

# Youtube: Mainsail Timelapse mit jeder Kamera

Mit dieser Dokumentation könnt ihr nahezu jede beliebige Kamera für Timelapse nutzen.

Folgendes Video diene als Quelle:

<https://www.youtube.com/watch?v=KsCaPWglf5E>

## 1. Installation gphoto2

### Kompatibilitätsliste

<http://gphoto.org/doc/remote/>

Öffne Putty in und gebe folgendes Kommando ein und wähle die 1 (install latest development version):

```
wget https://raw.githubusercontent.com/gonzalo/gphoto2-updater/master/gphoto2-updater.sh && wget  
https://raw.githubusercontent.com/gonzalo/gphoto2-updater/master/.env && chmod +x gphoto2-updater.sh  
&& sudo ./gphoto2-updater.sh  
sudo apt-get install libgphoto2-dev
```

Die Installation kann ein paar Minuten beanspruchen.

Lösche ältere Versionen mit diesem Kommando (optional):

```
sudo apt-get remove gphoto2 libgphoto2-6 libgphoto2-dev libgphoto2-l10n libgphoto2-port12
```

Starte den Raspberry Pi neu

```
sudo reboot
```

## 2. gphoto2 testen und konfigurieren

Prüfe, ob eine Verbindung zu deiner Kamera hergestellt werden kann. Kann eine Verbindung aufgebaut werden, so wird dies in Putty wie folgt angezeigt.

```
gphoto2 --auto-detect
```

### Screenshot

```
pi@raspberrypi:~ $ gphoto2 --auto-detect
Model                               Port
-----
Canon EOS 400D (PTP mode)          usb:001,006
pi@raspberrypi:~ $ gphoto2 --auto-detect
```

oder

```
pi@raspberrypi:~ $ gphoto2 --auto-detect
Model                               Port
-----
Sony ZV-E10 (Control)              usb:001,017
pi@raspberrypi:~ $
```

Sollte keine Kamera angezeigt werden, dann starte den Raspberry Pi erneut und prüfe ob die Kamera vorher in den richtigen USB Modus versetzt wurde. In der Kompatibilitätsliste wird unter Bemerkungen aufgeführt, wenn ein besonderer Treiber für die Kamera benötigt wird.

Erstelle als nächstes einen Snapshot mit der Kamera.

```
gphoto2 --auto-detect --capture-image
```

Sollte dies erfolgreich sein, dann wird dies in Putty ausgegeben. Dabei wird der Snapshot auf der Kamera gespeichert.

### Screenshot

```
pi@raspberrypi:~ $ gphoto2 --auto-detect --capture-image
Model                               Port
-----
Sony ZV-E10 (Control)              usb:001,017
New file is in location /capt0000.jpg on the camera
pi@raspberrypi:~ $
```

Mit diesem Kommando wird das Image auf den Raspberry Pi übertragen.

```
gphoto2 --auto-detect --capture-image-and-download
```

## Speicherort

Der Snapshot wird unter /home/pi abgelegt.

/home/pi/			
Name	Größe	Geändert	Rechte
Documents		22.09.2022 02:34:46	rw-r--r--
Desktop		22.09.2022 02:34:44	rw-r--r--
Bookshelf		22.09.2022 02:14:33	rw-r--r--
.local		22.09.2022 02:14:32	rw-r--r--
.gphoto		03.02.2023 12:26:17	rw-r--r--
.config		24.01.2023 20:37:58	rw-r--r--
.cache		24.01.2023 20:46:24	rw-r--r--
install-gphoto2.sh	7 KB	03.02.2023 11:44:32	rw-r--r--
gphoto2-updater.sh.1	7 KB	03.02.2023 12:44:24	rw-r--r--
gphoto2-updater.sh	7 KB	03.02.2023 11:55:02	rw-r--r--
capt0000.jpg	10.558 KB	03.02.2023 13:01:41	rw-r--r--

Prüfe die möglichen Speicherorte deiner Kamera:

```
gphoto2 --get-config capturetarget
```

## Putty Fenster

```
pi@raspberrypi:~ $ gphoto2 --get-config capturetarget
Label: Capture Target
Readonly: 0
Type: RADIO
Current: sdram
Choice: 0 sdram
Choice: 1 card+sdram
Choice: 2 card
END
pi@raspberrypi:~ $
```

Ändere den Speicherort auf die Speicherkarte (hier 2)

```
gphoto2 --set-config capturetarget=2
```

## 3. KIAUH installieren (wenn noch nicht vorhanden)

Mit folgendem Kommando wird KIAUH installiert

```
cd ~
```

```
git clone https://github.com/th33xitus/kiauh.git
```

## 4. Shell Command installieren

Die Installation wird über KIAUH vorgenommen.

Zunächst wird Kiauh geöffnet

```
./kiauh/kiauh.sh
```

Wähle 4 (Advanced)

### KIAUH Menü

```
\=====/  
/=====/  
| Please read the newest changelog carefully: |  
| https://git.io/JnmlX |  
/=====/  
/=====/  
|           [ Main Menu ]           |  
|-----|  
| 0) [Log-Upload] | Klipper: Installed: 1(py3) |  
|                 | Repo: Klipper3d/klipper |  
| 1) [Install]    |  
| 2) [Update]     | Moonraker: Installed: 1 |  
| 3) [Remove]    |  
| 4) [Advanced]   | Mainsail: Installed! |  
|                 | Fluididd: Not installed! |  
|                 | KlipperScreen: Not installed! |  
| 6) [Settings]  | Telegram Bot: Not installed! |  
|                 | Obico: Not installed! |  
| v5.0.0-14      | Octoprint: Not installed! |  
|-----|  
|                 | Q) Quit |  
/=====/  
##### Perform action: 4
```

Wähle 8 (G-Code Shell Command)

### KIAUH Menü

```

/=====\
|          [ KIAUH ]          |
|  Klipper Installation And Update Helper  |
|          ~~~~~~            |
/=====\
/=====\
|          [ Advanced Menu ]          |
|-----|
| Klipper & API:      | Mainsail:      |
| 1) [Rollback]      | 6) [Theme installer] |
|                    |                    |
| Firmware:          | System:          |
| 2) [Build only]    | 7) [Change hostname] |
| 3) [Flash only]    |                    |
| 4) [Build + Flash] | Extras:          |
| 5) [Get MCU ID]    | 8) [G-Code Shell Command] |
|-----|
|                    | B) « Back          |
\=====|
##### Perform action: 8

```

Bestätige mit Y

## KIAUH Menü

```

/=====\
|          [ KIAUH ]          |
|  Klipper Installation And Update Helper  |
|          ~~~~~~            |
/=====\
/=====\
| You are about to install the 'G-Code Shell Command' |
| extension. Please make sure to read the instructions |
| before you continue and remember that potential risks |
| can be involved after installing this extension!      |
|                                                       |
| You accept that you are doing this on your own risk! |
\=====|
##### Do you want to continue? (Y/n): Y

```

Bestätige die Frage, ob eine Beispieldatei angelegt wird mit ja.

Schließe KIAUH mit B und danach Q

## 5. Skripte und Macros installieren.







### shell\_command.cfg

Wenn noch nicht vorhanden, erstelle eine shell\_command.cfg Datei

Inkludiere diese Datei in deine printer.cfg

```
1 [include macro.cfg]
2 [include mainsail.cfg]
3 [include print_area_bed_mesh.cfg]
4 [include shell_command.cfg]
5
```

### Mainsail Oberfläche

<input type="checkbox"/>	Name ↑	Filesize ↑	Last modified
<input type="checkbox"/>	 macro.cfg	8.5 kB	Jan 24, 2023 9:21 P
<input type="checkbox"/>	 mainsail.cfg	2.8 kB	Jan 24, 2023 9:22 P
<input type="checkbox"/>	 moonraker.conf	0.6 kB	Jan 24, 2023 9:00 P
<input type="checkbox"/>	 print_area_bed_mesh.cfg	7.2 kB	Jan 24, 2023 9:27 P
<input type="checkbox"/>	 printer.cfg	3.9 kB	Jan 27, 2023 1:03 P
<input type="checkbox"/>	 <b>shell_command.cfg</b>	0.1 kB	Feb 3, 2023 1:15 P

Files 10 ▾ 1-6 of 6 <

Füge folgenden Code in die shell\_command.cfg ein

```
#=====
# SET CAMERA SAVE LOCATION
#=====
```

```
[gcode_shell_command gphoto2_set_config]
command: gphoto2 --set-config capturetarget=1
timeout: 1. ; make sure this is long enough to account for focus time
verbose: False
```

```
#=====
# DSLR CAPTURE IMAGE
#=====
```

```
[gcode_shell_command gphoto2_trigger_snapshot]
command: gphoto2 --auto-detect --trigger-capture
timeout: 10. ; make sure this is long enough to account for focus time
verbose: False
```

## macro.cfg und printer.cfg

Quelle: <https://github.com/EB3DPrinting/Klipper/blob/main/Ender-3Pro%232/macros.cfg>

Füge diese Makros in deine macro.cfg ein

### macro.cfg

```
#=====
# MANUAL SET CAPTURE TARGET
#=====
```

```
[gcode_macro GPHOTO_SET_TARGET]
gcode:
    RUN_SHELL_COMMAND CMD=gphoto2_set_config
```

```
#=====
# TAKE SNAPSHOT COMMAND
#=====
```

```
[gcode_macro TAKE_SNAPSHOT]
description: Saves current location, moves toolhead to rear of bed, and takes a snapshot before
```

returning.

gcode:

{% set delayMS = 2000 %} ; Delay after moving and before taking the snapshot, deals with buffering so we don't get the print head in motion

{% set snapshotPositionX = printer.toolhead.axis\_maximum.x / 2 %} ; Middle of bed

{% set snapshotPositionY = printer.toolhead.axis\_maximum.y - 5 %} ; rear of bed w/ some buffer

{% set currentX = printer.gcode\_move.gcode\_position.x %}

{% set currentY = printer.gcode\_move.gcode\_position.y %}

{% set currentZ = printer.gcode\_move.gcode\_position.z %} ; could be used for z-hop which we are not doing in this macro, so it's best to run this in AFTER\_LAYER\_CHANGE

SAVE\_GCODE\_STATE NAME=take\_snapshot\_state

{% if printer.extruder.can\_extrude|lower == 'true' %}

G10 ; retract

{% endif %}

G90

G0 X{snapshotPositionX} Y{snapshotPositionY} F5000.0 ;Move to snapshot position

G4 P{delayMS} ; Dwell for delayMS seconds

RUN\_SHELL\_COMMAND CMD=gphoto2\_trigger\_snapshot

G0 X{currentX} Y{currentY} F5000.0 ;Return to original position

{% if printer.extruder.can\_extrude|lower == 'true' %}

G11 ; unretract

{% endif %}

RESTORE\_GCODE\_STATE NAME=take\_snapshot\_state

#=====

# MANUAL SNAPSHOT TEST

#=====

[gcode\_macro GPHOTO\_SNAPSHOT\_TEST]

gcode:



```
RUN_SHELL_COMMAND CMD=gphoto2_trigger_snapshot
```

Füge diese Sektion in deine printer.cfg ein und passe die Werte an deine druckerspezifischen Werte an

### printer.cfg

```
#=====
=====
# FIRMWARE RETRACTION
#=====
=====
[firmware_retraction]
# The length of filament (in mm) to retract when G10 is activated,
# and to unretract when G11 is activated (but see
# unretract_extra_length below). The default is 0 mm.
retract_length: 1
# The speed of retraction, in mm/s. The default is 20 mm/s.
retract_speed: 35
# The length (in mm) of *additional* filament to add when
# unretracting.
#unretract_extra_length: 0
# The speed of unretraction, in mm/s. The default is 10 mm/s.
unretract_speed: 35
```

```
#=====
# FIRMWARE RETRACTION
#=====
[firmware_retraction]
# The length of filament (in mm) to retract when G10 is activated,
# and to unretract when G11 is activated (but see
# unretract_extra_length below). The default is 0 mm.
retract_length: 1
# The speed of retraction, in mm/s. The default is 20 mm/s.
retract_speed: 35
# The length (in mm) of *additional* filament to add when
# unretracting.
#unretract_extra_length: 0
# The speed of unretraction, in mm/s. The default is 10 mm/s.
unretract_speed: 35
```

## 6. Makros einstellen und testen

Passe die Werte im Makro TAKE\_SNAPSHOT nach deinen Bedürfnissen an

### TAKE\_SNAPSHOT

```
[gcode_macro TAKE_SNAPSHOT]
description: Saves current location, moves toolhead to rear of bed, and takes a snapshot before returning.
gcode:
(% set delayMS = 2000 %) ; delay after moving and before taking the snapshot, deals with buffering so we don't get the print head in motion
(% set snapshotPositionX = printer.toolhead.axis_maximum.x / 2 %) ; Middle of bed
(% set snapshotPositionY = printer.toolhead.axis_maximum.y - 5 %) ; rear of bed w/ some buffer
(% set currentX = printer.gcode_move.gcode_position.x %)
(% set currentY = printer.gcode_move.gcode_position.y %)
(% set currentZ = printer.gcode_move.gcode_position.z %) ; could be used for z-hop which we are not doing in this macro, so it's best to run this in AFTER_LAYER_CHANGE
SAVE_GCODE_STATE NAME=take_snapshot_state

{% if printer.extruder.can_extrude|lower == 'true' %}
  G10 ; retract
{% endif %}

G90

G0 X[snapshotPositionX] Y[snapshotPositionY] F5000.0 ;Move to snapshot position

G4 P[delayMS] ; Dwell for delayMS seconds

RUN_SHELL_COMMAND CMD=gphoto2_trigger_snapshot

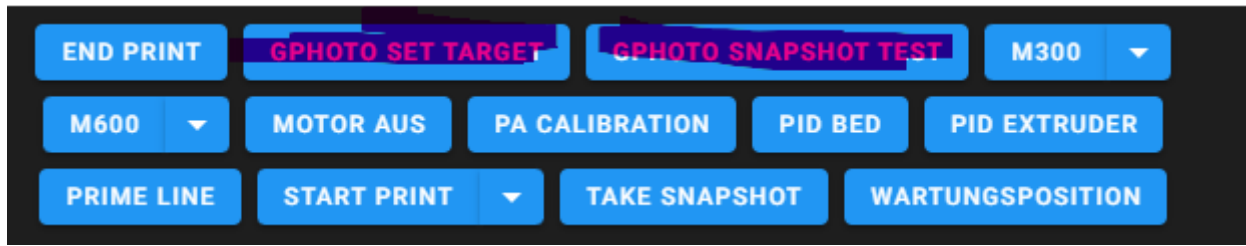
G0 X[currentX] Y[currentY] F5000.0 ;Return to original position

{% if printer.extruder.can_extrude|lower == 'true' %}
  G11 ; unretract
{% endif %}

RESTORE_GCODE_STATE NAME=take_snapshot_state

#=====
# MANUAL SNAPSHOT TEST
#=====
[gcode_macro GPHOTO_SNAPSHOT_TEST]
gcode:
  RUN_SHELL_COMMAND CMD=gphoto2_trigger_snapshot
```

Es sollten unter der Makro Sektion neue Makros sichtbar sein



Teste mit dem Befehl "GPHOTO SNAPSHOT TEST" ob deine Kamera auslöst.

Sollte der Fehler "Unknown command: "RUN\_SHELL\_COMMAND"" erscheinen, dann wurde die shell\_command.cfg nicht eingefügt bzw. inkludiert.

Teste mit dem Befehl "TAKE SNAPSHOT" ob dein Druckkopf zur korrekten position fährt und die Kamera auslöst (vorher muss der Drucker gehomed werden).

## 7. Slicer Einstellung

Für unter dem Reiter

Printer Settings / Custom G-code / After layer change G-code folgendes Kommando ein

TAKE\_SNAPSHOT

## 8. Bilder bearbeiten

Die gespeicherten Bilder können nun zu einem Video editiert werden. DaVinci Resolve bietet eine Möglichkeit. Es gibt aber noch viele weitere Programme wie bspw. Shotcut.

Revision #2

Created 17 March 2023 12:35:35 by Cryd

Updated 22 March 2023 10:32:02 by Robin