

Opinnäytetyö (AMK)

Kone- ja tuotantotekniikka

NKONTK12

2017

Teemu Hentula

ROBOTIIKAN LABORATORIOTÖIDEN KEHITTÄMINEN

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Kone- ja tuotantotekniikka

2017 | 34 sivua

Teemu Hentula

ROBOTIIKAN LABORATORIOTÖIDEN KEHITTÄMINEN

Tämän opinnäytetyön aiheena on robotiikan laboriotöiden kehittäminen ja ohjeistaminen Turun ammattikorkeakoulun konetekniikan opiskelijoille. Kehittämistyö ja robottiharjoitukset suoritettiin Turun ammattikorkeakoulun tarjoamassa #Tehdas-oppimisympäristössä. Työn tavoitteena on ohjata opiskelijoita robottisoluilla, sekä laatia ja kehittää helppokäyttöinen ohjekirja robotin käyttöä ja ohjelmointia varten. Ohjeiden avulla pyritään ohjeistamaan opiskelijoita robotiikan perusteista mahdollisimman havainnollisesti. Lisäksi harjoitusten tarkoituksena on saada opiskelijat soveltamaan ohjeista opittuja taitoja ja kokeilemaan robottisolun yksinkertaista ohjelmointia. Käyttöoppaasta karsittiin pois kaikki sellainen tieto, jota opiskelija ei tarvitse alkaessaan perehtyä robotiikkaan. Käyttöoppaan tavoitteena on selkeyttää ja kerätä tarvittava tieto olemassa olevista käyttömanuaaleista yhteen helposti luettavaan muotoon. Opiskelijoilta saatu palaute harjoituksista, sekä opetustilanteessa syntyneet keskustelut ja esiin nousseet kysymykset ovat vaikuttaneet osaltaan materiaalin kehitykseen. Työtä kehitettiin oppilaiden kanssa kuuden viikon ajan oppimisympäristössä laboriotöissä ja opiskelijoiden palauttamia raportteja on käytetty hyväksi ohjeiden kehitystyössä.

ASIASANAT:

Robotiikka, robottisolu, ohjeistus

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Mechanical and production engineering

2017 | 34 pages

Teemu Hentula

DEVELOPMENT OF ROBOTICS LABORATORY

The subject of this thesis is the development of robotics laboratory to machine automation students of Turku University of Applied Sciences. The development and guidance were performed at the #Factory-learning environment. The main objective of the project was to make and develop useful instruction manual and easy exercises for the teaching the basics of robot programming. The intention of exercises was to encourage students to apply abilities that they had learned. All information that was already available on existing user manuals was not necessary for understanding the basics of robotics and was left out of the final user's guide. Feedback about exercises and interaction with the students has affected to the development of the robotics material. The project was improved in laboratory work environment for six weeks and the reports and ideas from the students were exploited for the manual.

KEYWORDS:

Robotics, robot cell, guidance

SISÄLTÖ

1	JOHDANTO	5
1.1	Aiheen määrittely	5
1.2	Meta-analyysi	6
2	ROBOTIIKKA	8
2.1	Robotti	9
2.2	Robottityypit	9
2.3	Robottien koordinaatistot	10
2.4	Robotin ohjelmointi	11
2.4.1	Robotin ohjelmointi opettamalla	13
2.4.2	Muut mahdolliset ohjelmointi tavat	14
2.5	Liikkeen käskyt ja liiketyypit	15
2.6	Tarraimet ja työkalut	17
2.6.1	Mekaaniset tarraimet	18
2.6.2	Imu- ja tyhjiötarttijat	18
2.6.3	Robotissa käytettävät muut työkalut	19
2.7	Teollisuusrobottien käyttökohteet	20
3	ROBOTIIKKAKURSSIN OHJAUS	22
3.1	Ohjauksessa käytetty laitteisto	23
4	KEHITYSTYÖHÖN LIITTYVÄÄN MATERIAALIIN TUTUSTUMINEN	25
5	OPPIMATERIAALIN KEHITTÄMINEN	27
5.1	Opetuksen suunnittelun vaiheet ja tasot	28
5.2	Olemassa olevan materiaalin hyödyntäminen kehittämistyössä	28
6	YHTEENVETO JA POHDINTAA	30
6.1	Oppimateriaalin ja ohjeiden kehittäminen tulevaisuudessa	30
6.2	Palaute ja kehittämistyö	31
	LÄHTEET	33
	LIITTEET	34

1 JOHDANTO

Opinnäytetyön tavoitteena on suunnitella ja kehittää olemassa olevan materiaalin pohjalta käyttö- ja koulutusmanuaalia paremmin vastaamaan oppilaiden tarpeita sekä selkiinnyttää ohjeita. Helpoiten tämä on saavutettavissa tuomalla tarvittava tieto yhteen helposti luettavaan oppaaseen monen eri käyttöohjeen selaamisen ja hakemisen sijaan. Tällöin kurssin harjoitustehtävät on helppo lisätä ohjeiden liitteeksi oppilaiden sovellettaviksi opasta hyväksi käyttäen. Osana toteutusta on tarkoitus opastaa ja opettaa oppilaita robotiikassa sekä näyttää harjoitustehtävän onnistunut suoritus oppilaille ennen heidän perehtymistä kyseiseen tehtävään.

1.1 Aiheen määrittely

Sopivaa opinnäytetyön aihetta puntaroidessa, käytiin keskustelua Turun ammattikorkeakoulun koulutus- ja tutkimusvastaavan, Sakari Koivusen kanssa mahdollisuudesta suorittaa työ koulun oppimisympäristössä. Keskustelussa esiin nousi ajatus robottikoulutuskurssin ohjauksesta aloitteleville konetekniikan AMK –opiskelijoille. Yhtenä osana työhön kuuluisi itse opetus ja lähituntien vetäminen opiskelijoille Turun ammattikorkeakoulun #Tehdas-oppimisympäristössä, ja toisena osana ohjeiden sekä käyttöoppaan laatiminen kurssin edetessä. Kehitystyö ja laboratorioharjoitusten ohjaus sijoittuisi vuoden 2016 kevääseen.

Aikaisemmin robotiikkaan perehtyvässä peruskurssissa laboratoriotöitä on suoritettu alkuohjeistuksen ja tehtävänannon jälkeen opiskelijoiden kesken omatoimisesti opettajan kierrellessä eri työpisteiden välillä. Huonoimmassa tapauksessa opiskelijat ovat jumiutuneet heti alkuun ohjeistuksen puutteellisuuden vuoksi, jonka seurauksena harjoitustehtävät ovat jääneet puutteellisesti suoritetuiksi. Tämän takia arvokkaasta, työelämään valmentavasta käytännönkokemuksesta ei ole saatu parasta hyötyä irti oppimisen kannalta.

Opiskelijat pääsevät käyttämään kurssin aikana robotiikkaan perehtymisessä #Tehdas-oppimisympäristön tarjoamia nykyaikaisia robottisoluja, joihin itsekin olen saanut koulutuksen opiskelujeni aikana Turun ammattikorkeakoulussa. Harjoitustöitä ja manuaalia laatiessani myös oma kokemus ja tietämys kyseisistä robottisoluista kehittyi huomattavasti eteenpäin aikaisempien perusteiden ympärille.

1.2 Meta-analyysi

Meta-analyysin tavoitteena on tutkia ja analysoida aiemmin tehtyjä opinnäytetöitä, joissa on tutkittu robottisoluja ja robotiikan soveltamista opiskelijoille. Lisäksi tarkastelun tavoitteena on pohtia, mitä hyvää ja huonoa kyseisissä opinnäytetöissä on. Tämän tarkastelun ansioista pyritään ohjaamaan omaa toimintaa oikeaan suuntaan. Valittavana oli useita robotiikkaan pohjautuvia opinnäytetyö vaihtoehtoja, joista valitsin Tommi Mikkosen Fanuc-robotin käyttöoppaan kehittäminen (Mikkonen, 2012) sekä Tomi Haapalaisen ABB robotin käyttöönottoasennus ja ohjelmointi konenäkösovelluksessa (Haapalainen, 2011).

Opinnäytetyössään *Fanuc-robotin käyttöoppaan kehittäminen* Tommi Mikkonen pyrkii kehittämään robottisolun käyttöoppaan Tampereen ammattiopiston kone- ja metalliosaston käyttöön. Käyttöohjeen avulla tuli voida opettaa robottiohjauksen perusteet mahdollisimman havainnollisesti. Käyttöoppaan yhtenä tavoitteena on esitellä robotiikka kiinnostavana aiheena käyttäjälleen. Työssään hän on kerännyt opiskelijoiden ja opettajien mielipiteitä kyselyjen sekä haastattelujen pohjalta.

Opinnäytetyön alkuosassa Mikkonen keskittyy Tampereen Ammattiopiston esittelyyn sekä tulevaisuuden näkymiin. Tässä hän vetoaakin ammattiopiston haluun vastata kone- ja metallialan jatkuvasti kehittyviin haasteisiin ja varmistaa opetussuunnitelmissa esitettyjen ammatillisen koulutuksen tehtävien ja tavoitteiden täyttymisestä. Robottikoulutus kaipaisi uutta ja selkeää käyttömanuaalia, jossa oppilaille syntyisi innostusta ja edellytyksiä itsenäiseen opiskeluun ja tiedonhankintaan.

Työn keskiosa koostuu robotiikan historiasta ja ohjelmoinnista, aina nykyaikaiseen oliopohjaiseen etäohjelmointiin saakka. Mikkonen on käynyt perusteellisesti läpi robotiikkaan ja robottisoluun liittyvät kohdat, joissa kerrotaankin yksityiskohtaisesti eri koordinaatistojärjestelmistä, ohjelmoinnista, liiketyypeistä, työkaluista sekä käyttökohteista. Lopuksi Mikkonen keskittyy olemassa olevaan materiaalin sekä pedagogisiin lähtökohtiin. Tästä aihe siirtyy itse oppimateriaalin kehittämistyöhön, eri käyttöopasversioihin ja tulosten analysointiin.

Kaiken kaikkiaan Mikkosen työssä on todella laaja perehdytys robotiikkaan ja sen mahdollisuuksiin ennen käyttöoppaan kehittämisvaihetta. Tämä on osaltaan ymmärrettävää, jotta lukija pystyy saamaan tarpeeksi selkeän kuvan, mistä robotiikassa ylipäätään on

kyse. Mikkonen on käyttänyt paljon suoria lainauksia eri lähteistä teoriaosuuden kasamiseksi. Teksti on selkeää, vaikka aiheessa mennään joissain osin mielestäni turhaan perusteita syvemmälle. Työn tarkoituksena oli tehdä selkeä käyttöopas opiskelijoille, mutta robotiikan teoria-osuudesta olisi voinut karsia perusteet selvästi ylittävät osiot pois. Kehittämistyö on käsitelty vähintään yhtä laajasti kuin robotiikan teoriaosuuskin. Mikkonen esittää kehitystyön kohteet ja kehittämisen syklit, jonka jälkeen hän kertoo vuorovai-
kutuksista opiskelijoihin. Havainnot, kysymykset sekä haastattelut on otettu huomioon ennen eri käyttöoppaiden versiointia sekä tulosten esittämistä. Loppuun Mikkonen on jaksanut pohtia työtään ja onnistumistaan. Hän ottaa esille myös tulevaisuuden näkymät käyttöoppaan kehittämisessä.

Tomi Haapalainen kirjoitti opinnäytetyönsä ABB robotin käyttöönottoasennuksesta ja ohjelmoinnista. Työ on tehty konenäköjärjestelmään perustuvan projektin pohjalta, jossa konenäköjärjestelmän kanssa on asennettu yleisrobotti ja sen ohjainjärjestelmä. Opin-
näytetyö käsittelee yleisesti automaation ja robotiikan historiaa, kehitystä sekä konetur-
vallisuuteen liittyviä asioita.

Koska Haapalaisen opinnäytetyö koskee hyvin teknistä asennustyötä ja ohjelmointia, hän käy työssään perusteellisesti läpi robotin ohjelmoinnin periaatteet sekä käyttöönoton ja asennuksen kannalta tärkeimmät asiat. Työ onkin kokonaisuudessaan teknistä ja teo-
reettista robotiikan läpikäymistä. Suurimpana erona aikaisempaan läpikäymääni, Mikko-
sen opinnäytetyöhön, on turvallisuus, asennus ja huolto-osiot. Haapalainen on käynyt läpi samat robotiikan osa-alueet, mutta esittänyt paljon asioita erilaisin havainnollistavin
kuvin sekä taulukoin ilman minkäänlaisia viitteitä itse tekstiin. Kuvista saa kontekstin ku-
vatekstien avulla aiheeseen, mutta osa kuvista on tarpeettomia, työtä pitkittäviä objek-
teja. Haapalainen kertookin omasta työstään melko suppeasti kaiken teorian sekä työs-
tään otettujen kuvien ohella.

2 ROBOTIIKKA

Robottiikka on yksi edellytys yrityksen mukana pysymiseen jatkuvasti kehittyvässä teollisuuden kilpailussa. Vaatimusten noustessa sekä teknologian kehittyessä yritysten on vastattava kasvavaan kulutustuotteiden kysyntään tehostamalla valmistusmenetelmiään ja lisäämällä tuotteidensa tuotantomääriä. Edellä mainittujen asioiden saavuttamiseksi vaaditaan nyt ja varsinkin tulevaisuudessa yhä enemmän automaatioteknologiaa, johon myös robotiikka lukeutuu. Hyödyntämällä robotiikkaa teollisuudessa kyetään parantamaan maailmanlaajuisesti ihmisten elintasoa, palveluja sekä luomaan uusia työpaikkoja. Valitettavasti robotiikka mielletään helposti ihmisten työpaikkoja tuhoavaksi automaatioalan haaraksi, vaikka totuus on pitkällä aikavälillä kaikkea muuta. Robotin korvataessa ihmisen tekemää työtä, ohjaa se samaan aikaan osaavia ihmisiä kyseiselle alalle luoden näin uusia työpaikkoja robotiikan piiriin. Myös robottien huoltoon, valmistukseen ja opeointiin tarvitaan jatkuvasti uusia työntekijöitä. Robotit voivat myös suorittaa sellaiset työtehtävät, jotka eivät vaatimuksiensa tai puutteellisen turvallisuutensa takia sovellu ihmiselle. Kyseisissä tilanteissa robotti saattaa olla korvaamaton apuväline. Käytännössä ihmisten aiemmin suorittamien yksinkertaisempien, liukuhihnatöiden siirtäminen robotille vapauttaa työntekijöitä tehtäviin, joissa ihminen suoriutuu konetta paremmin. On myös asioita, joissa tarvitaan luovuutta tai aisteja tavalla, joihin ihminen kykenee konetta paremmin.

Vaikka robotit kykenevät jo nyt joissain määrin aistimaan ja tulkitsemaan ympärillään tapahtuvia asioita erilaisten antureiden avulla, ennustetaan ettei robotteja tarvitsisi lainkaan ihmisen toimesta ohjelmoida tekemään työtä tulevaisuudessa. Tällöin robotit kykenisivät tekemään itse omat ratkaisunsa sekä voisivat tekemisen kautta oppia uutta. Robottien itsenäistä oppimisprosessia nopeuttaisi entisestään robottiverkosto, jossa robotit olisivat jatkuvassa yhteydessä toisiinsa sijainnista piittaamatta. Jos robotit kykenisivät kommunikoidaan keskenään, robottien kyky suoriutua erittäin haastavistakin tehtävistä olisi entistä todennäköisempää (KIDE 2014).

2.1 Robotti


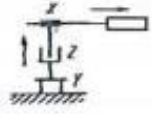








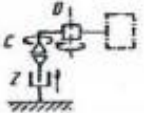


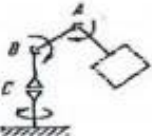


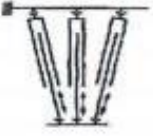

Robotin määritelmä vaihtelee eri lähteitä vertaamalla. Pääsääntöisesti käytetään esimerkiksi ISO 8373:2012 – standardia, jonka mukaan teollisuusrobotti on automaattisesti ohjattavissa oleva, uudelleen ohjelmoitava, monikäyttöinen laite, jota voidaan ohjelmoida toimimaan vähintään kolmella vapausasteella teollisessa käytössä. (ISO 8373:2012). Yleisesti voidaan todeta robotilla viitattavan mekaaniseen laitteeseen, joka ohjelmoidulla kykenee konkreettisiin fyysisiin suoritteisiin ja on tämän lisäksi monikäyttöinen sekä uudelleen ohjelmoitavissa muita tehtäviä varten. Lisäksi robotilla tulee olla vähintään kolme niveltä, joiden varassa se liikuttelee kappaleita, osia, työkaluja tai erikoislaitteita.

Robotti ohjelmoidaan usein ennalta kulkemaan eri liikeratoja sekä tekemään suoritteita teollisuuden sovellutuksissa. Robotin jalustan ja työkalun välissä on tukivarsia, joita nivelet liittävät toisiinsa. Teollisuusrobotit jaetaan yleensä eri luokkiin tai tyypeihin mekaanisen perusrakenteensa ja liikekoordinaatistonsa mukaan.

2.2 Robottityypit

Robotit jaetaan niiden nivelten lukumäärän ja käyttötarkoituksen mukaan moneen eri tyyppiin. Usein robottien rakenteessa on yritetty matkia ihmisen raajojen toimintaa ja robotin rakenteissa onkin ihmisen käsivartta, rannetta ja kouraa vastaavia nivelliikkeitä. Kuvassa 1 on esitetty yleisimpiä teollisuusrobottityyppejä, jotka ovat kiertyväniveliset-, nivelvarsi-, napakoordinaatisto-, rinnakkaisrakenteiset-, sylinteri-, suorakulmaiset-, sekä SCARA-robotit.

Teollisuudessa sekä Koneteknologiakeskuksella yleisin robottityyppi on kiertyvänivellinen robotti, mikä johtuu sen soveltuvuudesta moneen eri tehtävään ja käyttötarkoitukseen laajan työskentelyalueen vuoksi. Yleisesti käytetty kuuden vapausasteen robotti, jossa on siis kuusi pyörivää niveltä, kykenee käyttämään työkaluaan jokaisessa mahdollisessa asennossa sekä lähestymään luotua työkalupistettä ilman rajoitteita. Tämän robottityypin heikkous onkin sen kuormankantokyky kokoonsa nähden, jonka se korvaa rakenteensa tuomalla ulottuvuus edulla ahtaissakin paikoissa.

Nimitys pääakseleiden mukaan	Rakenne	Kinemaattinen kaavio	Työalue
Suorakulmainen robotti			
Sylinterirobotti			
Napa-koordinaatisto-robotti			
Scara-robotti			
Kiertyvänivelinen robotti			
Rinnakkaisrakenteinen robotti			

Kuva 1. Yleisimpien robottityyppien rakenteita, kinematiikkaa ja työalueita (Kuivanen 1999, 12).

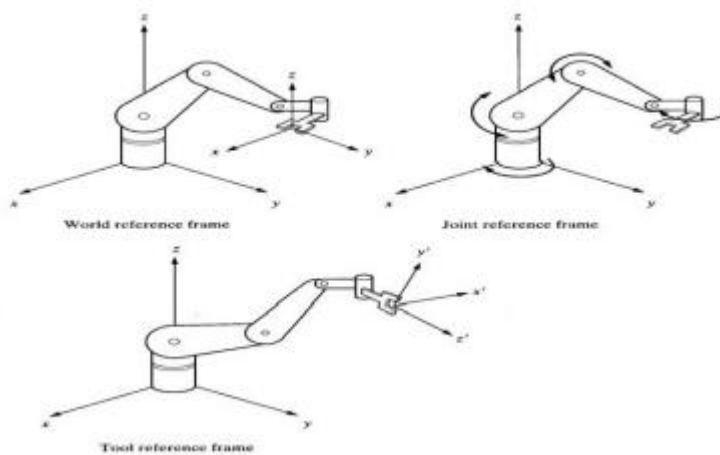
2.3 Robottien koordinaatistot

Teollisuusrobotin koordinaatistot jaetaan standardin ISO 9787- 1999 (Manipulating industrial robots, Coordinate systems and motion nomenclatures) mukaan maailma-, perus-, ja työkalukoordinaatistoon. (SFS-ISO 9787, 1999) Lisäksi robotille voidaan määrittää käyttäjäkoordinaatisto tilanteessa, jossa työstettävä kappale on erikoisen muotoinen tai on tarpeellista liikuttaa robottia eri suuntaan kuin muiden koordinaatistojen akselien linjat menevät.

Maailmakoordinaatisto on robotin toimintaympäristöön sidottu koordinaatisto, jolloin robotin asennolla ei ole vaikutusta koordinaatiston suuntiin. Robotti liikkuu pääakselien x-, y-, tai z-akselien suuntaisesti.

Nivel- eli peruskoordinaatistossa robotin jokaista yksittäistä niveltä liikutetaan erikseen.

Työkalukoordinaatisto on työkaluun tai sen siirtämään kappaleeseen luotu suorakulmainen koordinaatisto, joka on usein eri asennossa kuin suoraan työkalulaippaan sidottu koordinaatisto. Työkalukoordinaatisto liikkuu robotin liikkeiden mukana. Edellä mainitut koordinaatistot on esitetty alla. (Kuva 2.)



Kuva 2. Yleisimmät robottisolun koordinaatistojärjestelmät.

2.4 Robotin ohjelmointi

Ensimmäisten robottien ohjelmointi 1960-luvulla toteutettiin sähkömekaanisten kytkentöjen avulla. Tällä tavoin robotin eri nivelet saatiin ajettua haluttuja rajakatkaisijoita päin vaihe kerrallaan. Robottien ohjelmointi johdattamalla oli ensimmäinen ohjelmointitapa, jolla robotin nivelliikkeet saatiin muutettua suorista vaihe kerrallaan tapahtuneista lineaariliikkeistä hieman joustavammiksi. Opettamalla ohjelmointi on nykyään yleisesti käytetty ohjelmointitapa roboteille. Suurin osa nykyisistä sovelluksista on opetettu robotille liikuttamalla käsivarsi muutamiin asemiin, mutta luomalla toiminnan logiikka ja useita liikeratojen asemia tietokoneohjelmoinnilla. Kun robotista ja työympäristöstä on kolmiulotteinen tietokonemalli, voidaan tehdä mallipohjaista ohjelmointia erillisessä tietokoneessa.

Teleoperointi eli robottien liikuttamista etäältä teknisen käyttäjäliittymän avulla. Teleoperoinnissa robottia voidaan ohjata esimerkiksi joystickin ja graafisen käyttöliittymän avulla reaaliaikaisen kuvan perustella. Teleoperoinnissa voidaan hallita robottia täydellisesti ja suorittaa liikkeitä manuaalisesti etäältä. Useimmin teleoperoinnilla autetaan robottia suoriutumaan työtehtävässään eteenpäin vaiheesta seuraavaan. Teleoperointia hyödynnetään esimerkiksi avaruus- ja meriteollisuudessa haastavissa työympäristöissä.

Ohjelmoinnin tärkeimmät tehtävät:

- Laaditaan toimintajärjestys ja logiikka robottikäsivarren liikkeille sovelluksessa tarvittavien työkalun liikkeiden toteuttamiseksi
- Tahdistetaan käsivarren liikkeet ympäristön signaaleihin tai välitetään muihin laitteisiin tarvittavia tietoja
- Määritellään robotin toiminta virhetilanteissa (Robotiikka 2016, 47.)

Opetusohjelmointi-tilassa robottisolua ei voi käyttää tuotantokäytössä. Haluttaessa parantaa automatisoidun tuotantojärjestelmän käyttösuhdetta tarvitaan kehittyneempi ohjelmointikieli ja -laitteisto, jotka mahdollistavat etäohjelmoinnin. Paras tulos saavutetaan, jos ohjelma voidaan laatia ja testata erillisessä ohjelmointijärjestelmässä. Tällöin robotin tuotantokäyttö ei häiriinny ja ohjelman virheitä voidaan karsia pois ennen ohjelman viemistä tuotantoasteelle.

Liikeratojen ohjelmointi geometrinen peruselementtien, kuten ympyrän kaarien ja suorien avulla, parantaa ohjelmoidun radan tarkkuutta ja mahdollistaa vaativien robotisointien toteutuksen, esimerkkinä on kaarihitsausrobotin ohjelmointi. Geometria luodaan peruselementeistä kuten suorat, ympyrän kaaret ja spline -käyrät. Liikekäskyillä toteutetaan geometriasta yhdistelty liikerata. Ohjelmointijärjestelmä mahdollistaa ohjelman sisäiset laskutoimitukset, silmukat ja aliohjelmien käytön. (Robotiikka 2016, 48.)

Online ohjelmoinnilla voi hoitaa yksinkertaisimmat robotisoinnit esimerkiksi hitsaus- ja maalaussovelluksissa. Yleinen tapa työstökoneiden ja kokoonpanon robotisoinneissa on käyttää offline ohjelmointia. Tätä voidaan tehostaa liittämällä offline ohjelmointi osaksi tietokoneavusteista suunnittelua ja valmistusta. Liikeratojen simulointi vähentää usein ohjelman virheitä sekä lyhentää tuotannossa tehtävää ohjelman säätöaikaa.

2.4.1 Robotin ohjelmointi opettamalla

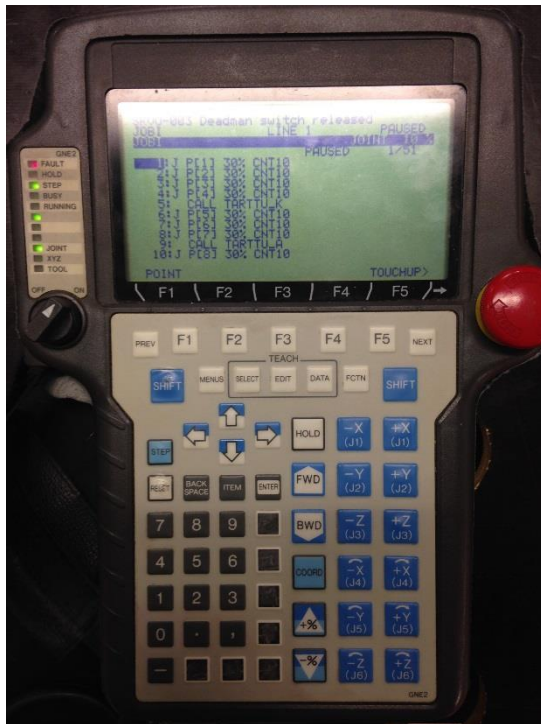
Opettamalla ohjelmointi perinteisesti tapahtuu viemällä robotin työkalu haluttuun paikkaan käsiohjaimen (esitetty kuvassa 3.) avulla ja tallettamalla asema muistiin. Tätä ohjelmointitapaa käytetään usein yhdessä tekstiä kirjoittavan ohjelmoinnin kanssa, jossa tarvittava teksti kirjoitetaan funktionäppäimistön avulla. Liikkumista eri asemien välillä ohjataan erilaisin hyppykäskyin ja aliohjelmin. Käyttökokemuksen kannalta on tärkeää, että robotin käsiohjaimen käskykanta ja aliohjelmia voidaan selata ilman erillistä ohjekirjaa.

Robottien ohjelmointikieliet muistuttivat alkuun Basic-kieltä, johon oli lisätty tieto nivelliikkeen liikekäskystä. Nykyään kieli muistuttaa maailmanlaajuisella muuttujavälilyksellä olevaa Pascal-kieltä. Valitettavasti eri toimittajien välillä on usein oma ohjelmointikielensä globaalin standardin puuttuessa. Onneksi eri robottivalmistajien ja -toimittajien vähentyessä kielet ovat alkaneet muistuttaa huomattavasti toisiaan.

Tyypillisen teollisuusrobotin peruskäskykanta:

- Liiketavan valinta eli lineaarinen, ympyrä tai nivelinterpolaatio
- Liikekäsky
- Liikekäsky suhteellisesti muunnettuun pisteeseen, mutta parametrina annetun aseman asennossa
- Liikeradan siirto kolmessa ulottuvuudessa
- Tehtäväkoordinaatiston siirto kuudessa vapausasteessa
- Etsintäliike eli pysähtyminen ulkoisen anturin tilan muuttuessa odotetusti
- Liikenopeuden ohjaus ulkoisella anturilla
- Ulkoisten antureiden avulla laskettujen korjausliikkeiden lisäys nimelliseen liikerataan
- Työkalukoordinaatistoliikkeet
- Lavaus säännöllisessä muodossa tai lavan purkaus vastaavasti
- Ehto, toisto -ja silmukkarakenteet

- Binääristen ja analogisten signaaleiden luku ja kirjoitus (Robotiikka 2016, 50-51.)



Kuva 3. Fanuc robottisolun käsiohjain ohjelmointiin ja operointiin.

2.4.2 Muut mahdolliset ohjelmointi tavat

Johdattamalla ohjelmointi tapahtui aikoinaan vapauttamalla robotin käsivarren toimilaitteet, jonka jälkeen ihminen liikkutti liikkeiden määrittelyvaiheessa halutun liikeradan. Nivelten paikka-antureiden lukemat talletettiin liikkeiden aikana nauhuriin ja kun liikkeitä toistettiin, yhdistettiin nauhuri toimilaitteiden säätöpiireihin.

Nykyään robotin käsivartta johdatetaan haluttua liikerataa pitkin siihen asennettua kahvaa ja aktivointinäppäimistöä käyttäen. Johdattamalla ohjelmoinnista tuli nopeasti maalauksrobotteissa käytetty menetelmä maalauksen onnistumisen johdosta, vaikka liikeradat eivät toistuisikaan aivan tarkasti. Nykyisin voimaohjauksen tullessa käyttöön useissa robottimalleissa, esimerkiksi niin sanotuissa collaborative eli yhteistoiminta roboteissa, johdattamalla ohjelmoinnista on tullutkin tavallaan kehittynein, nopein ja helpoin tapa ohjelmoida robotteja. (Robotiikka 2016, 49.)

Etäohjelmointi eli offline ohjelmointi on yleistymässä nykypäivänä, koska sen avulla voidaan ohjelmoida tuotantorobottia tuotannon ulkopuolisessa tietokoneessa käyttäen 3D

graafista käyttöliittymää sekä sen simulointimalleja. Simulointimalleilla voidaan tarkastaa ohjelman rata, sekä mahdollisten työkalujen ja kiinnittimien sopivuus etukäteen. Tämä mahdollisuus helpottaa kaikkein vaikeimmissa sovellutuksissa kuten hitsaus, leikkaus, jäysteen poisto, kiillotus, pinnoitus ja kappaleen työstö.

Ennen ohjelmointia robottisolusta on tehty simulointimalli, joka on kalibroitu vastaamaan todellista solua. Etäohjelmointi perustuu samalla tavalla robotin paikoituspisteiden tallentamiseen koordinaatistossa kuten opettamalla ohjelmointikin. Lähestymispisteet saadaan paikoitettua tuotteen 3D CAD-kuvan perusteella automaattisesti, jotka tarkistetaan ennen varsinaisen tuotantoajon alkua manuaalisesti törmäyksien välttämiseksi. Paikoituspisteisiin lisätään tarvittavat aliohjelmat, joilla kerrotaan mitä pisteessä halutaan tehdä. Ohjelman valmistuttua se tarkastetaan simulointiajolla ja sen suoriutuessa halutulla tavalla, siirretään se tuotantoon robotin ymmärtämään muotoon.

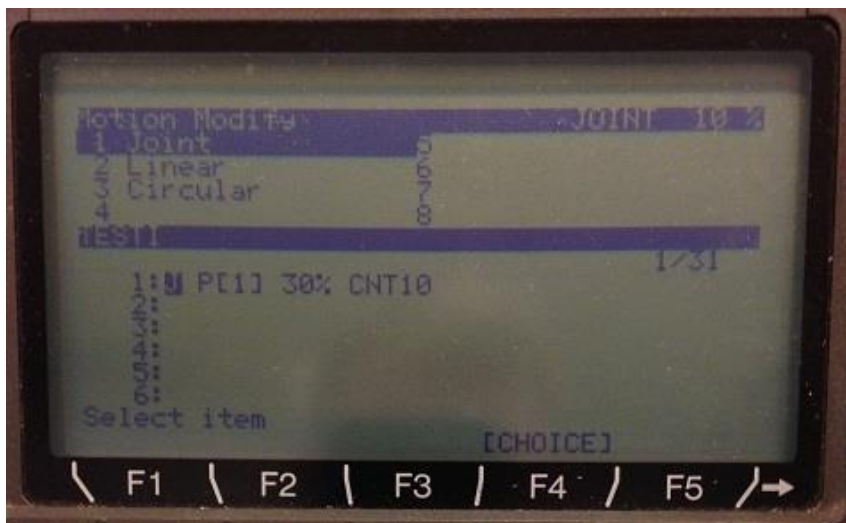
2.5 Liikkeen käskyt ja liiketyypit

Robottisolussa liikkeen käskyt ohjaavat robottia liikkumaan määrättyyn asemaan määrättyllä nopeudella. Liikkeen käskyihin kuuluvat liiketapa, paikan tyyppi, nopeus, paikoitustapa ja liikkeen valinnat.

Liiketyyppejä on kolmea perustyyppiä. Nivelliike (joint) liikuttaa robotin akselit samanaikaisesti päätepisteeseen. Liikenopeus määritellään prosentteina maksiminopeudesta. Nivelliike on nopein tapa robotille liikkua eri pisteiden välillä. Lineaarinen liike (linear) liikuttaa robottia vain suoraa rataa pitkin ja sen nopeus määritellään eri yksikössä, tyyppillisesti millimetreinä yhtä sekuntia kohti. Kolmas liike, ympyränkaariliike (circular), liikuttaa robotin kolmen pisteen kautta kulkevaa kaarta pitkin.

Liiketyyppi kertoo robotille, miten seuraavaan pisteeseen liikutaan. Esimerkiksi jos pisteeseen tullaan suoraa viivaa pitkin, liiketyyppinä tulisi käyttää lineaariliikettä. Ajettaessa robottia kaarta pitkin, liiketyyppinä tulisi olla ympyränkaariliike. Kuvassa 4 on esitetty liiketyyppejä Fanuc robottisolussa, jossa liiketapa voidaan valita kolmen perusliiketyypin väliltä. Kuvan 4 ohjelmointirivi kertoo lisäksi paikan tyyppiksi, P[1], ensimmäisen liikepisteen. Nopeus pisteeseen siirtymiselle on esitetty prosentteina (30%) maksiminopeudesta sekä paikoitustavan kerrotaan olevan tarkka, jatkuva liike (CNT10).

Paikoitustarkkuus voi olla tarkka piste, jossa robotti pysähtyy hetkeksi ennen kulkemista seuraavaan pisteeseen. Muussa tapauksessa käytetään jatkuvaa liikettä pisteestä toiseen, jossa pisteille annetaan erillinen tarkkuus, kuinka paljon robotti saa oikaista pisteiden välillä kulkiessaan ohjelmoidun liikeradan. Paikoitustarkkuuteen vaikuttaa myös itse robotin toistotarkkuus, jolla tarkoitetaan robotin tilastollista tarkkuutta palata takaisin opetettuun pisteeseen. Useimpien keskikokoisten teollisuusrobottien toistotarkkuus on $\pm 0,1-0,5$ millimetrin luokkaa kuormasta ja nopeudesta riippuen. Erityisesti kokoonpanoroboteilla, robotin on pystyttävä jopa $\pm 0,02$ millimetrin tarkkuuteen vaatimuksista riippuen.



Kuva 4. Liikkeen käskyt Fanuc robottisolussa.

Robotin ohjelmaa voidaan kontrolloida erilaisilla käskyillä. Yksinkertaisessa tapauksessa ohjelmoitua ohjelmaa toistetaan loputtomasti, robottisolun pysähtyessä käyttäjän toimesta tai virheilmoituksen johdosta. Erilaisiin käskyihin voidaan myös lisätä ehtoja, joissa robottia ohjataan tulo, lähtö tai rekisteriarvon perusteella.

Kuvassa 5 on esitetty yleisimpiä robotin ohjelmoinnissa käytettäviä käskyjä. Niitä on esimerkiksi:

- Hyppytoiminto (JUMP), jolloin riviä suorittaessa robotti hyppää ohjelmassa ennalta määrättyyn paikkaan.
- LABEL-käsky määrää paikan, johon hyppytoiminnolla siirrytään.
- Kutsutoiminto (CALL), jonka avulla voidaan suorittaa robotille määritettyjä aliohjelmaa. Aliohjelmaa tarvitaan usein erilaisten robotin työkalujen käyttämiseen ja ohjaamiseen.

- Ajastin (TIMER), jonka avulla robotti pysähtyy liikepisteeseen määrätyksi ajaksi sekunneissa.



Kuva 5. Motoman robottisolun käskyjä.

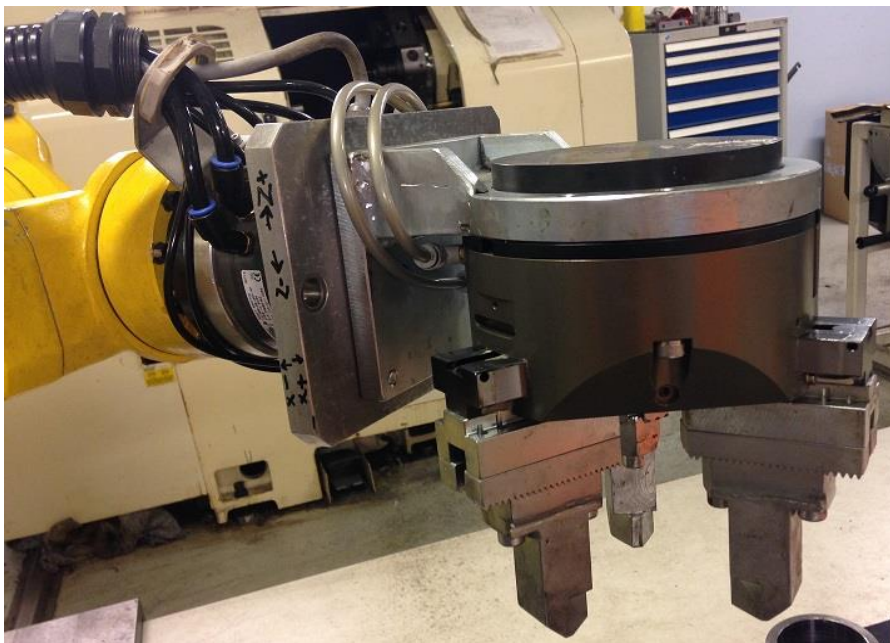
2.6 Tarraimet ja työkalut

Robotin työkalulla tarkoitetaan sen mekaanista osaa, jota robotti siirtää liikepisteestä toiseen. Tavallisin käytetty työkalu on jonkinlainen tarrain. Toinen työkaluryhmä on tiettyyn prosessiin osallistuvat työkalut, kuten hitsauspistooli tai maalausruisku. Työkalun ja sen tarraimen tulee olla mahdollisimman kevyt, jotta robotin kuorman käsittelykyky ei hupene pelkän tarttujan liikutteluun. Muita huomioitavia asioita suunnittelussa on soveltuvuus eri kappaleille, tartuntavoimien aiheuttamat mahdolliset muodonmuutokset, voimien välitys sekä tilavaatimukset.

Tarraimet voidaan jakaa erilaisiin ryhmiin, esimerkiksi karkeasti pneumaattisiin, hydraulisiin tai sähköisiin tarraimiin. Tällaisia voivat olla avautuvat ja sulkeutuvat tarraimet, kaksitai useampisormiset tarraimet, alipainetarraimet ja älykkäät anturoidut tarraimet.

2.6.1 Mekaaniset tarraimet

Mekaanisten tarrainten käyttövoimana on tyypillisesti sähkö, hydraulikka tai paineilma. Tarrain ottaa käsiteltävästä kappaleesta kiinni ja päästää irti vasta sille ohjatusta komennosta. Usein erilaisin mekanismehein pyritään matkimaan ihmisen sormien ja kouran toimintaa. Mekaanisen tarraimen rakenne voi perustua esimerkiksi epäkesko-, ruuvi-, nivel-, hammaspyörä- tai väkipyöräkinematiikkaan. Kuvassa 6 on esitetty Koneteknologiakeskuksen, Fanuc robottisolun vaihdettavissa oleva, sähköisesti ohjattu, paineilmatarrain. Tarraimella voidaan siirtää melko pieniä kappaleita robotin toiminta-alueella esimerkiksi työstökoneeseen koneistettavaksi.



Kuva 6. Sähköohjattu paineilmatarrain.

2.6.2 Imu- ja tyhjiötarttajat

Alipaineeseen perustuvia tartuntaelimiä käytetään sovelluksissa, joissa mekaanisen tarraimen käyttö on hankalaa. Imutartunnassa työkappaleeseen tartutaan yleensä vain yhdeltä suunnalta imukuppien ollessa samansuuntaisesti. (Kuva 7.) Kurviset tai muoviset

Imukupit eivät helposti naarmuta nostopintaa. Tartuntavoiman lisäys suurille kappaleille onnistuu lisäämällä imukuppeja. Usean imukupin järjestelmässä on turvallisuustekijänä huomattava, että yhdenkin imukupin irtoaminen aiheuttaa alipaineen häviämisen, mikä johtaa kappaleen irtoamiseen, ellei käytetä varolaitteita.

Imukupit vaativat yleensä riittävän tasaisen, sileän, puhtaan ja tiiviin pinnan. Imuvoima on paine-eron ja imupinta-alan tulo. Tarraimeen ei saa syntyä suuria, tarranta vastaan kohtisuoria sivuttaisvoimia, sillä tarraimen synnyttämä sivuttais-liikettä vastustava voima riippuu tarraimen ja kappaleen välisestä kitkakertoimesta. (Robotiikka 2016, 58.)

Imukupin etuina on yksinkertainen ja yleensä luotettava rakenne. Huonoina puolina voidaan pitää vaara- ja virhetilanteita, jos alipaine katoaa äkillisesti. Imukuppi ei myöskään pysty keskittämään kappaletta ilman lisälaitteita.



Kuva 7. Alipaineohjattu imukuppitarrain.

2.6.3 Robotissa käytettävät muut työkalut

Robotissa voidaan käyttää työstäviä työkaluja kuten jyrsintä tai hiomalaitetta. Työstettävien kappaleiden muoto- ja mitto-ongelmien aiheuttamaa epätarkkuuden ongelmaa pienennetään usein joustavien työkalujen avulla. Näin voimien kohdistuessa työkaluun, se joustaa ja voi olla jopa hyväksi jäljen tasoittamiseksi tai anturi-ohjauksen viiveiden poistamiseksi.

Tavallisia muita käytettyjä prosessityökaluja voivat olla erilaiset hitsauspääät, maalaus-suuttimet tai polttoleikkaimet.

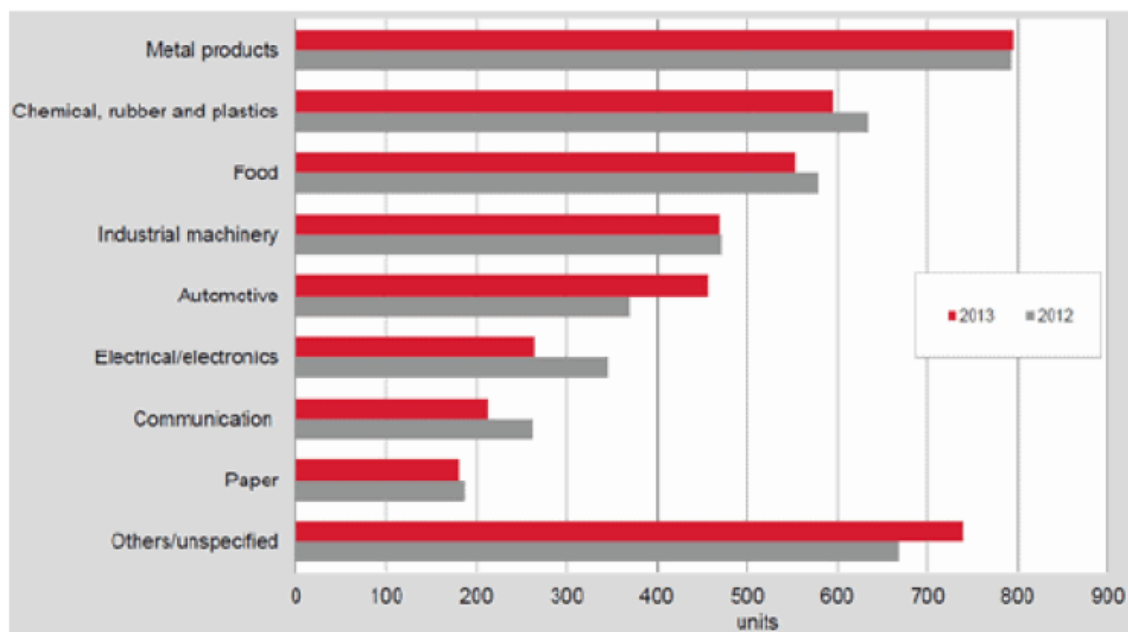
Robotissa voidaan käyttää myös magneettitarrainta erilaisille magneettisille aineille. Magneetin nostovoima riippuu kappaleen materiaalista, pinnanlaadusta, ilmaraosta ja muodosta. Tartunta on nopeaa, mutta magnetismi hidastaa irrotusta. Sähkömagneetilla voidaan kääntää magneetikentän suuntaa, jolloin irrotus tapahtuu nopeasti. Huonona puolena on sähkömagneetin lämpeneminen käytössä, joten työkierto on suunniteltava tarkasti ettei lämpötila pääse nousemaan liikaa.

Nykypäivänä anturoidut älytarraimet kasvattavat suosiotaan, koska niiden avulla voidaan estää turhat tuotannossa tapahtuvat katkokset sekä tarraimen tiedon perusteella pystytään tunnistamaan virhetilanteet. Älykkäässä tarraimessa toimilaitetta, mekanismia, työ-kappaletta ja ympäristöä valvotaan aistimin. Toimilaitteen ja mekanismin asentoa sekä nopeutta mittaamalla saadaan takaisinkytkentätietoa tarraimen sisäisestä tilasta. Ulkoista tilatietoa käytetään ohjamaan tarraimen ja koko robottijärjestelmän toimintaa. (Robotiikka 2016, 61.)

2.7 Teollisuusrobottien käyttökohteet

Robotiikalla on todellista potentiaalia luoda työpaikkoja, parantaa tuottavuutta ja turvallisuutta sekä parantaa vanhenevan väestön elämänlaatua. Robotiikka on jo nyt tärkeää monella suomalaisella teollisuudenalalla. Maamme työkoneteollisuus on kansainvälistä kärkeä automaatioteknologioitten hyödyntämisessä. (Robotiikka 2016, 4.) Suomessa robotteja käytetään eniten tilastojen mukaan metalli-, muovi- ja elintarviketeollisuudessa. Tavallisia robotisoinnin kohteita ovat myös paperi-, lääke-, ja autoteollisuus, joissa robotteja hyödynnetään yleisimmin kokoonpanotehtävissä, kappaleiden käsittelyssä, hitsauksessa tai muissa tuotannollisissa tehtävissä. (Kuva 8.)

Teollisuusrobotteja käytetään työstökeskuksien tai -koneiden, sorvien palvelutehtävissä työkalun vaihtajana tai kappaleiden panostajana. Robotti huolehtii työstettävien kappaleiden asettamisesta esim. sorvin leukoihin sekä niiden poistamisesta työvaiheen jälkeen. Yksi robotti voi palvella useita työstökoneita. Koska konepalvelujärjestelmä sisältää vähintäänkin kaksi eri konetta (robotin ja työstökoneen), on järjestelmän toimivuuden kannalta tärkeää, että koneet synkronoidusti työskentelevät keskenään. (Opinnäytetyö, Tommi Mikkonen, 2012, 28.)



Kuva 8. Suomen teollisuusrobottien lukumäärä tuotantotyypeittäin (Robotiikka 2016, 7).

3 ROBOTTIIKKAKURSSIN OHJAUS

Osana opinnäytetyötä robotiikkakurssin käytännön ohjaus on yksi vastuualoistani kuuden viikon ajan #Tehdas-oppimisympäristössä laboratoriotöissä. Tämä sisältää neljän eri opiskelijaryhmän opastuksen kahdelle eri robottisolutyöpisteelle kuuden tunnin ajan kerran viikossa. Ohjeiden sekä harjoitustehtävien laatimisen tukena harjoittelen useina päivinä viikossa robottisoluja ohjelmoiden. Ohjausosuus tuntuikin alkuun haastavimmalta, koska kokemusta oppilaiden ohjauksesta tai luottoa omien taitojen ja tietämyksen riittävyteen on lähes olematon. Kuitenkin ensimmäisten laboratoriotöiden ohjeistamisen jälkeen tehtävä alkoi tuntua luontevalta. Opiskelijoilta saatu suora palaute laboratoriotöiden jälkeen helpotti kehitystyötä, ja oli mukava kuulla opiskelijoilta itsessään, miten ohjeista ja tehtävien opastuksesta oli hyötyä heidän oppimisen edesauttamiseksi.

Oma osuus ohjauksessa koostui muun muassa ryhmän valvonnasta robottisolulla, onnistuneen tehtäväesimerkin näyttämisestä sekä opiskelijoiden perehtymisen aikana tehtävässä neuvomisesta ja ongelmatilanteissa eteenpäin auttamisesta. Ohjauksessa käytiin tehtävää vaihe vaiheelta läpi yhden ryhmän jäsenen toimiessa robottisolun operaattorina. Muut ryhmän jäsenet pyrkivät soveltamaan annettuja ohjeita sekä vaihtamaan operaattoria jäsenien välillä tasapuolisesti kaikkien kokemuksen kartuttamiseksi. Edistyneimpien ryhmien kohdalla saatoimme yhdessä kokeilla haastavampia lisätehtäviä robottisolulla ja perehtyä syvemmin robottisolulla suoritettaviin työtehtäviin Koneteknologiakeskuksen käytössä.

Käytännön taitoja ei voi oppia pelkästään lukemalla tai katsomalla. Työtaitojen oppimiseen tarvitaan myös tekemällä oppimista. Perusteiden oppimisen jälkeen kehittyminen tapahtuu tekemällä käytännön työtä. (Salakari 2007, 7.)

Opettajan rooli taitojen opetuksessa on alkuvaiheessa merkittävä. Esimerkiksi opetettaessa ympäristössä, jossa toimintaan konkreettisten asioiden, kuten lopputuotteen, valmistamisessa erityisen merkittäviä taitoja opettajan osalta ovat oppimisen ohjaus, palaute, arviointi, tavoitteiden asettaminen ja tiedon ja taidon siirtämisen edistäminen. (Salakari 2007, 76.)

Periaatteena demonstraatioissa on, että oppija jäljittelee opettajan esittämää suoritusta. Ongelmaksi voi tulla oppijoiden rohkeuden heikkeneminen, jos he eivät osaa suoritusta,

jonka opettaja on vaivattomasti heille esittänyt. Ratkaisuna opettaja voi jakaa suorituksen pienempiin osiin, jotka hän demonstroi vaihe vaiheelta. Suorituksen hidastaminen edistää myös oppijoiden mukana pysymistä. (Salakari 2007, 77.)

Eri ryhmien välillä, opetustavassa ja ohjaustyyliä oli myös jonkin verran eroja. Alkuun kiinnostusta vailla oleva ryhmä kaipasi enemmän konkreettista näyttöä ja usein nopeaa edistystä tehtävän perusvaiheiden yli esimerkiksi robottisolun työkalun käyttämiseen. Vasta tässä vaiheessa opetus hidastui ja kiinnostuneisuuden herätessä jakesittiin keskittyä alusta asti perusteiden läpikäymiseen, jotta mielenkiintoiseksi osoittautunut tehtävä saatiin suoritettua. Oma-aloitteiset ryhmät taas kuuntelivat ohjeistuksen ja malli esimerkiksi huolellisesti, jonka jälkeen he pyrkivät noudattamaan ja soveltamaan alusta asti ohjeita huolellisesti sekä ongelmatilanteissa selvittämään ratkaisunsa itse.

Vain riittävän harjoittelun myötä on mahdollista taitojen oppiminen. Taitojen kehittymisen kannalta on tärkeää saada kokeneen ammattilaisen tai opettajan palaute. Vain tekemällä voi oppia. (Salakari 2007, 78.)

3.1 Ohjauksessa käytetty laitteisto

Turun ammattikorkeakoulun #Tehdas-oppimisympäristössä on käytössä monipuolinen laite- ja konekanta, joka koostuu monesta eri konepajasovellutukseen käytettävästä, nykyaikaisesta solusta. Tässä osiossa on esitelty hieman laboratoriotöissä ohjeistettuja, opiskelijoiden käyttämiä soluja kurssin aikana. Käytettyjä robotteja on esitelty enemmän opiskelijoille laadituissa ohjeissa. Opiskelijoiden tehtävänä laboratoriotöissä on oppia robottisolun toiminnan perusteet, pohtia käyttökohteita sekä tutustua solun turvajärjestelmiin.

Motoman Robotics Finland Oy:n toimittaman robottisärmäysselun laitteisto koostuu Motoman UP50N-robotista ja Amada HFE 80–25-särmäyspuristimesta. Robotin työkalunvaihto-ominaisuuden ansiosta solulla voidaan käsitellä hyvin erikokoisia ja painoisia peltiosia. Solua käytetään erikokoisten peltilevyjen särmäykseen ja käsittelyyn. Robotti syöttää levyn imukupittarraimen avulla särmäyspuristimeen, joka työstää levyn haluttuun muotoon. Tämän jälkeen robotti ottaa lopputuotteen puristimesta ja siirtää sen pinoon/varastoon.

Särmäysselun turvalaitteet koostuvat törmäysanturista, nivelien ylitaitumisantureista, kuolleenmiehenkytkimestä sekä turva-aitauksesta. Törmäysanturi tunnistaa liian suuren

vastustavan voiman törmäyksessä. Robotti pysäyttää minkä tahansa käynnissä olevan tehtävän liikkeen vahinkojen estämiseksi.

Turun ammatti-instituutin sorvaussolu, joka koostuu Fanucin kappaleenkäsittelyrobotista, Nakamura-TOME TMC-20 NC-sorvista ja MCV-300-pystykaraisesta jrsimestä. Robotin työkalunvaihto-ominaisuuden ansiosta solulla voidaan käsitellä erikokoisia ja painoisia kappaleita. Solua käytetään pääosin opetustarkoitukseen robotin ohjelmointiin tutustumisessa, mutta solulla voidaan panostaa metallikappaleita NC-sorville. Robotti syöttää kappaleen mekaanisen tarraimen avulla sorviin, joka työstää kappaleen asetettujen parametrien mukaisesti.

Turvajärjestelmään kuuluu ohjaimen kuolleenmiehenkytkin ja hätäseis-katkaisija käsiohjaimessa sekä ohjainlaitteessa. Turvapiiri koostuu liukuoven fyysisestä rajakytkimestä sekä robotin törmäysanturista.

4 KEHITYSTYÖHÖN LIITTYVÄÄN MATERIAALIIN TUTUSTUMINEN

Ennen robottilaboratoriotuntien ohjastusta piti tutustua oppilaitoksesta ja #Tehdas ympäristöstä löytyvään robottimateriaaliin. #Tehtaalta löytyi robottivalmistajan ja -toimittajan mukana tulleita kansioita ja lehtiöitä. Tutustuin myös verkossa oleviin Turun AMK:n robotiikan työohjeisiin, jotka on tarkoitettu opiskelijoiden käytettäväksi mobiililaitteilla laboraoiden aikana qr-koodien kautta.

Yksi suurimmista ongelmista nousi tässä vaiheessa esiin, koska usein kansioista puuttivat ohjelmoinnin perusteista kertovat osiot kokonaan tai osa oli hävinnyt ajansaatossa. Kansiot ovat myös valtavan järeitä, täynnä soveltavaa ja edistynyttä käyttökoulutusta vaativaa tasoa. Robotin ohjelmoinnin perusasioihin perehtynyt tai vailla kokemusta oleva henkilö ei pystyisi käyttämään tietoa sellaisenaan ilman lisä tiedonhakua.

Ollessani yhteydessä henkilökuntaan sekä Sakari Koivuseen, löytyi tulostettava versio uudemman Fanuc-robottisolun käyttöoppaasta. Tästä ei suoranaisesti ollut hyötyä perusteiden sekä yksinkertaisten ohjelmien laatimiseen vanhempaan soluun, puhumattaakaan Motoman-särmäysselästä. Solun käynnistämiseen sekä operointipaneelin ja peruskäskyjen käyttämiseen siitä sai vinkkejä. Muuta aineistoa manuaalin tekoon kyllä löytyi eri puolelta robottisoluja, mutta se oli pahasti hajallaan eri kansioissa ja siksi vaikeasti hyödynnettävissä opetuskäyttöön.

Myöhemmin käyttöoppaan perusteita kasatessani eri lähteistä sain käsiini Fanuc robotiikkurssi 1 ja 2 ohjeet, joista sai hyviä vinkkejä robotin liikuttamisesta aina perus liikeohjelmointiin asti. Näissä ohjeissa oli käsitelty myös turvallisuutta, robotin työkaluja sekä ohjauskaapin toimintoja. Nämäkin ohjeet olivat suunniteltu uudemmalle Fanuc solulle, jolloin hyvät, havainnollistavat ohjelmakoodi kaappauskuvat näyttivät aivan erilaisilta kuin käsiohjaimessani ollut ohjelmakoodin pätkä. Näistä ohjeista oli kuitenkin eniten hyötyä tässä vaiheessa, olivathan ne sentään saman valmistajan ohjeet.

Toiseen, Motoman särmäysselästä, löytyi ohjauskaapin päältä erilaisia kansioita käyttöohjeineen. Materiaalia tutkiessani kävi kuitenkin ilmi, että suuriosa aineistosta käsitteli itse särmäyspuristinta ja sen käyttöä. Puristin ei ollut opiskelijoille opetettavien asioiden joukossa, joten lähes kaikki aineisto siitä voitiin sivuuttaa. Robottisolun käyttöönottoon ja

käsiohjaimen käyttöön löytyi materiaalia toisesta Motoman solusta, jota pystyttiin suoraan hyödyntämään ohjeiden teossa.

Lisäksi haastateltiin Koneteknologiakeskus Turku Oy:n muutamaa avuliasta, robottikoulutusta vetävää henkilöä mahdollisen materiaalin saamiseksi. Yksi henkilökunnan jäsenistä jakoikin tietämystään robotin käytöstä ja ohjelmoinnista, sekä vinkkejä mahdollisiin opiskelijoille tehtäviin harjoituksiin kyseisellä särmäyssolulla. Henkilökunnalta saatu palaute nykyisestä materiaalista kasvatti entisestään ajatustani opetusmateriaalin keräämisestä yhteen, helposti luettavaan ja saatavilla olevaan ohjekirjaan.

Lopuksi mietittiin yhdessä kurssin vetäjän, Sakari Koivusen kanssa mahdollisista robotiikan harjoitustöiden sisällöstä, sekä ohjelmoinnin tason laajuudesta opiskelijoille. Kurssin sisältö oli sen verran vapaa, että sain itse omien opiskelukokemusten perusteella valita robottisoluilla tehtävät harjoitukset ja kehittää niitä kurssin edetessä.

5 OPPIMATERIAALIN KEHITTÄMINEN

Robottisoluilla työskentely on suoraa konkreettista tekemistä, joka jäljittelee suoritusta oikeassa työympäristössä. Opiskelijat ovat saaneet teoreettisen tietämyksen robotiikan lähitunneilla ja pääsivät nyt hyödyntämään taitojaan käytännön oppimisympäristössä eri sovellutuksissa. Työympäristössä on tärkeää taitoja oppiessa muodostaa ohjaajan esimerkkisuorituksesta malliin perustuva kuva ja opiskelijan suoritus jäljittelee saatua mallia. Tässä tapauksessa malli työn suorittamiseen saadaan opiskelija-ohjaajalta, jolloin on tärkeää laatia opiskelijalle hänen taitojaan vastaava harjoitusmalli. Suurimman osan käyttäytymisestään ihminen oppii havainnoimalla malleja. Havainnointi eli mallioppimiseen perustuva oppiminen muodostuu neljästä prosessista, jotka voidaan jakaa alaprosesseihin:

- 1) huomion kiinnittäminen
- 2) muistaminen
- 3) motoristinen toistaminen
- 4) motivationaalinen

Oppiakseen havainnoimalla on kiinnitettävä huomiota tiettyihin toiminnan merkittäviin piirteisiin. Se, mitä asiaa havainnoidaan, määritetään huomion kiinnittämällä. Oppimisen kannalta osa havainnoista on olennaisempia kuin toiset. Opimme kohtaamalla asioita niissä asiayhteyksissä kuin ne ilmenevät. Oppimiseen vaikuttaa kykymme käsitellä tietoa. Aiemmat kokemukset ja taustamme vaikuttavat siihen, miten tulkitsemme huomioimamme ja huomioimme asioita. (Salakari 2007, 49.)

Tästä syystä on tärkeää saada aina ensikosketuksesta alkaen opiskelijan kiinnostus aiheeseen oikeanlaiseksi, koska myöhemmässä vaiheessa aikaisemmat kokemukset vaikuttavat, kuinka mieleiseltä robotiikka tuntuu. Siksi robottisoluilla tehtävien harjoitusten onnistumista ei voi liikaa korostaa. Olennainen osa onnistumiseen on selkeä ohjeistus ja olennaisen tiedon helppo saatavuus käytettävän ajan puitteissa.

5.1 Opetuksen suunnittelun vaiheet ja tasot

Opetuksen suunnittelun vaiheet ovat:

- 1) koulutustarpeen arviointi
- 2) oppija-analyysi
- 3) oppimistavoitteiden asettaminen
- 4) oppimissisältöjen valinta ja jäsenys
- 5) työtavat ja opetusmenetelmät
- 6) oppimisen ja opetuksen arviointi

Koulutustarpeen arviointi riippuu siitä, millaiset tavoitteet koulutukselle asetetaan. Kouluttajan on määriteltävä oppijoiden oppimistasot koulutuksen tavoitteisiin sekä tunnettava oppijoiden osaamisen tasot. Oppija-analyysissä selvitetään, mitä oppijat osaavat tai tietävät jo. Kiinnostuksen kohteet, taustat ja ikä vaikuttavat siihen, mitä oppijat jo osaavat ja miten opetusta on parasta toteuttaa. (Salakari 2007, 181.)

5.2 Olemassa olevan materiaalin hyödyntäminen kehittämistyössä

Kuten jo aikaisemmin mainittu, kehitetyt laboratorioharjoitukset perustuvat omiin, aikaisemmin saatuihin opiskelukokemuksiin robotiikan kursseista. Eri kansioista kerätty tieto, opitut taidot ja henkilökunnalta saatu pikainen robottisolun ohjelmointikoulutus on luotujen ohjeiden ja laadittujen harjoitustöiden taustalla. Myös omasta, vanhasta robotiikkakurssin raporttien palauteosiosta sai hyviä vinkkejä kehitystyöhön.

Materiaalia alettiin kehittää ensimmäisten ohjattujen laboratoriokertojen jälkeen, johon olin laatinut alustavat harjoitustyöt ajankulun sekä opiskelijoiden taitotason seuraukseksi. Ensimmäisten kertojen aikana huomasin, kuinka puutteelliset ohjeet saivat yksinkertaisen harjoituksen tuntumaan todella aikaa vievältä pitkien tiedonhaku taukojen

takia. Myös itsenäisen tekemisen ja opiskelun määrä oli vähäistä opiskelijoiden keskuudessa, koska seurattavia ohjeita harjoituksen suorittamiseen ei vielä ollut. Tämän takia ohjaajana olemisen rooli korostui entisestään positiivisen kuvan saamiseksi opiskelijoille.

Seuraavalla laboratorioskerralla olin saanut julkaistua wordpressiin opiskelijoiden saataville oleellisimpien vanhojen materiaalien pohjalta omat ohjeet molempiin (Motoman ja Fanuc) ohjeistettaviin robottisoluihin koskien käynnistystä, aloitustoimenpiteitä sekä perus ohjelmointia. Näiden ohjeiden pohjalle ruvettiin tekemään parannuksia jo samana laboratorio kertana opiskelijoilta saadun palautteen perusteella. Havainnot ensimmäisistä ohjeista olivat seuraavanlaisia:

- Hyvä johdanto ja robottisolun käyttöönotto-osio
- Liikaa vaikeaselkoista, teknistä tekstiä
- Turvallisuus ja työkaluosion puute
- Huonoja, vanhoja kuvia
- Turhan laaja harjoitustyötä silmällä pitäen
- Ei kaikkea tarvittavaa tietoa itsenäiseen toimintaan pystyäkseen

Materiaalin ja harjoitustöiden perusteella kehitys jatkui seuraavan kertaan miettimällä harjoitusta itse tehdessä, mitkä vaiheet tulisi kertoa suoraan opiskelijalle, ja missä asioissa vaadittaisiin oma-aloitteisuutta sekä taitojen ja ohjeiden soveltamista. Kerta kerralta ohjeita päivittäessä ja oman ohjauksen parantuessa palaute viimeisempiä laboratorioskertoja kohden alkoi olla erittäin positiivista ja osa jopa kaipasi vaikeampia harjoituksia. Palautetta ja yhteenvetoa kehitystyöstä on käsitelty enemmän seuraavassa luvussa.

6 YHTEENVETO JA POHDINTAA

Työtä aloittaessani suurimpia haasteita oli, ettei aikaisempaa kokemusta opiskelijoiden ohjaamisesta ollut lainkaan eikä luottoa omiin robotiikan taitoihin ollut riittävästi. Tärkeimpänä lähtökohtana olikin oman robotin ohjelmointitietämyksen lisääminen sekä insinöörin johtamistaitojen kehittäminen.

Kehitystyön aluksi tutustuin robotiikan materiaaliin ja kertosin opittuja taitoja robotiikasta ja ohjelmoinnista aiemmilta kursseilta. Selvitin ohjattavien robottisolujen toimintaa ja käyttökohteita Koneteknologiakeskuksella, sekä aloin valmistella materiaalia, josta saisi kasattua ohjeistusta ja harjoituksia opiskelijoille.

Mielestäni onnistuin hyvin ohjaajana ja pystyin parantamaan ulospäin suuntautumistani opiskelijoita kohtaan kerta kerran jälkeen, josta saatu palaute tukeekin hyvin omaa mielikuvaani. Opiskelijat olivat pääsääntöisesti innoissaan mukana harjoitustöissä ja kohtelivat minua yhtenä henkilökunnan jäsenenä. Oppilaat uskalsivat kysyä ja kertoa hyvin ongelmatilanteista, eikä yksikään ryhmä joutunut keskeyttämään harjoitusta. Kaikki ryhmät selvisivät harjoituksista hienosti ja kaiken kaikkiaan projektista jäi todella positiivinen tunne. Loppujen lopuksi hienointa projektissa olikin ohjaajana toimiminen ja vasta sen jälkeen harjoitusten laatiminen sekä itse robottisolun ohjelmointi.

Projektista jäi valtavasti oppia ja kokemusta kuuden viikon aikana itselleni. Varsinkin oma innostus ja tietämys robotiikasta lisääntyi sekä johtajataidot ja vastuunotto kyky paineessa lisääntyi. Saamani palaute oppilailta harjoitusten ohjauksesta ja robottisoluihin suunnitelluista ohjeista oli rohkaisevaa ja innoittavaa.

6.1 Oppimateriaalin ja ohjeiden kehittäminen tulevaisuudessa

Käyttöohjeet eivät missään nimessä ole vielä valmiit vaan käytössä olevat ohjeet toimivat pohjana ja perustana tulevaisuuden kehitystä ajatellen. Ohjeisiin on saatava lisää harjoitustöitä ja perusteellisempi osuus työkalujen vaihtomahdollisuudesta sekä turvallisuudesta. Harjoitustöiden laatua tulisi kehittää yhdessä opiskelijoiden ja opettajien kesken jatkossakin. Yhteistyötä vanhempien, robotiikkaan erikoistuneiden opiskelijoiden kanssa kannattaisi jatkossakin hyödyntää. Näin erikoistuvien, tulevien insinöörien johtamis- ja robotiikkataidot pääsisivät testiin työelämää ajatellen.

6.2 Palaute ja kehittämistyö

Huonoa palautetta ei ole, kunhan se on kehittävää ja rakentavaa. Loppuun olen kerännyt suoraan opiskelijoilta saamaani palautetta raporttien ja suullisen kanssakäymisen pohjalta. Mainittakoon, että pääasiallisesti palaute oli positiivista ja osa opiskelijoista oli sitä mieltä, ettei muuttaisi mitään opetuksessa tai ohjeissa. Palaute myös parani laboratorio-kertojen edetessä, kun harjoitustöitä ja ohjeita päästiin kehittämään opiskelijoiden tarpeita paremmin vastaaviksi. Itse pystyn tunnistamaan muutamia kehityksenkohteita, varsinkin ohjeistuksen laadussa sekä ajankäytön tehostamisessa.

Jotta opiskelijat ehtisivät perehtyä ja tehdä mahdollisimman vaativia harjoituksia ajan puitteissa, ohjeistus ja ennakotehtävä tulisi jakaa etukäteen tutustuttavaksi. Konkreettinen tekeminen pääsisi alkamaan suoraan ryhmän päästyä harjoituspisteelle robottisolulle ilman ylimääräistä ihmettelyä ja seisoskelua. Harjoitustehtäviä tulisi myös olla enemmän kuin yksi harjoituspistettä kohden. Näin kokemusta omaavat ja muuten nopeat ryhmät pääsisivät perehtymään robotiikkaan syvemmin ja tekemään solulla haastavampaa, työelämässä tarvittavaa ohjelmointia. Luppoaika ja peukaloiden pyörittely kitkettäisiin pois, ja saataisiin täysi hyöty käytännön harjoituksista ammattikorkeakoulun opiskelijoille.

Palaute:

”Mielestämme meillä oli hyvä opastaja sekä netistä löytyneet ohjeet olivat kattavia ja avuksi työssä.”

”Suullinen opastus on toimiva tapa, mutta jotkin asiat eivät jää siltikään muistiin. Kirjalliset ohjeet Optimaan tai vastaavaan auttaisi.”

”Toimiva harjoitus tuotantoautomaation perusteiden kurssille. Harjoitukseen voisi mahdollisesti sisällyttää Amada HFE 80-25 käyttöä.”

”Tässä labratyössä oli hyvää se, että ohjaaja osasi hyvin neuvoa ja opastaa robotin käyttöön.”

”Ongelmaksi asettui ohjauksen puute. Ohjaajamme meni välillä auttamaan toista ryhmää, jonka takia ajauduimme seinään robotin ohjaamisessa. Ongelmaa helpottaisi lyhet ohjeet, miten tietyt komennot toimivat.”

”On kätevää, että ainakin osa ohjeista on myös Wordpressissä sähköisessä muodossa, mutta työpisteellä A4-kokoiset ohjeet ovat huomattavasti paremmat.”

”Kehittämiskohteena näkisimme robotin ja särmäimen yhteiskäyttöä. Työkalunvaihdon voisi sisällyttää myös labrakertaan, jotta saataisiin enemmän sisältöä. Mielestämme sisältö oli aika laiha, eikä uutta juuri tullutkaan.”

”Särmäyskoneeseen olisi ollut mukava päästä myös tutustumaan, koska aikaa jäi jonkun verran ylimääräistä. Epäselvyyksiä ei juurikaan jäänyt, sillä toiminta oli suureksi osaksi jo opitun kertausta.”

”Ohjeet roboturku.wordpress – osoitteessa olivat toimivat ja ohjaajan tuella työstä selvitettiin kunnialla.”

”Ohjeet olivat erittäin toimivat emmekä lähtisi niitä mitenkään muokkaamaan. Annetusta nettiosoitteesta oli hyvä seurata projektin vaiheita.”

Projektissa onnistuttiin luomaan vaadittavat ohjeet ja ohjeistamaan laboratorioharjoitukset. Palautteen perusteella opiskelijat varmasti ottavat innolla vastaan myöhemmässä vaiheessa mahdollisen Robotiikan jatkokurssin, koska jo nyt harjoituksia olisi haluttu viedä pidemmälle. Kokonaisuudessaan opinnäytetyössä saavutettiin onnistuneesti kaikki asetetut tavoitteet.

LÄHTEET

ABB robotin käyttöönottoasennus ja ohjelmointi konenäkösovelluksessa. Tomi Haapalainen, Opinnäytetyö (AMK). [Online] https://www.theseus.fi/bitstream/handle/10024/33889/Haapalainen_Tomi.pdf?sequence=1

Human-Robot Interaction [Online] http://automation.tkk.fi/attach/AS-84-3147/Human_Robot_Interaction_caseWopa.pdf

ISO 8373:2012 Robots and robotic devices. Viitattu 23.1.2017

KIDE 2014. Robotiikka. [Online] http://kide.metropolia.fi/?page_id=85#sivu1

Koneteknologiakeskus Turku Oy [Online] <http://www.koneteknologiakeskus.fi/>

Kuivanen, R. (Toim) 1999. Robotiikka. Vantaa: Talentum Oyj/ Metallitekniikka. Viitattu 24.1.2017

Robottisolun suunnittelu – Piikkiö Works. Juho Rantala & Kim Kaski, Opinnäytetyö (AMK). [Online] https://publications.theseus.fi/bitstream/handle/10024/121897/Rantala_Juho_ja_%20Kaski%20Kim.pdf?sequence=1

Robotiikka sisällysluettelo – PDF, Tiina Halonen. [Online] <http://docplayer.fi/1226752-Robotiikka-sisallysluettelo.html>

Robotiikka, 2016. [Online] http://miniweb.lpt.fi/automaatio/opetus/luennot/pdf_tiedostot/Robotiikka_yleinen.pdf Viitattu 3.4.2017.

Salakari, H. 2007. Taitojen opetus. Saarijärvi; Saarijärven Offset. Viitattu 10.1.2017

Uuden robottisolun ohjelmointitavan mahdollisuudet pienille ja keskisuurille yrityksille, Esko Pekkanen, Lappeenrannan teknillinen yliopisto. [Online] <https://www.doria.fi/bitstream/handle/10024/63184/nbnfi-fe201007282240.pdf?sequence=3>

Wikipedia, Telerobotiikka [Online] <https://en.wikipedia.org/wiki/Telerobotics>

LIITTEET

Liite 1. Särmäyssolun Motoman-robotti.
Liite 2. Nakamura-solun Fanuc-robotti.

Säyrmäysselun Motoman-robotti

Motoman UP50N robotti



- 6-akselinen
- max kuorma 50kg
- ulottuvuus max 2046mm
- toistotarkkuus +- 0,07mm
- työkalunvaihtoasema
- pyöritysjalka VST – 600 SX

OHJEET

1. Käynnistäminen

Robottisolu käynnistetään kääntämällä pääkytkin ON-asentoon.



Paineilmat tulee laittaa myös päälle punaisesta venttiilistä, jotta imukuppitarraimen käyttö olisi mahdollista.



Särmäimen lähellä oleva operaattoripaneeli tulee kääntää niin, että se koskettaa anturia. Muuten robotin liikuttaminen ei ole mahdollista.



Robotin käsiohjaimesta painetaan SERVO ON READY -painiketta, jolloin vihreän led-valon tulisi syttyä kapulaan. Nyt robottia voidaan ajaa käsiajolla.

2. Robotin liikuttaminen käsiajolla

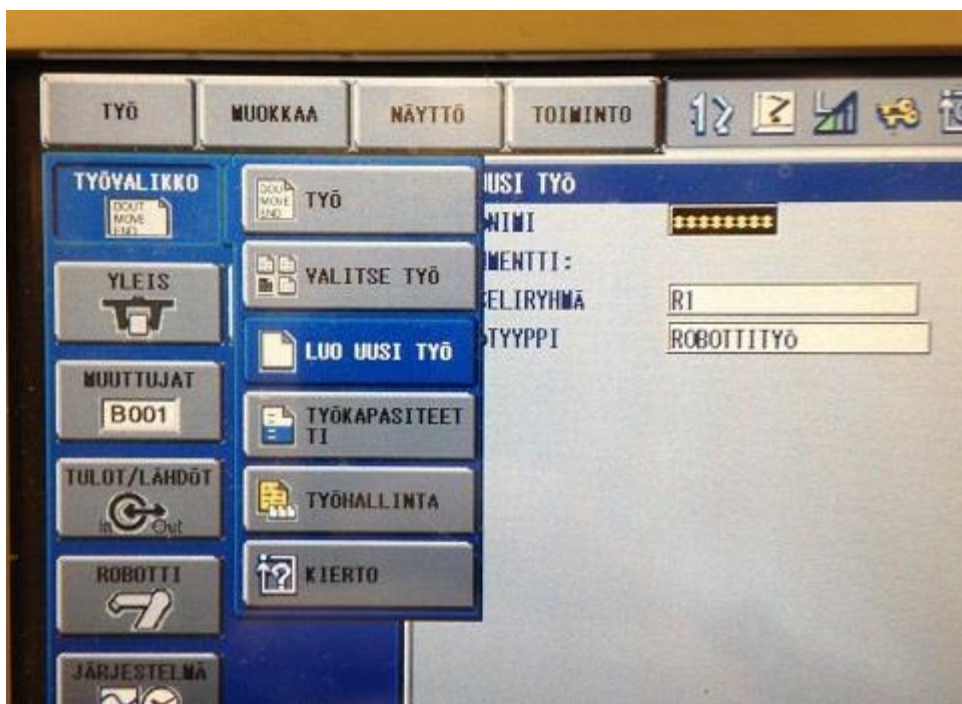
Kun tarvittavat käynnistys -vaiheet on suoritettu ja hätä-seis -piiri on kuitattu, voi robottia liikuttaa käsiajolla. Kapulasta tulee varmistaa, että Teach-tila on kytkettyä vasemmasta yläkulmasta. Lisäksi käsiohjaimen selkäpuolella oleva turvakytin pitää olla painettu puoliväliin ja mahdolliset virheet tulee kuitata CANCEL -näppäimestä.

Robottia liikutetaan painamalla mustia x-, y- ja z-käsiajonäppäimiä. Robotin liikkumisnopeus näkyy näytön yläreunassa ja sitä voi säätää käsiohjaimen FAST ja SLOW -näppäimistä.



3. Ohjelman valinta/uuden ohjelman luominen

Uusi ohjelma luodaan koskettamalla ruudusta työvalikko -> luo uusi työ. Työlle annetaan nimi painamalla SELECT -näppäintä, jolloin nimeäminen on mahdollista. Akseliryhmää tai työtyyppiä ei tarvitse muuttaa.

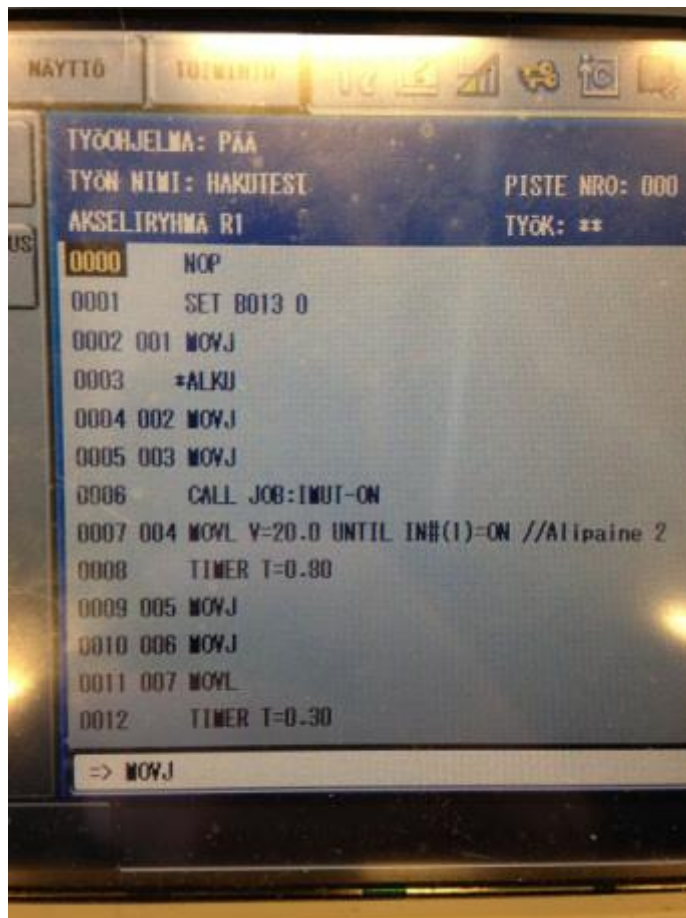


4. Liikepisteet

Robotin liikeohjelma tehdään opettamalla robotille joukko pisteitä. Robotti liikutetaan käsiajolla tiettyyn pisteeseen ja tallennetaan sijainti uudeksi riviksi ohjelmaan. Liikepiste tallennetaan painamalla ENTER -näppäintä.

Piste tallentuu oletusarvosta riippuen yleensä JOINT -liiketyypiksi, jolloin tallennettu piste ilmestyy rivinä muotoon: 0001 001 MOVJ.

Liiketyyppiä voidaan vaihtaa ennen tallennusta MOTION TYPE-näppäimestä.

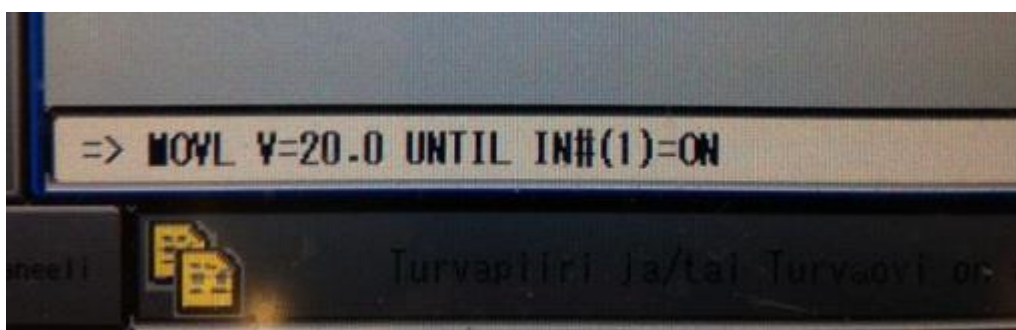


Liikepisteitä on kolme perustyyppiä:

1. **Nivelliike MOVJ** liikuttaa robotin akselit samanaikaisesti päätepisteeseen. Liikenopeus määritellään prosentteina maksiminopeudesta. Nivelliikkessä robotti kulkee pisteestä toiseen nopeinta mahdollista reittiä pitkin.
2. **Lineaarinen liike MOVL** liikuttaa robottia suoraa rataa pitkin. Liikenopeus voidaan määrittellä useassa eri yksikössä, tyypillisesti mm/s.
3. **Ympyränkaari MOVJ** liikuttaa robotin kolmen pisteen kautta kulkevaa kaarta pitkin. Liikenopeus määritellään kuten lineaariliikkeessä, useista eri vaihtoehdoista on tyypillisesti käytössä mm/s. Kaaren ensimmäinen piste voi olla tyypiltään nivel- tai lineaariliike. Liiketyyppi kertoo, miten *tähän* pisteeseen liikutaan. Jos pisteeseen tullaan suoraa viivaa pitkin, pitää liiketyypiksi valita lineaariliike. Jos pisteeseen tullaan kaarta pitkin, on pisteen liiketyyppinä ympyränkaariliike.

5. Peruskäskyt ja -toiminnot

Tallennetun pisteen liiketyyppiä ja nopeutta voidaan muokata valitsemalla piste menemällä pisteen (esim MOVJ) kohdan päälle ja painamalla SELECT -näppäintä. Liiketyypin voi vaihtaa pikanäppäimellä SHIFT + NUOLI ALAS tai painamalla uudestaan SELECT ja muuttaen liiketyyppiä. Myös nopeutta voidaan vaihtaa tästä valikosta. Halutut muutokset kuitataan ENTERillä. Alhaalla näytössä näkyy tämänhetkiset muokkaukset, johon muotoon piste tallentuu jos muutokset kuittaa uudestaan ENTERillä.



Tässä tallennettu piste MOVJ on muokattu MOVL -liiketyyppiin ja nopeus V on vaihdettu 20mm/s sekä pisteeseen otettu UNTIL lisämääritys käyttöön. Pisteitä voidaan ns. ajaa rivi riviltä eteenpäin ohjelmassa käsiäjäällä painamalla SHIFT + FWD (forward) tai taaksepäin SHIFT + BWD (backward) servojen ollessa päällä (selkäpuolella oleva turvakytkin puoleen väliin).

Huomioi robotin nopeus mahdollisten törmäysten välttämiseksi!

Suurinosa toiminnoista tai käskyistä saadaan otettua käyttöön INFORM LIST -näppäimen kautta. Näppäintä painettaessa uusi valikko aukeaa muokattavan työn päälle, josta voidaan valita tarvittavat toiminnot, joko kosketusnäyttöä käyttäen tai nuolinäppäimiä ja SELECTiä käyttäen. Harjoituksessa tarvittavia käskyjä ovat KONTROL. alasvetovalikosta JUMP, CALL, TIMER ja LABEL sekä MATEMAT alasvetovalikosta INC ja SET.

Valikko suljetaan painamalla INFORM LIST uudelleen.

JUMP= hyppytoiminto, jolloin riviä suorittaessa hyppää ennaltamäärättyyn paikkaan ohjelmassa.

LABEL= määrää paikan, johon JUMP -toiminnolla hypätään.

CALL= ns. kutsu toiminto, jonka avulla voidaan suorittaa aliohjelmiä. Tarvitaan imukuppitarraimen käyttämiseksi. IMUT-ON -aliohjelma kytkee alipaineen päälle kun taas IMUT-OFF -ohjelma kytkee paineet pois.

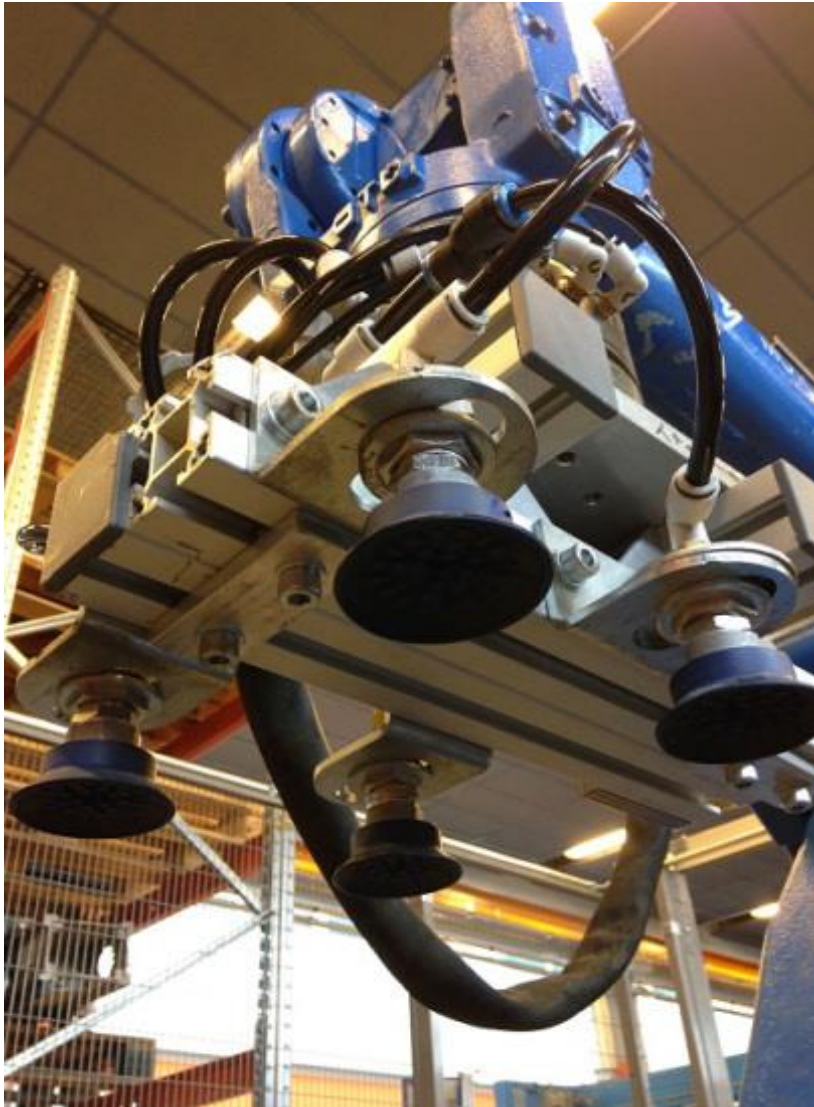
TIMER= ajastin, jonka avulla robotti saadaan pysähtymään pisteeseen määrätyksi ajaksi sekunneissa. Esim. tarvitaan imukuppitarrainta käytettäessä.

INC= lisäys/kerroin, jonka avulla laskuriin saadaan lisättyä ohjelmakiertojen lukumäärä.

SET= asetus, jolla voidaan nollata esim. laskuri.



6. Työkalut, imukupitarrain



Nakamura-solun Fanuc-robotti

Fanuc M710i robotti



- 6-akselinen
- ohjaus Fanuc system R-J3
- liukujalusta
- max kuorma 45kg
- max ulottuvuus 1706mm
- toistotarkkuus +/- 0,15mm

OHJEET

1. Käynnistäminen

Varmistetaan, että ohjainkaapin HÄTÄ-SEIS on vapautettuna. Robottisolu käynnistetään kääntämällä pääkytkin ON-asentoon. Tämän jälkeen painetaan vihreää ON-nappulaa. Robottisolu kytkeytyy päälle. Käsiohjaimesta tulee vapauttaa HÄTÄ-SEIS -painike.



2. Robotin liikuttaminen käsiajolla

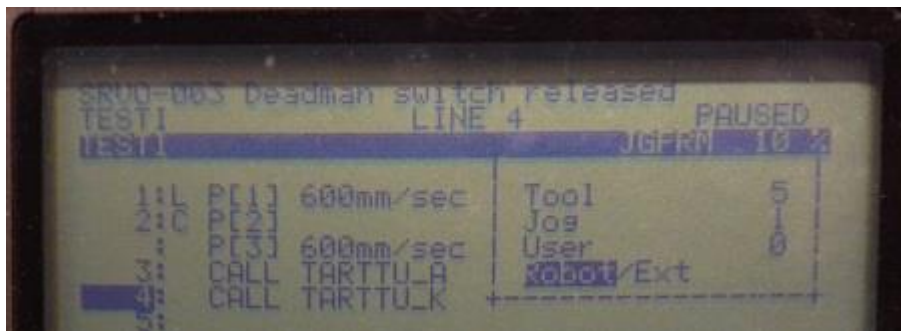
Robottia ei pysty liikuttamaan ennen kuin pitokytkintä on painettu kevyesti käsiohjaimen takapuolelta. Kytkin on kolmeosainen, joten sitä on pidettävä puolessa välissä servojen päälle kytkimiseksi. Mahdolliset virheet tulee kuitata RESET -painikkeesta.

(Jos virheitä jää edelleen näkyviin eikä liikuttaminen ole mahdollista, siirry Ohjeissa kohtaan **6. Virheilmoitukset ja ongelmat**).

Robottia liikutetaan painamalla SHIFT yhtäaikaan yhdessä mustien x-, y- ja z-käsiajonäppäimien kanssa. Robotin liikkumisnopeus näkyy näytön oikeassa yläreunassa prosentteina ja sitä voi säätää käsiohjaimen +-prosentti -näppäimistä. Koska robotti liikkuu eritavoin akseleidensa ympäri eri käyttäjäkoordinaatistoissa, niitä voi kokeilla ja vaihtaa COORD -näppäimestä.



Robotin paikkaa on mahdollista liikuttaa liukujalustaa pitkin. Tällöin koordinaatistona pitää olla nivelkoordinaatisto, JOINT. Seuraavanlaisen valikon saa auki painamalla SHIFT + COORD, jolloin voidaan valita robotin ja jalustan väliltä:



Täältä valikosta valitaan Robot -vaihtoehdon sijaan Ext, joka vastaa liukujalustaa. Nyt robottia voidaan liikuttaa paikasta toiseen jalustaa pitkin. Oikean paikan saavuttamisen jälkeen vaihdetaan Ext takaisin Robot asentoon. Valikon saa pois painamalla SHIFT + COORD.

3. Ohjelman valinta/uuden ohjelman luominen

Uusi ohjelma luodaan SELECT-näppäimestä avautuvassa valikossa. Painetaan F2, CREATE, jolloin avautuu nimeämisvalikko. Täällä ohjelma voidaan nimetä valitsemalla nuolinäppäimin joko UPPER CASE tai LOWER CASE ja kirjoittamalla F1-F5 nappuloita käyttäen.

Uusi, tyhjä ohjelma avautuu kuittaamalla annettu nimi ENTERillä. Seuraavaksi lisätään tyhjiä rivejä ohjelmaan, jotta liikepisteiden ja käskyjen antaminen olisi mahdollista. Tämä tapahtuu painamalla F5 näytöstä [EDCMD], mikä lukee näytön oikeassa alareunassa. Jos näytössä lukee TOUCHUP F5 näppäimen kohdalla, vaihda sivua NEXT-näppäimestä.

[EDCMD]n valittuasi ota kohta 1 Insert, jolloin näytössä tulisi lukea teksti ”How many line to insert?”. Anna jokin luku väliltä 20-60, jolloin rivit eivät lopu heti kesken. Kuittaa annettu lukumäärä ENTERillä ja nyt rivit on lisätty ohjelmaan.

Liikepisteiden ja käskyjen lisääminen ohjelmaan on nyt mahdollista.

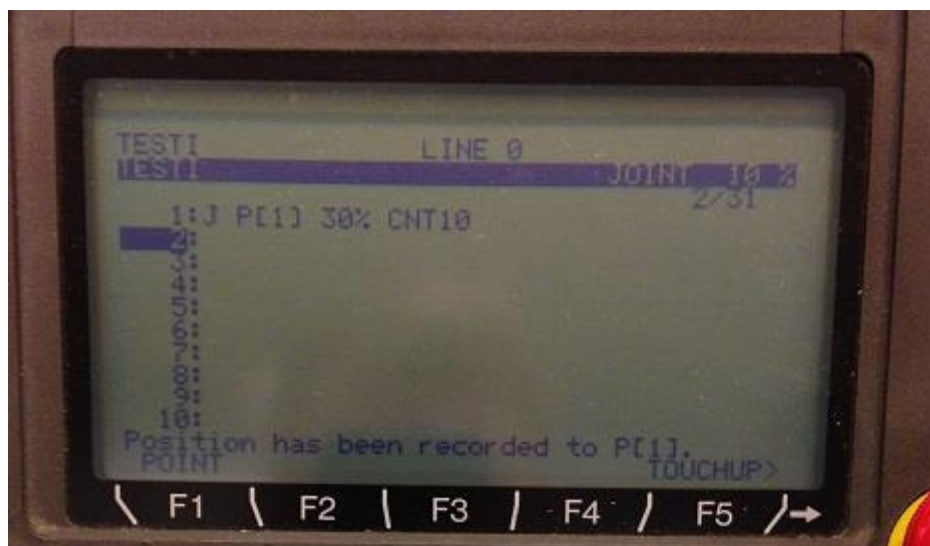
4. Liikepisteet

Robotin liikeohjelma tehdään opettamalla robotille joukko pisteitä. Robotti liikutetaan käsiajolla tiettyyn pisteeseen ja tallennetaan sijainti uudeksi riviksi ohjelmaan.

Aloita tekemällä äsken luotuun ohjelmaan liikepiste. Funktionäppäin F1 (POINT) avaa valikon, josta voit valita pisteen liiketyypin. Valitse nivelliike eli

J. Vaihtoehtoisesti voit käyttää pikanäppäintä F1 (POINT) + SHIFT, jolloin piste tallentuu asetetun oletusarvojen mukaan.

Ohjelmarivin tulisi näyttää suurinpiirtein seuraavalta:

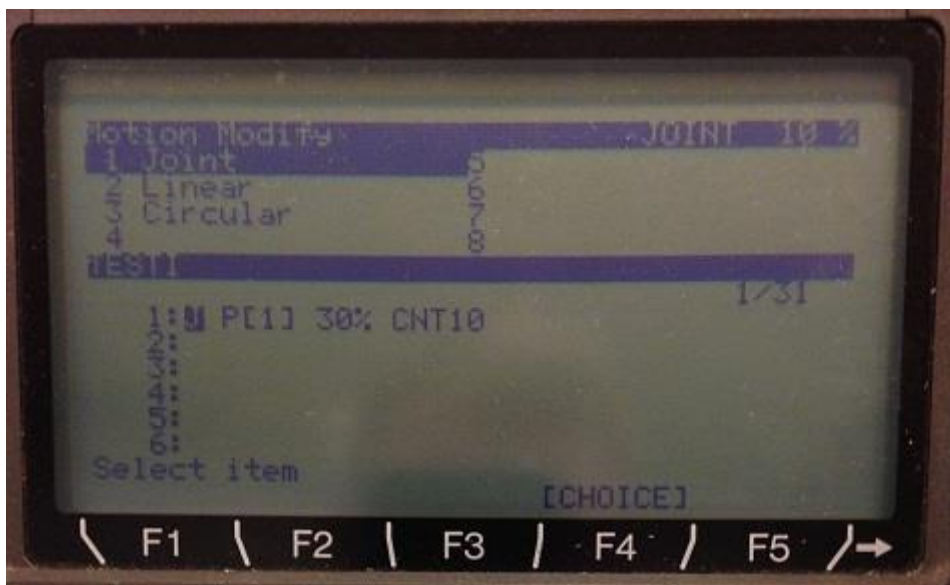


Ensimmäinen numero on rivinumero, J kertoo liiketyypin, P[1] tarkoittaa liikepisteen numeroa, 30 % on liikenopeus ja CNT10 kuvaa paikoitustarkkuutta.

Paikoitustarkkuus voi olla joko FINE tai CNT. FINE kuvastaa tarkkaa pistettä, jossa robotti pysähtyy hetkeksi. CNT tulee sanasta *continuous*, jatkuva liike. Luku CNT:n perässä on paikoitustarkkuus (0...100). CNT100 oikaisee eniten, CNT0 ei oikaise ollenkaan, mutta ei myöskään pysähdy pisteessä. Rata siis on sama kuin FINEn kanssa, mutta robotin liike on jatkuvaa.

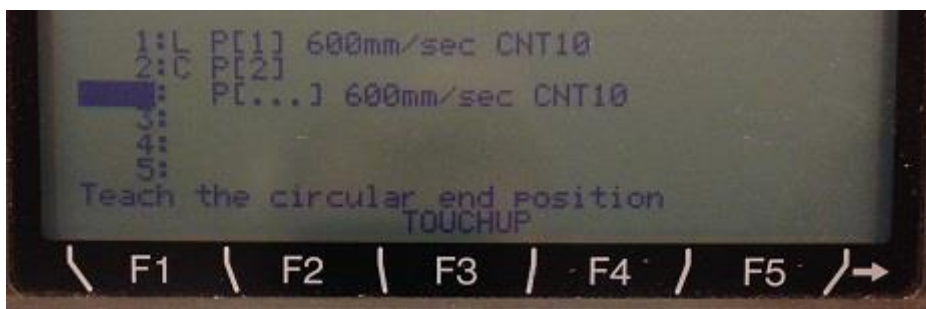
Liikepisteitä on kolme perustyyppiä:

1. **Nivelliike J** liikuttaa robotin akselit samanaikaisesti päätepisteeseen. Liikenopeus määritellään prosentteina maksiminopeudesta. Nivelliike on nopein tapa liikkua pisteiden välillä.
2. **Lineaarinen liike L** liikuttaa robottia suoraa rataa pitkin. Liikenopeus voidaan määritellä useassa eri yksikössä, tyypillisesti mm/s.
3. **Ympyränkaari C** liikuttaa robotin kolmen pisteen kautta kulkevaa kaarta pitkin. Kaaren ensimmäinen piste voi olla tyypiltään nivel- tai lineaariliike. Ympyränkaari tekee automaattisesti kaksi pistettä, joista jälkimmäisen paikka opetetaan touchup-funktionäppäimellä. Liiketyyppejä voidaan vaihtaa valitsemalla liiketyyppeä vastaava kirjain liikepisteessä ja painamalla F4 (CHOICE), jolloin näytössä tulisi näyttää seuraavanlaiselta:

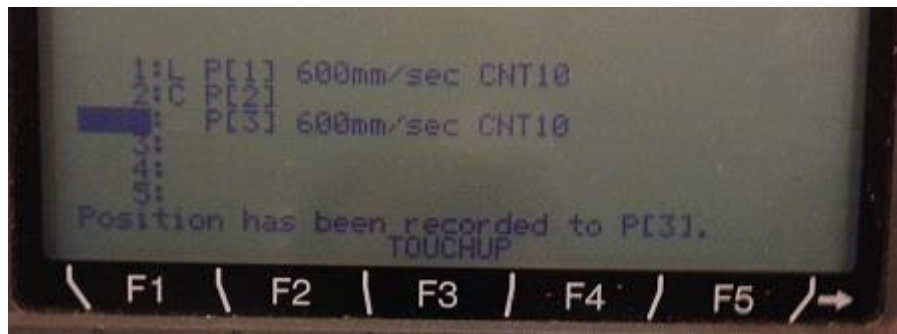


Liiketyyppi kertoo, miten tähän pisteeseen liikutaan. Jos pisteeseen tullaan suoraa viivaa pitkin, pitää liiketyypiksi valita lineaariliike. Jos pisteeseen tullaan kaarta pitkin, on pisteen liiketyypinä ympyränkaariliike. **Alla esimerkki ympyränkaariliikkeen tekemisestä.**

Jotta kaaren tekeminen olisi mahdollista, täytyy robotille opettaa ympyränkaaren kolmepistettä, alkupiste, keskipiste sekä loppupiste. Ensimmäisen pisteen liiketyypillä ei ole väliä. Tässä tapauksessa ensimmäinen liikepiste on lineaarinen. Kaaren toinen piste opetetaan normaalilla tavalla ja tallennetaan lineaariksi nivelliikkeenä. Tässä vaiheessa liiketyyppi vaihdetaan ympyränkaariliikkeeksi C.



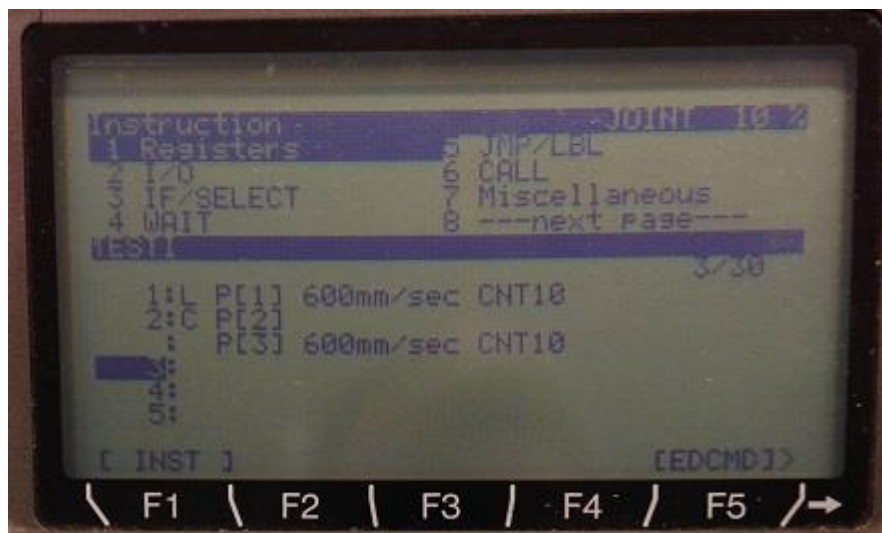
Kaaren viimeinen piste tallennetaan TOUCHUP toiminnolla ajamalla robotti haluttuun paikkaan ja painamalla SHIFT+F3 (TOUCHUP). Ohjelman tulisi nyt näyttää kutakuinkin tältä:



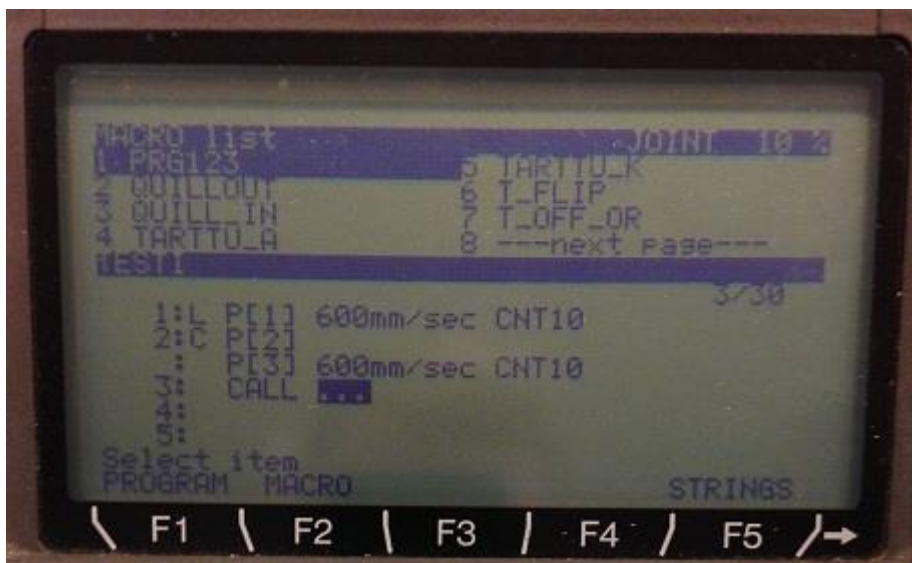
Näytön alareunassa tulisi lukea, että paikka on tallennettu pisteeseen x, P[x]. Voit halutessasi käsiajolla kokeilla, kulkeeko robotti kaariliikkeenä opetetut pisteet.

5. Peruskäskyt ja -toiminnot

Jotta harjoituksen hyväksytysti suorittaminen olisi mahdollista, tarvitaan tarttujatyökalua lieriön muotoisten palikoiden siirtelyyn. Tarttujatyökalua ohjataan antamalla sille joko käsky aueta (TARTTU_A), tai sulkeutua (TARTTU_K). Näiden käskyjen toteuttamiseksi tarvitaan CALL -käskyä, jonka avulla on mahdollista toteuttaa robotin aliohjelmia. CALL-käskyn saa valittua painamalla F1 [INST], jolloin seuraavanlainen valikko aukeaa:



Täältä valitse 6 CALL, jolloin aukeaa seuraava valikko:



Paina F2 MACRO, jolloin käyttäjän ei tarvitse etsiä oikeaa käskyä kaikkien käsiohjaimessa olevien pää- ja aliohjelmien joukosta, vaan pelkät aliohjelmat jäävät näkyviin. Etsi oikea sivu painamalla kahdesti seuraava sivu (8—next page—) -painiketta. Tästä valikosta voit nyt valita joko tarttuja_auki tai tarttuja_kiinni -vaihtoehdon. Ohjelmassasi tulisi näkyä nyt lisätty CALL-käskyrivi.

6. Virheilmoitukset ja ongelmat

Robottisolun ikääntymisen ja ahkeran käytön myötä robotti saattaa aika-ajoin testata käyttäjänsä hermoja. Jotta uusien käyttäjien käyttökokemus olisi mahdollisimman antoisa, yleisimmät virheilmoitukset, ongelmatilanteet ja ratkaisut on listattu tänne helpottamaan työskentelyä.

SRV-012 Power failure recovery

Ilmoitus siitä, että robotti on käynnistetty. Kuittaa painamalla RESET-nappia

SRVO-003 Deadman switch released

Kuolleenmiehenkytkin on vapautettu. Paina pitokytkintä puoleenväliin ja paina RESET-nappia. Jos virheilmoitus ei edelleenkään poistu, kytkin luultavasti jummutunut. Yritä painella kytkintä edestakaisin ja ”vapauttaa” se niin, että kytkimen normaali naksunta kytkintä puoleenväliin painettaessa palautuu. Yritä kuittaa RESET-napista niin kauan, että ongelma poistuu.

SYST-043 TP disabled in T1/T2 mode

Robotin käsiohjaimen ON/OFF-kytkin on OFF-asennossa. Käännä kytkin ON-asentoon ja paina RESET-nappia.

JOG-013 Stroke limit

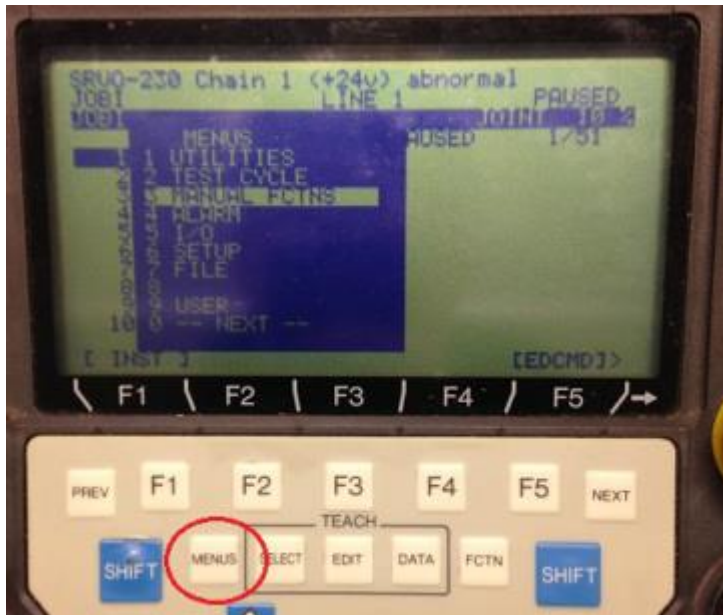
Robotin nivel ajettu ääriasentoon. Aja niveltä pois päin ääriasennosta (hitaalla nopeudella) ja paina RESET-nappia

SRVO-230 Chain 1 (+24v) abnormal tai SRVO-231 (0v) abnormal

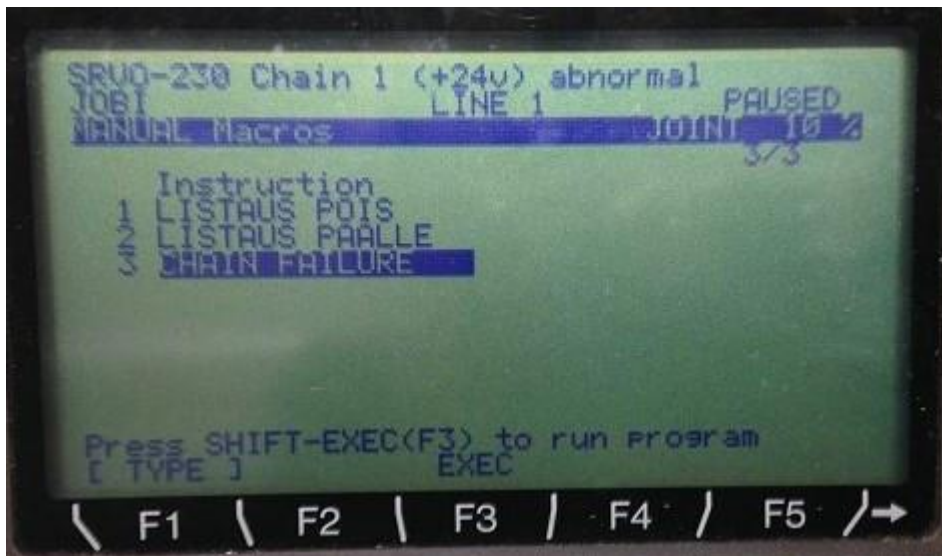
Turvapiirihäiriö. Paina pikanäppäintä SHIFT + väliviivamerkin vieressä olevaa mustaa näppäintä. Näppäinyhdistelmä saa solun käymään turvapiirin läpi, jolloin ongelma yleensä poistuu. Kuittaa painamalla RESET-nappia.



Jos virhe pysyy edelleen näytössä tee seuraavanlainen toimenpide. Paina MENUS näppäintä ja valitse MANUAL FCTNS.



Valitse 3 CHAIN FAILURE, jonka jälkeen näppäinyhdistelmällä SHIFT + F3 EXEC solu käy läpi turvapiirin. Ohjelmaan pääset takaisin EDIT näppäimestä. Kuittaa vikailmoitus vielä RESET-napilla.



7. Tarttujatyökalu ja turvallisuus

Tarttujatyökalu on vaihdettavissa oleva, sähköisesti ohjattu, paineilmatoiminen tarttujakoura, jolla voidaan tarrata ja siirtää melko pieniä kappaleita.