//set up wifi password first

const char\* ssid = "MY HOME"; //my network SSID

const char\* password = "93337808"; //my network password

// load the script and upload to drive

String myScript = "/macros/s/AKfycbwgIOXE1PJBGaf3W5fpm3bT7QYGw7bz2CbrLoEAK5qztGtmHBIN/exec"; //link of google script of esp32-cam A

//String myScript = "/macros/s/AKfycbwlINcdpcA9EH6vVMXnEYD0-RhvEWIMY3fvZt0x9npDk\_6O5Uo/exec";

//String myScript = "/macros/s/AKfycbyvJPQjGzgq0nYjozIw\_6-GbFSAaXhVcOXqzseZJC-YmfyHMcT4/exec"; // link of google script of esp32-cam B

const char\* myDomain = "script.google.com";

String myFilename = "filename=ESP32-CAM.jpg"; //naming

String mimeType = "&mimetype=image/jpeg"; //filetype

String myImage = "&data="; //package

#include <WiFi.h>

#include <WiFiClientSecure.h>

#include "soc/soc.h"

#include "soc/rtc\_cntl\_reg.h"

#include "Base64.h"

#include "esp\_camera.h"

//pin setup default

#define CAMERA\_MODEL\_AI\_THINKER

#define PWDN\_GPIO\_NUM 32

#define RESET\_GPIO\_NUM -1

#define XCLK\_GPIO\_NUM 0

#define SIOD\_GPIO\_NUM 26

#define SIOC\_GPIO\_NUM 27

#define Y9\_GPIO\_NUM 35

#define Y8\_GPIO\_NUM 34

#define Y7\_GPIO\_NUM 39

#define Y6\_GPIO\_NUM 36

#define Y5\_GPIO\_NUM 21

#define Y4\_GPIO\_NUM 19

#define Y3\_GPIO\_NUM 18

#define Y2\_GPIO\_NUM 5

#define VSYNC\_GPIO\_NUM 25

#define HREF\_GPIO\_NUM 23

#define PCLK\_GPIO\_NUM 22

void setup()

{

//wifi

WRITE\_PERI\_REG(RTC\_CNTL\_BROWN\_OUT\_REG, 0);

Serial.begin(115200);

delay(10);

pinMode(5, OUTPUT); //led as output

WiFi.mode(WIFI\_STA);

Serial.println("");

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

long int StartTime=millis();

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

if ((StartTime+10000) < millis()) break;

}

Serial.println("");

Serial.println("STAIP address: ");

Serial.println(WiFi.localIP());

Serial.println("");

if (WiFi.status() != WL\_CONNECTED) { //wifi not connect

Serial.println("Reset");

//led set up

ledcAttachPin(4, 3);

ledcSetup(3, 5000, 8);

ledcWrite(3,10);

delay(200);

ledcWrite(3,0);

delay(200);

ledcDetachPin(3);

delay(1000);

ESP.restart();

}

else {

//wifi connected

ledcAttachPin(4, 3);

ledcSetup(3, 5000, 8);

for (int i=0;i<5;i++) {

ledcWrite(3,10);

delay(200);

ledcWrite(3,0);

delay(200);

}

ledcDetachPin(3);

}

camera\_config\_t config;

config.ledc\_channel = LEDC\_CHANNEL\_0;

config.ledc\_timer = LEDC\_TIMER\_0;

config.pin\_d0 = Y2\_GPIO\_NUM;

config.pin\_d1 = Y3\_GPIO\_NUM;

config.pin\_d2 = Y4\_GPIO\_NUM;

config.pin\_d3 = Y5\_GPIO\_NUM;

config.pin\_d4 = Y6\_GPIO\_NUM;

config.pin\_d5 = Y7\_GPIO\_NUM;

config.pin\_d6 = Y8\_GPIO\_NUM;

config.pin\_d7 = Y9\_GPIO\_NUM;

config.pin\_xclk = XCLK\_GPIO\_NUM;

config.pin\_pclk = PCLK\_GPIO\_NUM;

config.pin\_vsync = VSYNC\_GPIO\_NUM;

config.pin\_href = HREF\_GPIO\_NUM;

config.pin\_sscb\_sda = SIOD\_GPIO\_NUM;

config.pin\_sscb\_scl = SIOC\_GPIO\_NUM;

config.pin\_pwdn = PWDN\_GPIO\_NUM;

config.pin\_reset = RESET\_GPIO\_NUM;

config.xclk\_freq\_hz = 20000000;

config.pixel\_format = PIXFORMAT\_JPEG;

// set up camera

esp\_err\_t err = esp\_camera\_init(&config);

if (err != ESP\_OK) {

Serial.printf("Camera init failed with error 0x%x", err);

delay(1000);

ESP.restart();

}

sensor\_t \* s = esp\_camera\_sensor\_get();

//frame size set up (resolution)

s->set\_framesize(s, FRAMESIZE\_CIF ); // UXGA|SXGA|XGA|SVGA|VGA|CIF|QVGA|HQVGA|QQVGA

}

void loop()

{

// led set up of taking photo

ledcAttachPin(4, 3);

ledcSetup(3, 5000, 8);

ledcWrite(3,50);

saveCapturedImage();

ledcWrite(3,0);

delay(15000);

}

void saveCapturedImage() {

Serial.println("Connect to " + String(myDomain));

WiFiClientSecure client;

if (client.connect(myDomain, 443)) {

Serial.println("Connection successful");

camera\_fb\_t \* fb = NULL;

fb = esp\_camera\_fb\_get();

if(!fb) {

Serial.println("Camera capture failed");

delay(1000);

ESP.restart();

return;

}

char \*input = (char \*)fb->buf;

char output[base64\_enc\_len(3)];

String imageFile = "";

for (int i=0;i<fb->len;i++) {

base64\_encode(output, (input++), 3);

if (i%3==0) imageFile += urlencode(String(output));

}

String Data = myFilename+mimeType+myImage;

esp\_camera\_fb\_return(fb);

Serial.println("Send a captured image to Google Drive.");

client.println("POST " + myScript + " HTTP/1.1");

client.println("Host: " + String(myDomain));

client.println("Content-Length: " + String(Data.length()+imageFile.length()));

client.println("Content-Type: application/x-www-form-urlencoded");

client.println();

client.print(Data);

int Index;

for (Index = 0; Index < imageFile.length(); Index = Index+1000) {

client.print(imageFile.substring(Index, Index+1000));

}

Serial.println("Waiting for response.");

long int StartTime=millis();

while (!client.available()) {

Serial.print(".");

delay(100);

if ((StartTime+30000) < millis()) {

Serial.println();

Serial.println("No response.");

//If no response, maybe need a greater value of waitingTime

break;

}

}

Serial.println();

while (client.available()) {

Serial.print(char(client.read()));

}

} else {

Serial.println("Connected to " + String(myDomain) + " failed.");

}

client.stop();

}

String urlencode(String str)

{

// url encode

String encodedString="";

char c;

char code0;

char code1;

char code2;

for (int i =0; i < str.length(); i++){

c=str.charAt(i);

if (c == ' '){

encodedString+= '+';

} else if (isalnum(c)){

encodedString+=c;

} else{

code1=(c & 0xf)+'0';

if ((c & 0xf) >9){

code1=(c & 0xf) - 10 + 'A';

}

c=(c>>4)&0xf;

code0=c+'0';

if (c > 9){

code0=c - 10 + 'A';

}

code2='\0';

encodedString+='%';

encodedString+=code0;

encodedString+=code1;

//encodedString+=code2;

}

yield();

}

return encodedString;

}