

User Manual

JCICVT

2022

V0.01

1.OUTLINE

JCICVT is an ESP32-PICO-V3-02 series processor with two buttons and USB-C interface, RJ12 interface, 2.4G antenna and screen, and TF card slot.



1.1 Hardware Composition

JCICVT Hardware: ESP32-PICO-V3-02 chip, TFT display, RJ15 interface, buttons, Type-C interface, SCT9330DCDC.

The ESP32-PICO-V3-02 is a System-in-Package (SiP) device that is based on ESP32 with ECO V3 wafer, providing complete Wi-Fi and Bluetooth® functionalities. It integrates a 8 MB SPI flash and a 2 MB SPI PSRAM.

The TFT Screen is a 1.14-inch color Screen that drives the ST7789V2 and has a resolution of 135 x 240. The operating voltage range is 2.6~3.3V, and the operating temperature range is -25~55°C.

The power management chip is SCT9330. The operating voltage ranges from 3.8V to 28V, and the power supply current is 3A.

2. PIN DESCRIPTION

2.1. USB INTERFACE

JCICVT Configuration Type-C type USB interface, support USB2.0 standard communication protocol.



2.2. RJ-12 INTERFACE

6P RJ-12 ports with a spacing of 1.02mm are configured. Internal cables connect to the GND, power supply, RS485_P, and RS485_N ports.

3.FUNCTIONAL DESCRIPTION

This chapter describes the ESP32-PCIO-V3-02 various modules and functions.

3.1.CPU AND MEMORY

Xtensa®single-/dual-core32-bitLX6microprocessor(s), upto600MIPS
(200MIPSforESP32-S0WD/ESP32-U4WDH, 400 MIPS for ESP32-D2WD):

MCU

- ESP32 embedded, Xtensa® dual-core 32-bit LX6 microprocessor, up to 240 MHz
- · 448 KB ROM
- · 520 KB SRAM
- · 16 KB RTC SRAM

3.2.STORAGE DESCRIPTION

3.2.1.External Flash and SRAM

ESP32 support multiple external QSPI flash and static random access memory (SRAM), having a hardware-based AES encryption to protect the user programs and data.

- ESP32 access external QSPI Flash and SRAM by caching. Up to 16 MB external Flash code space is mapped into the CPU, supports 8-bit, 16-bit and 32-bit access, and can execute code.
- Up to 8 MB external Flash and SRAM mapped to the CPU data space, support for 8-bit, 16-bit and 32-bit access. Flash supports only read operations, SRAM supports read and write operations.

3.3.CRYSTAL

Esp32-pico-v3-02 SiP has seamlessly integrated all peripheral components such as crystal oscillator, Flash, PSRAM, filter capacitor, RF matching link into the package, and no peripheral components are needed to work.

3.4.RTC MANAGEMENT AND LOW POWER CONSUMPTION

ESP32 uses advanced power management techniques may be switched between different power saving modes. (See Table 5).

- *Power saving mode*
 - Active Mode: RF chip is operating. Chip may receive and transmit a sounding signal.
 - Modem-sleep mode: CPU can run, the clock may be configured. Wi-Fi / Bluetooth baseband and RF
 - Light-sleep mode: CPU suspended. RTC and memory and peripherals ULP coprocessor operation. Any wake-up event (MAC, host, RTC timer or external interrupt) will wake up the chip.
 - Deep-sleep mode: only the RTC memory and peripherals in a working state. Wi-Fi and Bluetooth connectivity data stored in the RTC. ULP coprocessor can work.
 - Hibernation Mode: 8 MHz oscillator and a built-in coprocessor ULP are disabled. RTC memory to restore the power supply is cut off. Only one RTC clock timer located on the slow clock and some RTC GPIO at work. RTC RTC clock or timer can wake up from the GPIO Hibernation mode.
- *Deep-sleep mode*
 - related sleep mode: power save mode switching between Active, Modem-sleep, Light-sleep mode. CPU, Wi-Fi, Bluetooth, and radio preset time interval to be awakened, to ensure connection Wi-Fi / Bluetooth.
 - Ultra Low-power sensor monitoring methods: the main system is Deep-sleep mode, ULP coprocessor is periodically opened or closed to measure sensor data. The sensor measures data, ULP coprocessor decide whether to wake up the main system.

Functions in different modes: TABLE 5

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

4. ELECTRICAL CHARACTERISTICS

4.1 WIFI/BLUETOOTH

RF Function: 2.4G Wi-Fi

Operating Band/Frequency: 2412-2462 MHz

Maximum Output Power:20.99dBm

RF Function: BLE

Operating Band /Frequency: 2402-2480 MHz

Maximum Output Power:2.34dBm

RF Function: BT

Operating Band /Frequency: 2402-2480 MHz

Maximum Output Power:-0.02dBm

This product is a field control bus with multiple communication protocols in one or A central control device for communication between a universal digital controller and an extender module, It is used in digital scenarios such as building automation and industrial air conditioning security.

Parameters: 1. Supply voltage: 5V DC
2. RS485 bus voltage: 12V DC
3. Working temperature: -20°C - 60°C
4. Humidity: 5% - 95%



Key functions:

- A: Function selection button
- B: Communication mode toggle button
- C: 1.14 inch LCD screen
- D: Type C interface
- E: SD card slot
- F: RS485 interface
- G: RS485 communication protocol switch

Operation Process:

1. Select the appropriate communication mode (SD card data reading mode, RS485 communication or serial communication) to insert the SD card or access the RS485 data line.
2. Connect the Type C data cable to power on.
3. Press the key B to switch the communication mode.
4. Press key A to select the required function.

1. QUICK START

1.1. ARDUINO IDE

Visit Arduino's official website(<https://www.arduino.cc/en/Main/Software>), Select the installation package for your own operating system to download.

>1. Open up Arduino IDE, navigate to `File` -> `Preferences` -> `Settings`

>2. Copy the following M5Stack Boards Manager url to `Additional Boards Manager URLs`:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_dev_index.json

>3. Navigate to `Tools` -> `Board:` -> `Boards Manager...`

>4. Search `ESP32` in the pop-up window, find it and click `Install`

>5. select `Tools` -> `Board:` -> `ESP32-Arduino-ESP32 DEV Module`

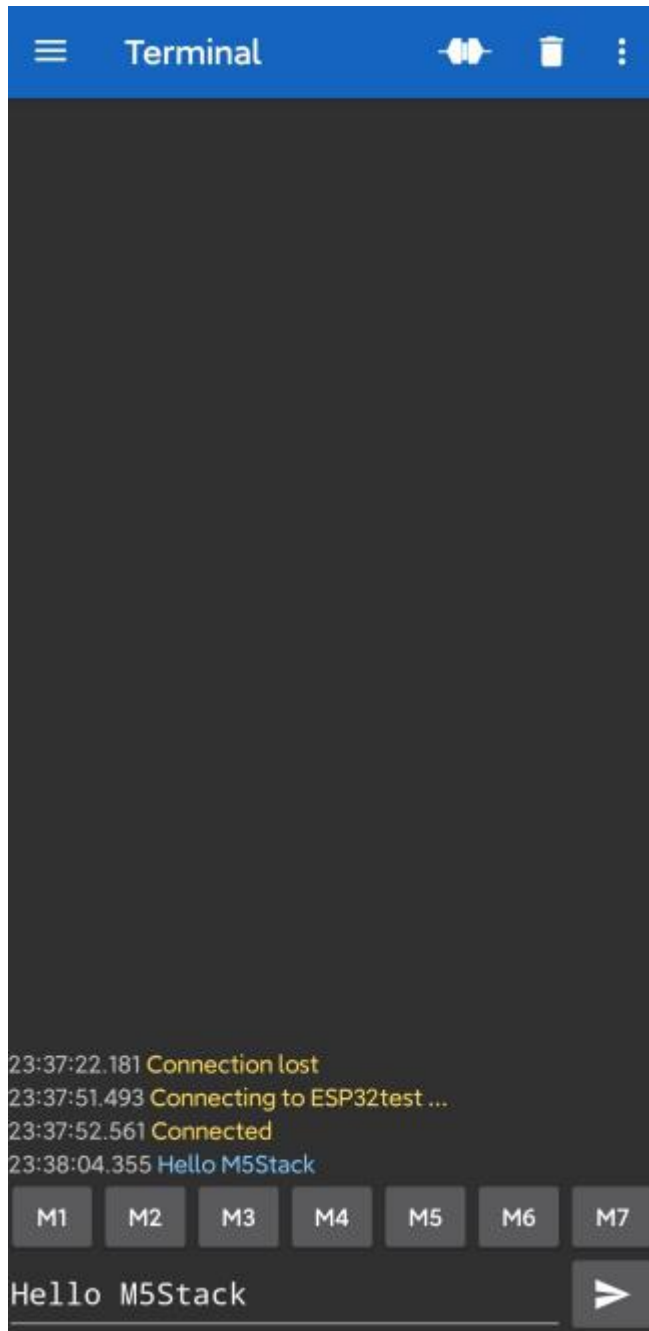
>6 Please install CH9102 driver before use esp32 downloader kit burn the firmware:

<https://shop.m5stack.com/products/esp32-downloader-kit>

1.2. BLUETOOTH SERIAL

Open the Arduino IDE and open the example program

`File` -> `Examples` -> `BluetoothSerial` -> `SerialToSerialBT`. Connect the device to the computer and select the corresponding port to burn. After completion, the device will automatically run Bluetooth, and the device name is `ESP32test`. At this time, use the Bluetooth serial port sending tool on the PC to realize the transparent transmission of Bluetooth serial data.



```
#include "BluetoothSerial.h"

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif
```

```

BluetoothSerial SerialBT;

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32test"); //Bluetooth device name
  Serial.println("The device started, now you can pair it with bluetooth!");
}

void loop() {
  if (Serial.available()) {
    SerialBT.write(Serial.read());
  }
  if (SerialBT.available()) {
    Serial.write(SerialBT.read());
  }
  delay(20);
}

```

1.3. WIFI SCANNING

Open the Arduino IDE and open the example program `File` -> `Examples` -> `WiFi` -> `WiFiScan`. Connect the device to the computer and select the corresponding port to burn. After completion, the device will automatically run the WiFi scan, and the current WiFi scan result can be obtained through the serial port monitor that comes with the Arduino.

The screenshot shows the Arduino IDE interface with the 'WiFiScan' example code loaded. The serial monitor is open, displaying the results of a WiFi scan performed by the ESP32. The code in the background includes a setup phase that prints 'Setup done' and a loop phase that prints 'scan start' and 'scan done', followed by a list of 17 detected WiFi networks. The serial monitor output matches this, showing the scan results in a numbered list format.

```

WiFiScan | Arduino 1.8.12
File Edit Sketch Tools Help

WiFiScan
// WiFiScan
WiFi.disconnect();
delay(100);

Serial.println("Setup done");
}

void loop()
{
  Serial.println("scan start");
  // WiFi.scanNetworks will return the number of networks found
  int n = WiFi.scanNetworks();
  Serial.println("scan done");
  if (n == 0) {
    Serial.println("no networks found");
  } else {
    Serial.print(n);
    Serial.println(" networks found");
    for (int i = 0; i < n; i++) {
      // Print SSID of each network found
      Serial.print(i+1);
      Serial.print(": ");
      Serial.print(WiFi.SSID(i));
      Serial.print(" (");
      Serial.print(WiFi.RSSI(i));
      Serial.print(")");
      delay(10);
    }
    Serial.println("");
    // Wait a bit before scanning again
    delay(5000);
  }
}

```

```

COM85
scan start
scan done
17 networks found
1: cam (-47)*
2: M5-2.4G (-50)*
3: WirelessNet (-55)*
4: M5-2.4G (-60)*
5: M5-2.4G (-62)*
6: ChinaNet-yeTW (-65)*
7: TP-LINK_6666BA (-69)*
8: DIRECT-9d-HP M277 LaserJet (-71)*
9: 905 (-72)*
10: boluojun (-72)*
11: TP-LINK_CS2_666 (-78)*
12: CFSZ1 (-84)*
13: fuxiwenhua (-86)*
14: XM-Web (-87)
15: XM-Guest (-88)
16: CFSZ1 (-90)*
17: XM-free (-91)*

```

```

#include "WiFi.h"

void setup()
{
    Serial.begin(115200);
    // Set WiFi to station mode and disconnect from an AP if it was previously connected
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    delay(100);

    Serial.println("Setup done");
}

void loop()
{
    Serial.println("scan start");

    // WiFi.scanNetworks will return the number of networks found
    int n = WiFi.scanNetworks();
    Serial.println("scan done");
    if (n == 0) {
        Serial.println("no networks found");
    } else {
        Serial.print(n);
        Serial.println(" networks found");
        for (int i = 0; i < n; ++i) {
            // Print SSID and RSSI for each network found
            Serial.print(i + 1);
            Serial.print(": ");
            Serial.print(WiFi.SSID(i));
            Serial.print(" (");
            Serial.print(WiFi.RSSI(i));
            Serial.print(")");
            Serial.println((WiFi.encryptionType(i) == WIFI_AUTH_OPEN)?"
":"*");
            delay(10);
        }
        Serial.println("");

        // Wait a bit before scanning again
        delay(5000);
    }
}

```

FCC Statement

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

- (1) This device may not cause harmful interference, and
- (2) This device must accept any interference received, including interference that may cause undesired operation.

FCC Radiation Exposure Statement:

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.

Note : This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

ISED RSS Warning/ISED RF Exposure Statement**ISED RSS Warning:**

This device complies with Innovation, Science and Economic Development Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions:

- (1) this device may not cause interference, and
- (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'ISED applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes:

- (1) l'appareil ne doit pas produire de brouillage, et
- (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

ISED RF exposure statement:

This equipment complies with ISED radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body. This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

Le rayonnement de la classe B respecte ISED fixaient un environnement non contrôlés. Installation et mise en œuvre de ce matériel devrait avec échangeur distance minimale entre 20 cm ton corps. Lanceurs ou ne peuvent pas coexister cette antenne ou capteurs avec d'autres.