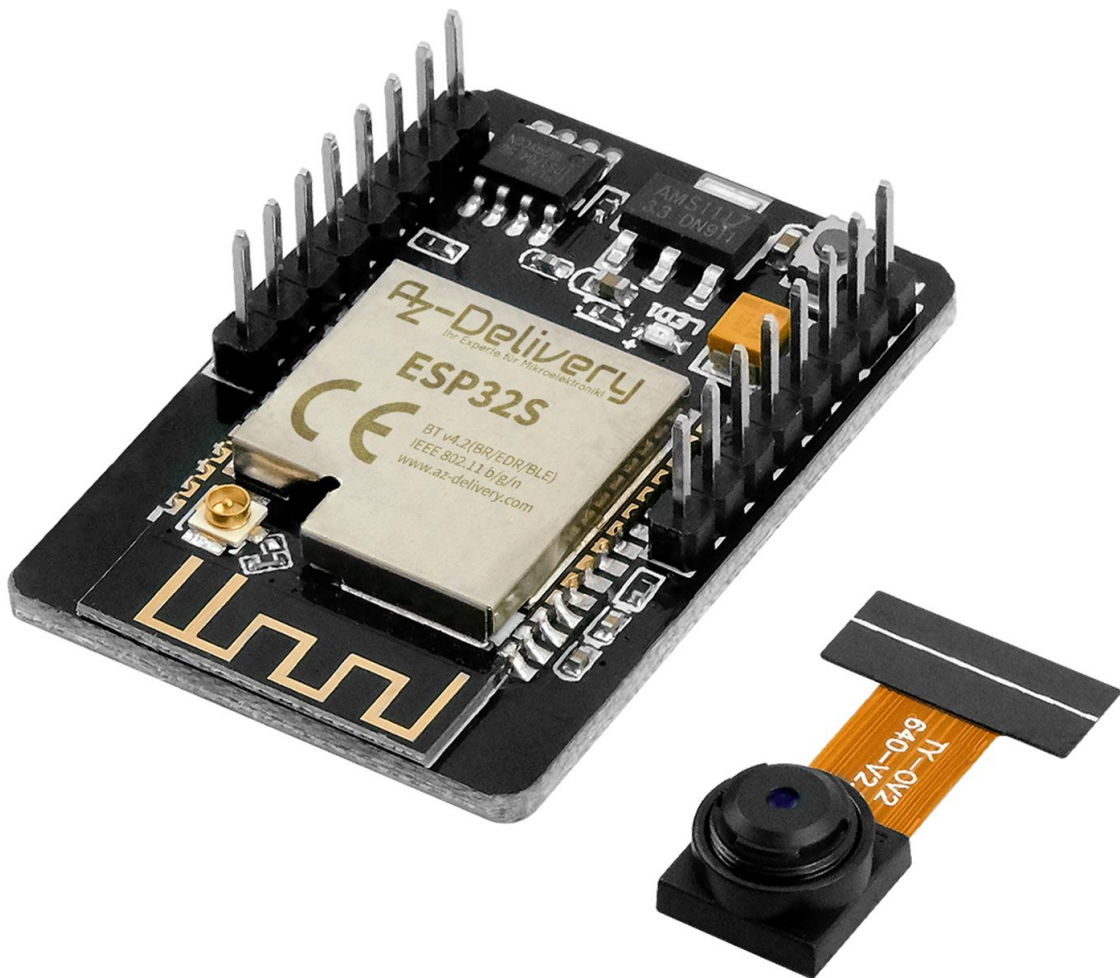


ESP32-Cam

ebook

ESP32-Cam

AI-Thinker Model



Inhaltsverzeichnis

Einführung	3
Spezifikationen	4
Pinout	5
Installation der Arduino IDE	6
Zusätzliche Einstellungen	10
Anschlussdiagramm	12
Blink Beispiel	13
Kamera Webserver Beispiel	14
Gesichtserkennung	16

Einführung

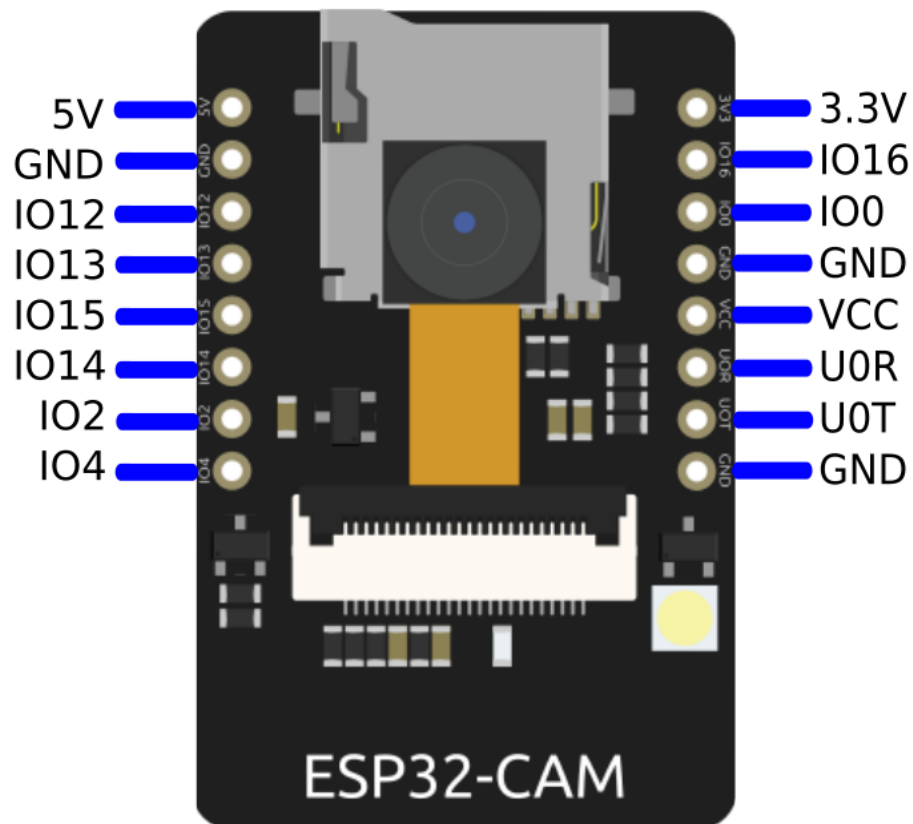
Die ESP32-Cam ist ein sehr wettbewerbsfähiges, kleines Kameramodul, das als Minimalsystem mit einer Baugröße von nur 27 x 40,5 x 4,2 mm unabhängig arbeiten kann.

Es kann in verschiedenen IoT-Anwendungen eingesetzt werden. Sie eignet sich für Smart-Home-Geräte, industrielle drahtlose Steuerung, drahtlose Überwachung, drahtlose QR-Identifikation und andere IoT-Anwendungen.

Spezifikationen

SPI Flash	4 MB
RAM	520KB SRAM + 4M PSRAM
Bluetooth	Bluetooth 4.2 BR/EDR and BLE standards
Wi-Fi	802.11 b/g/n
Support interface	UART, SPI, I2C, PWM
Support TF card	Max. 4GB
IO port	9
UART Baudrate	Default 115200 bps
Image Output Format	JPEG (OV2640 support only), BMP, GRAYSCALE
Spectrum Range	2412~2484 MHz
Antenna	Onboard PCB antenna, gain 2dBi
Transmit Power	802.11b: 17±2 dBm (@11Mbps) 802.11g: 14±2 dBm (@54Mbps) 802.11n: 13±2 dBm (@MCS7)
Receiving Sensitivity	CCK, 1 Mbps: -90 dBm CCK, 11 Mbps: -85 dBm 6 Mbps (½ BPSK): -88 dBm 54 Mbps(¾ 6-QAM): -70 dBm MCS7(65 Mbps, 72.2 Mbps): -67 dBm
Power Supply Range	5V
Operating Temperature	-20°C ~ 85°C
Dimensions	27 x 40.5 x 4.5 mm

Pinout



Hinweis:

Das Modul kann entweder über den 3,3V pin oder über den 5V pin versorgt werden.

Es wird empfohlen das Modul über den 5V pin(mit 5V) zu betreiben.

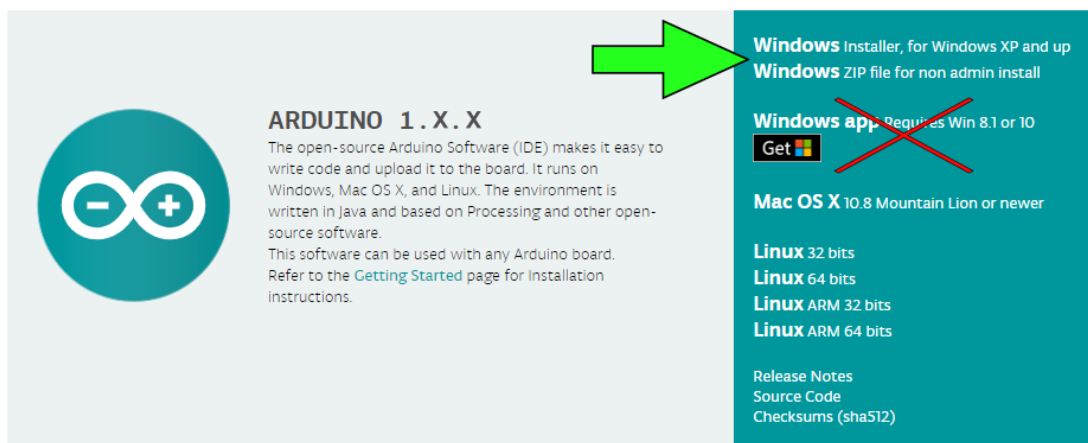
Der VCC pin ist ein 3,3V ausgang.

Installation der Arduino IDE

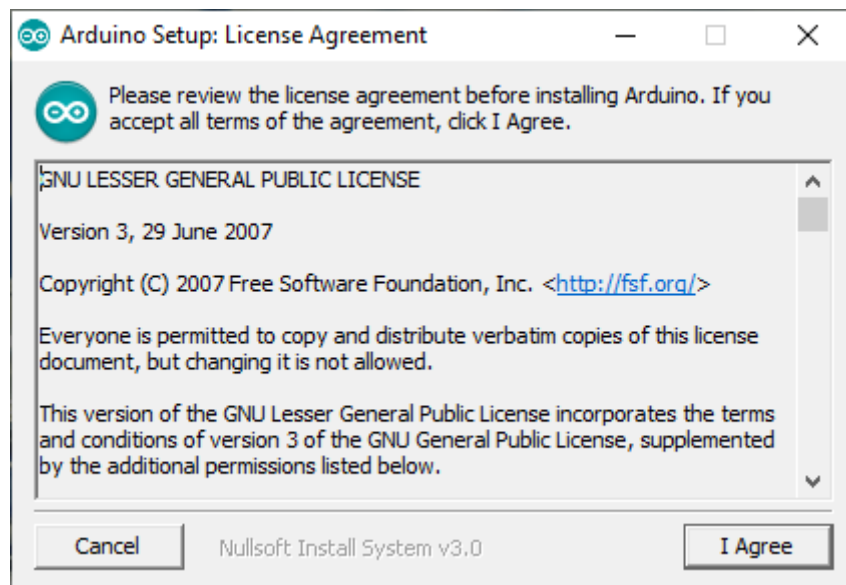
Die kostenlose Arduino IDE Entwicklungsumgebung können Sie unter folgendem Link herunterladen: <https://www.arduino.cc/en/Main/Software>

Windows Benutzer sollten unbedingt eine der ersten beiden Download-Optionen für die Arduino IDE verwenden. Die "Windows App" Version aus dem Windows Store führt insbesondere bei der Verwendung von Board-Definitionen von Drittanbietern zu Verbindungsproblemen.

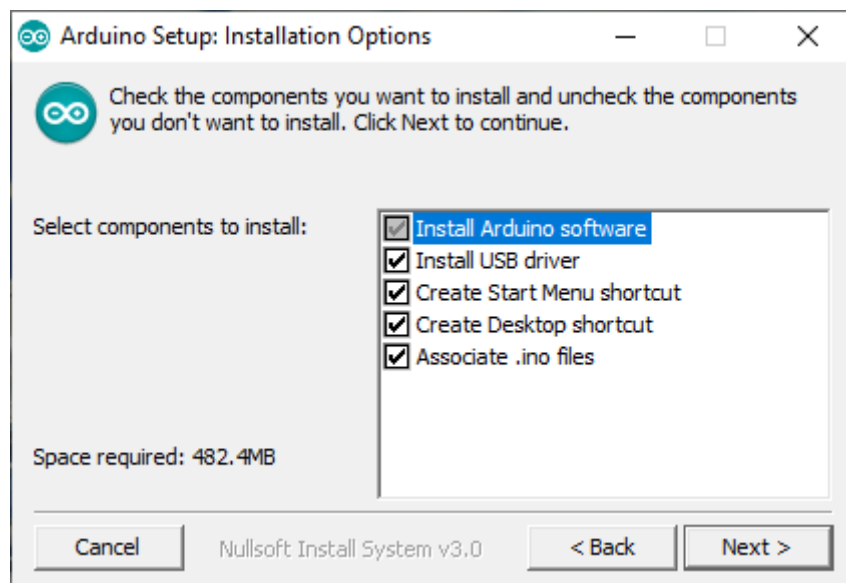
Download the Arduino IDE



Nach dem Starten der Arduino IDE Installationsdatei "arduino-1.X.X-windows.exe" müssen die Lizenzbedingungen der Software gelesen und akzeptiert werden:



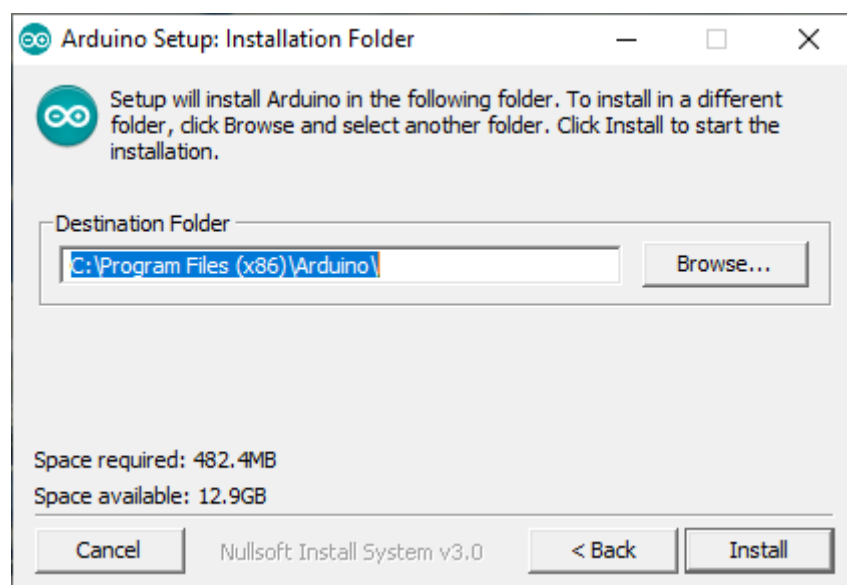
Im nächsten Schritt können unterschiedliche Optionen zur Installation ausgewählt werden.



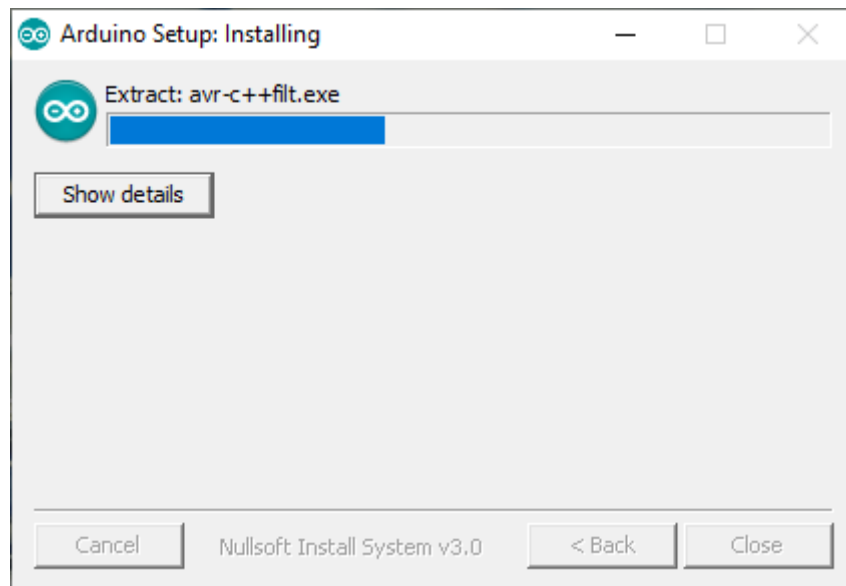
Es folgt eine kurze Übersicht der verschiedenen Optionen, mit einer kurzen Erklärung zu jeder Option:

Option	Erklärung
Install Arduino Software	Installiert die Arduino IDE - Diese Option kann nicht abgewählt werden
Install USB Driver	Installiert USB Treiber für verschiedene andere Microcontroller. Für die Verwendung der Software mit dem D1 mini sind diese nicht erforderlich, wir empfehlen jedoch unbedingt die Installation, falls Sie auch andere Mikrocontroller verwenden
Create Start Menu shortcut	Erstellt eine Verknüpfung im Windows Startmenü (Optional)
Create Desktop shortcut	Erstellt eine Verknüpfung auf dem Arbeitsplatz (Optional)
Associate .ino files	Erstellt eine Dateinamenerweiterung für Dateien mit der Endung .ino und verknüpft diese mit der Arduino IDE

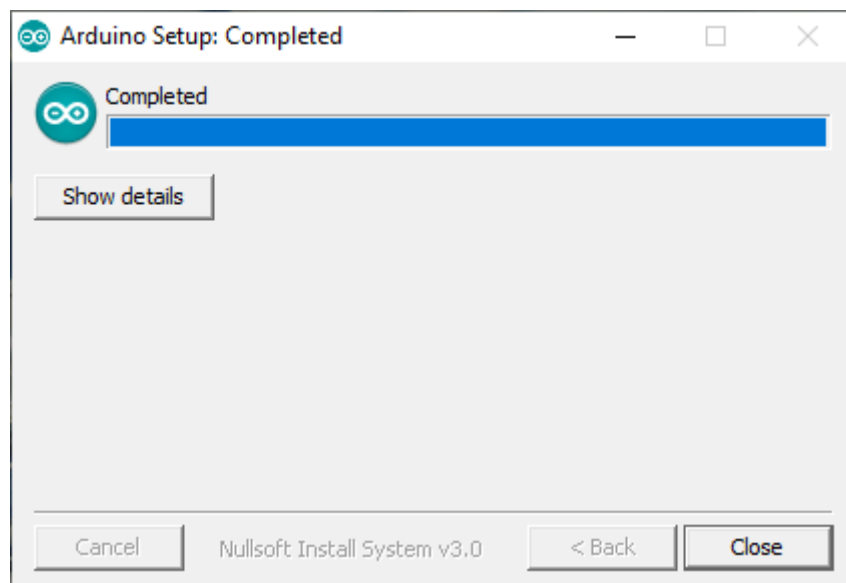
Zuletzt muss noch der Zielordner angegeben werden. Die Installation benötigt ca. 500MB freien Speicherplatz.



Klicken Sie auf "Install", um die Installation zu starten.



Nach erfolgreicher Installation kann das Installationsprogramm über die Schaltfläche "Close" beendet werden:



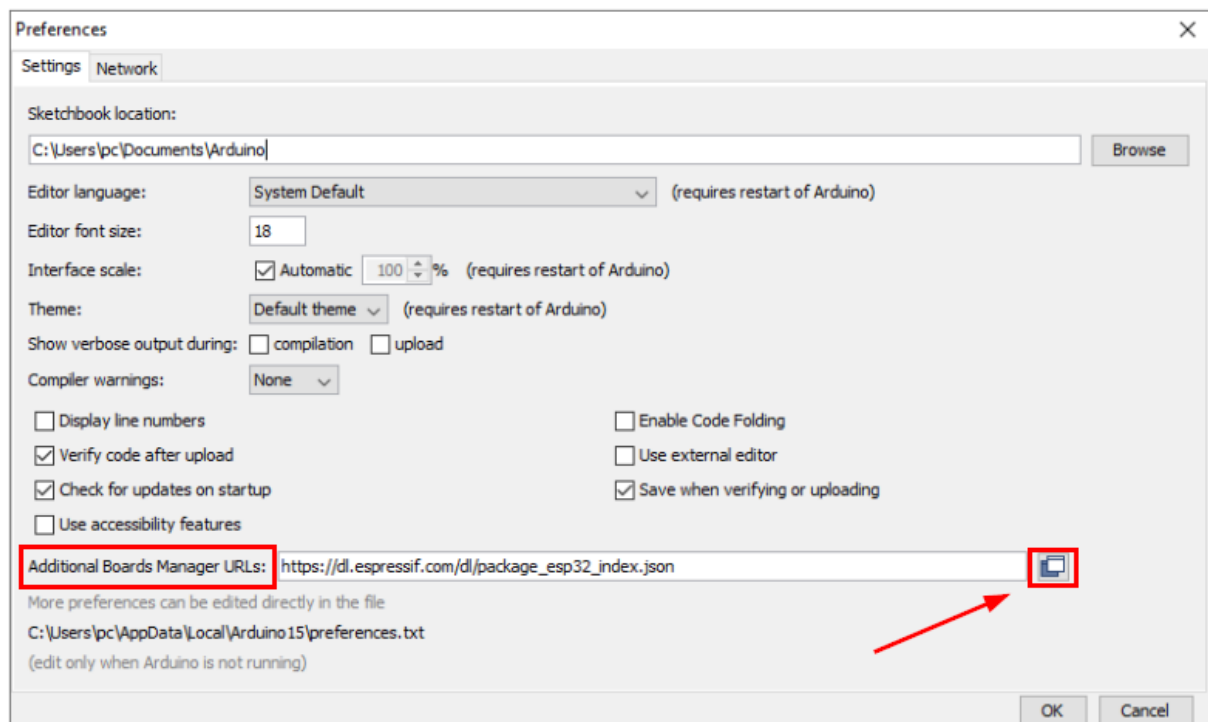
Zusätzliche Einstellungen

Um die ESP32-Cam in der Arduino IDE programmieren zu können müssen Sie den Support für die ESP32-Plattform installieren.

Öffnen Sie die Arduino IDE und gehen Sie zu: File > Preferences, und finden Sie das Feld "Additional URLs".

Kopieren Sie folgenden Link:

https://dl.espressif.com/dl/package_esp32_index.json



Fügen Sie diesen Link in das Feld "Additional URLs" ein. Wenn Sie bereits einen oder mehrere Links in diesem Feld haben, fügen Sie einfach ein Komma nach dem letzten Link ein, fügen Sie den neuen Link nach dem Komma ein und klicken Sie auf die Schaltfläche OK. Schließen Sie dann die Arduino-IDE.

ESP32-Cam

Öffnen Sie die Arduino IDE erneut und gehen Sie zu:

Tools > Board > Boards Manager

Es öffnet sich ein neues Fenster, geben Sie esp32 in das Suchfeld ein und installieren Sie das Board mit dem Namen esp32 von Espressif Systems, wie unten abgebildet:

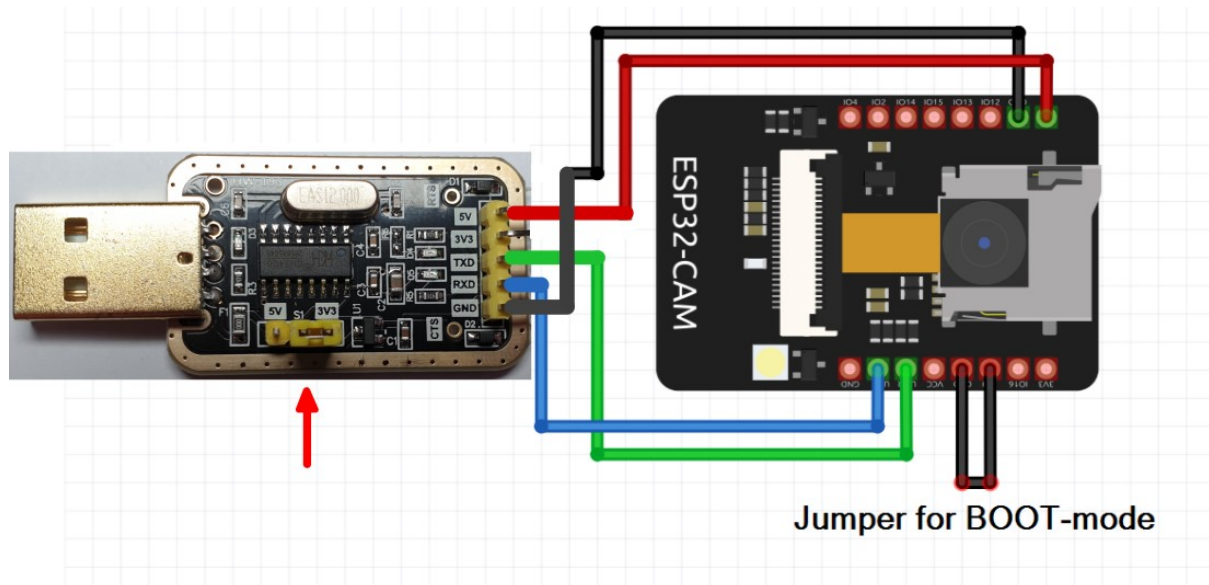


Jetzt können Sie unter Tools > Board > ESP32 Arduino > AI-Thinker ESP32-CAM auswählen.

Anschlussdiagramm

Um die ESP32-Cam flashen zu können benötigen Sie einen USB-to-UART Adapter, stellen Sie sicher dass der dafür nötige Treiber installiert ist.

Folgend Abbildung zeigt den Anschluss der ESP32-Cam mit dem AZ-Delivery CH340 Adapter(es kann auch ein anderer Adapter verwendet werden). Der Jumper für den **Logikpegel** muss auf **3,3V** gesetzt werden da ansonsten die pins am ESP Modul beschädigt werden.



CH340	ESP32-Cam
VCC	5V
GND	GND
TX	U0R
RX	U0T

Um die ESP32-Cam in den BOOT-Modus zu setzen wird ein Jumper zwischen IO0 und GND benötigt.

Hinweis: Die ESP32-Cam besitzt keinen verpolschutz, achten Sie darauf 5V und GND nicht zu vertauschen da ansonsten das Modul zerstört wird.

Blink Beispiel

Als erstes Beispiel eignet sich Blink ganz gut. Auf der ESP32-Cam befindet sich eine flash-light LED, diese ist mit GPIO 4 verbunden.

```
#define flash 4
void setup() {
  pinMode(flash, OUTPUT);
}

void loop() {
  digitalWrite(flash, HIGH);
  delay(1000);
  digitalWrite(flash, LOW);
  delay(1000);
}
```

Wählen Sie unter Tools > Board > ESP32 Arduino > AI-Thinker ESP32-CAM, und den Port des Adapters aus. Klicken Sie anschließend auf “Hochladen”.

Wenn das Hochladen erfolgreich war können Sie den Jumper zwischen IO0 und GND entfernen und anschließend die RESET-Taste betätigen. Seien Sie gewarnt, **die Flash-light LED ist sehr hell.**

Kamera Webserver Beispiel

Mit diesem Beispiel kann man sich einen Video Stream auf dem Webinterface anzeigen lassen. Um das Beispiel zu öffnen gehen Sie auf
File > examples > ESP32 > camera > CameraWebServer

Ab Zeile 10 soll man das Kamera Modell wählen. Hier muss das Model AI-Thinker gewählt werden. Passen Sie die Zeilen an, es sollte wie folgt aussehen:

```
// 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
// #define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
// #define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
// #define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM
#define CAMERA_MODEL_AI_THINKER // Has PSRAM
// #define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
```

In Zeile 23 und 24 müssen Sie noch die SSID und Passwort ihres Routers eintragen. z.B.:

```
const char* ssid = "hier-SSID";
const char* password = "Hier-Passwort";
```


Anschließend kann dieses Beispiel Hochgeladen werden. Sobald das hochladen erfolgreich war, entfernen Sie den jumper zwischen IO0 und GND und betätigen anschließend die reset Taste. Öffnen Sie den Seriellen Monitor der Arduino IDE hier sollte die IP-Adresse ausgegeben werden unter der die ESP32-Cam erreichbar ist.

Geben Sie die IP-Adresse der ESP32-Cam in Ihren Browser ein(Sie müssen im gleichen Netzwerk sein). Jetzt befinden Sie sich auf dem Webinterface der ESP32-Cam. Es gibt diverse Einstellmöglichkeiten mit denen man sich spielen kann. Unten gibt es den Button "Start Stream" mit dem können Sie die Videoübertragung starten.

Gesichtserkennung

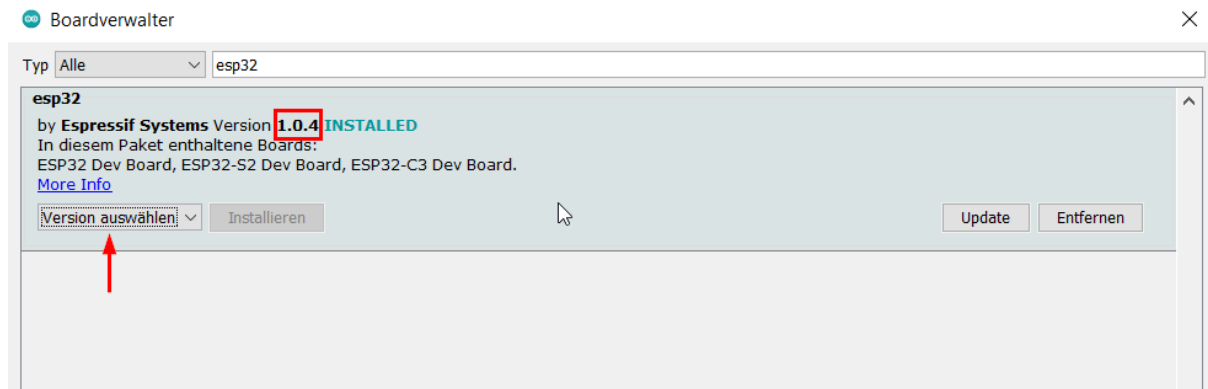
Das vorherige Beispiel liefert auch die Funktion der Gesichtserkennung. Wenn der Stream läuft kann man die Option "Face detection" aktivieren, in der Videoansicht wird ein gelbes Rechteck um das Gesicht gelegt. Mit der Option "Face recognition" und dem Button "Enroll Face" kann ein Gesicht abgespeichert werden.

Nutzen Sie die aktuelle Version der ESP32 Boardinformationen (zur Zeit 2.0.2) funktioniert die Gesichtserkennung nicht. Diese Funktion sollte in Zukunft wieder zur Verfügung stehen.

Um die Gesichtserkennung dennoch nutzen zu können müssen Sie die Version downgraden(eine ältere Version installieren). Das können Sie machen indem Sie auf Tools > Board > Boardmanager gehen. geben Sie "esp32" in die Suchleiste ein.

ESP32-Cam

Klicken Sie auf “Version auswählen” und wählen Sie die Version 1.0.4 aus und klicken auf “installieren”, starten sie anschließend die Arduino IDE neu



Ist die Arduino IDE neu gestartet müssen Sie das Beispiel neu öffnen. File > Examples > ESP32 > camera > CameraWebServer

Nehmen Sie die gleichen Einstellungen vor wie im vorherigen Beispiel und laden es auf die ESP32-Cam.

Die Gesichtserkennung funktioniert jetzt.

Jetzt ist es an der Zeit, zu lernen und eigene Projekte zu erstellen. Das können Sie mit Hilfe von vielen Beispielskripten und anderen Tutorials tun, die Sie im Internet finden können.

Wenn Sie auf der Suche nach den qualitativ hochwertigen Produkten für Arduino und Raspberry Pi sind, sind Sie bei der AZ-Delivery Vertriebs GmbH genau richtig. Sie erhalten zahlreiche Anwendungsbeispiele, vollständige Installationsanleitungen, eBooks, Bibliotheken und Unterstützung durch unsere technischen Experten.

<https://az-delivery.de>

Viel Spaß!

Impressum

<https://az-delivery.de/pages/about-us>