

Elektro_{xy}

ESP32 Cam

Kameramodul mit WLAN, OV2640 und microSD-Steckplatz

Kurzanleitung

ElektroXY · Bach IT Service UG (haftungsbeschränkt)

Lindenstr. 7 · 53506 Kesseling · www.elektroxy.de

Worum es geht

Das ESP32 Cam vereint einen ESP32 mit WLAN, eine 2-Megapixel-Kamera und einen Steckplatz für microSD-Karten auf einer Platine von der Größe einer Streichholzschachtel. Damit bauen Sie eine Überwachungskamera, eine Wildkamera mit Kartenaufzeichnung oder einen Türspion, der Bilder ins heimische Netz schickt — für einen Bruchteil dessen, was fertige Geräte kosten.

Ein Punkt vorweg, und er ist der wichtigste dieser Anleitung: Das Modul hat keinen USB-Anschluss. Sie können es nicht einfach anstecken und losprogrammieren. Es braucht einen zusätzlichen USB-Seriell-Adapter und eine bestimmte Verdrahtung. Genau daran scheitern die meisten ersten Versuche — nicht am Modul.

Wer sich diese Hürde sparen möchte, greift zum passenden Programmier-Shield (oft als ESP32-CAM-MB angeboten). Das Modul wird dort aufgesteckt, der Rest verhält sich dann wie bei jedem anderen Board mit USB-Buchse.

Technische Daten

Eigenschaft	Wert
Prozessor	ESP32, zwei Kerne, bis 240 MHz
Arbeitsspeicher	520 KB SRAM, zusätzlich 4 MB PSRAM
Programmspeicher	4 MB Flash
Kamera	OV2640, 2 Megapixel, bis 1600 × 1200 Pixel
Bildformate	JPEG, BMP, Graustufen
Speicherkarte	microSD, bis 4 GB zuverlässig, formatiert als FAT32
WLAN	802.11 b/g/n, 2,4 GHz
Bluetooth	vorhanden, wird aber selten genutzt
Antenne	Leiterbahn-Antenne oder externer Anschluss, per Lötbrücke wählbar
Versorgung	5 V, kurzzeitig über 300 mA
USB-Anschluss	nicht vorhanden

Was Sie zusätzlich brauchen

Weg	Zubehör	Aufwand
Programmier-Shield	ESP32-CAM-MB zum Aufstecken	gering — Modul aufstecken, USB-Kabel anschließen
USB-Seriell-Adapter	Adapter mit CP2102, CH340 oder FT232, auf 5 V eingestellt	sechs Drähte stecken, siehe nächster Abschnitt

Achten Sie beim Adapter darauf, dass er 5 V liefern kann. Viele Adapter haben dafür einen Jumper oder eine Lötbrücke. Die Signalleitungen arbeiten mit 3,3 V — das passt bei den gängigen Adaptern, deren TX/RX-Pegel der ESP32 verträgt.

Verdrahtung zum Programmieren

USB-Adapter	ESP32 Cam	Bemerkung
5V	5V	nicht an 3V3 anschließen
GND	GND	
TX	U0R	Adapter sendet, Modul empfängt
RX	U0T	Modul sendet, Adapter empfängt
—	GPIO 0 auf GND	nur zum Übertragen, danach wieder trennen

Die Brücke von GPIO 0 nach GND versetzt das Modul beim Einschalten in den Programmiermodus. Ohne sie bricht die Übertragung mit einer Zeitüberschreitung ab. Nach dem Übertragen nehmen Sie die Brücke wieder ab und drücken die Reset-Taste — erst dann startet Ihr Programm.

TX und RX werden über Kreuz verbunden. Wenn nichts passiert, ist ein Tausch dieser beiden Leitungen der erste Versuch, der sich lohnt.

Stromversorgung

Das Modul ist beim Stromhunger ehrlicher als die meisten: Sobald Kamera und WLAN gleichzeitig arbeiten, zieht es kurzzeitig deutlich über 300 Milliampere. Bricht die Spannung dabei ein, schaltet der ESP32 ab und startet neu. Im seriellen Monitor erscheint dann eine Meldung über den Brownout-Detektor.

Die 3,3-V-Ausgänge vieler USB-Adapter reichen dafür nicht. Versorgen Sie das Modul über den 5-V-Anschluss, verwenden Sie kurze Leitungen, und wenn es weiterhin neustartet, hängen Sie ein eigenes Netzteil mit mindestens 1 A an 5V und GND.

Ein Kondensator von 100 bis 470 μF zwischen 5V und GND direkt am Modul fängt Spannungseinbrüche ab und löst hartnäckige Neustart-Schleifen oft auf Anhieb.

Arduino-IDE einrichten

Tragen Sie in den Einstellungen unter „Zusätzliche Boardverwalter-URLs“ ein:

https://espressif.github.io/arduino-esp32/package_esp32_index.json

Danach im Boardverwalter „esp32“ installieren und folgende Einstellungen wählen:

Einstellung	Wert
Board	AI Thinker ESP32-CAM
Partition Scheme	Huge APP (3MB No OTA/1MB SPIFFS)
Upload Speed	115200
Port	der COM-Port des Adapters

Das Partitionsschema ist wichtig: Der Beispielcode für den Webserver passt nicht in die Standardaufteilung. Bleibt es dabei, bricht die Übertragung mit der Meldung ab, das Programm sei zu groß.

Erstes Programm

Die Espressif-Bibliothek bringt ein fertiges Beispiel mit, das den vollen Funktionsumfang zeigt — Livebild im Browser, Auflösung und Bildparameter einstellbar. Sie finden es unter Datei, Beispiele, ESP32, Camera, CameraWebServer.

Drei Stellen müssen Sie anpassen:

- Das Kameramodell auswählen: die Zeile mit CAMERA_MODEL_AI_THINKER aktivieren, alle anderen Modelle auskommentieren.
- WLAN-Name und Passwort eintragen.
- Übertragen, Brücke an GPIO 0 abnehmen, Reset drücken.

Im seriellen Monitor bei 115200 Baud erscheint danach die IP-Adresse. Rufen Sie sie im Browser auf und klicken Sie unten auf „Start Stream“.

Wenn Sie lieber klein anfangen, prüft dieser Ausschnitt nur, ob die Kamera überhaupt erkannt wird — nützlich zur Eingrenzung, bevor WLAN ins Spiel kommt:

```
#include "esp_camera.h"

// Pinbelegung des AI-Thinker-Moduls
#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() {
    Serial.begin(115200);
    delay(500);

    camera_config_t cfg = {};
    cfg.ledc_channel = LEDC_CHANNEL_0;
    cfg.ledc_timer = LEDC_TIMER_0;
    cfg.pin_d0 = Y2_GPIO_NUM;  cfg.pin_d1 = Y3_GPIO_NUM;
    cfg.pin_d2 = Y4_GPIO_NUM;  cfg.pin_d3 = Y5_GPIO_NUM;
    cfg.pin_d4 = Y6_GPIO_NUM;  cfg.pin_d5 = Y7_GPIO_NUM;
    cfg.pin_d6 = Y8_GPIO_NUM;  cfg.pin_d7 = Y9_GPIO_NUM;
    cfg.pin_xclk = XCLK_GPIO_NUM;  cfg.pin_pclk = PCLK_GPIO_NUM;
    cfg.pin_vsync = VSYNC_GPIO_NUM;  cfg.pin_href = HREF_GPIO_NUM;
    cfg.pin_sccb_sda = SIOD_GPIO_NUM;
    cfg.pin_sccb_scl = SIOC_GPIO_NUM;
    cfg.pin_pwdn = PWDN_GPIO_NUM;
```

```

cfg.pin_reset = RESET_GPIO_NUM;
cfg.xclk_freq_hz = 20000000;
cfg.pixel_format = PIXFORMAT_JPEG;
cfg.frame_size = FRAMESIZE_SVGA;    // 800 x 600
cfg.jpeg_quality = 12;
cfg.fb_count    = 1;

esp_err_t fehler = esp_camera_init(&cfg);
if (fehler != ESP_OK) {
    Serial.printf("Kamera meldet Fehler 0x%x\n", fehler);
    return;
}
Serial.println("Kamera erkannt.");

camera_fb_t *bild = esp_camera_fb_get();
if (bild) {
    Serial.printf("Bild aufgenommen: %u Bytes\n", bild->len);
    esp_camera_fb_return(bild);
}
}

void loop() {
}

```

Freie Anschlüsse und die SD-Karte

Die Kamera belegt den Großteil der Anschlüsse. Übrig bleiben die Pins an den beiden Stiftleisten — allerdings teilen sich SD-Karte und freie Pins dieselben Leitungen:

Pin	Doppelnutzung	Hinweis
GPIO 0	Takt der Kamera, Startmodus	nur zum Programmieren auf GND legen
GPIO 1, 3	serielle Schnittstelle	belegt, solange der Adapter hängt
GPIO 2, 4, 12 – 15	microSD-Karte	frei, wenn keine Karte genutzt wird
GPIO 4	zusätzlich die helle Blitz-LED	leuchtet sehr grell, blendet
GPIO 16	frei verwendbar	der bequemste freie Anschluss
GPIO 33	kleine rote Status-LED	verkehrt herum: LOW schaltet ein

Wer die SD-Karte nutzt, verliert also die meisten freien Pins — und die Blitz-LED an GPIO 4 leuchtet beim Kartenzugriff unfreiwillig mit. Im Programm lässt sich das abstellen, indem die SD-Karte im Ein-Bit-Modus angesprochen wird.

Fehlersuche

Beobachtung	Wahrscheinliche Ursache	Abhilfe
„Failed to connect“ oder Zeitüberschreitung	GPIO 0 liegt nicht auf GND	Brücke setzen, Reset drücken, Übertragung erneut starten
Übertragung läuft, Programm startet nicht	Brücke an GPIO 0 noch gesteckt	Brücke abnehmen und Reset drücken

Beobachtung	Wahrscheinliche Ursache	Abhilfe
Kamerafehler 0x105	Kamera nicht erkannt	Flachbandkabel prüfen, Verriegelung richtig schließen
Meldung über den Brownout-Detektor	Spannung bricht ein	Über 5V versorgen, kräftigeres Netzteil, Kondensator setzen
Modul startet immer wieder neu	Versorgung zu schwach oder Leitungen zu dünn	eigenes Netzteil mit mindestens 1 A, kurze Leitungen
Programm zu groß beim Übertragen	Falsches Partitionsschema	Auf „Huge APP“ umstellen
Nichts im seriellen Monitor	TX und RX vertauscht oder falsche Baudrate	Leitungen tauschen, 115200 einstellen
Bild ruckelt oder bricht ab	Auflösung zu hoch für die Funkverbindung	Im Webserver eine kleinere Auflösung wählen
WLAN-Reichweite sehr gering	Lötbrücke auf externe Antenne gesetzt, aber keine angeschlossen	Brücke auf die Leiterbahn-Antenne zurücklegen

Angaben ohne Gewähr. Alle Werte beziehen sich auf das Modul ESP32-CAM in der Ausführung mit OV2640-Kamera; maßgeblich sind die jeweils aktuellen Datenblätter von Espressif und OmniVision.