



TALLINNA TEHNIKAÜLIKOOL

INSENERITEADUSKOND

Elektroenergeetika ja mehhatroonika instituut

HÜDRAULILISE PÄIKEST JÄRGIVA KINNITUSLAHENDUSE VÄLJATÖÖTAMINE PÄIKESEPANEELIDELE

DEVELOPMENT OF A HYDRAULIC SOLAR TRACKING FRAME FOR SOLAR
PANELS

BAKALAUREUSETÖÖ

Üliõpilane: Joel Õispuu

Üliõpilaskood: 179012EAAB

Juhendaja: Vahur Maask, doktorant-nooremteadur

Tallinn, 2020

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

Autor: Joel Õispuu

Lõputöö liik: Bakalaureusetöö

Töö pealkiri: Hüdraulilise päikest järgiva kinnituslahenduse väljatöötamine päikesepaneelidele

Kuupäev: 20.05.2020

67 lk

Ülikool: Tallinna Tehnikaülikool

Teaduskond: Inseneriteaduskond

Instituut: Elektroenergeetika ja mehhatroonika instituut

Töö juhendaja(d): doktorant-nooremteadur Vahur Maask

Töö konsultant (konsultandid): Enno Niit

Sisu kirjeldus:

Lõputöö eesmärgiks oli projekteerida ja valmis ehitada päikesepaneelidele päikest järgiv raam. Töö tulemuseks oli kaks töötavat päikesejärgijat.

Lõputöö põhiosa on jaotatud kolmeks. Esimeses osas käsitletakse päikesepaneelide raamide projekteerimist ja ehitamist. Projekteerimise osas antakse ülevaade raami disainist ja tehakse tugevusarvutusi. Lisaks arvutatakse välja hüdraulika süsteemi erinevad parameetrid. Ehitamise osas räägitakse mõnest huvitavamast osast raamide ehitusprotsessist ja aja ning raha kulust mehaanika osale. Teises osas käsitletakse kontrolleri valimist ning elektriskeemi koostamist. Kontrolleri valimisel lähtutakse lõputöö väljapakujaga koostöös paika pandud kriteeriumistes. Elektriskeemi paremaks hoomamiseks jagati see neljaks osaks: juhtosa, andurid, täiturid ja käsijuhtimine. Lisaks räägitakse päikesejärgija maketist ja aja ning raha kulust elektroonika osale. Kolmandas osas käsitletakse programmi ülesehitust ja kirjeldatakse selle alamosi täpsemalt.

Märksõnad: päikese järgija, hüdraulika, projekteerimine, makett, Arduino Mega 2560

ABSTRACT

Author: Joel Õispuu

Type of the work: Bachelor Thesis

Title: Development of a hydraulic solar tracking frame for solar panels

Date: 20.05.2020

67 pages

University: Tallinn University of Technology

School: School of Engineering

Department: Department of Electrical Power Engineering and Mechatronics

Supervisor(s) of the thesis: Early Stage Researcher Vahur Maask

Consultant(s): Enno Niit

Abstract:

The aim of the thesis was to design and build a solar tracking frame for solar panels. The thesis resulted in two working solar tracking frames.

The main part of the thesis is divided into three. The first part provides an overview of design and construction process of solar tracker frame. In the design section, an overview of the frame design is given and strength calculations are made. In addition, various parameters for the hydraulic system are calculated. In the construction section, an overview of some of the more interesting parts of the frame construction process is given and time and money spent on the mechanical part are calculated. The second part addresses the process of choosing a controller and making of the electrical circuit. Controller is selected based of the criteria, established in cooperation with the thesis idea provider. For the ease of understanding the wiring diagram, it was divided into four parts: the control unit, sensors, actuators and manual control. In addition, an overview of a model of the solar tracker and the time and money spent on the electronics part is given. The third part discusses the structure of the program and describes its functions in more detail.

Keywords: solar tracker, hydraulics, design, model, Arduino Mega 2560

LÕPUTÖÖ ÜLESANNE

Lõputöö teema:	Hüdraulilise päikest järgiva kinnituslahenduse väljatöötamine päikesepaneelidele
Lõputöö teema inglise keeles:	Development of a hydraulic solar tracking frame for solar panels
Üliõpilane:	Joel Õispuu, 179012EAAB
Eriala:	Elektroenergeetika ja mehhatroonika
Lõputöö liik:	bakalaureusetöö
Lõputöö juhendaja:	Vahur Maask
Lõputöö ülesande kehtivusaeg:	20.05.2020
Lõputöö esitamise tähtaeg:	20.05.2020

Üliõpilane (allkiri)

Juhendaja (allkiri)

Õppekava juht (allkiri)

1. Teema põhjendus

Töö eesmärgiks on suurendada päikesepaneelidega toodetud energia kogust võrreldes mitte pööravate paneelidega. Teema on peamiselt oluline päikeseelektrijaamaga väiketootjatele, kelle tootmine on piiratud võrgu läbilaske võimega. Päikesepaneelide keeramine päikese suunas aitab suurendada päikesejaama kasutegurit, mis omakorda suurendab omaniku kasumit.

Turul pakutavate valmis lahenduste hinnad on väga kallid, ning elueaks peetakse orienteeruvalt pool päikesepaneelide eluiga. Töö eesmärgiks on luua lahendus, mis oleks soodsam ning vastupidavam.

2. Töö eesmärk

Töö eesmärgiks on projekteerida, ehitada ning programmeerida hüdrauliline päikesepaneelide päikesejärgimise süsteem.

3. Lahendamisele kuuluvate küsimuste loetelu:

1. Milline on päikesepaneelide kinnituslahendusele sobiliku juhtimissüsteemi ülesehitus?
2. Milline on päikesepaneelide kinnituslahenduse sobilik mehaaniline ülesehitus ning kas piisab ühest suurest raamist või tuleks kasutada mitut väiksemat raami?
3. Kas ise ehitatud, hüdraulikaga juhitud, päikest järgiv ajam on odavam kui turul pakutavad valmis süsteemid?

4. Lähteandmed

Kontrolleri valimisel kasutatakse peamiselt kontrollerite andmelehti.

Paneelide asetuse ning raamide arvu väljaselgitamiseks kasutatakse päikesepaneelide arvu ja mõõtmeid. Lisaks tuleb arvestada kasutatava maaala suuruse ja kujuga ning tuule mõjuga alusraamile ja selle komponentidele.

Komponendid tellitakse erinevatest e-poodidest, ning selle põhjal on võimalik arvutada ka raami maksumus. Valmis süsteemide hindade kohta saadakse teavet internetist. Vaadeldakse ainult eesti pakkujaid ning vajadusel küsitakse ka hinnapakkumist.

5. Uurimismeetodid

Raami konstruktsiooni kavandamisel ja uurimisel kasutatakse 3D CAD-tarkvara (SOLIDWORKS). Programmi kirjutamise ja katsetamise lihtsustamiseks ehitatakse makett, mille abil on võimalik emuleerida hüdraulika ja andurite osa. Komponentide lihtsamaks ja visuaalsemaks võrdluseks koostatakse võrdlustabel, mille alusel on võimalik antud süsteemi jaoks parim komponent välja selgitada.

6. Graafiline osa

Olulisemateks joonisteks on päikesepaneelide raami 3D mudeli pilt ning fotod valmis süsteemist. Nende jooniste eesmärk on luua visuaalne pilt teksti juurde ja seetõttu asetsevad need peamiselt töö põhiosas. Tabelitest on olulisemad eelpool mainitud võrdlustabel, mis asetseb töö põhiosas. Skeemidest kasutatakse töös elektroonika ja hüdraulika skeeme. Oma suuruse tõttu asetsevad need skeemid töö lisa, kuid vajadusel kasutatakse väiksemaid skeemi osasid ka töö põhiosas teksti selgitamiseks.

7. Töö struktuur

1. Peatükk raami konstruktsiooni kohta
 - 1.1. Raami projekteerimine ja jooniste tegemine
 - 1.2. Kasutatud komponendid ja materjalid
 - 1.3. Hüdraulika süsteemist
2. Peatükk elektroonika osa kohta
 - 2.1. Kriteeriumid süsteemile
 - 2.2. Kontrolleri valimine
 - 2.3. Ülevaade elektriskeemist
3. Peatükk programmi ja selle ülesehituse kohta
 - 3.1. Täpsemalt programmi erinevatest osades

8. Kasutatud kirjanduse allikad

Peamiselt kasutatakse internetis leitud erinevat tüüpi allikaid, näiteks seadmete ning komponentide andmelehti ja erinevate lisamoodulite teekide näite koode.

- 1) TipSolar, "Päikese jälgimissüsteemid" [Võrgumaterjal]. Saadaval: <http://www.tipsolar.ee/18-tooted-paigaldussusteemid-paikese-jalgimissusteemid.html> . [Kasutatud: 23.02.2020].
- 2) "ARDUINO MEGA 2560 REV3" [Võrgumaterjal]. Saadaval: <https://store.arduino.cc/arduino-mega-2560-rev3>. [Kasutatud: 23.02.2020].
- 3) A+, Raspberry Pi 3 Model" [Võrgumaterjal]. Saadaval: <https://www.raspberrypi.org/products/raspberry-pi-3-model-a-plus/> . [Kasutatud: 23.02.2020]
- 4) „Hüdraulika teoreetilised alused,“ Tallinna Tööstushariduskeskus, [Võrgumaterjal]. Saadaval: <http://www.ene.ttu.ee/leonardo/hydro/hydro02.pdf>. [Kasutatud 09. 05. 2020].

9. Lõputöö konsultandid

Enno Niit - lõputöö väljapakkuja ja raami ehitamiseks vajalike komponentidega ja seadmetega varustaja.

10. Töö etapid ja ajakava

Paneelide asukoha välja valimine ja maapinna ettevalmistamine (15.09.2019).

Raami jooniste koostamine (29.09.2019).

Raamide konstruktsiooni valmimine (30.11.2019).

Maketil töötav programm (26.01.2020).

Programmi katsetamine raamil (22.03.2020).

Peatükk elektroonika osa kohta (01.03.2020).

Peatükk raami konstruktsiooni kohta (15.03.2020).

Peatükk programmi ja selle ülesehituse kohta (29.03.2020).

Töö esimene versioon valmis (19.04.2020).

Lõputöö lõplik versioon (01.05.2020).

SISUKORD

LÕPUTÖÖ LÜHIKOKKUVÕTE.....	3
ABSTRACT	4
EESSÕNA.....	12
LÜHENDITE JA TÄHISTE LOETELU	13
SISSEJUHATUS	14
1. RAAMIDE PROJEKTEERIMINE JA EHITAMINE	15
1.1 Raamide projekteerimine	15
1.2 Tugevusarvutused.....	19
1.3 Hüdroskeem.....	21
1.3.1 Hüdropumba valimine.....	22
1.3.2 Hüdropumba mootori valik	24
1.3.3 Õli paagi valimine	25
1.4 Raamide ehitusprotsess.....	26
1.4.1 Hüdraulika reguleerimine.....	27
1.4.2 Andurite ja lisa stopperite monteerimine.....	27
1.4.3 Hinnanguline aja ja raha kulu raamide ehitamiseks	28
2. ELEKTROONIKA.....	29
2.1 Kriteeriumid millele programm pidi vastama	29
2.2 Kontrolleri valik	30
2.2.1 Arduino Mega 2560-st.....	30
2.2.2 Raspberry Pi 3 A+-st	31
2.2.3 Seadmete võrdlus.....	32
2.3 Manussüsteemid.....	33
2.4 Elektriskeemi ja selle ülesehitus	35
2.4.1 Elektriskeemi juhtosa	35

2.4.2	Kontrolleri sisendid.....	36
2.4.3	Kontrolleri väljundid.....	37
2.4.4	Käsijuhtimine.....	37
2.5	Elektriahela valmimine.....	37
2.5.1	Ekraani instaleerimine.....	38
2.5.2	Raamide makett	38
2.5.3	Hinnanguline aja ja raha kulu elektriahela valmimiseks	39
3.	Programmi koostamine ja tööpõhimõte	41
3.1	Manussüsteemid.....	41
3.2	Kellaaja lugemine ja kuvamine.....	41
3.3	Häirefunktsioon.....	42
3.4	Juhtpinge ja hüdropumba sisse lülitamine	42
3.5	Õlitaseme kontroll.....	42
3.6	Tuule tugevuse kontrollimine	43
3.7	Raamide keeramine	43
3.7.1	Keeramine potentsiomeetri järgi	44
3.7.2	Keeramine lõpu induktiivini	44
3.7.3	Keeramine alguse induktiivini	44
	KOKKUVÕTE.....	46
	SUMMARY	48
	KASUTATUD KIRJANDUS.....	50
	LISAD.....	52
	Lisa 1 Hüdraulikaskeem.....	53
	Lisa 2 Juhtosa elektriskeem.....	54
	Lisa 3 Andurite elektriskeem.....	55
	Lisa 4 Täiturite elektriskeem	56

Lisa 5 Käsijuhtimise elektriskeem	57
Lisa 6 Programmi põhimõtteline plokkskeem.....	58
Lisa 7 Programmi kood.....	59

EESSÕNA

Lõputöö idee autor on Enno Niit, kelle soovist rajada päikeseelektri jaam tekkis vajadus päikest järgiva päikesepaneelide raami järgi. Tema varustas lõputöö autorit kõikide töö vahendite, materjalide ja seadmetega.

Sooviksin tänada lõputöö juhendajat Vahur Maaski ja idee autorit Enno Niiti, kes olid suureks abiks lõputöö koostamisel.

Eriline tänu kuulub aga autori elukaaslasele, kes oli kogu kirjutamise vältel igati toeks nii nõu kui jõuga.

LÜHENDITE JA TÄHISTE LOETELU

W	vatt (võimsus)
m	meeter (pikkus)
m ²	ruutmeeter (pindala)
m/s	meetrit sekundis (kiirus)
N	njuuton (jõud)
kg/m ³	kilogrammi kuupmeetri kohta (tihedus)
N/mm ²	njuutonit tuurmillimeetri kohta (mehaaniline pinge)
g	gramm (mass)
bar	baar (rõhk)
l/min	liitrit minutis (vooluhulk)
I/O	<i>input/output</i> (sisend/väljund)
IDE	<i>integrated development environment</i> (integreeritud programmeerimiskeskond)
USB	<i>universal serial bus</i> (universaalne järjestiksiin)
V	volt (pinge)
A	amper (elektrivoolu tugevus)
DC	<i>direct current</i> (alalisvool)
RAM	<i>random access memory</i> (muutmälu)
Hz	herts (sagedus)
PWM	<i>pulse width modulation</i> (impulsilaiusmodulatsioon)
HDMI	<i>high-definition multimedia interface</i> (kõrglahutusega multimeediumliides)
WLAN	<i>wireless local area network</i> (traadita side kohtvõrk)
GPIO	<i>general-purpose input/output</i> (mitmeotstarbeline sisend/väljund)
ADC	<i>analog-digital converter</i> (analoog-digitaalmuundur)
RTC	<i>real-time clock</i> (reaalajakell)
OLED	<i>organic light-emitting diode</i> (orgaaniline valgusdiode)
I ² C	<i>Inter-Integrated Circuit</i> (kahejuhtmeliides)

SISSEJUHATUS

Üha enam inimesi soovib osa saada taastuvenergia tootmisest. Põhjuseid võib olla mitmeid, alustades soovist maailma parandada ning lõpetades kasumi teenimise sooviga. Üheks lihtsamaks ja enamlevinud viisiks on päikeseelektrijaama rajamine – olgu see siis mikro- või suurtootjana. Päikesepaneelide installeerimisel tekib alati küsimus, et millisesse ilmakaarde ja millise nurga alla need panna tuleks. Kõige õigem vastus sellele küsimusele oleks suunaga päikese poole. Parim ja ainuke lahendus selle saavutamiseks on päikesepaneelide installeerimine päikest järgivale raamile.

Ka Enno Niit, lõputöö väljapakkuja, soovis enda majapidamise rajada päikeseelektrijaama. Esialgu soovist ta rajada 15kW võimsusega parki, mis oleks olnud võimeline ära katma kogu majapidamise energiatarbimise ning oleks ka üle jäänud, et võrku müüa. Kahjuks on majapidamise juurde viivad elektriliinid suhteliselt kehvast seisukorras ning seetõttu lubati paigaldada ainult 11kW võimsusega jaam. Piirangust lähtuvalt otsustati installeerida 52 päikesepaneeli. Selleks, et toota sarnast energiakogust väiksema jaamaga võeti vastu otsus teha päikesepaneelidele pöörav raam. Pöörava raami eelis suurema arvu päikesepaneelide ees on see, et genereeritav võimsus jaguneb päeva peale ühtlasemalt ning seetõttu koormab elektriline korraga vähem.

Valmis päikese järgija pakkuja kodulehelt selgub, et järgija maksumus moodustab kuni 70% ülejäänud jaama maksumusest [1]. Sellest lähtuvalt oleks järgijate hinnaks kujunenud ligikaudu 7500€. Lisaks selgub, et järgija eluiga on palju lühem kui päikesepaneelide oma.

Eelnevast lähtuvalt otsustati ise projekteerida ja ehitada päikesepaneelidele päikest järgiv raam. Võimaldamaks suurema paneelide arvuga ja tugevamat raami otsustati raami liigutamiseks kasutada hüdraulikat.

Projekteerimisel kasutatud tarkvaradeks olid: SOLIDWORKS, FluidSIM, CircuitMaker ja Arduino IDE.

Käesolevas töös käsitletakse järgmisi teemasid:

- Päikesepaneelidele järgiva raami projekteerimine ning ehitamine.
- Elektriskeemi koostamine, sh kontrolleri valimine, ja elektriahela valmistamine.
- Programmi koostamine ja selle tööpõhimõtte.

Lisades on välja toodud:

- Hüdraulikaskeem
- Elektriskeem neljas osas
- Programmi tööpõhimõtet selgitav plokk skeem
- Programmi kood

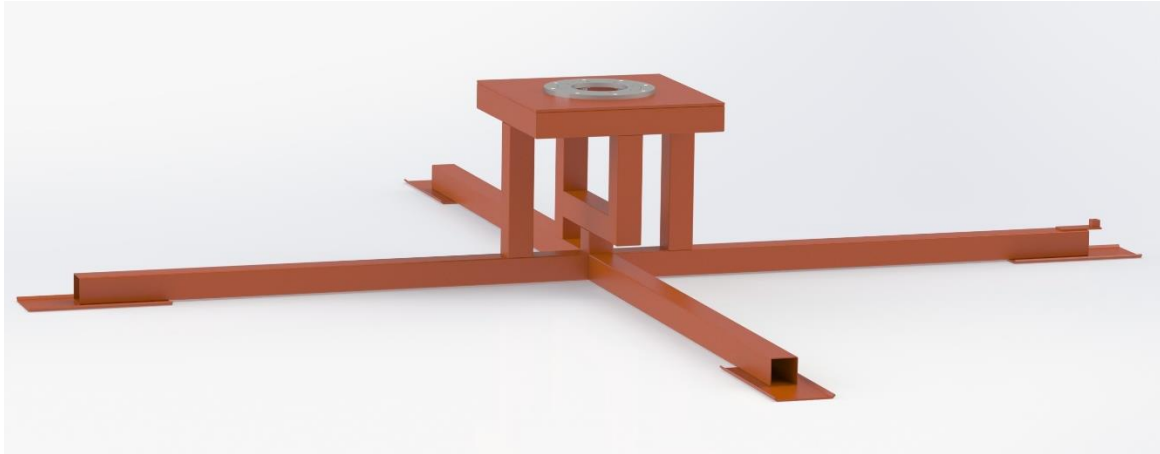
1. RAAMIDE PROJEKTEERIMINE JA EHITAMINE

Enne päikesepaneelide raami projekteerimist oli vaja selgeks teha kui mitu päikesepaneeli võiks ühel raamil olla, ning seeläbi teada saad mitu raami on vaja ehitada. Järgnevas oli vaja projekteerida raam, mille tarbeks tehti esmalt 2D visandeid ning lõpuks täielik 3D mudel. 3D mudeli koostamiseks kasutati SOLIDWORKS tarkvara. Kuna päikesepaneelide kallutamine ja keeramine toimub hüdraulika abil, siis oli vaja koostada ka hüdraulika skeem. Hüdraulika skeemi koostamiseks kasutati FluidSIM tarkvara. Viimase osana tuli valmis ehitada raamid ja päikesepaneelid üles seada.

Peamiseks põhjuseks, miks on vaja päikesepaneelid mitmele eraldi raamile panna ei ole mitte raami ja päikesepaneelide mass, nagu autor esmalt arvas, vaid hoopis päikesepaneelide pindala. Päikesepaneele on kokku 52 tükki. Üks paneel on mõõtmetega 1000*1600 mm [2], see tähendab, et ühele raamile pannes oleks päikesepaneelide kogupindala üle 80 m². See aga tekitab kaks probleemi, esmalt ideaalse kõrguse ja laiuse suhte leidmine ning teisalt tuule mõju raamile. Liiga laia raami ehitamine tekitab suure pöörderaadiuse probleemi ning kasutab seetõttu maa ala ebaefektiivselt. Liiga kõrge raami ehitamine võib aga tekitada raskusjõu tõttu probleeme päikesepaneelide alumises asendis olles või sealt üles tõstes. Lisaks tekib nii laia, kui ka kõrge raami puhul lisaprobleeme tuulega. Laia paneelide asetuse puhul hakkab raam tuule korral õõtsuma, ning tekib suurem oht keeramis süsteemi kahjustamiseks. Kõrge asetuse korral aga võib juba väiksem tuul hakata raami ümber lükkama. Liiga väikese paneelide arvu korral tuleb aga ehitada palju rohkem raame, mis suurendavad oluliselt kulutusi. Sellest lähtuvalt otsustati, et antud olukorras oleks optimaalne ehitada kaks raami, ning päikesepaneelid nende vahel võrdselt jaotada. Seega on kummalgi raamil 26 päikesepaneeli.

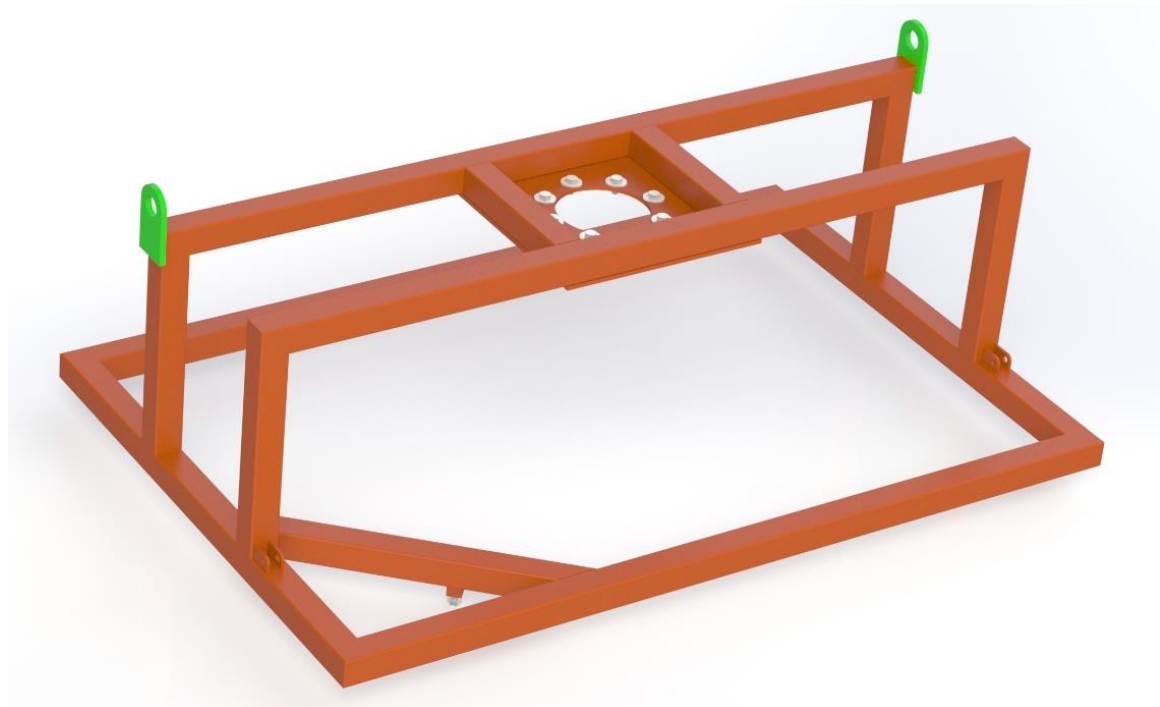
1.1 Raamide projekteerimine

Päikesepaneelide raamide projekteerimine jagati neljaks osaks. Esimeseks osaks sai raamide jalad (joonis 1.1). Ehituse lihtsustamiseks ning tagamaks ka võimalust vajadusel raamide asukohta muuta, sai otsustatud, et raamid ehitatakse maa peale ning neil puudub vundament. Raamide jalgadeks otsustati teha maapinna peal paiknevatest nelikant torudest rist, mille pikkus igas suunas on kolm meetrit. Lisaks paikneb iga jala otsas tald, mille tarbeks sai kasutatud kasutuna seisvaid 300*800*10 mm mõõduga plaate, mille alt kõvendati maapind killustikuga, et vältida konstruktsiooni vajumist. Jalgade risti keskele sai nelikanttorudest tehtud ühe meetri kõrgune platvorm, mille peale toetub pöördelaager. Kuigi pöördelaagri oleks saanud jalgade külge installeerida ka ilma platvormita, siis leidis vähemalt kaks põhjust, miks tasus kõrgendus ehitada. Esiteks ja peamiseks aitab see stabiliseerida kogu seda osa, mis toetub laagri peale ja seeläbi laagri eluiga pikendada. Teiseks on see seal paremini tolmu ja pori eest kaitstud.



Joonis 1.1 Raami jalad

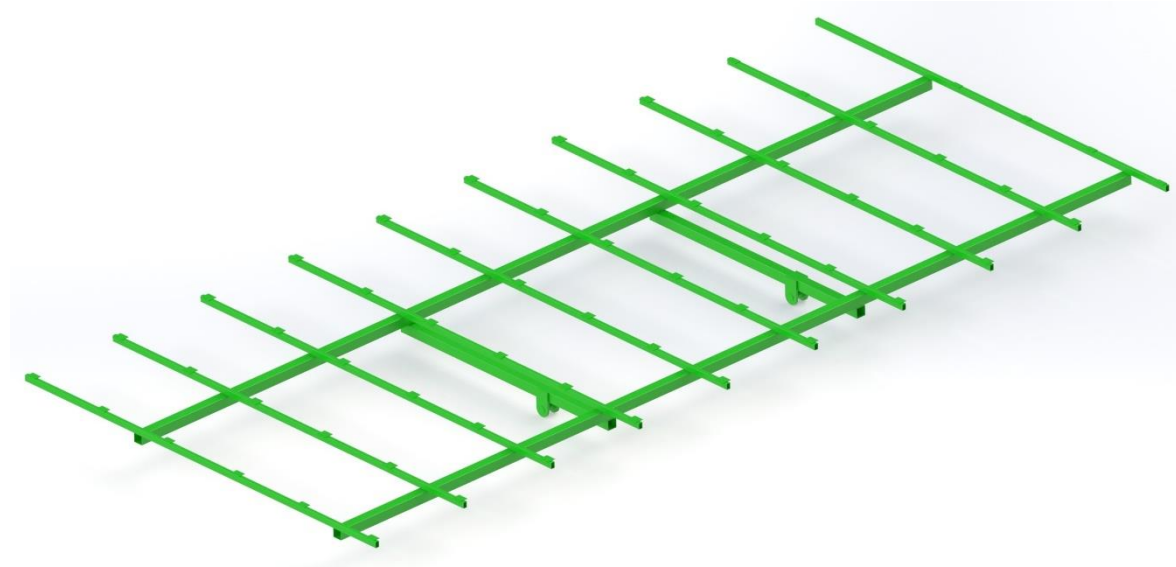
Teiseks osaks ehitati vaheraam (joonis 1.2), mis toetub pöördelaagri peale ning mille peale omakorda toetub päikesepaneele kandev raami osa. Vaheraam projekteeriti nii, et selle raskuskeske oleks palju madalamal kui pöördelaager, mille peale see toetub. Kuigi seeläbi ei õnnestunud kogu pöörava osa raskuskeset laagrist allapoole tuua, mis oleks stabiilsusele väga palju kaasa aidanud, siis õnnestus selle abil raskuskeset madalamale ehk laagrite lähemale tuua. Lisaks päikeseraame kandva osa kinnituskohdadele tuli selle osa projekteerimisel arvestada ka potentsiaalsete hüdrotsilindrite kinnituskohdadega.



Joonis 1.2 Raami keskmine osa

Kolmandaks osaks oli varem mainitud raami kõige pealmine ehk päikesepaneele kandev osa (joonis 1.3). Selle osa projekteerimise kõige suuremaks väljakutseks oli välja mõelda päikesepaneelide

asetus. Kõrguse poolest näis kõige optimaalsem asetada kolm paneeli püstiselt üksteise peale. Päikesepaneele oli ühel raamil 26 tükki ning see arv ei jagu kolmega, seetõttu jäeti kõige ülemisel real üks paneeli koht tühjaks. Päikesepaneelide elektrilise ühenduse mugavamaks tegemise ja visuaali parandamise eesmärgil jäeti tühjaks ülemise rea parem nurk. Samas jäi alles ka võimalus vajadusel tõsta ülemise rea keskmine paneel praeguse tühja augu asemele, kui peaksid tekkima probleemid tuule mõjuga raamile.



Joonis 1.3 Raami päikesepaneele kandev osa

Neljandaks osaks on hüdrotsilindrite kinnituste projekteerimine ning neile sobivate asukohtade leidmine raamil. Hüdrotsilindritest otsustati kasutada kummalgi raamil ühte kahe meetrise käiguga silindrit ja kahte ühe meetrise käiguga silindrit. Selliste pikkuste kasuks otsustati esiteks seetõttu, et selliste pikkustega silindrid olid kasutuna seisvalt olemas. Teiseks seetõttu, et need näisid sobivat juba valmis projekteeritud raami osadega. Kahe meetri pikkuse käiguga silindrit, mõõduga 80/50*2000 mm, kasutati päikesepaneelide pööramiseks ning see kinnitati ühest otsast alusraami jala külge ning teisest otsast keskise raami külge (joonis 1.4). Kahte ühe meetrist silindrit, mõõduga 63/45*1000 mm, kasutati päikesepaneelide tõstmiseks ja langetamiseks nii, et need aitaks tagada ülemise raami osa stabiilsust (joonis 1.5). Kuna silindrite käigu pikkust ei ole võimalik muuta, siis tuli kasuks 3D mudeli omamine, mille abil oli võimalik katseliselt ära määrata silindrite otsade kinnituste asukohad. Mudel võimaldas hõlpsasti katsetada kinnituste erinevate asukohtadega ning näha kuidas mõjutab silindri täielik pikendamine või lühendamine päikesepaneelide suunda. Juhul kui silindri asukoht ei olnud sobilik, näiteks tõstesilindri täielikul pikendamisel tõsteti päikesepaneelid liiga püsti ja langetamisel ei langetatud neid piisavalt madalale, siis võimaldas mudel väga lihtsalt silindrite otspunktide asukoha muutmist ning seejärel uuesti katsetamist. Selline katsetamine lihtsustas ja kiirendas projekteerimist.



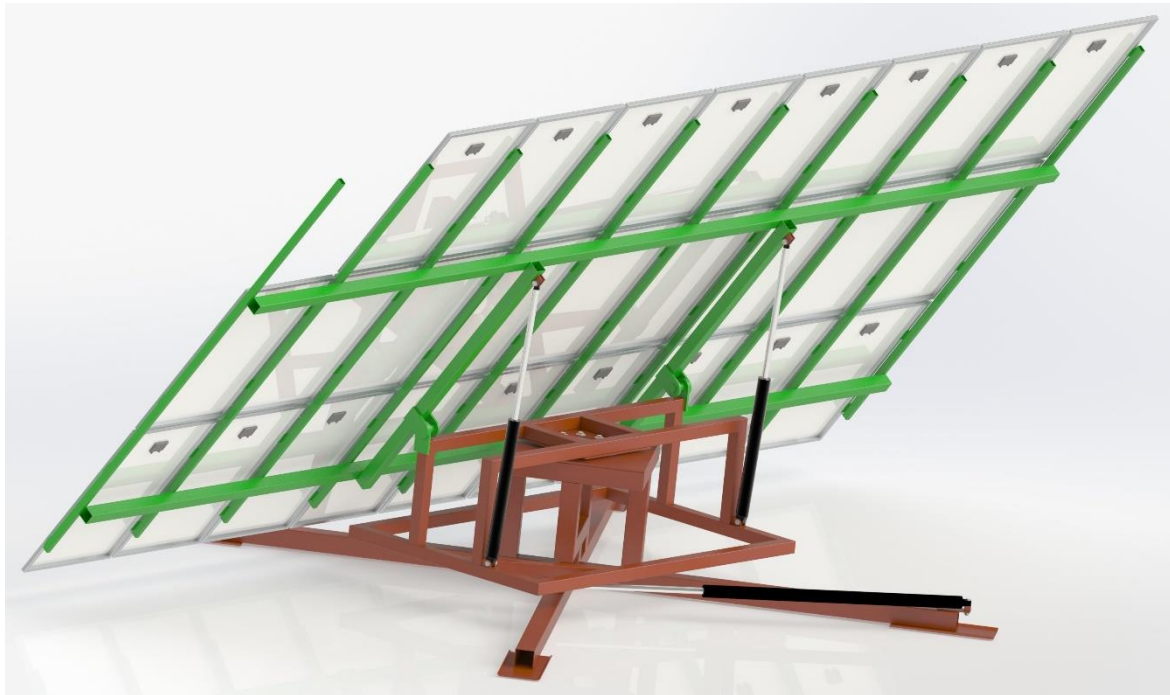
Joonis 1.4 Keeramissilindri asetus



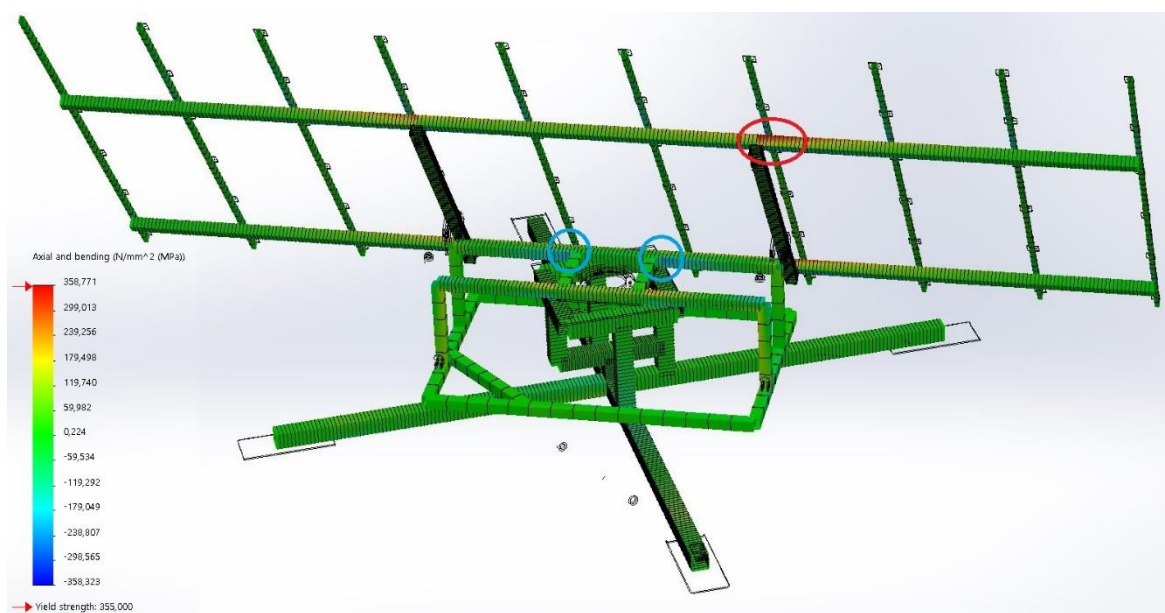
Joonis 1.5 Tõstesilindrite asetus

1.2 Tugevusarvutused

Eelpool mainitud osadest koostati päikesepaneelide raamide koost (joonis 1.6). Kuna hüdrocilindrid ja teised raami osad jäeti koostus liikuvaks nii nagu need tegelikkuses liiguks, siis oli võimalik kontrollida, et raami osade vahel ei tekiks konflikte. Lisaks võimaldas koostu olemasolu teha raamile tugevusarvutusi, mille abil oli võimalik ülesse leida raami nõrgimad kohad (joonis 1.7).



Joonis 1.6 Raami koost koos päikesepaneelidega



Joonis 1.7 Raami tugevusarvutused 35,5 m/s tuule korral. Punase ovaaliga on märgitud piirkond kus tekkisid kõige suuremad jõud. Siniste ringidega on märgitud piirkond, kus autori esialgsel hinnangul oleksid ka võinud tekkida suured jõud.

Tugevusarvutustest selgusid millised on raami nõrgimad kohad ning kui suurt tuult võiksid raamid vastu pidada. Raamide projekteerimisel seati eesmärgiks, et raam peab kindlast vastu pidama vähemalt 25 m/s tugevusega tuulele, kuid ideaalis võiks vastu pidada ka 30 m/s tuulele. Tuule poolt tekitatud jõu suurus arvutati ühe raami päikesepaneelide pindala ja tuule kiiruse järgi. Jõu arvutamise lihtsustamiseks arvutati jõud nii nagu tuule suund oleks risti päikesepaneelide pinnaga. Tegelikuses ei tõuse päikesepaneelid maapinna suhtes suurema nurga alla kui 55° ning seetõttu on reaalne jõud sama tuule kiiruse juures väiksem kui arvutuslikult saadud jõud, millega tehti ka tugevusarvutusi. Tuule poolt tekitatud jõu arvutamiseks kasutati järgnevaid valemeid. [3]

$$F = \frac{1}{2} * \rho * v^2 * A \quad (1.1)$$

kus F – tuule poolt tekitatav jõud, N,

ρ – õhu tihedus normaalrõhul, temperatuuril 0° C , 1,29 kg/m³ [4],

v – Tuule kiirus , m/s,

A – Päikesepaneelide pindala ühel raamil (26 tk), m².

$$A = (a * b) * 26 \quad (1.2)$$

kus a – päikesepaneeli lühema külje pikkus, a=1 m [2],

b – päikesepaneeli pikema külje pikkus, b=1.6 m [2].

Vastavalt valemile 1.2 arvutati ühe raami päikesepaneelide pindala.

$$A = (1 * 1.6) * 26 = 41,6 \text{ m}^2$$

Vastavalt valemile 1.1 arvutati välja erinevate tuule kiiruste korral tuule poolt tekitatav jõud raamile ning seejärel tehti saadud jõuga mudelil tugevusarvutused. Alustati tuule kiirusest 25 m/s ning liiguti suurem poole, kuni leiti maksimaalne tuule kiirus, mille puhul raamil tekkivad jõud ei ületa materjali voolavuspiiri. Saadud tulemused kanti tabelisse 1.1.

Tabel 1.1 Tugevusarvutuste tulemused

Tuule kiirus (m/s)	Tuule poolt tekitatav jõud (kN)	Suurim mehaaniline pinge (N/mm ²)
25	16,8	196
30	24,2	262
33	29,3	314
35	32,9	349
35,5	33,9	359

Tugevusarvutuste tulemustes pöörati peamiselt tähelepanu sellele, et kus tekkisid suuremad jõu kontsentratsioonid ning sellele, et jõud antud kohtades ei ületaks materjali voolavuspiiri. Materjalis

tekkivad jõud soovitud tuule kiiruse juures ei tohi ületada materjali voolavuspiiri, kuna vastasel korral läheb elastne deformatsioon üle plastseks deformatsiooniks. Plastse deformatsiooni korral tekib jääv deformatsioon, mis koormuse eemaldamisel ei kao, elastse deformatsiooni korral seda ei teki. [5]

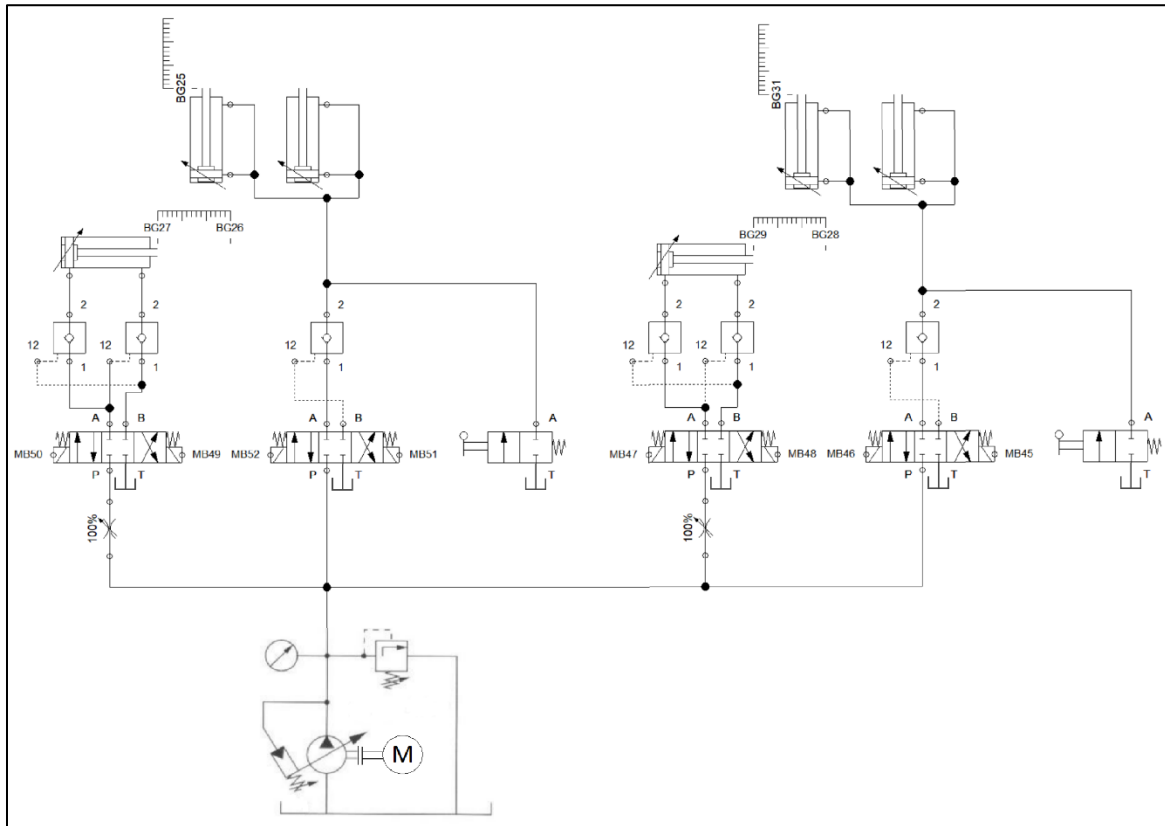
Päikesepaneelide raamide ehitamisel kasutati konstruktsiooniterasest S355 J2H valmistatud nelikanttorusid ja lehtmetailist osi. Antud terase voolavuspiiriks on 355 N/mm^2 (EN 10025-2) [6]. Suurim tuule kiirus, mille puhul jäid jõud alla lubatu suuruse oli 35 m/s , kus mõjus raamile jõud $33,9 \text{ kN}$ ning suurim mehaaniline pinge konstruktsioonis oli 349 N/mm^2 . Teras voolavuspiiri ületati alles $35,5 \text{ m/s}$ tuule puhul. Kõige suuremad jõud tekkisid joonis 1.7 punasega märgitud piirkonnas nagu ka autor esmalt arvas. Lisaks tekkisid suured jõud ka samal joonisel sinisega märgitud piirkondadesse, kuid need jäid arvatust palju väiksemaks, seetõttu ei tekita need mingit ohtu raamikonstruktsioonile ka suure tuule kiiruse korral.

Saadud tulemustest ja asjaolust, et antud paneelid paigaldatakse sisemaale võib järeldada, et tuul ei ole ohuks projekteeritud raami konstruktsioonile. Lisaks soodustab seda ka asjaolu, et päikesepaneelid langetatakse automaatselt üle 20 m/s tuule korral alumisse asendisse, et vähendada süsteemi kahjustus võimalusi.

1.3 Hüdroskeem

Kui raam oli valmis projekteeritud, oli vaja koostada hüdraulika skeem (joonis 1.8). Esmalt planeeriti kasutada ühel raamil kolme hüdrocilindrit ja kahte elektromagnetilist suunaventiili. Esimene suunaventiil oleks ühendatud keeramis silindri külge ning selle abil oleks saanud päikesepaneeli keerata edasi ja tagasi. Teine suunaventiil oleks ühendatud tõstvate silindrite külge ja silindrid oleks omavahel ühendatud paralleelselt. Selliselt ühendades oleks olnud võimalik paneelide alla laskmiseks kasutada hüdropumbalt saadavat rõhku. Antud konfiguratsiooniga tekib aga probleem juhul kui tekib elektrikatkestus ning mingil põhjusel, näiteks tuule tõttu, on vaja päikesepaneeli käsitsi alla lasta. Probleemi lahendamiseks muudeti paneelide tõstesilindrite ühendusviisi ning nende juurde lisati ka üks käsi juhtimisega suunaventiil. Tõstesilindrite mõlemad pordid ühendati suunaventiili porti nii, et päikesepaneeli tõstmisel tekiks rõhk silindrite mõlemasse poolde. Sellise ühenduse puhul toimub tõstmine hüdrocilindri kolvi pindalade erinevuse tõttu, ning sellist ühendust kasutatakse üldjuhul silindrite kiirliikumise tekitamiseks. Antud juhul aga ei olnud eesmärk saavutada kiirliikumist, vaid tekitada olukord kus päikesepaneelide alla laskmine toimub raskusjõu mõjul. Nii võimaldatakse päikesepaneelide käsitsi alla laskmist ka hüdropumpa käivitamata, kuna osa suurema ruumalaga silindri poolest välja surutavast õlist suunatakse väiksema ruumalaga silindri poolele ning ülejäänud suunatakse tagasi õlipaaki. Alternatiivselt oleks

olnud võimalik kasutada ühepoolse toimega hüdrocilindrit, kuid antud projekti raames oleks see tähendanud lisakulutuste tegemist.



Joonis 1.8 Hüdro skeem

1.3.1 Hüdropumba valimine

Hüdropumba valikul oli kaks tähtsat parameetrit, millega tuli arvestada. Esiteks peab pump suutma toota piisavat suurt rõhku, et võimaldada päikesepaneelide ülesse tõstmist. Teiseks peab olema pumba tootlikkus piisavalt suur, et tagada lühike päikesepaneelide keeramise aeg.

Süsteemi rõhu arvutamiseks oli vaja teada tõstesilindrite parameetreid, mis on varasemalt välja toodud, ja jõudu mida silinder peab olema võimeline arendama. Vajaliku jõu leidmiseks kasutati ülemise raami ja paneelide massi, mis on ligikaudu 1000kg ning see korrutati omakorda varuteguriga, milleks otsustati võtta 1,5. Varuteguri eesmärgiks on võimaldada ka pärast suurt lumesadu päikesepaneele ülesse tõsta ilma, et eelnevalt oleks vaja lumi maha lükata. Lisaks võimaldab see ka arvestamata jätta ka rõhulangud erinevates komponentides. Töörõhu leidmiseks kasutati järgnevaid valemeid. [7]

$$p = \frac{F}{A \cdot 0,1} \quad (1.3)$$

kus p – Töörõhk, bar,

F – Hüdrosilindri poolt arendatav jõud , N,

A – Tõstesilindrite säärite pindalade summa, mm².

$$F = m * g * K \quad (1.4)$$

kus m – Raami mass, kg,

g – Raskuskiirendus Maal, $g = 9,81 \text{ m/s}^2$,

K – Varutegur.

$$A = 2 * \frac{\pi * d^2}{4} \quad (1.5)$$

kus π – Archimedese konstant, $\pi = 3,14$,

d – Silindri sääre läbimõõt, mm.

Vastavalt valemile 1.4 arvutati vajalik hüdrosilindri poolt arendatav jõud.

$$F = 1000 * 9,81 * 1.5 = 14 \text{ kN}$$

Vastavalt valemile 1.5 arvutati hüdrosilindrite säärite pindala.

$$A = 2 * \frac{3,14 * 45^2}{4} = 3179 \text{ mm}^2$$

Vastavalt valemile 1.3 arvutati hüdropumba vajalik tööõhk.

$$p = \frac{14000}{3179 * 0,1} = 44 \text{ bar}$$

Hüdropumba sobiva tootlikkuse leidmiseks seati eesmärk, et päikesepaneelid peavad tagumisest äärmisest asendist algusasendisse jõudma keerata 45 sekundiga, ehk 0,75 minuti jooksul. Tootlikkuse arvutamiseks kasutati antud aega ja hüdrosilindri mõõtmeid. Kuna päikesepaneelide vastupäeva pööramine käib hüdrosilindri lühendamisega, siis kasutatakse arvutamisel kolvi sääre poolset ruumala. Arvutused tehti järgnevate valemitega:

$$Q = \frac{V}{t} \quad (1.6) [7]$$

kus Q – Vajalik vooluhulk, l/min,

V – Hüdrosilindri sääre poolne ruumala , l,

t – Aeg hüdrosilindri täitmiseks, min.

$$V = A * l * 10^{-6} \quad (1.7)$$

kus A – Silindri sääre poolne kolvi pindala, mm^2 ,

l – Keeramissilindri käigupikkus, mm .

$$A = \frac{\pi * D^2}{4} - \frac{\pi * d^2}{4} \quad (1.8)$$

kus π – Archimedese konstant, $\pi = 3,14$,

D – Keeramissilindri kolvi läbimõõt, mm ,

d – Keeramissilindri sääre läbimõõt, mm .

Vastavalt valemile 1.6 arvutati silindri sääre poolne kolvi pindala.

$$A = \frac{3,14 * 80^2}{4} - \frac{3,14 * 50^2}{4} = 3062 \text{ mm}^2$$

Vastavalt valemile 1.7 arvutati hüdrosilindri sääre poolne ruumala.

$$V = 3062 * 2000 * 10^{-6} = 6,12 \text{ l}$$

Vastavalt valemile 1.8 arvutati vajalik vooluhulk, et päikesepaneele keerata piisavalt kiiresti.

$$Q = \frac{6,12}{0,75} = 8,2 \text{ l/min}$$

Arvutatud rõhust 44 baari ja vooluhulgast 8,2 l/min lähtuvalt oli võimalik valida välja hüdropump. Olemasolevatest pumpadest valiti välja kaldplaadiga aksiaalkolbpump, mille maksimaalne tööõhk on 300 baari ja tootlikkus on 18 l/min. Kuigi mõlemad näitajad on palju suuremad kui vaja, siis ei olnud finantsiliselt kasulik osta väiksemate parameetritega pumpa. Antud pumba puhul on võimalik väga lihtsalt otse pumba pealt reguleerida tööõhku ning see seati väljaarvutatud väärtusele. Pumba maksimaalse vooluhulga puhul muutub päikesepaneelide keeramine väga kiireks ja rapsivaks, seetõttu lisati hüdroseemi kaks reguleeritavat drosselit, mille abil on võimalik piirata vooluhulka pööramise silindritesse.

1.3.2 Hüdropumba mootori valik

Kuigi valitud hüdropumbale oli juba varasemast külge kinnitatud kolmefaasiline elektrimootor siis veendumaks selle piisavas võimuses, arvutati välja pumba teoreetiline võimsustarve. Selleks kasutati varasemalt välja arvutatud maksimaalset rõhku ja pumba maksimaalset tootlikkust. Maksimaalset tootlikkust kasutatakse päikesepaneelide ülesse tõstmisel, millal kasutatakse ka maksimaalset rõhku. Pumba teoreetilise võimsuse arvutamiseks kasutatakse järgnevat valemit:

$$P = \frac{Q_{max} * p}{600} \quad (1.9) [7]$$

kus P – Pumba teoreetiline võimsustarve, kW,

Q_{max} – Pumba maksimaalne vooluhulk, l,

p – Töörõhk, bar.

Vastavalt valemile 1.9 arvutati pumba teoreetiline võimsustarve.

$$P = \frac{18 * 44}{600} = 1.32 \text{ kW}$$

Lähtuvalt pumba teoreetilisest võimusest, sobiks ideaalselt 1,5 kW elektrimootor, mis töötaks kuni 90% koormusega. Sellise võimsusega mootorit ei olnud käepärast võtta, seega otsustati kasutada pumba küljes olnud 2.2 kW mootorit. Sellise võimsusega mootor ei tööta küll oma kõige efektiivsemas alas, kuid kuna mootor töötab päevas keskmiselt alla 15 minuti, siis ei ole energiakadu ka väga suur.

1.3.3 Õlipaagi valimine

Hüdraulika õlipaagiks sooviti kasutada olemasolevat 70 liitrist paaki. Veendumaks selle sobivuses oli vaja kokku arvutada kõigi hüdrotsilindrite säärte ruumala. Lisaks tuli arvestada sellise varuga, et ka kõikide silindrite maksimaalsel õliga täitumisel oleks õli niipalju mahutis, et pumba imitoru ots oleks sellesse sukeldatud. Süsteemi hüdrotsilindrite säärte ruumala arvutamiseks arvutati välja ühe tõstesilindri ruumala mis korrutati neljaga ja ühe keeramise silindri ruumala mis korrutati kahega. Ruumala arvutamiseks kasutati järgnevat valemit:

$$V = \left[\left(\frac{\pi * d_1^2}{4} * l_1 \right) * 4 + \left(\frac{\pi * d_2^2}{4} * l_2 \right) * 2 \right] * 10^{-6} \quad (1.10)$$

kus V – Hüdrotsilindrite säärte kogu ruumala, l,

d_1 – Tõstmissilindri sääre läbimõõt, mm,

l_1 – Tõstmissilindri käigupikkus, mm,

d_2 – Keeramissilindri sääre läbimõõt, mm,

l_2 – Keeramissilindri käigupikkus, mm.

Vastavalt valemile 1.10 arvutati hüdrotsilindrite säärte kogu ruumala.

$$V = \left[\left(\frac{3,14 * 45^2}{4} * 1000 \right) * 4 + \left(\frac{3,14 * 50^2}{4} * 2000 \right) * 2 \right] * 10^{-6} = 14,2 \text{ l}$$

Hüdrosilindrite säärite ruumalaks saadi 14,2 l. Varu arvestamine sõltub peamiselt paagi kujust, ehitusest, näiteks vaheseinte olemasolu, ja imitoru asukohast. Väljavalitud 70 liitrisel mahutil on piisavalt suur varu, seega sobib antud mahuti päikesepaneelide keeramis süsteemi.

1.4 Raamide ehitusprotsess

Eialgu otsustati ehitada valmis esimene raam nii, et seda saaks katsetada käsijuhtimisega (joonis 1.9). Alles peale katsetuste tegemist alustati teise raami ehitusega. Selline viis võimaldas veenduda kas projekteeritud disain töötab ning vajadusel viia sisse muudatusi teisel raamil. Peamine muudatus, mis teise raami ehitamisel sisse viidi oli päikesepaneelide asetuse raamil (joonis 1.10). Muudatuse kasuks otsustati peamiselt tuule poolt tekkivate mõjude tõttu, mis panid raami kõikumama tema pöörlemise teljes ning mis oli tõenäoliselt tingitud raami laiusel. Teisel raamil asetati päikesepaneelid 7*4 formaadis, kus mõlemad ülemised nurgad jäeti tühjaks. Kuigi teine päikesepaneelide asetuse leevendas kõikumise probleemi, siis otsustati, et probleem ei ole piisavalt suur, et oleks vaja esimene raam ümber ehitada.



Joonis 1.9 Esimene raam koos päikesepaneelidega



Joonis 1.10 Teine raam koos päikesepaneelidega

1.4.1 Hüdraulika reguleerimine

Hüdropumba rõhu sobivaks reguleerimiseks kasutati manomeetrit. Alustati pumba madalaimast rõhust ning seda suurendati nii kaua kuni saavutati soovitud rõhk 44 baari. Drosselite, mis piiravad päikesepaneelide keeramise kiirust, reguleerimiseks kasutati katselist meetodit. Selleks mõõdeti aega, kui kaua läheb päikesepaneelide ühest äärmisest asendist teise keeramiseks. Saadud tulemuse järgi piirati või suurendati drosseli läbilaskevõimet vastavalt vajadusele ning tehti uus katse. Lõpptulemusena saavutati päikesepaneelide soovitud tagasi keeramise aeg 45 sekundit.

1.4.2 Andurite ja lisa stopperite monteerimine

Päikesepaneelide automaatjuhtimiseks oli tarvis lisada mõlemale raamile kolm induktiivandurit ja üks potentsiomeeter. Esimene induktiivandur annab signaali siis kui raam on keeratud algusaasendisse ehk itta. Teine andur annab signaali siis kui raam on keeratud lõppu ehk läände. Kolmas andur annab signaali siis kui raam on alumises asendis. Potentsiomeeter installeeriti raami keskele, pöördelaagri kohale nii, et raami keerates muutub liuguri takistus ja seeläbi ka pinge kontrolleri sisendis.

Raami täielikul algusesse või lõppu keeramisel tekib olukord, kus hüdrosilinder jääb sellise nurga alla, et raamil on suur jõuõlg keeramise suunal. Antud olukordades tekib raamil soodne võimalus, tuule abil, pöörlemiseljes kõikuma hakata. Selle takistamiseks installeeriti kaks kummpatja, üks algusasendi ja teine lõpuasendi jaoks, mille vastu surutakse raami keerav osa (joonis 1.11). Selliselt on raam antud asendites pinge all ning ei hakka tuulega kõikuma.



Joonis 1.11 Lõpuasendisse keeratud päikesepaneelide raam. Punase noolega on märgitud päikesepaneelide keeramise suund. Sinise ristkülikuga on märgitud kummipadi, mille vastu päikesepaneelide raam keeratakse. Kollase ringiga on märgitud raami lõpuinduktiiv.

1.4.3 Hinnanguline aja ja raha kulu raamide ehitamiseks

Kogu ehitusprotsess võttis aega umbkaudu 100 töötundi. Selle aja sisse on arvestatud nii projekteerimisele, asukoha ettevalmistamisel kui ka ehitusele ja hüdraulika katsetusele kulunud aeg.

Eesmärgiks seati ehitamisel kasutada ära võimalikult palju juba olemas olnud materjale. Juurde ostetud materjalide hinnaks kujunes 3400 €. Ligikaudu 80% sellest summast kulus nelikanttorude ostmiseks. Ülejäänud raha kulus peamiselt hüdraulika komponentidele. Lisaks osteti pisidetaile nagu polte, mutreid ja keevituselektroode.

Ostes kõik detailid, kuid tehes mööndusi mõningates juppides, kujuneks raamide ehitamise hinnaks ligikaudu 6000-6500 €. Möönduste all peetakse silmas näiteks kolbpumba asemel palju soodsama hammasratas pumba kasutamist. Lisaks on võimalik palju raha kokku hoida kalli pöördelaagri pealt, soetades mõne kasutatud, kuid töökorras, kraana või ekskavaatori pöördelaagri.

2. ELEKTROONIKA

Enne elektroonika osa koostamist oli vaja lõputöö väljapakujaga kokku leppida kriteeriumites, mida peab kontrolleri ja programm suutma täita. Järgnevalt valiti välja kontrolleri, mis oleks võimeline täita etteantud kriteeriumeid. Kontrolleri järgi oli võimalik välja valida ka teised elektriskeemi koostamiseks vajalikud komponendid ning seejärel koostati elektriskeem. Elektriskeemi koostamiseks kasutati tarkvara CircuitMaker.

2.1 Kriteeriumid millele programm pidi vastama.

Järgnevalt on üles loetletud kriteeriumid, mille põhjal valiti välja kontrolleri ja elektriskeemi komponendid. Lähtuvalt nendest kirjutati ka kontrolleri programm. Lisatud on ka lühike selgitus või põhjendus kriteeriumi kohta.

- 1) Päikesepaneelide õigesse asendisse keeramine vastavalt kellaajale – Päikesepaneeli keeratakse päikese poole mitte päikese intensiivsuse, vaid programmi kirjutatud kellaegade ja etteantud asendite järgi. See aitab vähendada hüdraulika pumba sisselülitamise kordade arvu ning seeläbi energiat säästa.
- 2) Kellaaja hoidmine täpsusega ± 1 sekund nädalas – Selline täpsus tagab selle, et ühe aasta jooksul erineb süsteemi kell tegelikust kellaajast vähem kui ühe minuti võrra. Päikesepaneelide keeramine ei ole väga aegkriitiline, seega võib seade töötada vähemalt viis aastat ilma, et vajaks kellaaja täpsustamist. Elektrikatkestuse ajal ja peale seda peab süsteem olema võimeline säilitama õiget kellaega.
- 3) Õlitase kontroll hüdropaagis – Vältimaks pumba kuivalt ehk ilma õlita töötamist, on vaja kontrollida õlitaset õlimahutis enne hüdropumba käivitamist ja selle töös oleku ajal. Juhul kui õlitase langeb alla lubatud piiri läheb süsteem häire režiimi, mis peatab pumba töö.
- 4) Tuule kiiruse kontrollimine – Tuule kiiruse kontrollimine on vajalik selleks, et vältida päikesepaneelide kahjustada saamist tuule tõttu. Anemomeetrilt saadud info põhjal teeb kontrolleri otsuse, kas tuule kiirus on liiga pikalt olnud üle lubatud piiri ning vajadusel lastakse paneelid alumisse asendisse ja viiakse seade häire režiimi.
- 5) Häire režiimi nullimine ainult käsitsi – Häire režiimi nullimist saab teha ainult käsitsi ja seadme juurest. Selle eesmärk on kaitsta seadet, takistades selle iseenesest taas käivitamist häire tekitanud signaali katkemisel.
- 6) Käsitsi juhtimise võimalus – Käsijuhtimis võimalus on oluline peamiselt hooldustöödeks. Selle tarbeks koostatakse eraldi stand, millel olevate nuppudega on võimalik juhtida hüdrotsilindreid.

- 7) Häirete kuvamine ekraanil - Süsteem saab häire režiimi minna erinevatel põhjustel, seetõttu on oluline, et oleks võimalik lihtsalt ja kiiresti teada saada häire tekke põhjus. See lihtsustab tekkinud vea kõrvaldamist.

2.2 Kontrolleri valik

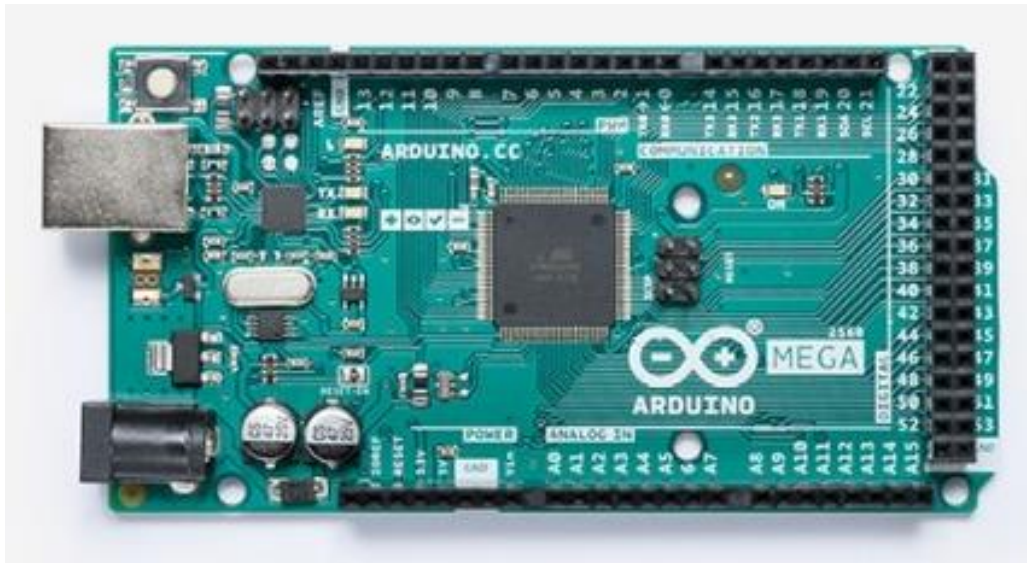
Päikesepaneelide raamide automaatseks juhtimiseks oli vaja välja valida kontroller, millega oleks võimalik: juhtida hüdraulika suunaventiile, võtta vastu infot nii paneelide asukoha kui ka tuule kiiruse kohta ja vajadusel kuvada ekraanile veateateid. Sellest lähtuvalt otsustati vaadelda lähemalt Arduino ja Raspberry Pi tootevalikut. Arduino perekonnast sai välja valitud kõige suurema I/O arvuga plaat, milleks osutus Arduino Mega2560 ja Raspberry Pi perekonnast sai välja valitud Raspberry Pi 3 A+. viimane sai välja valitud seetõttu, et see on eelmisest põlvkonnast ja seetõttu natukene soodsam kui praegune mudel. Järgnevalt antakse mõlemast lühike ülevaade ning võrreldakse neid kahte omavahel käesolevast projektist lähtuvalt.

2.2.1 Arduino Mega 2560-st

Arduino eesmärgiks on olla lihtsasti kasutatava tark- ja riistvaraga elektroonikaplatvorm, mida saaks kasutada väga erinevate keerukus astmetega projektides. Operatsioonisüsteemi puudumine annab talle mitmeid eeliseid. Esimeseks suureks eeliseks on see, et esmakordselt plaati kasutusele võttes ei ole vaja installeerida operatsioonisüsteemi, mis võib võtta palju aega ning olla keeruline. Tänu operatsioonisüsteemi puudumisele puudub ka oht selle poolt tekitatud vigadeks ja riketeks ning plaadi taaskäivitamisel hakkab see automaatselt programmi uuesti algusest täitma. Arduino programmeerimiseks võib kasutada interneti versiooni Arduino IDE-st või alla laadida ja installida Arduino-IDE programm. Mõlemal juhul on see palju lihtsam kui operatsioonisüsteemi installimine. Arduino IDE põhineb C++ programmeerimiskeelel, kuid on mõnel määral muudetud ja lihtsustatud. Programmeerimise ja katsetamise käigus saab Arduino oma toitepinge USB juhtme kaudu arvatist, millega toimub programmeerimine. Lisaks on osadel Arduino plaatidel, nagu näiteks „UNO“ ja „MEGA“, külge ehitatud „DC21MMX“ port, mis võimaldab kasutada 7-12 V ja vähemalt 250 mA DC toiteallikat. [8]

Arduino Mega 2560 (joonis 2.1) on disainitud suuremate projektide jaoks. Seetõttu on sellel teistest Arduino plaatidest rohkem I/O klemme, RAM-i ja programmimälu. Mega kasutab Atmel-i ATmega2560 protsessorit, mis töötab taktsagedusega 16 MHz. Plaadil on 54 digitaalset klemmi, mida saab programmi abil sisenditeks ja väljunditeks deklareerida. Neist 54-st on 15 võimalised tekitama PWM signaali. Lisaks digitaal sisenditele on plaadil ka 16 analoog sisendit. I/O klemmide

väljundpinge on 5 V ning neist iga võimaldab maksimaalselt 20 mA voolutarvet. Lisaks on plaadil üks 3.3 V, neli 5 V ja viis maanduse pesa. [9]



Joonis 2.1 Arduino Mega 2560 [9]

2.2.2 Raspberry Pi 3 A+-st

Raspberry Pi eesmärgiks on pakkuda soodsalt väikeste mõõtudega arvutit, millega oleks võimalik arendada programmeerimisoskusi ning kasutada väliseid I/O klemme erinevate projektide koostamisel. Raspberry pi võimaldab installida erinevaid operatsioonisüsteeme, näiteks Windows 10 IoT, Ubuntu või Raspberry Pi enda lahendust Raspbian. Raspberry Pi on ise väikeste mõõtudega arvuti, seega pole selle programmeerimiseks eraldi arvuti vaja. Programmeerimiseks on vaja kuvarit, mille saab lihtsalt ühendada HDMI pordi kaudu, ning klaviatuuri ja hiirt, mille saab ühendada USB porti. Toiteallikana saab kasutada 5.1 V ja 2.5 A DC toiteploki, mis ühendatakse olenevalt Raspberry Pi tüübist kas USB-C või micro-USB kaabli abil. [10]

Raspberry Pi 3 A+ (joonis 2.2) kasutab 1.4 GHz taktsagedusega nelja tuumalist protsessorit, mis kasutab ARMv8 64-bitilist käsustikku. Plaadile on integreeritud ka 2.4 GHz ja 5 GHz WLAN, täissuuruses HDMI, USB 2.0 ja audio ühendus. Nagu paljudel teistel Raspberry Pi mudelitel on ka sellel kasutusel 40 GPIO klemmi. Kuigi kaks GPIO klemm väljastavad pinget 5 V, siis ülejäänud kasutavad 3.3 V – see on ka maksimaalne sisendpinge antud klemmidele. Maksimaalne voolutugevus kõikide klemmide peale kokku on 50 mA. Vaikimisi on iga klemmi maksimaalne vool seatud 8 mA peale, kuid seda on tarkvara abil võimalik muuta vahemikus 2 mA kuni 16 mA. Kõik GPIO klemmid on võimalik seadistada PWM signaali tekitama, kuid kuna puudub ADC siis pole sellega võimalik analoog signaale mõõta. [11]



Joonis 2.2 Raspberry Pi 3 A+ [11]

2.2.3 Seadmete võrdlus

Neist kahest süsteemist parima välja selgitamiseks koostati võrdlustabel (tabel 2.1), milles on välja toodud võrreldav omadus ja selle tähtsus käesoleva projekti jaoks. Kõige olulisem asi selle projekti jaoks on piisav sisendite ja väljundite arv. Kahe paneeli raami juhtimiseks ning tuule lipult info saamiseks on vaja kokku ligikaudu 26 sisendit ja väljundit. Arduino Megal on 56 I/O klemmi ja Raspberry Pil on neid 40, millest ainult 28 on kasutatavad väljundite või sisenditena. Seetõttu osutus paremaks Arduino, kuna see võimaldab soovi korral tulevikus projekti laiendada. Projekti üheks eesmärgiks on hoida kulud võimalikult madalad, seega on ühel tähtsamal kohal ka hind. Selles kategoorias edestas Raspberry Pi Arduinot. Hinnad olid vastavalt 30 € [12] ja 37,50 € [13]. Järgnevalt oluline on analoog sisendite olemasolu, mida võiks olla vähemalt kolm. Arduino Mega omab koguni 16 analoog sisendit, Raspberry Pil pole mitte ühtegi, seega parem on Arduino. Kuigi ADC lisamooduli abil on võimalik ka Raspberry Pi-ga analoog signaali lugeda tähendab selle lisamine umbes 4€ lisa väljaminekut ja lisatööd programmeerimisel. Programmeerimise keerukus sõltub iga programmeerija enda kogemustest ning võimest lisainformatsiooni otsida, mistõttu anti plaatide võrdlemisel selles kategoorias subjektiivne hinnang. Varasematest kokkupuudetest lähtuvana osutus ka selles kategoorias paremaks Arduino platvorm. Oluline on arvestada ka sellega, et plaadid vajavad töötamiseks ka toiteallikat. Kummalgi plaadil ei ole kaasas toiteplokki, seega on vaja need eraldi juurde osta. Toiteplokkide hinnad on suhteliselt sarnased, mistõttu jäi otsustavaks aspektiks

see, et Arduino suudab töötada väga suures toitepingevahemikus 6-20 V. Väiksemat plaati on lihtsam paigutada, seega osutus see selles kategoorias paremaks Raspberry Pi, mis on laiuselt suhteliselt sarnase mõõduga, kuid pikkuse poolest umbes 35% lühem. Raspberry Pi 100 korda kiirem protsessori tuum suudab programmi palju kiiremini töödelda kui Arduino, seega on Pi parem valik. Teisalt on tegevusi vähe ja need ei ole aegkriitilised, mistõttu on piisavalt kiirust ka Arduino 16 Mhz taktsagedusega protsessoril. HDMI porti olemasolu võimaldab väga lihtsalt ühendada ekraani süsteemi külge, seetõttu on see hea lisa Raspberry-l. Võrdlustabelist selgus, et parem kontroller käesoleva projekti jaoks on Arduino Mega 2560.

Tabel 2.1 Arduino Mega 2560 ja Raspberry Pi 3A+ võrdlustabel

	Arduino	Raspberry Pi	Tähtsus
I/O arv	56	40	5
Hind	37,5 €	30 €	5
Analoog sisend	16	0	4
Programmeerimis lihtsus	Lihtsam	Keerulisem	4
Toide	USB / 7-12 V	USB	3
Maksimaalne vool väljundist (mA)	20	16	3
Plaadi mõõtmed (mm)	102x53	65x56	2
Protsessori kiirus	16 MHz	4x1,6 GHz	1
HDMI port	puudub	olemas	1

2.3 Manussüsteemid

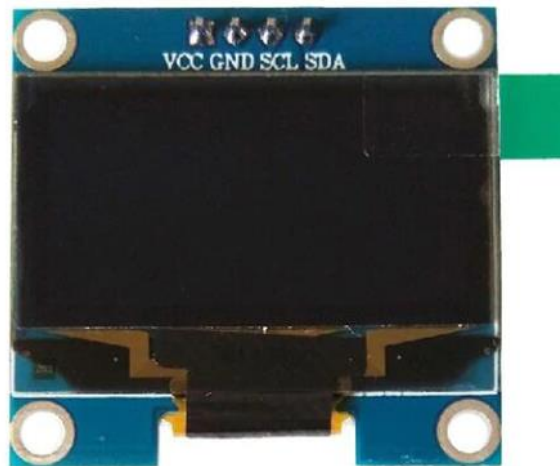
Süsteemi ehitamiseks valitud kontrollerial puudusid mitmed osad, mida on vaja raamide juhtimiseks kasutada. Nende puuduste kõrvaldamiseks kasutati erinevaid manussüsteeme, mis telliti e-poest Aliexpress.com. Sealt tellides olid need palju soodsamad, kui Eestist või mujalt Euroopast ostes. Valikus oli palju mooduleid, mis täitsid samu ülesandeid ja olid sarnase välimusega. Neist sobivaima välja valimise peamiseks kriteeriumiks oli leida just selline mudel, millele oleks valmis kirjutatud Arduino Mega peal toimiv teek. Infot manussüsteemide teekide kohta saadi peamiselt GitHub ja Arduino foorumitest.

Esimeseks ja kõige tähtsamaks mooduliks oli RTC moodul (joonis 2.3) mille abil saab keeramist ajastada. Välja valiti DS3231 moodul mis kasutab AT24C32 kiipi. Kellaaja õigena hoidmiseks, ka siis kui väline toide välja lülitada, on vaja seadmesse ühendada CR2032 tüüpi patarei, mille väljundpinge on 3 V. Arduino külge ühendades saab kasutada nii 3.3 V kui ka 5 V toitepinget. Programmeerimise lihtsustamiseks kasutati „DS3231“ teeki, mille on koostanud A. Wickert, E. Ayars, J-C. Wippler, N. Widget.



Joonis 2.3 RTC moodul [14]

Järgnevas oli vaja ekraani, et kuvada seadme hetke kellaaega ja vajadusel ka veateateid. Välja valiti SH1106 OLED ekraani (joonis 2.4). Tegu on 1,3 tollise ekraaniga, resolutsiooniga 128*64 pikslit. Ekraan saab toite Arduino 5 V pesast ja suhtlemiseks kasutatakse I²C protokoll. Kuna kuvatavat informatsiooni on vähe ning tekstid on lühikesed, siis ei ole tarvis suuremat ekraani ega paremat resolutsiooni. Programmeerimise lihtsustamiseks kasutati „U8glib“ teeki, mille on koostanud „Olikraus“.



Joonis 2.4 1,3 tolli OLED ekraan [15]

Viimaseks kasutatud mooduliks oli anemomeeter (joonis 2.5), mille abil saab süsteem teavet tuule kiiruse kohta. Välja valiti selline mudel, mis väljastab teavet 0-5 V signaalina. Selline signaali tüüp võimaldab teavet saada ilma ühegi spetsiifilise andmeedastusprotokollita. Tänu sellele pole vaja kasutada ühtegi teeki, mis omakord lihtsustab programmeerimist.



Joonis 2.5 Anemomeeter [16]

2.4 Elektriskeemi ja selle ülesehitus

Elektriskeem koosneb neljast osast. Esimeses osas asuvad kontrolleri koos RTC ja OLED mooduliga ning 12 V toiteplokk. Teiseks osaks on sisendite skeem kus on ära märgitud andurid ja neile vajalikud osad, et kontrolleri külge ühenduda. Kolmandaks osaks on väljundite skeem kus on välja toodud raamide juhtimiseks vajaminevad komponendid ja nende ühendused. Neljandaks osaks on käsitsi juhtimise osa, mille abil on võimalik päikesepaneeli juhtida ka kontrolleri rikke korral.

2.4.1 Elektriskeemi juhtosa

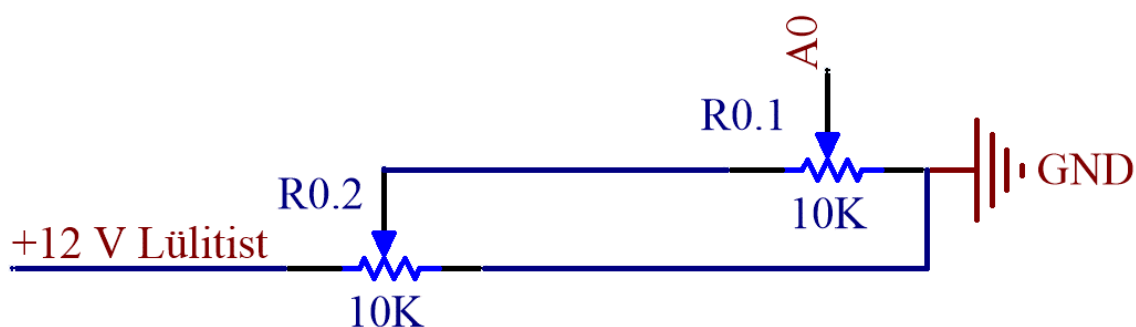
RTC ja OLED mooduli ühendamine Arduino külge oli väga lihtne tänu sellele, et moodulitele portide juurde oli märgitud millise Arduino pordiga see ühendada tuleb. Mõlemad moodulid kasutavad kommunikatsiooniks I2C protokollit ja 5 V toitepinget, seega saavad need jagada ühiseid Arduino porte. Nii RTC kui ka OLED moodulil oli vaja ühendada ainult neli klemmi.

12 V allikana kasutati impulss toiteplokk, mille maksimaalne väljundvool on 2 A. Sellest toiteplokkist saavad toitepinge nii Arduino, mis ühendati otse toiteploki külge, kui ka andurid, mille ühendused tehti läbi juhtpinge relee. Juhtosa elektriskeem – lisa 2.

2.4.2 Kontrolleri sisendid

Kontrolleril kasutati 14 sisendit, millest 11 olid digitaal sisendid ja 3 analoog sisendid. Kuute digitaal sisendit kasutati induktiivandurite signaali saamiseks - üks sisend oli vajalik nivoo anduri jaoks. Nivooandur ja induktiivandurid väljastavad 12 V signaali, seega oli tarvis need kontrolleri külge ühendada läbi relee, mille abil muudeti 12 V signaal 5 V signaaliks. Ülejäänud nelja digitaal sisendit kasutati veakoodide nullimiseks ning tuule kiiruse kontrollimise ja kahe raami hooldusrežiimi sisse ja välja lülitamiseks. Antud sisendite puhul kasutati ümberlülitavaid lüliteid. Lülitite ühised otsad ühendati Arduino sisendite külge. Lülitite teised kaks otsa ühendati vastavalt Arduino 5 V pesasse ja maandusesse. Ümberlülitavaid lüliteid kasutati seetõttu, et kui sisend on lülitatud madalasse olekusse, siis ei teki signaali ujumist.

Ühte analoog sisendit kasutati anemomeetri signaali lugemiseks. Nagu varasemalt mainitud, väljastab see 0-5 V signaali – seega saab selle otse Arduino analoog sisendisse ühendada. Päikesepaneelide suuna teada saamiseks kasutati samuti analoog sisendeid. Varasemalt mainitud raamide keskele paigaldatud potentsiomeetrilt saadakse pinget 0-5 V ning selle alusel teatakse päikesepaneelide suunda. Raamid keeravad ligikaudu 150°, kuid lähim potentsiomeeter sellele keerab 180°. Välja mõeldi lahendus, mille abil on võimalik 180° keeraval potentsiomeetril saada pinget 0-5 V vahemikus 0° kuni ligikaudu 150°. Selleks kasutati joonis 2.6 näidatud skeemi. Pinge reguleerimise potentsiomeetri abil on võimalik reguleerida paika asendi potentsiomeetril olev pinget. Selleks keerati esmalt päikesepaneelid lõpu asendisse. Seejärel vaadeldi Arduino sisendil olevat pinget ja reguleeriti see 5 V peale. Selliselt oli võimalik pinge mõõtmise resolutsioonis kaotamata kasutada suurema pöördenurgaga potentsiomeetrit. Andurite elektriskeem – lisa 3.



Joonis 2.6 Keeramise asendi potentsiomeetri ühendus, kus: „+12 V lülitist“ – toitepinge välisest allikast, „R0.2“ – pinge reguleerimise potentsiomeeter, „R0.2“ – päikesepaneelide asendi potentsiomeeter, „A0“ – Arduino analoog sisend, „GND“ Arduino ja välise toiteallika kokku ühendatud maandus.

2.4.3 Kontrolleri väljundid

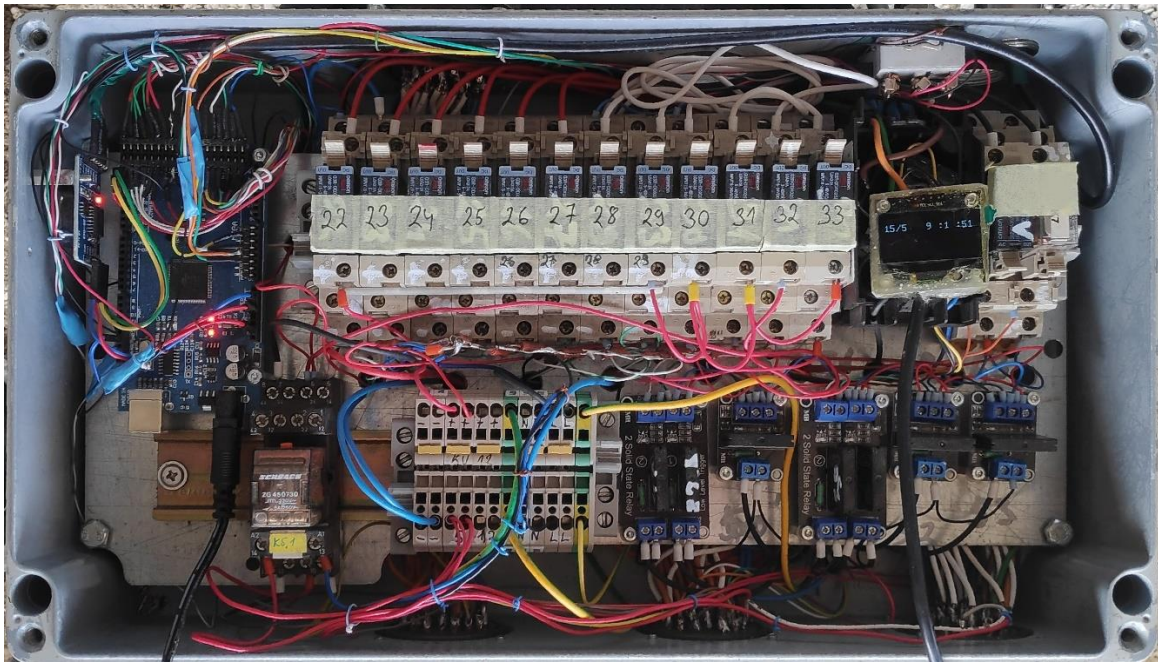
Kontrolleril kasutati kümmet väljundit, millest kaheksat hüdraulika suunaventiilide juhtimiseks, ühte hüdropumba mootori juhtimiseks ja ühte juhtpinge juhtimiseks. Kõik väljundite poolt juhitud seadmed toimivad 230 V vahelduvpinge mõjul, seega ühendati need läbi rele. Suunaventiili tööasendis hoidmiseks lastakse pinge vastavale suunaventiili mähisele – selle tagastamine algasendisse toimub aga vedru jõul. Hüdropumba mootori käivitina kasutatakse varasemalt mootori külge ühendatud käivitit. Mootori käivitamiseks antakse pinge käiviti juhtklemmidele ja selle peatamiseks katkestatakse pinge. Juhtpinge ülesanne on läbi rele anda toitepinge teistele releedele ja anduritele. Ühest küljest aitab see kaasa ohutusele, teisalt aga aitab ka energiat säästa, lülitades välja andurid sellel ajal kui nende signaale ei loeta. Täiturite elektriskeem – lisa 4.

2.4.4 Käsijuhtimine

Käsijuhtimine koosneb kokku kuuest lülitist, mille abil on võimalik raame eraldi pöörata, ning ülesse ja alla lasta. Mõlema raami jaoks on üks surunupp ja kaks kolmeasendilist mittefikseeruvat lülitit. Surunupu abil lülitatakse signaalpinge hüdropumba mootori käivitile ja kahele ülejäänud lülitile. Viimastest ühega on võimalik raami keerata ning teisega üles ja alla lasta. Käsijuhtimise elektriskeem – lisa 5.

2.5 Elektriahela valmimine

Nii kontroller koos RTC mooduliga kui ka sisendite ja väljundite releed koondati ühte metallkarpi (joonis 2.7). Sellise karbi loomisel oli kaks peamist eesmärki. Esimeseks eesmärgiks oli kaitsta selles olevaid komponente ilmastikuolude eest ja teiseks eesmärgiks oli luua lahendus, mille abil oleks võimalik rikke korral veaotsingut teostada tubastes tingimustes. Eriti kasulik on selline lahendus kui rike peaks tekkima sajuse ilma korral. Lahenduse teostamiseks kasutati karbi külgedel olevaid viite pistikut. Antud pistikute lahti ja kokku ühendamine on väga kiire ja lihtne, tuleb vaid lukusti keerata õiges suunas. Kahte pistikut kasutati raamide ühendamiseks ja kahte maketi ühendamiseks. Elektroonika karbi kinnitus päikesepaneelide juures tehti kiirkinnitustega, et seda oleks võimalikult kerge lahti ühendada.



Joonis 2.7 Elektrikarp avatult

2.5.1 Ekraani installeerimine

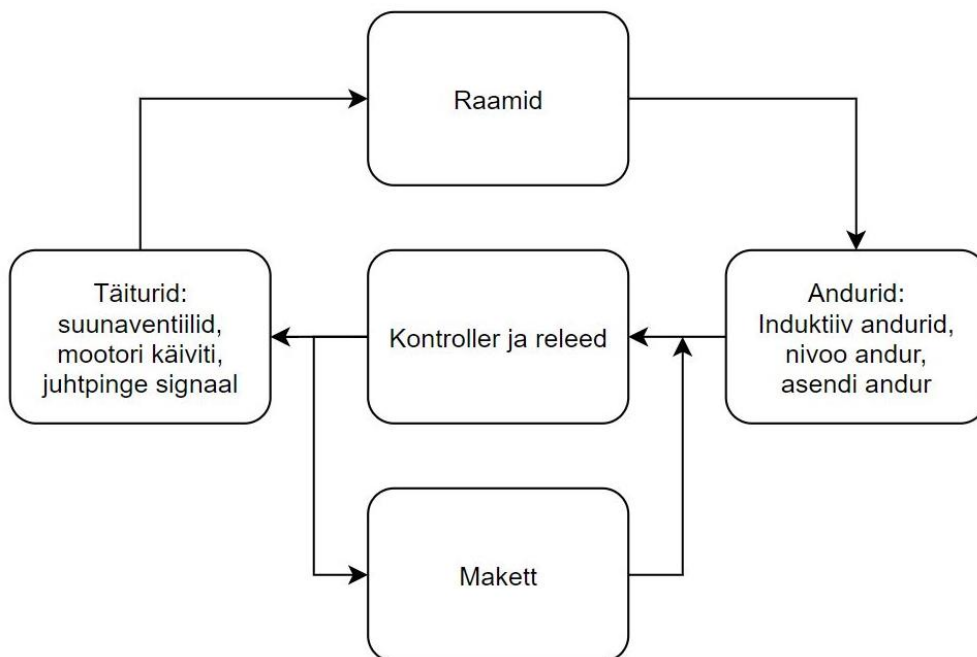
Selleks, et ekraanilt oleks võimalik välja lugeda süsteemi kellaega ja vajadusel ka veakoode, installeeriti see elektrikarbist eemale, nähtavale kohale. Moodul otsustati kinnitada umbkaudu silmade kõrgusele seina külge. Ühendamise jaoks vajalike kontaktide elektroonika karbist välja viimise tarbeks kasutati viiest karbi küljest olnud pistikust väikseimat. Kuigi selle pistikul oli seitse kontakti siis OLED ekraani ühendamiseks kasutati neist ainult nelja ning ülejäänud kolm jäid vabaks. Ilmastikukindluse tagamiseks valati moodul läbipaistva epoksiidliimi sisse.

2.5.2 Raamide makett

Programmikoodi kirjutamise lihtsustamiseks ning selle testimiseks ja hiljem vajadusel tõrkeotsingu teostamiseks koostati raamide makett (joonis 2.8). Maketi saab ühendada siis kui raamid ei ole juhtkarbi külge ühendatud ja ka siis kui raamid on ühendatud ning süsteem töötab automaatselt. Maketil väljendab iga leed ühte täiturit: suunaventili mähist, mootori käivitit või juhtpinge signaali. Kui täitur on aktiveeritud, läheb leed maketil põlema. Induktiiv, nivoo ja raami asendi andurite väljendamiseks kasutati maketil lüliteid ja potentsiomeetreid, mille abil on võimalik signaale emuleerida. Kõik leedid ja lülitid ühendatakse vastava täituri või anduriga rööpühendusse, või ainult maketi ühendamise korral nende asemele. Maketi ühendamist selgitav plokkskeem joonis 2.9.



Joonis 2.8 Suletud elektrikarp (vasakul) ja makett (paremal)



Joonis 2.9 Süsteemi plokk skeem

2.5.3 Hinnanguline aja- ja rahakulu elektriahela valmimiseks

Kogu elektriahela valmimise protsess koos programmi kirjutamisega võttis aega ligikaudu 50 tundi. Aega arvestatakse koos programmeerimisega kuna need toimusid suurel määral paralleelselt. Ligikaudu 70% antud ajast kulus programmi kirjutamisele ja selle testimisele.

Elektriahela valmimisel sai võimalikult palju kasutatud juba olemas olnud komponente – induktiivandurid, takistid, potentsiomeetrid, releed, juhtmed ja kaablid. Juurte ostetud osade kogumaksumus oli umbkaudu 150 €. Suuremad kuluartiklid olid Arduino Mega 2560 ja anemomeeter. Muuhulgas kulus raha teistele manussüsteemidele, mõnele releele ja induktiivandurile.

Ostes kõik komponendid kujuneb elektriahela hinnaks ligikaudu 250-300€. Hinna sisse on ka arvestatud maketi maksumus. Väikeste kokkudevõetavate eesmärgil ahela koostamisel on võimalik näiteks välja vahetada sisendite releed, mis muudavad 12V anduri signaali 5V signaaliks, takistitega tehtavate pingejagurite vastu.

3. Programmi koostamine ja tööpõhimõte

Päikesepaneelide keeramiseks vajaliku programmi koostamiseks Arduinole kasutati vabavaralist programmi „Arduino IDE“ (v 1.8.5). Kõik kasutatud teegid on vabavaralised ja allalaetavad otse Arduino IDE teekide haldamise keskkonnast. Programmi põhimõtteline plokk skeem – lisa 6. Programmi kood – lisa 7.

3.1 Manussüsteemid

Programmeerimist alustati manussüsteemide katsetamistega teekide autorite poolt valmistatud programmide näidiste abil. Esmalt oli vaja õigeks saada RTC mooduli kella aeg, mille seadmiseks oli olemas valmis kood. Programmi kompileerimisel lisati kompileeriva seadme poolt hetke kella aeg programmi ning see edastati Arduinosse. Programmi käivitamisel andis Arduino käsu RTC moodulile salvestada hetke kella ajaks programmi kirjutatud kella aeg. Edasi hakkas Arduino täitma põhitsükli, milleks oli kellaaja küsimine RTC-ilt ning seejärel kellaaja kirjutamine *serial monitor*-i. Programmi edastamine ja selle käivitamine ei toimunud hetkeliselt, seega tekkis umbes 5 sekundiline vahe tegeliku ja RTC mooduli aja vahel. Tekkinud 5 sekundiline erinevus ei oma mitte mingit tähtsust päikesepaneelide keeramisel. Järgnevalt katsetati OLED ekraani näidisprogrammi abil, mille sisuks oli erinevate kujundite kuvamine ekraanil.

3.2 Kellaaja lugemine ja kuvamine

Kui manussüsteemid olid üksikult tööle saadud oli järgmiseks eesmärgiks saada koos tööle RTC ja OLED moodulid. Selleks oli vaja uurida näidiskoodi põhjalikumalt, et välja selgitada mida iga osa koodist teeb ning millised osad võib vaja olla päikesepaneelide keeramiseks. Järgnevalt püstitatud ülesandeks oli kirjutada kood, millega saab Arduino kellaaja RTC-ilt ja kuvab selle OLED moodulil. Kuigi kellaaja saamine oli teegi abi tehtud väga lihtsaks, siis selle kuvamine oli palju keerulisem, tulenevalt sellest, et kellaaja lugemisel ja kasutamisel kasutatakse erinevat tüüpi muutujaid. Abi probleemi lahendamiseks saadi Arduino foorumi erinevatest postitustest. Kui püstitatud vaheülesanne oli täidetud, ning moodulid koos tööle saadud, siis otsustati teha katse kellaaja täpsuse püsivuse kohta. Katse läbiviimiseks seati taas paika kell ning seejärel laeti Arduinole peale kood millega katsetati RTC ja OLED ekraani koos töötamist. Järgmiseks eemaldati Arduinolt toide. Toitejuhe ühendati taas ligikaudu nädala aja möödudes, et võrrelda RTC mooduli ja arvutil olevat kella. Nii nagu katse alguses, oli ka katse lõpus umbes 5 sekundiline staatiline viga, mille tekkimisest on kirjutatud varasemalt. Nädal aega võib olla liiga lühike aeg täpsete tulemuste saamiseks, mistõttu tehti uus aegade võrdlus ka kuu aja möödudes ja saadi samasugused tulemused. Sellest võib järeldada, et RTC mooduli ajast maha jäämine või ette liikumine on peaaegu

olematu ja seade vastab eelpool mainitud +/-1 sekundit nädalas kriteeriumile. Kella ja ekraani katsetamiseks loodud koodi sai hõlpsasti muuta funktsiooniks „int KELL“ mida kasutati paneelide keeramise programmis. „Int“ tüüpi funktsiooni kasutati seetõttu, et see võimaldab tagastada funktsioonist andmeid ning selle funktsiooni puhul tagastatakse põhiprogrammi kellaaja tundide arv.

3.3 Häirefunktsioon

Järgnevas kirjutati valmis süsteemi häirefunktsioon. Funktsiooni nimeks sai „STOPP“ ja tüübiks „void“ ning lisaks antakse kaasa üks „int“ tüüpi muutuja. Funktsiooni ülesandeks on peatada kogu süsteemi töö. Esmalt lülitatakse välja juhtpinge mis on piisav, et peatada kogu süsteem. Lisa ohutuse tagamiseks viiakse algolekusse ka ülejäänud kasutusel olevad väljundid: hüdropump ja hüdrocilindrite suunaventiidid. Kaasa antud muutuja järgi tehakse kindlaks millist veakoodi on vaja kuvada. Järgnevalt kuvatakse kellaega ja vea tekkepõhjust niikaua, kuni veasignaali tühistamise nuppu on all hoitud vähemalt 1 sekund. Kui veasignaal on tühistatud jätkab seade tööd.

3.4 Juhtpinge ja hüdropumba sisse lülitamine

Raamide keeramiseks on vaja sisse lülitada juhtpinge ja hüdropump. Selle tarbeks koostati „void“ tüüpi funktsioon nimega „JUHTPINGE“. Antud funktsiooni toimib järgnevalt: esmalt lülitatakse sisse juhtpinge signaal ning seejärel tehakse 1 sekundiline viide, mis tagab selle, et kõik lülitused oleks tehtud. Järgmiseks kontrollitakse õlitaseme olemasolu ja kui see on olemas, siis lülitatakse järgi hüdropump ning tehakse uus 1 sekundiline viide, et hüdropump jõuaks käivituda. Õlitaseme puudumisel läheb süsteem häire režiimi ning ekraanile kuvatakse vea tekkepõhjusena kiri „NIVOO“.

3.5 Õlitaseme kontroll

Raamide keeramine toimub hüdraulikaga, mistõttu on oluline tagada, et õli tase hüdropaagis ei langeks liiga madalale ja seetõttu ei satuks õhku hüdrostsüsteemi. Selle tarbeks koostati „void“ tüüpi funktsioon nimega „OLITASE“, mille abil on võimalik keeramise ajal kontrollida õli taseme olemasolu. Funktsiooni väljakutsumisel kontrollitakse koheselt nivoo anduri signaali olemasolu. Juhul kui signaal on olemas väljutakse funktsioonist koheselt. Juhul kui signaal puudub, ehk vastav sisend on madalas olekus, tehakse ühe sekundiline viivitus ja seejärel kontrollitakse uuesti signaali. Kui teisel kontrollimisel on signaal olemas, siis oli arvatavasti tegu häiringuga ning seade saab tööd jätkata. Signaali puudumisel teisel kontrollil lõpetatakse keeramine ning süsteem läheb häire režiimi.

3.6 Tuule tugevuse kontrollimine

Päikesepaneelide rohke arvu tõttu raamid on vajadus arvestada ka tuule mõjuga. Selle tarbeks koostati „void“ tüüpi funktsioon nimega „TUUL“, mille ülesanne on tugeva tuule korral päikesepaneelide alumisse ehk horisontaalsesse asendisse viimine. Selle funktsiooni välja kutsumisel tehakse esmalt kontroll, kas tuule kontrollimine on sisse lülitatud. Tuule kontrollimist on võimalik käsitsi välja lülitada kahel põhjusel, esiteks selleks, et tagada ohutus raamide hooldamise ajal. Teiseks selleks, et anemomeetri rikke korral oleks võimalik seadme tööd jätkata. Kui tuule kontrollimine on välja lülitatud, siis väljutakse funktsioonist ja seadme töö jätkub. Sisselülitatud kontrolli korral loetakse anemomeetrilt saadav pinge ja võrreldakse kas see ületab lubatavat pinget. Kui pinge on lubatust pingest väiksem, siis väljutakse funktsioonist. Juhul kui pinge on suurem kui lubatud pinge, siis tehakse ühe sekundiliste vahedega kuus uut kontrolli. Ka siis, kui mõnel järgnenud mõõtmise korral saadakse lubatust väiksem pinge, väljutakse funktsioonist. Kokku tehakse seitse mõõtmist, et välistada häiringute ja lühiajaliste tuuleilide mõju. Juhul kui kõik mõõtmised ületavad lubatud pinget siis viiakse päikesepaneelid alumisse asendisse. Mõlema raami paneelid langetatakse samaaegselt, ühteaegu kontrollides, kas kumbki paneel on alla jõudnud. Kui üks raamidest jõuab alumisse asendisse lülitatakse selle hüdrocilindri klapp välja, ning langetatakse teist raami edasi niikaua kuni ka see on alumises asendis. Seejärel läheb süsteem häire režiimi, kus kuvatakse veateade „TUUL“.

3.7 Raamide keeramine

Päikesepaneelide keeramine toimub funktsioonis „KEERAMINE“. Antud funktsioon on „void“ tüüpi. Selles funktsioonis toimub nii päikesepaneelide edasi ehk päripäeva kui ka tagasi ehk vastupäeva keeramine vastavalt hetke kellaajale. Esmalt kutsutakse välja funktsioon „KELL“, mille abil saadakse teada praegune kella tundide arv. Saadud kellaega võrreldakse eelmisel korral saadud kellaajaga, ning kui kellaag ei ole muutunud siis väljutakse funktsioonist ja päikesepaneele ei keerata. Juhul kui kellaag on muutunud, siis toimub päikesepaneelide keeramine, mis on kellaegade järgi jagatud kolme ossa. Esimeses osas keeratakse paneele vastavalt potentsiomeetrilt saadavale informatsioonile. Teises osas toimub päikesepaneelide lõppu keeramine ning lõpus olemise kohta saadakse informatsiooni lõpu induktiivandurilt. Kolmandas osas toimub päikesepaneelide algusesse tagasi keeramine, ning alguses olemise kohta saadakse informatsiooni alguse induktiivandurilt.

3.7.1 Keeramine potentsiomeetri järgi

Potentsiomeetrilt saadava pinge järgi toimub päikesepaneelide keeramine sellisel juhul kui kellaaeg jääb vahemikku 9-17. Esmalt leitakse kahe massiivi abil kuhu maani on vaja päikesepaneeli keerata. Esimeses massiivis on loetletud üheksa kellaaega („9“; „10“; „11“; ... „16“; „17“). Selle abil leitakse mitmes massiivi element on praegune kellaaeg. Teises massiivis on loetletud üheksa pinge väärtust („0,5“; „1“; „1,5“; ... „4“; „4,5“). Sellest võetakse nii mitmes element, kui mitmes element esimesest massiivist on praegune kellaaeg. Teisest massiivist saadud pinge väärtus on see väärtus, mille saamiseni potentsiomeetrilt tuleb päikesepaneeli keerata. Kui pinge väärtus on leitud, siis hakatakse päikesepaneeli ühekaupa keerama. Enne esimese raami keeramist kontrollitakse, et see ei ole alumises asendis või hoolduses. Kui vähemalt üks neist tingimustest on täidetud, siis ei hakata seda paneeli keerama, ning liigutakse edasi teise raami keeramise juurde. Juhul kui esimene raam ei ole hoolduses ega allalastud asendis, siis alustatakse raami keeramist. Esmalt kutsutakse välja funktsioon „JUHTPINGE“, mille abil lülitatakse sisse seadme juhtpinge ja hüdropump. Järgmiseks käivitatakse stopper, mille abil kontrollitakse, et päikesepaneelide keeramine ei oleks liiga kaua kestnud. Raami edasi keeramiseks lülitatakse sisse esimese raami keeramissilindri edasi liigutamise suunaventiil. Seejärel algab tsükkel, milles kutsutakse välja funktsioon „OLITASE“, et kontrollida õli taseme olemasolu mahutis, seejärel kontrollitakse potentsiomeetrilt saadud pinget ja viimaseks kontrollitakse stopperi käivitamisest möödunud aega. Tsüklist väljutakse kahel juhul, esiteks saavutatakse potentsiomeetril soovitud pinge ning sellisel juhul liigutakse edasi teise raami keeramise juurde, või teiseks minnakse õlitaseme puuduse või keeramise liiga kaua kestmise tõttu häire režiimi. Teise raami keeramine toimub identselt esimese raami keeramisele.

3.7.2 Keeramine lõpu induktiivini

Päikesepaneelide lõpu keeramine toimub siis kui kellaaeg on 18. See keeramise osa algab samamoodi nagu esimene keeramise osa. Esmalt kontrollitakse paneeli keeramise vajalikust, seejärel lülitatakse järgi juhtpinge ja hüdropump, käivitatakse stopper ning lülitatakse sisse ka raami edasi liigutamise suunaventiil. Edasi kontrollitakse kordamööda õli taset ja lõpu induktiivanduri signaali. Keeramine lõpetatakse siis kui saadakse lõpu induktiivandurilt signaal, või kui seade läheb häire režiimi. Induktiivandurilt signaali saamisel hoitakse suunaventiili veel viis sekundit sisselülitatult – seejärel lõpetatakse keeramise tsükkel. Selle eesmärk on veenduda, et raam on surutud vastu varem mainitud kummipatja, et vältida kõikumist.

3.7.3 Keeramine alguse induktiivini

Päikesepaneelide algusesse keeramine toimub siis kui kellaaeg on 23. See keeramine toimub sarnaselt lõpu keeramisega, kuid erineb selle poolest, et keeramise silindril lülitatakse sisse tagasi

liigutamise suunaventiil. Lisaks kontrollitakse lõpu induktiivanduri asemel alguse induktiivanduri signaali.

Nii nagu teiste keeramis osade lõpus on ka selle osa lõpus hüdropumba väljalülitamine ning peale ühe sekundilist viidet juhtpinge välja lülitamine. Lisaks salvestatakse praegune kella-aeg eelmiseks kellaajaks, et järgmine kord funktsiooni välja kutsudes oleks võimalik neid omavahel võrrelda.

KOKKUVÕTE

Töö esimeses peatükis anti ülevaade raami projekteerimisest, tugevusarvutustest, hüdrooskeemi koostamisest ja valmis ehitamisest. Raamide projekteerimisel selgus, et 52 päikesepaneeli jaoks on optimaalseim ehitada kaks raami. Projekteerimise lõpus tehtud tugevusarvutustest lähtuvalt võib öelda, et ka sisemaa suuremad tuulepuhangud ei tohiks ohustada raamide konstruktsiooni, sest need kannatavad tuule tugevust kuni 35 m/s. Sellest hoolimata lastakse paneelid ohutuse tagamiseks alumisse asendisse, kui püsiv tuule kiirus ületab kiirust 20 m/s. Hüdraulikaskeem koostati nii, et oleks võimalik elektrikatkestuse ajal käsitsi päikesepaneeli alumisse asendisse lasta. Ehitusprotsessi käigus leiti sobivaimad asukohad anduritele, mille abil saadakse tagasisides raamide kohta.

Teises peatükis käsitleti kriteeriumeid, mida pidid kontrolleri ja programmi suutma täita, kontrolleri valikut, manussüsteeme, elektriskeemi ülesehitust ja elektriahela valmimist. Kriteeriumide paika panemisel tehti koostööd lõputöö välja pakkujaga, kes avaldas enda poolsed kriteeriumid süsteemile. Kontrolleri valikul osutus väljavalituks Arduino Mega 2560 tänu oma kasutamiskihtsusele, sisendite suurele arvule ja ADC olemasolule. Manussüsteemisest kasutati RTC moodulit, OLED ekraani ja anemomeetrit. Nende väljavalimisel lähtuti programmeerimise kihtsusest, näiteks teekide olemasolust. Elektriskeem jagati neljaks osaks – kontrolleri, kontrolleri sisendid, kontrolleri väljundid ja käsijuhtimine. Elektriskeemi valmimise ajal otsustati ehitada ka makett, mille abil oli võimalik programmi katsetada ja mis tulevikus vajadusel kihtsustab tõrke põhjuse otsimist.

Kolmandas peatükis anti ülevaade erinevatest programmi osadest ja nende töö põhimõtetest. Programmi kirjutamise ajal toimus töö ka raamide ehitamise kallal. Sellest lähtuvalt käis programmi kirjutamine ja katsetamine peamiselt maketi abil. Viimased katsetused ja programmi parandused viidi läbi raamide endiga katsetades.

Töö kirjutamise hetkeks on süsteem töötnud üle nädala aja ilma tõrgeteta. Sellest lähtuvalt võib öelda, et peamine püstitatud ülesanne sai täidetud. Lisaks püstitati töö alguses küsimus, kas ise ehitatud hüdraulikaga süsteem on soodsam, kui valmis lahenduse ostmine. Antud küsimusele on võimalik välja tuua kaks vastust. Esimene vastus lähtuks konkreetse süsteemi loomisest, mille käigus kasutati ära võimalikult palju kasutuna seisvat materjali. Teine vastus lähtuks olukorrast, kus kõik materjalid tuleb osa. Valmis raami lahenduse maksumuseks arvestati ligikaudu 7500 €. Arvestades kokku esimese ja teise peatüki lõpus välja toodud hinnad, saadakse käesoleva projekti maksumuseks 3550 €. Ostes kõik komponendi kujuneb maksumuseks ligikaudu 6250 – 6800 €, lähtuvalt komponentide valikust.

Töö edasiarendamise võimaluseks on panna päikesepaneelid automaatselt ka teises teljes keerama, mida kindlasti tulevikus tehakse. Lisaks on võimalik luua erinevad keeramise profiilid erinevateks aastaegadeks. Päikeseelektrijaama laiendamisel on võimalik süsteemi lihtsa vaevaga integreerida veel pööravaid raame.

SUMMARY

In the first chapter of thesis an overview of the frame design, statics calculations, development and compilation of the hydraulic circuit was given. While designing the frames, it turned out that it is optimal to build two frames for 52 solar panels. Based on the strength calculations made at the end of the design process, it can be said that even the largest inland wind gusts should not endanger the structure of the frames, as they can withstand wind speeds of up to 35 m/s. Nevertheless, for safety reasons, the panels are lowered if the wind speed exceeds 20 m/s. The hydraulic scheme was designed so that the solar panels could be lowered manually during a power outage. During the construction process, most suitable locations for the sensors, which provide feedback from the frames, were found.

In the second chapter an overview of criteria, that the controller and the program had to be able to meet, the choice of the controller, the embedded systems, the structure of the electrical scheme and the completion of the electrical circuit was given. The criteria were set in collaboration with the provider of the thesis. The Arduino Mega 2560 proved to be the controller of choice for its ease of use, large number of pins and ADC. The embedded system used where an RTC module, an OLED display and an anemometer. They were selected based on ease of programming, for example availability of libraries. The wiring diagram was divided into four parts - controller, controller inputs, controller outputs and manual control. During wiring, it was decided to build a model, which could be used to test the program and, if necessary, will make it easier to troubleshoot the system in the future.

The third chapter gave an overview of different parts of the program and their working principles. The frames were being built at the same time as the program was written, thus the program was written and tested mainly on the model. The last tests and program improvements were performed while testing on the real frames.

At the time of writing, the system has been running for more than a week without any problems. Based on that, it can be said that the main task was fulfilled. In addition, at the beginning of the thesis, the question was raised, whether it is cheaper to build a system with hydraulics or to buy a ready-made solution. There are two possible answer this question. The first answer would be about the specific system, that used as much useless material as possible. The second answer would be based on a situation where all the materials must be purchased. The cost of the ready-made frame solution was estimated at approximately 7500 €. When adding up the prices presented at the end of the first and the second chapter, the cost of this project turns out to be 3550 €. When buying all

the components, the cost will be approximately 6250 - 6800 €, depending on the choice of components.

One of the possibilities for further development is to automate second axis tracking, which will be done in the future. In addition, it is possible to create different turning profiles for different seasons. When expanding the solar power plant, it is easily possible to integrate more rotating frames into the system.

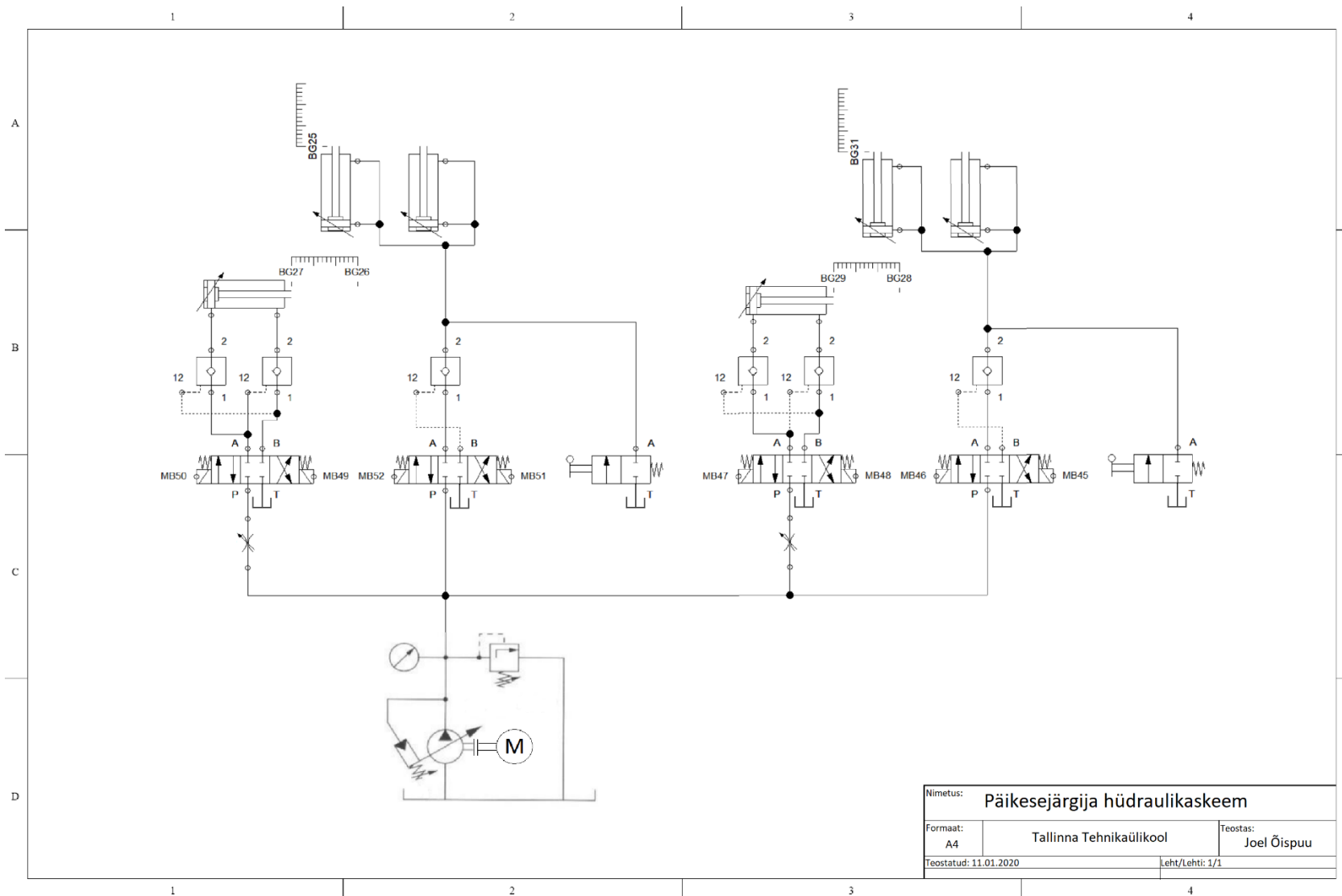
KASUTATUD KIRJANDUS

- [1] „Päikese jälgimissüsteemid,“ TipSolar, [Võrgumaterjal]. Saadaval: <http://www.tipsolar.ee/18-tooted-paigaldussusteemid-paikese-jalgimissusteemid.html>. [Kasutatud 12. 05. 2020].
- [2] Trinasolar, „Splitmax TSM-PE05H,“ [Võrgumaterjal]. Saadaval: https://static.trinasolar.com/sites/default/files/EN_TSM_PE05H_datasheet_B_2017.pdf. [Kasutatud 05. 05. 2020].
- [3] „Wind load on surface,“ The Engineering Toolbox, [Võrgumaterjal]. Saadaval: https://www.engineeringtoolbox.com/wind-load-d_1775.html. [Kasutatud 12. 05. 2020].
- [4] „Ainete tiheduste tabel,“ Eesti Füüsika Selts, [Võrgumaterjal]. Saadaval: <https://opik.fyysika.ee/index.php/book/section/13830#/section/13830>. [Kasutatud 09. 05. 2020].
- [5] P. Kulu, M. Saarna, R. Tarbe, J. Kers ja R. Veinthal, „Materjaliõpetuse praktikumide ja kodutööde juhendid,“ TTÜ kirjastus, Tallinn, 2010.
- [6] „Material Properties of S355 Steel,“ Meadinfo, [Võrgumaterjal]. Saadaval: <https://www.meadinfo.org/2015/08/s355-steel-properties.html>. [Kasutatud 12. 05. 2020].
- [7] „Hüdraulika teoreetilised alused,“ Tallinna Tööstushariduskeskus, [Võrgumaterjal]. Saadaval: <http://www.ene.ttu.ee/leonardo/hydro/hydro02.pdf>. [Kasutatud 09. 05. 2020].
- [8] „What is Arduino?,“ Arduino, [Võrgumaterjal]. Saadaval: <https://www.arduino.cc/en/Guide/Introduction>. [Kasutatud 05. 05. 2020].
- [9] „Arduino Mega 2560,“ Arduino, [Võrgumaterjal]. Saadaval: <https://store.arduino.cc/arduino-mega-2560-rev3>. [Kasutatud 04. 05. 2020].
- [10] „About Raspberry Pi,“ Raspberry Pi, [Võrgumaterjal]. Saadaval: <https://www.raspberrypi.org/about/>. [Kasutatud 05. 05. 2020].
- [11] „Raspberry Pi 3 A+,“ Raspberry Pi, [Võrgumaterjal]. Saadaval: <https://www.raspberrypi.org/products/raspberry-pi-3-model-a-plus/>. [Kasutatud 05. 05. 2020].

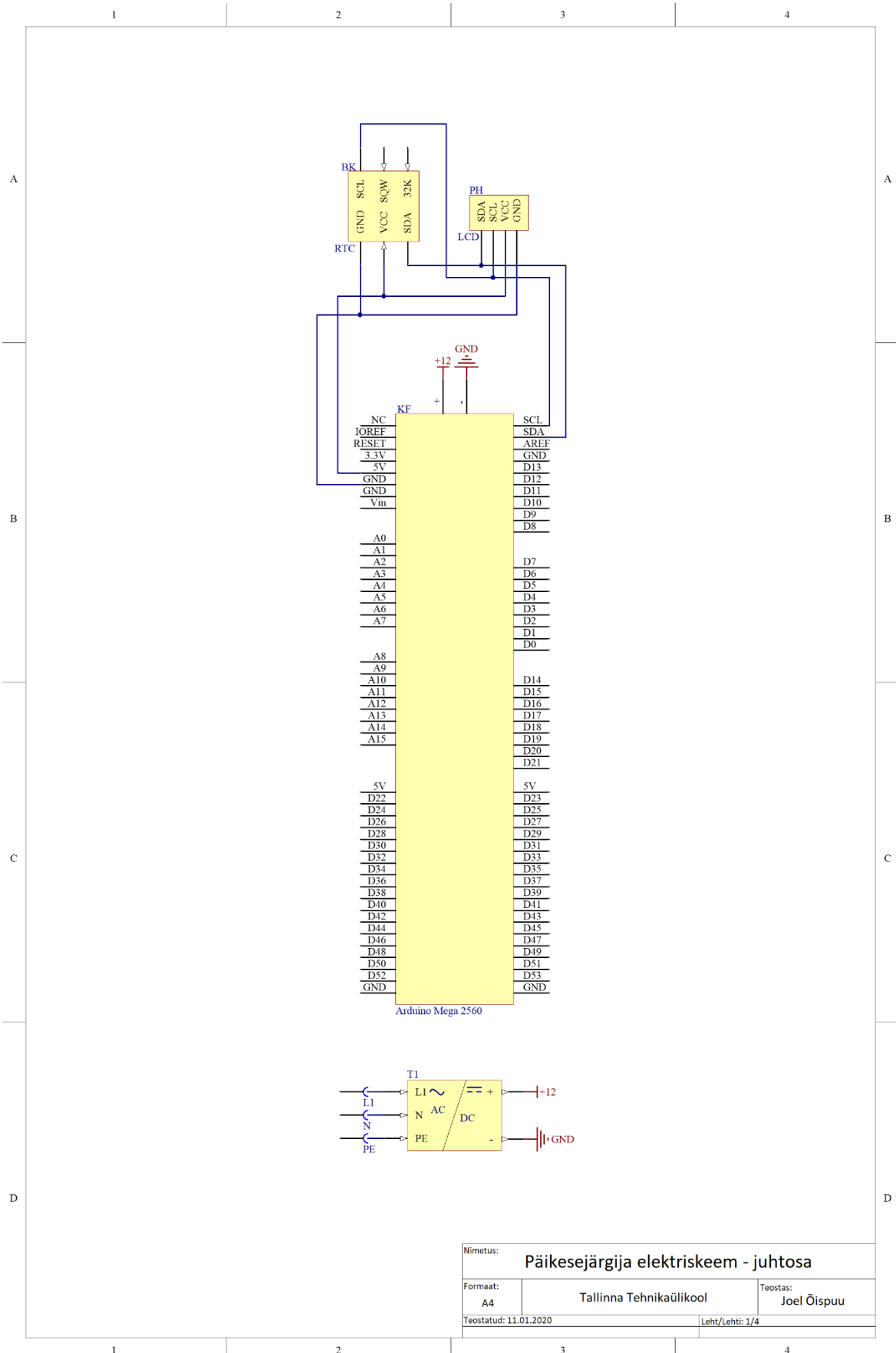
- [12] „Raspberry Pi 3 Model A+“, AliExpress, [Võrgumaterjal]. Saadaval: https://www.aliexpress.com/item/4000170312173.html?spm=a2g0o.productlist.0.0.22241a48cTvGaz&algo_pvid=056f068f-ef4c-4b43-81d9-ac6a0b1025ad&algo_expid=056f068f-ef4c-4b43-81d9-ac6a0b1025ad-2&btsid=0ab6f83a15885981959274964e431c&ws_ab_test=searchweb0_0,searc. [Kasutatud 04. 05. 2020].
- [13] „Arduino Mega 2560“, Yeint, [Võrgumaterjal]. Saadaval: <https://www.yeint.ee/elektronika-1/arendusvahendid/arduino/arduino-platvormid/arduino-mega-2560-rev3.> [Kasutatud 04. 05. 2020].
- [14] „RTC DS323 moodul“, Aliexpress, [Võrgumaterjal]. Saadaval: https://www.aliexpress.com/item/4000042855547.html?spm=a2g0o.productlist.0.0.54217374Ljk7zl&algo_pvid=84f1cb4c-287e-44ea-b8b4-08b4384360d1&algo_expid=84f1cb4c-287e-44ea-b8b4-08b4384360d1-0&btsid=0ab6f82315886778010367063e486b&ws_ab_test=searchweb0_0,searc. [Kasutatud 05. 05. 2020].
- [15] „1,3 tolli OLED ekraan“, Aliexpress, [Võrgumaterjal]. Saadaval: https://www.aliexpress.com/item/32845056725.html?spm=a2g0o.productlist.0.0.4eaba4d0QsLbSK&algo_pvid=998b6dc9-fa07-4d7d-9aaf-8f78fb7bdfd7&algo_expid=998b6dc9-fa07-4d7d-9aaf-8f78fb7bdfd7-24&btsid=0ab6f82215886806668574000e23f3&ws_ab_test=searchweb0_0,search. [Kasutatud 05. 05. 2020].
- [16] „Anemometer“, Aliexpress, [Võrgumaterjal]. Saadaval: https://www.aliexpress.com/item/32816999132.html?spm=a2g0o.productlist.0.0.8fa77830jYKiHI&algo_pvid=b330af5b-fc1b-4699-9b1b-b3c65b0c9f60&algo_expid=b330af5b-fc1b-4699-9b1b-b3c65b0c9f60-29&btsid=0ab6f83a15835912756903497e7d48&ws_ab_test=searchweb0_0,sea. [Kasutatud 05. 05. 2020].

LISAD

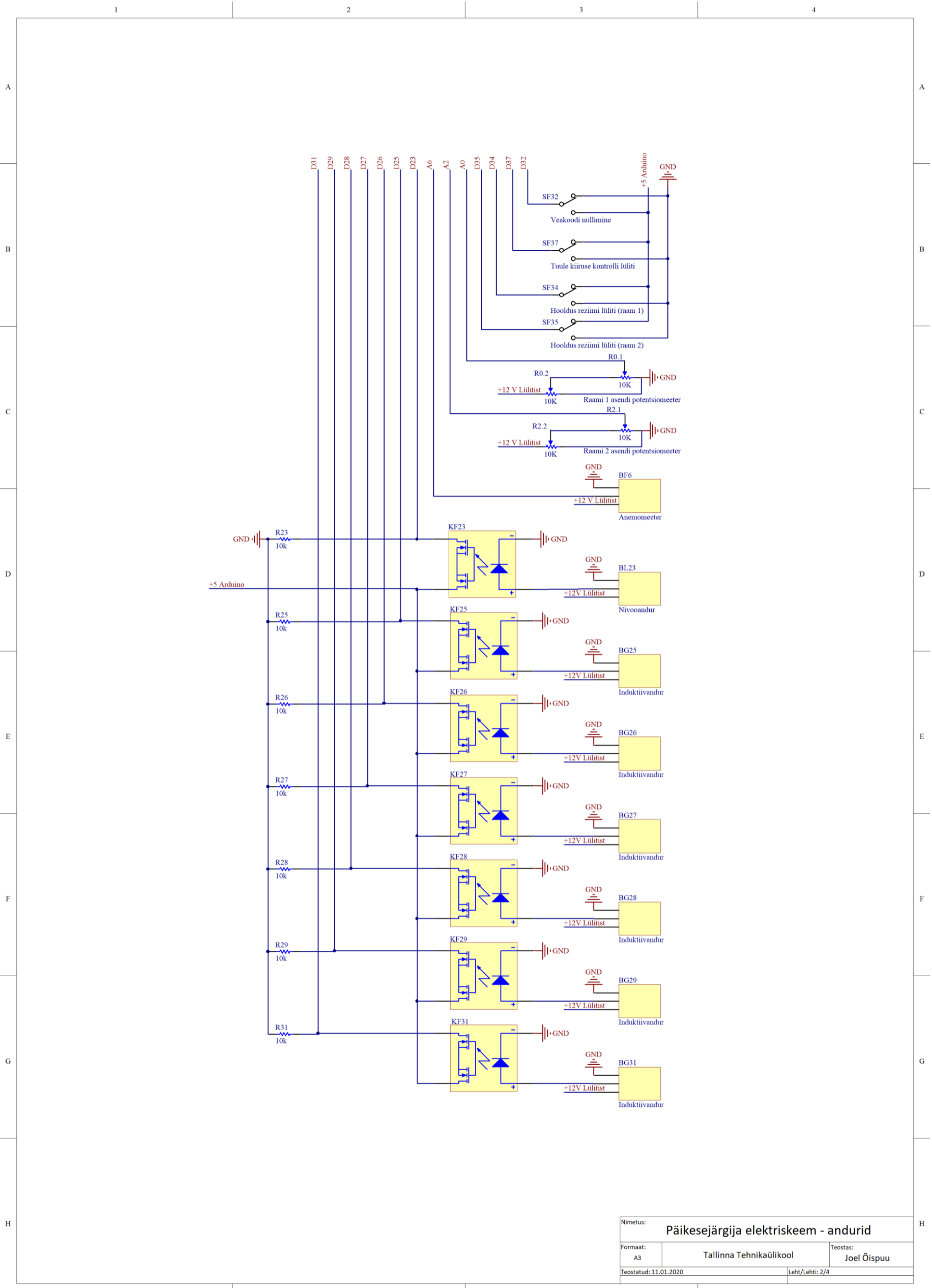
Lisa 1 Hüdraulikaskeem



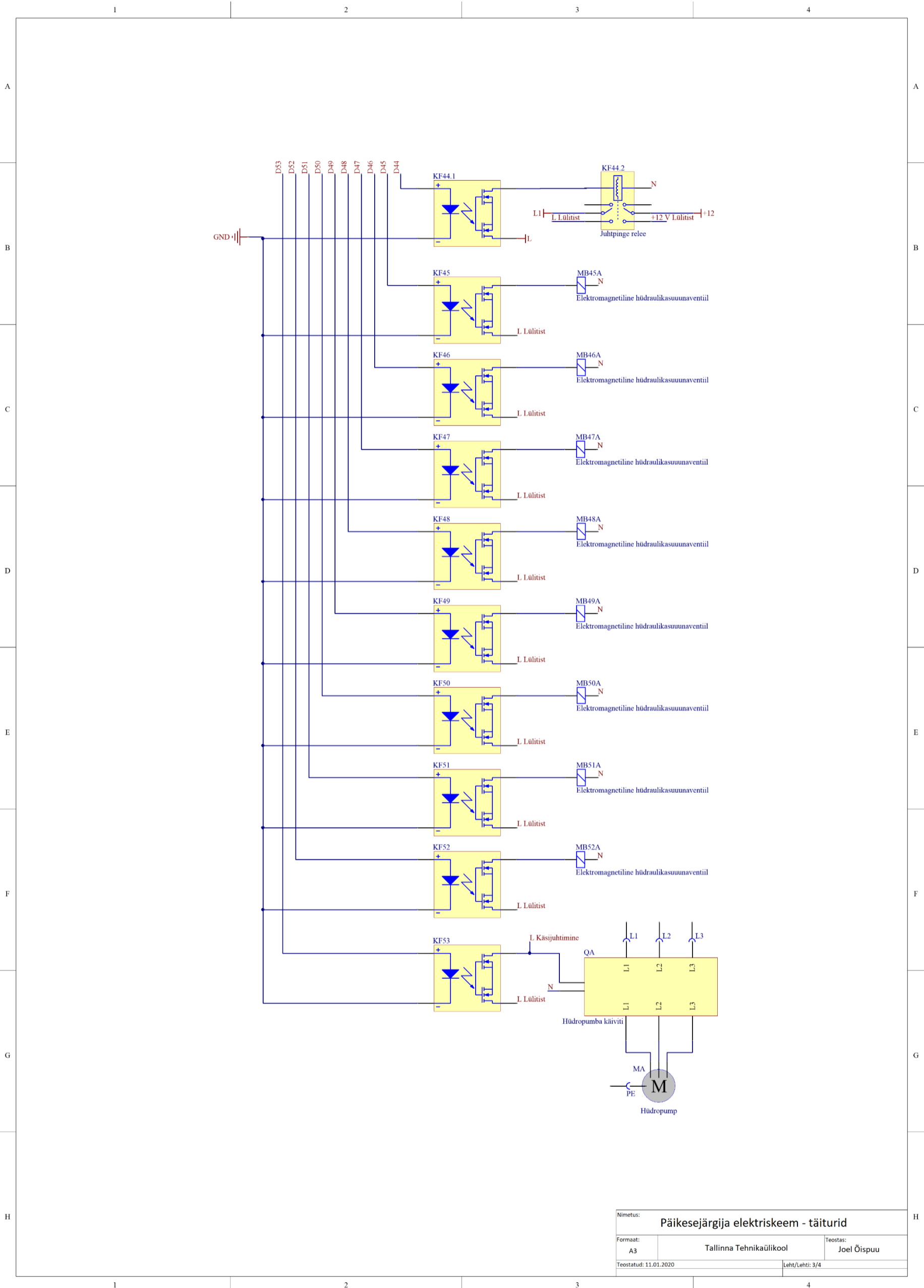
Lisa 2 Juhtosa elektriskeem



Nimetus: Pääkesejärgija elektriskeem - juhtosa		
Formaat: A4	Tallinna Tehnikaülikool	Teostas: Joel Õispuu
Teostatud: 11.01.2020		Leht/Lehti: 1/4

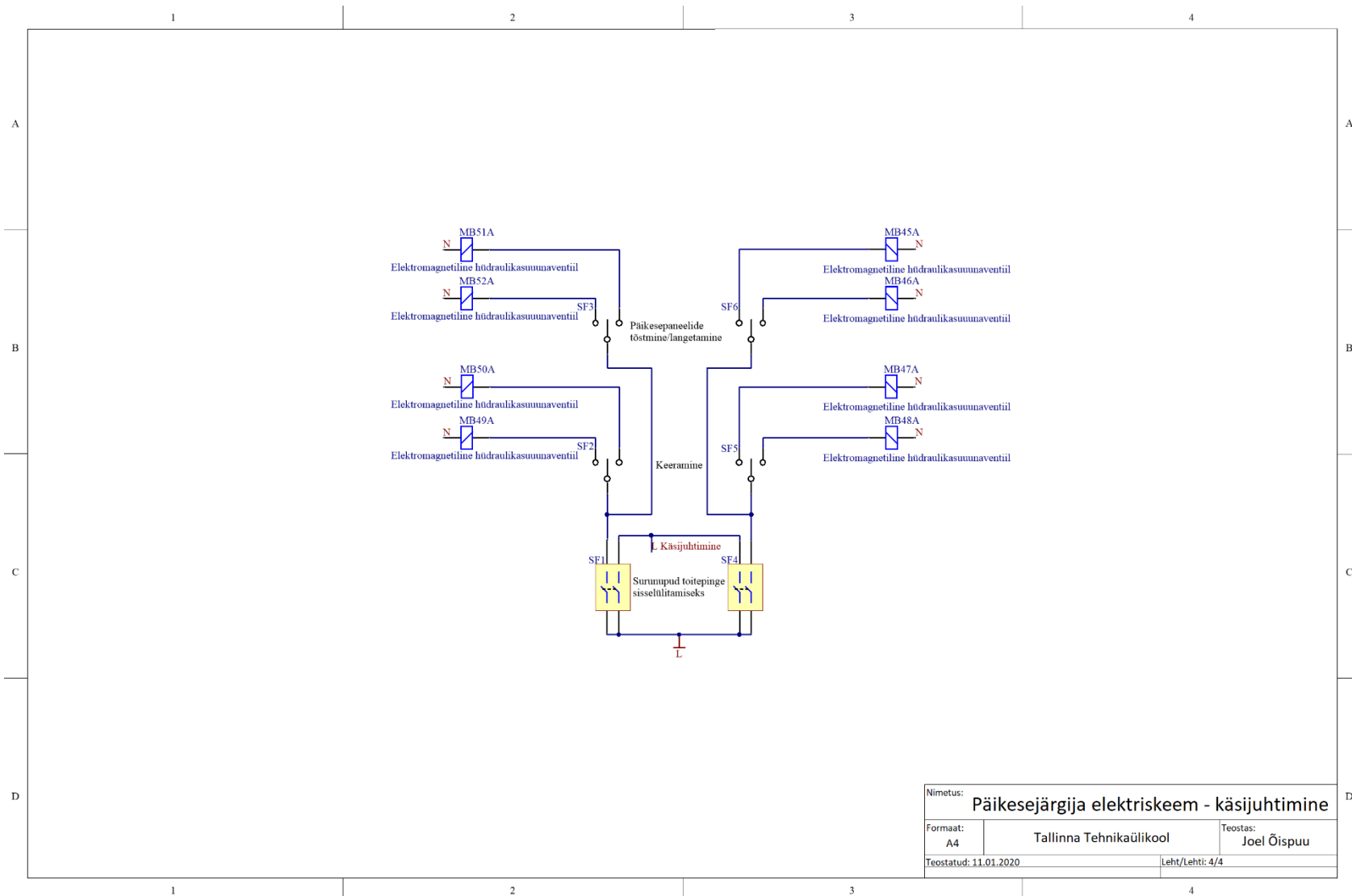


Nimetus: Päikeseljõrgija elektriskeem - andurid		
Formaat: A3	Tallinna Tehnikaülikool	Teostas: Joel Õispuu
Teostatud: 11.01.2020		Leht/Lehti: 2/4

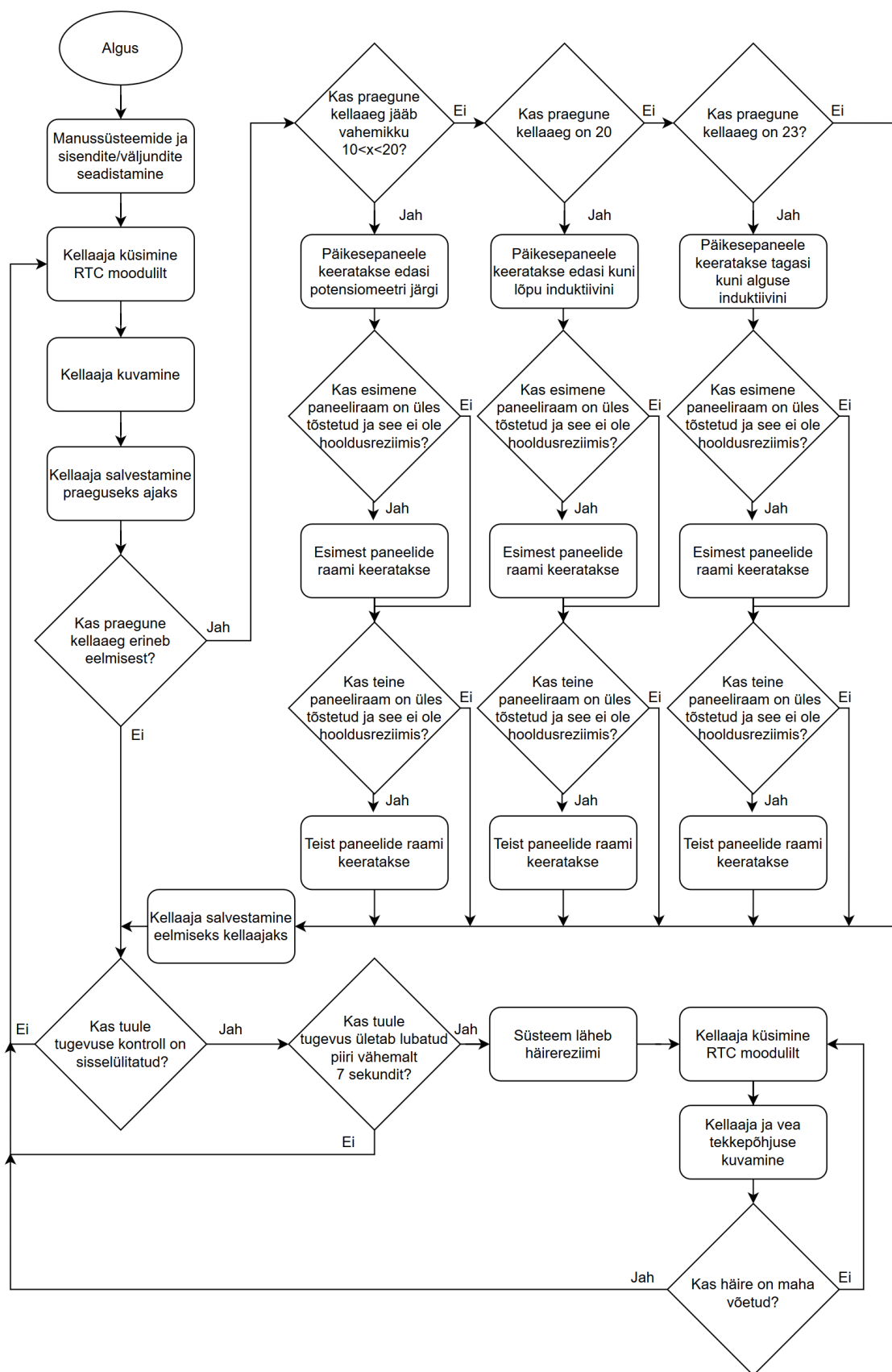


Nimetus: Päkesejärgija elektriskeem - täiturid		
Formaat: A3	Tallinna Tehnikaülikool	Teostas: Joel Öispuu
Teostatud: 11.01.2020		Leht/Lehti: 3/4

Lisa 5 Käsijuhtimise elektriskeem



Lisa 6 Programmi põhimõtteline plokkskeem



```

#include <Wire.h>
#include <StopWatch.h>
#include <DS3231.h>
#include <U8glib.h>

U8GLIB_SH1106_128X64 u8g(U8G_I2C_OPT_NONE); //õige ekraanitüübi valimine
RTClib RTC; //reaalajakell initsialiseerimine
StopWatch MySW; //stopperi initsialiseerimine

int juhtpinge=44; //juhtpinge sisselülitamine
int olitase=23; //õli taseme signaal
int pump=53; //hüdro pumba sisselülitamine
int restart=32; //veasignaali nullimiseks mõeldud nupp
int tuul=A6; //tuule kiiruse anduri signaal

int all1=45; //paneeli 1 alla laskmine
int ules1=46; //paneeli 1 üles tõstmine
int all1=31; //paneeli 1 all induktiiv
int ulevall=30; //paneeli 1 üleval induktiiv
int asend1=A2; //paneeli 1 asend
int paneeledasi1=47; //paneeli 1 edasi keeramine
int paneeltagasi1=48; //paneeli 1 tagasi keeramine
int paneellopus1=28; //paneeli 1 lõpu induktiiv
int paneelalguses1=29; //paneeli 1 alguse induktiiv
int hooldus1=34; //paneeli 1 hoolduses olemise kontroll

int alla2=51; //paneeli 2 alla laskmine
int ules2=52; //paneeli 2 üles tõstmine
int all2=25; //paneeli 2 all induktiiv
int ulevall2=24; //paneeli 2 üleval induktiiv
int asend2=A3; //paneeli 2 asend
int paneeledasi2=50; //paneeli 2 edasi keeramine
int paneeltagasi2=49; //paneeli 2 tagasi keeramine
int paneellopus2=26; //paneeli 2 lõpu induktiiv
int paneelalguses2=27; //paneeli 2 alguse induktiiv
int hooldus2=35; //paneeli 2 hoolduses olemise kontroll

int kellnuud; //praegusel kontrollil saadud kellaeg
int kelleelmine=0; //eelmisel kontrollil saadud kellaeg (vajalik et teada saada kas kella aeg on muutunud)
float potel=0; //esimese paneeli potentsiomeetri pinge (näitab esimese paneeli asendit pingena)
float pote2=0; //teise paneeli potentsiomeetri pinge (näitab teise paneeli asendit pingena)
int juhtpingekontroll=0; //kontrollitakse kas juhtsignaal on sisse lülitatud
long keeramisaeg=40000; //maksimaalne paneeli keeramis aeg millisekundites
long keeramisaeg2=150000; //maksimaalne paneeli tagasi keeramis aeg millisekundites
int mootevahe = 200; //aeg kahe mõõtmise vahel millisekundites
int n=0;
int i=0;
long kontrollaaeg=0; //keeramise ajalise kestvuse kontrollimiseks
int tuulekontroll=37; //tuule kontrolli sisse/välja lülitamiseks
float tuulepinge=3.3; //maksimaalse tuulekiiruse piiraja
int kellaag[9]={9, 10, 11, 12, 13, 14, 15, 16, 17}; //ette antud kellaajad millal paneelid peavad keerama
float pinge[9]={0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5}; //ette antud asendid millesse paneelid peavad teatud kella ajal keerama

void setup () {
//sisendite ja väljundite seadmine
pinMode(juhtpinge, OUTPUT);
pinMode(olitase, INPUT);

```

```

pinMode(pump, OUTPUT);
pinMode(restart, INPUT);
pinMode(tuulekontroll, INPUT);
pinMode(hooldus1, INPUT);
pinMode(hooldus2, INPUT);

pinMode(alla1, OUTPUT);
pinMode(ules1, OUTPUT);
pinMode(all1, INPUT);
pinMode(ulevall, INPUT);
pinMode(tuul, INPUT);
pinMode(asend1, INPUT);
pinMode(paneeledasi1, OUTPUT);
pinMode(paneeltagasi1, OUTPUT);
pinMode(paneellopus1, INPUT);
pinMode(paneelalguses1, INPUT);

pinMode(alla2, OUTPUT);
pinMode(ules2, OUTPUT);
pinMode(all2, INPUT);
pinMode(uleval2, INPUT);
pinMode(asend2, INPUT);
pinMode(paneeledasi2, OUTPUT);
pinMode(paneeltagasi2, OUTPUT);
pinMode(paneellopus2, INPUT);
pinMode(paneelalguses2, INPUT);

digitalWrite(juhtpinge, LOW); //väljundite seadmine algolekusse olekusse
digitalWrite(pump, HIGH);
digitalWrite(alla1, LOW);
digitalWrite(ules1, LOW);
digitalWrite(paneeledasi1, HIGH);
digitalWrite(paneeltagasi1, HIGH);
digitalWrite(alla2, HIGH);
digitalWrite(ules2, HIGH);
digitalWrite(paneeledasi2, HIGH);
digitalWrite(paneeltagasi2, HIGH);

Serial.begin(57600); //seeriali käivitamine
Wire.begin(); //käivitatakse "Wire" teek ja hakatakse I2C siini ülemaks
u8g.setFont(u8g_font_unifont); //ekraani ereduse ja kirja suuruse seadistamine
if ( u8g.getMode() == U8G_MODE_R3G3B2 ) {
    u8g.setColorIndex(255); // valge
}
else if ( u8g.getMode() == U8G_MODE_GRAY2BIT ) {
    u8g.setColorIndex(3); // maksimaalne eredus
}
else if ( u8g.getMode() == U8G_MODE_BW ) {
    u8g.setColorIndex(1);
}
else if ( u8g.getMode() == U8G_MODE_HICOLOR ) {
    u8g.setHiColorByRGB(255,255,255);
}
}

void loop () {
    KEERAMINE(); //paneelide keeramine (vajadusel)
    TUUL(); //tuule olemasolu kontrollimine
    delay(500); //viide et kella ei uuendataks liiga sageli
}

```

```

}

//"Void JUHTPINGE"-t kasutatakse juhtpinge ja hüdropumba sisselülitamiseks.
//Juhul kui õlitase on liiga madal ei lülitata pumpa järgi ja süsteem läheb häire režiimi.
void JUHTPINGE(){
digitalWrite(juhtpinge, HIGH); //juhtpinge sisselülitamine
delay(1000); //viide, et juhtpingega kasnevad lülitused kõik tehtud oleks (kaasaarvatud õlitaseme anduri sisse lülitamine)
if (digitalRead(olitase) == 1){ //juhul kui õli tase on olemas lülitatakse hüdropump järgi
digitalWrite(pump, LOW);
delay(1000); //viide et hüdropump jõuaks järgi lülituda
}
else{ //juhul kui õlitase on liiga madal läheb seade häire režiimi
STOPP(1); //häire funktsiooni väljakutsumine, kaasa antakse häire tekkepõhjuse kood (NIVOO)
return;
}
}

//"Void OLITASE"-T kasutatakse õli taseme kontrollimiseks paneelide liigutamise ajal.
//Juhul kui õlitase on liiga madal kahel 1 sekundilise vahega mõõtmisel läheb süsteem häire režiimi.
void OLITASE(){
if (digitalRead(olitase) == 0){ //esimene kontroll
delay (1000); //juhul kui õli tase oli liiga madal tehakse ühe sekundiline viide
if (digitalRead(olitase) == 0){ //teine kontroll
STOPP(1); //juhul kui õlitase oli uuesti liiga madal kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (NIVOO)
return;
}
}
}

//"Void KEERAMINE" kasutatakse paneelide keeramiseks.
//Keeramine on jagatud kolmeks osaks:
// Esimeses osas keeratakse vastavalt potentsiomeetrilt saadava pinge järgi.
// Teises osas tehakse paneeli lõppu keeramine, lõpus olemise signaal saadakse induktiivandurilt.
// kolmandas osas tehakse paneeli tagasipööramine, alguses olemise signaal saadakse induktiivandurilt.
//Lisaks on kõik kolm veel omakorda kaheks jagatud: esimese paneeli keeramine ja teise paneeli keeramine.
void KEERAMINE(){
n=0;
i=0;
kontrollaeg=0; //kontrollaeg nullitakse
kellnuud=KELL(); //praeguse kella aja saamine ja ekraanil oleva kella aja uuendamine
if (kellnuud != kelleelmine){ //keeramine saab toimuda ainult siis kui praegune kellaag ei ole sama mis eelmine kellaag
if (kellnuud > (8) && kellnuud < (18) ){ //juhul kui kell on vahemikus 9 kuni 17 siis toimub potentsiomeetri järgi keeramine.
digitalWrite(juhtpinge, HIGH); //juhtpinge signaal pannakse kõrgesse olekusse et saaks kontrollida ega paneel(-id) pole käsitsi alla lastud
delay(200); //viide, et juhtpingega kasnevad lülitused kõik tehtud oleks
while (kellnuud != kellaag[n] || n >10){ //kella aja järgi soovitud pinge leidmine
n++;
}
if ((digitalRead(all1)==0)&&(digitalRead(hooldus1)!=0)){ //esimese paneeli keeramine juhul kui paneel pole alumises asendis
JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
juhtpingekontroll=1; //juhtpinge kontroll kirjutatakse üheks et teist paneeli keermates oleks teada et pole vaja juhtpinge funktsiooni
välja kutsuda (hüdropump on järgi juba)
MySW.start(); //kontrollaja jaoks stopperi käivitamine
while (i!=1) { //paneel keerab kuni kahel mõõtevahega määratud vahega mõõtmisel on pinge olnud sobiv
digitalWrite(paneelidasil, LOW); //paneeli hakatakse edas keerama
OLITASE(); //kontrollitakse õlitaseme olemasolu
potel = ((analogRead(asend1)/1023.0)*5.0); //paneeli asendit näitava potentsiomeetri mõõtmine ja pingeks konverteerimine
Serial.println(potel);
if (potel > pinge[n] ) { //juhul kui esimesel korral on pinge olnud sobiv mõõdetakse ka teisel korral
delay (mootevahe); //viivitatus kahe mootmise vahel
}
}
}
}

```

```

    pote1 = ((analogRead(asend1)/1023.0)*5.0); //teine potentsiomeetri mõõtmine
    if (pote1 > pingel[n] ) { //võrdlus, kas teisel mõõtmisel on pingel olnud sobiv
        digitalWrite(paneledasil, HIGH); //keeramine peatatakse
        i=1; //while tsüklist väljumiseks
        MySW.stop(); //taimer seisma
        MySW.reset(); //taimer reset
    }
}
else if (MySW.elapsed()>keeramisaeg){ //juhul kui paneelid on liiga kaua keeranud läheb seade häire režiimi
    digitalWrite(paneledasil, HIGH); //keeramine peatatakse
    i=1; //while tsüklist väljumine
    MySW.stop(); //taimer seisma
    MySW.reset(); //taimer reset
    STOPP(2); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 1)
}
}
}
i=0;
if ((digitalRead(all2)==0)&&(digitalRead(hooldus2)!=0)){ //teise paneeli keeramine juhul kui paneel pole alumises asendis
    if (juhtpingekontroll != 1){ //kontrollitakse kas juhtpinge ja hüdro pump on juba sees (kui ei ole siis lülitatakse juhtpinge ja hüdro pump
järgi)
        JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
    }
    MySW.start(); //kontrollaja jaoks stopperi käivitamine
    while (i!=1) { //paneel keerab kuni kahel mõõtevahega määratud vahega mõõtmisel on pingel olnud sobiv
        digitalWrite(paneledasi2, LOW); //paneeli hakatakse edas keerama
        OLITASE(); //kontrollitakse õlitaseme olemasolu
        pote2 = ((analogRead(asend2)/1023.0)*5.0); //paneeli asendit näitava potentsiomeetri mõõtmine ja pingeks konverteerimine
        Serial.println(pote2);
        if (pote2 > pingel[n] ) { //juhul kui esimesel korral on pingel olnud sobiv mõõdetakse ka teisel korral
            delay (mootevahe); //viivitus kahe mootmise vahel
            pote2 = ((analogRead(asend2)/1023.0)*5.0); //teine potentsiomeetri mõõtmine
            if (pote2 > pingel[n] ) { //võrdlus, kas teisel mõõtmisel on pingel olnud sobiv
                digitalWrite(paneledasi2, HIGH); //keeramine peatatakse
                i=1; //while tsüklist väljumiseks
                MySW.stop(); //taimer seisma
                MySW.reset(); //taimer reset
            }
        }
        else if (MySW.elapsed()>keeramisaeg){ //juhul kui paneelid on liiga kaua keeranud läheb seade häire režiimi
            digitalWrite(paneledasi2, HIGH); //keeramine peatatakse
            i=1; //while tsüklist väljumine
            MySW.stop(); //taimer seisma
            MySW.reset(); //taimer reset
            STOPP(3); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 2)
        }
    }
}
delay(300); //viide silindri ja hüdro pumba seisma jätmise vahel
digitalWrite(pump, HIGH); //hüdro pump seisma
delay(1000); //viide hüdropumba seisma jätmise ja juhtpinge maha võtmise vahel
digitalWrite(juhtpinge, LOW); //juhtpinge maha
}
else if (kellnuud == (18)){ //juhul, kui kell on saanud 18 siis teevad paneelid viimase keeramise edasi
    digitalWrite(juhtpinge, HIGH); //juhtpinge signaal pannakse kõrgesse olekusse et saaks kontrollida ega paneel(-id) pole käsitsi alla lastud
    delay(200); //viide, et juhtpingega kasnevad lülitused kõik tehtud oleks
    if ((digitalRead(all1)==0)&&(digitalRead(hooldus1)!=0)){ //esimese paneeli keeramine lõppu, juhul kui paneel pole alumises asendis
        JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
        juhtpingekontroll=1; //juhtpinge kontroll kirjutatakse üheks et teist paneeli keerates oleks teada et pole vaja juhtpinge funktsiooni
välja kutsuda (hüdropump on järgi juba)
    }
}

```

```

MySW.start(); //kontrollaja jaoks stopperi käivitamine
do { //paneeli keeratakse nii kaua kuni lõpu induktsioonilt saadakse signaal või kuni kontrollaeg on ületatud
  OLITASE(); //kontrollitakse õlitaseme olemasolu
  digitalWrite(paneelidasil, LOW); //paneeli hakatakse edas keerama
  delay(300);
  kontrollaaeg = MySW.elapsed(); //keeramise algusest möödunud aja salvestamine
} while (digitalRead(paneellopus1) == 0 && kontrollaaeg < keeramisaeg);
if (kontrollaaeg > keeramisaeg){ //juhul kui keeramine lõpetati liiga pika keeramise aja tõttu läheb seade häire režiimi
  digitalWrite(paneelidasil, HIGH); //keeramine peatatakse
  MySW.stop(); //taimer seisma
  MySW.reset(); //taimer reset
  STOPP(2); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 1)
}
else{ //juhul kui keeramine lõpetati paneelide lõppu jõudmise tõttu
  MySW.stop(); //taimer seisma
  MySW.reset(); //taimer reset
  delay(5000);
  digitalWrite(paneelidasil, HIGH); //keeramine peatatakse
  kontrollaaeg=0; //kontrollaja nullimine
}
}
if ((digitalRead(all2)==0)&&(digitalRead(hooldus2)!=0)){ //teise paneeli keeramine lõppu, juhul kui paneel pole alumises asendis
  if (juhtpingekontroll != 1){ //kontrollitakse kas juhtpinge ja hüdro pump on juba sees (kui ei ole siis lülitatakse juhtpinge ja hüdro pump
järgi)
    JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
  }
  MySW.start(); //kontrollaja jaoks stopperi käivitamine
  do { //paneeli keeratakse nii kaua kuni lõpu induktsioonilt saadakse signaal või kuni kontrollaeg on ületatud
    OLITASE(); //kontrollitakse õlitaseme olemasolu
    digitalWrite(paneelidasi2, LOW); //paneeli hakatakse edas keerama
    delay(300);
    kontrollaaeg = MySW.elapsed(); //keeramise algusest möödunud aja salvestamine
  } while (digitalRead(paneellopus2) == 0 && kontrollaaeg < keeramisaeg);
  if (kontrollaaeg > keeramisaeg){ //juhul kui keeramine lõpetati liiga pika keeramise aja tõttu läheb seade häire režiimi
    digitalWrite(paneelidasi2, HIGH); //keeramine peatatakse
    MySW.stop(); //taimer seisma
    MySW.reset(); //taimer reset
    STOPP(3); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 2)
  }
  else{ //juhul kui keeramine lõpetati paneelide lõppu jõudmise tõttu
    MySW.stop(); //taimer seisma
    MySW.reset(); //taimer reset
    delay(5000);
    digitalWrite(paneelidasi2, HIGH); //keeramine peatatakse
    kontrollaaeg=0; //kontrollaja nullimine
  }
}
delay(300); //viide silindri ja hüdro pumba seisma jätmise vahel
digitalWrite(pump, HIGH); //hüdro pump seisma
delay(1000); //viide hüdropumba seisma jätmise ja juhtpinge maha võtmise vahel
digitalWrite(juhtpinge, LOW); //juhtpinge maha
}
else if (kellnuud == (23)){ //juhul, kui kell on saanud 23 siis keeravad paneelid ennast algusesse ehk koduasendisse
  digitalWrite(juhtpinge, HIGH); //juhtpinge signaal pannakse kõrgesse olekusse et saaks kontrollida ega paneel(-id) pole käsitsi alla lastud
  delay(200); //viide, et juhtpingega kasnevad lülitused kõik tehtud oleks
  if ((digitalRead(all1)==0)&&(digitalRead(hooldus1)!=0)){ //esimese paneeli keeramine koju, juhul kui paneel pole alumises asendis
    JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
    juhtpingekontroll=1; //juhtpinge kontroll kirjutatakse üheks et teist paneeli keerates oleks teada et pole vaja juhtpinge funktsiooni
välja kutsuda (hüdropump on järgi juba)
    MySW.start(); //kontrollaja jaoks stopperi käivitamine
  }
}

```

```

do {
    OLITASE(); //paneele keeratakse nii kaua kuni algus induktsioonilt saadakse signaal või kuni kontrollaeg on ületatud
    digitalWrite(paneeltagasi1, LOW); //kontrollitakse õlitaseme olemasolu
    delay(300); //paneeli hakatakse tagasi keerama
    kontrollaeg = MySW.elapsed(); //keeramise algusest möödunud aja salvestamine
} while (digitalRead(paneelalguses1) == 0 && kontrollaeg < keeramisaeg2);
if (kontrollaeg > keeramisaeg2){ //juhul kui keeramine lõpetati liiga pika keeramise aja tõttu läheb seade häire režiimi
    digitalWrite(paneeltagasi1, HIGH); //keeramine peatatakse
    MySW.stop(); //taimer seisma
    MySW.reset(); //taimer reset
    STOPP(2); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 1)
}
else{ //juhul kui keeramine lõpetati paneelide lõppu jõudmise tõttu
    delay(5000);
    digitalWrite(paneeltagasi1, HIGH); //keeramine peatatakse
    MySW.stop(); //taimer seisma
    MySW.reset(); //taimer reset
    kontrollaeg=0; //kontrollaja nullimine
}
}
if ((digitalRead(all2)==0)&&(digitalRead(hooldus2)!=0)){ //teise paneeli keeramine koju, juhul kui paneel pole alumises asendis
    if (juhtpingekontroll != 1){ //kontrollitakse kas juhtpinge ja hüdro pump on juba sees (kui ei ole siis lülitatakse juhtpinge ja hüdro pump
järgi)
        JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
    }
    MySW.start(); //kontrollaja jaoks stopperi käivitamine
    do {
        OLITASE(); //paneele keeratakse nii kaua kuni algus induktsioonilt saadakse signaal või kuni kontrollaeg on ületatud
        digitalWrite(paneeltagasi2, LOW); //kontrollitakse õlitaseme olemasolu
        delay(300); //paneeli hakatakse tagasi keerama
        kontrollaeg = MySW.elapsed(); //keeramise algusest möödunud aja salvestamine
    } while (digitalRead(paneelalguses2) == 0 && kontrollaeg < keeramisaeg2);
    if (kontrollaeg > keeramisaeg2){ //juhul kui keeramine lõpetati liiga pika keeramise aja tõttu läheb seade häire režiimi
        digitalWrite(paneeltagasi2, HIGH); //keeramine peatatakse
        MySW.stop(); //taimer seisma
        MySW.reset(); //taimer reset
        STOPP(3); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG 2)
    }
    else{ //juhul kui keeramine lõpetati paneelide lõppu jõudmise tõttu
        delay(5000);
        digitalWrite(paneeltagasi2, HIGH); //keeramine peatatakse
        MySW.stop(); //taimer seisma
        MySW.reset(); //taimer reset
        kontrollaeg=0; //kontrollaja nullimine
    }
}
delay(300); //viide silindri ja hüdro pumba seisma jätmise vahel
digitalWrite(pump, HIGH); //hüdro pump seisma
delay(1000); //viide hüdropumba seisma jätmise ja juhtpinge maha võtmise vahel
digitalWrite(juhtpinge, LOW); //juhtpinge maha
}
}
kelleelmine=kellnuud; //praeguse kella aja salvestamine et saaks kontrollida ega kella aeg muutunud pole
return;
}

```

```

//"Void TUUL" kasutatakse tuule kiiruse mõõtmiseks
//Juhul kui tuule kiiruse mõõtjalt saadakse seitsmel (ühe sekundilise vahega) mõõtmisel
//suurem pinge kui ette antud, siis langetatakse paneelid ning seade läheb häire režiimi.

```



```

void TUUL(){
if (digitalRead(tuulekontroll) == 0){
kontrollaeg=0; //kontrollaja nullimine
juhtpingekontroll=0; //juhtpinge kontrolli nullimine
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
delay(1000);
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
delay(1000);
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
delay(1000);
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
delay(1000);
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
delay(1000);
if (((analogRead(tuul)/1023.0)*5.0)>=tuulepinge){
MySW.reset(); //taimer reset
MySW.start(); //taimer käima
digitalWrite(juhtpinge, HIGH); //juhtpinge signaal pannakse kõrgesse olekusse et saaks kontrollida ega paneel(-id) pole käsitsi alla lastud
delay(200); //viide, et juhtpingega kasnevad lülitused kõik tehtud oleks
if (digitalRead(all1) == 0){ //esimest paneeli hakatakse alla laskma juhul kui paneel pole alumises asendis
JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
juhtpingekontroll=1; //juhtpinge kontroll kirjutatakse üheks et teist paneeli alla lastes oleks teada et pole vaja juhtpinge
funktsiooni välja kutsuda (hüdropump on järgi juba)
digitalWrite(alla1, HIGH); //esimest paneeli hakatakse alla laskma
}
if (digitalRead(all2) == 0){ //teist paneeli hakatakse alla laskma juhul kui paneel pole alumises asendis
if (juhtpingekontroll != 1){ //kontrollitakse kas juhtpinge ja hüdro pump on juba sees (kui ei ole siis lülitatakse juhtpinge ja hüdro pump
järgi)
JUHTPINGE(); //juhtpinge funktsiooni väljakutsumine, et lülitda järgi hüdropump
}
digitalWrite(alla2, LOW); //teist paneeli hakatakse alla laskma
}
while (((digitalRead(alla1) == 1) || (digitalRead(alla2) == 0)) && kontrollaeg < keeramisaeg){ //paneele lastakse alla niikaua kuni mõlemad
paneelid on all või kuni kontrollaeg pole ületatud
OLITASE(); //kontrollitakse õlitaseme olemasolu
if (digitalRead(all1) == 1){ //juhul kui esimene paneel on all, siis lõpetatakse selle allalaskmine
digitalWrite(alla1, LOW); //esimese paneeli alla laskmine peatatakse
}
if (digitalRead(all2) == 1){ //juhul kui teine paneel on all, siis lõpetatakse selle allalaskmine
digitalWrite(alla2, HIGH); //teise paneeli alla laskmine peatatakse
}
delay(200);
kontrollaeg=MySW.elapsed(); //kalla laskmise algusest möödunud aja salvestmine
}
digitalWrite(alla1, LOW); //esimese paneeli alla laskmine peatatakse
digitalWrite(alla2, HIGH); //teise paneeli alla laskmine peatatakse
if (kontrollaeg > keeramisaeg){ //juhul kui alla laskmine lõpetati liiga pika aja tõttu läheb seade häire režiimi
MySW.stop(); //taimer seisma
MySW.reset(); //taimer reset
STOPP(4); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (AEG + TUUL)
}
else{ //kui paneelid jõudsid alla kontrollaja jooksul läheb seade häire režiimi (teise häirekoodiga)
MySW.stop(); //taimer seisma
MySW.reset(); //taimer reset
kontrollaeg=0; //kontrollaja nullimine
STOPP(5); //kutsutakse välja häirefunktsioon, kaasa antakse häire tekkepõhjuse kood (TUUL)
}
}
}
}

```



```

else if (K==3){ //juhul kui häire kood on 3 siis kuvatakse veakood "AEG 2"
    u8g.print("AEG 2");
}
else if (K==4){ //juhul kui häire kood on 4 siis kuvatakse veakood "TUUL+AEG"
    u8g.print("TUUL+AEG");
}
else if (K==5){ //juhul kui häire kood on 5 siis kuvatakse veakood "TUUL"
    u8g.print("TUUL");
}
else { //juhul kui häire koodi ei ole kuvatakse ekraanile "..."
    u8g.print("...");
}
} while( u8g.nextPage() );
if ((digitalRead(restart) == 1)) //häire tühistatakse kui resettimiseks mõeldud nuppu hoitakse all vähemalt 1 sekund
delay(1000);
if ((digitalRead(restart) == 1)){
    i=1;
}
}
i=0;
return;
}

```

```

int KELL() { //RTC moodulilt kella aja saamine
    DateTime now = RTC.now();
    int X = now.second();
    String Aeg = String(X);
    const char* aeg = (const char*) Aeg.c_str();
    int Y = now.minute();
    String Aeg2 = String(Y);
    const char* aeg2 = (const char*) Aeg2.c_str();
    int Z = now.hour();
    String Aeg3 = String(Z);
    const char* aeg3 = (const char*) Aeg3.c_str();
    int x = now.day();
    String Aeg4 = String(x);
    const char* aeg4 = (const char*) Aeg4.c_str();
    int y = now.month();
    String Aeg5 = String(y);
    const char* aeg5 = (const char*) Aeg5.c_str();
    u8g.firstPage();
    do { //kellaaega kuvatakse kuni järgmise uuenduseni
        u8g.drawStr( 110, 22, aeg); //sekundi kuvamine ekraanile
        u8g.drawStr( 102, 22, ":");
        u8g.drawStr( 85, 22, aeg2); //minuti kuvamine ekraanile
        u8g.drawStr( 77, 22, ":");
        u8g.drawStr( 60, 22, aeg3); //tunni kuvamine ekraanile

        u8g.drawStr( 25, 22, aeg5); //kuu kuvamine ekraanile
        u8g.drawStr( 17, 22, "/");
        u8g.drawStr( 0, 22, aeg4); //päeva kuvamine ekraanile
    } while( u8g.nextPage() );
    return Z;
}

```