

# **Puukuitulusikan käsittely ja pakkaus ruiskuvalusolussa**

Tuomo Marjola

Opinnäytetyö  
Helmikuu 2020  
Tekniikan ala  
Insinööri (AMK), sähkö- ja automaatiotekniikka

Tekijä(t) Marjola, Tuomo	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä Helmikuu 2020
	Sivumäärä 109 sivua	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x
Työn nimi <b>Puukuitulusikan käsittely ja pakkaus ruiskuvalusolussa</b>		
Tutkinto-ohjelma Insinööri (AMK), sähkö- ja automaatiotekniikan tutkinto-ohjelma		
Työn ohjaaja(t) Markku Ström, Vesa Hytönen		
Toimeksiantaja(t) Kaptas Oy		
<p>Tiivistelmä</p> <p>Euroopan Unioni kieltää kertakäyttöisten muovisten ruokailuvälineiden käytön vuoteen 2021 mennessä. Toimeksiantajan asiakkaana toimiva yritys toi markkinoille puukuidusta valmistetut ruokailuvälineet, joiden on tarkoitus korvata perinteiset muoviset kertakäyttö-aterimet. Asiakasyritys halusi vastata aterimien kasvaneeseen kysyntään investoimalla lusikka-, haarukka- ja veitsiautomaatteihin, joilla saavutettiin korkeampi tuotantotehokkuus.</p> <p>Opinnäytetyön tavoitteena oli automatisoida puukuitulusikan käsittely- ja pakkaustoimenpiteet. Tavoitteena oli kaksinkertaistaa nykyinen tuotantomäärä ja poistaa operaattoreiden käsikokoonpanon työvaiheet. Työn toteuttamiseksi oli valittava toimilaitteiden automaattisen ohjauksen, kappaleenkäsittelyyn ja turvallisuustoteutukseen vaaditut ohjausjärjestelmä, teollisuusrobotti ja turvaohjain.</p> <p>Opinnäytetyössä otettiin käyttöön automaatiojärjestelmä, johon kenttäväyläyhteyksiä hyödyntäen liitettiin turvaohjain ja teollisuusrobotti. Automaatiojärjestelmä, turvaohjain ja teollisuusrobotti ohjelmoitiin ja toiminta testattiin. Automaatti otettiin käyttöön asiakkaan toimitiloissa ja automaatille suoritettiin toimintajärjestelmän mukaiset hyväksyntäajot. Tietoperustaa kerättiin kirjallisuudesta, verkkojulkaisuista ja laiteoppaista. Tutkimusmenetelmänä hyödynnettiin benchmarking- ja teemahaastatteluja, joiden avulla kerättiin tutkimusaineistoa.</p> <p>Opinnäytetyön tuloksena ruiskuvalukoneen ruiskuvalamat puukuitulusikat saatiin käsiteltyä ja pakattua siten, että automaatin lopputuotteena saatiin pakattuja aterinpusseja. Toimeksiantajan asiakkaan tuotannollisiin tavoitteisiin pystyttiin vastaamaan kaksinkertaistamalla tuotanto ja poistamalla operaattoreiden työvaiheet.</p>		
Avainsanat (asiasanat)		
Teollisuusautomaatio, ohjausjärjestelmä, turvaohjain, robotiikka, Ethernet/IP, EtherCAT		
Muut tiedot (Salassa pidettävät liitteet)		

Author(s) Marjola, Tuomo	Type of publication Bachelor's thesis	Date February 2020
		Language of publication: Finnish
	Number of pages 109 pages	Permission for web publication: x
Title of publication <b>Handling and packing of wood fiber spoons in an injection molding cell</b>		
Degree programme Degree program in Electrical and Automation Engineering		
Supervisor(s) Ström, Markku and Hytönen, Vesa		
Assigned by Kaptas Oy		
<p>Abstract</p> <p>The European Union will ban the use of disposable plastic cutlery by 2021. The customer of the principal launched cutlery made of wood fiber which will replace the traditional plastic cutlery. Environmentally friendly materials increased the demand for wood fiber cutlery. The customer company wanted to respond to the increased demand for cutlery by investing in spoon, fork and knife machines, which achieved higher production efficiency.</p> <p>The aim of this thesis was to automate the handling and packing of wood fiber spoons. The aim was to double the current production volume and to eliminate the operational steps of the manual assembly of operators. The work was carried out by selecting a control steering system, an industrial robot and a safety controller. These devices were used to control the automaton, to process the parts and to implement safety.</p> <p>The work began with the introduction of an automation system. A safety controller and an industrial robot were connected to the automation system via field buses. The control system, the safety controller and the industrial robot were programmed and function tested. The machine was taken into use at the customer's premises and the machine was approved according to the operating system. The knowledge based was collected from literature, online publications and device manuals. Benchmarking and thematic interviews were used as a research method to collect research material.</p> <p>As a result of this thesis, wood fiber spoons made by an injection molding machine were handled and packed automatically. The customer's wishes were fulfilled with a solution that doubled production and eliminated operator workflows.</p>		
Keywords/tags (subjects) Industrial automation, control system, safety controller, robotics, Ethernet/IP, EtherCAT		
Miscellaneous (Confidential information)		

## Sisältö

<b>1</b>	<b>Johdanto.....</b>	<b>7</b>
1.1	Opinnäytetyön lähtökohta .....	7
1.2	Opinnäytetyön tavoite.....	7
1.3	Kaptas Oy.....	9
<b>2</b>	<b>Käytetyt tutkimusmenetelmät .....</b>	<b>9</b>
<b>3</b>	<b>Ohjelmitava logiikka .....</b>	<b>10</b>
3.1	Rakenne .....	11
3.2	Logiikan valinta .....	13
3.3	Logiikan ohjelmointi .....	13
3.4	Turvalogiikka.....	17
3.5	Rele ja turvarele .....	17
<b>4</b>	<b>Kenttäväylät .....</b>	<b>18</b>
4.1	OSI-malli .....	18
4.2	Ethernet/IP-kenttäväylä .....	21
4.2.1	Toimintaperiaate .....	21
4.2.2	Osoitteet .....	23
4.2.3	EDS-tiedosto .....	24
4.3	EtherCAT-kenttäväylä.....	24
4.3.1	Toimintaperiaate .....	25
4.3.2	Osoitteet .....	25
4.3.3	Turvallisuusprotokolla .....	26
<b>5</b>	<b>Teollisuusrobotit.....</b>	<b>26</b>
5.1	Teollisuusrobotin tyypit.....	27
5.2	Teollisuusrobotin valinta .....	28
5.3	Teollisuusrobotin ohjainyksikkö .....	28
5.4	Teollisuusrobotin koordinaatistot .....	29
5.5	Teollisuusrobotin ohjelmointi .....	29
5.6	Teollisuusrobottitarrain ja työkalut.....	30



5.7	Digitaaliset tulo- ja lähtösignaalit .....	31
<b>6</b>	<b>Koneturvallisuus .....</b>	<b>32</b>
6.1	Standardit .....	33
6.2	Riskien arviointi .....	33
6.3	Luokat, suoritustasot ja turvallisuuden eheytystasot.....	36
6.4	Turvatoiminnot ja komponentit .....	41
6.4.1	Turvasuojat .....	41
6.4.2	Hätäpysäytys.....	43
<b>7</b>	<b>Paineilmatoimilaitteet ja lähestymiskytkimet .....</b>	<b>43</b>
7.1	Paineilma .....	44
7.2	Sylinterit .....	44
7.3	Anturit .....	45
7.4	Venttiilit.....	47
<b>8</b>	<b>Työn toteutus .....</b>	<b>50</b>
8.1	Lähtöasetelma .....	50
8.2	Omron NX1P2- ohjelmoitava logiikka.....	51
8.2.1	Ohjelmoitavan logiikan valinta .....	51
8.2.2	NX1P2- ohjelmoitavan logiikan rakenne.....	52
8.2.3	Sysmac Studio-ohjelmisto.....	55
8.3	Omron NX- turvaohjain .....	55
8.3.1	Turvaohjaimen valinta .....	55
8.3.2	Turvayksiköt.....	56
8.4	Yaskawa GP12 -teollisuusrobotti.....	58
8.4.1	Robotin valinta.....	58
8.4.2	Yaskawa GP-12 -teollisuusrobotti .....	59
8.4.3	YRC1000- robottiohjain.....	60
8.4.4	Robotin työkalu.....	61
<b>9</b>	<b>Laitekonfiguraatiot .....</b>	<b>62</b>
9.1	Projektin luominen ja yhteysasetusten määrittäminen .....	62
9.2	EtherCAT-kenttäväylän konfiguraatio .....	63

9.3	Ethernet/IP-kenttäväylän konfiguraatio .....	65
<b>10</b>	<b>Ohjelmointi.....</b>	<b>71</b>
10.1	Logiikkaohjelmointi .....	71
10.1.1	Fyysisten tulojen ja lähtöjen yhdistäminen muuttujiin .....	72
10.1.2	Ohjelmarakenne .....	73
10.1.3	Pakkaukoneen ohjaus .....	74
10.1.4	Leikkaimen ohjaus .....	75
10.1.5	Kääntöpöydän ohjaus .....	77
10.2	Turvaohjaimen ohjelmointi .....	81
10.2.1	Riskianalyysi .....	81
10.2.2	Turvapiirin rakenne.....	82
10.2.3	Turvakomponenttien yhdistäminen turva I/O- yksiköihin.....	83
10.2.4	Turvaohjaimen ohjelmointi .....	85
10.3	Teollisuusrobotiohjelmointi .....	87
10.3.1	Työkalu-, I/O- ja muuttujamäärittelyt.....	88
10.3.2	Ohjelmarakenne .....	90
10.3.3	Alkuehdot ja laatikointi.....	91
10.3.4	Kääntöpöydältä haku .....	92
10.3.5	Pakkauskoneelle vienti .....	94
10.3.6	Tarttujan ohjaus ja häiriögenerointi .....	94
10.3.7	Ohjelmasuorituksen aloittaminen .....	95
<b>11</b>	<b>Käyttöönotto .....</b>	<b>96</b>
11.1	Toiminnan testaus .....	96
11.2	Automaatin testaukset ja hyväksyntäaio .....	97
<b>12</b>	<b>Tutkimustulokset ja päätelmät .....</b>	<b>98</b>
12.1	Tuotannon automatisointi.....	98
12.2	Tuotannolliset hyödyt.....	99
12.3	Päätelmät .....	100

<b>13 Pohdinta .....</b>	<b>101</b>
<b>Lähteet.....</b>	<b>103</b>
<b>Liitteet .....</b>	<b>106</b>
Liite 1. Puukuitulusikka-automaatin layout-kuva .....	106

## **Kuviot**

Kuvio 1. Ohjelmoitavan logiikan rakenne.....	11
Kuvio 2. Ladder Diagram- esitystapa .....	15
Kuvio 3. Function Block Diagram- esitystapa .....	16
Kuvio 4. Ethernet/IP-verkon CIP:n yhteys OSI-malliin .....	22
Kuvio 5. Ethernet/IP-verkon yhteys OSI-malliin .....	23
Kuvio 6. Matriisitaulukko riskien arviointiin .....	35
Kuvio 7. Suoritustaso, luokan vastaavuus ja turvallisuuden eheytytaso SIL.....	37
Kuvio 8. PL-tason arviointi.....	38
Kuvio 9. Matriisimenetelmä SIL-tason määrittämiseksi .....	39
Kuvio 10. Ohjausjärjestelmän merkitys riskien hallinnassa.....	40
Kuvio 11. Suojauksen kytkennän liittäminen koneen toimintaan .....	42
Kuvio 12. Johdesylinteri. ....	45
Kuvio 13. 3/2-suuntaventtiilin rakenne.....	48
Kuvio 14. 5/2-suuntaventtiilin rakenne.....	49
Kuvio 15. 5/3-suuntaventtiilin rakenne.....	49
Kuvio 16. Omron NX1P2-9024DT1-ohjelmoitava logiikka .....	53
Kuvio 17. Omron NX1P2-9024DT1-tulotermiinaali .....	54
Kuvio 18. Omron NX1P2-9024DT1-lähtötermiinaali .....	54
Kuvio 19. Turvaohjaimen master ja slave järjestelmäkokoontaminen.....	57
Kuvio 20. Yaskawa GP12 -teollisuusrobotti.....	59
Kuvio 21. YRC1000-robottiohjain .....	60
Kuvio 22. YRC1000-robottiohjaimen IN 1-8 tuloryhmä.....	61
Kuvio 23. YRC1000-robottiohjaimen OUT 1-8 lähtöryhmä .....	61
Kuvio 24. Yhteysasetusten määrittäminen. ....	63

Kuvio 25. EtherCAT-slave-laitteen lisäys. ....	64
Kuvio 26. Turva I/O- yksiköiden lisäys. ....	65
Kuvio 27. Ethernet/IP- adapterin asetukset. ....	66
Kuvio 28. I/O:n osoitealueet. ....	67
Kuvio 29. Ethernet/IP I/O- datatyypin määritys. ....	68
Kuvio 30. Ethernet/IP I/O- muuttujat. ....	68
Kuvio 31. Sysmac Studio Ethernet/IP yhteysasetukset. ....	69
Kuvio 32. Sysmac Studio Ethernet/IP- yhteys robotille ....	70
Kuvio 33. Ethernet/IP- yhteyden testaus. ....	70
Kuvio 34. Ohjelmoitavan logiikan ja turvaohjaimen I/O-kaavio. ....	72
Kuvio 35. Ohjelmoitavan logiikan I/O:n yhdistäminen muuttujiin. ....	73
Kuvio 36. Logiikkaohjelmamoduuli pakkaus. ....	74
Kuvio 37. Logiikkaohjelmamoduuli leikkaimen sekvenssiohjaus. ....	76
Kuvio 38. Logiikkaohjelmamoduuli leikkaimen lähdöt. ....	77
Kuvio 39. Logiikkaohjelmamoduuli kääntöpöydän ohjaus. ....	79
Kuvio 40. Logiikkaohjelmamoduuli sekvenssiohjaus. ....	80
Kuvio 41. Ohjelmamoduuli kääntöpöydän ulkoiset kättelyt. ....	81
Kuvio 42. Turvatulojen yhdistäminen turvatuloyksikköön. ....	84
Kuvio 43. Turvalähtöjen yhdistäminen turvalähtöyksikköön. ....	84
Kuvio 44. Turva I/O:n muuttujamääritys. ....	85
Kuvio 45. Hätä-seis- piirin ohjaus. ....	86
Kuvio 46. Turvapiirin ohjaus. ....	87
Kuvio 47. Työkalupisteen määrittely. ....	88
Kuvio 48. Teollisuusrobotin I/O-määrittelyt. ....	89
Kuvio 49. Teollisuusrobotin pääohjelmarakenne. ....	91
Kuvio 50. Teollisuusrobotin LAATIKKO- aliohjelma. ....	92
Kuvio 51. Teollisuusrobotin KÄÄNTÖPÖYTÄ- aliohjelma ....	93
Kuvio 52. Teollisuusrobotin TARTTUJA- aliohjelma. ....	95

**Taulukot**

Taulukko 1. OSI-viitemallin kerrokset.....	19
Taulukko 2. Suoritettavat toimenpiteet riskien arviointiluokan mukaisesti. ....	35

# 1 Johdanto

## 1.1 Opinnäytetyön lähtökohta

Automaatiolaitteiden laitevalmistajana toimiva Kaptas Oy on saanut asiakkaaltaan toimeksiannon toteuttaa puukuituaterimien käsittely- ja pakkaustoimenpiteiden automatisointi. Automaateilla pakattavien puukuituaterimien on tarkoitus korvata perinteiset muovista valmistetut kertakäyttöaterimet. Ympäristökuorman pienentämiseksi puukuituaterimien raaka-aineena käytetään puukuitumassaa, joka on sekoitus puukuitupellettiä ja sokeriruo'osta valmistettua polyeteeniä.

Euroopan Unioni kieltää kertakäyttöiset muoviset ruokailuvälineet vuoteen 2021 mennessä. Euroopan komission mukaan yli 80 % meressä olevasta roskasta on muoviva. Vuonna 2021 kiellettävät muovituotteet koskevat kertakäyttöisiä muovista valmistettuja pumpulipuikkoja, pillejä, juomasekoittimia ja ruokailuvälineitä. (Parlamentti sinetöi kertakäyttömuovin... 2019.)

Muovista valmistettujen ruokailuaterimien kieltö vuoteen 2021 on jo nyt hidastanut kertakäyttöisten muoviaterimien kysyntää Euroopassa. Markkinoille tarvitaan uusia, ympäristöystävällisistä materiaaleista valmistettuja aterimia, joilla korvataan kertakäyttöiset muoviaterimet. Tähän kysyntään vastataan tulevaisuudessa puukuidusta valmistetuilla ruokailuaterimilla. Toimeksiantajan asiakasyrityksen tavoitteena on vastata kasvaneeseen kysyntään investoimalla uusiin lusikka-, haarukka- ja veitsiautomaatteihin, joilla saavutetaan korkea tuotantotehokkuus.

## 1.2 Opinnäytetyön tavoite

Tämän opinnäytetyön tavoitteena oli automatisoida puukuitulusikan käsittelyn pakkaustoimenpiteet. Tavoitteena oli luoda ratkaisu, jossa ruiskuvalukoneen valamat puukuituaterimet käsitellään ja pakataan automaattisesti niin, että lopputuotteena saadaan pakattuja aterinpusseja. Automatisoinnin tavoitteena on kaksinkertaistaa nykyinen tuotantomäärä. Lisäksi opinnäytetyön tavoitteena pyrittiin vähentämään

operaattoreille aterimien käsikokoonpanosta aiheutuvia terveysriskejä. Operaattorin tehtävänä käsikokoonpanossa oli aterimien irrotus ruiskuvalukoneen valamasta aterinviuhkasta, josta operaattori joutui irrottamaan lusikat yksitellen. Aterimet järjesteltiin ja pinottiin käsin, minkä jälkeen ne asennettiin pussiin. Operaattori lämpösau-masi ja asetti tarran pussiin, jolloin tuotepussi oli valmiina jatkokäsittelyyn. Automa-tisoinnin tavoitteena oli saada poistettua operaattorin käsikokoonpanon työvaiheet.

Puukuituaterimien käsittely ja pakkaustoimenpiteiden automatisointi projektina oli laaja kokonaisuus. Projektin läpivieminen vaati yhteistyötä yrityksen johdon, meka-niikkasuunnittelun, sähkösuunnittelun, laatutiimin sekä sähkö- ja mekaniikka-asen-nuksen kesken. Tämä opinnäytetyö rajattiin sähkösuunnittelun osuuteen, joka pai-nottui ohjausjärjestelmä- ja toimilaittevalintoihin, toimilaitteiden liityntärajapintoihin, ohjelmointiin ja käyttöönottoon.

Ohjausjärjestelmänä toimivat ohjelmoitava logiikka, turvaohjain ja teollisuusrobotti oli valittava projektivaatimusten mukaisesti. Automaatin sisältämien toimilaitteiden liitántärajapinnat oli suunniteltava siten, että ne ovat ohjattavissa ohjelmoitavalla lo-giikalla. Logiikka- ja robottiohjelmointi oli toteutettava siten, että toimilaitteiden liik-keet ja kappaleenkäsittely voidaan suorittaa automaattisesti. Automaatin käyttäjä-turvallisuuden toteuttamiseen käytettiin turvaohjainta. Turvaohjaimen ohjelmointi ja siihen liitettävät turvakomponentit oli suunniteltava siten, että käyttäjäturvallisuus toteutui riskianalyysin mukaisesti. Käyttöönottovaiheessa automaatin toiminta oli to-dennettava tekemällä automaatille tilaajayrityksen toimintajärjestelmän mukaiset testaukset. Operaattoreille pidettiin käyttäjäkoulutus automaatin turvalliseen käyt-töön liittyen ja toimitettiin automaatin käyttöohje. Viralliset hyväksyntäajot suoritet-tiin asiakkaan ja toimeksiantajan sopimuksen mukaisesti.

Laajuuden takia aiheesta rajattiin pois sähkösuunnittelu-osio, joka sisälsi komponent-tiluettelon ja sähkökuvatoteutuksen. Lisäksi aihe rajattiin yhteen automaattiin, eli puukuitulusikanpakkaamiseen. Lusikka valittiin kohteeksi, koska sen automaatin val-mistusajankohta oli aikataulutettu ensimmäiseksi. Haarukka- ja veitsiautomaatit pys-tytään toteuttamaan tämän opinnäytetyön tutkimustulosten perusteella.

### 1.3 Kaptas Oy

Opinnäytetyön tilaaja on automaatioteknologian asiantuntijayritys Kaptas Oy. Yritys sijaitsee Liperissä Itä-Suomessa. Kaptas tarjoaa asiakkailleen asiantuntijapalveluita, kappaleen käsittelyä, tuotannon automatisointiratkaisuita sekä automaatiolaitteiden huolto- ja ylläpitopalveluita. Asiakkaina on lääketeollisuuden, metallin, muovin, elektroniikan ja elintarvikealan yrityksiä. Tuotantomenetelmien ja prosessien kehityksessä hyödynnetään kattavasti robotiikkaa, simulointia, konenäköä, mittaustekniikkaa, kappaleensyöttöä, keinoälyä ja big dataa.

Automaatio käsitteenä tarkoittaa itsestään toimivaa. Automaatiossa toiminta tapahtuu ilman ihmisen ohjaavaa tai suorittavaa osuutta. Automaatti on automaattisesti eli itsestään toimiva kone tai laite. Tuotantoautomaatio voidaan jakaa prosessi- ja koneautomaatioon. Prosessiautomaatio käsittelee virtaavia nesteitä ja niiden ohjaustekniikkaa. Koneautomaatio ymmärretään yleensä kappaletavara-automaatioksi. Kappaletavara-automaatiossa tuotteet kulkevat kuljettimia pitkin ja niitä kokoonpannaan, pakataan, varastoidaan ja lajitellaan automaattisesti. (Keinänen, Kärkkäinen, Lähetkangas & Sumujärvi 2007.)

## 2 Käytetyt tutkimusmenetelmät

Tämä opinnäytetyö toteutettiin kehittämistutkimuksena. Opinnäytetyötä voi pitää kehittämistutkimuksena, koska sillä kehitettiin olemassa olevaa tuotantomenetelmää. Kehittämistutkimuksella tähdätään muutokseen. Muutoksen aikaansaamiseksi kehittämistutkimuksessa kehitetään menetelmää, tuotetta tai organisaatiota yms. Kehittämistutkimuksen taustalla on aina teoria tai teorian, joihin kehittäminen perustuu. Kehittäminen vaatii myös tutkimuksellista otetta, jolloin voidaan puhua kehittämistutkimuksesta. (Kananen 2012, 76.)



Kehittämistutkimuksessa hyödynnetään erilaisia tutkimusmenetelmiä. Tässä opin-  
näytetyössä käytettiin laadullisen eli kvalitatiivisen tutkimuksen menetelmiä. Kvalita-  
tiivisessa tutkimuksessa ensin perehdytään tutkimusongelmaan ja sen määrittämi-  
seen, jota seuraa tavoitteet, joihin pyritään hankkimalla tarvittava aineisto. Tutkimus  
prosessi voidaan myös vaiheistaa tutkimuksen suunnittelu-, tiedonkeruu-, analyysi- ja  
tulkintavaiheisiin. (Kananen 2010, 36.) Tämän opinnäytetyön tärkeimpinä tutkimus-  
kysymyksinä, joihin haetaan vastaus kehittämistutkimusmenetelmien avulla ovat:

- Kuinka tuotanto automatisoidaan?
- Kuinka tuotannolliset hyödyt saavutetaan?

Tässä opinnäytetyössä tutustuttiin ensin projektin lähtötietoihin, jotka oli luotu nyky-  
tilan kartoittamisen tuloksena. Lähtötiedot sisälsivät asiakkaan hyväksymät mekaani-  
set ratkaisut ja toimintaperiaatteen automaatioimintaan liittyen. Lisäksi tietoläh-  
teitä kerättiin sähkö- ja mekaniikkasuunnittelijoiden haastatteluilla. Mekaniikkasuun-  
nittelijan haastattelun tavoitteena oli kerätä tutkimusaineistoa automaatin mekaa-  
nisten toimilaitteiden määrästä ja toimintarutiineista, joiden tietämys oli oleellisen  
osana automaatin ohjausjärjestelmävalintaa ja ohjelmointia. Ohjausjärjestelmä valin-  
noissa sekä niiden ohjelmoinnissa hyödynnettiin benchmarking- menetelmää. Säh-  
kösuunnittelijoiden benchmarking-haastatteluilla kerätyllä tutkimusaineistolla saatiin  
tietoa aiemmin hyväksi havaituista logiikkamalleista ja ohjelmarutiineista.

Kerättyjen tietolähteiden ja projektin lähtötietojen perusteella automaatin ohjausjär-  
jestelmä ja toimilaittevalinnat saatiin valittua. Logiikan-, turvaohjaimen- ja robotin oh-  
jelmointi ja kenttäväyläkonfiguraatiot toteutettiin benchmarking tutkimustuloksia ja  
laitevalmistajan laiteoppaita hyödyntämällä.

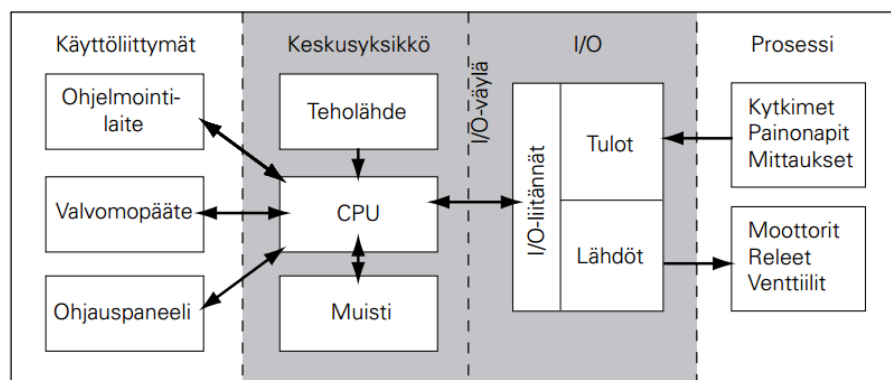
### **3 Ohjelmoitava logiikka**

Ohjelmoitava logiikka (PLC = Programmable Logic Controller) on yksittäinen, itsenäi-  
nen (Stand Alone) automaatiojärjestelmä, jota käytetään reaaliaikaisten automaatio-  
prosessien, esimerkiksi koneiden tai tuotantolinjojen ohjaukseen. Ohjelmoitavassa

logiikassa on tulo- ja lähtöportteja, joihin kytketään kaikki kenttälaitteet. Ohjelmoitava logiikka ohjaa kenttälaitteita ladatun sovellusohjelman mukaisesti. Logiikka kehitettiin alun perin korvaamaan monimutkaisia relekytkentöjä. Logiikalla voi korvata satoja tai tuhansia releitä ja ajastimia. (Keinänen ym. 2007, 212.)

### 3.1 Rakenne

Kuviossa 1 on esitetty ohjelmoitavan logiikan rakenne. Teholähteen (Power) tehtävänä on tuottaa logiikan keskusyksikön ja I/O-yksiköiden vaatima teho. Kenttälaitteiden vaatima teho otetaan yleensä erillisestä teholahteesta, jonka käyttöjännite on yleensä 24VDC. Teholähteen käyttöjännitteenä käytetään 230VAC tai 24VDC. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 107.)



Kuvio 1. Ohjelmoitavan logiikan rakenne. (Kippo & Tikka 2008, 56)

Keskusyksikkö eli CPU (Central Processing Unit) lukee käsittelyohjeet ohjelmamuistista ja keskusyksikkö ohjaa toimilaitteita ohjelman mukaisesti. Keskusyksikkö on toteutettu poikkeuksetta mikroprosessorilla. Logiikassa voi olla useampia prosessoreita ja jokaisella prosessorilla on oma erikoistehtävänsä. (Kippo & Tikka 2008, 54,57.)

Logiikan ohjelma- ja datamuisti ovat tyypiltään RAM (Random Access Memory) – muistia. Tämän tyyppinen muisti voi olla haihtumaton tai haihtuva. Ohjelmamuisti on yhdellä RAM-alueella, jonka on säilyttävä sähkökatkon yli. Vanhemmat logiikkajärjes-

telmät tarvitsevat akun varmentamaan ohjelmamuistin säilymisen. Näissä järjestelmissä akun loppuminen aiheutti ohjelman katoamisen ohjelmamuistista. Uusissa alustoissa ohjelmamuisti tallennetaan ei haihtuvalle alueelle esimerkiksi Secude Digital (SD) RAM:iin. Logiikoiden muistikoko ilmoitetaan yleensä ohjelmarivien määränä, jonka perusyksikkö on  $1K = 1024$  käskyä. Muistikoot vaihtelevat pienlogiikan 0,25K muisteista aina isojen järjestelmien 256 kiloon. (Keinänen ym. 2007, 225; PLC Memory 2006.)

Ohjelmoitavassa logiikassa on tulo- ja lähtöportteja, joihin on kytketty kaikki prosessin kenttälaitteet. Tulopiirien avulla kentältä tulevat signaalit kytketään ohjelmoitavaan logiikkaan, esimerkiksi rajakytkimet, mittausanturit, painonapit ja lähettimet. Lähtöpiirit ohjaavat automaatin sisältämiä toimilaitteita kuten venttiileitä, taajuusmuuttajia tai servo-ohjaimia. Tulo- ja lähtöpiirien yhteisnimityksenä käytetään nimitystä I/O (input/output). (Kippo & Tikka 2008, 59.)

Tulot- ja lähdöt ovat monesti rinnakkaisia bittiryhmiä eli tavuja (Byte) ja sanoja (Word). Bitti on pienin muistiyksikkö, joka voi saada arvon 0 (epätosi) tai 1 (tosi). Kahdeksan rinnakkaista bittiä muodostaa tavun. Tavu voi saada  $2^8$  eli 256 arvoa ja sana  $2^{16}$  eli 65536 arvoa. (Keinänen ym. 2007, 227.)

Logiikat voivat olla varustettuja kiinteällä I/O-määrällä tai ne voidaan koota monesta yksittäisestä yksiköistä. Kiinteällä I/O-määrällä varustetut logiikat soveltuvat yksittäisten laitteiden ohjaukseen. Näissä laitteissa ei yleensä ole mahdollisuutta laajentaa I/O-määrää. Suuremmissa modulaarisissa logiikkajärjestelmissä tulo- ja lähtöportit voidaan hajauttaa kentälle erillisiin kenttäväyläyksiköihin, jolloin puhutaan kenttäväylästä. Kenttäväylä on kytketty logiikkaan tiedonsiirtokaapelilla käyttäen logiikan tukemaa kenttäväyläprotokollaa. Protokollalla tarkoitetaan kieltä tai kielioppia, mitä laitteet käyttävät kussakin kenttäväylässä. (Keinänen ym. 2007, 212,214.)

HMI (Human Machine Interface) tarkoittaa koneen ja operaattorin välisen vuorovaikutuksen tarvittavia laitteita ja ohjelmia, joiden avulla operaattori ohjaa konetta tai prosessia. Tätä vuorovaikutusta kutsutaan käyttöliittymäksi. (Kippo & Tikka 2008,

46.) Käyttöliittymänä voi toimia yksinkertaisimmillaan käynnistys- ja pysäytyspainikkeet. Tavallisesti käyttöliittymänä käytetään kosketusnäytöllä varustettua operointipaneelia. Automaatiolaitteen sisältämän teollisuusrobotin käsiohjainta voidaan käyttää robottitoimintojen ohjauksen lisäksi, myös käyttöliittymänä. Käyttöliittymä liitetään ohjelmoitavaan logiikkaan tiedonsiirtokaapelilla käyttäen logiikan tukemaa tiedonsiirtoprotokollaa. (Keinänen ym. 2007, 220.)

### 3.2 Logiikan valinta

Logiikan valintaan vaikuttavia tekijöitä ovat I/O-määrä, liitettävien toimilaitteiden määrä ja protokolla, toimintojen monimutkaisuus, turvalaitteiden tarve ja hinta. Ensimmäisenä on selvitettävä tarvittavien digitaalisten sekä analogisten tulojen ja lähtöjen määrä. Lisäksi on selvitettävä, onko osa I/O:sta tarve hajauttaa kentälle, jolloin on tarve kenttäväylälle. Modulaariset logiikat mahdollistava monien erikoisyksiköiden liittämisen logiikkaan, joilla voi toteuttaa esimerkiksi säätöjä, mittauksia tai paikoituksia. Modulaarisen logiikan etuna on myös laajennettavuus mahdollisten muutosten varalta. Pienemmissä logiikoissa yksiköiden lisääminen ei aina ole mahdollista. Toimilaitteet, mukaan lukien turvalaitteet liitetään logiikkaan käyttäen standardisoitua tiedonsiirtoprotokollaa, joka on huomioitava logiikan valinnassa. (Fonselius ym. 1996, 107.)

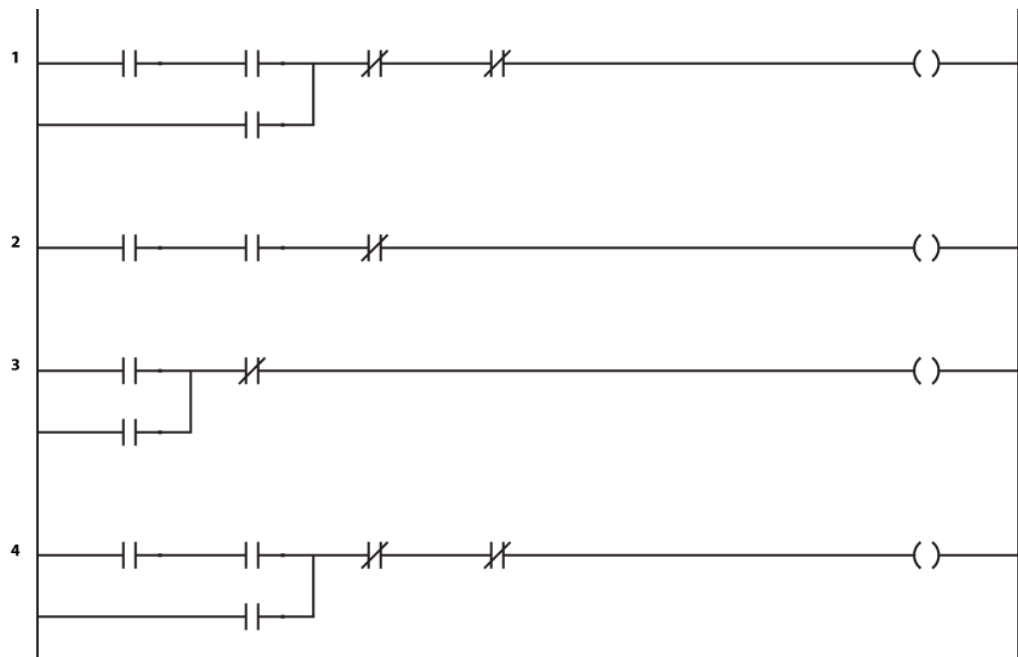
### 3.3 Logiikan ohjelmointi

Logiikkavalmistajia on monia, joten ohjelmoinnista ei ole mahdollista esittää aukotonta kokonaisuutta ohjelmasisällön toteutustavasta. Ohjelmointi on luovaa työtä ja samaan lopputulokseen voidaan päästä useilla eri ohjelmointitavoilla ja ohjelmarakenteilla. Ohjelmoinnissa on pyrittävä mahdollisimman selkeään ja luettavaan kokonaisuuteen, jota myös ulkopuolisen ohjelmoijan on helppo ymmärtää. (Keinänen ym. 2007, 227.)

Ohjelmoinnin lähtökohtana on tarkat lähtötiedot ohjelmoitavasta ohjauskohteesta. Saatujen lähtötietojen perusteella ohjelmoija tekee tarvittavat muistiinpanot, relekaaviot, logiikkakaaviot tai toimintadiagrammit, joiden perusteella itse ohjelmointi kirjoitetaan logiikalle ohjelmointilaitteen avulla. Ohjelman kirjoittamiseen käytetään yleisesti logiikkavalmistajan ohjelmistoympäristöä, jossa on käytössä monenlaisia esitystapoja. Yleisimmin käytettyjä esitystapoja ovat standardissa IEC 61131-3 määritetyt logiikkakaavio (Function Block Diagram), relekaavio- (Ladder Diagram), strukturoitu teksti (Structured Text) ja sekvenssiohjausohjelmointi (Sequential Function Chart) sekä käskylista (Instruction List). Ohjelmayksiköt koostuvat virtapiireistä, jotka on toteutettu joillakin edellä mainituista esitystavoista. (Fonselius ym. 1996, 117.)

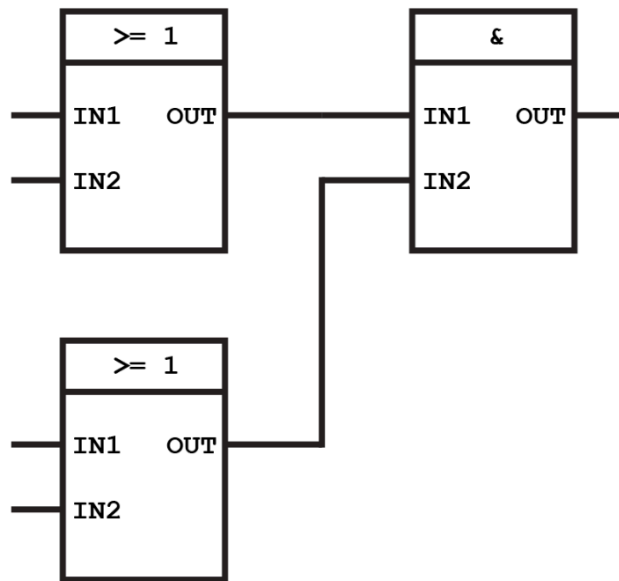
Ohjelman sisällöstä tehdään modulaarinen, joka tarkoittaa, että ohjelmatoiminnot jaetaan pienempiin kokonaisuuksiin. Osa ohjelmamoduulien toiminnoista suoritetaan jokaisella ohjelmakierrolla, osa tietyin väliajoin tai osa tietyn ehdon täytyttyä. Ohjelmamoduulin sisältöä voi jakaa osakokonaisuuksiin, joita kutsutaan nimellä askel (Rung). Ohjelma sisältää ulkoisia- (Global) ja paikallisia- (Local) muuttujia. Ulkoiset muuttujat ovat jokaisen ohjelmamoduulin käytettävissä ja paikalliset ainoastaan yhden ohjelmamoduulin sisällä, jonne ne on määritetty. (Fonselius ym. 1996, 117.)

Relekaavio-ohjelmointi (Ladder Diagram, LD) on perinteisin ohjelmoinnin esitystapa. Sen käyttö aloitettiin logiikoiden kehityksen alkuvaiheessa, jolloin logiikka kehitettiin korvaamaan perinteisiä releitä. Releohjaus on helppo muuttaa relekaavio-ohjelmaksi. Relekaaviosta käytetään nimitystä kosketinkaavio tai tikapuukaavio. Relekaavio-ohjelma sisältää virtapiirejä, jotka koostuvat JA- ja TAI-kytkentäisistä koskettimista. Lisäksi ohjelma voi koostua toimilohkoista (Function Block, FB), joita ovat ajastimet, RS-piirit, matemaattiset toimilohkot, laskurit, datamuunnokset ja trigonometriset funktiot. (Fonselius ym. 1996, 119-120.) Kuviossa 2 esitellään Ladder Diagram- ohjelmaesitystapaa.



Kuvio 2. Ladder Diagram- esitystapa. (Ladder Logic Tutorial 2019)

Logiikkakaavio-ohjelmointi (Function Block Diagram, FBD) perustuu IEC 61131-3 standardissa määritettyjen logiikkasymboleiden käyttöön. FBD-kaavion esitystavassa etuna on valmiit ohjelmatoimintosymbolit, joita voi esittää havainnollisesti ja tiivistysti kuvion 3 mukaisesti. Esimerkiksi JA- ja TAI-operaatiot ovat valmiina yksittäisinä ohjelmasympboleina. (Fonselius ym. 1996, 121.)



Kuvio 3. Function Block Diagram- esitystapa (Function Block Diagram... 2018)

Käskylista (Instruction List, IL) ohjelmointi muistuttaa Assembly-kieltä. IL-ohjelmoinnissa käskyrivit koostuvat käskystä ja siihen liitetystä operandista ja mahdollisesta kommentista. Käskyt muistuttavat logiikkakaavion ja relekaavion ohjelmarakenteita. Virtapiiri aloitetaan latauskäskyllä, jonka jälkeen seuraa joukko JA- ja TAI-lukituksia. Virtapiiri päättyy lähdön tai muistipaikan ohjaukseen. (Fonselius ym. 1996, 122.)

Strukturoitu teksti (Structured Text, ST) on korkeamman tason ohjelmointikieli, joka sisältää modernin ohjelmointikielen tärkeät elementit. ST-kieli sisältää valintausekkeitä (THEN-IF-ELSEIF-END-ENDIF ja CASE-ELDE-END\_CASE) sekä erilaisia silmukkarakenteita (FOR-END\_FOR-NEXT, WHILE-END-END\_WHILE ja UNTIL-REPEAT-END\_REPEAT, EXIT). Tekstipohjaisen ohjelmaympäristön etuna on, että ohjelmakoodi saadaan pienempään tilaan ja ST-kieleen voi yhdistää eri ohjelmointikieliä. Esimerkiksi ohjelman sisältämän toimilohkon sisällä voi olla strukturoidussa tekstissä kirjoitettuja toimintoja. (Fonselius ym. 1996, 123.)

Sekvenssiohjausohjelmointi (Sequentiel Function Chart, SFC) on tarkoitettu askelusohjauksien suorittamiseen. SFC-ohjelmointia käytetään, kun toimintoja suorite-

taan peräkkäin. Seuraavalle käskyriville siirrytään vasta, kun edelliset ehdot ovat täyttyneet. Sekvenssiohjelmointi on yleistä kappaletavara-automaatioissa, jossa toimilaitteet toistavat samoja toimintoja kerta toisensa jälkeen. (Fonselius ym. 1996, 123.)

### 3.4 Turvalogiikka

Turvalogiikka (Safety Programmable Logic Controller) toimii tavallisen ohjelmoitavan logiikan tavoin, mutta se on rakennettu, sertifioitu ja testattu vastaamaan kansainvälisiä turvallisuusvaatimuksia. Turvalogiikka tukee integroituja turvatoimintoja ja sitä käytetään hallitsemaan automaatiolaitteiden turvatoimintoja. Turvalogiikka valvoo jatkuvasti turvalaitteiden tulo- ja lähtöporttien tilatietoja. Turvalaitteiden aktivoituessa turvalogiikka on ohjelmoitu suorittamaan automaattisesti toimenpiteet prosessin tai koneen turvallisen tilan saavuttamiseksi.

Turvalogiikka mahdollistaa turvallisuuden hallinnan ja standardiohjauksen. Tavallinen ohjelmoitava logiikka tarjoaa ainoastaan standardiohjauksen. Turvalogiikan käyttö turvallisuustoimintojen toteutuksessa vähentää kenttäjohdotusten määrää, koska turvareleitä ei tarvita. Tästä johtuen turvalogiikan ympärille suunniteltu turvajärjestelmä on erittäin joustava, jota on helppo muokata. Muokkaus vaatii ainoastaan ohjelmamuutoksia, mutta ei johdotuksen muutoksia tai lisäreleitä. (Goble 2003.)

### 3.5 Rele ja turvarele

Rele on sähkömekaaninen tai elektroninen kytkentälaitte, jonka kelaan ohjatulla virralla voidaan ohjata ohjausvirtaan nähden moninkertaisia virtoja. Releitä käytetään heikon ohjaussignaalin vahvistamiseen, toimilaitteiden välisten rajapintojen potentiaalierotukseen, tasa- ja vaihtojännitteiden sovittamiseen sekä koneturvallisuuden parantamiseen (turvarele). Mekaanisen rele koostuu kelasta, palautusjousesta, ankkurista ja kosketinryhmästä. Kun kelaan tuodaan sähkövirta, kela rautasydämineen muodostaa sähkömagneetin, joka vetää rautaista ankkuria puoleensa. Ankkuri vaihtaa kosketinkärkien paikkaa, jolloin virtapiiri sulkeutuu. Jousi palauttaa ankkurin, kun kelan ohjausvirta poistetaan. (Keinänen ym. 2007, 52.)



Turvareleen tehtävänä on lisätä koneturvallisuutta. Turvareleitä käytetään hätä-seispiireihin, kaksikäsihjauksiin, turvakytkeisiin, valoverhoihin ja koneiden tulo sekä lähtötietojen valvontaan. Turvareleitä käytetään erilaisissa koneautomaatio sovelluksissa, kuten automaatiotuotanto ja robottisovelluksissa. Ne ovat yksi tai kaksikanavaisia turva- tai hätä-seispiirin kytkeviä laitteita, joiden avulla voidaan estää hallitsemattomat toiminnot ja siten vähentää inhimillisiä ja aineellisia vahinkoja. (Keinänen ym. 2007, 54.)

## 4 Kenttäväylät

Ihmisten tavoin koneiden välinen kommunikointi vaatii tietyt rajat ja säännöt, joita noudattamalla koneet pystyvät keskustelemaan keskenään. Näitä rajoja kutsutaan protokolliksi. Koneiden välinen tiedonsiirto vaatii yhteisen protokollan lisäksi yhteensopivat laitteistot, joiden on kyettävä lähettämään ja vastaanottamaan tietoa. Laitteiden on oltava yhteensopivia, jotta kommunikaatio saadaan toimimaan. Laiteverkot ja kokonaisuudet vaativat poikkeuksetta omat sääntömallinsa, jotta laitekokonaisuuksien tiedonsiirto onnistuu ongelmitta. (Pyyskänen 2007, 9.)

Kenttäväylä on kaksisuuntainen digitaalinen väyläliityntäinen tiedonsiirtoratkaisu, joka yhdistää erilaisia mittaus- ja ohjauslaitteet yhdeksi automaatiojärjestelmäkokoaisuudeksi. Kenttäväylän avulla automaatiojärjestelmä kytketään kenttälaitteisiin. Kenttälaitteina voivat olla, hajautettu I/O, käyttöliittymä, robotit, turvaohjain sekä älykkäät mittaus ja ohjauslaitteet. (Piikkilä & Salhstén 2017, 28.)

### 4.1 OSI-malli

OSI- (Open System Interconnection) malli on kansainvälisen ISO- (International Standards Organization) standardin kehittämä seitsemän kerroksen rakenne tiedonsiirtojärjestelmälle. OSI-malli itsessään ei ole tiedonsiirto protokolla, vaan se mahdollistaa standardisoidun toimintamallin tiedonsiirtoprotokollien käyttöön. OSI-mallin neljä

alinta kerrosta muodostavat tietoliikennepalvelut, johon rajoittuu tiedonsiirron tekninen toteutus. Ylempien kerrosten tarkoituksena on palvella sovellusohjelmistoja ja käyttäjätoimintoja. (Piikkilä & Salhstén 2017, 62.) Taulukossa 1 on esitetty OSI-mallin seitsemän kerroksen tiedonsiirtomalli.

Taulukko 1. OSI-viitemallin kerrokset. (Piikkilä & Salhstén 2017)

<b>7</b>	<b>Sovelluskerros</b>	<b>Application layer</b>
<b>6</b>	<b>Esiytystapakerros</b>	<b>Presentation</b>
<b>5</b>	<b>Yhteyskerros</b>	<b>Session</b>
<b>4</b>	<b>Kuljetuskerros</b>	<b>Transport</b>
<b>3</b>	<b>Verkkokerros</b>	<b>Network</b>
<b>2</b>	<b>Siirto(yhteys)kerros</b>	<b>Link</b>
<b>1</b>	<b>Fyysinen kerros</b>	<b>Physical</b>

Fyysisen (Physical) kerroksen tehtävänä on huolehtia tiedonsiirrosta fyysisellä tasolla tarjoamalla mekaaniset ja sähköiset ominaisuudet sekä menetelmät fyysisen yhteyden ylläpitämiseksi, purkamiseksi ja muodostamiseksi. Fyysinen kerros signaloi bittivirtoja sähköiseksi signaaliksi, moduloi signaalit medialle ja vastaavasti purkaa modulaatiot vastaanottavassa laitteessa ja muuntaa signaalit takaisin bittivirroiksi. Liitäntä fyysiseen siirtotiehen toteutetaan määrittämällä tiedonsiirtonopeus, signaalin jännitetasot ja ajoitukset bittitasolla. (Piikkilä & Salhstén 2017, 64.)

Siirto(yhteys)kerroksen tehtävänä on tiedonsiirto vierekkäisten laitteiden ja järjestelmien välillä. Tämä on toteutettu kerroksessa siten, että se huolehtii sanomien siirrosta solmupisteiden välillä yhteiskäytännön mukaisesti. Lisäksi siirtokerros huolehtii siirtoyhteyden purkamisesta ja rakentamisesta sekä bittitason virheentarkastuksesta. Se siirtää muodostettuja kehyksiä vieruslaitteelle ja valvoo siirtotapahtuman läpinäkyvyyttä ja eheyttä. (Piikkilä & Salhstén 2017, 64.)

Verkkokerros huolehtii verkkoon liittyvien ongelmien ratkaisemisesta. Siellä käytetään ja tulkitaan verkko-osoitteita sekä siirretään sanomia välittävien laitteiden yli. Tehtävänä on tietojen välitystoiminnot ja tiedon siirtotien toteutus. Siirtämistä välinlaitteiden yli, kutsutaan reititykseksi, koska kerros päättää jokaisesta laitteesta,

mistä portista seuraavaksi sanoma lähetetään eteenpäin. (Piikkilä & Salhstén 2017, 63.)

Kuljetuskerroksen tehtävänä varmistaa tiedonsiirto päästä päähän. Se valvoo lähetyserien sisältämien osien keskinäistä järjestystä ja hoitaa lähettäjän ja vastaanottajan välistä vuon ohjausta. Kuljetuskerros huolehtii yläpuolelleen jatkuvan tiedonsyötön, jotta ylemmät kerrokset voivat olla välittämättä alempien kerrosten järjestelmän toiminnasta. Kuljetuskerros tarjoaa läpinäkyvän kuljetuspalvelun yhteysjaksokerroksille, jotka kommunikoivat keskenään. Kerros huolehtii esimerkiksi siirtopalveluiden optimoinnista, liikenteen valvonnasta ja osoitteen muuttamisesta verkko-osoitteeksi. (Piikkilä & Salhstén 2017, 63.)

Yhteyskerroksen tehtävänä on huolehtia tahdistustoiminnoista ja istunnoista eli viestintäsuhteiden hallinnasta. Näitä ovat huolehtiminen loogisesta yhteydestä esimerkiksi yhden- tai kahden sovelluksen ja käyttäjän välillä sekä purkamisesta, ylläpitämisestä ja rakentamisesta huolehtiminen. Tiivistettynä yhteyskerros huolehtii tapahtumien korkean tason tahdistamisesta. (Piikkilä & Salhstén 2017, 63.)

Esitystapakerroksen tehtävänä on tietojen esittäminen, esimerkiksi salaaminen (kryptaus) ja tiedon tiivistäminen (kompensointi). Tässä kerroksessa huolehditaan siirrettävien tietojen kuvaamisesta sekä suoritetaan tiedon koodimuutokset- ja esitystapa. Kahden ylemmän kerroksen osalta tarkka työnjako on vielä selkiytymätön. Suosituksina on käynnistys, töiden siirto, pääteikäytäntö ja tiedonsiirto. (Piikkilä & Salhstén 2017, 63.)

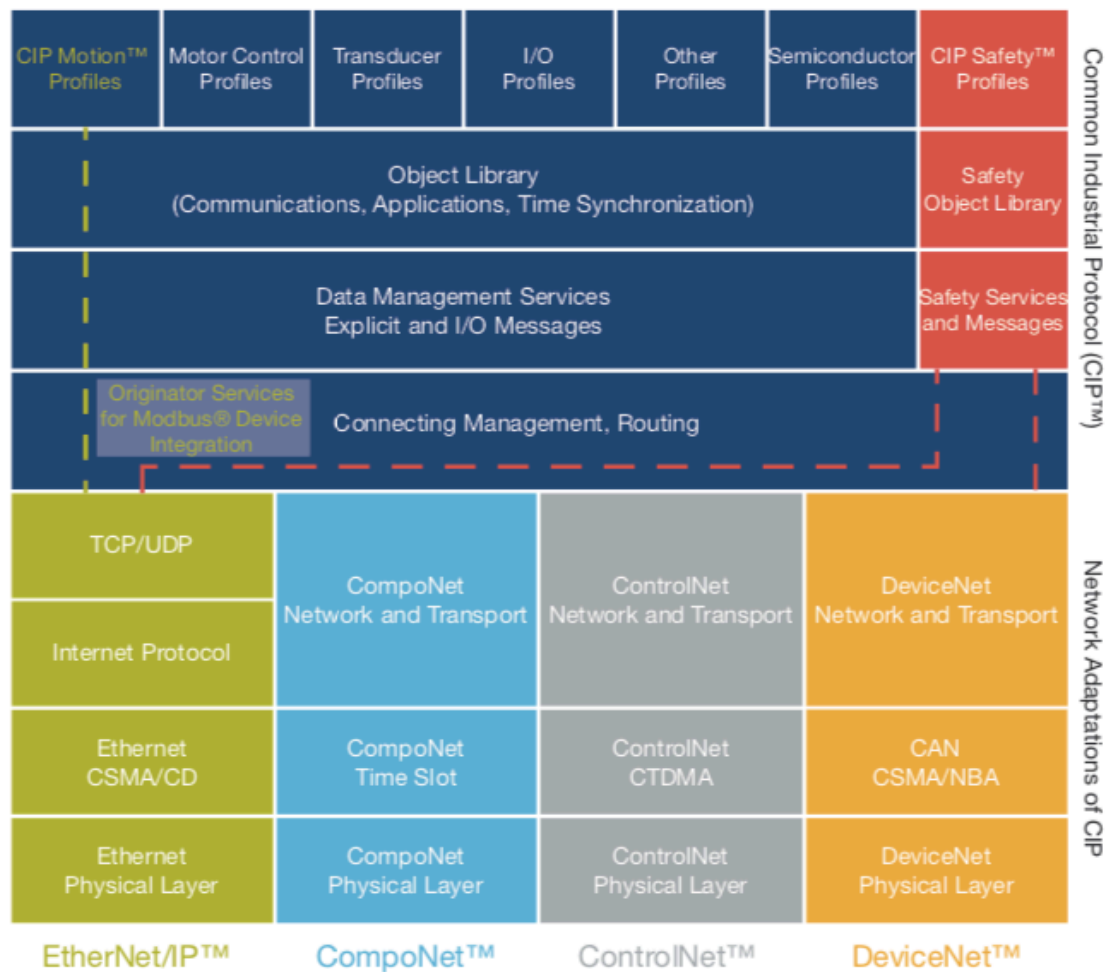
Sovelluskerros sisältää tietoliikennearkkitehtuurin käyttöpalvelut. Sovellusviitekehys määrittää liitännän OSI-viestintäpalveluun. Sen tehtävä on palvella suoraan ylempää kerrosta eli käyttäjää (sovellusta), joita ovat esimerkiksi merkkivalikoima- ja ohjelmointistandardit. Palveluina ovat sähköposti, tiedoston siirto, verkonhallintapalvelut ja sovellusten tarvitsemat tekniset palvelut. (Piikkilä & Salhstén 2017, 63.)

## 4.2 Ethernet/IP-kenttäväylä

Ethernet/IP (Ethernet Industrial Protocol) on vuonna 2001 käyttöönotettu kenttäväyläprotokolla, joka pystyy käsittelemään suuria määriä dataa nopeudella 10 Mbps tai 100 Mbps ja enimmillään 1500 tavua pakettia kohden. Ethernet/IP on yksi suosituimmista teollisuuden ethernet-verkkoratkaisuista ja sitä käytetään monilla teollisuuden toimialoilla. (Technology overview series: Ethernet/IP 2016.)

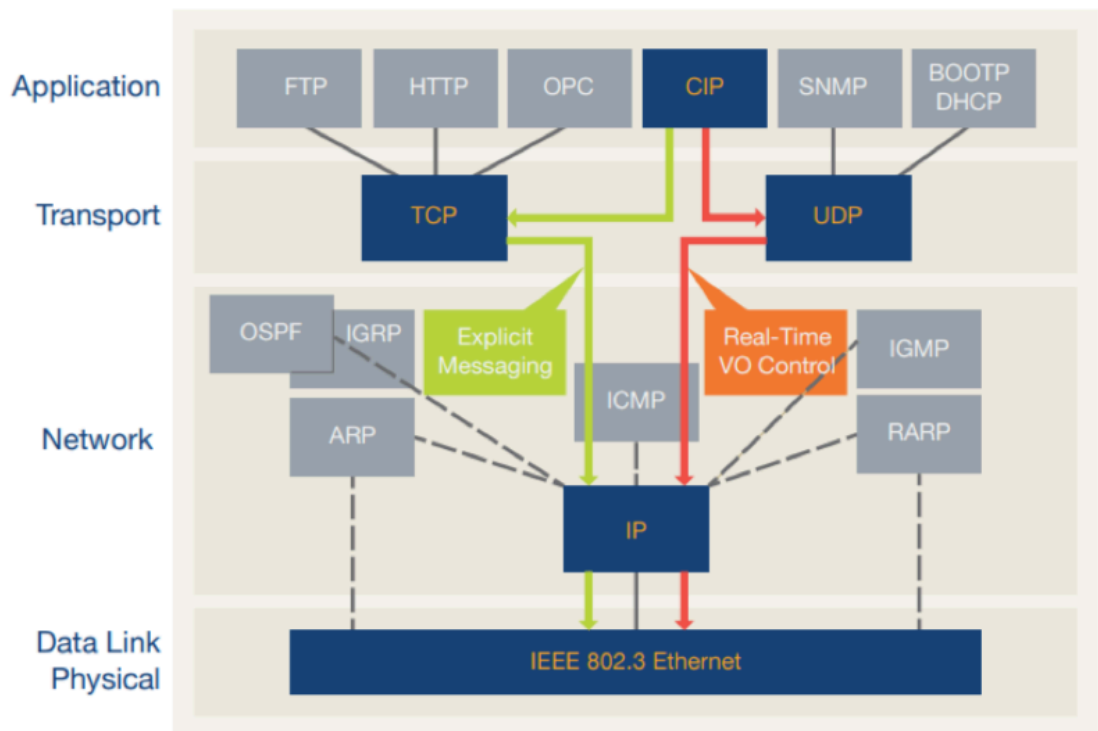
### 4.2.1 Toimintaperiaate

Ethernet/IP-verkko perustuu OSI-malliin, kuten muutkin CIP:n kenttäväylät. Se käyttää OSI-mallin neljää alinta kerrosta sekä ylintä sovelluskerrosta. Ethernet/IP kuuluu Common Industrial Protocol (CIP)-perheeseen. CIP sisältää kattavan valikoiman viesti- ja palvelupaketteja teollisuuden eri sovelluksiin. Käyttö on yleistä esimerkiksi ohjaus-, turva-, liikkeen ohjaus- tai ohjelmoitavien logiikoiden sovelluksissa. Media riippumattomana protokollana CIP tarjoaa käyttäjilleen yhtenäisen viestintäarkkitehtuurin koko tuotantoympäristössä. Ethernet/IP toteuttaa OSI-mallia siirto- ja verkkokerroksella sekä CIP:n tehtäviä yhteyskerroksella ja sen yläpuolella. (Technology overview series: Ethernet/IP 2016.) Kuviossa 4 on havainnollistettu CIP:n toimintaa Ethernet/IP-verkon ja OSI-mallin yhteydessä.



Kuvio 4. Ethernet/IP-verkon CIP:n yhteys OSI-malliin (Technology overview series: Ethernet/IP 2016)

Ethernet/IP käyttää TCP/IP- ja UDP-protokollia. TCP/IP-protokollaa käytetään tiedonsiirtoon laitteiden välillä ja UDP-protokolla pääasiassa I/O-viesteihin. TCP/IP-protokollan TCP-osa on suuntautunut pisteestä pisteeseen (Point To Point) -kuljetusmekanismiin. Solmut vastaanottavat viestit ja kuittaavat viestin lähettäjälle vastaanotetun viestin. Ethernet/IP käyttää TCP/IP:tä kapseloimaan CIP-viestit, joita käytetään yleensä konfigurointi- ja diagnostiikkaviestien lähettämiseen sekä reaaliaikaiseen tiedonsiirtoon laitteiden välillä. TCP/IP-protokollan IP-osa varmistaa paketin reitityksen useiden polkujen läpi. Sillä on kyky lähettää viestejä määränpäähän, vaikka ensisijainen polku olisi häiriintynyt. (Technology overview series: Ethernet/IP 2016.) Kuvio 5 esittää OSI-mallin yhteyden TCP- ja UDP-protokollien viestirakenteen Ethernet/IP-verkossa.



Kuvio 5. Ethernet/IP-verkon yhteys OSI-malliin (Technology overview series: Ethernet/IP 2016)

#### 4.2.2 Osoitteet

Jokaisella Ethernet/IP laitteella tulee olla oma IP-osoite, jotta yhteyden muodostaminen verkkokomponenttien välillä olisi mahdollista. Ethernet/IP käyttää tiedonsiirrossa TCP- ja UDP-protokollia, tulee jokaisella verkon laitteella olla IP-osoite yhteyden muodostamista varten. IP-osoitteet asetellaan jokaiselle verkon laitteelle ja suoritetaan konfigurointi ennen verkon käynnistämistä. (Technology overview series: Ethernet/IP 2016.)

IP-osoitteen lisäksi Ethernet/IP-verkon laitteille on määritettävä MAC-osoite. MAC-osoite koostuu 6-tavun numerosta. MAC-osoitetta käytetään kehyksen lähdeosoite (SA) ja kohdeosoite (DA) kentässä. SA-kenttä osoittaa, mikä solmu kehyksen lähetti ja DA-kenttä osoittaa kehyksen määrän. (Technology overview series: Ethernet/IP 2016.)

Ethernet/IP verkkotopologia voi olla tähti, väylä, puu tai rengasrakenteinen. Rakenteet ovat myös yhdistettävissä samaan verkkoon. Tiedonsiirtoon Ethernet/IP-verkossa voidaan käyttää CAT5-, CAT5e- ja CAT6-kaapeleita. Kaapelin liittimenä voi olla M12-liitin tai RJ-45-liitin. (Technology overview series: Ethernet/IP 2016.)

#### 4.2.3 EDS-tiedosto

EDS- (Electronic Data Sheet) tiedostot ovat yksinkertaisuudessaan ASCII-tiedostoja, jotka kuvaavat, kuinka laitetta voidaan käyttää Ethernet/IP-verkossa. Tiedostossa kuvataan laitteessa käytettävissä olevat attribuutit, objektit ja palvelut. (EtherNet/IP EDS Update 2012.)

EDS-tiedosto välittää laitteen identiteettitiedot, joita tarvitaan laitteen tunnistamiseen. Ethernet/IP-verkossa tiedonsiirto tapahtuu I/O-sovittimien (I/O-Adapter) ja skannereiden (I/O-Scanner) välillä. I/O-sovitin vastaanottaa kommunikaatioyhteyspyynnön I/O-skannerilta. I/O-sovitin lähettää I/O-datan sovitulla nopeudella I/O-skannerille. (EtherNet/IP EDS Update 2012.)

EDS-tiedostoon tallennetun tiedon määrä vaihtelee laitteiden välillä. Toiset valmistajat tallentavat pienimmän mahdollisen määrän tietoa EDS-tiedostoon, kun toiset tallentavat kaikki attribuutit, objektit ja palvelut. EDS-tiedosto yleensä tulee laitteen mukana, kuten muistitikulla. Toisinaan EDS-tiedostot ovat saatavilla valmistajan verkkosivuilta. (EtherNet/IP EDS Update 2012.)

#### 4.3 EtherCAT-kenttäväylä

Vuonna 2003 Beckhoff:n kehittämä EtherCAT (Ethernet Control Automation Technology) on Ethernetin reaaliaikaratkaisu. EtherCAT on IEC-61158 standardissa esitelty protokolla ja sitä käytetään nopeasta vasteajasta johtuen monissa automaatiotekniikan sovelluksissa, kuten I/O:ssa, liikkeenohjauksessa, mittauksissa ja testauksessa. (EtherCAT 2019.)

#### 4.3.1 Toimintaperiaate

EtherCAT standardin protokollaratkaisu on Ethernet standardista poikkeava. Tiedon siirtoverkossa isäntälaitte (master), ja verkon loput laitteet ovat orjalaitteita (slave). Isäntälaitte lähettää yhden kehyksen, joka kulkee jokaisen orjalaitteen läpi ja viimeinen orjalaitelaitte lähettää kehyksen takaisin isäntälaitteelle. Kehyksen kulkiessa orjalaitteen ohi, orjalaitte lukee sille osoitetut tiedot ja kirjoittaa sen tiedot kehykseen. Kehys luetaan ja kirjoitetaan lähes samanaikaisesti, kehyksen ohittaessa orjalaitteen. EtherCAT isäntälaitte on segmentin ainoa solmu, jolla on oikeus lähettää kehys eteenpäin. Orjalaitteet ainoastaan siirtävät kehystä eteenpäin. Kyseisillä toiminnoilla estetään kehyksen prosessoimiseen ja vastaanottamiseen käytettävää aikaa. (EtherCAT 2019.)

EtherCAT-verkko perustuu muiden Ethernet-väylien tavoin OSI-malliin. EtherCAT-verkko käyttää OSI-mallin kahta alinta kerrosta sekä sovelluskerrosta. Kahden alimman kerroksen tehtävänä on huolehtia aikakriittisistä toiminnoista kuten, muistitoiminnoista ja kehysten reitittämisestä. Sovelluskerrosta käytetään syklisten ja asyklisen tietojen välittämiseen sekä sovellustoimintaan. EtherCAT-verkossa tämä kerros toteutetaan yleisesti mikrokontrollerin laiteohjelmistossa. Toinen vaihtoehto on käyttää kehyksen kuljettamiseen UDP-protokollaa (User Datagram Protocol), joka mahdollistaa normaalin IP-reitityksen käytön. UDP-protokollaa käytetään vähemmän aikakriittisissä sovelluksissa, koska tiedonsiirtonopeus laskee huomattavasti sen käytössä. EtherCAT-verkko ei tarvitse ulkoisia kytkimiä, vaan jokainen laite sisältää kaksi RJ-45-porttia. Yksi portti on kytketty verkon edelliseen solmuun ja yksi on vapaana seuraavan solmun kytkemistä varten. (EtherCAT – the Ethernet fieldbus n.d.)

#### 4.3.2 Osoitteet

EtherCAT-väylässä voi olla vain yksi isäntälaitte. Isäntälaitte käyttää vakiona MAC-osoitejärjestelmää ilman ylimääräistä viestintäprosessoria. MAC tarjoaa 100Mbit/s kaksisuuntaisen rajapinnan laitteiden välille. Isäntänä voi toimia mikä tahansa laitteistoalusta, jossa on tarvittava ohjelmisto. (EtherCAT – the Ethernet fieldbus n.d.)



EtherCAT- orjalaitteiden osoitteet voidaan määrittää automaattisesti tai kiinteästi. Osoitekonfiguraatiot määritellään isäntälaitteen asetuksista. Automaattisessa osoitekonfiguraatiossa ensimmäinen orjalaite saa osoitteen 0 seuraava 1 ja niin edelleen. Kiinteän osoite määritetään kullekin orjalaitteelle erikseen. Osoitteita voi orjalaitteilla olla yhteensä 65535 eli 16 bitin verran. Orjalaite saa osoitteen väylän käynnistymien jälkeen ensimmäisellä suorituskierröksellä ja osoite tallennetaan laitteen pysyväismuistiin. (EtherCAT – the Ethernet fieldbus n.d.; Principle of operation 2016.)

#### 4.3.3 Turvallisuusprotokolla

Normaalin tiedonsiirron lisäksi EtherCAT hyödyntää FSoE (Fail Safe over EtherCAT) -avointa turvallisuusprotokollaa. FSoE on kehitetty standardin IEC 61508 mukaisesti ja se on sertifioitu ja standardisoitu standardiin IEC 61784-3. Protokolla sopii käytettäväksi enintään SIL 3-tason turvallisuussovelluksiin. Tämän vuoksi EtherCAT:n käyttö turvalogiikoiden tiedonsiirto-protokollana on yleistä.

EtherCAT-vakioviestijärjestelmä käyttää yhtä kanavaa vakio- sekä turvallisuuskriittisen datan siirtämiseen. EtherCAT-kehysturvallisuus, jota tunnetaan ”turvasäiliönä” sisältää turvallisuuskriittisiä prosessitietoja ja lisätietoa turvatietojen suojaamiseksi. Turvasäiliöt kuljetetaan osana viestinnän prosessitietoja. Turvasäiliö on analysoitu turvallisesti laitteissa sovellustasolla. (EtherCAT:Technology n.d.)

## 5 Teollisuusrobotit

Teollisuusrobotti on sähköisesti, paineilmalla tai hydraulisesti liikkuva, kappaleita tai työkaluja käsittelevä tietokoneohjattu laite, jonka liikeradat ovat ohjelmallisesti muutettavissa. Robotin määrittelyyn käytetään vähintään kolmea vapaasti ohjelmoitavaa liikeakselia ja yhtä työkalua. Pääkäyttötarkoituksena kappaleen käsittely, kokoonpano, hitsaus, paketointi ja pakkaus. Robotilla pyritään vähentämään ihmiselle raskaita tai vaarallisia työvaiheita. Robotteja käytetään tuotannon tehostamiseen ja saa-

vuttamaan tasaisempi tuotannon laatu. Teollisuusrobotteja on mekaaniselta rakenteeltaan useanlaisia ja robottityyppi on valittava sovelluskohteen mukaisesti. (Keinänen ym. 2007, 259.)

## 5.1 Teollisuusrobotin tyypit

**Nivelrobotti** on teollisuudessa yleisimmin käytetty robottityyppi, koska sitä voidaan käyttää monissa eri teollisuuden työtehtävissä. Siinä on yleisesti 6- liikkuvaa akselia ja sen liikealue on pallomainen. Robotin tarttujan asento on vapaasti ohjelmoitavissa mihin asentoon tahansa robotin liikealueella. 6- akselisia robotteja on saatavilla moniin eri sovelluksiin kappaleenkäsittelykyvyn sekä ulottuvuuden mukaisesti. (Keinänen ym. 2007, 260)

**SCARA-robotti** (Selective Compliance Assembly Robot Arm) luetaan nivelrobotteihin ja on yleisesti teollisuudessa käytetty robottityyppi. SCARA-roboteissa on kaksi tai kolme samassa tasossa liikkuvaa niveltä sekä yksi lineaarinen pystyliike. Näillä ominaisuuksilla saavutetaan suuret liikenopeudet ja hyvä liiketarkkuus. SCARA-robottia käytetään kappaletavara automaatioissa tuotteiden kokoonpanoon ja pinoamiseen. (Keinänen ym. 2007, 259)

**Lineaarirobotti** eli lineaarisesti liikkuva robotti toimii kolmen suorakulmaisen vapausasteen X,Y ja Z mukaisesti. Liikuteltava työkalu voi olla nivelöity, joka mahdollistaa kappaleen kääntämisen eriasentoihin. Lineaarirobotteja käytetään pakkaus- ja lavaustehtävissä sekä ruiskuvalukoneiden palvelustehtävissä. (Kuivanen 1999, 16.)

**Yhteistyörobotit** ovat tarkoitettuja työskentelemään yhdessä operaattoreiden kanssa. Yhteistyörobotit eivät tarvitse erillisiä turva-aitoja ympärilleen, vaan käyttäjäturvallisuus on otettu huomioon robotin sisäisillä ominaisuuksilla. Yhteistyörobotin työtehtävät keskittyvät monesti prosessin lopputuotantoon valmiiden tuotteiden nouto- ja pakkaustehtäviin. (Shikany 2014.) Robotille voidaan luoda erilaisia turva-alueita. Operaattorin saapuessa turva-alueelle robotti hidastaa nopeutta ja pysähtyy

operaattorin saapuessa liian lähelle. Robotin jokaisessa akselissa on omat törmäystunnistimet, jotka aktivoituvat törmäyksen tapahtuessa. (Bélander-Barette 2015.)

## 5.2 Teollisuusrobotin valinta

Robottivalmistajia ja robottityyppejä on useita moniin eri käyttötarkoituksiin. Robotti valitaan aina tapauskohtaisesti käyttötarkoituksen mukaisesti. Robottien ominaisuudet ja ohjelmaympäristöt vaihtelevat eri valmistajien välillä. Tavallisesti robottia valinnassa huomioidaan sen työalue sivulta ja päältä, kantokyky (payload), nivelten suurimmat nopeudet sekä paino. Toistotarkkuudella tarkoitetaan tilastollista tarkkuutta, jolla robotin työkalun piste palaa takaisin aiemmin opetettuun pisteeseen. Robotin absoluuttinen tarkkuus kertoo millä tarkkuudella robotin runkoon sidotussa koordinaatistossa robotti saadaan haluttuun pisteeseen. Absoluuttinen tarkkuus on usein jopa kymmeniä kertoja epätarkempi, kuin toistotarkkuus. (Kuivanen 1999, 14.)

Tavallisimmat robotit ovat 6- vapausasteen teollisuusrobotit, jotka ovat monikäyttöisiä. Työkalu voidaan asettaa mihin tahansa asentoon sen liikealueella. 6- vapausasteen robotin hankintaa voidaan perustella työkaluaseman erityismuodoilla. (Kuivanen 1999, 17-18.)

## 5.3 Teollisuusrobotin ohjainyksikkö

Teollisuusrobotin ohjainyksikkö koostuu keskusyksiköstä, käsiohjaimesta, ohjelmuistista ja akselikohtaisista paikoitusjärjestelmistä sekä ulkoisista liitynnöistä. Ohjainyksikön tehtävänä on tulkita ohjelmakomennot liikekäskyiksi ja ohjata sen perusteella liikeakseleita. Robottiohjaimen on pystyttävä ohjaamaan jokaista liikeakselia samanaikaisesti ja oikeassa asemassa sekä oikealla nopeudella, jotta kuudella akselilla varustettu robotti voi suoriutua suoraviivaisesta liikkeestä. Robottiohjaimen on kyettävä ohjaamaan jokaista liikeakselia tuhansia kertoja sekunnissa. Tästä syystä robottiohjaimet ovat suuritehoisia prosessitietokoneita. Jokaisella robottivalmistajalla

on oma ohjainyksikkö, eikä kaikille roboteille yleispätevää ohjainta ole saatavilla. Ohjauksyksiköitä on saatavilla suurempina versioina, joissa on varattu kytkentätilaa ulkoisten laitteiden kytkennöille. (Keinänen ym. 2007, 261.)

## 5.4 Teollisuusrobotin koordinaatistot

Teollisuusrobotin koordinaatistolla on tarkoituksena kertoa robotille, minkä ympäristön suhteen robotin akseleita liikutetaan. Yleisimmät koordinaatistot ovat maailma-, työkalu- ja peruskoordinaatisto.

**Maailmakoordinaatisto** on sidottu robotin toimintaympäristöön ja se on muutettavissa sovelluksen mukaisesti. Maailmakoordinaatiston käyttö on suositeltavaa, kun robotin asemat määräytyvät ulkoisten laitteiden perusteella. Esimerkiksi robotin ollessa asennettuna seinälle tai solun kattoon, on hyvä muuttaa koordinaatisto työaseman suuntaisesti. (Keinänen ym. 2007, 260.)

**Peruskoordinaatisto** on robotin jalan suuntaisesti sidottu X, Y ja Z koordinaatisto. X ja Y akselit muodostavat vaakasuoran tason robotin alapinnan tasolle. Z akseli on pystyakseli, joka kulkee robotin 1- akselin, eli pyörähdyskeskipisteen kautta. (Keinänen ym. 2007, 261.)

**Työkalukoordinaatisto** on suorakulmainen koordinaatisto, jonka origo on sidottu oletuksena robotin työkalun laippaan. Työkalupiste on muutettavissa käytettävän työkalun mukaisesti. Työkalukoordinaatistossa X, Y ja Z liikkeet ovat aina työkalun suuntaisia liikkeitä. Esimerkiksi työkalupisteen määrittämisen pihtitarttujan päähän auttaa liikeratojen ohjelmoinnissa siten, että robottia voi pyörittää työkalupisteen ympärillä vapaasti työkalun kärjen pysyessä paikallaan. (Keinänen ym. 2007, 261.)

## 5.5 Teollisuusrobotin ohjelmointi

Ohjelmoinnilla robotille opetetaan tehtävät, joita sen tulee suorittaa luodun ohjelman mukaisesti. Ohjelmointitapoja ja ohjelmointi ohjelmistoja on erilaisia riippuen

robottivalmistajasta. Yleisimmät ohjelmointitavat ovat opettamalla ohjelmointi, off-line ohjelmointi ja johdattamalla ohjelmointi.

**Opettamalla ohjelmoinnissa** robotin työkalu ajetaan käsiohjaimella haluttuun paikkaan. Tämän jälkeen paikka tallennetaan robotin muistiin. Opetettavat paikkapisteet nimetään kuvaavasti, jolloin ohjelman luku ja pisteen uudelleenkäyttö ohjelmassa helpottuu. Jokaiselle pisteelle on annettava liikekomento sen mukaisesti millä nopeudella ja millä tavalla pisteeseen halutaan siirtyä. Käsiohjaimella ohjelmointi soveltuu yksinkertaisiin sovelluksiin, missä liikekäskyt ovat pisteestä pisteeseen komentoja. Ohjelmointitapaa on käytetty yleisesti pakkaus ja paletointisovelluksissa. (Keinänen ym. 2007, 262.)

**Off-line ohjelmointi** tarkoittaa tietokoneella tapahtuvaa ohjelmointia robottivalmistajan ohjelmiston avulla ilman robottiin luotua yhteyttä. Monesti robottiohjelmia voi testata simuloidun 3D-mallin avulla, jotka toimivat vastaavasti, kuin todellinen robotti. Off-line ohjelmointi on yleisin ohjelmointitapa ja sen etuna on havainnollinen käyttöliittymä ja monipuoliset liikekäskyt. Robottia voi simuloida ja ohjelmoida robotin liikkeessa. (Keinänen ym. 2007, 262.)

**Johdattamalla ohjelmoinnissa** robotin liikeakselit vapautetaan ja ihminen voi liikuttaa robotin työkalua haluttua liikerataa pitkin. Robottiohjaimen ollessa opetustilassa se lukee liikeradan muistiinsa ja opetuksen jälkeen pystyy toistamaan opetetun liikeradan. Useat yhteistyörobotit ovat johdattamalla ohjelmoitavia. Etuna on nopea ja yksinkertainen liikeratojen muokattavuus. (Keinänen ym. 2007, 262.)

## 5.6 Teollisuusrobottitarrain ja työkalut

Teollisuusrobotin yleisin työkalu on tarrain. Työkaluiksi määritellään esimerkiksi prosessiin osallistuvat ja robotilla käsiteltävät maalausruisku tai hitsauspistooli. Tarrain voi olla pneumaattinen, sähköinen tai hydraulinen toimilaite ja sen ohjaus voi olla yksi- tai kaksitoimen. Yleensä se varustetaan magneettitunnistimella, jolloin siitä saa-

daan tilatieto automaatiojärjestelmään. (Keinänen ym. 2007.) Pneumaattista tarttujaa käytetään sen nopean toiminnan ja pehmeän kiinnitystavan takia. Tarttumismenetelmät jakautuvat kolmeen yleiseen ryhmään: mekaaniseen-, imu- ja tyhjiö- sekä sähkömagneettitartuntaan. (Kuivanen 1999, 60.)

**Mekaaninen tarra**in on yleisin ja yksinkertaisin tarttumismenetelmä. Mekaaninen tartunta toteutetaan pihtitarttujalla, jonka liikettä voidaan ohjata pneumaattisesti, sähköisesti tai hydraulisesti. Tarkoissa asennuksissa pihti muotoillaan käsiteltävän kappaleen ottopinnan mukaisesti, jolloin kappale saadaan paikoitettua tarttujaan. (Kuivanen 1999, 60.)

**Imu- ja tyhjiötarra**imen kappaletta käsitellä imukuppien avulla. Imutartunnassa kappaletta käsitellään yleensä vain yhdeltä suunnalta. Tarttumismenetelmää käytetään sovelluksissa, jossa mekaanisen tartunnan käyttö on hankalaa. Imukupit eivät naarmuta käsiteltävää pintaa. Imukupit vaativat tasaisen, puhtaan ja sileän ottopinnan. (Kuivanen 1999, 63.)

**Sähkömagneettitarra**imella voidaan käsitellä vain magneettisia kappaleita. Etuna on, että pienikokoisella tarraimella voidaan saada aikaan suuri nostovoima. Nostovoimaan vaikuttaa kuitenkin käsiteltävän pinnanlaatu, muoto, lämpötila ja ilmarako. (Kuivanen 1999, 64.)

## 5.7 Digitaaliset tulo- ja lähtösignaalit

Ohjelmoitavan logiikan tavoin, teollisuusrobotit sisältävät sähköisiä tulo- ja lähtösignaaleja, joita voidaan käsitellä robotin ohjelmassa. Tulosignaalien avulla robotti lukee antureiden tilatietoja. Lähtösignaaleilla robotti voi ohjata oheislaitteita. Yksinkertaisissa järjestelmissä, robotilla voi korvata ohjelmoitavan logiikan käytön. Signaaliviesityyppinä voi käyttää analogisia, digitaalisia tai väyläviestejä. (Kuivanen 1999.)

Digitaalisten tulo- ja lähtöpiirien käyttö on yleistä nykypäivän robottisovelluksissa. Puolijohdetulot (+24V) ovat yleensä galvaanisesti erotettuja valodiodituloja. Tulotietoon kytketään 24 voltin jännite, jolloin tulo aktivoituu. Tulot on yleensä jaoteltu kahdeksan tulon ryhmiksi, joilla on yhteiset signaalimaat. Tulopiireissä käytetään releitä galvaanista erotusta varten. Tulosignaalin jännite ohjaa tulopiirin releen käämiä, jolloin rele saa ohjauksen. Releen kärkitieto ohjaa tilan eteenpäin ohjausjärjestelmälle.

Puolijohdelähtöjen ohjauksessa käytetään joko PNP- tai NPN-transistoria, joiden ohjauspiirit on erotettu galvaanisesti optoerottimen avulla. Puolijohdelähdöt sijaitsevat piirilevyllä, joten niiden maksimilähtövirta on yleensä vain puoli ampeeria. Tämä riittää yleensä yhden venttiilin ohjaukseen. Induktiivisia kuormia ohjattaessa, on lähtöaste suojattava negatiivisilta virtapiikeiltä diodin avulla. (Kuivanen 1999, 52-53.)

## 6 Koneturvallisuus

Tuhansien vuosien ajan ihmiset ovat käyttäneet koneita, joiden toimintaperiaate perustuivat tuuli- ja vesivoimaan sekä ihmisten ja eläinten lihasvoimaan. Suomessakin sahat ja teollisuuslaitokset perustettiin jokien ja koskien varsiin, jolloin koneiden käyttöön voitiin käyttää koskissa virtaavan veden voimaa. Teollistumisen alkuvaiheessa ei ollut olemassa lainsäädäntöjä eikä valvontaa, jotka aiheuttivat työpaikoilla työkyvyttömyyttä ja jopa kuolemia. Teollisuuden maat ryhtyivät saattamaan voimaan työolosuhteita sääteleviä lakeja, jolloin perustettiin virkoja valvomaan lakien noudattamista. (Siirilä & Tytykoski 2016, 24-25.)

Suomessa nykyisin voimassaolevat koneturvallisuutta koskevat säädökset perustuvat direktiiveihin, jotka saatetaan voimaan yleensä valtioneuvoston asetuksina. Koneita koskevat direktiivit ovat pääasiassa valmistajia koskevia tuotedirektiivejä tai työnantajaa koskevia työolosuhtedirektiivejä. Nykyisin vahvistetut direktiivit ovat Euroopan Unionin direktiivejä, joiden tunnus on EU. Tuotedirektiivit asetetaan voimaan sellaisenaan koko Euroopan talousalueelle. Tuote voidaan saattaa markkinoille, valmistaa

ja ottaa käyttöön sekä siirtää toiseen maahan tai myydä, kun tuote on sitä koskevien direktiivien mukainen. (Siirilä & Tytykoski 2016, 29.)

## 6.1 Standardit

Standardeilla tarkoitetaan sopimuksia, jotka mahdollistavat tuotteiden yhteensopivuuden eri valmistajien välillä. Standardijärjestöjä on useita, kuten sähköalan kansainvälinen IEC (International Electrotechnical Commission), joka perustettiin 1906. (Piikkilä & Salhstén 2017, 24.) Kansainvälisiä standardijärjestö ISO (International Standardization Organization), joka perustettiin vuonna 1946. Suomen Sähköteknillinen Standardikomitea asetettiin laatimaan sähköalan SFS-standardeja ja muita suosituksia vuonna 1926. Vuonna 1949 Suomi liittyi kansainvälisen standardijärjestön IEC:n jäseneksi. (Siirilä & Tytykoski 2016, 85-86.)

Direktiivejä täsmentävät yksityiskohtaiset tekniset turvallisuusvaatimukset esitetään eurooppalaisessa EN-standardissa. Yleensä yhdenmukaistettu standardi liittyy vain yhteen tiettyyn direktiiviin. Tällöin standardien vaatimukset katsotaan täsmentävän kyseistä direktiiviä tai sen jotain vaatimusta. Koneturvallisuusvaatimukset ovat melko yksityiskohtaisia mutta niihin jää silti tulkinnanvaraa. Standardeista löytyy täsmennys lähes kaikkiin koneasetuksiin liittyen. (Siirilä & Tytykoski 2016, 93.)

Vaikka yhdenmukaistettujen standardien tarkoituksena on täsmentää konedirektiiviä, on niistä kumminkin mahdollisuus poiketa. Poikkeavaa ratkaisua käytettäessä on huolehdittava, että standardissa määritetty turvallisuustaso saavutetaan. Poikkeavan ratkaisun osoittaminen standardin tasoa vastaavaksi ei aina ole yksinkertaista. Tästä syystä standardia kannattaa noudattaa aina, kuin mahdollista. (Siirilä & Tytykoski 2016, 94-95.)

## 6.2 Riskien arviointi

Koneiden riskien arvioimiseen on käytössä kymmeniä erilaisia menetelmiä. Laitevalmistajan on valittava omaan tarkoitukseensa sopiva menetelmä, jonka käyttö tuntuu



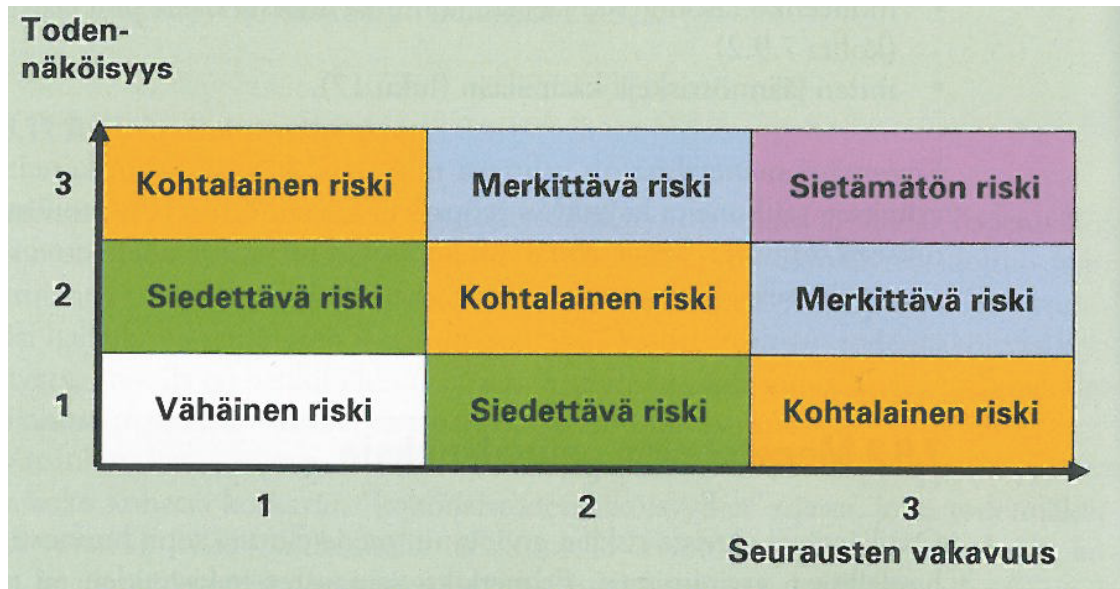
luontevalta ja joka ei systemaattisesti anna liian pieniä arvoja. Kaikille riskien arvioimismenetelmille yhteistä on seuraavien riskien arvioinnin yleisen menettelytavan käyttäminen:

1. Tunnistetaan koneen ominaisuudet ja niihin liittyvät vaarat.
2. Arvioidaan jokaisesta vaaratekijästä aiheutuvien haitallisten seurausten vakavuus.
3. Arvioidaan haitallisten seurausten toteutumisen todennäköisyys.
4. Määritetään riskin suuruus seurausten vakavuuden ja niiden toteutumisen todennäköisyyden perusteella.
5. Arvioidaan riskin merkittävyys: onko riski riittävän pieni vai onko riskiä pienennettävä.
6. Jos riskiä on pienennettävä, päätetään pienentämistavoista ja suoritetaan arviointi uudelleen.
7. Käsitellään jäännösriskit.

(Siirilä & Tytykoski 2016, 222-223.)

Yleisimmin käytetty menetelmä riskienhallintaan on matriisimenetelmä, jossa verrataan taulukossa seurausten vakavuutta seurausten todennäköisyyteen. Kun seurausten vakavuus ja todennäköisyys on valittu, matriisissa päädytään tiettyyn riskitasoon.

(Siirilä & Tytykoski 2016, 224)



Kuvio 6. Matriisitaulukko riskien arviointiin. (Siirilä & Tytykoski 2016, 224)

Kuvion 6 mukaisesti seurausten ja todennäköisyyden riskit on jaettu kolmeen eri tasoon sekä riskin suuruus viiteen tasoon. Esimerkiksi seurausten vakavuuden ollessa 3 tällöin päädytään kohtalaiseen riskiin, vaikka todennäköisyys olisi pienin mahdollinen. Taulukko 2 määrittää toimenpiteet riskien arviointiluokan mukaisesti. Kohtalaisessa, merkittävässä ja sietämättömässä riskissä olisi pienennettävä tai kohennettava koneen turvallisuusominaisuuksia. Mikäli seurausten vakavuutta ei voi pienentää, on tällöin pyrittävä pienentämään todennäköisyyttä. (Siirilä & Tytykoski 2016, 224.)

Taulukko 2. Suoritettavat toimenpiteet riskien arviointiluokan mukaisesti. (Siirilä & Tytykoski 2016, 228)

RISKI	TARVITTAVAT TOIMENPITEET
Sietämätön	Riskiä on pienennettävä. Suunnittelua on jatkettava ja koneen ominaisuuksia on muutettava tai koneeseen on lisättävä suojuksia tai turvalaitteita tai muita turvallisuusominaisuuksia.
Merkittävä	
Kohtalainen	
Siedettävä	Kone voidaan valmistaa ja ottaa käyttöön. Aktiivinen turvallisuuden seuranta ja kokemusten kerääminen on tarpeen.
Vähäinen	Kone voidaan valmistaa ja ottaa käyttöön.

Koneita suunnitellessa on monesti mahdollista muuttaa koneen ominaisuuksia siten, että riskit saadaan vähäisiksi tai siedettäviksi. Yleensä koneen voimaa, nopeutta tai muita ominaisuuksia ei ole mahdollista muuttaa. Tässä tapauksessa on pyrittävä pienentämään todennäköisyyttä. Taulukko 2. Esittää toimenpiteet, mitkä ovat tarpeen milläkin riskitasolla. (Siirilä & Tytykoski 2016, 228.)

Riskien arvioinnit on dokumentoitava koneturvallisuuden perusstandardin mukaisesti. Standardi määrittää dokumentoinnille sisällölle muun muassa seuraavia vaatimuksia:

- koneen tiedot, jolle arviointi on tehty (tekniset tiedot, koneen kuvaus, käyttötarkoitus jne.)
- tehdyt oletukset (koneen tai sen rakenneosien käyttöikä, kuormitukset jne.)
- tunnistetut vaaratekijät, vaaratilanteet, riskit ja seurausten toteutumisen todennäköisyys
- valitut turvatoimenpiteet tunnistettujen vaarojen poistamiseksi tai riskien pienentämiseksi
- jäännösriskit
- arvioinnin johtopäätös
- mahdollinen täsmentävä aineisto

Dokumentointi on toteutettava selkeästi ja yksityiskohtaisesti, jotta arviointiin osallistuneet henkilöt saavat oikean käsityksen arvioinnin tuloksista. Arvioinnin ei tarvitse olla yhteen paikkaan tulostettuna tai koottuna. Yhteenvedoasiakirjassa on oltava viitteet, jotta lisädokumentointi on löydettävissä helposti. (Siirilä & Tytykoski 2016, 251-252.)

### 6.3 Luokat, suoritustasot ja turvallisuuden eheytytaset

Koneen turvatoimintoja suunniteltaessa on toivottavaa, että turvatoiminnot olisivat käytettävissä, kun niitä tarvitaan. Nykyisin turvatoiminnot toteutetaan ohjausjärjestelmän avulla. Ohjausjärjestelmien on toimittava luotettavasti siten, että vikaantumi-

nen on epätodennäköistä. Tärkeiden turvallisuustoteutuksien osalta ohjausjärjestelmän on tarvittaessa pystyttävä toimimaan, vaikka järjestelmässä ilmenisi vika. (Siirilä & Tytykoski 2016, 560.)

Ohjausjärjestelmän luotettavuutta turvatoiminnon suorittamista vikaantuneena käsitellään useissa standardeissa, joita verrattu kuviossa 7. Eurooppalainen standardijärjestö nimesi järjestelmät turvallisuusluokkiin B, 1, 2, 3 ja 4 niiden rakenteen ja luotettavuuden perusteella. Kansainvälinen standardijärjestö IEC nimesi järjestelmät turvallisuuseheytyksen SIL (Safety Integrity Level) tasoihin 1, 2, 3 ja 4, sen mukaisesti miten vaarallinen vikaantuminen arvioidaan. SIL-tasoa 4 ei normaalisti käytetä koneturvallisuussovelluksissa. Eurooppalaisen ohjausjärjestelmästandardin uusimisen yhteydessä siitä tuli kansainvälinen ISO-standardi. Luokkien vaatimuksia täsmennettiin IEC-standardimallin mukaiseksi vikaantumisen todennäköisyyttä tuntia kohden. Tätä todennäköisyyttä nimitetään ISO-standardissa PL (Performance Level) a, b, c, d ja e suoritustasoksi. (Siirilä & Tytykoski 2016, 560-561.)

Suoritustaso (PL)	Luokka (kategoria)	Keskimääräinen vaarallisen vian todennäköisyys tunnissa (1/h)	Vastaavuus eheystasoihin (SIL)
a	B	$10^{-5} \leq PFH_d < 10^{-4}$	ei
b	1-2	$3 \cdot 10^{-6} \leq PFH_d < 10^{-5}$	1
c	1-3	$10^{-6} \leq PFH_d < 3 \cdot 10^{-6}$	1
d	3	$10^{-7} \leq PFH_d < 10^{-6}$	2
e	4	$10^{-8} \leq PFH_d < 10^{-7}$	3

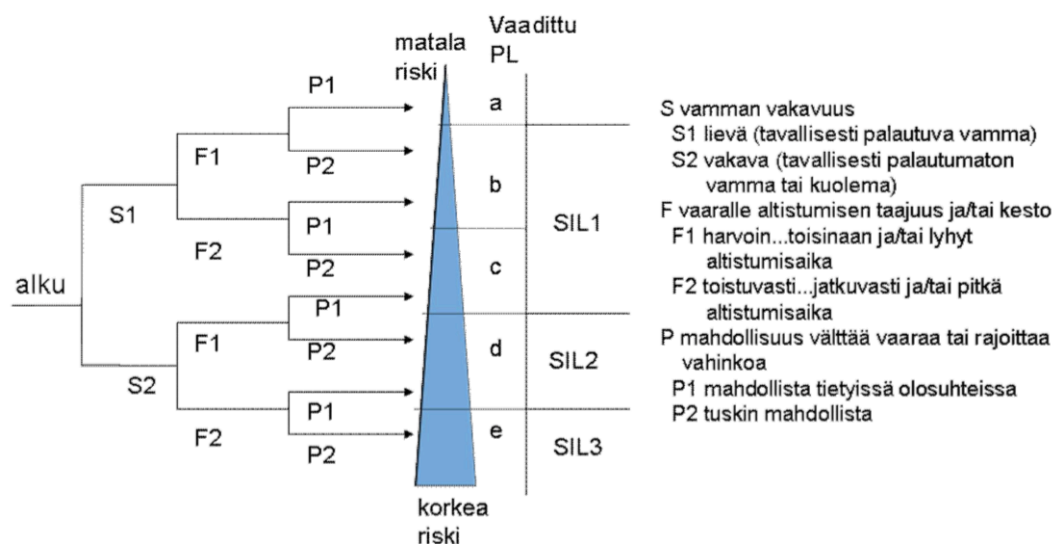
Kuvio 7. Suoritustaso, luokan vastaavuus ja turvallisuuden eheytystaso SIL. (Malm, Venho-Ahonen & Vanhala 2010)

Ohjausjärjestelmän toiminnallisen turvallisuuden tasoa voidaan arvioida riskin arvioinnin tai standardin perusteella. PL-turvatasot määritellään konekohtaisissa ISO-

standardeissa. Suunnittelija on kuitenkin arvioitava, onko standardin luokitus kyseiseen kohteeseen sopiva. Standardeista voidaan poiketa hyvillä perusteilla. Mikäli standardista ei löydy turvatoiminnolle PL-tasoa, on päätös tehtävä riskien arvioinnin perusteella. SFS EN ISO 13849-1 esittää riskinarviointiin riskigraafia, jossa PL-taso määritellään kolmella kysymyksellä:

1. Mikä on vamman vakavuus?
2. Mikä on vaaralle altistumisen taajuus (tai kesto)?
3. Onko mahdollista välttää vaaraa tai pienentää vahinkoa?

Kuvion 8. menetelmä on tarkoitettu PL-tason arviointiin.



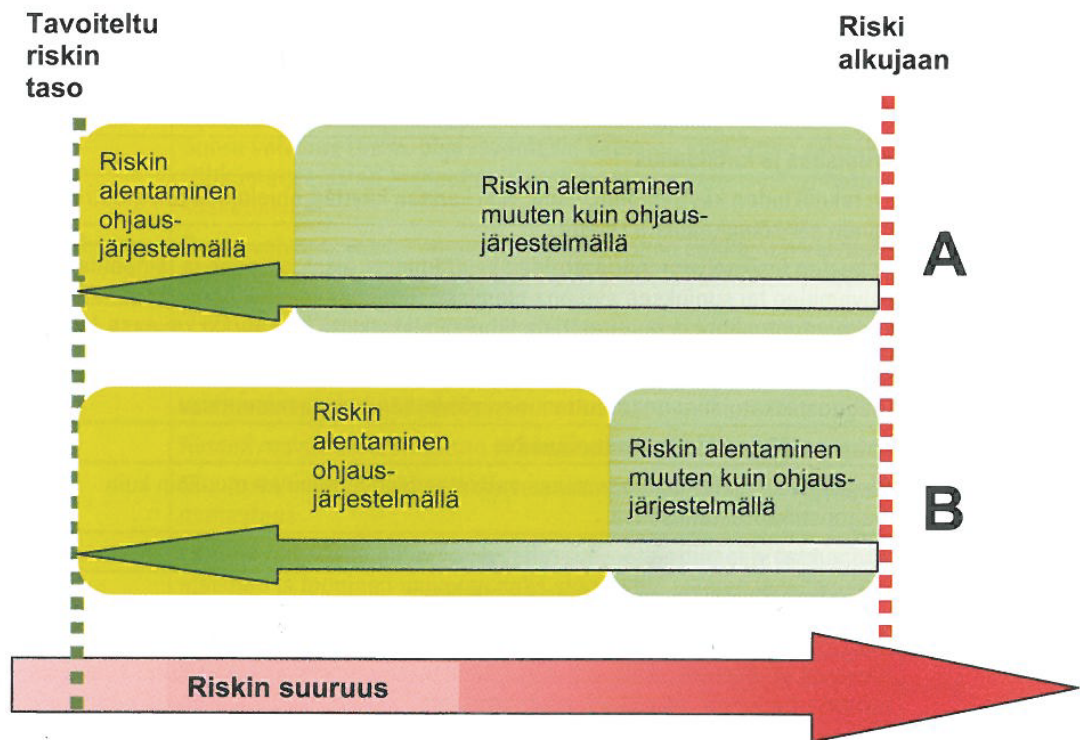
Kuvio 8. PL-tason arviointi. (Malm, Venho-Ahonen & Vanhala 2010)

SIL- tason määrittämiseen käytetään SFS EN 62601 standardin määrittävää matriisi-menetelmää kuvion 9 mukaisesti. Menetelmässä arvioidaan seurausten todennäköisyyttä, taajuutta, vakavuutta ja välttävyyttä. Menetelmässä lasketaan saadut pisteet yhteen (ei vakavuus), joka mukaisesti taulukko näyttää saavutetun pistemäärän perusteella vaaditun SIL-tason.

Seuraukset		Vakavuus Se	Luokka CI				
			3-4	5-7	8-10	11-13	14-15
Kuolema, näön tai käden menetys		4	SIL 2	SIL 2	SIL 2	SIL 3	SIL 3
Palutumaton, sormen menetys		3			SIL 1	SIL 2	SIL 3
Palautuva, sairaanhoito		2				SIL 1	SIL 2
Palautuva, ensiapu		1					SIL 1
Taajuus ja kesto Fr		Vaarallisen tapahtuman todennäköisyys, Pr		Vältettävyyden Av.			
<= 1 tunti		5	Erittäin todennäköinen	5			
> 1 t - <= päivä		5	Todennäköinen	4			
>1 päivä - <=2 viikkoa		4	Mahdollinen	3	Mahdoton		5
>2 vko - <=1 vuosi		3	Harvoin	2	Mahdollista		3
> 1 vuosi		2	Ei huomioitava	1	Todennäköistä		1

Kuvio 9. Matriisimenetelmä SIL-tason määrittämiseksi. (Malm, Venho-Ahonen & Vanhala 2010)

Ohjausjärjestelmää käytetään yleisesti turvatoimintojen toteuttamisessa. Merkittävä osa riskien alentamisesta tehdään kuitenkin muilla toimenpiteillä, kuin ohjausjärjestelmän turvatoiminnoilla. Esimerkiksi kiinteiden suojien käyttö koneen riskienhallinnassa on tärkeää mutta ohjausjärjestelmällä ei voida vaikuttaa tällaisen riskin alentamistoimenpiteeseen. Samoin koneen kompastumis- ja putoamisriskiä alentavat työtasot, portaat ja kaiteet ovat ohjausjärjestelmästä riippumattomia toimenpiteitä riskien hallinnassa.



Kuvio 10. Ohjausjärjestelmän merkitys riskien hallinnassa. (Siirilä & Tytykoski 2016, 570)

Kuvion 10. Mukaisesti osaa riskeistä voidaan pienentää ohjausjärjestelmän turvatoimilla tai ohjausjärjestelmästä riippumattomilla toimenpiteillä. Vaihtoehdossa A suurempi osa riskeistä on poistettu ohjausjärjestelmästä riippumattomilla toimenpiteillä, jolloin ohjausjärjestelmällä toteutettujen turvallisuustoimien merkitys jää pienemmäksi. Vaihtoehdossa B ohjausjärjestelmällä on toteutettu suurin osa riskien alentamisesta, jolloin ohjausjärjestelmästä riippumattomat toimenpiteet jäävät vähäisemmiksi. (Siirilä & Tytykoski 2016, 570.)

Useimmilla työssä käytettävillä koneilla päästään turvallisuustasoon e tai d (turvallisuuden eheytystasoon 3 tai 2). Tämä on turvallisuuden kannalta hyvä asia, mutta joissakin tapauksissa se aiheuttaa turhaa ylimitoitusta. (Siirilä & Tytykoski 2016, 570.)

## 6.4 Turvatoiminnot ja komponentit

Koneen yleisen määritelmän mukaisesti koneessa on aina liikkuvia osia. Tästä johtuen koneeseen liittyviä vaaroja ovat liikkuvista osista aiheutuvat tapaturmat. Ihmiseen osuessaan liikkuvat osat iskevät, leikkaavat, hankaavat tai tekevät jotain muuta haitallista. Turvakomponenteilla ja suojilla pyritään pienentämään tai poistamaan edellä mainittuja riskejä. Turvalaitteita on olemassa moniin eri käyttötarkoituksiin. Tässä opinnäytetyössä perehdytään puukuitulikanpakkaus automaatin sisältämiin turvakomponentteihin ja niiden ominaisuuksiin sekä liittämiseen turvalogiikkaan.

Turvakomponentti on nimensä mukaisesti komponentti, jolla vaikutetaan turvallisuuden ja jota ilmeikkin kone pystisi toimimaan. Komponenttia voi pitää asetusten tarkoittaman turvakomponenttina, kun se on erikseen hankittavissa eikä esimerkiksi koneen valmistajan itse omaan koneeseen rakentama. (Siirilä & Tytykoski 2016, 251-252.)

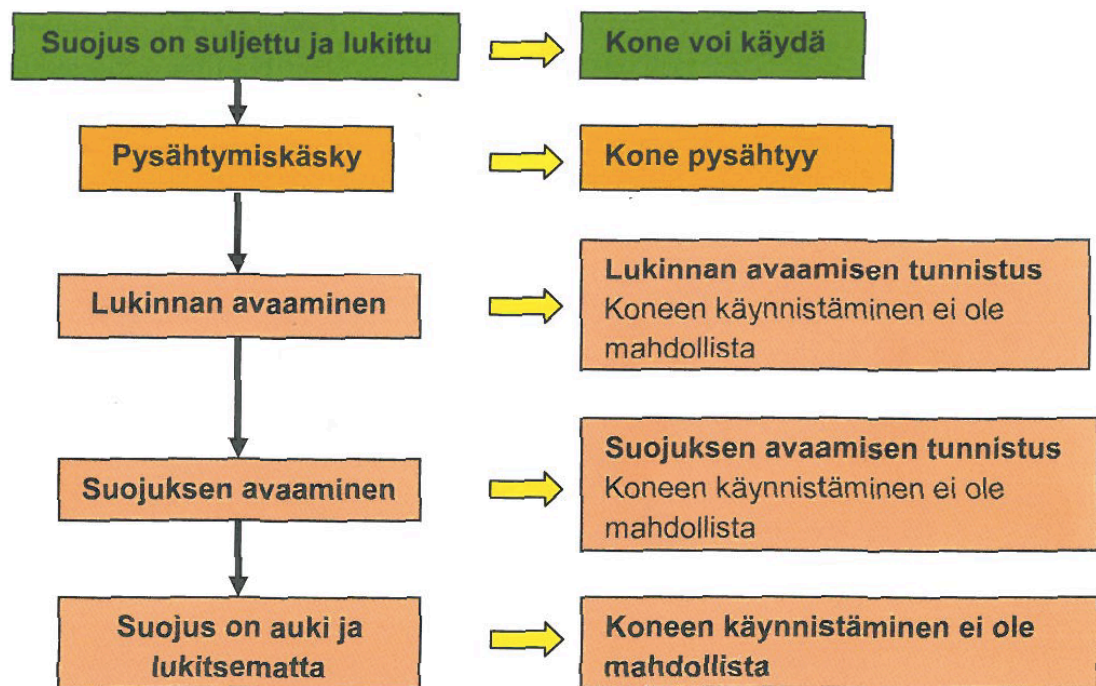
### 6.4.1 Turvasuojat

Suojien käyttö kappaletavara-automaateissa on yleistä. Useimpien automaatin sisältämien toimilaitteiden rakennetta tai toimintaperiaatetta ei suunnittelutoimilla pystytä kokonaan poistamaan. Tällöin fyysisten esteiden (suojien) lisääminen koneeseen on välttämätöntä. Suojan tarkoituksena on estää ihmisen ulottumisen liikkuvaan osaan ja liikkuvan osan osumisen ihmiseen. Yleisin suojausmenetelmä on kiinteän suojan asentaminen. Kiinteällä suojalla tarkoitetaan suojaa, jonka irrottamiseen tarvitaan työkalua. Kun suoja voidaan avata ilman työkalua, sen on oltava kytketty koneen toimintaan. Suurissa konejärjestelmissä suojaukseen käytetään turvavyöhykkeen ympäröivää suoja-aitaa. Suojissa on oltava ovia tai aukkoja, joista ihminen pystyy kulkemaan. Aitojen ovet on oltava koneen toimintaan liitettyjä ja turvalaitteilla suojattuja. (Siirilä & Tytykoski 2016, 370,382.)

Kun koneen suojissa tai muussa osassa oleva turvalaite saa aikaan pysäytyskäskyn, on koneen pysähdyttävä tilaan, jossa sen liikkeelle lähteminen tai käynnistyminen on



riittävän epätodennäköistä. Suojauksen kytkennän toteuttamiseen käytetään yleisesti turvakomponenteiksi luokiteltuja rajakytkimiä, jotka voivat olla varustettuja lukinnalla tai ilman. Lukinnalla varustettuja rajakytkimiä käytetään yleisesti suoja-aitojen ovissa. Lukinnalla varustetun suojauksen toiminnan voi kuvata kuvion 11 mukaisesti.



Kuvio 11. Suojauksen kytkennän liittäminen koneen toimintaan. (Siirilä & Tytykoski 2016, 426)

Ohjausjärjestelmä sallii lukinnan avaamisen vasta, kun koneen liike on pysähtynyt tai suojan takana oleva muu vaara on poistunut. Lukinnan avaaminen ja suojauksen avaaminen on tunnistettava erikseen. Koneen käynnistäminen sallitaan vasta, kun ohjausjärjestelmä tunnistaa suojan olevan suljettu ja lukittu. Monesti käynnistämisen yhteydessä käytetään kuittausta, jolla aktivoidaan suojien lukinta ja käynnistys sallitaan. (Siirilä & Tytykoski 2016, 426.)

### 6.4.2 Hätäpysäytys

Hätäpysäytyksen tarkoituksena on toimia varatoimena, jos normaali pysäytys ei toimi. Sillä voidaan myös estää tapaturma tai koneessa tapahtuva odottamaton tapahtuma. Hätäpysäytystä pidetään turvallisuutta täydentävänä toimenpiteenä, eikä varsinaisena turvatoimintona. (Siirilä & Tytykoski 2016, 496.)

Hätäpysäytystoiminto on suunniteltava siten, että hätäpysäytyksen aktivoituessa koneen saavuttaminen turvalliseen tilaan tapahtuu mahdollisimman nopeasti. Koneen uudelleen käynnistäminen ei ole sallittua niin pitkään, kuin hätäpysäytin on lukittu-neena SEIS-asennossa. Lukituksen palauttaminen toimintavalmiiksi ei saa käynnistää konetta. Monimutkaisemmissa koneissa vaaditaan erillinen kuittaus, ennen kuin kone on mahdollista käynnistää uudelleen. (Siirilä & Tytykoski 2016, 499,501.)

## 7 Paineilmatoimilaitteet ja lähestymiskytkimet

Liikkeen toteutuksessa puhutaan yleisesti ohjauksesta ja säädöstä, joilla voidaan vaikuttaa prosessin toimintaan. Ohjauksen peruseräite on, että ohjaus toimii ilman takaisinkytkentää. Ohjausviesti on yleensä binäärinen eli kaksitilainen, esimerkiksi käyntiin/seis ja auki/kiinni ohjausviesti. Säättöön puolestaan liittyy jatkuvatoiminen takaisinkytkentä. Säädössä prosessisuureen arvoa verrataan säätimeen asennettuun ohjearvoon ja tämän vertailun perusteella ohjataan prosessiin vaikuttavaa toimintayksikköä. Kappaletavara-automaatiossa säätöä käytetään esimerkiksi servo-ohjauksissa. Suurinta voimaa vaativat liikkeet toteutetaan yleensä hydraulisesti ja nopeimmat liikkeet pneumaattisesti. Servo- ja askelmoottoreilla saavutetaan tarkimmat säädöt. (Fonselius ym. 1996, 11.)

## 7.1 Paineilma

Paineilman käyttö perustuu pääasiassa toimilaitteiden liikkeiden toteuttamiseen. Liikkeitä aikaansaavat komponentit ovat esimerkiksi sylintereitä, tarttujia tai paineilmamooottoreita. Liikkeiden ohjaus tapahtuu erilaisten venttiilien avulla, joita ohjataan sähköisesti, pneumaattisesti tai mekaanisesti. Kappaletavara-automaatiossa liikkeiden ohjaukseen käytetään sähköistä ohjausta, jolloin toimilaitteita voidaan ohjata ohjelmoitavan logiikan avulla. Paineilma väliaineena on herkkäliikkeistä, jolloin saadaan aikaan nopeita liikkeitä. (Keinänen & Kärkkäinen 2005, 21.)

Paineilman tuotto tapahtuu paineilmakompressorin avulla. Kompressoriksi kutsutaan laitetta, jolla voidaan nostaa kaasun painetta vähintään kaksinkertaiseksi verrattuna imupaineeseen. Pienempiä paineita tuottavia laitteita kutsutaan ahtimiksi tai puhaltimiksi. Kompressorin tuottaman paineen määrä vaihtelee, muutamista litroista tuhansiin kuutiolitroihiin minuutissa. Kompressorit jaotellaan yleisesti mäntä-, ruuvi- ja lamellikompressoreiksi. Ruuvikompressorit ovat yleisimmin käytettyjä, joilla mahdollistetaan hyvät tuotto- ja paineominaisuudet. Lamellikompressorit soveltuvat pienemmille työpaineille, jonka tuotto jää ruuvikompressoreita pienemmäksi. Mäntäkompressorit ovat parhaimmillaan järjestelmissä, joissa ilmanpaine vaihtelee käytön aikana ja jos järjestelmän painevaatimus on korkea. (Keinänen & Kärkkäinen 2005, 26-27.)

## 7.2 Sylinterit

Paineilmasyylinterit ovat pneumatiikan yleisimpiä toimilaitteita. Sylintereitä voidaan käyttää kappaleiden kiinnittämiseen, siirtämiseen, leikkaamiseen, painantaan tai kääntöön. Standardimitoitettuja sylinterityyppejä on saatavilla vakiosylintereitä, johdesylintereitä, männänvarrettomia sylintereitä, lukkosylintereitä, vääntösylintereitä ja lyhytiskusylintereitä. (Keinänen & Kärkkäinen 2005, 21.) Yleisesti käytetty johdesylinteri esitetty kuviossa 12.



Kuvio 12. Johdesylinteri. (Tuotteet tehdas- ja prosessiautomaatioon 2018)

Sähköisiä tilatietoja varten sylinterin männän asento on voitava tunnistaa. Paineilmasyylinterin rungon urajohteeseen voidaan kiinnittää tunnistusanturi. Sylinterijohteeseen on yleensä kiinnitetty kestopagneettipala, joka tunnistetaan sylinterin uraan kiinnitetyn anturin avulla. Anturit ovat yleensä NO-tyyppisiä ja varustettu merkkivallolla.

Sähköpneumaattisen sylintereiden ohjaus tapahtuu paineilmalla, jota ohjataan suuntaventtiileiden avulla. Sylinterin eteen/taakse- ohjaukseen käytetään magneettikeilloilla varustettuja suuntaventtiileitä. Venttiilit voivat olla yksittäisiä venttiileitä tai terminaalirakenteisia. Terminaalin etuna on suuret läpivirtausmäärät ja moninapakaapelin käyttö. Lisäksi kaikki venttiilit voidaan liittää yhteen terminaaliin. (Keinänen, Kärkkäinen, Metso & Putkonen 2001, 74.)

### 7.3 Anturit

Koneautomaatiolaitteet tarvitsevat tietojen keräämiseen ja toimilaitteiden tilatietojen havaitsemiseen laitteita, joita kutsutaan yleisnimellä anturit. Anturi muuttaa mitattavan prosessisuureen arvon siihen verrannolliseksi virtaviestiksi. Anturin tuntoelin, jota voidaan kutsua tunnistimeksi, mittaelimeksi tai mittauselimeksi, määrittää suureen arvon, jonka jälkeen anturiosa muuttaa tuloksen halutun muotoiseksi

virtaviestiksi. Anturilla voidaan tunnistaa suureita kuten, paikkaa, lämpötilaa, painetta, voimaa, pituutta, nestepintaa tai kulmaa. Prosessiteollisuudessa käytetään yleisesti analogia-antureita, jonka suure muunnetaan lähettimessä standardiviestiksi. Usein anturi ja lähetin on rakennettu kiinteästi yhteen. (Keinänen ym. 2007, 187-188.)

Seuraavaksi käsitellään kappaletavara-automaatiossa yleisesti käytettäviä läsnäolon havaitsemiseen käytettäviä lähestymiskytkimiä ja antureita, joita voidaan käyttää pneumaattisen sylintereiden asennon tunnistamiseen, kappaleiden- ja siirtoliikkeiden havaitsemiseen. Lähestymiskytkimeksi kuvataan kytkintä, joka avaa ja sulkee virtapiirinsä kappaleen saapuessa toimintaetäisyydelle. Toimintaperiaatteen mukaisesti lähestymiskytkimet jaotellaan induktiivisiin-, kapasitiivisiin-, magneettisiin- ja optisiin kytkimiin. Kyseisten lähestymiskytkinten toimintaetäisyydet ovat 0-150 mm välillä. Lähestymiskytkimet koostuvat elektronisista komponenteista, eikä niissä ole liikkuvia osia, jonka vuoksi niiden käyttöikä on mekaanisia antureita pidempi.

Induktiiviset lähestymiskytkimet tunnistavat ainoastaan metalleja. Tunnistettaessa metallin lähestyvän anturin tuntopäätä magneettikenttä vaimenee. Tästä seuraa keulan virran pieneneminen, jolloin anturin elektroniikka muuttaa tiedon virtamuutoksen on/ei-tiedoksi. Mitä suurempi on anturin tuntopinta, sitä suurempi on kappaleen tunnistusmatka. Yleensä induktiivisen lähestymiskytkimen lukumatka on 2-5mm mutta voi olla jopa 150mm. (Fonselius ym. 1996, 33-35.)

Kapasitiivinen lähestymiskytkin reagoi melkein mihin tahansa materiaaliin. Anturi kehittää ympärilleen sähkökentän, joka heikkenee kappaleen saapuessa lukuetaisyydelle. Tuntopää ja anturin runko muodostavat yhdessä kondensaattorin, jossa ilma toimii eristeenä. Tunnistettavan kappaleen lähestyessä kondensaattorin kapasitanssi ja anturin sisältämän oskillaattorin taajuus muuttuvat. Vahvistin antaa on- tai ei-tyyppistä signaalia. Kapasitiivisen anturin lukuetaisyys on yleensä aseteltavissa. Anturityypin kytkentätaajuus vaihtelee 10-1500 Hz välillä. Kapasitiivisiä antureita käytetään induktiivisten antureiden tapaan tunnistamaan muita materiaaleja, kuin metallisia. Esimerkiksi makasiinien, levyjen, muovisten kappaleiden ja säiliössä olevien nesteiden

tunnistamiseen. Toisen aineen lävitse tunnistettaessa tunnistettavan aineen permittiivisyyden on oltava suurempi, kuin väliaineen. (Fonselius ym. 1996, 37-38.)

Magneettisiin lähestymiskytkimiin voidaan luokitella Reed- kytkimet ja Hall-anturit. Kyseiset anturit vaativat tunnistamiseen tunnistusalueelle magneetin. Reed-kytkimessä on toiminnallinen kielikytkin, jonka kosketin sulkeutuu joutuessa magneettikenttään. Kytkentäetäisyys on tyypillisesti 5-10 mm ja katkaisuetäisyys 10-15 mm. Reed- kytkimiä käytetään sylintereiden asentotunnistimina ja akseleiden pyörimisen tunnistamiseen. Hall- anturi on mikropiirikoteloon rakennettu anturi. Se generoi magneettikentän voimakkuuteen verrannollisen signaalin, joka lähestymiskytkinsovellus muokkaa on/ei-tiedoksi. Hall anturin etuna on, että se voi havaita nopeita pulsseja, jopa 100 000 pulssia/s. Hall- antureita käytetään esimerkiksi moottoreiden sytytysjärjestelmissä ja aseman ilmaisimissa. (Fonselius ym. 1996, 38-39.)

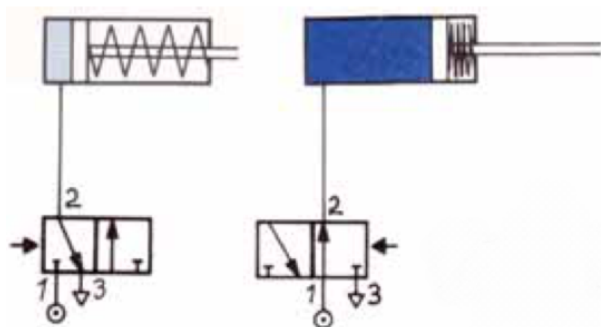
Optiset lähestymiskytkimet käyttävät lähettimenä näkyvää valoa tai infrapunavaloa lähettävää (emitoitavaa) diodia. Säte on moduloitu eli sädettä katkotaan alle 10 kHz taajuudella, jolloin vastaanotin reagoi lähetintaajuisiin valopulsseihin. Anturin herkkyttä on mahdollista säätää säätöruuvilla. Maksimi kytkentätaajuus vaihtelee 25-10000 Hz anturityypistä riippuen. Optinen lähestymiskytkin voi sisältää yhteen rakennetun lähetinvastaanottimen, joka voi olla kappaleesta heijastava tai heijastinta käytävä. Toinen vaihtoehto on, että käytetään erillisiä lähetin ja vastaanotin yksiköitä. (Fonselius ym. 1996, 39-40.)

## 7.4 Venttiilit

Pneumaattinen venttiili yleisnimitystä käytetään niille, toimilaitteille, joilla säädetään ja ohjataan pneumaattista järjestelmää. Venttiilit sijaitsevat painelähteen ja toimilaitteiden välissä. Venttiilit jaetaan suuntaventtiileihin, vastaventtiileihin, paineventtiileihin sekä virta- ja sulkuventtiileihin. Tässä opinnäytetyössä käsitellään ainoastaan sylintereiden ohjauksiin käytettäviä suuntaventtiileitä.

Suuntaventtiileiden tarkoituksena on määrittää paineilman virtaussuunta. Paineilma-sylinterin mäntää voidaan ohjata eteen ja taakse. Venttiili voi olla erillinen venttiili, tai useat venttiilit voidaan koota yhteen venttiiliterminaaliin. Automaatiosovelluk-sissa venttiiliterminaalit ovat yleisiä ja venttiileinä käytetään sähköohjattuja suunta-venttiileitä. Venttiiliterminaalin etuna on moninapakaapelin käyttö ja terminaali sallii suuret läpivirtausmäärät. Suuntaventtiilit ovat yleensä 3/2-, 5/2- tai 5/3-venttiilejä. (Keinänen & Kärkkäinen 2005, 60-61.)

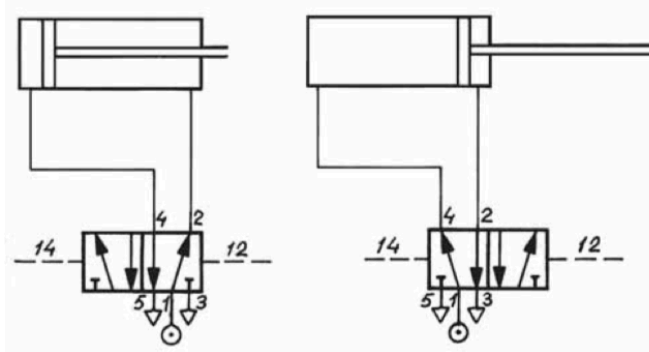
3/2-suuntaventtiileissä on kuvion 13 mukaisesti kolme liitäntäaukkoa, yksi tulo (1) -, yksi lähtö (2) - ja yksi poistoaukko (3). Normaalisti suljetun suuntaventtiilin ohjauksen ollessa pois päältä, ilma ei pääse tuloliittimestä 1 lähtöliitännään 2. Ohjauksen akti-voituessa poistoliitäntä (3) sulkeutuu ja tuloliitännästä (1) avautuu yhteys lähtöliitän-tään (2) ja sylinteri ohjataan eteen. 3/2 suuntaventtiileitä käytetään pääasiassa pai-neohjattujen yksitoimisten sylintereiden ohjaamiseen. (Hulkkonen 2008.)



Kuvio 13. 3/2-suuntaventtiilin rakenne. (Hulkkonen 2008)

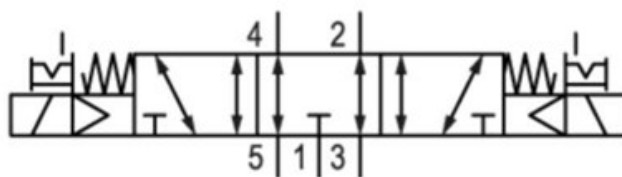
5/2-suuntaventtiileissä kuvion 14 mukaisesti on viisi liitäntäaukkoa, tuloaukko (1), kaksi lähtöaukkoa (2 ja 4) ja kaksi poistoaukkoa (3 ja 5). Sylinterin männänvarren si-sään ohjauksessa ilmavirta ohjataan tuloliittimestä (1) lähtöliittimeen (2). Samanai-kaisesti lähtöliitännästä (4) on yhteys poistoliitännään (5). Sylinterin männän ulos oh-jauksessa ilmavirta ohjataan tuloliittimestä (1) lähtöliittimeen (4). Samanaikaisesti

lähtöliitännästä (2) avautuu yhteys poistoliitännään (3). 5/2-suuntaventtiileitä käytetään kaksitoimisten sylintereiden ohjaamiseen, kahteen suuntaa pyörivän paineilma-moottorin ohjaamiseen tai kääntöpöytäsovelluksiin. (Hulkkonen 2008.)



Kuvio 14. 5/2-suuntaventtiilin rakenne. (Hulkkonen 2008)

5/3-suuntaventtiilissä on kuvion 15 mukaisesti viisi liitännäaukkoa ja kolme toiminta-asentoa. Ääriasennot ja liitännäaukot ovat samat, kuin 5/2-venttiileissä. Mikäli venttiilissä on suljettu keskiasento, ilma ei pääse sen lävitse. 5/3-venttiilissä kaksitoimisen sylinterin männän asento voidaan lukita mihin asentoon tahansa. Tuloliittimen ollessa suljettu ja muut liitännät avoimia, sylinteriä on mahdollista liikuttaa ulkoisella voimalla. (Hulkkonen 2008.)



Kuvio 15. 5/3-suuntaventtiilin rakenne. (Hulkkonen 2008)



## 8 Työn toteutus

### 8.1 Lähtöasetelma

Puukuitulusikan käsittely- ja pakkausautomaatin projektikokonaisuus aloitettiin nykytilanteen kartoittamisella, jonka pohjalta saatiin lähtötiedot automatisointiprojektin toteuttamiselle. Lähtöasetelmana toimeksiantajayrityksellä oli käytössä ruiskuvalukone, muotti ja ympäristöystävällinen puukuituraaka-aine, joilla puukuitulusikoiden valmistaminen toteutettiin. Ruiskuvalukone pudotti valaman lusikkaviuhkan laatikkoon, jonka täytyttyä operaattori vaihtoi uuden laatikon ruiskuvalukoneen täytettäväksi. Operaattorin tehtävänä oli irrottaa lusikat yksitellen ruiskuvaletusta lusikkaviuhkasta. Lusikat järjesteltiin ja pinottiin käsin, minkä jälkeen ne asetettiin pussiin. Operaattori lämpösaumasi ja tarroitti pussin, minkä jälkeen tuotepussi oli valmis.

Toimeksiantajan toiveena oli saada automatisoitua puukuitulusikan käsittely- ja pakkaustoimenpiteet. Toimeksiantaja tarjosi asiakkaalle seuraavaa ratkaisua.

Ruiskuvalukoneen palvelurobottina toimiva lineaaribotti noutaa ruiskuvaletun lusikkaviuhkan ruiskuvalumuotista ja vie viuhkan leikkaimeen, joka leikkaa tuotteista valuylijäämät. Tämän jälkeen lineaaribotti vie lusikkaviuhkan pyöreälle alustalle, jossa on lokerot jokaiselle muotista tulevalle lusikalle. Kun alusta on täynnä, kääntöpöytä kääntää uuden alustan lineaaribotin täytettäväksi. Yaskawa-teollisuusrobotti purkaa täydet lusikkalokerot kääntöalustalta pino kerrallaan ja vie lusikat pakkauskoneen hihnalle, jossa on lokero jokaiselle tuotepinolle. Pakkauskone siirtää kuljetintaan yhden lokeron verran eteenpäin jokaisen jätetyn lusikkapinon jälkeen. Lopulta lusikkapino on kulkenut pakkauskoneen hihnan päästä toiseen, minkä aikana pakkauskoneen mekaniikka on tehnyt tarroitettun pussin lusikkapinon ympärille.

Toimeksiantajayritykselle tarjottu automaattioratkaisu sisälsi seuraavat toimilaitteet:

- palvelu- ja teollisuusrobotin tarttijat.
- valuylijäämän leikkain.
- kääntöpöytä.

- Yaskawa-teollisuusrobotti.

Ruiskuvalukone ja sen palvelurobotti sekä pakkauskone toimivat omina kokonaisuuksinaan, ja näiden laitteiden käyttöönotto oli kyseisten laitteiden laitetoimittajien vastuulla. Automaattikokonaisuuden hallitsemiseksi näihin laitteisiin oli toteutettava tarvittavat I/O-liityntärajapinnat ja suunniteltava yhtenäinen turvallisuustoteutus siten, että automaattikokonaisuuden käyttö on turvallista. Käyttöpaneelina toimii Yaskawa-teollisuusrobotin operointipaneeli.

## 8.2 Omron NX1P2- ohjelmoitava logiikka

Automaatiojärjestelmänä puukuitulusikka-automaatissa käytettiin ohjelmoitavaa logiikkaa. Logiikan tehtävänä on ohjata toimilaitteita loogisessa järjestyksessä ohjelmoidun sovellusohjelman mukaisesti. Logiikka määrää liikekäskyjen suoritusjärjestyksen lineaarirobotille, teollisuusrobotille, kääntöpöydälle ja laikkaimelle. Leikkaimen ja kääntöpöydän ohjaukseen sekä ruiskuvalukoneen ja pakkauskoneen I/O-liityntärajojen toteutukseen käytettiin logiikan fyysistä I/O:ta. Logiikan ja turvalogiikan välinen tiedonsiirto toteutettiin EtherCAT-kenttäväylällä. Logiikan ja Yaskawa-teollisuusrobotin välinen tiedonsiirto toteutettiin Ethernet/IP-kenttäväyläyhteyden avulla.

### 8.2.1 Ohjelmoitavan logiikan valinta

Tässä opinnäytetyössä käytettiin Omronin valmistamaa NX1P2-9024DT1-mallin ohjelmoitavaa logiikkaa. Logiikan valintaan hyödynnettiin sähkösuunnittelun benchmarking-haastattelun tutkimustuloksia. Toimeksiantaja on keskittänyt automaattien ohjauksen Omronin valmistamien logiikkajärjestelmien ympärille. Laitevalmistajana toimivan yrityksen uusien laitteiden ohjelmointiin käytettävä aika lyhenee sekä tekee automaatiolaitteista ohjelmarakenteeltaan yhdenmukaisia, kun ohjelmointiympäristö ja tietyt ohjelmarakeenteet pysyvät samana. Omronin valmistamat logiikkajärjestelmät ovat hyväksi havaittuja toimeksiantajan käyttötarpeisiin. Tästä syystä logiikkavalinnan rajaus oli helppo tehdä Omronin valmistamiin logiikkajärjestelmiin.

Mekaniikkasuunnittelijan haastattelun tuloksena selvisi, että automaatin sisältämä fyysisen I/O:n tarve ei ollut kovin suuri, koska mekaanisia toimilaitteita oli ainoastaan valuylijäämän leikkain ja kääntöpöytä. Automaatin tilan ilmoitukseen käytettiin valotornia ja ruiskuvalukoneen sekä pussituskoneen kättelyihin tarvittiin fyysistä I/O:ta. Kaikkiaan fyysisen I/O:n tarve oli 11 tulosignaalia ja 10 lähtösignaalia. Mekaaniset toimilaitteet sijaitsevat lähellä toisiaan, jolloin I/O toteutus oli mahdollista toteuttaa ilman kenttäväylää. Vähäinen I/O: tarve ja lyhyet toimilaitteiden etäisyydet mahdollistivat fyysisellä I/O:lla varustetun ohjelmoitavan logiikan käytön.

Yaskawa-teollisuusrobotin ja logiikan välille oli luotava yhteys, joka mahdollistaa ohjaukskäskyjen ja robotin tilatietojen lähettämisen laitteiden välille. Yaskawa-robottiin on mahdollisuus liittyä Ethernet/IP- kenttäväylän avulla, jolloin logiikkavalinnassa oli huomioitava Ethernet/IP- kenttäväylämahdollisuus.

Omronin NX1P2-9024DT1-mallin ohjelmoitava logiikka valikoitui käyttötarpeisiin. Tämä logiikka sisältää yhteensä 24 digitaali-I/O:ta, joista 10 lähtösignaalia ja 14 tulosignaalia, jotka riittivät toimeksiantajan tarpeisiin. Logiikka sisältää Yaskawa- teollisuusrobotille vaaditun Ethernet/IP- kenttäväylän lisäksi EtherCAT- kenttäväylän, johon oli mahdollista liittää Omronin turvaohjain (Safety CPU) ja turva I/O-yksiköitä. Turvaohjausten toteuttamiseen tämä sopi hyvin, koska erillistä turvalogiikkaa ei tarvittu. Turvalogiikkana toimiva Omronin turvaohjain pystyttiin liittämään osaksi ohjelmoitavaa logiikkaa, jolloin logiikan ja turvaohjaimen ohjelmointi pystyttiin toteuttamaan samalla Omronin Sysmac Studio- ohjelmistolla.

### 8.2.2 NX1P2- ohjelmoitavan logiikan rakenne

NX1P2-sarja kuuluu Omronin NX/NJ-koneohjainten tuoteperheeseen. NX1P2- mahdollistaa tehokkaan ja synkronoidun ohjauksen kaikille konelaitteille, kuten I/O, turvallisuus, liike ja konenäkö integroidussa ympäristössä. (NX1P Series machine controller n.d.) NX1P2-9024DT1- ohjelmoitava logiikka on esitetty kuviossa 16.

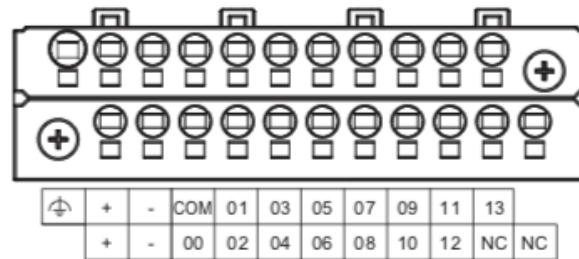


Kuvio 16. Omron NX1P2-9024DT1-ohjelmoitava logiikka (NX1P Series machine controller n.d.)

NX1P2-9024DT1- ohjelmoitava logiikka sisältää seuraavat ominaisuudet:

- nopein jaksonaika 2ms.
- max 4 synkronoidun akselin liittäminen.
- sekvenssi ohjaukset ja liikkeen hallinta.
- max 8 paikallisen I/O-yksikön liittäminen.
- Ethernet/IP ja EtherCAT- portit.
- max 16 EtherCAT orjalaitetta.
- 2 lisäkorttia sarjaliikenteelle tai analogiselle I/O toiminnolle.

NX1P2-9024DT1-logiikan CPU-yksikön muistikapasiteetti on 1,5 MB. Muuttujien muistikapasiteetista 32 KB säilyy virtakatkon yli ja 2MB hävitetään virtakatkon aikana. (NX1P Series machine controller n.d.)



Symbol	Name	Description
	Functional ground terminal	Connect the ground wire to the terminal
+/-	Unit power supply terminals	These terminals are connected to the unit power supply The + and - terminals are internally connected to each other
COM	Common terminal	Common terminal for the input circuits
00 to 13	Input terminals	General-purpose input A
NC	NC	Do not connect anything

Kuvio 17. Omron NX1P2-9024DT1-tulotermiinaali. (NX1P Series machine controller n.d.)

Kuviossa 17 esitetty 9024DT1-mallin logiikan tulotermiinaali ja sen kytkennät. Logiikan jännitteensyöttö ja termiinaalin saattaminen toimintaan vaatii +24VDC jännitteensyötön + liittimeen ja -24VDC syötön COM ja – liittimeen. 24VDC PNP-tulosignaaleja on käytössä 14, jotka kytketään paikkoihin (00-13). Tulosignaaleihin kytketyt tulot näkyvät logiikassa tulo (input)-tietoina. (NX1P Series machine controller n.d.)

NC	C0 (+V)	00	02	04	06	08	NC	NC	NC	NC
	0V0	01	03	05	07	09	NC	NC	NC	NC

Symbol	Name	Description
C0 (+V)	Common terminal	Connected to the 24 V side of the I/O power supply
0V0	0 V terminal	Supplies 0 V for the internal circuits for driving
00 to 09	Output terminals	PNP (sourcing) type output with the load short-circuit protection function
NC	NC	Do not connect anything

Kuvio 18. Omron NX1P2-9024DT1-lähtötermiinaali. (NX1P Series machine controller n.d.)

Kuvion 18 mukaisesti lähtötermiinaalissa on käytössä 10 digitaalilähtöä (output), jotka kytketään liittimiin (00-09). Lähtötermiinaalin toimintaan saattaminen vaatii +24VDC

jännitteensyötön +V liittimeen ja -24VDC syötön 0V0 liittimeen. Lähtömäärittelyissä on määritelty lähdön maksimi kytkentävirraksi 300 mA. (NX1P Series machine controller n.d.)

### 8.2.3 Sysmac Studio-ohjelmisto

Omronin Sysmac Studio- ohjelmisto mahdollistaa NJ/NX-sarjan ohjainten ohjelmoinnin ja asetusten määrittämisen ohjelmointi PC:n avulla ja on IEC 61131-3-standardin mukainen. Sysmac Studio mahdollistaa konfiguroinnin, integroinnin, simuloinnin ja valvonnan samassa käyttöliittymässä. Sysmac Studio integroi logiikkaohjauksen, turvallistamisen, ohjaimet ja robotiikan samaan ohjelmaympäristöön, joka mahdollistaa pienemmät ohjelmisto- ja komponenttikustannukset. (Automation software: Sysmac Studio for machine creators 2019.)

Sysmac Studio- ohjelmisto perustuu relekaavio- (Ladder Diagram) ja strukturoitu teksti- (Structured Text) sekä logiikkakaavio (Function Block Diagram, FBD) ohjelmointikieliin. Ohjelmisto määrittää CPU:n muistialueet automaattisesti luoduille muuttujille, joka nopeuttaa ohjelmointia ja vähentää virheenmahdollisuuksia. (Automation software: Sysmac Studio for machine creators 2019.)

## 8.3 Omron NX- turvaohjain

Automaattikokonaisuuden turvallistaminen tapahtui turvalogiikkana toimivan turvaohjaimen avulla. Turvaohjain valvoi turvakomponenttien tilatietoja. Näiden perusteella turvalogiikka ohjasi hätä-seis ja turvareleitä, jotka ohjaavat edelleen automaattin turvatoimintoja, kuten paineilma- ja tasajänniteohjauksia sekä salli liikekomentojen suorittamisen.

### 8.3.1 Turvaohjaimen valinta

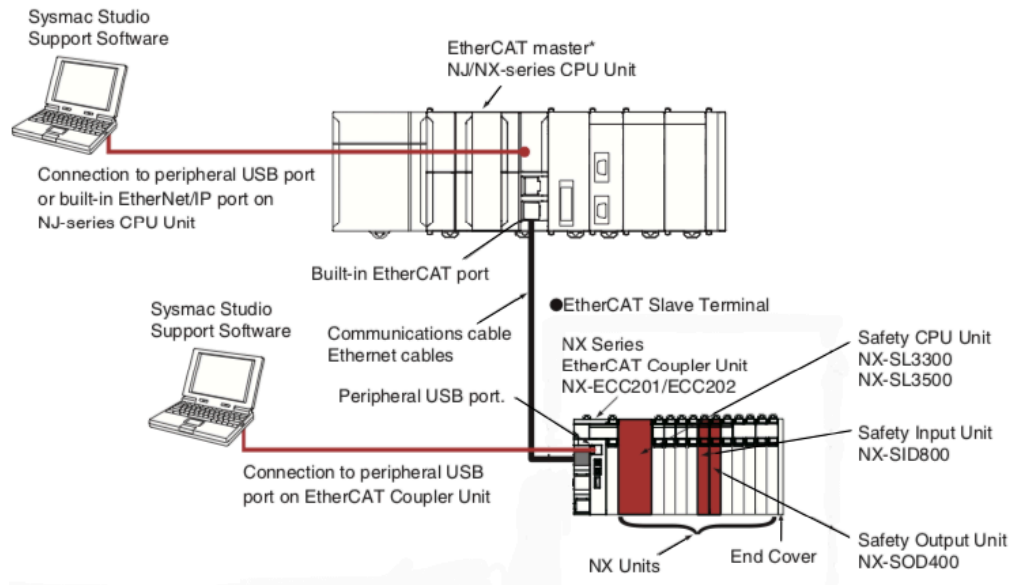
Tässä opinnäytetyössä käytettiin Omronin NX1P2- logiikkaan liitettävää Omronin NX-turvaohjainta. Automaattikokonaisuuden turvallistamisen vaihtoehtoisena menetelmänä olisi voinut käyttää turvalogiikan sijasta turvareleitä. Turvarele toimii hyvänä

vaihtoehtona yksinkertaisten turvatoimintojen toteuttamiseen, joissa yleensä hätäteispiirille ja turvapiirille on kummallekin omat turvareleensä. Turvareleen huonona puolena on muokattavuus, mikäli jälkikäteen on tarve lisätä turvakomponentteja tai muokata olemassa olevia turvatoimintoja. Tämä vaatisi turvapiirien uudelleen suunnittelun ja turvareleiden fyysisten kytkentöjen muuttamisen.

Tässä opinnäytetyössä turvallisuustoteutus vaati liittynän ruiskuvalukoneen, teollisuusrobotin ja pakkauskoneen turvakomponentteihin. Tästä johtuen turvatoimintojen toteutus oli helpompi toteuttaa turvalogiikan avulla, jolloin turvatoiminnot voitiin helposti muokata- ja määrittää tarvittavia ehtoja turvalogiikkaohjelman avulla. Erilistä turvalogiikkaa ei tarvittu, koska Omronin NX1P2- ohjelmoitava logiikka mahdollisti turvallisuuden toteuttamisen NX- turvaohjeimen avulla.

### 8.3.2 Turvayksiköt

NX1P2- ohjelmoitavan logiikan liittäminen NX- turvaohjaimeen tapahtui EtherCAT-kenttäväylän avulla. Kuvion 19 mukaisesti turvayksikön liittäminen logiikkaan vaatii käytettäväksi EtherCAT Slave- kommunikaatio yksikön (NX-ECC203), johon liitettiin turvaohjain (Safety CPU NX-SL3300) ja tarvittavat turva I/O-yksiköt (NX-SID800/NX-SOD400).



Kuvio 19. Turvaohjaimen master ja slave järjestelmäkoonpano. (NX-SL/SI/SO: NX-series Safety Control Units 2019)

Ohjelmoitava logiikka toimii EtherCAT isäntälaitteena (master) ja EtherCAT kommunikaatioyksikkö NX-ECC203 toimii orjalaitteena (slave), joka toimii linkkinä logiikan ja NX-sarjan I/O-yksiköiden välillä. (NX-SL/SI/SO: NX-series Safety Control Units 2019.) Orjalaitteita olisi mahdollista hajauttaa kentälle ketjuttamalla kommunikaatioyksiköitä toisiinsa. Tässä opinnäytetyössä ei ollut tarvetta hajautukselle, koska kaikki turva I/O oli mahdollista kytkeä yhteen ohjauskeskukseen.

NX-ECC203- yksikköön liitettiin NX-SL3300-turvaohjain, joka mahdollistaa 32 turva I/O-yksikön liittämisen. Turvaohjain ja turva I/O-yksiköt mahdollistavat standardin IEC 61508 SIL 3- turvallisuustason. Turvaohjaimeen liitettiin turva I/O:n virransyöttöön vaadittu NX-PF0630-virransyöttökortti, joka mahdollistaa maksimissaan 4A virransyötön turva I/O-korteille. Virransyöttökorttiin liitettiin turva I/O-yksiköiksi NX-SID800 turvatuloyksikkö, johon mahdollisuus liittää 8-turvatuloa. Turvatuloyksikköön liitettiin NX-SOD400 turvalähtö-yksikkö, joka mahdollistaa 4-turvalähdön ohjauksen. (NX-SL/SI/SO: NX-series Safety Control Units 2019.)



Turvatulaja suunniteltaessa oli huomioitava, että SIL 3- turvaluokituksen täyttyminen vaatii turvakomponenteilta kahdennetut kärkitiedot. Tämä tarkoittaa, että yksi turvalaite vie turvatulokortilta kaksi turvatuloa.

## 8.4 Yaskawa GP12 -teollisuusrobotti

Yaskawa-teollisuusrobotilla suoritettiin lusikkapinojen käsittely. Lusikkapinot haettiin kääntöalustalta ja vietiin pakkauskoneen hihnalle. Yaskawa-robotin tarttujan ohjaus toteutettiin robotin oman I/O:n avulla. Logiikan ja Yaskawa-robotin väliseen tiedonsiirtoon käytettiin Ethernet/IP-kenttäväylää. Tiedonsiirtoyhteyden välityksellä logiikka sallii robotille haku- ja vientiluvan. Robotti puolestaan välittää logiikalle tilatietoja suoritetuista ohjelmavaiheista.

### 8.4.1 Robotin valinta

Tässä opinnäytetyössä käytettiin 6- akselista Yaskawa GP12- teollisuusrobottia. Kääntöalustan tyhjentämiseksi robotin oli kyettävä poimimaan kääntöalustalle erilokeroihin asetetut lusikat. Kukin lusikkapino oli poimittava erikohdasta kääntäen tarttujan kulmaa edelliseen pinoon nähden. Pystyasennossa poimitut lusikkapinot oli aseteltava vaakatasossa pakkauskoneelle, joka vaati ennen pinon jättämistä tarkkoja lineaariliikkeitä. Hakuaseman monimuotoisuuden ja lusikkapinon asettelun liikeratojen vaativuuden vuoksi, 6-akselisen robotin käyttö tässä opinnäytetyössä oli välttämätöntä.

Robotin valintaan hyödynnettiin mekaniikkasuunnittelun haastattelun tuloksia. Mekaniikkasuunnittelun 3D-mallissa pystyi mallintamaan robotille vaaditut liikematkat, jonka perusteella robotin ulottuvuutta pystyi arvioimaan. Robotilla käsiteltävät lusikkapinot olivat massaltaan kevyitä, jonka vuoksi robotin kantokyky ei asettanut robotivalintaan rajoitteita. Loppuasiakkaan toiveena oli, että kappaleen käsittelyyn käytettäisiin Yaskawa teollisuusrobottia. Asiakkaan toive otettiin huomioon ja robotti valittiin Yaskawan robottivalikoimasta. Robotti ilmoitettuja liikenopeuksia verrattiin ruiskuvalukoneen jaksonaikoihin, jonka mukaisesti pystyttiin määrittämään robotille vaadittu liikenopeus. Robotin ohjainyksikkönä käytettävä YRC1000- kontrolleri sisälsi

tiedonsiirtoon tarkoitettun Ethernet/IP- kenttäväyläportin sekä fyysistä digitali I/O:ta 24 tuloa ja 24 lähtöä, joita käytettiin robotin tarttujan ohjauksiin ja rajatietoihin.

#### 8.4.2 Yaskawa GP-12 -teollisuusrobotti

Kuvion 20 mukainen Yaskawa GP-12- teollisuusrobotti on nopea ja tarkka, jonka kantokyky (payload) on 12 kg. Ulottuvuus 1440 mm ja toistotarkkuus  $\pm 0.08$  mm. Robotti on tarkoitettu kokoonpano ja käsittelytehtäviin sekä CNC-työstö ja pakkaustehtäviin. Robotin rungossa on sisäinen reititys tarttujan syöttöjohdoille, joka vähentää ulkoisten johtimien tarvetta tarttujajohdotuksia ajatellen. GP12-robotin ohjainyksikkönä käytetään YRC1000 robottiohjainta. (Handling & General Applications with the GP-series: Motoman GP12 2019.)



Kuvio 20. Yaskawa GP12 -teollisuusrobotti. (Handling & General Applications... 2019)

### 8.4.3 YRC1000- robottiohjain

YRC1000 on kuvion 21 mukainen joustava ja kompakti ohjain Yaskawa-roboteille, se sisältää kevyen kosketusnäytöllisen käsiohjaimen. YRC1000 on rakennettu maailmanlaajuiseen standardiin ja sen käyttöjännite on kytkettävissä suoraan 380 - 480VAC-jännitteille ilman muuntajaa. Robottiohjaimen ja robotin välille vaaditaan ainoastaan yksi kaapeli toimivan yhteyden saavuttamiseksi.

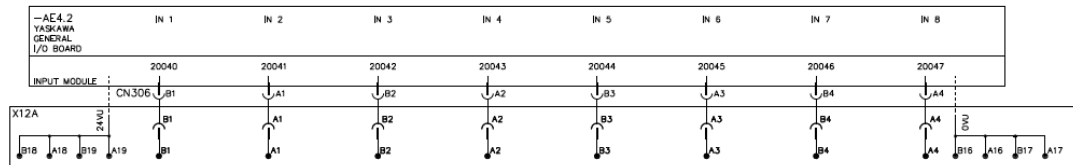


Kuvio 21. YRC1000-robottiohjain. (Single Functions and Packages... 2019)

YRC1000- robottiohjain tukee vakiona RS232- ja Ethernet-liitäntää. Näiden lisäksi on saatavilla monia laajennuskortteja kenttäväyläyhteyden toteuttamiselle. Yleisesti väylärajapinnoissa käytetään laitteiden välillä Boolean-datatyyppeä (pälle/pois). RS232 ja Ethernet mahdollistaa korkeamman tason dataviestinnän, kuten tilaviestien ja paikkatietojen lähettämisen laitteiden välillä. (Single Functions and Packages... 2019.)

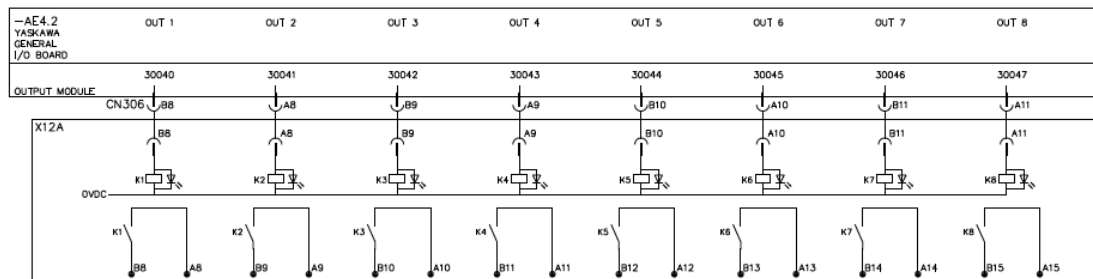
Robottiohjaimessa on käytössä vakiona 24 digitaalista tulo- ja lähtösignaalia. Signaalit on jaoteltu kahdeksan tulon/lähdön ryhmiksi. Kuviossa 22 esitettynä ensimmäinen tuloryhmät IN 1-8, jonka tuloliittiminä (B1-A4). Jokaisessa kahdeksan tulon ryhmässä

on neljä +24V ja 0V0 virransyöttöliitintä. Näillä mahdollistetaan anturisyötteiden kytken tulokortille.



Kuvio 22. YRC1000-robottiohjaimen IN 1-8 tuloryhmä. (Robot Controller: User Manual 2017)

Kuvion 23 mukaisesti ensimmäisessä lähtökortissa OUT 1-8 on relelähde, joissa potentiaalivapaat koskettimet liittimissä (B8-A15). Potentiaalivapaat koskettimet (K1-K8) mahdollistavat ohjaus-signaalin jännitesyötön robotin omaa- tai ulkopuolista jännitettä hyödyntämällä.



Kuvio 23. YRC1000-robottiohjaimen OUT 1-8 lähtöryhmä. (Robot Controller: User Manual 2017)

#### 8.4.4 Robotin työkalu

Yaskawa GP12- robotin työkaluna käytetään mekaanista pihtitarttujaa, jolla lusikkapinon poimiminen tapahtuu. Tarttujan paineilmasylinteri toimii pneumaattisena toimilaitteena, jota ohjattiin 5/2- suuntaventtiilin avulla. Venttiiliä ohjattiin YRC1000- oh-

ohjaimen lähtösignaalilla OUT 1. Pihtitarttujan auki- ja kiinniasento tunnistettiin paineilmasyylinterin urajohteeseen kiinnitettävillä kahdella magneettikytkimillä, jotka kytkettiin YRC1000- tuloon IN 1 ja IN 2.

Pihtitarttujan lisäksi robotintyökalu sisältää paininsyylinterin, jonka tarkoituksena on tukea lusikkapinoa poiminnan aikana. Painantasyylinteri toimii niin ikään pneumaattisena toimilaitteena, jota ohjattiin 5/2-suuntaventtiilin avulla. Venttiilin sähköinen ohjaus kytkettiin YRC1000-ohjaimen lähtösignaaliin OUT 2 sekä magneettikytkinten tilatiedot kytkettiin tuloihin IN 3 ja IN 4. Kyseiset tulo- ja lähtösignaalit nimettiin robotiohjelmointivaiheessa.

## 9 Laitekonfiguraatiot

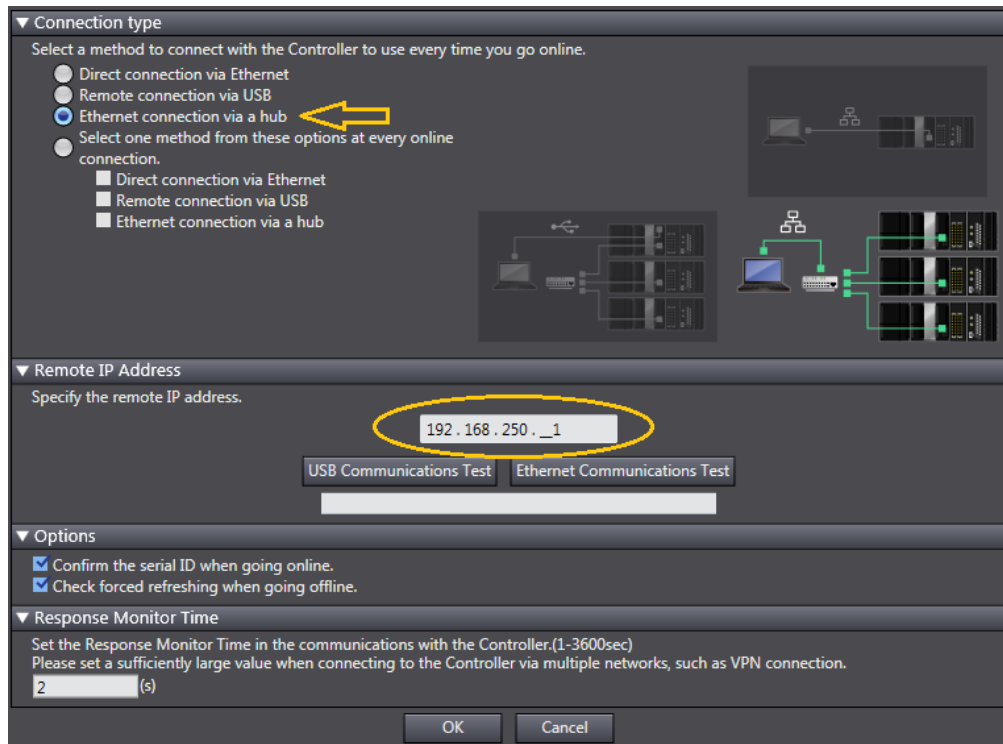
### 9.1 Projektin luominen ja yhteysasetusten määrittäminen

Uuden projektin luonti ja yhteysasetusten määrittäminen toteutettiin Omronin Sysmac Studio- käyttöohjeen mukaisesti. Uuden projektin määrittelyihin kirjattiin uuden projektin nimi, tekijä ja mahdolliset lisätiedot sekä projektissa käytettävä ohjain.

Tässä opinnäytetyössä logiikkaohjaimena toimi NX1P2, jonka malliksi valittiin 9024DT1. Ohjaimen versioksi valittiin 1.18, joka ilmoitettu logiikan tyyppikilvessä.

Projektin luomisen jälkeen avautui Sysmac Studion sovellusikkuna, johon logiikkaohjelmointi aikanaan suoritettiin.

Projektin luomisen jälkeen logiikalle oli määritettävä yhteydenmuodostamistapa ja IP-osoite. Kuvion 24 mukaisesti yhteyden muodostaminen voitiin toteuttaa logiikan USB- tai Ethernet/IP- porttiin. Vaihtoehtoisesti yhteys voitiin muodostaa Ethernet-verkossa käytetyn kytkimen välityksellä, joka mahdollisti tiedonsiirron useiden Ethernet-laitteiden välillä. Tässä opinnäytetyössä käytettiin Ethernet-kytkintä, joka mahdollisti logiikkaohjelmoinnin ja simuloinnin katkaisematta logiikan ja teollisuusrobotin välistä Ethernet/IP- yhteyttä. Tästä johtuen logiikan yhteydenmuodostamistavaksi valittiin "Ethernet connection via a hub".



Kuvio 24. Yhteysasetusten määrittäminen.

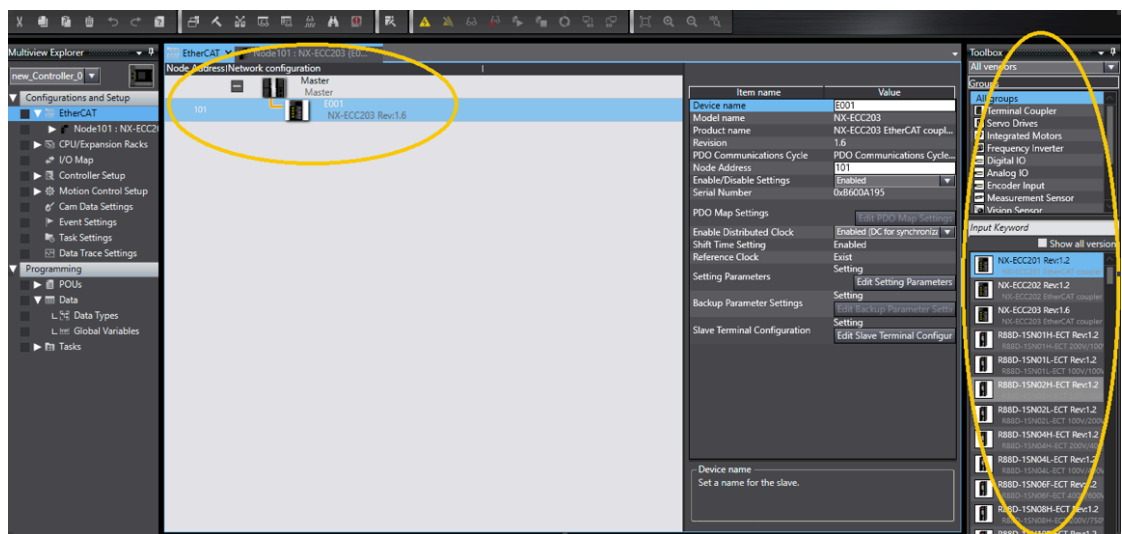
Logiikalle annettiin IP-osoitteeksi 192.168.250.1. Ennen yhteyden testaamista, oli varmistettava ohjelmointi PC:n Ethernet-adapterin IP- asetukset, jotka tulivat olla samalla 192.168.250- alueella, jotta yhdistäminen logiikkaan oli mahdollista. Fyysinen yhteys logiikan ja ohjelmointi PC:n välille luotiin RJ-45 Ethernet-kaapelilla. Yhteyden muodostamisen jälkeen Sysmac Studio- ohjelmisto oli mahdollista kytkeä online-tilaan, joka tarkoitti fyysistä yhteyttä ohjelmointi PC:n ja logiikan välillä.

## 9.2 EtherCAT-kenttäväylän konfiguraatio

EtherCAT-kenttäväyläkonfiguraatioissa määritettiin master-laitteena toimivan logiikan ja slave-laitteena toimivan NX-ECC203 EtherCAT-kommunikaatioyksikön väliset asetukset, jolloin laitteet pystyivät kommunikoimaan keskenään. Tämän jälkeen oli määritettävä EtherCAT-kenttäväylässä olevat NX-SL3300-turvaohjain sekä NX turva I/O- yksiköt.

Sysmac Studioon käyttöohjeen mukaisesti EtherCAT-kenttäväyläkonfiguraation voi toteuttaa automaattisesti siten, että logiikka etsii itse EtherCAT-kenttäväylään kytketyt laitteet. Vaihtoehtoisesti yksiköt oli mahdollista hakea Sysmac Studioon laitekirjastosta. Tässä opinnäytetyössä slave-laitteita ja turva I/O- yksiköitä oli vähäinen määrä, jonka vuoksi laitteet haettiin laitekirjastosta.

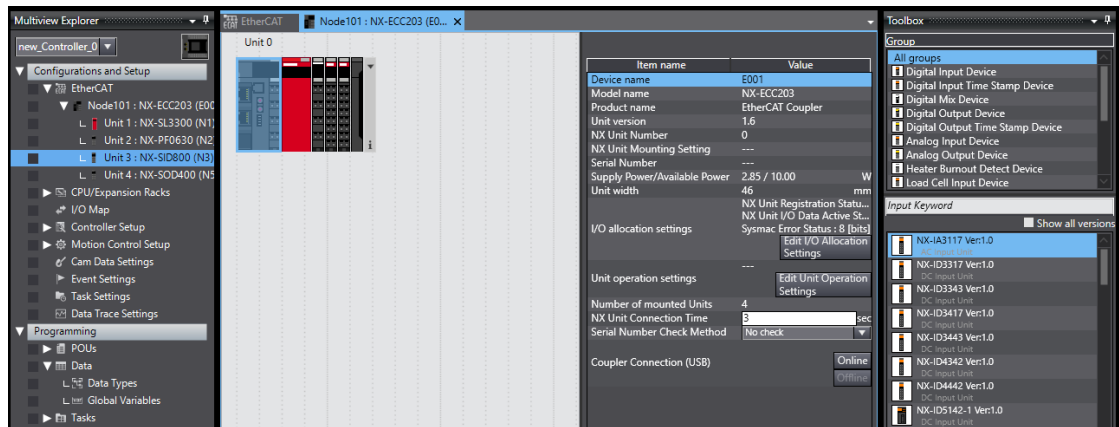
Kuvion 25 mukaisesti EtherCAT-konfiguraatioasetuksissa logiikka oli määritellyt automaattisesti NX1P2- logiikan EtherCAT-master laitteeksi. Kuvion 25 oikealla reunalla olevasta kirjastosta valittiin slave-laitteeksi NX-ECC203, joka yhdistettiin master-laitteeseen. NX-ECC203- orjayksikköön oli mahdollista asettaa osoite DIP-kytkimistä, jolloin ohjelmaan oli määritettävä sama orjalaitteen osoite. Oletusasetuksena orjalaitteeseen oli määritetty osoitteeksi 0, joka salli ohjelmassa vapaavalintaisen osoitteen. Slave-laitteen osoitteeksi määritettiin osoite 101.



Kuvio 25. EtherCAT-slave-laitteen lisäys.

NX-ECC203- kommunikaatioyksikön lisäämisen jälkeen oli määritettävä turvaohjain ja turva I/O- yksiköt. Kuvion 26 mukaisesti NX-ECC203 kommunikaatioyksikköön lisättiin Sysmac Studioon kirjastosta NX-SL3300- turvaohjain, NX-PF0630- virransyöttökortti, NX-SID800- turvatuloyksikkö ja NX-SOD400- turvalähtöyksikkö. Jokainen orjalaite sai oman yksikkö- (Unit) numeron, jonka Sysmac Studio määrittä automaattisesti lisätyille

turvallaitteille yksikkönumerosta 0- alkaen. Turvayksiköitä lisättäessä oli tarkistettava jokaisen yksikön versionumero, joka ilmoitettiin turvamoduulin tuotekilvessä.



Kuvio 26. Turva I/O- yksiköiden lisäys.

Osoitteiden määrittämisen jälkeen väylä oli käytettävä virrattomana ja ohjelma oli synkronisoitava, jolloin väylälaitteiden konfigurointi viimeisteltiin ja yhteys laitteiden välillä otettiin käyttöön.

### 9.3 Ethernet/IP-kenttäväylän konfiguraatio

Ethernet/IP- kenttäväyläyhteys muodostettiin logiikan ja Yaskawa- teollisuusrobotin väliseen tiedonsiirtoon. Ennen Sysmac Studio- ohjelmistoon tehtävää Ethernet/IP- konfiguraatiota oli teollisuusrobottiin määritettävä yhteysasetukset. Asetusten määrittämisen jälkeen robotilta ladattiin EDS- tiedosto, joka vaadittiin Sysmac Studion Ethernet/IP konfigurointiin. Robotin Ethernet/IP- asetukset ja EDS- tiedoston luonti toteutettiin Yaskawa YRC1000- robottiohjaimen käyttöohjeen mukaisesti.

YRC1000- robottiohjain käynnistettiin käyttöohjeen mukaisesti huolto- tilaan (Maintenance mode), jossa yhteysasetuksien määrittäminen ja EDS-tiedoston luonti suoritettiin. Kuvion 27 mukaisesti huoltotilassa määritettiin Ethernet adapterin- asetukset, jossa määritettiin I/O:n koko ja instanssit. Input- ja output kooksi valittiin 8 byteä, eli

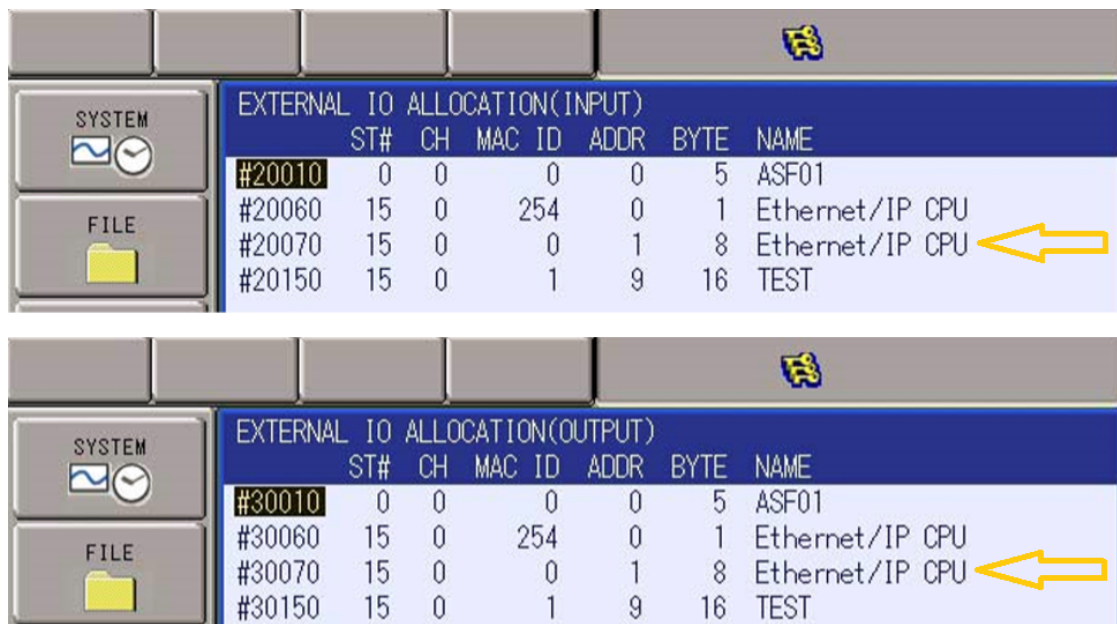


64 bittiä input ja output- tietoa laitteiden välille. Input-, output- ja konfiguraatio instanssiarvoksi määritettiin 101, 102 ja 103 käyttöohjeen mukaisesti. Ethernet- scannerin asetuksia ei määritetty, koska yhteys määritettiin logiikan ja robotin välille.



Kuvio 27. Ethernet/IP- adapterin asetukset.

Adapteriasetusten määrittämisen jälkeen robotti varasi kuvion 28 mukaisesti ulkoista input- tietoa osoitteesta #20070 alkaen 8 byteä eteenpäin. Ulkoista output- tietoa robotti varasi osoitteesta #30070 alkaen 8 byteä. Robotissa kyseiset osoitealueet olivat käytettävissä robotin ulkoisina tuloina IN#0049 alkaen ja ulkoisin lähtöinä OUT#0049 alkaen. Tavoitteena oli, että logiikalta tuleva ensimmäinen byten ensimmäinen bitti näkyi robotilla input- tietona tulossa IN#0049 ja ensimmäisen byten toinen bitti robotin tulossa IN#0050 ja jne. Robotin ulkoisen lähdön OUT#0049 ohjaus, tuli näkyä logiikalla ensimmäisen byten ensimmäinen bitin aktivoitumisena ja OUT#0050 ohjaus ensimmäisen byten toisen bitin aktivoitumisena ja jne. Ethernet/IP- datan jakaminen byteistä bittitasolle tapahtui logiikanpuolen Ethernet/IP- määrittelyissä.



EXTERNAL IO ALLOCATION(INPUT)						
	ST#	CH	MAC ID	ADDR	BYTE	NAME
#20010	0	0	0	0	5	ASF01
#20060	15	0	254	0	1	Ethernet/IP CPU
#20070	15	0	0	1	8	Ethernet/IP CPU
#20150	15	0	1	9	16	TEST

EXTERNAL IO ALLOCATION(OUTPUT)						
	ST#	CH	MAC ID	ADDR	BYTE	NAME
#30010	0	0	0	0	5	ASF01
#30060	15	0	254	0	1	Ethernet/IP CPU
#30070	15	0	0	1	8	Ethernet/IP CPU
#30150	15	0	1	9	16	TEST

Kuvio 28. I/O:n osoitealueet.

Ethernet/IP- adapteriasetusten määrittämisen jälkeen määritettiin Ethernet- portin asetukset. Robotti sisälsi kolme Ethernet- porttia, joista ensimmäinen LAN1 oli robotin omassa käytössä. IP-osoite ja aliverkon peite määritettiin ensimmäiselle vapaalle portille LAN2. IP- osoitteeksi valittiin 192.168.250.20 ja aliverkon peitteeksi 255.255.255.0.

Yhteysasetusten määrittämisen jälkeen robotilta oli ladattava EDS- tiedosto, joka vaadittiin Sysmac Studion Ethernet/IP- yhteysasetusmäärittelyissä. EDS- tiedosto ladataan USB- muistitikulle robotin käsiohjaimena toimivan operointipaneelin USB- portista. Lataus suoritettiin yhteysasetusmäärittelyjen tavoin huolto- tilassa. Ennen latausta määriteltiin ladattava tiedosto ja käytettävä USB- portti.

Seuraavaksi määritettiin logiikalle Ethernet/IP- konfiguraatiot Sysmac Studio ohjelmiston käyttöohjeen mukaisesti. Konfigurointi aloitettiin Ethernet/IP- datatyypin määrittämisellä, jossa luotiin datatyyppi tulevalle ja lähtevälle Ethernet/IP datalle. Datatyyppien tarkoituksen oli jaotella Ethernet/IP- väylänyli tuleva data logiikalle siten, että byte- ja bittinumerot täsmäsivät laitteiden välillä. Kuvion 29 mukaisesti input- ja output datatyyppiksi valittiin STRUCT (Structured), joka voi sisältää useita yksittäisiä

datatyyppejä. STRUCT- tyyppiä valittiin CJ, joka mahdollisti Sysmac Studio- ohjelmistossa datan jakamisen bittitasolle. STRUCT- tietotyyppiin lisättiin jäseniä (members), jotka määriteltiin Boolean- datatyypeiksi. Boolean- datatyyppi voi saada kaksi mahdollista arvoa (tosi ja epätosi). Jäseneksi lisätyt BOOL- datatyyppit toimivat yksittäisinä bitteinä, jotka Sysmac Studio jakoi automaattisesti bytestä 0 alkaen. Ensimmäinen nimetty BOOL- datatyyppi lisättiin 0- byten bitiksi 0. Seuraava lisätty jäsen lisättiin 0- byten bitiksi 1 ja jne. Tiedonsiirtoon määritetty 8 byten koko mahdollistaa 64 Boolean- datatyyppien käytön tulo- sekä lähtöpuolella.

Structures	Name	Base Type	Offset Type	Offset Byte	Offset Bit	Comment
Union	NumberData	STRUCT	NJ			
Enumerated	PLC1_Data	STRUCT	NJ			
	YaskawaRobotEIP_InputData	STRUCT	CJ			
	DI1_RobotPotEmpty	BOOL		0	0	HandshakeFromRobot
	DI2_PackingReadyForStarting	BOOL		0	1	HandshakeFromRobot
	DI3_RobotInThePott	BOOL		0	2	HandshakeFromRobot
	DI4_RobotRestart	BOOL		0	3	
	DI6	BOOL		0	4	

Structures	Name	Base Type	Offset Type	Offset Byte	Offset Bit	Comment
Union	NumberData	STRUCT	NJ			
Enumerated	PLC1_Data	STRUCT	NJ			
	YaskawaRobotEIP_InputData	STRUCT	CJ			
	YaskawaRobotEIP_OutputData	STRUCT	CJ			
	DO1_SpoonsOnTheTurntable	BOOL		0	0	HandshakeToRobot
	DO2_PackingReadyToReceiveSp...	BOOL		0	1	HandshakeToRobot
	DO3_DrivelnTheTrash	BOOL		0	2	

Kuvio 29. Ethernet/IP I/O- datatyyppimäärittelyt.

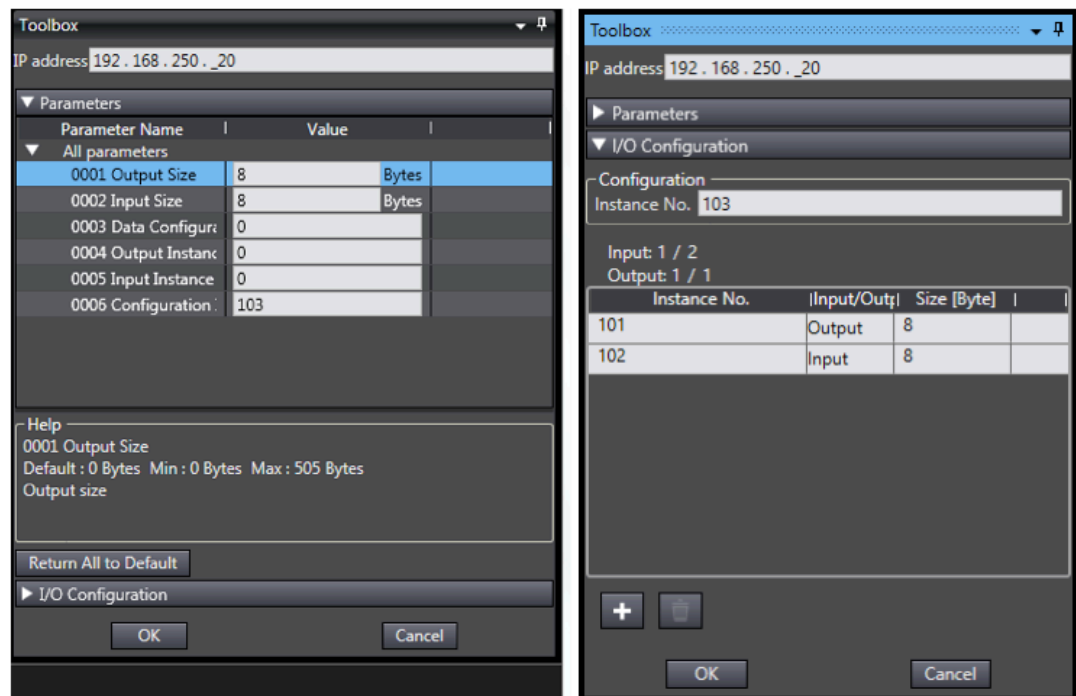
Datatyyppien määrittämisen jälkeen logiikalle oli luotava kaksi globaali- muuttujaa, joihin Ethernet/IP- data tallennettiin. Kuvion 30 muuttujat nimettiin RobEIPInput- ja RobEIPOutput nimillä. Muuttujiin oli määritettävä datatyyppit, jotka oli nimettävä kuvion 29 nimettyjen STRUCT- tietotyyppien mukaisesti.

Name	Data Type	Initial Value	AT	Retain	Constant	Network Publish
RobEIPInput	YaskawaRobotEIP_InputData			<input type="checkbox"/>	<input type="checkbox"/>	Input
RobEIPOutput	YaskawaRobotEIP_OutputData			<input type="checkbox"/>	<input type="checkbox"/>	Output

Kuvio 30. Ethernet/IP I/O- muuttujat.

Sysmac Studion Ethernet/IP yhteysasetuksiin oli luotava uusi yhteys robotille. Ennen yhteyden lisäämistä robotilta ladattu EDS-tiedosto oli siirrettävä Sysmac Studion EDS-

kirjastoon, josta se oli ladattavissa yhteyden lisäämisen yhteydessä. Kuvion 29 mukaisesti luodut globaali muuttujat rekisteröitiin Sysmac Studion Ethernet/IP yhteysasetuksiin. Rekisteröinnin jälkeen lisättiin uusi yhteys, johon määriteltiin robotille annettu IP-osoite, robotin nimi ja ehdotettu revisionumero. Ohjelmisto haki robotin nimen Sysmac Studio- kirjastoon lisätyn EDS- tiedoston mukaisesti. Kuvion 31 mukaisesti luotuun yhteyteen oli määritettävä tarvittavat asetukset.

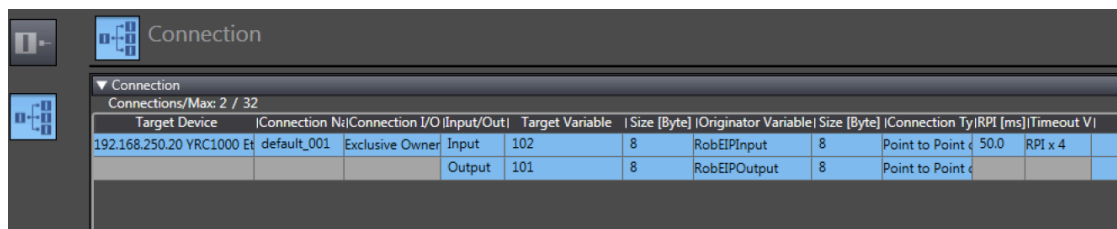


Kuvio 31. Sysmac Studio Ethernet/IP yhteysasetukset.

Asetuksiin oli määritettävä vastaavat asetukset, kuin robotille. Parametri välilehdelle määritettiin I/O kooksi 8 byteä ja konfiguraatio välilehdelle määritettiin instanssiar-

vot ja instanssien koko kuvion 31 mukaisesti. Logiikan output- instanssiarvoksi määritettiin 101, joka robotin asetuksissa oli input- instanssiarvo. Samoin logiikan output- instanssiarvo 102 arvo määritettiin robotilla input- instanssiarvoksi.

Yhteysasetusten määrittämisen jälkeen Sysmac Studio ohjelmassa näkyi luotu yhteys kuvio 32, johon oli kertaalleen lisättävä jo määritellyt instanssiarvot, muuttujien koot, I/O-tiedot ja yhteystyyppi.



Connection											
Connections/Max 2 / 32											
Target Device	Connection No	Connection I/O	Input/Output	Target Variable	Size [Byte]	Originator Variable	Size [Byte]	Connection Type	RPI [ms]	Timeout V	
192.168.250.20 YRC1000 Et	default_001	Exclusive Owner	Input	102	8	RobEIPInput	8	Point to Point	50.0	RPI x 4	
			Output	101	8	RobEIPOutput	8	Point to Point			

Kuvio 32. Sysmac Studio Ethernet/IP- yhteys robotille

Konfiguraatio oli ladattava logiikalle, jolloin logiikka otti käyttöön luodut yhteydet. Yhteyden toimivuutta pystyi monitoroimaan Ethernet/IP yhteysasetuksista tai tekemällä kuvion 33 mukainen testiohjelma. Robotin operointipaneelilta ohjattaessa sen ulkoista lähtöä OUT#0049, kuvion 33 logiikkaohjelmassa rung 3 RobEIPInput.DI1- kosketin aktivoitui. Rung 4 RobEIPOutput.DO1- lähtöön ohjattiin 1s pulssia, jolloin robotin ulkoinen tulo IN#0049 aktivoitui 1s pulssilla. Tällöin tiedettiin, että yhteys logiikalta robottiin ja robotilta logiikkaan oli toimiva.



Kuvio 33. Ethernet/IP- yhteyden testaus.

## 10 Ohjelmointi

Ohjelmointi aloitettiin logiikkaohjelmoinnilla ja turvaohjaimenohjelmoinnilla, jotka voitiin suorittaa etukäteen toimiston työpisteellä. Ohjelmoinnissa hyödynnettiin sähkösuunnittelun benchmarking-haastattelun tuloksia. Tutkimustulosten perusteella sekvenssiaskelluksessa pystyttiin hyödyntämään aiemmin luotua siirtorekisteri-pohjaohjelmarakennetta. Työn toteutuksessa ei tarvinnut keskittyä sekvenssin siirtorekisteri pohjaohjelman ohjelmointiin, vaan ohjelmoinnin pystyi aloittamaan suoraan sekvenssirakenteen suunnittelulla. Logiikkaohjelmoinnissa hyödynnettiin Sysmac Studio-ohjelmiston käyttöohjetta, jonka mukaisesti ohjelmassa käytettyjen ohjelmamoduulien valinta suoritettiin.

### 10.1 Logiikkaohjelmointi

Logiikkaohjelmointi aloitettiin fyysisten tulojen ja lähtöjen yhdistämisellä muuttujiin. Ennen muuttujamäärittelyjä oli tiedettävä mihin fyysisiin tuloihin ja lähtöihin toimilaitteet yhdistetään. Tarvittavat tulo- ja lähtösignaalit selvitettiin sähkösuunnittelun tuloksena luodun I/O-kaavion perusteella, jossa logiikan ja turvalogiikan fyysiset tulo- ja lähtöportit liitettiin toimilaitteisiin. Kuvion 34 mukaisesti logiikan käyttöön tulee 11 tulosignaalia paikat (00-10) ja 10 lähtösignaalia paikat (00-09). Turvalogiikan turva I/O:ta tarvitaan 8- tuloa ja 4- lähtöä, jotka veivät NX-SID800 yksiköltä kaikki turvatulot (si0-si7) ja NX-SOD400- yksiköltä kaikki turvalähdöt (So0-So3). NX-SIH400 ja NX-SOD200 turva I/O-yksiköt toimivat varalla, mikäli turva I/O:ta olisi tarvittu suunniteltua enemmän.

PLC1		Signals	Safety PLC		Signals
DI	00	In_ControlVoltageOn	NX-SID800	Si0	EmergencySignalFromEngelCH1
	01	In_SafetyRelayOn		Si1	EmergencySignalFromEngelCH2
	02	In_PressureOn		Si2	SafetySignalFromEngelCH1
	03	In_CutterRear		Si3	SafetySignalFromEngelCH2
	04	In_CutterFront		Si4	EmergencyStopRelayFeedbackFromK20.1-K20.2
	05	In_TurnTableRear		Si5	SafetyRelayFeedbackFromK21.1-K21.2
	06	In_TurnTableFront		Si6	EmergencyStopResetButton
	07	In_TurnTable ReadyToTurning		Si7	SafetyStopResetButton
	08	In_SpoonsReadyToCutting	NX-SIH400	Si0	
	09	Packing_Ready		Si1	
	10	Packing_AutoOn		Si2	
	11	Reserved	NX-SOD400	Si3	
	12	Rederved		So0	EmergencyStopRelay K20.1
	13	Reserved		So1	EmergencyStopRelay K20.2
DO	00	O_LampGreen	NX-SOH200	So2	SafetyStopRelay K21.1
	01	O_LampOrange		So3	SafetyStopRelay K21.2
	02	O_LampRed		So0	
	03	O_Cutter		So1	
	04	O_TurntableBacward			
	05	O_TurntableForward			
	06	O_TurntableReadyForFilling			
	07	O_CuttingComplete			
	08	O_PackingStart			
	09	O_FestoAirSupply			

Kuvio 34. Ohjelmoitavan logiikan ja turvaohjaimen I/O-kaavio.

### 10.1.1 Fyysisten tulojen ja lähtöjen yhdistäminen muuttujiin

NX1P2- logiikka sisälsi 24- kappaletta fyysistä I/O:ta, jonka muuttujat olivat määritet-  
tävässä Sysmac Studion I/O Map- osiossa. Kuvion 35 mukaisesti kullekin tulolle ja läh-  
dölle oli nimettävä muuttuja (variable). Muuttuja nimettiin sähkösuunnittelussa luo-  
dun I/O- listan perusteella. Muuttujien nimeämisessä kaikki tulot käyttötarkoituk-  
sesta riippumatta nimettiin I- alkuisiksi, joka tarkoittaa Input- signaalia. Kaikkien läh-  
döt merkattiin O- alkuisiksi, joka viittasi lähtöön eli output- signaaliin. Muuttuja-  
nimessä käytetty alkukirjain helpotti ohjelman lukua ja symbolien hakua ohjelmointi-  
vaiheessa.

Position	Port	Description	R/W	Data Type	Variable	Variable Comment	Variable Type
Node10	EtherCAT Network Configuration						
	NX-ECC203						
	CPU/Expansion Racks						
	Built-in I/O Settings						
	Input Bit 00	Input Bit 00	R	BOOL	I_ControlVoltageOn	DC2	Global Variables
	Input Bit 01	Input Bit 01	R	BOOL	I_SafetyReadyOn	DC3	Global Variables
	Input Bit 02	Input Bit 02	R	BOOL	I_PressureOn		Global Variables
	Input Bit 03	Input Bit 03	R	BOOL	I_CutterRear		Global Variables
	Input Bit 04	Input Bit 04	R	BOOL	I_CutterFront		Global Variables
	Input Bit 05	Input Bit 05	R	BOOL	I_TurntableRear		Global Variables
	Input Bit 06	Input Bit 06	R	BOOL	I_TurntableFront		Global Variables
	Input Bit 07	Input Bit 07	R	BOOL	I_TurntableReadyToTurning	HandshakeFromEngel	Global Variables
	Input Bit 08	Input Bit 08	R	BOOL	I_SpoonsReadyToCutting	HandshakeFromEngel	Global Variables
	Input Bit 09	Input Bit 09	R	BOOL	I_PackingReady	HandshakeFromPacking	Global Variables
	Input Bit 10	Input Bit 10	R	BOOL	I_PackingAutoOn	HandshakeFromPacking	Global Variables
	Input Bit 11	Input Bit 11	R	BOOL	I_Reserved11		Global Variables
	Input Bit 12	Input Bit 12	R	BOOL	I_Reserved12		Global Variables
	Input Bit 13	Input Bit 13	R	BOOL	I_Reserved13		Global Variables
	Output Bit 00	Output Bit 00	RW	BOOL	O_LampGreen		Global Variables
	Output Bit 01	Output Bit 01	RW	BOOL	O_LampOrange		Global Variables
	Output Bit 02	Output Bit 02	RW	BOOL	O_LampRed		Global Variables
	Output Bit 03	Output Bit 03	RW	BOOL	O_Cutter		Global Variables
	Output Bit 04	Output Bit 04	RW	BOOL	O_TurntableRwd		Global Variables
	Output Bit 05	Output Bit 05	RW	BOOL	O_TurntableFwd		Global Variables
	Output Bit 06	Output Bit 06	RW	BOOL	O_TurntableReadyForFilling	HandshakeToEngel	Global Variables
	Output Bit 07	Output Bit 07	RW	BOOL	O_CuttingCompleted	HandshakeToEngel	Global Variables
	Output Bit 08	Output Bit 08	RW	BOOL	O_PackingStart	StartSignalToPM	Global Variables
	Output Bit 09	Output Bit 09	RW	BOOL	O_FestoAirSupply	K40_FullPressure	Global Variables
OptionB	Option Board Settings						
NXBusV	NX Bus Master						

Kuvio 35. Ohjelmoitavan logiikan I/O:n yhdistäminen muuttujiin.

Pakkaus- ja ruiskuvalukoneen kättelytiedoille lisättiin kommenttikenttään lisäkommentti, jolla selvennettiin signaalin kulkusuuntaa laitteiden välillä. Jokainen muuttuja määritettiin ulkoiseksi muuttujaksi (Global Variables), jolloin ne olivat jokaisen ohjelmamoduulin käytettävissä. Ulkoisiksi muuttujiksi määritetyt muuttujat lisättiin automaattisesti Sysmac Studion ulkoiset muuttujat (Global Variables) kirjastoon, jossa sijaitsee kaikki luodut ulkoiset muuttujat.

### 10.1.2 Ohjelmarakenne

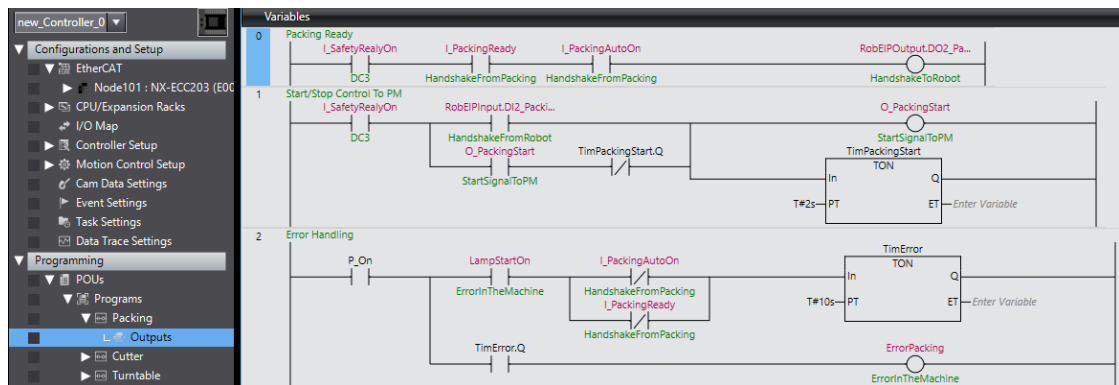
Jokaiselle toimilaitteelle muodostettiin oma ohjelmamoduuli, jossa kyseisen ohjelman ohjelmointi tapahtui. Ohjelmat- osioon luotiin ohjelmamoduulit pakkauksen (Packing), leikkaimen (Cutter), kääntöpöydän (Turntable), asetuksien (Settings) ja häiriöiden (Error) ohjauksille. Kukin ohjelmamoduuli suoritti omaa ohjelmakiertoa välittämättä toisten ohjelmakierroista. Jokainen ohjelmamoduuli oli jaettu pienempiin ohjelmamoduuleihin, niin sanottuihin aliohjelmiin. Leikkaimen ja kääntöpöydän ohjelmamoduulit sisälsivät aliohjelmamoduulit asetuksille (Settings), sekvenssille (Sequence) ja lähdöille (Outputs). Aliohjelmissa sijaitsi kirjoitettu ohjelmakoodi. Ohjelma jaettiin osakokonaisuuksiin eli askeleisiin (Rung). Askeleet on jaettu numeroin nollasta alkaen.



### 10.1.3 Pakkaukoneen ohjaus

Pakkaus- (Packing) ohjelmamoduulissa käsiteltiin ainoastaan robotin ja pakkauskoneen välisiä I/O-liityntärajapintaohjauksia. Logiikan ja pakkauskoneen väliset I/O-liityntärajapinnat oli välireleistetty, jolloin kummankin laitteen jännitteet pysyivät omassa potentiaalissaan. Teollisuusrobotti tarvitsi pakkauskoneelta tiedon, milloin lusikat sai viedä pakkauskoneelle. Robotti puolestaan aktivoi käynnistys- signaalin pakkauskoneelle jätetyn lusikkapinon jälkeen. Pakkauskoneen ohjelmointi toteutettiin relekaavio-ohjelmointina.

Kuvion 36 pakkausohjelma sisälsi kolme askelta, joista askel 0 oli ohjaussignaali Ethernet/IP- kenttäväylän yli robotille, jossa annettiin robotille vientilupa pakkauskoneelle. Vientiluvan ehtona turvapiirien oli oltava suljettuna ja pakkauskone oli oltava valmiustilassa sekä asetettu automaattiajolle (AutoOn). Pakkauskoneen valmius- ja automaattiajo tulosignaalit tulivat pakkauskoneelta liityntärajapinnan yli logiikan fyysisiin tuloihin I/O-listan mukaisesti paikkoihin IN 09 ja IN 10.



Kuvio 36. Logiikkaohjelmamoduuli pakkaus.

Kuvion 36 askel 1 käsitteli pakkauskoneen käynnistys- signaalia. Robotin antama käynnistyskäsky ohjasi suoraan logiikan fyysistä lähtöä OUT 08 (PackingStart), joka antoi käynnistyskäskyn pakkauskoneelle. Robotin antama käynnistyspulsси oli niin lyhyt, että käynnistyskäskyyn tehtiin 2s omapito, jolla pidennettiin käynnistyskäskyn pituutta.

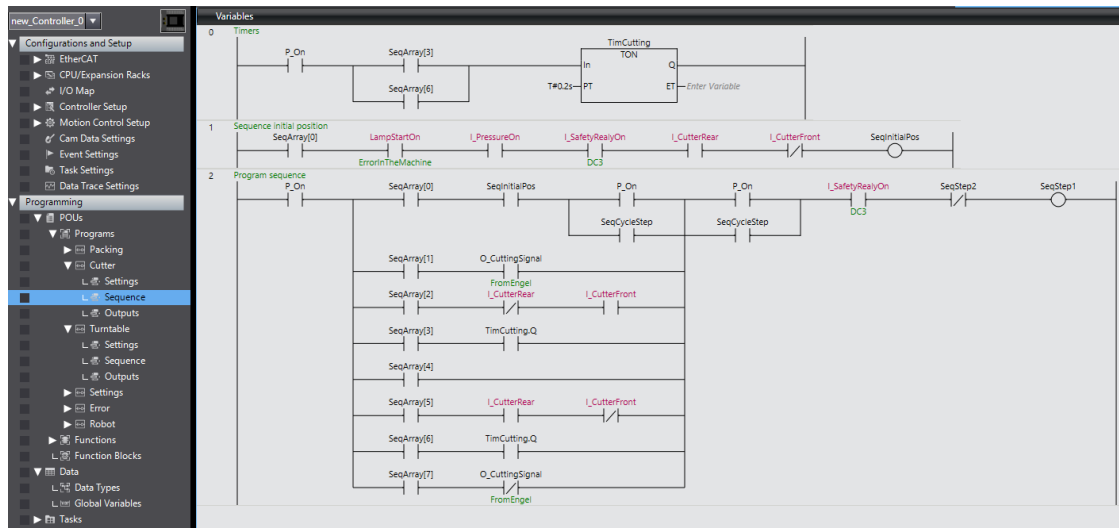
Askel 2 käsitteli häiriögenerointia mahdollisen pakkauskonehäiriön ilmoittamiseksi. Tuotantoajon aikana ilmaantuvat häiriöt pakkauskoneessa tai pakkauskoneen automaattiajotilasta poistuminen aktivoi häiriön 10s kuluessa. Häiriö aktivoi valotornin punaisen valon ja robotille annettu lusikoiden venttilupa poistuu välittömästi. Käynnistyskäsken omapidolle ja häiriöaktivoinnissa käytetyt ajastimet lisättiin paikallismuuttujiin, jolloin kyseiset muuttujat olivat käytettävissä ainoastaan pakkaus- ohjelmamoduulin sisällä.

#### 10.1.4 Leikkaimen ohjaus

Leikkaimen ohjaukseen tarvittiin ruiskuvalukoneen lineaarirobotilta kättelytieto logiikalle, milloin lusikkaviuhka on valmis leikattavaksi. Logiikan oli välitettävä tieto lineaarirobotille suoritetusta valuylijäämäleikkauksesta. I/O-liityntärajapinnat laitteiden välillä toteutettiin fyysisellä I/O:lla välireleitä käyttäen, jolloin laitteiden jännitteet pysyivät omassa potentiaalissaan. Logiikka ohjasi leikkaussylinteriä ohjaavaa 5/2- paineilmaventtiiliä, jonka liikettä seurattiin sylinteriin kiinnitetyillä antureilla.

Leikkaimen ohjauksen ohjelmointi toteutettiin relekaavio-ohjelmointina ja sekvenssiohjausohjelmointina, jolloin työvaiheet etenivät askel kerrallaan. Leikkaimen ohjelmamoduuli jaettiin asetukset, sekvenssi ja lähdöt ohjelmamoduuleihin. Asetukset ohjelmamoduuli sisälsi sekvenssin askellukseen vaaditun askelrekisterin. Sekvenssi ohjelmamoduuli sisälsi ohjaussekvenssin leikkaussylinterin ohjaukselle ja sen suoritettujen liikkeiden valvonnalle. Lähdöt ohjelmamoduuli sisälsi leikkaimen ohjauksen ja I/O-liityntärajapintojen kättelytiedot ruiskuvalukoneen lineaarirobotin välille.

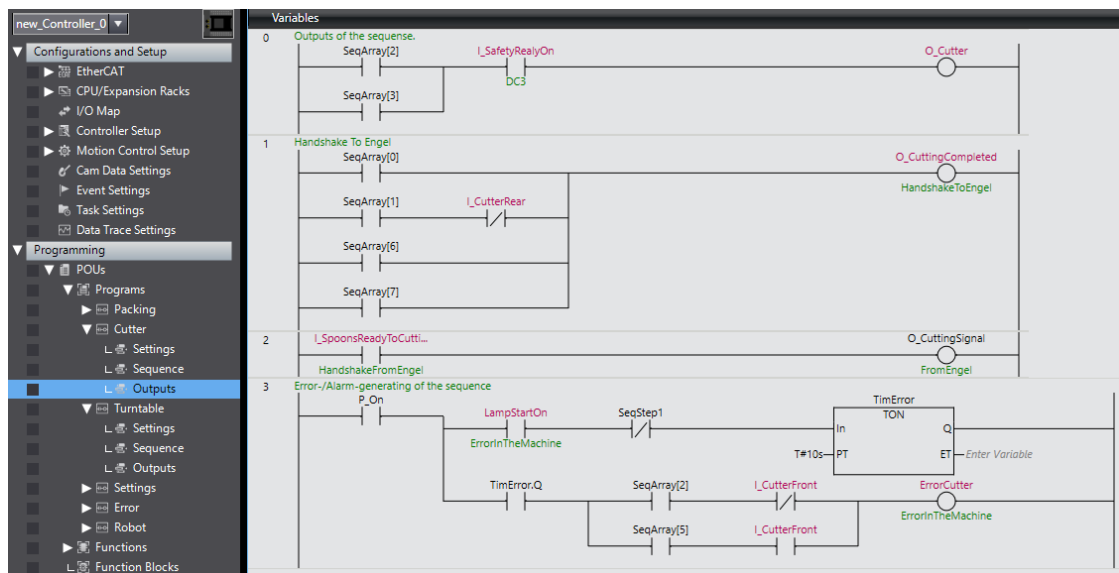
Kuvion 37 esittää sekvenssiohjelmamoduulin, jossa ohjelma on jaettu kolmeen askeleseen. Askel 0 sisältää ainoastaan leikkausviive ajastimen, jolla oli tarkoitus rauhoittaa leikkaustoimenpidettä. Leikkausviivettä käytettiin sylinterin ohjauksen jälkeen, jolloin sylinteri odotti etuasennossa 0.2 sekuntia eturajan aktivoitumisen jälkeen. Viivettä käytettiin myös sylinterin palatessa perusasentoonsa, jolloin suoritetusta leikkauksesta ruiskuvalukoneen lineaarirobotti sai signaalin 0.2 sekuntia ylärajan aktivoitumisen jälkeen.



Kuvio 37. Logiikkaohjelmamoduuli leikkaimen sekvenssiohjaus.

Kuvion 37 askel 1 määrittä alkuehdot sekvenssiaskellukselle. Ennen sekvenssin askelluksen aloittamista varmistettiin, että leikkaussylinteri oli halutussa alkuasennossa, paineilmat kytketty, turvapiirit kuitattu ja robotti asetettu automaattijolle.

Ehtojen täytyttyä askel 2 tikapuukaavio askelsi ensimmäiselle riville, jossa odotettiin ruiskuvalukoneen lineaarirobotin signaalia leikkauksen aloituksen sallimisesta. Signaalin aktivoituessa sekvenssi askelsi eteenpäin, jolloin leikkaus sylinterinohjaus sallittiin kuvion 38 askeleen 0 mukaisesti. Tikapuukaaviossa edettiin seuraaville askelleille sylinterin saavuttamien rajatietojen perusteella. Leikkauksen jälkeen logiikka välitti tiedon lineaarirobotille suoritetusta leikkauksesta kuvion 38 askel 1 mukaan. Lusikkaviuhkan tuonti leikkaimelle sallittiin ainoastaan leikkaus suoritettu- tiedon ollessa pois päältä. Leikkaus- suoritettu signaalia pidettiin päällä niin kauan, että sylinterinliike oli palannut perusasentoonsa. Tällä varmistettiin, että leikkain on perusasennossa ennen, kuin lusikkaviuhkan tuontilupa lineaarirobotille aktivoidaan.



Kuvio 38. Logiikkaohjelmamoduuli leikkaimen lähdöt.

Leikkainsylinterin häiriögenerointi tapahtui kuvion 38 askeleen 3 mukaisesti. Auto-  
maattiajolla generoitiin häiriö, mikäli leikkainsylinteri ei saavuttanut päätyrajaa 10s  
kuluessa sitä ohjattaessa.

### 10.1.5 Kääntöpöydän ohjaus

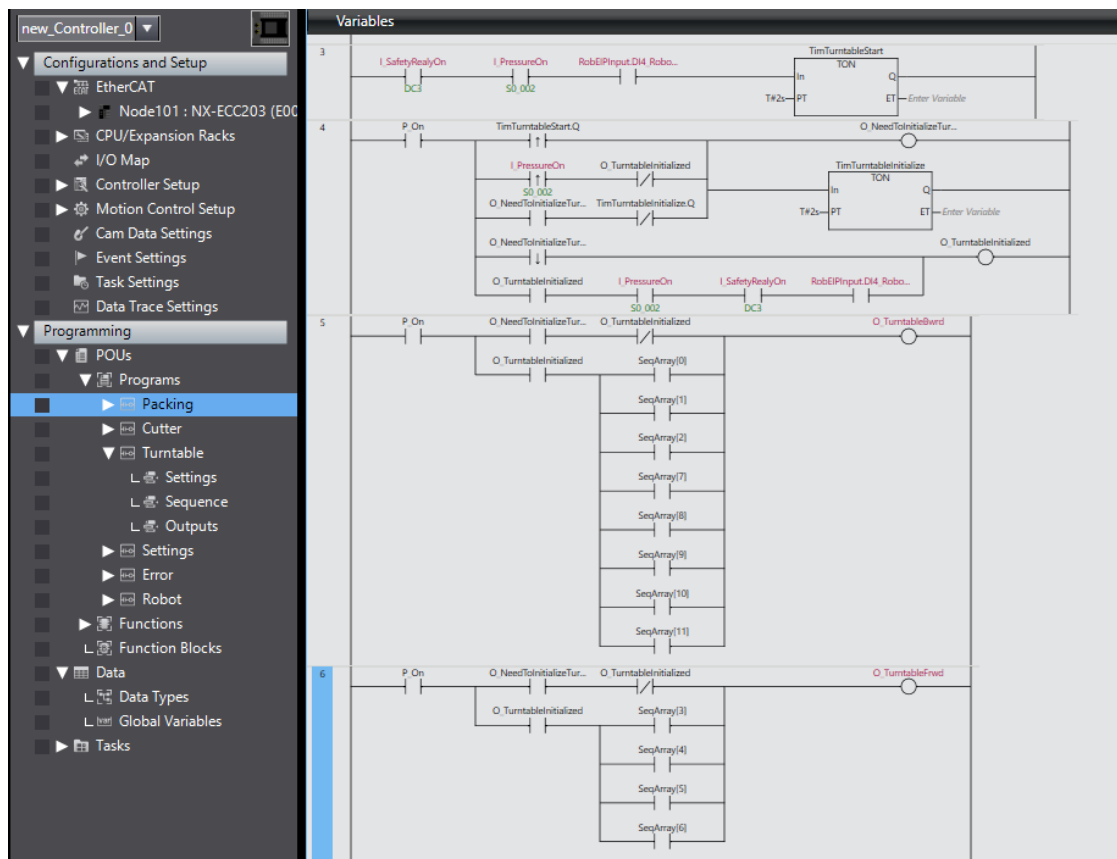
Kääntöpöydän ohjelmamoduulien rakenne toteutettiin yhdenmukaisesti leikkaimen  
ohjelmamoduulien kanssa. Kääntöpöydän ohjelmamoduulit sisältävät asetukset, sek-  
venssi ja lähdöt ohjelmamoduulit. Ohjelmointi toteutettiin relekaavio-ohjelmointina  
ja sekvenssiohjausohjelmointina.

Logiikka ohjasi kääntöpöytää ruiskuvalukoneen lineaarirobotin ja lusikoiden käsitte-  
lyyn käytetyn teollisuusrobotin työvaiheiden mukaisesti. Logiikka antoi lineaarirobo-  
tille lusikoiden tuontiluvan (OUT 06), kun kääntöpöytä oli perusasennossa. Lineaari-  
robotti täytti alustan ja välitti logiikalle kääntöalusta täynnä tiedon täytetystä kääntö-  
alustasta (IN 07). Logiikka suoritti kääntöalustan käännön, jolloin täysi lusikkapino  
siirtyi teollisuusrobotin tyhjennettäväksi ja tyhjä alusta lineaarirobotin täytettäväksi.  
Käännön jälkeen logiikka välitti teollisuusrobotille tyhjennysluvan ja lineaarirobotille

täyttöluvan. Kääntöpöytää ei käännetty ennekuin kummatkin robotit olivat ilmoittaneet työnsä suoritetuksi.

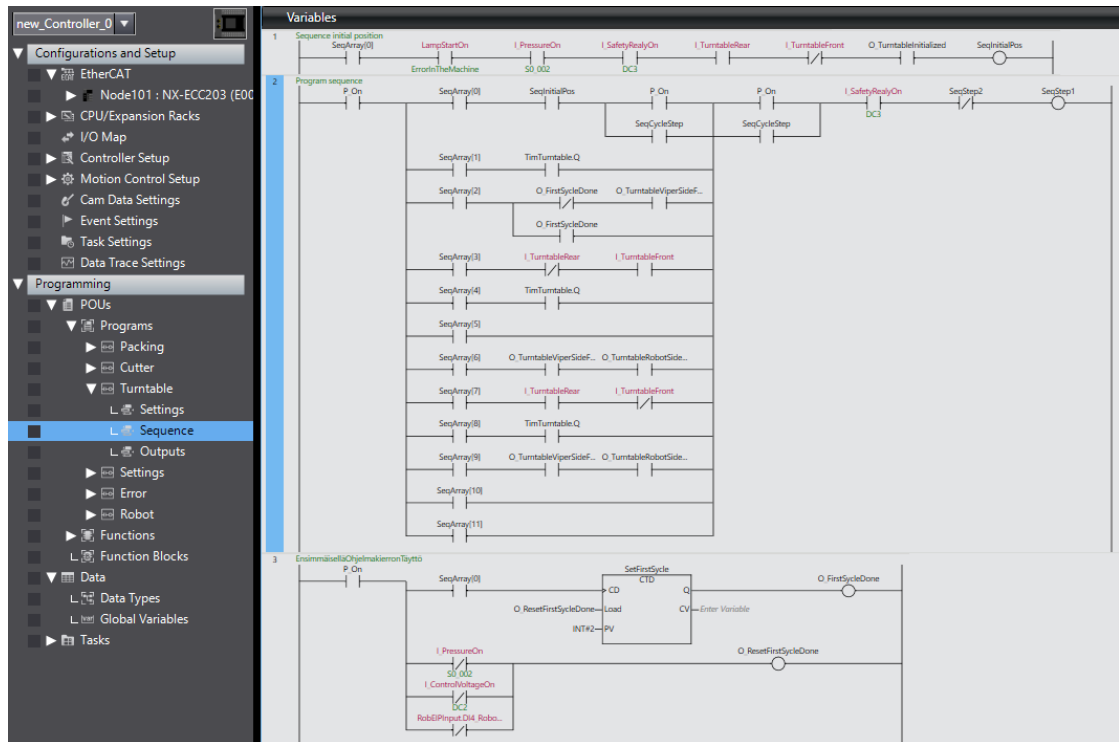
Kääntöpöydän ohjaus toteutettiin kahdella 5/2- suuntaventtiilillä, jolla mahdollistettiin hallittu kääntötoimenpide kumpaankin kääntösuuntaan. Ennen kääntöpöydän paineilmaohjauksen kytkemistä kääntöpöytä täytyi alustaa. Tämä tarkoitti, että kääntöpöydän asettaminen perusasentoon toteutettiin siten, että paine ohjattiin molempiin pyöröpöydän suuntaventtiileihin lähdöillä OUT 4 ja OUT 5. Paineiden kytkemisen jälkeen ohjaus poistettiin OUT 5 lähdöltä, jolloin pyöröpöytä pystyttiin kääntämään perusasentoonsa hallitusti.

Kuvion 39 esittää kääntöpöydän ohjauksen toteuttavaa ohjelmaa. Askel 3 ja 4 ohjelmalohkoissa määritellään pyöröpöydän alustus. Alustus tarvittiin, mikäli turvasuojat oli avattu, paineilmasyöttö poistunut tai robottiohjelmasuoritus aloitettiin alusta. Alustusta vaadittaessa paine kytkettiin kummallekin puolelle kääntöpöytää 2 sekunnin kuluttua, kun paineilma oli aktivoitu, turvasuojat kuitattu ja robotti asetettu automaattiajolle. OUT 5- ohjaus poistettiin 2 sekunnin kuluttua yhdenaikaisesta paineenkytkennästä ja kääntöpöytä ohjattiin perusasentoon askel 5 mukaisesti. Automaattiajossa kääntöpöydän ohjaus tapahtuu askel 5 ja askel 6 mukaisesti.



Kuvio 39. Logiikkaohjelmamoduuli kääntöpöydän ohjaus.

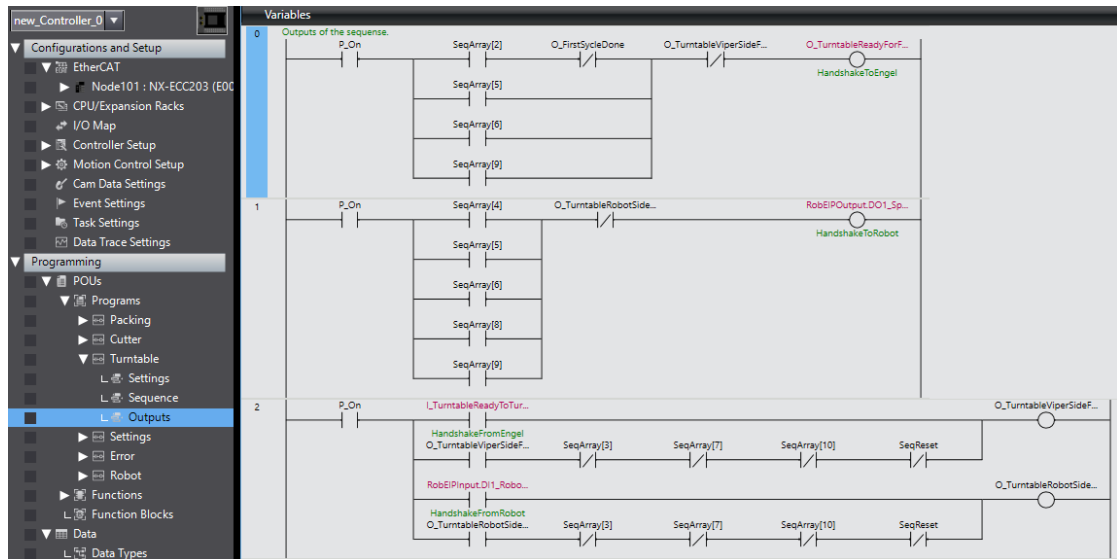
Kääntöpöydän sekvenssimoduulissa kuvio 40 askel 1 määritti sekvenssin alkuehdot, jossa tarkistettiin, että robotti on automaattiajolla, kääntöpöytä perusasennossa, paineilmansyöttö aktivoitu ja turvapiirit kuitattu. Ehtojen täytyttyä kuvion 40 askel 2 mukaisesti sekvenssi jäi odottamaan kääntöpöydän täyttymistä. Ensimmäisellä ohjelmakierrolla teollisuusrobotille ei voitu antaa tyhjennyslupaa, koska kääntöpöytä oli vielä tyhjä. Ensimmäisen ohjelmakierron jälkeen teollisuusrobotin sai tyhjennysluvan jokaisen suoritettun käännön jälkeen. Tästä johtuen ensimmäinen ohjelmakierto oli tunnistettava ohjelmallisesti, joka oli toteutettu askel 3 mukaisesti. Kyseisessä ohjelmalohkossa laskettiin ohjelmakierrat ja ensimmäisen ohjelmakierron jälkeen paikallismuuttuja aktivoitiin. Muuttuja ohitti askel 2 sekvenssitikapuu askeleen 2, jossa odotettiin ainoastaan lineaarirobotin täyttöö.



Kuvio 40. Logiikkaohjelmamoduuli sekvenssiohjaus.

Kuvion 40 askel 2 tikapuukaavion ohjelmakierto toteutui kääntöpöydän ohjauksesta ja käännön rajatietojen seuraamisesta, jonka jälkeen odotettiin lineaarirobotin ja teollisuusrobotin työvaiheiden suorittamista. Suoritettujen työvaiheiden jälkeen kääntöpöytä sai luvan kääntyä ja robottien työvaiheet alkoivat alusta, kunnes tehtävät oli jälleen suoritettu.

Kuvio 41 askel 0 ohjelmalohkoissa annettiin täyttölupa lineaarirobotille ja askel 1 ohjelmalohkossa tyhjennyslupa teollisuusrobotille. I/O-liityntärajapinnat aktivoitiin aina pyöröpöydän käännön jälkeen. Kääntöpöydälle ei annettu kääntölupaa ennen, kuin lineaarirobotti ja teollisuusrobotti olivat suorittaneet työvaiheensa kuvion 41 askel 2 mukaisesti. Robottien työvaiheille luotiin sisäiset apumuuttujat, joita käytettiin tikapuukaaviossa. Apumuuttijien avulla kättelysignaaleille pystyttiin ohjelmoimaan omapito, joka aktivoitui lineaarirobotin ja teollisuusrobotin pulssimuotoisesta signaalista. Omapito nollattiin, kun kumpikin robotti oli saanut työvaiheensa suoritettua ja kääntöpöytä sai luvan kääntyä.



Kuvio 41. Ohjelmamoduuli kääntöpöydän ulkoiset kättelyt.

## 10.2 Turvaohjaimen ohjelmointi

Turvaohjaimen ohjelmointi aloitettiin turvakomponenttien liittämällä turva I/O-yksiköihin ja muuttujan nimeämisillä. Turvakomponentit lisättiin sähkösuunnittelun tuloksena luodun kuvion 34 I/O-kaavion mukaisesti. Turvaohjaimen ohjelmointi toteutettiin Sysmac Studiolla FBD- (Function Block Diagram) ohjelmointina. Häätä-seis- ja turvapiirin ohjauksille luotiin omat ohjelmamoduulit. Ohjelmointi toteutettiin Sysmac Studio- käyttöohjetta hyödyntäen.

### 10.2.1 Riskianalyysi

Ennen turvapiirien suunnittelua ja ohjelmointia oli tiedettävä millaisilla turvatoimilla ja turvakomponenteilla automaattikäyttöön liittyviä riskejä lähdettiin minimoimaan. Riskien arviointi suoritettiin riskianalyysin avulla, johon osallistui tilaajayrityksen johto sekä sähkö- ja mekaniikkasuunnittelu. Riskianalyysi suoritettiin matriisimenetelmän avulla, jossa riskien todennäköisyyttä verrattiin riskin vakavuuteen. Riskianalyysin tuloksena määritettiin mekaaniset ja sähköiset ratkaisut, joiden yhteistuloksena käyttäjäturvallisuusriskit saatiin vähäiseksi tai siedettäväksi.



### 10.2.2 Turvapiirin rakenne

Turvapiirien rakenne koostui hätä-seispiiristä ja turvapiiristä. Hätä-seispainikkeiden turvalliseen tilaan saattaminen mahdollisti ohjausjännitteiden kytkennän automaattiin. Ohjausjännitteet kytkivät ohjauskeskukseen sarjaan kytkettyjä hätä-seis releitä, jotka ohjasivat 24VDC- jännitesyötön ohjauskeskuksen DC2- riviliittimeen. DC2- puoleltaan aktivoi alennetunpaineen kytkeytymisen pneumaattisiin toimilaitteisiin sekä jännitteensyötön logiikan I/O:lle. DC2- jännitteiden kytkennät eivät salli toimilaitteiden liikekäskyjen suorittamista.

Turvakomponenttina toimivien turvasuojien ovikytkimien aktivointi mahdollisti automaatin turvapiirien kuittaamisen. Turvaovien kuittauspainike oli turvaoven vieressä, joka aktivoi ohjauskeskuksessa olevat sarjaan kytketyt turvareleet. Releet ohjasivat 24VDC jännitteensyötön DC3- riviliittimelle. DC3- jännitesyöttö aktivoi täyden työpaineen mekaanisille toimilaitteille ja salli toimilaitteiden ja teollisuusrobotin liikekäskyjen suorittamisen.

Turvaohjaimen lähdeillä ohjattujen hätä-seis ja turvareleiden tarkoituksena oli vähentää turvaohjaimessa tarvittavien turvalähtöjen määrää. Ohjatut turvareleet sisälsivät kuusi kärkitietoa, jotka olivat vapaasti käytettävissä hätä-seis ja turvapiireihin kytketyille toimilaitteille. Ohjattuja releitä oli hätä-seis ja turvapiirillä kaksi kappaletta, joiden koskettimet olivat sarjaan kytkettyjä.

Kaikki automaatin hätä-seis- painikkeet mukaan lukien teollisuusrobotin ja pakkaus-koneen kytkettiin osaksi ruiskuvalukoneen kahdennettua hätä-seispiiriä. Ruiskuvalukoneen hätä-seis releeltä otettiin kahdennettu kärkitieto turvaohjaimen turvatuloihin (Si0 ja Si1). Tällöin turvaohjain sai tiedon minkä tahansa hätä-seispainikkeen aktivoinnista. Hätä-seispainikkeen aktivoituessa turvaohjain poisti välittömästi ohjauskeskuksen hätä-seis releiden ohjaukset (So0 ja So1), jotka poistivat automaatin paineilma- ja jännitteensyötöt.

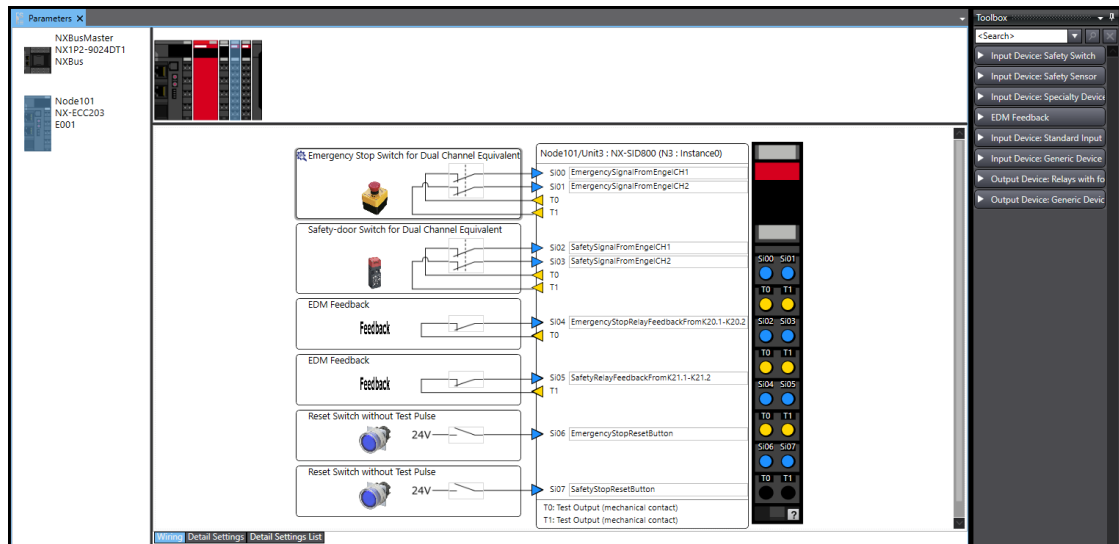
Linjan kaikki turvasuojat kytkettiin osaksi ruiskuvalukoneen kahdennettua turvapiiriä ja ruiskuvalukoneen turvareleeltä otettiin turvaohjaimeen turvatuloihin (Si2 ja Si3)

kahdennettu kärkitieto. Ruiskuvalukoneen turvareleen kuittaus mahdollistettiin, kun kaikki automaatin turvasuojat olivat aktivoituna turvalliseen tilaan. Tällöin turvapiiri oli mahdollista kuitata turvapiirin kuittauspainikkeesta. Kuittaus mahdollisti ohjauskeskuksen turvareleiden ohjaukset (So2 ja So3), jotka mahdollistivat DC3- jännitteensyötön aktivoitumisen. Turvaovissa käytettiin lukollisia turvakytkimiä, jotka lukitsivat turvaovet kuittauksen jälkeen. Ovien avaus mahdollistettiin avauspyyntöpainikkeen aktivoimisen jälkeen.

### 10.2.3 Turvakomponenttien yhdistäminen turva I/O- yksiköihin

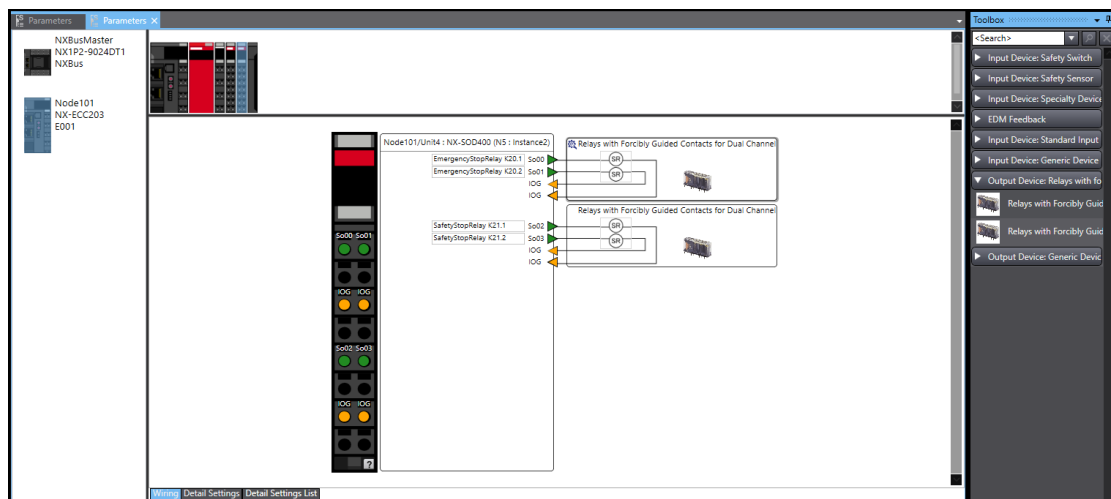
Laitekonfiguraatioissa määritetyille turva I/O:lle oli määritettävä käytettävät turvakomponentit. Turvakomponentit lisättiin sähkösuunnittelun tuloksena luodun kuvion 34 I/O-kaavion mukaisesti.

Kuvio 42 esittää Sysmac Studio- ohjelmiston NX-SID800- turvatuloyksikköön liitetyt turvakomponentit. Turvakomponentit lisättiin kuvion 42 oikean reunan laitekirjastosta. 8- paikkaisen turvatuloyksikön turvatuloihin Si00 ja Si01 liitettiin ruiskuvalukoneen hätä-seis- releen kärkitiedot. Turvatuloihin Si02 ja Si03 ruiskuvalukoneen turvareleen kärkitiedot. Si04 turvatuloon kytkettiin takaisinkytkentä turvalogiikan lähdöillä So00 ja So01 ohjatuista hätä-seis releistä. Si05 turvatuloon kytkettiin takaisinkytkentä turvalogiikan lähdöillä So02 ja So03 ohjatuista turvareleistä. Turvatuloihin Si04 ja Si05 kytketyillä takaisinkytkentätiedoilla valvottiin ohjattujen turva- ja hätä-seis releiden toimintaa. Si06 turvatuloon kytkettiin ohjausjännitepainike, jolla aktivoitiin automaatin hätä-seispiiri. Si07 turvatuloon kytkettiin turvaovien kuittauspainike, jolla aktivoittiin turvapiiri.



Kuvio 42. Turvatulojen yhdistäminen turvatuloyksikköön.

Kuvio 43 esittää NX-SOD400 turvalähtöyksikköön kytketyt ohjattavat turvakomponentit. 4- paikkaisen turvalähtöyksikön turvalähdöt So00 ja So01 on kytketty ohjauskeskuksen hätä-seis- releiden ohjaukseen. Turvalähtöihin So02 ja So03 on kytketty ohjauskeskuksen turvareleiden ohjaukseen.



Kuvio 43. Turvalähtöjen yhdistäminen turvalähtöyksikköön.

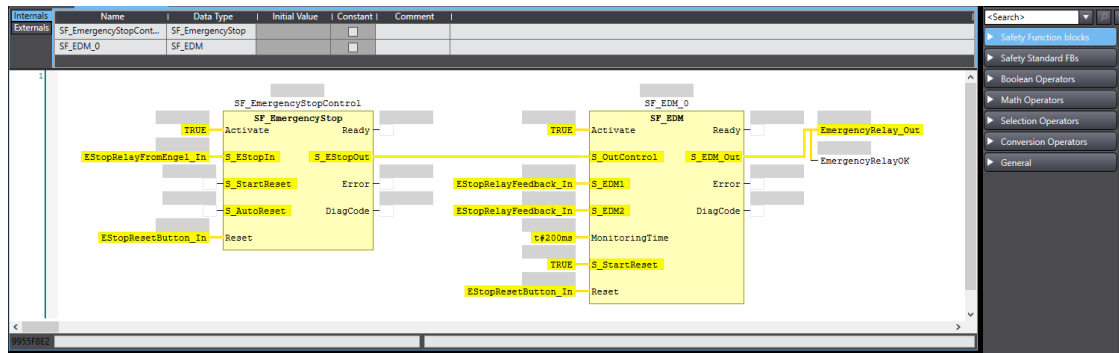
Turvakomponenttien lisäyksen jälkeen käytössä oleville turvatuloille ja turvalähdöille nimettiin muuttujat kuvion 44 mukaisesti. Komponenttien lisäysvaiheessa lisätty kommentti tallentui automaattisesti muuttujakommenttikenttään (Variable Comment). Muuttujat määritettiin ulkoisiksi muuttujiksi, jolloin ne olivat jokaisen ohjelmamoduulin käytettävissä. Kahden kärkitiedon turvakomponenteille oli mahdollista määrittää yksi muuttujan nimi, jolloin molemmat kärkisignaalit olivat yhden muuttujan alla.

Position	Port	R/W	Data Type	Variable	Variable Comment	Variable Type
Parameters I/O Map X						
▼ CPU/Expansion Racks						
NXBusV	NX1P2-9024DT1					
▼ EtherCAT Network						
EtherCAT	Master					
Node10	▼ NX-SID800					
	▼ Safety Inputs					
	Si00 Logical Value	R	SAFEBOOL	EStopRelayFromEngel_In	EmergencySignalFromEngelCH1	Global Variables
	Si01 Logical Value	R	SAFEBOOL		EmergencySignalFromEngelCH2	
	Si02 Logical Value	R	SAFEBOOL	SafetyRelayFromEngel_In	SafetySignalFromEngelCH1	Global Variables
	Si03 Logical Value	R	SAFEBOOL		SafetySignalFromEngelCH2	
	Si04 Logical Value	R	SAFEBOOL	EStopRelayFeedback_In	EmergencyStopRelayFeedbackFromK20.1	Global Variables
	Si05 Logical Value	R	SAFEBOOL	SafetyRelayFeedback_In	SafetyRelayFeedbackFromK21.1-K21.2	Global Variables
	Si06 Logical Value	R	SAFEBOOL	EStopResetButton_In	EmergencyStopResetButton	Global Variables
	Si07 Logical Value	R	SAFEBOOL	SafetyResetButton_In	SafetyStopResetButton	Global Variables
	► Status					
Node10	▼ NX-SOD400					
	► Status					
	▼ Safety Outputs					
	So00 Output Value	W	SAFEBOOL	EmergencyRelay_Out	EmergencyStopRelay K20.1	Global Variables
	So01 Output Value	W	SAFEBOOL		EmergencyStopRelay K20.2	
	So02 Output Value	W	SAFEBOOL	SafetyRelay_Out	SafetyStopRelay K21.1	Global Variables
	So03 Output Value	W	SAFEBOOL		SafetyStopRelay K21.2	

Kuvio 44. Turva I/O:n muuttujamäärittelyt.

## 10.2.4 Turvaohjaimen ohjelmointi

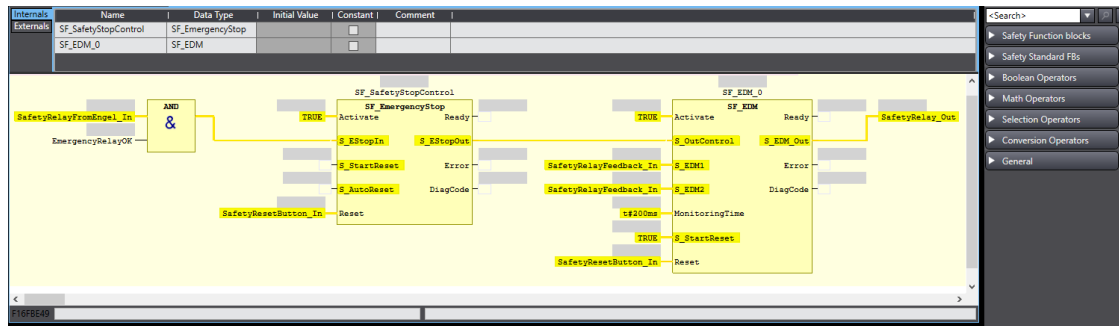
Kuvion 45 esittää hätä-seispiirin ohjaukseen käytettyjä ohjelmasympboleita ja niiden keskinäisiä kytkentöjä. Kuvion 45 SF\_EmergencyStop- ohjelmasympolin tarkoituksena oli monitoroida hätä-seispiirin toimintaa. Ohjelmasympolin tulo S\_EstopIn kytkettiin turvatuloihin Si00 ja Si01 määritettyyn muuttujaan, joka aktivoitui ruiskuvalukoneen hätä-seisreleen kärkitietojen mukaisesti. Ohjelmasympolin aktivointiin vaadittiin ehjän hätä-seispiirin lisäksi kuittaus ohjausjännitepainikkeelta EmergencyReset- (Si6) tuloon, jolloin ohjelmasympolin S\_EstopOut- lähtö aktivoitui.



Kuvio 45. Hätä-seis- piirin ohjaus.

Ohjelmasympolin SF\_EDM tarkoituksena oli seurata ohjattujen hätä-seis releiden toimintaa takaisinkytkennän avulla. SF\_EmergencyStop ohjelmasympolin aktivoiduttua SF\_EDM ohjelmasympoli aktivoi turvalähdöt So00 ja So01, jotka ohjasivat ohjauskeskuksen hätä-seisreleitä. Releiltä oli tultava takaisinkytkentä 200ms kuluessa tuloihin S\_EDM1 ja S\_EDM2, jolloin kyseinen ohjelmasympoli jäi aktiiviseen tilaan jättäen S\_EDM\_Out- lähdön päälle ja hätä-seispiirin aktiiviseksi. SF\_EmergencyStop- ja SF\_EDM ohjelmasympolit olivat keskenään sarjaan kytkettyjä, jolloin hätä-seispiiri katkaisti välittömästi minkä tahansa hätä-seispainikkeen aktivoituessa, tai ohjattujen hätä-seisreleiden takaisinkytkentätiedon katkettua. Hätä-seispiirin uudelleen käynnistäminen vaati kaikkien hätä-seispainikkeiden saattamista valmiustilaan, jonka jälkeen hätä-seispiiri voitiin kuitata ohjausjännitepainikkeesta.

Kuvio 46 esittää turvapiirin ohjaukseen käytettyjä ohjelmasympoleita ja niiden keskinäistä kytkentää. Turvapiiriohjauksessa hyödynnettiin samoja SF\_EmergencyStop ja SF\_EDM ohjelmasympoleita, kuin hätä-seispiirin ohjauksessa. SF\_EmergencyStop- ohjelmasympolia käytettiin turvapiirin monitorointiin. Ohjelmasympolin tulo S\_EStopIn kytkettiin turvatuloihin Si02 ja Si03 määritettyyn muuttuun, joka aktivoitui ruiskukoneen turvareleen karkitietojen mukaisesti. Lisäksi vaadittiin ehjä ja aktivoitu hätä-seispiiri. Aktivointiin vaadittiin kuitatun hätä-seispiirin ja ehjän turvapiirin lisäksi turvapiirinkuittauspainikkeen aktivointi SafetyReset- (Si7) tuloon, jolloin ohjelmasympolin S\_EStopOut- lähtö aktivoitui.



Kuvio 46. Turvapiirin ohjaus.

SF\_EDM- ohjelmasympolin tarkoituksena oli seurata ohjattujen turvareleiden toimintaa takaisinkytkennän avulla. SF\_EmergencyStop ohjelmasympolin aktivoiduttua SF\_EDM ohjelmasympoli aktivoi turvalähdöt So02 ja So03, jotka ohjasivat ohjauskeskuksen turvareleitä. Releiltä oli tultava takaisinkytkentä 200ms kuluessa tuloihin S\_EDM1 ja S\_EDM2, jolloin kyseinen ohjelmasympoli jäi aktiiviseen tilaan jättäen S\_EDM\_Out- lähdön päälle ja turvapiirin aktiiviseksi. Ohjelmoidun turvapiiriohjelman mukaisesti hätä-seispiirin tai minkä tahansa ovirajakytkinpiirin- tai takaisinkytkentäsignaalin katketessa turvaohjain poisti turvareleohjaukset turvalähdöiltä So02 ja So03. Turvapiirin saattaminen aktiiviseen tilaan vaati kuitattua hätä-seispiiriä ja kaikkien turvaovien sulkemista, jonka jälkeen turvapiiri voitiin kuitata turvapiirinkuittauspainikkeesta.

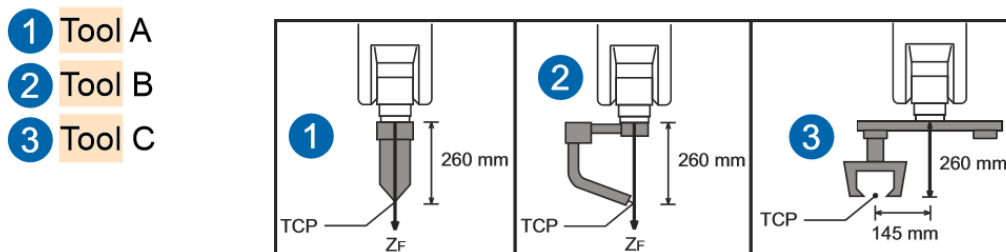
### 10.3 Teollisuusrobottiohjelmointi

Yaskawa- teollisuusrobotin ohjelmointi suoritettiin ohjelmointitehtävistä viimeisenä. Lopulliset pisteenopetukset ja ohjelman hienosäädön pystyi toteuttamaan käyttöönottoaiheessa asiakkaan tuotantotilassa. Ohjelmointi toteutettiin robotin käsiohjaimella ja ohjelmointitapana käytettiin opettamalla ohjelmointia. Robottiohjelmointin työkalu- ja I/O- määrittelyissä sekä ohjelmakäskyjen toteuttamisessa hyödynnettiin Yaskawa- käyttöohjetta.

### 10.3.1 Työkalu-, I/O- ja muuttujamäärittäykset

Teollisuusrobotin ja logiikan välisen Ethernet/IP- kenttäväyläyhteyden muodostamisen jälkeen robotille oli määritettävä uuden työkalun mukainen työkalupiste, nimettävä robotilla käytettävä I/O sekä määritettävä ohjelmassa käytettävät muuttujat.

Robotin työkalupiste oli sidottu oletuksena robotin laippaan, jolloin työkalun käyttö robotissa vaati uuden työkalupisteen määrittämisen. Kuvion 47 esimerkinmukaisesti työkalu määritetään yleensä työkalun kärkeen, jossa kappaletta käsitellään tai työvaiheita suoritetaan. Työkalupiste määritettiin robotin työkalukoordinaatiston X,Y ja Z mukaisesti. Kuvion 47 esimerkissä kohdan 1 ja 2 (Tool A ja Tool B) työkalujen kärjet ovat robotin laipan keskipisteestä kohtisuoraan alaspäin, joka vastaa robotin koordinaatistossa suuntaa Z. Tämä vaati ainoastaan Z- suunnan muuttamisen työkalun pituuden verran. Kuvion 47 esimerkissä kohdan 3 (Tool C) työkalu ei ole työkalulaipan alapuolella, jolloin Z- koordinaatin lisäksi on muutettava sivusuunnan koordinaattia Y.



*In the case of tool A, B*

X	0.000	mm	Rx	0.0000	deg.
Y	0.000	mm	Ry	0.0000	deg.
Z	260.000	mm	Rz	0.0000	deg.

*In the case of tool C*

X	0.000	mm	Rx	0.0000	deg.
Y	145.000	mm	Ry	0.0000	deg.
Z	260.000	mm	Rz	0.0000	deg.

Kuvio 47. Työkalupisteen määrittäminen. (Yaskawa Instructions 2019: User manual)

Tässä opinnäytetyössä pihtitarttuja sijaitsi kohtisuoraan robotin laipan suuntaisesti, jolloin työkalumäärittelyissä oli muutettava Z-koordinaattia työkalupihdin pituuden verran. Pihdin tarkka pituus saatiin mekaniikkakuvan avulla. Työkalupisteen määrittäminen helpotti tarkkojen haku- ja jättöpisteiden opetuksessa. Robottia pystyi ajamaan käsiajolla työkalukoordinaatistossa siten, että tarttujan asentoa pystyi muuttamaan pihdinkärjen pysyessä paikallaan. Robotin liikekäskyjä oli mahdollista suorittaa työkalukoordinaatiston mukaisesti, jolloin lineaariliikettä robotti ajoi kohtisuoraan määritetyn työkalun suuntaisesti.

Robotin omaa I/O:ta käytettiin tarttujan pihdin- ja paininsylinterinohjaukseen. Sylinterien sähköiset ohjaukset ja anturisignaalien fyysiset kytkennät oli toteutettu sähkösuunnittelun tuloksena luodun ohjauspiirikaavion mukaisesti. Kuvio 48 esittää robotille määritetyt tulo- ja lähtösignaalit. Pihtitarttujaa sekä paininsylinteriä ohjattiin 5/2- paineilmaventtiilin avulla. Pihtiä ohjattiin robotin lähdöllä OUT#0001 ja paininta lähdöllä OUT#0002. Pihdin anturisignaalin takaraja aktivoitui tulossa IN#0001 ja eturaja tulossa IN#0002. Painimen anturisignaalin takaraja aktivoitui tulossa IN#0003 ja eturaja tulossa IN#0004. Nimettyjä tulo- ja lähtösignaaleja oli mahdollista käyttää robotiohjelmassa määritettyjen nimien perusteella.

<div>TYÖ</div> <div>DOUT MOVE END</div> <div>YLEIS</div> <div>MUUTTUJAT</div>	YLEISTULOT			
	RYHMÄ	IG#001	6:DEC.	06:HEX.
	IN#0001	#00010	<input type="radio"/>	S1_000_TART_TAKA
	IN#0002	#00011	<input checked="" type="radio"/>	S1_001_TART_ETU
	IN#0003	#00012	<input checked="" type="radio"/>	S1_002_PAININ_TA
	IN#0004	#00013	<input type="radio"/>	S1_003_PAININ_ET
	IN#0005	#00014	<input type="radio"/>	

<div>TYÖ</div> <div>DOUT MOVE END</div> <div>YLEIS</div> <div>MUUTTUJAT</div>	YLEISLÄHDÖT			
	RYHMÄ	OG#001	0:DEC.	00:HEX.
	OUT#0001	#10010	<input type="radio"/>	Y1_100_PIHTI
	OUT#0002	#10011	<input type="radio"/>	Y1_101_PAININ
	OUT#0003	#10012	<input type="radio"/>	
	OUT#0004	#10013	<input type="radio"/>	

Kuvio 48. Teollisuusrobotin I/O-määritykset.



Logiikan ja robotin väliset kättelysignaalit alkoivat robotin tuloissa paikasta IN#0049 alkaen ja lähdöt OUT#0049 alkaen seuraavasti:

- IN#0049 = Lusikat padassa (Hakulupa)
- IN#0050 = Pussitusvalmis (Vientilupa)
- OUT#0049 = Pata tyhjennetty
- OUT#0050 = Lusikat asennettu
- OUT#0051 = Tyhjennetään pataa
- OUT#0052 = Ohjelmasuoritus aloitettu

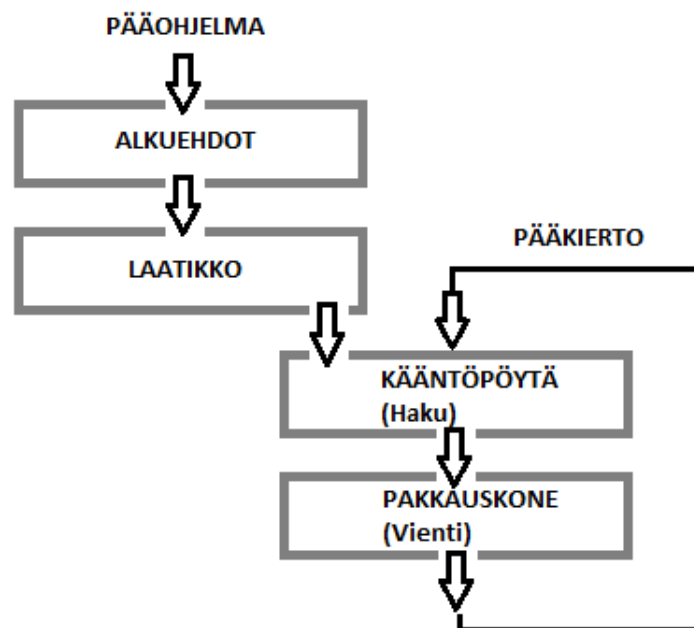
Robotin muuttujiin lisättiin yksi integer- (kokonaisluku) muuttuja. Muuttujaa käytettiin hakukertojen laskennassa. Jokaisen hakukerran jälkeen muuttuja-arvoa lisättiin yhdellä, jolloin robottiohjelmassa tiedettiin, mitkä lokerot on tyhjennetty. Viimeisen hakukerran jälkeen ilmoitettiin logiikalle, että kaikki lusikat on haettu ja integer- arvo asetettiin takaisin arvoon 1, jolloin seuraavan täyden kääntöalustan tyhjennys alkoi jälleen lokerosta 1.

### 10.3.2 Ohjelmarakenne

Robottiohjelmointi alkoi ohjelmarunon suunnitelulla, jossa robotin toiminnot jaettiin pienempiin ohjelmamoduuleihin eli aliohjelmiin. Aliohjelmia oli mahdollisuus käyttää pääohjelmassa (Main) haluamallaan tavalla. Perustoimintona lähtötietojen perusteella oli kääntöpöydän tyhjennys ja pakkauskoneelle vienti. Ohjelmakokonaisuuden hallinta ja mahdolliset häiriötilanteet oli huomioitava ohjelmarakennetta suunniteltaessa.

Kuvio 49 esittää pääohjelmarakennetta (Main). Pääohjelmarakenne sisältää luotuja aliohjelmia, joita robotti suoritti järjestyksessä aliohjelma ja ohjelmarivikerrallaan. Suoritetun rivin jälkeen robotti siirtyy seuraavalle ohjelmariville ja viimeisen suoritetun rivin jälkeen aloittaa ohjelmasuorituksen alusta. Pääohjelma sisälsi aliohjelmat ALKUEHDOT, LAATIKKO, KÄÄNTÖPÖYTÄ ja PAKKAUSKONE. Edellä mainittujen aliohjelmien lisäksi ohjelmassa käytettiin TARTTUJA- ja HAKU 1-12 aliohjelmia. Robotin päästyä pääkiertoon kuvio 49 mukaisesti, se suoritti ainoastaan KÄÄNTÖPÖYTÄ

(haku) ja PAKKAUSKONE (vienti) aliohjelmia. ALKUEHDOT- ja LAATIKKO- ohjelmat suoritettiin ainoastaan, mikäli robottiohjelman suoritus aloitettiin alusta.



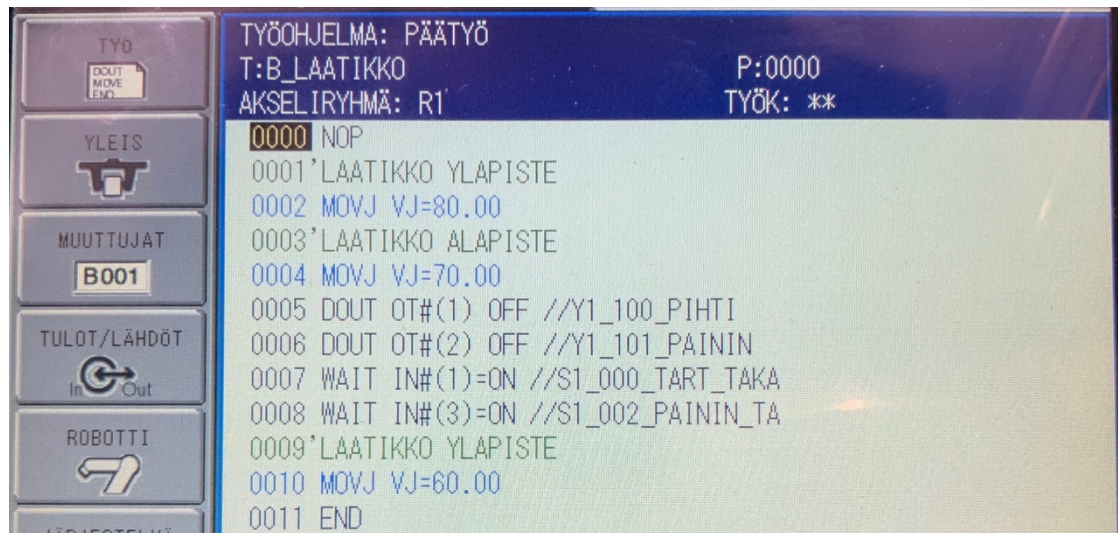
Kuvio 49. Teollisuusrobotin pääohjelmarakenne.

### 10.3.3 Alkuehdot ja laatikointi

Robotti aloitti pääohjelman suorituksen (Main) ALKUEHDOT- aliohjelmasta. Alkuehdoissa nollattiin mahdolliset aktiiviset kättelysignaalit logiikalle. Kääntöpöydän integer- arvoksi asetettiin 1, jolloin robotti aloitti kääntöpöydän tyhjentämisen paikasta 1.

LAATIKKO- aliohjelmassa robotintarttujaan mahdollisesti jääneet kappaleet tyhjennettiin laatikkoon, aina ennen pääohjelmakierron aloittamista. Robotille oli määrätty solun sisälle laatikkopaikka, jonka päällä tarttujantyhjennys suoritettiin. Robotti ajettiin laatikon päälle opetettuihin ylä- ja alapisteisiin MOVJ- liikekäskyllä, nopeuksilla VJ=80 ja VJ=70. Lähtöjen OUT#0001 ja OUT#0002 ohjaukset poistettiin, jotka

poistivat ohjaukset pihtitarttujalta ja painimelta kuvion 50 mukaisesti. Tarttujanpihdin ja paininsylinterin takarajojen aktivoituminen varmistettiin tulojen IN#0001 ja IN#0003 aktivoitumisella.

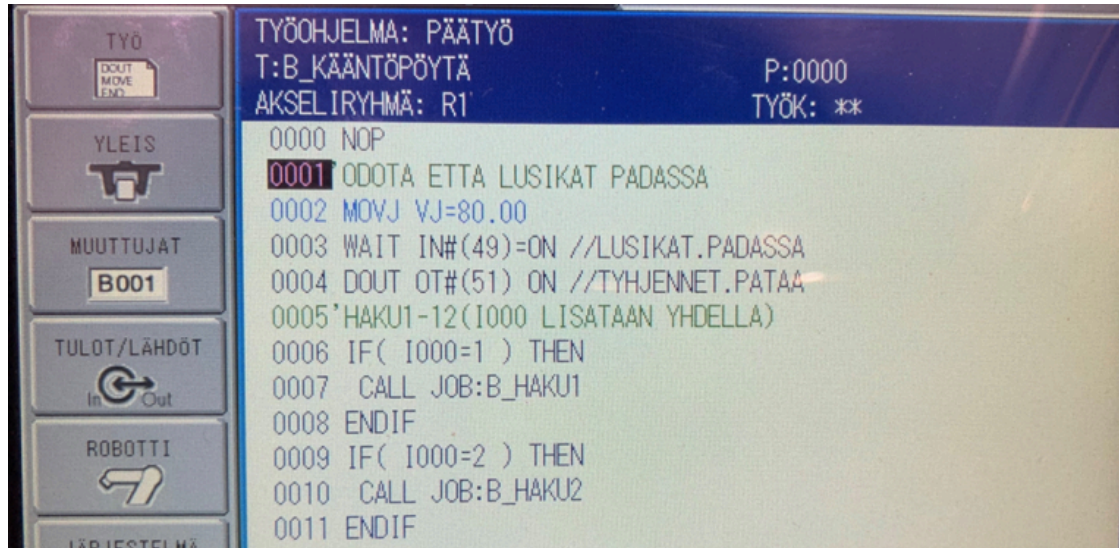


Kuvio 50. Teollisuusrobotin LAATIKKO- aliohjelma.

#### 10.3.4 Kääntöpöydältä haku

ALKUEHDOT ja LAATIKKO- ohjelmien suorittamisen jälkeen voitiin aloittaa pääohjelmakierto, joka aloitettiin KÄÄNTÖPÖYTÄ- aliohjelmasta, joka sisälsi myös aliohjelmat HAKU 1-12. Robotti ajoi odotuspisteeseen kääntöpöydän läheisyyteen MOVJ- liikekäskeillä, jossa robotti jäi odottamaan logiikanantamaa hakulupakäskyä IN#0049 kuvion 51 mukaisesti. Kääntöpöydän käännettyä robotin työalueelle, logiikka antoi robotille luvan suorittaa kääntöpöydän tyhjennys. Alkuasetuksissa määritettiin hakulasurin I000 integer- arvoksi 1, jolloin kuvion 51 mukaisesti ohjelmarivillä 0006 haku suoritettiin aliohjelmasta HAKU 1. Jokaisen hakukerran jälkeen integer- arvoa lisättiin

yhdellä. Seuraavalla hakukerralla integer- arvo oli 2, jolloin lusikat haettiin aliohjelmasta HAKU 2. Viimeisen 12 hakukerran jälkeen integer- arvoksi asetettiin jälleen 1 ja robotti ilmoitti logiikalle tyhjennetystä kääntöalustasta.



Kuvio 51. Teollisuusrobotin KÄÄNTÖPÖYTÄ- aliohjelma.

HAKU 1-12- aliohjelma suorittaminen toteutettiin IF- ehtolauseen mukaisesti. Kuvion 51 ohjelmarivillä 0006 oleva IF- lauseessa tarkasteltiin integer- arvoa I000. Mikäli integer arvo I000 = 1, tällöin siirryttiin suorittamaan aliohjelmaa HAKU 1. Jos integer- arvo ei olisi ollut 1, tällöin oltaisi siirrytty ohjelmariville 0009 ja tarkasteltu onko integer- arvo 2. Integer- arvon ollessa 2 suoritettiin aliohjelma HAKU 2. Mikäli integer- arvo ei olisi ollut 2 oltaisiin tarkasteltu integer- arvoa 3 ja jne.

Haku 1-12 aliohjelmat olivat keskenään identtisiä, ainoastaan jokaisen ohjelman pisteet oli opetettu kullekin haulle omaan lokeroonsa. Ohjelma sisälsi yläpisteen haku- paikan päällä ja alapisteen, josta lusikkapino oli mahdollista poimia. Alapisteessä tarttuja suljettiin, jonka jälkeen robotti nousi takaisin yläpisteeseen ja oli valmis siirtymään PAKKAUSKONE- aliohjelmaan.

### 10.3.5 Pakkauskoneelle vienti

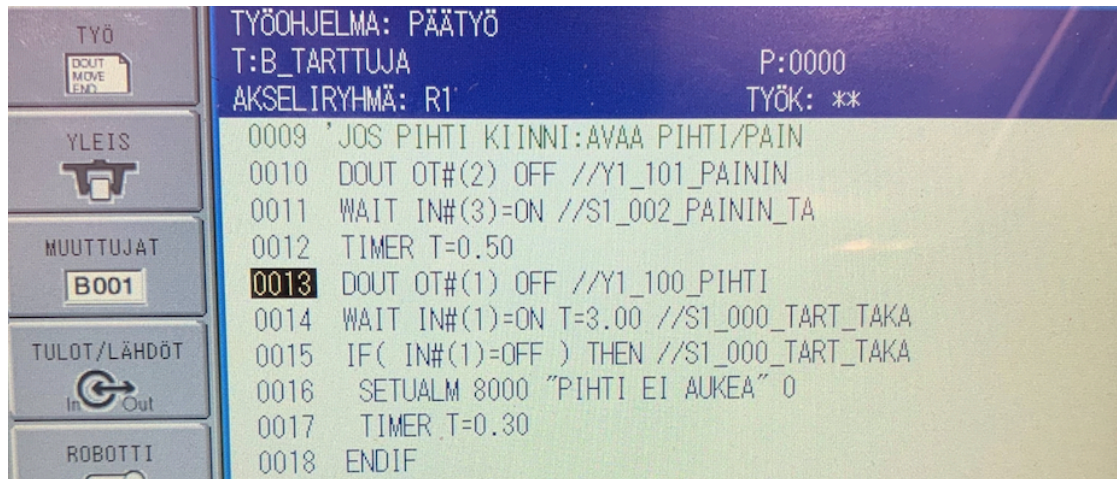
PAKKAUSKONE- aliohjelmassa varmistettiin IF- lausekkeella logiikan antama vienti-lupa pakkauskoneelle tulosta IN#0050. Mikäli pakkauskoneessa ilmeni häiriö, vienti-lupaa pakkauskoneelle ei ollut ja kääntöpöydältä noudetut lusikat ajettiin laatikkoon. Pakkauskoneen häiriö ei saanut estää ruiskuvalukoneen toimintaa, jonka takia kääntöpöytä oli tyhjennettävä pakkauskoneessa ilmenneestä häiriöstä huolimatta.

Vientiluvan IN#0050 ollessa aktiivinen, lusikkapinon vienti sallittiin pakkauskoneelle. Lusikkapino jätettiin usean välipisteen kautta pakkauskoneen hihnalle vaakatasossa. Tarttuoja avattiin ja robotti ajettiin pois pakkauskoneen hihnan läheisyydestä. Robotti asetti lähdön OUT#0050 aktiiviseksi, jolloin logiikka tiesi robotin jättäneen lusikkapinon pakkauskoneen hihnalle ja antoi pakkauskoneelle käynnistys-signaalin. Pakkauskone ajoi hihnaa yhden lusikkapinon verran eteenpäin. Robotti jatkoi ohjelmakiertoaan hakemalla seuraavan lusikkapinon kääntöpöydältä.

### 10.3.6 Tarttujan ohjaus ja häiriöngenerointi

Pihtitarttujan avaaminen ja sulkeminen toteutettiin TARTTUJA- aliohjelmassa. Pihdin käyttöä tarvittiin jokaisessa 12 HAKU- aliohjelmassa, LAATIKKO- ja PAKKAUSKONE- aliohjelmissa. Ohjelman selkeyden vuoksi tarttujalle tehtiin oma aliohjelma, jota kutsuttiin aina, kun pihtitarttujaa oli tarve avata tai sulkea. Tarttujanpihdin avaus- ja sulkemishäiriöstä aktivoitiin häiriöilmoitus. Häiriö aktivoitiin mikäli pihti ei päässyt suljettaessa eturajalleen (IN#0002) tai avatessa takarajalleen (IN#0001).

Pihti ohjelman rakenne toteutettiin IF- lauseen sisälle, jossa pihdin ollessa kiinni, se avattiin ja pihdin ollessa auki, se suljettiin. Kuvio 52 esittää tarttujan avaamisohjelmaa. Paininsylinterin ohjaus OUT#0002 poistettiin ja odotettiin, että painimen takaraja IN#0003 aktivoituu. Odotettiin 0.5 sekuntia ja poistettiin tarttujapihdin ohjaus OUT#0001. Pihdin takarajan IN#0001 oli aktivoiduttava 3 sekunnin kuluessa, muuten aktivoitiin häiriöilmoitus "PIHTI EI AUKEA". Häiriön ilmaantuessa robottiohjelman suoritus lopetettiin ja robotin käynnistäminen oli mahdollista häiriönpoistamisen jälkeen. Pihdin sulkemisen ohjelmarakenne toteutettiin vastaavalla ohjelmarakenteella.



Kuvio 52. Teollisuusrobotin TARTTUJA- aliohjelma.

### 10.3.7 Ohjelman suorituksen aloittaminen

Puukuitulikkeen automaatin käyttöpaneelina toimi robotin operointipaneeli. Automaatin käynnistäminen automaattiajolle aloitettiin ohjausjännitteiden kytkennällä, joka tapahtui solun sisällä olevan sähkökeskuksen ohjausjännitepainikkeesta. Varmistettiin, että kääntöalusta on tyhjä ennen ajon aloittamista ja robotti lähellä sen kotipistettä. Ruiskuvalukoneen ja solun ympäröivän suoja-aidan turvaovet suljettiin ja kuittattiin suoja-aidan vieressä olevasta kuittauspainikkeesta. Ruiskuvalukone ja pakkauskone asetettiin automaattiajolle. Robotin operointipaneelistä valittiin suoritettavaksi ohjelmaksi lusikanpakkauksen pääohjelma (Main). Robottiservot aktivoitiin ja ohjelman suoritus aloitettiin robotin operointipaneelin START- painikkeesta, jolloin robotti aloitti työsuorituksensa pääohjelman mukaisesti. Robotin START- komento mahdollisti myös logiikkasekvenssien toiminnan aloittamisen.

## 11 Käyttöönotto

### 11.1 Toiminnan testaus

Ennen puukuitulusikka automaatin käyttöönottoa ja asiakkaalle toimitusta automaatin mekaaniset- ja sähköiset rakenteet kokoonpantiin toimeksiantajan tuotantotiloissa. Automaatiokokonaisuuden sisältämät kääntöpöytä, leikkain ja robotti tarttujiin kytkettiin pneumaattisesti ja sähköisesti automaatiojärjestelmään. Niiden mekaaniset toiminnot, logiikka- ja robottiohjelmarakenteet testattiin. Automaattikokonaisuuden toiminta oli mahdollisuus todentaa ruiskuvalukonetta ja pakkauskonetta lukuun ottamatta. Ruiskuvalukoneen ja pakkauskoneen puuttuessa ei ollut mahdollisuutta suorittaa normaalisti suoritettavaa FAT- (Factory Acceptance Test) hyväksyntääjao. Laitteen lopullinen hyväksyntä suoritettiin asiakkaan tuotantotiloissa suoritettavalla SAT- (Site Acceptance Test) hyväksyntäajolla.

Automaatin todennetun toimivuuden jälkeen automaattikokonaisuus siirrettiin asiakkaan tuotantotiloihin. Teollisuusrobotti, leikkain, kääntöpöytä, sähkökeskus, pakkauskone ja robottiohjain asennettiin mekaniikkasuunnittelun laatiman layout-kuvan mukaisesti ruiskuvalukoneen yhteyteen. Automaatin ympärille pyytettiin suoja-aita, johon asennettiin hätä-seispainikkeet. Toimilaitteet kytkettiin sähköisesti ja pneumaattisesti sähkö- ja mekaniikkasuunnitelman mukaisesti. Automaatin layout-kuva esitetty opinnäytetyön liitteessä 1.

Liitäntärajapinnat logiikalta ruiskuvalukoneeseen ja pakkauskoneeseen suunniteltiin yhteistyössä kyseisten laitteiden laitetoimittajien kanssa. Suunnitelmat toteutettiin hyvissä ajoin ennen laitteiden toimitusta, jolloin kukin osasi varata tarvittavat I/O:t laitteiden väliseen kommunikaatioon. Tarvittavat välireleet lisättiin puukuituautomaatin sähkökeskukseen. I/O-liityntärajapintojen kytkennät suoritti kukin laitetoimittaja toimittamiinsa laitteisiin. Laitteidenväliset I/O-rajapinnat testattiin puolin ja toisin, signaali kerrallaan.

Teollisuusrobotille oli opetettava lopulliset paikkapisteet. Kaikki 12 hakupistettä, lusikoiden jättöpisteet ja laatikkopisteet opetettiin tarvittavine välipisteineen. Robotiohjelmalla testattiin käsiajolla aliohjelma kerrallaan ennen automaattiajon aloittamista. Ruiskuvalukoneen laitetoimittajan edustaja teki lineaaribotin ohjelmat, joilla lusikkaviuhkat ajettiin leikkaimen kautta kääntöalustalle. Automaattikonaisuuden testaus aloitettiin lusikkaviuhka kerrallaan tarkistaen lineaaribotin leikkain- ja kääntöalustan paikkapisteet sekä ruiskuvalukoneen ja logiikanvälisten kättelytietojen toimivuus. Lopulta lineaaribotti ajoi kääntöalustan täyteen ja suoritettiin kääntöalustan kääntö. Testausta jatkettiin teollisuusrobotin puolelle. Teollisuusrobotti poimi kääntöalustalta lusikkapinon kerrallaan vieden pinon pakkauskoneen hihnalle. Testauksessa oli tarkoitus havainnoida kaikkien kahdentoista opetetun hakupisteen paikoitus. Lusikkapino, se ei saanut osua hakulokeron reunoihin robotilla haettaessa. Pakkauskoneenhihnalle jättövaiheessa pihtitarttuja ei saanut osua jätettyyn lusikkapinoon. Testausvaiheessa simuloitiin myös mahdollisia häiriötilanteita. Pakkauskone asetettiin manuaaliajolle, jolla simuloitiin mahdollista pakkauskoneen häiriötilannetta. Robotilla ei ollut lupaa viedä lusikkapinoja pakkauskoneelle, jolloin robotti ajoi lusikat pakkauskoneen sijasta laatikkoon. Robotin hakulaskurin toimintaa testattiin siten, että robotin oli aloitettava aina hakupaikasta 1 ja siirryttävä ensimmäisen haetun lusikkapinon jälkeen hakupaikkaan 2 ja jne. Viimeisen viedyn lusikkapinon jälkeen robotin oli jäätävä odottamaan kääntöalustan kääntymistä ja aloitettava tyhjennys hakupaikasta 1.

## 11.2 Automaatin testaukset ja hyväksyntäajo

Ennen virallista SAT- hyväksyntäajoa automaatille oli suoritettava toimeksiantajan toimintajärjestelmän mukaiset testaukset. Sähköisten kytkentöjen jälkeen suoritettiin sähköiset koestusmittaukset, joka sisälsi eristysvastus-, jäännösjännite- ja suoja-johtimen jatkuvuusmittaukset. Lähtötarkastuslomakkeen sisältämien tarkastuskohteiden mukaisesti varmistettiin muun muassa turvasuojien- ja hätä-seispainikkeiden toiminta ja tarvittavien varoituskylttien asennus. Käyttäjäkoulutuksessa operaattorit koulutettiin turvalliseen koneenkäyttöön. Koulutuksen tärkeimpinä tavoitteina oli käydä läpi koneen käyttöön liittyvät alkuvalmistelut, automaattiajo ja robotin käsiajo.



SAT- hyväksyntäajan tavoitteet oli määritelty asiakkaan ja laitetoimittajan välisissä sopimuksissa. SAT- hyväksyntäajoissa varmistettiin automaatin toiminta ja sille luvattu jaksonaikataavoite. Hyväksytysti suoritettun SAT- hyväksyntäajan jälkeen asiakkaalle toimitettiin päivitetyt sähkö- ja mekaniikkakuvat. Konedirektiivien täyttymisestä asiakkaalle toimitettiin vaatimuksenmukaisuusvakuutus, jonka jälkeen automaattiin asennettiin CE- kyltti.

## 12 Tutkimustulokset ja päätelmät

### 12.1 Tuotannon automatisointi

Automaatti vaatii toimiakseen mekaanisten ratkaisuiden lisäksi oikeat toimilaitteet, ohjausjärjestelmät ja ohjelmat, joiden avulla liikekäskyjen suorittaminen automaattisesti on mahdollista. Jokainen automaatiolaite on yksilö ja sen toimilaitteet ja ohjausjärjestelmät on valittava tapauskohtaisesti. Valitut toimilaitteet on liitettävä osaksi ohjausjärjestelmää, jolloin ohjausjärjestelmä kykenee ohjaamaan toimilaitteita luodun sovellusohjelman mukaisesti. Ohjausjärjestelmän ohjaamat työvaiheet muodostavat lopulta toimintokokonaisuuden, jolla tuotannon toiminnot suoritetaan automaattisesti. Tällöin voidaan puhua tuotannon automatisoinnista.

Tämän opinnäytetyön sähkösuunnittelun ja mekaniikkasuunnittelun benchmarking-haastatteluiden tuloksena kerätyn aineiston perusteella ohjausjärjestelmän ja toimilaitteiden valinta saatiin rajattua. Lopulliset ohjausjärjestelmä- ja toimilaitteevalinnat suoritettiin hyödyntäen projektin lähtötietoja ja laitemanuaaleja. Ohjausjärjestelmän ja toimilaitteiden liittäminen toisiinsa toteutettiin laitteiden tukemaa kenttäväyläprotokolla ja fyysistä I/O:ta hyödyntäen. Tämä mahdollisti tiedonsiirron ohjausjärjestelmän ja toimilaitteiden välillä. Ohjelmointivaiheessa ohjausrutiinit toteutettiin logiikan laiteoppaan mukaisesti.

Oikeiden ohjausjärjestelmä- ja toimilaittevalintojen, luotujen kenttäväylärajapintojen ja ohjelmoidun sovellusohjelman yhteistuloksena saatiin aikaan automaattisesti toimiva kokonaisuus. Ohjausjärjestelmänä toimiva ohjelmoitava logiikka ohjaa toimintoja työvaihe kerrallaan. Toimilaitteena toimiva teollisuusrobotti suorittaa lusikoiden käsittelyä oman sovellusohjelmansa mukaisesti. Turvaohjain valvoo turvakomponenttien tilatietoja sovellusohjelman mukaisesti.

Tämän opinnäytetyön tuloksena puukuitulusikan käsittely- ja pakkaustoimenpiteet saatiin automatisoitua. Ruiskuvalukoneen ruiskuvalamat puukuitulusikat saatiin käsiteltyä ja pakattua siten, että automaatin loppupäästä saatiin valmiiksi pakattuja aterinpusseja.

## 12.2 Tuotannolliset hyödyt

Tuotannon automatisoinnilla tähdätään yleensä tuotannon tehostamiseen ja/tai operaattoreille raskaiden työvaiheiden poistamiseen. Tuotannon automatisoinnin tavoitteena tässä opinnäytetyössä oli tuotannon kaksinkertaistaminen ja operaattoreiden työvaiheiden poistaminen.

Mekaanisten toimilaitteiden ja nopeiden ohjausjärjestelmien ansiosta työvaiheiden suorittamiseen käytettävää aikaa saatiin tehostettua. Automatisoinnin tuloksena puukuitulusikoiden käsittely pystyttiin toteuttamaan ruiskuvalukoneen jaksonaikaa nopeammin. Tämä tarkoitti, että ruiskuvalukoneen maksimaallinen tuotantotehokkuus pystyttiin hyödyntämään hidastamatta tuotantoa käsittely- ja pakkausvaiheessa. Loppuasiakkaan kertoman perusteella työvuorossa tuotettu tuotantomäärä on saatu automatisoinnin tuloksena ylittämään tuotantotavoitteet.

Puukuitulusikoiden käsittelyn ja pakkauksen toteuttaminen käsin vaati useamman operaattorin työpanoksen. Mekaanisten- ja ohjelmallisten ratkaisujen sekä teollisuusrobotilla toteutetun käsittelyn tuloksena operaattoreiden työvaiheet saatiin poistettua kokonaan. Automatisoinnin tuloksena loppuasiakas pystyi keskittämään

operaattoreiden työpanoksen automaattilla pakattujen lusikkapussien jatkokäsittelyyn.

Tuotantotavoitteiden lisäksi, tämän opinnäytetyön tuloksena toteutetulla puukuitulusikka automaattilla pystyttiin vaikuttamaan maailmanlaajuiseen ongelmaan. Automaattilla käsiteltävät puukuitulusikat tulevat korvaamaan perinteiset muovista valmistetut kertakäyttölusikat. Lusikkamassa on valmistettu ympäristöystävällisiä materiaaleja hyödyntäen. Loppuasiakkaan kertoman perusteella puukuitulusikat ovat pesunkestävät ja uudelleen käytettävissä. Poltettaessa lusikat palavat täysin puhtaasti.

### 12.3 Päätelmät

Tämän opinnäytetyön tutkimustuloksista voidaan todeta, että työlle asetetut tavoitteet on saavutettu. Työhön hankitun lähdemateriaalin perusteella ohjausjärjestelmän ja teollisuusrobotin mahdollistaa myös nykyisen tuotannon jatkokehityksen. Työssä käytetyn automaatiojärjestelmän EtherCAT- kenttäväylään on mahdollista lisätä väyläyksiköitä, minkä kautta liitäntärajapinnat uusille toimilaitteille voidaan toteuttaa. 6- akselinen teollisuusrobotti antaa vapauden toteuttaa erilaisia kappaleenkäsittelyvariaatioita, mikäli tulevaisuudessa tuotantomenetelmää haluttaisiin kehittää.

Tämänhetkinen tuotanto mahdollistaa yhden lusikkapinon pussituksen. Automaattiin tehtävillä pienillä mekaanisilla ja ohjelmallisilla muutostöillä mahdollistettaisiin useamman lusikkapinon pakkaaminen samaan pussiin. Tällöin kuluttajalle saataisiin lusikkamäärällisesti erikokoisia tuotepusseja. Tämä vaatisi lusikkapinon pannoituksen, joka olisi mahdollista lisätä nykyiseen automaatiojärjestelmään. Lusikkapinot ajettaisi pannoituksen kautta pakkauskoneen hihnalle. Pakkauskoneelle ei annettaisi käynnistyslupaa, kuin vasta toisen jätetyn lusikkapinon jälkeen. Robottiohjelmaan tehtäisiin omat pääohjelmat sen mukaisesti, minkä kokoista tuotepakkausta haluttaisiin ajaa.

## 13 Pohdinta

Tässä opinnäytetyössä tavoitteena oli automatisoida puukuitulusikan käsittely- ja pakkaustoimenpiteet. Automatisoinnilla pyrittiin kaksinkertaistamaan nykyinen tuotantomäärä ja poistamaan operaattoreiden käsikokoonpanon työvaiheet. Opinnäytetyön tavoitteita käsiteltiin sähkösuunnittelun näkökulmasta, jossa aihe rajattiin ohjausjärjestelmä- ja toimilaittevalintoihin, toimilaitteiden liityntärajapintoihin, ohjelmointiin, käyttöönottoon. Tutkimuksen tutkimusaineisto kerättiin teemahaastatteluilla ja benchmarking-menetelmää hyödyntäen. Benchmarking- ja teemahaastattelut kohdennettiin opinnäytetyön tilaajayrityksen sähkö- ja mekaniikkasuunnitteluun. Haastatteluilla kerättyjä tutkimusaineistoa pitäisin luotettavana, koska mekaniikkasuunnittelijan haastattelu kohdennettiin suunnittelijaan, joka suunnitteli puukuitulusikka automaatin mekaaniset ratkaisut. Sähkösuunnittelijoiden benchmarking-haastattelut kohdennettiin useisiin sähkösuunnittelijoihin, jolloin tutkimusaineiston luotettavuutta pystyi arvioimaan tarkemmin.

Työn lähdemateriaalia kerättiin automaatin sisältävien ohjausjärjestelmien ja toimilaitteiden sekä työssä käytettävien toimintamenetelmien mukaisesti. Tiedonhankinnan haasteena oli löytää lähdemateriaalia uusimmista ohjausjärjestelmistä ja kenttäväyläratkaisuista. Kyseisten toimilaitteiden nopea kehitys tuo omat haasteensa ajankohtaisimpien tietolähteiden löytämiseen. Lisäksi ohjelmointiin liittyvän lähdemateriaalin hankinta osoittautui osittain haastavaksi. Tämä voi johtua siitä, että ohjelmoinnin toteuttamisessa on merkki- ja mallikohtaisia eroavaisuuksia, jolloin on mahdoton esittää aukotonta kokonaisuutta ohjelmoinnin toteutustavasta. Ajankohtaisimmat tietolähteet löytyivät vieraskielisistä materiaaleista, joita työssä hyödynnettiin etenkin kenttäväylien osalta. Työn toteutuksessa hyödynnetyt kenttäväylärajapintoihin ja ohjelmointiin liittyvät tietolähteet pohjautuivat pelkästään vieraskielisiin tietolähteisiin.

Kokonaisuutena työn toteutus käytettyjen tiedonhakumenetelmien ja hankitun lähdematerian pohjalta onnistui hyvin. Valituilla ohjausjärjestelmillä pystyi toteutta-

maan tavoitteiden mukaiset toimilaite- ja turvaohjaukset. Ohjausjärjestelmien käyttöönotto sekä ohjelmointi ilman suurempia ongelmia. Automaatin käyttöönotto vaati yhteistyötä ruiskuvalukoneen ja pakkauskoneen laitetoimittajien yhteyshenkilöiden kanssa. Yhteiset aikataulut laiterajapintojen testaamiseksi saatiin sovittua ja automaattikokonaisuuden testaus saatiin suoritettua aikataulussa.

Tämän opinnäytetyön tuloksia toimeksiantaja voi hyödyntää seuraavissa puukuitu haarukka- ja veitsiautomaattien automatisoinnissa. Seuraavien automaattien toteutus voidaan toteuttaa lähes täysin opinnäytetyöllä saavutettujen tulosten perusteella. Opinnäytetyön tuloksia voidaan jatkossa hyödyntää myös Omronin NX1P- logiikkamallien I/O käyttöönottoon sekä Ethernet/IP- ja EtherCAT- kenttäväyläyhteyksien muodostamiseen. Nykyaikaisten automaatiojärjestelmien ja kenttäväyläliityntärajapintojen muodostamisesta ei lähdeaineiston hankinnan yhteydessä löytynyt aikaisempia tutkimuksia.

Tämän opinnäytetyön tulosten perusteella loppuasiakkaan toiveet ja työlle asetetut tavoitteet saatiin toteutettua. Opinnäytetyön aiheesta teki erittäin mielenkiintoisen ajankohtainen tuote, jolla korvataan perinteiset kertakäyttömuoviaterimet. Loppuasiakas oli tyytyväinen saavutettuihin tuotantotavoitteisiin ja varmatoimiseen kokonaisuuteen. Työtä tullaan tarvittaessa kehittämään pidempiaikaisen tuotantokokeusten- ja loppuasiakkaan toiveiden mukaisesti.

## Lähteet

Automation Software: Sysmac Studio for machine creators. 2019. Laiteopas. Omron Corporation. Viitattu 27.12.2019. [https://assets.omron.eu/downloads/datasheet/en/v9/i181e\\_sysmac-se2\\_sysmac\\_studio\\_datasheet\\_en.pdf](https://assets.omron.eu/downloads/datasheet/en/v9/i181e_sysmac-se2_sysmac_studio_datasheet_en.pdf)

Bélander-Barette, M. 2015. What does collaborative robot mean?. Blogikirjoitus. Robotiq. Viitattu 06.11.2019. <https://blog.robotiq.com/what-does-collaborative-robot-mean>

EtherCAT – the Ethernet Fieldbus. 2012. EtherCAT Technology Group. Viitattu 11.12.2019. [http://www.ethercat.org/pdf/ethercat\\_e.pdf](http://www.ethercat.org/pdf/ethercat_e.pdf)

EtherNet/IP EDS Update. 2012. Blogikirjoitus. Viitattu 08.12.2019. <https://www.rtautomation.com/rtas-blog/ethernet-ip-eds-update/>

Fonselius, J., Pekkola, K., Selosmaa, S., Ström, M. & Välimaa, T. 1996. Automaatiolaitteet. Painos 2. Helsinki: Oy Edita Ab

Function Block Diagram (FBD) Programming Tutorial. 2018. PLC Programming & Automatin Online. Viitattu 06.11.2019. <https://www.plc-academy.com/category/plc-programming/>

Goble, W. 2003. Learn to trust safety PLCs. Control Design for Machine Builders. Putman Media. Viitattu 06.11.2019. <https://www.controldesign.com/articles/2003/-249/>

Handling & General Applications with the GP-series: Motoman GP12. 2019. Laiteopas. Viitattu 27.12.2019. <https://www.yaskawa.eu.com/index.php?elD=dump-File&t=f&f=17298-&token=849af23b4960e1efce2845fb115a505957c2f115>

How Does a Case Packing Machine Work. N.d. EconoCorp. Viitattu 12.12.2019. <https://www.econocorp.com/how-does-a-case-packing-machine-work/>

Hooper, R. N.d. Industrial Robots. Learn About Robots. Viitattu 06.11.2019 <https://www.learnaboutrobots.com/industrial.htm>

Hulkkonen, V. 2008. Pneumatiikka: Venttiilit. Fluid Klinikka. Viitattu 11.12.2019. <https://asiakas.kotisivukone.com/files/fluidfinland.kotisivukone.com/FluidKlinikat/16.pneumatiikka-venttiilit.pdf>

Järvinen, P. 2008. Uusi muovitieto. Porvoo: Muovifakta Oy

Järvinen, P. 2017. Muovit ja muovituotteiden valmistus. Porvoo: Muovifakta Oy

Kananen, J. 2010. Opinnäytetyön kirjoittamisen käytännön opas. Jyväskylä: Jyväskylän ammattikorkeakoulu.

Kananen, J. 2012. Kehittämistutkimus opinnäytetyönä. Jyväskylä: Jyväskylän ammattikorkeakoulu.

Kananen, J. 2015. Opinnäytetyön kirjoittajan opas. Jyväskylä: Jyväskylän ammattikorkeakoulu.

Keinänen, T., Kärkkäinen, P., Metso, T. & Putkonen, K. 2001. Logiikat ja ohjausjärjestelmät. Helsinki: WSOY Oppimateriaalit Oy.

Kuivanen, R. 1999. Robotiikka. Tampere: Suomen robotiikkayhdistys ry.

Ladder Logic Tutorial. 2019. Blogikirjoitus. Viitattu 17.11.2019. <https://www.plcacademy.com/category/plc-programming/>

Malm, T., Venho-Ahonen, O. & Vanhala, M. 2010. Automaatiouusintojen turvallisuus konejärjestelmissä. Tampere: VTT Expert Service. Viitattu 02.12.2019. <https://www.vtt.fi/inf/julkaisut/muut/2010/VTT-R-04369-10.pdf>

NX1P series machine controller. N.d. Laiteopas. Viitattu 27.12.2019. [https://assets.omron.eu/downloads/datasheet/en/v4/i179e\\_nx1p\\_nx1-series\\_machine\\_controller\\_datasheet\\_en.pdf](https://assets.omron.eu/downloads/datasheet/en/v4/i179e_nx1p_nx1-series_machine_controller_datasheet_en.pdf)

Omron Corporation. 2019. Safety control unit. Laiteopas. Viitattu 17.11.2019. [https://assets.omron.eu/downloads/manual/en/v5/z930\\_nx\\_series\\_safety\\_control\\_unit\\_users\\_manual\\_en.pdf](https://assets.omron.eu/downloads/manual/en/v5/z930_nx_series_safety_control_unit_users_manual_en.pdf)

Operation Manual: Sysmac Studio Version 1. 2019. Laiteopas. Omron Corporation. Viitattu 29.12.2019. [https://assets.omron.eu/downloads/manual/en/v11/w504\\_sysmac\\_studio\\_operation\\_manual\\_en.pdf](https://assets.omron.eu/downloads/manual/en/v11/w504_sysmac_studio_operation_manual_en.pdf)

Parlamentti sinetöi kertakäyttömuovi kiellon vuoteen 2021 mennessä. 2019. Viitattu 11.11.2019. <https://www.europarl.europa.eu/news/fi/pressroom/20190321IPR3211/parlamentti-sinetoi-kertakayttomuovin-kiellon-vuoteen-2021-menessa>

PLC Memory. 2016. Viitattu 28.11.2019. <https://automationprimer.com/2016/08/28/plc-memory/>

Principle of operation. 2016. Beckhoff Automation. Viitattu 11.12.2019. <https://www.beckhoff.com/english.asp?ethercat/aufbau.htm?id=35572043349>

Pyyskänen, S. 2007. Teollisuuden laiteverkot. Helsinki: Suomen Automaatioseura ry.

Robot Controller: User Manual. 2017. Laiteopas. Yaskawa Robotics.

Keinänen, T., Kärkkäinen, P., Lähetkangas, M. & Sumujärvi, M. 2007. Automaatiojärjestelmien logiikat ja ohjaustekniikat. Helsinki: WSOY oppimateriaalit.

Kippo, A & Tikka, A. 2008. Automaatiotekniikan perusteet. Helsinki: Edita.

Shikany, A. 2014. Collaborative Robots End User Industry Insight. Robotic Industries Association. Viitattu 06.11.2019. [https://www.robotics.org/userassets/riauploads/file/RIA\\_Collaborative\\_Robots\\_White\\_Paper\\_October\\_2014.pdf](https://www.robotics.org/userassets/riauploads/file/RIA_Collaborative_Robots_White_Paper_October_2014.pdf)

Siirilä, T & Tytykoski, K. 2016. Koneturvallisuuden käsikirja. Keuruu: Inspecta.

Technology. 2019. EtherCAT. Viitattu 22.11.2019. <https://www.ethercat.org/en/technology.html#1>

Technology overview series: Ethernet/IP. 2016. ODVA Inc. PUB00138R6. Viitattu 25.11.2019. [https://www.odva.org/Portals/0/Library/Publications\\_Numbered/PUB00138R6\\_Tech-Series-EtherNetIP.pdf](https://www.odva.org/Portals/0/Library/Publications_Numbered/PUB00138R6_Tech-Series-EtherNetIP.pdf)

Tuotteet tehdas- ja prosessiautomaatioon. 2018. Perustuoteluettelo. Viitattu 11.12.2019. [https://www.festo.com/net/SupportPortal/Files/585501/Distributor-Catalogue\\_2018\\_5Edition\\_FI\\_low.pdf](https://www.festo.com/net/SupportPortal/Files/585501/Distributor-Catalogue_2018_5Edition_FI_low.pdf)



## Liitteet

Liite 1. Puukuitulusikka-automaatin layout-kuva

