



LAHDEN AMMATTIKORKEAKOULU
Lahti University of Applied Sciences

TYÖNTÖMITAN LISÄLAITE

Sylinterimittasovellus

LAHDEN
AMMATTIKORKEAKOULU
Tekniikan ala
Tietotekniikka
Tietokone-elektroniikka
Opinnäytetyö
Syksy 2012
Nico Syrjäläinen

Lahden ammattikorkeakoulu
Tietotekniikka

SYRJÄLÄINEN, NICO:

Työntömitan lisälaite
Sylinterimittasovellus

Tietokone-elektroniikan opinnäytetyö, 25 sivua

Syksy 2012

TIIVISTELMÄ

Opinnäytetyön tarkoituksena oli tehdä tutkimustyötä mahdollisesta lisälaitteesta Mitutoyo Absolut Digimatic -työntömitalle. Lopuksi mahdollisuuksien mukaan oli myös tarkoitus toteuttaa laitteen toiminnallinen osuus.

Lisälaitteen tehtävänä oli vastaanottaa tietoa digitaaliselta työntömitalta ja muokata sitä haluttuun muotoon ja lopulta tulostaa mittauksia erilliselle näytölle. Laitteen prosessorina käytettiin mbed-mikrokontrolleria ja tulosteiden esittämiseen LCD-näyttöä, jollaista käytetään matkapuhelimissa. Tavoitteena oli laskea työntömitalla mitatusta tuloksesta erillisellä kaavalla suurien halkaisijoiden mitta. Kaavan opinnäytetyöhön toimitti Mäntsälän saha Oy.

Tutkimustyö muuttui lopulta laitteen toiminnalliseksi toteuttamiseksi. Vaatimuksia laitteen toiminnalle tuli työn edetessä. Sen toteuttaminen toiminnallisesti ja ohjelmallisesti onnistui ja laite luovutettiin testauskäyttöön Mäntsälän sahalle.

Asiasanat: mikrokontrolleri, digitaalinen työntömitta, lisälaite

Lahti University of Applied Sciences
Degree Programme in computer electronics

SYRJÄLÄINEN, NICO: Cylinder gauge application

Bachelor's Thesis in computer electronics, 25 pages

Autumn 2012

ABSTRACT

The subject of this Bachelor's Thesis was to do research on a possible accessory for Mitutoyo Absolut Digimatic slide-calibre. If such a device turned out to be feasible, the aim was to build it. This thesis was commissioned by Mäntsälän Saha Oy.

The purpose of the accessory was to receive information from the digital slide-calibre and modify it. Then the measurements were supposed to be printed on a separate screen. The device's processor was an mbed microcontroller, and Nokia's LCD screen was used for printing the measurements. The goal was to make it possible to measure diameters bigger than normal with a specific formula. The formula was given by Antti Perttilä, the CEO of Mäntsälän Saha Oy.

At the end of this project, the device was built. Requirements for the functions of the device multiplied during the process. The program and functional parts were successfully made. After finishing the device, it was given to Mäntsälän saha for test use.

Key words: microcontroller, digital slide-caliper, accessory

SISÄLLYS

1	JOHDANTO	1
2	LÄHTÖKOHDAT	2
2.1	Työn tausta	2
2.2	Kaavan johtaminen	2
2.3	Työn tarkoitus ja tavoitteet	3
3	KÄYTÖSSÄ OLLEITA LAITTEITA	5
3.1	Työntömitta	5
3.2	Mbed-mikrokontrolleri	7
3.3	Nokia LCD -näyttö	9
4	KYTKENTÄKAAVIOT JA NIIDEN TOIMINTA	10
5	TYÖNTÖMITAN DATA	12
5.1	Bittien painoarvo	13
5.2	Tavujen merkitys	14
5.3	Datan muunnos	14
5.4	Datan lähetys	15
6	OHJELMALLINEN TOTEUTUS	16
6.1	Tallennus taulukkoon	16
6.2	Tallennus muuttujaan	17
6.3	Painikkeiden ohjelmallinen toteutus	17
6.4	Matemaattiset toteutukset	18
6.5	Tulostus	18
6.5.1	Nollaus, tallennus ja b-mitta	19
6.5.2	Mittausten tulostus	19
7	TOIMINNALLINEN TOTEUTUS	22
8	YHTEENVETO	23
	LÄHTEET	25

1 JOHDANTO

Opinnäytetyön tarkoitus oli selvittää, onko mahdollista tehdä sovellus ja laite, joka sallii suurien halkaisijoiden mittauksen digitaalisella Mitutoyo-työntömitalla. Työn edetessä tarkoitukseksi muotoutui myös toteuttaa kyseinen laite. Tavoitteeksi asetettiin oman osaamisen ja taidon lisääminen tietokone-elektroniikan sekä ohjelmoinnin alueilla. Opinnäytetyön aihe saatiin Mäntsälän Saha Oy -nimiseltä yritykseltä, jolla itse idea kyseisestä sovelluksesta oli ollut jo muutamia vuosia.

Mäntsälän Sahan toimitusjohtaja Antti Perttilä on opiskellut aiemmin Lahden ammattikorkeakoulussa, ja hänellä oli selkeä visio sovelluksesta, joka mahdollistaisi suurien kappaleiden halkaisijan mittaamisen. Hänellä oli kaava, jonka avulla laite pystyttäisiin toteuttamaan. Idean ollessa valmis tarvittiin toteutus.

Tässä opinnäytetyössä käydään läpi sitä kuinka tieto digitaaliselta työntömitalta otetaan talteen mbed-mikrokontrollerille ja kuinka sitä käsitellään. Työssä käytettiin mbed:n lisäksi Nokian LCD-näyttöä, jolle työntömitalta saadut tiedot tulostettiin halutussa muodossa. Valmistuttuaan työntömitan lisälaite otettiin käyttöön Mäntsälän Sahalla.

2 LÄHTÖKOHDAT

Mäntsälän saha on perheyritys, joka nimensä mukaisesti sijaitsee Mäntsälässä. Sen perustivat yhteistyössä Vilho Perttilä sekä Martti Viljanen vuonna 1945, mutta myöhemmässä vaiheessa yritys siirtyi kokonaan Vilho Perttilän omistukseen. (Perttilä 2012.)

Vuosien saatossa Mäntsälän saha on kehittynyt höylästekniikan asiantuntijaksi ja on automatisoinut sen tuotantoa. Tällä hetkellä yritystä johtaa Antti Perttilä, kolmannessa sukupolvessa. (Perttilä 2012.)

2.1 Työn tausta

Jo vuosien ajan Mäntsälän sahan toimitusjohtaja Antti Perttilä on kaavaillut sovellusta, joka mahdollistaa suurien halkaisijoiden mittaamisen digitaalisella Mitutoyo -työntömitalla. Perttilä kehitti kaavan, jolla tämä laite voitaisiin toteuttaa. Mäntsälän Sahalla oli jo muutaman vuoden ajan yritetty toteuttaa laitetta, mutta valmista ei vielä ollut syntynyt.

2.2 Kaavan johtaminen

Kaavan johtaminen menee seuraavasti. Ympyrän säde on r ja halkaisija d .



KUVIO 1. Ympyrän säde ja halkaisija

$$d = 2 * r$$

Janojen x ja b summa on r .

$$x = r - b$$

Suorakulmaisesta kolmiosta saadaan Pythagooran lausella seuraavaa.

$$x^2 + a^2 = r^2$$

Kun sijoitetaan aiemmin saatu x yhtälöön, saadaan seuraavaa.

$$(r - b)^2 + a^2 = r^2$$

Kerrotaan yhtälö $(r - b)^2$ auki.

$$r^2 - 2br + b^2 + a^2 = r^2$$

r^2 kumoavat toisensa ja sen jälkeen yhtälö näyttää tältä.

$$-2br + b^2 + a^2 = 0$$

Siirretään $-2br =$ merkin toiselle puolelle.

$$b^2 + a^2 = 2br$$

Jaetaan yhtälö $2b$: llä, jotta saadaan selville r :n yhtälö.

$$r = \frac{a^2 + b^2}{2b}$$

Ja koska $d = 2r$, saadaan d :n tulokseksi seuraava.

$$d = \frac{a^2 + b^2}{b}$$

2.3 Työn tarkoitus ja tavoitteet

Työn tarkoituksena oli tutkia, olisiko mahdollista toteuttaa Perttilän suunnittelema sovellus ja mahdollisesti myös kokonaisen laitteen toiminnallinen toteutus.

Mitattuja tuloksia tuli laskea Perttilän antamalla kaavalla ja sen jälkeen tulostaa ne erilliselle näytölle. Tulokset tuli tallentaa mikrokontrollerin muistiin ja laskea mitattujen tulosten keskiarvo sekä minimi- ja maksimiarvot. Työn edetessä vaatimuksia laitteelle tuli lisää. Kun ohjelmallinen puoli saataisiin kuntoon, olisi vuorossa toiminnallinen toteutus mahdollisuuksien mukaan. Työn tavoitteeksi puolestaan asetettiin oman osaamisen ja taidon lisääminen tietokone-elektroniikan

sekä ohjelmoinnin alueilla. Käyttökohteita laitteella ovat ahtaat paikat, joihin iso työntömitta ei mahdu ja suuremman halkaisijan omaavat säiliöt sekä sylinterit.

3 KÄYTÖSSÄ OLLEITA LAITTEITA

Mäntsälän Sahan toimitusjohtaja Antti Perttilä toimitti tutkimustyöhöni Mitutoyo-merkkisen työntömitan, jossa on pieni digitaalinen näyttö. Samalla hän antoi työntömittaan kuuluvan kaapelin, jonka avulla oli tarkoitus ottaa tietoa talteen.

Mitutoyo-työntömitan lisäksi opinnäytetyössä käytettiin tiedon käsittelijänä mbed-mikrokontrolleria. Mitatuille tuloksille saatiin Lahden ammattikorkeakoulusta Nokian puhelimen näyttö. Tarkoituksena oli saada näyttö toimimaan mikrokontrollerin kanssa ja tulostaa haluttua tietoa erilliselle näytölle. Koska tulostettavaa tietoa oli paljon, oli tarkoitus saada hyötysuhteeltaan sopivan kokoinen näyttö projektiin.

3.1 Työntömitta

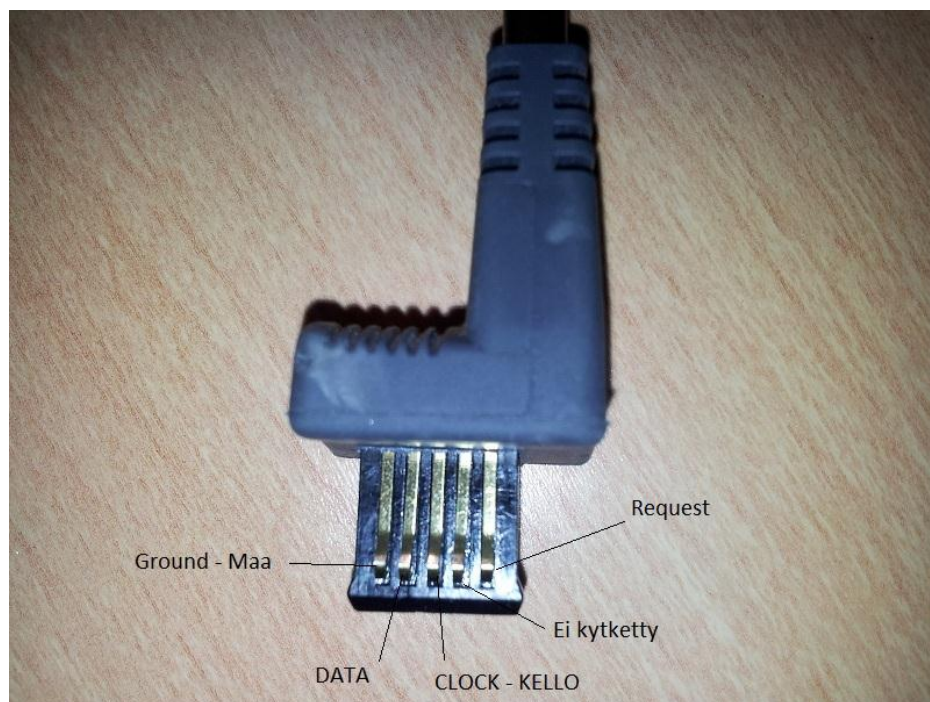
Työssäni oli käytössä Mitutoyo Absolute Digimatic -merkkinen digitaalinen työntömitta (KUVIO 2) pienellä näytöllä, josta näkee mitatun tuloksen kahden desimaalin tarkkuudella. Työntömitalla pystytään mittaamaan kappaleen halkaisija, sisämitta ja syvyys. Kyseisellä laitteella pystyttiin mittaamaan halkaisijoita väliltä 0 – 150 mm ja sen näyttötarkkuus on 0,1 mm.



KUVIO 2. Työssä käytetty työntömitta digitaalisella näytöllä

Mitutoyon Absolut Digimaticin mukana tuli myös datakaapeli (KUVIO 3), jonka avulla tulokset digitaaliselta näytöltä saatiin tallennettua tietokoneelle napin

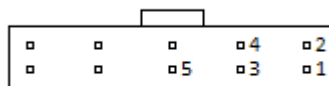
painalluksella. Työntömittaan tulevassa johdossa on 5 pinniä ja nappi, jota painamalla data lähetetään. Vasemmalta oikealle ensimmäisenä on maa, joka kytketään samaan maatasoon mikrokontrollerin kanssa. Toisena toimii DATA, jota pitkin työntömitan mittaamat tiedot lähetetään. Kolmantena on kellosignaali, joka tahdittaa datan lähetyksen oikealle hetkelle. Neljäs pinni ei ole kytketty tässä laitteessa mihinkään. Viimeinen eli viides pinni on request-pinni, joka maadoitettaessaan alkaa lähettää dataa 100 ms:n kuluttua. Kuviossa 4 nähdään, millainen on kaapelin toinen pää, joka tulee kiinni rakennettavaan laitteeseen.



KUVIO 3. Työntömittaan tuleva johdon pää



KUVIO 4. Rakennettavaan laitteeseen tuleva johdon pää



KUVIO 5. Laitteeseen kiinni tuleva johdonpään pinnien merkitykset

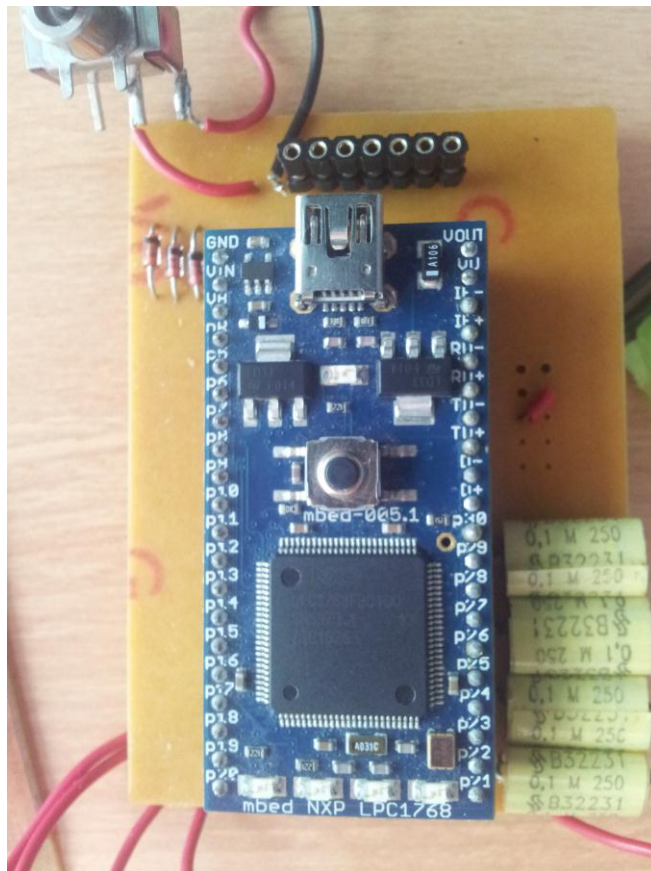
Kuvion 5 pinni 1 oikeassa alakulmassa on maa, joka yhdistyy työntömittaan tulevan johdonpään kanssa. Pinni 2 on yhteydessä DATA -pinniin ja pinni 3 on yhteydessä CLOCK:iin. Pinni 4 on yhteydessä maatasoon, kun työntömitassa olevasta johdonpäästä painetaan nappula pohjaan. Tämän takia pinnit 4 ja 5 yhdistetään sähköjohdolla.

3.2 Mbed-mikrokontrolleri

Työhön valittiin prototyyppiin ideaalinen laite: mbed-mikrokontrolleri, jonka ohjelmointikielenä toimii C++. Mbed:n tehokas prosessori mahdollistaa suuren laskentatehon ja sovelluksen toimivuuden. Se on kaikin puolin monipuolinen laite, jossa on yleisimmin käytetyt väylät ja toiminnot.

Mbed-mikrokontrolleri on mikroprosessorikehitysalusta, joka on tarkoitettu nopeaan, joustavaan sekä helppoon sovellusten kehittämiseen. Siinä on sisäänrakennettu USB-muisti, joka toimii mikrokontrollerin ohjelmointitiedostojen tallennukseen käytettävänä muistina ja on yhtä yksinkertainen käyttää kuin tavallinen USB-muistitikku. Ohjelmointi toteutetaan internetissä mbed:n omien kotisivujen Compiler-kääntäjällä, joka muuntaa kirjoitetun koodin binääritiedostoksi ja tallentaa sen mbed:n flash-muistiin. (Ford 2012.)

Mbed:ssä on ARM-pohjainen prosessori, joka toimii 100 Mhz:in taajuudella ja jossa on 512 kilotavua flash-muistia. Kyseinen mikrokontrolleri sisältää valmiit kirjastot, jotka mahdollistavat nopean ja helpon ohjelmoinnin. Kirjastot ovat tiedostoja, jotka sisältävät valmiita ohjelmointikäskyjä, muuttujia sekä muita ohjelmoinnille olennaisia määrittäjiä. (Ford 2012.)



KUVIO 6. mbed-mikrokontrolleri

3.3 Nokia LCD -näyttö

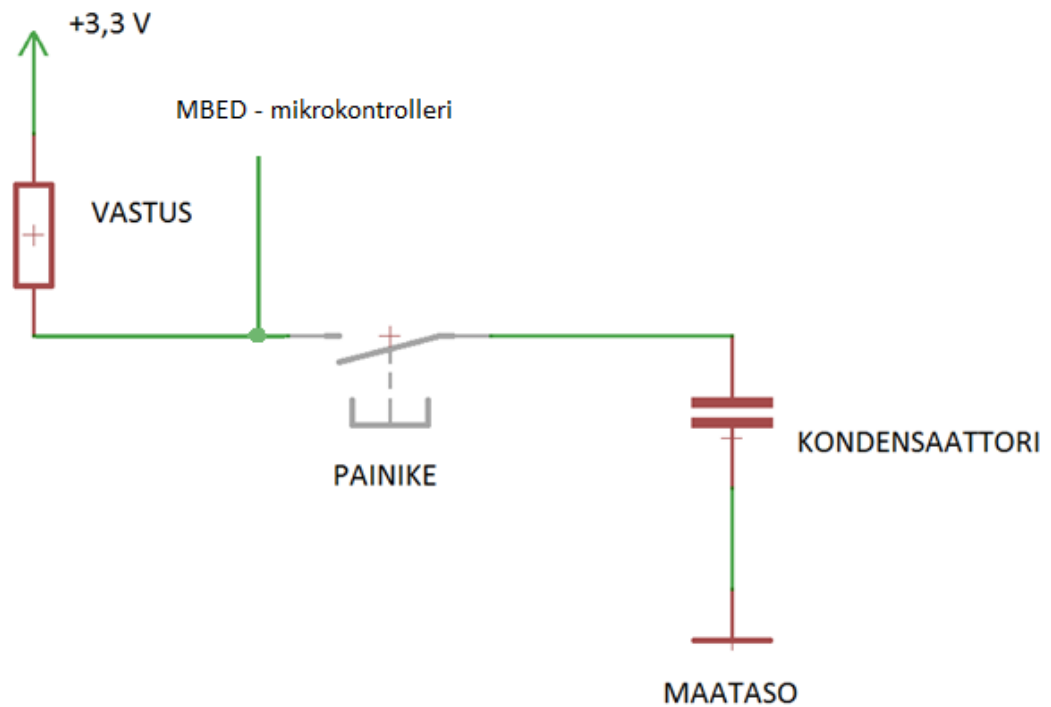
Tiedon esittämiseen käytettiin Nokian LCD-väri näyttöä. LCD-näyttö on nestekidenäyttö, jonka taustavaloina toimivat led-valaisimet. Halusin käyttää juuri kyseistä näyttöä, koska siihen mahtuu paljon tekstiä ja siihen oli valmis kirjasto mbed-mikrokontrollerissa. LCD-näytöllä on paljon mahdollisuuksia tulostaa haluttu tieto, esimerkiksi väritykset ja alleviivaukset.



KUVIO 7. Nokian LCD-näyttö

4 KYTKENTÄKAAVIOT JA NIIDEN TOIMINTA

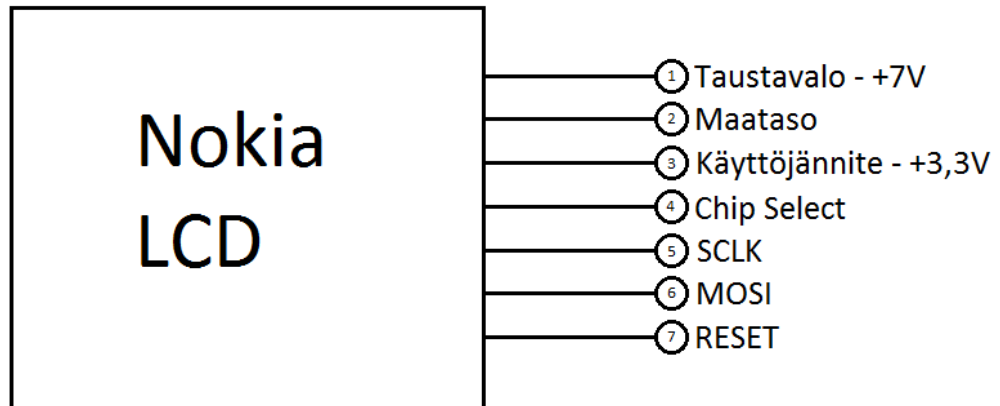
Laitteeseen tarvittiin neljä eri painiketta, jotta kaikki laitteen toiminnot saataisiin toteutettua. 15 k Ω :lla tehdään niin sanottu ylös veto, jonka avulla saadaan mikrokontrollerin pinnin jännitetaso nostettua noin 2,4 volttiin. On tärkeää ettei jännite nouse yli 3,3 volttiin, koska mikrokontrolleri saattaa vaurioitua muutoin. Ylös vetojännite otettiin mikrokontrollerin VREG-pinnistä, jonka jännite on noin 3,3 V. Kondensaattorilla estetään kytkinvärähtelyt painiketta painaessa. Kun painike painetaan pohjaan, mikrokontrollerin pinni maadoittuu ja tämän looginen taso on 0. Tämän avulla voidaan tarkkailla yksittäisten pinnien tasoa ja suorittaa eri ohjelmallisia toteutuksia.



KUVIO 8. Laitteen painikkeiden kytkentäkaavio

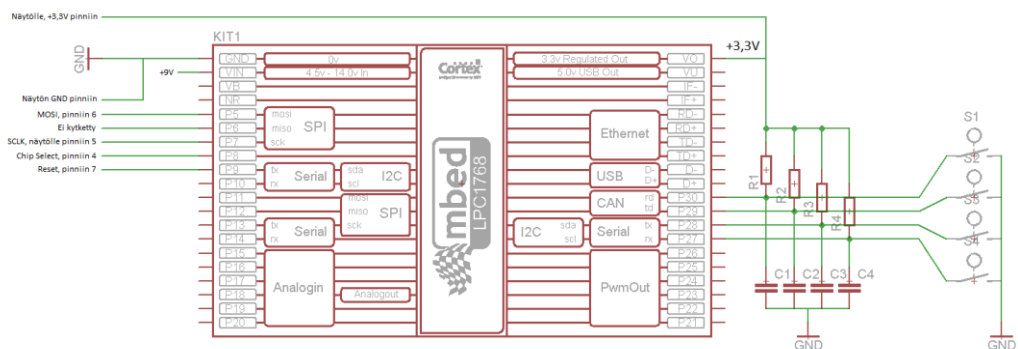
Nokian LCD-näyttö viestii mbed-mikrokontrollerin kanssa SPI-väylällä (KUVA 8). SPI on lyhenne sanoista Serial Peripheral Interface Bus. Se on väylä, joka koostuu kolmesta eri pinnistä: MOSI:stä, MISO:sta sekä SCLK:sta. MOSI:n (Master Out Slave In) toimintona on lähettää tietoa ulkopuoliselle laitteelle. MISO (Master In Slave Out) puolestaan toimii päinvastaisesti kuin MOSI, eli tämä väylä lähettää isäntä-laitteelle tietoa ulkopuoliselta laitteelta. Viimeisenä on SCLK

(Synchronous Clock), jonka tehtävänä on tahdistaa isäntä- ja ulkopuolisen laitteen kellotaajuudet. Tämä on yksi monista väylämahdollisuuksista, joita mbed tarjoaa.



KUVIO 9. Nokia LCD-näytön pinnit

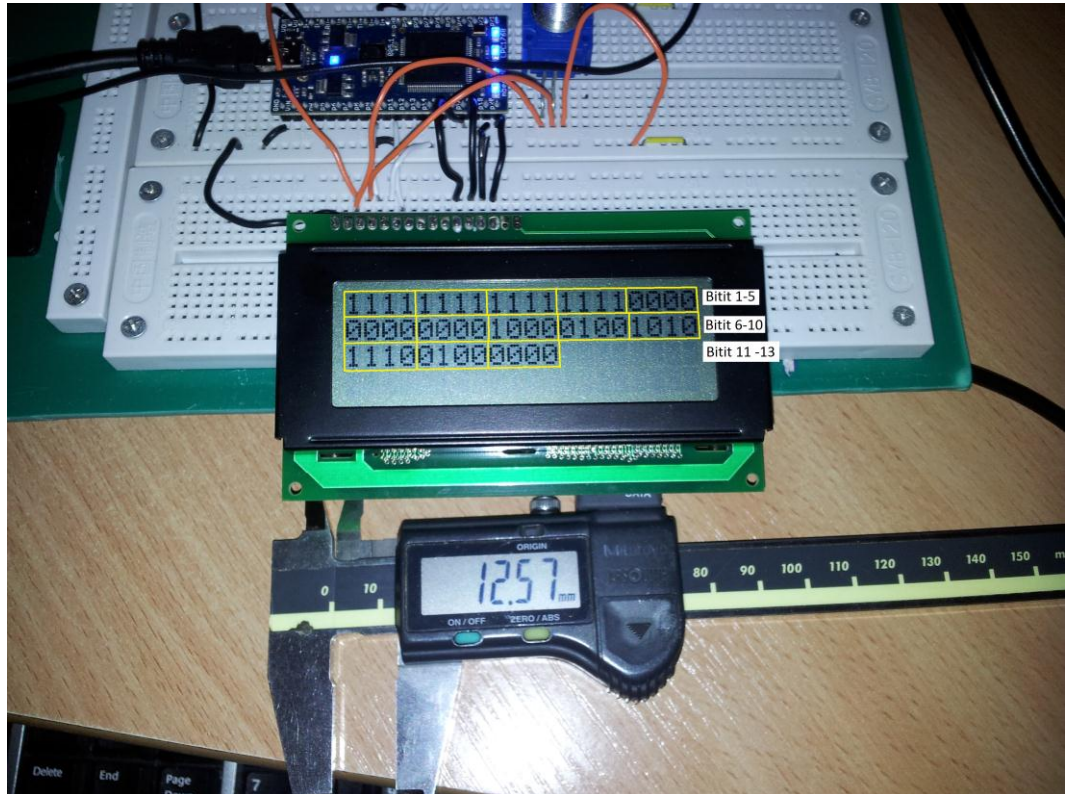
Mbed-mikrokontrolleri toimii laitteen sydämenä; se tekee kaiken oleellisen. Tulostukseen vaadittava näyttö liitetään kiinni mbed:iin Kuvion 9 osoittamalla tavalla. Itse mikrokontrollerissa on ennalta määritetyt pinnit, joihin näytön johtimet tulee liittää. Kuviossa 8 vasemmalla reunalla esitetään näytön käytettävät pinnit. Taustavalon jännite Nokian LCD-näyttöön otettiin ulkoisesta virtalähteestä, sillä mbed ei tuottanut +5 V:a suurempaa jännitettä. Työssä käytettiin ulkoisena virtalähteenä +9 V:n paristoa. Kuviossa 9 esitetään kokonaisuus painikkeiden ja näytön kytkentäkaaviosta.



KUVIO 10. KytKentäkaavio mbedistä, näytöstä sekä painikkeista

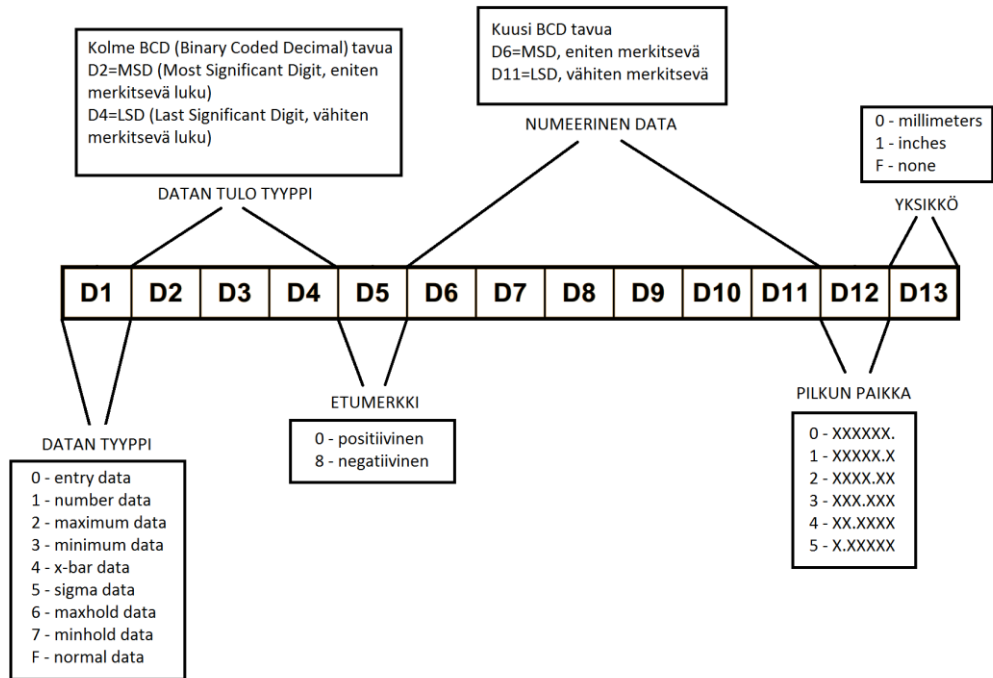
5 TYÖNTÖMITAN DATA

Tieto työntömitalta tulee sen omaa johtoa pitkin 13 puolitavun mittaisena jonona. Yksi puolitavu koostuu neljästä bitistä eli yhteensä bittejä tulee 52 kappaletta. Kuvio 11 nähdään, miten data muodostuu ja tulostuu näytölle.



KUVIO 11. Kuvassa työntömitan data binäärimuodossa

Kuvion 10 binäärilukujono sisältää työntömitan digitaalisella näytöllä olevan tiedon. Jokaisella bitillä on oma painoarvonsa tavussa ja biteistä muodostuu tavun sisältö ja merkitys (KUVIO 12).



KUVIO 12. Tavujen sisältämä data

5.1 Bittien painoarvo

Yksi puolitavu sisältää aina 4 bittiä. Jokaisella bitillä on oma painoarvonsa, ja esimerkiksi tavuissa 6-11 se määrittää jokaisen desimaalin numeerisen arvon. Kuviossa 10 työntömitan näytöllä on mitattu arvo 12,57 mm. Työntömitan yläpuolella on 4x20-merkkinen LCD-näyttö, jossa on tulostettuna luvun 12,57 mm binäärinen arvo.

Tavut 6-11 kertovat työntömitan näytössä olevan luvun binäärimuodossa.

Kuviosta 10 nähdään, että luvun 12,57 mm binäärinen arvo on 0000-0000-1000-0100-1010-1110. Koska bitit tulevat ulos vähiten merkitsevä bitti ensimmäisenä, ensimmäisen bitin painoarvo on 1, toisen bitin painoarvo on 2, kolmannen bitin painoarvo on 4 ja viimeisen bitin painoarvo on 8. Esimerkiksi tavujen 6-11 binäärijonosta 0000-0000-1000-0100-1010-1110 tulee 1257. Tavun 6 bittijono 0000 on nolla, koska siinä ei esiinny binäärilukua 1. Näin ollen myös seuraava bittijono eli tavu 7 on nolla. Kahdeksannen tavun arvoksi saadaan 1, koska $1+0+0+0$ tulos on 1. Yhdeksännestä tavusta eli 0100:sta saadaan bittien $0+2+0+0$ painoarvosta tulos 2. Tavun 1010 arvoksi saadaan painoarvojen mukaan $1+0+4+0$

eli 5. Viimeisen tavun bittien arvoksi saadaan $1+2+4+0$ eli 7. Näin ollen tavuista 6-11 saadaan arvot 1257.

5.2 Tavujen merkitys

Ensimmäinen puolittavu D1 määrittelee datan tyyppin. Työssä käytetyssä mallissa työntömitta antaa bittien arvon heksadesimaalilukuna FFFF eli 1111. Tavut 2-4 eli D2-D4 eivät niinkään olleet käytössä työssä, koska käytetyssä työntömitassa ei ollut erikoistoimintoja. Näin ollen ensimmäisten datojen arvo on aina FFFF. Työssä käytetyn mittalaitteen Mitutoyo Absolut Digimatic:in lisäksi myös muut Mitutoyo -laitteet hyödyntävät kyseistä esitystapaa, jonka vuoksi puolittavun 1 datatyyppiä on useita. (Mitutoyo 1997, 7.)

Tavu 5 eli D5 kertoo, onko tulos positiivinen tai negatiivinen. Datan arvon ollessa 1000 tulos on negatiivinen, kun puolestaan positiivinen tulos saatiin sen ollessa 0001. Tavut D6-D11 ovat mittauksen numeerinen arvo. Kuviosta 10 nähdään, miten tieto tulee binäärimuodossa työntömitalta, kun siihen on asetettu arvo 12,57 mm. (Mitutoyo 1997, 7.)

Pilkku määrittyy oikealle paikalle riippuen tavun 12 eli D12 bittien saamasta painoarvosta. Kuviosta 10 nähdään, miten pilkku sijoittuu, kun työntömitan arvona on 12,57 mm. Viimeisenä tavuna toimii D13, joka määrittää mitatun tuloksen mittayksikön. Riippuen asetuksista se voi olla joko mm (millimetriä) tai inch (tuumia). Suomessa käytetään millimetrejä eli työntömitan binäärinen arvo on 1000. (Mitutoyo 1997, 7.)

5.3 Datan muunnos

Data täytyy muuntaa, jotta se saataisiin tulostettua näytölle yksinkertaisemmassa muodossa. Seuraava kaava mahdollistaa mitatun tuloksen tulostamisen näytölle yksinkertaisemmassa muodossa, jossa painoarvot on laskettu yhteen ja pilkku asetettu oikeaan paikkaan. Esimerkissä käytetään sanaa LUKU muuttujana, johon tulos tallennetaan.

$$LUKU = \frac{(D11 + D10 * 10 + D9 * 100 + D8 * 1000 + D7 * 10000 + D6 * 100000)}{10^{D12}}$$

Kun kyseiseen kaavaan asetetaan aiemmin lasketut painoarvot tavuille D6-D11 ja pilkun painoarvo tavulle D12, saadaan tulos, jonka voi tulostaa näytölle yksinkertaisemmassa muodossa.

$$LUKU = \frac{(7 + 5 * 10 + 2 * 100 + 1 * 1000 + 0 * 10000 + 0 * 100000)}{10^2}$$

$$LUKU = \frac{1257}{100} = 12,57$$

5.4 Datan lähetys

Mitutoyon työntömitan mukana tullessa johdossa on nappula, jota painamalla tietoa lähetetään. Pinnit 4 ja 5 yhdistetään yksittäisellä sähköjohdolla, jotta pinni 4 on yhteydessä maatasoon datanappulan ollessa pohjassa. Nappia painamalla saadaan pinni 5 maatasoon yhdistettyä, mikä tarkoittaa datan lähetyksen alkavan 100 ms kuluttua. Kuvioista 13 nähdään, kuinka datan lähetys ajoittuu napin painalluksen kanssa. (Mitutoyo 1997, 5.)

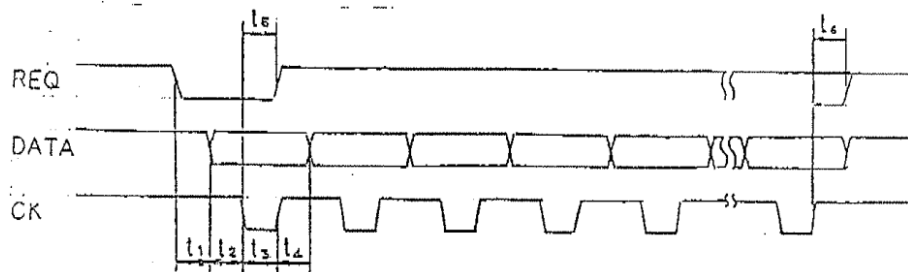


Fig.4.1

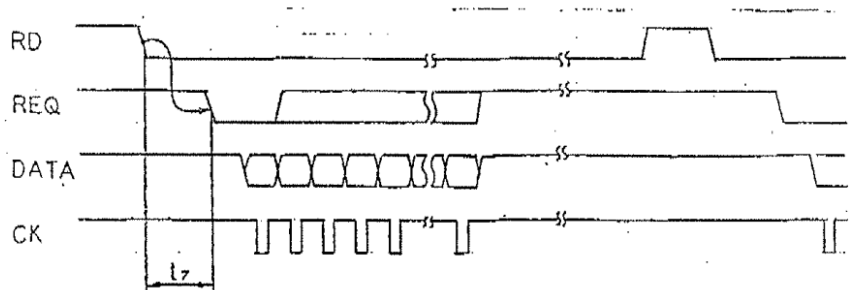


Fig.4.2

KUVIO 13. Tiedon lähetyksen ajoituskaavio

6 OHJELMALLINEN TOTEUTUS

6.1 Tallennus taulukkoon

Ensimmäinen tiedon tallennusvaihtoehto on datan tallentaminen 51-suuruiseen taulukkoon. Bittien saapuessa mikrokontrollerille ne tallennetaan taulukon soluihin, joista ne voidaan tarvittaessa hakea käsittelyyn. Koska bittejä on yhteensä 52 kappaletta, tarvitaan 52 solua. Taulukon määrittäminen C-kielellä tapahtuu seuraavasti.

```
DATA[51] = {0};
```

Ensimmäisenä on taulukon nimi, tässä esimerkissä se on DATA. Taulukon nimen jälkeen hakasulkeissa on taulukon koko, koska C-kielessä ensimmäinen muuttuja on 0, se merkitään hakasulkeisiin luvulla 51 52:n sijasta. Viimeisenä aaltosulkeissa määritetään solujen alkuarvo. Tässä esimerkissä alkuarvona on 0, joka tarkoittaa, että taulukon solut on alustettu ja arvoksi asetettu 0.

Seuraavaksi esimerkki koodista, joka datan saapuessa tallentaa bitin arvon taulukkoon nimeltä RAAKA.

```
void trigger() {                               //luodaan aliohjelma nimeltä
trigger, joka toimii datan tallennus aliohjelmana
    if (DATA == true) {                       //jos data on 1, silloin kirjoitetaan
taulukkoon paikkaan "n"
        RAAKA[n] = 1;
    } else {                                   //muuten jos data on 0, eli dataa ei tule,
kirjoitetaan 0
        RAAKA[n] = 0;
    }
    n++;                                       //n:n arvoa kasvatetaan aina yhdellä,
jotta saadaan koko 52 merkinen taulukko tayteen
    if (n == 52) {                             //jos n arvo saavuttaa 52
        muunnos();                             //kutsutaan muunnos aliohjelma, jossa
tehdään raakadatan muunnos
    }
```

6.2 Tallennus muuttujaan

Toisena vaihtoehtona datan tallennukseen on switch-case-rakenne, joka vertailee arvoja toisiinsa. Kyseistä menetelmää käytettiin työntömitan lisälaitteen ohjelmoinnissa. Koska bittejä on 52 kappaletta, tarvitaan 52 eri case-rakennetta, jossa vertaillaan saapuneen datan arvoa. Seuraavana näytetään esimerkki switch-case-rakenteesta. Esimerkissä tarkastellaan bittejä 17-20. Jos muuttuja DATA:n arvo on 1 eli dataa saapuu, asetetaan muuttujaan D5 arvo painoarvon mukaan. Bitit 17-20 esittää työntömitan etumerkin, yleensä se on positiivinen. Kaikki saapuvat bitit ovat nolliä eli muuttujan D5 arvoa ei muuteta ja se pysyy vakiona.

```
switch (TILAKONE) { //esitellään switch – case rakenteen nimi,
TILAKONE
    case 17: if(DATA){D5 = D5+1;} break;
    case 18: if(DATA){D5 = D5+2;} break;
    case 19: if(DATA){D5 = D5+4;} break;
    case 20: if(DATA){D5 = D5+8;} break;
}
```

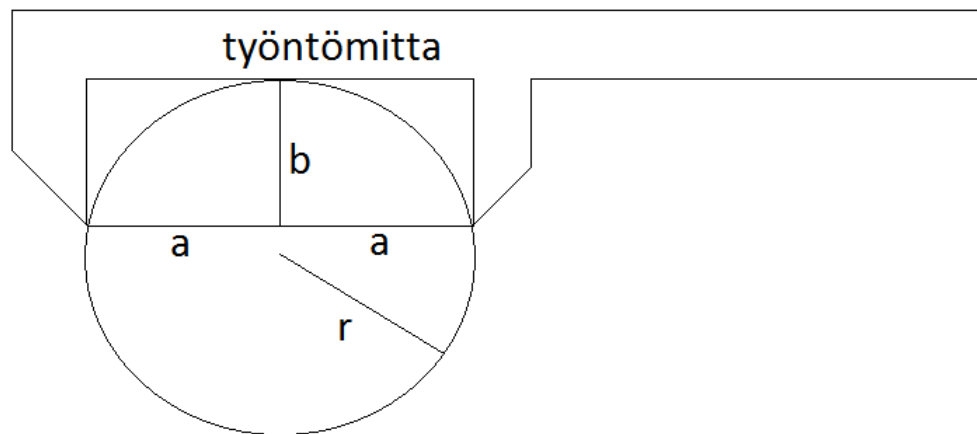
6.3 Painikkeiden ohjelmallinen toteutus

Jokaisella painikkeella on oma toimintonsa. Painikkeiden ohjelmallisessa toteutuksessa käytettiin mbed-mikrokontrollerin omia kirjastoja hyväksi. Mikrokontrolleri tarkkaili painikkeita, ja painettaessa yhtä painiketta pohjaan looginen taso muuttui nolllaksi. Kyseisen pinnan tason muuttuessa mbed toteutti erilaisia toimenpiteitä ohjelmassa. Painikkeita painamalla kutsuttiin joko eri aliohjelmaa, toteutettiin erinäisiä laskutoimituksia tai tulostuksia.

6.4 Matemaattiset toteutukset

Kaava halkaisijan ja säteen laskemiseen saatiin Mäntsälän Sahan toimitusjohtajalta. Kuvio 14 havainnollistaa alla näkyvää kaavaa, jolla mittaukset toteutettiin.

$$r = \frac{a^2 + b^2}{2 * b}$$



KUVIO 14. Säteen ja halkaisijan laskeminen työntömitalla

Halkaisijan laskeminen toteutuu laskemalla $r + r$. Ohjelmallisesti toteutettuna kaava näyttää seuraavalta.

```
SADE = ((pow(LUKU,2) + pow(b,2))/(2*b)); //lasketaan
Mantsalan-sahalta saadulla kaavalla sade, ja tallennetaan se
muuttujaan SADE
HALKAISIJA = SADE * 2; //ja halkaisijahan on 2 * SADE
muuttuja, se tallennetaan muuttujaan HALKAISIJA
```

6.5 Tulostus

Tulostuksen toteuttamiseen tarvittiin muutama toiminto. Jokaiselle napin painallukselle piti tehdä oma tulostuksensa LCD-näytölle, jotta selviää, mikä kyseisen painikkeen toiminto on. Pelkän tulostuksen yhteydessä ohjelma tekee muitakin toimintoja, kuten matemaattisia laskuja. Kaikki laitteen toiminnot

käytiin läpi Mäntsälän Sahan toimitusjohtajan kanssa, jotta saataisiin mahdollisimman tehokas kokonaisuus.

6.5.1 Nollaus, tallennus ja b-mitta

Nollauksen määritelmät olivat, että laitteen kaikki tiedot, lukuunottamatta välikappaleen b-mittaa, pystyttäisiin pyyhkimään uusia mittauksia varten. Laite voidaan nollata pitämällä nollauspainiketta pohjassa viiden sekunnin ajan. Jos painiketta ei pidetä pohjassa määrättyä aikaa, laite kehottaa painamaan painiketta pohjassa pidempään.

Mittatulosten tallennus sekä välikappaleen b-mitta olivat hyvin yksinkertaisia toteuttaa. Tallennuspainiketta painettaessa ohjelma haki tiedot johon ne oli tallennettu edellisissä aliohjelmissa. Seuraavassa esimerkissä tallennetaan taulukon i-soluun arvo, joka on ennalta laskettu. Lopussa i:n arvoa kasvatetaan yhdellä komennolla ++, jotta saadaan seuraava tallennettu arvo taulukon seuraavaan soluun.

TAULUKKO[i] = LUKU;

R[i] = SADE;

D[i] = HALKAISIJA;

i++;

Välikappaleen b-mittaus toteutettiin suorittamalla ensimmäinen mittaus ja painamalla b-mitan tallennuspainiketta. Mitta pysyi laitteen muistissa, kunnes virta katkaistiin kokonaan laitteesta.

6.5.2 Mittausten tulostus

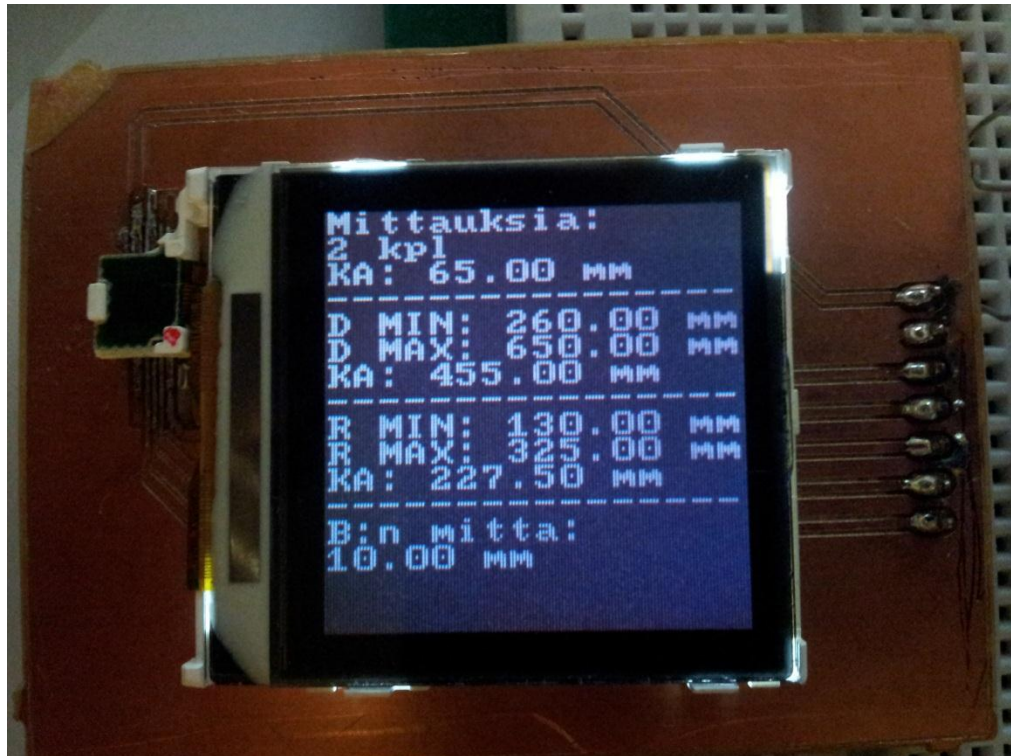
Mitattujen tulostusten ohjelmallinen toteutus oli haastavin osa tulosteista. Ennen graafisia tulostuksia jouduttiin hakemaan ennalta tallennettuja mittatuloksia taulukoista. Tämä toiminto voitiin toteuttaa for-silmukalla. Laitteen sovelluksessa for-silmukalla käytiin läpi kaikki taulukon solut ja vertailtiin niitä toisiin. Seuraavaksi esimerkki for-silmukasta ja taulukon solujen vertailusta.

```

for (n=0; n<=127; n++) {
    if (D[n] == 0) {
        break;
    } else if (D[n] <= D_MIN) {
        D_MIN = D[n];
    } else if (D[n] >= D_MAX) {
        D_MAX = D[n];
    }
}
}

```

Ylläolevassa esimerkissä käydään taulukon $D[n]$ solun arvoja läpi ja etsitään sieltä minimi- ja maksimiarvot. For-silmukan kaava on *for(alkuarvo; ehto, arvon kasvatus)*. Muuttujan n arvoa kasvatetaan yhdellä aina, kun koodi on käyty kertaalleen läpi. Näin toimitaan niin kauan kunnes arvo n saavuttaa 127. Ensimmäisenä tarkastellaan taulukon D solua 0. Arvon ollessa nolla *break*; käskyllä irtaudutaan for-silmukasta. Näin vältetään prosessorin laskentatehon käyttäminen turhaan taulukon jokaisen solun laskemiseen. Toisena on *else if*-lause. Jos D -taulukon solun arvo on pienempi kuin muuttuja D_MIN , tallennetaan taulukon D solun 0 arvo D_MIN -muuttujaan. Jos näin ei kuitenkaan tapahdu, mennään seuraavaan *else if*-lauseeseen, jossa tarkastetaan, onko taulukon solun arvo suurempi kuin D_MAX . Jos on, tallennetaan solun arvo muuttujaan D_MAX . Tässä esimerkissä for-silmukka laskee halkaisijan minimi- ja maksimiarvot, mutta samanlainen for-silmukka laskee myös säteelle ja normaalille mittatulokselle arvot. Kuvioista 15 nähdään, minkälainen lopullinen mittaustulosten tulostus näyttää.



KUVIO 15. Mittausten lopullinen tulostus.

7 TOIMINNALLINEN TOTEUTUS

Saatuani opinnäytetyön aiheen tarkoituksena oli aluksi tehdä tutkimustyötä siitä, olisiko mahdollista toteuttaa kyseinen sovellus. Tutkimustyö eteni vauhdilla, ja tulosta syntyi odotettua nopeammin. Sen vuoksi halusin koettaa, olisiko mahdollista toteuttaa myös laitteen toiminnallinen kokoaminen.

Lahden ammattikorkeakoulussa elektroniikkalaboratoriossa oli käytettävissä CNC-jyrsin, jolla pystyttiin jyrsimään piirilevyjä elektroniikkalaitteisiin. Käytin tätä laitetta hyväksi ja suunnittelin piirilevyn mbed-mikrokontrollerille sekä vaadittaville komponenteille.

Kokonaisuudessaan laitteen toiminnallinen toteutus ei ollut laaja (KUVIO 15). Haastavin osuus oli saada laitteelle kotelo. Hetken tutkittuani asiaa huomasin, että IP65-suojattu sähkörasiakotelo olisi juuri täydellinen laitteelle. Alkuperäinen suunnitelma oli asettaa näyttö piikkirimalla mbed:n päälle, mutta se osoittautui huonoksi vaihtoehdoksi. Lopulta liimasin näytön kanteen kiinni. Painikkeille porasin reiät kanteen ja näytölle oman aukon, jotta tulokset näkisi mahdollisimman hyvin.



KUVIO 16. Laite kokonaisuudessaan

8 YHTEENVETO

Laitteen toiminta ja käyttötarkoitus ovat hyvin yksinkertaisia. Sillä pystytään mittaamaan eri suuruisia halkaisijoita muutamasta kymmenestä senttimetristä jopa muutamaan metriin. Sovelluksen ideana on hyödyntää Antti Perttilän antamaa trigonometristä kaavaa, joka mahdollistaa mittauksia suurille halkaisijoille.

Työn alussa vaatimuksia laitteelta ei ollut paljoa. Kun työ eteni, vaatimuksia tuli lisää. Lähtökohtana oli tutkia, onko tällainen sovellus edes mahdollista toteuttaa. Ensimmäisenä piti tutkia, millaista dataa työntömitta syöttää ulos ja millaisessa muodossa. Seuraavaksi selvitettiin datan merkitys. Koska tutkimustyö eteni vauhdilla ja sain nopeasti selville datan muodon ja merkityksen, lisättiin sovellukselta vaadittavia toimintoja. Lukemat työntömitan digitaaliselta näytöltä tuli saada tulostettua erilliselle isommalle näytölle. Nämä valmistuivat odotettua nopeammin ja seuraavaksi haluttiin saada tarkempaa tietoa mitoista lisäämällä Perttilän kaava ohjelmaan. Tämän valmistuttua oli vuorossa suunnitella toimiva valikko ja mittatulosten tulosteet.

Laitteen toiminnallista toteutusta aloin miettiä, kun sovelluksen testaus koekytkentälevyllä onnistui ilman suurempia ongelmia. Ensimmäisenä piti suunnitella piirilevy mikrokontrollerille. Nokian LCD-näytölle oli valmis piirilevy, jonka sain koulun kautta. Laitteen koteloointi oli toiminnallisesta toteutuksesta haastavin. Tarkoituksena oli löytää mahdollisimman pieni kotelo ja asettaa laite koteloon ilman, että laite ei vaurioиду tai johtimet eivät mene oikosulkuun. Kokonaisuudessaan laitteen toiminnallinen toteutus onnistui. Harjoittelun ja muiden arkipäivien menojen takia toiminnallinen toteutus eteni odotettua hitaammin.

Aluksi saatuani tämän opinnäytetyön olin epävarma, olisiko minusta toteuttamaan sitä loppuun asti. Tutkimustyön alussa aihe näytti haastavalta ja vaikealta toteuttaa omiin tietoihini ja taitoihini verrattuna. Saatuaan lisätietoa aiheesta alkoi laitteen idea selkeytyä. Aloitin työni selvittämällä datan muodon ja merkityksen ja sen kuinka työntömitta itsessään toimii. Tähän prosessiin ei kulunut kauaa, ja melkein heti toisella viikolla pääsin suunnittelemaan itse ohjelmallista puolta laitteeseen. Työn alku eteni vauhdilla, koska työ oli mielestäni mielenkiintoinen.

Ohjelmallinen toteutus laitteelle oli mielestäni osuuden helpoin. Vaikka olin käynyt vain C++ -ohjelmointikielen peruskurssit, oli ohjelmointi yksinkertaista. Ohjelmallisessa toteutuksessa oli kaksi eri toteutusvaihtoehtoa: tallentaminen suoraan taulukoihin tai switch-case-rakenteella tallentaminen suoraan muuttujiin. Koska muuttujia piti pyöritellä useaan kertaan ohjelmassa, päädyin käyttämään switch-case-rakennetta ohjelmassani. Kokonaisuudessaan sovelluksen ohjelmallinen toteutus vei vain murto-osan koko opinnäytetyön ajasta ja yllätyin itsekin, kuinka helposti se toteutui.

Haastavin osa opinnäytetyöstä oli laitteen toiminnallinen toteutus ja se kuinka se asetetaan sille sopivaan koteloon. Tämän toteutukseen aikaa kului paljon, koska samaan aikaan minulla oli työharjoittelu, joka vei suuren osan ajastani. Onnistuin kuitenkin suunnittelemaan ja kasaamaan laitetta hiljalleen ja lopulta sainkin laitteen koteloitua.

Vaikka opinnäytetyön alku tuntui haastavalta, ei se sitä ollut. Ohjelmallinen toteutus sujui ilman suurempia ongelmia. Koteloinnissa tuli pieniä parannuksia viime hetkillä, jotta laite olisi mahdollisimman selkeä ja helppo käyttää. Kokonaisuudessaan opinnäytetyö oli antoisa kaikin puolin; opin C++ -ohjelmointikieltä runsaasti ja sain kokemusta käytännön työstä.

LÄHTEET

Perttilä, T. Mäntsälän Saha Oy [viitattu 7.6.2012.] Saatavissa:

Lähde: <http://www.mantsalan-saha.fi/>

Ford, S. 2012. Mbed microcontrollers [viitattu 7.6.2012.] Saatavissa:

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

Mitutoyo. 1997. Digimatic code output i/f specifications. Mitutoyo/Mti

Corporation [viitattu 3.4.2012.] Saatavissa: [http://www.green-](http://www.green-trust.org/digimatic/Digimatic%20interface.pdf)

[trust.org/digimatic/Digimatic%20interface.pdf](http://www.green-trust.org/digimatic/Digimatic%20interface.pdf)

