

Overview: ESP32 CAM Based Object Detection & Identification

This tutorial introduces the topic of ESP32 CAM Based Object Detection & Identification with OpenCV. OpenCV is an open-sourced image processing library that is very widely used not just in industry but also in the field of research and development.

In this tutorial, we will go through its features, pins description and the method to program ESP32 Camera Module using FTDI Module. We will also set up the Arduino IDE for the ESP32 Camera Module. We will also upload the firmware and then work on the object detection & identification part. The script for object detection is written in the python programming language, thus we will also have to install Python and its required Libraries.

In an earlier ESP32 CAM Based project we learned about [Face Detection System](#) & also [Color Detection System](#) using Python & OpenCV. This project also requires the use of OpenCV for Object Detection & Identification.

ESP32 CAM Module

The ESP32 Based Camera Module developed by AI-Thinker. The controller is based on a 32-bit CPU & has a combined Wi-Fi + Bluetooth/BLE Chip. It has a built-in 520 KB SRAM with an external 4M PSRAM. Its GPIO Pins have support like UART, SPI, I2C, PWM, ADC, and DAC.



Front Side



Back Side

The module combines with the OV2640 Camera Module which has the highest Camera Resolution up to 1600 × 1200. The camera connects to the ESP32 CAM

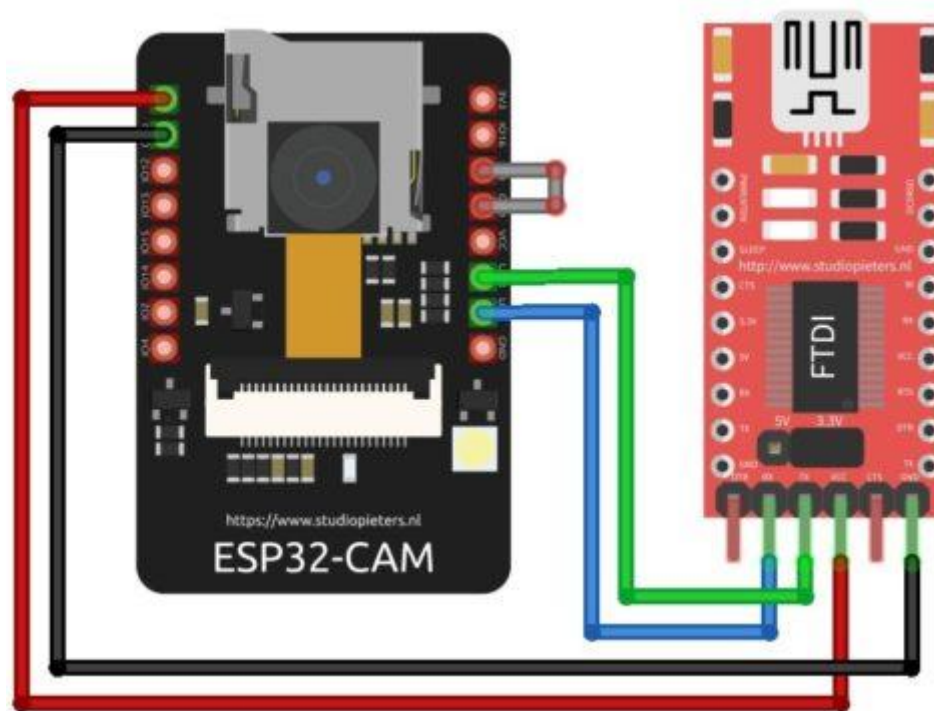
Board using a 24 pins gold plated connector. The board supports an SD Card of up to 4GB. The SD Card stores capture images.

To learn in detail about the ESP32 Camera Module you can refer to our previous [Getting Started Tutorial](#).

ESP32-CAM FTDI Connection

The board doesn't have a programmer chip. So In order to program this board, you can use any type of USB-to-TTL Module. There are so many FTDI Module available based on [CP2102](#) or [CP2104](#) Chip or any other chip.

Make a following connection between FTDI Module and ESP32 CAM module.



ESP32-CAM

GND

5V

U0R

U0T

GPIO0

FTDI Programmer

GND

VCC

TX

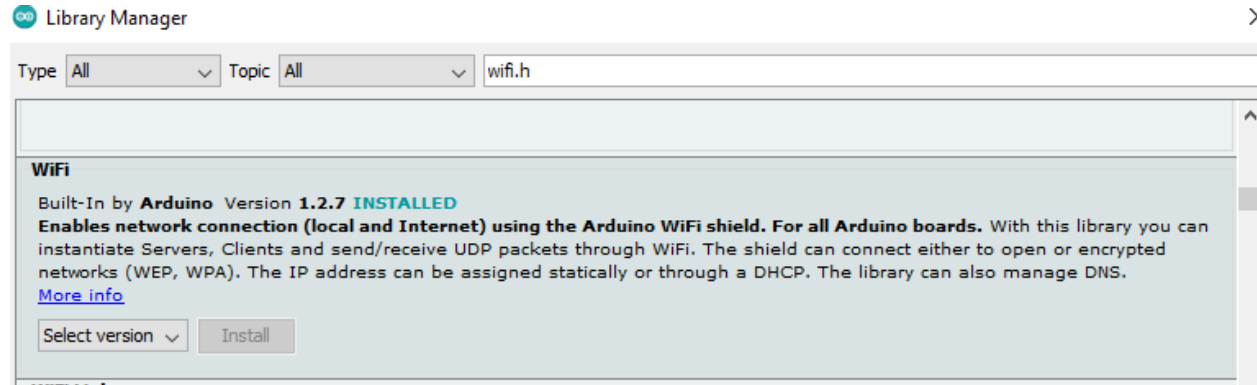
RX

GND

Connect the 5V & GND Pin of ESP32 to 5V & GND of FTDI Module. Similarly, connect the Rx to UOT and Tx to UOR Pin. And the most important thing, you need to short the IO0 and GND Pin together. This is to put the device in programming mode. Once programming is done you can remove it.

Installing ESP32CAM Library

Here we will not use the general ESP webserver example rather another streaming process. Therefore we need to add another ESPCAM library. The esp32cam library provides an object oriented API to use OV2640 camera on ESP32 microcontroller. It is a wrapper of esp32-camera library.



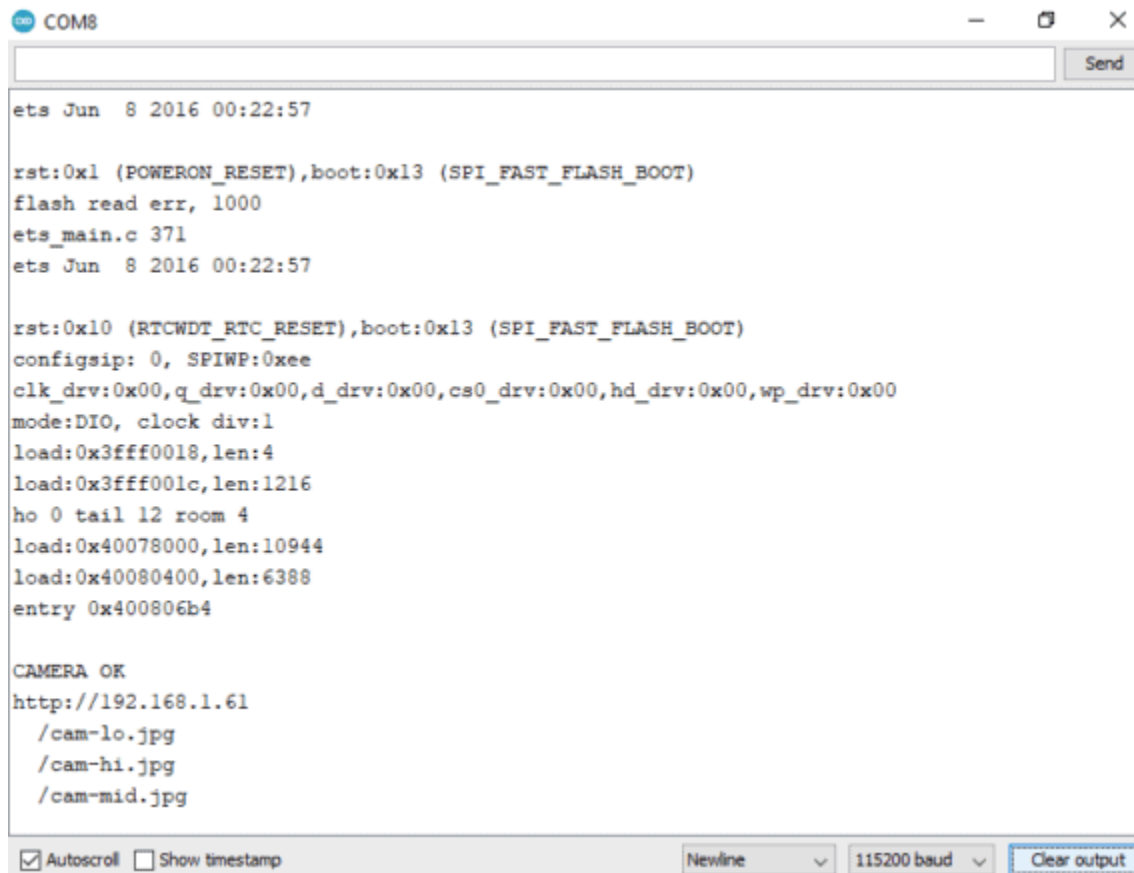
Source Code/Program for ESP32 CAM Module

Here is a source code for Object Detection & Identification with ESP32 Camera & OpenCV. Copy the code and paste it in the Arduino IDE. Before Uploading the code you have to make a small change to the code. Change the SSID and password variable and in accordance with your WiFi network.

Now compile and upload it to the ESP32 CAM Board. But during uploading, you have to follow few steps every time.

- Make sure the IO0 pin is shorted with the ground when you have pressed the upload button.
- If you see the dots and dashes while uploading press the reset button immediately
- Once the code is uploaded, remove the IO1 pin shorting with Ground and press the reset button once again.
- If the output is the Serial monitor is still not there then press the reset button again.

Now you can see a similar output as in the image below.



```

COM8
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
flash read err, 1000
ets_main.c 371
ets Jun  8 2016 00:22:57

rst:0x10 (RTCWDT_RTC_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1216
ho 0 tail 12 room 4
load:0x40078000,len:10944
load:0x40080400,len:6388
entry 0x400806b4

CAMERA OK
http://192.168.1.61
/cam-lo.jpg
/cam-hi.jpg
/cam-mid.jpg
  
```

Autoscroll ☐ Show timestamp Newline 115200 baud Clear output

Here, copy the IP address visible, we will be using it to edit the URL in python code

Python Library Installation

For the live stream of video to be visible on our computer we need to write a Python script that will enable us to retrieve the frames of the video. The first step is to install Python. Go to python.org and download Python. Once download, install Python.

Python Code for ESP32 CAM Object Detection/Identification

Now open the Idle code editor or any other python code editor.

Copy and paste the code from below and do the replacements as mentioned below.

Here we have to replace the IP address with the IP on Arduino Serial Monitor. For the first time, it will install a few files if they are not existing.

Once we have done that we can see two windows named live transmission and detect it.

Now in the detected window, one can view different detected objects as around them different colored boxes are visible.