

Klipper/Mainsail/Moonraker

- !!! WICHTIG !!! Klipper Änderung vom bed_mesh
- Cancel print button/Funktion geht nicht
- Der Drucker geht während des Drucks auf Pause
- Deutsch Übersetzung Klipper Dokumentation
- Drucker lässt sich nicht korrekt verbinden (ständiger Disconnect)
- Kiauh installieren
- Klipper auf ältere Version zurücksetzen
- Klipper Pin Vorzeichen
- Klipper Screen
- Pi Auslastung prüfen
- Timelapse funktioniert nach Update nicht mehr | Timelapse installieren
- Vyper Firmware aktualisieren / installieren
- Vyper flashen
- Webcam
- Wo finde ich Log Dateien
- Z-Tilt / Quad Gantry Level
- Mainsail ohne Drucker nutzen

!!! WICHTIG !!! Klipper Änderung vom bed_mesh

Bei einer bevorstehenden Änderung an Klipper wird die Standardkonfiguration `bed_mesh` nicht mehr automatisch beim Start geladen.

Wenn Sie heute eine `bed_mesh` Konfiguration verwenden, wird empfohlen, `BED_MESH_PROFILE LOAD=default` Ihrem `START_PRINT`-Makro (oder gleichwertigen Druckstartbefehlen in Ihrem Slicer) einen expliziten Befehl hinzuzufügen. Dieser Wechsel zu Klipper ist für den 1. Februar 2023 geplant.

 [Quelle](#)

Cancel print button/Funktion geht nicht

Dieses Makro in eurer **mainsail.cfg** tauschen und in den Settings den "cancel print button" an machen.

Makros

```
[gcode_macro CANCEL_PRINT]
```

```
description: Bricht den aktuellen Druck ab
```

```
rename_existing: CANCEL_PRINT_BASE
```

```
variable_park: True
```

```
gcode:
```

```
## Move head and retract only if not already in the pause state and park set to true
```

```
{% if printer.pause_resume.is_paused|lower == 'false' and park|lower == 'true'%}
```

```
_TOOLHEAD_PARK_PAUSE_CANCEL
```

```
{% endif %}
```

```
TURN_OFF_HEATERS
```

```
CANCEL_PRINT_BASE
```

```
#####
```

```
####
```

```
# Druckkop Parkbefehl
```

```
#####
```

```
####
```

```
[gcode_macro _TOOLHEAD_PARK_PAUSE_CANCEL]
```

```
description: Helper: Parkposition im PAUSE and CANCEL_PRINT
```

```
variable_extrude: 1.0
```

```
gcode:
```

```
##### set park positon for x and y #####
```

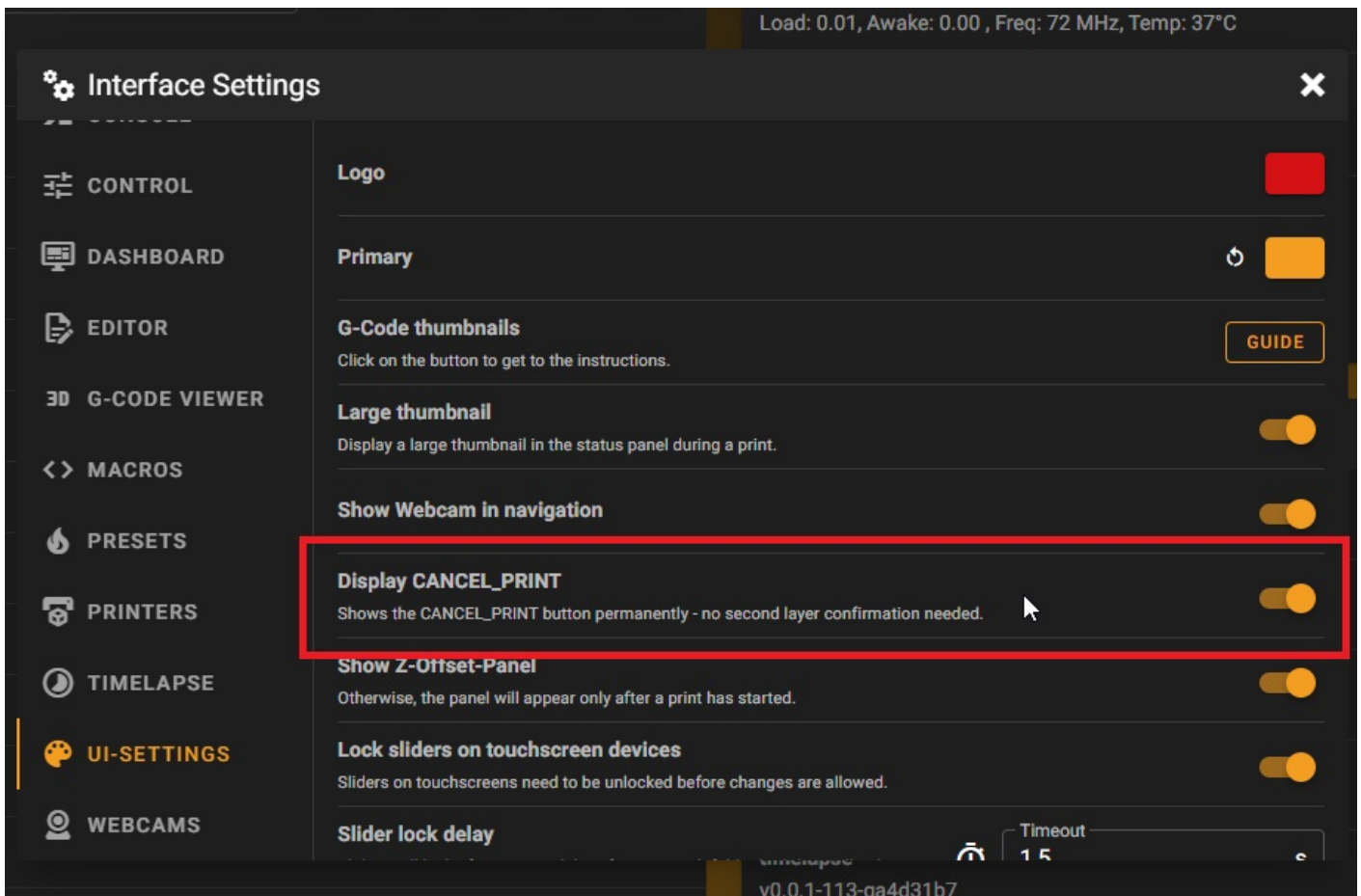
```
# default is your max posion from your printer.cfg
```

```
{% set x_park = printer.toolhead.axis_maximum.x|float - 5.0 %}
```

```

{% set y_park = printer.toolhead.axis_maximum.y|float - 5.0 %}
{% set z_park_delta = 2.0 %}
##### calculate save lift position #####
{% set max_z = printer.toolhead.axis_maximum.z|float %}
{% set act_z = printer.toolhead.position.z|float %}
{% if act_z < (max_z - z_park_delta) %}
    {% set z_safe = z_park_delta %}
{% else %}
    {% set z_safe = max_z - act_z %}
{% endif %}
##### end of definitions #####
{% if printer.extruder.can_extrude|lower == 'true' %}
    M83
    G1 E-{{extrude}} F2100
    {% if printer.gcode_move.absolute_extrude |lower == 'true' %} M82 {% endif %}
{% else %}
    {action_respond_info("Extruder not hot enough")}
{% endif %}
{% if "xyz" in printer.toolhead.homed_axes %}
    G91
    G1 Z5 F900
    G90
    G1 X{{x_park}} Y{{y_park}} F6000
    {% if printer.gcode_move.absolute_coordinates|lower == 'false' %} G91 {% endif %}
{% else %}
    {action_respond_info("Printer not homed")}
{% endif %}

```



Zusätzliche Makros

Druckpause

```
#####
#####
# Druckpause
#####
#####

[gcode_macro PAUSE]
description: Pausiert den aktuellen Druck
rename_existing: PAUSE_BASE
# change this if you need more or less extrusion
variable_extrude: 1.0
gcode:
    ##### read E from pause macro #####
    {% set E = printer["gcode_macro PAUSE"].extrude|float %}
    ##### set park position for x and y #####
```

```
# default is your max posion from your printer.cfg
{% set x_park = printer.toolhead.axis_maximum.x|float - 5.0 %}
{% set y_park = printer.toolhead.axis_maximum.y|float - 5.0 %}
##### calculate save lift position #####
{% set max_z = printer.toolhead.axis_maximum.z|float %}
{% set act_z = printer.toolhead.position.z|float %}
{% if act_z < (max_z - 2.0) %}
    {% set z_safe = 2.0 %}
{% else %}
    {% set z_safe = max_z - act_z %}
{% endif %}
{%set min_extrude_temp = printer.configfile.settings["extruder"]["min_extrude_temp"]|int %}
{%set act_extrude_temp = printer.extruder.temperature|int %}
##### end of definitions #####
PAUSE_BASE
G91
{% if act_extrude_temp > min_extrude_temp %}
    G1 E-{E} F2100
{% else %}
    {action_respond_info("Extruder not hot enough")}
{% endif %}
{% if "xyz" in printer.toolhead.homed_axes %}
    G1 Z{z_safe} F900
    G90
    G1 X{x_park} Y{y_park} F6000
{% else %}
    {action_respond_info("Printer not homed")}
{% endif %}
```

End Print

```
#####
# END PRINT          #
#####
```

[gcode_macro END_PRINT]

description: All what needs to be done at print end

gcode:

M117 Turn off Heaters

TURN_OFF_HEATERS ; Turn off bed and nozzle

□G91 ; Relative positioning

SET_GCODE_OFFSET Z_ADJUST=+0.06

□M117 Retract

G1 E-1 F3000 ; Retract

□G1 X-0.5 Y-0.5 Z20 E-5 ; Move a bit and retract filament even more

□G90 ; Absolute positioning

□G1 X0 Y180 Z35 F2200 ; Move bed to front

□M107 ; Turn off part fan

□M84 ; Steppers off

□G90 ; Absolute positioning

□M117 Print done ; Send finish to display

Druckwiederaufnahme

#####

#####

Druckwiederaufnahme

#####

#####

[gcode_macro RESUME]

description: Setzt den aktuellen Druck fort

rename_existing: RESUME_BASE

gcode:

read E from pause macro

{% set E = printer["gcode_macro PAUSE"].extrude|float %}

{%set min_extrude_temp = printer.configfile.settings["extruder"]["min_extrude_temp"]|int %}

{%set act_extrude_temp = printer.extruder.temperature|int %}

get VELOCITY parameter if specified

{% if 'VELOCITY' in params|upper %}

{% set get_params = ('VELOCITY=' + params.VELOCITY) %}

{%else %}

{% set get_params = "" %}

{% endif %}

end of definitions

{% if act_extrude_temp > min_extrude_temp %}

G91

G1 E{E} F2100

{% else %}

{action_respond_info("Extruder not hot enough")}

{% endif %}

RESUME_BASE {get_params}

Der Drucker geht während des Drucks auf Pause

Häufigste Fehlerquelle:

- Filamentsensor

Lösung:

Einfach abstecken und aus der Printer.cfg aus kommentieren.

```
#[filament_switch_sensor runout]
#pause_on_runout: True
#switch_pin: PA5
```

Deutsch Übersetzung Klipper Dokumentation

Hallo, In Facebook wurde von Jörg Müller die Dokumentation super übersetzt. Wir danken Ihm
vielmals dafür. Hier ist nochmal das PDF vom stand 08.06.2022

[Klipper-Deutsche-Übersetzung.pdf](#)

Drucker lässt sich nicht korrekt verbinden (ständiger Disconnect)

Drucker anstecken und dann " `sudo dmesg` " idProduct Merken

```
nano /etc/udev/rules/99-ch341.rule
```

```
SUBSYSTEM=="usb", ATTRS{idVendor}=="1a86", ATTRS{idProduct}=="$gemerkte idProduct", MODE:="0666"
```

```
systemctl restart udev
```

Kiauh installieren

Alle wichtigen Infos und die Dokumentation von Kiauh findet ihr [hier](#)

Installation

SSH Verbindung aufbauen (Putty)

```
cd ~
```

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

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

Navigation in Kiauh

- Ihr könnt das jeweilige Menü mit der Ziffer davon anwählen. Beispiel: Install hat die Nummer "1" -> "1 + ENTER"
- Zurück könnt ihr mit "b + ENTER"
- Kiauh schließen mit "q + ENTER"

Klipper auf ältere Version zurücksetzen

<https://www.youtube.com/embed/gbWBC41iQtk>

Vorher immer Backups erstellen. FTP -> klipper_config ordner sichern

Aktuelle Version des Klipper ordners anzeigen

```
cd ~/klipper  
git show
```

Letzte Commits anzeigen:

```
git log --stat
```

Klipper Ordner zurücksetzen

```
git reset --hard <COMMIT_NUMMER>
```

Klipper Pin Vorzeichen

pin: PA1

pin: !PA1 (negiert)

pin: ^PA1 (pullup)

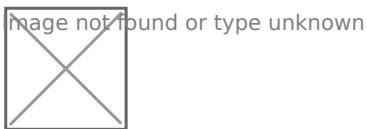
pin: ~PA1 (pulldown)

Allgemein

Ein Pullup- oder Pulldown-Widerstand wird dazu verwendet, einen Eingang auf einen definierten Wert zu "ziehen". Normalerweise befindet sich der Eingang im Zustand "schwebend/hochohmig", welcher sich irgendwo zwischen High und Low befindet. Nun sind Schaltungen leider nicht komplett ohne Störsignale, und durch Einstrahlungen von Signalen kann es nun passieren, dass kurzzeitig mal ein Wert über- oder unterschritten wird und der Eingang plötzlich ein High- oder Lowsignal bekommt. Dies führt dann zu unerklärlichen und unregelmäßig auftretenden Fehlern, die einen Neuling zur Verzweiflung treiben können.

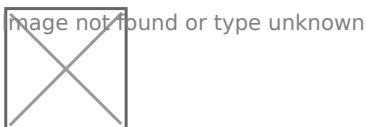
Pullup-Widerstand

Um dies zu vermeiden wird, während nichts am Eingang passiert, dieser auf High gezogen (pullup). Will man also einen Taster haben, der Low schaltet, so will man für den Rest der Zeit High am Eingang liegen haben. Dies kann man erreichen, indem man den Eingang fest mit VCC verbindet. Vom Eingang geht nun eine Leitung nach VCC und eine über den Taster nach GND. Leider würde dies zu einem Kurzschluss führen sobald man den Taster drückt und die Schaltung könnte (und würde höchstwahrscheinlich) Schaden nehmen. Deshalb wird nun zwischen VCC und dem Eingang ein hochohmiger Widerstand eingesetzt, der Pullup-Widerstand. Bei geschlossenem Taster wird nun der Strom über den Pullup-Widerstand nach GND fließen und der Input liegt auf GND (0V).



Pulldown-Widerstand

Der Pulldown-Widerstand funktioniert analog zum Pullup-Widerstand, nur dass nun VCC geschaltet werden soll und somit der Eingang auf GND gezogen werden muss. Dies geschieht in gleicher Weise wie beim Pullup, nur dass der Pulldown-Widerstand nun zwischen GND und dem Eingang platziert wird. Schließt man nun wieder den Taster, liegt am Eingang Vcc an --> High.



Quelle

Klipper Screen

Klipper-Screen läuft nicht oder hat eine hohe Auslastung

```
sudo systemctl stop KlipperScreen
```

```
cd ~/KlipperScreen
```

```
git reset --hard <commit>
```

```
sudo systemctl start KlipperScreen
```

Die commits findest ihr unter <https://github.com/jordanruthe/KlipperScreen> links unterhalb des grünen Code Buttons. Dort ist eine Uhr mit Bezeichnung "commits". Dort draufklicken und dann eine Commit ID kopieren (Bearbeitet)

Pi Auslastung prüfen

- Putty verbinden
- in die Konsole "htop" eingeben

```
htop
```

Ist htop nicht installiert `sudo apt-get install htop`

Timelapse funktioniert nach Update nicht mehr | Timelapse installieren

Prüfen ob Timelapse schon auf dem Pi ist (SSH Verbindung aufbauen)

```
cd ~/
git clone https://github.com/mainsail-crew/moonraker-timelapse.git
```

```
pi@mainsailos:~ $ git clone https://github.com/mainsail-crew/moonraker-timelapse.git
fatal: destination path 'moonraker-timelapse' already exists and is not an empty directory.
```

Timelapse updaten

```
cd ~/moonraker-timelapse
git pull
```

Timelapse neu installieren

```
cd ~/
git clone https://github.com/mainsail-crew/moonraker-timelapse.git
bash ~/moonraker-timelapse/install.sh
```

Bei der Fehlermeldung "**fatal: destination path 'moonraker-timelapse' already exists and is not an empty directory.**" -> die Befehle von **Timelapse updaten** durchführen!

Timelapse Update Manager in moonraker.conf überprüfen:

```
[update_manager timelapse]
type: git_repo
primary_branch: main
path: ~/moonraker-timelapse
```

origin: <https://github.com/mainsail-crew/moonraker-timelapse.git>

managed_services: klipper moonraker

Timelapse in der Moonraker.conf einfügen:

[timelapse]

Timelapse im Slicer einrichten:

Offizieller link zu der [Timelapse Doku](#).

Superslicer / PrusaSlicer

Printer Settings -> Custom G-code -> Before layer change Gcode -> `TIMELAPSE_TAKE_FRAME`

Cura:

Extensions -> Post Processing -> Modify G-Code ->

Add a script -> Insert at layer change -> G-code to insert = `TIMELAPSE_TAKE_FRAME`

Vyper Firmware aktualisieren / installieren

SSH Verbindung aufbauen (Putty) und Firmware erstellen

```
cd ~/klipper
```

```
make menuconfig
```

```
pi@mainsailos: ~/klipper
(Top)
Klipper Firmware Configura
[*] Enable extra low-level configuration options
  Micro-controller Architecture (STMicroelectronics STM32) --->
  Processor model (STM32F103) --->
[ ] Only 10KiB of RAM (for rare stm32f103x6 variant)
[*] Disable SWD at startup (for GigaDevice stm32f103 clones)
  Bootloader offset (32KiB bootloader) --->
  Clock Reference (8 MHz crystal) --->
  Communication interface (Serial (on USART1 PA10/PA9)) --->
(250000) Baud rate for serial port
() GPIO pins to set at micro-controller startup
```

Mit Q verlassen und speichern.

```
make clean
```

```
make
```

Firmware auf SD Karte packen

FTP Verbindung aufbauen (Filezilla, WinSCP) und klipper.bin downloaden Die klipper.bin liegt in
`/home/pi/klipper/out/klipper.bin`

SD Karte formatieren

File System: **FAT32**

Allocation Unit Size: **4096**

klipper.bin auf SD Karte schieben und in "main_board_20220909.bin" um nennen.

Der Name der .bin muss sich immer zu der vorherigen Version der Zahlenkombination am Schluss unterscheiden. alt: 20220909 -> neu: 20220910

Drucker flashen

- Drucker ausschalten
- Verbindungskabel zu Pi abziehen
- SD Karte einstecken
- Drucker einschalten
- Drucker sollte akustisches Signal geben
- nach 30 Sekunden Drucker ausschalten
- SD entfernen
- Pi wieder verbinden

Vyper flashen

1. ssh zum pi
2. `cd klipper`
3. `make menuconfig`
4. richtige Einstellung wählen und speichern

```
pi@mainsailos: ~/klipper
(Top)
Klipper Firmware Configuration
[*] Enable extra low-level configuration options
  Micro-controller Architecture (STMicroelectronics STM32) --->
  Processor model (STM32F103) --->
  [ ] Only 10KiB of RAM (for rare stm32f103x6 variant)
  [*] Disable SWD at startup (for GigaDevice stm32f103 clones)
  Bootloader offset (32KiB bootloader) --->
  Clock Reference (8 MHz crystal) --->
  Communication interface (Serial (on USART1 PA10/PA9)) --->
(250000) Baud rate for serial port
() GPIO pins to set at micro-controller startup
```

5. `make clean`
6. `make`
7. FTP verbindung aufbauen
8. klipper/out/klipper.bin auf SD Karte
9. Umbenennen "main_board_20200512.bin"
(Zahlenfolge ist das Datum und muss immer unterschiedlich zum Vorgänger sein)

Webcam

Wenig FPS bei einer Webcam

- WLAN Übertragung kann ein Problem sein -> Verbindet mal per LAN
- Mehrere Clients greifen parallel auf die Kamera zu
- Erhöhung der FPS in der crownsnest.conf (Neue Mainsail Version)

```
[cam 1]
mode: mjpg          # mjpg/rtsp
port: 8080          # Port
device: /dev/video0 # See Log for available ...
resolution: 640x480 # widthxheight format
max_fps: 15         # If Hardware Supports this it will be forced, otherwise
ignored/coerced.
```

Resolution = Auflösung -> Habt ihr bei euch eine HD Auflösung (1920x1080) eingetragen, kann es helfen diese auf 640x480 zu setzten

max_fps = Ihr könnt diese FPS mal auf 30 oder 60 hochsetzen

#	Resolution	Aspect Ratio	Framerates	Video	Image	FoV	Binni
1	1920x1080	16:9	1/10 <= fps <= 30	x		Partial	None
2	3280x2464	4:3	1/10 <= fps <= 15	x	x	Full	None
3	3280x2464	4:3	1/10 <= fps <= 15	x	x	Full	None
4	1640x1232	4:3	1/10 <= fps <= 40	x		Full	2x2
5	1640x922	16:9	1/10 <= fps <= 40	x		Full	2x2
6	1280x720	16:9	40 < fps <= 90	x		Partial	2x2
7	640x480	4:3	40 < fps <= 90	x		Partial	2x2

Ebenfalls könnt ihr auf dem Mainsail Dashboard in den Einstellungen unter Webcam Einstellungen die FPS mal höher stellen.

Interface Einstellungen

DASHBOARD

EDITOR

3D G-CODE BETRAC...

KONSOLE

MAKROS

PRESETS

REMOTE DRUCK...

STEUERUNG

UI-EINSTELLUNG...

WEBCAMS

Webcam bearbeiten

Name

Usb

Stream URL

/webcam/?action=stream

Schnappschuss URL

/webcam/?action=snapshot

Dienst

Adaptive MJPEG-Streamer (exp...

Ziel FPS

30

☐ Webcam waagerecht spiegeln

☐ Webcam vertikal spiegeln

ABBRUCH

WEBCAM AKTUALISIEREN

FPS: 00

Kein Bild der Webcam

Prüfen ob das USB Gerät erkannt wird.

```
lsusb
```

Die Ausgabe sollte nun die Kamera anzeigen. Wenn nicht bitte prüfen, ob die Kamera generell an einem anderen Gerät funktioniert.

```
pi@voron:~ $ lsusb
Bus 001 Device 005: ID 046d:085c Logitech, Inc. C922 Pro Stream Webcam
Bus 001 Device 004: ID 1d50:614e OpenMoko, Inc.
Bus 001 Device 006: ID 1d50:606f OpenMoko, Inc.
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp. SMC9514 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Videogeräte prüfen

```
v4l2-ctl --list-devices
```

```
pi@voron:~ $ v4l2-ctl --list-devices
bcm2835-codec-decode (platform:bcm2835-codec) :
    /dev/video10
    /dev/video11
    /dev/video12
    /dev/video18

bcm2835-isp (platform:bcm2835-isp) :
    /dev/video13
    /dev/video14
    /dev/video15
    /dev/video16

C922 Pro Stream Webcam (usb-3f980000.usb-1.4) :
    /dev/video0
    /dev/video1
```

crowsnest.conf prüfen

- Ist der richtige Port eingestellt `8080`
- Ist das richtige Gerät eingestellt `device: /dev/video0`

Die Codeblöcke müssen meist in Putty (SSH) ausgeführt werden.

Autofocus und Autohelligkeit ausschalten

Funktionsmöglichkeiten der Kamera herausfinden:

```
v4l2-ctl --list-ctrls
```

```
pi@voron:~$ v4l2-ctl --list-ctrls
        brightness 0x00980900 (int) : min=0 max=255 step=1 default=128 value=128
        contrast 0x00980901 (int) : min=0 max=255 step=1 default=128 value=128
        saturation 0x00980902 (int) : min=0 max=255 step=1 default=128 value=128
white_balance_temperature_auto 0x0098090c (bool) : default=1 value=1
        gain 0x00980913 (int) : min=0 max=255 step=1 default=0 value=0
        power_line_frequency 0x00980918 (menu) : min=0 max=2 default=2 value=2
white_balance_temperature 0x0098091a (int) : min=2000 max=6500 step=1 default=4000 value=4660 flags=inactive
        sharpness 0x0098091b (int) : min=0 max=255 step=1 default=128 value=128
        backlight_compensation 0x0098091c (int) : min=0 max=1 step=1 default=0 value=0
        exposure_auto 0x009a0901 (menu) : min=0 max=3 default=3 value=1
        exposure_absolute 0x009a0902 (int) : min=3 max=2047 step=1 default=250 value=1000
        exposure_auto_priority 0x009a0903 (bool) : default=0 value=1
        pan_absolute 0x009a0908 (int) : min=-36000 max=36000 step=3600 default=0 value=0
        tilt_absolute 0x009a0909 (int) : min=-36000 max=36000 step=3600 default=0 value=0
        focus_absolute 0x009a090a (int) : min=0 max=250 step=5 default=0 value=5
        focus_auto 0x009a090c (bool) : default=1 value=0
        zoom_absolute 0x009a090d (int) : min=100 max=500 step=1 default=100 value=100
```

Prüfen ob die Befehle richtig umgesetzt werden:

Dieses Beispiel zeigt die Deaktivierung des Autofokus durch das setzen des Wertes **0**.

```
v4l2-ctl --set-ctrl=focus_auto=0
```

Setzt den Focus Manuell auf den eingebenden Wert. Hier ist es 20

```
v4l2-ctl --set-ctrl=focus_absolute=20
```

Schaltet die Autohelligkeit aus.

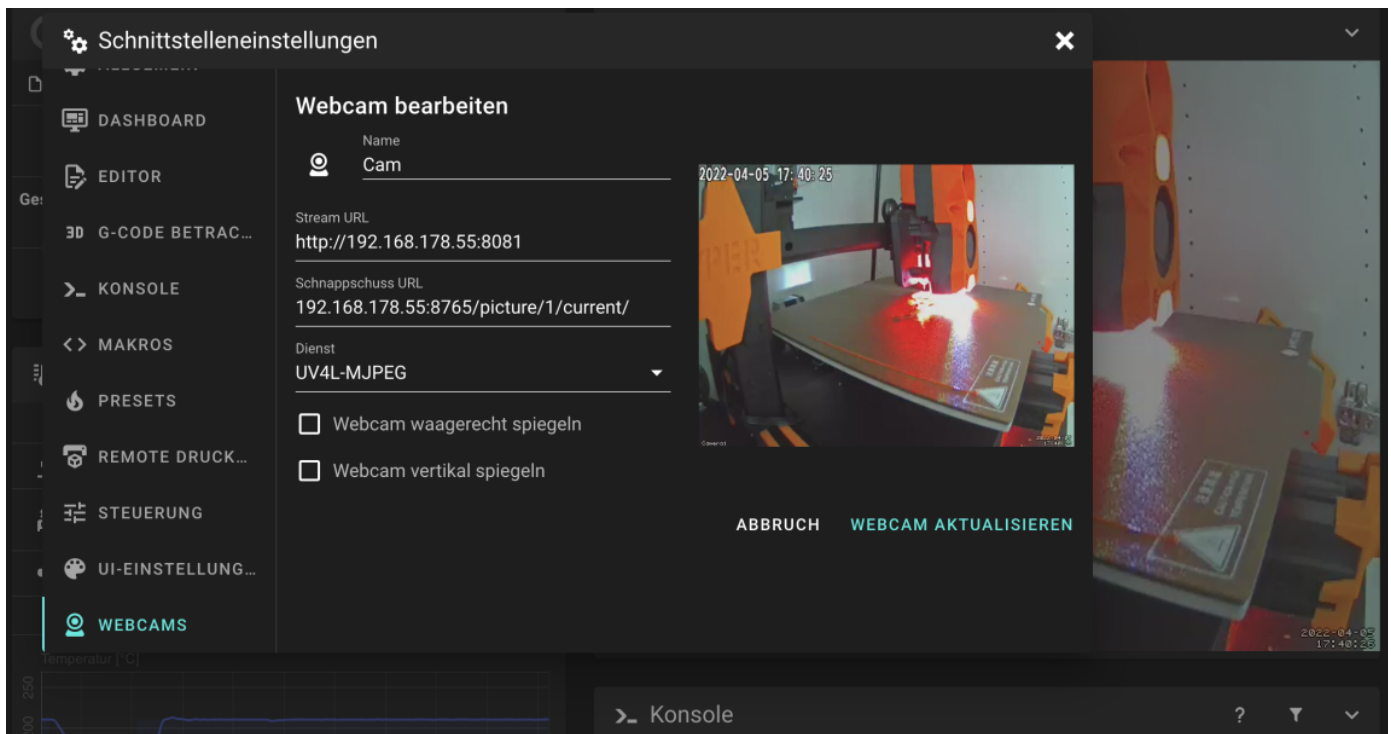
```
v4l2-ctl --set-ctrl=exposure_auto=0
```

Übertragung in Autostart:

Bearbeiten der crowsnest.conf

```
v4l2ctl: focus_auto=0,focus_absolute=5,exposure_auto=1,exposure_absolute=1000,zoom_absolute=100
```

RTSP Cam in ioBroker + MotionEye einbinden



Crowsnest Doku und Installation

Hier findest du die offizielle Doku zu Crowsnest:

<https://github.com/mainsail-crew/crowsnest>

Alle folgenden aufgeführten Commands sind auf dem Stand vom 14.12.2022

Wie wird crowsnest installiert:

```
cd ~  
git clone https://github.com/mainsail-crew/crowsnest.git  
cd ~/crowsnest  
sudo make install
```

So deinstallierst du crowsnest:

```
cd ~/crowsnest  
make uninstall
```

```
cd ~  
rm -rf crowsnest/
```

In -s /home/pi/klipper_logs/crowsnest.log

So updatest du crowsnest:

```
cd ~/crowsnest  
git pull
```

Link Verknüpfung erstellen

```
cd ~/printer_data/config  
ln -s /home/pi/klipper_config/crowsnest.conf
```

```
cd ~/printer_data/logs  
ln -s /home/pi/klipper_logs/crowsnest.log
```

Wo liegt die Crowsnest Konfig:

```
/home/pi/klipper_config/crowsnest.conf
```

Hier findest du die Logdatei zu Crowsnest:

```
/home/pi/klipper_logs/crowsnest.log
```

Update Manager Eintrag in der moonraker.conf

```
[update_manager crowsnest]
type: git_repo
path: ~/crowsnest
origin: https://github.com/mainsail-crew/crowsnest.git
install_script: tools/install.sh
```

Beispiel Configuration deiner Webcam:

```
#### crowsnest.conf
#### This is a typical default config.
#### Also used as default in mainsail / MainsailOS
#### See:
#### https://github.com/mainsail-crew/crowsnest/blob/master/README.md
#### for details to configure to your needs.

#####
####
####                                     #####
#### Information about ports and according URL's          #####
####                                     #####
#####
####
####                                     #####
#### Port 8080 equals /webcam/?action=[stream/snapshot]    #####
#### Port 8081 equals /webcam2/?action=[stream/snapshot]   #####
#### Port 8082 equals /webcam3/?action=[stream/snapshot]   #####
#### Port 8083 equals /webcam4/?action=[stream/snapshot]   #####
####                                     #####
#### Note: These ports are default for most Mainsail      #####
#### installations. To use any other port would involve   #####
#### changing the proxy configuration or using directly    #####
#### http://<ip>:<port>/?action=[stream/snapshot]          #####
####                                     #####
```

```
#####  
#####  
##### RTSP Stream URL: ( if enabled and supported ) #####  
##### rtsp://<ip>:<rtsp_port>/stream.h264 #####  
#####  
#####
```

[crowsnest]

log_path: %LOGPATH%

log_level: verbose # Valid Options are quiet/verbose/debug

delete_log: false # Deletes log on every restart, if set to true

no_proxy: false

[cam 1]

mode: ustreamer # ustreamer - Provides mjpg and snapshots. (All devices)

camera-streamer - Provides webrtc, mjpg and snapshots. (rpi + Raspi OS based only)

enable_rtsp: false # If camera-streamer is used, this enables also usage of an rtsp server

rtsp_port: 8554 # Set different ports for each device!

port: 8080 # HTTP/MJPEG Stream/Snapshot Port

device: /dev/video0 # See Log for available ...

resolution: 640x480 # widthxheight format

max_fps: 15 # If Hardware Supports this it will be forced, otherwise ignored/coerced.

#custom_flags: # You can run the Stream Services with custom flags.

#v4l2ctl: # Add v4l2-ctl parameters to setup your camera, see Log what your cam is capable of.

Wie findest du deine Kamera "Device":

Verbinde dich mit Putty oder einem SSH Client zu deinem Pi

Mit diesem Befehl kannst du dir alle Kamerageräte anzeigen lassen

```
v4l2-ctl --list-devices
```

```
pi@voron:~$ v4l2-ctl --list-devices
bcm2835-codec-decode (platform:bcm2835-codec):
/dev/video10
/dev/video11
/dev/video12
/dev/video18

bcm2835-isp (platform:bcm2835-isp):
/dev/video13
/dev/video14
/dev/video15
/dev/video16

C922 Pro Stream Webcam (usb-3f980000.usb-1.4):
/dev/video0
/dev/video1
```

Kamera in Mainsail einfügen:


Öffne das Menü über die Zahnräder oben rechts auf der Weboberfläche:



Navigiere über die Auswahlreiter zu **Webcam**.

Hier ist ein Beispiel meiner Konfiguration

Erstelle Webcam

 ▼

Stream URL

Schnappschuss URL

Dienst

Adaptive MJPEG-Streamer (exp...

 ▼

Ziel FPS

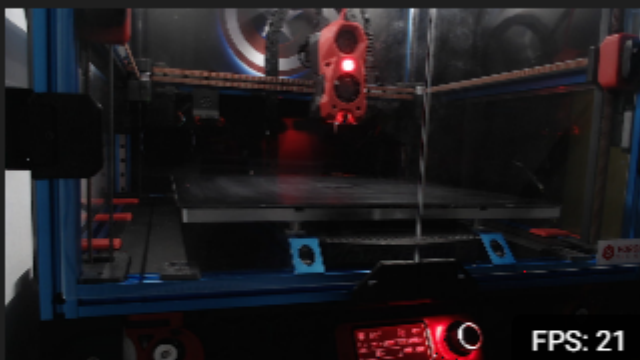
Drehen

0°

 ▼

☐ Webcam waagrecht spiegeln

☐ Webcam vertikal spiegeln



FPS: 21

Stream URL = Pfad zu deiner Kamera.

Für Kamera 1 ist das "/webcam/?action=stream"

Für Kamera 2 wäre es "/webcam2/?action=stream"

Snapshot URL = Pfad für einen Schnappschuss (Timelapse-relevant)

Kamera 1 "/webcam/?action=snapshot"

Kamera 2 "/webcam2/?action=snapshot"

Dienst = Empfehlung "Adaptive MJPEG-STREAMER"

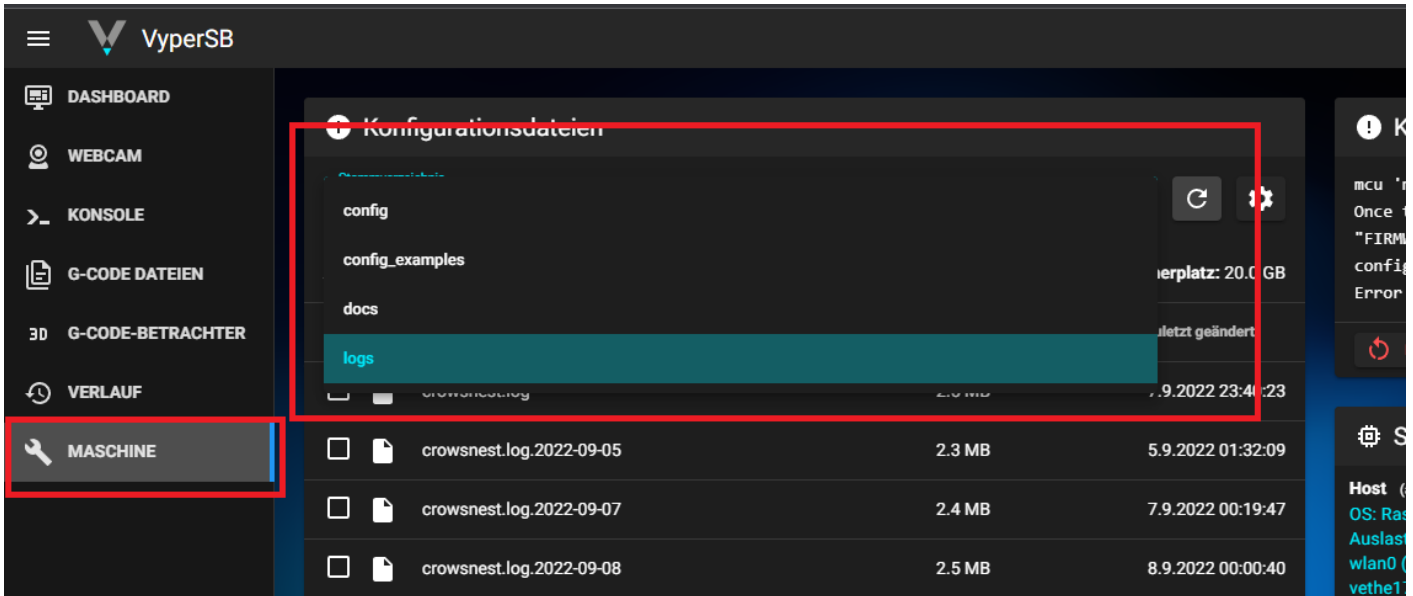
FPS = 30 -50

Probleme?!

Durchsuche unser [FAQ](#) nach deinem Fehler.

Wo finde ich Log Dateien

- Mainsail Dashboard öffnen
- links unter Maschine
- in der oberen Hälfte steht „configs“
- linker Mausklick öffnet Dropdown
- „Logs“ auswählen



The screenshot shows the VyperSB dashboard interface. On the left sidebar, the 'MASCHINE' menu item is highlighted with a red box. The main content area shows the 'Konfigurationsdateien' section, which has a dropdown menu open, highlighting the 'logs' option. Below this, a table lists log files.

File Name	Size	Timestamp
crowsnest.log.2022-09-05	2.3 MB	5.9.2022 01:32:09
crowsnest.log.2022-09-07	2.4 MB	7.9.2022 00:19:47
crowsnest.log.2022-09-08	2.5 MB	8.9.2022 00:00:40

Z-Tilt / Quad Gantry Level

Was ist Z-Tilt und Quad Gantry Level (QGL)

Erklärung

Hier zeige ich euch kurz, was genau die oben genannten Kommandos machen und was der Zweck davon ist.

Kurz gesagt gleicht das Z-Tilt (Für 2 Z-Motoren) und das Quad Gantry Level (4 Motoren) die Lage der "Gantry" oder auch X Führung des Druckkopfes aus.

Es werden meist die äußeren Punkte des Druckbettes angefahren und so die Achse parallel zum Bett gestellt.

Wichtig: Das Druckbett sollte vorher schon parallel zum Gestell oder Boden stehen.

Konsolenausgabe

Dies ist eine Beispielausgabe des QGL:

Making the following Z adjustments:

stepper_z = 0.004953

stepper_z1 = -0.000403

stepper_z2 = -0.000520

stepper_z3 = -0.004030

Average: 7.239051

Actuator Positions:

z: 7.234097 z1: 7.239454 z2: 7.239571 z3: 7.243081

Gantry-relative probe points:

0: 7.236478 1: 7.238353 2: 7.239915 3: 7.240540

probe at 250.000,25.000 is z=7.759460

probe at 250.000,225.000 is z=7.760085

probe at 25.000,225.000 is z=7.761647

probe at 25.000,25.000 is z=7.763522

probe at 250.000,25.000 is z=7.759460

Wie Ihre sehen könnt werden als ersten die in der Konfiguration definierten Punkte angefahren und in Z-Höhe vermessen.

Making the following Z adjustments:

Es werden jetzt für die jeweiligen Motoren Änderungen der Z-Höhe vorgenommen und nochmals die Messpunkte angefahren.

Die Wiederholungen sind abhängig von den von euch definierten Toleranz in der Konfiguration in der Variable:

retry_tolerance: 0.0075

Alle Infos zu den Parametern der Konfiguration findet ihr in der Dokumentation von Klipper.

[Klipper Dokumentation](#)

Mainsail ohne Drucker nutzen

Hallo.

Hier zeigen wir euch, wie ihr die Mainsail Oberfläche nutzen könnt, ohne dass ein Drucker aktiv laufen muss.

Dies kann euch für verschiedene Anwendungen nützlich sein.

Quellen:

- [Beitrag von Meteyou auf dem Mainsail Discord](#)
- [Link zu der original Klipper Dokumentation](#)

1. Klipper-mcu.service anlegen

1. Klipper MCU Service anlegen:

Verbindung via SSH Client zum Pi/Host herstellen

2. In das klipper Verzeichnis navigieren

```
cd ~/klipper/
```

3. Den klipper-mcu Service in das "Anwendungsstammverzeichnis kopieren

```
sudo cp ./scripts/klipper-mcu.service /etc/systemd/system/
```

4. Den klipper-mcu.service als Autostart anlegen

```
sudo systemctl enable klipper-mcu.service
```

Autostart deaktivieren:

```
sudo systemctl disable klipper-mcu.service
```

2. Microcontroller flashen

1. In das Klipper Verzeichnis navigieren

```
cd ~/klipper/
```

2. Die Klipper Firmwareerstellung öffnen

```
make menuconfig
```

3. Als Microcontroller Architektur "**Linux process**" wählen

4. Den Klipper service Stoppen - Firmware flashen - Klipper Service starten

```
sudo service klipper stop
```

```
make flash
```

```
sudo service klipper start
```

3. Nutzer in TTY Gruppe anlegen

Wichtig:

"pi" wird hier als Nutzernamen gewählt. Habt ihr einen andern Nutzernamen, müsst ihr diesen anpassen.

```
sudo usermod -a -G tty pi
```

Pi neustarten

```
sudo reboot
```

4. Printer.cfg anpassen

Ihr müsst jetzt folgenden Eintrag in eurer Printer.cfg anpassen.

Wir empfehlen hier eine separate printer.cfg anzulegen.

Kopiert euch einfach den Inhalt eurer standard printer.cfg in eine neu erstellte "backup.printer.cfg"

```
[include mainsail.cfg]
```

```
[mcu]
```

```
serial: /tmp/klipper_host_mcu
```

```
[printer]
```

```
kinematics: none
```

```
max_velocity: 1
```

```
max_accel: 1
```

Optional:

Printer.cfg per Command wechseln

1. [kiauh](#) installieren

2. gcode shellcommander installieren

3. Config Dateien anlegen

- default-mcu.cfg -> Das ist die normal config Datei

- klipper-mcu.cfg -> Das ist die config Datei ohne Drucker

- "**[include shell_command.cfg]**" muss in beiden cfgs vorhanden sein

4. Macros anlegen

```
[gcode_shell_command load_klipper_mcu]
```

```
command: cp /home/pi/printer_data/config/klipper-mcu.cfg /home/pi/printer_data/config/printer.cfg
```

```
timeout: 2.
```

```
verbose: True
```

```
[gcode_macro LOAD_KLIPPER_MCU]
```

```
gcode:
```

```
    RUN_SHELL_COMMAND CMD=load_klipper_mcu
```

```
    M117 Config changed to klipper-mcu .... Restart Firmware
```

```
    FIRMWARE_RESTART
```

```
[gcode_shell_command load_default_mcu]
```

```
command: cp /home/pi/printer_data/config/default-mcu.cfg /home/pi/printer_data/config/printer.cfg
```

```
timeout: 2.
```

```
verbose: True
```

```
[gcode_macro LOAD_DEFAULT_MCU]
```

```
gcode:
```

RUN_SHELL_COMMAND CMD=load_default_mcu

M117 Config changed to default-mcu Restart Firmware

FIRMWARE_RESTART