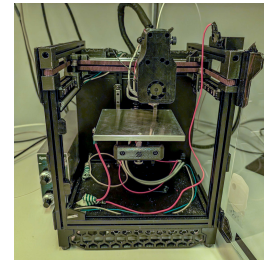


Nov 19, 2024

Conversion of a Voron FDM printer to a MEWron melt electrowriting printer

DOI

<https://dx.doi.org/10.17504/protocols.io.q26g71kk3gwz/v1>



Sönke Menke¹, Biranche Tandon¹, Juergen Brugger¹

¹Microsystems Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, CH-1015 Switzerland



Sönke Menke

EPFL

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.q26g71kk3gwz/v1>

Protocol Citation: Sönke Menke, Biranche Tandon, Juergen Brugger 2024. Conversion of a Voron FDM printer to a MEWron melt electrowriting printer. **protocols.io** <https://dx.doi.org/10.17504/protocols.io.q26g71kk3gwz/v1>

License: This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited



Protocol status: Working

We use this protocol and it's working

Created: September 23, 2024

Last Modified: November 19, 2024

Protocol Integer ID: 108179

Keywords: MEW, melt electrowriting, Printer modification, Voron, FDM, mewron melt electrowriting printer this protocol, voron fdm printer, melt electrowriting printer, voron printer, electrowriting printer, fdm printer, mewron melt, mewron, voron trident

Funders Acknowledgements:

Swiss National Science Foundation

Grant ID: 200021_219481

Abstract

This protocol aims to help researchers and enthusiasts in building a melt electrowriting printer (MEWron) from a Voron printer.

The modification of a FDM printer can potentially be done with many different types but is shown here on a Voron Trident.

Image Attribution

CC-BY Sönke Menke

Materials

- Motor with high gear ratio (e.g. [stepperonline Nema 11 27:1 Planetary gearbox motor](#))
- Toothed motor gear (e.g. [Bondtech](#))
- ceramic sleeves (e.g. [Nonporous Alumina Ceramic Tube 3/8" OD, 1/4" ID, 24" Long](#)) one for thermistor, one for heater, check sizes!
- Stepper driver (e.g. <https://www.fysetc.com/products/tmc2209-v3-0-uart-stepping-motor-driver-stepsticks-mute-driver-256-microsteps-current-2-8a-peak-vs-tmc2208-tmc2100>)
- Jumper cables
- Soldering iron

Safety warnings

❗ In addition to the already substantial and often underappreciated dangers of building and modifying a 3D printer, which involves mains voltage wiring and manually installing and configuring hundreds of watts of electrical heating power into a small box, melt electrowriting (MEW) further introduces a high voltage hazard in the same printer. We thus ask readers to be mindful of these risks and to heed the warnings outlined in the official Voron manuals as well as the following information on safely interacting with high voltage for MEW.

Melt electrowriting requires an electrical potential in the low Kilovolt (kV) regime, and a properly designed system can achieve this easily while operating at below 10 Microamperes (μA) of current. Due to availability, the most used high voltage power supplies have a maximum output of approximately 10 kV and 1 mA, but have their maximum current output reduced to 1% (10 μA) for safe operation in case of arcing or electric shock. Since we anticipate readers to build and experiment with these machines, we ask the readers to err on the side of caution and inform themselves thoroughly based on their specific high voltage sources of the dangers and intricacies of their specific setup before they implement their ideas in practice, and consider safety interlocks, warning lights and proper signage an essential part of their experimental work.

Warning taken from the SI of "MEWron: an Open-Source Melt Electrowriting Platform"
(<https://doi.org/10.1016/j.addma.2023.103604>)

Before start

Read the full protocol before starting and make sure you are accustomed to the work you will perform. In some regions, work on high voltage may only be performed by authorised personnel.

We always recommend consulting professionals before working on high voltage electrical wiring.

Talk to your local Health and Safety coordinator before conducting any electrical work.

We take no responsibility for any claims, damages, injuries, death, or other liabilities whether in an action of contract, tort or otherwise, arising from, out of or in connection with the protocol or the use or other dealings in the protocol.

Building of a Voron

- 1 Build a Voron of your choice (<https://vorondesign.com/>).
We recommend using a Raspberry Pi, Fysetc Spyder board, and Fysetc Stepper drivers. This protocol has been used on Octoprint, but can potentially be modified for other OS. We also recommend having 4 wires for the motor from the board to the head installed during building where the stepper gets connected so it can be swapped easily.
- 2  [go to step #11](#) If you already know where you want to apply high voltage go to step 11 and prepare the printer for future modifications.
- 3 Make sure it is fully functional and you have printed a Benchy.
- 4 Request a serial number (<https://www.reddit.com/r/voroncorexy/>).
- 5 Remove panels to access printhead and controller board.

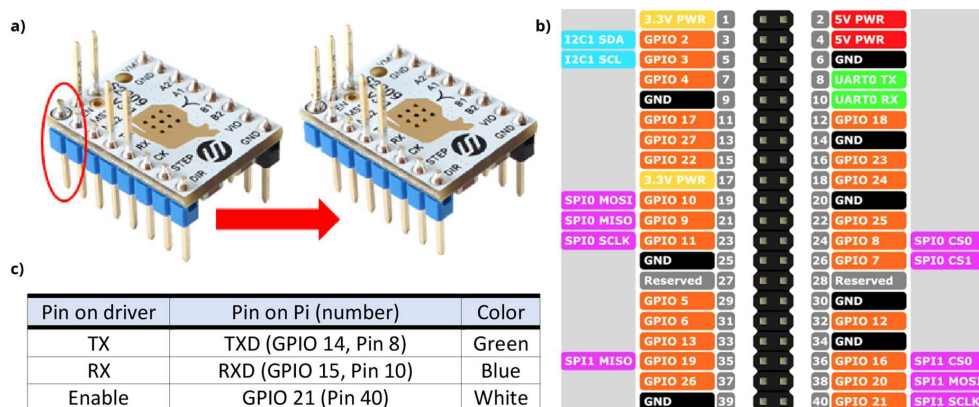
1w

*

*

Hardware modification

- 6 On the stepper driver, invert the enable pin to be able so it makes no connection to the control board and can be accessed from the top.
- 7 Add pins to the RX and TX pins as well to be able to connect to it from the top.



a) Example of extruder stepper driver board inversion, b) Raspberry PI GPIO pins overview, c) summary of stepper motor driver pins to the Raspberry PI GPIO connections. Image taken from Reizabal et al. (2023) under CC-BY license.



For Fysetc S2209 V3.1 driver, no further modification is needed. For others, like the Fysetc S2209 V4.0 driver, a 1 k Ω resistor must be added between the TX pin and Pi.

8 Add the stepper driver to the Spyder board in the location of the extruder stepper driver.

9 For the printhead, print the holder of the motor from the Github respo or create your own if you use a different motor.

Dataset

MEWron

NAME

<https://github.com/mewron/mewron>

LINK

10 Remove the old motor and add the new one.

For testing, it might be best to plug the old motor into the board so functionality can be tested more easily.

11 Add a ground or HV cable to the print bed. this can be done by drilling a tapped hole into the bed.

Add a ground or HV cable to the head. We recommend a tapped hole in the head for this.

 [go to step #2](#) If you came here during the initial build, go back to the beginning.

Make sure that the connections are safe and that there are no exposed HV wires where not intended. Make sure the printer is properly grounded and install more ground connections if needed.

12 Isolate the charged parts from the rest of the printer. At the hotend, we recommend placing heater and thermistor in ceramic sleeves. At the bed we had no issues thus far using the bed as is with heather glued to it.

13 Change the nozzle to a different one if needed.

14 Turn the printer on and check basic functionality (extrusion will not work anymore). Apply HV to check for issues with it.





Safety information

When applying high voltage, start at a low voltage and slowly increase it while keeping nozzle and collector at a safe distance. Do NOT touch charged parts of the printer. Use a multimeter to test that the HV is only where intended.

Software modification

15

Start your printer and use any SSH-shell to log into the printer.

Run the following commands

```
sudo apt update
sudo apt full-upgrade
/home/pi/oprint/bin/python3 -m pip --disable-pip-version-check
install https://github.com/kantlivelong/OctoPrint-
GCodeSystemCommands/archive/master.zip --no-cache-dir
```

Now the serial port has to be enabled:

```
sudo raspi-config
```

Navigate and select: 3. Interface options → I6.Serial port → Login shell to be accessible over serial? → No → Serial port hardware to be enabled? → Yes → Finish.

16

The motor runs mostly on the script by Chr157i4ns TMC2209_Raspberry_Pi library, which needs to be added to the system:

```
git clone https://github.com/Chr157i4n/TMC2209_Raspberry_Pi
```

Dataset

TMC_2209_Raspberry_Pi library

NAME

https://github.com/Chr157i4n/TMC2209_Raspberry_Pi^{LINK}

Now you can test his codes and see if the printer works. You might need to change enable pin in the code to the pin on the Pi you are connecting the enable pin of the driver



to (see Step 7). As we don't connect neither the DIR, STEP, nor CK pin, their pins can be ignored.

First the **demo_script_01_uart_connection.py** script can be run in the ssh-shell:

```
python3 FILEPATH/demo_script_01_uart_connection.py
```

please replace filepath with the one on your printer

Expected result

This should not return any error.

Similarly demo scripts 2 and 3 can be run.

Expected result

After demo script 2 is run the following should be visible: As we do not connect the STEP and DIR pin, when using the test script 2 an error for those pins should be shown but not for pin EN.

During the running of demo script 3, the motor should move 6 times, one revolution back and forth. For this we recommend using the pancake stepper driver that came with the voron.

Note

Issues with "module serial has no attribute":

This following code might help:

```
pip3 install bitstring  
pip3 uninstall serial  
pip3 uninstall pyserial  
pip3 install pyserial  
sudo apt-get install python3-pip  
sudo pip3 install gpiozero
```

- 17 Now move the speed.py and speed.sh files from github into the home/pi/TMC2209_Raspberry_Pi folder on the Pi.



Dataset

MEWron

NAME

<https://github.com/mewron/mewron>

LINK

Connect to the printer in Octoprint and go to settings → GCODE system commands. Configure OCTO1 to be "sh /home/pi/TMC2209_Raspberry_Pi/speed.sh". Along the way, you should check if speed.py is working from the ssh shell, as well as speed.sh. Now you can check in the Octoprint terminal if OCTO1 works (e.g. with OCTO1 1000).

As the codes are intended for the motor with high microstep resolution, we recommend using the other stepper motor now. Double check in the speed.py file if the enable pin is correct as well as the microstep resolution.

Note

Issues with motor heating:

Sometimes, the motor can start heating strongly with no apparent reason when trying to run the codes. During first testing keep an eye on this and shut the system down if needed to prevent the motor from being damaged. This might be solved by reinstalling the software, lowering the current of the motor in the speed.py file (under `tmc.set_current(*number*)` in the file) or testing different settings.

- 18 To control the filament MEWron extrusion, an extrusion speed value (E) must be sent into the Octoprint command board by running the *OCTO1 <E>* command (e.g. *OCTO1 1000*, to set an E value of 1000). *OCTO1 0* commands must be used to stop the extrusion. Additionally, the E value can be included in the g-code (*OCTO1 <E> command*), to dynamically modify the extrusion rate throughout a print job.
- 19 Via the formula published in Reizabal et al. (2023) the flow rate can be calculated from the E value if the same components are being used:

$$Q(\mu\text{L}/\text{min}) = 5.62 \times 10^{-4} \cdot E$$



Protocol references

A. Reizabal, T. Kangur, P. G. Saiz, S. Menke, C. Moser, J. Brugger, P. D. Dalton, S. Luposchainsky, Additive Manufacturing 2023, 71, 103604. DOI [10.1016/j.addma.2023.103604](https://doi.org/10.1016/j.addma.2023.103604)

MEWron github (University of Oregon): <https://github.com/mewron/mewron>

Stepper driver wiki: <https://wiki.fysetc.com/Silent2209/>

Stepper driver github: https://github.com/Chr157i4n/TMC2209_Raspberry_Pi