ESP32-H2 Series

Datasheet Version 1.2

RISC-V 32-bit single-core microprocessor
Bluetooth® Low Energy and IEEE 802.15.4
3.3 V flash in the chip's package
19 GPIOs
QFN32 (4×4 mm) Package

Including:

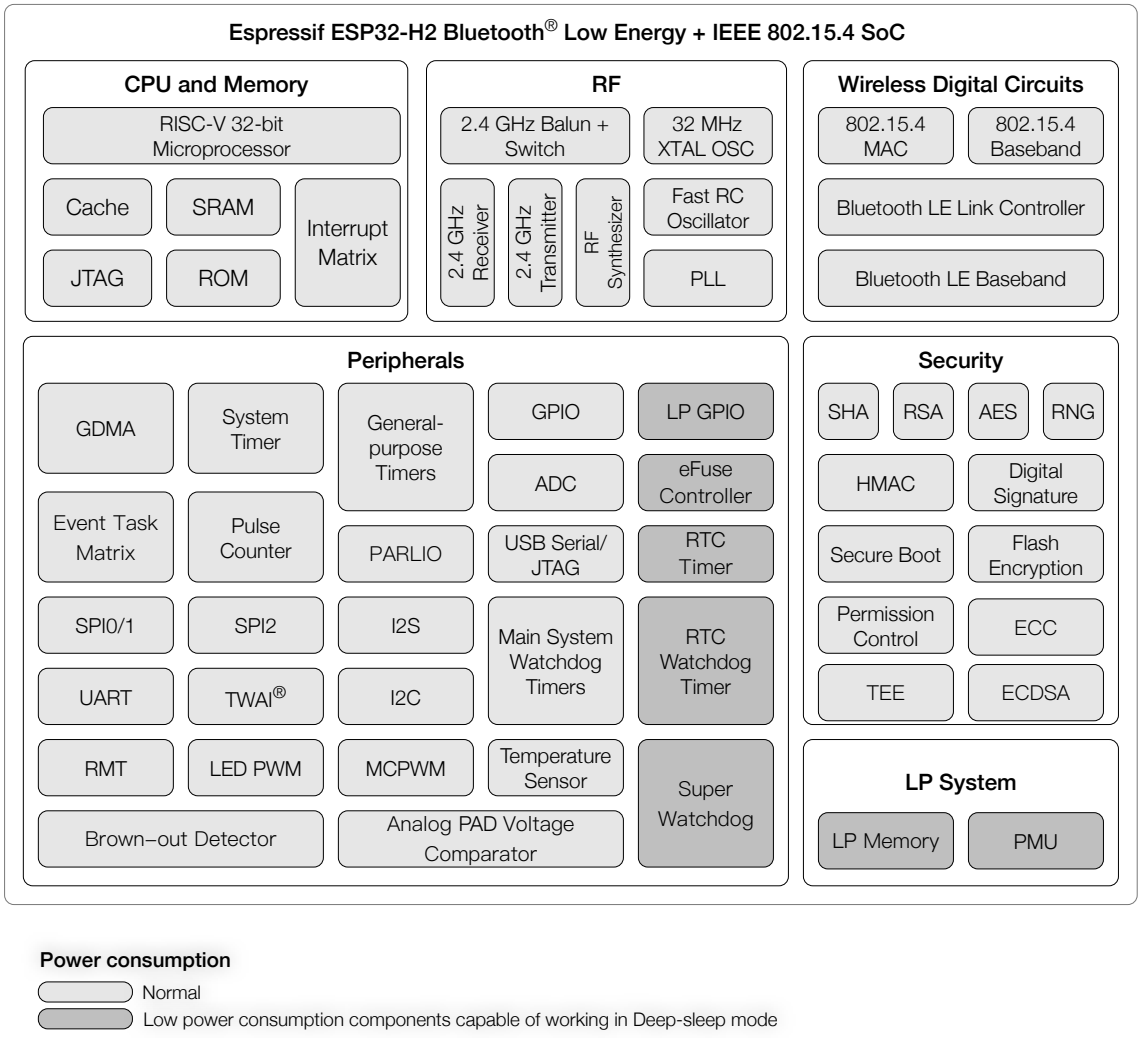
ESP32-H2FH2S

ESP32-H2FH4S

Product Overview

ESP32-H2 is a low-power MCU-based system on a chip (SoC) with integrated 2.4 GHz Bluetooth® Low Energy (Bluetooth LE) and 802.15.4. It consists of an RISC-V 32-bit microprocessor, a Bluetooth LE baseband, an 802.15.4 baseband, RF module, and numerous peripherals.

The functional block diagram of the SoC is shown below.



ESP32-H2 Functional Block Diagram

For more information on power consumption, see Section 4.1.3.6 *Power Management Unit*.

Features

Bluetooth®

- Bluetooth Low Energy (Bluetooth 5.3 certified)
- Bluetooth mesh
- Bluetooth Low Energy long range (Coded PHY, 125 Kbps and 500 Kbps)
- Bluetooth Low Energy high speed (2 Mbps)
- Bluetooth Low Energy advertising extensions and multiple advertising sets
- Simultaneous operation of Broadcaster, Observer, Central, and Peripheral devices
- Multiple connections
- LE power control

802.15.4

- IEEE Standard 802.15.4-2015 compliant
- Supports 250 Kbps data rate in 2.4 GHz band and OQPSK PHY
- Supports Thread
- Supports Zigbee 3.0
- Supports Matter
- Supports other application-layer protocols (HomeKit, MQTT, etc)

CPU and Memory

- 32-bit RISC-V single-core processor
- Clock speed: up to 96 MHz
- CoreMark® score:
  - at 96 MHz: 303.38 CoreMark; 3.16 CoreMark/MHz
- Four-stage pipeline
- 128 KB ROM
- 320 KB SRAM
- 4 KB LP Memory
- 2 MB or 4 MB in-package flash
- 16 KB cache
- Supported SPI protocols: SPI, Dual SPI, Quad SPI, and QPI interfaces that allow connection to flash and other SPI devices
- Flash in-Circuit Programming (ICP)

Advanced Peripheral Interfaces

- 19 programmable GPIOs
  - Three strapping pins: GPIO8, GPIO9, and GPIO25
- Digital interfaces:
  - Two SPI ports for communication with flash
  - General-purpose SPI port
  - Two UART
  - Two I2C
  - I2S
  - RMT, with up to 2 transmit channels and 2 receive channels
  - Pulse count controller
  - LED PWM controller, up to 6 channels
  - USB Serial/JTAG controller
  - Motor Control PWM (MCPWM)
  - General DMA controller, with 3 transmit channels and 3 receive channels
  - TWAI® controller, compatible with ISO 11898-1 (CAN Specification 2.0)
  - SoC event task matrix (ETM)
  - Parallel IO (PARLIO) controller
- Analog interfaces:
  - 12-bit SAR ADC, up to 5 channels
  - Temperature sensor
- Timers:
  - Two 54-bit general-purpose timers
  - 52-bit system timer
  - Three watchdog timers

Power Management

- Fine-resolution power control through a selection of clock frequency, duty cycle, RF 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 μA
- LP memory remains powered on in Deep-sleep mode

Security

- Secure boot - ensuring firmware integrity
- Flash encryption - memory encryption and decryption
- 4096-bit OTP, up to 1792 bits for users
- Cryptographic hardware acceleration:
  - AES-128/256 (FIPS PUB 197)
    - ECB/CBC/CFB/OFB/CTR (FIPS PUB 800-38A)
  - SHA Accelerator (FIPS PUB 180-4)
  - RSA Accelerator
  - ECC Accelerator
  - ECDSA (Elliptic Curve Digital Signature Algorithm)
  - HMAC
  - Digital signature
- Access permission management (APM)
- Random Number Generator (RNG)
- Power Glitch Detector

RF Module

- Antenna switches, RF balun, power amplifier, low-noise receive amplifier
- Up to -106.5 dBm of sensitivity for Bluetooth LE receiver (125 Kbps)
- Up to -102.5 dBm of sensitivity for 802.15.4 receiver (250 Kbps)

Applications

With low power consumption, ESP32-H2 is an ideal choice for IoT devices in the following areas:

- Smart Home
- Industrial Automation
- Health Care
- Consumer Electronics
- Smart Agriculture
- Matter Solutions
- Service Robot
- 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-h2_datasheet_en.pdf

Contents

List of Tables

List of Figures

1 ESP32-H2 Series Comparison

1.1 Nomenclature

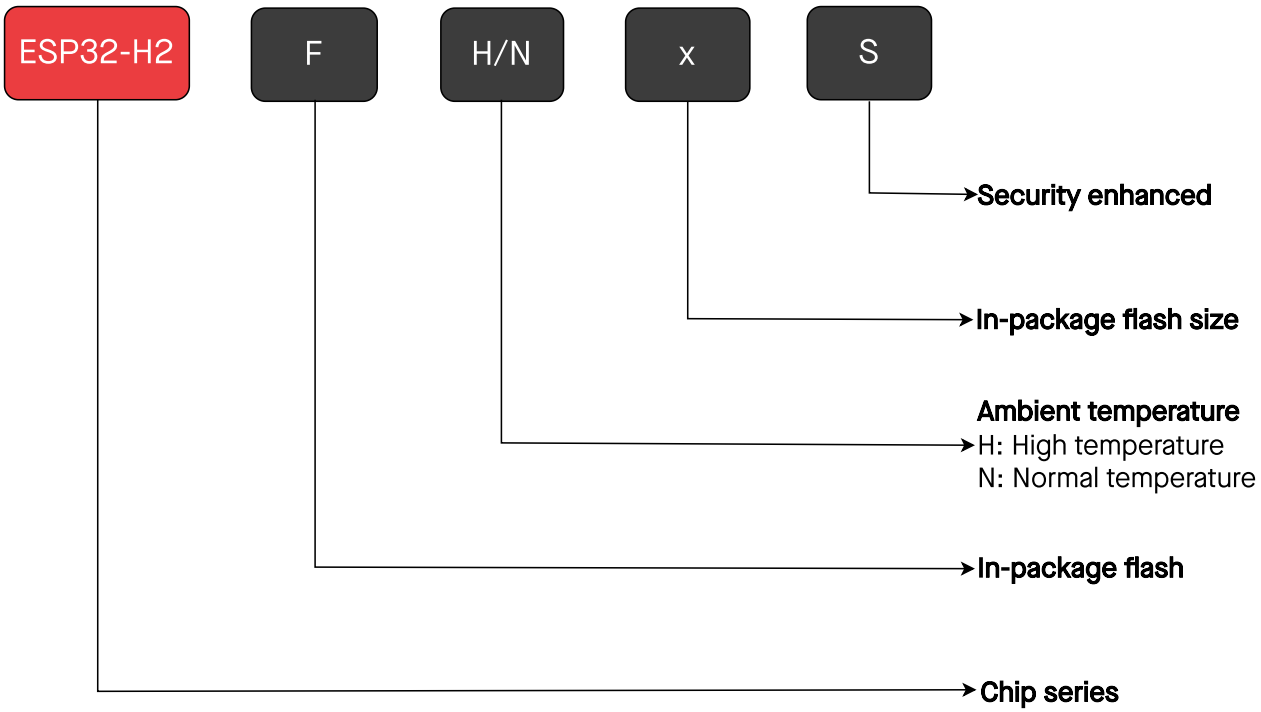


Figure 1-1. ESP32-H2 Series Nomenclature

1.2 Comparison

Table 1-1. ESP32-H2 Series Comparison

Ordering Code [1]	In-Package Flash [2]	Ambient Temp. [3]	SPI Voltage
ESP32-H2FH2S	2 MB (Quad SPI)	–40~105 °C	3.3 V
ESP32-H2FH4S	4 MB (Quad SPI)	–40~105 °C	3.3 V

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

[2] For chip variants with in-package flash (namely variants in QFN32 package), the pins allocated for communication with in-package flash are not routed out.

[3] Ambient temperature specifies the recommended temperature range of the environment immediately outside an Espressif chip.

2 Pins

2.1 Pin Layout

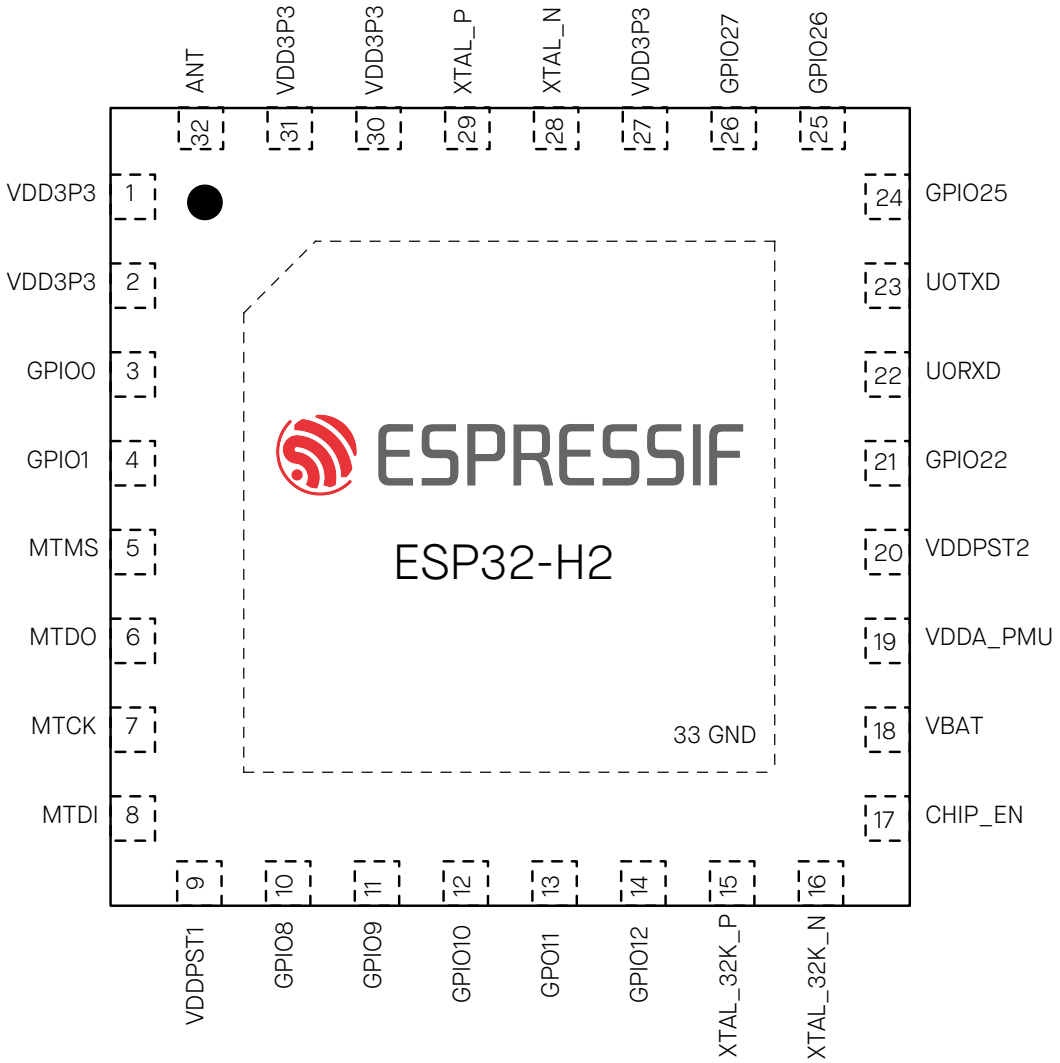


Figure 2-1. ESP32-H2 Pin Layout (Top View)

2.2 Pin Overview

The ESP32-H2 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-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*).

All in all, the ESP32-H2 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 2-3 *IO MUX Pin Functions*
  - **Some** IO pins have predefined **analog functions** – see Table 2-5 *Analog Functions*

  Predefined functions means that each IO pin has a set of direct connections to certain signals from on-chip components. During run-time, the user can configure which component signal from a predefined set to connect to a certain pin at a certain time via memory mapped registers.
- **Analog pins** that have exclusively-dedicated **analog functions** – see Table 2-6 *Analog Pins*
- **Power pins** that supply power to the chip components and non-power pins – see Table 2-7 *Power Pins*

Depending on whether can work in Deep-sleep mode or Light-sleep mode, the pins of ESP32-H2 can also be divided into:

- **Digital pins** (GPIO0 ~ GPIO5, GPIO22 ~ GPIO27): are unable to work in Deep-sleep mode, but can work in Light-sleep mode only if the power domain controlled by the XPD TOP does not power off.
- **LP pins** (GPIO8 ~ GPIO14): are able to work in any chip mode.

Table 2-1 *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-H2 Consolidated Pin Overview.

Table 2-1. Pin Overview

Pin No.	Pin Name	Pin Type	Pin Providing Power [2]	Pin Settings [3]		Pin Function Sets [1]	
				At Reset	After Reset	IO MUX	Analog
1	VDD3P3	Power					
2	VDD3P3	Power					
3	GPIO0	IO	VDDPST1			**IO MUX**	
4	GPIO1	IO	VDDPST1			**IO MUX**	Analog
5	MTMS	IO	VDDPST1	IE	IE	**IO MUX**	Analog
6	MTDO	IO	VDDPST1	IE	IE	**IO MUX**	Analog
7	MTCK	IO	VDDPST1		IE [4]	**IO MUX**	Analog
8	MTDI	IO	VDDPST1		IE	**IO MUX**	Analog
9	VDDPST1	Power					
10	GPIO8	IO	VDDPST1	IE	IE	**IO MUX**	
11	GPIO9	IO	VDDPST1	IE, WPU	IE, WPU	**IO MUX**	
12	GPIO10	IO	VDDPST1			**IO MUX**	Analog
13	GPIO11	IO	VDDPST1			**IO MUX**	Analog
14	GPIO12	IO	VDDA_PMU/VBAT			**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]	Pin Settings [3]		Pin Function Sets [1]	
				At Reset	After Reset	IO MUX	Analog
15	XTAL_32K_P	IO	VDDA_PMU/VBAT			**IO MUX**	Analog
16	XTAL_32K_N	IO	VDDA_PMU/VBAT			**IO MUX**	Analog
17	CHIP_EN	Analog	VBAT				
18	VBAT	Power					
19	VDDA_PMU	Power					
20	VDDPST2	Power					
21	GPIO22	IO	VDDPST2			**IO MUX**	
22	U0RXD	IO	VDDPST2		IE, WPU	**IO MUX**	
23	U0TXD	IO	VDDPST2		IE, WPU	**IO MUX**	
24	GPIO25	IO	VDDPST2	IE	IE	**IO MUX**	
25	GPIO26	IO	VDDPST2		IE	IO MUX	**Analog**
26	GPIO27	IO	VDDPST2		IE, USB_PU	IO MUX	**Analog**
27	VDD3P3	Power					
28	XTAL_N	Analog					
29	XTAL_P	Analog					
30	VDD3P3	Power					
31	VDD3P3	Power					
32	ANT	Analog					
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. Default drive strength for GPIO26 and GPIO27 is 40 mA, and 20 mA for the other GPIOs.
3. 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., GPIO26 and GPIO27), and the pin pull-up is decided by the USB pull-up resistor. This resistor is controlled by USB_SERIAL_JTAG_DP/DM_PULLUP, and the pull-up value is managed by USB_SERIAL_JTAG_PULLUP_VALUE. For details, see *ESP32-H2 Technical Reference Manual* > Chapter *USB Serial/JTAG Controller*.
     - When the USB function is disabled, USB pins are used as regular GPIOs and the pin's internal weak pull-up and pull-down resistors are disabled by default (configurable by IO_MUX_GPIO*n*_FUN_WPU/WPD). For details, see *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix (GPIO, IO MUX)*).
4. Depends on the value of EFUSE_DIS_PAD_JTAG
   - `0` - WPU is enabled
   - `1` - pin floating

2.3 IO Pins

2.3.1 IO MUX Functions

The IO MUX allows multiple input/output signals to be connected to a single input/output pin. Each IO pin of ESP32-H2 can be connected to one of the five signals (IO MUX functions, i.e., F0-F4), as listed in Table 2-3 *IO MUX Pin Functions*.

Among the five 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-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.
- Some are directly routed from certain peripherals (**U0TXD, MTCK, etc.**), including UART0, JTAG, and SPI2 - see Table 2-2 *Peripheral Signals Routed via IO MUX*.

Table 2-2. Peripheral Signals Routed via IO MUX

Pin Function	Signal	Description
U0TXD U0RXD	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
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

Table 2-3 *IO MUX Pin Functions* shows the IO MUX functions of IO pins.

Table 2-3. IO MUX Pin Functions

Pin No.	IO MUX / GPIO Name [2]	IO MUX Function [1, 2, 3]									
		F0	Type [3]	F1	Type	F2	Type	F3	Type	F4	Type
3	GPIO0	**GPIO0**	I/O/T	GPIO0	I/O/T	FSPIQ	I1/O/T				
4	GPIO1	**GPIO1**	I/O/T	GPIO1	I/O/T	FSPICS0	I1/O/T				
5	GPIO2	**MTMS**	I1	GPIO2	I/O/T	FSPIWP	I1/O/T				
6	GPIO3	**MTDO**	O/T	GPIO3	I/O/T	FSPIHD	I1/O/T				
7	GPIO4	**MTCK**	I1	GPIO4	I/O/T	FSPICLK	I1/O/T				
8	GPIO5	**MTDI**	I1	GPIO5	I/O/T	FSPID	I1/O/T				
10	GPIO8	**GPIO8**	I/O/T	GPIO8	I/O/T						

Cont'd on next page

Table 2-3 – cont'd from previous page

Pin No.	IO MUX / GPIO Name [2]	IO MUX Function [1, 2, 3]									
		F0	Type [3]	F1	Type	F2	Type	F3	Type	F4	Type
11	GPIO9	**GPIO9**	I/O/T	GPIO9	I/O/T						
12	GPIO10	**GPIO10**	I/O/T	GPIO10	I/O/T						
13	GPIO11	**GPIO11**	I/O/T	GPIO11	I/O/T						
14	GPIO12	**GPIO12**	I/O/T	GPIO12	I/O/T						
15	GPIO13	**GPIO13**	I/O/T	GPIO13	I/O/T						
16	GPIO14	**GPIO14**	I/O/T	GPIO14	I/O/T						
21	GPIO22	**GPIO22**	I/O/T	GPIO22	I/O/T						
22	GPIO23	**U0RXD**	I1	GPIO23	I/O/T	FSPICS1	O/T				
23	GPIO24	**U0TXD**	O	GPIO24	I/O/T	FSPICS2	O/T				
24	GPIO25	**GPIO25**	I/O/T	GPIO25	I/O/T	FSPICS3	O/T				
25	GPIO26	GPIO26	I/O/T	GPIO26	I/O/T	FSPICS4	O/T				
26	GPIO27	GPIO27	I/O/T	GPIO27	I/O/T	FSPICS5	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 2.3.3 *Restrictions for GPIOs*.

[3] Each IO MUX function (F*n*, *n* = 0 ~ 4) 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`.
- I0 – input; if the pin is assigned a function other than F*n*, the input signal of F*n* is always `0`.

2.3.2 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 2-4 *Analog Signals Routed to Analog Functions*.

Table 2-4. Analog Signals Routed to Analog Functions

Pin Function	Signal	Description
ADC1_CH*n*	ADC1 channel *n* signal	ADC1 channel *n* interface
XTAL_32K_N XTAL_32K_P	Negative clock signal Positive clock signal	32 kHz external clock input/output connected to ESP32-H2's oscillator/crystal
USB_D- USB_D+	Data - (negative USB signal) Data + (positive USB signal)	USB signals from USB Serial/JTAG Controller
ZCD*n*	Voltage from GPIO Pad	Analog Pad voltage comparator interface

Table 2-5 *Analog Functions* shows the analog functions of IO pins.

Table 2-5. Analog Functions

Pin No.	Analog IO Name [1]	Analog Function [2] F0	F1
4	GPIO1		ADC1_CH0
5	GPIO2		ADC1_CH1
6	GPIO3		ADC1_CH2
7	GPIO4		ADC1_CH3
8	GPIO5		ADC1_CH4
12	GPIO10	ZCD0	
13	GPIO11	ZCD1	
15	XTAL_32K_P	XTAL_32K_P	
16	XTAL_32K_N	XTAL_32K_N	
25	GPIO26	**USB_D-**	
26	GPIO27	**USB_D+**	

[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 2.3.3 *Restrictions for GPIOs*.

2.3.3 Restrictions for GPIOs

All IO pins of ESP32-H2 have 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 chapter, some pin functions are highlighted. The non-highlighted GPIO pins are recommended for use first. If more pins are needed, the highlighted GPIOs should be chosen carefully to avoid conflicts with important pin functions.

The highlighted IO pins have the following important pin functions:

- **Strapping pins** – need to be at certain logic levels at startup. See Section 3 *Boot Configurations*.
- **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 2-2 *Peripheral Signals Routed via IO MUX*. To free these pins up, the pin functions USB_D+/- of the USB Serial/JTAG Controller can be used instead. See also Section 3.3 *JTAG Signal Source Control*.
- **UART interface** – often used for debugging. See Table 2-2 *Peripheral Signals Routed via IO MUX*.

See also Appendix A – ESP32-H2 Consolidated Pin Overview.

2.4 Analog Pins

Table 2-6. Analog Pins

Pin No.	Pin Name	Pin Type	Pin Function
17	CHIP_EN	I	High: on, enables the chip (powered up). Low: off, disables the chip (powered down). Note: Do not leave the CHIP_EN pin floating.
28	XTAL_N	—	External clock input/output connected to chip's crystal or oscillator. P/N means differential clock positive/negative.
29	XTAL_P	—	
32	ANT	I/O	RF LNA input/output signals

2.5 Power Supply

2.5.1 Power Pins

The chip is powered via the power pins described in Table 2-7 *Power Pins*.

Table 2-7. Power Pins

Pin No.	Pin Name	Direction	Power Supply [1,2]	
			Power Domain/Other	IO Pins
1	VDD3P3	Input	Analog power domain	
2	VDD3P3	Input	Analog power domain	
9	VDDPST1	Input	IO power domain	Digital IO, LP IO [3]
18	VBAT	Input	Analog power domain or battery power supply	GPIO12, XTAL_32K_P, XTAL_32K_N
19	VDDA_PMU	Input	Analog power domain	GPIO12, XTAL_32K_P, XTAL_32K_N
20	VDDPST2	Input	IO power domain	Digital IO
27	VDD3P3	Input	Analog power domain	
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] For the classification of digital IO and LP IO, see Section 2.2 *Pin Overview*.

2.5.2 Power Scheme

The power scheme is shown in Figure 2-2 *ESP32-H2 Power Scheme*.

The components on the chip are powered via voltage regulators.

Table 2-8. Voltage Regulators

Voltage Regulator	Output	Power Supply
Digital	1.1 V	Digital power domain
Low-power	1.1 V	LP power domain

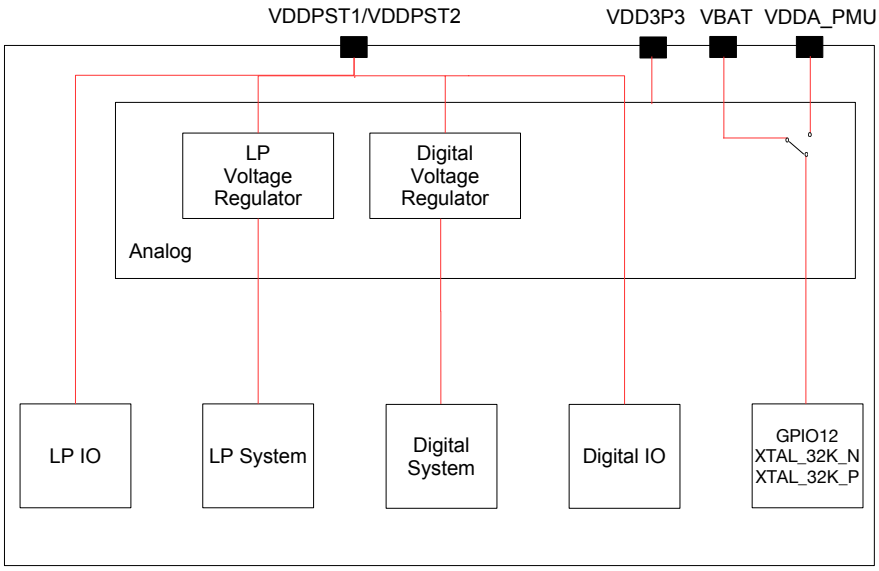


Figure 2-2. ESP32-H2 Power Scheme

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_EN – the pin used for power-up and reset – is pulled high to activate the chip. For information on CHIP_EN as well as power-up and reset timing, see Figure 2-3 and Table 2-9.

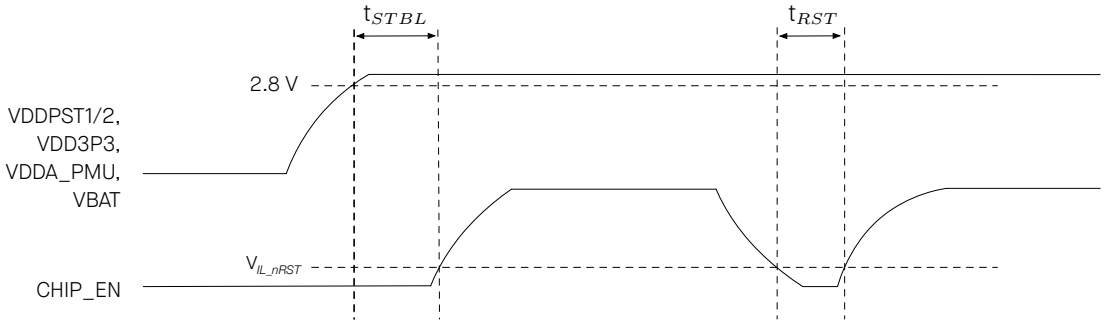


Figure 2-3. Visualization of Timing Parameters for Power-up and Reset

Table 2-9. Description of Timing Parameters for Power-up and Reset

Parameter	Description	Min (μs)
t_{STBL}	Time reserved for the power rails of VDDPST1, VDDPST2, VDD3P3, VDDA_PMU, and VBAT to stabilize before the CHIP_EN pin is pulled high to activate the chip	50
t_{RST}	Time reserved for CHIP_EN to stay below V_{IL_nRST} to reset the chip (see Table 5-3)	50

3 Boot Configurations

The chip allows for configuring the following boot parameters through strapping pins, eFuse bits, and registers at power-up or a hardware reset, without microcontroller interaction.

- **Chip boot mode**
  - Strapping pin: GPIO8 and GPIO9
- **ROM message printing**
  - Strapping pin: GPIO8
  - eFuse bits: EFUSE_UART_PRINT_CONTROL and EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT
  - Register: LP_AON_STORE4_REG[0]
- **JTAG signal source**
  - Strapping pin: GPIO25
  - eFuse bits: EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG, and EFUSE_JTAG_SEL_ENABLE

The default values of all the above eFuse bits are 0, which means that they are not burnt. Given that eFuse is one-time programmable, once an eFuse bit is programmed to 1, it can never be reverted to 0. For how to program eFuse bits, please refer to *ESP32-H2 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
GPIO8	Floating	—
GPIO9	Weak pull-up	1
GPIO25	Floating	—

To change the bit values, the strapping pins should be connected to external pull-down/pull-up resistances. If the ESP32-H2 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 system 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_EN 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_EN is already high and before these pins start operating as regular IO pins.	3

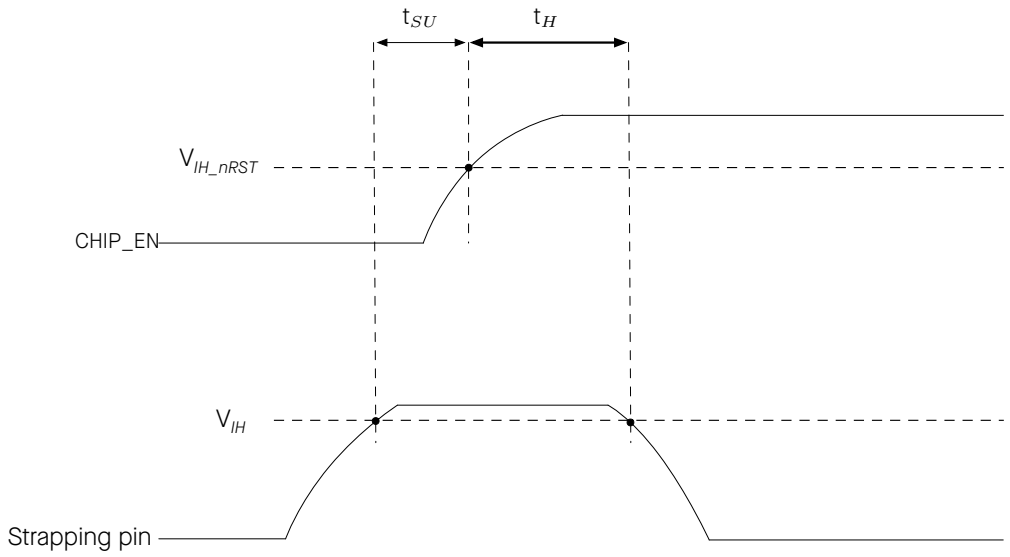


Figure 3-1. Visualization of Timing Parameters for the Strapping Pins

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 [1]	GPIO8	GPIO9
SPI Boot	Any value	**1**
Joint Download Boot [2]	1	0

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

[2] Joint Download Boot mode supports the following download methods:

- USB Download Boot:
  - USB-Serial-JTAG Download Boot
- UART Download Boot

3.2 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) UART0 and USB Serial/JTAG controller**
- USB Serial/JTAG controller
- UART0

EFUSE_UART_PRINT_CONTROL, LP_AON_STORE4_REG[0], and GPIO8 control ROM messages printing to **UART0** as shown in Table 3-4 *UART0 ROM Message Printing Control*.

Table 3-4. UART0 ROM Message Printing Control

UART0 ROM Message Printing [1]	LP_AON_STORE4_REG[0]	EFUSE_UART_PRINT_CONTROL	GPIO8
Enabled	**0**	**0**	Ignored
		1	0
		2	1
Disabled	0	1	1
		2	0
		3	Ignored
	1	Ignored	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 3-5 *USB Serial/JTAG ROM Message Printing Control*.

Table 3-5. USB Serial/JTAG ROM Message Printing Control

USB Serial/JTAG ROM Message Printing Control [1]	LP_AON_STORE4_REG[0]	EFUSE_DIS_USB_SERIAL_JTAG_ROM_PRINT
Enabled	**0**	**0**
Disabled	0	1
	1	Ignored

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

3.3 JTAG Signal Source Control

The strapping pin GPIO25 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 3-6 shows GPIO25 is used in combination with EFUSE_DIS_PAD_JTAG, EFUSE_DIS_USB_JTAG, and EFUSE_JTAG_SEL_ENABLE.

Table 3-6. JTAG Signal Source Control

JTAG Signal Source [1]	EFUSE_DIS_PAD_JTAG	EFUSE_DIS_USB_JTAG	EFUSE_STRAP_JTAG_SEL_ENABLE	GPIO25
USB Serial/JTAG Controller	**0**	**0**	**0**	Ignored
JTAG pins [2]			1	0
USB Serial/JTAG Controller				1
JTAG pins [2]	0	1	Ignored	Ignored
USB Serial/JTAG Controller	1	0	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 ESP-RISC-V CPU

The ESP-RISC-V CPU is a 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 of up to 96 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 machine mode and user 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*
- Debugger with a direct system bus access (SBA) to memory and peripherals
- 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
- 32-bit AHB system bus for peripheral access
- Configurable events for core performance metrics

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *ESP-RISC-V CPU*.

4.1.1.2 RISC-V Trace Encoder

The RISC-V Trace Encoder in the ESP32-H2 chip provides a way to capture detailed trace information from the CPU's execution, enabling deeper analysis and optimization of the system. It connects to the 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
- Arbitrary address range of the trace memory size
- Two synchronization modes:
  - synchronization counter counts by packet
  - synchronization counter counts by cycle
- Trace lost status to indicate packet loss
- Automatic restart after packet loss
- Configurable memory writing mode: loop mode or non-loop mode
- FIFO (128 × 8 bits) to buffer packets

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *RISC-V Trace Encoder (TRACE)*.

4.1.1.3 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 without 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

- AHB bus architecture
- 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 324 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-H2 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-H2.

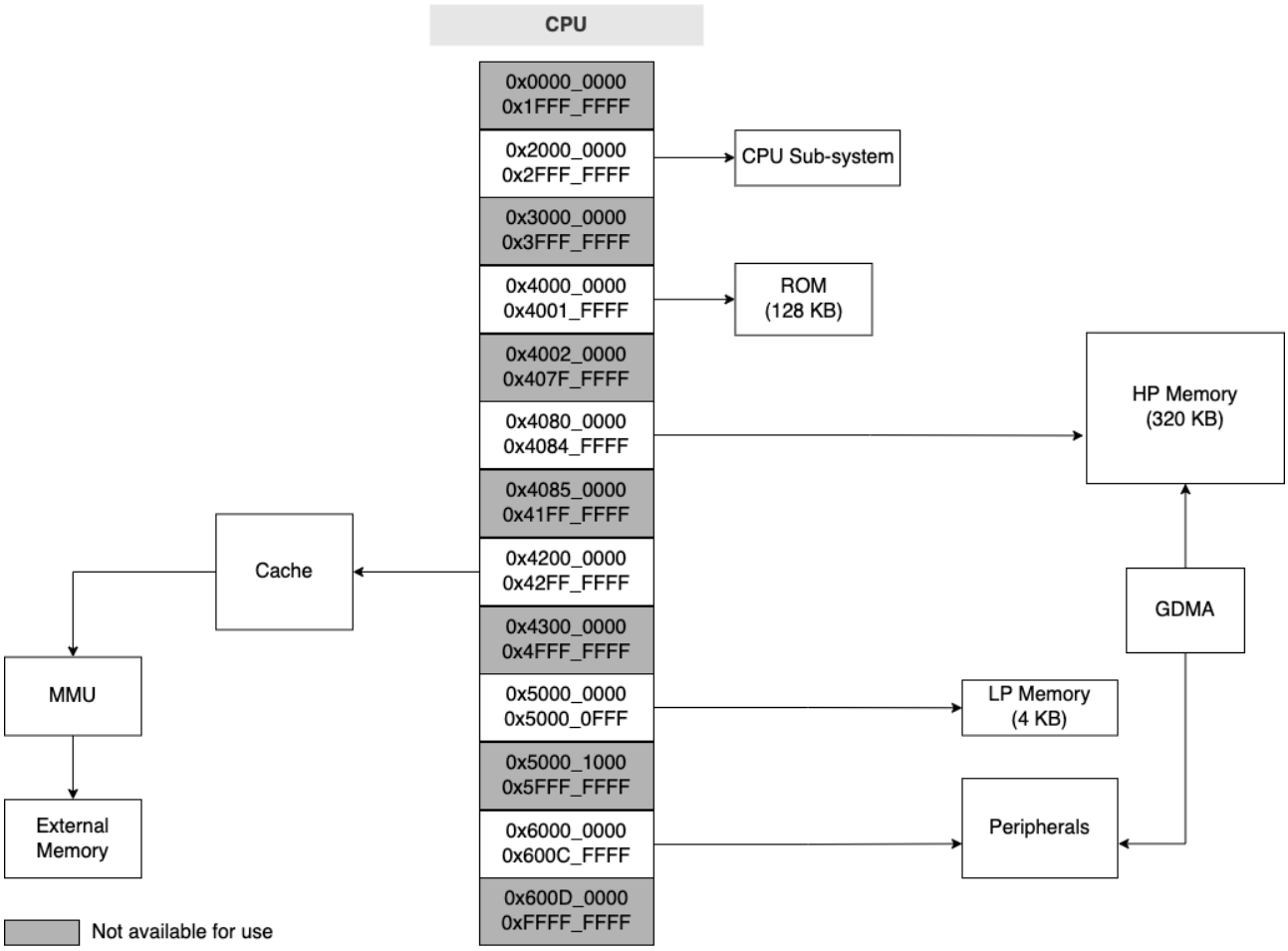


Figure 4-1. Address Mapping Structure

4.1.2.1 Internal Memory

The internal memory of ESP32-H2 refers to the memory integrated on the chip die or in the chip package, including ROM, SRAM, eFuse, and flash.

Feature List

- 128 KB of ROM for booting and core functions
- 320 KB of high-performance SRAM (HP SRAM) for data and instructions
- 4 KB of low-power SRAM (LP SRAM) that can be accessed by 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*
- In-package flash
  - Flash size

    * ESP32-H2FH2S chip variant: 2 MB
    * ESP32-H2FH4S chip variant: 4 MB
  - More than 100,000 program/erase cycles
  - More than 20 years of data retention time
  - Clock frequency:
    * ESP32-H2FH2S chip variant: up to 64 MHz
    * ESP32-H2FH4S chip variant: up to 64 MHz

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *System and Memory*.

4.1.2.2 External Memory

ESP32-H2 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 16 KB read-only cache
  - Eight-way set associative
  - 32-byte cache block
  - Critical word first and early restart

For details, see *ESP32-H2 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. The eFuse controller in the ESP32-H2 is responsible for programming and reading this memory.

Feature List

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

For details, see *ESP32-H2 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-H2 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

- 19 GPIO pins for general-purpose I/O or connection to internal peripheral signals
- GPIO matrix:
  - Routing 78 peripheral input and 99 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
  - GPIO simple input and output
- IO MUX for directly connecting certain digital signals (SPI, JTAG, UART) to pins
- Support for Event Task Matrix

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.1.3.2 Reset

The ESP32-H2 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-H2 Technical Reference Manual* > Chapter *Reset and Clock*.

4.1.3.3 Clock

The ESP32-H2 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 and slow speed clocks.

Feature List

- High speed clocks (used for devices working at higher frequencies)
  - 32 MHz external crystal clock

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

  - 96 MHz internal PLL clock
  - 64 MHz internal PLL clock
- Slow speed clocks (used for LP system and some peripherals working in low-power mode)
  - 32 kHz external crystal clock
  - Internal fast RC oscillator with adjustable frequency (8 MHz by default)
  - 130 kHz internal slow RC oscillator
  - Internal PLL clock
  - External slow clock input through XTAL_32K_P (32 kHz by default)

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Reset and Clock*.

4.1.3.4 Interrupt Matrix

The Interrupt Matrix in the ESP32-H2 chip routes interrupt requests generated by various peripherals and events to CPU interrupts.

Feature List

- 65 peripheral interrupt sources accepted as input
- 28 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-H2 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, GDMA, and PMU.

Feature List

- 50 channels that can be enabled and configured independently
- Receive 122 events from multiple peripherals
- Generate 129 tasks for multiple peripherals

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Event Task Matrix*.

4.1.3.6 Power Management Unit

The ESP32-H2 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.

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 CPU, RF circuits, and all peripherals are on. The chip can process data, receive, transmit, and listen.
- **Modem-sleep mode** – The 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 CPU stops running, and can be optionally powered on. The LP peripherals can be woken up periodically by the timer. The chip can be woken up via all wake up mechanisms: Modem, RTC timer, or external interrupts. Wireless connections can remain active. Some groups of digital peripherals can be optionally powered off.
- **Deep-sleep mode** – Only LP system is powered on. Wireless connection data is stored in LP memory.

For power consumption in different power modes, see Section 5.5 *Current Consumption*.

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Low-Power Management*.

4.1.3.7 System Timer

The System Timer (SYSTIMER) in the ESP32-H2 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-H2 Technical Reference Manual* > Chapter *System Timer*.

4.1.3.8 Timer Groups

The Timer Group (TIMG) in the ESP32-H2 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-H2 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
- Level interrupt generation
- Support for several ETM tasks and events

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Timer Group (TIMG)*.

4.1.3.9 Watchdog Timers

The Watchdog Timers (WDT) in ESP32-H2 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-H2 Technical Reference Manual* > Chapter *Watchdog Timers*.

4.1.3.10 Permission Control

The Permission Control module in ESP32-H2 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-H2 Technical Reference Manual* > Chapter *Permission Control (PMS)*.

4.1.3.11 System Registers

The System Registers in the ESP32-H2 chip are used to configure various auxiliary chip features.

Feature List

- Control external memory encryption and decryption
- Control Bus timeout protection

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *System Registers*.

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 CPU bus reads from or writes to a specified memory address space
- Stack pointer (SP) monitoring: Monitor whether the stack pointer is out of a limited range (overflows), and generates an interrupt if overflow occurs. violation will trigger an interrupt.

- Program counter (PC) logging: Record PC value. The developer can get the last PC value at the most recent CPU reset
- Bus access logging: Record information about bus access when the CPU or DMA writes a specified value

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Debug Assistant (ASSIST_DEBUG)*.

4.1.4 Cryptography and Security Components

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

4.1.4.1 AES Accelerator

ESP32-H2 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-H2 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
- Anti-attack pseudo-round function, to enhance the chip's anti-attack performance

For details, see *ESP32-H2 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)
- 11 working modes that support Base Point Verification, Base Point Multiplication, Jacobian Point Verification, Jacobian Point Multiplication, and mod operations
- High anti-attack performance. Each point multiplication calculation of the ECC accelerator consumes:
  - the same amount of time
  - the same amount of power

For details, see *ESP32-H2 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
- Compatibility with challenge-response authentication algorithm
- Generates required keys for the Digital Signature Algorithm (DSA) peripheral in downstream mode
- Re-enables soft-disabled JTAG in downstream mode
- Hash result only accessible by configurable hardware peripheral (in downstream mode)

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *HMAC Accelerator* (HMAC).

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 *ESP32-H2 Technical Reference Manual* > Chapter *RSA Accelerator (RSA)*.

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 details, see *ESP32-H2 Technical Reference Manual* > Chapter *SHA Accelerator (SHA)*.

4.1.4.6 Digital Signature

The Digital Signature (DS) module in the ESP32-H2 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 details, see *ESP32-H2 Technical Reference Manual* > Chapter *Digital Signature (DS)*.

4.1.4.7 Elliptic Curve Digital Signature Algorithm (ECDSA)

In cryptography, the Elliptic Curve Digital Signature Algorithm (ECDSA) offers a variant of the Digital Signature Algorithm (DSA) which uses elliptic-curve cryptography. ESP32-H2's ECDSA accelerator provides a secure and efficient environment for computing ECDSA signatures. It offers fast computations while ensuring the confidentiality of the signing process to prevent information leakage.

Feature List

- Digital signature generation and verification
- Two different elliptic curves, namely P-192 and P-256
- Dynamic access permission in different operation statuses to ensure information security
- High anti-attack performance. Each time a signature is generated and verified, ECDSA consumes:
  - the same amount of time
  - the same amount of power

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Elliptic Curve Digital Signature Algorithm (ECDSA)*.

4.1.4.8 External Memory Encryption and Decryption

The External Memory Encryption and Decryption (XTS_AES) module in the ESP32-H2 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
- Pseudo-round anti-DPA function

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *External Memory Encryption and Decryption (XTS_AES)*.

4.1.4.9 Random Number Generator

The Random Number Generator (RNG) in the ESP32-H2 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-H2 Technical Reference Manual* > Chapter *Random Number Generator (RNG)*.

4.1.4.10 Power Glitch Detector

The ESP32-H2 chip integrates a power glitch detector that monitors the voltage of power supply pins, including VDDPST1, VDDPST2, and VDD3P3 (PIN 27), in real time. It can detect voltage abnormalities occurring at these pins. For example, when a sudden voltage drop or voltage glitch is detected, the chip triggers a power glitch reset to ensure the system safely returns to a controllable state. This prevents logic errors, data loss, or hardware damage caused by power glitch.

Features

- Real-time monitoring of the voltage on specific power pins
- Ability to trigger a power glitch reset to prevent power glitch attacks
- Enabled by default at power-up

For more details about the Power Glitch Detector, refer to *ESP32-H2 Technical Reference Manual* > Chapter *Power Supply Detector*.

4.1.4.11 Secure Boot

The Secure boot feature in the ESP32-H2 chip ensures that only the signed firmware can be booted.

Feature List

- Supported signature type
  - RSA-RSS signature
  - ECDSA signature

For details, see ESP-IDF Programming Guide > Secure Boot v2.

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 Interfaces

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-H2 chip facilitates the transmission and reception of asynchronous serial data between the chip and external UART devices. It consists of two UARTs in the system.

Feature List

- Programmable baud rates up to 5 MBaud
- 260 x 8 bit 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
- IrDA protocol support
- High-speed data communication using GDMA
- Receive timeout feature
- UART as the wake-up source
- Software and hardware flow control

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *UART Controller (UART)*.

Pin Assignment

The pins connected to receive and transmit signals (U0RXD and U0TXD) for **UART0** are multiplexed with GPIO23 ~ GPIO24 and FSPICS1 ~ FSPICS2 via IO MUX. Other signals can be routed to any GPIOs via the GPIO matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.2 SPI Controller

ESP32-H2 has the following SPI interfaces:

- **SPI0/SPI1** are reserved for system use.
- **SPI2** is a general-purpose SPI (GP-SPI) controller with access to general-purpose DMA channels.

Features of SPI2

- Supports operation as a master or slave
- Support for DMA
- 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 48 MHz
  - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 48 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 32 MHz
  - Supports 1-, 2-, 4-line half-duplex communication with clock frequency up to 32 MHz

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *SPI Controller (SPI)*.

Pin Assignment

- Via IO MUX
  For SPI2, the pins for data and clock signals are multiplexed with GPIO0, GPIO2 ~ GPIO5, and JTAG interface via the IO MUX. The pins for chip select signals are multiplexed with GPIO1, GPIO23 ~ GPIO27, UART0 interface, and USB interface via the IO MUX.
- Via GPIO Matrix
  The pins for SPI2 can be chosen from any GPIOs via the GPIO matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

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
- Communication with multiple external devices
- Master and slave modes

- 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-H2 Technical Reference Manual* > Chapter *I2C Controller (I2C)*.

Pin Assignment

The pins used for I2C can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.4 I2S Controller

The I2S Controller in the ESP32-H2 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, 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-H2 Technical Reference Manual* > Chapter *I2S Controller (I2S)*.

Pin Assignment

The pins for the I2S Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.5 Pulse Count Controller

The Pulse Count Controller (PCNT) is designed to count input pulses by tracking the 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-H2 Technical Reference Manual* > Chapter *Pulse Count Controller (PCNT)*.

Pin Assignment

The pins for the Pulse Count Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.6 USB Serial/JTAG Controller

The USB Serial/JTAG controller in the ESP32-H2 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
- Internal PHY

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *USB Serial/JTAG Controller (USB_SERIAL_JTAG)*.

Pin Assignment

The pins USB_D+ and USB_D- for the USB Serial/JTAG Controller are multiplexed with GPIO26 ~ GPIO27 and FSPICS4 ~ FSPICS5 via IO MUX.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

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-H2 Technical Reference Manual* > Chapter *Two-wire Automotive Interface*.

Pin Assignment

The pins for the Two-wire Automotive Interface can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.8 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-H2 Technical Reference Manual* > Chapter *LED PWM Controller*.

Pin Assignment

The pins for the LED PWM Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.9 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
  - The control of the PWM signal can be updated asynchronously
  - 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
  - Period, time stamps, and important control registers have shadow registers with flexible updating methods
- 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-H2 Technical Reference Manual* > Chapter *Motor Control PWM (MCPWM)*.

Pin Assignment

The pins for the Motor Control PWM can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.10 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 transmitter for each TX channel
- 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 details, see *ESP32-H2 Technical Reference Manual* > Chapter *Remote Control Peripheral (RMT)*.

Pin Assignment

The pins for the Remote Control Peripheral can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.1.11 Parallel IO Controller

The Parallel IO Controller (PARLIO) in the ESP32-H2 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-bit configurable data bus width
- Full-duplex communication with 8-bit data bus width
- Bit reordering in 1/2/4-bit data bus width mode
- RX unit supports eight 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 TXD

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *Parallel IO Controller*.

Pin Assignment

The pins for the Parallel IO Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

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-H2 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 five 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 details, see *ESP32-H2 Technical Reference Manual* > Chapter *SAR ADC and Temperature Sensor*.

Pin Assignment

The pins for the SAR ADC are multiplexed with GPIO1 ~ GPIO5, JTAG interface, and SPI2 interface.

For more information about the pin assignment, see Section 2.3 *IO Pins* and *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.2.2.2 Temperature Sensor

The Temperature Sensor in the ESP32-H2 chip allows for real-time monitoring of temperature changes inside the chip.

Feature List

- Measurement range: –40°C ~ 125°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 details, see *ESP32-H2 Technical Reference Manual* > Chapter *SAR ADC and Temperature Sensor*.

4.2.2.3 Analog PAD Voltage Comparator

ESP32-H2 integrates two analog voltage comparators. These comparators rely on special pads that support voltage comparison functionality to monitor voltage changes on these pads. Each analog voltage comparator has two pads associated with it, for the main voltage and the reference voltage respectively. The voltage comparison result generated by the analog voltage comparator can be used as Event Task Matrix (ETM) events to drive ETM tasks of other peripherals or trigger interrupts.

Feature List

- Voltage comparison
  - Configurable voltage comparison mode
  - Configurable reference voltage
- Interrupt upon changes of voltage comparison result

- ETM event generation

For details, see *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix* > *Function of Analog PAD Voltage Comparator*.

Pin Assignment

The pins for the analog voltage pad comparators are multiplexed with GPIO10 ~ GPIO11.

For more details, see *ESP32-H2 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*.

4.3 Wireless Communication

This section describes the chip's wireless communication capabilities, spanning radio technology, Bluetooth Low Energy, 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.

4.3.1.1 2.4 GHz Receiver

The 2.4 GHz receiver demodulates the 2.4 GHz RF signal to baseband signals and converts them to the digital domain with two high-resolution ADCs. To adapt to varying signal channel conditions, ESP32-H2 integrates RF filters, Automatic Gain Control (AGC), DC offset cancellation circuits, and baseband filters.

4.3.1.2 2.4 GHz Transmitter

The 2.4 GHz transmitter modulates the baseband signals to the 2.4 GHz RF signal, and drives the antenna with a CMOS power amplifier.

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

- Carrier leakage
- I/Q amplitude/phase 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 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. 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 Bluetooth LE

ESP32-H2 includes a Bluetooth Low Energy subsystem that integrates a link controller, an RF/modem block and a feature-rich software protocol stack. It supports the core features of Bluetooth 5 and Bluetooth mesh.

4.3.2.1 Bluetooth LE PHY

ESP32-H2's Bluetooth Low Energy PHY 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)

4.3.2.2 Bluetooth LE Link Controller

ESP32-H2's Bluetooth Low Energy Link Controller 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
- 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 connectable directed advertising
- Link layer encryption
- LE Ping

4.3.3 802.15.4

ESP32-H2 includes an IEEE Standard 802.15.4 subsystem that integrates PHY and MAC layers. It supports various software stacks including Thread, Zigbee, Matter, HomeKit, MQTT, and so on.

4.3.3.1 802.15.4 PHY

ESP32-H2's 802.15.4 PHY supports:

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

4.3.3.2 802.15.4 MAC

ESP32-H2 supports most key features defined in IEEE Standard 802.15.4-2015, includes:

- 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	—	1.3	A
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 Operating Conditions

Parameter [1]	Description	Min	Typ	Max	Unit
VDD3P3, VBAT, VDDA_PMU, VDDPST1, VDDPST2 [2]	Recommended input voltage	3.0	3.3	3.6	V
I_{VDD}	Cumulative input current [2]	0.35	—	—	A
T_A	Ambient temperature	–40	—	105	°C

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

[2] If writing to eFuses, the voltage on its power supply pin VDDPST2 should not exceed 3.3 V as the circuits responsible for burning eFuses are sensitive to higher voltages.

[3] If you use a single power supply, the recommended output current is 0.35 A or more.

5.3 DC Characteristics (3.3 V, 25 °C)

Table 5-3. 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 × VDD [1]	—	VDD [1] + 0.3	V
V_{IL}	Low-level input voltage	–0.3	—	0.25 × VDD [1]	V
I_{IH}	High-level input current	—	—	50	nA
I_{IL}	Low-level input current	—	—	50	nA

Cont'd on next page

Table 5-3 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
V_{OH} [2]	High-level output voltage	0.8 × VDD [1]	—	—	V
V_{OL} [2]	Low-level output voltage	—	—	0.1 × VDD [1]	V
I_{OH}	High-level source current (VDD [1] = 3.3 V, V_{OH} >= 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	—	kΩ
R_{PD}	Internal weak pull-down resistor	—	45	—	kΩ
V_{IH_nRST}	Chip reset release voltage (CHIP_EN voltage is within the specified range)	0.75 × VDD [1]	—	VDD [1] + 0.3	V
V_{IL_nRST}	Chip reset voltage (CHIP_EN voltage is within the specified range)	–0.3	—	0.25 × 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.

5.4 ADC Characteristics

The measurements in this section are taken with an external 100 nF capacitor connected to the ADC, using DC signals as input, 3.3 V voltage, and at an ambient temperature of 25 °C with the disabled modem.

Table 5-4. 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 5-5. For higher accuracy, you may implement your own calibration methods.

Table 5-5. ADC Calibration Results

Parameter	Description	Min	Max	Unit
Total error	ATTEN0, effective measurement range of 0 ~ 1000	–7	7	mV
	ATTEN1, effective measurement range of 0 ~ 1300	–8	8	mV
	ATTEN2, effective measurement range of 0 ~ 1900	–12	12	mV
	ATTEN3, effective measurement range of 0 ~ 3300	–23	23	mV

Note:

The above ADC measurement range and accuracy are applicable to chips manufactured on and after the Date Code **342023** on shielding cases, or assembled on and after the D/C 1 and D/C 2 **2334** 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*.

5.5 Current Consumption

5.5.1 RF Current Consumption in Active Mode

The current consumption measurements are taken with a 3.3 V supply at 25 °C of ambient temperature at the RF port. All transmitters' measurements are based on a 100% duty cycle.

Table 5-6. Bluetooth LE Current Consumption in Active Mode

Work Mode	Description		Peak (mA)
Active (RF working)	TX	Bluetooth LE @ 20.0 dBm	140
		Bluetooth LE @ 9.0 dBm	60
		Bluetooth LE @ 0 dBm	36
		Bluetooth LE @ -24.0 dBm	24
	RX	Bluetooth LE	24

Table 5-7. 802.15.4 Current Consumption in Active Mode

Work Mode	Description		Peak (mA)
Active (RF working)	TX	802.15.4 @ 20.0 dBm	140
		802.15.4 @ 9.0 dBm	60
		802.15.4 @ 0 dBm	36
		802.15.4 @ –24.0 dBm	24
	RX	802.15.4	25

5.5.2 Current Consumption in Other Modes

The measurements below are applicable to ESP32-H2FH2S and ESP32-H2FH4S.

Table 5-8. Current Consumption in Modem-sleep Mode

Work mode	Frequency (MHz)	Description	Typ [1] (mA) All Peripheral Clocks Disabled	Typ [1] (mA) All Peripheral Clocks Enabled
Modem-sleep [2]	96	CPU running	10	17
		CPU in idle	6	13
	64	CPU running	8	13
		CPU in idle	5	10

Cont'd on next page

Table 5-8 – cont'd from previous page

Work mode	Frequency (MHz)	Description	Typ [1] (mA) All Peripheral Clocks Disabled	Typ [1] (mA) All Peripheral Clocks enabled
	48	CPU running	7	11
		CPU in idle	5	9
	32	CPU running	4	8
		CPU in idle	3	7

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

[2] In Modem-sleep mode, the current consumption might be higher when accessing flash.

Table 5-9. Current Consumption in Low-Power Modes

Work mode	Description	Typ (μA)
Light-sleep	CPU and wireless communication modules are powered down, peripheral clocks are disabled, and all GPIOs are high-impedance	85
	CPU, wireless communication modules and peripherals are powered down, and all GPIOs are high-impedance	25
Deep-sleep	LP timer and LP memory are powered on	7
Power off	CHIP_EN is set to low level, the chip is powered off	1

5.6 Reliability

Table 5-10. 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] ± 2000 V	JS-001
	CDM (Charge Device Mode) [2] ± 1000 V	JS-002
Latch up	Current trigger ± 200 mA	JESD78
	Voltage trigger 1.5 × 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

Cont'd on next page

Table 5-10 – cont'd from previous page

Test Item	Test Conditions	Test Standard
HTSL (High Temperature Storage Life)	150 °C, 1000 hours	JESD22-A103
LTSL (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 0 Ω 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 (±5%) supply at 25 °C ambient temperature.

6.1 Bluetooth LE Radio

Table 6-1. Bluetooth LE RF Characteristics

Name	Description
Center frequency range of operating channel	2402 ~ 2480 MHz
RF transmit power range	–24.0 ~ 20.0 dBm

6.1.1 Bluetooth LE RF Transmitter (TX) Characteristics

Table 6-2. Bluetooth LE - Transmitter Characteristics - 1 Mbps

Parameter	Description	Min	Typ	Max	Unit
Carrier frequency offset and drift	Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$	—	1.5	—	kHz
	Max. $\lvert f_0 - f_n \rvert_{n=2,\,3,\,4,\,\ldots k}$	—	2.8	—	kHz
	Max. $\lvert f_n - f_{n-5} \rvert_{n=6,\,7,\,8,\,\ldots k}$	—	1.3	—	kHz
	$\lvert f_1 - f_0 \rvert$	—	2.3	—	kHz
Modulation characteristics	$\Delta\, F1_{avg}$	—	251.8	—	kHz
	Min. $\Delta\, F2_{max}$ (for at least 99.9% of all $\Delta\, F2_{max}$)	—	217.0	—	kHz
	$\Delta\, F2_{avg} / \Delta\, F1_{avg}$	—	0.87	—	—
In-band emissions	± 2 MHz offset	—	–28	—	dBm
	± 3 MHz offset	—	–32	—	dBm
	> ± 3 MHz offset	—	–34	—	dBm

Table 6-3. Bluetooth LE - Transmitter Characteristics - 2 Mbps

Parameter	Description	Min	Typ	Max	Unit
Carrier frequency offset and drift	Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$	—	3.3	—	kHz
	Max. $\lvert f_0 - f_n \rvert_{n=2,\,3,\,4,\,\ldots k}$	—	3.3	—	kHz
	Max. $\lvert f_n - f_{n-5} \rvert_{n=6,\,7,\,8,\,\ldots k}$	—	1.6	—	kHz
	$\lvert f_1 - f_0 \rvert$	—	2.3	—	kHz

Cont'd on next page

Table 6-3 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
Modulation characteristics	$\Delta F1_{avg}$	—	499.9	—	kHz
	Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$)	—	492.0	—	kHz
	$\Delta F2_{avg}/\Delta F1_{avg}$	—	0.90	—	—
In-band emissions	± 4 MHz offset	—	–31	—	dBm
	± 5 MHz offset	—	–34	—	dBm
	> ± 5 MHz offset	—	–36	—	dBm

Table 6-4. Bluetooth LE - Transmitter Characteristics - 125 Kbps

Parameter	Description	Min	Typ	Max	Unit
Carrier frequency offset and drift	Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$	—	1.0	—	kHz
	Max. $\lvert f_0 - f_n \rvert_{n=1,\,2,\,3,\,\ldots k}$	—	0.5	—	kHz
	$\lvert f_0 - f_3 \rvert$	—	0.4	—	kHz
	Max. $\lvert f_n - f_{n-3} \rvert_{n=7,\,8,\,9,\,\ldots k}$	—	0.9	—	kHz
Modulation characteristics	$\Delta F1_{avg}$	—	250.5	—	kHz
	Min. $\Delta F1_{max}$ (for at least 99.9% of all $\Delta F1_{max}$)	—	234.0	—	kHz
In-band emissions	± 2 MHz offset	—	–23	—	dBm
	± 3 MHz offset	—	–34	—	dBm
	> ± 3 MHz offset	—	–42	—	dBm

Table 6-5. Bluetooth LE - Transmitter Characteristics - 500 Kbps

Parameter	Description	Min	Typ	Max	Unit
Carrier frequency offset and drift	Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$	—	2.3	—	kHz
	Max. $\lvert f_0 - f_n \rvert_{n=1,\,2,\,3,\,\ldots k}$	—	0.7	—	kHz
	$\lvert f_0 - f_3 \rvert$	—	0.3	—	kHz
	Max. $\lvert f_n - f_{n-3} \rvert_{n=7,\,8,\,9,\,\ldots k}$	—	1.1	—	kHz
Modulation characteristics	$\Delta F2_{avg}$	—	230.6	—	kHz
	Min. $\Delta F2_{max}$ (for at least 99.9% of all $\Delta F2_{max}$)	—	221.8	—	kHz
In-band emissions	± 2 MHz offset	—	–28	—	dBm
	± 3 MHz offset	—	–33	—	dBm
	> ± 3 MHz offset	—	–35	—	dBm

Note that the In-band emissions in Table 6-2 and Table 6-5 above are tested at 15 dBm of TX power. However, the test result still meets the Bluetooth SIG standard even if the TX power is increased up to 20 dBm.

6.1.2 Bluetooth LE RF Receiver (RX) Characteristics

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

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER		—	—	–99.0	—	dBm
Maximum received signal @30.8% PER		—	—	8	—	dBm
C/I and receiver selectivity performance	Co-channel	F = F0 MHz	—	4	—	dB
	Adjacent channel	F = F0 + 1 MHz	—	2	—	dB
		F = F0 – 1 MHz	—	0	—	dB
		F = F0 + 2 MHz	—	–29	—	dB
		F = F0 – 2 MHz	—	–29	—	dB
		F = F0 + 3 MHz	—	–35	—	dB
		F = F0 – 3 MHz	—	–36	—	dB
		F ≥ F0 + 4 MHz	—	–30	—	dB
		F ≤ F0 – 4 MHz	—	–36	—	dB
	Image frequency	—	—	–35	—	dB
	Adjacent channel to image frequency	F = F_{image} + 1 MHz	—	–30	—	dB
		F = F_{image} – 1 MHz	—	–29	—	dB
Out-of-band blocking performance		30 MHz ~ 2000 MHz	—	–16	—	dBm
		2003 MHz ~ 2399 MHz	—	–12	—	dBm
		2484 MHz ~ 2997 MHz	—	–16	—	dBm
		3000 MHz ~ 12.75 GHz	—	0	—	dBm
Intermodulation		—	—	–35	—	dBm

Table 6-7. Bluetooth LE - Receiver Characteristics - 2 Mbps

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER		—	—	–96.0	—	dBm
Maximum received signal @30.8% PER		—	—	8	—	dBm
C/I and receiver selectivity performance	Co-channel	F = F0 MHz	—	5	—	dB
	Adjacent channel	F = F0 + 2 MHz	—	1	—	dB
		F = F0 – 2 MHz	—	-2	—	dB
		F = F0 + 4 MHz	—	–27	—	dB
		F = F0 – 4 MHz	—	–32	—	dB
		F = F0 + 6 MHz	—	–33	—	dB
		F = F0 – 6 MHz	—	–36	—	dB
		F ≥ F0 + 8 MHz	—	–36	—	dB
		F ≤ F0 – 8 MHz	—	–36	—	dB
	Image frequency	—	—	–26	—	dB
	Adjacent channel to image frequency	F = F_{image} + 2 MHz	—	–30	—	dB
		F = F_{image} – 2 MHz	—	3	—	dB
Out-of-band blocking performance		30 MHz ~ 2000 MHz	—	–17	—	dBm
		2003 MHz ~ 2399 MHz	—	–27	—	dBm
		2484 MHz ~ 2997 MHz	—	–17	—	dBm

Cont'd on next page

Table 6-7 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
	3000 MHz ~ 12.75 GHz	—	0	—	dBm
Intermodulation	—	—	–27	—	dBm

Table 6-8. Bluetooth LE - Receiver Characteristics - 125 Kbps

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER		—	—	–106.5	—	dBm
Maximum received signal @30.8% PER		—	—	8	—	dBm
C/I and receiver selectivity performance	Co-channel	F = F0 MHz	—	0	—	dB
	Adjacent channel	F = F0 + 1 MHz	—	–4	—	dB
		F = F0 – 1 MHz	—	–6	—	dB
		F = F0 + 2 MHz	—	–31	—	dB
		F = F0 – 2 MHz	—	–34	—	dB
		F = F0 + 3 MHz	—	–39	—	dB
		F = F0 – 3 MHz	—	–48	—	dB
		F ≥ F0 + 4 MHz	—	–35	—	dB
		F ≤ F0 – 4 MHz	—	–48	—	dB
	Image frequency	—	—	–39	—	dB
	Adjacent channel to image frequency	F = F_{image} + 1 MHz	—	–35	—	dB
		F = F_{image} – 1 MHz	—	–31	—	dB

Table 6-9. Bluetooth LE - Receiver Characteristics - 500 Kbps

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER		—	—	–102.5	—	dBm
Maximum received signal @30.8% PER		—	—	8	—	dBm
C/I and receiver selectivity performance	Co-channel	F = F0 MHz	—	2	—	dB
	Adjacent channel	F = F0 + 1 MHz	—	–1	—	dB
		F = F0 – 1 MHz	—	–4	—	dB
		F = F0 + 2 MHz	—	–28	—	dB
		F = F0 – 2 MHz	—	–29	—	dB
		F = F0 + 3 MHz	—	–38	—	dB
		F = F0 – 3 MHz	—	–41	—	dB
		F ≥ F0 + 4 MHz	—	–33	—	dB
		F ≤ F0 – 4 MHz	—	–41	—	dB
	Image frequency	—	—	–38	—	dB
	Adjacent channel to image frequency	F = F_{image} + 1 MHz	—	–33	—	dB
		F = F_{image} – 1 MHz	—	–28	—	dB

6.2 802.15.4 Radio

Table 6-10. 802.15.4 RF Characteristics

Name	Description [1]
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.

6.2.1 802.15.4 RF Transmitter (TX) Characteristics

Table 6-11. 802.15.4 Transmitter Characteristics - 250 Kbps

Parameter	Min	Typ	Max	Unit
RF transmit power range	–24.0	—	20.0	dBm
EVM	—	3.5%	—	—

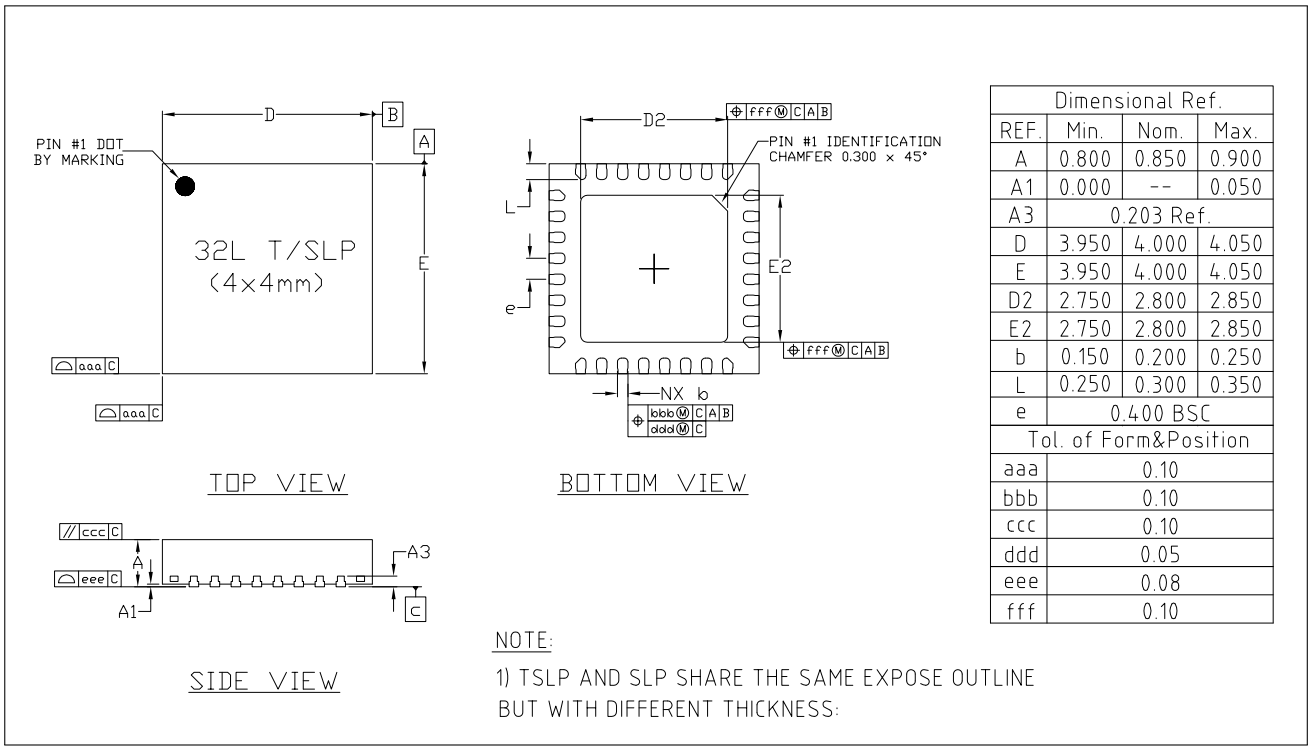
6.2.2 802.15.4 RF Receiver (RX) Characteristics

Table 6-12. 802.15.4 Receiver Characteristics - 250 Kbps

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @1% PER		—	—	–102.5	—	dBm
Maximum received signal @1% PER		—	—	8	—	dBm
Relative jamming level	Adjacent channel	F = F0 + 5 MHz	—	31	—	dB
		F = F0 – 5 MHz	—	43	—	dB
	Alternate channel	F = F0 + 10 MHz	—	49	—	dB
		F = F0 – 10 MHz	—	54	—	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-H2 Pin Layout (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.



Dimensional Ref.			
REF.	Min.	Nom.	Max.
A	0.800	0.850	0.900
A1	0.000	--	0.050
A3	0.203 Ref.		
D	3.950	4.000	4.050
E	3.950	4.000	4.050
D2	2.750	2.800	2.850
E2	2.750	2.800	2.850
b	0.150	0.200	0.250
L	0.250	0.300	0.350
e	0.400 BSC		
Tol. of Form&Position			
aaa	0.10		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

Figure 7-1. QFN32 (4×4 mm) Package

Related Documentation and Resources

Related Documentation

- ESP32-H2 Technical Reference Manual – Detailed information on how to use the ESP32-H2 memory and peripherals.
- ESP32-H2 Hardware Design Guidelines – Guidelines on how to integrate the ESP32-H2 into your hardware product.
- ESP32-H2 Series SoC Errata – Descriptions of known errors in ESP32-H2 series of SoCs.
- *Certificates*
  https://espressif.com/en/support/documents/certificates
- *ESP32-H2 Product/Process Change Notifications (PCN)*
  https://espressif.com/en/support/documents/pcns?keys=ESP32-H2
- *ESP32-H2 Advisories* – Information on security, bugs, compatibility, component reliability.
  https://espressif.com/en/support/documents/advisories?keys=ESP32-H2
- *Documentation Updates and Update Notification Subscription*
  https://espressif.com/en/support/download/documents

Developer Zone

- ESP-IDF Programming Guide for ESP32-H2 – 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-H2 Series SoCs* – Browse through all ESP32-H2 SoCs.
  https://espressif.com/en/products/socs?id=ESP32-H2
- *ESP32-H2 Series Modules* – Browse through all ESP32-H2-based modules.
  https://espressif.com/en/products/modules?id=ESP32-H2
- *ESP32-H2 Series DevKits* – Browse through all ESP32-H2-based devkits.
  https://espressif.com/en/products/devkits?id=ESP32-H2
- *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

Appendix A – ESP32-H2 Consolidated Pin Overview

Pin No.	Pin Name	Pin Type	Pin Providing Power	Pin Settings		Analog Function					
				At Reset	After Reset	0	1	0	Type	1	Ty
1	VDD3P3	Power									
2	VDD3P3	Power									
3	GPIO0	IO	VDDPST1					**GPIO0**	I/O/T	GPIO0	I/O
4	GPIO1	IO	VDDPST1				ADC1_CH0	**GPIO1**	I/O/T	GPIO1	I/O
5	MTMS	IO	VDDPST1	IE	IE		ADC1_CH1	**MTMS**	I1	GPIO2	I/O
6	MTDO	IO	VDDPST1	IE	IE		ADC1_CH2	**MTDO**	O/T	GPIO3	I/O
7	MTCK	IO	VDDPST1		IE*		ADC1_CH3	**MTCK**	I1	GPIO4	I/O
8	MTDI	IO	VDDPST1		IE		ADC1_CH4	**MTDI**	I1	GPIO5	I/O
9	VDDPST1	Power									
10	GPIO8	IO	VDDPST1	IE	IE			**GPIO8**	I/O/T	GPIO8	I/O
11	GPIO9	IO	VDDPST1	IE, WPU	IE, WPU			**GPIO9**	I/O/T	GPIO9	I/O
12	GPIO10	IO	VDDPST1			ZCD0		**GPIO10**	I/O/T	GPIO10	I/O
13	GPIO11	IO	VDDPST1			ZCD1		**GPIO11**	I/O/T	GPIO11	I/O
14	GPIO12	IO	VDDA_PMU/VBAT					**GPIO12**	I/O/T	GPIO12	I/O
15	XTAL_32K_P	IO	VDDA_PMU/VBAT			XTAL_32K_P		**GPIO13**	I/O/T	GPIO13	I/O
16	XTAL_32K_N	IO	VDDA_PMU/VBAT			XTAL_32K_N		**GPIO14**	I/O/T	GPIO14	I/O
17	CHIP_EN	Analog	VBAT								
18	VBAT	Power									
19	VDDA_PMU	Power									
20	VDDPST2	Power									
21	GPIO22	IO	VDDPST2					**GPIO22**	I/O/T	GPIO22	I/O
22	U0RXD	IO	VDDPST2		IE, WPU			**U0RXD**	I1	GPIO23	I/O
23	U0TXD	IO	VDDPST2		IE, WPU			**U0TXD**	O	GPIO24	I/O
24	GPIO25	IO	VDDPST2	IE	IE			**GPIO25**	I/O/T	GPIO25	I/O
25	GPIO26	IO	VDDPST2		IE	**USB_D-**		GPIO26	I/O/T	GPIO26	I/O
26	GPIO27	IO	VDDPST2		IE, USB_PU	**USB_D+**		GPIO27	I/O/T	GPIO27	I/O
27	VDD3P3	Power									
28	XTAL_N	Analog									
29	XTAL_P	Analog									
30	VDD3P3	Power									
31	VDD3P3	Power									
32	ANT	Analog									
33	GND	Power									

* For details, see Section 2 *Pins*. Regarding highlighted cells, see Section 2.3.3 *Restrictions for GPIOs*.

Datasheet Versioning

Datasheet Version	Status	Watermark	Definition
v0.1 ~ v0.5 (excluding v0.5)	Draft	Confidential	This datasheet is under development for products in the design stage. Specifications may change without prior notice.
v0.5 ~ v1.0 (excluding v1.0)	Preliminary release	Preliminary	This datasheet is actively updated for products in the verification stage. Specifications may change before mass production, and the changes will be documentation in the datasheet's Revision History.
v1.0 and higher	Official release	—	This datasheet is publicly released for products in mass production. Specifications are finalized, and major changes will be communicated via Product Change Notifications (PCN).
Any version	—	Not Recommended for New Design (NRND)[1]	This datasheet is updated less frequently for products not recommended for new designs.
Any version	—	End of Life (EOL)[2]	This datasheet is no longer mtained for products that have reached end of life.

[1] Watermark will be added to the datasheet title page only when all the product variants covered by this datasheet are not recommended for new designs.

[2] Watermark will be added to the datasheet title page only when all the product variants covered by this datasheet have reached end of life.

Revision History

Date	Version	Release notes
2025-10-13	v1.2	• Updated Figure 3-1 *Visualization of Timing Parameters for the Strapping Pins* • Updated Section 4.1.3.6 *Power Management Unit* • Updated Section 4.2.2.1 *SAR ADC* • Updated Table 5-2 *Recommended Operating Conditions* • Added Section 7 *Datasheet Versioning*
2025-02-28	v1.1	• Improved CoreMark scores in Section CPU and Memory • Updated the description of internal slow RC oscillator in Section 4.1.3.3 *Clock* • Added a note below Table 5-2 *Recommended Operating Conditions* • Updated the ordering code in Table 1-1 *ESP32-H2 Series Comparison* • Updated or added the following sections based on chip revision v1.2: – Updated sections 4.1.4.1 *AES Accelerator*, 4.1.4.7 *Elliptic Curve Digital Signature Algorithm (ECDSA)*, and 4.1.4.8 *External Memory Encryption and Decryption* – Added Section 4.1.4.10 *Power Glitch Detector*
2024-09-27	v1.0	• Official release • Improved the content, formatting, structure, and wording of the whole document
2023-10-17	v0.7	• Added Section 4.5 • Updated the description in Section 3.5.1 • Updated measurements in Table 4-9
2023-08-02	v0.6	• Updated the description in Section 2.4.1 • Updated the note about USB under Table 3-1 • Updated the description in Section 3.2 • Updated the list of peripherals that support ETM in Section 3.5.10 • Reordered the table content in Table 4-9 from the highest CPU frequency to lowest CPU frequency; • Updated all the measurements in Table 4-9 and Table 4-10 to integers • Added two notes in Chapter 6
2023-05-24	v0.5	Preliminary release

Disclaimer and Copyright Notice

www.espressif.com

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View ESP32-H2FH2S on WIN SOURCE
- Espressif Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management