

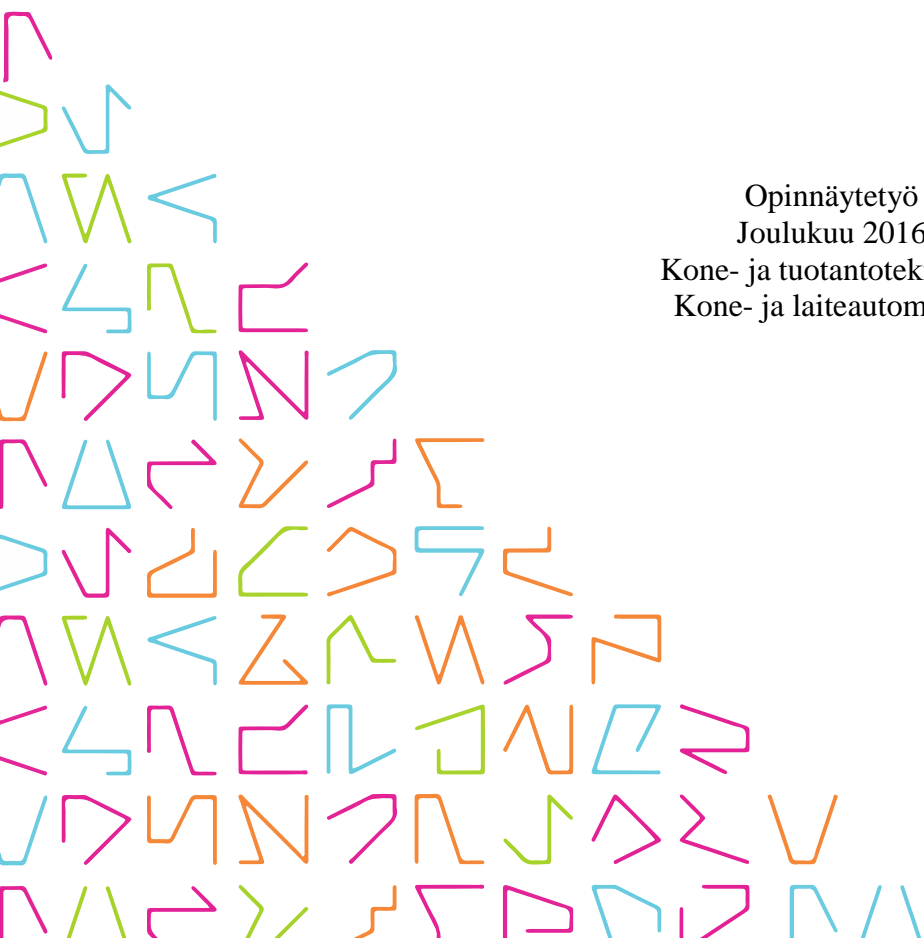


TAMPEREEN
AMMATTIKORKEAKOULU

Teollisuusrobotin varustaminen kunnonvalvonta- laitteiston demonstraatioalustaksi

Juuso Ojala

Opinnäytetyö
Joulukuu 2016
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio

OJALA, JUUSO:

Teollisuusrobotin varustaminen kunnonvalvontalaitteiston demonstraatioalustaksi

Opinnäytetyö 36 sivua, joista liitteitä 2 sivua
Joulukuu 2016

Tämä opinnäytetyö käsittelee robottijärjestelmää, sen käyttöönottoa sekä varustamista tuottavaan toimintaan. Opinnäytetyön tavoitteena oli tehdä robottisovellus tuotetestaukseen. Yrityksessä ei ollut aikaisempaa kokemusta roboteista, mikä aiheutti haasteita projektin edetessä ja teki siitä myös todella mielenkiintoisen ja opettavaisen. Opinnäytetyö koostuu käyttöönotosta, testausjärjestelmän suunnittelusta ja asennuksesta sekä järjestelmän kuvauksesta. Työ tehtiin Meluta Oy:lle.

Robotin tilasta ja kunnosta ei ollut aikaisempaa tietoa, joten projekti lähti liikkeelle pienin askelin. Alussa jouduttiin tekemään paljon tutkimustyötä ja tiedonhakua ennen varsinaisen työn aloittamista. Robotin taustasta ei ollut paljon tietoa, joten sen kunto tarkistettiin ja tutustuttiin sen toimintaan. Robotin käyttökuntoon saamiseksi robotille hankittiin myös työtila ja jalusta, jotka lainattiin Tampereen teknilliseltä yliopistolta. Yliopiston kanssa tehtiin muutenkin paljon yhteistyötä, koska yrityksen toimipiste sijaitsee samoissa tiloissa. Käyttöönottoon kuului myös huoltotoimenpiteet, kalibroinnit ja järjestelmän alkuasennukset sekä käsiohjauksen opettelu ja tiedonsiirtoväylien luonti.

Työn suurimmat haasteet aiheutuivat robotin ohjelmistosta. Robotin ohjausyksikkö ei ollut yhteensopiva uusimpien ohjelmistojen kanssa, joten jouduttiin toimimaan vanhojen järjestelmien kanssa ja soveltamaan aiemmin opittuja tietoja uusista järjestelmistä. Robotin muun laitteiston kanssa ei tullut suurempia ongelmia ja asennustyöt olivat suoraviivaisia ja sujuivat hyvin. Myös ohjelmointi sujui hyvin, kun ohjelmakielen perusteet olivat aiemmin opittuja, eikä suurempia ongelmia syntynyt.

Opinnäytetyön tavoitteena ollut tuotetestaussovelluksen laitteisto koostui paineilmajärjestelmästä ja siihen liitetystä tarraimesta, jotka suunniteltiin ja asennettiin alusta loppuun. Valmiin järjestelmän avulla voidaan testata tuotteen mekaanisia ominaisuuksia ja sen komponenttien kestävyyttä kestopesteillä, jotka simuloivat käyttöä teollisuusympäristössä.

Asiasanat: robottijärjestelmä, ohjelmointi, tuotetestaus, käyttöönotto

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Mechanical and Production Engineering
Machine Automation

OJALA, JUUSO:

Deployment of an Industrial Robot for Demonstrating Condition Monitoring Equipment

Bachelor's thesis 36 pages, appendices 2 pages
December 2015

This bachelor's thesis deals with a robot system, its commissioning and equipment for productive use. The aim of this thesis was to create a robotic application for product testing. There was no previous experience of robotic systems in the company which caused challenges during the project, but also made it a very interesting and educational experience for the author. The thesis discusses the deployment of the robot, designing and installation of a test system and a description of the robotic system. The thesis was made for Meluta Oy.

There was no information of the former state or condition of the robot so the project started with small steps. In the beginning there was lot of research and study to be done before the actual work could start. Because of the lack of information on the robot's earlier use, its condition had to be checked and the operational principles studied. In order to get the robot running it needed a proper working space and a pedestal that were borrowed from Tampere University of Technology. Furthermore, the collaboration with the university was intense because the company office was located at the same premises. Deployment also included service operations, calibrations and robot system installation together with learning the jogging of the manipulator and creating data connections with the robot controller.

The biggest challenges in this project were caused by the software of the robot system. The robot controller was not compatible with the latest computer software, so the author's knowledge on the new software had to be applied to the old one which caused a few setbacks. With the hardware of this project only minor problems occurred and the assembly was straightforward and went well. Also the programming went smoothly because the fundamentals of the programming language had been learned earlier and there were no significant problems

The hardware of the product testing application consists of a pneumatic system and a gripper mounted to the robot's manipulator. Those were designed and assembled from start to finish. With the complete testing system it is possible to test the mechanical features of the product and also the lifetime of components with endurance tests that simulate their use in industrial environments.

Key words: robot system, programming, product testing, deployment

SISÄLLYS

1	JOHDANTO.....	5
2	JÄRJESTELMÄN KUVAUS	7
2.1	Manipulaattori.....	7
2.2	Ohjainyksikkö	8
2.3	Käsiohjain	10
2.4	Ohjelmointi	11
3	TYÖN KULKU	14
3.1	Alkutilanne.....	14
3.2	Työn aloitus	14
3.3	Käyttöönotto	15
3.3.1	Ensikäynnistys	15
3.3.2	Kalibrointi	16
3.3.3	Käsiohjaus	21
3.3.4	Öljynvaihto.....	21
3.4	Tiedonsiirto	23
3.5	Oman robottijärjestelmän asennus.....	24
4	TUOTETESTAUS	26
4.1.1	Tuotekuvaus	26
4.1.2	Paineilmajärjestelmän suunnittelu	27
4.1.3	Ohjelmakoodi.....	28
5	POHDINTA.....	31
	LÄHTEET.....	34
	LIITTEET	35
	Liite 1. Tuotetestauksessa käytetty ohjelmakoodi.....	35

1 JOHDANTO

Teollisuudessa automaatio on yhä tärkeämpää tuotteen valmistuksen eri vaiheissa ja etenkin robotit ovat suuressa osassa tuotteen valmistuksessa ja testauksessa. Tämän opinnäytetyön tavoitteena oli varustaa yrityksen ostama teollisuusrobotti tuottavaan käyttöön. Yrityksellä ei ollut aiempaa kokemusta roboteista tai niiden käyttöönotosta, joten tarvittiin ulkopuolista apua. Olin keväällä suorittanut koulussa robotiikan kurssin, jonka aikana syttyi kiinnostus ja kipinä robottijärjestelmiin ja niiden ohjelmoitiin. Työn saadessani olin varma, että työstä tulee mielenkiintoinen ja haastava.

Opinnäytetyössä kuvataan vaiheittain robotin käyttöönotto, työkalun asennus ja ohjelmointi. Työn alussa kuvataan yleisesti järjestelmän eri osien perusominaisuuksia ja käyttöä. Tässä osiossa kerrotaan myös robotin ohjelmointikielestä ja sen käytöstä. Robotin käyttöönotto toteutettiin tutkimalla asennusohjeita ja muita vastaavanlaisia projekteja. Käyttöönottoon kuului ensikäynnistys ja sen yhteydessä tehtävä järjestelmän alkuasennus, kalibrointi ja öljynvaihto. Käyttöönoton osana oli myös robotin mukana tulleiden tietokoneohjelmistojen asentaminen ja niiden käytön opettelu. Kun robotti oli saatu käynnistettyä ja toimimaan kunnolla, alettiin tutustua sen ohjaukseen ja ohjelmointiin sekä varustaa robottia tuotetestaukseen. Testausta varten robottiin asennettiin paineilmajärjestelmä ja tarrain, jota ohjattiin digitaalisilla tuloilla ja lähdöillä. Testijärjestelmää varten haluttiin luoda oma robottijärjestelmä. Robottijärjestelmään voitaisiin luoda testausta varten oma ohjelmamoduuli ja tallentaa tiedot omia koordinaatistoja ja työkalutietoja varten. Robottijärjestelmän luominen kuitenkin muodostui työn suurimmaksi haasteeksi ja siitä on kerrottu työn kolmannessa luvussa.

Työn eri vaiheissa oltiin myös yhteistyössä Tampereen teknillisen yliopiston ja ABB:n kanssa. Etenkin Tampereen teknillisen yliopiston kanssa tehty yhteistyö oli hyödyllistä ja auttoi paljon työn valmiiksi saattamisessa. Sen avulla saatiin paljon esimerkkejä erilaisista robottikokoonpanoista esimerkiksi paineilmajärjestelmistä ja työkaluista, sekä niiden asennuksesta.

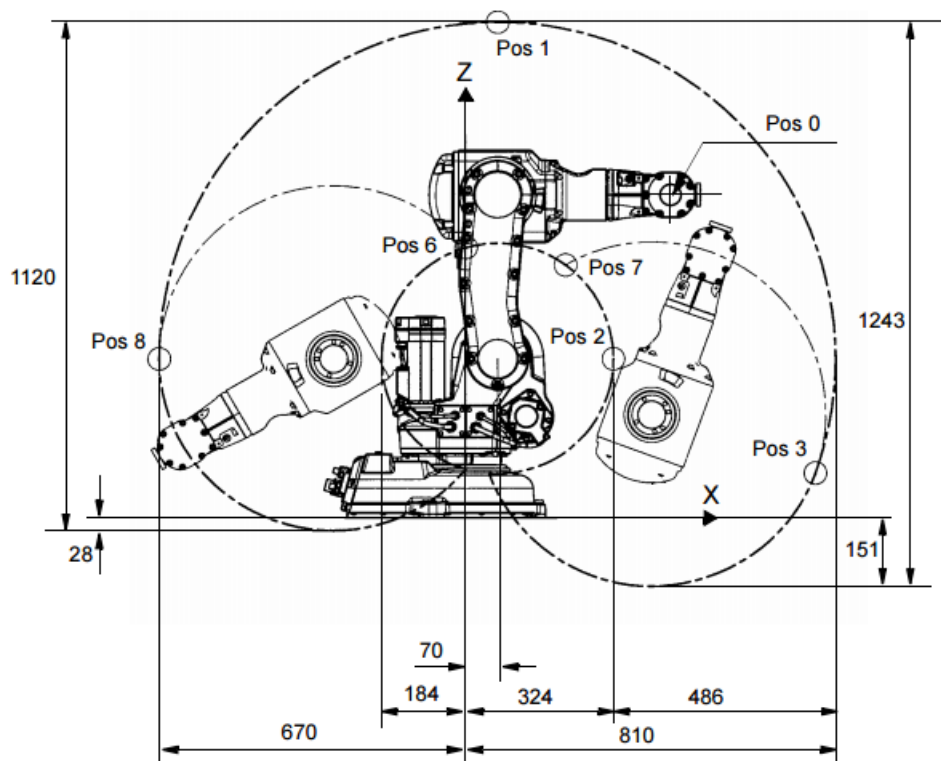
Tämän työn teettäjä oli Meluta Oy. Meluta Oy on vuonna 2015 Tampereella perustettu korkean teknologian start-up yritys, joka työllistää 12 henkilöä. Yrityksen ydinosia ovat sensoriteknologiat ja datan analysointi, erityisesti runkoäänten mittaus ja niille

toteutettavat algoritmit. Meluta Oy toteuttaa ratkaisuja kunnonvalvontaan ja tuotteiden elinkaaren seurantaan sekä digitalisointiin. Yritys on kehittänyt langattoman etäkuuntelulaitteen, joka helpottaa laitteiden kunnonvalvontaa ja vikojen havaitsemista. Laite tulee tuotantoon vuoden 2017 alussa. (<http://www.meluta.fi/>)

2 JÄRJESTELMÄN KUVAUS

2.1 Manipulaattori

Tässä työssä manipulaattorina oli ABB:n IRB 140, jossa on kuusi niveltä, joita ohjataan erillisillä servomoottoreilla. Manipulaattori on ABB:n malliston toiseksi pienin, sen kantokyvyn ollessa viisi kilogrammaa. (3, s.7) Kuvassa 1 on esitetty manipulaattorin ulottuvuus sivulta eri asennoissa. Kokonaisulottuvuus eteenpäin on 810 mm ja ylöspäin noin 1000mm. Mitat on otettava huomioon, kun määritetään manipulaattorin turva-alueita. Manipulaattori voidaan sijoittaa käyttötarpeen mukaan joko lattiaan, seinälle tai kattoon mihin kulmaan tahansa.



KUVA 1. Manipulaattorin ulottuvuus (3, s.13)

Manipulaattoria voidaan ohjata eri koordinaatistojen mukaan. Peruskoordinaatistossa origo on manipulaattorin alla. Peruskoordinaatistoa kutsutaan myös maailmakoordinaatistoksi. Robotilla voidaan luoda myös käyttäjäkoordinaatistoja, jotka kiinnitetään johonkin erilliseen työtasoon tai kohdekoordinaatistoja, jotka luodaan työn kohteena olevaan kappaleeseen. Robotissa on näiden lisäksi vielä työkalukoordinaatisto, jonka origo kutsutaan työkalupisteeksi. Työkalupiste on oletuksena 6. akselin keskellä, mutta se

voidaan uudelleen määrittää työkalun mukaan. Työkalukoordinaatisto on pallomainen, joten sillä voidaan pyörittää työkalua kohteen ympärillä ja saada näin haluttu kulma suhteessa kohteeseen. (2, s.18-19)

Manipulaattorin kiinnityksessä on otettava huomioon sen liikkeen aiheuttamat voimat. Kuvassa 2 on robotin alustaan aiheuttamat voimat. Kiinnityksessä on erityisesti otettava huomioon hätäpysäytyksessä aiheutuvat voimat, jotka voivat kasvaa suuriksi, mikäli robotti on täydessä vauhdissa. Robotissa on myös törmäystunnistin, joka pysäyttää manipulaattorin, sen törmätessä kiinteään esteeseen. Kun manipulaattori on pysähtynyt törmäyksessä, sen jarrut täytyy manuaalisesti vapauttaa ja manipulaattori irrottaa törmäyksen kohteesta. (3, s.14)

Mounting the manipulator

Maximum load in relation to the base coordinate system.

		Endurance load in operation	Max. load at emergency stop
Force xy	floor	$\pm 1300\text{N}$	$\pm 3200\text{N}$
	suspended	$\pm 1300\text{N}$	$\pm 3200\text{N}$
	wall	$\pm 2200\text{N}$	$\pm 3900\text{N}$
Force z	floor	$-1000 \pm 1000\text{N}$	$-1000 \pm 2000\text{N}$
	suspended	$+1000 \pm 1000\text{N}$	$+1000 \pm 2000\text{N}$
	wall	$\pm 1000\text{N}$	$\pm 2200\text{N}$
Torque Mxy		$\pm 1300\text{Nm}$	$\pm 2200\text{Nm}$
Torque Mz		$\pm 300\text{Nm}$	$\pm 750\text{Nm}$

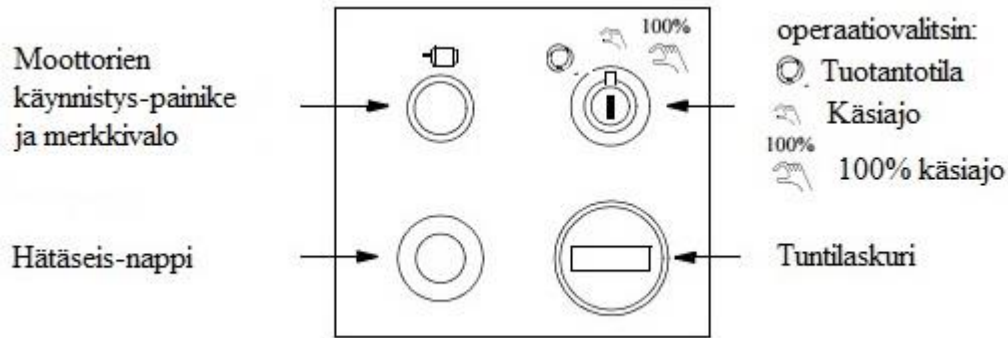
KUVA 2. Manipulaattorin kiinnitykseen aiheuttamat voimat (3,s.7)

2.2 Ohjainyksikkö

Järjestelmässä ohjainyksikkönä on ABB:n S4Cplus, joka ohjaa koko järjestelmän toimintaa. Ohjainyksikkö on järjestelmän tietokone, jonka muistissa on kaikki asennetut parametrit, tallennetut koordinaatit ja ohjelmat. S4C on jo tekniikaltaan, ohjelmistoltaan ja tietoliittymänsä osittain vanhentunut, mikä aiheutti haasteita työn edetessä. Esimerkiksi uusimmat ohjelmistot eivät olleet yhteensopivia ohjainyksikön kanssa.

Ohjainyksikköä ohjataan operaattoripaneelilla (KUVA 3), johon kuuluu

- pääkytkin
- hätäseis-painike,
- moottorien käynnistys- painike,
- operaatiovalitsin
- tuntilaskuri.



KUVA 3. Operaattoripaneeli (2, s.9)

Pääkytkimestä voidaan kääntää laitteen virta päälle tai pois. Hätäseis-painikkeella voidaan pysäyttää järjestelmä koska tahansa ja sen jälkeen se täytyy manuaalisesti vapauttaa, kun hätätilan aiheuttaja on poistettu. Moottorien käynnistys-painike kertoo järjestelmän tilan ja sillä käynnistetään moottorit tuotantotilassa. Kun painikkeen merkkivalo palaa jatkuvasti, järjestelmä on valmis suorittamaan ohjelman. Kun merkkivalo vilkkuu neljän Hertsin taajuudella, järjestelmää ei ole kalibroitu tai moottorien kierroslaskureita päivitetty. Silloin kun merkkivalo vilkkuu Hertsin taajuudella, järjestelmä on valmis ohjelman suorittamiseen, mutta moottorit eivät ole päällä. Operaatiovalitsimessa on kolme asentoa, tuotantotila, käsiajo ja 100% käsiajo. Tuotantotilassa robotti toimii automaattisesti suorittamalla siihen tehtyä ohjelmaa täydellä nopeudella. Käsiajo on ohjelmointia varten tehty tila, jossa manipulaattoria voidaan ohjailta käsiohjaimella. Tässä tilassa manipulaattorin maksiminopeus on rajoitettu 250mm/s. 100% käsiajo on tila, joka on tehty ohjelman testausta varten. Tässä tilassa nopeutta ei ole rajoitettu, mutta ohjelman suorittamiseksi täytyy pitää pohjassa käsiohjaimen ohjelman suoritus-painiketta (Hold-to-run), sekä sormikytkintä (Enabling device). Jos toisesta päästä irti, manipulaattori pysähtyy välittömästi. Tuntilaskuri näyttää manipulaattorin käyttötunnit. (2, s.9)

Ohjainyksiköllä voidaan ohjata yhteensä kuutta ulkoista akselia. Ulkoisella akselilla tarkoitetaan vaihtovirtamoottoria, jolla voidaan ohjata esimerkiksi manipulaattoria lii-

kuttavaa laitetta, kuljetinta tai kappaleenkäsittelypöytää. Ulkoisia akseleita hallitaan käsiohjaimella samalla tavalla kuin manipulaattoria. (2,s.20)

Tässä työssä käytettiin robotin digitaalista I/O-väylää. Robotissa on 16 digitaalista 24 V DC tuloa ja lähtöä. Kaikki tulot ja lähdöt voidaan nimetä erikseen käyttötarkoituksen mukaan ja niillä on fyysisesti eri osoitteet. Jokaisella tulolla ja lähdöllä on oma LED-valo, joka indikoi sen tilaa. Tuloja ja lähtöjä voidaan ohjata manuaalisesti yksi kerrallaan käsiohjaimella. Signaalit voidaan myös yhdistää ryhmäksi ja käyttää esimerkiksi viivakoodina. Robotissa voidaan käyttää myös simuloituja tuloja ja lähtöjä, joilla voidaan simuloida ristiin kytkentöjä tai loogisia komentoja ilman varsinaisten komponenttien asennusta. Optiona ohjausyksikköön saa myös digitaalisen 120V AC I/O -väylän, analogisen $\pm 10V$ I/O:n, rele I/O:n sekä Interbus että Profibus- väylät. (2, s.22-25)

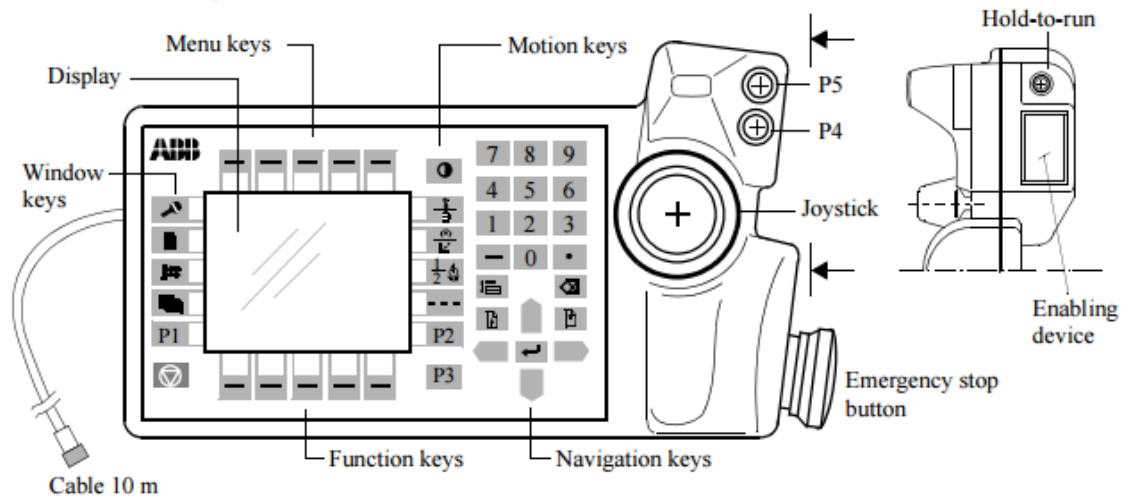
2.3 Käsiohjain

Käsiohjain (KUVA 4) toimii robottijärjestelmän käyttöliittymänä. Sillä voidaan tehdä käytännössä kaikki, mitä robotilla tarvitsee tehdä. Sillä voidaan mm. manuaalisesti ohjata robottia, tehdä sekä muokata ohjelmia, vaihtaa asetuksia ja hallita ulkoisia laitteita. Käsiohjaimen pääkomponentit ovat näyttö, ohjaussauva (joystick) ja sormikytkin (enabling device). Käsiohjaimessa on myös hätä-seis – painike, jolla järjestelmä voidaan pysäyttää milloin tahansa. (2, s.7-8)

Ohjaussauvaa käytetään manipulaattorin käsiohjaukseen. Ohjaussauvaa toimii portaattomasti kaikkiin suuntiin ympyrän muotoisella alueella sekä pyörii akselinsa ympäri. Manipulaattoria manuaalisesti ohjattaessa nopeus kasvaa sitä suuremmaksi mitä kauemmaksi keskikohdasta sauva viedään. Manipulaattoria voidaan ohjata kolmella tapaa: nivel kerrallaan, lineaarisesti koordinaatiston mukaan ja työkalupisteen ympäri. Ohjaimella robotti tiettyyn pisteeseen, sille voidaan opettaa pisteitä ja lisätä ne ohjelmaan, jotka toimivat usein ohjelman runkona. (2, s.8)

Sormikytkin on eräänlainen ”kuolleenmiehen kytkin”, jolla moottorit aktivoidaan käsiajotilassa. Moottoreiden aktivoinnin jälkeen manipulaattoria voidaan ohjata ohjaussauvalla. Kytkimessä on kolme asentoa. Puoleen väliin painettuna moottorit menevät päälle ja manipulaattoria voidaan ohjata. Kun kytkin on painettuna täysin pohjaan tai

päästetty ylös, moottorit ovat pois päältä ja manipulaattori pysähtyy välittömästi. Tämä tekee käytöstä turvallisempaa. (2, s.8)



KUVA 4. Käsiohjain (2, s.7)

2.4 Ohjelmointi

Robotteja voidaan ohjelmoida monella eri tavalla. Yleisimmät tavat ovat opettamalla ohjelmointi, etäohjelmointi ja näiden yhdistelmät. Opettamalla ohjelmoinnissa robotti ajetaan manuaalisesti tiettyyn asemaan, joka tallennetaan robotin muistiin. Asemien välinen liiketyyppi ja nopeus voidaan erikseen määrätä. Etäohjelmointia käytetään, kun robotti ja sen ympäristö voidaan mallintaa, jolloin robottia ajetaan simuloituna eikä tuotantoa tarvitse pysäyttää ohjelmoinnin ajaksi. Monesti käytetään näiden yhdistelmää, jossa ohjelman runko muodostuu opetetuista asemista ja muut komennot lisätään ohjelmaan ohjelmaeditorilla.

ABB:n roboteissa käytetään RAPID-ohjelmointikieltä. RAPID-ohjelmat ovat tekstitiedostoja. Niitä voidaan helposti muokata tavallisella tekstieditorilla tai ohjelman muokkaukseen tarkoitetuilla tietokoneohjelmalla. Ohjelmamoduuli rakentuu ohjelmadatasta, pääohjelmista ja aliohjelmista. Ohjelmadata koostuu esimerkiksi tallennetuista paikkatiedoista ja numeerisista arvoista kuten laskureista. Pääohjelma on ohjelman runko ja määrää järjestyksen, jossa ohjelmaa suoritetaan. Pääohjelma voi sisältää useita aliohjelmiä, jotka usein sisältävät jonkin tietyn toiminnon ohjelman sisällä. Esimerkiksi pinoamissovelluksessa alirutiinit voitaisiin muodostaa kappaleen noutamisesta ja pinoa-

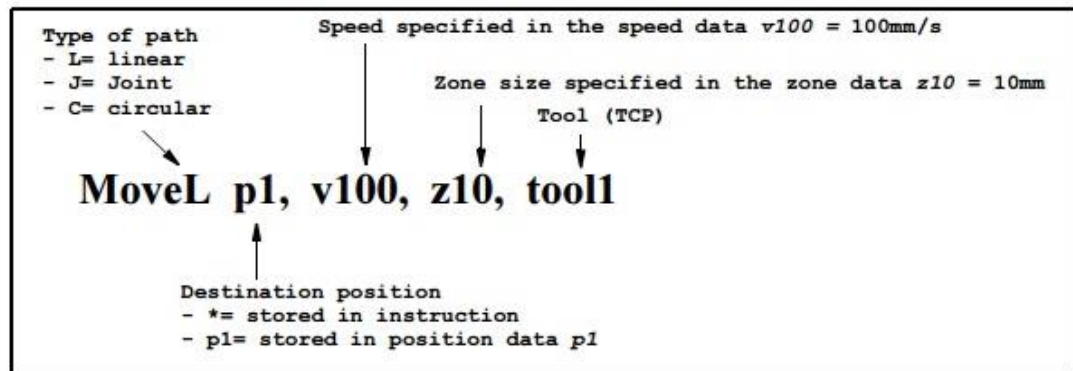
misesta, jolloin ohjelma on helpompi jäsenellä ja mahdolliset virheet on helpompi löytää. Samassa muistissa ohjelman kanssa toimii myös järjestelmämoduulit, jotka sisältävät enemmänkin asennukseen, kuten työkaluihin ja huoltotoimiin liittyviä ohjelmia. (5, lk.6, s5-6)

Ohjelmat sisältävät käskyjä, jotka suoritetaan yksi kerrallaan rivijärjestyksessä. Ohjelmakäsky voi olla esimerkiksi liikekäsky, tulon tai lähdön asettaminen, työkalun vaihto tai alirutiiniin siirtyminen. Yleisimpiä ohjelmakäskyjä ovat:

- MoveJ Työkalupiste liikkuu nopeinta ja helpointa reittiä pisteeseen
- MoveC Työkalupiste liikkuu tehden kaaren kolmen pisteen välille
- MoveL Työkalupiste liikkuu lineaarisesti pisteeseen
- Set Asettaa digitaalisen lähdön arvoksi yksi
- Reset Asettaa digitaalisen lähdön arvoksi nolla
- WaitTime Odottaa asetetun ajan ennen kuin jatkaa ohjelman toteuttamista
- For Toistaa ohjelmakierron niin monta kertaa kuin halutaan
- IF Ehtojen täytyessä käsky toteutetaan
- ProcCall Kutsu toinen rutiini
- Return Palaa alkuperäiseen rutiiniin
- := Asettaa muuttujalle arvon

Kuvassa 5 on kuvattu tavallinen liikekäsky, jossa määritellään liikkeen tyyppi, kohdepiste, nopeus, tarkkuus ja käytetty työkalu. Liikkeen tyyppi on MoveL, jolloin työkalupiste liikkuu lineaarisesti haluttuun kohdepisteeseen. Kohdepiste voi olla ohjelman muistiin tallennettu vakio, jota voidaan käyttää useaan kertaan. Mikäli kohdepistettä käytetään vain kerran ja sitä ei ole tallennettu, se näytetään koordinaatteina. Liikekäskyssä nopeus on merkitty suoraan millimetreinä sekunneissa, joka on helppo ymmärtää ja säätää sopivaksi. Kuvan 5 esimerkissä nopeudeksi on asetettu v100, mikä tarkoittaa, että liikkeen nopeus on 100 mm/s. Liikkeen tarkkuus merkitään vyöhykkeinä (Zone size). Vyöhyke tarkoittaa sitä, kuinka lähelle työkalupiste tulee kohdepistettä, ennen kuin aletaan toteuttaa seuraavaa ohjelmakomentoa. Esimerkissä on z10, jolloin työkalupisteen ollessa kymmenen millin päässä kohdepisteestä, se alkaa toteuttaa seuraavaa käskyä. Kun halutaan saada työkalupiste tarkasti kohteeseen, käytetään arvoa ”fine”. Viimeisenä näkyy käytetty työkalu, jonka tiedot on tallennettu järjestelmään. Jos ohjel-

massa käytetään käyttäjäkoordinaatistoa, se on määritelty työkalun jälkeen merkinnällä ”WObj”, jonka perässä on käyttäjäkoordinaatiston nimi. (5, lk7, s.3-5)



KUVA 5. Tavallinen liikekäsky RAPID-ohjelmakielessä. (5, lk7, s.3)

3 TYÖN KULKU

3.1 Alkutilanne

Meluta Oy oli ostanut käytetyn käsivarsirobotin, joka oli aikaisemmin ollut hitsauskäytössä, ja jonka kunnosta ei ollut tarkkaa tietoa. Alussa työllä ei ollut vielä tarkkaa tavoitetta, vaan työ aloitettiin robotin kunnan tarkistamisella. Sen jälkeen suunniteltiin, miten siitä saataisiin tuottava ja yrityksen toimintaa tukeva. Työn edetessä suunnitelmaksi muodostui tehdä testausjärjestelmä Meluta Oy:n kehittämälle etäkuuntelulaitteelle. Testin tarkoituksena on ottaa selville, miten etäkuuntelulaitteen komponentit ja liimaukset kestävät toistuvaa pudottamista ja kiinnityksessä käytettävän magneetin aiheuttamia voimia.

3.2 Työn aloitus

Työ käynnistettiin robotin kunnan tarkistamisella. Sekä manipulaattorin, että ohjainyksikön ulkoinen kunto tarkistettiin. Ohjainyksikön sisällä oli paljon toisesta päästä kytkemättömiä sähköjohtoja, joiden tarkoitus tutkittiin ja tarkistettiin myös muut kytkennät. Kävi ilmi, että kytkemättömät johdot oli tarkoitettu ulkoisten laitteiden ohjaukseen ja ne oli kytketty ohjausyksikön ulkopistokkeisiin, joten niistä ei aiheutunut vaaraa. Tämän jälkeen aloitettiin robotin mukana tulleen materiaalin tutkiminen, joista tärkeimmät olivat käyttö- ja asennusohjeet sekä tietokoneohjelmistot.

Meluta Oy:llä ei vielä tässä vaiheessa ollut paikkaa, mihin robotti voitaisiin sijoittaa eikä jalustaa, johon se voitaisiin asentaa. Alettiin selvittää mistä voitaisiin saada sopiva laboratoriotila robotille ja päädyttiin ottamaan yhteyttä Tampereen teknillisen yliopiston tilapalvelujohtajaan. Tätä kautta käyttöön saatiin juuri uusittu laboratoriotila, joka oli tyhjillään. Kun olimme siirtäneet robotin laboratorioon, kohdattiin pieniä haasteita vanhan ohjainyksikön kanssa. Ohjainyksikkö tarvitsee 32A-pistokkeen, jollainen olikin laboratoriotilassa, mutta sitä ei oltu kytketty. Tämän johdosta jouduttiin ottamaan yhteyttä teknillisen yliopiston huoltoon, jotta saisimme robotin toimimaan. 32A-vaatimus tuli yllätyksenä, koska nykyään suurimmalle osalle laitteita riittää 16A, kuten esimerkiksi uusimpaan ABB:n IRC5-ohjainyksikköön.

Saman kontaktin kautta pääsimme käsiksi myös valmiiseen robottipöytään, joka saatiin lainaksi kesän ajaksi. Pöytä saatiin teknillisen yliopiston koneosastolta, josta yksi robottipöytä oli jäänyt ylimääräiseksi. Pöytä oli rakenteeltaan kaksitasoinen alumiinirunkoinen robottipöytä, jossa pöytätasona oli raskas metallilevy. Pöydän alemmalle tasolle oli suunniteltu laitettavaksi ohjainyksikkö, joka vakauttaisi pöytää, mutta tässä tapauksessa se ei onnistunut ohjainyksikön suuren koon vuoksi. Tästä syystä pöytä oli hieman kevyt, mutta projektissa ei ajateltu käytettävän suuria liikenopeuksia tai liikelaajuuksia, joten pöytä sopisi työn tarkoitukseen. Robottipöydän saaminen oli erittäin tärkeä askel projektille ja mahdollisti robotin asennuksen ja käyttöönoton. Tämän pöydän pohjalta saatiin myös ideoita ja malleja oman robottipöydän suunnitteluun myöhemmässä vaiheessa.

3.3 Käyttöönotto

3.3.1 Ensikäynnistys

Robotti oli ollut pitkään käyttämättömänä, eikä sen tilasta tiedetty mitään ennen ensimmäistä käynnistystä. Ensi käynnistyksen yhteydessä on kolme tapaa tehdä järjestelmän alkuasennus: hiljainen tila (silent mode), helppo kysely (easy query mode) ja kysely (query mode). Hiljainen tila lataa automaattisesti robottijärjestelmään ladatut asetukset. Helppo kysely-tilassa voidaan valita kieli, poistaa valittuja optioita, sekä valita huolto- tai normaali-tila. Kysely-tilassa voidaan valita robotin malli, DC-linkki, poistaa valittuja optioita, valita kieli ja valita huolto- tai normaali-tila. Näistä valittiin kysely, koska kyseessä oli ensi käynnistys, eikä tiedetty minkälainen robottijärjestelmä oli käytössä. Haluttiin myös varmistua siitä, että valitaan oikeat komponentit ja asetukset, jotta järjestelmä toimisi oikein. Kyselyssä robotin malliksi valittiin IRB 140, DC-linkiksi valittiin DC2T, mikä voidaan tarkistaa ohjainyksikön sisällä olevan komponentin kyljestä. Kieleksi valittiin englanti ja tilaksi standardi. (6, s.87-88)

Kun järjestelmä oli käynnistetty, tarkistettiin järjestelmän ilmoittamat vikakoodit ja ne pyrittiin poistamaan. Tämän jälkeen voitiin ladata manipulaattori- ja järjestelmäparametrit, jotka olivat erillisillä robotin mukana tulleilla levykkeillä. Manipulaattoripa-

rametreihin kuuluu esimerkiksi kalibroinnin offset-arvot ja järjestelmäparametreihin kuuluu esimerkiksi I/O-signaalit. Samalla saatiin myös tutustua käsiohjaimen käyttöliittymään ja muokata asetuksia haluamukseen.

3.3.2 Kalibrointi

Käynnistyksen jälkeen robotti täytyi kalibroida, jotta sitä voitaisiin käyttää tarkoituksen mukaisesti. Kalibrointi aloitettiin robotin hienokalibroinnilla. Alkuperäinen hienokalibrointi on suoritettu jo tehtaalla, missä jokainen servomoottori on kalibroitu erikseen ja niistä saadut offset arvot annetaan robotin mukana. Käynnistyksen yhteydessä hienokalibroinniksi riittää siis syöttää servomoottoreiden offset-arvot parametritiedostona. Tämän jälkeen kalibrointi voidaan tarkistaa ohjelmalla, joka ajaa moottorit nolla-asentoon MoveAbsJ-ohjelmakomennon avulla:

```
MODULE CALPOS
PROC main()
    !-Robotin nollapisteen tarkastusohjelma
    MoveAbsJ
    [[0,0,0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]]\NoEOffs,
    v200,fine,tool0;
    Break;
    !-Tarkasta akselimerkit [1-6]
ENDPROC
ENDMODULE.
```

Ohjelman alussa on moduulin nimi ja pääohjelman aloituskomento. Tämän jälkeen kuvaus ohjelmasta, joka ei vaikuta ohjelman kulkuun. MoveAbsJ-komento ajaa manipulaattorin nivelet absoluuttiseen nollapisteeseen. Ensimmäiset kolme nollaa hakasulkeiden sisällä tarkoittavat koordinaatteja x,y ja z, toiset kolme nollaa ovat kulma-arvoja. Tämän jälkeen ovat järjestelmässä olevat vakioarvot moottoreille. ”NoEOffs” (no external offsets) tarkoittaa, ettei komentoon vaikuta mitkään ulkoiset alkuarvot. Lopuksi komennossa on nopeus, tarkkuus ja käytetty työkalu, jotka ovat kaikissa tavallisissa liikekomennossa. ”Break”-komennolla saadaan ohjelmatoteutus pysähtymään välittömästi, jolloin manipulaattori pysähtyy varmasti oikeaan kohtaan. (7, s.9, s.149)

Jos offset-arvot ovat oikeat, manipulaattori liikkuu kalibrointiasentoon (KUVA 6). Kalibrointiasennossa voidaan tarkistaa, että manipulaattorin kalibrointipisteet ovat kohdallaan (KUVAT 7-12). Kun kalibrointipisteet on tarkistettu, voidaan päivittää moottoreiden kierroslaskurit. Kierroslaskureiden päivitys tehdään käsiohjaimen huoltovälilehdeltä, jossa kaikki akselit voidaan päivittää samaan aikaan tai akseli kerrallaan.



KUVA 6. Robotti kalibrointiasennossa.

Kuvassa 6 manipulaattori on kalibrointiasennossa, jossa kalibrointipisteet voidaan tarkistaa. Siinä työkalupisteen kaikki koordinaattiarvot ovat nollija ja myös servomoottoreiden kulma-arvot ovat nollija. Manipulaattori voidaan ajaa tähän asentoon myös käsiohjaimella nivel kerrallaan ja riittää, että kalibrointipisteet tarkistetaan silmämääräisesti.



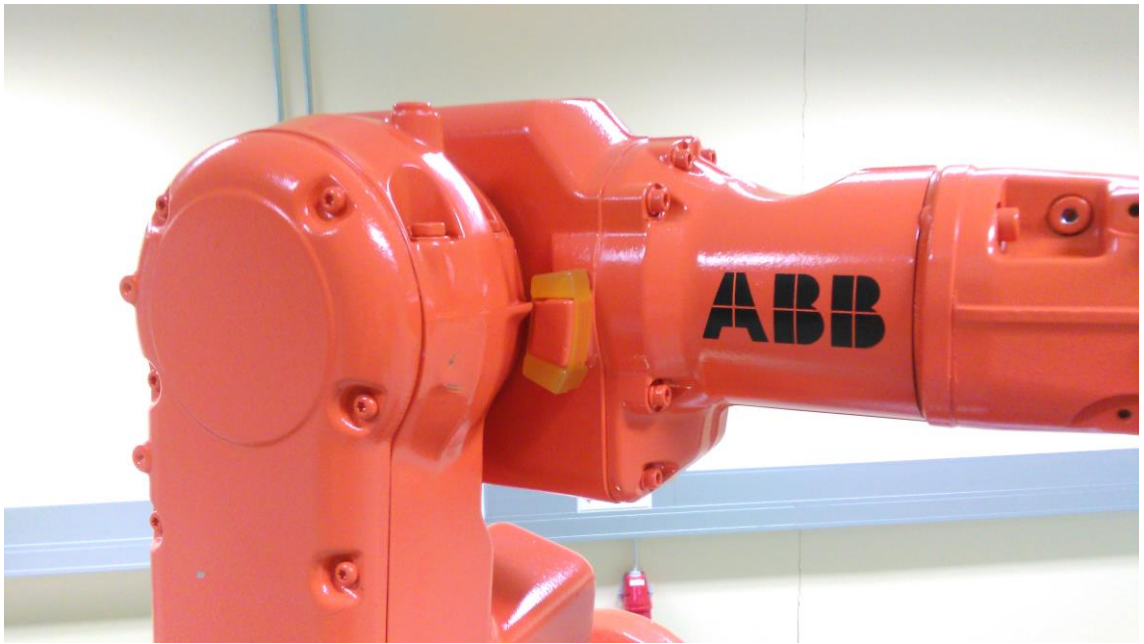
KUVA 7. Nivelen 1 kalibrointipiste.

Kuvassa 7 on nivelen yksi kalibrointipiste. Tässä asennossa manipulaattori on ajettu asentoon, jossa jalustassa olevat kaksi tappia on samassa linjassa ensimmäisen nivelen merkkitaipin kanssa.



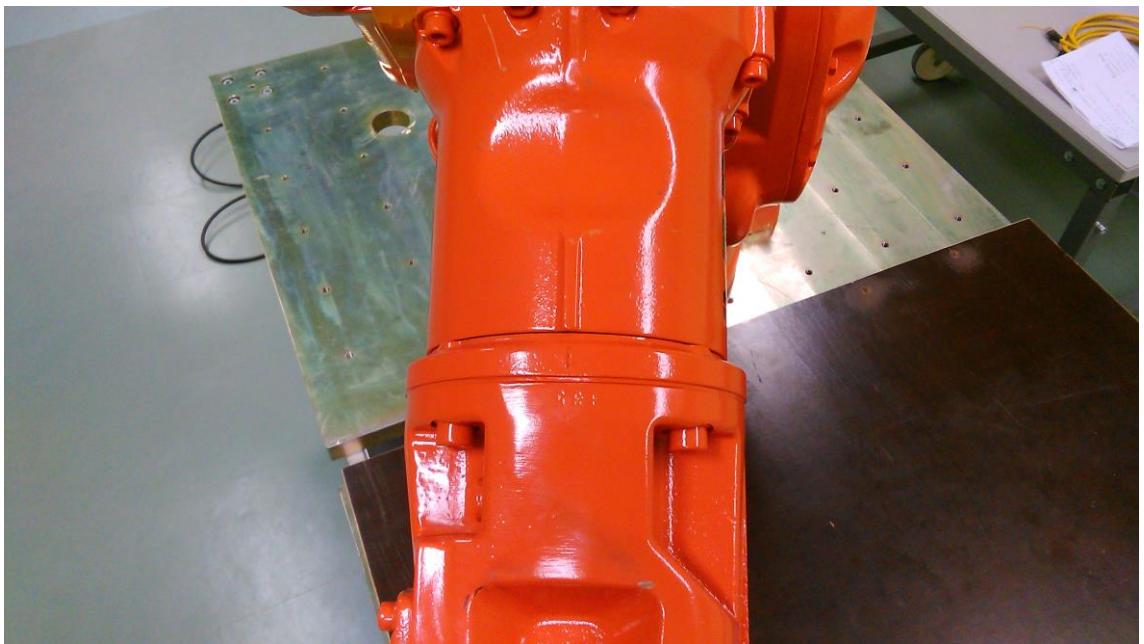
KUVA 8. Nivelen 2 kalibrointipiste.

Kuvassa 8 on toisen nivelen kalibrointipiste. Siinä toinen nivel on ajettu pystyasentoon ja kalibrointimerkkinä toimii pysäyttimen ruuvi, joka on moottorikotelossa olevan merkin keskellä.



KUVA 9. Nivelen 3 kalibrointipiste.

Kuvassa 9 on kolmannen nivelen kalibrointipiste. Siinä kolmannen nivelen servomoottorikotelossa oleva merkkikieleke on ajettu pysäyttimessä olevan loven kohdalle.



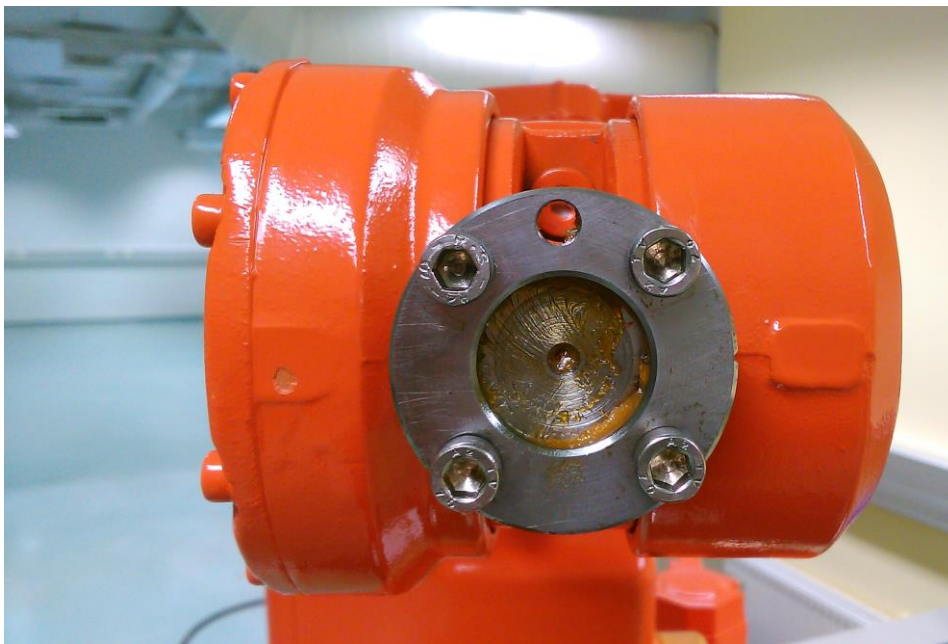
KUVA 10. Nivelen 4 kalibrointipiste.

Kuvassa 10 on nivelen neljä kalibrointiasento. Tässä asennossa neljännessä nivelessä oleva hahlo ajetaan akselikotelon keskellä olevan sauman keskikohtaan.



KUVA 11. Nivelen 5 kalibrointipiste

Nivelen 5 kalibrointipiste on näytetty kuvassa 11. Nivel ajetaan asentoon, jossa siihen jysitty hahlo on kohdakkain moottorikotelon hahlon kanssa.



KUVA 12. Nivelen 6 kalibrointipiste.

Kuvassa 12 on 6 nivelen kalibrointipiste. Kuudennen nivelen kalibroinniksi työkalulaippa pyöritetään asentoon, jossa sen sivulla oleva lovi on samassa linjassa 5 nivelen moottorikotelon sauman kanssa.

3.3.3 Käsiohjaus

Kalibroinnin jälkeen robottia pystyi ohjaamaan vapaasti käsiohjaimella ja päästiin opettelemaan robotin manuaalista ohjausta. Erityisen hyödyllistä tulevan ohjelmoinnin kannalta oli opetella ohjaamaan robottia eri koordinaatistojen mukaan, jolloin robottia voi liikuttaa niin, että robotin työkalu pysyy samassa asennossa. Koska robotti asennettiin pöytään pöydän suuntaan nähden vinoon, robotille opetettiin uusi pöydän suuntainen käyttäjäkoordinaatisto, mikä helpottaa robotin ohjaamista. Uusi koordinaatisto opetettiin näyttämällä ja tallentamalla kolme pistettä koordinaattiakselien suunnassa, kaksi x-akselilta ja yksi y-akselilta, pöydän pinnalta. Näitä koordinaatistoja voidaan tallentaa järjestelmään useita ja niitä voidaan vaihtaa milloin vain. (5, lk.8, s.23-27) Ensimmäiseksi ohjelmaksi tehtiin robotin ajo kalibrointiasentoon. Ajateltiin, että pitkän seisomisen jälkeen tietokoneen akku olisi ”kuoleentunut” ja sammutuksen jälkeen kierros-laskurit pitäisi päivittää uudestaan.

3.3.4 Öljynvaihto

Vaikka robotti oli lähes käyttämätön, siihen päätettiin tehdä öljynvaihto pitkän seisonnan takia. Öljy ei kulu seisonnan aikana, mutta sen sisältämät mineraalit vanhenevat ja sen voitelukyky heikkenee. Ohjekirjassa öljynvaihtoväliksi 5. ja 6. niveleen on merkitty kolme vuotta, jonka jälkeen öljyt tulisi vaihtaa käyttötunneista huolimatta. Muihin niveliin öljyä ei tarvitse vaihtaa. 5. ja 6. niveleen kohdistuu suurin paine ja rasitus, kun työkalulla on painoa, joten siinä vaihteistolla on suurin kulutus ja öljynvaihto on tarpeen. Öljyn vaatimuksina oli vaihteistoöljy, jonka viskositeetti on $320 \text{ mm}^2/\text{s}$ ja jossa oli äärimmäisen paineen kestävät ominaisuudet. Öljyksi valittiin Mobilgear 600 XP 320, joka on suunniteltu suljettuihin hammaspyörävaihteistoihin ja raskaille kuormituksille. (8, 2.9.2016)

Muilta osin robotti on tehty erittäin huoltovapaaksi. Huolto koostuu pääosin käytön aikaisesta tarkkailusta ja puhdistuksesta. Öljyn vaihdon lisäksi robottiin tehtäviä säännöllisiä huoltoja ovat tietokoneen jäähdyttimen suodattimen vaihto, tietokoneen vara-akun vaihto ja tuulettimen vaihto. (3, s.12)



KUVA 13. Öljysäiliön tyhjennys

Kuvissa 13 ja 14 on kuvattu manipulaattorin öljynvaihto. Nivelillä viisi ja kuusi on sama öljysäiliö, jossa on kaksi tulppaa öljynvaihtoa varten. Öljysäiliön tyhjennetään ajamalla manipulaattori kuvan 13 mukaiseen asentoon ja avaamalla molemmat tulpat öljysäiliöstä. Öljysäiliö täytetään neljännen nivelen tulpasta. Myös toinen tulppa on hyvä pitää auki, jotta öljysäiliö täyttyy tasaisesti ja vältetään ilmakuplien muodostumista. Öljysäiliö täytetään niin, että öljyn yläpinta on öljytulpan kierteen tasolla.



KUVA 14. Öljysäiliön täyttö.

3.4 Tiedonsiirto

Kommunikointi ohjainyksikön kanssa tapahtuu ethernet-yhteyden välillä. Alussa oli haasteita käsiohjaimen käyttöliittymän kanssa, kun ei tiedetty miten robotin IP-osoitetta saadaan muokattua. Saatiin selville, että ohjainyksiköllä on erilaisia käynnistyksiä, joilla voidaan muokata ohjainyksikön perusasetuksia ja poistaa esimerkiksi järjestelmätiedostoja. Eri käynnistyksiä ovat

- normaali restart, uudelleen käynnistää järjestelmän ilman muutoksia
- I-start, palauttaa tehdasasetukset ja poistaa esimerkiksi parametrit ja tallennetut asetukset
- P-start, poistaa kaikki ohjelmat ja ohjelmamoduulit järjestelmästä
- C-start, poistaa aktiivisen robottijärjestelmän, jonka jälkeen voidaan asentaa oma luotu järjestelmä parametreineen ja asetuksineen
- X-start, voidaan vaihtaa aktiivista robottijärjestelmää, ladata järjestelmä, ja muokata yhteysasetuksia

Järjestelmälle suoritettiin X-start, joka aukaisee käynnistysikkunan. Käynnistysikkunasta voidaan valita verkkoasetukset, josta voitiin vaihtaa IP-osoite ja aliverkkopeite. Tämän jälkeen järjestelmä voitiin yhdistää tietokoneeseen, mikä helpotti suuresti tiedostonhallintaa. Ennen verkkoyhteyttä ainoa keinoa saada varmuuskopioita järjestelmästä

tai ladata tiedostoja järjestelmään oli käyttää levykkeitä, joita ei voida edes lukea moderneilla tietokoneilla ilman lisävarusteita. (5, lk12 s.17-21)

Kun verkkoyhteys saatiin luotua, tiedostojen hallinnassa käytettiin FTP:tä. Ohjelmana käytettiin WinSCP:tä, jonka kautta tiedostojen siirto onnistuu vaivattomasti. FTP:n käyttö helpotti ohjelmointia huomattavasti, koska RobotStudio ei ollut suoraan yhteensopiva ohjainyksikön kanssa. FTP:n kautta kuitenkin ohjelmat saatiin siirrettyä RobotStudioon, jossa ohjelman muokkaus ja virheenkorjaus on helppoa. Tiedonsiirto onnistuu myös NFS:llä tai 3 ½ tuuman levykkeillä.

3.5 Oman robottijärjestelmän asennus

Kun yhteys ohjainyksikön kanssa oli saatu luotua, päätettiin luoda oma robottijärjestelmä. Robottijärjestelmään kuuluu käyttöjärjestelmä RobotWare ja ladatut parametri- ja konfiguraatiotiedostot. Järjestelmä luodaan ja ladataan RobotInstaller-tietokoneohjelmalla, sillä voidaan myös päivittää järjestelmiä. Robottijärjestelmä vaatii 25MB tilaa kovalevyllä. Järjestelmän luomiseksi RobotInstallerilla tarvittiin ABB:n lisenssiavain. Lisenssiavain ei tullut robotin mukana ja jouduttiin ottamaan yhteyttä ABB:n asiakaspalveluun. Asiakaspalvelusta ei saatu tarvittavaa avainta, mutta selvisi, että avain saataisiin selville, kun robotista ladattu backup-tiedosto avataan RobotInstallerin järjestelmäpäivitys-ikkunassa. Lisenssiavaimen avulla voitiin aloittaa järjestelmän luominen, kunnes huomattiin, ettei robotin mukana tulleissa materiaaleissa ollut tarvittavia tiedostoja uusimman RobotWare-version lataamiseksi. Otettiin jälleen yhteyttä ABB:lle, josta saatiin tarvittavat tiedostot ja järjestelmä voitiin luoda loppuun.

Tilan tarve aiheutti ongelmia, koska kovalevyllä ei ollut tarpeeksi vapaata muistia. Muistin vapauttamiseksi alettiin selvittää, mitä tiedostoja voidaan poistaa ja mitkä tiedostot ovat järjestelmän toiminnan kannalta tärkeitä. Aluksi luotiin varmuuskopiot kaikista tiedostoista, jotta välttyttäisiin peruuttamattomilta virheiltiltä.

Tilan lisäämiseksi järjestelmälle suoritettiin P-start, mikä poistaa kaikki asennetut ohjelmat ja ohjelmamoduulit. Tämä ei auttanut tilannetta, koska poistetut tiedostot olivat niin pieniä. P-startin jälkeen tehtiin C-start, joka poistaisi käytössä olevan robottijärjestelmän ja siihen linkitetyn käyttöjärjestelmän version. Tämän jälkeen tilaa vapautui,

mutta ei tarpeeksi oman järjestelmän asentamiseksi. Jotta robotti saataisiin uudelleen käyntiin, siihen asennettiin levykkeiltä robotin mukana tullut alkuperäinen tehtaalla luotu robottijärjestelmä. Tehtaalla luodussa robottijärjestelmässä oli vanha versio käyttöjärjestelmästä, mikä oli juuri päinvastaista, mitä haluttiin. Myös parametrien asennus ja kalibrointi piti tehdä uudestaan.

Huomattiin, että tilaa ei saatu tarpeeksi, koska kovalevyllä oli ylimääräinen RobotWare-käyttöjärjestelmä, joka ei ollut linkitetty mihinkään järjestelmään. Tämä RobotWare poistamalla saataisiin tarpeeksi tilaa oman järjestelmän lataamiseksi. Tiedostoa yritettiin poistaa FTP:llä, mutta kävi ilmi, että se on kirjoitussuojattu, joten sitä ei voi muokata tai poistaa. Tässä kohtaa otettiin yhteyttä ABB:lle ja kysyttiin, voisivatko he neuvoa, miten asian kanssa toimitaan. ABB:ltä saatiin erilaisia ohjeita tiedoston poistamiseksi lähinnä tekemällä C-start eri robottijärjestelmillä, mutta mikään näistä ei toiminut. Lopulta he tarjoutuivat tyhjentämään kovalevyn ja asentamaan uuden käyttöjärjestelmän toimipisteellään. Tähän ei kuitenkaan haluttu tarttua, koska ei tiedetty kuinka kauan siinä kestäisi ja projekti haluttiin saada tavoitteeseen aikataulun puitteissa. Vaikka uutta järjestelmää ei saatu asennettua, alkuperäinen robottijärjestelmä oli täysin toimiva ja sitä voitaisiin käyttää ohjelmointiin ja projektin loppuun saattamiseen.

4 TUOTETESTAUS

4.1.1 Tuotekuvaus

Opinnäytetyön tavoitteeksi muodostui tuotetestaukseen soveltuvan robottisovelluksen luominen. Tuote on Meluta Oy:n kehittämä etäkuuntelulaite, joka on suunniteltu koneen runkoäänien kuunteluun. Sitä käytetään koneiden kunnonvalvonnassa havaitsemaan ja paikantamaan vikoja, jotka aiheuttavat normaalista poikkeavaa ääntä esimerkiksi laakerivioissa. Etäkuuntelulaite eristää runkoäänet muusta taustamelusta, mikä tekee kuuntelusta ja vian havaitsemisesta tarkempaa. Vian havaitseminen perustuu käyttäjän subjektiiviseen ammattitaitoon ja kokemukseen koneen käytöstä.

Etäkuuntelulaite on kooltaan melko pieni ja magneetin avulla kiinnitettävissä haluttuun paikkaan koneen rungossa ja se on suunniteltu kestävästi teollisuuden olosuhteet. Etäkuuntelulaite havaitsee runkoäänet pietsoanturin avulla, joka havaitsee pinnan värähtelyn ja toimii etäkuuntelulaitteen mikrofonina. Ääntä voi kuunnella langattomasti kuulokkeilla bluetooth yhteyden avulla. Kuulokkeiden avulla saavutetaan laadukas kuunteluelämys ympäristössä. Langaton kuuntelu parantaa myös turvallisuutta, koska laitetta kuunneltaessa ei tarvitse olla sen toiminta-alueella, toisin kuin tavallisella stetoskoopilla. Avoimessa tilassa kantomatka on 150 metriä ja akun kesto on noin 11 tuntia. Laitteen käyttöliittymä koostuu virtanapista, bluetooth paritus-napista ja äänenvoimakkuuden säädöstä. Tuotetestaus on tarkoitettu testaamaan pietsoanturin liimauksen kestävyttä, mutta samalla siinä seurataan myös muuta mekaanista kestävyttä.



Kuva 15. Etäkuuntelulaite

4.1.2 Paineilmajärjestelmän suunnittelu

Tuotetestaus päätettiin toteuttaa paineilmatarraimella. Kun tarrainta valittiin, tuli ottaa huomioon kuuntelunapin pyöreä muoto ja muovinen ulkokuori. Vaihtoehtoina tarraimeksi oli kaksi- tai kolmisorminen tarrain tai imukuppitarrain. Imukuppitarrain hylättiin, koska kuuntelunapin pinta oli epätasainen, jolloin tarvittavaa alipainetta ei välttämättä saataisi aikaiseksi, ja lisäksi imukupin runko olisi pitänyt suunnitella ja rakentaa itse. Kolmisormiset tarraimet olivat liian kalliita ja isoja projektiin, joten päädyttiin kaksisormiseen ja kaksitoimiseen tarraimeen. Tarraimen valinnan jälkeen suunniteltiin leuat ja kiinnike robottiin. Ensin ajateltiin, että leuat ja kiinnike teetettäisiin alumiinista, joka olisi kevyt ja vahva materiaali. Päädyimme kuitenkin tekemään osat muovista, koska se on materiaalina edullinen ja nopeasti ja helposti muokattava. Syynä muovin käytölle oli myös se, että kuuntelunappi on kevyt ja sen nostamiseen ei tarvita suuria voimia.

Kun tarrain oli saatu asennettua, manipulaattorille piti kalibroida uusi työkalupiste. Työkalupiste tulee määrittää jokaiselle työkalulle erikseen ja tallentaa robotin muistiin, jotta robottiin voidaan helposti ja nopeasti vaihtaa eri työkaluja ilman tuotannon pitkää keskeytystä. Mikäli työkalupisteen koordinaatit tiedetään, ne voidaan syöttää robottiin, kun sille luodaan uutta tiedostoa. Uutta tiedostoa luotaessa tarvitaan työkalupisteen koordinaattien lisäksi myös työkalun massa. Mikäli työkalupistettä ei tiedetä, se täytyy määrittellä. Työkalupiste määritetään kiinteän kalibrointipisteen avulla, jonka mikään koordinaatti ei ole nolla. Tähän tarkoitukseen sopii, esimerkiksi päästään teroitettu tanko, joka on pystyssä robotin työalueella. Koordinaatit saadaan lähestymällä tätä pistettä neljästä eri suunnasta niin, että haluttu työkalupiste koskettaa sauvan päätä mahdollisimman tarkasti. Tässä työssä tarraimen työkalupiste määritettiin virittämällä kuminauha leukojen välille ja merkitsemällä sen keskikohta. Tällä tavalla saatiin hyvä toistuvuus ja varmistettiin, että sama piste koskettaa kalibrointipistettä jokaisesta suunnasta. Tämän jälkeen saadut koordinaatit voidaan tallentaa erilliseen tiedostoon, jos ne häviävät robotin muistista. Robotille voidaan määrittää myös kiinteän työkalun työkalupiste, joka voi olla esimerkiksi lattialla oleva työstökone. (5, lk.8 s.12-21)

Tampereen teknillisellä yliopistolla on oma paineilmaverkko, joten erillistä kompressoria tai muuta paineilmalähdettä ei tarvittu. Paineilmaverkon paine oli noin 10 bar, joten

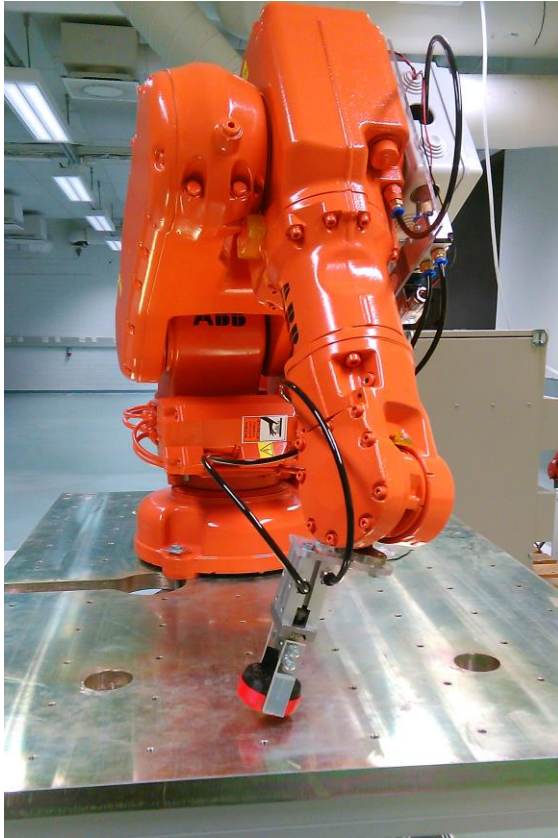
järjestelmään tarvittiin paineenalennusventtiili, jolla paine saatiin tarraimen painealueeseen (2-7 bar) sopivaksi. Painetta muuttamalla voitiin suoraan säätää myös tarraimen etäkuuntelulaitteeseen kohdistama puristusvoima. Paineenalennusventtiiliin lisäksi tilattiin 24V 5/2-suuntaventtiili, jolla voidaan avata ja sulkea tarrain. Venttiili kytkettiin robotin I/O väylään, jolloin sitä voidaan ohjata käsiohjaimella, sekä lisätä ohjelmaan. Tämän lisäksi järjestelmään tilattiin sopivat letkut ja liittimet.

4.1.3 Ohjelmakoodi

Etäkuuntelulaitteen kiinnityksessä käytettävä magneetti aiheuttaa kiinnitettäessä suuria kiihtyvyyksiä, jotka toistuvassa irrotuksessa ja kiinnityksessä saattavat irrottaa komponentteja. Erityisesti tarkoitus oli testata pietsoanturin liimausta. Testissä haluttiin simuloida mahdollisimman hyvin ihmisen kuuntelunappiin kohdistamaa käsittelyä, eli säädettiin pudotuskorkeus, -kulma ja puristusvoima sellaisiksi, joiden ajateltiin niiden loppukäytössä olevan. Kokemuksen perusteella arvioitiin, että käytössä etäkuuntelulaitteen voimakkaat magneetit vetävät laitteen käyttäjän sormista, kun se viedään tarpeeksi lähelle kiinnitettävää pintaa. Tuotteen testauksen kannalta tämä tarkoittaa käytännössä sitä, että laite pudotettaisiin noin 1-2 cm korkeudelta kulma edellä. Robotti ohjelmoitiin niin, että se pudottaa kuuntelunapin neljälle eri kulmalle ohjelmakierron aikana. Haasteeksi muodostui etäkuuntelulaitteen paikan muuttuminen pudotuksen jälkeen. Koska etäkuuntelulaite pudotettiin kulma edellä, magneetti liikutti sitä aina jonkin verran. Ohjelmakierron aikana pieni liike kertautui ja etäkuuntelulaite karkasi niin paljon, että manipulaattori törmäsi sitä päin. Opinnäytetyön aikataulun puitteissa ei ollut aikaa tehdä sensorointia, joka keskittäisi leuat automaattisesti. Ohjelmallisesti kokeiltiin muutamaa erilaista vaihtoehtoa, jolla saataisiin pidettyä etäkuuntelulaite paikallaan niin, että ohjelmaa voitaisiin toistaa useita kertoja. Lopulta päätettiin, että etäkuuntelulaite keskitettäisiin jokaisen pudotuksen jälkeen, kääntämällä tarraimen leuat 90°. Tällöin leuat kiinnittyivät jokaisella kerralla eri suunnasta keskittäen etäkuuntelulaitteen ja ohjelmaa voidaan toistaa useita kertoja niin, että etäkuuntelulaite ei karkaa liikaa. Ohjelmakoodi näkyy opinnäytetyön liitteenä (liite1).

Kuvassa 16 tuotetestausohjelma on käynnissä ja manipulaattori on asennossa, josta pudotus tapahtuu. Siinä näkyy kulma ja etäisyys pöydästä, jolla irrotus todennäköisimmin tapahtuu. Irrotuksen kulma ja etäisyys ovat samoja myös muissa irrotuskohdissa, joita

on neljä kappaletta. Paineilmajärjestelmän ollessa kiinnitettynä manipulaattoriin, täytyi ottaa huomioon paineilmaletkut ja pitää manipulaattorin liikkeitä tarpeeksi pieninä, jotta letkut eivät kiertyisi manipulaattorin ympäri ja rikkoutuisi.



KUVA 16. Etäkuuntelulaitteen testaus

Liitteenä 1 oleva ohjelmakoodi rakentuu alussa olevista paikkatiedoista, jotka on tallennettu järjestelmään vakioiksi (CONST robtarget). Niin kuin aiemminkin tämän työn ohjelmointi-kappaleessa on kuvailtu, paikkatieto sisältää pisteen koordinaatit, kulmarvot ja moottorivakiot. Tähän ohjelmaan paikkatiedoiksi on tallennettu alkupiste, joka toimii ensimmäisenä tartuntapisteenä. Muut paikkatiedot sisältävät, jokaisen tartunta- ja pudotuspisteen. Tämän jälkeen alkaa varsinainen ohjelma, jonka ensimmäisenä käskynä muodostetaan silmukka. ”FOR”-käsky muodostaa alla olevasta ohjelmasta muuttujan ”i”. ”FROM 1 TO 20 DO”-komennolla edellä muodostettu muuttuja toistetaan 20 kertaa, jolloin saadaan silmukka toteutettua. Lukua muuttamalla voidaan säätää testin kestoa halutulla tavalla. Silmukka päätetään komennolla ”ENDFOR”. Ensimmäinen liikekäsky ajaa manipulaattorin kotipisteestä lähelle ensimmäistä tartuntapistettä. Tämän jälkeen ajetaan ensimmäiseen tartuntapisteeseen. Tartuntapistessä komennolla ”set gripperi” asetetaan lähtö arvoon yksi, jolloin tarttujan leuat kiinnittyvät. Tartunnan jälkeen etäkuuntelulaite nostetaan kulmaan, jonka jälkeen pudotus tapahtuu komennolla

”reset gripperi”. Komento asettaa lähdön arvoksi nolla ja avaa tarttujan leuat. Etäkuuntelulaite pudotetaan neljästä eri kulmasta, niin kuin aiemmin tässä työssä on jo kerrottu. Digitaalinen lähtö numero 13 on varattu etäkuuntelulaitteen toiminnan testaamiseen jokaisen pudotuksen jälkeen. Lähtöön liitetään laite, joka lähettää signaalia pöytää pitkin etäkuuntelulaitteeseen sekunnin ajan. Signaali luetaan etäkuuntelulaitteella ja näin saadaan tieto, kuinka monta pudotusta laite kestää. Signaalin sekunnin lähetysaika määritetään komennolla ”WaitTime 1”. Tällä komennolla saadaan ohjelman toteutus odottamaan ennen seuraavaa komentoa, mikä asettaa signaalilaitteen lähdön nollassi.

5 POHDINTA

Projekti oli jo lähtökohtaisesti haastava, sillä kellään yrityksen sisällä ei ollut aikaisempaa kokemusta robottien kanssa työskentelystä. Tämän vuoksi kaikki jouduttiin opettelemaan alusta asti ja tekemään paljon niin sanottua ”turhaa työtä”, kun selvitettiin järjestelmän perustoimintoja ja periaatteita. Onneksi yrityksessä oli paljon tietotaitoa ja yhteistyö Tampereen teknillisen yliopiston kanssa sujui hienosti, mikä mahdollisti työn valmistumisen. Yhteistyö myös ABB:n suuntaan oli sujuvaa ja oli tunne, että kaikilla oli todellinen halu auttaa. Projektin alkaessa tavoitteet eivät olleet täysin selvät, koska ei ollut tietoa edes toimiiko robotti ja kuinka suuri työmäärä tarvitaan robotin käyttökuntoon saamiseksi. Projektin tavoite muodostui sen jälkeen, kun robotti oli saatu käyttökuntoon. Yrityksen kehittämälle etäkuuntelulaitteelle oli suunniteltu erilaisia testejä, mutta sille ei ollut vielä sopivaa testijärjestelmää mekaaniseen testaukseen. Syntyi ajatus, että robotti voitaisiin ohjelmoida etäkuuntelulaitteen testaukseen sopivilla työkaluilla. Tämän jälkeen tavoitteeksi muodostui robotin varustaminen tuotteen mekaaniseen testaukseen.

Suurimmaksi haasteeksi projektin edetessä muodostui robotin ikä, mikä teki työstä oletettua haastavamman. Järjestelmän tekniikka oli vanhentunut ja etenkin ohjainyksikkö aiheutti ongelmia, koska se ei ollut yhteensopiva modernien ohjelmistojen kanssa. Tämän takia alkuperäisiä suunnitelmia jouduttiin muuttamaan. Aluksi suunniteltiin, että robottisovellus voitaisiin suunnitella, mallintaa ja ohjelmoida kokonaan Robotstudiolla. Tämä olisi tehnyt suunnittelusta sujuvampaa, eikä vaihtoehtoisia ohjelmistoja olisi tarvinnut etsiä ja ladata. Myös muita aiemmin koulussa opittuja tietoja tästä robotista jouduttiin soveltamaan vanhan järjestelmän takia. Esimerkiksi vanhan ohjainyksikön käyttöliittymä ja manipulaattorin ohjaaminen käsiohjaimen kautta oli erilaista kuin uusissa versioissa. Isoin pettymys projektissa oli se, että robotin käyttöjärjestelmää ei saatu päivitettyä, eikä omaa robottijärjestelmää saatu ladattua kovalevyllä olevan tilan puutteen vuoksi. Järjestelmäpäivitys ja kovalevyn tyhjennys oltaisiin voitu tehdä ABB:n toimipisteellä, mutta siihen ei lähdetty aikataulun vuoksi. Projektin jälkeen pidettiin kuitenkin yhtenä vaihtoehtona lähettää robotin tietokone ABB:lle päivitettäväksi. Muut haasteet pystyttiin selättämään tutkimalla vastaavia projekteja ja käyttämällä hyväksi yliopistolta saatua apua. Samanlaisia tilanteita ja ongelmakohtia oli ollut muillakin, joten ratkaisuja löytyi hyvin.

Projekti onnistui vastoinikäymisistä huolimatta kokonaisuutena hyvin ja tavoitteeseen päästiin aikataulun puitteissa. Kun robotti oli saatu käyttökuntoon, sen laitteiston kanssa ei syntynyt suurempia ongelmia ja asennustyöt sujuivat hyvin. Ainoa asia joka hidasti testausjärjestelmän valmistumista, oli tilattujen komponenttien saapumisen odottaminen. Paineilmajärjestelmä saatiin toimimaan suunnitellusti ja se pystyttiin valjastamaan tuotetestauksen tarpeisiin. Nyt etäkuuntelulaitteen mekaanisia ominaisuuksia voidaan mitata kestopesteillä, kun ohjelmallisesti voidaan määrätä pudotusten määrä. Vuosien jatkuvaa käyttöä voidaan simuloida nopeasti tekemällä esimerkiksi tuhat nostoa ja pudotusta. Samalla voidaan seurata milloin pietsoanturi ja muut komponentit alkavat irtotaamaan ja signaali heikentyä. Pietsoanturin toimintaa seurataan jokaisen pudotuksen jälkeen signaalilla, joka lähetetään pöytää pitkin etäkuuntelulaitteeseen. Etäkuuntelulaitteen käytön kannalta on tärkeää tietää, kuinka monta kertaa laite voidaan kiinnittää ja irrottaa. Kun saadaan tieto kuinka paljon rasiitusta komponentit kestävät, niiden kiinnitystä voidaan kehittää paremmaksi ja tarvittaessa vaihtaa komponentteja paremmin kestäviin. Näin varmistetaan, että tuote kestää pitkäaikaisen käytön teollisuusympäristössä.

Jatkossa, kun robotti on käyttökunnossa, sille voidaan suhteellisen vähällä työllä tehdä uusia sovelluksia esimerkiksi tuotetestaukseen. Etenkin paineilmaan pohjautuvat sovellukset ovat jatkossa helppoja toteuttaa. Muita tuotetestauksen kannalta hyödyllisiä sovelluksia, joita mietittiin, voisivat olla esimerkiksi etäkuuntelulaitteen käyttöliittymän painikkeiden testaaminen. Tällaisissa laitteissa painikkeet on todettu usein heikoiksi kohdiksi. Kovassa käytössä teollisuusympäristössä niiden väliin voi kerääntyä likaa, ne voivat irrota ja lakata toimimasta. Myös painikkeiden mekaanista testausta mietittiin tämän opinnäytetyön tavoitteeksi. Todettiin kuitenkin, että painikkeet eivät ole niin oleellisia laitteen toiminnan kannalta, joten siihen ei päädytty. Nappien testaaminen voitaisiin toteuttaa samalla paineilmajärjestelmällä esimerkiksi paineilmasylinterillä tai sähköisesti solenoidilla. Robotin manipulaattorin avulla nappeja voitaisiin painella eri kulmista, jolloin se simuloisi hyvin oikeaa käyttöä.

Nykyisellä järjestelmällä ja paineilmatarraimella voitaisiin toteuttaa myös jonkinlainen demonstraatio esimerkiksi messuille, jossa esitellään etäkuuntelulaitteen toimintaa ja yrityksen monipuolista osaamista. Robotilla etäkuuntelulaite voitaisiin asettaa erilaisille pinnoille, joissa on erilaisia äänilähteitä. Samalla tulisi esille yrityksen monipuolinen

osaaminen tekniikassa. Muita sovelluksia joita voitaisiin käyttää esimerkiksi tuotannossa, ovat maalaus tai pinnoitus ja paikoitus tuotantolinjalla.

LÄHTEET

1. <http://www.meluta.fi/>
2. Product Specification S4Cplus 3HAC 9039-1/ M2000 BaseWare OS 4.0
3. Product Specification IRB 140 3HAC 9041-1 / M2000
4. Product Specification RobotWare Options 3HAC 9218-1 for BaseWare OS 4.0
5. BaseWare User's guide 3HAC 7793-1 Revision F RobotWare-OS 4.0
6. Product Manual Controller 3HAC 7564-1 / M2000
7. RAPID reference part 1
8. http://pds.mobil.com/Finland-Finnish/Lubes/PDS/GLXXFIINDMOMobilgear_600_XP.aspx luettu 2.9.2016

LIITTEET

Liite 1. Tuotetestauksessa käytetty ohjelmakoodi.

```

MODULE M3

  CONST robtarget klm4:=[[401.55,361.57,-
16.56],[0.196933,0.016455,0.980278,-0.001401],[0,-
1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget grip3:=[[401.63,361.58,-
31.96],[0.011347,0.016313,0.999801,0.001614],[0,0,0,0],[9E+09,9E
+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget klm3:=[[397.88,358.03,-
17.57],[0.096513,0.706087,-0.693344,0.106775],[0,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget grip2:=[[397.89,358.05,-32.36],[0.008529,-
0.714155,0.699936,0.000285],[0,0,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget klm2:=[[397.9,358.05,-14.89],[0.205468,-
0.021264,-0.978369,-
0.011211],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget grip1:=[[397.9,358.02,-
32.79],[0.000892,0.023167,0.99971,0.006493],[0,0,0,0],[9E+09,9E+
09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget klm1:=[[400.03,357.64,-15.79],[0.132116,-
0.697474,0.691881,0.131819],[0,0,-
2,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  CONST robtarget alku:=[[396.62,357.67,-
30.62],[0.000845,0.709761,-0.704439,0.002297],[0,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

  PROC main()
    FOR i FROM 1 TO 20 DO
      MoveJ [[394.68,357.61,17.16],[0.000943,0.713414,-
0.700738,0.002359],[0,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]],v300,fine,tarrain1\W
Obj:=poytal;
      MoveL alku,v100,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;
      Set gripperi;
      MoveL klm1,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;

```

```
Reset gripperi;  
Set do13;  
WaitTime 1;  
Reset do13;  
MoveL grip1,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Set gripperi;  
MoveL klm2,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Reset gripperi;  
Set do13;  
WaitTime 1;  
Reset do13;  
MoveL grip2,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Set gripperi;  
MoveL klm3,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Reset gripperi;  
Set do13;  
WaitTime 1;  
Reset do13;  
MoveL grip3,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Set gripperi;  
MoveL klm4,v50,fine\Inpos:=inpos50,tarrain1\WObj:=poytal;  
Reset gripperi;  
Set do13;  
WaitTime 1;  
Reset do13;  
ENDFOR  
ENDPROC  
ENDMODULE
```