Termostaatti (joka ohjaa tuuletinta)

Aku-Oskari Vilkki

Tieto- ja tietoliikennetekniikan PT, ICT-asentaja

Careeria

Sisällysluettelo

[1. Laitteen toiminta 1](#_Toc30666437)

[2. Laitteen elektroniikka 2](#_Toc30666438)

[3. Ohjelman sisältö ja rakenne 3](#_Toc30666439)

[3.1 Keskiarvoistus 3](#_Toc30666440)

[3.2 Suoran yhtälö 4](#_Toc30666441)

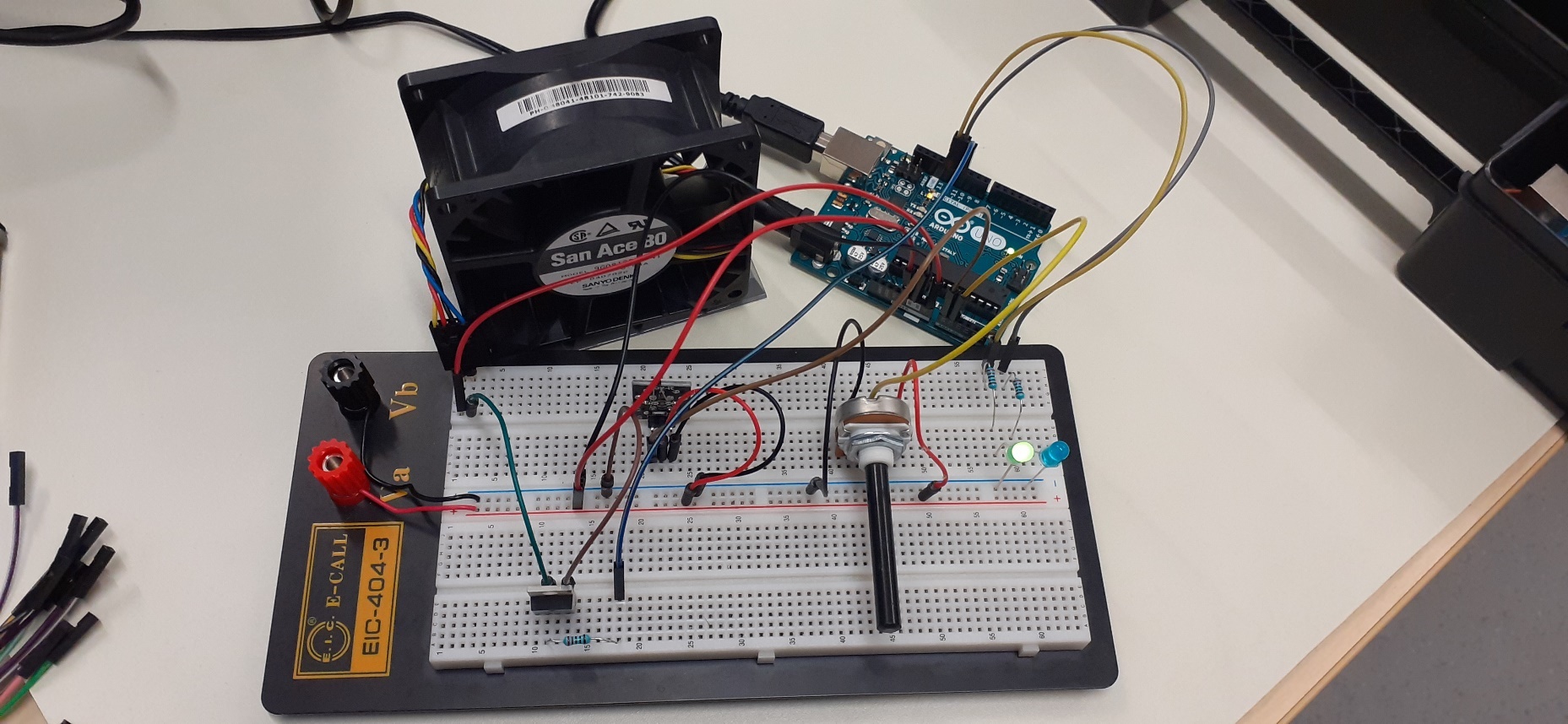
[3.3 PWM-ohjaus 5](#_Toc30666442)

[4. Oma tavoitteeni 6](#_Toc30666443)

[5. Projektikansio 7](#_Toc30666444)

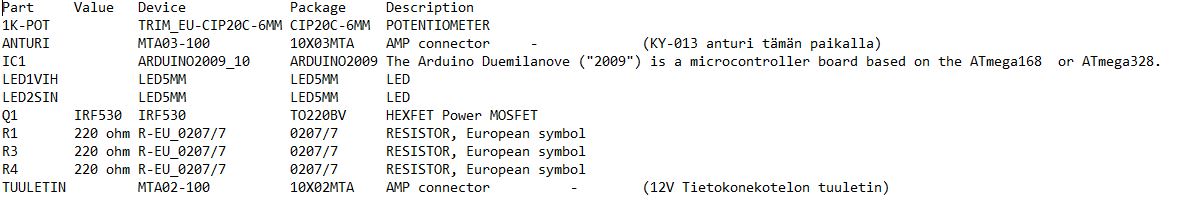
# 1. Laitteen toiminta

Laite reagoi lämpötilan nousuun käynnistämällä tuulettimen. Mitä suuremman lämpötilan nousun laite havaitsee, tai mitä suuremmaksi valitaan lämpötilan erotus potentiometrillä, sitä suuremmalla nopeudella puhallin pyörii. Laitteessa on vihreä ja sininen merkkivalo; aina kun lämpötila on alle halutun tason, palaa vihreä ledi. Jäähdyttäessä sininen ledi palaa.



# 2. Laitteen elektroniikka

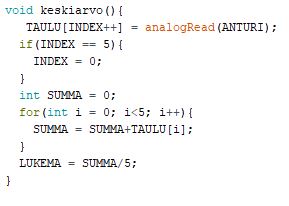
Lämpötila-anturina on käytössä KY-013. Anturissa on kolme pinniä A0, plus ja miinus. Tuuletin tarvitsi toimiakseen 12V jännitettä, jota varten lisättiin erillinen virtalähde. Tuulettimen toimiaksi lisättiin MOSFET-piiri IRF530N. Samaa MOSFET-piiriä hyödynnettiin myös tuulettimen PWM-ohjausta varten. Lämpötila-anturin peruskäyttöönottoon käyttämäni ohje: https://arduinomodules.info/ky-013-analog-temperature-sensor-module/. Piirilevy suunniteltiin EAGLE-ohjelmalla.



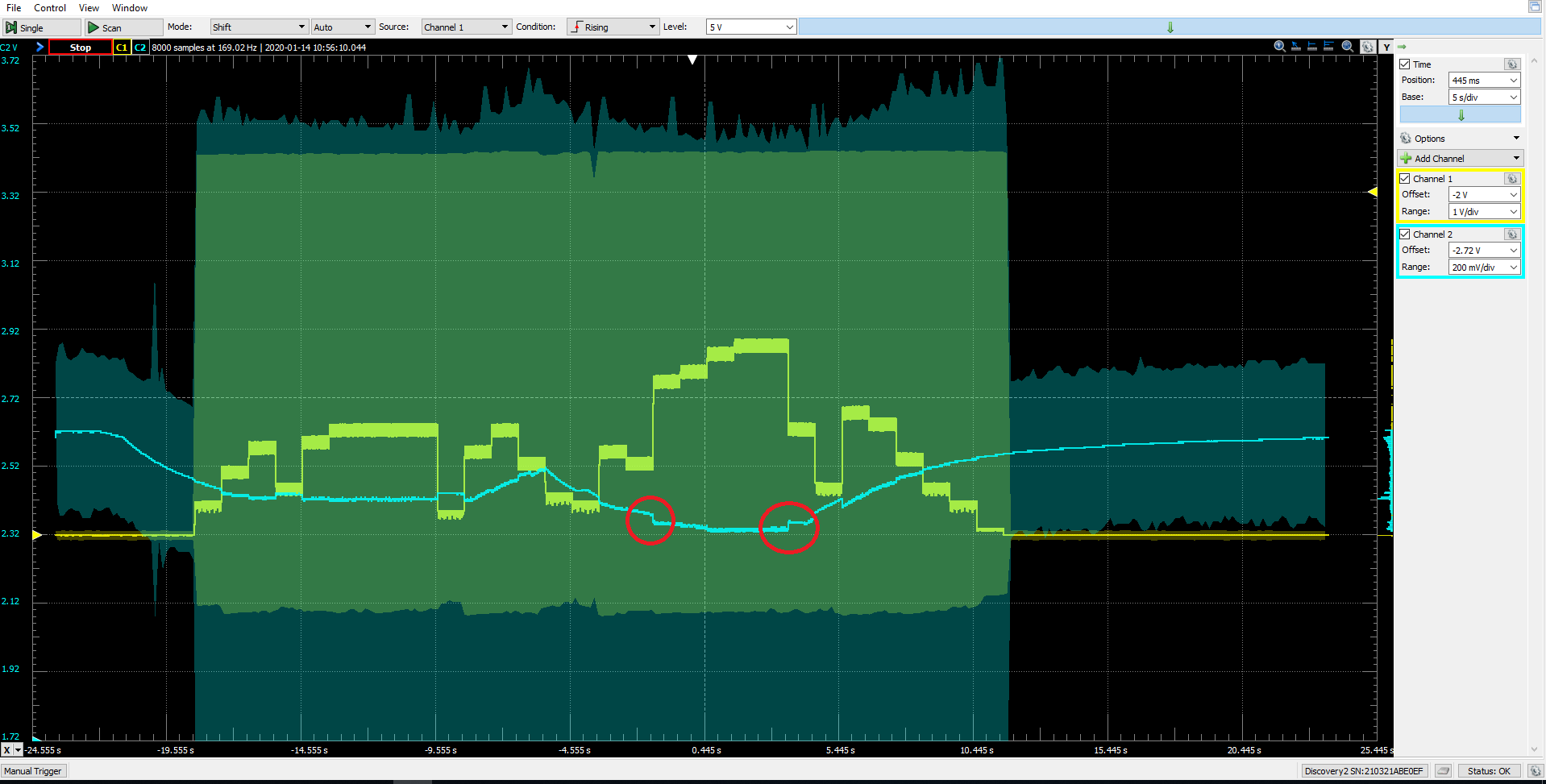
# 3. Ohjelman sisältö ja rakenne

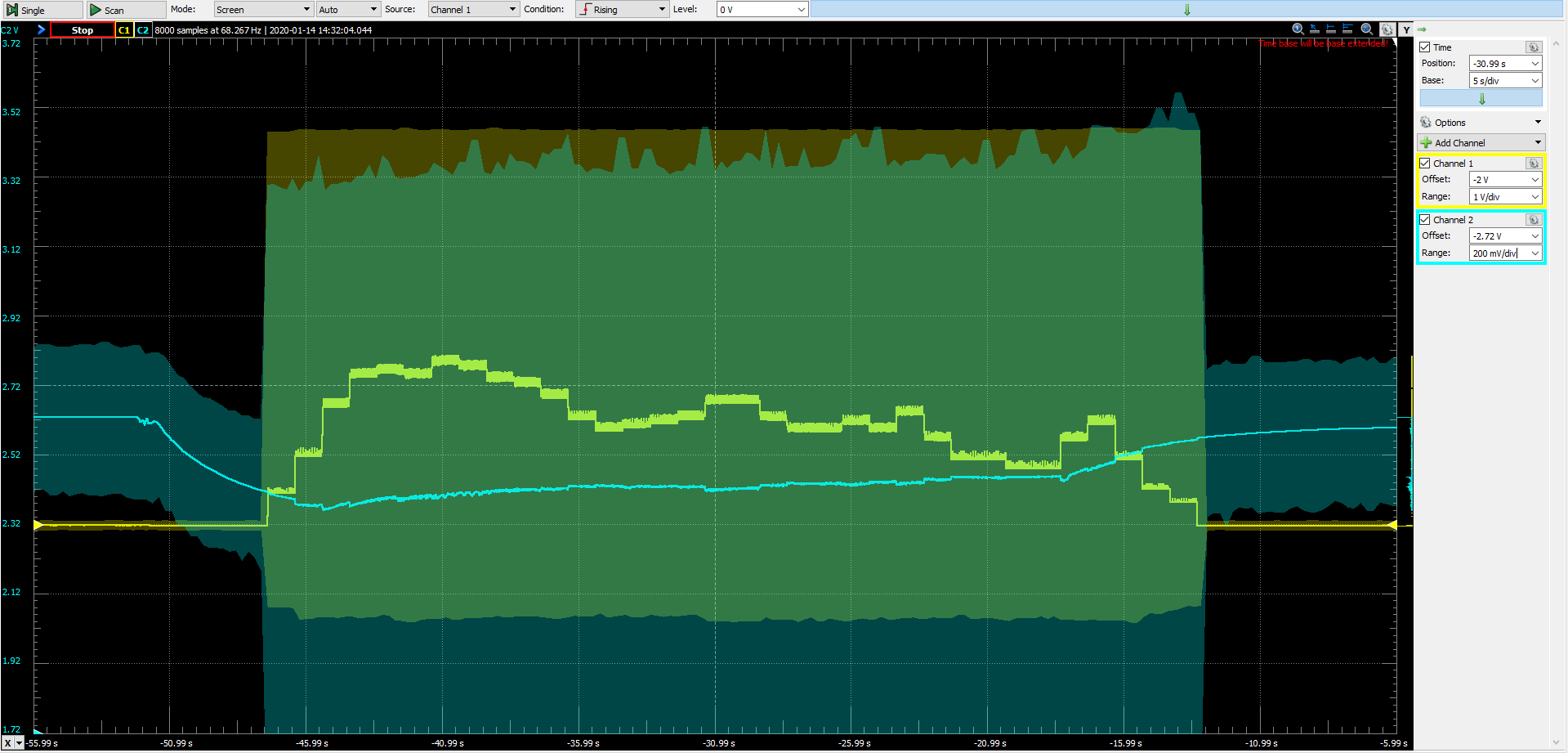
## 3.1 Keskiarvoistus

Ongelmaksi ohjelmassa muodostui anturin mittausten lukemien suuri vaihtelu, joka aiheutti laitteen epävakaata toimintaa. Toiminnan parantamiseksi lisättiin seuraava keskiarvoistus:



Ensimmäisessä kuvassa näkyy ympyröitynä lämpötilan poukkoilua, joka näkyy myös tuulettimen pwm-ohjauksen vastaavana poukkoiluna. Toisessa kuvassa lisätty keskiarvoistus, jossa laitteen toiminta on selvästi rauhoittunut.

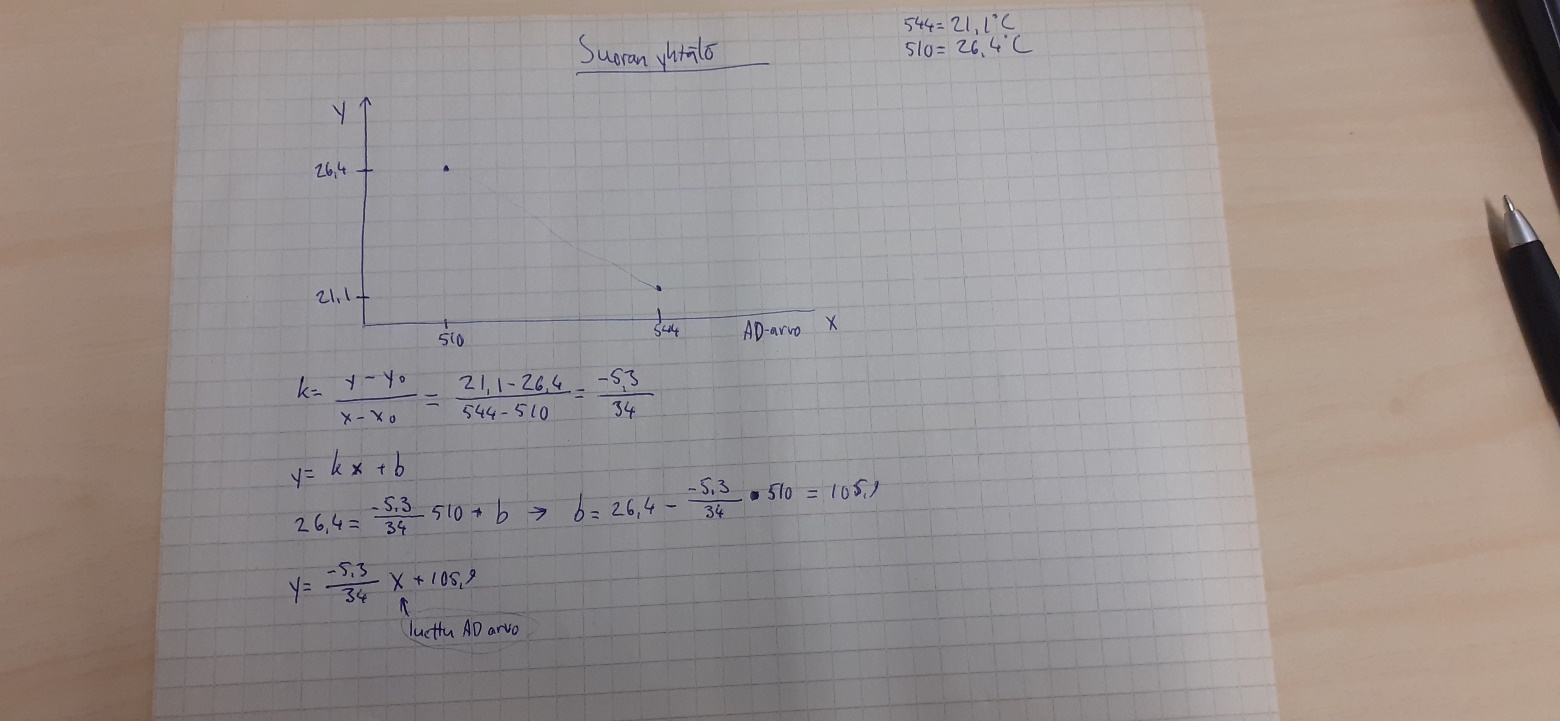




## 3.2 Suoran yhtälö

ASTE = (-5.30/34.00)\*LUKEMA+105.9;

Luettujen analogiarvojen muuttaminen celsius asteikkoon tehtiin hyödyntäen suoran yhtälöä. Mitattaessa huonelämpötila 21,1 celsius astetta vastasi AD arvoa 544, ja kädenlämpötila 26,4 celsius astetta vastasi AD arvoa 510. Näitä lukuja ja kuvassa näkyviä kaavoja hyödyntäen voitiin ratkaista yhtälö, joka lisättiin ohjelmaan. LUKEMA on keskiarvoistuksella saatu luku.

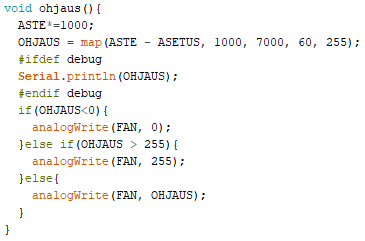


ASTE = map(LUKEMA, 510.0, 544.0, 26.4 , 21.1);

Saman ”muunnoksen” voi myös tehdä näin mäppäämällä, kunhan tietää AD arvoja vastaavat celsius asteet. Suoran yhtälöllä tarkkuus oli kuitenkin parempi, joten käytin mieluummin sitä.

## 3.3 PWM-ohjaus

Puhaltimen ohjauksesta olisi voinut tehdä myös paljon yksinkertaisemman, jossa puhallin olisi alkanut puhaltaa täydellä teholla heti kun lämpötila olisi noussut yli raja-arvon. Jotta laitteen toiminta olisi ollut monipuolisempaa ohjelmoinnin kannalta, hyödynnettiin arduino unon PWM-ohjausta. Näin puhaltimen ohjaus saatiin porrastetuksi. Anturin lukeman arvon ja potentiometrin asetusarvon erotus vaikuttaa siihen, miten nopeasti puhallin pyörii; mitä suurempi erotus sitä kovempaa puhallin pyörii. PWM-ohjaus toimii pulssimaiseen tapaan, jolloin 100%:lla (255 AD-arvo) jännitettä syötetään jatkuvana syöttönä. 50%:lla (127 AD-arvo) jännite vaihtelee symmetrisesti pulssina täyden jännitteen ja 0V:n välillä.



# 4. Oma tavoitteeni

Alkuperäinen ideani oli kehittää jonkunlainen termostaatti. Tein ennen tätä projektia vastaavanlaisen lämpötilahälyttimen, josta sain perusajatuksen tähän projektiin. Alunperin projekti alkoi siitä, että halusin kokeilla lämpötila-anturin toimintaa. Siitä projekti kehittyi, kun aloin lisäämään siihen osia vähän kerrallaan. Aiempi projekti oli melko helppo tähän verrattuna, joten halusin jatkokehittää sitä.

# 5. Projektikansio

<https://github.com/aov-0/termostaatti.git>