



DR



P

VEDEN LAATUMITTARI

OHJELMOINNIN SOVELLUSPROJEKTI K2023  
OAMK INFORMAATIOTEKNOLIGIAN OSASTO

MINNA LEPPÄNEN  
ANNA REHULA  
JUHA-MATTI HUHTA  
TVT23KMO

- Ideaa pallotellessa nousi esille, että haluttiin tehdä jotain käyttökelpoista.
- Minnalla on useita akvaarioita. Tästä heräsi ajatus vedenlaadun mittaamisesta.
- Projektin tarkoituksena oli valmistaa vedenlaadun mittari kolmella eri anturilla.

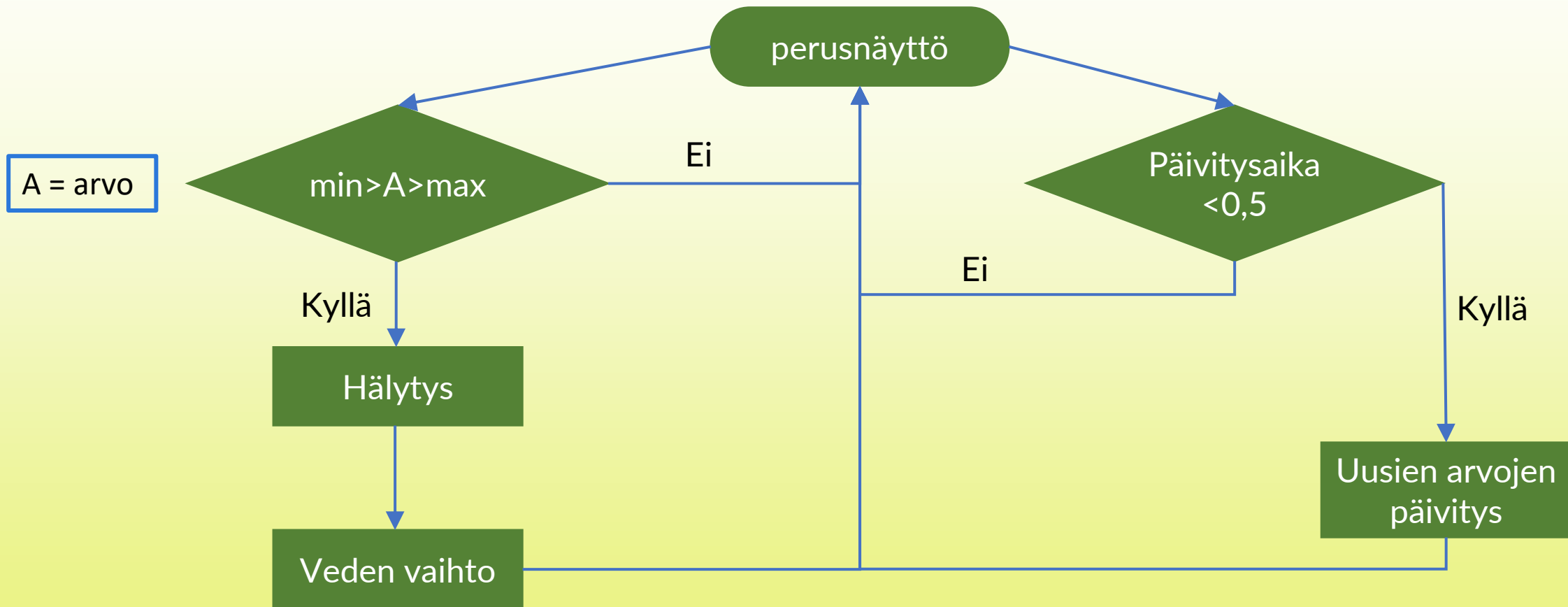




- Dropissa on LCD-näyttö, anturit, rotary enkooderi, LED ja summeri sekä virtalähde.
- Näytölle tulostetaan antureiden antamat arvot.
- Rotary enkooderilla päästään näytölle tulostuvaan valikkoon, jossa voidaan asettaa raja-arvot mitatuille suureille.
- Dropissa oli alun perin tarkoitus olla lämpötila-, TDS- ja pH-anturit.
  - pH vaihtui kuitenkin ultraääni anturiksi mittaamaan veden pinnan korkeutta





# Näytön päivitys ja hälytyksien toiminta




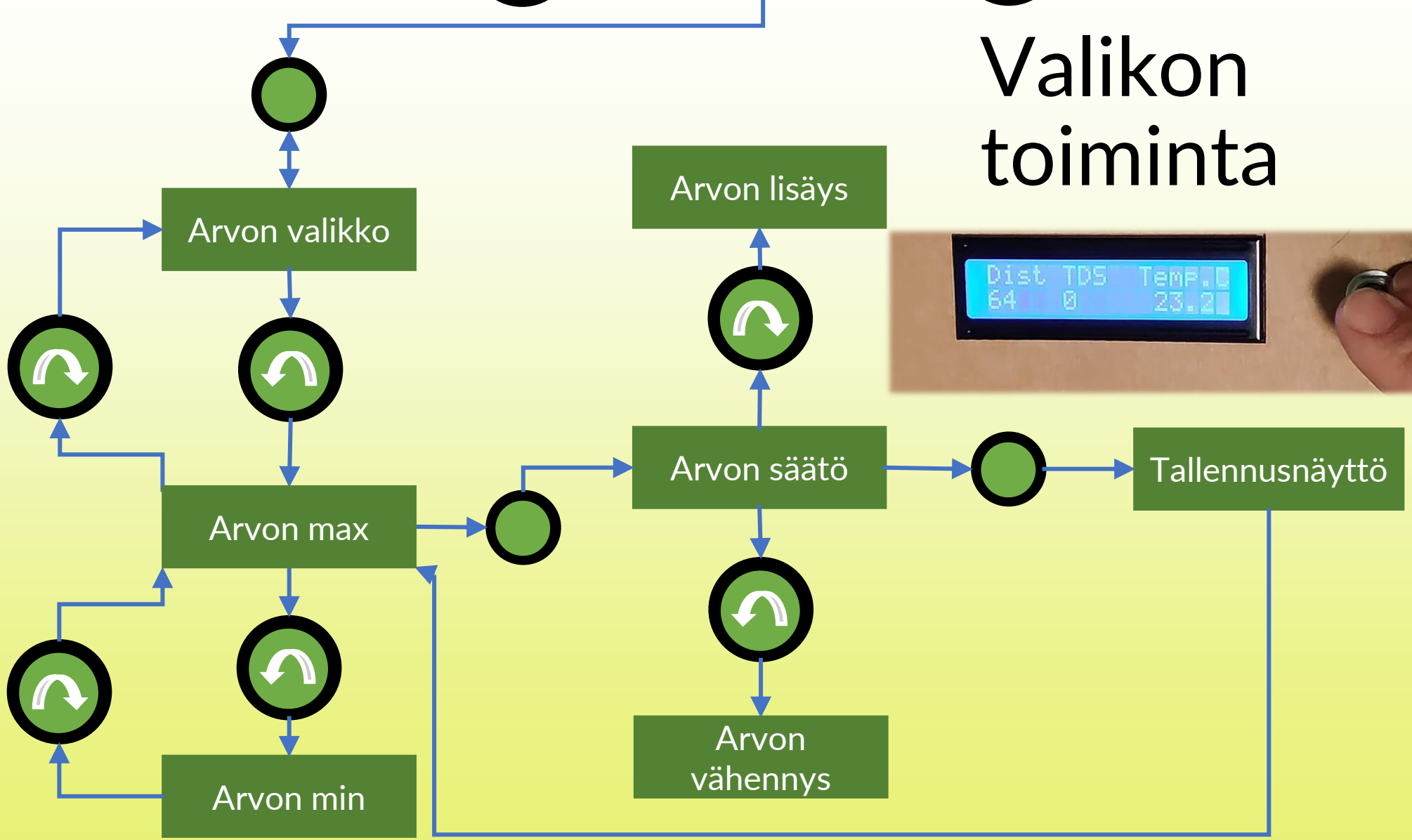


# Valikon toiminta

 Nappi

 Pyöräytys vastapäivään

 Pyöräytys myötäpäivään



# Koodi 1

## Aloitusnäyttö

```
//näytön tulostus|
lcd_1.clear();//tyhjentää näytön
lcd_1.noBlink();//asettaa kursorin vilkkumattomaksi
//tulostetaan 1. rivi
lcd_1.print("Dist TDS  Temp.C"); //kirjoitetaan ekasta merkitä lähtien näytölle.
lcd_1.setCursor(0, 1); //asetetaan kursori näytöllä toiselle riville, jotta voidaan tulostaa 2. rivi
//tulostetaan 2. rivi
lcd_1.print(dist);//tulostetaan dist arvo
lcd_1.setCursor(5, 1); //asetetaan kursori näytöllä TDS:n sijaintiin
lcd_1.print(TDS);//tulostetaan TDS:n arvo
lcd_1.setCursor(11, 1); //asetetaan kursori näytöllä Lämpötilan sijaintiin
lcd_1.print(temp,1);//tulostetaan lämpötilan arvo

//Hälyytykset:
if (distMin>dist||dist>distMax){
lcd_1.setCursor(4,1); //asetetaan kursori näytöllä dist-arvon perään
lcd_1.print("!");//tulostetaan huutomerkki dist-arvon perään
}

if (TDSMin>TDS||TDS>TDSMax){
lcd_1.setCursor(8,1); //asetetaan kursori näytöllä TDS-arvon perään
lcd_1.print("!");//tulostetaan huutomerkki TDS-arvon perään
}

if (tempMin>temp||temp>tempMax){
lcd_1.setCursor(15,1); //asetetaan kursori näytöllä temp-arvon perään
lcd_1.print("!");//tulostetaan huutomerkki temp-arvon perään
}
```

# Koodi 2

## Aloitusnäytön päivitys

```
//päivitysviiveen muuttujat
unsigned long previousMillis = 0; //alustetaan kokonaisluku muuttuja previousMillis
const long interval = 500; // alustetaan vakio interval 500ms, eli 5 sekuntia

unsigned long currentMillis = millis(); //alustetaan kokonaisluku muuttuja currentMillis ja asetetaan sen arvoksi juoksevat millisekunnit.

//Luetaan anturit ja tulostetaan ne näytölle, jos viive-aika on kulunut ja valikonKohta=0
if (currentMillis - previousMillis >= interval)
{
    previousMillis = currentMillis;
    v0(); //päivitetään näyttö
}
```

# Koodi 3

## Enkooderin painonappi

```
//encoderin nappi
int switchPin = 2;
int switchState = HIGH;
uint8_t btn_prev;

//encoderin koodia varten:
  pinMode (switchPin, INPUT_PULLUP);
  btn_prev = digitalRead(switchPin);

uint8_t btn = digitalRead(switchPin); // Luetaan switchPin, onko nappi painettu vai ei

btn_prev = digitalRead(switchPin); //jos nappia ei ole painettu asetetaan btn_prev arvoksi napin nykyinen arvo ja pyöritään loopissa
```

// Asetetaan enkooderin nappi pinniin 2 (enkooderissa SW)  
// napin arvo aluksi high, kun painetaan -> low  
// luodaan etumerkitön kokonaislukumuuttuja, johon tallennetaan napin edellinen arvo

// mahdollistaa switchPinin käyttämään Arduinon omaa pullup resistoria



# Koodi 4

## Enkooderin pyöritys

```
int dirOfRot=0; //luodaan kokonaisluku muuttuja dirOfRot, johon tallennetaan suunta setup-aliohjelmassa

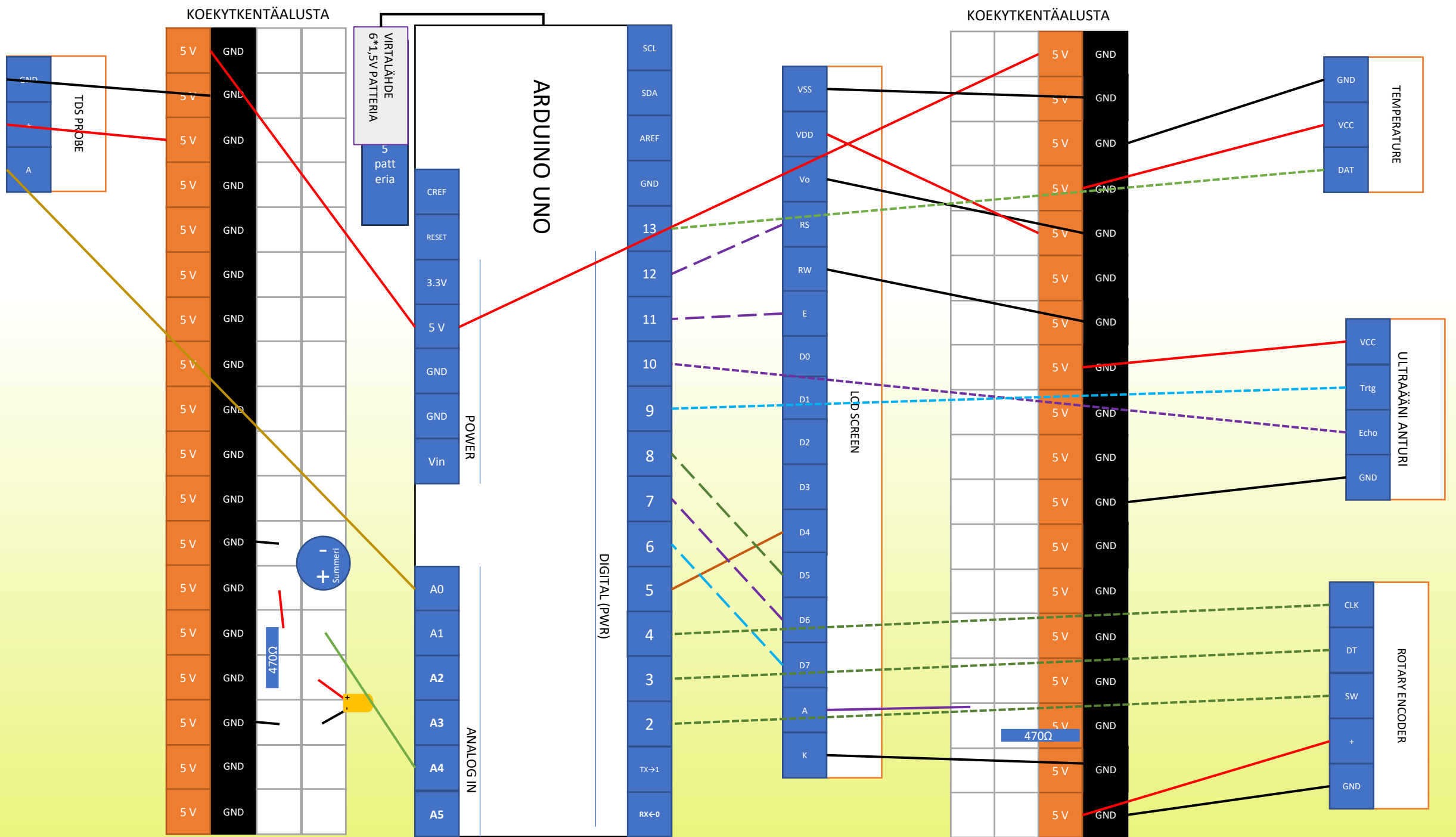
//encoderin pyörintä:
int pinA = 4; // Asetetaan enkooderin toinen arvo pinniin 4 ja sen nimeksi pinA (encoderissa CLK)
int pinB = 3; // Asetetaan enkooderin toinen arvo pinniin 3 ja sen nimeksi pinB (encoderissa DT)
int pinAstateCurrent = LOW; // Luodaan integer muuttujat pyörityksen arvoille
int pinAstateLast = pinAstateCurrent; // Luodaan integer muuttujat pyörityksen arvoille

void update() {
    // PYÖRITYSSUUNTA
    pinAstateLast = pinAstateCurrent; // Store the latest read value in the current state variable
    pinAstateCurrent = digitalRead(pinA); // Luetaan pinA ja tallenta se currentStatena
    // Jos oli liikettä, halutaan olla varmoja, että se teki yhden stepin muutoksen
    if ((pinAstateLast == LOW) && (pinAstateCurrent == HIGH))
    {
        if (digitalRead(pinB) == HIGH) { // If Pin B is HIGH
            dirOfRot=2;
        }
        else
        {
            dirOfRot=1;
        }
    }
    else{
        pinAstateLast = pinAstateCurrent; // Store the latest read value in the current state variable
        dirOfRot=0;
    }
}
```

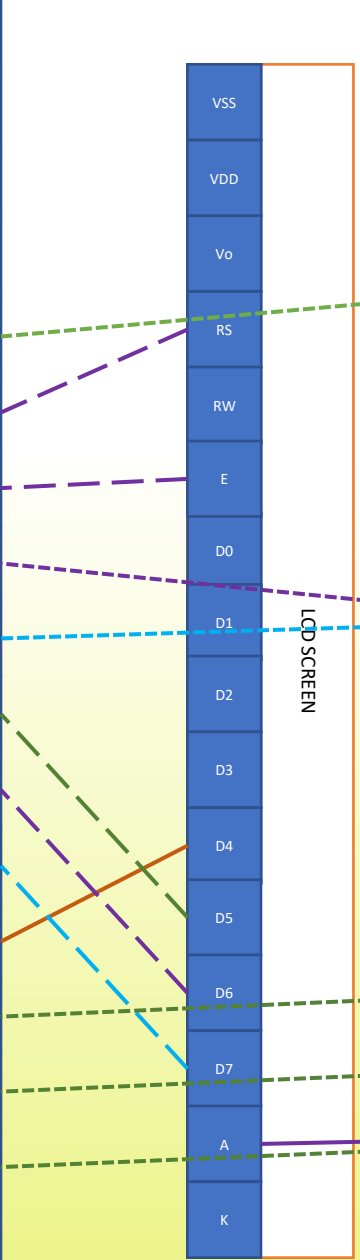
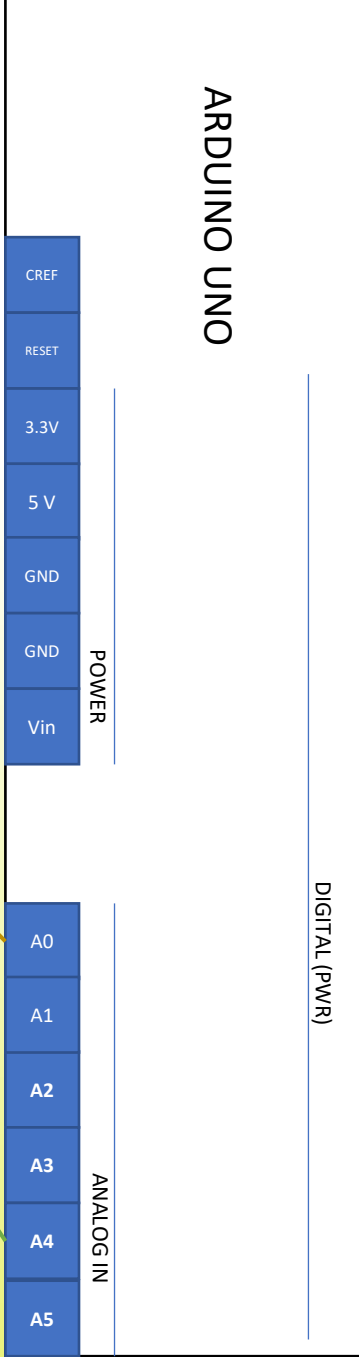
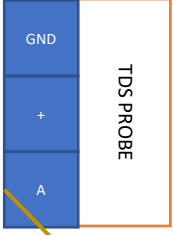
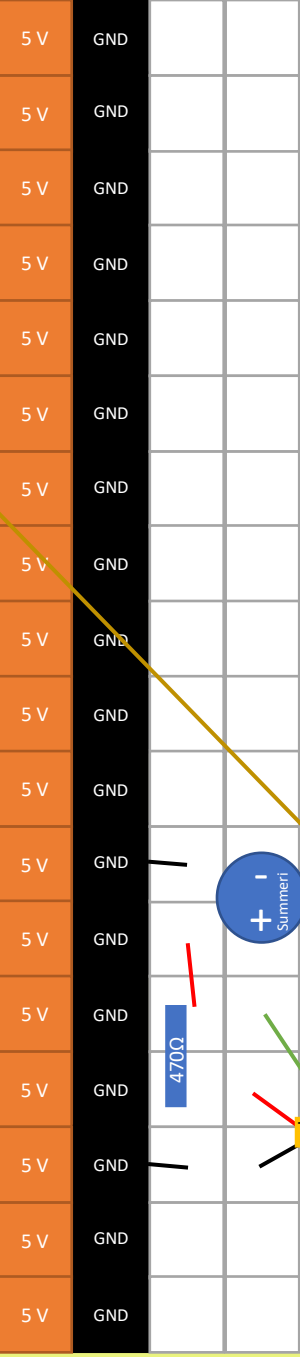
# Kytkenäkaavio

- Kahdella seuraavalla dialla nähdään kytkennät

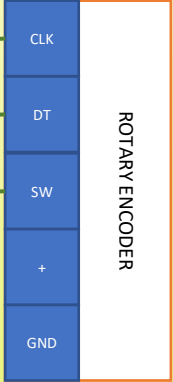
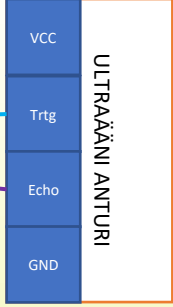
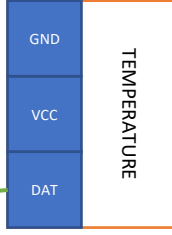
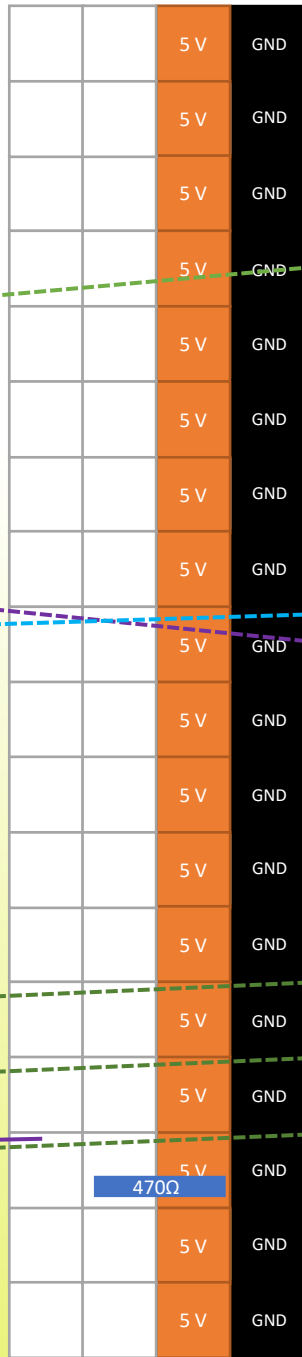




# KOEKYTKENTÄALUSTA

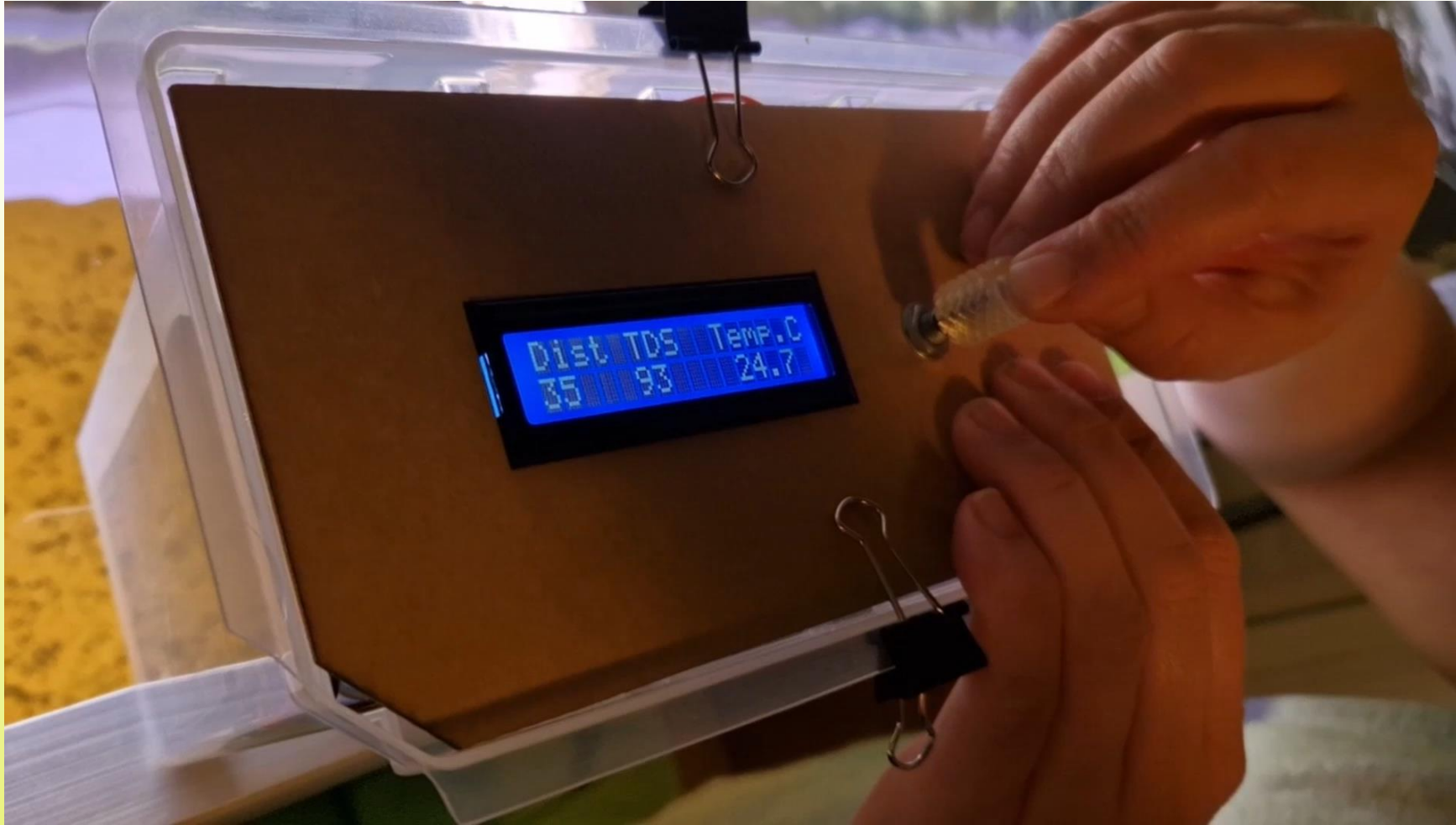


# KOEKYTKENTÄALUSTA





# Demonstrointi



# Projektin onnistumiset

- Onnistuimme valmistamaan toimivan, käytännöllisen laitteen, josta on ollut jo apua isojen akvaarioiden veden vaihdossa.
- Onnistumisia koettiin myös enkooderin toiminnan sulauttamisessa valikkoon.
- Sekä antureiden testaamisessa. pH-anturi osattiin havaita toimimattomaksi.



# Miten kehittäisimme eteenpäin?

- Jos projektia jatko kehiteltäisiin myyntiin, seuraavia asioita kannattaisi kehittää:
  - Virtanappi
  - Kotelon jatkokehitys
  - Kiinnittyminen akvaarion reunalle
  - Hälytyksen kuittaus
  - Tallennusnäytön oheen animaatiota
  - Säätoivastukset näytön kirkkaudelle ja kontrastille





# Mitä tekisimme toisin?

- Koodi on pitkä. Näin jälkikäteen ajatellen sitä voisi lyhentää.
- Jos ennen projektin aloitusta Mikrokontrollerin rakenne ja toimintaperiaate kurssilla olisi käyty LCD-näyttö ja tilakone, koodi olisi todennäköisesti lyhyempi.





# Risut ja ruusut

- Ryhmästä oli mielekästä, että projektin aihe päätettiin itse. Näin pystyimme valmistamaan hyödyllisen, käyttöön tulevan laitteen.
- Alustava viestintä ennen kurssia mahdollisti meidän projektin valmistumisen ajallaan.



# Kysymyksiä?

Vapaata pulputusta

