



VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Joonas Perälä

STEP-DOWN-HAKKURIN SUUNNITTELU
TUULIGENERAATTORIN LATAUSLAITTEE-
SEEN

Tekniikka
2022

TIIVISTELMÄ

Tekijä	Joonas Perälä
Opinnäytetyön nimi	Step-Down-hakkurin suunnittelu tuuligeneraattorin latauslaitteeseen
Vuosi	2022
Kieli	suomi
Sivumäärä	26
Ohjaaja	Jani Ahvonen

Tämän opinnäytetyön tarkoituksena on luoda Step-Down-hakkuri tuuligeneraattorin latauslaitteeseen. Piirilevy pystyy laskemaan sisään tulevan jännitteen suuruutta. Näin voimme antaa tehoa pienivirtaisille laitteille ilman, että ne vaurioituvat.

Piirilevy suunniteltiin valmiin piirikaavion pohjalta käyttäen PADS-ohjelmaa. Levy oli suunniteltu niin, että sen sai sovitettua Arduino Unon päälle.

Projekti aloitettiin ensin testaamalla piirin toimivuutta koekytkentälevyllä. Sen jälkeen ryhdyttiin suunnittelemaan piirilevyn rakennetta asettelemalla komponentit sopiville pakoille ja reititettiin piirikaavion perusteella. Suunnitteluvaiheen jälkeen valmistettiin piirilevy poraamalla reiät levyyn, komponentit aseteltiin paikoilleen ja lopuksi kolvattiin kiinni.

Lopuksi testattiin piirilevyn toimintaa ja tulokset olivat onnistuneet.

ABSTRACT

Author	Joonas Perälä
Title	Design of Step-Down Converter for Wind Generator charging device.
Year	2022
Language	Finnish
Pages	26
Name of Supervisor	Jani Ahvonen

The purpose of this thesis was to create a Step-Down converter for a wind generator charger. This circuit board can control the magnitude of the incoming voltage, enables providing power to smaller devices without damaging them.

The circuit board was designed based on the finished circuit diagram using the PADS program. The board was designed to fit on top of an Arduino Uno.

The project was first started by testing the functionality of the circuit on a breadboard. Then the design of the circuit board structure was started by arranging the components to suitable positions on the board using PADS program. After the design step, the circuit board was fabricated by drilling holes in the board, and the components were aligned and finally put into correct positions.

Finally, the operation of the circuit board was tested, and the results were successful.

SISÄLLYS

TIIVISTELMÄ

ABSTRACT

LYHENTEET

LIITTEET

1	JOHDANTO.....	6
2	STEP-DOWN-HAKKURI.....	7
	2.1 Step-Down-Hakkurin Perusteet.....	7
	2.2 Pulssinleveysmodulaatio ja Duty Cycle.....	7
	2.3 LC-piiri.....	8
3	PIIRILEVYN SUUNNITTELU.....	10
	3.1 PADS.....	10
	3.2 Arduino Uno.....	10
	3.3 Piirin Testaus KoekytKentälevyllä.....	12
	3.4 Piirikaavion Piirto.....	13
	3.5 PADS-layout.....	14
	3.6 Piirilevyn luonti.....	15
	3.7 NTC-Vastus.....	17
4	KOODI.....	18
5	PIIRILEVYN TESTAUS.....	20
	5.1 Manuaalisesti syötetty arvo.....	20
	5.2 NTC-Vastuksella Ohjattu Arvo.....	22
6	LOPPUPÄÄTELMÄT.....	23
	LÄHTEET.....	24
	LIITTEET.....	26

LYHENTEET

PWM = Pulssinleveysmodulaatio

LC-Piiri = Piiri, jossa on kela ja kondensaattori

PADS = Piirilevyn suunnitteluohjelma

OCR2B = Output Compare Register

Duty cycle = Aikaväli missä systeemi tai signaali on ylätilassa

NTC = Termistori, jolla on negatiivinen lämpötilakerroin

1 JOHDANTO

Tuulivoima on yksi uusiutuvista energian lähteistä. Voimme hyödyntää tuulienergiaa käyttämällä tuuliturbiineja, missä turbiinin lapoihin puhaltava tuuli pyörittää turbiinia. Tästä syntyy liike-energiaa, jonka voimme muuttaa sähköenergiaksi.

Projektissa suunniteltiin Step-Down-hakkuri tuuligeneraattorin latauslaitteelle. Step-Down-hakkuri tehtiin valmiin piirikaavion ja koodin pohjalta. Step-Down-hakkureita on käytössä monessa eri paikassa, missä käytetään pienijännitteisiä komponentteja. Esimerkiksi tietokoneen emolevyssä on laitteita kuten prosessori sekä näennäismuisti, jotka vaativat alle 5V jännitteitä. Silloin pystymme käyttämään Step-Down-hakkuria pienentämään jännitettä tarpeeksi, ettei komponentit hajoa. Hakkureilla saadaan myös parempi hyötysuhde kuin perinteisissä lineaarisissa regulaattoreissa, jotka pienentävät jännitettä lämmönhaihtumisella.

2 STEP-DOWN-HAKKURI

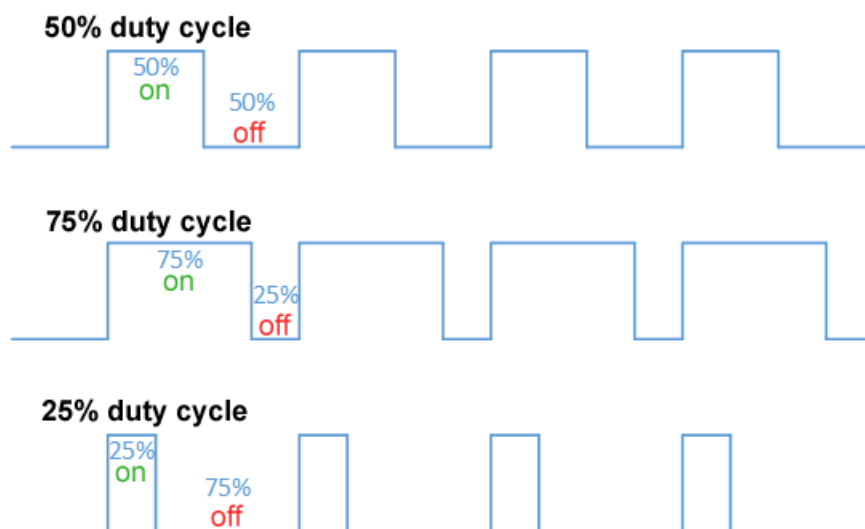
Tässä osiossa käydään läpi Step-Down-hakkurin toimintaperiaatteita.

2.1 Step-Down-Hakkurin perusteet

Step-Down-hakkurin avulla voidaan laskea tulojännitettä ja samalla säilyttää korkea hyötysuhde, että tehoa ei mene niin paljon hukkaan toisin kuin lineaarisissa tehojännitteissä.

2.2 Pulssinleveysmodulaatio ja Duty Cycle

Käytämme pulssinleveysmodulaatiota ohjaamaan transistoria, joka toimii kytkimen tavoin. PWM voimme nopeasti kytkeä transistorin päälle pois ja näin voimme ohjata kuinka paljon haluamme laskea jännitettä. Duty Cycle:llä ilmoitamme, että kuinka kauan PWM-signaali on ylätilassa prosentuaalisesti. **(Kuva 1.)**

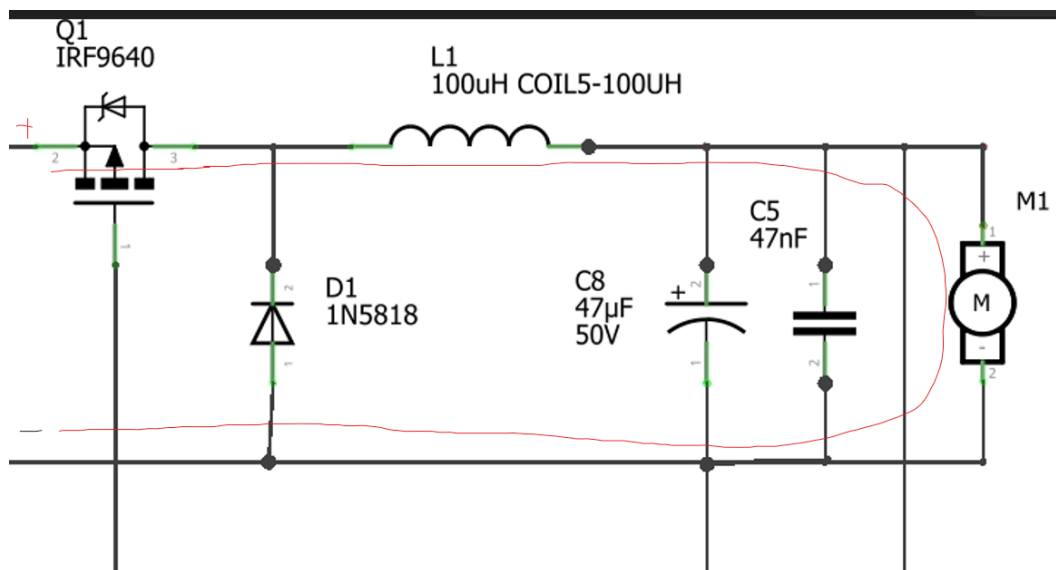


Kuva 1. PWM-signaali

2.3 LC-piiri

Tässä osiossa käydään läpi, miten käytämme PWM-signaalia ohjaamaan LC-piiriä.

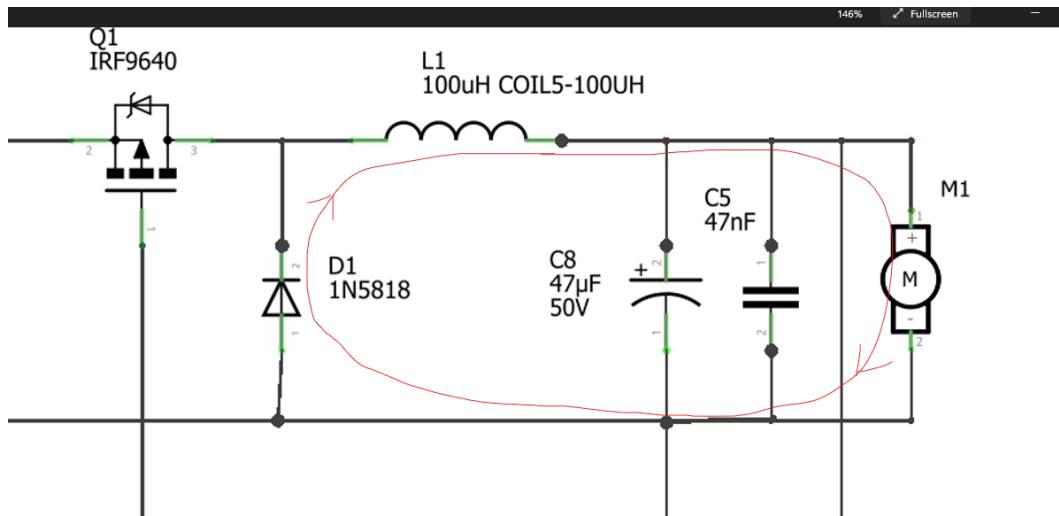
Kun kytkin on kiinni, niin virta kulkee kelan läpi, mihinkä varautuu jännitettä (**Kuva 2**). Kondensaattori auttaa poistamaan häiriötä piiristä tasaamalla jännitettä.



Kuva 2. Kytкин kiinni, jolloin Mosfet transistori johtaa

Kun kytkin on auki niin virta ei kulje enää jännitelähteen kautta ja kelaan ei varaudu enää jännitettä. (**Kuva 3.**) Kelan varautunut virta alkaa pienentyä hitaasti sen virran muutosta vastustavan ominaisuuden vuoksi, jolloin jännite säilyy vielä piirissä. Diodi antaa polun elektroneille, mistä ne voivat kulkea silloin, kun kytkin on auki.

Näin säätelemme jännitteen suuruutta kytkemällä transistorin kiinni ja auki nopeasti PWM-signaalilla.



Kuva 3. Kytkin auki, jolloin Mosfet transistori ei johda

3 PIIRILEVYN SUUNNITTELU

Tässä osiossa lähdetään suunnittelemaan piirilevyä. Suunnittelussa käytin apunani PADS Pikaopasta. (Ahvonen, j. 2022. PADS Pikaopas)

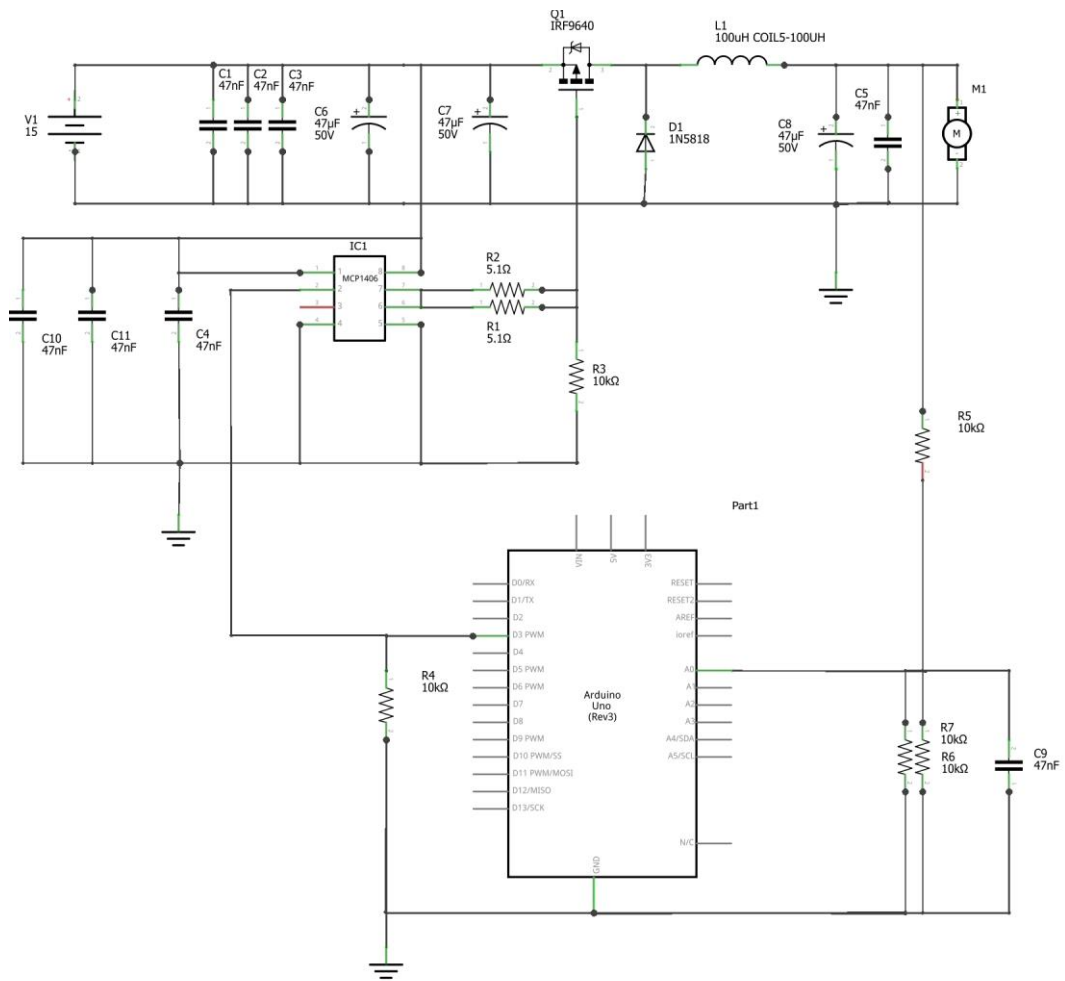
3.1 PADS

PADS on ohjelma, jolla voidaan suunnitella piirilevyjä ja jota tulemme käyttämään tässä projektissa. Komponentteja voidaan asettaa halutusti ja kytkeä toisiinsa käyttäen PADS-logic-ohjelmaa. Ohjelmalla pystyy myös tarvittaessa luoda omia komponenttejakin.

Kun piirikaavio on piirretty PADS-logicilla, voimme siirtää sen PADS-layouttiin. Layoutissa asettelemme komponentit ja juotokset niin, miten ne tulisivat fyysisesti itse piirilevylle.

3.2 Arduino Uno

Tässä projektissa käytettiin Arduino Unoa ohjaamaan piirilevyn toimintaa. Arduinolla pystymme helposti tuottamaan PWM-signaalia, jolla voimme säädellä piirilevyä. Pystymme myös tarkastelemaan eri arvoja Arduinolla helposti.



fritzing

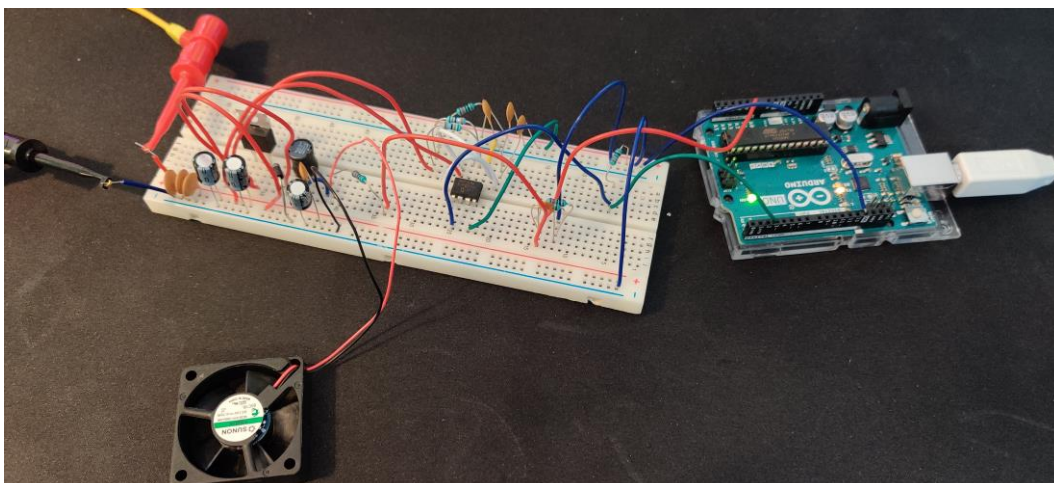
Kuva 4. Piirikaavio

3.3 Piirin testaus koekytkentälevyllä

Aluksi testattiin piirin toimivuutta koekytkentälevyllä.

Kytkeä on kiinnitetty 12V virtalähteeseen (**Kuva 5**). Arduino UNO sisältää koodin, missä Arduinon kolmas nastasta tulee PWM signaali, mikä menee MCP1406 mikropiiriin sisääntuloon ja Transistori on kytketty sen ulostuloon. MCP1406 pystyy lähettämään voimakkaamman virtapulssin transistoriin, joka tekee siitä johtavamman.

Piirilevy toimi koekytkentälevyllä ja tuloksissa kuormassa olevan laitteen virta oli pienempi, kuin syöttöjännite.

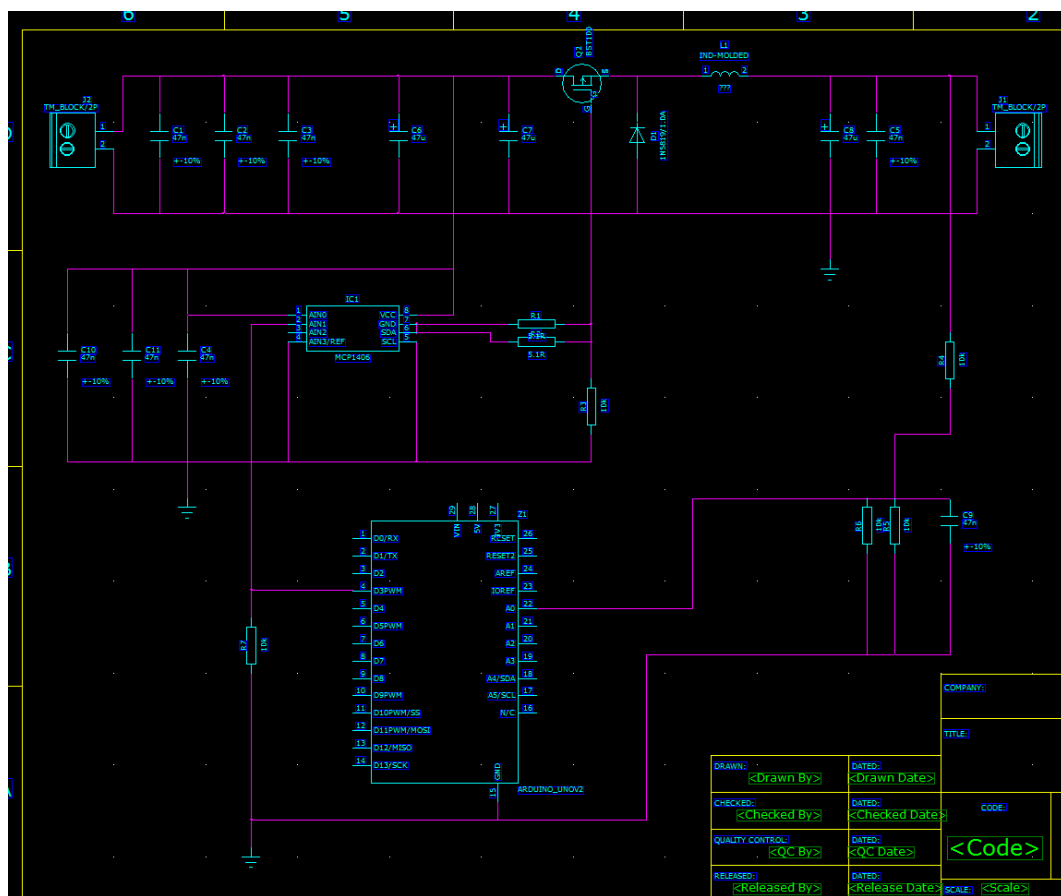


Kuva 5. Piirin testausta koekytkentälevyllä

3.4 Piirikaavion piirto

Ensin piirretään piirikaavio PADS-logic-ohjelmaan valmiiksi annetun piirikaavion pohjalta. **(Kuva 4.)** Kaikki komponentit löytyivät valmiista kirjastosta paitsi Arduino sekä MCP1406, jolle täytyi luoda omat komponentit ohjelmassa.

Piirikaavio on nyt piirretty valmiiksi ja voimme viedä piirikaavion PADS-layouttiin, missä voimme asettaa komponentit paikoilleen piirilevyllä ja reitittää kaikki juotokset piirilevyllä.

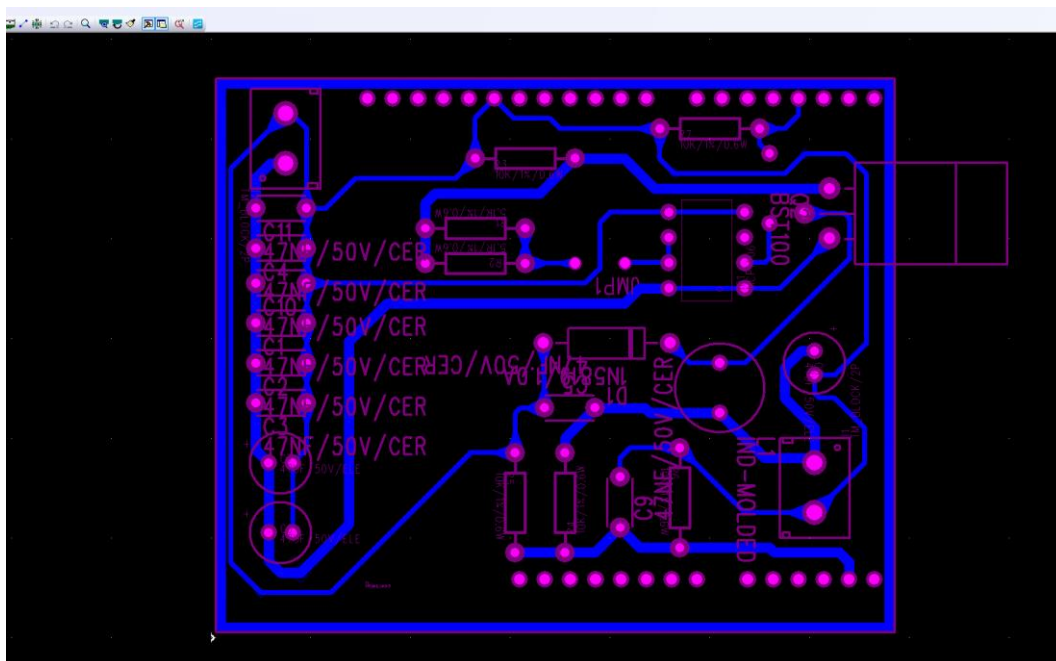


Kuva 6. Piirikaavio piirretty PADS-logicilla

3.5 PADS-layout

Seuraava vaihe on asetella komponentit sopiville paikoille ja reitittää ne oikein. Tähän käytämme PADS-layouttia. Tämä vaihe voi olla myös haastavin, koska sopivan paikan löytäminen komponenteille vaatii hieman pientä säätöä. Pitää myös huomioida reitityksen etäisyys, ettei se ole liian lähellä toista reititystä, mistä voi syntyä häiriöitä. Reitittäessä kannattaa myös välttää yli 45 asteen kulmia kääntyessä, koska siitä voi myös syntyä häiriöitä. Komponentit kannattaa myös asetella niin, että on paljon tilaa juotos vaiheessa sen helpottamiseksi. Ohjelman asetuksista pystyy säätämään reititysten leveyttä. Reititykset, mitkä menevät komponentin positiiviselta puolelta kannattaa asettaa leveämpi reititys. Niissä kulkee yleensä isompi signaali, niin ne on hyvä asettaa vähän leveämmiksi.

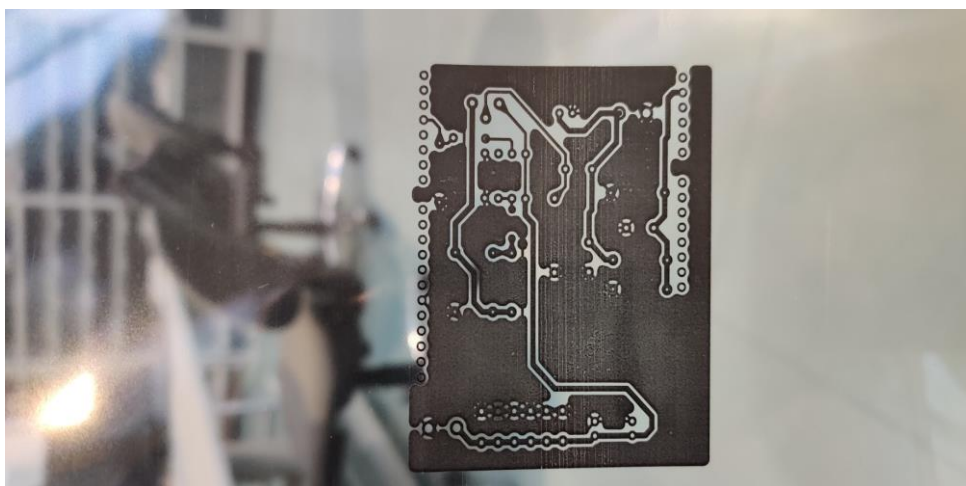
Seuraavaksi lisätään vielä kuparointi, jonka jälkeen voimme tulostaa piirin laminaatille seuraavaa vaihetta varten. **(Kuva 7.)**



Kuva 7. PADS-layoutissa suunniteltu piirilevy

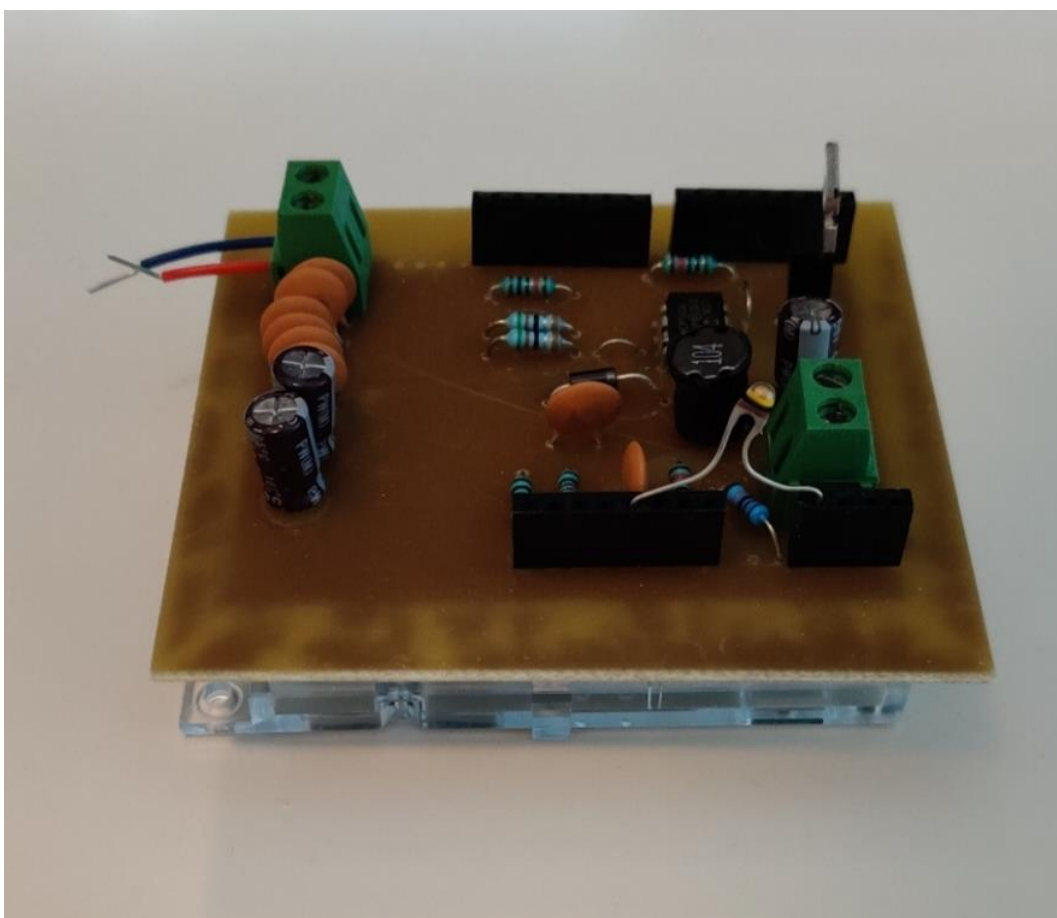
3.6 Piirilevyn luonti

Aluksi tulostamme PADS-layoutissa tehdyn suunnitelman laminaatille. **(Kuva 8.)** Tämän jälkeen asetamme laminaatin levyn päälle, jossa on kuparipinnoite. Sitten altistamme sen UV-säteilylle noin 35 sekunniksi.



Kuva 8. Laminaattituloste

Seuraavaksi uitamme levyä lakassa noin 20 sekuntia, kunnes kuparissa alkaa näky-
mään piirilevymme jälkiä. Sitten upotamme levyn happoon. Tämä poistaa sen ku-
parin, mikä ei altistunut UV-säteilylle. Pidämme levyä hapossa noin 15–20 minuut-
tia, kunnes ylimääräinen kupari on hävinnyt. Tämän jälkeen vielä pesemme vedellä
ja hiomme kevyesti hiekkapaperilla ylimääräisen lakan pois. Silloin kolvatessa juo-
tokset tarttuvat kupariin paremmin. Viimeiseksi porataan vielä levyyn merkattui-
hin kohtiin reiät komponenteille.

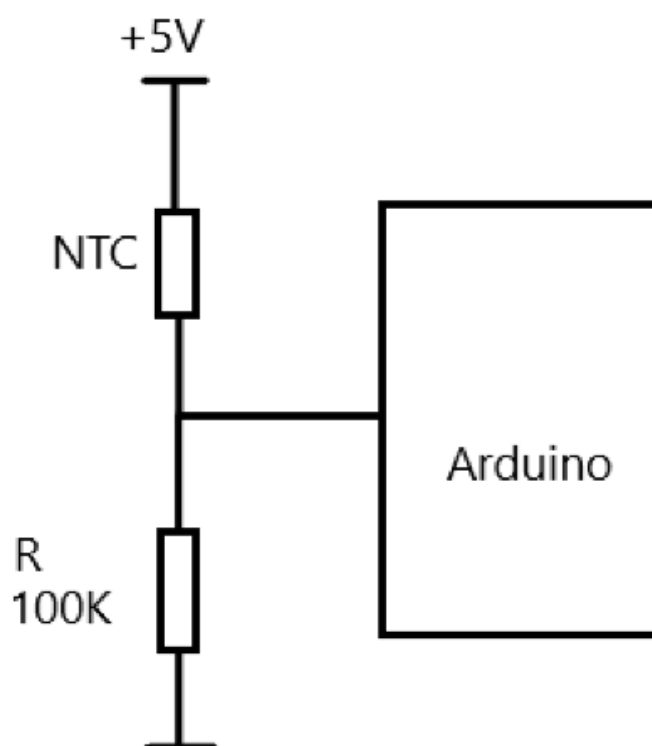


Kuva 9. Valmis Piirilevy

Tässä kuvassa on nyt valmis piirilevy. **(Kuva 9.)** Levy on painettu Arduinin päälle niin sanotuksi Arduinohatuksi.

3.7 NTC-vastus

Piirilevyyn on lisätty lisäksi vielä NTC vastus, jolla voidaan mitata lämpötilaa. Tässä käytämme NTC vastusta ohjaamaan tuulettimen nopeutta lämpötilan perusteella. NTC vastus on kytketty Arduinon +5V nastaan ja 100K ohmin vastukseen. **(Kuva 10.)**



Kuva 10. NTC-piirikaavio

4 KOODI

Tässä on Arduinossa käytetty valmis koodi.

```
//Alustetaan setpoint ja i arvo.
int setpoint=0, i=0;

void setup()
{
  /*
   * WGM21=1, WGM20=1 = Fast PWM
   * COM2B0=1, COM2B1=1 = Inverted mode. If COM2B0=0 then non-inverted mode
   */

  //Tämä asettaa rekisterissä TCCR2A bitit asentoon, että fast PWM tila tulee päälle pinni kolmosessa. Lisäksi invertoidaan signaali, koska käyttämämme transistori toimii
  TCCR2A = _BV(WGM21) | _BV(WGM20) | _BV(COM2B1);

  //Tämä asettaa rekisterissä TCCR2B bitit asentoon niin, että ei ole esijakaja päällä. Näin saamme mahdollisimman suuren taajuuden Arduinosta.
  TCCR2B = _BV(CS20); //16 MHz / 256 = 62500 Hz
  //Asetetaan PIN 3 ulostuloksi.
  pinMode(3, OUTPUT);
  //Asettaa datan nopeuden serial datan välitykselle.
  Serial.begin(115200);
}

//Silmukka missä on Logiikka PWM signaalin ohjaamiseen.
void loop()
{
  //Tässä kohtaa voidaan manuaalisesti säätää setpoint arvoa serialmonitorissa.
  if(Serial.available()>0)
  {
    setpoint=Serial.parseInt();
  }
  //Asetetaan A4 pin Lukemaan piirilevyn takaisinkytkennän jännitettä
  int ADCC=analogRead(A4);

  //Asetetaan A2 pin Lukemaan NTC vastuksen jännitettä
  int NTC=analogRead(A2);

  //tässä lasketaan NTC vastuksen lämpötilan kulmakerroin käyttäen lineaarifunktiota
  float temp=0.0925*NTC-21.925;

  //Tässä lasketaan tuulettimen pyörimisnopeus lämpötilan funktion perusteella ja asetetaan se setpoint muuttujaan.
  setpoint=512.0/30.0*temp+306.0;

  //tässä otetaan eroarvo setpoint arvon ja takaisinkytkentäjännitteen perusteella.
  int error=setpoint-ADCC;

  //Tässä osassa ohjataan PWM signaalin tilaa
  //jos eroarvo on suurempi kuin viisi
  if(error>5)
  {
    //kasvatetaan i arvoa ja asetetaan se OCR2B rekisteriin.
    if(i<255)
      i++;
    OCR2B = i;
  }
  //jos eroarvo on pienempi kuin -5
  else if(error<-5)
  {
    //Pienennetään i arvoa ja asetetaan se OCR2B rekisteriin
  }
}
```

```

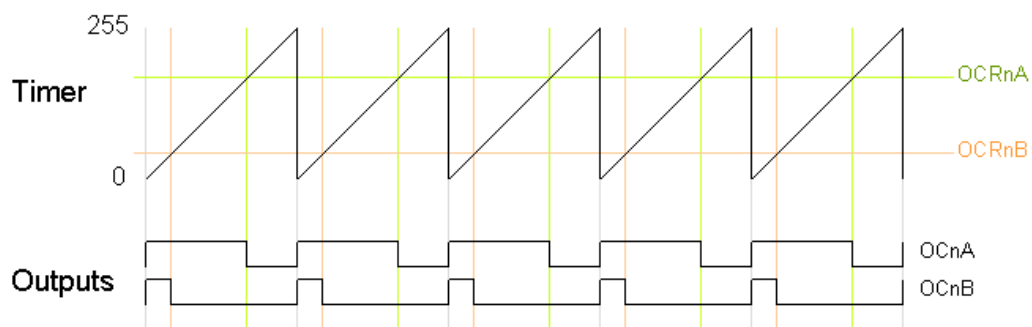
    if(i>0)
        i--;
        OCR2B = i;
    }
    else
        //Jos ei tarvitse tehdä mitään niin asetetaan sama arvo OCR2B rekisteriin.
        OCR2B=OCR2B;

    //Jos OCR2B arvo menee nolnaan saakka, asetetaan nasta 3 alatilaa.
    if(OCR2B==0)
    {
        digitalWrite(3, LOW);
    }
    else
    {
        //Tässä kohtaa asetetaan varmuudenvuoksi uudestaan rekisterin bitit.

        TCCR2A = _BV(WGM21) | _BV(WGM20) | _BV(COM2B1);
        TCCR2B = _BV(CS20); //16 MHz / 256 = 62500 Hz
    }
    //Tässä kohtaa tulostetaan serialmonitoriin lämpötilaa (Temperature), takaisinkytkennän
    jännite (Voltage) ja pulssin leveys prosenttiarvona (Duty cycle)
    Serial.print("Temperature: ");
    Serial.println(temp); //Temperature
    Serial.print("Voltage: ");
    Serial.println(12.0/818.0*ADCC); //Voltage
    Serial.print("Duty cycle: ");
    Serial.println(OCR2B/255.0*100.0); //Duty cycle
    Serial.println("");
}

```

OCR2B rekisterin avulla ohjaamme, kuinka kauan PWM-signaali on ylätilassa. Se on 8-bittinen rekisteri, johon voi syöttää arvoja 0–255 väliltä. Mitä suurempi tuo arvon on, sitä kauemmin signaali on ylätilassa. **(Kuva 11.)**



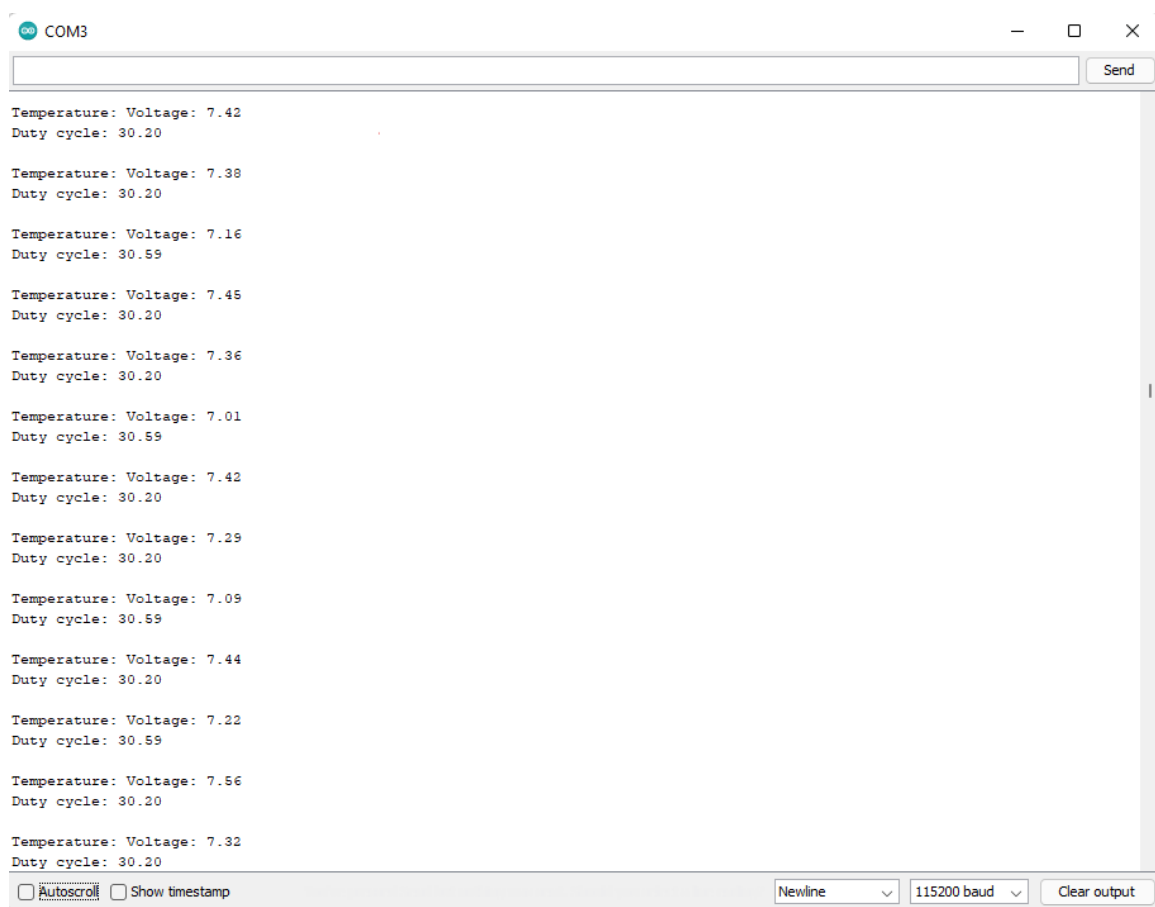
Kuva 11. OCR-Kuvaaja (Arduino.cc, 2022 b)

5 PIIRILEVYN TESTAUS

Tässä osiossa testattiin piirilevyn toimivuutta.

5.1 Manuaalisesti syötetty arvo

Asetin setpoint arvon ensin manuaalisesti ja tulosten perusteella piirilevy näyttäisi laskevan jännitettä onnistuneesti kuvassa. (Kuva 12.)



```
COM3
Temperature: Voltage: 7.42
Duty cycle: 30.20

Temperature: Voltage: 7.38
Duty cycle: 30.20

Temperature: Voltage: 7.16
Duty cycle: 30.59

Temperature: Voltage: 7.45
Duty cycle: 30.20

Temperature: Voltage: 7.36
Duty cycle: 30.20

Temperature: Voltage: 7.01
Duty cycle: 30.59

Temperature: Voltage: 7.42
Duty cycle: 30.20

Temperature: Voltage: 7.29
Duty cycle: 30.20

Temperature: Voltage: 7.09
Duty cycle: 30.59

Temperature: Voltage: 7.44
Duty cycle: 30.20

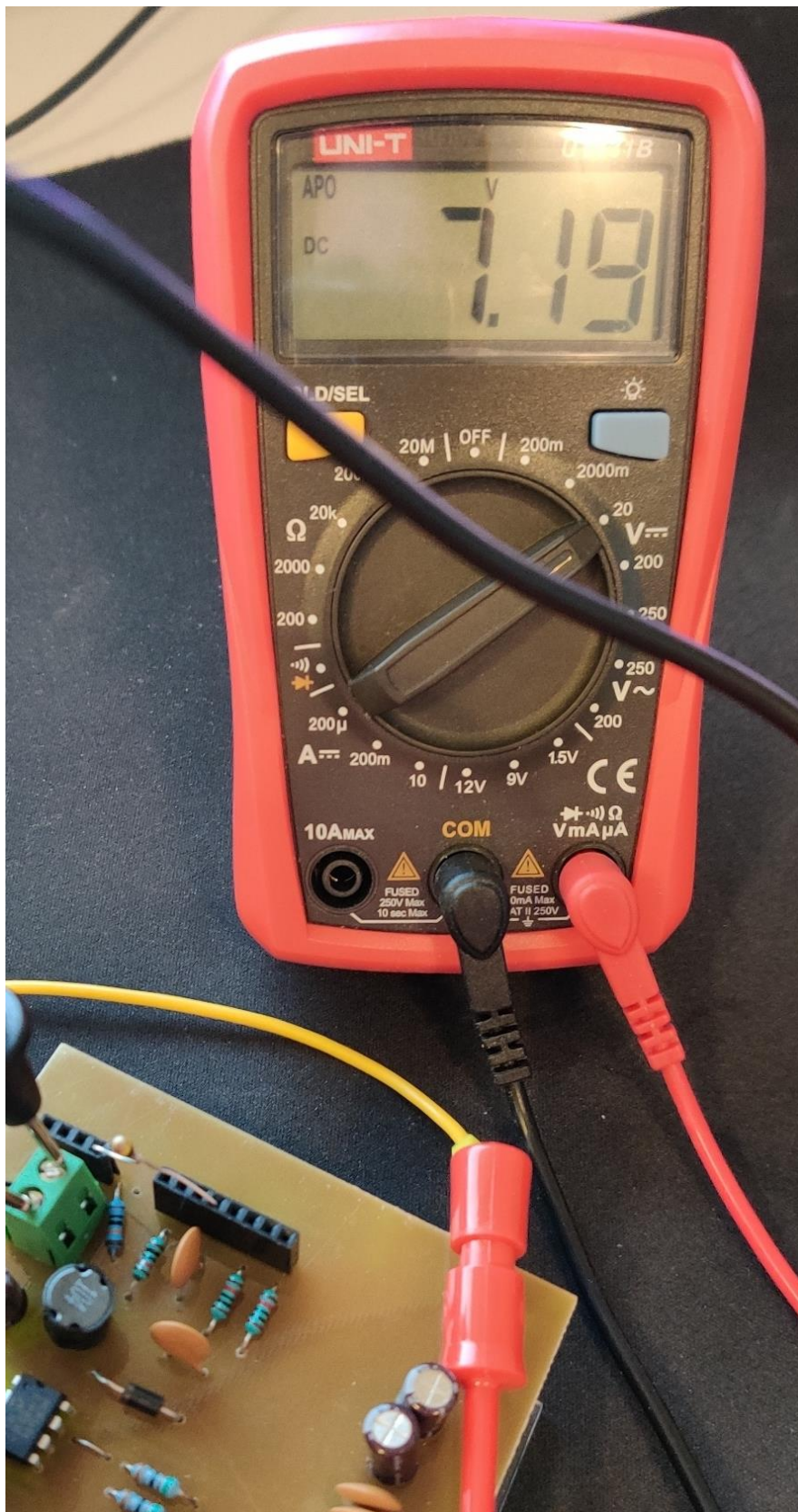
Temperature: Voltage: 7.22
Duty cycle: 30.59

Temperature: Voltage: 7.56
Duty cycle: 30.20

Temperature: Voltage: 7.32
Duty cycle: 30.20
```

Autoscroll Show timestamp Newline 115200 baud Clear output

Kuva 12. Terminaali tulokset



Kuva 13. Yleismittarin arvo

Mittasin kuorman jännitteen yleismittarilla ja tulokset vastasivat samaa mitä Arduino mittasi. **(Kuva 13.)**

5.2 NTC-vastuksella ohjattu arvo

Seuraavaksi testasin lämpötilan avulla säätää tuulettimen pyörimisnopeutta ja sain seuraavat arvot. **(Kuva 14.)**

```
Temperature: 27.10  
Voltage: 11.25  
Duty cycle: 93.33  
  
Temperature: 27.19  
Voltage: 11.28  
Duty cycle: 93.33
```

Kuva 14. Arvot käyttäen NTC-vastusta

Kun lämpötila kasvoi, niin tuuletin pyöri nopeammin ja sitten taas lämpötilan laskeutuessa se pyöri hitaammin.

6 LOPPUPÄÄTELMÄT

Projektissa pääosin suunniteltiin itse piirilevyä ja se vei suurimman osan ajasta. Piirilevyn suunnittelu meni muuten sulavasti, mutta oli välillä muutamia haasteita. Vaikka saatiin mitoitettua piirilevyn nastat, jotka kytketään Arduinon päälle painettaessa, tarkasti niin juotosvaiheessa nastat jäivät hieman vinoon ja se oli hieman hankalaa asettaa levy Arduinon päälle. Tästä johtuen, myös nastat jäivät vähän liian ylös ja välillä täytyi painaa levyä, jotta nastat saivat kontaktin kunnolla Arduinon. Komponenttien juotos vaiheessa, jotkut komponentit olivat hieman haastavia juottaa. Luultavasti johtuen siitä, että kupariin oli jäänyt hieman lakkaa, joka vaikeutti juottamista. Lopuksi saatiin kuitenkin juotettua kaikki komponentit ilman, että syntyi oikosulkuja.

Työtä olisi voinut jatkaa vielä niin, että se tulisi aitoon tuuligeneraattorin käytettäväksi. Piirilevy toimi kuitenkin onnistuneesti ja sillä saatiin ohjattua kuormassa olevaa laitetta koodissa asetettujen arvojen avulla. Käytännössä piirin kuormaan voisi kytkeä nyt mitä tahansa pienjännitelaitteita ja voisimme säätää jännitteen laitteen vaatimusten mukaan. Tällainen piiri levy on myös hyödyllinen, jos ei ole käytössä säädettävää jännitelähdettä.

LÄHTEET

Ahvonen, J. Lehtori. 2022. Arduino PWM. Sulautetut järjestelmät. Luentomateriaali. Vaasa. Vaasan ammattikorkeakoulu. Viitattu 3.4.2022

https://www.youtube.com/watch?v=4CWLR7C2IEQ&ab_channel=JaniAhvonen

Ahvonen, J. 2022. Arduino SMPS. Sulautetut järjestelmät. Luentomateriaali. Vaasa. Vaasan ammattikorkeakoulu. Viitattu 3.4.2022

https://www.youtube.com/watch?v=cqt6gMGXllg&ab_channel=JaniAhvonen

Ahvonen, J. 2022. PADS pikaopas. Tietokoneavusteinen elektroniikkasuunnittelu. Luentomateriaali. Vaasa. Vaasan ammattikorkeakoulu. Viitattu 3.3.2022

http://www.cc.puv.fi/~ja/PADS/PADS_2007_pikaopas_3.0.pdf

Arduino.cc, 2022 a. Arduino UNO R3. Viitattu 4.4.2022

<https://docs.arduino.cc/hardware/uno-rev3>

Arduino.cc, 2022 b. Secrets of Arduino PWM. Viitattu 9.5.2022

<https://docs.arduino.cc/tutorials/generic/secrets-of-arduino-pwm>

Mack, Raymond. 2008. Basic Switching Circuits. Power Sources and Supplies. Viitattu 17.5.2022

<https://www.sciencedirect.com/science/article/pii/B9780750686266000168>

Microchip, 2022. ATmega328P Datasheet.pdf. Viitattu 20.4.2022

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

LIITTEET

KOMPONENTTILISTA			
1.	1 kpl NTC 100 k/25C transistori		
2.	1 kpl IRF9640 mosfet transistori		
3.	1 kpl MCP1406 IC-piiri		
4.	1 kpl kela 100 uH		
5.	8 kpl 47 nF kondensaattori		
6.	3 kpl 47 uF/50V kondensaattori		
7.	1 kpl 100k ohm vastus		
8.	5 kpl 10k ohm vastus		
9.	1 kpl 150 ohm vastus		
10.	2 kpl 5,1 ohm vastus		
11.	1 kpl LTS12-35 12 V laitetuuletin		
12.	1 kpl 1N5818 diodi		
13.	1 kpl, Arduino UNO-kehitysalusta		

