

## 4. Harjoitus

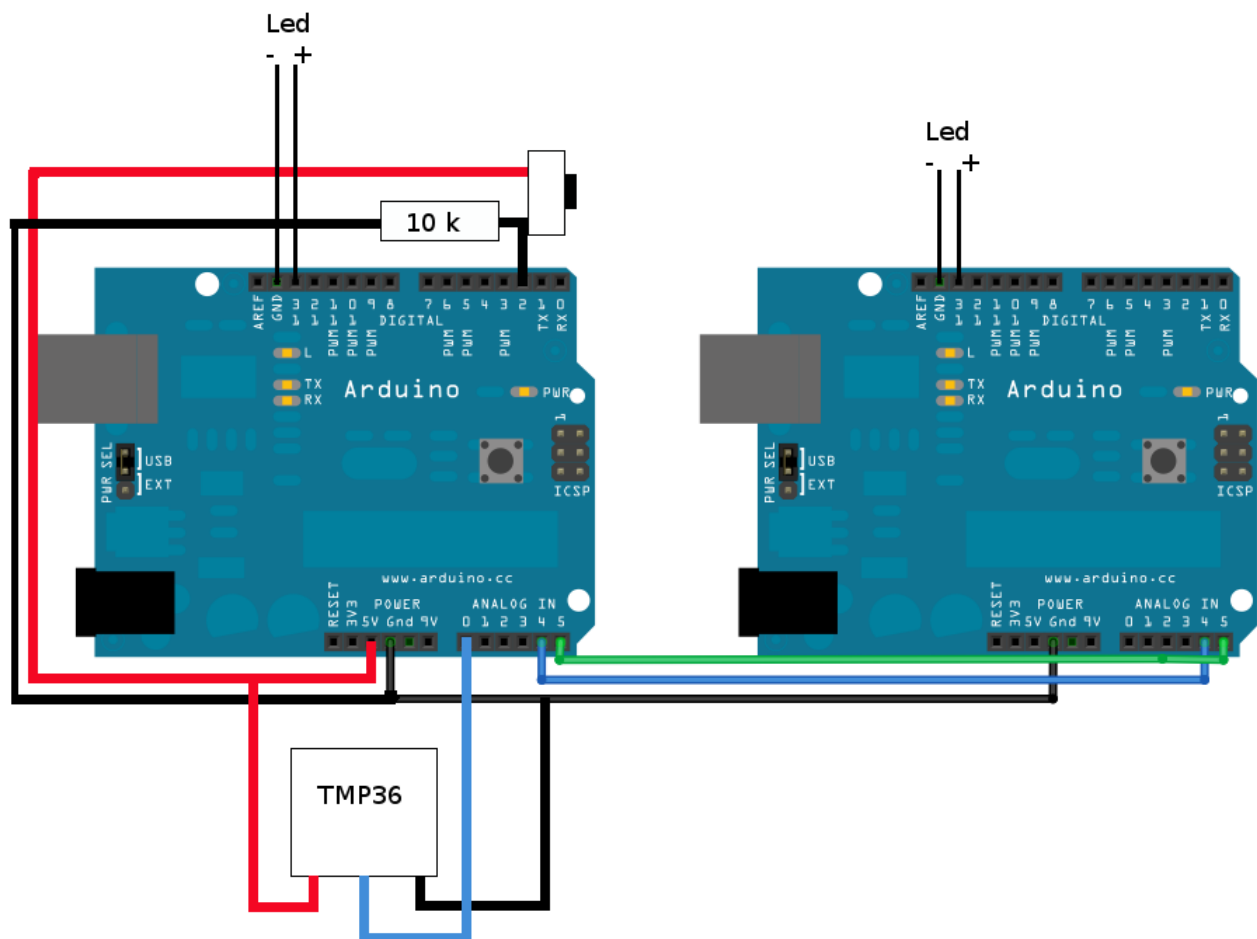
Harjoituksen tavoitteena tutustua kahden Arduinon väliseen väylään.

### Tavoitteet:

- Kytke Arduinot yhteen. Toinen on master (ohjaava) ja toinen Slave (ohjattava).
- Master antaa käskyjä ja tietoja Slave Arduinolle.
- Lisäksi tarvitaan virtaa kummallekin arduinolle.

Arduinot kytetään kuvan 1 osoittamalla tavalla. Maadoitukset kytetään yhteen (GROUND=GND) maatason balansoimiseksi, ja varsinainen dataliikenne kulkee ANALOG IN porttien 4 ja 5 kautta.

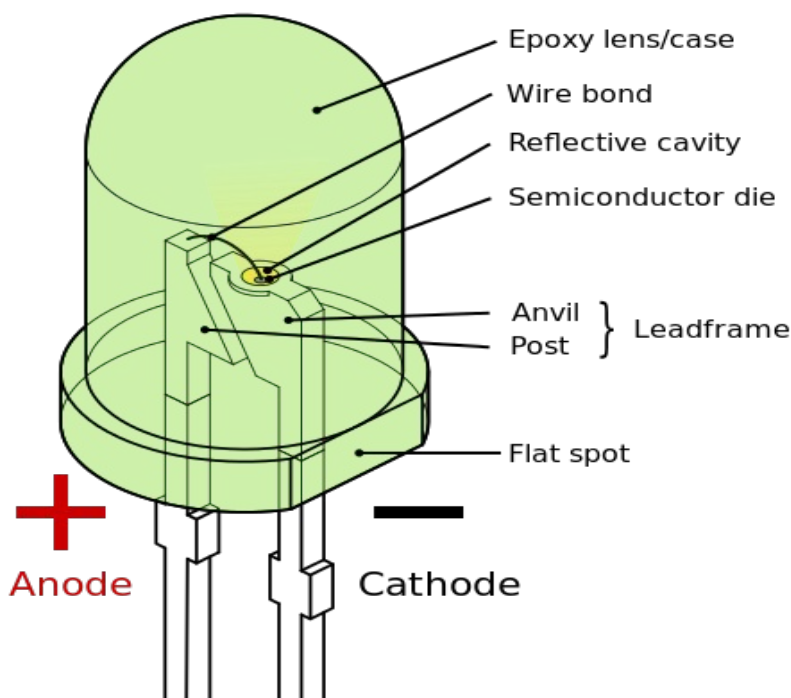
(Kyseinen väylä on suoraan rinnastettavissa autoissa käytettävään CAN parikaapeliin.)



Kuva 1. Arduinon väyläkytkentä.

MASTER: master arduinoon kytketään TMP36 lämpötila-anturi. Se tarvitsee virtaa, ja sitä saadaan Arduinon 5V portista/navasta. Lisäksi anturi tarvitsee maadoituksen (GROUND=GND) portista. Lämpötila-anturi on tarkkuusmittari, jossa ulostuleva jännite vaihtuu 10 mV astetta kohti. Ulostuleva signaali on keskimmäinen 'jalka' anturissa, ja se kytketään arduinon ANALOG IN 0-porttiin kuvan 1 osoittamalla tavalla.

Lisäksi kytketään sekä masteriin että slaveen LEDit (vinkkiä kytkentään voi ottaa kuvasta 2). Arduinon kytkentä kestää suoran liittämisen portteihin, vaikkakin oikeaoppinen tapa olisi rajoittaa virtaa pienien vastuksien avulla. Masteriin kytketään lisäksi painokytkin joka nostaa sisääntuloportin (DIGITAL 2) tilaan "high". Muussa tapauksessa portti kytketään nolaksi maadoituksen avulla. Oikosulkua vastaan maadoituksen ja + väliin kytketään 10 kohm vastus.



**Kuva 2.** ledin positiivinen ja negatiivinen napa. Negatiivisessa navassa on nuolen kärki ylhäällä (osoittaen elektronien suunnan). Vaihtoehtoisesti katkaisemattomista jaloista voidaan päätellä, että pidempi jalka on (+).

## Oheismateriaali

Virallinen Arduino esimerkki

<http://arduino.cc/en/Tutorial/MasterWriter>

Harjoitustehtävät 1, 2 ja 3, josta saadaan nappulan käyttö, lämpötila-anturi ja lcd näyttö ohjeet.

## Master

```
#include <Wire.h>

String arvo="0";

const int buttonPin = 2;

int buttonState = 0;

int sensorPin = 0;//lampotilaa varten

void setup()
{
  pinMode(buttonPin, INPUT);
  Wire.begin(); // join i2c bus (address optional for master)
}

byte x = 0;

void loop()
{
  int reading = analogRead(sensorPin);

  float voltage = reading * 5.0;

  voltage /= 1024.0;

  float temperatureC = (voltage - 0.5) * 100 ;

  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed.
  // if it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
```

```
x=101;

}

else {

  //x=100;

  x=(int)(temperatureC);

}

Wire.beginTransmission(4); // transmit to device #4

Wire.write("x is ");      // sends five bytes

Wire.write(x);

Wire.endTransmission(); // stop transmitting

delay(500);

}
```

## SLAVE

```
#include <Wire.h>

int led = 13;

void setup()

{

  pinMode(led, OUTPUT);

  Wire.begin(4);          // join i2c bus with address #4

  Wire.onReceive(receiveEvent); // register event

  Serial.begin(9600);      // start serial for output

}

void loop()

{

}
```

```
// function that executes whenever data is received from master
// this function is registered as an event, see setup()
void receiveEvent(int howMany)
{
  while(1 < Wire.available()) // loop through all but the last
  {
    char c = Wire.read(); // receive byte as a character
    Serial.print(c);      // print the character
  }
  int x = Wire.read(); // receive byte as an integer
  Serial.println(x);    // print the integer
  if(x==101)
    digitalWrite(led, HIGH);
  else
    digitalWrite(led, LOW);
}
```