



**LAHDEN AMMATTIKORKEAKOULU**  
*Lahti University of Applied Sciences*

# MUUNTAJANKÄÄMINTÄKONE

LAHDEN  
AMMATTIKORKEAKOULU  
Tekniikan ala  
Tietotekniikka  
Tietokone-elektronikka  
Opinnäytetyö  
Syksy 2013  
Olli Lehto

Lahden ammattikorkeakoulu  
Tietotekniikan koulutusohjelma

LEHTO, OLLI:

Muuntajankäämintäkone

Tietokone-elektroniikan opinnäytetyö, 30 sivua, 6 liitesivua

Syksy 2013

TIIVISTELMÄ

---

Tämän opinnäytetyön tavoitteena oli suunnitella ja toteuttaa laite, jolla kyetään kelaamaan muuntajarunkoon käyttäjän haluama kierrosmäärä käämilankaa. Laite on ensisijaisesti suunnattu käytettäväksi opetustarkoituksiin ja on yksittäiskappale, joten se on pidetty mahdollisimman kustannustehokkaana valmistaa ja siinä on käytetty vain yleisimmin saatavia osia laitteen korjattavuuden maksimoimiseksi.

Laite koostuu sähkömoottorista (DC), rungosta, elektroniikkalevystä, jolla on mBed-pikaprototyyppien tekemiseen suunnattu mikro-ohjain, logiikkatason FET, 12 V:n rele sekä näyttö indikaattorina ja numeronäppäimistö hallintalaitteena. Käyttöjännite tulee 11.1 V Li-ion-akusta, jota ladataan mukana olevalla laturilla. Mikro-ohjain on ohjelmoitu C++-kielellä käyttäen mbed.org-kääntäjää.

Työn tavoitteena oli tuottaa Lahden ammattikorkeakoulun tietokone-elektroniikan Tehoelektroniikan kurssin opetuksessa käytettävä laite, jolla voidaan käämiä helposti ja nopeasti prototyyppimuuntajia opetustarkoituksiin. Tämä tavoite saavutettiin.

Asiasanat: käämintä, muuntaja, näppäimistö, hd44780, E20/10/6

Lahti University of Applied Sciences  
Degree Programme in Information Technology

LEHTO, OLLI:

Transformer winding machine

Bachelor's Thesis in Computer Electronics, 30 pages, 6 pages of appendices

Autumn 2013

ABSTRACT

---

The objective of this thesis was to design and build a machine, which is capable of winding transformers with selected amount of turns. The machine is first and foremost meant to be used for educational purposes only and therefore it is constructed using only commonly available cheap parts which are easily replaceable in the future, should the need occur.

The machine consists of a self-made electronic board with an electronic motor (DC), frame, an mBed rapid prototyping microcontroller unit, a logic level FET, a relay and a HD44780 compatible LCD -display as an indicator, and a keypad for controlling the device. An 11.1V Li-ion battery works as the power source. The program code for the microcontroller is written in C++ language, using mbed.org compiler.

The purpose of the machine was to provide a device for teaching computer electronics power electronics, which could be used for rapid prototyping of transformers for educational purposes. This objective was met.

Key words: transformer, winding, keyboard, hd44780, E20/10/6

## SISÄLLYS

1	JOHDANTO	4
2	MUUNTAJAN KÄÄMINTÄKONE	5
2.1	Valmiina saatavat osat	5
2.2	Itse tehtävät osat	6
2.2.1	Alusta, runko ja ohjauspaneeli	6
2.2.2	Poljin ja työkalu	8
2.3	Muuntajat	10
3	ELEKTRONIIKKA	11
3.1	Jännitteiden regulointi	11
3.2	Mikro-ohjain	13
3.3	Näppäimistö	15
3.4	Näyttö	16
3.5	Rele ja MOSFET	18
3.6	Kierroksienlukuanturi	20
4	OHJELMAKOODI	20
4.1	Ohjelmointiympäristö	23
4.2	Ohjelmiston suunnittelu ja toteutus	24
5	KAUPALLISET RATKAISUT	26
6	TESTAUS JA MUUTOKSET	28
7	YHTEENVETO	32
	LÄHTEET	34
	LIITTEET	35

## 1 JOHDANTO

Tämä työ käsittelee elektronisesti ohjatun muuntajankäämintäkoneen valmistusta. Ajatus tämän työn valmistamiseen tuli tietokone-elektroniikan Tehoelektroniikan kurssilta, kurssilla olisi tarvittu konetta, jolla nopeasti ja yksinkertaisesti voidaan opetustarkoituksiin käämiä prototyypimuuntajiin haluttu määrä kierroksia.

Ensimmäisissä luvuissa käsitellään käämintäkoneen mekaanista toteutusta sekä osavalintoihin johtaneita perusteita ja sitä miksi käytetään E20/10/6-kelarunkoa. Mekaanisen osuuden jälkeen käsitellään itse laitteen elektroniikkaa ja ohjelmakoodia.

Työssä myös verrataan laitetta kaupallisiin versioihin sekä arvioidaan kustannuksia valmiisiin muuntajiin tuottajilta, jotka tekevät prototyyppejä asiakkaille.

## 2 MUUNTAJAN KÄÄMINTÄKONE

Työnantajan vaatimuslistassa oli määriteltynä, että laitteeseen pitää käyttäjän helposti voida antaa haluttujen kierrosten määrä, laitteen tulee osata laskea käämityt kierrokset ja pysähtyä kun käyttäjän antama lukumäärä on tullut täyteen. Nopeuden tulee olla säädettävissä samaan tapaan kuin ompelukoneessa eli polkimella, joka sijaitsee lattialla ja muuntajan on oltava käsin pyöritettävissä vielä sen jälkeen kun haluttu kierrosmäärä on täynnä. Osien tuli myös olla helposti saatavilla olevia yleisosa laitteiden huollettavuuden takia.

### 2.1 Valmiina saatavat osat

Työnantajan laitekuvauksessa oli määriteltynä, että laitteessa tulee soveltaa kaupallista porakonerunkoa, josta saadaan sopiva sähkömoottori, välityslaatikko sekä istukka. Porakoneeksi valikoitui POWERPLUS X0062LI porakone paketti hinnaltaan 45 euroa, jonka mukana tuli myös 11,1 V 1300 mAh Li-ion akku, sekä akulle laturi. Näin laitteen materiaalikustannukset saatiin pidettyä mahdollisimman alhaisena, kun kalleimmat osat saatiin yhdestä paketista. Pelkästään akun ja laturin osalta säästö oli huomattava esimerkiksi verrattuna kiinalaisiin harraste li-po akkuihin ja niille sopiviin latureihin, jotka olisivat tulleet maksamaan noin 50 euroa ilman postikuluja (HobbyKing 2013).

Työssä on pyritty hyödyntämään mahdollisimman paljon Lahden ammattikorkeakoulun elektroniikanlaboratorion komponenttivalikoimaa, jota löytyy hyllystä. Tästä syystä kaikki ratkaisut eivät ole täysin optimaalisia, mutta ovat kuitenkin toimivia.

## 2.2 Itse tehtävät osat

Laitteen prototyypiluonteen takia jouduin suunnittelemaan paljon mekaanisia osia itse. Pääasiassa kaikki käytetyt osat ovat alumiinia, ellei toisin ole mainittu. Suunnittelin itse työkalun, johon kelarunko tulee kiinni, alustan, johon kaikki osat kiinnittyvät, rungon, johon työkalu tulee kiinni, polkimen, sensorin kiinnityksen sekä LCD-näytön ja numeronäppäimistön kiinnityspaneelein.

### 2.2.1 Alusta, runko ja ohjauspaneeli

Laite vaatii alustan, johon kaikki osat tulevat kiinni tukevasti ja joka tekee laitteen käyttämisestä helpompaa. Materiaalina toimii 3 mm paksuudeltaan oleva alumiinilevy, johon tein sopivat kiinnitysraudat akulle, rungolle sekä hallintapaneelille. Alustassa on myös kiinnikepaikat polkimelle ja laturille. Mitat alustalle ovat 277 mm x 347 mm. Alustassa on kumiset tassut kooltaan 22 mm x 10 mm.

Runkona toimii porakoneen runko, josta on poistettu toiminnan kannalta epäoleelliset osat, kuten nopeudensäädin sekä akku. Porakoneen runko valikoitui käyttöön, koska se pitää moottorin, välityslaatikon sekä istukan tukevasti paikallaan ja korottaa ne riittäväälle korkeudelle alustasta, jotta työkalu mahtuu pyörimään ja käyttäjän kädet eivät tule liian lähelle pöytää. Runko on esiteltynä kuviossa 1.



KUVIO 1. Työkalun runko-osa

Ohjauspaneelin pohjana on 3 millinen alumiinilevy, johon on tehty reiät 4x4 numeronäppäimistöille ja 16x2 kokoiselle Hitachin HD44780 standardinmukaiselle näytölle. Nämä on kiinnitetty pehmeällä kuumaliimalla paneeliin ja ovat tarvittaessa irrotettavissa lievää voimaa käyttäen. Ohjauspaneeli on ruuvikiinnitteinen alustaan ja tarvittaessa irrotettavissa. Kuviossa 2 on esitelty ohjauspaneelin asettelu.



KUVIO 2. Ohjauspaneeli



### 2.2.2 Poljin ja työkalu

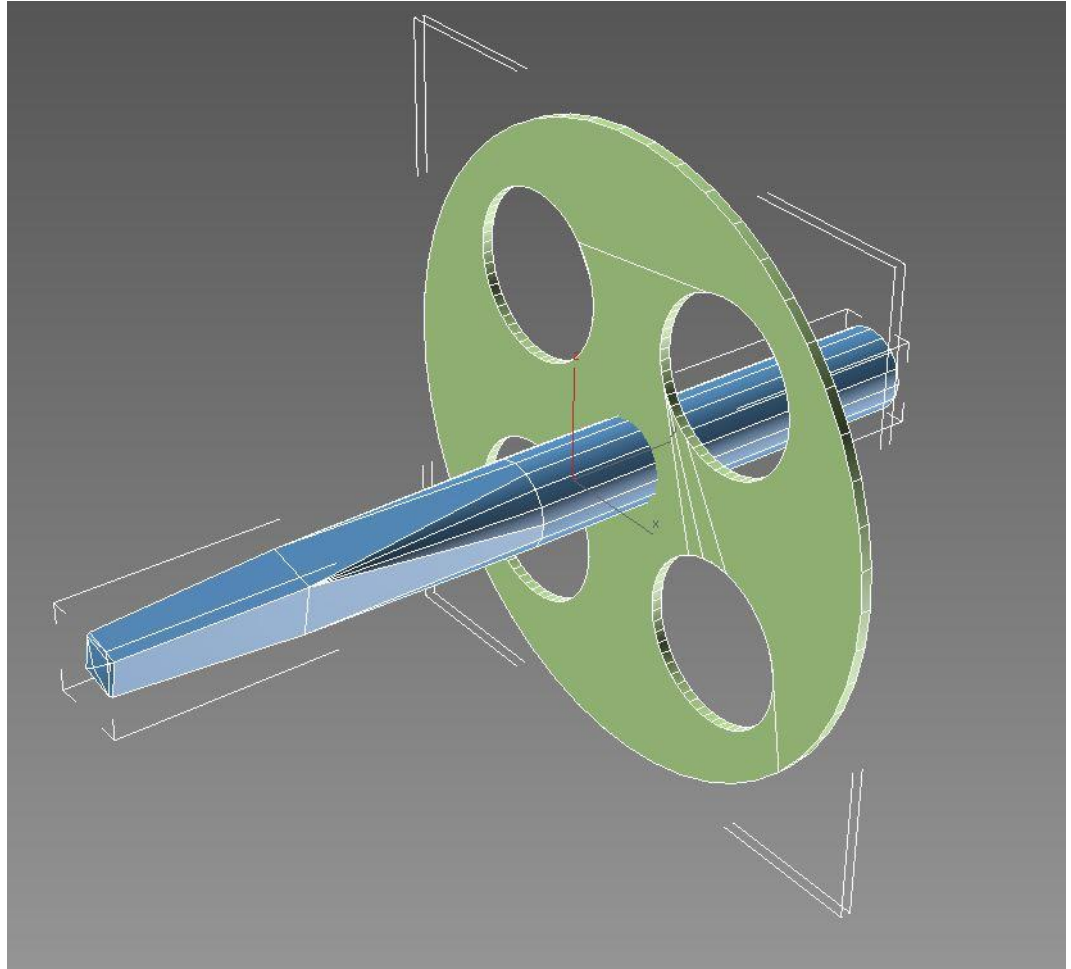
Poljindesign oli määritelty, ja sen pitää olla mahdollisimman lähellä toiminnaltaan ompelukoneen poljinta, jotta käyttäjä saa helposti säädettyä laitteen nopeutta.

Polkimen valmistin itse 2 mm teräspelistä, ja se vastaa leveydeltään ja pituudeltaan noin keskivertoihmisen kengän leveyttä ja pituutta. Kuviossa 3 on poljin esiteltyinä lopullisessa muodossaan. Polkimen sisälle on kiinnitetty porakoneen alkuperäinen nopeudensäätöyksikkö, jota käytetään laitteen pyörimisnopeuden säätelyyn.

Järjestelmän helpomman kuljetuksen ja säilytyksen kannalta poljin on nopeasti kytkettävällä liittimellä irrotettavissa alustasta. Lisäksi polkimen pinta on kumitettu Plasti Dip -merkkisellä kumipinnoitteella.



KUVIO 3. Käämintäkoneen nopeudensäätö poljin



KUVIO 4. Reikäpyörän 3D-malli

Työkalu, johon muuntajan kelarunko tulee kiinni, on suunniteltu pitämään E20/10/6-kelarunkoa paikallaan, ja työkaluun on integroitu laskurille tarvittava reikäpyörä, josta laite kykenee laskemaan ajettut kierrokset. Työkalun hahmotelman tein käyttäen Autodeskin 3ds Max 3D -mallinnusohjelmaa ja päädyin yksinkertaiseen ratkaisuun, jossa 8 mm teräsakselin päitä viistetään neliön muotoon, jolloin muuntajan kelarunko pysyy tukevasti paikallaan. Reikäpyörän hahmotelman mukaan se oli tarpeellista valmistaa 2 mm teräspellistä, jottei se taivu käytössä. (Kuvio 5)

### 2.3 Muuntajat

Muuntaja on passiivinen komponentti, jonka tehtävänä on muuntaa vaihtojännitteen tasoa jännitteestä toiseen. Aikaa ennen hakkurivirtalähteitä muuntajia käytettiin melko laajasti verkkojännitteen 230 VAC muuntoon pienoisjännitteille 0 – 50 VAC, jotta saatiin jännite laskettua sovelluksen tarpeita vastaavaksi tai muunnettua verkkojännite turvallisemmaksi harrastekäyttöön. Nykyaikana muuntajille on toinenkin käyttökohde flyback-hakkureissa, joissa niitä myös käytetään nykypäivänä.

Muuntajia saa eri kokoja ja eri kierrossuhteilla, mutta sopivan löytäminen saattaa joissakin tapauksissa olla kokeilemista tai sopivaa ei löydy valmiina vaan täytyy tehdä prototyypimuuntaja, jonka ominaisuuksien pohjalta tilataan joltain valmistajalta isompi erä muuntajia, jotka vastaavat prototyypin ominaisuuksia. Prototyypimuuntajien teettäminen on kallista ja projektin odotusaika nousee helposti päivillä tai viikoilla riippuen prototyypin valmistajan tilausaikojen jonotusajasta. Tästä syystä on hyvä olla muuntajankäämintäkone valmiina, jolla voidaan tehdä laskelmien mukaisia muuntajia ja testata niitä käyttökohteessa suhteellisen halvalla ja lähes olemattomalla odotusajalla.

Opinnäytetyön työkalu on suunniteltu E20/10/6-kelarungolle, joka on yleinen kelarunko, jota valmistaa ainakin Ferroxcube sekä Epcos. E20/10/6 on pieni muuntajakoko, jossa lukemat tarkoittavat leveyttä, korkeutta ja ytimen halkaisijaa. Kyseinen kelarunkokokoo on valittu siitä syystä, että tehoelektronikan opetuksessa on tarve käämiä flyback-hakkurivirtalähteille prototyypimuuntajia ja esitellä niitä opetuskäytössä. Kuviossa 2 on nähtävissä E20/10/6-kelarunko sovitettuna työkaluun.

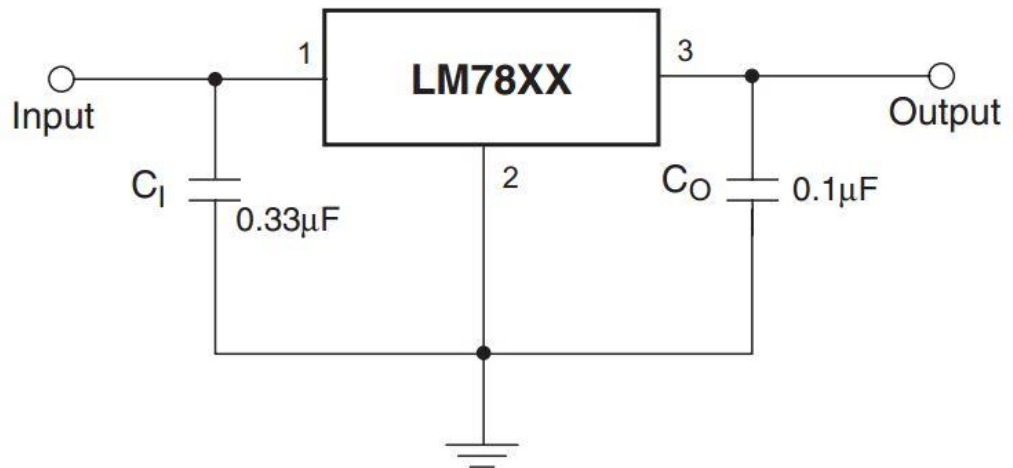
### 3 ELEKTRONIIKKA

Laitteen toteutuskuvauksessa määriteltiin laite tasavirralla toimivaksi, joten käytössä on vain tasavirtakomponentteja. Toteutuksessa on koitettu hakea yksinkertaisinta tapaa toteuttaa ohjauselektroniikka.

#### 3.1 Jännitteiden regulointi

Koska mikro-ohjaimet ja oheislaitteet tyypillisesti toimivat vain rajoitetulla jännitealueella, ei niille voida syöttää jännitettä suoraan akulta. Tästä syystä oli valittava jonkinlainen ratkaisu rajoittaa käyttöjännite toiminta-alueelle, joka tässä tapauksessa oli 4,5 V – 9,0 V. Yleinen käyttöjännite mBed-mikrokontrollereille on 5 V, joka on hyväksi havaittu jännite ja takaa laajan yhteensopivuuden myös oheislaitteiden kanssa, vaikkakin mBedin sisäinen regulointi tekee tästä vielä LPC1768-yhteensopivan 3,3 V jännitteen.

Päädyin valitsemaan Fairchild Semiconductorin LM7805CT lineaarisen jänniteregulaattorin, koska laitteen kuorma on pieni eikä pienellä virtahukalla ole isoa merkitystä. Regulaattorilla toteutettu jännitteensyöttö oli yksinkertainen ja halpa toteuttaa. Lisäksi jänniteregulaattorit sietävät erittäin hyvin erilaisia sisääntulojännitteitä. Akussa on kolme li-ion-kennoa, ja yhden kennon jännite vaihtelee 3,2 V:n ja 4,2 V:n välillä Näin kokonaisjännite vaihtelee 9,6 V:n ja 12,6 V:n välillä. 78xx-sarja kykenee pitämään jännitteen vakaana, jos sisääntulojännite on 3 voltin päässä reguloitavasta jännitteestä, tässä tapauksessa  $5\text{ V} + 3\text{ V} = 8\text{ V}$ . Valmistaja lupaa 2 V jännitehäviötä ja 7 V minimijännitettä, mutta olen havainnut, ettei tämä aina riitä.



KUVIO 5. Tyypillinen LM78xx-sarjan kytkentä jännitteen regulointia varten (Fairchild 2008)

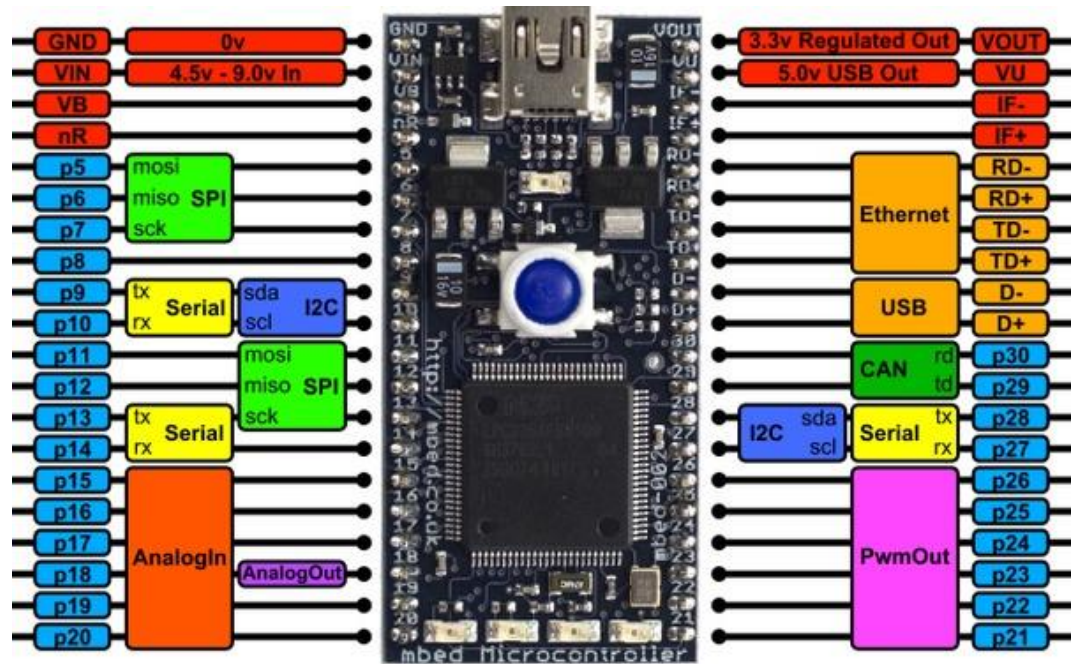
Kuviossa 5 on esitelty tyypillinen tapa toteuttaa jänniteregulaattorikytkentä 78xx-tyyppisellä lineaarisella jänniteregulaattorilla. Kyseisenlainen kytkentä on myös tehty elektroniikkalevyille, jossa ohjauslogiikka ja oheislaitteet sijaitsevat. Elektroniikkalevyllä on myös erillinen 7805-tyypin lineaariregulaattori, jonka tehtävänä on ainoastaan syöttää virtaa HD44780 yhteensopivalle näytölle.

## 3.2 Mikro-ohjain

Mikro-ohjain, toisinaan tunnettu nimellä mikrokontrolleri, on mikropiiri, joka on käytännössä pieni tietokone. Piiriin on integroitu prosessori, muistit, input/output-ohjelmit sekä usein joitain muita ohjelmitteita, kuten A/D- tai D/A-muuntimia. Mikro-ohjaimet ovat suosittuja tehtävissä, joissa ei tarvita suurikokoisempia tietokonelaitteistoja suurine muisteineen vaan ohjelmitava piiri, joka kykenee suorittamaan tarvittavat toimenpiteet reaaliajassa.

Mikro-ohjaimeksi valikoitui ARM Cortex M3-prosessoriin pohjautuva NXP:n valmistama LPC1768-piiriin pohjautuva mBed, joka on tarkoitettu nopeiden prototyyppilaitteiden valmistamiseen. Piiri oli myös laitekuvauksessa määritelty työssä käytettäväksi. Piiri on käyttökohteeseen tehoiltaan ja ominaisuuksiltaan erittäin ylimitoitettu, mutta kontrolleri ei tarvitse erillistä ohjelmointilaitetta, vaan sitä voi ohjelmoida USB:n kautta ilman erikoisajureita. Tästä syystä se soveltuu erinomaisesti tähän käyttökohteeseen, koska ohjelmointi on helppoa ja vaivatonta kotiympäristössäkin ja ei ole käyttöjärjestelmästä riippuvainen, koska ohjelmointiympäristö pyörii verkossa. mBed on myös tuttu ympäristö koulukäytöstä, joten sen käyttöönotto ei aiheuttanut erityistä päänvaivaa.

LPC1768-piirin kelloaajuus on 96 MHz, se sisältää 512 kB flash-muistia sekä 64 kB RAM-muistia. Kontrollerilta löytyy myös erilaisia liitäntöjä, esimerkiksi ethernet, USB, CAN, kaksi sarjaväylää ja PWM-lähdöt. (mbed.org 2013.)

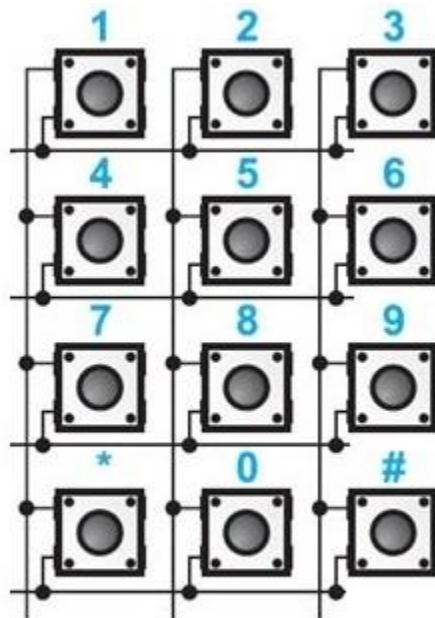


KUVIO 6. NXP mBed pinnijärjestys (mbed.org 2013)

mBed-kontrollerilla hoidetaan laitteen kaikki toiminnot, kuten kierroslaskuri, poljinyksikön virtojenkytkentä, hallintapaneelin ohjaus ja näytölle kirjoittaminen. LCD-näyttöä ohjataan pinneistä p15 – p20, numeronäppäimistö on yhdistetty pinneihin p21 – p28 ja anturi tulee pinniin p30 ja FETiä ohjataan pinnistä p29. Kuviossa 6 on esiteltyä mBed:n pinnijärjestys.

### 3.3 Näppäimistö

Laitteen toiminta vaatii oheislaitteiksi 4x4 kokoisen numeronäppäimistön, josta löytyy tarvittava määrä näppäimiä laitteen oleellisimpiin tarkoituksiin eli numerot 0 – 9. Näppäimistön kytkennässä käytetään seitsemää kytkentäpistettä, jotka muodostavat 3x4-matriisin. Ensimmäiset kolme pinniä määrittävät pystyrit ja seuraavat 4 pinniä määrittävät vaakarivit. Lukemalla muodostettua bittikuviota ohjelmallisesti päätellään mitä näppäintä tai näppäimiä käyttäjä milläkin hetkellä painaa. Näppäimistö kytkettiin pinneihin p21 – p28. Kuviossa 7 on esiteltyä, kuinka näppäimet tekevät matriisin.

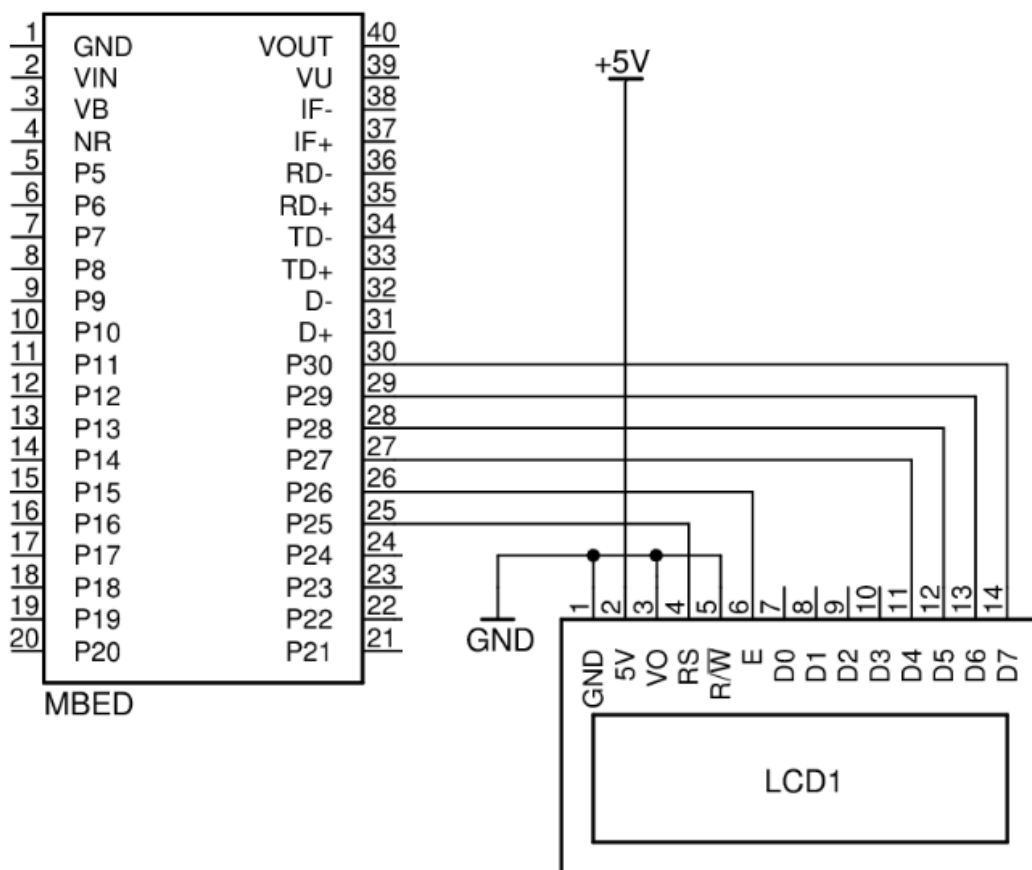


KUVIO 7. Näppäimistö matriisi



### 3.4 Näyttö

Tarvittavan informaation syöttöön ja seuraamiseen tarvitaan LCD-näyttö, jolta dataa on käyttäjän helppo lukea. Näytöksi valikoitui Hitachin HD44780-standardia tukeva 2-rivinen näyttö, jolle kummallekin riville mahtuu 16 merkkiä. Näytön päällimmäinen valintakriteeri oli se, että niitä löytyi valmiiksi hyllystä eikä sitä erikseen tarvinnut tilata. Näytölle syötetään ajotilassa jatkuvasti dataa, joista ilmenee käämityt kierrokset, käämittävät kierrokset sekä alkuperäinen kierros määrä, joka on asetettu. Kuviossa 8 on esiteltyä esimerkkityöntä, kuinka näyttö kytketään mBed-mikrokontrolleriin.

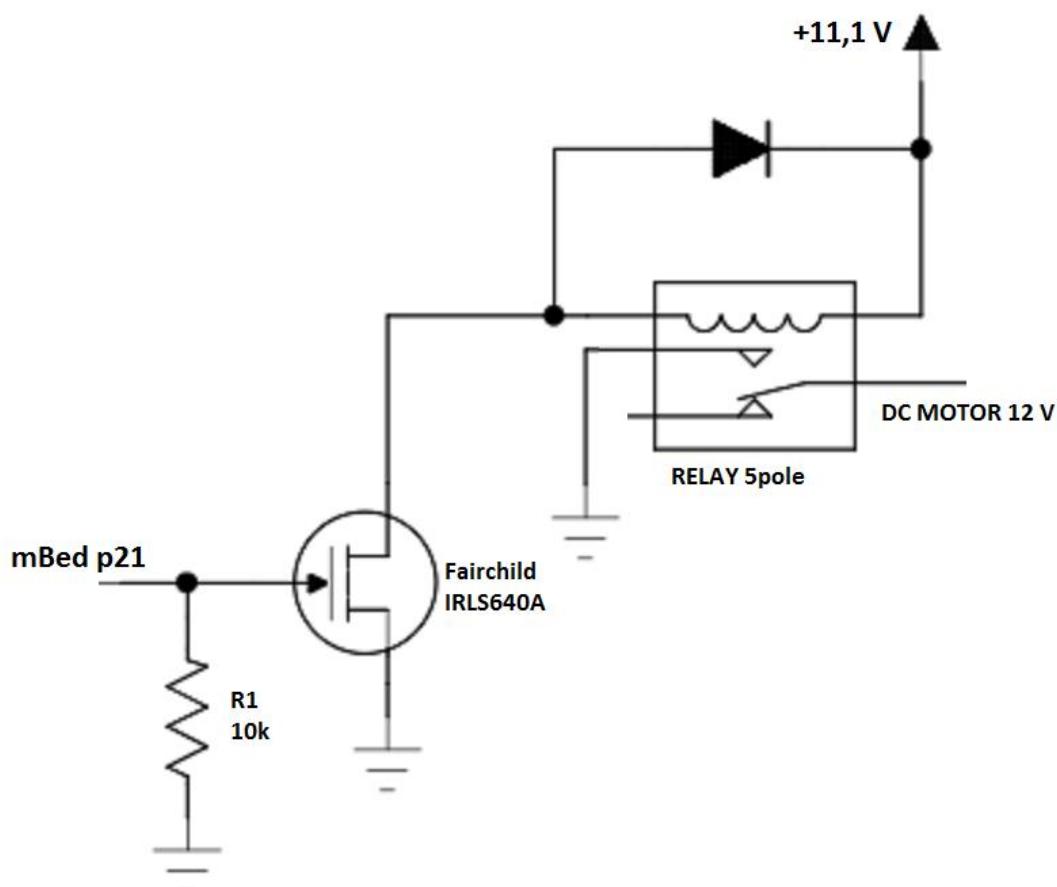


KUVIO 8. HD44780 yhteensopivan näytön kytkentä mBediin

Näyttö tarvitsee +5 V toimiakseen, ja tämä saadaan LM7805 +5 V regulaattorin muodostamasta +5 V:n linjasta. Näytön spesifikaatiossa on määritetty, että se voi toimia 4- ja 8-bittisten mikrokontrollerien kanssa; tästä syystä näyttöä on mahdollista ajaa 4 bitin tilassa isommillakin mikrokontrollereilla ja säästää 4 nastaa muihin tarkoituksiin (Hitachi 1998). Näytössä olisi vihreä taustavalo olemassa, mutta tässä tapauksessa sitä ei tarvita, koska laitetta käytetään ainoastaan hyvin valaistuissa olosuhteissa.

### 3.5 Rele ja MOSFET

Moottorille menevää virtaa kytketään 12 V 5-napaisella vaihtoreleellä ja relettä kytketään Fairchild Semiconductorin IRLS640A-logiikkatason mosfetin kautta, jota kytkee mBed-mikro-ohjain. Koska releessä on kela sisällä, noudattaa se perusfysiikkaa kelan osalta, jossa virran katketessa kelalle magneettikenttä purkautuu aiheuttaen hetkellisen jännitepiikin. Tästä syystä releelle on asetettu oma snubber diodi, jonka tehtävänä on eliminoida tämä jännitepiikki mahdollisimman tehokkaasti (TE Connectivity 1998). Diodiksi on valittu 1N4007 sen helpon saatavuuden vuoksi. Kuviossa 9 on esiteltynä snubber diodin kytkentä releelle.



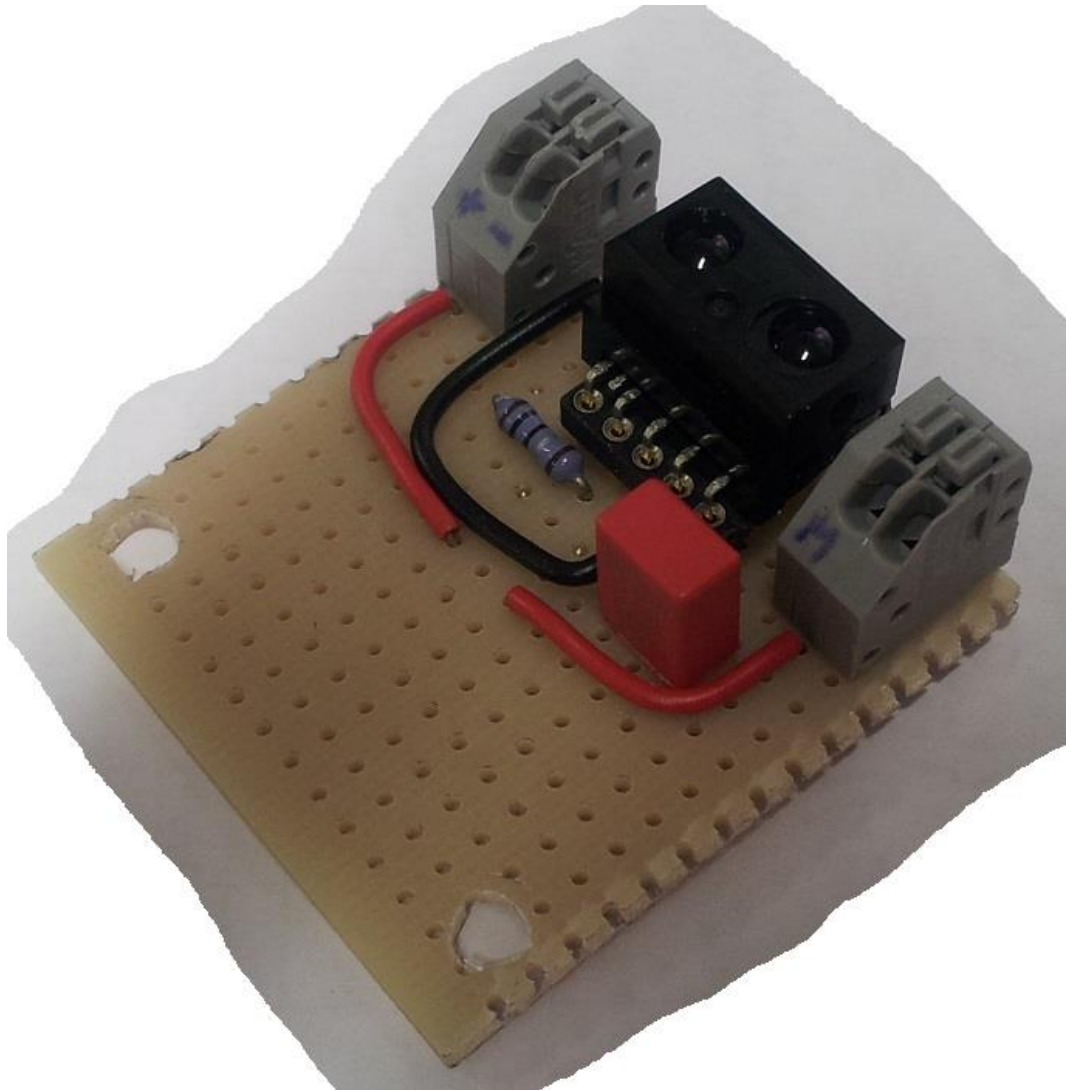
KUVIO 9. IRLS640A FETin ja releen kytkentä mBediin

Valitsin MOSFETiksi Fairchild Semiconductorin IRLS640A:n, koska se on logiikkatason MOSFET. Logiikkatasolla tarkoitetaan MOSFETin kanavan aukeamisjännitteen pistettä, joka soveltuu paremmin käytettäväksi TTL-logiikan (+5 V) kanssa, kun perinteiset MOSFETit on tarkoitettu +10 V:n käyttöön (esim IRF530). (Fairchild 2008)

mBed kykenee tuottamaan vain 3,3 V ulostulojännitteen ja tästä syystä ei-logiikkatason MOSFETit eivät kelvannut, koska ei ollut varmaa, kykeneekö ne avaamaan kanavaa riittävästi. Kokemus kuitenkin osoitti IRLS640A:n toimivan hyvin mBed kohteissa.

### 3.6 Kierroslukuanturi

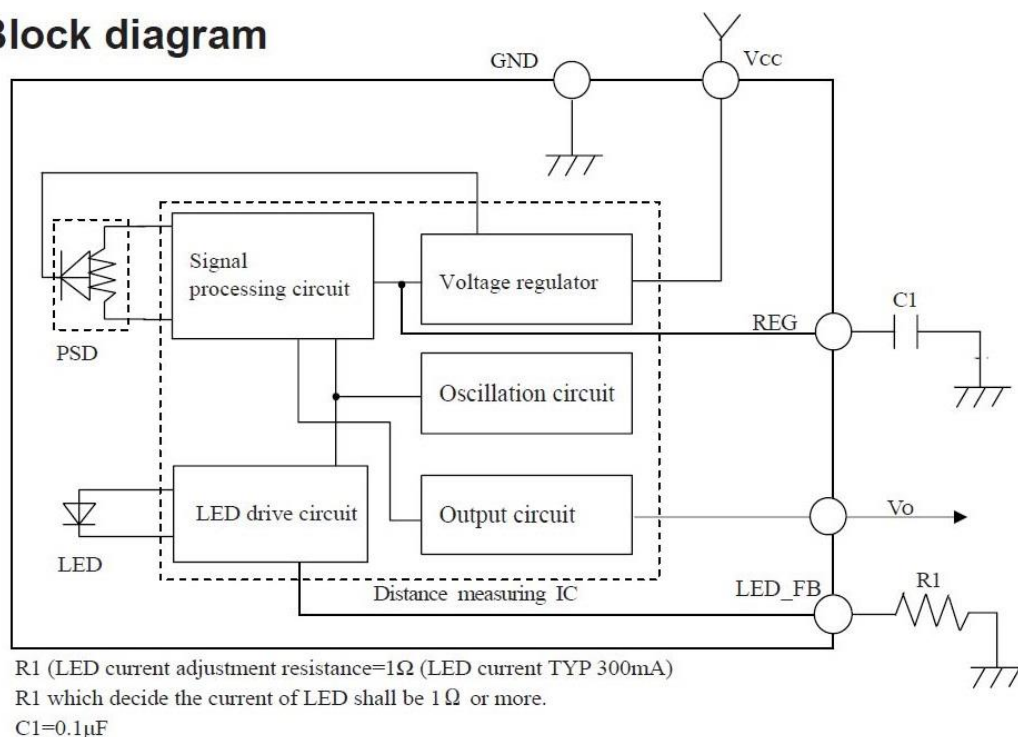
Reikäpöydän lukusensorina toimii Sharpin valmistama optinen digitaalinen infrapunalla toimiva lähestymisanturi. Sensori on tyypiltään GP2Y0D340K, joka on tarkoitettu lähestymisanturiksi sellaisiin kohteisiin kuten kosketuksettomat vesihanat ja ohjaamaan laitteiden taustavalaistusta virransäästöä vaativissa kohteissa (Sharp 2013).



KUVIO 10. Sharpin optinen anturi + oheiskytkennät

Sensorin tehtävänä on lukea hammaspyörältä reikiä, joista se muodostaa signaalin mikro-ohjaimelle. Kyseisestä sensorityyppistä ulostulo on puhtaasti digitaalinen kantiaalto. Sensori on esiteltyä kuvion 10 oikeassa yläkulmassa, sensorin on musta laatikko, jossa on kaksi linssiä.

## Block diagram



KUVIO 11. Sharp GP2Y0D340K-lohkokaavio (Sharp 2013)

Sharpin datalehteä tutkimalla selviää sensorin sisältävän enemmän elektroniikkaa kuin lähitarkastelulla voisi olettaa. Kuviossa 11 on esiteltyä sensorin sisältämä elektroniikka, joka koostuu signaalinprosessointipiiristä, jänniteregulaattorista, ledinohjauspiiristä sekä niiden oheislaitteista. Lohkokaaviosta myös selviää, että sensorin tarvitsee oheiskytkennät toimiakseen. Datalehdessä on esitelty, että vastus R1 on 1 Ω, regulaattorin ja maan väliin tulee 0.1μF kondensaattori. Jos datalehdessä olevan C1 asettaa oheiskytkentään, ei laite toimi, vaan se pitää kytkä suoraan maahan. Kyseiselle ongelmalle ei löytynyt mitään selitystä.

Alkuperäinen suunnitelma oli käyttää Variable Reluctance -tyypin sensoria lukemaan reikäpyörän kierroksia, koska VR-tyypin sensorilla on muutamia vahvuuksia, jotka tekevät siitä halutun vastaavanlaiseen käyttöön. VR-sensorin vahvuuksiin lukeutuu sen mekaaninen yksinkertaisuus, jossa ei ole yhtään

liikkuvaa osaa. Sensori koostuu kestopagneetista, kelasta, ferriittisestä sydäimestä sekä lukukelasta. Lisäksi sensorit ovat erittäin halpoja ja laajasti saatavilla.

Koska VR-sensorin ulostulo on sinimuotoista ja ulostulevan signaalin amplitudi on vahvasti riippuvainen ohittavan metallisen kappaleen, nopeudesta ne vaativat signaalinkäsittelyyn tarkoitettua erillisen piirin (Hearst Electronic Products 2000). VR-sensori ei lähemmän tarkastelun jälkeen sovellu opinnäytetyön käyttökohteeseen, koska nopeudet ovat hitaimmillaan noin 1 kierros per sekunti ja VR-sensorilta tuleva signaali ei näissä nopeuksissa tuota riittävän voimakasta signaalia, josta olisi mahdollista signaalinkäsittelijän lukea senhetkistä tilaa.

Työssä käytetyn minimi pyörintänopeuden johdosta päädyin käyttämään Sharpin valmistamaa optista sensoria, joka toimii puhtaasti lähestymisperiaatteella. Lähestymisperiaatteessa on omat heikkoutensa, esimerkiksi jos sensorin taakse sijoitetaan jotain häiritsemään sen toimintaa. Laite on suunniteltu käytettäväksi pöydällä, ja ainoa tapa häiritä sensorin toimintaa on asettaa sormet lähelle reikäpyörää kesken kääminnän. Tämä ominaisuus on tiedostettu, ja siitä tulee informoida tulevia käyttäjiä. Sharpin sensorin vahvuutena voidaan pitää sitä, että se ei tarvitse erillistä signaalinkäsittelypiiriä, vaan siltä saatu signaali voidaan käyttää sellaisenaan mBed mikro-ohjaimella.

## 4 OHJELMAKOODI

Ohjelmakoodin suunnittelun aloitin tekemällä vuokaavioita ohjelman toiminnasta ja hahmottelemalla, millä tavalla ohjelmakoodista saadaan mahdollisimman yksinkertainen ja toimiva. Valmiiden vuokaavioiden pohjalta ohjelmakoodin kirjoittamisesta tuli paljon helpompi ja selkeämpi prosessi. Aloin tehdä ohjelmakoodia C++-kielellä käyttäen mbed.org-pilvipalvelun kehitysympäristöä. Koodia ei kyetä simuloimaan kehitysympäristössä, vaan se oli aina testattava piirillä, koska toiminnot pääasiassa pyörivät oheislaitteiden ympärillä ja ympäristö ei sisällä minkäänlaista debuggeria.

### 4.1 Ohjelmointiympäristö

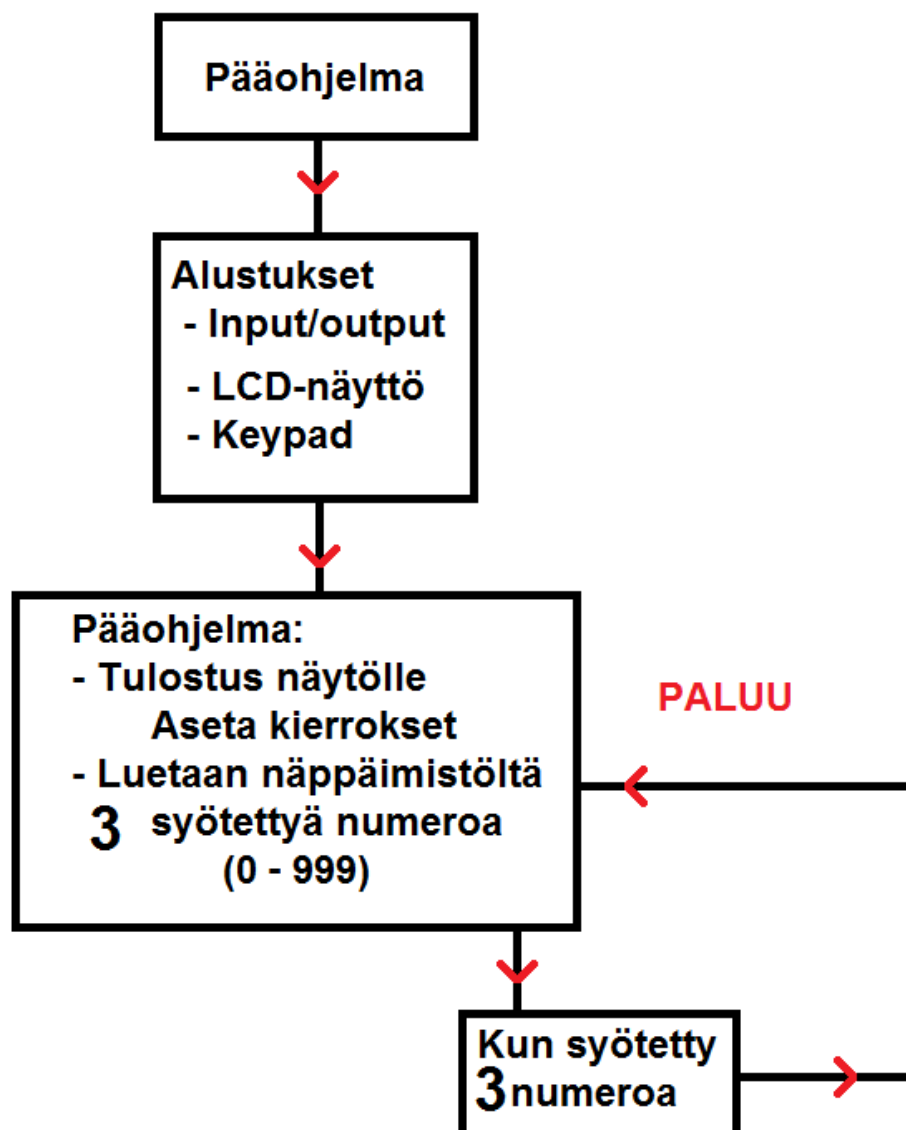
Ohjelmointiympäristönä mbed.org on käytetystä alustasta riippumaton, koska se pyörii täysin verkossa. Kaikki käyttäjän kirjoittama lähdekoodi säilyy mbed.org-palvelimilla, ja ainoat asiat, jotka tarvitaan työn tekemiseen, on internet-yhteys ja riittävän uusi web-selain, joka tukee tarvittavia standardeja. Ohjelmistoeditori osaa automaattisesti sisentää ja värjätä koodilohkot niiden luettavuuden helpottamiseksi.

Ohjelmistoeditorin kääntäjä tukee C- ja C++- ohjelmistokieliä. Opinnäytetyön ohjelmisto on kirjoitettu käyttäen C++-kieltä sen monipuolisuuden ja helpomman käyttöönoton vuoksi.



## 4.2 Ohjelmiston suunnittelu ja toteutus

Kuviossa 12 on esitelty pääohjelman toiminnot, missä ohjelmakoodin suoritus aloitetaan suorittamalla pakolliset alustukset näytölle ja numeronäppäimistöille. Näyttö ei toimi, jollei sille suoriteta tarvittavia alustuksia, ja se on helppo tunnistaa, kun näytössä ylärivissä on mustia neliöitä ja alarivillä ei näy mitään. Numeronäppäimistö vaatii alustuksessaan jokaisen pysty- ja vaakarivien näppäinyhdistelmien määrittelyt, jotta ohjelma tietää, mitä näppäintä milloinkin painetaan.



KUVIO 12. Pääohjelman vuokaavio

Kun alustukset on suoritettu, siirrytään ikuiseen ohjelmistolliseen silmukkaan, jonka sisällä on pääohjelma. Pääohjelmassa ensimmäisenä esitetään LCD-näytön kautta käyttäjälle viesti ”insert 000-999” ja tämän jälkeen siirrytään toimintoon odottamaan käyttäjän syötettä näppäimistöön. Järjestelmään on määritetty luvun 999 olevan maksimi, jonka sille voi antaa, koska suurempien määrien kääntämisessä meni aivan kohtuuttoman kauan aikaa ja E20/10/6-kelarunkoon ei tämä määräkään mahdu fyysisesti. Porakoneen moottori ja vaihdelaatikko kykenevät pyörittämään työkalua maksimissaan 650 kierrosta per minuutti, joten on luonnollista, ettei isommissa lukemissa olisi mitään ideaa ja tarvittaessa maksimi kierrosmäärää on helppoa nostaa ohjelmakoodia muuttamalla. Kun käyttäjä on syöttänyt kolme numeroa, siirrytään kuviossa 12 esitettyyn aliohjelmaan suorittamaan itse laskuritoimintoa.

Alkuperäisessä suunnitelmassa näppäimistölle syötetään lukema, joka halutaan ajaa, ja se hyväksytään erillistä painiketta käyttäen. Tästä suunnitelmasta täytyi luopua, koska mBedin ohjelmointiympäristöstä puuttuvan debuggerin takia en kyennyt selvittämään, miksi muuttujajonoa ei kyetty kääntämään stringistä integeriksi. Tästä syystä näppäimistöä käsitellään niin, että siitä on syötettävä aina 3 numeroa ja tämä ominaisuus mainitaan laitteen käyttöohjeessa.

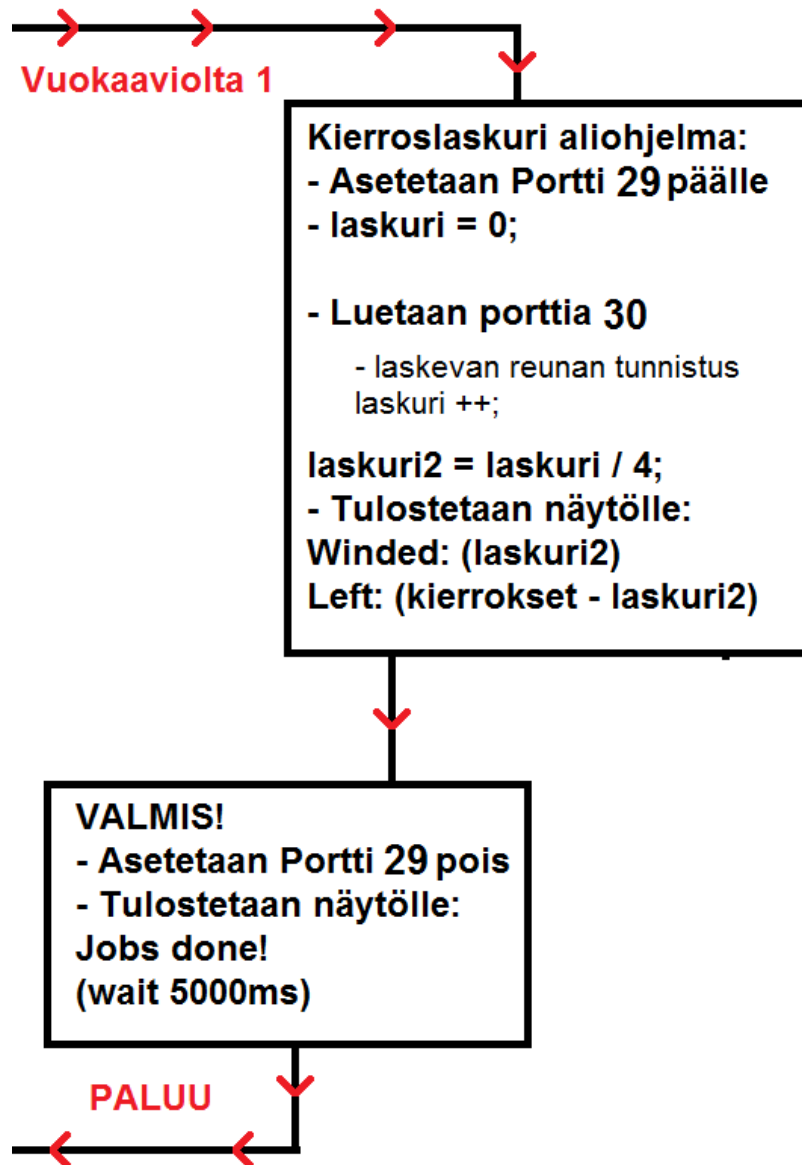
Näppäimistöltä syötetyt numerot käsitellään numero kerrallaan, minkä jälkeen ne kerrotaan oikeaan dekadiin ja lisätään omaan kokonaislukumuuttujaansa. Jokainen annettu numero tulostetaan näytölle, jotta käyttäjä näkee, mitä on syöttänyt, ja ei jää arvauksien varaan, mitä on saattanut syöttää.

Pääohjelman sisällä on kaksi while-silmukkaa, joista ensimmäisessä odotetaan käyttäjän syöttävän haluamansa lukeman laitteelle, minkä jälkeen muutetaan silmukan ajolle olennainen muuttuja tilaan 0, jolloin siirrytään seuraavaan silmukkaan. Jälkimmäisessä silmukassa suoritetaan itse kierroslaskuritoiminto, joka pysäytetään kierrosten tullessa täyteen.

Näppäimistöä käsitellään neljänä ulostulona ja neljänä sisääntulona, ohjelmassa kytketään yksi ulostulo kerrallaan päälle, tutkitaan, tuleeko yhdestäkään sisääntulosta mitään sisälle, ja jos tulee, niin palautetaan lukema pääohjelmalle. Esimerkiksi lukeman 5 syöttämiseksi vertaillaan, ensin tuleeko mitään riviltä 1

mitään ulos, koska mitään ei tule, siirrytään riville 2, josta ohjelma vertaa jonon toisesta sisääntulosta tulevan signaalin sisälle. Tästä saadaan muodostettua matriisityyppinen havainnointi. Tekniikan heikkona puolena on, ettei ole mahdollista painaa kuin yhtä painiketta kerrallaan ilman, että tulee virheitä syötteeseen. Hyvänä puolena on sen vähäinen tarve mikrokontrollerin sisään ja ulostulojen. Etenkin suuremmilla näppäinmatriiseilla saadaan säästettyä paljon ohjausportteja verrattuna yksittäisiin painikkeisiin.

Näytönohjaukseen käytetään mBed.orgista löytyvää valmista TextLCD-kirjastoa, jonka on kirjoittanut henkilö nimeltä Simon Ford. Kirjastolla on pyritty tekemään HD44780-yhteensopivien LCD-näyttöjen ajaminen mahdollisimman helpoksi. Kirjasto kaipaa vain tiedon näytön rs-, enable- ja data4 – data7-johtojen sijainnista mikrokontrollerilla. Tämän jälkeen LCD-näyttöä käsitellään ohjelmassa samalla tavalla kuten PC-ohjelmistoympäristöissä konsoliin tulostusta.

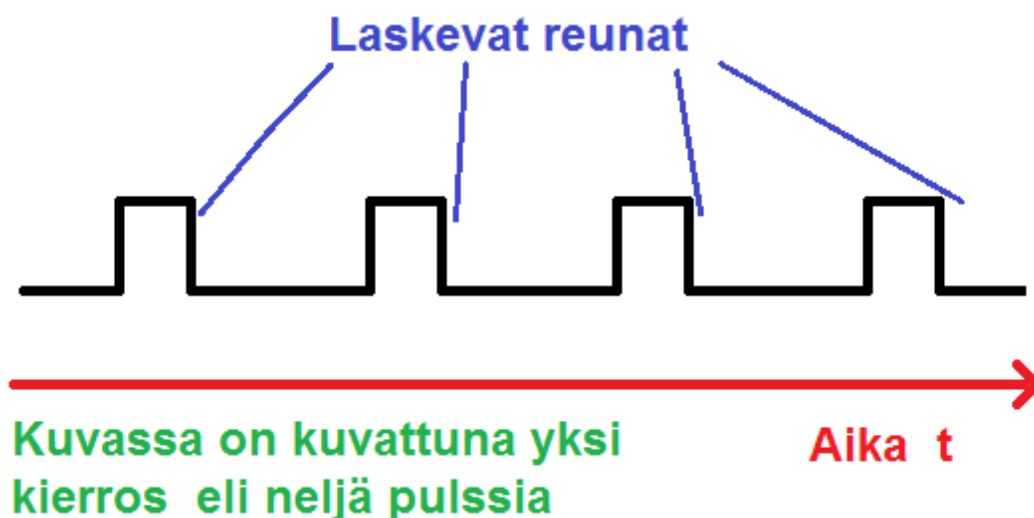


KUVIO 13. Aliohjelman vuokaavio

Kuviossa 13 on esitettyä aliohjelma, jossa suoritetaan itse laskenta, ja se missä käyttäjä viettää suurimman osan ajasta. Ensimmäisenä asetetaan poljinyksikölle virrat päälle kytkemällä p29 päälle. Muuttujia on kaksi kappaletta, joista toinen sisältää lasketut pulssit ja toisen tehtävänä on laskea laskettujen pulssien määrästä kierrosten lukumäärä ja näyttää se käyttäjälle näytöllä.

Pulssit halutaan laskea laskevalta reunalta, koska muussa tapauksessa mikro-ohjain tulkitseisi olevansa jatkuvasti nousevalla reunalla ja laskuri pyörisi niin nopeasti ympäri kuin ohjelman suorittamisen kannalta olisi mahdollista. Mikro-ohjain ei osaa tulkita laskevia reunoja kuin yhden kappaleen per pulssi. Tästä

saadaan luotettavasti laskettua pulssien lukumäärä. Kuviossa 13 on esitelty laskevat reunat ja niiden määrä yhdeltä kierrokselta.



KUVIO 14. Laskevat reunat yhden kierroksen ajalta

Laskevan reunan tunnistus on toteutettu ohjelmallisesti, mBed sisältää valmiin toiminnon, jonka voi asettaa lukemaan joko nousevaa tai laskevaa reunaa ja jokaisen havaitun reunan kohdalla suorittaa haluttua toimenpidettä tai aliohjelmää. Erillisessä aliohjelmassa lasketaan jokainen laskeva reuna ja tulos syötetään globaaliin muuttujaan, josta ohjelma vertaa sitten ajatut kierrokset.

Kun laskuri saavuttaa asetetun raja-arvon, ohjelmisto kytkee virrat polkimelta pois ja pyöriminen pysähtyy. Tämän jälkeen näytölle tulostetaan teksti, jossa ilmoitetaan käyttäjälle toimenpiteen olevan valmis ja kierrokset, jotka kelarungolle ajettiin.

## 5 KAUPALLISET RATKAISUT

Prototyypimuuntajien käämintäkoneet ovat hyvin pieni markkinasegmentti markkinoilla, koska niitä tarvitsevat vain todella harvat yritykset, ja tästä syystä valikoima ei ole erityisen suuri pienempien laitteiden kohdalla. Yleisesti halvemmat laitteet ovat lihasvoimin pyöritettäviä laitteita, joissa on mekaaninen kierroslaskuri. Tällaisesta laitteesta esimerkkinä toimii Ingrid West Machineryn valmistama WH-737, joka on esiteltyinä kuviossa 15. Laitteille ei ole määriteltynä hintoja yleisestisaataville, vaan ne tulee kysyä myyjältä itseltään.



KUVIO 15. WH-737

Lihaskäyttöisissä laitteissa on huonoina puolina niiden vaatima keskittyminen käyttäjältä kierroslaskuriin, pyörittämiseen sekä käämilangan ohjaamiseen. Sähköllä pyörivät laitteet ovat harvinaisempia, ja nekin sisältävät useimmiten mekaanisen kierroslaskurin, mutta käyttäjän tulee silti keskittyä kierroslaskuriin ja käämilangan ohjaamiseen.

Ingrid West Machineryltä löytyy myös kehittyneempiä käämintäkoneita, jotka ovat mikro-ohjaimella ohjattuja, ja ne kykenevät itsenäiseen käämintään.

Tällaisesta esimerkkinä toimii WH-800, joka on 37 kiloa painavana aika iso laitos. Välimaastosta löytyy todella heikosti tarjontaa laitteiden osalta.

Muiden valmistajien ratkaisut olivat hyvin kalliita, suuria ja kompleksisia tuotoksia. Kiinasta löytyi Sanshine Electronics Co.Ltd:lta erilaisia isoja koneita, mutta pienempiä ei heillä ollut. Pääasiassa laitteet olivat kaikilla valmistajilla joko 230 V:n verkkovirrasta toimivia tai lihasvoimalla toimivia.

Työssä tehdylle laitteelle saattaisi olla kaupallista kysyntää, jos metallirungon tekisi koneilla sekä suunnittelisi rungon jäykemmäksi ja työkalun monikäyttöisemmäksi. Suunnittelemani laite eroaa muista akkukäyttöisyydellään, helpolla siirrettävyydellään, halpuudellaan ja sijoittumalla välimaastoon ominaisuuksiltaan.

## 6 TESTAUS JA MUUTOKSET

Laitteen toimivuuden varmistamiseksi, testasimme Juha Hyytiäisen kanssa sitä erilaisilla kelarungoilla ja yritimme saada mahdolliset virheet suunnittelussa esille. Huomasimme muutaman koekäytön jälkeen laitteen pohjalevyn joustavan, mikä johti rungon värähtelyyn. Tämän takia kierroslaskuri ei ollut täysin tarkka laskuissaan. Laskurivirheen korjaamiseksi tein ohjelmallisen muutoksen, joka eliminoi pientä värähtelyherkkyyttä kierroslaskurista. Myös pohjalevyyn tulee vanerilevy jäykistämään rakennetta ja estämään värähtelyä.

Huomioimme laitteen pysyvän paikallaan pöydällä ongelmitta sekä käyttöliittymän olevan riittävän selkeä. Ohjelmakoodiin oli jäänyt muutamat virheet, jotka aiheuttivat ongelmia, kun järjestelmä oli kerran ajettu ympäri. Korjasin nämä ohjelmakoodin ongelmat pikaisesti paikanpäällä, minkä jälkeen totesimme sen täyttävän vaatimukset.

Kelarungon kiinnitystyökalu oli suunniteltu alkuperäisen spesifikaation mukaan E20/10/6-kelarungolle, ja uusi spesifikaatio vaati sen toimivan myös E16-kelarungon kanssa, joka vaatii työkaluun pieniä muutoksia. Työkaluun tarvittavat muutokset toteutetaan myöhemmässä vaiheessa.

Laitteen polkimen toiminta oli riittävän sulavaa, eikä sen toimintaan kiinnitetty juurikaan huomiota. Poljin pysyi paikallaan koko käytön ajan, eikä se siirtynyt jalan painon alla ja vaikeuttanut käyttöä. Suunnittelu polkimen osalta todettiin onnistuneeksi. Testauksen pohjalta kirjoitin laitteelle lopuksi A4-kokoisen käyttöohjeen, jossa kuvataan laitteen käytön kannalta oleelliset osat.



## 7 YHTEENVETO

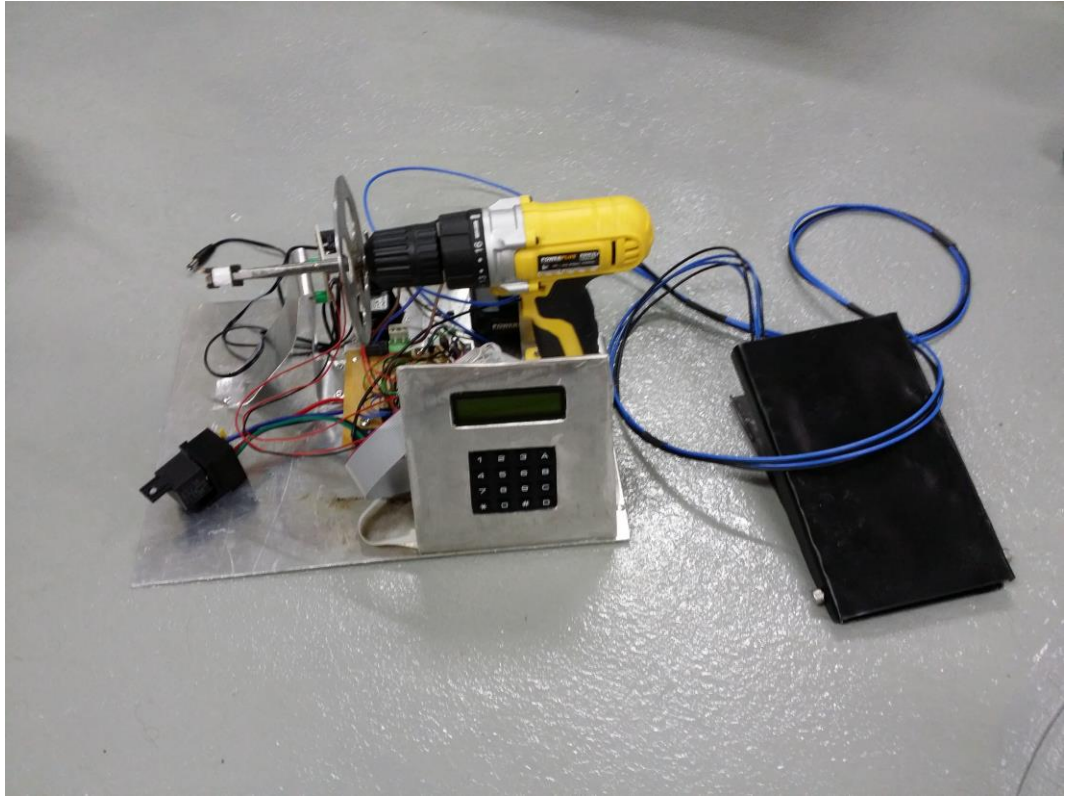
Ottaen huomioon työn yksinkertaisen luonteen epäilin aluksi, olisiko kyseessä ehkä liian yksinkertainen työ ollakseen opinnäytetyö. Mielipiteeni asiaa kohtaan kuitenkin muuttui työn edetessä, sillä ratkaistavana oli paljon mekaanisia pulmia sekä muutama elektroninen ja ohjelmistollinen seikka, jotka tuottivat ihan riittävästi haasteita. Koitin pitää laitteen rakennuskustannukset mahdollisimman alhaisina ja siinä onnistuin mielestäni kohtuullisen hyvin tuloksin.

Paljon aikaa kului itse työn mekaanisten osien valmistamiseen ja viimeistelyyn. Piti suunnitella ja valmistaa kaikki osat työhön itse, koska minulla ei ole pääsyä konepajaan saatika riittäviä teknisiä taitoja käyttää niitä koneita. Tästä syystä työstä tuli erittäin käsintehtyn näköinen. Tämän ei kuitenkaan pitäisi haitata koska kyseessä on vain yksittäinen prototyyppi, jota käytetään opetuskäytössä.

Laitteen elektroniikkapuoli on erittäin yksinkertainen, eikä sen suunnittelu tuottanut suurempaa päänvaivaa. LCD-näytöt olivat entuudestaan tuttuja ja helppokäyttöisiä Hitachin HD44780-standardin mukaisia näyttömoduuleita. Numeronäppäimistön käyttöönotto aiheutti hieman mietintää, mutta tutkimalla erilaisia toteutuksia löysin siihenkin ratkaisun. Eniten mietintää aiheutti moottorin nopeudensäätö ja siihen liittyvät laitteet, mutta nekin sain yksinkertaisesti toteutettua osilla, joita on helposti saatavilla nyt ja tulevaisuudessa.

Ohjelmakoodi osoittautui yksinkertaisemmaksi kuin alkuun oletin. Hahmoteltuani sen ensin lohkoakaaviona tuli ohjelmasta yllättävän yksinkertainen ja lyhyt. Hankalin osuus oli näppäimistöltä saadun datan näyttäminen näytöllä ja sen käyttäminen laskurissa. Ohjelmakoodi kirjoitettiin käyttäen mbed.org kääntäjää ja sen kirjoittamiseen käytettiin Windows8-PC:tä ja USB-kaapelia.

Laite toimii työvaatimuksessa kuvatulla tavalla, joten voin olettaa sen täyttävän vaatimuksensa ja olevan onnistunut työnä. Siinä yhdistyy mekaaninen työ ja elektroninen työ samassa paketissa.



KUVIO 16. Käämintäkone valmiina

Kuviossa 16 on esitelty laite lopullisessa muodossaan. Siitä käy selville laitteen kokoonpano, jossa näkyy poljin, hallintapaneeli, työkalu, elektroniikka ja koneen runko. Tällaisena toimitin laitteen Juha Hyytiäiselle opetuskäyttöön ja ainoa muutos, joka kokoonpanoon tulee on paksu puulevy pohjaan, jonka tehtävänä on jäykistää laitteen runkoa.

## LÄHTEET

Fairchild. 2008. LM78XX 3-terminal positive voltage regulator datasheet [viitattu 1.4.2013]. Saatavissa: <http://www.farnell.com/datasheets/192157.pdf>

Hearst Electronic Products. Which speed sensor is wrong for you.[Viitattu 3.4.2013] Saatavilla:

[http://www.electronicproducts.com/Electromechanical\\_Components/Hall\\_vs\\_VR\\_Which\\_speed\\_sensor\\_is\\_wrong\\_for\\_you.aspx](http://www.electronicproducts.com/Electromechanical_Components/Hall_vs_VR_Which_speed_sensor_is_wrong_for_you.aspx)

Hitachi. Dot Matrix Liquid Crystal Display Controller/Driver [Viitattu 1.4.2013]. Saatavilla: <http://lcd-linux.sourceforge.net/pdffdocs/hd44780.pdf>

Ingrid West Machinery [viitattu 20.10.2013] Saatavilla:

<http://www.coilwindingmachines.eu/>

mBed.org handbook mbed microcontrollers [1.4.2013]. Saatavilla:

<http://mbed.org/handbook/mbed-Microcontrollers>

NXP. LPC1768 [viitattu 1.4.2013] Saatavilla:

[http://www.nxp.com/documents/data\\_sheet/LPC1769\\_68\\_67\\_66\\_65\\_64\\_63.pdf](http://www.nxp.com/documents/data_sheet/LPC1769_68_67_66_65_64_63.pdf)

Sharp GP2Y0D340K [Viitattu 1.4.2013] Saatavilla:[http://sharp-](http://sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y0d340k_e.pdf)

[world.com/products/device/lineup/data/pdf/datasheet/gp2y0d340k\\_e.pdf](http://sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y0d340k_e.pdf)

TE Connectivity. The application of relay coil suppression with DC relays.

[viitattu 1.4.2013]. Saatavilla: [http://relays.te.com/appnotes/app\\_pdfs/13c3311.pdf](http://relays.te.com/appnotes/app_pdfs/13c3311.pdf)

# Käämintäkoneen KÄYTTÖOHJE

- Aseta numerot laitteeseen 000-999 väliltä seuraavalla tavalla:
  - o Jos haluat käämiä 150 kierrosta, syötä 1 5 ja 0
  - o Jos haluat käämiä alle 100, aloita **nollalla**... esim 45 -> **045**
  - o alle 10 noudattaa myös samaa, mutta **2kpl** nollaa eteen
  - o Jos mokaat kierroksien syöttämisen niin paina mBed **reset** nappulaa
- Laite käämiä lankaa ainoastaan yhteen suuntaan ja takaisinpäin kääntäminen lasketaan myös kierroksina, joten koita saada muuntajan kierrokset kerralla oikein.
- Jos laitteen pyörimissuuntaa tulee muuttaa niin polkimessa on kytkin suunanvaihdolle. Varo jättämästä kytkintä keskelle, sillä silloin poljin ei painu.
- Aloita käämintä normaalisti, varo kuitenkin painamasta polkimesta liikaa kierroksia.
- Kun käämintä on valmis, laite näyttää ajatut kierrokset +-1 tarkkuudella. 4sekuntin kuluttua laite resetoituu alkutilaan, jonka jälkeen voidaan syöttää taas kierrokset joita halutaan ajaa seuraavan kerran.

Ohjelmakoodi:

```
#include "TextLCD.h"
#include "mbed.h"

DigitalOut led1(LED1);
DigitalOut led2(LED2);
DigitalOut led3(LED3);
DigitalOut led4(LED4);
DigitalOut fetdrive(p29);
InterruptIn sensor(p30);

// näppiksen input columnit
DigitalIn col1(p28);
DigitalIn col2(p27);
DigitalIn col3(p26);
DigitalIn col4(p25);

// näppiksen output rivejä varten
DigitalOut row1(p24);
DigitalOut row2(p23);
DigitalOut row3(p22);
DigitalOut row4(p21);

TextLCD lcd(p20,p19,p18,p17,p16,p15); // rs, e, d4-d7

int edgecount = 0;
int kierrokset = 0;

int getKeyNum();

void falling_edge(void)
{
    edgecount++;
}

int main(void)
{
    while(1) {
        int muut1 = 0; // välimuuttuja ekalle numerolle
        int muut2 = 0; // välimuuttuja tokalle numerolle
        int muut3 = 0; // välimuuttuja kolmannelle numerolle
        int muut4 = 0; // välimuuttuja johon tungetaan ekat 3 numeroa

        int x = 10; // lcd näytön tekstiä varten
        int counter = 0; // tämä on välimuuttuja numerojen syöttöä varten
        int senscalc = 0; // tähän lasketaan kaikki sensorilta saatu data
```

```

float senscalc2 = 0; // tämä näyttää ajetut kierrokset kierroksina, eli senscalc / 4
int run = 1; // välimuuttuja run tiloille, eli otetaanko vastaan vai syötetäänkö
numeroa
int run2 = 0;
lcd.printf("insert 000-999:");

while(run == 1) {
  int a = getKeyNum();
  //lcd.cls();
  wait(0.1);

  if(a != -1) {
    counter++;
    x++;
    lcd.locate(x,1);
    lcd.printf("%d\n",a);
    wait(0.4);

    if (counter == 1) {
      muut1 = a;
    }

    if (counter == 2) {
      muut2 = a;
    }

    if (counter == 3) {
      muut3 = a;
      counter = 0;
      run = 0;
      run2 = 1;
    }
  }
}

muut4 = muut3 + (muut2 * 10) + (muut1 * 100);
kierrokset = (muut4 * 4); /* Because the sensor wheel has 4 slots, the value must
be multiplied by 4 */

}

lcd.cls();
lcd.printf("Set turns: %d", muut4);
lcd.locate(0,1);
lcd.printf("Turns now:");
fetdrive = 1; // FET- ON! Toggled power to the motor!

while (run2 == 1) {

  lcd.locate(10,1);
  lcd.printf("%3.1f", senscalc2);

```

```

    sensor.fall(&falling_edge);

    senscalc = edgecount;

    if (senscalc == kierrokset) {
        run2 = 0;
        fetdrive = 0;
    }

    senscalc2 = (senscalc / 4);
}

fetdrive = 0; // jobs done, power off from motor.

lcd.cls();
lcd.locate(0,0);
wait(0.1);
lcd.printf("Jobs done!\n Turned %4.1f", senscalc2);
wait(2.5);

}
}

```

```

int getKeyNum()
{

```

```

    row1 = 1;
    row2 = 0;
    row3 = 0;
    row4 = 0;

```

```

    if (col1)
        return 1;
    else if (col2)
        return 2;
    else if (col3)
        return 3;

```

```

    row1 = 0;
    row2 = 0;
    row3 = 0;
    row4 = 0;

```

```

    if (col1)
        return 4;
    else if (col2)
        return 5;
    else if (col3)
        return 6;

```

```

    row1 = 0;
    row2 = 0;
    row3 = 1;

```

```
row4 = 0;

if (col1)
    return 7;
else if (col2)
    return 8;
else if (col3)
    return 9;

row1 = 0;
row2 = 0;
row3 = 0;
row4 = 1;

if (col1)
    return -2;
else if (col2)
    return 0;
else if (col3)
    return -3;

row1 = 0;
row2 = 0;
row3 = 0;
row4 = 0;

return -1;
}
```