Sulautetut sovellukset ja projektityöt

Näyttöprojekti, valikko

Saari Marko

2018

Sisällys

[1 Kuvaus laitteesta 1](#_Toc532308690)

[2 Kuvaus elektroniikasta 2](#_Toc532308691)

[3 Kuvaus ohjelmasta 3](#_Toc532308692)

[4 Linkit, ohjeet ja inspiraationlähteet 5](#_Toc532308693)

[5 YTO Kestävän kehityksen edistäminen 6](#_Toc532308694)

[6 Projektikansio 7](#_Toc532308695)

# 1 Kuvaus laitteesta

Laite on kolmella painikkeella ohjattava valikko rakenne,

jossa on neljä digitaalista lähtöä D3-D6, ja neljä analogista sisääntuloa A0-A3.

Tässä projektissa suunnittelin valikkorakenteen,

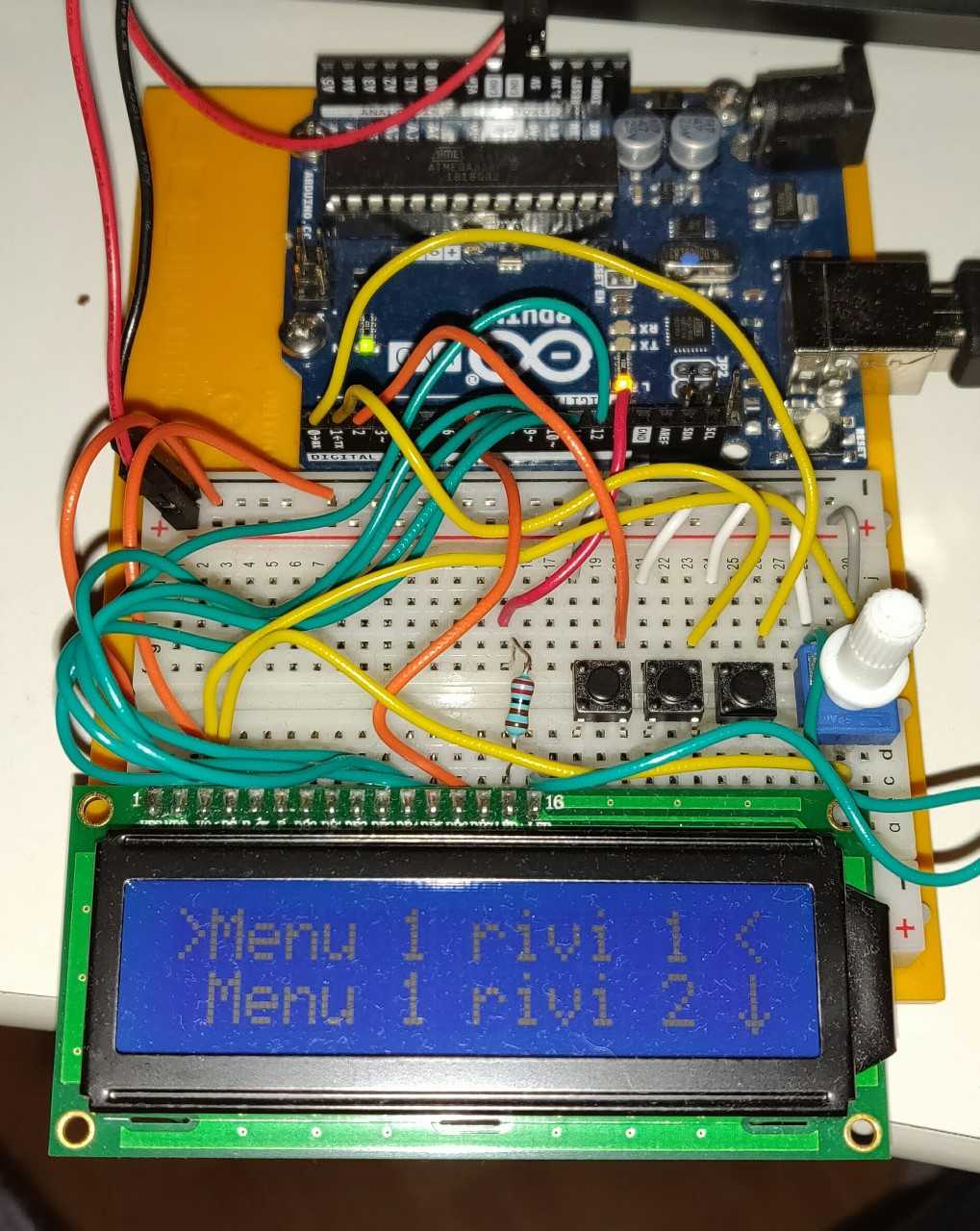
joka on laajennettavissa esimerkiksi ohjaamaan lähtöjä,

ottamaan analogisia mittauksia, tai sovellettavissa muihin projekteihin.

Esimerkki valikoissa tällä hetkellä taustavalon ohjaus, ja analogimittaukset pinneistä A0 – A3.

Projektia testatessa valikoissa olivat mm. valikosta käynnistettävä 4-ledin ritari ässä ja digitaalilähtöjen päälle/pois kytkentä.

Nykyisellään laitteessa on 1:ssä valikossa neljä riviä, joissa kullakin kaksi riviä, joissa taas kullakin yksi valikkorivi. Lisäksi jokaisella on lisätty tyhjä (NULL), ja sisempiin valikoihin paluu ensimmäiseen ja/tai paluu edelliseen valikkoon.



# 2 Kuvaus elektroniikasta

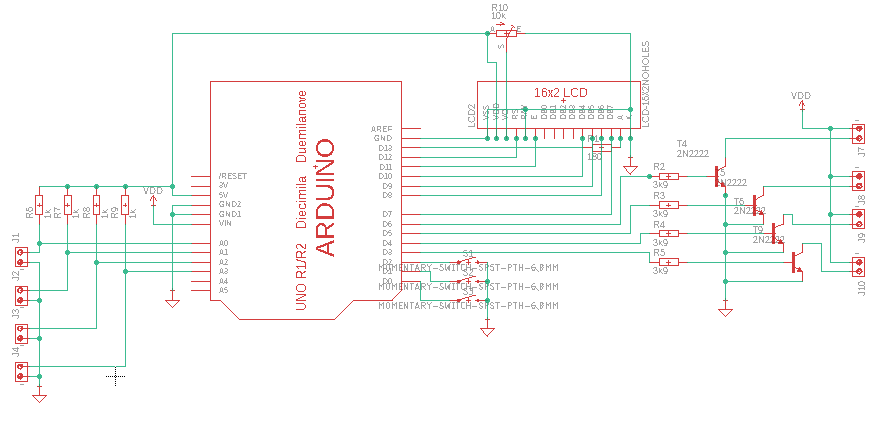
Laitteessa on käytetty Arduino Uno alustaa, 16x2 LCD näyttöä,

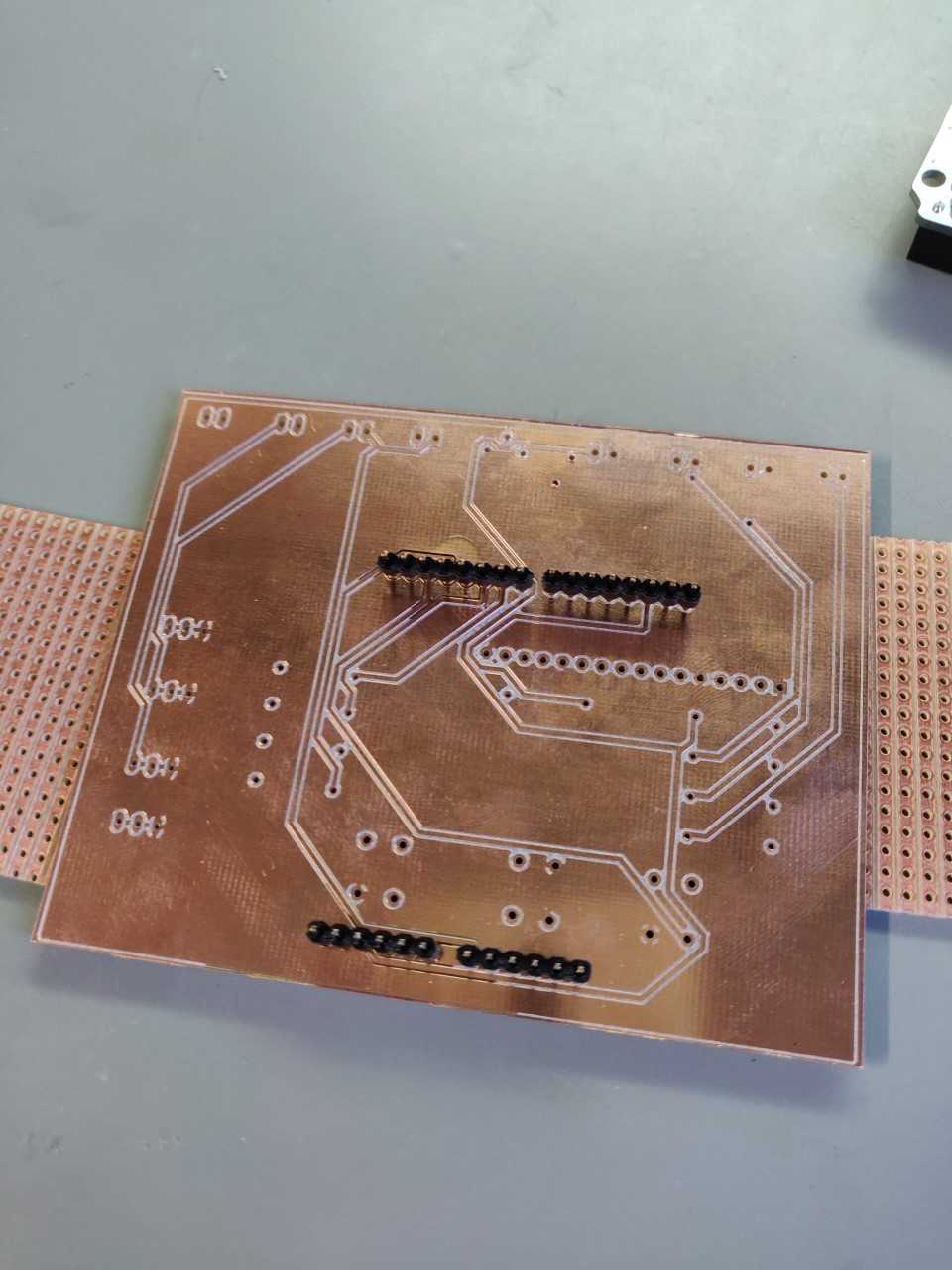
kolmea painiketta, 10K:n trimmeriä LCD:n kontrastin säätöä varten, 1K vastuksia analogituloille,

3K9 vastuksia ja BC547-NPN transistoreita digitaalilähdöille sekä

180 Ohmin vastusta LCD-näytön taustavaloa varten.

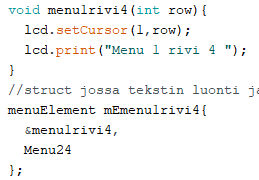
Laite alun perin Jani Lehdon suunnittelema,

ja ohjelma suunniteltu ja rakennettu hänen rakentamaa Arduino shieldiä käyttäen.

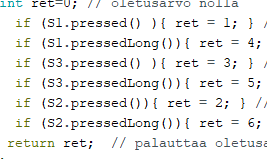


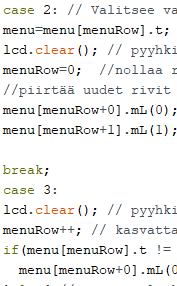
# 3 Kuvaus ohjelmasta

Ohjelman tarkoitus oli luoda helposti eri projekteihin muunneltavissa oleva valikkorakenne.



Ohjelma koostuu struktuureista, joilla luodaan teksti funktioista ja sisältävät linkin seuraavaan taulukkorakenteeseen.

Painikkeiden funktiossa buttons() on määritelty kukin painike palauttamaan erilainen arvo, jonka tarkoitus on antaa menuControl() funktioon arvo, jolla saadaan aikaan tapahtuma painiketta painattaessa, jonka jälkeen arvo palaa aina takaisin vakioarvoon.

Loopissa pyörivä menuControl() funktio vastaanottaa painikkeilta saatuja arvoja, ja palauttaa tapahtuman tämän mukaan.

Esimerkiksi painiketta S2 painattaessa buttons() funktio saa arvon kaksi, menuControl() funktio palauttaa arvon kaksi,

joka hakee valikkorakenteen valikkorivimuuttujan mukaan toisesta taulukosta, pyyhkii lcd-näytön,

asettaa valikkorivimuuttujan nolla-arvoon ja piirtää uudet rivit uudesta valikkorakenteesta.

Button-kirjasto lainattu Jani Lehdon tekemästä kirjasto esimerkistä.

Paljon tuli projektin aikana luettua eri aineistoja, ja paljon uutta opittua ohjelmistojen rakenteesta sekä tietoon paljon uusia erilaisia toteuttamistapoja. Suunnittelusta varmaankin juuri se, millä tavalla haluaa toteuttaa, oli hankalin.

# 4 Linkit, ohjeet ja inspiraationlähteet

Alkuun tutkin monia erilailla toteutettuja valikkoja ja ohjeita, ja alkuun toteutinkin projektin eritavalla. Kuitenkin jatkokäyttö olisi ollut hieman hankalampaa, mikäli valikkoa olisi haluttu käyttää useampiin projekteihin. Kuitenkin siitäkin tuli toimiva kahden rivin valikko. Nykyiseen muotoon valikko lähti käyntiin Janin opastuksella ja ehdotuksella käyttää ja tutustua struktuureihin.

Ohjeita ja esimerkkejä luettiin mm.

<https://www.hackster.io/>

<https://www.instructables.com/>

<http://playground.arduino.cc/>

<https://arduino.stackexchange.com/>

<https://github.com/>

Lisäksi kurssin aikana tuli aika paljon keskusteltua ja autettua puolin jos toisin muiden oppilaiden kanssa, joilta sai paljon hyviä erilaisia näkökulmia ja lähestymistapoja eri tilanteisiin kurssin edetessä.

Jos olisi ollut enemmän aikaa olisi ollut hauska vaikka liittää toisen oppilaan projekti tähän projektiin.

Jollain tapaa kieli oli ennestään tuttu tai siinä oli tuttuja piirteitä muista kielistä, ja ehkä siksi kurssi lähti nopeasti käyntiin.

Mielestäni funktioiden muodostus sujuu jo hyvin, paljon silti opittavaa. Hauskinta oli ehkä kokeilla mitä tapahtuu kun muuttaa jotain pientä asiaa ja huomaa kuinka paljon erilaiseksi se saattaa muuttaa koko ohjelman.

# 5 YTO Kestävän kehityksen edistäminen

Tietotekniikan vastuuhaasteet liittyvät esimerkiksi työntekijöiden oikeuksiin, työturvallisuuteen, ihmisoikeuksiin, suureen ympäristökuormitukseen ja veronmaksuun.

Raaka-aineet, komponentit ym. tulisi hankkia vastuullisilta tahoilta. Monesti komponenteissa käytetyt raaka-aineet ovat peräisin kaivoksilta maissa, joissa perusihmisoikeuksien turvaaminen on vaikeaa.

Myös suuri osa elektroniikkalaitteistamme valmistetaan tehtaissa, joiden työolot eivät täytä kansainvälisesti sovittuja vähimmäisstandardeja.

Suurimmat elektroniikkalaitteiden ympäristörasitukset ovat energiankulutus ja siitä aiheutuvat kasvihuonepäästöt, elektroniikkajäte sekä laitteissa käytetyt myrkylliset kemikaalit ja raskasmetallit.

Energiankulutusta sekä uusiokäytön tarvetta lisää laitteiden lyhyt käyttöikä.

Tämän laitteen voi uudelleen ohjelmoida moniin eri tarkoituksiin, siihen voi lisätä osia, joten elinkaari voi olla hyvinkin pitkä, mikäli vain komponentit kestävät.

Toisaalta, rikkoutuneen komponentin voi myös vaihtaa, mikäli onnistuu paikantamaan vika. Kun viimein laite tulee matkansa päähän, se tulisi kierrättää asianmukaisesti. Kierrätyksessä tulisi huomioida myös raaka-aineiden tehokas uudelleenkäyttö.

# 6 Projektikansio

<https://github.com/ritsarassi/menu>

* Komponenttiluettelo BOM.txt
* Visio valikkokartta menu.vsdx
* Piirilevy suunnitelmat shield-kansio
* Kuva kytkentäkaaviosta shield.png
* Ohjelmiston lähdekoodit
* Sulautetut sovellukset ja projektityöt lehtiö, Word