

Tomi Haapalainen

ABB ROBOTIN KÄYTTÖÖNOTTOASENNUS JA OHJELMOINTI
KONENÄKÖSOVELLUKSESSA

Tekniikka ja merenkulku, Pori
Automaatiotekniikan suuntautumisvaihtoehto
2011

ABB ROBOTIN KÄYTTÖÖNOTTOASENNUS JA OHJELMOINTI

Haapalainen, Tomi
Satakunnan ammattikorkeakoulu
Automaatiotekniikan koulutusohjelma
Syyskuu 2011
Ohjaaja: Auramo, Yrjö, Leino, Mirka
Sivumäärä: 89
Liitteitä: 15

Asiasanat: robotti, asennus, ohjelmointi, konenäkö

Tämän opinnäytetyön aiheena oli ABB robotin käyttöönottoasennus ja ohjelmointi konenäkösovelluksessa. Opinnäytetyö tehtiin konenäköjärjestelmän demonstraatioon perustuvan projektin pohjalta. Projektissa asennettiin yleisrobotti, sen robottiohjain järjestelmä ja robotti ohjelmoitiin se toimimaan yhdessä konenäköjärjestelmän kanssa.

Tämä opinnäytetyö käsittelee yleiseltä osalta automaation ja robotiikan historiaa, kehitystä sekä yleisiä koneturvallisuuteen liittyviä asioita. Projektin liittyen opinnäytetyössä käydään läpi robotin asennuksen ja käyttöönoton kannalta tärkeimmät asiat sekä robotin ohjelmoinnin periaatteita ja ohjelman rakennetta.

Projektin case-yrityksen toimi Puustelli Oy, joka on lähialueella keittiökalusteita valmistava yritys. Projektin tarkoituksena oli demonstroida robotin ja konenäkösovelluksen käyttöä levyjen käsittelyprosessissa.

ASSEMBLING AND PROGRAMMING OF ABB ROBOT

Haapalainen, Tomi

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in Automation Technics

September 2011

Supervisor: Auramo, Yrjö, Leino, Mirka

Number of pages: 89

Appendices: 15

Keywords: robot, assembly, programming, machine vision

The purpose of this thesis was assembling and programming ABB robot with machine vision application. This thesis was done on basis of machine vision demonstration project. During the project, general robot was assembled and it was programmed to function with machine vision application.

This thesis describes generally history and development of automation and robotics and also basic principles of safety of machinery. On project point of view, this thesis involves assembling of robot and principles and structures of programming robot.

Case-company for the project was Puustelli Oy, which manufactures kitchen furnitures on near-by area. The purpose of the project was to demonstrate using of robot and machine vision application in plate handling process.

SISÄLLYS

1	JOHDANTO.....	6
2	ROBOTIIKKA.....	9
2.1	Historia ja kehitys.....	9
2.1.1	Alkuaika.....	9
2.2	Robotiikan kehitys.....	11
2.2.1	Nyky aika.....	14
2.2.2	Tulevaisuus.....	17
2.3	Robottimallit.....	20
2.3.1	Portaalirobotti.....	20
2.3.2	Sylinterirobotti.....	21
2.3.3	Napakoordinaatistorobotti.....	22
2.3.4	SCARA-robotti.....	22
2.3.5	Kiertyvänivelinen robotti.....	23
2.3.6	Rinnakkaisrakenteinen robotti.....	24
3	ABB ROBOTTI.....	26
3.1	Järjestelmän yleiskuvaus.....	26
3.2	Robottijärjestelmä.....	30
3.3	Manipulaattori.....	32
3.4	S4C Robottiohjain.....	32
3.5	Operoitin paneeli.....	33
3.6	Robotin liikealueet.....	34
3.7	Robotin kuormitus.....	36
3.8	Robotin kalibrointi.....	37
4	TARTTUJA.....	38
4.1	Robottitarraimet ja työkalut.....	38
4.1.1	Imutartunta.....	39
4.1.2	Magneettitarrain.....	39
4.1.3	Vakiotarrain.....	40
4.2	Tarraimen suunnittelu.....	40
4.3	Projektin tarttuja.....	41
5	KOMMUNIKOINTI.....	44
5.1	Tiedonsiirtoliitännät.....	44
5.2	Projektin tiedonsiirto.....	46
6	OHJELMOINTI.....	48
6.1	Robottien ohjelmointi.....	48

6.1.1	Johdattamalla ohjelmointi	48
6.1.2	Opettamalla ohjelmointi	49
6.1.3	Etäohjelmointi	49
6.2	Ohjelman rakenne	50
6.3	Ohjelmakoodi.....	53
7	ASENNUS JA HUOLTO.....	56
7.1	Huolto ja kunnossapito	56
7.2	Kalibrointi.....	57
7.3	Järjestelmäasennus.....	66
7.4	Huoltotoimenpiteet	69
8	TURVALLISUUS.....	73
8.1	Standardit	73
8.2	Lait ja direktiivit	73
8.3	Turvatoiminnot	75
8.4	Ohjausjärjestelmä.....	76
8.4.1	Ohjausjärjestelmän rakenne	77
8.4.2	Ohjauksen tiedonsiirto.....	79
8.5	Robotin käynnistyminen ja pysähtyminen.....	80
8.5.1	Käynnistysvaroitusta	81
8.5.2	Käynnistys.....	82
8.5.3	Pysäytys.....	82
8.6	Eristäminen aidalla	83
8.7	Robottisolu.....	83
8.8	Pneumatiikka	84
8.9	Asentaminen ja käyttöönotto	85
9	JATKOKEHITYS	87
	LÄHTEET.....	88
	LIITTEET	

1 JOHDANTO

Robottien historia kulkee yhdessä ihmiskunnan teollistumisen kanssa. Kirjallisuudessa robotit olivat vuoteen 1950 asti ihmisiä tuhoavia hirviöitä, sitten Isaac Asimov esitteli uuden robottityypin. Sen sisälle oli rakennettu käskykanta, joka pohjautui ihmisen olemassaolon suojaamiseksi tehtyihin robotiikan kolmeen lakiin.

1. Robotti ei saa vahingoittaa ihmistä eikä laiminlyönnin johdosta saattaa tätä vahingoittumaan.
2. Robotin on toteltava ihmisen sille antamia määräyksiä paitsi milloin ne ovat ristiriidassa ensimmäisen pääsäännön kanssa.
3. Robotin on varjeltava omaa olemassaoloaan niin kauan kuin se ei ole ristiriidassa ensimmäisen eikä toisen pääsäännön kanssa.

Robottien varsinainen kehitys tapahtui kuitenkin teollisuudessa, eikä kirjallisuudessa vaikka tieteiskirjallisuuden antamaa insinööreille antamaa inspiraatiota ei voida väheksyä. Erityisesti 1980-luvulla robottien määrä kasvoi valtavasti, samalla alkoi varsinainen robotiikan tutkimus ja kehitystyö. Ajatuksena oli joustavuuden lisääminen vapauttamalla robotit kiinteistä paikoista ja ennalta määrättyistä reiteistä. Ajatus oli tarkalleen oikea ja siihen pohjautuukin tämän päivän robotiikka./6, s. 158-160/

Roboteille sopivia työkohteita ovat sellaiset, joissa tarvitaan nopeutta, ehdotonta tarkkuutta, ihmiselle vaikeat, raskaat ja vaaralliset työt. Tällöin robottijärjestelmä on ihmistä tarkempi ja pitkäkestoisempi. Teollisuurobottien soveltaminen lääketieteessä ja terveystaloudessa on lisääntynyt. Robotti pystyy ihmistä tarkempaan ja nopeampaan työskentelyyn esimerkiksi lonkkaleikkauksen yhteydessä, jossa luuhun joudutaan työstämään liityntäkolot proteeseja varten./5, s. 112-115/

Ihmisen arkielämässä robotit alkavat yleistyä, ei pelkästään työpaikoilla vaan myös kotona. Erilaiset kotitalousrobotit ovat alkamassa kasvattamaan markknoitaan. Imuri- ja nurmikoneleikkurobotit ovat jo nyt saatavilla. Pienen tehonsa vuoksi ne joutuvat

lataamaan itseään usein, mutta se korvaantuu jatkuvalla toimimisella. Robotti imuroi tai leikkaa nurmikkoa jatkuvasti./5, s. 117/

Robottien yleinen käyttö on kuitenkin vielä pääasiallisesti vahvasti teollisuudessa, jossa ne soveltuvat hyvin kappaletavaran käsittelyyn liittyvään teollisuuteen ja ovat siksi käytettyjä siellä erilaisissa automaation sovelluksissa tuotantolinjan osana. Robotteja käytetään tuotannon apuna erityisesti metalli-, sähkö- ja elektroniikkateollisuudessa, mutta myös vaatetus- ja puusepänteollisuudessa, jota case-yrityksemme edustaa. Kappaletavaratuonnossa automatisointi on usein viety pitkälle ja ohjattujen työstökoneiden, teollisuusrobottien ja automaattisten kuljettimien laaja käyttö on mahdollistanut tehokkaan ja joustavan tuotannon. Tämän tuloksena voidaan valmistaa tehokkaasti ja pienillä kustannuksilla korkealaatuisia ja asiakkaan vaatimusten mukaisesti räätälöityjä tuotteita./5, s. 102-103/

Case

Tähän opinnäytetyöhön liittyvän projektin tavoitteena oli tuottaa case-yrityksenä toimineelle Puustelli Oy:lle uusinta teknologiaa hyödyntäen levyntyöstökoneita palveleva älykamera- ja robottijärjestelmä. Järjestelmä syöttää levyjä työstökoneelle ja siirtää ne edelleen odottamaan jatkokäsittelyä mahdollisimman nopeasti ja kustannustehokkaasti. Nykyinen työstökone on toiminnoiltaan vanhentunut ja läpimenoajaltaan hitaampi kuin mitä yritys tuotannossaan tarvitsee. Järjestelmän uudistaminen on yritykselle ajankohtainen ja tällä projektilla haettiin tietoja ja mahdollisuuksia uutta järjestelmää varten. Case-yrityksen tavoitteena on ajanmukaistaa nykyisin käytössään oleva levyntyöstömenetelmä. Projektiin osallistuneen ryhmän tiedossa ei ole yrityksen mahdollisia projektille asetettuja liiketoiminnallisia tavoitteita.

Puustelli Oy

Puustelli Group Oy on osa 90-vuotiasta Harjavalta-konsernia, johon kuuluu myös Kastelli-talot. Yritys valmistaa keittiökalusteita Harjavallassa sijaitsevilla tuotantolaitoksillaan. Konsernin liikevaihto oli vuonna 2010 noin 151,3 miljoonaa euroa ja se työllistää verkostoineen Suomessa noin 1400 henkilöä.

Puustelli-keittiöt ja Kastelli-talot ovat jo vuosia olleet alansa markkinajohtajia Suomessa. Puustelli Group Oy on vahvasti kansainvälistyvä organisaatio. Tällä hetkellä Puustelli-keittiöitä viedään Ruotsiin, Venäjälle, Baltian maihin sekä Englantiin.

Paikalliset maanviljelijät perustivat Harjavalta Oy:n 1920, jolloin yhtiö harjoitti sahaustoimintaa. Yhtiö laajensi toimintaansa rakennuspuusepäntehtäksi 1944. Kolme vuotta myöhemmin yhtiö siirtyi uusille omistajille, jotka nostivat yhtiön sotakorvausten myötä nopeaan kasvuun.

Puustelli-keittiöiden tuotanto alkoi vuonna 1977 ja vuonna 1984 se oli jo Suomen ostetuin keittiö. 1990-luvun alussa markkinat laajenivat Ruotsiin, Viroon ja Venäjälle. Lisäksi yhtiö alkoi valmistaa toimistokalusteita. Tällä hetkellä Puustellilla on Suomessa 45 myymälää ja Ruotsissa 8, lisäksi Pietarissa 6 ja Moskovassa 3. Muita Puustellin myymäläkaupunkeja ovat Tallinna ja Riika./22/

Puustelli Group Oy

-keittiöt, kylpyhuoneet ja kalusteet

-noin 200 000 keittiökalustetta/vuosi

Kastelli

-talo- ja huvilapaketit

-yli 1200 talo- ja huvilatoimitusta/vuosi

2 ROBOTIIKKA

2.1 Historia ja kehitys

2.1.1 Alkuaika

Automaatio

Automaation historia voidaan katsoa alkaneen jo Leonardo da Vincin (1452-1519) ja James Wattin (1736-1819) tärkeistä havainnoista ja sovelluksista. Kuitenkin automaatio käsitteenä liitettiin vasta 1900-luvun alkupuolella teollisuuden prosessien tai vastaavien toimintojen suorittamiseen ihmistyövoimaa korvaavana. Nykyaikaisen automaation voidaan katsoa kuitenkin syntyneen 1940-luvulla suoritetun tutkimus- ja kehitystyön seurauksena. Tämän jälkeen automaation eri sovellukset laajenivat nopeasti jokapäiväiseen elämään./5, s. 13/

Yhdysvalloissa 1920-luvulla Henry Ford kehitti henkilöautojen massatuotannon. Tuotantoa varten Ford kehitti tuotantolinjan, joka perustui liukuhihnaan. Valmistettavat autot kuljetettiin jatkuvana virtanan työpisteestä toiseen, joissa työ tehtiin mahdollisimman nopeasti./5, s. 100-101/



Kuva 2.1 Nykyaika/21/

Fordin tuotannossa ei ollut mukana varsinaisesti automaatiota, sillä tuotantolinjan tai liukuhihnan nopeutta ja toimintaa ohjattiin käsin. Lisäksi kaikki työvaiheet tapahtuivat käsityönä. Fordin aikaiselle massatuotannolle keskeisin asia olikin mekanisaatio, varsinaisen automaation esiaste./5, s. 100-101/

Mekanisaatioonkin perusedellytys automaatiolle, mutta vasta kun koneiden ja laitteiden toiminnan ohjaus tapahtuu ilman ihmisen jatkuvaa ohjausta, voidaan puhua automaatiosta./5, s. 101/

Fordin aikaisessa massatuotannossa työ oli usein erittäin yksitoikkoista ja rasittavaa. Liukuhihnan ääressä toistettiin samaa yksittäistä työvaihetta uudestaan ja uudestaan. Liukuhihnan tahdissa pysyminen oli hyvin rasittavaa ja vaikeaa./5, s. 103/

Nykyisin ohjatut koneet ja robotit suorittavat lähes kaiken yksitoikkoisen ja rasittavan työn. Ihmisen työnkuva on muuttunut tuotantolinjan valvonnaksi ja erilaisten häiriö- ja vikatilanteiden poistamiseksi. Monien kappalevaratuotannossa työskentelevien työnkuva on muuttunut monipuoliseksi ja koulutusta vaativaksi verrattuna alkuaukojen massatuotantoon. Se on tehnyt työstä mielenkiintoisen ja motivoivan. Tämä on työntekijän kannalta positiivinen asia.

Automaatio on parantanut työn laatua ja sisältöä työntekijöiden osalta, mutta samalla se on myös vähentänyt rajusti työvoiman tarvetta tuotannossa. Tästä syystä automaatiota on syytetty työttömyyden lisäämisestä, vaikka syynä saattaa kuitenkin olla sosiaalisen ja taloudellisen järjestelmän sopeutumis- ja kehityskyvyn puutteellisuus. Kaikesta huolimatta automaatio jatkaa kasvuaan eri elämän alueilla, niin tuotannossa kuin kotonakin./5, s. 103-104/

Robottiikka

Robotti-sana on muotoutunut tšekinkielisestä sana "robota", mikä tarkoittaa maaorjaa. Robotti-sanana otettiin käyttöön noin 1920 Josef Èapek veljensä Karlin näytelmään "Rossum's Universal Robots", joka kertoi keinotekoisista työntekijöistä. Vuonna 1938 suunniteltiin ensimmäinen ruiskumaalausmekanismi ja ensimmäinen teollisuusrobotti suunniteltiin vuonna 1954. Teollisuuteen toimitettiin ensimmäinen robotti vuonna 1962, ja se meni autoteollisuuteen valukoneiden käsittelyyn./5, 6/

Aluksi tarkoituksena oli käyttää yhtä teollisuusrobottityyppiä kaikkii eri tehtäviin, mutta pian kuitenkin huomattiin, että ideaalisempaa on käyttää eri robottimalleja eri tehtäviin. Valmistajien halu pystyä toimittamaan robotti kaikkiin teollisuuden tehtäviin, lisäsi teollisuusrobottien määrää. Vuoden 2001 lopussa arvioitiin teollisuudessa käytettäviä robotteja olevan 757 000 kappaletta sadoista eri malleista. Nykyisin robotit ovat koneita koneiden joukossa./5, s. 109/

Ensimmäinen kuvankäsittelyllä varustettu liikkuva robotti tehtiin vuonna 1968. Siihen kiinnitetty tietokone oli kuitenkin huoneen kokoinen, joten sen käyttömahdollisuudet olivat lähinnä kokeiluluontoiset. 1980-luvulta lähtien on työkoneita automatisoitu toimimaan itsenäisesti ilman ihmisen suoraa ohjausta./5, s. 109-110/

2.2 Robottiikan kehitys

Robottien määrä kasvoi erityisesti 1980-luvulla valtavasti. Jokainen tehdas, jossa ei ollut robotteja tekemässä tarkasti määriteltyjä töitä, ei voinut pitää itseään modernina

tuotantolaitoksena. Tehdassaleihin ilmestyi nopeassa tahdissa vihivaunuja, jotka maahan upotetun johdinverkoston avulla kuljettivat tavaroita työpisteiden välillä ja varastoon. Useisiin laitoksiin rakennettiin vihivaunujärjestelmä osaksi automaatiojärjestelmiä (FMS) ilman tarkempaa ajatusta. Kehityksen kärjessä olemisesta muodostui itsetarkoitus, eikä normaalia yrityksen tuotannon kasvattamista voiton tavoittelussa otettu huomioon. Taloustilanteen kiristyessä huomio siirtyi takaisin liikkuvien robottien tutkimiseen. Tutkimustyön tausta-ajatuksena oli joustavuuden lisääminen vapauttamalla robotit niiden kiinteistä paikoista ja ennalta määrättyistä reiteistä. Ajatus olikin kehityksen kannalta tarkalleen oikea ja siihen pohjautuukin tämän päivän robotiikka./6, s. 158-160/

Vihivaunut ovat tietokoneiden ohjaamia automaattisia itsestään liikkuvia ajoneuvoja. Niitä on ollut käytössä tehtaissa ja varastoissa jo 1970-luvulta lähtien. Ne liikkuvat valmiiksi merkittyä reittiä pitkin, induktiivisen kaapelin tai maalin avulla, tai itsenäisesti opetettuihin pisteisiin käyttäen avuksi erilaisia antureita ja kamerajärjestelmää. Nykyisin liikkuvien robottien soveltamista on alettu käyttämään enemmän tehdasympäristön ulkopuolella. Niitä käytetään erilaisiin tehtäviin muun muassa:

- kuljetuksiin
- rakentamiseen ja purkamiseen
- palontorjuntaan
- pelastustehtäviin
- kaivoksissa ja satamissa
- maa- ja metsätalouden tehtäviin
- vartiointiin
- opastukseen
- siivoukseen
- satelliittien huoltotöihin ja avaruuden tutkimiseen.

Roboteista puhuttaessa yleensä huomio keskittyy robotin käsivarteeseen, joka kuitenkin on vain mekaaninen osa. Robotin ohjausjärjestelmä on vaikeammin toteutettavissa, se vastaa ohjelmistoinen robotin varsinaisesta työstä. Siitä syystä ohjausjärjestelmä vastaa suurimmasta osasta robotin kehityksestä, ja se on samalla myös kallein osa teollisuusrobotissa./5, s. 112-114/

Teollisuusrobotisovellusten tekemistä on pyritty helpottamaan ja tuotteistamaan mahdollisimman paljon. Sekä robotin mekaanisissa osissa että ohjausjärjestelmässä pyritään käyttämään valmiita moduuleita, joista voidaan rakentaa asiakkaalle sopiva kokonaisuus. Aikaisemmin tehtyjä sovelluksia pyritään kopioimaan uuteen järjestelmään mahdollisuuksien mukaan. Tätä tehdään jo siinä määrin, että esimerkiksi autonvalmistajat tekevät tarjouspyyntöjä vain muutamille robotisovelluksille ohjelmistotaloille.

Robottien ja niiden sovellusten käyttäminen lääketieteessä ja terveyspalveluissa on lisääntynyt. Esimerkiksi leikkaussaleissa robotisovelluksia oli vuoden 2001 lopussa käytössä lähes 2000 kappaletta, ja vuonna 2005 käytössä oli jo 8000 robottia. Robotti pystyy ihmistä tarkempaan ja nopeampaan työskentelyyn kuvantamismenetelmissä ja erilaisten leikkausten yhteydessä, joissa tarvitaan luun tarkkaa leikkaamista./5, s. 112-113/

Sellaiset työkohteet, jotka ovat ihmiselle vaikeita, raskaista tai vaarallisia, ovat roboteille sopivia työkohteita. Samoin työt joka vaativat nopeutta ja ehdotonta tarkkuutta. Robottijärjestelmä voi avustaa satamissa konttilukkien kuljettajia ajamisessa. Usein konttirivien väleissä voi olla vain puolisen metriä tilaa ja kuljettajan on vaikea nähdä toisen sivurungon edessä olevalle alueelle. Kaivostunneleissa, jotka halutaan optimoida, voi vapaata tilaa olla vain muutama kymmenen senttimetriä. Tällöin on robottijärjestelmä ihmistä tarkempi ja pitkäkestoisempi./5, s. 115/

Itsenäisesti liikkuvien robottien yleistymistä on hidastanut ehdottoman turvallisuuden vaatimus. Ympäröivän tilan hahmottaminen on koneelle yleisesti vaikeampaa kuin ihmiselle vaikka koneen reaktioajat ovat paljon nopeammat yllättävissä tilanteissa. Isoja itsenäisesti toimivia työkoneita ei turvallisuussyistä johtuen vielä uskalleta päästää alueille, joissa on ihmisiä tai muita odottamattomia asioita. /5, s. 115/

Robotit ovat kuitenkin koneita ja ne yleensä eristetään ihmisten läheisyydestä turva-aidoin ja -portein. Turvaportin auetessa robotti pysähtyy turvalliseen tilaan, jolloin ihminen voi mennä robotin työalueelle. Robotti käynnistyy vasta kun turvaportti on

kiinni ja robotti käynnistetään uudelleen turva-alueen ulkopuolelta. Itsenäisesti liikkuvissa kulkuneuvoissa on myös etäisyysanturit, jotka varmistavat sen, että kulkusuunta on vapaa ja esteen lähestyessä hidastavat nopeutta./5, 8/

Robottien yleistyminen erityisesti palvelualalla ja viihdeteollisuudessa on ollut viime vuosina erittäin nopeaa. Ihmishahmoiset robotit, jotka kävelevät portaissa, ovat robottikoirien kanssa muokanneet ihmisten mielikuvaa roboteista. Robotiikassa onkin viime aikoina tapahtunut todellisia kehitysaskelia ja robotiikan tutkimus on ottanut pitkiä harppauksia eteenpäin. Kehityksen nopeus mahdollistaa jatkuvasti uusia sovelluksia. Jo nyt, ja varsinkin tulevaisuudessa, uuden sukupolven robotit ovat valmiita palvelemaan meitä kaikkia./6, s. 157/

2.2.1 Nykyaika

Tällä hetkellä robotiikan kärkikehityksestä vastaavat erilaiset viihdeteollisuuden sovellukset. AIBO, Sonyn valmistama robottikoira on varustettu suurella määrällä ympäristöä aistivia antureita ja erilaisia liikkeitä suorittavia moottoreita. Uusimpaan tekniikkaan on yhdistetty oppiva ja adaptoituvaa ohjelmistoa. Tämän ansiosta on saatu valmistettua käyttäytymiseltään hyvin aidon oloinen lemmikkirobotti.

Samoille markkinoille AIBO:n kanssa tuli iRobotin ja Hasbron kehittänyt My Real Baby-robotinukke. Kolmas kilpailija markkoinne on myös Sonyn valmistama pieni humanoidirobotti, QRIO. Se on noin 50 cm korkea ja sisältää erittäin kehittyntä tekniikkaa moottoreiden ja vaihteistojen alueella./6, 16/

Toinen viihdeteollisuuden kanssa suuren markkinapotentiaalin omaava sovellus on erilaiset palvelurobotit. Palvelurobotit ovat teollisuuden ulkopuolelle sijoitettuja robotteja, joissa on käsivarsi. Tällaisia palvelurobotteja ovat esimerkiksi maataloilla käyttävät lypsyrobotit ja tankkausrobotit. Ensimmäiset lypsyrobotit toteutettiin Belgiassa 1980-luvulla. Mallina lypsyrobotille oli Australiassa kehitetty lampaiden automaattinen keritsemisjärjestelmä. Järjestelmässä lammasta kerrittäessä sen iho

vaurioitui vähemmän kuin ihmisen tekemänä. Järjestelmän kehitys kaatui kuitenkin sen korkeaan hintaan./6, s.164/

Autojen tankkausrobotteja on ollut käytössä muutamia lähinnä linja-autoille. Henkilöautoille markkinat ovat suuremmat, mutta turvallisuusvaatimusten vuoksi ne eivät ole aikaisemmin päässeet kehittymään riittävälle tasolle. Nyt on Hollannissa kehitetty ensimmäinen varsinainen henkilöautoille suunnattu automaattinen tankkausrobotti./5, 18/



Kuva 2.2 Tankkausrobotti/18/

Suuriin tiloihin, esimerkiksi lentokentille ja teollishalleihin, on ollu jo jonkin aikaa olemassa erilaisia robotteja siivousta varten. Nämä robotit liikkuvat vain rajatulla alueella ja ennalta määritellyn algoritmin mukaisesti. Vasta viime vuosina ovat imureiden valmistajat esitelleet kotitalouksiin soveltuvia pieniä robotti-imureita. Lähiaikoina on odotettavissa entistä parempia ja tehokkaampia robotti-imureita, jota pystyvät imuroimaan jopa pieniä kiviä. Lisäksi ne pystyvät paremmin suoriutumaan vaikeistakin paikoista./6, s. 164/

Yhdysvaltojen puolustusvoimien panostus robotiikan tutkimiseen ja soveltamiseen on ollut valtavaa. On kehitetty itsenäisesti liikkuvia robottiajoneuvoja, jotka pystyvät hyökkäämään erilaisissa muodostelmissa, erilaisia miinanraivausrobotteja sekä pieniä tiedustelurobotteja, jotka voivat liikkua maassa tai ilmassa. Kuitenkin sotateknologiset saavutukset ovat salaisia, eivätkä ne tule yleisön tietoisuuteen kuin vahingossa tai erilaisten kriisien yhteydessä./6, s. 166/

Lääketeiteessä robotin toistotarkkuus ja luotettavuusvaatimukset ovat lähes avaruusrobotiikan luokkaa. Merkkittäessä ja poratessa millimetrin tarkkaa reikää luuhun, ei ihmiskäsi vedä vertoja robotin tarkkuudelle. Ihmisen työksi muodostuukin todella suurta tarkkuutta vaativissatoimenpiteissä niiden valmistelu, valvonta ja potilaan jälkihoito./5, 6/

Autoteollisuus kehittää robotiikan sovelluksia sekä autotuotannon tarpeisiin että itse autoihin. Autopilot-ominaisuuden tuleminen autojen varusteeksi on vain ajan kysymys. Carnegie Mellonin Robotiikka-instituutiossa on jo kehitetty joitakin vuosia sitten auto, joka pystyy itsenäisesti liikkumaan normaalissa liikenteessä noudattaen liikennesääntöjä. Tämän jälkeen kehitystyö autorobotiikan saralla on mennyt valtavasti eteenpäin. Myös Japanissa on itsenäisiä ajoneuvojärjestelmiä kehitetty menestyksekkäästi. On vain ajan kysymys, milloin ihminen antaa koneelle vastuun kuljettamisesta./6, s. 167/

Robottien kehityksessä on otettu oppia luonnosta. Evoluution armottomassa kilpailussa voittajina selvinneet ominaisuudet ja ratkaisut ovat osoittaneet toimivuutensa muuttuvissa ja usein vihamielisissä olosuhteissa. Samanlaiset ominaisuudet ovat haluttuja ominaisuuksia myös vaikeissa olosuhteissa toimiville palveluroboteille. Soveltumiseen on otettu oppia eläinten liikuntatavoista, aistitavoista ja yksinkertaisista ohjausrakenteista. Ominaista luonnonjärjestelmille on sopeutuminen tiettyyn ekologian osaan kuitenkin välttäen globaalia optimointia. Luonnossa tarpeeksi hyvä, on riittävän hyvää hengissä pysymiseksi. Insinöörien taipumus optimoida kaikki viimeiseen asti, aiheuttaa ongelmia robottien siirtyessä pois valvotuista laboratorio-olosuhteista dynaamisiin ja kompleksisiin ympäristöihin, kuten valtamerten syvyyksiin ja kaupunkien ilmatilaan./6, s. 168/

Perinteinen SMPA-ohjausarkkitehtuuri (Sense-Model-Plan-Act), joka perustui ympäristön aistimiseen, mallintamiseen, reitin suunnitteluun ja toteuttamiseen, sai 1980-luvun puolessa välissä haastajan uudesta liikkuviin robotteihin soveltuvasta ohjausstrategiasta. Perinteinen kankea funktionaalinen ja sekventiaalinen ohjausjärjestelmä muutettiin rinnakkaismuotoiseksi, modulaariseksi ja vikoja sietäväksi järjestelmäksi, jossa itsenäiset modulit toimivat suoraan yhteydessä antureihin ja toimilaitteisiin. Kompleksinen herkkä ohjaussysteemi hajoitettiin useisiin eri tehtäviä suorittaviin osiin, joiden yhteisvaikutus mahdollisti aivan uusia järjestelmiä mihin perinteisellä menetelmällä olisi pystytty. Muutos, hitaasti liikkuvista ja reittejä suunnittelevista roboteista, nopeasti reagoiviin robotteihin oli valtavan suuri./6, s. 160-161/

Vähitellen alettiin kuitenkin yhdistellä molempien ohjaustapojen parhaita ominaisuuksia: mahdollisimman pitkälle pyrittiin selviämään reaktiivisten komponenttien avulla ja vasta tarpeen vaatiessa käytettiin hitaampaa ennaltasuunnittelua. Tätä hybridiarkkitehtuuria alettiin käyttää kaikkialla ja siitä kehittyi vähitellen de-facto ohjausmalli./6, s. 160-161/

2.2.2 Tulevaisuus

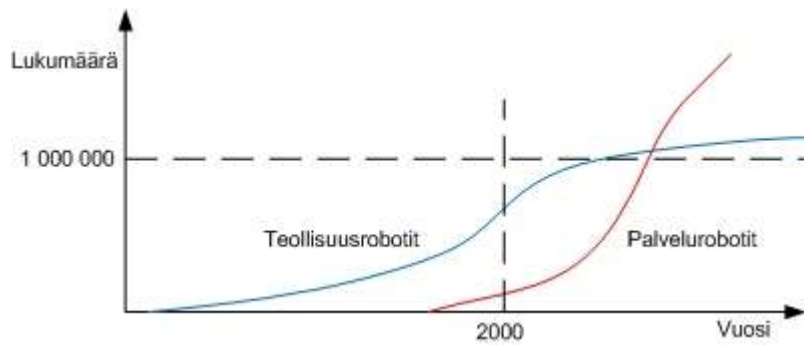
Isojen teollisuusrobottien kehitys on myös kulkenut pitkän matkan. Isot robotit, jotka toistivat samoja liikeratoja jatkuvasti työkierron mukana, ovat vihdoinkin kehittyneet pitemmälle ja saaneet riittävästi antureita ja muuta tekniikkaa tuekseen. Nyt robotit siirtyvät tehtaista jokapäiväiseen elämäämme. Honda on kehittänyt humanoidirobotteja pitkään ja kymmenen vuotta sitten julkaistiin ensimmäinen ASIMO (Advanced Step in Innovative MObility), joka pystyi kulkemaan vakaasti portaissa. ASIMOn kehitys on mennyt noista ajoista eteenpäin ja Honda käyttää uuden polven ASIMOja toimistossaan kerrospalvelijoina, viestien välittäjinä ja oppaina. ASIMOt pystyvät kommunikoimaan keskenään työtehtävistä, tarjoilemaan työntekijöille virvokkeita ja ne voivat tunnistaa 10-30 ihmisen kasvot. ASIMOn korkeantasoiset aistit- ja toimilaittejärjestelmät mahdollistavat vuorovaikutuksen

ihmisten kanssa. Ne voivat vastata kysymyksiin nyökkäämällä, pudistamalla päätään tai sanallisesti./6, 19/



Kuva 2.3 Asimo Hondan-messuilla/19/

Teollisuusrobottien ja palvelurobottien määrän ennustetaan kasvavan alla esitetyn kaavion mukaisesti. Teollisuusrobottien määrän kasvun uskotaan vakiintuvan, sillä teollisuus on jo nyt pitkälle automatisoitu ja robotit ovat laajalti käytössä. Uusien teollisuuslaitosten, jotka ottavat robotteja käyttöönsä, määrä on suhteellisen pieni verrattaessa jo olemassa oleviin. Yritykset kierrättävät robotteja tehtävistä toiseen kustannusten säästämiseksi ja vaihtavat uuteen robottiin vain tarvittaessa. Tällainen toiminta vähentää uusien robottien tarvetta teollisuudessa. Sen sijaan palvelurobotit ovat vasta tulossa markkinoille ja siksi niiden kasvu on nopeaa. Lisäksi palvelurobottien kirjon uskotaan olevan paljon suurempi kuin teollisuusrobottien, tästä syystä palvelurobottien määrän uskotaan ohittavan teollisuusrobottien määrän nopeasti ja moninkertaisesti.



Kuva 2.4 Teollisuus- ja palvelurobottien määrän kasvuennuste./6, s.171/

Ensimmäisiä ihmisen näköisiä robotteja, eli androideja, on valmistettu vuodesta 1997 lähtien. Japanilaiset ovat androidien tuotekehityksessä erittäin pitkällä. Robotteja jotka jäljittelevät ihmisen toimintoja tutkitaan monipuolisesti. Markkinoilta löytyy jo nukkeja, jotka ovat ilmeiltään ja eleiltään ihmisen kaltaisia. Näitä käytetään messuilla ja viihteen puolella./20/



Kuva 2.5 Actroid/19/

Jo nyt robotit ovat ihmisten kumppaneita ja tulevaisuudessa vielä enemmän.

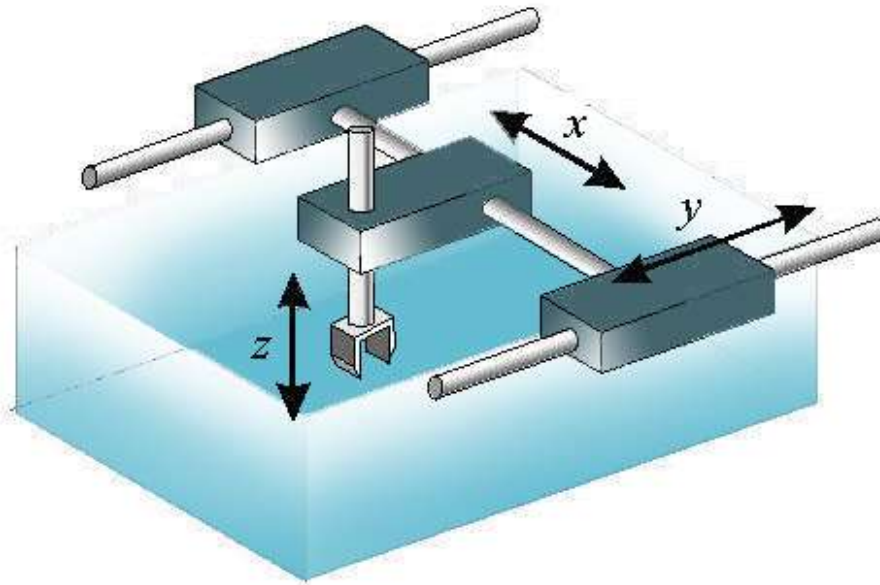
2.3 Robottimallit

Teollisuusrobotti on ISO 8373 määritelmän mukaan uudelleen ohjelmoitava, monipuolinen, vähintään kolminivelinen mekaaninen laite. Se on tarkoitettu kappaleiden liikuttamiseen, osien, työkalujen käyttöön teollisuuden sovelluksissa ja sitä ohjataan ohjelmoitavilla liikkeillä. Yksinkertaisesti, teollisuusrobotti on kone, joka liikuttaa kiinnityslaippaan kiinnitettyä työkalua halutulla tavalla. Robotin liikerata voi olla etukäteen määritelty, työympäristön perusteella valittava tai antureilla työliikkeiden aikana määritelty. Robotti koostuu jalustasta, työkalusta ja nivelillä varustetuista tukivarsista, jotka servomoottorien avulla mahdollistavat robotin liikkeet./5, 10/

Teollisuusrobotit jaetaan rakenteensa ja koordinaatistonsa mukaan eri luokkiin. Yleisin robottityyppi on kiertyvänivelinen. Siinä on yleensä kuusi kiertyvää niveltä ja sen suoraviivaiset liikkeet toteutetaan laskennallisesti. SCARA-robotti tyyppi soveltuu erinomaisesti ylhäältä alaspäin suuntautuviin asennustehtäviin. Sen kolme ensimmäistä niveltä on kiertyviä, ja ne paikoittavat työkalun kohteen yläpuolelle. Tätä robottityyppiä käytetään paljon elektroniikkateollisuudessa erilaisten tuotteiden valmistuksessa./5, 9/

2.3.1 Portaalirobotti

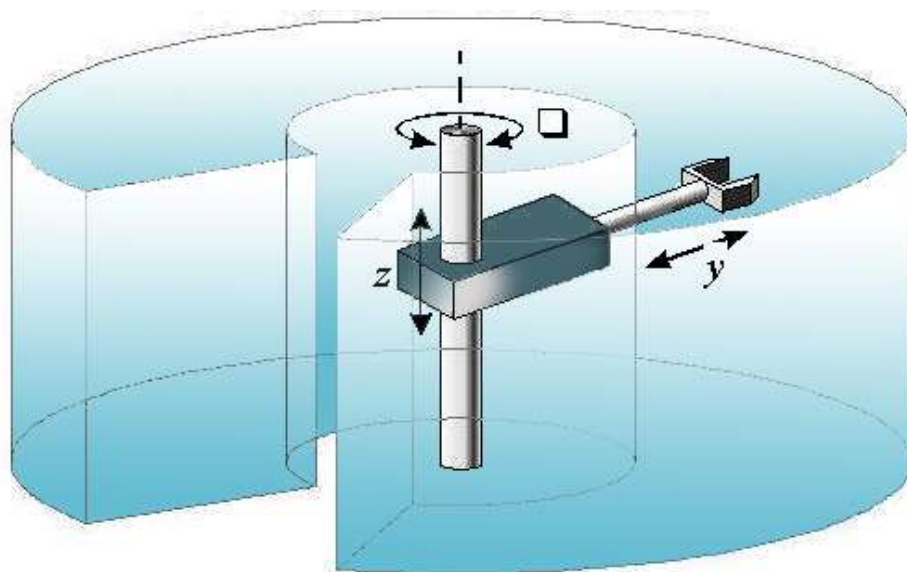
Portaalirobotin ovat suorakulmaisia robotteja. Niiden toimintaperiaatte perustuu kolmeen ensimmäiseen vapausasteeseen, joiden liikerata on lineaarinen. Portaalirobotin rakenne on tuettu palkeilla sen työalueen nurkista. Pystypalkit ja niiden välissä oleva poikittainen palkki muodostavat robotin rungon. Näin ollen robotilla on kolme vapausastetta. Tavallisimmin portaalirobotteja käytetään erilaisissa logistiikan automaatiosovelluksissa./9, 13/



Kuva 2.6 Portaalirobotti/9, 13/

2.3.2 Sylinterirobotti

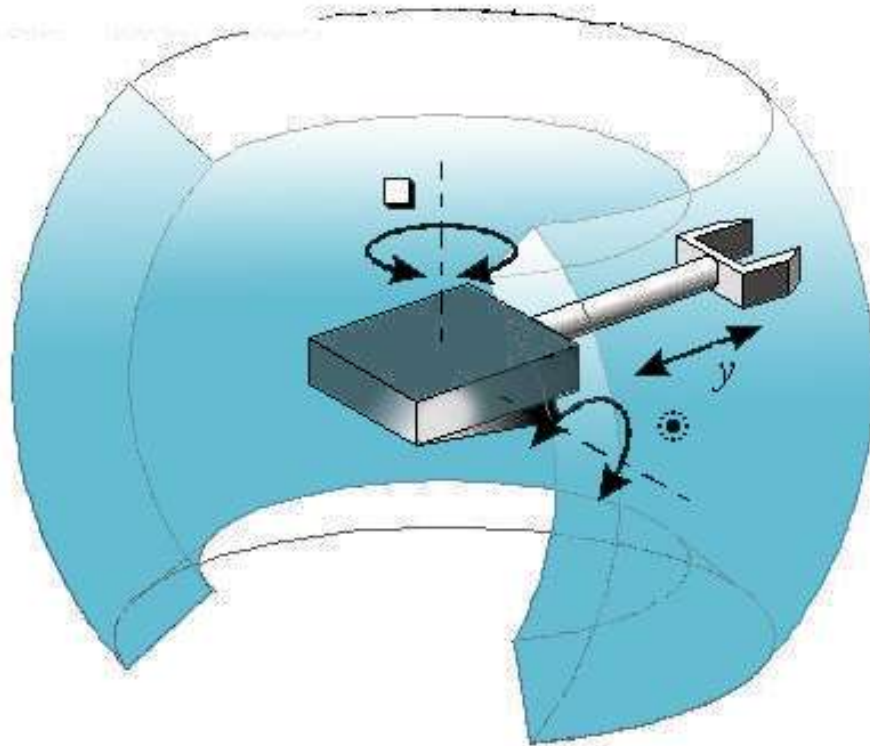
Sylinterirobotissa on yksi pyörivä nivel, sekä kaksi lineaarisesti liikkuvaa niveltä. Sylinterirobotin nimi tulee sen koordinaatiojärjestelmästä ja työalueen sylinterimäisestä muodosta. Sylinterirobotteja käytetään yleisesti manipulaattorina erilaisissa pakkaus- ja lajittelusovelluksissa, kuten suurten tietokoneyksiköiden varmuustalennevarastoissa./9, 13/



Kuva 2.7 Sylinterirobotti/9, 13/

2.3.3 Napakoordinaatistorobotti

Napakoordinaatistorobotin työskentelytila voidaan katsoa olevan täysin pyöreä, vaikka todellisuudessa sen on vaikea päästä joka paikkaan. Kiertyvänivelisenrobotin liikkeet perustuvat napakoordinaatiorobotin liikkeisiin. Tällaista robottia käytetään yleisesti pistehitsauksessa, kaasus- ja kaarihitsauksessa, ruiskuvalutöissä sekä työstökoneiden palvelutehtävissä./9, 13/



Kuva 2.8 Napakoordinaatistorobotti/9, 13/

2.3.4 SCARA-robotti

Scara-robotti (Selective Compliance Assembly Robot Arm) on suunniteltu pystysuoraan asennus- ja kokoamistyöhön. Scara-robotin voidaan sanoa muistuttavan vaakatasossa olevaa ihmisen kättä, paitsi ranne on robotissa pystysuoraan liikkuva pystyjohde. Yleisesti Scara-roboteilla on vain neljä vapausastetta. Robotin vahvuuksia on sen jäykkä runko, suuri liikenopeus ja tarkkuus. Scara-robotteja käytetään yleisemmin pienien kappaleiden kokoonpano- ja tarkastustöihin.

Elektroniikkateollisuus käyttää Scara-robotteja yleisesti matkapuhelinten ja muiden elektronisten laitteiden kokoonpanolinjoissa./9, 13/



Kuva 2.9 Scara-robotti/17/

2.3.5 Kiertyvänivelinen robotti

Kiertyväniveliseksi robotiksi lasketaan sellaiset robotit, joissa on vähintään kolme kiertyvää niveltä. Kuitenkin yleisimmissä robottimalleissa on kuusi niveltä. Kiertyvänivelisen robotin toimintaperiaate muistuttaa hyvin pitkälle ihmisen kättä. Nykyiset teollisuusrobottien rakenteet perustuvat lähes aina tähän mekaniikkaratkaisuun sen monipuolisuuden perusteella. Kiertyvänivelistä robottia pidetäänkin teollisuudessa yleisrobottina sen monipuolisuuden vuoksi. Näiden robottien eduksi lasketaan niiden hyvä monipuolisuus, mukautuvuus ja suurehko ulottuvuus. Tämän robottimallin huonona puolena voidaan kuitenkin pitää pienehköä kuormankantokykyä, tosin erilaisilla mekaanisilla ratkaisuilla on tämäkin ongelma

saatu ratkaistua. Kiertyvänivelisiä robotteja nähdäänkin teollisuudessa kaikenlaisissa sovelluksissa ja tehtävissä./9, 13/



Kuva 2.10 Kiertyvänivelinen robotti/15/

2.3.6 Rinnakkaisrakenteinen robotti

Rinnakkaisrakenteinen robotti eroaa muista roboteista sen mekaanisen rakenteen vuoksi. Robotin mekaaninen rakenne muodostuu kolmesta liikeakselista, jotka on asennettu rinnan. Nämä robotit ovat erittäin nopeita ja tarkkoja. Rinnakkaisen rakenteen vuoksi robottien työalue jää pieneksi. Näitä robotteja näkee teollisuuden pick&place-työkohteissa, joissa kuljettimelta otetaan kappale ja siirretään toiselle kuljettimelle. Elintarviketeollisuus käyttää usein rinnakkaisrakenteisia robotteja tuotantolinjoissaan./9, 13/



Kuva 2.11 Rinnakkaisrakenteinen robotti/15/

Nykyaikana robotti ei ole enää vain itsenäisesti toimiva työkalu, vaan se on osa kokonaista toimintaympäristöä, puhutaan robottisolusta. Robotti muodostaa liikeratansa tuotteiden suunnittelutiedoista ja ympäristömallista. Robottijärjestelmään kytketyt anturit päivittävät robotin liikerataa jatkuvasti työprosessin aikana.

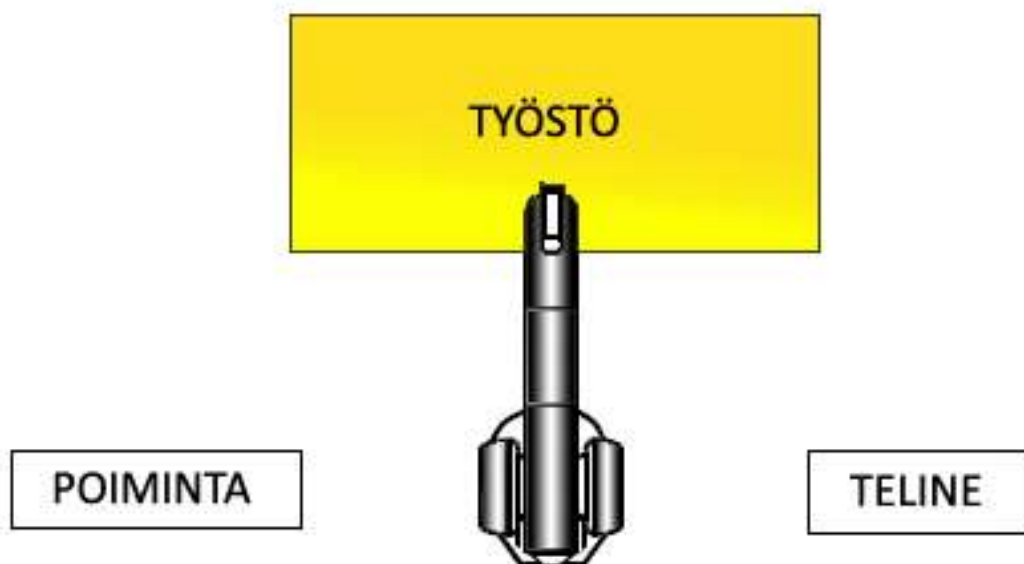
Robottijärjestelmä koostuu seuraavista osista:

- työkalu
- anturit, jotka tarkkailevat robotin liikkeitä ja ympäristöä
- robotin käsivarsi
- robotin ohjausjärjestelmä, robotin toimintakyky on riippuvainen siihen ohjelmoidusta ohjelmasta
- oheislaitteet, jotka ovat liitetty robottijärjestelmään
- mahdolliset ulkoiset tietokoneet, jotka ohjaavat robotin toimintaa/5, s. 112-113/

3 ABB ROBOTTI

3.1 Järjestelmän yleiskuvaus

Tämän projektin tavoitteena oli suunnitella ja toteuttaa case-yrityksenä toimineelle Puustelli Oy:lle uusinta teknologiaa hyödyntävä demonstraatio levyntyöstökonetta palvelevasta konenäköjärjestelmästä, jossa yleisrobotti poimii yksittäisiä levyjä pinoista konenäön ohjeistuksella.



Kuva 3.1 Projektin järjestelmän yleiskuvaus

Järjestelmässä kattoon sijoitettu älykamera luki levyissä olevista tarroista matriisikoodit. Ensimmäiseksi kamera kuvasi poiminta-alueella olevat levyt ja tunnisti kuvasta, missä koordinaateissa (x, y ja z) levyt olivat. Tässä tapauksessa z-koordinaatti tarkoitti levyn yläpinnan etäisyyttä poiminta-alueen pinnasta.

Ylhäältä kuvaava kamera tunnisti poiminta-alueella olevat levyt niissä olevien tarrojen perusteella ja paikansi tarrojen keskipisteiden koordinaatit (x ja y). Kamera

määritti tunnistettujen tarrojen korkeudet poiminta-alueen pinnasta (z-koordinaatti) perustuen laserviivan poikittaissuuntaiseen siirtymään pinon korkeudesta riippuen.

Tämän jälkeen korkeimman eli suurimman z-arvon omaavat koordinaatit lähetettiin robotille, joka poimi levyn annetuista koordinaateista.



Kuva 3.2 Levyn poiminta

Robotin avulla levy käännettiin sivukameran kuvattavaksi. Alipainetunnistimia käytettiin robotin tartunnassa apuna, jotta korkeussuunnan koordinaatin toleranssi ei olisi aiheuttanut virhettä tartuntapisteeseen.



Kuva 3.3 Kuvaus sivukameralla

Sivusta kuvaavan kameran ohjelma mittasi kuvasta levyn korkeuden ja leveyden sekä määrittä kappaleen pyörimiskulman.



Kuva 3.4 Sivukamera

Kulmatietoa hyväksi käyttäen robotti käänsi levyn aina samaan asentoon viedessään sen pöydälle, joka toimi demonstraatiossa levyntyöstökoneen alkupisteenä.



Kuva 3.5 Levyn laitto työstökoneeseen

Kun robotti oli jättänyt levyn koneelle työstettäväksi, se palasi kotipisteeseensä ja lähti uudestaan hakemaan levyä työstökoneesta. Lopulta robotti vei levyntyöstökoneesta hakemansa levyn lokerikon ensimmäiseen vapaaseen paikkaan. Varsinaisessa järjestelmässä robotti olisi poiminut levyjä työstökoneen liukuhihnalle kunnes työstökone olisi ilmoittanut valmistuneesta levystä, jonka voi siirtää lokerikkoon.

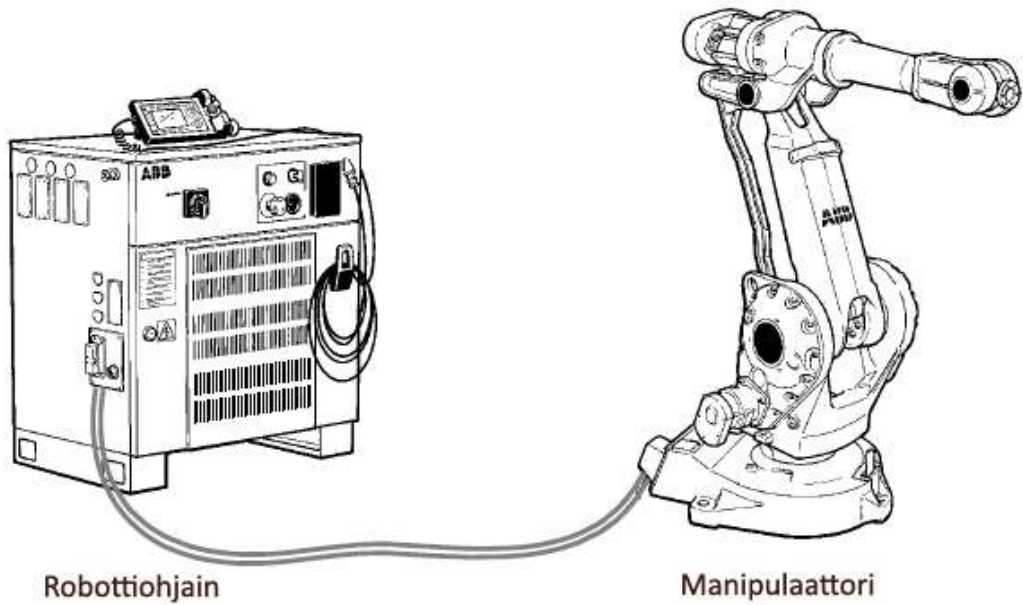


Kuva 3.6 Levyn laitto telineeseen

Jätettyään levyn lokerikkoon, robotti palasi odotuspisteensä ja aloitti työkierron uudestaan alusta. Robotti käsitteli levyjä niin kauan kuin levyjä oli poiminta-alueella tai kunnes lokerikko tuli täyteen. Lokerikon täytyttyä robotti antoi äänimerkin ja jäi odottamaan lokerikon vaihtoa tai tyhjennystä.

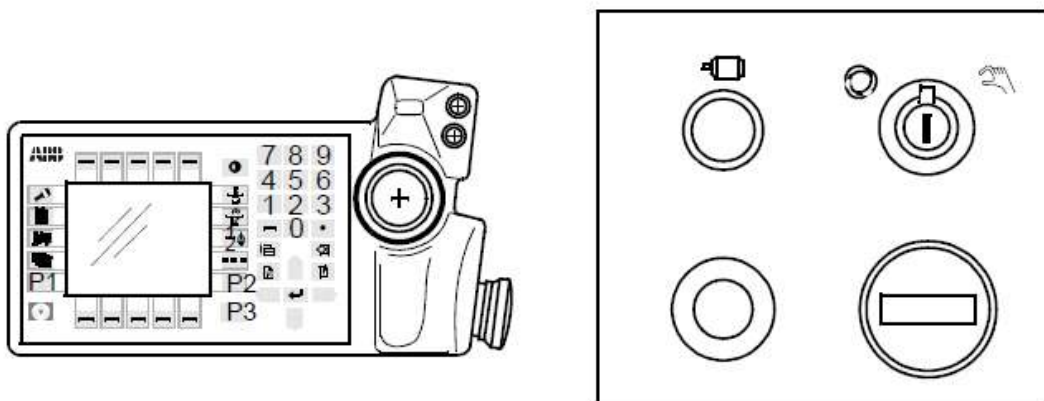
3.2 Robottijärjestelmä

Robotti koostui kahdesta pääasiallisesta osasta: robottiohjaimesta ja manipulaattorista./1, 3/



Kuva 3.7 Robotin osat/1, 3/

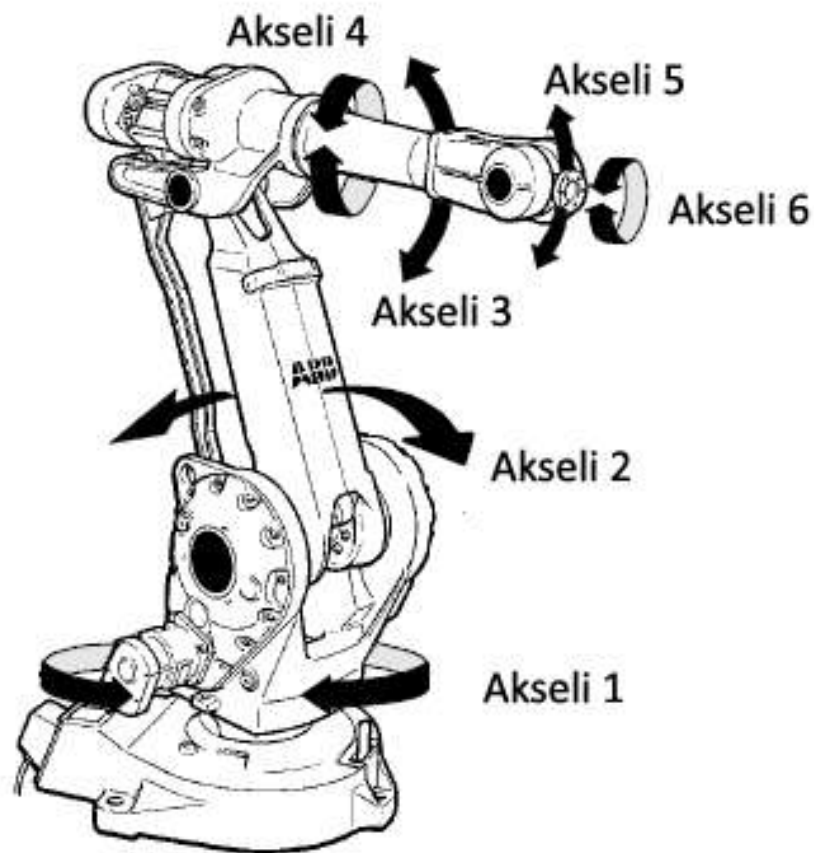
Robotin kanssa voitiin kommunikoida käyttämällä joko käsiohjainta tai robottiohjaimen käyttöpaneelia. Paneelin käyttö kuitenkin rajoittui robotin käynnistämiseen, sammuttamiseen ja toimintatilojen vaihtamiseen. Käsiohjaimella tapahtui varsinainen kommunikointi robotin kanssa. Sillä pystyi antamaan liike- ja muita toimintakäskejä robotille, ohjaamaan robottia manuaalisesti sekä robotti antoi virheilmoitukset ja muut tiedot käyttäjälle käsiohjaimen välityksellä./1, 3/



Kuva 3.8 Robotin ohjaimet/1, 3/

3.3 Manipulaattori

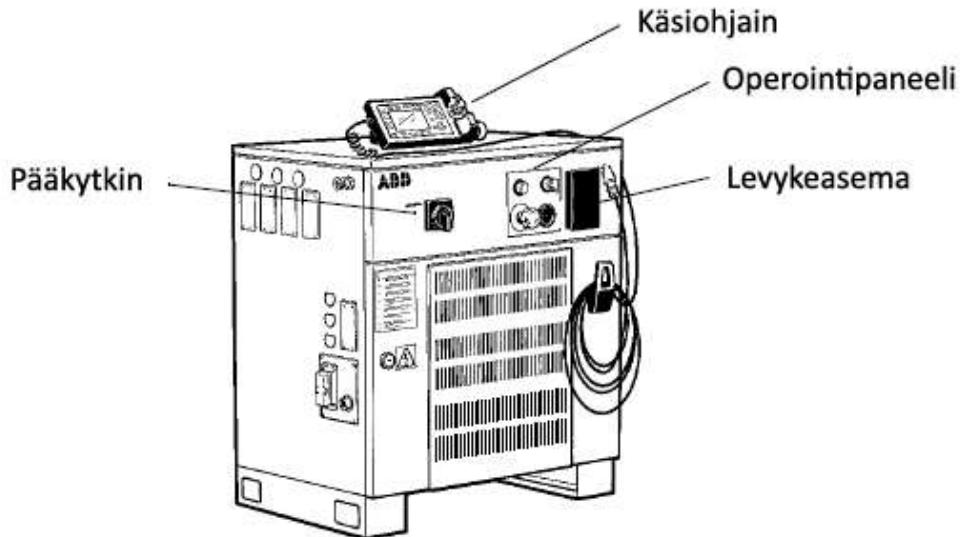
ABB:n yleisrobotin manipulaattori oli itseasiassa robotin varsinainen runko ilman siihen kiinnitettyä tartunta- tai muuta työkalua. Projektin käytössä ollut robotti oli kiertyvänivelinen 6-akselinen yleisrobotti, joka oli varsinaisesti suunniteltu maalausrobotiksi. Alla olevasta kuvasta voi nähdä robotin akselien liikesuunnat. Kuuden akselin käyttäminen mahdollisti robotin käyttämisen lähes täydellisesti toimintasäteen alueella./1, 3/



Kuva 3.9 Robotin akselit ja liikesuunnat/1, 3/

3.4 S4C Robottiohjain

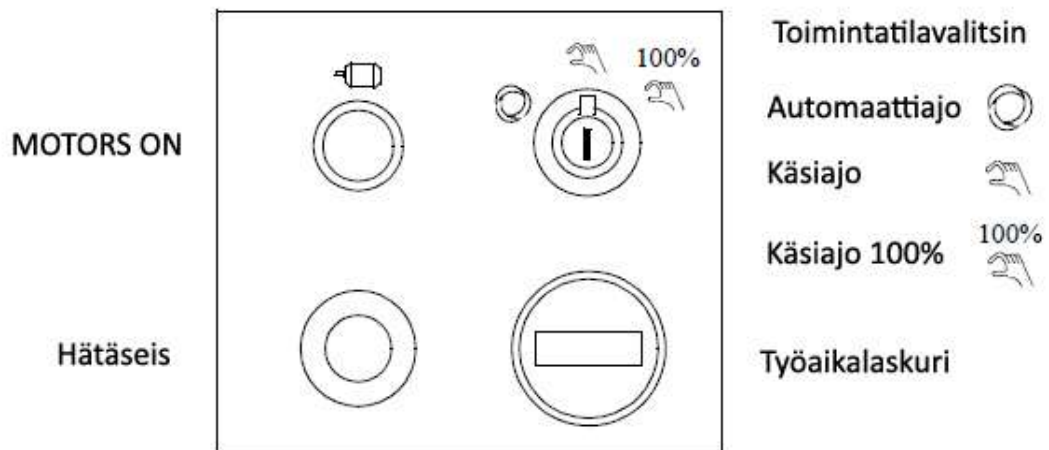
Yleisrobotin robottiohjaimen kuului neljä pääosaa: pääkytkin, käsiohjain, operointipaneeli ja levykeasema. Käsiohjaimella tapahtui varsinainen ohjaus ja kommunikointi robotin kanssa. Virransyöttö tapahtui pääkytkimellä ja operointipaneelilla valittiin robotin toimintatila. Levykeasemaa tarvittiin apuna robotin ohjelmoinnissa./1, 3/



Kuva 3.10 Robottiohjain/1, 3/

3.5 Operoitin paneeli

Robottiohjaimessa olevalla operointipaneelin avulla voitiin valita robotin käyttömukainen toimintatila. Lisäksi operointipaneelissa oli hätäseis-kytkin vaaratilanteiden hallintaa varten.



Kuva 3.11 Operointipaneeli/1, 3/

Operointipaneelin eri osilla oli omat merkityksensä, jotka vaikuttivat suuresti robotin toimintaan. Erityisesti toimitilavalitsimella oli merkitystä robotin käyttäytymiseen ja nopeuteen. Ohessa on esitetty robottiohjaimen operointipaneelin toiminnot./1, 3/

MOTORS ON

Robotin ollessa **MOTORS ON** tilassa, robotin moottorit ovat aktiiviset ja merkkivalo palaa.

Operointi tila AUTOMATIC (Tuotanto tila)

Käytetään suoritettaessa valmista tuotannollista ohjelmaa. Tässä tilassa robottia ei voi ohjata käsin. Automaattitilassa on ehdotonta, ettei robotin työalueella ole ketään, sillä robotin liikkeet voivat olla äkillisiä ja toimiessaan robotti voi aiheuttaa hengenvaarallisen tapaturmatilanteen.

Operointi tila KASIAJO (Ohjelmointi tila)

Käytetään työskenneltäessä robotin työalueella ja ohjelmoitaessa robottia. Tässä toimintatilassa on robotin maksimi liikenopeus rajoitettu 250mm/s.

Operointi tila KÄSIAJO (Testaus tila)

Käytetään testattaessa robotin ohjelmaa täydellä nopeudella. Tässä tilassa robotti toimii kuten automaattitilassa, sillä erotuksella, että käsiohjaimen START-nappulaa on pidettävä aktiivisena. On suositeltavaa, että tätä toimintatilaa käyttävät vain koulutetut henkilöt.

Hätäseis

Robotti pysähtyy välittömästi riippumatta siitä, missä operoinnin tilassa robotti on. Kytkin pysyy vaikuttuneena ja robotin moottorit voidaan aktivoida vasta kun hätäseis-kytkin vapautetaan.

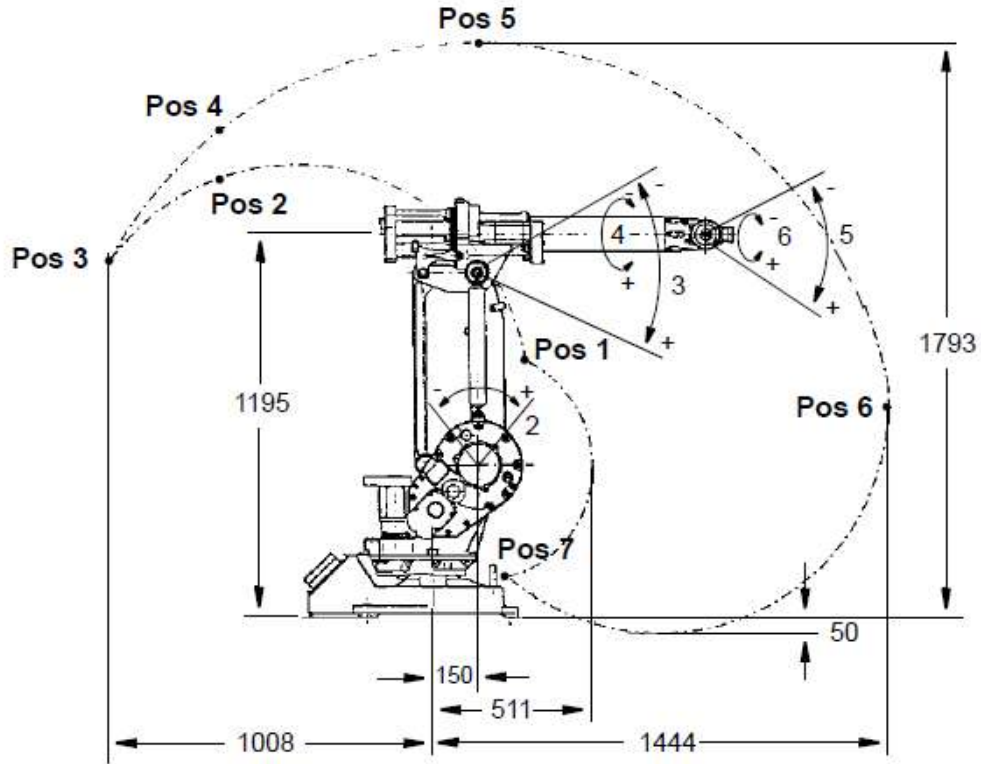
Työaikalaskuri

Ilmoittaa robotin käyttöajan.

/3, s. 43/

3.6 Robotin liikealueet

Kuuden vapaasti kiertyvän akselin käyttö robotissa mahdollisti vapaan ja laaja-alaisen robotin käytön koko sen toiminta-alueella. Koska robotti pystyi pyörittämään lähes täydet 360 astetta sivusuunnassa, muodostui robotin toiminta-alueeksi lähes täydellinen pallokuvio.



Kuva 3.12 Robotin liikealueet ja ääripisteet/1, 3/

Taulukko 3.1 Robotin akselien liikealueet

Liiketyyppi	Liikealue
Akseli 1 kiertoliike	+170° _ -170°
Akseli 2 puomiliike	+70° _ -70°
Akseli 3 puomiliike	+70° _ -70°
Akseli 4 ranneliike	+150° _ -150°
Akseli 5 taivutusliike	+115° _ -115°
Akseli 6 pyöritysliike	+300° _ -300°

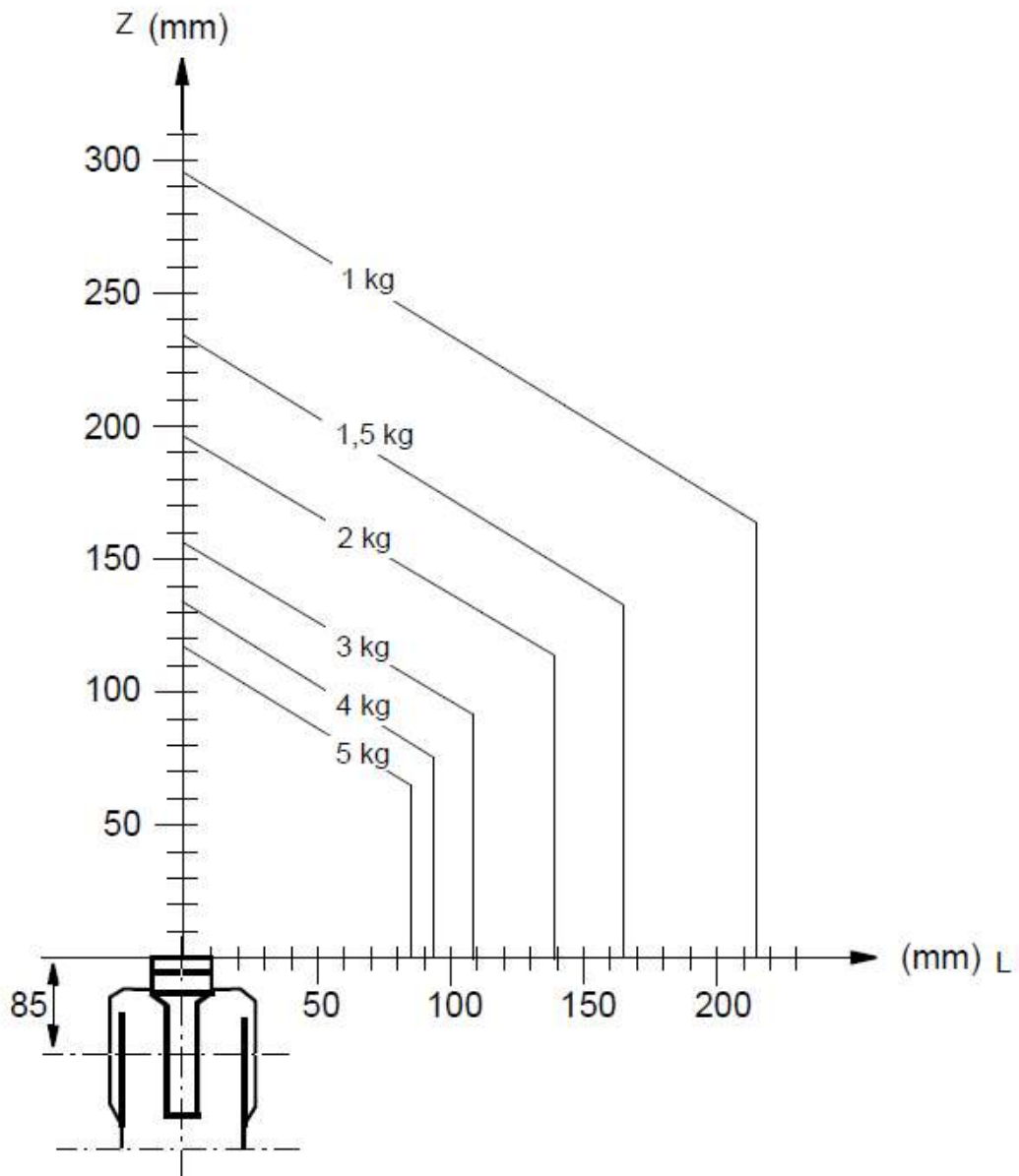
Ylläolevasta taulukosta nähdään robotin liikealueiden laajuus akseleittain. Alla olevasta taulukosta nähdään sen sijaan kuvaan 3.12 merkittyjen ääriasentojen kulmat akselien 2 ja 3 suhteen. Robotin muiden akselien asennoilla ääripisteissä, ei ole suurta merkitystä. Kuvasta nähdään myös robotin liikealueen maksimi ulottovuudet millimetreinä./1, 3/

Taulukko 3.2 Akselien 2 ja 3 kulmat ääripisteissä

	Akseli 2	Akseli 3
Pos 1	-70°	+70°
Pos2	-70°	-35°
Pos 3	-70°	-65°
Pos 4	-43°	-65°
Pos 5	-6°	-65°
Pos 6	+70°	-65°
Pos 7	+70°	+70°

3.7 Robotin kuormitus

Projektin käytössä ollut ABB: yleisrobotti oli alkujaan suunniteltu maalausrobotiksi. Tästä syystä robotin maksimi kantokyky oli varsin pieni, vain 5 kiloa. Myös robottiin asennettu työkalu vähensi robotin maksimi kuorman määrää. Työkalun painon lisäksi myös sen fyysisellä koolla oli merkitystä robotin kuorman suuruuteen./1, 3/



Kuva 3.13 Työkalun painopisteen vaikutus robotin maksimi kuormaan/1, s.36/

Kuvasta nähdään robottiin asennetun työkalun fyysisen koon vaikutus maksimi kuormaan. Mitä suurempi työkalu on, sitä pienempi on robotin kuorman kantokyky. Paras kuorman kantokyky on työkalulla, joka on kooltaan ja painoltaan pieni ja sen

massakeskipiste sijaitsee mahdollisimman lähellä robotin työkalulaipan keskipistettä./1, s. 36/

3.8 Robotin kalibrointi

Robotin kaikki akselit ja moottorien kierroslaskurit pitää olla kalibroituina ja synkronoituina, jotta robotti toimisi luotettavasti ja oikein. Robotin kalibroinnista on kerrottu tarkemmin tämän opinnäytetyön luvussa 7.2 Kalibrointi.

4 TARTTUJA

4.1 Robottitarraimet ja työkalut

Se mekaaninen osa, jota robotti siirtää paikasta toiseen, sanotaa robotin työkaluksi. Yleisin robotin työkalu on tarrain, mutta myös maaliruisku, hitsauspistooli, hiomakone ovat robotin usein käyttämiä työkaluja. Tarraimen suunnittelu on tärkeä osa robottisovellusta. Tarraimen suunnittelussa ja valinnassa on tunnettava eri tarraintyypit ja tartuntatavat. Niiden perusteella tarraimet voidaan seuraaviin luokkiin:

- avautuvat ja sulkeutuvat tarraimet liikkeen mukaan
- kiertyväsormiset ja rinnakkain suoraan liikkuvat sormitarraimet
- pneumaattiset, hydrauliset tai sähköiset tarraimet
- kaksi-, kolmi- tai useampisormiset tarraimet
- jäykät ja joustavat tarraimet
- kappalekohtaiset tai yleistarraimet
- keskittävät tarraimet
- magneettiset tarraimet
- alipainetarraimet
- sisäisesti laajenevat tarraimet
- yksittäinen, kaksois- tai revolveritarrain
- älykkäät anturoidut tarraimet
- erikoistarraimet

Yksittäinen tarrain voi kuulua samanaikaisesti useampaan luokkaan. Tarraimen sormien liikkeet saadaan aikaiseksi mekaanisesti. Sovelluksissa joissa mekaanisen tarraimen käyttö on hankalaa tai yksinkertaisesti poissuljettu mahdollisuus, käytetään alipaineeseen perustuvia tartuntaelimiä./9, s. 60/

4.1.1 Imutartunta

Imutartunnassa kappaleeseen tartutaan vain yhdestä suunnasta. Imukupit, jotka ovat yleensä kumia tai muovia, eivät naarmuta kappaleen pintaa. Isompia kappaleita varten tarvitaan suurempi tartuntavoima, jota voidaan kasvattaa helposti lisäämällä imukuppien lukumäärää. On kuitenkin huomattava usean imukupin järjestelmässä, että yhden imukupin irtoaminen aiheuttaa alipaineen häviämisen koko järjestelmästä. Imukupit tarvitsevat tartuntapinnaksi sileän, puhtaan ja tiiviin pinnan. Imukupitartunta toteutetaan yleensä keskitetysti kappaleen painopisteen lähelle, jolloin saadaan mahdollisimman vakaa tartunta. Robottisovelluksesta ja paikasta riippuen voidaan tarvittava alipaine tuottaa robotille joko erillisellä alipainepumpulla tai ejectorilla/venturilla. Jälkimmäistä tapaa käytetään jos on mahdollista käyttää valmista paineilmaverkostoa./9, s. 63-64/

Imukupin rakenne on yksikertainen ja se on luotettava, koska siinä ei ole liikkuvia ja kuluvia osia. Suurimpana ongelmana on kappaleen putoaminen alipaineen kadotessa, lisäksi keskittämisen puute voidaan katsoa ongelmakohtaksi./9, s. 64/

4.1.2 Magneettitarrain

Vain magneettisille aineille voidaan käyttää magneettitarrainta. Sen nostovoima on riippuvainen nostettavan kappaleen materiaalista, muodosta, pinnanlaadusta ja magneetin lämpötilasta. Yleensä vaatimuksena on riittävä tasainen pinta, sillä ilmaerot heikentävä tarttumista. Magneettitarrain tarttuu nopeasti, mutta irrotus on hitaanpaa jälkimagnetismin vuoksi. Kestomagneetteja käytettäessä tarvitaan irroitusta varten erillinen laite. Sähkömagneeteilla voidaan kääntää magneettikentän suunta, jolloin irroitus tapahtuu nopeasti./9, s. 64/

4.1.3 Vakiotarrain

Robottien valmistajien tuotevalikoimista löytyy yleensä muutama erilainen vakiotarrain, joista voi helposti koota ja muokata sovellukseen sopivan tarraimen. Yleensä tarrainta täytyy kuitenkin muokata sovellukseen sopivaksi, yksinkertaisimmillaan tartuntapintojen muokkauksella./9, s. 64/

4.2 Tarraimen suunnittelu

Yksi robottijärjestelmän suunnittelun vaiheista on tarraimen suunnittelu. Tarraimen tai työkalun suunnittelu on pieni, mutta tärkeä osa robottijärjestelmän kokonaisuudesta. Tarraimelta toivotaan pientä kokoa ja painoa, sillä robotin hyötykuorma lasketaan ilman työkalua jolloin työkalu pienentää hyötykuormaa. Tarraimen pitää tarttua luotettavasti kappaleeseen, kappale voi tarvita keskittämistä ja sovelluksesta riippuen tarraimen pitää olla lepotilassa kiinni. Luotettavan toiminnan kannalta tärkeimmät asiat ovat robotin hyötykuorma, tartuntamenetelmä, toleranssit, tarraimen luoksepäästävyys ja kunnossapitomahdollisuus. Tarraimen suunnittelussa on tärkeää huomioida tarraimelle tehtävän mukaisesti asetetut vaatimukset ja määrittelyt.

Tarraimen suunnittelussa haetaan tehtävään soveltuvaa toimintaperiaatetta, tartuntapintoja ja määritellään tarraimen anturoinnin tarve. Työkappaleeseen liittyvät ongelmat (koko, muoto, massa, materiaali, paikka) vaikuttavat suoraan tartuntatyökalun suunnitteluun ja valintaan. Ne määräävät käytettävän tartuntamekanismin, tartuntavoiman ja tarvittavat anturit. Myös robotin toimitaympäristö vaikuttaa omalta osaltaan esimerkiksi tarraimen liikealueeseen./9, s. 64/

Tartunta perustuu joko kappaleen muotoihin (muotosulkeinen) tai puristusvoimaan (kitkasulkeinen), se voi kuitenkin olla myös näiden yhdistelmä. Työkalun tartuntavoiman pitää olla riittävän suuri, että kappale pysyy paikallaan luistamatta tarraimesta. Liian suuri tartuntavoima voi rikkoa kappaleen ja liian pieni ei pidä

kappaletta tarraimessa. Herkille ja pienille kappaleille on parempi käyttää muotosulkeista tartuntaa, sillä silloin tartuntavoima voi olla pienempi, eikä kappale vaurioidu sitä käsiteltäessä./9, s. 65-72/

Tarraimen suunnittelun yhtenä tavoitteena on luoda taloudellisesti turvallinen ja luotettava tartuntatyökalu. Koko suunnitteluprosessin aikana turvallisuustekijät on otettava huomioon kaikilla tasoilla. Erilaiset määräykset ja standardit asettavat omia turvallisuusvaatimuksia, jotka pitää ottaa huomioon yhdessä työtehtävän vaatimusten kanssa. Teollisuusrobotti ja tarrain ovat yhdessä kokonaisuus, jonka turvallisuus ja luotettavuus on suunniteltava kokonaisuutena. Muotoilulla ja turvallisesti vikaantuvan toiminnan periaatteella voidaan lisätä tarraimen turvallisuutta./9, s. 65-72/

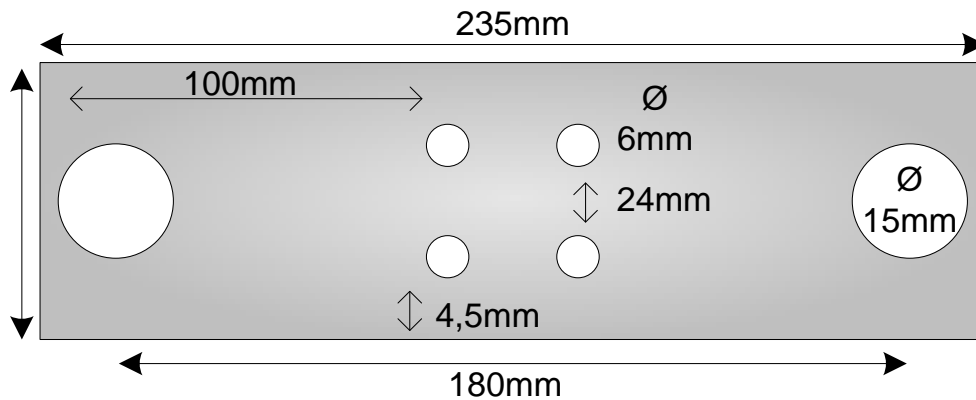
Tarraimen tai työkalun pitäisi olla kevyt, joten alumiinia ja muoviosia on suositeltava käyttää. Rakenteellisesti voidaan lisätä kevyttä lisäämällä reikiä. Tarraimen pitäisi olla kooltaan pieni, mutta samalla sillä pitäisi olla laaja erikokoisten kappaleiden käsittelykapasiteetti. Hankalissakin tilanteissa tarraimen pitää olla riittävän jäykkä ja puristusvoimainen. Mikäli tarraimessa on antureita, ne tulee olla suojatut sekä mekaanista rasitusta että sähköisiä häiriöitä vastaan. Huollon kannalta tarraimen modulaarinen, yksinkertainen rakenne ja johtojen vaihdettavuus ovat huoltoa helpottavia tekijöitä. Kokoonpantavuus, huollettavuus, valmistettavuus sekä kustannukset ovat erityisen tärkeitä asioita tarraimen yksityiskohtien suunnittelussa. Yksinkertainen laite on halvin ja luotettavin./9, s. 65-72/

4.3 Projektin tarttuja

Tässä projektissa käsiteltävät tuotteet (kalustelevyt) asettivat robotissa käytettävälle tartuntatyökalulle vaatimuksia. Levyjen pinta ei käsiteltäessä saanut naarmuuntua, likaantua eikä muutenkaan vioittua millään muulla tavoin. Tämän vaatimuksen perusteella robotin tartuntatavaksi kelpasi vain imutartunta. Levyjen koko ja paino aiheuttivat ongelmia, sillä osa levyistä oli huomattavasti painavempia kuin mihin robotin kantokyky riitti. Robotti ei jaksanut nostaa painavimpia levyjä poiminta-alueena toimineelta pöydältä. Levyt jotka olivat lähellä robotin kuorman maksimi

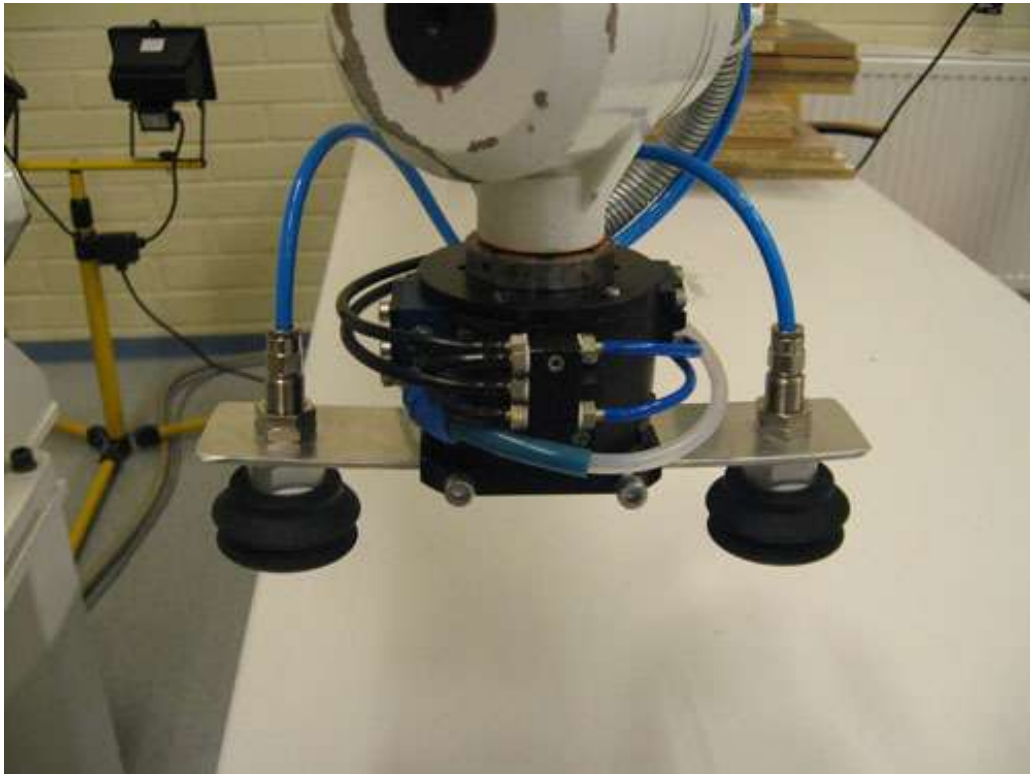
painoa, robotti sai nostettua ylös poiminta-alueelta, mutta ne eivät pysyneet tarttujassa robotin liikuessa sivusuunnassa. Tästä syystä projektissa ei voitu käyttää kaikki levyhällejä, joita oli alkuperäisesti tarkoitus käyttää. Levyiksi valittiin ne, joiden paino alitti riittävästi robotin kantokyvyn maksimi painon.

Myös levyjen koko tuotti myös ongelmia. Levyjen suuri pinta-ala aiheuttise, että levyä liikuteltaessa sen hallittavuus heikkeni ja syntyi heiluntaa. Levyn heilumisesta johtuen imukupissa oleva alipaine pääsi helposti katoamaan ja robotin käsittelyssä oleva levy pääsi tippumaan.



Kuva 4.1 Tarttujan tukirakenne

Levyjen vakausongelman ratkaisemiseksi suunniteltiin yllä olevan kuvan mukainen tuki imukupitartunnalle. Samalla lisättiin toinen imukuppi, jolloin tartuntavoima kasvoi ja vakaus lisääntyi huomattavasti.



Kuva 4.2 projektin imukupitarttuja

Yllä olevassa kuvassa on projektissa käytetty tartuntatyökalu valmiiksi asennettuna robotin työkalulaippaan. Tarttujassa käytetyt 50 mm imukupit takasivat riittävät tartuntavoiman suurempienkin kappaleiden käsittelyyn. Myöskään robotin liikkeiden nopeudella ei ollut enää vaikutusta, vaan robotti pystyi pitämään tartuntaotteessaan käsiteltäviä levyjä täydelläkin nopeudella toimiessaan.

5 KOMMUNIKOINTI

5.1 Tiedonsiirtoliitännät

Tiedonsiirtoliityntä on välttämätön osa nykyaikaista joustavaa reaaliaikaista robottijärjestelmää. Sen avulla robotille voidaan antaa käskyjä ja välittää tuotantoa ohjaavia parametreja ylemmän tason järjestelmästä. Tiedonsiirtoliityntää tarvitaan myös kun robottiin halutaan liittää kamerajärjestelmiä, toinen robotti tai nykyaikainen työstökone. Lisäksi tietokoneiliitynnällä voidaan robottiin liittää ulkoinen PC, joka pystyy toimimaan massamuistiasemana robotille. Robottiohjaimiin asennettujen verkkokorttien kanssa robotista tulee osa yrityksen lähiverkkoa. Tämä auttaa ylemmän tason ohjauksen hajautusta useammalle järjestelmälle.

Robottien tiedonsiirto alkoi varsinaisesti 1970-luvulla, kun ABB otti S2-ohjaimessaan käyttöön oman ISO-7001 standardiin perustuvan tiedonsiirtoprotokollan. Sitä käytettiin aina vuoteen 1994 asti, jolloin julkaistiin S4-ohjain. Tällöin siirryttiin käyttämään TCP/IP-protokollaa. Muiden valmistajien ensimmäiset tietokoneiliitännät ilmestyivät 1980-luvulla, kun ensimmäiset varsinaiset PC-tietokoneet yleistyivät./9, s. 48-50/

Taulukko 5.1 Tiedonsiirtoliitännät

Valmistaja	Ohjaintyyppi	Liityntä	Siirtotiet	Protokolla
ABB	S2/S3	Computer Link	RS-232	ARAP/ADLP10
ABB	S4	RAP	RS-232, 10Base-T	TCP/IP, RPC
Fanuc	R1/R12	Data Transfer, Sensor Interface	RS-232	OI
Motoman	ERM/MRC	Datatransmission Function	RS-232	BSCLIKE
Motoman	MRC	Datatransmission Function	10Base2	TCP/IP

Nykyisin robottiohjain on varustettu vähintään RS-232 -sarjaliikenneportilla, mutta ohjaimessa voi olla myös RS-422 -liityntä. Sarjaportit ovat olleet edullisia ja yksikertaisia, kuitenkin useamman kuin kahden robottiohjaimen liittäminen samaan tietokoneeseen on vaatinut erillisen sarjaliikennekortin, sillä tietokoneissa on ollut vakiona vain 1-2 sarjaporttia. Robottiohjaimissa olleet sarjaportit ovat yleensä olleet tietokoneita vastaavat D-liittimet, mutta esimerkiksi ABB:n roboteissa ovat sarjaportit vaihdelleen ruuviliittimistä lähtien erilaisiin eksoottisiin liittimiin./9, s. 49/

Vuoden 1997 aikana on suurimmilta robottien valmistajilta ilmestynyt sarjaportin rinnalle ethernet-liityntä. Siinä on käytetty standardin mukaista parikaapelia, joka on helppo ja halpa ratkaisu robotin ja sen oheislaitteiden liittämiseksi lähiverkkoon.

Tietokoneen ja robotin välisen liitynnän voi joko tehdä itse tai käyttää valmistajan tarjoamaa optiota. Itsetehtynä liityntä tapahtuu osana robotin työkierto-ohjelmaa, jossa käytetään robotin ohjelmointikieltä. Tämä vaihtoehto ei kuitenkaan aina ole mahdollista, sillä ohjelmointikieli ja sen eri versiot voivat olla puuttelliset. Robottien valmistajat tarjoavat tiedonsiirtoliityntää usein optiona. Tämä tarkoittaa yleensä erillistä tiedonsiirtokorttia (sarja- tai ethernetkortti) ja ohjelmistomoduulia, jotka on asennettu robottiohjaimeen. Tietokone-liitynnällä toteutettu kommunikointi on taustalla toimivaa, itsenäistä, eikä se ole riippuvainen robotin työkierto-ohjelmasta. Tiedonsiirtoyhteys toimii vaikka robotti olisi pysähdyksissä./9, s. 48-50/

Tiedonsiirtoliitynnällä toteutetaan yleensä seuraavia toimintoja:

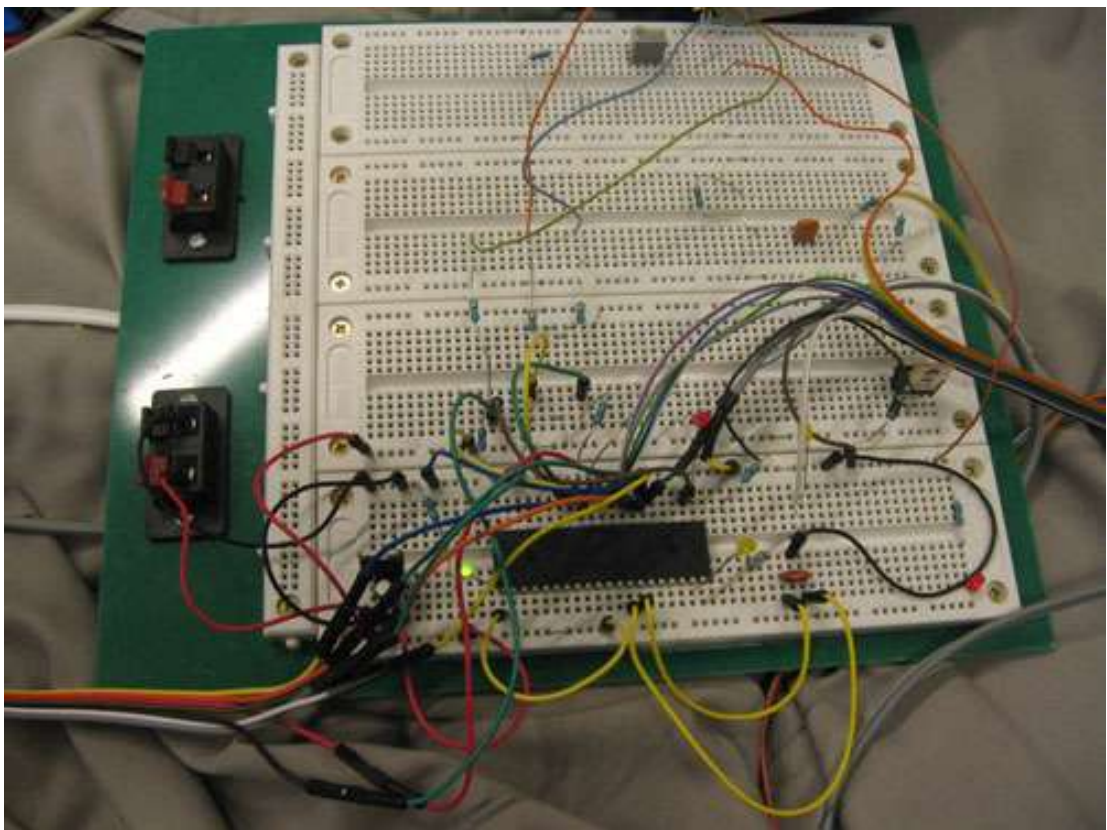
- ohjelmien lataukset robottiohjaimeen
- ohjelmien käynnistys ja pysäytys
- tulojen ja lähtöjen luku- ja kirjoitusoperaatiot
- rekisteritietojen luku ja kirjoitus
- virhetilannekoodien siirto tietokoneelle
- robotin toimitilan monitorointi ja asetus

Tiedonsiirtoprotokollat vaihtelet valmistajakohtaisesti. Robotin tiedonsiirrolle ei ole yhteistä standardia, vaan valmistajat käyttävät oman mieltymyksensä mukaan markkinoilla olevia liityntöjä. Nykyisin ethernet-pohjainen tiedonsiirto on yleistynyt kaikkialla, joten robottien valmistajat ovat siirtyneet käyttämään sitä. Uusissa

tietokoneissa ei ole enää sarjaportteja, vaan ne on korvattu USB-liitännöillä, joita robottien valmistajat eivät vielä käytä. Lisäksi tiedonsiirron määrä on kasvanut ja järjestelmät ovat tulleet reaaliaikaisemmiksi, jolloin tiedonsiirron nopeudelle asetetaan suurempia vaatimuksia, joita ei perinteinen sarjamuotoinen tiedonsiirtotäytä./9, s. 48-50/

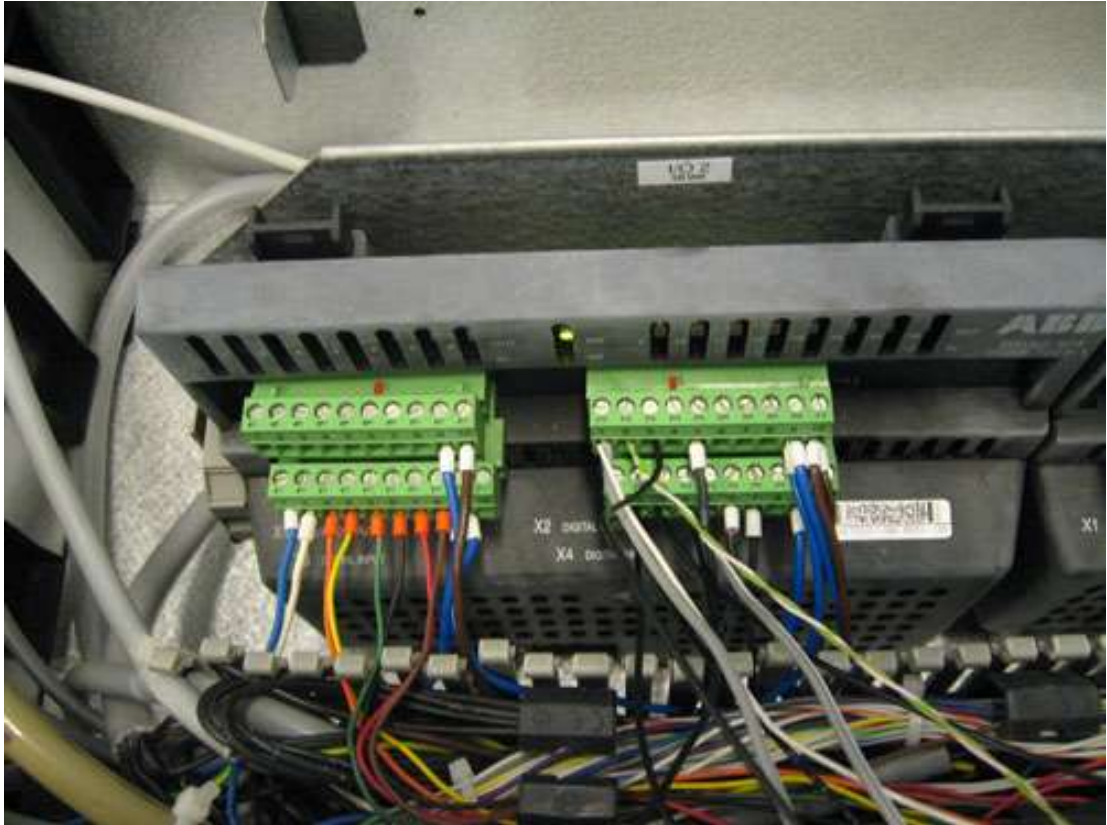
5.2 Projektin tiedonsiirto

Projektissa jouduttiin käyttämään poikkeuksellista tiedonsiirtotapaa, sillä projektin käytössä olleen robottin ohjainjärjestelmän kommunikointi oli toimi puutteellisesti projektin tarpeisiin nähden. S4C-robotiohjaimessa ollut Ethernet-yhteys toimi yhdensuuntaisesti, eikä sitä voitu siitä syystä käyttää. Sarjayhteyskään ei toiminut projektille sopivalla tavalla, sillä se vaati erillisen tietokoneen yhteyttä varten. Projektin tavoitteena yhteyden osalta oli toteuttaa itsenäinen robotin ja kameran välillä toimiva yhteys.



Kuva 5.1 Ohjauselektronikka

Projektin käytössä ollut robotti sai ohjauksen kameroilta sarjamuotoisena datana, joka muutettiin PIC-piirillä neljäksi kolmen bitin Output-tiedoksi, ja kahdeksi Input-tiedoksi robotin I/O-väylään.



Kuva 5.2 Robotin I/O-väylät

Vastaanotettuaan kameralta tulleet I/O-tiedot, robotti kuittasi ohjauselektronikalle jokaisen kolmen bitin sarjan ennen kuin vastaanotti seuraavan kolmen bitin sarjan. Projektin käyttämän robotin IO-listaus kokonaisuudessaan löytyy tämän opinnäytetyön liitteenä (liite 1).

6 OHJELMOINTI

6.1 Robottien ohjelmointi

Ensimmäiset robotit ohjelmoitiin käyttämällä sähkömekaanisia kytkentöjä. Robotin nivelet ajettiin päin haluttuja rajakytkimiä yksi vaihe kerrallaan. Sitten robottia opetettiin johdattamalla nivelten paikka-antureita ja tallentamalla liikkeet robotin muistiin. Nykyisin suurin osa robottisovelluksista on ohjelmoitu opetamalla robotille muutamia paikkoja, mutta toiminnallinen logiikka on ohjelmoitu erikseen tietokoneella. Aivan uusimmat sovellukset on ohjelmoitu kokonaan erillisessä tietokoneessa käyttäen apuna robotista ja sen ympäristöstä luotua kolmiulotteista mallia.

6.1.1 Johdattamalla ohjelmointi

1960-luvulla ensimmäiset robotit palvelivat valukoneita sähkömekaanisten kytkentöjen avulla vaihe kerrallaan sähkömekaanisten logiikoiden ohjauksessa. Myöhemmin robotin käsivarren toimilaitteet vapautettiin ja ihminen pystyi nauhoittamaan robotille liikeradan liikuttamalla robotin työkalua halutun liikeradan mukaisesti. Liikkeiden aikana antureiden lukemat tallennettiin instrumenttinauhuriin. Liikkeitä toistettaessa nauhuri kytkettiin toimilaitteiden säätöpiirien ohjearvoksi. Johdattamalla ohjelmointi levisi robottien valmistien keskuudessa nopeasti ja siitä tuli hallitseva ohjelmoititapa erityisesti maalausroboteissa. Se mahdollisti maalausrobottien nopean yleistymisen, asiaa auttoi myös maalauksen onnistuminen vaikka liikerata ei toistunut aivan samanlaisesti./9, s. 78-82/

6.1.2 Opettamalla ohjelmointi

Nykyään perinteinen robotin ohjelmointitapa on viedä työkalu haluttuun paikkaan käsiohjaimella ja tallentamalla paikka robotin ohjelman muistiin. Liikkumista eri paikkojen välillä ohjataan erilaisilla hyppykäskyillä ja aliohjelmilla. Robotin ohjelmoinnissa käytetään apuna käsiohjainta, editorilla varustettua tavallista PC:tä ohjelman luomiseksi sekä levykeasemia ohjelman tallentamista ja siirtämistä varten. Pelkästään ulkoisen tietokoneen käyttö ohjelmoinnissa ei ole riittävää, sillä robotin ja työympäristön koordinaatit ja 3D-mallit ovat epätarkkoja. Työkalusta riippuen epätarkkuus voi olla useita senttimetrejä. Tästä syystä käsiohjainta käytetään paikkojen tarkkaan määrittämiseen. Mikäli robottisovellus on varustettu paremmilla aistijärjestelmillä, jotka huomioivat todellisen ja mallin väliset erot, voidaan koko ohjelmointiprosessi tehdä erillisessä tietokoneessa.

Ohjelmoinnissa käytettävät kielet muistuttivat aluksi Basic-kieltä, jossa oli lisänä käsivarren liikekäskyjä. Nyt ne ovat lähempänä Pascal-kieltä, jossa on käytössä kaikki ohjelmoinnin rakenteelliset keinot. Ohjelmointikielien ongelmana on niiden valmistajakohtaisuus, eli jokaisella robottivalmistajalla on oma ohjelmointikielensä. Joissakin robottien ohjelmoinnissa käytetyissä kielissä voi olla jopa 150 erilaista käskyä. Käskyjen joukossa on yleensä myös ulkoisia järjestelmiä tukevia käskyjä, jotka ovat yksinkertaisesti ohjelmoitavissa. Lisäksi on luoto tehtävätason ohjelmoitikieliä, mutta niiden kaupallinen hyväksikäyttö on jäänyt vähäiseksi./9, s. 78-82/

6.1.3 Etäohjelmointi

Ohjelmointi- ja etäohjelmointitekniikoista voidaan erottaa mallipohjainen etäohjelmointi, sillä siinä robotti voidaan ohjelmoida ilman tuotantorobottia käyttäen apuna 3D-grafiikkaa robotista, oheislaitteista, työympäristöstä ja käsiteltävästä tuotteesta. Nykyään mallipohjaiset järjestelmät tukevat kymmeniä robottimalleja ja eri CAD-järjestelmiä. Mallipohjaiset järjestelmät perustuvatkin robottien ja oheislaitteiden simulointimalleihin ja tuotteen muototietoihin. Nämä mallinnusjärjestelmät mahdollistavat ohjelmien tarkastamisen ennen robotille

lataamista. Mallipohjaiset ohjelmointijärjestelmät ovat tuotekehityksen ja valmistukseen väliin sijoitettavia työsuunnittelun ohjelmointijärjestelmiä./9, s. 80-82/

6.2 Ohjelman rakenne

Robotin ohjelmointi voidaan tehdä itsenäisesti tietokoneella käyttäen tekstieditoria taikka erillistä ohjelmointityökalua. Robotti voidaan kuitenkin aina ohjelmoida myös käyttäen käsiohjainta ja sen ohjelmointi valikkoja. Ohjelmointi käsiohjaimella on kuitenkin hitaampaa kuin tietokoneella, joten se ei ole kovin yleinen tapa ohjelmoida robottia. Tätä tapaa käytetäänkin enemmän ohjelman tarkistamiseen testauksen aikana ja pienimuotoiseen ohjelman korjaukseen testauksen aikana. Nykyisin on kuitenkin etäohjelmointi yleistynyt robottien ohjelmoinnissa, koska tällöin voidaan robotin ohjelmaa ajaa ja testata virtuaalisesti tuotannon häiriintymättä.

Ennen varsinaisen ohjelmoinnin aloittamista pitää määritellä robotin käyttämä työkalu, työalue ja työkohteet mahdollisimman tarkasti. Itse ohjelmointi kannattaa aloittaa miettimällä ohjelman rakennetta. Itse ohjelma kannattaa jakaa pääohjelmaan ja useisiin aliohjelmiin, ohjelman luettavuuden kannalta. Usein toistuvat toiminnot on järkevää tehdä omiksi ohjelmiksi.

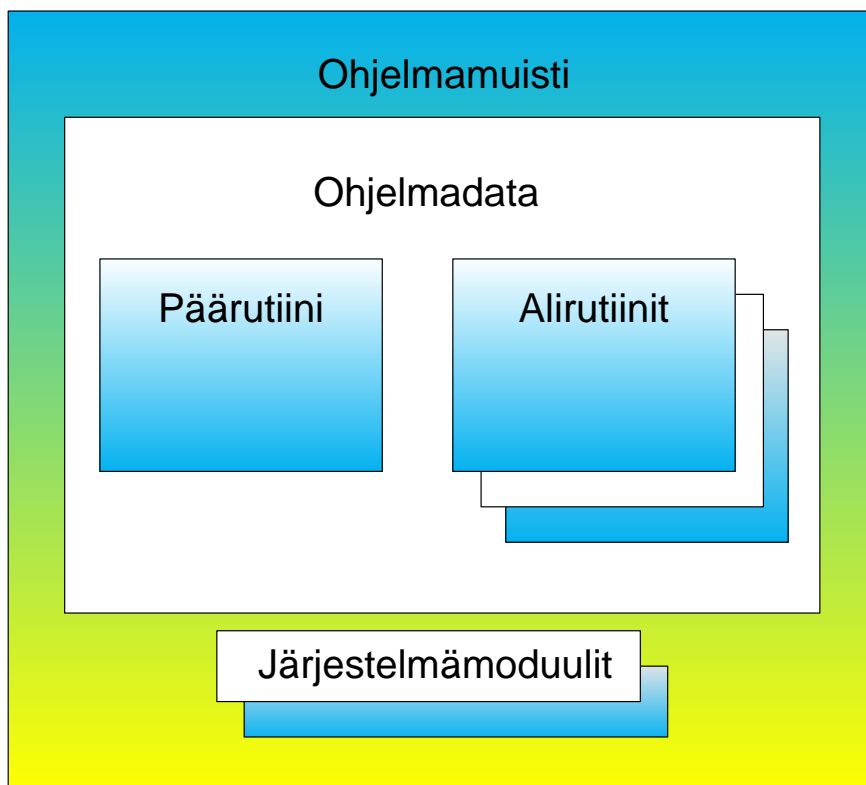
ABB:n yleisrobotille voidaan ladata kahdeksan (8) työohjelmaa eli taskia, joista se käyttää kuitenkin yhtä ohjelmaa kerrallaan työssään. Kun samalla robotilla voidaan helposti tehdä erilaisia työtehtäviä, se helpottaa tuotannon muutosten hallinnassa. Esimerkiksi kun työstetävä kappale muuttuu, ei robottia tarvitse aina ohjelmoida kokonaan uudestaan, vaan voidaan vain ladata haluttu työohjelma robotille./1, 3/

Robotti siis tarvitsee ohjelman toimiakseen tuotannollisesti. Robotin ohjelmisto koostuu pääsääntöisesti neljästä osasta:

- **Päärutiini:** Ohjelman suorittaminen alkaa aina päärutiinista, voidaan pitää myös työohjelman (task) pääohjelmana. Se vastaa robotin yhtä työkierrosta.
- **Useita alirutiineja:** Alirutiineja, eli aliohjelmiä, käytetään ohjelman selkeyttämiseksi ja sillä jaetaan ohjelma pienempiin hallittavampiin osiin. Alirutiineja kutsutaan joko päärutiinista tai jostain toisesta rutiinista.

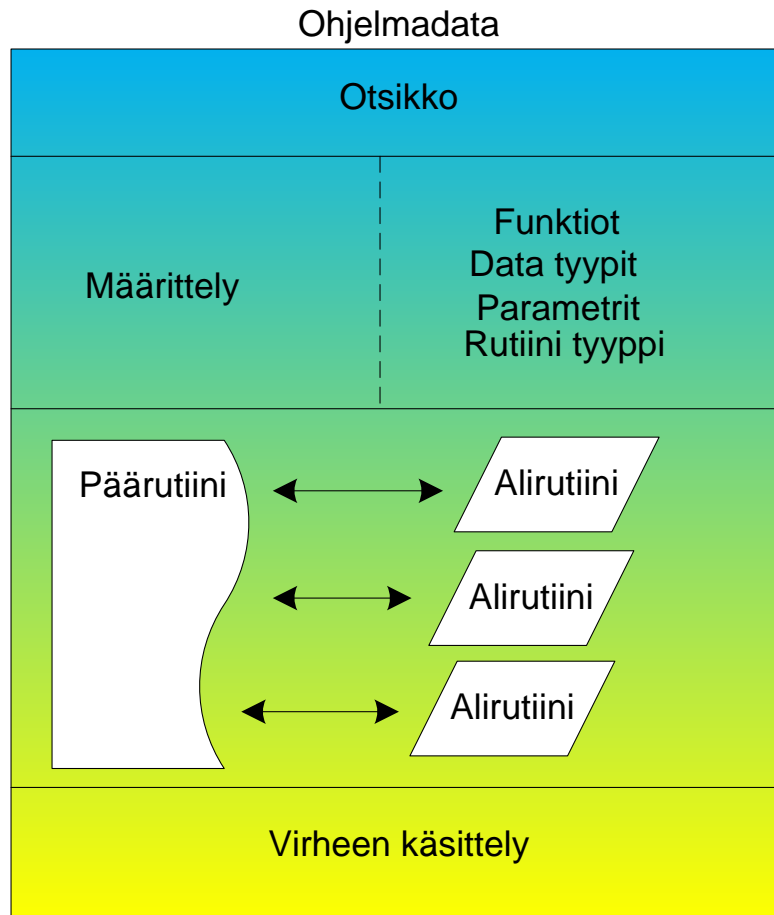
Suoritetaan tehtävänsä loppuun, ohjelma siirtyy takaisin seuraavaan pääruutiinin käskyyn.

- **Ohjelmadata:** Ohjelmadataa käytetään oletusarvojen, funktioiden, koordinaatiojärjestelmien, numeeristen arvojen (rekisterit ja laskurit), sijaintien yms. määrittämiseen. Ohjelmadatojen arvoja voidaan muuttaa itse manuaalisesti, mutta myös ohjelma voi sen tehdä automaattisesti. Datojen arvot voivat muuttua automaattisesti kun tehdään esim. sijainnin uudelleenmäärittäyksessä tai kun päivitetään laskuria.
- **Järjestelmämoduulit:** Järjestelmämoduulissa säilytetään niitä rutiineja ja datatietoja, jotka liittyvät asennuskokonaisuuteen. Järjestelmämoduulit eivät liity mitenkään ohjelmaan, vaan sisältävät työkaluihin ja huoltoihin liittyviä rutiineja.



Kuva 6.1 Robottiohjaimen ohjelmamuistin rakenne/3, s. 135/

Yksi robotin ohjelma (ohjelmadata) koostuu alla olevan kuvan mukaisista osista./1, 2, 3/



Kuva 6.2 Robotin ohjelman rakenne

Otsikkokenttä on yksilöllinen ja riippuu käytettävästä ohjelmointikielestä. Ilman otsikkokenttää järjestelmä ei tunnista ohjelmaa.

Määrittelyosiossa määritellään ohjelmassa käytetyt datatyytit, parametrit ja eri muuttujien alkuarvot, jotka voivat olla staattisia tai ohjelman muutettavissa työkierron aikana.

Pääohjelma(päärutiini) on robotin työkierron ohjelma, johon on määritelty robotin suorittamien toiminnallisuuksien järjestys. Mikäli käytetään paljon aliohjelmaa, pääohjelma voi minimissään sisältää vain suoritettavien aliohjelmien kutsut.

Aliohjelma(alirutiini) on robotin toiminnallinen ohjelma, johon on yleensä ohjelmoitu robotin usein suoritettava selkeä toiminta tai liikesarja, esimerkiksi levyn kääntö työstökoneessa./2, 3/

Lisäksi ohjelmaan on järkevää sijoittaa jonkinasteinen virheen käsittely metodi, jossa määritellään tehtävät toimenpiteet virhetilanteen tapahtuessa.

6.3 Ohjelmakoodi

Robotin ohjelmoimikseksi projektissa käytettiin RAPID-ohjelmointikieltä, joka on ABB:n roboteissa yleisesti käytetty ohjelmointikieli. Projektissa käytetty ohjelmakoodi poikkesi hieman puhtaasti Rapidilla tehdystä koodista, sillä robottiohjaimen asennettu ohjausjärjestelmä tuki vain ohjelmointikielen peruskäskyjä. Osa robotin liikkeistä ja kameran välisestä kommunikoinnista olisi ollut helpompi toteuttaa ohjelmointikielen täysversiolla. Alla on esitetty Rapid-ohjelmointikielen peruskäskyt. Ohjelmointikielen käskyt ja funktiot löytyvät liitteestä (liite 3).

:=	Nimitetään arvo datalle
Compact IF	Toteuttaa käskyn vain jos käskytila on päällä
For	Toistaa jakson niin monta kertaa kun ohjelmoija haluaa
IF	Toteuttaa jakson erilaisia käskyjä riippumatta onko tilat päällä.
MoveC	Työkalupiste liikkuu ympyrämuotoisesti. On robotin perusliikekäsky.
MoveL	Työkalupiste liikkuu lineaarisesti. On robotin perusliikekäsky.
MoveJ	Yhdistetty liike
ProcCall	Kutsu toinen rutiini
Reset	Resetoi digitaalisen ulostulon signaalin nolnaan.
Return	Palaa alkuperäiseen rutiiniin
Set	Asettaa digitaalisen ulostulon signaalin ykköseksi
Wait DI	Odottaa kunnes digitaalinen input on asettunut
WaitDo	Odottaa kunnes digitaalinen output on asettunut
WaitTime	Odottaa ohjelmoijan asettaman ajan tai odottaa robotin pysähtymistä
WaitUnit	Odottaa kunnes tila on asettunut päälle
WHILE	Toistaa jaksoa kunnes tila on vaihtunut.

Tässä kappaleessa käsitellään projektissa käytettyä ohjelmakoodia ja sen osia yleisluontoisesti. Ohjelmakoodi on kokonaisuudessaan tämän opinnäytetyön liitteenä (liite 2).

ENDPROC

Yllä olevassa aliohjelmassa, robotti nousee levyn noutopisteen yläpuolelle, siirtyy kiertopisteen kautta telineelle jättöpisteen yläpuolelle. Sitten robotti laskee levyn alas, katkaisee imun ja odottaa puoli sekuntia tartunnan irtoamisen varmistamiseksi. Sen jälkeen robotti nousee kohtisuoraan ylös jättöpisteen yläpuolelle, siirtyy vapaalla liikkeellä kiertopisteen kautta apupisteeseen. Lopuksi robotti lisää yhdellä telineessä olevien levyjen laskurin lukua.

Projektissa pääohjelma sisälsi digitaalisten lähtöjen nollauksen ja START-aliohjelman, jotka suoritettiin robotin käynnistyessä. Mikäli robotti oli jo käynnissä, pääohjelman kierto sisälsi vain työkiertoon sopivien aliohjelmien kutsuja. Suoritettuaan kaikki ohjelmakäskyt, pääohjelma palaa käskylistan alkukohtaan.

```
PROC main()
  Reset DO10_9;
  Reset DO10_10;
  Reset DO10_11;
  Reset DO10_13;
  Reset DO11_10;
  Reset DO11_11;
  Palautus:=0;
  IF palautus<>2 THEN
    set DO11_9;
    START;
    !WaitDO DO11_1,1;
  ELSE
    alku:
    AJA_ODOTUS;
    !WaitDO DO11_11,1;
    PAIKKA;
    LIIMA;
    AJA_ODOTUS;
    WaitTime 3.0;
    HAE_LEVY;
    VIE_TELINE;
  ENDIF
  GOTO alku;
ENDPROC
```

7 ASENNUS JA HUOLTO

7.1 Huolto ja kunnossapito

Robottijärjestelmän huollossa pyritään ennakoivaan toimintaan. Laitteiston häiriötön toimintakyky pyritään säilyttämään ja samalla saavuttamaan investoinnille maksimituotto. Robotin lisäksi järjestelmän huoltoon kuuluvat järjestelmään kytketyt oheislaitteet. Päivittäisten tarkastusten merkitys on suuri laitteiston toimintakunnon havainnoinnissa. Hitsaussovelluksissa tarkastetaan polttimen kunto, kappaleenkäsittelyssä tarkastetaan kuljettimet ja alipainetarttujen imukuppien kunto./9, s. 128/

Määräaikaishuollot tehdään robottijärjestelmän käyttötuntien perusteella. Näiden huoltojen yhteydessä voidaan ennakoida ja korjata tarvittaessa tulevaisuudessa robotin toimintakykyyn vaikuttavat asiat. Huollot voidaan jakaa pienempiin ja isompiin huoltoihin. Pienemmät huollot, kiristykset ja tarkastukset suoritetaan yleensä muutaman sadan käyttötunnin välein, ja niistä huolehtivat järjestelmän käyttäjät. Isommat, suojarakenteiden avaamista vaativat, huollot tehdään muutaman tuhannen käyttötunnin välein. Isommista huolloista vastaa yleensä robotin maahantuojat, valmistajat tai isommissa laitoksissa tähän erikseen koulutettu huoltohenkilöstö./9, s. 128-129/

Vikatilanteiden korjaus kuuluu olennaisena osana huoltotoimintaan. Robottijärjestelmien yleisemmät viat voidaan jakaa kahteen ryhmään: sähkövikoihin ja mekaanisiin vikoihin. Tyypillisimpiä vikoja ovat sulakkeiden palamiset, antureiden toimimattomuudet ja releiden kärkien hapettumiset. Mekaaniset viat johtuvat yleensä robotin käyttövirheestä aiheutuneesta törmäyksestä./9, s. 129-130/

Robotin mallista ja käyttöenergiasta riippuvat robotin huoltokohteet. Tärkeimmät kohteet ovat voimansiirron ja laakeroinnin osat, sillä niiden kunto ratkaisee voimansiirron käyttöiän./9, s. 130/

Oheislaitteiden huollon tarpeellisuus on riippuvainen laitteesta ja sen käytöstä. Työskentelyolosuhteet vaikuttavat myös huoltojen tarpeeseen: vaikeissa olosuhteissa huoltoväli on pienempi, jotta robotin toimintakyky säilytetään. Tarraimet ovat robottijärjestelmässä tärkeitä osia, niiden vikaantuminen vaikuttaa koko järjestelmään ja estää robotin tuotannollisen käytön. Tästä voi tulla huomattaviakin tuotannon menetyksiä. Sovelluksesta riippuen tarrainten huoltoväliksi suositellaan noin 200 käyttötuntia./9, s. 131-132/

Kustannukset huollosta jakautuvat työn, tarvikkeiden ja osien kesken. Nykyisissä järjestelmissä robotin huollon kokonaiskustannukset ovat alle 5 % robotin hankintahinnasta ensimmäisten kymmenen vuoden aikana. Oikea-aikaiset ja riittävät huoltotoimenpiteet antavat perustan robottijärjestelmän käyttöiälle ja vikaantumistiheydelle. Huoltokustannuksissa säästämällä jättämällä määrääaikaishuollot väliin, robotin käyttöikä lyhennetään nopeasti./9, s. 132/

7.2 Kalibrointi

Robotin kalibrointi käsittää sarjan akselien ja moottorien nollakohtien määrittämiä. Mikäli robotin akseleita ei kalibroida oikein, se vaikuttaa haitallisesti robotin käyttäytymiseen. Kalibroinnin yhteydessä myös moottorien kierroslaskurit nollataan. Kun robotin akselit ja moottorit on kalibroitu oikein, robotti pystyy laskemaan sijaintinsa käynnistyksen yhteydessä./3, s. 214/

Robottijärjestelmä sisältää takaisinkytkennän jokaisesta akselistä mittauspiirille, joka seuraa robotin liikkeitä. Mittauspiiriä käytetään hyväksi määriteltäessä robotin paikkaa sen koordinaatistossa. Mittauspiiri on varmennettu akuilla, jottei robottia tarvitsisi kalibroida uudestaan jokaisen sähkökatkoksen tai robotin sammuttamisen jälkeen.

Mittauspiiri tarvitsee akselin ja kierroslaskurien kalibrointitietoja aina kun jokin manipulaattorin kalibrointipaikka on muutettu. Koko järjestelmä saattaa menettää kalibrointitietonsa kun järjestelmän varmistusakut vaihdetaan, kierroslaskuri aiheuttaa virhetilanteen tai signaali kierroslaskurien ja mittauspiirin välillä häiriintyy./1, s. 392/

Robotin kalibroinnin voi tarkistaa käyttämällä käsiohjainta. Käsiohjaimesta valitaan huoltoikkuna ja valitaan kalibrointi. Kalibrointi-ikkuna näyttää käyttäjälle robotin ja siihen mahdollisesti kytkettyjen laitteiden tilan.

Robotin kalibroinnin tila voi olla jokin seuraavista:

- Synkronoitu

Kaikki akselit ovat kalibroitu ja niiden paikka on tiedossa. Yksikkö on käyttövalmis.

- Kierroslaskuri ei päivitetty

Kaikki akselit on kalibroitu, mutta yksi tai useampi kierroslaskuri päivittämätön.

- Ei kalibroitu

Yksi tai useampi akseleista ei ole kalibroitu.

- Synkronoimaton

Vähintään yhden akselin paikka ei ole tiedossa. /3, s. 215/

Robotin kalibrointi tapahtuu kahdessa vaiheessa:

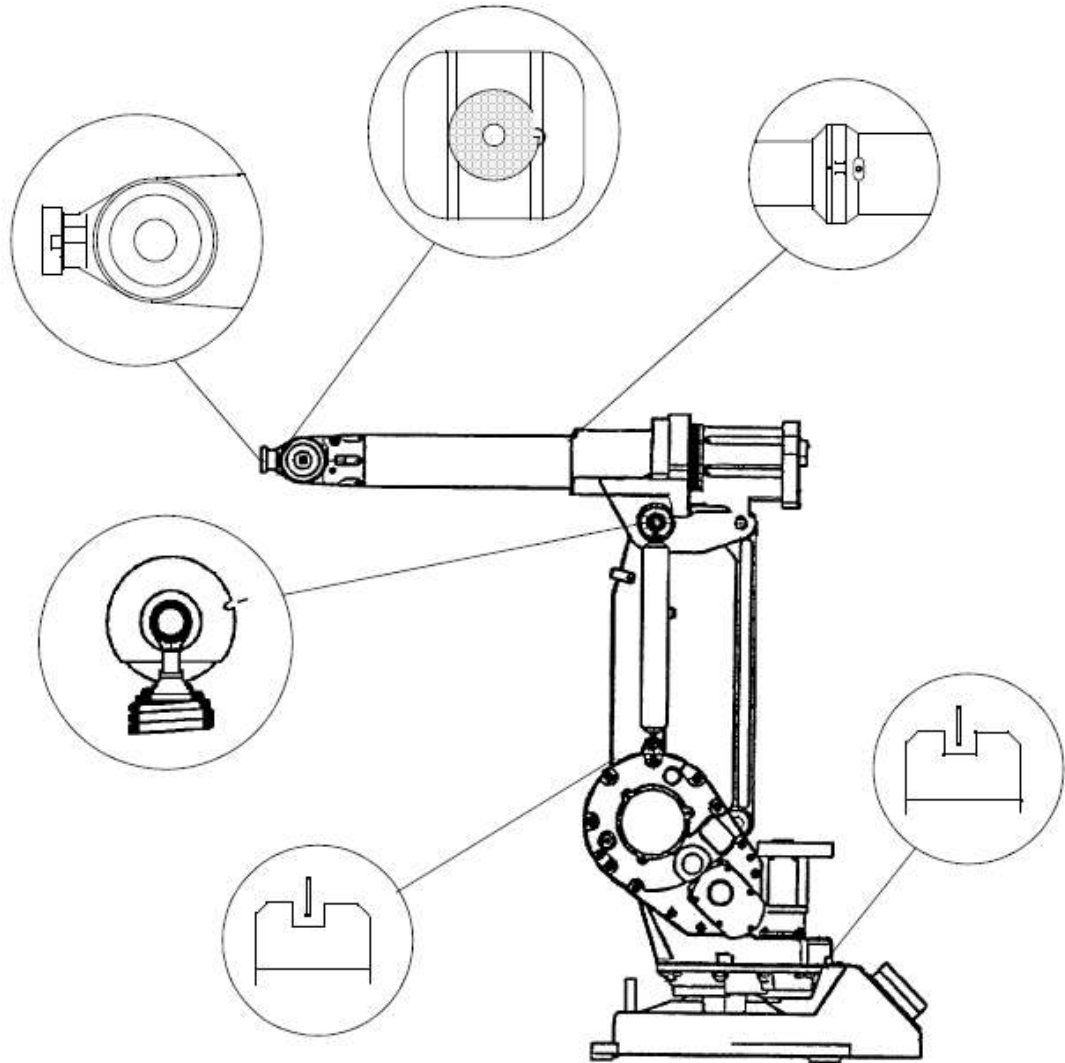
-Robotin akselit ajetaan määriteltyihin kalibrointiasentoihin ja niiden paikkatieto määritellään nollakohdaksi.

-Moottorien kierroslaskurit päivitetään.

Molemmat menetelmät voidaan tehdä erikseen, mutta yleisin ja helpoin tapa on suorittaa akselin kalibrointi, sillä samalla tapahtuu myös kierroslaskurien nollaus.

Robotin kaikki akselit pitää olla sykronoituna, jotta robotti olisi täysin toimintakuntoinen.

Akselien kalibrointi tapahtuu akseleittain järjestyksessä 1-6:een. Kalibrointi aloitetaan liikuttamalla robotin akselit määriteltyihin kalibrointipisteisiin käsiohjainta käyttäen./1, s.392-394/



Kuva 7.1 ABB yleisrobotin kalibrointi/1, s.211/

Yllä olevassa kuvassa on esitetty kaaviokuvana ABB yleisrobotin kalibrointiasento ja eri akselien kalibrointi merkinnät. Seuraavassa kuvasarjassa on kuvattuna projektissa käytetyn robotin kalibrointi.



Kuva 7.2 Projekin robotti kalibrointiasennossa



Kuva 7.3 Akselin 1 kalibrointipiste



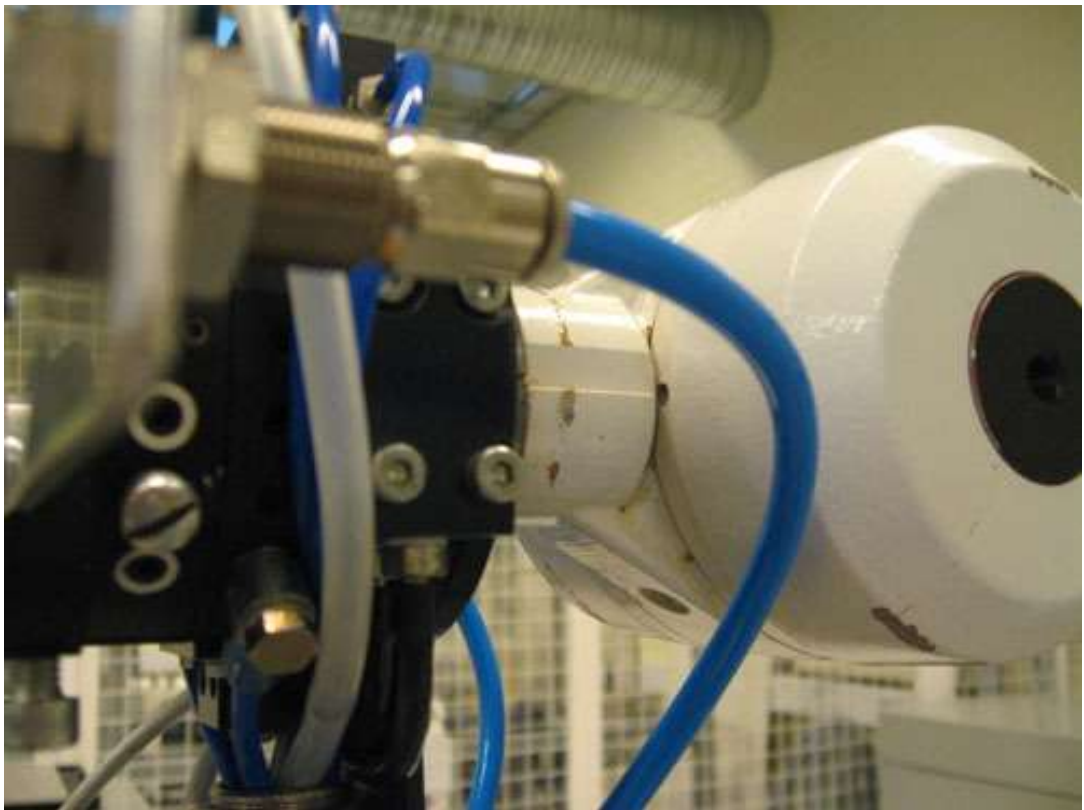
Kuva 7.4 Akselin 2 kalibrointipiste



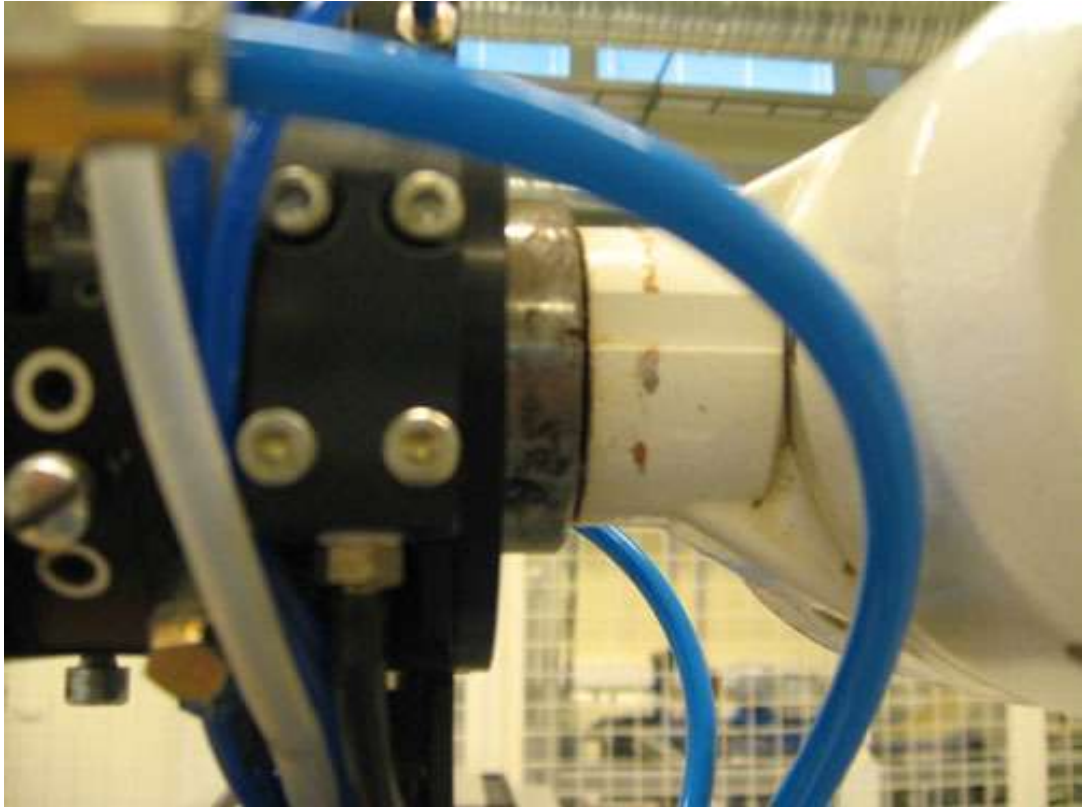
Kuva 7.5 Akselin 3 kalibrointipiste



Kuva 7.6 Akselin 4 kalibrointipiste

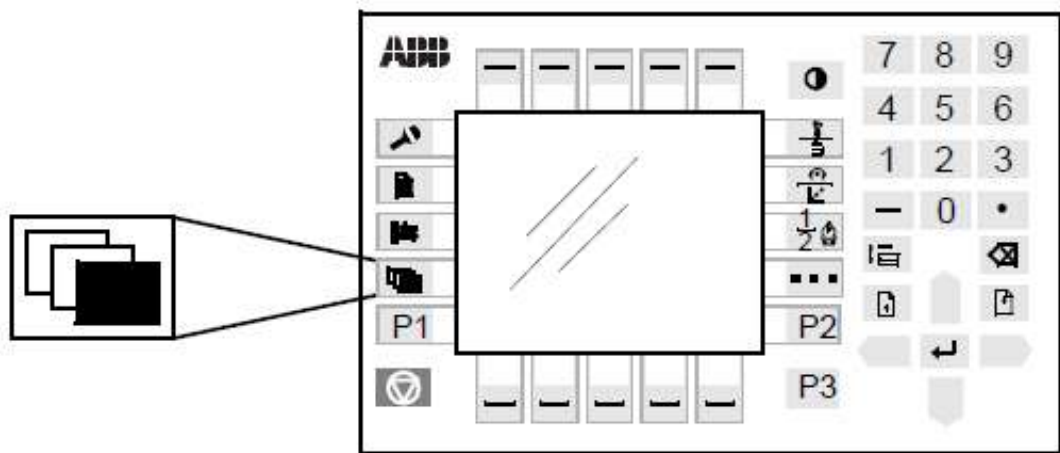


Kuva 7.7 Akselin 5 kalibrointipiste



Kuva 7.8 Akselin 6 kalibrointipiste

Tämän jälkeen valitaan huoltoikkuna kalibroinnin suorittamiseksi.



Kuva 7.9 Käsiohjaimen huoltoikkuna/1, s. 208/

Ikkuna näyttää robotin kalibroinnin tilan. Valittaessa Näytä (View)-valikosta kalibrointi, järjestelmä näyttää yksittäisten akselien kalibroinnin. Kalibrointi suoritetaan valitsemalla akselit joko yksittäin tai kaikki kerralla. Valinnan jälkeen järjestelmä haluaa varmistuksen kalibroinnista, koska alkuperäiset mittaus- ja

akselien paikkatiedot nollautuvat. Kun robotti on kalibroitu, se on käyttövalmis ohjelmaa varten./1, s. 392-394/

File	Edit	View	Com
Service Commutation			
Mech Unit		Status	
			1(4)
Robot		Not Calibrated	

Kuva 7.10 Kalibroimaton robotti/1, s. 393/

Kun akselit on valittu, yksittäin tai kaikki kerralla, voidaan kalibrointi suorittaa valitsemalla käsiohjaimen valikosta kalibrointi. Vaihtoehtoina on kalibrointi tai kierroslaskurien päivitys. Kalibroinnin aikana tapahtuu kierroslaskurien päivitys, joten sitä ei tarvitse suorittaa uudelleen.

Calibration!			
Robot			
To calibrate, include axes and press OK.			
Axis		Status	
			1 (6)
X	1	Not Fine Calibrated	
X	2	Not Fine Calibrated	
	3	Fine Calibrated	
	4	Fine Calibrated	
X	5	Not Fine Calibrated	
X	6	Not Fine Calibrated	
Incl	All	Cancel	OK

Kuva 7.11 Robotin akselit osittain kalibroitu/1, s. 393/

Kun robotin akselit on kalibroitu, muuttuu akselien tila huoltoikkunassa, kuten yllä olevasta kuvasta näkyy. Robotin tila muuttuu synkronoiduksi, kun moottorien kieroslaskurit on nollautuneet ja tahdistuneet akselin kalibroinnin kanssa. Tämän jälkeen robotti on täysin toimintakykyinen.

File	Edit	View	Calib
Service Calibration			
Unit		Status	
		1 (4)	
Robot	Synchronized		
Manip1	Synchronized		
Manip2	Synchronized		
Trackm	Synchronized		

Kuva 7.12 Robotin akselit synkronoituna/3, s. 215/

7.3 Järjestelmäasennus

ABB yleisrobotin muisti on varmennettu akuilla. Se tarkoittaa, että järjestelmän ohjelmat ja asetukset pysyvät tallessa vaikka järjestelmästä katkeaa virta. Ohjausjärjestelmän asennus tapahtuu käyttämällä asennuslevykeitä./1, s. 267/

Järjestelmälevykkeet

- **Key disk** (yksi levy)

Jokainen robotti tarvitsee yksilöllisen järjestelmän avainlevykeen(key disk), jossa on määritelty robotin tyyppi ja valittu ohjelmiston kokoonpano. Saman robotin eri mallit voivat käyttää samaa avainlevykettä ja lisenssinumeron kanssa.

- **Järjestelmälevyt** (8kpl)

Sisältää robottiohjaimen asennettavat ohjelmistot eri vaihtoehtoineen.

- **Ohjaimen parametit** (yksi levy)

Sisältää määrittysten mukaiset robotin I/O-asetukset sekä kaikki ohjaimen parametrit.

- **Manipulaattorin parametrit** (yksi levy)

Sisältää manipulaattorin tehdasasetukset.

Asennus

Robotin ohjaimen järjestelmäasennuksen voi tehdä myös toiminnassa olevalle robotille. Tällöin on ohjaimen muisti tyhjennettävä kokonaan. Sen voi tyhjentää nopeasti ja helposti irrottamalla ohjaimen akut 10 sekunniksi. Toinen vaihtoehto on suorittaa järjestelmälle kylmäkäynnistys(cold start). Kun muisti on tyhjä, voi asennuksen aloittaa laittamalla järjestelmään virta päälle. Asennus tapahtuu seuraamalla käsiohjaimen antamia ohjeita./1, s. 267-268/

Kun asennus on pyytännyt avainlevykkeen(key disk), se tarkistaa asennettavan kokoonpanon. Tämän jälkeen voidaan valita miten asennus suoritetaan.

Asennustapoja on kolme:

- **Silent** = (hiljainen) Asennus tapahtuu avainlevykkeen(key disk) määritysten mukaisesti. Käyttäjälle ei tule muuta tietoa kuin asennuslevykkeiden vaihdon yhteydessä.
- **Add Opt** =(lisäoptiot) Asennus tapahtuu avainlevykkeen(key disk) mukaan siten, että muita ominaisuuksia on mahdollista lisätä myöhemmin.
- **Query** = (kysely) Asennuksen aikana järjestelmä kysyy käyttäjältä robotin tyyppiä, asennettavaa kieltä ja muita ohjaimen järjestelmään asennettavia parametreja.

Tässä projektissa valittiin asennustavaksi Query, jotta voitiin valita järjestelmään oikeat komponentit. Asennustavan valinnan jälkeen valittiin parametrit liikeradoille. Järjestelmän vaihtoehdot ovat palvelu(service) ja standardi(standard). Projektissa valittiin standardi asennus, koska robotin ei tarvinnut kommunikoida tai käyttää järjestelmän ulkopuolisia koneita ja laitteita kuten työstökoneita.

Asennuksen edetessä valittiin käsiohjaimen käyttökieli(TP language). Käyttökieleksi valittiin suomi, sillä robotin pääasialliset käyttäjät ovat suomenkielisiä opiskelijoita, joiden on helpompi seurata käsiohjaimen toimintoja omalla kielellään.

Seuraavaksi valittiin DC Linkki ja robotin tyyppi. DC Linkiksi valittiin DC0, koska se oli robotin ohjaimen määritysten mukainen. Robotin tyyppiä valittiin IRB1400, se oli käytössä olevan robotin perusmalli perusasennuksella. Muita vaihtoehtoja oli robotin seinä- tai kattoasennus. Ne olisivat vaikuttaneet haitallisesti robotin koordinaatistoon ja sen kautta robotin käyttäytymiseen.

Lopuksi valittiin asennettavaksi RAP kommunikointi(RAP Communication), koska ohjelmoitaessa robottia RAP-ohjelmointikielellä, se mahdollistaa merkkipohjaisen kommunikoinnin järjestelmän muiden laitteiden kanssa.

Robotin ohjaimen järjestelmän asennuksen aikana käytettiin seuraavat asennuslevykkeet: 1, 2, 3, 6, 7 ja 8. Kun järjestelmä oli suorittanut asennuksen, se uudelleen käynnistettiin. Tämän jälkeen ohjaimen järjestelmään ladattiin vielä manipulaattorin ja ohjaimen parametrit. Tämä tapahtui valitsemalla käsiohjaimen **Tiedostot**-valikosta **Lataa**-vaihtoehto. Järjestelmään ladattiin Controller Parameters S4C 3.0 ja Manipulator Parameters –levykkeiden sisältö. Parametrit tallentuivat ohjaimen järjestelmän **SYSPAR**-hakemistoon. Parametrien muutosten jälkeen järjestelmä pitää aina uudelleen käynnistää, jonka jälkeen robotti on kalibrointia vaille käyttövalmis.

Robotin ohjaimen järjestelmäasennuksen jälkeen on suositeltavaa tehdä asennuksesta varmuuskopio, josta sen saa tarvittaessa palautettua ilman koko järjestelmän täydellistä asennusta. Palautettaessa järjestelmä varmuuskopioista, tarvitaan pääkäyttäjän tunnus. Se on tehtaalla määritelty oletuskoodi, joka on ABB:n yleisroboteissa 007./1, 3/

Kylmäkäynnistys

Kylmäkäynnistykseen voi tehdä robottiohjainen järjestelmäasennuksen yhteydessä. Kylmäkäynnistys tyhjentää robottiohjaimen muistin samoin kuin akkujen irroittaminen. Kylmäkäynnistykseen voi suorittaa suoraan käsiohjaimen Huolto-ikkunan kautta. Huolto-ikkunan valikoista löytyy uudelleenkäynnistys. Kun se on valittu, syötetään ohjaimen numeronappuloilla koodi 1 3 4 6 7 9, jolloin yksi funktionappuloista muuttuu **C-Startiksi** (Cold start). Sitä painamalla ohjain suorittaa kylmäkäynnistykseen, joka vie jonkin verran aikaa. Kylmäkäynnistykseen aikana on huomattava, ettei robottiohjaimelle saa tehdä mitään ennenkuin järjestelmä pyytää painamaan jotain nappulaa./1, s.268/

Kielen ja IRB tyyppin vaihto

Käsiohjaimen asennetun kielen voi vaihtaa tarvittaessa myöhemmin. Samoin robotin tyyppin voi muuttaa saman robottiperheen sisällä. Muutokset tapahtuvat samoin kuin

kylmäkäynnistyskin Huolto-ikkunan uudelleenkäynnistyksestä. Muutoksia tehdessä syötettävä koodi on 1 4 7, jolloin yksi funktionappuloista muuttuu **I-Startiksi** (Installation start). Ennen nappulan painamista pitäisi varmistua siitä, että levykeasemassa on järjestelmän asennuslevykeistä numero 3 tai 5, riippuen asennettavasta ohjelmistosta. Tämän jälkeen käsiohjaimeen tulee ohjeita asennuksesta./1, s.269/

7.4 Huoltotoimenpiteet

Robotti on suunniteltu toimimaan vaativissa olosuhteissa mahdollisimman pienellä huollolla. Kaikesta huolimatta määräaikaishuollot ja rutiini tarkastukset on syytä tehdä säännöllisin väliajoin. Tämä pidentää robotin toiminta-aikaa ja minimoi ongelmat jotka vaativat suurempaa huoltoa ja korjausta. Robotin työympäristö olisi tarvittaessa siivottava, jotta välttyttäisiin robotin nivelten ja kaapeloinnin vaurioitumiselta. Myös robottiohjain tarvitsee puhdistusta säännöllisesti varsinkin erityisen pölyisissä työympäristöissä.

Huoltoväli

Robotin huoltotoimenpiteet riippuvat robotin käytön määrästä ja laitteistosta. Käytön aikana tulee robotti tarkastettua silmämääräisesti ja tarvittaessa tehdä korjauksia. Kuitenkin on toimenpiteitä ja tarkastettavia kohteita, joita ei helposti huomaa yleissilmäystä tehdessä.

Suosittelavaa on tarkistaa robottiohjaimen jäähdytyksen suodattimet vähintään puolen vuoden välein. Samoin robotin rungon tukijousien kiinnitykset olisi hyvä rasvata ylimääräisen liikekitkan välttämiseksi.

Vuosittaiseen perushuoltoon kuuluvat nivelten 5 ja 6 rasvaus, kaapeloinnin ja 1. akselin mekaanisen rajoittimen kunnan tarkastus. Rikkoutuneet ja vialliset osat on aina vaihdettava, jotta välttyttäisiin suuremmilta vaurioilta ja mahdolliselta tapaturmalta.

Sekä manipulaattorin mittauspiiri ja robottiohjaimen RAM-muisti ovat akuilla varmennettuja. Akut suositellaan vaihdettavan myös määrääjain, jotta järjestelmä pysyisi toimintakykyisenä. Akkujen heikkeneminen aiheuttaa ongelmia robotin käytössä./1, s. 295/

Mittauspiirin varmuusakku

Mittauspiirin varmuusakku sijaitsee robotin rungossa, kannen alla. Robotti toimitetaan asiakkaalle nikkeli-kadmium akulla. Akun suositeltava maksimi käyttöikä on kolme (3) vuotta. Akku voidaan vaihtaa myös ladattavaan litium-akkuun, jonka kesto on pitempi kuin nikkeli-kadmium-akun. Akut sisältävät yleensä kolme tai kuusi ladattavaa solua, riippuen akun kapasiteetista./1, s. 297-298/



Kuva 7.13 Mittauspiirin varmuusakku (sininen)

Akun kesto on riippuvainen siitä, kuinka usein robotti on ilman virtaa. Alla on arvioitu akun maksimi käyttöikä robotin käyttötavasta riippuen.

Taulukko 7.1 Akkujen vaihtoväli

Käyttötapa

1. Virta pois 4 viikon lomalla
2. Virta pois viikonloppuisin + tapa 1
3. Virta pois öisin + tavat 1 ja 2

Vaihtoväli 3-solua

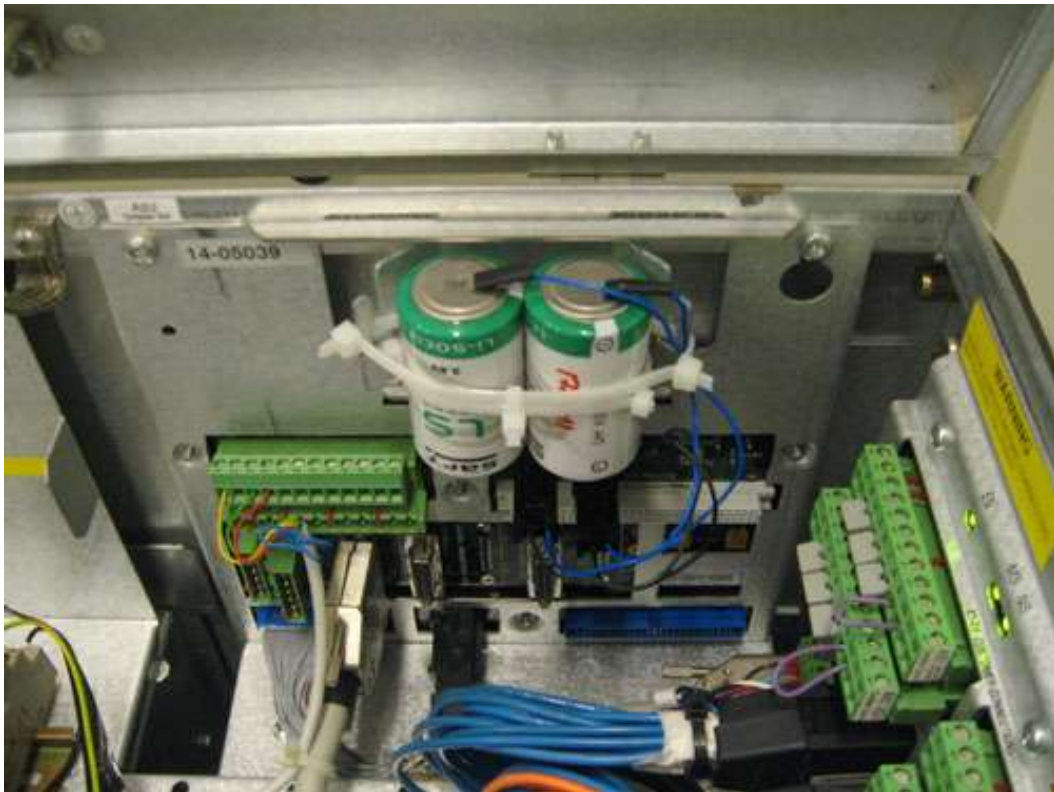
- joka 5 vuosi
joka 2 vuosi
joka vuosi

Vaihtoväli 6-solua

- joka 5 vuosi
joka 4 vuosi
joka 2 vuosi

RAM-muistin varmuusakku

Robottiohjaimen RAM-muisti on varmennettu kahdella litium-akulla, jotka ovat ohjaimen kannen alla. Akkujen maksimi käyttöikä on viisi vuotta, ja niiden kestävyys on riippuvainen asennetun muistin määrästä ja siitä kuinka kauan järjestelmä on ilman virtaa./1, s. 299-300/



Kuva 7.14 Robottiohjaimen RAM-muistin varmuusakut

Alla oleva taulukko kertoo vähimmäisajan minkä akkujen pitää säilyttää järjestelmän muistin ilman ohjaimeen kytkettyä virtaa.

Taulukko 7.2 RAM-muistin kesto ilman virtaa

Muistin koko	1. Akun kesto	Molempien kesto
4 MB	6 kk	12 kk
6 MB	5 kk	10 kk
8 MB	6,5 kk	13 kk
16 MB	5 kk	10 kk

Järjestelmä tarkistaa akut ennen robotin ohjaimen ohjelmiston asennusta ja jokaisen käynnistyksen yhteydessä. Järjestelmä ilmoittaa mikäli jompi kumpi tai molemmat akut ovat tyhjentyneet. Akkujen virran vähentyessä, robotin ohjelmisto saattaa toimia epävakaasti.

8 TURVALLISUUS

8.1 Standardit

Projektin aikana noudatettiin SFS-6002 sähköturvallisuusmääräyksiä, SFS-6001 sähköasennusmääräyksiä ja SFS-12100 koneturvallisuusmääräyksiä. Lisäksi voitiin soveltuvien osien noudattaminen muita koneisiin ja laitteisiin liittyviä määräyksiä.

Projektin aikana tukeuduttiin seuraaviin standardeihin:

Taulukko 8.1 Standardit

SFS ISO 12100-1 + A1	Koneturvallisuus: yleiset suunnitteluperiaatteet osa 1: käsitteet ja menetelmät
SFS ISO 12100-2 +A1	Koneturvallisuus: yleiset suunnitteluperiaatteet osa 1: tekniset periaatteet
SFS ISO 13850	Koneturvallisuus: hätäpysäytys
SFS ISO 6001	Sähköasennusmääräykset
SFS ISO 6002	Sähköturvallisuusmääräykset

8.2 Lait ja direktiivit

Konelaki, eli laki eräiden teknisten laitteiden vaatimustenmukaisuudesta (1016/2004) asettaa vaatimuksia koneiden turvallisuudesta.

Lain tärkeimmät vaatimukset ovat seuraavat:

- Kone on valmistettava ja suunniteltava siten, ettei se aiheuta terveydelle haittaa eikä tapaturman vaaraa.
- Myös myyjän on varmistuttava koneen vaatimusten mukaisuudesta.
- Konedirektiivin vaatimuksia on noudatettava.

-Vaikka kone ei täytä turvallisuusvaatimuksia, se voidaan asettaa näytteille. Tällöin kuitenkin pitää selkeästi merkitä, ettei kone ole vaatimusten mukainen.

Työturvallisuuslain (738/2002) 41§ täydentää konelakia koneiden työturvallisuuden kannalta.

-Työssä saa käyttää vain säännösten mukaisia ja työhön soveltuvia tarkoituksenmukaisia koneita.

-On huolehdittava koneiden oikeasta asennuksesta ja suojalaitteista.

-Koneiden käyttäminen ei saa aiheuttaa haittaa tai vaaraa koneiden käyttäjille eikä muille työpaikalla oleville henkilöille.

-Turvalaitteiden, suojusten ja sijoituksen avulla on rajoitettava pääsyä koneiden vaara-alueille.

Huolto-, korjaus-, häiriö- ja poikkeustilanteisiin on varauduttava, eivätkä ne saa aiheuttaa vaaraa tai haittaa turvallisuudelle eikä terveydelle.

Konedirektiivi eli koneasetus (EY/42/2006)(SF/400/2008) tuli voimaan 29.12.2009. Se on säädös koneiden turvallisuudesta, ja koskee vuoden 1994 jälkeen hankittuja koneita. Säädöksen velvoitteet on suunnattu pääsääntöisesti koneiden valmistajille, mutta myös työturvallisuuslaki ja käyttöasetus velvoittavat työnantajia varmistamaan että koneet ovat konedirektiivin mukaisia./4, 7/

Konedirektiivi koskee periaatteessa kaikki koneita, mutta käytännön syistä sitä ei sovelleta koneisiin joita koskee joku erityisdirektiivi, kuten kulkuneuvot. Myös kotona ja toimistoissa käytettävät pienkoneet ja -laitteet on rajattu soveltamisalueen ulkopuolelle. Lisäksi direktiivi ei koske sotateollisuuden tuotteita./4, 7/

Kone pitää rakentaa siten, että se on käytettävissä, säädettävissä ja huollettavissa henkilöitä vaarantamatta. Myös koneen kokoonpano- ja purkamisvaiheet on turvallistettava niin työtä suorittavien henkilöiden kuin ulkopuolistenkin osalta. Koneen suunnittelun ja turvallisuuden varmistamiseksi tehtävien asioiden perimmäinen tarkoitus on, jokaisen tapaturmavaaran poistaminen ja minimoiminen koneen oletetun elinkaaren aikana.

Konedirektiivin turvallisuustavoitteet vaativat koneen olevan turvallinen koko sen oletetun käyttöajan. Käyttöaikaan katsotaan kuuluvaksi myös koneen rakentaminen ja purkaminen. Direktiivin mukaan vastuu järjestelmästä on työnantajalla järjestelmän elinkaaren ajan. Ei siis ole riittävää, että järjestelmä on tehty ja otettu käyttöön, vaan turvallisuuden on oltava hallinnassa myös käytön ja poistamisen ajan.

Työnantajan pitää huolehtia vähintäänkin seuraavista asioista:

- Turvallisuuteen liittyvän järjestelmän suunnitelmat ovat dokumentoitu ja ovat jäljitettävissä.
- Järjestelmän toimivuus ja luotettavuus on testattava työpaikalla suunnittelijan ohjeiden mukaisesti ja vaatimustenmukaisin väliajoin.
- Korjaukset ja muutokset on suunniteltava ja toteutettava hallitusti.
- Turvatoimintojen instrumentoinnin ja muiden yksityiskohtien tekniset tiedot ja tiedostot ylläpidetään. Korjausten, huollon ja tuotantomuutosten aikana tehtävät muutokset kirjataan.
- Vikaantumiset ja häiriötilanteet sekä niiden korjaustoimenpiteet analysoidaan ja kirjataan.

/8, s. 35-36/

8.3 Turvatoiminnot

Ohjausjärjestelmän, joka liittyy turvallisuuteen, tehtävänä on riskien hallitsemiseksi toteuttaa tiettyjä turvatoimintoja kun järjestelmä saa turvatoimintoja vaativan tiedon antureilta, ohjelmakäskyn tai havaitsee sellaisen muulla tavoin.

Ohjausjärjestelmällä on turvatoimintojen toteuttamista varten seuraavia tehtäviä:

- Yhdistettynä turvalaitteisiin ohjausjärjestelmä pysäyttää koneen ennenkuin vaarakohtaan on mahdollista päästä.
- Turvalaitteisiin ja lukitukseen yhdistettynä ohjausjärjestelmä sallii vaara-alueelle pääsyn vasta koneen pysähtyttyä.
- Vaara-alueella oltaessa ohjausjärjestelmä pitää koneen pysähtyneenä eli estää odottamattoman käynnistyksen.

- Ohjausjärjestelmä estää koneen, sen osan, työkalun tai vastaavan rikkoutumisen sekä siitä aiheutuvat vaarat.
- Tietyissä toimintatiloissa ohjausjärjestelmä pitää koneen nopeuden, liikkeen pituuden, tehon ja vastaavat ominaisuudet alhaisina.
- Tarvittaessa ohjausjärjestelmä sallii koneen käytön vain pakkokäytöllä tai sallintalaitteen avulla.
- Ohjausjärjestelmä tarvittaessa passivoi turvalaitteen.
- Hätätilanteessa pysäyttää koneen nopeasti.
- Ohjausjärjestelmän tulee mahdollistaa puristuksiin tai loukkuun jääneen vapauttamisen.
- Kaatumisnopeuden, kuormituksen tai muun vastaavan turvallisuuden liittyvän muuttujan valvonta ja niiden vaaralliselle alueelle siirtymisen estäminen kuuluvat ohjausjärjestelmän tehtäviin./8, 14/

8.4 Ohjausjärjestelmä

Koneen turvallisuuden kannalta on ohjausjärjestelmän oikealla toteuttamisella suuri merkitys. Ohjausjärjestelmän kautta hoidetaan useimmat turvatoiminnot, kuten suojuksen avaamisen aiheuttama pysähtyminen.

Ohjausjärjestelmille on asetettu eräitä turvallisuuden perusvaatimuksia:

- Ohjelmoitavan logiikan tai vastaavan virheet ja häiriöt eivät saa aiheuttaa vaaratilanteita. Energiansyötön katkeaminen ei saa aiheuttaa vaaroja. Varsinkin
 - -koneen odottamaton käynnistyminen on estettävä
 - -koneen pysäyttäminen ei saa estyä
 - -turvalaitteiden on oltava aina toimintakykyisiä.
- Koneen hallintalaitteet on oltava riittävän etäällä vaarakohdista. Painikkeet ja vivut on suojattava tahattoman vaikutuksen mahdollisuudelta.

- Käynnistyminen saa tapahtua vain vaikuttamalla koneen käynnistyselimeen. Häiriötä kuitattaessa tai sähkön palatessa, kone ei saa lähteä käyntiin.
- Koneessa on oltava normaalia pysäyttämistä varten painike tai vastaava. Käyttöenergian on katkettava koneen pysähdettyä.
- Mikäli kone voi toimia tilapäisesti ilman normaaleja turvatoimia, on turvallisuus varmistettava muulla tavoin, esim. pakkokäyttö ja hitaat liikkeet.
- Koneessa on oltava luotettava energiansyötön katkaisu (pääkytkin, sulkuventtiili tms.) ja se on voitava lukita "erotettu" asentoon.

/7, 8, 14/

8.4.1 Ohjausjärjestelmän rakenne

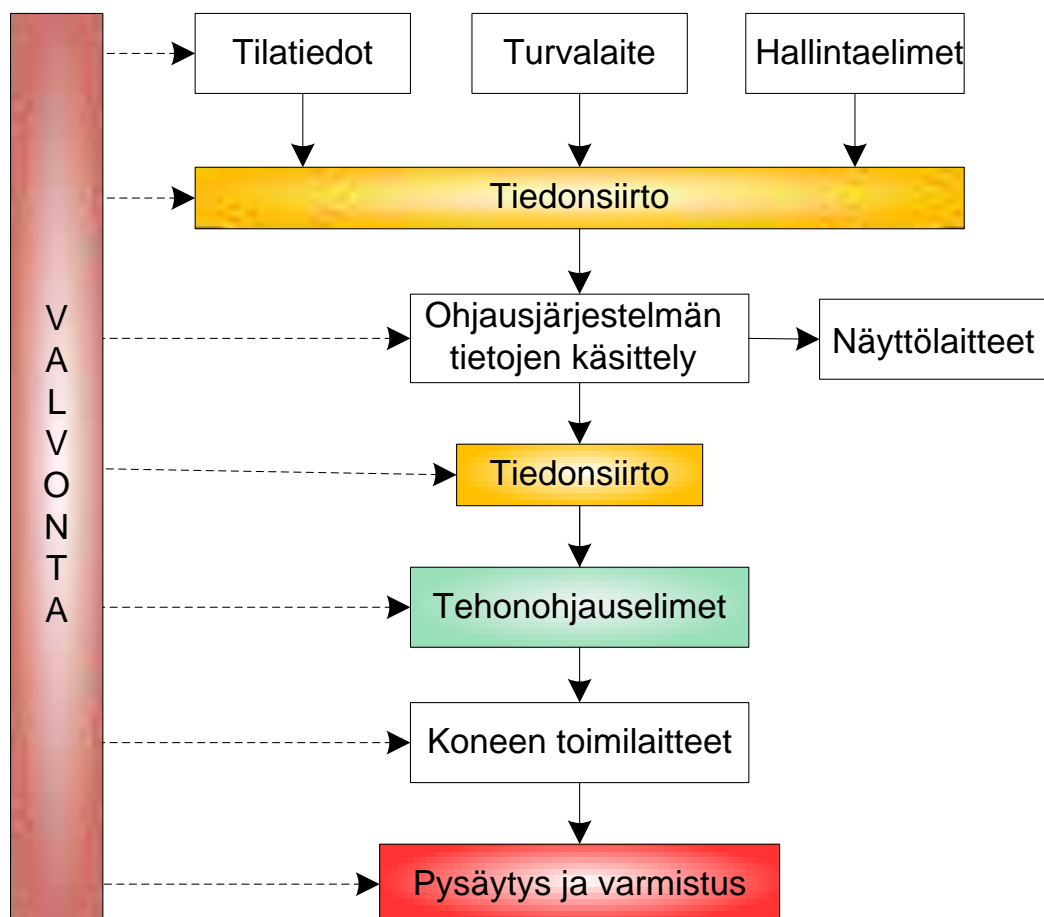
Koneen ohjausjärjestelmä alkaa turvalaitteista, käsin ohjattavista hallintaelimistä sekä antureista. Niillä saadaan aikaiseksi signaaleja, jotka ohjausjärjestelmä käsittelee. Yksinkertaisissa järjestelmissä hallintalaitteet ohjaavat suoraan koneen toimilaitetta, moottoria. Ohjausjärjestelmän käsittelemien anturitietojen perusteella se muodostaa ohjauskäskyt, jotka välittyvät tehonohjauselimiin ja muihin konetta säättäviin laitteisiin. Turvatoimintojen yhteydessä ohjausjärjestelmä suorittaa useimmiten pysäytyskäskyn, jolloin koneen liikkeet pysäytetään ja tehonohjauselimet katkaisevat energian syötön koneen toimilaitteille./8, s.69-72/

Ohjausjärjestelmään kytketyt aukeavat suojat, valoverhot ja muut turvalaitteet ovat ohjausjärjestelmään signaaleja antavia antureita. Myös käsin ja jalalla käytettävät pysäytyspainikkeet ja -polkimet ovat vastaavanlaisia antureita. Lisäksi koneessa on usein myös lämpötilaa, pyörimisnopeutta tai painetta mittaavia antureita.

Tiedon siirtäminen pienissä koneissa on yksinkertaista ja välitöntä. Kytkimen jousen palautuminen 0-asentoon avaa kontaktorin tai vastaavan tehonohjauselimen koskettimet ja kone pysähtyy. Suurissa järjestelmissä siirretään kiinteästi

langoitettuna tai väyliä pitkin satoja metrejä suuria tietomääriä, jolloin tiedonsiirtoon tulee viivettä.

Ohjausjärjestelmän käskyt ohjataan tehonohjaukselle, kuten kontaktoreille ja konevoimalla ohjattaville venttiileille. Tehonohjaukselle säätävät sähkön, paineen tai muun energian pääsyä moottoreille, sylintereille tai muille liikkeen aikaan saaville laitteille. Turvallisuuteen liittyvät käskyt aiheuttavat tehon syötön katkaisemisen. Lisäksi pysähtymiseen ja pysähtyneenä pysymiseen saattaa liittyä jarrujen, salpojen tai muiden mekaanisten laitteiden toiminta. Tehonohjaukselle ja anturit ovat tavallisesti vikaantuvia komponentteja, siksi turvallisuuden varmistamiseksi on pyrittävä parantamaan näiden luotettavuutta ja ohjausjärjestelmä on tehtävä sellaiseksi, että se havaitsee näiden laitteiden vikaantumisen ja pystyy suorittamaan turvatoiminnot viasta huolimatta. Tavallisesti tämä hoidetaan tehonohjaukselle kahdentamisella./8, 14/



Kuva 8.1 Ohjausjärjestelmän rakenne/8/

Koneen toimilaitteilla tarkoitetaan koneturvallisuusstandardeissa moottoreita, sylintereitä tai muita liikkuvia osia. Turvatoimintojen viimeisenä vaiheena on liikkeiden pysäyttäminen ja koneen saattaminen turvalliseen tilaan. Sähkömoottorit pysäytetään usein siten, että moottoria hidastetaan hallitusti tai jarrutetaan sähköisesti, jolloin saadaan nopea ja hallittu pysähtyminen. Liikkeen pysähtyttyä moottoria ohjaava kontaktori avautuu ja moottori on energiansyötöstä erotettu. Hydraulikassa ja paineilmassa venttiilin sulkeutuminen vastaa kontaktorin avautumista. Pysähtymisen jälkeen on varmistettava liikkuvien osien paikallaan pysyminen. Mikäli voimansiirron välitykset eivät riitä, tarvitaan luotettava jarru, mekaaninen salpa tai jokin muu ratkaisu. Monimutkaisemmissa koneissa ja laitteissa on oltava jonkinasteinen vikaantumisen valvonta./4,7, 8, 14/

8.4.2 Ohjauksen tiedonsiirto

Kun turvalaitteen anturiin vaikutetaan, pitää tiedon siirtyä oikein ohjausjärjestelmälle, jonka käskyt pitää välittyä oikeille tehonohjaukselimille. Käytettäessä perinteisiä johdotuksia ja turvareleitä, tieto tulee yleensä oikein perille ja viat paljastuvat. Turvarele paljastaa osan vioista ja osa on taas oikein suunnitellussa järjestelmässä turvallisia; kone pysähtyy kun johdin irtoaa tai katkeaa. Mekaanisissa releissä, sen toiset koskettimet ovat kytketty ohjauspiiriin ja toiset valvontapiiriin. Valvonta edellyttää releiden käyttöä, muutoin ne eivät paljasta vikoja./8, s. 88/

Koneiden ohjausviestit lähetetään sarja- tai rinnakkaismuotoisena. Sarjamuotoisen viestinnän etuna on johdotuksen yksinkertaisuus ja virhekytkentöjen vähäisyys. Sarjamuotoisen viestinnän käyttäminen edellyttää ohjelmoitavaa järjestelmää. Sarjamuotoisessa tiedonsiirrossa turvallisuuteen liittyy muutakin, sillä tieto kulkee usein ei-turvalliseksi luokiteltujen laitteiden kautta, kuten elektroniikka. Tästä syystä on tiedon oikeellisuus varmistettava sekä lähetettäessä että vastaanotettaessa. Elektroniikan tavalliset viat, katkos tai oikosulku, estävät sarjamuotoisen tiedonsiirron, mutta paljastuvat tietoa siirtävissä järjestelmissä./8, s. 89/

Erityisissä turvaväylissä on otettu huomioon riskit ja niissä on riskejä pienentäviä ominaisuuksia. Turvaväylät ovat normaalien standardiväyliä sovelluksia, joissa lisävarmistukset on toteutettu ylätasolla, eivätkä ne vaikuta perusviestintään.

Turvallinen tiedonsiirto edellyttää tiedon oikeellisuutta ja sen siirtoa taatussa vasteajassa. Turvalaitteen ja ohjausjärjestelmän tekemän koneen pysäyttämisen tai muun turvatoiminnon vasteajan on oltava mahdollisimman lyhyt, jotta nopeakin ihmisen vaara-alueelle siirtyminen ei aiheuta vaaraa. Vaara-alueelle johtavassa ovelussa tai suojuksessa on oltava lukitus, mikäli kokonaisvasteaika on sen verran pitkä, että vaaratilanne voisi syntyä./8, 12/

Vasteaika on erityisen tärkeä sellaisissa koneissa ja järjestelmissä, missä ympäristön ja vaara-alueen välissä ei ole fyysistä estettä. Tällaisia turvalaitteita ovat esimerkiksi valoverhot. Tällöin turvalaite on asennettava riittävän etäälle vaara-alueesta, jotta standardin SFS-EN 999 mukaiset vähimmäisetäisyydet täyttyvät suurimmallakin vasteajalla.

Taulukko 9.1 Viivesuosituksien

	Liikkeen ohjaus	Prosessin ohjaus	Käyttöliittymät	Tuotannon hallinta
Viive tai päivitysväli	alle 1 ms	100 ms	alle 1 s	1 s
Viiveen vaihtelu	1 ms	5 ms	ennakoitava	

Tiedonsiirron aika voi vaihdella huomattavasti kaikissa väylä- ja verkkotyypeissä kuormituksesta riippuen. Kuitenkin on taattava, että turvallisuuteen liittyvä tieto on siirrettävä vaaditussa vasteajassa. Ohjelmoitavassa järjestelmässä tiedon käsittelyyn kuluva aika voi vaihdella 5-100 ms välillä. 100 ms lisäys merkitsee esimerkiksi valoverhoilla varustetussa järjestelmässä, että valoverhot pitää asentaa 200 mm kauemmas vaara-alueesta./8, 12/

8.5 Robotin käynnistyminen ja pysähtyminen

Ennen virran kytkemistä robottiin on tärkeää tarkistaa, ettei robotin työalueella ole ketään.

Käynnistettäessä robotin pääkytkimestä, robotti tarkistaa automaattisesti itsensä ja siihen kytketyt laitteet. Tämän jälkeen, jos virheitä ei ole löytynyt, robotti menee valmiustilaan. Mikäli robotti on automaattitilassa, se jatkaa toimintaansa siitä kohdasta missä se oli aikaisemmin. Yleensä kuitenkin ohjelma aloitetaan alusta jolloin robotti ajaa itsensä ohjelmoituun kohtaan ja jää odottamaan erillistä käynnistyskäskyä. Mekaaniset yksiköt aktivoidaan ja IO-tilat muuttuvat ohjelman määritysten mukaisiksi. Mikäli käynnistysvaiheessa ilmenee virhetilanteita (ohjelma-, järjestelmä- tai mekaanisia virheitä) joista robotti saa tiedon, robotti ei käynnisty kunnolla ja se antaa virheilmoituksen käsiohjaimeen ja se kirjautuu myös lokitietoihin./8, 14/

8.5.1 Käynnistysvaroitukset

"(Käyttäjän on pääkäyttöpaikalta pystyttävä varmistamaan, että vaaravyöhykkeellä ei ole alttiina olevia henkilöitä.) Jos tämä ei ole mahdollista, hallintajärjestelmän on oltava siten suunniteltu ja rakennettu, että ääni- tai valomerkki tai molemmat annetaan aina, kun kone on käynnistymässä. Alttiina olevalla henkilöllä, on oltava aikaa ja mahdollisuus nopeasti estää koneen käynnistyminen."

Konedirektiivi

Isojen koneiden tai konejärjestelmien käynnistyspaikasta ei aina ole mahdollisuutta nähdä koko vaara-alueelle. Suuret konejärjestelmät ja -linjat suojataan yleensä turvaovilla varustetuilla suoja-aidoilla. Alueelta poistuttaessa vaaditaan kuittaus, jotta kone voidaan käynnistää pääkäyttöpaikalta. Tällä varmistetaan, ettei vaara-alueelle ole jäänyt ketään. Käynnistysvaroitukset lisävarotoimenpiteenä on paikallaan, koska usein kuittauksesta tulee rutiini toimenpide, eikä täydellistä varmuutta vaara-alueen tyhjyydestä ole. Varoituksen kuulumisen tai näkymisen jälkeen on oltava riittävästi aikaa poistua vaara-alueelta tai estää käynnistyminen. Hälytys tarvitaan vain automaattiajon käynnistyessä, kun automaattisen koneen vaara-alue on hyvin suojattu koneen rakenteen, suojusten ja turvalaitteiden avulla./4, 8/

8.5.2 Käynnistys

Laitteen käynnistyminen saa olla mahdollista vain tietoisesti vaikuttamalla käynnistuksen suorittavaan hallintaelimeen. Tämä koskee myös minkä tahansa syyn takia tapahtuneen pysähtymisen jälkeistä käynnistymistä. Poikkeuksena on kuitenkin turvalaitteilla suojatuilla alueilla tapahtuvat automatiikan suorittaman työkierron aikaiset pysähtymiset ja käynnistymiset./14, s. 38/

Yleisesti koneiden vaarakohtien pitää olla suojattuja siten, ettei käynnistymisestä aiheudu vaaraa. Suuret koneyhdistelmät, robotit ja konelinjat ovat yleensä eroteltu turva-aidalla, siten että jos joku on vaara-alueella, kone ei käynnisty. Käyttäjän pitää voida varmistua siitä, että vaara-alueella ei ole ketään. Jos tämä ei ole mahdollista, on laitteiston annettava selkeä varoitussignaali ennen koneen käynnistymistä. Varoitussignaali voi olla ääni- tai valomerkki, usein käytetään molempia tehostamaan vaaratilanteen mahdollisuutta. Projektissa tämä otettiin huomioon laittamalla ohjelmaan robotin käynnistymisestä varoittava äänimerkki.

```
PROC START()
  Reset DO11_11;
  !WaitTime 3;
  Set DO10_8;
  Set DO10_5;
  !Summeri soi
  WaitTime 0.5;
  Reset DO10_8;
  Set DO10_8;
  WaitTime 0.5;
  Reset DO10_8;
ENDPROC
```

8.5.3 Pysäytys

Jokaisessa koneessa on oltava täydellisen ja turvallisen pysäytyksen tekevä hallintalaite. Kun kone on pysähtynyt, pitää myös energiansyötön koneelle katketa. Täydellinen ja turvallinen pysähtyminen tarkoittaa, ettei kone tai mikään sen osa saa lähteä itsenäisesti liikkeelle painovoiman vaikutuksesta tai muusta vastaavasta syystä. Koneessa on oltava myös tavallisen pysäytyksen ja turvalaitteiden aiheuttamien pysäytysten lisäksi hätäpysäytystoiminto./7, s. 52-53/

8.6 Eristäminen aidalla

Vaara-alueen eristäminen aidalla on yksinkertainen, luotettava ja tehokas ratkaisu. Vaara-alueelle pääsee vain aidassa olevien ovien kautta, jotka ovat kytketty koneen toimintaan. Mikäli vaara-alueelle joudutaan menemään toistuvasti, voidaan ovet korvata valopuomeilla. Verkkoaitaa käytettäessä on alueelle kohtalainen näkyvyys ja koneen toimintaa voidaan seurata. Mikäli alueelle tarvitaan parempaa näkyvyyttä, voidaan verkkoaita osittain tai kokonaan korvata läpinäkyvällä muovilla. Käyttöympäristö voi kuitenkin rajoittaa muovin käyttöä, sillä se on herkkä pölyyntymään ja likaantumaan.

Turva-aidat kiinnitetään alaosaan ja ovat kohtalaisen korkeita. Jotta aita olisi tukeva, pitää kiinnitysten ja pystytukien olla riittävät. Aitaan ei saa jäädä sellaisia kohtia, joista voisi ulottua vaara-alueelle, lisäksi aitaan kiinnitetyt turvalaitteet vaativat riittävää jäykkyyttä ja paikallaan pysymistä.

Konekohtaisesti saattaa olla erityisiä turvamääräyksiä turva-aidan tai muiden turvalaitteiden osalta. Esimerkiksi robotin tarttujasta kesken liikkeen irtoava kappale ei saa päästä lentämään turva-aidan ylitse./4, 7, 8, 12, 14/

8.7 Robottisolu

Robottisolu on tyypillinen aidattu ja turvalaitteilla eristetty alue, jossa pitää tarvittaessa työskennellä koneen ollessa käynnissä. Robotin ohjelmointi tehdään usein käsiohjaimella. Robotti ohjataan käsin tiettyyn pisteeseen ja se tallennetaan robotin ohjelman muistiin. Samalla menetelmällä opetetaan robotille sen työkalujen liikkeet ja muut toiminnot. Tarkkuusvaatimusten vuoksi, opetustyö joudutan tekemään usein erittäin lähellä robottia. Käsiohjausta tarvitaan usein myös häiriötilanteissa./7, s. 185/

Robottistandardissa (SFS-EN ISO 10218-1) on esitetty yksityiskohtaiset vaatimukset robotin käsiohjelmoinnin turvallisuudelle. Robotin käsiohjelmoitavaihe pyritään saada mahdollisimman turvalliseksi rajoittamalla robotin liikkeitä hitaiksi (alle 250mm/s), pakkokäyttöisillä hallintaelimillä ja sallintakytkimellä./7, 14/

Kun valintakytkin on siirretty automaattiajosta käsiajolle, voidaan robotin työalueelle mennä ilman, että rajakytkimen tai valokennojen laukeaminen pysäyttäisivät robotin toiminnan. Tällöin on mahdollista ohjata robottia ja robottisolun muita laitteita vain käsiohjaimella. Lisäksi käsiohjauslaitteessa on oltava erillinen ohjelmistosta riippumaton sallintakytkin jolla varmistetaan, etteivät ohjelmistossa olevat virheet aiheuta vaaratilanteita./7, 14/

8.8 Pneumatiikka

Pneumaattisten järjestelmien yleiset vaatimukset on määritelty standardissa SFS-EN 983. Pneumaattisen järjestelmän kaikkiin komponentit pitää valita siten, että ne ovat käytön aikana turvallisia ja toimivat suunnitellusti. Sellaiset komponentit, jotka vikaantuessaan voivat aiheuttaa vaaraa, tarvitsevat erityistä huomiota. Järjestelmän suunnittelussa on otettava huomioon myös järjestelmän tai sen osan maksimityöpaineen tai komponentille määritellyn nimellispaineen ylittyminen. Ylipainetta voidaan rajoittaa, siten että käytetään paineenrajoitusventtiilejä, jotka on asennettu järjestelmään siten, että ne rajoittavat koko järjestelmän painetta./11, s. 18/

Ennakoitavissa olevat toiminnot, häiriötilanteet ja viat eivät saa missään olosuhteissa aiheuttaa ihmisille vaaraa. Ylipaineen lisäksi järjestelmää suunniteltaessa on otettava huomioon myös:

- heilahduspaine
- painevahvistus
- paineen äkillinen aleneminen tai häviäminen kokonaan
- järjestelmän sisäiset tai ulkoiset vuodot
- syötön katkeaminen tai palautuminen katkon jälkeen
- syötön kytkeminen päälle tai pois.

Lyhyt kestoisia vaara-alueella käyntejä varten voidaan paineilmajärjestelmän syötöstä erottaminen hoitaa normaalisti ohjausjärjestelmän sulkemilla venttiileillä. Pitkäkestoisia syötöstä erottamisia varten, on järjestelmässä oltava käsikäyttöinen, lukittavissa oleva sulkuventtiili./7, s. 319-321

Venttiilin sulkeutumisen lisäksi on paineilmajärjestelmissä tärkeää, että järjestelmässä oleva paine purkautuu. Järjestelmään jäänyt paineilma voi aiheuttaa vaarallisia liikkeitä, vaikka järjestelmä olisikin erotettu paineilmaverkosta. Syötöstä erottamisen pitäisi samalla purkaa paineen järjestelmän kaikista osista turvallisesti. Mikäli tämä ei ole mahdollista, on vastaava turvallisuus varmistettava muulla tavoin.
/7, 11/

8.9 Asentaminen ja käyttöönotto

Asentaminen on yksi vaarallisimpia työvaiheita varsinkin suurissa koneissa ja konelinjoissa. Asennuspaikalla on usein paljon ihmisiä useista eri yrityksistä ja työt ovat eri vaiheissa. Tapaturmien välttämiseksi on asennusvaiheen turvallisuutta myös mietittävä. Isoissa laitoksissa on huolehdittava turvallisten kulkuväylien, portaiden ja tasojen olemassaolosta jo asennusvaiheen alkaessa. Asennus- ja käyttöönottovaiheissa on sattunut vakavia tapaturmia, jotka ovat aiheutuneet puutteellisesti suojattujen koneiden odottamattomasta käynnistymisestä.

Huoltojen ja korjausten tarve on otettava huomioon työtasoissa, valaistuksessa ja muussa koneen ympäristössä. Koneiden on oltava vakaita ja turvallisia myös kun niistä huoltojen ja korjausten yhteydessä mahdollisesti irroitetaan osia.

Myös vikojen ja niiden etsimisen yhteydessä saattaa tapahtua tapaturmia, siksi koneturvallisuusasetuksen mukaan, kone pitää varustaa erityisellä vikadiagnostiikalla tai liitännämahdollisuudella siihen./7, s. 335-336/

Asennusvirheiden minimoimiseksi voidaan koneen suunnitteluvaiheessa osien muotoilulla ja suunnittelulla varmistaa, että osat voidaan kiinnittää koneeseen miten

päin tahansa tai vain yhdellä tavalla. Oikea asentaminen on pyrittävä varmistamaan selkeiden merkintöjen ja ohjeiden avulla.

Mekaanisten osien oikean asennuksen ja liikesuunnan lisäksi on varmistettava irrotettavien neste- tai kaasuletkujen sekä sähkö- ja tiedonsiirtokaapeleiden oikea liittäminen suunnittelun ja rakenneratkaisujen avulla. Liitospäälaitteiden suositellaan olevan niin erilaisia, että vain oikeiden liitosten ja kytkentöjen tekeminen on mahdollista./7, s. 337/

9 JATKOKEHITYS

Projekti oli tarkoitettu kertaluonteiseksi konenäköjärjestelmän demonstraatioksi case-yrityksenä toimineelle Puustelli Oy:lle. Robotin ohjelmaa voidaan kuitenkin hyödyntää ja muokata oppilaitoksen omiin tarkoituksiin, esimerkiksi oppilaiden harjoitustöihin.

Robottiohjaimen järjestelmän päivitys olisi hyvä tehdä. Robotin ohjelmistoa voisi helposti kehittää yksinkertaisemmaksi ja vakaammaksi. Lisäksi ohjelmaan voisi lisätä aliohjelma levyn kääntämiseksi työstövaiheen aikana.

```
PROC kaanto()
  MoveL Offs(liimaus,0,0,100),v1000,fine,tool0;
  IndAMove Robot,6\ToAbsNum:=180,20;
  MoveL Offs(liimaus,0,0,0),v1000,fine,tool0;
  WaitTime 0.5;
  MoveL Offs(liimaus,0,0,250),v1000,fine,tool0;
  IndAMove Robot,6\ToAbsNum:=180,20;
ENDPROC
```

Turvallisuutta voisi lisätä robotin liikkeiden hallitulla keskeyttämisellä ja jatkamisella. Robotti pysähtyisi turva-alueen rajakytkimen tai muun vastaavan aktivoituessa ja jatkaisi liikettä vasta kun saa luvan.

```
StopMove;
WaitDI ready_input,1;
StartMove;
```

Lisäksi sarja- ja Ethernet-tiedonsiirron saaminen robottijärjestelmään kehittäisi robotin toimintamahdollisuuksia eteenpäin ja monipuolisemmaksi.

Projektin varsinainen kehitystyö jää case-yrityksen vastuulle, sillä demonstration tarkoituksena oli esitellä heille robotti- ja konenäköjärjestelmän mahdollisuuksia levynkäsittelyssä. Puustelli Oy tulee tekemään omat päätöksensä levyjen työstöjärjestelmän kehittämisestä demonstration aikana tulleiden kehitysideoiden, havaintojen ja suositusten perusteella.

LÄHTEET

1. ABB Flexible Automation AB. Product Manual IRB 1400 M97A. Document 3HAC 2914-1 M98
2. ABB Flexible Automation AB. RAPID Reference On-line Manual 3.0
3. ABB Flexible Automation AB. User´s guide For BaseWare OS 3.1. Document 3HAC 0966-21
4. EY:42/2006 Euroopan parlamentin ja neuvoston koneidirektiivi eli koneasetus(SF/400/2008).
5. Heinonkoski, R., Asp, R. & Hyppönen, H. 2008. Automaatio-helppoa elämää? Vammalan kirjapaino.
6. Hyvönen, E., 2001. Inhimillinen kone - konemainen ihminen. Yliopistopaino.
7. Siirilä, T. 2008. Koneturvallisuus - EU-määräysten mukainen koneiden turvallisuus. Inspecta. Otavan Kirjapaino Oy. Keuruu.
8. Siirilä, T. 2005. Koneturvallisuus - Ohjausjärjestelmät ja turvalaitteet. Inspecta. Otavan Kirjapaino Oy. Keuruu.
9. Suomen Robotiikkayhdistys ry. 1999. Robotiikka. Tummavuoden Kirjapaino Oy. Vantaa.
10. Suomen Standardoimisliitto. Standardi ISO 8373. Manipulating industrial robots-vocabulary. Suomen standardoimisliitto ry. Helsinki.
11. Suomen Standardoimisliitto. Standardi SFS-ISO 983. Koneturvallisuus-hydraulisten ja pneumaattisten järjestelmien turvallisuusvaatimukset. Suomen standardoimisliitto ry. Helsinki.
12. Suomen Standardoimisliitto. Standardi SFS-EN 999. Koneturvallisuus-Turvalaitteiden sijoitus ottaen huomioon kehon osien lähestymisnopeudet. Suomen standardoimisliitto ry. Helsinki.
13. Suomen Standardoimisliitto. Standardi SFS-EN 9787. Manipulating industrial robots. Coordinate systems and motion nomenclatures. Suomen standardoimisliitto ry. Helsinki.
14. Suomen Standardoimisliitto. Standardi SFS-ISO 10218-1. Teollisuusrobotit-Turvallisuus vaatimukset-robotti. Suomen standardoimisliitto ry. Helsinki.

Internet lähteet:

15. ABB WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011]. Saatavissa:
<http://www.abb.fi/product/fi/9AAC100735.aspx?country=FI>
16. Aibon WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011]. Saatavissa:
<http://www.aibo.com>
17. Arizona Machineryn WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011].
Saatavissa: http://www.arizonam.com/en/scara_robots.php
18. Tankrobotin WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011].
Saatavissa: <http://www.tankpitstop.com>
19. Asimo-Hondan WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011].
Saatavissa: <http://asimo.honda.com/>
20. Actroidin WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011]. Saatavissa:
<http://www.kokoro-dreams.co.jp>
21. Modern Times WWW-sivu [verkkodokumentti].[Viitattu 27.9.2011].
Saatavissa: <http://moviesovermatter.com>

Muita apuna käytettyjä lähteitä:

22. Harjavalta Oy - yrityksen esittely

LIITE 1

ABB yleisrobotin digitaaliset tulot ja lähdöt

DI		DO	
10.1	Kuljettimen 1. anturi	10.1	Kuljetin eteen
10.2	Kuljettimen 2. anturi	10.2	Punainen hätävalo
10.3		10.3	
10.4		10.4	Puhallus
10.5	Vacuumin tunnistus 1	10.5	Työkalun lukitus
10.6	Vacuumin tunnistus 2	10.6	Imu
10.7		10.7	
10.8		10.8	Summeri
10.9		10.9	
10.10		10.10	
10.11		10.11	Kameran 1 paikkatiedon indikointi
10.12		10.12	Kameran 2 paikkatiedon indikointi
10.13		10.13	Sequence acknowledgement
10.14		10.14	
10.15		10.15	
10.16		10.16	
11.1		11.1	
11.2		11.2	
11.3		11.3	
11.4		11.4	
11.5		11.5	
11.6		11.6	
11.7		11.7	
11.8		11.8	
11.9	LSB	11.9	Power
11.10	Bit	11.10	Ack
11.11	MSB	11.11	Sequency Restart
11.12	Bit Ready	11.12	
11.13		11.13	
11.14		11.14	
11.15		11.15	
11.16		11.16	

LIITE 2

```
%%%
VERSION:1
LANGUAGE:ENGLISH
%%%
lohjelman nimi ja määrittelyosio. Robotille määritely eräitä pisteitä koordinaatistossa
MODULE ekatesti
CONST robtarget hakupaikka:=[[-1030.10,547.00,190.60],[1,0,0,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget koti:=[[1000.00,10.00,1150.00],[0.70713,0,-0.70708,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget liimaus:=[[836,-220,160],[1,0,0,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget odotus:=[[628.00,-55.00,500.00],[0.99405,-0.03866,0.02148,0.09959],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget lapu1:=[[626.70,168.60,1000.00],[0,0.70711,0,-0.70711],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget tapu2:=[[618.40,-305.80,882.10],[0.74313,0.44086,-0.44066,-0.24337],[1,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget tapu3:=[[588.60,-1135.20,853.80],[0.60214,0.37649,-0.57293,-0.40916],[1,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget kp:=[[304.60,-607.70,886.10],[0.63818,0.29190,-0.55112,-0.45144],[1,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget jatto2:=[[658.60,-1325.20,803.00],[0,0.70711,0,-0.70711],[1,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

```

lprojektissa käytettävien muuttujien määrittelyt ja parametrit

```
VAR num PoytaX:=0;
VAR num PoytaY:=0;
VAR num PoytaZ:=0;
VAR num Sekvenssi:=0;
VAR num DX:=0;
VAR num DY:=0;
VAR num DZ:=0;
VAR num KunnesImu:=0;
VAR num MuutettuDZ:=0;
VAR num kosinikulma:=0;
VAR num Q1kesken:=0;
VAR num Q4kesken:=0;
VAR num quu1:=1.00000;
VAR num quu4:=0.00000;
VAR num kulma:=500;
VAR num LaskKulm:=0;
VAR num Kulmarajotus:=0;
VAR num TelineesLevyi:=0;
VAR num TelineXD:=0;
VAR num Ykkoset:=0;
VAR num Nollat:=0;
VAR num Nappula:=0;
VAR num Palautus:=0;

```

```
VAR robtarget Ottopaikka;
```

lensimmäinen aliohjelma, tässä määritellään robotille haettavan levyn paikka

```
PROC PAIKKA()
!indikoidaan kameran 1 paikka tieto
set DO10_11;
Ykkoset:=0;
Nollat:=0;
PoytaX:=0;
PoytaY:=0;
PoytaZ:=0;

Sekvenssi:=0;
WHILE Sekvenssi<10 DO
    IF DI11_10<>Nollat THEN
        Set DO10_2;
        Incr Sekvenssi;
        Nollat:=DI11_10;
        InvertDO DO11_10;
    ELSEIF DI11_9<>Ykkoset THEN
        Set DO10_2;

```

```

        Incr Sekvenssi;
        Ykkoset:=DI11_9;
        PoytaX:=PoytaX+Pow(2,(Sekvenssi-1));
        InvertDO DO11_10;
    ENDIF
ENDWHILE
Reset DO10_11;
Sekvenssi:=0;
WHILE Sekvenssi<10 DO
    IF DI11_10<>Nollat THEN
        Incr Sekvenssi;
        Nollat:=DI11_10;
        InvertDO DO11_10;
    ELSEIF DI11_9<>Ykkoset THEN
        Incr Sekvenssi;
        Ykkoset:=DI11_9;
        PoytaY:=PoytaY+Pow(2,(Sekvenssi-1));
        InvertDO DO11_10;
    ENDIF
ENDWHILE
Sekvenssi:=0;
WHILE Sekvenssi<10 DO
    IF DI11_10<>Nollat THEN
        Incr Sekvenssi;
        Nollat:=DI11_10;
        InvertDO DO11_10;
    ELSEIF DI11_9<>Ykkoset THEN
        Incr Sekvenssi;
        Ykkoset:=DI11_9;
        PoytaZ:=PoytaZ+Pow(2,(Sekvenssi-1));
        InvertDO DO11_10;
    ENDIF
ENDWHILE
!sammutetaan punainen valo, joka indikoi saatua datapakettia
Reset DO10_2;

!tulostetaan käsiohjaimen näytölle levyn koordinaattitieto pikseleinä
TPWrite "X = "\Num:=PoytaX;
TPWrite "Y = "\Num:=PoytaY;
TPWrite "Z = "\Num:=PoytaZ;

!muutetaan pikselit millimetreiksi ja sopimaan yhteen robotin koordinaatiston kanssa
DX:=PoytaX*1.8164;
DY:=PoytaY*1.8164;
DZ:=PoytaZ+20;

!robotti siirtyy kamera määrittämään poimintapisteen yläpuolelle ja lähtee laskeutumaan alas
MoveJ Offs(hakupaikka,DX,DY,DZ),v1000,fine,tImu\WObj:=wobj0;
!imu laitetaan päälle, kunnes alipaineanturi tunnistaa paineen
set DO10_6;
kunnslmu:=0;
WHILE kunnslmu<2 do
    kunnslmu:=DI10_5;
    kunnslmu:=kunnslmu+DI10_6;
    Decr DZ;
    MoveL Offs(hakupaikka,DX,DY,DZ),v1000,fine,tImu\WObj:=wobj0;
ENDWHILE

Ottopaikka := CRobT(\Tool:=tImu \WObj:=wobj0);

DZ:=DZ+150;
MoveL Offs(hakupaikka,DX,DY,DZ),v500,z100,tImu\WObj:=wobj0;
ENDPROC

!start-ohjelmassa robotti käynnistyy ensimmäisen kerran tai uudestaan pysäytyksen jälkeen
PROC START()
    Reset DO11_11;
    !WaitTime 3;
    !järjestelmä varoittaa robotin käynnistymisestä kahdella lyhyellä äänimerkillä
    Set DO10_8;
    Set DO10_5;

```

```
WaitTime 0.5;
Reset DO10_8;
Set DO10_8;
WaitTime 0.5;
Reset DO10_8;
!robotin vapaa liike alkupisteeseen
MoveJ Offs(koti,0,0,0),v1000,fine,tImu\WObj:=wobj0;
```

!järjestelmä kysyy kuinka monta levyä telineesä on lähtötilanteessa, laskuri päivittyy vastaamaan annettua lukua
!jos luku on enemmän kuin telineeseen mahtuu, järjestelmä pyytää lukua uudestaan
TPReadNum Telineeslevyi, "Kuinka monta kappaletta telinees on jo?";

```
Uudestaan:
IF Telineeslevyi>9 THEN
    TPReadNum Telineeslevyi, "Virheellinen arvo annettu. Anna uudestaan!";
    Goto Uudestaan;
ENDIF
ENDPROC
```

!aliohjelma joka ajaa robotin odotuspisteeseen
PROC AJA_ODOTUS()
 MoveJ offs(odotus,0,0,0),v1000,fine,tImu\WObj:=wobj0;
ENDPROC

!tässä ohjelmassa robotti vie levyn sivukameralle kuvattavaksi ja odottaa pienen hetken
PROC LIIMA()

```
    MoveJ Offs(lapu1,0,0,0),v1000,fine,tImu\WObj:=wobj0;
    Nollat:=DI11_10;
    Ykkoset:=DI11_9;
    WaitTime 1;
    Set DO10_12;
    WaitTime 1;
    Set DO10_13;
    PoytaX:=0;
    PoytaY:=0;
    PoytaZ:=0;
```

!kommunikointi kameran kanssa ja levyn koon määrittäminen
alku:

```
Sekvenssi:=0;
WHILE Sekvenssi<10 DO
    IF DI11_10<>Nollat THEN
        Set DO10_2;
        Incr Sekvenssi;
        Nollat:=DI11_10;
        InvertDO DO11_10;
    ELSEIF DI11_9<>Ykkoset THEN
        Set DO10_2;
        Incr Sekvenssi;
        Ykkoset:=DI11_9;
        PoytaX:=PoytaX+Pow(2,(Sekvenssi-1));
        InvertDO DO11_10;
    ENDIF
ENDWHILE
Reset DO10_12;
Reset DO10_13;
Sekvenssi:=0;
WHILE Sekvenssi<10 DO
    IF DI11_10<>Nollat THEN
        Incr Sekvenssi;
        Nollat:=DI11_10;
        InvertDO DO11_10;
    ELSEIF DI11_9<>Ykkoset THEN
        Incr Sekvenssi;
        Ykkoset:=DI11_9;
        PoytaY:=PoytaY+Pow(2,(Sekvenssi-1));
        InvertDO DO11_10;
    ENDIF
ENDWHILE
Sekvenssi:=0;
WHILE Sekvenssi<10 DO
```

```

IF DI11_10<>Nollat THEN
    Incr Sekvenssi;
    Nollat:=DI11_10;
    InvertDO DO11_10;
ELSEIF DI11_9<>Ykkoset THEN
    Incr Sekvenssi;
    Ykkoset:=DI11_9;
    PoytaZ:=PoytaZ+Pow(2,(Sekvenssi-1));
    InvertDO DO11_10;
ENDIF
ENDWHILE

Reset DO10_2;

!tulostetaan käsiohjaimen näytölle levyn keskipisteen x- ja y-koordinaatit ja kiertokulma
TPWrite "X arvo "\Num:=PoytaX;
TPWrite "Y arvo "\Num:=PoytaY;
TPWrite "Kulma "\Num:=PoytaZ;

!TPReadFK Palautus, "Jatketaanko matkaa, vai palautetaanko kappale takaisin?", "Jatka", "Palauta", "", "", "";

!mikäli robotti ei saa kameralta kaikkea tietoa levystä levy voidaan palauttaa takaisin poimintapisteeseen
IF Palautus=2 THEN
    MoveJ Offs(Ottopaikka,0,0,250),v1000,z100,tImu\WObj:=wobj0;
    MoveJ Offs(Ottopaikka,0,0,0),v100,fine,tImu\WObj:=wobj0;
    Reset DO10_6;
    Waittime 0.5;
    MoveJ Offs(Ottopaikka,0,0,250),v100,z200,tImu\WObj:=wobj0;
    AJA_ODOTUS;
    main;
ENDIF

!pöydän ja kiertokulman laskeminen ja muuttaminen robotin koordinaatistoon sopivaksi
PoytaX:=600-PoytaX;
PoytaY:=300-PoytaY;

LaskKulm:=PoytaZ-500;

kosinikulma:=Cos(LaskKulm);
Q1kesken:=kosinikulma+kosinikulma+1+1;
Q4kesken:=1-kosinikulma-kosinikulma+1;
Q1kesken:=Sqrt(Q1kesken);
Q4kesken:=Sqrt(Q4kesken);
quu1:=Q1kesken/2;
quu4:=Q4kesken/2;
IF LaskKulm<0 THEN
    quu4:=(-1)*quu4;
ENDIF

PoytaX:=(-1)*PoytaX;

TPWrite "X muokattuna "\Num:=PoytaX;
TPWrite "Quu1 "\Num:=quu1;
TPWrite "Quu4 "\Num:=quu4;
TPWrite "LaskKulm "\Num:=LaskKulm;

LaskKulm:=(-1)*LaskKulm;

!kulmarajoituksella määritellään robotin 6. akselin kiertosuunta
Kulmarajotus:=(-1);
IF LaskKulm>27.9 THEN
    Kulmarajotus:=0;
ELSEIF LaskKulm<(-62.1) THEN
    Kulmarajotus:=(-2);
ENDIF

!robotti vie levyn työstökoneelle ja kääntää sen oikeinpäin
MoveJ Offs(liimaus,PoytaY,PoytaX,250),v1000,z50,tImu\WObj:=wobj0;
Set DO11_11;

```

```

MoveJ                               Offs([[836,-
220,160],[quu1,0.00000,0.00000,quu4],[0,0,Kulmarajotus,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]],PoytaY,PoytaX,0),v200
,fine,tlmu\WObj:=wobj0;

```

!imu suljetaan, odotetaan puoli sekuntia levyn irtoamista , resetoidaan kommunikointi ja nostetaan tarttuja ylös

```

Reset DO10_6;
WaitTime 0.5;
Reset DO11_11;
MoveL Offs(liimaus,PoytaY,PoytaX,250),v200,fine,tlmu\WObj:=wobj0;

```

ENDPROC

!haetaan levy työstökoneelta, robotti menee ottopisteen yläpuolelle, laittaa imun päälle, laskeutuu alas kunnes alipaineanturi
!tunnistaa paineen, odotetaan tarttumista ja siirretään levy odotuspisteelle

```

PROC HAE_LEVY()
MoveJ Offs(liimaus,PoytaY,PoytaX,250),v1000,z100,tlmu\WObj:=wobj0;
MoveL Offs(liimaus,150,-300,0),v500,fine,tlmu\WObj:=wobj0;
Set DO10_6;
WaitTime 0.5;
kunnessimu:=0;
MuutettuDZ:=0;
WHILE kunnessimu<2 DO
    kunnessimu:=DI10_5;
    kunnessimu:=kunnessimu+DI10_6;
    Decr MuutettuDZ;
    MoveL Offs(liimaus,150,-300,MuutettuDZ),v100,fine,tlmu\WObj:=wobj0;
ENDWHILE
WaitTime 0.5;
MoveL Offs(liimaus,150,-300,250),v500,z100,tlmu\WObj:=wobj0;
MoveJ offs(odotus,0,0,0),v1000,z200,tlmu\WObj:=wobj0;

```

ENDPROC

!viedään levy telineeseen

```

PROC VIE_TELINE()
TelineXD:=Telineeslevyi*(-99);

!robotti kulkee apu- ja kiertopisteen kautta telineen vapaan lokeron jättöpisteeseen
MoveJ Offs(tapu2,0,0,0),v500,z200,tlmu\WObj:=wobj0;
MoveJ Offs(kp,0,0,0),v500,z150,tlmu\WObj:=wobj0;
MoveJ Offs(jatto2,TelineXD,0,0),v500,fine,tlmu\WObj:=wobj0;
MoveL Offs(jatto2,TelineXD,0,-273),v100,fine,tlmu\WObj:=wobj0;

```

!sammutetaan imu, odotetaan hetki tartunnan irtoamista, siirretään robotin tarttuja kierto- ja apupisteeseen

```

!lisätään levylaskurin arvoa yhdellä
Reset DO10_6;
WaitTime 0.5;
MoveL Offs(jatto2,TelineXD,0,0),v1000,z100,tlmu\WObj:=wobj0;
MoveJ Offs(kp,0,0,0),v2500,z150,tlmu\WObj:=wobj0;
MoveJ Offs(tapu2,0,0,0),v2500,z200,tlmu\WObj:=wobj0;
Incr Telineeslevyi;

```

ENDPROC

!pääohjelma, suoritetaan alustukset ja ohjelmaa suoritetaan syklistesti

```

PROC main()
!digitaalisten lähtöjen nollaus
Reset DO10_9;
Reset DO10_10;
Reset DO10_11;
Reset DO10_13;
Reset DO11_10;
Reset DO11_11;
Palautus:=0;
IF palautus<>2 THEN
    set DO11_9;
!suoritetaan aliohjelma start jos robotti käynnistetään ensimmäisen kerran tai ohjelma on keskeytynyt
!muutoin kysytään käyttäjältä jatketaanko keskeytynyttä ohjelmaa
START;
!WaitDO DO11_1,1;
ELSE
TPReadFK Nappula, "Jatketaanko?", "Jatka", "", "", "", "";
nappula:=0;
ENDIF

```

!työkierron paluupiste

alku:

!tarkastetaan onko teline täynnä. Jos teline on täynnä, robotti ajaa odotuspisteeseen ja antaa äänimerkin
!laskuri nollataan tai käyttäjä antaa telineessä olevien levyjen lukumäärän

IF Telineeslevyi>9 THEN

AJA_ODOTUS;

!Summeri soi lyhyesti

Set DO10_8;

WaitTime 2;

Reset DO10_8;

TPReadFK Nappula, "Kaikki telinepaikat on taynna. Nollataan tai anna kaytettyjen paikkojen maara?", "Nollaa",

"Ann arv", "", "", "";

!teline tyhjä, nollataan laskuri

IF Nappula=1 THEN

Telineeslevyi:=0;

Nappula:=0;

!telineessä levyjä, päivitetään laskuri käyttäjän ilmoituksen mukaan

ELSEIF Nappula=2 THEN

Nappula:=0;

TPReadNum Telineeslevyi, "Kuinka monta telinepaikkaa on taytetty?";

!jos luku suurempi kuin telineessä on paikkoja, palataan alkuun

IF Telineeslevyi>9 THEN

Goto alku;

ENDIF

ENDIF

ELSE

!suoritettavien aliohjelmien kutsut

AJA_ODOTUS;

!WaitDO DO11_11,1;

PAIKKA;

LIIMA;

AJA_ODOTUS;

!3 sekunnin odotusaika, kuvaa tässä työstökoneen levynkäsittelyaika

WaitTime 3.0;

HAE_LEVY;

VIE_TELINE;

ENDIF

!pääohjelman sykli suoritettu, palataan alkupisteeseen

GOTO alku;

ENDPROC

!ohjelmiston lopetusmerkki

ENDMODULE

ABB IRB140 robotin RAPID-ohjelmointikielen käskylista

Prog Flow

Break	Stop program execution temporarily for debugging purposes.
Compact IF	Execute one instruction only if a condition is satisfied.
Exit Cycle	Stop the current cycle and move the program pointer to the first instruction in the main routine. When the execution mode <i>CONT</i> is selected, execution will continue with the next program cycle.
GOTO	Jump to a label Label Specify a label (line name)
Return	Return to the original routine
CallByVar	Call procedures with specific names
EXIT	Stop program execution when a program restart is not allowed
FOR	Repeat a section of the program a number of times.
IF	Execute a sequence of different instructions depending on whether or not a condition is satisfied.
ProcCall	Call (jump to) another routine
Stop	Stop program execution
SystemStopAction	Stop program execution and robot movement
WHILE	Repeat a sequence of different instructions as long as a given condition is satisfied
TEST	Execute different instructions depending on the value of an expression

Various E.g

:=	Assign a value to data
EraseModule	Erase a module from the program memory.
Save	Save a program module.
UnLoad	Unload a program module from the program memory
WaitDo	Wait until a digital output is set
CancelLoad	Cancel the loading of a module that is being or has been loaded with the instruction <i>StartLoad</i>
Comment	Comment on the program
Load	Load a program module into the program memory
StartLoad	Load a program module into the program memory during execution
WaitDI	Wait until a digital input is set.
WaitLoad	Connect the module, if loaded with <i>StartLoad</i> , to the program task
WaitTime	Wait a given amount of time or to wait until the robot stops moving

WaitUntil Wait a given amount of time or to wait until the robot stops moving

Settings

AccSet Define the maximum acceleration.

Confl Configuration check on/off during linear motion DitherDeact

Eoffson Activate an external axis offset

GripLoad The payload of the gripper

MotionSup Deactivates/activates motion supervision

ConfJ Configuration control on/off during joint motion

DitherAct Enables dither functionality for soft servo

EOffsOff Deactivate an external axis offset

EOffsSet Activate an external axis offset by specifying a value

MechUnitlLoad Defines a payload for a mechanical unit

PDispOff Deactivate program displacement

PDispOn Activate program displacement

SingArea The interpolation method through singular points

SoftDeact Deactivate the soft servo

PDispSet Activate program displacement by specifying a value

SoftAct Activate the soft servo for one or more axes

VelSet The maximum velocity and velocity override

Motion&Proc

ActUnit Activate an external mechanical unit

MoveAbsJ Absolute joint movement

MoveCDO Moves the robot circularly and sets a digital output in the middle of the corner path.

MoveJ Joint movement

MoveL TCP moves along a linear path

SearchC TCP along a circular path

DeactUnit Deactivate an external mechanical unit

MoveC TCP moves along a circular path

MoveExtJ Moves a linear or rotational external axis without TCP

MoveJDo Moves the robot by joint movement and sets a digital output in the middle of the corner path.

MoveLDo Moves the robot linearly and sets a digital output in the middle of the corner path.

SearchL TCP along a linear path

I/O

AliasIO	Define a signal with an alias name
IOBusStart	Start an I/O bus.
IODisable	Disable an I/O module
PulseDo	Generate a pulse on a digital output signal
Set	Set a digital output signal (to 1)
SetDo	Change the value of a digital output signal (symbolic value; e.g. <i>high/low</i>)
InvertDo	Invert the value of a digital output signal
IOBusState	Get current status of the I/O bus.
IOEnable	Enable an I/O module
Reset	Reset a digital output signal (to 0)
SetAO	Change the value of an analog output signal
SetGo	Change the value of a group of digital output signals
WaitDI	Wait until a digital input is set or reset
WaitDO	Wait until a digital output is set on reset

Communicate

ClearIOBuff	Clear input buffer of a serial channel
Close	Close the channel/file
CopyRowBytes	Copy from one rawbytes variable to another
Open	Open a serial channel/file for binary transfer of data
ReadAnyBin	Read from any binary serial channel
ReWind	Set the file position to the beginning of the file
ClearRawBytes	Set a rawbytes variable to zero
Copyfile	Copy a file.
ErrWrite	Write text on the FlexPendant disp. and simultaneously store that message in the program's error log.
PackRawBytes	Pack the contents of a variable into a "container" of type rawbytes
ReadRawBytes	Read data of type rawbytes from a binary serial channel/file/ field bus
SCWrite	Send a message to the superordinate computer
TPEraser	Clear the FlexPendant operator display
TPReadNum	Read a numeric value from the FlexPendant
TPWrite	Write text on the FlexPendant operator display
UIMShow	Open an application on the FlexPendant from RAPID
Write	Write text to the channel/file
WriteBin	Write to a binary serial channel/file
TPReadFK	Label the function keys and to read which key is pressed

TPShow	Choose a window on the FlexPendant from RAPID
UnPackRawBytes	Unpack the contents of a “container” of type rawbytes to a variable
WriteAnyBin	Write to any binary serial channel/file
WriteRawBytes	Write data of type rawbytes to a binary serial channel/file/fieldbus
WriteStrBin	Write a string to a binary serial channel/file

Interrupts

CONNECT	Connect a variable (interrupt identity) to a trap routine
IDelete	Cancel (delete) an interrupt
IEnable	Enable all interrupts
IPers	An interrupt when changing a persistent.
ISignalAO	An interrupt from an analog output signal
ISignalDO	An interrupt from a digital output signal
Gettrap	Data in a trap routine to obtain all information about the interrupt that caused the trap routine to be executed.
IDisable	Disable all interrupts
IError	Order and enable an interrupt when an error occurs
ISignalAI	An interrupt from an analog input signal
ISignalDI	An interrupt from a digital input signal
ISignalGI	An interrupt from a group of digital input signals
ISignalGO	An interrupt from a group of digital output signals
ITimer	A timed interrupt
RaiseToUser	From a NOSTEPIN routine, the error is raised to the error handler at user level.
ISleep	Deactivate an individual interrupt
IWatch	Activate an individual interrupt
ReadErr	Data in a trap routine, to obtain numeric information (domain, type and number) about an error, a state change, or a warning, that caused the trap routine to be executed.

Error Rec.

BookErrNo	Book a new RAPID system error number.
ErrRaise	Create an error in the program and then call the error handler of the routine
ProcerrRecovery	Generate process error during robot movement.
ResetRetrycount	Reset the number of counted retries.
Return	Return to the routine that called the current routine
Trynext	Execute the instruction following the instruction that caused the error
ErrLog	Display an error message on the teach pendant and write it in the robot message log.

Exit	Stop program execution in the event of a fatal error
Raise	Call the error handler of the routine that called the current routine
Retry	Re-execute the instruction that caused the error
Skipwarn	Skip the latest requested warning message.

System&Time

ClkReset	Reset a clock used for timing
ClkStop	Stop a clock used for timing
MakeDir	Create a new directory.
ReadCfgData	Read one attribute of a named system parameter.
RemoveFile	Remove a file.
WriteCfgData	Write one attribute of a named system parameter.
ClkStart	Start a clock used for timing
CloseDir	Close a directory in balance with <i>OpenDir</i> .
OpenDir	Open a directory for further investigation.
RemoveDir	Remove a directory.
RenameFile	Rename a file.

Matematics Käskyt

:=	Perform calculations on any type of data
BitClear	Clear a specified bit in a defined <i>byte</i> data.
Clear	Clear the value
Incr	Increment by 1
Add	Add or subtract a value
BitSet	Set a specified bit to 1 in a defined <i>byte</i> data.
Decr	Decrement by 1
TryInt'	Test if data object is a valid integer

MotionSetAdv

CirPathMode	Choose the way the tool reorientates during circular interpolation
PathResol	Adjust the geometric path resolution
TuneReset	Reset tuning to normal
WZBoxDef	Define a box-shaped global zone
WZDisable	Deactivate supervision of a temporary global zone
PathAccLim	Set or reset limitations on TCP acceleration and/or TCP deceleration along the movement path.
SpeedRefresh	Update speed override for ongoing movement

TuneServo	Adjust the robot tuning values
WorldAcclim	Limiting the acceleration/deceleration of the tool (and griplload) in the world coordinate system.
WZClyDef	Define a cylindrical global zone
WZDoSet	Activate global zone to set digital outputs
WZEnable	Activate supervision of a temporary global zone
WZHomeJointDef	Define a global zone in joints coordinates
WZLimSup	Activate limit supervision for a global zone
WZFree	Erase supervision of a temporary global zone
WZLimJointDef	Define a global zone in joints coordinates for limitation of working area.
WZSphDef	Define a spherical global zone

MotionAdv

ClearPath	Clear the whole motion path on the current motion path level.
MoveJSync	Moves the robot by joint movement and executes a RAPID procedure
RestoPath	Regenerate a path stored earlier
StartMoveRetry	Restart the robot movements and make a retry in one indivisible sequence
StopMove	Stop the robot movements
StorePath	Store the last path generated
MoveCSync	Moves the robot circularly and executes a RAPID procedure
MoveLSync	Moves the robot linearly and executes a RAPID procedure
StartMove	Restart the robot movements
StepBwdPath	Move backwards on its path in a RESTART event routine
StopMoveReset	Reset the stop move status, but don't start the robot movements
TriggC	Run the robot (TCP) circularly with an activated trigg Condition
TriggCheckIO	Define an IO check at a given position
TriggInt	Define a trigg condition to execute a trap routine at a given position
TriggJ	Run the robot axis-by-axis with an activated trigg condition
TriggSpeed	Define conditions and actions for control of an analog output signal with output value proportional to the actual TCP speed.
TriggEquip	Define a trigg condition to set an output at a given position with the possibility to include time compensation for the lag in the external equipment
TriggIO	Define a trigg condition to set an output at a given position
TriggL	Run the robot (TCP) linearly with an activated trigg condition
TriggRampAO	Define a trigg condition to ramp up or down analog output signal at a given position with the possibility to include time compensation for the lag in the external equipment

TriggStopProc Create an internal supervision process in the system for zero setting of specified process signals and the generation of restart data in a specified persistent variable at every program stop (STOP) or emergency stop (QSTOP) in the system

MultiTaskin&WaitTestAndSet

SyncMoveUndo Reset synchronized movements

WaitTestAndSet Retrieve exclusive right to specific RAPID code areas or system resources (type interrupt control)

RAPID Support

GatDataVal Get a value from a data object that is specified with a string variable

SetDataSearch Together with *GetNextSym* data objects can be retrieved from the system.

SetSysData Activate a specified system data name for a specified data type.

WarmStart Restart the controller e.g. when you have changed system parameters from RAPID.

GetSysData Fetch data and name of current active Tool or Work Object.

SetAllDataVal Set a new value to all data objects of a certain type that match a given grammar.

SetDataVal Set a value for a data object that is specified with a string variable

TextTabInstall Install a text table in the system.

Calib&Service

MToolRotCalib Calibrate the rotation of a moving tool.

SPyStart Start the recording of instruction and time data during execution.

SToolRotCalib Calibrate the TCP and rotation of a stationary tool.

TestSignDefine Define a test signal

MToolTCPCalib Calibrate Tool Centre Point - TCP for a moving tool.

SPyStop Stop the recording of time data during execution

SToolTCPCalib Calibrate Tool Centre Point - TCP for a stationary tool

TestSignReset Reset all test signals definitions

M.C.1

MoveJ Joint movement

MoveC TCP moves along a circular path

Set Set a digital output signal (to 1)

WaitTime Wait a given amount of time or to wait until the robot stops moving

:= Assingn a value to data

MoveL TCP moves along a linear path

ProcCall	Call (jump to) another routine
Reset	Reset a digital output signal (to 0)
WaitDI	Wait until a digital input is set or reset
CompactIF	Execute one instruction only if a condition is satisfied
Incr	Increment by 1
Add	Add or subtract a value
Decr	Decrement by 1