



TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND

Elektroenergeetika ja mehhatroonika instituut

SEADE SOOJUSARVESTI ARVUTUSPLOKI VOOKULU SIGNAALI SIMULEERIMISEKS

BAKALAUREUSETÖÖ
MEHHATROONIKA ÕPPEKAVA

Üliõpilane: Alexander Tsupsman

Üliõpilaskood: 134249 MAHB

Juhendaja: Andrei Pokatilov

2017

AUTORIDEKLARATSIOON

Deklareerin, et käesolev lõputöö, mis on minu iseseisva töö tulemus.

Esitatud materjalide põhjal ei ole varem akadeemilist kraadi taotletud.

Lõputöös kasutatud kõik teiste autorite tööd ja seisukohad ning materjalid on varustatud vastavate viidetega.

Töö valmis Andrei Pokatilovi juhendamisel

“.....”.....2017. a.

Töö autor: /allkiri/

Töö vastab lõputööle esitatavatele nõuetele

“.....”.....2017. a.

Juhendaja: /allkiri/

Lubatud kaitsmisele

“.....”.....2017. a.

MAHB õppekava lõputööde kaitsmiskomisjoni esimees: /allkiri/

BSC LÕPUTÖÖ ÜLESANNE

2017 aasta kevadsemester

Üliõpilane: Alexander Tsupsman, 134249MAHB

Õppekava: MAHB02/13

Eriala: Mehhatroonika

Juhendaja: teadur, Andrei Pokatilov

Konsultandid: Kristjan Tammik, etalonihoidja, mõõtevaldkonna spetsialist, AS Metrosert

Jaanus Vaht, IT juht, AS Metrosert

LÕPUTÖÖ TEEMA:

(eesti keeles) **Seade soojusarvesti arvutusploki vookulu signaali simuleerimiseks**

(inglise keeles) **Device for Flow Rate Signal Simulation of Heat Meter Calculator**

Töös lahendatavad ülesanded ja nende täitmise ajakava:

Nr	Ülesande kirjeldus	Täitmise tähtaeg
1.	Teema püstitamine. Soojusarvesti taatlusmetoodikaga tutvumine	14.03.2017
2.	Seadme projekteerimine.	28.03.2017
3.	Seadme valmistamine. Tarkvara koostamine.	28.04.2017
4.	Mehaanika projekteerimine – seadme korpuse projekteerimine ja tööjooniste koostamine.	10.05.2017
5.	Tehtud töö lõplik vormistamine, köitmine ja hindamisele esitamine.	22.05.2017

Lahendatavad insenertehnilised ja majanduslikud probleemid: Töö eesmärgiks on soojusarvesti taatlusmetoodikaga tutvumine, soojusarvesti arvutusploki vookulu signaali simuleerimiseks kasutatava seadme projekteerimine, realiseerimine ning korpuse projekteerimine. Töö käigus koostatakse elektroonikaskeem, juhtimistarkvara ja korpuse joonised.

Täiendavad märkused ja nõued: Seadme nõueteks on tema kompaktsus, jõudlus ja vastavus etteantud parameetritele.

Töö keel: Eesti

Kaitsmistaotlus esitada hiljemalt 15.05.2017. Töö esitamise tähtaeg 25.05.2017.

Üliõpilane	Alexander Tsupsman	/allkiri/	kuupäev 10.03.2017
Juhendaja	Andrei Pokatilov	/allkiri/	kuupäev 10.03.2017

SISUKORD

BSC LÕPUTÖÖ ÜLESANNE	3
SISSEJUHATUS	7
1. SOOJUSARVESTIGA TUTVUMINE	8
1.1 Mis on soojusarvesti? Mõisted.....	8
1.2 Soojusarvesti arvutusploki taatlusmetoodika	9
1.2.1 Mõõtevahendid	9
1.2.5 Soojusarvesti lubatud piirvead	11
2. SEADME PROJEKTEERIMINE	12
3. ELEKTROONIKA	14
3.1 Signaalgeneraatori valik	14
3.1.1 Seadmete ülevaade	14
3.1.2 Seadme valik	16
3.2 Relee valik.....	16
3.3 Elektriskeemi koostamine	17
4. JUHTIMINE	19
4.1 Täidetavad ülesanded	19
4.2 Programmeerimiskeele valik.....	19
4.3 Programmi juhtimise põhimõte	19
5. MEHAANIKA	22
5.1 Korpuse projekteerimine.....	22
6. SEADME TESTIMINE	26
6.1 Seadme majanduslik analüüs.....	27
KOKKUVÕTTE	29
CONCLUSION	30
KASUTATUD KIRJANDUS	32
LISAD	33
Lisa 1 – Arduino Uno juhtimiskood	33
Lisa 2 – Korpuse joonised	36
Lisa 3 – Pildid komplekteeritud seadmest.....	39

EESSÕNA

Käesoleva lõputöö teema anti autorile AS Metrosert poolt. Töö koostamine, seadme projekteerimine, prototüübi valmistamine ja katsetamine toimus AS Metrosert poolt mulle eraldatud labori osas. Teema püstitamise ja algandmetega abistas Kristjan Tammik, kes on etalonihoidja, mõõtevaldkonna spetsialist AS Metroserdis. Tarkvara koostamisega ja katsetamisega abistas Jaanus Vahi, AS Metrosert IT juht. Üldküsimumuste osas pöördusin juhendaja poole.

SISSEJUHATUS

Lõputöö teema on soojusarvesti arvutusploki vookulu signaali simuleerimise seadme projekteerimine, realiseerimine, testimine ja korpuse projekteerimine. Minu meelest on antud teema aktuaalne ja tähtis, sest ettevõttes eelnevalt kasutatud arvutusploki vookulu signaali simuleerimise seade lakkas töötamast. Selle seadme asemel kasutatav signaalgeneraator asub teises töökohas ning on kasutatav teiseks otstarbeks. Arvutusploki vookulu signaali simuleerimiseks tuleb mõõtesüsteem enne iga kord mõõtmist uuesti üles seada. Lisaks vookulu signaali genereerimise automatiseerimine lihtsustab töö käiku ja säästab aega selle ülesannete sooritusele.

Lisaks eelnevale on minu ülesandeks soojusarvesti taatlusmetoodikaga tutvumine ja laboris tööolukorras prototüübi katsetamine.

Uurimismeetodina kasutan vastava kirjanduse lugemist ja analüüsimist, soojusarvesti taatlusmeetodika ja EVS-EN standartidega tutvustamist. Töökäigus kasutan aja intervallide mõõturit ja ostsillograafi seadme testimiseks, Arduino keskkonda mikrokontrolleri programmeerimiseks ja koodi testimiseks, Solidworksi seadme korpuse projekteerimiseks.

Töö on koostatud kuues osas. Esimese osas on soojusarvesti mõistega ja tema taatlusmetoodikaga tutvumine. Seadme ja mõõtevahendite taatlemiseks uurimine. On esitatud peamised valemid, mõisted ja töötlemismeetod.

Teises osas kirjeldatakse seadme projekteerimist- koos plokk skeemiga. Genereeritava impulsside parameetrite leidmine. Tehtud ka lühike ülevaade vajalikes komponentidest.

Kolmandas osas on elektroonika. Põhiliselt põhjendatud impulsside generaatori ja relee valik. Koostatud seadme elektroonikaskeem.

Neljandas osas püstitakse juhtimisülesandeid ning nende lahendamist. Koostatud programmi andmevoo skeem.

Viiendas osas on mehaanika ning kirjeldatakse korpuse disain koos CAD mudeliga.

Ja viimases osas on tehtud terve töö analüüs. Selle sisse kuulub seadme testimine kui ka majandusklik analüüs.

1. SOOJUSARVESTIGA TUTVUMINE

1.1 Mis on soojusarvesti? Mõisted.

Soojusarvesti – mõõtevahend, mis on ette nähtud tarbitud või toodetud soojushulga mõõtmiseks.

Soojusarvesti on kas komplektne mõõtevahend või kombineeritud mõõtevahend, mis koosneb alakoostudest, milleks on kulumõõtemuundur, temperatuuriandurite paar ja arvutusplakk.

Komplektne soojusarvesti – soojusarvesti, mille osad ei ole omavahel lahtivõetavad ega vahetatavad.

Kombineeritud soojusarvesti – soojusarvesti, mille osad on omavahel lahtivõetavad ja vahetatavad.



Sele 1.1 Komplektne soojusarvesti MEGACONTROL-CF50

Alakoost – soojusarvesti koosneb alakoostudest, milleks kulumõõtemuundur, temperatuuriandurite paar ja arvutusplakk.

Maksimaalne kulu Q_{max} või q_s või Q_s - kulu ülempiiri väärtus, mille juures kulumuundur töötab normdokumentides ettenähtud aja jooksul lubatud piirvõlgasid ületamata.

Nominaalkulu Q_n või q_p – kulu, mille väärtus võrdub $1/2 Q_{max}$, ja mille juures kulumuundur töötab lubatud piirvõlgasid ületamata.

Minimaalne kulu Q_{min} või q_i või Q_i – minimaalse kulu väärtus, mille juures kulumuundur töötab lubatud piirvõlgasid ületamata.

Suurim temperatuuride vahe $\Delta\theta_{max}$ – ülemine piir temperatuuride vahel, mis saab esineda temperatuurianduritel sisenevas ja tagastuvas torus, mille juures soojusarvesti töötab piirvigasid ületamata.

Vähim temperatuuride vahe $\Delta\theta_{min}$ – alumine piir temperatuuride vahel, mis saab esineda temperatuurianduritel sisenevas ja tagastuvas torus, mille juures soojusarvesti töötab piirvigasid ületamata.

1.2 Soojusarvesti arvutusploki taatlusmetoodika

Kombineeritud soojusarvestit taadeldakse tema koosseisu kuuluvate elementide kaupa. Arvutusploki taatlemisel kasutatakse meetodit, mille puhul temperatuuriandureid imiteeritakse etalontakistustega ja kulumuunduri signaali impulssidega. Lahutamatu arvutusploki ja temperatuuriandurite paari taatlemisel kulumuunduri signaali imiteeritakse impulssidega ning lahutamatu arvutusploki ja kulumõõtuuri taatlemisel temperatuuriandureid imiteeritakse etalontakistustega.

1.2.1 Mõõtevahendid

Arvutusploki taatlemisel temperatuuriandurite ja veehulga (-kulu) imiteerimiseks kasutatakse:

- etalontakistumõõte, takistusega (100...1200) Ω ;
- impulssgeneraatorit, impulsside sagedus 1 Hz...100kHz;
- impulsside loendurit, mõõtepiirkonnaga (10...100000) impulssi;
- arvutusploki taatlemiseks ettenähtud spetsiaalset taatlusseadet.

1.2.2 Väline vaatlus

Arvutusplokil peavad olema kõik displei sektsioonid töökorras ja nupud/lülitid peavad toimima.

1.2.3 Soojusarvestite taatlemine

Taatlemisele kuuluvate parameetrite valik.

Kui soojusarvestit käsitletakse kombineeritud mõõtevahendina, tuleb kulumuundurit, temperatuuriandureid ja arvutusploki taadelda eraldi.

Arvutusploki taatlemine

Integreeriva arvutusploki taatlemine põhineb taadeldava seadme näidu ja arvutatud teoreetilise soojushulga võrdlemisel. Temperatuuride väärtused ja veekulu imiteeritakse elektriliselt. Vajadusel asendatakse temperatuuriandurid täppistakistitega ja veekulu imiteeritakse impulssgeneraatoriga.

Taatlemiseks vajalikud seadmed valmistatakse ette vastavalt nende kasutamisujuhenditele. Integreeriv arvutusplokk ühendatakse takistussalvede ja impulssgeneraatoriga vastavalt arvutusploki kasutamisujuhendile. Esialgsel proovimisel tehakse kindlaks, et arvutusplokk registreeriks saabuvad impulsid, pealevoolu ja tagasivoolu andurite temperatuurid ning temperatuuride vahe.

Välise vaatluse ajal tehakse kindlaks, et:

- arvutusploki ei oleks mehaanilisi vigastusi;
- mittevõrgutoitega arvutusplokkidel oleks nõuetele vastavad toiteelemendid;
- numeraatorid ja/või digitaalnäidikud oleksid selgelt loetavad;
- digitaalnäidikutel oleksid loetavad kõik elemendid.

Impulssgeneraatori ja etalontakistusmõõtude kasutamisel võib läbiviidavate katsete kiirendamiseks kulu imiteerivate impulsside sageduse valida võimalikult suure, kuna üldjuhul integreeriva arvutusploki mõõtehälve ei olene saabuvate impulsside sagedusest. Jälgima peab ainult, et impulsside sagedus ei ületaks integreeriva arvutusploki poolt maksimaalset vastuvõetavat sagedust.

Keskkonna normaaltingimuse mõõtmise ajal:

- ümbritseva keskkonna temperatuur (5...30) °C, kusjuures temperatuuri muutus taatluse ajal ei tohi ületada 5 °C;
- õhurõhk 95 kPa kuni 105 kPa;
- suhteline õhuniiskus 30 % kuni 80 %.

1.2.4 Mõõtetulemuste töötlemine

Arvutusploki mõõtehälve δ_c arvutatakse seosest:

$$\delta_c = \frac{Q_c - Q_{ARV}}{Q_{ARV}} \cdot 100\%$$

(1.1)

kus Q_c – arvutusploki poolt mõõdetus soojushulk, MWh

Q_{ARV} – arvutatud soojushulk (soojushulga etalonväärtus), MWh

$$Q_{ARV} = M \cdot n \cdot \rho (h_s - h_r) = M \cdot n \cdot \rho \cdot \Delta h$$

(1.2)

kus M – taatlemisel plokki sisestatud elektrilise impulsi väärtus, (näit $2 \text{ m}^3/\text{imp}$),
 n – taatlemistsükli käigus plokki sisestatud impulsside arv,
 ρ – vee eritihedus taatluspunkti temperatuuril ja rõhul, t/m^3
 h_s ; h_r – küttevee erientalpia sisenevas (h_s) või tagastuvas (h_r) toorus taatluspunkti temperatuuril ja rõhul, kWh/t.

Korrutist $\rho \cdot (h_s - h_r)$ nimetatakse soojusteguriks K_s . Seega $Q_{ARV} = M \cdot n \cdot K_s$.

Vee eritihedus $\rho = \rho_s$, kui soojusarvesti komplekti kuuluv kulumuundur on paigaldatud sisenevale küttevee torule ning $\rho = \rho_r$, kui soojusarvesti komplekti kuulu kulumuundur on paigaldatud tagastuvale küttevee torule.

Suuruste ρ ja Δh allikateks on käsiraamatutes toodud väärtused.

Q_{ARV} arvutamisel kasutada ρ_s ; ρ_r ; h_s ja h_r tabuleeritud väärtusi [1].

1.2.5 Soojusarvesti lubatud piirvead

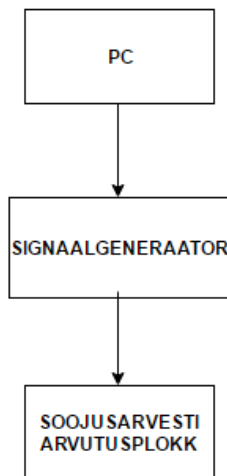
Arvutusploki lubatud piirvea väärtused avalduvad valemiga

$$E_c = \pm(0,5 + \Delta\theta_{min}/\Delta\theta) \tag{1.3}$$

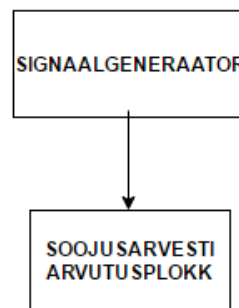
kus $\Delta\theta_{min}$ - vähim temperatuuride vahe

2. SEADME PROJEKTEERIMINE

Soojusarvesti arvutusploki vookulu signaali simuleerimiseks on vaja projekteerida elektrooniline seade. Komponentide valikul on vaja jälgida komponentide sobivust nõutud ülesande täitmiseks, nende hindu ja vastavust etteantud parameetritele. Allpool on toodud kaks lihtsustatud seadme ploki skeemi.



Sele 2.1 Seadme ploki skeem (1)



Sele 2.2 Seadme ploki skeem (2)

Idee seisneb selles, et asendada kallist ja keerukat impulssgeneraatori lihtsama ja odavamaga lahendusega, mis suudab genereerida impulsse ette antud parameetritega. Enne koodi kirjutamist on vaja leida sageduse piirväärtused.

Sageduse piirväärtuste leidmine

$$f = \frac{1}{t} = \frac{q}{M} \quad [2.1]$$

kus f – sagedus, Hz

t – aeg, s

q – küttevee kulu arvutusploki, m³/h

M – läbi kuumaveearvesti voolanud küttevee maht, m³/imp

$$f_{max} = \frac{25}{0,0025 \times 3600} = 2,8 \text{ Hz}$$

$$f_{min} = \frac{25}{0,25 \times 3600} = 0,028 \text{ Hz}$$

M ja q väärtused sain tellijast kätte.

Impulsside kuju	Nelinurkne
Impulsside arv	Kuni 1000
Min. sagedus (Hz)	0,028 Hz
Töötuskel (%)	50
Max. sagedus (Hz)	2,8 Hz

Sele 2.3 Genereeritava impulsside parameetrid

Pärast konsulteerimist tellijaga otsustasime, et seadme projekteerimisest kasutame esimest plokk skeemi, sest pärast saadud andmed on vaja integreerida temperatuuriandurist saadud tulemustega ja töödelda Excelis. Järgnevalt on tehtud ülevaade vajalikest komponentidest.

Signaalgeneraator

Signaali genereerimiseks on vaja impulss generaator. See võiks olla mikrokontroller, juhtarvuti või sagedus generaator. Ainsad nõuded on kompaktsus, odavus, kättesadavus ja võimalus kiiresti asendada, kui mingid komponendid rivist välja lähevad.

Relee

Arvutusplokki sisendite lühistamiseks on vaja releed.

3. ELEKTROONIKA

3.1 Signaalgeneraatori valik

Signaalgeneraatorid – signaalide allikad, mis väljastavad teatud suurusega (võimsusega või amplituudiga), teadaoleva sagedusega ning mitmesuguse lainekujuga väljundsignaali. On olemas erinevad liigid: siinussignaali generaatorid, funktsioonigeneraatorid, impulssgeneraatorid jne. Meid huvitab impulssgeneraator nelinurkimpulsside saamiseks [2].

3.1.1 Seadmete ülevaade

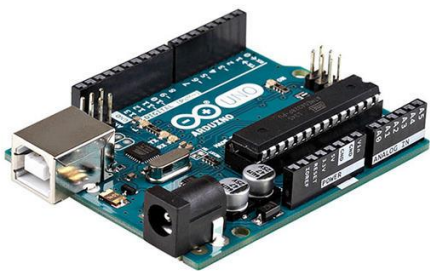
Signaalgeneraator



Sele 3.1 Signaalgeneraator Agilent 33210A

Seade genereerib nelinurk- ja siinussignaale sagedusega kuni 10 MHz. Ainult puudus on tema kõrge hind.

Mikrokontroller



Sele 3.2 Arduino Uno

Arduino Uno on lihtne mikrokontrolleri arendusplaat, millel on kõik vajalik olemas mikroprotsessorite maailma avastamiseks. Arduino riistvara koosneb lihtsast vaba riistvara disainiga trükiplaadist, millel on Atmel AVR protsessor, ning mugavamaks kasutamiseks välja toodud 16 erinevat sisend- ja väljundviigud. Arduino platvorm on teatavasti avatud lähtekoodiga ühe plaadi mikrokontroller. Selle

tarkvara koosneb standardsest programmeerimiskeele kompilaatorist ja mikrokontrolleris olevast algladimise haldurist. Programmeerimiseks kasutatav keel on sarnane C++ keelega, erinedes sellest mõne muudatuse ja lihtsustuse poolsest [3].

Allpool on toodud tabel tehniliste andmetega.

Tabel 3.3 Arduino Uno tehniline spetsifikatsioon [3]

Mikrokontroller	ATmega328
Tööpinge	7-12 V
Digitaalseid sisend/väljund kontakte	14 (6 PWM)
Analoog sisend kontakte	6
Alalisvoolutugevus kontakti kohta	20 mA
Mälu	32 KB Flash, 2KB SRAM, 1KB EEPROM
USB	1 USB 2.0
Mõõtmed	68,6 x 53,4 mm
Hind	18 €

Raspberry Pi



Sele 3.4 Raspberry Pi 3

Raspberry Pi on kompaktne, ühest trükkplaadist koosnev juhtarvuti, mida saab kasutada väga erinevates valdkondades – elektroonika juhtimiseks, robotikas, veebi sirvimiseks. Raspberry Pi kasutab peamiselt Linuxi kerneli põhjal töötavaid operatsioonisüsteeme, näiteks Raspbian, mis on eraldi Raspberry Pi'le optimeeritud Linux-i-põhine tasuta operatsioonisüsteem. Ja selle soovitatav programmeerimiskeel on Python [4].

Raspberry Pi 3 tehnilised andmed on välja toodud tabelis 3.5.

Tabel 3.5 Raspberry Pi 3 tehniline spetsifikatsioon [5]

Protsessor	Quad Cortex A53, 1.2 GHz
Mälu	1 GB SDRAM
Voolupesa	Micro USB 5V, 2 A
Ethernet	10/100 BaseT Ethernet
USB	4 USB 2.0
GPIO	40-pin
Mõõtmed	85 x 56 x 17 mm
Hind	46,20 €

3.1.2 Seadme valik

Raspberry puhul impulssi genereerimise stabiilsusele mõjub operatsioonsüsteemi viivitus, sest operatsioonsüsteem ei ole reaajas. Sellepärast Arduino Uno on eelis kui juhtarvuti Raspberry Pi.

Arduino eelised on suhteliselt väike hind; kättesaadavus – paljudel elektroonika kauplustest saab osta mikrokontrollerit ja tema mooduleid; programmeermise lihtsus ja palju kirjandust internetis. Puudusteks vaid tema töökindlus ja nõrga liidesed.

Signaalgeneraator ei sobi kõrgema hinna ja alterantiivlahenduste pärast.

Arduino eelis teise mikrokontrolleri ees on lihtne kasutajaliides, lihtne I/O ühendus ja suhteliselt madal hind. Peale selle on Arduino kohta leitav kirjanduse maht väga suur. Internetis on palju erinevaid juhendeid ja näiteid, mis võivad aidata programmi koostamiseks.

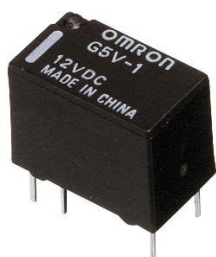
Asja üle kaaludes mina valisin mikrokontrolleri. Arduino Uno sobib kasutamiseks lihtsate ülesannete täitmiseks. Kuna impulsside genereerimine on just selline ülesanne oleks Arduino Uno valik põhjendatud.

3.2 Relee valik

Soojusarvesti sisendite lühistamiseks on vaja relee. Relee nõued: toitepinge 5V ja töösagedus peaks olema kuni 5 Hz. Leidsin 1A relee Omron G5V-1, mis vastab antud nõuedmetele.

Tabel 3.6 Omron G5V1-1 rele tehniline spetsifikatsioon [6]

Toitepinge	5 V
Voolutugevus	1 A
Tööpinge	125 AC / 24 DC
Töösagedus	kuni 10 Hz
Mõõtmed	12,5 x 7,5 x 10 mm
Hind	1,60 €

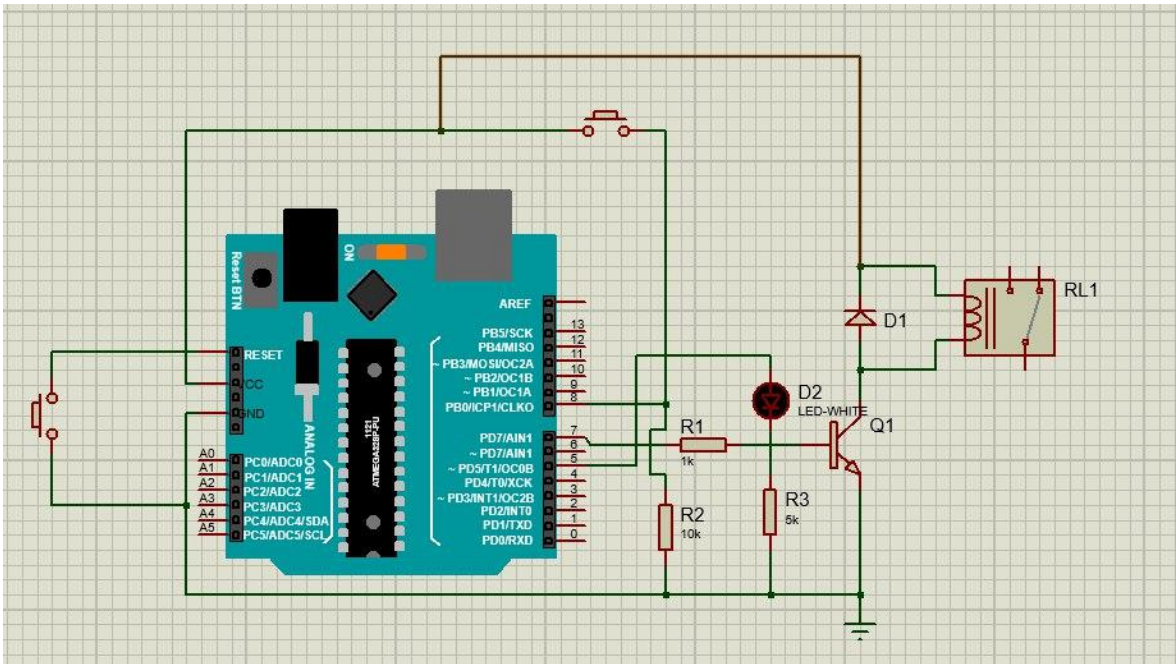


Sele 3.7 Relee Omron G5V1-1

3.3 Elektroonikaskeemi koostamine

Enne komponendi trükkplaadile paigaldamisest ja jootmist koostasid seadme elektronikaskeemi. Selleks kasutasid tarkvara Proteus 8.5.

Selle skeemis kasutatakse Arduino Uno-t rele juhtimiseks [7]. Mikrokontrolleri 7. -es (seitsmes) väljund on ühendatud transistori Q1 baasiga reele juhtimiseks. Tänu sellele antakse toite relele RL1 ja sees lühistatakse kontakte. Lisa impulsside nupp on ühendatud 8 väljundiga ja vasakpoolne on taaskäivituse nupp. Informatiivse valgusdiodi panin Arduino 5-le väljundile.



3.8 Vookulu signaali simuleerimiseks seadme elektroonikaskeem

4. JUHTIMINE

4.1 Täidetavad ülesanded

Impulsside genereerimine

Mikrokontroller peab genereerima oma väljundites impulssid antud sagedusetäpsusega 0,1% ja impulsside arvuga. Operaatoril peab olema võimalus sisestada sagedust ja impulsside arvu arvuti abil.

Lisa impulsid

Pärast seda, kui kõik antud impulssid lõppevad peab olema võimalus anda lisa impulssid. Selleks on vaja üksiku impulsi genereerimise nupp

Taaskäivitus

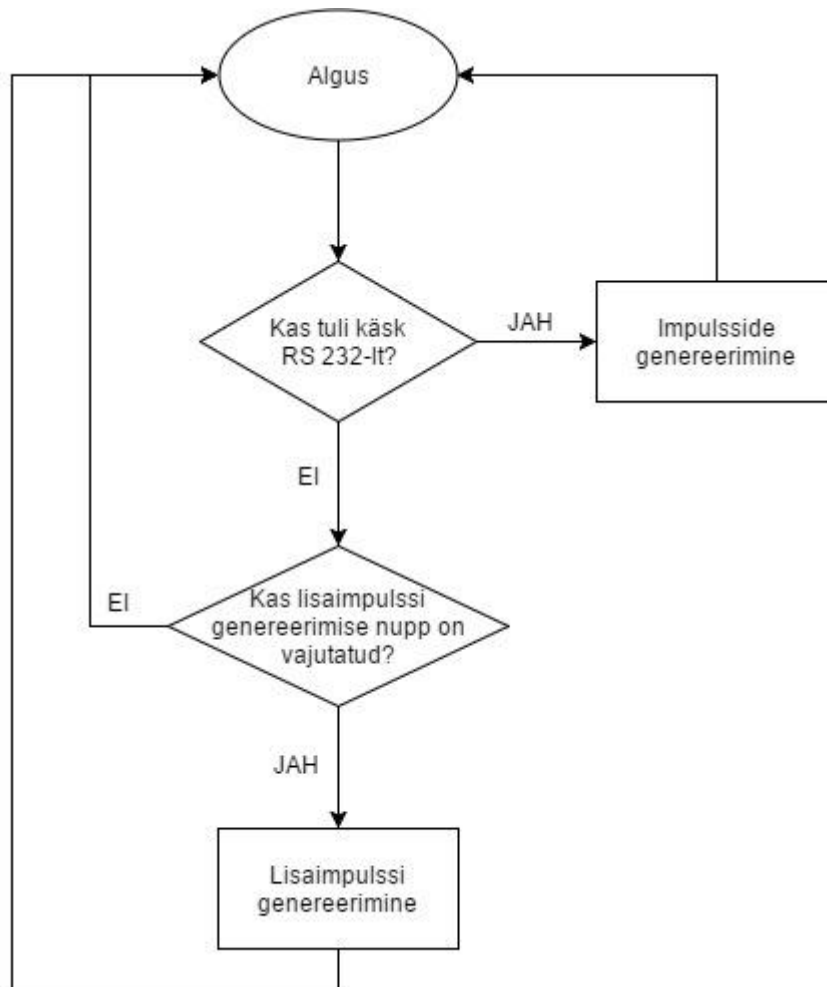
Valesti sisestamise juhul operaatoril peab olema võimalus peatuda impulsside ülekannet. Selleks on vaja taaskäivituse nupp.

4.2 Programmeerimiskeele valik

Arduino Uno programmeerimiseks kasutatakse Arduino Integrated Development Environment (IDE). Arduino programme saab kirjutada keeltes C/C++ kasutades ainult kaks funktsiooni `setup()` ja `loop()`. Oma varasema kogemuse põhjal kasutasin C++ keelt.

4.3 Programmi juhtimise põhimõte

Kõik eelmises punktis kirjeldatud ülesannete täitmine peab töötama loogilises järjekorras. Sellepärast koostas algoritmi, mis tagab programmi tõrgete vaba töö. (Sele 4.1)



Sele 4.1 Programmi andmevoo skeem

Kui tuli käsk RS 232-lt siis seade alustab impulsside genereerimist operaatori poolt sisestatud sagedusega ja arvuga. Ja kui operatoor vajutas nuppu siis genereerib üksik impulssi. Arduino Uno juhtimiskood on toodud lisa 1.

Koodi esimeses osas on deklareeritud kõik muutujad mis on vaja edasi kasutamiseks

Setup() funktsiooni kasutan muutujate initsialiseerimiseks ja väljundite määramiseks. Selline funktsioon käivitub ainult üks kord pärast toite andmist või mikrokontrolleri tühistamisest [8].

Peamine koodi osa asub loop() funktsioonis, mis kasutatakse Arduinojuhtimiseks. Kuna impulssi genereerimiseks ei soovinud kasutada katkestusi, pidin arvutama aja intervallid, mis kasutasin „if „ tsükli sees [9].

Aja intervallide arvutamine

Et täpsus oleks 0,1% piiri sees, valisin viivitust 16000 mikrosekundit või 0,016 sekundit. Sageduse genereerimiseks aja intervallid leidsin järgmiselt:

$$\frac{T}{2} = b \times 0,016 + a \times 0,016 \quad [4.1]$$

kus $T/2$ – poolperiood, s

b – aja intervallide täisarv vajalik antud sageduse genereerimiseks

a – aja intervallide jääk vajalik antud sageduse genereerimiseks

Pärast koodi laadimist mikrokontrolleri sisse, tegin kolm katset ajaintervallide mõõturiga. Tulemus oli järgmine:

Tellitud sagedus, Hz	Tulemus, Hz
2,8000	2,7809
0,056000	0,055621
0,028000	0,027810

Keskmine parand oli 0,685 %.

Parandi arvutamine

$$d = T/2 \times 0,685/100 \quad [4.2]$$

kus d – parand, s

$T/2$ – poolperiood, s

Lõplik seadme testimine on toodud punktis 6.

5. MEHAANIKA

Elektroonikale projekteerisin korpus, mis teeb seadme kompaktsemaks. Tellija poolt antud nõueteks olid kompaktsus ja kindlus.

Korpus on projekteeritud teatud põhimõtetest lähtuvalt:

- 1) Korpus peab kõik komponendid kindlalt enda külge siduma
- 2) Korpus peab mehaaniliselt vastupidav
- 3) Korpus peab tagama komponentide kompaktsus
- 4) Korpuses peavad olema avad komponentide ühendusportidele: USB, mikrokontrolleri välisliidised ja BNC pesad
- 5) Korpuse komponendid peavad olema kättesaadavad – on võimalik need tellida ja suhteliselt odavad

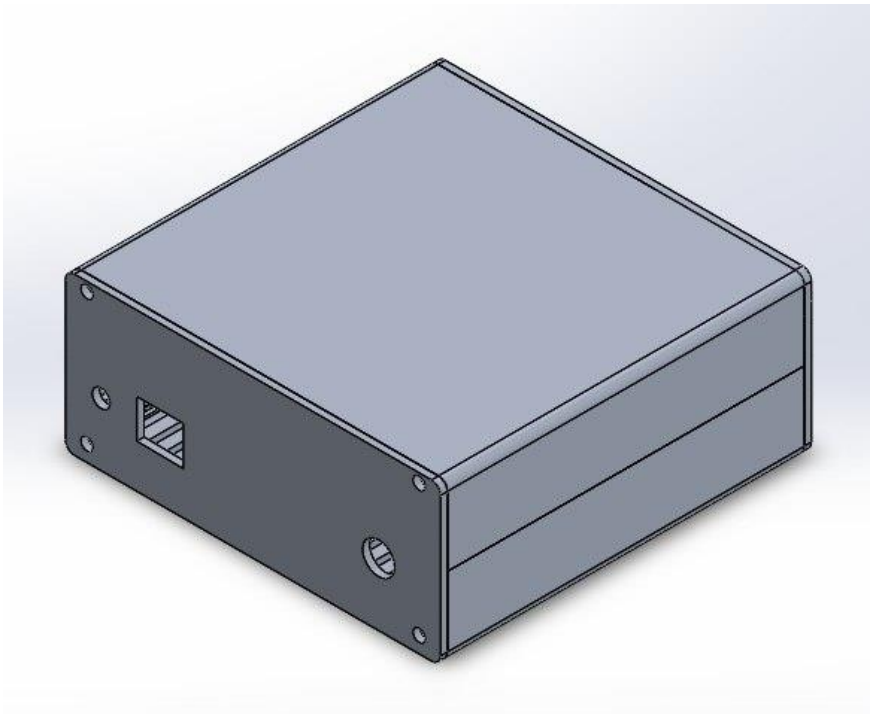
3D modelli projekteerimiseks kasutasin CAD tarkvarat Solidworks abil.

5.1 Korpuse projekteerimine

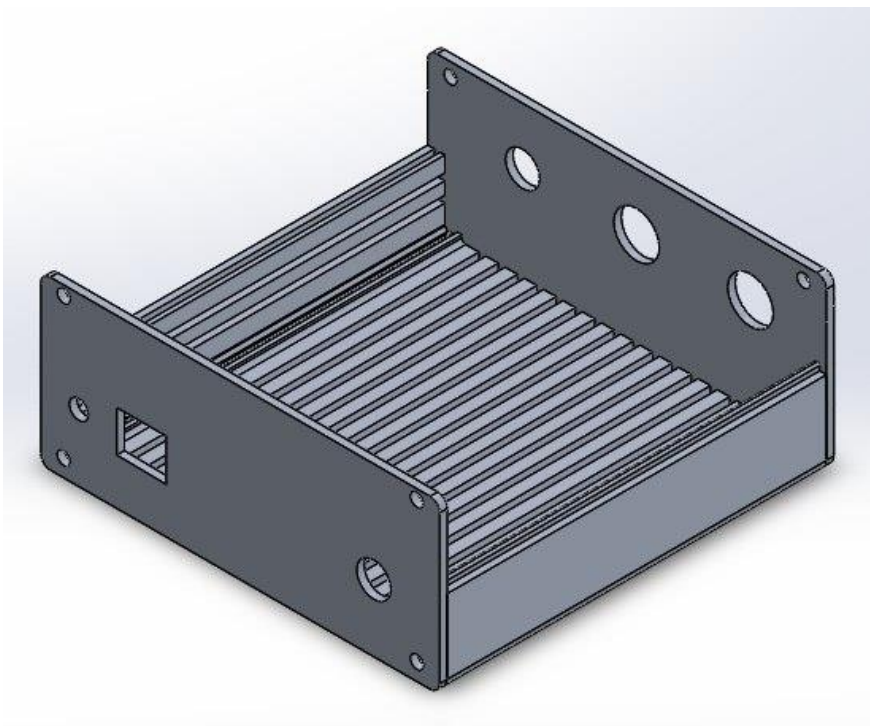
Seadme korpus koosneb kolmest osast: U-tüübi alusest (sele 5.5), 2-mm ees- ja tagaplaadist (sele 5.3 ja sele 5.4). Põrandale kinnitatakse prototüüpimisplaat ja üleval Arduino Uno koos releega ja teiste komponentidega. Kokkupanekul pannakse kaks alust kokku, taga ja ees plaadil on avad, mille kaudu kinnitatakse osad poltidega. Lisaks korpuse eesplaadil on Arduino USB ja LED'i avad, on ka olemas lisaimpulsside genereerimise roheline nupp. Tagaplaadil on kaks BNC pesa – esimene on relee jaoks, teine generaatori kalibreerimiseks ja taaskäivituse punane nupp. Korpuse osade joonised on toodud lisas 2.

Korpuse osad on tehtud alumiiniumist ja neid saab tellida kauplusest [10-11]. Kasutades tehnilisi jooniseid, mis olid kaupluse internetileheküljel koostasid 3 CAD mudelid ja panin neid kokku.

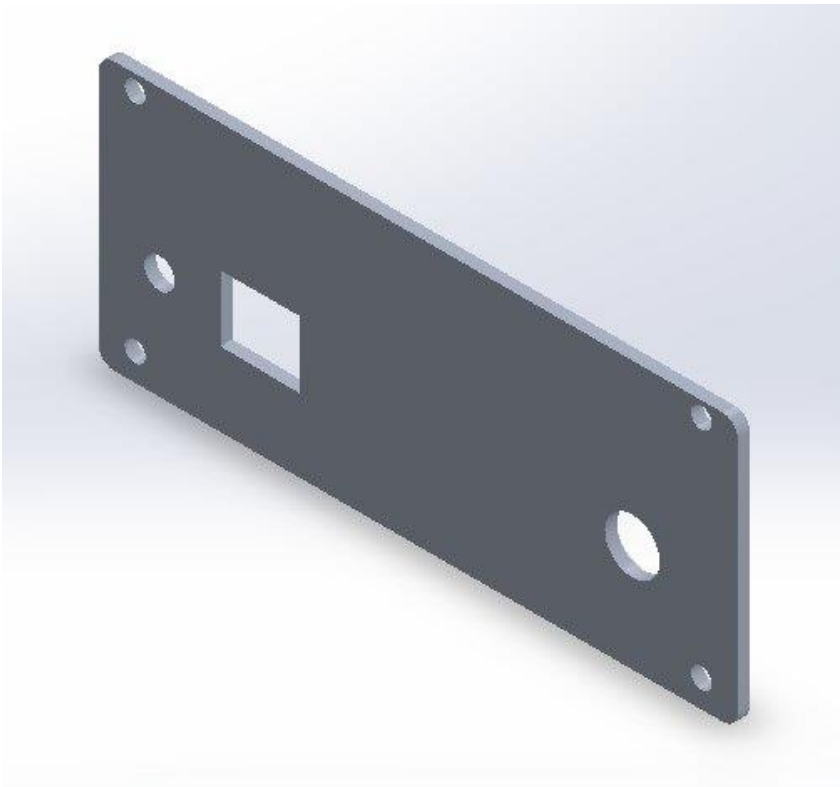
Allpool on toodud korpuse pildid.



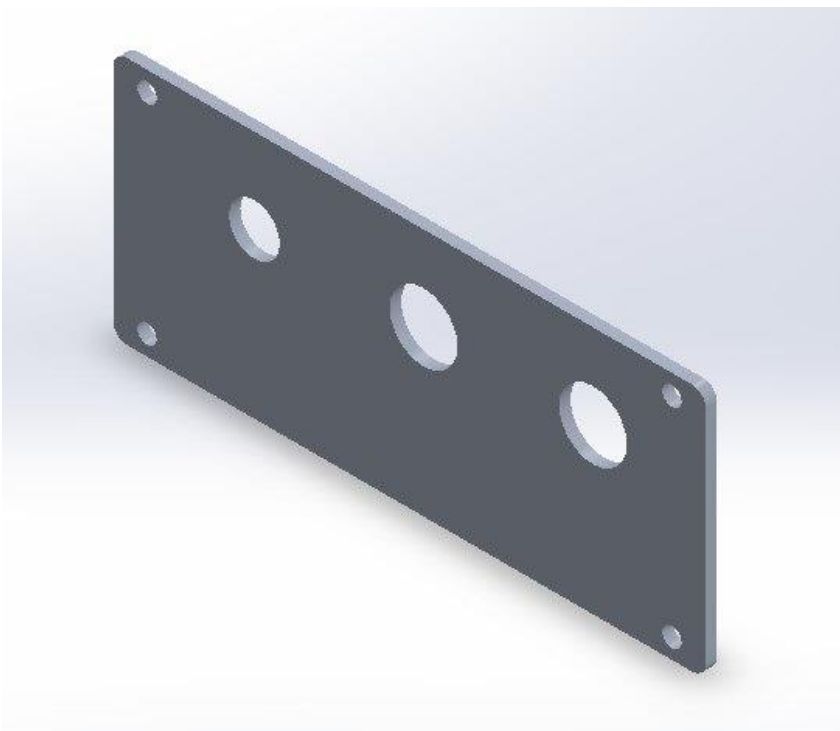
Sele 5.1 Kokkupandud korpus



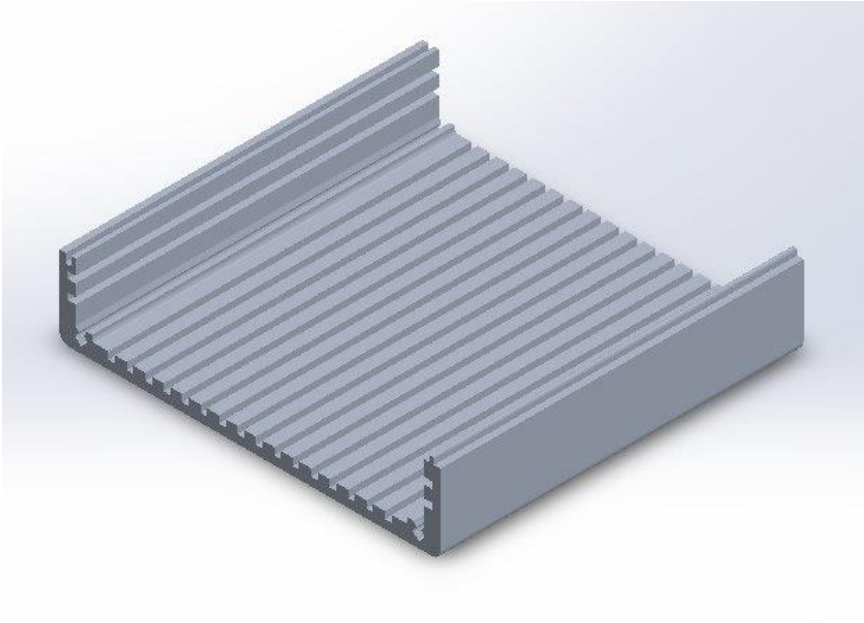
Sele 5.2 Korpuse ülemine vaade



Sele 5.3 Korpuse eesplaat



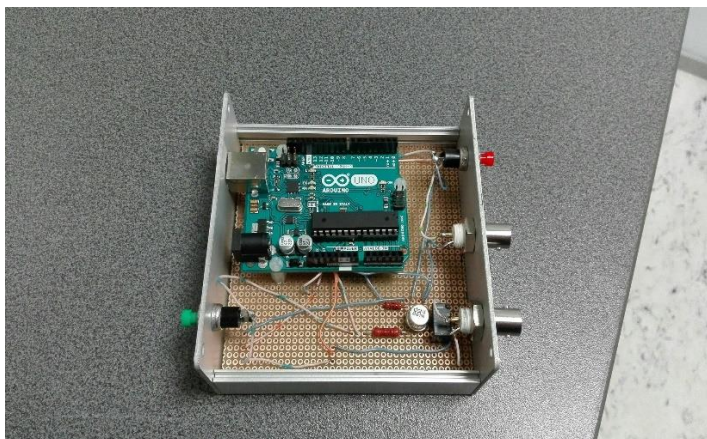
Sele 5.4 Korpuse tagaplaat



Sele 5.5 Korpuse alus

6. SEADME TESTIMINE

Pärast komponentide saabumist joodi neid prototüüpimisplaadile ja pandi seadme kokku (Sele 6.1).



Sele 6.1 Seadme kokkupanek

Sedme testimiseks kasutasin ajaintervallide mõõturit FLUKE PM6681. Esimesena kontrollisin sageduse väljundit, valides intervallist 0,028 Hz – 2,8 Hz viis punkti. Lõplik testimine näes välja niimodi:

Tabel 6.3 Sageduse mõõtmine

Tellitud sagedus, Hz	Tulemus, Hz	Parand, %
2,80000	2,79990	0,003
2,07900	2,07905	0,002
1,38600	1,38603	0,002
0,69300	0,69301	0,001
0,02800	0,02800	0,000

Saadud tulemused vastavad nõuetele.

Teisena kontrollisin impulsside arvu. Selleks võtsin kolm sagedust, mis kontrollisin varem ja iga sagedusega andsin impulssidvahemikus 10 kuni 1000. Testimiseks kasusin ajaintervallide mõõturit ja reele väljundil oli soojusarvesti. Iga impulssi kohta soojusarvesti näit muutub ühe võrra. Tulemused on toodud tabelis 6.4

Tabel 6.4 Impulsside arvu mõõtmine

Sagedus, Hz	Tellitud impulsside arv	Mõõteaeg, s	Ajaintervallide mõõturi näit	Näit soojusarvestist
2,8	10	3,6	10	10
	50	17,9	50	50
	100	35,7	100	100
	250	89,3	250	250
	500	178,6	500	500
	750	267,9	750	750
	1000	357,1	1000	1000
1,386	10	7,2	10	10
	50	36,7	50	50
	100	72,1	100	100
	250	180,4	250	250
	500	360,8	500	500
	750	541,1	750	750
0,028	10	357,1	10	10
	50	1785,7	50	50
Manuaalne	5		5	5
	10		10	10
	15		15	15

Tegin kokku kolm katset ja tulemus oli korratav.

6.1 Seadme majanduslik analüüs

Kogu seadme hind on kõigi komponentide summa.

Tabel 6.5 Komponentide maksumus

Komponent	Hind, €
Arduino Uno	18,00
Relay OMRON G5V-1	1,60
Diod DO-35 75 V	0,03
HF-Transistor TO-39 NPN	5,71
Rear panel DKOH02KOH02ME	9,94
Housing half 100 mm KOH2100ME	6,23

Hinnaarvutus:

$$18 + 1,6 + 0,03 + 5,71 + 9,94 * 2 + 6,23 * 2 = 57,68 (\text{€})$$

Seadme maksumuseks saan 57 euro ja 68 senti.

KOKKUVÖTTE

Käesoleva töö käigus projekteerisin seadme soojusarvesti arvutusploki vookulu signaali simuleerimiseks. Seadme on valmis ning laboris testitud.

Töö esimeses osas tutvusin soojusarvestiga. Uurisin, mida selline seade endast kujutab. Kuidas toimub soojusarvesti taatlemine, missugused seadmed ja mõõtevahendid selleks kasutatakse ja kuidas toimub tulemuste arvutamine.

Teises osas tegelesin seadme projekteerimisega. Kirjeldasin kuidas kujutab tellija endale seda ette. Loendasin põhilised genereeritava impulsside parameetrid, koostas plokki skeemi ja leidsin sageduse piirväärtused. Tegin lühikese ülevaade vajalikest komponentidest.

Kolmandas osas tegelesin seadme elektroonikaga. Impulsside genereerimiseks ja rele lülitamiseks valisin mikrokontroller Arduino Uno. Põhjalikult põhjendasin oma valikut, võrdlesin teiste lahendustega ja tegin ülevaade komponentide parameetritest. Tutvusin kuidas ühendada rele Arduino ja lõpuks koostas elektroonikaskeemi kasutades tarkvara Proteus 8.5.

Neljandas osas kirjeldasin programmi juhtimist. Püstitasin juhtimisega seotud ülesande ja realiseerin need. Esitasin programmi andmevoo skeemi. Arduino Uno programmeerimiskeeleks oli C++.

Viies osas disainisin seadmele korpuse. Korpus oli projekteeritud niimodi, et seadme elektroonika oli hästi kinnitatud ning kaitstud välismõjude eest. Korpus ka tagab seadme kompaktsuse. Kõik osad olid tellitud ja kokku pandud. Esitasin pilte projekteeritud korpuse CAD-mudelist.

Viimases osas uurisin seadme töö tulemusi. Selleks tegin erinevaid testimisi, kasutades aja intervallide mõõturit FLUKE PM 6681. Esiteks kontrollisin segaduse täpsust, et see ei ületaks 0,1 protsenti ja pärast impulsside arvu. Kokku tegin neli katset. Tulemused olid rahuldavad. Koostatud ka ehitatud seadme majanduslik analüüs.

Edasiseks arenduseks tuleks teha kasutajaliides. Et operaatoril oleks mugavam seadmega tööd teha. Seda võiks teha kasutades Pythoni programmeermiskeelt või LabView keskkonda. Samuti tuleks kasuks ka seadmele väikese kuvari lisamine, kust on näha sageduse ja impulsside arvu muutmist.

Seade töötab korrektselt ettemääratud parameetritega. Kõik töös püstitatud eesmärgid said saavutanud.

CONCLUSION

During this bachelor's thesis a device for flow rate signal simulation of heat meter calculator was designed. Device has been completed and tested in the laboratory.

The first part of the thesis explains how to use the heat meter calculator. To be specific, the thesis goes into what the device represents, how verification of the heat meter calculator works, what equipment and instruments are used for that and how results are calculated.

The second part of the thesis describes the device design. To be precise, the document gives information about customer requirements for the device, basic parameters of generated pulses. Also, a block diagram, frequency threshold values and overview of the necessary components are provided.

The third part gives information about the electronic part of the device. The paragraph provides information regarding why the Arduino Uno microcontroller was chosen for pulse generation and for relay switching. It also compares the microcontroller with other microcontrollers that could have been chosen and gives an overview of the parameters of the device components.

The fourth part explains the logic of the program (software) which is written in the C++ programming language. It also describes software requirements and how they were implemented. The data flow diagram of the program is provided.

The fifth part of thesis describes the mechanical design of the device, in other words, the device case. The case was designed so that the electronics of the device would be securely fixed and protected from external influences. Also, it makes the device compact. Pictures of the CAD-model of the designed case are presented.

In the last part of the thesis the results of the performed work are studied. Different tests were performed using FLUKE PM6681 Frequency Counter. To be specific, the frequency output of the device has been evaluated. The frequency error does not exceed 0,1 percent. The number of pulses is checked as well. The test suite was tested four times and the results of the performed tests were satisfactory. Also, the economic analysis of the built device is provided.

Further development consists of building a user interface. It is needed so that it would be more convenient to work with the device. It could be done by using Python or the LabView programming environment. A small display on the device also would be a good addition as it could show changes in the frequency and the number of impulses.

The device works correctly with assigned parameters. All goals that were set the beginning of the thesis were accomplished.

KASUTATUD KIRJANDUS

- 1 Soojusarvesti taatlusmetoodika (Dokumendi omab AS Metrosert)
- 2 Signaalgeneraatorid <http://test.kose.ee/cb/materjalid/sig-gen.pdf>
- 3 Arduino Uno kirjeldus, tehnilised andmed ja hind
<http://www.ittgroup.ee/et/arduino-arendusplaadid/85-arduino-uno.html>
- 4 Raspberry Pi https://et.wikipedia.org/wiki/Raspberry_Pi
- 5 Raspberry Pi tehnilised andmed ja hind
http://www.ittgroup.ee/et/raspberry-pi-jt-miniavutid/721-raspberry-pi-3-mudel-b.html?search_query=raspberry+pi+3&results=57
- 6 Relee Omron G5V1-1 andmed
https://www.omron.com/ecb/products/pry/111/g5v_1.html
- 7 Relee juhtimine Arduino abil
<http://arduino-diy.com/arduino-rele>
- 8 Setup() funktsioon
<https://www.arduino.cc/en/reference/setup>
- 9 Loop() funktsioon
<https://www.arduino.cc/en/reference/loop>
- 10 Front/rear panel DKOH02KOH02ME
https://www.elfadistelec.ee/et/front-rear-panel-fischer-elektronik-dkoh02koh02me/p/15069920?q=*&filter_Category3=Plastist+ja+metallist+kaitse%C3%BCmbrised&filter_categoryCodePathROOT%2Fcat-L1D_379509=cat-L2D_379531&filter_Category5=Profile+housing%2C+clamshell+type&filter_Category4=Profiil-kaitse%C3%BCmbrised%2C+p%C3%B6%C3%B6rdluugi+t%C3%BC%C3%BCpi&filter_Buyable=1&filter_categoryCodePathROOT%2Fcat-L1D_379509%2Fcat-L2D_379531=cat-L3D_525402&filter_categoryCodePathROOT=cat-L1D_379509&page=2&origPos=2&origPageSize=10&simi=98.8
- 11 Housing half KOH2100ME
https://www.elfadistelec.ee/et/housing-half-100-mm-fischer-elektronik-koh2100me/p/15069890?q=*&filter_Category3=Plastist+ja+metallist+kaitse%C3%BCmbrised&filter_categoryCodePathROOT%2Fcat-L1D_379509=cat-L2D_379531&filter_Category5=Profile+housing%2C+clamshell+type&filter_Category4=Profiil-kaitse%C3%BCmbrised%2C+p%C3%B6%C3%B6rdluugi+t%C3%BC%C3%BCpi&filter_Buyable=1&filter_categoryCodePathROOT%2Fcat-L1D_379509%2Fcat-L2D_379531=cat-L3D_525402&filter_categoryCodePathROOT=cat-L1D_379509&page=17&origPos=17&origPageSize=10&simi=98.8

LISAD

Lisa 1 – Arduino Uno juhtimiskood

```
#include <math.h>
double a,T,b;// T - poolperiood parandiga, a - jaotise jääk, b - täis arv
double d; // d - parand
double g; // poolperiood parandita
double f;// sagedus
int i;//
int counter; // loendur
int c; //impulsside arv
boolean Start = false;

int commaIndex, secondCommaIndex, incomingByte;
String myString;
String str_firstValue;
String str_secondValue;
int i_firstValue;
double d_secondValue;
boolean stringComplete = false;
String inputString = "";

const int pinButton = 8;
int stateButton;

void setup() {
  // put your setup code here, to run once:
  DDRD = B10100000;
  pinMode(pinButton, INPUT);
  Serial.begin(9600);
  inputString.reserve(200);
}

void loop()
{
  while (Serial.available()) {
    char inChar = (char)Serial.read();
```

```

inputString += inChar;
if (inChar == '\n') {
    stringComplete = true;
}
}

if (stringComplete)
    {Serial.println("Echo oli " + inputString);

    commaIndex= inputString.indexOf(',');
    secondCommaIndex = inputString.indexOf(',', commaIndex + 1);
    str_firstValue = inputString.substring(0, commaIndex);
    str_secondValue = inputString.substring(commaIndex + 1, secondCommaIndex);
    i_firstValue = str_firstValue.toInt();
    d_secondValue = str_secondValue.toDouble();
    stringComplete = false;
    inputString = "";
    Start = true;

    c = i_firstValue;
    f = d_secondValue;
    g = 1/(2*f);
    d = g * 0.685 / 100;
    T = g - d;
    a = modf(T/0.016, &b);
    }

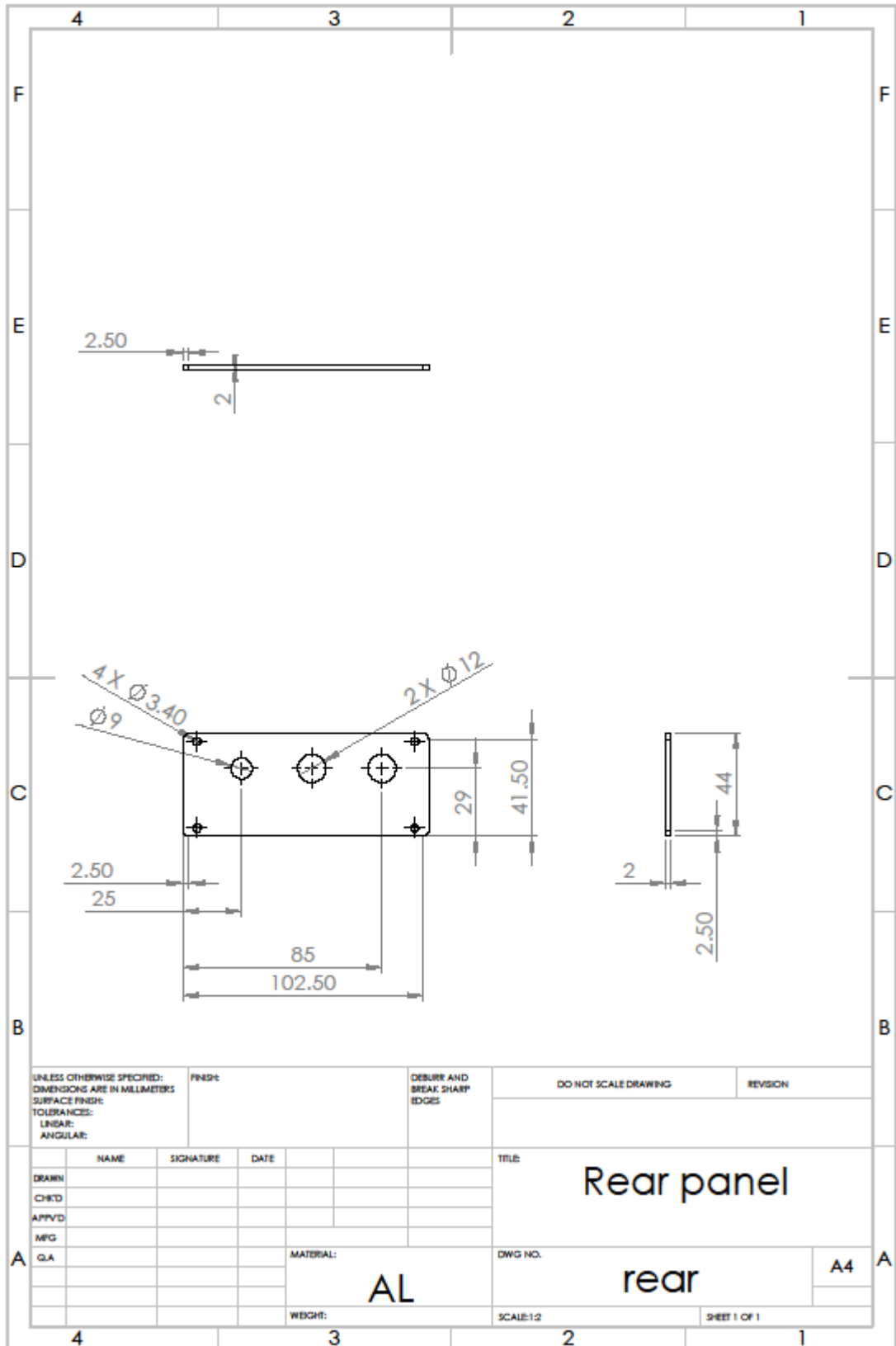
stateButton = digitalRead(pinButton);
if(stateButton == 1)
{
    Serial.println("bt");

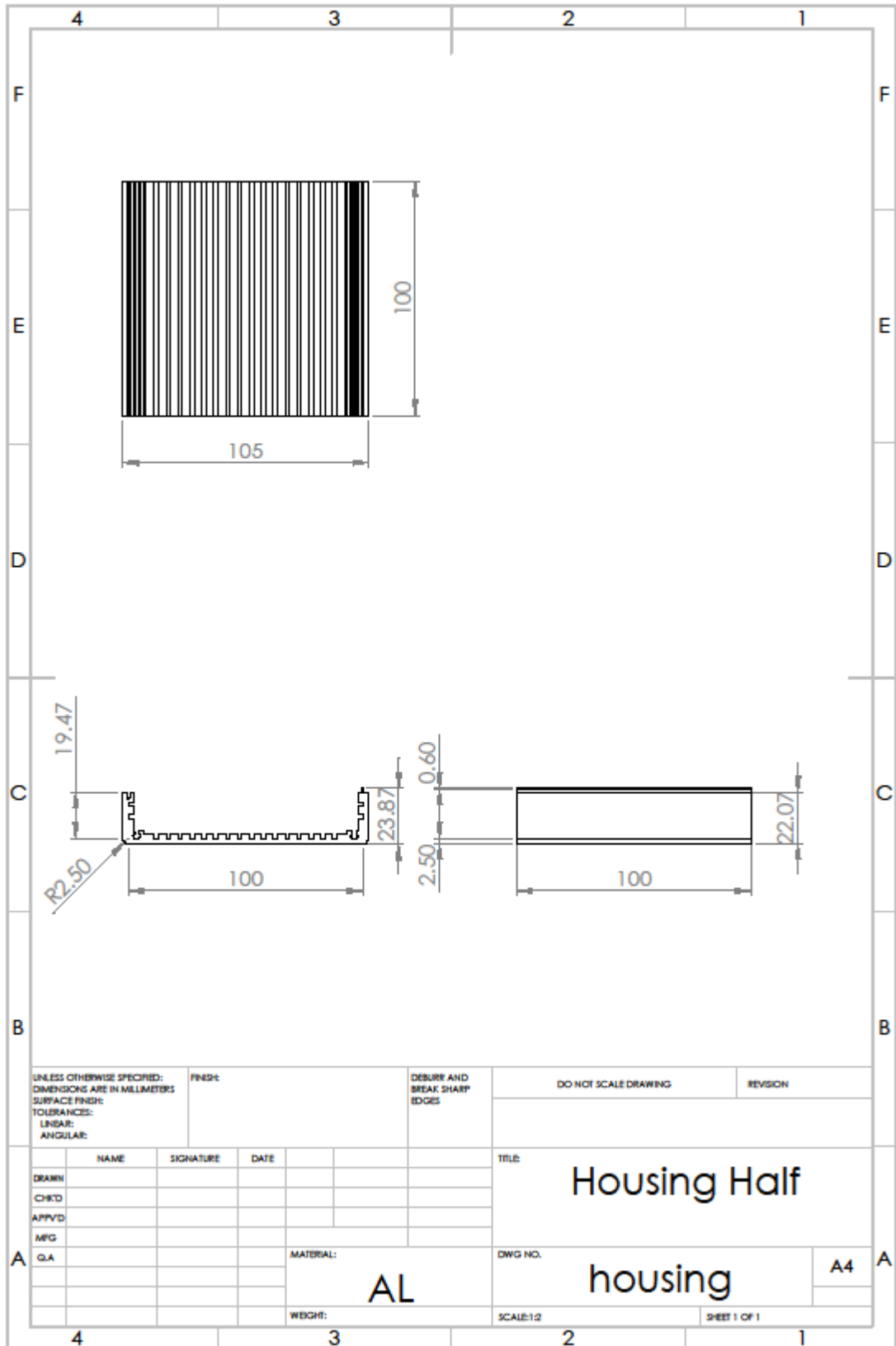
    c = 1;
    f = 2;
    g = 1/(2*f);
    d = g * 0.685 / 100;
    T = g - d;
    a = modf(T/0.016, &b);
}

```

```
Start = true;
}

if(Start == true)
{
for(counter=1; counter<=c; counter++)
{
PORTD = B10100000;
for(i=1; i<=b; i++)
{
delayMicroseconds(16000);
}
delayMicroseconds(a*16000);
PORTD = B00000000;
for(i=1; i<=b; i++)
{
delayMicroseconds(16000);
}
delayMicroseconds(a*16000);
}
Start = false;
Serial.println("ok");
delay(500);
}
}
```



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:				FINISH:	DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION
NAME	SIGNATURE	DATE				TITLE: Housing Half	
DRAWN						DWG NO. housing	
CHK'D						A4	
APP'VD						SCALE: 1:2	
MFG						SHEET 1 OF 1	
Q.A.					MATERIAL: AL	WEIGHT:	

Lisa 3 – Pildid komplekteeritud seadmest



Sele lisa 3.1 Komplekteeritud seadme eestvaade



Sele lisa 3.2 Komplekteeritud seadme tagavaade