

Updating and Flashing the the Toolhead MCU

By Einlander

The Qidi 3 series toolhead is based on the Makerbase THR36 and THR42 toolheads (<https://github.com/makerbase-mks/MKS-THR36-THR42-UTC>). Flashing the qidi toolhead is a similar experience. They both use a Raspberry Pi 2040 MCU and can/are controlled by USB. The major difference is the Qidi toolhead runs 24v over the USB lines. The Qidi printers run a version of Armbian Linux running Debian Buster.

The Process.

Here, we will break down the main steps to update and flash toolhead:

1. Compile the Klipper firmware.
2. Set the toolhead to DFU mode
3. Upload the firmware

Assumptions and Prerequisites

Wherever possible, we will proceed using the easiest method. Though updating and flashing the toolhead can be done completely in Linux, we will assume that a Windows PC will be used. Software that will be used is any SSH client (I think Windows 10 has one built-in). I will assume that you know how to acquire, setup and use ssh.

MAKE SURE TO REMOVE ALL USB DRIVES AND MEMORY CARDS!

They may conflict or make this guide difficult to follow or complete.

Compiling Klipper firmware:

Connect to printer

First we need to log into the printer using ssh as the user mks. You connect by directly connecting to your printer by using it's IP address.

The address format is :

mks@your.ip.address.here

The default password is

makerbase

```
C:\Users\Einlander>ssh mks@192.168.68.110
mks@192.168.68.110's password:

  _ _ _ _ _ | | _ _ _ _ _ ( _ )
 | ' ' ' _ _ | / / _ _ | ' ' _ | | | |
 | | | ' _ _ | < \ _ _ \ _ _ | | |
 | _ | _ | | _ | \ _ _ / _ _ / | _ |
              | _ |
Welcome to Armbian 22.05.0-trunk with bleeding edge Linux 5.16.20-rockchip64

No end-user support: built from trunk

System load:   35%           Up time:           1:15
Memory usage:  19% of 976M   IP:           192.168.68.110
CPU temp:      53°C          Usage of /:     73% of 6.6G

[ General system configuration (beta): armbian-config ]

Last login: Tue Dec 20 06:24:40 2022 from 192.168.68.112
mks@mkspi:~$
```

Set the time

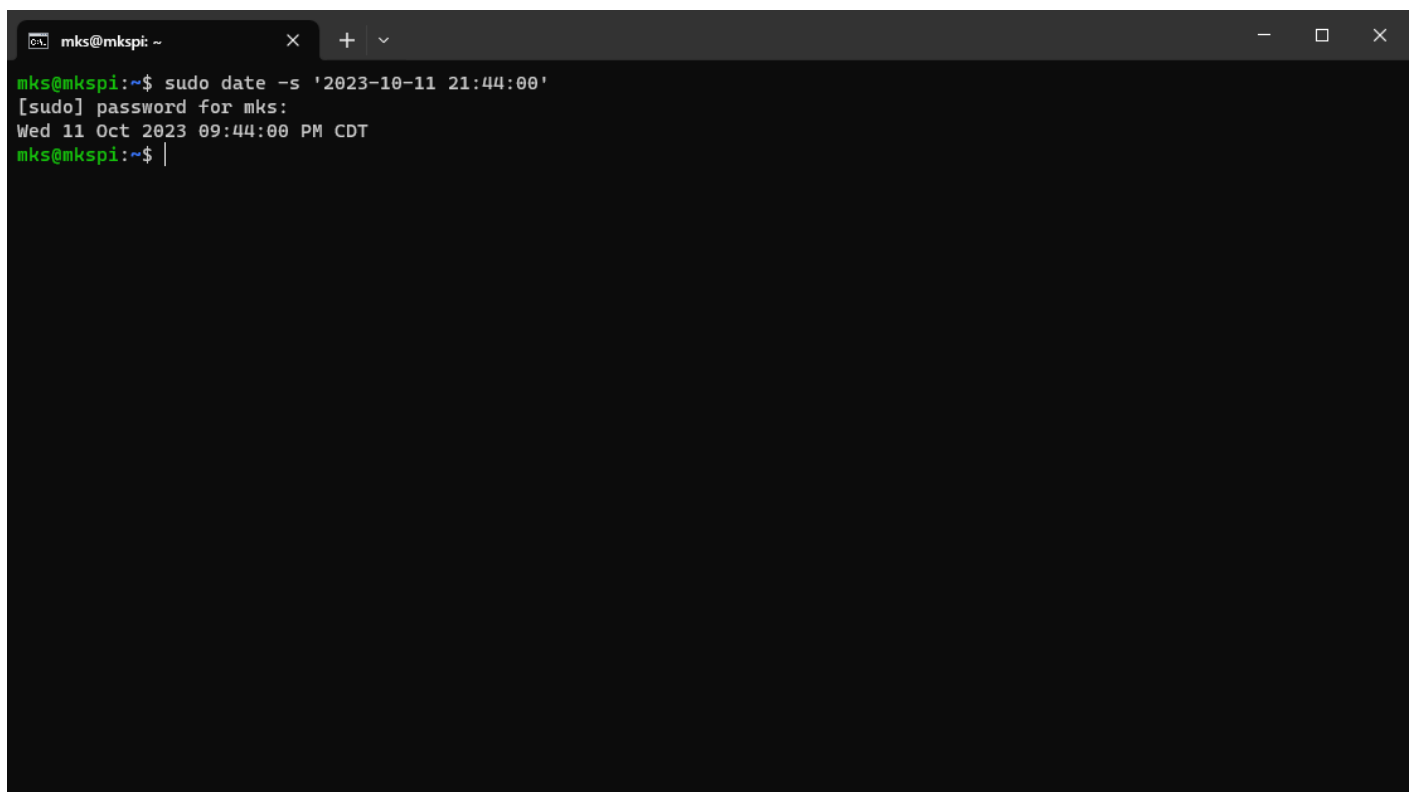
Before we do anything else, we need to set the time.

Failing to do this step may cause the update process to fail or become more difficult.

The command to set the date is

```
sudo date -s 'YYYY-MM-DD HH:MM:SS'
```

the password is: makerbase

A terminal window with a dark background. The title bar shows 'mks@mkspi: ~'. The prompt is 'mks@mkspi:~\$'. The user enters 'sudo date -s '2023-10-11 21:44:00''. The prompt changes to '[sudo] password for mks:'. The user enters the password. The output shows 'Wed 11 Oct 2023 09:44:00 PM CDT'. The prompt returns to 'mks@mkspi:~\$'.

```
mks@mkspi:~$ sudo date -s '2023-10-11 21:44:00'
[sudo] password for mks:
Wed 11 Oct 2023 09:44:00 PM CDT
mks@mkspi:~$
```

You can also try this command to try to set it automatically.

```
sudo date -s "$($(wget -qSO- --max-redirect=0 google.com 2>&1 | grep Date: | cut -d' ' -f5-8)Z"
```

```
mks@mkspi: ~  
mks@mkspi:~$ sudo date -s "$ (wget -qSO- --max-redirect=0 google.com 2>&1 | grep Date: | cut -d' ' -f5-8)Z"  
[sudo] password for mks:  
Sat 11 Nov 2023 08:22:29 AM CST  
mks@mkspi:~$ d|
```

Kiauh

Now, we are at one of the main parts of updating and flashing the toolhead. For this, we will use **KIAUH** the Klipper Installer and Update Helper. In our case, it is already installed on the printer. To run it, we must first type this command:

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

This command will only work when logged in as mks. It will warn you if you use it as root.

KIAUH will ask you if you want to update, type:

```
Y
```

```
mks@mkspi: /  
/=====\  
|           New KIAUH update available!           |  
|-----|  
| View Changelog: https://git.io/JnmlX |  
| It is recommended to keep KIAUH up to date. Updates |  
| usually contain bugfixes, important changes or new |  
| features. Please consider updating! |  
|-----|  
\=====/  
##### Do you want to update now? (Y/n):
```

After it updates, we need to delete some folders.

This assumes you have never gone through this tutorial before. Qidi has customized some of the files, and KIAUH will fail when it finds them. You shouldn't need to do this in future updates. For our purposes, we will delete the directories.

```
sudo rm -r ~/klipper/  
sudo rm -r ~/moonraker/
```

Start KIAUH again by typing:

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

You should have arrived at the Main Menu screen:

```
mks@mkspi: /
=====
[ KIAUH ]
Klipper Installation And Update Helper
=====
[ Main Menu ]
=====
0) [Log-Upload] | Klipper: Installed: 1
                  Repo: Klipper3d/klipper
1) [Install]    |
2) [Update]     | Moonraker: Installed: 1
3) [Remove]     |
4) [Advanced]   | Mainsail: Not installed!
5) [Backup]     | Fluididd: Installed!
                  KlipperScreen: Installed!
6) [Settings]   | Telegram Bot: Not installed!
                  Crowsnest: Not installed!
                  Obico: Not installed!
                  OctoEverywhere: Not installed!
                  Mobileraker: Not installed!
                  Octoprint: Not installed!
v5.0.0-67      | Changelog: https://git.io/JnmlX
Q) Quit
##### Perform action:
```

First, we type 3 to get the Remove Menu.

```
mks@mkspi: ~
=====
[ KIAUH ]
Klipper Installation And Update Helper
=====
[ Remove Menu ]
=====
INFO: Configurations and/or any backups will be kept!
=====
Firmware & API: | 3rd Party Webinterface:
1) [Klipper]    | 8) [OctoPrint]
2) [Moonraker] |
Klipper Webinterface: | Webcam Streamer:
3) [Mainsail]   | 9) [Crowsnest]
4) [Mainsail-Config] | 10) [MJPG-Streamer]
5) [Fluididd]   | Other:
6) [Fluididd-Config] | 11) [PrettyGCode]
                  | 12) [Telegram Bot]
                  | 13) [Obico for Klipper]
Touchscreen GUI: | 14) [OctoEverywhere]
7) [KlipperScreen] | 15) [Mobileraker]
                  | 16) [NGINX]
B) « Back
##### Perform action:
```

Then we remove 1 Klipper , 2 Moonraker, and 3 Fluididd.

When it returns to the Remove Menu, we will Type B to return to the main menu.

Type 1 for the installation menu

Type 1 to install Klipper.

```
mks@mkspi: ~  
===== [ KIAUH ] =====  
Klipper Installation And Update Helper  
===== [ Installation Menu ] =====  
You need this menu usually only for installing  
all necessary dependencies for the various  
functions on a completely fresh system.  
-----  
Firmware & API:      | 3rd Party Webinterface:  
1) [Klipper]         | 6) [OctoPrint]  
2) [Moonraker]       |  
-----  
Klipper Webinterface: | Other:  
3) [Mainsail]         | 7) [PrettyGCode]  
4) [Fluidd]           | 8) [Telegram Bot]  
-----  
Touchscreen GUI:     | 9) [Obico for Klipper]  
5) [KlipperScreen]    | 10) [OctoEverywhere]  
-----  
                    | 11) [Mobileraker]  
Webcam Streamer:     |  
12) [Crowsnest]      |  
-----  
                    | B) « Back  
##### Perform action:
```

Type 1 to use Python 3.x

```
mks@mkspi: ~  
===== [ KIAUH ] =====  
Klipper Installation And Update Helper  
=====
```

Initializing Klipper installation ...

```
=====
```

Please select your preferred Python version.
The recommended version is Python 3.x.

```
=====
```

1) [Python 3.x] (recommended)
2) [Python 2.7] (legacy)

```
=====
```

B) « Back

```
=====
```

Select Python version: 1

Type 1 to run one instance of Klipper

```
mks@mkspi: ~  
===== [ KIAUH ] =====  
Klipper Installation And Update Helper  
=====
```

Initializing Klipper installation ...

```
=====
```

Please select your preferred Python version.
The recommended version is Python 3.x.

```
=====
```

1) [Python 3.x] (recommended)
2) [Python 2.7] (legacy)

```
=====
```

B) « Back

```
=====
```

Select Python version: 1
[>] Python 3.x

```
=====
```

Please select the number of Klipper instances to set
up. The number of Klipper instances will determine
the amount of printers you can run from this host.

WARNING:
Setting up too many instances may crash your system.

```
=====
```

B) « Back

```
=====
```

Number of Klipper instances to set up: 1

Wait for it to install.

When KIAUH returns to the Installation menu, Type 2 to install Moonraker and wait for the install. Then install 4 Fluidd, answering yes to the questions it asks.

```
mks@mkspi: ~  
[✓ OK] brltty service masked!  
##### Installed brltty-udev package detected, masking brltty-udev service ...  
[✓ OK] brltty-udev service masked!  
##### Installed ModemManager package detected, masking ModemManager service ...  
[✓ OK] ModemManager service masked!  
#####  
Klipper has been set up!  
#####  
/=====\  
|          [ Installation Menu ]          |  
|-----|  
| You need this menu usually only for installing |  
| all necessary dependencies for the various |  
| functions on a completely fresh system. |  
|-----|  
| Firmware & API: | 3rd Party Webinterface: |  
| 1) [Klipper] | 6) [OctoPrint] |  
| 2) [Moonraker] | |  
| Klipper Webinterface: | Other: |  
| 3) [Mainsail] | 7) [PrettyGCode] |  
| 4) [Fluidd] | 8) [Telegram Bot] |  
| | 9) [Obico for Klipper] |  
| | 10) [OctoEverywhere] |  
| | 11) [Mobileraker] |  
| Touchscreen GUI: | Webcam Streamer: |  
| 5) [KlipperScreen] | 12) [Crowsnest] |  
|-----|  
| B) « Back |  
|-----|  
\=====/  
##### Perform action: |
```

Type B to return to the main menu and 2 to go to the Upgrade Menu.

Type 4 to upgrade Fluidd, then return to the main menu.

Type 4 to go to the Advanced Menu.

In the Advanced Menu, we just want to build the firmware. Select: 2

```
mks@mkspi: /
=====
[ 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] | Extras:
4) [Build + Flash] | 8) [G-Code Shell Command]
5) [Get MCU ID]   |
B) « Back
=====
##### Perform action: 2
```

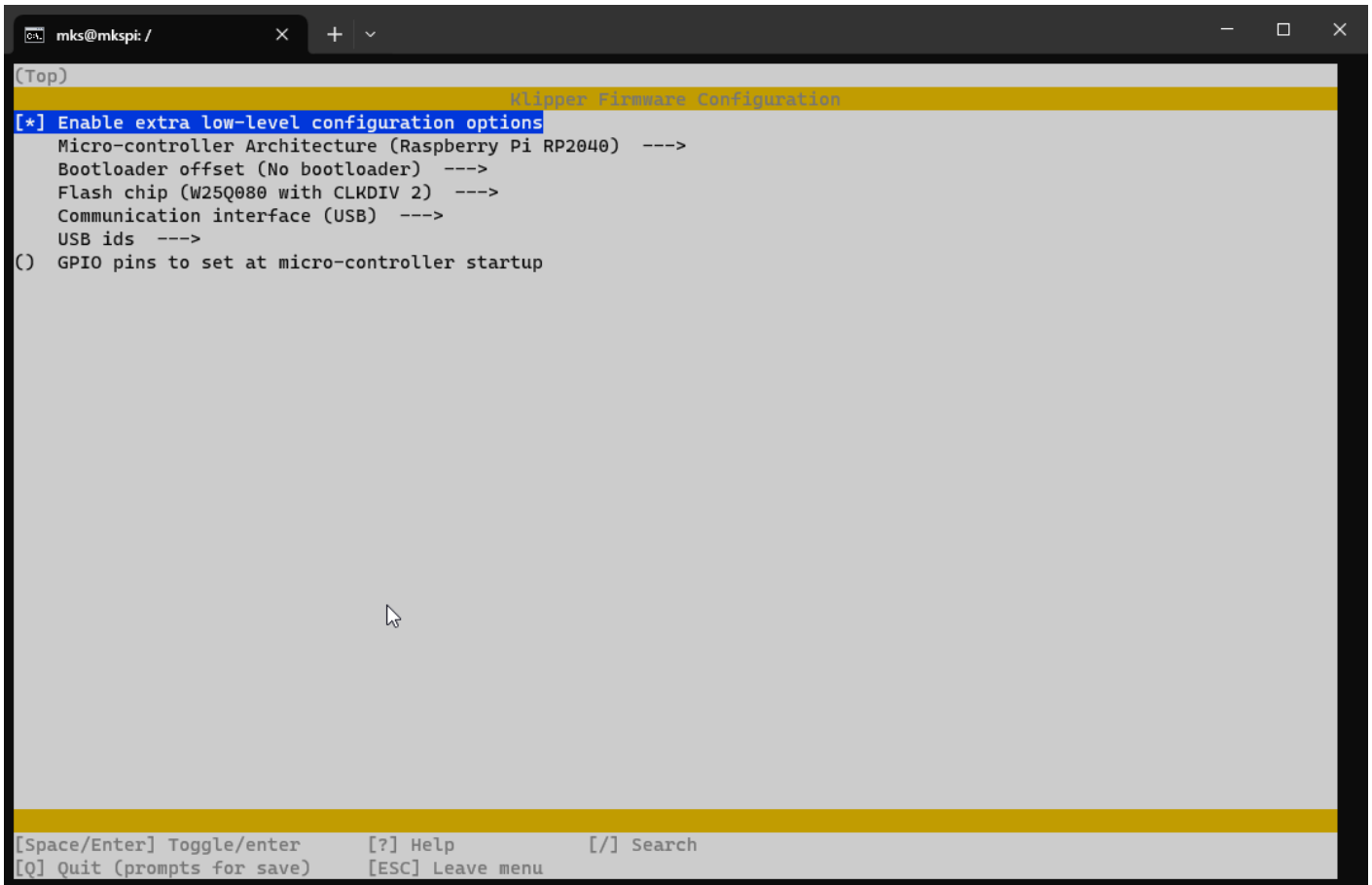
MCU Settings

The next screen you will see is the Klipper Firmware Configuration screen. Here, we will need to change the settings to match our toolhead mcu. Press space or enter to select items.

Select: Enable extra low-level configuration options

Select: Micro-controller Architecture (Raspberry Pi RP2040)

Then, leave everything else alone.

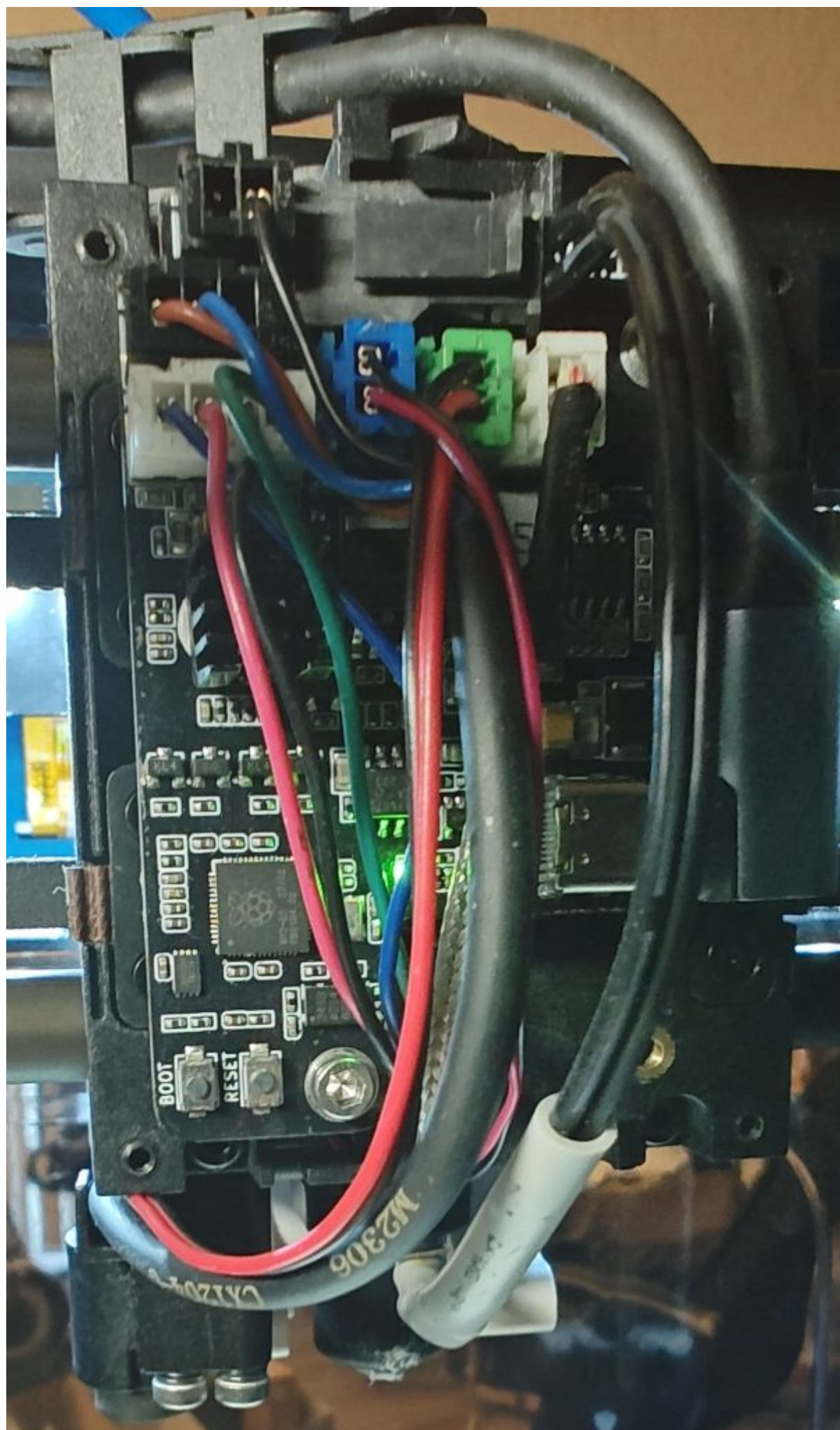


Press Q, then Y to save. The firmware will be built, and it will return us to the Advanced Menu.

Now, we want to quit KIAHU. Press B to return to the Main Menu and Q to quit.

Setting the toolhead to DFU mode

Setting the toolhead to DFU mode requires you to remove the back cover of the toolhead. On the bottom left, there are two buttons labeled BOOT and RESET.



The easiest way to put it into DFU mode is to press and hold the BOOT button, press and release the RESET button, and then release the BOOT button.

You have to press the reset button while still holding the boot button.

To check if your toolhead is in DFU mode, type:

```
lsusb  
lsblk
```

If you see an OpenMoko, Inc. device, it is not in DFU mode.

```
mks@mkspi: /  
mks@mkspi:/$ lsusb  
Bus 005 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 004 Device 004: ID 0bda:b711 Realtek Semiconductor Corp.  
Bus 004 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub  
Bus 002 Device 002: ID 1a40:0101 Terminus Technology Inc. Hub  
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 001 Device 008: ID 1d50:614e OpenMoko, Inc.  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
mks@mkspi:/$ lsblk  
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
mmcblk1         179:0    0   7.3G  0 disk  
├─mmcblk1p1      179:1    0   256M  0 part /boot  
└─mmcblk1p2      179:2    0    6.9G  0 part /  
mmcblk1boot0    179:32   0     4M   1 disk  
mmcblk1boot1    179:64   0     4M   1 disk  
zram0           252:0    0  488.2M  0 disk [SWAP]  
zram1           252:1    0     50M  0 disk /var/log  
mks@mkspi:/$
```

If you do not see an OpenMoko, Inc. device AND a sda disk with a MOUNTPOINT of /home/mks/gcode_files/sda1 the toolhead is in DFU mode.

The disk name may also be named sda, sdb, sdc, etc.

```
mks@mkspi: ~  
mks@mkspi:~$ lsusb  
Bus 005 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub  
Bus 004 Device 003: ID 0bda:b711 Realtek Semiconductor Corp.  
Bus 004 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub  
Bus 002 Device 002: ID 1a40:0101 Terminus Technology Inc. Hub  
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 001 Device 002: ID 2e8a:0003  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
mks@mkspi:~$ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sda           8:0    1   128M  0 disk  
└─sda1        8:1    1   128M  0 part /home/mks/gcode_files/sda1  
mmcblk1     179:0    0    7.3G  0 disk  
├─mmcblk1p1  179:1    0   256M  0 part /boot  
├─mmcblk1p2  179:2    0    6.9G  0 part /  
mmcblk1boot0 179:32   0     4M   1 disk  
mmcblk1boot1 179:64   0     4M   1 disk  
zram0        252:0    0 488.2M  0 disk [SWAP]  
zram1        252:1    0     50M  0 disk /var/log  
mks@mkspi:~$
```

Uploading the Firmware

Now that the toolhead is in DFU mode, it will show up as a USB drive to the printer. Luckily QIDI decided to auto-mount USB drives for us.

To upload the firmware, we simply need to copy the compiled Klipper firmware file to the toolhead.

To do that type:

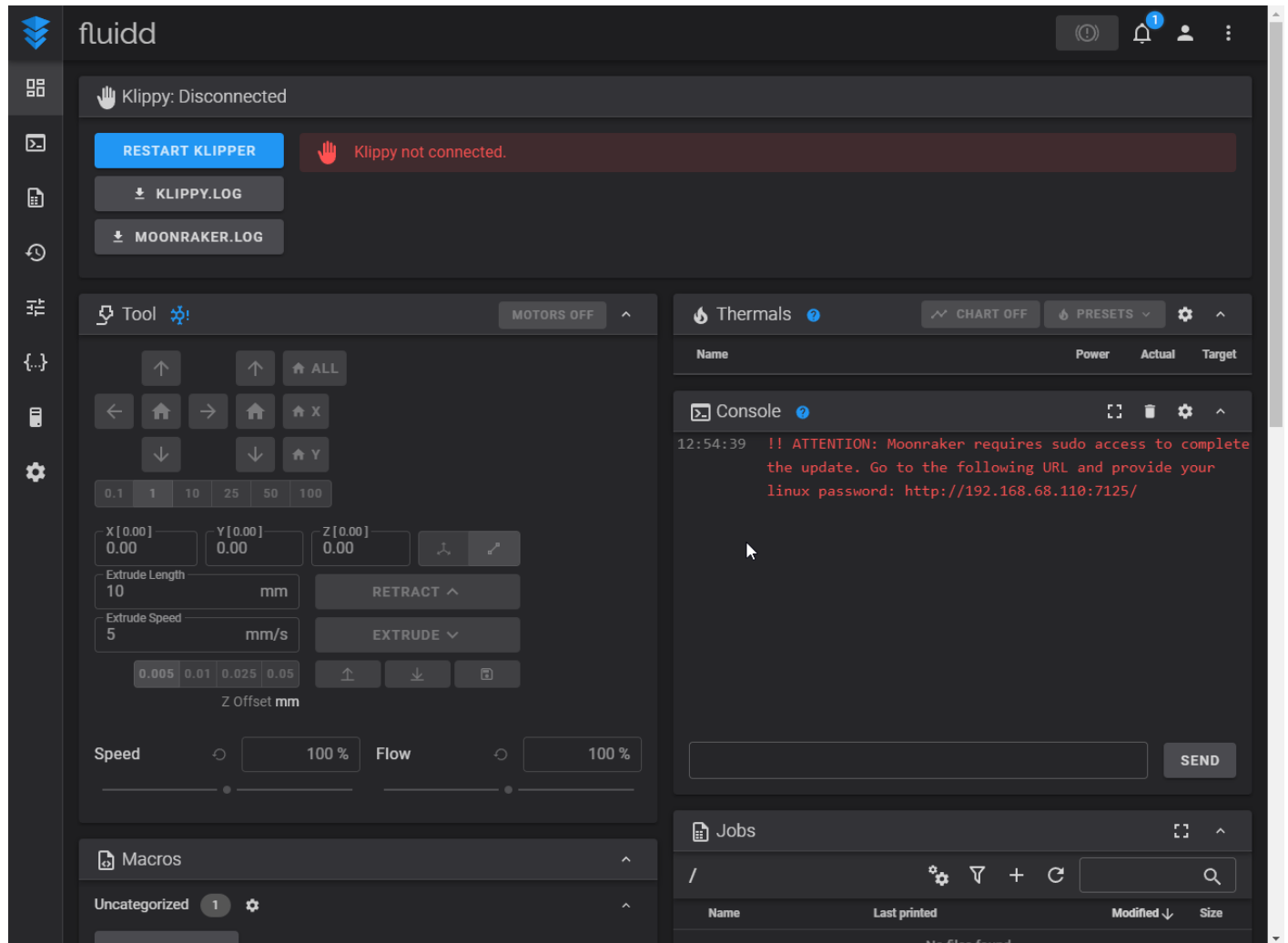
```
cp ~/klipper/out/klipper.uf2 ~/gcode_files/sda1/
```

As soon as the file is copied, the toolhead will no longer be in DFU mode and it will disconnect the drive.

Finishing Up

Completing Moonraker Update

Turn the printer off and on. Then, connect to the web interface. In the console, you may see Moonraker prompting you to go to a specific IP address to complete the update.



The screenshot displays the fluidd web interface. At the top, the status bar shows "Klippy: Disconnected". Below this, there are buttons for "RESTART KLIPPER", "KLIPPY.LOG", and "MOONRAKER.LOG". A red error message states "Klippy not connected.".

The main interface is divided into several panels. On the left is the "Tool" panel with a "MOTORS OFF" button and various movement controls. The center panel contains settings for "Extrude Length" (10 mm), "Extrude Speed" (5 mm/s), and "Z Offset mm" (0.005 to 0.05). The right panel shows "Thermals" and a "Console" window.

The "Console" window displays a message at 12:54:39: **!! ATTENTION: Moonraker requires sudo access to complete the update. Go to the following URL and provide your linux password: <http://192.168.68.110:7125/>**

At the bottom, there is a "Jobs" panel showing a list of files with columns for Name, Last printed, Modified, and Size. The current list is empty, showing "No files found".

Welcome to Moonraker

You may have intended to navigate to one of Moonraker's front ends, if so check that you entered the correct port in the address bar.

Moonraker Sudo Password Request

Request IP: 192.168.1.100
Authorized: 75011d-dirty
CORS Enabled: 2
sconnected

Service Name: moonraker.service
Host Name: mkspi
Host IP Address: 192.168.68.110
Sudo password required to update Moonraker's systemd service. Please enter the password for linux user **mks**:

Your device is authorized to access Moonraker's API.

CORS is enabled. Cross origin requests will be allowed for origins that match one of the patterns specified in the 'cors_domain' option of the [authorization] section.

Moonraker is not currently connected to Klipper. Make sure that the klipper service has successfully started and that its unix is enabled.

Announcements

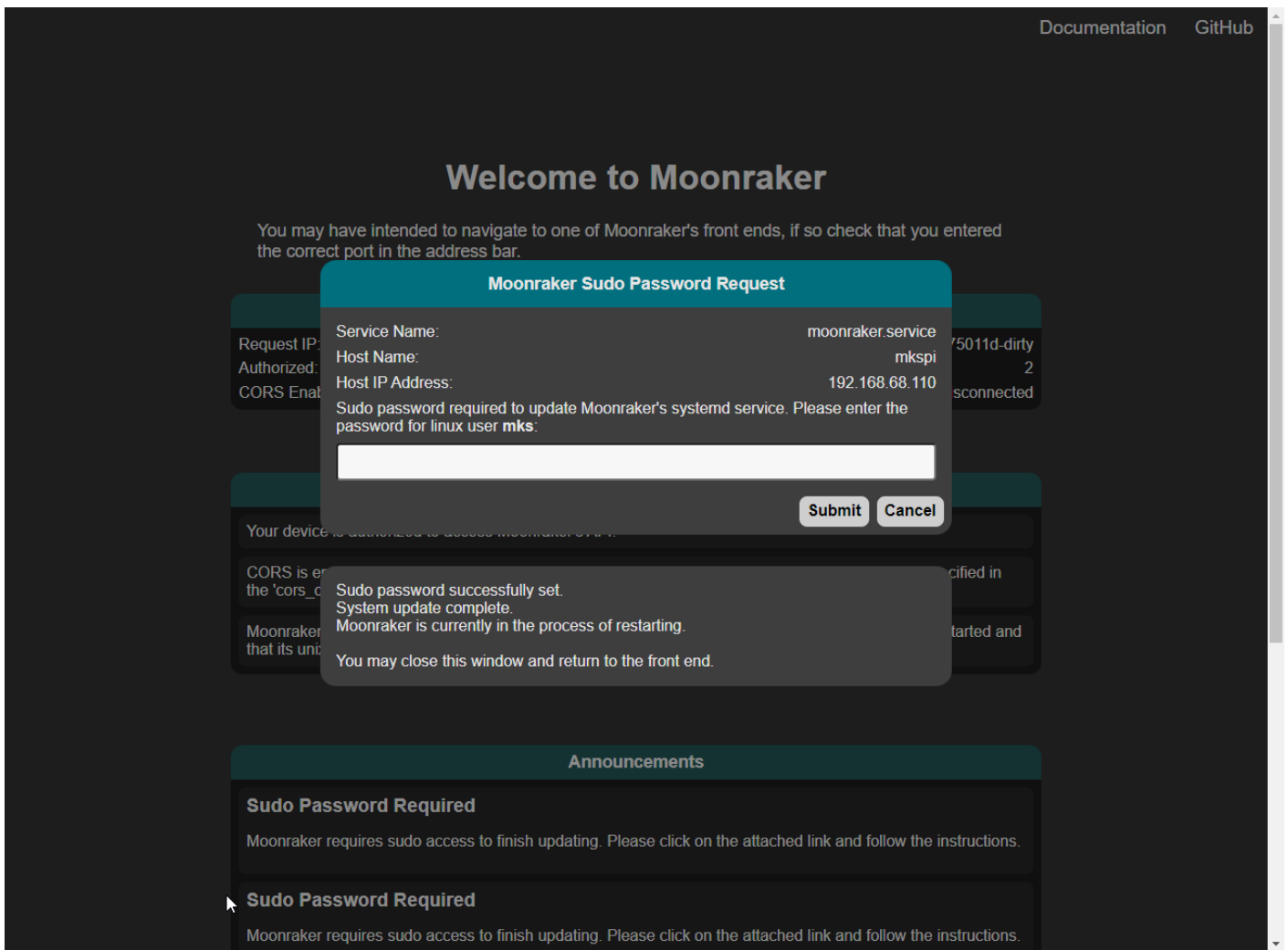
Sudo Password Required

Moonraker requires sudo access to finish updating. Please click on the attached link and follow the instructions.

Sudo Password Required

Moonraker requires sudo access to finish updating. Please click on the attached link and follow the instructions.

Type your mks user password and close the window.



Restoring Settings

The new version of Klipper and Moonraker keep their save settings in a different location. We will copy them there.

```
cp -r ~/klipper_config/* ~/printer_data/config
```

Connecting Klippy

The location of the klippy_uds_address has changed. If it tells you "klippy not connected" this may help repair it. From the web interface, edit moonraker.conf to change it to this:

```
klippy_uds_address: ~/printer_data/comms/klippy.sock
```

Restoring Qidi Gcode functionality

QIDI added a feature that automatically mounted USB drives which is nice. But in their infinite wisdom, they decided to hard-code the location in their client. So, updating Klipper and Moonraker breaks it. There are 2 methods to fix it.

Patch it:

```
sed -i 's/printer_data/gcodes/gcode_files/g' ~/printer_data/systemd/moonraker.env
```

This method will keep the functionality and directories as it was, but may break in a future update.

Create a Symlink:

To fix it, a folder needs to be deleted and the old folder symlinked into its place, and the config changed. This method should last longer.

```
rm -r /home/mks/printer_data/gcodes/
```

```
ln -s /home/mks/gcode_files /home/mks/printer_data/gcodes
```

```
sed -i 's:~/gcode_files:~/printer_data/gcodes:g' ~/printer_data/config/printer.cfg
```

Updating the printer.cfg

Some commands in printer.cfg have been changed. To update them, run:

```
sed -i 's:printer.probe\[\"x_offset\"\]:printer.configfile.settings.probe.x_offset:g' ~/printer_data/config/printer.cfg
```

```
sed -i 's:printer.probe\[\"y_offset\"\]:printer.configfile.settings.probe.y_offset:g' ~/printer_data/config/printer.cfg
```

Precautions and incompatibilities

The versions of Klipper, Moonraker, and Fluidd installed will be updated. Avoid using the Qidi updates, it will overwrite the updated installation and the versions may be incompatible.

There is an issue with saving the z offset if you run the official/mainline version of Klipper. The software Qidi runs on the machine saves it's own copy of the zoffset. It interferes with klipper by adding it's copy of the zoffset to klippers zoffset resulting in either printing in the air, or grinding into your print bed.

Revision #5

Created 1 January 2024 10:15:04 by McSneaky

Updated 1 January 2024 10:26:41 by McSneaky