

Harjoitus 3

HUOM! Tutustu ensin Harjoitustehtävien palauttaminen -dokumenttiin ja -videoon, niin säästyt monelta harmilta! Dokumentin löydät Harjoitukset (3 op) -osion ohjeistustekstistä Palautuslaatikkojen yläpuolelta yhdessä mallivastauspohjan kanssa.

Ohjelmointitehtävät

Harjoitukset 2-5 ovat ohjelmointitehtäviä. Tehtävät vaikeutuvat loppua kohti ja niissä käsitellään sellaisia asioita, joita tarvitaan koulutusohjelman myöhemmissä vaiheissa. Tehtävät on siis laadittu yhteistyössä tulevien kurssien opettajien kanssa.

Myös ohjelmointiharkat palautetaan kukin omaan palautuslaatikkoonsa. **HUOM!** Lue tämä ohje tehtävien palauttamistaarten ennen kuin alat niitä tekemään! Näin säästyt monelta harmilta.

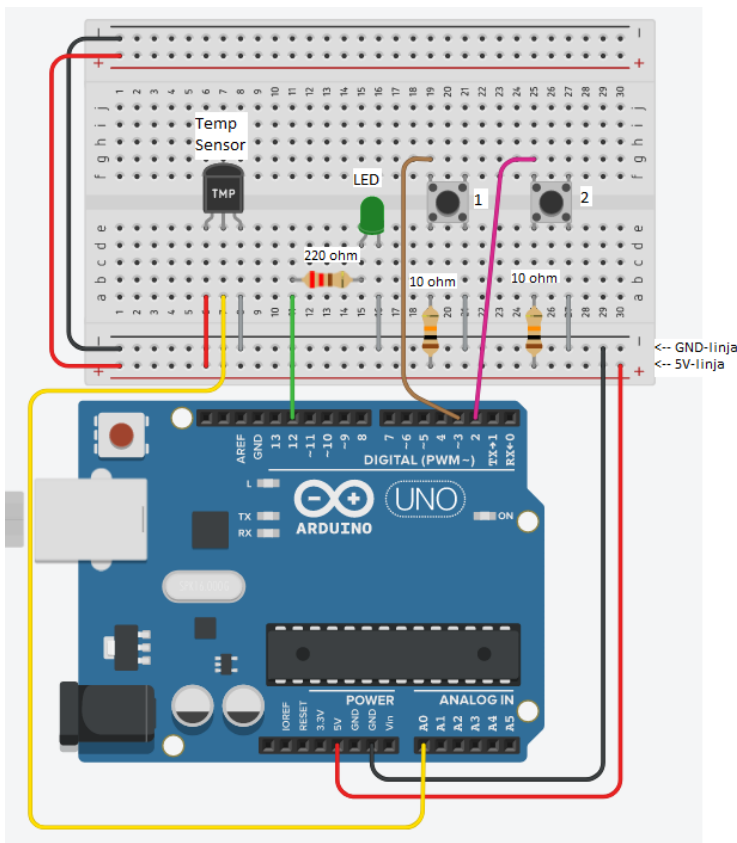
Tästä löydät Mallivastauspohja.

Palautuslaatikoissa on päällä nk. Lähetä arvioitavaksi -toiminto. Voit siis säilyttää halutessasi keskeneräistä vastausta palautuslaatikossa. Muista painaa lähetä arvioitavaksi -nappia kun työ on valmis.

Tähän harjoitukseen kuuluu kolme tehtävää, jotka oikeuttavat eri arvosanoihin. Ensimmäisen tehtävän tekemällä saavuttaa arvosanan 1. Ensimmäisen ja toisen tehtävän tekemällä saavuttaa arvosanan 3. Kaikki kolme tehtävää tekemällä saavuttaa arvosanan 5. Tee tehtäviä oman osaamisesi ja ajankäyttömahdollisuuksiesi mukaisesti. Kurssin arvosana muodostuu tehtävien arvosanojen keskiarvosta. Hyväksytty kurssisuoritus edellyttää arvosanan 1 saavuttamista kaikista harjoitusosioista.

Tutustumiskurssilaiset: Kurssin arvosana ei vaikuta opiskelijavalintaan (valintaa varten kurssi tulee olla kuitenkin hyväksytysti suoritettu).

Harjoitustehtävissä käytetään alla olevan kuvan mukaista kytkentää. Tehtävissä harjoitellaan painonappien ja keskeytysrutiinien käyttöä.



Johdatus sulautettuihin järjestelmiin

Tehtävä 1 (ARVOSANA 1): Ledien sytyttäminen painonapilla

Tarvitset seuraavia komponentteja tehtävän tekemisessä:

- LED ja etuvastus (220 Ω)
- Painonappi 1 ja ylös-/alasvetovastus (10 k Ω) (pin 3)
- Painonappi 2 ja ylös-/alasvetovastus (10 k Ω) (pin 2)
- Temp sensor (TMP36) (ks. kytkentä esimerkiksi Arduino-paketin projektikirjasta)
- 5V:n jännitelinja (punainen) ja maalinjat (GND, musta)

Tee edellisen sivun kuvan mukainen kytkentä. Tässä tehtävässä et tarvitse vielä Temp-sensoria, etkä toista painonappia, joten ne voi jättää myös kytkemättä toistaiseksi. Samaa kytkentää käytetään kuitenkin myös seuraavassa tehtävässä, joten kytkentä kannattaa tallentaa simulaattorissa.

Huomioi Temp-sensorin liitinjärjestys (kts. esim Arduino-projektikirjasta ohjeita tai googleta TMP36 sensoria). Temp-sensori tulee laitettua helposti väärin päin.

Kytkimen yhteydessä käytetään 10 k Ω vastusta, joka varmistaa, että jännitetaso on varmasti 0 V tai 5 V (looginen 0 tai 1). Kun painonappia ei paineta, digitaalisen pinnin jännite on 0 volttia. Kun painonappia painetaan, digitaalisen pinnin jännite on +5 volttia. 10 kilo-ohmin vastuksen värikoodi on ruskea-musta-oranssi (tai ruskea-musta-musta-punainen). Apua napin kytkemiseen ja ohjelmointiin löytyy esimerkiksi Arduino-paketin projektikirjasta (Project 02) **HUOM!** Mieti käytätkö ylös- vai alasvetovastusta ja miten se vaikuttaa toimintaan?

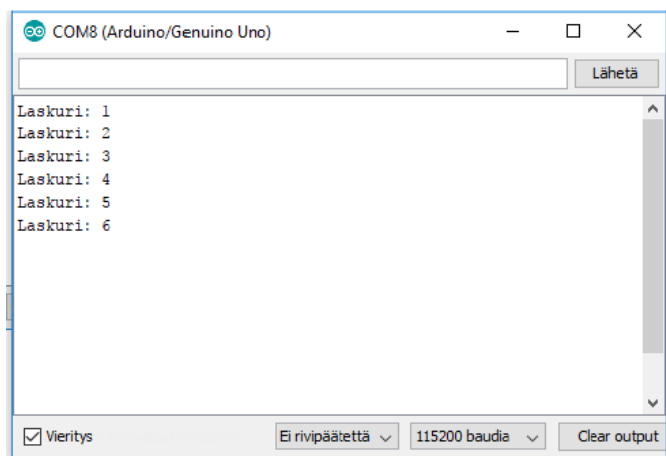
Tehtävä: Tee ohjelma, joka sytyttää valon, kun nappi on painettuna pohjaan. Muutoin valo on sammunut.

- vinkki: voit toteuttaa tämän esimerkiksi niin, että lue painonapin tila muuttujaan (arvo on high tai low) ja tarkastele (muuttujan arvoa tarkastelemalla), onko nappi painettu vai ei. Käytä tarkasteluun if-else rakennetta. Sytytä ja sammuta ledi digitalWrite komennolla napin tilan mukaisesti.

Tehtävä 2 (ARVOSANA 3): Painonappi ja keskeytysrutiini

Tehtävä: toteuta **keskeytysrutiini**, joka suoritetaan painonappia 1 painettaessa. Keskeytysrutiinissa kasvatetaan kokonaislukumuuttujaa ja tulostetaan sen arvo sarjaporttiin. Lisäksi keskeytysrutiinissa sytytetään pinnissä 12 oleva LED palamaan, jos se on sammuksissa ja sammutetaan jos se palaa.

Tulostusesimerkki: Nappia painettaessa laskurin arvo kasvaa ja Led on sammuksissa parillisilla arvoilla ja palaa parittomilla.



Tehtävä 3 (ARVOSANA 5): Analoginen lämpötila-anturi

Tehtävänä on lukea mikrokontrolleriin liitetyn analogisen TMP36 lämpötila-anturin arvo. Ohjelmoi kontrolleri niin, että nappia painettaessa tapahtuu **keskeytysrutiini** ja keskeytysaliohjelmassa lämpötila-anturilta luetaan 20 millisekunnin välein 50 ADC-arvoa ja niistä lasketaan keskiarvo. Tämä keskiarvo (avgADC) ja sitä vastaava lämpötila-arvo (LT) tulostetaan sarjaporttiin (mallitulostus alla). Lämpötila-arvon saat laskettua seuraavalla muunnoskaavalla:

$$LT = 0.7 * avgADC - 77.4$$

Joissakin järjestelmissä ja joillakin laitteistoilla tämä antaa kuitenkin virheellisiä arvoja. Silloin voit koittaa muunnoskaavaa

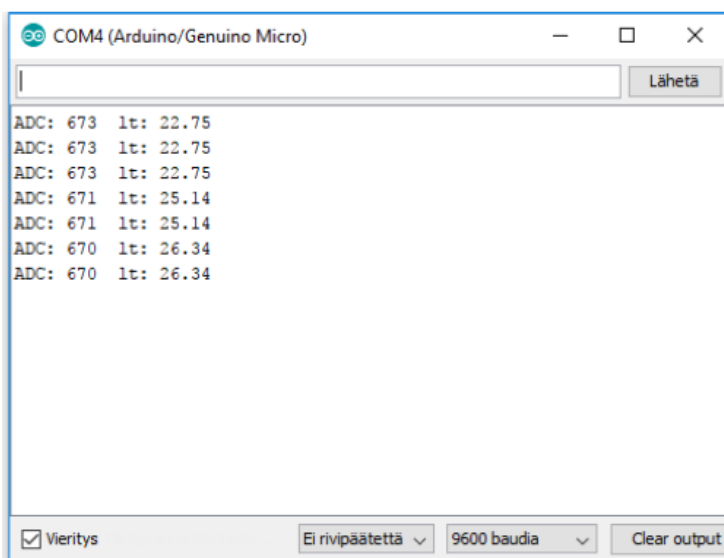
$$((5 * avgADC / 1024.0) - 0.5) * 100$$

Muunnoskaavan tarkalla toimivuudella ei ole työn hyväksyttävyyden kannalta merkitystä. Tärkeintä on, että saat työn muutoin toimimaan halutulla tavalla.

Vinkki: tarvitset muuttujan, jota kasvatat silmukassa 50 kertaa anturilta luetulla arvolla (tee siis for-silmukka, joka pyörähtää 50 kertaa, lukee kullakin pyörähdyksellä lämpötila-anturin arvon ja summaa arvon muuttujaan ja odottaa 20ms ennen seuraavaa pyörähdystä). Muuttujaan tallennettu summa jaetaan lopuksi 50:llä, jolloin saadaan mittausten keskiarvo, joka laitetaan muuttujaan avgADC. Lisäksi tarvitset vielä yhden muuttujan, johon talletat yllä annetulla kaavalla lasketun lämpötilan.

Huom: Tässä tehtävässä tehdään mittauksia keskeytysaliohjelmassa. Tämä ei ole ideaalitilanne, koska keskeytysaliohjelmassa ei tulisi viipyä kovin kauaa. Tätä ei tarvitse tässä tehtävässä huomioida, koska kyseessä on johdantokurssi, jolla ei kiinnitetä ihan kaikkiin ohjelmoinnin optimiteuteuksiin huomiota. Tärkeintä on, että opitaan keskeytyksen tekeminen ja hoksataan, että mittausta on hyvä tehdä keskiarvona. Hyvä kuitenkin mieltää, että optimiratkaisussa keskeytyksessä viivytettäisiin mahdollisimman vähän aikaa. Toki jos asia häiritsee, niin voi tehdä ohjelmoinnin oikeellisuuden näkökulmasta tyylikkäämmän ratkaisun, jossa keskeytysrutiinissa muutetaan esim. boolean tyyppistä lippumuuttujaa, jonka arvoa tarkastellaan sitten pääloopissa, jossa lipun perusteella käynnistetään mittaukset tai siirrytään aliohjelmaan. Mutta siis tässä harkassa mittaukset voi hyvin hoitaa tuolla keskeytysrutiinissa.

Tulostusesimerkki: Napin painalluksella monitoriin tulostuu 50 mitatun ADC:n keskiarvo ja siitä laskettu lämpötila. Kuvassa painonappia on painettu 7 kertaa. Välillä lämpötila on muuttunut ja välillä ei. Tehtävän hyväksymisen kannalta ei ole tärkeää, onko tulostus täsmälleen alla olevan näköinen.



Koskee jokaista tehtävää!

MUISTA KOMMENTOIDA KOODIIN KOODIRIVIT TAI KERRO SANALLISESTI JOKAISEN KOODILOHKON TOIMINTA

Kommentointiesimerkit:

```
int mittausarvot[10];    //varataan mittausarvoille 10 muistipaikkaa
int i = 0;               //mittausarvotaulukon indeksi
int keskiarvo, summa;    //alustetaan keskiarvo ja summa muuttujat
```

Muuttujat kannattaa nimeta selvillä nimillä. Lisäksi vielä kommentit

```
//funktiossa laskenta() nollataan ensin muuttujat keskiarvo ja summa. For silmukassa
//lasketaan mittausarvot[] taulun kaikki 10 arvoa yhteen.
//Silmukan jälkeen lasketaan keskiarvo ja tulostetaan se sarjaporttiin.
void laskenta()
```

Koodilohkojen alkuun (esim funktiot) kannattaa kirjoittaa koodin logiikkaa. Toki lohkon sisälläkin kannattaa kommentoida yksittäisiä rivejä tai tapahtumia

```
{
    keskiarvo = 0;
    summa = 0;
    for(int i=0; i<10; i++)
    {
        summa = summa + mittausarvot[i];
    }
    keskiarvo = summa/i;
    Serial.print("Keskiarvo on ");
    Serial.println(keskiarvo);
}
```