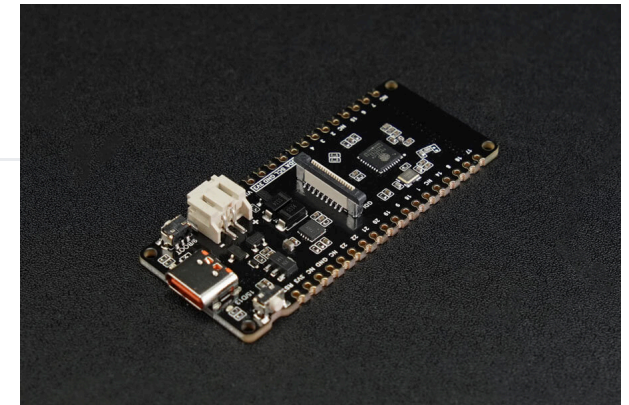


SKU:DFR1075 (<https://www.dfrobot.com/product-2771.html>)

(<https://www.dfrobot.com/product-2771.html>)

Introduction

FireBeetle 2 ESP32-C6 is a low-power IoT main control board designed based on the ESP32-C6 chip. It is suitable for smart home projects. The ESP32-C6 supports communication protocols such as Wi-Fi 6, Bluetooth 5, Zigbee 3.0, and Thread 1.3, enabling connectivity to various IoT networks. FireBeetle 2 ESP32-C6 supports Type-C, 5V DC, and solar power, providing more options for power supply during deployment.



Multiple protocol support for expanded wireless connectivity

FireBeetle 2 ESP32-C6 supports Wi-Fi and Thread communication protocols, allowing seamless communication and collaboration between Matter Wi-Fi terminal devices and Matter Thread terminal devices in multi-system, multi-platform smart home setups. Additionally, FireBeetle 2 ESP32-C6 also supports BLE and Zigbee communication protocols, enabling it to serve as a Thread border router, Matter gateway, and Zigbee bridge when combined with other MCUs.

Wi-Fi 6 support for ultra-low-power IoT devices

ESP32-C6 is Espressif's first chip to support the Wi-Fi 6 (802.11ax) protocol. Wi-Fi 6 offers improved network capacity, enabling devices to work with higher efficiency and lower latency. Moreover, Wi-Fi 6's Target Wake Time (TWT) technology effectively reduces device power consumption, extends battery life, and enables long-lasting device operation.

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents

>

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents



Excellent power system for convenient device power supply

FireBeetle 2 ESP32-C6 integrates lithium battery charging management, allowing the lithium battery to be charged via Type-C, 5V DC, and solar panels. Solar charging solves the problem of deploying devices in scenarios without power lines, such as balconies, windows, and rooftops. FireBeetle 2 ESP32-C6 uses the same solar power management chip as the Solar Power Manager 5V (<https://www.dfrobot.com/product-1712.html>) and has a maximum power point tracking MPPT algorithm that can maximize the output power of the solar panel under various lighting conditions. FireBeetle 2 ESP32-C6 also supports battery level monitoring to take measures when the battery is low, ensuring continuous device operation.

If you encounter difficulties with programming, please refer to the FAQ (https://wiki.dfrobot.com/SKU_DFR1075_FireBeetle_2_Board_ESP32_C6#target_9) section.

Features

- Equipped with the ESP32-C6 chip, it supports communication protocols including Wi-Fi, BLE, Zigbee, and Thread.
- Supports Wi-Fi 6 protocol for lower latency and lower power consumption.
- Solar charging enables deployment without the need for power lines.
- Battery level detection allows monitoring of device battery information.
- Ultra-low power consumption with deep sleep at 16uA.
- Onboard GDI interface for easy connection to screens.

Specification

- Basic Parameters
 - Operating Voltage: 3.3V
 - Type-C Input Voltage: 5V DC
 - VCC Input Voltage: 5V DC or 4.5-6V Solar Panel

[Introduction](#)[Features](#)[Specification](#)[Board Overview](#)[Pin Diagram](#)[Tutorial – First Time Use](#)[Basic Tutorial](#)[Advanced Tutorial](#)[Tutorial for MicroPython](#)[FAQ](#)[More Documents](#)

>

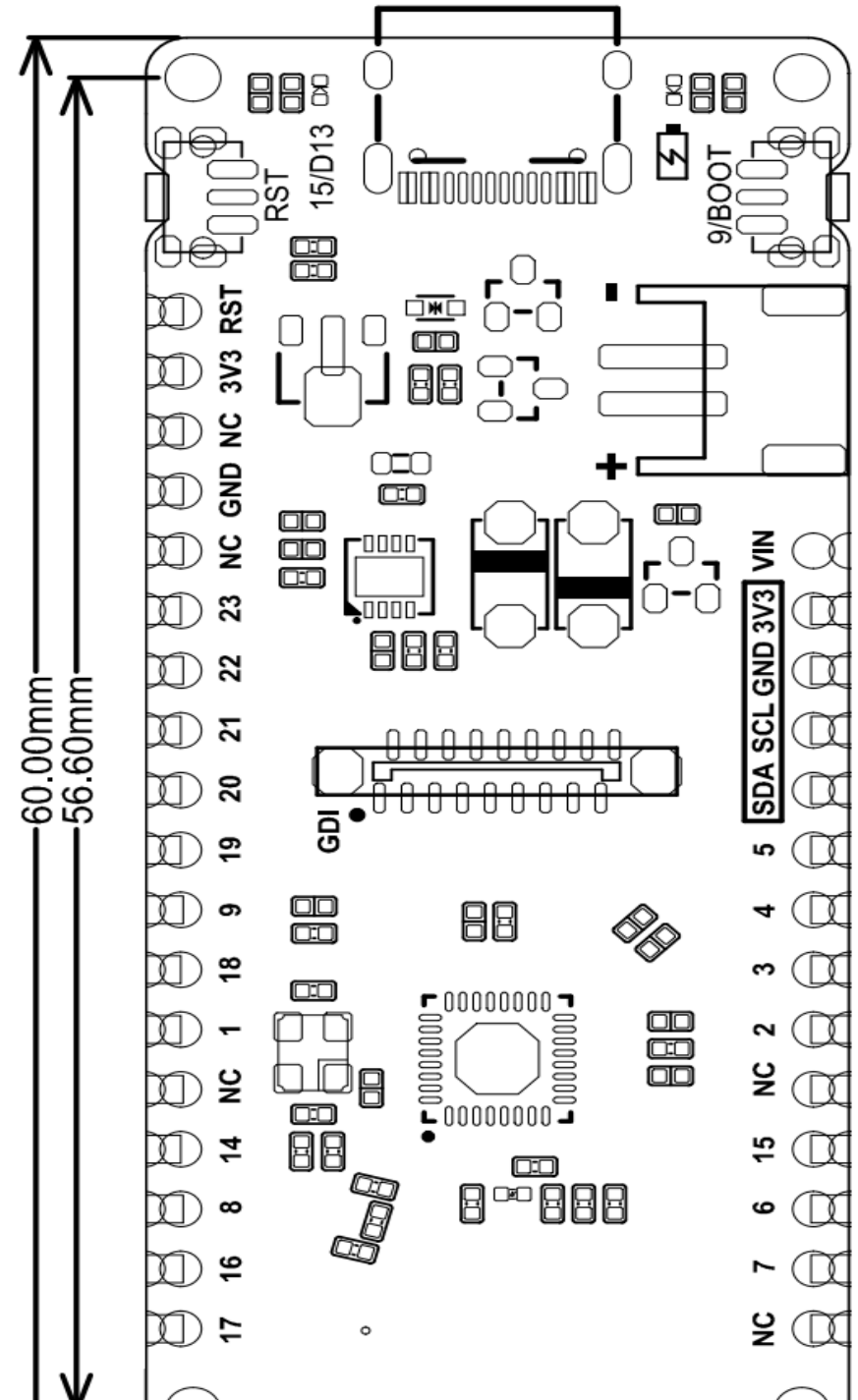
- Max Charging Current: 0.5A
- Sleep current: 16uA (in deep sleep mode, powered by battery)
- Operating Temperature: -10~60°C
- Dimension: 25.4x60mm/1x2.36"

- Hardware Information

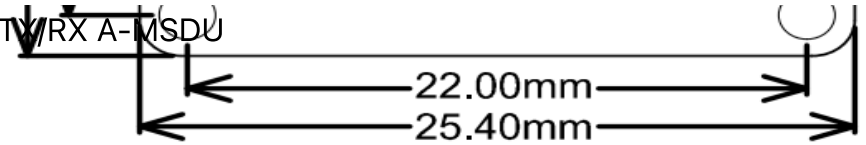
- Processor: RISC-V single-core processor
- Main Frequency: 160 MHz
- SRAM: 512KB
- ROM: 320KB
- Flash: 4MB
- RTC SRAM: 16KB
- USB: USB 2.0 CDC

- WIFI

- WIFI Protocol: IEEE 802.11b/g/n
 - IEEE 802.11ax (20 MHz-only non-AP mode)
- Bandwidth: Support 20 MHz and 40 MHz at 2.4 GHz band
- WIFI Mode: Station, SoftAP, SoftAP+Station combined mode
- WIFI Frequency: 2.4GHz



- Frame Aggregation: TX/RX A-MPDU, TX/RX A-MSDU
- Bluetooth
 - Bluetooth Protocol: Bluetooth 5, Bluetooth mesh
 - Bluetooth Frequency: 125 Kbps, 500 Kbps, 1 Mbps, 2 Mbps
- IEEE 802.15.4
 - Compatible with IEEE 802.15.4-2015 protocol
 - Frequency band: 2.4GHz
 - Data rate: 250Kbps
 - Supports Thread 1.3 and Zigbee 3.0
- Ports
 - Digital I/O x19
 - LED PWM 6 Channel
 - SPI x1
 - UART x3 (LP UART x1)
 - I2C x2 (LP I2C x1)
 - I2S x1
 - IR Transceiver: transmit channel x5, receive channel x5
 - 1 × 12-bit SAR ADC, 7 Channel
 - DMA Controller: transmit channel x3, receive channel x3



[Introduction](#)

[Features](#)

[Specification](#)

[Board Overview](#)

[Pin Diagram](#)

[Tutorial – First Time Use](#)

[Basic Tutorial](#)

[Advanced Tutorial](#)

[Tutorial for MicroPython](#)

[FAQ](#)

[More Documents](#)

>

Board Overview

[Introduction](#)

[Features](#)

[Specification](#)

[Board Overview](#)

[Pin Diagram](#)

[Tutorial – First Time Use](#)

[Basic Tutorial](#)

[Advanced Tutorial](#)

[Tutorial for MicroPython](#)

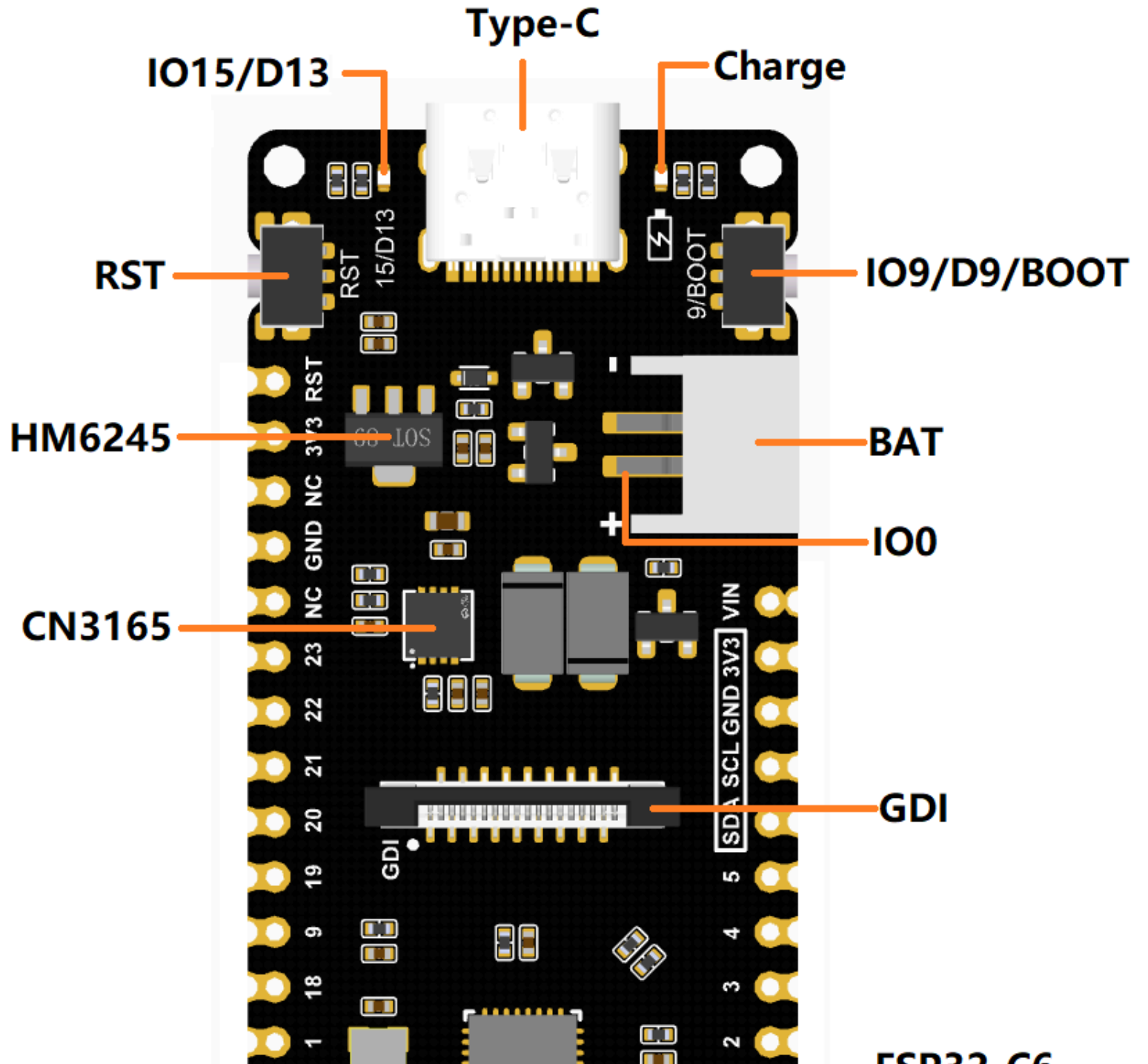
[FAQ](#)

[More Documents](#)

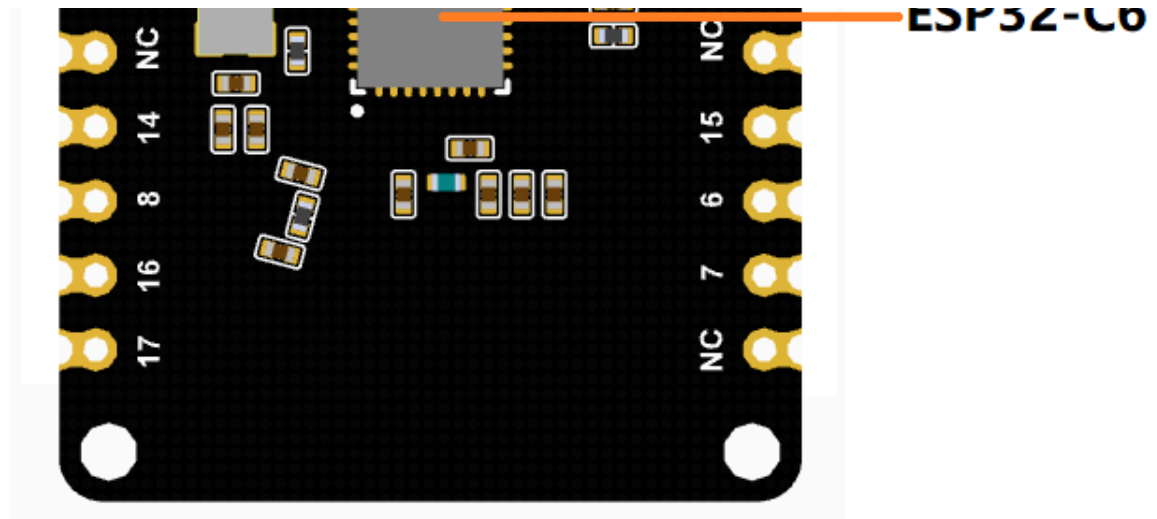


- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

>



- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents



- **Type-C:**Type-C USB port
- **IO15/D13:**onboard LED pin
- **Charge:** Charging indicator
 - Off: not plugged in power supply or fully charged
 - On: charging
 - Blinking: battery not connected
- **RST:** Reset button
- **IO9/D9/BOOT:** GPIO9 / Boot button
- **HM6245**
 (<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/6f630301d84caf0e92266e3c5cf11edc.PDF>): 3.3V Low power LDO
- **BAT:**Lithium ion battery or lithium polymer battery interface
- **IO0:** Battery voltage detection pin
- **CN3165**
 (<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/85759bb076bf6bb24fd8ca5683f19603.pdf>): Solar management chip

- **GDI:** GDI display interface
- **ESP32-C6**
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/5eeffd974782b6cc442f7a2ee7f00577.pdf>): ESP32-C6FH4 chip

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents



Pin Diagram

[Introduction](#)

[Features](#)

[Specification](#)

[Board Overview](#)

[Pin Diagram](#)

[Tutorial – First Time Use](#)

[Basic Tutorial](#)

[Advanced Tutorial](#)

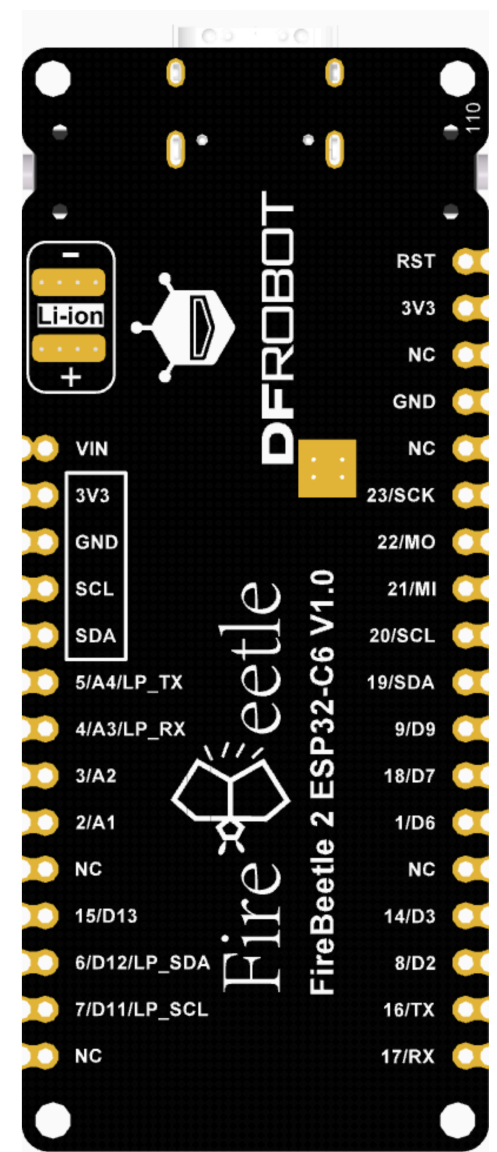
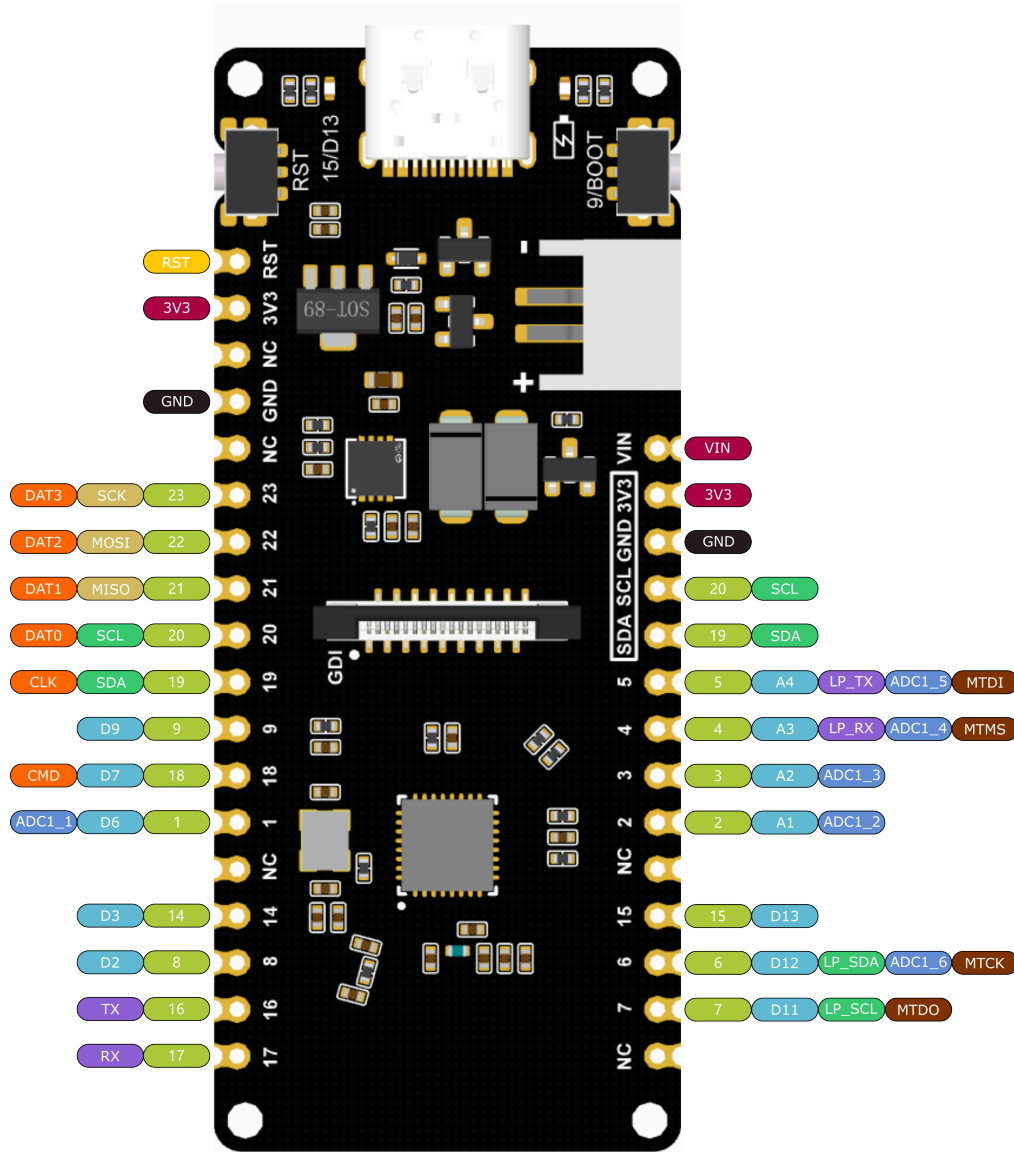
[Tutorial for MicroPython](#)

[FAQ](#)

[More Documents](#)



- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents



| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |

Pin Definition

[Introduction](#)

[Features](#)

[Specification](#)

[Board Overview](#)

[Pin Diagram](#)

[Tutorial – First Time Use](#)

[Basic Tutorial](#)

[Advanced Tutorial](#)

[Tutorial for MicroPython](#)

[FAQ](#)

[More Documents](#)

>

- Power: Power pin
 - VIN: 5V DC or 4.5-6V Solar Panel
 - 3V3: 3.3V stable output
- GND: common ground pin
- GPIO: ESP32 default GPIO number
- Arduino: FirebBeetle 2 ESP32-C6 GPIO mapping in Arduino
- ADC: ESP32 default analog-to-digital conversion pin
- I2C: I2C interface
 - FirebBeetle 2 ESP32-C6 I2C mapping in Arduino
 - LP_SDA/SCL: Low power I2C pin
- UART: UART interface
 - LP_TX/RX: Low power UART pin
- SPI: FirebBeetle 2 ESP32-C6 SPI mapping in Arduino
- SDIO: ESP32 default SDIO pin
- JTAG: debug interface

Connect Solar Panels

[Introduction](#)

[Features](#)

[Specification](#)

[Board Overview](#)

[Pin Diagram](#)

[Tutorial – First Time Use](#)

[Basic Tutorial](#)

[Advanced Tutorial](#)

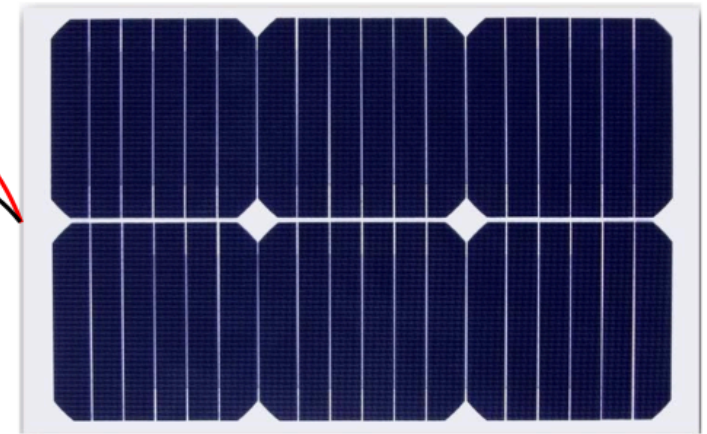
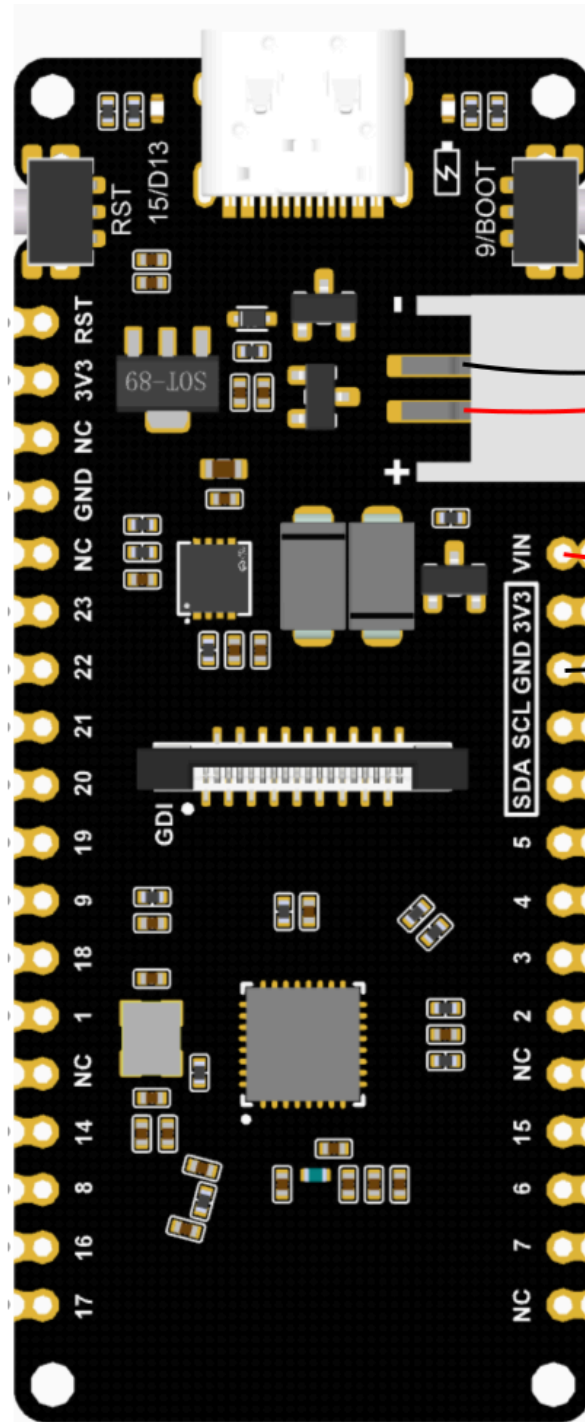
[Tutorial for MicroPython](#)

[FAQ](#)

[More Documents](#)



- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents



GDI Display Interface

This interface is a DFRbot dedicated GDI display interface for connecting a screen using a 18pin-FPC wire.

The pin list for using GDI camera interface is shown below:

| FPC PINS | FireBeetle 2 ESP32-C6 Pins | Description |
|----------|----------------------------|--------------------------|
| VCC | 3V3 | 3.3V |
| LCD_BL | 15/D13 | Backlight |
| GND | GND | GND |
| SCLK | 23/SCK | SPI clock |
| MOSI | 22/MOSI | Host output, slave input |
| MISO | 21/MISO | Host input, slave output |
| LCD_DC | 8/D2 | Data/command |
| LCD_RST | 14/D3 | Reset |
| LCD_CS | 1/D6 | TFT Chip Select |
| SD_CS | 18/D7 | SD card chip select |
| FCS | NC | Font library chip select |
| TCS | 6/D12 | Touch chip select |
| SCL | 20/SCL | I2C clock |
| SDA | 19/SDA | I2C data |

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents

>

[Introduction](#)[Features](#)[Specification](#)[Board Overview](#)[Pin Diagram](#)[Tutorial – First Time Use](#)[Basic Tutorial](#)[Advanced Tutorial](#)[Tutorial for MicroPython](#)[FAQ](#)[More Documents](#)

>

| FPC PINS | FireBeetle 2 ESP32-C6 Pins | Description |
|----------|----------------------------|----------------|
| INT | 7/D11 | INT |
| BUSY | NC | Tearproof pins |
| X1 | NC | custom pin 1 |
| X2 | NC | custom pin 2 |

When using FPC to connect the screen, please configure the corresponding pin numbers according to the GDL demo. Normally, only three pins need to be configured on different main controllers.

Displays that support GDI:

- 1.54" 240x240 IPS wide viewing angle TFT display (<https://www.dfrobot.com/product-2072.html>)
- 1.8" 128x160 IPS TFT LCD Display (<https://www.dfrobot.com/product-2580.html>)
- 2.0" 320x240 IPS wide viewing angle TFT display (<https://www.dfrobot.com/product-2071.html>)
- 2.8" 320x240 IPS TFT resistive touch display (<https://www.dfrobot.com/product-2106.html>)
- 3.5" 480x320 IPS TFT capacitive touch display (<https://www.dfrobot.com/product-2107.html>)
- 1.51" OLED Transparent Display with Converter (<https://www.dfrobot.com/product-2521.html>)

Tutorial - First Time Use

Arduino IDE Configuration

Please pay attention to the followings when using FireBeetle 2 ESP32-C6 for the first time.

1. Add the json link in the IDE
2. Download the core of the MCU
3. Select the development board and serial port

4. Open the sample code and burn it into the board
5. Get to know the serial monitor

Arduino IDE compiler environment config

- Configure URL to the Arduino IDE
1. Open Arduino IDE and click File->Preferences, as shown below.

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

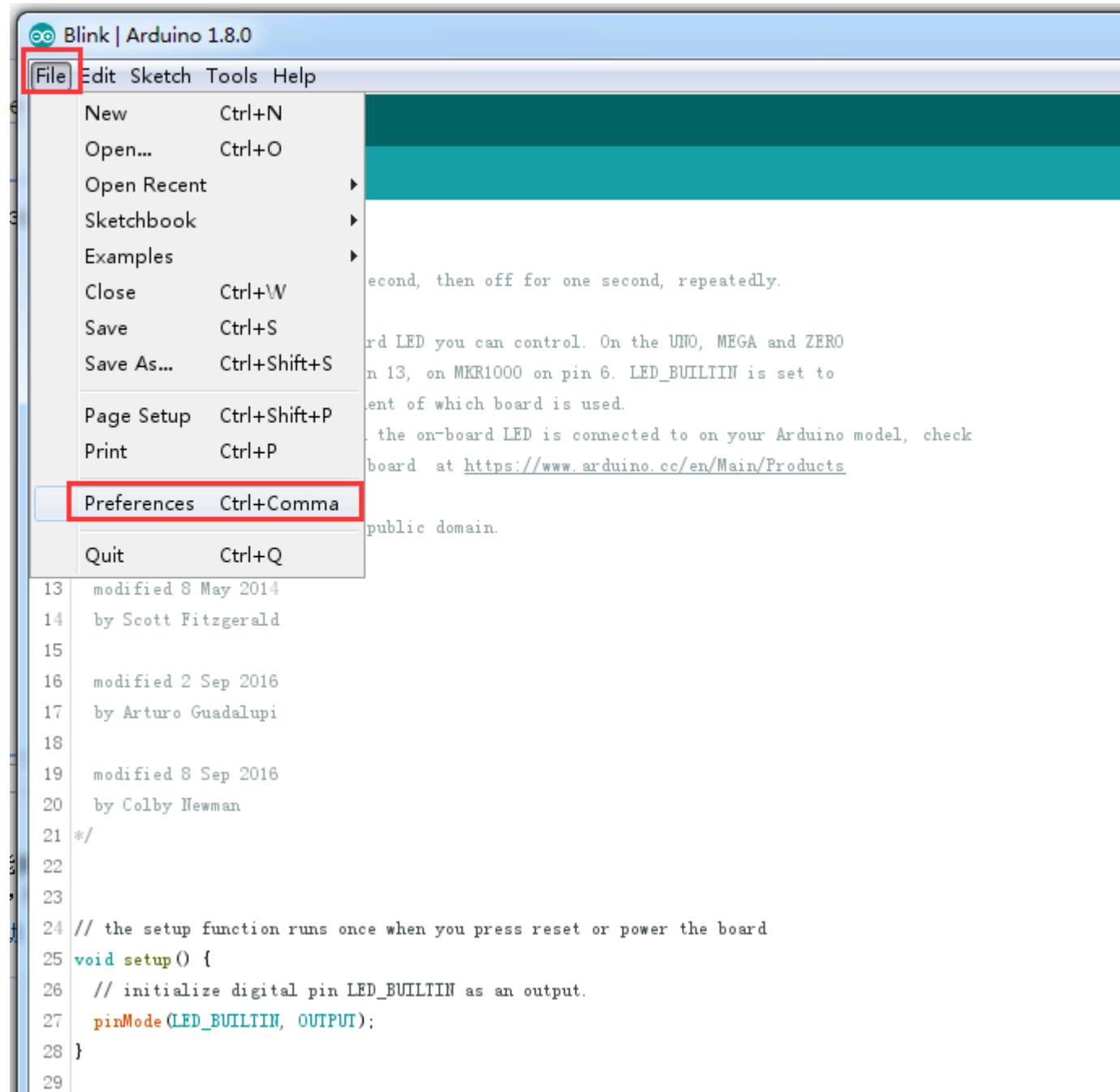
FAQ

More Documents

>

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

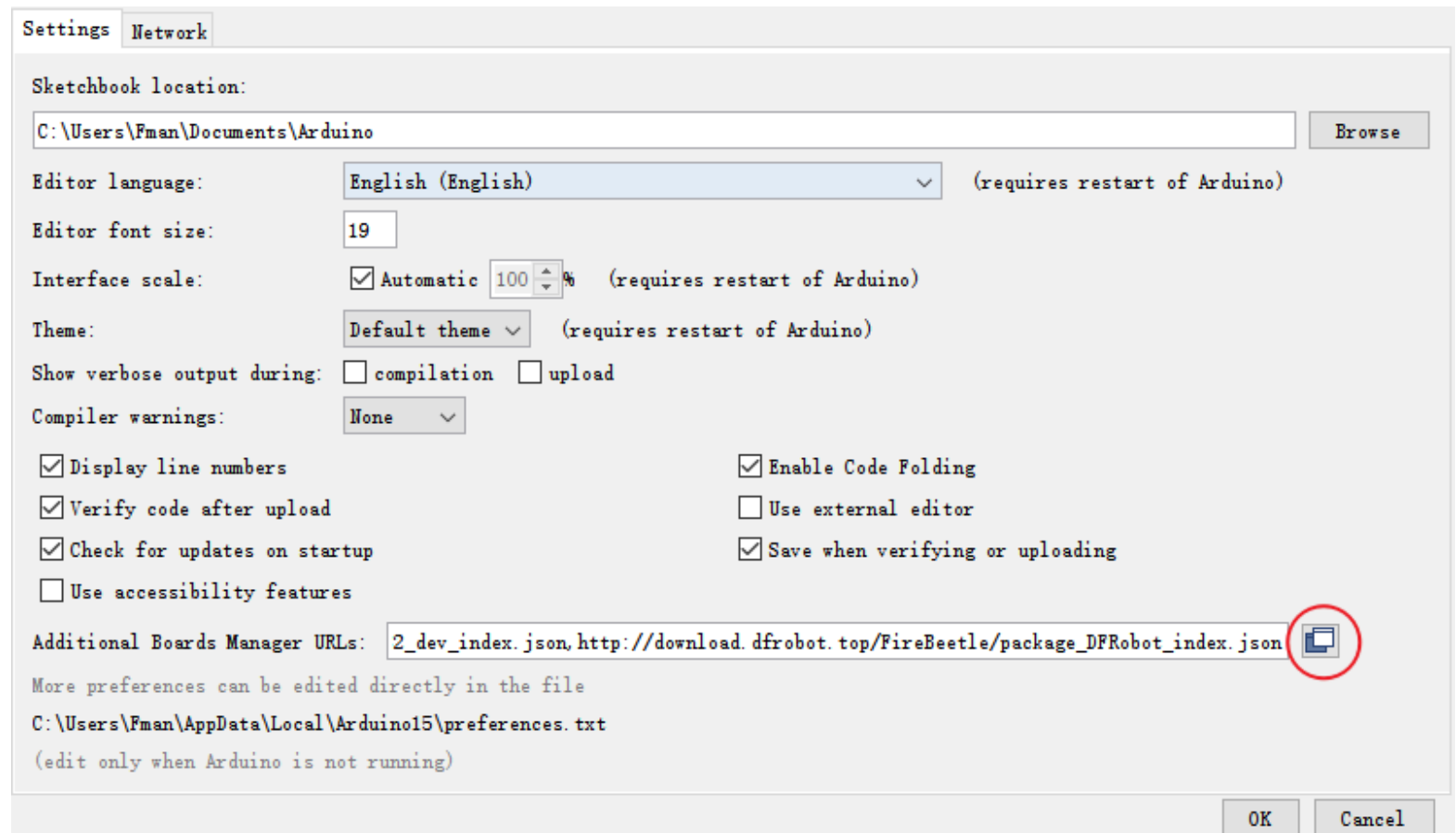
>



2. In the newly opened interface, click the button in the red circle as shown below

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

>



3. Copy the following link into the new pop-up dialog box:

Stable version: https://espressif.github.io/arduino-esp32/package_esp32_index.json

(https://espressif.github.io/arduino-esp32/package_esp32_index.json)

Development release: https://espressif.github.io/arduino-esp32/package_esp32_dev_index.json

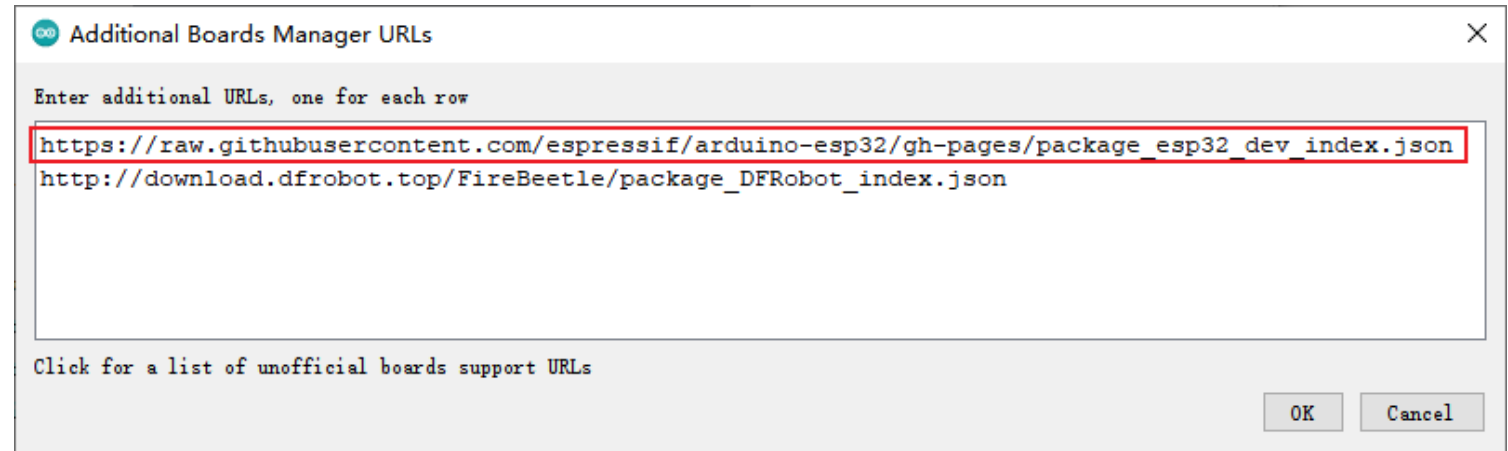
(https://espressif.github.io/arduino-esp32/package_esp32_dev_index.json)

Note:

- Please choose the appropriate version according to Chip Support Situation (<https://github.com/espressif/arduino-esp32#supported-chips>).

- If you have installed another environment before, you can press Enter key at the beginning or end of the previous link and paste the link at a new line.

Introduction
Features
Specification
Board Overview
Pin Diagram
Tutorial – First Time Use
Basic Tutorial
Advanced Tutorial
Tutorial for MicroPython
FAQ
More Documents

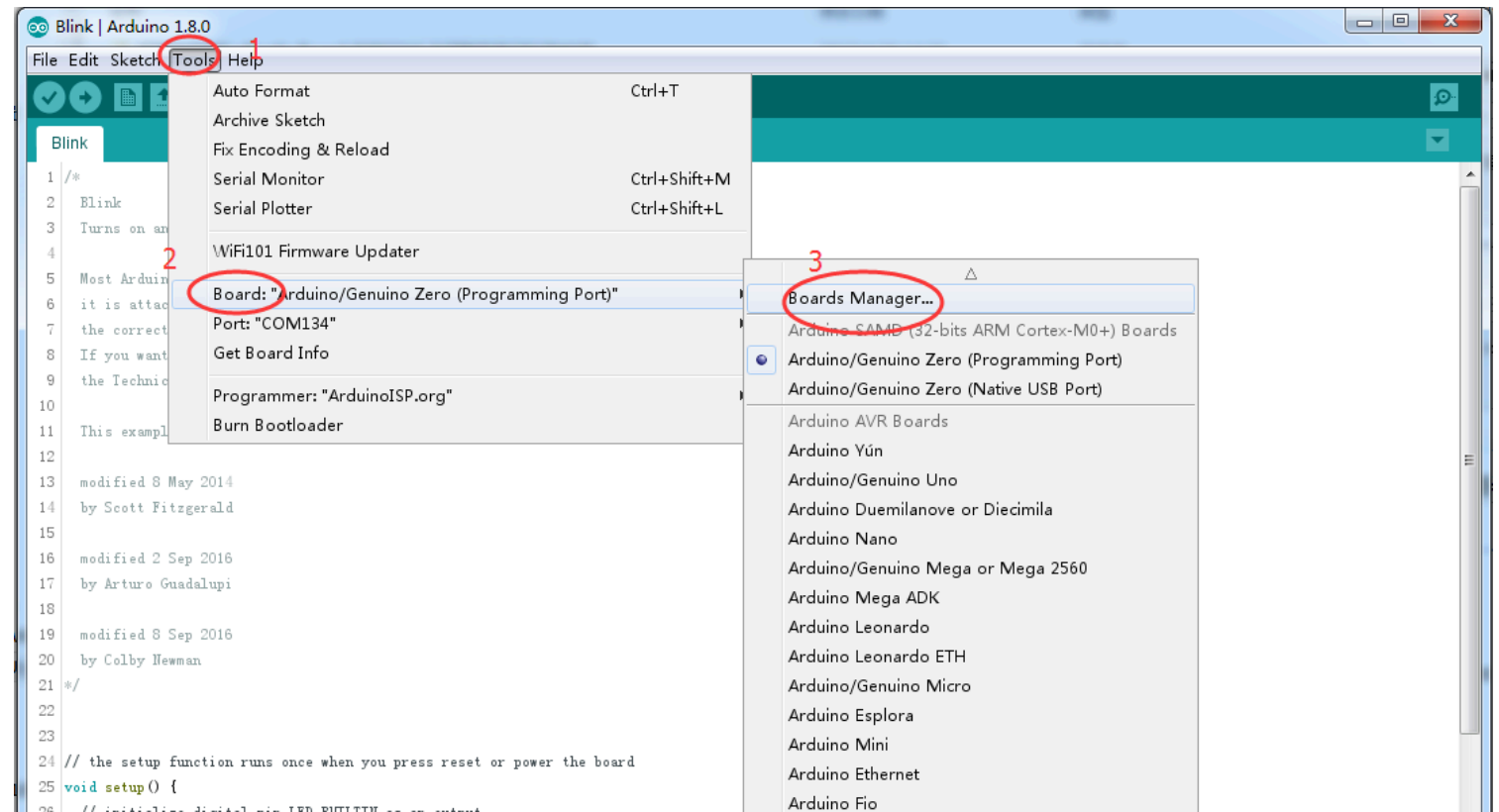


4. Click OK. Update the board. Open Tools->Board:->Boards Manager... as shown below:

>

- Introduction
 - Features
 - Specification
 - Board Overview
 - Pin Diagram
 - Tutorial – First Time Use
 - Basic Tutorial
 - Advanced Tutorial
 - Tutorial for MicroPython
 - FAQ
 - More Documents
-

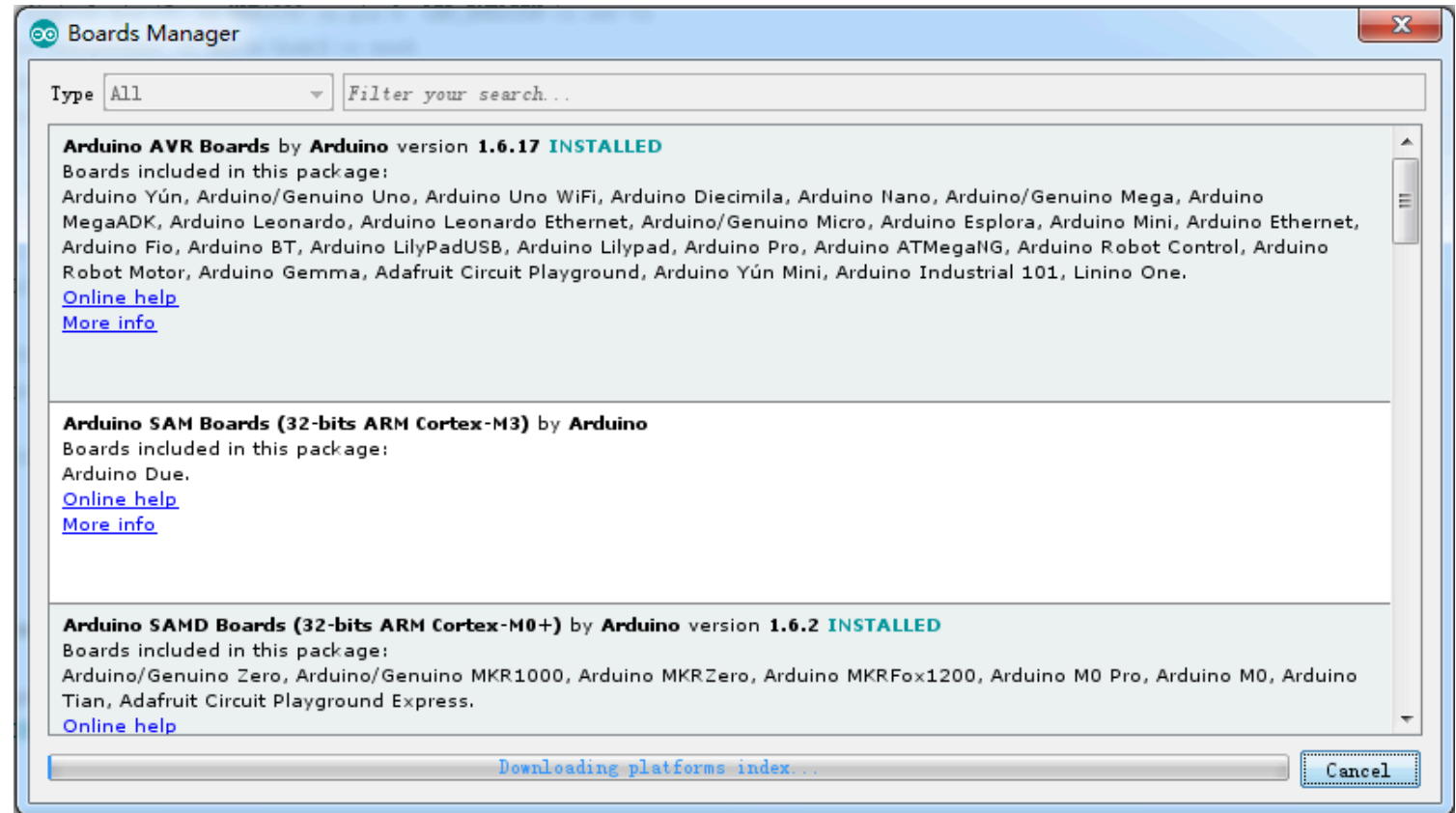
>



5. Boards Manager will automatically update the boards as shown below:

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

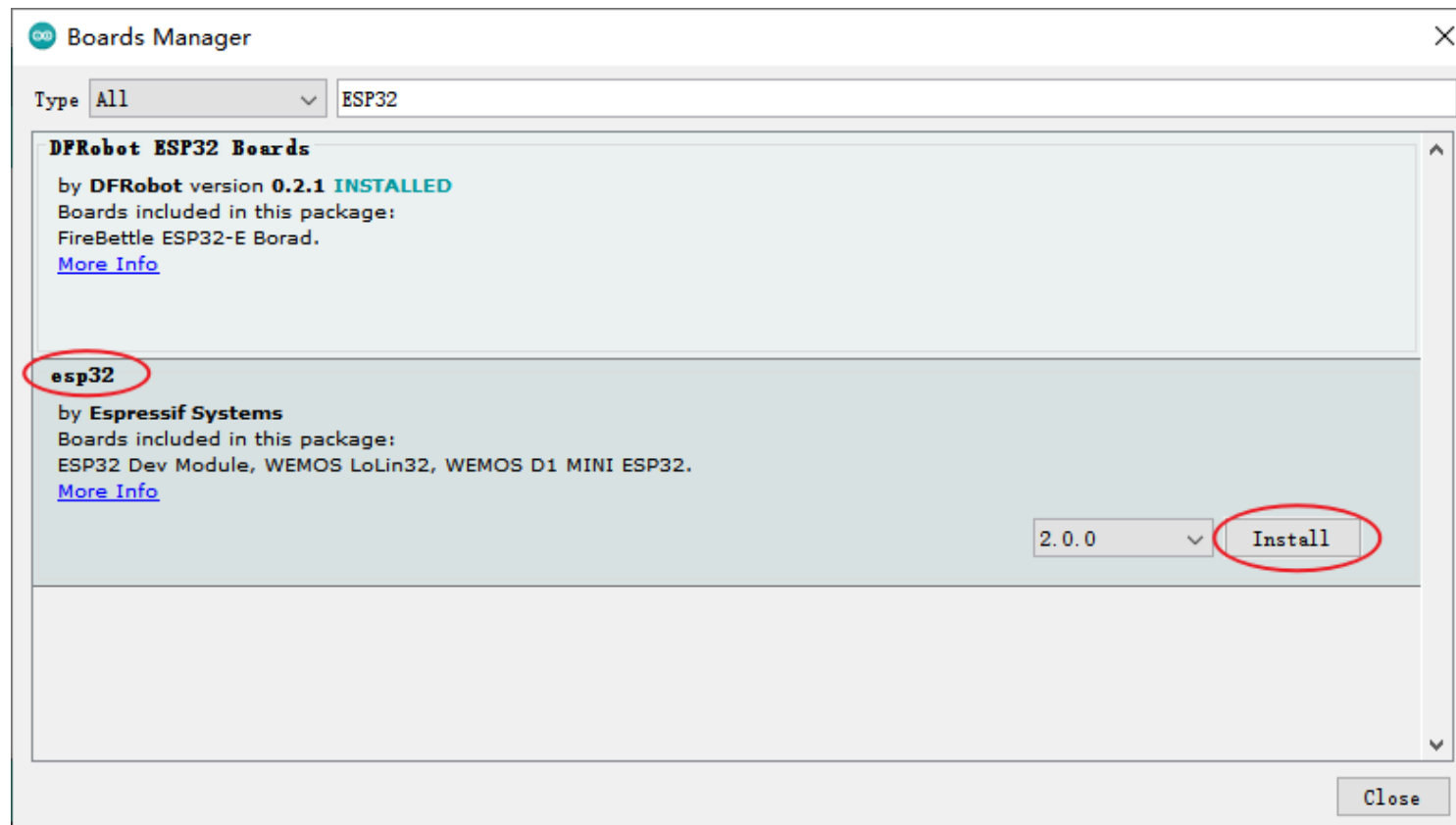
>



6. After completing the update, you can enter esp32 at the top, select esp32 and click install when the following occurs (It's recommended to install the latest version):

- Introduction
 - Features
 - Specification
 - Board Overview
 - Pin Diagram
 - Tutorial – First Time Use
 - Basic Tutorial
 - Advanced Tutorial
 - Tutorial for MicroPython
 - FAQ
 - More Documents
-

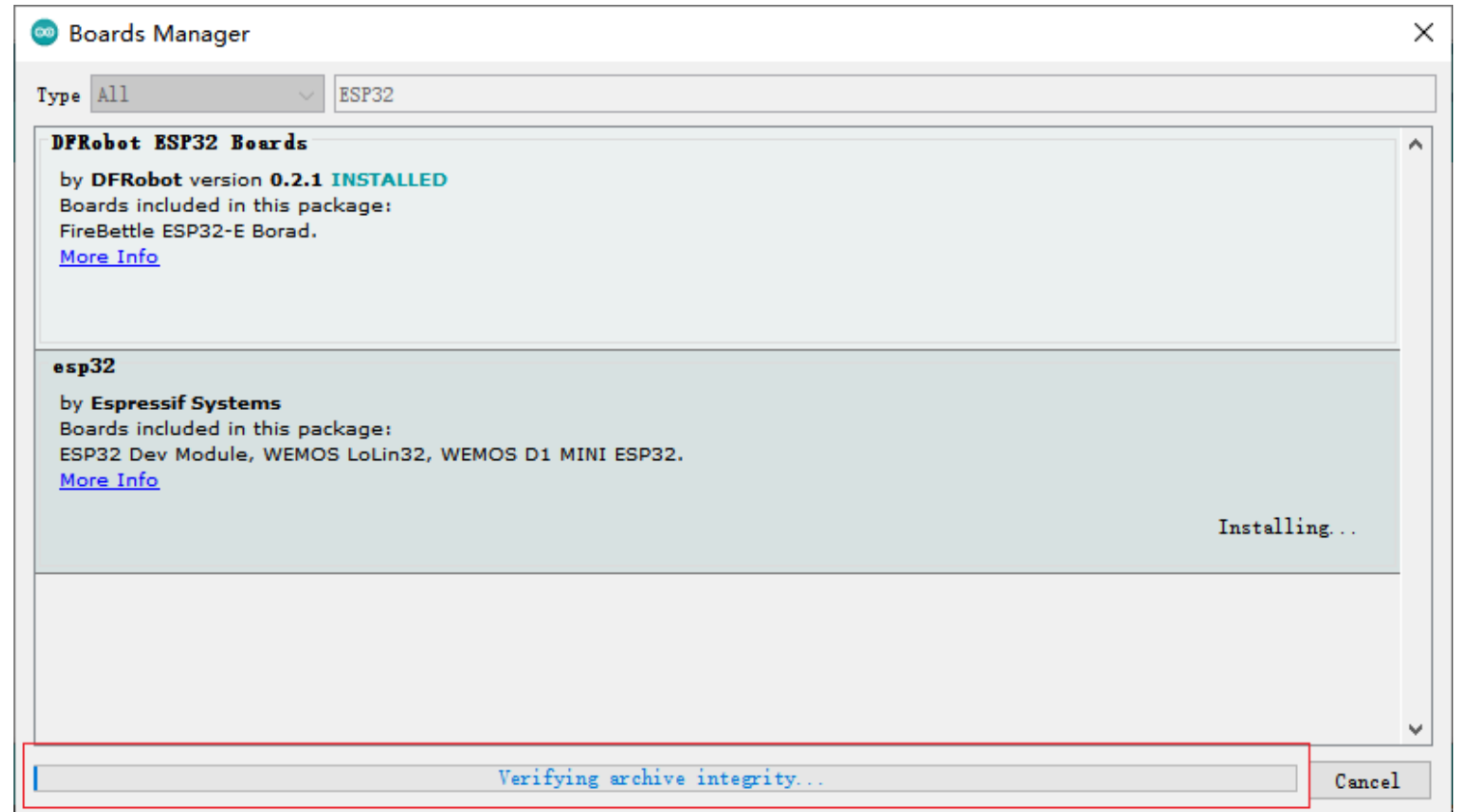
>



7. Wait for the end of the following progress bar:

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

>



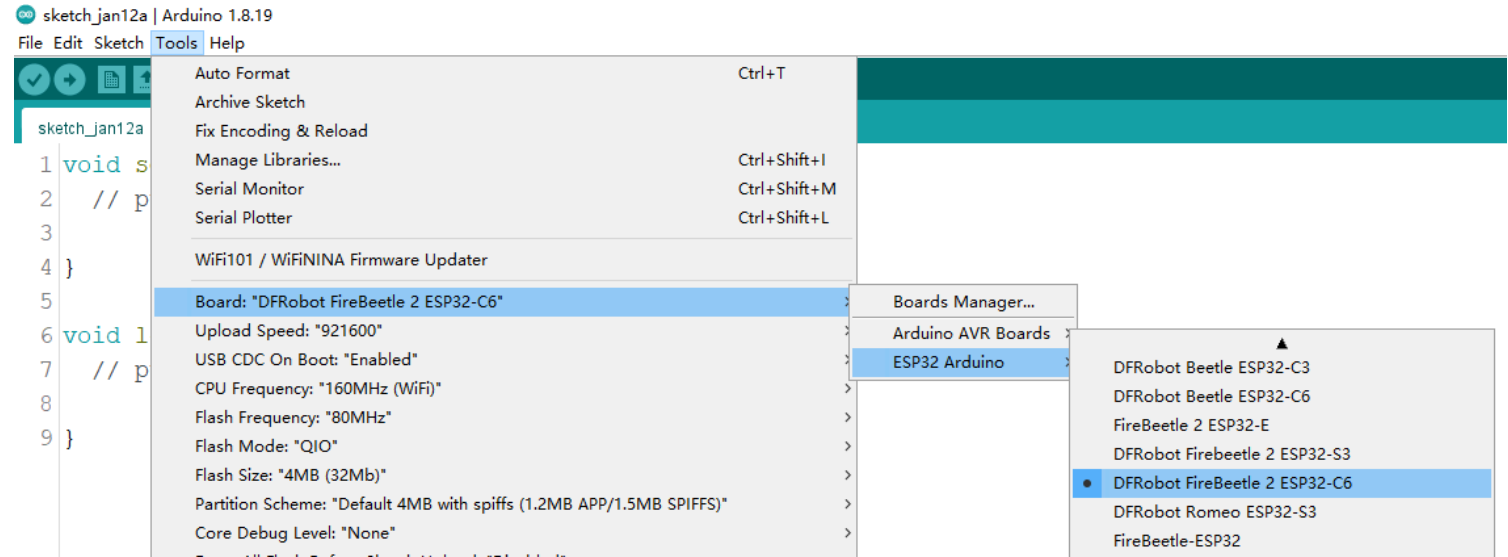
8. After completing the installation, the list will show that the esp32 has been installed, as shown below:

- Introduction
 - Features
 - Specification
 - Board Overview
 - Pin Diagram
 - Tutorial – First Time Use
 - Basic Tutorial
 - Advanced Tutorial
 - Tutorial for MicroPython
 - FAQ
 - More Documents
-



A screenshot of the 'Boards Manager' window in an IDE. The window title is 'Boards Manager' with a close button in the top right. At the top, there is a search bar with 'Type' set to 'All' and a filter set to 'ESP32'. The main content area is divided into two sections. The first section is titled 'DFRobot ESP32 Boards' and shows it is 'by DFRobot version 0.2.1 INSTALLED'. Below this, it lists 'Boards included in this package: FireBeetle ESP32-E Borad, FireBeetle ESP32 Borad.' and includes a 'More Info' link. The second section is titled 'esp32' and shows it is 'by Espressif Systems version 2.0.0 INSTALLED'. Below this, it lists 'Boards included in this package: ESP32 Dev Module, WEMOS LoLin32, WEMOS D1 MINI ESP32.' and includes a 'More Info' link. At the bottom of the second section, there is a 'Select version' dropdown menu, an 'Install' button, and a 'Remove' button. A 'Close' button is located at the bottom right of the window.

9. Click Tools->Board, select DFRobot FireBeetle 2 ESP32-C6.



10. Before starting, you need to configure the following settings (when you select Disabled, the serial port is RX(17), TX(16), if you need to print on the Arduino monitor via USB, you need to select Enable)

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents

>

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

>

The screenshot shows the Arduino IDE interface with the 'Tools' menu open. The menu items are as follows:

- Auto Format (Ctrl+T)
- Archive Sketch
- Fix Encoding & Reload
- Manage Libraries...
- Serial Monitor (Ctrl+Shift+M)
- Serial Plotter (Ctrl+Shift+L)
- WiFi101 / WiFiNINA Firmware Updater
- Board: "DFRobot FireBeetle 2 ESP32-C6" >
- Upload Speed: "921600" >
- USB CDC On Boot: "Enabled" >
- CPU Frequency: "160MHz (WiFi)" >
- Flash Frequency: "80MHz" >
- Flash Mode: "QIO" >
- Flash Size: "4MB (32Mb)" >
- Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" >
- Core Debug Level: "None" >
- Erase All Flash Before Sketch Upload: "Disabled" >
- JTAG Adapter: "Disabled" >
- Port: "COM151 (ESP32H2 Dev Module)" >
- Get Board Info
- Programmer >
- Burn Bootloader

11. Click Port to select the corresponding serial port.

5.2 LED Blinking

The default pin for the onboard LED is pin 15.

Sample Code

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

```
int led = 15;
void setup() {
  pinMode(led,OUTPUT);
}

void loop() {
  digitalWrite(led,HIGH);
  delay(1000);
  digitalWrite(led,LOW);
  delay(1000);
}
```

- Copy the codes above to the code editing box.
- Click the arrow to compile the program and burn it into your development board.

Burning Successful



The image above shows that your codes have been successfully loaded into the board. Then, the onboard LED will start blinking.

- Burning failed? Click here.
(https://wiki.dfrobot.com/SKU_DFR1075_FireBeetle_2_Board_ESP32_C6#target_9)

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents



Basic Tutorial

The basic tutorial (https://wiki.dfrobot.com/SKU_DFR1075_FireBeetle_2_Board_ESP32_C6_Basic_Tutorial) includes the use of Battery voltage detection, PWM, interrupt, serial port, servo, and SD card.

Advanced Tutorial

The advanced tutorial (https://wiki.dfrobot.com/SKU_DFR1075_FireBeetle_2_Board_ESP32_C6_Advanced_Tutorial) demonstrates how to use screen, Bluetooth, WiFi, ESP-NOW, one-key for networking config and sample projects.

Tutorial for MicroPython

Build Environment for MicroPython

To run microPython on the FireBeetle 2 ESP32-C6, you need to burn the firmware into FireBeetle 2 ESP32-C6 first.

1. Click to download microPython firmware.

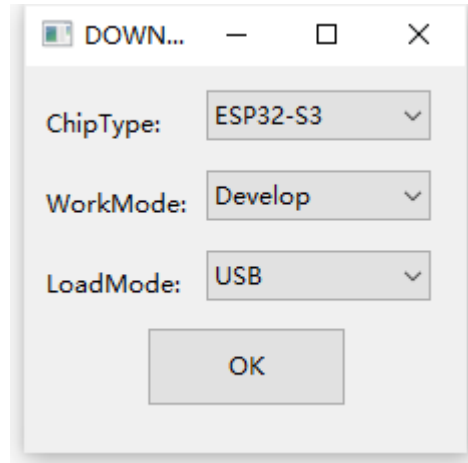
- MicroPython official firmware (<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/2ad9b053ae696dbcd307dd27a8acb0b5.zip>) (by HonestQiao)

2. Click to download esptool Flash burning tool

(https://www.espressif.com/sites/default/files/tools/flash_download_tool_3.9.6_1.zip)

3.Run flash_download_tool.exe

4.Select the ESP32-C6 main controller, than Press Boot, press RST and release both, then try burning again.



5.Select the downloaded firmware, clear the flash and then burn the firmware.

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

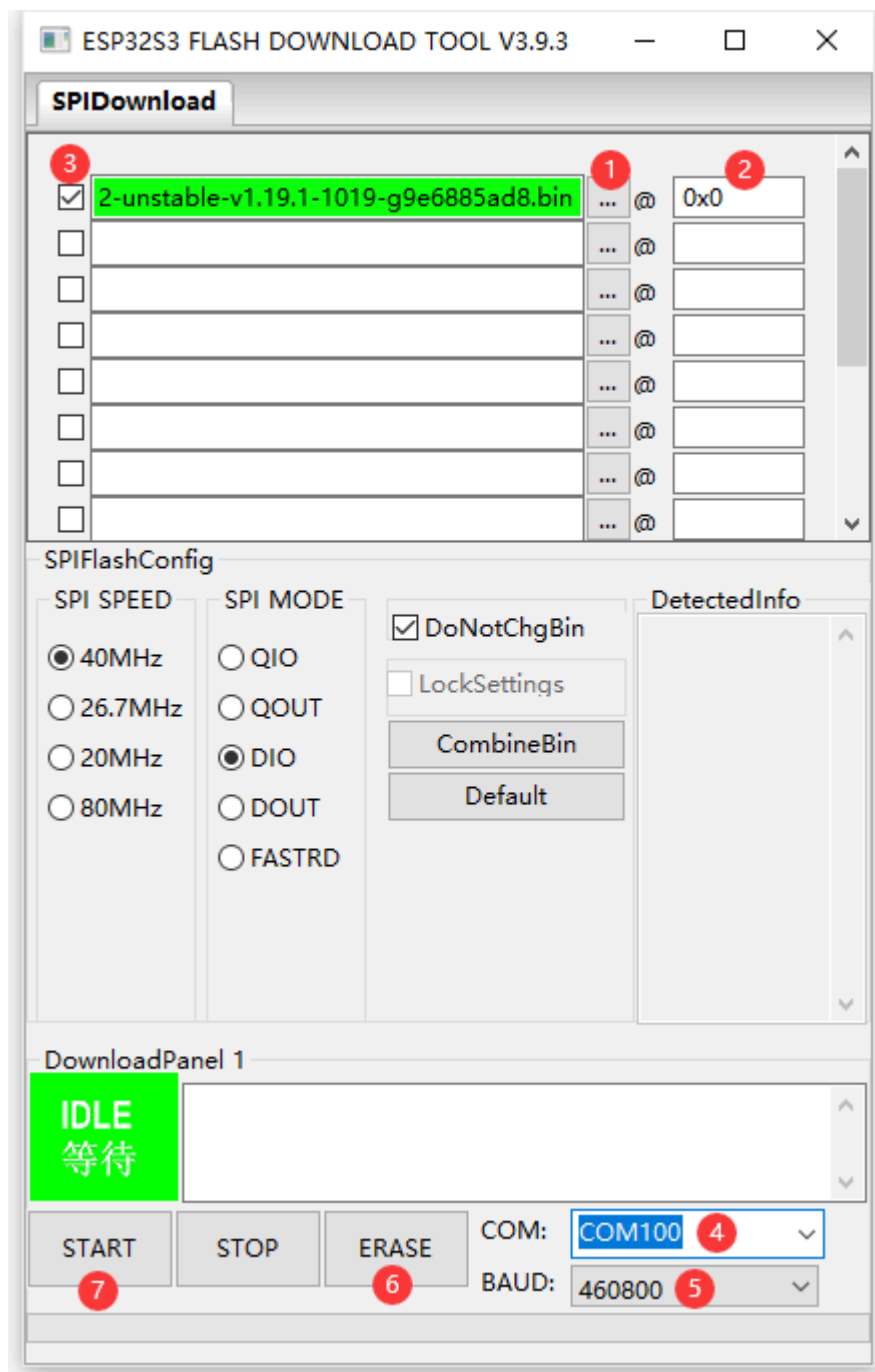
FAQ

More Documents

>

- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

>



MicroPython Interpreter

1. Click to download thonny (<https://thonny.org/>)

2. Open the software and set up the interpreter (Run->Configure interpreter...)

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

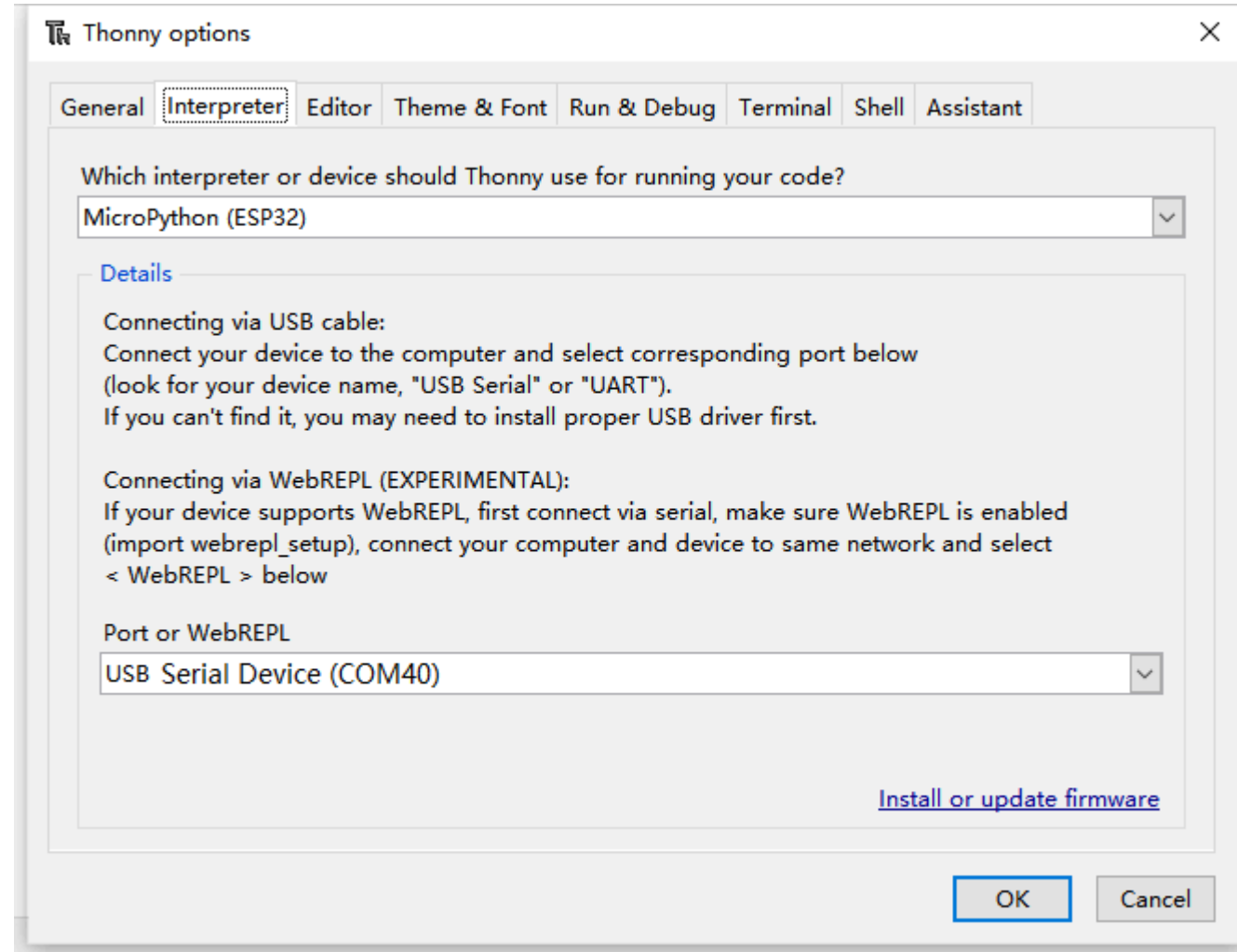
Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents

>



3. Copy and paste the code into the code box, save the file to the MicroPython device and name it main.py

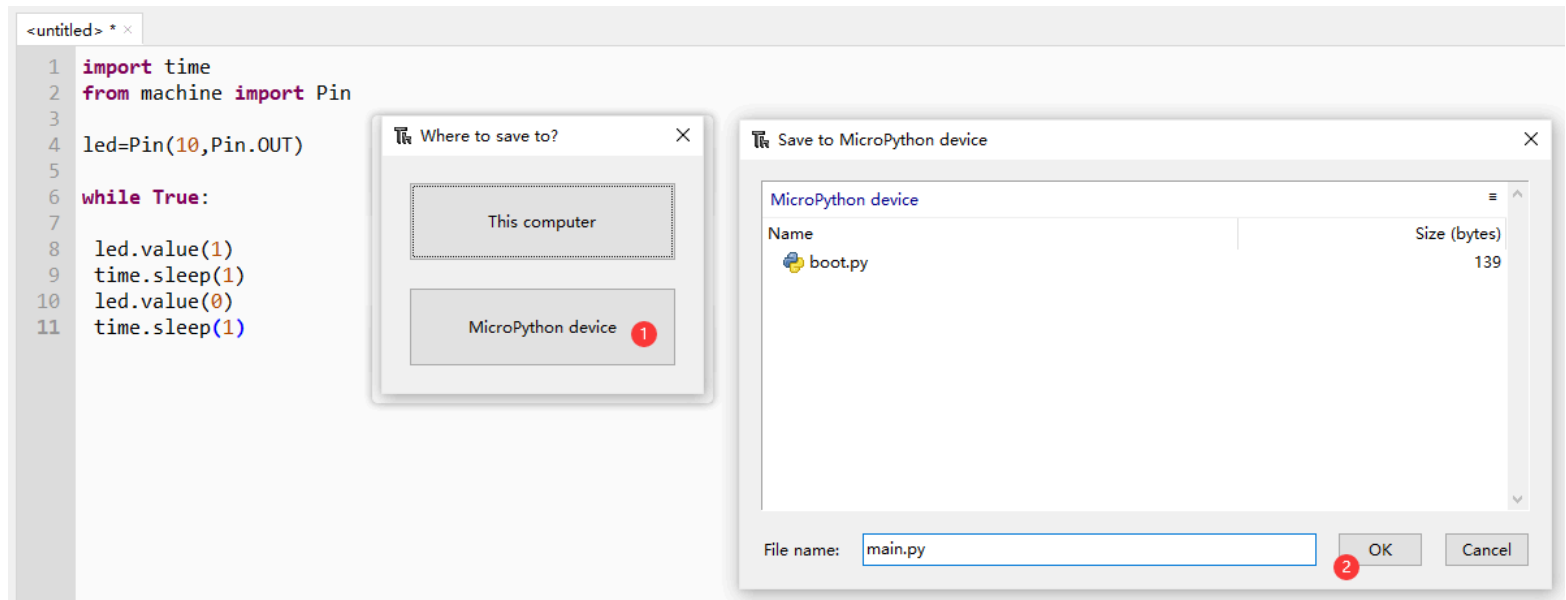
- Introduction
- Features
- Specification
- Board Overview
- Pin Diagram
- Tutorial – First Time Use
- Basic Tutorial
- Advanced Tutorial
- Tutorial for MicroPython
- FAQ
- More Documents

```
import time
from machine import Pin

led=Pin(15,Pin.OUT)

while True:

    led.value(1)
    time.sleep(1)
    led.value(0)
    time.sleep(1)
```



4.Reboot the FireBeetle 2 ESP32-C6, and you can see the L LED flashing.

FAQ

1. What will cause burning error?

- There is no delay or too short delay in Loop.

```
A fatal error occurred: Timed out waiting for packet header
A fatal error occurred: Timed out waiting for packet header
```

Introduction

Features

Specification

Board Overview

Pin Diagram

Tutorial – First Time Use

Basic Tutorial

Advanced Tutorial

Tutorial for MicroPython

FAQ

More Documents

- The USB cannot be recognized by the PC as some functions are incorrectly called.



How to solve

- Press and hold BOOT, click RST, and then release the BOOT button to burn.

Principle During the initialization process, ESP32 undertakes a verification of the voltage level on the BOOT (IO9) pin. If the voltage level is determined to be high, the system proceeds with a normal startup. In contrast, if the voltage level is deemed to be low, the device enters into the programming mode. By default, the BOOT pin maintains a high voltage level, but it transitions to a low level when a button is pressed.

2. Data cannot be printed on serial port

- Check if the USB CDC is enabled
- Check print information using other serial debugger.

For any questions, advice or cool ideas to share, please visit the **DFRobot Forum** (<https://www.dfrobot.com/forum/>).

Introduction
Features
Specification
Board Overview
Pin Diagram
Tutorial – First Time Use
Basic Tutorial
Advanced Tutorial
Tutorial for MicroPython
FAQ
More Documents



More Documents

- ESP32-C6 Chip Datasheet
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/5eefd974782b6cc442f7a2ee7f00577.pdf>)
- DFR1075-Schematics.pdf
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/65df25004a7d1e8bc128894c75ce4089.pdf>)
- DFR1075-Dimension.pdf
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/fc5aabe8620a8c9519643a7e8c997ecb.pdf>)
- CN3165 Chip Datasheet
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/85759bb076bf6bb24fd8ca5683f19603.pdf>)
- HM6245 Chip Datasheet
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/6f630301d84caf0e92266e3c5cf11edc.PDF>)
- FireBeetle2 Shell stl
(<https://dfimg.dfrobot.com/5d57611a3416442fa39bffca/wiki/e98b6467f3636d6d33c8d894b29bad89.zip>)

 Get **FireBeetle 2 Board ESP32-C6** (<https://www.dfrobot.com/product-2771.html>) from DFRobot Store or **DFRobot Distributor**. (<https://www.dfrobot.com/distributor>)

Turn to the Top