

# RpiZero2W als YouTube Headlesstreamer

Der Grund diesen Tutorials ist das mit einem Raspberry Pi Zero2W allein als Streamer die Zuschauer qualitativ auf Einen begrenzt sind. Mit einem YouTube- Kanal lässt sich die Anzahl an Zuschauern erhöhen und deren Management ist mittels des YouTube-Webinterfaces recht komfortabel.

Es wird beschrieben wie Ihr einen YouTube-Account anlegt und einen Raspberry Pi Zero2W mit angeschlossenem csi- Kameramodul als konfigurationsfreie YT-Kamera nutzen könnt. Nach dem Anlegen des YT-Accounts müsst Ihr 24 Stunden Warten bis Ihr diesen zum streamen nutzen könnt, deshalb solltet Ihr gleich damit beginnen

## Vorraussetzungen

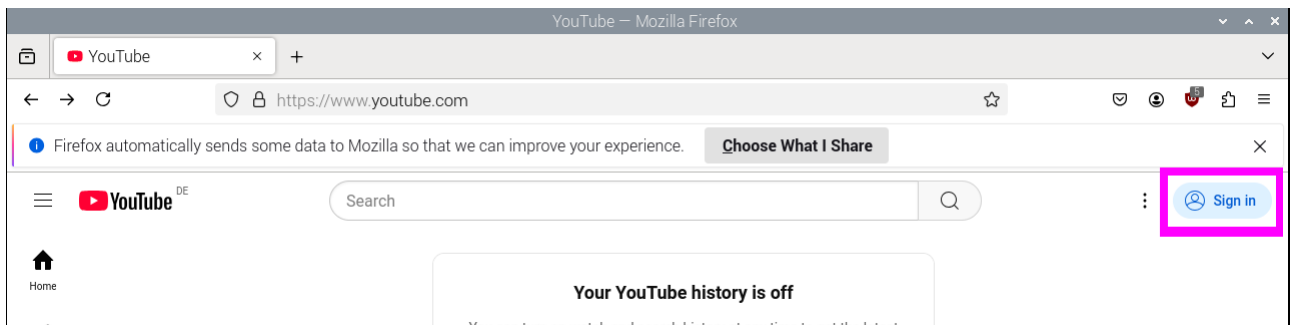
- Eine gültige gmail-Mailadresse
- Ein Handy für die Bestätigungs-SMS
- Internetzugang/Browser
- Einen RPiZero2W mit RaspbianOS Bookworm und ssh- oder Terminallogin
- Ein bereits mittels csi-Anschlusskabel dem RPiZero2W korrekt verbundenes Kameramodul

## Vorgehensweise

Zuerst legen wir ein YouTube- Account und Kanal an,  
dann warten wir 24 Stunden,  
um im Anschluss einen Livestream im bereits angelegten Kanal zu erstellen  
damit wir den Installations- und Konfigurationsprozess auf dem Raspi starten,  
und dann das Livebild des Kameramoduls zu sehen und im Kanal nach belieben veröffentlichen.

## YouTube- Account und Kanal anlegen

Wir öffnen unserer Browser und geben in dessen Sucheingabe die Internetadresse youtube.com ein,  
und klicken wie im Bild unten magentafarben markiert zu sehen ist auf „Sign in“



Damit wir hier unsere gültige E-Mail und darauf folgend deren Paasswort eingeben

Email or phone

[Forgot email?](#)

Enter your password


☐ Show password

Not your computer? Use a Private Window to sign in. [Learn more about using Guest mode](#)


[Create account](#) [Next](#) [Forgot password?](#) [Next](#)


Google

### Recovery information

 momefilo.versuch@gmail.com

Verify your phone number and email so that Google can help you if you forget the password to your account. [Learn more](#)

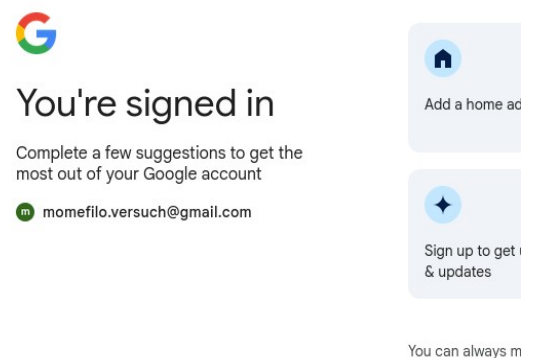
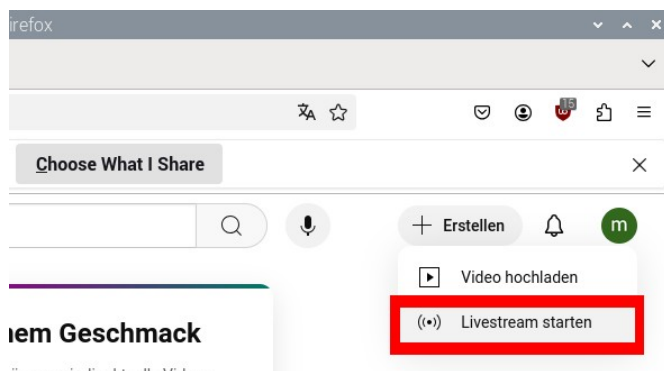
 Add a recovery phone



Enter phone


Hier genügt es unsere Handynummer einzugeben und den Dialog zu bestätigen.

Und dann wie im rechten Bild auf den rot markierten „not now“ Button klicken um, wie im Unteren Bild rot markiert zu sehen, auf „Livestream starten“ klicken



Den darauf folgend, unten links abgebildeten Dialog bestätigen wir einfach mit „Kanal erstellen“

Dein Profil



[Bild auswählen](#)

Name


momefilo

Alias

@momefilo

Wenn du auf „Kanal erstellen“ klickst, erklärst du dich mit den Nutzungsbedingungen von YouTube einverstanden. Änderungen an deinem Namen und Profilbild sind nur auf YouTube und nicht bei anderen Google-Diensten sichtbar. [Weitere Informationen](#)

[Abbrechen](#) [Kanal erstellen](#)



### Zugriff auf das Livestreaming beantragen

Nach Anfrage kann es bis zu 24 Stunden dauern, bis du das Livestreaming nutzen kannst

[Anfrage](#)

### Zugang zu dieser Funktion erhalten

Du musst zuerst deine Telefonnummer bestätigen, um Zugang zum Livestreaming am Computer zu erhalten.

[Schließen](#) [Bestätigen](#)

Damit wir eine Anfrage zum Livestreamen durch klick stellen und bestätigen.

In einem neu erscheinenden Fenster geben wir unsere Telefonnummer ein und erhalten unmittelbar einen Code auf unser Handy gesendet, den wir im nächsten Dialog eingeben und ebenfalls bestätigen

🔍 Livestreaming - YouTube x Telefonbestätigung

← → ↻ 🔒 https://www.youtube

**YouTube**

**Telefonbestätigung (Schritt 1 von 2)**  
Wenn du gern noch mehr Funktionen auf YouTube nutzen möchtest

Wie sollen wir dir den Bestätigungscode zukommen lassen?

☒ Ich möchte den Bestätigungscode per SMS erhalten

☐ Ich möchte eine automatisierte Sprachnachricht erhalten

Land auswählen

Deutschland ▾

Wie lautet deine Telefonnummer?

(201) 555-5555

Wichtig: Pro Telefonnummer können im Jahr bis zu zwei Kanäle bestätigt werden.

WEITER

🔍 Livestreaming - YouTube x Telefonbestätigung

← → ↻ 🔒 http

**YouTube**

**Telefonbestätigung (Schritt 2 von 2)**  
Der Bestätigungscode wurde per SMS an 0176:

Solltest du keine SMS erhalten, gehe zurück und

Gib deinen 6-stelligen Bestätigungscode ein.

ZURÜCK SENDEN

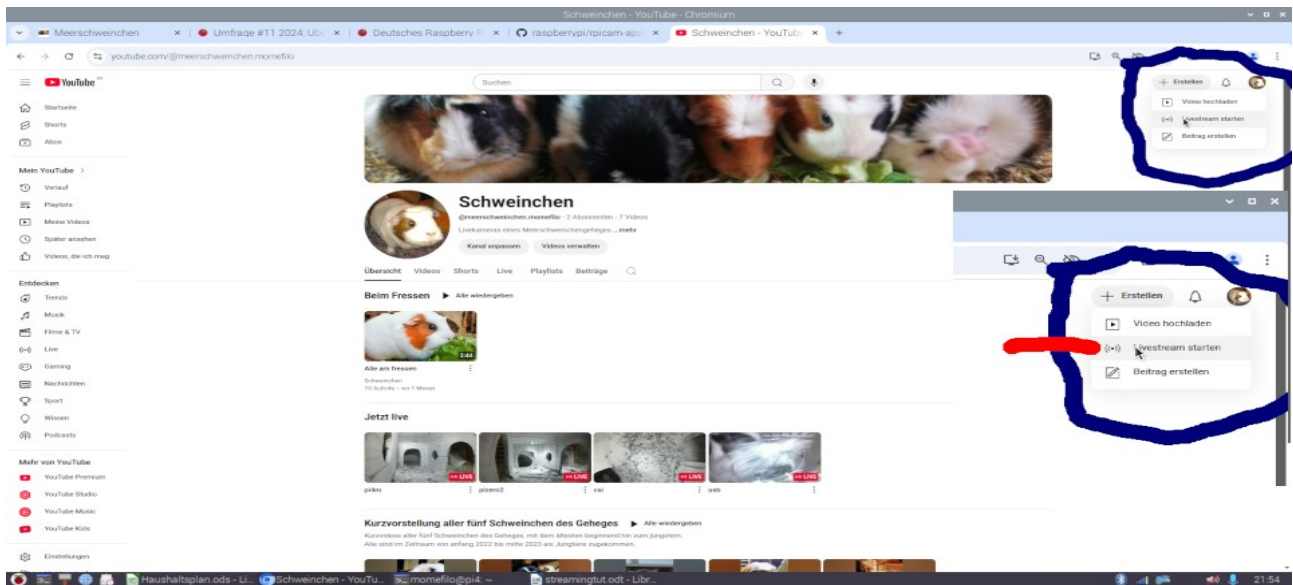
*Bilder: Heute letzter schritt, die Telefoncode-Bestätigung*

Nun warten wir 24 Stunden

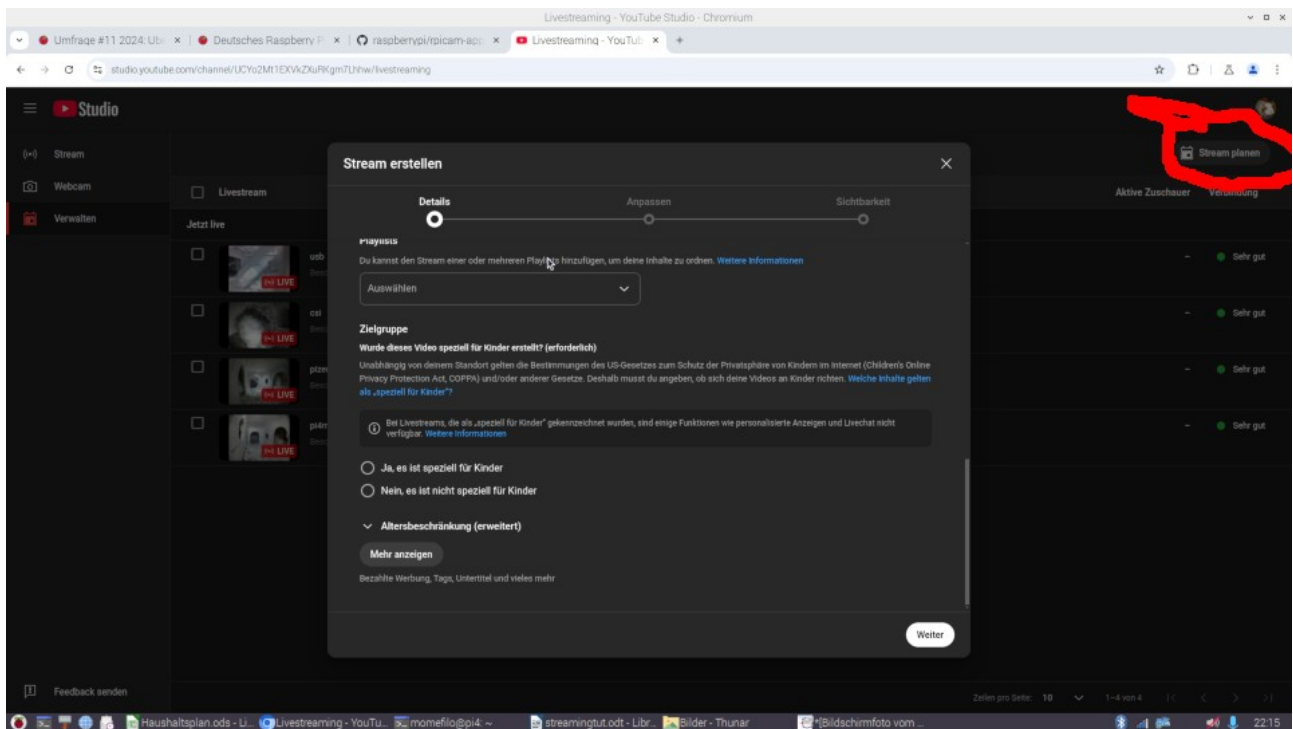


# YouTube-Benutzerkonto-konfiguration

Wir navigieren mit einem Browser zu unserem Yutubekonto unter der Adresse „[youtube.com/@dein\\_youtube\\_benutzername](https://youtube.com/@dein_youtube_benutzername)“ und starten mit zwei klicks, wie im Bild markiert zu sehen den Einstiegsdialog zur Livestreamkonfiguration.

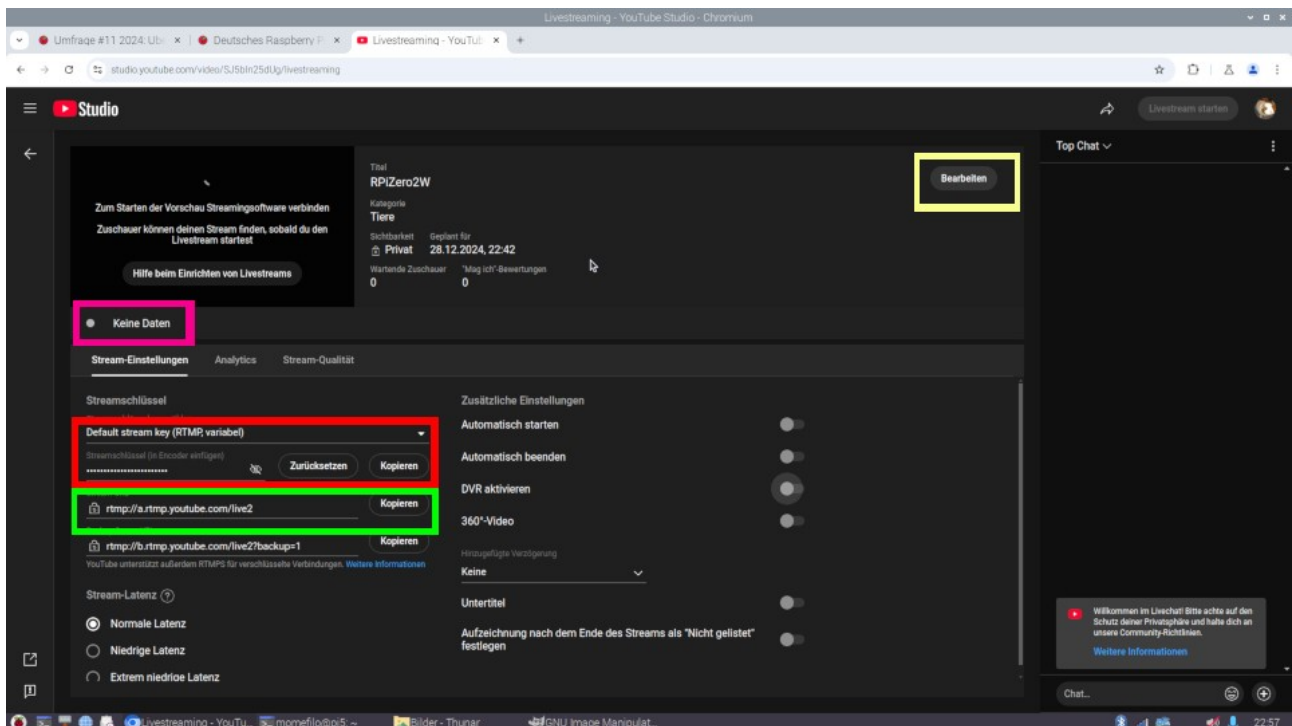


Sollte der Dialog nicht automatisch starten muss man im rot Markierten Bereich des unteren Bildes auf „Stream planen“ klicken



In diesem Dialog werden selbsterklärende Dinge erfragt dies sich nach dessen Abschluss noch vor dem Start des Streams, und viele auch danach noch bearbeiten lassen. Wie z.B. die Option den Stream auch eingebettet in Internetseiten wiederzugeben, welche beim Start unabhängig ihrer Auswahl jetzt, immer deaktiviert ist und danach geändert werden kann.

Wenn wir mit diesem Dialog fertig sind gelangen wir zu dem uns wichtigen Teil der Konfiguration in dem wir die Zieladresse des Streams erfahren. **Wir benötigen nur den rot markierten Teil und Kopieren uns den in ein Textdokument**



Der magentafarben markierte Punkt zeigt uns uns in vier Stufen bis zu „sehr gute Qualität“ an das wir keine Daten senden. Gelb markiert die Möglichkeit jene Einstellungen des Einstiegsdialogs zu bearbeiten. Interessant zu wissen das die Latenz nach dem Start des Streams nicht mehr geändert werden kann.

Das lassen wir nun einfach so wie es ist denn das benötigen gleich wieder zur Überprüfung, und wenden uns dem Terminal zu, das uns mit blinkendem Cursor die Bereitschaft des RPiZero2W zur Befehlseingabe signalisiert

## Installation/Konfiguration auf dem Raspi Starten

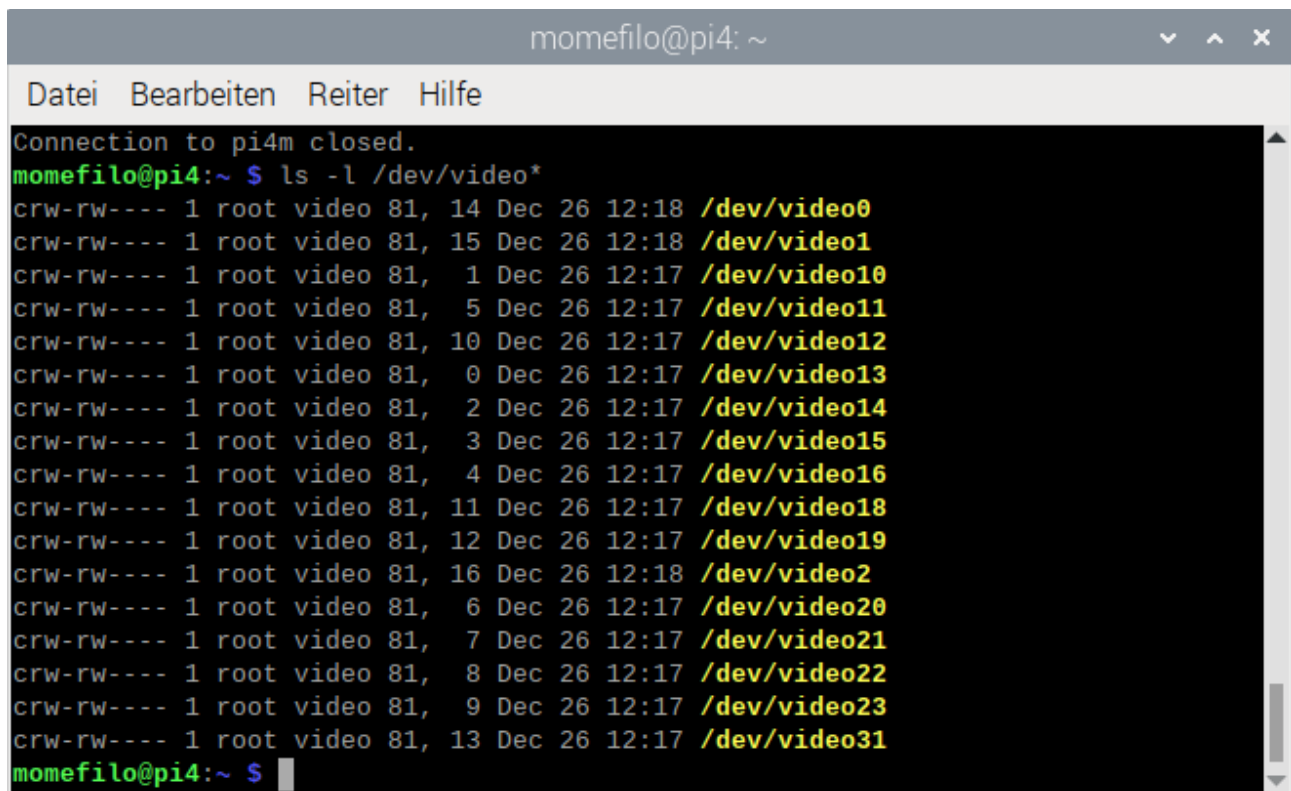
Zunächst allgemein, die bereits Vorinstallierte Software `rpivid` führt die Sensordaten des Kameramoduls über einen Teilbereich des Prozessors, welcher besonders geeignet ist Datenströme in Echtzeit zu bearbeiten. Wir werden anweisen den Datenstrom in das etablierte Format `h264`, welches unserem zu vor konfiguriertem YouTube-Streamingdienst genügt um zuwandeln. Den umgewandelten Datenstrom leiten wir an die von uns installierte Software `ffmpeg` weiter, welche diesem einen Audiodatenstrom hinzufügt der für das Streamen über YouTube unerlässlich ist. Der Einfachheit wegen ist dieser stumm, kann aber ebenso einfach durch eigene Recherche ersetzt werden. Auch das Senden des Streams über das Internet mittels RTMP, ein Echtzeitnachrichten Protokoll erledigt `ffmpeg`. Die Software `rpivid` und `ffmpeg` wird durch Aufruf einer Befehlszeile in einer ausführbaren Datei mit dem Systemdienst `systemd` zur Ausführung gebracht, mit dessen Konfiguration wir nun beginnen

## Gerätedatei ermitteln

Um zu wissen das unsere Kamera als solche vom System erkannt ist, schauen wir zunächst in dessen Gräteverzeichnis nach Einträgen der Form „videoX“, wobei das X für eine beginnend von 0 fortlaufenden Ziffer steht die jeweils einem erkanntem Videogerät zugewiesen wird; Wichtig ist zu wissen das manche Kameras zwei videoX Gerätedateien zugewiesen wird von denen die erste immer den Videodatestrom führt. Ist neben dem csi-Kameramodul z.B. noch eine USB-Kamera angeschlossen, so erzeugt die eingabe folgendes Befehls in die Eingabeaufforderung des Terminals

```
ls -l /dev/video*
```

folgende Ausgabe



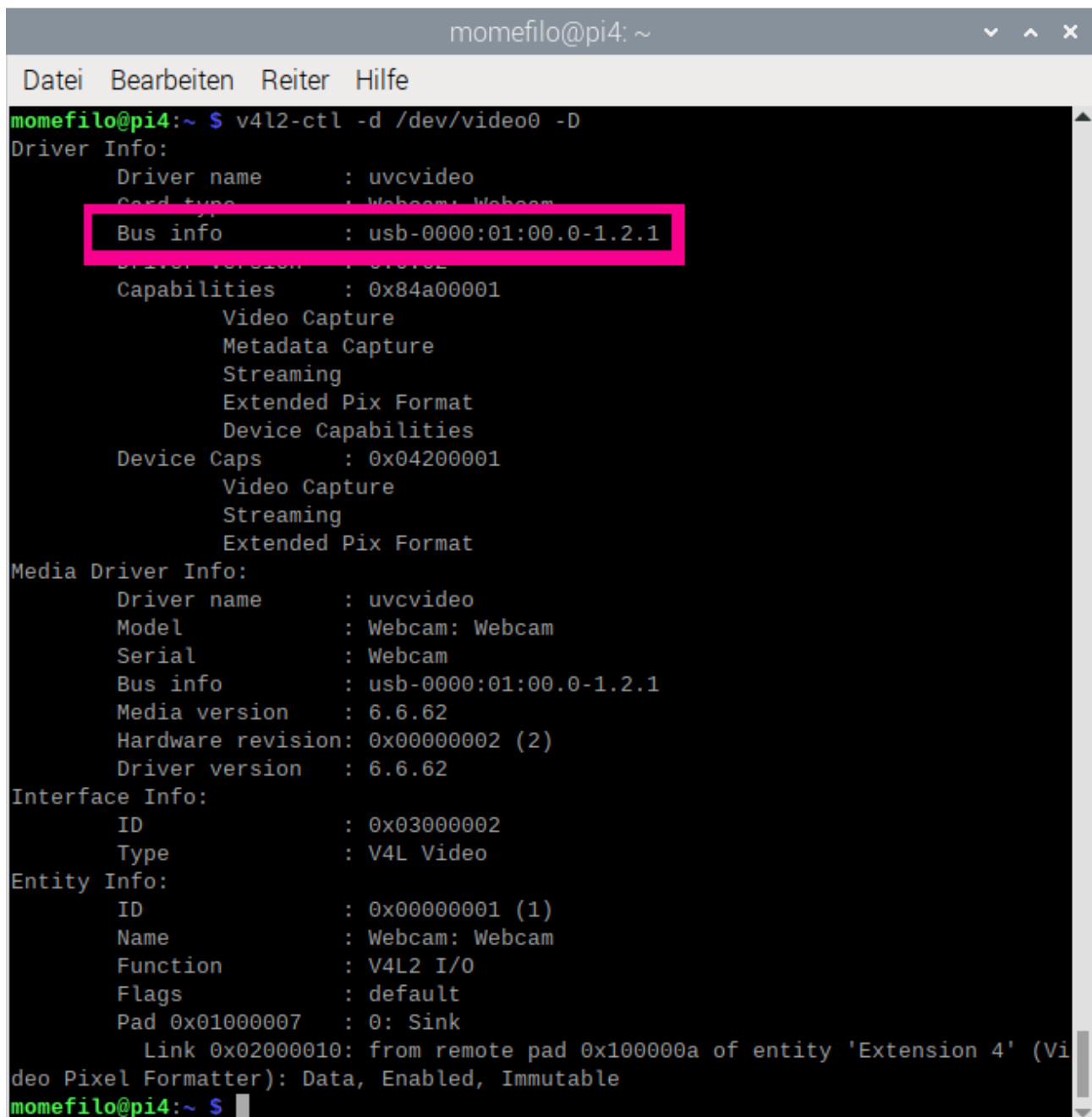
```
momefilo@pi4: ~  
Datei Bearbeiten Reiter Hilfe  
Connection to pi4m closed.  
momefilo@pi4:~ $ ls -l /dev/video*  
crw-rw---- 1 root video 81, 14 Dec 26 12:18 /dev/video0  
crw-rw---- 1 root video 81, 15 Dec 26 12:18 /dev/video1  
crw-rw---- 1 root video 81,  1 Dec 26 12:17 /dev/video10  
crw-rw---- 1 root video 81,  5 Dec 26 12:17 /dev/video11  
crw-rw---- 1 root video 81, 10 Dec 26 12:17 /dev/video12  
crw-rw---- 1 root video 81,  0 Dec 26 12:17 /dev/video13  
crw-rw---- 1 root video 81,  2 Dec 26 12:17 /dev/video14  
crw-rw---- 1 root video 81,  3 Dec 26 12:17 /dev/video15  
crw-rw---- 1 root video 81,  4 Dec 26 12:17 /dev/video16  
crw-rw---- 1 root video 81, 11 Dec 26 12:17 /dev/video18  
crw-rw---- 1 root video 81, 12 Dec 26 12:17 /dev/video19  
crw-rw---- 1 root video 81, 16 Dec 26 12:18 /dev/video2  
crw-rw---- 1 root video 81,  6 Dec 26 12:17 /dev/video20  
crw-rw---- 1 root video 81,  7 Dec 26 12:17 /dev/video21  
crw-rw---- 1 root video 81,  8 Dec 26 12:17 /dev/video22  
crw-rw---- 1 root video 81,  9 Dec 26 12:17 /dev/video23  
crw-rw---- 1 root video 81, 13 Dec 26 12:17 /dev/video31  
momefilo@pi4:~ $
```

Um nun zu ermitteln welche der Gerätedateien dem csi-Kameramodul zugewiesen ist, müssen wir nacheinander die Gerätedateien /dev/video0 bis /dev/video2 genauer untersuchen. Denn eine unbewiesene Annahme sagt das Videogerätedateien mit zwei Ziffern keine extern angeschlossenen Videogeräte darstellen. Mit dem bereits vorinstallierten Programm v4l2-ctl lässt sich eine solche Überprüfung leicht durchführen in dem man die Videogerätedatei beim Programmaufruf als Parameter mit übergibt.

So überprüfen wir das Gerät /dev/video0 mit folgendem Programmaufruf in der Eingabeaufforderung des Terminals

```
v4l2-ctl -d /dev/video0 -D
```

Beispielhaft erscheint folgende Bildschirmausgabe



```
momefilo@pi4: ~  
Datei Bearbeiten Reiter Hilfe  
momefilo@pi4:~ $ v4l2-ctl -d /dev/video0 -D  
Driver Info:  
  Driver name      : uvcvideo  
  Card type        : Webcam: Webcam  
  Bus info         : usb-0000:01:00.0-1.2.1  
  Driver version   : 6.6.62  
  Capabilities     : 0x84a00001  
    Video Capture  
    Metadata Capture  
    Streaming  
    Extended Pix Format  
    Device Capabilities  
  Device Caps     : 0x04200001  
    Video Capture  
    Streaming  
    Extended Pix Format  
Media Driver Info:  
  Driver name      : uvcvideo  
  Model           : Webcam: Webcam  
  Serial          : Webcam  
  Bus info         : usb-0000:01:00.0-1.2.1  
  Media version    : 6.6.62  
  Hardware revision: 0x00000002 (2)  
  Driver version   : 6.6.62  
Interface Info:  
  ID              : 0x03000002  
  Type            : V4L Video  
Entity Info:  
  ID              : 0x00000001 (1)  
  Name            : Webcam: Webcam  
  Function        : V4L2 I/O  
  Flags           : default  
  Pad 0x01000007  : 0: Sink  
    Link 0x02000010: from remote pad 0x1000000a of entity 'Extension 4' (Video Pixel Formatter): Data, Enabled, Immutable  
momefilo@pi4:~ $
```

Welche im magetafarben markierten Bereich zeigt das es sich hier um die USB-Cam handeln muss. Das Komando

```
v4l2-ctl -d /dev/video1 -D
```

zeigte bei mir auch ein USB-Gerät was darauf schließen lässt das die USB-Cam zwei Gerätedateien belegt. Erst der Aufruf

```
v4l2-ctl -d /dev/video2 -D
```

offenbarte folgendes



```
momefilo@pi4: ~  
Datei Bearbeiten Reiter Hilfe  
momefilo@pi4:~ $ v4l2-ctl -d /dev/video2 -D  
Driver Info:  
Driver name      : unicam  
Card type        : unicam  
Bus info         : platform:fe801000.csi  
Driver version    : 6.6.62  
Capabilities     : 0xa5a00001  
Video Capture  
Metadata Capture  
Read/Write  
Streaming  
Extended Pix Format  
Device Capabilities  
Device Caps      : 0x25200001  
Video Capture  
Read/Write  
Streaming  
Extended Pix Format  
Media Driver Info:  
Driver name      : unicam  
Model           : unicam  
Serial          :  
Bus info         : platform:fe801000.csi  
Media version    : 6.6.62  
Hardware revision: 0x00000000 (0)  
Driver version   : 6.6.62  
Interface Info:  
ID              : 0x03000005  
Type            : V4L Video  
Entity Info:  
ID              : 0x00000003 (3)  
Name            : unicam-image  
Function        : V4L2 I/O  
Flags           : default  
Pad 0x01000004  : 0: Sink  
Link 0x02000007: from remote pad 0x10000002 of entity 'ov5647 10-0036'  
(Camera Sensor): Data, Enabled, Immutable  
momefilo@pi4:~ $
```

und im markierten Bereich die Gerätedatei als dem csi-Kameramodul zugewiesen zeigt.

**Wichtig:** Wenn nur ein Videogerät/Kamera am RPiZero2W angeschlossen ist, ist die zugewiesene Gerätedatei immer /dev/video0. Wir benötigen nur die tatsächliche Ziffer

## Installation/Konfiguration auf den Raspi starten.

Das script erwartet als ersten parameter die DeviceID der cam, also bei einer Cam 0, und als zweiten den im Textdokument kopierten Streamschlüssel

Wir geben folgende befehle nach einander ein:

```
wget raw.githubusercontent.com/momefilo/streaming/refs/heads/main/install.sh  
chmod +x install.sh  
sudo ./install.sh meine_DeviceID mein_streamkey
```

Nun warten wir bis der RPi rebootet, gehen zu Youtube zurück und schauen

und warten noch mal eine Minute. Da ist das Bild! Nun könnt Ihr nach belieben konfigurieren.

Mit nicht langsamen SD-Karten dauert der Vorgang ca. fünf Minuten. Sollten Ihr eine Minute nach dem Reebot nicht im YT-Interface sehen wenden Euch sich an <https://forum-raspberrypi.de/>

## Hinweis

Im install.sh Script kann die Zeile 39 aukommentiert werden und eine der drei Darunterliegenden aktiviert werden für NoIr-Cams mit ov5647Sensor, USB-Cams, und USB-Cams an Intel-Maschinen