

dr4wbac_k wiki (jakob, nikl,
sophia, tem)

dr4wbac

TODOs

Mechanical

- ☒ End V-Kugellager
- ☒ Motor-halterung
 - ☒ Capston Halterung
- ☐ improve motor-halterung to circumvent flex
- ☐ add end stops to motorhalterung
- ☒ Mittlere Halterung für den Schlitten
- ☐ enclosing ramps (jakob)
- ☐ feet

- ☐ Finale Teile herstellen & assembly
- ☒ Händisch bewegen + Stresstest
- ☐ teile zuschneiden

optional (wenn nicht, gaffern wir nen stift fest und generieren gcodes ohne absetzen)

- ☐ Kopfhalterung
- ☐ Kabelführung
- ☐ Stifthalterung

IMPORTANT:

- Capstion Line muss auf einer ebene sein
- Check, das alles stabil ist

Electrical

- RAMPS testen
 - ☒ Basis RAMPS test
 - ☒ Marlin auf RAMPS und g-code ausführen
 - ☐ Marlin für drawback konfigurieren
- ☐ Motorsteuerung PCB (URUMBU?)
- ☐ PCB für verbindung der Motoren und PC?
- ☐ Eigene PCB flashen
 - ☐ URUMBU testen
 - ☐ Klipper?
- ☐ toolpath generation

Requirements

- size for painting a3 paper
- capstan drive
- h bot for movement
- easily to assemble / disassemble
 - removable thread of the capstan drive

Roadmap

Syntax: 1, 2, 1.1

- task 1 and 2 are independent from each other
- task 1.1 can only be executed after task 1

- 1 cad & assemble all parts
- 1.1 print parts (until friday ev)
- 2 power electronics
 - 2.1 user interface (svg to motor control code)
- 3 toolpath planning: svg in -> toolpath out
- 100 make video

availability friday

- matthias 9 bis 14.30, ab 18.00
- niklas 15.00 bis ...
- jakob ab 10.00

next steps

- friday 12.00: all single parts modeled
- friday 15.00: all parts assembled (the other person does driver)

References

- https://github.com/SeanP2001/Urumbu_USB_Stepper_Motor
- [cable carriers 3d printable](#)
- [urumbu further work](#)
- [sampo](#)
- [OSAP](#) web framework for talking with usb-driven hardware modules (e.g. motors), shoving everything you would make on pcbs onto the desktop for rapid prototyping. overkill for our project.
- [capston drive](#)
- [alternative draw bot with v guides](#)
- [flexible-XYstage](#)
- [Urumbu IO](#)
- [UrumbubotXY 2.0](#)

- Laid Up Gantries with Capstan Drives
- Other Capston mechanism

Other drawing bots

- Watercolorbot
- axidraw
 - Youtube
- Easydraw

Mechanical Design

- TODO insert niclas' sketches
- cable carrier
- CAD organization
- keep folder structure as flat as possible
- Have everything as .FCStd. Convert if needed.
- one FCStd file per part/assembly
- reason: one part can be included multiple times easily (especially if you download separate parts)
- motor
- pen holder:
 - <http://www.makelangelo.com/>
 - <https://wiki.opensourceecology.org/wiki/Ferdi>

BOM

take 3d models from urumbot

model in cad or find 3d models from somewhere else, e.g. from here

probably easy to get

- 1. v-bearing (5x)
- 2. capstan (2x) (take from urumbot) □
- 5. idler (2x) (take from urumbot) □
- 7.1. extrusion vertical (1x) □
- 7.2. extrusion horizontal (1x) □

- 4. stepper motor (2x) □
- 10. wire (1x)
- 11. servo motor □

probably model ourselves

- 3. mounting plate v bearing (1x)
- 7. sledge □ + assemble it (2x)
- 11. extrusion cap head (1x)
- 12. flexture
- 13. mounting plate servo (1x)
- 14. mounting plate stepper motor (2x)
- 16. pcbs

do not model

- 5. motor enclosure right (1x) (DEPRECATED)
- 6. motor enclosure left (1x) (DEPRECATED)
- 8. sledge case (1x) (DEPRECATED)

to ask ferdi

- what do you think of our design?
 - you somehow have to attach the two carriers. do that in 4 points. maybe print clamps for that
 - for the upper carrier, design something on your own. reason is, you need to add somehow the v wheels
- which material do we take for attaching motors and v bearings to the end of the extrusions?
 - alimunum sheets. thickness 4 to 5mm (we have 5mm, 41mm width). they are in the top right shelf next to the carvera
- any ideas for aufnahme of the pencil?
- wie geht man bei pcbs vor? erst pcbs entwerfen oder nimmt man irgendein mock?
 - einfach nen quader reinklatschen
- wie macht man das kabel fest?
 - siehe fahrrad baudenzug

design changes

- removing 9 to place v bearings on same level as vertical extrusion
- to do that we have to put v bearings at the end of the extrusion caps
- do not use motor end caps but use metal or 3d printed plastic caps. ask ferdi, maybe also acrylic

Motor Driver

- <https://www.digikey.com/en/products/detail/texas-instruments/DRV8428PWPR/13563046>
- actually, they are in the lab
- alternative: hbot firmware together with ramps
- we found those at the 3d printer in the hackspace:
<https://www.allegromicro.com/~media/Files/Datasheets/A4988-Datasheet.ashx>
- can you use different stepper drivers in one project?
- just use the full ramps kit.

Ferdi's Tipps:

- first, try stuff out with cartesian system, then switch to corexy
- Den Strom der Treiber stellt ihr so ein (deepseek):

Stepper Motor Driver Voltage (Vref) Adjustment Guide for RAMPS 1.4

⚠ Safety First

Power Off before adjusting.

Use a multimeter (DC 0-2V range).

Ensure proper heatsinks/fans are installed.

🔧 How to Adjust Vref

Locate the potentiometer on each driver.

Connect multimeter:

Black probe → RAMPS ground (e.g., power supply negative)

Red probe → Potentiometer

Adjust screw:

Clockwise → Increase voltage

Counter-clockwise → Decrease voltage

📏 Recommended Vref Values

Calculation Formulas

Driver Formula

A4988 $V_{ref} = I_{max} \times 0.7$

DRV8825 $V_{ref} = I_{max} \times 0.5$

TMC2xxx Configure via UART

Common Presets (Volts)

Motor Current A4988 DRV8825

1.0A 0.7V 0.5V

1.5A 1.05V 0.75V

2.0A 1.4V 1.0V

🔧 Troubleshooting

Symptom□Solution

Skipping steps□Slightly ↑ Vref

Overheating□↓ Vref + check cooling

Grinding noise□Check current + wiring

□□ Pro Tips

Start with lower values and test.

For TMC drivers, use software configuration.

Label adjusted drivers with a sharpie.

□□ Always verify motor specs in their datasheet!

- Die Motor-Kabel identifiziert Ihr, indem ihr erstmal die beiden paare identifiziert. Dazu könnt ihr zwei Kabel zusammendrehen (z.B. rot+grün) und schauen ob sich der Motor dann schwerer drehen lässt. Sobald Ihr ein paar gefunden habt, ist das andere das zweite Paar (schwarz gelb). Dann schraubt ihr zuerst Paar1 und dann Paar2 an die vier Klemmen z.B. rot, grün, schwarz, gelb. Wenn der Motor dann falsch rum dreht, invertiert Ihr ein Paar. grün, rot, schwarz, gelb. Ihr könnt aber auch einfach bis Sonntag warten wenn ich wieder da bin.
- even more tips: Hi! Some Tipps for the ramps board:
 - You plug the power to all 4 inputs on the Ramps board (either with 4 cables or 2 cables and two jumpers)
 - Use all three jumpers on the drivers to do 1/32 microstepping.
 - identify the pairs of the motors
 - connect the motors to "X, Y, Z"
 - connect x,y to the microswitches to the min-Endstopps z to max in Ramps
 - adjust the current (as explained)
 - install printrun/pronterface
 - home each axis seperatly and see if it is going in the right direction. If not you invert a motor by twisting the two wires.
 - ALWAYS TURN OFF THE POWER WHEN MESSING WITH THE MOTOR WIRES
 - The is a file in Marlin configuration.h where you can adjust the settings for direction, size, min, max-endstops...
 - Once you are able to home each axis, you can try to home all of them by clicking "Home" (and pressing X, Y, Z Endstop switches twice)
 - Then you can start to see if the axes are moving the right distance and adjust the "steps per millimeter"
 - Use the "rule of three" to find the right setting. You will have to repeat this a few times for each axis until it is right. Excel or OOCalc are your friends!
 - There is a page that might help a bit:
https://blog.prusa3d.com/calculator_3416/
 - You should do this with the normal XYZ firmware first. Once this works and you have the right steps per mm, you can switch to CoreXY.
 - Oh...you know that you can enter Gcode-commands in pronterface. M119 is very useful for debugging the endstops!
- jakob checked:

- notation: ij (pair i, cable j)
- find out pairs: measuring current. current is induced when turning if both are connected to one coil, current can be measured
- pairs are:
 - blue, red (1)
 - green, black (2)
- in the end we want to have order
 - 1A 1B 2A 2B
- stepper motors behave in the following way:
 - if you change the order one of the cable pairs 1A 1B 2B 2A, the direction of the motor changes
 - if you change the order of both of the pairs: 1B 1A 1B 1A, this is invariant to the direction of the motor direction
 - if you change both cable directions and you switch the pairs 2B 2A 1B 1A, this changes the direction of the motor
- procedure: align pairs next to each other, check if the motor turns right way. if not: change one of the pairs
- what is right direction? we will find out when running everything, but one can also change that in software
- setting all jumpers to do 1/16th microsteps: https://reprap.org/wiki/RAMPS_1.4#Drivers
- connect motor to x
- end stops. man will min und max end stops für z. für x und y reicht min.
- adjust current limit: <https://www.pololu.com/blog/484/video-setting-the-current-limit-on-pololu-stepper-motor-driver-carriers>
 - connect everything but the motor
 - find out max current, check datasheet
 - we use wantai stepper motor model number 42BYGHW811
 - max amperage: 2.5A
 - choose a driver with continuous current > motor driver
 - set current limit to max of motor limit
 - set even lower if
 - set lower if want system to run cooler, less power etc.
 - calculate vref according to product page
 - for DRV8834 its Current Limit = VREF x 2
 - our VREF is 1.25V
 - we have exactly that driver
 - measuring
 - connect to power supply
 - dont use the power supply current for current measurement since driver
- installing printrun
- installing marlin using platformio
- getting it to run. tstraight forward
- add servo
 - <https://3digitalcooks.com/2013-12-setup-servo-with-marlin/>
 - <https://marlinfw.org/docs/gcode/M280.html> and other gcodes
 - servo: possible

- corexy einstellen
 - <https://marlinfw.org/docs/configuration/configuration.html>
- waiting for machine to be finished
 - adjust params
 - x: 10cm -> tatsächlich 7.6
 - y: 10 -> 7.7
 - having cartesian system: one motor turns at a time when going in one direction. this is okay. we just have to measure how far a string travels:w
 - using M92 gcode
 - left motor: 100mm -> 78.4:
 - steps per mm is originally 80
 - $100/78.4 = */80$
 - new steps per mm = $80 * 100/78.4 = 102$
 - 100mm -> 99.4
 - new steps per mm: $102 * 100/99.4 = 102.62$
- now transferring to corexy
- setting corexy and try if we have to change something.
- works the same.
- switching axes: rotate one socket of the two
- switching direction: switch motors

homing

- am anfang schon homing buttons hinpacken. besser

steps calibrieren:

- eine umdrehung, messen wie weit. durch anzahl steps pro umdrehung teilen.
- iterativ machen. 3 bis 4 iterationen

Toolpaths

- gcode to motor action
- urumbu uses https://github.com/SeanP2001/Urumbu_USB_Stepper_Motor
- if urumbu is used, use sampo stuff: <https://gitlab.com/fab-lab-oulu/sampo>

Revision #106

Created 10 April 2025 16:13:08 by Jakob

Updated 17 April 2025 00:12:20 by Jakob