ESP32-PICO Series

Datasheet Version 1.1

2.4 GHz Wi-Fi + Bluetooth® + Bluetooth LE SiP
Integrating all peripheral components in one single package

Including:

ESP32-PICO-V3

ESP32-PICO-V3-02

ESP32-PICO-D4 – Not Recommended for New Designs (NRND)

ESPRESSIF

Product Overview

The ESP32-PICO series is a System-in-Package (SiP) device that is based on the ESP32 SoC. The ESP32-PICO series include ESP32-PICO-D4, ESP32-PICO-V3, and ESP32-PICO-V3-02 variants. In this document, unless otherwise stated, "ESP32-PICO" refers to all the variants.

ESP32-PICO provides Wi-Fi 802.11b/g/n, Bluetooth® v4.2 BR/EDR, and Bluetooth LE functionalities. It integrates all peripheral components seamlessly in a single package, including a crystal oscillator, filter capacitors, SPI flash/PSRAM (optional), and RF matching circuit. ESP32-PICO is built in an ultra-small size, with robust performance and low energy consumption. It is well suited for any space-limited or battery-operated applications, such as wearable electronics, medical equipment, sensors, and other IoT products.

The ESP32-PICO series of variants are similar to each other, but still vary in some aspects, for example, the embedded chip revision, pin layout, dimensions, etc. Table 1 lists the differences between these variants. For detailed description please go to specific sections.

If you would like to migrate from the existing module design based on older ESP32-PICO variants to a new design based on newer ESP32-PICO variants, please refer to section 13 Migration Guide.

Table 1: Differences Between ESP32-PICO Series of Variants

Differences in	Section
Chip revision	Section 1 ESP32-PICO Series Comparison
In-package flash and PSRAM	Section 1 ESP32-PICO Series Comparison
Package and dimensions	Section 9 Packaging
Pin layout	Section 2.1 ESP32-PICO-D4, 2.2 ESP32-PICO-V3 and ESP32-PICO-V3-02
Pin compatibility	Section 2.4 Pin Compatibility Between ESP32-PICO Variants
Schematics	Section 7 Schematics
Peripheral schematics	Section 8 Peripheral Schematics

Block Diagram

The functional block diagram of the SoC is shown below.

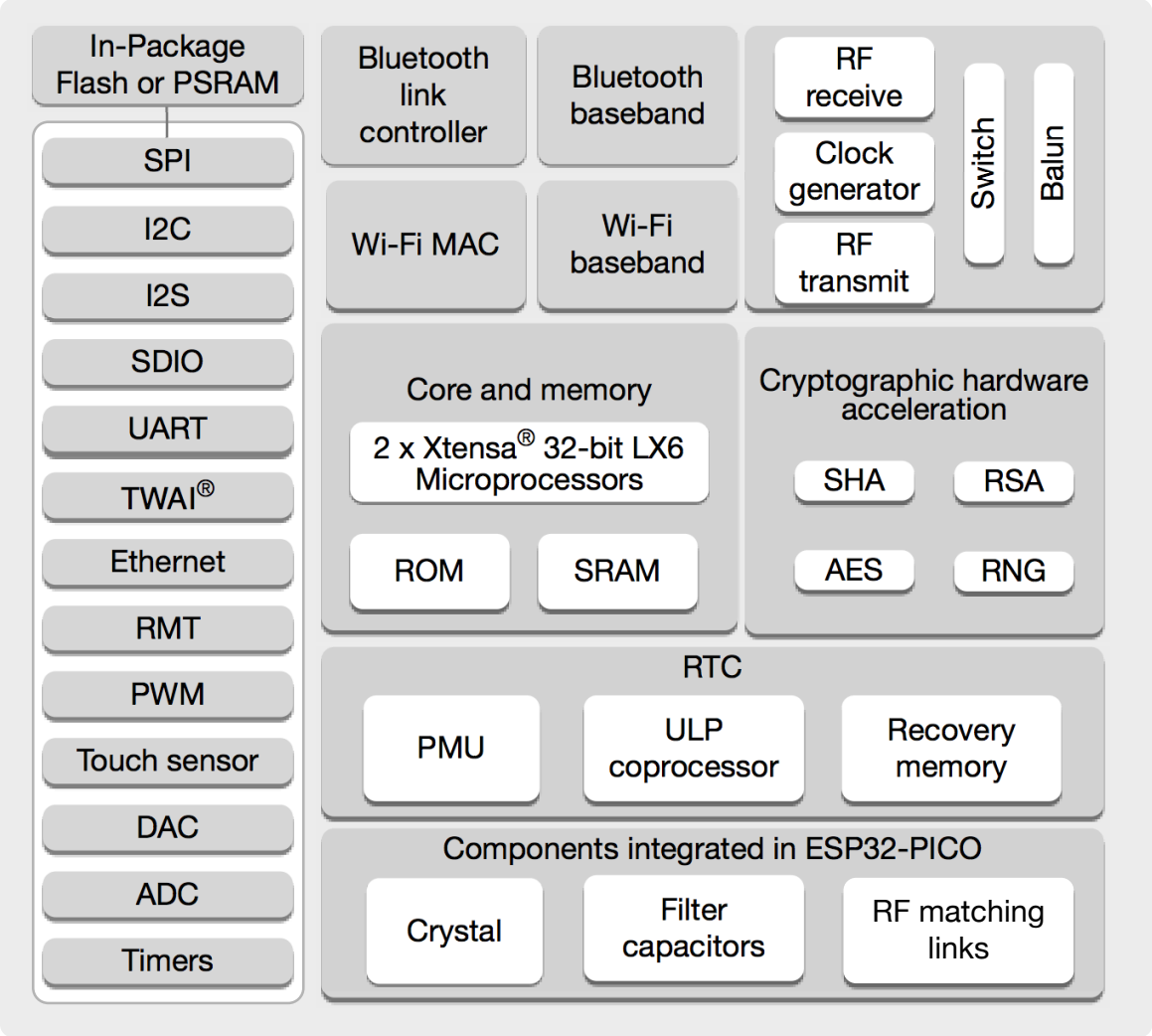


Figure 1: ESP32-PICO Functional Block Diagram

Features

CPU and On-Chip Memory

- Xtensa® dual-core 32-bit LX6 microprocessor, up to 240 MHz
- 448 KB ROM
- 520 KB SRAM
- 16 KB SRAM in RTC

Wi-Fi

- Complies with 802.11b/g/n
- 1T1R mode with data rate up to 150 Mbps
- TX/RX A-MPDU and RX A-MSDU aggregation
- 0.4 μs guard interval support
- Center frequency range of operating channel: 2412 ~ 2484 MHz

Bluetooth

- Compliant with Bluetooth v4.2 BR/EDR and Bluetooth LE specifications
- Class-1, class-2 and class-3 transmitter
- Adaptive Frequency Hopping (AFH)
- CVSD and SBC for audio codec

Peripherals

- Up to 34 GPIOs for ESP32-PICO-D4
  - 5 strapping GPIOs
  - 6 GPIOs used for in-package flash
  - 6 input-only GPIOs
- Up to 31 GPIOs for ESP32-PICO-V3
  - 5 strapping GPIOs
  - 2 GPIOs used for in-package flash
  - 6 input-only GPIOs
- Up to 31 GPIOs for ESP32-PICO-V3-02
  - 5 strapping GPIOs
  - 4 GPIOs used for in-package flash and PSRAM
  - 6 input-only GPIOs
- SD/SDIO/MMC Host Controller, UART, SPI, SDIO/SPI Slave Controller, I2C, LED PWM, Motor PWM, I2S, infrared remote controller, pulse counter, capacitive touch sensor, ADC, DAC, Ethernet MAC, TWAI® (compatible with ISO 11898-1, i.e. CAN 2.0 Specifications)

In-Package Flash and PSRAM

- ESP32-PICO-D4: 4 MB flash
- ESP32-PICO-V3: 4 MB flash
- ESP32-PICO-V3-02: 8 MB flash, 2 MB PSRAM

Operating Conditions

- Operating voltage/Power supply: 3.0 ~ 3.6 V
- Operating ambient temperature: −40 ~ 85 °C

Note:
For a detailed description of the features listed above, please refer to *ESP32 Series Datasheet* > Section *Functional Description*.

Applications

With low power consumption, ESP32-PICO 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
- Cameras for Video Streaming
- Speech Recognition
- Image Recognition
- SDIO Wi-Fi + Bluetooth Networking Card

Note:

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

Contents

List of Tables

List of Figures

1 ESP32-PICO Series Comparison

1.1 ESP32-PICO Series Nomenclature

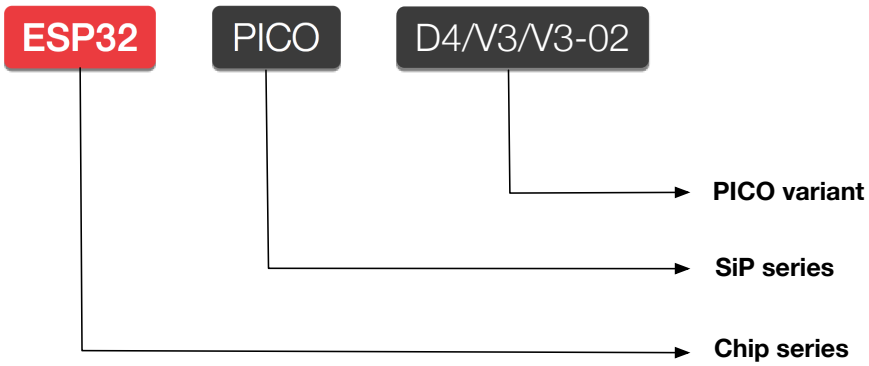


Figure 2: ESP32-PICO Series Nomenclature

1.2 Comparison

Table 1: ESP32-PICO Series Comparison

Ordering Code	Chip Revision[1]	In-Package Flash[5]	In-Package PSRAM	Dimensions (mm)
ESP32-PICO-D4 (NRND)	v1.0/v1.1[2]	4 MB (Quad SPI)	—	7.0 x 7.0 x 0.94
ESP32-PICO-V3	v3.0/v3.1[3, 4]	4 MB (Quad SPI)	—	7.0 x 7.0 x 0.94
ESP32-PICO-V3-02	v3.0/v3.1[3, 4]	8 MB (Quad SPI)	2 MB (Quad SPI)	7.0 x 7.0 x 1.11

[1] For chip revision identification and chip revision-specific errata, see *ESP32 Series SoC Errata*.

[2] The ESP32 chip revision on ESP32-PICO-D4 is upgraded from v1.0 to v1.1. See PCN20220901 for more details.

[3] The ESP32 chip revision on ESP32-PICO-V3 and ESP32-PICO-V3-02 is upgraded from v3.0 to v3.1. See PCN20220901 for more details.

[4] For differences between chip revision v3.0 and previous ESP32 chip revisions, please refer to *ESP32 Chip Revision v3.0 User Guide*.

[5] The in-package flash supports:

- More than 100,000 program/erase cycles
- More than 20 years data retention time

2 Pins

2.1 ESP32-PICO-D4

2.1.1 Pin Layout

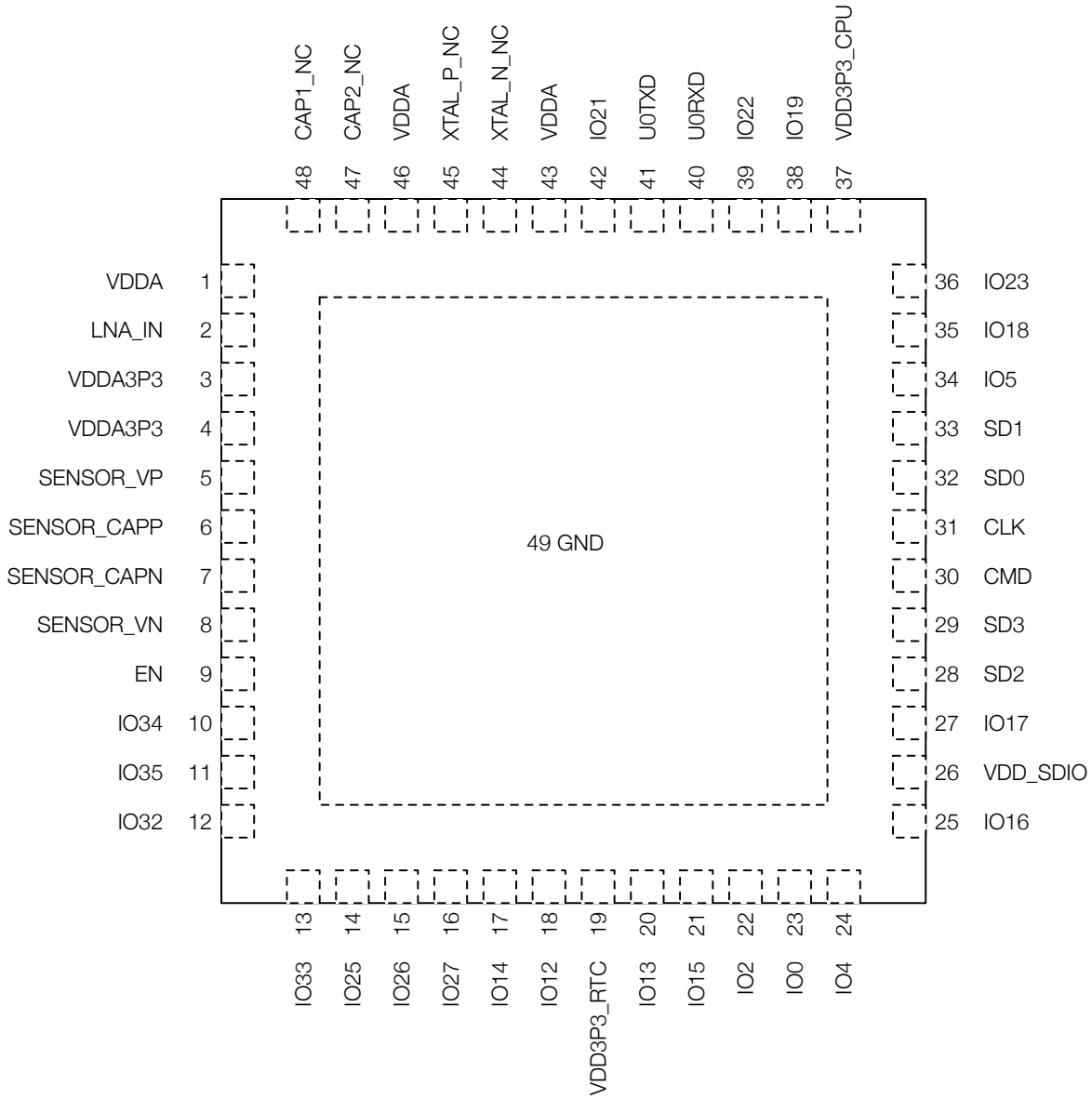


Figure 3: Pin Layout of ESP32-PICO-D4 (Top View)

2.1.2 Pin Description

Notes for Table 2 Pin Description:

1. some pin functions are highlighted, specically:
   - GPIO – **Input only pins**, output is not supported due to lack of pull-up/pull-down resistors.
   - The highlighted cells indicate pins that are connected to the in-package flash. For details see Section 2.1.3 Pin Mapping Between ESP32-PICO-D4 and Flash/PSRAM.
   - GPIO – have one of the following important functions:
     - **Strapping pins** – need to be at certain logic levels at startup. See Section 3 *Boot Configurations*.
     - **JTAG interface** – often used for debugging.
     - **UART interface** – often used for debugging.
2. For definition of functions in column **Function**, see Section 2.3 Pin Function Description.
3. **Type**: I/O — Input/Output; I — Input.

Table 2: Pin Description of ESP32-PICO-D4

Name	No.	Type	Function	
Analog				
VDDA	1	Power	Analog power supply	
LNA_IN	2	I/O	RF input and output	
VDDA3P3	3	Power	Analog power supply	
VDDA3P3	4	Power	Analog power supply	
VDD3P3_RTC				
SENSOR_VP	5	I	GPIO36,	ADC1_CH0, RTC_GPIO0
SENSOR_CAPP	6	I	GPIO37,	ADC1_CH1, RTC_GPIO1
SENSOR_CAPN	7	I	GPIO38,	ADC1_CH2, RTC_GPIO2
SENSOR_VN	8	I	GPIO39,	ADC1_CH3, RTC_GPIO3

Cont'd on next page

Table 2 – cont'd from previous page

Name	No.	Type	Function
EN	9	I	High: On; enables the SiP Low: Off; the SiP shuts down Note: Do not leave this pin floating.
IO34	10	I	GPIO34, ADC1_CH6, RTC_GPIO4
IO35	11	I	GPIO35, ADC1_CH7, RTC_GPIO5
IO32	12	I/O	ADC1_CH4,TOUCH9, RTC_GPIO9,32K_XP (32.768 kHz crystal oscillator input)
IO33	13	I/O	GPIO33, ADC1_CH5, TOUCH8, RTC_GPIO8, 32K_XN (32.768 kHz crystal oscillator output)
IO25	14	I/O	GPIO25, DAC_1, ADC2_CH8,RTC_GPIO6, EMAC_RXD0
IO26	15	I/O	GPIO26, DAC_2, ADC2_CH9,RTC_GPIO7, EMAC_RXD1
IO27	16	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
IO14	17	I/O	GPIO14, ADC2_CH6,TOUCH6, RTC_GPIO16,HSPICLK, HS2_CLK, SD_CLK, MTMS, EMAC_TXD2
IO12	18	I/O	GPIO12, ADC2_CH5,TOUCH5, RTC_GPIO15,HSPIQ, HS2_DATA2, SD_DATA2, MTDI, EMAC_TXD3
VDD3P3_RTC	19	Power	Input power supply for RTC IO
IO13	20	I/O	GPIO13, ADC2_CH4,TOUCH4, RTC_GPIO14,HSPID, HS2_DATA3, SD_DATA3, MTCK, EMAC_RX_ER
IO15	21	I/O	GPIO15, ADC2_CH3,TOUCH3, RTC_GPIO13,HSPICS0, HS2_CMD, SD_CMD, MTDO, EMAC_RXD3
IO2	22	I/O	GPIO2, ADC2_CH2,TOUCH2, RTC_GPIO12,HSPIWP, HS2_DATA0, SD_DATA0
IO0	23	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1,EMAC_TX_CLK
IO4	24	I/O	GPIO4, ADC2_CH0,TOUCH0, RTC_GPIO10,HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
VDD_SDIO			
IO16	25	I/O	GPIO16, HS1_DATA4,U2RXD, EMAC_CLK_OUT
VDD_SDIO	26	Power	Output power supply
IO17	27	I/O	GPIO17, HS1_DATA5,U2TXD, EMAC_CLK_OUT_180
SD2	28	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SD3	29	I/O	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
CMD	30	I/O	GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS
CLK	31	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
SD0	32	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS

Cont'd on next page

Table 2 – cont'd from previous page

Name	No.	Type	Function
SD1	33	I/O	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
VDD3P3_CPU			
IO5	34	I/O	GPIO5, VSPICS0, HS1_DATA6,EMAC_RX_CLK
IO18	35	I/O	GPIO18, VSPICLK, HS1_DATA7
IO23	36	I/O	GPIO23, VSPID, HS1_STROBE
VDD3P3_CPU	37	Power	Input power supply for CPU IO
IO19	38	I/O	GPIO19, VSPIQ, U0CTS, EMAC_TXD0
IO22	39	I/O	GPIO22, VSPIWP, U0RTS, EMAC_TXD1
U0RXD	40	I/O	GPIO3, U0RXD, CLK_OUT2
U0TXD	41	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
IO21	42	I/O	GPIO21, VSPIHD, EMAC_TX_EN
Analog			
VDDA	43	Power	Analog power supply
XTAL_N_NC	44	—	NC
XTAL_P_NC	45	—	NC
VDDA	46	Power	Analog power supply
CAP2_NC	47	—	NC
CAP1_NC	48	—	NC

[1] For chip revision identification and chip revision-specific errata, see *ESP32 Series SoC Errata*.

[2] The ESP32 chip revision on ESP32-PICO-D4 is upgraded from v1.0 to v1.1. See PCN20220901 for more details.

[3] The ESP32 chip revision on ESP32-PICO-V3 and ESP32-PICO-V3-02 is upgraded from v3.0 to v3.1. See PCN20220901 for more details.

[4] For differences between chip revision v3.0 and previous ESP32 chip revisions, please refer to *ESP32 Chip Revision v3.0 User Guide*.

[5] The in-package flash supports:

- More than 100,000 program/erase cycles
- More than 20 years data retention time

2.1.3 Pin Mapping Between ESP32-PICO-D4 and Flash/PSRAM

Table 3 lists ESP32 pins exposed on the package that are also used to connect the in-package flash and off-package PSRAM. **It is not recommended to use the pins connected to flash/PSRAM for any other purposes.**

Table 3: Pin Mapping Between ESP32-PICO-D4 and Flash/PSRAM

Pin No.	Pin Name	In-Package Flash	Off-Package PSRAM
31	CLK	FLASH_CLK	PSRAM_CLK
25	IO16	FLASH_CS	—
29	SD3[1]	—	PSRAM_CS
33	SD1	SI/SIO0	SI/SIO0
27	IO17	SO/SIO1	SI/SIO1
32	SD0	WP/SIO2	SIO2
30	CMD	HOLD/SIO3	SIO3

[1] SD3 is recommended for PSRAM_CS. You can also choose any available GPIO as PSRAM_CS.

2.2 ESP32-PICO-V3 and ESP32-PICO-V3-02

2.2.1 Pin Layout

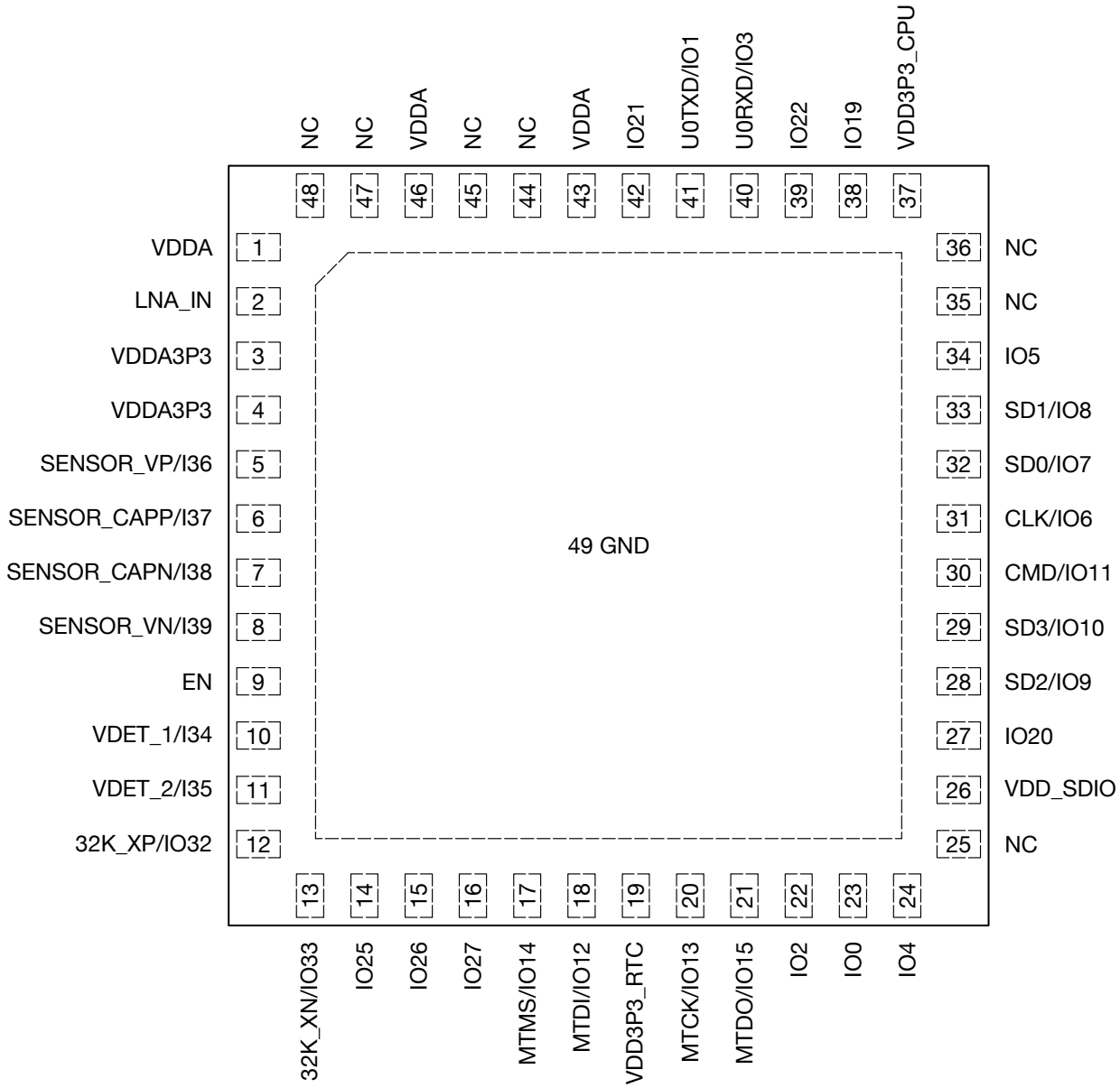


Figure 4: Pin Layout of ESP32-PICO-V3 and ESP32-PICO-V3-02 (Top View)

2.2.2 Pin Description

Notes for Table 4 Pin Description:

1. some pin functions are highlighted, specically:
   - GPIO – **Input only pins**, output is not supported due to lack of pull-up/pull-down resistors.
   - The highlighted cells indicate pins that are connected to the in-package flash. For details see Section 2.1.3 Pin Mapping Between ESP32-PICO-D4 and Flash/PSRAM.
   - GPIO – have one of the following important functions:
     - **Strapping pins** – need to be at certain logic levels at startup. See Section 3 *Boot Configurations*.
     - **JTAG interface** – often used for debugging.
     - **UART interface** – often used for debugging.
2. For ESP32-PICO-V3: IO6/IO7/IO8/IO9/IO10/IO11/IO20 belong to VDD_SDIO power domain and can not work when VDD_SDIO power shuts down.
3. For definition of functions in column **Function**, see Section 2.3 Pin Function Description.
4. **Type**: I/O — Input/Output; I — Input.

Table 4: Pin Description of ESP32-PICO-V3 and ESP32-PICO-V3-02

Name	No.	Type	Function	
Analog				
VDDA	1	Power	Analog power supply	
LNA_IN	2	I/O	RF input and output	
VDDA3P3	3	Power	Analog power supply	
VDDA3P3	4	Power	Analog power supply	
VDD3P3_RTC				
SENSOR_VP/I36	5	I	GPIO36,	ADC1_CH0, RTC_GPIO0
SENSOR_CAPP/I37	6	I	GPIO37,	ADC1_CH1, RTC_GPIO1
SENSOR_CAPN/I38	7	I	GPIO38,	ADC1_CH2, RTC_GPIO2

Cont'd on next page

Table 4 – cont'd from previous page

Name	No.	Type	Function
SENSOR_VN/I39	8	I	GPIO39, ADC1_CH3, RTC_GPIO3
EN	9	I	High: On; enables the SiP Low: Off; the SiP powers off Note: Do not leave this pin floating.
VDET_1/I34	10	I	ADC1_CH6, RTC_GPIO4
VDET_2/I35	11	I	ADC1_CH7, RTC_GPIO5
32K_XP/IO32	12	I/O	ADC1_CH4, TOUCH9, RTC_GPIO9, 32K_XP (32.768 kHz crystal oscillator input)
32K_XN/IO33	13	I/O	ADC1_CH5, TOUCH8, RTC_GPIO8, 32K_XN (32.768 kHz crystal oscillator output)
IO25	14	I/O	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
IO26	15	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
IO27	16	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
MTMS/IO14	17	I/O	ADC2_CH6, TOUCH6, RTC_GPIO16, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2, MTMS
MTDI/IO12	18	I/O	ADC2_CH5, TOUCH5, RTC_GPIO15, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3, MTDI
VDD3P3_RTC	19	Power	Input power supply for RTC IO
MTCK/IO13	20	I/O	ADC2_CH4, TOUCH4, RTC_GPIO14, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER, MTCK
MTDO/IO15	21	I/O	ADC2_CH3, TOUCH3, RTC_GPIO13, HSPICS0, HS2_CMD, SD_CMD, EMAC_RXD3, MTDO
IO2	22	I/O	ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0
IO0	23	I/O	ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
IO4	24	I/O	ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
VDD_SDIO			
NC	25	—	NC
VDD_SDIO	26	Power	Output power supply
IO20	27	I/O	GPIO20
SD2/IO9	28	I/O	**ESP32-PICO-V3:** GPIO9, SD_DATA2, HS1_DATA2, U1RXD **ESP32-PICO-V3-02:** Used for connecting in-package PSRAM.
SD3/IO10	29	I/O	**ESP32-PICO-V3:** GPIO10, SD_DATA3, HS1_DATA3, U1TXD **ESP32-PICO-V3-02:** Used for connecting in-package PSRAM.

Cont'd on next page

Table 4 – cont'd from previous page

Name	No.	Type	Function
CMD/IO11	30	I/O	Used for connecting in-package flash
CLK/IO6	31	I/O	Used for connecting in-package flash
SD0/IO7	32	I/O	GPIO7, SD_DATA0, HS1_DATA0, U2RTS
SD1/IO8	33	I/O	GPIO8, SD_DATA1, HS1_DATA1, U2CTS
VDD3P3_CPU			
IO5	34	I/O	GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK
NC	35	—	NC
NC	36	—	NC
VDD3P3_CPU	37	Power	Input power supply for CPU IO
IO19	38	I/O	GPIO19, VSPIQ, U0CTS, EMAC_TXD0
IO22	39	I/O	GPIO22, VSPIWP, U0RTS, EMAC_TXD1
U0RXD/IO3	40	I/O	GPIO3, U0RXD, CLK_OUT2
U0TXD/IO1	41	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
IO21	42	I/O	GPIO21, VSPIHD, EMAC_TX_EN
Analog			
VDDA	43	Power	Analog power supply
NC	44	—	NC
NC	45	—	NC
VDDA	46	Power	Analog power supply
NC	47	—	NC
NC	48	—	NC

[1] SD3 is recommended for PSRAM_CS. You can also choose any available GPIO as PSRAM_CS.

2.2.3 Pin Mapping Between ESP32-PICO-V3/ESP32-PICO-V3-02 and Flash/PSRAM

Table 5 lists ESP32 pins exposed on the package that are also used to connect the in-package flash/PSRAM. **It is not recommended to use the pins connected to flash/PSRAM for any other purposes.**

Table 5: Pin Mapping Between ESP32-PICO-V3/ESP32-PICO-V3-02 and Flash/PSRAM

Pin No.	Pin Name	In-Package Flash	In-Package PSRAM
31	CLK/IO6	FLASH_CLK	—
30	CMD/IO11	FLASH_CS	—
28	SD2/IO9	—	PSRAM_CS
29	SD3/IO10	—	PSRAM_CLK

2.3 Pin Function Description

Table 6 provides description of pin functions.

Table 6: Pin Function Description

Function Name	Description
GPIO*x*	General-purpose input and output (*x* is GPIO number). GPIO pins can be assigned various functions, including digital and analog functions. For more information, please refer to *ESP32 Series Datasheet* > Appendix *IO_MUX*.
MTCK/MTDO/MTDI/MTMS	JTAG interface signals.
32K_XP/XN	32 KHz external clock input/output (connecting to ESP32-PICO's oscillator). P/N means differential clock positive/negative.
RTC_GPIO*x*	RTC domain GPIO function for low power management.
TOUCH*x*	Analog function for touch sensing.
ADC*x*_CH*y*	Analog to digital conversion channel. *x* is ADC number, *y* is channel number.
DAC_*x*	Digital to analog conversion module. *x* is DAC number.
CLK_OUT*x*	Clock output for debugging. *x* is clock number.
SPI*	Signals of SPI0/1 module. * is CLK, CS0, D (MOSI), Q (MISO), WP (write-protect), HD (hold).
HSPI*	Signals of SPI2 module. * is CLK, CS0, D, Q, WP, HD.
VSPI*	Signals of SPI3 module. * is CLK, CS0, D, Q, WP, HD.
U0*	Signals of UART0 module. * is CTS, RTS, RXD, TXD.
U1*	Signals of UART1 module. * is CTS, RTS, RXD, TXD.
U2*	Signals of UART2 module. * is CTS, RTS, RXD, TXD.
SD_*	Signals of SDIO slave. * is CLK, CMD, DATA0 ~ DATA3.
HS1_*	Port 1 signals of the SDIO host, * is CLK, CMD, STROBE, DATA0 ~ DATA7.
HS2_*	Port 2 signals of the SDIO host, * is CLK, CMD, DATA0 ~ DATA3.
NC	Not connected.

2.4 Pin Compatibility Between ESP32-PICO Variants

While the ESP32-PICO variants are very similar from a pin-out perspective, there are several changes to the pins and their functions, as shown in Table 7. The differences in pins require attention when migrating from

one variant to another.

Table 7: Pin Compatibility Between ESP32-PICO Variants

Pin No.	ESP32-PICO-D4	ESP32-PICO-V3	ESP32-PICO-V3-02
5, 6, 7, 8, 10, 11	Input-only and RTC GPIO	Input-only and RTC GPIO	Input-only and RTC GPIO
12, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24	RTC GPIO	RTC GPIO	RTC GPIO
25	GPIO16, used for in-package flash	NC	NC
27	GPIO17, used for in-package flash	GPIO20, can be used freely	GPIO20, can be used freely
28	GPIO9, can be used freely	GPIO9, can be used freely	GPIO9, used for in-package PSRAM
29	GPIO10, can be used freely	GPIO10, can be used freely	GPIO10, used for in-package PSRAM
30	GPIO11, used for in-package flash	GPIO11, used for in-package flash	GPIO11, used for in-package flash
31	GPIO6, used for in-package flash	GPIO6, used for in-package flash	GPIO6, used for in-package flash
32	GPIO7, used for in-package flash	GPIO7, can be used freely	GPIO7, can be used freely
33	GPIO8, used for in-package flash	GPIO8, can be used freely	GPIO8, can be used freely
34, 38, 39, 42	GPIO, can be used freely	GPIO, can be used freely	GPIO, can be used freely
35	GPIO18, can be used freely	NC	NC
36	GPIO23, can be used freely	NC	NC
40	U0RXD	U0RXD	U0RXD
41	U0TXD	U0TXD	U0TXD

3 Boot Configurations

Note:
The content below is excerpted from *ESP32 Series Datasheet* > Section *Boot Configurations*.

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

- **Chip boot mode**
  - Strapping pin: GPIO0 and GPIO2
- **Internal LDO (VDD_SDIO) Voltage**
  - Strapping pin: MTDI
  - eFuse bit: EFUSE_SDIO_FORCE and EFUSE_SDIO_TIEH
- **U0TXD printing**
  - Strapping pin: MTDO
- **Timing of SDIO Slave**
  - Strapping pin: MTDO and GPIO5
- **JTAG signal source**
  - eFuse bit: EFUSE_DISABLE_JTAG

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 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 8: Default Configuration of Strapping Pins

Strapping Pin	Default Configuration	Bit Value
GPIO0	Pull-up	1
GPIO2	Pull-down	0
MTDI	Pull-down	0
MTDO	Pull-up	1
GPIO5	Pull-up	1

To change the bit values, the strapping pins should be connected to external pull-down/pull-up resistances. If the ESP32 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 9 and Figure 5.

Table 9: 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.	1

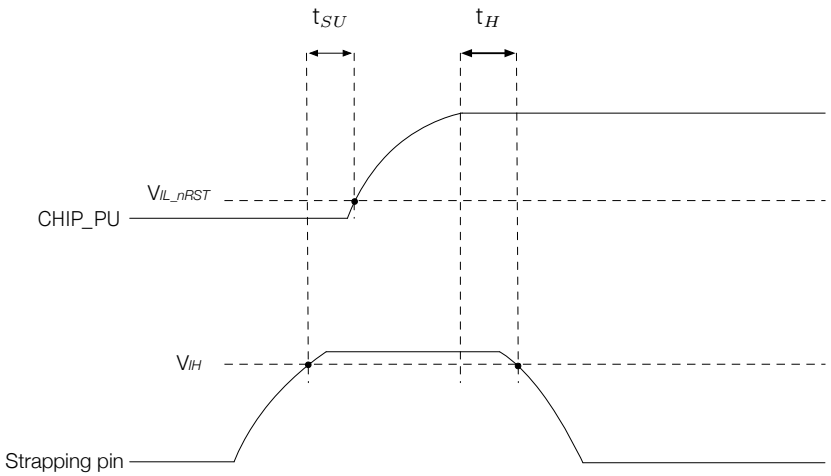


Figure 5: Visualization of Timing Parameters for the Strapping Pins

3.1 Chip Boot Mode Control

GPIO0 and GPIO2 control the boot mode after the reset is released. See Table 10 *Chip Boot Mode Control*.

Table 10: Chip Boot Mode Control

Boot Mode	GPIO0	GPIO2
SPI Boot Mode	**1**	Any value
Joint Download Boot Mode [2]	0	0

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

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

- SDIO Download Boot
- UART Download Boot

In Joint Download Boot mode, the detailed boot flow of the chip is put below 6.

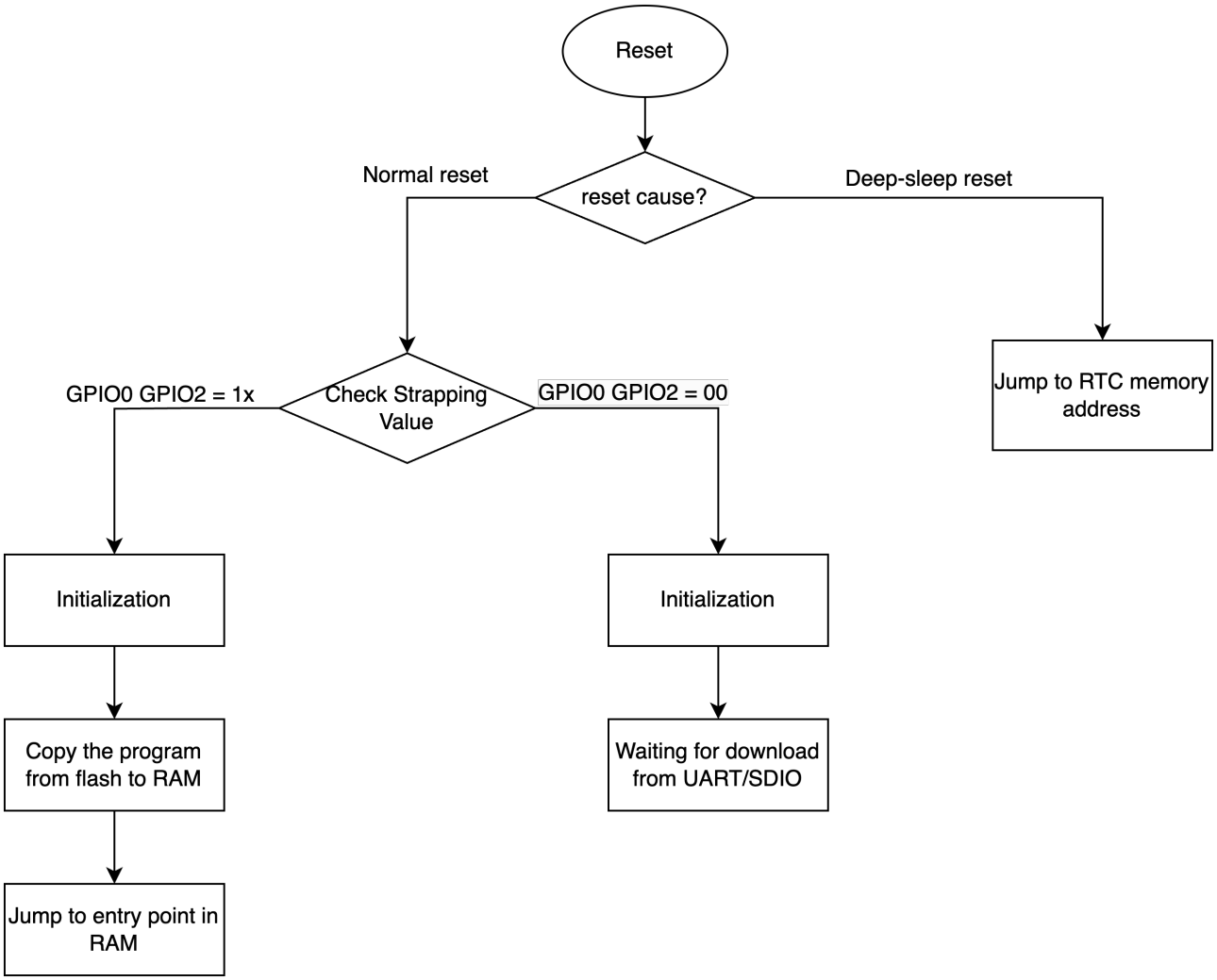


Figure 6: Chip Boot Flow

uart_download_dis controls boot mode behaviors:

It permanently disables Download Boot mode when uart_download_dis is set to 1 (valid only for ESP32 chip revisions v3.0 and higher).

3.2 Internal LDO (VDD_SDIO) Voltage Control

MTDI is used to select the VDD_SDIO power supply voltage at reset:

- MTDI = 0 (by default), VDD_SDIO pin is powered directly from VDD3P3_RTC. Typically this voltage is 3.3 V. For more information, see *ESP32 Series Datasheet* > Section *Power Scheme*.
- MTDI = 1, VDD_SDIO pin is powered from internal 1.8 V LDO.

This functionality can be overridden by setting EFUSE_SDIO_FORCE to 1, in which case the EFUSE_SDIO_TIEH determines the VDD_SDIO voltage:

- EFUSE_SDIO_TIEH = 0, VDD_SDIO connects to 1.8 V LDO.
- EFUSE_SPI_TIEH = 1, VDD_SDIO connects to VDD3P3_RTC.

3.3 U0TXD Printing Control

During booting, the strapping pin MTDO can be used to control the U0TXD Printing, as Table 11 shows.

Table 11: U0TXD Printing Control

U0TXD Printing Control	MTDO
Enabled [1]	**1**
Disabled	0

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

3.4 Timing Control of SDIO Slave

The strapping pin MTDO and GPIO5 can be used to control the timing of SDIO slave, see Table 12 *Timing Control of SDIO Slave*.

Table 12: Timing Control of SDIO Slave

Edge behavior	MTDO	GPIO5
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] **Bold** marks the default value and configuration.

3.5 JTAG Signal Source Control

If EFUSE_DISABLE_JTAG is set to 1, the source of JTAG signals can be disabled.

4 Peripherals

4.1 Peripheral Overview

ESP32-PICO integrates a rich set of peripherals including SPI, I2S, UART, I2C, pulse count controller, TWAI®, ADC, DAC, touch sensor, etc.

To learn more about on-chip components, please refer to *ESP32 Series Datasheet* > Section *Functional Description*.

Note:
The ESP32-PICO series of variants are similar to ESP32. The content below is sourced from *ESP32 Series Datasheet* > Section *Functional Description*. Some information may not be applicable to ESP32-PICO as not all the IO signals are exposed on the SiP.
To learn more about peripheral signals, please refer to *ESP32 Technical Reference Manual* > Section *Peripheral Signal List*.

4.2 Digital Peripherals

4.2.1 General Purpose Input / Output Interface (GPIO)

ESP32 has 34 GPIO pins which can be assigned various functions by programming the appropriate registers. There are several kinds of GPIOs: digital-only, analog-enabled, capacitive-touch-enabled, etc. Analog-enabled GPIOs and Capacitive-touch-enabled GPIOs can be configured as digital GPIOs.

Most of the digital GPIOs can be configured as internal pull-up or pull-down, or set to high impedance. When configured as an input, the input value can be read through the register. The input can also be set to edge-trigger or level-trigger to generate CPU interrupts. Most of the digital IO pins are bi-directional, non-inverting and tristate, including input and output buffers with tristate control. These pins can be multiplexed with other functions, such as the SDIO, UART, SPI, etc. (More details can be found in *ESP32 Series Datasheet* > Appendix, Table *IO_MUX*.) For low-power operations, the GPIOs can be set to hold their states.

For details, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations*, *ESP32 Series Datasheet* > *Appendix A – ESP32 Pin Lists* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.2 Serial Peripheral Interface (SPI)

ESP32 features three SPIs (SPI, HSPI and VSPI) in slave and master modes in 1-line full-duplex and 1/2/4-line half-duplex communication modes.

Features of General Purpose SPI (GP-SPI)

- Programmable data transfer length, in multiples of 1 byte
- Four-line full-duplex/half-duplex communication and three-line half-duplex communication support
- Master mode and slave mode

- Programmable CPOL and CPHA
- Programmable clock

For details, see *ESP32 Technical Reference Manual* > Chapter *SPI Controller*.

Pin Assignment

For SPI, the pins are multiplexed with GPIO6 ~ GPIO11 via the IO MUX. For HSPI, the pins are multiplexed with GPIO2, GPIO4, GPIO12 ~ GPIO15 via the IO MUX. For VSPI, the pins are multiplexed with GPIO5, GPIO18 ~ GPIO19, GPIO21 ~ GPIO23 via the IO MUX.

For more information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.3 Universal Asynchronous Receiver Transmitter (UART)

The UART in the ESP32 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 rate
- RAM shared by TX FIFOs and RX FIFOs
- Supports input baud rate self-check
- Support for various lengths of data bits and stop bits
- Parity bit support
- Asynchronous communication (RS232 and RS485) and IrDA support
- Supports DMA to communicate data in high speed
- Supports UART wake-up
- Supports both software and hardware flow control

For details, see *ESP32 Technical Reference Manual* > Chapter *UART Controller*.

Pin Assignment

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

For more information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.4 I2C Interface

ESP32 has two I2C bus interfaces which can serve as I2C master or slave, depending on the user's configuration.

Feature List

- Two I2C controllers: one in the main system and one in the low-power system
- Standard mode (100 Kbit/s)
- Fast mode (400 Kbit/s)
- Up to 5 MHz, yet constrained by SDA pull-up strength
- Support for 7-bit and 10-bit addressing, as well as dual address mode
- Supports continuous data transmission with disabled Serial Clock Line (SCL)
- Supports programmable digital noise filter

Users can program command registers to control I2C interfaces, so that they have more flexibility.

For details, see *ESP32 Technical Reference Manual* > Chapter *I2C Controller*.

Pin Assignment

For regular I2C, the pins used can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.5 I2S Interface

The I2S Controller in the ESP32 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
- A variety of audio standards supported
- Configurable high-precision output clock
- Supports PDM signal input and output
- Configurable data transmit and receive modes

For details, see *ESP32 Technical Reference Manual* > Chapter *I2S Controller*.

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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.6 Remote Control Peripheral

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

Feature List

- Eight channels for sending and receiving infrared remote control signals
- Independent transmission and reception capabilities for each channel
- Clock divider counter, state machine, and receiver for each RX channel
- Supports various infrared protocols

For details, see *ESP32 Technical Reference Manual* > Chapter *Remote Control Peripheral*.

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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.7 Pulse Counter Controller (PCNT)

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

Feature List

- Eight independent pulse counter units
- Each pulse counter unit has a 16-bit signed counter register and two channels
- 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 Technical Reference Manual* > Chapter *Pulse Count Controller*.

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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.8 LED PWM Controller

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

Feature List

- Sixteen independent PWM generators
- Maximum PWM duty cycle resolution of 20 bits
- Eight 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

For details, see *ESP32 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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.9 Motor Control PWM

The Pulse Width Modulation (PWM) controller can be used for driving digital motors and smart lights. The controller consists of PWM timers, the PWM operator and a dedicated capture sub-module. Each timer provides timing in synchronous or independent form, and each PWM operator generates a waveform for one PWM channel. The dedicated capture sub-module can accurately capture events with external timing.

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
  - A hardware sync can trigger a reload on the PWM timer with a phase register. It will also trigger the prescaler' restart, so that the timer's clock can also be synced, 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
- 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
- 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

For details, see *ESP32 Technical Reference Manual* > Chapter *Motor Control PWM*.

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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.10 SD/SDIO/MMC Host Controller

An SD/SDIO/MMC host controller is available on ESP32.

Feature List

- Supports two external cards
- Supports SD Memory Card standard: version 3.0 and version 3.01)
- Supports SDIO Version 3.0
- Supports Consumer Electronics Advanced Transport Architecture (CE-ATA Version 1.1)
- Supports Multimedia Cards (MMC version 4.41, eMMC version 4.5 and version 4.51)

The controller allows up to 80 MHz clock output in three different data-bus modes: 1-bit, 4-bit, and 8-bit modes. It supports two SD/SDIO/MMC4.41 cards in a 4-bit data-bus mode. It also supports one SD card operating at 1.8 V.

For details, see *ESP32 Technical Reference Manual* > Chapter *SD/MMC Host Controller*.

Pin Assignment

The pins for SD/SDIO/MMC Host Controller are multiplexed with GPIO2, GPIO4, GPIO6 ~ GPIO15 via IO MUX.

For more information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.11 SDIO/SPI Slave Controller

ESP32 integrates an SD device interface that conforms to the industry-standard SDIO Card Specification Version 2.0, and allows a host controller to access the SoC, using the SDIO bus interface and protocol. ESP32 acts as the slave on the SDIO bus. The host can access the SDIO-interface registers directly and can access shared memory via a DMA engine, thus maximizing performance without engaging the processor cores.

Feature List

The SDIO/SPI slave controller supports the following features:

- SPI, 1-bit SDIO, and 4-bit SDIO transfer modes over the full clock range from 0 to 50 MHz
- Configurable sampling and driving clock edge
- Special registers for direct access by host
- Interrupts to host for initiating data transfer
- Automatic loading of SDIO bus data and automatic discarding of padding data
- Block size of up to 512 bytes
- Interrupt vectors between the host and the slave, allowing both to interrupt each other
- Supports DMA for data transfer

For details, see *ESP32 Technical Reference Manual* > Chapter *SDIO Slave Controller*.

Pin Assignment

The pins for SDIO/SPI Slave Controller are multiplexed with GPIO2, GPIO4, GPIO6 ~ GPIO15 via IO MUX.

For more information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.12 TWAI® Controller

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 25 Kbit/s to 1 Mbit/s in chip revision v0.0/v1.0/v1.1
  - From 12.5 Kbit/s to 1 Mbit/s in chip revision v3.0/v3.1
- Multiple modes of operation: Normal, Listen Only, and Self-Test
- 64-byte receive FIFO
- Special transmissions: single-shot transmissions and self reception
- Acceptance filter (single and dual filter modes)
- Error detection and handling: error counters, configurable error interrupt threshold, error code capture, arbitration lost capture

For details, see *ESP32 Technical Reference Manual* > Chapter *Two-wire Automotive Interface (TWAI)*.

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 *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.2.13 Ethernet MAC Interface

An IEEE-802.3-2008-compliant Media Access Controller (MAC) is provided for Ethernet LAN communications. ESP32 requires an external physical interface device (PHY) to connect to the physical LAN bus (twisted-pair, fiber, etc.). The PHY is connected to ESP32 through 17 signals of MII or nine signals of RMII.

Feature List

- 10 Mbps and 100 Mbps rates
- Dedicated DMA controller allowing high-speed transfer between the dedicated SRAM and Ethernet MAC
- Tagged MAC frame (VLAN support)
- Half-duplex (CSMA/CD) and full-duplex operation
- MAC control sublayer (control frames)
- 32-bit CRC generation and removal
- Several address-filtering modes for physical and multicast address (multicast and group addresses)
- 32-bit status code for each transmitted or received frame
- Internal FIFOs to buffer transmit and receive frames. The transmit FIFO and the receive FIFO are both 512 words (32-bit)
- Hardware PTP (Precision Time Protocol) in accordance with IEEE 1588 2008 (PTP V2)
- 25 MHz/50 MHz clock output

For details, see *ESP32 Technical Reference Manual* > Chapter *Ethernet Media Access Controller (MAC)*.

Pin Assignment

For information about the pin assignment of Ethernet MAC Interface, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.3 Analog Peripherals

4.3.1 Analog-to-Digital Converter (ADC)

ESP32 integrates two 12-bit SAR ADCs and supports measurements on 18 channels (analog-enabled pins). The ULP coprocessor in ESP32 is also designed to measure voltage, while operating in the sleep mode, which enables low-power consumption. The CPU can be woken up by a threshold setting and/or via other triggers.

Table 13 describes the ADC characteristics.

Table 13: ADC Characteristics

Parameter	Description	Min	Max	Unit
DNL (Differential nonlinearity)	RTC controller; ADC connected to an external 100 nF capacitor; DC signal input; ambient temperature at 25 °C; Wi-Fi&Bluetooth off	–7	7	LSB
INL (Integral nonlinearity)		–12	12	LSB
Sampling rate	RTC controller	—	200	ksps
	DIG controller	—	2	Msps

Notes:

- When atten = 3 and the measurement result is above 3000 (voltage at approx. 2450 mV), the ADC accuracy will be worse than described in the table above.
- To get better DNL results, users can take multiple sampling tests with a filter, or calculate the average value.
- The input voltage range of GPIO pins within VDD3P3_RTC domain should strictly follow the DC characteristics provided in Table 18. Otherwise, measurement errors may be introduced, and chip performance may be affected.

By default, there are ±6% differences in measured results between chips. ESP-IDF provides couple of calibration methods for ADC1. Results after calibration using eFuse Vref value are shown in Table 14. For higher accuracy, users may apply other calibration methods provided in ESP-IDF, or implement their own.

Table 14: ADC Calibration Results

Parameter	Description	Min	Max	Unit
Total error	Atten = 0, effective measurement range of 100 ~ 950 mV	–23	23	mV
	Atten = 1, effective measurement range of 100 ~ 1250 mV	–30	30	mV
	Atten = 2, effective measurement range of 150 ~ 1750 mV	–40	40	mV
	Atten = 3, effective measurement range of 150 ~ 2450 mV	–60	60	mV

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Pin Assignment

With appropriate settings, the ADCs can be configured to measure voltage on 18 pins maximum. For detailed information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.3.2 Digital-to-Analog Converter (DAC)

Two 8-bit DAC channels can be used to convert two digital signals into two analog voltage signal outputs. The design structure is composed of integrated resistor strings and a buffer. This dual DAC supports power supply as input voltage reference. The two DAC channels can also support independent conversions.

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Pin Assignment

The DAC can be configured by GPIO 25 and GPIO 26. For detailed information about the pin assignment, see *ESP32 Series Datasheet* > Section *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.3.3 Touch Sensor

ESP32 has 10 capacitive-sensing GPIOs, which detect variations induced by touching or approaching the GPIOs with a finger or other objects. The low-noise nature of the design and the high sensitivity of the circuit allow relatively small pads to be used. Arrays of pads can also be used, so that a larger area or more points can be detected.

Pin Assignment

The 10 capacitive-sensing GPIOs are listed in Table 15.

Table 15: Capacitive-Sensing GPIOs Available on ESP32

Capacitive-Sensing Signal Name	Pin Name
T0	GPIO4
T1	GPIO0
T2	GPIO2
T3	MTDO
T4	MTCK
T5	MTDI
T6	MTMS
T7	GPIO27
T8	32K_XN
T9	32K_XP

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Note:
ESP32 Touch Sensor has not passed the Conducted Susceptibility (CS) test for now, and thus has limited application scenarios.

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stresses above those listed in Table 16 *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 Power Supply Characteristics is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Table 16: Absolute Maximum Ratings

Parameter	Description	Min	Max	Unit
VDDA, VDD3P3, VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO[1]	Allowed input voltage	−0.3	3.6	V
I_{output}[2]	Cumulative IO output current	—	1100	mA
T_{STORE}	Storage temperature	−40	85	°C

[1] For IO's power domain, please see *ESP32 Series Datasheet* > Appendix *IO MUX*.

[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 Power Supply Characteristics

Table 17: Recommended Power Supply Characteristics

Parameter	Description	Min	Typ	Max	Unit
VDDA, VDD3P3, VDD3P3_RTC[1], VDD_SDIO[2]	Recommended input voltage	3.0	3.3	3.6	V
VDD3P3_CPU	Recommended input voltage	1.8	3.3	3.6	V
I_{VDD}	Cumulative input current	0.5	—	—	A
T	Operating ambient temperature	−40	—	85	°C

[1] When writing eFuse, VDD3P3_RTC should be at least 3.3 V.

[2] VDD_SDIO:

- VDD_SDIO is powered by VDD3P3_RTC via 6 Ω resistor for 3.3 V flash/PSRAM, therefore, there will be some voltage drop from VDD3P3_RTC.
- VDD_SDIO can also be driven by an external power supply.

5.3 DC Characteristics (3.3 V, 25 °C)

Table 18: DC Characteristics (3.3 V, 25 °C)

Symbol	Parameter		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
V_{OH}[4]	High-level output voltage		0.8 × VDD[1]	—	—	V
V_{OL}[4]	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, output drive strength set to the maximum)	VDD3P3_CPU power domain [1, 2]	—	40	—	mA
		VDD3P3_RTC power domain [1, 2]	—	40	—	mA
		VDD_SDIO power domain [1, 3]	—	20	—	mA
I_{OL}	Low-level sink current (VDD[1] = 3.3 V, V_{OL} = 0.495 V, output drive strength set to the maximum)		—	28	—	mA
R_{PU}	Resistance of internal pull-up resistor		—	45	—	kΩ
R_{PD}	Resistance of internal pull-down resistor		—	45	—	kΩ
V_{IL_nRST}	Low-level input voltage of CHIP_PU to power off the chip		—	—	0.6	V

[1] VDD is the I/O voltage for a particular power domain of pins. For IO's power domain, please see *ESP32 Series Datasheet* > Appendix *IO MUX*.

[2] For VDD3P3_CPU and VDD3P3_RTC power domain, per-pin current sourced in the same domain is gradually reduced from around 40 mA to around 29 mA, V_{OH}>=2.64 V, as the number of current-source pins increases.

[3] Pins occupied by flash and/or PSRAM in the VDD_SDIO power domain were excluded from the test.

[4] V_{OH} and V_{OL} are measured using high-impedance load.

5.4 Current Consumption Characteristics

5.4.1 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 19: 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 @ 19.5 dBm	370
		802.11g, 54 Mbps, OFDM @ 14 dBm	270
		802.11n, HT20, MCS7 @ 13 dBm	250
		802.11n, HT40, MCS7 @ 13 dBm	205
	RX	802.11b/g/n, HT20	113
		802.11n, HT40	120

5.4.2 Current Consumption in Other Modes

Table 20: Current Consumption Depending on Work Modes

Work mode	Description		Current consumption (Typ)
Modem-sleep[1, 2]	The CPU is powered on[3]	240 MHz	30-68 mA
		160 MHz	27-44 mA
		Normal speed: 80 MHz	20-31 mA
Light-sleep	—		0.8 mA
Deep-sleep	The ULP coprocessor is powered on[4]		150 μA
	ULP sensor-monitored pattern[5]		100 μA @1% duty
	RTC timer + RTC memory		10 μA
	RTC timer only		5 μA
Power off	CHIP_PU is set to low level, the chip is powered off		1 μA

[1] The current consumption figures in Modem-sleep mode are for cases where the CPU is powered on and the cache idle.

[2] When Wi-Fi is enabled, the chip switches between Active and Modem-sleep modes. Therefore, current consumption changes accordingly.

[3] In Modem-sleep mode, the CPU frequency changes automatically. The frequency depends on the CPU load and the peripherals used.

[4] During Deep-sleep, when the ULP coprocessor is powered on, peripherals such as GPIO and RTC I2C are able to operate.

[5] The "ULP sensor-monitored pattern" refers to the mode where the ULP coprocessor or the sensor works periodically. When ADC works with a duty cycle of 1%, the typical current consumption is 100 μA.

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 Wi-Fi Radio (2.4 GHz)

Table 21: Wi-Fi RF Characteristics

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

6.1.1 Wi-Fi RF Transmitter (TX) Characteristics

Table 22: TX Power with Spectral Mask and EVM Meeting 802.11 Standards

Rate	Min (dBm)	Typ (dBm)	Max (dBm)
802.11b, 1 Mbps, DSSS	–	19.5	–
802.11b, 11 Mbps, CCK	–	19.5	–
802.11g, 6 Mbps, OFDM	–	18.0	–
802.11g, 54 Mbps, OFDM	–	14.0	–
802.11n, HT20, MCS0	–	18.0	–
802.11n, HT20, MCS7	–	13.0	–
802.11n, HT40, MCS0	–	18.0	–
802.11n, HT40, MCS7	–	13.0	–

Table 23: TX EVM Test[1]

Rate	Min (dB)	Typ (dB)	Limit (dB)
802.11b, 1 Mbps, DSSS	–	-26.5	-10.0
802.11b, 11 Mbps, CCK	–	-26.5	-10.0
802.11g, 6 Mbps, OFDM	–	-24.0	-5.0
802.11g, 54 Mbps, OFDM	–	-30.0	-25.0

Cont'd on next page

Table 23 – cont'd from previous page

Rate	Min (dB)	Typ (dB)	Limit (dB)
802.11n, HT20, MCS0	—	–24.0	–5.0
802.11n, HT20, MCS7	—	–30.5	–27.0
802.11n, HT40, MCS0	—	–24.0	–5.0
802.11n, HT40, MCS7	—	–30.5	–27.0

[1] EVM is measured at the corresponding typical TX power provided in Table *22 Wi-Fi RF Transmitter (TX) Characteristics* 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.

Table 24: RX Sensitivity

Rate	Min (dBm)	Typ (dBm)	Max (dBm)
802.11b, 1 Mbps, DSSS	—	–97.0	—
802.11b, 2 Mbps, DSSS	—	–94.0	—
802.11b, 5.5 Mbps, CCK	—	–92.0	—
802.11b, 11 Mbps, CCK	—	–88.0	—
802.11g, 6 Mbps, OFDM	—	–93.0	—
802.11g, 9 Mbps, OFDM	—	–91.0	—
802.11g, 12 Mbps, OFDM	—	–89.0	—
802.11g, 18 Mbps, OFDM	—	–87.0	—
802.11g, 24 Mbps, OFDM	—	–84.0	—
802.11g, 36 Mbps, OFDM	—	–80.0	—
802.11g, 48 Mbps, OFDM	—	–77.0	—
802.11g, 54 Mbps, OFDM	—	–75.0	—
802.11n, HT20, MCS0	—	–92.0	—
802.11n, HT20, MCS1	—	–88.0	—
802.11n, HT20, MCS2	—	–86.0	—
802.11n, HT20, MCS3	—	–83.0	—
802.11n, HT20, MCS4	—	–80.0	—
802.11n, HT20, MCS5	—	–76.0	—
802.11n, HT20, MCS6	—	–74.0	—
802.11n, HT20, MCS7	—	–72.0	—
802.11n, HT40, MCS0	—	–89.0	—
802.11n, HT40, MCS1	—	–85.0	—
802.11n, HT40, MCS2	—	–83.0	—
802.11n, HT40, MCS3	—	–80.0	—
802.11n, HT40, MCS4	—	–76.0	—
802.11n, HT40, MCS5	—	–72.0	—

Cont'd on next page

Table 24 – cont'd from previous page

Rate	Min (dBm)	Typ (dBm)	Max (dBm)
802.11n, HT40, MCS6	—	–71.0	—
802.11n, HT40, MCS7	—	–69.0	—

Table 25: 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	—	0	—
802.11g, 54 Mbps, OFDM	—	–8	—
802.11n, HT20, MCS0	—	0	—
802.11n, HT20, MCS7	—	–8	—
802.11n, HT40, MCS0	—	0	—
802.11n, HT40, MCS7	—	–8	—

Table 26: RX Adjacent Channel Rejection

Rate	Min (dB)	Typ (dB)	Max (dB)
802.11b, 1 Mbps, DSSS	—	35	—
802.11b, 11 Mbps, CCK	—	35	—
802.11g, 6 Mbps, OFDM	—	27	—
802.11g, 54 Mbps, OFDM	—	13	—
802.11n, HT20, MCS0	—	27	—
802.11n, HT20, MCS7	—	12	—
802.11n, HT40, MCS0	—	16	—
802.11n, HT40, MCS7	—	7	—

6.2 Bluetooth Radio

6.2.1 Receiver – Basic Data Rate

Table 27: Receiver Characteristics – Basic Data Rate

Parameter	Description	Min	Typ	Max	Unit
Sensitivity @0.1% BER	—	—	–92	—	dBm
Maximum received signal @0.1% BER	—	0	—	—	dBm
Co-channel C/I	—	—	+7	—	dB
	F = F0 + 1 MHz	—	—	–6	dB

Cont'd on next page

Adjacent channel selectivity C/I

Table 27 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
	F = F0 – 1 MHz	—	—	–6	dB
	F = F0 + 2 MHz	—	—	–25	dB
	F = F0 – 2 MHz	—	—	–33	dB
	F = F0 + 3 MHz	—	—	–25	dB
	F = F0 – 3 MHz	—	—	–45	dB
Out-of-band blocking performance	30 MHz ~ 2000 MHz	–10	—	—	dBm
	2000 MHz ~ 2400 MHz	–27	—	—	dBm
	2500 MHz ~ 3000 MHz	–27	—	—	dBm
	3000 MHz ~ 12.5 GHz	–10	—	—	dBm
Intermodulation	—	–36	—	—	dBm

6.2.2 Transmitter – Basic Data Rate

Table 28: Transmitter Characteristics – Basic Data Rate

Parameter	Description	Min	Typ	Max	Unit
RF transmit power*	—	—	0	—	dBm
Gain control step	—	—	3	—	dB
RF power control range	—	–12	—	+9	dBm
+20 dB bandwidth	—	—	0.9	—	MHz
Adjacent channel transmit power	F = F0 ± 2 MHz	—	–55	—	dBm
	F = F0 ± 3 MHz	—	–55	—	dBm
	F = F0 ± > 3 MHz	—	–59	—	dBm
$\Delta f1_{avg}$	—	—	—	155	kHz
$\Delta f2_{max}$	—	127	—	—	kHz
$\Delta f2_{avg}/\Delta f1_{avg}$	—	—	0.92	—	—
ICFT	—	—	–7	—	kHz
Drift rate	—	—	0.7	—	kHz/50 μs
Drift (DH1)	—	—	6	—	kHz
Drift (DH5)	—	—	6	—	kHz

* There are a total of eight power levels from 0 to 7, and the transmit power ranges from –12 dBm to 9 dBm. When the power level rises by 1, the transmit power increases by 3 dB. Power level 4 is used by default and the corresponding transmit power is 0 dBm.

6.2.3 Receiver – Enhanced Data Rate

Table 29: Receiver Characteristics – Enhanced Data Rate

Parameter	Description	Min	Typ	Max	Unit
$\pi/4$ DQPSK					
Sensitivity @0.01% BER	—	--	–92	--	dBm
Maximum received signal @0.01% BER	—	—	0	—	dBm

Cont'd on next page

Table 29 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
Co-channel C/I	—	—	11	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	–7	—	dB
	F = F0 – 1 MHz	—	–7	—	dB
	F = F0 + 2 MHz	—	–25	—	dB
	F = F0 – 2 MHz	—	–35	—	dB
	F = F0 + 3 MHz	—	–25	—	dB
	F = F0 – 3 MHz	—	–45	—	dB
8DPSK					
Sensitivity @0.01% BER	—	—	–86	—	dBm
Maximum received signal @0.01% BER	—	—	–5	—	dBm
C/I c-channel	—	—	18	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	2	—	dB
	F = F0 – 1 MHz	—	2	—	dB
	F = F0 + 2 MHz	—	–25	—	dB
	F = F0 – 2 MHz	—	–25	—	dB
	F = F0 + 3 MHz	—	–25	—	dB
	F = F0 – 3 MHz	—	–38	—	dB

6.2.4 Transmitter – Enhanced Data Rate

Table 30: Transmitter Characteristics – Enhanced Data Rate

Parameter	Description	Min	Typ	Max	Unit
RF transmit power (see note under Table 28)	—	—	0	—	dBm
Gain control step	—	—	3	—	dB
RF power control range	—	–12	—	+9	dBm
$\pi/4$ DQPSK max w0	—	—	–0.72	—	kHz
$\pi/4$ DQPSK max wi	—	—	–6	—	kHz
$\pi/4$ DQPSK max |wi + w0|	—	—	–7.42	—	kHz
8DPSK max w0	—	—	0.7	—	kHz
8DPSK max wi	—	—	–9.6	—	kHz
8DPSK max |wi + w0|	—	—	–10	—	kHz
$\pi/4$ DQPSK modulation accuracy	RMS DEVM	—	4.28	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	13.3	—	%
8 DPSK modulation accuracy	RMS DEVM	—	5.8	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	14	—	%
In-band spurious emissions	F = F0 ± 1 MHz	—	–46	—	dBm
	F = F0 ± 2 MHz	—	–44	—	dBm
	F = F0 ± 3 MHz	—	–49	—	dBm
	F = F0 +/– > 3 MHz	—	—	–53	dBm

Cont'd on next page

Table 30 – cont'd from previous page

Parameter	Description	Min	Typ	Max	Unit
EDR differential phase coding	—	—	100	—	%

6.3 Bluetooth LE Radio

Table 31: Bluetooth LE RF Characteristics

Name	Description
Center frequency range of operating channel	2402 ~ 2480 MHz
RF transmit power range	−12.0 ~ 9.0 dBm

6.3.1 Bluetooth LE RF Transmitter (TX) Characteristics

Table 32: Bluetooth LE - Transmitter Characteristics

Parameter	Description	Min	Typ	Max	Unit
Carrier frequency offset and drift	Max. $\lvert f_n \rvert_{n=0,\,1,\,2,\,3,\,\ldots k}$	—	2.2	—	kHz
	Max. $\lvert f_0 - f_n \rvert_{n=2,\,3,\,4,\,\ldots k}$	—	1.3	—	kHz
	Max. $\lvert f_n - f_{n-5} \rvert_{n=6,\,7,\,8,\,\ldots k}$	—	1.5	—	kHz
	$\lvert f_1 - f_0 \rvert$	—	0.6	—	kHz
Modulation characteristics	$\Delta\, F1_{avg}$	—	247.5	—	kHz
	Min. $\Delta\, F2_{max}$ (for at least 99.9% of all $\Delta\, F2_{max}$)	—	206.0	—	kHz
	$\Delta\, F2_{avg} / \Delta\, F1_{avg}$	—	0.86	—	—
In-band emissions	± 2 MHz offset	—	−55	—	dBm
	± 3 MHz offset	—	−57	—	dBm
	> ± 3 MHz offset	—	−59	—	dBm

6.3.2 Bluetooth LE RF Receiver (RX) Characteristics

Table 33: Bluetooth LE - Receiver Characteristics

Parameter		Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER		—	—	−96.5	—	dBm
Maximum received signal @30.8% PER		—	—	5	—	dBm
C/I and receiver selectivity performance	Co-channel	F = F0 MHz	—	10	—	dB
	Adjacent channel	F = F0 + 1 MHz	—	2	—	dB
		F = F0 − 1 MHz	—	4	—	dB
		F = F0 + 2 MHz	—	−21	—	dB
		F = F0 − 2 MHz	—	−20	—	dB
		F = F0 + 3 MHz	—	−32	—	dB
		F = F0 − 3 MHz	—	−45	—	dB
		F ≥ F0 + 4 MHz	—	−29	—	dB

Cont'd on next page

Table 33 – cont'd from previous page

Parameter		Description	Min	Typ	Max	Unit
		F $\leq$ F0 – 4 MHz	—	–40	—	dB
	Image frequency	—	—	–29	—	dB
	Adjacent channel to image frequency	F = F_{image} + 1 MHz	—	–29	—	dB
		F = F_{image} – 1 MHz	—	–32	—	dB
Out-of-band blocking performance		30 MHz ~ 2000 MHz	—	–10	—	dBm
		2003 MHz ~ 2399 MHz	—	–27	—	dBm
		2484 MHz ~ 2997 MHz	—	–27	—	dBm
		3000 MHz ~ 12.75 GHz	—	–10	—	dBm
Intermodulation		—	—	–36	—	dBm

7 Schematics

This section provides the reference designs for the ESP32-PICO series variants.

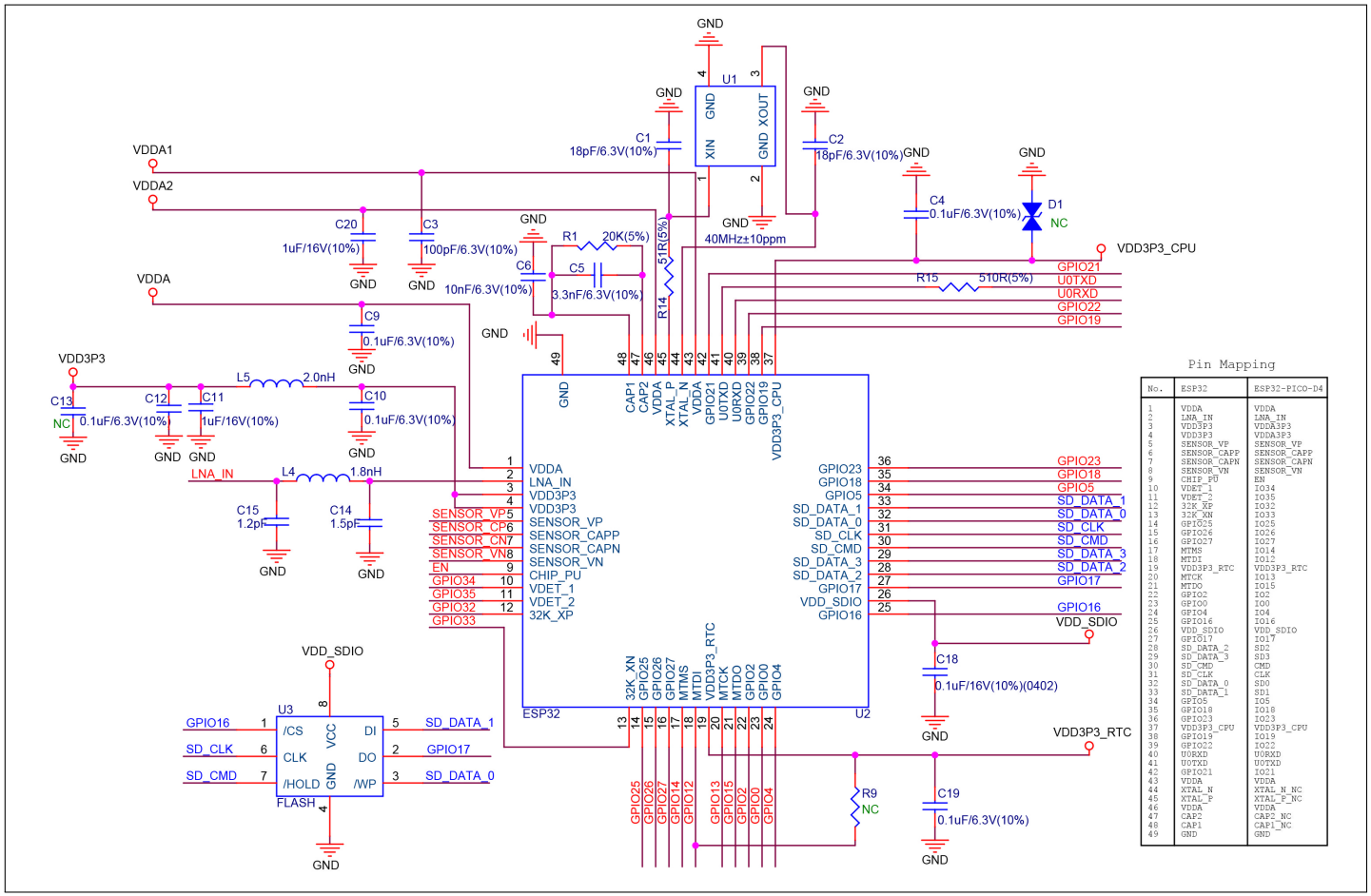


Figure 7: ESP32-PICO-D4 Schematics

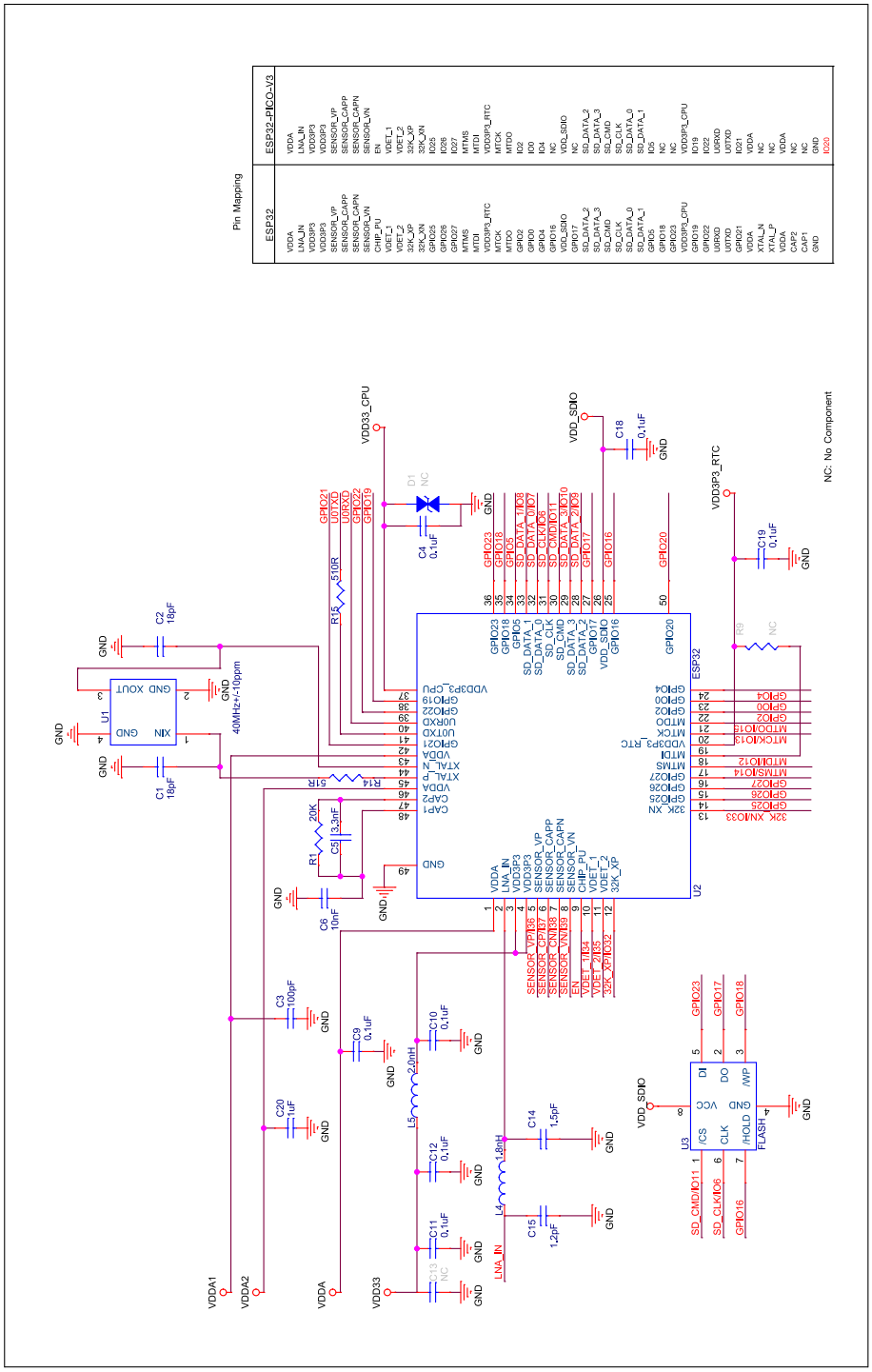


Figure 8: ESP32-PICO-V3 Schematics

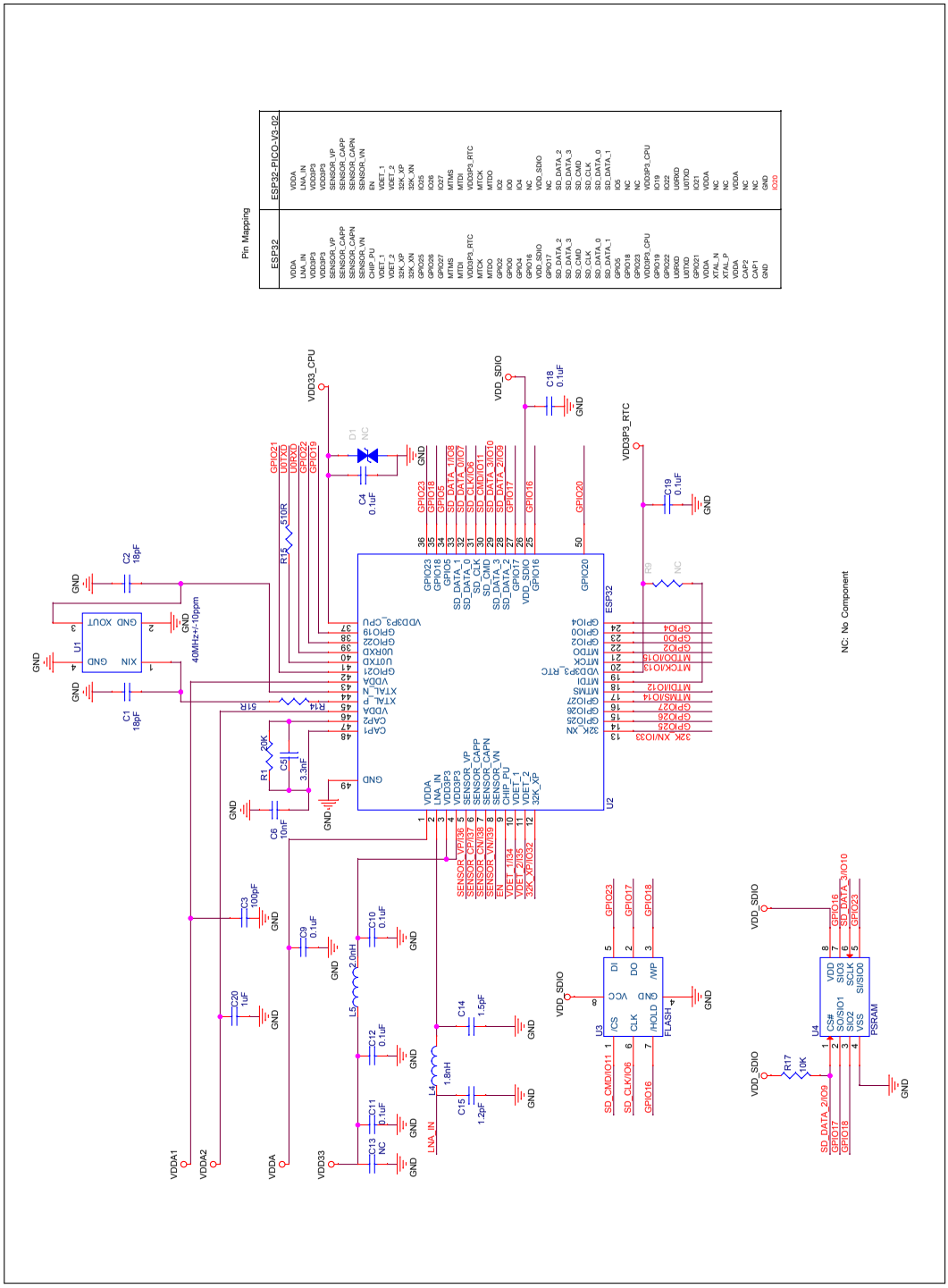
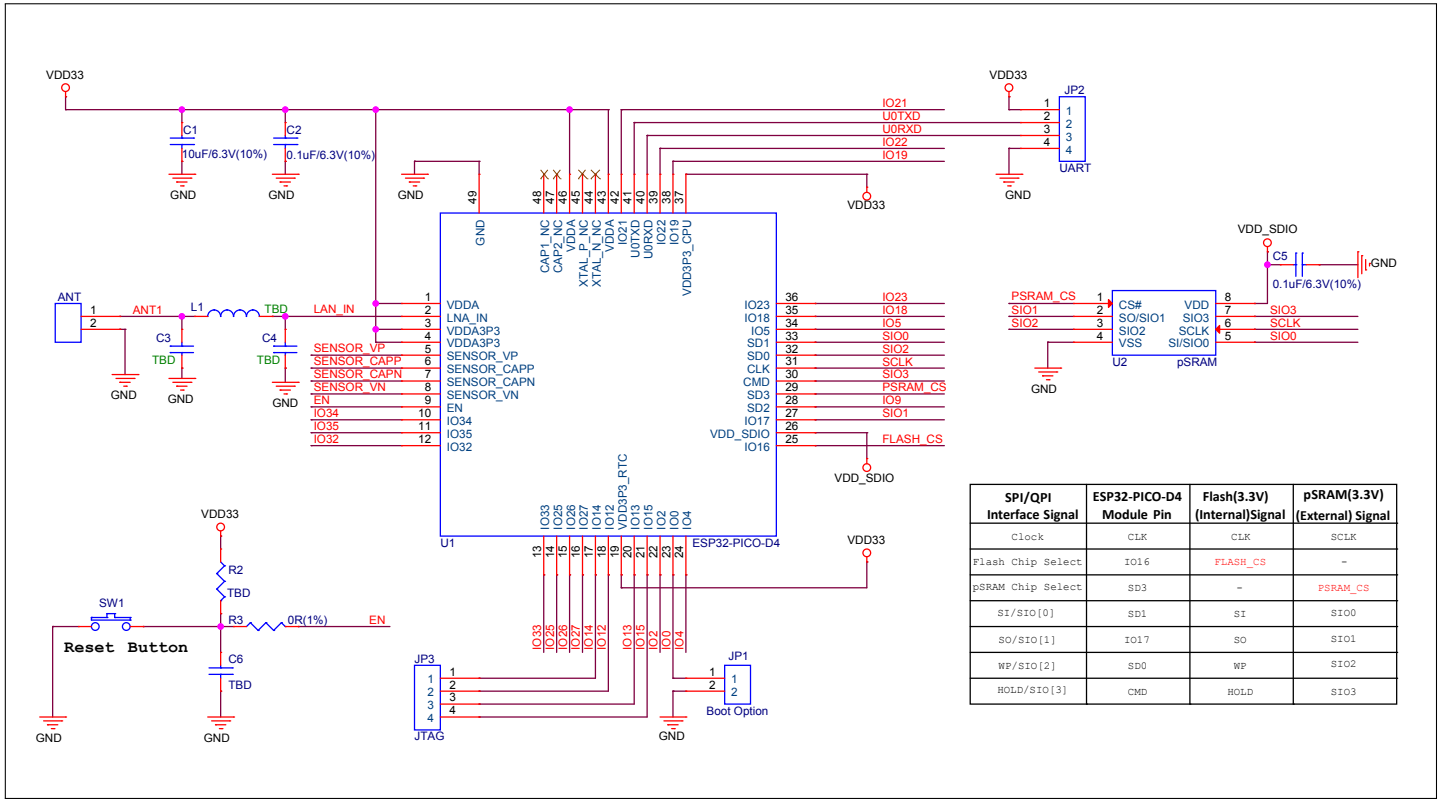


Figure 9: ESP32-PICO-V3-02 Schematics

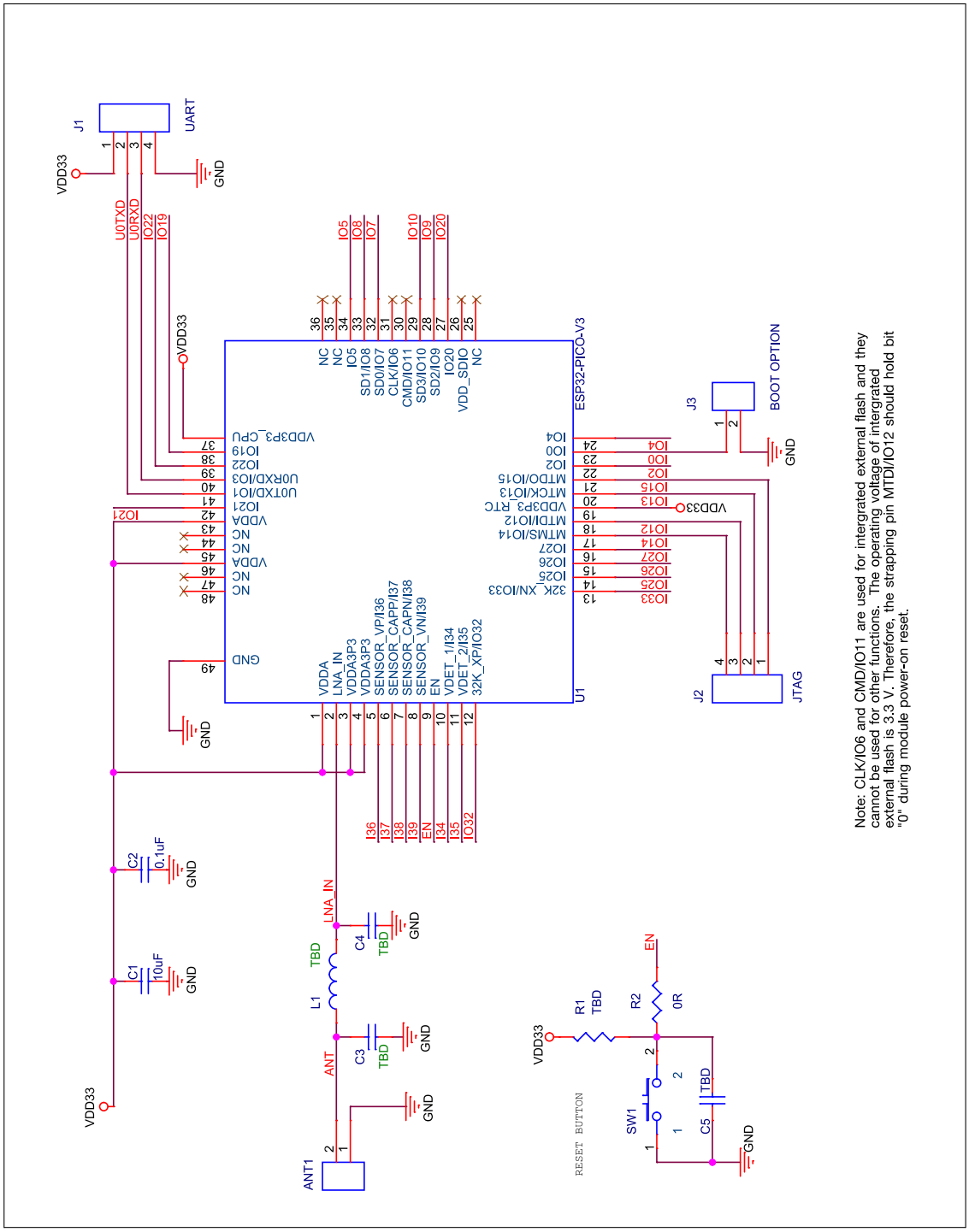
8 Peripheral Schematics

This is the typical application circuit of the ESP32-PICO connected with peripheral components (for example, power supply, antenna, reset button, JTAG interface, and UART interface).



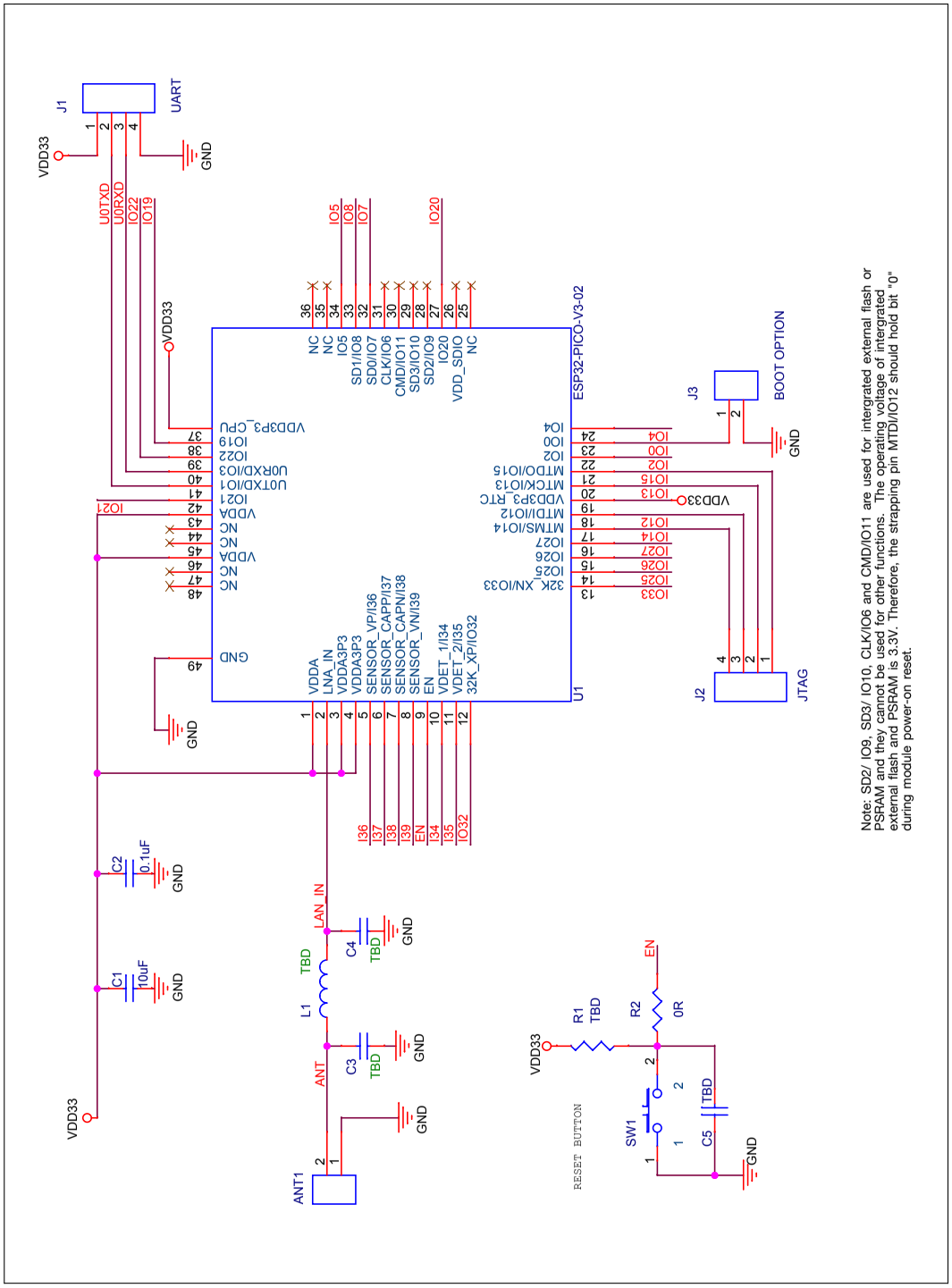
SPI/QPI Interface Signal	ESP32-PICO-D4 Module Pin	Flash(3.3V) (Internal)Signal	pSRAM(3.3V) (External) Signal
Clock	CLK	CLK	SCLK
Flash Chip Select	IO16	FLASH_CS	-
pSRAM Chip Select	SD3	-	PSRAM_CS
SI/SIO[0]	SD1	SI	SIO0
SO/SIO[1]	IO17	SO	SIO1
WP/SIO[2]	SD0	WP	SIO2
HOLD/SIO[3]	CMD	HOLD	SIO3

Figure 10: ESP32-PICO-D4 Peripheral Schematics



Note: CLK/IO6 and CMD/IO11 are used for intergrated external flash and they cannot be used for other functions. The operating voltage of intergrated external flash is 3.3 V. Therefore, the strapping pin MTDI/IO12 should hold bit "0" during module power-on reset.

Figure 11: ESP32-PICO-V3 Peripheral Schematics



Note: SD2/ IO9, SD3/ IO10, CLK/IO6 and CMD/IO11 are used for intergrated external flash or PSRAM and they cannot be used for other functions. The operating voltage of intergrated external flash and PSRAM is 3.3V. Therefore, the strapping pin MTDI/IO12 should hold bit "0" during module power-on reset.

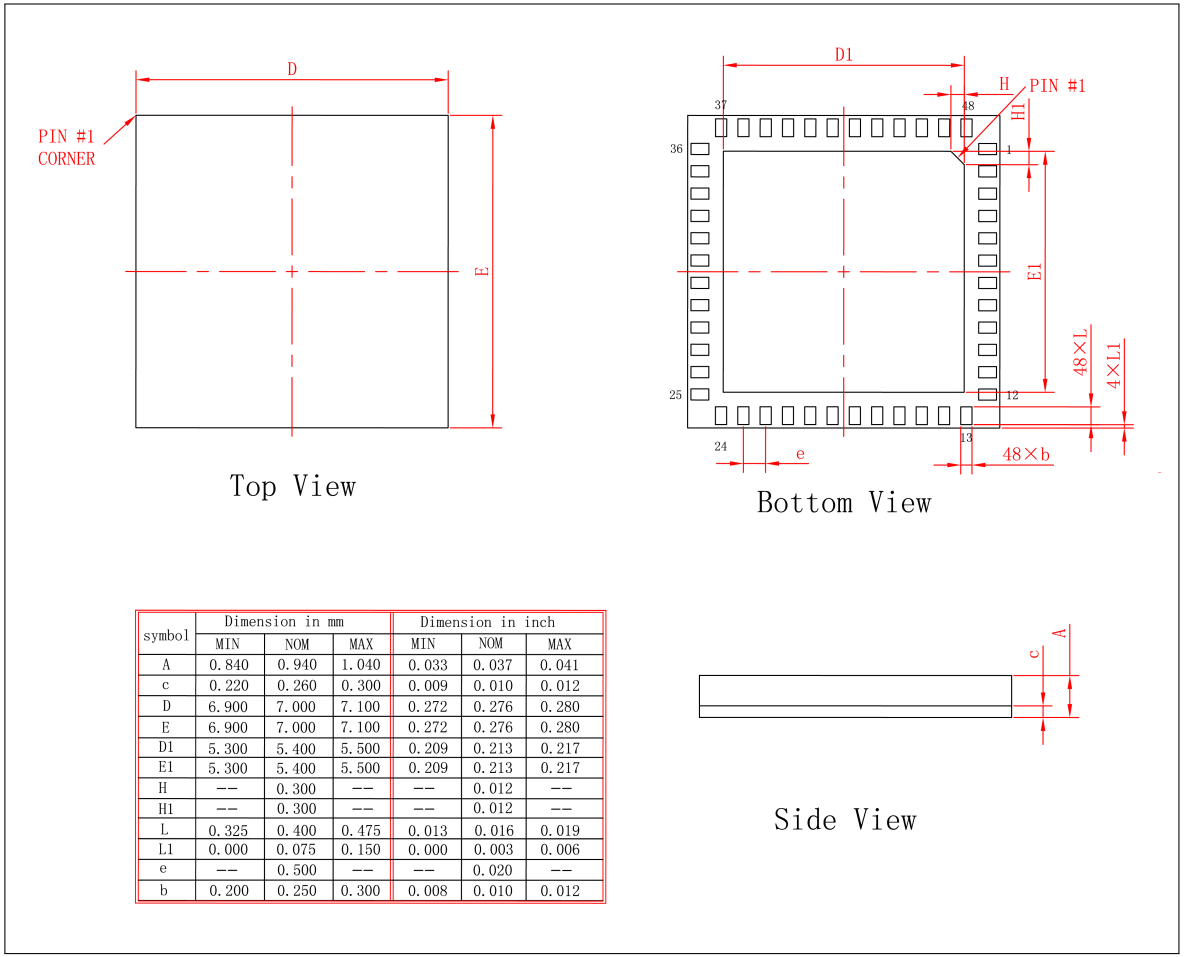
Figure 12: ESP32-PICO-V3-02 Peripheral Schematics

Note:

To ensure the power supply to the ESP32 chip during power-up, it is advised to add an RC delay circuit at the EN pin. The recommended setting for the RC delay circuit is usually R = 10 kΩ and C = 1 μF. However, specific parameters should be adjusted based on the power-up timing of the SiP and the power-up and reset sequence timing of the chip. For ESP32's power-up and reset sequence timing diagram, please refer to Section *Power Scheme* in *ESP32 Series Datasheet*.

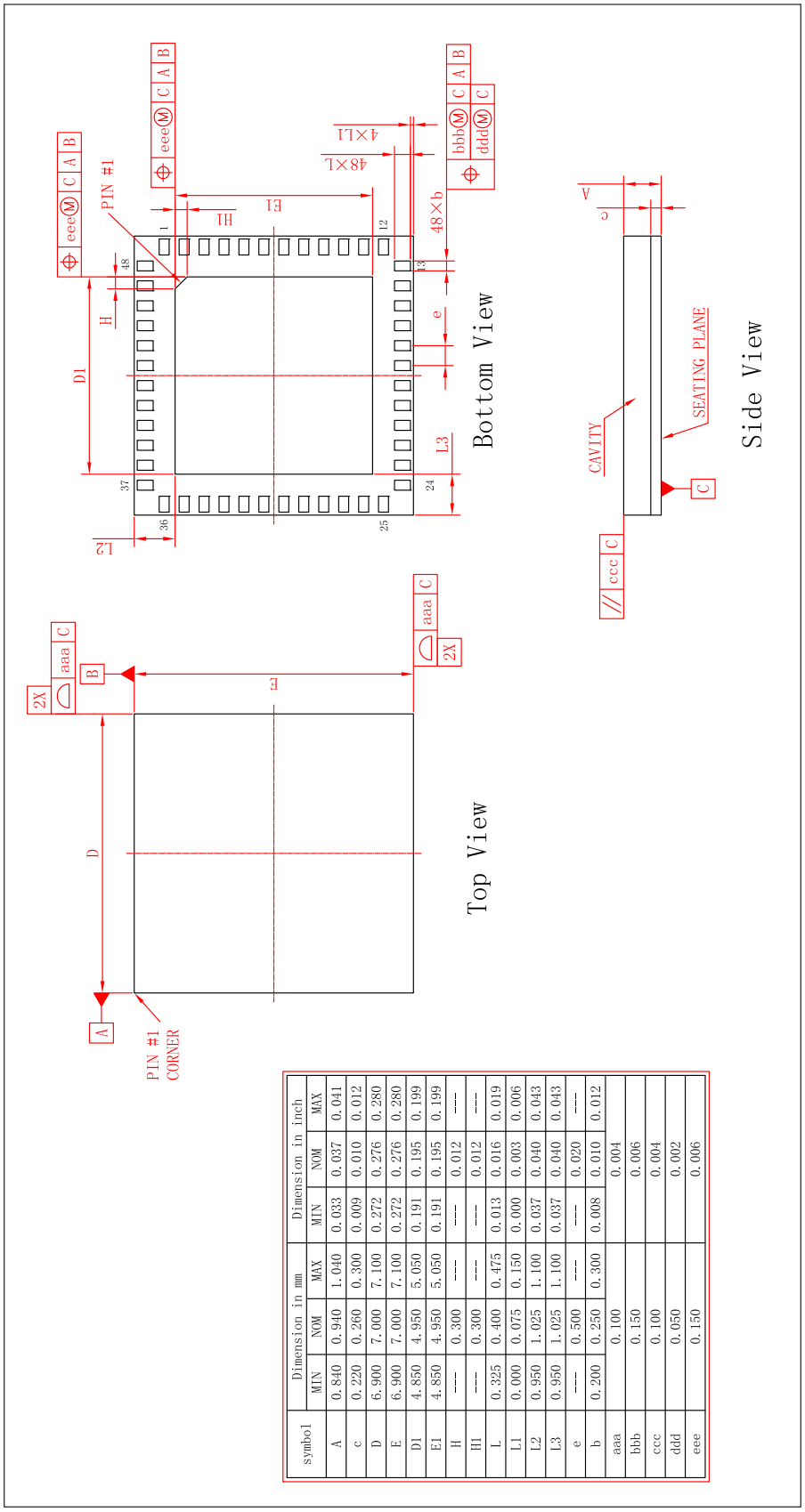
9 Packaging

- 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 pin layout figures in Section 2.1 ESP32-PICO-D4.



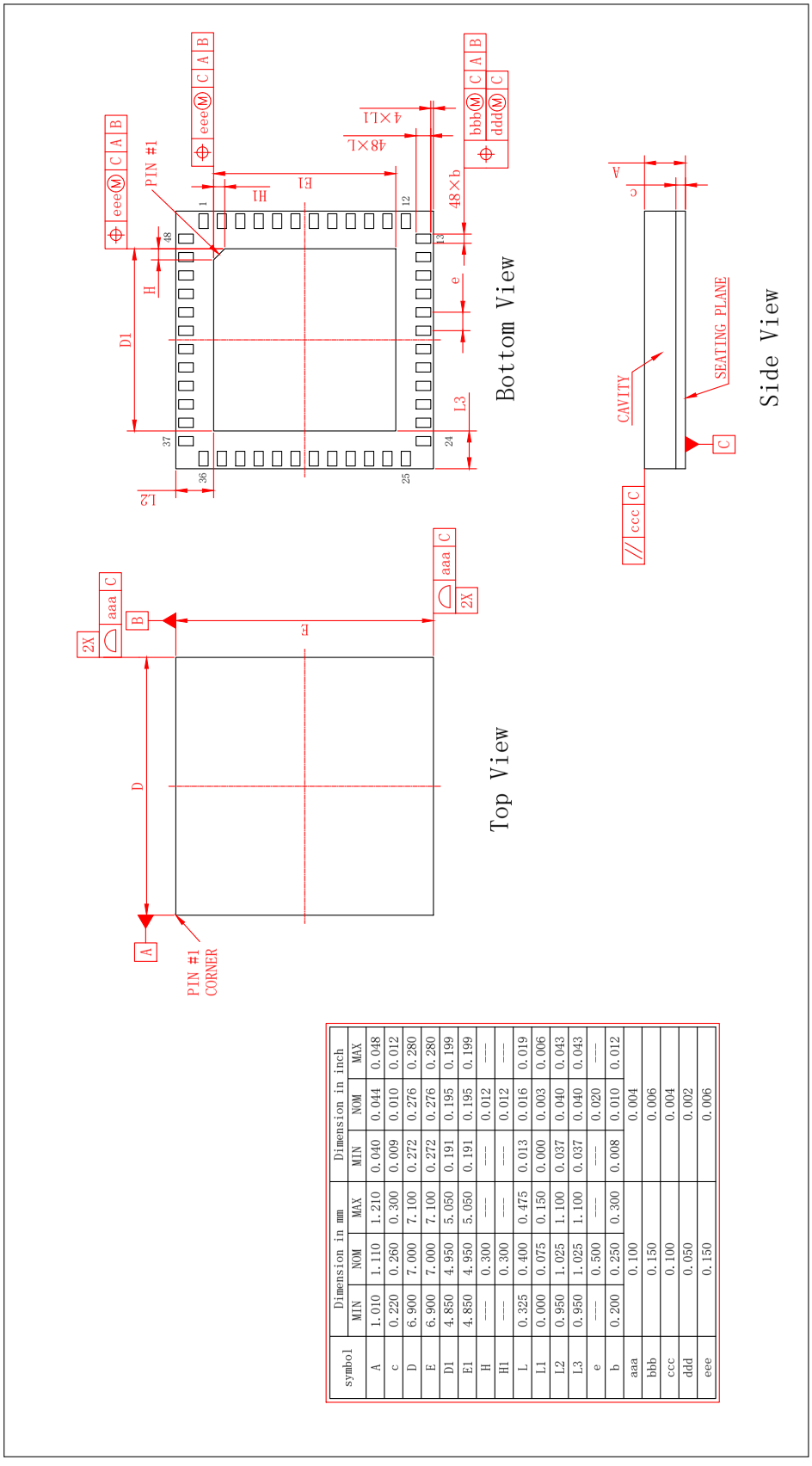
symbol	Dimension in mm			Dimension in inch		
	MIN	NOM	MAX	MIN	NOM	MAX
A	0.840	0.940	1.040	0.033	0.037	0.041
c	0.220	0.260	0.300	0.009	0.010	0.012
D	6.900	7.000	7.100	0.272	0.276	0.280
E	6.900	7.000	7.100	0.272	0.276	0.280
D1	5.300	5.400	5.500	0.209	0.213	0.217
E1	5.300	5.400	5.500	0.209	0.213	0.217
H	--	0.300	--	--	0.012	--
H1	--	0.300	--	--	0.012	--
L	0.325	0.400	0.475	0.013	0.016	0.019
L1	0.000	0.075	0.150	0.000	0.003	0.006
e	--	0.500	--	--	0.020	--
b	0.200	0.250	0.300	0.008	0.010	0.012

Figure 13: ESP32-PICO-D4 Package



symbol	Dimension in mm			Dimension in inch		
	MIN	NOM	MAX	MIN	NOM	MAX
A	0.840	0.940	1.040	0.033	0.037	0.041
c	0.220	0.260	0.300	0.009	0.010	0.012
D	6.900	7.000	7.100	0.272	0.276	0.280
E	6.900	7.000	7.100	0.272	0.276	0.280
D1	4.850	4.950	5.050	0.191	0.195	0.199
E1	4.850	4.950	5.050	0.191	0.195	0.199
H	----	0.300	----	----	0.012	----
H1	----	0.300	----	----	0.012	----
L	0.325	0.400	0.475	0.013	0.016	0.019
L1	0.000	0.075	0.150	0.000	0.003	0.006
L2	0.950	1.025	1.100	0.037	0.040	0.043
L3	0.950	1.025	1.100	0.037	0.040	0.043
e	----	0.500	----	----	0.020	----
b	0.200	0.250	0.300	0.008	0.010	0.012
aaa		0.100			0.004	
bbb		0.150			0.006	
ccc		0.100			0.004	
ddd		0.050			0.002	
eee		0.150			0.006	

Figure 14: ESP32-PICO-V3 Package



symbol	Dimension in mm			Dimension in inch		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.010	1.110	1.210	0.040	0.044	0.048
c	0.220	0.260	0.300	0.009	0.010	0.012
D	6.900	7.000	7.100	0.272	0.276	0.280
E	6.900	7.000	7.100	0.272	0.276	0.280
D1	4.850	4.950	5.050	0.191	0.195	0.199
E1	4.850	4.950	5.050	0.191	0.195	0.199
H	----	0.300	----	----	0.012	----
H1	----	0.300	----	----	0.012	----
L	0.325	0.400	0.475	0.013	0.016	0.019
L1	0.000	0.075	0.150	0.000	0.003	0.006
L2	0.950	1.025	1.100	0.037	0.040	0.043
L3	0.950	1.025	1.100	0.037	0.040	0.043
e	----	0.500	----	----	0.020	----
b	0.200	0.250	0.300	0.008	0.010	0.012
aaa		0.100			0.004	
bbb		0.150			0.006	
ccc		0.100			0.004	
ddd		0.050			0.002	
eee		0.150			0.006	

Figure 15: ESP32-PICO-V3-02 Package

10 PCB Land Pattern

This section provides the following resources for your reference:

- Figures for recommended PCB land patterns with all the dimensions needed for PCB design. See Figure 16 *ESP32-PICO PCB Land Pattern*.
- Source files of recommended PCB land patterns to measure dimensions not covered in Figure 16. You can view the source files for ESP32-PICO-D4, ESP32-PICO-V3, and ESP2-PICO-V3-02 with Autodesk Viewer.

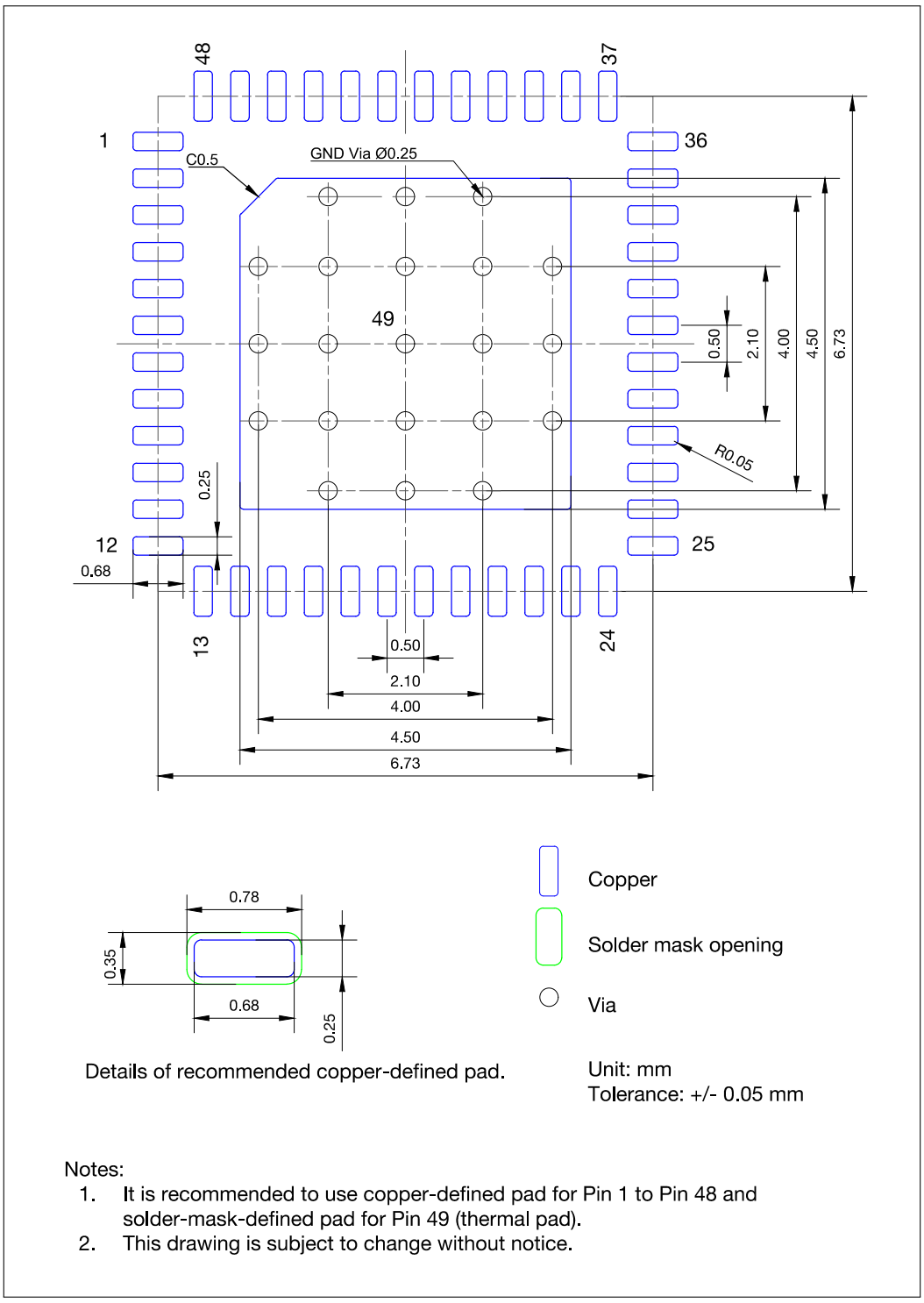


Figure 16: ESP32-PICO PCB Land Pattern

11 ESP32-PICO PCB Stencil

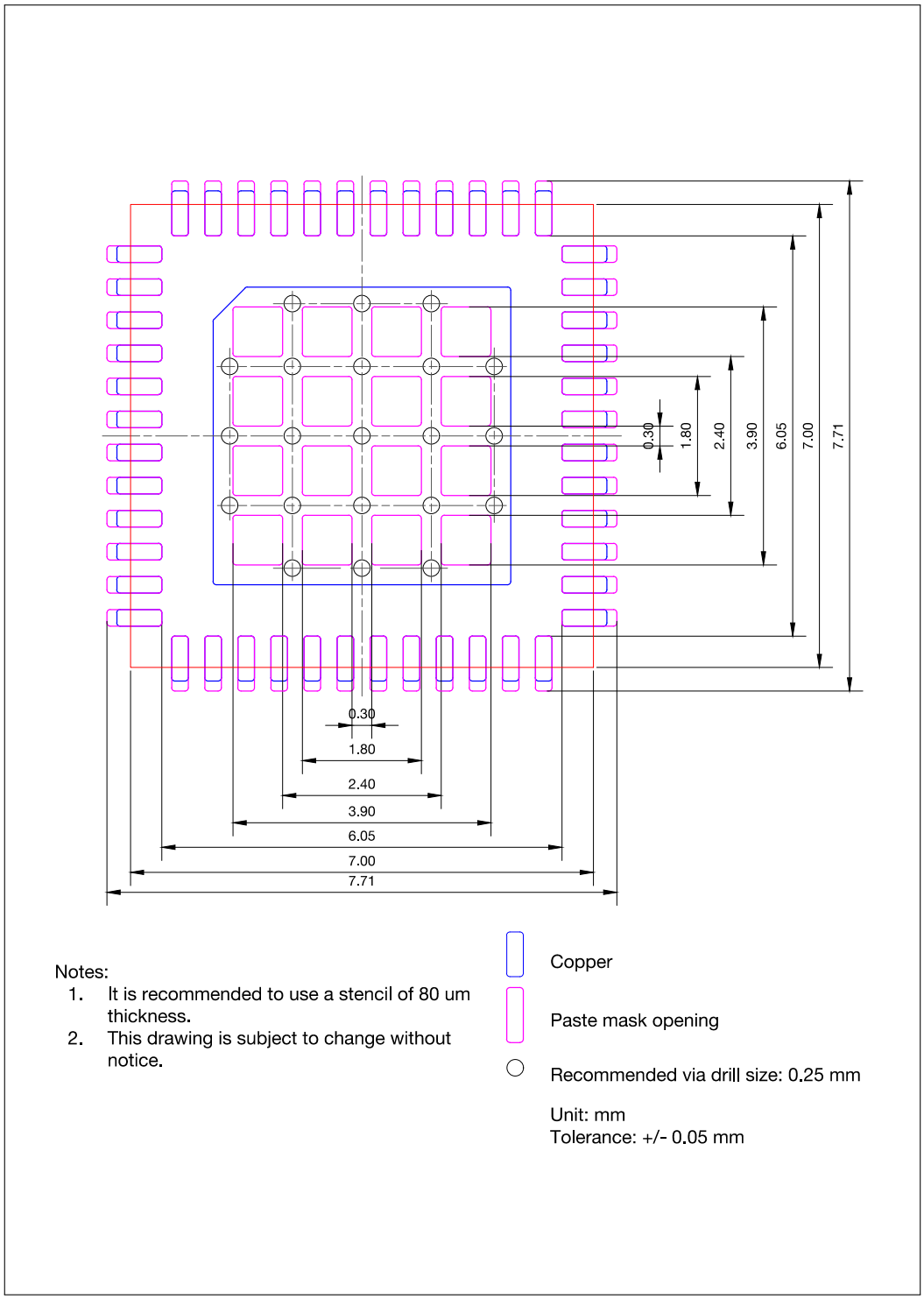


Figure 17: ESP32-PICO PCB STENCIL

12 Ultrasonic Vibration

Avoid exposing the device to vibration from ultrasonic equipment, such as ultrasonic welders or ultrasonic cleaners. This vibration may induce resonance in the crystal and lead to its malfunction or even failure. As a

consequence, **the device may stop working or its performance may deteriorate**.

13 Migration Guide

This section provides an overview of the software and hardware changes when migrating from the existing module design based on an older ESP32-PICO variant to the new design based on a newer ESP32-PICO variant.

13.1 Migrating from ESP32-PICO-D4 to ESP32-PICO-V3

ESP32-PICO-D4 is the first introduced ESP32-PICO variant with ESP32 chip revision v1.0 or v1.1. ESP32-PICO-V3 contains a newer ESP32 chip inside (v3.0 or v3.1). For information about possible hardware and software changes in ESP32 chip revision v3.0, refer to *ESP32 Chip Revision v3.0 User Guide* .

ESP32-PICO-D4 and ESP32-PICO-V3 are not 1:1 pin-compatible. For changes in pin layout and functions, refer to Section 2.4 Pin Compatibility Between ESP32-PICO Variants.

Both ESP32-PICO-D4 and ESP32-PICO-V3 are produced with in-package flash. ESP32-PICO-D4 can connect to external PSRAM. However, ESP32-PICO-V3 cannot connect to external PSRAM.

EMC compliance and RF performance tests should be repeated after a design is updated to use ESP32-PICO-V3.

13.2 Migrating from ESP32-PICO-V3 to ESP32-PICO-V3-02

ESP32-PICO-V3-02 is a memory upgrade of ESP32-PICO-V3. ESP32-PICO-V3-02 has both in-package flash and PSRAM, while ESP32-PICO-V3 has only flash. It is not possible for both variants to connect to external PSRAM.

ESP32-PICO-V3-02 is designed to be largely pin-compatible with ESP32-PICO-V3, and thus requires only minor changes during migration. For changes in pin layout and functions, refer to Section 2.4 Pin Compatibility Between ESP32-PICO Variants.

EMC compliance and RF performance tests should be repeated after a design is updated to use ESP32-PICO-V3-02.

13.3 Summary

As summarized in this migration guide, there are minimal or no hardware and software changes required when migrating to newer ESP32-PICO variants. If you have any problems with migration, please contact Espressif Technical Support.

Related Documentation and Resources

Related Documentation

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

Developer Zone

- ESP-IDF Programming Guide for ESP32 – 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 Series SoCs* – Browse through all ESP32 SoCs.
  https://espressif.com/en/products/socs?id=ESP32
- *ESP32 Series Modules* – Browse through all ESP32-based modules.
  https://espressif.com/en/products/modules?id=ESP32
- *ESP32 Series DevKits* – Browse through all ESP32-based devkits.
  https://espressif.com/en/products/devkits?id=ESP32
- *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
2025.01-17	v1.1	• Improved the formatting, structure, and wording in the following sections: – Updated Section Features – Updated Section 2 *Pins* – Updated Section 3 *Boot Configurations* (used to be named as "Strapping Pins") – Added Chapter 4 *Peripherals* • Fixed a typo about GPIO11 in Table 7 • Added Not Recommended for New Designs (NRND) label to ESP32-PICO-D4
2023-12-5	v1.0	Consolidated the three datasheets of ESP32-PICO variants into one. During the migration, some updates, improvements, and clarifications were made throughout the documentation. Major updates include: • Added Section 2.4 Pin Compatibility Between ESP32-PICO Variants • Added Section 12 Ultrasonic Vibration • Added Section 13 Migration Guide If you would like to check previous versions of the individual datasheets, please submit documentation feedback.

Disclaimer and Copyright Notice

www.espressif.com