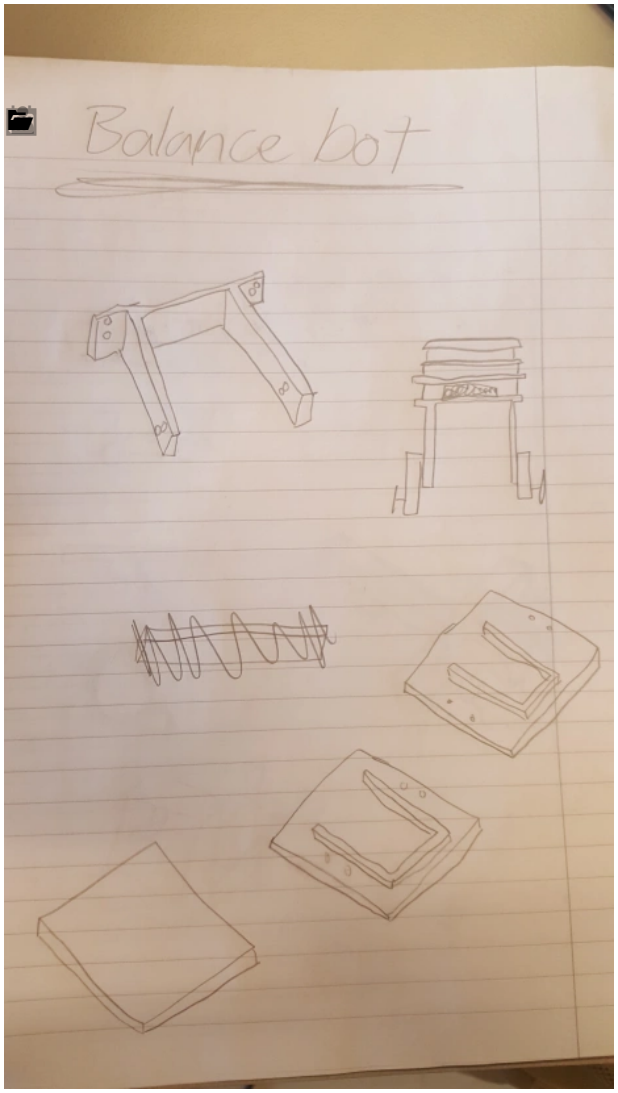
This site has no title!

BalanceBot with arduino

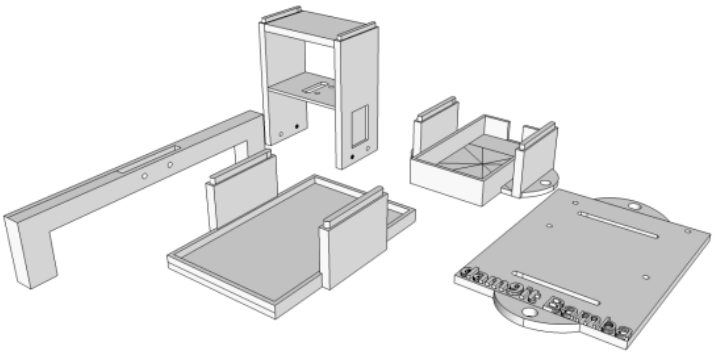
By Leif Salminen & Jaakko Rantala

Made during a prototyping course: http://terokarvinen.com/2017/prototyypin-rakentaminen-bus4tn007-8-w22 (http://terokarvinen.com/2017/prototyypin-rakentaminen-bus4tn007-8-w22)

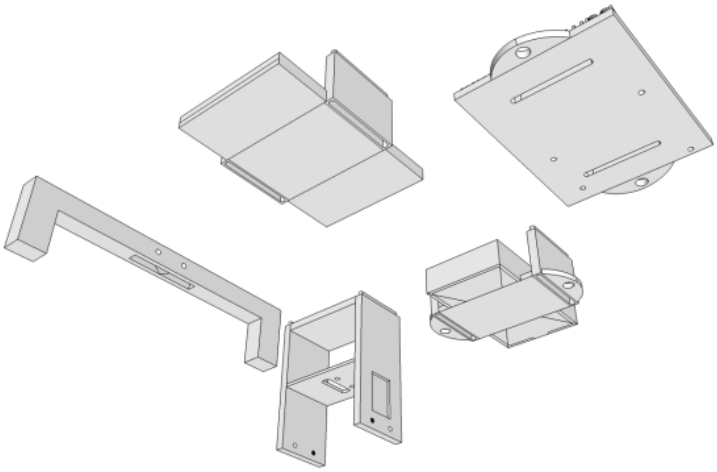
We enrolled on a week long prototyping course and decided try to make a two wheeled self balancing robot. We came up with idea late in the Monday evening so we only had a few days to do it. We started Monday by quickly drawing on paper the part that we needed to 3d print (drawing below).



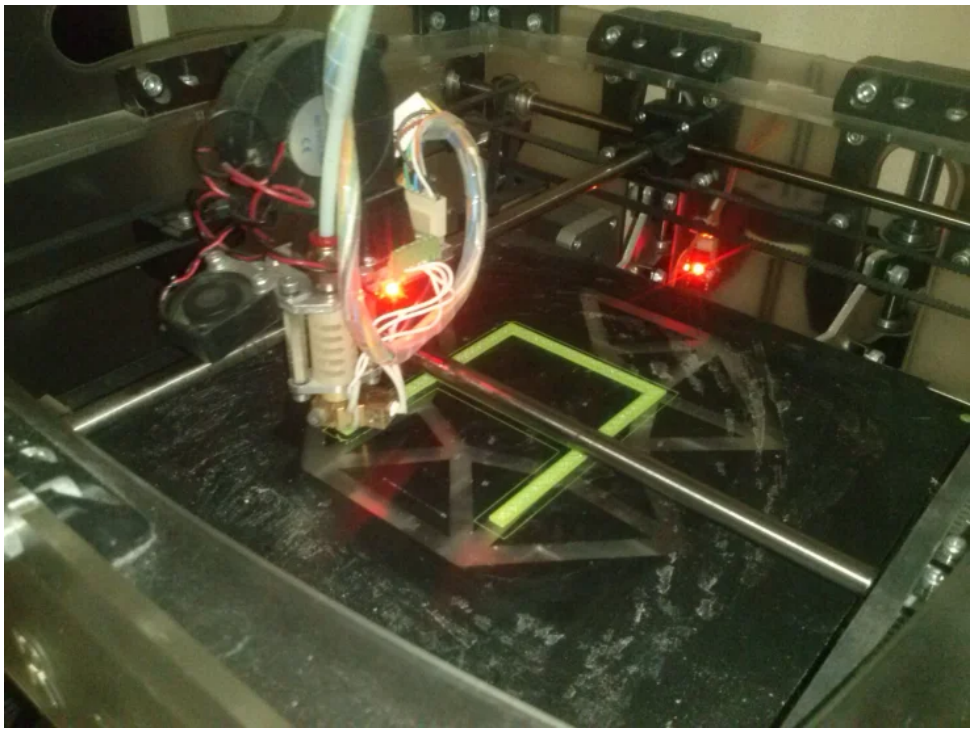
(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/20170601_141509/)



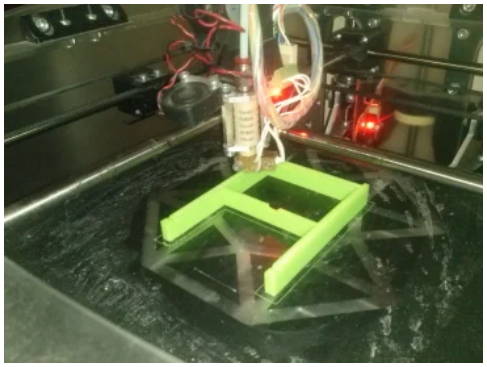
(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/balancebottop/)

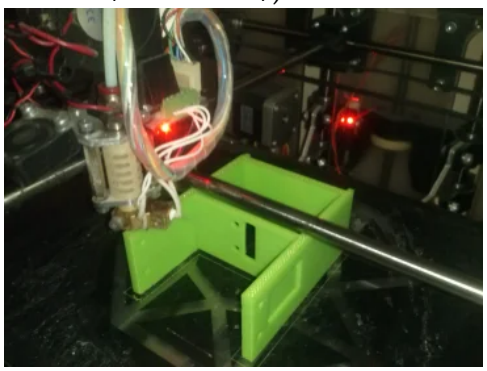


(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/balancebotbottom/)
After that I got on Sketchup to make 3d models of the drawings. During the modeling the design changed a little bit. I printed the bottom part at home the same night.

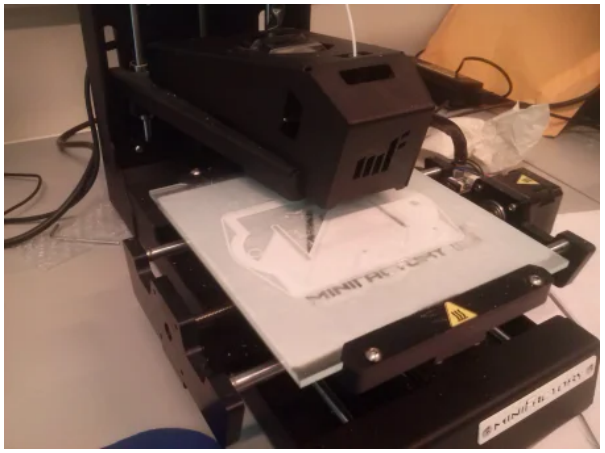


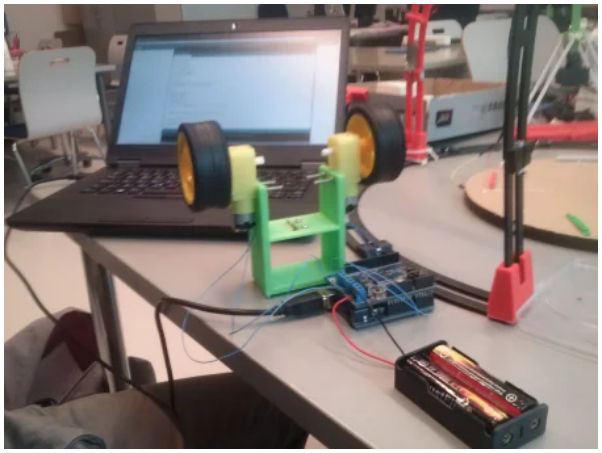
(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/cam00989/)


(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/cam00991/)


(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/cam00994/)

Tuesday we began our day by printing the rest of the parts at school. After printing we sanded the parts so that they would fit together nicely and then assembled the machine.


(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/cam00997/)

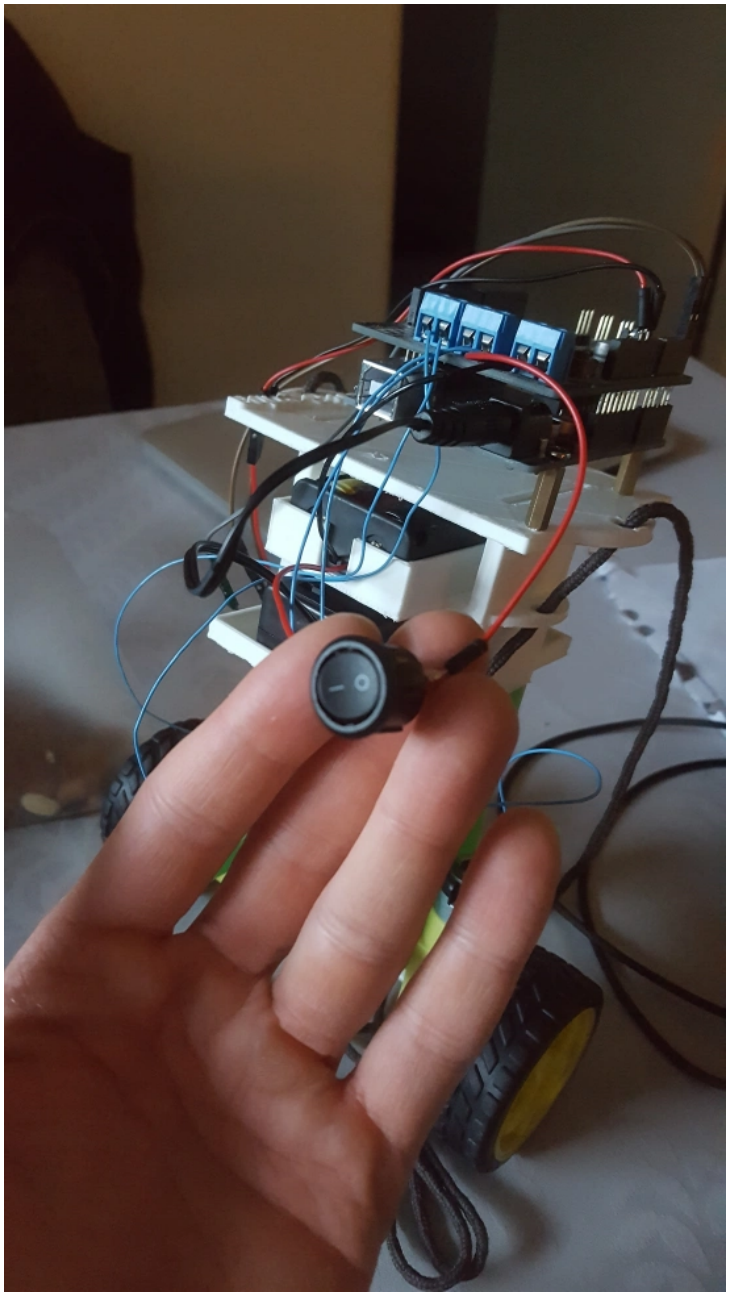

(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/cam00999/)

(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/20170530_134901/)

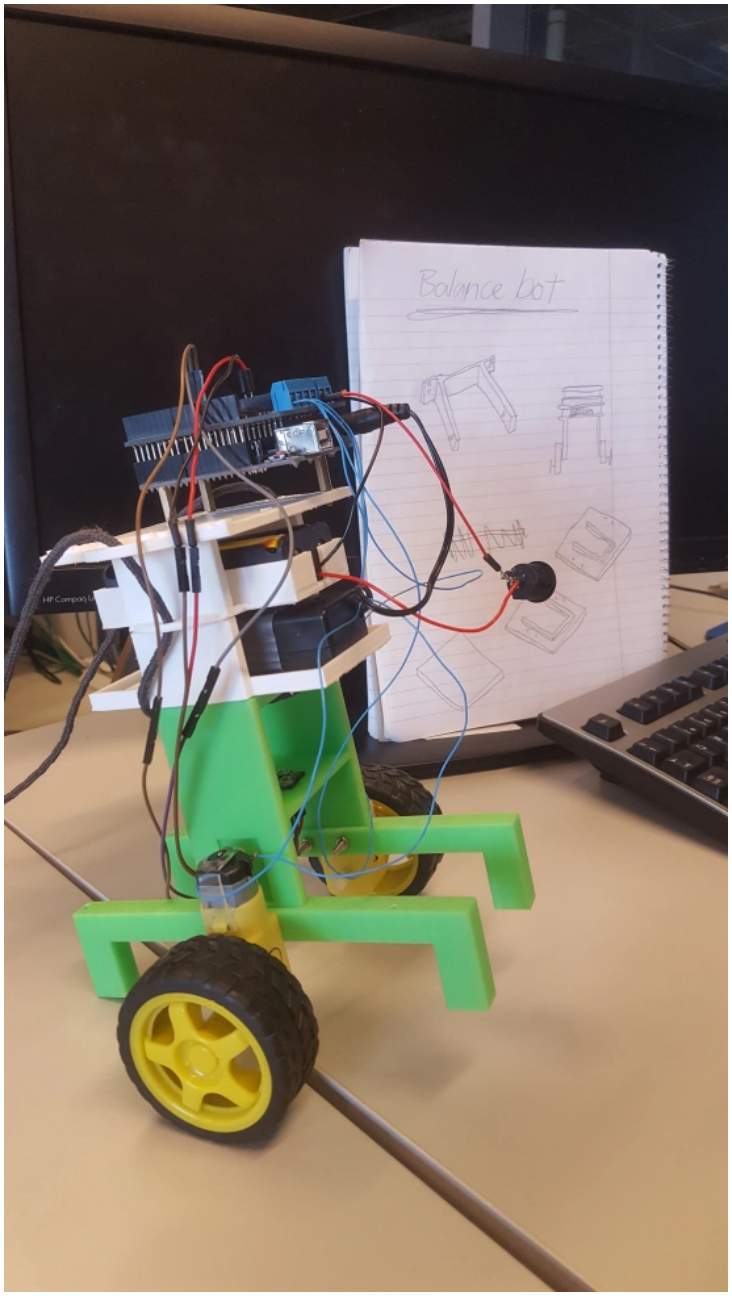
After assembly we started to look what kind of values does the MPU6050 give us. It was way too sensitive to any vibrations and we were practically making a shaking machine. We started to develop a filter to clean the MPU6050 data. We were not successful at this. Then Jaakko found a Kalman filter that was being used in another balancing bot (http://blog.tkjelectronics.dk/2012/09/a-practical-approach-to-kalman-filter-and-how-to-implement-it/ (http://blog.tkjelectronics.dk/2012/09/a-practical-approach-to-kalman-filter-and-how-to-implement-it/)). We downloaded the filter from the original makers Github (https://github.com/TKJElectronics/KalmanFilter (https://github.com/TKJElectronics/KalmanFilter)). This was a great find and helped a lot. We run the KalmanFilter on our arduino and studied the serial monitor until we found what was the variable that we wanted. From the code we looked where the serial print commands were and after them we added a call to our motor-controlling function and passed the needed variable to it. We started to test driving the motor according to the kalAngleY value that the filter gave us. Later that day we got the device to stand up, but it very soon wonder of and fell. We also needed to model and print bumpers so that the device wouldn't destroy itself.

BalanceBot early test

Wednesday we started by installing the bumpers. After the installation we started to fine tune the device and try to get it to stand as still as possible. No real progress was made that day. Later in day Jaakko added a kill switch to the motor power supply. That made testing a lot easier.

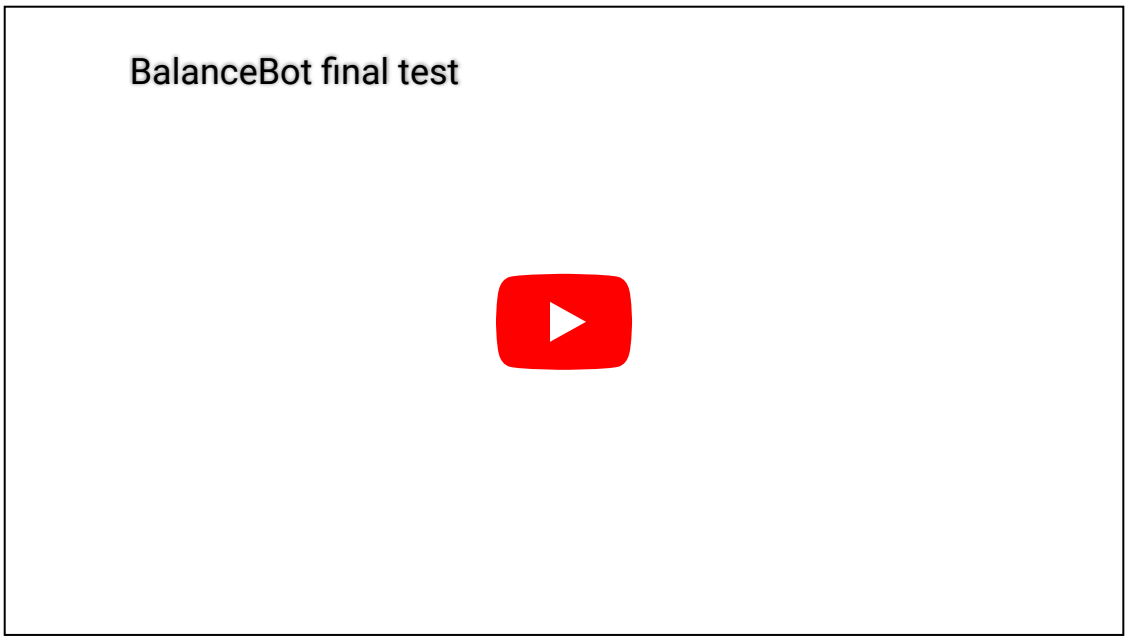


(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/20170531_220315/)



(https://dam9it.wordpress.com/2017/06/01/balancebot-with-arduino/20170601_161656/)

Thursday we continued with the tuning and testing. We were finally successful!



We originally thought we would need PID-controller, but at the end we didn't. Adding a PID-controller in the future is a good idea and we will probably look into it. What was really interesting in this project was that what we thought would be easy was hard and what we thought would be hard was easy.

Below is the code we integrated with the KalmanFilter. All our variables are floats.

```
void motorControl(float currentAng){
value = 0;
if(currentAng > wanted){
MotorShield.setMotorDir(1, 0);
MotorShield.setMotorDir(2, 0);
value = value + 1;
}
if(currentAng < wanted){
MotorShield.setMotorDir(1, 1);
MotorShield.setMotorDir(2, 1);
}
if(currentAng > wanted){
currentAng = currentAng + 1;
currentAng = currentAng * 1.50688905;
}
if(currentAng < 0){
currentAng = currentAng * -1;
}

value = value + value * 12;
if(value > 99){
value = 99;
}
if(currentAng + wanted > 1){
if(value < 65){
value = 65;
}
MotorShield.setMotorSpeed(1, value);
MotorShield.setMotorSpeed(2, value);
}else{
value = 0;
MotorShield.setMotorSpeed(1, value);
MotorShield.setMotorSpeed(2, value);
}}
```


The wanted variable is what the sensor gives when the bot is in the position where it should stand on its own. We gave it the value of -1.548. We also think that the wanted variable is the one that you would want to change if you want it to drive forwards and backwards.

In the future it would be nice to test how much movement changes if we make the bot taller. Also we would like to try a PID-controller and better motors.

We don't know why it works, but it does!

Parts used:

- Arduino uno (Funduino)

- Microbot dual motor controller (https://www.microbot.it/product/60/Dual-DC-Motor-Shield.html (https://www.microbot.it/product/60/Dual-DC-Motor-Shield.html))
- 2 X generic dc motors with wheels (the ones in a yellow casing)
- 2 X 18650 Batteries with a holder
- 9 volt battery with holder
- 1 X shoelace
- 1 X switch
- MPU6050(https://www.dfrobot.com/product-880.html (https://www.dfrobot.com/product-880.html))

Link Thingiverse where you can download the 3d files we made: https://www.thingiverse.com/thing:2359025 (https://www.thingiverse.com/thing:2359025)

Link for Github where we have the whole code: https://github.com/TheSteezer/Balancebot (https://github.com/TheSteezer/Balancebot)

Uncategorized (https://dam9it.wordpress.com/category/uncategorized/)
dam9it
June 1, 2017June 2, 2017