



Zakaria Goole

# Tinkercadin käyttöönotto Arduinolle

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Konetekniikka

Insinöörityö

31.03.2023

# Tiivistelmä

Tekijä: Zakaria Goole  
Otsikko: Tinkercadin käyttöönotto Aduinolle  
Sivumäärä: 29 sivua  
Aika: 31.03.2023

Tutkinto: Insinööri (AMK)  
Tutkinto-ohjelma: Konetekniikka tutkinto-ohjelma  
Ammatillinen pääaine: Koneautomaatio  
Ohjaaja: Lehtori Timo Junell

---

Tämä opinnäytetyö keskittyy Arduino-ohjelmointiin ja Tinkercad-simulaation käyttöön. Tavoitteena on luoda käyttöohje ja opetusmateriaali, joka sopii tekniikan alkuvaiheessa oleville opiskelijoille ja tekniikasta kiinnostuneille henkilöille.

Opinnäytetyö tehdään Digitally-hankkeelle, jonka tarkoituksena on kehittää uudenlaisia hydridioppimisympäristöjä tukemaan koneteknologia-alan digitalisaatiota. Teoriaosuudessa käsitellään Digitally-hankkeen tavoitteita, Tinkercadin ja Arduino-kehitysalustan rakennetta sekä ohjelmoinnin perusteita.

Opetusmateriaali koostuu Tinkercadin käyttöohjeesta ja viidestä harjoituksesta, joiden tarkoituksena on perehdyttää opiskelija Tinkercad-suunnittelutyökalun hallintaan ja Arduino-ohjelmoinnin peruskäsitteisiin.

Lopputuloksena muodostui yksinkertainen opintokokonaisuus, jossa oppija pääsee hallitsemaan perusasiat, kuten Tinkercadin toimintaa ja kehitysalustan erilaisten liitännöiden merkityksiä. Opintokokonaisuudessa käydään myös läpi erilaisten Arduino-komponenttien ymmärrystä sekä ohjelmoinnin perusteiden hallitsemista.

Avainsanat: Arduino, IDE, Tinkercad, käyttöohje, opetusmateriaali, suunnittelutyökalu, C++, elektroniikka-alusta

## Abstract

Author: Zakaria Goole  
Title: Tinkercad implementation for Arduino  
Number of Pages: 29 pages  
Date: 31 March 2023

Degree: Bachelor of Engineering  
Degree Programme: Mechanical Engineering  
Professional Major: Machine Automation  
Supervisor: Lecturer Timo Junell

---

This thesis focuses on Arduino programming and the use of Tinkercad simulations. The aim is to create a user manual and teaching material suitable for students in the early stages of engineering and for people interested in engineering.

The thesis is done for the Digitally project, which aims to develop new types of hybrid learning environments to support the digitalization of the engineering technology sector. The theoretical part covers the objectives of the Digitally project, the structure of Tinkercad and the Arduino development platform, and the basics of programming.

The teaching material consists of Tinkercad user manual and five exercises aimed at familiarizing the student with the management of the Tinkercad design tool and the basic concepts of Arduino programming.

The end result was a simple tutorial in which the learner gets to master the basics such as the operation of Tinkercad and the meaning of the different interfaces of the development platform, understanding the different Arduino components and mastering the basics of programming.

Keywords: Arduino, IDE, Tinkercad, user-manual, teaching material, design tool, C++, hardware

# Sisällys

## Lyhenteet

1	Johdanto	1
2	Arduino ja Tinkercad	2
2.1	Arduino	2
2.2	Arduinon komponentit	3
2.3	Arduinon ohjelmointi	4
2.4	Ohjelmoinnin rakenne	5
2.5	Ohjelmointikomennot	7
2.6	Tinkercad	8
2.7	Tinkercadin hyödyt Arduinolle	9
3	Tinkercadin käyttöohje	10
3.1	Tilin tekeminen	10
3.2	Piirin rakentaminen	12
4	Opetusmateriaali	13
4.1	LED-valon simulointi	14
4.2	LED-valon ohjelmointi	15
4.3	LED-valon päälle/pois-Painonapilla simulointi	17
4.4	LED-valon päälle/pois-Painonapilla ohjelmointi	18
4.5	Liikennevalon simulointi	18
4.6	Liikennevalon ohjelmointi	19
4.7	Servomoottorin simulointi	21
4.8	Servomoottorin ohjelmointi	22
4.9	TMP36-lämpötila-anturin simulointi	23
4.10	TMP36-lämpötila-anturin ohjelmointi	25
5	Yhteenveto	27
	Lähteet	28

## Lyhenteet

Arduino: Avoimeen laitteistoon perustuva mikro-ohjain-/elektroniikka-alusta ja ohjelmointiympäristö.

Tinkercad: Verkkopohjainen 3D-suunnittelu- ja mallinnustyökalu, jonka avulla käyttäjät voivat luoda ja simuloida elektronisia piirejä.

IDE: *Integrated Development Environment*, Arduinon integroitua kehitysympäristöä.

C/C++: Ohjelmointikieli

LED: *Light Emitting Diode*, valodiodi

GND: *Ground*, Jännitteen 0-taso (maa)

I/O: Input/Output-liittimet, lähtö- ja sisääntulo

IC: *Integrated Circuit*, mikrokontrolleri

## 1 Johdanto

Tämän opinnäytetyön tavoitteena oli tehdä Tinkercad-käyttöönotto, käyttöohje ja opetusmateriaalikonaisuus harjoituksineen Aduinolle. Lisäksi tarkoituksena on perehtyä syvemmin Arduino- ja Tinkercad-ohjelmiin ja niiden tuomiin mahdollisuuksiin. Opinnäytetyö tehdään Digitally-hankkeelle, jonka tarkoitus on kehittää uudenlaisia hydridioppimisympäristöjä tukemaan koneteknologia-alan digitalisoitumista (DigiTally-hanke 2020).

Hankkeen päärahoittajana toimii Euroopan sosiaalirahasto (ESR) yhdessä Hämeen elinkeino-, liikenne- ja ympäristökeskuksen kanssa. Hanke rahoitetaan osana Euroopan unionin covid-19-pandemian johdosta toteuttamia toimia. (DigiTally-hanke 2020.)

Opinnäytetyön teoriaosuudessa käydään läpi Tinkercadin ja Arduino-kehitysalustan rakenne ja toimintaperiaate. Lisäksi käydään läpi Arduinon ohjelmoinnin perusteet. Opinnäytetyön kohderyhmä on tekniikan alkuvaiheessa olevat opiskelijat ja tekniikasta kiinnostuneet henkilöt. Lähtökohtana opetusmateriaalissa ei vaadita aiempaa kokemusta tietotekniikasta ja materiaalissa on pyritty käymään asiat läpi mahdollisimman yksinkertaisesti.

Opinnäytetyön toisessa osassa käydään läpi Tinkercadin piirivaihtoehdon hallintaa ja sen tarjoamat ominaisuudet, sekä Arduinon soveltaminen Tinkercadin piirisuunnittelussa ja ohjelmoinnin valinta. Opetusmateriaali koostuu käyttöohjeesta ja viidestä harjoituksesta. Tavoitteena on tehdä oppilaalle tutuksi Tinkercad-simulaation hallinta sekä Arduino-ohjelmoinnin peruskäsitteisiin perehtyminen.

Lopputuloksena muodostui yksinkertainen opintokokonaisuus, jossa oppija pääsee hallitsemaan perusasiat kuten, Tinkercadin toimintaa ja kehitysalustan erilaisten liitännöiden merkityksiä erilaisten Arduino-komponenttien ymmärrystä sekä ohjelmoinnin perusteen hallitsemista

## 2 Arduino ja Tinkercad

### 2.1 Arduino

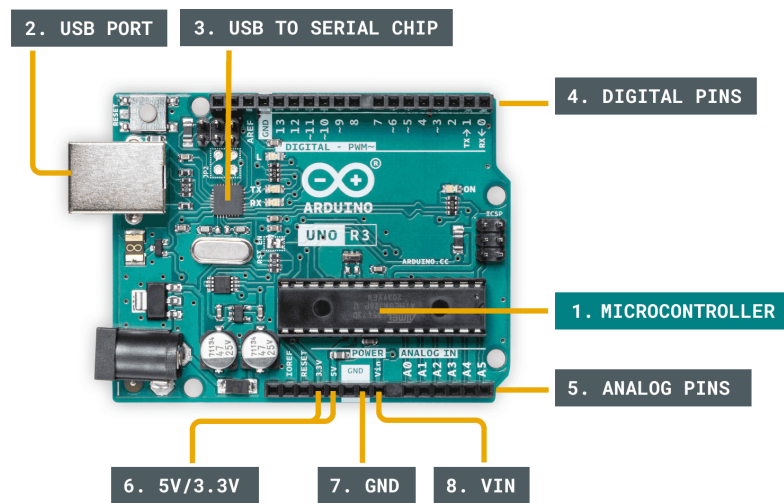
Arduino on avoimen lähdekoodin elektroniikka-alusta, jota käytetään erilaisten elektronisten laitteiden rakentamiseen. Se perustuu mikrokontrollerilevyyn, jonka avulla käyttäjät voivat helposti kirjoittaa ja ladata levyille koodia, jota voidaan sitten käyttää monenlaisten elektroniikkakomponenttien, kuten LEDien, moottoreiden ja antureiden ohjaamiseen. (Arduino Introduction 2018.)

Arduinon tärkeimmistä ominaisuuksista on sen käyttäjäystävällisyys ja avoin lähdekoodi. Alusta on suunniteltu siten, että se on kaikkien taitotasojen käyttäjien saatavilla, ja sen helppokäyttöiset ohjelmistot ja laitteistot mahdollistavat sen, että kuka tahansa voi luoda omia elektroniikkaprojektejaan. Tämä on tehnyt Arduinosta suosittua valintaa niin harrastajien, opettajien kuin ammattilaistenkin keskuudessa. (Arduino Introduction 2018.)

Arduinon keskeinen ominaisuus on sen monipuolisuus. Alustan avulla voidaan luoda monenlaisia elektronisia laitteita yksinkertaisista LED-vilkkuvaloista monimutkaisempiin projekteihin, kuten roboteihin ja kotiautomaatiojärjestelmiin. Tämä tekee siitä loistavan työkalun kaikille, jotka ovat kiinnostuneita elektroniikan maailmaan tutustumisesta ja omien projektien rakentamisesta. (Savasgard 2015.)

Käyttäjäystävällisyytensä ja monipuolisuutensa lisäksi Arduino tunnetaan myös laajasta ja aktiivisesta käyttäjäyhteisöstään. Alustalla on vilkas käyttäjäkunta, joka jakaa omia projektejaan ja ideoitaan ja tarjoaa tukea ja opastusta muille, jotka ovat kiinnostuneita käyttämään Arduinoa omissa projekteissaan. Tämä tekee siitä loistavan paikan, jossa aloittelijat voivat oppia ja saada tukea elektroniikan maailmaan tutustuessaan. (Savasgard 2015.)

## 2.2 Arduinon komponentit



Kuva 1. Arduino Uno R3-levy (Anatomy of an Arduino Board 2023)

1. **Mikrokontrolleri** - tämä on Arduinon aivot (Integrated Circuit), ja se on komponentti, johon ladataan ohjelmia. Ajatellaan sitä pienenä tietokoneena, joka on suunniteltu suorittamaan vain tiettyä määrää asioita.
2. **USB-portti** - käytetään Arduino-levyn liittämiseen tietokoneeseen.
3. **USB to Serial -siru** - USB to Serial on tärkeä komponentti, koska se auttaa muuntamaan esimerkiksi tietokoneelta tulevaa dataa sisäiseen mikrokontrolleriin. Tämä mahdollistaa Arduino-levyn ohjelmoinnin tietokoneeltasi.
4. **Digitaaliset nastat** - nastat, jotka käyttävät digitaalista logiikkaa (0,1 tai LOW/HIGH). Käytetään yleisesti kytkimiin ja LED-valojen kytkemiseen päälle/pois.
5. **Analogiset nastat** - nastat, jotka voivat lukea analogisia arvoja 10 bitin resoluutiolla (0-1023).



6. **5 V / 3,3 V nastat** - näitä nastoja käytetään ulkoisten komponenttien virransyöttöön.
7. **GND** - tunnetaan myös nimellä ground, negative tai yksinkertaisesti -, käytetään täydentämään piiri, jossa sähkötaso on 0 voltia.
8. **VIN** - tarkoittaa Voltage In, johon voi liittää ulkoisia virtalähteitä. (Anatomy of an Arduino Board 2023.)

### 2.3 Arduinon ohjelmointi

Arduinon ohjelmointia varten pitää käyttää Arduino-ohjelmointikieltä, joka perustuu C/C++:aan. Voidaan käyttää Arduinon integroitua kehitysympäristöä (Arduino Integrated Development Environment, IDE) ohjelmoinnin kirjoittamiseen ja lataamiseen Arduino-levylle. (Overview of the Arduino IDE 2023.)

Arduino IDE on saatavilla ilmaiseksi, ja sen voi ladata Arduinon verkkosivuilta. Jos halutaan aloittaa ohjelmointi Arduinolla, voidaan tutustua Arduinon viralliseen dokumentaatioon, joka tarjoaa kattavan johdannon Arduino-alustaan ja Arduino-ohjelmointikieleen. Verkossa on myös monia opetusohjelmia ja foorumeita, joissa voit oppia lisää Arduinosta ja saada apua projekteihisi. (Arduino IDE 2 Tutorials 2023.)

Harrastajat, opiskelijat ja ammattilaiset käyttävät Arduino IDE:tä laajasti ohjelmointiin ja vuorovaikutukseen Arduino-levyjen kanssa, ja sillä on suuri yhteisö ja tuki. Se on helppokäyttöinen ja yhteensopiva monien arduino-levyjen kanssa, ja se on saatavilla myös eri käyttöjärjestelmille, kuten Windows, MacOS ja Linux. (Arduino IDE 2022.)

## 2.4 Ohjelmoinnin rakenne

Arduino-ohjelmointikieli jaetaan kolmeen pääosaan: funktiot, arvot (muuttujat ja vakiot) ja rakenne. Kuvan 2 esimerkissä nähdään pääfunktiot `setup()` ja `loop()`, jotka ovat oleellisia Arduino-ohjelmointikielen rakenteeseen liittyvät komennot. Alla selitetty lyhyesti rakenteen toimintaperiaate ja yleisiä ohjelmointikomentoja. (Fezari & Dahoud 2018: 10 - 11.):



```
Classic_Blink_LED | Arduino 1.0.5-r2
File Edit Sketch Tools Help
[Icons] Verify
Classic_Blink_LED $
const int LED = 13;

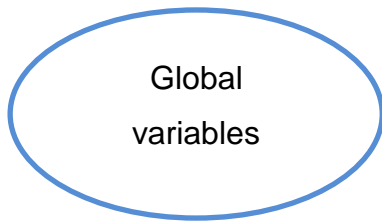
void setup()
{
  pinMode(LED,OUTPUT);
}

void loop()
{
  digitalWrite(LED,HIGH);
  delay(1000);
  digitalWrite(LED,LOW);
  delay(1000);
}

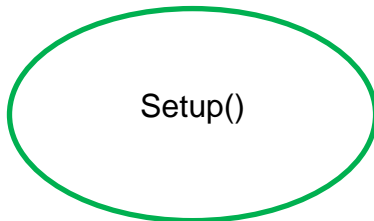
Done compiling.

Binary sketch size: 1,076 bytes (of a 32,256 byte maximum)
```

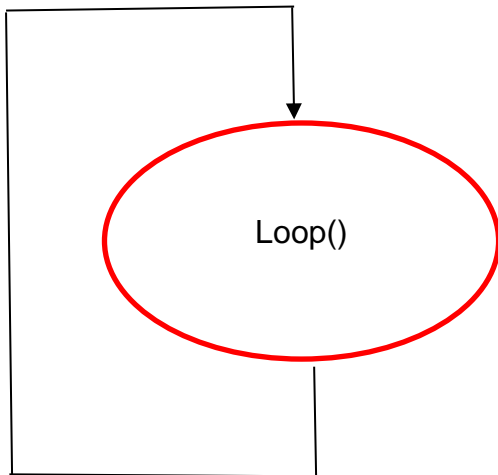
Kuva 2. Arduino ide -ohjelmointiesimerkki (Blink LED with Arduino 2014)



Ilmoitettava globaalit muuttujat ja instanssit, joita käytetään myöhemmin. Lyhyesti sanottuna muuttujan avulla voit nimetä ja tallentaa arvon, jota käytetään tulevaisuudessa. (Fezari & Dahoud 2018: 10 - 11.)



Jokaisessa Arduino-luonnoksessa on oltava setup-funktio. Tämä funktio määrittelee Arduinon alkutilan käynnistyksen yhteydessä, ja se suoritetaan vain kerran. (Fezari & Dahoud 2018: 10 - 11.)



Loop-funktio on myös välttämätön jokaisessa Arduino-luonnoksessa ja se suoritetaan, kun setup() on valmis. Se on pääfunktio, joka toimii toistuvasti yhä uudelleen ja uudelleen. (Fezari & Dahoud 2018: 10 - 11.)

## 2.5 Ohjelmointikomennot

Ohjelmointikomennot mahdollistavat ohjelmien jäsentämisen koodisegmenteiksi yksittäisten tehtävien suorittamiseksi. Tyypillinen tapaus funktion luomiselle on, kun sama toiminto on suoritettava useita kertoja ohjelmassa. (Arduino - Functions 2023.)

Alla on koottu lista muuttamatta yleisiä ohjelmointikomentoja ja selitetty niiden toiminta (Arduino programming language 2023.):

Taulukko 1. Funktiot

Funktiot	
pinMode()	Funktion avulla voidaan määritellä pinni sisääntuloksi/ulostuloksi.
digitalWrite()	Asettaa digitaalisen pinnin tilan, HIGH/LOW.
digitalRead()	Lukee digitaalisen pinnin tilan, HIGH/LOW.
analogWrite()	Muuttaa tulo annettua PWM (pulssinleveysmodulaatio), arvo väliltä 0-255.
analogRead()	Lukee analogisessa sisääntulopinnissä valitseva jännitteen, arvo väliltä 0-1023.
delay()	Pysäyttää ohjelman annettujen millisekuntien ajaksi.
serial()	Kommunikoimiseen arduinon ja muiden laitteiden välillä.

Taulukko 2. Muuttujat ja vakiot

Muuttujat, vakiot	
INPUT	Käytetään silloin kun pinni halutaan sisääntuloksi.
OUTPUT	Käytetään silloin kun pinni halutaan ulostuloksi.
HIGH	Käytetään asettamaan tietty liitinnasta päälle.
LOW	Käytetään asettamaan tietty liitinnasta pois päältä.
void	Käytetään funktioiden kanssa.
int	(integer=kokonaisluku)Tallettaa lukuja -32,768 ja 32,767 väliltä
string	Käytetään tekstiä sisältävien arvojen kanssa.
char	Tallentaa kirjaimia ja numeroita ASCII muodossa.

## 2.6 Tinkercad

Tinkercad on käyttäjäystävällinen, verkkopohjainen 3D-suunnittelu- ja mallinustyökalu, jonka avulla käyttäjät voivat luoda ja simuloida elektronisia piirejä. Tinkercadin piirivaihtoehtojen avulla käyttäjät voivat luoda ja testata omia elektronikkapiirejään, jolloin he voivat kokeilla eri osia ja nähdä, miten ne ovat vuorovaikutuksessa keskenään. (Reyes: 3.)

Erilaisten komponenttien, kuten kondensaattoreiden, transistorien ja jopa mikrokontrollereiden käyttäminen piirien rakentamiseen on yksi Tinkercadin piirivaihtoehtojen parhaista puolista. Tämän ansiosta käyttäjät voivat rakentaa ja testata monenlaisia piirejä yksinkertaisista LED-lampuista monimutkaisempiin piireihin, jotka voivat käyttää moottoreita tai lukea antureita. (Reyes: 3.)

Tinkercad tarjoaa erilaisia komponentteja sekä useita käytännön ominaisuuksia, jotka helpottavat käyttäjien piirien luomista ja testaamista. Käyttäjät voivat esimerkiksi käyttää simulointityökalua tutkiakseen, miten heidän piirinsä toimisi reaaliajassa, jolloin he voivat testata ja säätää suunnittelua ennen sen varsinaista

luomista. Käyttäjät voivat löytää ja korjata mahdolliset virheet ennen piirin rakentamista, mikä voi säästää paljon aikaa ja työtä. (Official guide to Tinkercad Circuits 2021.)

Tinkercadin käyttäjäystävällinen käyttöliittymä on toinen erinomainen ominaisuus, jonka ansiosta aloittelijoiden on helppo aloittaa piirien suunnittelu. Käyttäjät voivat helposti navigoida käyttöliittymässä sen yksinkertaisuuden ja selkeiden ohjeiden ansiosta, jotka tarjoavat myös hyödyllisiä vihjeitä. Tämän ansiosta jopa ne, joilla ei ole aiempaa tietoa piirien suunnittelusta, voivat omaksua perusteet nopeasti ja aloittaa omien piiriensä suunnittelun. (Tinkercad circuits 2023.)

## 2.7 Tinkercadin hyödyt Arduinolle

Tinkercad-simulaatio on saanut sen suursuosion siitä, että se auttaa harrastajat, opiskelijat ja ammattilaiset suunnittelemaan Arduinon ilman fyysistä elektronikka-alustaa (Sankhla 2023). Alle on kirjattu tinkercadin hyödyt arduinolle:

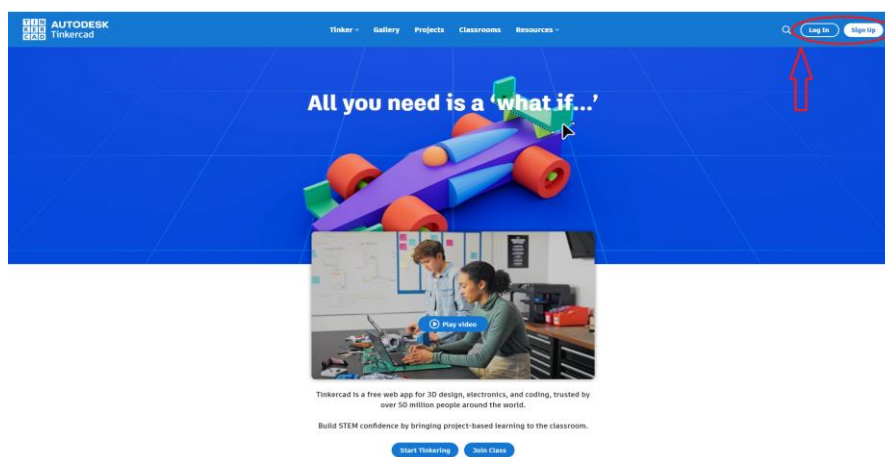
- Arduinon luominen ja oppiminen on helppo aloittelijoille.
- Opiskelija voi suunnitella ja simuloida piirejä ja laitteita ennen niiden fyysistä rakentamista.
- Siinä on sisäänrakennettu koodieditori Arduino-ohjelmoinin kirjoittamiseen ja testaamiseen.
- Auttaa opiskelijoita löytämään virheensä helposti simulaattorin avulla.
- Ei tarvitse huolehtia levyn vaurioista eikä sotkuisista lankarakenteista.
- Piirien rakentaminen, koodaus ja simulointi onnistuvat missä tahansa vain tietokoneella ja internetyhteydellä.
- Oman suunnittelun jakaminen muiden kanssa onnistuu vaivatta.
- Säästää aikaa ja resursseja antamalla käyttäjien kokeilla erilaisia suunnittelukokoonpanoja virtuaaliympäristössä. (Tukhtanazarovich 2021: 42 - 43.)

### 3 Tinkercadin käyttöohje

Tällä käyttöohjeella oppija oppii käyttämään Tinkercad-alustaa Arduino-simulointiin. Käyttäjätasvällisen rakenteensa ansiosta Tinkercad on yksi parhaista alustoista piirien luomiseen ja simulointiin. Tämä opas toimii pohjana Arduinolla toimivien piirien luomiselle Tinkercadissa. Aloitetaan tutustumalla Tinkercadiin.

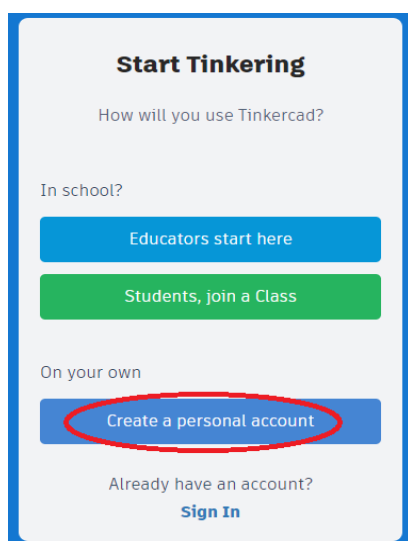
#### 3.1 Tilin tekeminen

Ensimmäiseksi pitää rekisteröityä ja luoda tili Tinkercad-selaimeen. Kuten kuvassa 3 näkyy, rekisteröityminen löytyy oikeasta yläkulmasta.



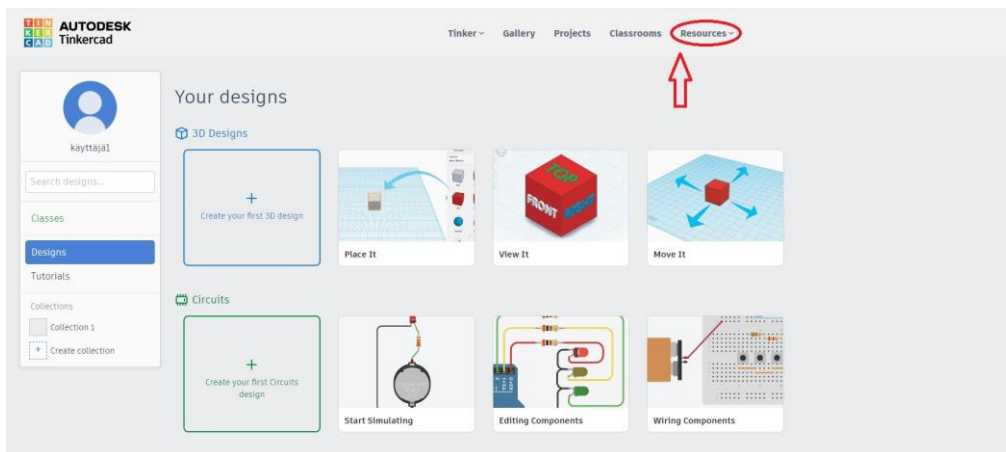
Kuva 3. Tinkercadin kotisivu

Tämän jälkeen saat monta eri vaihtoehtoa tilin luomiseen. Voit joko kirjautua sähköpostitunnuksillasi tai Google/Apple ID: lläsi.

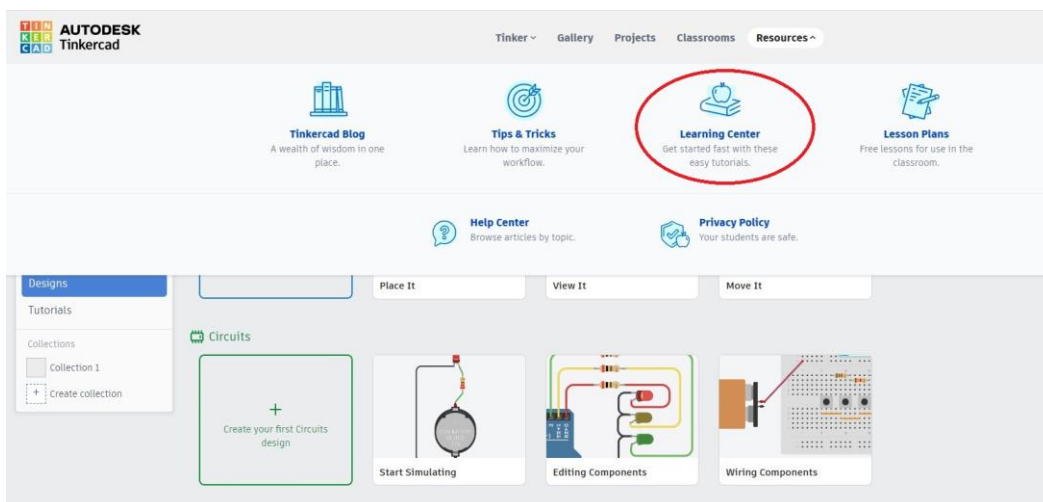


Kuva 4. Tilin luominen

Kun olet tehnyt tilin ja kirjautunut sisään, nähdään Tinkercadin etusivu (kuva 5), jossa näkyy erilaisia suunnittelumahdollisuuksia. Valitsemalla oikeasta yläkulmasta ”resource”-kohtaa, ja sen jälkeen klikkaamalla kohdasta ”learning center”, pääset seuraamaan opetusohjelmia alustan eri osien käytöstä (kuva 6).



Kuva 5. Tinkercadin etusivu



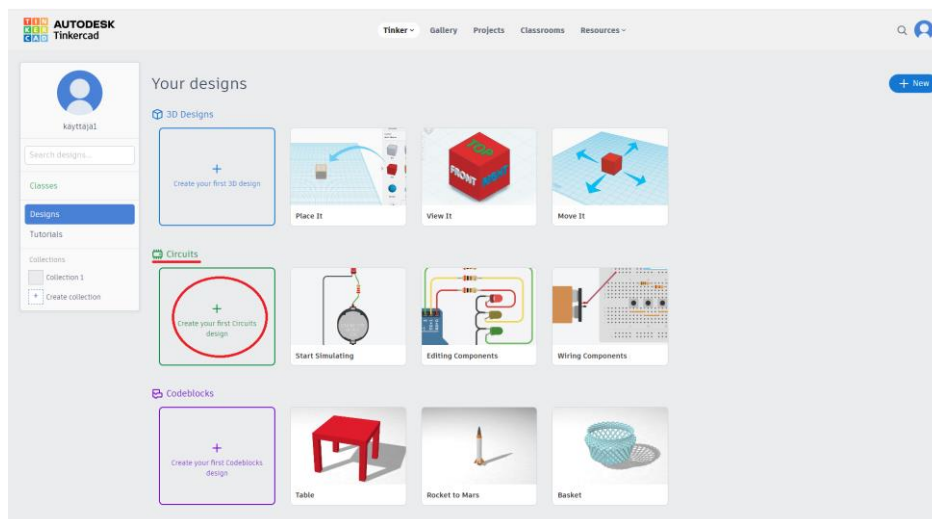
Kuva 6. Opetusohjelman valinta



## 3.2 Piirin rakentaminen

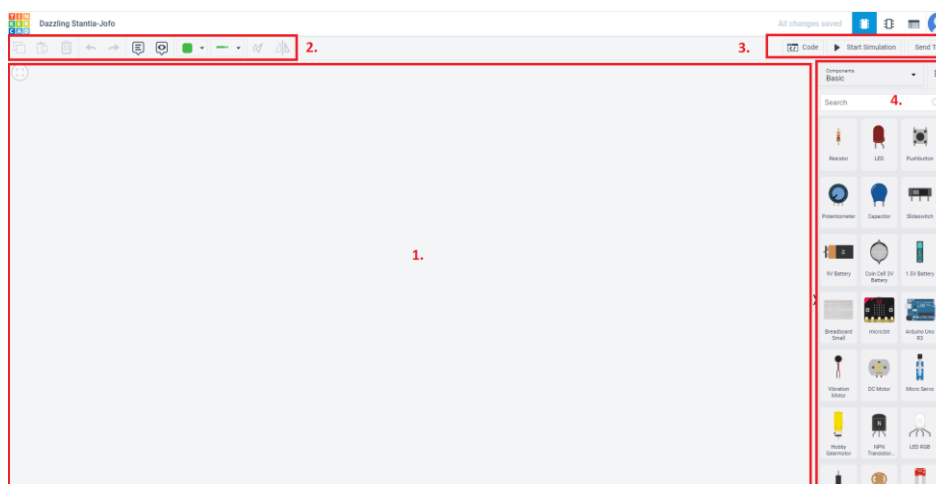
Aloittaaksesi piirien tekemisen, paina kohdasta ”create new circuit design”.

(Kuva 7)

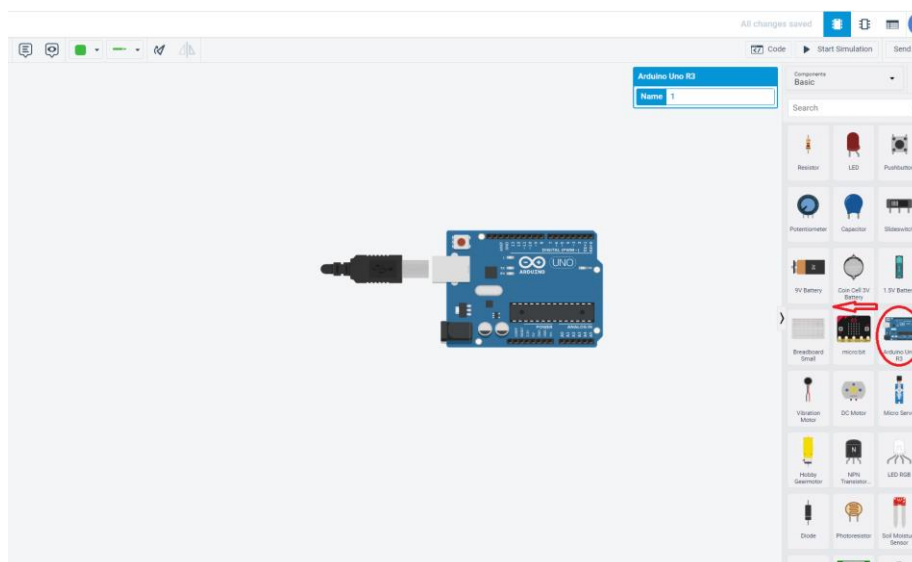


Kuva 7. Piirivalikoiman valinta

1. Kaikki komponentit menevät tähän paikkaan. Ne voidaan kiinnittää toisiinsa, siirtää ja muuttaa.
2. Kierrä, poista, kumoa tai tee uudelleen tällä välilehdellä. Käyttäjät voivat myös luoda ja nimetä komponenteille tarroja sen avulla.
3. Voit käyttää sarjamonitoria, käynnistää reaaliaikaisen simulaation, viedä koodia ja jakaa projektisi muiden kanssa käyttämällä tätä vaihtoehtoa.
4. Kaikki komponentit sisältyvät tähän osioon. Voit käyttää komponenttityyppejä selaamalla alas.



Kuva 8. Piirisuunnittelu sivu



Kuva 9. Arduinon valitseminen komponenttivalikoimasta

Arduinon rakentaminen tapahtuu valitsemalla vasemmasta kohdasta haluamasi levypiirin ja komponentti, sen jälkeen raahataan se työtilaan oikealle, kuten kuvassa 9 näkyy. Täällä voidaan yhdistää eri komponentteja Arduinoon, lisätä koodia ja simuloida, ja nähdä miltä näyttäisi, jos ladataan koodi fyysiseen Arduino-piiriin.

## 4 Opetusmateriaali

Opetusmateriaali koostuu viidestä harjoituksesta, joiden tarkoituksena on perehdyttää opiskelija Tinkercad-suunnittelutyökalun hallintaan ja Arduino-komponenttien ja ohjelmoinnin peruskäsitteisiin. Tavoitteena on, että oppija pääsee hallitsemaan perusasiat. Perusasioita ovat esimerkiksi Tinkercadin toiminta ja kehitysalustan erilaisten liitännöiden merkitykset, erilaisten Arduino komponenttien ymmärrys sekä ohjelmoinnin perusteen hallitseminen.

## 4.1 LED-valon simulointi

### Tavoite

Tavoitteena on saada LED-valo vilkkumaan.

### Tarvittavat komponentit

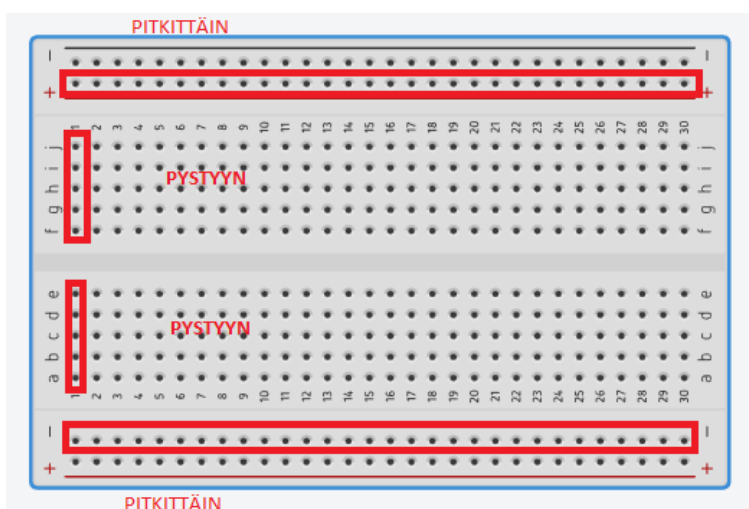
LED-valon simulointia varten tarvitsemme koekytkentäkaavion, resistanssin ja LED-lampun.

### KytKentäohje

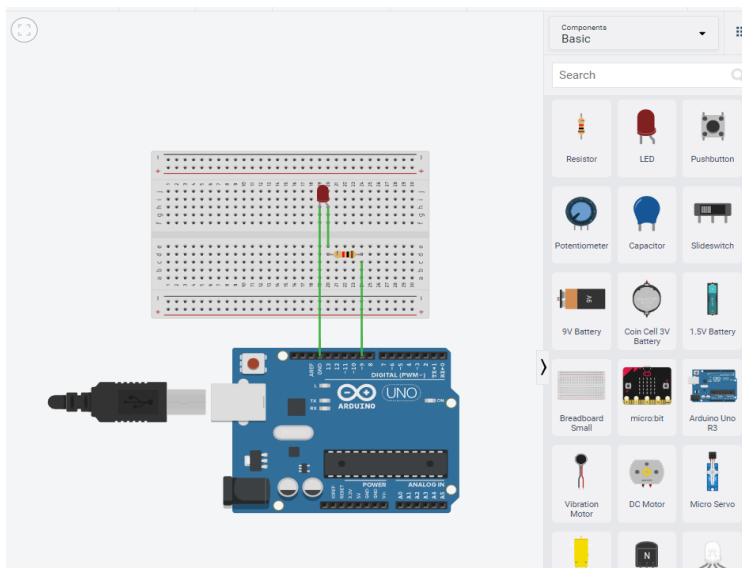
Ensimmäiseksi kytketään LED-lampun katodi (-) Arduino-maadoitukseen ja anodi (+) mihin tahansa digitaalisiin pinneihin (esim. 9). Jotta läpi kulkevan virran taso olisi samanlainen, lisätään resistanssi. Vahvuudeksi annetaan  $1\Omega$ .

Koekytkentälevy tai koekytkentäalusta (engl. breadboard) on levy, jolla voi testata elektronisia kytkentöjä tarvitsematta juottaa komponentteja (Kuva 10).

Kaksi ensimmäistä ja viimeistä riviä koekytkentälevyissä on virtalähde (+) punaisella ja maadoitus (-) mustalla ja siihen kytkeminen tapahtuu pitkittäin. A-J-reikiin kytketään komponentit ja yhdistetään toisiinsa pystyyn.



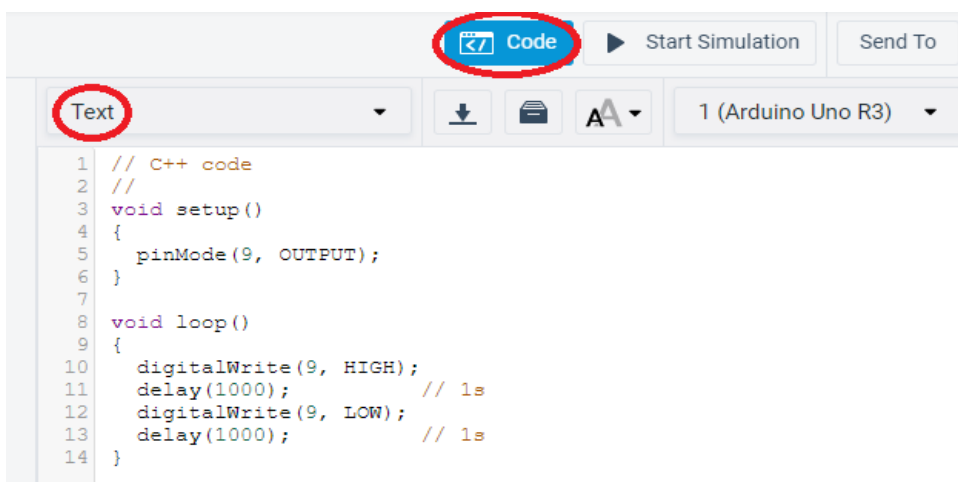
Kuva 10. Koekytkentälevy



Kuva 11. Kytetty LED-valo

## 4.2 LED-valon ohjelmointi

Piirien rakentamisen lisäksi Tinkercadissa pääset ohjelmoimaan. Arduinon-ohjelmointia varten Tinkercad antaa kolme vaihtoehtoa lohko-, teksti- tai lohko/tekstikoodaus. Tässä harjoittelussa keskitytään tekstivaihtoehtoon käyttämällä C/C++. Kuvassa 12 esimerkki ohjelmoinnin valikoinnista.



Kuva 12. Tekstiohjelmoinnin valinta

Setup()-funktiossa kirjoitettu koodi suoritetaan vain kerran. Tässä käytetään pinMode()-funktiota pinnin määrittämiseen syötteen/ulostuloksi. Tässä harjoituksessa määritetään pin 9 ulostuloksi kirjoittamalla LED\_BUILTIN kohtaan.

Seuraavaksi Loop()-funktio kohtaan kirjoitettu teksti toistuu niin pitkään kunnes simulaatio pysäytetään. digitalWrite()-funktioon kirjoitetaan tehtävä HIGH/LOW arvo pinneihin. Tällä tavoin saadaan Ledi päälle tai pois.

Lopuksi määritetään aika delay()-funktio. Tällä tavoin nähdään, kun led-valo vilkkuu. Ajaksi annetaan 1 sekunti.

### Ohjelmoinninohje

```
void setup()
{
  pinMode(9, OUTPUT);
}

void loop()
{
  digitalWrite(9, HIGH);
  delay(1000);
  // 1s

  digitalWrite(9, LOW);
  delay(1000);
  // 1s
}
```

### 4.3 LED-valon päälle/pois-Painonapilla simulointi

#### Tavoite

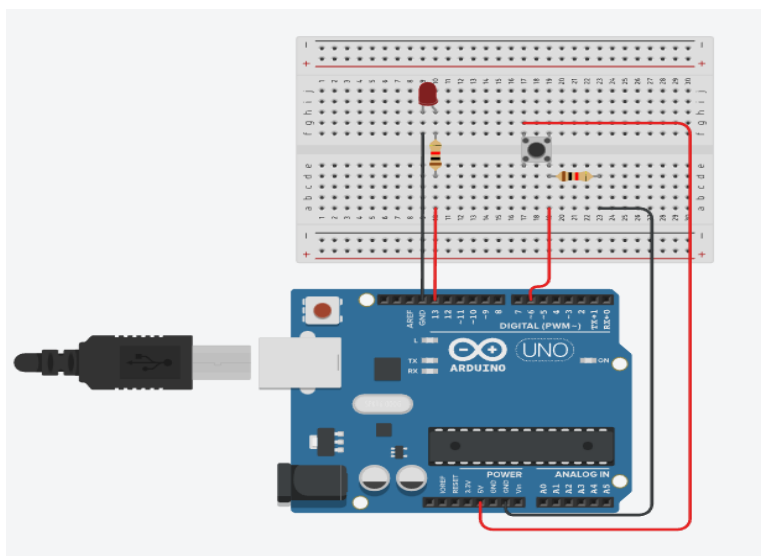
Tavoitteena on saada LED-valo päälle painiketta painamalla. LED-valo paalaa vain, kun pidetään painiketta pohjassa.

#### Tarvittavat komponentit

Tässä harjoittelussa tarvitaan painonappikytkin, LED-valo, kytkentäalusta ja kaksi resistanssia.

#### Kytkeäohje

Ensimmäiseksi kytketään LED-lampun katodi (-) Arduino-maadoitukseen ja anodi (+) digitaaliseen pinniin 13. Jotta läpi kulkevan virran taso olisi samanlainen, lisätään resistanssi. Seuraavaksi painonappikytkin kytketään Arduino-maadoitukseen (musta johto) ja lisätään resistanssi, sekä kytketään Arduinon 5V-virtalähteeseen (punainen johto). Lopuksi vielä painonappikytkin kytketään Arduinon digitaalipinniin 6. Kuvassa 13 nähdään valmis kytketty LED-valon päälle/pois-Painonapilla.



Kuva 13. Kytketty LED-valon päälle/pois-Painonapilla

#### 4.4 LED-valon päälle/pois-Painonapilla ohjelmointi

##### Ohjelmointiohje

```

int button = 0;
void setup ()
{
  pinMode (6, INPUT);
  pinMode (13, OUTPUT);
}
void loop ()
{
  button = digitalRead(6);
  if (button == HIGH) {
    digitalWrite (13, HIGH);
  } else {
    digitalWrite (13, LOW);
  }
  delay (10);      // Viivyttele hieman simulaation suorituskyvyn parantamiseksi
}

```

#### 4.5 Liikennevalon simulointi

##### Tavoite

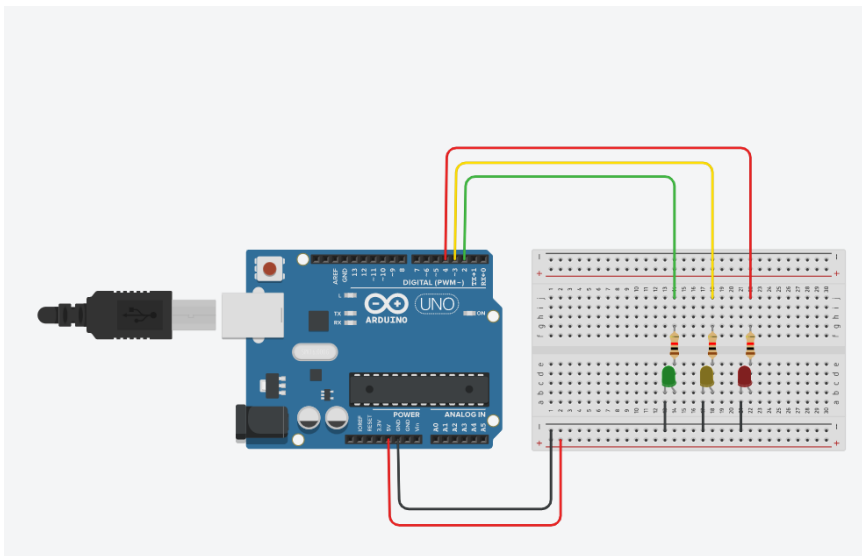
Tässä harjoittelussa simuloidaan liikennevaloa. Tavoitteena on saada kolme LED-lamppua vilkkumaan eri aikoihin (vihreä on päällä 5 s, keltainen vilkkuu 3 kertaa ja punainen päällä 5 s).

##### Tarvittavat komponentit

Komponentit joita tarvitaan tässä harjoittelussa ovat koekytkentälevy, kolme LED-lamppua (punainen, vihreä ja keltainen) sekä resistanssia kontrolloimaan kulkeva virta.

## KytKentäohje

Ensimmäiseksi kytketään koekytkentäkaavion maadoitus, arduinon maadoitukseen (-) mustalla johdolla. Punainen johto kytketään arduinon 5V-virtalähteeseen ja koekytkentäkaavion virtalähteen (+). Tämän jälkeen kolme lediä (vihreä, keltainen, ja punainen) kytketään pinneihin 2,3,4. Kuvassa 14 nähdään valmis kytketty liikennevalo.



Kuva 14. Kytketty liikennevalo

### 4.6 Liikennevalon ohjelmointi

1. Tavoitteena on saada toimiva koodi setup- ja loop-funktion, jotta vihreä, keltainen ja punainen syttyvät ja sammuvat järjestyksessä.
2. Määritetään jokainen LED-lampun digitaalinen pinni ulostuloksi setup-funktiossa.
3. Ohjelmoidaan Ledit järjestyksessä syttymään ja sammumaan loop-funktiossa.
4. Käytetään seuraavia neljää komentoa asianmukaisilla muuttujilla (vihreä, keltainen ja punainen) ja asianmukaisilla delay (viive) -käskyillä koodin täydentämiseksi:
  - pinMode (vihreä, OUTPUT); // vihreä LED pinni ulostuloksi.
  - digitalWrite (vihreä, HIGH); // LED päällä.



- digitalWrite (vihreä, LOW); // LED sammunut.
- delay (5000); // 5 sekunnin viive.

### Ohjelmointiohje

```
void setup()
{
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
}
void loop()
{
  digitalWrite(2,HIGH); //vihreä
  digitalWrite(3,LOW);
  digitalWrite(4,LOW);
  delay(5000); //5s

  digitalWrite(2,LOW);
  for (int counter=0;counter<3;++counter) //vilkuu 3 kertaa
  {
    digitalWrite(3,LOW); //keltainen sammunut
    delay(1000); //1s
    digitalWrite(3,HIGH); //keltainen päällä
    delay(1000); //1s
  }
  digitalWrite(2,LOW); //punainen
  digitalWrite(3,LOW);
  digitalWrite(4,HIGH);
  delay(5000); //5s
}
```

## 4.7 Servomoottorin simulointi

### Tavoite

Tässä harjoitellussa tavoitteena on ohjata servomoottoria Arduinolla. Servomootoreita käytetään usein robotiikkaprojekteissa, mutta niitä löytyy myös RC-autoista, lentokoneista jne. Ne ovat erittäin hyödyllisiä, kun tarvitaan tarkkaa asennon ohjausta ja/tai suurta vääntömomenttia.

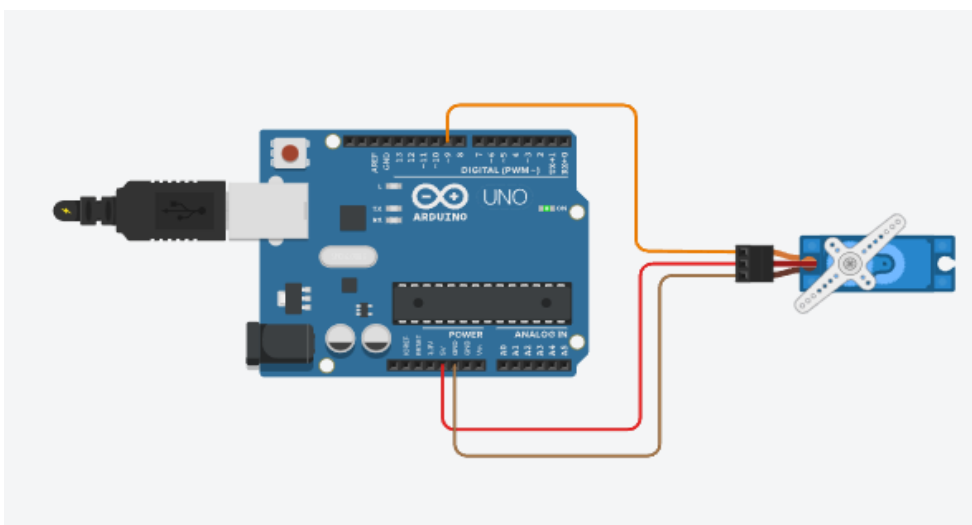
Opit liittämään servomoottorin Arduinoon ja ohjaamaan sitä koodilla.

### Tarvittavat komponentit

Tarvitaan kolme johtoa (ruskea, oranssi, ja punainen) sekä micro-servomoottori.

### Kytkeäohje

Kytke ruskea johto Arduinon maadoitusnapaan. Signaalijohto (oranssi) voidaan liittää mihin tahansa Arduinon digitaalisiin nastoihin. Tässä tapauksessa liitetään se digitaaliseen nastaan 9. Punainen johto kytketään Arduinon 5V-virtalähteeseen. Kuvassa 15 nähdään valmis kytketty servomoottori.



kuva 15. Kytketty servomoottori

## 4.8 Servomoottorin ohjelmointi

Yhdistetään määritetty servo-objekti nastaan, joka ohjaa servoa koodin setup-funktiossa. Ohjataan servomoottoria vain siirtymään tiettyyn kulmaan loop-funktion ensimmäisessä osassa. Huomaa, että tarvittavat delay-komentoja antaaksesi servomoottorille jonkin aikaa siirtyä asetettuun asentoon.

Kahdessa loop-koodin viimeisessä osassa määritetään servomoottorin pyyhkäisemiseen edestakaisin. Jos halutaan hallita servomoottorin nopeutta, tämä koodirivi voi olla hyödyllinen. Servovarren nopeutta voidaan muuttaa muuttamalla viive arvoa for-loopin lopussa.

### Ohjelmointiohje

```
// Luo uusi servo-objekti
Servo myservo;

// Määrittele servopinni
#define servoPin 9

// Luo muuttuja servoaseman tallentamiseksi
int angle = 0;

void setup()
{
  myservo.attach(9);           //Kiinnitä servo määrättyyn pinniin
}

void loop()
{
  //Määritetään servon siirtymään tiettyyn kulmaan
  myservo.write(90);
  delay(1000);
  myservo.write(180);
```

```
delay(1000);
myservo.write(0);
delay(1000);
for (angle = 0; angle <= 180; angle += 1)           //Pyyhkäisy 0 - 180 astetta
{
  myservo.write(angle);
  delay(15);
}
for (angle = 180; angle >= 0; angle -= 1)         //Ja takaisin 180 astetta 0 asteeseen
{
  myservo.write(angle);
  delay(30);
}
delay(1000);
}
```

#### 4.9 TMP36-lämpötila-anturin simulointi

##### **Tavoite**

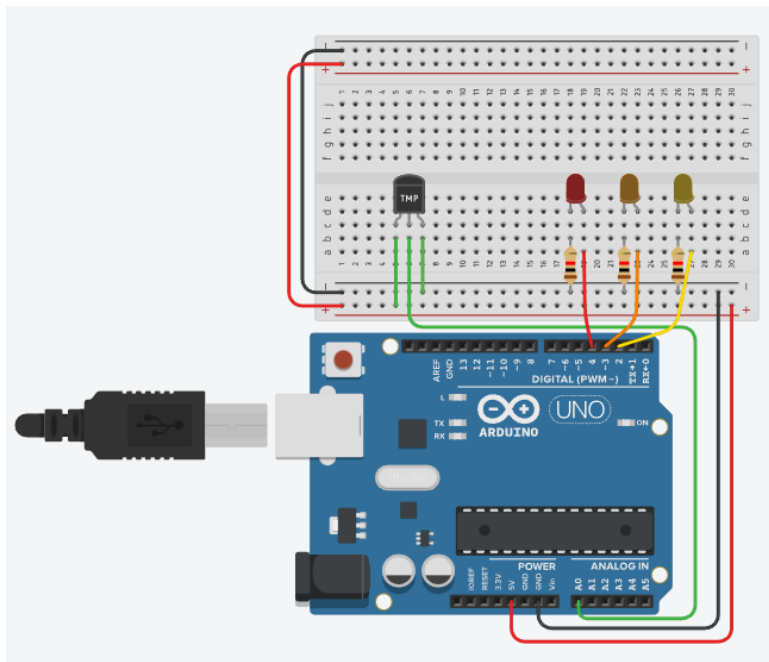
Tässä harjoittelussa tavoitteena on muuttaa Arduinosta lämpömittari. Käytetään lämpötila-anturia (TMP36) mitataksaan lämpötilaa, joka sitten rekisteröi ulostuloksi kolmella LEDillä. Koska Arduino on digitaalinen työkalu, se voi lukea signaaleja analogisista tuloista, kuten TMP36-lämpötila-anturista, käyttämällä sisäistä ADC-muunninta (Analog-to-Digital).

## Tarvittavat komponentit

Komponentit joita tarvitaan tässä harjoittelussa ovat koekytkentälevy, kolme LED-lamppua (punainen, oranssi ja keltainen) sekä resistanssia kontrolloimaan kulkevan virran. Tarvitaan vielä lämpömittauskomponentti, koska Tinkercadissa on ainoastaan yksi lämpötila-anturi (TMP36). Anturi mittaa pienen muutoksen ja antaa sen perusteella analogisen jännitteen välillä 0 - 1,75 VDC.

## KytKentäohje

- Vie kolme LEDiä kytkentälevyn riville D, jotka on erotettu toisistaan kahdella kytkentälevypistorasialla.
- Kytke kunkin LEDin katodi (vasen jalka) kytkentälevyn maadoitusjohtimeen 220 ohmin vastuksen avulla.
- Kytke Arduinon digitaaliset nastat 4, 3 ja 2 LED-anodeihin (oikea jalka).
- LED-katodi (-) yhdistetään maadoituskiskoon.
- Aseta lämpötila-anturi kytkentälevyn riville C.
- Kytke lämpötila-anturi niin, että sen nastat kytketään 5V:hen, A0:aan ja GND:hen.



Kuva 16. Kytetty TMP36-lämpötila-anturi

## 4.10 TMP36-lämpötila-anturin ohjelmointi

### Ohjelmointiohje

```
int baselineTemp = 0;
int celsius = 0;
int fahrenheit = 0;

void setup ()
{
  pinMode (A0, INPUT);
  Serial.begin (9600);

  pinMode (2, OUTPUT);
  pinMode (3, OUTPUT);
  pinMode (4, OUTPUT);
}

void loop ()
{
  // Aseta kynnyslämpötila LED-valojen aktivoimiseksi
  baselineTemp = 40;

  // mittaa lämpötila celsiusasteina
  celsius = map((analogRead(A0) - 20) * 3.04, 0, 1023, -40, 125);

  // muuntaa Fahrenheitiksi
  fahrenheit = ((celsius * 9) / 5 + 32);
  Serial.print (celsius);
  Serial.print (" C, ");
  Serial.print (fahrenheit);
  Serial.println (" F");
}
```

```
if (celsius < baselineTemp) {  
    digitalWrite(2, LOW);  
    digitalWrite(3, LOW);  
    digitalWrite(4, LOW);  
}  
if (celsius >= baselineTemp && celsius < baselineTemp + 10) {  
    digitalWrite(2, HIGH);  
    digitalWrite(3, LOW);  
    digitalWrite(4, LOW);  
}  
if (celsius >= baselineTemp + 10 && celsius < baselineTemp + 20) {  
    digitalWrite(2, HIGH);  
    digitalWrite(3, HIGH);  
    digitalWrite(4, LOW);  
}  
if (celsius >= baselineTemp + 20 && celsius < baselineTemp + 30) {  
    digitalWrite(2, HIGH);  
    digitalWrite(3, HIGH);  
    digitalWrite(4, HIGH);  
}  
if (celsius >= baselineTemp + 30) {  
    digitalWrite(2, HIGH);  
    digitalWrite(3, HIGH);  
    digitalWrite(4, HIGH);  
}  
    delay (1000);  
}
```

## 5 Yhteenveto

Opinnäytetyön tavoitteena oli kehittää Tinkercadin käyttöönotto Arduinolle, käyttöohje ja opetusmateriaali digitally-hankeelle, edistää koneteknologia-alan digitalisoitumista ja yhdistää lähi- ja etäopiskelu sujuvaksi kokonaisuudeksi. Kaikki opinnäytetyölle asetetut tavoitteet saavutettiin.

Työn alkuvaiheessa käytiin läpi ja opeteltiin ensin itse suuri määrä tietoa, josta sitten muodostui selkeä käyttöohje ja opetuskokonaisuus. Opinnäytetyön tiedon keräämistä helpotti Arduinon ja Tinkercadin vilkas käyttäjäyhteisö, jotka jakavat omia ideoitaan, tukevat ja opastavat muita.

Käyttöohje aloitettiin tekemällä käyttäjätili Tinkercad-sivustoon, josta sitten dokumentoitiin kaikki vaiheet. Opetusmateriaalia tehdessä kerättiin kymmenkunta harjoitusta, joista sitten valittiin viisi varsinaiseen materiaalidokumenttiin. Ensimmäiset harjoitukset on tarkoituksella tehty mahdollisimman yksinkertaiseksi, jotta oppilas ymmärtää sovelluksen periaatteen.

Lopputuloksena muodostui käyttöohje ja oppimateriaali, jossa käydään läpi Tinkercadin toimintaperiaate ja Arduinon ohjelmointi ja simulointi. Suurimpana tavoitteena oli kehittää mahdollisimman yksinkertainen opintokokonaisuus, jossa oppija pääsee hallitsemaan perusasiat kuten Tinkercadin toimintaa ja kehitysalustan erilaisten liitännöiden merkityksiä, erilaisten Arduino-komponenttien ymmärrystä sekä ohjelmoinnin perusteen hallitsemista.



## Lähteet

Anatomy of an Arduino Board. 2023. Verkkoaineisto. <<https://docs.arduino.cc/learn/starting-guide/getting-started-arduino>> Luettu 5.12.2022

Arduino - Functions. 2023 Verkkoaineisto. < [https://www.tutorialspoint.com/arduino/arduino\\_functions.htm#](https://www.tutorialspoint.com/arduino/arduino_functions.htm#)> Luettu 2.2.2023

Arduino IDE 2 Tutorials. 2023. Verkkoaineisto. <<https://docs.arduino.cc/software/ide-v2/tutorials/getting-started-ide-v2>> Luettu 8.12.2022

Arduino IDE. 2022. Verkkoaineisto. <https://encyclopedia.pub/entry/30906>> Luettu 23.1.2023

Arduino introduction. 2018. Verkkoaineisto.<<https://www.arduino.cc/en/Guide/Introduction>>Luettu 5.12.2022

Arduino programming language. 2023. Verkkoaineisto. <<https://www.arduino.cc/reference/en/>> Luettu 17.1.2023

Blink LED with Arduino. 2014 Verkkoaineisto. <<https://www.circuitstoday.com/blink-led-with-arduino>> Luettu 17.1.2023

Digitally -hanke. 2020. Verkkoaineisto. <<https://www.digitally.fi/>>Luettu 2.12.2022

Fezari & Dahoud. 2018. Integrated Development Environment "IDE" For Arduino. PDF-aineisto. <[https://www.researchgate.net/publication/328615543\\_Integrated\\_Development\\_Environment\\_IDE\\_For\\_Arduino](https://www.researchgate.net/publication/328615543_Integrated_Development_Environment_IDE_For_Arduino)> Luettu 12.1.2023

Official guide to Tinkercad Circuits. PDF-aineisto. 2021.<[https://assets.ctfassets.net/jl5ii4oqrdbc/4sMFqe3rDlBjUymJt0l4yh/85a4487f7fe274e74c19870ae4679fc1/tinkercad-guides\\_circuits-Printable.pdf](https://assets.ctfassets.net/jl5ii4oqrdbc/4sMFqe3rDlBjUymJt0l4yh/85a4487f7fe274e74c19870ae4679fc1/tinkercad-guides_circuits-Printable.pdf)> Luettu 25.1.2023

Overview of the Arduino IDE 1. 2023. Verkkoaineisto. <<https://docs.arduino.cc/software/ide-v1/tutorials/Environment>> Luettu 7.12.2022

Reyes, Alex. Electronic Circuit Basics with Tinkercad. E-kirja. <<http://www.energiazero.org/simulazioni/Intro%20thinkercad.pdf>> Luettu 21.1.2023

Sankhla. 2023. Basics of Arduino (TINKERCAD). 2023. PDF-aineisto. <<https://ibles-api.tinkercad.com/json-api/viewPdf?id=ELR3AR7LEBEG2MD&trackTinkercadUsage=1>> Luettu 13.2.2023

Savasgard, Erik. 2015. Arduino - 101 Beginners Guide How to get started with Your Arduino. E-kirja. Luettu 2.2.2023

Tinkercad circuits. 2023. Verkkoaineisto. <<https://www.tinkercad.com/circuits>> Luettu 25.1.2023

Tukhtanazarovich, Jumabayev. 2021. Advantage of the Arduino platform in forming creative skills in youth. Novateur Publication. JournalNX. Vol 7,s.41-43. <<https://repo.journalnx.com/index.php/nx/article/download/3505/3407/6844>> Luettu 1.2.2023