

ESP32-S2

esp-dev-kits Documentation



Release master
Espressif Systems
Mar 06, 2026

Table of contents

Table of contents	i
1 ESP32-S2-DevKitC-1	3
1.1 ESP32-S2-DevKitC-1	3
1.1.1 Getting Started	3
1.1.2 Hardware Reference	6
1.1.3 Hardware Revision Details	9
1.1.4 Related Documents	9
2 ESP32-S2-DevKitM-1	11
2.1 ESP32-S2-DevKitM-1	11
2.1.1 Getting Started	12
2.1.2 Hardware Reference	14
2.1.3 Hardware Revision Details	17
2.1.4 Related Documents	17
3 EOL (End of Life) Boards	19
3.1 ESP32-S2-HMI-DevKit-1	19
3.1.1 ESP32-S2-HMI-DevKit-1 V1.0	19
3.1.2 Reference Documentation	26
3.2 ESP32-S2-Touch-Devkit-1	31
3.2.1 ESP32-S2-Touch-Devkit-1	31
3.3 ESP32-S2-Kaluga-1	37
3.3.1 ESP32-S2-Kaluga-1 Kit v1.3	37
3.4 ESP32-S2-Saola-1	72
3.4.1 ESP32-S2-Saola-1	72
4 Related Documentation and Resources	79
4.1 Related Documentation	79
4.2 Developer Zone	79
4.3 Products	80
4.4 Contact Us	80
5 Disclaimer and Copyright Notice	81

This document provides detailed user guides and examples for ESP32-S2 series development boards.

Note: For the full list of Espressif development boards, please go to [ESP DevKits](#).

Chapter 1

ESP32-S2-DevKitC-1

ESP32-S2-DevKitC-1 is an entry-level development board. This board integrates complete Wi-Fi functions. Most of the I/O pins are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-DevKitC-1 on a breadboard.

1.1 ESP32-S2-DevKitC-1

This user guide will help you get started with ESP32-S2-DevKitC-1 and will also provide more in-depth information.

ESP32-S2-DevKitC-1 is an entry-level development board. This board integrates complete Wi-Fi functions. Most of the I/O pins are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-DevKitC-1 on a breadboard.

The document consists of the following major sections:

- *Getting Started*: Overview of ESP32-S2-DevKitC-1 and hardware/software setup instructions to get started.
- *Hardware Reference*: More detailed information about the ESP32-S2-DevKitC-1's hardware.
- *Hardware Revision Details*: Revision history, known issues, and links to user guides for previous versions (if any) of ESP32-S2-DevKitC-1.
- *Related Documents*: Links to related documentation.

1.1.1 Getting Started

This section provides a brief introduction of ESP32-S2-DevKitC-1, instructions on how to do the initial hardware setup and how to flash firmware onto it.

Description of Components

The key components of the board are described in a clockwise direction.

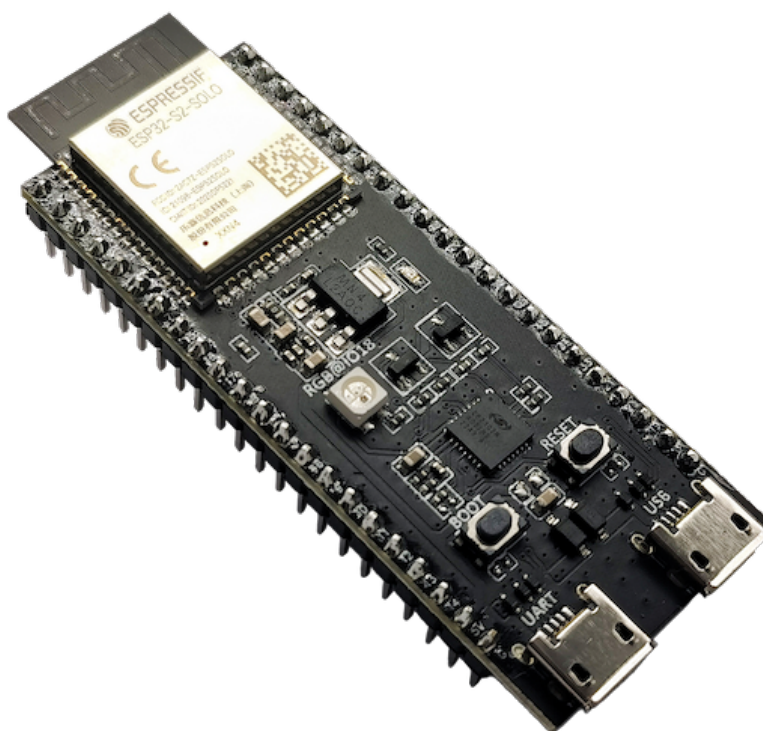


Fig. 1: ESP32-S2-DevKitC-1 with the ESP32-S2-SOLO Module

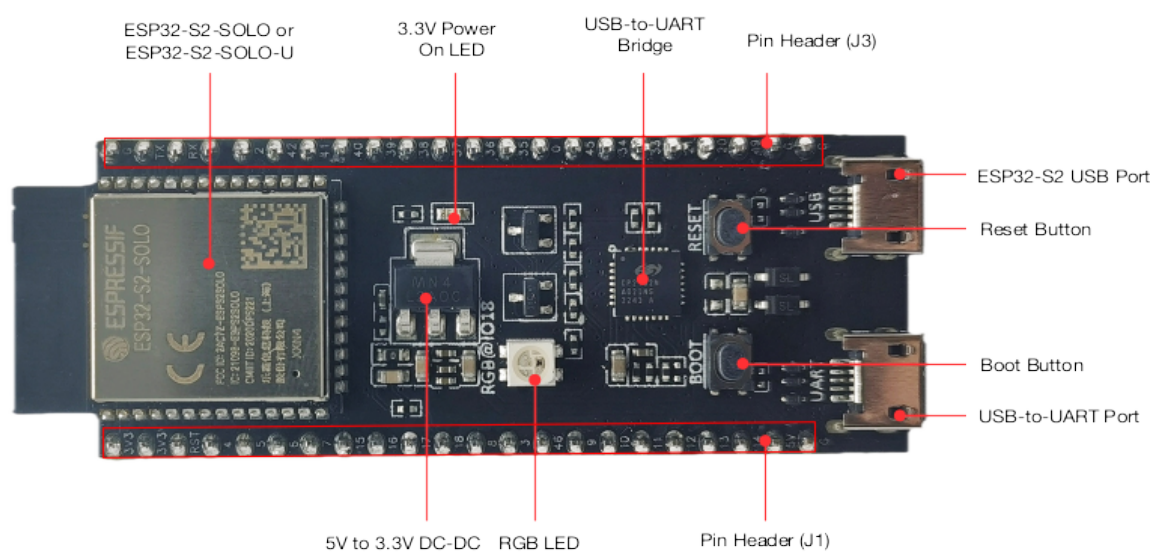


Fig. 2: ESP32-S2-DevKitC-1 - front

Key Component	Description
On-board module (ESP32-S2-SOLO or ESP32-S2-SOLO-U in the above figure)	ESP32-S2-SOLO series modules with an on-board PCB antenna or a connector for an external antenna. This series of modules have multiple options for flash and PSRAM size. For more information, please refer to Ordering Information .
3.3 V Power On LED	Turns on when the USB power is connected to the board.
USB-to-UART Bridge	Single USB-to-UART bridge chip provides transfer rates up to 3 Mbps.
Pin Headers	All available GPIO pins (except for the SPI bus for flash) are broken out to the pin headers on the board. For details, please see Header Block .
ESP32-S2 USB Port	ESP32-S2 full-speed USB OTG interface, compliant with the USB 1.1 specifications. The interface is used for power supply to the board, for flashing applications to the chip, and for communication with the chip using USB 1.1 protocols.
Reset Button	Press this button to restart the system.
Boot Button	Download button. Holding down Boot and then pressing Reset initiates Firmware Download mode for downloading firmware through the serial port.
USB-to-UART Port	A Micro-USB port used for power supply to the board, for flashing applications to the chip, as well as the communication with the chip via the on-board USB-to-UART bridge.
RGB LED	Addressable RGB LED, driven by GPIO18.
5 V to 3.3 V LDO	Power regulator that converts a 5 V supply into a 3.3 V output.

Start Application Development

Before powering up your ESP32-S2-DevKitC-1, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- ESP32-S2-DevKitC-1
- USB 2.0 cable (Standard-A to Micro-B)
- Computer running Windows, Linux, or macOS

Note: Be sure to use an appropriate USB cable. Some cables are for charging only and do not provide the needed data lines nor work for programming the boards.

Hardware Setup Connect the board with the computer using **USB-to-UART Port** or **ESP32-S2 USB Port**. In subsequent steps, **USB-to-UART Port** will be used by default.

Software Setup Please proceed to [Get Started](#), where Section [Installation](#) will quickly help you set up the development environment and then flash an application example into your ESP32-S2-DevKitC-1.

Contents and Packaging

Ordering Information The development board has a variety of variants to choose from, as shown in the table below.

Ordering Code	On-board Module	Flash	PSRAM	Antenna
ESP32-S2-DevKitC-1-N8R2	ESP32-S2-SOLO-2 (Recommended)	8 MB	2 MB	PCB on-board antenna
ESP32-S2-DevKitC-1U-N8R2	ESP32-S2-SOLO-2U (Recommended)	8 MB	2 MB	External antenna connector
ESP32-S2-DevKitC-1 (EOL)	ESP32-S2-SOLO (EOL)	4 MB	—	PCB on-board antenna
ESP32-S2-DevKitC-1U (EOL)	ESP32-S2-SOLO-U (EOL)	4 MB	—	External antenna connector
ESP32-S2-DevKitC-1R (EOL)	ESP32-S2-SOLO (EOL)	4 MB	2 MB	PCB on-board antenna
ESP32-S2-DevKitC-1RU (EOL)	ESP32-S2-SOLO-U (EOL)	4 MB	2 MB	External antenna connector

Retail Orders If you order a few samples, each ESP32-S2-DevKitC-1 comes in an individual package in either antistatic bag or any packaging depending on your retailer.

For retail orders, please go to <https://www.espressif.com/en/contact-us/get-samples>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

1.1.2 Hardware Reference

Block Diagram

The block diagram below shows the components of ESP32-S2-DevKitC-1 and their interconnections.

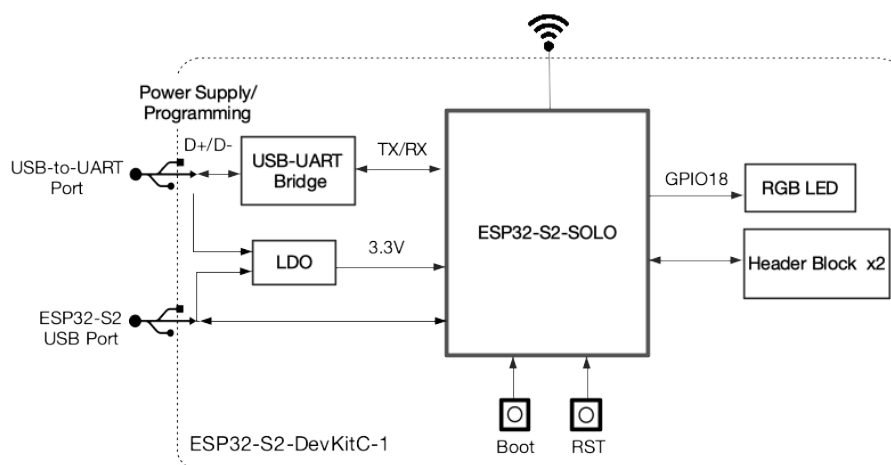


Fig. 3: ESP32-S2-DevKitC-1 (click to enlarge)

Power Supply Options There are three mutually exclusive ways to provide power to the board:

- USB-to-UART Port and ESP32-S2 USB Port (either one or both), default power supply (recommended)
- 5V and G (GND) pins
- 3V3 and G (GND) pins

Header Block

The two tables below provide the **Name** and **Function** of the pin headers on both sides of the board (J1 and J3). The pin header names are shown in *ESP32-S2-DevKitC-1 - front*. The numbering is the same as in the [ESP32-S2-DevKitC-1 Schematic](#) (PDF).

J1

No.	Name	Type ¹	Function
1	3V3	P	3.3 V power supply
2	3V3	P	3.3 V power supply
3	RST	I	CHIP_PU
4	4	I/O/T	RTC_GPIO4, GPIO4, TOUCH4, ADC1_CH3
5	5	I/O/T	RTC_GPIO5, GPIO5, TOUCH5, ADC1_CH4
6	6	I/O/T	RTC_GPIO6, GPIO6, TOUCH6, ADC1_CH5
7	7	I/O/T	RTC_GPIO7, GPIO7, TOUCH7, ADC1_CH6
8	15	I/O/T	RTC_GPIO15, GPIO15, U0RTS, ADC2_CH4, XTAL_32K_P
9	16	I/O/T	RTC_GPIO16, GPIO16, U0CTS, ADC2_CH5, XTAL_32K_N
10	17	I/O/T	RTC_GPIO17, GPIO17, U1TXD, ADC2_CH6, DAC_1
11	18 ²	I/O/T	RTC_GPIO18, GPIO18, U1RXD, ADC2_CH7, DAC_2, CLK_OUT3, RGB LED
12	8	I/O/T	RTC_GPIO8, GPIO8, TOUCH8, ADC1_CH7
13	3	I/O/T	RTC_GPIO3, GPIO3, TOUCH3, ADC1_CH2
14	46	I	GPIO46
15	9	I/O/T	RTC_GPIO9, GPIO9, TOUCH9, ADC1_CH8, FSPiHD
16	10	I/O/T	RTC_GPIO10, GPIO10, TOUCH10, ADC1_CH9, FSPICS0, FSPiIO4
17	11	I/O/T	RTC_GPIO11, GPIO11, TOUCH11, ADC2_CH0, FSPiD, FSPiIO5
18	12	I/O/T	RTC_GPIO12, GPIO12, TOUCH12, ADC2_CH1, FSPiCLK, FSPiIO6
19	13	I/O/T	RTC_GPIO13, GPIO13, TOUCH13, ADC2_CH2, FSPiQ, FSPiIO7
20	14	I/O/T	RTC_GPIO14, GPIO14, TOUCH14, ADC2_CH3, FSPiWP, FSPiDQS
21	5V	P	5 V power supply
22	G	G	Ground

¹ P: Power supply; I: Input; O: Output; T: High impedance.

² GPIO18 is not pulled up on boards with an ESP32-S2-SOLO-2 or ESP32-S2-SOLO-2U module.

J3

No.	Name	Type	Function
1	G	G	Ground
2	TX	I/O/T	U0TXD, GPIO43, CLK_OUT1
3	RX	I/O/T	U0RXD, GPIO44, CLK_OUT2
4	1	I/O/T	RTC_GPIO1, GPIO1, TOUCH1, ADC1_CH0
5	2	I/O/T	RTC_GPIO2, GPIO2, TOUCH2, ADC1_CH1
6	42	I/O/T	MTMS, GPIO42
7	41	I/O/T	MTDI, GPIO41, CLK_OUT1
8	40	I/O/T	MTDO, GPIO40, CLK_OUT2
9	39	I/O/T	MTCK, GPIO39, CLK_OUT3
10	38	I/O/T	GPIO38, FSPIWP
11	37	I/O/T	SPIDQS, GPIO37, FSPIQ
12	36	I/O/T	SPIIO7, GPIO36, FSPICLK
13	35	I/O/T	SPIIO6, GPIO35, FSPID
14	0	I/O/T	RTC_GPIO0, GPIO0
15	45	I/O/T	GPIO45
16	34	I/O/T	SPIIO5, GPIO34, FSPICS0
17	33	I/O/T	SPIIO4, GPIO33, FSPIHD
18	21	I/O/T	RTC_GPIO21, GPIO21
19	20	I/O/T	RTC_GPIO20, GPIO20, U1CTS, ADC2_CH9, CLK_OUT1, USB_D+
20	19	I/O/T	RTC_GPIO19, GPIO19, U1RTS, ADC2_CH8, CLK_OUT2, USB_D-
21	G	G	Ground
22	G	G	Ground

ESP32-S2-DevKitC-1

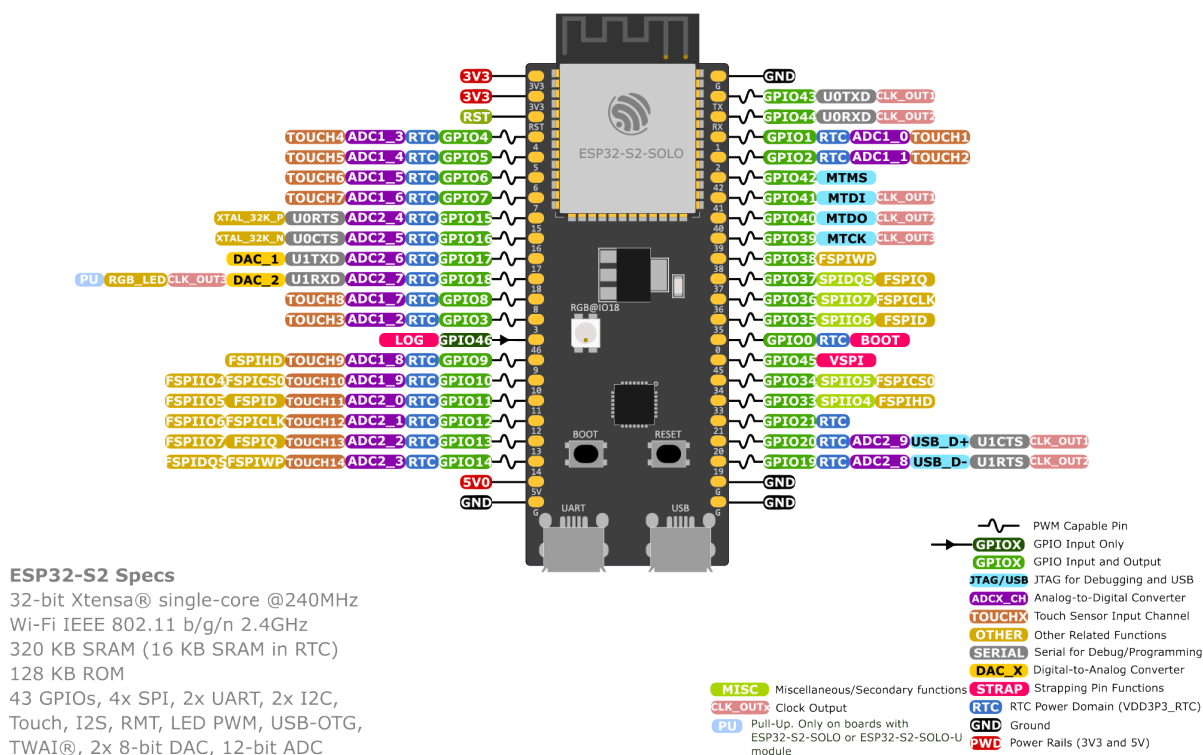


Fig. 4: ESP32-S2-DevKitC-1 Pin Layout (click to enlarge)

Pin Layout

1.1.3 Hardware Revision Details

This is the first revision of this board released.

1.1.4 Related Documents

- [ESP32-S2 Series Chip Datasheet \(PDF\)](#)
- [ESP32-S2 Series SoC Errata \(PDF\)](#)
- [ESP32-S2-SOLO-2 & ESP32-S2-SOLO-2U Module Datasheet \(PDF\)](#)
- [ESP32-S2-SOLO & ESP32-S2-SOLO-U Module Datasheet \(PDF\)](#)
- [ESP32-S2-DevKitC-1 Schematic \(PDF\)](#)
- [ESP32-S2-DevKitC-1 PCB Layout \(PDF\)](#)
- [ESP32-S2-DevKitC-1 Dimensions \(PDF\)](#)
- [ESP32-S2-DevKitC-1 Dimensions source file \(DXF\)](#) - You can view it with [Autodesk Viewer](#) online

For further design documentation for the board, please contact us at sales@espressif.com.

Chapter 2

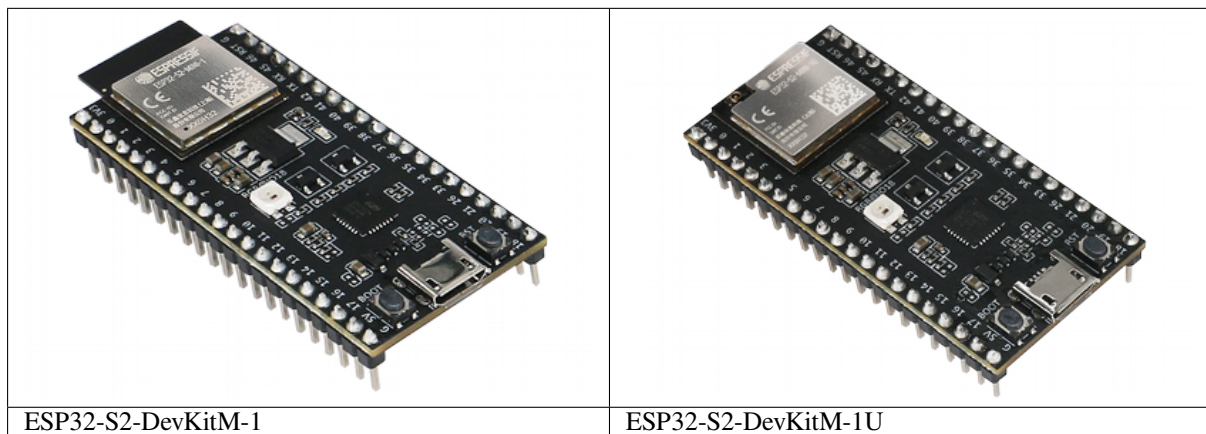
ESP32-S2-DevKitM-1

ESP32-S2-DevKitM-1 is entry-level development board. Most of the I/O pins on the module are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-DevKitM-1 on a breadboard.

2.1 ESP32-S2-DevKitM-1

This user guide provides information on Espressif's small-sized development board ESP32-S2-DevKitM-1.

ESP32-S2-DevKitM-1 is entry-level development board. Most of the I/O pins on the module are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-DevKitM-1 on a breadboard.



The document consists of the following major sections:

- [Getting started](#): Provides an overview of the ESP32-S2-DevKitM-1 and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP32-S2-DevKitM-1's hardware.
- [Hardware Revision Details](#): Revision history, known issues, and links to user guides for previous versions (if any) of ESP32-S2-DevKitM-1.
- [Related Documents](#): Gives links to related documentation.

2.1.1 Getting Started

This section describes how to get started with ESP32-S2-DevKitM-1. It begins with a few introductory sections about the ESP32-S2-DevKitM-1, then Section *Start Application Development* provides instructions on how to get the ESP32-S2-DevKitM-1 ready and flash firmware into it.

Contents and Packaging

Ordering Information The development board has a variety of variants to choose from, as shown in the table below.

Ordering Code	On-board Module	Flash	PSRAM	Antenna
ESP32-S2-DevKitM-1-N4R2	ESP32-S2-MINI-2 (Recommended)	4 MB	2 MB	PCB on-board antenna
ESP32-S2-DevKitM-1U-N4R2	ESP32-S2-MINI-2U (Recommended)	4 MB	2 MB	External antenna connector
ESP32-S2-DevKitM-1 (EOL)	ESP32-S2-MINI-1 (EOL)	4 MB	—	PCB on-board antenna
ESP32-S2-DevKitM-1U (EOL)	ESP32-S2-MINI-1U (EOL)	4 MB	—	External antenna connector
ESP32-S2-DevKitM-1R (EOL)	ESP32-S2-MINI-1 (EOL)	4 MB	2 MB	PCB on-board antenna
ESP32-S2-DevKitM-1RU (EOL)	ESP32-S2-MINI-1U (EOL)	4 MB	2 MB	External antenna connector

Retail Orders If you order a few samples, each ESP32-S2-DevKitM-1 comes in an individual package in either antistatic bag or any packaging depending on your retailer.

For retail orders, please go to <https://www.espressif.com/en/contact-us/get-samples>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

Description of Components

The key components of the board are described in a clockwise direction.

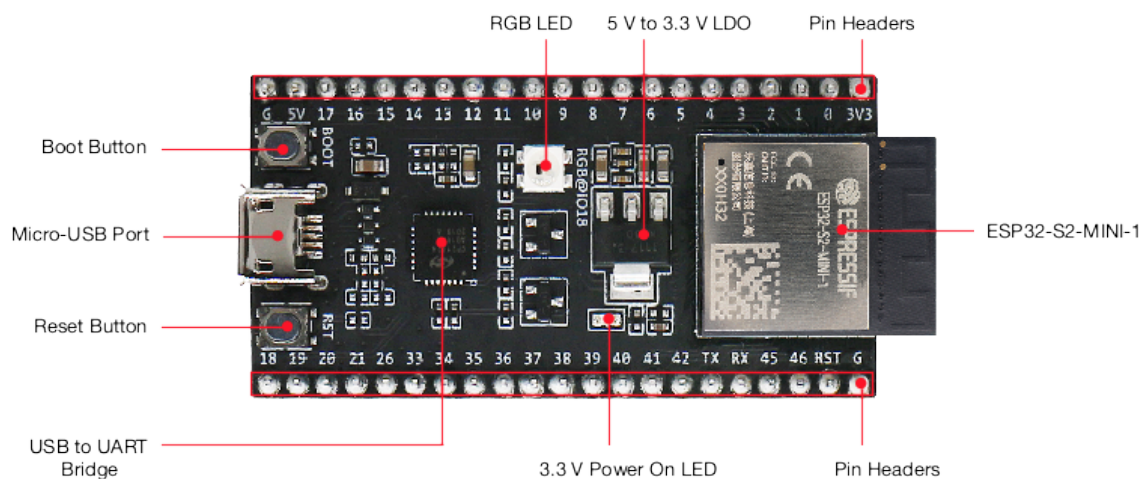


Fig. 1: ESP32-S2-DevKitM-1 - front

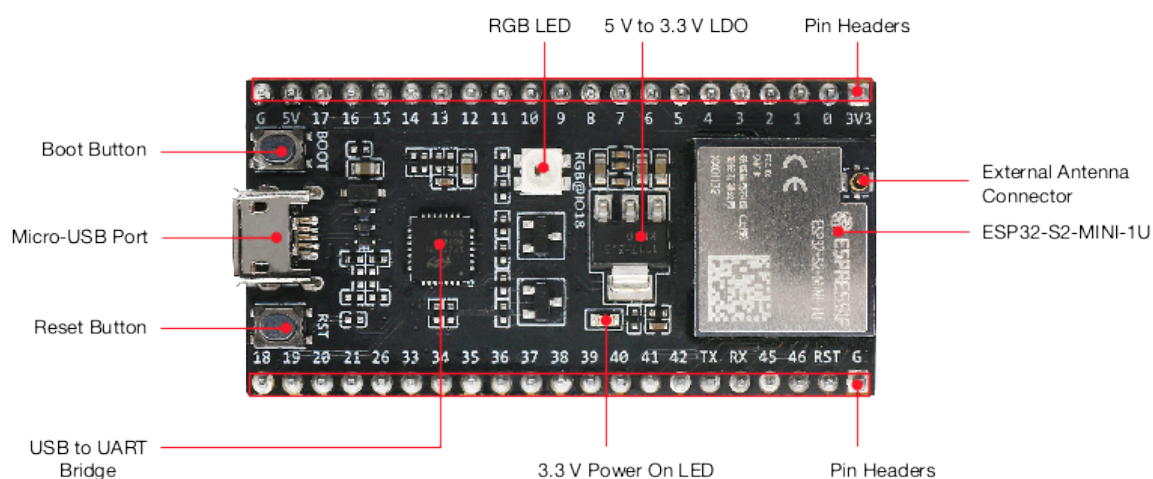


Fig. 2: ESP32-S2-DevKitM-1U - front

Key Component	Description
On-board module (ESP32-S2-MINI-1 or ESP32-S2-MINI-1U in figures above)	ESP32-S2-MINI series modules with an on-board PCB antenna or a connector for an external antenna. This series of modules, known for its small size, have a flash and/or a PSRAM integrated in the chip package. For more information, please refer to Ordering Information .
Pin Headers	All available GPIO pins (except for the SPI bus for flash) are broken out to the pin headers on the board. Users can program ESP32-S2FH4 chip to enable multiple functions such as SPI, I2S, UART, I2C, touch sensors, PWM etc. For details, please see Header Block .
3.3 V Power On LED	Turns on when the USB power is connected to the board.
USB to UART Bridge	Single USB-UART bridge chip provides transfer rates up to 3 Mbps.
Reset Button	Reset button.
Micro-USB Port	USB interface. Power supply for the board as well as the communication interface between a computer and the ESP32-S2FH4 chip.
Boot Button	Download button. Holding down Boot and then pressing Reset initiates Firmware Download mode for downloading firmware through the serial port.
RGB LED	Addressable RGB LED, driven by GPIO18.
5 V to 3.3 V LDO	Power regulator that converts a 5 V supply into a 3.3 V output.
External Antenna Connector	On ESP32-S2-MINI-2U and ESP32-S2-MINI-1U module only. For connector dimensions, please refer to Section External Antenna Connector Dimensions in module datasheet.

Start Application Development

Before powering up your ESP32-S2-DevKitM-1, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- ESP32-S2-DevKitM-1
- USB 2.0 cable (Standard-A to Micro-B)
- Computer running Windows, Linux, or macOS

Note: Be sure to use an appropriate USB cable. Some cables are for charging only and do not provide the needed data lines nor work for programming the boards.

Software Setup Please proceed to [Get Started](#), where Section [Installation](#) will quickly help you set up the development environment and then flash an application example into your ESP32-S2-DevKitM-1.

Note: ESP32-S2 series of chips only is only supported in ESP-IDF master or version v4.2 and higher.

2.1.2 Hardware Reference

Block Diagram

A block diagram below shows the components of ESP32-S2-DevKitM-1 and their interconnections.

Power Supply Options There are three mutually exclusive ways to provide power to the board:

- Micro-USB Port, default power supply
- 5V and GND pin headers

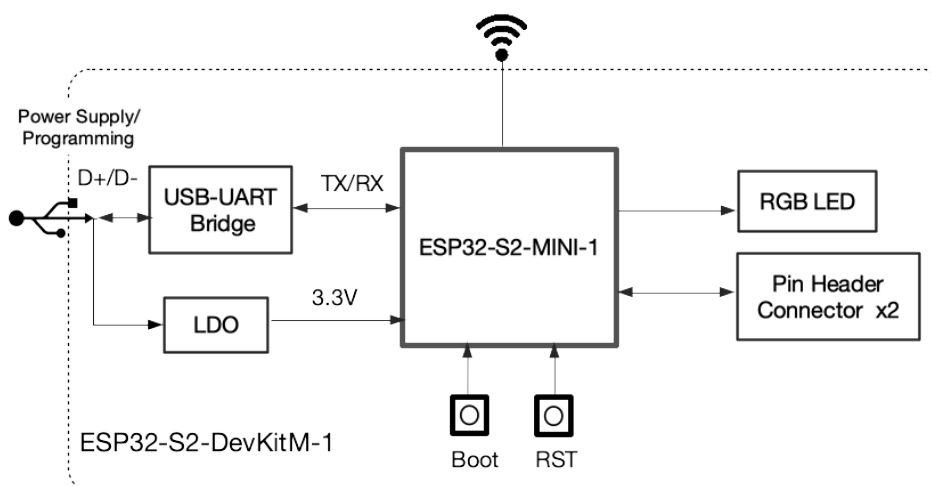


Fig. 3: ESP32-S2-DevKitM-1 (click to enlarge)

- 3V3 and GND pin headers

It is recommended to use the first option: micro USB Port.

Header Block

The two tables below provide the **Name** and **Function** of the pin headers on both sides of the board (J1 and J3). The pin header names are shown in *ESP32-S2-DevKitM-1 - front*. The numbering is the same as in the [ESP32-S2-DevKitM-1 Schematics](#) (PDF).

J1

No.	Name	Type ¹	Function
1	3V3	P	3.3 V power supply
2	0	I/O/T	RTC_GPIO0, GPIO0
3	1	I/O/T	RTC_GPIO1, GPIO1, TOUCH1, ADC1_CH0
4	2	I/O/T	RTC_GPIO2, GPIO2, TOUCH2, ADC1_CH1
5	3	I/O/T	RTC_GPIO3, GPIO3, TOUCH3, ADC1_CH2
6	4	I/O/T	RTC_GPIO4, GPIO4, TOUCH4, ADC1_CH3
7	5	I/O/T	RTC_GPIO5, GPIO5, TOUCH5, ADC1_CH4
8	6	I/O/T	RTC_GPIO6, GPIO6, TOUCH6, ADC1_CH5
9	7	I/O/T	RTC_GPIO7, GPIO7, TOUCH7, ADC1_CH6
10	8	I/O/T	RTC_GPIO8, GPIO8, TOUCH8, ADC1_CH7
11	9	I/O/T	RTC_GPIO9, GPIO9, TOUCH9, ADC1_CH8, FSPICHD
12	10	I/O/T	RTC_GPIO10, GPIO10, TOUCH10, ADC1_CH9, FSPICS0, FSPHIO4
13	11	I/O/T	RTC_GPIO11, GPIO11, TOUCH11, ADC2_CH0, FSPID, FSPHIO5
14	12	I/O/T	RTC_GPIO12, GPIO12, TOUCH12, ADC2_CH1, FSPICLK, FSPHIO6
15	13	I/O/T	RTC_GPIO13, GPIO13, TOUCH13, ADC2_CH2, FSPIQ, FSPHIO7
16	14	I/O/T	RTC_GPIO14, GPIO14, TOUCH14, ADC2_CH3, FSPWP, FSPIDQS
17	15	I/O/T	RTC_GPIO15, GPIO15, U0RTS, ADC2_CH4, XTAL_32K_P
18	16	I/O/T	RTC_GPIO16, GPIO16, U0CTS, ADC2_CH5, XTAL_32K_N
19	17	I/O/T	RTC_GPIO17, GPIO17, U1TXD, ADC2_CH6, DAC_1
20	5V	P	5 V power supply
21	G	G	Ground

¹ P: Power supply; I: Input; O: Output; T: High impedance.

J3

No.	Name	Type	Function
1	G	G	Ground
2	RST	I	CHIP_PU
3	46	I	GPIO46
4	45	I/O/T	GPIO45
5	RX	I/O/T	U0RXD, GPIO44, CLK_OUT2
6	TX	I/O/T	U0TXD, GPIO43, CLK_OUT1
7	42	I/O/T	MTMS, GPIO42
8	41	I/O/T	MTDI, GPIO41, CLK_OUT1
9	40	I/O/T	MTDO, GPIO40, CLK_OUT2
10	39	I/O/T	MTCK, GPIO39, CLK_OUT3
11	38	I/O/T	GPIO38, FSPIWP
12	37	I/O/T	SPIDQS, GPIO37, FSPIQ
13	36	I/O/T	SPIIO7, GPIO36, FSPICLK
14	35	I/O/T	SPIIO6, GPIO35, FSPID
15	34	I/O/T	SPIIO5, GPIO34, FSPICS0
16	33	I/O/T	SPIIO4, GPIO33, FSPIHD
17	26	I/O/T	SPICS1, GPIO26
18	21	I/O/T	RTC_GPIO21, GPIO21
19	20	I/O/T	RTC_GPIO20, GPIO20, U1CTS, ADC2_CH9, CLK_OUT1, USB_D+
20	19	I/O/T	RTC_GPIO19, GPIO19, U1RTS, ADC2_CH8, CLK_OUT2, USB_D-
21	18	I/O/T	RTC_GPIO18, GPIO18, U1RXD, ADC2_CH7, DAC_2, CLK_OUT3, RGB LED

ESP32-S2-DevKitM-1

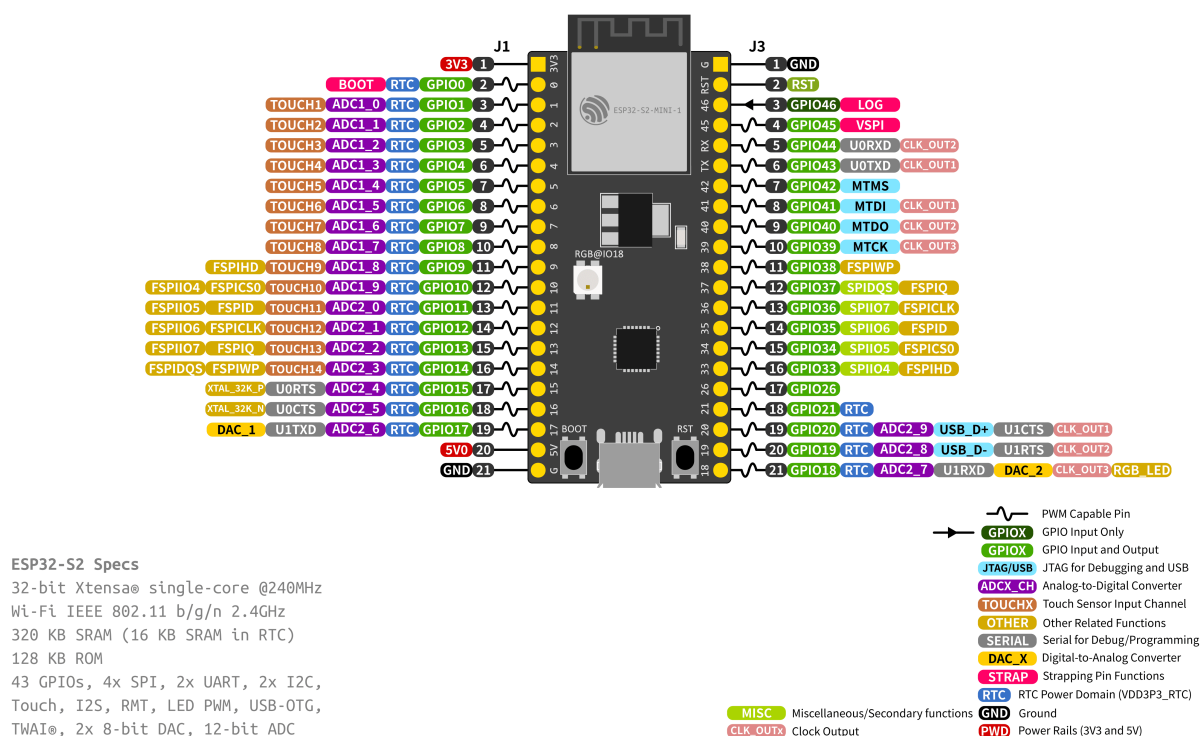


Fig. 4: ESP32-S2-DevKitM-1 Pin Layout (click to enlarge)

Pin Layout

2.1.3 Hardware Revision Details

This is the first revision of this board released.

2.1.4 Related Documents

- [ESP32-S2 Series Chip Datasheet \(PDF\)](#)
- [ESP32-S2 Series SoC Errata \(PDF\)](#)
- [ESP32-S2-MINI-2 & ESP32-S2-MINI-2U Module Datasheet \(PDF\)](#)
- [ESP32-S2-MINI-1 & ESP32-S2-MINI-1U Module Datasheet \(PDF\)](#)
- [ESP32-S2-DevKitM-1 Schematics \(PDF\)](#)
- [ESP32-S2-DevKitM-1 PCB Layout \(PDF\)](#)
- [ESP32-S2-DevKitM-1 Dimensions \(PDF\)](#)
- [ESP Product Selector](#)

For other design documentation for the board, please contact us at sales@espressif.com.

Chapter 3

EOL (End of Life) Boards

This section contains user guides for the ESP32-S2 end-of-life development boards and is provided for reference only. While these boards may still be available on the market or used in legacy systems, they no longer receive updates, bug fixes, or official support. It is recommended to switch to newer development boards for better performance and more features.

3.1 ESP32-S2-HMI-DevKit-1

ESP32-S2-HMI-DevKit-1 has been specifically designed for human-machine interfaces in smart-home automation controllers, smart speakers with display, smart alarm clocks, etc.

Application examples for this board can be found at [Examples](#).

3.1.1 ESP32-S2-HMI-DevKit-1 V1.0

This user guide will help you get started with ESP32-S2-HMI-DevKit-1 and will also provide more in-depth information.

ESP32-S2-HMI-DevKit-1 has been specifically designed for human-machine interfaces in smart-home automation controllers, smart speakers with display, smart alarm clocks, etc. This development kit supports rapid secondary development, since developers can take advantage of the kit's various onboard resources and expansion interfaces, in order to develop various functions.

The main features of the board are listed below:

- **Module Embedded:** ESP32-S2-WROVER module with 4 MB flash and 2 MB PSRAM
- **Display:** 4.3-inch TFT-LCD which uses 16-bit 8080 parallel port with 480×800 resolution and 256-level hardware DC backlight adjustment circuit, connected to an I2C capacitive touch panel
- **Audio:** Audio amplifier, built-in microphone, speaker connector
- **Storage:** microSD card connector
- **Sensors:** 3-axis accelerometer, 3-axis gyroscope, ambient light sensor, temperature and humidity sensors
- **Expansion:** SPI header, TWAI interface (compatible with CAN 2.0), I2C ADC, UART/Prog header
- **LEDs:** Programmable RGB LED and IR LED
- **Buttons:** Wake Up and Reset buttons
- **USB:** 1 x USB-C OTG (DFU/CDC) port, 1 x USB-C debug port
- **Power Supply:** 5V and 3.3V power headers
- **Optional Rechargeable Battery:** 1,950 mAh single-core lithium battery with a charge IC



Fig. 1: ESP32-S2-HMI-DevKit-1 with ESP32-S2-WROVER module

The document consists of the following major sections:

- *Getting started*: Overview of the board and hardware/software setup instructions to get started.
- *Hardware Reference*: More detailed information about the board's hardware.
- *Related Documents*: Links to related documentation.

Getting Started

This section provides a brief introduction of ESP32-S2-HMI-DevKit-1, instructions on how to do the initial hardware setup and how to flash firmware onto it.

Description of Components ESP32-S2-HMI-DevKit-1 is an HMI development board designed based on the ESP32-S2. The following figure describes its key on-board resources:

The key components of the board are described in a clockwise direction.

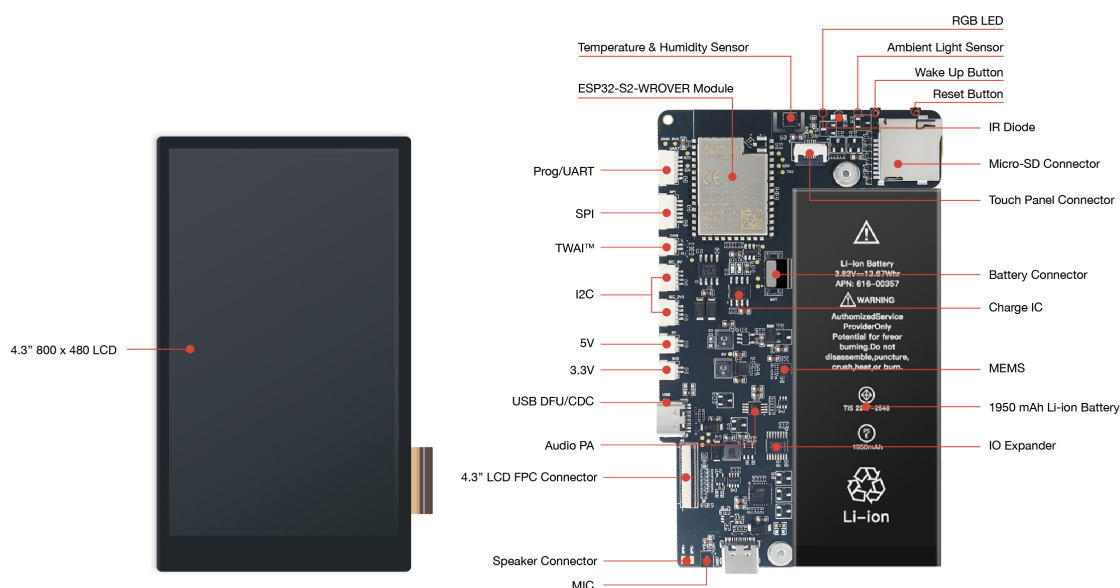


Fig. 2: ESP32-S2-HMI-DevKit-1 (click to enlarge)

Key Component	Description
ESP32-S2-WROVER Module	ESP32-S2-WROVER is a powerful, generic Wi-Fi MCU module that integrates ESP32-S2. It has a PCB antenna, a 4 MB external SPI flash and an additional 2 MB PSRAM.
Temperature & Humidity Sensor	Temperature & humidity sensors for detecting ambient temperature and humidity. Read via the I2C bus.
RGB LED	Addressable RGB LED (WS2812), driven by GPIO21. Can switch between the LED and IR LED via IO expander.
Ambient Light Sensor	Ambient Light Sensor used to detect ambient light intensity. Read via the I2C bus.
Wake Up Button	Download button. Press and hold the Boot button while pressing the Reset button to initiate the “firmware download” mode to download the firmware via the serial port. This button can also be configured to wake up the device from deep sleep mode.
Reset Button	Press this button to restart the system.
IR LED	Infrared emitting diode, driven by GPIO21. Can switch between RGB LED and IR LED via IO expander.
Charge IC	Charge battery.
MEMS	3-axis accelerometer and 3-axis gyroscope.
1950 mAh Li-ion Battery	An optional 1,950 mAh rechargeable lithium battery.
IO Expander	Expand GPIO through I2C bus.
MIC	On-board simulated microphone.
Audio PA	Audio amplifier.
4.3” 800 × 480 LCD	A 4.3-inch TFT-LCD which uses 16-bit 8080 parallel port with 480×800 resolution, and 256-level hardware DC backlight adjustment circuit. It is connected with an I2C capacitive touch panel overlay.

Start Application Development Before powering up your board, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- 1 x PC loaded with Windows, macOS or Linux (Linux operating system is recommended)
- 1 x ESP32-S2-HMI-DevKit-1
- 1 x USB-C cable (it is recommended to prepare two USB-C cables if you want to evaluate MCU's USB functions)
- 1 x Speaker (8 Ohm, 2 W)
- 1 x microSD card (some examples may have large storage needs)

Note: Be sure to use an appropriate USB cable. Some cables are for charging only and do not provide the needed data lines nor work for programming the boards.

Hardware Setup To facilitate your quick evaluation of all examples, please follow these steps to set up the board:

1. Insert microSD card into the connector. Please make sure all the important data is backed up, as the microSD card may be formatted if its partition format is not FAT.
2. If you need to evaluate the audio playback function, please connect the speaker pad near the USB port on the bottom of the board to the supplied speaker, or to another speaker with a similar size (8 Ohm, 2 W).

Software Setup First, please make sure you have configured the ESP-IDF development environment correctly. To ensure this, please enter `idf.py --version` in your terminal and if the output is similar to `ESP-IDF v4.2-dev-2084-g98d5b5dfd`, it means you have installed ESP-IDF correctly. For detailed information about installation and configuration, please refer to [ESP-IDF Get Started](#).

After configuration completed, you can switch back to the `esp-dev-kits/esp32-s2-hmi-devkit-1` directory. All code examples are placed under the examples directory, you can build projects by running `idf.py build`.

Project Options Various examples are provided for ESP32-S2-HMI-DevKit-1 as shown below:

- Printing “Hello world!” on screen: [examples/esp32-s2-hmi-devkit-1/examples/get-started/hello_world](#)
- Blinking WS2812 LED and showing the color on screen: [examples/esp32-s2-hmi-devkit-1/examples/get-started/led_blink](#)
- Starting a UI to configure Wi-Fi credential: [examples/esp32-s2-hmi-devkit-1/examples/get-started/provision](#)
- Acquiring audio with ADC from the output of analog MIC: [examples/esp32-s2-hmi-devkit-1/examples/audio/audio_record](#)
- Playing music: [examples/esp32-s2-hmi-devkit-1/examples/audio/music_player](#)
- Shutting down selected board area into a deep sleep: [examples/esp32-s2-hmi-devkit-1/examples/power](#)
- Using Freetype to render fonts: [examples/esp32-s2-hmi-devkit-1/examples/freetype](#)
- Using on-board sensors: [examples/esp32-s2-hmi-devkit-1/examples/sensors](#)
- Using smart panel: [examples/esp32-s2-hmi-devkit-1/examples/smart-panel](#)
- Viewing files on SD card: [examples/esp32-s2-hmi-devkit-1/examples/storage/sdcard_fatfs](#)
- USB flash disk: [examples/esp32-s2-hmi-devkit-1/examples/storage/usb_msc](#)

You can configure project options by entering `idf.py menuconfig` in each example directory.

Please make sure to correctly configure the following options in menuconfig:

- (Top) > HMI Board Config > HMI board: Select board version. By default, please select ESP32-S2-HMI-DevKit-V2;
- (Top) > HMI Board Config > Audio HAL: Select audio output interface, whether to use PWM or DAC;
- (Top) > HMI Board Config > LCD Drivers: Select display IC type for LCD. By default, ESP32-S2-HMI-DevKit-1 uses RM68120 as its display IC;
- In (Top) > Component config > ESP32S2-specific, please go to the Support for external, SPI-connected RAM option:
 - Go to SPI RAM config > Set RAM clock speed, and set the PSRAM clock as 80 MHz clock speed;

- (Top) -> Component config -> FreeRTOS:set Tick rate (Hz) as 1000.

In each example folder, we have provided a default configuration file named `sdkconfig.defaults`, with above options configured correctly.

ESP-IDF Version Dependencies The `examples/esp32-s2-hmi-devkit-1/examples/storage/usb_msc` example needs to be built in IDF v4.3, while other examples can be built in IDF v4.2 and later versions.

Notice: Due to strict regulation on battery export, for deliveries outside of China mainland, we are shipping ESP32-S2-HMI-DevKit-1 without the battery. As a substitute, you can use the iPhone 5 replacement battery (the connector type is non-standard). The battery capacity is not critical.

Contents and Packaging

Retail Orders If you order one or several samples of the kit, each ESP32-S2-HMI-DevKit-1 development kit comes in an individual package.



Fig. 3: ESP32-S2-HMI-DevKit-1 package

The contents are as follows:

- Development board - ESP32-S2-HMI-DevKit-1
- Cables - SH1.25 to 2.54mm cables x 7

For retail orders, please go to <https://www.espressif.com/en/company/contact/buy-a-sample>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

Hardware Reference

Block Diagram The block diagram below shows the components of ESP32-S2-HMI-DevKit-1 and their interconnections.

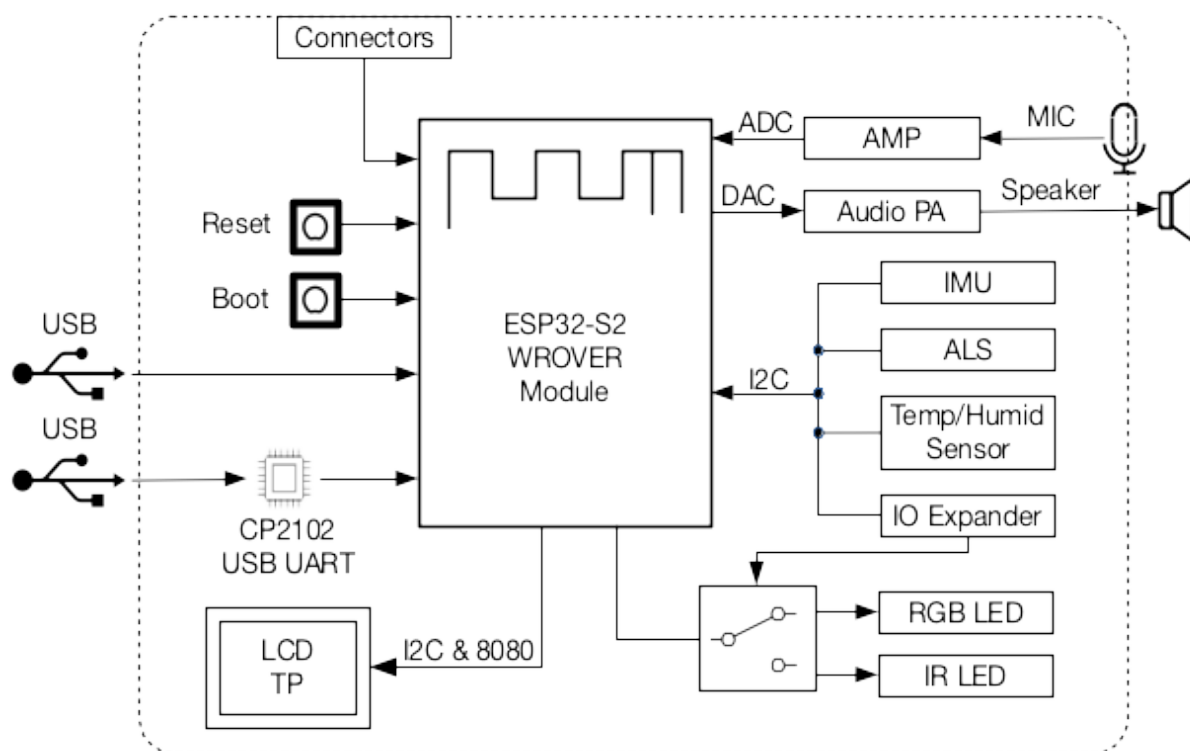


Fig. 4: ESP32-S2-HMI-DevKit-1 block diagram

Power Supply Options The power of the ESP32-S2-HMI-DevKit-1 development board is divided into a 5 V power domain and a 3.3 V power domain, so as to reduce power consumption, improve power efficiency and support battery charging, part of which can be controlled by software whereas the other part is fixed in the hardware design.

To reduce current consumption, the preloaded firmware will power off all controlled power domains and put all ICs in low-power mode.

For more information, please refer to [Power](#).

Connectors It provides multiple extended interfaces for customized development. The connectors of the board are described in a clockwise direction. Please refer to [ESP32-S2-HMI-DevKit-1 key on-board resources](#).

Connectors	Description
Speaker Connector	To connect a speaker.
4.3" LCD FPC Connector	(Reserved) Connect to the supported 4.3" LCD extension board using the FPC cable.
USB DFU/CDC	1 x USB-C OTG (DFU/CDC) port, 1 x USB-C debug port.
3.3 V Connector	3.3 V power header.
5 V Connector	5 V power header.
I2C Connector	I2C connector with 5 V/3.3 V power supply options.
TWAI interface (compatible with CAN 2.0)	Two-wire automotive interface.
SPI	Can connect devices on the SPI bus if the SD card is not in use.
Prog/UART	This interface is used to observe log output and firmware flash.
microSD Connector	Insert microSD card to expand the storage space of the device.
Battery Connector	To connect a battery.

Below is the back view of the board for your reference.



Fig. 5: ESP32-S2-HMI-DevKit-1 - back view

Related Documents

Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP32-S2 Datasheet \(PDF\)](#)
- [ESP32-S2-WROVER Datasheet \(PDF\)](#)
- [ESP32-S2-HMI-DevKit-1 Schematic \(PDF\)](#)
- [ESP32-S2-HMI-DevKit-1 PCB Layout \(PDF\)](#)
- [ESP32-S2-HMI-DevKit-1 Dimensions \(PDF\)](#)
- [ESP32-S2-HMI-DevKit-1 Dimensions Source File \(DXF\)](#) - You can view it with [Autodesk Viewer](#) online

For further design documentation for the board, please contact us at sales@espressif.com.

3.1.2 Reference Documentation

Audio

The ESP32-S2-HMI-DevKit-1 development board supports audio playback and recording. You can find such examples under the `examples/esp32-s2-hmi-devkit-1/examples/audio/` directory.

Audio Playback The ESP32-S2-HMI-DevKit-1 development board can output audio via DAC or PWM. It is recommended to use PWM for audio output since it has lower noise and higher resolution (DAC has 8-bit resolution, while PWM can reach up to 12-bit resolution at 19.2 kHz of sampling rate).

The output signal generated through the IO port goes to the digital potentiometer TPL0401 first for lossless volume adjustment, and then passes the 100 nF isolation capacitor C33 and the 200 kOhm resistor R52. This RC circuit controls the cut-off frequency at around 8 Hz. On top of that, this signal will be sent to the 3 W class-D audio power amplifier NS4150 to set the gain to 1.5 times, thus amplifying the maximum output signal from 3.3 V to 4.95 V (slightly lower than the PA supply, 5 V) so as to maximize the output volume while minimizing saturation distortion.

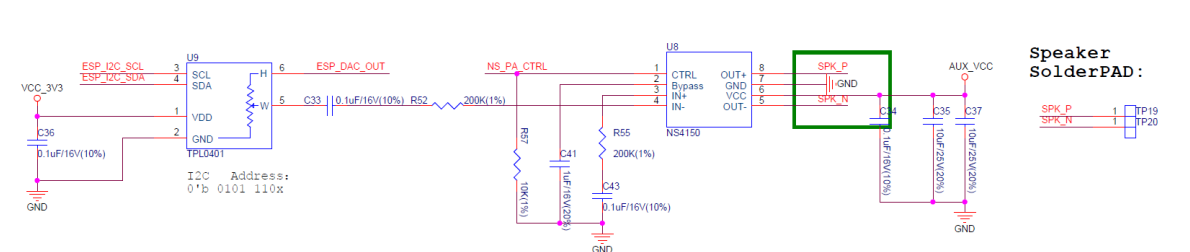


Fig. 6: ESP32-S2-HMI-DevKit-1 audio playback schematic (click to enlarge)

The audio PA is powered by the 5 V power domain. Before using the audio playback function, please make sure this power domain is powered on (refer to [5 V Power Domain](#) section).

Audio Recording The ESP32-S2-HMI-DevKit-1 development board can record audio data from the analog microphone via an internal ADC.

The board is equipped with an analog microphone with a sensitivity of -38 dB. And it will send the output signal to the operational amplifier TLV6741 with a fixed gain to amplify the signal.

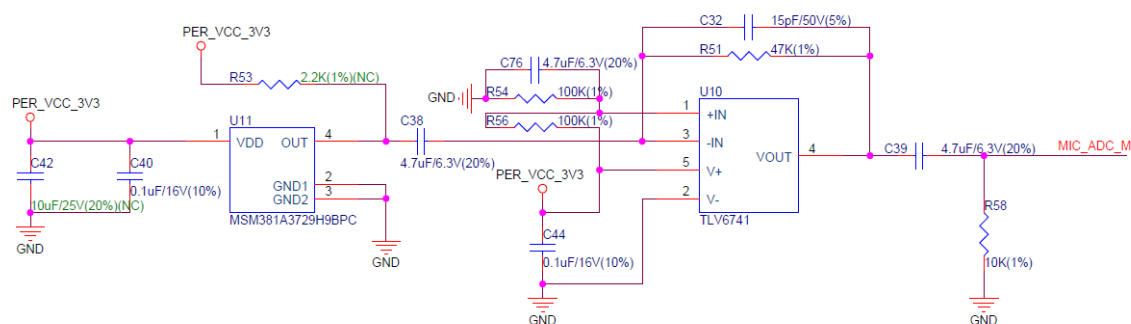


Fig. 7: ESP32-S2-HMI-DevKit-1 microphone schematic (Click to enlarge)

The microphone and operational amplifier mentioned above are powered by a controlled 3.3 V power domain. Before using the audio recording function, please make sure this power domain is powered on (refer to [3.3 V Power Domain](#) section).

Please use the Timer Group interrupt to record audio data. Do not use code such as the following format in tasks for audio recording:

```
size_t index = 0;
uint16_t audio_data[configMAX_ACQUIRE_COUNT];
do {
    audio_data[index] = adc1_get_raw(CONFIG_AUDIO_CHANNEL);
    ets_delay_us(1000000 / CONFIG_AUDIO_FREQ);
} while (++index < CONFIG_MAX_ACQUIRE_COUNT)
```

The above format will cause the CPU to be occupied, thus triggering the task watchdog (if it is not disabled), and make other tasks with lower priorities (e.g., IDLE Task) not able to operate normally.

When recording data via ADC with the interrupt function, you need to re-write the ADC recording function to IRAM_ATTR so as to reduce response time, and place the variables to DRAM. Also, please do not use any semaphore in this function. For more information about implementation examples, please refer to `audio/audio_record` under the `examples` directory.

ADC Accuracy The ADC of ESP32-S2 has a high level of repeatability despite the fact that the lack of reference voltage and using Buck power supply may result a high overall noise.

The ADC is configured with 13-bit resolution and 11 dB attenuation, corresponding to a full-scale voltage of 2.6 V. After polling the 2.5 V voltage of the AD584T reference voltage via ADC1_CH8, we convert the 4096-time uncalibrated raw values into voltage values and get the following data:

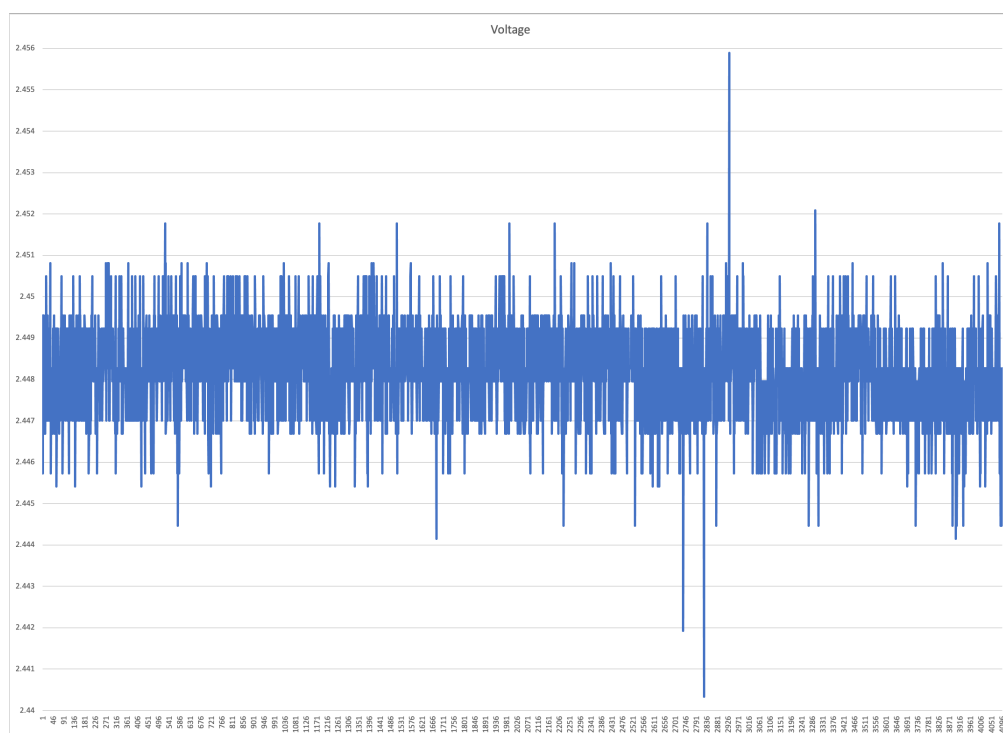


Fig. 8: ESP32-S2-HMI-DevKit-1 ADC (click to enlarge)

As shown in the above figure, most of the uncalibrated data error is within the range of ± 0.005 V with a standard deviation (σ) of 3.339 LSB (0.00106 V). These errors are mainly from the absolute accuracy, i.e., the bias. Therefore, the distortion and noise of sounds sampled via ADC can be kept at a relatively low level.

The AD584T has a peak output noise of 50 μ V at 10 V within the range of 0.1 ~ 10 Hz, and a peak output noise divided by the internal high-precise laser-adjusted resistor at 2.5 V. And an up to 30 mA push-pull capability is

provided by the transistor measured at 10 V. Its output noise at 2.5 V is lower than the resolving power of a 16-bit ADC, which therefore can be used as a testing reference.

Display and Touch Panel

The ESP32-S2-HMI-DevKit-1 has a 4.3-inch color TFT-LCD with 800×480 resolution, and a touch panel with an I2C interface. The interface type and data bit width of this display is controlled by programmable pins. This development board has been configured as 16-bit 8080 parallel communication via resistors.

This LCD uses RM68120 as its display IC and FT5436 as its touch IC.

Communication The display IC of the LCD used in ESP32-S2-HMI-DevKit-1 has been configured for 16-bit 8080 parallel communication, with a total of 18 GPIOs used, i.e., 16 data lines (LCD_D0…LCD_D15), a bit clock signal (LCD_WR) and a data/command distinguish signal (LCD_DC/LCD_RS).

The touch IC uses the I2C interface to communicate with MCU and can share this interface with other I2C's ICs, and thus does not need to use additional GPIOs. The touch IC supports interrupt signal output. The interrupt signal is first sent to the P2 pin of the IO expander, and the falling edge from this pin will generate a low level in the interrupt output pin of the IO expander, so that the MCU receives this interrupt signal. In this case, you can read the input level register of the IO expander to check whether this interrupt is from the touch IC. Once a read operation completed, the interrupt flag will be cleared.

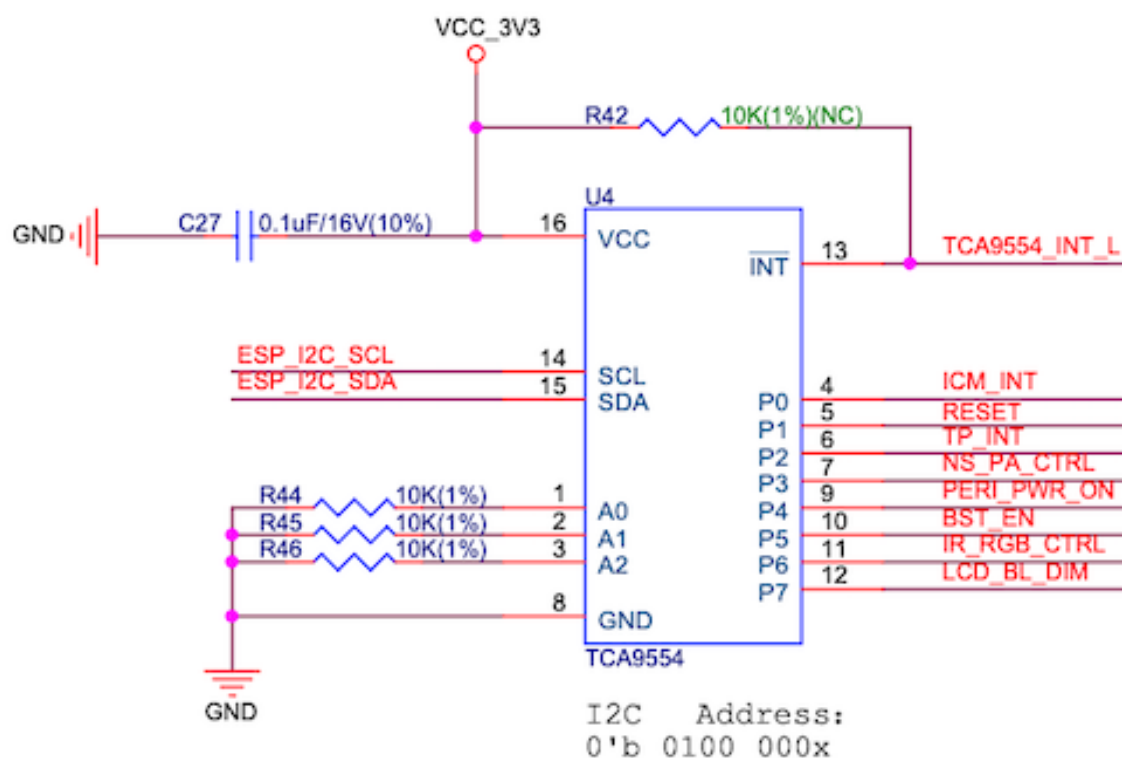


Fig. 9: ESP32-S2-HMI-DevKit-1 IO expander schematic

Backlight As the LEDs are connected in series, they need to be driven by constant current via the Booster circuit. The rated current is 18 mA and the voltage is approximately 24 V (may not be accurate, only for reference). To prevent the feedback voltage of the Booster circuit always being 0 when the display is not connected, and thus causing high voltage loaded to both ends of the backlight filter capacitor C21, please make sure this capacitor can withstand 38 V.

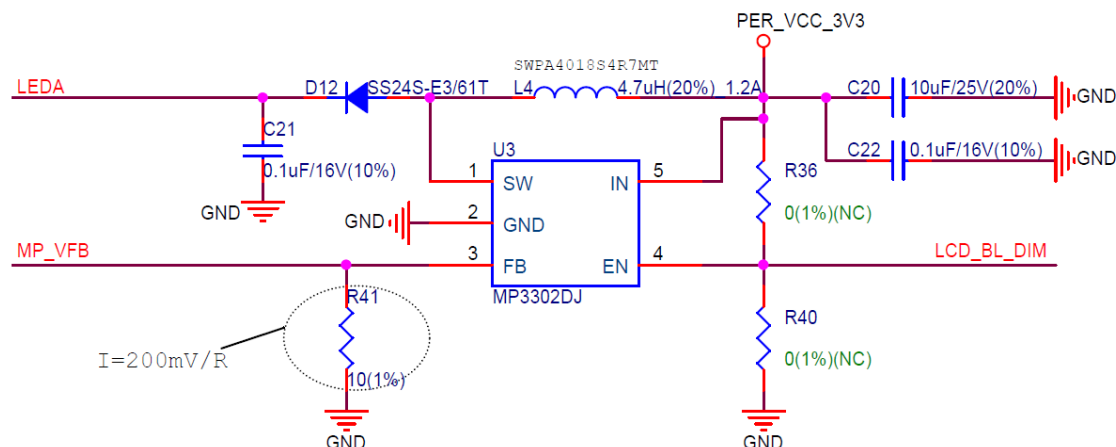


Fig. 10: ESP32-S2-HMI-DevKit-1 backlight PWM dimming schematic (click to enlarge)

Since PWM dimming may cause display flicker and some Booster IC do not support high-frequency PWM signal control, this development board provides an option to use DC dimming circuit to reach high performance, as shown in the figure below:

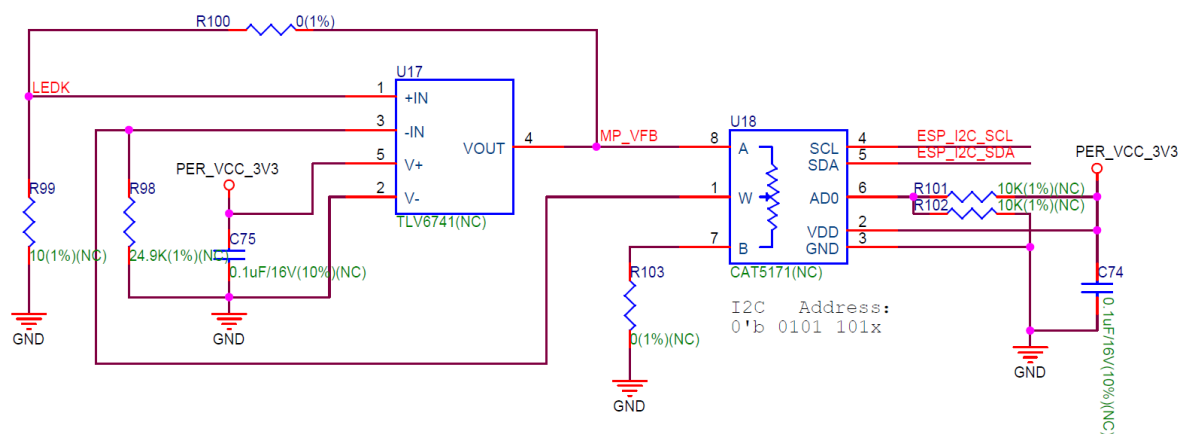


Fig. 11: ESP32-S2-HMI-DevKit-1 DC dimming circuit schematic (click to enlarge)

This DC dimming circuit inputs the VFB voltage to the operational amplifier TLV6741, whose gain resistor is a digital potentiometer that can be modified via the I2C bus. This digital potentiometer is CAT5171, with 256 levels of resolution and a maximum resistance value of 50 kOhm.

The EN pin of the Booster IC is controlled by the P7 pin of the IO expander in high level. If you want to keep the contents while turning off the display, please set this pin to low level so as to disable backlight.

Touch The capacitive touch panel on the development board uses a touch IC with a resolution of 800×480 and supports up to 5-point touch and hardware gesture recognition.

The hardware of this display IC does not support screen rotation itself. Therefore, for scenarios where the panel is needed to be rotated, you may need to convert the data read by the touch IC through calculating its relative value to the resolution or via certain software. Multi-touch is supported by hardware and we provide some APIs for reading the multiple touch points. However, since the LVGL used in the GUI library does not support multi-touch processing for now, you may need to handle the data of these touch points in the application layer yourself.

Power

The power of the ESP32-S2-HMI-DevKit-1 development board is divided into a 5 V power domain and a 3.3 V power domain, so as to reduce power consumption, improve power efficiency and support battery charging. Part of the power domain can be controlled by software whereas the other part is configured as permanently enabled in hardware design.

To reduce current consumption, the preloaded firmware will power off all controlled power domains and put all ICs to low-power mode.

3.3 V Power Domain Most of the ICs and modules are powered by the 3.3 V power domain, which can be divided into an uncontrolled 3.3 V power domain and a controlled 3.3 V power domain.

The uncontrolled 3.3 V power domain cannot be powered off via software, and provides power for the Buck circuit. When there is a power supply from USB, this power domain will obtain power from the 5 V input through the USB cable; when USB is disconnected, it will obtain 3.6 ~ 4.2 V power from the lithium battery. This power domain mainly provides power for the ESP32-S2-WROVER module and other devices which can enter low-power mode via software control.

The controlled 3.3 V power domain comes from the uncontrolled 3.3 V power domain and is turned on/off via a PMOS control switch, which is connected to the P4 pin of the IO expander. This power domain mainly provides power for ICs with higher static power consumption and cannot enter low-power mode.

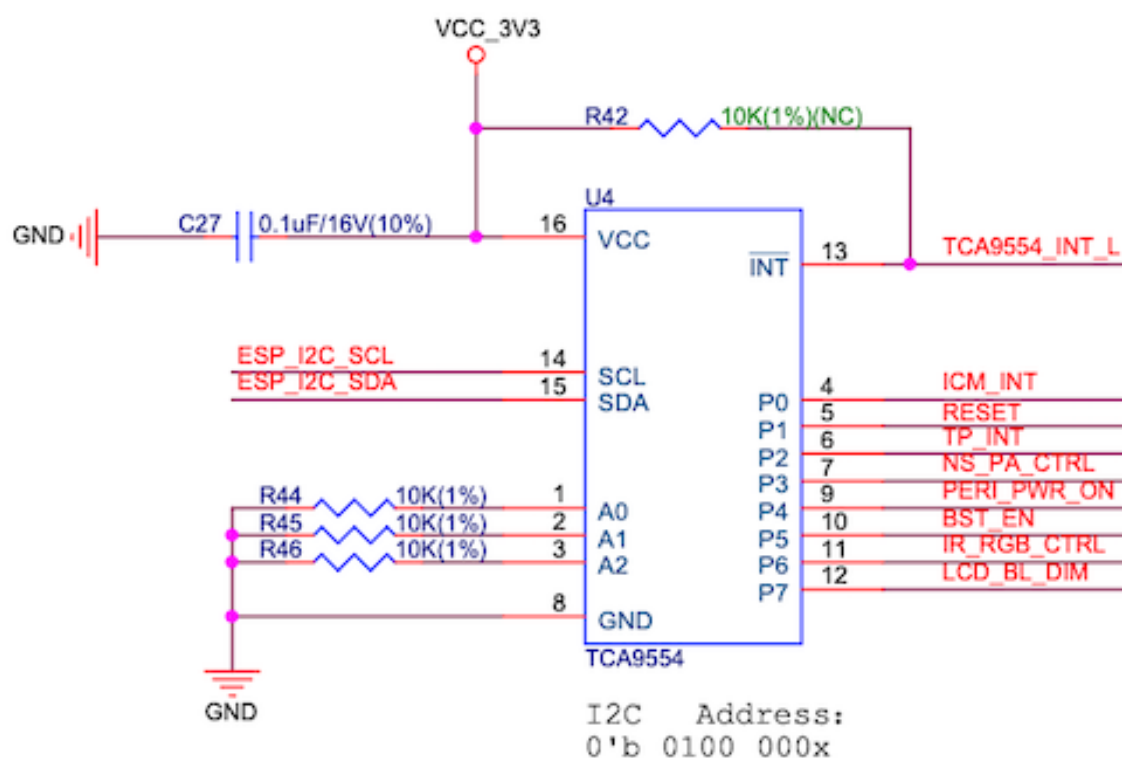


Fig. 12: ESP32-S2-HMI-DevKit-1 IO expander schematic

5 V Power Domain The 5 V power domain of the development board provides power for the audio amplifier and the TWAI® transceiver. It obtains power from the following resources:

- The USB port
- The power input from external 5 V power port

- The power passing through the Booster circuit from the lithium battery

The power obtained from USB and the external 5 V power input supplies power for all devices (except CP2102N) that require 5 V power and cannot be disconnected by software. When obtaining power from the lithium battery, the EN pin level of the Booster IC can be controlled via the P5 pin of the IO expander to enable 5 V power in high level.

The power input through the USB port on the bottom of the board is split into two lines: one provides power for CP2102N while the other becomes USB_5V after passing through a diode. The CP2102N will only be powered up when this USB port is connected, since it only needs to be in operation when the PC is connected. Any 5 V power input will cause the Booster IC to be powered off and charge the on-board lithium battery via the charging IC.

Power Dependencies The following functions depend on the 5 V power domain:

- TWAI® (selects available power supply from USB 5 V or Booster 5 V automatically)
- Audio amplifier (gets power supply from USB 5 V, if it fails, will try from the battery)
- 5 V power output connector (the same as TWAI®)

The following functions depend on the controlled 3.3 V power domain:

- Micro-SD card
- Microphone and its bias circuit, and operational amplifier
- Display and touch function
- WS2812 RGB LED and IR LED
- IR LED

Power State When the development board is connected via the USB cable, the 5 V power domain is powered on automatically and the charging IC outputs voltage to supply power for the battery. In this case, the controlled 3.3 V power domain is controlled by the P4 pin of the IO expander.

When the development board is powered by the battery, the controlled 3.3 V power domain is controlled by the P4 pin of the IO expander while the 5 V power domain is controlled by the P5 pin of the IO expander, and the charging IC will not work.

3.2 ESP32-S2-Touch-Devkit-1

ESP32-S2-Touch-Devkit-1 is a development kit that is aimed at helping evaluate and develop capacitive touch sensor applications on ESP32-S2. It is made up of a Motherboard-Subboard structure.

3.2.1 ESP32-S2-Touch-Devkit-1

Overview

ESP32-S2-Touch-Devkit-1 is a development kit that is aimed at helping evaluate and develop capacitive touch sensor applications on ESP32-S2. It is made up of Motherboard-Subboard structure. The motherboard of ESP32-S2-Touch-Devkit-1 integrates [ESP32-S2-MINI-1](#) controller module and several useful little components such as buzzer, digital tube, RGB light, and so on. There are several kinds of subboards in ESP32-S2-Touch-Devkit-1 with different kinds of capacitive touch sensor pads, developers can choose one of them and connect it with motherboard so that they could develop different kinds of capacitive touch sensor applications. The motherboard and subboard use the pin header/female pin header as the socket connector which makes it plug in and plug out smoothly.

Motherboard

Subboards

- Button-board: Three capacitive touch buttons with waterproof sensor.

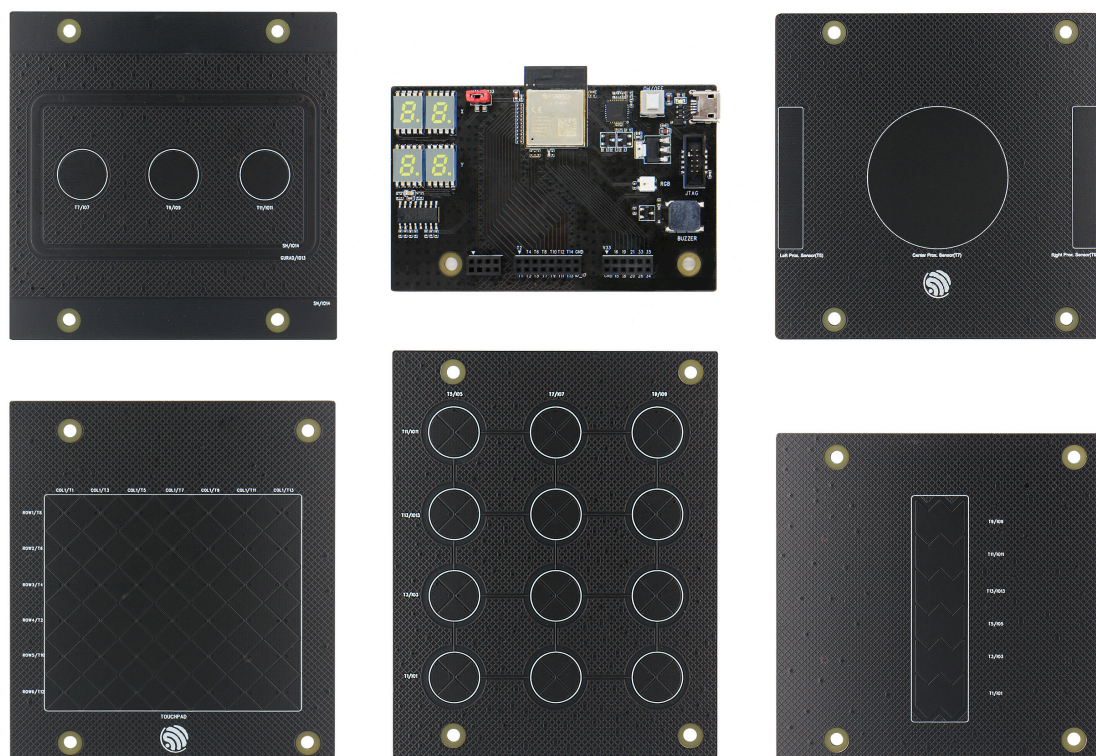


Fig. 13: ESP32-S2-Touch-Devkit-1 Board-set

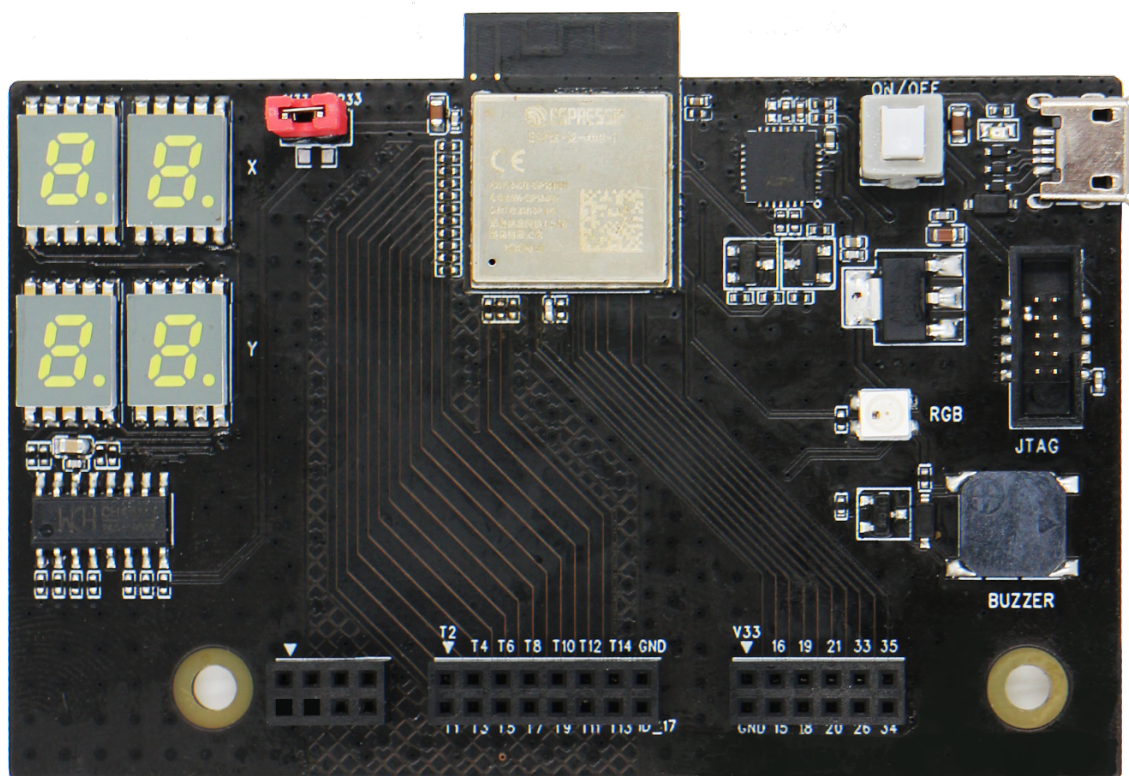
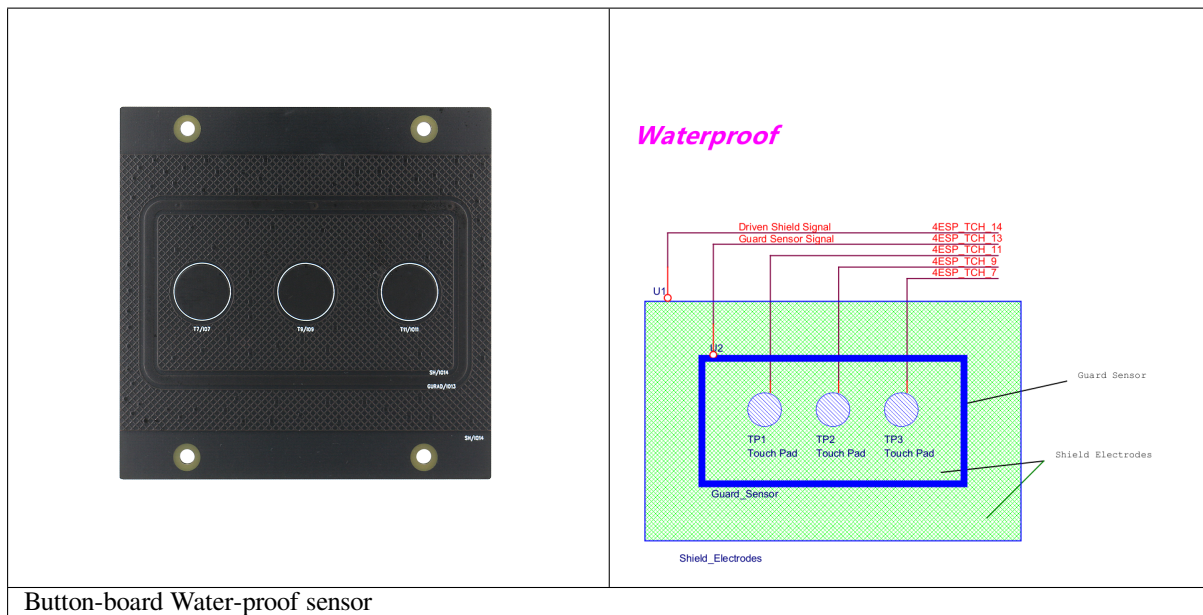
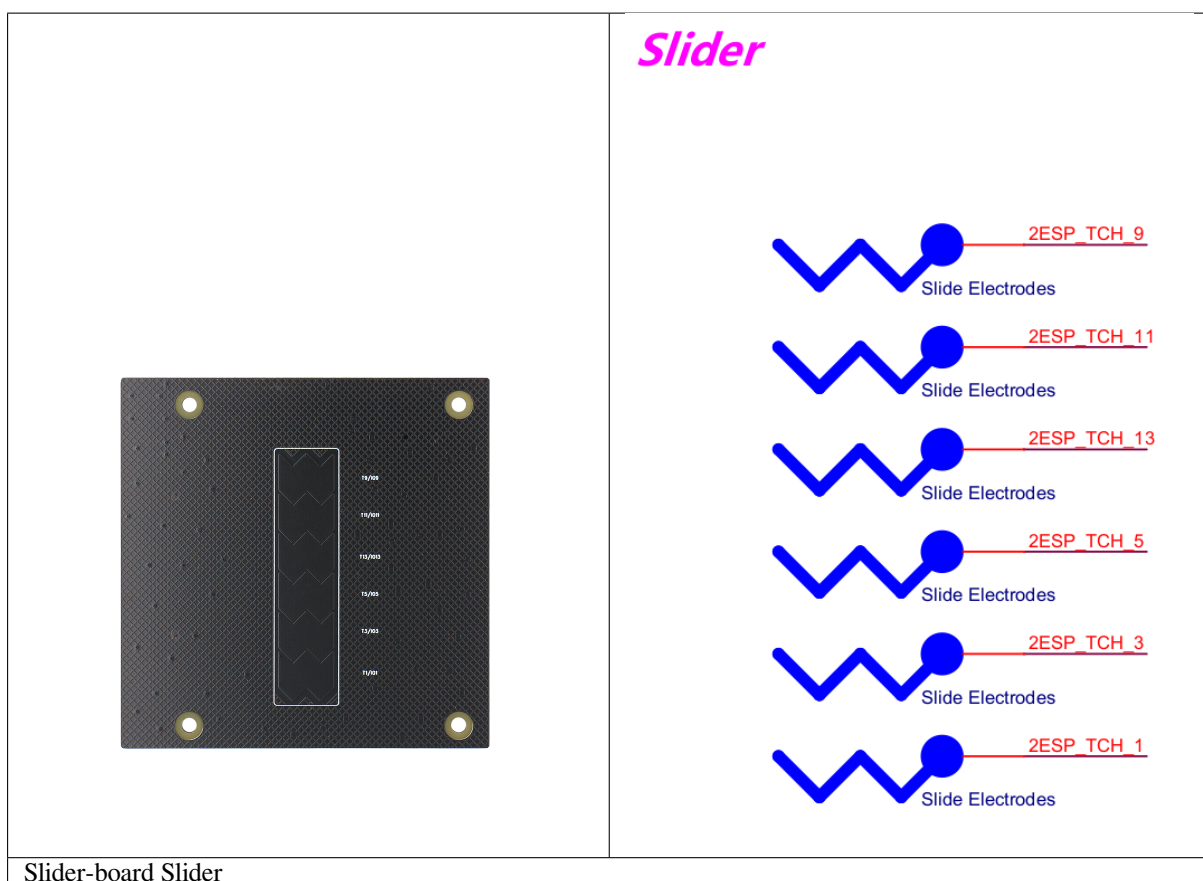


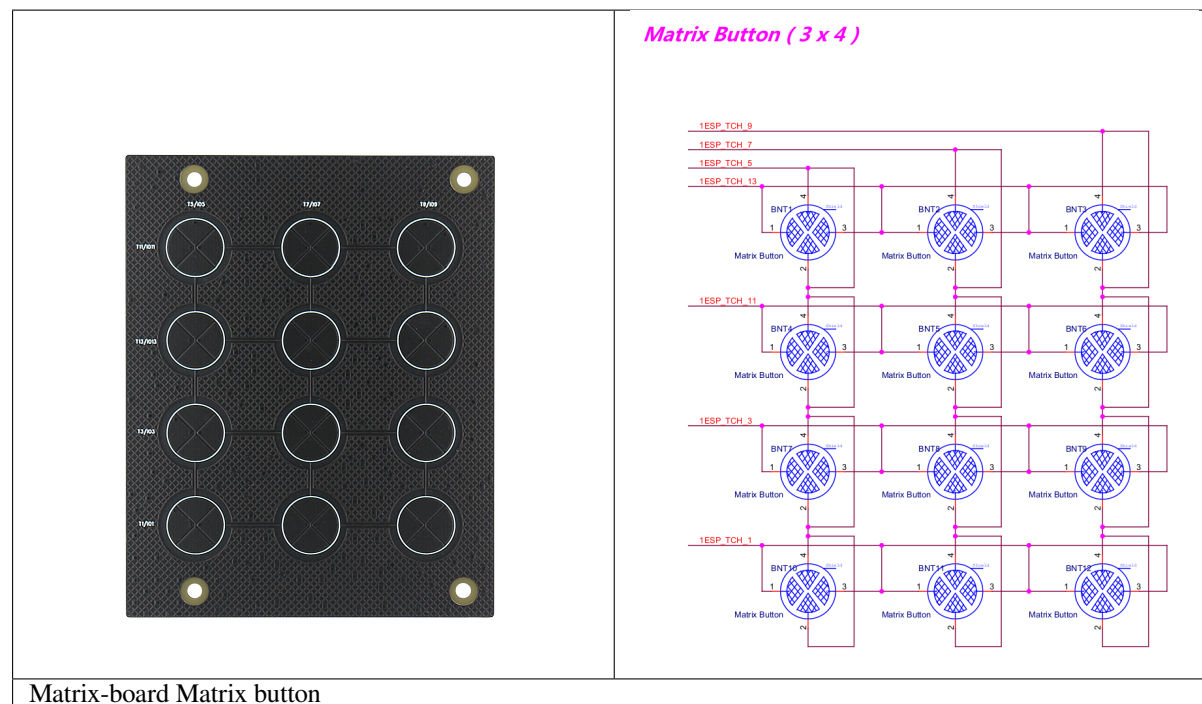
Fig. 14: Motherboard



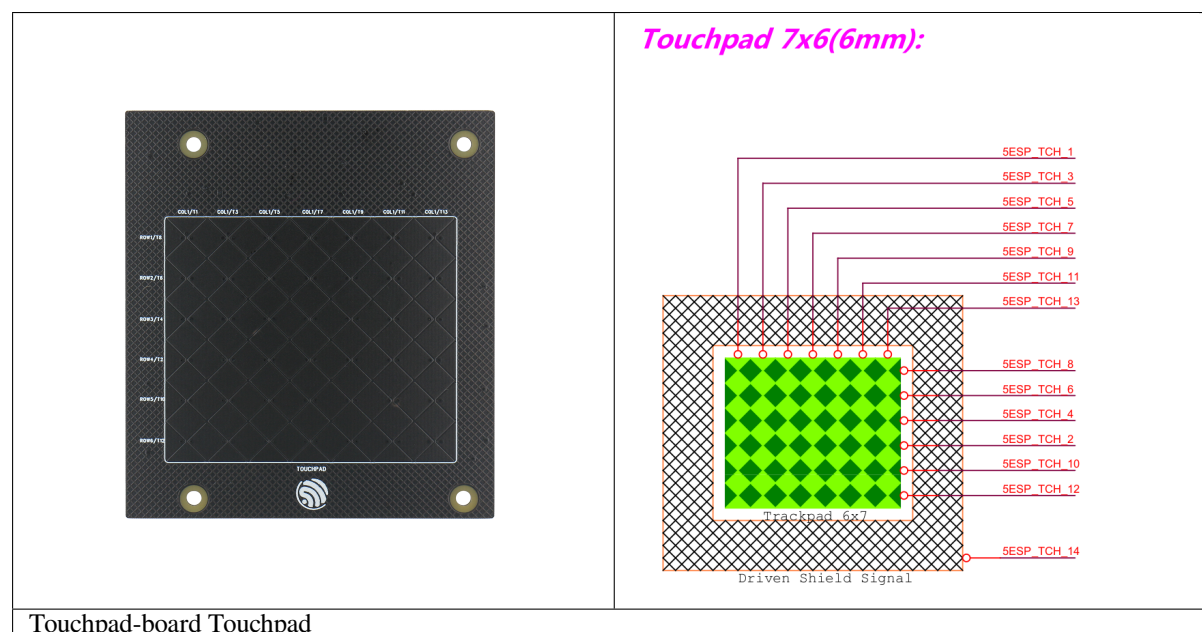
- Slider-board: Capacitive touch linear slider, the relative distance of slider has up to 8-bit precision.



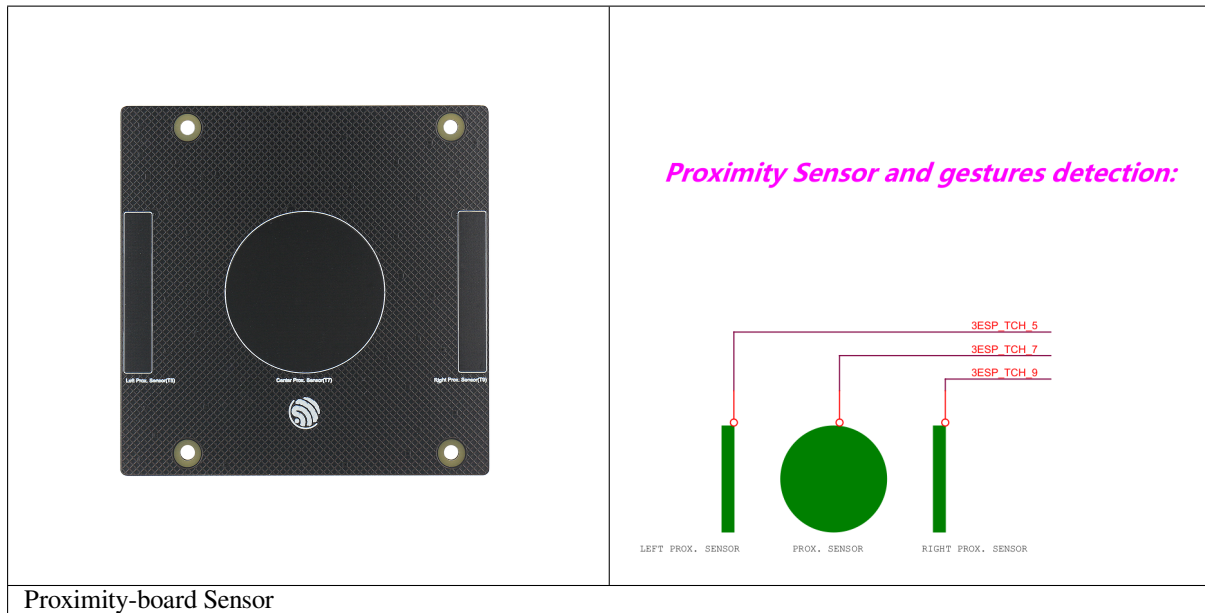
- Matrix-board: 3 x 4 capacitive touch matrix button, 7 channels of touch sensor make up of 12 capacitive touch buttons.



- Touchpad-board: 7 x 6 two-dimension capacitive touchpad.



- Proximity-board: Three capacitive touch proximity sensors.



Get Started

Install SDK and it's dependencies:

1. Install ESP-IDF, please refer to [ESP-IDF Programming Guide](#).
2. Use the release-4.3 version of ESP-IDF.

```
cd esp-idf #Enter esp-idf folder
git checkout release/v4.3 #Checkout release-v4.3 version of ESP-IDF
git submodule update #Maybe is needed
```

Get ESP32-S2-Touch-Devkit-1 project:

1. Clone esp-dev-kits repo.

```
git clone https://github.com/espressif/esp-dev-kits.git
```

2. Enter ESP32-S2-Touch-Devkit-1 project folder.

```
cd esp-dev-kits/esp32-s2-touch-devkit-1
```

The project' s structure is shown as followed:

```
├── CMakeLists.txt
├── components                #components' driver
│   ├── board_detect        #subboard detector
│   ├── buzzer              #buzzer driver
│   ├── digital_tube        #digital tube controller driver
│   ├── rgb_light           #RGB light driver(ws2812)
│   ├── subboards           #Subboards' application source file
│   └── touch_element        #Touch element library
├── main                     #Main application demo logic
│   ├── CMakeLists.txt
│   └── main.c
```

Build & Flash ESP32-S2-Touch-Devkit-1 demo project:

1. Enable ESP-IDF environment variable.

```
cd esp-idf
. ./export.sh
```

2. Build & Flash.

```
cd esp-dev-kits/esp32-s2-touch-devkit-1
idf.py set-target esp32s2 #Enable esp32s2 platform
idf.py build flash
```

3. Monitor log output.

```
idf.py monitor
```

Example output:

```
I (2880) Touch Demo: Slider sub-board plug in
I (22480) Touch Demo: Slider sub-board plug out
I (22480) Touch Demo: Nothing detected
I (41540) Touch Demo: Touchpad sub-board plug in
I (47700) Touchpad Board: Touchpad pressed, position: [0, 5]
I (47710) Touchpad Board: Position: [0, 5]
I (47720) Touchpad Board: Position: [0, 5]
I (47730) Touchpad Board: Position: [0, 6]
I (47740) Touchpad Board: Position: [0, 6]
I (47750) Touchpad Board: Position: [0, 6]
I (47760) Touchpad Board: Position: [0, 6]
I (47770) Touchpad Board: Position: [0, 7]
I (47780) Touchpad Board: Position: [0, 8]
I (47790) Touchpad Board: Position: [0, 9]
I (47800) Touchpad Board: Position: [0, 9]
I (47810) Touchpad Board: Position: [1, 10]
I (47820) Touchpad Board: Position: [2, 11]
I (47830) Touchpad Board: Position: [2, 12]
I (47840) Touchpad Board: Position: [3, 13]
I (47850) Touchpad Board: Position: [4, 14]
I (47860) Touchpad Board: Position: [5, 15]
I (47870) Touchpad Board: Position: [6, 16]
I (47880) Touchpad Board: Position: [7, 16]
I (47890) Touchpad Board: Position: [9, 17]
I (47900) Touchpad Board: Position: [10, 18]
I (47910) Touchpad Board: Position: [11, 18]
I (47920) Touchpad Board: Position: [11, 19]
I (47930) Touchpad Board: Position: [12, 20]
I (47940) Touchpad Board: Position: [13, 21]
I (47950) Touchpad Board: Position: [14, 21]
I (47960) Touchpad Board: Position: [14, 22]
I (47970) Touchpad Board: Position: [15, 23]
I (47980) Touchpad Board: Position: [15, 23]
I (47990) Touchpad Board: Position: [15, 24]
I (48000) Touchpad Board: Position: [15, 24]
I (48010) Touchpad Board: Position: [15, 24]
I (48020) Touchpad Board: Position: [16, 24]
I (48030) Touchpad Board: Position: [16, 24]
I (48040) Touchpad Board: Position: [16, 24]
I (48050) Touchpad Board: Position: [16, 23]
I (48060) Touchpad Board: Position: [16, 23]
I (48070) Touchpad Board: Position: [16, 22]
I (48080) Touchpad Board: Position: [16, 22]
I (48090) Touchpad Board: Position: [16, 21]
I (48100) Touchpad Board: Touchpad released, position: [16, 21]
```

Notes

- Some new Touch Sensor features (Touchpad, Touch proximity) are not supported in ESP-IDF [Touch Element](#) library, so we copy Touch Element from ESP-IDF components into this demo project's components' folder and add the necessary features. They will appear in the future version of ESP-IDF.

Troubleshooting

Q1: Why Proximity-board is connected with Motherboard, they don't work or work abnormally?

A1: Though all of those subboards are hot-swappable theoretically, the Proximity-board needs to startup in an ideal environment(Far away from your hands). If it goes wrong, you can reset it mandatorily by releasing the power-switch.

Related Documents

Schematic

- [ESP32-S2-Touch-Devkit-1 Motherboard Schematic](#)
- [Button Subboard Schematic](#)
- [Slider Subboard Schematic](#)
- [Matrix Button Subboard Schematic](#)
- [Touchpad Subboard Schematic](#)
- [Proximity Subboard Schematic](#)

Other Documents

- [Touch Element Library Programming Guide](#)
- [ESP32-S2-MINI-1 Datasheet](#)
- [ESP32-S2 Datasheet](#)
- [ESP32-S2 Technical Reference Manual](#)

3.3 ESP32-S2-Kaluga-1

The ESP32-S2-Kaluga-1 kit is a development kit by Espressif.

Application examples for this kit can be found at [Examples](#).

3.3.1 ESP32-S2-Kaluga-1 Kit v1.3

Older version: [ESP32-S2-Kaluga-1 Kit v1.2](#)

The ESP32-S2-Kaluga-1 kit v1.3 is a development kit by Espressif that is mainly created to:

- Demonstrate the ESP32-S2's human-computer interaction functionalities
- Provide the users with the tools for development of human-computer interaction applications based on the ESP32-S2

There are many ways of how the ESP32-S2's abundant functionalities can be used. For starters, the possible use cases may include:

- **Smart home:** From simplest smart lighting, smart door locks, smart sockets, to video streaming devices, security cameras, OTT devices, and home appliances
- **Battery-powered equipment:** Wi-Fi mesh sensor networks, Wi-Fi-networked toys, wearable devices, health management equipment
- **Industrial automation equipment:** Wireless control and robot technology, intelligent lighting, HVAC control equipment, etc.
- **Retail and catering industry:** POS machines and service robots

The ESP32-S2-Kaluga-1 kit consists of the following boards:

- Main board: *ESP32-S2-Kaluga-1*
- Extension boards:

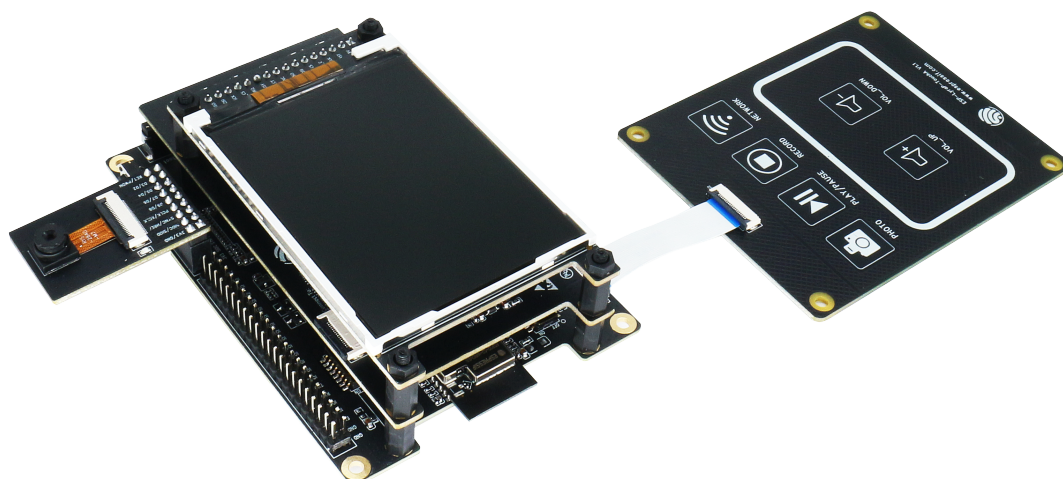


Fig. 15: ESP32-S2-Kaluga-1-Kit Overview (click to enlarge)

- [ESP-LyraT-8311A v1.3](#) - audio player
- [ESP-LyraP-TouchA v1.1](#) - touch panel
- [ESP-LyraP-LCD32 v1.2](#) - 3.2" LCD screen
- [ESP-LyraP-CAM v1.1](#) - camera board

Due to the presence of multiplexed pins on ESP32-S2, certain extension board combinations have limited compatibility. For more details, please see [Compatibility of Extension Boards](#).

This document is **mostly dedicated to the main board** and its interaction with the extension boards. For more detailed information on each extension board, click their respective links.

This guide covers:

- [Getting Started](#): Provides an overview of the ESP32-S2-Kaluga-1 and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP32-S2-Kaluga-1's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP32-S2-Kaluga-1.
- [Related Documents](#): Gives links to related documentation.

Getting Started

This section describes how to get started with the ESP32-S2-Kaluga-1. It begins with a few introductory sections about the ESP32-S2-Kaluga-1, then Section [Start Application Development](#) provides instructions on how to do the initial hardware setup and then how to flash firmware onto the ESP32-S2-Kaluga-1.

Overview The ESP32-S2-Kaluga-1 main board is the heart of the kit. It integrates the ESP32-S2-WROVER module and all the connectors for extension boards. This board is the key tool in prototyping human-computer interaction interfaces.

The ESP32-S2-Kaluga-1 board has connectors for boards with:

- Extension header (ESP-LyraT-8311A, ESP-LyraP-LCD32)
- Camera header (ESP-LyraP-CAM)
- Touch FPC connector (ESP-LyraP-TouchA)
- LCD FPC connector (no official extension boards yet)
- I2C FPC connector (no official extension boards yet)

All the four extension boards are specially designed to support the following features:

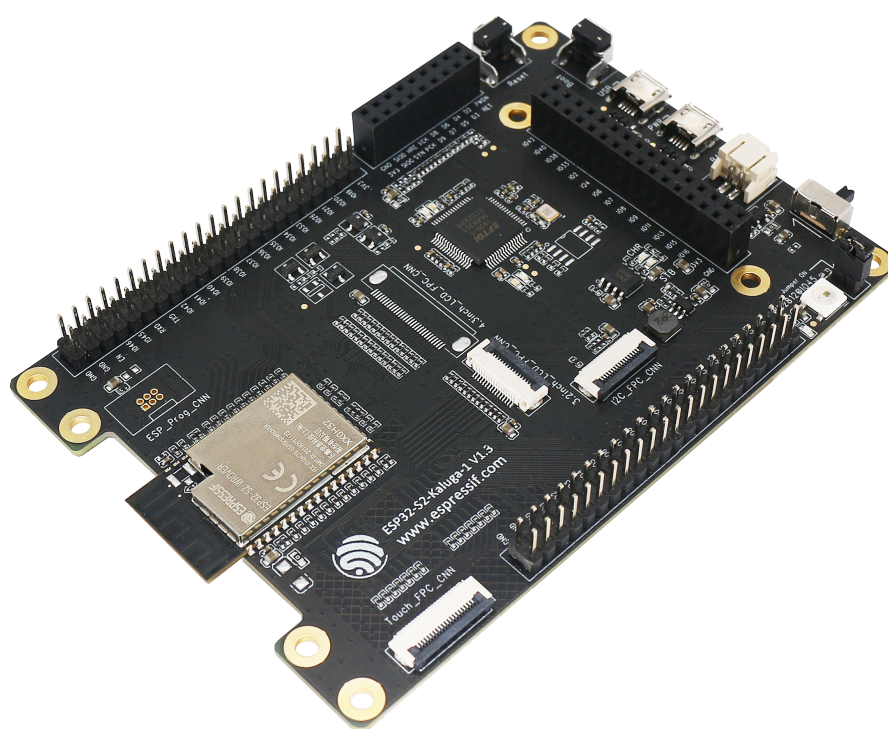


Fig. 16: ESP32-S2-Kaluga-1 (click to enlarge)

- **Touch panel control**
 - Six touch buttons
 - Supports acrylic panels up to 5 mm
 - Wet hand operation
 - Water rejection, ESP32-S2 can be configured to disable all touchpads automatically if multiple pads are simultaneously covered with water and to re-enable touchpads if the water is removed
- **Audio playback**
 - Connect speakers to play audio
 - Use together with the Touch panel to control audio playback and adjust volume
- **LCD display**
 - LCD interface (8-bit parallel RGB, 8080, and 6800 interface)
- **Camera image acquisition**
 - Supports OV2640 and OV3660 camera modules
 - 8-bit DVP image sensor interface (ESP32-S2 also supports 16-bit DVP image sensors, you can design it yourself)
 - Clock frequency up to 40 MHz
 - Optimized DMA transmission bandwidth for easier transmission of high-resolution images

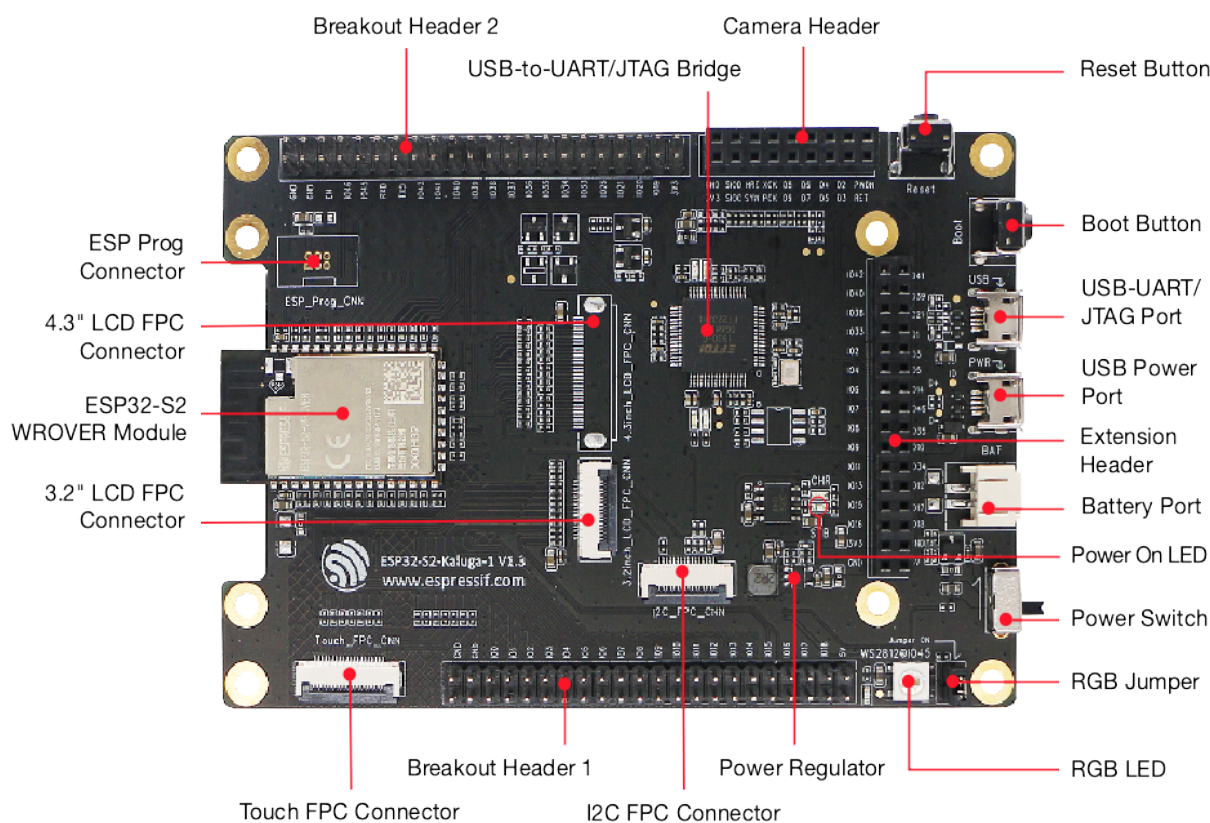


Fig. 17: ESP32-S2-Kaluga-1 - front (click to enlarge)

Description of Components The description of components starts from the ESP32-S2 module on the left side and then goes clockwise.

Reserved means that the functionality is available, but the current version of the kit does not use it.

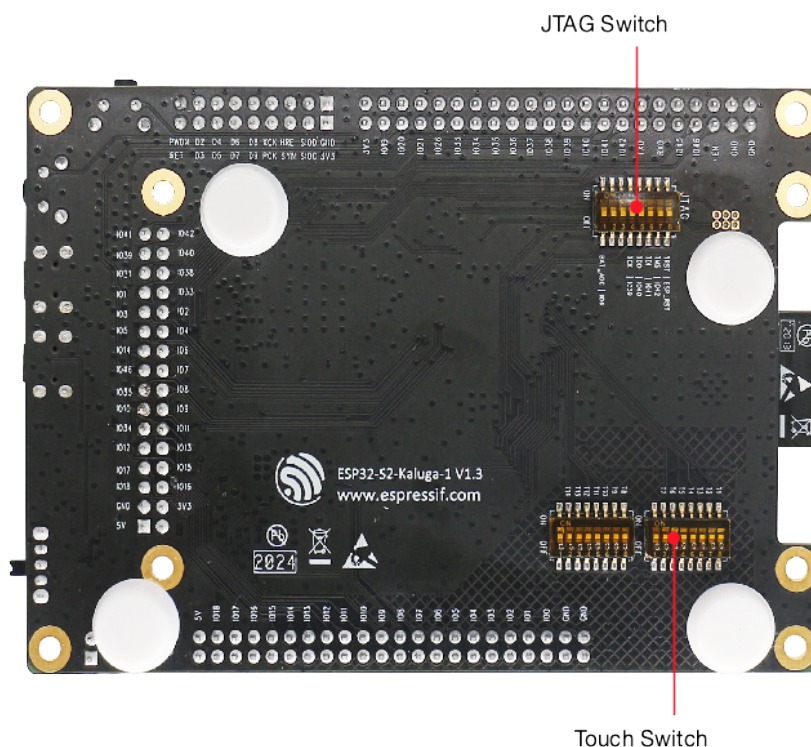


Fig. 18: ESP32-S2-Kaluga-1 - back (click to enlarge)

Key Component	Description
ESP32-S2-WROVER Module	Module integrating the ESP32-S2 chip that provides Wi-Fi connectivity, data processing power, and flexible data storage.
4.3” LCD FPC Connector	(Reserved) Connect to a 4.3” LCD extension board using the FPC cable.
ESP Prog Connector	(Reserved) Connection for Espressif’s download device (ESP-Prog) to flash ESP32-S2 system.
JTAG Switch	Switch to ON to enable connection between ESP32-S2 and FT2232; JTAG debugging will then be possible using USB-UART/JTAG Port. See also JTAG Debugging .
Breakout Header 2	Some GPIO pins of the ESP32-S2-WROVER module are broken out to this header, see labels on the board.
USB-to-UART/JTAG Bridge	FT2232 adapter board allowing for communication over USB port using UART/JTAG protocols.
Camera Header	Mount a camera extension board here (e.g., ESP-LyraP-CAM).
Extension Header	Mount the extension boards having such connectors here.
Reset Button	Press this button to restart the system
Boot Button	Holding down Boot and then pressing Reset initiates Firmware Download mode for downloading firmware through the serial port.
USB-UART/JTAG Port	Communication interface (UART or JTAG) between a PC and the ESP32-S2 module.
USB Power Port	Power supply for the board.
Battery Port	Connect an external battery to the 2-pin battery connector.
Power On LED	Turns on when the USB or an external power supply is connected to the board.
Power Switch	Switch to ON to power the system.
RGB Jumper	To have access to the RGB LED, place a jumper onto the pins.
RGB LED	Programmable RGB LED and controlled by GPIO45. Before using it, you need to put RGB Jumper ON.
Power Regulator	Regulator converts 5 V to 3.3 V.
I2C FPC Connector	(Reserved) Connect to other I2C extension boards using the FPC cable.
Breakout Header 1	Some GPIO pins of the ESP32-S2-WROVER module are broken out to this header, see labels on the board.
ESP32-S2 FPC Connector	Connect the ESP-LyraP-TouchA extension board using the FPC cable.
Touch Switch	In Submit Document Feedback mode, can be used for connection to touch sensors; switch to ON if you want to use them for other purposes.
3.2” LCD FPC connector	Connect a 3.2” LCD extension board (e.g., ESP-LyraP-LCD32) using the FPC cable.

Start Application Development Before powering up your ESP32-S2-Kaluga-1, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- ESP32-S2-Kaluga-1
- Two USB 2.0 cables (Standard-A to Micro-B)
 - For power supply
 - For UART/JTAG communication
- Computer running Windows, Linux, or macOS
- Any extension boards of your choice

Hardware Setup

1. Connect the extension board(s) of your choice (go to their respective user guides if necessary)
2. Plug in both USB cables
3. Turn the **Power Switch** to ON - the Power On LED will light up

Software Setup Please proceed to [Get Started](#), where Section [Installation](#) will quickly help you set up the development environment.

The programming guide and application examples for your ESP32-S2-Kaluga-1 kit can be found in [Examples](#) on GitHub.

A Board Support Package can be found in [ESP Component Registry](#).

Contents and Packaging

Retail Orders If you order one or several samples of the kit, each ESP32-S2-Kaluga-1 development kit comes in an individual package.

The contents are as follows:

- **Main Board**
 - ESP32-S2-Kaluga-1
- **Extension Boards:**
 - ESP-LyraT-8311A
 - ESP-LyraP-CAM
 - ESP-LyraP-TouchA
 - ESP-LyraP-LCD32
- **Connectors**
 - 20-pin FPC cable (to connect ESP32-S2-Kaluga-1 to ESP-LyraP-TouchA)
- **Fasteners**
 - Mounting bolts (x8)
 - Screws (x4)
 - Nuts (x4)

For retail orders, please go to <https://www.espressif.com/en/contact-us/get-samples>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP32-S2-Kaluga-1 and their interconnections.



Fig. 19: ESP32-S2-Kaluga-1 - package

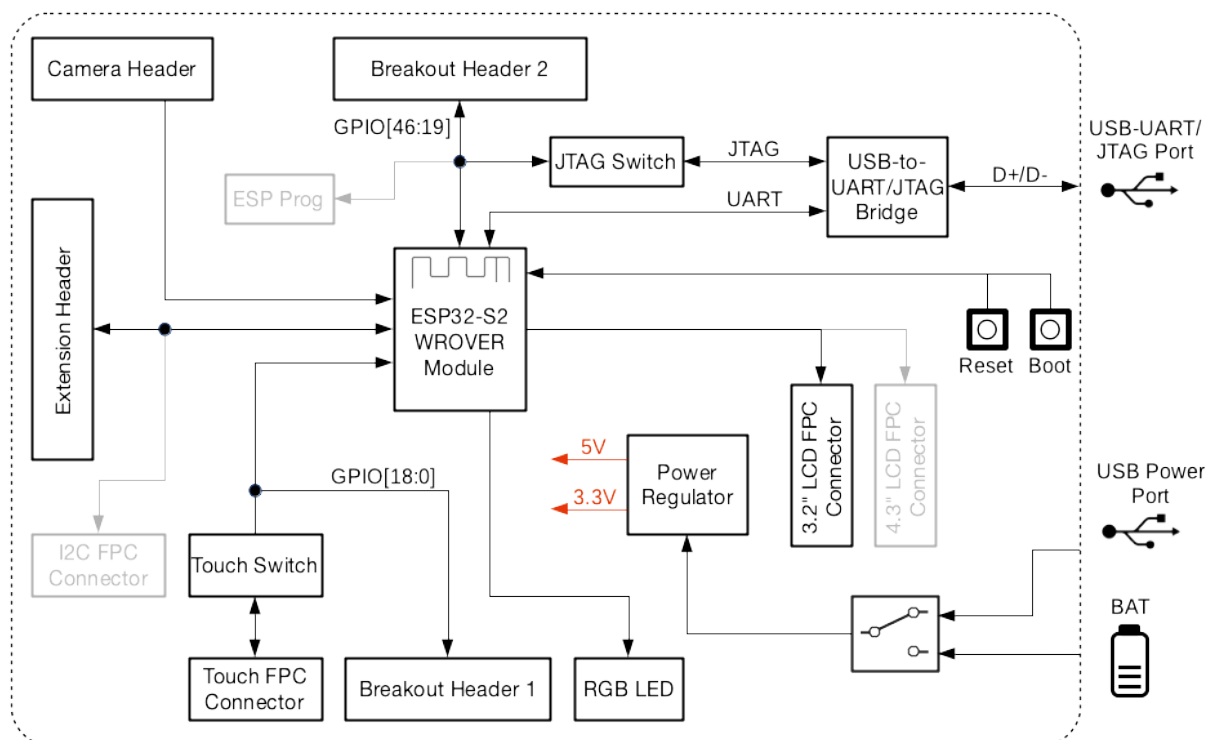


Fig. 20: ESP32-S2-Kaluga-1 block diagram

Power Supply Options There are four ways to provide power to the board:

- Micro USB port, default power supply
- External battery via the 2-pin battery connector
- 5V and GND header pins
- 3V3 and GND header pins

Compatibility of Extension Boards If you want to use more than one extension board at the same time, please check the table given below.

Boards Used	HW Conflict	Limitations	Solution
8311A v1.3 + CAM v1.1	I2S Controller	ESP32-S2 has only one I2S interface. But both extension boards require connection via the ESP32-S2's I2S interface (LyraT-8311A in Standard mode, ESP-LyraP-CAM in Camera mode).	Utilize time division multiple access, or use a different audio module that can be connected via other GPIOs or DAC.
TouchA v1.1 + LCD32 v1.2	IO11, IO6	Touch actions cannot be triggered because of the multiplexed pin IO11. ESP-LyraP-LCD32 will not be affected because its BLCT pin will be disconnected from IO6.	Do not initialize IO11 (NETWORK) for your ESP-LyraP-TouchA, or configure the BLCT pin to -1 (= do not use BLCT) for your ESP-LyraP-LCD32.
8311A v1.3 + LCD32 v1.2	IO6	BLCT pin of ESP32-S2-Kaluga-1 will be disconnected from IO6.	Configure the BK pin to -1 (= do not use BLCT) for your ESP-LyraP-LCD32.
TouchA v1.1 + 8311A v1.3	Pin BT_ADC on ESP-LyraT-8311A	This pin is required for initialization of the six button on ESP-LyraT-8311A. At the same time, ESP-LyraP-TouchA needs this pin for its touch actions.	If you plan to use buttons on ESP-LyraT-8311A, do not initialize pin IO6 (PHOTO) for your ESP-LyraP-TouchA.
TouchA v1.1 + CAM v1.1	IO1, IO2, IO3	Cannot be used simultaneously because of the mentioned multiplexed pins.	For ESP-LyraP-TouchA, do not initialize IO1 (VOL_UP), IO2 (PLAY), and IO3 (VOL_DOWN).
TouchA v1.1 + LCD32 v1.2 + CAM v1.1	IO1, IO2, IO3, IO11	Conflicts on the mentioned multiplexed pins.	For ESP-LyraP-TouchA, do not initialize IO1 (VOL_UP), IO2 (PLAY), IO3 (VOL_DOWN), and IO11 (NETWORK).
TouchA v1.1 + LCD32 v1.2 + 8311A v1.3	IO6, IO11	If ESP-LyraT-8311A's pin BT_ADC is used to initialize the board's six buttons, IO6 and IO11 will not be available for the other boards.	Do not initialize IO11 (NETWORK) for your ESP-LyraP-TouchA. Also, if you need to use BT_ADC, do not initialize IO6 (PHOTO).

Also, all extension boards and the [JTAG interface](#) share the same pins IO39, IO40, IO41 and IO42. For this reason, the following may disturb the JTAG operation:

- Plugging in any extension board
- Debugging an application that is using an extension board

Hardware Revision Details

ESP32-S2-Kaluga-1 Kit v1.3

- The following pins re-assigned to fix the download issue
 - Camera D2: GPIO36
 - Camera D3: GPIO37
 - AU_I2S1_SDI: GPIO34
 - AU_WAKE_INT: GPIO46
- RGB pin header moved to the board's edge
- All dip switches moved to the flip side for convenient operation

ESP32-S2-Kaluga-1 Kit v1.2 *Initial release*

Related Documents

ESP32-S2-Kaluga-1 Kit v1.2

New version available: [ESP32-S2-Kaluga-1 Kit v1.3](#)

The ESP32-S2-Kaluga-1 kit v1.2 is a development kit by Espressif that is mainly created to:

- Demonstrate the ESP32-S2's human-computer interaction functionalities
- Provide the users with the tools for development of human-computer interaction applications based on the ESP32-S2

There are many ways of how the ESP32-S2's abundant functionalities can be used. For starters, the possible use cases may include:

- **Smart home:** From simplest smart lighting, smart door locks, smart sockets, to video streaming devices, security cameras, OTT devices, and home appliances
- **Battery-powered equipment:** Wi-Fi mesh sensor networks, Wi-Fi-networked toys, wearable devices, health management equipment
- **Industrial automation equipment:** Wireless control and robot technology, intelligent lighting, HVAC control equipment, etc.
- **Retail and catering industry:** POS machines and service robots

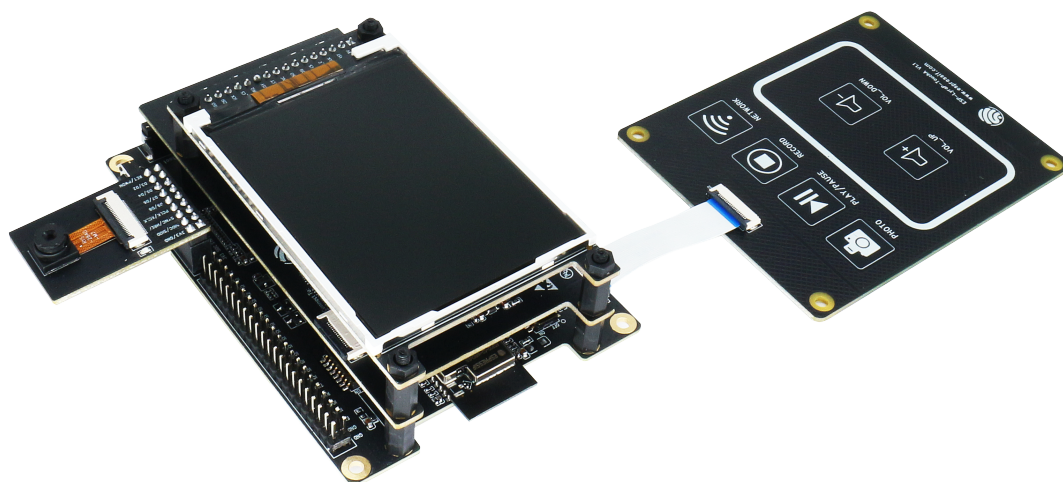


Fig. 21: ESP32-S2-Kaluga-1-Kit Overview (click to enlarge)

The ESP32-S2-Kaluga-1 kit consists of the following boards:

- Main board: *ESP32-S2-Kaluga-1*
- Extension boards:
 - *ESP-LyraT-8311A v1.2* - audio player
 - *ESP-LyraP-TouchA v1.1* - touch panel
 - *ESP-LyraP-LCD32 v1.1* - 3.2" LCD screen
 - *ESP-LyraP-CAM v1.0* - camera board

Due to the presence of multiplexed pins on ESP32-S2, certain extension board combinations have limited compatibility. For more details, please see [Compatibility of Extension Boards](#).

This document is **mostly dedicated to the main board** and its interaction with the extension boards. For more detailed information on each extension board, click their respective links.

This guide covers:

- [Getting Started](#): Provides an overview of the ESP32-S2-Kaluga-1 and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP32-S2-Kaluga-1's hardware.

- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP32-S2-Kaluga-1.
- [Related Documents](#): Gives links to related documentation.

Getting Started This section describes how to get started with the ESP32-S2-Kaluga-1. It begins with a few introductory sections about the ESP32-S2-Kaluga-1, then Section [Start Application Development](#) provides instructions on how to do the initial hardware setup and then how to flash firmware onto the ESP32-S2-Kaluga-1.

Overview The ESP32-S2-Kaluga-1 main board is the heart of the kit. It integrates the ESP32-S2-WROVER module and all the connectors for extension boards. This board is the key tool in prototyping human-computer interaction interfaces.

The ESP32-S2-Kaluga-1 board has connectors for boards with:

- Extension header (ESP-LyraT-8311A, ESP-LyraP-LCD32)
- Camera header (ESP-LyraP-CAM)
- Touch FPC connector (ESP-LyraP-TouchA)
- LCD FPC connector (no official extension boards yet)
- I2C FPC connector (no official extension boards yet)

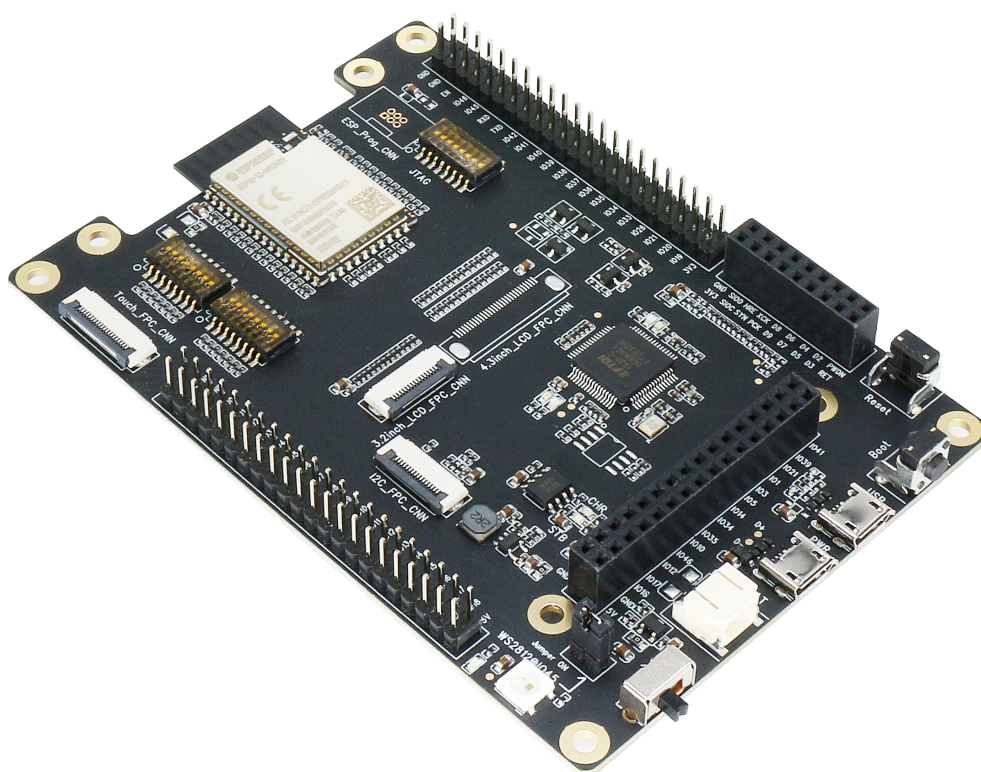


Fig. 22: ESP32-S2-Kaluga-1 (click to enlarge)

All the four extension boards are specially designed to support the following features:

- **Touch panel control**
 - Six touch buttons
 - Supports acrylic panels up to 5 mm
 - Wet hand operation

- Water rejection, ESP32-S2 can be configured to disable all touchpads automatically if multiple pads are simultaneously covered with water and to re-enable touchpads if the water is removed
- **Audio playback**
 - Connect speakers to play audio
 - Use together with the Touch panel to control audio playback and adjust volume
- **LCD display**
 - LCD interface (8-bit parallel RGB, 8080, and 6800 interface)
- **Camera image acquisition**
 - Supports OV2640 and OV3660 camera modules
 - 8-bit DVP image sensor interface (ESP32-S2 also supports 16-bit DVP image sensors, you can design it yourself)
 - Clock frequency up to 40 MHz
 - Optimized DMA transmission bandwidth for easier transmission of high-resolution images

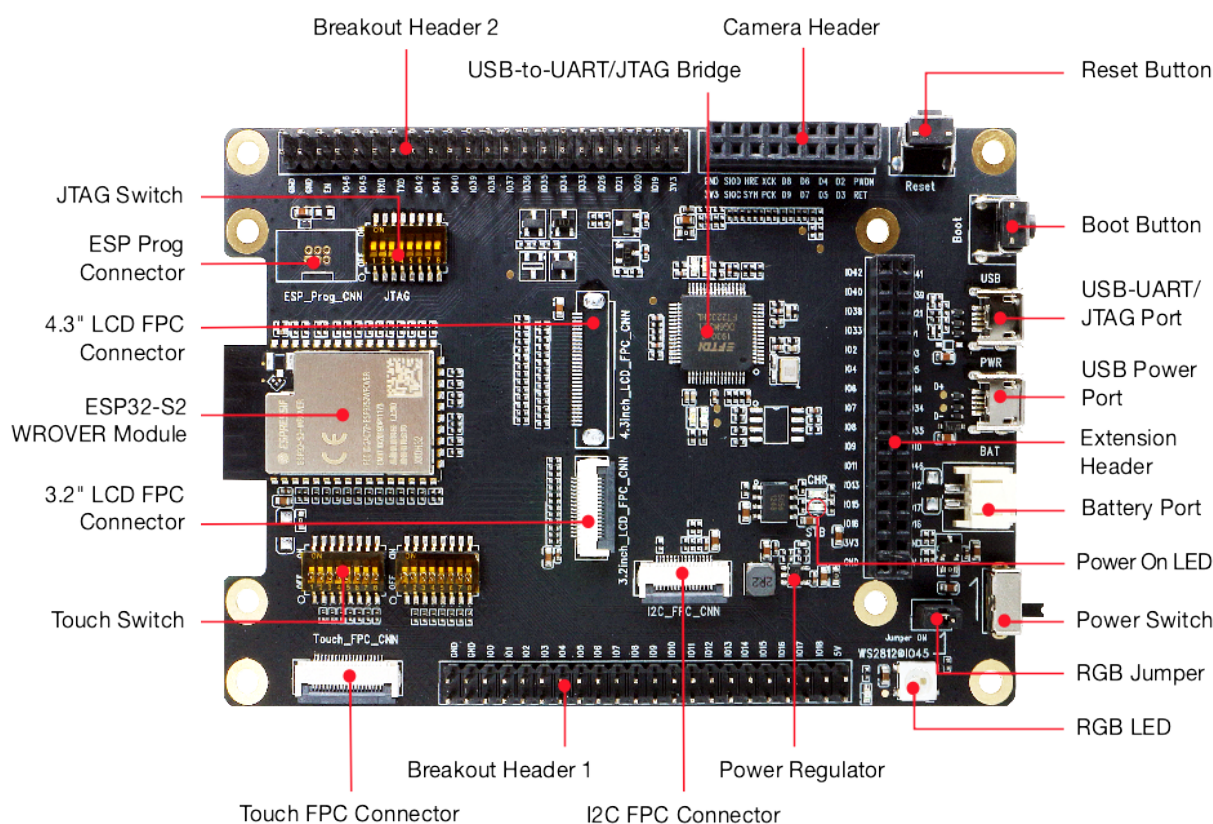


Fig. 23: ESP32-S2-Kaluga-1 - front (click to enlarge)

Description of Components The description of components starts from the ESP32-S2 module on the left side and then goes clockwise.

Reserved means that the functionality is available, but the current version of the kit does not use it.

Key Component	Description
ESP32-S2-WROVER Module	Module integrating the ESP32-S2 chip that provides Wi-Fi connectivity, data processing power, and flexible data storage.
4.3" LCD FPC Connector	(Reserved) Connect to a 4.3" LCD extension board using the FPC cable.
ESP Prog Connector	(Reserved) Connection for Espressif's download device (ESP-Prog) to flash ESP32-S2 system.
JTAG Switch	Switch to ON to enable connection between ESP32-S2 and FT2232; JTAG debugging will then be possible using USB-UART/JTAG Port. See also JTAG Debugging .
Breakout Header 2	Some GPIO pins of the ESP32-S2-WROVER module are broken out to this header, see labels on the board.
USB-to-UART/JTAG Bridge	FT2232 adapter board allowing for communication over USB port using UART/JTAG protocols.
Camera Header	Mount a camera extension board here (e.g., ESP-LyraP-CAM).
Extension Header	Mount the extension boards having such connectors here.
Reset Button	Press this button to restart the system.
Boot Button	Holding down Boot and then pressing Reset initiates Firmware Download mode for downloading firmware through the serial port.
USB-UART/JTAG Port	Communication interface (UART or JTAG) between a PC and the ESP32-S2 module.
USB Power Port	Power supply for the board.
Battery Port	Connect an external battery to the 2-pin battery connector.
Power On LED	Turns on when the USB or an external power supply is connected to the board.
Power Switch	Switch to ON to power the system.
RGB Jumper	To have access to the RGB LED, place a jumper onto the pins.
RGB LED	Programmable RGB LED and controlled by GPIO45. Before using it, you need to put RGB Jumper ON.
Power Regulator	Regulator converts 5 V to 3.3 V.
I2C FPC Connector	(Reserved) Connect to other I2C extension boards using the FPC cable.
Breakout Header 1	Some GPIO pins of the ESP32-S2-WROVER module are broken out to this header, see labels on the board.
Touch FPC Connector	Connect the ESP-LyraP-TouchA extension board using the FPC cable.
Touch Switch	In OFF position, GPIO1 to GPIO14 are used for connection to touch sensors; switch to ON if you want to use them for other purposes.
3.2" LCD FPC connector	Connect a 3.2" LCD extension board (e.g., ESP-LyraP-LCD32) using the FPC cable.

Start Application Development Before powering up your ESP32-S2-Kaluga-1, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- ESP32-S2-Kaluga-1
- Two USB 2.0 cables (Standard-A to Micro-B)
 - For power supply
 - For UART/JTAG communication
- Computer running Windows, Linux, or macOS
- Any extension boards of your choice

Hardware Setup

1. Connect the extension board(s) of your choice (go to their respective user guides if necessary)
2. Plug in both USB cables
3. Turn the **Power Switch** to ON - the Power On LED will light up

Software Setup Please proceed to [Get Started](#), where Section [Installation](#) will quickly help you set up the development environment.

The programming guide and application examples for your ESP32-S2-Kaluga-1 kit can be found in [Examples](#) on GitHub.

Contents and Packaging

Retail Orders If you order one or several samples of the kit, each ESP32-S2-Kaluga-1 development kit comes in an individual package containing:

- **Main Board**
 - ESP32-S2-Kaluga-1
- **Extension Boards:**
 - ESP-LyraT-8311A
 - ESP-LyraP-CAM
 - ESP-LyraP-TouchA
 - ESP-LyraP-LCD32
- **Connectors**
 - 20-pin FPC cable (to connect ESP32-S2-Kaluga-1 to ESP-LyraP-TouchA)
- **Fasteners**
 - Mounting bolts (x8)
 - Screws (x4)
 - Nuts (x4)

For retail orders, please go to <https://www.espressif.com/en/contact-us/get-samples>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP32-S2-Kaluga-1 and their interconnections.

Power Supply Options There are four ways to provide power to the board:

- Micro USB port, default power supply
- External battery via the 2-pin battery connector
- 5V and GND header pins
- 3V3 and GND header pins

Compatibility of Extension Boards If you want to use more than one extension board at the same time, please check the table given below.

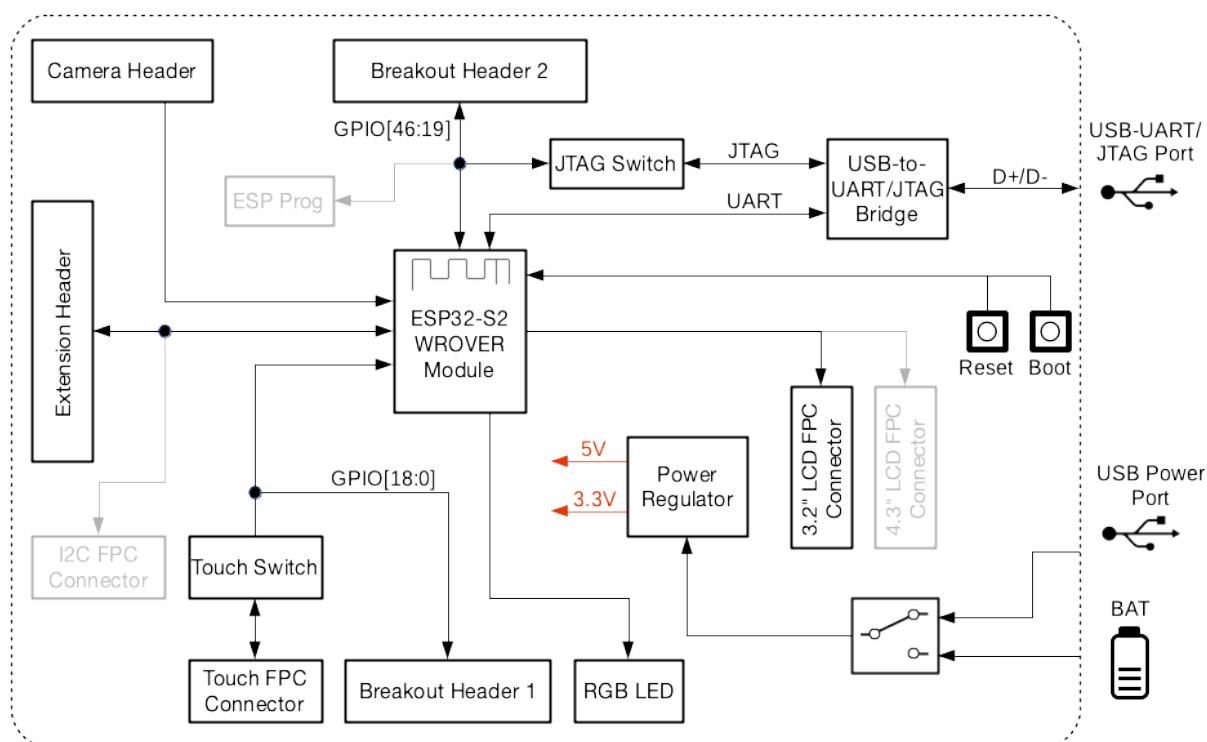


Fig. 24: ESP32-S2-Kaluga-1 block diagram

Boards Used	HW Conflict	Limitations	Solution
8311A v1.2 + CAM v1.0	I2S Controller, IO46	ESP32-S2 has only one I2S interface. But both extension boards require connection via the ESP32-S2's I2S interface (ESP-LyraT-8311A in Standard mode, ESP-LyraP-CAM in Camera mode). If IO46 is used by both extension boards at the same time, ESP-LyraP-CAM experiences interferences when used.	No ready solution for now.
TouchA v1.1 + LCD32 v1.1	IO11, IO6	Touch actions cannot be triggered because of the multiplexed pin IO11. ESP-LyraP-LCD32 is also affected because its BK (BLCT) pin is connected to pin IO6.	Do not initialize IO11 (NETWORK) and IO6 (PHOTO) for your ESP-LyraP-TouchA.
8311A v1.2 + LCD32 v1.1	IO6	The two extension boards can be used at the same time. However, since the BK (BLCT) pin of ESP32-S2-Kaluga-1 is connected to IO6, ESP-LyraT-8311A's pin BT_ADC cannot be used and the board's six buttons will not be available.	There is a solution that will allow you to use ESP-LyraT-8311A's pin BT_ADC, but will stop you from controlling the display background brightness with software: on your ESP-LyraP-LCD32 board, remove R39, change R41 to 100 Ohm, switch BLCT_L to on.
TouchA v1.1 + 8311A v1.2	Pin BT_ADC on ESP-LyraT-8311A	The two extension boards can be used at the same time. However, ESP-LyraP-TouchA cannot be triggered if ESP-LyraT-8311A's pin BT_ADC is used	If you plan to use ESP-LyraT-8311A's pin BT_ADC, do not initialize pin IO6 (PHOTO) for your ESP-LyraP-TouchA.
Espressif Systems		to initialize the board's six buttons	Release master
TouchA v1.1 + CAM v1.0	IO1, IO2, IO3	Cannot be used simultaneously because of the mentioned multiplexed pins	For ESP-LyraP-TouchA, do not initialize IO1 (VOL_UP), IO2 (PLAY), and IO3 (VOL_DOWN)

Also, all extension boards and the [JTAG interface](#) share the same pins IO39, IO40, IO41 and IO42. For this reason, the following may disturb the JTAG operation:

- Plugging in any extension board
- Debugging an application that is using an extension board

Known Issues

Hardware Issue	Description	Reason for Failure	Solution
ESP-LyraP-CAM v1.0, pin IO45, IO46	Flashing firmware might be impossible with the extension board connected to the main board.	Incorrect timing sequence is fed to strapping pins IO45 and IO46 when the board is powered on. It stops the board from booting successfully.	While flashing the main board, keep the extension board disconnected.
ESP-LyraP-CAM v1.0, pin IO45, IO46	Rebooting the board by pressing Reset might not lead to desired results.	Incorrect timing sequence is fed to strapping pins IO45 and IO46 when the board is powered on. It stops the board from booting successfully.	No ready solution for v1.2. This bug is fixed in ESP32-S2-Kaluga-1 V1.3.
ESP-LyraT-8311A v1.2, pin IO46	Flashing firmware might be impossible with the extension board connected to the main board.	Incorrect timing sequence is fed to strapping pin IO46 when the board is powered on. It stops the board from booting successfully.	While flashing the main board, keep the extension board disconnected.
ESP-LyraT-8311A v1.2, pin IO46	Rebooting the board by pressing Reset might not lead to desired results.	Incorrect timing sequence is fed to strapping pin IO46 when the board is powered on. It stops the board from booting successfully.	No ready solution for v1.2. This bug is fixed in ESP32-S2-Kaluga-1 V1.3.

Hardware Revision Details No previous versions available.

Related Documents

ESP-LyraP-CAM v1.0

This user guide provides information on the ESP-LyraP-CAM extension board.

This extension board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraP-CAM v1.0 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.2](#).

The ESP-LyraP-CAM extends the functionality of your main board by adding a camera.

The document consists of the following major sections:

- [Overview](#): Provides an overview and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP-LyraP-CAM's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraP-CAM.
- [Related Documents](#): Gives links to related documentation.

Overview This extension board adds a camera to your main board.

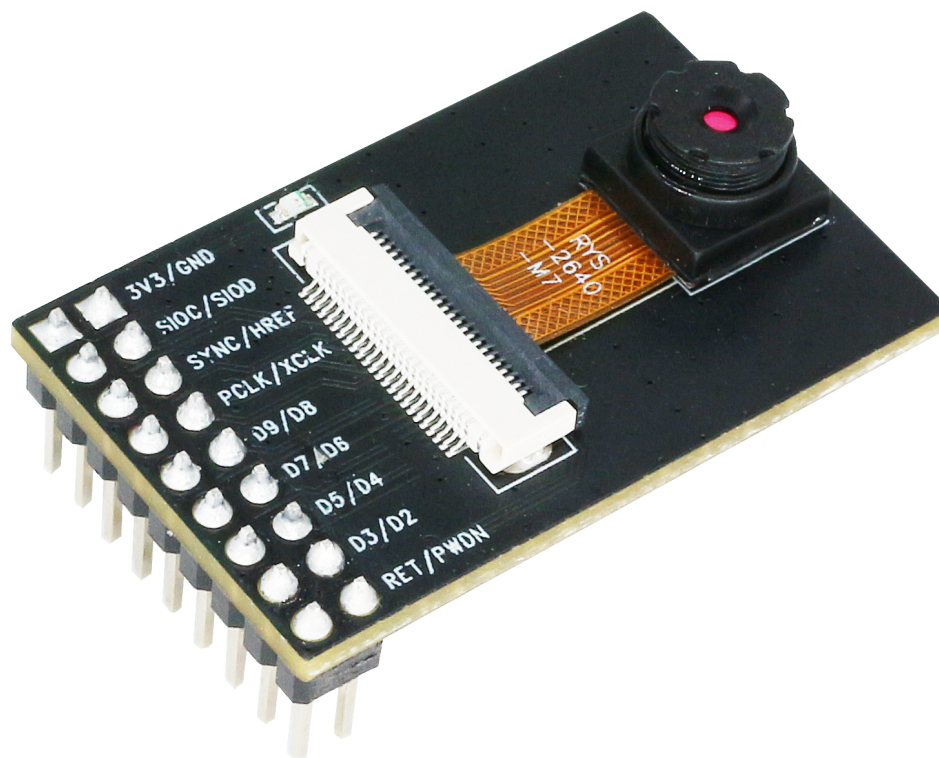


Fig. 25: ESP-LyraP-CAM

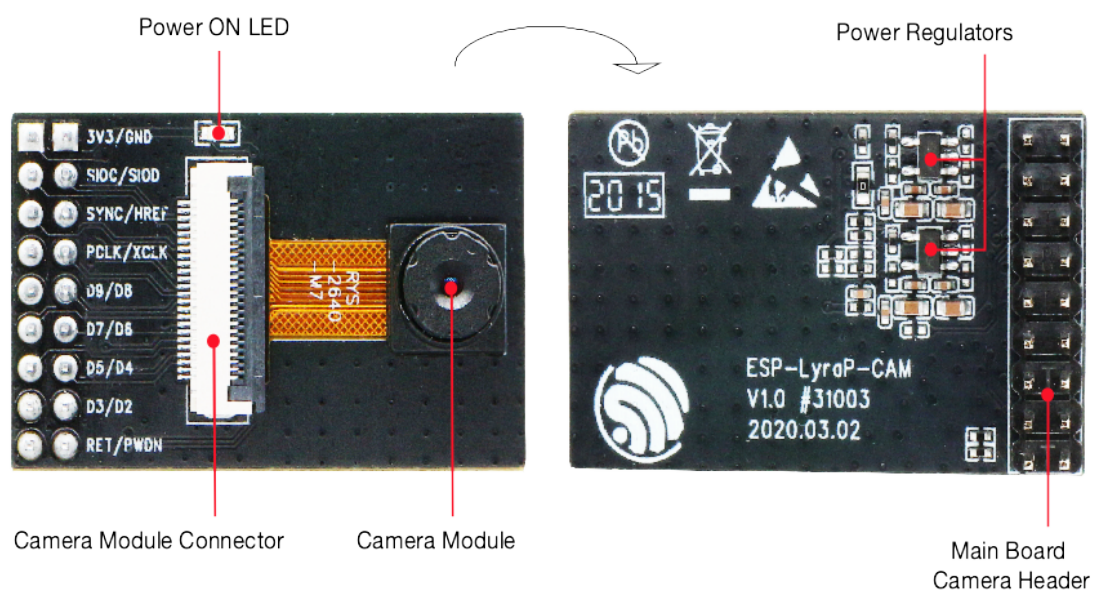


Fig. 26: ESP-LyraP-CAM - front and back

Description of Components

Key Component	Description
Main Board Camera Header	Mount onto main board's Camera Header
Power ON LED	Red LED is on if the power supply voltage is applied
Camera Module Connector	Supports OV2640 and OV3660 camera modules; this extension board is supplied with an OV2640 camera module
Power Regulators	LDO Regulators converting 3.3 V to 2.8 V and 1.5 V

Start Application Development Before powering up your ESP-LyraP-CAM, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Camera Header (e.g., ESP32-S2-Kaluga-1)
- ESP-LyraP-CAM extension board
- Computer running Windows, Linux, or macOS

Hardware Setup Insert the ESP-LyraP-CAM extension board into your board's female Camera Header.

Software Setup See Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP-LyraP-CAM and their interconnections.

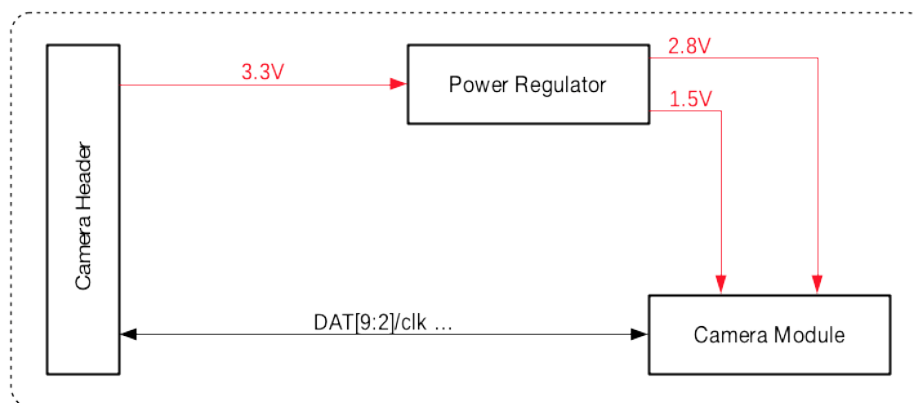


Fig. 27: ESP-LyraP-CAM block diagram

Hardware Revision Details No previous versions available.

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraP-CAM Schematic \(PDF\)](#)
- [ESP-LyraP-CAM PCB Layout \(PDF\)](#)
- [Camera OV2640 \(PDF\)](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraP-LCD32 v1.1

This user guide provides information on the ESP-LyraP-LCD32 extension board.

This extension board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraP-LCD32 v1.1 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.2](#).

The ESP-LyraP-LCD32 extends the functionality of your main board by adding an LCD graphic display.

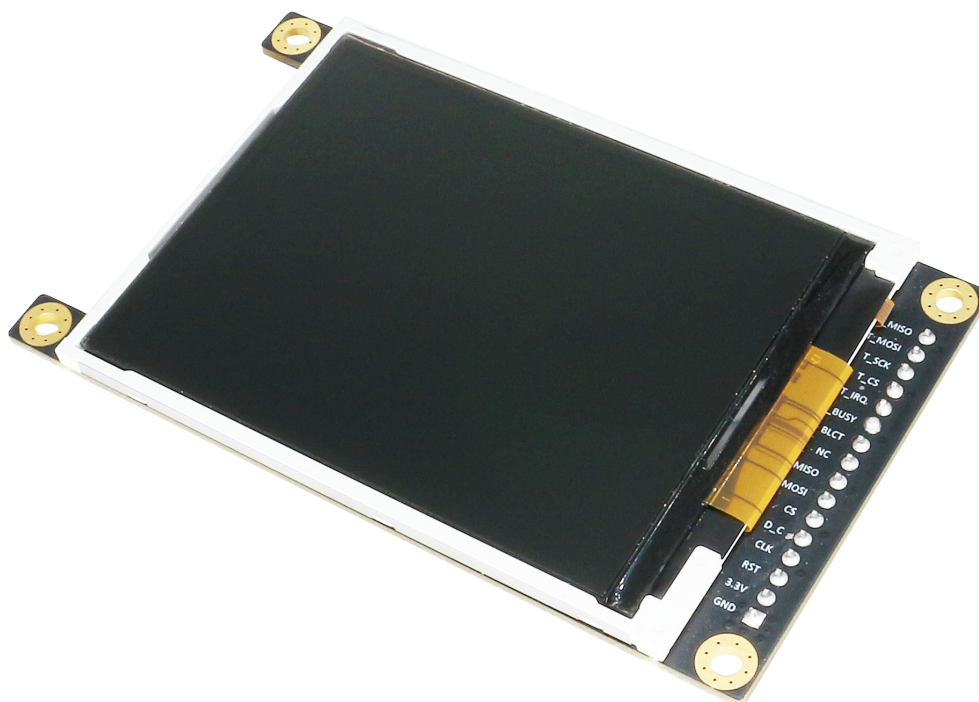


Fig. 28: ESP-LyraP-LCD32 (click to enlarge)

The document consists of the following major sections:

- [Overview](#): Provides an overview and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP-LyraP-LCD32's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraP-LCD32.
- [Related Documents](#): Gives links to related documentation.

Overview This extension board adds a 3.2" LCD graphic display with the resolution of 320x240. This display is connected to ESP32-S2 over the SPI bus.

Description of Components In the description of components below, **Reserved** means that the functionality is available, but the current version of the kit does not use it.

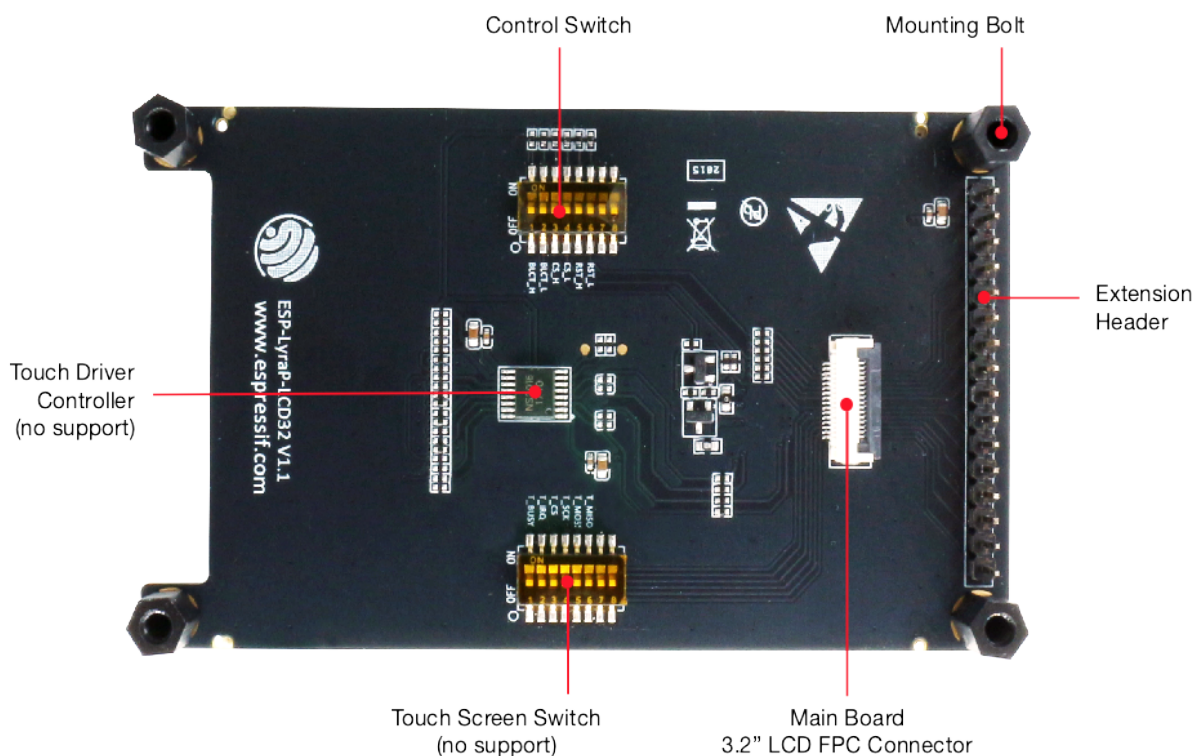


Fig. 29: ESP-LyraP-LCD32 - front (click to enlarge)

Key Component	Description
Extension Header	Male Extension Header for mounting onto a female Extension Header
LCD display	This version has a 3.2" 320x240 SPI LCD display module; the display driver/controller is Sitronix ST7789V
Touch Screen Switch	No support for touch screens, keep the switches to OFF to make the pins available for other uses
Main Board 3.2" LCD FPC Connector	(Reserved) Connect to main board's 3.2" LCD FPC connector
Control Switch	Switch to ON to set Reset/Backlight_control/CS to default high or low; switch to OFF to make the pins available for other uses

Start Application Development Before powering up your ESP-LyraP-LCD32, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Extension Header (e.g., ESP32-S2-Kaluga-1, ESP-LyraT-8311A)
- ESP-LyraP-LCD32 extension board
- Four mounting bolts (for stable mounting)
- Computer running Windows, Linux, or macOS

Hardware Setup To mount your ESP-LyraP-LCD32 onto the board with a female Extension Header:

1. Install the four mounting bolts onto the board with a female Extension Header
2. Align the ESP-LyraP-LCD32 with the bolts and Extension Header and insert it carefully

Software Setup See Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP-LyraP-LCD32 and their interconnections.

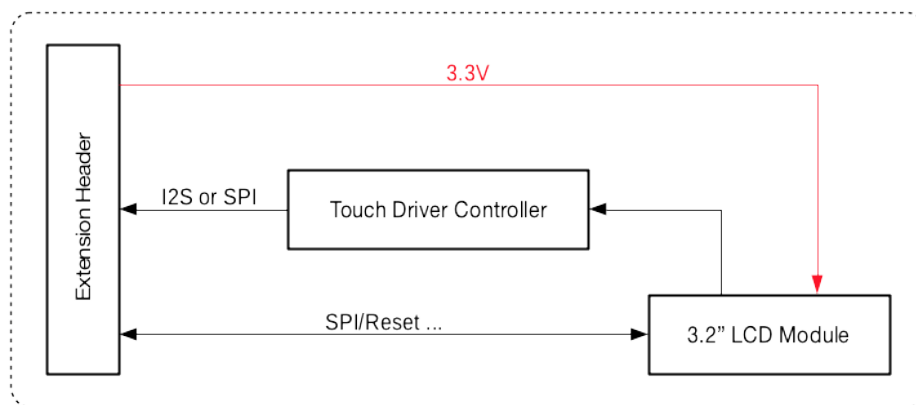


Fig. 30: ESP-LyraP-LCD32 block diagram

Hardware Revision Details No previous versions available.

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraP-LCD32 Schematic \(PDF\)](#)
- [ESP-LyraP-LCD32 PCB Layout \(PDF\)](#)
- [LCD ST7789 \(PDF\)](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraP-TouchA v1.1

This user guide provides information on the ESP-LyraP-TouchA extension board.

This board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraP-TouchA v1.1 is sold as part of the following kits:

- [ESP32-S2-Kaluga-1 Kit v1.3](#)
- [ESP32-S2-Kaluga-1 Kit v1.2](#)

The ESP-LyraP-TouchA extends the functionality of your main board by adding touch buttons.

The document consists of the following major sections:

- [Overview](#): Provides an overview and hardware setup instructions.
- [Hardware reference](#): Provides more detailed information about the ESP-LyraP-TouchA's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraP-TouchA.
- [Related Documents](#): Gives links to related documentation.

Overview The ESP-LyraP-TouchA has six touch buttons and is mainly designed for audio applications. However, the touch buttons can also be used for any other purposes.



Fig. 31: ESP-LyraP-TouchA

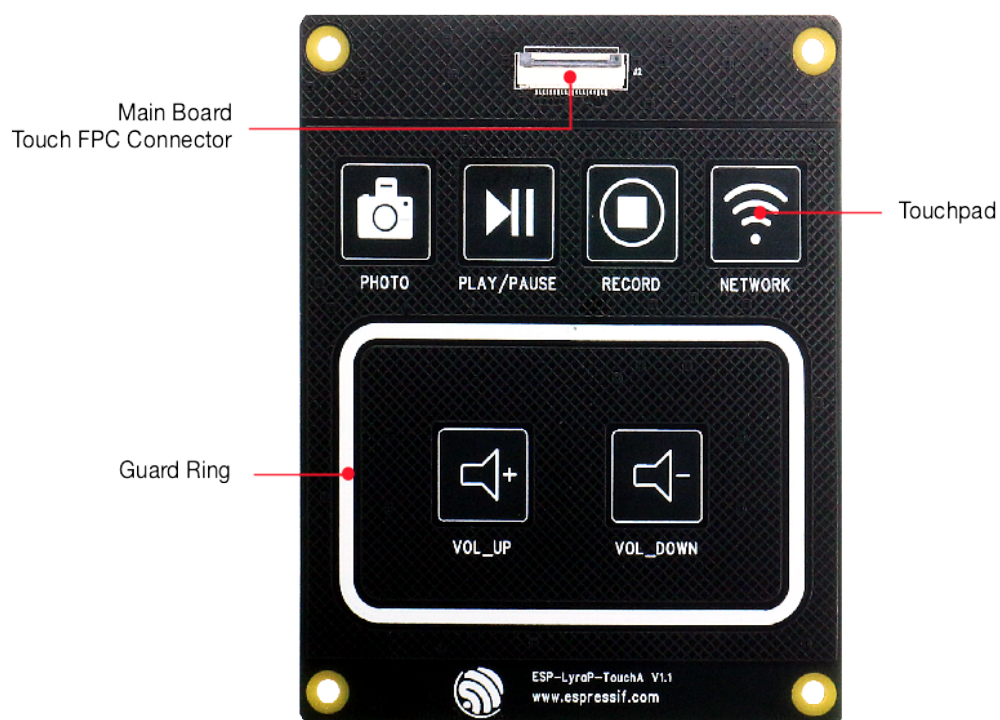


Fig. 32: ESP-LyraP-TouchA

Description of Components

Key Component	Description
Main Board Touch FPC Connector	Connect to main board's Touch FPC Connector.
Touchpad	Capacitive touch electrode.
Guard Ring	Connected to a touch sensor, the guard ring triggers an interrupt if wet (Water rejection). It indicates that the sensor array is also wet and most (if not all) touchpads are unusable due to the false detection of touches. After receiving this interrupt, the user might consider disabling all the touch sensors via software.

Start Application Development Before powering up your ESP-LyraP-TouchA, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a Touch FPC connector (e.g., ESP32-S2-Kaluga-1)
- ESP-LyraP-TouchA extension board
- FPC cable
- Computer running Windows, Linux, or macOS

Hardware Setup Connect the two FPC connectors with the FPC cable.

Software Setup See Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide.

Hardware Reference

Block Diagram A block diagram below shows the components of ESP-LyraP-TouchA and their interconnections.

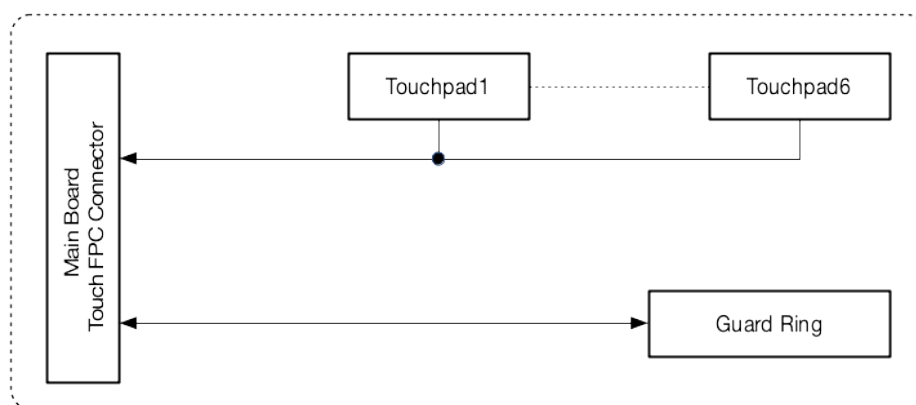


Fig. 33: ESP-LyraP-TouchA-v1.1 block diagram

Hardware Revision Details No previous versions available.

Related Documents

- [ESP-LyraP-TouchA Schematic \(PDF\)](#)
- [ESP-LyraP-TouchA PCB Layout \(PDF\)](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraT-8311A v1.2

This user guide provides information on the ESP-LyraT-8311A extension board.

This board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraT-8311A v1.2 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.2](#).

The ESP-LyraT-8311A extends the functionality of your main board by adding sound processing functionality:

- Audio playback/recording
- Processing of audio signals
- Programmable buttons for easy control

This extension board can be used in many ways. The applications might include voice user interface, voice control, voice authorization, recording and playback of sound, etc.

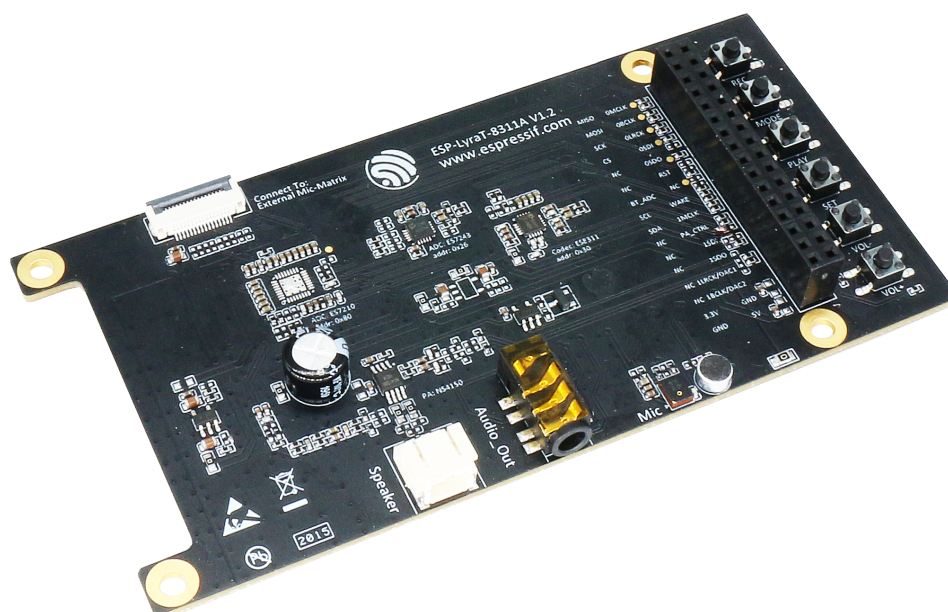


Fig. 34: ESP-LyraT-8311A (click to enlarge)

The document consists of the following major sections:

- [Overview](#): Provides an overview and hardware setup instructions.
- [Hardware reference](#): Provides more detailed information about the ESP-LyraT-8311A's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraT-8311A.
- [Related Documents](#): Gives links to related documentation.

Overview The ESP-LyraT-8311A is mainly designed for audio applications. However, you can use your creativity to come up with any other use cases.

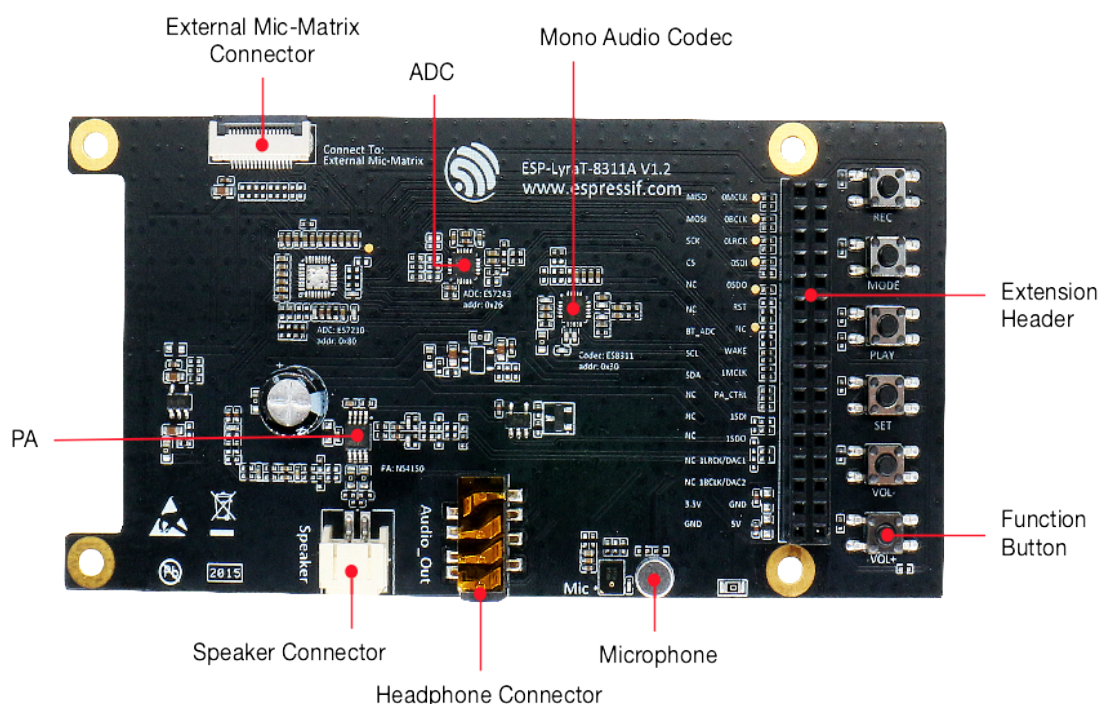


Fig. 35: ESP-LyraT-8311A - front (click to enlarge)

Description of Components The description of components starts from the top right corner and then goes clockwise.

Reserved means that the functionality is available, but the current version of the kit does not use it.

Key Component	Description
Extension Header	Male Extension Header on the flip side is for mounting onto main board's Extension Header; Female Extension Header is for mounting other boards that have a Male Extension Header
Function Button	This board has six programmable buttons
Microphone	Supports Electret and MEMS microphones; this extension board is supplied with an electret microphone
Headphone Connector	6.3 mm (1/8") stereo headphone connector
Speaker Connector	Connect an external speaker to the 2-pin connector
PA	3 W Audio signal amplifier for the external speaker
External Mic-Matrix Connector	(Reserved) FPC connector for external Mic-Matrix (microphone boards)
ADC	(Reserved) high-performance ADC/ES7243: 1 channel for microphone, 1 channel for acoustic echo cancellation (AEC) function
Mono Audio Codec	ES8311 audio ADC and DAC; it can convert the analog signal picked up by the microphone or convert digital signal to play it back through a speaker or headphones

Start Application Development Before powering up your ESP-LyraT-8311A, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Extension Header (e.g., ESP32-S2-Kaluga-1)
- ESP-LyraT-8311A extension board
- Four mounting bolts (for stable mounting)
- Computer running Windows, Linux, or macOS

Hardware Setup To mount your ESP-LyraT-8311A onto the board with a female Extension Header:

1. Install the four mounting bolts onto the board with a female Extension Header
2. Align the ESP-LyraT-8311A with the bolts and Extension Header and insert it carefully

Software Setup Depending on your application, see:

- [ESP-ADF Getting Started Guide](#) if you develop with ESP-ADF (Espressif Audio Development Framework).
- Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide if you develop directly with ESP-IDF (Espressif IOT Development Framework).

Hardware Reference

Block Diagram A block diagram below shows the components of ESP-LyraT-8311A and their interconnections.

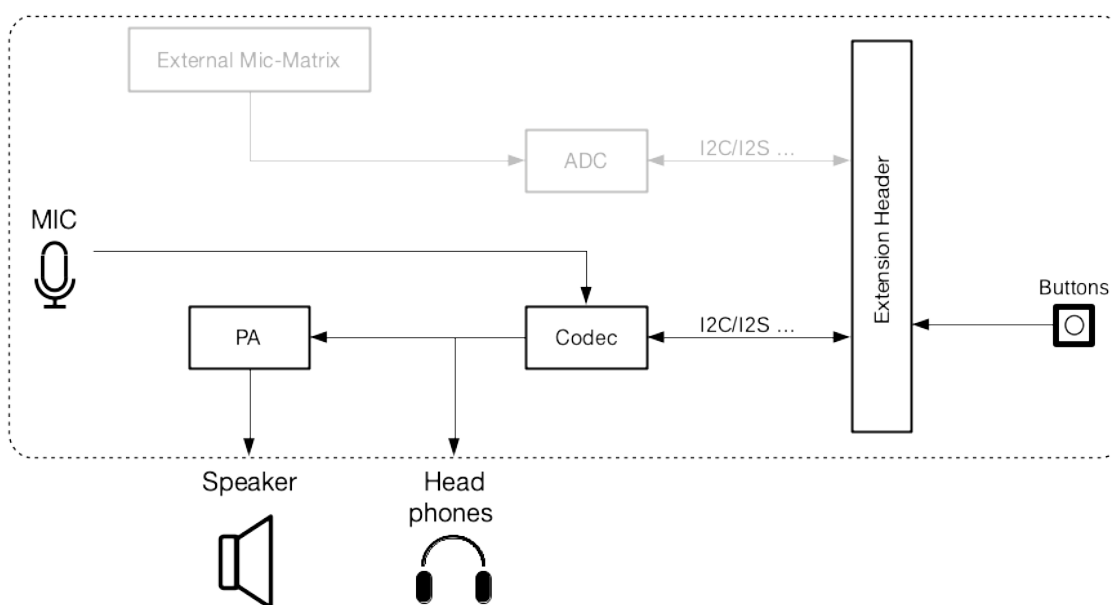


Fig. 36: ESP-LyraT-8311A block diagram

Hardware Revision Details No previous versions available.

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraT-8311A Schematic](#) (PDF)
- [ESP-LyraT-8311A PCB Layout](#) (PDF)
- [Audio ES8311](#) (PDF)

For other design documentation for the board, please contact us at sales@espressif.com.

- [ESP32-S2 Datasheet](#) (PDF)
- [ESP32-S2-WROVER Datasheet](#) (PDF)

- [ESP Product Selector](#)
- [JTAG Debugging](#)
- [ESP32-S2-Kaluga-1 Schematic \(PDF\)](#)
- [ESP32-S2-Kaluga-1 PCB Layout \(PDF\)](#)
- [ESP32-S2-Kaluga-1 Pin Mapping \(Excel\)](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraP-CAM v1.1

This user guide provides information on the ESP-LyraP-CAM extension board.

This extension board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraP-CAM v1.1 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.3](#).

The ESP-LyraP-CAM extends the functionality of your main board by adding a camera.

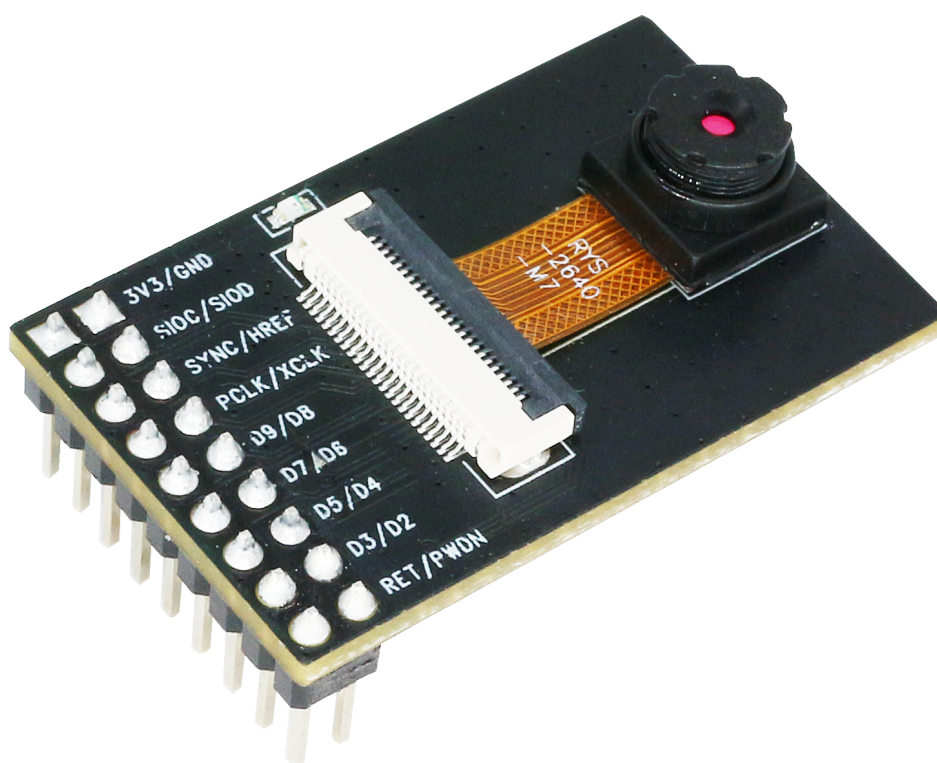


Fig. 37: ESP-LyraP-CAM

The document consists of the following major sections:

- [Overview](#): Provides an overview and hardware/software setup instructions to get started.
- [Hardware reference](#): Provides more detailed information about the ESP-LyraP-CAM's hardware.
- [Hardware Revision Details](#): Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraP-CAM.
- [Related Documents](#): Gives links to related documentation.

Overview This extension board adds a camera to your main board.

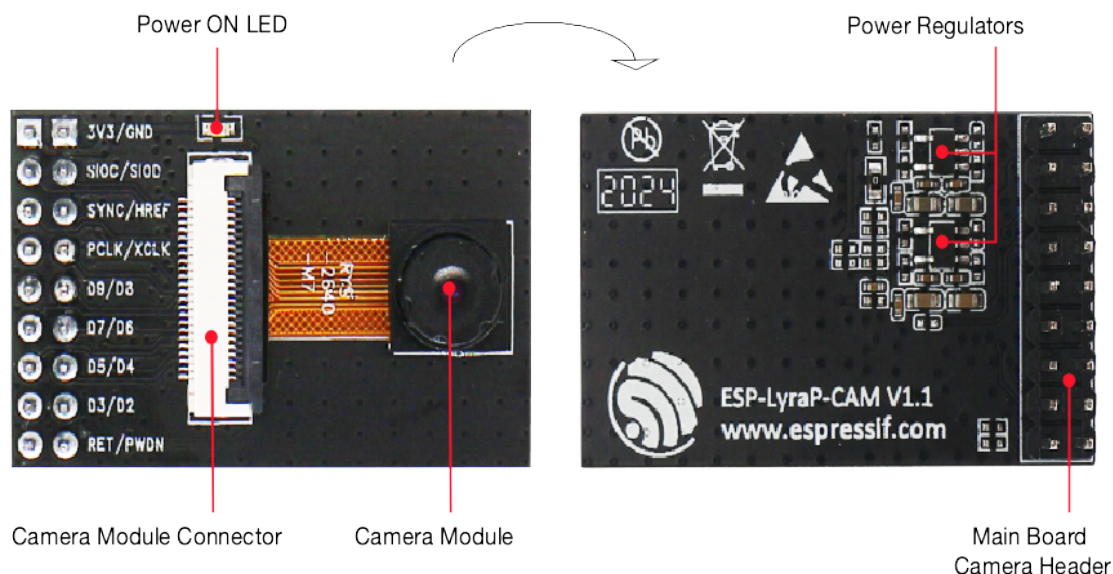


Fig. 38: ESP-LyraP-CAM - front and back

Description of Components

Key Component	Description
Main Board Camera Header	Mount onto main board's Camera Header
Power ON LED	Red LED is on if the power supply voltage is correct
Camera Module Connector	Supports OV2640 and OV3660 camera modules; this extension board is supplied with an OV2640 camera module
Power Regulators	LDO Regulators converting 3.3 V to 2.8 V and 1.5 V

Start Application Development Before powering up your ESP-LyraP-CAM, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Camera Header (e.g., ESP32-S2-Kaluga-1)
- ESP-LyraP-CAM extension board
- Computer running Windows, Linux, or macOS

Hardware Setup Insert the ESP-LyraP-CAM extension board into your board's female Camera Header.

Software Setup See Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP-LyraP-CAM and their interconnections.

Hardware Revision Details

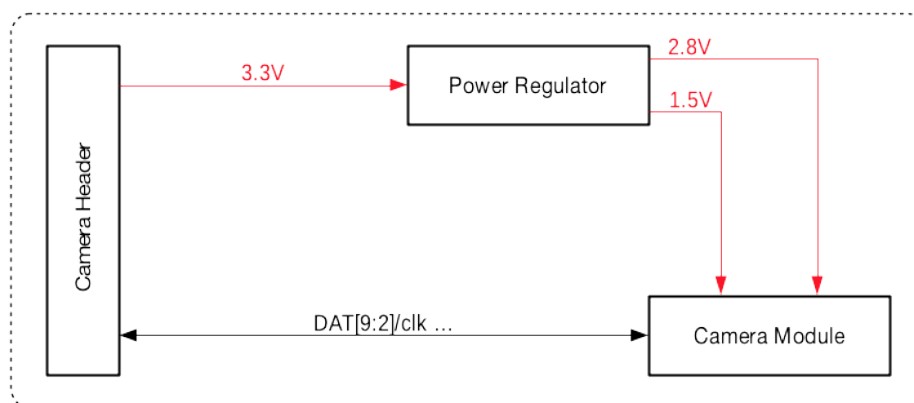


Fig. 39: ESP-LyraP-CAM block diagram

ESP-LyraP-CAM v1.1

- Silk screen updated
- No actual hardware updates

ESP-LyraP-CAM v1.0 *Initial release*

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraP-CAM Schematic](#) (PDF)
- [ESP-LyraP-CAM PCB Layout](#) (PDF)
- [Camera OV2640](#) (PDF)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraP-LCD32 v1.2

This user guide provides information on the ESP-LyraP-LCD32 extension board.

This extension board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraP-LCD32 v1.2 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.3](#).

The ESP-LyraP-LCD32 extends the functionality of your main board by adding an LCD graphic display.

The document consists of the following major sections:

- **Overview:** Provides an overview and hardware/software setup instructions to get started.
- **Hardware reference:** Provides more detailed information about the ESP-LyraP-LCD32's hardware.
- **Hardware Revision Details:** Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraP-LCD32.
- **Related Documents:** Gives links to related documentation.

Overview This extension board adds a 3.2" LCD graphic display with the resolution of 320x240. This display is connected to ESP32-S2 over the SPI bus.

Description of Components In the description of components below, **Reserved** means that the functionality is available, but the current version of the kit does not use it.

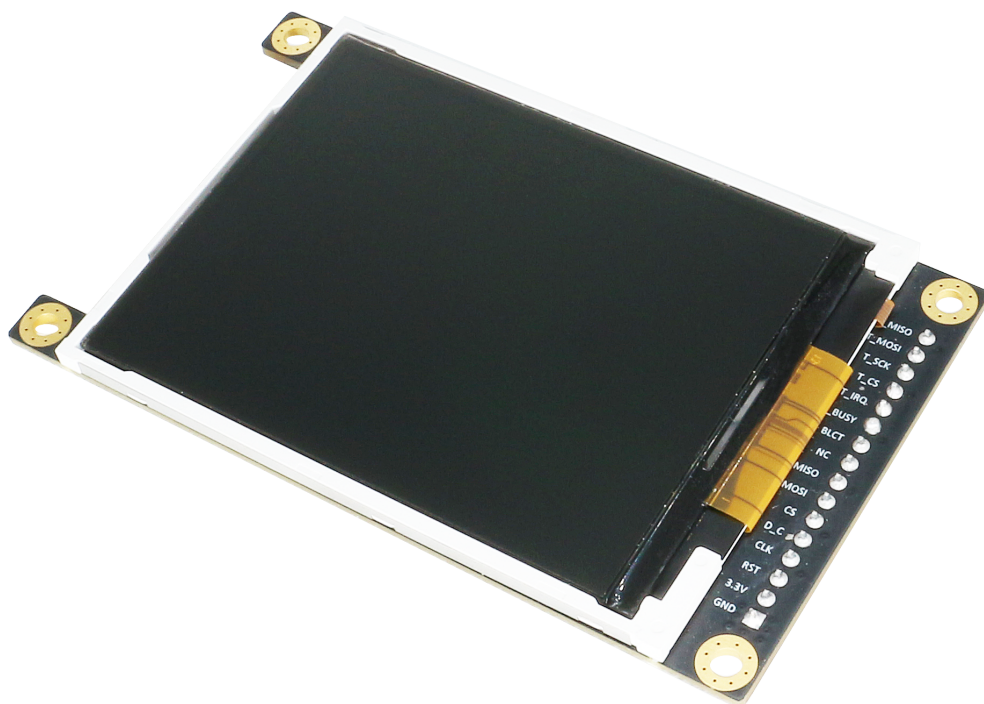


Fig. 40: ESP-LyraP-LCD32 (click to enlarge)



Fig. 41: ESP-LyraP-LCD32 - front (click to enlarge)

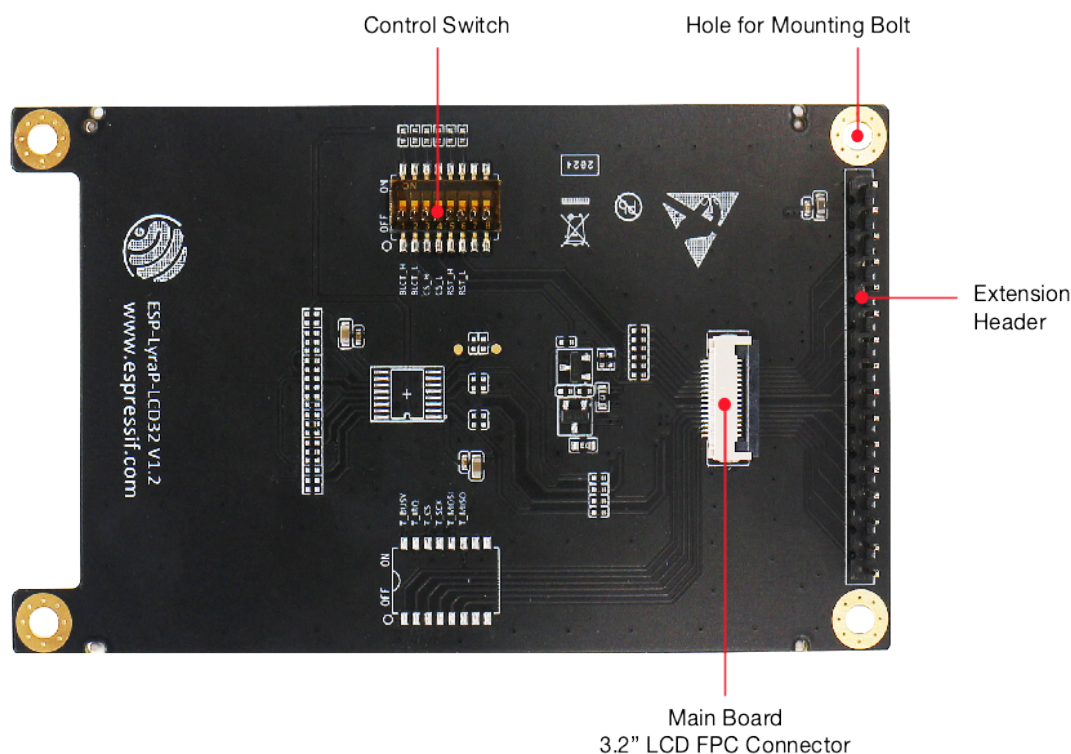


Fig. 42: ESP-LyraP-LCD32 - back (click to enlarge)

Key Component	Description
Extension Header	Male Extension Header for mounting onto a female Extension Header
LCD Display	This version has a 3.2" 320x240 SPI LCD display module; the display driver/controller is either Sitronix ST7789V or Ilitek ILI9341
Touch Screen Switch	No support for touch screens, keep the switches to OFF to make the pins available for other uses
Main Board 3.2" LCD FPC Connector	(Reserved) Connect to main board's 3.2" LCD FPC connector
Control Switch	Switch to ON to set Reset/Backlight_control/CS to default high or low; switch to OFF to make the pins available for other uses

Start Application Development Before powering up your ESP-LyraP-LCD32, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Extension Header (e.g., ESP32-S2-Kaluga-1, ESP-LyraT-8311A)
- ESP-LyraP-LCD32 extension board
- Four mounting bolts (for stable mounting)
- Computer running Windows, Linux, or macOS

Hardware Setup To mount your ESP-LyraP-LCD32 onto the board with a female Extension Header:

1. Install the four mounting bolts onto the board with a female Extension Header
2. Align the ESP-LyraP-LCD32 with the bolts and Extension Header and insert it carefully

Software Setup See Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide.

Hardware Reference

Block Diagram A block diagram below shows the components of the ESP-LyraP-LCD32 and their interconnections.

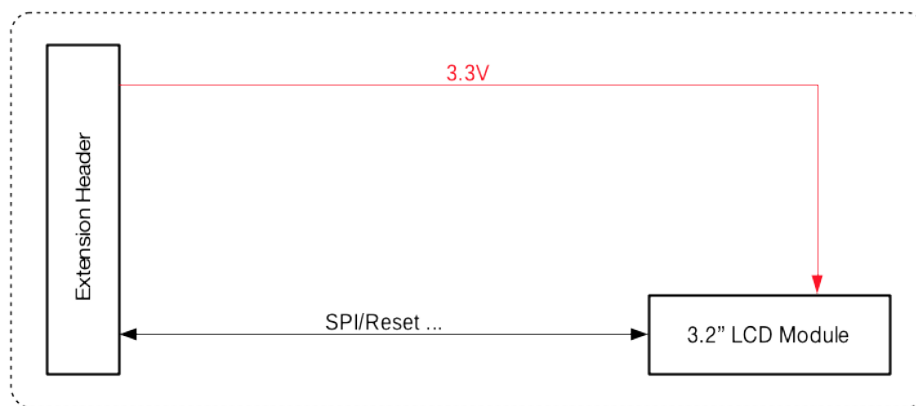


Fig. 43: ESP-LyraP-LCD32 block diagram

Hardware Revision Details

ESP-LyraP-LCD32 v1.2

- LCD backlight default ON, cannot be controlled by MCU
- Touch Driver and related switch removed for major limitations caused by multiplexed pins

ESP-LyraP-LCD32 v1.1 *Initial release*

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraP-LCD32 Schematic \(PDF\)](#)
- [ESP-LyraP-LCD32 PCB Layout \(PDF\)](#)
- [LCD ST7789 \(PDF\)](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP-LyraT-8311A v1.3

This user guide provides information on the ESP-LyraT-8311A extension board.

This board cannot be bought separately and is usually sold together with other Espressif development boards (e.g., ESP32-S2-Kaluga-1), which will be referred to as *main boards* below.

Currently, ESP-LyraT-8311A v1.3 is sold as part of the [ESP32-S2-Kaluga-1 Kit v1.3](#).

The ESP-LyraT-8311A extends the functionality of your main board by adding sound processing functionality:

- Audio playback/recording
- Processing of audio signals
- Programmable buttons for easy control

This extension board can be used in many ways. The applications might include voice user interface, voice control, voice authorization, recording and playback of sound, etc.

The document consists of the following major sections:

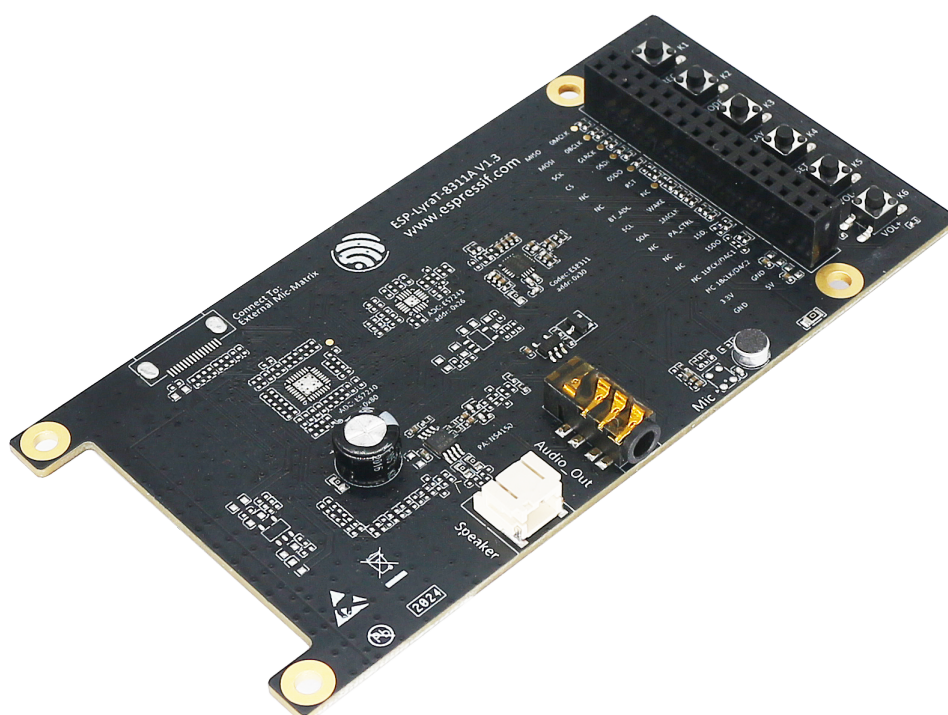


Fig. 44: ESP-LyraT-8311A (click to enlarge)

- **Overview:** Provides an overview and hardware setup instructions.
- **Hardware reference:** Provides more detailed information about the ESP-LyraT-8311A's hardware.
- **Hardware Revision Details:** Covers revision history, known issues, and links to user guides for previous versions of the ESP-LyraT-8311A.
- **Related Documents:** Gives links to related documentation.

Overview The ESP-LyraT-8311A is mainly designed for audio applications. However, you can use your creativity to come up with any other use cases.

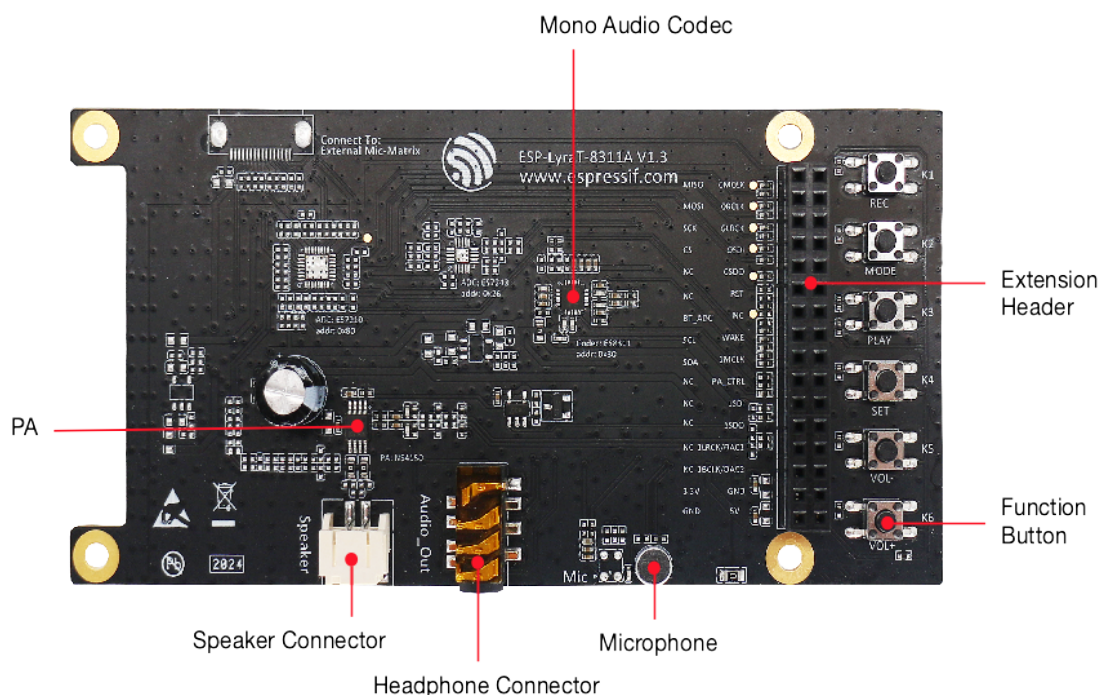


Fig. 45: ESP-LyraT-8311A - front (click to enlarge)

Description of Components The description of components starts from the top right corner and then goes clockwise.

Reserved means that the functionality is available, but the current version of the kit does not use it.

Key Component	Description
Extension Header	Male Extension Header on the flip side is for mounting onto main board's Extension Header; Female Extension Header is for mounting other boards that have a Male Extension Header
Function Button	This board has six programmable buttons
Microphone	Supports Electret and MEMS microphones; this extension board is supplied with an electret microphone
Headphone Connector	6.3 mm (1/8") stereo headphone connector
Speaker Connector	Connect an external speaker to the 2-pin connector
PA	3 W Audio signal amplifier for the external speaker
External Mic-Matrix Connector	(Reserved) FPC connector for external Mic-Matrix (microphone boards)
ADC	(Reserved) high-performance ADC/ES7243: 1 channel for microphone, 1 channel for acoustic echo cancellation (AEC) function
Mono Audio Codec	ES8311 audio ADC and DAC; it can convert the analog signal picked up by the microphone or convert digital signal to play it back through a speaker or headphones

Start Application Development Before powering up your ESP-LyraT-8311A, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- Board with a female Extension Header (e.g., ESP32-S2-Kaluga-1)
- ESP-LyraT-8311A extension board
- Four mounting bolts (for stable mounting)
- Computer running Windows, Linux, or macOS

Hardware Setup To mount your ESP-LyraT-8311A onto the board with a female Extension Header:

1. Install the four mounting bolts onto the board with a female Extension Header
2. Align the ESP-LyraT-8311A with the bolts and Extension Header and insert it carefully

Software Setup Depending on your application, see:

- [ESP-ADF Getting Started Guide](#) if you develop with ESP-ADF (Espressif Audio Development Framework).
- Section [Software Setup](#) of the ESP32-S2-Kaluga-1 kit user guide if you develop directly with ESP-IDF (Espressif IOT Development Framework).

Hardware Reference

Block Diagram A block diagram below shows the components of ESP-LyraT-8311A and their interconnections.

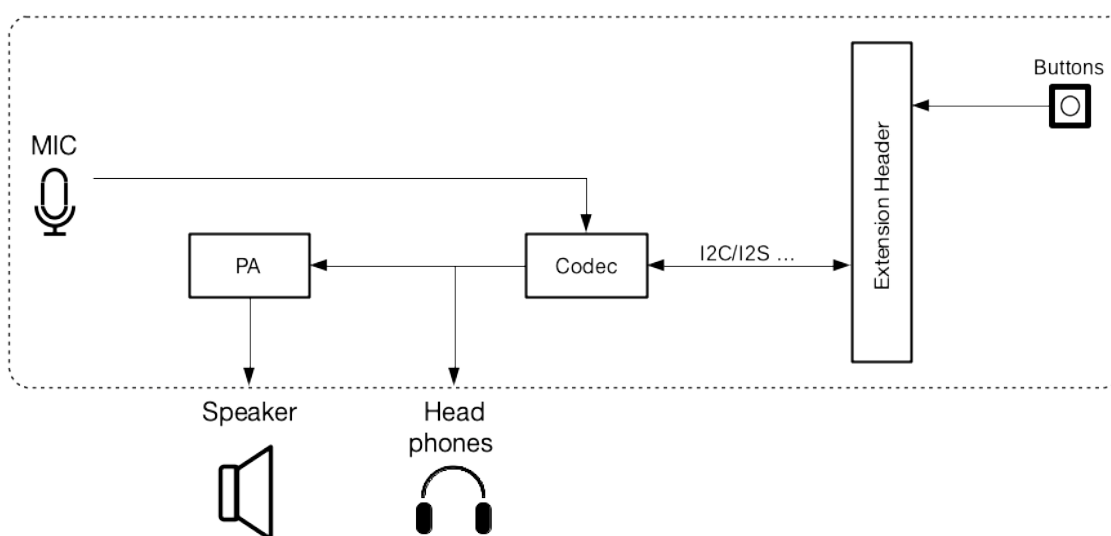


Fig. 46: ESP-LyraT-8311A block diagram

Hardware Revision Details

ESP-LyraT-8311A v1.3

- ADC/ES7243 and ADC/ES7210 removed as the Mono Audio Codec chip provides all the needed functionality.

ESP-LyraT-8311A v1.2 *Initial release*

Related Documents Please download the following documents from [the HTML version of esp-dev-kits Documentation](#).

- [ESP-LyraT-8311A Schematic](#) (PDF)
- [ESP-LyraT-8311A PCB Layout](#) (PDF)
- [Audio ES8311](#) (PDF)

For other design documentation for the board, please contact us at sales@espressif.com.

- [ESP32-S2 Datasheet](#) (PDF)
- [ESP32-S2-WROVER Datasheet](#) (PDF)
- [ESP Product Selector](#)
- [JTAG Debugging](#)
- [ESP32-S2-Kaluga-1 Schematic](#) (PDF)
- [ESP32-S2-Kaluga-1 PCB Layout](#) (PDF)
- [ESP32-S2-Kaluga-1 Pin Mapping](#) (Excel)

For other design documentation for the board, please contact us at sales@espressif.com.

3.4 ESP32-S2-Saola-1

ESP32-S2-Saola-1 is a small-sized ESP32-S2 based development board produced by Espressif. Most of the I/O pins are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-Saola-1 on a breadboard.

3.4.1 ESP32-S2-Saola-1

This user guide provides information on ESP32-S2-Saola-1, a small-sized [ESP32-S2 Datasheet](#) based development board produced by Espressif.

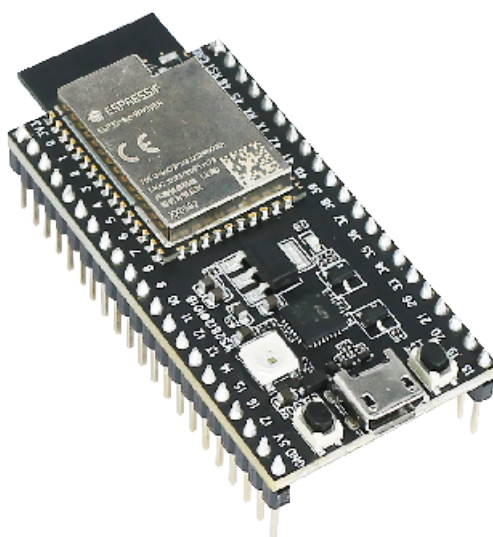


Fig. 47: ESP32-S2-Saola-1

The document consists of the following major sections:

- *Getting started*: Provides an overview of the ESP32-S2-Saola-1 and hardware/software setup instructions to get started.

- **Hardware reference:** Provides more detailed information about the ESP32-S2-Saola-1's hardware.
- **Hardware Revision Details:** Revision history, known issues, and links to user guides for previous versions (if any) of ESP32-S2-Saola-1.
- **Related Documents:** Gives links to related documentation.

Getting Started

This section describes how to get started with ESP32-S2-Saola-1. It begins with a few introductory sections about the ESP32-S2-Saola-1, then Section [Start Application Development](#) provides instructions on how to get the ESP32-S2-Saola-1 ready and flash firmware into it.

Overview ESP32-S2-Saola-1 is a small-sized ESP32-S2 based development board produced by Espressif. Most of the I/O pins are broken out to the pin headers on both sides for easy interfacing. Developers can either connect peripherals with jumper wires or mount ESP32-S2-Saola-1 on a breadboard.

To cover a wide range of users' needs, ESP32-S2-Saola-1 supports:

- [ESP32-S2-WROVER](#)
- [ESP32-S2-WROVER-I](#)
- [ESP32-S2-WROOM](#)
- [ESP32-S2-WROOM-I](#)

In this guide, we take ESP32-S2-Saola-1 equipped with ESP32-S2-WROVER as an example.

Contents and Packaging

Retail Orders If you order a few samples, each ESP32-S2-Saola-1 comes in an individual package in either anti-static bag or any packaging depending on your retailer.

For retail orders, please go to <https://www.espressif.com/en/contact-us/get-samples>.

Wholesale Orders If you order in bulk, the boards come in large cardboard boxes.

For wholesale orders, please go to <https://www.espressif.com/en/contact-us/sales-questions>.

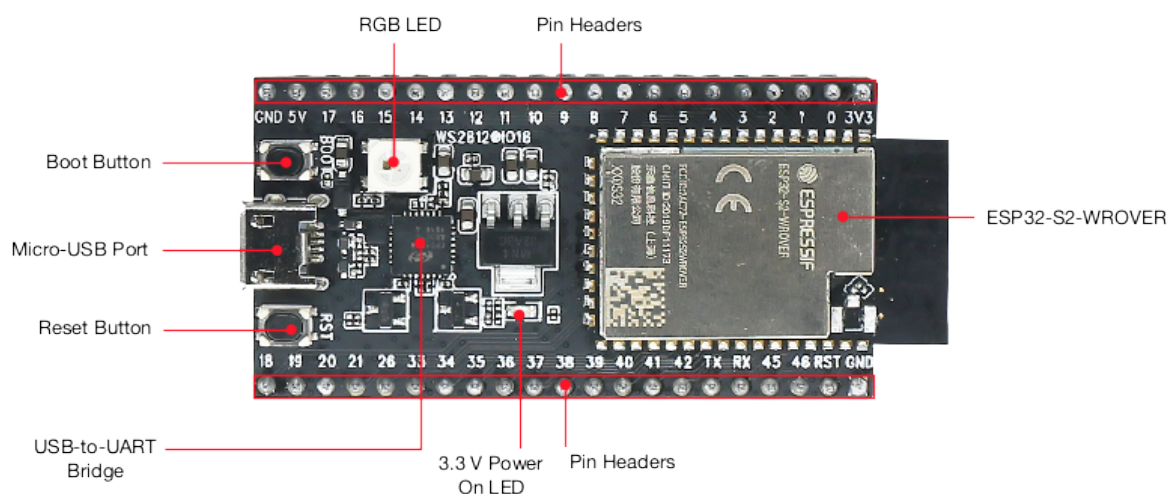


Fig. 48: ESP32-S2-Saola-1 - front

Description of Components The key components of the board are described in a clockwise direction.

Key Component	Description
ESP32-S2-WROVER	ESP32-S2-WROVER is a powerful, generic Wi-Fi MCU module that integrates ESP32-S2. It has a PCB antenna, a 4 MB external SPI flash and an additional 2 MB PSRAM.
Pin Headers	All available GPIO pins (except for the SPI bus for flash and PSRAM) are broken out to the pin headers on the board. Users can program ESP32-S2 chip to enable multiple functions such as SPI, I2S, UART, I2C, touch sensors, PWM etc.
3.3 V Power On LED	Turns on when the USB power is connected to the board.
USB-to-UART Bridge	Single USB-UART bridge chip provides transfer rates up to 3 Mbps.
Reset Button	Reset button.
Micro-USB Port	USB interface. Power supply for the board as well as the communication interface between a computer and the ESP32-S2 chip.
Boot Button	Download button. Holding down Boot and then pressing Reset initiates Firmware Download mode for downloading firmware through the serial port.
RGB LED	Addressable RGB LED (WS2812), driven by GPIO18.

Start Application Development Before powering up your ESP32-S2-Saola-1, please make sure that it is in good condition with no obvious signs of damage.

Required Hardware

- ESP32-S2-Saola-1
- USB 2.0 cable (Standard-A to Micro-B)
- Computer running Windows, Linux, or macOS

Note: Be sure to use an appropriate USB cable. Some cables are for charging only and do not provide the needed data lines nor work for programming the boards.

Software Setup Please proceed to [Get Started](#), where Section [Installation](#) will quickly help you set up the development environment and then flash an application example into your ESP32-S2-Saola-1.

Note: ESP32-S2 series of chips only supports ESP-IDF master or version v4.2 and higher.

Hardware Reference

Block Diagram A block diagram below shows the components of ESP32-S2-Saola-1 and their interconnections.

Power Supply Options There are three mutually exclusive ways to provide power to the board:

- Micro-USB port, default power supply
- 5V and GND pin headers
- 3V3 and GND pin headers

It is recommended to use the first option: Micro-USB Port.

Header Block The two tables below provide the **Name** and **Function** of the pin headers on both sides of the board (J2 and J3). The pin header names are shown in [ESP32-S2-Saola-1 - front](#). The numbering is the same as in the [ESP32-S2-Saola-1 Schematics](#) (PDF).

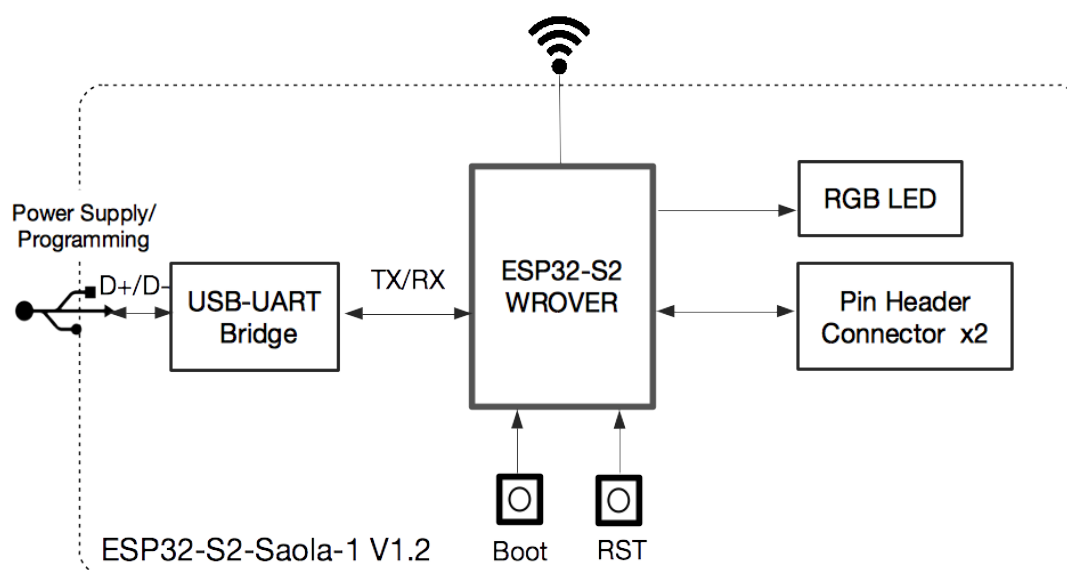


Fig. 49: ESP32-S2-Saola-1 (click to enlarge)

J2

No.	Name	Type ^{Page 75, 1}	Function
1	3V3	P	3.3 V power supply
2	IO0	I/O	GPIO0, Boot
3	IO1	I/O	GPIO1, ADC1_CH0, TOUCH_CH1
4	IO2	I/O	GPIO2, ADC1_CH1, TOUCH_CH2
5	IO3	I/O	GPIO3, ADC1_CH2, TOUCH_CH3
6	IO4	I/O	GPIO4, ADC1_CH3, TOUCH_CH4
7	IO5	I/O	GPIO5, ADC1_CH4, TOUCH_CH5
8	IO6	I/O	GPIO6, ADC1_CH5, TOUCH_CH6
9	IO7	I/O	GPIO7, ADC1_CH6, TOUCH_CH7
10	IO8	I/O	GPIO8, ADC1_CH7, TOUCH_CH8
11	IO9	I/O	GPIO9, ADC1_CH8, TOUCH_CH9
12	IO10	I/O	GPIO10, ADC1_CH9, TOUCH_CH10
13	IO11	I/O	GPIO11, ADC2_CH0, TOUCH_CH11
14	IO12	I/O	GPIO12, ADC2_CH1, TOUCH_CH12
15	IO13	I/O	GPIO13, ADC2_CH2, TOUCH_CH13
16	IO14	I/O	GPIO14, ADC2_CH3, TOUCH_CH14
17	IO15	I/O	GPIO15, ADC2_CH4, XTAL_32K_P
18	IO16	I/O	GPIO16, ADC2_CH5, XTAL_32K_N
19	IO17	I/O	GPIO17, ADC2_CH6, DAC_1
20	5V0	P	5 V power supply
21	GND	G	Ground

¹ P: Power supply; I: Input; O: Output; T: High impedance.

J3

No.	Name	Type	Function
1	GND	G	Ground
2	RST	I	CHIP_PU, Reset
3	IO46	I	GPIO46
4	IO45	I/O	GPIO45
5	IO44	I/O	GPIO44, U0RXD
6	IO43	I/O	GPIO43, U0TXD
7	IO42	I/O	GPIO42, MTMS
8	IO41	I/O	GPIO41, MTDI
9	IO40	I/O	GPIO40, MTDO
10	IO39	I/O	GPIO39, MTCK
11	IO38	I/O	GPIO38
12	IO37	I/O	GPIO37
13	IO36	I/O	GPIO36
14	IO35	I/O	GPIO35
16	IO34	I/O	GPIO34
17	IO33	I/O	GPIO33
17	IO26	I/O	GPIO26
18	IO21	I/O	GPIO21
19	IO20	I/O	GPIO20, ADC2_CH9, USB_D+
20	IO19	I/O	GPIO19, ADC2_CH8, USB_D-
21	IO18	I/O	GPIO18, ADC2_CH7, DAC_2, RGB LED

ESP32-S2-Saola-1

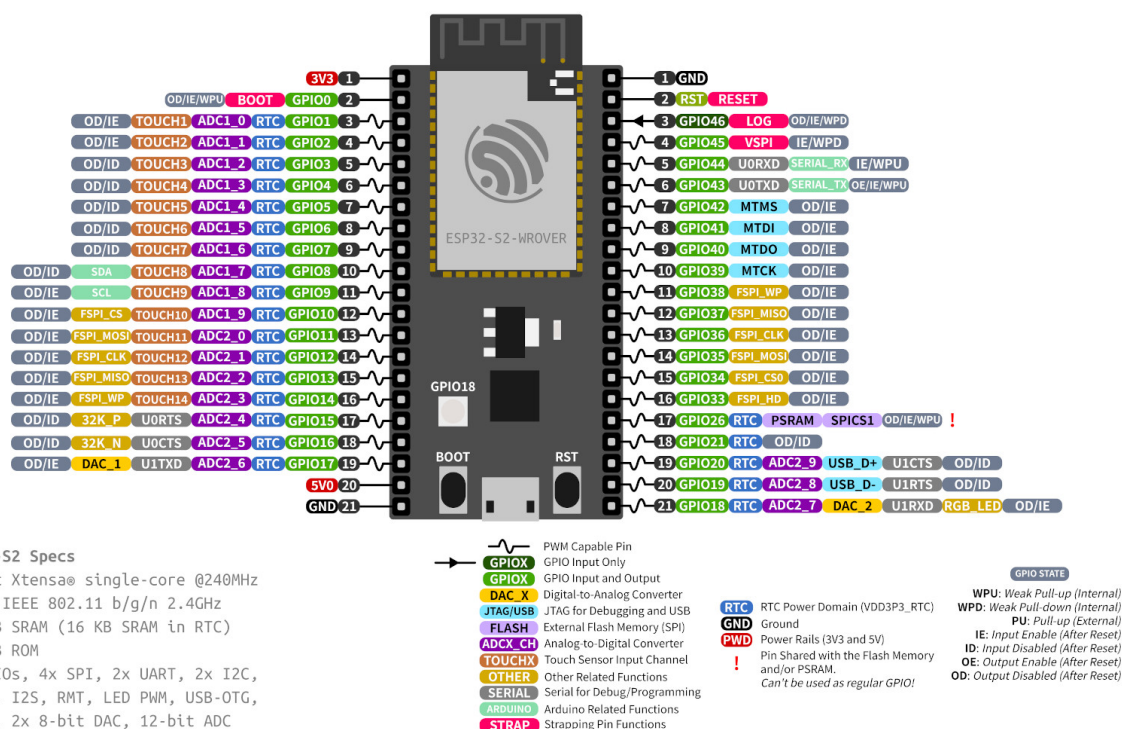


Fig. 50: ESP32-S2 Saola-1 Pin Layout (click to enlarge)

Pin Layout

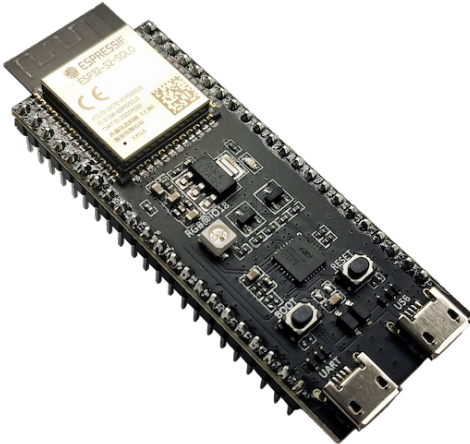
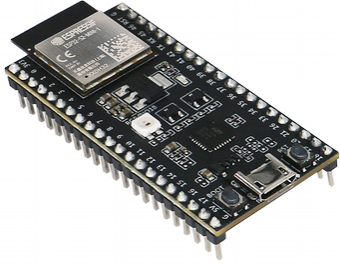
Hardware Revision Details

This is the first revision of this board released.

Related Documents

- [ESP32-S2-Saola-1 Schematics \(PDF\)](#)
- [ESP32-S2-Saola-1 Dimensions \(PDF\)](#)
- [ESP32-S2 Datasheet \(PDF\)](#)
- [ESP32-S2-WROVER and ESP32-S2-WROVER-I Datasheet \(PDF\)](#)
- [ESP32-S2-WROOM and ESP32-S2-WROOM-I Datasheet \(PDF\)](#)
- [ESP Product Selector](#)

For other design documentation for the board, please contact us at sales@espressif.com.

ESP32-S2 Development Boards	
	
ESP32-S2-DevKitC-1	ESP32-S2-DevKitM-1

Chapter 4

Related Documentation and Resources

4.1 Related Documentation

- [ESP32-S2 Datasheet](#) –Specifications of the ESP32-S2 hardware.
- [ESP32-S2 Technical Reference Manual](#) –Detailed information on how to use the ESP32-S2 memory and peripherals.
- [ESP32-S2 Hardware Design Guidelines](#) –Guidelines on how to integrate the ESP32-S2 into your hardware product.
- ESP32-S2 Product/Process Change Notifications (PCN)
<https://espressif.com/en/support/documents/pcns?keys=ESP32-S2>
- ESP32-S2 Advisories –Information on security, bugs, compatibility, component reliability.
<https://espressif.com/en/support/documents/advisories?keys=ESP32-S2>
- Certificates
<https://espressif.com/en/support/documents/certificates>
- Documentation Updates and Update Notification Subscription
<https://espressif.com/en/support/download/documents>

4.2 Developer Zone

- [ESP-IDF Programming Guide for ESP32-S2](#) –Extensive documentation for the ESP-IDF development framework.
- [ESP-IoT-Solution Programming Guide](#) - Extensive documentation for the ESP-IoT-Solution development framework.
- [ESP-FAQ](#) - A summary document of frequently asked questions released by Espressif.
- 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>

4.3 Products

- ESP32-S2 Series SoCs –Browse through all ESP32-S2 SoCs.
<https://espressif.com/en/products/socs?id=ESP32-S2>
- ESP32-S2 Series Modules –Browse through all ESP32-S2-based modules.
<https://espressif.com/en/products/modules?id=ESP32-S2>
- ESP32-S2 Series DevKits –Browse through all ESP32-S2-based devkits.
<https://espressif.com/en/products/devkits?id=ESP32-S2>
- ESP Product Selector –Find an Espressif hardware product suitable for your needs by comparing or applying filters.
<https://products.espressif.com/#/product-selector>

4.4 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>

Chapter 5

Disclaimer and Copyright Notice

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

All third party's information in this document is provided as is with no warranties to its authenticity and accuracy.

No warranty is provided to this document for its merchantability, non-infringement, fitness for any particular purpose, nor does any warranty otherwise arising out of any proposal, specification or sample.

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

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

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