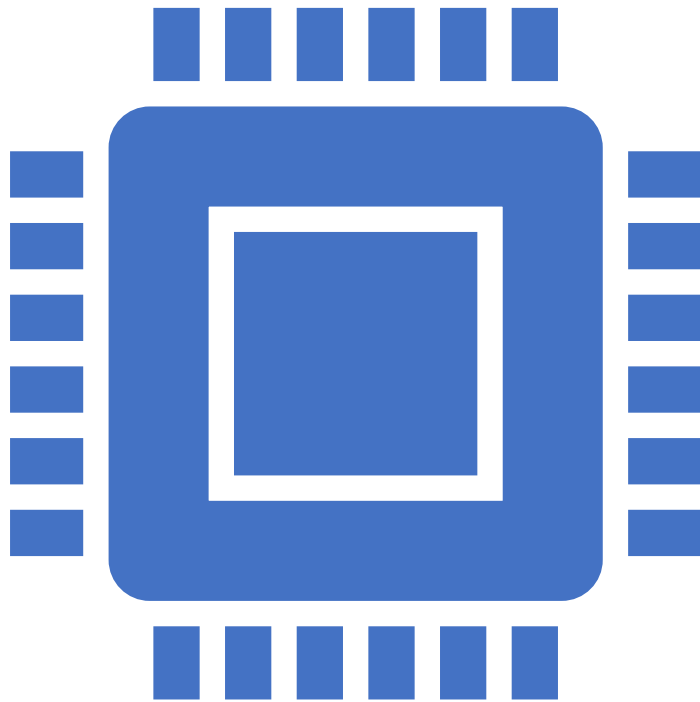
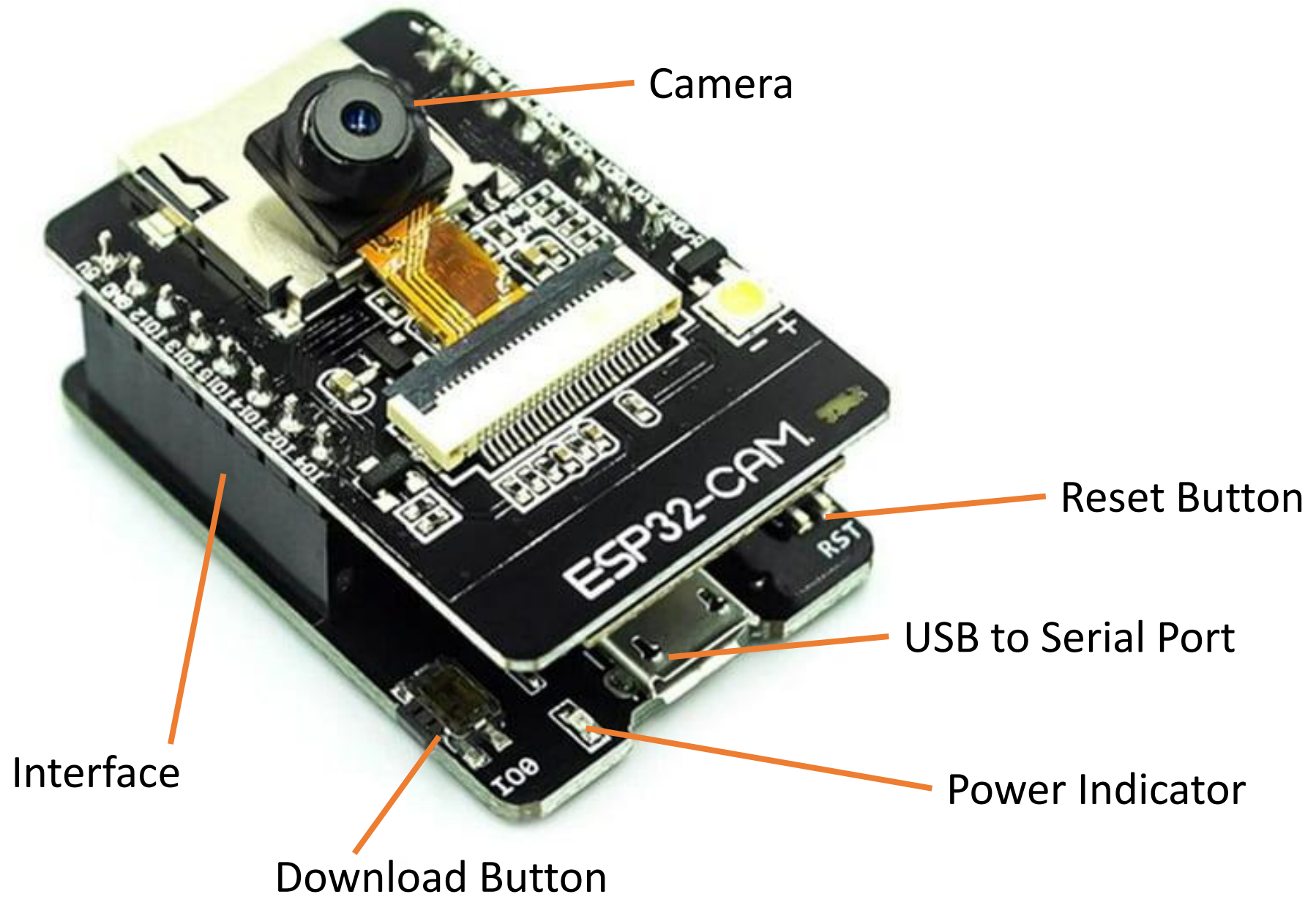




CIS4951

AI ON LOW-COST CAMERA

ESP32-CAM + ARDUINO IDE
USER GUIDE/MANUAL



Downloads



Arduino IDE 2.1.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

Windows Win 10 and newer, 64 bits

Windows MSI installer

Windows ZIP file

Linux AppImage 64 bits (X86-64)

Linux ZIP file 64 bits (X86-64)

macOS Intel, 10.14: "Mojave" or newer, 64 bits

macOS Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

1. Now to the Arduino IDE. To download you must go to <https://www.Arduino.cc/en/software>. From there you'll choose the option you want to download per your operating system.

2. When you click an option you'll be directed to a download page where you'll click one of the two options provided. Once you've clicked an executable file will be downloaded.

Support the Arduino IDE

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **74,662,161** times — impressive! Help its development with a donation.

\$3

\$5

\$10

\$25

\$50

Other

JUST DOWNLOAD

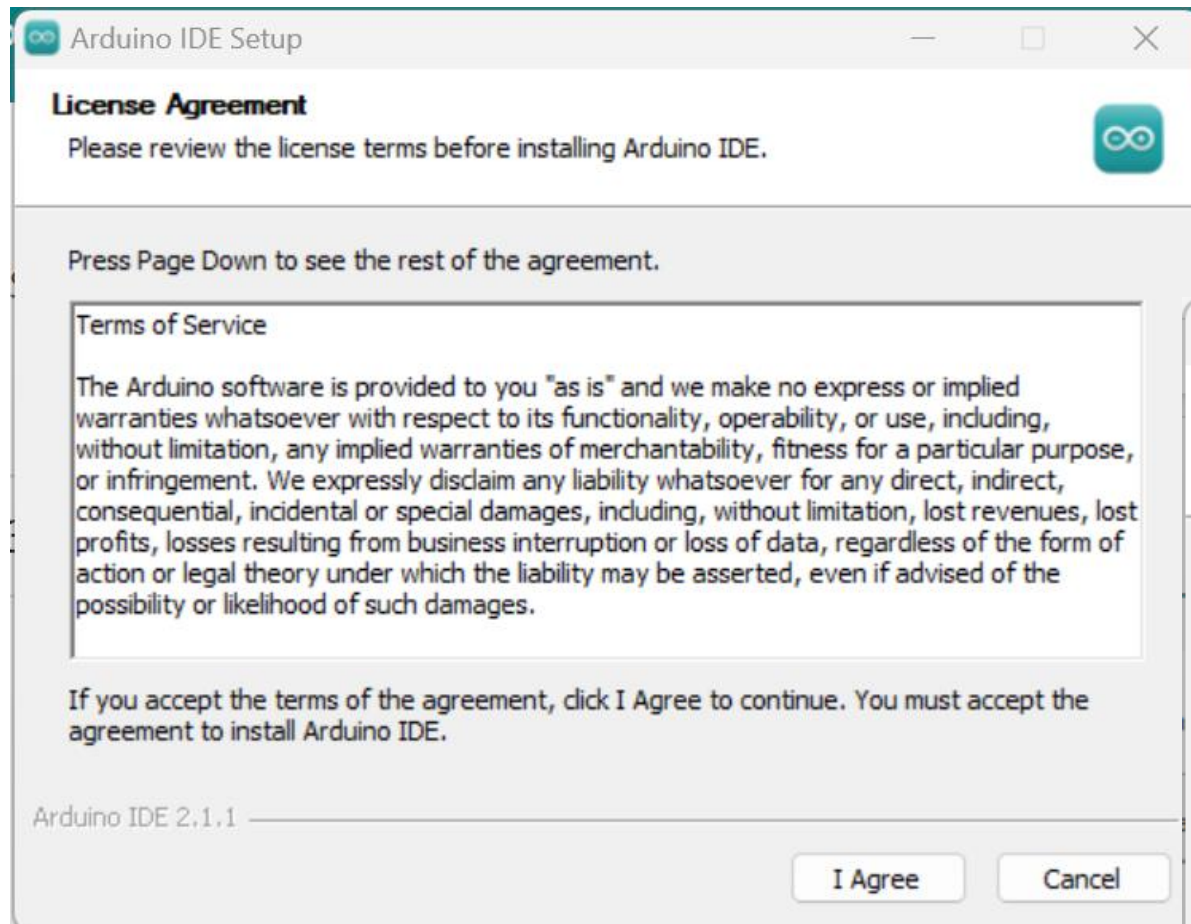
CONTRIBUTE & DOWNLOAD



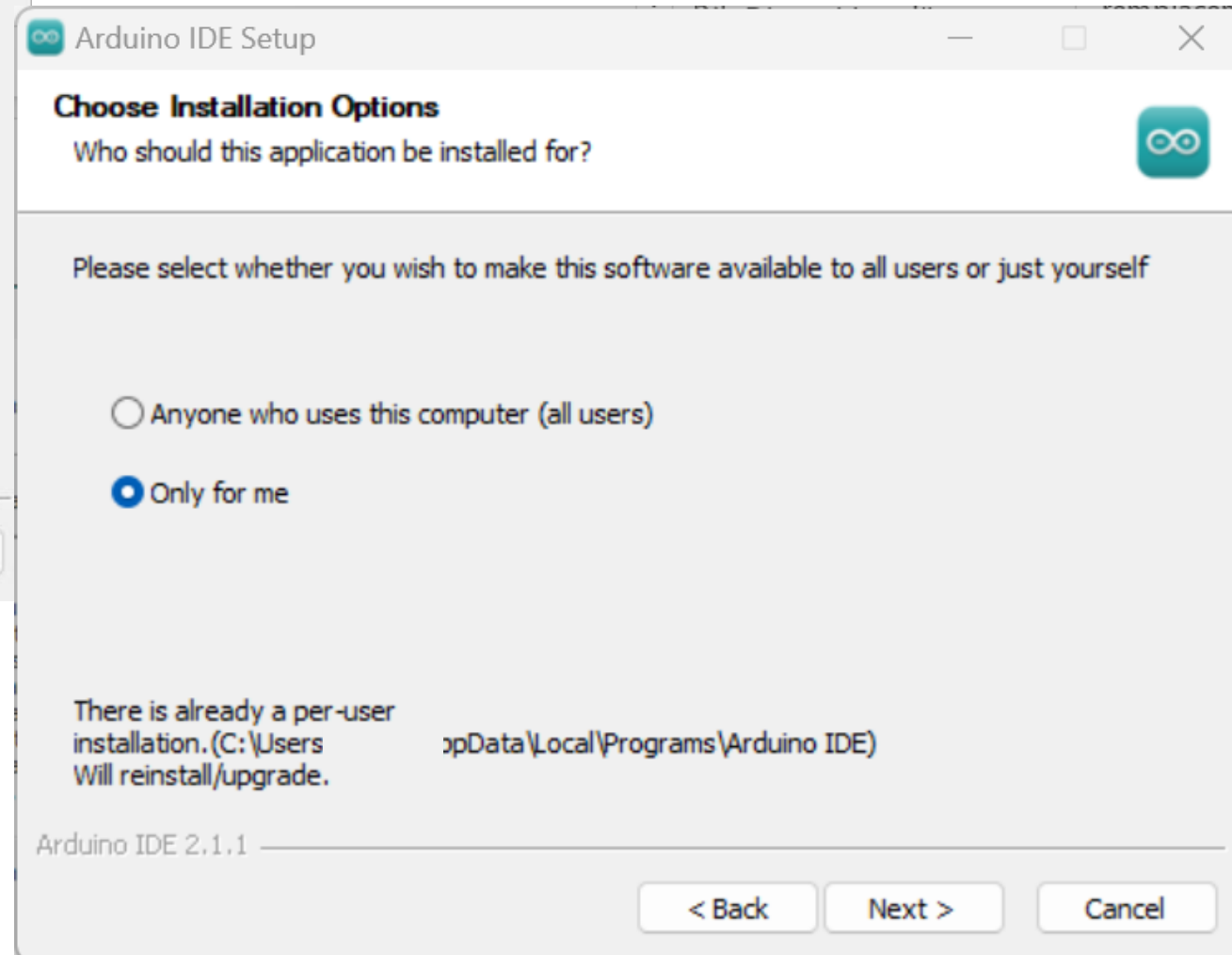
<https://www.arduino.cc/en/software>

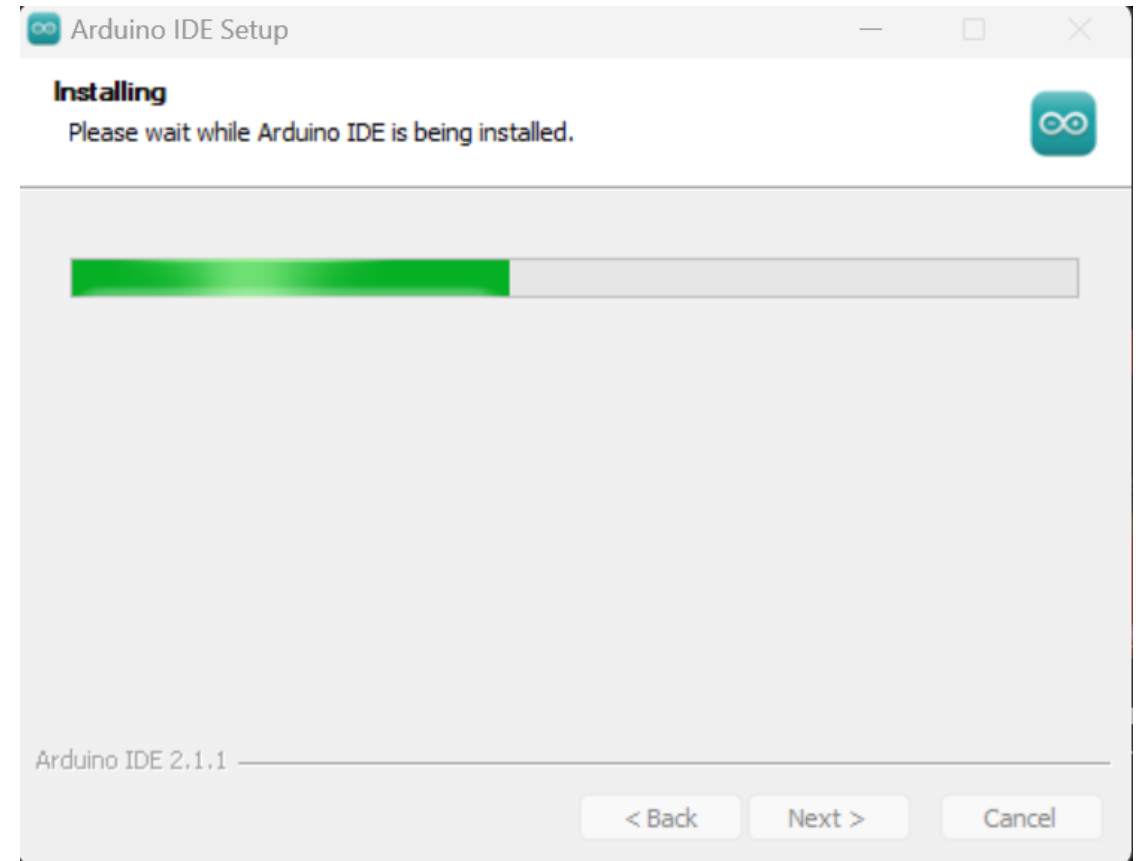
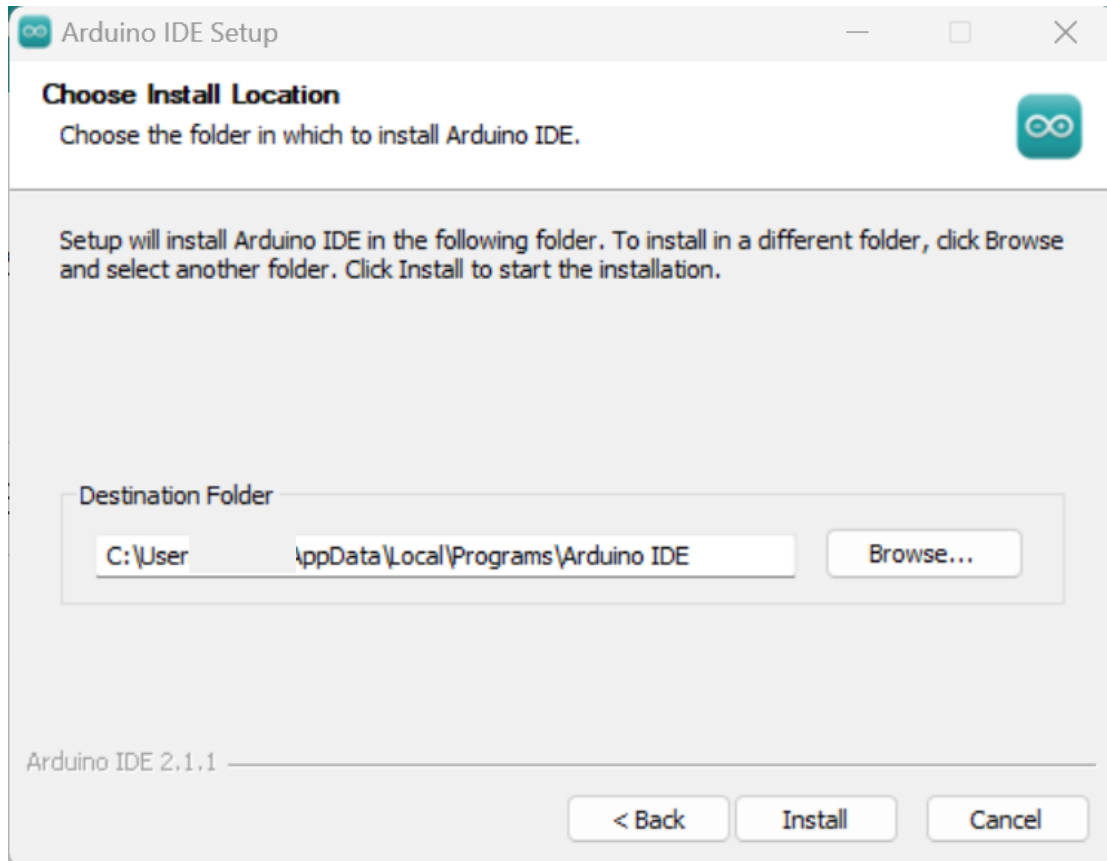
arduino-ide_2.1.1_...exe

60.9/162 MB, 16 secs left



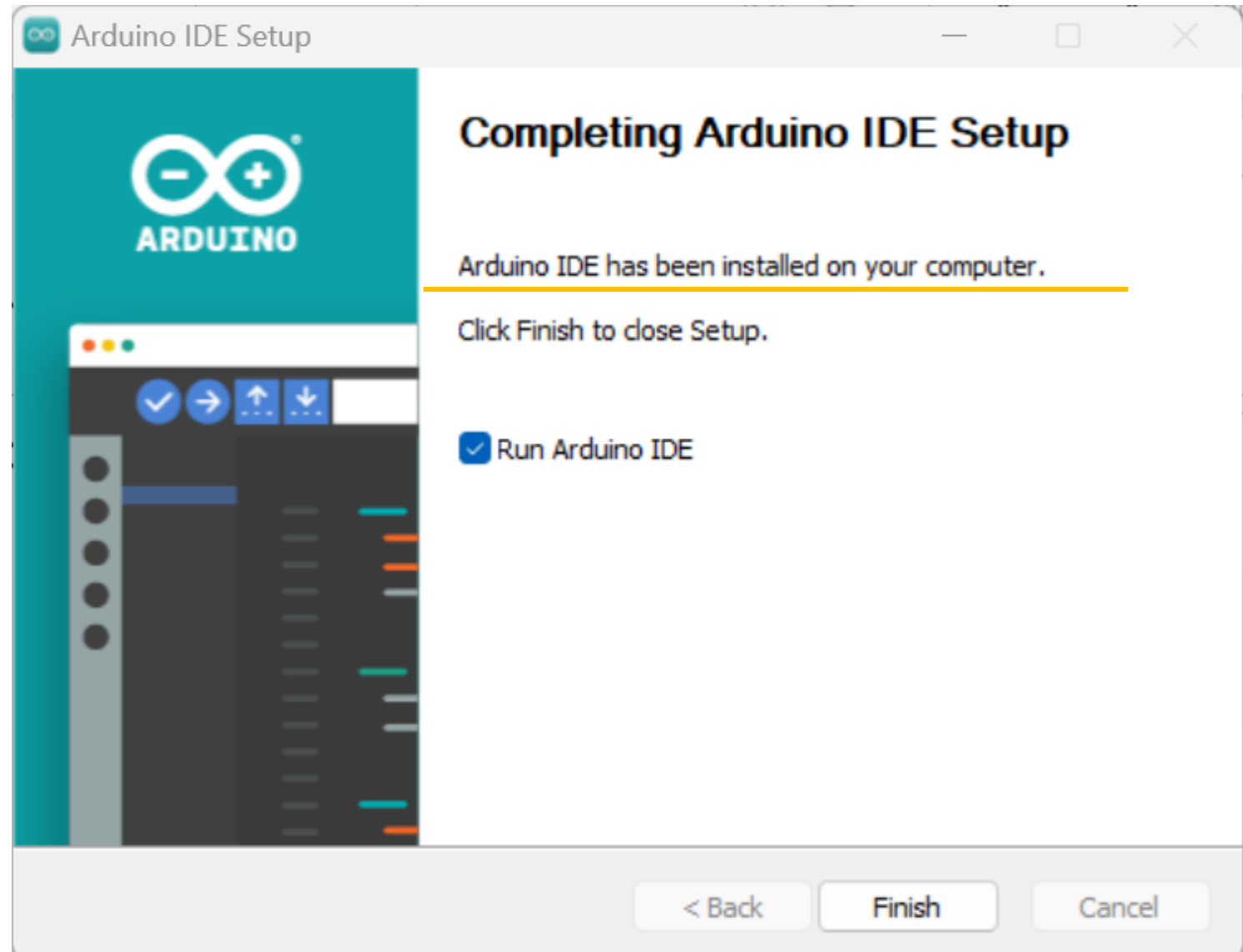
You'll be prompted to read and accept the license agreement. Proceeding that you can choose whether you'll want to install the software for yourself only or for any who uses the computer. It's up to you at this point.





After that you'll be asked to choose a location to install the IDE. It's up to you where you want it to be installed.

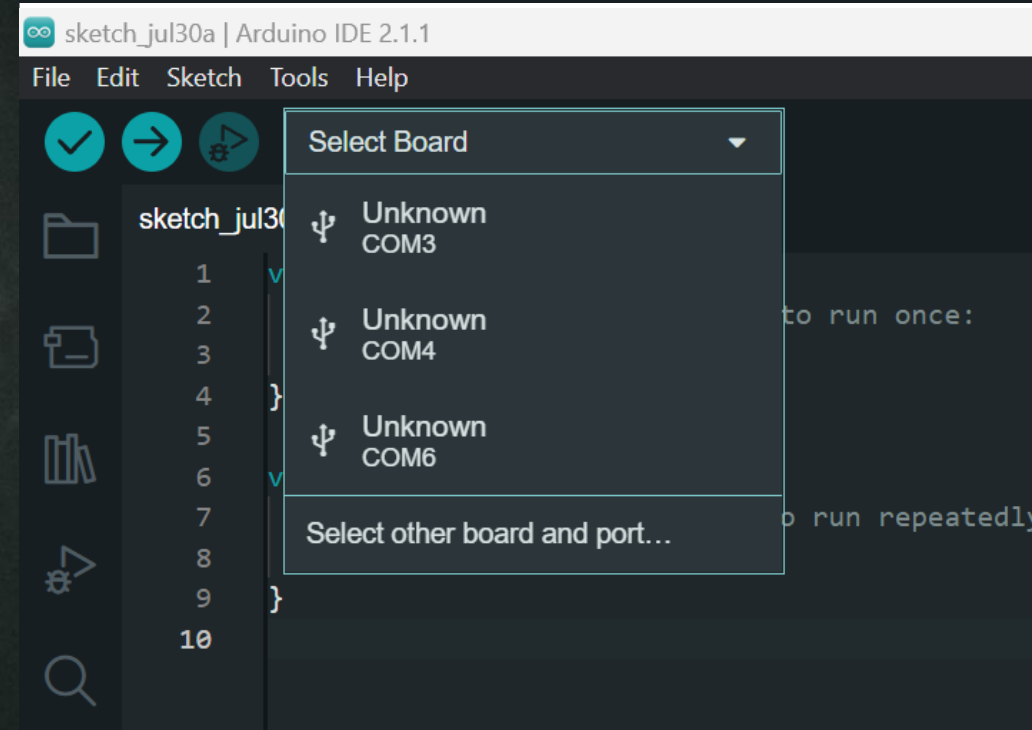
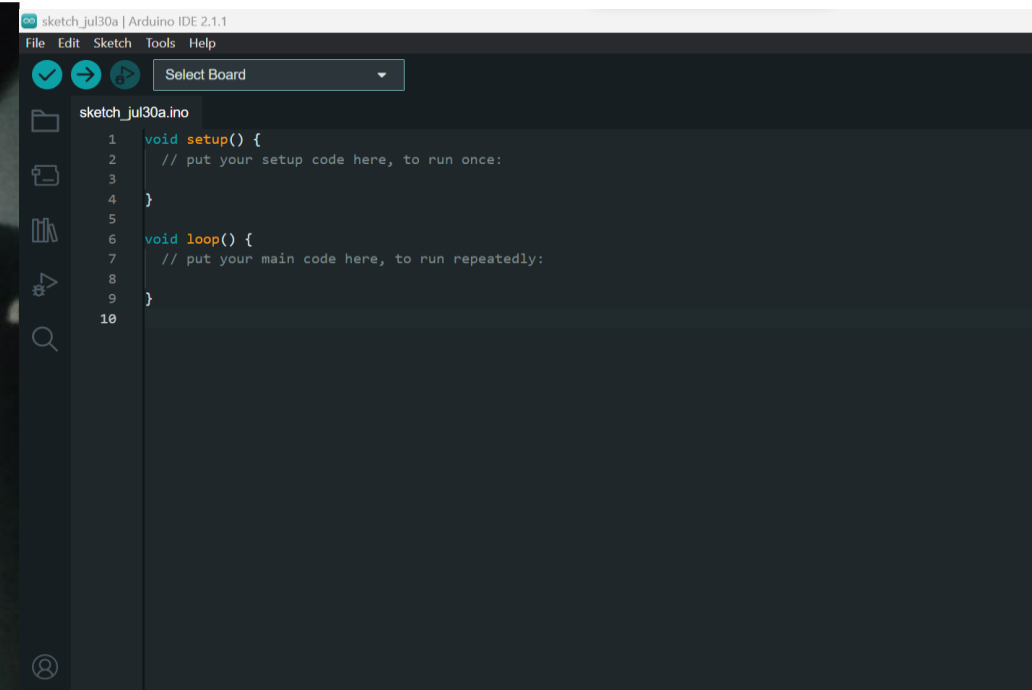
The Arduino IDE installation is now complete. You can choose to run the IDE immediate after clicking finish or you can uncheck the checkbox to run the IDE when you wish.

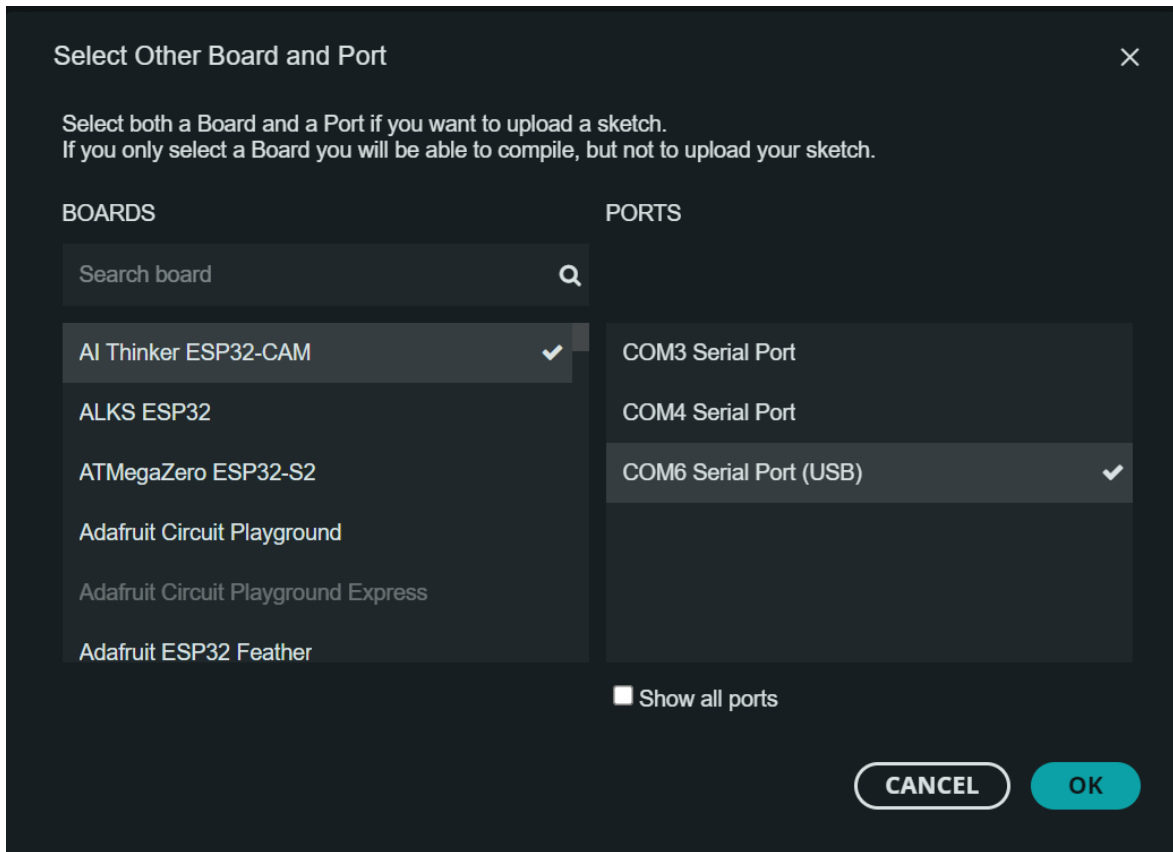




Once the IDE has been opened you must connect the ESP32 CAM to your external computer with the correct port.

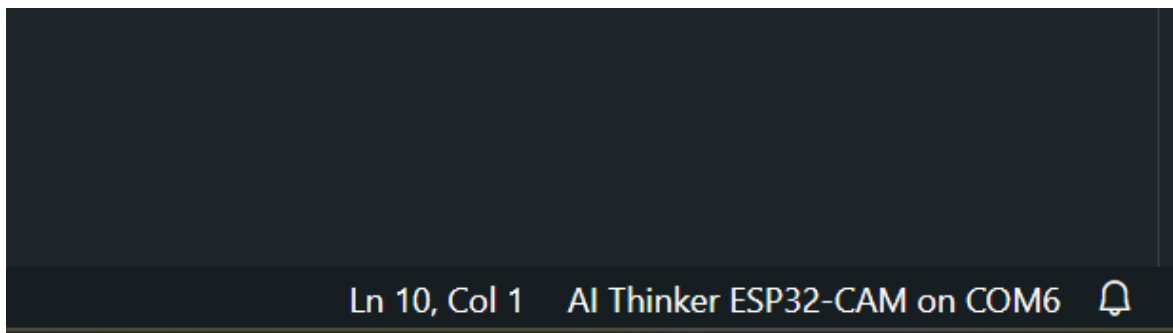
The power indicator will show you that it's receiving power from your computer via the USB

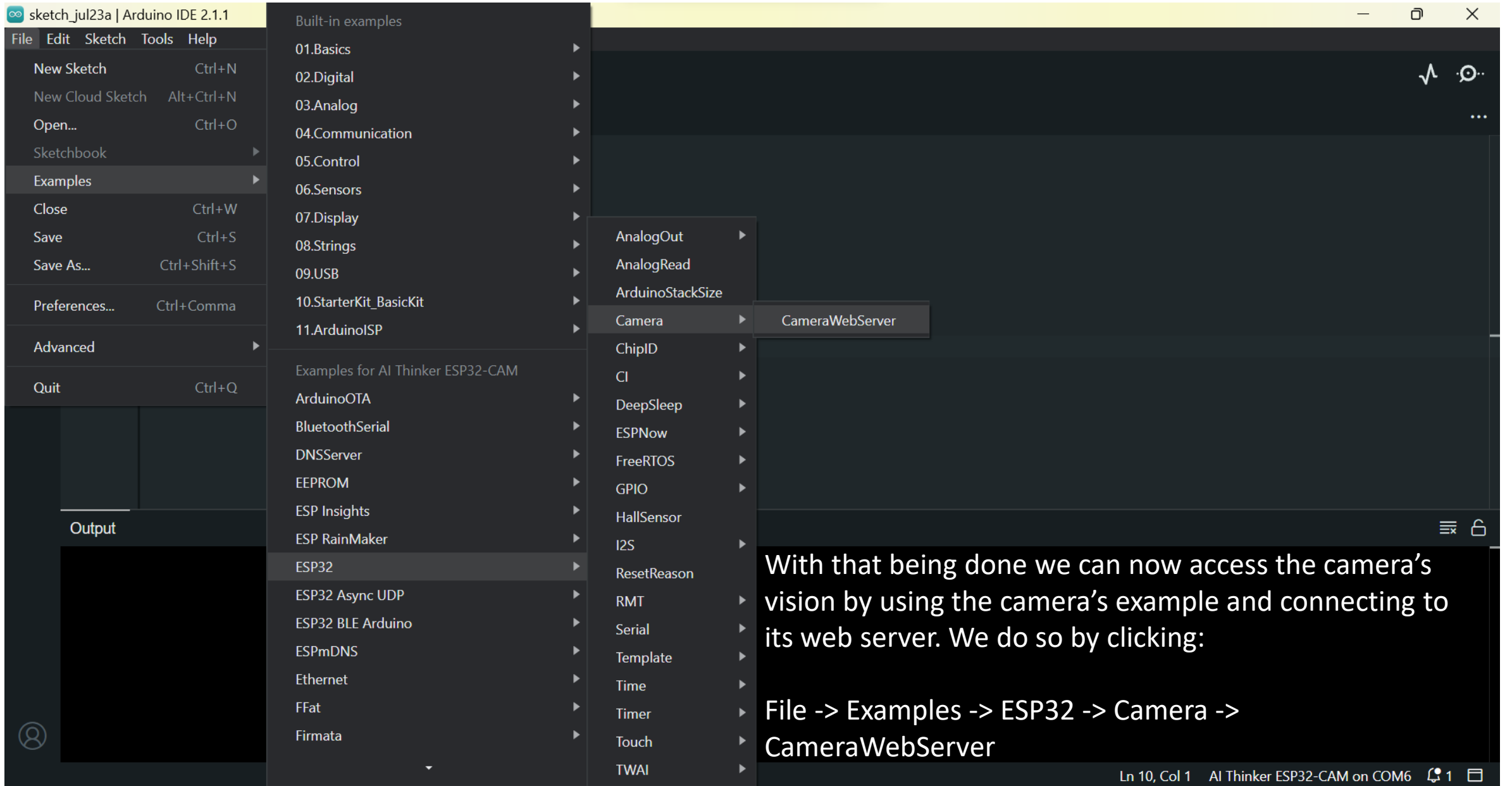


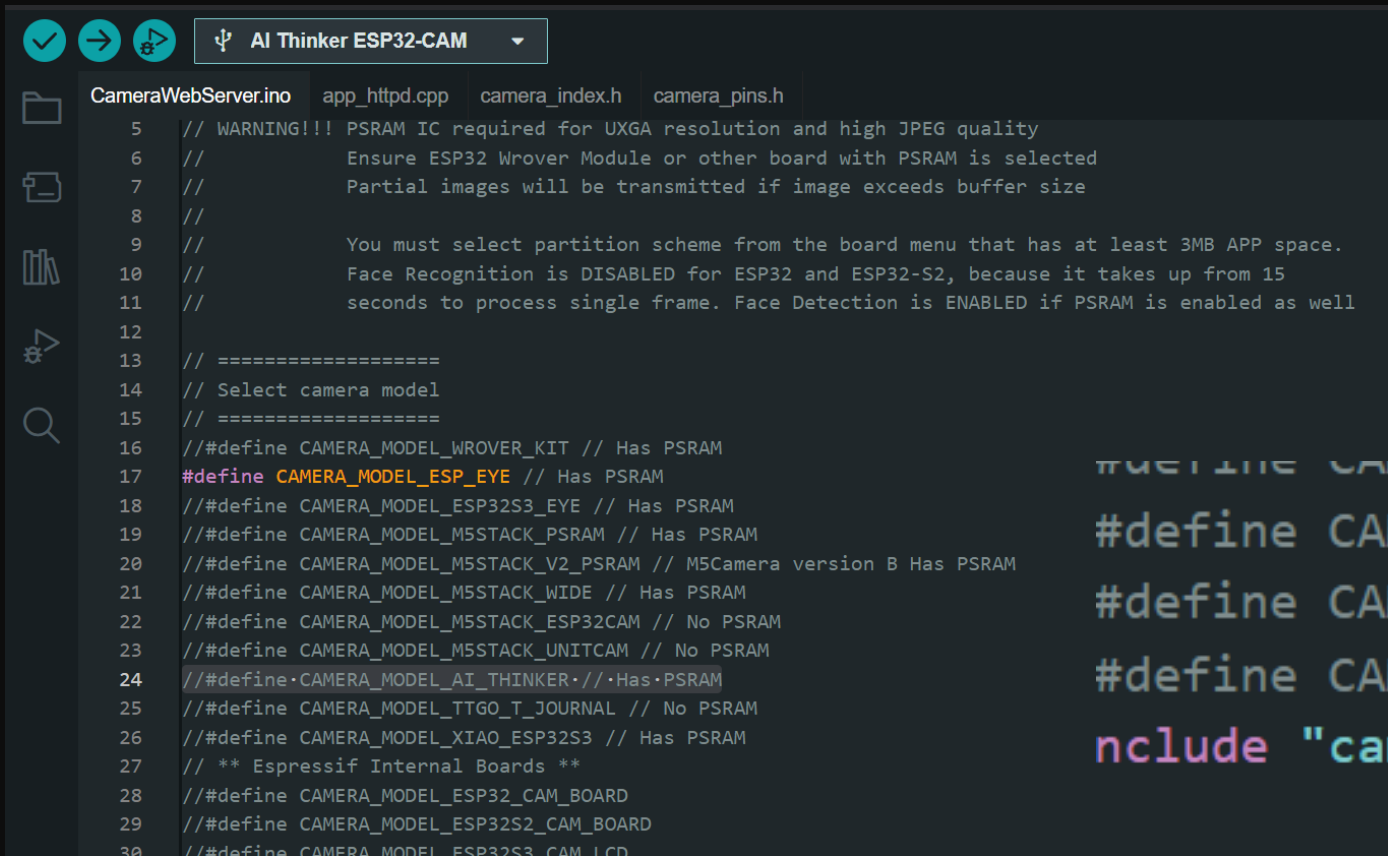


When you click ‘Select other board and port’ you’ll be shown different board and various ports to choose from. Choose the one that fits you ESP32-CAM model and the port you’re connected to.

Once connected to the IDE you should see in the bottom right corner the connected camera and port.







```
CameraWebServer.ino  app_httpd.cpp  camera_index.h  camera_pins.h
5  // WARNING!!! PSRAM IC required for UXGA resolution and high JPEG quality
6  //     Ensure ESP32 Wrover Module or other board with PSRAM is selected
7  //     Partial images will be transmitted if image exceeds buffer size
8  //
9  //     You must select partition scheme from the board menu that has at least 3MB APP space.
10 //     Face Recognition is DISABLED for ESP32 and ESP32-S2, because it takes up from 15
11 //     seconds to process single frame. Face Detection is ENABLED if PSRAM is enabled as well
12 //
13 // =====
14 // Select camera model
15 // =====
16 // #define CAMERA_MODEL_WROVER_KIT // Has PSRAM
17 #define CAMERA_MODEL_ESP_EYE // Has PSRAM
18 // #define CAMERA_MODEL_ESP32S3_EYE // Has PSRAM
19 // #define CAMERA_MODEL_M5STACK_PSRAM // Has PSRAM
20 // #define CAMERA_MODEL_M5STACK_V2_PSRAM // M5Camera version B Has PSRAM
21 // #define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
22 // #define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
23 // #define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM
24 // #define CAMERA_MODEL_AI_THINKER // Has PSRAM
25 // #define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
26 // #define CAMERA_MODEL_XIAO_ESP32S3 // Has PSRAM
27 // ** Espressif Internal Boards **
28 // #define CAMERA_MODEL_ESP32_CAM_BOARD
29 // #define CAMERA_MODEL_ESP32S2_CAM_BOARD
30 // #define CAMERA_MODEL_ESP32S3_CAM_LCD
```

1. A 'CameraWebServer.ino' window should now appear with a bunch of camera model's being listed. You'll choose the one that fits your model by uncommenting it.

2. After that you'll scroll down a bit where you'll be asked to enter your WiFi credentials because we do need internet to connect to the web server.

```
#define CAMERA_MODEL_ESP32S2_CAM_BOARD
#define CAMERA_MODEL_ESP32S3_CAM_LCD
#define CAMERA_MODEL_DFRobot_FireBeetle2_ESP32S
#define CAMERA_MODEL_DFRobot_Romeo_ESP32S3 // H
#include "camera_pins.h"

=====
Enter your WiFi credentials
=====

const char* ssid = "*****";
const char* password = "*****";
```



AI Thinker ESP32-CAM



CameraWebServer.ino

app_httpd.cpp

camera_index.h

camera_pins.h



```
41 void startCameraServer();
42 void setupLedFlash(int pin);
43
44 void setup() {
45     Serial.begin(115200);
46     Serial.setDebugOutput(true);
47     Serial.println();
48
49     camera_config_t config;
50     config.ledc_channel = LEDC_CHANNEL_0;
51     config.ledc_timer = LEDC_TIMER_0;
52     config.pin_d0 = Y2_GPIO_NUM;
53     config.pin_d1 = Y3_GPIO_NUM;
54     config.pin_d2 = Y4_GPIO_NUM;
55     config.pin_d3 = Y5_GPIO_NUM;
56     config.pin_d4 = Y6_GPIO_NUM;
57     config.pin_d5 = Y7_GPIO_NUM;
```

Proceeding that you'll upload/compile the code by clicking the right-facing arrow. It will turn a golden color when clicked.

At the bottom-right you'll see the code being compiled.

Output



Compiling sketch...



```
47 Serial.println();
```

Output Serial Monitor

```
Writing at 0x0016d775... (95 %)
Writing at 0x001730a7... (96 %)
Writing at 0x0017a213... (98 %)
Writing at 0x0017fbe1... (100 %)
Wrote 1512624 bytes (987265 compressed) at 0x00010000
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

1

Output Serial Monitor x

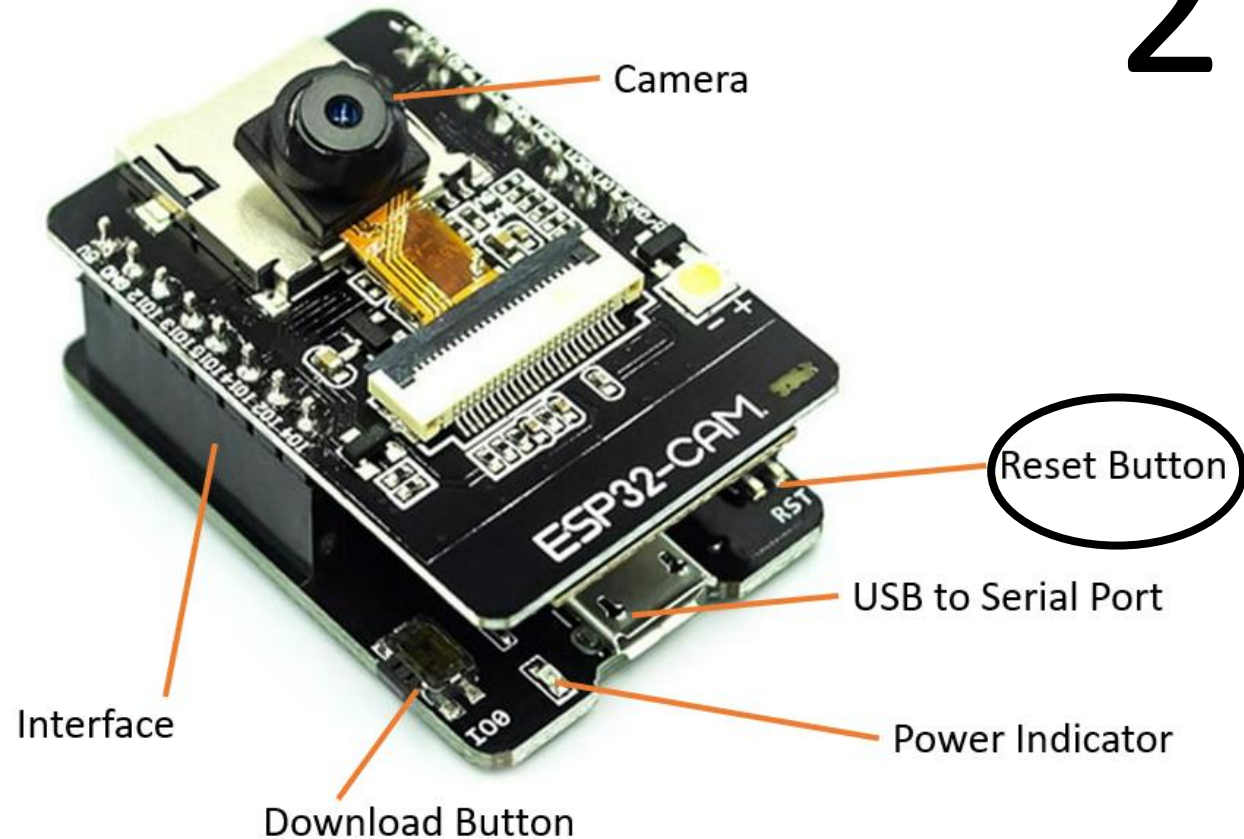
Message (Enter to send message to 'AI Thinker ESP32-CAM' on 'COM6')

```
cam_hal: EV-EOF-OVF
cam_hal: EV-EOF-OVF
cam_hal: EV-EOF-OVF
cam_hal: EV-EOF-OVF
.....
WiFi connected
Camera Ready! Use 'http://10.0.0.10' to connect
```

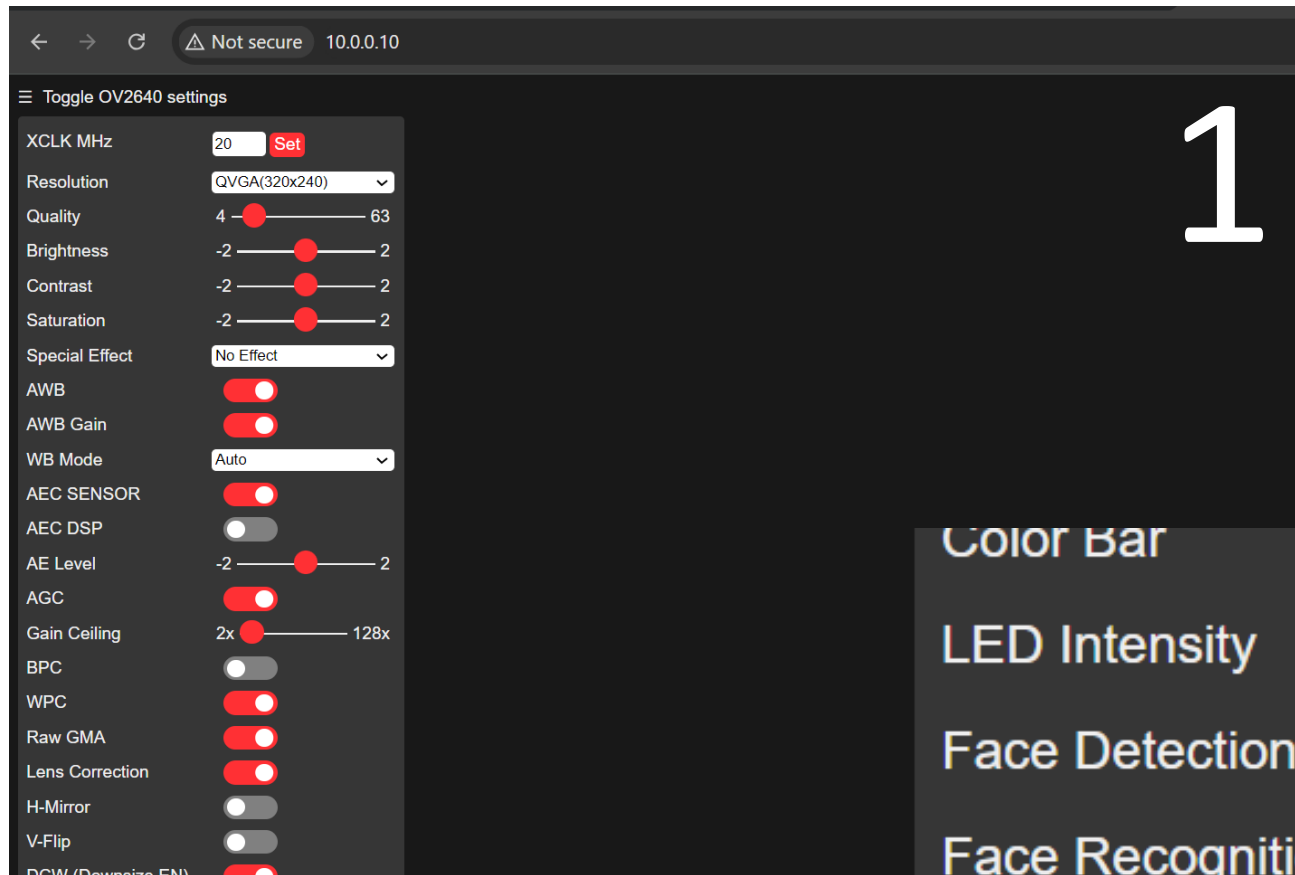
3

1. After the compilation is done you'll the compiler writing the instructions from the code.
2. When it's done you'll click the ESP32-CAM reset button
3. Then you'll access the serial monitor by clicking:
Sketch -> Serial monitor

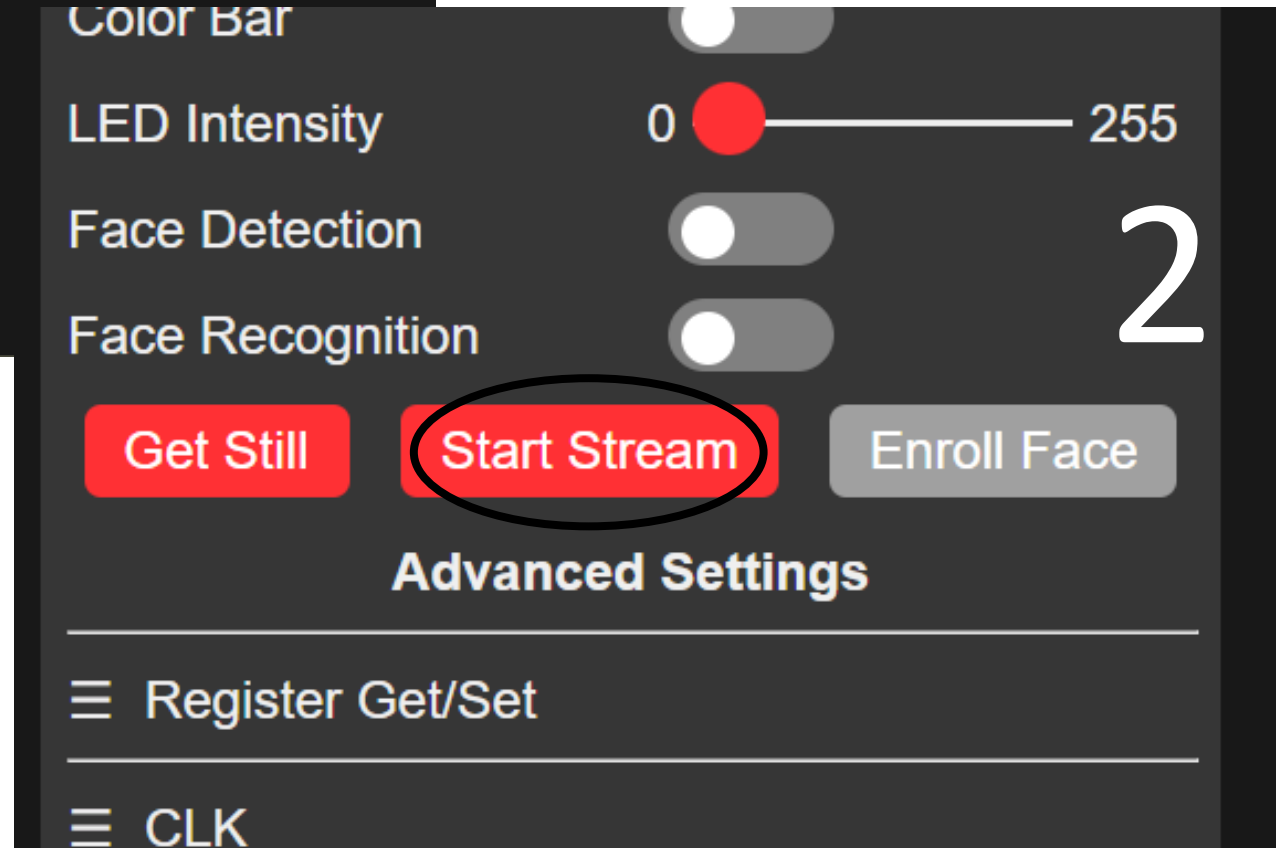
2



3. Then you'll see the server link to connect to. All you have to do it copy and paste the link to a web browser.



1. Once loaded you should see the web server's interface.
2. You'll then scroll down and click 'Start Stream' to access the ESP32-CAM's vision



With that you now have a system that you can take advantage of for various computer vision projects!