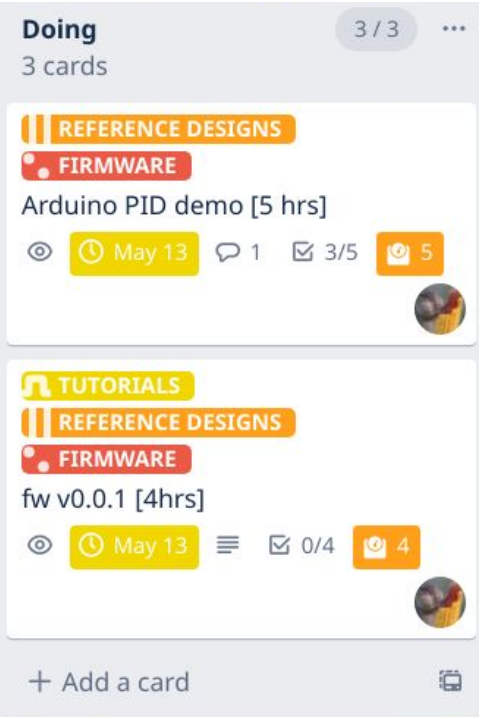
MAY SPRINT REPORT

GIT
ECAD
WIRING
ARDUINO

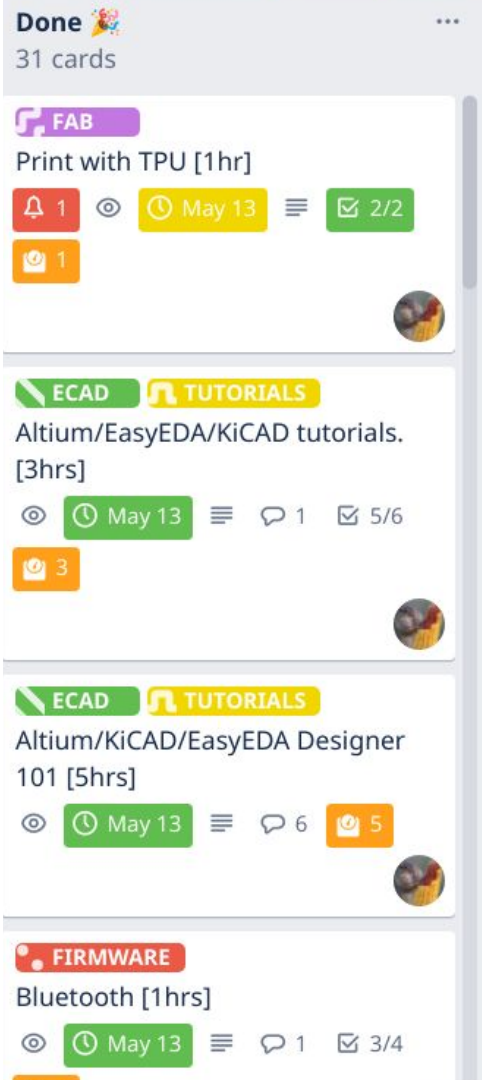
Matevž Zorec
13-05-2022

Robotics Engineering Project (LOTI.05.032)

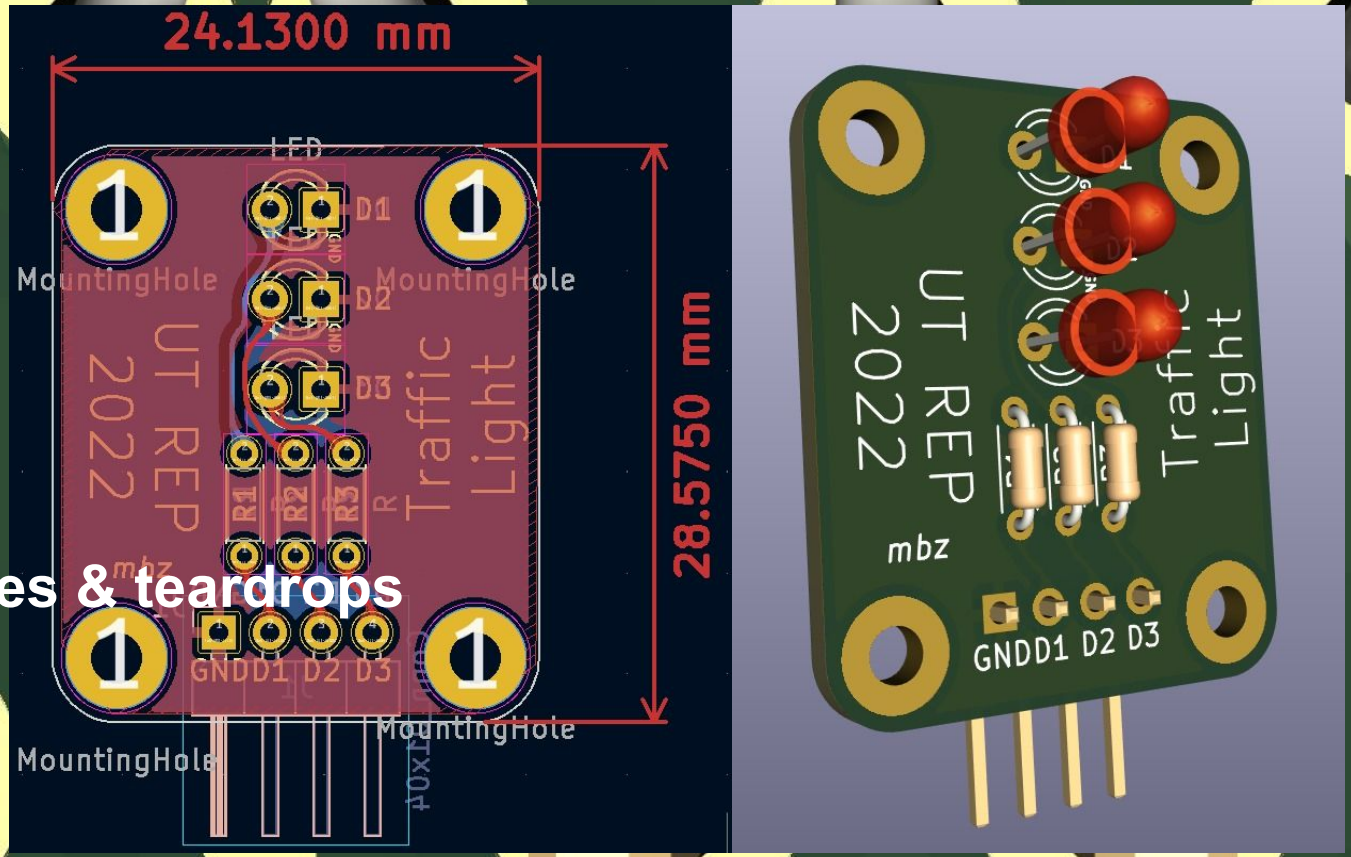


% progress on all of the cards of the sprint
~80%
time spent
<meetings, Arduino, ECAD 101, soldering, testing>
TOTAL ~26hrs out of planned 31hrs

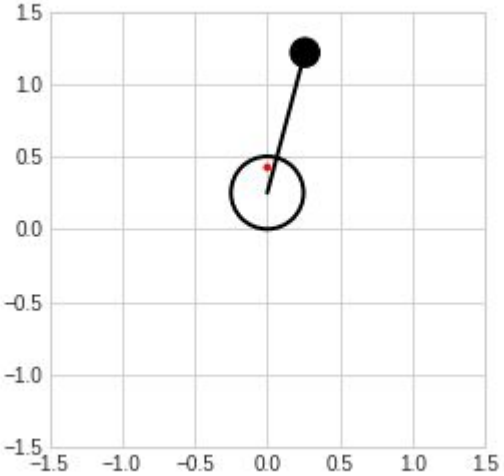
 # cards start of sprint
10
 # cards each column rn
design 7 | backlog 0 | doing 2 | done 30
 # cards matching/over/under time estm.
met: 4
over: 0
under: 4
2 of the best cards:
Altium/EasyEDA/KiCAD tutorials. [3hrs]
stepper re-wiring: wire modding [3hrs]
2 of the worst cards:
fw v0.0.1 [4hrs] @1hr *DOING
Arduino PID demo [5 hrs] @3.5hrs *DOING

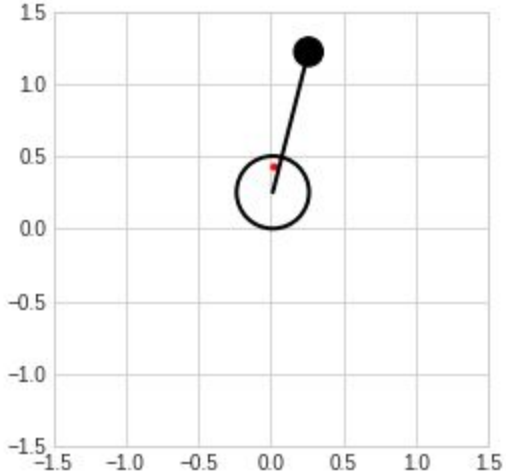


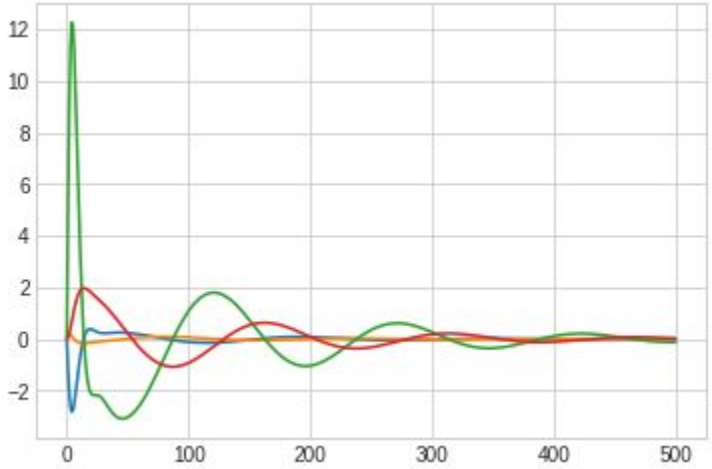
KiCAD
EasyEDA
Altium Designer
+ ECAD 101
+ 1 tutorial
+ gerbers
⇒ melted traces & teardrops



Self-balancing robot simulation







Public Member Functions

	AccelStepper (uint8_t interface=**AccelStepper::FULL4WIRE**, uint8_t pin1=2, uint8_t pin2=3, uint8_t pin3=4, uint8_t pin4=5, bool enable=true)
	AccelStepper (void(*forward)(), void(*backward)())
void	**moveTo** (long absolute)
void	**move** (long relative)
boolean	**run** ()
boolean	**runSpeed** ()
void	**setMaxSpeed** (float **speed**)
float	**maxSpeed** ()
void	**setAcceleration** (float acceleration)
void	**setSpeed** (float **speed**)
float	**speed** ()
long	**distanceToGo** ()
long	**targetPosition** ()
long	**currentPosition** ()
void	**setCurrentPosition** (long position)
void	**runToPosition** ()
boolean	**runSpeedToPosition** ()
void	**runToNewPosition** (long position)
void	**stop** ()
virtual void	**disableOutputs** ()
virtual void	**enableOutputs** ()
void	**setMinPulseWidth** (unsigned int minWidth)
void	**setEnablePin** (uint8_t enablePin=0xff)
void	**setPinsInverted** (bool directionInvert=false, bool stepInvert=false, bool enableInvert=false)
void	**setPinsInverted** (bool pin1Invert, bool pin2Invert, bool pin3Invert, bool pin4Invert, bool enableInvert)
bool	**isRunning** ()

http://www.airspayce.com/mikem/arduino/AccelStepper/

```
#include <AccelStepper.h>

// Define some steppers and the pins the will use
AccelStepper stepper1(AccelStepper::FULL4WIRE, 11, 10, 9, 8);
AccelStepper stepper2(AccelStepper::FULL4WIRE, 4, 5, 6, 7);
```

https://forum.arduino.cc/t/stepper-motor-basics/275223

Now until finish…?

Plans:

a) **finish PID demo ⇒ self-balancing**
b) **finish fw0.0.1 ⇒ self-balancing, reliably**
c) all in one PCB:
    i) stepper driver boards
    ii) ESP32 socket OR Arduino & ESP32
    iii) Powerpath
    iv) step down
    v) IMU [I2C]
    vi) ToF [I2C]
    vii) motor connectors

schematic to gerber to Fab

d) modify truss, accommodate:
    i) PCB
    ii) charging port
    iii) reset button
    iv) power switch
e) remote control w/ ESP32 ⇒ **fw0.2.0**

Ideas:

a) Leap Motion ⇒ control
b) servo landing leg
c) auto start feature

https://blog.keithkim.com/2020/07/note-leap-motion-on-ubuntu-2004.html