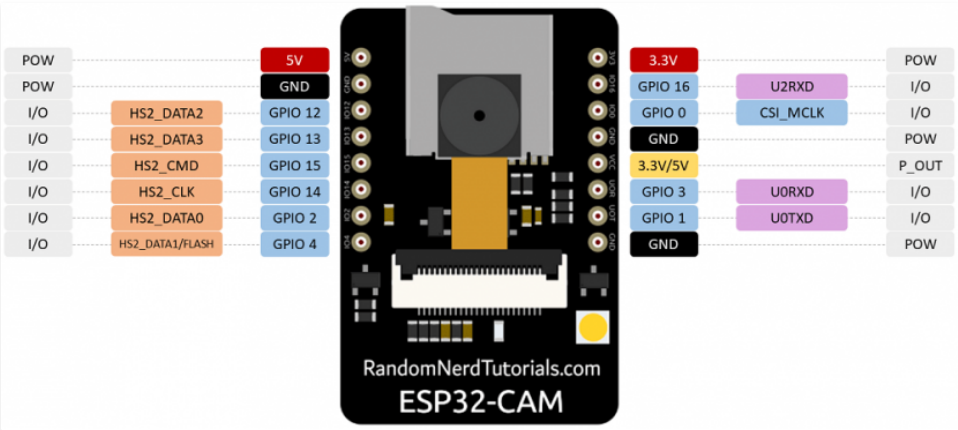
Edgoad.com

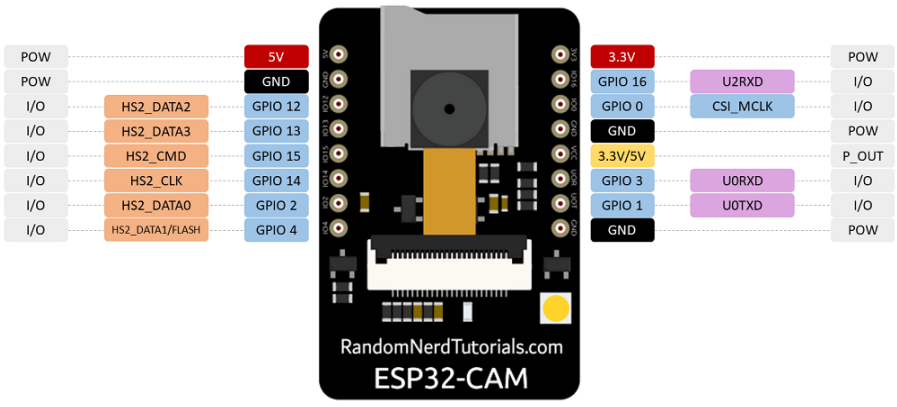


Electronics

Programming ESP32CAM using Arduino Uno

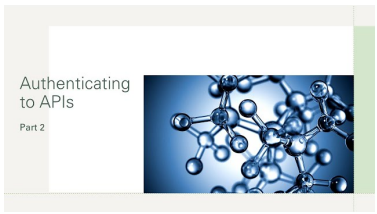
February 18, 2021 Ed Goad 0 Comments

This is inspired by several others who have previously completed this task. This is a compilation of what I learned from various sources in case it needs to be reproduced in the future. NOTE: Most images are taken from other sites and I claim no copyright.



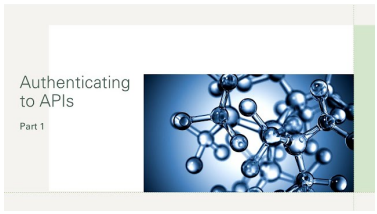
I purchased the ESP32Cam in a 2 pack from Amazon (2 Pack ESP32-CAM WiFi Bluetooth Camera Module Development Board ESP32 with Camera Module OV2640) for $18.99. I was actually looking for an ESP32 board for a different project,

Recent Videos



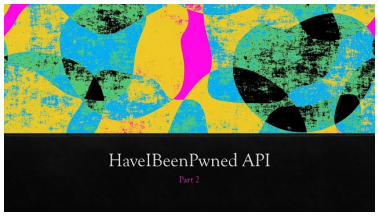
Authenticating to APIs par...

Continuing our discussion of authenticating to APIs to better keep our authentication tokens



Authenticating to APIs par...

Introduction to using APIs that require authentication. Walking through the GitHub API



HaveIBeenPwned API part 2

Continuing our use of the HaveIBeenPwned API and using

but the addition of the CAM module looked interesting and I believe will work for this .

Add ESP32 to Arduino IDE

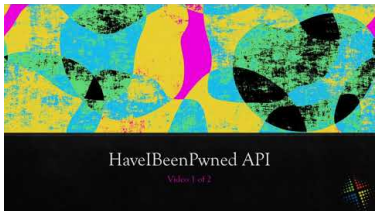
To work with ESP32 variants, we need to add the boards into the IDE. To do this we do the following steps:

1. Open the Arduino IDE
2. Go to **File | Preferences**
    - In the **Additional Boards Manager URLs** field, add the following 2 URLS separated with a comma:
    - `https://dl.espressif.com/dl/package_esp32_index.json`, `http://arduino.esp8266.com/stable/package_esp8266com_index.json`
    - Click **OK**
3. Go to **Tools | Board | Boards Manager**
    - In the **Filter your search** bar, enter ESP32
    - Select **esp32** by **Espressif Systems** and click **Install**
    - Once installed, click **Close**
4. Go to **Tools | Board | ESP32 Arduino** and select **ESP32 Wrover Module**

Connect the Uno to the ESP32

Using jumpers or other temporary wiring, create the diagram as shown below to connect the ESP32CAM to the Arduino Uno for programming.

this in a more dynamic Python

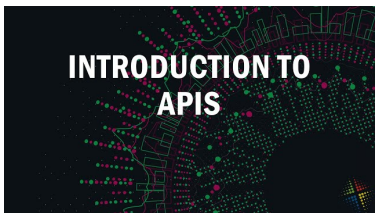


HaveIBeenPwned API part 1

Using the HaveIBeenPwned public API to test passwords for security. Validating if passwords

JSON and APIs

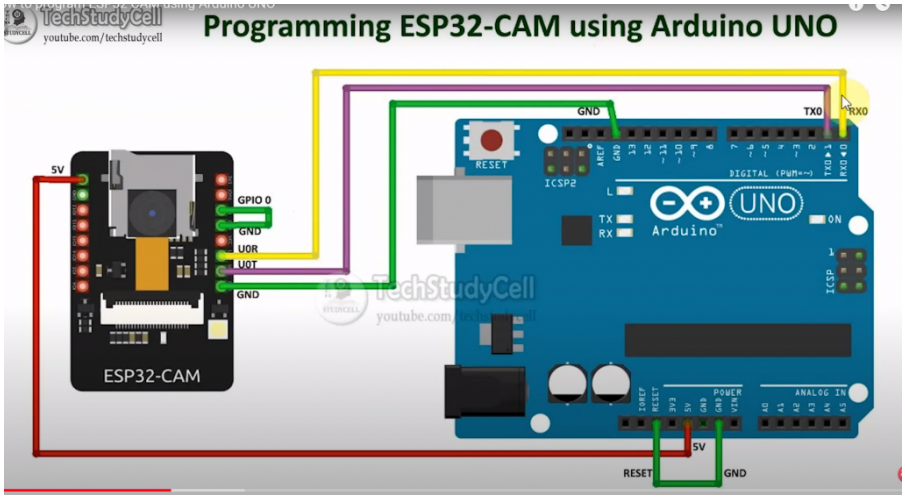
An introduction to working with JSON data returned from APIs. Specifically, creating a script to



Introduction to APIs

A brief introduction to working with online APIs and the tools to access them

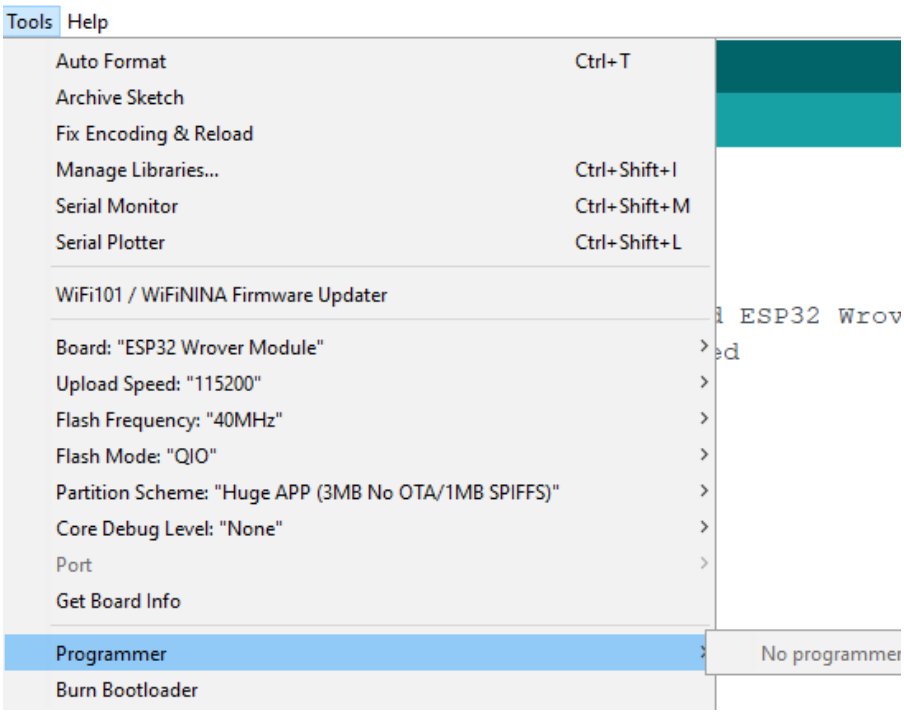
Load more



Open Camera sample sketch

Once the ESP32 board is loaded into the IDE, now we can load the sample camera sketch to test the board.

1. Begin by attaching the Arduino UNO to your computer
2. In the IDE, go to **File | Examples | ESP32 | Camera | CameraWebServer**
3. At the beginning where it says **Select camera model**
    - Add a comment (double forward slash) in front of the line `#define CAMERA_MODEL_WROVER_KIT`
    - Uncomment the line `#define CAMERA_MODEL_AI_THINKER`
    - **NOTE:** The model may be different depending on where and when you purchase your version
4. Add the **SSID** and **Password** for your WIFI
5. On the **Tools** menu, make sure you have the following selected:
6. FlashMode: **QIO**
    - Partition Scheme: **Huge App (3MB...**
    - Flash Frequency: **40MHz**
    - Upload Speed: **115200**
    - Port: **Arduino Uno**
    - Programmer: **AVR ISP**
    - **NOTE:** This last last option wasn't available for me and worked without a programmer set. I am leaving it here because it was in the initial instructions



Upload sketch

Once the UNOand ESP32cam are connected and the sketch is ready, we can upload the code.

1. In the Arduino IDE, click **Upload**
2. Watch the black window at the bottom until it says **Connecting**
3. On the ESP32cam, click the **Reset** button once and wait
4. The upload will take about 1 minute
    - NOTE: If this step fails for some reason, disconnect the USB cable from the UNO, reconnect it, and restart the Upload
5. When the upload is finished, select **Tools | Serial Monitor**
6. Ensure the speed is set to **115200**
7. On the ESP32cam disconnect the cable between **GPIO 0 and GND** and click the **Reset** button
8. Watch the **Serial Monitor** for the IP address of the ESP32
    - If you missed the IP address in the output, clicking the Reset button will restart the ESP32cam module and will reprint the address

Connect to web page

Once you have the IP address from the Serial Monitor, enter it into a web browser. Note the web page may be slow to respond, especially when first accessed.

In the web browser, click **Start Stream** to view the camera.

To continue using the ESP32 cam, only power is needed to the 5V and **GND** pins.

A few online references for use here:

- https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/
- https://lastminuteengineers.com/esp32-arduino-ide-tutorial/
- https://lastminuteengineers.com/esp32-ota-updates-arduino-ide/
- https://www.youtube.com/watch?v=q-KIpFIbRMk

← Installing VS Code on Raspberry Pi

Python for Cybersecurity →

Leave a Reply

You must be logged in to post a comment.