets the DivIrqReg register's CRCIRq bit after processing all the FIFO buffer data which is indicated by CRCReady bit = 1.

The ComIrqReg register's RxIRq bit indicates an interrupt when the end of the received data is detected. The ComIrqReg register's IdleIRq bit is set if a command finishes and the Command[3:0] value in the CommandReg register changes to idle (see [Table 149 on page 70](#)).

The ComIrqReg register's HiAlertIRq bit is set to logic 1 when the Status1Reg register's HiAlert bit is set to logic 1 which means that the FIFO buffer has reached the level indicated by the WaterLevel[5:0] bits.

The ComIrqReg register's LoAlertIRq bit is set to logic 1 when the Status1Reg register's LoAlert bit is set to logic 1 which means that the FIFO buffer has reached the level indicated by the WaterLevel[5:0] bits.

The ComIrqReg register's ErrIRq bit indicates an error detected by the contactless UART during send or receive. This is indicated when any bit is set to logic 1 in register ErrorReg.

Table 18. Interrupt sources

| Interrupt flag | Interrupt source | Trigger action |
|----------------|--------------------|--|
| IRq | timer unit | the timer counts from 1 to 0 |
| TxIRq | transmitter | a transmitted data stream ends |
| CRCIRq | CRC coprocessor | all data from the FIFO buffer has been processed |
| RxIRq | receiver | a received data stream ends |
| IdleIRq | ComIrqReg register | command execution finishes |
| HiAlertIRq | FIFO buffer | the FIFO buffer is almost full |
| LoAlertIRq | FIFO buffer | the FIFO buffer is almost empty |
| ErrIRq | contactless UART | an error is detected |

8.5 Timer unit

The MFRC522A has a timer unit which the external host can use to manage timing tasks. The timer unit can be used in one of the following timer/counter configurations:

- Timeout counter
- Watchdog counter
- Stop watch
- Programmable one shot
- Periodical trigger

The timer unit can be used to measure the time interval between two events or to indicate that a specific event occurred after a specific time. The timer can be triggered by events explained in the paragraphs below. The timer does not influence any internal events, for example, a time-out during data reception does not automatically influence the reception process. Furthermore, several timer-related bits can be used to generate an interrupt.

The timer has an input clock of 13.56 MHz derived from the 27.12 MHz quartz crystal oscillator. The timer consists of two stages: prescaler and counter.

The prescaler (TPrescaler) is a 12-bit counter. The reload values (TReloadVal_Hi[7:0] and TReloadVal_Lo[7:0]) for TPrescaler can be set between 0 and 4095 in the TModeReg register's TPrescaler_Hi[3:0] bits and TPrescalerReg register's TPrescaler_Lo[7:0] bits.

The reload value for the counter is defined by 16 bits between 0 and 65535 in the TReloadReg register.

The current value of the timer is indicated in the TCounterValReg register.

When the counter reaches 0, an interrupt is automatically generated, indicated by the ComIrqReg register's TimerIRq bit setting. If enabled, this event can be indicated on pin IRQ. The TimerIRq bit can be set and reset by the host. Depending on the configuration, the timer will stop at 0 or restart with the value set in the TReloadReg register.

The timer status is indicated by the Status1Reg register's TRunning bit.

The timer can be started manually using the ControlReg register's TStartNow bit and stopped using the ControlReg register's TStopNow bit.

The timer can also be activated automatically to meet any dedicated protocol requirements by setting the TModeReg register's TAuto bit to logic 1.

The delay time of a timer stage is set by the reload value + 1. The total delay time (t_{d1}) is calculated using [Equation 5](#):

$$t_{d1} = \frac{(TPrescaler \times 2 + 1) \times (TReloadVal + 1)}{13.56 \text{ MHz}} \quad (5)$$

An example of calculating total delay time (t_d) is shown in [Equation 6](#), where the TPrescaler value = 4095 and TReloadVal = 65535:

$$39.59 \text{ s} = \frac{(4095 \times 2 + 1) \times (65535 + 1)}{13.56 \text{ MHz}} \quad (6)$$

Example: To give a delay time of 25 μs requires 339 clock cycles to be counted and a TPrescaler value of 169. This configures the timer to count up to 65535 time-slots for every 25 μs period.

The MFRC522 version 2.0 offers in addition a second prescaler timer. Due to the fact that the prescaler counts down to 0 the prescaler period always count an odd number of clocks (1, 3, 5, ..). This may lead to inaccuracy. The second available prescaler timer implements the possibility to change the prescaler reload value to odd numbers, which results in an even prescaler period. This new prescaler can be enabled only in version 2.0 using the register bit DemodeReg, see [Table 72](#). Within this option, the total delay time (t_{d2}) is calculated using [Equation 5](#):

$$t_{d2} = \frac{(TPrescaler \times 2 + 2) \times (TReloadVal + 1)}{13.56 \text{ MHz}} \quad (7)$$

8.6 Power reduction modes

8.6.1 Hard power-down

Hard power-down is enabled when pin NRSTPD is LOW. This turns off all internal current sinks including the oscillator. All digital input buffers are separated from the input pins and clamped internally (except pin NRSTPD). The output pins are frozen at either a HIGH or LOW level.

8.6.2 Soft power-down mode

Soft Power-down mode is entered immediately after the CommandReg register's PowerDown bit is set to logic 1. All internal current sinks are switched off, including the oscillator buffer. However, the digital input buffers are not separated from the input pins and keep their functionality. The digital output pins do not change their state.

During soft power-down, all register values, the FIFO buffer content and the configuration keep their current contents.

After setting the PowerDown bit to logic 0, it takes 1024 clocks until the Soft power-down mode is exited indicated by the PowerDown bit. Setting it to logic 0 does not immediately clear it. It is cleared automatically by the MFRC522 when Soft power-down mode is exited.

Remark: If the internal oscillator is used, you must take into account that it is supplied by pin AVDD and it will take a certain time (t_{osc}) until the oscillator is stable and the clock cycles can be detected by the internal logic. It is recommended for the serial UART, to first send the value 55h to the MFRC522. The oscillator must be stable for further access to the registers. To ensure this, perform a read access to address 0 until the MFRC522 answers to the last read command with the register content of address 0. This indicates that the MFRC522 is ready.

8.6.3 Transmitter power-down mode

The Transmitter Power-down mode switches off the internal antenna drivers thereby, turning off the RF field. Transmitter power-down mode is entered by setting either the TxControlReg register's Tx1RFEEn bit or Tx2RFEEn bit to logic 0.

8.7 Oscillator circuit

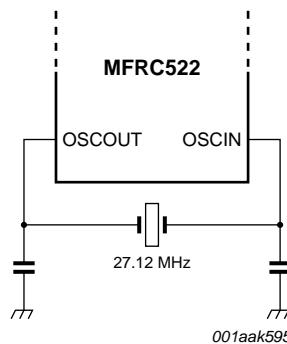


Fig 22. Quartz crystal connection

The clock applied to the MFRC522 provides a time basis for the synchronous system's encoder and decoder. The stability of the clock frequency, therefore, is an important factor for correct operation. To obtain optimum performance, clock jitter must be reduced as much as possible. This is best achieved using the internal oscillator buffer with the recommended circuitry.

If an external clock source is used, the clock signal must be applied to pin OSCIN. In this case, special care must be taken with the clock duty cycle and clock jitter and the clock quality must be verified.

8.8 Reset and oscillator start-up time

8.8.1 Reset timing requirements

The reset signal is filtered by a hysteresis circuit and a spike filter before it enters the digital circuit. The spike filter rejects signals shorter than 10 ns. In order to perform a reset, the signal must be LOW for at least 100 ns.

8.8.2 Oscillator start-up time

If the MFRC522 has been set to a Power-down mode or is powered by a V_{DDX} supply, the start-up time for the MFRC522 depends on the oscillator used and is shown in [Figure 23](#).

The time (t_{startup}) is the start-up time of the crystal oscillator circuit. The crystal oscillator start-up time is defined by the crystal.

The time (t_d) is the internal delay time of the MFRC522 when the clock signal is stable before the MFRC522 can be addressed.

The delay time is calculated by:

$$t_d = \frac{1024}{27 \mu\text{s}} = 37.74 \mu\text{s} \quad (8)$$

The time (t_{osc}) is the sum of t_d and t_{startup} .

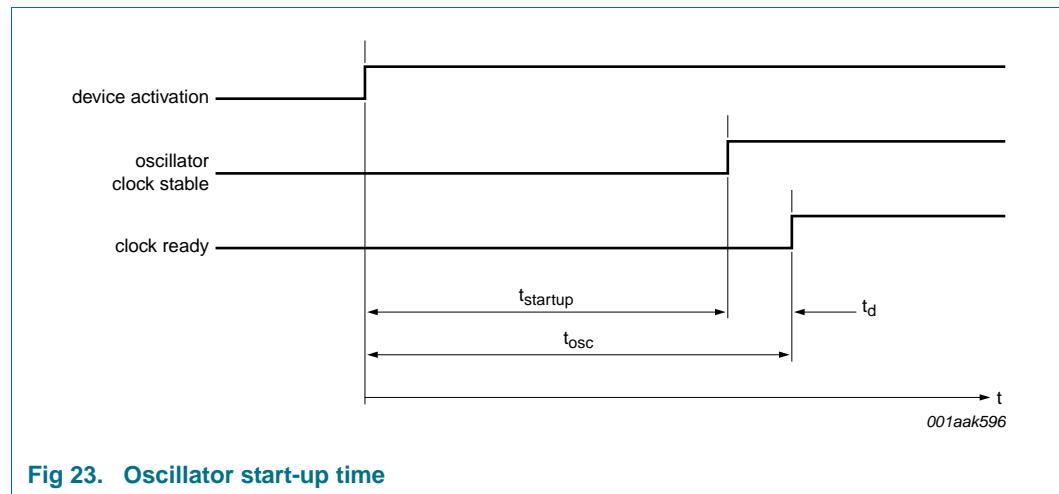


Fig 23. Oscillator start-up time

9. MFRC522 registers

9.1 Register bit behavior

Depending on the functionality of a register, the access conditions to the register can vary. In principle, bits with same behavior are grouped in common registers. The access conditions are described in [Table 19](#).

Table 19. Behavior of register bits and their designation

| Abbreviation | Behavior | Description |
|--------------|----------------|---|
| R/W | read and write | These bits can be written and read by the microcontroller. Since they are used only for control purposes, their content is not influenced by internal state machines, for example the ComIEnReg register can be written and read by the microcontroller. It will also be read by internal state machines but never changed by them. |
| D | dynamic | These bits can be written and read by the microcontroller. Nevertheless, they can also be written automatically by internal state machines, for example the CommandReg register changes its value automatically after the execution of the command. |
| R | read only | These register bits hold values which are determined by internal states only, for example the CRCReady bit cannot be written externally but shows internal states. |
| W | write only | Reading these register bits always returns zero. |
| reserved | - | These registers are reserved for future use and must not be changed. In case of a write access, it is recommended to always write the value "0". |
| RFT | - | These register bits are reserved for future use or are for production tests and must not be changed. |

9.2 Register overview

Table 20. MFRC522 register overview

| Address (hex) | Register name | Function | Refer to |
|-----------------------------------|----------------|--|-------------------------------------|
| Page 0: Command and status | | | |
| 00h | Reserved | reserved for future use | Table 21 on page 38 |
| 01h | CommandReg | starts and stops command execution | Table 23 on page 38 |
| 02h | ComlEnReg | enable and disable interrupt request control bits | Table 25 on page 38 |
| 03h | DivlEnReg | enable and disable interrupt request control bits | Table 27 on page 39 |
| 04h | ComlrqReg | interrupt request bits | Table 29 on page 39 |
| 05h | DivlrqReg | interrupt request bits | Table 31 on page 40 |
| 06h | ErrorReg | error bits showing the error status of the last command executed | Table 33 on page 41 |
| 07h | Status1Reg | communication status bits | Table 35 on page 42 |
| 08h | Status2Reg | receiver and transmitter status bits | Table 37 on page 43 |
| 09h | FIFODataReg | input and output of 64 byte FIFO buffer | Table 39 on page 44 |
| 0Ah | FIFOLevelReg | number of bytes stored in the FIFO buffer | Table 41 on page 44 |
| 0Bh | WaterLevelReg | level for FIFO underflow and overflow warning | Table 43 on page 44 |
| 0Ch | ControlReg | miscellaneous control registers | Table 45 on page 45 |
| 0Dh | BitFramingReg | adjustments for bit-oriented frames | Table 47 on page 46 |
| 0Eh | CollReg | bit position of the first bit-collision detected on the RF interface | Table 49 on page 46 |
| 0Fh | Reserved | reserved for future use | Table 51 on page 47 |
| Page 1: Command | | | |
| 10h | Reserved | reserved for future use | Table 53 on page 47 |
| 11h | ModeReg | defines general modes for transmitting and receiving | Table 55 on page 48 |
| 12h | TxModeReg | defines transmission data rate and framing | Table 57 on page 48 |
| 13h | RxModeReg | defines reception data rate and framing | Table 59 on page 49 |
| 14h | TxControlReg | controls the logical behavior of the antenna driver pins TX1 and TX2 | Table 61 on page 50 |
| 15h | TxASKReg | controls the setting of the transmission modulation | Table 63 on page 51 |
| 16h | TxSelReg | selects the internal sources for the antenna driver | Table 65 on page 51 |
| 17h | RxSelReg | selects internal receiver settings | Table 67 on page 52 |
| 18h | RxThresholdReg | selects thresholds for the bit decoder | Table 69 on page 53 |
| 19h | DemodReg | defines demodulator settings | Table 71 on page 53 |
| 1Ah | Reserved | reserved for future use | Table 73 on page 54 |
| 1Bh | Reserved | reserved for future use | Table 75 on page 54 |
| 1Ch | MfTxReg | controls some MIFARE communication transmit parameters | Table 77 on page 55 |
| 1Dh | MfRxReg | controls some MIFARE communication receive parameters | Table 79 on page 55 |
| 1Eh | Reserved | reserved for future use | Table 81 on page 55 |
| 1Fh | SerialSpeedReg | selects the speed of the serial UART interface | Table 83 on page 55 |
| Page 2: Configuration | | | |
| 20h | Reserved | reserved for future use | Table 85 on page 57 |

Table 20. MFRC522 register overview ...continued

| Address (hex) | Register name | Function | Refer to |
|---------------|----------------|--|--------------------------------------|
| 21h | CRCResultReg | shows the MSB and LSB values of the CRC calculation | Table 87 on page 57 |
| 22h | | | Table 89 on page 57 |
| 23h | Reserved | reserved for future use | Table 91 on page 58 |
| 24h | ModWidthReg | controls the ModWidth setting | Table 93 on page 58 |
| 25h | Reserved | reserved for future use | Table 95 on page 58 |
| 26h | RFCfgReg | configures the receiver gain | Table 97 on page 59 |
| 27h | GsNReg | selects the conductance of the antenna driver pins TX1 and TX2 for modulation | Table 99 on page 59 |
| 28h | CWGSPReg | defines the conductance of the p-driver output during periods of no modulation | Table 101 on page 60 |
| 29h | ModGsPReg | defines the conductance of the p-driver output during periods of modulation | Table 103 on page 60 |
| 2Ah | TModeReg | defines settings for the internal timer | Table 105 on page 60 |
| 2Bh | TPrescalerReg | | Table 107 on page 61 |
| 2Ch | TReloadReg | defines the 16-bit timer reload value | Table 109 on page 62 |
| 2Dh | | | Table 111 on page 62 |
| 2Eh | TCounterValReg | shows the 16-bit timer value | Table 113 on page 63 |
| 2Fh | | | Table 115 on page 63 |

Page 3: Test register

| | | | |
|------------|-----------------|---|---|
| 30h | Reserved | reserved for future use | Table 117 on page 63 |
| 31h | TestSel1Reg | general test signal configuration | Table 119 on page 63 |
| 32h | TestSel2Reg | general test signal configuration and PRBS control | Table 121 on page 64 |
| 33h | TestPinEnReg | enables pin output driver on pins D1 to D7 | Table 123 on page 64 |
| 34h | TestPinValueReg | defines the values for D1 to D7 when it is used as an I/O bus | Table 125 on page 65 |
| 35h | TestBusReg | shows the status of the internal test bus | Table 127 on page 65 |
| 36h | AutoTestReg | controls the digital self test | Table 129 on page 66 |
| 37h | VersionReg | shows the software version | Table 131 on page 66 |
| 38h | AnalogTestReg | controls the pins AUX1 and AUX2 | Table 133 on page 67 |
| 39h | TestDAC1Reg | defines the test value for TestDAC1 | Table 135 on page 68 |
| 3Ah | TestDAC2Reg | defines the test value for TestDAC2 | Table 137 on page 68 |
| 3Bh | TestADCReg | shows the value of ADC I and Q channels | Table 139 on page 68 |
| 3Ch to 3Fh | Reserved | reserved for production tests | Table 141 to Table 147 on page 69 |

9.3 Register descriptions

9.3.1 Page 0: Command and status

9.3.1.1 Reserved register 00h

Functionality is reserved for future use.

Table 21. Reserved register (address 00h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 22. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|--------|-------------|
| 7 to 0 | - | reserved |

9.3.1.2 CommandReg register

Starts and stops command execution.

Table 23. CommandReg register (address 01h); reset value: 20h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------|----------|---|--------|-----------|---|--------------|---|---|
| Symbol: | reserved | | RcvOff | PowerDown | | Command[3:0] | | |
| Access: | - | | R/W | D | | D | | |

Table 24. CommandReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------|-------|--|
| 7 to 6 | reserved | - | reserved for future use |
| 5 | RcvOff | 1 | analog part of the receiver is switched off |
| 4 | PowerDown | 1 | Soft power-down mode entered |
| | | 0 | MFRC522 starts the wake up procedure during which this bit is read as a logic 1; it is read as a logic 0 when the MFRC522 is ready; see Section 8.6.2 on page 33 Remark: The PowerDown bit cannot be set when the SoftReset command is activated |
| 3 to 0 | Command[3:0] | - | activates a command based on the Command value; reading this register shows which command is executed; see Section 10.3 on page 70 |

9.3.1.3 ComIEnReg register

Control bits to enable and disable the passing of interrupt requests.

Table 25. ComIEnReg register (address 02h); reset value: 80h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------|-------|-------|---------|------------|------------|--------|----------|
| Symbol | IRqlInv | TxIEn | RxIEn | IdleIEn | HiAlertIEn | LoAlertIEn | ErrIEn | TimerIEn |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Table 26. ComIEnReg register bit descriptions

| Bit | Symbol | Value | Description |
|-----|------------|-------|---|
| 7 | IRqlInv | 1 | signal on pin IRQ is inverted with respect to the Status1Reg register's IRq bit |
| | | 0 | signal on pin IRQ is equal to the IRq bit; in combination with the DivIEnReg register's IRqPushPull bit, the default value of logic 1 ensures that the output level on pin IRQ is 3-state |
| 6 | TxIEn | - | allows the transmitter interrupt request (TxIRq bit) to be propagated to pin IRQ |
| 5 | RxIEn | - | allows the receiver interrupt request (RxIRq bit) to be propagated to pin IRQ |
| 4 | IdleIEn | - | allows the idle interrupt request (IdleIRq bit) to be propagated to pin IRQ |
| 3 | HiAlertIEn | - | allows the high alert interrupt request (HiAlertIRq bit) to be propagated to pin IRQ |
| 2 | LoAlertIEn | - | allows the low alert interrupt request (LoAlertIRq bit) to be propagated to pin IRQ |
| 1 | ErrIEn | - | allows the error interrupt request (ErrIRq bit) to be propagated to pin IRQ |
| 0 | TimerIEn | - | allows the timer interrupt request (TimerIRq bit) to be propagated to pin IRQ |

9.3.1.4 DivIEnReg register

Control bits to enable and disable the passing of interrupt requests.

Table 27. DivIEnReg register (address 03h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------------|----------|------------|----------|--------|----------|---|---|
| Symbol | IRQPushPull | reserved | MfinActIEn | reserved | CRCIEn | reserved | | |
| Access | R/W | - | R/W | - | R/W | | - | |

Table 28. DivIEnReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-------------|-------|---|
| 7 | IRQPushPull | 1 | pin IRQ is a standard CMOS output pin |
| | | 0 | pin IRQ is an open-drain output pin |
| 6 to 5 | reserved | - | reserved for future use |
| 4 | MfinActIEn | - | allows the MFIN active interrupt request to be propagated to pin IRQ |
| 3 | reserved | - | reserved for future use |
| 2 | CRCIEn | - | allows the CRC interrupt request, indicated by the DivIrqReg register's CRCIRq bit, to be propagated to pin IRQ |
| 1 to 0 | reserved | - | reserved for future use |

9.3.1.5 ComIrqReg register

Interrupt request bits.

Table 29. ComIrqReg register (address 04h); reset value: 14h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------|-------|-------|---------|------------|------------|--------|----------|
| Symbol | Set1 | TxIRq | RxIRq | IdleIRq | HiAlertIRq | LoAlertIRq | ErrIRq | TimerIRq |
| Access | W | D | D | D | D | D | D | D |

Table 30. ComIrqReg register bit descriptions*All bits in the ComIrqReg register are cleared by software.*

| Bit | Symbol | Value | Description |
|-----|------------|-------|---|
| 7 | Set1 | 1 | indicates that the marked bits in the ComIrqReg register are set |
| | | 0 | indicates that the marked bits in the ComIrqReg register are cleared |
| 6 | TxIRq | 1 | set immediately after the last bit of the transmitted data was sent out |
| 5 | RxIRq | 1 | receiver has detected the end of a valid data stream if the RxModeReg register's RxNoErr bit is set to logic 1, the RxIRq bit is only set to logic 1 when data bytes are available in the FIFO |
| 4 | IdleIRq | 1 | If a command terminates, for example, when the CommandReg changes its value from any command to the Idle command (see Table 149 on page 70) if an unknown command is started, the CommandReg register Command[3:0] value changes to the idle state and the IdleIRq bit is set The microcontroller starting the Idle command does not set the IdleIRq bit |
| 3 | HiAlertIRq | 1 | the Status1Reg register's HiAlert bit is set in opposition to the HiAlert bit, the HiAlertIRq bit stores this event and can only be reset as indicated by the Set1 bit in this register |
| 2 | LoAlertIRq | 1 | Status1Reg register's LoAlert bit is set in opposition to the LoAlert bit, the LoAlertIRq bit stores this event and can only be reset as indicated by the Set1 bit in this register |
| 1 | ErrIRq | 1 | any error bit in the ErrorReg register is set |
| 0 | TimerIRq | 1 | the timer decrements the timer value in register TCounterValReg to zero |

9.3.1.6 DivIrqReg register

Interrupt request bits.

Table 31. DivIrqReg register (address 05h); reset value: x0h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------|----------|---|------------|----------|--------|----------|---|
| Symbol | Set2 | reserved | | MfinActIRq | reserved | CRCIRq | reserved | |
| Access | W | - | | D | - | D | - | |

Table 32. DivIrqReg register bit descriptions*All bits in the DivIrqReg register are cleared by software.*

| Bit | Symbol | Value | Description |
|--------|------------|-------|---|
| 7 | Set2 | 1 | indicates that the marked bits in the DivIrqReg register are set |
| | | 0 | indicates that the marked bits in the DivIrqReg register are cleared |
| 6 to 5 | reserved | - | reserved for future use |
| 4 | MfinActIRq | 1 | MFIN is active this interrupt is set when either a rising or falling signal edge is detected |
| 3 | reserved | - | reserved for future use |
| 2 | CRCIRq | 1 | the CalcCRC command is active and all data is processed |
| 1 to 0 | reserved | - | reserved for future use |

9.3.1.7 ErrorReg register

Error bit register showing the error status of the last command executed.

Table 33. ErrorReg register (address 06h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------|---------|----------|------------|---------|--------|-----------|-------------|
| Symbol | WrErr | TempErr | reserved | BufferOvfl | CollErr | CRCErr | ParityErr | ProtocolErr |
| Access | R | R | - | R | R | R | R | R |

Table 34. ErrorReg register bit descriptions

| Bit | Symbol | Value | Description |
|-----|------------------------|-------|--|
| 7 | WrErr | 1 | data is written into the FIFO buffer by the host during the MFAuthent command or if data is written into the FIFO buffer by the host during the time between sending the last bit on the RF interface and receiving the last bit on the RF interface |
| 6 | TempErr ^[1] | 1 | internal temperature sensor detects overheating, in which case the antenna drivers are automatically switched off |
| 5 | reserved | - | reserved for future use |
| 4 | BufferOvfl | 1 | the host or a MFRC522's internal state machine (e.g. receiver) tries to write data to the FIFO buffer even though it is already full |
| 3 | CollErr | 1 | a bit-collision is detected cleared automatically at receiver start-up phase only valid during the bitwise anticollision at 106 kBd always set to logic 0 during communication protocols at 212 kBd, 424 kBd and 848 kBd |
| 2 | CRCErr | 1 | the RxModeReg register's RxCRCEn bit is set and the CRC calculation fails automatically cleared to logic 0 during receiver start-up phase |
| 1 | ParityErr | 1 | parity check failed automatically cleared during receiver start-up phase only valid for ISO/IEC 14443 A/MIFARE communication at 106 kBd |
| 0 | ProtocolErr | 1 | set to logic 1 if the SOF is incorrect automatically cleared during receiver start-up phase bit is only valid for 106 kBd during the MFAuthent command, the ProtocolErr bit is set to logic 1 if the number of bytes received in one data stream is incorrect |

[1] Command execution clears all error bits except the TempErr bit. Cannot be set by software.

9.3.1.8 Status1Reg register

Contains status bits of the CRC, interrupt and FIFO buffer.

Table 35. Status1Reg register (address 07h); reset value: 21h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|-------|----------|-----|----------|----------|---------|---------|
| Symbol | reserved | CRCOk | CRCReady | IRq | TRunning | reserved | HiAlert | LoAlert |
| Access | - | R | R | R | R | - | R | R |

Table 36. Status1Reg register bit descriptions

| Bit | Symbol | Value | Description |
|-----|----------|-------|--|
| 7 | reserved | - | reserved for future use |
| 6 | CRCOk | 1 | the CRC result is zero for data transmission and reception, the CRCOk bit is undefined: use the ErrorReg register's CRCErr bit indicates the status of the CRC coprocessor, during calculation the value changes to logic 0, when the calculation is done correctly the value changes to logic 1 |
| 5 | CRCReady | 1 | the CRC calculation has finished only valid for the CRC coprocessor calculation using the CalcCRC command |
| 4 | IRq | - | indicates if any interrupt source requests attention with respect to the setting of the interrupt enable bits: see the ComIEnReg and DivIEnReg registers |
| 3 | TRunning | 1 | MFRC522's timer unit is running, i.e. the timer will decrement the TCounterValReg register with the next timer clock Remark: in gated mode, the TRunning bit is set to logic 1 when the timer is enabled by TModeReg register's TGated[1:0] bits; this bit is not influenced by the gated signal |
| 2 | reserved | - | reserved for future use |
| 1 | HiAlert | 1 | the number of bytes stored in the FIFO buffer corresponds to equation: $HiAlert = (64 - FIFOLength) \leq WaterLevel$ example: FIFO length = 60, WaterLevel = 4 → HiAlert = 1 FIFO length = 59, WaterLevel = 4 → HiAlert = 0 |
| 0 | LoAlert | 1 | the number of bytes stored in the FIFO buffer corresponds to equation: $LoAlert = FIFOlength \leq WaterLevel$ example: FIFO length = 4, WaterLevel = 4 → LoAlert = 1 FIFO length = 5, WaterLevel = 4 → LoAlert = 0 |

9.3.1.9 Status2Reg register

Contains status bits of the receiver, transmitter and data mode detector.

Table 37. Status2Reg register (address 08h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------|-------------------------|----------|-------------|-----------------|---|---|---|
| Symbol | TempSensClear | I ² CForceHS | reserved | MFCrypto1On | ModemState[2:0] | | | |
| Access | R/W | R/W | - | D | | R | | |

Table 38. Status2Reg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-------------------------|-------|--|
| 7 | TempSensClear | 1 | clears the temperature error if the temperature is below the alarm limit of 125 °C |
| 6 | I ² CForceHS | | I ² C-bus input filter settings: |
| | | 1 | the I ² C-bus input filter is set to the High-speed mode independent of the I ² C-bus protocol |
| | | 0 | the I ² C-bus input filter is set to the I ² C-bus protocol used |
| 5 to 4 | reserved | - | reserved |
| 3 | MFCrypto1On | - | indicates that the MIFARE Crypto1 unit is switched on and therefore all data communication with the card is encrypted can only be set to logic 1 by a successful execution of the MFAuthent command only valid in Read/Write mode for MIFARE standard cards this bit is cleared by software |
| 2 to 0 | ModemState[2:0] | - | shows the state of the transmitter and receiver state machines: |
| | | 000 | idle |
| | | 001 | wait for the BitFramingReg register's StartSend bit |
| | | 010 | TxWait: wait until RF field is present if the TModeReg register's TxWaitRF bit is set to logic 1 the minimum time for TxWait is defined by the TxWaitReg register |
| | | 011 | transmitting |
| | | 100 | RxWait: wait until RF field is present if the TModeReg register's RxWaitRF bit is set to logic 1 the minimum time for RxWait is defined by the RxWaitReg register |
| | | 101 | wait for data |
| | | 110 | receiving |

9.3.1.10 FIFODataReg register

Input and output of 64 byte FIFO buffer.

Table 39. FIFODataReg register (address 09h); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------|---|---|---|---|---|---|---|
| Symbol | FIFOData[7:0] | | | | | | | |
| Access | D | | | | | | | |

Table 40. FIFODataReg register bit descriptions

| Bit | Symbol | Description |
|--------|---------------|---|
| 7 to 0 | FIFOData[7:0] | data input and output port for the internal 64-byte FIFO buffer FIFO buffer acts as parallel in/parallel out converter for all serial data stream inputs and outputs |

9.3.1.11 FIFOLevelReg register

Indicates the number of bytes stored in the FIFO.

Table 41. FIFOLevelReg register (address 0Ah); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------------|---|---|---|---|---|---|---|
| Symbol | FlushBuffer | | | | | | | |
| Access | W R | | | | | | | |

Table 42. FIFOLevelReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|--|
| 7 | FlushBuffer | 1 | immediately clears the internal FIFO buffer's read and write pointer and ErrorReg register's BufferOvfl bit reading this bit always returns 0 |
| 6 to 0 | FIFOLevel [6:0] | - | indicates the number of bytes stored in the FIFO buffer writing to the FIFODataReg register increments and reading decrements the FIFOLevel value |

9.3.1.12 WaterLevelReg register

Defines the level for FIFO under- and overflow warning.

Table 43. WaterLevelReg register (address 0Bh); reset value: 08h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | R/W | | | | | | | |

Table 44. WaterLevelReg register bit descriptions

| Bit | Symbol | Description |
|--------|------------------|--|
| 7 to 6 | reserved | reserved for future use |
| 5 to 0 | WaterLevel [5:0] | <p>defines a warning level to indicate a FIFO buffer overflow or underflow:</p> <p>Status1Reg register's HiAlert bit is set to logic 1 if the remaining number of bytes in the FIFO buffer space is equal to, or less than the defined number of WaterLevel bytes</p> <p>Status1Reg register's LoAlert bit is set to logic 1 if equal to, or less than the WaterLevel bytes in the FIFO buffer</p> <p>Remark: to calculate values for HiAlert and LoAlert see Section 9.3.1.8 on page 42.</p> |

9.3.1.13 ControlReg register

Miscellaneous control bits.

Table 45. ControlReg register (address 0Ch); reset value: 10h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|-----------|----------|---|---|-----------------|---|---|
| Symbol | TStopNow | TStartNow | reserved | | | RxLastBits[2:0] | | |
| Access | W | W | - | | | R | | |

Table 46. ControlReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|--|
| 7 | TStopNow | 1 | timer stops immediately reading this bit always returns it to logic0 |
| 6 | TStartNow | 1 | timer starts immediately reading this bit always returns it to logic 0 |
| 5 to 3 | reserved | - | reserved for future use |
| 2 to 0 | RxLastBits[2:0] | - | indicates the number of valid bits in the last received byte if this value is 000b, the whole byte is valid |

9.3.1.14 BitFramingReg register

Adjustments for bit-oriented frames.

Table 47. BitFramingReg register (address 0Dh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----------|--------------|---|---|----------|-----------------|---|---|
| Symbol | StartSend | RxAlign[2:0] | | | reserved | TxLastBits[2:0] | | |
| Access | W | R/W | | | - | R/W | | |

Table 48. BitFramingReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|--|
| 7 | StartSend | 1 | starts the transmission of data only valid in combination with the Transceive command |
| 6 to 4 | RxAlign[2:0] | | used for reception of bit-oriented frames: defines the bit position for the first bit received to be stored in the FIFO buffer example: |
| | | 0 | LSB of the received bit is stored at bit position 0, the second received bit is stored at bit position 1 |
| | | 1 | LSB of the received bit is stored at bit position 1, the second received bit is stored at bit position 2 |
| | | 7 | LSB of the received bit is stored at bit position 7, the second received bit is stored in the next byte that follows at bit position 0 These bits are only to be used for bitwise anticollision at 106 kBd, for all other modes they are set to 0 |
| 3 | reserved | - | reserved for future use |
| 2 to 0 | TxLastBits[2:0] | - | used for transmission of bit oriented frames: defines the number of bits of the last byte that will be transmitted 000b indicates that all bits of the last byte will be transmitted |

9.3.1.15 CollReg register

Defines the first bit-collision detected on the RF interface.

Table 49. CollReg register (address 0Eh); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----------------|----------|-----------------|--------------|---|---|---|---|
| Symbol | ValuesAfterColl | reserved | CollPosNotValid | CollPos[4:0] | | | | |
| Access | R/W | - | R | | | | R | |

Table 50. CollReg register bit descriptions

| Bit | Symbol | Value | Description |
|-----|-----------------|-------|--|
| 7 | ValuesAfterColl | 0 | all received bits will be cleared after a collision only used during bitwise anticollision at 106 kBd, otherwise it is set to logic 1 |
| 6 | reserved | - | reserved for future use |
| 5 | CollPosNotValid | 1 | no collision detected or the position of the collision is out of the range of CollPos[4:0] |

Table 50. CollReg register bit descriptions ...continued

| Bit | Symbol | Value | Description |
|--------|--------------|-------|--|
| 4 to 0 | CollPos[4:0] | - | shows the bit position of the first detected collision in a received frame only data bits are interpreted example: |
| | | 00h | indicates a bit-collision in the 32 nd bit |
| | | 01h | indicates a bit-collision in the 1 st bit |
| | | 08h | indicates a bit-collision in the 8 th bit These bits will only be interpreted if the CollPosNotValid bit is set to logic 0 |

9.3.1.16 Reserved register 0Fh

Functionality is reserved for future use.

Table 51. Reserved register (address 0Fh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 52. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.2 Page 1: Communication

9.3.2.1 Reserved register 10h

Functionality is reserved for future use.

Table 53. Reserved register (address 10h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 54. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.2.2 ModeReg register

Defines general mode settings for transmitting and receiving.

Table 55. ModeReg register (address 11h); reset value: 3Fh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|----------|----------|----------|---------|----------|----------------|---|
| Symbol | MSBFirst | reserved | TxWaitRF | reserved | PolMFin | reserved | CRCPreset[1:0] | |
| Access | R/W | - | R/W | - | R/W | - | R/W | |

Table 56. ModeReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|---|
| 7 | MSBFirst | 1 | CRC coprocessor calculates the CRC with MSB first in the CRCResultReg register the values for the CRCResultMSB[7:0] bits and the CRCResultLSB[7:0] bits are bit reversed Remark: during RF communication this bit is ignored |
| 6 | reserved | - | reserved for future use |
| 5 | TxWaitRF | 1 | transmitter can only be started if an RF field is generated |
| 4 | reserved | - | reserved for future use |
| 3 | PolMFin | | defines the polarity of pin MFIN Remark: the internal envelope signal is encoded active LOW, changing this bit generates a MFinActIRq event |
| | | 1 | polarity of pin MFIN is active HIGH |
| | | 0 | polarity of pin MFIN is active LOW |
| 2 | reserved | - | reserved for future use |
| 1 to 0 | CRCPreset [1:0] | | defines the preset value for the CRC coprocessor for the CalcCRC command Remark: during any communication, the preset values are selected automatically according to the definition of bits in the RxModeReg and TxModeReg registers |
| | | 00 | 0000h |
| | | 01 | 6363h |
| | | 10 | A671h |
| | | 11 | FFFFh |

9.3.2.3 TxModeReg register

Defines the data rate during transmission.

Table 57. TxModeReg register (address 12h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------|--------------|---|---|--------|----------|---|---|
| Symbol | TxCRCEn | TxSpeed[2:0] | | | InvMod | reserved | | |
| Access | R/W | D | | | R/W | - | | |

Table 58. TxModeReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------|-------|---|
| 7 | TxCRCEn | 1 | enables CRC generation during data transmission Remark: can only be set to logic 0 at 106 kBd |
| 6 to 4 | TxSpeed[2:0] | | defines the bit rate during data transmission the MFRC522 handles transfer speeds up to 848 kBd |
| | | 000 | 106 kBd |
| | | 001 | 212 kBd |
| | | 010 | 424 kBd |
| | | 011 | 848 kBd |
| | | 100 | reserved |
| | | 101 | reserved |
| | | 110 | reserved |
| | | 111 | reserved |
| 3 | InvMod | 1 | modulation of transmitted data is inverted |
| 2 to 0 | reserved | - | reserved for future use |

9.3.2.4 RxModeReg register

Defines the data rate during reception.

Table 59. RxModeReg register (address 13h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------|--------------|---|---|---------|------------|----------|---|
| Symbol | RxCRCEn | RxSpeed[2:0] | | | RxNoErr | RxMultiple | reserved | |
| Access | R/W | D | | | R/W | R/W | - | |

Table 60. RxModeReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------|-------|---|
| 7 | RxCRCEn | 1 | enables the CRC calculation during reception Remark: can only be set to logic 0 at 106 kBd |
| 6 to 4 | RxSpeed[2:0] | | defines the bit rate while receiving data the MFRC522 handles transfer speeds up to 848 kBd |
| | | 000 | 106 kBd |
| | | 001 | 212 kBd |
| | | 010 | 424 kBd |
| | | 011 | 848 kBd |
| | | 100 | reserved |
| | | 101 | reserved |
| | | 110 | reserved |
| | | 111 | reserved |
| 3 | RxNoErr | 1 | an invalid received data stream (less than 4 bits received) will be ignored and the receiver remains active |

Table 60. RxModeReg register bit descriptions ...continued

| Bit | Symbol | Value | Description |
|--------|------------|-------|---|
| 2 | RxMultiple | 0 | receiver is deactivated after receiving a data frame |
| | | 1 | able to receive more than one data frame only valid for data rates above 106 kBd in order to handle the polling command after setting this bit the Receive and Transceive commands will not terminate automatically. Multiple reception can only be deactivated by writing any command (except the Receive command) to the CommandReg register, or by the host clearing the bit if set to logic 1, an error byte is added to the FIFO buffer at the end of a received data stream which is a copy of the ErrorReg register value. For the MFRC522 version 2.0 the CRC status is reflected in the signal CRCOk, which indicates the actual status of the CRC coprocessor. For the MFRC522 version 1.0 the CRC status is reflected in the signal CRCErr. |
| 1 to 0 | reserved | - | reserved for future use |

9.3.2.5 TxControlReg register

Controls the logical behavior of the antenna driver pins TX1 and TX2.

Table 61. TxControlReg register (address 14h); reset value: 80h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------------|----------------|-----------------|-----------------|-------|----------|---------------------|---------------------|
| Symbol | InvTx2RF On | InvTx1RF On | InvTx2RF Off | InvTx1RF Off | Tx2CW | reserved | Tx2RFE _n | Tx1RFE _n |
| Access | R/W | R/W | R/W | R/W | R/W | - | R/W | R/W |

Table 62. TxControlReg register bit descriptions

| Bit | Symbol | Value | Description |
|-----|---------------------|-------|---|
| 7 | InvTx2RFOn | 1 | output signal on pin TX2 inverted when driver TX2 is enabled |
| 6 | InvTx1RFOn | 1 | output signal on pin TX1 inverted when driver TX1 is enabled |
| 5 | InvTx2RFOff | 1 | output signal on pin TX2 inverted when driver TX2 is disabled |
| 4 | InvTx1RFOff | 1 | output signal on pin TX1 inverted when driver TX1 is disabled |
| 3 | Tx2CW | 1 | output signal on pin TX2 continuously delivers the unmodulated 13.56 MHz energy carrier |
| | | 0 | Tx2CW bit is enabled to modulate the 13.56 MHz energy carrier |
| 2 | reserved | - | reserved for future use |
| 1 | Tx2RFE _n | 1 | output signal on pin TX2 delivers the 13.56 MHz energy carrier modulated by the transmission data |
| 0 | Tx1RFE _n | 1 | output signal on pin TX1 delivers the 13.56 MHz energy carrier modulated by the transmission data |

9.3.2.6 TxASKReg register

Controls transmit modulation settings.

Table 63. TxASKReg register (address 15h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|-------------|---|---|---|---|----------|---|
| Symbol | reserved | Force100ASK | | | | | reserved | |
| Access | - | R/W | | | | | - | |

Table 64. TxASKReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-------------|-------|---|
| 7 | reserved | - | reserved for future use |
| 6 | Force100ASK | 1 | forces a 100 % ASK modulation independent of the ModGsPReg register setting |
| 5 to 0 | reserved | - | reserved for future use |

9.3.2.7 TxSelReg register

Selects the internal sources for the analog module.

Table 65. TxSelReg register (address 16h); reset value: 10h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------|----------|---|----------------|---|---|---------------|---|---|
| Symbol: | reserved | | DriverSel[1:0] | | | MFOutSel[3:0] | | |
| Access: | - | | R/W | | | R/W | | |

Table 66. TxSelReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|---|
| 7 to 6 | reserved | - | reserved for future use |
| 5 to 4 | DriverSel [1:0] | - | selects the input of drivers TX1 and TX2 |
| | | 00 | 3-state; in soft power-down the drivers are only in 3-state mode if the DriverSel[1:0] value is set to 3-state mode |
| | | 01 | modulation signal (envelope) from the internal encoder, Miller pulse encoded |
| | | 10 | modulation signal (envelope) from pin MFIN |
| | | 11 | HIGH; the HIGH level depends on the setting of bits InvTx1RFOOn/InvTx1RFOff and InvTx2RFOOn/InvTx2RFOff |

Table 66. TxSelReg register bit descriptions ...continued

| Bit | Symbol | Value | Description |
|--------|-------------------|--------------|--|
| 3 to 0 | MFOutSel [3:0] | | selects the input for pin MFOUT |
| | | 0000 | 3-state |
| | | 0001 | LOW |
| | | 0010 | HIGH |
| | | 0011 | test bus signal as defined by the TestSel1Reg register's TstBusBitSel[2:0] value |
| | | 0100 | modulation signal (envelope) from the internal encoder, Miller pulse encoded |
| | | 0101 | serial data stream to be transmitted, data stream before Miller encoder |
| | | 0110 | reserved |
| | | 0111 | serial data stream received, data stream after Manchester decoder |
| | | 1000 to 1111 | reserved |

9.3.2.8 RxSelReg register

Selects internal receiver settings.

Table 67. RxSelReg register (address 17h); reset value: 84h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------|---|---|---|-------------|---|---|---|
| Symbol | UARTSel[1:0] | | | | RxWait[5:0] | | | |
| Access | R/W | | | | R/W | | | |

Table 68. RxSelReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|------------------|-------|--|
| 7 to 6 | UARTSel [1:0] | | selects the input of the contactless UART |
| | | 00 | constant LOW |
| | | 01 | Manchester with subcarrier from pin MFIN |
| | | 10 | modulated signal from the internal analog module, default |
| | | 11 | NRZ coding without subcarrier from pin MFIN which is only valid for transfer speeds above 106 kBd |
| 5 to 0 | RxWait [5:0] | - | after data transmission the activation of the receiver is delayed for RxWait bit-clocks, during this 'frame guard time' any signal on pin RX is ignored this parameter is ignored by the Receive command all other commands, such as Transceive, MFAuthent use this parameter the counter starts immediately after the external RF field is switched on |

9.3.2.9 RxThresholdReg register

Selects thresholds for the bit decoder.

Table 69. RxThresholdReg register (address 18h); reset value: 84h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|---------------|---|---|---|----------|----------------|---|---|--|
| Symbol | MinLevel[3:0] | | | | reserved | CollLevel[2:0] | | | |
| Access | R/W | | | | - | R/W | | | |

Table 70. RxThresholdReg register bit descriptions

| Bit | Symbol | Description |
|--------|-----------------|--|
| 7 to 4 | MinLevel [3:0] | defines the minimum signal strength at the decoder input that will be accepted if the signal strength is below this level it is not evaluated |
| 3 | reserved | reserved for future use |
| 2 to 0 | CollLevel [2:0] | defines the minimum signal strength at the decoder input that must be reached by the weaker half-bit of the Manchester encoded signal to generate a bit-collision relative to the amplitude of the stronger half-bit |

9.3.2.10 DemodReg register

Defines demodulator settings.

Table 71. DemodReg register (address 19h); reset value: 4Dh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------------|---|-------|---------------|-------------|---|--------------|---|
| Symbol | AddIQ[1:0] | | FixIQ | TPrescal Even | TauRcv[1:0] | | TauSync[1:0] | |
| Access | R/W | | R/W | R/W | R/W | | R/W | |

Table 72. DemodReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-------------|-------|--|
| 7 to 6 | AddIQ [1:0] | - | defines the use of I and Q channel during reception Remark: the FixIQ bit must be set to logic 0 to enable the following settings: |
| | | 00 | selects the stronger channel |
| | | 01 | selects the stronger channel and freezes the selected channel during communication |
| | | 10 | reserved |
| | | 11 | reserved |
| 5 | FixIQ | 1 | if AddIQ[1:0] are set to X0b, the reception is fixed to I channel if AddIQ[1:0] are set to X1b, the reception is fixed to Q channel |

Table 72. DemodReg register bit descriptions ...continued

| Bit | Symbol | Value | Description |
|--------|--------------|-------|---|
| 4 | TPrescalEven | R/W | Available on RC522 version 1.0 and version 2.0: If set to logic 0 the following formula is used to calculate the timer frequency of the prescaler: $f_{timer} = 13.56 \text{ MHz} / (2^{TPreScaler}+1)$. Only available on version 2.0: If set to logic 1 the following formula is used to calculate the timer frequency of the prescaler: $f_{timer} = 13.56 \text{ MHz} / (2^{TPreScaler}+2)$. Default TPrescalEven bit is logic 0, find more information on the prescaler in Section 8.5 . |
| 3 to 2 | TauRcv[1:0] | - | changes the time-constant of the internal PLL during data reception Remark: if set to 00b the PLL is frozen during data reception |
| 1 to 0 | TauSync[1:0] | - | changes the time-constant of the internal PLL during burst |

9.3.2.11 Reserved register 1Ah

Functionality is reserved for future use.

Table 73. Reserved register (address 1Ah); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 74. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.2.12 Reserved register 1Bh

Functionality is reserved for future use.

Table 75. Reserved register (address 1Bh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 76. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.2.13 MfTxReg register

Controls some MIFARE communication transmit parameters.

Table 77. MfTxReg register (address 1Ch); reset value: 62h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|-------------|---|
| Symbol | reserved | | | | | | TxWait[1:0] | |
| Access | - | | | | | | R/W | |

Table 78. MfTxReg register bit descriptions

| Bit | Symbol | Description |
|--------|----------|--|
| 7 to 2 | reserved | reserved for future use |
| 1 to 0 | TxWait | defines the additional response time 7 bits are added to the value of the register bit by default |

9.3.2.14 MfRxReg register

Table 79. MfRxReg register (address 1Dh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---------------|----------|---|---|---|
| Symbol | reserved | | | ParityDisable | reserved | | | |
| Access | - | | | R/W | - | | | |

Table 80. MfRxReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|---------------|-------|--|
| 7 to 5 | reserved | - | reserved for future use |
| 4 | ParityDisable | 1 | generation of the parity bit for transmission and the parity check for receiving is switched off the received parity bit is handled like a data bit |
| 3 to 0 | reserved | - | reserved for future use |

9.3.2.15 Reserved register 1Eh

Functionality is reserved for future use.

Table 81. Reserved register (address 1Eh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | - | |

Table 82. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.2.16 SerialSpeedReg register

Selects the speed of the serial UART interface.

Table 83. SerialSpeedReg register (address 1Fh); reset value: EBh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------------|---|---|------------|---|---|---|---|
| Symbol | BR_T0[2:0] | | | BR_T1[4:0] | | | | |
| Access | R/W | | | R/W | | | | |

Table 84. SerialSpeedReg register bit descriptions

| Bit | Symbol | Description |
|--------|------------|---|
| 7 to 5 | BR_T0[2:0] | factor BR_T0 adjusts the transfer speed: for description, see Section 8.1.3.2 on page 12 |
| 4 to 0 | BR_T1[4:0] | factor BR_T1 adjusts the transfer speed: for description, see Section 8.1.3.2 on page 12 |

9.3.3 Page 2: Configuration

9.3.3.1 Reserved register 20h

Functionality is reserved for future use.

Table 85. Reserved register (address 20h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---|---|---|---|----------|---|---|---|
| Symbol | | | | | - | | | |
| Access | | | | | reserved | | | |

Table 86. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.3.2 CRCResultReg registers

Shows the MSB and LSB values of the CRC calculation.

Remark: The CRC is split into two 8-bit registers.

Table 87. CRCResultReg (higher bits) register (address 21h); reset value: FFh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---|---|---|---|-------------------|---|---|---|
| Symbol | | | | | CRCResultMSB[7:0] | | | |
| Access | | | | | R | | | |

Table 88. CRCResultReg register higher bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|--|
| 7 to 0 | CRCResultMSB [7:0] | shows the value of the CRCResultReg register's most significant byte only valid if Status1Reg register's CRCReady bit is set to logic 1 |

Table 89. CRCResultReg (lower bits) register (address 22h); reset value: FFh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---|---|---|---|-------------------|---|---|---|
| Symbol | | | | | CRCResultLSB[7:0] | | | |
| Access | | | | | R | | | |

Table 90. CRCResultReg register lower bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|--|
| 7 to 0 | CRCResultLSB [7:0] | shows the value of the least significant byte of the CRCResultReg register only valid if Status1Reg register's CRCReady bit is set to logic 1 |

9.3.3.3 Reserved register 23h

Functionality is reserved for future use.

Table 91. Reserved register (address 23h); reset value: 88h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 92. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.3.4 ModWidthReg register

Sets the modulation width.

Table 93. ModWidthReg register (address 24h); reset value: 26h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------|---|---|---|---|---|---|---|
| Symbol | ModWidth[7:0] | | | | | | | |
| Access | R/W | | | | | | | |

Table 94. ModWidthReg register bit descriptions

| Bit | Symbol | Description |
|--------|---------------|---|
| 7 to 0 | ModWidth[7:0] | defines the width of the Miller modulation as multiples of the carrier frequency (ModWidth + 1 / f _{clk}) the maximum value is half the bit period |

9.3.3.5 Reserved register 25h

Functionality is reserved for future use.

Table 95. Reserved register (address 25h); reset value: 87h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 96. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.3.6 RFCfgReg register

Configures the receiver gain.

Table 97. RFCfgReg register (address 26h); reset value: 48h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|----------|-------------|---|---|---|----------|---|---|--|
| Symbol | reserved | RxGain[2:0] | | | | reserved | | | |
| Access | - | R/W | | | | - | | | |

Table 98. RFCfgReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------|-------|--|
| 7 | reserved | - | reserved for future use |
| 6 to 4 | RxGain [2:0] | | defines the receiver's signal voltage gain factor: |
| | | 000 | 18 dB |
| | | 001 | 23 dB |
| | | 010 | 18 dB |
| | | 011 | 23 dB |
| | | 100 | 33 dB |
| | | 101 | 38 dB |
| | | 110 | 43 dB |
| | | 111 | 48 dB |
| 3 to 0 | reserved | - | reserved for future use |

9.3.3.7 GsNReg register

Defines the conductance of the antenna driver pins TX1 and TX2 for the n-driver when the driver is switched on.

Table 99. GsNReg register (address 27h); reset value: 88h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------------|---|---|---|-------------|---|---|---|
| Symbol | CWGsN[3:0] | | | | ModGsN[3:0] | | | |
| Access | R/W | | | | R/W | | | |

Table 100. GsNReg register bit descriptions

| Bit | Symbol | Description |
|--------|--------------|--|
| 7 to 4 | CWGsN [3:0] | defines the conductance of the output n-driver during periods without modulation which can be used to regulate the output power and subsequently current consumption and operating distance Remark: the conductance value is binary-weighted during soft Power-down mode the highest bit is forced to logic 1 value is only used if driver TX1 or TX2 is switched on |
| 3 to 0 | ModGsN [3:0] | defines the conductance of the output n-driver during periods without modulation which can be used to regulate the modulation index Remark: the conductance value is binary weighted during soft Power-down mode the highest bit is forced to logic 1 value is only used if driver TX1 or TX2 is switched on |

9.3.3.8 CWGsPReg register

Defines the conductance of the p-driver output during periods of no modulation.

Table 101. CWGsPReg register (address 28h); reset value: 20h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|----------|---|------------|---|---|---|---|---|--|
| Symbol | reserved | | CWGsP[5:0] | | | | | | |
| Access | - | | R/W | | | | | | |

Table 102. CWGsPReg register bit descriptions

| Bit | Symbol | Description |
|--------|------------|--|
| 7 to 6 | reserved | reserved for future use |
| 5 to 0 | CWGsP[5:0] | defines the conductance of the p-driver output which can be used to regulate the output power and subsequently current consumption and operating distance Remark: the conductance value is binary weighted during soft Power-down mode the highest bit is forced to logic 1 |

9.3.3.9 ModGsPReg register

Defines the conductance of the p-driver output during modulation.

Table 103. ModGsPReg register (address 29h); reset value: 20h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|----------|---|-------------|---|---|---|---|---|--|
| Symbol | reserved | | ModGsP[5:0] | | | | | | |
| Access | - | | R/W | | | | | | |

Table 104. ModGsPReg register bit descriptions

| Bit | Symbol | Description |
|--------|-------------|--|
| 7 to 6 | reserved | reserved for future use |
| 5 to 0 | ModGsP[5:0] | defines the conductance of the p-driver output during modulation which can be used to regulate the modulation index Remark: the conductance value is binary weighted during soft Power-down mode the highest bit is forced to logic 1 if the TxASKReg register's Force100ASK bit is set to logic 1 the value of ModGsP has no effect |

9.3.3.10 TModeReg and TPrescalerReg registers

These registers define the timer settings.

Remark: The TPrescaler setting higher 4 bits are in the TModeReg register and the lower 8 bits are in the TPrescalerReg register.

Table 105. TModeReg register (address 2Ah); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------|-------------|---|--------------|--------------------|---|---|---|
| Symbol | TAuto | TGated[1:0] | | TAutoRestart | TPrescaler_Hi[3:0] | | | |
| Access | R/W | R/W | | R/W | R/W | | | |

Table 106. TModeReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------------|-------|---|
| 7 | TAuto | 1 | timer starts automatically at the end of the transmission in all communication modes at all speeds if the RxModeReg register's RxMultiple bit is not set, the timer stops immediately after receiving the 5th bit (1 start bit, 4 data bits) if the RxMultiple bit is set to logic 1 the timer never stops, in which case the timer can be stopped by setting the ControlReg register's TStopNow bit to logic 1 |
| | | 0 | indicates that the timer is not influenced by the protocol |
| 6 to 5 | TGated[1:0] | | internal timer is running in gated mode Remark: in gated mode, the Status1Reg register's TRunning bit is logic 1 when the timer is enabled by the TModeReg register's TGated[1:0] bits this bit does not influence the gating signal |
| | | | 00 non-gated mode |
| | | | 01 gated by pin MFIN |
| | | | 10 gated by pin AUX1 |
| | | | 11 - |
| 4 | TAutoRestart | 1 | timer automatically restarts its count-down from the 16-bit timer reload value instead of counting down to zero |
| | | 0 | timer decrements to 0 and the ComlrqReg register's TimerIRq bit is set to logic 1 |
| 3 to 0 | TPrescaler_Hi[3:0] | - | defines the higher 4 bits of the TPrescaler value The following formula is used to calculate the timer frequency if the DemodReg register's TPrescalEven bit in Demot Regis set to logic 0: $f_{\text{timer}} = 13.56 \text{ MHz} / (2^{\text{TPrescaler}} + 1)$. Where TPrescaler = [TPrescaler_Hi:TPrescaler_Lo] (TPrescaler value on 12 bits) (Default TPrescalEven bit is logic 0) The following formula is used to calculate the timer frequency if the DemodReg register's TPrescalEven bit is set to logic 1: $f_{\text{timer}} = 13.56 \text{ MHz} / (2^{\text{TPrescaler}} + 2)$. See Section 8.5 "Timer unit" . |

Table 107. TPrescalerReg register (address 2Bh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------------|---|---|---|---|---|---|---|
| Symbol | TPrescaler_Lo[7:0] | | | | | | | |
| Access | R/W | | | | | | | |

Table 108. TPrescalerReg register bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|--|
| 7 to 0 | TPrescaler_Lo[7:0] | <p>defines the lower 8 bits of the TPrescaler value</p> <p>The following formula is used to calculate the timer frequency if the DemodReg register's TPrescalEven bit is set to logic 0: $f_{timer} = 13.56 \text{ MHz} / (2^{TPrescaler_Lo} + 1)$.</p> <p>Where TPreScaler = [TPrescaler_Hi:TPrescaler_Lo] (TPrescaler value on 12 bits) (Default TPrescalEven bit is logic 0)</p> <p>The following formula is used to calculate the timer frequency if the DemodReg register's TPrescalEven bit inDemoReg is set to logic 1: $f_{timer} = 13.56 \text{ MHz} / (2^{TPrescaler_Lo} + 2)$.</p> <p>See Section 8.5 "Timer unit".</p> |

9.3.3.11 TReloadReg register

Defines the 16-bit timer reload value.

Remark: The reload value bits are contained in two 8-bit registers.

Table 109. TReloadReg (higher bits) register (address 2Ch); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------------|---|---|---|---|---|---|---|
| Symbol | TReloadVal_Hi[7:0] | | | | | | | |
| Access | R/W | | | | | | | |

Table 110. TReloadReg register higher bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|---|
| 7 to 0 | TReloadVal_Hi[7:0] | defines the higher 8 bits of the 16-bit timer reload value on a start event, the timer loads the timer reload value changing this register affects the timer only at the next start event |

Table 111. TReloadReg (lower bits) register (address 2Dh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------------|---|---|---|---|---|---|---|
| Symbol | TReloadVal_Lo[7:0] | | | | | | | |
| Access | R/W | | | | | | | |

Table 112. TReloadReg register lower bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|--|
| 7 to 0 | TReloadVal_Lo[7:0] | defines the lower 8 bits of the 16-bit timer reload value on a start event, the timer loads the timer reload value changing this register affects the timer only at the next start event |

9.3.3.12 TCounterValReg register

Contains the timer value.

Remark: The timer value bits are contained in two 8-bit registers.

Table 113. TCounterValReg (higher bits) register (address 2Eh); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------------|---|---|---|---|---|---|---|
| Symbol | TCounterVal_Hi[7:0] | | | | | | | |
| Access | R | | | | | | | |

Table 114. TCounterValReg register higher bit descriptions

| Bit | Symbol | Description |
|--------|---------------------|---------------------------|
| 7 to 0 | TCounterVal_Hi[7:0] | timer value higher 8 bits |

Table 115. TCounterValReg (lower bits) register (address 2Fh); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---------------------|---|---|---|---|---|---|---|
| Symbol | TCounterVal_Lo[7:0] | | | | | | | |
| Access | R | | | | | | | |

Table 116. TCounterValReg register lower bit descriptions

| Bit | Symbol | Description |
|--------|---------------------|--------------------------|
| 7 to 0 | TCounterVal_Lo[7:0] | timer value lower 8 bits |

9.3.4 Page 3: Test

9.3.4.1 Reserved register 30h

Functionality is reserved for future use.

Table 117. Reserved register (address 30h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 118. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------|
| 7 to 0 | reserved | reserved for future use |

9.3.4.2 TestSel1Reg register

General test signal configuration.

Table 119. TestSel1Reg register (address 31h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 120. TestSel1Reg register bit descriptions

| Bit | Symbol | Description |
|--------|--------------------|---|
| 7 to 3 | reserved | reserved for future use |
| 2 to 0 | TstBusBitSel [2:0] | selects a test bus signal which is output at pin MFOUT if AnalogSelAux2[3:0] = FFh in AnalogTestReg register, test bus signal is also output at pins AUX1 or AUX2 |

9.3.4.3 TestSel2Reg register

General test signal configuration and PRBS control.

Table 121. TestSel2Reg register (address 32h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------------|-------|--------|-----------------|---|---|---|---|
| Symbol | TstBusFlip | PRBS9 | PRBS15 | TestBusSel[4:0] | | | | |
| Access | R/W | R/W | R/W | R/W | | | | |

Table 122. TestSel2Reg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|---|
| 7 | TstBusFlip | 1 | test bus is mapped to the parallel port in the following order: TstBusBit4, TstBusBit3, TstBusBit2, TstBusBit6, TstBusBit5, TstBusBit0; see Section 16.1 on page 82 |
| 6 | PRBS9 | - | starts and enables the PRBS9 sequence according to ITU-T0150 Remark: all relevant registers to transmit data must be configured before entering PRBS9 mode the data transmission of the defined sequence is started by the Transmit command |
| 5 | PRBS15 | - | starts and enables the PRBS15 sequence according to ITU-T0150 Remark: all relevant registers to transmit data must be configured before entering PRBS15 mode the data transmission of the defined sequence is started by the Transmit command |
| 4 to 0 | TestBusSel[4:0] | - | selects the test bus; see Section 16.1 “Test signals” |

9.3.4.4 TestPinEnReg register

Enables the test bus pin output driver.

Table 123. TestPinEnReg register (address 33h); reset value: 80h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------------|----------------|---|---|---|---|----------|---|
| Symbol | RS232LineEn | TestPinEn[5:0] | | | | | reserved | |
| Access | R/W | R/W | | | | | - | |

Table 124. TestPinEnReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|-----------------|-------|--|
| 7 | RS232LineEn | 0 | serial UART lines MX and DTRQ are disabled |
| 6 to 1 | TestPinEn [5:0] | - | <p>enables the output driver on one of the data pins D1 to D7 which outputs a test signal</p> <p>Example:</p> <ul style="list-style-type: none"> setting bit 1 to logic 1 enables pin D1 output setting bit 5 to logic 1 enables pin D5 output <p>Remark: If the SPI is used, only pins D1 to D4 can be used. If the serial UART interface is used and the RS232LineEn bit is set to logic 1 only pins D1 to D4 can be used.</p> |
| 0 | reserved | - | reserved for future use |

9.3.4.5 TestPinValueReg register

Defines the HIGH and LOW values for the test port D1 to D7 when it is used as I/O.

Table 125. TestPinValueReg register (address 34h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-------|---|---|---|-------------------|---|---|----------|
| Symbol | UselO | | | | TestPinValue[5:0] | | | reserved |
| Access | R/W | | | | R/W | | | - |

Table 126. TestPinValueReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|--------------------|-------|--|
| 7 | UselO | 1 | <p>enables the I/O functionality for the test port when one of the serial interfaces is used</p> <p>the input/output behavior is defined by value TestPinEn[5:0] in the TestPinEnReg register</p> <p>the value for the output behavior is defined by TestPinValue[5:0]</p> |
| 6 to 1 | TestPinValue [5:0] | - | <p>defines the value of the test port when it is used as I/O and each output must be enabled by TestPinEn[5:0] in the TestPinEnReg register</p> <p>Remark: Reading the register indicates the status of pins D6 to D1 if the UselO bit is set to logic 1. If the UselO bit is set to logic 0, the value of the TestPinValueReg register is read back.</p> |
| 0 | reserved | - | reserved for future use |

9.3.4.6 TestBusReg register

Shows the status of the internal test bus.

Table 127. TestBusReg register (address 35h); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|---|---|---|---|--------------|---|---|---|
| Symbol | | | | | TestBus[7:0] | | | |
| Access | | | | | R | | | |

Table 128. TestBusReg register bit descriptions

| Bit | Symbol | Description |
|--------|--------------|---|
| 7 to 0 | TestBus[7:0] | shows the status of the internal test bus the test bus is selected using the TestSel2Reg register; see Section 16.1 on page 82 |

9.3.4.7 AutoTestReg register

Controls the digital self-test.

Table 129. AutoTestReg register (address 36h); reset value: 40h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|--------|-----|---|---|---------------|---|---|
| Symbol | reserved | AmpRcv | RFT | | | SelfTest[3:0] | | |
| Access | - | R/W | - | | | R/W | | |

Table 130. AutoTestReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|---------------|-------|--|
| 7 | reserved | - | reserved for production tests |
| 6 | AmpRcv | 1 | internal signal processing in the receiver chain is performed non-linearly which increases the operating distance in communication modes at 106 kBd Remark: due to non-linearity, the effect of the RxThresholdReg register's MinLevel[3:0] and the CollLevel[2:0] values is also non-linear |
| 5 to 4 | RFT | - | reserved for production tests |
| 3 to 0 | SelfTest[3:0] | - | enables the digital self test the self test can also be started by the CalcCRC command; see Section 10.3.1.4 on page 71 the self test is enabled by value 1001b Remark: for default operation the self test must be disabled by value 0000b |

9.3.4.8 VersionReg register

Shows the MFRC522 software version.

Table 131. VersionReg register (address 37h); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------|---|---|---|---|---|---|---|
| Symbol | Version[7:0] | | | | | | | |
| Access | R | | | | | | | |

Table 132. VersionReg register bit descriptions

| Bit | Symbol | Description |
|--------|---------|--|
| 7 to 4 | Chiptye | '9' stands for MFRC522 |
| 3 to 0 | Version | '1' stands for MFRC522 version 1.0 and '2' stands for MFRC522 version 2.0. |

MFRC522 version 1.0 software version is: 91h.

MFRC522 version 2.0 software version is: 92h.

9.3.4.9 AnalogTestReg register

Determines the analog output test signal at, and status of, pins AUX1 and AUX2.

Table 133. AnalogTestReg register (address 38h); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------------------|---|---|---|---|--------------------|---|---|
| Symbol | AnalogSelAux1[3:0] | | | | | AnalogSelAux2[3:0] | | |
| Access | R/W | | | | | R/W | | |

Table 134. AnalogTestReg register bit descriptions

| Bit | Symbol | Value | Description |
|--------|------------------------|-------|--|
| 7 to 4 | AnalogSelAux1 [3:0] | | controls pin AUX1 |
| | | 0000 | 3-state |
| | | 0001 | output of TestDAC1 (AUX1), output of TestDAC2 (AUX2) ^[1] |
| | | 0010 | test signal Corr ^[1] |
| | | 0011 | reserved |
| | | 0100 | DAC: test signal MinLevel ^[1] |
| | | 0101 | DAC: test signal ADC_I ^[1] |
| | | 0110 | DAC: test signal ADC_Q ^[1] |
| | | 0111 | reserved |
| | | 1000 | reserved, test signal for production test ^[1] |
| | | 1001 | reserved |
| | | 1010 | HIGH |
| | | 1011 | LOW |
| | | 1100 | TxActive: at 106 kBd: HIGH during Start bit, Data bit, Parity and CRC at 212 kBd: 424 kBd and 848 kBd: HIGH during data and CRC |
| | | 1101 | RxActive: at 106 kBd: HIGH during Data bit, Parity and CRC at 212 kBd: 424 kBd and 848 kBd: HIGH during data and CRC |
| | | 1110 | subcarrier detected: 106 kBd: not applicable 212 kBd: 424 kBd and 848 kBd: HIGH during last part of data and CRC |
| | | 1111 | test bus bit as defined by the TestSel1Reg register's TstBusBitSel[2:0] bits Remark: all test signals are described in Section 16.1 on page 82 |
| 3 to 0 | AnalogSelAux2 [3:0] | - | controls pin AUX2 (see bit descriptions for AUX1) |

[1] **Remark:** Current source output; the use of 1 kΩ pull-down resistor on AUXn is recommended.

9.3.4.10 TestDAC1Reg register

Defines the test value for TestDAC1.

Table 135. TestDAC1Reg register (address 39h); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|----------|---|---------------|---|---|---|---|---|--|
| Symbol | reserved | | TestDAC1[5:0] | | | | | | |
| Access | - | | R/W | | | | | | |

Table 136. TestDAC1Reg register bit descriptions

| Bit | Symbol | Description |
|--------|---------------|--|
| 7 | reserved | reserved for production tests |
| 6 | reserved | reserved for future use |
| 5 to 0 | TestDAC1[5:0] | defines the test value for TestDAC1 output of DAC1 can be routed to AUX1 by setting value AnalogSelAux1[3:0] to 0001b in the AnalogTestReg register |

9.3.4.11 TestDAC2Reg register

Defines the test value for TestDAC2.

Table 137. TestDAC2Reg register (address 3Ah); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------|----------|---|---------------|---|---|---|---|---|--|
| Symbol | reserved | | TestDAC2[5:0] | | | | | | |
| Access | - | | R/W | | | | | | |

Table 138. TestDAC2Reg register bit descriptions

| Bit | Symbol | Description |
|--------|---------------|--|
| 7 to 6 | reserved | reserved for future use |
| 5 to 0 | TestDAC2[5:0] | defines the test value for TestDAC2 output of DAC2 can be routed to AUX2 by setting value AnalogSelAux2[3:0] to 0001b in the AnalogTestReg register |

9.3.4.12 TestADCReg register

Shows the values of ADC I and Q channels.

Table 139. TestADCReg register (address 3Bh); reset value: xxh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|------------|---|---|---|------------|---|---|---|
| Symbol | ADC_I[3:0] | | | | ADC_Q[3:0] | | | |
| Access | R | | | | R | | | |

Table 140. TestADCReg register bit descriptions

| Bit | Symbol | Description |
|--------|------------|---------------------|
| 7 to 4 | ADC_I[3:0] | ADC I channel value |
| 3 to 0 | ADC_Q[3:0] | ADC Q channel value |

9.3.4.13 Reserved register 3Ch

Functionality reserved for production test.

Table 141. Reserved register (address 3Ch); reset value: FFh bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----|---|---|---|---|---|---|---|
| Symbol | RFT | | | | | | | |
| Access | - | | | | | | | |

Table 142. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------------|
| 7 to 0 | reserved | reserved for production tests |

Table 143. Reserved register (address 3Dh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----|---|---|---|---|---|---|---|
| Symbol | RFT | | | | | | | |
| Access | - | | | | | | | |

Table 144. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------------|
| 7 to 0 | reserved | reserved for production tests |

Table 145. Reserved register (address 3Eh); reset value: 03h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----|---|---|---|---|---|---|---|
| Symbol | RFT | | | | | | | |
| Access | - | | | | | | | |

Table 146. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------------|
| 7 to 0 | reserved | reserved for production tests |

Table 147. Reserved register (address 3Fh); reset value: 00h bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----------|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Access | - | | | | | | | |

Table 148. Reserved register bit descriptions

| Bit | Symbol | Description |
|--------|----------|-------------------------------|
| 7 to 0 | reserved | reserved for production tests |

10. MFRC522 command set

10.1 General description

The MFRC522 operation is determined by a state machine capable of performing a set of commands. A command is executed by writing a command code (see [Table 149](#)) to the CommandReg register.

Arguments and/or data necessary to process a command are exchanged via the FIFO buffer.

10.2 General behavior

- Each command that needs a data bit stream (or data byte stream) as an input immediately processes any data in the FIFO buffer. An exception to this rule is the Transceive command. Using this command, transmission is started with the BitFramingReg register's StartSend bit.
- Each command that needs a certain number of arguments, starts processing only when it has received the correct number of arguments from the FIFO buffer.
- The FIFO buffer is not automatically cleared when commands start. This makes it possible to write command arguments and/or the data bytes to the FIFO buffer and then start the command.
- Each command can be interrupted by the host writing a new command code to the CommandReg register, for example, the Idle command.

10.3 MFRC522 command overview

Table 149. Command overview

| Command | Command code | Action |
|-------------------|--------------|---|
| Idle | 0000 | no action, cancels current command execution |
| Mem | 0001 | stores 25 bytes into the internal buffer |
| Generate RandomID | 0010 | generates a 10-byte random ID number |
| CalcCRC | 0011 | activates the CRC coprocessor or performs a self test |
| Transmit | 0100 | transmits data from the FIFO buffer |
| NoCmdChange | 0111 | no command change, can be used to modify the CommandReg register bits without affecting the command, for example, the PowerDown bit |
| Receive | 1000 | activates the receiver circuits |
| Transceive | 1100 | transmits data from FIFO buffer to antenna and automatically activates the receiver after transmission |
| - | 1101 | reserved for future use |
| MFAuthent | 1110 | performs the MIFARE standard authentication as a reader |
| SoftReset | 1111 | resets the MFRC522 |

10.3.1 MFRC522 command descriptions

10.3.1.1 Idle

Places the MFRC522 in Idle mode. The Idle command also terminates itself.

10.3.1.2 Mem

Transfers 25 bytes from the FIFO buffer to the internal buffer.

To read out the 25 bytes from the internal buffer the Mem command must be started with an empty FIFO buffer. In this case, the 25 bytes are transferred from the internal buffer to the FIFO.

During a hard power-down (using pin NRSTPD), the 25 bytes in the internal buffer remain unchanged and are only lost if the power supply is removed from the MFRC522.

This command automatically terminates when finished and the Idle command becomes active.

10.3.1.3 Generate RandomID

This command generates a 10-byte random number which is initially stored in the internal buffer. This then overwrites the 10 bytes in the internal 25-byte buffer. This command automatically terminates when finished and the MFRC522 returns to Idle mode.

10.3.1.4 CalcCRC

The FIFO buffer content is transferred to the CRC coprocessor and the CRC calculation is started. The calculation result is stored in the CRCResultReg register. The CRC calculation is not limited to a dedicated number of bytes. The calculation is not stopped when the FIFO buffer is empty during the data stream. The next byte written to the FIFO buffer is added to the calculation.

The CRC preset value is controlled by the ModeReg register's CRCPreset[1:0] bits. The value is loaded in to the CRC coprocessor when the command starts.

This command must be terminated by writing a command to the CommandReg register, such as, the Idle command.

If the AutoTestReg register's SelfTest[3:0] bits are set correctly, the MFRC522 enters Self Test mode. Starting the CalcCRC command initiates a digital self test. The result of the self test is written to the FIFO buffer.

10.3.1.5 Transmit

The FIFO buffer content is immediately transmitted after starting this command. Before transmitting the FIFO buffer content, all relevant registers must be set for data transmission.

This command automatically terminates when the FIFO buffer is empty. It can be terminated by another command written to the CommandReg register.

10.3.1.6 NoCmdChange

This command does not influence any running command in the CommandReg register. It can be used to manipulate any bit except the CommandReg register Command[3:0] bits, for example, the RcvOff bit or the PowerDown bit.

10.3.1.7 Receive

The MFRC522 activates the receiver path and waits for a data stream to be received. The correct settings must be chosen before starting this command.

This command automatically terminates when the data stream ends. This is indicated either by the end of frame pattern or by the length byte depending on the selected frame type and speed.

Remark: If the RxModeReg register's RxMultiple bit is set to logic 1, the Receive command will not automatically terminate. It must be terminated by starting another command in the CommandReg register.

10.3.1.8 Transceive

This command continuously repeats the transmission of data from the FIFO buffer and the reception of data from the RF field. The first action is transmit and after transmission the command is changed to receive a data stream.

Each transmit process must be started by setting the BitFramingReg register's StartSend bit to logic 1. This command must be cleared by writing any command to the CommandReg register.

Remark: If the RxModeReg register's RxMultiple bit is set to logic 1, the Transceive command never leaves the receive state because this state cannot be cancelled automatically.

10.3.1.9 MFAuthent

This command manages MIFARE authentication to enable a secure communication to any MIFARE Mini, MIFARE 1K and MIFARE 4K card. The following data is written to the FIFO buffer before the command can be activated:

- Authentication command code (60h, 61h)
- Block address
- Sector key byte 0
- Sector key byte 1
- Sector key byte 2
- Sector key byte 3
- Sector key byte 4
- Sector key byte 5
- Card serial number byte 0
- Card serial number byte 1
- Card serial number byte 2
- Card serial number byte 3

In total 12 bytes are written to the FIFO.

Remark: When the MFAuthent command is active all access to the FIFO buffer is blocked. However, if there is access to the FIFO buffer, the ErrorReg register's WrErr bit is set.

This command automatically terminates when the MIFARE card is authenticated and the Status2Reg register's MFCrypto1On bit is set to logic 1.

This command does not terminate automatically if the card does not answer, so the timer must be initialized to automatic mode. In this case, in addition to the IdleIRq bit, the TimerIRq bit can be used as the termination criteria. During authentication processing, the RxIRq bit and TxIRq bit are blocked. The Crypto1On bit is only valid after termination of the MFAuthent command, either after processing the protocol or writing Idle to the CommandReg register.

If an error occurs during authentication, the ErrorReg register's ProtocolErr bit is set to logic 1 and the Status2Reg register's Crypto1On bit is set to logic 0.

10.3.1.10 SoftReset

This command performs a reset of the device. The configuration data of the internal buffer remains unchanged. All registers are set to the reset values. This command automatically terminates when finished.

Remark: The SerialSpeedReg register is reset and therefore the serial data rate is set to 9.6 kBd.

11. Limiting values

Table 150. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

| Symbol | Parameter | Conditions | Min | Max | Unit |
|----------------|---------------------------------|---|----------------------|----------------------|------|
| V_{DDA} | analog supply voltage | | -0.5 | +4.0 | V |
| V_{DDD} | digital supply voltage | | -0.5 | +4.0 | V |
| $V_{DD(PVDD)}$ | PVDD supply voltage | | -0.5 | +4.0 | V |
| $V_{DD(TVDD)}$ | TVDD supply voltage | | -0.5 | +4.0 | V |
| $V_{DD(SVDD)}$ | SVDD supply voltage | | -0.5 | +4.0 | V |
| V_I | input voltage | all input pins except pins MFIN and RX | $V_{SS(PVSS)} - 0.5$ | $V_{DD(PVDD)} + 0.5$ | V |
| | | pin MFIN | $V_{SS(PVSS)} - 0.5$ | $V_{DD(SVDD)} + 0.5$ | V |
| P_{tot} | total power dissipation | per package; and V_{DDD} in shortcut mode | - | 200 | mW |
| T_j | junction temperature | | - | 100 | °C |
| V_{ESD} | electrostatic discharge voltage | HBM; 1500 Ω, 100 pF; JESD22-A114-B | - | 2000 | V |
| | | MM; 0.75 μH, 200 pF; JESD22-A114-A | - | 200 | V |
| | | Charged device model; JESD22-C101-A | | | |
| | | on all pins | - | 200 | V |
| | | on all pins except SVDD in TBFBA64 package | - | 500 | V |

12. Recommended operating conditions

Table 151. Operating conditions

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit | |
|----------------|------------------------|---|--------|-----|-----|------|----|
| V_{DDA} | analog supply voltage | $V_{DD(PVDD)} \leq V_{DDA} = V_{DDD} = V_{DD(TVDD)}$; $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | [1][2] | 2.5 | 3.3 | 3.6 | V |
| V_{DDD} | digital supply voltage | $V_{DD(PVDD)} \leq V_{DDA} = V_{DDD} = V_{DD(TVDD)}$; $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | [1][2] | 2.5 | 3.3 | 3.6 | V |
| $V_{DD(TVDD)}$ | TVDD supply voltage | $V_{DD(PVDD)} \leq V_{DDA} = V_{DDD} = V_{DD(TVDD)}$; $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | [1][2] | 2.5 | 3.3 | 3.6 | V |
| $V_{DD(PVDD)}$ | PVDD supply voltage | $V_{DD(PVDD)} \leq V_{DDA} = V_{DDD} = V_{DD(TVDD)}$; $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | [3] | 1.6 | 1.8 | 3.6 | V |
| $V_{DD(SVDD)}$ | SVDD supply voltage | $V_{SSA} = V_{SSD} = V_{SS(PVSS)} = V_{SS(TVSS)} = 0 \text{ V}$ | | 1.6 | - | 3.6 | V |
| T_{amb} | ambient temperature | HVQFN32 | | -25 | - | +85 | °C |

[1] Supply voltages below 3 V reduce the performance (the achievable operating distance).

[2] V_{DDA} , V_{DDD} and $V_{DD(TVDD)}$ must always be the same voltage.[3] $V_{DD(PVDD)}$ must always be the same or lower voltage than V_{DDD} .

13. Thermal characteristics

Table 152. Thermal characteristics

| Symbol | Parameter | Conditions | Package | Typ | Unit |
|---------------|---|---|---------|-----|------|
| $R_{th(j-a)}$ | thermal resistance from junction to ambient | in still air with exposed pin soldered on a 4 layer JEDEC PCB | HVQFN32 | 40 | K/W |

14. Characteristics

Table 153. Characteristics

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|------------------------------------|--|-------------------|-----|-------------------|----------|
| Input characteristics | | | | | | |
| Pins EA, I₂C and NRSTPD | | | | | | |
| I_{LI} | input leakage current | | -1 | - | +1 | μA |
| V_{IH} | HIGH-level input voltage | | $0.7V_{DD(PVDD)}$ | - | - | V |
| V_{IL} | LOW-level input voltage | | - | - | $0.3V_{DD(PVDD)}$ | V |
| Pin MFIN | | | | | | |
| I_{LI} | input leakage current | | -1 | - | +1 | μA |
| V_{IH} | HIGH-level input voltage | | $0.7V_{DD(SVDD)}$ | - | - | V |
| V_{IL} | LOW-level input voltage | | - | - | $0.3V_{DD(SVDD)}$ | V |
| Pin SDA | | | | | | |
| I_{LI} | input leakage current | | -1 | - | +1 | μA |
| V_{IH} | HIGH-level input voltage | | $0.7V_{DD(PVDD)}$ | - | - | V |
| V_{IL} | LOW-level input voltage | | - | - | $0.3V_{DD(PVDD)}$ | V |
| Pin RX[1] | | | | | | |
| V_i | input voltage | | -1 | - | $V_{DDA} +1$ | V |
| C_i | input capacitance | $V_{DDA} = 3 V$; receiver active; $V_{RX(p-p)} = 1 V$; 1.5 V (DC) offset | - | 10 | - | pF |
| R_i | input resistance | $V_{DDA} = 3 V$; receiver active; $V_{RX(p-p)} = 1 V$; 1.5 V (DC) offset | - | 350 | - | Ω |
| <i>Input voltage range; see Figure 24</i> | | | | | | |
| $V_{i(p-p)(min)}$ | minimum peak-to-peak input voltage | Manchester encoded; $V_{DDA} = 3 V$ | - | 100 | - | mV |
| $V_{i(p-p)(max)}$ | maximum peak-to-peak input voltage | Manchester encoded; $V_{DDA} = 3 V$ | - | 4 | - | V |
| <i>Input sensitivity; see Figure 24</i> | | | | | | |
| V_{mod} | modulation voltage | minimum Manchester encoded; $V_{DDA} = 3 V$; RxGain[2:0] = 111b (48 dB) | - | 5 | - | mV |
| Pin OSCIN | | | | | | |
| I_{LI} | input leakage current | | -1 | - | +1 | μA |
| V_{IH} | HIGH-level input voltage | | $0.7V_{DDA}$ | - | - | V |
| V_{IL} | LOW-level input voltage | | - | - | $0.3V_{DDA}$ | V |

Table 153. Characteristics ...continued

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|-------------------------------------|---------------------------|--|--|----------------------|-----|----------------------|------|
| C_i | input capacitance | $V_{DDA} = 2.8 \text{ V}$; DC = 0.65 V; AC = 1 V (p-p) | | - | 2 | - | pF |
| Input/output characteristics | | | | | | | |
| pins D1, D2, D3, D4, D5, D6 and D7 | | | | | | | |
| I_{LI} | input leakage current | | | -1 | - | +1 | μA |
| V_{IH} | HIGH-level input voltage | | | $0.7V_{DD(PVDD)}$ | - | - | V |
| V_{IL} | LOW-level input voltage | | | - | - | $0.3V_{DD(PVDD)}$ | V |
| V_{OH} | HIGH-level output voltage | $V_{DD(PVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{DD(PVDD)} - 0.4$ | - | $V_{DD(PVDD)}$ | V |
| V_{OL} | LOW-level output voltage | $V_{DD(PVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{SS(PVSS)}$ | - | $V_{SS(PVSS)} + 0.4$ | V |
| I_{OH} | HIGH-level output current | $V_{DD(PVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| I_{OL} | LOW-level output current | $V_{DD(PVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| Output characteristics | | | | | | | |
| Pin MFOUT | | | | | | | |
| V_{OH} | HIGH-level output voltage | $V_{DD(SVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{DD(SVDD)} - 0.4$ | - | $V_{DD(SVDD)}$ | V |
| V_{OL} | LOW-level output voltage | $V_{DD(SVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{SS(PVSS)}$ | - | $V_{SS(PVSS)} + 0.4$ | V |
| I_{OL} | LOW-level output current | $V_{DD(SVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| I_{OH} | HIGH-level output current | $V_{DD(SVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| Pin IRQ | | | | | | | |
| V_{OH} | HIGH-level output voltage | $V_{DD(PVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{DD(PVDD)} - 0.4$ | - | $V_{DD(PVDD)}$ | V |
| V_{OL} | LOW-level output voltage | $V_{DD(PVDD)} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{SS(PVSS)}$ | - | $V_{SS(PVSS)} + 0.4$ | V |
| I_{OL} | LOW-level output current | $V_{DD(PVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| I_{OH} | HIGH-level output current | $V_{DD(PVDD)} = 3 \text{ V}$ | | - | - | 4 | mA |
| Pins AUX1 and AUX2 | | | | | | | |
| V_{OH} | HIGH-level output voltage | $V_{DDD} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{DDD} - 0.4$ | - | V_{DDD} | V |
| V_{OL} | LOW-level output voltage | $V_{DDD} = 3 \text{ V}$; $I_O = 4 \text{ mA}$ | | $V_{SS(PVSS)}$ | - | $V_{SS(PVSS)} + 0.4$ | V |
| I_{OL} | LOW-level output current | $V_{DDD} = 3 \text{ V}$ | | - | - | 4 | mA |
| I_{OH} | HIGH-level output current | $V_{DDD} = 3 \text{ V}$ | | - | - | 4 | mA |
| Pins TX1 and TX2 | | | | | | | |

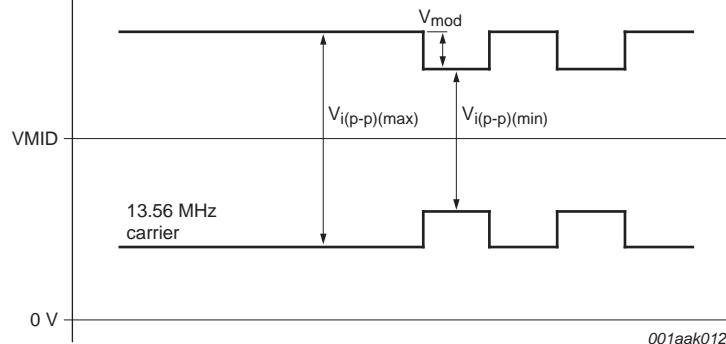
Table 153. Characteristics ...continued

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|----------------------------|---------------------------|---|-----------|------------------------------|-------|------|------|
| V _{OH} | HIGH-level output voltage | V _{DD(TVDD)} = 3 V; I _{DD(TVDD)} = 32 mA; CWGsP[5:0] = 3Fh | | V _{DD(TVDD)} – 0.15 | - | - | V |
| | | V _{DD(TVDD)} = 3 V; I _{DD(TVDD)} = 80 mA; CWGsP[5:0] = 3Fh | | V _{DD(TVDD)} – 0.4 | - | - | V |
| | | V _{DD(TVDD)} = 2.5 V; I _{DD(TVDD)} = 32 mA; CWGsP[5:0] = 3Fh | | V _{DD(TVDD)} – 0.24 | - | - | V |
| | | V _{DD(TVDD)} = 2.5 V; I _{DD(TVDD)} = 80 mA; CWGsP[5:0] = 3Fh | | V _{DD(TVDD)} – 0.64 | - | - | V |
| V _{OL} | LOW-level output voltage | V _{DD(TVDD)} = 3 V; I _{DD(TVDD)} = 32 mA; CWGsP[5:0] = 0Fh | | - | - | 0.15 | V |
| | | V _{DD(TVDD)} = 3 V; I _{DD(TVDD)} = 80 mA; CWGsP[5:0] = 0Fh | | - | - | 0.4 | V |
| | | V _{DD(TVDD)} = 2.5 V; I _{DD(TVDD)} = 32 mA; CWGsP[5:0] = 0Fh | | - | - | 0.24 | V |
| | | V _{DD(TVDD)} = 2.5 V; I _{DD(TVDD)} = 80 mA; CWGsP[5:0] = 0Fh | | - | - | 0.64 | V |
| Current consumption | | | | | | | |
| I _{pd} | power-down current | V _{DDA} = V _{DDD} = V _{DD(TVDD)} = V _{DD(PVDD)} = 3 V | | | | | |
| | | hard power-down; pin NRSTPD set LOW | [2] | - | - | 5 | µA |
| | | soft power-down; RF level detector on | [2] | - | - | 10 | µA |
| I _{DDD} | digital supply current | pin DVDD; V _{DDD} = 3 V | | - | 6.5 | 9 | mA |
| I _{DDA} | analog supply current | pin AVDD; V _{DDA} = 3 V; CommandReg register's bit RcvOff = 0 | | - | 7 | 10 | mA |
| | | pin AVDD; receiver switched off; V _{DDA} = 3 V; CommandReg register's bit RcvOff = 1 | | - | 3 | 5 | mA |
| I _{DD(PVDD)} | PVDD supply current | pin PVDD | [3] | - | - | 40 | mA |
| I _{DD(TVDD)} | TVDD supply current | pin TVDD; continuous wave | [4][5][6] | - | 60 | 100 | mA |
| I _{DD(SVDD)} | SVDD supply current | pin SVDD | [7] | - | - | 4 | mA |
| Clock frequency | | | | | | | |
| f _{clk} | clock frequency | | | - | 27.12 | - | MHz |
| δ _{clk} | clock duty cycle | | | 40 | 50 | 60 | % |
| t _{jit} | jitter time | RMS | | - | - | 10 | ps |
| Crystal oscillator | | | | | | | |

Table 153. Characteristics ...continued

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|----------------------------|------------------------------|------------|--|-----|-------|-----|----------|
| V_{OH} | HIGH-level output voltage | pin OSCOUT | | - | 1.1 | - | V |
| V_{OL} | LOW-level output voltage | pin OSCOUT | | - | 0.2 | - | V |
| C_i | input capacitance | pin OSCOUT | | - | 2 | - | pF |
| | | pin OSCIN | | - | 2 | - | pF |
| Typical input requirements | | | | | | | |
| f_{xtal} | crystal frequency | | | - | 27.12 | - | MHz |
| ESR | equivalent series resistance | | | - | - | 100 | Ω |
| C_L | load capacitance | | | - | 10 | - | pF |
| P_{xtal} | crystal power dissipation | | | - | 50 | 100 | mW |

- [1] The voltage on pin RX is clamped by internal diodes to pins AVSS and AVDD.
- [2] I_{pd} is the total current for all supplies.
- [3] $I_{DD(PVDD)}$ depends on the overall load at the digital pins.
- [4] $I_{DD(TVDD)}$ depends on $V_{DD(TVDD)}$ and the external circuit connected to pins TX1 and TX2.
- [5] During typical circuit operation, the overall current is below 100 mA.
- [6] Typical value using a complementary driver configuration and an antenna matched to 40 Ω between pins TX1 and TX2 at 13.56 MHz.
- [7] $I_{DD(SVDD)}$ depends on the load at pin MFOUT.

**Fig 24. Pin RX input voltage range**

14.1 Timing characteristics

Table 154. SPI timing characteristics

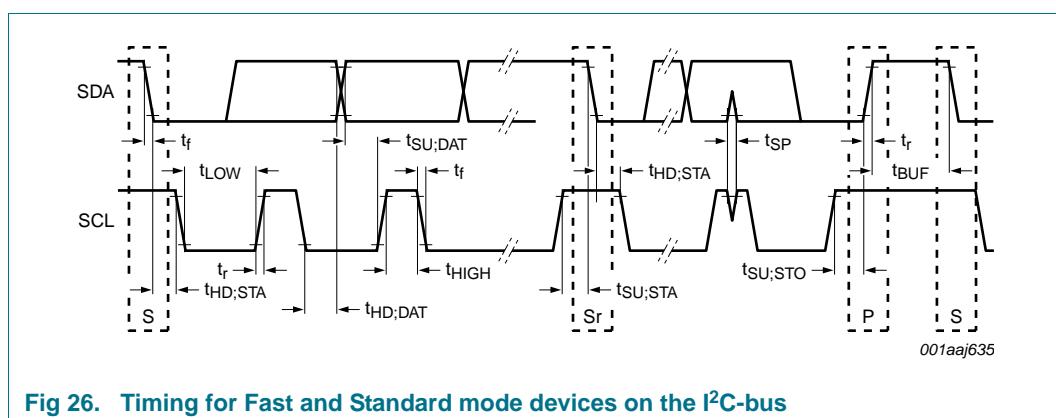
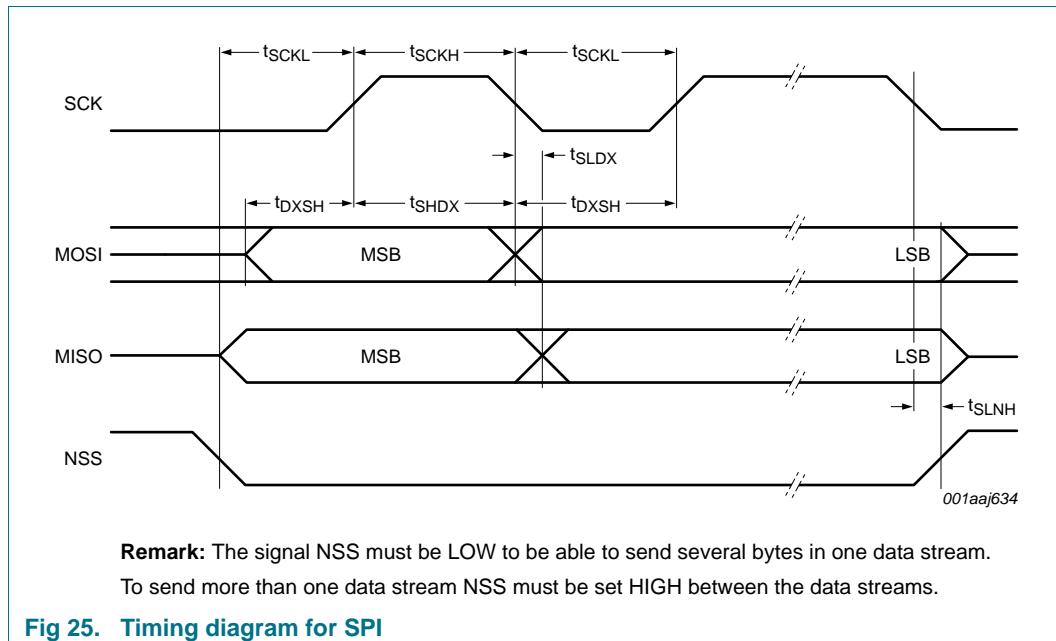
| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---------------|----------------------------------|----------------------|-----|-----|-----|------|
| t_{WL} | pulse width LOW | line SCK | 50 | - | - | ns |
| t_{WH} | pulse width HIGH | line SCK | 50 | - | - | ns |
| $t_h(SCKH-D)$ | SCK HIGH to data input hold time | SCK to changing MOSI | 25 | - | - | ns |

Table 154. SPI timing characteristics ...continued

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|-------------------|------------------------------------|----------------------|------------|------------|------------|-------------|
| $t_{su(D-SCKH)}$ | data input to SCK HIGH set-up time | changing MOSI to SCK | 25 | - | - | ns |
| $t_h(SCKL-Q)$ | SCK LOW to data output hold time | SCK to changing MISO | - | - | 25 | ns |
| $t_{(SCKL-NSSH)}$ | SCK LOW to NSS HIGH time | | 0 | - | - | ns |
| t_{NHNL} | NSS high before communication | | 50 | - | - | ns |

Table 155. I²C-bus timing in Fast mode

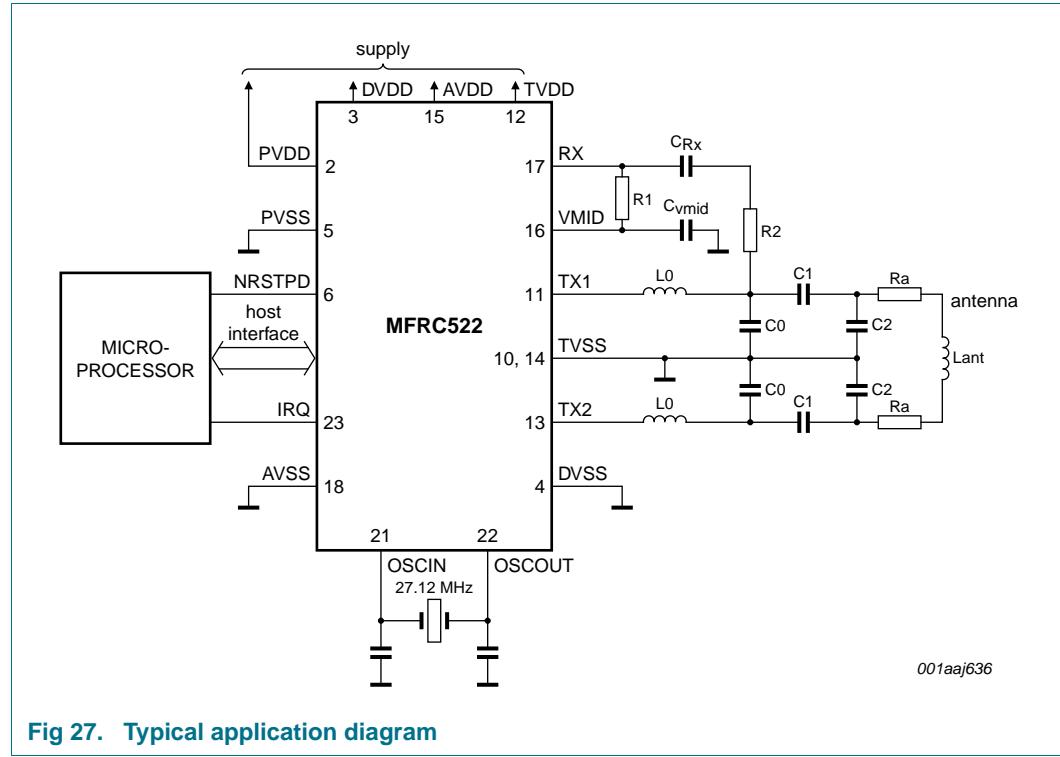
| Symbol | Parameter | Conditions | Fast mode | | High-speed mode | | Unit |
|---------------|--|---|------------------|------------|------------------------|------------|-------------|
| | | | Min | Max | Min | Max | |
| f_{SCL} | SCL clock frequency | | 0 | 400 | 0 | 3400 | kHz |
| $t_{HD;STA}$ | hold time (repeated) START condition | after this period, the first clock pulse is generated | 600 | - | 160 | - | ns |
| $t_{SU;STA}$ | set-up time for a repeated START condition | | 600 | - | 160 | - | ns |
| $t_{SU;STO}$ | set-up time for STOP condition | | 600 | - | 160 | - | ns |
| t_{LOW} | LOW period of the SCL clock | | 1300 | - | 160 | - | ns |
| t_{HIGH} | HIGH period of the SCL clock | | 600 | - | 60 | - | ns |
| $t_{HD;DAT}$ | data hold time | | 0 | 900 | 0 | 70 | ns |
| $t_{SU;DAT}$ | data set-up time | | 100 | - | 10 | - | ns |
| t_r | rise time | SCL signal | 20 | 300 | 10 | 40 | ns |
| t_f | fall time | SCL signal | 20 | 300 | 10 | 40 | ns |
| t_r | rise time | SDA and SCL signals | 20 | 300 | 10 | 80 | ns |
| t_f | fall time | SDA and SCL signals | 20 | 300 | 10 | 80 | ns |
| t_{BUF} | bus free time between a STOP and START condition | | 1.3 | - | 1.3 | - | μs |



15. Application information

A typical application diagram using a complementary antenna connection to the MFRC522 is shown in [Figure 27](#).

The antenna tuning and RF part matching is described in the application note [Ref. 1](#) and [Ref. 2](#).



16. Test information

16.1 Test signals

16.1.1 Self test

The MFRC522 has the capability to perform a digital self test. The self test is started by using the following procedure:

1. Perform a soft reset.
2. Clear the internal buffer by writing 25 bytes of 00h and implement the Config command.
3. Enable the self test by writing 09h to the AutoTestReg register.
4. Write 00h to the FIFO buffer.
5. Start the self test with the CalcCRC command.
6. The self test is initiated.
7. When the self test has completed, the FIFO buffer contains the following 64 bytes:

FIFO buffer byte values for MFRC522 version 1.0:

00h, C6h, 37h, D5h, 32h, B7h, 57h, 5Ch,
C2h, D8h, 7Ch, 4Dh, D9h, 70h, C7h, 73h,
10h, E6h, D2h, AAh, 5Eh, A1h, 3Eh, 5Ah,
14h, AFh, 30h, 61h, C9h, 70h, DBh, 2Eh,
64h, 22h, 72h, B5h, BDh, 65h, F4h, ECh,
22h, BCh, D3h, 72h, 35h, CDh, AAh, 41h,
1Fh, A7h, F3h, 53h, 14h, DEh, 7Eh, 02h,
D9h, 0Fh, B5h, 5Eh, 25h, 1Dh, 29h, 79h

FIFO buffer byte values for MFRC522 version 2.0:

00h, EBh, 66h, BAh, 57h, BFh, 23h, 95h,
D0h, E3h, 0Dh, 3Dh, 27h, 89h, 5Ch, DEh,
9Dh, 3Bh, A7h, 00h, 21h, 5Bh, 89h, 82h,
51h, 3Ah, EBh, 02h, 0Ch, A5h, 00h, 49h,
7Ch, 84h, 4Dh, B3h, CCh, D2h, 1Bh, 81h,
5Dh, 48h, 76h, D5h, 71h, 061h, 21h, A9h,
86h, 96h, 83h, 38h, CFh, 9Dh, 5Bh, 6Dh,
DCh, 15h, BAh, 3Eh, 7Dh, 95h, 03Bh, 2Fh

16.1.2 Test bus

The test bus is used for production tests. The following configuration can be used to improve the design of a system using the MFRC522. The test bus allows internal signals to be routed to the digital interface. The test bus comprises two sets of test signals which are selected using their subaddress specified in the TestSel2Reg register's TestBusSel[4:0] bits. The test signals and their related digital output pins are described in [Table 156](#) and [Table 157](#).

Table 156. Test bus signals: TestBusSel[4:0] = 07h

| Pins | Internal signal name | Description |
|------|----------------------|--|
| D6 | s_data | received data stream |
| D5 | s_coll | bit-collision detected (106 kBd only) |
| D4 | s_valid | s_data and s_coll signals are valid |
| D3 | s_over | receiver has detected a stop condition |
| D2 | RCV_reset | receiver is reset |
| D1 | - | reserved |

Table 157. Test bus signals: TestBusSel[4:0] = 0Dh

| Pins | Internal test signal name | Description |
|----------|---------------------------|---------------------------------------|
| D6 | clkstable | oscillator output signal |
| D5 | clk27/8 | oscillator output signal divided by 8 |
| D4 to D3 | - | reserved |
| D2 | clk27 | oscillator output signal |
| D1 | - | reserved |

16.1.3 Test signals on pins AUX1 or AUX2

The MFRC522 allows the user to select internal signals for measurement on pins AUX1 or AUX2. These measurements can be helpful during the design-in phase to optimize the design or used for test purposes.

[Table 158](#) shows the signals that can be switched to pin AUX1 or AUX2 by setting AnalogSelAux1[3:0] or AnalogSelAux2[3:0] in the AnalogTestReg register.

Remark: The DAC has a current output, therefore it is recommended that a 1 kΩ pull-down resistor is connected to pin AUX1 or AUX2.

Table 158. Test signal descriptions

| AnalogSelAux1[3:0] or AnalogSelAux2[3:0] value | Signal on pin AUX1 or pin AUX2 |
|--|------------------------------------|
| 0000 | 3-state |
| 0001 | DAC: register TestDAC1 or TestDAC2 |
| 0010 | DAC: test signal Corr1 |
| 0011 | reserved |
| 0100 | DAC: test signal MinLevel |
| 0101 | DAC: test signal ADC_I |
| 0110 | DAC: test signal ADC_Q |
| 0111 to 1001 | reserved |
| 1010 | HIGH |
| 1011 | LOW |
| 1100 | TxActive |

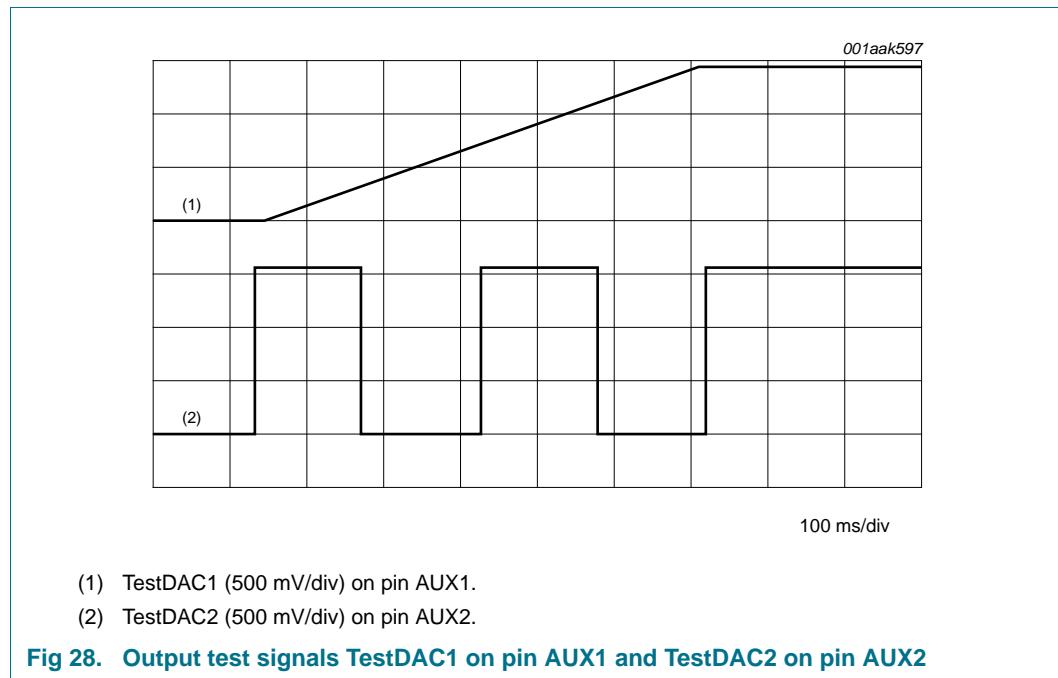
Table 158. Test signal descriptions ...continued

| AnalogSelAux1[3:0] or AnalogSelAux2[3:0] value | Signal on pin AUX1 or pin AUX2 |
|--|--------------------------------|
| 1101 | RxActive |
| 1110 | subcarrier detected |
| 1111 | TstBusBit |

16.1.3.1 Example: Output test signals TestDAC1 and TestDAC2

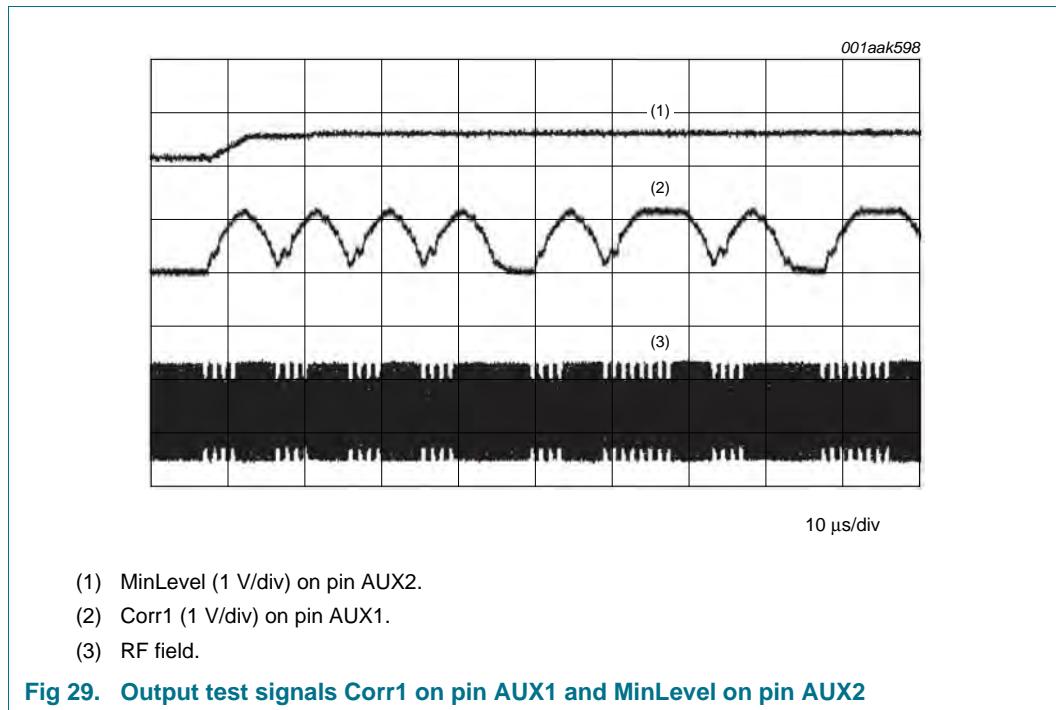
The AnalogTestReg register is set to 11h. The output on pin AUX1 has the test signal TestDAC1 and the output on pin AUX2 has the test signal TestDAC2. The signal values of TestDAC1 and TestDAC2 are controlled by the TestDAC1Reg and TestDAC2Reg registers.

[Figure 28](#) shows test signal TestDAC1 on pin AUX1 and TestDAC2 on pin AUX2 when the TestDAC1Reg register is programmed with a slope defined by values 00h to 3Fh and the TestDAC2Reg register is programmed with a rectangular signal defined by values 00h and 3Fh.



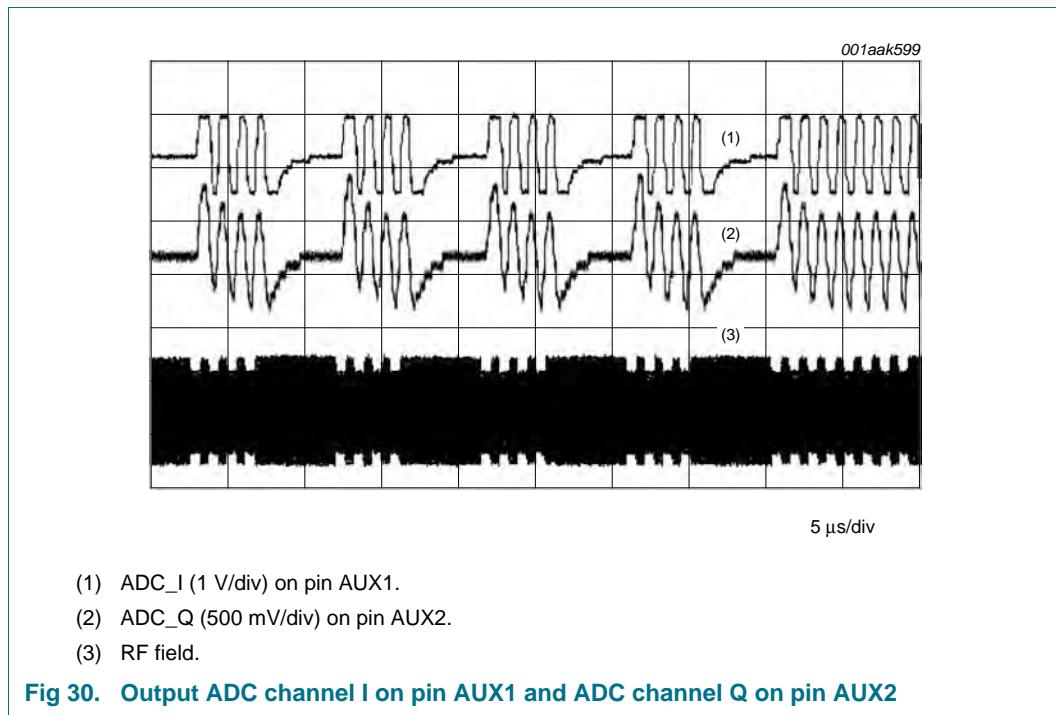
16.1.3.2 Example: Output test signals Corr1 and MinLevel

[Figure 29](#) shows test signals Corr1 and MinLevel on pins AUX1 and AUX2, respectively. The AnalogTestReg register is set to 24h.



16.1.3.3 Example: Output test signals ADC channel I and ADC channel Q

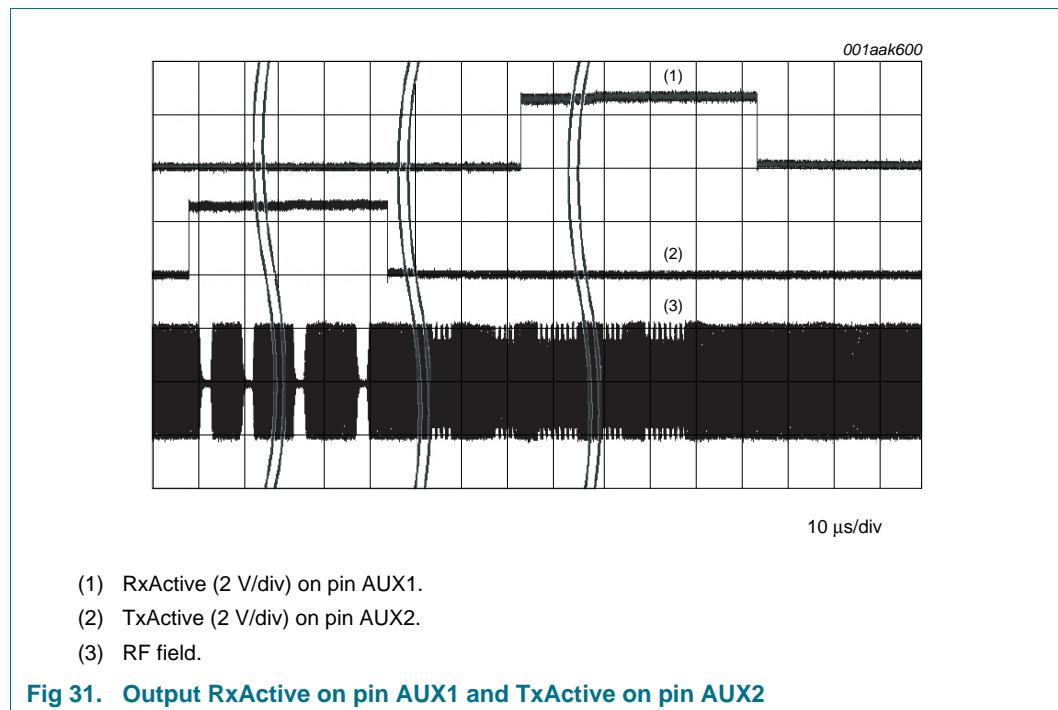
Figure 30 shows the channel behavior test signals ADC_I and ADC_Q on pins AUX1 and AUX2, respectively. The AnalogTestReg register is set to 56h.



16.1.3.4 Example: Output test signals RxActive and TxActive

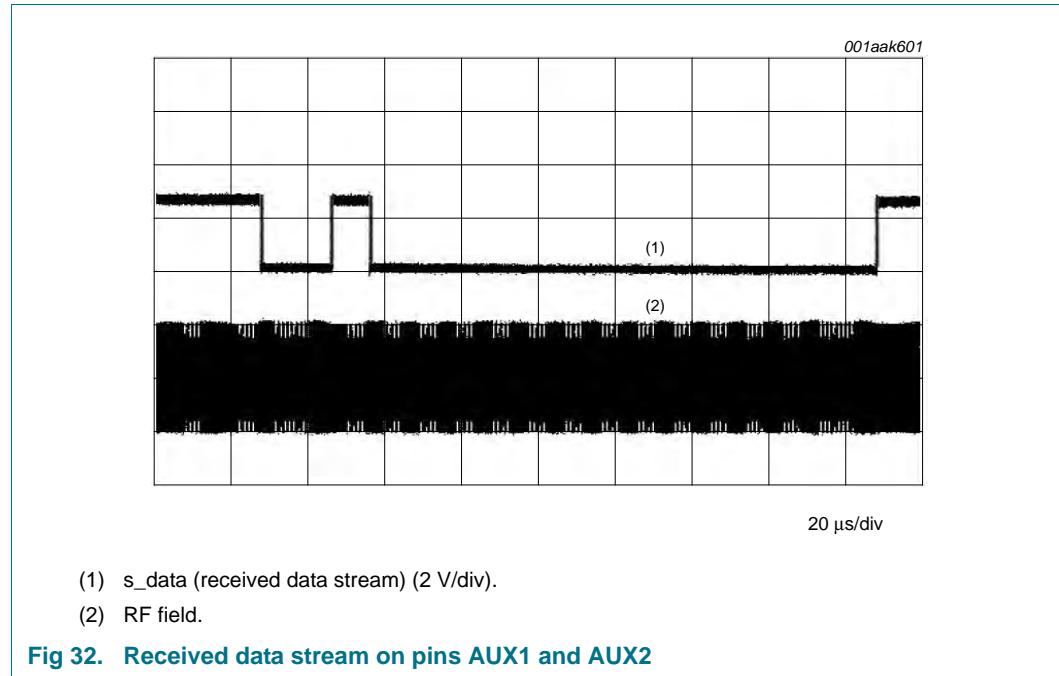
Figure 31 shows the RxActive and TxActive test signals relating to RF communication. The AnalogTestReg register is set to CDh.

- At 106 kBd, RxActive is HIGH during data bits, parity and CRC reception. Start bits are not included
- At 106 kBd, TxActive is HIGH during start bits, data bits, parity and CRC transmission
- At 212 kBd, 424 kBd and 848 kBd, RxActive is HIGH during data bits and CRC reception. Start bits are not included
- At 212 kBd, 424 kBd and 848 kBd, TxActive is HIGH during data bits and CRC transmission



16.1.3.5 Example: Output test signal RX data stream

Figure 32 shows the data stream that is currently being received. The TestSel2Reg register's TestBusSel[4:0] bits are set to 07h to enable test bus signals on pins D1 to D6; see [Section 16.1.2 on page 82](#). The TestSel1Reg register's TstBusBitSel[2:0] bits are set to 06h (pin D6 = s_data) and AnalogTestReg register is set to FFh (TstBusBit) which outputs the received data stream on pins AUX1 and AUX2.



16.1.3.6 PRBS

The pseudo-random binary sequences PRBS9 and PRBS15 are based on ITU-T0150 and are defined with the TestSel2Reg register. Transmission of either data stream is started by the Transmit command. The preamble/sync byte/start bit/parity bit are automatically generated depending on the mode selected.

Remark: All relevant registers for transmitting data must be configured in accordance with ITU-T0150 before selecting PRBS transmission.

17. Package outline

HVQFN32: plastic thermal enhanced very thin quad flat package; no leads;
32 terminals; body 5 x 5 x 0.85 mm

SOT617-1

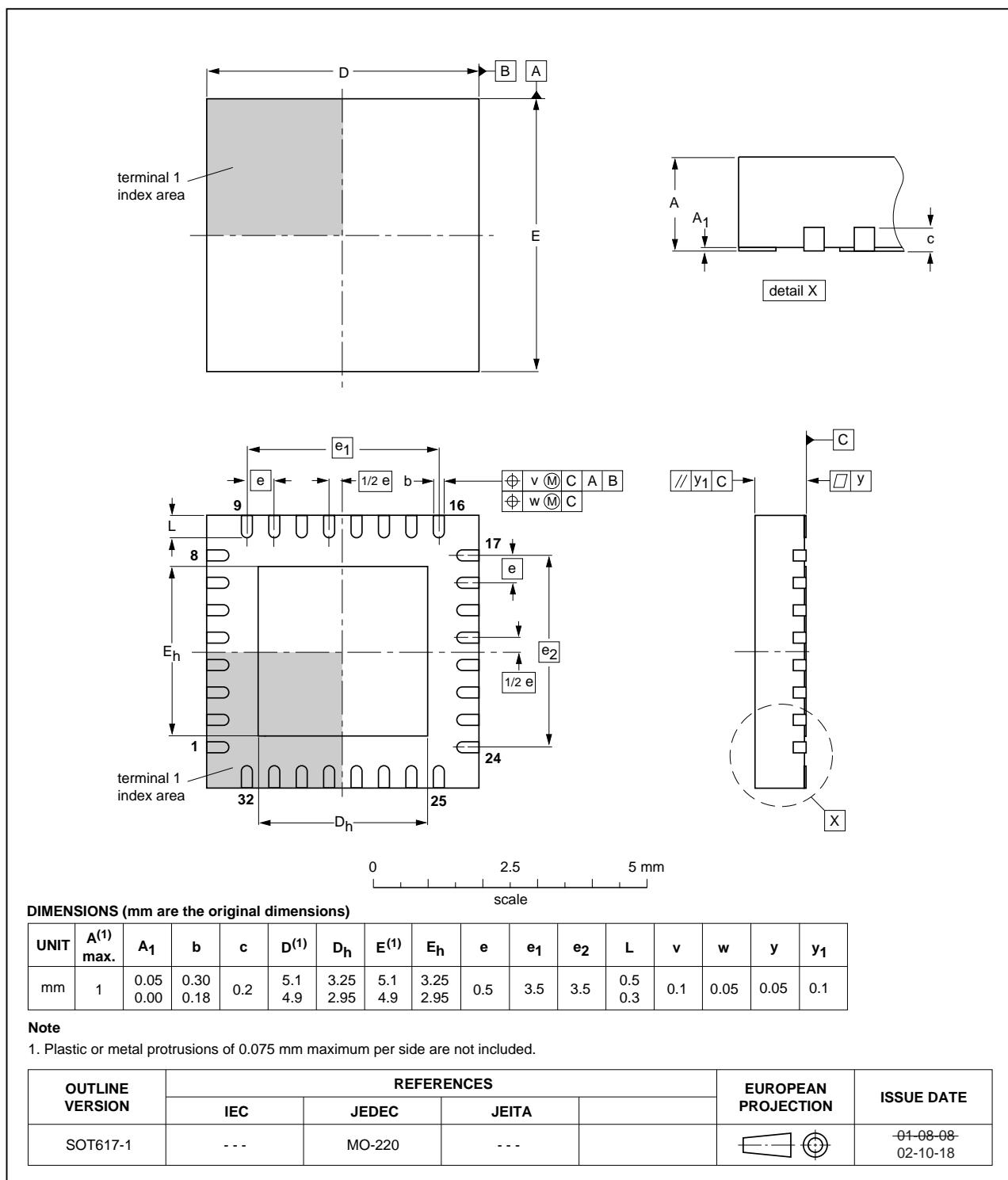


Fig 33. Package outline SOT617-1 (HVQFN32)

Detailed package information can be found at:
<http://www.nxp.com/package/SOT617-1.html>.

18. Handling information

Moisture Sensitivity Level (MSL) evaluation has been performed according to SNW-FQ-225B rev.04/07/07 (JEDEC J-STD-020C). MSL for this package is level 1 which means 260 °C convection reflow temperature.

Dry pack is not required.

Unlimited out-of-pack floor life at maximum ambient 30 °C/85 % RH.

19. Packing information

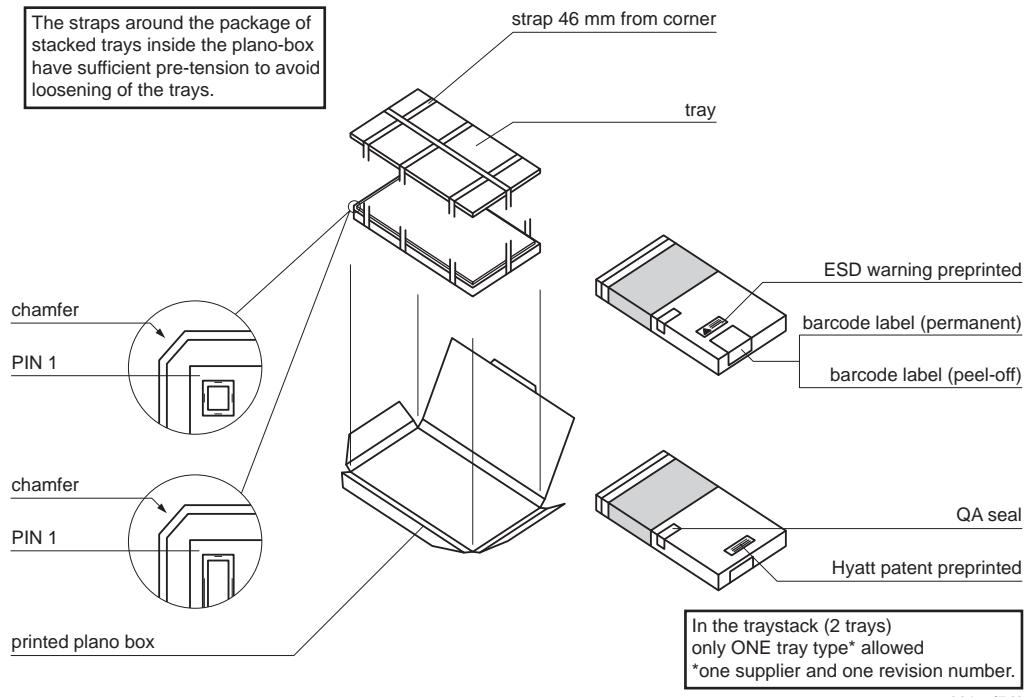


Fig 34. Packing information 1 tray

20. Abbreviations

Table 159. Abbreviations

| Acronym | Description |
|------------------|---|
| ADC | Analog-to-Digital Converter |
| BPSK | Binary Phase Shift Keying |
| CRC | Cyclic Redundancy Check |
| CW | Continuous Wave |
| DAC | Digital-to-Analog Converter |
| HBM | Human Body Model |
| I ² C | Inter-integrated Circuit |
| LSB | Least Significant Bit |
| MISO | Master In Slave Out |
| MM | Machine Model |
| MOSI | Master Out Slave In |
| MSB | Most Significant Bit |
| NRZ | Not Return to Zero |
| NSS | Not Slave Select |
| PLL | Phase-Locked Loop |
| PRBS | Pseudo-Random Bit Sequence |
| RX | Receiver |
| SOF | Start Of Frame |
| SPI | Serial Peripheral Interface |
| TX | Transmitter |
| UART | Universal Asynchronous Receiver Transmitter |

21. References

- [1] Application note — MFRC52x Reader IC Family Directly Matched Antenna Design
- [2] Application note — MIFARE (ISO/IEC 14443 A) 13.56 MHz RFID Proximity Antennas

22. Revision history

Table 160. Revision history

| Document ID | Release date | Data sheet status | Change notice | Supersedes |
|----------------|--------------|--|---------------|---------------|
| MFRC522 v.3.9 | 20160427 | Product data sheet | - | MFRC522 v.3.8 |
| Modifications: | | <ul style="list-style-type: none"> • Section 1 "Introduction" and Section 2 "General description": updated and NTAG functionality added • Descriptive title updated | | |
| MFRC522 v.3.8 | 20140917 | Product data sheet | - | MFRC522 v.3.7 |
| Modifications: | | <ul style="list-style-type: none"> • Table 150 "Limiting values": updated | | |
| MFRC522 v.3.7 | 20140326 | Product data sheet | - | MFRC522 v.3.6 |
| Modifications: | | <ul style="list-style-type: none"> • Change of descriptive title • Section 23.4 "Licenses" removed | | |
| MFRC522 v.3.6 | 20111214 | Product data sheet | - | MFRC522_35 |
| Modifications: | | <ul style="list-style-type: none"> • Section 1.1 "Differences between version 1.0 and 2.0" on page 1: added • Table 2 "Ordering information" on page 3: updated • Section 9.3.2.10 "DemodReg register" on page 53: register updated and add reference to Timer unit • Section 8.5 "Timer unit" on page 31: Pre Scaler Information for version 2.0 added • Section 9.3.4.8 "VersionReg register" on page 66: version information structured in chip information and version information updated, including version 1.0 and 2.0 • Section 16.1 "Test signals" on page 82: selftest result including values for version 1.0 and 2.0 | | |
| MFRC522_35 | 20100621 | Product data sheet | | MFRC522_34 |
| Modifications: | | <ul style="list-style-type: none"> • Section 9.3.2.10 "DemodReg register" on page 53: register updated • Section 9.3.3.10 "TModeReg and TPrescalerReg registers" on page 60: register updated • Section 8.5 "Timer unit" on page 31: timer calculation updated • Section 9.3.4.8 "VersionReg register" on page 66: version B2h updated • Section 16.1 "Test signals" on page 82: selftest result updated | | |
| MFRC522_34 | 20100305 | Product data sheet | | MFRC522_33 |
| Modifications: | | <ul style="list-style-type: none"> • Section 8.5 "Timer unit": information added • Table 106 "TModeReg register bit descriptions": bit 7 updated • Table 154 "SPI timing characteristics": row added | | |
| MFRC522_33 | 20091026 | Product data sheet | - | 112132 |

23. Legal information

23.1 Data sheet status

| Document status ^{[1][2]} | Product status ^[3] | Definition |
|-----------------------------------|-------------------------------|---|
| Objective [short] data sheet | Development | This document contains data from the objective specification for product development. |
| Preliminary [short] data sheet | Qualification | This document contains data from the preliminary specification. |
| Product [short] data sheet | Production | This document contains the product specification. |

[1] Please consult the most recently issued document before initiating or completing a design.

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL <http://www.nxp.com>.

23.2 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Short data sheet — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

Product specification — The information and data provided in a Product data sheet shall define the specification of the product as agreed between NXP Semiconductors and its customer, unless NXP Semiconductors and customer have explicitly agreed otherwise in writing. In no event however, shall an agreement be valid in which the NXP Semiconductors product is deemed to offer functions and qualities beyond those described in the Product data sheet.

23.3 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at <http://www.nxp.com/profile/terms>, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Quick reference data — The Quick reference data is an extract of the product data given in the Limiting values and Characteristics sections of this document, and as such is not complete, exhaustive or legally binding.

Non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond

NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

23.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

I²C-bus — logo is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

24. Contact information

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

25. Contents

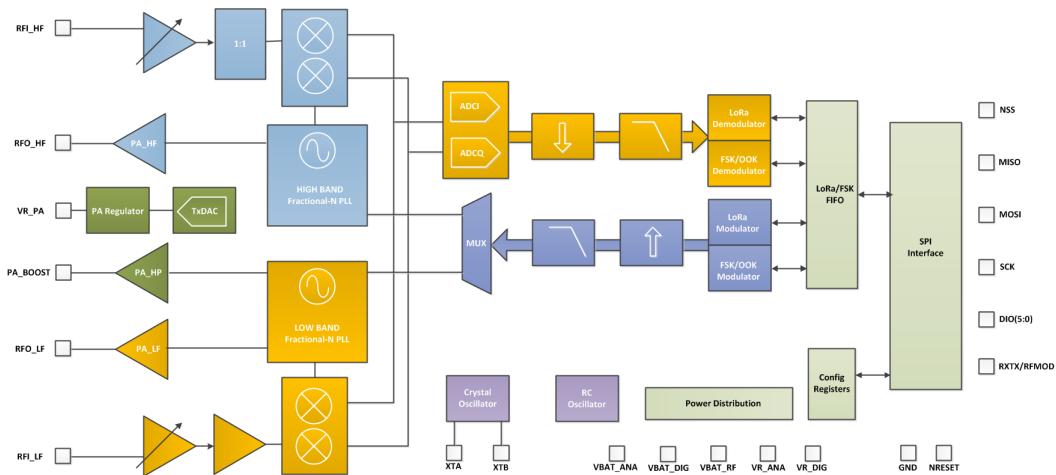
| | | | | | |
|----------|---|----------|----------|--|----|
| 1 | Introduction | 1 | 8.6 | Power reduction modes | 32 |
| 2 | General description | 1 | 8.6.1 | Hard power-down | 32 |
| 2.1 | Differences between version 1.0 and 2.0 | 1 | 8.6.2 | Soft power-down mode | 32 |
| 3 | Features and benefits | 2 | 8.6.3 | Transmitter power-down mode | 32 |
| 4 | Quick reference data | 2 | 8.7 | Oscillator circuit | 32 |
| 5 | Ordering information | 3 | 8.8 | Reset and oscillator start-up time | 33 |
| 6 | Block diagram | 4 | 8.8.1 | Reset timing requirements | 33 |
| 7 | Pinning information | 6 | 8.8.2 | Oscillator start-up time | 33 |
| 7.1 | Pin description | 6 | 9 | MFRC522 registers | 34 |
| 8 | Functional description | 8 | 9.1 | Register bit behavior | 34 |
| 8.1 | Digital interfaces | 9 | 9.2 | Register overview | 35 |
| 8.1.1 | Automatic microcontroller interface detection | 9 | 9.3 | Register descriptions | 37 |
| 8.1.2 | Serial Peripheral Interface | 10 | 9.3.1 | Page 0: Command and status | 37 |
| 8.1.2.1 | SPI read data | 10 | 9.3.1.1 | Reserved register 00h | 37 |
| 8.1.2.2 | SPI write data | 11 | 9.3.1.2 | CommandReg register | 37 |
| 8.1.2.3 | SPI address byte | 11 | 9.3.1.3 | ComlEnReg register | 37 |
| 8.1.3 | UART interface | 11 | 9.3.1.4 | DivlEnReg register | 38 |
| 8.1.3.1 | Connection to a host | 11 | 9.3.1.5 | ComlRqReg register | 38 |
| 8.1.3.2 | Selectable UART transfer speeds | 12 | 9.3.1.6 | DivlRqReg register | 39 |
| 8.1.3.3 | UART framing | 13 | 9.3.1.7 | ErrorReg register | 40 |
| 8.1.4 | I ² C-bus interface | 16 | 9.3.1.8 | Status1Reg register | 41 |
| 8.1.4.1 | Data validity | 17 | 9.3.1.9 | Status2Reg register | 42 |
| 8.1.4.2 | START and STOP conditions | 17 | 9.3.1.10 | FIFODataReg register | 43 |
| 8.1.4.3 | Byte format | 17 | 9.3.1.11 | FIFOLevelReg register | 43 |
| 8.1.4.4 | Acknowledge | 18 | 9.3.1.12 | WaterLevelReg register | 43 |
| 8.1.4.5 | 7-Bit addressing | 19 | 9.3.1.13 | ControlReg register | 44 |
| 8.1.4.6 | Register write access | 19 | 9.3.1.14 | BitFramingReg register | 45 |
| 8.1.4.7 | Register read access | 20 | 9.3.1.15 | CollReg register | 45 |
| 8.1.4.8 | High-speed mode | 21 | 9.3.1.16 | Reserved register 0Fh | 46 |
| 8.1.4.9 | High-speed transfer | 21 | 9.3.2 | Page 1: Communication | 46 |
| 8.1.4.10 | Serial data transfer format in HS mode | 21 | 9.3.2.1 | Reserved register 10h | 46 |
| 8.1.4.11 | Switching between F/S mode and HS mode | 23 | 9.3.2.2 | ModeReg register | 47 |
| 8.1.4.12 | MFRC522 at lower speed modes | 23 | 9.3.2.3 | TxModeReg register | 47 |
| 8.2 | Analog interface and contactless UART | 24 | 9.3.2.4 | RxModeReg register | 48 |
| 8.2.1 | General | 24 | 9.3.2.5 | TxControlReg register | 49 |
| 8.2.2 | TX p-driver | 24 | 9.3.2.6 | TxAASKReg register | 50 |
| 8.2.3 | Serial data switch | 26 | 9.3.2.7 | TxSelReg register | 50 |
| 8.2.4 | MFIN and MFOUT interface support | 26 | 9.3.2.8 | RxSelReg register | 51 |
| 8.2.5 | CRC coprocessor | 28 | 9.3.2.9 | RxThresholdReg register | 52 |
| 8.3 | FIFO buffer | 28 | 9.3.2.10 | DemodReg register | 52 |
| 8.3.1 | Accessing the FIFO buffer | 28 | 9.3.2.11 | Reserved register 1Ah | 53 |
| 8.3.2 | Controlling the FIFO buffer | 28 | 9.3.2.12 | Reserved register 1Bh | 53 |
| 8.3.3 | FIFO buffer status information | 28 | 9.3.2.13 | MfTxReg register | 53 |
| 8.4 | Interrupt request system | 29 | 9.3.2.14 | MfRxReg register | 54 |
| 8.4.1 | Interrupt sources overview | 29 | 9.3.2.15 | Reserved register 1Eh | 54 |
| 8.5 | Timer unit | 30 | 9.3.2.16 | SerialSpeedReg register | 54 |
| | | | 9.3.3 | Page 2: Configuration | 56 |
| | | | 9.3.3.1 | Reserved register 20h | 56 |

continued >

| | | | | | |
|-----------|---|-----------|----------|--|----|
| 9.3.3.2 | CRCResultReg registers | 56 | 16.1.2 | Test bus | 81 |
| 9.3.3.3 | Reserved register 23h | 57 | 16.1.3 | Test signals on pins AUX1 or AUX2 | 82 |
| 9.3.3.4 | ModWidthReg register | 57 | 16.1.3.1 | Example: Output test signals TestDAC1 and TestDAC2 | 83 |
| 9.3.3.5 | Reserved register 25h | 57 | 16.1.3.2 | Example: Output test signals Corr1 and MinLevel | 83 |
| 9.3.3.6 | RFCfgReg register | 58 | 16.1.3.3 | Example: Output test signals ADC channel I and ADC channel Q | 84 |
| 9.3.3.7 | GsNReg register | 58 | 16.1.3.4 | Example: Output test signals RxActive and TxActive | 85 |
| 9.3.3.8 | CWGsPReg register | 59 | 16.1.3.5 | Example: Output test signal RX data stream | 86 |
| 9.3.3.9 | ModGsPReg register | 59 | 16.1.3.6 | PRBS | 86 |
| 9.3.3.10 | TModeReg and TPrescalerReg registers | 59 | 17 | Package outline | 87 |
| 9.3.3.11 | TReloadReg register | 61 | 18 | Handling information | 88 |
| 9.3.3.12 | TCounterValReg register | 61 | 19 | Packing information | 88 |
| 9.3.4 | Page 3: Test | 62 | 20 | Abbreviations | 89 |
| 9.3.4.1 | Reserved register 30h | 62 | 21 | References | 89 |
| 9.3.4.2 | TestSel1Reg register | 62 | 22 | Revision history | 90 |
| 9.3.4.3 | TestSel2Reg register | 63 | 23 | Legal information | 91 |
| 9.3.4.4 | TestPinEnReg register | 63 | 23.1 | Data sheet status | 91 |
| 9.3.4.5 | TestPinValueReg register | 64 | 23.2 | Definitions | 91 |
| 9.3.4.6 | TestBusReg register | 64 | 23.3 | Disclaimers | 91 |
| 9.3.4.7 | AutoTestReg register | 65 | 23.4 | Trademarks | 92 |
| 9.3.4.8 | VersionReg register | 65 | 24 | Contact information | 92 |
| 9.3.4.9 | AnalogTestReg register | 66 | 25 | Contents | 93 |
| 9.3.4.10 | TestDAC1Reg register | 67 | | | |
| 9.3.4.11 | TestDAC2Reg register | 67 | | | |
| 9.3.4.12 | TestADCReg register | 67 | | | |
| 9.3.4.13 | Reserved register 3Ch | 67 | | | |
| 10 | MFRC522 command set | 69 | | | |
| 10.1 | General description | 69 | | | |
| 10.2 | General behavior | 69 | | | |
| 10.3 | MFRC522 command overview | 69 | | | |
| 10.3.1 | MFRC522 command descriptions | 70 | | | |
| 10.3.1.1 | Idle | 70 | | | |
| 10.3.1.2 | Mem | 70 | | | |
| 10.3.1.3 | Generate RandomID | 70 | | | |
| 10.3.1.4 | CalcCRC | 70 | | | |
| 10.3.1.5 | Transmit | 70 | | | |
| 10.3.1.6 | NoCmdChange | 70 | | | |
| 10.3.1.7 | Receive | 71 | | | |
| 10.3.1.8 | Transceive | 71 | | | |
| 10.3.1.9 | MFAuthent | 71 | | | |
| 10.3.1.10 | SoftReset | 72 | | | |
| 11 | Limiting values | 73 | | | |
| 12 | Recommended operating conditions | 73 | | | |
| 13 | Thermal characteristics | 73 | | | |
| 14 | Characteristics | 74 | | | |
| 14.1 | Timing characteristics | 77 | | | |
| 15 | Application information | 80 | | | |
| 16 | Test information | 81 | | | |
| 16.1 | Test signals | 81 | | | |
| 16.1.1 | Self test | 81 | | | |

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver



GENERAL DESCRIPTION

The SX1276/77/78/79 transceivers feature the LoRa™ long range modem that provides ultra-long range spread spectrum communication and high interference immunity whilst minimising current consumption.

Using Semtech's patented LoRa™ modulation technique SX1276/77/78/79 can achieve a sensitivity of over -148dBm using a low cost crystal and bill of materials. The high sensitivity combined with the integrated +20 dBm power amplifier yields industry leading link budget making it optimal for any application requiring range or robustness.

LoRa™ also provides significant advantages in both blocking and selectivity over conventional modulation techniques, solving the traditional design compromise between range, interference immunity and energy consumption.

These devices also support high performance (G)FSK modes for systems including WMBus, IEEE802.15.4g. The SX1276/77/78/79 deliver exceptional phase noise, selectivity, receiver linearity and IIP3 for significantly lower current consumption than competing devices.

ORDERING INFORMATION

| Part Number | Delivery | MOQ / Multiple |
|-------------|----------|----------------|
| SX1276IMLRT | T&R | 3000 pieces |
| SX1277IMLRT | T&R | 3000 pieces |
| SX1278IMLRT | T&R | 3000 pieces |
| SX1279IMLRT | T&R | 3000 pieces |

- ◆ QFN 28 Package - Operating Range [-40;+85°C]
- ◆ Pb-free, Halogen free, RoHS/WEEE compliant product

KEY PRODUCT FEATURES

- ◆ LoRa™ Modem
- ◆ 168 dB maximum link budget
- ◆ +20 dBm - 100 mW constant RF output vs. V supply
- ◆ +14 dBm high efficiency PA
- ◆ Programmable bit rate up to 300 kbps
- ◆ High sensitivity: down to -148 dBm
- ◆ Bullet-proof front end: IIP3 = -11 dBm
- ◆ Excellent blocking immunity
- ◆ Low RX current of 9.9 mA, 200 nA register retention
- ◆ Fully integrated synthesizer with a resolution of 61 Hz
- ◆ FSK, GFSK, MSK, GMSK, LoRa™ and OOK modulation
- ◆ Built-in bit synchronizer for clock recovery
- ◆ Preamble detection
- ◆ 127 dB Dynamic Range RSSI
- ◆ Automatic RF Sense and CAD with ultra-fast AFC
- ◆ Packet engine up to 256 bytes with CRC
- ◆ Built-in temperature sensor and low battery indicator

APPLICATIONS

- ◆ Automated Meter Reading.
- ◆ Home and Building Automation.
- ◆ Wireless Alarm and Security Systems.
- ◆ Industrial Monitoring and Control
- ◆ Long range Irrigation Systems

Table of contents

| Section | Page |
|---|------|
| 1. General Description | 9 |
| 1.1. Simplified Block Diagram | 9 |
| 1.2. Product Versions | 10 |
| 1.3. Pin Diagram | 10 |
| 1.4. Pin Description | 11 |
| 1.5. Package Marking | 12 |
| 2. Electrical Characteristics | 13 |
| 2.1. ESD Notice | 13 |
| 2.2. Absolute Maximum Ratings | 13 |
| 2.3. Operating Range..... | 13 |
| 2.4. Thermal Properties | 13 |
| 2.5. Chip Specification | 14 |
| 2.5.1. Power Consumption | 14 |
| 2.5.2. Frequency Synthesis..... | 14 |
| 2.5.3. FSK/OOK Mode Receiver | 16 |
| 2.5.4. FSK/OOK Mode Transmitter | 17 |
| 2.5.5. Electrical Specification for LoRaTM Modulation | 19 |
| 2.5.6. Digital Specification | 22 |
| 3. SX1276/77/78/79 Features..... | 23 |
| 3.1. LoRaTM Modem | 24 |
| 3.2. FSK/OOK Modem | 24 |
| 4. SX1276/77/78/79 Digital Electronics | 25 |
| 4.1. The LoRaTM Modem | 25 |
| 4.1.1. Link Design Using the LoRaTM Modem | 26 |
| 4.1.2. LoRaTM Digital Interface | 34 |
| 4.1.3. Operation of the LoRaTM Modem..... | 36 |
| 4.1.4. Frequency Settings | 37 |
| 4.1.5. Frequency Error Indication | 37 |
| 4.1.6. LoRaTM Modem State Machine Sequences | 38 |
| 4.1.7. Modem Status Indicators..... | 46 |
| 4.2. FSK/OOK Modem | 46 |
| 4.2.1. Bit Rate Setting | 46 |
| 4.2.2. FSK/OOK Transmission | 47 |
| 4.2.3. FSK/OOK Reception | 48 |
| 4.2.4. Operating Modes in FSK/OOK Mode | 54 |
| 4.2.5. Startup Times | 56 |
| 4.2.6. Receiver Startup Options | 59 |
| 4.2.7. Receiver Restart Methods | 60 |
| 4.2.8. Top Level Sequencer | 61 |

Table of contents

| Section | Page |
|--|------|
| 4.2.9. Data Processing in FSK/OOK Mode | 65 |
| 4.2.10. FIFO | 66 |
| 4.2.11. Digital IO Pins Mapping | 69 |
| 4.2.12. Continuous Mode | 70 |
| 4.2.13. Packet Mode | 71 |
| 4.2.14. io-homecontrol® Compatibility Mode | 79 |
| 4.3. SPI Interface | 80 |
| 5. SX1276/77/78/79 Analog & RF Frontend Electronics..... | 81 |
| 5.1. Power Supply Strategy | 81 |
| 5.2. Low Battery Detector | 81 |
| 5.3. Frequency Synthesis | 81 |
| 5.3.1. Crystal Oscillator | 81 |
| 5.3.2. CLKOUT Output | 82 |
| 5.3.3. PLL | 82 |
| 5.3.4. RC Oscillator | 82 |
| 5.4. Transmitter Description | 83 |
| 5.4.1. Architecture Description | 83 |
| 5.4.2. RF Power Amplifiers | 83 |
| 5.4.3. High Power +20 dBm Operation | 84 |
| 5.4.4. Over Current Protection | 85 |
| 5.5. Receiver Description | 85 |
| 5.5.1. Overview | 85 |
| 5.5.2. Receiver Enabled and Receiver Active States | 85 |
| 5.5.3. Automatic Gain Control In FSK/OOK Mode | 85 |
| 5.5.4. RSSI in FSK/OOK Mode | 86 |
| 5.5.5. RSSI and SNR in LoRaTM Mode | 87 |
| 5.5.6. Channel Filter | 88 |
| 5.5.7. Temperature Measurement | 89 |
| 6. Description of the Registers..... | 90 |
| 6.1. Register Table Summary | 90 |
| 6.2. FSK/OOK Mode Register Map | 93 |
| 6.3. Band Specific Additional Registers | 106 |
| 6.4. LoRaTM Mode Register Map | 108 |
| 7. Application Information | 115 |
| 7.1. Crystal Resonator Specification | 115 |
| 7.2. Reset of the Chip | 115 |
| 7.2.1. POR | 115 |
| 7.2.2. Manual Reset | 116 |
| 7.3. Top Sequencer: Listen Mode Examples | 116 |

Table of contents

| Section | Page |
|--|------|
| 7.3.1. Wake on Preamble Interrupt | 116 |
| 7.3.2. Wake on SyncAddress Interrupt | 119 |
| 7.4. Top Sequencer: Beacon Mode | 122 |
| 7.4.1. Timing diagram..... | 122 |
| 7.4.2. Sequencer Configuration..... | 122 |
| 7.5. Example CRC Calculation | 124 |
| 7.6. Example Temperature Reading | 125 |
| 8. Packaging Information | 127 |
| 8.1. Package Outline Drawing | 127 |
| 8.2. Recommended Land Pattern | 128 |
| 8.3. Tape & Reel Information | 129 |
| 9. Revision History..... | 130 |

Table of contents

| Section | Page |
|---|------|
| Table 1. SX1276/77/78/79 Device Variants and Key Parameters | 10 |
| Table 2. Pin Description | 11 |
| Table 3. Absolute Maximum Ratings | 13 |
| Table 4. Operating Range | 13 |
| Table 5. Thermal Properties | 13 |
| Table 6. Power Consumption Specification | 14 |
| Table 7. Frequency Synthesizer Specification | 14 |
| Table 8. FSK/OOK Receiver Specification | 16 |
| Table 9. Transmitter Specification | 17 |
| Table 10. LoRa Receiver Specification | 19 |
| Table 11. Digital Specification | 22 |
| Table 12. Example LoRaTM Modem Performances, 868MHz Band | 25 |
| Table 13. Range of Spreading Factors | 27 |
| Table 14. Cyclic Coding Overhead | 27 |
| Table 15. LoRa Bandwidth Options | 28 |
| Table 16. LoRaTM Operating Mode Functionality | 36 |
| Table 17. LoRa CAD Consumption Figures | 45 |
| Table 18. DIO Mapping LoRaTM Mode | 46 |
| Table 19. Bit Rate Examples | 47 |
| Table 20. Preamble Detector Settings | 53 |
| Table 21. RxTrigger Settings to Enable Timeout Interrupts | 54 |
| Table 22. Basic Transceiver Modes | 54 |
| Table 23. Receiver Startup Time Summary | 57 |
| Table 24. Receiver Startup Options | 60 |
| Table 25. Sequencer States | 61 |
| Table 26. Sequencer Transition Options | 62 |
| Table 27. Sequencer Timer Settings | 63 |
| Table 28. Status of FIFO when Switching Between Different Modes of the Chip | 67 |
| Table 29. DIO Mapping, Continuous Mode | 69 |
| Table 30. DIO Mapping, Packet Mode | 69 |
| Table 31. CRC Description | 77 |
| Table 32. Frequency Bands | 82 |
| Table 33. Power Amplifier Mode Selection Truth Table | 83 |
| Table 34. High Power Settings | 84 |
| Table 35. Operating Range, +20dBm Operation | 84 |
| Table 36. Operating Range, +20dBm Operation | 84 |
| Table 37. Trimming of the OCP Current | 85 |
| Table 38. LNA Gain Control and Performances | 86 |
| Table 39. RssiSmoothing Options | 86 |

Table of contents

| Section | Page |
|---|------|
| Table 40. Available RxBw Settings | 88 |
| Table 41. Registers Summary | 90 |
| Table 42. Register Map | 93 |
| Table 43. Low Frequency Additional Registers | 106 |
| Table 44. High Frequency Additional Registers | 107 |
| Table 45. Crystal Specification | 115 |
| Table 46. Listen Mode with PreambleDetect Condition Settings | 118 |
| Table 47. Listen Mode with PreambleDetect Condition Recommended DIO Mapping | 118 |
| Table 48. Listen Mode with SyncAddress Condition Settings | 121 |
| Table 49. Listen Mode with PreambleDetect Condition Recommended DIO Mapping | 121 |
| Table 50. Beacon Mode Settings | 123 |
| Table 51. Revision History | 130 |

Table of contents

| Section | Page |
|--|------|
| Figure 1. Block Diagram | 9 |
| Figure 2. Pin Diagrams | 10 |
| Figure 3. Marking Diagram | 12 |
| Figure 4. SX1276/77/78/79 Block Schematic Diagram | 23 |
| Figure 5. LoRaTM Modem Connectivity | 26 |
| Figure 6. LoRaTM Packet Structure | 29 |
| Figure 7. Interrupts Generated in the Case of Successful Frequency Hopping Communication. | 33 |
| Figure 8. LoRaTM Data Buffer | 34 |
| Figure 9. LoRaTM Modulation Transmission Sequence. | 38 |
| Figure 10. LoRaTM Receive Sequence. | 39 |
| Figure 11. LoRaTM CAD Flow | 43 |
| Figure 12. CAD Time as a Function of Spreading Factor | 44 |
| Figure 13. Consumption Profile of the LoRa CAD Process | 45 |
| Figure 14. OOK Peak Demodulator Description | 49 |
| Figure 15. Floor Threshold Optimization | 50 |
| Figure 16. Bit Synchronizer Description | 51 |
| Figure 17. Startup Process | 56 |
| Figure 18. Time to RSSI Sample | 57 |
| Figure 19. Tx to Rx Turnaround | 58 |
| Figure 20. Rx to Tx Turnaround | 58 |
| Figure 21. Receiver Hopping | 59 |
| Figure 22. Transmitter Hopping | 59 |
| Figure 23. Timer1 and Timer2 Mechanism | 63 |
| Figure 24. Sequencer State Machine | 64 |
| Figure 25. SX1276/77/78/79 Data Processing Conceptual View | 65 |
| Figure 26. FIFO and Shift Register (SR) | 66 |
| Figure 27. FifoLevel IRQ Source Behavior | 67 |
| Figure 28. Sync Word Recognition | 68 |
| Figure 29. Continuous Mode Conceptual View | 70 |
| Figure 30. Tx Processing in Continuous Mode | 70 |
| Figure 31. Rx Processing in Continuous Mode | 71 |
| Figure 32. Packet Mode Conceptual View | 72 |
| Figure 33. Fixed Length Packet Format | 73 |
| Figure 34. Variable Length Packet Format | 74 |
| Figure 35. Unlimited Length Packet Format | 74 |
| Figure 36. Manchester Encoding/Decoding | 78 |
| Figure 37. Data Whitening Polynomial | 79 |
| Figure 38. SPI Timing Diagram (single access) | 80 |
| Figure 39. TCXO Connection | 81 |

Table of contents

| Section | Page |
|---|------|
| Figure 40. RF Front-end Architecture Shows the Internal PA Configuration. | 83 |
| Figure 41. Temperature Sensor Response | 89 |
| Figure 42. POR Timing Diagram | 115 |
| Figure 43. Manual Reset Timing Diagram | 116 |
| Figure 44. Listen Mode: Principle | 116 |
| Figure 45. Listen Mode with No Preamble Received | 117 |
| Figure 46. Listen Mode with Preamble Received | 117 |
| Figure 47. Wake On PreambleDetect State Machine | 118 |
| Figure 48. Listen Mode with no SyncAddress Detected | 119 |
| Figure 49. Listen Mode with Preamble Received and no SyncAddress | 119 |
| Figure 50. Listen Mode with Preamble Received & Valid SyncAddress | 120 |
| Figure 51. Wake On SyncAddress State Machine | 120 |
| Figure 52. Beacon Mode Timing Diagram | 122 |
| Figure 53. Beacon Mode State Machine | 122 |
| Figure 54. Example CRC Code | 124 |
| Figure 55. Example Temperature Reading | 125 |
| Figure 56. Example Temperature Reading (continued) | 126 |
| Figure 57. Package Outline Drawing | 127 |
| Figure 58. Recommended Land Pattern | 128 |
| Figure 59. Tape and Reel Information | 129 |

1. General Description

The SX1276/77/78/79 incorporates the LoRaTM spread spectrum modem which is capable of achieving significantly longer range than existing systems based on FSK or OOK modulation. At maximum data rates of LoRaTM the sensitivity is 8dB better than FSK, but using a low cost bill of materials with a 20ppm XTAL LoRaTM can improve receiver sensitivity by more than 20dB compared to FSK. LoRaTM also provides significant advances in selectivity and blocking performance, further improving communication reliability. For maximum flexibility the user may decide on the spread spectrum modulation bandwidth (BW), spreading factor (SF) and error correction rate (CR). Another benefit of the spread modulation is that each spreading factor is orthogonal - thus multiple transmitted signals can occupy the same channel without interfering. This also permits simple coexistence with existing FSK based systems. Standard GFSK, FSK, OOK, and GMSK modulation is also provided to allow compatibility with existing systems or standards such as wireless MBUS and IEEE 802.15.4g.

The SX1276 and SX1279 offer bandwidth options ranging from 7.8 kHz to 500 kHz with spreading factors ranging from 6 to 12, and covering all available frequency bands. The SX1277 offers the same bandwidth and frequency band options with spreading factors from 6 to 9. The SX1278 offers bandwidths and spreading factor options, but only covers the lower UHF bands.

1.1. Simplified Block Diagram

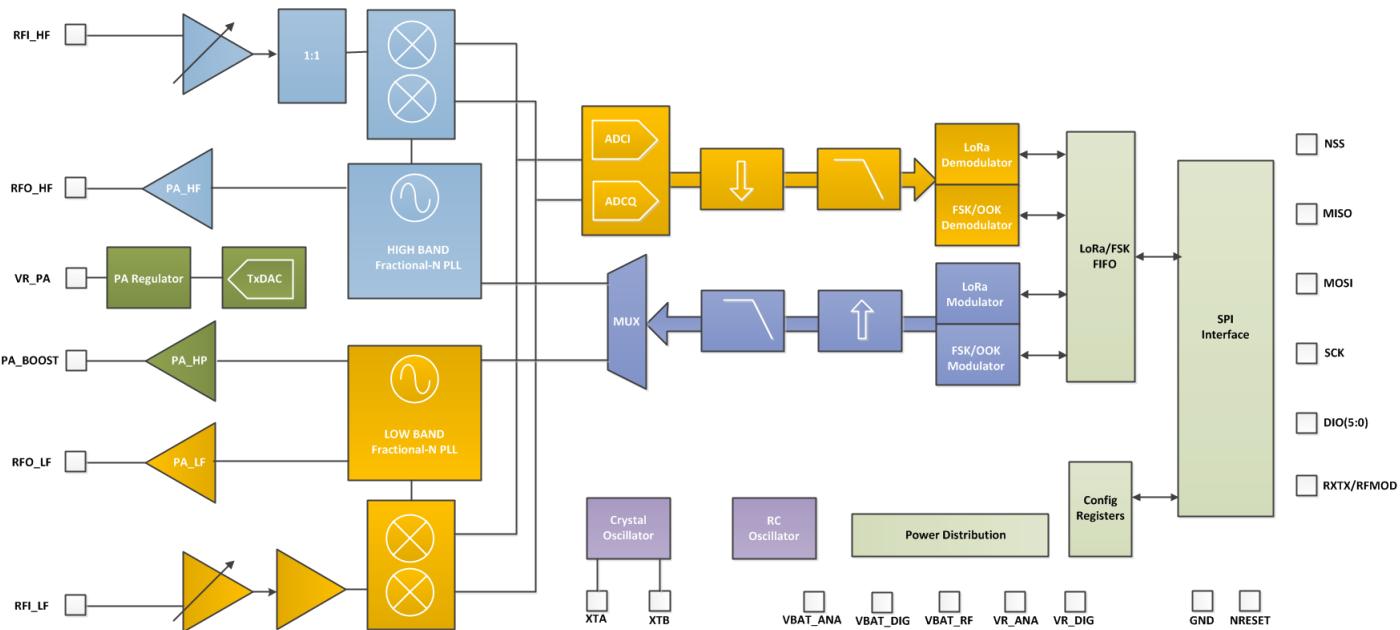


Figure 1. Block Diagram

1.2. Product Versions

The features of the four product variants are detailed in the following table.

Table 1 SX1276/77/78/79 Device Variants and Key Parameters

| Part Number | Frequency Range | Spreading Factor | Bandwidth | Effective Bitrate | Est. Sensitivity |
|-------------|-----------------|------------------|---------------|-------------------|------------------|
| SX1276 | 137 - 1020 MHz | 6 - 12 | 7.8 - 500 kHz | .018 - 37.5 kbps | -111 to -148 dBm |
| SX1277 | 137 - 1020 MHz | 6 - 9 | 7.8 - 500 kHz | 0.11 - 37.5 kbps | -111 to -139 dBm |
| SX1278 | 137 - 525 MHz | 6- 12 | 7.8 - 500 kHz | .018 - 37.5 kbps | -111 to -148 dBm |
| SX1279 | 137 - 960MHz | 6- 12 | 7.8 - 500 kHz | .018 - 37.5 kbps | -111 to -148 dBm |

1.3. Pin Diagram

The following diagram shows the pin arrangement of the QFN package, top view.

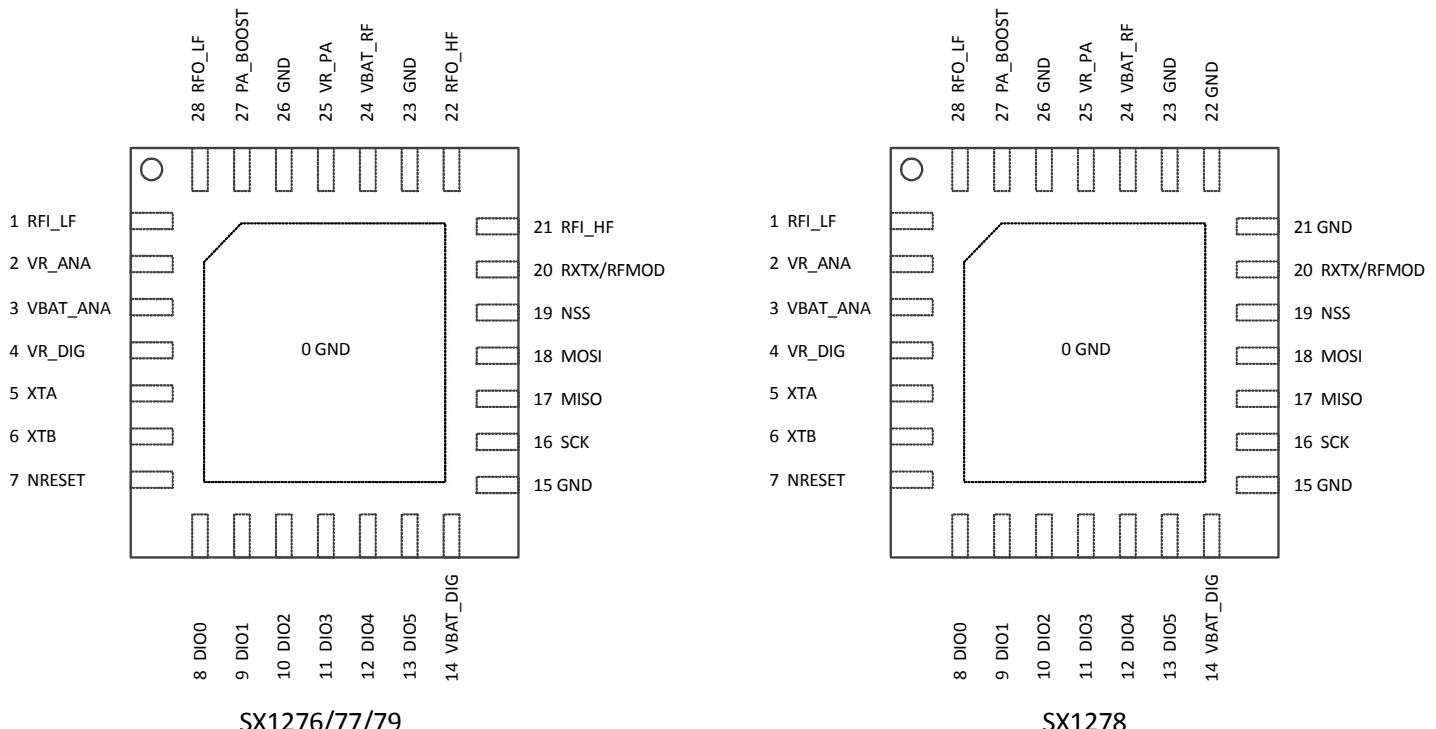


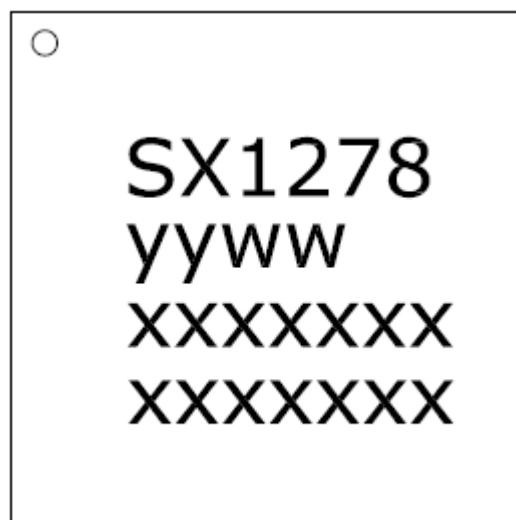
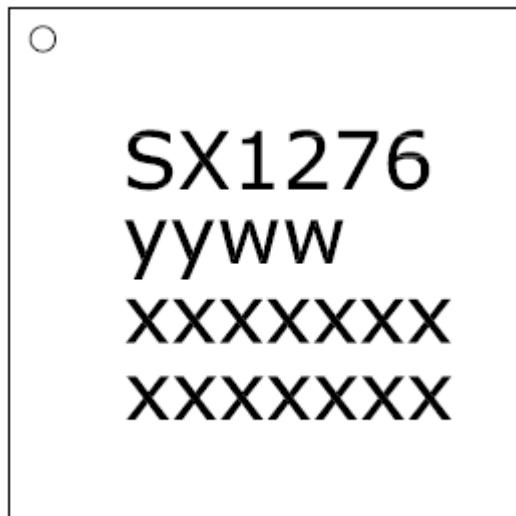
Figure 2. Pin Diagrams

1.4. Pin Description

Table 2 Pin Description

| Number | Name | Type | Description |
|--------|-------------------|-------------------|--|
| | SX1276/77/79/(78) | SX1276/77/79/(78) | SX1276/77/79/(78) |
| 0 | GROUND | - | Exposed ground pad |
| 1 | RFI_LF | I | RF input for bands 2&3 |
| 2 | VR_ANA | - | Regulated supply voltage for analogue circuitry |
| 3 | VBAT_ANA | - | Supply voltage for analogue circuitry |
| 4 | VR_DIG | - | Regulated supply voltage for digital blocks |
| 5 | XTA | I/O | XTAL connection or TCXO input |
| 6 | XTB | I/O | XTAL connection |
| 7 | NRESET | I/O | Reset trigger input |
| 8 | DIO0 | I/O | Digital I/O, software configured |
| 9 | DIO1/DCLK | I/O | Digital I/O, software configured |
| 10 | DIO2/DATA | I/O | Digital I/O, software configured |
| 11 | DIO3 | I/O | Digital I/O, software configured |
| 12 | DIO4 | I/O | Digital I/O, software configured |
| 13 | DIO5 | I/O | Digital I/O, software configured |
| 14 | VBAT_DIG | - | Supply voltage for digital blocks |
| 15 | GND | - | Ground |
| 16 | SCK | I | SPI Clock input |
| 17 | MISO | O | SPI Data output |
| 18 | MOSI | I | SPI Data input |
| 19 | NSS | I | SPI Chip select input |
| 20 | RXTX/RF_MOD | O | Rx/Tx switch control: high in Tx |
| 21 | RFI_HF (GND) | I (-) | RF input for band 1 (Ground) |
| 22 | RFO_HF (GND) | O (-) | RF output for band 1 (Ground) |
| 23 | GND | - | Ground |
| 24 | VBAT_RF | - | Supply voltage for RF blocks |
| 25 | VR_PA | - | Regulated supply for the PA |
| 26 | GND | - | Ground |
| 27 | PA_BOOST | O | Optional high-power PA output, all frequency bands |
| 28 | RFO_LF | O | RF output for bands 2&3 |

1.5. Package Marking



| TOP MARK | |
|-----------|------|
| CHAR | ROWS |
| 7/7/7/7/7 | 5 |

Marking for the 6 x 6 mm MLPQ 28Id Lead package:

nnnnnn = Part Number (Example: SX1276)
yyWW = Date Code (Example: 1352)
xxxxxx = Semtech Lot No. (Example: EA90101)
 0101-10)

Figure 3. Marking Diagram

2. Electrical Characteristics

2.1. ESD Notice

The SX1276/77/78/79 is a high performance radio frequency device. It satisfies:

- ◆ Class 2 of the JEDEC standard JESD22-A114 (Human Body Model) on all pins.
- ◆ Class III of the JEDEC standard JESD22-C101 (Charged Device Model) on all pins



It should thus be handled with all the necessary ESD precautions to avoid any permanent damage.

2.2. Absolute Maximum Ratings

Stresses above the values listed below may cause permanent device failure. Exposure to absolute maximum ratings for extended periods may affect device reliability.

Table 3 Absolute Maximum Ratings

| Symbol | Description | Min | Max | Unit |
|--------|----------------------|------|------|------|
| VDDmr | Supply Voltage | -0.5 | 3.9 | V |
| Tmr | Temperature | -55 | +115 | °C |
| Tj | Junction temperature | - | +125 | °C |
| Pmr | RF Input Level | - | +10 | dBm |

Note Specific ratings apply to +20 dBm operation (see Section 5.4.3).

2.3. Operating Range

Table 4 Operating Range

| Symbol | Description | Min | Max | Unit |
|--------|-----------------------------------|-----|-----|------|
| VDDop | Supply voltage | 1.8 | 3.7 | V |
| Top | Operational temperature range | -40 | +85 | °C |
| Clop | Load capacitance on digital ports | - | 25 | pF |
| ML | RF Input Level | - | +10 | dBm |

Note A specific supply voltage range applies to +20 dBm operation (see Section 5.4.3).

2.4. Thermal Properties

Table 5 Thermal Properties

| Symbol | Description | Min | Typ | Max | Unit |
|----------|--|-----|--------|-----|------|
| THETA_JA | Package θ_{ja} (Junction to ambient) | - | 22.185 | - | °C/W |
| THETA_JC | Package θ_{jc} (Junction to case ground paddle) | - | 0.757 | - | °C/W |

2.5. Chip Specification

The tables below give the electrical specifications of the transceiver under the following conditions: Supply voltage VDD=3.3 V, temperature = 25 °C, FXOSC = 32 MHz, F_{RF} = 169/434/868/915 MHz (see specific indication), Pout = +13dBm, 2-level FSK modulation without pre-filtering, FDA = 5 kHz, Bit Rate = 4.8 kb/s and terminated in a matched 50 Ohm impedance, shared Rx and Tx path matching, unless otherwise specified.

Note Specification whose symbol is appended with “_LF” corresponds to the performance in Band 2 and/or Band 3, as described in section 5.3.3. “_HF” refers to the upper Band 1

2.5.1. Power Consumption

Table 6 Power Consumption Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|---------|---|--|------------------|-----------------------|-----|----------------------|
| IDDSL | Supply current in Sleep mode | | - | 0.2 | 1 | uA |
| IDDIDLE | Supply current in Idle mode | RC oscillator enabled | - | 1.5 | - | uA |
| IDDST | Supply current in Standby mode | Crystal oscillator enabled | - | 1.6 | 1.8 | mA |
| IDDFS | Supply current in Synthesizer mode | FSRx | - | 5.8 | - | mA |
| IDDR | Supply current in Receive mode | <i>LnaBoost</i> Off, band 1 <i>LnaBoost</i> On, band 1 Bands 2&3 | - - - | 10.8 11.5 12.0 | - | mA |
| IDDT | Supply current in Transmit mode with impedance matching | RFOP = +20 dBm, on PA_BOOST RFOP = +17 dBm, on PA_BOOST RFOP = +13 dBm, on RFO_LF/HF pin RFOP = + 7 dBm, on RFO_LF/HF pin | - - - - | 120 87 29 20 | - | mA mA mA mA |

2.5.2. Frequency Synthesis

Table 7 Frequency Synthesizer Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|--------|---|---|--------------------------|-------------|---|------|
| FR | Synthesizer frequency range | Programmable (*for SX1279) Band 3 Band 2 Band 1 | 137 410 862 (*779) | - - - | 175 (*160) 525 (*480) 1020 (*960) | MHz |
| FXOSC | Crystal oscillator frequency | | - | 32 | - | MHz |
| TS_OSC | Crystal oscillator wake-up time | | - | 250 | - | us |
| TS_FS | Frequency synthesizer wake-up time to PLLock signal | From Standby mode | - | 60 | - | us |

| | | | | | | |
|--------|--|--|-------|------|--------|------|
| TS_HOP | Frequency synthesizer hop time at most 10 kHz away from the target frequency | 200 kHz step | - | 20 | - | us |
| | | 1 MHz step | - | 20 | - | us |
| | | 5 MHz step | - | 50 | - | us |
| | | 7 MHz step | - | 50 | - | us |
| | | 12 MHz step | - | 50 | - | us |
| | | 20 MHz step | - | 50 | - | us |
| | | 25 MHz step | - | 50 | - | us |
| | | | | | | |
| FSTEP | Frequency synthesizer step | FSTEP = FXOSC/2 ¹⁹ | - | 61.0 | - | Hz |
| FRC | RC Oscillator frequency | After calibration | - | 62.5 | - | kHz |
| BRF | Bit rate, FSK | Programmable values (1) | 1.2 | - | 300 | kbps |
| BRA | Bit rate Accuracy, FSK | ABS(wanted BR - available BR) | - | - | 250 | ppm |
| BRO | Bit rate, OOK | Programmable | 1.2 | - | 32.768 | kbps |
| BR_L | Bit rate, LoRa Mode | From SF6, BW=500kHz to SF12, BW=7.8kHz | 0.018 | - | 37.5 | kbps |
| FDA | Frequency deviation, FSK (1) | Programmable FDA + BRF/2 <= 250 kHz | 0.6 | - | 200 | kHz |

Note: For Maximum Bit rate, the maximum modulation index is 0.5.

2.5.3. FSK/OOK Mode Receiver

All receiver tests are performed with RxBw = 10 kHz (Single Side Bandwidth) as programmed in *RegRxBw*, receiving a PN15 sequence. Sensitivities are reported for a 0.1% BER (with Bit Synchronizer enabled), unless otherwise specified. Blocking tests are performed with an unmodulated interferer. The wanted signal power for the Blocking Immunity, ACR, IIP2, IIP3 and AMR tests is set 3 dB above the receiver sensitivity level.

Table 8 FSK/OOK Receiver Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|----------|--|---|-----------------------|-------------------------------------|-----------------------|---------------------------------|
| RFS_F_LF | Direct tie of RFI and RFO pins, shared Rx, Tx paths FSK sensitivity, highest LNA gain. Bands 2&3 | FDA = 5 kHz, BR = 1.2 kb/s FDA = 5 kHz, BR = 4.8 kb/s FDA = 40 kHz, BR = 38.4 kb/s* FDA = 20 kHz, BR = 38.4 kb/s** FDA = 62.5 kHz, BR = 250 kb/s*** | - - - - - | -121 -117 -107 -108 -95 | - - - - - | dBm dBm dBm dBm dBm |
| | Split RF paths, the RF switch insertion loss is not accounted for. Bands 2&3 | FDA = 5 kHz, BR = 1.2 kb/s FDA = 5 kHz, BR = 4.8 kb/s FDA = 40 kHz, BR = 38.4 kb/s* FDA = 20 kHz, BR = 38.4 kb/s** FDA = 62.5 kHz, BR = 250 kb/s*** | - - - - - | -123 -119 -109 -110 -97 | - - - - - | dBm dBm dBm dBm dBm |
| RFS_F_HF | Direct tie of RFI and RFO pins, shared Rx, Tx paths FSK sensitivity, highest LNA gain. Band 1 | FDA = 5 kHz, BR = 1.2 kb/s FDA = 5 kHz, BR = 4.8 kb/s FDA = 40 kHz, BR = 38.4 kb/s* FDA = 20 kHz, BR = 38.4 kb/s** FDA = 62.5 kHz, BR = 250 kb/s*** | - - - - - | -119 -115 -105 -105 -92 | - - - - - | dBm dBm dBm dBm dBm |
| | Split RF paths, <i>LnaBoost</i> is turned on, the RF switch insertion loss is not accounted for. Band 1 | FDA = 5 kHz, BR = 1.2 kb/s FDA = 5 kHz, BR = 4.8 kb/s FDA = 40 kHz, BR = 38.4 kb/s* FDA = 20 kHz, BR = 38.4 kb/s** FDA = 62.5 kHz, BR = 250 kb/s*** | - - - - - | -123 -119 -109 -109 -96 | - - - - - | dBm dBm dBm dBm dBm |
| RFS_O | OOK sensitivity, highest LNA gain shared Rx, Tx paths | BR = 4.8 kb/s BR = 32 kb/s | - - | -117 -108 | - - | dBm dBm |
| CCR | Co-Channel Rejection, FSK | | - | -9 | - | dB |
| ACR | Adjacent Channel Rejection | FDA = 5 kHz, BR=4.8kb/s Offset = +/- 25 kHz or +/- 50kHz Band 1 Band 2 Band 3 | - - - | 50 56 60 | - - - | dB dB dB |
| BI_HF | Blocking Immunity, Band 1 | Offset = +/- 1 MHz Offset = +/- 2 MHz Offset = +/- 10 MHz | - - - | 71 76 84 | - - - | dB dB dB |
| BI_LF | Blocking Immunity, Bands 2&3 | Offset = +/- 1 MHz Offset = +/- 2 MHz Offset = +/- 10 MHz | - - - | 71 72 78 | - - - | dB dB dB |

| | | | | | | |
|---------|---|---|------------|-----------|-----|------------|
| IIP2 | 2nd order Input Intercept Point Unwanted tones are 20 MHz above the LO | Highest LNA gain | - | +55 | - | dBm |
| IIP3_HF | 3rd order Input Intercept point Unwanted tones are 1MHz and 1.995 MHz above the LO | Band 1 Highest LNA gain G1 LNA gain G2, 5dB sensitivity hit | - | -11 | - | dBm dBm |
| IIP3_LF | 3rd order Input Intercept point Unwanted tones are 1MHz and 1.995 MHz above the LO | Band 2 Highest LNA gain G1 LNA gain G2, 2.5dB sensitivity hit | - | -22 | - | dBm dBm |
| | | Band 3 Highest LNA gain G1 LNA gain G2, 2.5dB sensitivity hit | - | -15 | - | dBm dBm |
| BW_SSB | Single Side channel filter BW | Programmable | 2.7 | - | 250 | kHz |
| IMR | Image Rejection | Wanted signal 3dB over sensitivity BER=0.1% | - | 50 | - | dB |
| IMA | Image Attenuation | | - | 57 | - | dB |
| DR_RSSI | RSSI Dynamic Range | AGC enabled | Min Max | -127 0 | - | dBm dBm |

* RxBw = 83 kHz (Single Side Bandwidth)

** RxBw = 50 kHz (Single Side Bandwidth)

*** RxBw = 250 kHz (Single Side Bandwidth)

2.5.4. FSK/OOK Mode Transmitter

Table 9 Transmitter Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|------------|--|---|--------|-----------|--------|------------|
| RF_OP | RF output power in 50 ohms on RFO pin (High efficiency PA). | Programmable with steps Max Min | - - | +14 -1 | - - | dBm dBm |
| ΔRF_OP_V | RF output power stability on RFO pin versus voltage supply. | VDD = 2.5 V to 3.3 V VDD = 1.8 V to 3.7 V | - - | 3 8 | - - | dB dB |
| RF_OPH | RF output power in 50 ohms, on PA_BOOST pin (Regulated PA). | Programmable with 1dB steps Max Min | - - | +17 +2 | - - | dBm dBm |
| RF_OPH_MAX | Max RF output power, on PA_BOOST pin | High power mode | - | +20 | - | dBm |
| ΔRF_OPH_V | RF output power stability on PA_BOOST pin versus voltage supply. | VDD = 2.4 V to 3.7 V | - | +/-1 | - | dB |
| ΔRF_T | RF output power stability versus temperature on PA_BOOST pin. | From T = -40 °C to +85 °C | - | +/-1 | - | dB |

| | | | | | | |
|-------|--|---|---|------|-----|------------|
| PHN | Transmitter Phase Noise | 169 MHz, Band 3 | | | | |
| | | 10kHz Offset | - | -118 | - | dBc/ Hz |
| | | 50kHz Offset | - | -118 | - | |
| | | 400kHz Offset | - | -128 | - | |
| | | 1MHz Offset | - | -134 | - | |
| | | 433 MHz, Band 2 | | | | |
| | | 10kHz Offset | - | -110 | - | dBc/ Hz |
| | | 50kHz Offset | - | -110 | - | |
| | | 400kHz Offset | - | -122 | - | |
| | | 1MHz Offset | - | -129 | - | |
| | | 868/915 MHz, Band 1 | | | | |
| | | 10kHz Offset | - | -103 | - | dBc/ Hz |
| | | 50kHz Offset | - | -103 | - | |
| | | 400kHz Offset | - | -115 | - | |
| | | 1MHz Offset | - | -122 | - | |
| ACP | Transmitter adjacent channel power (measured at 25 kHz offset) | BT=1. Measurement conditions as defined by EN 300 220-1 V2.3.1 | - | - | -37 | dBm |
| TS_TR | Transmitter wake up time, to the first rising edge of DCLK | Frequency Synthesizer enabled, $\text{PaR-amp} = 10\text{us}$, BR = 4.8 kb/s | - | 120 | - | us |

2.5.5. Electrical Specification for LoRaTM Modulation

The table below gives the electrical specifications for the transceiver operating with LoRaTM modulation. Following conditions apply unless otherwise specified:

- ◆ Supply voltage = 3.3 V
- ◆ Temperature = 25° C
- ◆ f_{XOSC} = 32 MHz
- ◆ bandwidth (BW) = 125 kHz
- ◆ Spreading Factor (SF) = 12
- ◆ Error Correction Code (EC) = 4/6
- ◆ Packet Error Rate (PER)= 1%
- ◆ CRC on payload enabled
- ◆ Output power = 13 dBm in transmission
- ◆ Payload length = 64 bytes
- ◆ Preamble Length = 12 symbols (programmed register *PreambleLength*=8)
- ◆ With matched impedances

Table 10 LoRa Receiver Specification

| Symbol | Description | Conditions | Min. | Typ | Max | Unit |
|-----------|---|---|------|-----------------|-----|----------------|
| IDDR_L | Supply current in receiver LoRa TM mode, <i>LnaBoost</i> off | Bands 2&3, BW=7.8 to 62.5 kHz | - | 11.0 | - | mA |
| | | Bands 2&3, BW = 125 kHz | - | 11.5 | - | mA |
| | | Bands 2&3, BW = 250 kHz | - | 12.4 | - | mA |
| | | Bands 2&3, BW = 500 kHz | - | 13.8 | - | mA |
| | | Band 1, BW=7.8 to 62.5 kHz | - | 9.9 | - | mA |
| | | Band 1, BW = 125 kHz | - | 10.3 | - | mA |
| | | Band 1, BW = 250 kHz | - | 11.1 | - | mA |
| | | Band 1, BW = 500 kHz | - | 12.6 | - | mA |
| IDDT_L | Supply current in transmitter mode | RFOP = 13 dBm RFOP = 7 dBm | - | 28 20 | - | mA mA |
| IDDT_H_L | Supply current in transmitter mode with an external impedance transformation | Using PA_BOOST pin RFOP = 17 dBm | - | 90 | - | mA |
| BI_L | Blocking immunity, CW interferer | offset = +/- 1 MHz offset = +/- 2 MHz offset = +/- 10 MHz | - | 89 94 100 | - | dB dB dB |
| IIP2_L | 2nd order Input Intercept Point Unwanted tones are 20 MHz above the LO | Highest LNA gain | - | +55 | - | dBm |
| IIP3_L_HF | 3rd order Input Intercept point Unwanted tones are 1MHz and 1.995 MHz above the LO | Band 1 Highest LNA gain G1 LNA gain G2, 5dB sensitivity hit | - | -11 -6 | - | dBm dBm |

| Symbol | Description | Conditions | Min. | Typ | Max | Unit |
|-------------|--|---|------|--|-----|---|
| IIP3_L_LF | 3rd order Input Intercept point Unwanted tones are 1MHz and 1.995 MHz above the LO | Band 2 Highest LNA gain G1 LNA gain G2, 2.5dB sensitivity hit | - | -22 -15 | - | dBm dBm |
| RFS_L10_HF | RF sensitivity, Long-Range Mode, highest LNA gain, <i>LnaBoost</i> for Band 1, using split Rx/Tx path 10.4 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 11 | - | -131 -134 -138 -146 | - | dBm dBm dBm dBm |
| RFS_L62_HF | RF sensitivity, Long-Range Mode, highest LNA gain, <i>LnaBoost</i> for Band 1, using split Rx/Tx path 62.5 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - | -121 -126 -129 -132 -135 -137 -139 | - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L125_HF | RF sensitivity, Long-Range Mode, highest LNA gain, <i>LnaBoost</i> for Band 1, using split Rx/Tx path 125 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - | -118 -123 -126 -129 -132 -133 -136 | - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L250_HF | RF sensitivity, Long-Range Mode, highest LNA gain, <i>LnaBoost</i> for Band 1, using split Rx/Tx path 250 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - | -115 -120 -123 -125 -128 -130 -133 | - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L500_HF | RF sensitivity, Long-Range Mode, highest LNA gain, <i>LnaBoost</i> for Band 1, using split Rx/Tx path 500 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - | -111 -116 -119 -122 -125 -128 -130 | - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L7.8_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 2 or 3, using split Rx/Tx path 7.8 kHz bandwidth | SF = 12 SF = 11 | - | -148 -145 | - | dBm dBm |
| RFS_L10_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 3, 10.4 kHz bandwidth | SF = 6 SF = 7 SF = 8 | - | -132 -136 -138 | - | dBm dBm dBm |
| RFS_L62_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 3, 62.5 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - | -123 -128 -131 -134 -135 -137 -140 | - | dBm dBm dBm dBm dBm dBm dBm |

| Symbol | Description | Conditions | Min. | Typ | Max | Unit |
|-------------|---|---|---------------------------------|--|---------------------------------|---|
| RFS_L125_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 3, 125 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - - - - - - - | -121 -125 -128 -131 -134 -136 -137 | - - - - - - - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L250_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 3 250 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - - - - - - - | -118 -122 -125 -128 -131 -133 -134 | - - - - - - - | dBm dBm dBm dBm dBm dBm dBm |
| RFS_L500_LF | RF sensitivity, Long-Range Mode, highest LNA gain, Band 3 500 kHz bandwidth | SF = 6 SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - - - - - - - | -112 -118 -121 -124 -127 -129 -130 | - - - - - - - | dBm dBm dBm dBm dBm dBm dBm |
| CCR_LCW | Co-channel rejection Single CW tone = Sens +6 dB 1% PER | SF = 7 SF = 8 SF = 9 SF = 10 SF = 11 SF = 12 | - - - - - - | 5 9.5 12 14.4 17 19.5 | - - - - - - | dB dB dB dB dB dB |
| CCR_LL | Co-channel rejection | Interferer is a LoRa™ signal using same BW and same SF. Pw = Sensitivity + 3 dB | | -6 | | dB |
| ACR_LCW | Adjacent channel rejection | Interferer is 1.5*BW_L from the wanted signal center frequency 1% PER, Single CW tone = Sens + 3 dB SF = 7 SF = 12 | | | | |
| IMR_LCW | Image rejection after calibration. | 1% PER, Single CW tone = Sens +3 dB | - | 66 | - | dB |
| FERR_L | Maximum tolerated frequency offset between transmitter and receiver, no sensitivity degradation, SF6 thru 12 | All BW, +/-25% of BW The tighter limit applies (see below) | | +/-25% | | BW |
| | Maximum tolerated frequency offset between transmitter and receiver, no sensitivity degradation, SF10 thru 12 | SF = 12 SF = 11 SF = 10 | -50 -100 -200 | - - - | 50 100 200 | ppm ppm ppm |

2.5.6. Digital Specification

Conditions: Temp = 25° C, VDD = 3.3 V, FXOSC = 32 MHz, unless otherwise specified.

Table 11 Digital Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|---------------------|------------------------------------|--|-----|-----|-----|------|
| V _{IH} | Digital input level high | | 0.8 | - | - | VDD |
| V _{IL} | Digital input level low | | - | - | 0.2 | VDD |
| V _{OH} | Digital output level high | I _{max} = 1 mA | 0.9 | - | - | VDD |
| V _{OL} | Digital output level low | I _{max} = -1 mA | - | - | 0.1 | VDD |
| F _{SCK} | SCK frequency | | - | - | 10 | MHz |
| t _{ch} | SCK high time | | 50 | - | - | ns |
| t _{cl} | SCK low time | | 50 | - | - | ns |
| t _{rise} | SCK rise time | | - | 5 | - | ns |
| t _{fall} | SCK fall time | | - | 5 | - | ns |
| t _{setup} | MOSI setup time | From MOSI change to SCK rising edge. | 30 | - | - | ns |
| t _{hold} | MOSI hold time | From SCK rising edge to MOSI change. | 20 | - | - | ns |
| t _{nsetup} | NSS setup time | From NSS falling edge to SCK rising edge. | 30 | - | - | ns |
| t _{nhold} | NSS hold time | From SCK falling edge to NSS rising edge, normal mode. | 100 | - | - | ns |
| t _{nhigh} | NSS high time between SPI accesses | | 20 | - | - | ns |
| T _{_DATA} | DATA hold and setup time | | 250 | - | - | ns |

3. SX1276/77/78/79 Features

This section gives a high-level overview of the functionality of the SX1276/77/78/79 low-power, highly integrated transceiver. The following figure shows a simplified block diagram of the SX1276/77/78/79.

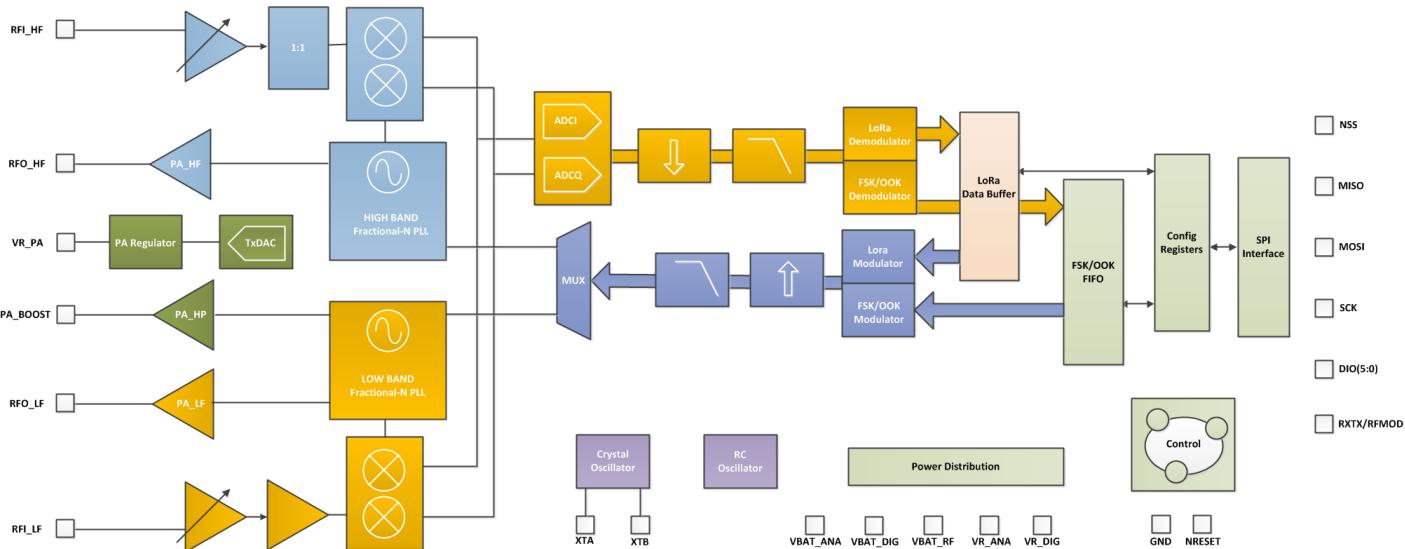


Figure 4. SX1276/77/78/79 Block Schematic Diagram

SX1276/77/78/79 is a half-duplex, low-IF transceiver. Here the received RF signal is first amplified by the LNA. The LNA inputs are single ended to minimize the external BoM and for ease of design. Following the LNA inputs, the conversion to differential is made to improve the second order linearity and harmonic rejection. The signal is then down-converted to in-phase and quadrature (I&Q) components at the intermediate frequency (IF) by the mixer stage. A pair of sigma delta ADCs then perform data conversion, with all subsequent signal processing and demodulation performed in the digital domain. The digital state machine also controls the automatic frequency correction (AFC), received signal strength indicator (RSSI) and automatic gain control (AGC). It also features the higher-level packet and protocol level functionality of the top level sequencer (TLS), only available with traditional FSK and OOK modulation schemes.

The frequency synthesizers generate the local oscillator (LO) frequency for both receiver and transmitter, one covering the lower UHF bands (up to 525 MHz), and the other one covering the upper UHF bands (from 779 MHz). The PLLs are optimized for user-transparent low lock time and fast auto-calibrating operation. In transmission, frequency modulation is performed digitally within the PLL bandwidth. The PLL also features optional pre-filtering of the bit stream to improve spectral purity.

SX1276/77/78/79 feature three distinct RF power amplifiers. Two of those, connected to RFO_LF and RFO_HF, can deliver up to +14 dBm, are unregulated for high power efficiency and can be connected directly to their respective RF receiver inputs via a pair of passive components to form a single antenna port high efficiency transceiver. The third PA, connected to the PA_BOOST pin and can deliver up to +20 dBm via a dedicated matching network. Unlike the high efficiency PAs, this high-stability PA covers all frequency bands that the frequency synthesizer addresses.

SX1276/77/78/79 also include two timing references, an RC oscillator and a 32 MHz crystal oscillator.

All major parameters of the RF front end and digital state machine are fully configurable via an SPI interface which gives access to SX1276/77/78/79's configuration registers. This includes a mode auto sequencer that oversees the transition and calibration of the SX1276/77/78/79 between intermediate modes of operation in the fastest time possible.

The SX1276/77/78/79 are equipped with both standard FSK and long range spread spectrum (LoRaTM) modems. Depending upon the mode selected either conventional OOK or FSK modulation may be employed or the LoRaTM spread spectrum modem.

3.1. LoRaTM Modem

The LoRaTM modem uses a proprietary spread spectrum modulation technique. This modulation, in contrast to legacy modulation techniques, permits an increase in link budget and increased immunity to in-band interference. At the same time the frequency tolerance requirement of the crystal reference oscillator is relaxed - allowing a performance increase for a reduction in system cost. For a detailed description of the design trade-offs and operation of the SX1276/77/78/79 please consult Section 4.1 of the datasheet.

3.2. FSK/OOK Modem

In FSK/OOK mode the SX1276/77/78/79 supports standard modulation techniques including OOK, FSK, GFSK, MSK and GMSK. The SX1276/77/78/79 is especially suited to narrow band communication thanks the low-IF architecture employed and the built-in AFC functionality. For full information on the FSK/OOK modem please consult Section 4.2 of this document.

4. SX1276/77/78/79 Digital Electronics

4.1. The LoRaTM Modem

The LoRaTM modem uses spread spectrum modulation and forward error correction techniques to increase the range and robustness of radio communication links compared to traditional FSK or OOK based modulation. Examples of the performance improvement possible, for several possible settings, are summarised in the table below. Here the spreading factor and error correction rate are design variables that allow the designer to optimise the trade-off between occupied bandwidth, data rate, link budget improvement and immunity to interference.

Table 12 Example LoRaTM Modem Performances, 868MHz Band

| Bandwidth (kHz) | Spreading Factor | Coding rate | Nominal Rb (bps) | Sensitivity indication (dBm) | Frequency Reference |
|-----------------|------------------|-------------|------------------|------------------------------|---------------------|
| 10.4 | 6 | 4/5 | 782 | -131 | TCXO |
| | 12 | 4/5 | 24 | -147 | |
| 20.8 | 6 | 4/5 | 1562 | -128 | TCXO |
| | 12 | 4/5 | 49 | -144 | |
| 62.5 | 6 | 4/5 | 4688 | -121 | XTAL |
| | 12 | 4/5 | 146 | -139 | |
| 125 | 6 | 4/5 | 9380 | -118 | |
| | 12 | 4/5 | 293 | -136 | |

Notes - for all bandwidths lower than 62.5 kHz, it is advised to use a TCXO as a frequency reference. This is required to meet the frequency error tolerance specifications given in the Electrical Specification

- Higher spreading factors and longer transmission times impose more stringent constraints on the short term frequency stability of the reference. Please get in touch with a Semtech representative to implement extremely low sensitivity products.

For European operation the range of crystal tolerances acceptable for each sub-band (of the ERC 70-03) is given in the specifications table. For US based operation a frequency hopping mode is available that automates both the LoRaTM spread spectrum and frequency hopping spread spectrum processes.

Another important facet of the LoRaTM modem is its increased immunity to interference. The LoRaTM modem is capable of co-channel GMSK rejection of up to 20 dB. This immunity to interference permits the simple coexistence of LoRaTM modulated systems either in bands of heavy spectral usage or in hybrid communication networks that use LoRaTM to extend range when legacy modulation schemes fail.

4.1.1. Link Design Using the LoRaTM Modem

4.1.1.1. Overview

The LoRaTM modem is setup as shown in the following figure. This configuration permits the simple replacement of the FSK modem with the LoRaTM modem via the configuration register setting *RegOpMode*. This change can be performed on the fly (in Sleep operating mode) thus permitting the use of both standard FSK or OOK in conjunction with the long range capability. The LoRaTM modulation and demodulation process is proprietary, it uses a form of spread spectrum modulation combined with cyclic error correction coding. The combined influence of these two factors is an increase in link budget and enhanced immunity to interference.

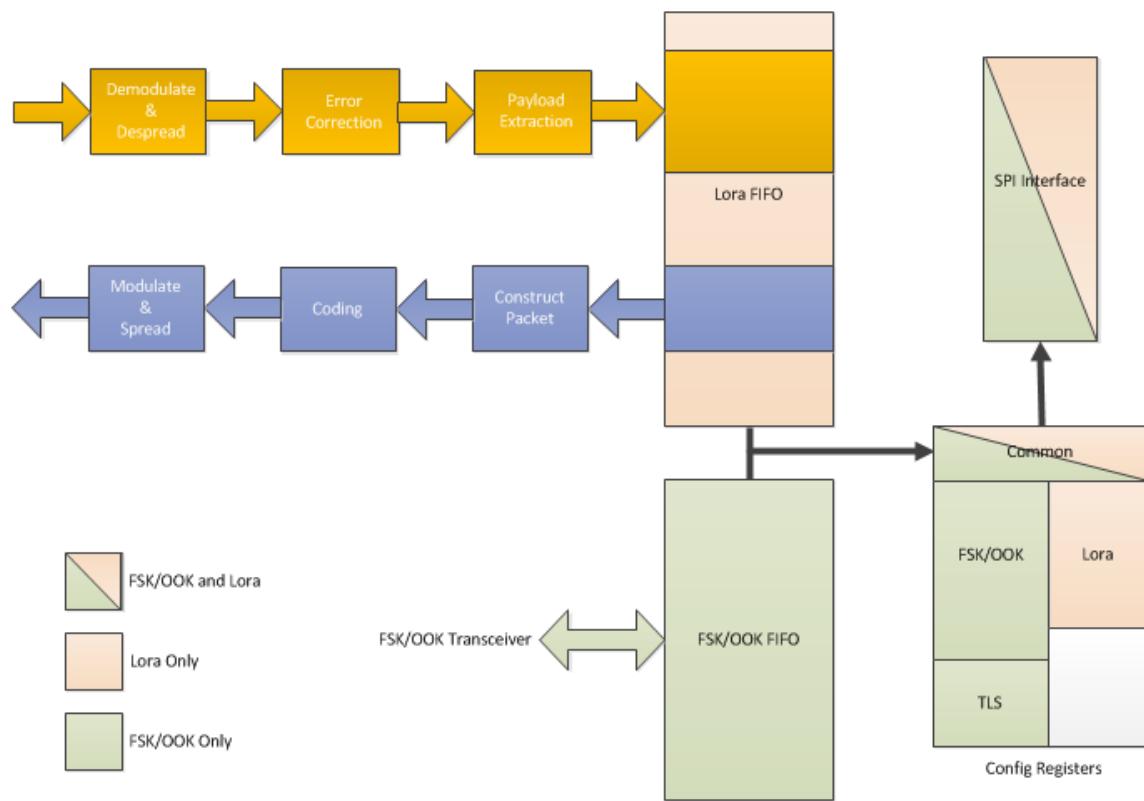


Figure 5. LoRaTM Modem Connectivity

A simplified outline of the transmit and receive processes is also shown above. Here we see that the LoRaTM modem has an independent dual port data buffer FIFO that is accessed through an SPI interface common to all modes. Upon selection of LoRaTM mode, the configuration register mapping of the SX1276/77/78/79 changes. For full details of this change please consult the register description of Section 6.

So that it is possible to optimise the LoRaTM modulation for a given application, access is given to the designer to three critical design parameters. Each one permitting a trade off between link budget, immunity to interference, spectral occupancy and nominal data rate. These parameters are spreading factor, modulation bandwidth and error coding rate.

4.1.1.2. Spreading Factor

The spread spectrum LoRaTM modulation is performed by representing each bit of payload information by multiple chips of information. The rate at which the spread information is sent is referred to as the symbol rate (Rs), the ratio between the nominal symbol rate and chip rate is the spreading factor and represents the number of symbols sent per bit of information. The range of values accessible with the LoRaTM modem are shown in the following table.

Table 13 Range of Spreading Factors

| <i>SpreadingFactor</i> (RegModulationCfg) | Spreading Factor (Chips / symbol) | LoRa Demodulator SNR |
|--|--------------------------------------|-------------------------|
| 6 | 64 | -5 dB |
| 7 | 128 | -7.5 dB |
| 8 | 256 | -10 dB |
| 9 | 512 | -12.5 dB |
| 10 | 1024 | -15 dB |
| 11 | 2048 | -17.5 dB |
| 12 | 4096 | -20 dB |

Note that the spreading factor, *SpreadingFactor*, must be known in advance on both transmit and receive sides of the link as different spreading factors are orthogonal to each other. Note also the resulting signal to noise ratio (SNR) required at the receiver input. It is the capability to receive signals with negative SNR that increases the sensitivity, so link budget and range, of the LoRa receiver.

Spreading Factor 6

SF = 6 is a special use case for the highest data rate transmission possible with the LoRa modem. To this end several settings must be activated in the SX1276/77/78/79 registers when it is in use. These settings are only valid for SF6 and should be set back to their default values for other spreading factors:

- ◆ Set *SpreadingFactor* = 6 in *RegModemConfig2*
- ◆ The header must be set to Implicit mode.
- ◆ Set the bit field *DetectionOptimize* of register *RegLoRaDetectOptimize* to value "0b101".
- ◆ Write 0x0C in the register *RegDetectionThreshold*.

4.1.1.3. Coding Rate

To further improve the robustness of the link the LoRaTM modem employs cyclic error coding to perform forward error detection and correction. Such error coding incurs a transmission overhead - the resultant additional data overhead per transmission is shown in the table below.

Table 14 Cyclic Coding Overhead

| <i>CodingRate</i> (RegTxCfg1) | Cyclic Coding Rate | Overhead Ratio |
|----------------------------------|-----------------------|----------------|
| 1 | 4/5 | 1.25 |
| 2 | 4/6 | 1.5 |
| 3 | 4/7 | 1.75 |
| 4 | 4/8 | 2 |

Forward error correction is particularly efficient in improving the reliability of the link in the presence of interference. So that the coding rate (and so robustness to interference) can be changed in response to channel conditions - the coding rate can optionally be included in the packet header for use by the receiver. Please consult Section 4.1.1.6 for more information on the LoRa™ packet and header.

4.1.1.4. Signal Bandwidth

An increase in signal bandwidth permits the use of a higher effective data rate, thus reducing transmission time at the expense of reduced sensitivity improvement. There are of course regulatory constraints in most countries on the permissible occupied bandwidth. Contrary to the FSK modem which is described in terms of the single sideband bandwidth, the LoRa™ modem bandwidth refers to the double sideband bandwidth (or total channel bandwidth). The range of bandwidths relevant to most regulatory situations is given in the LoRa™ modem specifications table (see Section 2.5.5).

Table 15 LoRa Bandwidth Options

| Bandwidth (kHz) | Spreading Factor | Coding rate | Nominal Rb (bps) |
|-----------------|------------------|-------------|------------------|
| 7.8 | 12 | 4/5 | 18 |
| 10.4 | 12 | 4/5 | 24 |
| 15.6 | 12 | 4/5 | 37 |
| 20.8 | 12 | 4/5 | 49 |
| 31.2 | 12 | 4/5 | 73 |
| 41.7 | 12 | 4/5 | 98 |
| 62.5 | 12 | 4/5 | 146 |
| 125 | 12 | 4/5 | 293 |
| 250 | 12 | 4/5 | 586 |
| 500 | 12 | 4/5 | 1172 |

Note In the lower band (169 MHz), the 250 kHz and 500 kHz bandwidths are not supported.

4.1.1.5. LoRa™ Transmission Parameter Relationship

With a knowledge of the key parameters that can be controlled by the user we define the LoRa™ symbol rate as:

$$R_S = \frac{BW}{2^{SF}}$$

where BW is the programmed bandwidth and SF is the spreading factor. The transmitted signal is a constant envelope signal. Equivalently, one chip is sent per second per Hz of bandwidth.

4.1.1.6. LoRaTM Packet Structure

The LoRaTM modem employs two types of packet format, explicit and implicit. The explicit packet includes a short header that contains information about the number of bytes, coding rate and whether a CRC is used in the packet. The packet format is shown in the following figure.

The LoRaTM packet comprises three elements:

- ◆ A preamble.
- ◆ An optional header.
- ◆ The data payload.

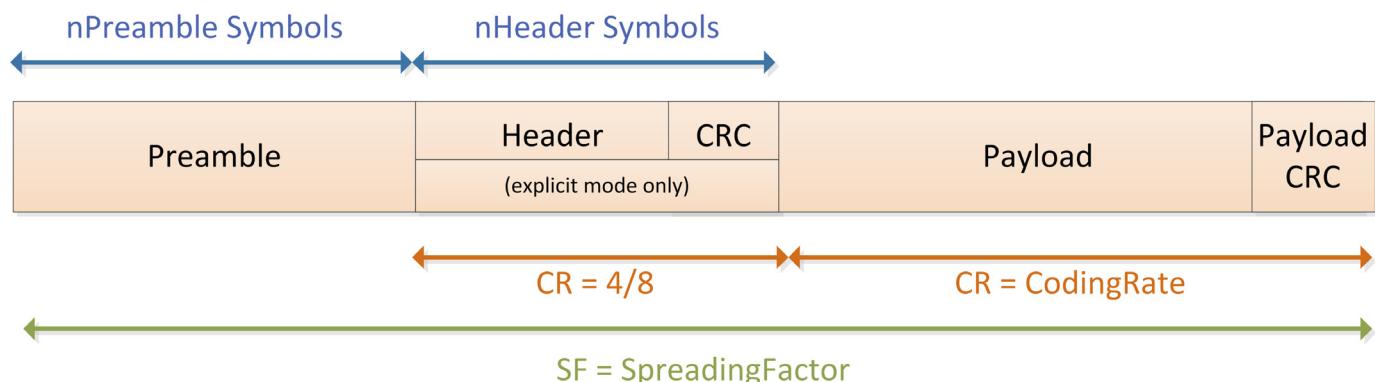


Figure 6. LoRaTM Packet Structure

Preamble

The preamble is used to synchronize receiver with the incoming data flow. By default the packet is configured with a 12 symbol long sequence. This is a programmable variable so the preamble length may be extended, for example in the interest of reducing receiver duty cycle in receive intensive applications. However, the minimum length suffices for all communication. The transmitted preamble length may be changed by setting the register *PreambleLength* from 6 to 65535, yielding total preamble lengths of 6+4 to 65535+4 symbols, once the fixed overhead of the preamble data is considered. This permits the transmission of a near arbitrarily long preamble sequence.

The receiver undertakes a preamble detection process that periodically restarts. For this reason the preamble length should be configured identical to the transmitter preamble length. Where the preamble length is not known, or can vary, the maximum preamble length should be programmed on the receiver side.

Header

Depending upon the chosen mode of operation two types of header are available. The header type is selected by the *ImplicitHeaderModeOn* bit found within the *RegModemConfig1* register.

Explicit Header Mode

This is the default mode of operation. Here the header provides information on the payload, namely:

- ◆ The payload length in bytes.
- ◆ The forward error correction code rate
- ◆ The presence of an optional 16-bits CRC for the payload.

The header is transmitted with maximum error correction code (4/8). It also has its own CRC to allow the receiver to discard invalid headers.

Implicit Header Mode

In certain scenarios, where the payload, coding rate and CRC presence are fixed or known in advance, it may be advantageous to reduce transmission time by invoking implicit header mode. In this mode the header is removed from the packet. In this case the payload length, error coding rate and presence of the payload CRC must be manually configured on both sides of the radio link.

Note With SF = 6 selected, implicit header mode is the only mode of operation possible.

Explicit Header Mode:

In Explicit Header Mode, the presence of the CRC at the end of the payload is selected only on the transmitter side through the bit *RxPayloadCrcOn* in the register *RegModemConfig1*.

On the receiver side, the bit *RxPayloadCrcOn* in the register *RegModemConfig1* is not used and once the payload has been received, the user should check the bit *CrcOnPayload* in the register *RegHopChannel*. If the bit *CrcOnPayload* is at '1', the user should then check the IRQ Flag *PayloadCrcError* to make sure the CRC is valid.

If the bit *CrcOnPayload* is at '0', it means there was no CRC on the payload and thus the IRQ Flag *PayloadCrcError* will not be triggered even if the payload has errors.

| Explicit Header | Transmitter | Receiver | CRC Status |
|---|-------------|----------|--------------------|
| Value of the bit <i>RxPayloadCrcOn</i> | 0 | 0 | CRC is not checked |
| | 0 | 1 | CRC is not checked |
| | 1 | 0 | CRC is checked |
| | 1 | 1 | CRC is checked |

Implicit Header Mode:

In Implicit Header Mode, it is necessary to set the bit *RxPayloadCrcOn* in the register *RegModemConfig1* on both sides (TX and RX)

| Implicit Header | Transmitter | Receiver | CRC Status |
|---|-------------|----------|---------------------|
| Value of the bit <i>RxPayloadCrcOn</i> | 0 | 0 | CRC is not checked |
| | 0 | 1 | CRC is always wrong |
| | 1 | 0 | CRC is not checked |
| | 1 | 1 | CRC is checked |

Low Data Rate Optimization

Given the potentially long duration of the packet at high spreading factors the option is given to improve the robustness of the transmission to variations in frequency over the duration of the packet transmission and reception. The bit *LowDataRateOptimize* increases the robustness of the LoRa link at these low effective data rates. Its use is mandated when the symbol duration exceeds 16ms. Note that both the transmitter and the receiver must have the same setting for *LowDataRateOptimize*.

Payload

The packet payload is a variable-length field that contains the actual data coded at the error rate either as specified in the header in explicit mode or in the register settings in implicit mode. An optional CRC may be appended. For more information on the payload and how it is loaded from the data buffer FIFO please see Section 4.1.2.3.

4.1.1.7. Time on air

For a given combination of spreading factor (SF), coding rate (CR) and signal bandwidth (BW) the total on-the-air transmission time of a LoRaTM packet can be calculated as follows. From the definition of the symbol rate it is convenient to define the symbol rate:

$$T_s = \frac{1}{R_s}$$

The LoRa packet duration is the sum of the duration of the preamble and the transmitted packet. The preamble length is calculated as follows:

$$T_{preamble} = (n_{preamble} + 4.25)T_{sym}$$

where $n_{preamble}$ is the programmed preamble length, taken from the registers *RegPreambleMsb* and *RegPreambleLsb*. The payload duration depends upon the header mode that is enabled. The following formula gives the number of payload symbols.

$$n_{payload} = 8 + \max\left(\left\lceil \frac{(8PL - 4SF + 28 + 16CRC - 20IH)}{4(SF - 2DE)} \right\rceil (CR + 4), 0\right)$$

With the following dependencies:

- ◆ PL is the number of Payload bytes (1 to 255)
 - ◆ SF is the spreading factor (6 to 12)
 - ◆ IH=0 when the header is enabled, IH=1 when no header is present
 - ◆ DE=1 when *LowDataRateOptimize*=1, DE=0 otherwise
 - ◆ CR is the coding rate (1 corresponding to 4/5, 4 to 4/8)
- The Payload duration is then the symbol period multiplied by the number of Payload symbols

$$T_{payload} = n_{payload} \times T_s$$

The time on air, or packet duration, is simply then the sum of the preamble and payload duration.

$$T_{packet} = T_{preamble} + T_{payload}$$

4.1.1.8. Frequency Hopping with LoRaTM

Frequency hopping spread spectrum (FHSS) is typically employed when the duration of a single packet could exceed regulatory requirements relating to the maximum permissible channel dwell time. This is most notably the case in US operation where the 902 to 928 MHz ISM band which makes provision for frequency hopping operation. To ease the implementation of FHSS systems the frequency hopping mode of the LoRaTM modem can be enabled by setting *FreqHoppingPeriod* to a non-zero value in register *RegHopPeriod*.

Principle of Operation

The principle behind the FHSS scheme is that a portion of each LoRaTM packet is transmitted on each hopping channel from a look up table of frequencies managed by the host microcontroller. After a predetermined hopping period the transmitter and receiver change to the next channel in a predefined list of hopping frequencies to continue transmission and reception of the next portion of the packet. The time which the transmission will dwell in any given channel is determined by *FreqHoppingPeriod* which is an integer multiple of symbol periods:

$$\text{HoppingPeriod} = Ts \times \text{FreqHoppingPeriod}$$

The frequency hopping transmission and reception process starts at channel 0. The preamble and header are transmitted first on channel 0. At the beginning of each transmission the channel counter *FhssPresentChannel* (located in the register *RegHopChannel*) is incremented and the interrupt signal *FhssChangeChannel* is generated. The new frequency must then be programmed within the hopping period to ensure it is taken into account for the next hop, the interrupt *ChangeChannelFhss* is then to be cleared by writing a logical '1'.

FHSS Reception always starts on channel 0. The receiver waits for a valid preamble detection before starting the frequency hopping process as described above. Note that in the eventuality of header CRC corruption, the receiver will automatically request channel 0 and recommence the valid preamble detection process.

Timing of Channel Updates

The interrupt requesting the channel change, *FhssChangeChannel*, is generated upon transition to the new frequency. The frequency hopping process is illustrated in the diagram below:

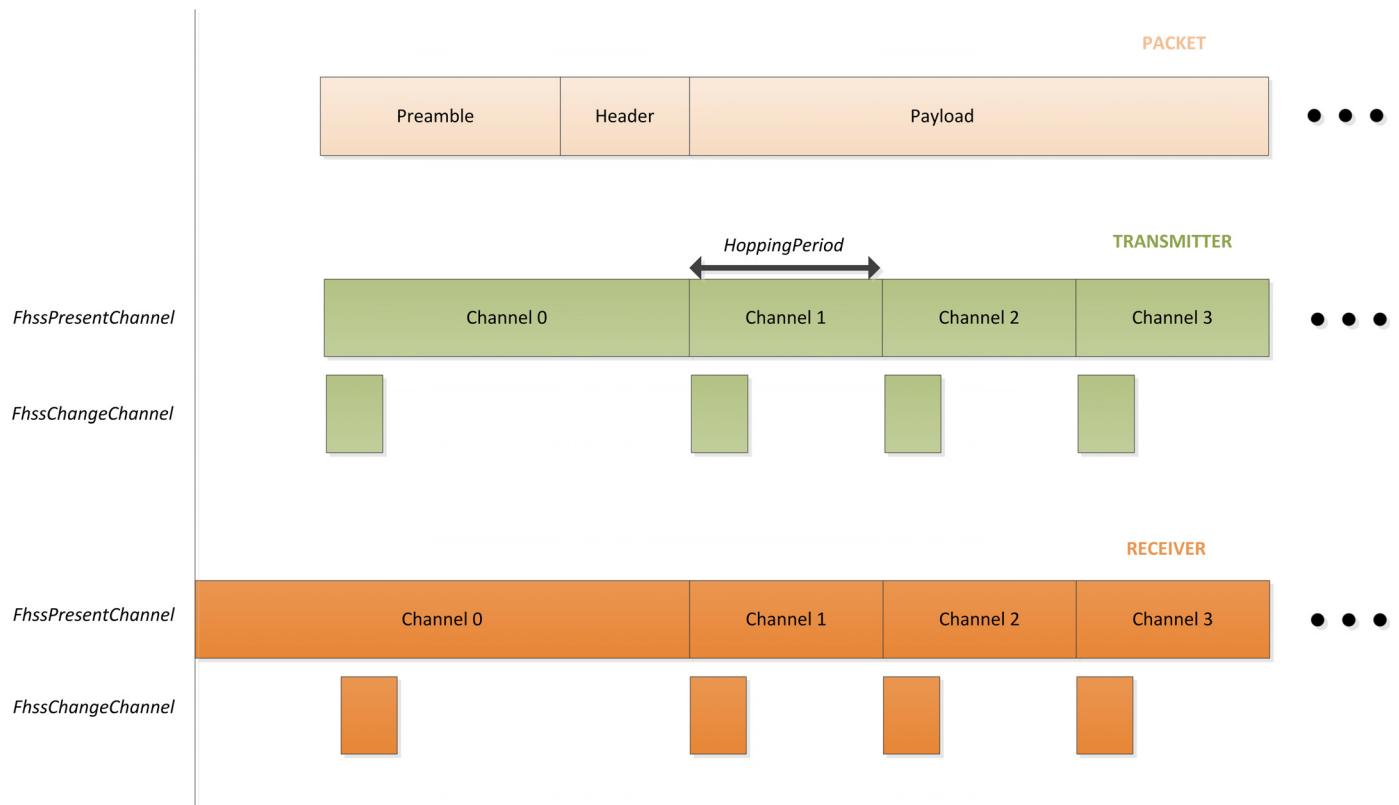


Figure 7. Interrupts Generated in the Case of Successful Frequency Hopping Communication.

4.1.2. LoRaTM Digital Interface

The LoRaTM modem comprises three types of digital interface, static configuration registers, status registers and a FIFO data buffer. All are accessed through the SX1276/77/78/79's SPI interface - full details of each type of register are given below. Full listings of the register addresses used for SPI access are given in Section 6.4.

4.1.2.1. LoRaTM Configuration Registers

Configuration registers are accessed through the SPI interface. Registers are readable in all device mode including Sleep. However, they should be **written only in Sleep and Standby modes**. Please note that **the automatic top level sequencer (TLS modes) are not available in LoRaTM mode and the configuration register mapping changes as shown in Table 41**. The content of the LoRaTM configuration registers is retained in FSK/OOK mode. For the functionality of mode registers common to both FSK/OOK and LoRaTM mode, please consult the Analog and RF Front End section of this document (Section 5).

4.1.2.2. Status Registers

Status registers provide status information during receiver operation.

4.1.2.3. LoRaTM Mode FIFO Data Buffer

Overview

The SX1276/77/78/79 is equipped with a 256 byte RAM data buffer which is uniquely accessible in LoRa mode. This RAM area, herein referred to as the FIFO Data buffer, is fully customizable by the user and allows access to the received, or to be transmitted, data. All access to the LoRaTM FIFO data buffer is done via the SPI interface. A diagram of the user defined memory mapping of the FIFO data buffer is shown below. These FIFO data buffer can be read in all operating modes except sleep and store data related to the last receive operation performed. It is automatically cleared of old content upon each new transition to receive mode.

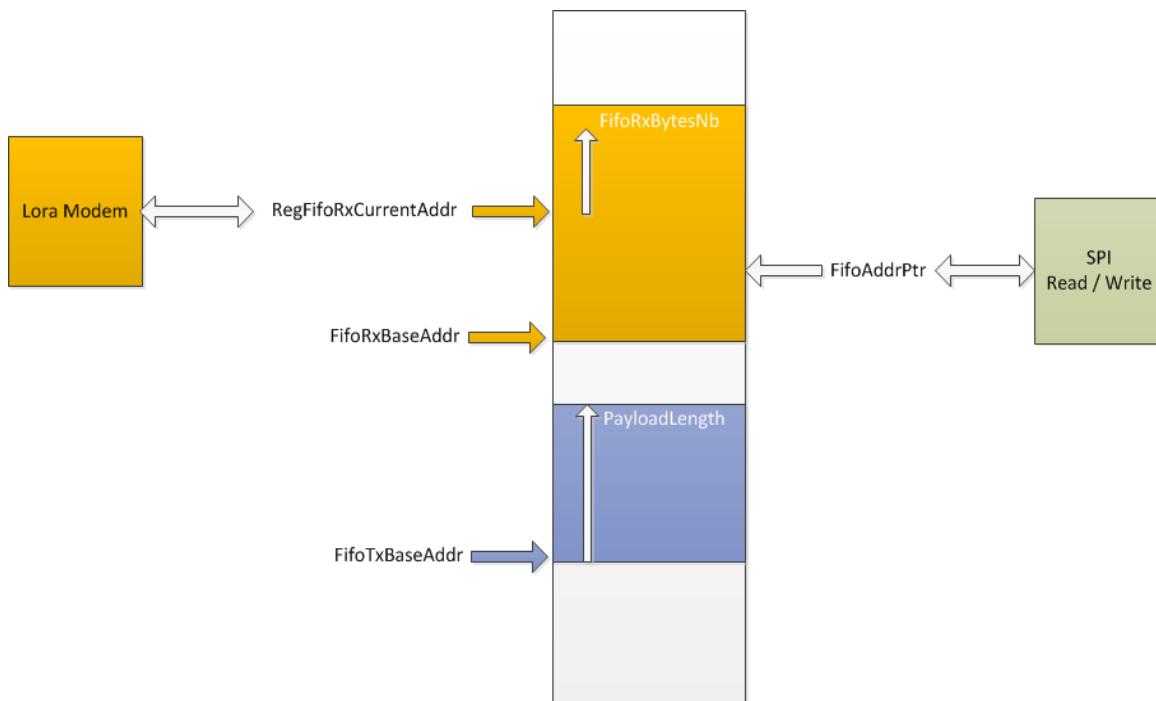


Figure 8. LoRaTM Data Buffer

Principle of Operation

Thanks to its dual port configuration, it is possible to simultaneously store both transmit and receive information in the FIFO data buffer. The register *RegFifoTxBaseAddr* specifies the point in memory where the transmit information is stored. Similarly, for receiver operation, the register *RegFifoRxBaseAddr* indicates the point in the data buffer where information will be written to in event of a receive operation.

By default, the device is configured at power up so that half of the available memory is dedicated to Rx (*RegFifoRxBaseAddr* initialized at address 0x00) and the other half is dedicated for Tx (*RegFifoTxBaseAddr* initialized at address 0x80).

However, due to the contiguous nature of the FIFO data buffer, the base addresses for Tx and Rx are fully configurable across the 256 byte memory area. Each pointer can be set independently anywhere within the FIFO. To exploit the maximum FIFO data buffer size in transmit or receive mode, the whole FIFO data buffer can be used in each mode by setting the base addresses *RegFifoTxBaseAddr* and *RegFifoRxBaseAddr* at the bottom of the memory (0x00).

The FIFO data buffer is cleared when the device is put in SLEEP mode, consequently no access to the FIFO data buffer is possible in sleep mode. However, the data in the FIFO data buffer are retained when switching across the other LoRaTM modes of operation, so that a received packet can be retransmitted with minimum data handling on the controller side. The FIFO data buffer is not self-clearing (unless if the device is put in sleep mode) and the data will only be “erased” when a new set of data is written into the occupied memory location.

The FIFO data buffer location to be read from, or written to, via the SPI interface is defined by the address pointer *RegFifoAddrPtr*. Before any read or write operation it is hence necessary to initialize this pointer to the corresponding base value. Upon reading or writing to the FIFO data buffer (*RegFifo*) the address pointer will then increment automatically.

The register *RegRxNbBytes* defines the size of the memory location to be written in the event of a successful receive operation. The register *RegPayloadLength* indicates the size of the memory location to be transmitted. In implicit header mode, the register *RegRxNbBytes* is not used as the number of payload bytes is known. Otherwise, in explicit header mode, the initial size of the receive buffer is set to the packet length in the received header. The register *RegFifoRxCurrentAddr* indicates the location of the last packet received in the FIFO so that the last packet received can be easily read by pointing the register *RegFifoAddrPtr* to this register.

It is important to notice that all the received data will be written to the FIFO data buffer even if the CRC is invalid, permitting user defined post processing of corrupted data. It is also important to note that when receiving, if the packet size exceeds the buffer memory allocated for the Rx, it will overwrite the transmit portion of the data buffer.

4.1.2.4. Interrupts in LoRa Mode

Two registers are used to control the IRQ in LoRa mode, the register *RegIrqFlagsMask* which is used to mask the interrupts and the register *RegIrqFlags* which indicates which IRQ has been triggered.

In the register *RegIrqFlagsMask*, setting a bit to ‘1’ will mask the interrupt, meaning this interrupt is deactivated. By default all the interrupt are available.

In the register *RegIrqFlags*, a ‘1’ indicates a given IRQ has been triggered and then the IRQ must be clear by writing a ‘1’.

4.1.3. Operation of the LoRaTM Modem

4.1.3.1. Operating Mode Control

The operating modes of the LoRaTM modem are accessed by enabling LoRaTM mode (setting the *LongRangeMode* bit of *RegOpMode*). Depending upon the operating mode selected the range of functionality and register access is given by the following table:

Table 16 LoRaTM Operating Mode Functionality

| Operating Mode | Description |
|---------------------|---|
| SLEEP | Low-power mode. In this mode only SPI and configuration registers are accessible. Lora FIFO is not accessible. Note that this is the only mode permissible to switch between FSK/OOK mode and LoRa mode. |
| STANDBY | both Crystal oscillator and Lora baseband blocks are turned on. RF part and PLLs are disabled |
| FSTX | This is a frequency synthesis mode for transmission. The PLL selected for transmission is locked and active at the transmit frequency. The RF part is off. |
| FSRX | This is a frequency synthesis mode for reception. The PLL selected for reception is locked and active at the receive frequency. The RF part is off. |
| TX | When activated the SX1276/77/78/79 powers all remaining blocks required for transmit, ramps the PA, transmits the packet and returns to Standby mode. |
| RXCONTINUOUS | When activated the SX1276/77/78/79 powers all remaining blocks required for reception, processing all received data until a new user request is made to change operating mode. |
| RXSINGLE | When activated the SX1276/77/78/79 powers all remaining blocks required for reception, remains in this state until a valid packet has been received and then returns to Standby mode. |
| CAD | When in CAD mode, the device will check a given channel to detect LoRa preamble signal |

It is possible to access any mode from any other mode by changing the value in the *RegOpMode* register.

4.1.4. Frequency Settings

Recalling that the frequency step is given by:

$$F_{STEP} = \frac{F_{XOSC}}{2^{19}}$$

In order to set LO frequency values following registers are available.

*Fr*f is a 24-bit register which defines carrier frequency. The carrier frequency relates to the register contents by following formula:

$$F_{RF} = F_{STEP} \times Frf(23,0)$$

4.1.5. Frequency Error Indication

The SX1276/77/78/79 derives its RF centre frequency from a crystal reference oscillator which has a finite frequency precision. Errors in reference frequency will manifest themselves as errors of the same proportion from the RF centre frequency.

In LoRa receive mode the SX1276/77/78/79 is capable of measuring the frequency offset between the receiver centre frequency and that of an incoming LoRa signal. The modem is intolerant of frequency offsets in the region of +/- 25% of the bandwidth and will accurately report the error over this same range.

The error is read by reading the three *RegFei* registers. The contents of which are a signed 20 bit two's compliment word, *FreqError*. The frequency error is determined from the register contents by:

$$F_{Error} = \frac{F_{FreqError} \times 2^{24}}{F_{xtal}} \times \frac{BW[kHz]}{500}$$

Where *Fxtal* is the crystal frequency.

To correct the measured frequency error there are two steps to be taken. First the frequency error is subtracted from the RF centre frequency. This calculation must be performed locally (or in a look-up-table), no provision is made in the circuit to apply the correction automatically.

Secondly, assuming that the frequency error is due to reference oscillator drift, the data rate of the LoRa modem must also be compensated accordingly. This is done by

$$PpmOffset = 0.95 * measured\ Offset [PPM]$$

Where *PpmOffset* is the value to be programmed into register 0x27 and the measured Offset is the PPM drift equivalent to the frequency error reported by the LoRa frequency error indicator. The *PpmOffset* value is a signed two's compliment value.

4.1.6. LoRaTM Modem State Machine Sequences

The sequence for transmission and reception of data to and from the LoRaTM modem, together with flow charts of typical sequences of operation, are detailed below.

Data Transmission Sequence

In transmit mode power consumption is optimized by enabling RF, PLL and PA blocks only when packet data needs to be transmitted. Figure 9 shows a typical LoRaTM transmit sequence.

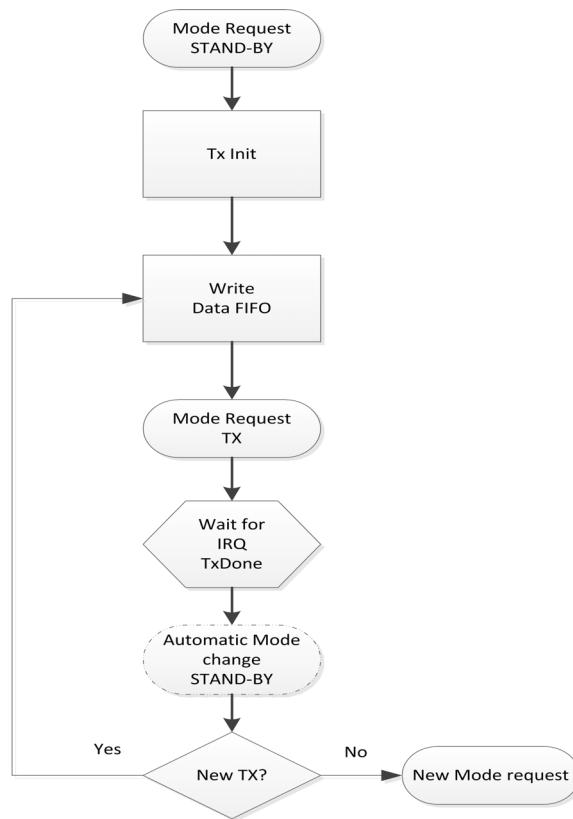


Figure 9. LoRaTM Modulation Transmission Sequence.

- ◆ Static configuration registers can only be accessed in Sleep mode, Standby mode or FSTX mode.
- ◆ The LoRaTM FIFO can only be filled in Standby mode.
- ◆ Data transmission is initiated by sending TX mode request.
- ◆ Upon completion the *TxDone* interrupt is issued and the radio returns to Standby mode.
- ◆ Following transmission the radio can be manually placed in Sleep mode or the FIFO refilled for a subsequent Tx operation.

LoRaTM Transmit Data FIFO Filling

In order to write packet data into FIFO user should:

- 1 Set *FifoPtrAddr* to *FifoTxPtrBase*.
- 2 Write *PayloadLength* bytes to the FIFO (*RegFifo*)

Data Reception Sequence

Figure 10 shows typical LoRaTM receive sequences for both single and continuous receiver modes of operation.

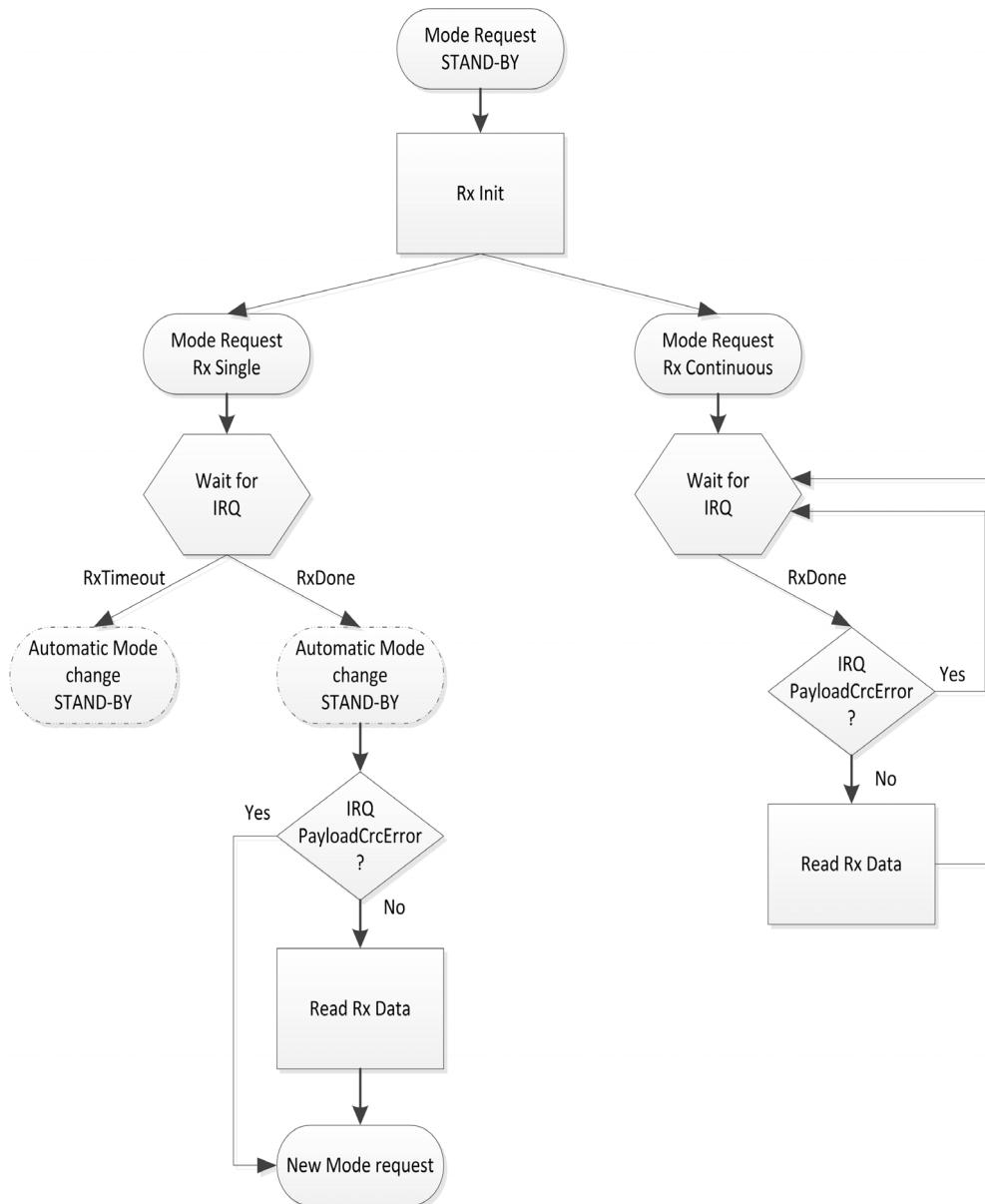


Figure 10. LoRaTM Receive Sequence.

The LoRa receive modem can work in two distinct mode

1. Single receive mode
2. Continuous receive mode

Those two modes correspond to different use cases and it is important to understand the subtle differences between them.

Single Reception Operating Mode

In this mode, the modem searches for a preamble during a given period of time. If a preamble hasn't been found at the end of the time window, the chip generates the *RxTimeout* interrupt and goes back to Standby mode. The length of the reception window (in symbols) is defined by the *RegSymbTimeout* register and should be in the range of 4 (minimum time for the modem to acquire lock on a preamble) up to 1023 symbols.

At the end of the payload, the *RxDone* interrupt is generated together with the interrupt *PayloadCrcError* if the payload CRC is not valid. However, even when the CRC is not valid, the data are written in the FIFO data buffer for post processing. Following the *RxDone* interrupt the radio goes to Standby mode.

The modem will also automatically return in Standby mode when the interrupts *RxDone* is generated. Therefore, this mode should only be used when the time window of arrival of the packet is known. In other cases, the RX continuous mode should be used.

In Rx single mode, low-power is achieved by turning off PLL and RF blocks as soon as a packet has been received. The flow is as follows:

- 1 Set *FifoAddrPtr* to *FifoRxBaseAddr*.
- 2 Static configuration register device can be written in either Sleep mode, Standby mode or FSRX mode.
- 3 A single packet receive operation is initiated by selecting the operating mode RXSINGLE.
- 4 The receiver will then await the reception of a valid preamble. Once received, the gain of the receive chain is set. Following the ensuing reception of a valid header, indicated by the *ValidHeader* interrupt in explicit mode. The packet reception process commences. Once the reception process is complete the *RxDone* interrupt is set. The radio then returns automatically to Standby mode to reduce power consumption.
- 5 The receiver status register *PayloadCrcError* should be checked for packet payload integrity.
- 6 If a valid packet payload has been received then the FIFO should be read (See Payload Data Extraction below). Should a subsequent single packet reception need to be triggered, then the RXSINGLE operating mode must be re-selected to launch the receive process again - taking care to reset the SPI pointer (*FifoAddrPtr*) to the base location in memory (*FifoRxBaseAddr*).

Continuous Reception Operating Mode

In continuous receive mode, the modem scans the channel continuously for a preamble. Each time a preamble is detected the modem tracks it until the packet is received and then carries on waiting for the next preamble.

If the preamble length exceeds the anticipated value set by the registers *RegPreambleMsb* and *RegPreambleLsb* (measured in symbol periods) the preamble will be dropped and the search for a preamble restarted. However, this scenario will not be flagged by any interrupt. In continuous RX mode, opposite to the single RX mode, the *RxTimeout* interrupt will never occur and the device will never go in Standby mode automatically.

It is also important to note that the demodulated bytes are written in the data buffer memory in the order received. Meaning, the first byte of a new packet is written just after the last byte of the preceding packet. The RX modem address pointer is never reset as long as this mode is enabled. It is therefore necessary for the companion microcontroller to handle the address pointer to make sure the FIFO data buffer is never full.

In continuous mode the received packet processing sequence is given below.

- 1 Whilst in Sleep or Standby mode select RXCONT mode.
- 2 Upon reception of a valid header CRC the *RxDone* interrupt is set. The radio remains in RXCONT mode waiting for the next RX LoRa™ packet.
- 3 The *PayloadCrcError* flag should be checked for packet integrity.
- 4 If packet has been correctly received the FIFO data buffer can be read (see below).
- 5 The reception process (steps 2 - 4) can be repeated or receiver operating mode exited as desired.

In continuous mode status information are available only for the last packet received, i.e. the corresponding registers should be read before the next *RxDone* arrives.

Rx Single and Rx Continuous Use Cases

The LoRa single reception mode is used mainly in battery operated systems or in systems where the companion microcontroller has a limited availability of timers. In such systems, the use of the timeout present in Rx Single reception mode allows the end user to limit the amount of time spent in reception (and thus limiting the power consumption) while not using any of the companion MCU timers (the MCU can then be in sleep mode while the radio is in the reception mode). The RxTimeout interrupt generated at the end of the reception period is then used to wake-up the companion MCU. One of the advantages of the RxSingle mode is that the interrupt RxTimeout will not be triggered if the device is currently receiving data, thus giving the priority to the reception of the data over the timeout. However, if during the reception, the device loses track of the data due to external perturbation, the device will drop the reception, flag the interrupt RxTimeout and go in StandBy mode to decrease the power consumption of the system.

On the other hand, The LoRa continuous reception mode is used in systems which do not have power restrictions or on system where the use of a companion MCU timer is preferred over the radio embedded timeout system. In RxContinuous mode, the radio will track any LoRa signal present in the air and carry on the reception of packets until the companion MCU sets the radio into another mode of operation. Upon reception the interrupt *RxDone* will be triggered but the device will stay in Rx Mode, ready for the reception of the next packet.

Payload Data Extraction from FIFO

In order to retrieve received data from FIFO the user must ensure that *ValidHeader*, *PayloadCrcError*, *RxDone* and *RxTimeout* interrupts in the status register *RegIrqFlags* are not asserted to ensure that packet reception has terminated successfully (i.e. no flags should be set).

In case of errors the steps below should be skipped and the packet discarded. In order to retrieve valid received data from the FIFO the user must:

- ◆ *RegRxNbBytes* Indicates the number of bytes that have been received thus far.
- ◆ *RegFifoAddrPtr* is a dynamic pointer that indicates precisely where the Lora modem received data has been written up to.
- ◆ Set *RegFifoAddrPtr* to *RegFifoRxCurrentAddr*. This sets the FIFO pointer to the location of the last packet received in the FIFO. The payload can then be extracted by reading the register *RegFifo*, *RegRxNbBytes* times.
- ◆ Alternatively, it is possible to manually point to the location of the last packet received, from the start of the current packet, by setting *RegFifoAddrPtr* to *RegFifoRxByteAddr* minus *RegRxNbBytes*. The payload bytes can then be read from the FIFO by reading the *RegFifo* address *RegRxNbBytes* times.

Packet Filtering based on Preamble Start

The LoRa modem does automatically filter received packets based upon any addressing. However the SX1276/77/78/79 permit software filtering of the received packets based on the contents of the first few bytes of payload. A brief example is given below for a 4 byte address, however, the address length can be selected by the designer.

The objective of the packet filtering process is to determine the presence, or otherwise, of a valid packet designed for the receiver. If the packet is not for the receiver then the radio returns to sleep mode in order to improve battery life.

The software packet filtering process follows the steps below:

- ◆ Each time the RxDone interrupt is received, latch the *RegFifoRxByteAddr[7:0]* register content in a variable, this variable will be called *start_address*. The *RegFifoRxByteAddr[7:0]* register of the SX1276/77/78/79 gives in real time the address of the last byte written in the data buffer + 1 (or the address at which the next byte will be written) by the receive LoRa modem. So by doing this, we make sure that the variable *start_address* always contains the start address of the next packet.
- ◆ Upon reception of the interrupt *ValidHeader*, start polling the *RegFifoRxByteAddr[7:0]* register until it begins to increment. The speed at which this register will increment depends on the Spreading factor, the error correction code and the modulation bandwidth. (Note that this interrupt is still generated in implicit mode).
- ◆ As soon as *RegFifoRxByteAddr[7:0] >= start address + 4*, the first 4 bytes (address) are stored in the FIFO data buffer. These can be read and tested to see if the packet is destined for the radio and either remaining in Rx mode to receive the packet or returning to sleep mode if not.

Receiver Timeout Operation

In LoRaTM Rx Single mode, a receiver timeout functionality is available that permits the receiver to listen for a predetermined period of time before generating an interrupt signal to indicate that no valid packets have been received. The timer is absolute and commences as soon as the radio is placed in single receive mode. The interrupt itself, *RxTimeout*, can be found in the interrupt register *RegIrqFlags*. In Rx Single mode, the device will return to Standby mode as soon as the interrupt occurs. The user must then clear the interrupt or go into Sleep mode before returning into Rx Single mode. The programmed timeout value is expressed as a multiple of the symbol period and is given by:

$$\text{TimeOut} = \text{LoraRxTimeout} \cdot T_s$$

Channel Activity Detection

The use of a spread spectrum modulation technique presents challenges in determining whether the channel is already in use by a signal that may be below the noise floor of the receiver. The use of the RSSI in this situation would clearly be impracticable. To this end the channel activity detector is used to detect the presence of other LoRa™ signals. Figure 11 shows the channel activity detection (CAD) process:

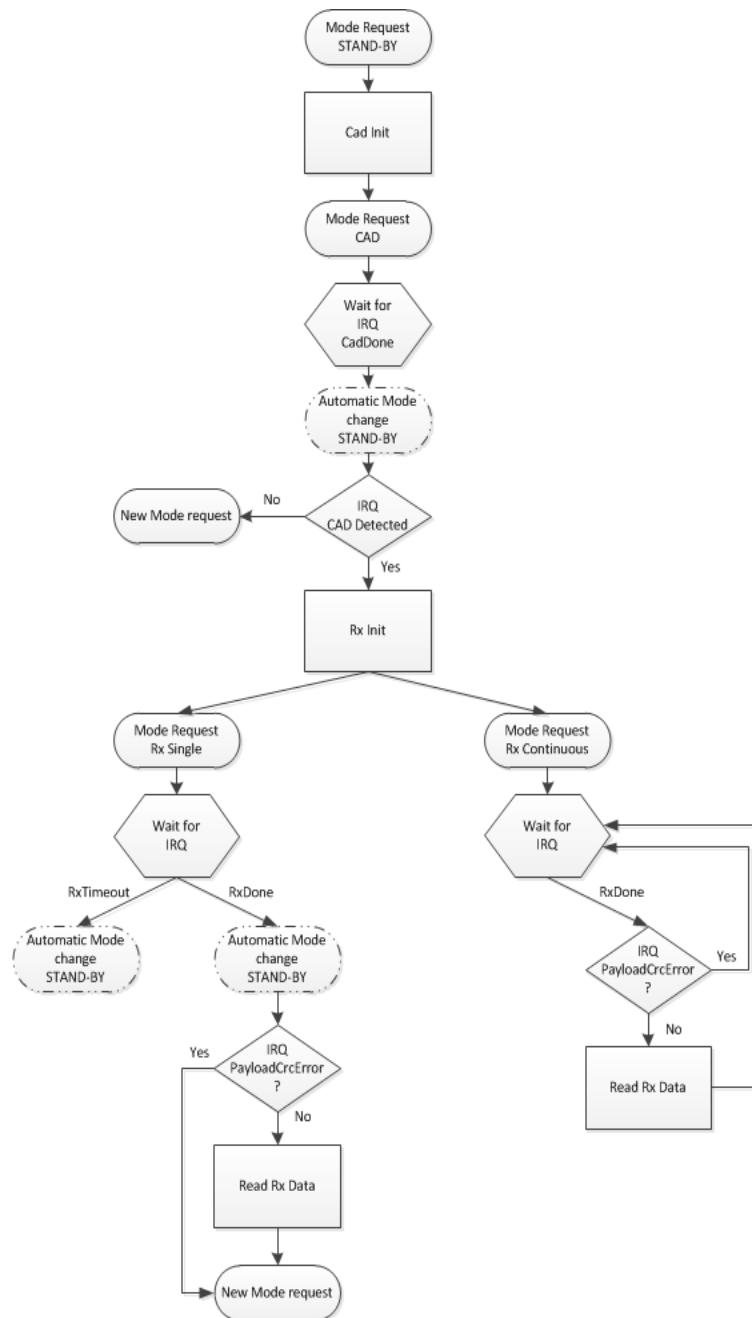


Figure 11. LoRa™ CAD Flow

Principle of Operation

The channel activity detection mode is designed to detect a LoRa preamble on the radio channel with the best possible power efficiency. Once in CAD mode, the SX1276/77/78/79 will perform a very quick scan of the band to detect a LoRa packet preamble.

During a CAD the following operations take place:

- ◆ The PLL locks
- ◆ The radio receiver captures LoRa preamble symbol of data from the channel. The radio current consumption during that phase corresponds to the specified Rx mode current
- ◆ The radio receiver and the PLL turn off, and the modem digital processing starts.
- ◆ The modem searches for a correlation between the radio captured samples and the ideal preamble waveform. This correlation process takes a little bit less than a symbol period to perform. The radio current consumption during that phase is greatly reduced.
- ◆ Once the calculation is finished the modem generates the CadDone interrupt. If the correlation was successful, CadDetected is generated simultaneously.
- ◆ The chip goes back to Standby mode.
- ◆ If a preamble was detected, clear the interrupt, then initiate the reception by putting the radio in RX single mode or RX continuous mode.

The time taken for the channel activity detection is dependent upon the LoRa modulation settings used. For a given configuration the typical CAD detection time is shown in the graph below, expressed as a multiple of the LoRa symbol period. Of this period the radio is in receiver mode for $(2^{SF} + 32) / \text{BW}$ seconds. For the remainder of the CAD cycle the radio is in a reduced consumption state.

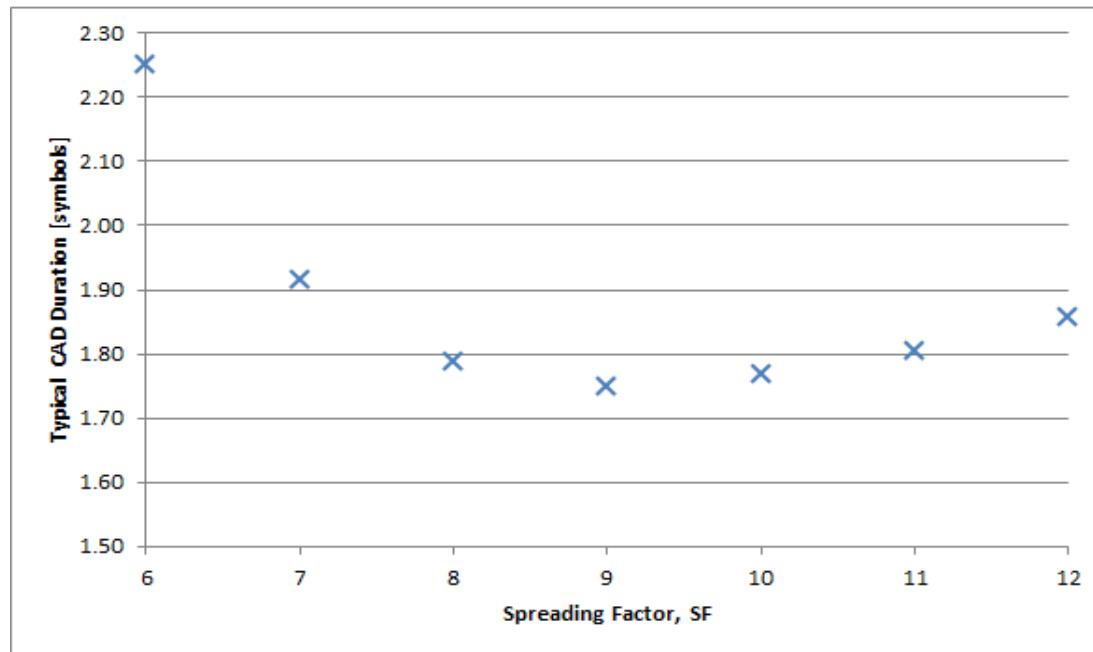


Figure 12. CAD Time as a Function of Spreading Factor

To illustrate this process and the respective consumption in each mode, the CAD process follows the sequence of events outlined below:

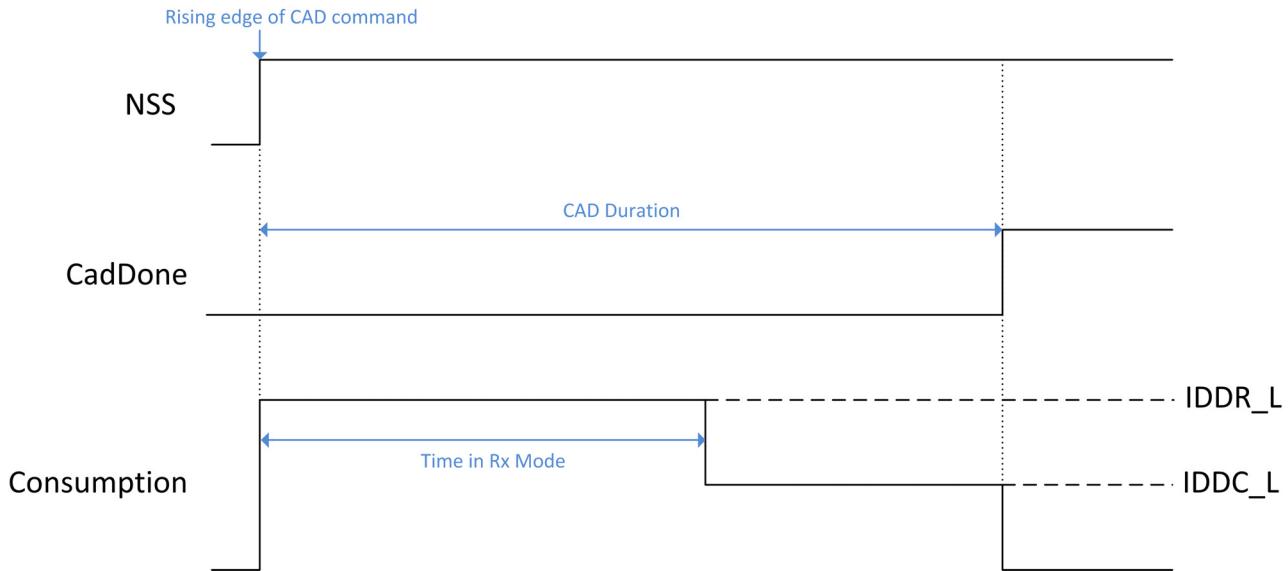


Figure 13. Consumption Profile of the LoRa CAD Process

The receiver is then in full receiver mode for just over half of the activity detection, followed by a reduced consumption processing phase where the consumption varies with the LoRa bandwidth as shown in the table below.

Table 17 LoRa CAD Consumption Figures

| Bandwidth (kHz) | Full Rx, IDDR_L (mA) | Processing, IDDC_L (mA) |
|-----------------|----------------------|-------------------------|
| 7.8 to 41.7 | 11 | 5.2 |
| 62.5 | 11 | 5.6 |
| 125 | 11.5 | 6 |
| 250 | 12.4 | 6.8 |
| 500 | 13.8 | 8.3 |

Note These numbers can be slightly lower when using Band 2 and 3, on the low frequency port.

4.1.6.1. Digital IO Pin Mapping

Six of SX1276/77/78/79's general purpose IO pins are available used in LoRa™ mode. Their mapping is shown below and depends upon the configuration of registers *RegDioMapping1* and *RegDioMapping2*.

Table 18 DIO Mapping LoRaTM Mode

| Operating Mode | DIOx Mapping | DIO5 | DIO4 | DIO3 | DIO2 | DIO1 | DIO0 |
|----------------|--------------|-----------|-------------|-----------------|-------------------|-------------------|---------|
| ALL | 00 | ModeReady | CadDetected | CadDone | FhssChangeChannel | RxTimeout | RxDone |
| | 01 | ClkOut | PllLock | ValidHeader | FhssChangeChannel | FhssChangeChannel | TxDone |
| | 10 | ClkOut | PllLock | PayloadCrcError | FhssChangeChannel | CadDetected | CadDone |
| | 11 | - | - | - | - | - | - |

4.1.7. Modem Status Indicators

The state of the LoRa modem is accessible with the *ModemStatus* bits in *RegModemStat*. They can mostly be used for debug in Rx mode and the useful indicators are:

- ◆ Bit 0: Signal Detected indicates that a valid LoRa preamble has been detected
- ◆ Bit 1: Signal Synchronized indicates that the end of Preamble has been detected, the modem is in lock
- ◆ Bit 3: Header Info Valid toggles high when a valid Header (with correct CRC) is detected

4.2. FSK/OOK Modem

4.2.1. Bit Rate Setting

The bitrate setting is referenced to the crystal oscillator and provides a precise means of setting the bit rate (or equivalently chip) rate of the radio. In continuous transmit mode (Section 4.2.12) the data stream to be transmitted can be input directly to the modulator via pin 10 (DIO2/DATA) in an asynchronous manner, unless Gaussian filtering is used, in which case the DCLK signal on pin 9 (DIO1/DCLK) is used to synchronize the data stream. See section 4.2.2.3 for details on the Gaussian filter.

In Packet mode or in Continuous mode with Gaussian filtering enabled, the Bit Rate (BR) is controlled by bits *Bitrate* in *RegBitrateMsb* and *RegBitrateLsb*

$$\text{BitRate} = \frac{\text{EXOSC}}{\text{BitRate}(15,0) + \frac{\text{BitrateFrac}}{16}}$$

Note: *BitrateFrac* bits have **no effect** (i.e. may be considered equal to 0) **in OOK modulation mode**.

The quantity *BitrateFrac* is hence designed to allow very high precision (max. 250 ppm programming resolution) for any bitrate in the programmable range. Table 19 below shows a range of standard bitrates and the accuracy to within which they may be reached.

Table 19 Bit Rate Examples

| Type | BitRate (15:8) | BitRate (7:0) | (G)FSK (G)MSK | OOK | Actual BR (b/s) |
|---|-------------------|------------------|------------------|-------------|--------------------|
| Classical modem baud rates (multiples of 1.2 kbps) | 0x68 | 0x2B | 1.2 kbps | 1.2 kbps | 1200.015 |
| | 0x34 | 0x15 | 2.4 kbps | 2.4 kbps | 2400.060 |
| | 0x1A | 0x0B | 4.8 kbps | 4.8 kbps | 4799.760 |
| | 0x0D | 0x05 | 9.6 kbps | 9.6 kbps | 9600.960 |
| | 0x06 | 0x83 | 19.2 kbps | 19.2 kbps | 19196.16 |
| | 0x03 | 0x41 | 38.4 kbps | | 38415.36 |
| | 0x01 | 0xA1 | 76.8 kbps | | 76738.60 |
| | 0x00 | 0xD0 | 153.6 kbps | | 153846.1 |
| Classical modem baud rates (multiples of 0.9 kbps) | 0x02 | 0x2C | 57.6 kbps | | 57553.95 |
| | 0x01 | 0x16 | 115.2 kbps | | 115107.9 |
| Round bit rates (multiples of 12.5, 25 and 50 kbps) | 0x0A | 0x00 | 12.5 kbps | 12.5 kbps | 12500.00 |
| | 0x05 | 0x00 | 25 kbps | 25 kbps | 25000.00 |
| | 0x80 | 0x00 | 50 kbps | | 50000.00 |
| | 0x01 | 0x40 | 100 kbps | | 100000.0 |
| | 0x00 | 0xD5 | 150 kbps | | 150234.7 |
| | 0x00 | 0xA0 | 200 kbps | | 200000.0 |
| | 0x00 | 0x80 | 250 kbps | | 250000.0 |
| | 0x00 | 0x6B | 300 kbps | | 299065.4 |
| Watch Xtal frequency | 0x03 | 0xD1 | 32.768 kbps | 32.768 kbps | 32753.32 |

4.2.2. FSK/OOK Transmission

4.2.2.1. FSK Modulation

FSK modulation is performed inside the PLL bandwidth, by changing the fractional divider ratio in the feedback loop of the PLL. The high resolution of the sigma-delta modulator, allows for very narrow frequency deviation. The frequency deviation F_{DEV} is given by:

$$F_{DEV} = F_{STEP} \times Fdev(13,0)$$

To ensure correct modulation, the following limit applies:

$$F_{DEV} + \frac{BR}{2} \leq (250)\text{kHz}$$

Note No constraint applies to the modulation index of the transmitter, but the frequency deviation must be set between 600 Hz and 200 kHz.

4.2.2.2. OOK Modulation

OOK modulation is applied by switching on and off the power amplifier. Digital control and ramping are available to improve the transient power response of the OOK transmitter.

4.2.2.3. Modulation Shaping

Modulation shaping can be applied in both OOK and FSK modulation modes, to improve the narrow band response of the transmitter. Both shaping features are controlled with *PaRamp* bits in *RegPaRamp*.

- ◆ In FSK mode, a Gaussian filter with BT = 0.5 or 1 is used to filter the modulation stream, at the input of the sigma-delta modulator. If the Gaussian filter is enabled when the SX1276/77/78/79 is in Continuous mode, DCLK signal on pin 10 (DIO1/DCLK) will trigger an interrupt on the uC each time a new bit has to be transmitted. Please refer to section 4.2.12.2 for details.
- ◆ When OOK modulation is used, the PA bias voltages are ramped up and down smoothly when the PA is turned on and off, to reduce spectral splatter.

Note *The transmitter must be restarted if the ModulationShaping setting is changed, in order to recalibrate the built-in filter.*

4.2.3. FSK/OOK Reception

4.2.3.1. FSK Demodulator

The FSK demodulator of the SX1276/77/78/79 is designed to demodulate FSK, GFSK, MSK and GMSK modulated signals. It is most efficient when the modulation index of the signal is greater than 0.5 and below 10:

$$0.5 \leq \beta = \frac{2 \times F_{DEV}}{BR} \leq 10$$

The output of the FSK demodulator can be fed to the Bit Synchronizer to provide the companion processor with a synchronous data stream in Continuous mode.

4.2.3.2. OOK Demodulator

The OOK demodulator performs a comparison of the RSSI output and a threshold value. Three different threshold modes are available, configured through bits *OokThreshType* in *RegOokPeak*.

The recommended mode of operation is the “Peak” threshold mode, illustrated in Figure 14:

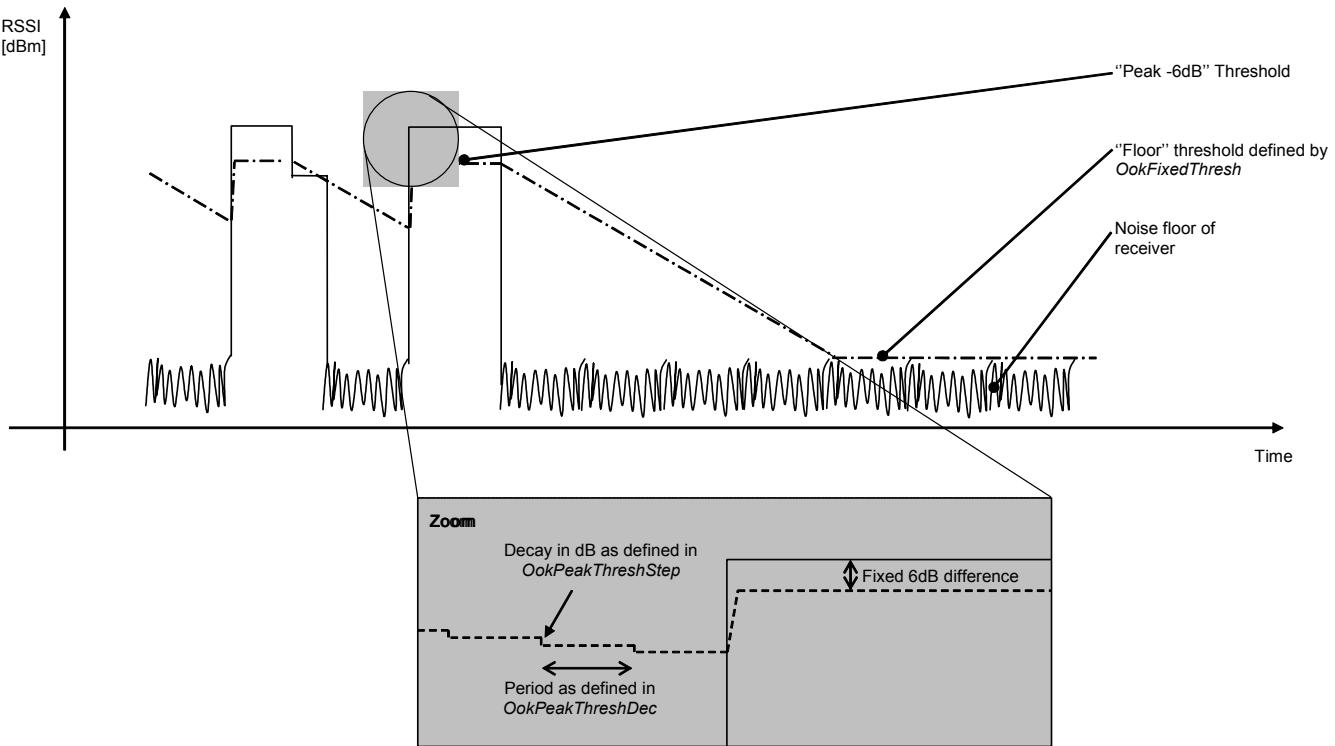


Figure 14. OOK Peak Demodulator Description

In peak threshold mode the comparison threshold level is the peak value of the RSSI, reduced by 6dB. In the absence of an input signal, or during the reception of a logical ‘0’, the acquired peak value is decremented by one *OokPeakThreshStep* every *OokPeakThreshDec* period.

When the RSSI output is null for a long time (for instance after a long string of “0” received, or if no transmitter is present), the peak threshold level will continue falling until it reaches the “Floor Threshold”, programmed in *OokFixedThresh*.

The default settings of the OOK demodulator lead to the performance stated in the electrical specification. However, in applications in which sudden signal drops are awaited during a reception, the three parameters should be optimized accordingly.

Optimizing the Floor Threshold

OokFixedThresh determines the sensitivity of the OOK receiver, as it sets the comparison threshold for weak input signals (i.e. those close to the noise floor). Significant sensitivity improvements can be generated if configured correctly.

Note that the noise floor of the receiver at the demodulator input depends on:

- ◆ The noise figure of the receiver.
- ◆ The gain of the receive chain from antenna to base band.
- ◆ The matching - including SAW filter if any.
- ◆ The bandwidth of the channel filters.

It is therefore important to note that the setting of *OokFixedThresh* will be application dependent. The following procedure is recommended to optimize *OokFixedThresh*.

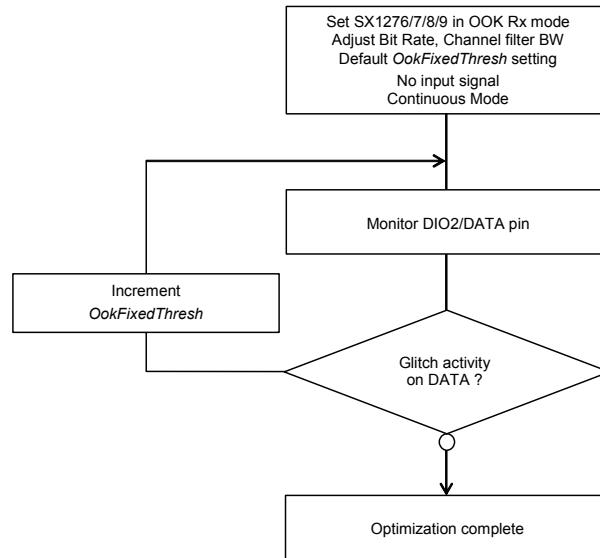


Figure 15. Floor Threshold Optimization

The new floor threshold value found during this test should be used for OOK reception with those receiver settings.

Optimizing OOK Demodulator for Fast Fading Signals

A sudden drop in signal strength can cause the bit error rate to increase. For applications where the expected signal drop can be estimated, the following OOK demodulator parameters *OokPeakThreshStep* and *OokPeakThreshDec* can be optimized as described below for a given number of threshold decrements per bit. Refer to *RegOokPeak* to access those settings.

Alternative OOK Demodulator Threshold Modes

In addition to the Peak OOK threshold mode, the user can alternatively select two other types of threshold detectors:

- ◆ Fixed Threshold: The value is selected through *OokFixedThresh*
- ◆ Average Threshold: Data supplied by the RSSI block is averaged, and this operation mode should only be used with DC-free encoded data.

4.2.3.3. Bit Synchronizer

The bit synchronizer provides a clean and synchronized digital output based upon timing recovery information gleaned from the received data edge transitions. Its output is made available on pin DIO1/DCLK in Continuous mode and can be disabled through register settings. However, for optimum receiver performance, especially in Continuous receive mode, its use is strongly advised.

The Bit Synchronizer is automatically activated in Packet mode. Its bit rate is controlled by *BitRateMsb* and *BitRateLsb* in *RegBitrate*.

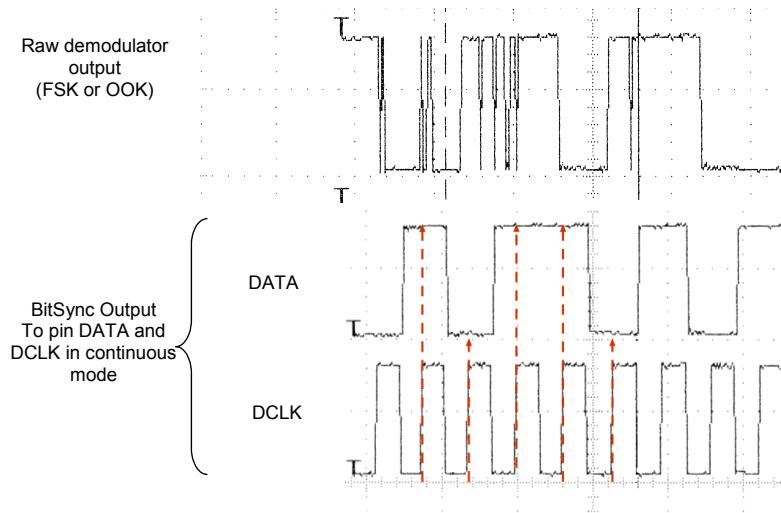


Figure 16. Bit Synchronizer Description

To ensure correct operation of the Bit Synchronizer, the following conditions have to be satisfied:

- ◆ A preamble (0x55 or 0xAA) of at least 12 bits is required for synchronization, the longer the synchronization phase is the better the ensuing packet detection rate will be.
- ◆ The subsequent payload bit stream must have at least one edge transition (either rising or falling) every 16 bits during data transmission.
- ◆ The absolute error between transmitted and received bit rate must not exceed 6.5%.

4.2.3.4. Frequency Error Indicator

This frequency error indicator measures the frequency error between the programmed RF centre frequency and the carrier frequency of the modulated input signal to the receiver. When the FEI is performed, the frequency error is measured and the signed result is loaded in *FeiValue* in *RegFei*, in 2's complement format. The time required for an FEI evaluation is 4 bit periods.

To ensure correct operation of the FEI:

- ◆ The measurement must be launched during the reception of preamble.
- ◆ The sum of the frequency offset and the 20 dB signal bandwidth must be lower than the base band filter bandwidth. i.e. The whole modulated spectrum must be received.

The 20 dB bandwidth of the signal can be evaluated as follows (double-side bandwidth):

$$BW_{20dB} = 2 \times \left(F_{DEV} + \frac{BR}{2} \right)$$

The frequency error, in Hz, can be calculated with the following formula:

$$FEI = F_{STEP} \times FeiValue$$

The FEI is enabled automatically upon the transition to receive mode and automatically updated every 4 bits.

4.2.3.5. AFC

The AFC is based on the FEI measurement, therefore the same input signal and receiver setting conditions apply. When the AFC procedure is performed the *AfcValue* is directly subtracted from the register that defines the frequency of operation of the chip, F_{RF} . The AFC is executed each time the receiver is enabled, if *AfcAutoOn* = 1.

When the AFC is enabled (*AfcAutoOn* = 1), the user has the option to:

- ◆ Clear the former AFC correction value, if *AfcAutoClearOn* = 1. Allowing the next frequency correction to be performed from the initial centre frequency.
- ◆ Start the AFC evaluation from the previously corrected frequency. This may be useful in systems in which the centre frequency experiences cumulative drift - such as the ageing of a crystal reference.

The SX1276/77/78/79 offers an alternate receiver bandwidth setting during the AFC phase allowing the accommodation of larger frequency errors. The setting *RegAfcBw* sets the receive bandwidth during the AFC process. In a typical receiver application the, once the AFC is performed, the radio will revert to the receiver communication or channel bandwidth (*RegRxBw*) for the ensuing communication phase.

Note that the FEI measurement is valid only during the reception of preamble. The provision of the *PreambleDetect* flag can hence be used to detect this condition and allow a reliable AFC or FEI operation to be triggered. This process can be performed automatically by using the appropriate options in *StartDemodOnPreamble* found in the *RegRxConfig* register.

A detailed description of the receiver setup to enable the AFC is provided in section 4.2.6.

4.2.3.6. Preamble Detector

The Preamble Detector indicates the reception of a carrier modulated with a 0101...sequence. It is insensitive to the frequency offset, as long as the receiver bandwidth is large enough. The size of detection can be programmed from 1 to 3 bytes with *PreambleDetectorSize* in *RegPreambleDetect* as defined in the next table.

Table 20 Preamble Detector Settings

| PreambleDetectorSize | # of Bytes |
|-----------------------------|-------------------|
| 00 | 1 |
| 01 | 2 (recommended) |
| 10 | 3 |
| 11 | reserved |

For normal operation, *PreambleDetectTol* should be set to be set to 10 (0x0A), with a qualifying preamble size of 2 bytes.

The *PreambleDetect* interrupt (either in *RegIrqFlags1* or mapped to a specific DIO) then goes high every time a valid preamble is detected, assuming *PreambleDetectorOn*=1.

The preamble detector can also be used as a gate to ensure that AFC and AGC are performed on valid preamble. See section 4.2.6. for details.

4.2.3.7. Image Rejection Mixer

The SX1276/77/78/79 employs an image rejection mixer (IRM) which, uncalibrated, 35 dB image rejection. A low phase noise PLL is used to perform calibration of the receiver chain. This increases the typical image rejection to 48 dB.

4.2.3.8. Image and RSSI Calibration

An automated process is implemented to calibrate the phase and gain imbalances of I and Q receive paths. This calibration enhances image rejection and improves RSSI precision. It is launched under the following circumstances:

- ◆ Automatically at Power On Reset or after a Manual Reset of the chip (refer to section 7.2), only for the Low Frequency front-end, and is performed at 434MHz
- ◆ Automatically when a pre-defined temperature change is observed, if the option is enabled. A selectable temperature change, set with *TempThreshold* (5, 10, 15 or 20°C), is detected and reported in *TempChange*, if the temperature monitoring is turned On with *TempMonitorOff*=0. This interrupt flag can be used by the application to launch a new image calibration at a convenient time if *AutoImageCalOn*=0, or immediately when this temperature variation is detected, if *AutoImageCalOn*=1
- ◆ Upon user request, by setting bit *ImageCalStart* in *RegImageCal*, when the device is in Standby mode

Notes

- The calibration procedure takes approximately 10ms. It is recommended to disable the fully automated (temperature-dependent) calibration, to better control when it is triggered (and avoid unexpected packet losses)
- To perform the calibration, the radio must be temporarily returned to FSK/OOK mode
- The automatic IQ and RSSI calibration done at POR and Reset is only valid at 434 MHz (the value of *RegFrF* at POR). To improve accuracy of RSSI and image rejection, this calibration should be replicated at the frequency (ies)

of interest, for instance a calibration should be launched with *FrF* set to 868.3 MHz if the high frequency port supports communication in this frequency band. Conversely if the product is used at 169 MHz, the calibration should be repeated with *FrF*=169MHz

- FormerTemp and TempChange in SX1276/77/79 are frequency-specific and the IC keeps a copy of these variables when switching between the low frequency and the high frequency domains (along with the corresponding calibration values, stored in test registers)
- FormerTemp and TempChange cannot be read in Sleep mode (although they are saved). They should be read in Standby mode

4.2.3.9. Timeout Function

The SX1276/77/78/79 includes a Timeout function, which allows it to automatically shut-down the receiver after a receive sequence and therefore save energy.

- ◆ Timeout interrupt is generated *TimeoutRxRssi* x 16 x *Tbit* after switching to Rx mode if the *Rssi* flag does not raise within this time frame (*RssiValue* > *RssiThreshold*)
- ◆ Timeout interrupt is generated *TimeoutRxPreamble* x 16 x *Tbit* after switching to Rx mode if the *PreambleDetect* flag does not raise within this time frame
- ◆ Timeout interrupt is generated *TimeoutSignalSync* x 16 x *Tbit* after switching to Rx mode if the *SyncAddress* flag does not raise within this time frame

This timeout interrupt can be used to warn the companion processor to shut down the receiver and return to a lower power mode. To become active, these timeouts must also be enabled by setting the correct *RxTrigger* parameters in *RegRxConfig*:

Table 21 RxTrigger Settings to Enable Timeout Interrupts

| Receiver Triggering Event | RxTrigger (2:0) | Timeout on Rssi | Timeout on Preamble | Timeout on SyncAddress |
|----------------------------------|------------------------|------------------------|----------------------------|-------------------------------|
| None | 000 | Off | Off | Active |
| Rssi Interrupt | 001 | Active | Off | |
| PreambleDetect | 110 | Off | Active | |
| Rssi Interrupt & PreambleDetect | 111 | Active | Active | |

4.2.4. Operating Modes in FSK/OOK Mode

The SX1276/77/78/79 has several working modes, manually programmed in *RegOpMode*. Fully automated mode selection, packet transmission and reception is also possible using the Top Level Sequencer described in Section 4.2.8.

Table 22 Basic Transceiver Modes

| Mode | Selected mode | Symbol | Enabled blocks |
|-------------|---------------------------------------|---------------|---|
| 000 | Sleep mode | Sleep | None |
| 001 | Standby mode | Stdby | Top regulator and crystal oscillator |
| 010 | Frequency synthesiser to Tx frequency | FSTx | Frequency synthesizer at Tx frequency (FrF) |
| 011 | Transmit mode | Tx | Frequency synthesizer and transmitter |

| Mode | Selected mode | Symbol | Enabled blocks |
|------|---------------------------------------|--------|--|
| 100 | Frequency synthesiser to Rx frequency | FSRx | Frequency synthesizer at frequency for reception (Fr-IF) |
| 101 | Receive mode | Rx | Frequency synthesizer and receiver |

When switching from a mode to another the sub-blocks are woken up according to a pre-defined optimized sequence.

4.2.5. Startup Times

The startup time of the transmitter or the receiver is Dependant upon which mode the transceiver was in at the beginning. For a complete description, Figure 17 below shows a complete startup process, from the lower power mode "Sleep".

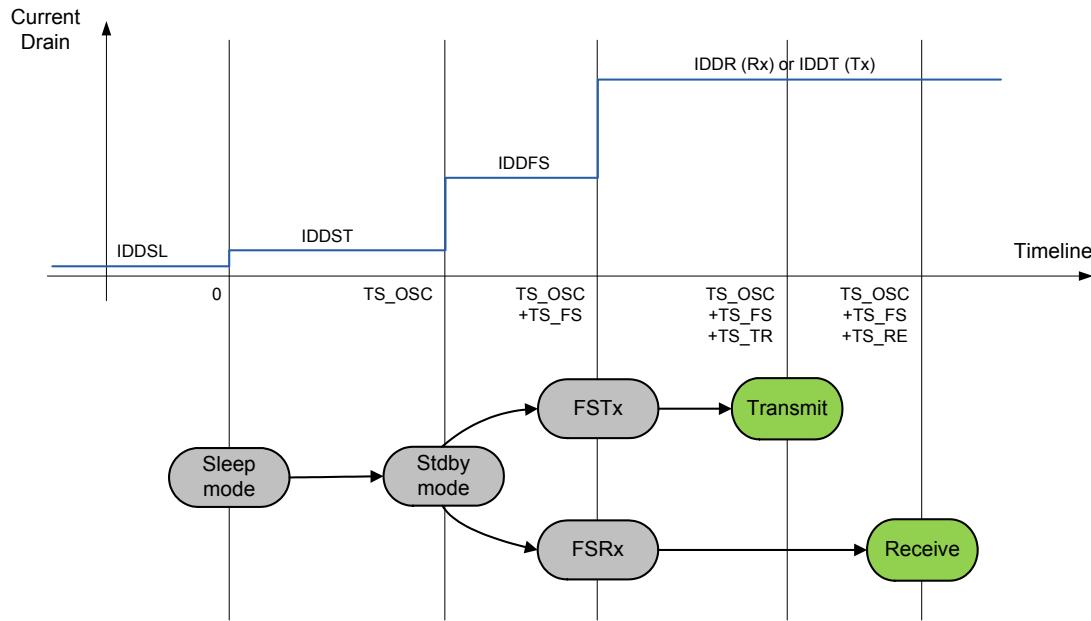


Figure 17. Startup Process

TS_OSC is the startup time of the crystal oscillator which depends on the electrical characteristics of the crystal. TS_FS is the startup time of the PLL including systematic calibration of the VCO.

Typical values of TS_OSC and TS_FS are given in Section 2.5.2.

4.2.5.1. Transmitter Startup Time

The transmitter startup time, TS_TR, is calculated as follows in FSK mode:

$$TS_TR = 5\mu s + 1.25 \times PaRamp + \frac{1}{2} \times Tbit ,$$

where *PaRamp* is the ramp-up time programmed in *RegPaRamp* and *Tbit* is the bit time.

In OOK mode, this equation can be simplified to the following:

$$TS_TR = 5\mu s + \frac{1}{2} \times Tbit$$

4.2.5.2. Receiver Startup Time

The receiver startup time, TS_RE, only depends upon the receiver bandwidth effective at the time of startup. When AFC is enabled (*AfcAutoOn*=1), *AfcBw* should be used instead of *RxBw* to extract the receiver startup time:

Table 23 Receiver Startup Time Summary

| <i>RxBw if AfcAutoOn=0 RxBwAfc if AfcAutoOn=1</i> | <i>TS_RE (+/-5%)</i> |
|---|--------------------------|
| 2.6 kHz | 2.33 ms |
| 3.1 kHz | 1.94 ms |
| 3.9 kHz | 1.56 ms |
| 5.2 kHz | 1.18 ms |
| 6.3 kHz | 984 us |
| 7.8 kHz | 791 us |
| 10.4 kHz | 601 us |
| 12.5 kHz | 504 us |
| 15.6 kHz | 407 us |
| 20.8 kHz | 313 us |
| 25.0 kHz | 264 us |
| 31.3 kHz | 215 us |
| 41.7 kHz | 169 us |
| 50.0 kHz | 144 us |
| 62.5 kHz | 119 us |
| 83.3 kHz | 97 us |
| 100.0 kHz | 84 us |
| 125.0 kHz | 71 us |
| 166.7 kHz | 85 us |
| 200.0 kHz | 74 us |
| 250.0 kHz | 63 us |

TS_{RE} or later after setting the device in Receive mode, any incoming packet will be detected and demodulated by the transceiver.

4.2.5.3. Time to RSSI Evaluation

The first RSSI sample will be available TS_{RSSI} after the receiver is ready, in other words TS_{RE} + TS_{RSSI} after the receiver was requested to turn on.

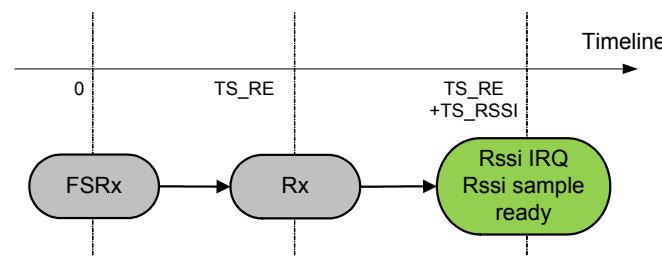


Figure 18. Time to RSSI Sample

TS_{RSSI} depends on the receiver bandwidth, as well as the *RssiSmoothing* option that was selected. The formula used to calculate TS_{RSSI} is provided in section 5.5.4.

4.2.5.4. Tx to Rx Turnaround Time

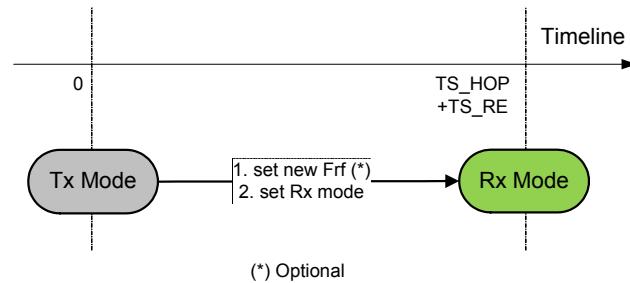


Figure 19. Tx to Rx Turnaround

Note The SPI instruction times are omitted, as they can generally be very small as compared to other timings (up to 10MHz SPI clock).

4.2.5.5. Rx to Tx

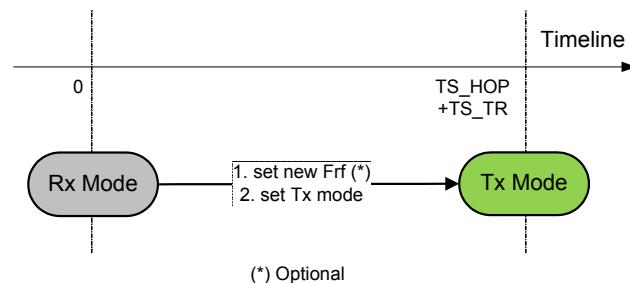


Figure 20. Rx to Tx Turnaround

4.2.5.6. Receiver Hopping, Rx to Rx

Two methods are possible:

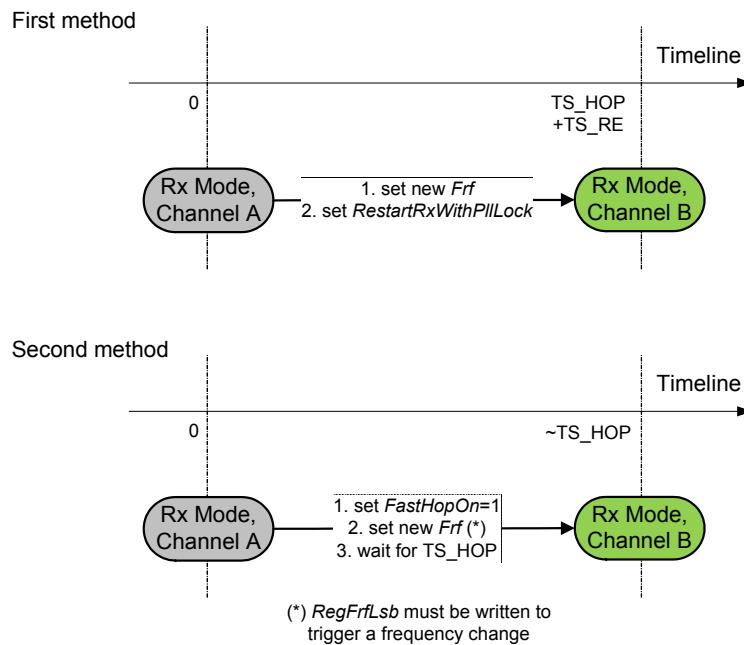


Figure 21. Receiver Hopping

The second method is quicker, and should be used if a very quick RF sniffing mechanism is to be implemented.

4.2.5.7. Tx to Tx

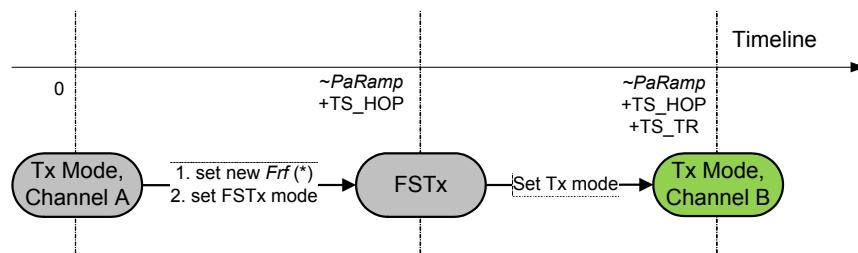


Figure 22. Transmitter Hopping

4.2.6. Receiver Startup Options

The SX1276/77/78/79 receiver can automatically control the gain of the receive chain (AGC) and adjust the receiver LO frequency (AFC). Those processes are carried out on a packet-by-packet basis. They occur:

- ◆ When the receiver is turned On.
- ◆ When the Receiver is restarted upon user request, through the use of trigger bits *RestartRxWithoutPllLock* or *RestartRxWithPllLock*, in *RegRxConfig*.
- ◆ When the receiver is automatically restarted after the reception of a valid packet, or after a packet collision.

Automatic restart capabilities are detailed in Section 4.2.7.

The receiver startup options available in SX1276/77/78/79 are described in Table 24.

Table 24 Receiver Startup Options

| Triggering Event | Realized Function | AgcAutoOn | AfcAutoOn | RxTrigger (2:0) |
|---|--------------------------|------------------|------------------|------------------------|
| None | None | 0 | 0 | 000 |
| <i>Rssi Interrupt</i> | AGC | 1 | 0 | 001 |
| | AGC & AFC | 1 | 1 | 001 |
| <i>PreambleDetect</i> | AGC | 1 | 0 | 110 |
| | AGC & AFC | 1 | 1 | 110 |
| <i>Rssi Interrupt</i> & <i>PreambleDetect</i> | AGC | 1 | 0 | 111 |
| | AGC & AFC | 1 | 1 | 111 |

When *AgcAutoOn*=0, the LNA gain is manually selected by choosing *LnaGain* bits in *RegLna*.

4.2.7. Receiver Restart Methods

The options for restart of the receiver are covered below. This is typically of use to prepare for the reception of a new signal whose strength or carrier frequency is different from the preceding packet to allow the AGC or AFC to be re-evaluated.

4.2.7.1. Restart Upon User Request

In Receive mode the user can request a receiver restart - this can be useful in conjunction with the use of a Timeout interrupt following a period of inactivity in the channel of interest. Two options are available:

- ◆ No change in the Local Oscillator upon restart: the AFC is disabled, and the *FrF* register has not been changed through SPI before the restart instruction: set bit *RestartRxWithoutPllLock* in *RegRxConfig* to 1.
- ◆ Local Oscillator change upon restart: if AFC is enabled (*AfcAutoOn*=1), and/or the *FrF* register had been changed during the last Rx period: set bit *RestartRxWithPllLock* in *RegRxConfig* to 1.

Note *ModeReady* must be at logic level 1 for a new *RestartRx* command to be taken into account.

4.2.7.2. Automatic Restart after valid Packet Reception

The bits *AutoRestartRxMode* in *RegSyncConfig* control the automatic restart feature of the SX1276/77/78/79 receiver, when a valid packet has been received:

- ◆ If *AutoRestartRxMode* = 00, the function is off, and the user should manually restart the receiver upon valid packet reception (see section 4.2.7.1).
- ◆ If *AutoRestartRxMode* = 01, after the user has emptied the FIFO following a *PayloadReady* interrupt, the receiver will automatically restart itself after a delay of *InterPacketRxDelay*, allowing for the distant transmitter to ramp down, hence avoiding a false RSSI detection on the ‘tail’ of the previous packet.
- ◆ If *AutoRestartRxMode* = 10 should be used if the next reception is expected on a new frequency, i.e. *FrF* is changed after the reception of the previous packet. An additional delay is systematically added, in order for the PLL to lock at a new frequency.

4.2.7.3. Automatic Restart when Packet Collision is Detected

In receive mode the SX1276/77/78/79 is able to detect packet collision and restart the receiver. Collisions are detected by a sudden rise in received signal strength, detected by the RSSI. This functionality can be useful in network configurations where many asynchronous slaves attempt periodic communication with a single master node.

The collision detector is enabled by setting bit *RestartRxOnCollision* to 1.

The decision to restart the receiver is based on the detection of RSSI change. The sensitivity of the system can be adjusted in 1 dB steps by using register *RssiCollisionThreshold* in *RegRxConfig*.

4.2.8. Top Level Sequencer

Depending on the application, it is desirable to be able to change the mode of the circuit according to a predefined sequence without access to the serial interface. In order to define different sequences or scenarios, a user-programmable state machine, called Top Level Sequencer (Sequencer in short), can automatically control the chip modes.

NOTE THAT THIS FUNCTIONALITY IS ONLY AVAILABLE IN FSK/OOK MODE.

The Sequencer is activated by setting the *SequencerStart* bit in *RegSeqConfig1* to 1 in Sleep or Standby mode (called initial mode).

It is also possible to force the Sequencer off by setting the *Stop* bit in *RegSeqConfig1* to 1 at any time.

Note *SequencerStart and Stop bit must never be set at the same time.*

4.2.8.1. Sequencer States

As shown in the table below, with the aid of a pair of interrupt timers (T1 and T2), the sequencer can take control of the chip operation in all modes.

Table 25 Sequencer States

| Sequencer State | Description |
|---------------------------|---|
| SequencerOff State | The Sequencer is not activated. Sending a <i>SequencerStart</i> command will launch it. When coming from LowPowerSelection state, the Sequencer will be Off, whilst the chip will return to its initial mode (either Sleep or Standby mode). |
| Idle State | The chip is in low-power mode, either <i>Standby</i> or <i>Sleep</i> , as defined by <i>IdleMode</i> in <i>RegSeqConfig1</i> . The Sequencer waits only for the <i>T1</i> interrupt. |
| Transmit State | The transmitter is on. |
| Receive State | The receiver is on. |
| PacketReceived | The receiver is on and a packet has been received. It is stored in the FIFO. |
| LowPowerSelection | Selects low power state (SequencerOff or Idle State) |
| RxTimeout | Defines the action to be taken on a RxTimeout interrupt. RxTimeout interrupt can be a <i>TimeoutRxRssi</i> , <i>TimeoutRxPreamble</i> or <i>TimeoutSignalSync</i> interrupt. |

4.2.8.2. Sequencer Transitions

The transitions between sequencer states are listed in the forthcoming table.

Table 26 Sequencer Transition Options

| Variable | Transition |
|---------------------------|--|
| <i>IdleMode</i> | Selects the chip mode during Idle state: 0: Standby mode 1: Sleep mode |
| <i>FromStart</i> | Controls the Sequencer transition when the SequencerStart bit is set to 1 in Sleep or Standby mode: 00: to LowPowerSelection 01: to Receive state 10: to Transmit state 11: to Transmit state on a <i>FifoThreshold</i> interrupt |
| <i>LowPowerSelection</i> | Selects Sequencer LowPower state after a <i>to LowPowerSelection</i> transition 0: SequencerOff state with chip on Initial mode 1: Idle state with chip on Standby or Sleep mode depending on IdleMode Note: Initial mode is the chip LowPower mode at Sequencer start. |
| <i>FromIdle</i> | Controls the Sequencer transition from the Idle state on a <i>T1</i> interrupt: 0: to Transmit state 1: to Receive state |
| <i>FromTransmit</i> | Controls the Sequencer transition from the Transmit state: 0: to LowPowerSelection on a <i>PacketSent</i> interrupt 1: to Receive state on a <i>PacketSent</i> interrupt |
| <i>FromReceive</i> | Controls the Sequencer transition from the Receive state: 000 and 111: unused 001: to PacketReceived state on a <i>PayloadReady</i> interrupt 010: to LowPowerSelection on a <i>PayloadReady</i> interrupt 011: to PacketReceived state on a <i>CrcOk</i> interrupt. If CRC is wrong (corrupted packet, with CRC on but <i>CrcAutoClearOn</i> is off), the <i>PayloadReady</i> interrupt will drive the sequencer to <i>RxTimeout</i> state. 100: to SequencerOff state on a <i>Rssi</i> interrupt 101: to SequencerOff state on a <i>SyncAddress</i> interrupt 110: to SequencerOff state on a <i>PreambleDetect</i> interrupt Irrespective of this setting, transition to LowPowerSelection on a <i>T2</i> interrupt |
| <i>FromRxTimeout</i> | Controls the state-machine transition from the Receive state on a <i>RxTimeout</i> interrupt (and on <i>PayloadReady</i> if <i>FromReceive</i> = 011): 00: to Receive state via <i>ReceiveRestart</i> 01: to Transmit state 10: to LowPowerSelection 11: to SequencerOff state Note: RxTimeout interrupt is a <i>TimeoutRxRssi</i> , <i>TimeoutRxPreamble</i> or <i>TimeoutSignalSync</i> interrupt. |
| <i>FromPacketReceived</i> | Controls the state-machine transition from the PacketReceived state: 000: to SequencerOff state 001: to Transmit on a <i>FifoEmpty</i> interrupt 010: to LowPowerSelection 011: to Receive via <i>FS</i> mode, if frequency was changed 100: to Receive state (no frequency change) |

4.2.8.3. Timers

Two timers (Timer1 and Timer2) are also available in order to define periodic sequences. These timers are used to generate interrupts, which can trigger transitions of the Sequencer.

T1 interrupt is generated (Timer1Resolution * Timer1Coefficient) after **T2 interrupt** or **SequencerStart**. command.

T2 interrupt is generated (Timer2Resolution * Timer2Coefficient) after **T1 interrupt**.

The timers' mechanism is summarized on the following diagram.

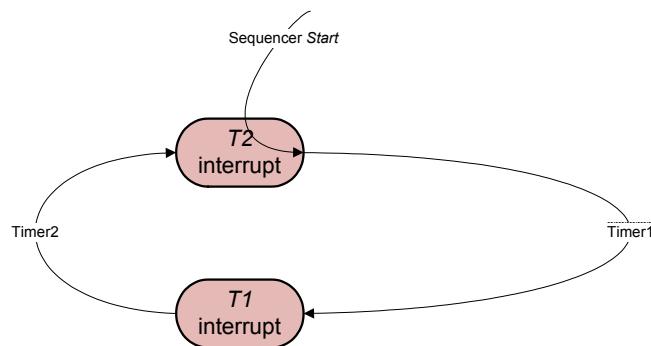


Figure 23. Timer1 and Timer2 Mechanism

Note *The timer sequence is completed independently of the actual Sequencer state. Thus, both timers need to be on to achieve periodic cycling.*

Table 27 Sequencer Timer Settings

| Variable | Description |
|-------------------|---|
| Timer1Resolution | Resolution of Timer1 00: disabled 01: 64 us 10: 4.1 ms 11: 262 ms |
| Timer2Resolution | Resolution of Timer2 00: disabled 01: 64 us 10: 4.1 ms 11: 262 ms |
| Timer1Coefficient | Multiplying coefficient for Timer1 |
| Timer2Coefficient | Multiplying coefficient for Timer2 |

4.2.8.4. Sequencer State Machine

The following graphs summarize every possible transition between each Sequencer state. The Sequencer states are highlighted in grey. The transitions are represented by arrows. The condition activating them is described over the transition arrow. For better readability, the start transitions are separated from the rest of the graph.

Transitory states are highlighted in light grey, and exit states are represented in red. It is also possible to force the Sequencer off by setting the *Stop* bit in *RegSeqConfig1* to 1 at any time.

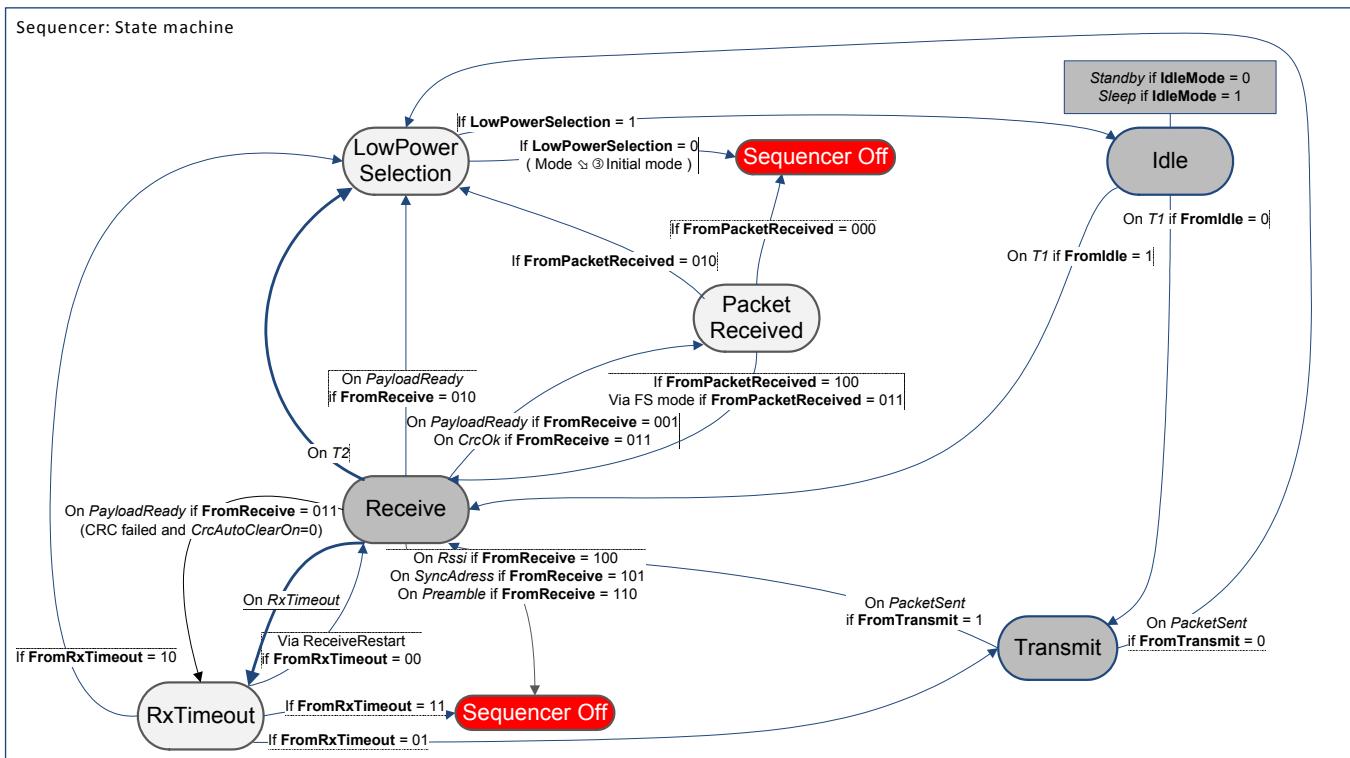
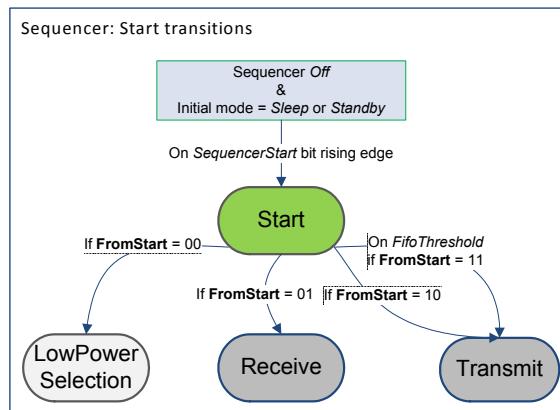


Figure 24. Sequencer State Machine

4.2.9. Data Processing in FSK/OOK Mode

4.2.9.1. Block Diagram

Figure below illustrates the SX1276/77/78/79 data processing circuit. Its role is to interface the data to/from the modulator/demodulator and the uC access points (SPI and DIO pins). It also controls all the configuration registers.

The circuit contains several control blocks which are described in the following paragraphs.

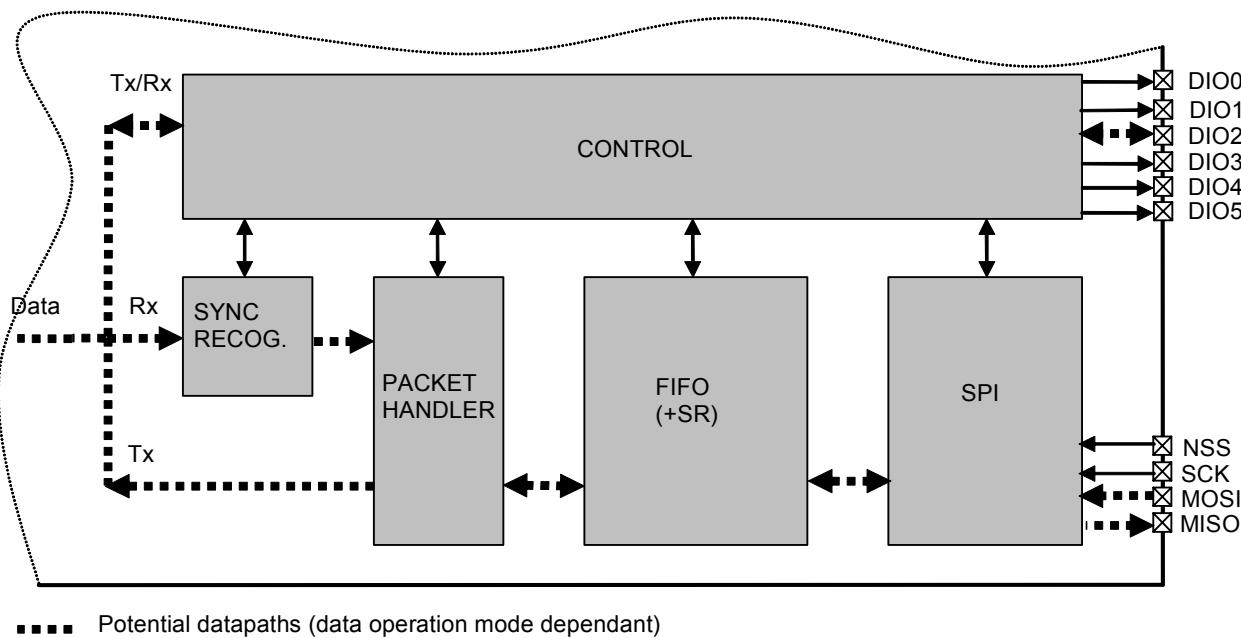


Figure 25. SX1276/77/78/79 Data Processing Conceptual View

The SX1276/77/78/79 implements several data operation modes, each with their own data path through the data processing. Depending on the data operation mode selected, some control blocks are active whilst others remain disabled.

4.2.9.2. Data Operation Modes

The SX1276/77/78/79 has two different data operation modes selectable by the user:

- ◆ Continuous mode: each bit transmitted or received is accessed in real time at the DIO2/DATA pin. This mode may be used if adequate external signal processing is available.
- ◆ Packet mode (recommended): user only provides/retrieves payload bytes to/from the FIFO. The packet is automatically built with preamble, Sync word, and optional CRC and DC-free encoding schemes. The reverse operation is performed in reception. The uC processing overhead is hence significantly reduced compared to Continuous mode. Depending on the optional features activated (CRC, etc) the maximum payload length is limited to 255, 2047 bytes or unlimited.

Each of these data operation modes is fully described in the following s.

4.2.10. FIFO

Overview and Shift Register (SR)

In packet mode of operation, both data to be transmitted and that has been received are stored in a configurable FIFO (First In First Out) device. It is accessed via the SPI interface and provides several interrupts for transfer management.

The FIFO is 1 byte wide hence it only performs byte (parallel) operations, whereas the demodulator functions serially. A shift register is therefore employed to interface the two devices. In transmit mode it takes bytes from the FIFO and outputs them serially (MSB first) at the programmed bit rate to the modulator. Similarly, in Rx the shift register gets bit by bit data from the demodulator and writes them byte by byte to the FIFO. This is illustrated in figure below.

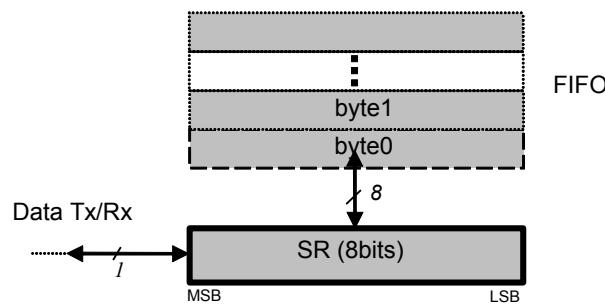


Figure 26. FIFO and Shift Register (SR)

Note When switching to Sleep mode, the FIFO can only be used once the ModeReady flag is set (quasi immediate from all modes except from Tx)

The FIFO size is fixed to 64 bytes.

Interrupt Sources and Flags

- ◆ *FifoEmpty*: *FifoEmpty* interrupt source is high when byte 0, i.e. whole FIFO, is empty. Otherwise it is low. Note that when retrieving data from the FIFO, *FifoEmpty* is updated on NSS falling edge, i.e. when *FifoEmpty* is updated to low state the currently started read operation must be completed. In other words, *FifoEmpty* state must be checked after each read operation for a decision on the next one (*FifoEmpty* = 0: more byte(s) to read; *FifoEmpty* = 1: no more byte to read).
- ◆ *FifoFull*: *FifoFull* interrupt source is high when the last FIFO byte, i.e. the whole FIFO, is full. Otherwise it is low.
- ◆ *FifoOverrunFlag*: *FifoOverrunFlag* is set when a new byte is written by the user (in Tx or Standby modes) or the SR (in Rx mode) while the FIFO is already full. Data is lost and the flag should be cleared by writing a 1, note that the FIFO will also be cleared.
- ◆ *PacketSent*: *PacketSent* interrupt source goes high when the SR's last bit has been sent.
- ◆ *FifoLevel*: Threshold can be programmed by *FifoThreshold* in *RegFifoThresh*. Its behavior is illustrated in figure below.

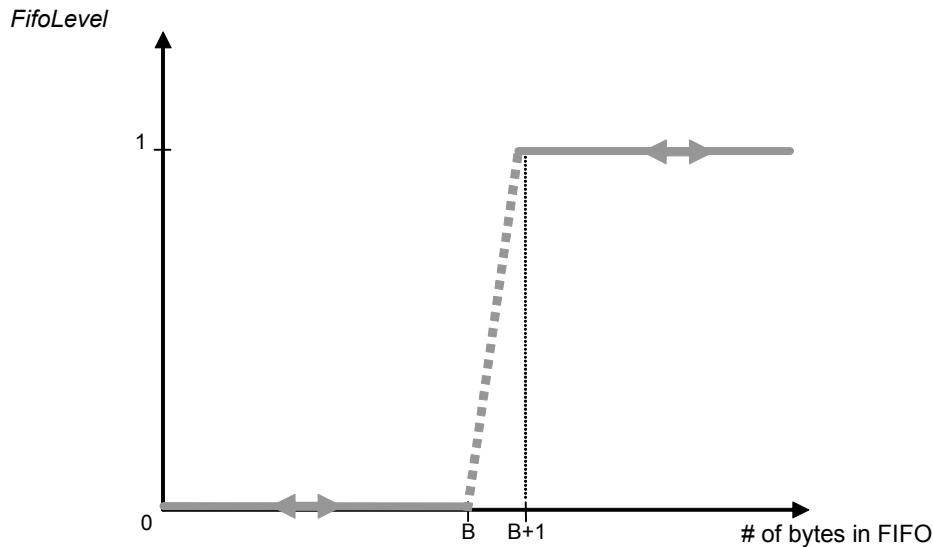


Figure 27. FifoLevel IRQ Source Behavior

Notes - *FifoLevel interrupt is updated only after a read or write operation on the FIFO. Thus the interrupt cannot be dynamically updated by only changing the FifoThreshold parameter*

- *FifoLevel interrupt is valid as long as FifoFull does not occur. An empty FIFO will restore its normal operation*

FIFO Clearing

Table below summarizes the status of the FIFO when switching between different modes

Table 28 Status of FIFO when Switching Between Different Modes of the Chip

| From | To | FIFO status | Comments |
|-------------|-------------|-------------|--|
| Stdby | Sleep | Not cleared | |
| Sleep | Stdby | Not cleared | |
| Stdby/Sleep | Tx | Not cleared | To allow the user to write the FIFO in Stdby/Sleep before Tx |
| Stdby/Sleep | Rx | Cleared | |
| Rx | Tx | Cleared | |
| Rx | Stdby/Sleep | Not cleared | To allow the user to read FIFO in Stdby/Sleep mode after Rx |
| Tx | Any | Cleared | |

4.2.10.1. Sync Word Recognition

Overview

Sync word recognition (also called Pattern recognition) is activated by setting *SyncOn* in *RegSyncConfig*. The bit synchronizer must also be activated in Continuous mode (automatically done in Packet mode).

The block behaves like a shift register; it continuously compares the incoming data with its internally programmed Sync word and sets *SyncAddressMatch* when a match is detected. This is illustrated in Figure 28 below.

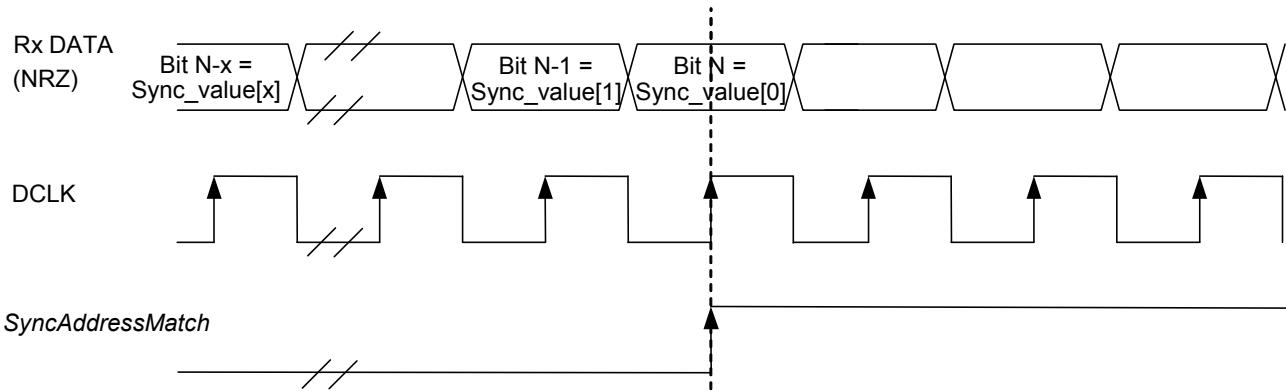


Figure 28. Sync Word Recognition

During the comparison of the demodulated data, the first bit received is compared with bit 7 (MSB) of *RegSyncValue1* and the last bit received is compared with bit 0 (LSB) of the last byte whose address is determined by the length of the Sync word.

When the programmed Sync word is detected the user can assume that this incoming packet is for the node and can be processed accordingly.

SyncAddressMatch is cleared when leaving Rx or FIFO is emptied.

Configuration

- ◆ Size: Sync word size can be set from 1 to 8 bytes (i.e. 8 to 64 bits) via *SyncSize* in *RegSyncConfig*. In Packet mode this field is also used for Sync word generation in Tx mode.
- ◆ Value: The Sync word value is configured in *SyncValue(63:0)*. In Packet mode this field is also used for Sync word generation in Tx mode.

Note *SyncValue choices containing 0x00 bytes are not allowed*

Packet Handler

The packet handler is the block used in Packet mode. Its functionality is fully described in Section 4.2.13.

Control

The control block configures and controls the full chip's behavior according to the settings programmed in the configuration registers.

4.2.11. Digital IO Pins Mapping

Six general purpose IO pins are available on the SX1276/77/78/79, and their configuration in Continuous or Packet mode is controlled through *RegDioMapping1* and *RegDioMapping2*.

Table 29 DIO Mapping, Continuous Mode

| | DIOx Mapping | Sleep | Standby | FSRx/Tx | Rx | Tx |
|------|--------------|--------------|---------|---------------------|-----------------------|---------|
| DIO0 | 00 | | - | | SyncAddress | TxReady |
| | 01 | | - | | Rssi / PreambleDetect | - |
| | 10 | | - | | RxReady | TxReady |
| | 11 | | | | | |
| DIO1 | 00 | | - | | Dclk | - |
| | 01 | | - | | Rssi / PreambleDetect | - |
| | 10 | | | | | |
| | 11 | | | | | |
| DIO2 | 00 | | - | | Data | |
| | 01 | | - | | Data | |
| | 10 | | - | | Data | |
| | 11 | | - | | Data | |
| DIO3 | 00 | | - | | Timeout | - |
| | 01 | | - | | Rssi / PreambleDetect | - |
| | 10 | | | | | |
| | 11 | - | | TempChange / LowBat | TempChange / LowBat | |
| DIO4 | 00 | | - | | TempChange / LowBat | |
| | 01 | | - | | PllLock | |
| | 10 | | | | TimeOut | - |
| | 11 | - | | ModeReady | ModeReady | |
| DIO5 | 00 | ClkOut if RC | | ClkOut | | ClkOut |
| | 01 | | - | | PllLock | |
| | 10 | | | | Rssi / PreambleDetect | - |
| | 11 | - | | ModeReady | ModeReady | |

Table 30 DIO Mapping, Packet Mode

| | DIOx Mapping | Sleep | Standby | FSRx/Tx | Rx | Tx |
|------|--------------|--------------|-----------|---------------------|-----------------------|------------|
| DIO0 | 00 | | - | | PayloadReady | PacketSent |
| | 01 | | - | | CrcOk | - |
| | 10 | | | | | |
| | 11 | - | | TempChange / LowBat | TempChange / LowBat | |
| DIO1 | 00 | FifoLevel | FifoLevel | | FifoLevel | |
| | 01 | FifoEmpty | FifoEmpty | | FifoEmpty | |
| | 10 | FifoFull | FifoFull | | FifoFull | |
| | 11 | | - | | | |
| DIO2 | 00 | FifoFull | FifoFull | | FifoFull | |
| | 01 | | | | RxReady | - |
| | 10 | FifoFull | | | TimeOut | FifoFull |
| | 11 | FifoFull | | | SyncAddress | FifoFull |
| DIO3 | 00 | FifoEmpty | FifoEmpty | | FifoEmpty | |
| | 01 | | - | | | TxReady |
| | 10 | FifoEmpty | FifoEmpty | | FifoEmpty | |
| | 11 | FifoEmpty | FifoEmpty | | FifoEmpty | |
| DIO4 | 00 | - | | TempChange / LowBat | TempChange / LowBat | |
| | 01 | | - | | PllLock | |
| | 10 | | | | TimeOut | - |
| | 11 | | | | Rssi / PreambleDetect | - |
| DIO5 | 00 | ClkOut if RC | ClkOut | | ClkOut | |
| | 01 | | - | | PllLock | |
| | 10 | | | | Data | |
| | 11 | - | | ModeReady | ModeReady | |

4.2.12. Continuous Mode

4.2.12.1. General Description

As illustrated in Figure 29, in Continuous mode the NRZ data to (from) the (de)modulator is directly accessed by the uC on the bidirectional DIO2/DATA pin. The FIFO and packet handler are thus inactive.

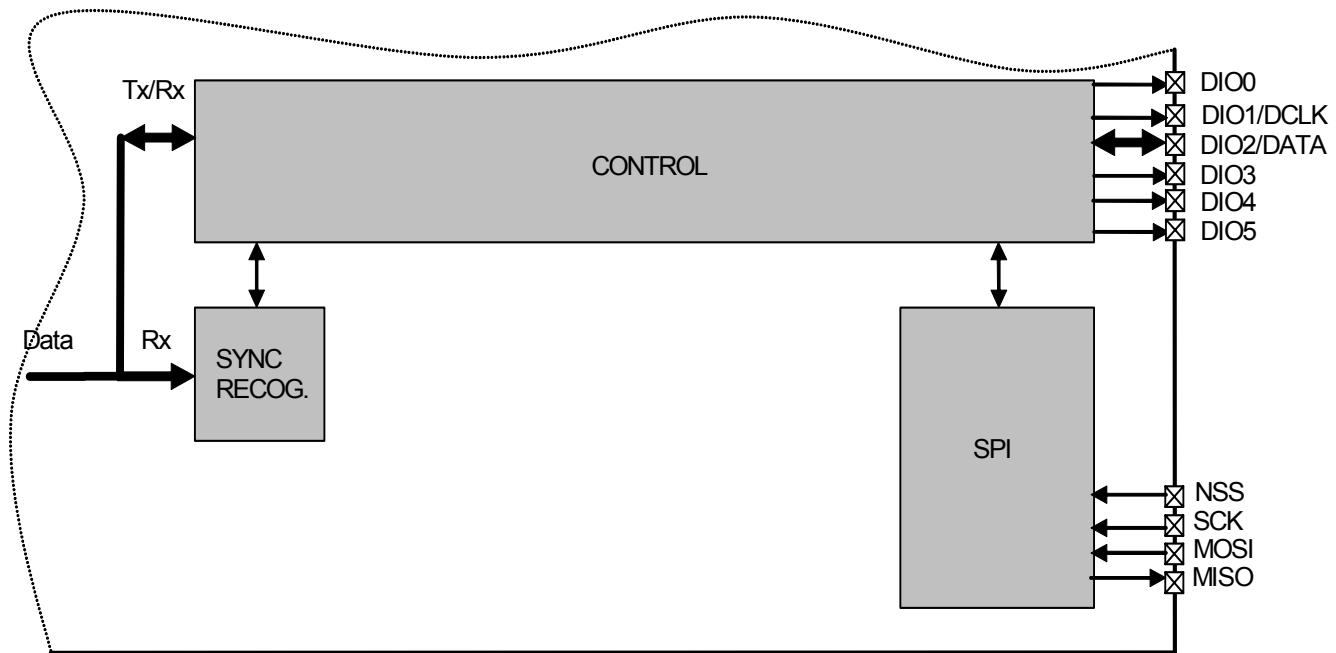


Figure 29. Continuous Mode Conceptual View

4.2.12.2. Tx Processing

In Tx mode, a synchronous data clock for an external uC is provided on DIO1/DCLK pin. Clock timing with respect to the data is illustrated in Figure 30. DATA is internally sampled on the rising edge of DCLK so the uC can change logic state anytime outside the grayed out setup/hold zone.

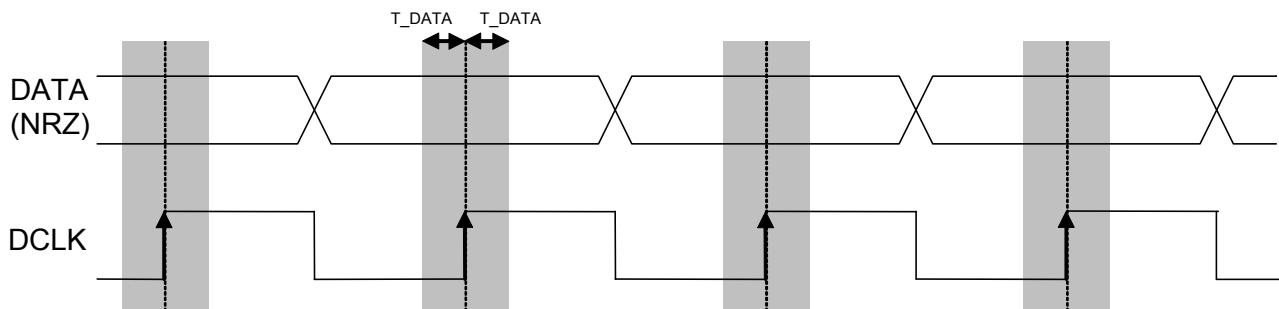


Figure 30. Tx Processing in Continuous Mode

Note the use of DCLK is required when the modulation shaping is enabled.

4.2.12.3. Rx Processing

If the bit synchronizer is disabled, the raw demodulator output is made directly available on DATA pin and no DCLK signal is provided.

Conversely, if the bit synchronizer is enabled, synchronous cleaned data and clock are made available respectively on DIO2/DATA and DIO1/DCLK pins. DATA is sampled on the rising edge of DCLK and updated on the falling edge as illustrated below.

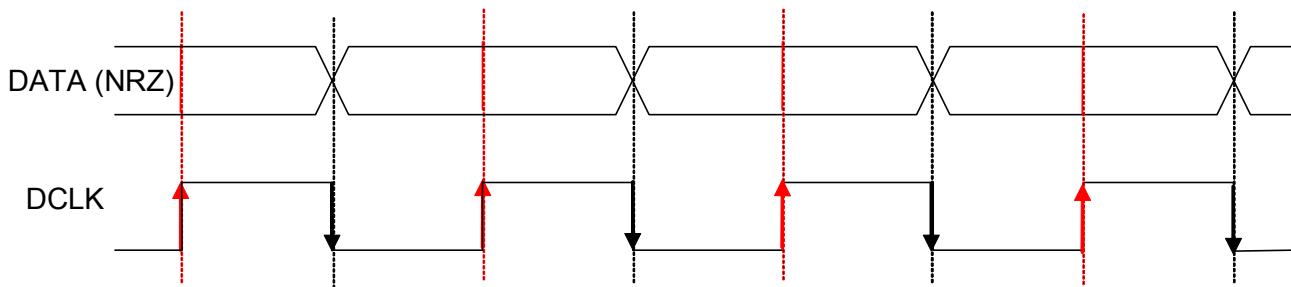


Figure 31. Rx Processing in Continuous Mode

Note In Continuous mode it is always recommended to enable the bit synchronizer to clean the DATA signal even if the DCLK signal is not used by the uC (bit synchronizer is automatically enabled in Packet mode).

4.2.13. Packet Mode

4.2.13.1. General Description

In Packet mode the NRZ data to (from) the (de)modulator is not directly accessed by the uC but stored in the FIFO and accessed via the SPI interface.

In addition, the SX1276/77/78/79 packet handler performs several packet oriented tasks such as Preamble and Sync word generation, CRC calculation/check, whitening/dewhitening of data, Manchester encoding/decoding, address filtering, etc. This simplifies software and reduces uC overhead by performing these repetitive tasks within the RF chip itself.

Another important feature is ability to fill and empty the FIFO in Sleep/Stdby mode, ensuring optimum power consumption and adding more flexibility for the software.

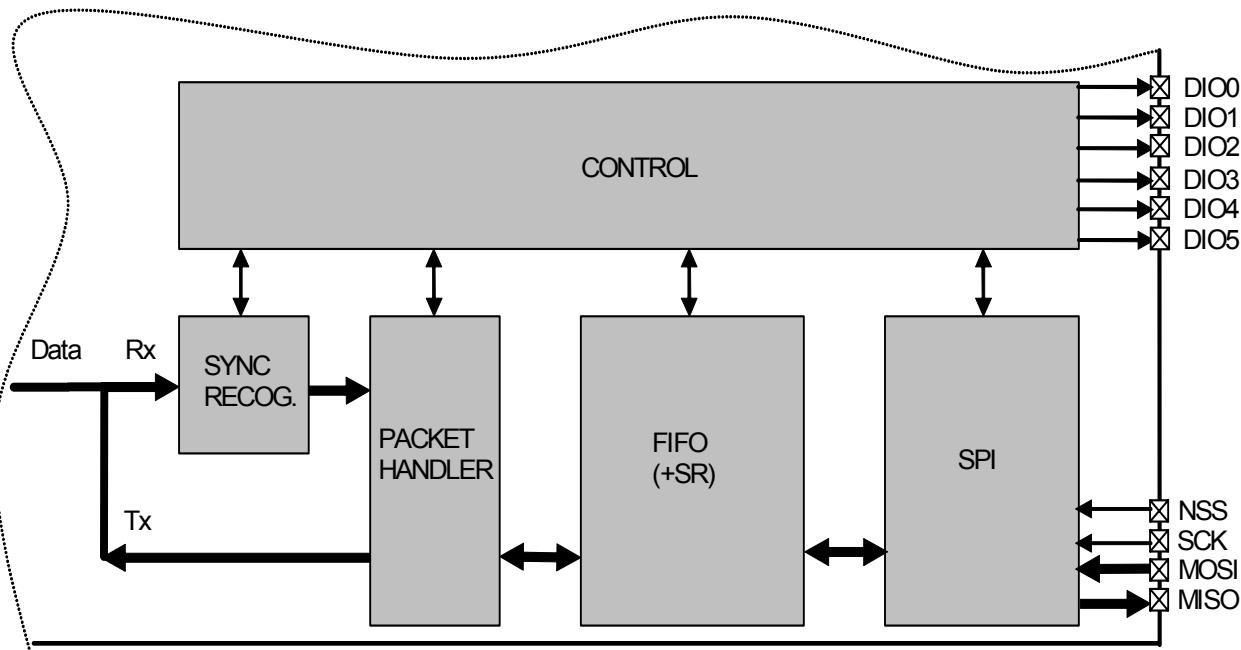


Figure 32. Packet Mode Conceptual View

Note The Bit Synchronizer is automatically enabled in Packet mode.

4.2.13.2. Packet Format

Fixed Length Packet Format

Fixed length packet format is selected when bit *PacketFormat* is set to 0 and *PayloadLength* is set to any value greater than 0.

In applications where the packet length is fixed in advance, this mode of operation may be of interest to minimize RF overhead (no length byte field is required). All nodes, whether Tx only, Rx only, or Tx/Rx should be programmed with the same packet length value.

The length of the payload is limited to 2047 bytes.

The length programmed in *PayloadLength* relates only to the payload which includes the message and the optional address byte. In this mode, the payload must contain at least one byte, i.e. address or message byte.

An illustration of a fixed length packet is shown below. It contains the following fields:

- ◆ Preamble (1010...)
- ◆ Sync word (Network ID)
- ◆ Optional Address byte (Node ID)
- ◆ Message data
- ◆ Optional 2-bytes CRC checksum

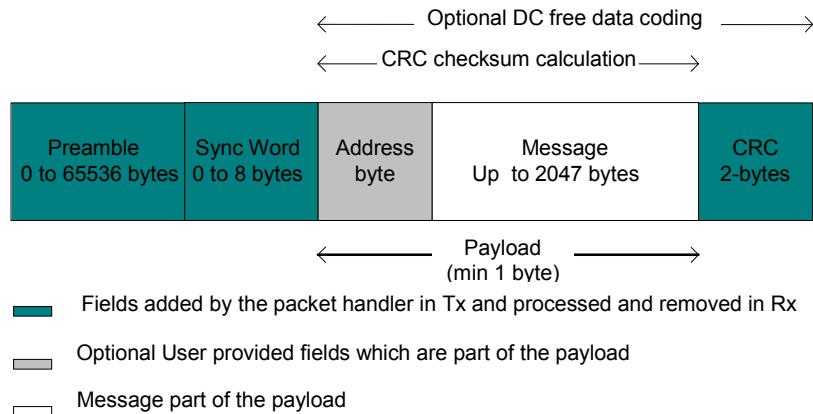


Figure 33. Fixed Length Packet Format

Variable Length Packet Format

Variable length packet format is selected when bit *PacketFormat* is set to 1.

This mode is useful in applications where the length of the packet is not known in advance and can vary over time. It is then necessary for the transmitter to send the length information together with each packet in order for the receiver to operate properly.

In this mode the length of the payload, indicated by the length byte, is given by the first byte of the FIFO and is limited to 255 bytes. Note that the length byte itself is not included in its calculation. In this mode, the payload must contain at least 2 bytes, i.e. length + address or message byte.

An illustration of a variable length packet is shown below. It contains the following fields:

- ◆ Preamble (1010...)
- ◆ Sync word (Network ID)
- ◆ Length byte
- ◆ Optional Address byte (Node ID)
- ◆ Message data

- ◆ Optional 2-bytes CRC checksum

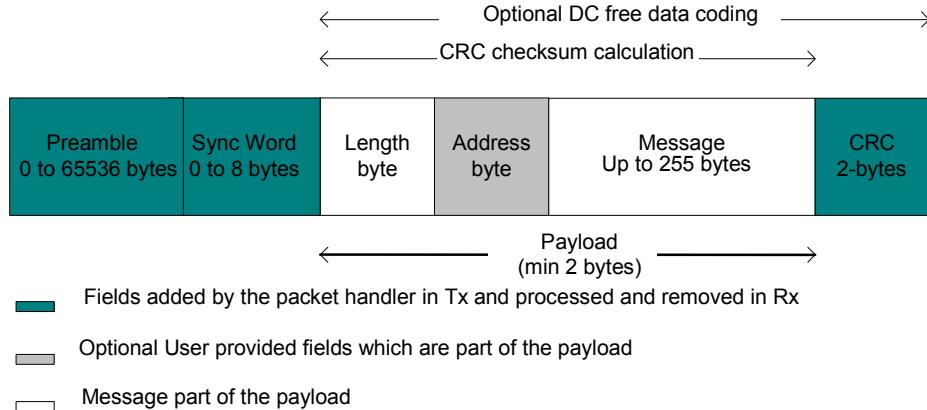


Figure 34. Variable Length Packet Format

Unlimited Length Packet Format

Unlimited length packet format is selected when bit *PacketFormat* is set to 0 and *PayloadLength* is set to 0. The user can then transmit and receive packet of arbitrary length and *PayloadLength* register is not used in Tx/Rx modes for counting the length of the bytes transmitted/received.

In Tx the data is transmitted depending on the *TxStartCondition* bit. On the Rx side the data processing features like Address filtering, Manchester encoding and data whitening are not available if the sync pattern length is set to zero (*SyncOn* = 0). The CRC detection in Rx is also not supported in this mode of the packet handler, however CRC generation in Tx is operational. The interrupts like *CrcOk* & *PayloadReady* are not available either.

An unlimited length packet shown below is made up of the following fields:

- ◆ Preamble (1010...).
- ◆ Sync word (Network ID).
- ◆ Optional Address byte (Node ID).
- ◆ Message data
- ◆ Optional 2-bytes CRC checksum (Tx only)

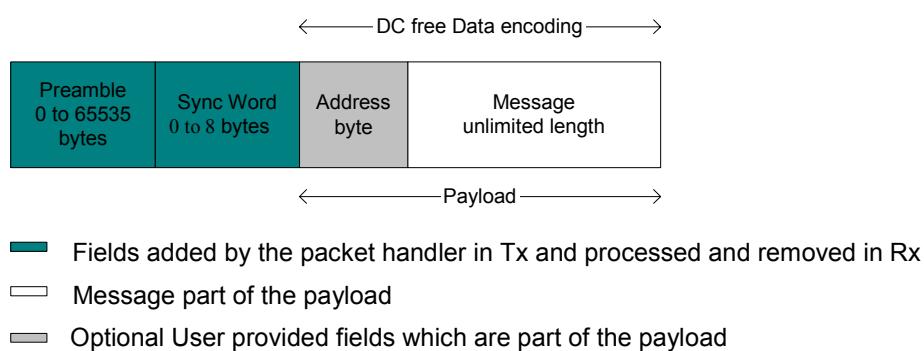


Figure 35. Unlimited Length Packet Format

4.2.13.3. Tx Processing

In Tx mode the packet handler dynamically builds the packet by performing the following operations on the payload available in the FIFO:

- ◆ Add a programmable number of preamble bytes
- ◆ Add a programmable Sync word
- ◆ Optionally calculating CRC over complete payload field (optional length byte + optional address byte + message) and appending the 2 bytes checksum.
- ◆ Optional DC-free encoding of the data (Manchester or whitening)

Only the payload (including optional address and length fields) is required to be provided by the user in the FIFO.

The transmission of packet data is initiated by the Packet Handler only if the chip is in Tx mode and the transmission condition defined by *TxStartCondition* is fulfilled. If transmission condition is not fulfilled then the packet handler transmits a preamble sequence until the condition is met. This happens only if the preamble length /= 0, otherwise it transmits a zero or one until the condition is met to transmit the packet data.

The transmission condition itself is defined as:

- ◆ if *TxStartCondition* = 1, the packet handler waits until the first byte is written into the FIFO, then it starts sending the preamble followed by the sync word and user payload
- ◆ If *TxStartCondition* = 0, the packet handler waits until the number of bytes written in the FIFO is equal to the number defined in *RegFifoThresh* + 1
- ◆ If the condition for transmission was already fulfilled i.e. the FIFO was filled in Sleep/Stdby then the transmission of packet starts immediately on enabling Tx

4.2.13.4. Rx Processing

In Rx mode the packet handler extracts the user payload to the FIFO by performing the following operations:

- ◆ Receiving the preamble and stripping it off
- ◆ Detecting the Sync word and stripping it off
- ◆ Optional DC-free decoding of data
- ◆ Optionally checking the address byte
- ◆ Optionally checking CRC and reflecting the result on *CrcOk*.

Only the payload (including optional address and length fields) is made available in the FIFO.

When the Rx mode is enabled the demodulator receives the preamble followed by the detection of sync word. If fixed length packet format is enabled then the number of bytes received as the payload is given by the *PayloadLength* parameter.

In variable length mode the first byte received after the sync word is interpreted as the length of the received packet. The internal length counter is initialized to this received length. The *PayloadLength* register is set to a value which is greater than the maximum expected length of the received packet. If the received length is greater than the maximum length stored in *PayloadLength* register the packet is discarded otherwise the complete packet is received.

If the address check is enabled then the second byte received in case of variable length and first byte in case of fixed length is the address byte. If the address matches to the one in the *NodeAddress* field, reception of the data continues otherwise it's stopped. The CRC check is performed if *CrcOn* = 1 and the result is available in *CrcOk* indicating that the

CRC was successful. An interrupt (*PayloadReady*) is also generated on DIO0 as soon as the payload is available in the FIFO. The payload available in the FIFO can also be read in Sleep/Standby mode.

If the CRC fails the *PayloadReady* interrupt is not generated and the FIFO is cleared. This function can be overridden by setting *CrcAutoClearOff* = 1, forcing the availability of *PayloadReady* interrupt and the payload in the FIFO even if the CRC fails.

4.2.13.5. Handling Large Packets

When *PayloadLength* exceeds FIFO size (64 bytes) whether in fixed, variable or unlimited length packet format, in addition to *PacketSent* in Tx and *PayloadReady* or *CrcOk* in Rx, the FIFO interrupts/flags can be used as described below:

- ◆ For Tx:

FIFO can be prefilled in Sleep/Standby but must be refilled “on-the-fly” during Tx with the rest of the payload.

- 1) Pre-fill FIFO (in Sleep/Standby first or directly in Tx mode) until *FifoThreshold* or *FifoFull* is set
- 2) In Tx, wait for *FifoThreshold* or *FifoEmpty* to be set (i.e. FIFO is nearly empty)
- 3) Write bytes into the FIFO until *FifoThreshold* or *FifoFull* is set.
- 4) Continue to step 2 until the entire message has been written to the FIFO (*PacketSent* will fire when the last bit of the packet has been sent).

- ◆ For Rx:

FIFO must be unfilled “on-the-fly” during Rx to prevent FIFO overrun.

- 1) Start reading bytes from the FIFO when *FifoEmpty* is cleared or *FifoThreshold* becomes set.
- 2) Suspend reading from the FIFO if *FifoEmpty* fires before all bytes of the message have been read
- 3) Continue to step 1 until *PayloadReady* or *CrcOk* fires
- 4) Read all remaining bytes from the FIFO either in Rx or Sleep/Standby mode

4.2.13.6. Packet Filtering

The SX1276/77/78/79 packet handler offers several mechanisms for packet filtering, ensuring that only useful packets are made available to the uC, reducing significantly system power consumption and software complexity.

Sync Word Based

Sync word filtering/recognition is used for identifying the start of the payload and also for network identification. As previously described, the Sync word recognition block is configured (size, value) in *RegSyncConfig* and *RegSyncValue(i)* registers. This information is used, both for appending Sync word in Tx, and filtering packets in Rx.

Every received packet which does not start with this locally configured Sync word is automatically discarded and no interrupt is generated.

When the Sync word is detected, payload reception automatically starts and *SyncAddressMatch* is asserted.

Note Sync Word values containing 0x00 byte(s) are forbidden

Address Based

Address filtering can be enabled via the *AddressFiltering* bits. It adds another level of filtering, above Sync word (i.e. Sync must match first), typically useful in a multi-node networks where a network ID is shared between all nodes (Sync word) and each node has its own ID (address).

Two address based filtering options are available:

- ◆ *AddressFiltering* = 01: Received address field is compared with internal register *NodeAddress*. If they match then the packet is accepted and processed, otherwise it is discarded.
- ◆ *AddressFiltering* = 10: Received address field is compared with internal registers *NodeAddress* and *BroadcastAddress*. If either is a match, the received packet is accepted and processed, otherwise it is discarded. This additional check with a constant is useful for implementing broadcast in a multi-node networks

Please note that the received address byte, as part of the payload, is not stripped off the packet and is made available in the FIFO. In addition, *NodeAddress* and *AddressFiltering* only apply to Rx. On Tx side, if address filtering is expected, the address byte should simply be put into the FIFO like any other byte of the payload.

As address filtering requires a Sync word match, both features share the same interrupt flag *SyncAddressMatch*.

Length Based

In variable length Packet mode, *PayloadLength* must be programmed with the maximum payload length permitted. If received length byte is smaller than this maximum then the packet is accepted and processed, otherwise it is discarded.

Please note that the received length byte, as part of the payload, is not stripped off the packet and is made available in the FIFO.

To disable this function the user should set the value of the *PayloadLength* to 2047.

CRC Based

The CRC check is enabled by setting bit *CrcOn* in *RegPacketConfig1*. It is used for checking the integrity of the message.

- ◆ On Tx side a two byte CRC checksum is calculated on the payload part of the packet and appended to the end of the message
- ◆ On Rx side the checksum is calculated on the received payload and compared with the two checksum bytes received. The result of the comparison is stored in bit *CrcOk*.

By default, if the CRC check fails then the FIFO is automatically cleared and no interrupt is generated. This filtering function can be disabled via *CrcAutoClearOff* bit and in this case, even if CRC fails, the FIFO is not cleared and only *PayloadReady* interrupt goes high. Please note that in both cases, the two CRC checksum bytes are stripped off by the packet handler and only the payload is made available in the FIFO. Two CRC implementations are selected with bit *CrcWhiteningType*.

Table 31 CRC Description

| Crc Type | <i>CrcWhiteningType</i> | Polynomial | Seed Value | Complemented |
|----------|-------------------------|-----------------------------|------------|--------------|
| CCITT | 0 (default) | $X^{16} + X^{12} + X^5 + 1$ | 0x1D0F | Yes |
| IBM | 1 | $X^{16} + X^{15} + X^2 + 1$ | 0xFFFF | No |

A C code implementation of each CRC type is proposed in Application Section 7.

4.2.13.7. DC-Free Data Mechanisms

The payload to be transmitted may contain long sequences of 1's and 0's, which introduces a DC bias in the transmitted signal. The radio signal thus produced has a non uniform power distribution over the occupied channel bandwidth. It also introduces data dependencies in the normal operation of the demodulator. Thus it is useful if the transmitted data is random and DC free.

For such purposes, two techniques are made available in the packet handler: Manchester encoding and data whitening.

Note *Only one of the two methods can be enabled at a time.*

Manchester Encoding

Manchester encoding/decoding is enabled if *DcFree* = 01 and can only be used in Packet mode.

The NRZ data is converted to Manchester code by coding '1' as "10" and '0' as "01".

In this case, the maximum chip rate is the maximum bit rate given in the specifications and the actual bit rate is half the chip rate.

Manchester encoding and decoding is only applied to the payload and CRC checksum while preamble and Sync word are kept NRZ. However, the chip rate from preamble to CRC is the same and defined by *BitRate* in *RegBitRate* (Chip Rate = Bit Rate NRZ = 2 x Bit Rate Manchester).

Manchester encoding/decoding is thus made transparent for the user, who still provides/retrieves NRZ data to/from the FIFO.

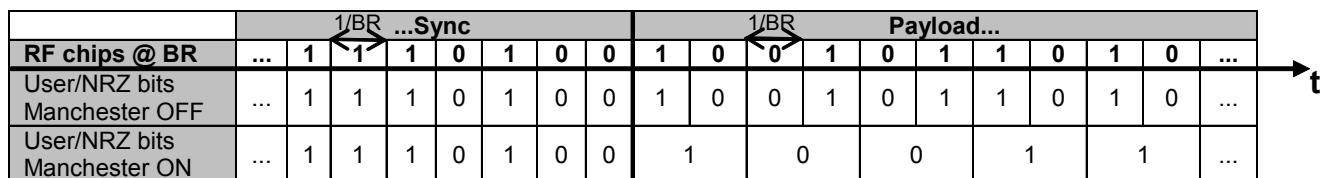


Figure 36. Manchester Encoding/Decoding

Data Whitening

Another technique called whitening or scrambling is widely used for randomizing the user data before radio transmission. The data is whitened using a random sequence on the Tx side and de-whitened on the Rx side using the same sequence. Comparing to Manchester technique it has the advantage of keeping NRZ data rate i.e. actual bit rate is not halved.

The whitening/de-whitening process is enabled if *DcFree* = 10. A 9-bit LFSR is used to generate a random sequence. The payload and 2-byte CRC checksum is then XORed with this random sequence as shown below. The data is de-whitened on the receiver side by XORing with the same random sequence.

Payload whitening/de-whitening is thus made transparent for the user, who still provides/retrieves NRZ data to/from the FIFO.

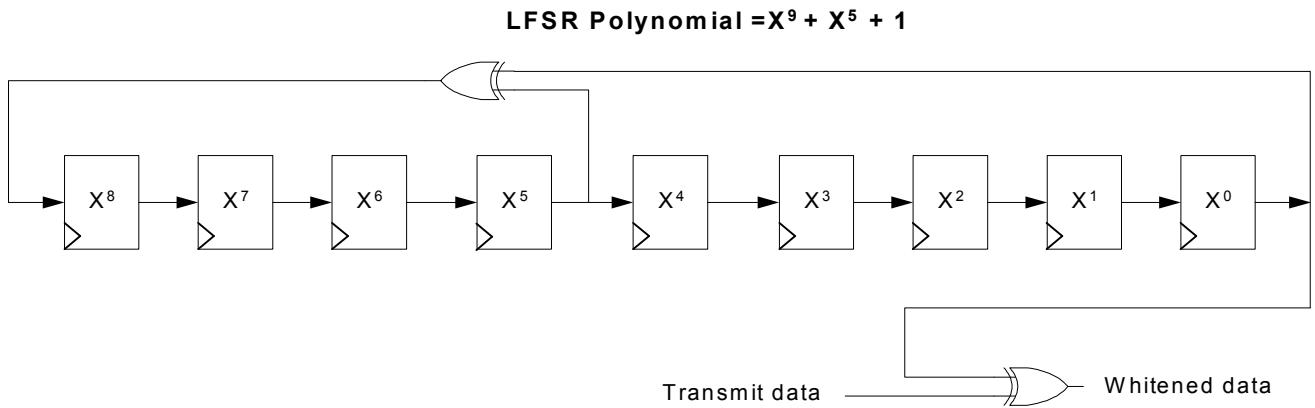


Figure 37. Data Whitening Polynomial

4.2.13.8. Beacon Tx Mode

In some short range wireless network topologies a repetitive message, also known as beacon, is transmitted periodically by a transmitter. The Beacon Tx mode allows for the re-transmission of the same packet without having to fill the FIFO multiple times with the same data.

When *BeaconOn* in *RegPacketConfig2* is set to 1, the FIFO can be filled only once in Sleep or Stdby mode with the required payload. After a first transmission, *FifoEmpty* will go high as usual, but the FIFO content will be restored when the chip exits Transmit mode. *FifoEmpty*, *FifoFull* and *FifoLevel* flags are also restored.

This feature is only available in Fixed packet format, with the Payload Length smaller than the FIFO size. The control of the chip modes (Tx-Sleep-Tx....) can either be undertaken by the microcontroller, or be automated in the Top Sequencer. See example in Section 4.2.13.8.

The Beacon Tx mode is exited by setting *BeaconOn* to 0, and clearing the FIFO by setting *FifoOverrun* to 1.

4.2.14. io-homecontrol® Compatibility Mode

The SX1276/77/78/79 features a io-homecontrol® compatibility mode. Please contact your local Semtech representative for details on its implementation.

4.3. SPI Interface

The SPI interface gives access to the configuration register via a synchronous full-duplex protocol corresponding to CPOL = 0 and CPHA = 0 in Motorola/Freescale nomenclature. Only the slave side is implemented.

Three access modes to the registers are provided:

- ◆ SINGLE access: an address byte followed by a data byte is sent for a write access whereas an address byte is sent and a read byte is received for the read access. The NSS pin goes low at the beginning of the frame and goes high after the data byte.
- ◆ BURST access: the address byte is followed by several data bytes. The address is automatically incremented internally between each data byte. This mode is available for both read and write accesses. The NSS pin goes low at the beginning of the frame and stay low between each byte. It goes high only after the last byte transfer.
- ◆ FIFO access: if the address byte corresponds to the address of the FIFO, then succeeding data byte will address the FIFO. The address is not automatically incremented but is memorized and does not need to be sent between each data byte. The NSS pin goes low at the beginning of the frame and stay low between each byte. It goes high only after the last byte transfer.

The figure below shows a typical SPI single access to a register.

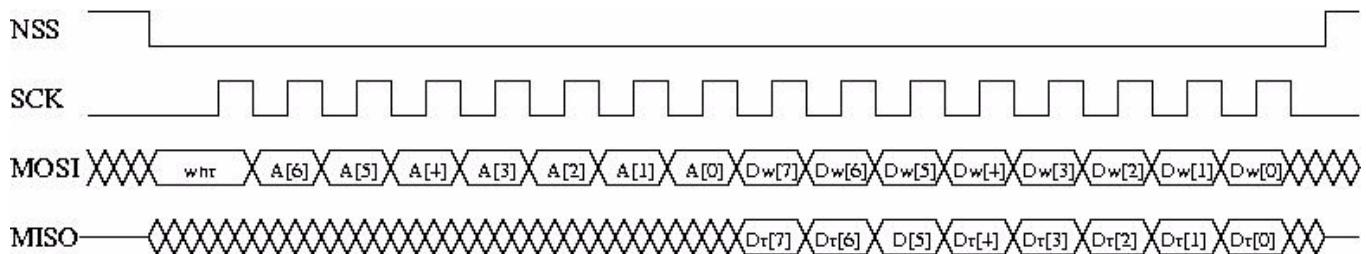


Figure 38. SPI Timing Diagram (single access)

MOSI is generated by the master on the falling edge of SCK and is sampled by the slave (i.e. this SPI interface) on the rising edge of SCK. MISO is generated by the slave on the falling edge of SCK.

A transfer is always started by the NSS pin going low. MISO is high impedance when NSS is high.

The first byte is the address byte. It is comprises:

- ◆ A wnr bit, which is 1 for write access and 0 for read access.
- ◆ Then 7 bits of address, MSB first.

The second byte is a data byte, either sent on MOSI by the master in case of a write access or received by the master on MISO in case of read access. The data byte is transmitted MSB first.

Proceeding bytes may be sent on MOSI (for write access) or received on MISO (for read access) without a rising NSS edge and re-sending the address. In FIFO mode, if the address was the FIFO address then the bytes will be written / read at the FIFO address. In Burst mode, if the address was not the FIFO address, then it is automatically incremented for each new byte received.

The frame ends when NSS goes high. The next frame must start with an address byte. The SINGLE access mode is therefore a special case of FIFO / BURST mode with only 1 data byte transferred.

During the write access, the byte transferred from the slave to the master on the MISO line is the value of the written register before the write operation.

5. SX1276/77/78/79 Analog & RF Frontend Electronics

5.1. Power Supply Strategy

The SX1276/77/78/79 employs an internal voltage regulation scheme which provides stable operating voltage, and hence device characteristics, over the full industrial temperature and operating voltage range of operation. This includes up to +17 dBm of RF output power which is maintained from 1.8 V to 3.7 V and +20 dBm from 2.4 V to 3.7 V.

The SX1276/77/78/79 can be powered from any low-noise voltage source via pins VBAT_ANA, VBAT_RF and VBAT_DIG. Decoupling capacitors should be connected, as suggested in the reference design of the applications section of this document, on VR_PA, VR_DIG and VR_ANA pins to ensure correct operation of the built-in voltage regulators.

5.2. Low Battery Detector

A low battery detector is also included allowing the generation of an interrupt signal in response to the supply voltage dropping below a programmable threshold that is adjustable through the register *RegLowBat*. The interrupt signal can be mapped to any of the DIO pins by programming *RegDioMapping*.

5.3. Frequency Synthesis

5.3.1. Crystal Oscillator

The crystal oscillator is the main timing reference of the SX1276/77/78/79. It is used as the reference for the PLL's frequency synthesis and as the clock signal for all digital processing.

The crystal oscillator startup time, TS_OSC, depends on the electrical characteristics of the crystal reference used, for more information on the electrical specification of the crystal see section 7.1. The crystal connects to the Pierce oscillator on pins XTA and XTB. The SX1276/77/78/79 optimizes the startup time and automatically triggers the PLL when the oscillator signal is stable.

Optionally, an external clock can be used to replace the crystal oscillator. This typically takes the form of a tight tolerance temperature compensated crystal oscillator (TCXO). When using an external clock source the bit *TcxoInputOn* of register *RegTcxo* should be set to 1 and the external clock has to be provided on XTA (pin 5). XTB (pin 6) should be left open.

The peak-peak amplitude of the input signal must never exceed 1.8 V. Please consult your TCXO supplier for an appropriate value of decoupling capacitor, C_D.

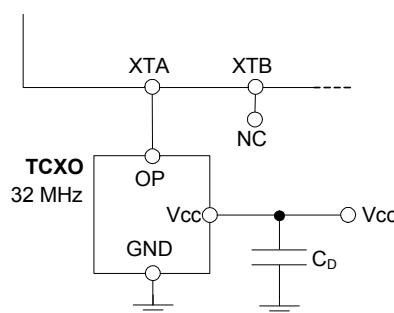


Figure 39. TCXO Connection

5.3.2. CLKOUT Output

The reference frequency, or a fraction of it, can be provided on DIO5 (pin 13) by modifying bits *ClkOut* in *RegDioMapping2*. Two typical applications of the CLKOUT output include:

- ◆ To provide a clock output for a companion processor, thus saving the cost of an additional oscillator. CLKOUT can be made available in any operation mode except Sleep mode and is automatically enabled at power on reset.
- ◆ To provide an oscillator reference output. Measurement of the CLKOUT signal enables simple software trimming of the initial crystal tolerance.

Note *To minimize the current consumption of the SX1276/77/78/79, please ensure that the CLKOUT signal is disabled when not required.*

5.3.3. PLL

The local oscillator of the SX1276/77/78/79 is derived from two almost identical fractional-N PLLs that are referenced to the crystal oscillator circuit. Both PLLs feature a programmable bandwidth setting where one of four discrete preset bandwidths may be accessed.

The SX1276/77/78/79 PLL uses a 19-bit sigma-delta modulator whose frequency resolution, constant over the whole frequency range, is given by:

$$F_{STEP} = \frac{F_{XOSC}}{2^{19}}$$

The carrier frequency is programmed through *RegFrF*, split across addresses 0x06 to 0x08:

$$F_{RF} = F_{STEP} \times Frf(23,0)$$

Note *The Frf setting is split across 3 bytes. A change in the center frequency will only be taken into account when the least significant byte FrfLsb in RegFrfLsb is written. This allows the potential for user generation of m-ary FSK at very low bit rates. This is possible where frequency modulation is achieved by direct programming of the programmed RF centre frequency. To enable this functionality set the FastHopOn bit of register RegPllHop.*

Three frequency bands are supported, defined as follows:

Table 32 Frequency Bands

| Name | Frequency Limits | Products |
|-------------|----------------------------|-----------------|
| Band 1 (HF) | 862 (*779)-1020 (*960) MHz | SX1276/77/79 |
| Band 2 (LF) | 410-525 (*480) MHz | SX1276/77/78/79 |
| Band 3 (LF) | 137-175 (*160)MHz | SX1276/77/78/79 |

* For SX1279

5.3.4. RC Oscillator

All timing operations in the low-power Sleep state of the Top Level Sequencer rely on the accuracy of the internal low-power RC oscillator. This oscillator is automatically calibrated at the device power-up not requiring any user input.

5.4. Transmitter Description

The transmitter of SX1276/77/78/79 comprises the frequency synthesizer, modulator (both LoRa™ and FSK/OOK) and power amplifier blocks, together with the DC biasing and ramping functionality that is provided through the VR_PA block.

5.4.1. Architecture Description

The architecture of the RF front end is shown in the following diagram:

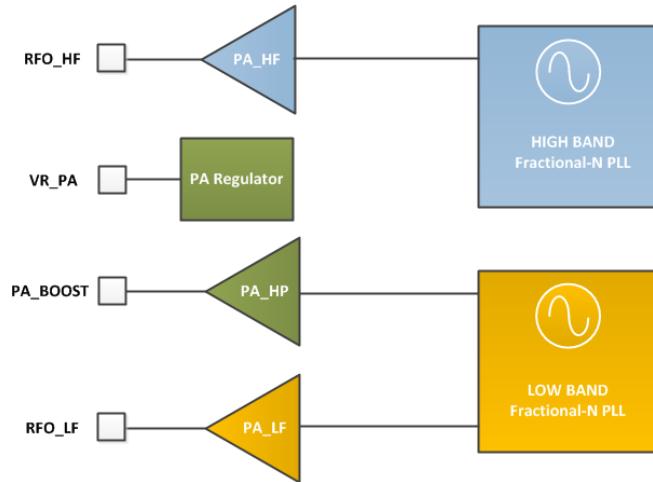


Figure 40. RF Front-end Architecture Shows the Internal PA Configuration.

5.4.2. RF Power Amplifiers

PA_HF and PA_LF are high efficiency amplifiers capable of yielding RF power programmable in 1 dB steps from -4 to +14dBm directly into a 50 ohm load with low current consumption. PA_LF covers the lower bands (up to 525 MHz), whilst PA_HF will cover the upper bands (from 779 MHz). The output power is sensitive to the power supply voltage, and typically their performance is expressed at 3.3V.

PA_HP (High Power), connected to the PA_BOOST pin, covers all frequency bands that the chip addresses. It permits continuous operation at up to +17 dBm and duty cycled operation at up to +20dBm. For full details of operation at +20dBm please consult section 5.4.3

Table 33 Power Amplifier Mode Selection Truth Table

| PaSelect | Mode | Power Range | Pout Formula |
|----------|------------------------------------|--------------|---|
| 0 | PA_HF or PA_LF on RFO_HF or RFO_LF | -4 to +15dBm | $P_{out} = P_{max} - (15 - \text{OutputPower})$ $P_{max} = 10.8 + 0.6 * \text{MaxPower}$ [dBm] |
| 1 | PA_HP on PA_BOOST, any frequency | +2 to +17dBm | $P_{out} = 17 - (15 - \text{OutputPower})$ [dBm] |

Notes - For +20 dBm restrictions on operation please consult the following .

- To ensure correct operation at the highest power levels ensure that the current limiter OcpTrim is adjusted to permit delivery of the requisite supply current.
- If the PA_BOOST pin is not used it may be left floating.

5.4.3. High Power +20 dBm Operation

The SX1276/77/78/79 have a high power +20 dBm capability on PA_BOOST pin, with the following settings:

Table 34 High Power Settings

| Register | Address | Value for High Power | Default value PA_HF/LF or +17dBm | Description |
|----------|---------|----------------------|----------------------------------|------------------------------|
| RegPaDac | 0x4d | 0x87 | 0x84 | Set Pmax to +20dBm for PA_HP |

Notes - High Power settings must be turned off when using PA_LF or PA_HF

- The Over Current Protection limit should be adapted to the actual power level, in RegOcp

Specific Absolute Maximum Ratings and Operating Range restrictions apply to the +20 dBm operation. They are listed in Table 35 and Table 36.

Table 35 Operating Range, +20dBm Operation

| Symbol | Description | Min | Max | Unit |
|------------|--|-----|-----|------|
| DC_20dBm | Duty Cycle of transmission at +20 dBm output | - | 1 | % |
| VSWR_20dBm | Maximum VSWR at antenna port, +20 dBm output | - | 3:1 | - |

Table 36 Operating Range, +20dBm Operation

| Symbol | Description | Min | Max | Unit |
|-------------|--------------------------------|-----|-----|------|
| VDDop_20dBm | Supply voltage, +20 dBm output | 2.4 | 3.7 | V |

The duty cycle of transmission at +20 dBm is limited to 1%, with a maximum VSWR of 3:1 at antenna port, over the standard operating range [-40;+85°C]. For any other operating condition, contact your Semtech representative.

5.4.4. Over Current Protection

The power amplifiers of SX1276/77/78/79 are protected against current over supply in adverse RF load conditions by the over current protection block. This has the added benefit of protecting battery chemistries with limited peak current capability and minimising worst case PA consumption in battery life calculation. The current limiter value is controlled by the *OcpTrim* bits in *RegOcp*, and is calculated according to the following formulae:

Table 37 Trimming of the OCP Current

| <i>OcpTrim</i> | I_{MAX} | I_{max} Formula |
|----------------|---------------|--------------------------------|
| 0 to 15 | 45 to 120 mA | $45 + 5 * OcpTrim$ [mA] |
| 16 to 27 | 130 to 240 mA | $-30 + 10 * OcpTrim$ [mA] |
| 27+ | 240 mA | 240 mA |

Note I_{max} sets a limit on the current drain of the Power Amplifier only, hence the maximum current drain of the SX1276/77/78/79 is equal to I_{max} + IDDFS.

5.5. Receiver Description

5.5.1. Overview

The SX1276/77/78/79 features a digital receiver with the analog to digital conversion process being performed directly following the LNA-Mixers block. In addition to the LoRa™ modulation scheme the low-IF receiver is able to demodulate ASK, OOK, (G)FSK and (G)MSK modulation. All filtering, demodulation, gain control, synchronization and packet handling is performed digitally allowing a high degree of programmable flexibility. The receiver also has automatic gain calibration, this improves the precision of RSSI measurement and enhances image rejection.

5.5.2. Receiver Enabled and Receiver Active States

In the receiver operating mode two states of functionality are defined. Upon initial transition to receiver operating mode the receiver is in the ‘receiver-enabled’ state. In this state the receiver awaits for either the user defined valid preamble or RSSI detection criterion to be fulfilled. Once met the receiver enters ‘receiver-active’ state. In this second state the received signal is processed by the packet engine and top level sequencer. For a complete description of the digital functions of the SX1276/77/78/79 receiver please see section 4 of the datasheet.

5.5.3. Automatic Gain Control In FSK/OOK Mode

The AGC feature allows receiver to handle a wide Rx input dynamic range from the sensitivity level up to maximum input level of 0dBm or more, whilst optimizing the system linearity.

The following table shows typical NF and IIP3 performances for the SX1276/77/78/79 LNA gains available.

Table 38 LNA Gain Control and Performances

| <i>RX input level (Pin)</i> | <i>Gain Setting</i> | <i>LnaGain</i> | <i>Relative LNA Gain [dB]</i> | <i>NF Band 3/2/1 [dB]</i> | <i>IIP3 Band 3/2/1 [dBm]</i> |
|---|---------------------|----------------|-------------------------------|---------------------------|------------------------------|
| Pin <= AgcThresh1 | G1 | '001' | 0 dB | 4/5.5/7 | -15/-22/-11 |
| AgcThresh1 < Pin <= AgcThresh2 | G2 | '010' | -6 dB | 6.5/8/12 | -11/-15/-6 |
| AgcThresh2 < Pin <= AgcThresh3 | G3 | '011' | -12 dB | 11/12/17 | -11/-12/0 |
| AgcThresh3 < Pin <= AgcThresh4 | G4 | '100' | -24 dB | 20/21/27 | 2/3/9 |
| AgcThresh4 < Pin <= AgcThresh5 | G5 | '110' | -26 dB | 32/33/35 | 10/10/14 |
| AgcThresh5 < Pin | G6 | '111' | -48 dB | 44/45/43 | 11/12/14 |

5.5.4. RSSI in FSK/OOK Mode

The RSSI provides a measure of the incoming signal power at RF input port, measured within the receiver bandwidth. The signal power is available in *RssiValue*. This value is absolute in units of dBm and with a resolution of 0.5 dB. The formula below relates the register value to the absolute input signal level at the RF input port:

$$RssiValue = -2 \cdot RF\ level\ [dBm] + RssiOffset\ [dB]$$

The RSSI value can be compensated to take into account the loss in the matching network or even the gain of an additional LNA by using *RssiOffset*. The offset can be chosen in 1 dB steps from -16 to +15 dB. When compensation is applied, the effective signal strength is read as follows:

$$RSSI\ [dBm] = -\frac{RssiValue}{2}$$

The RSSI value is smoothed on a user defined number of measured RSSI samples. The precision of the RSSI value is related to the number of RSSI samples used. *RssiSmoothing* selects the number of RSSI samples from a minimum of 2 samples up to 256 samples in increments of power of 2. Table 39 gives the estimation of the RSSI accuracy for a 10 dB SNR and response time versus the number of RSSI samples programmed in *RssiSmoothing*.

Table 39 RssiSmoothing Options

| <i>RssiSmoothing</i> | <i>Number of Samples</i> | <i>Estimated Accuracy</i> | <i>Response Time</i> |
|----------------------|--------------------------|---------------------------|--|
| '000' | 2 | ± 6 dB | |
| '001' | 4 | ± 5 dB | |
| '010' | 8 | ± 4 dB | |
| '011' | 16 | ± 3 dB | |
| '100' | 32 | ± 2 dB | |
| '101' | 64 | ± 1.5 dB | |
| '110' | 128 | ± 1.2 dB | |
| '111' | 256 | ± 1.1 dB | $\frac{2^{(RssiSmoothing+1)}}{4 \cdot RxBw\ [kHz]} [ms]$ |

The RSSI is calibrated when the image and RSSI calibration process is launched.

5.5.5. RSSI and SNR in LoRaTM Mode

The RSSI values reported by the LoRaTM modem differ from those expressed by the FSK/OOK modem. The following formula shows the method used to interpret the LoRaTM RSSI values:

$$\text{RSSI (dBm)} = -157 + Rssi, \text{ (when using the High Frequency (HF) port)}$$

or

$$\text{RSSI (dBm)} = -164 + Rssi, \text{ (when using the Low Frequency (LF) port)}$$

The same formula can be re-used to evaluate the signal strength of the received packet:

$$\text{Packet Strength (dBm)} = -157 + Rssi, \text{ (when using the High Frequency (HF) port)}$$

or

$$\text{Packet Strength (dBm)} = -164 + Rssi, \text{ (when using the Low Frequency (LF) port)}$$

Due to the nature of the LoRa modulation, it is possible to receive packets below the noise floor. In this situation, the SNR is used in conjunction of the PacketRssi to compute the signal strength of the received packet:

$$\text{Packet Strength (dBm)} = -157 + \text{PacketRssi} + \text{PacketSnr} * 0.25 \text{ (when using the HF port and SNR} < 0)$$

or

$$\text{Packet Strength (dBm)} = -164 + \text{PacketRssi} + \text{PacketSnr} * 0.25 \text{ (when using the LF port and SNR} < 0)$$

Note:

1. *PacketRssi* (in RegPktRssiValue), is an averaged version of *Rssi* (in RegRssiValue). *Rssi* can be read at any time (during packet reception or not), and should be averaged to give more precise results.
2. The constants, -157 and -164, may vary with the front-end setup of the SX1276/77/78/79 (*LnaBoost* =1 or 0, presence of an external LNA, mismatch at the LNA input...). It is recommended to adjust these values with a single-point calibration procedure to increase RSSI accuracy.
3. As signal strength increases (RSSI>-100dBm), the linearity of *PacketRssi* is not guaranteed and results will diverge from the ideal 1dB/dB ideal curve. When very good RSSI precision is required over the whole dynamic range of the receiver, two options are proposed:
 - *Rssi* in RegRssiValue offers better linearity. *Rssi* can be sampled during the reception of the payload (between ValidHeader and RxDone IRQ), and used to extract a more high-signal RSSI measurement
 - When SNR>=0, the standard formula can be adjusted to correct the slope:

$$\text{RSSI} = -157 + 16/15 * \text{PacketRssi} \text{ (or RSSI} = -164 + 16/15 * \text{PacketRssi})$$

5.5.6. Channel Filter

The role of the channel filter is to reject noise and interference outside of the wanted channel. The SX1276/77/78/79 channel filtering is implemented with a 16-tap finite impulse response (FIR) filter. Rejection of the filter is high enough that the filter stop-band performance is not the dominant influence on adjacent channel rejection performance. This is instead limited by the SX1276/77/78/79 local oscillator phase noise.

Note *To respect sampling criterion in the decimation chain of the receiver, the communication bit rate cannot be set at a higher than twice the single side receiver bandwidth (BitRate < 2 x RxBw)*

The single-side channel filter bandwidth *RxBw* is controlled by the parameters *RxBwMant* and *RxBwExp* in *RegRxBw*:

$$RxBw = \frac{FXOSC}{RxBwMant \times 2^{RxBwExp + 2}}$$

The following channel filter bandwidths are hence accessible in the case of a 32 MHz reference oscillator:

Table 40 Available RxBw Settings

| RxBwMant (binary/value) | RxBwExp (decimal) | RxBw (kHz) |
|------------------------------------|------------------------------|-------------------|
| | | FSK/OOK |
| 10b / 24 | 7 | 2.6 |
| 01b / 20 | 7 | 3.1 |
| 00b / 16 | 7 | 3.9 |
| 10b / 24 | 6 | 5.2 |
| 01b / 20 | 6 | 6.3 |
| 00b / 16 | 6 | 7.8 |
| 10b / 24 | 5 | 10.4 |
| 01b / 20 | 5 | 12.5 |
| 00b / 16 | 5 | 15.6 |
| 10b / 24 | 4 | 20.8 |
| 01b / 20 | 4 | 25.0 |
| 00b / 16 | 4 | 31.3 |
| 10b / 24 | 3 | 41.7 |
| 01b / 20 | 3 | 50.0 |
| 00b / 16 | 3 | 62.5 |
| 10b / 24 | 2 | 83.3 |
| 01b / 20 | 2 | 100.0 |
| 00b / 16 | 2 | 125.0 |
| 10b / 24 | 1 | 166.7 |
| 01b / 20 | 1 | 200.0 |
| 00b / 16 | 1 | 250.0 |
| Other settings | | reserved |

5.5.7. Temperature Measurement

A stand alone temperature measurement block is used in order to measure the temperature in any mode except Sleep and Standby. It is enabled by default, and can be stopped by setting *TempMonitorOff* to 1. The result of the measurement is stored in *TempValue* in *RegTemp*.

Due to process variations, the absolute accuracy of the result is +/- 10 °C. Higher precision requires a calibration procedure at a known temperature. The figure below shows the influence of just such a calibration process. For more information, including source code, please consult the applications section of this document.

Example temperature curve, typical device

| Actual Temp [Celsius] | RegTemp [Dec] | Temp before calibration [°C] | Temp after calibration [°C] |
|-----------------------|---------------|------------------------------|-----------------------------|
| 85 | 181 | 74 | 89 |
| 75 | 190 | 65 | 80 |
| 65 | 201 | 54 | 69 |
| 55 | 211 | 44 | 59 |
| 45 | 222 | 33 | 48 |
| 35 | 232 | 23 | 38 |
| 25 | 245 | 10 | 25 |
| 15 | 0 | 0 | 15 |
| 5 | 10 | -10 | 5 |
| -5 | 21 | -21 | -6 |
| -15 | 33 | -33 | -18 |
| -25 | 44 | -44 | -29 |
| -35 | 56 | -56 | -41 |
| -40 | 63 | -63 | -48 |

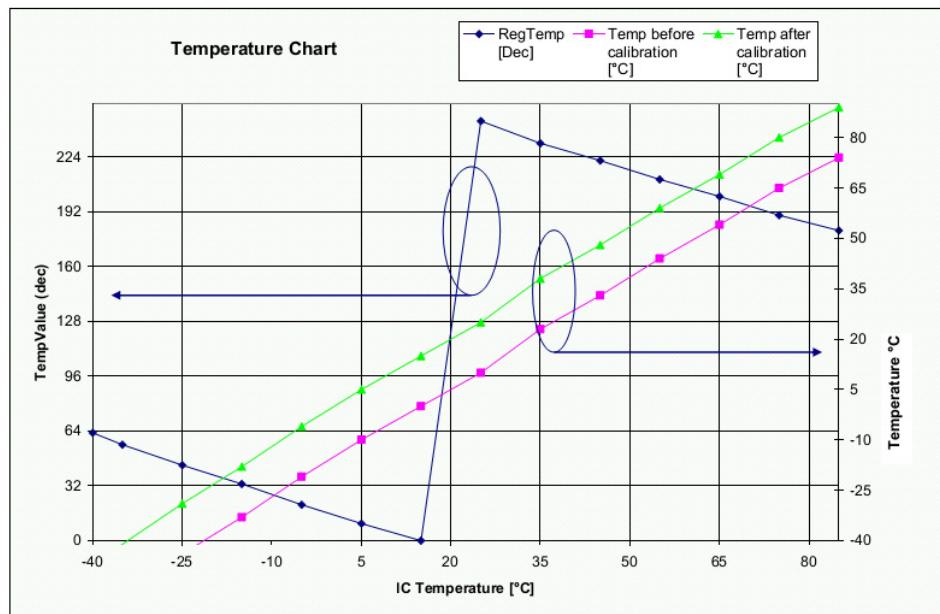


Figure 41. Temperature Sensor Response

When using the temperature sensor in the application, the following sequence should be followed:

- ◆ Set the device to Standby and wait for oscillator startup
- ◆ Set the device to FSRx mode
- ◆ Set *TempMonitorOff* = 0 (enables the sensor). It is not required to wait for the PLL Lock indication
- ◆ Wait for 140 microseconds
- ◆ Set *TempMonitorOff* = 1
- ◆ Set device back to Sleep or Standby mode
- ◆ Access temperature value in *RegTemp*

6. Description of the Registers

The register mapping depends upon whether FSK/OOK or LoRaTM mode has been selected. The following table summarises the location and function of each register and gives an overview of the changes in register mapping between both modes of operation.

6.1. Register Table Summary

Table 41 Registers Summary

| Address | Register Name | | Reset (POR) | Default (FSK) | Description | | |
|---------|------------------|-------------------------|----------------|------------------|---|---------------------------------------|--|
| | FSK/OOK Mode | LoRa TM Mode | | | FSK Mode | LoRa TM Mode | |
| 0x00 | RegFifo | | 0x00 | | FIFO read/write access | | |
| 0x01 | RegOpMode | | 0x01 | | Operating mode & LoRa TM / FSK selection | | |
| 0x02 | RegBitrateMsb | Unused | 0x1A | | Bit Rate setting, Most Significant Bits | | |
| 0x03 | RegBitrateLsb | | 0x0B | | Bit Rate setting, Least Significant Bits | | |
| 0x04 | RegFdevMsb | | 0x00 | | Frequency Deviation setting, Most Significant Bits | | |
| 0x05 | RegFdevLsb | | 0x52 | | Frequency Deviation setting, Least Significant Bits | | |
| 0x06 | RegFrFMsb | | 0x6C | | RF Carrier Frequency, Most Significant Bits | | |
| 0x07 | RegFrFMid | | 0x80 | | RF Carrier Frequency, Intermediate Bits | | |
| 0x08 | RegFrFLsb | | 0x00 | | RF Carrier Frequency, Least Significant Bits | | |
| 0x09 | RegPaConfig | | 0x4F | | PA selection and Output Power control | | |
| 0x0A | RegPaRamp | | 0x09 | | Control of PA ramp time, low phase noise PLL | | |
| 0x0B | RegOcp | | 0x2B | | Over Current Protection control | | |
| 0x0C | RegLna | | 0x20 | | LNA settings | | |
| 0x0D | RegRxConfig | RegFifoAddrPtr | 0x08 | 0x0E | AFC, AGC, ctrl | FIFO SPI pointer | |
| 0x0E | RegRssiConfig | RegFifoTxBaseAddr | 0x02 | | RSSI | Start Tx data | |
| 0x0F | RegRssiCollision | RegFifoRxBaseAddr | 0x0A | | RSSI Collision detector | Start Rx data | |
| 0x10 | RegRssiThresh | FifoRxCurrentAddr | 0xFF | | RSSI Threshold control | Start address of last packet received | |
| 0x11 | RegRssiValue | RegIRQFlagsMask | n/a | n/a | RSSI value in dBm | Optional IRQ flag mask | |
| 0x12 | RegRxBw | RegIRQFlags | 0x15 | | Channel Filter BW Control | IRQ flags | |
| 0x13 | RegAfcBw | RegRxNbBytes | 0x0B | | AFC Channel Filter BW | Number of received bytes | |
| 0x14 | RegOokPeak | RegRxHeaderCntValueMsb | 0x28 | | OOK demodulator | Number of valid headers received | |
| 0x15 | RegOokFix | RegRxHeaderCntValueLsb | 0x0C | | Threshold of the OOK demod | | |
| 0x16 | RegOokAvg | RegRxPacketCntValueMsb | 0x12 | | Average of the OOK demod | Number of valid packets received | |
| 0x17 | Reserved17 | RegRxPacketCntValueLsb | 0x47 | | - | | |
| 0x18 | Reserved18 | RegModemStat | 0x32 | | - | Live LoRa TM modem status | |
| 0x19 | Reserved19 | RegPktSnrValue | 0x3E | | - | Espimation of last packet SNR | |
| 0x1A | RegAfcFei | RegPktRssiValue | 0x00 | | AFC and FEI control | RSSI of last packet | |

| Address | Register Name | | Reset (POR) | Default (FSK) | Description | |
|---------------|------------------------|----------------------------|----------------|------------------|--|--------------------------------------|
| | FSK/OOK Mode | LoRa™ Mode | | | FSK Mode | LoRa™ Mode |
| 0x1B | RegAfcMsb | RegRssiValue | 0x00 | n/a | Frequency correction value of the AFC | Current RSSI |
| 0x1C | RegAfcLsb | RegHopChannel | 0x00 | n/a | | FHSS start channel |
| 0x1D | RegFeiMsb | RegModemConfig 1 | 0x00 | n/a | Value of the calculated frequency error | Modem PHY config 1 |
| 0x1E | RegFeiLsb | RegModemConfig 2 | 0x00 | n/a | | Modem PHY config 2 |
| 0x1F | RegPreambleDe- tect | RegSymbTimeout Lsb | 0x40 | 0xAA | Settings of the Preamble Detector | Receiver timeout value |
| 0x20 | RegRxTimeout1 | RegPreambleMsb | 0x00 | | Timeout Rx request and RSSI | |
| 0x21 | RegRxTimeout2 | RegPreambleLsb | 0x00 | | Timeout RSSI and Pay- loadReady | Size of preamble |
| 0x22 | RegRxTimeout3 | RegPay- loadLength | 0x00 | | Timeout RSSI and SyncAd- dress | LoRaTM payload length |
| 0x23 | RegRxDelay | RegMaxPayloadL ength | 0x00 | | Delay between Rx cycles | LoRaTM maximum pay- load length |
| 0x24 | RegOsc | RegHopPeriod | 0x05 | 0x07 | RC Oscillators Settings, CLK- OUT frequency | FHSS Hop period |
| 0x25 | RegPreambleMsb | RegFifoRxByteAd dr | 0x00 | | Preamble length, MSB | Address of last byte written in FIFO |
| 0x26 | RegPreambleLsb | RegModemCon- fig3 | 0x03 | | Preamble length, LSB | Modem PHY config 3 |
| 0x27 | RegSyncConfig | RESERVED | 0x93 | | Sync Word Recognition control | RESERVED |
| 0x28 | RegSyncValue1 | RegFeiMsb | 0x55 | 0x01 | Sync Word bytes 1 | Estimated frequency error |
| 0x29 | RegSyncValue2 | RegFeiMid | 0x55 | 0x01 | Sync Word bytes 2 | |
| 0x2A | RegSyncValue3 | RegFeiLsb | 0x55 | 0x01 | Sync Word bytes 3 | |
| 0x2B | RegSyncValue4 | RESERVED | 0x55 | 0x01 | Sync Word bytes 4 | |
| 0x2C | RegSyncValue5 | RegRssiWide- band | 0x55 | 0x01 | Sync Word bytes 5 | Wideband RSSI meas- urement |
| 0x2D- 0x2F | RegSyncValue6-8 | RESERVED | 0x55 | 0x01 | Sync Word bytes, 6 to 8 | RESERVED |
| 0x30 | RegPacketConfig1 | RESERVED | 0x90 | | Packet mode settings | |
| 0x31 | RegPacketConfig2 | RegDetectOpti- mize | 0x40 | | Packet mode settings | LoRa detection Optimize for SF6 |
| 0x32 | RegPayloadLength | RESERVED | 0x40 | | Payload length setting | RESERVED |
| 0x33 | RegNodeAdrs | RegInvertIQ | 0x00 | | Node address | Invert LoRa I and Q signals |
| 0x34 | RegBroadcastAdrs | RESERVED | 0x00 | | Broadcast address | RESERVED |
| 0x35 | RegFifoThresh | | 0x0F | 0x1F | Fifo threshold, Tx start condition | |
| 0x36 | RegSeqConfig1 | | 0x00 | | Top level Sequencer settings | |
| 0x37 | RegSeqConfig2 | RegDetection- Threshold | 0x00 | | Top level Sequencer settings | LoRa detection threshold for SF6 |
| 0x38 | RegTimerResol | RESERVED | 0x00 | | Timer 1 and 2 resolution control | RESERVED |
| 0x39 | RegTimer1Coef | RegSyncWord | 0xF5 | 0x12 | Timer 1 setting | LoRa Sync Word |

| Address | Register Name | | Reset (POR) | Default (FSK) | Description | | | |
|---------|----------------|------------|----------------|------------------|---|------------|--|--|
| | FSK/OOK Mode | LoRa™ Mode | | | FSK Mode | LoRa™ Mode | | |
| 0x3A | RegTimer2Coef | RESERVED | 0x20 | | Timer 2 setting | RESERVED | | |
| 0x3B | RegImageCal | | 0x82 | 0x02 | Image calibration engine control | | | |
| 0x3C | RegTemp | | - | | Temperature Sensor value | | | |
| 0x3D | RegLowBat | | 0x02 | | Low Battery Indicator Settings | | | |
| 0x3E | RegIRQFlags1 | | 0x80 | | Status register: PLL Lock state, Timeout, RSSI | | | |
| 0x3F | RegIRQFlags2 | | 0x40 | | Status register: FIFO handling flags, Low Battery | | | |
| 0x40 | RegDioMapping1 | | 0x00 | | Mapping of pins DIO0 to DIO3 | | | |
| 0x41 | RegDioMapping2 | | 0x00 | | Mapping of pins DIO4 and DIO5, ClkOut frequency | | | |
| 0x42 | RegVersion | | 0x12 | | Semtech ID relating the silicon revision | | | |
| 0x44 | RegPIIHop | Unused | 0x2D | | Control the fast frequency hopping mode | Unused | | |
| 0x4B | RegTxo | | 0x09 | | TCXO or XTAL input setting | | | |
| 0x4D | RegPaDac | | 0x84 | | Higher power settings of the PA | | | |
| 0x5B | RegFormerTemp | | - | | Stored temperature during the former IQ Calibration | | | |
| 0x5D | RegBitRateFrac | Unused | 0x00 | | Fractional part in the Bit Rate division ratio | Unused | | |
| 0x61 | RegAgcRef | | 0x13 | | Adjustment of the AGC thresholds | | | |
| 0x62 | RegAgcThresh1 | | 0x0E | | | | | |
| 0x63 | RegAgcThresh2 | | 0x5B | | | | | |
| 0x64 | RegAgcThresh3 | | 0xDB | | | | | |
| 0x70 | RegPLL | | 0xD0 | | Control of the PLL bandwidth | | | |
| others | RegTest | | - | | Internal test registers. Do not overwrite | | | |

- Note**
- Reset values are automatically refreshed in the chip at Power On Reset
 - Default values are the Semtech recommended register values, optimizing the device operation
 - Registers for which the Default value differs from the Reset value are denoted by a * in the tables of section 6.2

6.2. FSK/OOK Mode Register Map

This section details the SX1276/77/78/79 register mapping and the precise contents of each register in FSK/OOK mode.

Convention: r: read, w: write, t:trigger, c: clear

Table 42 Register Map

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-------------------------------|------|--------------------|------|---------------|--|
| RegFifo (0x00) | 7-0 | Fifo | rw | 0x00 | FIFO data input/output |
| Registers for Common settings | | | | | |
| RegOpMode (0x01) | 7 | LongRangeMode | r | 0x00 | 0 → FSK/OOK Mode 1 → LoRa™ Mode This bit can be modified only in Sleep mode. A write operation on other device modes is ignored. |
| | 6-5 | ModulationType | rw | 0x00 | Modulation scheme: 00 → FSK 01 → OOK 10 → 11 → reserved |
| | 4 | reserved | r | 0x0 | reserved |
| | 3 | LowFrequencyModeOn | rw | 0x01 | Access Low Frequency Mode registers (from address 0x61 on) 0 → High Frequency Mode (access to HF test registers) 1 → Low Frequency Mode (access to LF test registers) |
| | 2-0 | Mode | rw | 0x01 | Transceiver modes 000 → Sleep mode 001 → Stdby mode 010 → FS mode TX (FSTx) 011 → Transmitter mode (Tx) 100 → FS mode RX (FSRx) 101 → Receiver mode (Rx) 110 → reserved 111 → reserved |
| RegBitrateMsb (0x02) | 7-0 | BitRate(15:8) | rw | 0x1a | MSB of Bit Rate (chip rate if Manchester encoding is enabled) |
| RegBitrateLsb (0x03) | 7-0 | BitRate(7:0) | rw | 0x0b | LSB of bit rate (chip rate if Manchester encoding is enabled) $\text{BitRate} = \frac{\text{FXOSC}}{\text{BitRate}(15,0) + \frac{\text{BitrateFrac}}{16}}$ Default value: 4.8 kb/s |
| RegFdevMsb (0x04) | 7-6 | reserved | rw | 0x00 | reserved |
| | 5-0 | Fdev(13:8) | rw | 0x00 | MSB of the frequency deviation |
| RegFdevLsb (0x05) | 7-0 | Fdev(7:0) | rw | 0x52 | LSB of the frequency deviation $\text{Fdev} = \text{Fstep} \times \text{Fdev}(15,0)$ Default value: 5 kHz |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-------------------------------|------|-------------------|------|---------------|---|
| RegFrMsb (0x06) | 7-0 | Fr(23:16) | rw | 0x6c | MSB of the RF carrier frequency |
| RegFrMid (0x07) | 7-0 | Fr(15:8) | rw | 0x80 | MSB of the RF carrier frequency |
| RegFrLsb (0x08) | 7-0 | Fr(7:0) | rw | 0x00 | LSB of RF carrier frequency $Fr = Fstep \times Frf(23:0)$ Default value: 434.000 MHz The RF frequency is taken into account internally only when: - entering FSRX/FSTX modes - re-starting the receiver |
| Registers for the Transmitter | | | | | |
| RegPaConfig (0x09) | 7 | PaSelect | rw | 0x00 | Selects PA output pin 0 → RFO pin. Maximum power of +14 dBm 1 → PA_BOOST pin. Maximum power of +20 dBm |
| | 6-4 | MaxPower | rw | 0x04 | Select max output power: Pmax=10.8+0.6*MaxPower [dBm] |
| | 3-0 | OutputPower | rw | 0x0f | Pout=Pmax-(15-OutputPower) if PaSelect = 0 (RFO pins) Pout=17-(15-OutputPower) if PaSelect = 1 (PA_BOOST pin) |
| RegPaRamp (0x0A) | 7 | unused | r | 0x00 | unused |
| | 6-5 | ModulationShaping | rw | 0x00 | Data shaping: In FSK: 00 → no shaping 01 → Gaussian filter BT = 1.0 10 → Gaussian filter BT = 0.5 11 → Gaussian filter BT = 0.3 In OOK: 00 → no shaping 01 → filtering with fcutoff = bit_rate 10 → filtering with fcutoff = 2*bit_rate (for bit_rate < 125 kb/s) 11 → reserved |
| | 4 | reserved | rw | 0x00 | reserved |
| | 3-0 | PaRamp | rw | 0x09 | Rise/Fall time of ramp up/down in FSK 0000 → 3.4 ms 0001 → 2 ms 0010 → 1 ms 0011 → 500 us 0100 → 250 us 0101 → 125 us 0110 → 100 us 0111 → 62 us 1000 → 50 us 1001 → 40 us (d) 1010 → 31 us 1011 → 25 us 1100 → 20 us 1101 → 15 us 1110 → 12 us 1111 → 10 us |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|----------------------------|------|---------------|------|---------------|---|
| RegOcp (0x0B) | 7-6 | unused | r | 0x00 | unused |
| | 5 | OcpOn | rw | 0x01 | Enables overload current protection (OCP) for the PA: 0 → OCP disabled 1 → OCP enabled |
| | 4-0 | OcpTrim | rw | 0x0b | Trimming of OCP current: $I_{max} = 45 + 5 \cdot OcpTrim$ [mA] if $OcpTrim \leq 15$ (120 mA) / $I_{max} = -30 + 10 \cdot OcpTrim$ [mA] if $15 < OcpTrim \leq 27$ (130 to 240 mA) $I_{max} = 240$ mA for higher settings Default $I_{max} = 100$ mA |
| Registers for the Receiver | | | | | |
| RegLna (0x0C) | 7-5 | LnaGain | rw | 0x01 | <p>LNA gain setting: 000 → reserved 001 → G1 = highest gain 010 → G2 = highest gain – 6 dB 011 → G3 = highest gain – 12 dB 100 → G4 = highest gain – 24 dB 101 → G5 = highest gain – 36 dB 110 → G6 = highest gain – 48 dB 111 → reserved</p> <p>Note: Reading this address always returns the current LNA gain (which may be different from what had been previously selected if AGC is enabled).</p> |
| | 4-3 | LnaBoostLf | rw | 0x00 | Low Frequency (RFI_LF) LNA current adjustment 00 → Default LNA current Other → Reserved |
| | 2 | reserved | rw | 0x00 | reserved |
| | 1-0 | LnaBoostHf | rw | 0x00 | High Frequency (RFI_HF) LNA current adjustment 00 → Default LNA current 11 → Boost on, 150% LNA current |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|----------------------------|------|-------------------------|------|---------------|--|
| RegRxConfig (0x0d) | 7 | RestartRxOnCollision | rw | 0x00 | Turns on the mechanism restarting the receiver automatically if it gets saturated or a packet collision is detected 0 → No automatic Restart 1 → Automatic restart On |
| | 6 | RestartRxWithoutPllLock | wt | 0x00 | Triggers a manual Restart of the Receiver chain when set to 1. Use this bit when there is no frequency change, RestartRxWithPllLock otherwise. |
| | 5 | RestartRxWithPllLock | wt | 0x00 | Triggers a manual Restart of the Receiver chain when set to 1. Use this bit when there is a frequency change, requiring some time for the PLL to re-lock. |
| | 4 | AfcAutoOn | rw | 0x00 | 0 → No AFC performed at receiver startup 1 → AFC is performed at each receiver startup |
| | 3 | AgcAutoOn | rw | 0x01 | 0 → LNA gain forced by the LnaGain Setting 1 → LNA gain is controlled by the AGC |
| | 2-0 | RxTrigger | rw | 0x06 * | Selects the event triggering AGC and/or AFC at receiver startup. See Table 24 for a description. |
| RegRssiConfig (0x0e) | 7-3 | RssiOffset | rw | 0x00 | Signed RSSI offset, to compensate for the possible losses/gains in the front-end (LNA, SAW filter...) 1dB / LSB, 2's complement format |
| | 2-0 | RssiSmoothing | rw | 0x02 | Defines the number of samples taken to average the RSSI result: 000 → 2 samples used 001 → 4 samples used 010 → 8 samples used 011 → 16 samples used 100 → 32 samples used 101 → 64 samples used 110 → 128 samples used 111 → 256 samples used |
| RegRssiCollision (0x0f) | 7-0 | RssiCollisionThreshold | rw | 0x0a | Sets the threshold used to consider that an interferer is detected, witnessing a packet collision. 1dB/LSB (only RSSI increase) Default: 10dB |
| RegRssiThresh (0x10) | 7-0 | RssiThreshold | rw | 0xff | RSSI trigger level for the Rssi interrupt: - RssiThreshold / 2 [dBm] |
| RegRssiValue (0x11) | 7-0 | RssiValue | r | - | Absolute value of the RSSI in dBm, 0.5dB steps. RSSI = - RssiValue/2 [dBm] |
| RegRxBw (0x12) | 7 | unused | r | - | unused |
| | 6-5 | reserved | rw | 0x00 | reserved |
| | 4-3 | RxBwMant | rw | 0x02 | Channel filter bandwidth control: 00 → RxBwMant = 16 10 → RxBwMant = 24 01 → RxBwMant = 20 11 → reserved |
| | 2-0 | RxBwExp | rw | 0x05 | Channel filter bandwidth control |
| RegAfcBw (0x13) | 7-5 | reserved | rw | 0x00 | reserved |
| | 4-3 | RxBwMantAfc | rw | 0x01 | RxBwMant parameter used during the AFC |
| | 2-0 | RxBwExpAfc | rw | 0x03 | RxBwExp parameter used during the AFC |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|----------------------------|------|----------------------|------|----------------------|---|
| RegOokPeak (0x14) | 7-6 | reserved | rw | 0x00 | reserved |
| | 5 | BitSyncOn | rw | 0x01 | Enables the Bit Synchronizer. 0 → Bit Sync disabled (not possible in Packet mode) 1 → Bit Sync enabled |
| | 4-3 | OokThreshType | rw | 0x01 | Selects the type of threshold in the OOK data slicer: 00 → fixed threshold 10 → average mode 01 → peak mode (default) 11 → reserved |
| | 2-0 | OokPeakTheshStep | rw | 0x00 | Size of each decrement of the RSSI threshold in the OOK demodulator: 000 → 0.5 dB 001 → 1.0 dB 010 → 1.5 dB 011 → 2.0 dB 100 → 3.0 dB 101 → 4.0 dB 110 → 5.0 dB 111 → 6.0 dB |
| RegOokFix (0x15) | 7-0 | OokFixedThreshold | rw | 0x0C | Fixed threshold for the Data Slicer in OOK mode Floor threshold for the Data Slicer in OOK when Peak mode is used |
| RegOokAvg (0x16) | 7-5 | OokPeakThreshDec | rw | 0x00 | Period of decrement of the RSSI threshold in the OOK demodulator: 000 → once per chip 001 → once every 2 chips 010 → once every 4 chips 011 → once every 8 chips 100 → twice in each chip 101 → 4 times in each chip 110 → 8 times in each chip 111 → 16 times in each chip |
| | 4 | reserved | rw | 0x01 | reserved |
| | 3-2 | OokAverageOffset | rw | 0x00 | Static offset added to the threshold in average mode in order to reduce glitching activity (OOK only): 00 → 0.0 dB 10 → 4.0 dB 01 → 2.0 dB 11 → 6.0 dB |
| | 1-0 | OokAverageThreshFilt | rw | 0x02 | Filter coefficients in average mode of the OOK demodulator: 00 → $f_C \approx \text{chip rate} / 32.\pi$ 01 → $f_C \approx \text{chip rate} / 8.\pi$ 10 → $f_C \approx \text{chip rate} / 4.\pi$ 11 → $f_C \approx \text{chip rate} / 2.\pi$ |
| RegRes17 to RegRes19 | 7-0 | reserved | rw | 0x47 0x32 0x3E | reserved. Keep the Reset values. |
| RegAfcFei (0x1a) | 7-5 | unused | r | - | unused |
| | 4 | AgcStart | wt | 0x00 | Triggers an AGC sequence when set to 1. |
| | 3 | reserved | rw | 0x00 | reserved |
| | 2 | unused | - | - | unused |
| | 1 | AfcClear | wc | 0x00 | Clear AFC register set in Rx mode. Always reads 0. |
| | 0 | AfcAutoClearOn | rw | 0x00 | Only valid if AfcAutoOn is set 0 → AFC register is not cleared at the beginning of the automatic AFC phase 1 → AFC register is cleared at the beginning of the automatic AFC phase |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|----------------------------------|------|----------------------|------|---------------|---|
| RegAfcMsb (0x1b) | 7-0 | AfcValue(15:8) | rw | 0x00 | MSB of the AfcValue, 2's complement format. Can be used to overwrite the current AFC value |
| RegAfcLsb (0x1c) | 7-0 | AfcValue(7:0) | rw | 0x00 | LSB of the AfcValue, 2's complement format. Can be used to overwrite the current AFC value |
| RegFeiMsb (0x1d) | 7-0 | FeiValue(15:8) | rw | - | MSB of the measured frequency offset, 2's complement. Must be read before RegFeiLsb. |
| RegFeiLsb (0x1e) | 7-0 | FeiValue(7:0) | rw | - | LSB of the measured frequency offset, 2's complement <i>Frequency error = FeiValue x Fstep</i> |
| RegPreambleDetect (0x1f) | 7 | PreambleDetectorOn | rw | 0x01 * | Enables Preamble detector when set to 1. The AGC settings supersede this bit during the startup / AGC phase. 0 → Turned off 1 → Turned on |
| | 6-5 | PreambleDetectorSize | rw | 0x01 * | Number of Preamble bytes to detect to trigger an interrupt 00 → 1 byte 01 → 2 bytes 10 → 3 bytes 11 → Reserved |
| | 4-0 | PreambleDetectorTol | rw | 0x0A * | Number of chip errors tolerated over PreambleDetectorSize. 4 chips per bit. |
| RegRxTimeout1 (0x20) | 7-0 | TimeoutRxRssi | rw | 0x00 | <i>Timeout</i> interrupt is generated $\text{TimeoutRxRssi}^*16^*\text{T}_{\text{bit}}$ after switching to Rx mode if <i>Rssi</i> interrupt doesn't occur (i.e. <i>RssiValue > RssiThreshold</i>) 0x00: <i>TimeoutRxRssi</i> is disabled |
| RegRxTimeout2 (0x21) | 7-0 | TimeoutRxPreamble | rw | 0x00 | <i>Timeout</i> interrupt is generated $\text{TimeoutRxPreamble}^*16^*\text{T}_{\text{bit}}$ after switching to Rx mode if <i>Preamble</i> interrupt doesn't occur 0x00: <i>TimeoutRxPreamble</i> is disabled |
| RegRxTimeout3 (0x22) | 7-0 | TimeoutSignalSync | rw | 0x00 | <i>Timeout</i> interrupt is generated $\text{TimeoutSignalSync}^*16^*\text{T}_{\text{bit}}$ after the Rx mode is programmed, if <i>SyncAddress</i> doesn't occur 0x00: <i>TimeoutSignalSync</i> is disabled |
| RegRxDelay (0x23) | 7-0 | InterPacketRxDelay | rw | 0x00 | Additional delay before an automatic receiver restart is launched: Delay = <i>InterPacketRxDelay</i> *4*Tbit |
| RC Oscillator registers | | | | | |
| RegOsc (0x24) | 7-4 | unused | r | - | unused |
| | 3 | RcCalStart | wt | 0x00 | Triggers the calibration of the RC oscillator when set. Always reads 0. RC calibration must be triggered in Standby mode. |
| | 2-0 | ClkOut | rw | 0x07 * | Selects CLKOUT frequency: 000 → FXOSC 001 → FXOSC / 2 010 → FXOSC / 4 011 → FXOSC / 8 100 → FXOSC / 16 101 → FXOSC / 32 110 → RC (automatically enabled) 111 → OFF |
| Packet Handling registers | | | | | |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|--------------------------|------|--------------------|------|---------------|--|
| RegPreambleMsb (0x25) | 7-0 | PreambleSize(15:8) | rw | 0x00 | Size of the preamble to be sent (from <i>TxStartCondition</i> fulfilled). (MSB byte) |
| RegPreambleLsb (0x26) | 7-0 | PreambleSize(7:0) | rw | 0x03 | Size of the preamble to be sent (from <i>TxStartCondition</i> fulfilled). (LSB byte) |
| RegSyncConfig (0x27) | 7-6 | AutoRestartRxMode | rw | 0x02 | Controls the automatic restart of the receiver after the reception of a valid packet (PayloadReady or CrcOk): 00 → Off 01 → On, without waiting for the PLL to re-lock 10 → On, wait for the PLL to lock (frequency changed) 11 → reserved |
| | 5 | PreamblePolarity | rw | 0x00 | Sets the polarity of the Preamble 0 → 0xAA (default) 1 → 0x55 |
| | 4 | SyncOn | rw | 0x01 | Enables the Sync word generation and detection: 0 → Off 1 → On |
| | 3 | reserved | rw | 0x00 | reserved |
| | 2-0 | SyncSize | rw | 0x03 | Size of the Sync word: (SyncSize + 1) bytes, (SyncSize) bytes if <i>ioHomeOn</i> =1 |
| RegSyncValue1 (0x28) | 7-0 | SyncValue(63:56) | rw | 0x01* | 1 st byte of Sync word. (MSB byte) Used if <i>SyncOn</i> is set. |
| RegSyncValue2 (0x29) | 7-0 | SyncValue(55:48) | rw | 0x01* | 2 nd byte of Sync word Used if <i>SyncOn</i> is set and (SyncSize +1) >= 2. |
| RegSyncValue3 (0x2a) | 7-0 | SyncValue(47:40) | rw | 0x01* | 3 rd byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) >= 3. |
| RegSyncValue4 (0x2b) | 7-0 | SyncValue(39:32) | rw | 0x01* | 4 th byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) >= 4. |
| RegSyncValue5 (0x2c) | 7-0 | SyncValue(31:24) | rw | 0x01* | 5 th byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) >= 5. |
| RegSyncValue6 (0x2d) | 7-0 | SyncValue(23:16) | rw | 0x01* | 6 th byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) >= 6. |
| RegSyncValue7 (0x2e) | 7-0 | SyncValue(15:8) | rw | 0x01* | 7 th byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) >= 7. |
| RegSyncValue8 (0x2f) | 7-0 | SyncValue(7:0) | rw | 0x01* | 8 th byte of Sync word. Used if <i>SyncOn</i> is set and (SyncSize +1) = 8. |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|----------------------------|------|---------------------|------|---------------|---|
| RegPacketConfig1 (0x30) | 7 | PacketFormat | rw | 0x01 | Defines the packet format used: 0 → Fixed length 1 → Variable length |
| | 6-5 | DcFree | rw | 0x00 | Defines DC-free encoding/decoding performed: 00 → None (Off) 01 → Manchester 10 → Whitening 11 → reserved |
| | 4 | CrcOn | rw | 0x01 | Enables CRC calculation/check (Tx/Rx): 0 → Off 1 → On |
| | 3 | CrcAutoClearOff | rw | 0x00 | Defines the behavior of the packet handler when CRC check fails: 0 → Clear FIFO and restart new packet reception. No <i>PayloadReady</i> interrupt issued. 1 → Do not clear FIFO. <i>PayloadReady</i> interrupt issued. |
| | 2-1 | AddressFiltering | rw | 0x00 | Defines address based filtering in Rx: 00 → None (Off) 01 → Address field must match <i>NodeAddress</i> 10 → Address field must match <i>NodeAddress</i> or <i>BroadcastAddress</i> 11 → reserved |
| | 0 | CrcWhiteningType | rw | 0x00 | Selects the CRC and whitening algorithms: 0 → CCITT CRC implementation with standard whitening 1 → IBM CRC implementation with alternate whitening |
| RegPacketConfig2 (0x31) | 7 | unused | r | - | unused |
| | 6 | DataMode | rw | 0x01 | Data processing mode: 0 → Continuous mode 1 → Packet mode |
| | 5 | IoHomeOn | rw | 0x00 | Enables the io-homecontrol® compatibility mode 0 → Disabled 1 → Enabled |
| | 4 | IoHomePowerFrame | rw | 0x00 | reserved - Linked to io-homecontrol® compatibility mode |
| | 3 | BeaconOn | rw | 0x00 | Enables the Beacon mode in Fixed packet format |
| | 2-0 | PayloadLength(10:8) | rw | 0x00 | Packet Length Most significant bits |
| RegPayloadLength (0x32) | 7-0 | PayloadLength(7:0) | rw | 0x40 | If PacketFormat = 0 (fixed), payload length. If PacketFormat = 1 (variable), max length in Rx, not used in Tx. |
| RegNodeAdrs (0x33) | 7-0 | NodeAddress | rw | 0x00 | Node address used in address filtering. |
| RegBroadcastAdrs (0x34) | 7-0 | BroadcastAddress | rw | 0x00 | Broadcast address used in address filtering. |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-------------------------|------|-------------------|------|---------------|---|
| RegFifoThresh (0x35) | 7 | TxStartCondition | rw | 0x01 * | Defines the condition to start packet transmission: 0 → <i>FifoLevel</i> (i.e. the number of bytes in the FIFO exceeds <i>FifoThreshold</i>) 1 → <i>FifoEmpty</i> goes low (i.e. at least one byte in the FIFO) |
| | 6 | unused | r | - | unused |
| | 5-0 | FifoThreshold | rw | 0x0f | Used to trigger <i>FifoLevel</i> interrupt, when: number of bytes in FIFO >= FifoThreshold + 1 |
| Sequencer registers | | | | | |
| RegSeqConfig1 (0x36) | 7 | SequencerStart | wt | 0x00 | Controls the top level Sequencer When set to '1', executes the "Start" transition. The sequencer can only be enabled when the chip is in Sleep or Standby mode. |
| | 6 | SequencerStop | wt | 0x00 | Forces the Sequencer Off. Always reads '0' |
| | 5 | IdleMode | rw | 0x00 | Selects chip mode during the state: 0: Standby mode 1: Sleep mode |
| | 4-3 | FromStart | rw | 0x00 | Controls the Sequencer transition when <i>SequencerStart</i> is set to 1 in Sleep or Standby mode: 00: to LowPowerSelection 01: to Receive state 10: to Transmit state 11: to Transmit state on a <i>FifoLevel</i> interrupt |
| | 2 | LowPowerSelection | rw | 0x00 | Selects the Sequencer LowPower state after a to <i>LowPowerSelection</i> transition: 0: SequencerOff state with chip on Initial mode 1: Idle state with chip on <i>Standby</i> or <i>Sleep</i> mode depending on <i>IdleMode</i> <i>Note:</i> Initial mode is the chip LowPower mode at Sequencer Start. |
| | 1 | FromIdle | rw | 0x00 | Controls the Sequencer transition from the Idle state on a T1 interrupt: 0: to Transmit state 1: to Receive state |
| | 0 | FromTransmit | rw | 0x00 | Controls the Sequencer transition from the Transmit state: 0: to LowPowerSelection on a <i>PacketSent</i> interrupt 1: to Receive state on a <i>PacketSent</i> interrupt |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-------------------------|------|--------------------|------|---------------|--|
| RegSeqConfig2 (0x37) | 7-5 | FromReceive | rw | 0x00 | <p>Controls the Sequencer transition from the Receive state 000 and 111: unused</p> <p>001: to PacketReceived state on a <i>PayloadReady</i> interrupt</p> <p>010: to LowPowerSelection on a <i>PayloadReady</i> interrupt</p> <p>011: to PacketReceived state on a <i>CrcOk</i> interrupt (1)</p> <p>100: to SequencerOff state on a <i>Rssi</i> interrupt</p> <p>101: to SequencerOff state on a <i>SyncAddress</i> interrupt</p> <p>110: to SequencerOff state on a <i>PreambleDetect</i> interrupt</p> <p>Irrespective of this setting, transition to LowPowerSelection on a T2 interrupt</p> <p>(1) If the CRC is wrong (corrupted packet, with CRC on but <i>CrcAutoClearOn</i>=0), the <i>PayloadReady</i> interrupt will drive the sequencer to RxTimeout state.</p> |
| | 4-3 | FromRxTimeout | rw | 0x00 | <p>Controls the state-machine transition from the Receive state on a <i>RxTimeout</i> interrupt (and on <i>PayloadReady</i> if FromReceive = 011):</p> <p>00: to Receive State, via ReceiveRestart</p> <p>01: to Transmit state</p> <p>10: to LowPowerSelection</p> <p>11: to SequencerOff state</p> <p><i>Note:</i> <i>RxTimeout</i> interrupt is a <i>TimeoutRxRssi</i>, <i>TimeoutRxPreamble</i> or <i>TimeoutSignalSync</i> interrupt</p> |
| | 2-0 | FromPacketReceived | rw | 0x00 | <p>Controls the state-machine transition from the PacketReceived state:</p> <p>000: to SequencerOff state</p> <p>001: to Transmit state on a <i>FifoEmpty</i> interrupt</p> <p>010: to LowPowerSelection</p> <p>011: to Receive via FS mode, if frequency was changed</p> <p>100: to Receive state (no frequency change)</p> |
| RegTimerResol (0x38) | 7-4 | unused | r | - | unused |
| | 3-2 | Timer1Resolution | rw | 0x00 | <p>Resolution of Timer 1</p> <p>00: Timer1 disabled</p> <p>01: 64 us</p> <p>10: 4.1 ms</p> <p>11: 262 ms</p> |
| | 1-0 | Timer2Resolution | rw | 0x00 | <p>Resolution of Timer 2</p> <p>00: Timer2 disabled</p> <p>01: 64 us</p> <p>10: 4.1 ms</p> <p>11: 262 ms</p> |
| RegTimer1Coef (0x39) | 7-0 | Timer1Coefficient | rw | 0xf5 | Multiplying coefficient for Timer 1 |
| RegTimer2Coef (0x3a) | 7-0 | Timer2Coefficient | rw | 0x20 | Multiplying coefficient for Timer 2 |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-----------------------|------|-----------------|------|---------------|---|
| Service registers | | | | | |
| RegImageCal (0x3b) | 7 | AutoImageCalOn | rw | 0x00 * | Controls the Image calibration mechanism 0 → Calibration of the receiver depending on the temperature is disabled 1 → Calibration of the receiver depending on the temperature enabled. |
| | 6 | ImageCalStart | wt | - | Triggers the IQ and RSSI calibration when set in Standby mode. |
| | 5 | ImageCalRunning | r | 0x00 | Set to 1 while the Image and RSSI calibration are running. Toggles back to 0 when the process is completed |
| | 4 | unused | r | - | unused |
| | 3 | TempChange | r | 0x00 | IRQ flag witnessing a temperature change exceeding TempThreshold since the last Image and RSSI calibration: 0 → Temperature change lower than TempThreshold 1 → Temperature change greater than TempThreshold |
| | 2-1 | TempThreshold | rw | 0x01 | Temperature change threshold to trigger a new I/Q calibration 00 → 5 °C 01 → 10 °C 10 → 15 °C 11 → 20 °C |
| | 0 | TempMonitorOff | rw | 0x00 | Controls the temperature monitor operation: 0 → Temperature monitoring done in all modes except Sleep and Standby 1 → Temperature monitoring stopped. |
| RegTemp (0x3c) | 7-0 | TempValue | r | - | Measured temperature -1°C per Lsb Needs calibration for absolute accuracy |
| RegLowBat (0x3d) | 7-4 | unused | r | - | unused |
| | 3 | LowBatOn | rw | 0x00 | Low Battery detector enable signal 0 → LowBat detector disabled 1 → LowBat detector enabled |
| | 2-0 | LowBatTrim | rw | 0x02 | Trimming of the LowBat threshold: 000 → 1.695 V 001 → 1.764 V 010 → 1.835 V (d) 011 → 1.905 V 100 → 1.976 V 101 → 2.045 V 110 → 2.116 V 111 → 2.185 V |
| Status registers | | | | | |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|------------------------|------|------------------|------|---------------|---|
| RegIrqFlags1 (0x3e) | 7 | ModeReady | r | - | Set when the operation mode requested in <i>Mode</i> , is ready - Sleep: Entering Sleep mode - Standby: XO is running - FS: PLL is locked - Rx: RSSI sampling starts - Tx: PA ramp-up completed Cleared when changing the operating mode. |
| | 6 | RxReady | r | - | Set in Rx mode, after RSSI, AGC and AFC. Cleared when leaving Rx. |
| | 5 | TxReady | r | - | Set in Tx mode, after PA ramp-up. Cleared when leaving Tx. |
| | 4 | PllLock | r | - | Set (in FS, Rx or Tx) when the PLL is locked. Cleared when it is not. |
| | 3 | Rssi | rwc | - | Set in Rx when the <i>RssiValue</i> exceeds <i>RssiThreshold</i> . Cleared when leaving Rx or setting this bit to 1. |
| | 2 | Timeout | r | - | Set when a timeout occurs Cleared when leaving Rx or FIFO is emptied. |
| | 1 | PreambleDetect | rwc | - | Set when the Preamble Detector has found valid Preamble. bit clear when set to 1 |
| | 0 | SyncAddressMatch | rwc | - | Set when Sync and Address (if enabled) are detected. Cleared when leaving Rx or FIFO is emptied. This bit is read only in Packet mode, rwc in Continuous mode |
| RegIrqFlags2 (0x3f) | 7 | FifoFull | r | - | Set when FIFO is full (i.e. contains 66 bytes), else cleared. |
| | 6 | FifoEmpty | r | - | Set when FIFO is empty, and cleared when there is at least 1 byte in the FIFO. |
| | 5 | FifoLevel | r | - | Set when the number of bytes in the FIFO strictly exceeds <i>FifoThreshold</i> , else cleared. |
| | 4 | FifoOverrun | rwc | - | Set when FIFO overrun occurs. (except in Sleep mode) Flag(s) and FIFO are cleared when this bit is set. The FIFO then becomes immediately available for the next transmission / reception. |
| | 3 | PacketSent | r | - | Set in Tx when the complete packet has been sent. Cleared when exiting Tx |
| | 2 | PayloadReady | r | - | Set in Rx when the payload is ready (i.e. last byte received and CRC, if enabled and <i>CrcAutoClearOff</i> is cleared, is Ok). Cleared when FIFO is empty. |
| | 1 | CrcOk | r | - | Set in Rx when the CRC of the payload is Ok. Cleared when FIFO is empty. |
| | 0 | LowBat | rwc | - | Set when the battery voltage drops below the Low Battery threshold. Cleared only when set to 1 by the user. |
| IO control registers | | | | | |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|--------------------------|------|-------------------|------|---------------|---|
| RegDioMapping1 (0x40) | 7-6 | Dio0Mapping | rw | 0x00 | Mapping of pins DIO0 to DIO5 See Table 18 for mapping in LoRa mode |
| | 5-4 | Dio1Mapping | rw | 0x00 | |
| | 3-2 | Dio2Mapping | rw | 0x00 | |
| | 1-0 | Dio3Mapping | rw | 0x00 | |
| RegDioMapping2 (0x41) | 7-6 | Dio4Mapping | rw | 0x00 | See Table 29 for mapping in Continuous mode See Table 30 for mapping in Packet mode |
| | 5-4 | Dio5Mapping | rw | 0x00 | |
| | 3-1 | reserved | rw | 0x00 | reserved. Retain default value |
| | 0 | MapPreambleDetect | rw | 0x00 | Allows the mapping of either <i>Rssi</i> Or <i>PreambleDetect</i> to the DIO pins, as summarized on Table 29 and Table 30 0 → <i>Rssi</i> interrupt 1 → <i>PreambleDetect</i> interrupt |
| Version register | | | | | |
| RegVersion (0x42) | 7-0 | Version | r | 0x12 | Version code of the chip. Bits 7-4 give the full revision number; bits 3-0 give the metal mask revision number. |
| Additional registers | | | | | |
| RegPIIHop (0x44) | 7 | FastHopOn | rw | 0x00 | Bypasses the main state machine for a quick frequency hop. Writing RegFrflsb will trigger the frequency change. 0 → Frf is validated when FSTx or FSRx is requested 1 → Frf is validated triggered when RegFrflsb is written |
| | 6-0 | reserved | rw | 0x2d | reserved |
| RegTcxo (0x4b) | 7-5 | reserved | rw | 0x00 | reserved. Retain default value |
| | 4 | TcxoInputOn | rw | 0x00 | Controls the crystal oscillator 0 → Crystal Oscillator with external Crystal 1 → External clipped sine TCXO AC-connected to XTA pin |
| | 3-0 | reserved | rw | 0x09 | Reserved. Retain default value. |
| RegPaDac (0x4d) | 7-3 | reserved | rw | 0x10 | reserved. Retain default value |
| | 2-0 | PaDac | rw | 0x04 | Enables the +20dBm option on PA_BOOST pin 0x04 → Default value 0x07 → +20dBm on PA_BOOST when OutputPower=1111 |
| RegFormerTemp (0x5b) | 7-0 | FormerTemp | rw | - | Temperature saved during the latest IQ (RSSI and Image) calibration. Same format as <i>TempValue</i> in <i>RegTemp</i> . |
| RegBitrateFrac (0x5d) | 7-4 | unused | r | 0x00 | unused |
| | 3-0 | BitRateFrac | rw | 0x00 | Fractional part of the bit rate divider (Only valid for FSK) If <i>BitRateFrac</i> > 0 then: $\text{BitRate} = \frac{\text{FXOSC}}{\text{BitRate}(15,0) + \frac{\text{BitRateFrac}}{16}}$ |

| Name (Address) | Bits | Variable Name | Mode | Default value | FSK/OOK Description |
|-------------------------|------|-------------------|------|------------------|---|
| RegAgcRef (0x61) | 7-6 | unused | r | - | unused |
| | 5-0 | AgcReferenceLevel | rw | 0x19 | Sets the floor reference for all AGC thresholds: AGC Reference[dBm]= -174dBm+10*log(2*RxBw)+SNR+AgcReferenceLevel SNR = 8dB, fixed value |
| RegAgcThresh1 (0x62) | 7-5 | unused | r | - | unused |
| | 4-0 | AgcStep1 | rw | 0x0c | Defines the 1st AGC Threshold |
| RegAgcThresh2 (0x63) | 7-4 | AgcStep2 | rw | 0x04 | Defines the 2nd AGC Threshold: |
| | 3-0 | AgcStep3 | rw | 0x0b | Defines the 3rd AGC Threshold: |
| RegAgcThresh3 (0x64) | 7-4 | AgcStep4 | rw | 0x0c | Defines the 4th AGC Threshold: |
| | 3-0 | AgcStep5 | rw | 0x0c | Defines the 5th AGC Threshold: |

6.3. Band Specific Additional Registers

The registers in the address space from 0x61 to 0x73 are specific for operation in the lower frequency bands (below 525 MHz), or in the upper frequency bands (above 779 MHz). Their programmed value may differ, and are retained when switching from lower to high frequency and vice-versa. The access to the band specific registers is granted by enabling or disabling the bit 3 *LowFrequencyModeOn* of the *RegOpMode* register. By default, the bit *LowFrequencyModeOn* is at '1' indicating that the registers are configured for the low frequency band.

Table 43 Low Frequency Additional Registers

| Name (Address) | Bits | Variable Name | Mode | Default value | Low Frequency Additional Registers |
|---------------------------|------|-------------------|------|------------------|--|
| RegAgcRefLf (0x61) | 7-6 | unused | r | - | unused |
| | 5-0 | AgcReferenceLevel | rw | 0x19 | Sets the floor reference for all AGC thresholds: AGC Reference[dBm]= -174dBm+10*log(2*RxBw)+SNR+AgcReferenceLevel SNR = 8dB, fixed value |
| RegAgcThresh1Lf (0x62) | 7-5 | unused | r | - | unused |
| | 4-0 | AgcStep1 | rw | 0x0c | Defines the 1st AGC Threshold |
| RegAgcThresh2Lf (0x63) | 7-4 | AgcStep2 | rw | 0x04 | Defines the 2nd AGC Threshold: |
| | 3-0 | AgcStep3 | rw | 0x0b | Defines the 3rd AGC Threshold: |
| RegAgcThresh3Lf (0x64) | 7-4 | AgcStep4 | rw | 0x0c | Defines the 4th AGC Threshold: |
| | 3-0 | AgcStep5 | rw | 0x0c | Defines the 5th AGC Threshold: |
| RegPllLf (0x70) | 7-6 | PllBandwidth | rw | 0x03 | Controls the PLL bandwidth: 00 → 75 kHz 10 → 225 kHz 01 → 150 kHz 11 → 300 kHz |
| | 5-0 | reserved | rw | 0x10 | reserved. Retain default value |

Table 44 High Frequency Additional Registers

| Name (Address) | Bits | Variable Name | Mode | Default value | Low Frequency Additional Registers |
|---------------------------|------|-------------------|------|---------------|--|
| RegAgcRefHf (0x61) | 7-6 | unused | r | - | unused |
| | 5-0 | AgcReferenceLevel | rw | 0x1c | Sets the floor reference for all AGC thresholds: AGC Reference[dBm]= -174dBm+10*log(2*RxBw)+SNR+AgcReferenceLevel SNR = 8dB, fixed value |
| RegAgcThresh1Hf (0x62) | 7-5 | unused | r | - | unused |
| | 4-0 | AgcStep1 | rw | 0x0e | Defines the 1st AGC Threshold |
| RegAgcThresh2Hf (0x63) | 7-4 | AgcStep2 | rw | 0x05 | Defines the 2nd AGC Threshold: |
| | 3-0 | AgcStep3 | rw | 0x0b | Defines the 3rd AGC Threshold: |
| RegAgcThresh3Hf (0x64) | 7-4 | AgcStep4 | rw | 0x0c | Defines the 4th AGC Threshold: |
| | 3-0 | AgcStep5 | rw | 0x0c | Defines the 5th AGC Threshold: |
| RegPllHf (0x70) | 7-6 | PllBandwidth | rw | 0x03 | Controls the PLL bandwidth: 00 → 75 kHz 10 → 225 kHz 01 → 150 kHz 11 → 300 kHz |
| | 5-0 | reserved | rw | 0x10 | reserved. Retain default value |

6.4. LoRaTM Mode Register Map

This details the SX1276/77/78/79 register mapping and the precise contents of each register in LoRaTM mode.

It is essential to understand that the LoRaTM modem is controlled independently of the FSK modem. Therefore, care should be taken when accessing the registers, especially as some register may have the same name in LoRaTM or FSK mode.

The LoRa registers are only accessible when the device is set in Lora mode (and, in the same way, the FSK register are only accessible in FSK mode). However, in some cases, it may be necessary to access some of the FSK register while in LoRa mode. To this aim, the *AccessSharedReg* bit was created in the *RegOpMode* register. This bit, when set to '1', will grant access to the FSK register 0x0D up to the register 0x3F. Once the setup has been done, it is strongly recommended to clear this bit so that LoRa register can be accessed normally.

Convention: r: read, w: write, c : set to clear and t: trigger.

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa TM Description |
|--------------------------|------|--------------------|------|-------|---|
| RegFifo (0x00) | 7-0 | Fifo | rw | 0x00 | LoRa TM base-band FIFO data input/output. FIFO is cleared and not accessible when device is in SLEEP mode |
| Common Register Settings | | | | | |
| RegOpMode (0x01) | 7 | LongRangeMode | rw | 0x0 | 0 → FSK/OOK Mode 1 → LoRa TM Mode This bit can be modified only in Sleep mode. A write operation on other device modes is ignored. |
| | 6 | AccessSharedReg | rw | 0x0 | This bit operates when device is in Lora mode; if set it allows access to FSK registers page located in address space (0x0D:0x3F) while in LoRa mode 0 → Access LoRa registers page 0x0D: 0x3F 1 → Access FSK registers page (in mode LoRa) 0x0D: 0x3F |
| | 5-4 | reserved | r | 0x00 | reserved |
| | 3 | LowFrequencyModeOn | rw | 0x01 | Access Low Frequency Mode registers 0 → High Frequency Mode (access to HF test registers) 1 → Low Frequency Mode (access to LF test registers) |
| | 2-0 | Mode | rwt | 0x01 | Device modes 000 → SLEEP 001 → STDBY 010 → Frequency synthesis TX (FSTX) 011 → Transmit (TX) 100 → Frequency synthesis RX (FSRX) 101 → Receive continuous (RXCONTINUOUS) 110 → receive single (RXSINGLE) 111 → Channel activity detection (CAD) |
| (0x02) | 7-0 | reserved | r | 0x00 | - |
| (0x03) | 7-0 | reserved | r | 0x00 | - |
| (0x04) | 7-0 | reserved | rw | 0x00 | - |
| (0x05) | 7-0 | reserved | r | 0x00 | - |
| RegFrMsb (0x06) | 7-0 | Fr(23:16) | rw | 0x6c | MSB of RF carrier frequency |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|-------------------------|------|---------------|------|-------|--|
| RegFrMid (0x07) | 7-0 | FrF(15:8) | rw | 0x80 | MSB of RF carrier frequency |
| RegFrLsb (0x08) | 7-0 | FrF(7:0) | rwt | 0x00 | <p>LSB of RF carrier frequency</p> $f_{RF} = \frac{F(XOSC) \cdot Frf}{2^{19}}$ <p>Resolution is 61.035 Hz if F(XOSC) = 32 MHz. Default value is 0x6c8000 = 434 MHz. Register values must be modified only when device is in SLEEP or STAND-BY mode.</p> |
| Registers for RF blocks | | | | | |
| RegPaConfig (0x09) | 7 | PaSelect | rw | 0x00 | Selects PA output pin 0 → RFO pin. Output power is limited to +14 dBm. 1 → PA_BOOST pin. Output power is limited to +20 dBm |
| | 6-4 | MaxPower | rw | 0x04 | Select max output power: Pmax=10.8+0.6*MaxPower [dBm] |
| | 3-0 | OutputPower | rw | 0x0f | Pout=Pmax-(15-OutputPower) if PaSelect = 0 (RFO pin) Pout=17-(15-OutputPower) if PaSelect = 1 (PA_BOOST pin) |
| RegPaRamp (0x0A) | 7-5 | unused | r | - | unused |
| | 4 | reserved | rw | 0x00 | reserved |
| | 3-0 | PaRamp(3:0) | rw | 0x09 | Rise/Fall time of ramp up/down in FSK 0000 → 3.4 ms 0001 → 2 ms 0010 → 1 ms 0011 → 500 us 0100 → 250 us 0101 → 125 us 0110 → 100 us 0111 → 62 us 1000 → 50 us 1001 → 40 us 1010 → 31 us 1011 → 25 us 1100 → 20 us 1101 → 15 us 1110 → 12 us 1111 → 10 us |
| | 7-6 | unused | r | 0x00 | unused |
| RegOcp (0x0B) | 5 | OcpOn | rw | 0x01 | Enables overload current protection (OCP) for PA: 0 → OCP disabled 1 → OCP enabled |
| | 4-0 | OcpTrim | rw | 0x0b | Trimming of OCP current: Imax = 45+5*OcpTrim [mA] if OcpTrim <= 15 (120 mA) / Imax = -30+10*OcpTrim [mA] if 15 < OcpTrim <= 27 (130 to 240 mA) Imax = 240mA for higher settings Default Imax = 100mA |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|---------------------------------|------|------------------------|------|-------|---|
| RegLna (0x0C) | 7-5 | LnaGain | rwx | 0x01 | LNA gain setting: 000 → not used 001 → G1 = maximum gain 010 → G2 011 → G3 100 → G4 101 → G5 110 → G6 = minimum gain 111 → not used |
| | 4-3 | LnaBoostLf | rw | 0x00 | Low Frequency (RFI_LF) LNA current adjustment 00 → Default LNA current Other → Reserved |
| | 2 | reserved | rw | 0x00 | reserved |
| | 1-0 | LnaBoostHf | rw | 0x00 | High Frequency (RFI_HF) LNA current adjustment 00 → Default LNA current 11 → Boost on, 150% LNA current |
| Lora page registers | | | | | |
| RegFifoAddrPtr (0x0D) | 7-0 | FifoAddrPtr | rw | 0x00 | SPI interface address pointer in FIFO data buffer. |
| RegFifoTxBaseAd dr (0x0E) | 7-0 | FifoTxBaseAddr | rw | 0x80 | write base address in FIFO data buffer for TX modulator |
| RegFifoRxBaseAd dr (0x0F) | 7-0 | FifoRxBaseAddr | rw | 0x00 | read base address in FIFO data buffer for RX demodulator |
| RegFifoRxCurrent Addr (0x10) | 7-0 | FifoRxCurrentAddr | r | n/a | Start address (in data buffer) of last packet received |
| RegIRQFlagsMask (0x11) | 7 | RxTimeoutMask | rw | 0x00 | Timeout interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 6 | RxDoneMask | rw | 0x00 | Packet reception complete interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 5 | PayloadCrcErrorMask | rw | 0x00 | Payload CRC error interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 4 | ValidHeaderMask | rw | 0x00 | Valid header received in Rx mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 3 | TxDoneMask | rw | 0x00 | FIFO Payload transmission complete interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 2 | CadDoneMask | rw | 0x00 | CAD complete interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 1 | FhssChangeChannelM ask | rw | 0x00 | FHSS change channel interrupt mask: setting this bit masks the corresponding IRQ in RegIRQFlags |
| | 0 | CadDetectedMask | rw | 0x00 | Cad Detected Interrupt Mask: setting this bit masks the corresponding IRQ in RegIRQFlags |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|--------------------------------------|------|-------------------------|------|-------|---|
| RegIrqFlags (0x12) | 7 | RxTimeout | rc | 0x00 | Timeout interrupt: writing a 1 clears the IRQ |
| | 6 | RxDone | rc | 0x00 | Packet reception complete interrupt: writing a 1 clears the IRQ |
| | 5 | PayloadCrcError | rc | 0x00 | Payload CRC error interrupt: writing a 1 clears the IRQ |
| | 4 | ValidHeader | rc | 0x00 | Valid header received in Rx: writing a 1 clears the IRQ |
| | 3 | TxDone | rc | 0x00 | FIFO Payload transmission complete interrupt: writing a 1 clears the IRQ |
| | 2 | CadDone | rc | 0x00 | CAD complete: write to clear: writing a 1 clears the IRQ |
| | 1 | FhssChangeChannel | rc | 0x00 | FHSS change channel interrupt: writing a 1 clears the IRQ |
| | 0 | CadDetected | rc | 0x00 | Valid Lora signal detected during CAD operation: writing a 1 clears the IRQ |
| RegRxNbBytes (0x13) | 7-0 | FifoRxBytesNb | r | n/a | Number of payload bytes of latest packet received |
| RegRxHeaderCnt ValueMsb (0x14) | 7-0 | ValidHeaderCntMsb(15:8) | r | n/a | Number of valid headers received since last transition into Rx mode, MSB(15:8). Header and packet counters are reseted in Sleep mode. |
| RegRxHeaderCnt ValueLsb (0x15) | 7-0 | ValidHeaderCntLsb(7:0) | r | n/a | Number of valid headers received since last transition into Rx mode, LSB(7:0). Header and packet counters are reseted in Sleep mode. |
| RegRxPacketCntV alueMsb (0x16) | 7-0 | ValidPacketCntMsb(15:8) | rc | n/a | Number of valid packets received since last transition into Rx mode, MSB(15:8). Header and packet counters are reseted in Sleep mode. |
| RegRxPacketCntV alueLsb (0x17) | 7-0 | ValidPacketCntLsb(7:0) | r | n/a | Number of valid packets received since last transition into Rx mode, LSB(7:0). Header and packet counters are reseted in Sleep mode. |
| RegModemStat (0x18) | 7-5 | RxCodingRate | r | n/a | Coding rate of last header received |
| | 4 | ModemStatus | r | '1' | Modem clear |
| | 3 | | r | '0' | Header info valid |
| | 2 | | r | '0' | RX on-going |
| | 1 | | r | '0' | Signal synchronized |
| | 0 | | r | '0' | Signal detected |
| RegPktSnrValue (0x19) | 7-0 | PacketSnr | r | n/a | Estimation of SNR on last packet received. In two's compliment format multiplied by 4. $SNR[dB] = \frac{PacketSnr[two's complement]}{4}$ |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|-------------------------------|------|----------------------|------|-------|---|
| RegPktRssiValue (0x1A) | 7-0 | PacketRssi | r | n/a | RSSI of the latest packet received (dBm): $\text{RSSI[dBm]} = -157 + \text{Rssi}$ (using HF output port, SNR ≥ 0) or $\text{RSSI[dBm]} = -164 + \text{Rssi}$ (using LF output port, SNR ≥ 0) (see section 5.5.5 for details) |
| RegRssiValue (0x1B) | 7-0 | Rssi | r | n/a | Current RSSI value (dBm) $\text{RSSI[dBm]} = -157 + \text{Rssi}$ (using HF output port) or $\text{RSSI[dBm]} = -164 + \text{Rssi}$ (using LF output port) (see section 5.5.5 for details) |
| RegHopChannel (0x1C) | 7 | PlTimeout | r | n/a | PLL failed to lock while attempting a TX/RX/CAD operation 1 → PLL did not lock 0 → PLL did lock |
| | 6 | CrcOnPayload | r | n/a | CRC Information extracted from the received packet header (Explicit header mode only) 0 → Header indicates CRC off 1 → Header indicates CRC on |
| | 5-0 | FhssPresentChannel | r | n/a | Current value of frequency hopping channel in use. |
| RegModemConfig 1 (0x1D) | 7-4 | Bw | rw | 0x07 | Signal bandwidth: 0000 → 7.8 kHz 0001 → 10.4 kHz 0010 → 15.6 kHz 0011 → 20.8 kHz 0100 → 31.25 kHz 0101 → 41.7 kHz 0110 → 62.5 kHz 0111 → 125 kHz 1000 → 250 kHz 1001 → 500 kHz other values → reserved In the lower band (169MHz), signal bandwidths 8&9 are not supported) |
| | 3-1 | CodingRate | rw | '001' | Error coding rate 001 → 4/5 010 → 4/6 011 → 4/7 100 → 4/8 All other values → reserved In implicit header mode should be set on receiver to determine expected coding rate. See 4.1.1.3 |
| | 0 | ImplicitHeaderModeOn | rw | 0x0 | 0 → Explicit Header mode 1 → Implicit Header mode |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|-------------------------------|------|------------------------|------|-------|---|
| RegModemConfig 2 (0x1E) | 7-4 | SpreadingFactor | rw | 0x07 | SF rate (expressed as a base-2 logarithm) 6 → 64 chips / symbol 7 → 128 chips / symbol 8 → 256 chips / symbol 9 → 512 chips / symbol 10 → 1024 chips / symbol 11 → 2048 chips / symbol 12 → 4096 chips / symbol other values reserved. |
| | 3 | TxContinuousMode | rw | 0 | 0 → normal mode, a single packet is sent 1 → continuous mode, send multiple packets across the FIFO (used for spectral analysis) |
| | 2 | RxPayloadCrcOn | rw | 0x00 | Enable CRC generation and check on payload: 0 → CRC disable 1 → CRC enable If CRC is needed, RxPayloadCrcOn should be set: - in Implicit header mode: on Tx and Rx side - in Explicit header mode: on the Tx side alone (recovered from the header in Rx side) |
| | 1-0 | SymbTimeout(9:8) | rw | 0x00 | RX Time-Out MSB |
| RegSymbTimeoutLsb (0x1F) | 7-0 | SymbTimeout(7:0) | rw | 0x64 | RX Time-Out LSB RX operation time-out value expressed as number of symbols: $TimeOut = SymbTimeout \cdot Ts$ |
| RegPreambleMsb (0x20) | 7-0 | PreambleLength(15:8) | rw | 0x0 | Preamble length MSB, = PreambleLength + 4.25 Symbols See 4.1.1 for more details. |
| RegPreambleLsb (0x21) | 7-0 | PreambleLength(7:0) | rw | 0x8 | Preamble Length LSB |
| RegPayloadLength (0x22) | 7-0 | PayloadLength(7:0) | rw | 0x1 | Payload length in bytes. The register needs to be set in implicit header mode for the expected packet length. A 0 value is not permitted |
| RegMaxPayloadLength (0x23) | 7-0 | PayloadMaxLength(7:0) | rw | 0xff | Maximum payload length; if header payload length exceeds value a header CRC error is generated. Allows filtering of packet with a bad size. |
| RegHopPeriod (0x24) | 7-0 | FreqHoppingPeriod(7:0) | rw | 0x0 | Symbol periods between frequency hops. (0 = disabled). 1st hop always happen after the 1st header symbol |
| RegFifoRxByteAddr (0x25) | 7-0 | FifoRxByteAddrPtr | r | n/a | Current value of RX databuffer pointer (address of last byte written by Lora receiver) |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|-------------------------------------|------|---------------------|------|-------|---|
| RegModemConfig 3 (0x26) | 7-4 | Unused | r | 0x00 | |
| | 3 | LowDataRateOptimize | rw | 0x00 | 0 → Disabled 1 → Enabled; mandated for when the symbol length exceeds 16ms |
| | 2 | AgcAutoOn | rw | 0x00 | 0 → LNA gain set by register LnaGain 1 → LNA gain set by the internal AGC loop |
| | 1-0 | Reserved | rw | 0x00 | Reserved |
| (0x27) | 7-0 | PpmCorrection | rw | 0x00 | Data rate offset value, used in conjunction with AFC |
| RegFeiMsb (0x28) | 7-4 | Reserved | r | n/a | Reserved |
| | 3-0 | FreqError(19:16) | r | 0x0 | Estimated frequency error from modem MSB of RF Frequency Error $F_{Error} = \frac{FreqError \times 2^{24}}{F_{xtal}} \times \frac{BW[kHz]}{500}$ |
| RegFeiMid (0x29) | 7-0 | FreqError(15:8) | r | 0x0 | Middle byte of RF Frequency Error |
| RegFeiLsb (0x2A) | 7-0 | FreqError(7:0) | r | 0x0 | LSB of RF Frequency Error |
| (0x2B) | - | Reserved | r | n/a | Reserved |
| RegRssiWideband (0x2C) | 7-0 | RssiWideband(7:0) | r | n/a | Wideband RSSI measurement used to locally generate a random number |
| (0x2D) - (0x30) | - | Reserved | r | n/a | Reserved |
| RegDetectOptimiz e (0x31) | 7-3 | Reserved | r | 0xC0 | Reserved |
| | 2-0 | DetectionOptimize | rw | 0x03 | LoRa Detection Optimize 0x03 → SF7 to SF12 0x05 → SF6 |
| (0x32) | - | Reserved | r | n/a | Reserved |
| RegInvertIQ (0x33) | 7 | Reserved | rw | 0x0 | Reserved |
| | 6 | InvertIQ | rw | 0x0 | Invert the LoRa I and Q signals 0 → normal mode 1 → I and Q signals are inverted |
| | 5-0 | Reserved | rw | 0x27 | Reserved |
| (0x34) - (0x36) | 7-0 | Reserved | r | n/a | Reserved |
| RegDetectionThre shold (0x37) | 7-0 | DetectionThreshold | rw | 0x0A | LoRa detection threshold 0x0A → SF7 to SF12 0x0C → SF6 |
| (0x38) | - | Reserved | r | n/a | Reserved |
| RegSyncWord (0x39) | 7-0 | SyncWord | rw | 0x12 | LoRa Sync Word Value 0x34 is reserved for LoRaWAN networks |

| Name (Address) | Bits | Variable Name | Mode | Reset | LoRa™ Description |
|-------------------|------|---------------|------|-------|-------------------|
| (0x3A) - (0x3F) | - | Reserved | r | n/a | Reserved |

7. Application Information

7.1. Crystal Resonator Specification

Table 45 shows the crystal resonator specification for the crystal reference oscillator circuit of the SX1276/77/78/79. This specification covers the full range of operation of the SX1276/77/78/79 and is employed in the reference design.

Table 45 Crystal Specification

| Symbol | Description | Conditions | Min | Typ | Max | Unit |
|--------|---------------------------|-------------------------|-----|-----|-----|------|
| FXOSC | XTAL Frequency | | - | 32 | - | MHz |
| RS | XTAL Serial Resistance | | - | 15 | 100 | ohms |
| C0 | XTAL Shunt Capacitance | | - | 1 | 3 | pF |
| CFOOT | External Foot Capacitance | On each pin XTA and XTB | 10 | 15 | 22 | pF |
| CLOAD | Crystal Load Capacitance | | 6 | - | 12 | pF |

Notes - the initial frequency tolerance, temperature stability and aging performance should be chosen in accordance with the target operating temperature range and the receiver bandwidth selected.

- the loading capacitance should be applied externally, and adapted to the actual Cload specification of the XTAL.

7.2. Reset of the Chip

A power-on reset of the SX1276/77/78/79 is triggered at power up. Additionally, a manual reset can be issued by controlling pin 7.

7.2.1. POR

If the application requires the disconnection of VDD from the SX1276/77/78/79, despite of the extremely low Sleep Mode current, the user should wait for 10 ms from the end of the POR cycle before commencing communications over the SPI bus. Pin 7 (NRESET) should be left floating during the POR sequence.

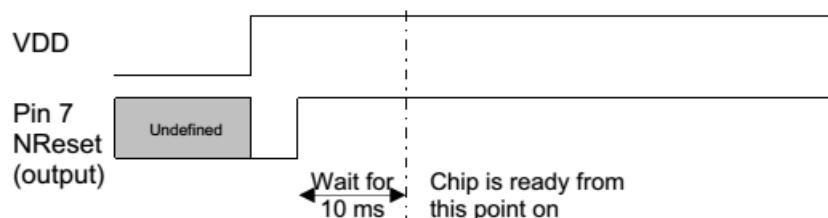


Figure 42. POR Timing Diagram

Please note that any CLKOUT activity can also be used to detect that the chip is ready.

7.2.2. Manual Reset

A manual reset of the SX1276/77/78/79 is possible even for applications in which VDD cannot be physically disconnected. Pin 7 should be pulled low for a hundred microseconds, and then released. The user should then wait for 5 ms before using the chip.

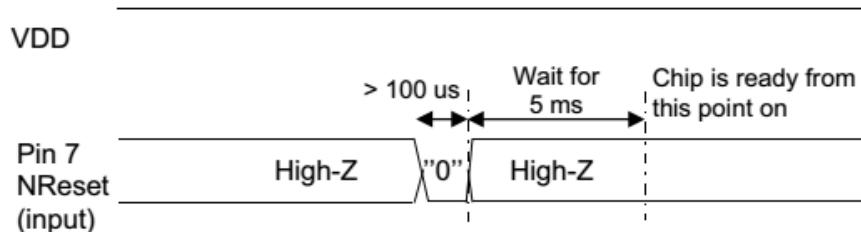


Figure 43. Manual Reset Timing Diagram

Note whilst pin 7 is driven low, an over current consumption of up to one milliampere can be seen on VDD.

7.3. Top Sequencer: Listen Mode Examples

In this scenario, the circuit spends most of the time in Idle mode, during which only the RC oscillator is on. Periodically the receiver wakes up and looks for incoming signal. If a wanted signal is detected, the receiver is kept on and data are analyzed. Otherwise, if there was no wanted signal for a defined period of time, the receiver is switched off until the next receive period.

During Listen mode, the Radio stays most of the time in a Low Power mode, resulting in very low average power consumption. The general timing diagram of this scenario is given in Figure 44.

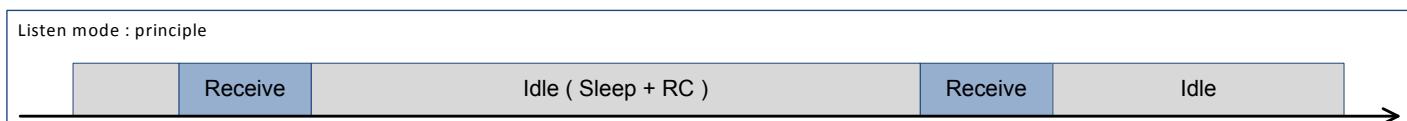


Figure 44. Listen Mode: Principle

An interrupt request is generated on a packet reception. The user can then take appropriate action.

Depending on the application and environment, there are several ways to implement Listen mode:

- ◆ Wake on a *PreambleDetect* interrupt
- ◆ Wake on a *SyncAddress* interrupt
- ◆ Wake on a *PayloadReady* interrupt

7.3.1. Wake on Preamble Interrupt

In one possible scenario, the sequencer polls for a Preamble detection. If a preamble signal is detected, the sequencer is switched off and the circuit stays in Receive mode until the user switches modes. Otherwise, the receiver is switched off until the next Rx period.

7.3.1.1. Timing Diagram

When no signal is received, the circuit wakes every Timer1 + Timer2 and switches to Receive mode for a time defined by Timer2, as shown on the following diagram. If no Preamble is detected, it then switches back to Idle mode, i.e. Sleep mode with RC oscillator on.

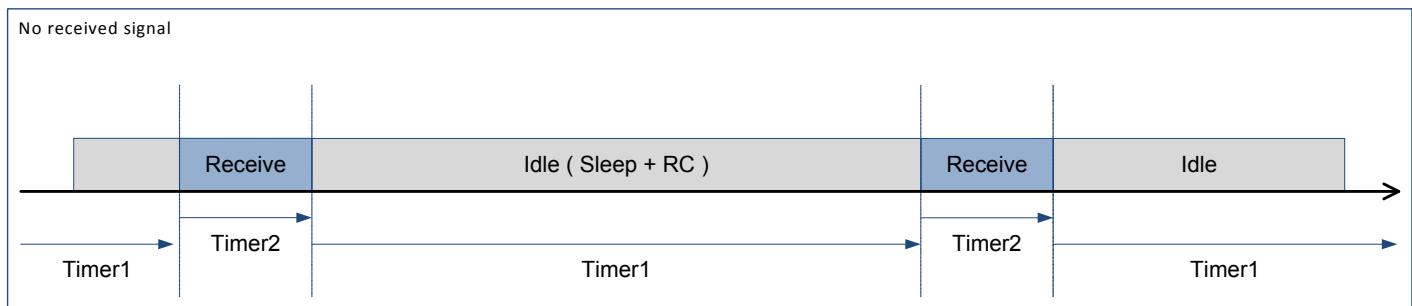


Figure 45. Listen Mode with No Preamble Received

If a Preamble signal is detected, the Sequencer is switched off. The *PreambleDetect* signal can be mapped to DIO4, in order to request the user's attention. The user can then take appropriate action.

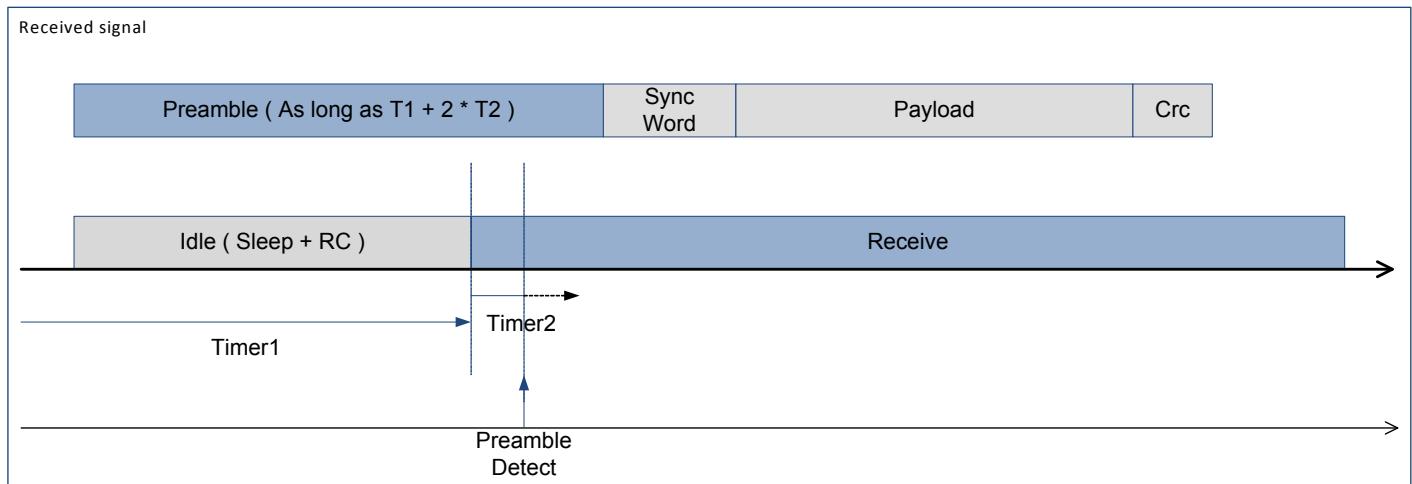
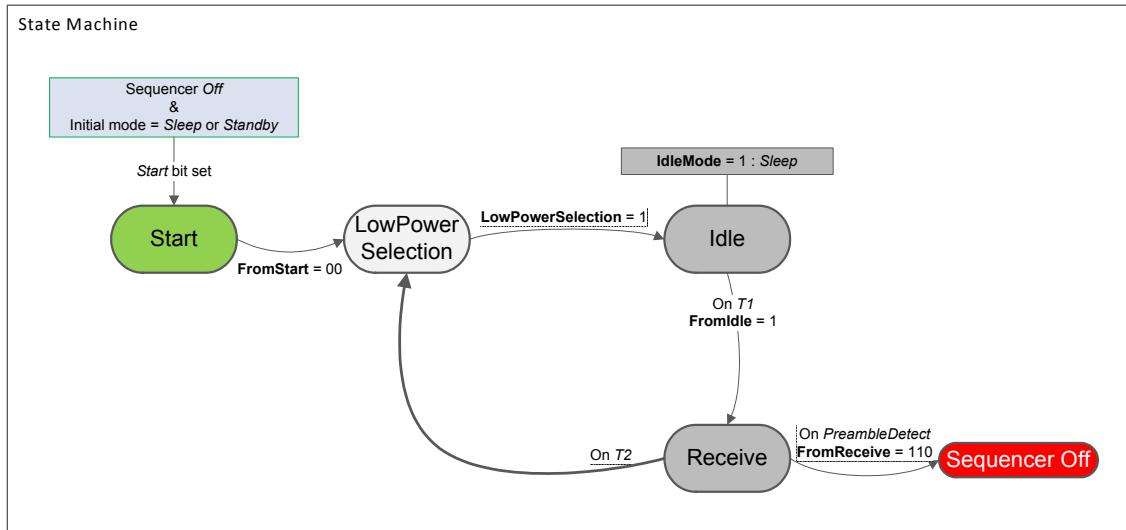


Figure 46. Listen Mode with Preamble Received

7.3.1.2. Sequencer Configuration

The following graph shows Listen mode - Wake on *PreambleDetect* state machine:



*Figure 47. Wake On *PreambleDetect* State Machine*

This example configuration is achieved as follows:

*Table 46 Listen Mode with *PreambleDetect* Condition Settings*

| Variable | Effect |
|-------------------|---|
| IdleMode | 1: Sleep mode |
| FromStart | 00: To LowPowerSelection |
| LowPowerSelection | 1: To Idle state |
| FromIdle | 1: To Receive state on <i>T1</i> interrupt |
| FromReceive | 110: To Sequencer Off on <i>PreambleDetect</i> interrupt |

T_{Timer2} defines the maximum duration the chip stays in Receive mode as long as no Preamble is detected. In order to optimize power consumption, Timer2 must be set just long enough for Preamble detection.

$T_{Timer1} + T_{Timer2}$ defines the cycling period, i.e. time between two Preamble polling starts. In order to optimize average power consumption, Timer1 should be relatively long. However, increasing Timer1 also extends packet reception duration.

In order to insure packet detection and optimize the receiver's power consumption, the received packet Preamble should be as long as $T_{Timer1} + 2 \times T_{Timer2}$.

An example of DIO configuration for this mode is described in the following table:

*Table 47 Listen Mode with *PreambleDetect* Condition Recommended DIO Mapping*

| DIO | Value | Description |
|-----|-------|---|
| 0 | 01 | CrcOk |
| 1 | 00 | FifoLevel |
| 3 | 00 | FifoEmpty |
| 4 | 11 | PreambleDetect – Note: MapPreambleDetect bit should be set. |

7.3.2. Wake on SyncAddress Interrupt

In another possible scenario, the sequencer polls for a Preamble detection and then for a valid SyncAddress interrupt. If events occur, the sequencer is switched off and the circuit stays in Receive mode until the user switches modes. Otherwise, the receiver is switched off until the next Rx period.

7.3.2.1. Timing Diagram

Most of the sequencer running time is spent while no wanted signal is received. As shown by the timing diagram in Figure 48, the circuit wakes periodically for a short time, defined by RxTimeout. The circuit is in a Low Power mode for the rest of Timer1 + Timer2 (i.e. Timer1 + Timer2 - TrxTimeout)

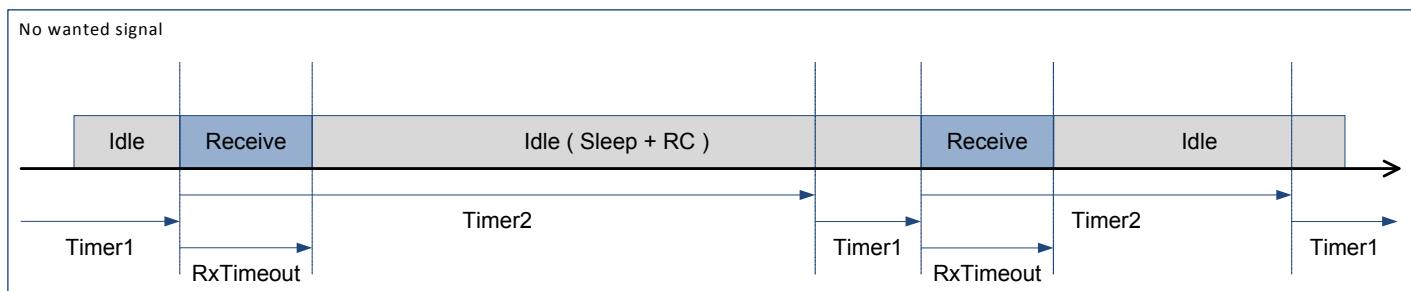


Figure 48. Listen Mode with no SyncAddress Detected

If a preamble is detected before RxTimeout timer ends, the circuit stays in Receive mode and waits for a valid SyncAddress detection. If none is detected by the end of Timer2, Receive mode is deactivated and the polling cycle resumes, without any user intervention.

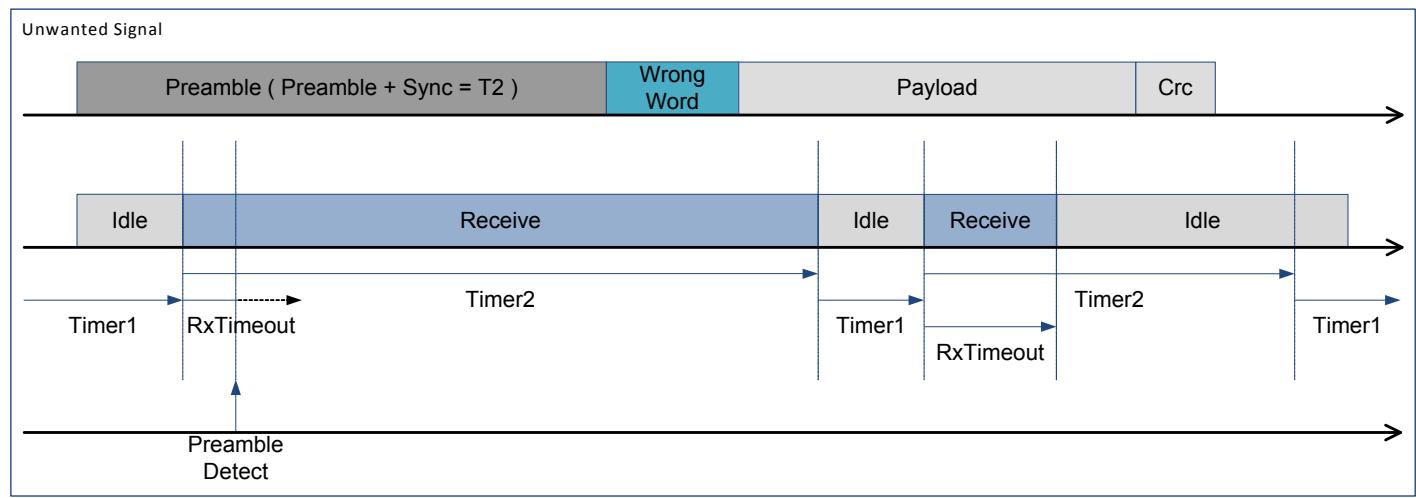


Figure 49. Listen Mode with Preamble Received and no SyncAddress

But if a valid Sync Word is detected, a SyncAddress interrupt is fired, the Sequencer is switched off and the circuit stays in Receive mode as long as the user doesn't switch modes.

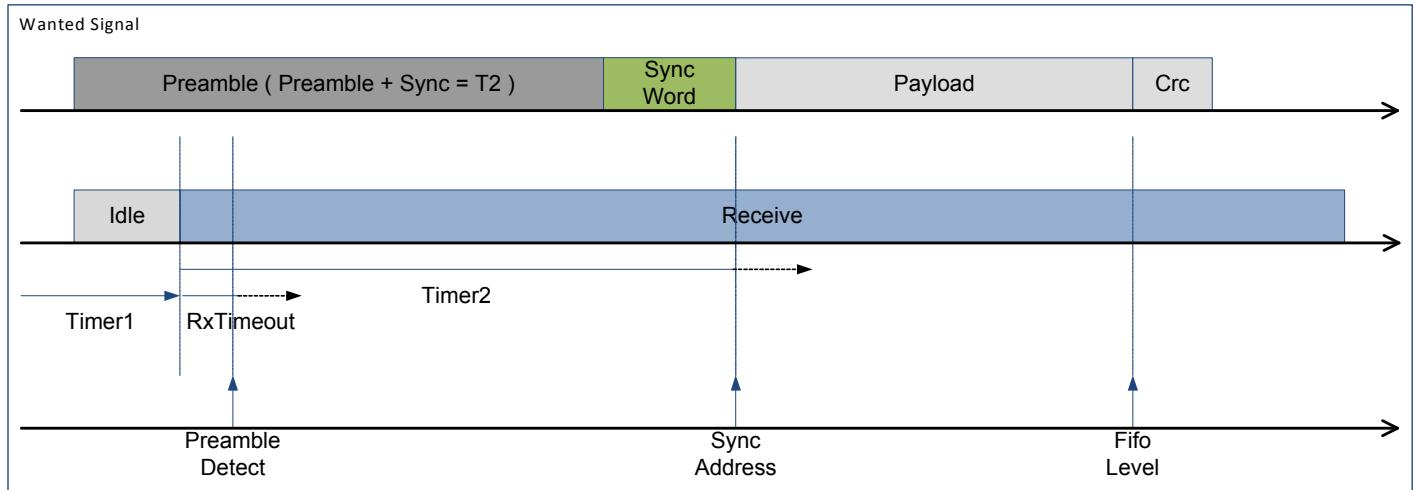


Figure 50. Listen Mode with Preamble Received & Valid SyncAddress

7.3.2.2. Sequencer Configuration

The following graph shows Listen mode - Wake on SyncAddress state machine:

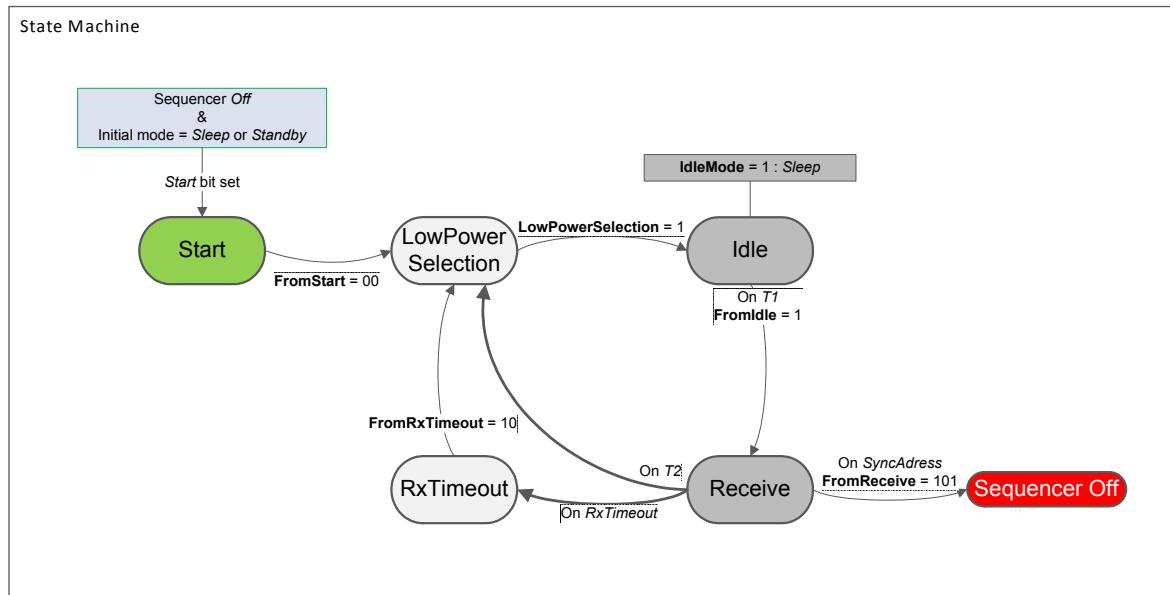


Figure 51. Wake On SyncAddress State Machine

This example configuration is achieved as follows:

Table 48 Listen Mode with SyncAddress Condition Settings

| Variable | Effect |
|-------------------|--|
| IdleMode | 1: Sleep mode |
| FromStart | 00: To LowPowerSelection |
| LowPowerSelection | 1: To Idle state |
| FromIdle | 1: To Receive state on T_1 interrupt |
| FromReceive | 101: To Sequencer off on SyncAddress interrupt |
| FromRxTimeout | 10: To LowPowerSelection |

$T_{TimeoutRxPreamble}$ should be set to just long enough to catch a preamble (depends on *PreambleDetectSize* and *BitRate*).

T_{Timer1} should be set to 64 μ s (shortest possible duration).

T_{Timer2} is set so that $T_{Timer1} + T_{Timer2}$ defines the time between two start of reception.

In order to insure packet detection and optimize the receiver power consumption, the received packet Preamble should be defined so that $T_{Preamble} = T_{Timer2} - T_{SyncAddress}$ with $T_{SyncAddress} = (SyncSize + 1)*8/BitRate$.

An example of DIO configuration for this mode is described in the following table:

Table 49 Listen Mode with PreambleDetect Condition Recommended DIO Mapping

| DIO | Value | Description |
|-----|-------|--|
| 0 | 01 | CrcOk |
| 1 | 00 | FifoLevel |
| 2 | 11 | SyncAddress |
| 3 | 00 | FifoEmpty |
| 4 | 11 | PreambleDetect – Note: <i>MapPreambleDetect</i> bit should be set. |

7.4. Top Sequencer: Beacon Mode

In this mode, a repetitive message is transmitted periodically. If the Payload being sent is always identical, and *PayloadLength* is smaller than the FIFO size, the use of the *BeaconOn* bit in *RegPacketConfig2* together with the Sequencer permit to achieve periodic beacon without any user intervention.

7.4.1. Timing diagram

In this mode, the Radio is switched to Transmit mode every $T_{Timer1} + T_{Timer2}$ and back to Idle mode after *PacketSent*, as shown in the diagram below. The Sequencer insures minimal time is spent in Transmit mode, and therefore power consumption is optimized.

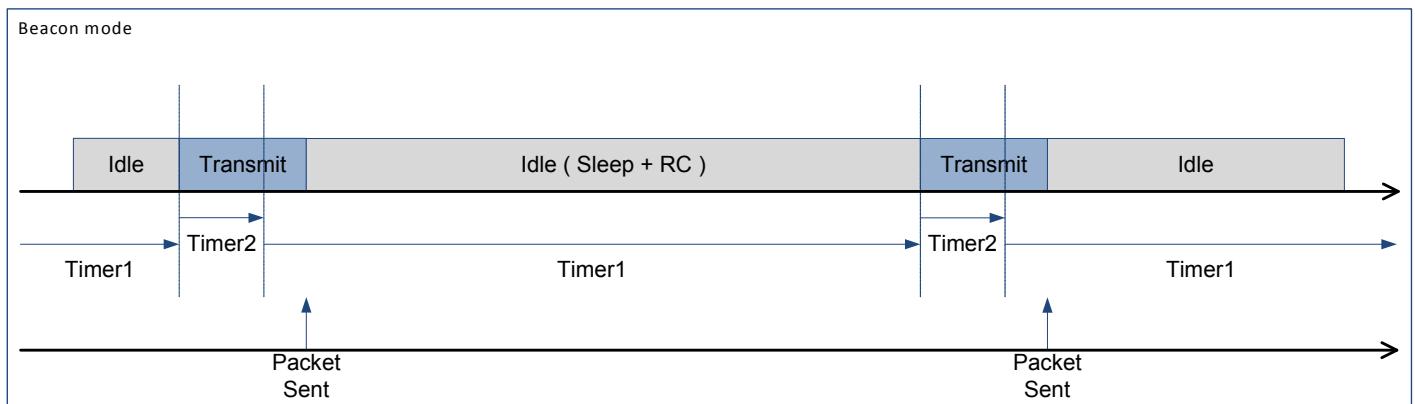


Figure 52. Beacon Mode Timing Diagram

7.4.2. Sequencer Configuration

The Beacon mode state machine is presented in the following graph. It is noticeable that the sequencer enters an infinite loop and can only be stopped by setting *SequencerStop* bit in *RegSeqConfig1*.

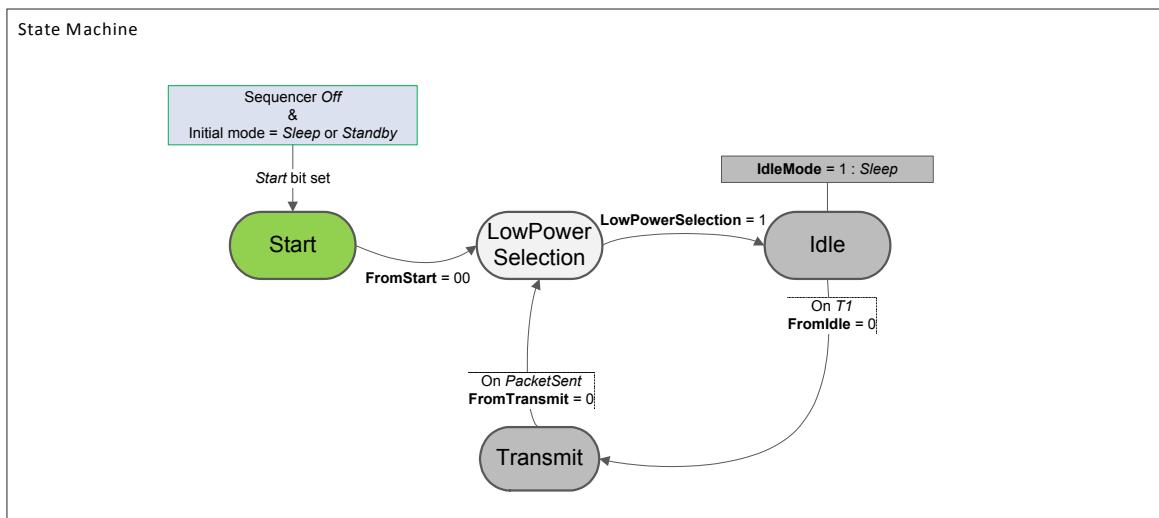


Figure 53. Beacon Mode State Machine

This example is achieved by programming the Sequencer as follows:

Table 50 Beacon Mode Settings

| Variable | Effect |
|-------------------|---|
| IdleMode | 1: Sleep mode |
| FromStart | 00: To LowPowerSelection |
| LowPowerSelection | 1: To Idle state |
| FromIdle | 0: To Transmit state on <i>T1</i> interrupt |
| FromTransmit | 0: To LowPowerSelection on <i>PacketSent</i> interrupt |

$T_{Timer1} + T_{Timer2}$ define the time between the start of two transmissions.

7.5. Example CRC Calculation

The following routine(s) may be implemented to mimic the CRC calculation of the SX1276/77/78/79:

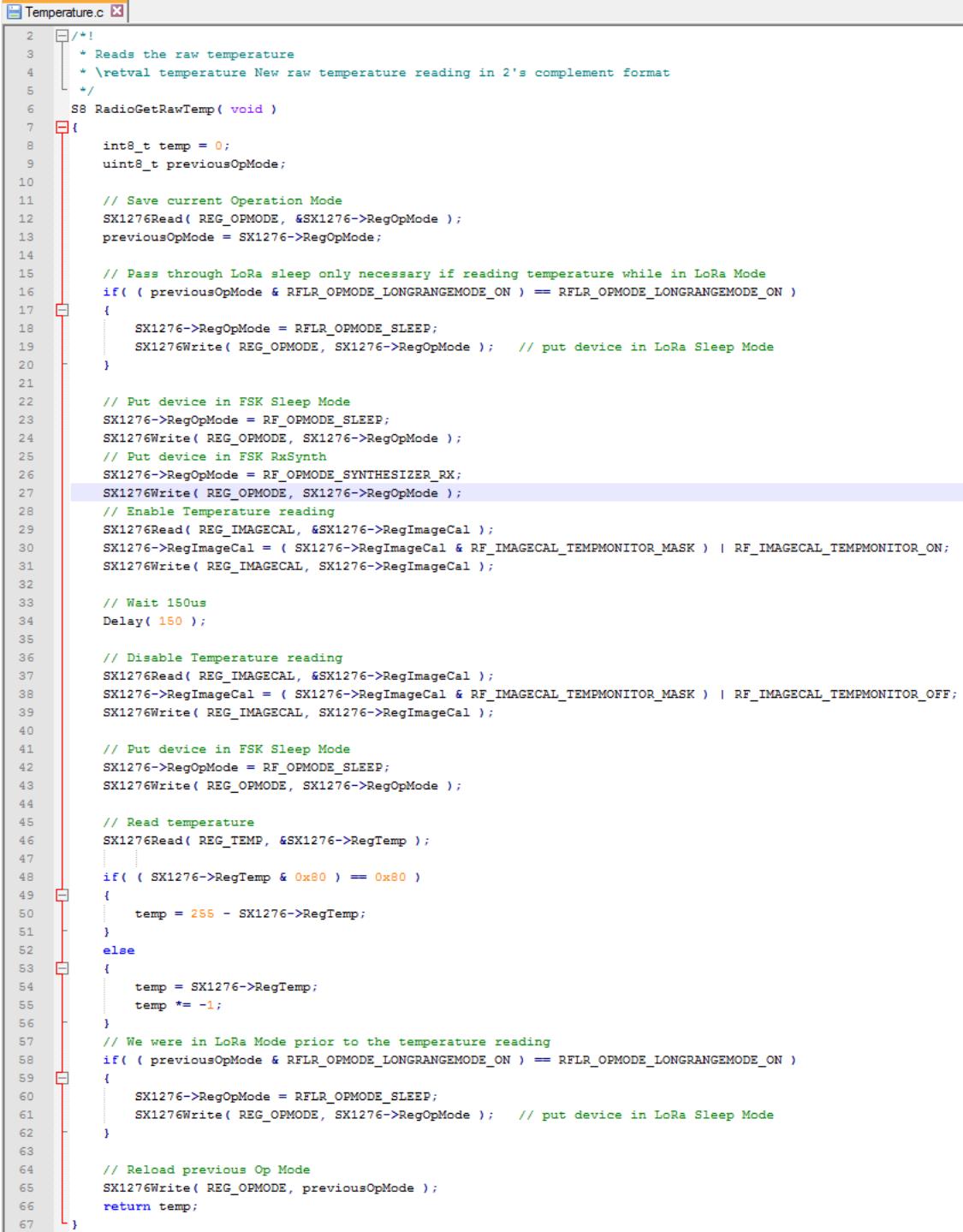
```

1 // CRC types
2 #define CRC_TYPE_CCITT ..... 0
3 #define CRC_TYPE_IEM ..... 1
4
5 // Polynomial = X^16 + X^12 + X^5 + 1
6 #define POLYNOMIAL_CCITT ..... 0x1021
7 // Polynomial = X^16 + X^15 + X^2 + 1
8 #define POLYNOMIAL_IEM ..... 0x8005
9
10 // Seeds
11 #define CRC_IEM_SEED ..... 0xFFFF
12 #define CRC_CCITT_SEED ..... 0x1D0F
13
14 */
15 * CRC algorithm implementation
16 *
17 * \param[in] crc Previous CRC value
18 * \param[in] data New data to be added to the CRC
19 * \param[in] polynomial CRC polynomial selection [CRC_TYPE_CCITT, CRC_TYPE_IEM]
20 *
21 * \retval crc New computed CRC
22 */
23 U16 ComputeCrc( U16 crc, U8 data, U16 polynomial )
24 {
25     U8 i;
26     for( i = 0; i < 8; i++ )
27     {
28         if( ( ( crc & 0x0000 ) >> 8 ) ^ ( data & 0x80 ) != 0 )
29         {
30             if( crc <= 1 ..... // shift left once
31             crc ^= polynomial ..... // XOR with polynomial
32             }
33             else
34             {
35                 if( crc <= 1 ..... // shift left once
36                 }
37                 data <<= 1 ..... // Next data bit
38             }
39         return crc;
40     }
41
42 */
43 * CRC algorithm implementation
44 *
45 * \param[in] buffer Array containing the data
46 * \param[in] bufferLength Buffer length
47 * \param[in] crcType Selects the CRC polynomial[CRC_TYPE_CCITT, CRC_TYPE_IEM]
48 *
49 * \retval crc Buffer computed CRC
50 */
51 U16 RadioPacketComputeCrc( U8 *buffer, U8 bufferLength, U8 crcType )
52 {
53     U8 i;
54     U16 crc;
55     U16 polynomial;
56
57     polynomial = ( crcType == CRC_TYPE_IEM ) ? POLYNOMIAL_IEM : POLYNOMIAL_CCITT;
58     crc = ( crcType == CRC_TYPE_IEM ) ? CRC_IEM_SEED : CRC_CCITT_SEED;
59
60     for( i = 0; i < bufferLength; i++ )
61     {
62         crc = ComputeCrc( crc, buffer[i], polynomial );
63     }
64
65     if( crcType == CRC_TYPE_IEM )
66     {
67         return crc;
68     }
69     else
70     {
71         return ( U16 )( ~crc );
72     }
73 }
```

Figure 54. Example CRC Code

7.6. Example Temperature Reading

The following routine(s) may be implemented to read the temperature and calibrate the sensor:



```

2  /*!
3   * Reads the raw temperature
4   * \return temperature New raw temperature reading in 2's complement format
5   */
6   S8 RadioGetRawTemp( void )
7  {
8      int8_t temp = 0;
9      uint8_t previousOpMode;
10
11     // Save current Operation Mode
12     SX1276Read( REG_OPMode, &SX1276->RegOpMode );
13     previousOpMode = SX1276->RegOpMode;
14
15     // Pass through LoRa sleep only necessary if reading temperature while in LoRa Mode
16     if( ( previousOpMode & RFLR_OPMode_LONGRANGemode_ON ) == RFLR_OPMode_LONGRANGemode_ON )
17     {
18         SX1276->ReqOpMode = RFLR_OPMode_SLEEP;
19         SX1276Write( REG_OPMode, SX1276->RegOpMode );    // put device in LoRa Sleep Mode
20     }
21
22     // Put device in FSK Sleep Mode
23     SX1276->ReqOpMode = RF_OPMode_SLEEP;
24     SX1276Write( REG_OPMode, SX1276->ReqOpMode );
25     // Put device in FSK RxSynth
26     SX1276->ReqOpMode = RF_OPMode_SYNTHESIZER_RX;
27     SX1276Write( REG_OPMode, SX1276->ReqOpMode );
28     // Enable Temperature reading
29     SX1276Read( REG_IMAGECAL, &SX1276->RegImageCal );
30     SX1276->RegImageCal = ( SX1276->RegImageCal & RF_IMAGECAL_TEMPMONITOR_MASK ) | RF_IMAGECAL_TEMPMONITOR_ON;
31     SX1276Write( REG_IMAGECAL, SX1276->RegImageCal );
32
33     // Wait 150us
34     Delay( 150 );
35
36     // Disable Temperature reading
37     SX1276Read( REG_IMAGECAL, &SX1276->RegImageCal );
38     SX1276->RegImageCal = ( SX1276->RegImageCal & RF_IMAGECAL_TEMPMONITOR_MASK ) | RF_IMAGECAL_TEMPMONITOR_OFF;
39     SX1276Write( REG_IMAGECAL, SX1276->RegImageCal );
40
41     // Put device in FSK Sleep Mode
42     SX1276->ReqOpMode = RF_OPMode_SLEEP;
43     SX1276Write( REG_OPMode, SX1276->ReqOpMode );
44
45     // Read temperature
46     SX1276Read( REG_TEMP, &SX1276->RegTemp );
47
48     if( ( SX1276->RegTemp & 0x80 ) == 0x80 )
49     {
50         temp = 255 - SX1276->RegTemp;
51     }
52     else
53     {
54         temp = SX1276->RegTemp;
55         temp *= -1;
56     }
57     // We were in LoRa Mode prior to the temperature reading
58     if( ( previousOpMode & RFLR_OPMode_LONGRANGemode_ON ) == RFLR_OPMode_LONGRANGemode_ON )
59     {
60         SX1276->ReqOpMode = RFLR_OPMode_SLEEP;
61         SX1276Write( REG_OPMode, SX1276->ReqOpMode );    // put device in LoRa Sleep Mode
62     }
63
64     // Reload previous Op Mode
65     SX1276Write( REG_OPMode, previousOpMode );
66
67 }

```

Figure 55. Example Temperature Reading

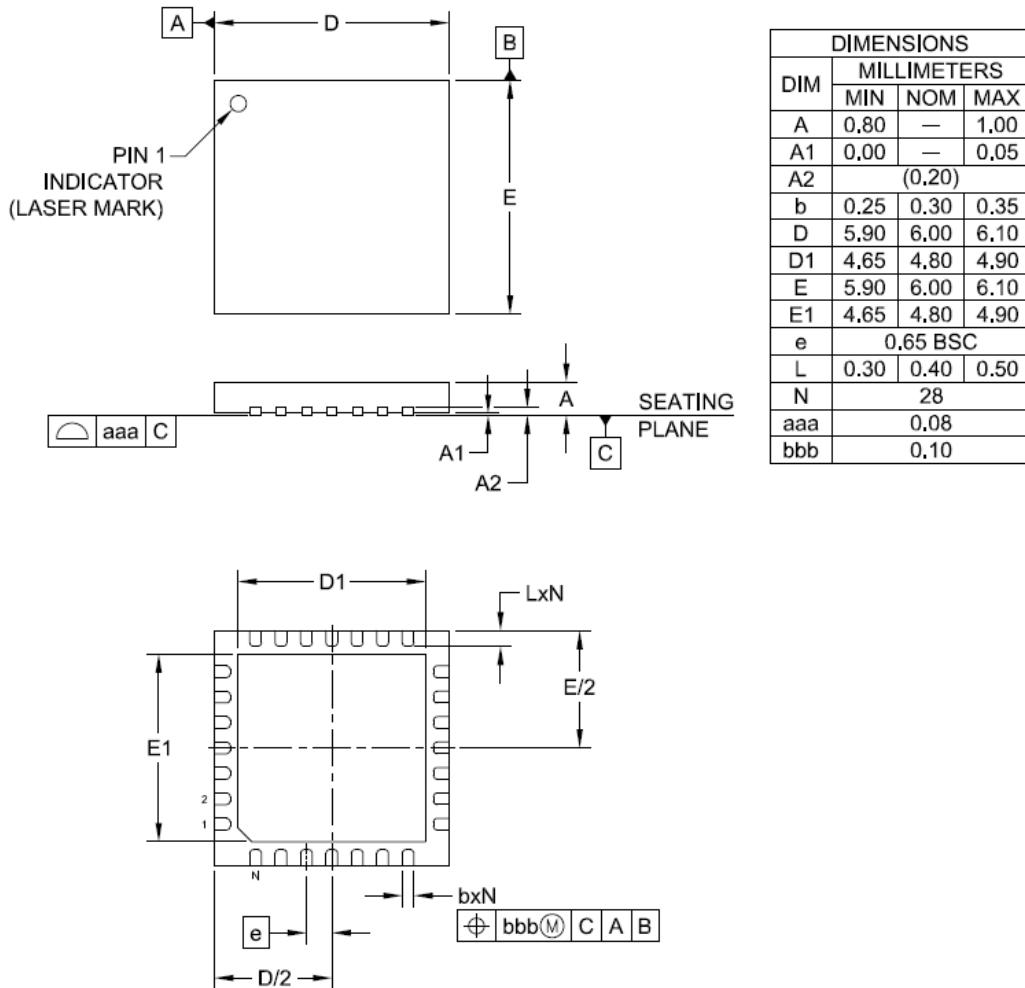
```
68
69  /*!
70   * Computes the temperature compensation factor
71   * \param [IN] actualTemp Actual temperature measured by an external device
72   * \retval compensationFactor Computed compensation factor
73   */
74   S8 RadioCalibrateTemp( S8 actualTemp )
75  {
76      return actualTemp - RadioGetRawTemp( );
77  }
78
79  /*!
80   * Gets the actual compensated temperature
81   * \param [IN] compensationFactor Return value of the calibration function
82   * \retval New compensated temperature value
83   */
84   S8 RadioGetTemp( S8 compensationFactor )
85  {
86      return RadioGetRawTemp( ) + compensationFactor;
87  }
88
89  /*!
90   * Usage example
91   */
92   void main( void )
93  {
94     S8 temp;
95     S8 actualTemp = 0;
96     S8 compensationFactor = 0;
97
98     // Ask user for the temperature during calibration
99     actualTemp = AskUserTemperature( );
100    compensationFactor = RadioCalibrateTemp( actualTemp );
101
102    while( True )
103    {
104        temp = RadioGetTemp( compensationFactor );
105    }
106 }
```

Figure 56. Example Temperature Reading (continued)

8. Packaging Information

8.1. Package Outline Drawing

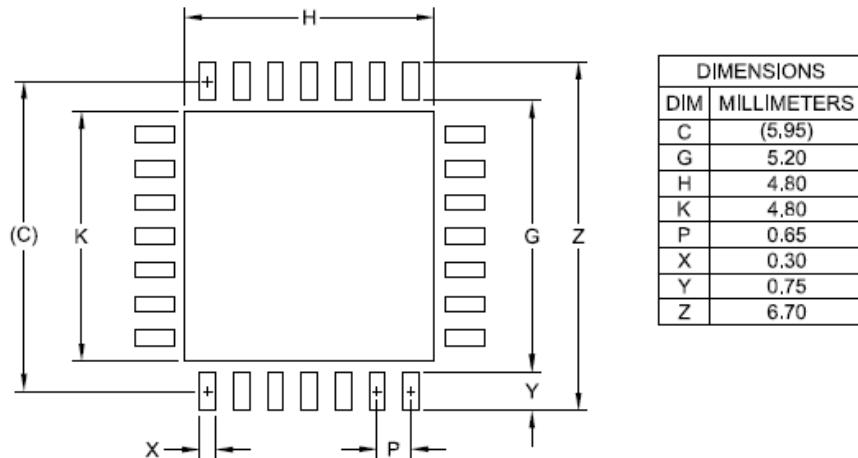
The SX1276/77/78/79 is available in a 28-lead QFN package as shown in Figure 57.


NOTES:

1. CONTROLLING DIMENSIONS ARE IN MILLIMETERS (ANGLES IN DEGREES).
2. COPLANARITY APPLIES TO THE EXPOSED PAD AS WELL AS THE TERMINALS.

Figure 57. Package Outline Drawing

8.2. Recommended Land Pattern


NOTES:

1. CONTROLLING DIMENSIONS ARE IN MILLIMETERS (ANGLES IN DEGREES).
2. THIS LAND PATTERN IS FOR REFERENCE PURPOSE ONLY.
CONSULT YOUR MANUFACTURING GROUP TO ENSURE YOUR COMPANY'S
MANUFACTURING GUIDELINES ARE MET.
3. THERMAL VIAS IN THE LAND PATTERN OF THE EXPOSED PAD SHALL BE CONNECTED
TO A SYSTEM GROUND PLANE. FAILURE TO DO SO MAY COMPROMISE THE THERMAL
AND/OR FUNCTIONAL PERFORMANCE OF THE DEVICE.
4. SQUARE PACKAGE - DIMENSIONS APPLY IN BOTH "X" AND "Y" DIRECTIONS.

Figure 58. Recommended Land Pattern

8.3. Tape & Reel Information

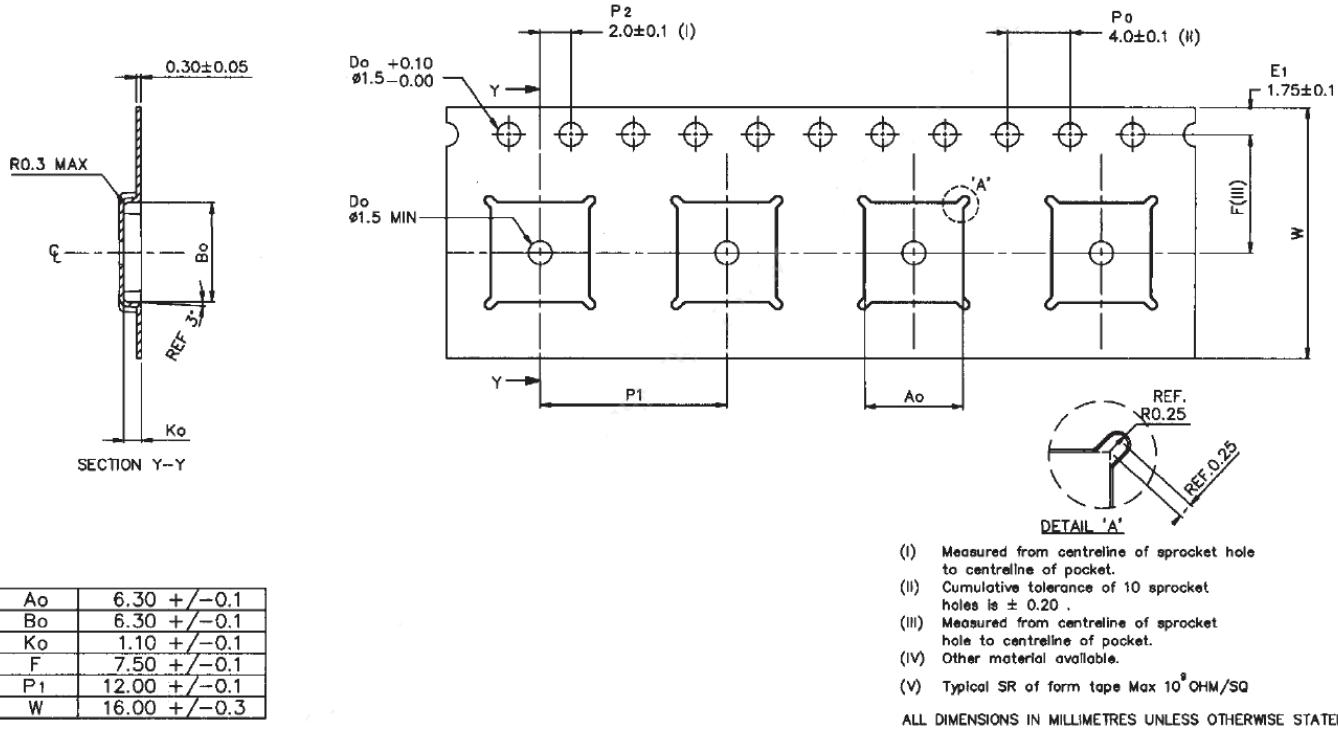


Figure 59. Tape and Reel Information

9. Revision History

Table 51 Revision History

| Revision | Date | Comment |
|----------|------------|--|
| 1 | Sept 2013 | First FINAL release |
| 2 | Nov 2014 | Miscellaneous typographical corrections Correction of RxPayloadCrcOn description Improve description in the RSSI and IQ calibration mechanism Correction of ToA formulae Inclusion of FEI and automatic frequency correction for LoRa Corrected Rssi Formula in Lora mode |
| 3 | Nov 2014 | Addition of part SX1279 |
| 4 | March 2015 | Clarified operation modes for Rx Single and Rx Continuous mode in LoRa Added use cases for Rx Single and Rx Continuous mode in LoRa mode Clarified used of LoRa RxPayloadCrcOn in Register Table Added description of register RegSyncWord in LoRa register table Changed Stand-By typo into Standby |

© Semtech 2015

All rights reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent or other industrial or intellectual property rights. Semtech assumes no responsibility or liability whatsoever for any failure or unexpected operation resulting from misuse, neglect, improper installation, repair or improper handling or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified range.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

Contact information

Semtech Corporation
Wireless, Sensing & Timing Products Division
200 Flynn Road, Camarillo, CA 93012
Phone: (805) 498-2111 Fax: (805) 498-3804
E-mail: sales@semtech.com
support_rf@semtech.com
Internet: <http://www.semtech.com>

TPS61023 3.7-A Boost Converter with 0.5-V Ultra-low Input Voltage

1 Features

- Input voltage range: 0.5 V to 5.5 V
- 1.8-V Minimum input voltage for start-up
- Output voltage setting range: 2.2 V to 5.5 V
- Two 47-mΩ (LS) / 68-mΩ (HS) MOSFETs
- 3.7-A Valley switching current limit
- 94% Efficiency at $V_{IN} = 3.6$ V, $V_{OUT} = 5$ V and $I_{OUT} = 1.5$ A
- 1-MHz Switching frequency when $V_{IN} > 1.5$ V and 0.5-MHz switching frequency when $V_{IN} < 1$ V
- Typical 0.1-µA shutdown current from V_{IN} and SW
- ±2.5% Reference voltage accuracy over –40°C to +125°C
- Auto PFM operation mode at light load
- Pass-through mode when $V_{IN} > V_{OUT}$
- True disconnection between input and output during shutdown
- Output overvoltage and thermal shutdown protections
- Output short-circuit protection
- 1.2-mm × 1.6-mm SOT563 (DRL) 6-pin package

2 Applications

- Electronic shelf label
- Video doorbell
- Remote controller

3 Description

TPS61023 device is a synchronous boost converter with 0.5-V ultra-low input voltage. The device provides a power supply solution for portable equipment and smart devices powered by various batteries and super capacitors. The TPS61023 has typical 3.7-A valley switch current limit over full temperature range. With a wide input voltage range of 0.5 V to 5.5 V, the TPS61023 supports super capacitor backup power applications, which may deeply discharge the super capacitor.

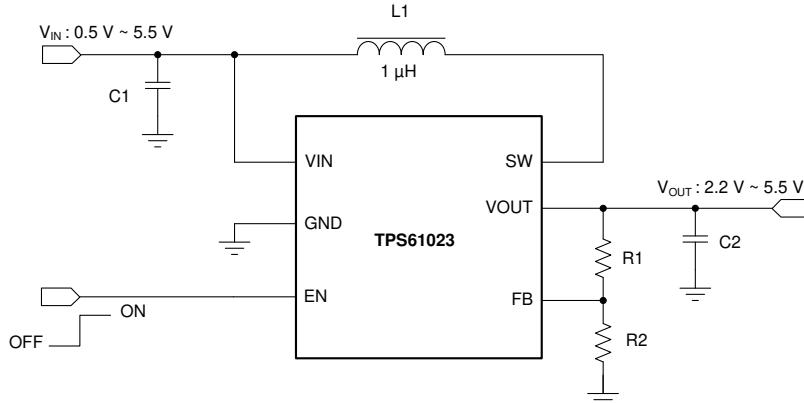
The TPS61023 operates at 1-MHz switching frequency when the input voltage is above 1.5 V. The switching frequency decreases gradually to 0.5 MHz when the input voltage is below 1.5 V down to 1 V. The TPS61023 enters power-save mode at light load condition to maintain high efficiency over the entire load current range. The TPS61023 consumes a 20-µA quiescent current from V_{OUT} in light load condition. During shutdown, the TPS61023 is completely disconnected from the input power and only consumes a 0.1-µA current to achieve long battery life. The TPS61023 has 5.7-V output overvoltage protection, output short circuit protection, and thermal shutdown protection.

The TPS61023 offers a very small solution size with 1.2-mm × 1.6-mm SOT563 (DRL) package and minimum amount of external components.

Device Information

| PART NUMBER | PACKAGE ⁽¹⁾ | BODY SIZE (NOM) |
|-------------|------------------------|-------------------|
| TPS61023 | SOT563 (6) | 1.20 mm × 1.60 mm |

(1) For all available packages, see the orderable addendum at the end of the data sheet.



Typical Application Circuit



An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

Table of Contents

| | | | |
|---|-----------|---|-----------|
| 1 Features..... | 1 | 8.1 Application Information..... | 12 |
| 2 Applications..... | 1 | 8.2 Typical Application..... | 12 |
| 3 Description..... | 1 | 9 Power Supply Recommendations..... | 17 |
| 4 Revision History..... | 2 | 10 Layout..... | 18 |
| 5 Pin Configuration and Functions..... | 3 | 10.1 Layout Guidelines..... | 18 |
| Pin Functions..... | 3 | 10.2 Layout Example..... | 18 |
| 6 Specifications..... | 4 | 10.3 Thermal Considerations..... | 19 |
| 6.1 Absolute Maximum Ratings..... | 4 | 11 Device and Documentation Support..... | 20 |
| 6.2 ESD Ratings..... | 4 | 11.1 Device Support..... | 20 |
| 6.3 Recommended Operating Conditions..... | 4 | 11.2 Receiving Notification of Documentation Updates..... | 20 |
| 6.4 Thermal Information..... | 4 | 11.3 Support Resources..... | 20 |
| 6.5 Electrical Characteristics..... | 5 | 11.4 Trademarks..... | 20 |
| 6.6 Typical Characteristics..... | 6 | 11.5 Electrostatic Discharge Caution..... | 20 |
| 7 Detailed Description..... | 8 | 11.6 Glossary..... | 20 |
| 7.1 Overview..... | 8 | 12 Mechanical, Packaging, and Orderable | |
| 7.2 Functional Block Diagram..... | 8 | Information..... | 20 |
| 7.3 Feature Description..... | 9 | 12.1 Package Option Addendum..... | 21 |
| 7.4 Device Functional Modes..... | 10 | 12.2 Tape and Reel Information..... | 23 |
| 8 Application and Implementation..... | 12 | | |

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

| Changes from Revision A (October 2019) to Revision B (August 2020) | Page |
|--|-------------|
| • Updated the numbering format for tables, figures and cross-references throughout the document..... | 1 |
| • Changed unit in Figure 6-6 to μA | 6 |
| • Changed 80 mA to 800 mA in Figure 8-7 | 16 |

| Changes from Revision * (September 2019) to Revision A (October 2019) | Page |
|--|-------------|
| • Changed Product Status to Production Data for Production release | 1 |

5 Pin Configuration and Functions

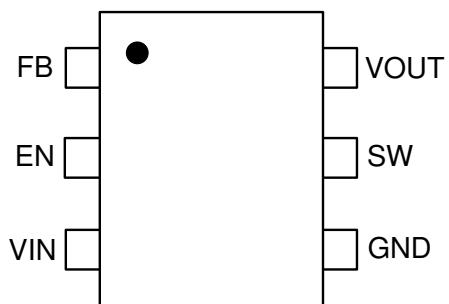


Figure 5-1. DRL Package 6-Pin SOT563 Top View

Pin Functions

| PIN | | I/O | DESCRIPTION |
|-----|------|-----|--|
| NO. | NAME | | |
| 1 | FB | I | Voltage feedback of adjustable output voltage |
| 2 | EN | I | Enable logic input. Logic high voltage enables the device. Logic low voltage disables the device and turns it into shutdown mode. |
| 3 | VIN | I | IC power supply input |
| 4 | GND | PWR | Ground pin of the IC |
| 5 | SW | PWR | The switch pin of the converter. It is connected to the drain of the internal low-side power MOSFET and the source of the internal high-side power MOSFET. |
| 6 | VOUT | PWR | Boost converter output |

6 Specifications

6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)⁽¹⁾

| | | MIN | MAX | UNIT | |
|--|-----------------------|------|-----|------|----|
| Voltage range at terminals ⁽²⁾ | VIN, EN, FB, SW, VOUT | -0.3 | 7 | V | |
| | SW spike at 10ns | -0.7 | 8 | V | |
| | SW spike at 1ns | -0.7 | 9 | V | |
| Operating junction temperature, T _J | | | -40 | 150 | °C |
| Storage temperature, T _{stg} | | | -65 | 150 | °C |

- (1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
- (2) All voltage values are with respect to network ground terminal.

6.2 ESD Ratings

| | | VALUE | UNIT |
|--------------------|-------------------------|--|-------|
| V _(ESD) | Electrostatic discharge | Human-body model (HBM), per ANSI/ESDA/JEDEC JS-001 ⁽¹⁾ | ±2000 |
| | | Charged-device model (CDM), per JEDEC specification JESD22-C101 ⁽²⁾ | ±500 |

- (1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process. Manufacturing with less than 500-V HBM is possible with the necessary precautions. Pins listed as ±2000 V may actually have higher performance.
- (2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process. Manufacturing with less than 250-V CDM is possible with the necessary precautions. Pins listed as ±500 V may actually have higher performance.

6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

| | | MIN | NOM | MAX | UNIT |
|------------------|------------------------------------|------|-----|------|------|
| V _{IN} | Input voltage range | 0.5 | | 5.5 | V |
| V _{OUT} | Output voltage setting range | 2.2 | | 5.5 | V |
| L | Effective inductance range | 0.37 | 1.0 | 2.9 | µH |
| C _{IN} | Effective input capacitance range | 1.0 | 4.7 | | µF |
| C _{OUT} | Effective output capacitance range | 4 | 10 | 1000 | µF |
| T _J | Operating junction temperature | -40 | | 125 | °C |

6.4 Thermal Information

| THERMAL METRIC ⁽¹⁾ | | TPS61023 | TPS61023 | UNIT |
|-------------------------------|--|-----------------------|-----------------------|------|
| | | DRL (SOT563) - 6 PINS | DRL (SOT563) - 6 PINS | |
| | | Standard | EVM ⁽²⁾ | |
| R _{θJA} | Junction-to-ambient thermal resistance | 142.7 | 91.4 | °C/W |
| R _{θJC} | Junction-to-case thermal resistance | 55.7 | N/A | °C/W |
| R _{θJB} | Junction-to-board thermal resistance | 31.0 | N/A | °C/W |
| Ψ _{JT} | Junction-to-top characterization parameter | 1.4 | 5.3 | °C/W |
| Ψ _{JB} | Junction-to-board characterization parameter | 30.7 | 38.1 | °C/W |

- (1) For more information about traditional and new thermal metrics, see the [Semiconductor and IC Package Thermal Metrics](#) application report.
- (2) Measured on TPS61023EVM, 4-layer, 2oz copper 50mm×38mm PCB.

6.5 Electrical Characteristics

$T_J = -40^\circ\text{C}$ to 125°C , $V_{IN} = 3.6 \text{ V}$ and $V_{OUT} = 5.0 \text{ V}$. Typical values are at $T_J = 25^\circ\text{C}$ (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|------------------------|---|--|------|------|------|------------------|
| POWER SUPPLY | | | | | | |
| V_{IN} | Input voltage range | | 0.5 | 5.5 | | V |
| V_{IN_UVLO} | Under-voltage lockout threshold | V_{IN} rising | | 1.7 | 1.8 | V |
| | | V_{IN} falling | | 0.4 | 0.5 | V |
| I_Q | Quiescent current into V_{IN} pin | IC enabled, No load, No switching $V_{IN} = 1.8 \text{ V}$ to 5.5 V , $V_{FB} = V_{REF} + 0.1 \text{ V}$, T_J up to 85°C | | 0.9 | 3.0 | μA |
| | Quiescent current into V_{OUT} pin | IC enabled, No load, No switching $V_{OUT} = 2.2 \text{ V}$ to 5.5 V , $V_{FB} = V_{REF} + 0.1 \text{ V}$, T_J up to 85°C | | 20 | 30 | μA |
| I_{SD} | Shutdown current into V_{IN} and SW pin | IC disabled, $V_{IN} = V_{SW} = 3.6 \text{ V}$, $T_J = 25^\circ\text{C}$ | | 0.1 | 0.2 | μA |
| OUTPUT | | | | | | |
| V_{OUT} | Output voltage setting range | | 2.2 | 5.5 | | V |
| V_{REF} | Reference voltage at the FB pin | PWM mode | 580 | 595 | 610 | mV |
| | | PFM mode | 585 | 601 | | mV |
| V_{OVP} | Output over-voltage protection threshold | V_{OUT} rising | 5.5 | 5.7 | 6.0 | V |
| V_{OVP_HYS} | Over-voltage protection hysteresis | | | 0.1 | | V |
| I_{FB_LKG} | Leakage current at FB pin | $T_J = 25^\circ\text{C}$ | | 4 | 20 | nA |
| | | $T_J = 125^\circ\text{C}$ | | 6 | | nA |
| I_{VOUT_LKG} | Leakage current into V_{OUT} pin | IC disabled, $V_{IN} = 0 \text{ V}$, $V_{SW} = 0 \text{ V}$, $V_{OUT} = 5.5 \text{ V}$, $T_J = 25^\circ\text{C}$ | | 1 | 3 | μA |
| t_{SS} | Soft startup time | From active EN to V_{OUT} regulation. $V_{IN} = 2.5 \text{ V}$, $V_{OUT} = 5.0 \text{ V}$, $C_{OUT_EFF} = 10\mu\text{F}$, $I_{OUT} = 0$ | | 700 | | μs |
| POWER SWITCH | | | | | | |
| $R_{DS(on)}$ | High-side MOSFET on resistance | $V_{OUT} = 5.0 \text{ V}$ | | 68 | | $\text{m}\Omega$ |
| | Low-side MOSFET on resistance | $V_{OUT} = 5.0 \text{ V}$ | | 47 | | $\text{m}\Omega$ |
| f_{sw} | Switching frequency | $V_{IN} = 3.6 \text{ V}$, $V_{OUT} = 5.0 \text{ V}$, PWM mode | | 1.0 | | MHz |
| | | $V_{IN} = 1.0 \text{ V}$, $V_{OUT} = 5.0 \text{ V}$, PWM mode | | 0.5 | | MHz |
| t_{ON_min} | Minimum on time | | 40 | 96 | 130 | ns |
| t_{OFF_min} | Minimum off time | | | 80 | 120 | ns |
| I_{LIM_SW} | Valley current limit | $V_{IN} = 3.6 \text{ V}$, $V_{OUT} = 5.0 \text{ V}$ | 2.7 | 3.7 | | A |
| I_{LIM_CHG} | Pre-charge current | $V_{IN} = 1.8 - 5.5 \text{ V}$, $V_{OUT} < 0.4 \text{ V}$ | 200 | 350 | | mA |
| | | $V_{IN} = 2.4 \text{ V}$, $V_{OUT} = 2.15 \text{ V}$ | 750 | 1200 | | mA |
| LOGIC INTERFACE | | | | | | |
| V_{EN_H} | EN logic high threshold | $V_{IN} > 1.8 \text{ V}$ or $V_{OUT} > 2.2 \text{ V}$ | | 1.2 | | V |
| V_{EN_L} | EN logic low threshold | $V_{IN} > 1.8 \text{ V}$ or $V_{OUT} > 2.2 \text{ V}$ | 0.35 | 0.42 | 0.45 | |
| PROTECTION | | | | | | |
| T_{SD} | Thermal shutdown threshold | T_J rising | | 150 | | $^\circ\text{C}$ |
| T_{SD_HYS} | Thermal shutdown hysteresis | T_J falling below T_{SD} | | 20 | | $^\circ\text{C}$ |

6.6 Typical Characteristics

$V_{IN} = 3.6\text{ V}$, $V_{OUT} = 5\text{ V}$, $T_J = 25^\circ\text{C}$, unless otherwise noted

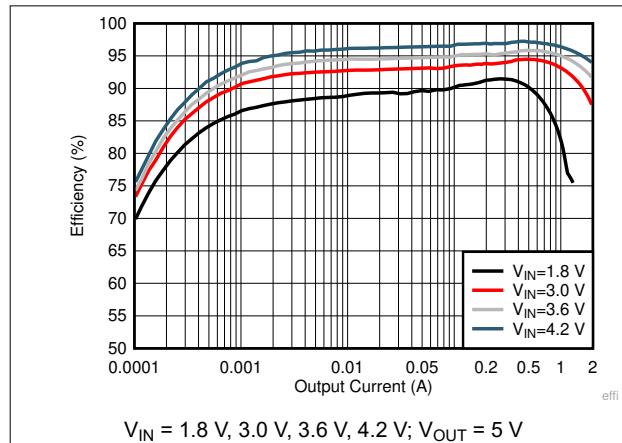


Figure 6-1. Load Efficiency With Different Input

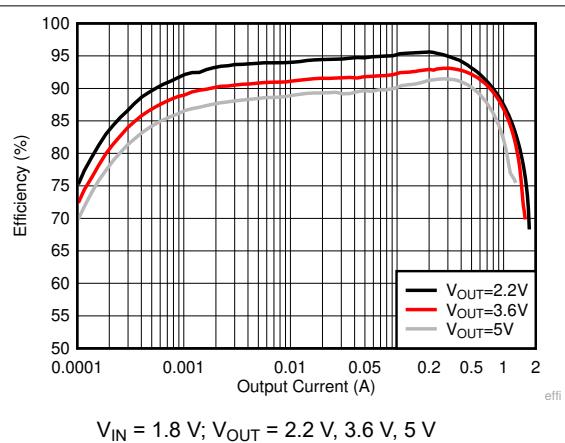


Figure 6-2. Load Efficiency With Different Output

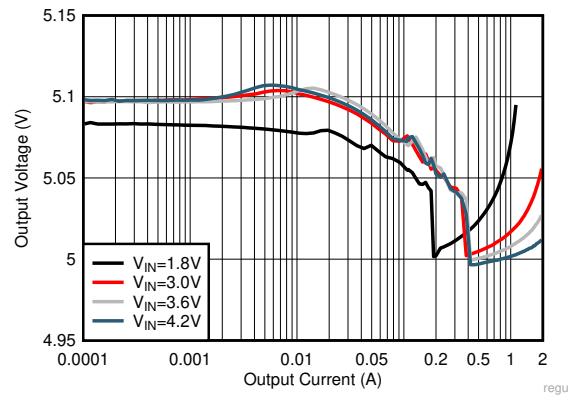


Figure 6-3. Load Regulation

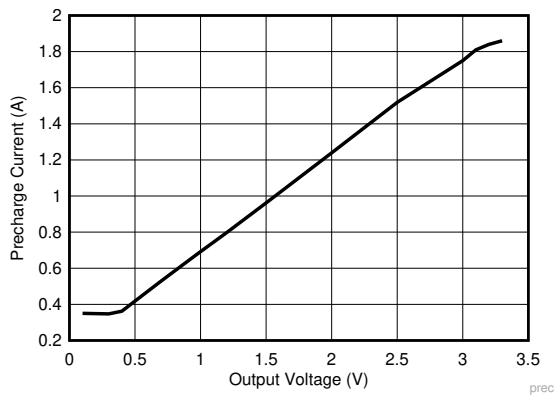


Figure 6-4. Pre-charge Current vs Output Voltage

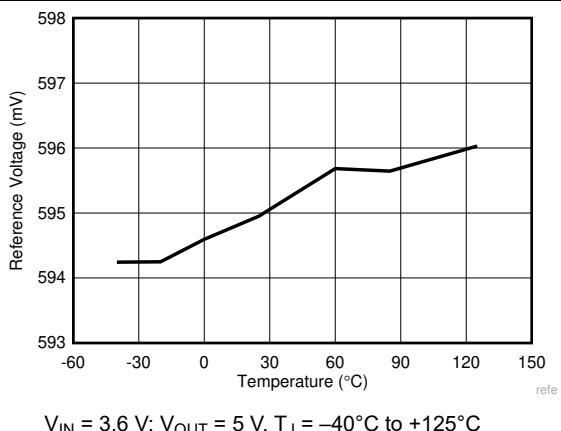


Figure 6-5. Reference Voltage vs Temperature

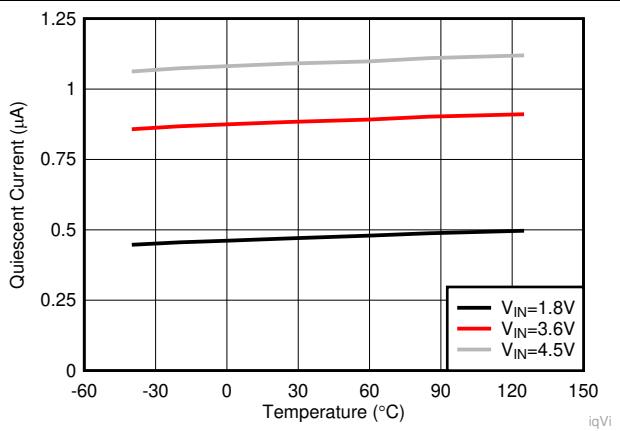
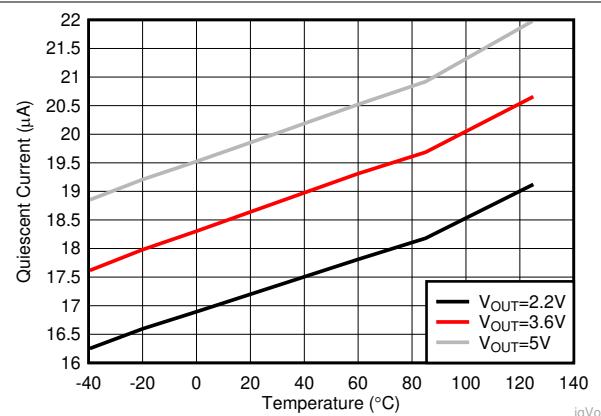
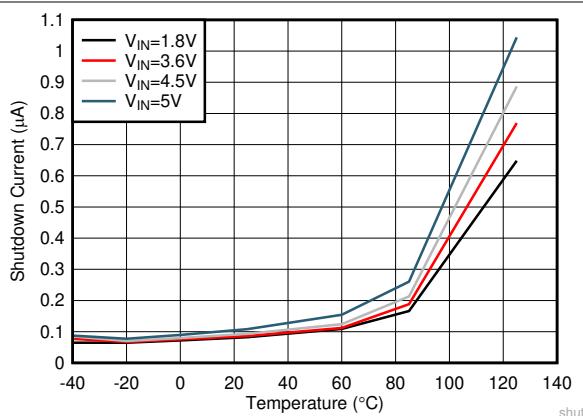


Figure 6-6. Quiescent Current into VIN vs Temperature



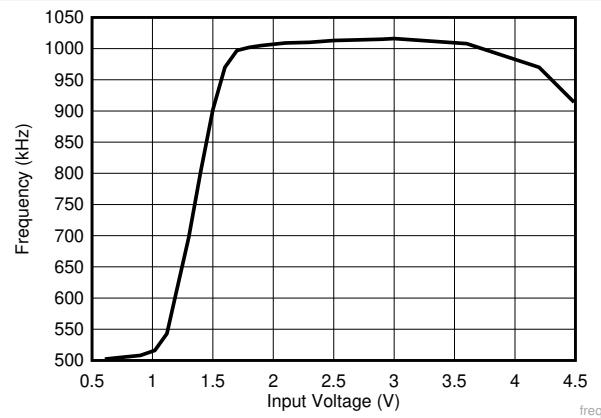
$V_{IN} = 1.8 \text{ V}$; $V_{OUT} = 2.2 \text{ V}, 3.6 \text{ V}, 5 \text{ V}$; $T_J = -40^\circ\text{C}$ to $+125^\circ\text{C}$, No switching

Figure 6-7. Quiescent Current into VOUT vs Temperature



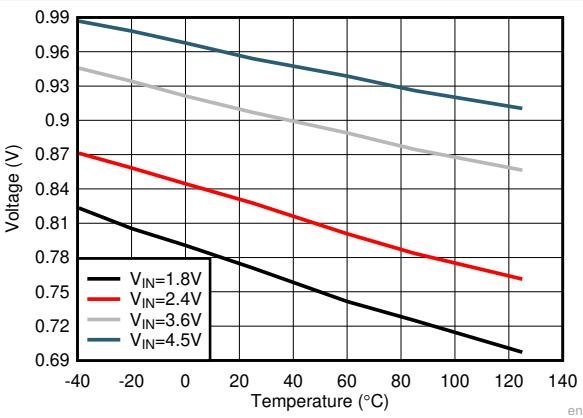
$V_{IN} = V_{SW} = 1.8 \text{ V}, 3.6 \text{ V}, 4.5 \text{ V}, 5 \text{ V}$; $V_{OUT} = 0 \text{ V}$; $T_J = -40^\circ\text{C}$ to $+125^\circ\text{C}$

Figure 6-8. Shutdown Current vs Temperature



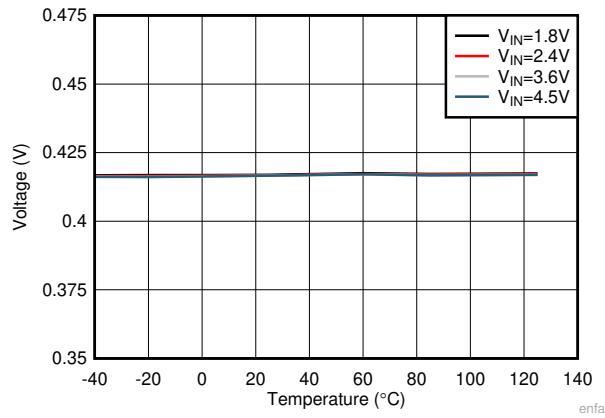
$V_{IN} = 0.5 \text{ V}$ to 4.5 V ; $V_{OUT} = 5 \text{ V}$

Figure 6-9. Switching Frequency vs Input Voltage



$V_{IN} = 1.8 \text{ V}, 2.4 \text{ V}, 3.6 \text{ V}, 4.5 \text{ V}$; $V_{OUT} = 0 \text{ V}$; $T_J = -40^\circ\text{C}$ to $+125^\circ\text{C}$

Figure 6-10. EN Rising Threshold vs Temperature



$V_{IN} = 1.8 \text{ V}, 2.4 \text{ V}, 3.6 \text{ V}, 4.5 \text{ V}$; $V_{OUT} = 0 \text{ V}$; $T_J = -40^\circ\text{C}$ to $+125^\circ\text{C}$

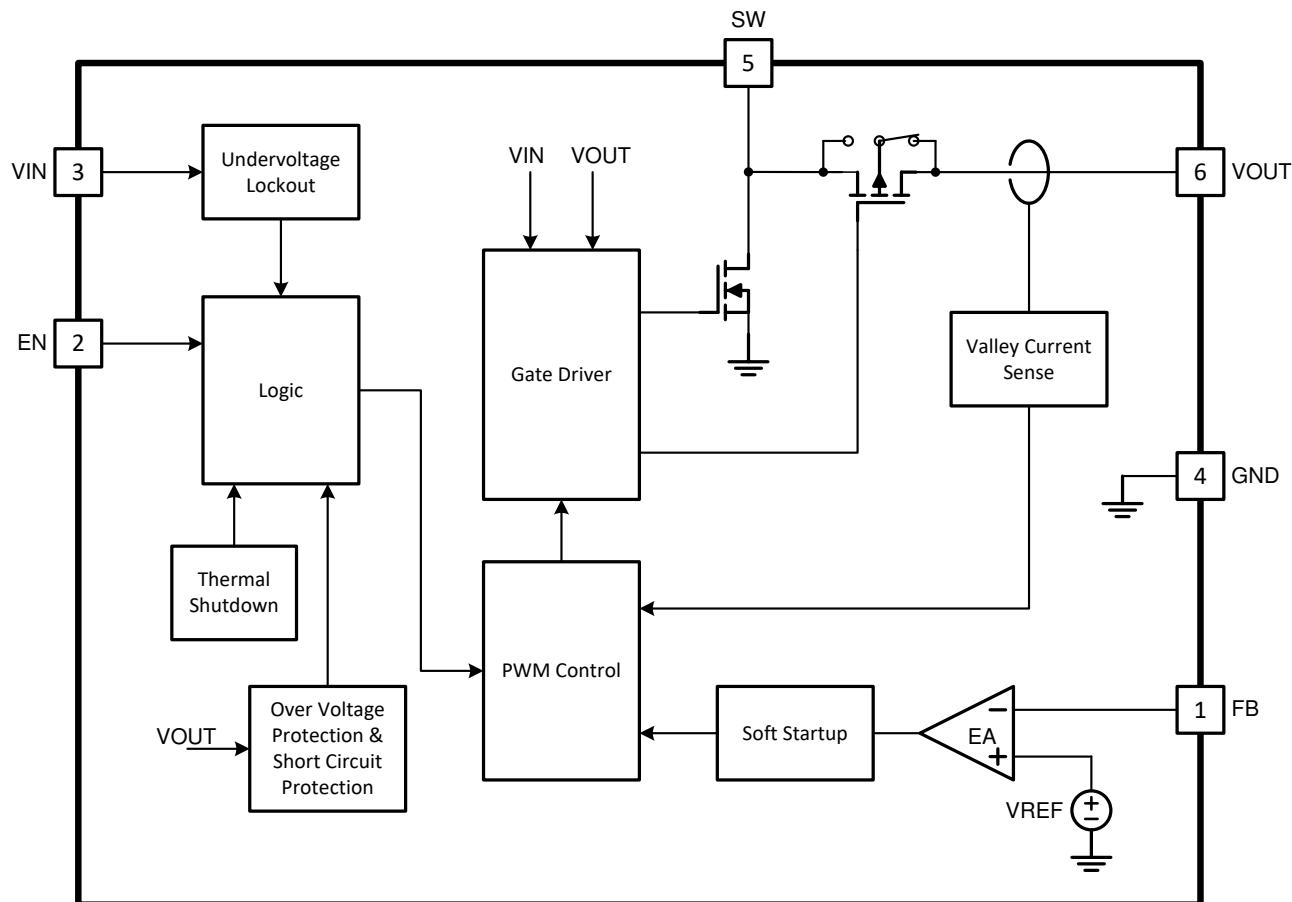
Figure 6-11. EN Falling Threshold vs Temperature

7 Detailed Description

7.1 Overview

The TPS61023 synchronous step-up converter is designed to operate from an input voltage supply range between 0.5 V and 5.5 V with 3.7-A (typical) valley switch current limit. The TPS61023 typically operates at a quasi-constant frequency pulse width modulation (PWM) at moderate to heavy load currents. The switching frequency is 1 MHz when the input voltage is above 1.5 V. The switching frequency reduces down to 0.5 MHz gradually when the input voltage goes down from 1.5 V to 1 V and keeps at 0.5 MHz when the input voltage is below 1 V. At light load conditions, the TPS61023 converter operates in power-save mode with pulse frequency modulation (PFM). During PWM operation, the converter uses adaptive constant on-time valley current mode control scheme to achieve excellent line regulation and load regulation and allows the use of a small inductor and ceramic capacitors. Internal loop compensation simplifies the design process while minimizing the number of external components.

7.2 Functional Block Diagram



7.3 Feature Description

7.3.1 Undervoltage Lockout

The TPS61023 has a built-in undervoltage lockout (UVLO) circuit to ensure the device working properly. When the input voltage is above the UVLO rising threshold of 1.8 V, the TPS61023 can be enabled to boost the output voltage. After the TPS61023 starts up and the output voltage is above 2.2 V, the TPS61023 works with input voltage as low as 0.5 V.

7.3.2 Enable and Soft Start

When the input voltage is above the UVLO rising threshold and the EN pin is pulled to a voltage above 1.2 V, the TPS61023 is enabled and starts up. At the beginning, the TPS61023 charges the output capacitors with a current of about 350 mA when the output voltage is below 0.4 V. When the output voltage is charged above 0.4 V, the output current is changed to having output current capability to drive the 2- Ω resistance load. After the output voltage reaches the input voltage, the TPS61023 starts switching, and the output voltage ramps up further. The typical start-up time is 700 μ s accounting from EN high to output reaching target voltage for the application with input voltage is 2.5 V, output voltage is 5 V, output effective capacitance is 10 μ F, and no load. When the voltage at the EN pin is below 0.4 V, the internal enable comparator turns the device into shutdown mode. In the shutdown mode, the device is entirely turned off. The output is disconnected from input power supply.

7.3.3 Switching Frequency

The TPS61023 switches at a quasi-constant 1-MHz frequency when the input voltage is above 1.5 V. When the input voltage is lower than 1.5 V, the switching frequency is reduced gradually to 0.5 MHz to improve the efficiency and get higher boost ratio. When the input voltage is below 1 V, the switching frequency is fixed at a quasi-constant 0.5 MHz.

7.3.4 Current Limit Operation

The TPS61023 uses a valley current limit sensing scheme. Current limit detection occurs during the off-time by sensing of the voltage drop across the synchronous rectifier.

When the load current is increased such that the inductor current is above the current limit within the whole switching cycle time, the off-time is increased to allow the inductor current to decrease to this threshold before the next on-time begins (so called frequency foldback mechanism). When the current limit is reached, the output voltage decreases during further load increase.

The maximum continuous output current ($I_{OUT(LC)}$), before entering current limit (CL) operation, can be defined by [Equation 1](#).

$$I_{OUT(CL)} = (1 - D) \times \left(I_{LIM} + \frac{1}{2} \Delta I_{L(P-P)} \right) \quad (1)$$

where

- D is the duty cycle
- $\Delta I_{L(P-P)}$ is the inductor ripple current

The duty cycle can be estimated by [Equation 2](#).

$$D = 1 - \frac{V_{IN} \times \eta}{V_{OUT}} \quad (2)$$

where

- V_{OUT} is the output voltage of the boost converter
- V_{IN} is the input voltage of the boost converter
- η is the efficiency of the converter, use 90% for most applications

The peak-to-peak inductor ripple current is calculated by [Equation 3](#).

$$\Delta I_{L(P-P)} = \frac{V_{IN} \times D}{L \times f_{SW}} \quad (3)$$

where

- L is the inductance value of the inductor
- f_{SW} is the switching frequency
- D is the duty cycle
- V_{IN} is the input voltage of the boost converter

7.3.5 Pass-Through Operation

When the input voltage is higher than the setting output voltage, the output voltage is higher than the target regulation voltage. When the output voltage is 101% of the setting target voltage, the TPS61023 stops switching and fully turns on the high-side PMOS FET. The device works in pass-through mode. The output voltage is the input voltage minus the voltage drop across the DCR of the inductor and the $R_{DS(on)}$ of the PMOS FET. When the output voltage drops below the 97% of the setting target voltage as the input voltage declines or the load current increases, the TPS61023 resumes switching again to regulate the output voltage.

7.3.6 Overvoltage Protection

The TPS61023 has an output overvoltage protection (OVP) to protect the device if the external feedback resistor divider is wrongly populated. When the output voltage is above 5.7 V typically, the device stops switching. Once the output voltage falls 0.1 V below the OVP threshold, the device resumes operating again.

7.3.7 Output Short-to-Ground Protection

The TPS61023 starts to limit the output current when the output voltage is below 1.8 V. The lower the output voltage reaches, the smaller the output current is. When the VOUT pin is short to ground, and the output voltage becomes less than 0.4 V, the output current is limited to approximately 350 mA. Once the short circuit is released, the TPS61023 goes through the soft start-up again to the regulated output voltage.

7.3.8 Thermal Shutdown

The TPS61023 goes into thermal shutdown once the junction temperature exceeds 150°C. When the junction temperature drops below the thermal shutdown recovery temperature, typically 130°C, the device starts operating again.

7.4 Device Functional Modes

The TPS61023 has two switching operation modes, PWM mode in moderate to heavy load conditions and power save mode with pulse frequency modulation (PFM) in light load conditions.

7.4.1 PWM Mode

The TPS61023 uses a quasi-constant 1.0-MHz frequency pulse width modulation (PWM) at moderate to heavy load current. Based on the input voltage to output voltage ratio, a circuit predicts the required on-time. At the beginning of the switching cycle, the NMOS switching FET. The input voltage is applied across the inductor and the inductor current ramps up. In this phase, the output capacitor is discharged by the load current. When the on-time expires, the main switch NMOS FET is turned off, and the rectifier PMOS FET is turned on. The inductor transfers its stored energy to replenish the output capacitor and supply the load. The inductor current declines because the output voltage is higher than the input voltage. When the inductor current hits the valley current threshold determined by the output of the error amplifier, the next switching cycle starts again.

The TPS61023 has a built-in compensation circuit that can accommodate a wide range of input voltage, output voltage, inductor value, and output capacitor value for stable operation.

7.4.2 Power-Save Mode

The TPS61023 integrates a power-save mode with PFM to improve efficiency at light load. When the load current decreases, the inductor valley current set by the output of the error amplifier no longer regulates the output voltage. When the inductor valley current hits the low limit, the output voltage exceeds the setting voltage as the load current decreases further. When the FB voltage hits the PFM reference voltage, the TPS61023 goes into the power-save mode. In the power-save mode, when the FB voltage rises and hits the PFM reference voltage, the device continues switching for several cycles because of the delay time of the internal comparator — then it stops switching. The load is supplied by the output capacitor, and the output voltage declines. When the FB voltage falls below the PFM reference voltage, after the delay time of the comparator, the device starts switching again to ramp up the output voltage.

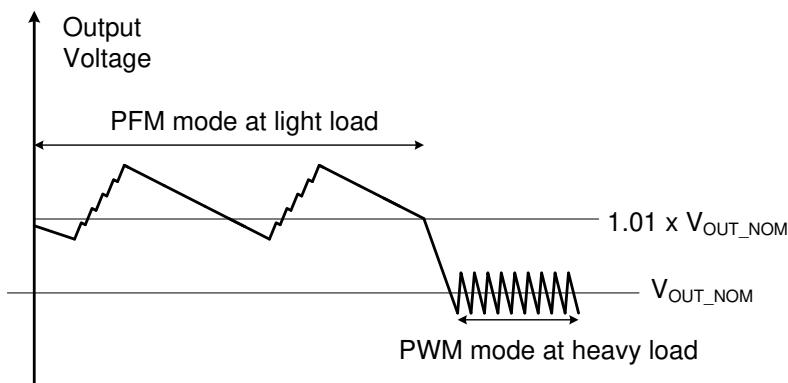


Figure 7-1. Output Voltage in PWM Mode and PFM Mode

8 Application and Implementation

Note

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

8.1 Application Information

The TPS61023 is a synchronous boost converter designed to operate from an input voltage supply range between 0.5 V and 5.5 V with a typically 3.7-A valley switch current limit. The TPS61023 typically operates at a quasi-constant 1-MHz frequency PWM at moderate-to-heavy load currents when the input voltage is above 1.5 V. The switching frequency changes to 0.5 MHz gradually with the input voltage changing from 1.5 V to 1 V for better efficiency and high step-up ratio. When the input voltage is below 1 V, the switching frequency is fixed at a quasi-constant 0.5 MHz. At light load currents, the TPS61023 converter operates in power-save mode with PFM to achieve high efficiency over the entire load current range.

8.2 Typical Application

The TPS61023 provides a power supply solution for portable devices powered by batteries or backup applications powered by super-capacitors. With typical 3.7-A switch current capability, the TPS61023 can output 5 V and 1.5 A from a single-cell Li-ion battery.

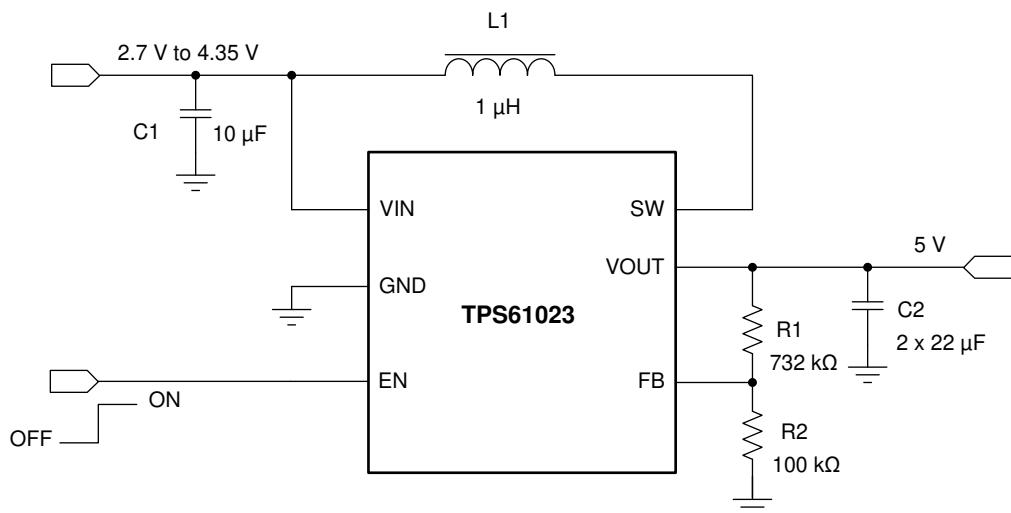


Figure 8-1. Li-ion Battery to 5-V Boost Converter

8.2.1 Design Requirements

The design parameters are listed in [Table 8-1](#).

Table 8-1. Design Parameters

| PARAMETERS | VALUES |
|-----------------------|-----------------|
| Input voltage | 2.7 V to 4.35 V |
| Output voltage | 5 V |
| Output current | 1.5 A |
| Output voltage ripple | ±50 mV |

8.2.2 Detailed Design Procedure

8.2.2.1 Setting the Output Voltage

The output voltage is set by an external resistor divider (R_1 , R_2 in [Figure 8-1](#)). When the output voltage is regulated, the typical voltage at the FB pin is V_{REF} . Thus the resistor divider is determined by [Equation 4](#).

$$R_1 = \left(\frac{V_{OUT}}{V_{REF}} - 1 \right) \times R_2 \quad (4)$$

where

- V_{OUT} is the regulated output voltage
- V_{REF} is the internal reference voltage at the FB pin

For the best accuracy, should be kept R_2 smaller than 300 kΩ to ensure the current flowing through R_2 is at least 100 times larger than the FB pin leakage current. Changing R_2 towards a lower value increases the immunity against noise injection. Changing the R_2 towards a higher value reduces the quiescent current for achieving highest efficiency at low load currents.

8.2.2.2 Inductor Selection

Because the selection of the inductor affects steady-state operation, transient behavior, and loop stability. The inductor is the most important component in power regulator design. There are three important inductor specifications, inductor value, saturation current, and dc resistance (DCR).

The TPS61023 is designed to work with inductor values between 0.37 μH and 2.9 μH. Follow [Equation 5](#) to [Equation 7](#) to calculate the inductor peak current for the application. To calculate the current in the worst case, use the minimum input voltage, maximum output voltage, and maximum load current of the application. To have enough design margins, choose the inductor value with –30% tolerances, and low power-conversion efficiency for the calculation.

In a boost regulator, the inductor dc current can be calculated by [Equation 5](#).

$$I_{L(DC)} = \frac{V_{OUT} \times I_{OUT}}{V_{IN} \times \eta} \quad (5)$$

where

- V_{OUT} is the output voltage of the boost converter
- I_{OUT} is the output current of the boost converter
- V_{IN} is the input voltage of the boost converter
- η is the power conversion efficiency, use 90% for most applications

The inductor ripple current is calculated by [Equation 6](#).

$$\Delta I_{L(P-P)} = \frac{V_{IN} \times D}{L \times f_{SW}} \quad (6)$$

where

- D is the duty cycle, which can be calculated by [Equation 2](#)
- L is the inductance value of the inductor
- f_{SW} is the switching frequency
- V_{IN} is the input voltage of the boost converter

Therefore, the inductor peak current is calculated by [Equation 7](#).

$$I_{L(P)} = I_{L(DC)} + \frac{\Delta I_{L(P-P)}}{2} \quad (7)$$

Normally, it is advisable to work with an inductor peak-to-peak current of less than 40% of the average inductor current for maximum output current. A smaller ripple from a larger valued inductor reduces the magnetic hysteresis losses in the inductor and EMI. But in the same way, load transient response time is increased. The saturation current of the inductor must be higher than the calculated peak inductor current. [Table 8-2](#) lists the recommended inductors for the TPS61023.

Table 8-2. Recommended Inductors for the TPS61023

| PART NUMBER ⁽¹⁾ | L (μH) | DCR MAX (mΩ) | SATURATION CURRENT (A) | SIZE (LxWxH) | VENDOR |
|----------------------------|--------|--------------|------------------------|-----------------|------------------|
| XEL4030-102ME | 1 | 9.78 | 9.0 | 4.0 × 4.0 × 3.1 | Coilcraft |
| 74438357010 | 1 | 13.5 | 9.6 | 4.1 × 4.1 × 3.1 | Wurth Elektronik |
| HBME042A-1R0MS-99 | 1 | 11.5 | 7.0 | 4.1 × 4.1 × 2.1 | Cyntec |

(1) See [Third-party Products](#) disclaimer

8.2.2.3 Output Capacitor Selection

The output capacitor is mainly selected to meet the requirements for output ripple and loop stability. The ripple voltage is related to capacitor capacitance and its equivalent series resistance (ESR). Assuming a ceramic capacitor with zero ESR, the minimum capacitance needed for a given ripple voltage can be calculated by [Equation 8](#).

$$C_{OUT} = \frac{I_{OUT} \times D_{MAX}}{f_{SW} \times V_{RIPPLE}} \quad (8)$$

where

- D_{MAX} is the maximum switching duty cycle
- V_{RIPPLE} is the peak-to-peak output ripple voltage
- I_{OUT} is the maximum output current
- f_{SW} is the switching frequency

The ESR impact on the output ripple must be considered if tantalum or aluminum electrolytic capacitors are used. The output peak-to-peak ripple voltage caused by the ESR of the output capacitors can be calculated by [Equation 9](#).

$$V_{RIPPLE(ESR)} = I_{L(P)} \times R_{ESR} \quad (9)$$

Take care when evaluating the derating of a ceramic capacitor under dc bias voltage, aging, and ac signal. For example, the dc bias voltage can significantly reduce capacitance. A ceramic capacitor can lose more than 50% of its capacitance at its rated voltage. Therefore, always leave margin on the voltage rating to ensure adequate capacitance at the required output voltage. Increasing the output capacitor makes the output ripple voltage smaller in PWM mode.

TI recommends using the X5R or X7R ceramic output capacitor in the range of 4-μF to 1000-μF effective capacitance. The output capacitor affects the small signal control loop stability of the boost regulator. If the output capacitor is below the range, the boost regulator can potentially become unstable. Increasing the output capacitor makes the output ripple voltage smaller in PWM mode.

8.2.2.4 Loop Stability, Feedforward Capacitor Selection

When the switching waveform shows large duty cycle jitter or the output voltage or inductor current shows oscillations, the regulation loop can be unstable.

The load transient response is another approach to check the loop stability. During the load transient recovery time, V_{OUT} can be monitored for settling time, overshoot or ringing that helps judge the stability of the converters. Without any ringing, the loop has usually more than 45° of phase margin.

A feedforward capacitor (C_3 in the [Figure 8-2](#)) in parallel with R_1 induces a pair of zero and pole in the loop transfer function. By setting the proper zero frequency, the feedforward capacitor can increase the phase margin to improve the loop stability. For large output capacitance more than 40 μF application, TI recommends a feedforward capacitor to set the zero frequency (f_{FFZ}) to 1 kHz. As for the input voltage lower than 1-V application, TI recommends to use the effective output capacitance is about 100 μF and set the zero frequency (f_{FFZ}) to 1 kHz. The value of the feedforward capacitor can be calculated by [Equation 10](#).

$$C_3 = \frac{1}{2\pi \times f_{FFZ} \times R_1} \quad (10)$$

where

- R_1 is the resistor between the V_{OUT} pin and FB pin
- f_{FFZ} is the zero frequency created by the feedforward capacitor

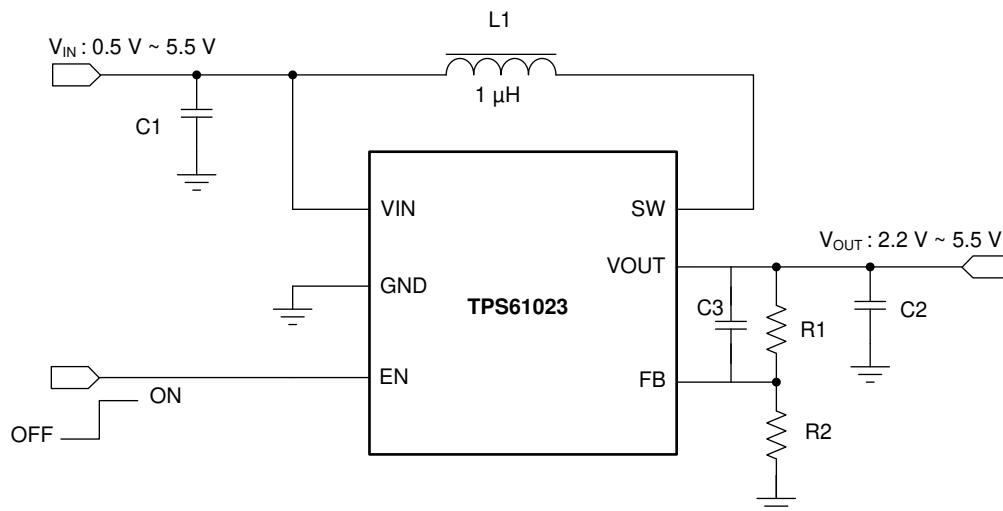


Figure 8-2. TPS61023 Circuit With Feedforward Capacitor

8.2.2.5 Input Capacitor Selection

Multilayer X5R or X7R ceramic capacitors are excellent choices for the input decoupling of the step-up converter as they have extremely low ESR and are available in small footprints. Input capacitors must be located as close as possible to the device. While a 10- μF input capacitor is sufficient for most applications, larger values may be used to reduce input current ripple without limitations. Take care when using only ceramic input capacitors. When a ceramic capacitor is used at the input and the power is being supplied through long wires, a load step at the output can induce ringing at the VIN pin. This ringing can couple to the output and be mistaken as loop instability or could even damage the part. In this circumstance, place additional bulk capacitance (tantalum or aluminum electrolytic capacitor) between ceramic input capacitor and the power source to reduce ringing that can occur between the inductance of the power source leads and ceramic input capacitor.

8.2.3 Application Curves

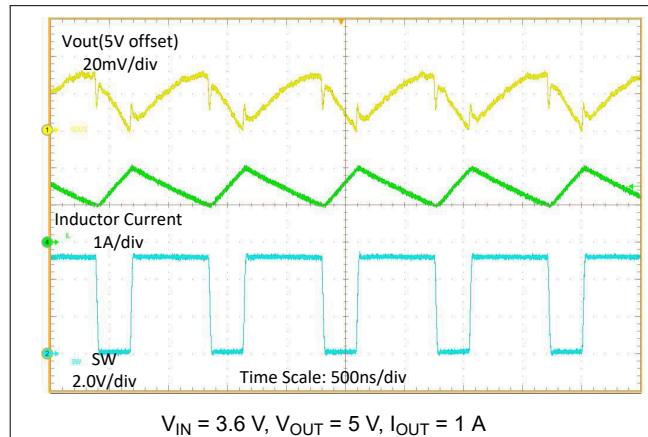


Figure 8-3. Switching Waveform at Heavy Load

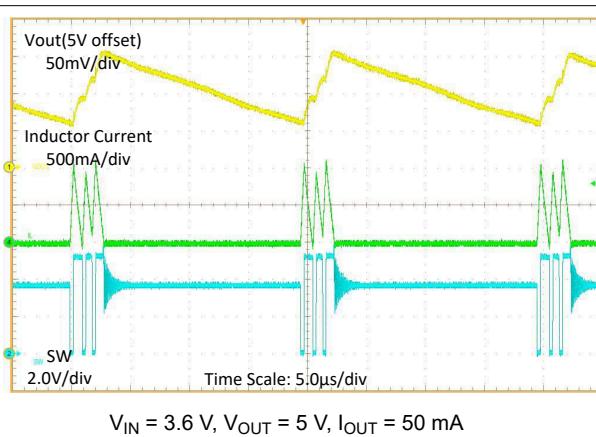


Figure 8-4. Switching Waveform at Light Load

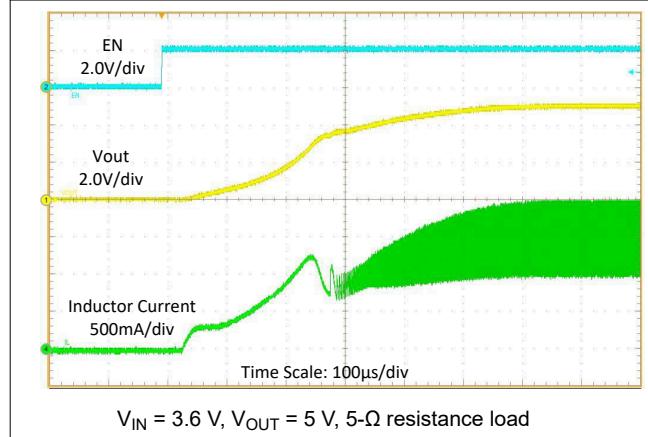


Figure 8-5. Start-up Waveform

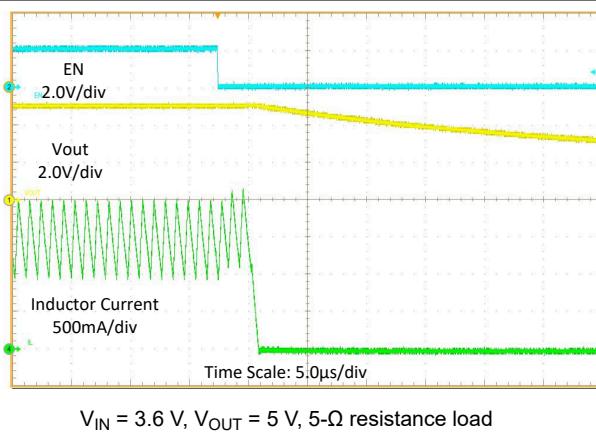


Figure 8-6. Shutdown Waveform

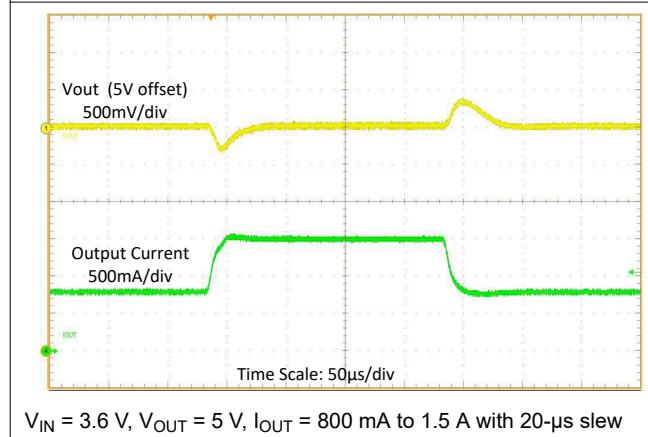


Figure 8-7. Load Transient

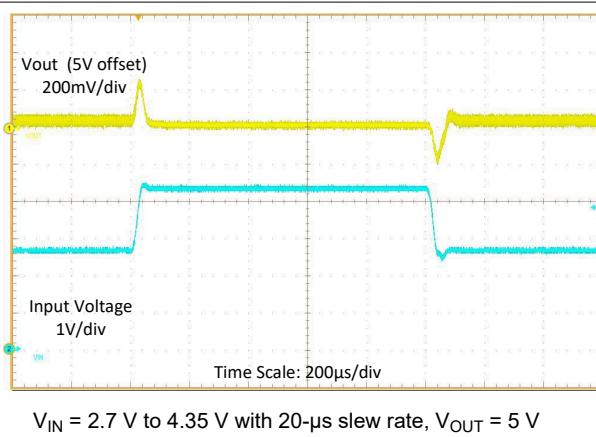


Figure 8-8. Line Transient

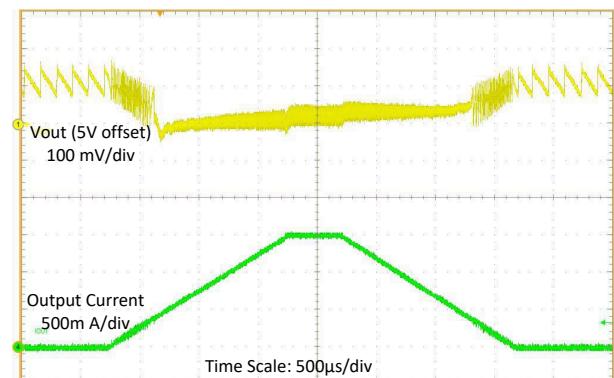


Figure 8-9. Load Sweep

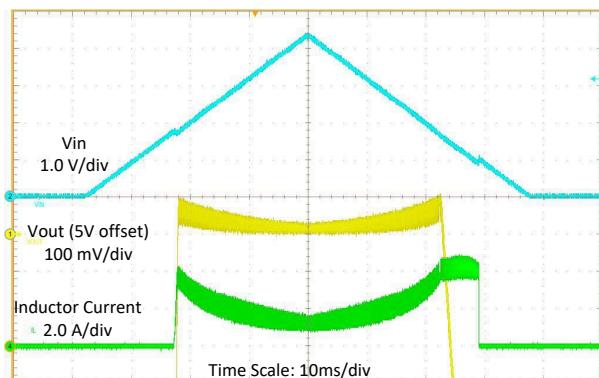


Figure 8-10. Line Sweep

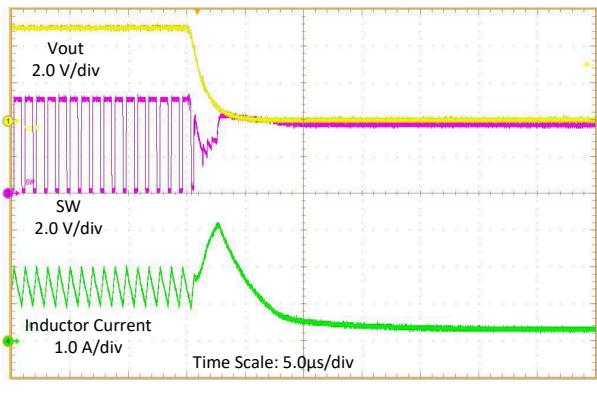


Figure 8-11. Output Short Protection (Entry)

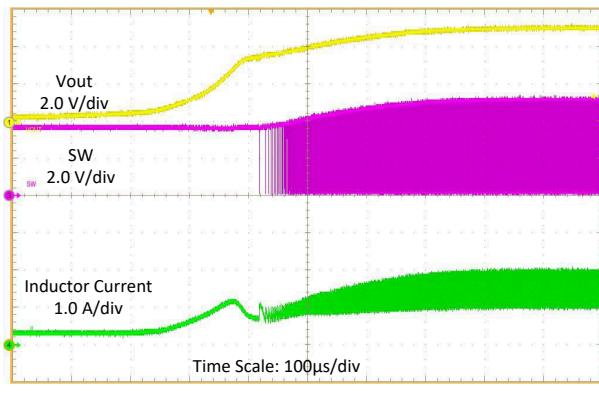


Figure 8-12. Output Short Protection (Recover)

9 Power Supply Recommendations

The device is designed to operate from an input voltage supply range between 0.5 V to 5.5 V. This input supply must be well regulated. If the input supply is located more than a few inches from the converter, additional bulk capacitance may be required in addition to the ceramic bypass capacitors. A typical choice is a tantalum or aluminum electrolytic capacitor with a value of 100 μ F. Output current of the input power supply must be rated according to the supply voltage, output voltage, and output current of the TPS61023.

10 Layout

10.1 Layout Guidelines

As for all switching power supplies, especially those running at high switching frequency and high currents, layout is an important design step. If the layout is not carefully done, the regulator could suffer from instability and noise problems. To maximize efficiency, switch rise and fall time are very fast. To prevent radiation of high frequency noise (for example, EMI), proper layout of the high-frequency switching path is essential. Minimize the length and area of all traces connected to the SW pin, and always use a ground plane under the switching regulator to minimize interplane coupling. The input capacitor needs not only to be close to the VIN pin, but also to the GND pin in order to reduce input supply ripple.

The most critical current path for all boost converters is from the switching FET, through the rectifier FET, then the output capacitors, and back to ground of the switching FET. This high current path contains nanosecond rise and fall time and must be kept as short as possible. Therefore, the output capacitor not only must be close to the VOUT pin, but also to the GND pin to reduce the overshoot at the SW pin and VOUT pin.

For better thermal performance, TI suggest to make copper polygon connected with each pin bigger.

10.2 Layout Example

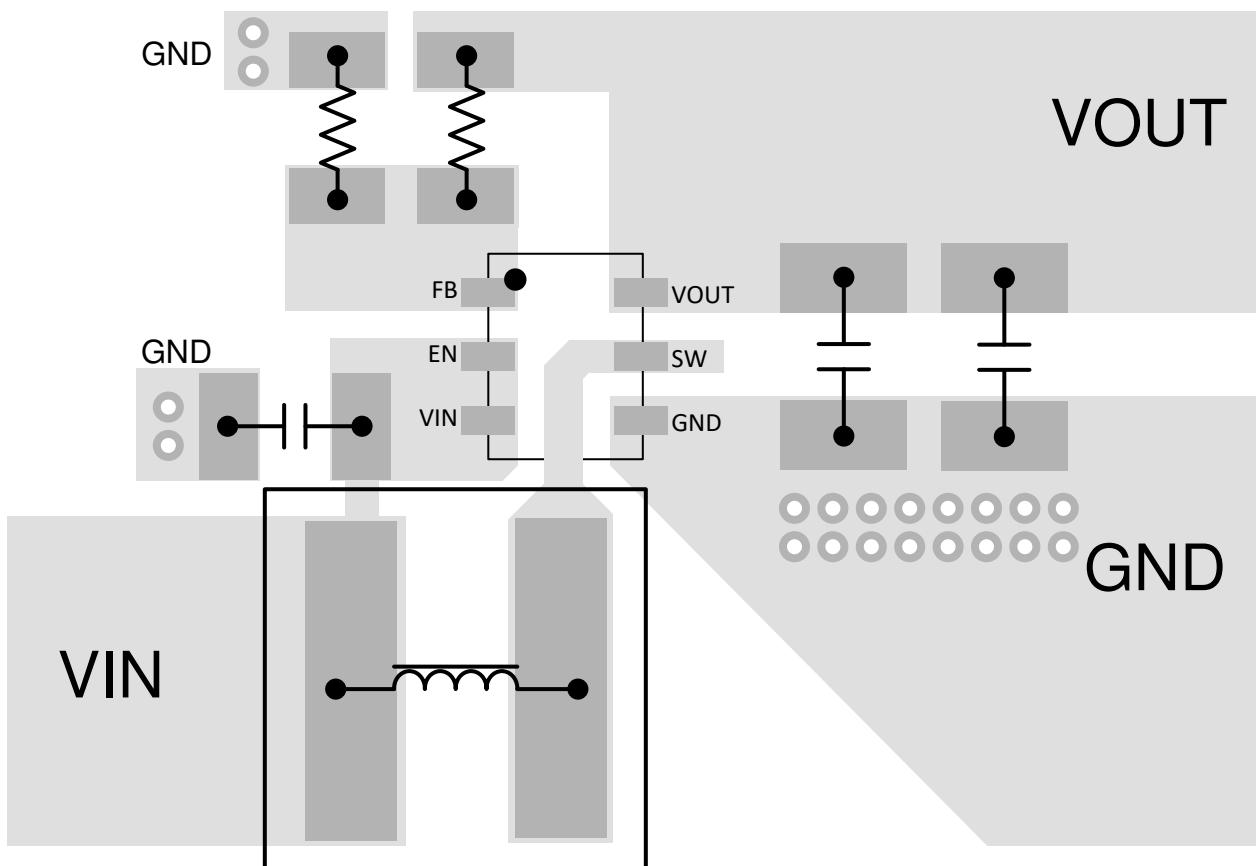


Figure 10-1. Layout Example

10.3 Thermal Considerations

Restrict the maximum IC junction temperature to 125°C under normal operating conditions. Calculate the maximum allowable dissipation, $P_{D(\max)}$, and keep the actual power dissipation less than or equal to $P_{D(\max)}$. The maximum-power-dissipation limit is determined using [Equation 11](#).

$$P_{D(\max)} = \frac{125 - T_A}{R_{\theta JA}} \quad (11)$$

where

- T_A is the maximum ambient temperature for the application
- $R_{\theta JA}$ is the junction-to-ambient thermal resistance given in [Thermal Information](#)

The TPS61023 comes in a SOT563 package. The real junction-to-ambient thermal resistance of the package greatly depends on the PCB type, layout. Using larger and thicker PCB copper for the power pads (GND, SW, and VOUT) to enhance the thermal performance. Using more vias connects the ground plate on the top layer and bottom layer around the IC without solder mask also improves the thermal capability.

11 Device and Documentation Support

11.1 Device Support

11.1.1 Third-Party Products Disclaimer

TI'S PUBLICATION OF INFORMATION REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE AN ENDORSEMENT REGARDING THE SUITABILITY OF SUCH PRODUCTS OR SERVICES OR A WARRANTY, REPRESENTATION OR ENDORSEMENT OF SUCH PRODUCTS OR SERVICES, EITHER ALONE OR IN COMBINATION WITH ANY TI PRODUCT OR SERVICE.

11.2 Receiving Notification of Documentation Updates

To receive notification of documentation updates, navigate to the device product folder on ti.com. Click on *Subscribe to updates* to register and receive a weekly digest of any product information that has changed. For change details, review the revision history included in any revised document.

11.3 Support Resources

[TI E2E™ support forums](#) are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's [Terms of Use](#).

11.4 Trademarks

TI E2E™ is a trademark of Texas Instruments.

All trademarks are the property of their respective owners.

11.5 Electrostatic Discharge Caution



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

11.6 Glossary

[TI Glossary](#) This glossary lists and explains terms, acronyms, and definitions.

12 Mechanical, Packaging, and Orderable Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

12.1 Package Option Addendum

Packaging Information

| Orderable Device | Status ⁽¹⁾ | Package Type | Package Drawing | Pins | Package Qty | Eco Plan ⁽²⁾ | Lead/Ball Finish ⁽⁶⁾ | MSL Peak Temp ⁽³⁾ | Op Temp (°C) | Device Marking ⁽⁴⁾ ⁽⁵⁾ |
|------------------|-----------------------|--------------|-----------------|------|-------------|-------------------------|---------------------------------|------------------------------|--------------|--|
| TPS61023DRLR | ACTIVE | SOT-5X3 | DRL | 6 | 4000 | Green (RoHS & no Sb/Br) | Call TISN | Level-1-260-UNLIM | -40 to 125 | 1GI |

1. The marketing status values are defined as follows:

ACTIVE: Product device recommended for new designs.

LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

PRE_PROD Unannounced device, not in production, not available for mass market, nor on the web, samples not available.

PREVIEW: Device has been announced but is not in production. Samples may or may not be available.

OBSOLETE: TI has discontinued the production of the device.

2. Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check www.ti.com/productcontent for the latest availability information and additional product content details.

TBD: The Pb-Free/Green conversion plan has not been defined.

Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.

Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.

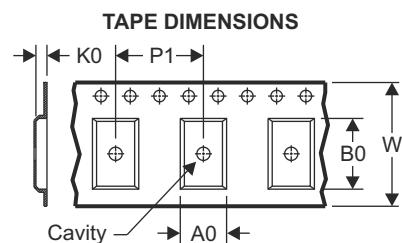
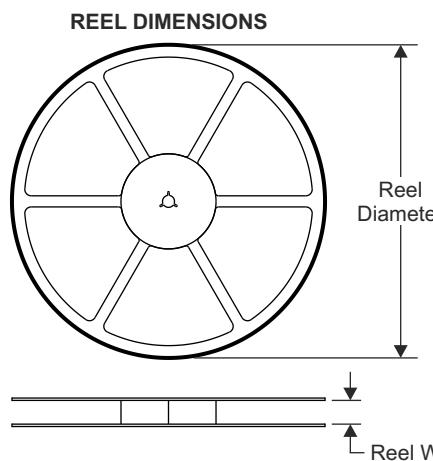
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material).

3. MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.
4. There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.
5. Multiple Device markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.
6. Lead/Ball Finish - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead/Ball Finish values may wrap to two lines if the finish value exceeds the maximum column width.

Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

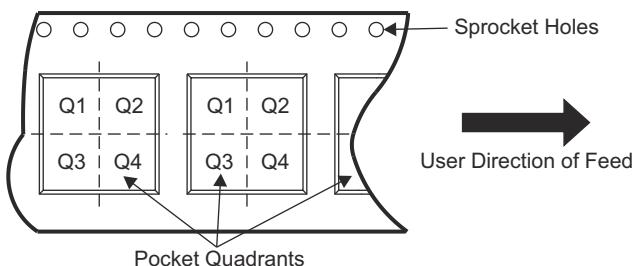
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

12.2 Tape and Reel Information



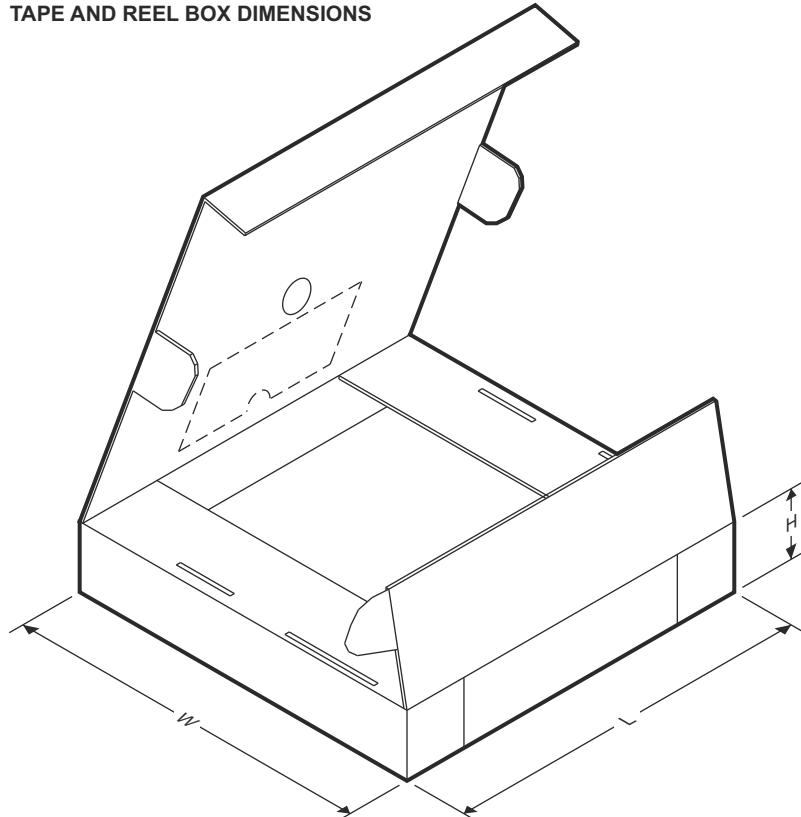
| | |
|----|---|
| A0 | Dimension designed to accommodate the component width |
| B0 | Dimension designed to accommodate the component length |
| K0 | Dimension designed to accommodate the component thickness |
| W | Overall width of the carrier tape |
| P1 | Pitch between successive cavity centers |

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



| Device | Package Type | Package Drawing | Pins | SPQ | Reel Diameter (mm) | Reel Width W1 (mm) | A0 (mm) | B0 (mm) | K0 (mm) | P1 (mm) | W (mm) | Pin1 Quadrant |
|--------------|--------------|-----------------|------|------|--------------------|--------------------|---------|---------|---------|---------|--------|---------------|
| TPS61023DRLR | SOT-5X3 | DRL | 6 | 4000 | 180.0 | 8.4 | 2.0 | 1.8 | 0.75 | 4.0 | 8.0 | Q3 |

TAPE AND REEL BOX DIMENSIONS



| Device | Package Type | Package Drawing | Pins | SPQ | Length (mm) | Width (mm) | Height (mm) |
|--------------|--------------|-----------------|------|------|-------------|------------|-------------|
| TPS61023DRLR | SOT-5X3 | DRL | 6 | 4000 | 182.0 | 182.0 | 20.0 |

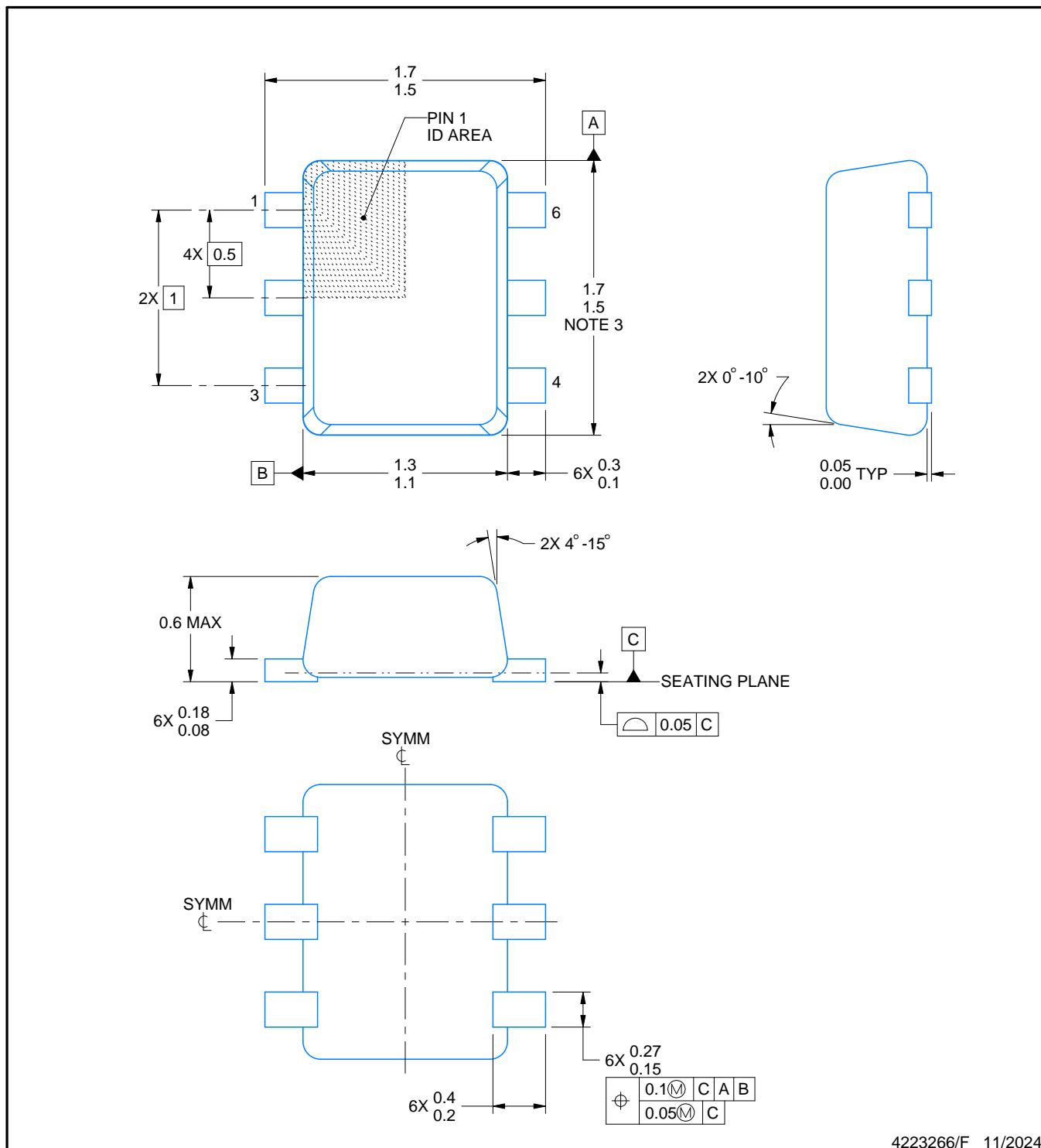
PACKAGE OUTLINE

DRL0006A



SOT - 0.6 mm max height

PLASTIC SMALL OUTLINE



4223266/F 11/2024

NOTES:

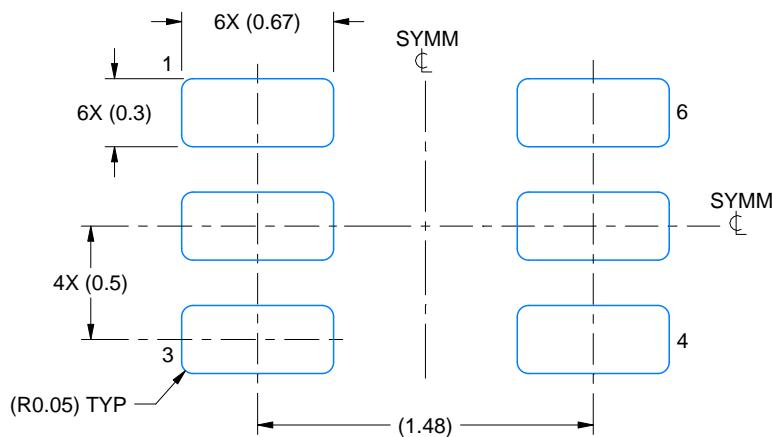
1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
 2. This drawing is subject to change without notice.
 3. This dimension does not include mold flash, protrusions, or gate burrs. Mold flash, protrusions, or gate burrs shall not exceed 0.15 mm per side.
 4. Reference JEDEC registration MO-293 Variation UAAD

EXAMPLE BOARD LAYOUT

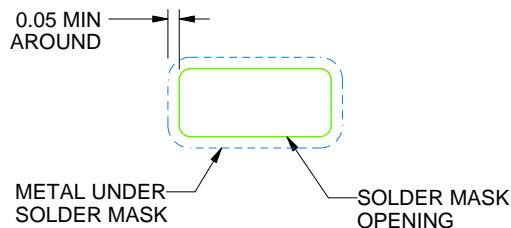
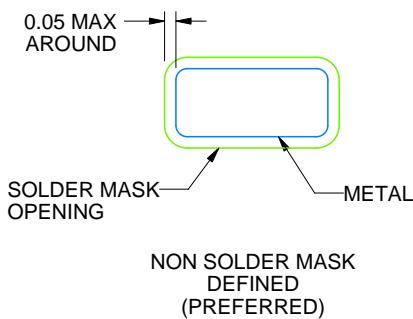
DRL0006A

SOT - 0.6 mm max height

PLASTIC SMALL OUTLINE



LAND PATTERN EXAMPLE
SCALE:30X



SOLDERMASK DETAILS

4223266/F 11/2024

NOTES: (continued)

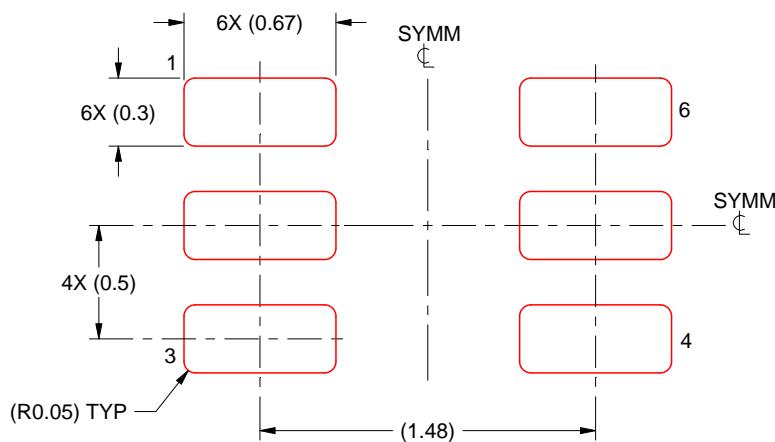
5. Publication IPC-7351 may have alternate designs.
6. Solder mask tolerances between and around signal pads can vary based on board fabrication site.
7. Land pattern design aligns to IPC-610, Bottom Termination Component (BTC) solder joint inspection criteria.

EXAMPLE STENCIL DESIGN

DRL0006A

SOT - 0.6 mm max height

PLASTIC SMALL OUTLINE



SOLDER PASTE EXAMPLE
BASED ON 0.1 mm THICK STENCIL
SCALE:30X

4223266/F 11/2024

NOTES: (continued)

8. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release. IPC-7525 may have alternate design recommendations.
9. Board assembly site may have different recommendations for stencil design.

IMPORTANT NOTICE AND DISCLAIMER

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

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](#) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2025, Texas Instruments Incorporated