

.:: ESP32-CAM - INSTALACIÓN EN ARDUINO IDE Y PRUEBA::.

Autor

Freddy Alcarazo | @surflaweb | @alcarazolabs

Materiales:

- X1 ESP32-CAM.
- X1 PROGRAMADOR USS TO TTL CON SALIDA DE 5V.
- X4 CABLES HEMBRA-HEMBRA.
- X1 JUMPER O UN CABLE HEMBRA HEMBRA PARA HACER PUENTE.

Pasos:

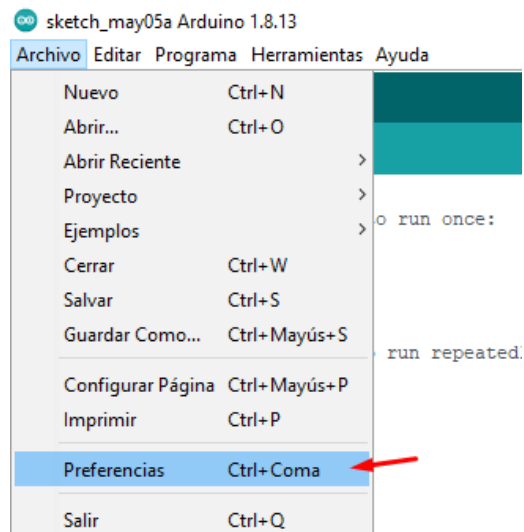
1. Instalar el paquete de las placas de tarjetas del ESP-32 en Arduino IDE

Copiar link:

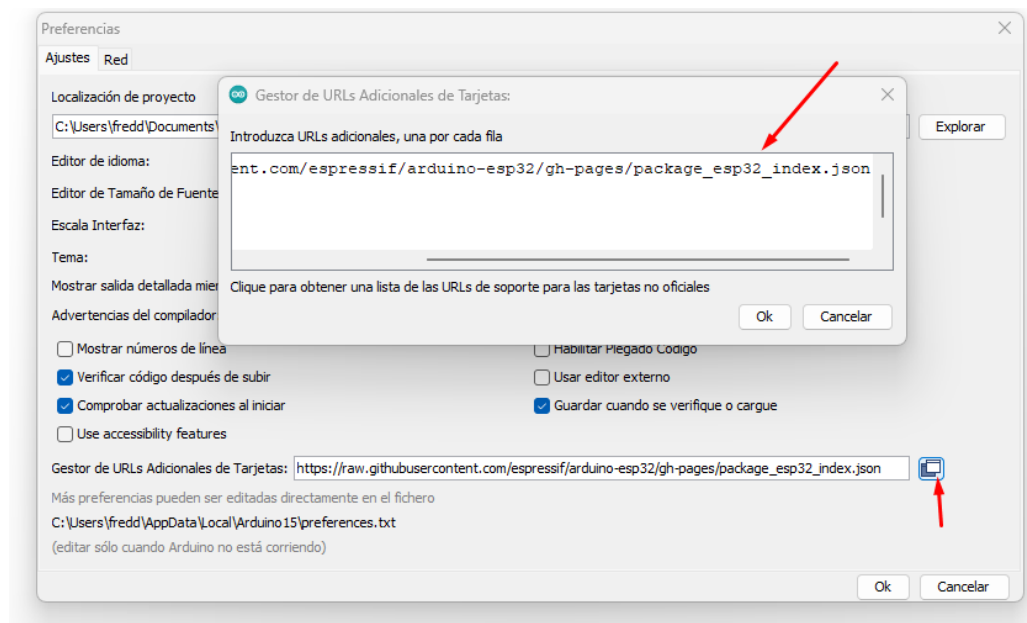
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

1.1 Abrir Arduino IDE

1.2 Ir a preferencias:

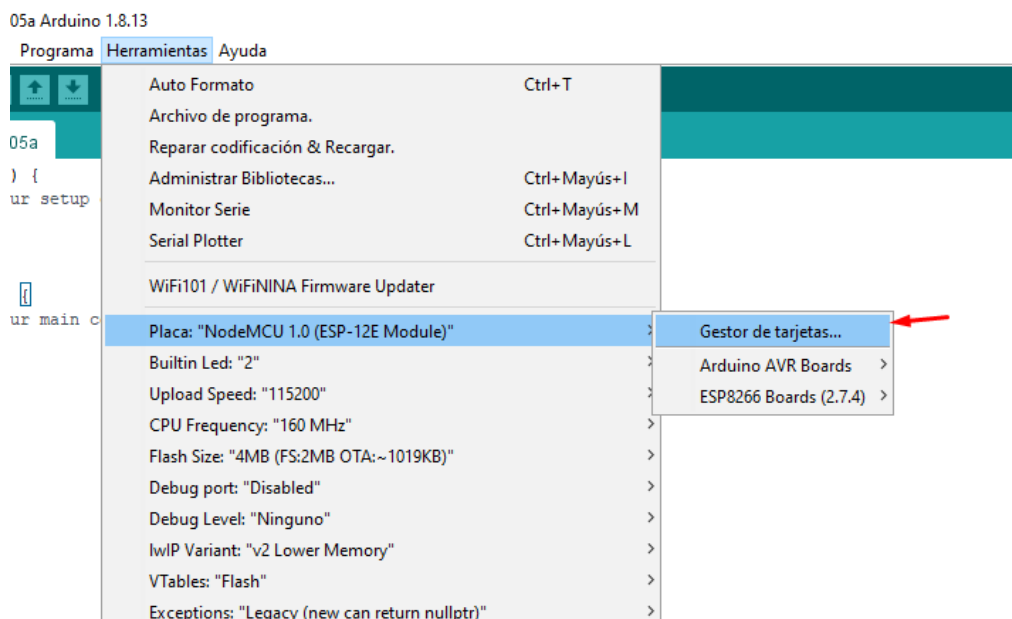


1.3 Agregar URL



Click en ok.

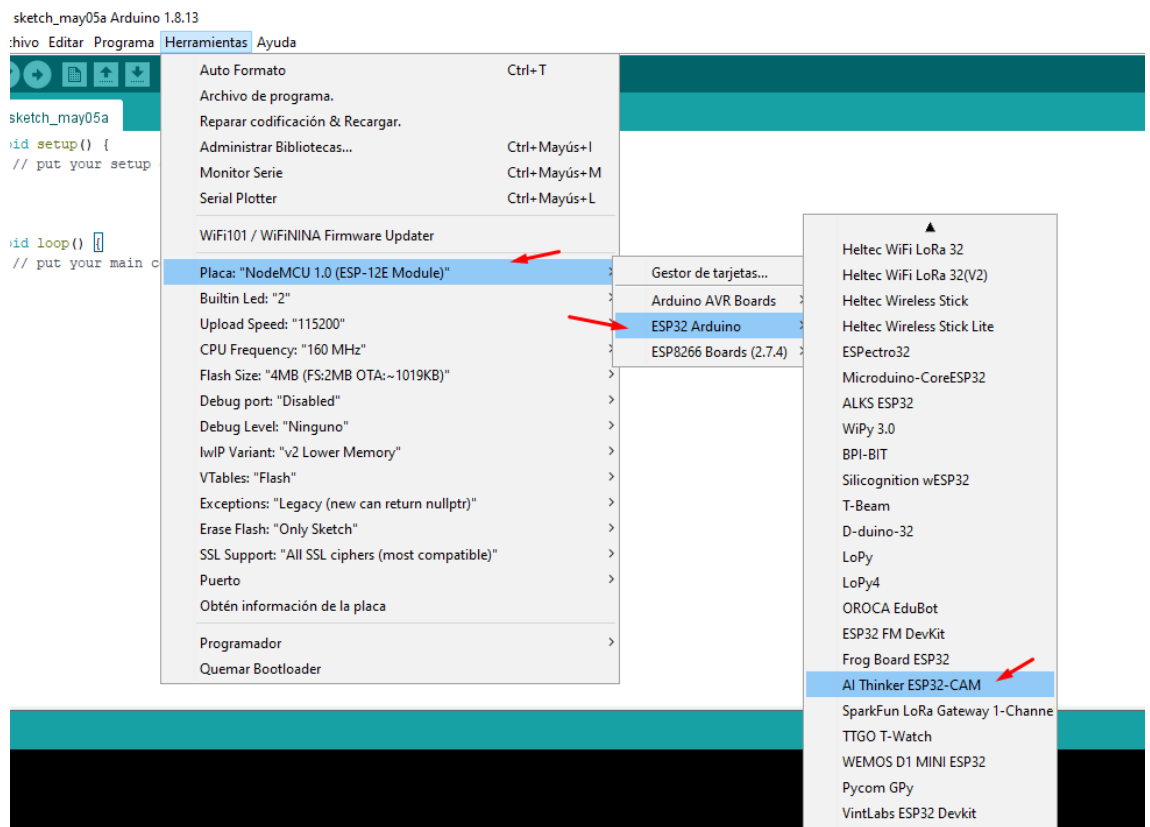
1.4 Abrir gestor de tarjetas



1.5 Buscar “esp32” e Instalar

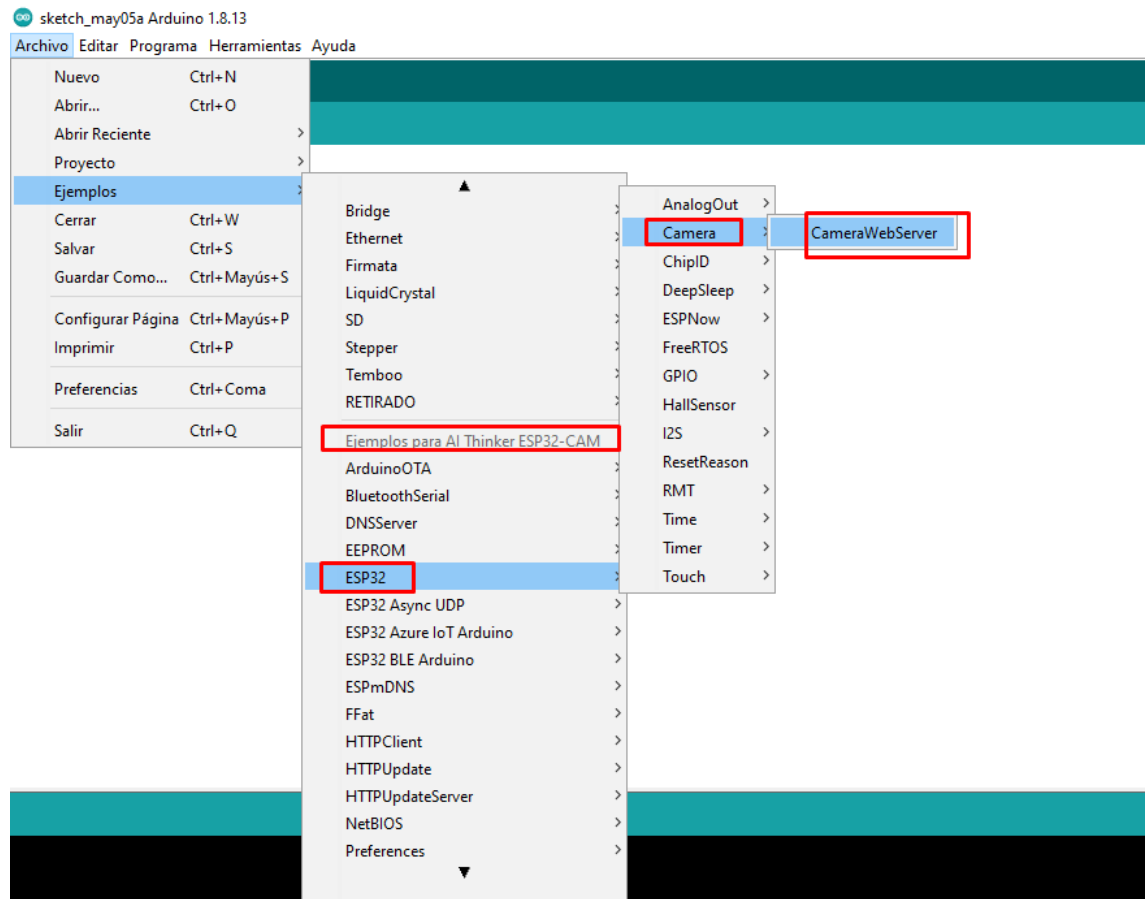


1.6 Elegir modelo de placa que vamos a conectar una “AI Thinker ESP32-CAM”



2. Elegir ejemplo a ejecutar en el ESP32-CAM

Lo que haremos ahora es elegir uno de los ejemplos disponibles del “AI Thinker ESP32-CAM”. Nos vamos a Ejemplos->ESP32->Camera y elegimos CameraWebServer.



Luego modificamos el código del ejemplo cargado:

```

CameraWebServer$ app_httpd.cpp camera_index.h camera_pins.h
#include "esp_camera.h"
#include <WiFi.h>

//
// WARNING!!! PSRAM IC required for UXGA resolution and high JPEG quality
//           Ensure ESP32 Wrover Module or other board with PSRAM is selected
//           Partial images will be transmitted if image exceeds buffer size
//

// 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 version B Has PSRAM
// #define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
// #define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
#define CAMERA_MODEL_AI_THINKER // Has PSRAM
// #define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM

#include "camera_pins.h"

const char* ssid = "CLARO_WAN";
const char* password = "942918777";

void startCameraServer();

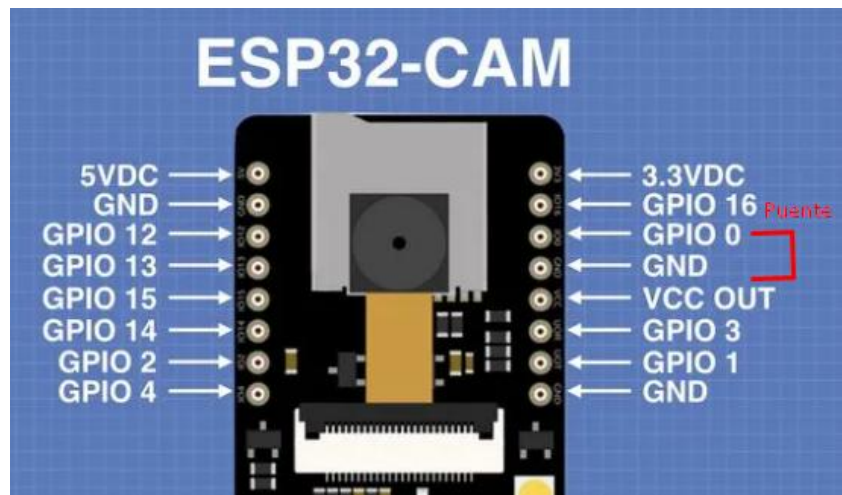
void setup() {
    Serial.begin(115200);
    Serial.println("Starting camera server");
}

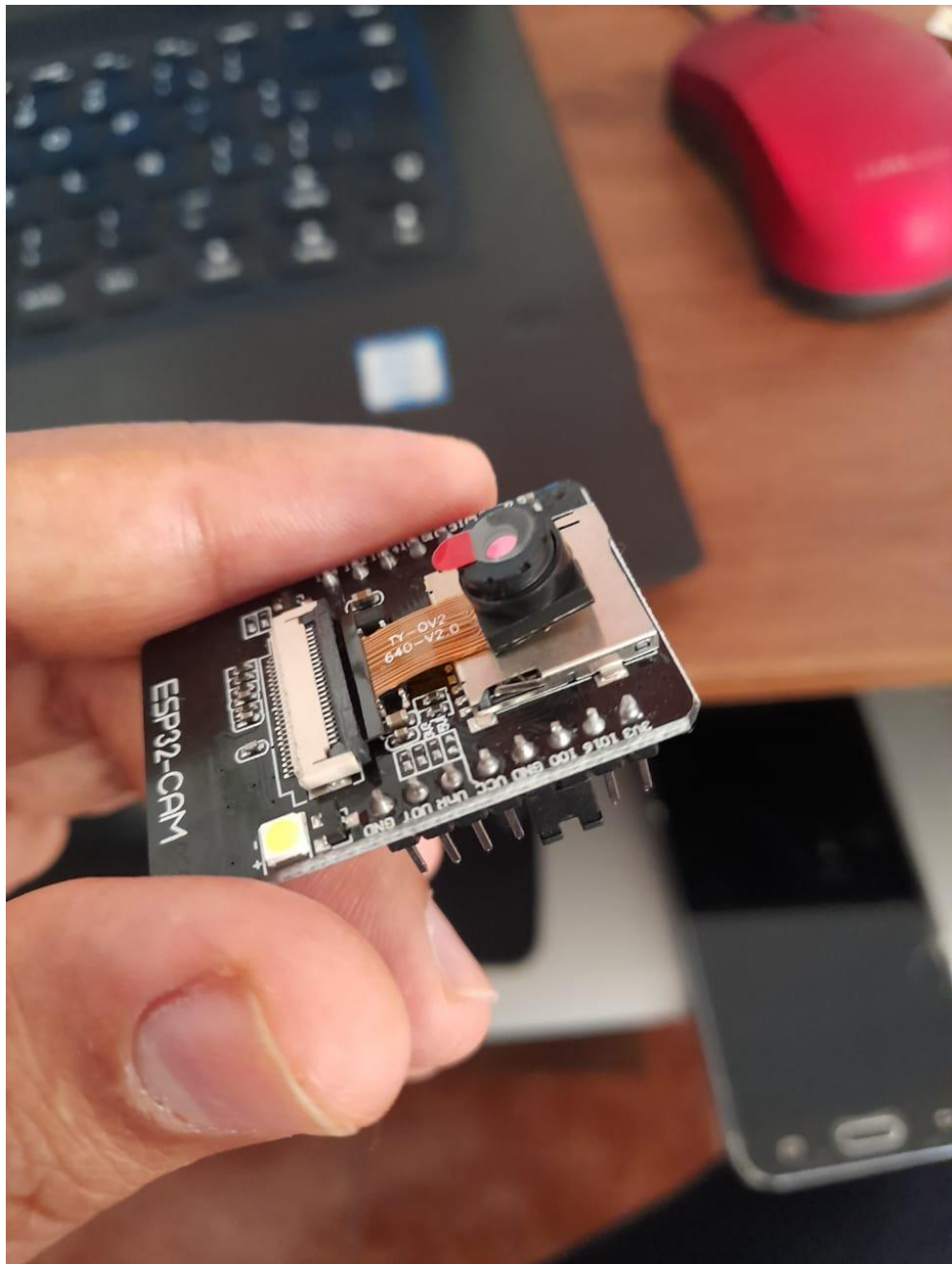
```

Descomentamos la línea de código “#define CAMERA_MODEL_AI_THINKER” y comentamos el modelo de camera que trae por defecto. Luego más abajo solo confirmamos el nombre de nuestra red wifi a la que se va a conectar y ponemos la clave.

3. Programar ESP32-CAM con el programador USB-TTL

Para poder programar la ESP32-CAM tenemos que primero agregar un jumper al ESP32-CAM esto se hace poniendo a tierra el PIN GPIO 0 al GND, este es el tercer pin del lado derecho de la placa y tiene como número 100 en mi caso. Si no tiene un jumper conectar los dos pines con un cable hembra-hembra o bien poner el ESP32-CAM en el protoboard.

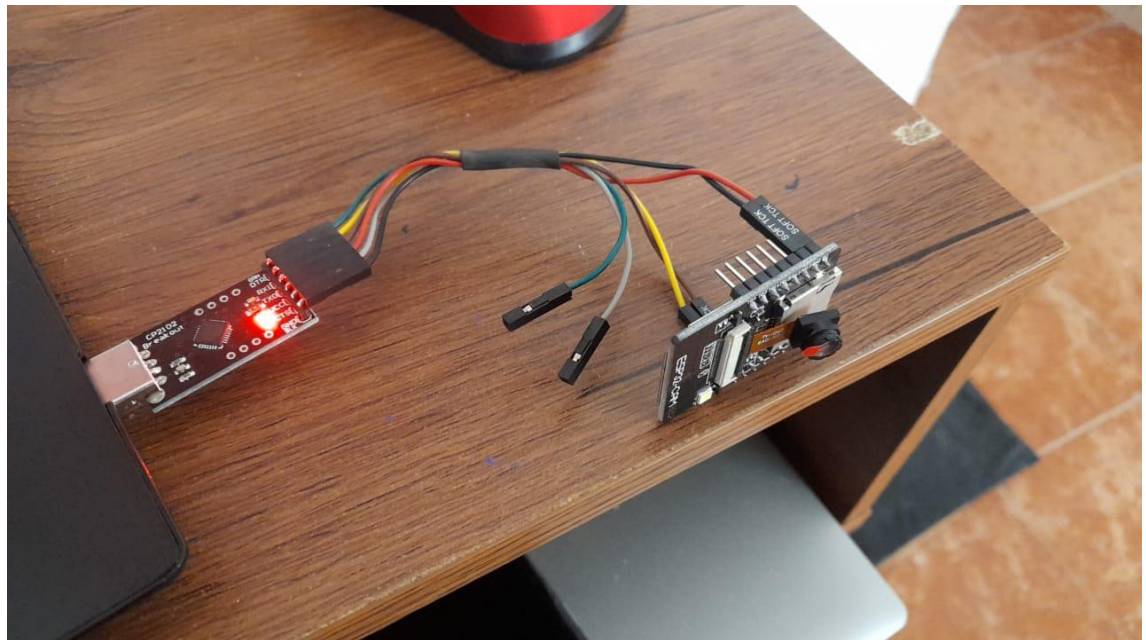
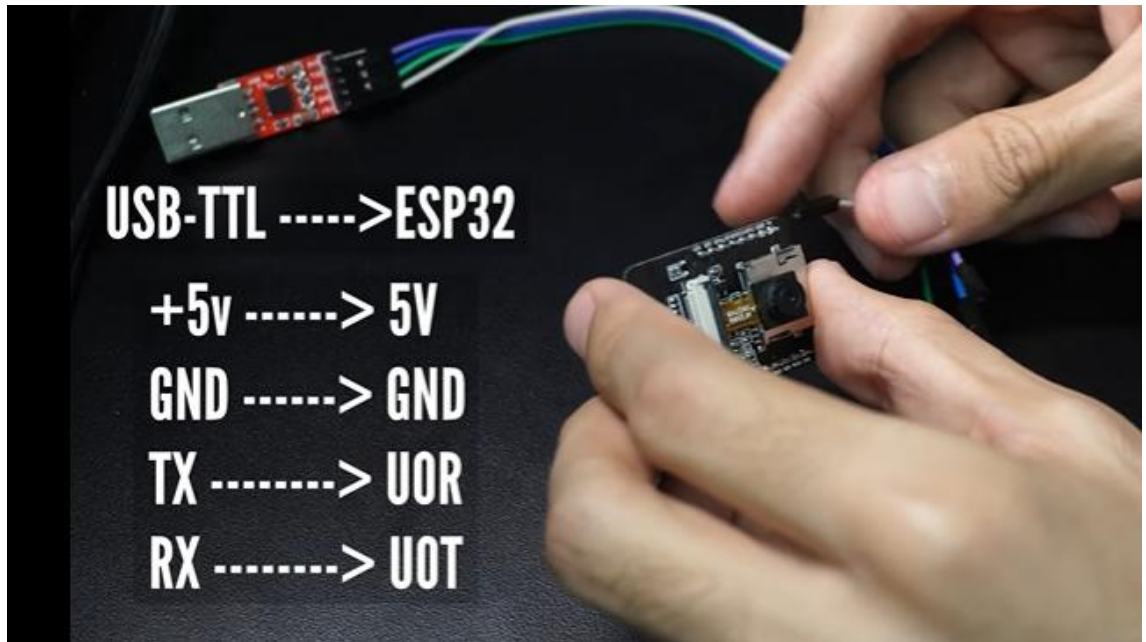




3.1 Conectar pines del programador USB-TTL al ESP-CAM:

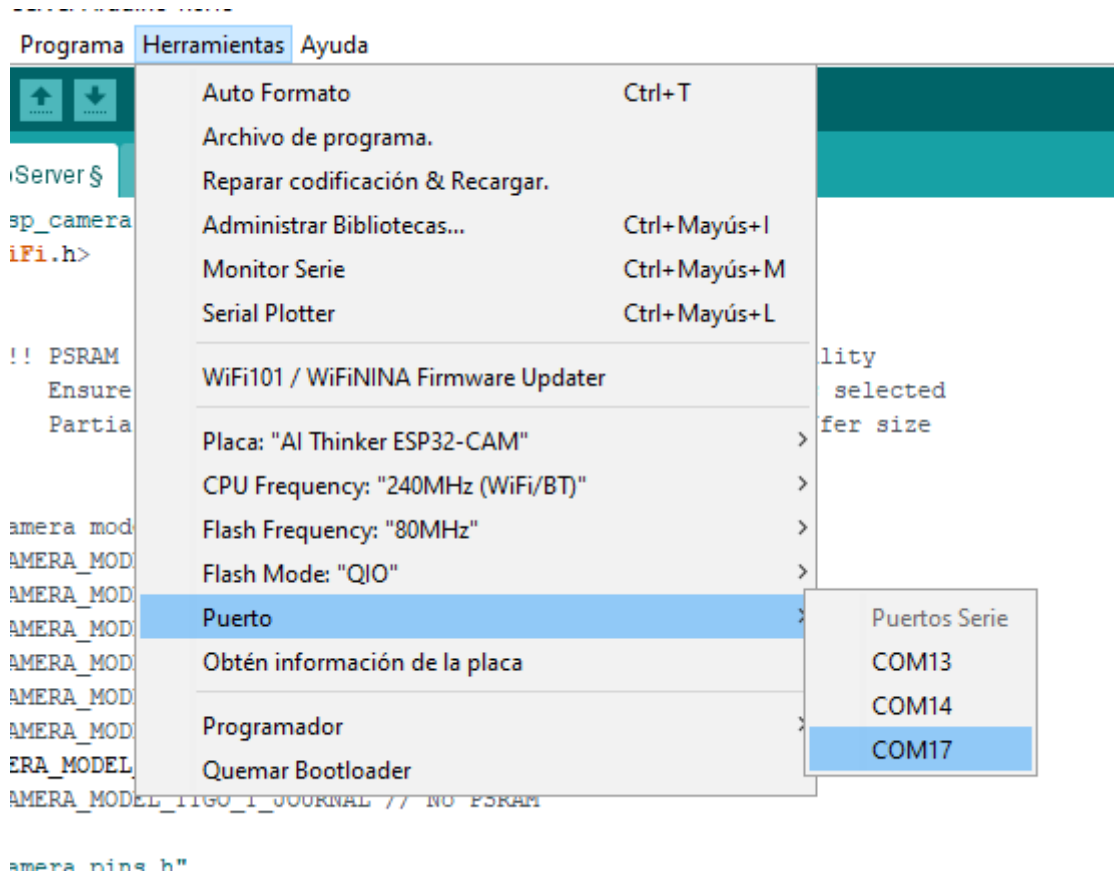
En la siguiente imagen podemos ver a que pines vamos a conectar los pines del USB-TTL al ESP32-CAM:

* Conectar a 5v. Si luego no se puede programar conectar una fuente externa de 5v ya que a veces los puertos USB no suministran bien los 5v.



3.2 Conectar el USB-TTL al puerto USB de la computadora

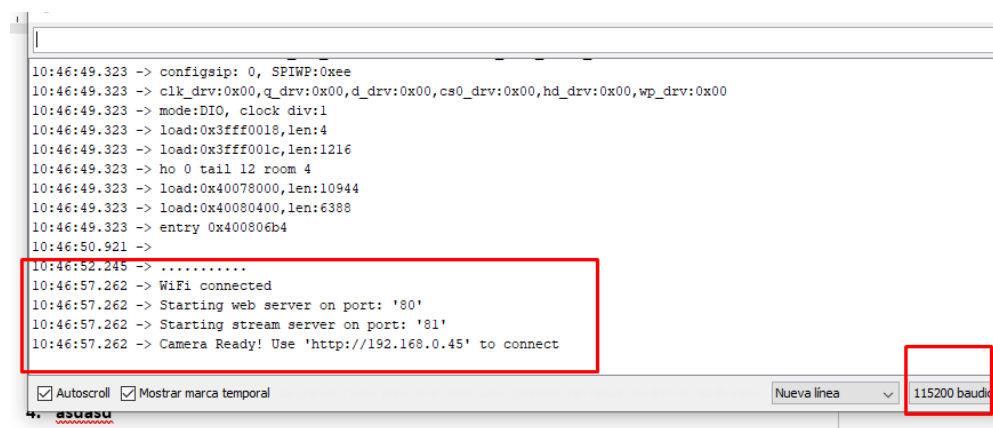
3.3 Elegir puerto COM



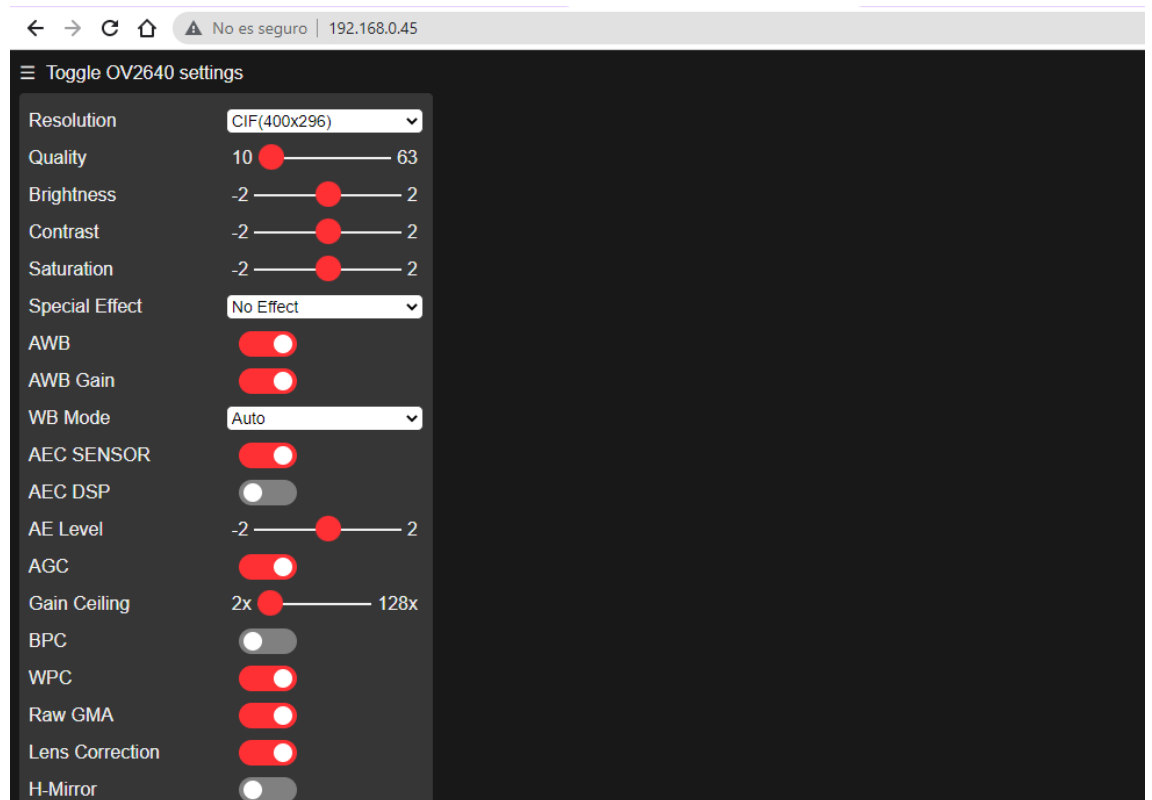
3.4 Enviar código.

3.5 Abrir monitor serial y luego quitar el JUMPER y presionar botón de reset del ESP32-CAM

Antes de quitar el JUMPER y presionar el botón de reset, abrimos el monitor serial para obtener la información que nos dará el código subido. Una vez hecho esto recién quitamos el jumper y reseteamos la placa con su botón que está detrás.



Y casi listo! Ahora solo copiamos la ip que nos da para acceder a ver la cámara.



Ahora solo presionamos el botón “Start Stream” y elegir la calidad de imagen que más nos guste.

