



TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND

Elektroenergeetika ja mehhatroonika instituut

MIKROKONTROLLERIL PÕHINEV INFOVAHETUSE JA SENSOORIKA MOODUL

MICROCONTROLLER BASED DATA EXCHANGE AND SENSORICS MODULE

BAKALAUREUSETÖÖ

Üliõpilane: Tom-Tristan Ots

Üliõpilaskood: 164543MAHB

Juhendaja(d): Martin Jaanus, vanemlektor
Alar Kuusik, PhD

Tallinn, 2019

AUTORIDEKLARATSIOON

Olen koostanud lõputöö iseseisvalt.

Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneridiplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

“.....” 201.....

Autor:

/ allkiri /

Töö vastab bakalaureusetöö/magistritööle esitatud nõuetele

“.....” 201.....

Juhendaja:

/ allkiri /

Kaitsmisele lubatud

“.....”201... .

Kaitsmiskomisjoni esimees

/ nimi ja allkiri /

LÕPUTÖÖ LÜHIKOKKUVÕTE

<i>Autor:</i> Tom-Tristan Ots	<i>Lõputöö liik:</i> Bakalaureusetöö
<i>Töö pealkiri:</i> Mikrokontrolleril põhinev infovahetuse ja sensorika moodul	
<i>Kuupäev:</i> 23.04.2019	<i>72 lk (lõputöö lehekülgede arv koos lisadega)</i>
<i>Ülikool:</i> Tallinna Tehnikaülikool	
<i>Teaduskond:</i> Inseneriteaduskond	
<i>Instituut:</i> Elektroenergeetika ja mehhatroonika instituut	
<i>Töö juhendaja(d):</i> vanemlektor Martin Jaanus	
<i>Töö konsultant (konsultandid):</i> Alar Kuusik, PhD	
<i>Sisu kirjeldus:</i> <p>Käesoleva bakalaureusetöö raames valmis mikrokontrollerile põhineva trükkplaadiga moodul, mis kuvab tudengivormeli prototüübi FEST19 piloodile LCD ekraanil üldist infot auto suhtlusvõrgustikust ning edastab suhtlusvõrgustiku kaudu elektroonika juhtsusele auto esiosas olevate sensorite kalibreeritud väärtuseid ja piloodi poolt muudetavaid parameetreid (nt. veepumba kiirus). Lihtsustatult öeldes jaguneb töö põhiline osa kaheks – trükkplaadi disainimine ning mikrokontrolleri programmeerimine (sh. ka kasutajaliidese disainimine). Lisaks sellele valmis ka LCD moodulile ja trükkplaatidele (disainiti veel 2 lihtsamat trükkplaati, millest räägitakse töö põhiosas lähemalt) korpus, mille autor disainis ning tootis freespingil. Mikrokontrolleri valimisel arvestati järgmiseid põhiparameetreid – võimalus töötleda analoogsignaale, suhelda autosisesse võrgustiku kaudu teiste trükkplaatidega, juhtida LCD ekraani ja lugeda/väljastada erinevaid digitaalsignaale. Põhiline eesmärk, mis autorile seati 2018 hooaja võistlusreisil oli ergonoomilise ning funktsionaalse kasutajaliidese disainimine, mille saavutamiseks kasutati spetsiaalset tarkvara ja innovatiivset armatuurlaua disaini. Lõputööst selgub, et trükkplaadi disainimisel on toetunud eelmise hooaja arendustele, kuid suurem osa tarkvarast, nii põhiline lähtekood (välja arvatud tiimi poolt arendatud kaks suhtlusteeki) kui ka kasutajaliidese ülesehitus, on töö autori arendatud.</p> <p>Lõputöö põhitulemusteks on eelnevalt kahe eraldiseisva trükkplaadi integreerimine ning nende osaline optimeerimine ja ergonoomilise kasutajaliidese disainimisele aluse panemine.</p>	

The outcome of this Bachelor's thesis is a module designed on a microcontroller-based PCB, which displays general data from the in-car communication bus to the pilot of the FEST19 formula student racecar prototype and also transmits calibrated sensors' information together with pilot-set parameters' (e.g. water pump speed) values to the electronical control unit. Simply put, the thesis' main body is divided into two main parts – designing the PCB and programming the embedded microcontroller (incl. user interface design). Additionally, a housing for the LCD module and PCBs (two simple auxiliary boards were designed, which will be further explained in the main body) was designed and manufactured by the author. While choosing the microcontroller, following main parameters were accounted for – possibility to process analog signals, communicate with other PCBs via in-car communication bus, drive an LCD screen and receive/transmit several digital signals. The main goal, which was set to the author during the 2018 season's competitive tour, was to design an ergonomic yet functional user interface, which was achieved by using specific software and innovative dashboard design. The thesis reveals, that the PCB design relied on last season's design, but most of the software, including the main source code(excluding a few libraries designed by the team) and the user interface structure, is developed by the thesis' author.

This thesis' main outcome is the integration and partial optimization of two previously separate PCBs as well as creating a foundation for an ergonomic user interface design.

Märksõnad: info kuvamine, andurite lugemine, CAN protokoll, ergonoomiline kasutajaliides, programmeerimine, trükkplaadi disain

ABSTRACT

<i>Author:</i> Tom-Tristan Ots	<i>Type of the work:</i> Bachelor Thesis
<i>Title:</i> Microcontroller based data exchange and sensorics module	
<i>Date:</i> 13.02.2015	<i>72 pages (the number of thesis pages including appendices)</i>
<i>University:</i> Tallinn University of Technology	
<i>School:</i> School of Engineering	
<i>Department:</i> Department of Electrical Power Engineering and Mechatronics	
<i>Supervisor(s) of the thesis:</i> Senior lecturer Martin Jaanus	
<i>Consultant(s):</i> Alar Kuusik, PhD	
<p><i>Abstract:</i></p> <p>In commercial markets there is a variety of racecar dashboard modules with analog inputs and the possibility to communicate via CAN bus, but they are usually gasoline engine oriented and the input amount as well as functionality is limited.</p> <p>This thesis describes three major parts of the development cycle – hardware designing, hand-soldering and testing the manufactured PCB, software developing (including GUI design). Together with the development description, arguments against using a commercially available module and previous season's solution is given.</p> <p>In the beginning of the thesis, there is a brief overview of previously used off-the-shelf and self-developed modules, and the to-be-built system requirements are given. Following these requirements, a justified microcontroller is selected and a PCB is designed. The MCU selection is made according to the project's requirements and the selected GUI developing library, which will be elaborated in the thesis' main body. There will also be calculations for sensor inputs' (low-pass) filters, which are necessary to isolate the data from the noisy environment.</p> <p>This thesis' outcome is a functioning display and sensorics module of the racecar FEST19, which is a prototype developed by student-only team Formula Student Team Tallinn.</p> <p>The thesis is written in Estonian and contains 72 pages.</p>	
<i>Keywords:</i> self-developed dashboard module, microcontroller, graphical user interface, processing analog and digital signals, dashboard design	

LÕPUTÖÖ ÜLESANNE

Lõputöö teema: **Mikrokontrolleril põhinev infovahetuse ja sensorika moodul**

Lõputöö teema inglise keeles: **Microcontroller based data exchange and sensorics module**

Üliõpilane: **Tom-Tristan Ots, 164543**

Eriala: **Mehhatroonika MAHB02/13**

Lõputöö liik: **bakalaureusetöö**

Lõputöö juhendaja(d): **Tallinna Tehnikaülikooli inseneriteaduskonna Elektroenergeetika ja mehatroonika instituudi vanemlektor Martin Jaanus
Alar Kuusik, PhD**

Lõputöö ülesande kehtivusaeg:

Lõputöö esitamise tähtaeg:

Üliõpilane (allkiri)

Juhendaja (allkiri)

Õppekava juht (allkiri)

1. Teema põhjendus

Käesolev bakalaureusetöö on oluline eelkõike tudengimeeskonna FS Team Tallinna arenguks. TTK ja TalTech tiim on iga aastaga hakanud aina tugevamat konkurentsi pakkuma suurte/tuntumate Euroopa tehnikaülikoolide meeskondadele, nagu näiteks TU Delft(Holland), GreenTeam(Stuttgardi ülikool, Saksamaa) ja AMZ(ETH Zürich, Sveits). Selle jätkamiseks tuleb kohtunikele silma jääda erakordsete lahendustega, mis meid eristaks konkurentide seast. Seetõttu on kriitiline vältida ostutooteid seal kus võimalik ning toetudes omandatud inseneriteadmiste peale arendada moodul, mis on optimeeritud(kaalu ja energiatarbe mõistes) ning suunatud spetsiifilisemale kasutusvaldkonnale.

2. Töö eesmärk

Töö eesmärgiks on panna alus tudengivormelile suunatud infovahetuse ja sensorika mooduli arendusele ning sellele kasutajaliidese disainimisele kasutades selleks ettenähtud teeki.

3. Lahendamisele kuuluvate küsimuste loetelu:

Kuidas integreeritaval trükkplaadil komponendid laduda töökindlalt ning optimaalselt?

Möötesisendite ahelate kontrollarvutuste teostamine

Kuidas programmeerimiskeeles „C“ luua ergonoomiline kasutajaliides?

Kuidas optimeerida vormeli kokpiti armatuurlaua disaini?

4. Lähteandmed

Lähteandmed seati meeskonna kapteni ning võistlussarja „Formula Student“ [1] reeglistiku poolt.

5. Uurimismeetodid

Sisendahelate skeemide valemarvutused

Projekteerimistarkvaradesse sisse ehitatud veatuvastusmeetodite kasutamine

Testimine spetsiaalsete tööriistadega(ostsiloskoop, multimeeter jpm)

6. Graafiline osa

Tabel 0.1 Armatuurlaua nuppude funktsioonid vormelil FEST18.....22

Tabel 0.2 Navigeerimislahenduse kriteeriumid.....25

Joonis 4.1 Üldistatud kuju programmi tööst.....**Error! Bookmark not defined.**

Joonis 4.5 Lihtsustatud CAN struktuur ja komponentide saadetavad sõnumid.....55

Lisa 4 Süsteeminõuete tabel.....65

Peamiselt on graafiline töö põhiosas, et anda paremat ülevaadet kirjeldatavast teemast, kuid osa materjalist on lisades põhiosa ruumi kokkuhoiu mõttes parema ülevaate saamiseks lõputööst.

7. Töö struktuur

Töö koosneb neljast peatükist.

Esimene peatükis käsitletakse eelmise hooaja lahenduse analüüsimist. Alapeatükkideks on trükkplaatide disainid ja üldine vormeli kokpiti disain. Esimene ja teine alapeatükk hõlmavad sensorika ja ekraani trükkplaatide skeemide ülevaatus ja kontrolli. Kolmandas alapeatükis analüüsitakse eelmise hooaja(FEST18) armatuurlaua ja kasutajaliidese disaini eraldi punktidenä.

Teises peatükis kirjeldatakse lähemalt armatuurlaua disaini uuendamist ning mooduli arendusprotsessi. Esimeses alapeatükis kirjeldatakse disainis tehtavat optimeerimist ning põhjendatakse uuendatut. Järgmisena kirjeldatakse trükkplaadi skeemide komponentide valikut kuueks punktiks jaotatud alapeatükis. Kolmandas ja neljandas alapeatükis käsitletakse

elektriskeemide ja trükkplaadi komponentide paigutuse disainimist, mis on jaotatud punktideks skeemiosade põhjal.

Kolmandas peatükis kirjeldatakse toodetud trükkplaadi komplekteerimistehnikat ja selle valideerimist ning eraldi punktina käsitletakse ka riistvara tarkvaralist testimist.

Viimases peatükis antakse väga üldistatud ülevaade programmeerimistehnikast ning mikrokontrolleri seadistamisest. Alampeatükkideks on CAN, ehk vormeli üldsuhtluse siini ning kasutajaliidese arendamise ülevaatlik kirjeldus.

8. Kasutatud kirjanduse allikad

S. Vesi, SENSOORIKAPLAAT TUDENGIVORMELILE FEST18, magistritöö, TTÜ 2018 [1]

Tudengitele suunatud tootearendusvõistluse „Formula Student“ reeglid [2]

Kasutajaliidese arendamise teegi LittlevGL dokumentatsioon [3]

„All About Circuits“, Tony R. Kuphaldt¹

9. Lõputöö konsultandid

Alar Kuusik, töö vormistamine.

10. Töö etapid ja ajakava

Konsulteerimine eelmise hooaja trükkplaatide arendajatega	15.08.18
Uue navigeerimise disaini välja töötamine	15.09.18
Mooduli sisendite/väljundite seadmine ehk nõuete nimekirja loomine	15.09.18
Kasutajaliidese teegi välja valimine	10.11.18
Trükkplaadi disainimine	30.11.18
Esimene revisjon joodetud ja testitud (sh. IC-de testimine)	15.01.19
Kasutajaliidese teegi integreerimise ja disainimise algus	16.01.19
Esimeselt revisjonilt välja tulnud vigade parandus ja plaadi tootmisesse saatmine	15.02.19
Programmeerimisel saavutatud põhifunktsionaalsus	01.04.19
Teine revisjon joodetud ja testitud	15.04.19
Juhendajale esimeseks läbilugemiseks saatmine	06.05.19
Paranduste sisse viimine ja teiseks läbilugemiseks saatmine	15.05.19
Töö lõplik versioon valmis	20.05.19

¹ <https://www.allaboutcircuits.com/textbook/>

SISUKORD

LÕPUTÖÖ LÜHIKOKKUVÕTE.....	3
ABSTRACT	5
LÕPUTÖÖ ÜLESANNE	6
EESSÕNA.....	12
LÜHENDITE JA TÄHISTE LOETELU	13
SISSEJUHATUS	15
1. VARASEMAD LAHENDUSED.....	17
1.1 Sensorika trükkplaat	17
1.1.1 Ülevaade arendusprotsessist	17
1.1.2 Analoogskeemide kontrollimine	18
1.2 Infovahetuse mooduli trükkplaat	21
1.2.1 Ülevaade arendusprotsessist	21
1.2.2 Skeemide ülevaade	21
1.3 Vormeli kokpiti disain	22
1.3.1 Kasutajaliidese disain	22
1.3.2 Armatuurilaua disain.....	22
2. INFOVAHETUSE JA SENSOORIKA MOODULI ARENDUS	24
2.1 Disaini ülevaade	24
2.1.1 Vormeli kokpiti disain.....	24
2.2 Mikrokontrolleriga trükkplaadi komponentide valik	28
2.2.1 Mikrokontroller	28
2.2.2 Kiirendusandurid	29
2.2.3 Väline muutmälu	30
2.2.4 <i>Non-volatile</i> välmälu	30
2.2.5 CAN siini transiiverid	31
2.2.6 LCD ekraan	31
2.3 Mikrokontrolleriga trükkplaadi elektriskeemid	32
2.3.1 Mikrokontroller	32
2.3.2 <i>SDRAM</i>	34
2.3.3 Kiirendusandurid	35
2.3.4 Välmälu ehk <i>flash</i>	36

2.3.5	Analoogsisendid	36
2.3.6	CAN suhtluse skeemid.....	37
2.3.7	Ekraani LED taustvalguse draiveri skeem.....	38
2.3.8	Pistikud	38
2.3.9	Toiteahelad	39
2.3.10	Ohutusahela e. <i>shutdown</i> -i lülitamine	40
2.4	Mikrokontrolleriga trükkplaadi disainimine	41
2.4.1	Pistikute paigutus.....	42
2.4.2	Analoogskeemide paigutus	43
2.4.3	Toiteskeemide paigutus	44
2.4.4	Dünaamilise muutmälu ehk <i>SDRAM</i> -i ja mikrokontrolleri paigutus	45
2.4.5	Kiirendusandurite paigutus	46
3.	INFOVAHETUSE JA SENSOORIKA MOODULI TESTIMINE	47
3.1	Riistvara jootmine ja testimine	47
3.1.1	Ülevaade jootmis- ja testimistehnikast.....	47
3.2	Riistvara tarkvaraline testimine	49
3.2.1	Mikrokontrolleri programmeerimine.....	49
3.2.2	CAN transiiverite testimine	49
3.2.3	<i>SDRAM</i> valideerimine.....	50
3.2.4	Välkmälu e. <i>flash</i> valideerimine	51
4.	PROGRAMMEERIMINE	52
4.1	Mikrokontrolleri perifeeriaste seadistamine.....	52
4.1.1	LTDC seadistamine	53
4.1.2	ADC seadistamine	53
4.2	CAN suhtluse programmeerimine	54
4.3	Kasutajaliidese programmeerimine	55
	KOKKUVÕTE	57
	SUMMARY	58
	Viidete loetelu.....	59
	Lisa 1: Mooduli korpuse algne disain	61
	Lisa 2: Mooduli korpuse lõplik disain	62
	Lisa 3: Kasutatud komponentide nimekiri.....	63

Lisa 4: Süsteeminõuete tabel	65
Lisa 5: Mikrokontrolleri konfiguratsioon.....	67
Lisa 6: Armatuurlaua disain FEST19	68
Lisa 7: <i>SDRAM</i> valideerimise fotomaterjal	69
Lisa 8: Kasutajaliidese arendusjärg.....	70
Lisa 9: Mooduli komponent - navigatsioonipadi	71

EESSÕNA

Käesolev lõputöö teema tekkis vajadusest optimeerida tudengivormeli FEST19 elektroonika poolt, viies kaks eraldiseisvat trükkplaati ühe plaadi peale. Esimene võrdlemisi suure LCD-ga isearendatud infovahetuse moodul valmis alles eelmisel hooajal vormelile FEST18. Ühtlasi oli see ka autori esimene aasta meeskonnas ning vahetult enne võistlusreisi haldas ta tollase mooduli kasutajaliideset, lisades sinna vajalikke funktsioone ja tehes parandusi ka üldisele lähtekoodile. Võistlusreisi käigus, aktiivselt arutades järgmise hooaja plaane meeskonnaliikmetega, selgus kaks eesmärki järgmiseks hooajaks. Esimese eesmärki seati vanemate, kogenumate tiimiliikmete poolt, milleks oli kahe trükkplaadi ühitamine – sensorika ja *dash*. See oli loogiline otsus - *dash*-i trükkplaadi mõõtmed tulenevad LCD ekraani mõõtmetest, mis tähendab, et trükkplaadile jääb väga palju vaba ruumi. Selle arvelt võidetakse nii juhtmestuse kui ka trükkplaatide kogumassi pealt. Teise eesmärgi seadis autor, pidades tiimi arenemist oluliseks – luua ergonomiline ja lihtsamini navigeeritav kasutajaliides, ning ühtlasi ka kirjutada lihtsasti hallatav ja loetav kood, et järgmisel hooajal selle projekti eest vastutav isik saaks arendusega jätkata. Ergonoomiline kasutajaliidese disain on oluline võistlustel kohtunike tähelepanu pälvimiseks, ning võib tuua tiimile just need kriitilised punktid, mis meeskonna järjekordselt disainikaitsmise finaali viib.

Lõputöö raames valminud trükkplaadi disainimisel olid abiks eelkõike vanemad meeskonnaliikmed, kes nõustasid praktiliste ja teoreetiliste teadmistega. Lisaks meeskonnaliikmetele olen tänulik ka moodulil kasutatud kasutajaliidese teegi autorile ja kogukonnale, kes seda avatud lähtekoodiga teeki haldavad ning arendavad ja foorumis aktiivselt aitavad. Samuti sooviksin tänada ka juhendajaid Martin Jaanust ja Alar Kuusikut meeldiva koostöö eest.

LÜHENDITE JA TÄHISTE LOETELU

<i>AIR</i>	Akumulaatorit isoleeriv rele(<i>Accumulator insulation relay</i>)
<i>back-end</i>	Tarkvara osa, mis tegeleb andmete töötusega, suhtlusega jms. riistvara lähedaste operatsioonidega
<i>Backlight(LCD)</i>	Taustavalguse allikas LCD tüüpi ekraanidele
<i>Breakdown voltage</i>	Diiodi läbilöögipinge
<i>Breakpoint</i>	Reaalajas lähtekoodi silumiseks kasutatav funktsioon
<i>CAN</i>	Sõidukitööstuses levinud siini standard, mis võimaldab mikrokontrolleritel informatsiooni vahetada. Eesliide „iso“ tähendab isoleeritud süsteemi. (<i>Controller Area Network</i>)
<i>Crimptool</i>	Kaablitangid
<i>dash</i>	Levinud väljend infovahetusmoodulile ja auto armatuuril olevale ekraanile
<i>Datalogger</i>	Andmeregistraator
<i>Discovery development kit</i>	Tarkvaraarendus platvorm
<i>D-pad</i>	Mängukonsoolide kontrolleritel levinud nupp navigeerimiseks
<i>Flash</i>	Välkmälu
<i>Flux(soldering)</i>	Räbusti, kasutatakse tina vooleomaduste parendamiseks
<i>front-end</i>	Tarkvara osa, mis tegeleb sisuliselt <i>back-end</i> 'i andmete esitamisega kasutajale ja/või registreerib kasutaja poolt tehtud muudatusi kasutajaliidesel
<i>GUI</i>	Graafiline kasutajaliides (<i>Graphical User Interface</i>)
<i>I2C</i>	Philipsi poolt välja töötatud jadasiini protokoll (<i>Inter-integrated circuit. I²C</i>)
<i>IC</i>	Integraallülituse kiip (<i>Integrated Circuit</i>)
<i>Interrupt(MCU)</i>	Teatud sündmuse esinemisel genereeritav katkestus mikrokontrollerile
<i>IP-rating</i>	Sissepääsu kaitse standardi reiting
<i>Joystick</i>	Juhtkang
<i>LDO</i>	Pingeregulaatori tüüp (<i>Low-dropout regulator</i>)
<i>LSB</i>	ADC vähim jaotise väärtus(<i>Least Significant Bit</i>)
<i>Non-volatile(memory)</i>	Mittelenduv mälutüüp - informatsioon säilib ka toite kadumisel
<i>Pad(PCB)</i>	Komponentide kontaktide monteerimispind
<i>p-channel MOSFET</i>	Metalloksiidi baasil väljatransistor, mis juhib voolu kui paisul puudub pinge
<i>Pinout configuration</i>	Mikrokontrolleri väljaviikudele seadistatud funktsionaalsuse konfiguratsioon
<i>Polygon</i>	Polügoon
<i>pull-down</i>	Takisti, mis hoiab liini loogilise madalana, kui liinil puudub signaal

<i>pull-up</i>		Takisti, mis hoiab liini loogilise kõrgena, kui liinil puudub signaal
<i>Quadrature encoder</i>	<i>rotary</i>	Enkoodri tüüp
<i>SCS</i>		Signaal, mis on <i>Formula Student</i> sarja reeglite poolt määratud süsteemikriitiliseks(<i>System Critical Signal</i>)
<i>SDRAM</i>		Suvapöördusmälu tüüp (<i>Synchronous dynamic random-access memory</i>)
<i>Shutdown</i>		Tudengivormeli sarjas reeglitega nõutud ahel, mis vajadusel tagab ohutuse ühendades kõrgpinge <i>AIR</i> -i lahti
<i>Step-down converter</i>	<i>buck</i>	Pingeregulaatori tüüp
<i>TVS</i>		Ülepinge kaitsediid(<i>Transient-voltage-suppression</i>)

SISSEJUHATUS

Käesoleva bakalaureusetöö eesmärgiks on optimeerida ning edasi arendada tudengivormeli meeskonna FS Team Tallinn kõige uuemat vormelit FEST19, integreerides kaks trükkplaati ja võimaldades ergonoomilist kasutajaliidese disaini. Ühitamisejärgselt langeb töö väljundile kaks põhilist ülesannet. Üks neist on vormeli eesosas olevate sensorite väärtuste lugemine, nende töötlemine kasutades kalibreerimisel saadud konstante ning saatmine elektroonika juhtüksusele(ECU). Teiseks ülesandeks on LCD ekraanil andmete kuvamine lõputöö raames disainitud graafilise kasutajaliidese abil. Lisaks neile kahele põhiülesandele tulevad trükkplaadile veel järgmised sisendid - navigatsioonipadja nupud(kasutajaliidesele liikumiseks, erinevate seadete muutmiseks), vormeli roolil ja armatuuril olevad nupud ning viimaks ka vormeli koguvõimsuse muutmise enkooder. Kõik nimetatud sisendid(va. navigatsioonipadi) saadetakse CAN siini kaudu ECU-le, mis omakorda suhtleb otse mootorite kontrollritega, andes nende nuppude väärtuste põhjal käsklusi nii sõidu alustamiseks kui ka sõidu ajal.

Võrdlemisi suure LCD-ga infovahetuse moodulit arendati alles eelmisel hooajal(2017/2018) esimest korda vormeli FEST18'le ning sensoorika trükkplaadi disaini on arendatud ühe meeskonnaliikme poolt kahe aasta jooksul vormelite FEST17 ja FEST18 peal. Viimasena nimetatud trükkplaadi skeemid näitasid selle aja jooksul töökindlust ning usaldusväärsust, kuid kindluse mõttes siiski need arvutused lõputöö käigus kontrollitakse ning vajadusel korrigeeritakse. Infovahetuse mooduli eelmise hooaja arenduse väljundiks oli töökindel trükkplaat väga algelise, kuid funktsioneeriva kasutajaliideselega. Lisaks sellele puudus ka igasugune dokumentatsioon ja kommentaarid lähtekoodis tarkvara ülesehituse kohta, sellest lähtuvalt ei olnud edasiarendus mõttekas. Selle asemel teostas autor uurimistööd ja analüüsis erinevate sardsüsteemide kasutajaliidese arendamiseks mõeldud tarkvarade/teekide omadusi ja kasutajatoe olemasolu, ning leitud esialgselt neljast võimalikust oli vaja välja valida üks lõplik, milleks sai LittlevGL [3]. Otsuse võeti vastu kahe põhilise omaduse tõttu – see on kirjutatud programmeerimiskeeles, millega autoril on varasemaid kogemusi - C keeles. Lisaks sellele omab aktiivset foorumit, kus suuremate kogemusteta programmeerija saab nõu pidada või abi küsida kogemustega inimestelt ja arendajatelt(sh. ka teegi loojaga). Õppeaasta jooksul valmib reaalne võidusõiduauto, millega minnakse tootearendusvõistlusele „Formula Student“ ning selline isearendatud lahendus infovahetuse moodulist pälvib palju tõenäolisemalt kohtunike tähelepanu ja poolehoidu, kui mõnda kallist, juba valmis arendatud ostutoodet kasutades.

Antud lõputöö raames ei saa kasutajaliidese disain lõplikult valmis ning on kirjeldatud kirjutamise hetkel oleva disaini arendusstaadiumit. Mooduli kasutajaliidese disainimine jätkub autori poolt käesoleva aasta juunikuu keskpaigani ning enne kaitsevärke minekut(4. juuli) koolitab välja uue meeskonnaliikme, kes järgmisel hooajal ning võistlusreisil selle mooduli eest vastutama hakkab.

Trükkplaadi disainimisel on kasutatud meeskonna sponsori Altium [4] tarkvara „Altium Designer“, ning modelleerimiseks/korpuse disainimiseks „Solidworks Student Edition“-i. Mikrokontrolleri seadistamisel kasutati STMicroelectronics'i „STM32CubeMX“ [5], ning tarkvara arendus toimus arenduskeskkonnas „Atollic TruStudio“ [6].

1. VARASEMAD LAHENDUSED

Alates FEST17 hooajast(2016/2017) võeti meeskonnas vastu otsus hakata arendama ise displeimoodulit(edaspidi *dash*), vältides sellega ostutoodete kasutamist, millega kaasnesid üldjuhul ka litsentsitasud. Enne seda hooaega olid kasutusel Motec'i tooted C125(elektrivormelitel FEST14-16) ja SDL3(sisepõlemismootoriga vormelitel FEST8-FEST12 ja ev. FEST13), mida kasutati nii CAN siini kaudu saadud info kuvamiseks, kui ka selle sama info logimiseks. Kuna sisepõlemismootoritega vormelitel puudus vajadus spetsiaalsete sensorite sisse lugemisele ning suur osa turul olevatest moodulitest on ekraani ülesehituse poolest ka orienteeritud just seda tüüpi ajamitega sõidukitele, oli ostutoote kasutamine õigustatud. Elektriajami(te)ga vormelile jäi funktsionaalsuselt SDL3 mudel väheseks, kuna oli vaja kuvada rohkem spetsiifilisemat infot ning lisaks sellele hakata ka sisse lugema lisandunud sensoreid, näiteks gaasi ja piduripedaali lineaarandureid, rooliasendi andur jpm.

Alates 2013 aastal elektriajamiga vormelile üleminekust, tekkis ka otsene vajadus eraldiseisva sensorite lugemise trükkplaadi järele. Tol aastal valminud auto(FEST14) peal leidis sensorika osa aset elektroonilise juhtüksusega samal trükkplaadil. Seda ülesehitust järgiti kuni 2016 aasta suveni ning sügisel, kui sai alguse vormeli FEST17 arendus, alustati eraldiseisva sensorika trükkplaadi arendamisega optimeerimise ja kaalu vähendamise eesmärgil [1]. Samal hooajal võeti vastu otsus ka dashi esmakordne tiimisisene arendamine, kaotades auto küljest järjekordse valmis ostutoote, vähendades sellega kulusid ja kaalu.

1.1 Sensorika trükkplaat

1.1.1 Ülevaade arendusprotsessist

FEST17-FEST18 hooajal oli antud trükkplaadi projekt pidevalt ühe meeskonnaliikme arenduses. Selle aja jooksul töötati välja töökindla ja väiksete mõõtmetega trükkplaat, millel oli võimekus lugeda kindla sagedusega analoogsisendeid, digitaalsisendeid-väljundeid, suhelda CAN siini kaudu elektroonika juhtüksusega ning I2C kaudu Boschi kiirendusanduritega. Aastal 2016, kui sai alguse FEST17 arendus, sai trükkplaadile kasutusse 144 väljaviiguga mikrokontroller, kuna lisaks analoogväärtuste lugemisele olid ka kõik kokpitis asetsevad nupud sensorikaplaadi hallata. Järgneval hooajal sai selgemaks PCB-l vajaminevate sisendite arv – analoogsisenditele reserveeritud läbiviike sai vähendada ning nõupidamise järel osad kohustused eelmisest hooajast integreeriti teistesse plaatidesse. Selle tulemusel saadi kasutusele võtta ST Microelectronicsi kõige väiksema läbiviikude arvuga mikrokontroller, millel oli neid 64. Lisaks sellele, kuna eelneva

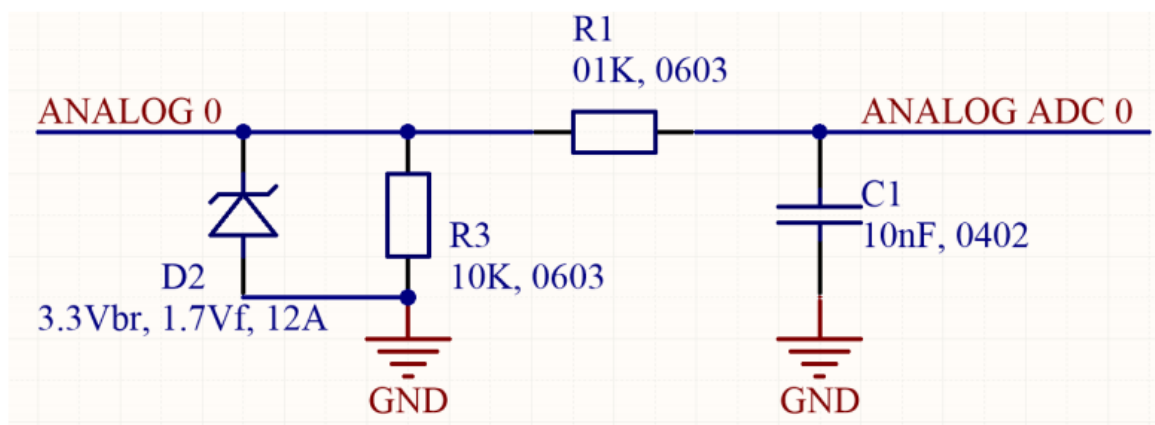
hooaja plaadil õnnestus meeskonnal tööle saada Bosch BMI160 kiirendusandur, siis täpsuse parandamiseks pandi neid plaadile kokku 4 tükki ja keskmistati nende väärtuseid. [1]

Kahe aastane arendustöö oli edukas – loodi töökindel trükkplaat, mis asetseb tudengivormelil FEST18 ja pidas hästi vastu kogu testperioodi ning võistlusreisi. Tänu plaadil asetsevatele kiirendusanduritele ja usaldusväärse ning kalibreeritud analoogväärtustele saab juhtsüsteem vajalikku infot töökindla stabiilsuskontrolli tagamiseks.

1.1.2 Analogskeemide kontrollimine

1.1.2.1 Madalapääsufiltrid

Komponenti on arendatud kaks aastat ja plaat oli tõestanud ennast töökindluse seisukohalt, viidi läbi siiski mõned kontrollivad arvutused, et veenduda kõikide kasutatud komponentide väärtuste õigsuses.



Joonis 1.1 FEST18 sensorikplaadi analoogskeem[5]

R1 – madalapääsufiltri takisti 10k oomi(autor kinnitas, et 01K on tegelikult 10K)

C1 – madalapääsufiltri kondensaator 10 nanofaradit

R3 – *pull-down* takisti 10k oomi

D2 – TVS kaitsedioid

Diiodi D2 eesmärk on antud liini kaitsta ülepinge eest, mis võib tekkida juhtmestuse veast või elektrostaatilisest pingest. Kuna antud liinil oleva anduri toitepinge on 3.3 volti, siis diiodi läbilöögipingeline on sobiv. *Pull-down* takisti eesmärgiks selle skeemi puhul on anda märku plaadil

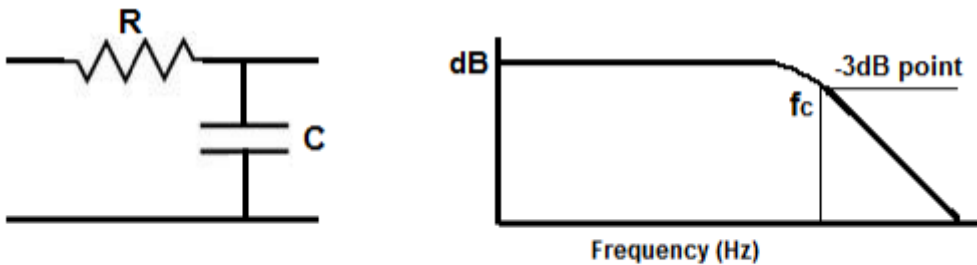
olevale mikrokontrollerile (mis edastab omakorda info ECU-le), kui anduriga peaks midagi juhtuma – näiteks üks kolmest juhtmest (maandus, signaal, toide) kaotab ühenduse nt. purunemise teel. Skeemil kaks parempoolset komponenti moodustavad esimese astme madalpääsufiltri, mille komponentide väärtused arvutatakse järgneva valemiga (valem 1.1):

$$F_c = \frac{1}{(2 * \pi * R * C)}, \quad (1.1)$$

kus F_c – äralõikesageduse punkt, kus signaali võimsus on langenud 3dB võrra, Hz

R – madalpääsufiltri takisti väärtus, Ω

C – madalpääsufiltri kondensaatori väärtus, F



$$f_c = \frac{1}{2\pi RC}$$

Joonis 1.2 madalpääsu filter [1]

$$f_c = \frac{1}{2\pi * (10000) * (10 * 10^{-9})} = \approx 1.5 \text{ kHz}$$

Üleval olevas arvutuses on näha, et antud skeemil on disainitud madalpääsufilter äralõikesagedusega ligikaudu 1.5 kHz, mis aitab kaasa kõrgsagedusliku müra signaaljuhtmest välja filtreerimisel, et analoog-digitaal konverteri lugem oleks usaldusväärne ning stabiilne.

Sarnase meetodiga kontrolliti üle kõik analoogskeemid, mille tulemusel selgus, et enamus filtritest on mitmekordselt üle dimensioneeritud, kuna maksimaalselt väljastatakse juhtsüsteemile CAN siini kaudu väärtuseid sagedusega 100 Hz. Sõltuvalt sellest faktist oli ratsionaalne madalpääsufiltri äralõikesageduse vähendamine, et signaali veel puhtamana hoida, saavutades sellega parema müraisolatsiooni - madalpääsu filtri 10nF kondensaator asendati 100nF kondensaatoriga. See annab meile ligikaudu 150 Hz äralõikesageduse, mis on väikse varuga, kuid siiski löikab ära enamus keskkonnast tingitud müra.

1.1.2.2 A/D konverterite sisendvoolu arvutamine

Lisaks kõrgsagedusliku müra eemaldamisele analoogsignaalist tuleb arvestada ka mõne skeemi puhul kasutatud ADC sisendvooluga rööbiti olevate pingejagurite komponentidega ja ADC sisendvoolust tekkiva nihkepingega. Selle arvutus toimub järgmise valemiga:

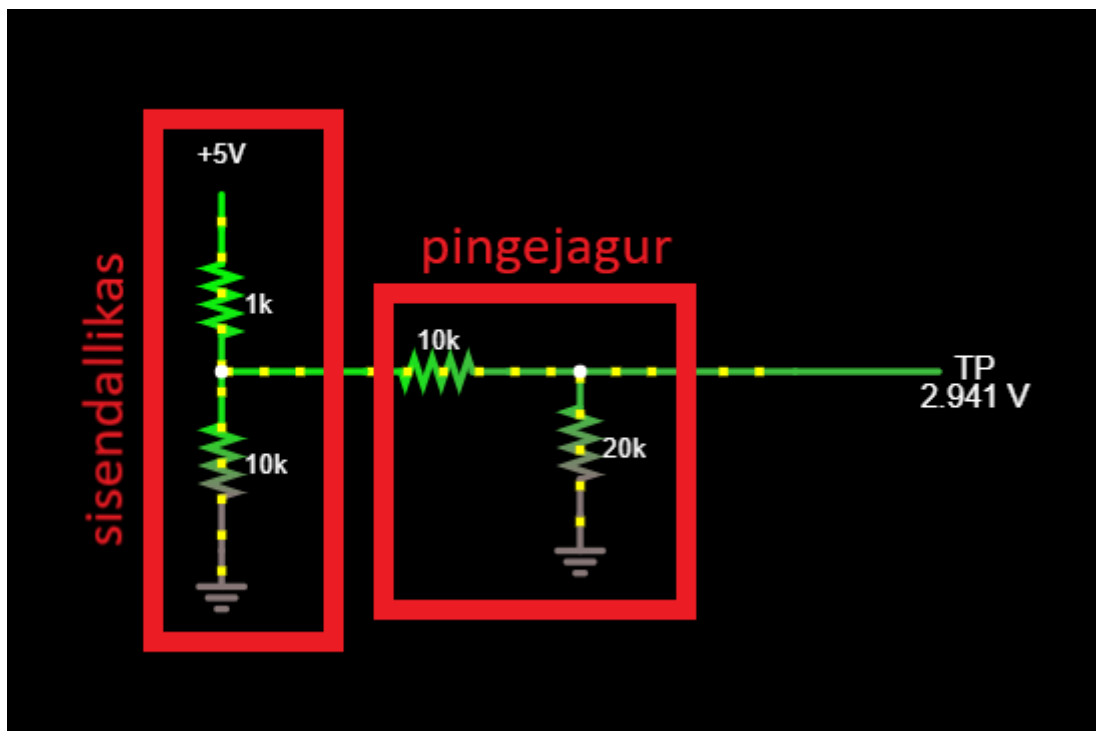
$$U(R_{AIN}) = R_{AIN} * I_{leke}, kus$$

$U(R_{AIN})$ – sisendvoolust tekkiv pingelang, V

R_{AIN} – sisendtakistus, Ω

I_{leke} – sisendvool, A

MCU andmelehest [7] saame, et maksimaalne sisendvool(juhul kui ei ületata sisendi nominaalpinget) on $1 \mu\text{A}$ ning sisendtakistus $10+1 \text{ k}\Omega$ (vt Joonis 1.2) + pingejaguri ja sisendallika takistus paralleelis, mis ekstreemumolukorras on $6.67 \text{ k}\Omega$ (10 paralleelis $20 \text{ k}\Omega$ -iga).



Joonis 1.2 Lihtsustatud kuju sisendallikast ning

mõõtesisendi skeemist(pingejagur) ja tema sisendallikast

Seega halvimal juhul võib tekkida lugemisse $17.67 * 10^3 \Omega * 1 * 10^{-6} \text{ A} = 17.67 \text{ mV}$ suurune mõõteviga, mis 12bitise resolutsiooniga konverteri puhul toob veaks 22 LSB-d 3V3 pinge juures, mis kalibreerimisjärgset väärtust mõjutab maksimaalselt ühiku võrra. See viga ei ole tühine, aga tegemist on ekstreemolukorraga ja tõenäosus, et mõõdise viga on pidevalt nii suur, on

minimaalne. Lisaks annab vajadusel selle vea mõjutust vältida tarkvaralise keskmestamisega, seega antud rakenduses see viga ei oma väga suurt mõju.

1.2 Infovahetuse mooduli trükkplaat

1.2.1 Ülevaade arendusprotsessist

2016. aasta suvel pandi alus tudengivormeli meeskonna FS Team Tallinn hüppeliseks arenguks. Tol ajal ees ootava FEST17 kontseptsiooni koostamisel võeti vastu otsus ise projekteerida ka näidikuteplokk ehk infovahetuse moodul, millel saaks kuvada vajalikku infot vormeli CAN siinilt. See lahendus oli suhteliselt algne, kuid andis arendajale suurema vabaduse kasutajaliidese disainimiseks programmeerimise teel. Järgneval hooajal, kui hakati disainima vormelit FEST18, võttis mooduli eest vastutav isik vastu otsuse, et trükkplaati disainides võtab eeskujuks STMicroelectronics'i *STM32F746G discovery kit*-i¹. Kuna plaadil kasutatud skeemid on STM poolt arendajatele kättesaadavaks tehtud, oli suhteliselt lihtne valida ainult vajalikud skeemiosad, mille abil oli võimalik formuleerida rakendusele optimeeritud trükkplaat. 2017/2018 hooaja ekraani moodulil oli juba konkreetsem kasutajaliides, millel olid erinevad leheküljed eritüüpi andmete jaoks ning mille vahel oli võimalik lehitseda enkoodritega. Lisaks sellele oli võimalik salvestada teatud piloodi poolt sisestatud andmeid mittelenduva väikmälukiibi peale.

1.2.2 Skeemide ülevaade

Nagu eelnevalt mainitud, siis suurem osa trükkplaadil vajaminevatest komponentidest valiti samad, mis on kasutusel samale mikrokontrollerile baseerual STM-i arendusplatvormil, et suurendada tõenäosust toimiva trükkplaadi valmimiseks. Valitud integraalkomponentide seadistamine ning nende funktsionaalsuse kasutamine ei olnud keeruline ja need ei valmistanud probleemseid olukordi arendusprotsessis. Sellest tulenevalt soovitas eelnev projekti eest vastutav meeskonnaliige kaaluda sama tehnikaga lähenemist aja kokkuhoiu ning töökindluse eesmärgil.

¹ <https://www.st.com/en/evaluation-tools/32f746gdiscovery.html>

1.3 Vormeli kokpiti disain

1.3.1 Kasutajaliidese disain

Kasutajaliidese disaini vormeli FEST18 ekraanimoodulile arendas nullist sel hooajal projekti eest vastutav olnud isik. Lõpplahendus nägi puhas(vt Joonis 1.3), kuid väga toores välja, aga samas täitis täies mahus oma eesmärgi. Kahjuks puudus arenduskäigust igasugune dokumentatsioon, ning ka C keeles kirjutatud koodi mõistmisele oleks kulunud liialt palju ajalist ressursi, kuna puudusid kommentaarid koodiridade vahelt.

1.3.2 Armatuurlaua disain

Vormeli FEST18 armatuur koosnes infovahetuse moodulist, kolmest enkoodrist ja neljast nupust(vt Joonis 1.4), mille funktsioonid on toodud all olevas tabelis(Tabel 1.1).

	Funktsionaalsus	Signaali lõpp-punkt
Enkooder 1 (POWER/SAVE)	Auto mootorite koguvõimsuse muutmine ja pikemal peale vajutamisel muudatuste välkmällu salvestamine	Ekraanimoodul
Enkooder 2 (PARAMETER)	Lehekülje vahetamine kasutajaliidesel	Ekraanimoodul
Enkooder 3 (VALUE)	Parameetrite vahel valimine ja nende redigeerimine	Ekraanimoodul
Nupp 1	Sõidurežiimi lülitamine	Sensorikamoodul
Nupp 2	Mootorikontrollerite lülitamine	Sensorikamoodul
Nupp 3	Ohutusahela kinni ühendamine	Ekraanimoodul
Nupp 4(vasakul, punane)	Ohutusahela lüli	Järgmine ohutusahela lüli

Tabel 1.2 Armatuurlaua nuppude funktsioonid vormelil FEST18

Nupud, mille lõpp-punktiks on sensorika trükkplaat(nupud 1 ja 2), loeti sisse ning väljastati üldise suhtlusliini kaudu elektroonika juhtüksusele, mis siis vastavalt lülitas sisse mootorite kontrollereid või andis kontrolleritele käsu sõidurežiimi minekuks.

Neljas, punane nupp vasakul pool ekraani(Joonis 1.4) on reeglitega [2] nõutud ohutuse eesmärgil **alpunktis T 11.4.4**. Nuppu vajutades katkestatakse *shutdown* ahel ning kõrgpingereleed, ehk *AIR*-id, avatakse. See nupp on puhtalt mehaaniline – vajutades avab ahela, keerates lähtestatakse algolekusse ning taassuletakse ahel. Lisaks sellele oli ekraani korpuse sisse freesitud kolm võistlussarja reeglitega nõutud vea indikatsioonitule ava ja kolm lühendit – AMS, IMD, BSPD. Juhul kui tekib süsteemis üks neist kolmest kriitilisest veast, antakse sellest kohe juhile märku võimsa valgusdiodiga ning avatakse *shutdown* ahel.



Joonis 1.3 FES18 armatuurlaud(hammasratta ja keti asemel oli varem rool)

2. INFOVAHETUSE JA SENSOORIKA MOODULI ARENDUS

2.1 Disaini ülevaade

2.1.1 Vormeli kokpiti disain

Eelmisel hooajal kasutajaliidese navigatsioon toimus läbi enkoodrite, mis valiti puhtalt selle omaduse tõttu, et neid saab peale keeramise ka vajutada nagu klassikalist nuppu. Kuna tegemist oli siiski *quadrature rotary encoder*-itega, ei piisanud klassikalisest enkoodri tarkvarast, et selle liikumist veatult jälgida. Eelmisel hooajal tuli tihti ette olukordi, kus keerates seda pidevalt samas suunas, hakkas kasutajaliidese kursor pendeldama kahe lehe või parameetri/koguvõimsuse väärtuse vahel.

Antud lõputöös käsitletud trükkplaadi revisjone teostades ei olnud autor ega ka eelneval hooajal selle eest vastutanud meeskonnaliige sellest teadlik ning arvas, et probleem on puhtalt tarkvaraline, mistõttu mootorite koguvõimsuse reguleerimiseks kasutatakse sama enkoodrit ilma töökindluseks vajalike lisadeta.

Käesoleva punkti alguses kirjeldatud probleem oli ka põhjuseks uue disaini arendamisele – kasutajaliidesele navigeerimiseks kasutatud kaks enkoodrit nõudsid enda asemele töökindlamat lahendust.

2.1.1.1 Uus lahendus navigeerimiseks

Uue lahenduse disainimisel tuli järgida peainseneri ning võistlussarja nõudeid [2] süsteemile, mis on toodud allolevas tabelis (Tabel 2.1).

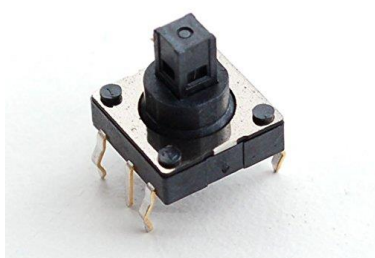
Kriteerium	Põhjendus
Veekindlus	<ul style="list-style-type: none">• Avatud kokpiti satub vesi muutlikust ilmastikust tingituna• Võistluseelisel tehnilisel kontrollil pritsitakse 5 minutit auto kohal vett – iga väiksema märk puudlikust veekindlusest tähendab võistlemiseks eelduste puudumist ning kontrolli uuesti läbimist
Töökindlus	<ul style="list-style-type: none">• Navigeerimine peab olema töökindel – lehekülgede vahetamine kasutajaliidesel ning seadistamine peab alluma konkreetselt kasutaja soovidele.

Tabel 2.1 Navigeerimislahenduse kriteeriumid

Esimene idee, mida tiimikaaslastega arutati, oli kasutada *D-padi*, ehk sisuliselt viiesuunalist lülitit (vt Joonis 2.1), millest iga suuna suhtes liigutamine on selgelt eristatav – suletakse ainult see kontakt, mille suunas nuppu liigutada. See lahendus nõudis endaga kahe lisa detaili disainimist – eraldi trükkplaati, kuhu nupp kinnitada ning nupu külge katet, mis võimaldaks seda liigutada kolmes sihis – üles-alla, vasakule-paremale ja sisse-välja.

Teiseks lahenduseks oli *joystick*-i (vt Joonis 2.2) laadi toode, mis käitub sarnaselt *D-pad*ile. Selle lahenduse eeliseks oli lisadetailide disainimise vajaduse puudumine.

Viimane pakutud lahendus oli kasutada navigatsioonipatja (vt Joonis 2.3), mis koosnes viiest erinevast lülitist. See idee nõudis lisadetailiks trükkplaati, kuhu nupud tasapinnaliselt kinnitada.



Joonis 2.1 5-suunaline navigatsioonilüliti



Joonis 2.2 Joystick



Joonis 2.3 Navigatsioonipatja lülitid

Navigatsioonilahenduste võrdlustabelist, et selgelt ratsionaalseim lahendus oleks *joystick*, kuid siiski tuli luua võrdlustabel (vt Tabel 2.2), kus võrdluselementideks on hind, *IP-rating* ning vajaminevate lisadetailide hulk, et välja selgitada optimaalseim lahendus.

Lahendus	hind	<i>IP-rating</i>	Lisadetailide hulk
5-suunaline navigatsioonilüliti Adafruit 5-way nav.switch ¹	9 €	puudub	2
<i>Joystick</i> Apem NVH1D1C0CP2S ²	104.2 €	IP69K	0
Navigatsioonipadja komplekt MEC NAVIKITSM09WL ³	22.02€ (sponsorsoodustusega 11.93€)	IP67	1

Tabel 2.2 navigatsioonilahenduste võrdlustabel

Kõige odavam lahendus oleks kasutada viie suunaga lüliti, kuid sellel puudub üks tähtsimatest omadustest rahuldamiseks disaini nõudeid – *IP-rating*, siis sellest lähtudes antud lahendus ei sobi. Järele jääb kaks võimalikku lahendust, millel on olemas tootja poolt piisav *IP-rating*’u sertifikaat. Kuigi navigatsioonipadjale on vaja disainida trükkplaat ning osta tootmisteenus, on lõppkokkuvõttes siiski see lahendus kõige ratsionaalsem ja piiratud rahalise ning ajalise ressursi suhtes ökonoomsem. Trükkplaadi mahutab edukalt sponsorkorras tasuta saadud trükkplaadi paneelile, ning nuppude jootmine trükkplaadile ei nõudnud rohkem kui 30 minutit. Tagamaks padja veekindlust, pihustati ka peale jootmist trükkplaati mõlemalt poolt hästi voolava dielektrilise spreilakiga, et vältida olukorda, kus voolav vesi võib lühistada omavahel nuppude kontakte. Valmislahendust ning trükkplaadi disaini on toodud lisades (vt Lisa 9).

¹ <https://www.amazon.com/Adafruit-Thru-hole-Navigation-switch-ADA504/dp/B01BMRTDJI>

² <https://www.elfadistrelec.ee/et/compact-joystick-apem-nvh1d1c0cp2s/p/30100621>

³ <https://ee.farnell.com/mec/navikitsm09wl/switch-kit-spno-black/dp/1390524?st=navigation>

2.1.1.2 Mooduli korpuse disain

Korpuse disainimisel põhiline nõue oli veekindlus just monokoki poole jääval osal - see puutub kokku väliste ilmastikuteguritega. Lisaks sellele tuli ka disainis arvestada veateate indikatsioonitulede trükkplaadiga, mille täpsem kirjeldus ja vajadus on selgitatud **punktis 2.1.1.3**. Korpus modelleeriti varasemate kogemuste olemasolu tõttu programmis „Solidworks“.

Korpuse võimalikud maksimaalsed mõõtmed seati vormeli põhikoostu eest vastutava meeskonnaliikme nõudmiste baasil - disain peab olema võimalikult minimaalsete mõõtmetega, et ei tekiks kontakti roolisüsteemiga. Sellest lähtudes valmis esimene mudel mõõtudega 120x85x18.5 mm, mille ülevaatlikud kuvatõmmised on lisades (vt Lisa 1). See disain jäi siiski ainult mudeli kujule, kuna tekkis korpuse nähtava osa disainiks parem idee – eemaldada ringikujulised polükarbonaadist „aknad“ ja eestvaatel nähtavad tähtede süvendid liigutada valgusdiodide trükkplaadi kohale, ning teha need läbivaks. Kasutades sama tehnikat, mida kasutati tähtede süvendite loomiseks, modelleeriti ka individuaalsed polükarbonaadist tähed, mis peale tootmist vastavatesse süvenditesse liimiti. Lõpptulemus oli väga rahuldav ning pilkupüüdev, aga tootmine võttis rohkem aega kui eelmise aasta lahendus. Mudeli ja valmis korpuse disain on toodud lisades (Lisa 2).

Korpuse kõik detailid tootis lõputöö autor ProtoTubA OÜ freespinki kasutades, materjalideks on must POM-C ja polükarbonaat. Materjalivalik on selline, kuna POM omadused on plasti kohta väga head, ning toon sobib kokku vormeli süsinikust monokokiga. Polükarbonaati kasutati ekraanikatteks ja läbipaistvate tähtede jaoks.

2.1.1.3 Indikatsiooni tulede trükkplaat

Käesoleva töö raames disainitava mooduli korpuses peitub kaks trükkplaati – põhiline trükkplaat mikrokontrolleriga ning trükkplaat valgusdiodidega kriitiliste veateadete inditseerimiseks. Inditseeritakse järgmiseid kriitilisi signaale – AMS viga, IMD viga ning TS OFF. AMS viga tähendab probleemi või ebanormaalselt olekut (nt aku tühjenemine alla lubatud taseme) akuhaldussüsteemiga. IMD viga tähendab, et akukasti ja auto kere vahel on tekkinud väga ohtlik ühendus. TS OFF märgutuli annab pilloodile märku, et kontrolleris ning auto elektrimootorites olev pinge ei ületa 60 V – akupakki isoleerivad releed on lahti. Kuna antud signaalid on süsteemikriitilised (SCS), siis LEDid põlevad ainult sel juhul, kui signaal on loogiliselt madala väärtusega, ehk 0 V. Sellest tingituna lähevad tuled põlema ainult kahel juhul – kui signaali väljastav trükkplaat tuvastab vea või signaali edastav juhe puruneb, ehk kaob ühendus inditseeriva trükkplaadi ning signaali allika vahel. Juhul kui ühendus on olemas ja kõik toimib nii

nagu peab, ei erita valgusdiodid valgust. Selle loogika teostamiseks on vaja kasutada P-kanaliga transistore, ehk antud juhul p-kanaliga MOSFET. Nende komponentide viimine eraldi trükkplaadile tagab organiseerituse – armatuurlaua taga ei ole juhtmepundart ja lahtiseid komponente, vaid kõik on korpuse poolt ilusti paigas. PCB disain on alloleval joonisel (Joonis 2.4).



Joonis 2.4 Märgutulede trükkplaadi disain

Kuna trükkplaadi disainimise hetkel ei olnud kindlaks tehtud signaalide pinge, siis takistitega R01-R06 saab tekitada pingejagureid, et tagada vajadusel transistori täielik juhtimine. LEDide arvu põhjenduseks on esteetilisus – polükarbonaadist tähtede täielikuks illumineerimiseks oli vaja iga lühendi alla vähemalt kahte valgusallikat. Sellest tulenevalt on kolmetäheliste lühendite(AMS, IMD) all kaks ning keskmise all kolm LEDi(TS OFF). Ülejäänud takistid on diodidest läbiva voolu kontrollimiseks – nii peavad need töökindlamalt/kauem vastu.

2.2 Mikrokontrolleriga trükkplaadi komponentide valik

Eelmise hooaja(2017/2018) lõpus võeti vastu otsus viia järgmisel hooajal(2018/2019) ühele trükkplaadile kaks varasemalt eraldiseisvat trükkplaati – infovahetus ja sensorika. Tuginedes faktile, et mõlemad projektid olid eelmisel hooajal usaldusväärsed ja töökindlad(st. töötasid nii nagu oli projekteeritud), jäid suurem osa komponente samaks. Vaatamata sellele faktile, siiski viidi läbi täiendav analüüs komponentide valikule ja skeemide disainile.

2.2.1 Mikrokontroller

Lõputöö raames disainitava trükkplaadi tuumaks on STMicroelectronics'i mikrokontroller, mis peab omama piisavat võimekust, et lugeda mitmete sensorite väärtuseid, kuvada pilti LCD ekraanile, lugeda erinevaid sisendeid, töödelda CAN siinist loetud andmeid jpm. Esimene kitsendus mikrokontrolleri valiku puhul on kõikidel meeskonna disainitud digitaalloomika trükkplaatidel ühine – kasutatakse STM F7-seeria mikrokontrollereid. See ühine kokkulepe tagab elektroonika alammeeskonnale tarkvaraarenduses mitu eelist, milleks on näiteks arendatud suhtlusteekide(I2C, CAN jne) probleemideta jagamine ning saab mikrokontrolleri seadistamisel/tarkvara arendamisel pakkuda üksteisele abi. Selle põhjal esialgne mikrokontrolleri valik tehti süsteeminõuete tabeli põhjal (vt Lisa 4). Esialgse arvestuse järgi läheb vaja kokku 120

mikrokontrolleri läbiviiku erinevate perifeeriatega ja sisendite/väljunditega töötamiseks. Lisaks sellele tuleb jätta ka varu, et vajadusel saaks lisada trükkplaadile funktsioone ilma mikrokontrolleri välja vahetamiseta. Sellele tuginedes valiti esialgu 208 väljaviiguga (millest kasulikke 159) mikrokontroller STM32F777BIT¹ järgnevatel põhjustel: omab kõiki vajalikke perifeeriaid ja mitukümmend läbiviiku jääb reservi.

Peale mikrokontrolleri välja valimist tuli uurida ka võimalusi kasutajaliidese arendamiseks. Uurimistöö tulemusel selgus peagi kõikide valitud tarkvarade puhul üks sarnane omadus – nad on suunatud STM poolt müüdavatele kompleksetele arendusplatvormidele (*discovery kits, development kits*). Sellest järeldades tuli programmeerimise lihtsustamiseks ja ühe õppeaastaga kindlalt töötava lahenduse välja töötamiseks esialgu kasutusele võtta mõni mikrokontroller, mis oleks *GUI* arendamiseks ettenähtud tarkvara poolt toetatud – see annab võimaluse foorumites nõu pidada kogemustega inseneridega.

Uue mikrokontrolleri valiku tegemisel toetuti eelmise hooaja trükkplaadi disainile, mis oli suuresti disainitud ühe eelmises lõigus kirjeldatud platvormi skeemidele tuginedes, sh oli kasutusel sama *MCU*, mis ühel arendusplatvormil (täpsemalt STM32F746G-DISCO²). Selleks hetkeks oli trükkplaadilt nõutav funktsionaalsus rohkem selginenud ning esialgne vajalik läbiviikude arv oli 120. Varem kasutatud mikrokontrolleri tootekoodiks on STM32F746IGT6 [7], mis sarnaselt esimesena valitud kiibiga omab kõiki vajalikke perifeeriaid ning jätab ka ruumi jooksvateks muudatusteks – kasulikke läbiviike on kokku 140, mis jätab paarkümmend viiku vabaks. Seega esimese trükkplaadi disain tehti sama mikrokontrolleri põhjal, mis eelmisel hooajal kasutati.

2.2.2 Kiirendusandurid

Aastal 2016 esimest korda tudengivormelil FEST17, ning ka järgneval hooajal vormel FEST18, sensorika trükkplaadil kasutatud kiirendusandur(id) Bosch BMI160 [8] on tõestanud end töökindluse ja vajaliku täpsusega [1]. Nende aastate käigus on välja arendatud I2C suhtluseks vajalikud tarkvarateegid, mis nende kasutamise jätkamise võrdlemisi lihtsaks muudab. Eeliseks on nende soodne hind, väga väikesed mõõtmed ning vähene lisakomponentide vajadus töötamiseks – komponendi jalajälg trükkplaadil on minimaalne. Selle põhjal jätkati samade kiirendusandurite kasutamist suurema edasise analüüsita.

¹<https://www.mouser.ee/ProductDetail/STMicroelectronics/STM32F777BIT6?qs=%2Fha2pyFadug9h41yNQpkwB3pSsiC7ZqX9tcxwUuWsghdSVykUOBMKw%3D%3D/2535501>

² <https://ee.farnell.com/stmicroelectronics/stm32f746g-disco/dev-board-cortex-m7-discovery/dp/2480961>

2.2.3 Väline muutmälu

Väline ekraanipuhvri kiip, milleks on antud trükkplaadi puhul *SDRAM* on vajalik võrdlemisi suurel LCD ekraanil(480x272 pikslit) pildi kuvamiseks. Kuna nimetatud ekraanile kuvatakse pilti formaadis RGB888, ehk iga piksli jaoks on vajalik $8+8+8 = 24$ bitti, siis antud juhul on vaja terve ekraani kuvamiseks $480 * 272 * 24 = 3133440$ bitti, ehk ligikaudu 392 kB. Valitud mikrokontrolleri andmelehes [7] on kirjas, et integreeritud muutmälu suurus on 320kB, ehk 2560000 bitti, mis ilmselgelt ei ole piisav ekraanipuhvrina kasutamiseks – see põhjendab kiibi vajalikkust.

Põhjalikum analüüs aja puudusel kiibi valiku suhtes(korpuse suurus, mälu maht jne) jäi teostamata ning kasutati sama dünaamilist muutmälu, mis eelneval hooajal – MICRON MT48LC4M32B2P-6A IT [9]. Kiibi muutmälu mahuks on 134217728 bitti, ehk ligikaudu 16777 kB, mis on kordades rohkem kui reaalselt vaja läheb antud rakenduses. Valiku põhjus oli lihtne, kuid piisav – puudus varasem nii spetsiifilise elektroonika kasutamise kogemus ning ekraanipuhvrit mahutava kiibi mitte töötamine tähendaks LCD ekraani mitte töötamist, mille tulemusel kaotatakse kriitilist ressursi – aega.

2.2.4 *Non-volatile* välmälu

Non-volatile (ehk „mittelenduv“) välmälu on mäluelement, mille omaduseks on kirjutatud informatsiooni säilitamine toite eemaldamisel. See on disainitaval moodulil vajalik pilootide profiilide parameetrite ja vormeli läbisõidu salvestamiseks(täpsemalt **peatükis nr.4**). Antud projektis on kasutatud selleks NOR-tüüpi mälu, mille eeliseks NAND-tüübi ees on suurem lugemiskiirus [10] – mikrokontrolleri alglaadimise ajakulu väheneb.

Aja kokkuhoiu eesmärgil kasutati trükkplaadi disainimisel sama NOR välmälu kiipi [11], mis on kasutusel eelnevalt kirjeldatud arendusplatvormil STM32F746G-DISCO. Antud komponendil on 16 MB mälumahtu(piisav eelnimetatud ülesannete täitmiseks) ning korpus *SOIC-8*, mille mõõdud on suhteliselt väikesed. Lisaks sellele vajab see vähe lisakomponente(ainult ühte toidet siluvat kondensaatorit ning *pull-up* takistit), mis teeb trükkplaadil vajamineva jalajälje suuruse minimaalseks. Sama kiibi valimise ajendiks oli ka sellega seotud tarkvara arendamisele kuluva aja vähendamine - mällu kirjutamine nõuab ka tarkvaralist draiverit, mille välja töötamine spetsiifiliste teadmiste puudumisel on liialt aega nõudev. STMicroelectronics'i arendusplatvormide näidiskoodid ja nende peal kasutatud komponentide draiverid on kättesaadavad kõigile ning seetõttu puudus vajadus pikemaks valiku analüüsiks.

2.2.5 CAN siini transiiverid

Trükkplaadi nõuete tabelist(vt Lisa 4) on näha, et kasutusel on kaks erinevat mikrokontrolleri CAN suhtluse kontrollerit. CAN1 on üldise auto võrgustiku suhtluse jaoks ning on isoleeritud, CAN2 on kasutusel logimise eesmärgina – trükkplaat loeb kõiki sõnumeid üldiselt suhtluse siinilt ning saadab need töötlemata edasi andmeregistraatorisse(*datalogger*).

Põhilise CAN siini transiiveriks valiti meeskonnas sel hooajal isoleeritud CANi kiip Analog Devices ADM3053BRWZ [12]. Eelmisel hooajal auto üldsiinil kasutuses olnud isoleerimata CANi transiiver, mis on Texas Instrumentsi [13] toodang, on antud trükkplaadil kasutuses teisel CAN kontrolleril(CAN2). Mõlemad transiiverid on kasutuses olnud meeskonnas üldise suhtlussiini peal, kuid eelmise hooaja lõpus tuli tõdeda, et isoleerimata CAN üldise suhtluse rakendusse ei sobi – meie keskkond on selle jaoks liiga mürarikas ning tekitab suhtlusesse probleeme. Kuna logimisseade on infovahetuse ja sensorika mooduli kõrval, ja moodul on auto kokpiti ees otsas, mootorite kõrgpinge kontrollerist kaugel, siis häiringud on minimaalsed.

2.2.6 LCD ekraan

LCD ekraani valik põhjalikumat analüüsi ei saanud, kuna eelmisel hooajal sellega mingeid probleeme ei esinenud ning ekraani suurus ning hind on sobiv oma rakenduseks. Ekraani¹ tootja on „Newhaven Display“, resolutsioon on 480x272 pikslit, 4.3“ diagonaaliga. Displei ühendub läbi ribakaabli trükkplaadil oleva vastava pistikuga. Lisaks sellele on vaja LCD tüüpi ekraanidele taustavalgust – see on integreeritud ekraanimooduli korpusesse LEDide maatriksi näol. Kuna tegemist on valgusdioodidega, siis ekraani töökindluse tagamiseks on vaja sellele riistvaralist draiverit, mis siluks pinget ja piiraks voolu vastavalt andmelehe soovitusetele. Kasutuses olev toiteskeem on võetud varasemalt mainitud STM-i arendusplatvormi skeemide seast.

¹<https://www.mouser.ee/ProductDetail/Newhaven-Display/NHD-43-480272EF-ATXL-T?q=sGAEpiMZZMu%2FRY1bNe3bOzXdmhjylONaEZeRDUDaU14%3D>

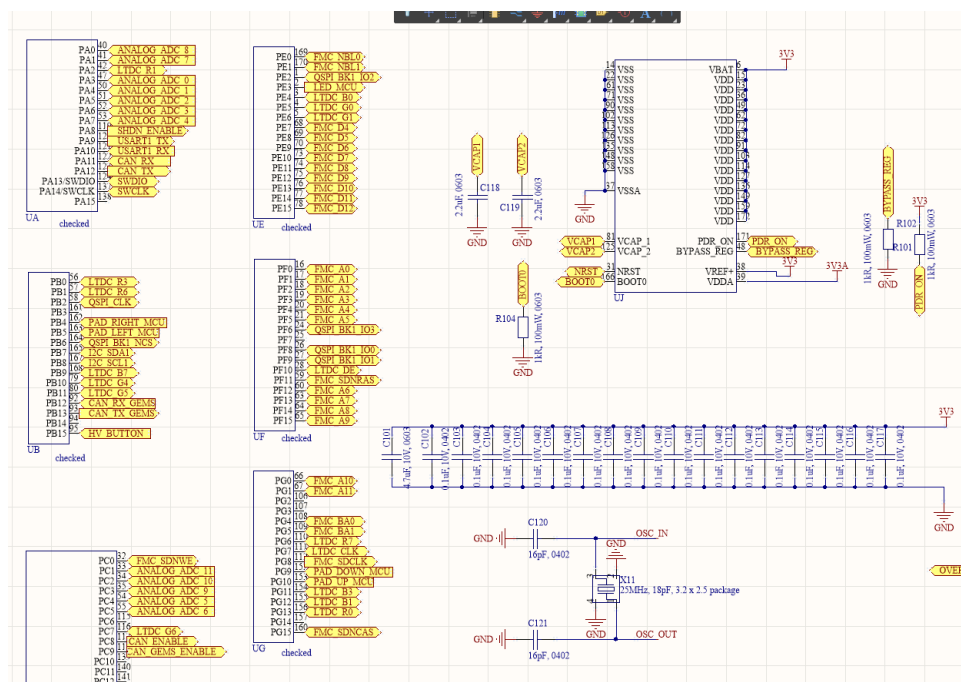
2.3 Mikrokontrolleriga trükkplaadi elektriskeemid

Elektriskeemide passiivsete komponentide valikul on arvestatud sellega, et need joodetakse käsitsi trükkplaadile ning tegemist on prototüüpimisega ja pideva arendamisega. Sellest tulenevalt on komponentide korpuste suurused varieeruvad ning ei pruugi olla optimaalse jalajälje suurusega.

Käesolevas alapeatükis olevad skeemid on kirjutamise hetkel kõige hilisemad ning esimese revisjoni testimise käigus ilmnunud vigade parandused on sisse viidud. Algsed vead ja nende leidmine testimise käigus on välja toodud kolmandas peatükis.

2.3.1 Mikrokontroller

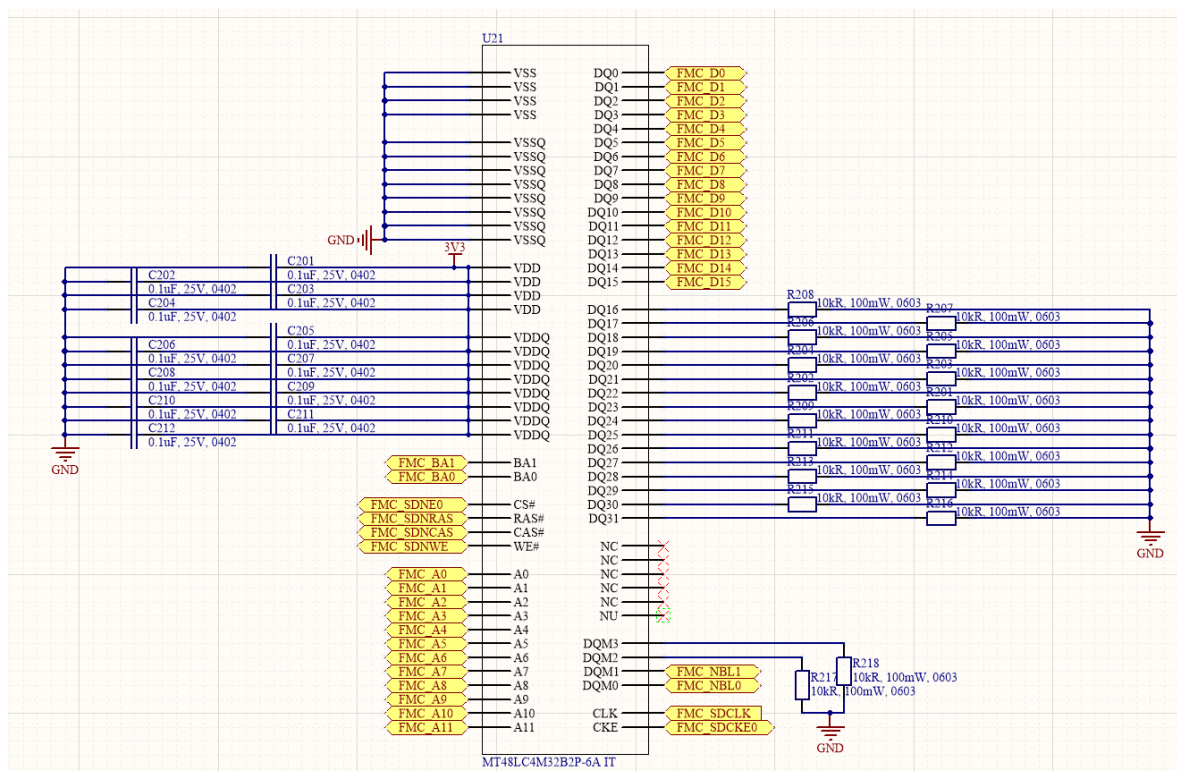
Nagu mainitud eelnevas alapeatükis, trükkplaadil kasutatav mikrokontroller on STMicroelectronics'i toodang STM32F746IGT6 [7], millel on 176 väljaviiku, millest 140 on kasulikud ja 120 neist on kasutuses (Lisa 5). Mikrokontrolleri toiteläbiviikude skeemide koostamisel on järgitud tootja andmelehel olevaid soovitusi. Töökindluse ja probleemide ennetamise eesmärgil on lisatud ka väline ostsillaator, kuna mikrokontrollerisse integreeritud ostsillaator väljub oma ettenähtud võnkumissageduse tolerantsi piirest, kui seda liialt kuumutada. See võib realselt ka juhtuda, kuna enamus trükkplaatide meeskonnas(s.h. käesoleva töö raames disainitud) joodetakse käsitsi kasutades jootekolbi või kuumaõhu jaama.



Joonis 2.5 Osa mikrokontrolleri skeemist

2.3.2 SDRAM

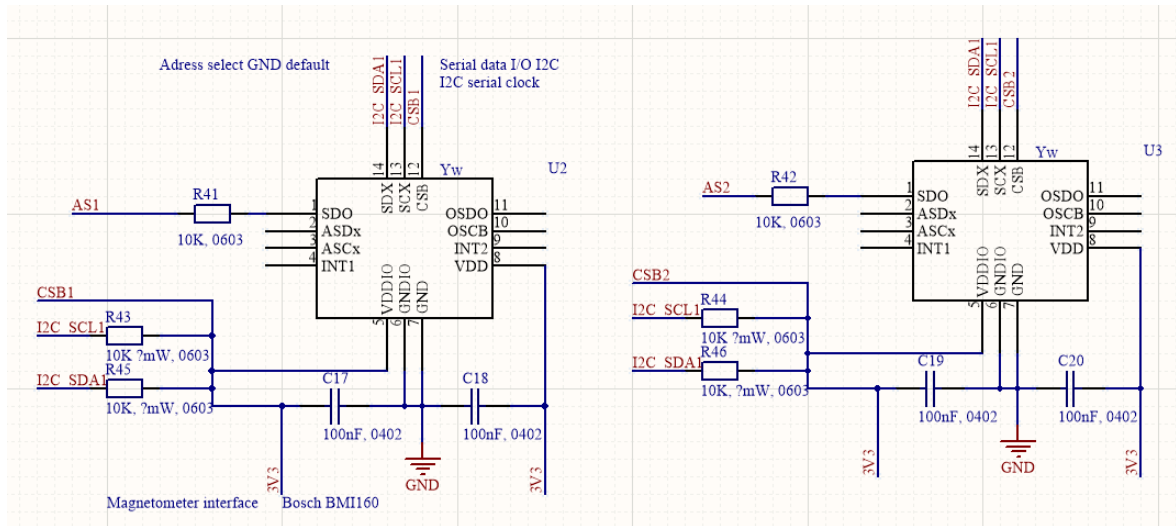
Ekraanil pildi kuvamiseks vajamineva välise suvapöördusmälu skeemid on võetud STM'i arendusplatvormi skeemide seast, nagu ka eelmisel hooajal. Probleemseks selle kiibi kasutamine ei osutunud, seega aja kokkuhoiu mõttes jätkati sama lahendusega. Nagu näha skeemilt (Joonis 2.7), siis on kasutuses ainult pooled andmesiini läbiviigud(DQ 0-15) - järgmine arendaja saaks optimeerida kiibi valikut ja sellega vähendada dünaamilise muutmälu jalajälje suurust trükkplaadil.



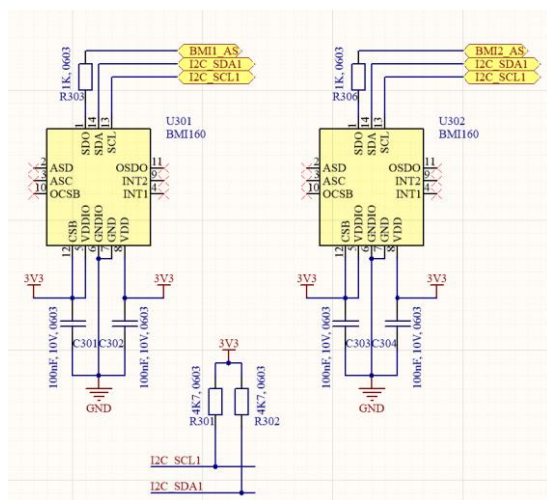
Joonis 2.7 SDRAM skeem

2.3.3 Kiirendusandurid

Kiirendusandurite skeemide tegemisel on tuginenud eelnevatel hooaegadel arendatud sensorika trükkplaadi skeemidele ja andurite andmelehele [8]. Skeemid on tehtud arusaadavamaks ning eemaldati üleliigsed *pull-up* takistid I2C suhtlusliinide(SDA, SCL) pealt - kummagi liini peale piisab ühest takistite komplektist.



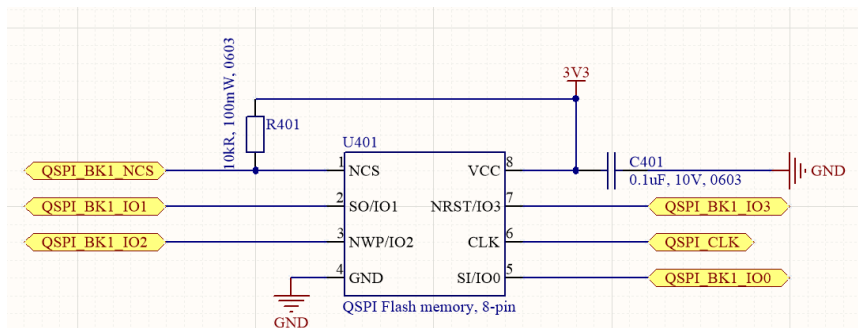
Joonis 2.8 FEST18 sensoriklaadi kiirendusandurite skeem [1]



Joonis 2.9 FEST19 kiirendusandurite skeem

2.3.4 Välmälu ehk *flash*

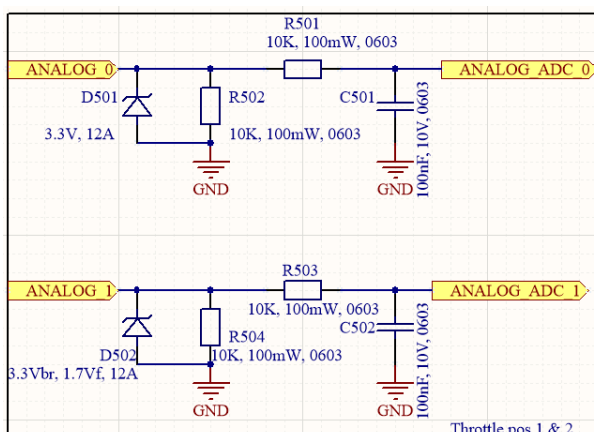
Välmälu skeemi on lisatud ainult 2 passiivset komponenti toote andmelehe¹ põhjal – *pull-up* takisti ja toidet stabiliseeriv kondensaator.



Joonis 2.10 Välmälu skeem

2.3.5 Analoogsisendid

Analoogsisendite skeemid on disainitud eelmise hooaja sensorika trükkplaadi arendaja soovitude ja võistlussarja reeglite [2] põhjal. Reeglite **punktis T 11.9.2** on nõutud juhtme purunemise või sensori rikke korral ohutu olukorra tagamine. Sellele nõudele vastamiseks on analoogskeemidele (näide Joonisel 2.11) lisatud *pull-down* takisti, mis tekitab mikrokontrolleri analoog-digitaal muundurisse null-pinge, mida sensorite normaalsel talitusel kunagi ei teki. Selle väärtuse põhjal ohutuse tagamiseks toiminguid teostab elektrooniline juhtüksus.

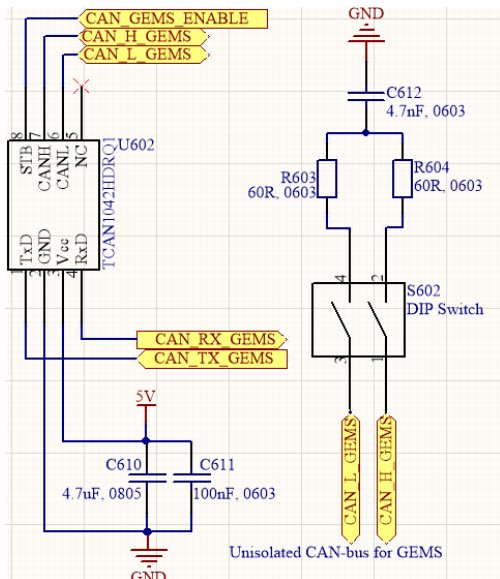


Joonis 2.11 Kiirenduspedaali sensorite analoogskeem

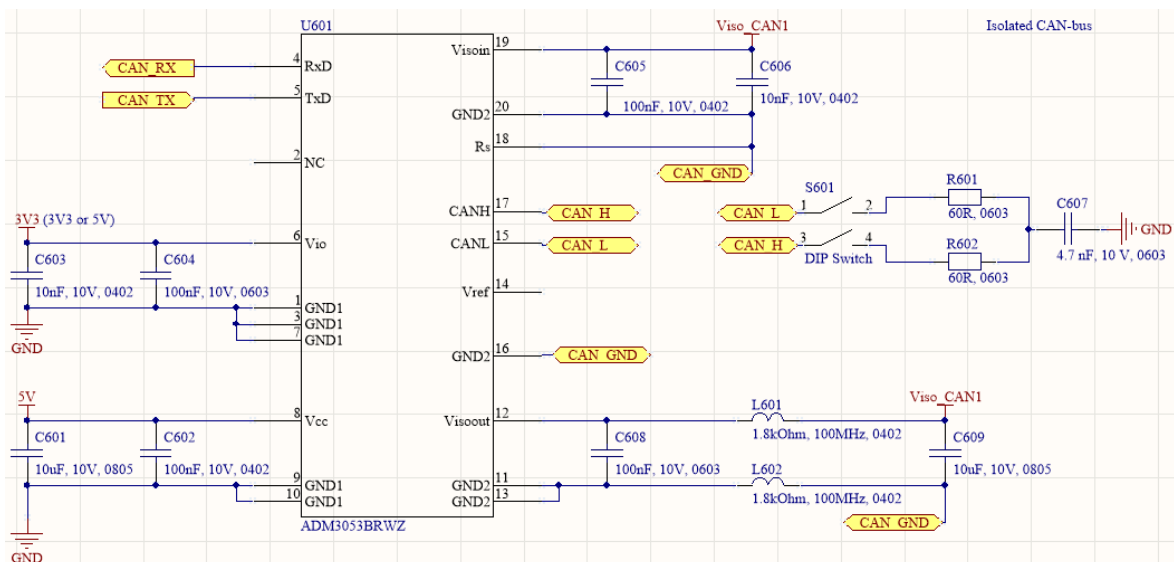
¹http://www.farnell.com/datasheets/2330438.pdf?_ga=2.180277028.1220324903.1556883029-2077225891.1538666943&_gac=1.219295339.1556883029.CjwKCAjwza_mBRBTEiwASDWVvgcp2eYr2s30lz7Jspi0lpENih2fJuNFWxU6bZuFikSGOZR2J9JPMxoCp3wQAvD_BwE

2.3.6 CAN suhtluse skeemid

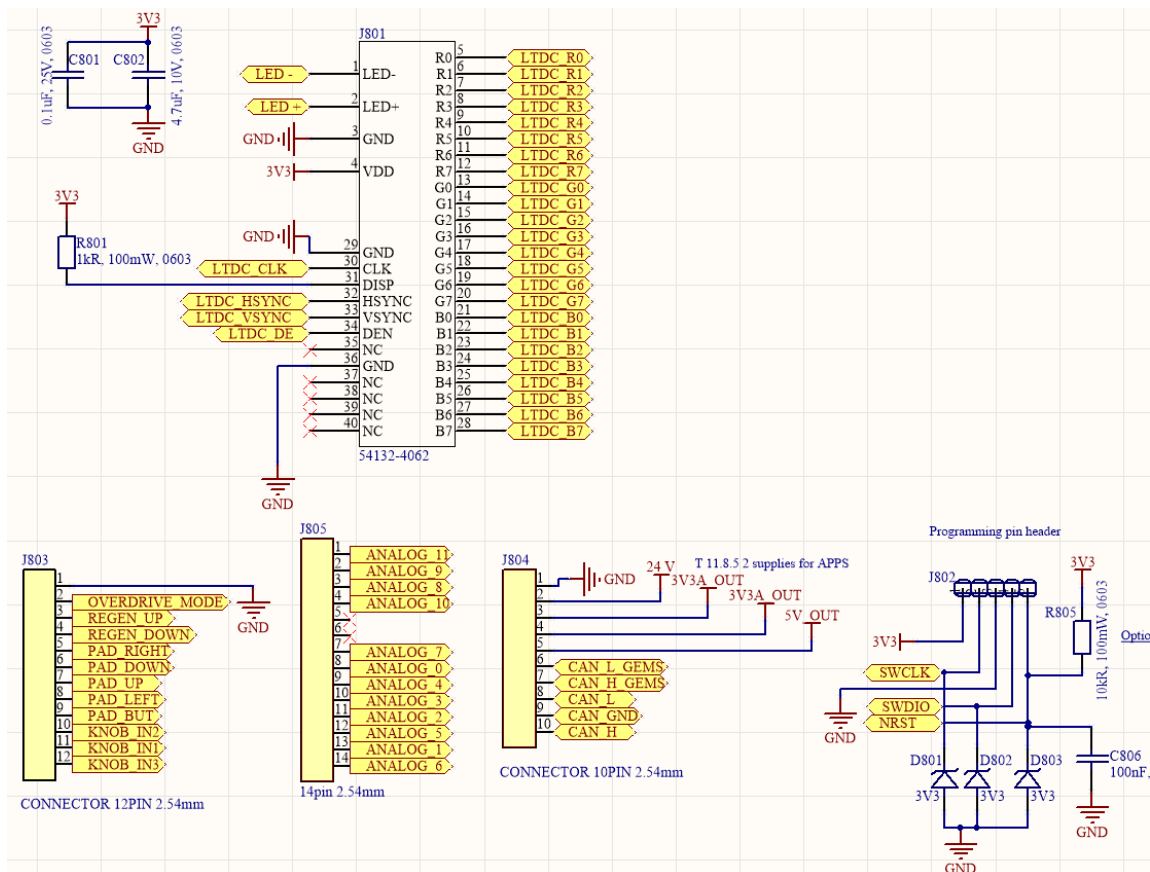
Isoleeritud CAN suhtluse skeem(vt Joonis 2.12b) on disainitud meeskonnaliikme poolt tootja andmelehe soovitude põhjal ning see on kasutusel kõikide CAN siinil olevate trükkplaatidel, et tagada skeemist tingitud erinevuste/probleemide tekkimist. Isoleerimata CAN skeem(vt Joonis 2.12a) on võetud eelmisel hooajal arendatud CAN siinil olevate trükkplaatide skeemidest.



Joonis 2.12a Isoleerimata CAN siini skeem



Joonis 2.12b isoleeritud CAN siini skeemid



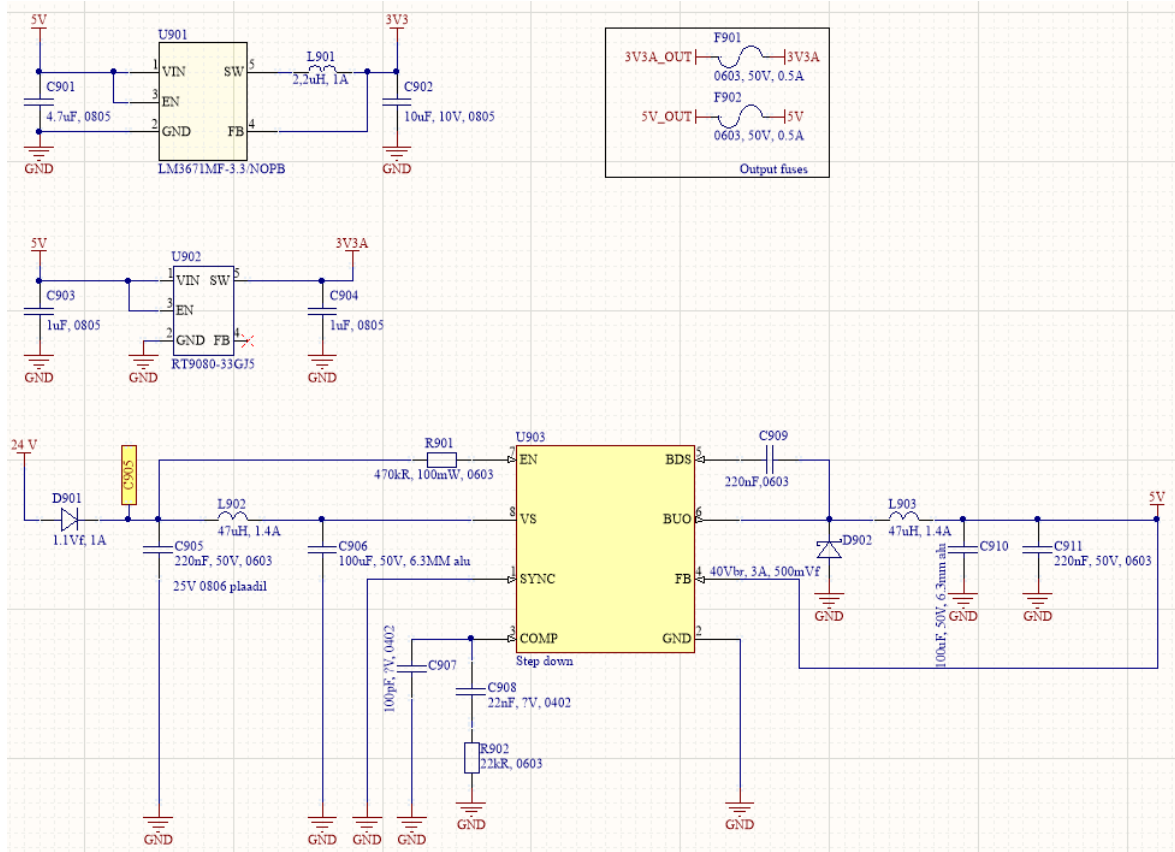
Joonis 2.14 Trükkplaadi pistikud

2.3.9 Toiteahelad

Trükkplaadil on kokku kolm toiteahelat: 24 → 5 V ja kahe erineva tehnoloogia baasil 5 → 3.3 V. Pingeregulaatorid, mis väljastavad pinget 3.3V on *step-down buck converter* (Joonis 2.15, „U901“) ja *LDO* (Joonis 2.15, „U902“). Otsus kasutada kahel erineval tehnoloogial töötavat kiipi tuleneb nende väljundpinge stabiilsusest – *LDO*, mille väljundpinge on stabiilsem efektiivsuse arvelt [14], kasutatakse ainult andurite toitenä, parandades sellega anduri lugemi stabiilsust. Teise regulaatori väljundit kasutatakse trükkplaadil olevate 3.3V pingde pealt töötavate komponentide toitenä.

Toiteahela pingeregulaatori skeemi, mille väljundiks on 5V (Joonis 2.15, „U903“), ning komponentide väärtuste arvutuse koostas sel hooajal üks meeskonna liikmetest. Skeem disainiti väljastama voolu kuni 2 A (suurim arvutuslik maksimaalne tarbimine autos olevate trükkplaatide seast), mis ka spetsiaalselt disainitud ja toodetud trükkplaadi peal valideeriti. Antud skeemi kasutatakse käesoleval hooajal disainitava vormeli FEST19 kõikide trükkplaatide peal, mis tarbivad 5 V pinget.

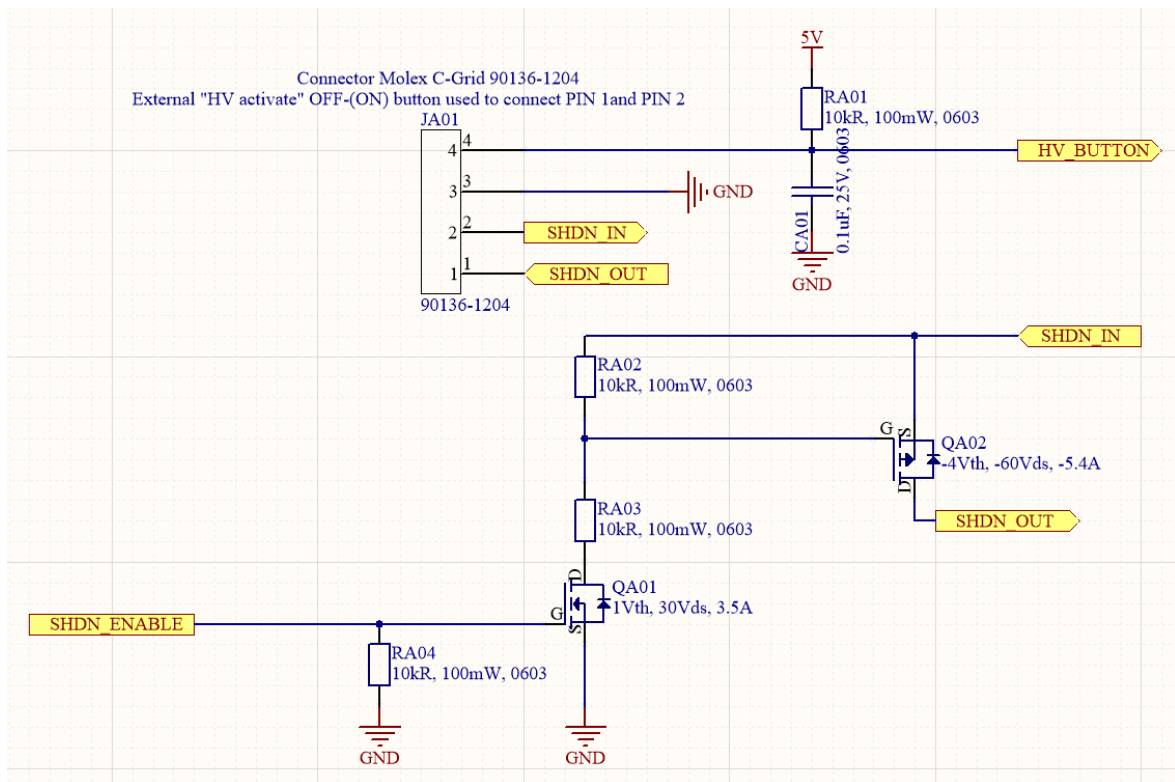
Nagu esimeses lõigus mainitud, siis anduritele toite andmine toimub läbi antud töö raames disainitava trükkplaadi. Pingeregulaatorite ning plaadi vaseradade kaitseks on andurite toiteliinidele lisatud 500 mA taluvad kiiretoimelised pindmontaaž kaitseelemendid, mida on vajadusel lihtsam vahetada, kui pingeregulaatoreid (Joonis 2.15, „Output fuses“).



Joonis 2.15 Toiteahelad ja kaitseelemendid

2.3.10 Ohutusahela e. *shutdown*-i lülitamine

Tervet autot läbiv ohutusahela *shutdown*-i üheks osaks on käesoleva töö raames disainitav moodul. Juhul kui madala ja kõrge pinge kraanid vormeli külje peal on keeratud suletud asendisse ning ülejäänud ahelasse kuuluvad komponendid on „heakskiidu“ andnud, jääb ainsaks kõrgepinge lülitamist takistavaks lülilik *dash*. Juhul kui vajutada vormeli armatuuril olevat kõige alumist nuppu, lülitab mikrokontroller N-tüüpi transistori QA01 juhtivaks, ning tekitab voolu liikumise ka RA02 ja RA03’st moodustuvas pingejaguris - P-tüüpi transistor QA02 hakkab juhtima (vt Joonis 2.16). Kõnealusel ohutusahelas liigub pidevalt keskmiselt 200 mA voolu, kuid lülitushetkel võib see mitmekordistuda – see oli skeemi üle dimensioneerimise põhjus. Tõenäoliselt saaks siiski skeemi optimeerida, kasutades liinil kaitselementi ja väiksema korpusega transistore.



Joonis 2.16 Shutdown lülitamise skeem

2.4 Mikrokontrolleriga trükkplaadi disainimine

Trükkplaadi disainimisel on üritatud järgida häid inseneritavasid, et vältida erinevate parasiitakistuste, -mahtuvuste ja läbikostvuse (ingl.k *crosstalk*) tekkimist, ära hoides sellega pealtnäha anomaalsete probleemide tekkimise. Vaatamata sellele saaks siiski trükkplaadi disaini märkimisväärselt optimeerida/parandada, sest ajapiirangu ning etteseadud PCB kihtide(4) tõttu komponentide paigutus tõenäoliselt ei ole parim. Lisaks veel on piiravaks faktoriks ekrani ribakaabli pikkus, ja moduli korpuse ettenähtud mõõtmete piiridesse jäämine – kõrgemad komponendid tuli tuua ribakaabli pistikust vastandpoolele.

Trükkplaadi mõõtmed on 106 x 67.5 mm, mis tulenes LCD ekrani maksimaalsetest mõõtmetest – see tagas korpuse võimalikult minimaalse disaini ning andis piisavalt ruumi kogu funktsionaalsuse mahutamiseks.

Põhilised punktid, mida disainimisel järgiti:

- Mürarikkad komponendid(näiteks toiteskeemid) paigutati eemale digitaalloomikale põhinevatest kiipidest
- Mürarikkad rajad(toide) hoida trükkplaadi äärtes, mitte tuua risti üle plaadi.
- Analoogsignaali rajad pistikust filterskeemideni ja neist mikrokontrollerini hoida võimalikult eraldatud teistest komponentidest.

2.4.1 Pistikute paigutus

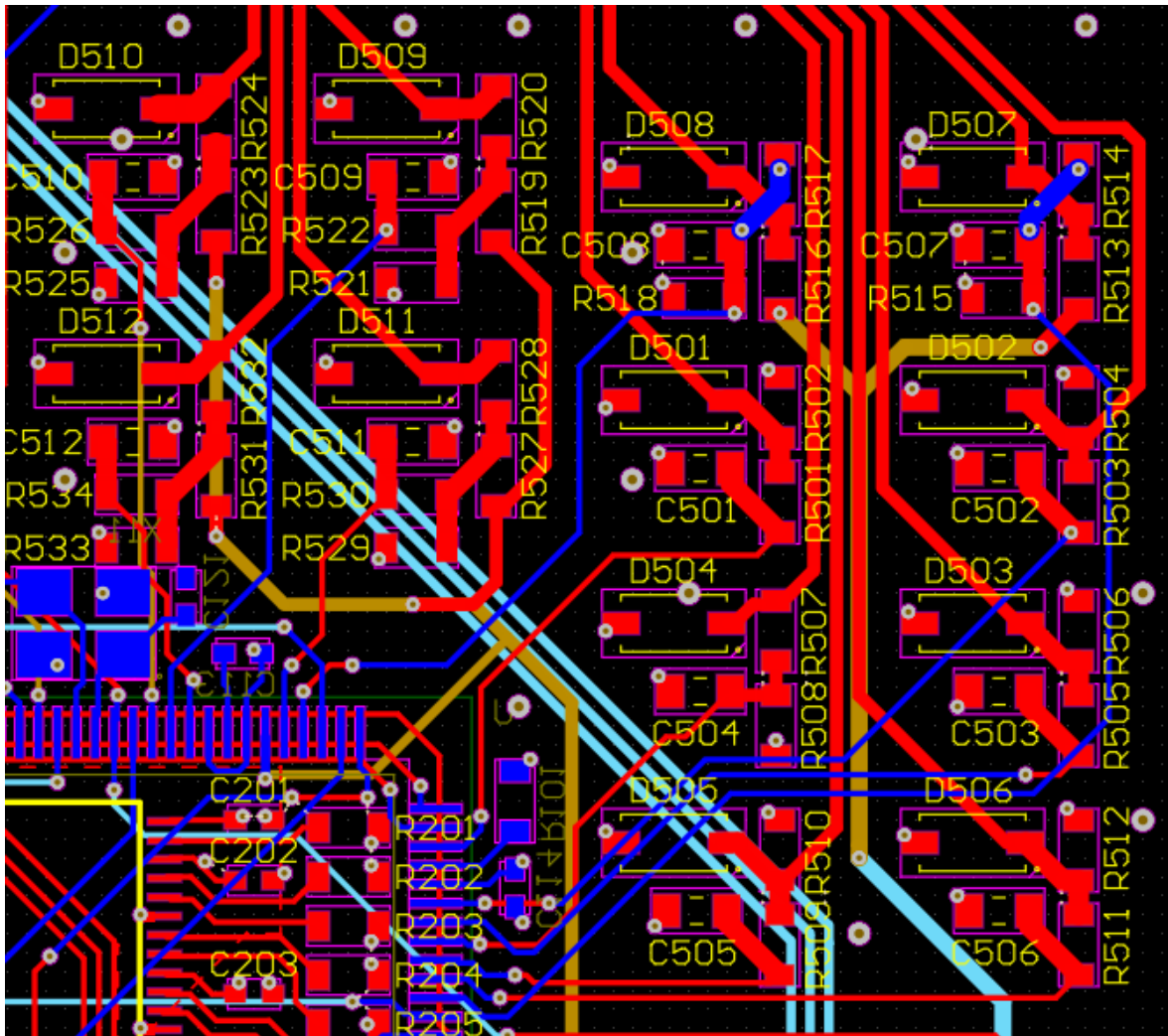
Vaadates trükkplaadi asetust autos, oli igati mõistlik paigutada kõik pidevalt küljes olevat pistikute pistikupesad trükkplaadi üles äärde, et piloot monokokki sisenedes (vt Lisa 6) ning sõidu ajal kuidagi neile pihta ei läheks ja sellega plaati/juhtmist kahjustaks. LCD ribakaabli pistiku asukohta valik tehti vastavalt ekraani tootja andmelehe tehnilisele joonisele.



Joonis 2.17 Trükkplaadi pistikute paigutus

2.4.2 Analoogskeemide paigutus

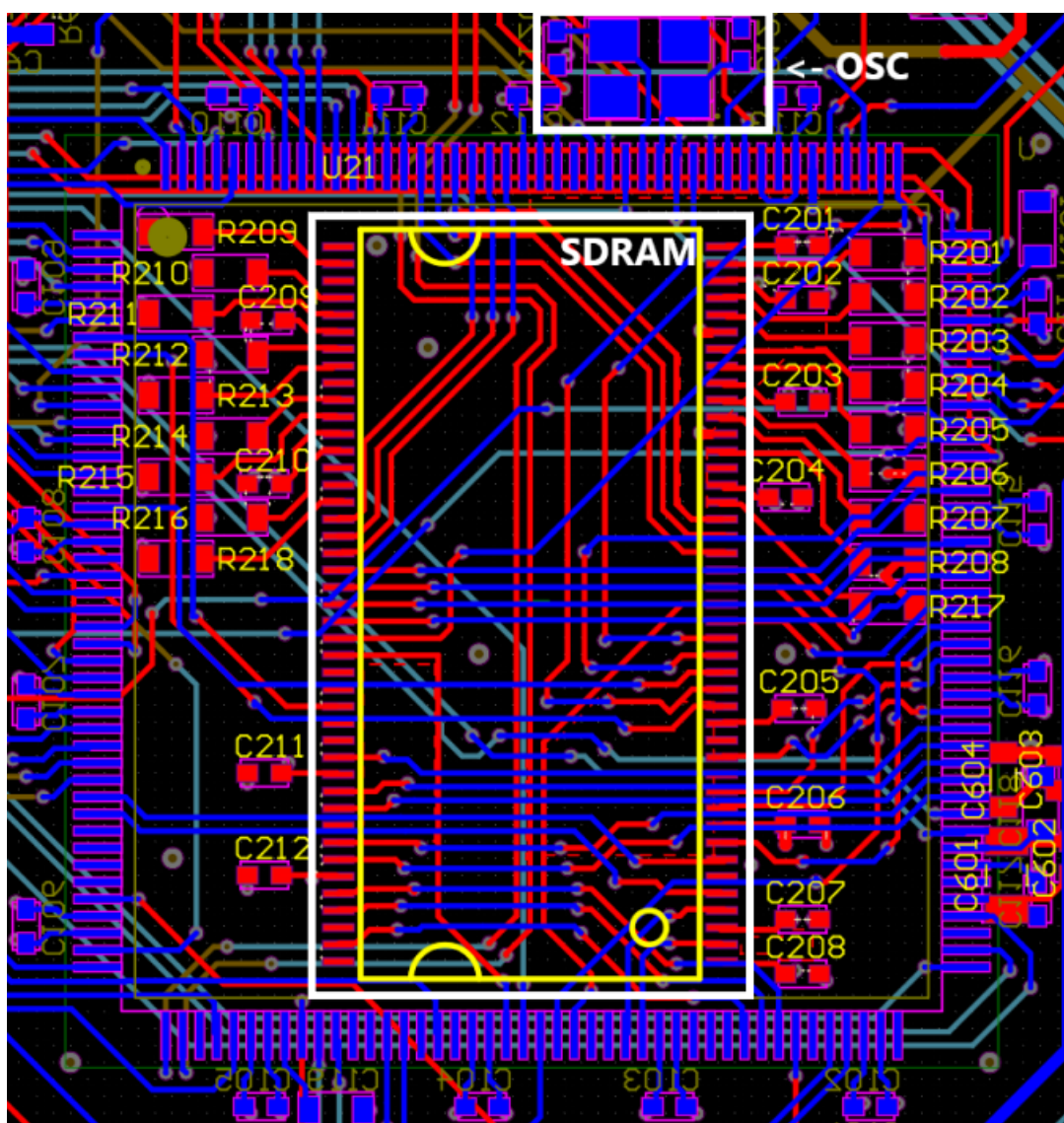
Analoogskeemide puhul oli eesmärgiks kompaktne paigutus, loogiline disain ning eraldatus teistest mürarikastest trükkplaadi komponentidest (eriti *switching regulator*’ile baseeruvatest toiteahelatest). Analoogsignaaside puhtuse hoidmiseks toimetati need mikrokontrollerini nii, et ükski muu toiterada või signaal ei oleks samal kihil läheduses.



Joonis 2.18 Analoogskeemid

2.4.4 Dünaamilise muutmälu ehk SDRAM-i ja mikrokontrolleri paigutus

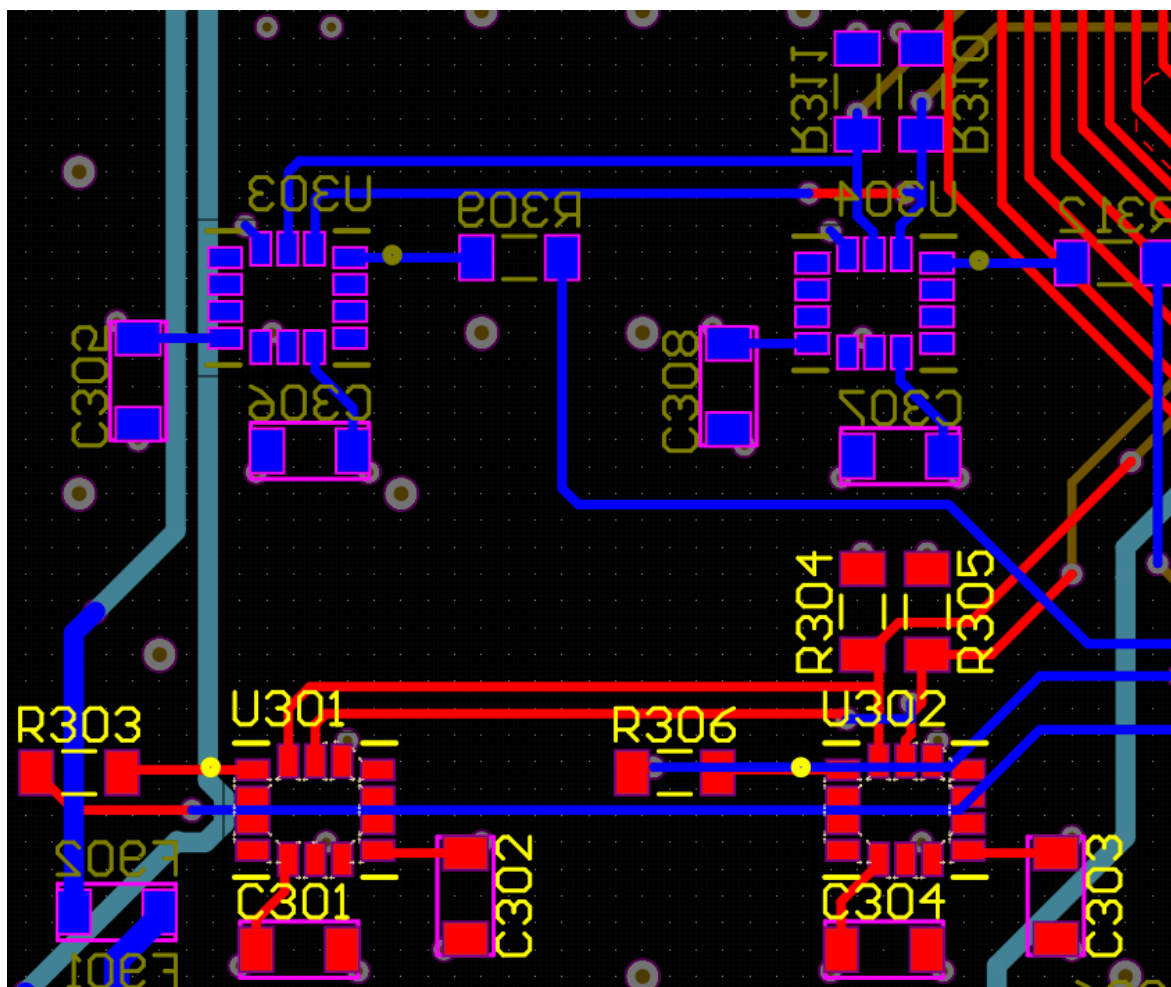
Mikrokontrolleri paigutuse puhul lähtuti järgnevast loogikast – võimalikult kaugelt mürast allikatest, kuid samas piisavalt plaadi keskel, et kasutuses olevatele läbiviikudele oleks võimalik mõistlikult ühendused teha. SDRAM'i paigutati täpselt mikrokontrollerist teisele poole trükkplaati (vt Joonis 2.20), mis tagab võimalikult lühikese andmevahetusradade pikkuse. Toidet siluvad kondensaatorid, ja *pull-up/pull-down* takistid paigutati kiipidele võimalikult lähedale, et neile passiivkomponentidele ettenähtud funktsionaalsus oleks parim. Ostsillaatori (Joonis 2.20, „OSC“) paigutusel arvestati kogenumate meeskonnaliikmete soovitusi – see peaks olema nii lähedal vastavatele läbiviikudele, kui võimalik ning ettenähtud koormuskondensaatorid peaksid olema ostillaatori kõrval või all (antud juhul kõrval).



Joonis 2.20 SDRAM (punased padjad) ja MCU paigutus (siniste patjade ruut)

2.4.5 Kiirendusandurite paigutus

Kiirendusanduri paigutusel lähtusin suuresti eelmisel hooajal häid tulemusi näidanud sensorika trükkplaadi kiirendusandurite paigutusest – kaks andurit ühele poole trükkplaati ja kaks teisele poole. Kõik sellega seotud passiivsed komponendid paigutati võimalikult lähedale, aga samas piisavalt kaugelt, et oleks võimalik käsitsi joota. Andurite ning nende passiivkomponentide paigutus on toodud joonisel 2.21.



Joonis 2.21 Kiirendusandurite paigutus

3. INFOVAHETUSE JA SENSOORIKA MOODULI TESTIMINE

3.1 Riistvara jootmine ja testimine

3.1.1 Ülevaade jootmis- ja testimistehnikast

Antud lõputöö raames disainitud trükkplaadid on joodetud autori poolt käsitsi, kasutades meeskonna temperatuuriregulaatoriga ja isoleeritud küttekehaga jootekolbi ning kuumaõhujaama. Enamus passiivkomponente on joodetud jootepasta ning kuuma õhuga, et vähendada jootmiseks kuluvat aega. Kõik digitaalloomika komponendid on joodetud kvaliteetse tina ja jootekolbiga, vältimaks sellega ülekuumenemisest tingitud probleeme ja kiipide funktsionaalsuse kaotamist.

Riistvara testimise esimene samm oli pingeregulaatorite skeemide kokku jootmine ning nende valideerimine. Esimesena joodeti 24 V -> 5 V muundamise toiteskeem ja testiti välise toiteploki, mille väljastuspingsiks seati 24 V ja vool piirati ohutuse eesmärgil 100 mA peale. Viimane väldib trükkplaadi kasutuskõlbmatuks muutumist lühise korral – voolu mitte piirates võib lühises olev vaskrada maha põleda toite lülitamise hetkel. Kõik töötas nii nagu pidi, trükkplaadil süttis 5 V ahelaga seotud indikatsioonituli ning voolu tarbiti ainult LEDi toiteks(vt joonis 3.1). Samasuguse loogika põhjal joodeti ja testiti ka mõlemad 3.3 V ahelad – kõik toimis nii nagu pidi(Joonisel 28 on võimalik näha nelja erineva ahela süttinud indikatsioonitulesid) ning võis jätkata järgmiste komponentide jootmise ja testimisega.

Kõik toiteahelad kontrolliti ka testeriga üle, mille tulemused on järgmisel leheküljel tabelis(Tabel 3.1). Ettenähtud pingest erinevus võib olla tingitud järgmistest võimalikest faktoritest – testeri ja/või pingeregulaatori ebatäpsus ja/või ideaalse kontakti puudumine(oksüdeerunud/puhastamata sond või mõõtepunkt).



Joonis 3.1 Süttinud toiteahelate indikatsioonituled

Ahel	Ettenähtud väljundpinge	Väljundpinge testeriga kontrollides
24 – 5 V	5 V	4.99 V
5 – 3.3 V (<i>switching reg.</i>)	3.3 V	3.28 V
5 – 3.3 V (<i>LDO</i>)	3.3 V	3.29 V

Tabel 3.1 Toiteahelate mõõdetud väljundpinged

Järgnevate sektorite jootmise järjekorral järgiti loogilist lähenemist – enne joodeti peale kõik võimalikud passiivkomponendid kuuma õhu ja jootepastaga ning siis alles liikuda kuumatundlike digitaallogika kiipide juurde, et vähendada kuumusest tingitud töökindluse vähenemist. Viimasena joodeti külge pistikupesad – nende korpused on plastikust ning hakkavad kiiresti deformeeruma kuuma käes.

3.2 Riistvara tarkvaraline testimine

Peale trükkplaadile enamus komponentide külge jootmist, võis alustada trükkplaadi valideerimist tarkvaraliselt – üks haaval kõikide kiipidega suhtlust kontrollida testimiseks ettenähtud lähtekoodiga.

3.2.1 Mikrokontrolleri programmeerimine

Esimesena oli vaja ligi pääseda mikrokontrollerile JTAG¹ programmeerimispistiku kaudu, kasutades STM32F4 Nucleo² platvormi, mida suurem osa meeskonnast kasutab STM'i mikrokontrollerite programmeerimiseks. See osutus algul probleemseks – tarkvaraarenduskeskkond „Atollic“ väitis, et programmaator ei ole ühendatud ühegi mikrokontrolleriga. Kontrollides mitmeid kordi üle, et ühendus on õige, ja pistikupesa on õiget pidi pistiku küljes, tuli kontrollida trükkplaadi skeeme ning leida skeemist viga. Probleem siiski esines trükkplaadi disainis - programmeerimispistiku signaalid „SWCLK“ ja „SWDIO“ olid vahetuses. Trükkplaati selle probleemi tõttu kõlbmatuks ei tunnustatud - tuli ümber tõsta pistikupesas kaks kontakti ja edaspidi mikrokontrolleri programmeerimine toimis.

3.2.2 CAN transiiverite testimine

CAN transiiverite testimiseks tuli esialgu leida eelmise aasta projektide tarkvara seast meeskonna arendatud CAN teek. Järgmisena tuli tekitada tarkvaras testsõnumid ning kontrollida CAN sõnumite liiklust spetsiaalse tööriistaga ja tarkvaraga – Kvaser Leaf Light v2³ ja Kvaser CanKing⁴. Esialgu töötas sõnumite liiklus ainult ühe-suunaliselt – sõnumid välja jõudsid, aga sisse ei loetud ühtegi sõnumit. Selle parandamiseks tuli alustuseks kindlaks teha, kas transiiverini jõuab CAN protokollile vastav signaal. Tõrkeotsingut hõlbustas meeskonnal olemasolev ostsilloskoop, mille sondid ühendati otse mikrokontrolleri väljaviikude külge. Instrument kinnitas fakti, et signaal jõuab mikrokontrollerini õigel kujul (vt Joonis 3.4) ning sellest võis järeldada, et probleem peab olema tarkvaraline. Võrreldes testkoodi eelmise aasta projektidega ja nõu pidades vanemate liikmetega leiti probleemile lahendus - puudu oli sisselugemise puhvri seadistamist välja kutsuv

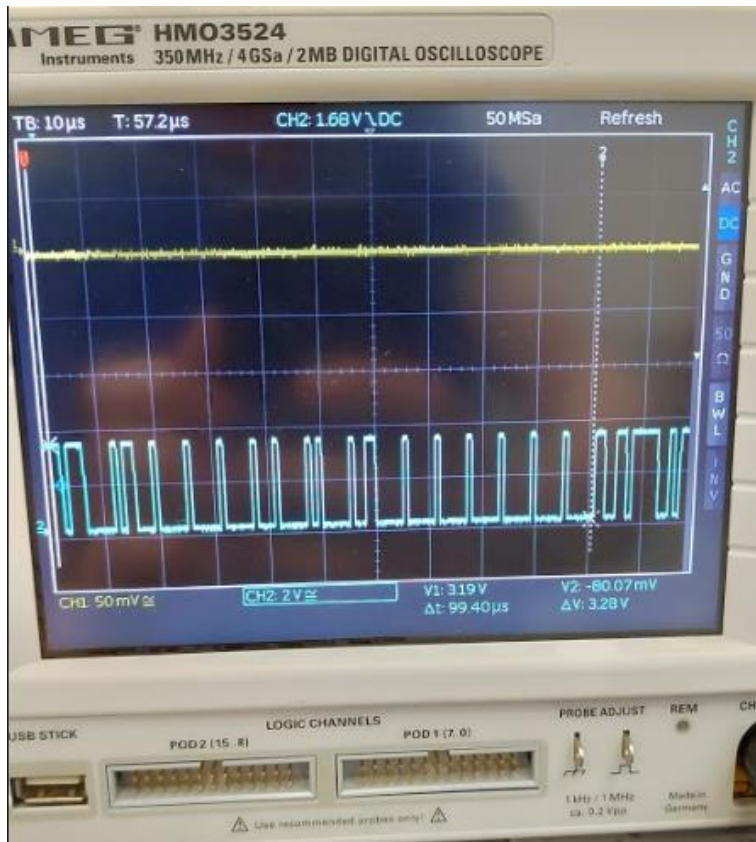
¹ <https://en.wikipedia.org/wiki/JTAG>

² <https://www.st.com/en/evaluation-tools/stm32-nucleo-boards.html>

³ <https://www.kvaser.com/product/kvaser-leaf-light-hs-v2/>

⁴ <https://www.kvaser.com/canking/>

rida lähtekoodis. Selle rea lisamisel toimis CAN suhtlus ootuspäraselt - CAN suhtluse võis lugeda valideerituks.



Joonis 3.2 CAN sõnumi osa ostsilloskoobil

3.2.3 SDRAM valideerimine

Välise muutmälu kiibi seadistamine põhines suuresti STM32CubeMX-ile, mis parameetrite ette seadmisel genereeris vajamineva lähtekoodi. Mäluaadress, mis mikrokontrollerile teada annab, et tegu on välise muutmäluga, oli kirjas MCU andmelehel. Laadides peale testimise eesmärgil loodud lähtekoodi, milles kasutati CubeMX-iga kaasa tulevaid näidis lähtekoode, pidi ekraanile ilmuma sõna „nord“, kuid nagu näha Lisa 7.1 joonisel, ei toimunud see nii, nagu ette nähtud – horisontaalsete ja vertikaalsete tulpade pikslid olid nihkes. Kogenenud tiimikaaslase soovitusel joodeti trükkplaadil kahtlasena näivad väljaviigud üle ning anomaalne roheline taust kadus ja horisontaalsihis olevad pikslid kuvati nii nagu on õige (vt Lisa 7.2). Siiski tekkis probleeme kasutajaliidese teegi esmasel lähtekoodi integreerimisel – värvid olid ebaloomulikud ning näha oli jätkuvalt ebanormaalsuseid. Tõrkeotsinguks kasutati ekraanil gradiendi joonistamise võtet, nii sisemise (mikrokontrolleri) kui ka välise muutmälu mäluaadressi kasutades, millest selgus, et

siiski on probleemseks komponendiks *SDRAM*-i kiip(vt Lisa 7.3 - 7.6). See probleem lahenes kõikide asjakohaste(nii *SDRAM* kui ka *MCU*) väljaviikude ülejootmisel.

3.2.4 Välmälu e. *flash* valideerimine

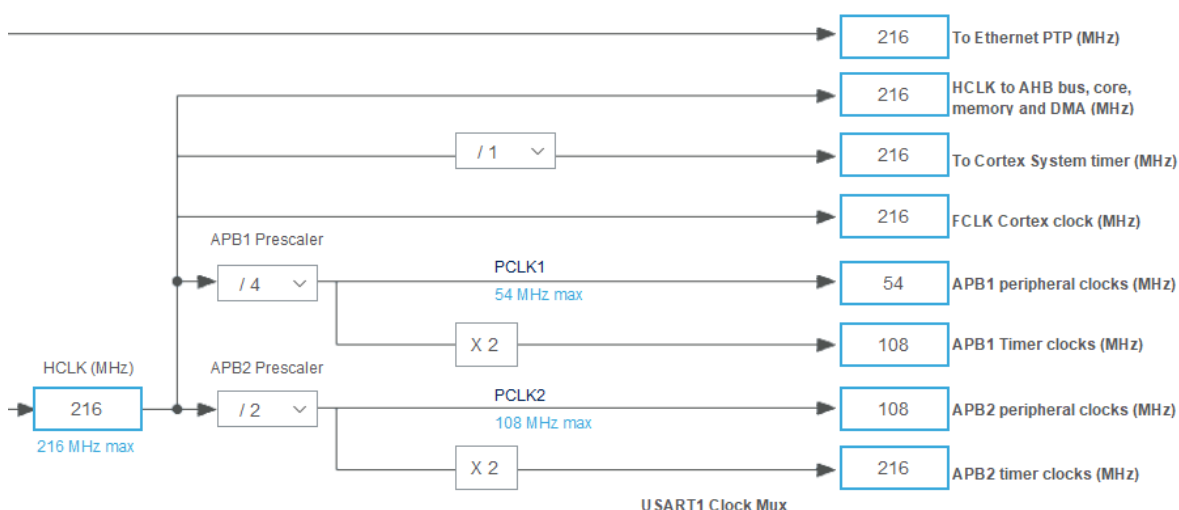
Antud töö kirjutamise hetkel ei ole välmälu kiipi õnnestunud tööle saada, kuid õnneks ei oma see kriitilist tähtsust vormeli sõitmisel, vaid on pigem boonuseks. Eesmärk on siiski plaadi peal oleva komponendi funktsionaalsus rakendusse võtta, kuid kuna see probleem on teisene, võetakse see käsile niipea kui pakilisemad probleemid lahendatud saavad.

4. PROGRAMMEERIMINE

Käesolevas peatükis antakse ülevaade programmeerimiskäigust, valminud programmist ning kasutajaliidese arengust ja tulevikuplaanidest. Kogu kood on uuesti algusest üles ehitatud, kasutades väga väheseid koodilõike eelmisest aastast. Kasutatud on kolme meeskonna poolt arendatud teeki – CAN, kiirendusanduritega suhtlemine ja infotöötlus, sündmuse taimer. Programmeerimiskeskonnaks on „Atollic TruStudio“, mikrokontrolleri ja tema perifeeriade algseadistus teostati „STM32CubeMx“ abiga ning mikrokontrolleri programmeerimine toimus läbi „STM32 Nucleo“ programmeatori, mis võimaldab piiratud arvu *breakpoint*ide lisamist koodi silumisel. Lähtekoodi üldistatud kuju kirjeldus on toodud alloleval plokkiagrammil(vt Lisa 10).

4.1 Mikrokontrolleri perifeeriade seadistamine

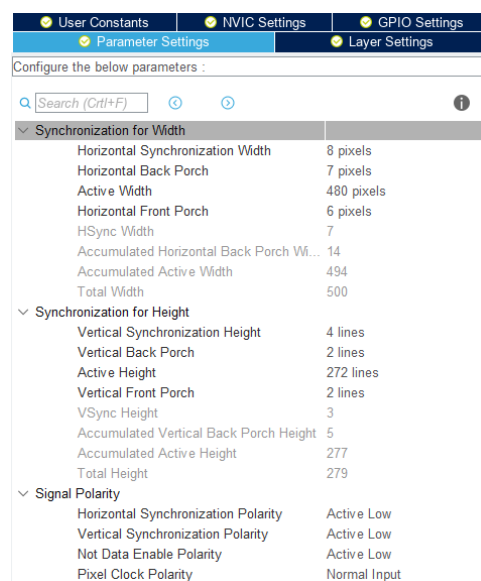
Nagu käesoleva peatüki alguses mainitud, mikrokontrolleri ja tema perifeeriade seadistamisel kasutati „STM32CubeMX“-i. See programm omab suhteliselt intuitiivset kasutajaliidest, mille kaudu on võrdlemisi lihtne sisse kanda väliste kiipidega suhtluse parameetreid ning seada erinevaid taktsagedusi(vt Joonis 4.2) jpm. Seadistamine on pigem otsekohene ning üldiselt lihtsalt andmelehe järgi seadistuse tegemine, siis põhjalikumalt on kirjeldatud ainult kahe perifeeria seadistusi – LTDC ja ADC.



Joonis 4.1 Taktsageduste seadistamine

4.1.1 LTDC seadistamine

LTDC ehk *LCD-TFT display controller* on LCD displei kontrollerr, mis antud juhul on STM poolt integreeritud mikrokontrollerisse (võimalik ka välise kiibina kasutada). Nagu nimest oletada võib, on see vajalik antud projekti puhul LCD ekraanil pildi kuvamiseks. Selle seadistamiseks parameetrid leiab ekraani tootja andmelehest [15].



Configure the below parameters :		
Search (Ctrl+F)		
Synchronization for Width		
Horizontal Synchronization Width		8 pixels
Horizontal Back Porch		7 pixels
Active Width		480 pixels
Horizontal Front Porch		6 pixels
HSync Width		7
Accumulated Horizontal Back Porch Wi...		14
Accumulated Active Width		494
Total Width		500
Synchronization for Height		
Vertical Synchronization Height		4 lines
Vertical Back Porch		2 lines
Active Height		272 lines
Vertical Front Porch		2 lines
VSynch Height		3
Accumulated Vertical Back Porch Height		5
Accumulated Active Height		277
Total Height		279
Signal Polarity		
Horizontal Synchronization Polarity		Active Low
Vertical Synchronization Polarity		Active Low
Not Data Enable Polarity		Active Low
Pixel Clock Polarity		Normal Input

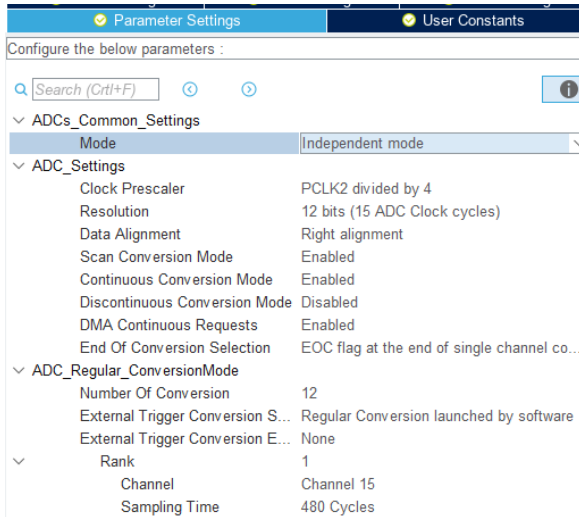
Joonis 4.2 LTDC seadistamine

4.1.2 ADC seadistamine

Andurite väärtuste lugemisel kasutatakse ADC1-te ning seadistamisel võeti kasutusse ainult vajalik arv kanaleid, milleks on 12, kuhu kuuluvad ka neli reservi. Lisaks sellele seadistati ka kõik kanalid maksimaalse proovivõtuaja peale (vt Joonis 4.4) – 480+12 tsüklit. See tagab lugemi võtmise hetkel pinge stabiliseerumise konverteril ning annab täpsema tulemuse kui kasutada näiteks 3+12 tsüklit¹. ADC väärtuste saatmine konverterist toimub üle DMA (*Direct Memory Access*), mis võimaldab väärtuste kirjutamist mikrokontrolleri põhimällu CPU-d kasutamata² – koormab vähem kriitlist ressursi ning tagab kiirema infoliikumise. ADC taktsageduseks on $PCLK2/4$, ehk $108\text{MHz}/4 = 27\text{ MHz}$, resolutsiooniks 12bitti (väärtus vahemikus 0-4095). Resolutsioonist on võimalik 3V3 pinge korral välja arvutada, et üheks eristatavaks ühikuks (*LSB*) kujuneb $3.3/4095 = 0.805\text{ mV}$, mis on piisav oma rakenduseks – konverteri väärtus kõigub vormelis 5 *LSB* vahel, ehk 4mV ulatuses.

¹ <https://electronics.stackexchange.com/questions/26329/what-is-the-purpose-of-adc-sampling-time>

² <https://www.techopedia.com/definition/2767/direct-memory-access-dma>



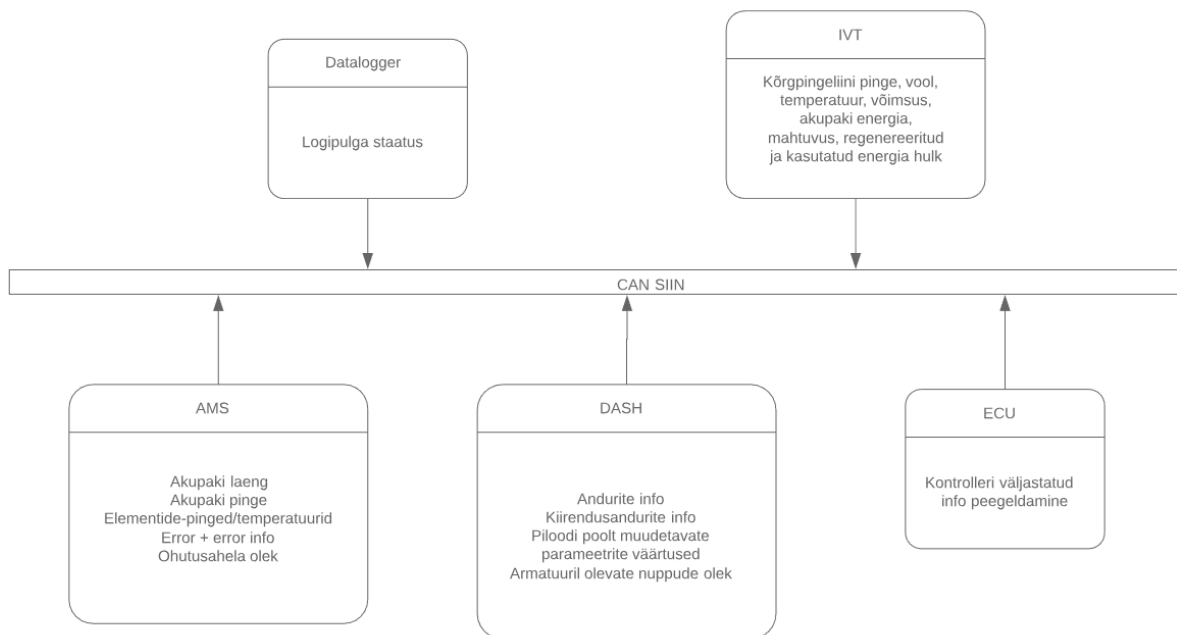
Joonis 4.3 ADC seadistamine

4.2 CAN suhtluse programmeerimine

CAN siini kaudu loeb moodulilt saadetud infot sensorite ja erinevate parameetrite kohta elektrooniline juhtüksus, ehk *ECU*. Mooduli mikrokontroller loeb CAN siinilt (transiiveri vahendusel) järgmiste trükkplaatide informatsiooni:

- AMS – akuhaldussüsteemi meistri poolt väljastatud elementide kriitilised temperatuurid, akupaki laetus, pinge jpm.
- ECU – elektrooniline juhtüksus on ainus trükkplaat, mis suhtleb otseselt mootorite kontrolleritega – peegeldab kontrollerite seisundite kohta infot.
- IVT – Erinevate elektriliste parameetrite seiret teostav ostutoode – täpsemad pingemõõtmised, energia kasutamise ja regenereerimise hulk sõidu jooksul jpm.
- *Datalogger* – Annab teada logimispulga olekust – kas viimane on ühendatud või mitte.

Suhtlusel on kasutatud tiimi poolt arendatud CAN teeki, mis lihtsustas programmeerimist ning oli vaja lihtsalt meeskonna sõnumite databaasi alusel välja filtreerida vajalikud sõnumid ja nendes sisalduv info ära kasutada seal kus vaja – näiteks kuvada kasutajaliidesele. Järgmisel leheküljel oleval joonisel (Joonis 4.5) on toodud kõik üldisel CAN siinil olevad moodulid koos nende poolt väljastatava infoga, mida arendatav moodul („*dash*“) loeb ja töötleb kuvamiseks.



Joonis 4.4 Lihtsustatud CAN struktuur ja komponentide saadetavad sõnumid

4.3 Kasutajaliidese programmeerimine

Kasutajaliidese programmeerimine oli kõige aeganõudvam osa lõputöö juures – puudus varasemalt spetsiaalselt kasutajaliidese disainimiseks mõeldud tarkvara kasutamine ning see tuli töö autoril iseseisvalt selgeks saada. LittlevGL [3] teegi eeliseks on suhteliselt lihtne integreeritavus ning seadistamine ei nõua suuremat varasemat programmeerimiskogemust. Vaatamata sellele, et algajale programmeerijale ei ole dokumentatsioon piisavalt põhjalik, on teegi autor ja nüüdseks lisandunud arendajad ning vanemad kasutajad foorumis aktiivsed, kes aitavad meeeldi probleemidele lahendusi leida ja aidata mõista teegi kasutamise põhifunktsionaalsust.

Kasutajaliidese arendamisel kaasati meeskonna liikmete (eelkõike pilootide) poolt tulnud soovitusi ning arvamusi, mis mingil määral lihtsustasid kogu disaini välja mõtlemist. Hetkeseisul koosneb *GUI* kuuest leheküljest:

- „Home“ – pealehekülg, kus on olemas kõik hädavajalik informatsioon, mida sõidu ajal jälgida
- „Conf“ – erinevate seadistuste muutmiseks (väärtused saadetakse ECUle) – nt veepumba kiirus, stabiilsuskontrolli versioon, akukasti jahutusventilaatorite kiirus jpm.
- „Sens“ – andurite kalibreeritud ja kalibreerimata väärtuste kuvamine

- „IVT“ – mitmesuguste elektriliste väärtuste kuvamine ostutootelt „Isabellenhütte IVT“
- „AMS“ – akuhaldussüsteemi tähtsamad parameetrid, veakoodi kuvamine, vea kirjeldus jne.
- „AMK“ – mootorite kontrollerite poolt saadetud informatsiooni kuvamine

Pilte reaalsest kasutajaliidese arengujärgust on võimalik näha **lisas 8**.

Kasutajaliidesele navigeerimiseks kasutatakse navigatsioonipatja, mille valik **peatükis 2.1.1** põhjendati. Horisontaalteljes olevaid nuppe, ehk vasak-parem, kasutatakse lehekülgede vahetamiseks. Vertikaalteljes olevad, ehk üles-alla nupud on vajalikud ekraanil olevate objektide vahel sirvimiseks ning nende väärtuste muutmiseks, kui keskmise nupuga sisenetakse redigeerimisrežiimi. Redigeerida saab kahte kategooriasse kuuluvaid parameetreid – üldised ja juhtimissüsteemile suunatud. Üldiste parameetrite all on näiteks jahutuse veepumba kiirus, akukasti jahutusventilaatorite kiirus jne. Juhtimissüsteemile suunatud parameetrid on näiteks kiirenduse režiimi aktiveerimine, veojõukontrolli ja stabiilsuskontrolli versiooni muutmise jne.

Kasutajaliidesele on plaanis veel funktsioone lisada, millest üks tähtsaim on sõidurežiimi ekraan, kus on juhile nähtav ainult hädavajalik informatsioon kriitilisematest parameetritest – nt. ringi- ja sektoriaega, aku temperatuuri jms. Selle arendus ilmselt valmib käesoleva aasta mai kuu lõpu poole, ning lõputöös selle arendus ei kajastu. Küll aga on juba aprilli alguse seisuga saavutatud sarnane funktsionaalsus, mis oli eelmise hooaja infovahetuse moodulil eelmise hooaja lõpuks.

KOKKUVÕTE

Käesoleva bakalaureusetööga integreeriti eelnevatel hooaegadel(FEST17/18) kaks eraldiseisvat trükkplaati ühele neljakihilisele trükkplaadile tudengivormelimeeskonna FS Team Tallinn poolt arendatud prototüübi FEST19 tarbeks. Disainimise käigus tugineti eelmise hooaja disainidele, kuid need lahendused kontrolliti(ja vajadusel muudeti) täiendavalt lisaarvutustega ning sama komponendi kasutamise puhul põhjendati selle vajalikkus. Valminud moodulil on võimekus suhelda transiiveri abil teiste mikrokontrollerile baseeruvate trükkplaatidega läbi autosiseses CAN siini, lugeda ning töötleda analoog- ja digitaalsignaale ning kuvada informatsiooni 4.3" TFT-LCD ekraanil.

Töö raames valminud mooduli eelviimase revisjoniga ja tema korpusega loodi vormelile FEST19 uuenduslik armatuurilaua disain, mis täidab kolme põhilist eesmärki. Esiteks, kuvab piloodile tähtsamaid andmeid autosiseselt suhtlusliinilt, teiseks võimaldab kasutajal intuiivselt muuta erinevaid parameetreid alustades veepumba kiirusest kuni veojõukontrolli versioonini ning kolmandaks loeb, vajadusel kalibreerib ja väljastab erinevate auto eesosas olevate sensorite analoogsignaali ja nuppude olekut digitaalkujul. Sellise kasutajaliidese disainimine, nagu antud töö käigus valmis, on oluline tudengivormeli meeskonna FS Team Tallinn arenguks ning konkureerimiseks tipp tehnikaülikoolidega tootearendusvõistlusel „*Formula Student*“. Lisaks sellele on mooduli arendamisprotsessi keerulisemad/tähtsamad punktid ja probleemid dokumenteeritud meeskonna „Confluence“ leheküljel, mis võimaldab järgmisel arendajal tugineda autori kogemustele. See omab väga suurt väärtust just meeskonnale, kuna järgneval hooajal saab jätkata ja optimeerida juba tehtud tööd, mitte ei tule alustada järjekordse mooduli disainimisega.

Mooduli viimase revisjoni muudatused ja kasutajaliidese lõplik disain antud lõputöö raamesse ei mahtunud, kuna meeskonnal on soov moodulile veel funktsionaalsust lisada, mis on aega nõudev.

SUMMARY

With current Bachelor's thesis, two, formerly (FEST17/18 seasons) separate printed circuit-boards were integrated into one four-layered board for the Formula Student team FS Team Tallinn developed prototype FEST19. The designing relied on last season's design, but the solution was revised (and changed where necessary) with additional calculations and the usage of identical components was justified. The achieved module has the capability to communicate with other microcontroller-based PCBs via in-car CAN bus, read and process analog – and digital signals and display information via a 4.3" TFT-LCD screen.

With the thesis, the second revision of this particular module and its housing design was completed and an innovative dashboard was created, which serves three main purposes. Firstly, the module displays the most important data to the pilot from the communication bus, secondly, it allows the user to intuitively edit different parameters, starting from the water pump speed all the way up to traction control version. Thirdly, it reads (also calibrates when needed) and transmits different sensors' (that are situated in front of the car) analog values and button states as a digital signal. The development of an user-interface like was designed within this thesis is crucial for the Formula Student team FS Team Tallinn development as well as competing against other top technical universities in a product development competition „Formula Student“. What is more, the most complex/important parts of the module's development process was documented to the team's „Confluence“ page, which allows for the next developer to rely on the author's experiences. This has a significant meaning to the team, as it gives the opportunity to continue and optimize already done work, which revokes the need to design another module from the start.

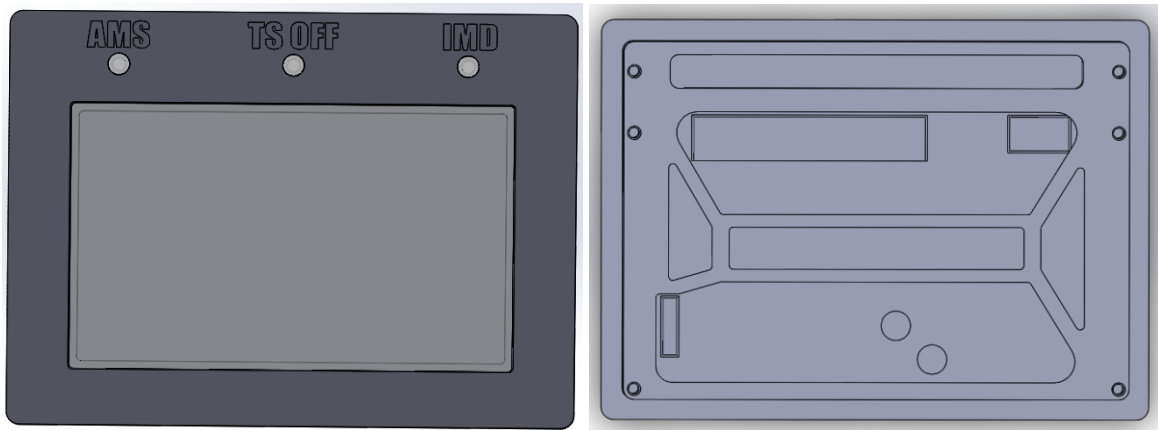
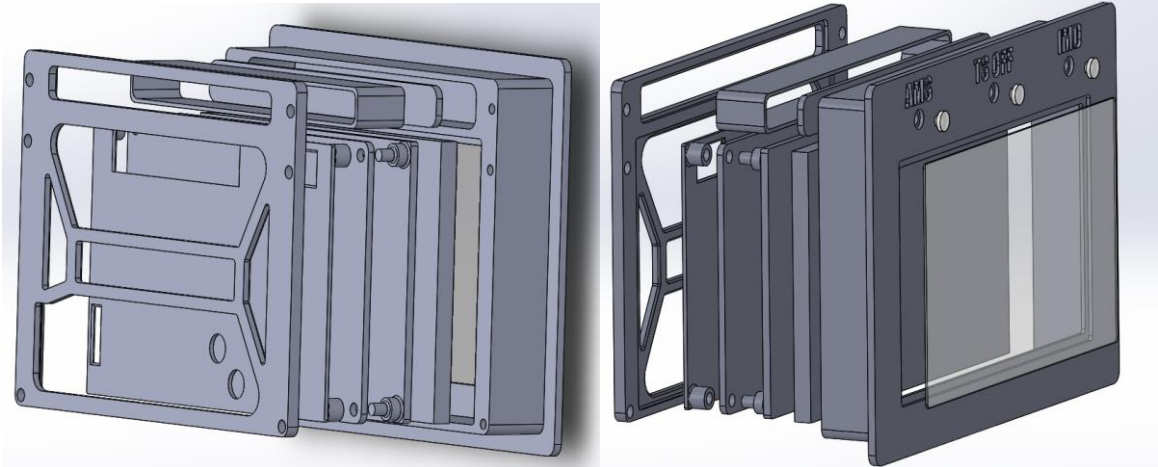
The module's final revision's changes and the user interface's final design is not a part of this thesis, since the team still plans to add functionality to the module, which is time consuming.

Viidete loetelu

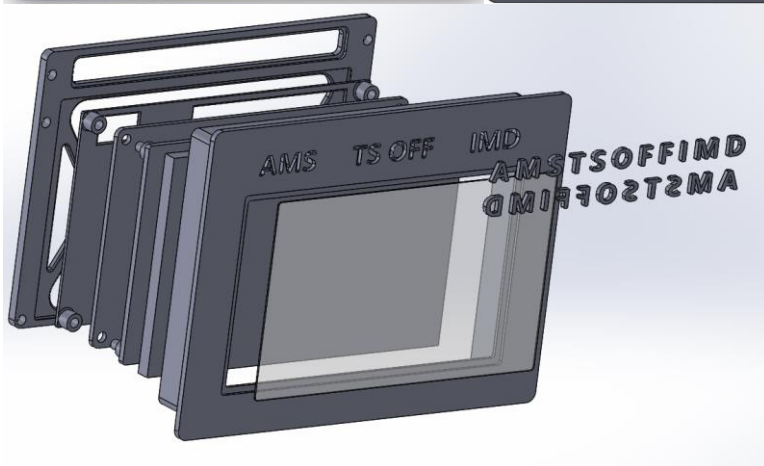
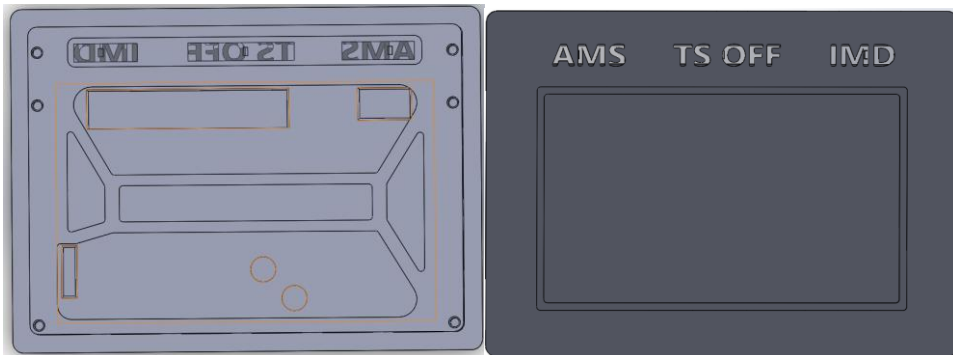
- [1] S. Vesi, „SENSOORIKAPLAAT TUDENGIVORMELILE FEST18, magistritöö,“ TTÜ, 2018. [Võrgumaterjal]. Available: <https://digi.lib.ttu.ee/i/file.php?DLID=10992&t=1>. [Kasutatud 30 April 2019].
- [2] F. S. Germany. [Võrgumaterjal]. Available: fsg.one/rules. [Kasutatud 1 May 2019].
- [3] „Littlevgl.com,“ [Võrgumaterjal]. Available: <https://littlevgl.com/>. [Kasutatud 29 April 2019].
- [4] „Altium,“ [Võrgumaterjal]. Available: <https://www.altium.com/>. [Kasutatud 30 April 2019].
- [5] STMicroelectronics, „STM32CubeMX dev. tool,“ [Võrgumaterjal]. Available: <https://www.st.com/en/development-tools/stm32cubemx.html>. [Kasutatud 30 April 2019].
- [6] „Atollic TruStudio,“ [Võrgumaterjal]. Available: <https://atollic.com/truStudio/>. [Kasutatud 30 April 2019].
- [7] STM, „Farnell,“ STM32F746IGT6, [Võrgumaterjal]. Available: http://www.farnell.com/datasheets/2309059.pdf?_ga=2.80096151.1220324903.1556883029-2077225891.1538666943&_gac=1.252964603.1556883029.CjwKCAjwza_mBRBTEiwASDWVvgcp2eYr2s30lz7Jspi0IpENih2fJuNFWxU6bZuFikSGOZR2J9JPMxoCp3wQAvD_BwE. [Kasutatud 3 May 2019].
- [8] Mouser. [Võrgumaterjal]. Available: <https://www.mouser.ee/ProductDetail/Bosch-Sensortec/BMI160?qs=sGAepiMZZMsF1ODjcwEocFh%252BXHDbm9xW9zpNSadhG1Y%3D>. [Kasutatud 3 May 2019].
- [9] M. SDRAM, „Farnell,“ [Võrgumaterjal]. Available: https://ee.farnell.com/micron/mt48lc4m32b2p-6a-it/sdram-128mbit-166mhz-86tsop/dp/2253746?ost=2253746&ddkey=https%3Aet-EE%2FElement14_Estonia%2Fsearch. [Kasutatud 3 May 2019].
- [10] SearchStorage, „NOR flash technology,“ [Võrgumaterjal]. Available: <https://searchstorage.techtarget.com/definition/NOR-flash-memory>. [Kasutatud 3 May 2019].

- [11] Väikmälu, „Farnell,“ Cypress Semiconductors, [Võrgumaterjal]. Available: <https://ee.farnell.com/cypress-semiconductor/s25fl127sabmfv101/flash-memory-128mbit-108mhz-soic/dp/2772749>. [Kasutatud 3 May 2019].
- [12] Farnell, Analog devices, [Võrgumaterjal]. Available: <https://ee.farnell.com/analog-devices/adm3053brwz/ic-can-transc-sig-power-20soic/dp/2062027>. [Kasutatud 3 May 2019].
- [13] T. Instruments, „Mouser,“ unisolated CAN trans., [Võrgumaterjal]. Available: <https://www.mouser.ee/ProductDetail/Texas-Instruments/TCAN1042HDRQ1?qs=%2Fha2pyFadugypkkiYbgeDHRmboWhDLpyO57ur51j02eLi9e%2FBjInt6b6I9JXn2Ea>. [Kasutatud 3 May 2019].
- [14] V. regulators. [Võrgumaterjal]. Available: <https://www.renesas.com/us/en/products/power-management/linear-vs-switching-regulators.html>. [Kasutatud 4 May 2019].
- [15] „Newhaven display datasheet,“ Newhaven Display, [Võrgumaterjal]. Available: <https://www.mouser.ee/datasheet/2/291/NHD-4.3-480272EF-ATXL-T-231008.pdf>. [Kasutatud 7 May 2019].
- [16] L. S. Sterling, The Art of Agent-Oriented Modeling, London: The MIT Press, 2009.
- [17] Analogictips. [Võrgumaterjal]. Available: <https://www.analogictips.com/adcs-sufficient-sampling-nyquists-rate/>. [Kasutatud 30 April 2019].
- [18] E. Distrelec, Apem Compact Joystick, [Võrgumaterjal]. Available: <https://www.elfadistrelec.ee/et/compact-joystick-apem-nvh1d1c0cp2s/p/30100621>. [Kasutatud 1 May 2019].
- [19] Mikropure, „Tombstone effect,“ [Võrgumaterjal]. Available: <http://mikropure.com/en/tombstone-effect/>. [Kasutatud 6 May 2019].

Lisa 1: Mooduli korpuse algne disain



Lisa 2: Mooduli korpuse lõplik disain



Lisa 3: Kasutatud komponentide nimekiri

Manufacturer	Product Name	Supplier	Supplier Number	Quantity
ROHM	SMLE13EC8TT86	Farnell	2507534	4
MULTICOMP	MC0603X475K100CT	Farnell	2320818	2
WALSIN	0603B104K250CT	Farnell	2496833	28
MURATA	GRM188R71A225KE15D	Farnell	1797012	2
muRata	GRM1555C1H160JA01D	Farnell	2218841	2
WALSIN	0603B104K250CT	Farnell	2496833	23
MULTICOMP	MC0402B102K250CT	Farnell	1833876	12
muRata	GRM21BR71A106KE51L	Farnell	1828828	3
MULTICOMP	MC0805B475K160CT	Farnell	2320854	1
MURATA	GRM188R72A104KA35D	Farnell	1828921	1
MULTICOMP	MC0603B472K500CT	Farnell	1759098	1
YAGEO	CC0603KRX7R9BB103	Farnell	722236	2
AVX	12065C475KAT2A	Farnell	1908177	2
muRata	GRM21BR71C475KA73L	Farnell	1797015	3
ON SEMICONDUCTOR	MMSZ5228BT1G	Farnell	1431271	3
SEMTECH	UCLAMP3301D.TCT	Farnell	1456422	12
STMICROELECTRONICS	STPS1L40M	Farnell	2629693	1
TAIWAN SEMICONDUCTOR	S1AL	Farnell	1559148	1
ON SEMICONDUCTOR	MBRS340T3G	Farnell	9557377	1
BOURNS	SF-0603S050-2	Farnell	2291892	2
Molex	54132-4062	Farnell	2614924	1
MOLEX	90130-1212	Farnell	9733914	1
MOLEX	90130-1210.	Farnell	9733906	1
Molex	90130-1214	Farnell	9733922	1
MOLEX	90136-1204	Farnell	9733817	1
Murata Electronics	BLM15HD182SN1D	Mouser	81-BLM15HD182SN1D	2
TDK	1669744	Farnell	MPZ1608S300A	2
muRata	LQH32PN4R7NN0L	Mouser	81-LQH32PN4R7NN0L	1
Coilcraft	XFL3012-222MEB	Farnell	2289187	1
BOURNS	SRN8040-470M	Farnell	2061674	2
ROHM	SMLE13EC8TT86	Farnell	2507534	2
TAIWAN SEMICONDUCTOR	TSM2306CX	Farnell	1864577	1
ON SEMICONDUCTOR	FQD7P06TM	Farnell	2825194	1
YAGEO	RC0603FR-071KL.	Farnell	9238484	17
MULTICOMP	MCWR08X1401FTL	Farnell	2694989	1
YAGEO	RC0603FR-0710KL	Farnell	9238603	36
MULTICOMP	MCWR06X1002FTL	Farnell	2447230RL	24
PANASONIC	ERJPA3F60R4V	Farnell	2562115	2
PANASONIC	ERJ3EKF60R4V	Farnell	2059280	2
VISHAY	CRCW04024K70FKED	Farnell	1469712	1
PANASONIC	ERJ2GE0R00X	Farnell	2059190	1
VISHAY	CRCW0402100KFKED	Farnell	1469671	1

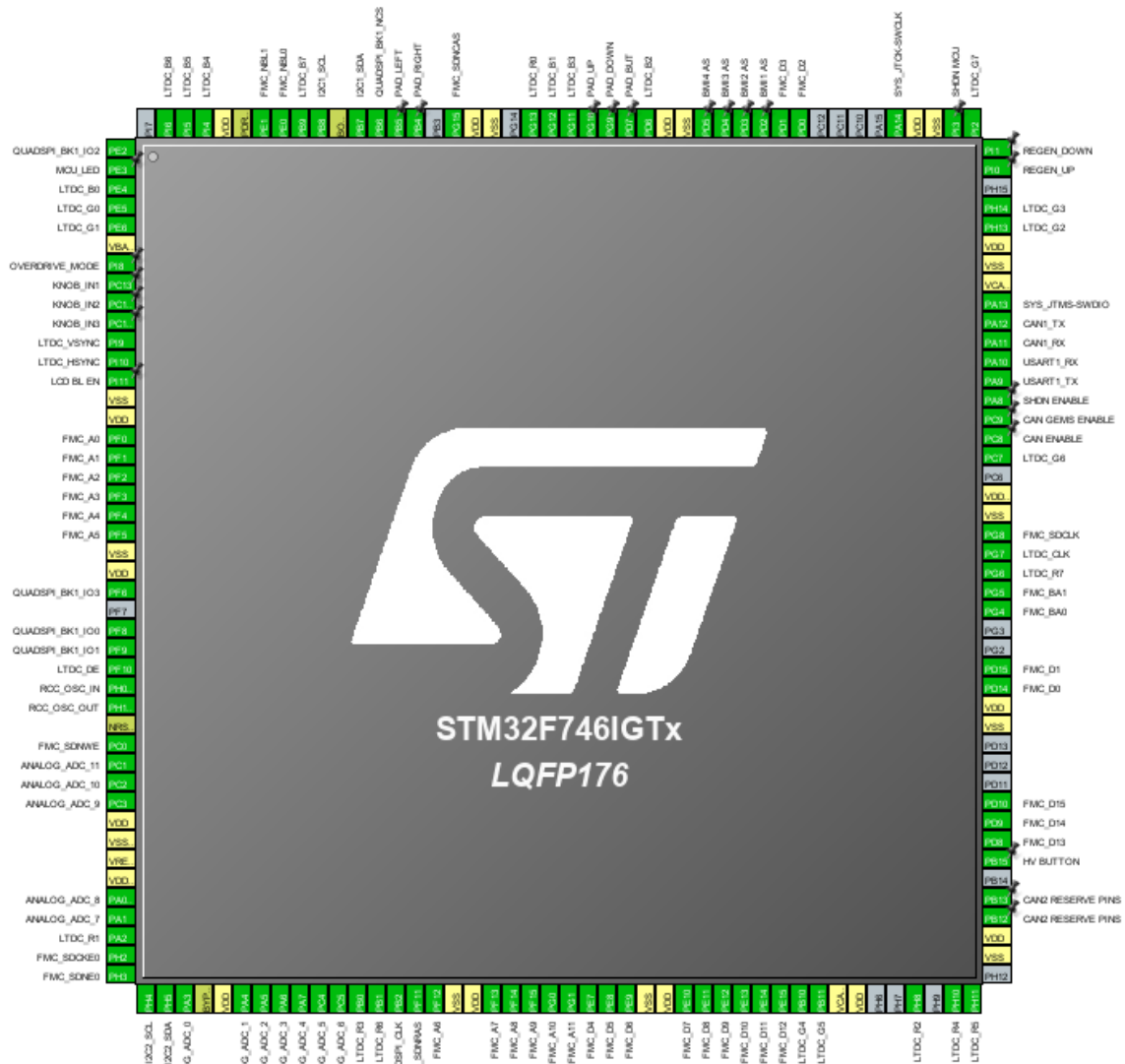
TE CONNECTIVITY	RP73PF2A10RBTDF	Farnell	2116948	1
MULTICOMP	MCWR06X1003FTL	Farnell	2447226	1
Yageo	RC0603FR-0710KL	Farnell	9238603	1
YAGEO	RC0603JR-071KL	Farnell	2309108	2
MULTICOMP	MCDHN-02F-V	Farnell	1605473	2
Micron	MT48LC4M32B2P-6A IT	Farnell	2253746	1
BOSCH	BMI-160	Mouser	262-BMI160	4
CYPRESS SEMICONDUCTOR	S25FL127SABMFV101	Farnell	2772749	1
Analog Devices	ADM3053BRWZ	Farnell	2062027	1
TI	TCAN1042HDRQ1	Mouser	595-TCAN1042HDRQ1	1
STMicroelectronics	STLD40DPUR	Mouser	511-STLD40DPUR	1
TI	LM3671MF-3.3/NOPB	Farnell	1685767	1
RICHTEK	RT9080-33GJ5	Farnell	2679852	1
ABRACON	ABM8G-25.000MHZ-18-D2Y-T	Farnell	2101339	1
Newhaven Displays	NHD-4.3-480272EF-ATXL#-ND	Digikey	NHD-4.3-480272EF-ATXL#-ND	1

Lisa 4: Süsteeminõuete tabel

Süsteeminõue	Vajalik perifeeria	Tüüp/pinge	Põhjendus
LCD ekraani draiver	LTDC	Digitaal, 3V3	RGB888
Väline kaadri puhver(frame buffer)	FMC	Digitaal, 3V3	Välise SDRAM-iga suhtlemiseks
Välkmälu	(Quad)SPI	Digitaal, 3V3	Pilootide seadistuste salvestamiseks
4 kiirendusandurit	2x I2C	Digitaal, 3V3	Kiirenduste ja nurkkiirenduste parema täpsusega lugemiseks
CAN siinil suhtlus	CAN1	Digitaal, 3V3	Sensorite info saatmiseks ning kuvatava info lugemiseks auto üldiselt suhtlusiinilt
CAN siini logimine <i>datalogger</i> -isse	CAN2	Digitaal, 5 V	Isoleeritud üldise suhtlussiini ümber konverteerimine isoleerimata siiniks
Gaasipedaali andur 1	ADC	Analoog, 3V3	
Gaasipedaali andur 2	ADC	Analoog, 3V3	
Piduripedaali asendi andur	ADC	Analoog, 3V3	
Esimese pidurisüsteemi rõhuandur	ADC	Analoog, 3V3	
Tagumise pidurisüsteemi rõhuandur	ADC	Analoog, 3V3	
Rooliasendi andur	ADC	Analoog, 3V3	
Esimese vasaku ülekande temperatuur	ADC	Analoog, 3V3	

Esimese parema ülekanne temperatuur	ADC	Analoog, 3V3	
Mootorite kontrollerite lülitamise nupp	GPIO	Digitaal, 3V3	
Sõidurežiimi lülitamise nupp	GPIO	Digitaal, 3V3	
Raadioside lülitamise nupp(roolil)	GPIO	Digitaal, 3V3	
Pidurienergia regeneerimise suurendamise nupp(roolil)	GPIO	Digitaal, 3V3	
Pidurienergia regeneerimise vähendamise nupp(roolil)	GPIO	Digitaal, 3V3	
Tankirežiimi/DRS lülitamise nupp(roolil)	GPIO	Digitaal, 3V3	
Navigatsioonipadja nupud	5x EXTI	Digitaal, 3V3	<i>Interruptide</i> baasil kiirema reageerimiskiiruse tõttu
Auto mootorite koguvõimsuse reguleerimise enkooder	3x EXTI	Digitaal, 3V3	<i>Interruptide</i> baasil kiirema reageerimiskiiruse tõttu

Lisa 5: Mikrokontrolleri konfiguratsioon



Lisa 6: Armatuurlaua disain FEST19



Lisa 7: SDRAM valideerimise fotomaterjal



Lisa 7.1 Fraasi kuvamine halva joodisega



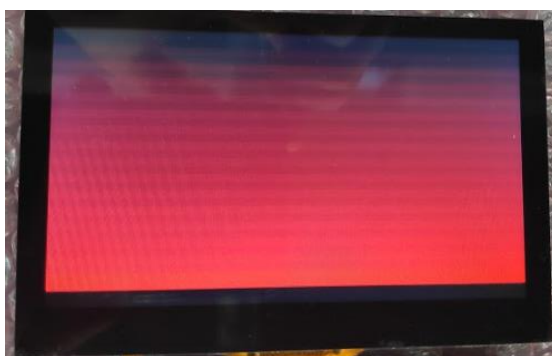
Lisa 7.2 Fraasi kuvamine peale uuesti jootmist



Lisa 7.5 Ebanormaalsed jooned kasutajaliidesel



Lisa 7.4 Gradiendi joonistamine MCU sisemälust

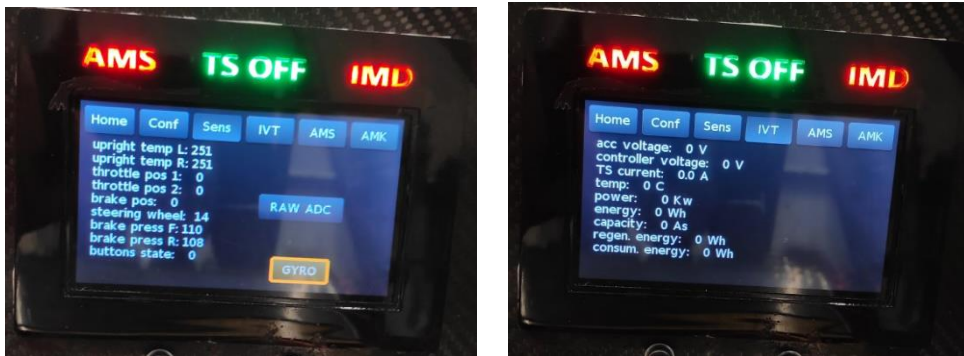


Lisa 7.3 Ebanormaalsed jooned ekraanil



Lisa 7.6 Normaalselt funktsioneeriva SDRAM gradiendi joonistamine

Lisa 8: Kasutajaliidese arendusjärg



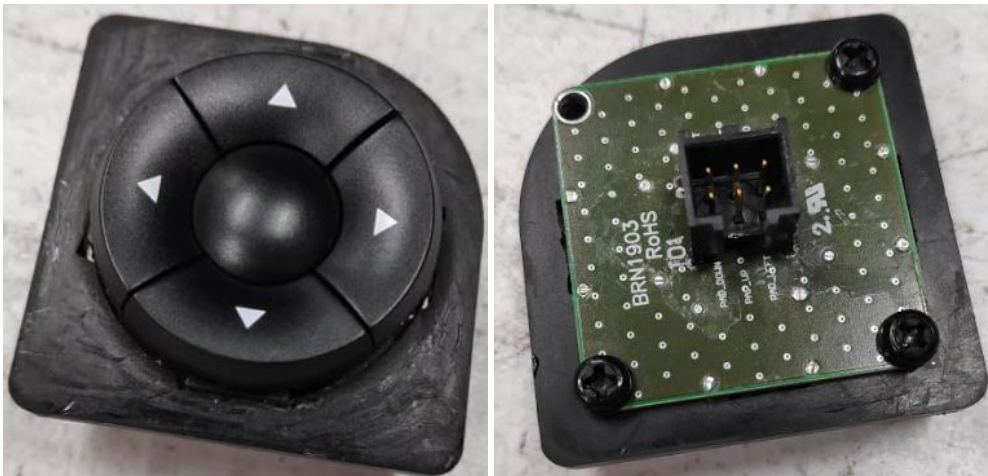
AMS lehekülj veata

AMS lehekülj vea ilmnemisel



AMK kontrolleri lehekülj koos veateatega

Lisa 9: Mooduli komponent - navigatsioonipadi



Lisa 10: Üldistatud kuju programmi tööst

