

RASPBERRY PI BAASIL LIINIKAAMERA RAKENDAMINE TÖÖSTUSES

RASPBERRY PI BASED LINE CAMERA IMPLEMENTATION IN INDUSTRY

RAKENDUSKÕRGHARIDUSEÕPPE LÕPUTÖÖ

Üliõpilane: Agris Reppo
Üliõpilaskood: 165981NDFR
Juhendaja: Ago Rootsi, lektor
Kaasjuhendajad: Kristjan Piir, Kaarel Adamson

Tartu 2020

(Tiitellehe pöördel)

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.

"....." 202.....

Autor:

/ allkiri /

Töö vastab bakalaureusetöö/magistritööle esitatud nõuetele

"....." 202.....

Juhendaja:

/ allkiri /

Kaitsmisele lubatud

"....."202... .

Kaitsmiskomisjoni esimees

/ nimi ja allkiri /

LÕPUTÖÖ ÜLESANNE

Üliõpilane: Agris Reppo, 165981NDFR

Õppekava, peaariala: NDFR14/15 - Küberfüüsikaline süsteemitehnika

Juhendaja(d): Lektor, Ago Rootsi, 620 4808

Lõputöö teema:

Raspberry PI baasil liinikaamera rakendamine tööstuses

Raspberry Pi based line camera implementation in industry

Lõputöö põhieesmärgid:

1. Leida ülesande püstitusele vastav liinikaamera rakendus
2. Ehitada liinikaamera
3. Analüüsida lahenduse tulemuslikkust ja leida arenduskohad

Lõputöö etapid ja ajakava:

Nr	Ülesande kirjeldus	Tähtaeg
1.	Tutvumine erinevate liinikaamera rakendustega	15.03
2.	Välja töötada esialgne liinikaamera lahendus	15.04
3.	Arendada ja analüüsida loodud liinikaamerat	31.04

Töö keel: eesti keel

Lõputöö esitamise tähtaeg: 26.05.2020

Üliõpilane: Agris Reppo ".....".....202....a
/allkiri/

Juhendaja: Ago Rootsi ".....".....202....a
/allkiri/

SISUKORD

LÕPUTÖÖ ÜLESANNE	Tõrge! Järjehoidjat pole määratletud.
EESSÕNA	Tõrge! Järjehoidjat pole määratletud.
Lühendite ja tähiste loetelu.....	Tõrge! Järjehoidjat pole määratletud.
SISSEJUHATUS	Tõrge! Järjehoidjat pole määratletud.
Seotud tööd	Tõrge! Järjehoidjat pole määratletud.
2 KAVANDAMINE.....	Tõrge! Järjehoidjat pole määratletud.
2.1 Kasutaja- ja süsteeminõuded	Tõrge! Järjehoidjat pole määratletud.
2.2 Riistvara.....	Tõrge! Järjehoidjat pole määratletud.
2.2.1 Kaamera mooduli valik	Tõrge! Järjehoidjat pole määratletud.
2.2.2 Juhtseadmevalik.....	Tõrge! Järjehoidjat pole määratletud.
2.2.3 Liikumise tuvastamine.....	Tõrge! Järjehoidjat pole määratletud.
2.2.4 Muud komponendid.....	Tõrge! Järjehoidjat pole määratletud.
2.2.5 Komponentide karpi paigutamine.....	Tõrge! Järjehoidjat pole määratletud.
2.2.6 Riistvara maksumus.....	Tõrge! Järjehoidjat pole määratletud.
2.3 Tarkvara	Tõrge! Järjehoidjat pole määratletud.
2.3.1 Operatsioonisüsteemi valik.....	Tõrge! Järjehoidjat pole määratletud.
2.3.2 Pildi parameetrid.....	Tõrge! Järjehoidjat pole määratletud.
2.3.3 Programmi ülesehitus	Tõrge! Järjehoidjat pole määratletud.
2.3.4 Raspberry Pi konfigureerimine	Tõrge! Järjehoidjat pole määratletud.
2.3.5 Võimalikud edasiarendused.....	Tõrge! Järjehoidjat pole määratletud.
3 SÜSTEEMI TESTIMINE.....	Tõrge! Järjehoidjat pole määratletud.
KOKKUVÕTE	Tõrge! Järjehoidjat pole määratletud.
SUMMARY	Tõrge! Järjehoidjat pole määratletud.
KASUTATUD KIRJANDUSE LOETELU	Tõrge! Järjehoidjat pole määratletud.

EESSÕNA

Lõputöö teema algatajaks on Enics Eesti AS. Uurimustöö on osa ettevõtte arendustegevusest. Ettevõtetepoolsed juhendajad on tehnoloogia osakonna juht Kristjan Piir ja THA protsessi insener Kaarel Adamson. Koolipoolne juhendaja on Ago Rootsi. Täna kõik, kes on hoidnud motivatsiooni kõrgel ja julgustanud uusi sihte võtma. Abiks on olnud kõik infokillud ja suunised, et teha töö käigus õigeid valikuid.

Lühendite ja tähiste loetelu

AOI – automaatne optiline kontroll (ingl k automated optical inspection)

MES tool – rakendus, millega jälgitakse toote liikumist tootmises. Fikseeritakse, millised etapid toode läbis, kaua see aega võttis, kes tegeles tootega ja palju muud tootmisprotsessiga seonduvat informatsiooni.

RTC – moodul, mille eesmärk on hoida õiget kuupäeva ja kellaega volukatkestuse korral (ingl k *real time clock*)

I²C – on mitme võimaliku ülemaga jadasiin

VNC – on graafiline kasutajaliides, mis jagab teise arvuti kuva ja mille abil on võimalik arvutit kaugjuhtida (ingl k *virtual network computing*)

SSH – võimaldab juhtseadme kaugjuhtimist, kuid käske saab anda ainult käsureal (ingl k *secure shell*)

THT – elektrooniliste komponentide paigaldamise meetod trükkplaadile, komponente paigaldatakse trükkplaadile läbi avade, mis on sinna eelnevalt loodud (ingl k *through hole technology*)

SMA – elektrooniliste komponentide paigaldamise meetod trükkplaadile, komponendid paigaldatakse otse trükkplaadi pinnale (ingl k *surface mount technology*)

SMB – suhtlusprotkoll, mida kasutatakse jagatud juurdepääsuga failide jagamiseks võrgus. (ingl k *server message block*)

CIFS – suhtlusprotokoll, mida kasutatakse jagatud juurdepääsuga failide jagamiseks võrgus. CIFS on üks osa SMB suhtlusprotokollist. (ingl k *common Internet file system*)

SISSEJUHATUS

Tihe konkurents sunnib ettevõtjaid arendama ja kasutusele võtma uusi tehnoloogiaid. Tehnoloogiate sihipärane ja massiline rakendamine tagab konkurentsisis püsimise, kui ka kiirendab ettevõtte arengut.

Uurimustöö ülesanne on võtta kasutusele uus tehnoloogia Enics Eesti AS-s (edaspidi nimetatud Ettevõtte). Ülesanne on püstitatud Enics Eesti AS poolt ja on mõeldud Ettevõtte tegevuse edendamiseks. Ettevõtte toodab tööstuselektronikat Elvas.

Trükkplaatide defekti(de) juurpõhjuse tuvastamiseks on Ettevõttel vaja andmebaasi piltidest, mille põhjal saaks tuvastada, millises etapis tootel ilmnes defekt. Tehases on palju erinevaid protsesse ja tihti juurpõhjuse tuvastamine on keeruline, sest puudub tõestusmaterjal. Loodav süsteem on liinikaamera, mis pildistab tooteid erinevates etappides ja seob pildid tootekoodidega.

Ettevõttes kasutatakse MES tool'i, mis ei lase töötajal tootega järgmist etappi alustada, kui toodet pole eelnevas etapis lõpetatud. Igal tootel on oma vöötkood, mis skaneeritakse MES andmebaasi igas etapis. Selleks peab töötaja skaneerima tootekoodi ja märkima, et etapp on edukalt läbitud.

Esimene liinikaamera prototüüp paigaldatakse käsiladumise liini järele, enne lainejoote konveierit. Selle abil on võimalik tuvastada, kas käsiladumises on laotud kõik komponendid trükkplaadile. Samuti on võimalik tuvastada, kui näiteks lainejoote masinas komponent kukub tinapatta või kas polaarne komponent on paigutatud õiget pidi. Kui hilisemas etapis avastatakse defektne toode, siis on võimalik kontrollida, millises etapis defekt esines.

Pikemas perspektiivis on võimalik selline lahendus luua mitmele käsiladumise ja lakiliinile, kui ka arendada edasi kasvõi mõned AOI elemendid, et tuvastada tüüp vigu.

Töö käigus viiakse läbi järgnevad tegevused:

- Leitakse ja analüüsitakse olemasolevaid liinikaamera lahendusi.
- Leitakse sobilikud riistvaralised lahendused.
- Luuakse juhtseadme juhtimisprogramm, kirjeldatakse selle loogikat.
- Kirjeldatakse, kuidas juhtseade on konfigureeritud ja milliseid vigu esines selle käigus.
- Paigaldatakse liinikaamera tooteliini kohale.
- Luuakse fotodest koosnev andmebaas ja kirjeldatakse selle haldamist.
- Kirjeldatakse, kuidas teha kettatõmmist ja miks luua kettatõmmiseid. Kuidas taasluua lahendus teisele tooteliinile lihtsalt viisil?

Seotud tööd

Jan Toodre lõputöö „Raspberry Pi-l põhinev liikumistuvastusega videojälgimissüsteem“, mille eesmärk on luua videovalvesüsteem, millega ta saaks maakodu jälgida, kui ta on kodust eemal. Töö eesmärgiks on luua täpselt oma vajadustele vastav videojälgimissüsteem võimalikult väikeste kulutustega. Süsteem peab suutma salvestada pilte ja videoid, tuvastada liikumist, jälgida kaameraid reaalsajas ja võimaldama kasutajal vaadata pilte ja videoid tagantjärgi. Oma töös ta loob Raspberry Pi baasil oleva süsteemi, millega on võimalik jälgida kodu ümber toimuvat. Tema töös võrreldakse erinevate toiteallikate kasutamist Raspberry Pi tarbeks (näiteks päikesepaneelidel põhinev toiteallikas). [1]

Sarnasusteks minu lõputöö ja Jan Toodre töö vahel on: Raspberry Pi ja kaamera rakendamine, materjali salvestamine kõvakettale, riistvaraline ülesehitus, komponentide valiku küsimus.

Erinevusteks on see, et Jan Toodre töö käsitleb videode salvestamist, serveri haldamist ja Raspberry Pi-le toitemooduli valikut. Minu töö puhul on vaja teha kõrgresolutsiooniga üksikpilte, seega kaadriedastuskiirus ei ole oluline. Toiteallikaga seotud probleeme pole vajalik käsitleda, sest toide on alati saadaval.

2 KAVANDAMINE

Selles peatükis käsitlen süsteemi kavandamise küsimusi ja seletan lahti, mille põhjal valiti riistvaralised lahendused ja kuidas on tehtud otsuseid tarkvaraliselt.

Töö eesmärk on teha automaatselt pildid konveieril liikuvatest trükkplaatidest enne seda kui need lähevad lainejoote ahju. Selleks on vaja juhtseadet, kaamerat ja trükkplaadi tuvastamiseks sensorit. Tuleb luua programm, millega juhitakse juhtseadme tööd ehk programm, mis tuvastab liinil oleva trükkplaadi ja seejärel pildistab seda. Samuti tuleb luua andmebaas piltidest, kus pildid on jagatud kuupäeva ja kellaja järgi kaustadesse. Piltide abil on võimalik tuvastada, kus on tekkinud defekt trükkplaadile. Süsteemi on lihtne taasluua teistele tooteliinidele, näiteks lakiliinile. Tehase sisevõrgus on võimalik töötajatel piltidele ligi pääseda.

Konveieri tööd selles töös ei käsitle, see on juba eelnevalt lahendatud. Liinikaamera juhtseade ei suhtle konveieriga, vaid juhtseade toimib iseseisvalt. Toote pildid tehakse liikumise pealt. Kaamera fookus on fikseeritud kaugusega, samuti säriaeg on fikseeritud suurus, see tagab piltide ühtlase kvaliteedi.

2.1 Kasutaja- ja süsteeminõuded

Kasutajanõuded kirjeldavad, millised ootused on kliendil süsteemi suhtes. Antud juhul on kliendiks Enics Eesti AS. Kasutajanõuded ei pea tingimata kirjeldama, kuidas süsteem toimib, vaid kasutajanõuete abil kirjeldatakse, millist väljundit süsteemilt oodatakse.

- Ligipääs trükkplaatidest tehtud piltidele, mis asuvad andmebaasis. Andmebaas on pildid toodetest, mis on jaotatud kaustadesse kuupäeva alusel.
- Pildid on asjakohased ja õigel ajal pildistatud. Õigel ajal pildistatud ehk toode asub tervikuna kaadris.
- Piltidel olev kellaaeg ja kuupäev on korrektne
- Pildid on jaotatud kuupäevalistesse kaustadesse
- Pildid on arusaadavad ja nendelt on võimalik näha komponente ja nende polaarsusi

Süsteeminõuete kaudu kirjeldatakse, millist süsteemi looma hakatakse ja millised peavad olema süsteemi omadused, et saavutada soovitud tulemus. Süsteeminõuded:

- Komplekti hind on võimalikult odav.
- Kõrge resolutsiooniga kaamera, mille piltidelt on võimalik tuvastada komponentide polaarsusi.
- Juhtseade on suuteline töötlemas kõrge resolutsiooniga pilte, mis vastavad ülaltoodud kriteeriumile
- Juhtseadme juhtimisprogramm käivitub automaatselt juhtseadme käivitamisel, programm töötab sekkumiseta.
- Pildistab õigel ajahetkel ehk kui toode asub tervikuna kaadris.
- Wi-Fi kaudu ühendus võrgukettaga
- Juhtseadmel on katkematu toide
- Juhtseadmel on õige kellaeg ka pärast toitekatkestust
- Süsteem on lihtsasti taasloodav uuele juhtseadmele

2.2 Riistvara

2.2.1 Kaamera mooduli valik

Kaamera mooduli valik on riistvara juures üks esimesi valikuid, sest kaamera moodulilt saadakse väljund – pilt. Pilt peab olema selge ja elektroonikakomponendid peavad olema pildilt tuvastatavad. Samuti on oluline, et pildilt on näha polaarseste komponentide polaarsusi. Kaamerat ja juhtseadet peab valima koos, sest kõik kaamerad ja juhtseadmed ühildu omavahel. Antud ülesande tarbeks on vaja, et kaameral on kõrgeresolutsioon, kuid tal ei pea olema automaatset fokuseerimist ega olema suur kaadrisagedus. Sobilikuks kaameramooduliks osutus Arducam IMX298, millel on Sony 1/2.8 tolli CMOS sensor, 16 MP resolutsiooniga. Kasutatakse seda moodulit ka nutitelefonides ja mänguasjades. Kaamera ühendatakse juhtseadme külge ribakaabli abil. Mooduli hinnaks on 55 eurot. [4]

2.2.2 Juhtseadmevalik

Juhtseadme valikul tuli välja selgitada, milleks juhtseadet täpselt kasutama hakatakse. Kui kiire peab olema keskprotsessor ja graafikaprotsessor? Kui palju on vaja vahemälu? Kas projekt vajab Internetiühendust? Oluline on ka juhtseadme mõõtmed ja kaal. Kui palju on vaja sisendeid ja palju väljundeid?[2]

Minu ülesande jaoks piisab ühetuumalisest protsessorist, kuna juhtseade tegeleb ühe primaarse ülesandega korraga. Vahemälu võiks olla 2 gigabaiti, kuna piltide pidev edastamine ja töötlemine vajab teatud koguses vahemälu. Tuleb arvestada väikese varuga, sest juhtseade peab olema töökindel. Graafikaprotsessor peab olema suuteline töötlemas pilte 16 MPx resolutsiooniga. Samuti on vajalik Interneti-ühendus Wi-Fi näol. Ülesande sisendid tulevad kaamera moodulist ja ultraheliandurist. Antud töö tarbeks ei ole vaja kasutada juhtseadme väljundeid. Väljundiks võib nimetada pilti, mis jäädvustatakse fotosensori abil.

Ülaltoodud andmete põhjal valisin juhtseadmeks Raspberry PI 4 B 1.5 GHz neljatuumalise protsessoriga, 2 GB vahemälu ja 500 MHz VideoCore VI graafikakaardiga. Mikroarvuti hinnaks tuleb circa 60€.

Otsisin ka teisi juhtseadmeid, kuid ei leidnud ühtegi paremini sobivat juhtseadet, kui Raspberry Pi. Arduino Uno ei oleks sobinud, sest juhtseade pole piisavalt võimekas, kui ka vajaks laienduskaarti, et sellel oleks Wi-Fi ühendus. Tööstuslikud juhtseadmed on ülesande jaoks liiga kallid. Juhtseadmevalik oli üsna lihtne, sest Raspberry Pi-l on olemas kõik omadused selles hinnaklassis. Samuti Raspberry Pi kohta on hulgaliselt informatsiooni Internetis, foorumites, kui ka Raspberry Pi enda meeskond on alati valmis vastama küsimustele ja pakkuma omapoolseid lahendusi.

Miks valida just Raspberry Pi 4 mudel ja miks mitte Raspberry Pi 3 mudel? Raspberry Pi 4 mudelil on tehnilised näitajad päris paljuski paremad, kui mudel 3-el. Mudel 4 on kiirem protsessor ja graafikakaart, kui ka vahemälu on oluliselt rohkem, kusjuures 4-jas mudel on kaalult kergem, kui 3-mas. Micro SD pesa kiirus on varasema 3.125 MB/s asemel 6.25 MB/s. Wi-Fi edastuskiirus on ~60 MB/s mõlematel mudelitel 2.4 GHz sagedusel. Raspberry Pi 4 tarbib rohkem voolu, kui varasem mudel. Uus mudel tarbib tegevusetas olekus 3.4 W ja koormatuna 7.6 W, vanem mudel 2.9 W tegevusetas olekus ja 6.4 W koormatuna, kuid antud ülesande puhul on alati ligipääs elektrivõrgule. Raspberry Pi 4 hinnaks on 60€ ja Raspberry Pi 3 hinnaks on 50€.[3]

Parameeter	Raspberry PI 4B 2Gb	Raspberry PI 3 B+
Väljalaske aasta	2019	2018
CPU	1.5 GHz Cortex A-72	1.4 GHz Cortex A-53
RAM	2 GB DDR4	1 GB DDR2

GPU	500 MHz VideoCore VI	400 MHz Videocore IV
Wi-Fi edastuskiirus 2.4 GHz	~60 MB/s	~60 MB/s
Plaadi mõõtmed	88 x 58 x 19.5 mm	82 x 56 x 19.5 mm
Kaal	46 g	50 g
Micro SD pesa kiirus	3.125 MB/s	6.25 MB/s
GPIO piikribade arv	40	40
Hind	60€	50€

Tabel 1. Raspberry Pi 4B ja Raspberry Pi 3B+ oluliste parameetrite võrdlus

Miks mitte osta valmis lahendus? Raspberry Pi on odav ja teda on alati võimalik ümber programmeerida teiste ülesannete jaoks. Raspberry programmeerimine on üsna lihtne ja selle kohta on võimalik leida palju infot. Samuti on teda hea kasutada spetsiifiliste lahenduste jaoks, sest teda on võimalik paindlikult ümber seadistada. Mitmed erinevad Raspberry-d on võimalik liidestada omavahel.

2.2.3 Liikumise tuvastamine

Liikumise tuvastamise anduri paigaldan konveieri kohale. Andur peab suutma tuvastada konveieril liikuva trükkplaadi. Kui konveieril ei ole toodet, siis anduri ja konveieri vaheline kaugus on umbes 60 cm ultraheliandurist, kui aga konveieril on toode, siis anduri ja konveieri vaheline kaugus on vähem kui 55 cm.

Konveieril liikumise tuvastamiseks on mitmeid viise. Esiteks on variant kasutada käigulülitit, kui plaat liigub konveieril ja puudutab lüliti kontakti, siis tehakse pilt. Teise variandina on kasutada ultraheliandurit, millega mõõdetakse pidevalt kaugust. Kui kaugus vastab teatud kriteeriumile, siis juhtseade annab käsu kaamerale pildistamiseks. Kolmandana saaks kasutada laserkaugus- mõõdikut, mille tööpõhimõte on sarnane, kuid heli asemel kasutatakse hoopis valgust. On olemas ka optilisi liiniandureid, kuid nende hind on liiga kallis.

Valisin töö jaoks ultrahelianduri HC-SR04, hinnaks 1.5€. Tegemist on küllaltki tüüpilise ultrahelianduriga, mida Raspberry Pi huvilised kasutavad. Andur ei ole sobilik täpsete kauguste mõõtmiseks, kuid on piisav, et tuvastada liini peal liikumist. Anduriga saab 50-60 cm kauguselt mõõta umbes 1 cm täpsusega, kuigi ametlik mõõtetäpsus on 0.3 cm. Tööpingeks on 5 V, mõõtekaugus on 2-400 cm. HC-SR04 on 4 piikriba vcc, GND, echo ja trigger. Vcc piik on mõeldud 5 V pingele jaoks, kuid echo ja trigger tolereerivad kuni 3.3 V pinget. Raspberry Pi GPIO piikriba kontaktid on kõik 5 V, seetõttu tuleb joota ultraheliandurile takistid, et tekitada pingejagur. Tuleb valida 2 takistit, mis annaks $R1/(R1+R2) = \sim 0.66$ ja takistid oleks väärtusega 1-50 kΩ. Elektroonika

tootmisettevõttes on takistid kohapeal olemas, kasutasin 4.7 kΩ ja 2.7 kΩ takistit. On olemas HC-SR04P, millel on pingejagur 3.3 V jaoks juba olemas ja ei ole vaja joota takisteid andurile, kuid HC-SR04 andur oli kohe käepärast võtta, seetõttu otsustasin seda kasutada.[5]

2.2.4 Muud komponendid

Muudeks komponentideks süsteemis on toiteplokk, micro SD kaart ja RTC moodul. Soovituslik on kasutada Raspberry Pi 4 originaal toiteplokki, mis on 5.1 V ja 3 A, toiteploki hind on 8.5€.

SD kaardina kasutasin Adata 16 GB mälukaarti, kiiruseklass 10, hinnaks 8€. Mälu riknes umbes pärast kuuajast kasutamist. Ei olnud võimalik mälu vormindada ega faile taastada. Olin sunnitud ostma uue mälukaardi, kuid seekord valisin San Disk 16 GB, kiiruseklass 10, hinnaks 9€.

Piltide salvestamisel on oluline, et kellaage oleks võimalikult täpne. Raspberry Pi ühendatakse sisevõrku, millel puudub Internetiühendus. Ühendus kohaliku kellaserveriga ei toiminud stabiilselt. Seetõttu otsustasin, et tuleb kasutada RTC moodulit. RTC moodul on mõeldud hoidma õiget kellaagega ka siis, kui toimub volukatkestus. Mooduli reservtoiteks kasutatakse CR2032 patareid. DS1307 RTC mooduli hind on 2€.

2.2.5 Komponentide karp paigutamine

Kõigepealt tuleb vaadata komponentide mõõtmeid, seejärel leida sobiv karp, kuhu kõik komponendid paigutada. Ma valisin selleks harukarbi. Karbis on varuga ruumi nii komponentide kui ka nende juhtmete jaoks.

Toode	Hind	Mõõtmed
Raspberry Pi 4B	60€	88 x 58 x 19.5 mm
IMX298 kaamera moodul	55€	40 x 40 x 9 mm
HC-SR04 ultraheli-andur	1€	45 x 20 x 15 mm
Harukarp	2€	104 x 104 x 39 mm (sisemõõt)

Tabel 2. Komponentide hinnad ja mõõtmed

Joonisel 1. on kujutatud harukarpi, milles ultraheliandur mõõdab kaugust ja kaamera pildistab. Samuti on Raspberry Pi toitejuhtme jaoks tehtud ava harukarbi küljele.



Joonis 1. Raspberry Pi, ultrahelianduri ja kaamera asetsemine harukarbis

2.2.6 Riistvara maksumus

Hinnad on kõik ümardatud lähima üheliseni. Soovituslik on kasutada LED valgustust ja dimmerit, et pildid oleksid ühtlase valgustusega. Tabelis 3. on näha, et riistvara hinnaks tuleb kokku koos dimmeriga LED valgustusega 162€, valgustuseta on hind 138€. Projekt tuli luua minimaalsete kuludega, kuid siiski kasutasin oma töös valgustust. Tabelis pole arvestatud kõvaketta maksumust, mis on vajalik piltide hoiustamiseks. Võimalik, et tabelis 3. hinnad võivad erineda hetke-hindadest.

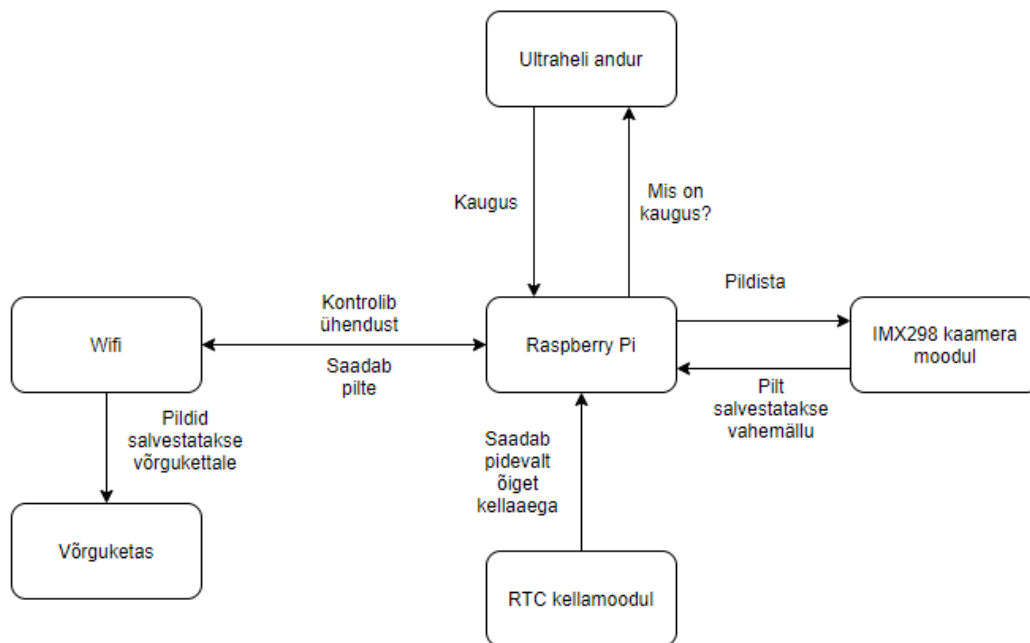
Toode	Hind
Raspberry Pi 4B	60€
IMX298 kaamera moodul	55€
HC-SR04 ultraheliandur	1€
Harukarp	2€
Raspberry Pi Toiteplokk	9€
SD kaart	9€
RTC moodul DS1307	2€
LED riba 1 m	3€

LED riba toiteplokk	16€
LED dimmer	5€
Kokku:	162€

Tabel 3. Komponentide hinnad

2.3 Tarkvara

Antud peatükis käsitletakse Raspberry Pi-ga seotud tarkvaralisi valikuid ja probleeme. Joonisel 2. on näha infovoogude liikumist läbi juhtseadme.



Joonis 2. Liinikaameras toimuvad interaktsioonid

2.3.1 Operatsioonisüsteemi valik

Raspberry Pi-le tuleb valida operatsiooni süsteem. Kasutajal on vabad käed, millist operatsioonisüsteemi ta kasutama hakkab. Raspberry ametlikult veebilehelt saab tõmmata tasuta operatsioonisüsteeme näiteks Raspbian või Noobs. Samuti toimivad mikroarvutil tuntud operatsioonisüsteemid nagu Ubuntu, Windows 10 Iot või Android.

Mina valisin operatsioonisüsteemiks Raspbiani. Raspbian on 32-bitine Linuxil baseeruv operatsioonisüsteem, olgugi et Raspberry Pi 4B protsessor on 64-bitine. Operatsioonisüsteem on spetsiaalselt loodud Raspberry Pi mikroarvutitele ja sellega

tuleb kaasa programmeerimiseks vajalik tarkvara näiteks Thonny, Geany, Scratch jne. Raspberry Pi mikroarvuti plussiks on kindlasti, et tarkvara on valdavalt tasuta võrdlusena Siemens kontrolleri puhul on vajalik osta ka tarkvara.

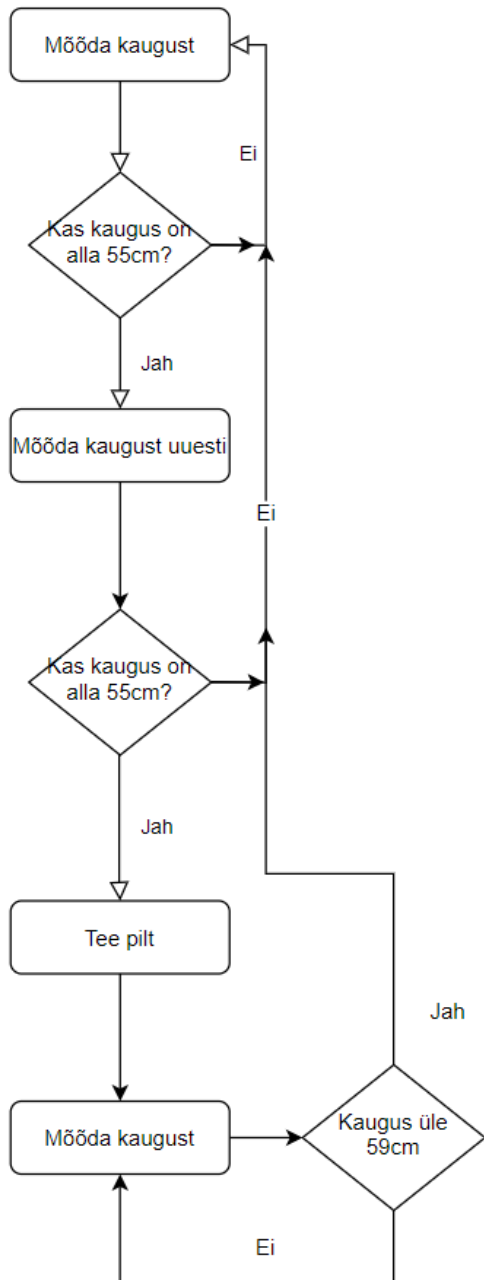
2.3.2 Pildi parameetrid

Arducam on loonud Githubi oma kaameratele sellise teegi nagu „MIPI_Camera“. Teek sisaldab erinevate Arducam kaamerate draivereid, näidiskoodi nii C keeles kui ka Pythonis. Teek on pidevalt arenduses ja seda täiendatakse igapäevaselt.

Manuaalselt saab muuta pildistamise parameetreid nagu fookuskaugus, säriaeg, RGB balanss, automaatne valge balanss, formaat ja resolutsioon. Arducam IMX298 puhul saab kasutada ainult manuaalset fookust. On võimalik kasutada nii automaatset säriaega, kui ka fikseeritud. Valisin fikseeritud säriajaks 450 ms, sest nii on pilt teravam, kuid pildi värvid on tuhmimad. Värvid ei ole piltidel olulised, kuid detailid on. Samuti automaatne säriaeg valgustab vahel pilte üle, kui tegemist on valge trükkplaadiga. RGB balanss tuli alguses ära seadistada manuaalselt, esimesed pildid olid üleni rohelised. Automaatset valge balansi on mõistlik kasutada, sest see parandab värvide kvaliteeti ja aitab peegeldusi vältida. Pilte saab teha erinevates formaatides jpeg, bmp, png ja raw. Kasutasin tüüpilist formaati jpeg. Resolutsioone on kaks, mille vahel valida: kas 2336x1748 (4 MPix) või 4672x3492 (16 MPix). Valisin kõrgeima resolutsiooni, sest nende kahe vahel on tuntuva kvaliteedi erinevus. 4 MPix režiimis on piltide suurus 1-2 MB ja 16 MPix režiimis on piltide suurus 9-11 MB. Kuu ajaga tekib ühel liinil ~31000 pilti, umbkaudu 300 GB jagu, kui kasutada 16 MPix resolutsiooni. [7]

2.3.3 Programmi ülesehitus

Programm on loodud Arducam „MIPI_Camera“ teegi toel. Teek sisaldab kaamera draivereid ja konfiguratsioonifaile. Mikroarvuti peamine juhtimissüsteem on loodud Pythonis, pildistamise ja pildi parameetrid on defineeritud C keeles eraldi failis. Juhtimissüsteem kontrollib ühendust võrgukettaga, mõõdab kaugust ultrahelianduriga ja annab käsu pildistamiseks soodsate olude tekkel. Pildistamise ja pildi parameetrite failis on defineeritud kõik pildiga seotud parameetrid, kui ka piltide salvestamise asukoht ja lahendatud piltidele nime andmine.[8]



Joonis 3. Liinikaamera pildistamise programmi tööpõhimõte

Vasakul oleval joonisel 3. on välja toodud liinikaamera pildistamise programmi üldine juhtimisskeem. Ultraheliandurid mõningatel juhtudel tuvastavad kaugust valesi. Kauguse mõõtmine toimub programmisiseselt 2 korda, et vältida valesid mõõtetulemusi. Kahel järjestikkusel mõõtmisel on kahe väär tulemuse tuvastamine väga väike. Samuti tegemist ei ole kriitilise tähtsusega süsteemiga: kui juhtubki olema, et pildistatakse tühja tooteliini, siis see ei ole suureks probleemiks. Pildistamine toimub, kui on mõõdetud kahel järjestikul mõõtmisel kauguseks alla 55 cm ultrahelianduri ja tooteliini vahel. Programm ei lähe algfaasi tagasi enne kui kauguseks mõõdetakse üle 59 cm ehk toode on ultrahelianduri eest ära liikunud.

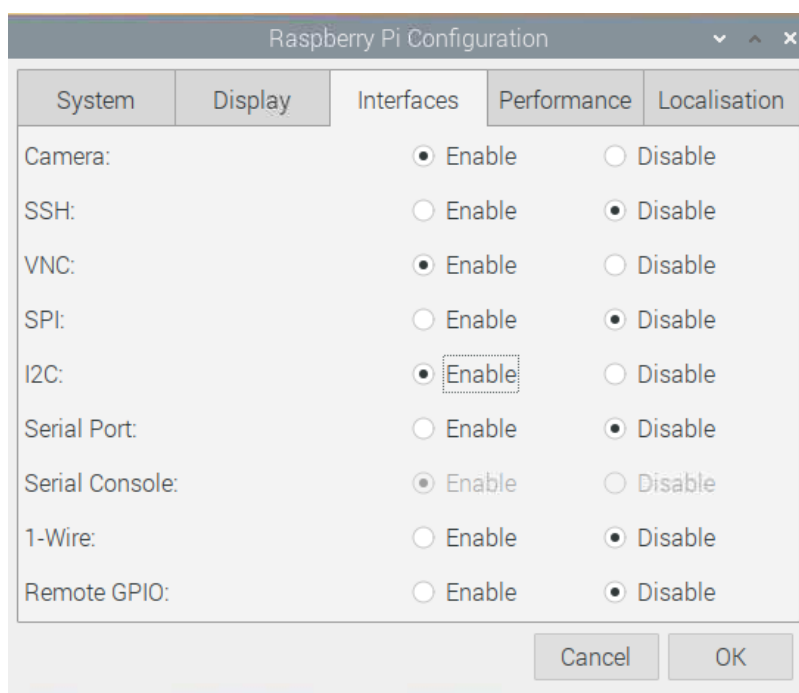
Programmi koodi käesolevale tööle ei lisa, kuna see sisaldab Enics sisevõrgu ja tehnoloogia kohta liiga detailset informatsiooni. Kood on kirjutatud Enicis töötamise ajal ja selle omandiõigus kuulub Enics Eesti AS-le.

2.3.4 Raspberry Pi konfigureerimine

Selles peatükis kirjeldan, millised konfiguratsiooni muudatused tuleb läbi viia, et Raspberry PI toimiks kui liinikaamera. Selleks, et alustada süsteemi konfigureerimisega, peab olema eelpool mainitud riistvara ühendatud Raspberry Pi külge ja Raspberry Pi ühendatud voluvõrku. Enne konfiguratsioonidega tegelemist peab süsteem olema uuendatud, et tagada töökindlus.

Raspberry PI üldine konfigureerimine:

Joonisel 4. on näidatud, milline on Raspberry PI konfiguratsioon liinikaamera lahenduse korral. Pildiloleva akna saab avada, kui liikuda start --> preference --> Raspberry Pi configuration --> interfaces. *Camera* ja I2C peavad olema lubatud, et kaamerat saaks kasutada. Kaamera suhtleb juhtseadmega läbi I2C liidese, seetõttu peab I2C olema lubatud. VNC peab olema lubatud juhul, kui on vaja juhtseadmele kaugelt juurdepääsu. Antud juhul on kaugligipääs vajalik, sest juhtseade on paigaldatud liini kohale ja on väga keeruline juhtseadmele külge ühendada ekraani, hiirt ja klaviatuuri keset tootmisprotsessi. VNC abil on võimalik teha kõik muudatused distantsilt. VNC kuvab Raspberry PI ekraani täpselt samamoodi, kui ta kuvaks seda kuvaril, mis oleks juhtseadmega ühendatud. SSH on vajalik siis, kui soovitakse ligipääseda ainult juhtseadme käsuviibale ja pole vajalik ligipääseda operatsioonisüsteemi rakendustele. Muud Raspberry Pi sätted võib jätta *disable* olekusse. Igasuguste ühenduste lubamine tekitab turvariske.



Joonis 4. Raspberry Pi üldised konfiguratsioonid

Järgmise sammuna installeerisin ekraanisäästja käsuga „sudo apt-get install xscreensaver“. Vaikimisi on ekraanisäästja kogu aeg sees, seetõttu tuleb installeerida xscreensaver, kus saab ekraanisäästja välja lülitada. Selleks tuleb vajutada start --> preference --> screensaver --> disable screensaver. Lülitasin ekraanisäästja seetõttu välja, et säästa aega VNC ühenduse loomisel, vastasel juhul on ekraanisäästja ees ja selle ekraanilt ära saamine võtab aega. Selleks, et VNC ühendus luua, on vaja teada juhtseadme kasutajanime ja parooli.

Kui juhtseadet juhitakse üle kaugühenduse ja soovitakse teha taaskäivitus, siis juhtseade ei pruugi käivituda uuesti. *Boot* konfiguratsiooni failis võib olla nõutud HDMI ühendus, et juhtseade käivituks. Seetõttu tuleb käsuviibalt minna „boot“ kausta ja sisestada käsk „sudo nano config.txt“. Käsu abil saab muuta config.txt faili. Failis tuleb leida rida „# hdmi_force_hotplug=1“ ja eemaldada sealt kommenteerimise märk „#“. Järgmise sammuna salvestada ja sulgeda fail.

Võrguketta ühendamise:

Selleks, et luua ühendus võrgukettaga, on vaja muuta faili nimega „fstab“. Faili viimasele reale tuleb lisada alltoodud rida, kus peale kahte kaldkriipsu tuleb märkida võrguketta IP aadress. (Konfidentsiaalsuse huvides eemaldasin IP aadressi lõputööst). Järgnev osa „Data/Datastore/Production/WS“ tähistab kausta, millele soovitakse ligipääs saada. Järgnev osa koodust „/home/pi/WS“ näitab asukohta, kus soovitakse kuvada lisatud võrguketast juhtseadmes.

```
//IP_aadress/Data/Datastore/Production/WS /home/pi/WS cifs credentials=/home/pi/.NASCREDS,vers=2.0,domain=DomeeniNimi
```

Selleks, et võrgukettale ligi saada on vaja sisse logida Enics AS kasutajaga, selleks tuleb lisada „credentials=/home/pi/.NASCREDS“. Tegemist on tekstifaili asukohaga, kus asuvad kasutajanimi ja parool. Tekstifaili saab luua käsuga „sudo nano .NASCREDS“. Faili tuleb märkida kasutajanimi ja parool vastavalt alltoodud kuvatõmmisele. „vers=2.0“ tähistab SMB versiooni. Võrguketta server kasutab versioon 2, sellest lähtuvalt tuleb täpsustada, millist versiooni peab juhtseade suhtlemisel kasutama. Identimiseks kasutatakse veel domeeni.[6]

```
username=KasutajaNimi  
password=MinuParool
```

Failide automaatne kustutamine:

Failide automaatne kustutamine on rajatud Linuxi ülesannete planeerimise rakenduses – *crontab*. Selleks tuleb sisestada käsk „crontab -e # pi“ ja valida number 1.

```
pi@raspberrypi:~ $ crontab -e # pi
no crontab for pi - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano      <---- easiest
 2. /usr/bin/vim.tiny
 3. /bin/ed

Choose 1-3 [1]: █
```

Allpoololeval kuvatõmmisel on näha koodirida, mis kustutab faile, mis on olnud kaustas „Raw” rohkem kui 28 päeva. Koodirida on lisatud crontab faili viimasele reale. „0 5 * * *” tähistavad vastavalt minutit, tundi, kuupäeva, kuud, nädalapäeva. Ülesanne käivitatakse igapäev kell 5.00. Muud parameetrid tähistasin tärniga, sest neid ei ole vaja täpsustada. Kui muudatused ellu viidud, tuleb fail salvestada ja sulgeda.

```
0 5 * * * sudo find /home/pi/WS/RaspberryPi/Raw/ -type f -mtime +28 -print0 |sudo xargs -0 -r rm -rf
```

Süsteemi teenuste loomine:

Järgmise sammuna lõin süsteemiteenuse, mis käivitab liinikaamera juhtimisprogrammi. Selleks tuleb luua uus fail käsuga „sudo nano /lib/systemd/system/teenuse.nimi”. Käsk loob uue tühja faili, kuhu on vaja lisada allpooltoodud kuvatõmmisel olevad read. Kui read on lisatud, siis tuleb fail sulgeda ja salvestada.

```
GNU nano 3.2 /lib/systemd/system/sample.service

[Unit]
Description=Juhtimisprogramm
After=multi-user.target

[Service]
Type=idle
ExecStart=/usr/bin/python3 /home/pi/Desktop/Driver_programm.py

[Install]
WantedBy=multi-user.target
```

Järgmine samm on anda programmile õigused. Selleks tuleb käsureal sisestada käsk „sudo chmod 644 /lib/systemd/system/sample.service”. Seejärel on vaja uuendada süsteemis tehtud konfiguratsioone käsuga „sudo systemctl daemon-reload”, kui ka lubada programmi kasutus juhtseadmes käsuga „sudo systemctl enable Juhtimisprogramm”. Seejärel on vaja Raspberry Pi-le teha taaskäivitus, mille järel käivitub „Driver_programm.py” automaatselt koos teiste süsteemi teenustega.

Toiteprobleemi tuvastamine:

Esialguses Raspberry Pi mikroarvuti katsetamises kasutasin mobiiltelefoni laadijat, millele oli märgitud voolutugevus 2 A ja toitepingeks 5 V, mis teeb kokku 10 W. Raspberry Pi-le on tootja poolt ette nähtud 3 A ja 5.1 V toitepinge ehk 15.3 W toiteallikas. Alguses toimis mikroarvuti ka 10 W toitega, kuid probleemid hakkasid ilmnema kui mikroarvuti külge sai ühendatud rohkem riistvara. Mikroarvuti hakkas hanguma aeg ajalt. Selleks et kontrollida, kas viga on toites tuli sisestada käsureale käsk „vcgencmd get_throttled“ ja muu vastuse korral kui „throttled=0x0“ on tegemist probleemiga. Probleemideks võib olla madal toitepinge või ülekuumenev protsessor, täpsemalt probleemist saab teada veakoodi alusel.

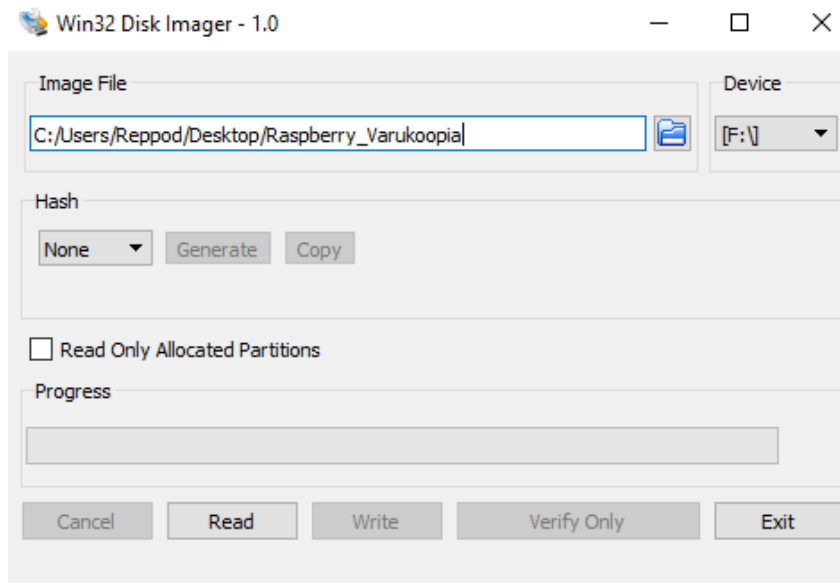
```
pi@raspberrypi:~ $ vcgencmd get_throttled  
throttled=0x50000
```

Ülaltoodud veakood viitab ebapiisavale toiteploki. Seetõttu võtsin kasutusele tootja poolt ettenähtud 15.3 W toiteploki. Pärast toiteploki vahetust sisestasin käsureal käsu „vcgencmd get_throttled“ ja tulemus näitab, et toide on piisav. Samuti lõppesid mikroarvuti hangumised suvalistel ajahetkedel.

```
pi@raspberrypi:~ $ vcgencmd get_throttled  
throttled=0x0
```

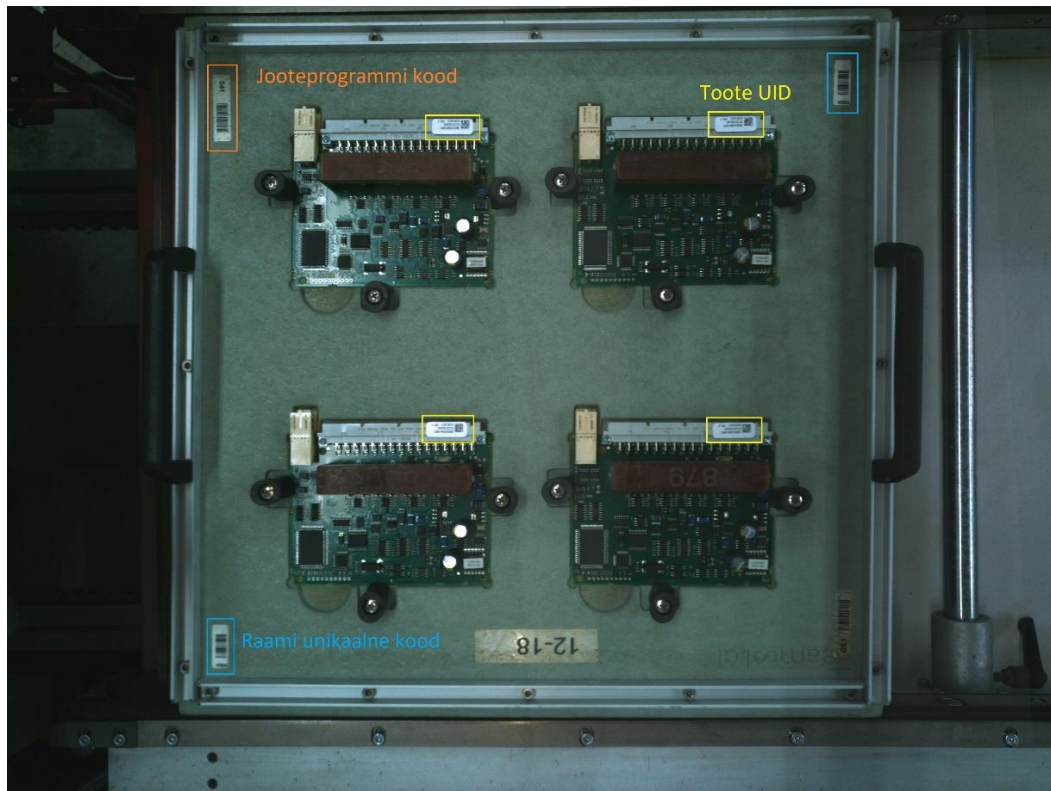
Kettatõmmise või varukoopia loomine:

Programmist tasub teha varukoopiad, kui on tehtud suuri muudatusi. Kõige lihtsam viis on kopeerida muudetud failide koopiad teisele kõvakettale. Varasem näide SD kaardi riknemisest võib tulla ootamatult ja faile ei pruugi olla võimalik taastada. Parem viis koopia loomisest on teha kettatõmmis SD kaardist. Sellega on garanteeritud, et iga muudatus, mis on tehtud operatsioonisüsteemiga või ükskõik millise failiga, on jäädvustatud ühte kettatõmmise faili. Samuti on see hea viis, kuidas kiirelt installeerida täpselt samasugune lahendus, mitmele SD kaardile. Kettatõmmise loomiseks kasutasin Win32 Disk Imager tarkvara.



2.3.5 Võimalikud edasiarendused

Edasi võiks tuvastada pildi abil, millise tootega on tegemist. Selle tegemiseks on vaja koguda MES tool andmeid erinevatelt ladumislauadelt ja tarkvara, mis suudab vöotkoodi lugeda. Lahenduse abil on võimalik liinikaamera andmebaasist üles leida sama toote pilte, mis on jäädvustatud erinevatel kellaegadel.



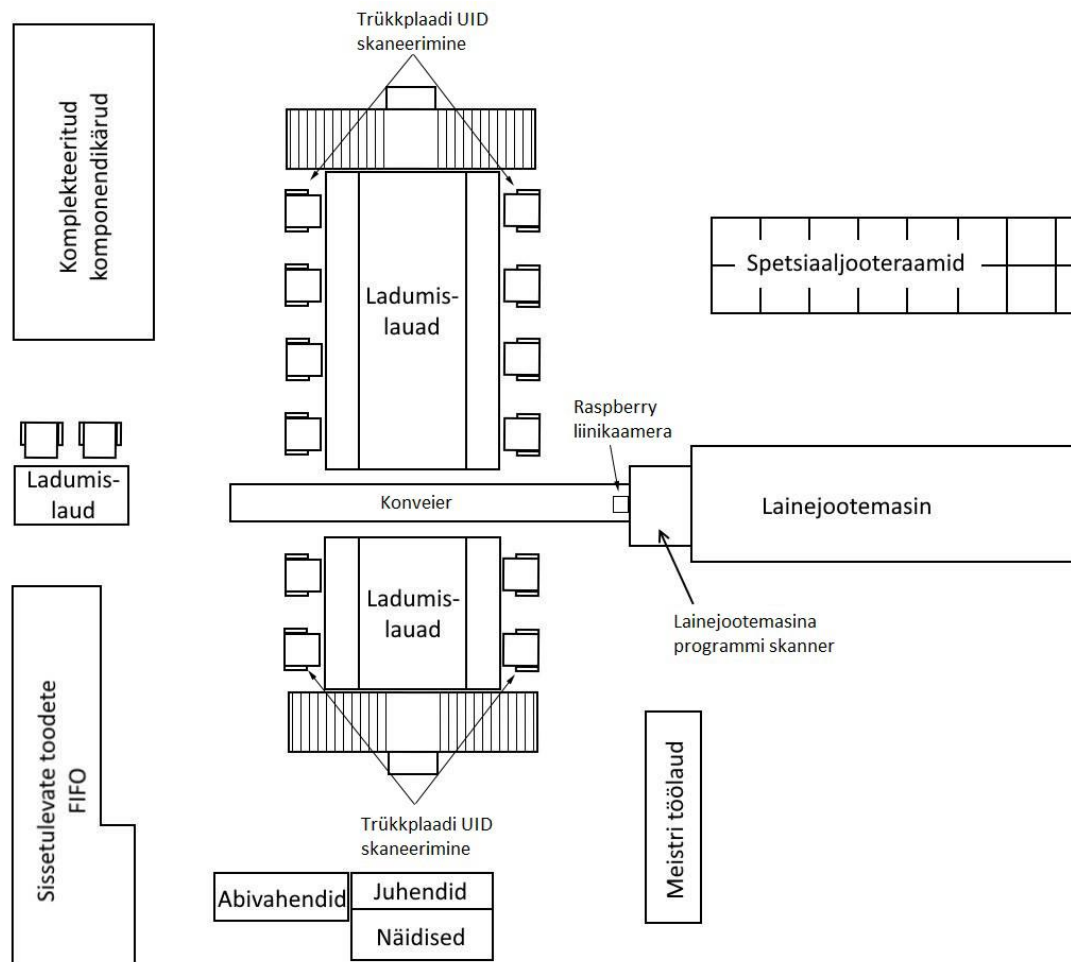
Joonis 4. Jooteraamil ja tootel asuvate vöotkoodide asukohad

Liinikaamera tehtud piltidel on erinevad vöotkoodid: jooteprogrammi kood, jooteraami unikaalne kood, toote UID ja paneeli UID. Paneeli UID on ainult toodetel, mis on paneelis mitmekaupana. Kui käsiladuja töö alustab, siis ta skaneerib jooteraami unikaalse koodi, toote UID ja vajadusel paneeli UID MES tool süsteemi, kus märgitakse töö alustatuks. Kui käsiladuja on töö valmis saanud, siis ta asetab jooteraami koos tootega konveierile. Liinikaamera teeb pildi, kui toode konveieril liigub. Konveieril on lainejoote ahju skanner, mis leiab jooteraamilt koodi ja selle alusel seadistab jootmisega seotud parameetrid vastavalt koodile.

Kui toode on läbinud lainejoote ahju, siis võetakse see jooteraamist välja ja saadetakse järgmise etappi. Konveieril on raamide tagastamise süsteem, mis skaneerib jooteraami unikaalse koodi ja saadab tagasi käsiladuja juurde, kes selle raamiga viimati tööd tegi. Pilt vöotkoodide asukohtadest on ülaltoodud joonisel 5.

Selleks, et selgitada välja, milline toode on pildil, tuleb:

- lugeda pildilt raami unikaalne vöotkood;
- leida raamide tagastamise andmebaasist, millise ladumise laua juurde raam tagastati;
- leida MES tool andmetest, millist tellimust käsiladuja parasjagu täitis



Joonis 6. Lainejoote protsessi üldine vaade

Kui esimene käsiladuja alustab tootega tööd, siis ta skaneerib toote MES tool rakenduses, kust on näha, et tööga on alustatud. Ühe tootega töötab 1-4 inimest, mille tõttu ei saa määrata, kudas toode täpselt konveierile jõuab, eriti kui peaks viivitusi tekkima. Seetõttu ei saa ainult MES tool andmete ja toote pildistamise aega omavahel sobitada, et leida millise tootega on tegemist.

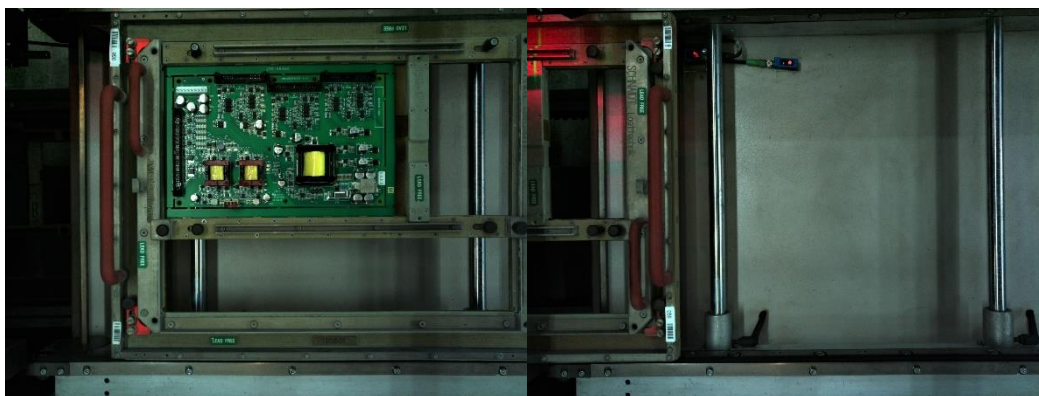
Teine probleem on selles, et üks lainejoote programm võib sobida mitmele tootele ja koodi järgi pole võimalik määrata, millise tootega tegemist on. Enne lainejoote ahju on skanner, mis skaneerib raamilt lainejoote programmi vöotkoodi. Kui plaat on läbinud lainejooteprotsessi, siis teises konveieri otsas võtavad plaate vastu masina operaatorid. Nad skaneerivad kõik tooted MES tooli, et näidata lainejooteprotsessi lõppemist. Operaatorid ei skaneeri alati plaate õiges järjekorras, seetõttu võib juhtuda, et pilt seotakse vale tootenumbriga.

Minu poolt pakutava lahenduse elluviimine on keerulisem, kuid see tagaks 100 % täpsuse. Selleks on vaja tarkvara, mis suudab piltidelt lugeda vöötkoode – täpsemalt on vaja lugeda raami unikaalset koodi. See on implementeeritav, sest nutitelefon on ka suuteline triipkoodi pildilt lugema. Raamide tagastamise andmebaasist on vaja leida, millise käsiladumise laua juurde raam tagastati. See on juba seetõttu oluline, et osadel toodetel kasutatakse universaalraame ja need sobivad mitmele erinevale tootele, kuid MES tool andmetest on võimalik leida, millist tellimust käsiladuja parasjagu täitis. Tellimuses on märgitud tootenumber. Tootenumbri saab lisada liinikaamera andmebaasi, mis koosneb piltidest. Vajadusel on võimalik otsida pilte tootenumbri põhjal.

3 SÜSTEEMI TESTIMINE

Testimine on süsteemi osa, millega tõestan, et süsteem toimib ootuspäraselt ja vastab kasutajanõuetele, mis on välja toodud punktis 2.1 Kasutaja- ja süsteeminõuded. Testid tuleb läbi viia pärast igat süsteemiuuendust, et veenduda, et kõik süsteemi osad on endiselt funktsionaalsed. Kaameramooduli IMX298 tarkvara uuendatakse iganädalaselt, kuid ettevõttesiseselt installeeritakse uus tarkvara süsteemile korra kvartalis. On näha, et pildi kvaliteet on oluliselt paranenud ja pildistamisel tekkinud ebaloomulikke moonutusi on harvemini. Allpool on toodud testid, mis tuleks läbi viia, et kontrollida süsteemi vastavust kasutajanõuetele.





1. Test, mis kinnitab, et kasutajal on ligipääs trükkplaatidest tehtud piltidele, mis asuvad andmebaasis. Andmebaasile saavad ligi kõik, kellele on antud vastavad ligipääsuõigused. Selle eest vastutav on ettevõtte IT osakond. Kui ei ole võimalik andmebaasi avada, siis kas kasutajal puudub Internetiühendus või tal puuduvad õigused andmebaasile ligi pääseda. Muid probleeme pole täheldanud ja see jääb n-ö teenusepakkuja kui "IT osakonna" vastutuseks.
2. Test, millega kontrollitakse, et pildid on asjakohased ja õigel ajal pildistatud ehk toode asub tervikuna kaadris. Otseselt ei ole probleem, kui ühest tootest tehakse 2 pilti. Selleks, et garanteerida, et pildid oleks alati üleni kaadris on mitmeid erinevaid võimalusi. Joonisel 7. on näha, et ühest tootest tehakse kaks pilti. Probleem on lihtne, tegemist on raamiga, mis ei ole üks ühtlane tasand. Tasandis on n-ö augud, seetõttu ultraheliandur tuvastab seda jooteraami kaks korda. Mikroarvuti juhtprogrammi „arvates“ on tegemist kahe erineva raamiga, kuigi tegemist on ühe raamiga. Probleemi lahendamaks ultrahelianduri paigaldamine raami äärde, kus puuduvad tasandis nõ augud. Hetkel selline defektne nähtus esineb ~3 % piltidest.



Joonis 7. Korrektnen foto ja ebakorrektnen duplikaat foto

Testimiseks on vaja programmi, mis tuvastab, et toode on 100 % kaadris. Jooteraame on väga erinevaid ja nende raamide "õpetamine" pildi tuvastamise rakendusele on liiga tömahukas, kuid igal jooteraamil on alguses ja lõpus vöötkoodid. Üks vöötkood on raami unikaalne UID ja teine kood on jooteprogrammi käivitamise kood. Testi rakendus saab kontrollida, kas ta leiab näiteks sajal pildil üles need koodid ja kui ta leiab, siis järelkult pildid on kaadris. Kui oleks olemas piisavalt keerukas tarkvara, siis oleks võimalik pildistada nende koodide järgi. Kui koodid jõuavad kindlatesse koordinaatidesse, siis jäädvustatakse foto. See eeldaks suuremat protsessori võimekust ja keerukamat tarkvara.

3. Test, millega kontrollitakse, et pildidel olev kellaeg ja kuupäev on korrektsed. Pildidel olevat kellaega ja kuupäeva on võimalik manuaalselt kontrollida. Kui pildid salvestatakse serverisse, siis lisatakse foto andmetesse salvestamise kellaeg ja kuupäev. Mikroarvuti määrab pildi nimeks kuupäeva ja kellaaja, mil ta foto jäädvustas. Serverisse salvestamise kellaega ja pildi nimes olevat kellaega on võimalik omavahel võrrelda ja kontrollida, kas kellaeg on õige. Alltoodud kuvatõmmisel on näha, et kellaajad kattuvad ja minuti täpsus on igati aktsepteeritav antud ülesande puhul.

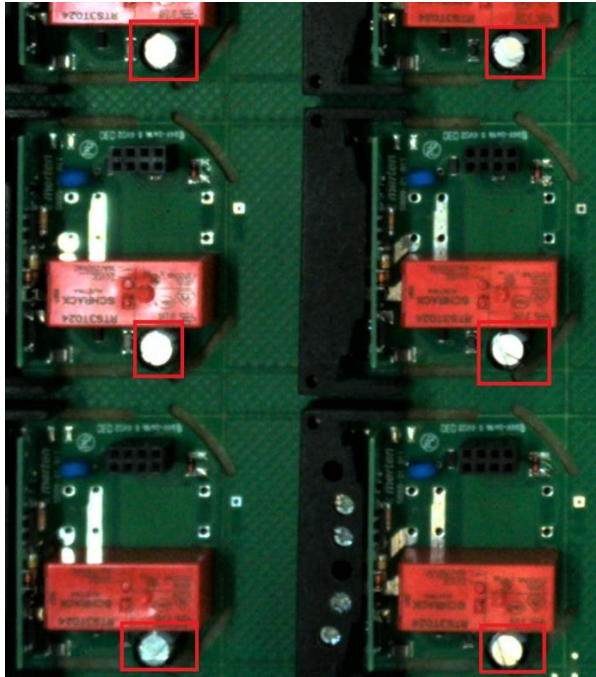
Name	Date modified	Type	Size
 capture 20-04-2020 14.28.29.jpg	20.04.2020 14:28	JPG File	10 848 KB
 capture 20-04-2020 14.27.05.jpg	20.04.2020 14:27	JPG File	10 940 KB
 capture 20-04-2020 14.26.43.jpg	20.04.2020 14:26	JPG File	11 446 KB
 capture 20-04-2020 14.26.26.jpg	20.04.2020 14:26	JPG File	10 762 KB

Selle süsteemi testimiseks on vaja programmi, mis võrdleb serverisse salvestamise kellaega ja pildi nimes olevat kellaega. Kui kellaajad ei jää lubatud hälbe sisse, siis saadab programm kasutajale kirja, et süsteemis on viga. Järgmise sammuna tuleb kasutajal välja selgitada, kas mikroarvuti või serveri kellaeg on vale.

4. Testida, et pildid asuvad õigetes kaustades, saab teha samamoodi nagu on välja toodud eelmises punktis.
5. Selleks, et kontrollida, kas pildilt on näha polaarsusi on mitu varianti: Esimene variant on see, et kasutaja vaatab pildi üle ja määrab, kas pilt on ok/not ok. Vastavalt saadud statistikale määrab, kas programm on toimiv või ei ole.

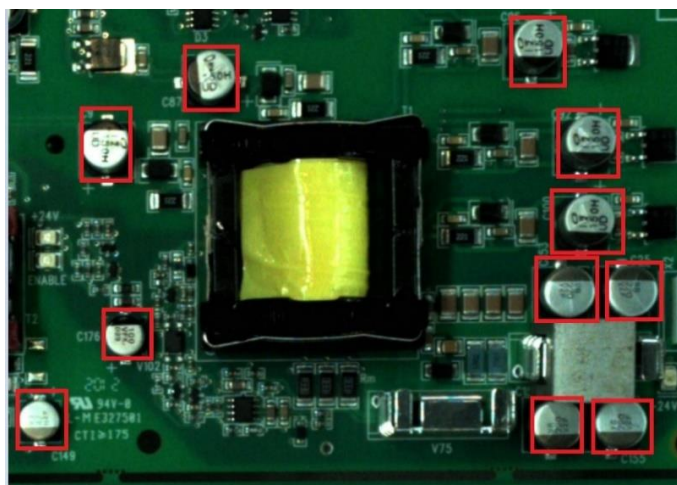
Teise variandina on kasutada masinõpet. Masinõppe programmile „õpetatakse“, milline komponent on polaarne ja millised võivad olla erinevad polaarsuse

tähistused komponentidel. Oluline on, et pildilt oleks leitavad THT komponendid, kuid kui on nähtavad polaarsete komponentide polaarsuse tähistused, siis on ka nähtavad THT komponendid. Joonisel 8. on näha, et teatud tüüpi kondensaatorid peegeldavad valgust ja tekitavad hägusust. Tegemist on väikese tootega ja pilte tehakse liikumise pealt.



Joonis 8. Kondensaatorite polaarsuse tuvastamine

Üldiselt polaarsuse märgistus tähistab negatiivset poolt. Väiksemad ehk SMA komponendid kontrollitakse juba eelnevas etapis. Teatud SMA komponente pole silmaga hästi nähagi. Joonisel 9. on kondensaatorite polaarsused kergelt määratavad.



Joonis 9. Kondensaatorite polaarsuse tuvastamine

KOKKUVÕTE

Lõputöö ülesanne on püstitatud Enics Eesti AS poolt ja on abiks ettevõtte tegevuse arendamiseks. Ettevõtte toodab tööstuselektronikat. Lõputöö eesmärk on luua liinikaamera, mis pildistab trükkplaate enne kui need lähevad lainejoote ahju. Liinikaamera jäädvustatud piltidest luuakse andmebaas. Jäädvustatud piltide abil on võimalik kontrollida, millises etapist alates esines trükkplaadil defekt, et korraldada korrigeerivaid tegevusi töö korralduses.

Töö koosneb kavandamisest nii riistvaraliselt kui ka tarkvaraliselt. Oluline osa kavandamisest on punkt 2.1 „Kasutaja- ja süsteeminõuded“, kus määratletakse ära, millised peavad olema liinikaamera süstemaatilised omadused ja millised on kasutaja ootused liinikaamera suhtes. Punktis 2.2 „Riistvara“ on seletatud lahti, mille põhjal on valitud riistvaralised lahendused. Liinikaamera riistvaralisteks põhilisteks osadeks on kaamera, juhtseade ja liikumise tuvastamise andur, kui ka RTC moodul, SD kaart, toiteplokk ja valgustus.

Punktis 2.3 „Tarkvara“ on seletatud lahti, mille põhjal on loodud süsteemi tarkvara. Liinikaamera loomisel on oluline rõhku pöörata operatsiooni süsteemi valikule, pildi parameetritele, liinikaamera juhtprogrammi ülesehitusele kui ka juhtseadme konfiguratsioonidele.

Lõputöö tulemuseks on Raspberry Pi baasil toimiv liinikaamera. Liinikaamera on paigaldatud tooteliini kohale. Liinikaamera vastab süsteeminõuetele ja kasutaja ootustele. Lisaks defektide tuvastamisele kasutatakse liinikaamera andmebaasi statistika tegemisel. Statistika eesmärk on jälgida lainejoote masina koormatust iga veerandtunni jooksul ja leida andmetest käitumismustreid.

Edasiarendusena võiks luua süsteemi, mis on suuteline tuvastama pildil olevaid tooteid. Lahendus võimaldaks liinikaamera fotode andmebaasist üles leida sama toote pilte, mis on jäädvustatud erinevatel kellaaegadel. Peatükis 3 „Süsteemi testimise“ on toodud välja veel võimalikke edasiarendusi, mida on võimalik tarkvaraliselt lahendada. Eriti perspektiivikas on kasutada liinikaamera fotosid automaatse optilise kontrolli tarbeks. Fotode põhjal kontrollida komponentide olemasolu, polaarsust, defekte, jootuvust ja palju muud toote spetsiifikale vastavat.

SUMMARY

Task of this thesis is established by Enics Eesti AS and its goal is to improve the company's operations. The company produces industrial electronics. Objective of the thesis is to create a line camera, which takes pictures of circuit boards before they enter the wave soldering oven. Pictures taken with a line camera will create a database. The pictures can be used to check from which stage the circuit board was defective.

The thesis consists of a hardware and software design of the line camera. An important part of the design is presented in section 2.1 "User and system requirements", where the systematic characteristics of the line camera and the user's expectations of the line camera are defined. Section 2.2 "System hardware" describes the basis on which the system hardware was chosen. The line camera consists of a camera module, control unit, motion detection sensor, RTC module, SD card, a power supply and illumination.

Section 2.3 "System software" describes how system software is created. It is important to focus on choosing a suitable operation system and image parameters. Also it is important to create suitable software and configuration for the line camera control unit.

Result of the thesis is a functional Raspberry Pi based line camera. The line camera is placed above the product line. The line camera meets the requirements of system characteristics and user expectations. In addition to detection of defects, the line camera is used for statistics. The purpose of statistics is to track loading of a wave soldering oven.

As a further development a system capable of identifying products in a image could be created. The solution would make it possible to find images of the same product taken at different times from a database of images. The chapter 3 "System Testing" brings out more possible developments that can be solved by software. There exists a prospect for creating an automatic optical inspection software for a line camera. Automatic optical inspection could check the presence of components, polarity, defects, solderability and other features depending on the product specification.

KASUTATUD KIRJANDUSE LOETELU

- [1] Jan Toodre „Raspberry PI-l põhinev liikumistuvastusega videojälgimissüsteem,“ 2015 [Võrgumaterjal]
Kättesaadav: <https://digikogu.taltech.ee/et/Item/142e4998-9f43-4893-8831-372a31c4c763>
- [2] „How to pick the right Raspberry Pi,“ 2018. [Võrgumaterjal].
Kättesaadav: <https://www.digikey.com/en/maker/blogs/2018/how-to-pick-the-right-raspberry-pi>
- [3] Avram Piltch ja Gareth Halfacree, „Raspberry Pi 4 Review: The Gold Standard for Single-Board Computing,“ 2019. [Võrgumaterjal].
Kättesaadav: <https://www.tomshardware.com/reviews/raspberry-pi-4-b,6193.html>
- [4] Tooteleht [Võrgumaterjal]
Kättesaadav:
https://www.sony-semicon.co.jp/products/common/pdf/ProductBrief_IMX298_20160210.pdf
- [5] Tooteleht [Võrgumaterjal]
Kättesaadav: https://www.elecrow.com/download/HC_SR04%20Datasheet.pdf
- [6] Tooteleht [Võrgumaterjal]
Kättesaadav: <https://cifs.com/>
- [7] Tooteleht [Võrgumaterjal]
Kättesaadav: <https://www.arducam.com/docs/camera-breakout-board/16mp-imx298/>
- [8] MIPI Camera teek [Võrgumaterjal]
Kättesaadav: https://github.com/ArduCAM/MIPI_Camera/tree/master/RPI