

"Spionage"-Video-Stream aufs Smartphone holen - mit ESP32-CAM und zwei NFC-Tags schnell gemacht

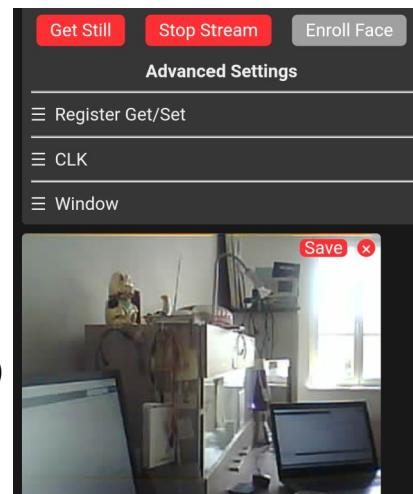
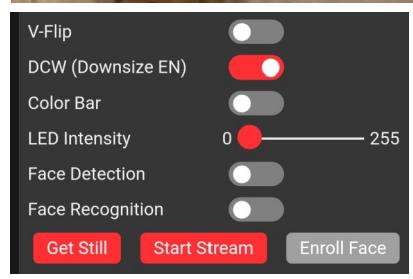
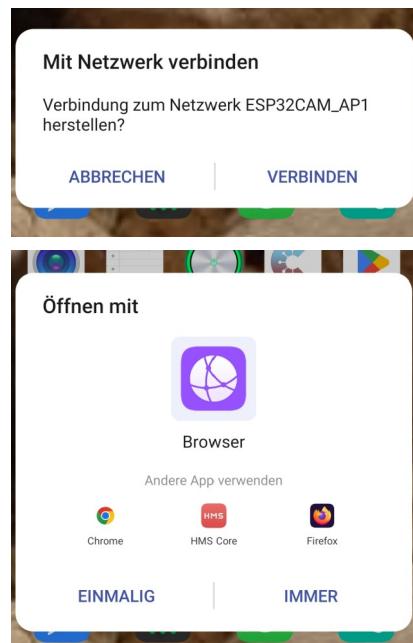
Es muss eine passend programmierte ESP32-Kamera in der Nähe und eingeschaltet sein!



Hier sollte das **NFC-Tag für das Netzwerk** aufgeklebt sein:



Hier sollte das **NFC-Tag für die Seite mit dem Videostream** aufgeklebt sein:

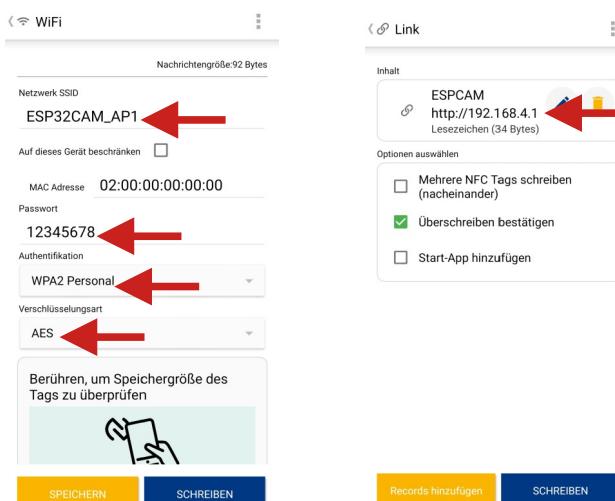


Eine praktische Hilfestellung für den Betrieb einer "Spionage-Kamera" bietet die Kombination einer ESP32-CAM mit zwei NFC-Tags. Zunächst zeigen wir, wie du Zugang zur Kamera bekommst, die über eine USB-Netzteil o.Ä. mit Strom versorgt ist und nicht an einen Rechner angeschlossen sein muss. Sie wurde von uns entsprechend vorbereitet.

Geh mit dem Smartphone über den Tag für die Netzverbindung und bestätige, dass du dein Phone mit dem Netzwerk "ESP32CAM-AP1" verbinden willst.

Anschließend suche den Tag mit der Seite "<http://192.168.4.2>", das ist die Seite mit dem Videostream aus der Kamera. Du musst noch wählen, mit welchem Browser du die Seite öffnen willst. Sobald du die Bedienelemente siehst, bist du auf der richtigen Seite. Scrolle nun soweit hinunter, bis du den Button "Start Stream" siehst und tippe darauf. Unter dem Button erscheint dann das Livebild, das die Kamera gerade aufnimmt, je nachdem, wohin sie ausgerichtet ist.

Hinweise zur Programmierung der NFC-Tags für das Netzwerk (unten links) und die Seite mit den Videostream mit dem NXP-Tagwriter (unten rechts):



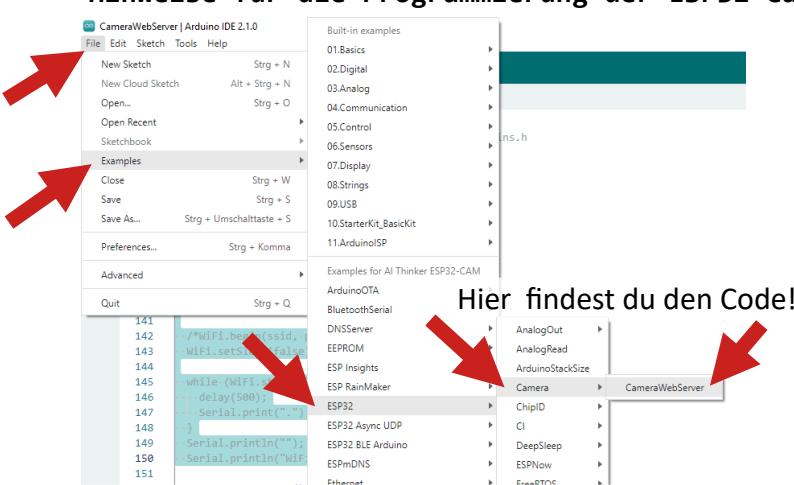
Falls du dir selbst ein ESP32-Kamera-Modul als Access-Point (also mit eigenen kleinen WLAN) einrichten möchtest, findest du auf der nächsten Seite einige Infos, die eventuell hilfreich sind. Insgesamt ist die Prozedur kompliziert, fordert Konzentration und Frustrationstoleranz. Du solltest du dich schon ziemlich gut mit Arduino eingearbeitet haben – das Ganze ist nicht so leicht zu durchschauen ...

Eine ESP32-Kamera als Access-Point einrichten

Um einen ESP32-Controllerboard mit Kamera (wie an dieser Station gezeigt mit der Typenbezeichnung ESP32-CAM) einrichten zu können, musst du es wahrscheinlich bestellen, aktuell kostet das etwa 12 € (Mai 2023). Dafür erhältst du das Board einschließlich Kamera und Programmer, also insgesamt drei Teile. Du solltest bei der Bestellung darauf achten, dass der ESP32 bereits auf einen Programmer aufgesteckt ist, ansonsten musst du letzteren zusätzlich besorgen und anschließen. Er ist notwendig, um den Code über die USB-Schnittstelle des Rechners hochladen zu können. (Das ist anders als bei einem Arduino Uno, bei dem der Programmer schon mit eingebaut ist.)

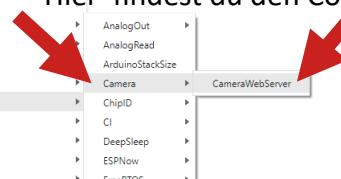
Außerdem benötigst du natürlich einen PC mit der Arduino-Entwicklungsumgebung sowie das passende USB-Kabel. Das Arduino-Programm, auch Sketch genannt, mit dem wir hier arbeiten, findest du erst dann, wenn du deine Arduino-Entwicklungsumgebung das ESP32-Add-on installiert hast. (Hinweise dazu findest du im Internet.) Den Sketch "CameraWebServer" findest du nämlich nur unter den Beispielen für den ESP32. Im Code musst du einige Änderungen vornehmen, damit alles zu diesem Beispiel passt.

Hinweise für die Programmierung der ESP32-Cam:



1. Im Code muss du
"CAMERA_MODEL_AI_THINKER"
wählen!

Hier findest du den Code!



2. In den Code müssen der Netzwerkname und das Passwort statt der Sterne eingetragen werden:

```
const char* ssid = "ESP32CAM_AP1";
const char* password = "12345678";
```

3. Folgender Code-Abschnitt wird auskommentiert:

```
/*WiFi.begin(ssid, password);
WiFi.setSleep(false);

while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}

Serial.println("");
Serial.println("WiFi connected");*/
```

6. Beim Hochladen des Codes über einen "Aufsteck-Programmer" (Dockingstation HK-ESP32) muss der IO0-Taster auf der HK-Platine gedrückt sein und währenddessen zusätzlich kurz der Reset-Taster auf der ESP-Platine gedrückt werden. Wird mit einem sogenannten FTDI-Programmer hochgeladen, muss währenddessen der PIN IO0 mit GND verbunden sein. Auch hier muss kurz der Reset-Button auf dem Board gedrückt werden. Nach Ende des Uploads muss die PinIO-Ground-Verbindung gelöst werden. Nach erfolgreichem Upload muss dann nochmals Reset auf der ESP32 Platine gedrückt werden.

4. Diese Zeilen müssen im Sketch dort eingefügt werden, wo der Abschnitt links auskommentiert wurde:

```
// We set the ESP32 as an access point
WiFi.softAP(ssid, password);

IPAddress IP = WiFi.softAPIP();
Serial.print("ACCESS POINT IP address: ");
Serial.println(IP);
```

5. Vor dem Hochladen des Sketches muss auch hier "CAMERA_MODEL_AI_THINKER" gewählt werden!



7. Hat das Hochladen funktioniert, erscheint im Monitor diese Meldung und der Videostream sollte zugänglich sein.

```
load:0x40078000,len:13924
ho 0 tail 12 room 4
load:0x40080400,len:3600
entry 0x400805f0
E (493) esp_core_dump_ffff: No core dump partition found!
E (494) esp_core_dump_flash: No core dump partition found!

ACCESS POINT IP address: 192.168.4.1
Camera Ready! Use 'http://0.0.0.0' to connect
```