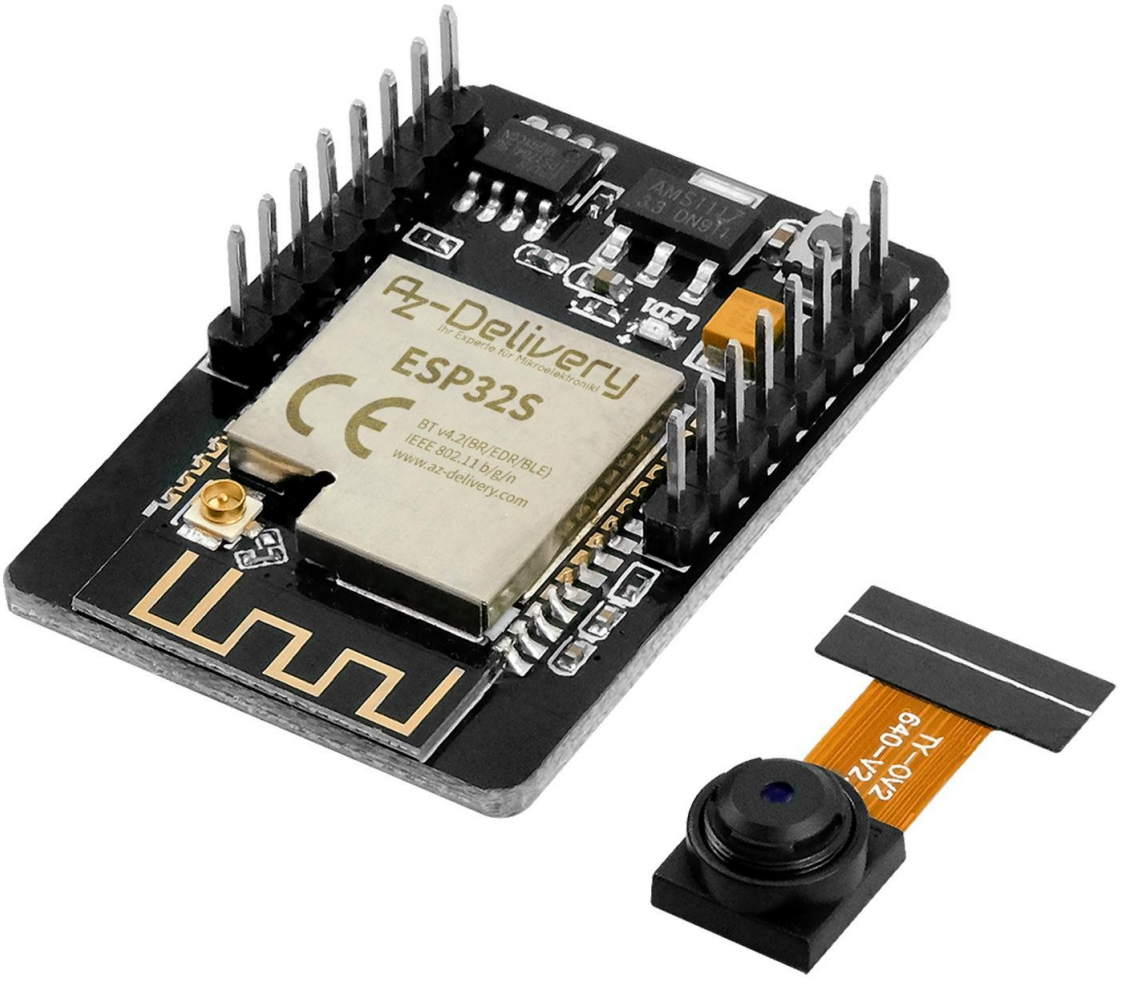
## ebook

# ESP32-Cam

## AI-Thinker Model



# Table of contents

# Introduction

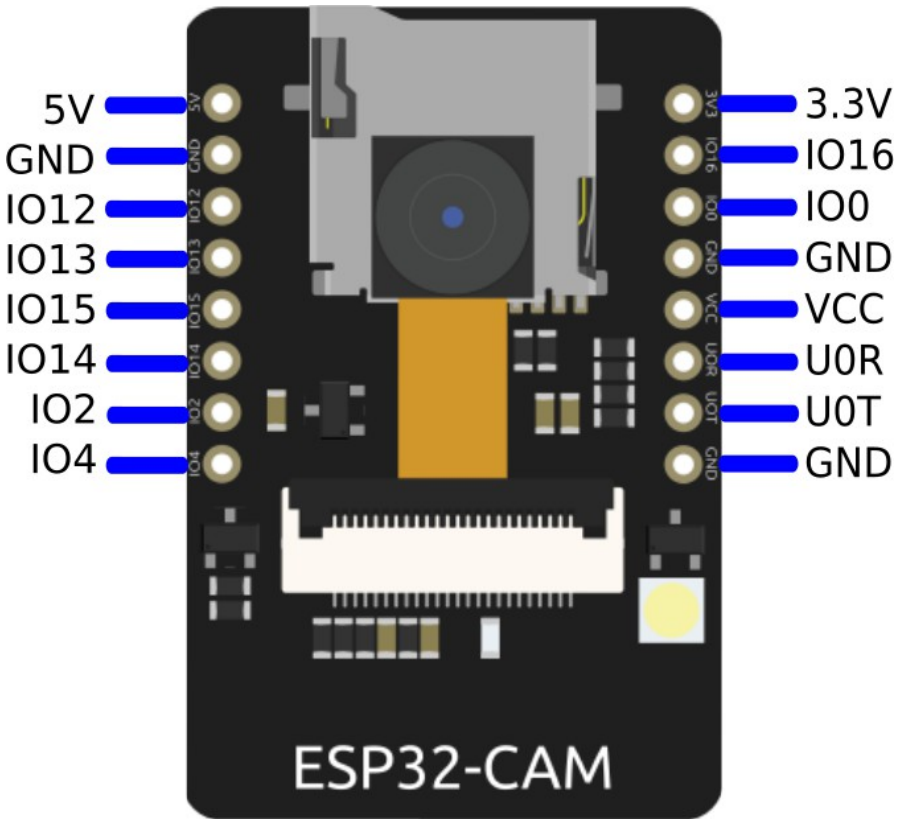
The ESP32-Cam is a very competitive, small camera module that can operate independently as a minimal system with a size of only 27 x 40.5 x 4.2 mm.

It can be used in various IoT applications. It is suitable for smart home devices, industrial wireless control, wireless monitoring, wireless QR identification and other IoT applications.

# Specifications

| | |
|---|---|
| SPI Flash | 4 MB |
| RAM | 520KB SRAM + 4M PSRAM |
| Bluetooth | Bluetooth 4.2 BR/EDR and BLE standards |
| Wi-Fi | 802.11 b/g/n |
| Support interface | UART, SPI, I2C, PWM |
| Support TF card | Max. 4GB |
| IO port | 9 |
| UART baud rate | Default 115200 bps |
| Image Output Format | JPEG (OV2640 support only), BMP, GRAYSCALE |
| Spectrum Range | 2412~2484 MHz |
| Antenna | Onboard PCB antenna, gain 2dBi |
| Transmit Power | 802.11b: 17±2 dBm (@11Mbps)<br>802.11g: 14±2 dBm (@54Mbps)<br>802.11n: 13±2 dBm (@MCS7) |
| Receiving Sensitivity | CCK, 1 Mbps: -90 dBm<br>CCK, 11 Mbps: -85 dBm<br>6 Mbps (½ BPSK): -88 dBm<br>54 Mbps(¾ 6-QAM): -70 dBm<br>MCS7(65 Mbps, 72.2 Mbps): -67 dBm |
| Power Supply Range | 5V |
| Operating Temperature | -20°C ~ 85°C |
| Dimensions | 27 x 40.5 x 4.5 mm |

# Pinout




**Note:**

The module can be supplied either via the 3.3V pin or via the 5V pin.

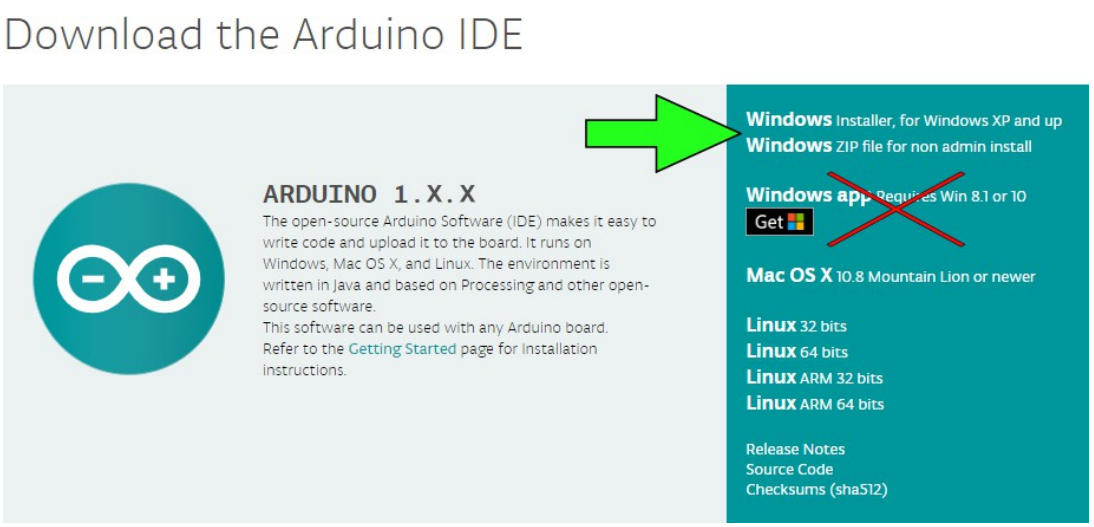
It is recommended to operate the module via the 5V pin (with 5V).

**The VCC Pin is a 3.3V output.**

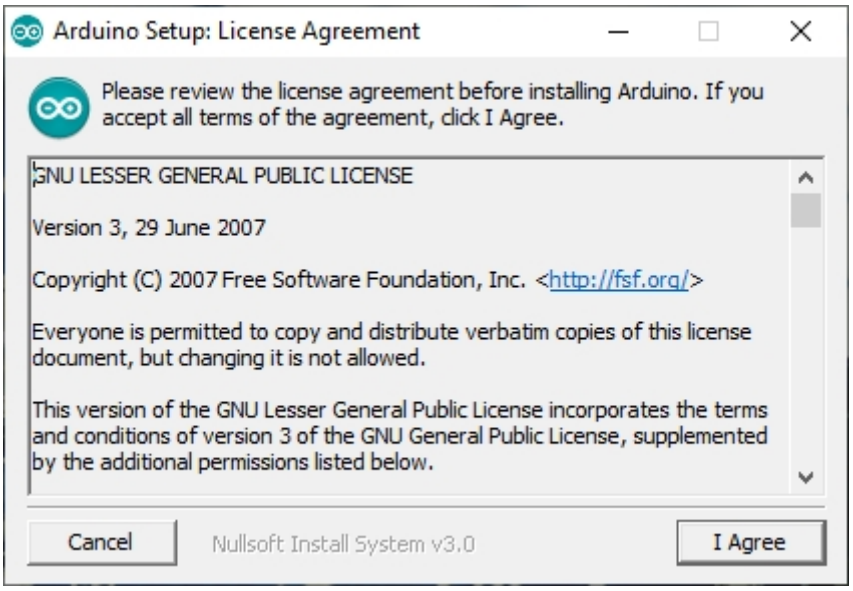
# Installation of the Arduino IDE

You can download the free Arduino IDE development environment from the following link: https://www.arduino.cc/en/Main/Software

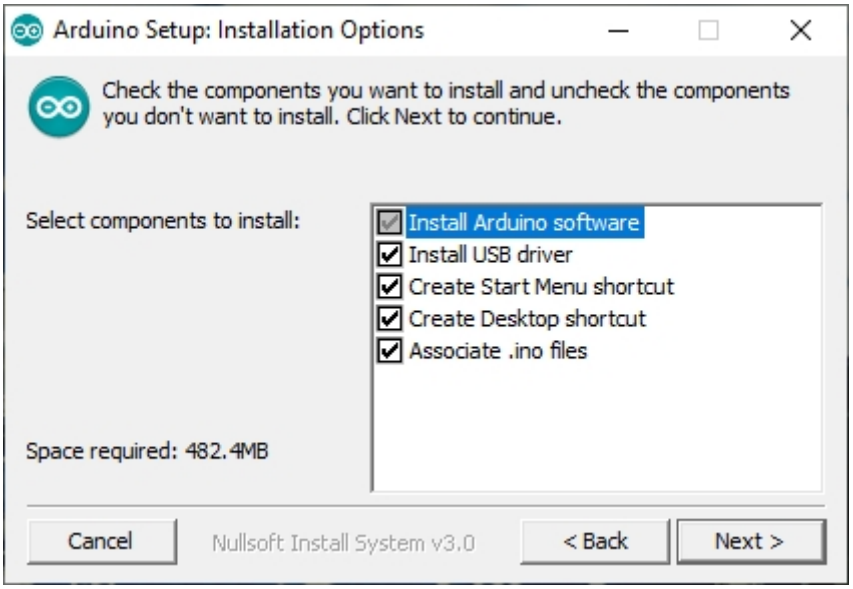
Windows users should definitely use one of the first two
Use the download options for the Arduino IDE. The "Windows App" version from the Windows Store leads to connection problems, especially when using third-party board definitions.




After starting the Arduino IDE installation file "arduino-1.X.X-windows.exe", the licence conditions of the software must be read and accepted:



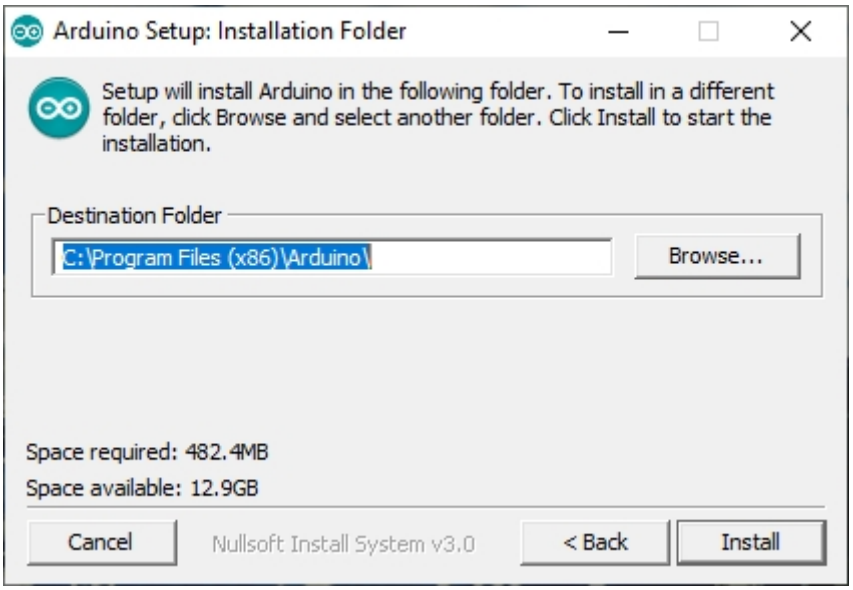

In the next step, different options can be selected for installation.



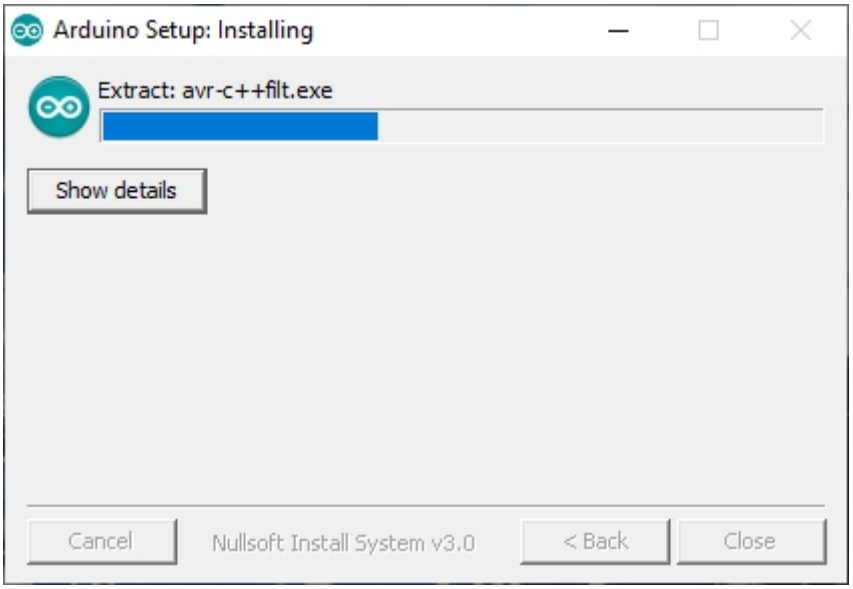

The following is a brief overview of the various options, with a short explanation of each option:

| Option | Explanation |
|---|---|
| Install Arduino software | Installs the Arduino IDE - This option cannot be deselected |
| Install USB Driver | Installs USB drivers for various other microcontrollers. These are not required to use the software with the D1 mini, but we strongly recommend installing them if you also use other microcontrollers |
| Create Start Menu shortcut | Creates a shortcut in the Windows Start menu (optional) |
| Create Desktop shortcut | Creates a shortcut on the workstation (optional) |
| Associate .ino files | Creates a file name extension for files with the extension .ino and links them to the Arduino IDE |

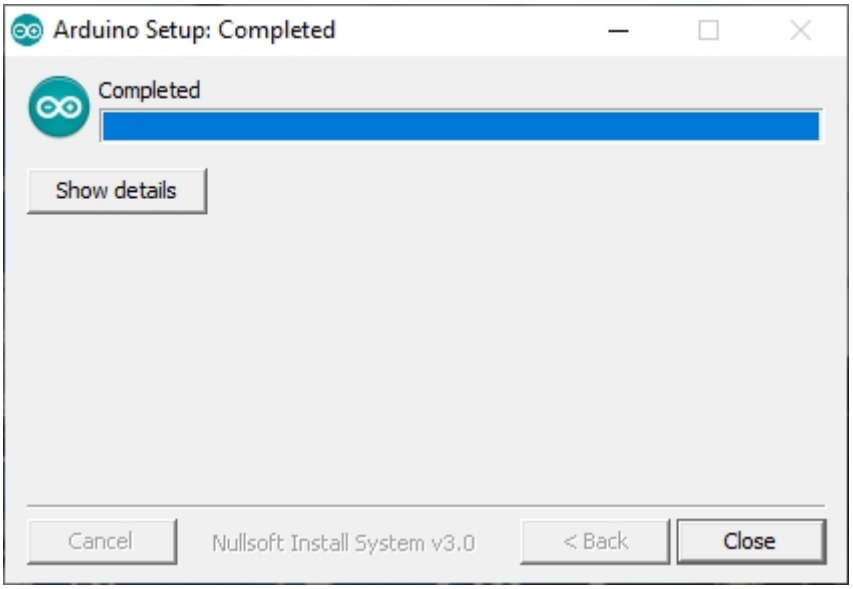
Finally, the destination folder must be specified. The installation requires approx. 500MB of free disc space.




Click on "Install" to start the installation.



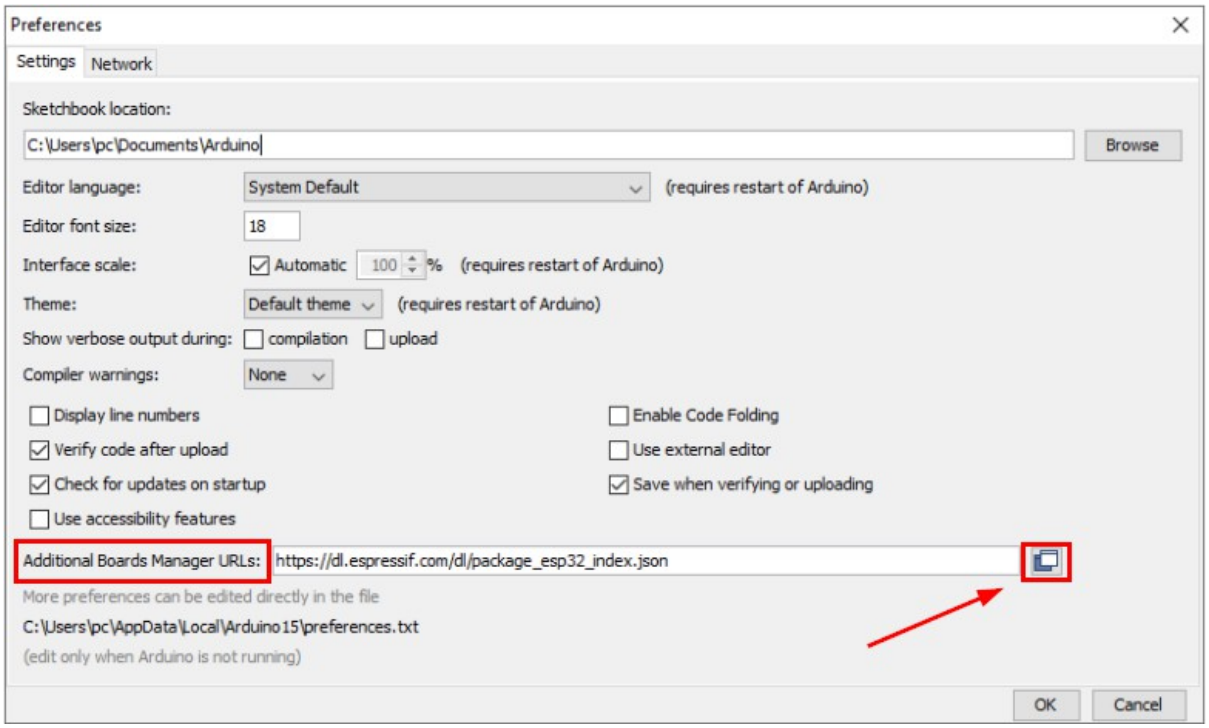

After successful installation, the installation programme can be closed using the "Close" button:



# Additional settings

To be able to programme the ESP32-Cam in the Arduino IDE, you must install the support for the ESP32 platform.
Open the Arduino IDE and go to: File > Preferences, and find the "Additional URLs" field.

Copy the following link: https://dl.espressif.com/dl/package_esp32_index.json




Insert this link in the "Additional URLs" field. If you already have one or more links in this field, simply insert a comma after the last link, insert the new link after the comma and click the OK button. Then close the Arduino IDE.

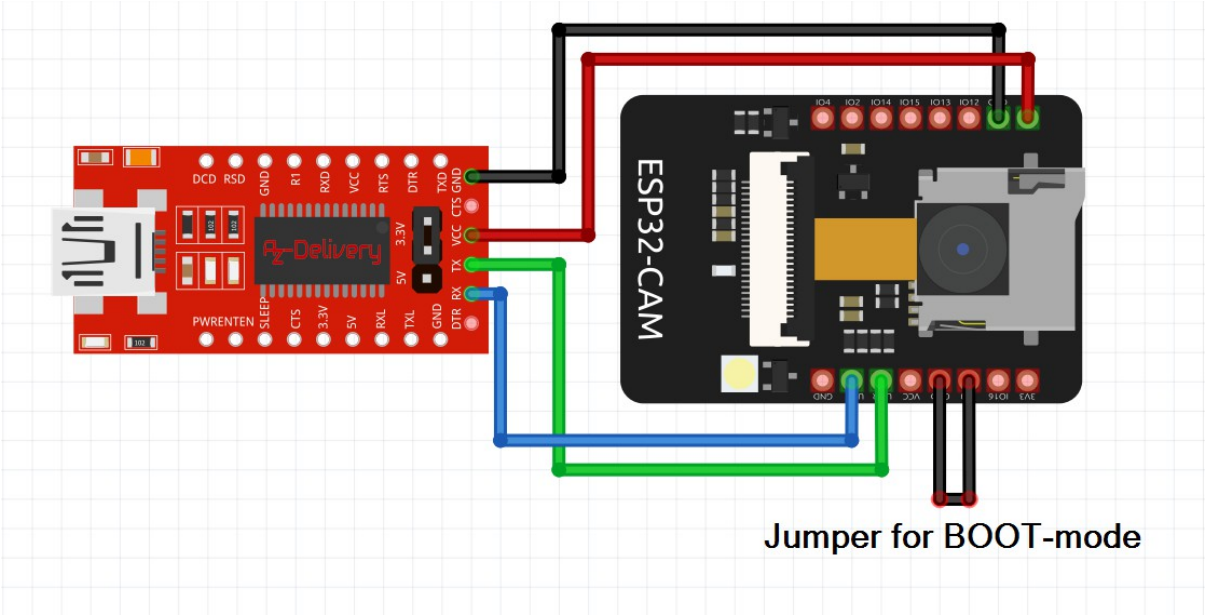
Open the Arduino IDE again and go to :
Tools > Board > Boards Manager
A new window will open, enter esp32 in the search box and install the board named esp32 from Espressif Systems as shown below:


Now you can select Tools > Board > ESP32 Arduino > AI-Thinker ESP32-CAM.

# Connection diagram

To be able to flash the ESP32-Cam you need a USB-to-UART adapter, make sure that the necessary driver is installed.

The following figure shows the connection of the ESP32-Cam with the AZ-Delivery FT232 USB-to-UART adapter (another adapter can also be used). The jumper for the **logic level** must be set to **3.3V**, otherwise the pins on the ESP module will be damaged.




| FT232 | ESP32-Cam |
|---|---|
| VCC | 5V |
| GND | GND |
| TX | U0R |
| RX | U0T |

To set the ESP32-Cam to BOOT mode, a jumper is required between IO0 and GND.

**Note:** The ESP32-Cam has no reverse polarity protection, make sure not to swap 5V and GND, otherwise the module will be destroyed.

# Blink Example

Blink is a good first example. There is a flash-light LED on the ESP32 cam, which is connected to GPIO 4.

```
#define flash 4
void setup() {
  pinMode(flash, OUTPUT);
}

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

Under Tools > Board > ESP32 Arduino > AI-Thinker ESP32-CAM, select the port of the adapter. Then click on "Upload".

If the upload was successful, you can remove the jumper between IO0 and GND and then press the RESET button. Be warned, **the flash-light LED is very bright.**

# Camera web server Example

With this example you can display a video stream on the web interface. To open the example, go to
File > examples > ESP32 > camera > CameraWebServer

From line 10 you should select the camera model. The AI-Thinker model must be selected here. Adjust the lines, it should look like this:

```
// Select camera model
//#define CAMERA_MODEL_WROVER_KIT // Has PSRAM
//#define CAMERA_MODEL_ESP_EYE // Has PSRAM
//#define CAMERA_MODEL_M5STACK_PSRAM // Has PSRAM
//#define CAMERA_MODEL_M5STACK_V2_PSRAM // M5Camera
//#define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
//#define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
//#define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM #define
CAMERA_MODEL_AI_THINKER // Has PSRAM
//#define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
```

In lines 23 and 24 you must enter the SSID and password of your router. e.g:

```
const char* ssid = "here-SSID";
const char* password = "Here password";
```

This example can then be uploaded. As soon as the upload was successful, remove the jumper between IO0 and GND and then press the reset button. Open the serial monitor of the Arduino IDE and you should see the IP address under which the
ESP32-Cam can be reached.

Enter the IP address of the ESP32-Cam in your browser (you must be in the same network). You are now on the web interface of the ESP32-Cam. There are various setting options with which you can play. At the bottom there is the button "Start Stream" with which you can start the video transmission.

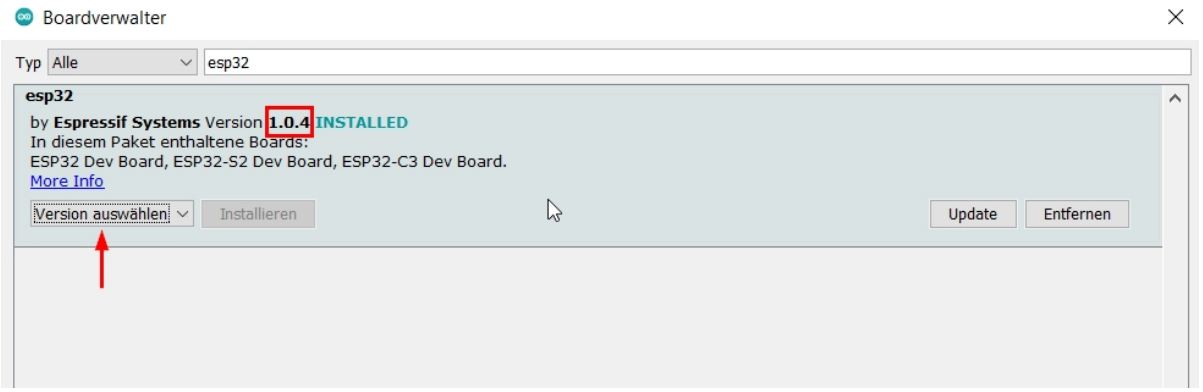
# Face recognition

The previous example also provides the face detection function. When the stream is running, you can activate the "Face detection" option; a yellow rectangle is placed around the face in the video view. A face can be saved using the "Face recognition" option and the "Enroll Face" button.

If you are using the current version of the ESP32 board information (currently 2.0.2), face recognition does not work. This function should be available again in the future.

To still be able to use face recognition, you must downgrade the version (install an older version). You can do this by going to Tools > Board > Board Manager and entering "esp32" in the search bar.

Click on "Select version" and select version 1.0.4 and click on "Install", then restart the Arduino IDE




Once the Arduino IDE has been restarted, you must reopen the example. File > Examples > ESP32 > camera > CameraWebServer

Make the same settings as in the previous example and load it onto the ESP32-Cam.
Face recognition now works.

Now it's time to learn and create your own projects. You can do this with the help of many example scripts and other tutorials that you can find on the Internet.

**If you are looking for high-quality products for Arduino and Raspberry Pi, AZ-Delivery Vertriebs GmbH is the right place for you. You will receive numerous application examples, complete installation instructions, eBooks, libraries and support from our technical experts.**

https://az-delivery.de

Have fun!

Imprint

https://az-delivery.de/pages/about-us