

Tallinna Tehnikaülikool
Infotehnoloogia teaduskond
Thomas Johann Seebecki elektroonikainstituut

Erki Koplímets
120786 IAEB

Puutetundliku LCD ekraani lisaplaad
Pelltechi pelletipõletile
Bakalaureusetöö

IEE40LT

Juhendaja: Mihhail Pikkov
Dotsent

Kaasjuhendaja: Martin Jaanus
Dotsent

Tallinn 2015

Autorideklaratsioon

Olen koostanud antud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd ning olulised seisukohad nii kirjandusallikatest kui ka mujalt on viidatud.

Käesolevat tööd ei ole varem esitatud kaitsmisele kusagil mujal.

Kuupäev:

Autor:

Allkiri:

Lõputöö ülesanne

Lõputöö teema: Puutetundliku LCD ekraani lisaplaat Pelltechi pelletipõletile

Touchscreen LCD Extension Board for a Pelltech Pellet Burner

Teema päritolu: Firma Pelltech OÜ

Lõputöö eesmärgid: Asendada praegune pelletipõleti LCD ekraan puutetundliku värvilise LCD ekraani lisaplaadiga. Panna paika projekti tingimused ning üldplaan. Valida parim võimalik riistvaraline lahendus ning selgitada põhikomponentide toimimist. Analüüsida saadud PCB-d ja tõstatada ülesanded edasiseks arenduseks.

Lahendatavad küsimused:

- Protsessori valik
- Puutetundliku LCD ekraani valik
- LCD ekraani juhtimine
- Puutetundliku ekraani kasutamine

Kuupäev:

Üliõpilane:

Juhendaja:

Kinnitaja:

Erki Koplímets

Puutetundliku LCD ekraani lisaplaat Pelltechi pelletipõletile

Bakalaureusetöö

Annotatsioon

Käesoleva lõputöö eesmärgiks on arendada Pelltechi pelletipõletitele puutetundliku LCD ekraani lisaplaat. See arendustöö hõlmab endas projekti nõudmiste sõnastamist, sobivate komponentide välja valimist, PCB plaadi disainimist ning protsessorit juhtiva programmikoodi kirjutamist.

Lõputöös on keskendunud ainult teatud arendustöö osadele nagu üldnõudmiste ning struktuuri kirjeldamine, protsessori, puutetundliku LCD ekraani valimine ja selle optimeeritud kasutamine, PCB üldnõudmiste paika panemine ja saadud plaadi analüüsimine, edasiste arendusülesannete tõstatamine.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35 leheküljel, 5 peatükki, 14 joonist ja 2 tabelit.

Erki Koplímets

Touchscreen LCD Extension Board for a Pelltech Pellet Burner

Bachelor Thesis

Abstract

The aim of this thesis is to develop a touchscreen LCD extension board for the Pelltech pellet burner. This development includes defining the project requirements, choosing the right components, designing the PCB and programming the processor software.

This thesis concentrates only on certain parts of the development process like describing the general project requirements and structure, picking the processor and touchscreen LCD and utilizing them, defining the requirements for the PCB, analysing the designed PCB and suggesting ideas for further development.

The thesis is in Estonian and contains 35 pages of text, 5 chapters, 14 figures and 2 tables.

Lühendite ja mõistete sõnastik

BMP (*Bitmap Image*)

PWM (*Pulse Width Modulation*)

UART (*universal asynchronous receiver/transmitter*)

EEPROM (*Electrically Erasable Programmable Read-Only Memory*)

PDI (*Program and Debug Interface*)

ADC (*Analog-to-digital converter*)

RTC (*Real-Time Clock*)

DMA (*Direct Memory Access*)

DAC (*Digital-to-analog converter*)

USB (*Universal Serial Bus*)

SPI (*Serial Peripheral Interface*)

TFT (*Thin-film-transistor*)

CS (*Chip Select*)

D/C (*Data/Command*)

RD (*Read*)

WR (*Write*)

TE (*Tearing Effect*)

CCx (*Capture Compare*)

SRAM (*Static random-access memory*)

Jooniste nimekiri

Joonis 1.1	Protsessori tarkvara struktuurskeem [18].	3
Joonis 1.2	Lisaplaadi tarkvara tööalgoritm.	4
Joonis 2.1	Diamex ALL-AVR® programmeatori PDI liides [29].	8
Joonis 3.1	SSD1963 graafikadraiveri juhtotsad [26].	10
Joonis 3.2	SSD1963 graafikadraiveri paralleelliidese andmeformaadid [26].	10
Joonis 3.3	LCD ekraani lintkaabel, nulltakisti asukoht.	12
Joonis 3.4	RGB formaat 656 [24].	13
Joonis 3.5	ILI9488 graafikadraiveri kuvatava ekraani suuna registrid [24].	14
Joonis 3.6	BitFontCreator Pro 3.5 andmeformaadid [16].	15
Joonis 3.7	LCD taustvalgustuse LED-ide paralleelühendus [25].	15
Joonis 3.8	<i>Dual-slope Pulse Width Modulation</i> [19].	16
Joonis 4.1	4-juhtmelise Takistusliku Puuteekraani ehitus [30].	17
Joonis 4.2	Ostsilloskoobi pildid.	19
Joonis 4.3	Puuteekraani kalibreerimispunktid [23].	21

Tabelite nimekiri

Tabel 3.1	ILI9488 graafikadraiveri kommunikatsiooniliidese valik [24].	12
Tabel 4.1	4-juhtmelise Puuteekraani skaneerimine [23].	18

Sisukord

Sissejuhatus	1
1. Plaadi vajalikud funktsionaalsused.....	2
1.1 Üldised projekti nõudmised.....	2
1.2 Protsessori tarkvara skeemid	3
2. Protsessor	5
2.1 Protsessori valiku kriteeriumid.....	5
2.2 Protsessori perekonna valik.....	5
2.3 Xmega® protsessori seeria valik.....	6
2.4 Xmega® D seeria mudeli valik	7
2.5 Prototüüp protsessor	8
2.6 Programmaatori valik	8
3. LCD ekraan	9
3.1 Ekraani valiku kriteeriumid.....	9
3.2 Prototüüp LCD ekraan.....	10
3.3 Valitud LCD ekraan	12
3.4 LCD Ekraani Juhtimine.....	14
4. Puutetundlik ekraan.....	17
4.1 Ülesehitus	17
4.2 Implementatsioon	18
4.3 Puute koordinaatide leidmine ADC tulemusest	18
4.4 Kalibratsioon	21
4.5 Erinevad puudutuste tüübid.....	22
5. PCB	23
5.1 Nõuded PCB-le.....	23
5.2 Disainitud PCB analüüs.....	24
Kokkuvõte	25
Kirjandus	26
Lisad.....	28

Sissejuhatus

Firma Pelltech tegutseb 2005. aastast ja tegeleb pelletipõletite tootmise ning müügiga [1].Erinevaid põletite mudeleid on palju, alates väikestest eramajadele mõeldud 20 kilovatistest kuni suurte tehastes kasutatavatele 1000 kilovatisteni. Kõigi mudelite puhul on väga oluline näidata kasutajale reaajas informatsiooni põletis toimuvast ning anda võimalusi kogu protsessi vastavalt vajadusele konfigureerida. Hetkel on pakutavaks kasutajaliideseks väikeste põletite puhul kahe realine värvideta LCD ekraan (Lisa 1). Suurte põletite puhul on LCD ekraan nelja realine. Mõlema juhtimine toimub nelja nupu abil (Lisa 2). See tekitab olukorra, kus oleks hädavajalik näidata kasutajale väga mahukat informatsiooni, kuid see pole lihtsalt ekraani suuruse tõttu võimalik.

Kasutajaliidese kvaliteedi parandamiseks otsustati vahetada vana ekraan värvilise puuetundliku LCD ekraani vastu, mis teeks põleti juhtimise paremini mõistetavamaks ning lihtsamaks. Lisaks asendaks puuetundlik ekraan kohmaka nupustiku. Kuna uus riistvara peab ühilduma vanade põletitega siis ei saa otse praeguse põleti külge ekraanipistikut teha. Lahenduseks on disainida uus plaat nii, et seda on võimalik põleti külge ühendada lisamoodulina.

Lisaplaadi eeliseks on võimalus peale LCD ekraani ka muid funktsionaalsusi lisada. Laienduse abil peaks saama pelletipõletiga kontrollida katelt ning ka kodu üldist keskkütet. Internetimooduli lisamisega on võimalik kõiki andmeid edastada internetti ning sealt ka käskusi saada. Seeläbi on võimalik luua nii arvutis kui ka nutitefonis kasutatav põleti kasutajaliides.

Lõputöös keskendutakse projekti üldiste nõudmiste ja struktuuri sõnastamisele, mille põhjal arendustööd alustada. Välja valitakse lisaplaadi tähtsamad elemendid nagu protsessor ning puuetundlik LCD ekraan ja kirjeldatakse nende juhtimisprogrammi sisu. Töötava tulemuse saavutamiseks esitatakse PCB nõudmised Pelltechi elektroonikule, kes vastava plaadi valmis disainib. Tulemust analüüsitakse ning hinnatakse, kas see vastab nõuetele ning mida tuleks parandada või juurde lisada.

Tarkvara kirjutamise keeleks on valitud C keel, sest see on optimaalne kasutuseks väikestel protsessoritel. Lisaks on see ainus keel, milles programmeerimiseks TTÜ elektroonika eriala tudengeid ette valmistatakse.

1. Plaadi vajalikud funktsionaalsused

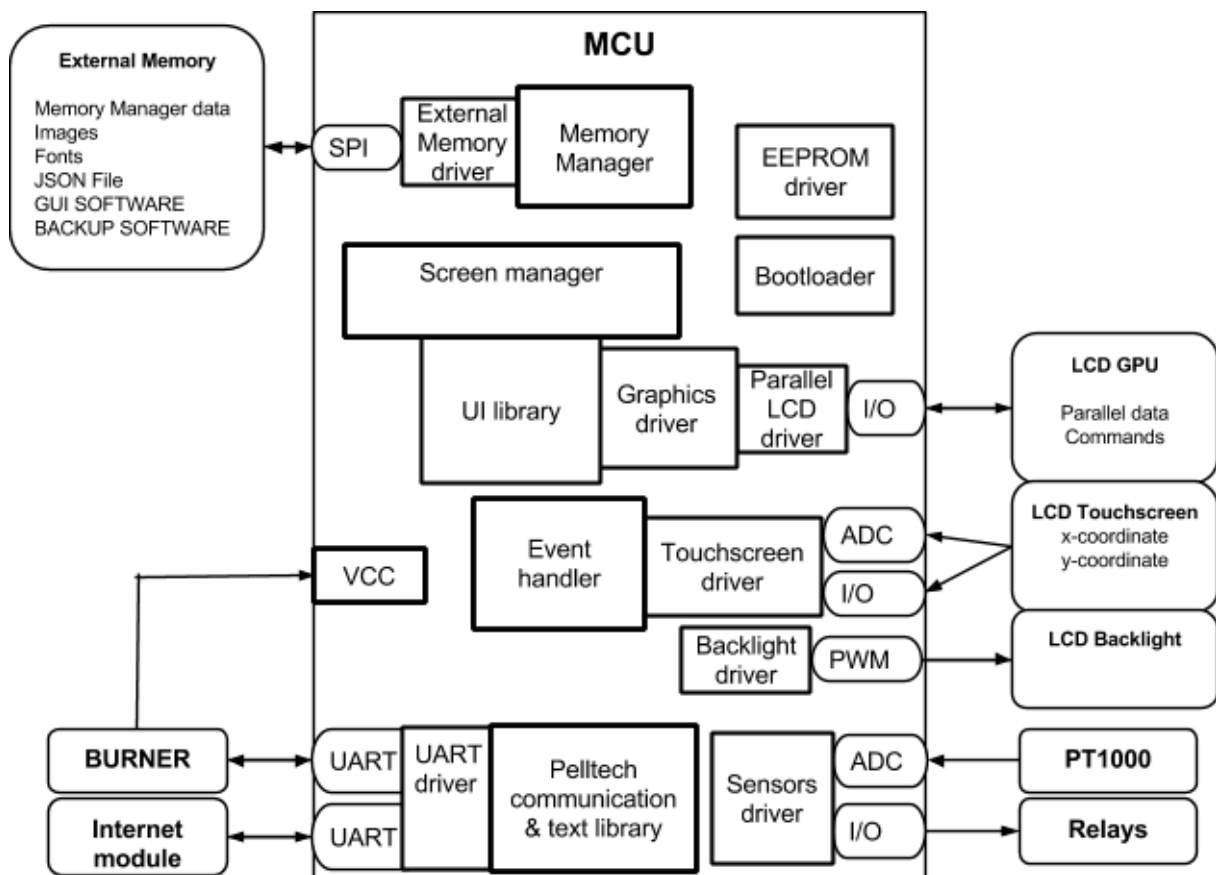
1.1 Üldised projekti nõudmised

- Lisaplaat peab täielikult asendama hetkel kasutuses oleva pelletipõleti LCD ekraani. Uuel LCD ekraanil kuvatud info peab sisaldama kõike, mida praegune LCD ekraan kuvab. Lisaks ka täiendav informatsioon.
- Informatsioon lisaplaadil peab vastama täielikult pelletipõleti protsessoris sisalduvale infole ning ei tohi olla sünkroonist väljas. Pelletipõleti reaalaaja informatsioon ei tohi hilistuda lisaplaadi ekraanil ega ka lisaplaadi juhtimissüsteemis rohkem kui sekundi võrra.
- Lisaplaadi ekraanil peab olema kuvatud internetimooduli informatsioon. Selle mooduli parameetrid peavad olema sisestatavad puuetundliku ekraani kaudu.
- LCD ekraani taustvalgustus peab töötama erinevate helendusastmetega. Lisaplaat peab olema võimeline kasutama LCD ekraani taustvalgustuse juhtimiseks tarkvaraliselt muudetavat PWM (*Pulse Width Modulation*) signaali.
- LCD ekraan peab toetama vähemalt 16-bitiseid värve. Graafikaliides peab suutma ise joonistada ekraane, mis sisaldavad nuppe, tekste ja pilte. Peab toetama BMP (*Bitmap Image*) pildiformaati [2]. Tekstid peavad olema kuvatavad erinevates fontides ning need fondid peavad olema vahelduva laiusega ning peavad toetama UTF-8 tähe kodeeringut [32].
- Lisaplaat peab omama välist mälu tekstifontide, piltide, lisaplaadi protsessori programmikoodi ning ka selle programmikoodi varukoopia jaoks. See mälu peab olema internetimooduli kaudu uuendatav. Lisaplaadi protsessori programmikood peab olema mittesõltuv välise mälu jaotusest ehk failidele ligipääs ei tohi käia mitte välismälu oleva aadressi vaid muu identifikaatori abil. See tähendab, et on vaja failisüsteemi olemasolu. Välise mälu uuendamine peab toimuma sarnaselt protsessori programmikoodi uuendamisega ning kasutama Intel Hexi failiformaati [17].
- Kasutajaliides peab olema tõlgitud mitmesse keelde. Ekraanidel kuvatavad tekstid peavad olema salvestatud JSON failina välisesse mällu, et neid saaks uuendada ilma lisaplaadi tarkvara muutmata [31].
- Lisaplaat peab olema programmeeritav internetimooduli kaudu. Lisaks peab saama internetimooduliga programmeerida ka pelletipõletit. Mõlemad programmeerimisprotokollid peavad kasutama Intel Hexi failiformaati [17].
- Puuetundlik ekraan peab asendama pelletipõleti küljes oleva nupustiku. Kasutajaliideses liikumine ning andmete sisestamine peavad toimuma puuetundliku ekraani abil. Kasutajaliides peab olema kasutatav töökeskkonnas. See tähendab, et on võimeline töötama väga tolmuses keskkonnas ning kasutatav töökindaga või ka näiteks krivikeeraja käepidemega.
- Lisaplaat peab kasutama Pelltechi kommunikatsiooniprotokollide pelletipõleti ja internetimooduliga suhtlemiseks [36]. Kõik pelletipõleti välja saadetud sõnumid peab lisaplaat edastama ka internetimoodulile. Internetimooduli poolt saadetud sõnumeid tuleb lisaplaadis analüüsida ning selle põhjal otsustada, kas sõnum tuleb pelletipõletile edasi saata või saadud info on mõeldud ainult lisaplaadile. Lisaplaat peab saatma enda infosõnumeid internetimoodulile ning saatma pelletipõletile selle tööks vajalikku lisaplaadi infot.
- Lisaplaat peab andma märku pelletipõletiga ühenduse puudumisest, probleemidest pelletipõletis, ühenduse puudumisest ja probleemidest internetimooduliga ning ka probleemidest lisaplaadi toimimises endas. Seda kas LCD ekraani abil või kui võimalik sõnumitega internetimoodulile.
- Lisaplaadi külge peab saama ühendada kindlaid sensoreid, mille abil teostatakse süsteemide juhtimist. Selleks otstarbeks peab plaat suutma juhtida vooluvõrguga ühendatud releesid.
- Lisaplaat peab saama kogu toite üle UART (*universal asynchronous receiver/transmitter*) ühenduse pelletipõletiga [27]. Peab suutma töötada ka üldise vooluvõrgu katkemise korral. Seda kasutades pelletipõletit küljes olevat akut.
- Internetimoodul peab saama kogu toite üle UART ühenduse lisaplaadiga ning töötama ka vooluvõrgu katkemise korral. Seda kasutades pelletipõletit küljes olevat akut.

- Lisaplaadi konfiguratsioonid peavad säilima ka kogu toite kadumise korral ehk protsessori väljalülitumisel. Selle jaoks peab olema võimalik kasutada protsessori sisemist EEPROM-i (*Electrically Erasable Programmable Read-Only Memory*).

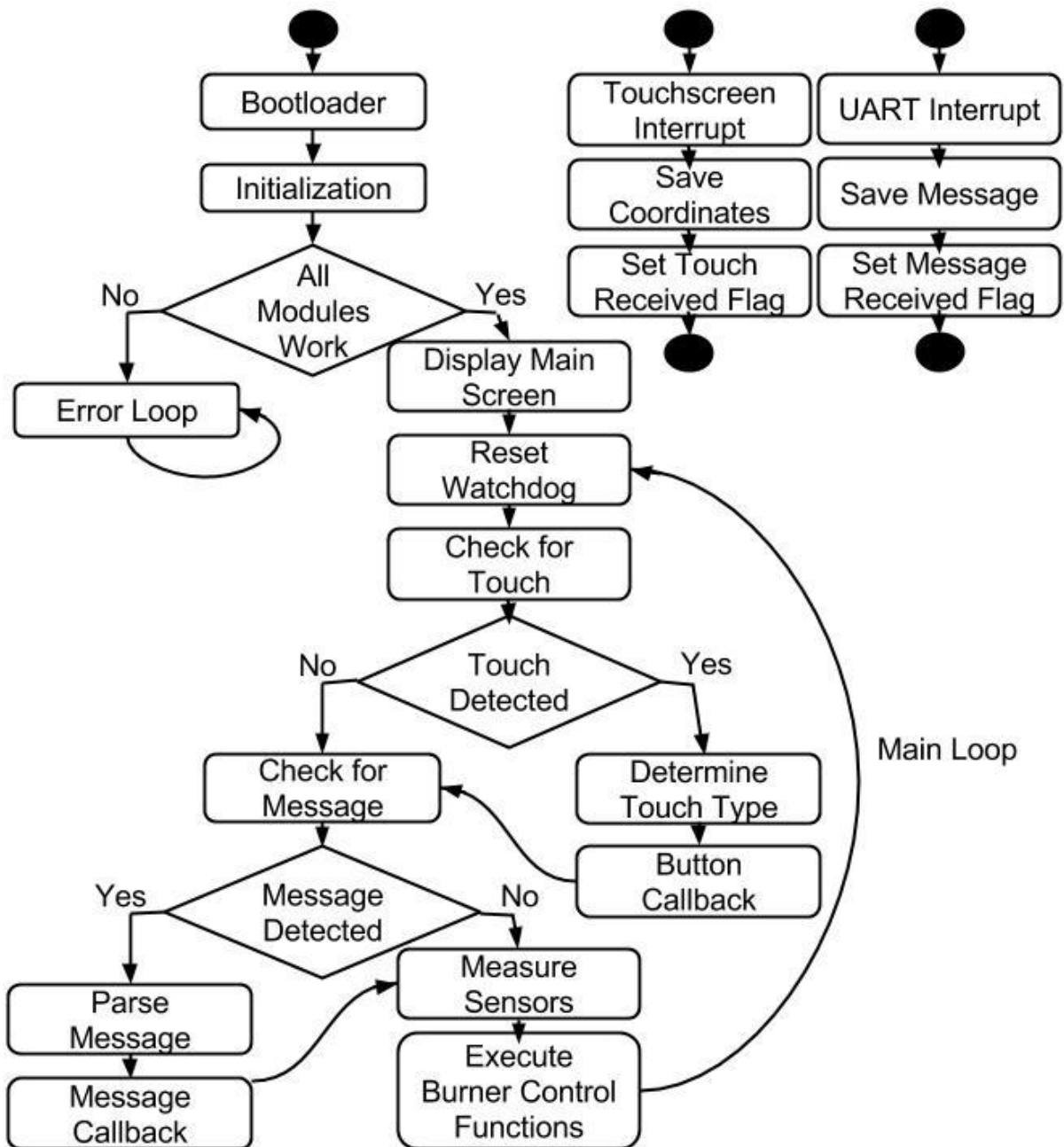
1.2 Protsessori tarkvara skeemid

Tarkvara ning riistvara disaini lihtsustamiseks on väga kasulik teha kogu projekti struktuurskeem. See hõlmab blokkdiagramme näitamaks, kuidas on seotud erinevad süsteemi osad. Keskne koht on protsessoril ning selle ümber on tema külge ühendatud moodulid. Väliste moodulite puhul on vajadusel blokkdiagrammina välja toodud ka erinevad aspektid, millega peab protsessor nende puhul tegelema. Lisaks on välja toodud tähisega VCC protsessori toiteallikas. Protsessori bloki sisse on märgitud, millise sisend/väljund struktuuri külge on väline moodul ühendatud. Lisaks on nooltega defineeritud info liikumise suunad. Iga sisendi/väljundiga tegelemiseks on välja toodud ka tarkvara hierarhiline struktuur, mis umbkaudselt defineerib ära, milliseid erinevaid tarkvara mooduleid tuleb luua. Struktuurskeemis on ka esile toodud protsessori sisemised tarkvaramoodulid, mis pole otseses seoses väliste moodulitega (Joonis 1.1).



Joonis 1.1 Protsessori tarkvara struktuurskeem [18].

Tarkvara programmeerimise ülesande ning ka lisaplaadi üldiseks mõistmiseks on kasulik teha tööalgoritm (Joonis 1.2). See seletab ära programmi kulgemise ning näitab oluliste otsuste kohad. Eraldi on välja toodud põhi programmi käik (*Main*) ning lisaks ka *Interruptid*. Tegu on väga üldistava pildiga, mis jätab välja detailsused.



Joonis 1.2 Lisaplaadi tarkvara tööalgoritm.

2. Protsessor

2.1 Protsessori valiku kriteeriumid

Projekti keskne ja tähtsaim osa on protsessor. Selle valik sõltub paljudest parameetritest:

- Eeltoodud struktuurskeemi (Joonis 1.1) funktsionaalsuse tagamine.
- Piisavalt odav hind masstootmise jaoks.
- Mugav ja kaasaegne arenduskeskkond.
- Lihtne programmeatori kasutamine.
- Firmale sobiva ning tuttava ülesehitusega.
- Hea dokumentatsiooni tase.

Pelletipõletite juhtautomaadina kasutatakse Atmeli® AVR® arhitektuuri [3] kuuluva megaAVR® perekonna protsessoreid Atmega32, Atmega64 ja Atmega128 [4]. Sellest sõltuvalt oli ülemuse poolt määratud ka tingimus, et uuel plaadil kasutatakse Atmeli® AVR® arhitektuuri protsessorit. Põhjuseks, et Atmeli® firma usaldusväärsus on ette teada, ostmise ja tarnimise kohad on Atmeli® protsessorite jaoks Pelltechil olemas, siseehitus ja tarkvaraline kasutus on AVR® arhitektuuri protsessoritel üsna sarnane.

Atmeli® protsessorite suureks eeliseks on mugava kasutajaliidesega integreeritud arendusplatvormi Atmel Studio 6® olemasolu [5]. See on pidevalt uuendatav programm Assembleri, C või C++ programmeerimiskeeles Atmeli® mikrokontrollerite programmeerimiseks ja debugimiseks. Arenduskeskkond pakub lisaks ka mugavat programmeatori liidest. Kasutades Atmeli® toega programmeatorit saab otse arenduskeskkonnast programmikoodi protsessorile peale laadida. Ainsad tingimused on eelnev draiverite installeerimine arvutile ning õige programmeatori PDI(*Program and Debug Interface*) liidese ühendus protsessoriga [21].

Atmeli® protsessorite suureks plussiks on väga mahukas dokumenteeritus. See on märksa põhjalikum kui teistel mikroprotsessoridel müüvatel firmadel ning ei eelda ka kasutajalt erilisi eelteadmisi. Pakutavad funktsionaalsused on andmelehes pikalt lahtiseletatud ning lisatud on ka pilte, skeeme ja tabeleid. Lisaks on Atmeli® kodulehel peale protsessorite andmelehtede ka eraldi pikemad õpetused iga funktsionaalsuse kasutamise kohta. Lisaboonusena sisaldavad need ka näitekoodi, mida lubatakse ilma mingite piiranguteta kopeerida ja kasutada. Seda isegi kommertsprojektides.

2.2 Protsessori perekonna valik

Protsessori valik tuli teha Atmel AVR® arhitektuuri kuuluvate mikroprotsessorite seast. Sinna kuuluvad perekonnad on järgmised [3]:

- **TinyAVR®**[6] - Sellel perekonnal on 0,5 kuni 16 kB programmimälu, piiratud sisend/väljund võimalused ning ka piiratud sisemised funktsionaalsused.
- **MegaAVR®**[4] - 4 kuni 512 kB programmimälu, suur valik sisend/väljund võimalusi ning ka suur valik sisemisi funktsionaalsusi.
- **XMEGA®**[7] - 16 kuni 384 kB programmimälu, suur valik sisend/väljund võimalusi ning veelgi laiendatud funktsionaalsused.
- **32-bit AVR®**[8]- Selle perekonna struktuur on päris erinev ülejäänud sama arhitektuuri kuuluvatest protsessoritest. Omab eriti laiendatud funktsionaalsust, sealhulgas ka audio ning video protsesseerimisvõimalusi.

Valikust langeb välja TinyAVR®, sest see on mõeldud väiksemate projektide jaoks ning pakutav programmimälu suurus pole kindlasti piisav. Veel langeb välja 32-bit AVR®, sest selle arhitektuur erineb liiga palju Pelltechis juba kasutusel olevatest protsessoritest ning lisaks puuduvad vajadused nii laiapõhjalise funktsionaalsuse jaoks.

Xmega® ja MegaAVR® vaheline valik on palju spetsiifilisem, sest tegu on päris sarnaste protsessorite perekondadega. Valik langeb Xmega® kasuks järgmistel põhjustel:

- Xmega® on Atmeli® uuem ja täiustatum verisoon MegaAVR-ist®.
- Xmega® omab 12-bitiseid täpsemaid ADC (*Analog-to-digital converter*) sisendeid võrreldes MegaAVR® 10-bitistega [9].
- Xmega® protsessori maksimum sisemise kella kiirus 32MHz [7] on suurem kui MegaAVR-i® 16 MHz [22].
- Xmega® perekonna hind on üldiselt madalam kui MegaAVR® oma, näiteks suurem 128 kB programmimäluga XmegaA128 hind on võrdne 32 kB Atmega32 hinnaga.
- Xmega® omab suuremat valikut funktsioone sisend/väljund portidele ning sealhulgas laseb neid paindlikumalt kasutada ning vajadusel ka ümber kaardistada [9].
- Xmega® omab palju suuremat sisend/väljund portide alternatiivfunktsioonide hulka [9].
- Xmega® pakub ka sisemisi funktsionaalsusi, mida MegaAVR-il® pole. Näiteks *Interrupt controller*.

2.3 Xmega® protsessori seeria valik

Spetsiifilisema Xmega® protsessori seeria valikul järgiti järgmisi vajadusi:

- Peab omama varuga programmimälu, et ära hoida potentsiaalselt esinevad probleemid selle mälu liigsel täitumisel.
- Peab omama piisavalt palju sisend/väljund porte, kuid samas hoida see arv ruumi ning kasutuse lihtsuse mõtteis madalaim.
- Seeria peab omama kindlaid kommunikatsiooni, mälu, *interrupt*, ADC, RTC (*Real-Time Clock*), taimerite ja *bootloaderi* funktsionaalsusi.

Xmega® perekonna mikroprotsessorid jagunevad viite tähega tähistatud seeriasse [7]:

- **A** - Kõige võimsam ja kallim seeria, omab kõiki selle perekonna pakutavaid funktsionaalsusi ning kasutatakse eelkõige eriti nõudlikes aplikatsioonides.
- **B** - LCD ekraani juhtimise eriseeria, omab enamus funktsionaalsusi ka A seeriast. See toetab kahevärvilisi LCD ekraane, mis on segmenteeritud sisendiga. Kuni 160 erinevat juhitavat segmenti.
- **C** - Sellel seerial puuduvad paljud A seeria spetsiaalfunktsionaalsused nagu DMA (*Direct Memory Access*) kontrollid, dual 12-bit ADC, 12-bit DAC (*Digital-to-analog converter*), krüptograafia moodul ning ka laiendatud kommunikatsiooni võimalused. Sellel seerial on olemas sisseehitatud USB (*Universal Serial Bus*) moodul.
- **D** - Sarnane C seeriale, kuid puudub ka USB moodul.
- **E** - Spetsiaalne väiksemõõtmeline seeria, kasutuseks piiratud mõõtmetega eriprojektides.

E seeria ei sobi, sest selles projektis pole vajadust iga hinna eest väikestest mõõtmetest kinni pidada. A seeria protsessori puhul on näha, et jääks kasutamata mitmeid pakutavaid mooduleid nagu DMA kontrollid, 12-bitine DAC või krüptograafia moodul. B seeria protsessor ei sobi, sest see on mõeldud kahevärviliste LCD ekraanide juhtimiseks, millel on segmenteeritud sisendid. Projekt näeb ette värvilist puutetundlikku ekraani. C seeria protsessori puhul on üleliigne USB moodul, mille vajadus projektis puudub. D seeria Xmega® protsessor on kõige sobivam valik, sest sellel on kõige vähem liigseid ebavajalikke funktsionaalsusi, piisavalt sisend/väljund porte ning hind on teiste seeriatega võrreldes kõige sobivam.

2.4 Xmega® D seeria mudeli valik

D seeria protsessoreid on saadaval erinevates programmimälu suurustes 16 kB-st kuni 386 kB-ni. Lisaks on seeria jaotatud kaheks mudeliks [7]:

- **Number 3** - kahest suurim mudel, omab 64 sisend/väljund PIN-i.
- **Number 4** - väiksem 44 sisend/väljund PIN-iga mudel.

Arvestades projekti vajadusi tuleb valida suurem mudel, sest 44 sisendist/väljundist ei piisa. Edasise arenduse jaoks on ka kindlam jätta piisavalt vabu sisend/väljund PIN-e.

Programmimälu suuruse valik tuleneb erinevate versioonide hindadest ning ka kogemustest praegu Pelltechis kasutusel olevate MegaAVR® protsessorite mälu suurustega. Probleeme tekib 32 kB-ise mäluga. 64 kB mälu on küll piisav, kuid mitte suure varuga tuleviku arendusteks. Järgmine aste 128 kB omab ka piisavalt varumälu ning ka hinnavahe 32kB ja 64 kB suhtes pole suur. Järgmine suurem versioon 192 kB programmimäluga omab juba tuntavat hinnavahet.

Lõplik otsus protsessori valiku osas on Xmega128D3. Sellel on järgmised vajalikud omadused [20]:

- Piisavaks programmimäluks on 128 kB.
- EEPROM on 2 kB.
- SRAM (*Static random-access memory*) on 8 kB.
- *Bootloader* on 8 kB.
- Protsessori kõigi PIN-ide arv on 64.
- Maksimum protsessori taktsagedus on 32 MHz.
- Protsessori sisendi/väljundiPIN-ide arv on 50.
- Programmeeritav mitmetasandiline *interrupt controller*.
- Üks 16-bitine RTC.
- Viis 16-bitist taimerit.
- Kuni kolm UART moodulit.
- Kuni kaks SPI(*Serial Peripheral Interface*)moodulit.
- Üks 16-bitise kanaliga 12-bitise täpsusega ADC.
- *Watchdog* taimer.

2.5 Prototüüp protsessor

Prototüüpimiseks kasutati projektis Xmega128A1 protsessorit. See protsessor oli osa MikroElektronika® firma poolt müüdavast Xmega-Ready® arendusplaadist [10]. Sellele plaadile on juba algselt müüja poolt peale pandud UART protokolliga kommuniqueeriv *bootloader*, millega saab protsessorile programmikoodi üles laadida ilma eraldi programmaatorita. Protsessori UART moodul on plaadil ühendatud USB-UART integraallülitusega, millega saab arvuti USB pordiga ühenduda ning kasutada MikroElektronika® tarkvara programmikoodi protsessorile laadimiseks. Arendusplaadil on välja toodud duubeldatult kõik protsessori sisend/väljund PIN-id ning lisaks ka 3,3 voldi ning 5 voldi otsad väliste moodulite toiteks. Prototüübi testimisel tuleb meeles pidada Xmega128A1 ja lõpliku protsessori Xmega128D3 erisusi.

2.6 Programmaatori valik

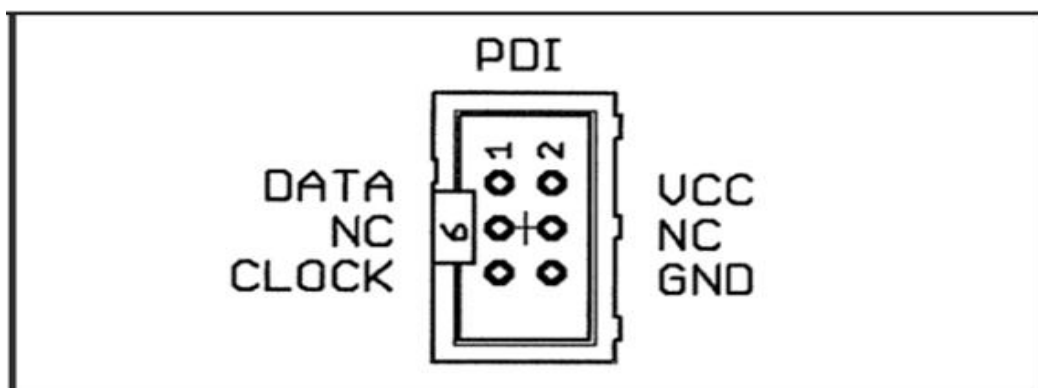
Programmaator on vajalik protsessorile programmikoodi peale kirjutamiseks. Ilma selleta pole võimalik tühjale programmeerimata protsessorile koodi peale kirjutada, sest sellel puudub vastav kommuniqueeriv tarkvara *bootloaderis*.

Programmaator valiti kolme kriteeriumi põhjal:

- Oleks võimeline PDI pordi kaudu protsessorit programmeerima.
- AVR® arhitektuuri protsessoritele mõeldud.
- Saadaval sobivas e-poes.

Valitud sai DIAMEX ALL-AVR® programmaator [11]. See on Atmeli® enda programmaatori AVRISP mkII® kloon [12]. See lubab valida nii 3,3 kui 5 voldise toitega programmeerimist ning toetab mitut erinevat liidest, sealhulgas ka PDI-d.

See on USB-ga arvuti külge ühendatav ning töötab otse Atmel Studio 6® kasutajaliidesest. Protsessori külge ühendus toimub PDI liidese kaudu (Joonis 2.1). VCC tuleb ühendada plaadi toite külge ning GND plaadi maanduse külge. NC (Not Connected) pole ühendatud. Data ots tuleb ühendada Xmega® protsessori puhul PDI_DATA külge ning Clock ots protsessori PDI_CLK külge [19].



Joonis 2.1 Diamex ALL-AVR® programmaatori PDI liides [29].

3. LCD ekraan

3.1 Ekraani valiku kriteeriumid

Projekti teine tähtsaim keskne osa on ekraan. Selle valik sõltub järgnevatest tingimustest:

- Piisavalt odav hind masstootmiseks.
- Ekraani suurus peab võimaldama näidata mahukat informatsiooni.
- Töötab töökeskkonnas (tehas, katlamaja), peab vastu tolmule.
- Puuteekraan vajutatav lisaks näpule ka töökinda või tõmbi esemega.
- Protsessoriga ühenduste arv võimalikult väike.
- Valitud protsessor peab olema võimeline piisavalt kiirelt ekraani juhtima.
- Väike energiakulu.
- Vähemalt 16-bitine (65536 erinevat värvi) värvide tugi.
- Ekraani taustvalgustuse tugevus peab olema konfigureeritav ning taustvalgus peab olema toidetav 5 voldise toitega.

Ekraani tahetav suurus võiks olla võimalikult lähedal puutetundlike nutitelefonide suurusele. Levinuim nutitelefonide suurus jääb nelja ja viie tolli vahele [13]. Põhjuseks see, et tarbija on harjunud sarnase suurusega ekraanidega ning sellel saab kuvada piisavalt palju informatsiooni.

Puutetundlikke ekraane on saadaval peamiselt kahes erinevas tehnoloogilises lahenduses: takistuslikud ning mahtuvuslikud. Kuna mahtuvuslik tehnoloogia on märksa kallim ning enamasti ei tööta ka kinda või eseme kasutamisel, siis langeb valik takistusliku puuteekraani kasuks. Lisaks on takistuslik ekraan palju veakindlam tolmuses töökeskkonnas.

Takistusliku tehnoloogia miinused:

- Halvem ekraani kontrastsus, sest ekraanil on toimimiseks ekstra kihid materjali.
- Ekraanile tuleb suhteliselt kõvasti vajutada, et registreerida puude.
- Eelmisest olenevalt võivad kergemini tekkida puuteekraani pinnale kahjustused.

LCD ekraane on saadaval erinevate juhtimisviisidega. Kõige algelisema ehitusega on *raw pixel-dot-clock* ekraan, millel puudub sisemise mälu graafikadraiver ekraanipildi salvestamiseks [14]. Selle asemel peab külge ühendatud protsessor ise uuendama kogu ekraanipilti umbes 60 Hz sagedusega. Taoline mahukas uuendamine nõuaks näiteks 480x272 resolutsiooniga ekraani puhul 130560 piksli uuendamist 60 korda sekundis. Arvestades, et iga värviline piksel on mälu salvestatud 8-bitisena ning iga piksli defineerib kolm värvibaiti on ühe ekraani info mahuks umbes 400 kilobaiti. Sekundis tuleks uuendamiste kogu andmemahuks $400 \text{ bilobaiti} * 60 \text{ Hz} = 24000 \text{ kilobaiti} = 24 \text{ megabaiti}$. Sellist mahtu ning raudvara kiirust, mis suudaks taolise ülesandega toime tulla, ei leidu projekti nõutud väikestes 8-bitistes protsessorites. Seetõttu peab LCD ekraan omama sisemist graafikadraiverit, mis ise tegeleb ekraani uuendamisega ning omab sisemist graafikamälu vähemalt ühe ekraanipildi salvestamiseks. Lisaks aitab ekraani sisemine mälu protsessorit sellega, et mingi väikse ala üle joonistamiseks ei ole vaja protsessoril uuendada kogu ekraani.

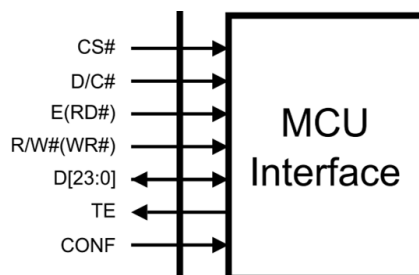
Enamus värvilisi LCD ekraane võimaldavad kasutada graafikadraiveriga suhtluseks erinevaid liideseid. Kõige lihtsam liides on paralleelliides, mis kasutab konfigureeritava laiusuga andmesiini ning kommunikatsiooni juhtotsi. Käskude saatmine toimub alati 8-bitise laiusuga, kuid ekraanile kuvatavate pikslite värviandmete siini laius on muudetav vastavalt riistvarale.

3.2 Prototüüp LCD ekraan

LCD ekraani õigeks valikuks katsetati mitut erinevat ekraani. Kõige olulisemaks prototüübi ülesandeks oli välja selgitada minimaalne paralleelliidese andmesiini laius, mille puhul ekraanipildi uuendamine liiga aeglaseks ei jää. Algseks prototüüpimiseks valiti välja 4,3 tollise TFT (*Thin-film-transistor*) LCD ekraaniga moodul [15], mille plaadi peal oli kasutatud graafikadraiverina kiipi SSD1963 [26]. Antud moodulit on lihtne juhtmetega protsessori külge ühendada, sest kõik vajalikud otsad on eraldi PIN-idenas selle jaoks välja toodud.

Kommunikatsiooni juhtimiseks oleva nelja juhtotsa funktsioonid sõltuvad graafikadraiveri konfiguratsioonist (Joonis 3.1). CONF juhtotsaga saab valida kahe erineva graafikadraiveri režiimi vahel[26]:

- **6800** - CONF otsa loogilise nulli korral. Konfigureerib neljaks juhtotsaks CS (*Chip Select*), D/C (*Data/Command*), E (*Enable*), R/W (*Read/Write*). CS otsa kasutatakse selles režiimis CLK otsana, andmeid või käsked sisestatakse andmesiinilt draiverisse ning andmeid loetakse draiverist välja CS PIN-i tõusval frondil. D/C loogilise nulli korral edastatakse käsked, loogilise ühe korral andmeid. E juhtotsaga aktiveeritakse ja deaktiveeritakse graafikadraiverit. R/W juhtots valib, kas tegu on kirjutamistsükliga või lugemistsükliga. R/W loogilise nulli korral loeb graafikadraiver andmeid siinilt sisse ning loogilise ühe korral paneb oma andmed siinile.
- **8080** - CONF otsa loogilise ühe korral. Konfigureerib neljaks juhtotsaks CS, D/C, RD (*Read*), WR (*Write*). CS otsa kasutatakse graafikadraiveri aktiveerimiseks ja deaktiveerimiseks. D/C töötab samal põhimõttel nagu 6800 režiimi puhul. RD juhtotsa langeval frondil pannakse siinile ettenähtud draiveri andmed ning see protsess lõpetatakse tõusval frondil. WR juhtotsa tõusval frondil loetakse andmed siinilt graafikadraiverisse sisse.



Joonis 3.1 SSD1963 graafikadraiveri juhtotsad [26].

Lihtsuse mõttes kasutati 8080 režiimi, sest sellisel juhul sai CS juhtotsa jäädavalt ühendada loogilise nulli ehk graafikadraiver oli pidevalt aktiveeritud. TE (*Tearing Effect*) juhtotsa ei kasutatud ning see jäi ühendamata (Joonis 3.1).

Katsetatud andmesiini laius oli 8 bitti, mille puhul tuli ühe piksli ekraanile kuvamiseks saata paralleelliidesel kokku kolm baiti värviinformatsiooni. Seda formaadis 666 ehk iga RGB värvi jaoks 6 bitti infot(Joonis 3.2). See pole väga optimaalne lahendus, sest iga baidiga saadetakse kaks tühist bitti.

Interface	Cycle	D[23]	D[22]	D[21]	D[20]	D[19]	D[18]	D[17]	D[16]	D[15]	D[14]	D[13]	D[12]	D[11]	D[10]	D[9]	D[8]	D[7]	D[6]	D[5]	D[4]	D[3]	D[2]	D[1]	D[0]	
24 bits	1 st	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0	
18 bits	1 st							R5	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1	B0	
16 bits (565 format)	1 st									R5	R4	R3	R2	R1	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1	
16 bits	1 st									R5	R4	R3	R2	R1	R0	X	X	G5	G4	G3	G2	G1	G0	X	X	
	2 nd									B5	B4	B3	B2	B1	B0	X	X	R5	R4	R3	R2	R1	R0	X	X	
	3 rd									G5	G4	G3	G2	G1	G0	X	X	B5	B4	B3	B2	B1	B0	X	X	
9 bits	1 st																	R5	R4	R3	R2	R1	R0	G5	G4	G3
	2 nd																	G2	G1	G0	B5	B4	B3	B2	B1	B0
8 bits	1 st																	R5	R4	R3	R2	R1	R0	X	X	
	2 nd																	G5	G4	G3	G2	G1	G0	X	X	
	3 rd																	B5	B4	B3	B2	B1	B0	X	X	

Joonis 3.2 SSD1963 graafikadraiveri paralleelliidese andmeformaadid [26].

Kogu LCD ekraani 480x272 piksli ülekirjutamiseks kuluva aja arvutamiseks on vaja teada ühe piksli kirjutamise aega. Selle saab välja arvutada protsessori taktsageduse ning tsüklis tehtavate protsessori käskude arvuga. Käskude arvu saab välja lugeda kompileeritud C koodi Assembleri kujust, mille saab kätte Atmel Studio 6@ Disassembly keskkonnast. Ühe piksli kirjutamise tsükkel on niiöelda *kuum tsükkel*, mida läbib programmikood kümneid tuhandeid kordi ning see peab olema kindlasti ajaliselt optimeeritud. Vastav Assembler kood koos C keele programmikoodiga on järgmine:

<i>Address</i>	<i>Assembler käsk</i>	<i>Käsu seletus</i>	<i>C keele käsk</i>
0000298C	STS 0x0604,R30	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_and;
0000298E	OUT 0x11,R28	(Out to I/O location)	GRAPHICS_DATABUS = color.Blue;
0000298F	STS 0x0604,R25	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_or;
00002991	STS 0x0604,R30	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_and;
00002993	OUT 0x11,R29	(Out to I/O location)	GRAPHICS_DATABUS = color.Green;
00002994	STS 0x0604,R25	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_or;
00002996	STS 0x0604,R30	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_and;
00002998	OUT 0x11,R17	(Out to I/O location)	GRAPHICS_DATABUS = color.Red;
00002999	STS 0x0604,R25	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_or;
0000299B	SUBI R18,0x01	(Subtract immediate)	while(count)
0000299C	SBC R19,R1	(Subtract with carry)	
0000299D	CP R18,R1	(Compare)	
0000299E	CPC R19,R1	(Compare with carry)	
0000299F	BRNE PC-0x13	(Branch if not equal)	

Maksimaalset protsessori taktsagedust 32 MHz kasutades ning eeldades, et ühe piksli kirjutamise tsükkel võtab aega ($0x299F - 0x298C = 0x13 = 19$) 19 takti (n_{takt}), saab kogu ekraani üle kuvamise aja (t_{upd}) arvutuse:

$$t_{upd} = t_{käsk} \times n_{takt} \times LCD_x \times LCD_y = \frac{1}{32000000} \times 19 \times 272 \times 480 = 0.07752 \text{ s} = 77,12 \text{ ms}$$

See aeg 77 ms on piisavalt väike, et ekraani üle kuvamine toimuks inimsilmale nähtamatu kiirusega. Samas on see liiga aeglane, et kuvada sujuvaid animatsioone. Lisaks on arvutus tehtud ekraani üle kuvamisel ühe ainsa värviga. Mitmevärvilise pildi kuvamisel tuleb veel lisaks igas tsüklis uus värv mälust valmis laadida ning nii muutub ekraani üle kuvamise aeg tunduvalt pikemaks. Prototüübi testimisel oli 8-bitise andmesüübi paraalleelliides piisavalt kiire, et ei olnud silmaga võimalik näha ekraani ülekirjutamist. Samas ei sobi prototüüp LCD lõplikuks valitud ekraaniks, sest on hinnalt kallim, võtab palju ruumi ning mooduli plaadil on palju elektroonikat, mis selles projektis oluliseks ei osutu.

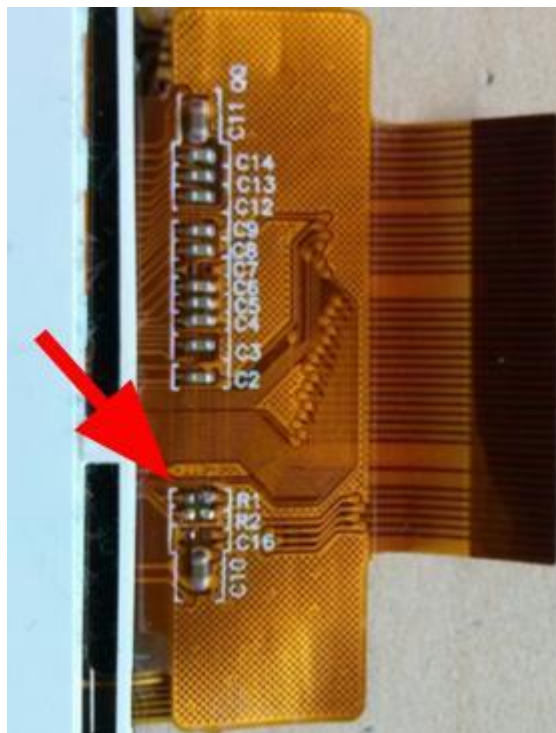
3.3 Valitud LCD ekraan

Lõplikuks valikuks osutus Hiina firma „Shenzhen Elida Technology Co., Ltd.“ TFT LCD puutetundlik ekraan, mille suurus on 3,95 tolli ning resolutsioon 320x480 [25]. Sisemiseks graafikadraiveriks firma Ilitek® mikrokiip ILI9488 [24]. Antud ekraani puhul toetatakse ainult kolme erinevat paralleelliidest. Graafikadraiveri kommunikatsiooniliidese valik toimub antud LCD ekraanil raudvaraliselt IMx PIN-idega [25]. PIN IM2 on ekraani sees ühendatud loogiliseks nulliks, IM1 loogiliseks üheks ning IM0 valik käib LCD ekraani lintkaabli pealnulltakisti abil (Tabel 3.1).

Tabel 3.1 ILI9488 graafikadraiveri kommunikatsiooniliidese valik [24].

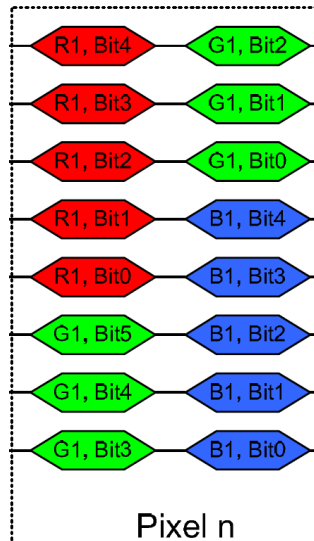
MIPI-DBI Type B				
IM2	IM1	IM0	Interface	Data Pin in Use
0	1	0	16-bit bus	DB [15:0]
0	1	1	8-bit bus	DB [7:0]

8-bitise andmesiini laiuse jaoks peab eemaldama takisti kohalt R2 (GND) ning jootma kohale R1 (VCC). R1 on tähistatud Joonisel 3.3 punase noolega.



Joonis 3.3 LCD ekraani lintkaabel, nulltakisti asukoht.

ILI9488 juhtotsad on üksühele samad, mis SSD1963 graafikadraiveril ning graafikadraiveri juhtimine toimub täpselt sama loogikaga nagu 8080 režiimis (Joonis 3.1). Prototüübiga võrreldes on võimalik värviinfo andmeid veelgi kokku pakkida, kasutades prototüübi 666 RGB formaadi (Joonis 3.2) asemel 656 formaati, mille kasutamiseks piisab ainult kahest infobaidist (Joonis 3.4).



Joonis 3.4 RGB formaat 656 [24].

Kogu ekraani üle kuvamise aeg muutub suurema ekraani resolutsiooni tõttu pikemaks, kuid optimeerituma RGB formaadi tõttu väheneb käskude arv kolme võrra. Muutunud ühe piksli kirjutamise *kuuma tsükli* Assembler kood:

<i>Address</i>	<i>Assembler käsk</i>	<i>Käsu seletus</i>	<i>C keele käsk</i>
0000299E	STS 0x0604,R21	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_and;
000029A0	OUT 0x11,R30	(Out to I/O location)	GRAPHICS_DATABUS = bluegreen;
000029A1	STS 0x0604,R20	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_or;
000029A3	STS 0x0604,R21	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_and;
000029A5	OUT 0x11,R22	(Out to I/O location)	GRAPHICS_DATABUS = greenred;
000029A6	STS 0x0604,R20	(Store direct to data space)	GRAPHICS_WRX_PORT_OUT = port_or;
000029A8	SUBI R18,0x01	(Subtract immediate)	while(count)
000029A9	SBC R19,R1	(Subtract with carry)	
000029AA	CP R18,R1	(Compare)	
000029AB	CPC R19,R1	(Compare with carry)	
000029AC	BRNE PC-0x0E	(Branch if not equal)	

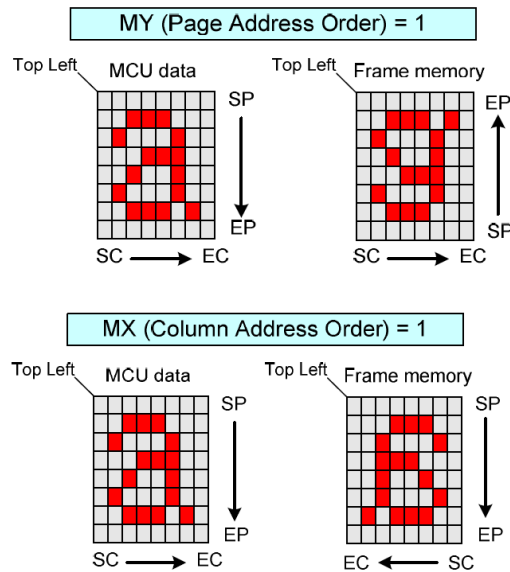
Ühe tsükli aeg on selle Assembler koodi lõigu järgi ($0x29AC - 0x299E = 0x0E = 14$) 14 takti (n_{takt}). Ekraani üle kuvamise aja (t_{upd}) arvutus tuleb järgmine:

$$t_{upd} = t_{käsk} \times n_{takt} \times LCD_x \times LCD_y = \frac{1}{32000000} \times 14 \times 272 \times 480 = 0.0672 \text{ s} = 67.2 \text{ ms}$$

Aeg 67 ms on piisavalt kiire, et ekraani üle kuvamine toimuks inimsilmale nähtamatu kiirusega. Protsessor peab kindlasti töötama maksimum taktsagedusega 32 MHz, sest muul juhul jääks ekraani kuvamine liiga aeglaseks. Seda näitas ka madalamate taktsageduste testimine. Madalamate taktsageduste puhul hakkas tekkima silmaga nähtavaid viiteid ekraani üle kuvamisel.

3.4 LCD Ekraani Juhtimine

LCD ekraani kasutamiseks tuleb initsialiseerida protsessori juhtotsad ning graafikadraiver. Otsad CS, D/C, WD, RD tuleb seadistada väljunditeks. 8-bitise andmesiini suund muutub vastavalt lugemis- või kirjutamistsüklile. Graafikadraiveris tuleb konfiguratsiooniregistrid saates üle paralleelliidese käskusid ning nendega kaasas käivaid andmeid. Algsete initsialiseerimata konfiguratsiooniseadete puhul on näiteks RGB pikslite formaat 18-bitine 666 formaat (Joonis 3.2). See tuleb ümber seadistada 16-bitisesse 656 formaati (Joonis 3.4). Lisaks tuleb kuvatava ekraani suund ümber keerata, sest ekraani alumine nurk peab asetsema lintkaabli vastasnurgas. Algväärtusena asub see lintkaabli juures. Selleks, et Y0 ja X0 koordinaadid oleksid hoopis algse seadistuse maksimumkoordinaatides ehk vastasnurgas, tuleb sisestada registrisse MY ja MX loogilised ühed (Joonis 3.5).

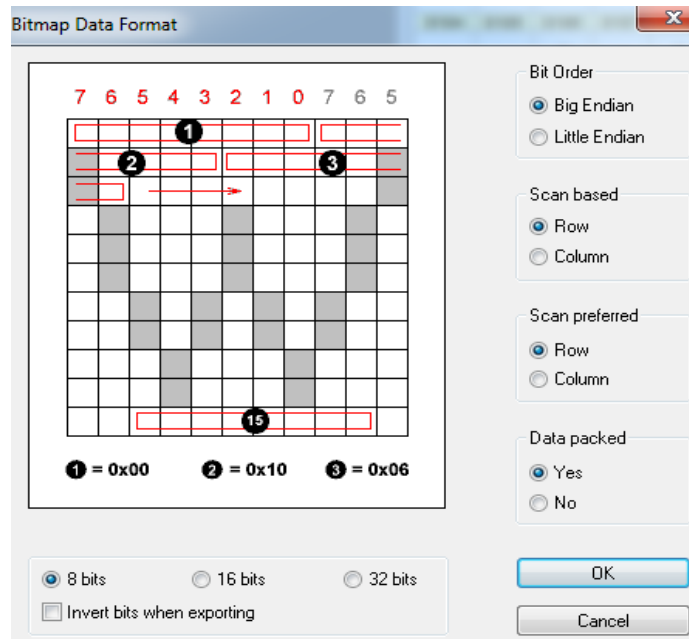


Joonis 3.5 ILI9488 graafikadraiveri kuvatava ekraani suuna registrid [24].

Ekraaniga kommunikatsiooni korrektse töötamise kontrollimiseks on võimalik kindlat käsku saates tagasi lugeda graafikadraiveri identifikaatorit. See peab iga LCD ekraani eksemplari puhul olema sama väärtus, sest kasutatud on alati sama graafikadraiverit ILI9488. Nii saab luua LCD ekraani graafikadraiveri lugemistsükli kasutava kontrollimisfunktsiooni, mis iga protsessori taaskäivituse alguses selgitab välja LCD ekraani staatuse. Kui graafikadraiverist tagasi loetud väärtus on vale, tuleb kasutajale probleemist teada anda taustavalgustust vilgutades ning mööda kommunikatsioonimoduleid veateadet esitades. Õige väärtuse puhul jätkatakse normaalse programmi käiguga.

Ekraanile info kuvamiseks tuleb kõigepealt seadistada X- ja Y-teljel kuvamise alg- ning lõppkoordinaadid. See määrab ekraani peal ruudu, kuhu sisse kirjutatakse uue pildi informatsiooni. Kirjutamine algab seadistatud ruudu nullkoordinaatidest ning andmeid kuvatakse piki X-telge ehk ruut täidetakse rea kaupa. Rea lõppu jõudes liigub mäluosuti teise rea algusesse, nii kuni kõik read on täidetud. See pildiandmete kuvamisviis on täpselt sama BMP failiformaati salvestatava info kujuga ning nii saab kirjutada BMP faili infot otse LCD ekraanile [2]. Optimeerimise eesmärgil tuleks kuvatavad pildid eelnevalt ka 16-bitisesse 565 RGB formaati viia (Joonis 3.4).

Fontide kuvamiseks ekraanil tuleks kasutada sarnast salvestusformaati. Fontide värviinfot saab salvestada kokkupakituna binaarsel kujul: 0 - taustavärv, 1 - tähevärv. Lisaks saab sellisel viisil paigutada ühte baiti 8 piksli jagu informatsiooni (Joonis 3.6). Fontide loomiseks ning taolise formaadi saamiseks kasutatakse programmi BitFontCreator Pro 3.5 [16]. See programm lubab salvestada UTF-8 formaadis fonte, mille abil saab kuvada tähti ka väljaspool inglise keele tähestikku [32].

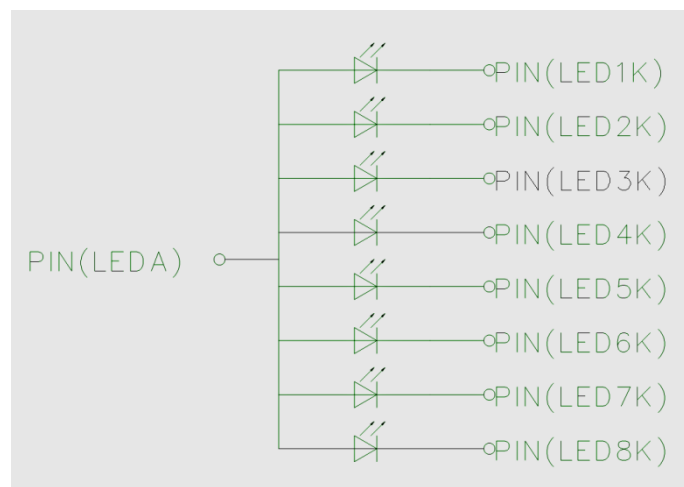


Joonis 3.6 BitFontCreator Pro 3.5 andmeformaad [16].

Tuleb luua kaks infomassiivi, üks täheandmete alguste aadressidega ning teine täheandmetega. UTF-8 põhjal saadav tähekood viitab esimeses massiivis olevale täheaadressile, milles olev väärtus suunab omakorda ekraanile kuvatava tähe pikslite informatsioonile. Taoline süsteem lubab varieeruva laiusega fonti kasutada. Laiuse saab välja arvutada kogu tähe informatsiooni pikkust jagades alati konstantseks jääva tähe kõrgusega. Fondi faili luues on seetõttu vajalik kindlale andmeväljale sisestada ka fondi kõrgus.

Taustvalgustus on antud LCD ekraanil implementeeritud paralleelselt ühendatud kaheksa LED-iga. Nende kõigi toide tuleb ühest PIN-ist LEDA, kuid LED-ide kateodid on eraldi väljundid (Joonis 3.7). LED-id on mõeldud töötama 3,3 voldise toitega [25]. Plaadi peal tuleb LED-ide vooluringi lisada voolu piirav takisti, mille väärtusest sõltub taustvalguse eredus: mida suurem takisti, seda tuhmimad on LED-id ning vastupidi. Ekraani andmeleht soovib LED-e läbivaks voolutugevuseks umbes 90 mA. Arvestades, et LED-i enda pingelang on umbes 0.6 volti, jääb pingelanguks üle takisti 2,7 volti. Ohmi seadust kasutades tuleb takistuse arvutus:

$$R = \frac{2,7 [V]}{90 [mA]} = 30 \Omega$$



Joonis 3.7 LCD taustvalgustuse LED-ide paralleelühendus [25].

Selleks, et tarkvaraliselt oleks võimalik juhtida ekraani valgustihedust, tuleb lisada LED-ide ühisesse vooleringi transistorlüliti. See avab ja sulgeb ühenduse maandusega. Juhtsignaalina võimaldab protsessor kasutada PWM signaali, millega saab täpselt määrata ära transistori lahtioleku ja ka LED-ide põlemise aja. Signaali sagedus peab olema piisavalt kõrge, et ei oleks võimalik inimsilmaga näha LED-ide ümberlülitusi. Minimaalne kasutatav sagedus on umbes 40 Hz [33]. PWM signaali sagedus (F_{PWM}) sõltub taimerisagedusest (F_{timer}) ning valitud taimeriperioodist (PER):

$$F_{PWM} = \frac{F_{timer}}{2 \times PER}$$

Madalaim võimalik taimerisagedus (F_{timer}):

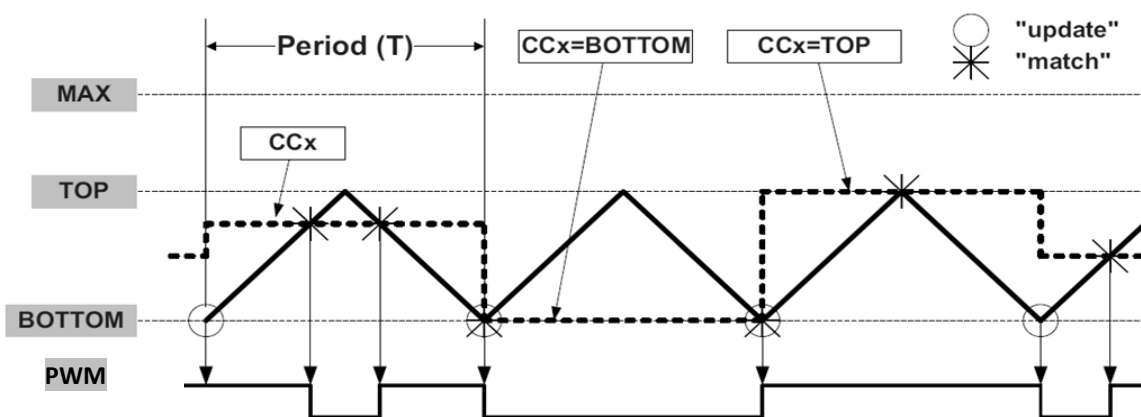
$$F_{timer} = \frac{32 \text{ MHz}}{1024} = 32 \text{ kHz}$$

Koodi mugavuse mõttes on valitud taimeriperioodi väärtus 100, sest sellisel juhul on lihtsam määrata CCx (*Capture Compare*) registrit. Sellisel juhul tuleb PWM signaali sagedus:

$$F_{PWM} = \frac{32000 \text{ Hz}}{2 \times 100} = 160 \text{ Hz}$$

See on suurem kui minimaalne kasutatav sagedus 40 Hz ning selle sageduse puhul pole LED-ide ümberlülitusi näha. CCx registriga saab määrata PWM signaali pulsi laiust ning sellega ka transistorlüliti lahtioleku aega. Joonis 3.8 on näha PWM signaali sõltuvus CCx -st, taimerilugem käib nurgana üles alla väärtuste 0 ja 100 (PER) vahel:

- Esimesel kolmandikul on CCx umbes 80 ning nii on PWM signaal 80% perioodist kõrge nivooga ning 20% madalaga. PWM signaal vahetab nivood hetkel, kui taimerilugem on võrdne CCx registris määratletuga.
- Teisel kolmandikul on CCx väärtus 0 ning PWM signaal kõrge 0% perioodist.
- Kolmandal kolmandikul CCx väärtus 100 ning PWM signaal kõrge 100% perioodist.



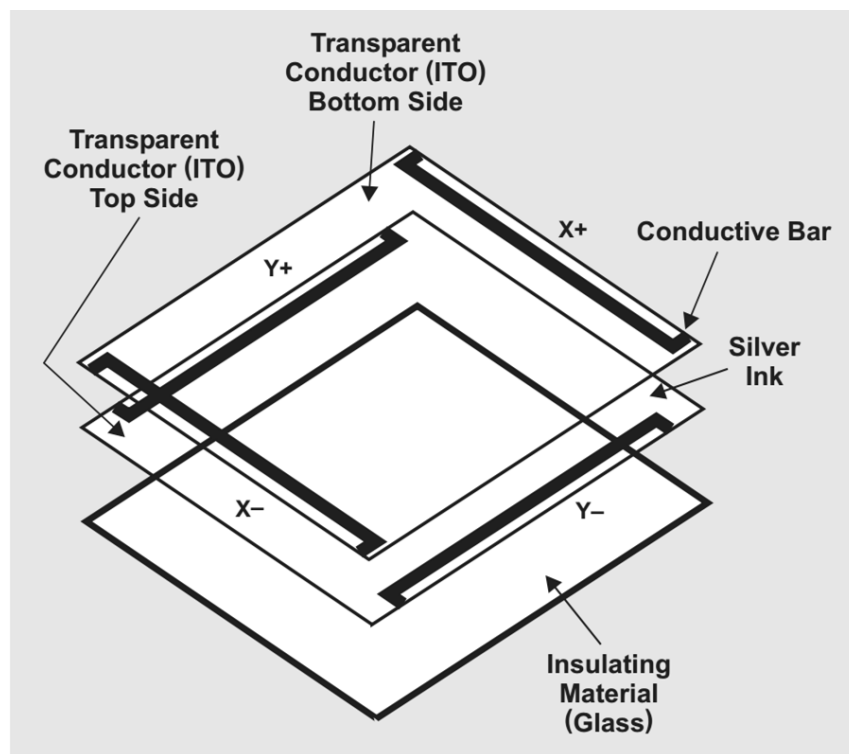
Joonis 3.8 Dual-slope Pulse Width Modulation [19].

LED-ide valgustihedus on otseselt määratud PWM signaali kõrge nivoo pikkusega ning on nii vastavuses CCx registrisse kirjutatava väärtusega. CCx väärtuse 100 puhul on toitesignaal kogu aeg sees ning LED-id aktiveeritud kogu aeg, tekitades maksimaalse valgustiheduse. CCx väärtuse 0 puhul puudub toitesignaal ning taustavalgustus on välja lülitatud. 0 ja 100 vahele jäävad väärtused määravad protsendiliselt ära valgustiheduse. Joonis 3.8 esimesel kolmandikul on CCx väärtus 80 ning valgustihedus 80% maksimaalsest valgustihedusest.

4. Puutetundlik ekraan

4.1 Ülesehitus

Valitud LCD ekraani puutetundlik ekraan kasutab takistuslikku nelja juhtme tehnoloogiat [28]. Selle puhul koosneb puutekraan kahest läbipaistvast plastkileplaadist, mille sisepinnad on kaetud üliõhukese kihiga resistiivse materjaliga. Mõlema plaadi vastasservades on elektrodid pinge rakendamiseks ja mõõtmiseks. Plaadid on üksteisega risti, see tähendab, et elektrodid ühel plaadil ei kattu elektrodide asukohtadega teisel plaadil, vaid on paigutatud risti. Nende nelja elektroodi järgi tulebki nimi nelja juhtme tehnoloogia. Kahte plaati eraldab isolatsioon nagu näiteks klaas, mis puudutuse korral kontaktkohal enam ei isoleeri ning plaatide vahel tekib elektriline kontakt (Joonis 4.1. Joonis 4.1).



Joonis 4.1 4-juhtmelise Takistusliku Puutekraani ehitus [30].

Koordinaatide saamiseks tuleb teha puudutuse ajal kiirelt kaks eraldi mõõtmist, üks X-koordinaadi saamiseks ja teine Y-koordinaadi jaoks [23]. Esimese X-koordinaadi mõõte puhul tuleb rakendada alalispinge ühele plaadile (X plaadile) ning mõõta pinget pealmise plaadi (Y plaadi) elektroodi ning maa vahel. Y-koordinaadi mõõte jaoks tuleb rakendada pinge teisele plaadile (Y plaadile) ning mõõta pinget alumise plaadi (X plaadi) elektroodi ja maa vahel. Saadud pinge väärtused olenevad kahe plaadi kontakti piirkonnast ning nii saab kahe pinge väärtuse kaudu välja lugeda sellele võrdelised koordinaadid. Kahe erineva suunadimensiooniga koordinaatide saamine mõõdetud pinge väärtustest on võimalik tänu sellele, et puutekraani kaks plaati on üksteisega risti ning mõõtetulemus on ühe plaadi puhul X-suunaline ja teisel plaadil eelmisega ristsuunaline ehk Y-suunaline.

4.2 Implementatsioon

Puutekraani neli väljundit tuleb ühendada nii, et ühe plaadi elektroodidest üks on protsessori ADC sisendis ning teine ühendatud tavalisse sisend/väljund PIN-i. Vastavalt LCD ekraani andmelehele on puutekraani väljundid nimetatud XL, XR, YU, YD ning need on vastavuses Tabel 4.1 ning Joonis 4.1 järgmiselt:

- XR = X+excite = X+
- XL = X-excite = X-
- YD = Y+excite = Y+
- YU = Y-excite = Y-

See tähendab, et XL ning YU on ühendatud protsessori ADC sisenditesse ning XR ja YD tava sisenditesse.

Puutetundliku ekraani draiveri tarkvara on tehtud Atmeli® õpetuse ning näitekoodi alusel [23]. Näitekood oli ehitatud teist struktuuri protsessoritele, see tuli suurel määral ümber ehitada. Tabel 4.1 näitab, kuidas tuleb juhtida puutekraani väljundeid koordinaatide mõõtmisel ning lisaks ka puute ootamisel. Kui alumine plaat (X-plaat) maandada ning ülemine plaat (Y-plaat) panna *pull-up* takisti abil alalispinge alla, siis saab implementeerida *interrupti*, mis ootab Y-plaadi elektroodi pinge muutumist madalaks. Puute korral ühenduvad plaadid kokku ning Y-plaadi pinge muutub nulliks. Vastava muutuse püüab *interrupt* kinni ning nii alustab tarkvaras X-koordinaadi ning Y-koordinaadi mõõteprotsessi.

Tabel 4.1 4-juhtmelise Puutekraani skaneerimine [23].

	X+Excite	X-Excite	Y+Excite	Y-Excite
Standby	Gnd	Hi-Z	Hi-Z	Pull up / Int
X-Coordinate	Gnd	Vcc	Hi-Z	Hi-Z / ADC
Y-Coordinate	Hi-Z	Hi-Z / ADC	Gnd	Vcc

4.3 Puute koordinaatide leidmine ADC tulemusest

Pinge mõõtmiseks on ADC seadistatud 12-bitisesse märgita tulemuse režiimi ja referentsiks kasutatakse protsessori 1-voldist sisemist referentspinget. Mõõteskeem tuleb koostada nii, et mõõtmise hetkel oleks maksimaalne pinge ADC sisendil võrdne referentspingega 1 volt (U_{touch}). Selle jaoks tuleb skeemi X-Excite ja Y-Excite otstele takistid lisada, mis maksimaalset puutekraani pinget limiteerivad (R_{lim}). Nende suurus tuleneb pingejaguri arvutusest ning puutekraani enda takistusest, mis on mõõtmise järgi 600Ω (R_{touch}).

$$I_{touch} = \frac{U_{touch}}{R_{touch}} = \frac{1 [V]}{600 [\Omega]} = 1,66 mA$$

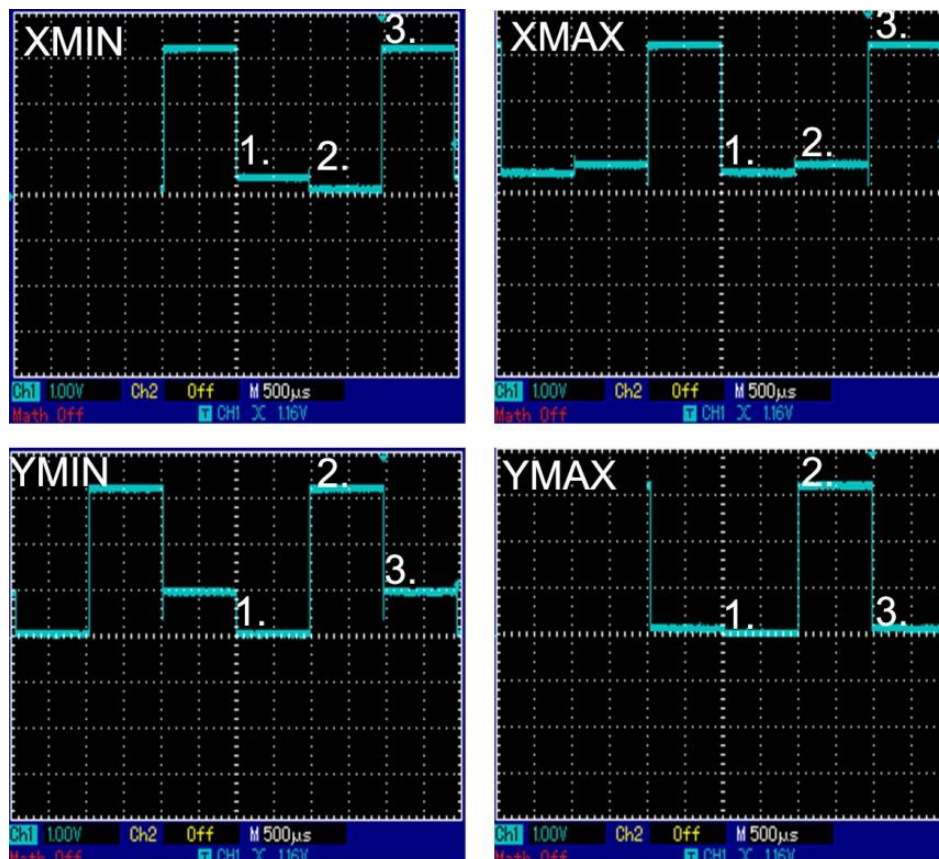
$$R_{lim} = \frac{VCC - U_{touch}}{I_{touch}} = \frac{3,3 [V] - 1 [V]}{1,66 [mA]} = 1380 \Omega$$

Maksimumpinge tekib ADC sisendile siis, kui puutekraani puute koordinaat on kas YMAX, YMIN, XMAX või XMIN olenevalt ekraani ühenduse orientatsioonist. Kuna ekraan pole ruudu kujuline vaid ristkülik, siis X-koordinaati mõõtes on ADC tulemus maksimaalselt 67.7% 4096-st ehk umbes 2754. Seda põhjusel, et X-suunas on puutekraan 320 pikslit lai ning Y-suunas 480 pikslit ning nende kahe suuruse suhe 320/480 on 0.6777. Ekraani andmelehe järgi puutekraani protsessori külge ühendades tekib olukord, kus Y-koordinaadi muutus on vastupidise loogikaga reaalse pingeväärtuse muutumisele [25]. Koordinaadi YMAX puhul on mõõdetav ADC pinge madalaim võimalik ning vastupidi koordinaadi YMIN puhul on ADC pinge maskimaalne. X-koordinaadi muutus on vastavuses pingemuutusega.

Peale *interrupti* saamist ehk puute tuvastamist alustatakse mõõtmisprotsessi. Esimese mõõtega tehakse uuesti kindlaks, et puudutus on veel olemas ning tuleb koordinaate mõõta. Seejärel seadistatakse ekraani juhtotsad X-koordinaadi režiimi. See tähendab, et XR tuleb seadistada protsessori väljundiks ning panna loogiliseks nulliks (GND), XL tuleb teha väljundiks ning seadistada loogiliseks üheks (VCC), YD ning YU tuleb määrata sisenditeks ning lisaks peab YU ka ADC sisend olema. Saadud ADC tulemus salvestatakse mälli.

Peale X-koordinaadi pingeväärtuse kätte saamist seadistatakse ekraani juhtotsad Y-koordinaadi režiimi ehk YD tuleb seadistada protsessori väljundiks ning panna loogiliseks nulliks (GND), YU tuleb teha väljundiks ning seadistada loogiliseks üheks (VCC), XR ning XL tuleb määrata sisenditeks ning lisaks peab XL ka ADC sisend olema. Saadud ADC tulemus salvestatakse ka mälli.

Nende kahe tsükli ostilloskoobi mõõtetulemused on näha Joonis 4.2.



Joonis 4.2 Ostilloskoobi pildid.

XMIN ja XMAX piltide puhul on näha järgmist mõõtettsükli ülesehitust (Joonis 4.2):

- Esimesel kolmandikul mõõdetakse, kas puude on veel olemas. Kuna YU on sellel hetkel loogiline null, siis on tõestatud puutekraani vajutus. Ilma puuteta oleks YU olekus loogiline üks ehk mõõdetav pinge oleks 3,3 volti (VCC).
- Teisel kolmandikul mõõdetakse X-koordinaati. XMIN puhul on tegu 0 koordinaadiga, sest pinge väärtus on minimaalne. ADC väärtus on XMIN korral 493. XMAX puhul on tegu max koordinaadiga, sest pinge on X-i võimalikus maksimumpunktis ehk 67.7 % 1 voldist. ADC väärtus on XMAX korral 2674.
- Kolmandal kolmandikul on toimumas Y- koordinaadi mõõtmine ning sellisel juhul on YU seadistatud väljundiks ning nivoole loogiline üks (VCC), mida ka pildil näha 3,3 voldise mõõtetulemusena.

YMIN ja YMAX piltide puhul on näha järgmist mõõtettsükli ülesehitust (Joonis 4.2):

- Esimesel kolmandikul mõõdetakse, kas puude on veel olemas. XL on sellel hetkel määratud sisendiks ning sõltub puutekraanist, kus X-plaat on sellel hetkel maandusesse ühendatud.
- Teisel kolmandikul mõõdetakse X-koordinaati. Sellisel juhul on XL seadistatud väljundiks ning andma välja loogilist ühte (VCC), mida ka pildil näha.
- Kolmandal kolmandikul mõõdetakse Y-koordinaati. YMIN puhul on tegu miinumkoordinaadiga, aga pinge on puutekraani ühendusest sõltuvalt sellel hetkel maksimumis. ADC väärtus on YMIN korral 4095. YMAX puhul on maksimumkoordinaat ning sellel hetkel on pinge minimaalne. ADC väärtus on YMAX korral 490.

Mõlema mõõtetulemuse puhul pole minimaalse pinge ADC väärtus 0, sest alati säilib mingi puutekraani takistus.

Mõõtetulemuste usaldatavuse suurendamiseks kasutatakse digitaalset filtrit. XY-koordinaatide lõpliku tulemuse saamiseks peab toimuma puute jooksul mitmeid mõõtmisi ning tulemusena väljastatakse nende mõõtmiste mediaantulemus. Kui nende mõõtmiste jooksul tuvastatakse puute lõpp, siis puute koordinaate ei salvestata. Lõpptulemusena väljastatakse kaks 12-bitist väärtust:

- YU sisendiga ADC väärtus, mis näitab X-koordinaati. XMIN ~ 490, XMAX ~ 2754.
- XL sisendiga ADC väärtus, mis näitab Y-koordinaati. YMIN ~ 4096, YMAX ~ 490.

4.4 Kalibratsioon

Puutekraani ADC väärtustest täpse LCD ekraani koordinaadi kätte saamiseks tuleb kasutada puutekraani kalibreerimist. Kompenseerima peab skaleerimise, nihke ning pöörde vigasid. Nende vigade elimineerimiseks tuleb kasutada puutetundliku ekraani ADC tulemustest LCD ekraani koordinaatide saamiseks kuue erineva koefitsientiga valemeid [23]:

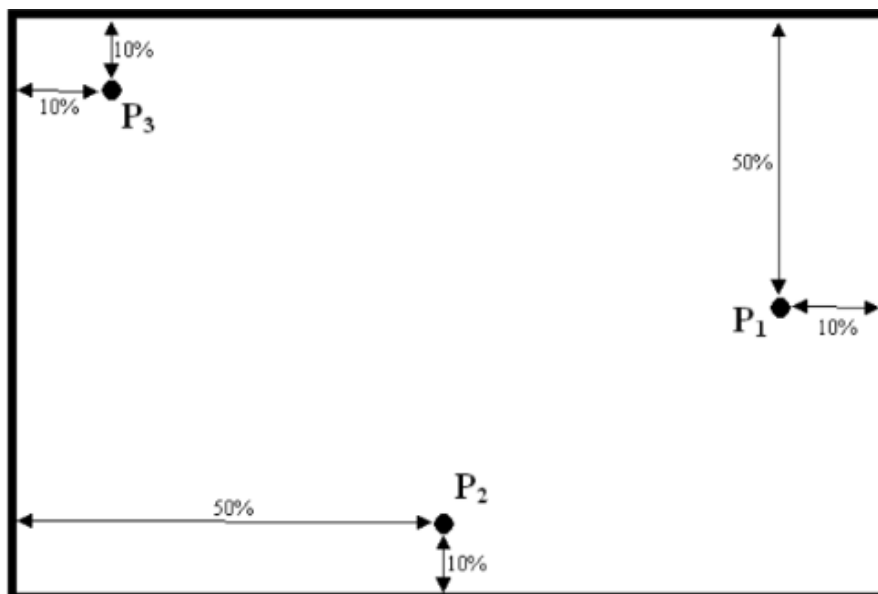
- $X_D = A(X_T) + B(Y_T) + C$
- $Y_D = D(X_T) + E(Y_T) + F$

X_D ja Y_D tähistavad LCD ekraani koordinaate. X_T ja Y_T tähistavad vastavaid ADC mõõtetulemusi. A, B, C, D, E, F tähistavad koefitsiente, mis on saadud kalibreerimise protsessi teel. Kalibreerimise jaoks on vaja kasutajal vajutada puutekraanil kolme kindlase punkti, mis annavad koefitsientide leidmiseks järgneva lineaarvõrrandite süsteemi [23]:

- $X_{D1} = A(X_{T1}) + B(Y_{T1}) + C$
- $X_{D2} = D(X_{T2}) + E(Y_{T2}) + F$
- $X_{D3} = A(X_{T3}) + B(Y_{T3}) + C$
- $Y_{D1} = D(X_{T1}) + E(Y_{T1}) + F$
- $Y_{D2} = A(X_{T2}) + B(Y_{T2}) + C$
- $Y_{D3} = D(X_{T3}) + E(Y_{T3}) + F$

X_{D1} tähistab esimese puutepunkti LCD ekraani X- koordinaati ning X_{T1} puutekraani puutepunkti X- mõõdu ADC tulemust. X_{D2} ja X_{T2} teise puutepunkti ning X_{D3} ja X_{T3} kolmanda punkti tulemusi. Y_D ja Y_T tähistused vastavad Y tähistustele.

Need punktid peaksid olema parima tulemuse saamiseks üksteisest võimalikult kaugel. Samas ei tohi need olla liiga lähedal puutekraani servale, sest seal puute täpsus langeb ning võib tekkida vigasid kalibreerimistulemuses. Selleks sobib Joonis 4.3 kalibratsioonipunktide paigutus, kus kõik mõõtepunktid on vähemalt 10% kaugusel ekraani äärest.



Joonis 4.3 Puutekraani kalibreerimispunktid [23].

Saadud lineaarvõrranditesüsteemist tuletatud valemid koefitsientide saamiseks [23]:

- $$A = \frac{X_{D1}(Y_{T2}-Y_{T3})+X_{D2}(Y_{T3}-Y_{T1})+X_{D3}(Y_{T1}-Y_{T2})}{X_{T1}(Y_{T2}-Y_{T3})+X_{T2}(Y_{T3}-Y_{T1})+X_{T3}(Y_{T1}-Y_{T2})}$$
- $$B = \frac{A(X_{T3}-X_{T2})+X_{D2}-X_{D3}}{(Y_{T2}-Y_{T3})}$$
- $$C = X_{D3} - A \times X_{T3} - B \times Y_{T3}$$
- $$D = \frac{Y_{D1}(Y_{T2}-Y_{T3})+Y_{D2}(Y_{T3}-Y_{T1})+Y_{D3}(Y_{T1}-Y_{T2})}{X_{T1}(Y_{T2}-Y_{T3})+X_{T2}(Y_{T3}-Y_{T1})+X_{T3}(Y_{T1}-Y_{T2})}$$
- $$E = \frac{D(X_{T3}-X_{T2})+Y_{D2}-Y_{D3}}{(Y_{T2}-Y_{T3})}$$
- $$F = Y_{D3} - D \times X_{T3} - E \times Y_{T3}$$

Kalibreerimistulemused tuleb salvestada EEPROM-i, et nad protsessori taaskäivitusel ei kustuks. Muidu peaks igal taaskäivitusel kasutaja uuesti kalibreerimisprotsessi läbima ning see oleks häiriv tavakasutaja jaoks. Selle asemel kontrollitakse igal taaskäivitusel EEPROMist kalibreerimistulemuste olemasolu. Tulemuse puudusel käivitatakse kalibratsioon. Kui tulemus on olemas loetakse koefitsiendid SRAMi ning kasutatakse iga puute korral LCD ekraani koordinaatide arvutuseks.

4.5 Erinevad puudutuste tüübid

Puutekraanilt saadud koordinaate ning puute pikkust analüüsides on võimalik implementeerida mitut erinevat tüüpi puudutusi. Puute pikkuse mõõtmiseks salvestatakse puute korral esimese saadud koordinaadi aeg ning puute lõpu koordinaadi aeg. Selle info põhjal on implementeeritud kolme erinevat sorti vajutused, mille ajalised parameetrid on saadud testijate tagasiside teel:

- **Lühike vajutus** - lühem kui 1 sekund. Kasutatakse kasutajaliideses ringi liikumiseks ning valikute tegemiseks.
- **Pikk vajutus** - pikem kui 1 sekund. Kasutatakse kasutajaliidese nuppudele lisafunktsioonide andmiseks. Näiteks *BACK* nupp lühikesel vajutusel viib eelmisele ekraanile, kuid pika vajutuse korral viib otse koduekraanile.
- **Väga pikk vajutus** - pikem kui 5 sekundit. Kasutatakse ainult kalibratsiooniprotsessi alustamiseks, sest see on ainus viis valesti kalibreeritud ekraani korral kasutajalt käsu saamiseks. Kalibreerimise alustamise nupp ekraanil ei töötaks, sest valede koefitsientide korral ei saa ühtegi kuvatavat nuppu vajutada.

Tehtud puudutuste koordinaate analüüsides on implementeeritud ka erinevates suundades libistamise (*SWIPE*) tuvastamine. Kasutaja puudutuse alguse korral salvestatakse puutekoordinaat. Puute lõpu korral võrreldakse lõpukoordinaati ning algkoordinaati ning selle järgi otsustatakse libistamise suund. Lisaks peab olema koordinaat muutunud kindla minimaalse suuruse võrra, et eristada libistamisi vajutustest. Libistamise suuna otsustamise korral kehtib ka lisatingimus, et kui üks koordinaatidest on muutunud üle minimaalse libistamise tingimuse, siis teine koordinaat peab olema muutunud vähem kui kindel maksimaalne suurus. See tingimus eemaldab võimalikest libistamistest diagonaalsed libistused, et hoida ära kasutaja arusaamatused kasutajaliidese käitumisega.

5. PCB

5.1 Nõuded PCB-le

- UART pistik põletiga ühenduseks. Selle ühenduse kaudu põletist tulev 5 voldi toide peab olema kogu plaadi energiaallikaks. Otsad RxD ja TxD peavad olema radadega ühendatud õige pordi ning PIN-ide külge protsessoril.
- Pingeregulaator, mis muudab 5 voldise toitepinge 3,3 voldiseks toiteks protsessorile ja muudele väiksemat pinget vajavatele moodulitele.
- Programmaatori pistik protsessori programmeerimiseks [29]. Peab olema ühendatud protsessori PDI PIN-ide külge (Joonis 2.1).
- UART pistik internetimooduliga ühenduseks. Sellesse pistikusse ühendatud 3,3 voldi toide peab olema internetimooduli toiteks. Otsad RxD ja TxD peavad olema radadega ühendatud õige pordi ning PIN-ide külge protsessoril. Ei tohi olla sama port, mis põleti UART.
- LCD ekraani pistik 45 PIN-iga [25]. Ära tuleb ühendada 8-bitine paralleelport, kolm juhtotsa D/C, RD, WD. Juhtots CS võib olla ühendatud maandusega, sest juhtotsi ei kasutata ühegi teise mooduli juhtimiseks ning sellisel juhul on LCD ekraani graafikadraiver alati aktiveeritud. Ka Reset otsa võib riistvaraliselt ära määrata ning ühendada loogilisse ühte, sest LCD ekraani taaskäivitust saab selle PIN-i kasutamise asemel teha tarkvaraliselt.
Puuteekraani juht- ning mõõteotsadest kaks - XR, YD - ühenduvad otse protsessoriga ning ülejäänud kaks otsa - XL, YU - tuleb suunata läbi umbes 1380-oomiste puuteekraani maksimumpinget limiteerivate takistite protsessori ADC sisenditesse.
LCD ekraani toiteotstele tuleb suunata pingeregulaatorist saadud 3,3 volti ning ekraani maanduseotsad suunata lisaplaadi maandusesse.
Taufvalgustuse toiteotsale tuleb määrata 3,3 voldine toide ning kaheksa üksiku LED-i otsadest tulevatele radadele tuleb plaadi peal lisada umbes 30-oomise väärtusega voolu piiravad takistid. Need rajad tuleb ühendada transistorlülitiga, mis avab ning sulgeb vooluringi ning sellega lülitab sisse ja välja taustvalgustuse LED-e. Transistorlülitit juhtimine peab toimuma PWM signaaliga ning selleks tuleb transistor ühendada vastava protsessori PIN-i külge. LCD ekraani ülejäänud ehk eelpool mittemainitud PIN-id võivad jääda ühendamata.
- SD-kaardi pistik. See ühendada protsessori SPI pordi külge.
- Temperatuuriandurite mõõteskeemid ning mõõteotsad. Need tuleb ühendada protsessori ADC sisenditesse.
- Releed ning transistorlülitid vahelduvvoolu moodulite kontrollimiseks. Transistorid tuleb ühendada protsessori tavaliste sisend/väljund PIN-idega.

5.2 Disainitud PCB analüüs

Pelltechi elektroonik disainis eeltoodud nõuete kohaselt PCB (Lisa 4) (Lisa 3). Sellel plaadil oli probleeme, mis tekkisid näpuvigadest või mittepiisavast andmelehtedejärgimisest.

- LCD ekraani taustavalgust juhtiva PWM signaali protsessori väljund oli määratud pordi D PIN-ile 4. Vastava programmikoodi kirjutamisel tuli välja, et Xmega128D3 protsessori puhul pole võimalik seda PIN-i kasutada PWM-i väljundina [20]. Põhjuseks viga protsessori ehituse juures, kus *PINREMAP* register lihtsalt ei tööta. See fakt on ära mainitud ainult andmelehe lisades. Selle tõttu tuleb PWM signaali väljund PIN 4 muuta pordi D PIN-iks 1.
- Praegune LCD ekraani toitepinge 3,3 volti on LCD ekraani andmelehe järgi liiga kõrge [25]. 3,3 volti on lubatav absoluutne maksimumväärtus ning selle kasutamine pole soovitatav. See toitepinge tuleb muuta LCD ekraani andmelehel pakutud tüüpiliseks toitepingeks 2,8 volti.
- Plaadile lisatud temperatuuriandurite mõõteskeemi referentspinge peab omama sama allikat, mida kasutatakse temperatuuriandurite mõõtmiseks. Praegusel plaadil kasutatud protsessori sisemine referentspinge ei oma sama allikat ning nii tekivad ebatäpsused mõõtmisel. Kattuva pingeallika puhul ei mängiks otseselt rolli pingekõikumised, sest mõõdetava pinge ning referentspinge suhe jääb samaks.
- Plaadile tuleks lisada kalendrikiip ning selle toiteks patarei. See on vajalik reaalaaja mõõtmiseks ning tulemuse säilitamiseks plaadi toitepinge kadumisel. Kiip peab olema ühendatud nii, et protsessori käimisel tuleb toide plaadilt ning ainult kogu plaadi toite kadumisel tuleb toide patareist. See on vajalik patarei eluea pikendamiseks. Kalendrikiip tuleb valida I²C toega ning ühendada tuleb protsessori I²C liidese külge [34].
- Plaadil olev SD-kaart tuleks asendada välise 2MB suuruse Flash mälu kiibiga Micron M25P16[35]. See tuleb ühendada protsessori SPI liidese külge SD-kaardi asemele. Põhjusteks SD-kaardi väiksem töökindlus ning suurem hind.

Kokkuvõte

Lõputöö eesmärk saavutada reaalne töötav LCD ekraani lisaplaat pelletipõletile õnnestus. Tulemus pole mitte mingil juhul lõplik variant ning PCB disaini tuleb veel suurel määral täiendada. Valmis on saadud ainult algne baas, mille peale disainida reaalne konkurentsivõimeline pelletipõleti kasutajaliides ning seda lisaplaadina realiseerida. Lõputöös sõnastatud projekti üldnõuded on hea pidepunkt edasiste arenduste tegemiseks.

Algse kasutajaliidese testimine tõestab, et komponendid on õigesti valitud ja ühendatud. Protsessor on piisavalt kiire LCD ekraani juhtimiseks ning ekraanide silmale nähtamatuks uuendamiseks. Ka vastavalt lõputöös välja toodud nõuetele kirjutatud algne programmikood funktsioneerib ning on piisavalt optimeeritud, et kasutaja saaks häirimatult pelletipõletit juhtida.

Kõigi komponentide valik on tehtud arvestades minimaalseimat võimalikku rahalist kulu. Saadud tulemus on rahuludav, sest lisaplaadi kogumaksumus on väga väike eriti võrreldes kogu põleti enda hinnaga. Edasiarendus ei näe ette suuri muutusi riistvaras ning nii võib hinnata ekraani füüsiliste komponentide omahinna hinnaklassi lõplikuks.

Kirjandus

1. **Pelltech** [WWW] <http://pelltech.eu/et/ettevottest/> (25.05.2015).
2. **BMP file format** [WWW] <http://www.digitalpreservation.gov/formats/fdd/fdd000189.shtml> (20.05.2015).
3. **Atmel AVR arhitecture** [WWW] <http://www.atmel.com/products/microcontrollers/avr/> (12.05.2015).
4. **Atmel megaAVR processor family** [WWW] <http://www.atmel.com/products/microcontrollers/avr/megaAVR.aspx> (12.05.2015).
5. **Atmel Studio 6** [WWW] http://www.atmel.com/microsite/atmel_studio6/ (12.05.2015).
6. **Atmel tinyAVR processor family** [WWW] <http://www.atmel.com/products/microcontrollers/avr/tinyAVR.aspx> (12.05.2015).
7. **Atmel XMEGA processor family** [WWW] http://www.atmel.com/products/microcontrollers/avr/AVR_XMEGA.aspx (12.05.2015).
8. **Atmel 32-bitAVR processorfamily** [WWW] <http://www.atmel.com/products/microcontrollers/avr/32-BitAVRuc3.aspx> (12.05.2015).
9. **Getting started with Xmega: differences from ATmega** [WWW] <https://omegacs.wordpress.com/2010/06/26/getting-started-with-xmega-differences-from-atmega-part-1/> (13.05.2015).
10. **Ready for XMEGA Board** [WWW] <http://www.mikroe.com/ready/xmega/> (15.05.2015).
11. **DIAMEX ALL-AVR** [WWW] <http://avr-programmer.com/all-avr/> (15.05.2015).
12. **AVRISP mkII** [WWW] <http://www.atmel.com/tools/avrismkii.aspx> (15.05.2015).
13. **A comprehensive look at smartphone screen size statistics and trends** [WWW] <https://medium.com/@somospostpc/a-comprehensive-look-at-smartphone-screen-size-statistics-and-trends-e61d77001ebe> (17.05.2015).
14. **4.3" 40-pin raw pixel-dot-clock TFT Display - 480x272 with Touchscreen** [WWW] <http://www.adafruit.com/products/1591> (17.05.2015).
15. **4.3"LCD Touch Screen Module Display TFT SSD1963 Controller,MCU** [WWW] <http://www.buydisplay.com/default/4-3-lcd-touch-screen-module-display-tft-ssd1963-controller-mcu> (17.05.2015).
16. **BitFontCreator Pro 3.5** [WWW] <http://www.iseasoft.com/font-editor/bitfontcreator-pro.html> (20.05.2015).
17. Yabanova, I., Taskin, S., Ekiz, H., Cimen, H. (2012) **Bootloader design application for embedded systems by using controller area network**. AWERProcedia Information Technology & Computer Science. 952-957.
18. White, E. (2011) **Making Embedded Systems**. O'Reilly Media.

19. Atmel Corporation. (2014) **8-bit Atmel XMEGA D Microcontroller.**
20. Atmel Corporation. (2014) **8/16-bit Atmel AVR XMEGA D3 Microcontroller.**
21. Atmel Corporation. (2010) **AVR1612: PDI programming driver.**
22. Atmel Corporation. (2011) **ATmega128, ATmega128L, 8-bit Atmel Microcontroller with 128KBytes In-System Programmable Flash.**
23. Atmel Corporation. (2007) **AVR341: Four and five-wire Touch Screen Controller.**
24. ILI TECHNOLOGY CORP. **ILI9488, a-Si TFT LCD Single Chip Driver 320(RGB) x 480 Resolution, 16.7M-color With Internal GRAM.**
25. Shenzhen Elida Technology Co.,Limited. (2015)**EL040T031-A320*480 dots matrix TFT LCD module.**
26. Solomon Systech. (2008)**SSD1963, 1215KB Embedded Display SRAM LCD Display Controller.**
27. Texas Instruments Inc. (2010) **Universal Asynchronous Receiver/Transmitter (UART).**
28. Texas Instruments Inc., Downs, R. (2005) **Using resistive touch screens for human/machine interface.**
29. Diamex. (2012) **ALL-AVR Programmer, Programming adapter for AVR controller with ISP, PDI, TPI protocol.**
30. Texas Instruments Inc. (2010) **4-Wire and 8-Wire Resistive Touch-Screen Controller Using the MSP430™.**
31. Ecma International. (2013) **Standard ECMA-404, The JSON Data Interchange Format.**
32. Unicode Consortium. (2014) **The Unicode Standard Version 7.0 – Core Specification.**
33. Arbetslivsinstitutet. Johansson, A., Sandström, M. (2003) **Sensitivity of the human visual system to amplitude modulated light.**
34. NXP Semiconductors. (2014) **UM10204 I 2C-bus specification and user manual.**
35. Micron Technology, Inc. (2011) **Micron M25P16 Serial Flash Embedded MemoryFeatures.**
36. Pelltech. (2015) **Pelltech Burner Serial Communication Protocol.**

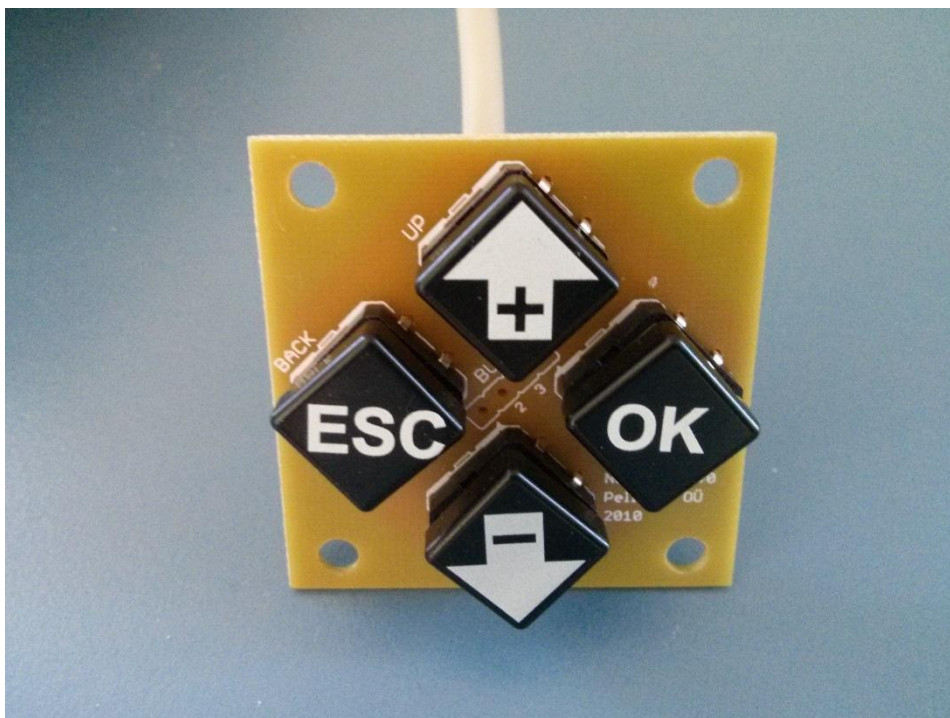
Lisad

Lisa 1. Pelltechi põleti praegune LCD ekraan.



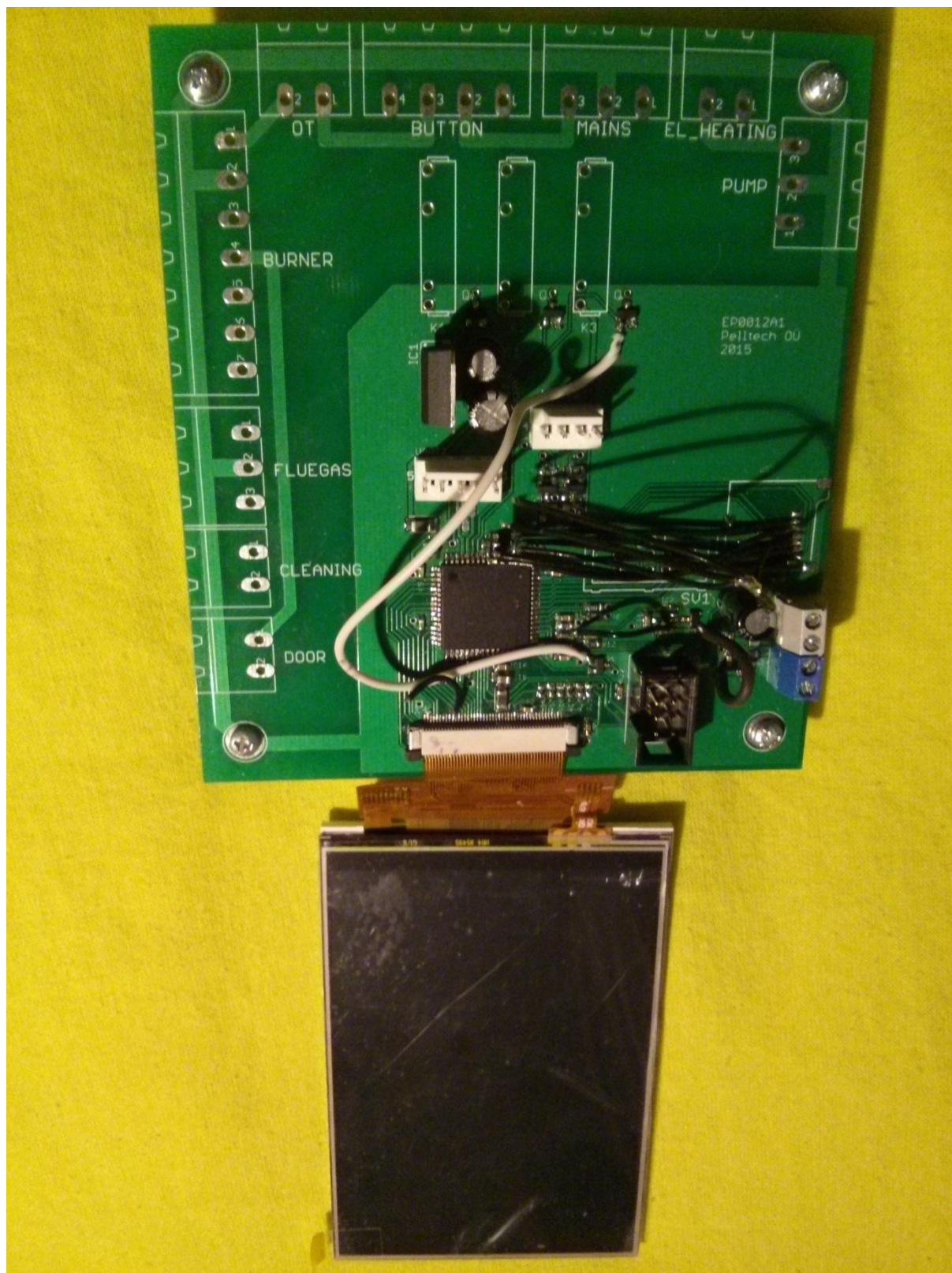
Lisa 1. Pelltechi põleti praegune LCD ekraan.

Lisa 2. Pelltechi põleti praegune nupustik.



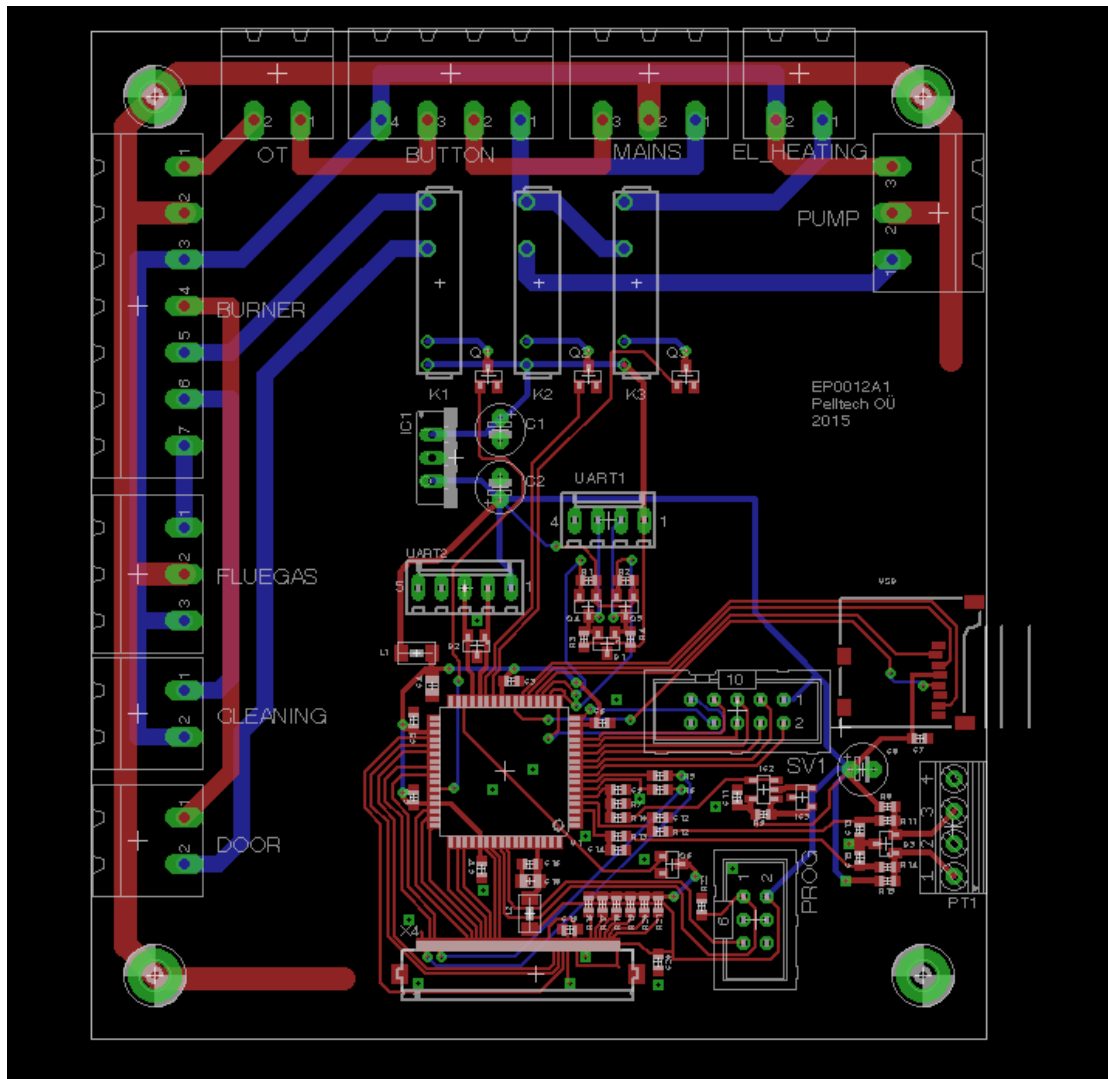
Lisa 2. Pelltechi põleti praegune nupustik.

Lisa 3. Lisaplaadi PCB koos puutetundliku LCD ekraaniga.



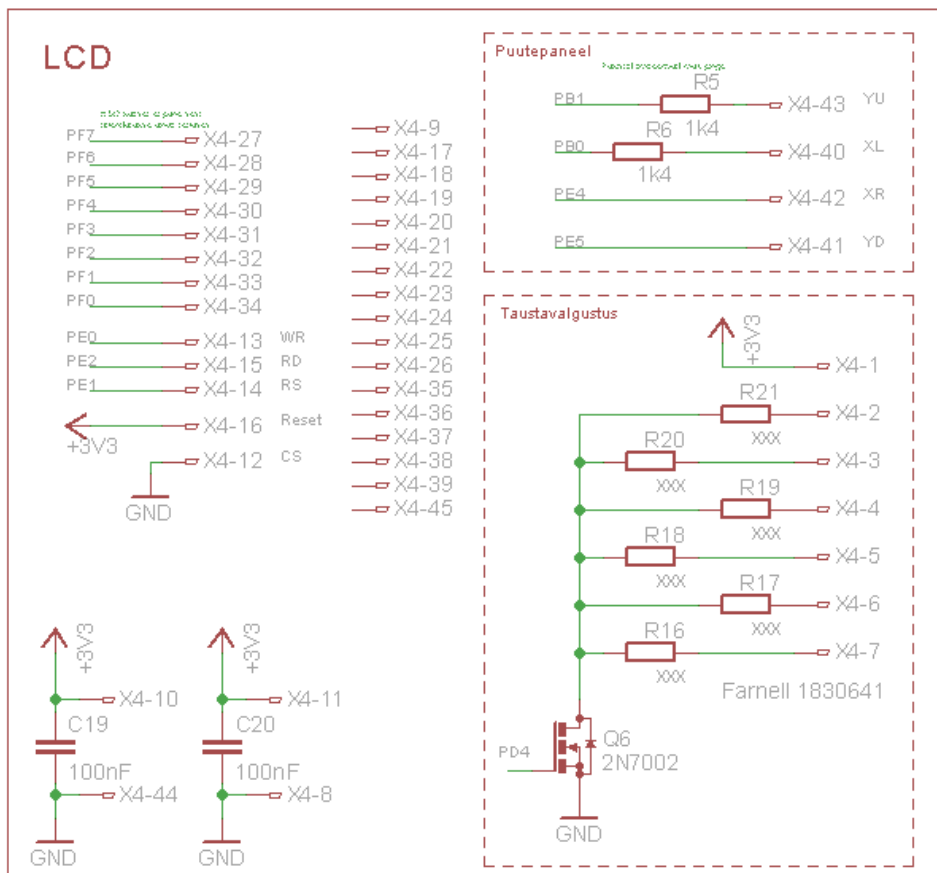
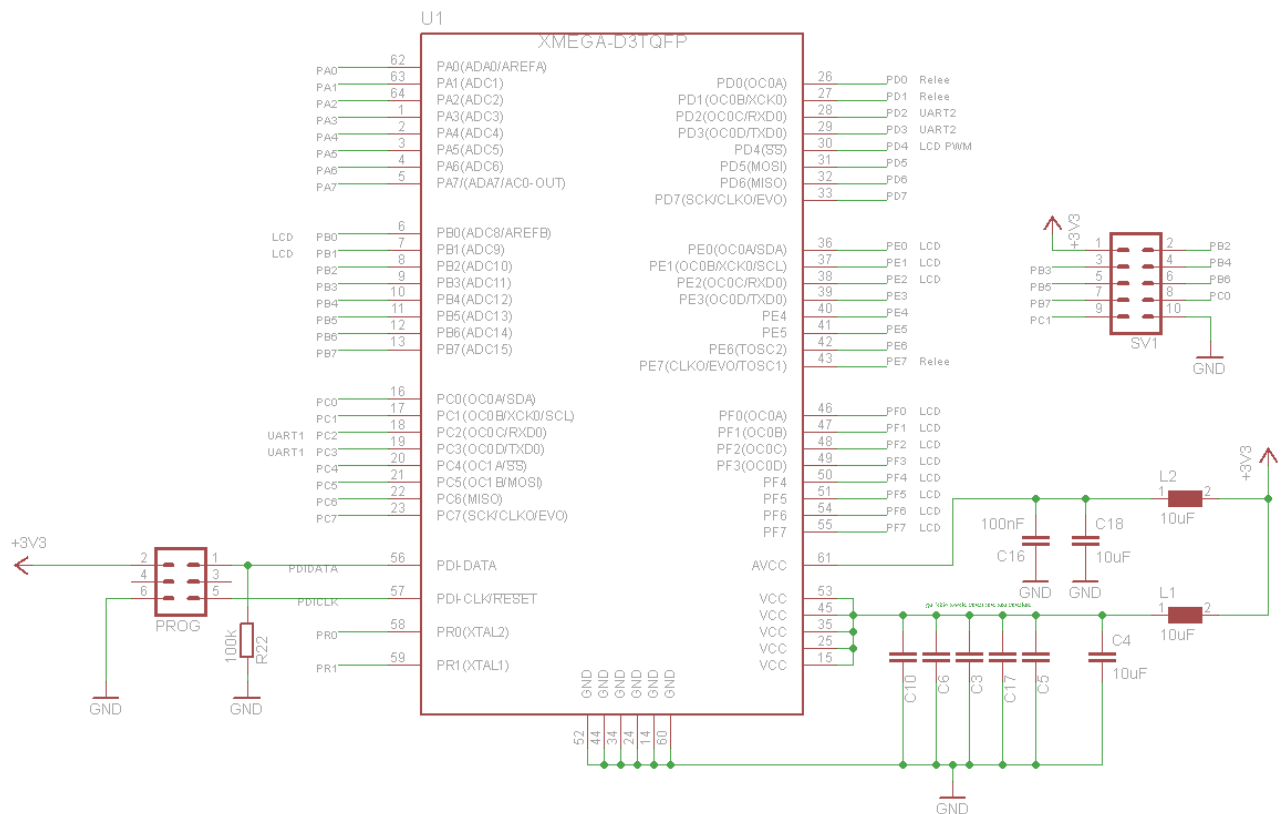
Lisa 3. Lisaplaadi PCB koos puutetundliku LCD ekraaniga.

Lisa 4. Lisaplaadi PCB paigutus.



Lisa 4. Lisaplaadi PCB paigutus.

Lisa 5. Lisaplaadi PCB skeem (Mittetäielik).



Lisa 5. Lisaplaadi PCB skeem (Mittetäielik).

Lihtlitsents lõputöö üldsusele kättesaadavaks tegemiseks ja reprodutseerimiseks

Mina **Erki Koplimes** (sünnikuupäev: **03.07.1993**)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
Puutetundliku LCD ekraani lisaplaat Pelltech'i pelletipõletile,

mille juhendajad on **Mihhail Pikkov ja Martin Jaanus,**

1.1. reprodutseerimiseks säilitamise ja elektroonilise avaldamise eesmärgil, sealhulgas TTÜ raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas TTÜ raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. Olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta kolmandate isikute intellektuaalomandi ega isikuandmete kaitse seadusest ja teistest õigusaktidest tulenevaid õigusi.

_____ (*allkiri*)

09.06.2015 (*kuupäev*)

METAANDMED

Töö pealkiri (eesti keeles): Puuetundliku LCD ekraani lisaplaat Pelltech'i pelletipõletile

Töö pealkiri (inglise keeles): Touchscreen LCD Extension Board for a Pelltech Pellet Burner

Autor: Erki Koplímets

Juhendaja(d): dotsent Mihhail Píkkov, dotsent Martin Jaanus

Kaitsmise kuupäev: 15.06.2015

Töö keel: est

Asutus (eesti keeles): Tallinna Tehnikaülikool

Asutus (inglise keeles): Tallinn University of Technology

Teaduskond (eesti keeles): Infotehnoloogia teaduskond

Teaduskond (inglise keeles): Faculty of Information Technology

Instituut (eesti keeles): Thomas Johann Seebecki elektroonikainstituut

Instituut (inglise keeles): Thomas Johann Seebeck Department of Electronics

Õppetool (eesti keeles): Siduselektronika õppetool

Õppetool (inglise keeles): Chair of Communicative Electronics

Märksõnad /kui on/ (eesti keeles):

Märksõnad /kui on/ (inglise keeles):

Õigused: