

Plotclock



joo

[VIEW IN BROWSER](#)

updated 12. 4. 2022 | published 30. 3. 2022

Summary

A clock that plots time. Designed with makeability in a fablab/makerspace/hackerspace in mind. You need: Laser cutter...

[Hobby & Makers](#) > [Other Ideas](#)

Tags: [clock](#)

A clock that plots time. Designed with makeability in a fablab/makerspace/hackerspace in mind.

You need:

- Laser cutter or 3D printer
- 1 Arduino
- 3 servos
- 1 dry wipe pen
- M3 nuts, bolts, thread tap

Update to V1.01:

added calibration routine and documentation to arduino code

see the instruction tab above for building instructions

<http://wiki.fablab-nuernberg.de/w/Ding:Plotclock> (german) <http://www.youtube.com/watch?v=iOLFP90DneY> <https://github.com/9a/plotclock> (old, not maintained!)

I'm not associated with any kits you are able to buy online

Instructions

Update 08/15: You might want to take a look at <http://www.thingiverse.com/thing:931498/#instructions> for a more detailed documentation

Update 10/15: You should also check out <http://www.instructables.com/id/Plot-Clock-for-dummies/?ALLSTEPS> Thanks to Kjetil Egeland!

Materials I used:

- Arduino uno
- Tower Pro 9g servos (those cheap ones from ebay)
- WB SL DRYWIPE MARKER, bullet tip
- A small custom servo adapter, there are shields or servo-arduinios that also can be used. You can also just use jumper wires: 5V to red servo wire, GND to brown or black servo wire and the arduino servopins connected to the yellow or orange servo wire

Building instructions:

(for version 1.01!)

1. Lasercut plotclock_beta1.pdf from 3mm material or 3D print plotclock_SUP7.skp
 - the black areas in the pdf on the servo arms should be used to engrave some material away, so the servohorns settle in
 - use <http://www.thingiverse.com/thing:250204> for single STL files if you 3D print the clock (Thanks corben33!)
2. 3D print swee.stl or reuse some other cylindrical part for the sweeper
3. Tap 2.5mm holes with M3 thread
4. Assemble arms with M3 screws, gently adjust the counternuts so that the joints under the screw heads just have enough play for smooth movement (see joint picture).
5. load Arduino sketch plotclock_V1, see code comments for further information
 - <http://playground.arduino.cc/Code/time> library is needed
6. Add some sort of cloth with double sided tape to sweeper
7. Attach servos to digital pin 2, 3 and 4, according to arduino sketch, test them without anything attached by plugging in the arduino (the photos show it being connected to 1, 2 and 3, which isn't optimal

since 1 is also the TX pin, so during arduino startup the lifting servo jumps around)

8. Put everything together

- You may have to redrill the holes that are used for securing the servos, since some servo screws are a tight fit.

Acrylic is brittle and easily breaks when the mounting holes are too small. Only push the servo arms onto their hubs, don't secure them with screws! For initial calibration do not mount the resting clip for the wiper (danger of collision with the pen). Glue or screw the servo arms to the acrylic parts.

9. When attaching the servo arms, make sure they can turn enough to reach every point on the drawing surface by manually turning them
10. Power the arduino, it should start a calibration cycle where every second the position of both servos changes by rotating exactly 90 degrees (see last image in gallery)
11. Change the SERVOFAKTOR, SERVOLEFTNULL and SERVORIGHTNULL values until everything matches up (see code for more information)
12. Attach the pen and the sweeper holder only after testing everything else

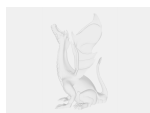
This description is still a bit of a WIP, so comment if you have any questions!

Category: DIY

Model files

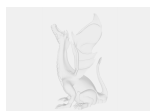


swee.stl



plotclock_sup7.skp

Other files



plotclock_beta1.pdf



plotclock_v1_01.ino



151018_plotclock.pdf



plotclock_v1.ino



plotclock_v1_03.ino

[Find source .stl files on Thingiverse.com](#)

License

This work is licensed under a
Creative Commons (4.0 International License)



Attribution

- ✗ | Sharing without ATTRIBUTION
- ✓ | Remix Culture allowed
- ✓ | Commercial Use
- ✓ | Free Cultural Works
- ✓ | Meets Open Definition