

Jarkko Puska

Kolmeakselisen CNC-jyrsimen suunnittelu ja rakennus

Opinnäytetyö

Kevät 2019

SeAMK Tekniikka

Koneautomaatio



SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

SEINÄJOEN AMMATTIKORKEAKOULU

Opinnäytetyön tiivistelmä

Koulutusyksikkö: SeAMK Tekniikka

Tutkinto-ohjelma: Automaatiotekniikan koulutusohjelma

Suuntautumisvaihtoehto: Koneautomaatio

Tekijä: Jarkko Puska

Työn nimi: Kolmeakselisen CNC-jyrsimen suunnittelu ja rakennus

Ohjaaja: Jarkko Pakkanen

Vuosi: 2019

Sivumäärä: 60

Liitteiden lukumäärä: 5

Työn tarkoituksena oli suunnitella ja valmistaa kolmeakselinen jyrsinkone. Työn edetessä tavoitteena oli myös saada kokemusta konesuunnittelun eri aihealueilta.

Työn alussa kerrotaan NC-tekniikasta yleisesti sekä lastuavasta työstöstä. Työssä määriteltiin myös jyrsimen käyttötarkoitus, jonka tietojen pohjalta pystyttiin aloittamaan koneen mekaniikka-, sähkö- ja automaatiosuunnittelu.

Mekaniikkasuunnittelu suoritettiin Autodesk Fusion 360 -ohjelmistolla. Sähkösuunnittelussa käsiteltiin koneen riskianalyysiä sekä komponentteja. Sähkökuvat piirrettiin EPLAN-sähkösuunnitteluohjelmistolla. Automaatiosuunnittelussa käytettiin kahta erilaista ohjausta, Arduino-pohjaista Grbl-ohjausta ja Beckhoffin TwinCat-ohjausta. Työssä käytiin läpi myös erilaisia ohjelmointikieliä sekä ohjelmoinnissa käytettäviä yleisiä standardeja.

Grbl-ohjauksella toteutettiin lineaariliikkeet kuularuuveilla ja askelmoottoreilla, mutta työn loppuvaiheessa koneen ohjaus vaihdettiin Beckhoffin TwinCat-ohjaukseen. Ohjauksen vaihdon takia koneeseen tehtiin sähkömuutoksia ja lineaariliikkeiden moottorit muutettiin servokäyttöiseksi.

CNC-jyrsin tehtiin kahdella erilaisella ohjauksella ja molemmilla ohjauksilla saatiin lopputuloksena toimiva CNC-jyrsin, jolla pystyttiin jyrsimään puuta, alumiinia ja terästä.

Avainsanat: CNC, NC, PLC, CAD, CAM, EPLAN, Sistema, TwinCat, Grbl

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Thesis abstract

Faculty: School of Technology

Degree programme: Automation Engineering

Specialisation: Machine Automation

Author: Jarkko Puska

Title of thesis: Design and Construction of a 3-axis CNC Mill

Supervisor: Jarkko Pakkanen

Year:2019

Number of pages:60

Number of appendices: 5

The purpose of the thesis was to design and manufacture a three axis milling machine. As the work progressed, the goal was to gain experience in various areas of machine engineering.

In the beginning of the work NC techniques were studied in general and attention was also paid to the milling process. The purpose of the milling machine was defined, so that mechanical, electrical and automation engineering could be started based on the defined values.

Mechanical engineering was performed with Autodesk Fusion 360 software. The machine risk analysis and electrical components were studied in the electrical engineering section. Electrical drawings were made with the EPLAN design software. In the automation engineering section, two different machine controls were handled. The first one was an Arduino based Grbl control and the second one was a Beckhoff TwinCat control. Various programming languages and general standards for programming were also discussed.

With the Grbl control, linear movements were made with ball screws and stepper motors. In the final stage, the Grbl control was changed to the TwinCat control, so the motors were also updated to servomotors. Because of this it was necessary to make two different electrical- and automation designs.

The CNC milling machine was made with two different controls. Both control systems produced a working CNC Mill, capable of cutting wood, aluminum and steel.

Keywords: CNC, NC, PLC, CAD, CAM, EPLAN, Sistema, TwinCat, Grbl

SISÄLTÖ

Opinnäytetyön tiivistelmä.....	2
Thesis abstract	3
SISÄLTÖ.....	4
Kuvaluettelo	6
Käytetyt termit ja lyhenteet	8
1 JOHDANTO.....	9
1.1 Työn tausta ja tavoite	9
1.2 Työn rakenne	9
2 CNC-TEKNIikka	10
2.1 Lastuava työstö.....	10
2.2 CNC-työstökoneen koordinaatisto.....	11
2.3 NC-ohjelma	13
3 CNC-JYRSIMEN MÄÄRITTELY	16
4 MEKANIikkASUUNNITTELU	17
4.1 Autodesk Fusion 360	17
4.2 Runkovaihtoehdot	17
4.3 Lineaarijohteet ja kuularuuvit.....	20
4.4 Y-akseli	21
4.5 X-akseli	23
4.6 Z-akseli.....	24
4.7 Runko kokonaisuudessaan	25
5 SÄHKÖSUUNNITTELU	26
5.1 EPLAN	26
5.2 Karamoottori ja taajuusmuuttaja.....	27
5.3 Sähkösuunnittelu Grbl-ohjaukseen	27
5.3.1 Virtalähteet.....	28
5.3.2 Askelmoottorit ja askelmoottoriohjaimet.....	29
5.3.3 Grbl-kytkentä.....	30
5.4 Sähkösuunnittelu TwinCat 3 -ohjaukseen	31
5.4.1 Virtalähteet.....	31

5.4.2	Boshrexroth-servomootorit.....	31
5.4.3	IndraDrive CS -servovahvistimet.....	31
5.5	Koneen toimintakuvaus automaatio suunnittelua varten	34
5.6	Hätäseispiiri ja Sistema-riskianalyysi	35
6	AUTOMAATIOSUUNNITTELU	39
6.1	IEC 61131-3 ja PLCopen	39
6.2	Grbl v1.1.....	39
6.2.1	Grbl-ohjauksen käyttöönotto	40
6.2.2	Grbl-ohjauksen käyttöliittymä	42
6.3	Beckhoff TwinCAT 3	42
6.3.1	TC3 PLC	43
6.3.2	TC3 I/O	44
6.3.3	TC3 Motion control.....	46
6.3.4	TC3-koneen käyttöliittymä.....	48
6.4	IndraDrive CS -vahvistimien ja -servomoottorien käyttöönotto.....	49
7	POHDINTA.....	56
	LÄHTEET	57
	LIITTEET	60

Kuvaluettelo

Kuva 1. Koordinaatisto.....	11
Kuva 2. Kappalekoordinaatiston origo.	12
Kuva 3. NC-ohjelmaesimerkki.....	13
Kuva 4. NC-ohjelmassa käytettäviä osoitekirjaimia.....	14
Kuva 5. NC-ohjelmassa käytettäviä M-koodeja.....	15
Kuva 6. Portaalimallinen tasojyrsinkone.....	18
Kuva 7. Polvimallinen tasojyrsinkone.	18
Kuva 8. Runkomallinen tasojyrsinkone.	19
Kuva 9. Lineaarijohde ja kuularuuvi.	20
Kuva 10. Y-akselin runko.	21
Kuva 11. X-akselin runko.	23
Kuva 12. Z-akselin runko.	24
Kuva 13. CNC-koneen runko kokonaisuudessaan.....	25
Kuva 15. WanTai Motor -paketin toimitussisältö.	28
Kuva 16. Grbl-ohjauksen kytkentäkuva.....	30
Kuva 17. IndraDrive Cs -vahvistin.....	32
Kuva 18. IndraDrive Cs HCS01 -kytkentäesimerkki.....	33
Kuva 19. CNC-jyrsimen layout automaattiosuunnittelua varten.	34
Kuva 20. Sistema suoritustason riskigraafi.	36
Kuva 21. Turvareleen PL-taso valmistajan ohjekirjasta.	38

Kuva 22. Sistema riskianalysin lopputulos.	38
Kuva 23. Grbl-ohjauksen asetusten määrittely Grbl Panel -ohjelmiston avulla.	41
Kuva 24. Grbl Panel -käyttöliittymän ulkonäkö.....	42
Kuva 25. PLC-lohkon näkymä.....	44
Kuva 26. I/O-lohkon näkymä.....	45
Kuva 27. Motion-lohkon näkymä.....	46
Kuva 28. TwinCat 3 visualization -käyttöliittymä.	48
Kuva 29. Yhteyden muodostaminen IndraDrive Cs -vahvistimille.....	49
Kuva 30. Kommunikointi vahvistimien ja ohjauksen välillä.....	50
Kuva 31. Akselien mekaniikan ja skaalauksen asetukset.	51
Kuva 32. Akselien säätöpiiriasetukset.....	52
Kuva 33. Servovahvistimen asettaminen Easy Startup -tilaan.....	52
Kuva 34. Akselin säätöpiirin automaattisen määrittelyn aloittaminen.	53
Kuva 35. Akselin säätöpiirin automaattisen määrittelyn ensimmäinen vaihe.....	54
Kuva 36. Akselin säätöpiirin automaattisen määrittelyn toinen vaihe.	55
Kuva 37. Akselin säätöpiirin automaattisen määrittelyn kolmas ja viimeinen vaihe.	55

Käytetyt termit ja lyhenteet

CNC	Computer numerical control, tietokoneohjattu numeerinen ohjaus.
PLC	Programmable logic control, ohjelmoitava logiikka.
CAD	Computer-aided drafting, tietokoneavusteinen piirtäminen.
CAM	Computer-aided manufacturing, tietokoneavusteinen NC-koodin tekeminen.
I/O	Inputs and outputs, tulot ja lähdöt.
Interpolointi	Rataohjaus, jossa useampaa akselia voidaan liikuttaa hallitusti yhdenaikaisesti.
Interpolaattori	NC-koodin kääntäjä ohjaukselle.
PL	Turvalaitteiden suoritustaso.
PLr	Turvalaitteilta vaadittava suoritustaso.

1 JOHDANTO

1.1 Työn tausta ja tavoite

Työssä ei ole virallista toimeksiantajaa vaan työn lopputuloksena valmistuva CNC-jyrsin tehtiin omaan käyttöön. Tavoitteena on suunnitella ja valmistaa kolmeakselinen CNC-jyrsin. Projektin edetessä on tarkoitus saada kokemusta koneensuunnittelun eri aihealueilta. Kone suunnitellaan kahdella erilaisella ohjauksella, joiden eroavaisuuksia käsitellään työn lopussa.

1.2 Työn rakenne

Työn alussa käydään läpi NC-tekniikkaa yleisesti, sekä käsitellään lastuavaa työstöä ja NC-ohjelmointia. Ennen suunnittelun aloittamista määritellään rakennettavan CNC-jyrsimen käyttötarkoitus sekä liikealueet.

Mekaniikkasuunnittelussa käydään läpi Autodesk Fusion 360 -ohjelmistoa, erilaisia runkovaihtoehtoja, sekä itse koneen rungon valmistusta.

Sähkösuunnittelussa käsitellään EPLAN-ohjelmistoa, sekä kahden erilaisen ohjauksen kanssa käytettäviä komponentteja ja niiden kytkentää. Koneesta tehdään myös turvalaitteiden riskianalyysi Sistema-ohjelmistolla.

Automaatiosuunnittelussa käsitellään kahta erilaista ohjausta. Ensimmäisenä tarkastellaan Arduino pohjaista Grbl-ohjausta ja tämän jälkeen TwinCat 3 -ohjausta.

2 CNC-TEKNIikka

CNC on lyhenne sanasta computer numerical control, joka tarkoittaa tietokoneohjattua numeerista ohjausta. CNC-koneen tarkoituksena on suorittaa hallittuja liikkeenohjauksia työkalun ja työkappaleen välillä. CNC-teknologiaa käytetään koneissa, jotka yleensä perustuvat materiaalin poistamiseen, lisäämiseen tai muodon muokkaukseen. Materiaalin poistaminen suoritetaan työkalulla jyrsimällä, sorvaamalla, hiomalla tai lävistämällä. (Kandray 2010, 15.) 3D-tulostin on hyvä esimerkki materiaalin lisäämisestä. Materiaalin muokkaus keinoja ovat esimerkiksi putken-taivutuskoneet ja levyntaivutusautomaatit.

CNC-työstö vaatii toimiakseen NC-ohjelman, jonka koneenkäyttäjä joutuu ohjelmoimaan. Ohjelmistot ovat kehittyneet niin paljon, että NC-ohjelman teko voidaan tehdä suoraan 3D-mallista. Yksinkertaisien NC-ohjelmien teko on mahdollista myös käsin. CNC-tekniikan hyödyt nousevat enemmän esille, jos työstettävät kappaleet ovat monimutkaisia tai niitä tehdään sarjatuotantona. (Maaranen 2012, 366.)

2.1 Lastuava työstö

Jyrsiminen luokitellaan lastuavaksi työstöksi, jossa pyörivä työkalu irrottaa työstettävästä kappaleesta lastuja. Lastuamisliikkeen jyrsinnässä tekee työkalu, jota pyöritetään karamoottorilla. Karamoottori koostuu sähkömoottorista ja siinä olevasta työkalunpidikkeestä. Moottorin pyörimisnopeus on yleensä portaattomasti säädettävissä. Työkalunpidikkeitä on useita erilaisia malleja ja koneistuskeskuksissa työkalun vaihto tapahtuu yleensä automaattisesti työkalumakasiinista. (Vesämäki 2014, 17.)

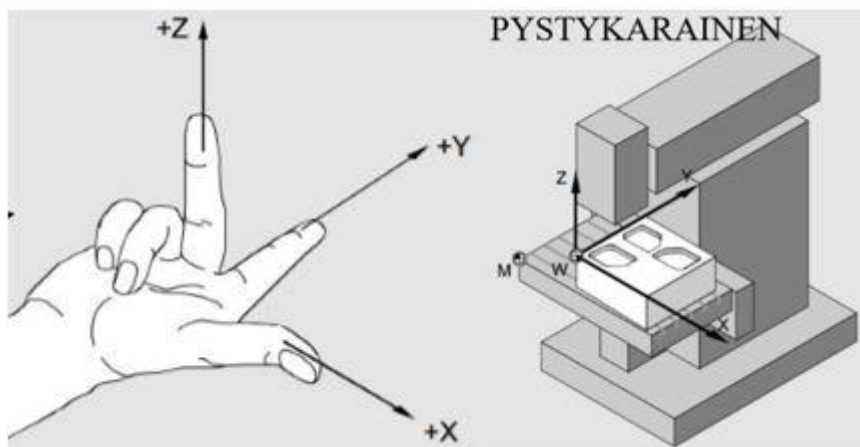
Manuaalityöstökone ja CNC-työstökone noudattavat samoja lastuavan työstön periaatteita. Suurin ero näiden koneiden välillä on ohjaustavassa. CNC-koneelle tehdään NC-ohjelma, jonka koneen ohjaus suorittaa. Manuaalikoneessa koneen käyttäjä huolehtii akselien liikkeistä sekä karamoottorin nopeudesta. (Vesämäki 2014, 10.)

Materiaalia koneistaessa pitää osata määritellä oikea lastuamisnopeus, joka tarkoittaa pyörivän työkalun ja koneistettavan kappaleen liikettä toisiinsa nähden. Sopivan lastuamisnopeuden pystyy tarkistamaan taulukoista, jos koneistettava materiaali ja työkalun materiaali on tiedossa. (Maaranen 2012, 16.)

Lastuamislakosten käyttö koneistuksessa pienentää työkalun ja kappaleen välistä kitkaa ja samalla jähdyttää kuljettaen koneistuksessa irronneet lastut pois. Lastuamislakosten valinta tulisi tehdä lakosten valmistajan ohjeiden mukaan. Koneen alla on yleensä metallinen kuljetin, joka kuljettaa lastut pois työstökoneen alueelta. (Maaranen 2012, 23.)

2.2 CNC-työstökoneen koordinaatisto

CNC-koneen koordinaatisto koostuu kolmesta pääakselista, joita ovat X, Y ja Z. Näiden suunnat määräytyvät oikean käden säännön mukaisesti. Koneen karamoottori kulkee keskisormen mukaisesti Z-akselia pitkin ja positiivinen suunta on ylöspäin. Peukalo ja etusormi näyttävät karamoottorin työkalun liikesuunnan X- ja Y-suunnassa. Jos karamoottorin työkalu liikkuu ja pöytä on kiinteästi paikallaan, X-akselin positiivinen suunta on oikealla. Jos työkalu pysyy paikallaan ja pöytä liikkuu, on pöydän liikkeen positiivinen suunta oikeasti vasemmalle. (Maaranen 2012, 373.)

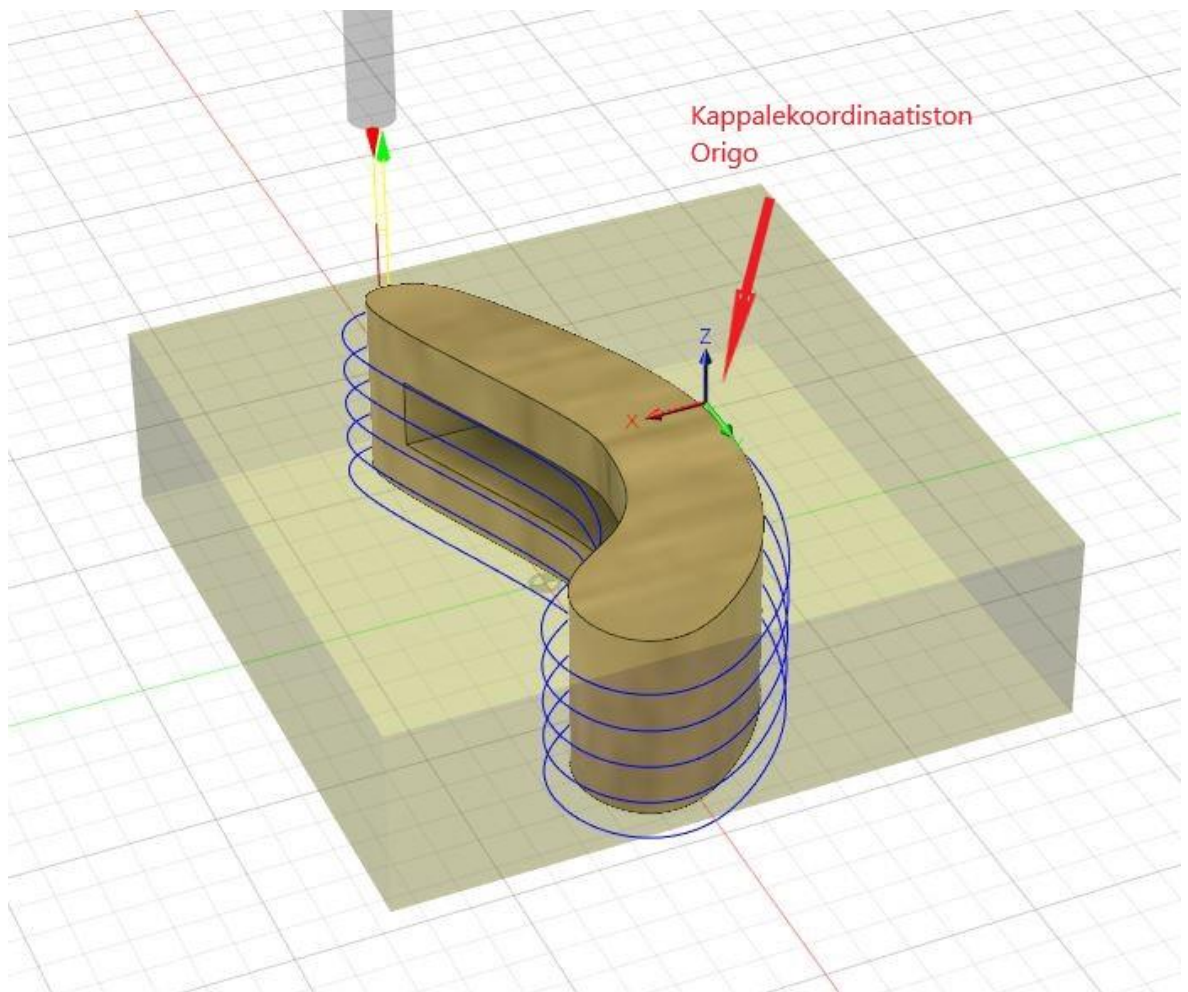


Kuva 1. Koordinaatisto (Siemens [Viitattu 25.5.2019]).

CNC-työstökoneissa voi olla useampia koordinaatiston nollapisteitä. Osa on kiinteitä pisteitä, joita ei voida muuttaa, mutta esimerkiksi kappalekoordinaatistoa voi-

daan muuttaa nollaamalla koneen akselien arvot NC-ohjelmassa määritettyyn pisteeseen. Kiinteitä koordinaatiston pisteitä ovat esimerkiksi koneen nollapiste, joka toimii koneen koordinaatiston perustana. Kaikki muut nollapistet käyttävät konekoordinaatiston arvoja, myös koneen maksimiliikealueet lasketaan konekoordinaatistosta. (Maaranen 2012, 378.)

Kun työstöohjelmaa tehdään, ohjelman teon aikana määritellään kappalekoordinaatiston origo. Origo tarkoittaa pistettä, jossa kaikkien akselien koordinaatit ovat nollassa. NC-ohjelmassa olevat liikkeet suoritetaan tämän pisteen mukaan, jos ohjelman alussa aktivoidaan kappalekoordinaatiston käyttö. (Maaranen 2012, 379.)



Kuva 2. Kappalekoordinaatiston origo.

NC-työstökoneita käytetään pääasiassa automaattikäytöllä, mutta manuaalikäytöllä voidaan esimerkiksi ajaa kone kappalekoordinaatiston nollauspaikkaan.

2.3 NC-ohjelma

NC-ohjelma voidaan tehdä joko käsin ohjelmoimalla tai tietokoneavusteisesti (Vesamäki 2014, 61).

Käsin ohjelmointi on helppoa sitä osaavalle, mutta siitäkin muodostuu työläs prosessi, jos työstettävä kappale on monimutkainen. Ohjelmoijan tulee määrittellä ja laskea akselien liikkeet, lastuamisnopeudet, työstöjärjestykset jne. sekä kirjoittaa NC-ohjelma. (Vesamäki 2014, 61.)

Tietokoneavusteisissa ohjelmissa (CAD/CAM) käytetään työstettävän kappaleen CAD-kuvaa pohjana työstöratoja tehdessä. CAD-ohjelmissa pystytään piirtämään 3D-mallinnus työstettävästä kappaleesta ja CAM-ohjelmasta tehdään työstökoneelle NC-ohjelma. CAM-ohjelmistoissa pystytään yleensä simuloimaan työstöohjelmaa ennen kuin sitä testataan oikealla työstökoneella. CAD- ja CAM-toiminnot voivat löytyä samasta ohjelmistosta tai voidaan käyttää kahta erillistä suunnitteluohjelmaa. (Pikkarainen & Mustonen 2010, 110.)

NC-ohjelma sisältää erilaisia sanoja ja näistä sanoista muodostuu NC-lause. Yksi NC-lause sisältää yhtä työvaihetta koskevia tietoja, näiden lauseiden muodostamaa listaa kutsutaan NC-ohjelmaksi. (Vesamäki 2014, 64.)



Kuva 3. NC-ohjelmaesimerkki.

Sanassa esitellään aluksi osoite, jonka jälkeen syötetään osoitteen lukuarvo. Osoite on kirjain. Tämä kirjain ilmaisee, mihin tarkoitukseen osoitteen perässä esiteltyä lukua käytetään. (Vesamäki 2014, 64.) Kuvassa 4 on esitelty yleisempiä osoitekirjaimia ja niiden merkitys.

O	- ohjelman numero
N	- lausenumero
G	- valmistelevat toiminnot, liiketila
X, Y, Z	- liikekoordinaatit koordinaattiakseleiden suuntaan
A, B, C	- lisäakseleiden liikekoordinaatit
U, V, W	- lisäakseleiden liikekoordinaatit
R	- ympyräkaaren säde, kulman pyöristys
I, J, K	- kaaren keskipisteen koordinaatit, viisteet
F	- syöttönopeus
S	- karan pyörimisnopeus
T	- työkalun numero
M	- kytkentätoiminto
B	- pyöröpöydän asema
H	- pituuden kompensointinumero
D	- säteen kompensointinumero
P, X	- viive
P	- aliohjelman numero, toistettavan lauseen lausenumero
L	- toistokertojen määrä
P, Q, R	- kiinteän työkierron parametrit.

Kuva 4. NC-ohjelmassa käytettäviä osoitekirjaimia (Opetushallitus [Viitattu 25.4.2019]).

G-koodit voivat tarkoittaa esimerkiksi liikekäslyn tyyppiä, joka määrittelee, onko kyse pikaliikkeestä, suoraviivaisesta liikkeestä tai akselien yhdenaikaisesta interpoloinnista. G-koodit voivat olla modaalisia, eli käsky pysyy voimassa niin kauan kun-

nes toinen käsky kumoaa edellisen. G-koodit on luokiteltu eri ryhmiin, kuten liiketyyppeihin, koordinaatiston ja mittayksikköjen hallintaan. Eri ryhmään kuuluvia G-koodeja pystytään esittelemään samassa NC-lauseessa. (Vesämäki 2014, 66.)

Aputoiminnoissa käytetään M-koodeja. M-koodeilla voidaan ohjata esimerkiksi karamoottorin toimintaa tai leikkuunesteen ohjausta. M-koodit eroavat G-koodeista siinä, että ne ovat vapaammin käytettävissä ohjauksen puolesta. Tämä tarkoittaa, että M-koodeissa voi olla koneiden valmistajakohtaisia eroja. G-koodien tarkoitus on vakiintunut, joten nämä käyttäytyvät samoin koneohjauksesta riippuen. (Vesämäki 2014, 66.) Kuvassa 5 on yleisesti käytettyjä M-koodeja.

M00 - ohjelman loppu
M01 - valinnainen pysäytys
M02 - ohjelman loppu
M03 - karan käynnistys myötäpäivään
M04 - karan käynnistys vastapäivään
M05 - karan pysäytys
M06 - työkalun vaihto
M08 - jäähdytys päälle
M09 - jäähdytys pois päältä
M10 - oheislaitteen lukinta
M11 - oheislaitteen lukinnan avaus
M19 - karan pysäytys vakiokulmaan
M30 - ohjelman loppu

Kuva 5. NC-ohjelmassa käytettäviä M-koodeja (Opetushallitus [Viitattu 25.4.2019]).

3 CNC-JYRSIMEN MÄÄRITTELY

Ennen suunnittelutyön aloittamista määriteltiin koneen käyttötarkoitus, liikealueet ja niihin käytettävät komponentit, jotta saatiin koneen suunnittelun aloittamiseen tarvittavat alkutiedot.

Koneessa on kolme akselia ja sillä on tarkoitus työstää puuta ja pehmeitä metalleja, esimerkiksi alumiinia. Teräksen koneistusta on tarkoitus kokeilla testimielessä. Koneen liikealue tulee olla vähintään X-suunnassa 1000 mm, Y-suunnassa 500 mm ja Z-suunnassa 400 mm. Jokainen lineaariliike suoritetaan lineaarijohteilla ja kuularuuveilla. Kuularuuveja pyöritetään moottoreilla.

Jokaisella akselilla on kaksi päätyrajaa, joista toista käytetään myös kotirajana. Kotirajojen avulla suoritetaan akseleiden koordinaatiston nollaus. Koneen alle tulee koneistuslastujen poistoa varten jätekuljetin. Koneeseen suunnitellaan suojaseinät, joissa on yksi käyntiovi. Häätäseis-painikkeita on yhteensä kaksi, toinen suojausten sisäpuolella ja toinen ulkopuolella.

Z-akselissa olevan karamoottorin tulee soveltua kierrosnopeudeltaan puun ja alumiinin jyrsimiseen. Tämä tarkoittaa yleensä suurnopeuskaramoottoria, jota ohjataan taajuusmuuttajalla. Jyrsintapin vaihto tapahtuu manuaalisesti, työkalun ympärille tulee lastusuoja, johon on mahdollista kiinnittää puruimuri.

4 MEKANIKKASUUNNITTELU

4.1 Autodesk Fusion 360

Autodesk Fusion 360 on 3D-suunnitteluohjelma, jossa on yhdistetty CAD-, CAM- ja CAE-ohjelmisto. Ohjelmistosta on saatavana ilmainen versio opiskelijakäyttöön. Ohjelmistolla on mahdollista suunnitella monimutkaisia mallinnuksia joko perinteisellä pursotustyylillä tai ohutlevytyylillä. Samalla ohjelmalla on mahdollista suorittaa kappaleen simulaatiota, jossa voidaan testata kappaleen kestävyyttä erilaisilla kuorimitustavoilla. Ohjelmalla voidaan myös suorittaa kokonaisuuden animaatio. CAM-ominaisuudella pystytään tekemään kappaleen 3D-mallinnuksesta valmis työstöohjelma, esimerkiksi CNC-jyrsimelle tai sorville. Ohjelma tukee useita eri ohjaustyypppejä, joten NC-ohjelman saa tehtyä koneen ohjauksen ehdoilla. (Autodesk [Viitattu 10.5.2019].)

Saatavilla on erilaisia 3D-suunnitteluohjelmia, mutta kyseinen ohjelmisto osoittautui sopivaksi ohjelmaksi jyrsimen työstöohjelman tekoon. Ohjelmisto on helppo käyttää, ja verkosta löytyy hyvin materiaalia ohjelman käytön aloittamiseksi. Koneen mallinnus on suoritettu myös kyseisellä ohjelmistolla.

4.2 Runkovaihtoehdot

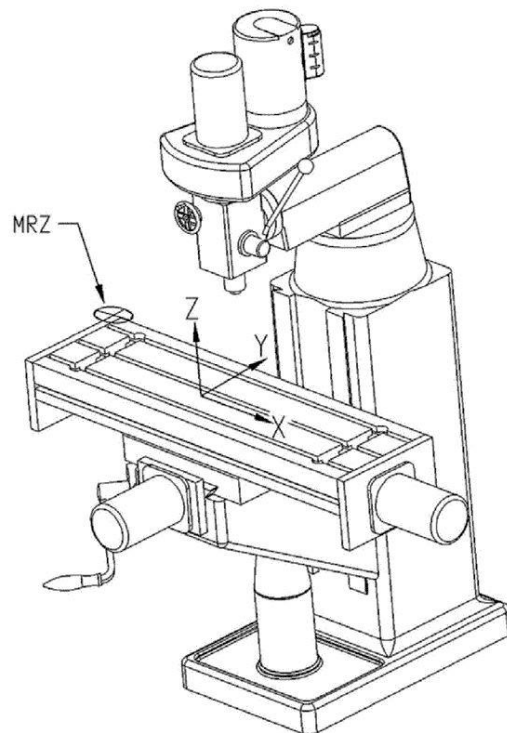
Runkovaihtoehtoina CNC-jyrsimeen harkittiin portaali-, polvi- tai runkomallista ratkaisua.

Portaalimalli tarkoittaa, että koneessa on joko kiinteä pöytä ja työkalu liikkuu X-, Y- ja Z-suunnassa, tai pöytä liikkuu Y-suunnassa ja työkalu X- ja Z-suunnassa. Tämän mallin ominaisuuksiin kuuluu yleensä suuremmat liikealueet, mutta samalla koneen mekaaninen jäykkyys on hankalampi toteuttaa. Yleisesti tätä mallia käytetään puunjyrsintä koneissa. (Bralla 2007, 270.)



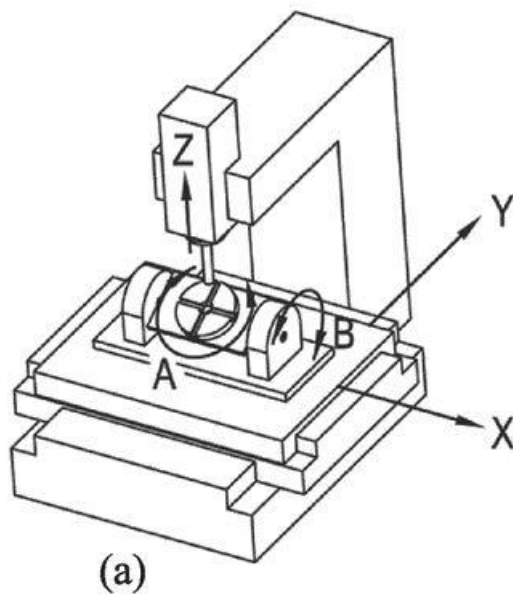
Kuva 6. Portaalimallinen tasojyrsinkone (Brala 2007, 271).

Polvimallisessa tasojyrsinkoneessa karamoottorin työkalu pysyy paikallaan ja pöytää liikuttamalla toteutetaan työstön syöttöliike X-, Y- ja Z-suunnassa. Tämä on yleisin malli, jota käytetään pienissä ja keskiuurissa manuaalityöstökoneissa. Näillä koneistetaan yleisemmin pienehköjä työkappaleita (Maaranen 2012, 244.)



Kuva 7. Polvimallinen tasojyrsinkone (Kandray 2010, 97).

Runkomallisessa tasojyrsinkoneessa karamoottorin työkalu liikkuu vain Z-suunnassa ja pöytä liikkuu X- ja Y-suunnassa. Tämän takia Z-akseli ja liikkuva pöytä on helpompi valmistaa jäykemmäksi, mutta liikkuvan pöydän takia Y-suuntainen liike on yleensä pienempi kuin portaalimallisessa. Tätä runkomallia käytetään koneistuskeskuksissa, joilla työstetään metallia ja työstettävät kappaleet ovat painavia. (Maaranen 2012, 245.)



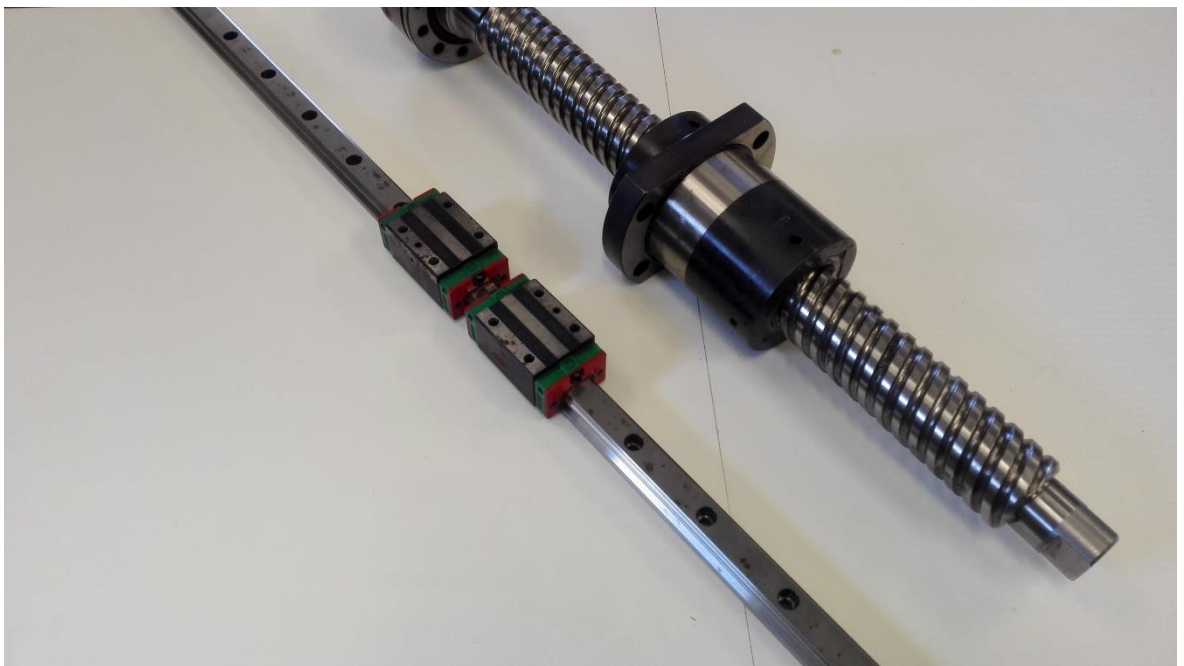
Kuva 8. Runkomallinen tasojyrsinkone (Kandray 2010, 98).

CNC-jyrsimen malliksi valittiin runkomallinen tasojyrsinkone, koska tämä vaikuttaa sopivan koneen käyttötarkoitukseen parhaiten. Jyrsimellä on tarkoituksena kokeilla myös teräksen koneistamista. CNC-jyrsin joudutaan valmistamaan ilman kappaleiden koneistusmahdollisuutta, joten tämä tuo haasteita koneen mekaanisiin säätötoimenpiteisiin.

4.3 Lineaarijohteet ja kuularuuvit

Lineaarijohde on profiloitu liukukisko, jota pitkin asennettu laakerikelkka pääsee liukumaan. Laakerikelkka sisältää pieniä kuulia, jotka mahdollistavat melkein kitkattoman liikkeen. Lineaarijohdepaketteja on saatavilla kaiken kokoisina ja mallisina. (Hiwin [Viitattu 2.5.2019].)

Kuularuuvi koostuu ruuvikarasta ja siinä olevasta kuulamutterista. Ruuvikaraa pyörittämällä kuulamutteri liikkuu ruuvikaran suuntaisesti. Kuulamutteri sisältää kuulia, joiden vierintäominaisuus tekee liikkeestä sujuvaa ja tarkkaa. Kuularuuveilla muutetaan pyörivä liike lineaariseksi liikkeeksi. Kuularuuveja käytetään yleisesti teollisuuskoneissa, joilta vaaditaan suurta tarkkuutta. (Hiwin [Viitattu 3.5.2019].)



Kuva 9. Lineaarijohde ja kuularuuvi.

CNC-jyrsimen akselien lineaarinen liike on toteutettu lineaarijohteilla ja kuularuuveilla. Tämä on nykypäivänä yleisimpiä tapoja tehdä koneen lineaariliikkeet (Maaranen 2012, 369). Kuularuuveja pyöritetään moottoreilla. Jokaisella akselilla on kaksi lineaarijohdetta, joissa kumpikin johde on varustettu kahdella laakerikelkalla. Työssä käytetään Hiwin-merkkisiä johteita ja kuularuuveja.

4.4 Y-akseli

Y-akselin rungon valmistukseen on käytetty 50 mm x 100 mm teräsprofiilia, jonka vahvuus on 5 mm. Tämä runko toimii koko koneen alustana. Runkoon on käytetty laserleikattuja osia, joita on kuularuuvin ja moottorin kiinnikkeissä sekä takana olevassa tasossa, johon Z-akselin pylväs kiinnitetään. Rungon teräsprofiilit on kiinnitetty toisiinsa hitsaamalla, samalla on pidetty rungon suoruus mahdollisimman hyvänä. Tulevat johdepinnat on kevyesti hiottu tasahiomakoneella.



Kuva 10. Y-akselin runko.

Y-akselin komponentit on asennettu ensimmäisenä koneen rungon päälle. Johteina on käytetty 930 mm pitkiä lineaarijohteita, ja molemmissa johteissa on kaksi lineaarikelkkaa. Johteiden asennuksessa on käytetty työkaluina pitkää ja lyhyttä tarkkuusvatupassia sekä mittakellolla varustettua magneettijalkaa.

Aluksi toinen johteista asennettiin rungon päälle ja johteen kyljen- ja pinnansuoruus tarkastettiin käyttämällä tarkkuusvatupasseja. Työvaiheessa käytetyn tarkkuusvatupassin yhden viivan välinen tarkkuus on 0,02 mm metriä kohti. Johteen alapinnalla

jouduttiin käyttämään korotuspaloja, jotta johde olisi mahdollisimman suorassa korkeussuunnassa. Tämän jälkeen johde pultattiin runkoon kiinni. Tätä johdetta voidaan käyttää toisen johteen asentamisessa. Asennettuun johteeseen asetettiin yksi laakerikelkka, johon on kiinnitetty magneettijalalla varustettu mittakello. Tämän jälkeen toinen johde asetellaan paikalleen ja mittakellon mittakärki laitetaan asennettavan johteen kylkeen. Mittakelloa siirtämällä saadaan uusi asennettava johde käännettyä samansuuntaiseksi jo asennetun johteen kanssa, se voidaan tämän jälkeen pultata kiinni. Uuden johteen korkeus voidaan tarkastaa asettamalla mittakellon kärki johteen yläpintaan ja siirtämällä mittakelloa. Tällä saadaan selville johteen alle asetettavien korotuspalojen määrä. Mittausten jälkeen molempien johteiden tulisi olla suorassa toisiinsa nähden.

Kuularuuvi on 800 mm pitkä ja 32 mm halkaisijaltaan. Kuularuuvien nousu on 40 mm. Nousu tarkoittaa, kuinka paljon kuularuuvissa oleva mutteri liikkuu yhden pyörähdysten aikana. Moottorin ja kuularuuvien välinen liitos tapahtuu hammashihnapyörillä ja hihnalla. Välityssuhde on 5:1 eli moottorin pitää pyörähtää viisi kertaa ympäri, että kuularuuvi saadaan pyörähtämään kerran. Samaa välityssuhdetta käytetään myös X- ja Z-akseleilla. Kuularuuvi asennetaan päädystä paikalleen ja kuulamutteri kiinnitetään seuraavassa vaiheessa X-liikkeen pöytään.

4.5 X-akseli

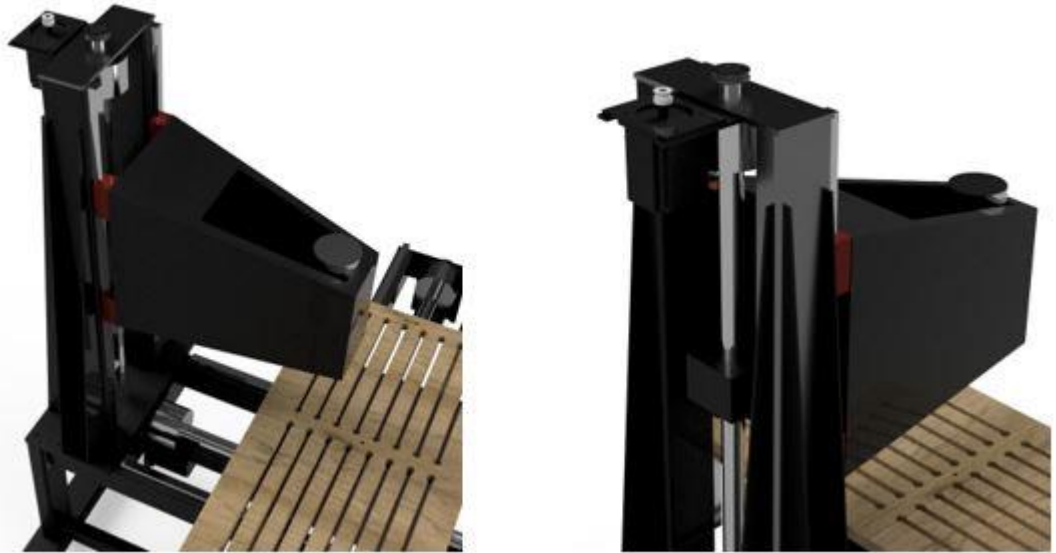
X-liikkeen rungon valmistuksessa käytetään samaa putkiprofiilia kuin Y-liikkeessä, ja kuularuuvien ja moottorin kiinnikkeet ovat samanlaiset. Johteet ovat X-liikkeessä 1510 mm pitkät ja kuularuuvi 1380 mm pitkä. Näiden asennus tapahtuu samalla tavalla kuin Y-liikkeellä. X-liikkeen laakerikelkkoihin ja kuularuuviin kiinnitetään laserleikattu metallinen pöytälevy, jonka päälle on tarkoituksena asentaa T-urapöytä. Kun X-liikkeen runko on koottu, se asetellaan Y-liikkeen laakerikelkkojen päälle ja pultataan kiinni. Y-akselin kuularuuvi kiinnitetään X-liikkeen runkoon. Nyt X- ja Y-liike ovat valmiina.



Kuva 11. X-akselin runko.

4.6 Z-akseli

Z-liikkeen pylväs kootaan erillisenä, siinä käytetään samoja työmenetelmiä kuin edellisissä vaiheissa. Tässäkin moottorin ja kuularuuvien kiinnikkeet ovat samanlaiset ja runkomateriaalina käytetään samaa putkiprofiilia kuin edellisissä vaiheissa. Putkiprofiilien kylkeen on lisäksi hitsattu 3 mm vahvat kolmiotuet tuomaan runkoon lisäjäykkyyttä. Kuularuuvien mutteri kiinnitetään Z-suunnassa liikkuvaan karamoottorin runkoon, joka on valmistettu 3 mm vahvasta teräksestä laserleikkaamalla ja hittaamalla. Johteet ovat Z-liikkeessä 810 mm pitkät ja kuularuuvi on 900 mm pitkä. Kun Z-liikkeen runko on koottu, asennetaan se kiinni Y-liikkeen runkoon pulteilla.



Kuva 12. Z-akselin runko.

4.7 Runko kokonaisuudessaan



Kuva 13. CNC-koneen runko kokonaisuudessaan.

Kuva 13 osoittaa, että 3D-suunnitelman ja todellisen koneen välillä ei ole suuria eroja. Asennus on aikaa vievää, johtuen koneistusmahdollisuuden puuttumisesta. Johteiden asennus tarkkuusvatupasseilla ja mittakellolla vie suurimman osan ajasta. X-liikkeen rungon suoruus Y-liikkeeseen nähden varmistetaan, kun koneella saadaan koneistettua kappaleita ja näistä saadaan mitattua kulmien ristimitta.

CNC-jyrsimen liikealueeksi muodostui Y-suunnassa 520 mm, X-suunnassa 1010 mm ja Z-suunnassa 410 mm.

5 SÄHKÖSUUNNITTELU

Tässä osiossa käsitellään kahden erilaisen koneenohjauksen sähkösuunnittelua ja niihin käytettäviä komponentteja. Grbl-ohjauksen kanssa on käytetty valmistajan sivustolta löytyviä kytkentäesimerkkejä, mutta TwinCat 3 -ohjaukseen on piirretty sähkösuunnitelmat EPLAN-sähkösuunnitteluohjelmistolla.

TwinCat 3 -ohjauksen sähkösuunnitteluun on sisällytetty myös toimintakuvauksen tekeminen automaattiosuunnittelun avuksi.

5.1 EPLAN

EPLAN on laaja-alainen suunnitteluohjelmisto. Ohjelmisto voi sisältää toiminnallisen suunnittelun, teknillisen esisuunnittelun, hydraulikkasuunnittelun, sähkösuunnittelun, keskusten suunnittelun ja johdinsarjojen suunnittelun. EPLAN Platform on alusta, joka yhdistää nämä kaikki suunnitteluohjelmistot toisiinsa ja mahdollistaa tiedonkulun eri ohjelmistojen välillä. (Eplan [Viitattu 12.5.2019].)

EPLAN Electric P8 -ohjelmaa käytetään sähkösuunnittelussa. Kyseistä sähkösuunnitteluohjelmaa käytetään CNC-jyrsimen piirikaaviokuvien tekemiseen.

EPLAN-ohjelmistossa on myös saatavilla Data Portal -verkkopalvelu, joka mahdollistaa tällä hetkellä yli 200 eri valmistajan komponenttitietoihin pääsyn. Komponentteja on järjestelmässä yli 800 000 kappaletta. Komponentit pystytään yksinkertaisesti siirtämään omaan työhön, tätä kautta valmistajan komponenttimäärittelyt ovat heti saatavilla. Komponentit sisältävät erilaisia määrittelyitä, jotka muuttuvat työpiirustuksen tyyppin mukaan. Esimerkiksi sähkökuvien piirikaaviossa näkyy komponentin piirikaaviokuva ja sähkökeskuksen suunnittelussa näkymäksi muuttuu komponentin keskuslayout-kuva. (Eplan [Viitattu 12.5.2019].)

5.2 Karamoottori ja taajuusmuuttaja

Karamoottoriksi on valittu teholtaan 1.5 kw:n kolmevaihemoottori, jota ohjataan taajuusmuuttajalla. Kyseisessä moottorissa on vesijäähdytys ja ER16-työkalunpidike. Karamoottorin maksimi kierrosnopeus minuutissa on 16 000, mikä soveltuu hyvin puunjyrsintään.

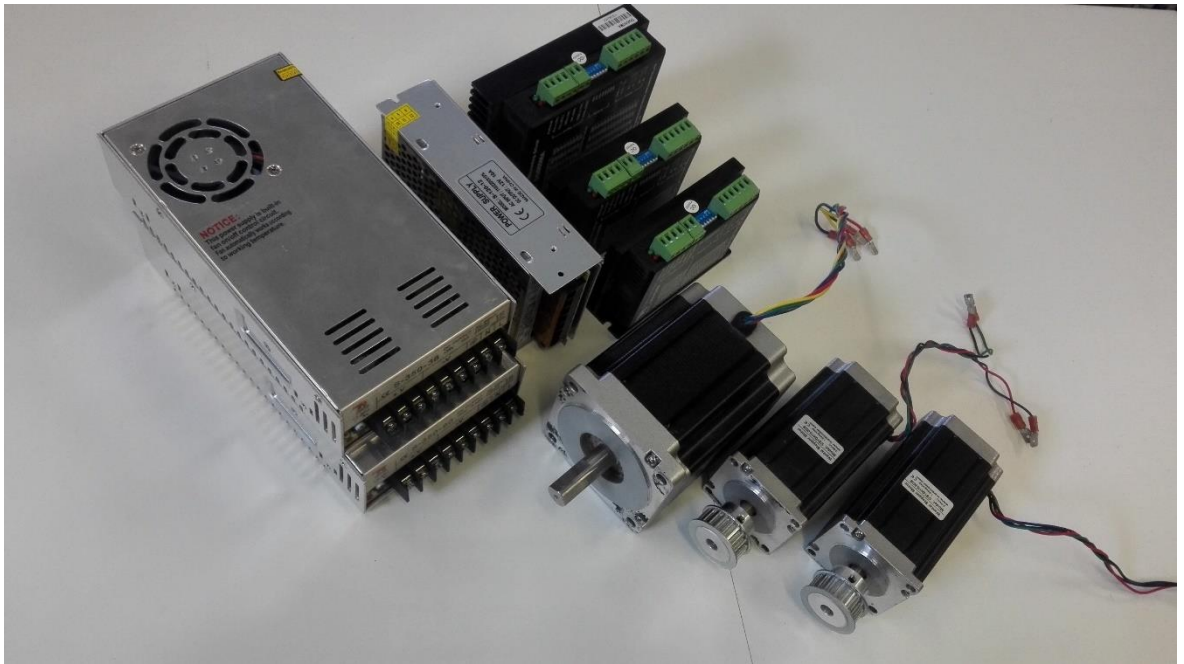
Karamoottorin nopeutta voidaan ohjata taajuusmuuttajan avulla. Taajuusmuuttajan asetuksiin voidaan määritellä pienin ja suurin sallittu taajuusarvo. Taajuusmuuttajan nopeustieto voidaan lähettää ohjaukselta, joko 0-5 V:n signaalilla, 0-10 V:n signaalilla tai 0-20 mA:n signaalilla. Nopeustiedon syöttötapa voidaan valita taajuusmuuttajan asetuksista. Grbl-ohjauksen kanssa käytetään 0-5 V:n signaalia ja TwinCat 3 -ohjauksen kanssa 0-20 mA:n signaalia.

Moottorin jäähdytys on tehty erillisellä jäähdytynyksiköllä, jota ohjaa Arduino nano -logiikka. Jäähdyttimessä on vesisäiliö ja lämmönvaihdin, joiden sisällä kiertää jäähdytinneste. Logiikkaan on kytketty karamoottorin lämpötila-anturi, jonka mukaan jäähdytynyksikön pumppu käynnistyy, pitäen lämpötilan halutulla alueella.

5.3 Sähkösuunnittelu Grbl-ohjaukseen

Grbl-ohjauksen kanssa X-, Y- ja Z-moottoreina on käytetty askelmoottoreita. Moottoreita ohjataan askelmoottorihjaimilla. Sähkösuunnitelmat on tehty ohjauksen valmistajan sivuilta löytyvien kytkentäohjeiden perusteella.

Työhön on valittu JiangSu WanTai Motor co., Ltd:n toimittamat askelmoottorit ja askelmoottorihjaimet. Toimitus sisälsi valmiin kolmen akselin paketin, jossa oli kolme askelmoottoria, kolme askelmoottorihjainta sekä näille virtalähteet.



Kuva 14. WanTai Motor -paketin toimitussisältö.

Toimitussisältö oli:

230 VAC → 12 VDC:n virtalähde

230 VAC → 36 VDC:n virtalähde

2 kpl DQ542MA-askelmoottoriohjainta

2 kpl 57BYGH115.003B -askelmoottoria

230 VAC → 60 VDC:n virtalähde

1 kpl DQ860MA-askelmoottoriohjain

1 kpl WT86STH118-6004A-askelmoottori

5.3.1 Virtalähteet

Arduino-ohjaus vaatii 5-12 V:n tasajännitteen. Z-liikkeen askelmoottorin DQ860MA-askelmoottoriohjain vaatii 60 V:n tasajännitteen. X- ja Y-liikkeen askelmoottorien

DQ542MA-askelmoottoriohjaimet vaativat 36 V:n tasajännitteen. Kaikki jännitelähteet tulivat WanTai Motorin toimituksessa. Jännitelähteet vaativat 230 VAC:n syötön ja näistä jokaisesta saadaan ulos tasajännitettä maksimissaan 10 A:n kuormalle.

5.3.2 Askelmoottorit ja askelmoottoriohjaimet

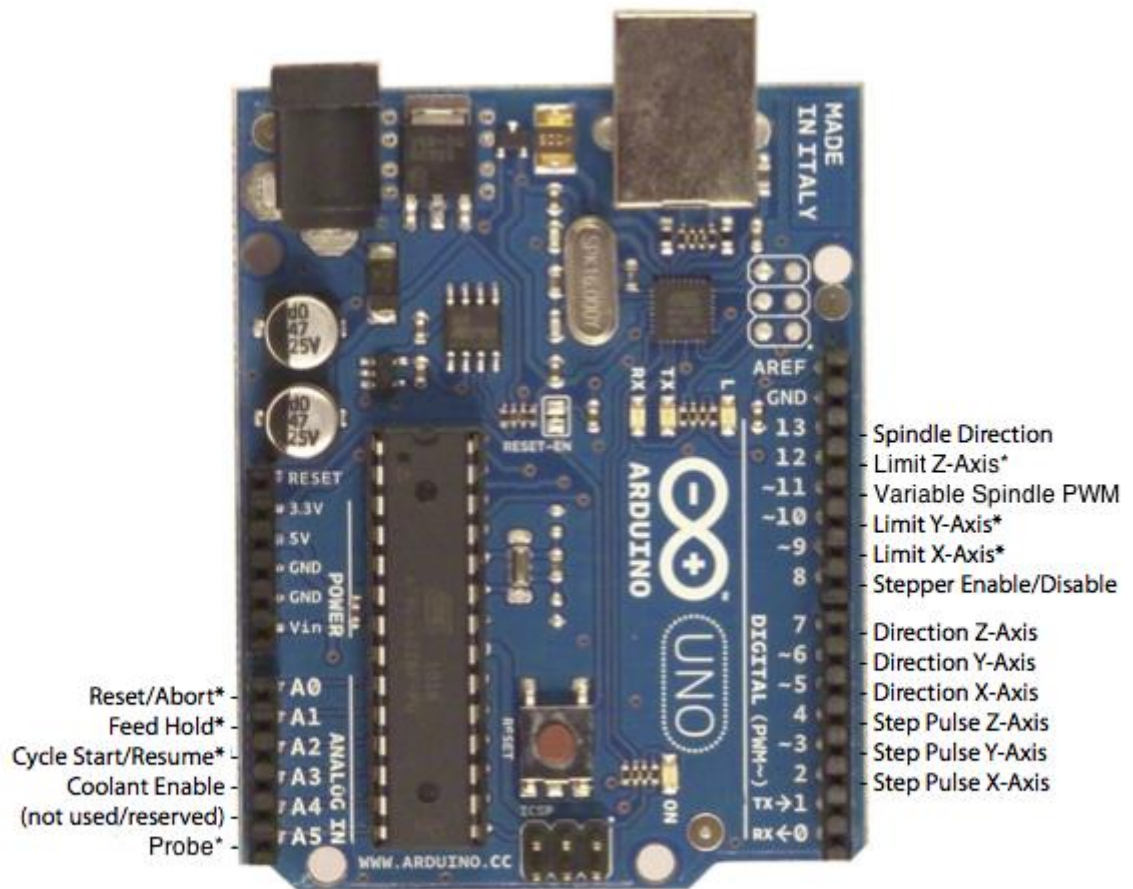
Askelmoottori on hiiliharjaton tasasähkömoottori, jota voidaan käyttää ilman pulssi-anturia. Askelmoottorin liike perustuu pulsseihin, joilla ohjataan moottorin sisällä olevia keloja päälle tai pois. Moottorin ohjaamisessa on hyvä käyttää askelmoottoriohjainta, koska nämä ohjaavat moottorin keloja hallitusti ja oikeassa järjestyksessä liikesuuntaan nähden. Askelmoottoriohjain vaatii CNC-koneen ohjaukselta päällekytkentätiedon, suuntatiedon ja askelpulssin. Kun askelmoottoriohjaimelle antaa pulssin, moottori pyörähtää yhden askeleen suuntatiedon mukaiseen suuntaan. Yhden kierroksen pulssimäärä riippuu askelmoottorin mallista. Yleensä askelmoottoreissa on 200 pulssia kierroksella, eli tämä tarkoittaa 1,8 asteen pyörähdystä jokaisella askelpulssilla. (Wantmotor 13.5.2016.)

Työssä käytettävissä askelmoottoreissa on kaikissa 1,8 asteen askelkulma.

57BYGH115.003B-askelmoottoreita on kaksi kappaletta, nämä tulevat X- ja Y-liikkeiden moottoreiksi. Nämä ovat Nema 23 -kokoluokan moottoreita, joissa pitomomentti on 3,2 N/m ja yhden moottorin ottama maksimivirta on 3 A. Kyseisiä askelmoottoreita ohjataan DQ542MA-askelmoottoriohjaimilla, jotka saavat käyttöjännitteen 36 VDC:n virtalähteestä.

WT86STH118-6004A-askelmoottori tulee Z-liikkeen moottoriksi. Tämä on Nema 34 -kokoluokan moottori, jossa pitomomentti on 9,4 N/m, ja moottorin ottama maksimivirta on 5,6 A. Kyseistä askelmoottorien ohjataan DQ860MA-askelmoottoriohjaimella, joka saa käyttöjännitteen 60 VDC:n virtalähteestä.

5.3.3 Grbl-kytkentä



* - Indicates input pins. Held high with internal pull-up resistors.

Kuva 15. Grbl-ohjauksen kytkentäkuva (Github [Viitattu 2.5.2019]).

Ohjauksen kytkentä voidaan toteuttaa Grbl-ohjauksen sivustolta löytyvän kytkentäohjeen mukaisesti. Vakiona ohjauksen tulot on ohjelmoitu niin, että tulosignaali on yhdistetty sisäisellä ylösvetovastuksella +5 volttiin, ja tulon ohjaus tapahtuu kytkemällä tulosignaali maahan. Ohjaussignaalit toimivat 0-5 voltin alueella, joten pienimmätkin häiriöt voivat vaikuttaa ohjauksen toimintaan. Tämä tarkoittaa käytännössä sitä, että kaikki Grbl-kytkentään liittyvät sähköiset komponentit tulee yhdistää samaan maapotentiaaliin, ja kytkennöissä on hyvä käyttää häiriösuojattuja kaapeleita. Erityisesti karamoottorin taajuusmuuttaja tuottaa häiriösignaaleja helposti.

Askelmoottorihajimet tarvitsevat suuntatiedon (Direction), askelpulssin (Step Pulse) ja luvan liikkua (Stepper Enable/Disable). Grbl-ohjauksen sivustolta löytyy muutamia kytkentäesimerkkejä erilaisille askelmoottorihajimille.

5.4 Sähkösuunnittelu TwinCat 3 -ohjaukseen

TwinCat 3 -ohjauksen kanssa X-, Y- ja Z-moottoreina on käytetty Bosch Rexrothin servomoottoreita. Moottoreita ohjataan saman valmistajan IndraDrive Cs -servovahvistimilla. Sähkösuunnitelmissa on käytetty valmistajan sivuilta löytyviä kytkentä-esimerkkejä.

5.4.1 Virtalähteet

TwinCat 3 -ohjauksen kanssa käytettävät sähköiset komponentit toimivat joko 3 x 400 V:n vaihtojännitteellä tai 24 V:n tasajännitteellä, joten työssä ei tarvita kuin 24 V:n tasajännitevirtalähde.

5.4.2 Boshrexroth-servomoottorit

Kolmivaiheservomoottorit eli epätahtivaihtovirtamoottorit ovat yleisin moottorityppi, jota käytetään CNC-koneissa. Servomoottorin staattorissa on kolmevaihekäämintä, ja moottorin akselissa olevassa roottorissa on kestopagneetti. Näissä moottoreissa on yleensä sisäänrakennettu pulssianturi, jolla moottorin pyörimisnopeus voidaan määrittellä. Kolmivaiheisien servomoottorien käyttö yleistyi taajuusmuuttajien tullessa markkinoille, tämä mahdollistaa kolmivaiheservojen tarkan ohjauksen. Taajuusmuuttajilla muutetaan joko moottorille menevän syötön taajuutta tai jännitettä. Servokäytöissä olevista taajuusmuuttajista käytetään myös nimitystä servovahvistin. (Pikkarainen & Mustonen 2010, 49.)

5.4.3 IndraDrive CS -servovahvistimet

IndraDrive Cs HCS01 -sarjan servovahvistimet ovat kompakteja vahvistimia, joissa on yhdistettynä tehonsyöttö- ja ohjausmoduuli. Vahvistimissa on tuki Ethernet-pohjaisille tiedonsiirtomenetelmille, erilaisille pulssiantureille, näissä on myös sisäänrakennettu turvatoimintoteknologia. (Boschrexroth [Viitattu 14.5.2019].)

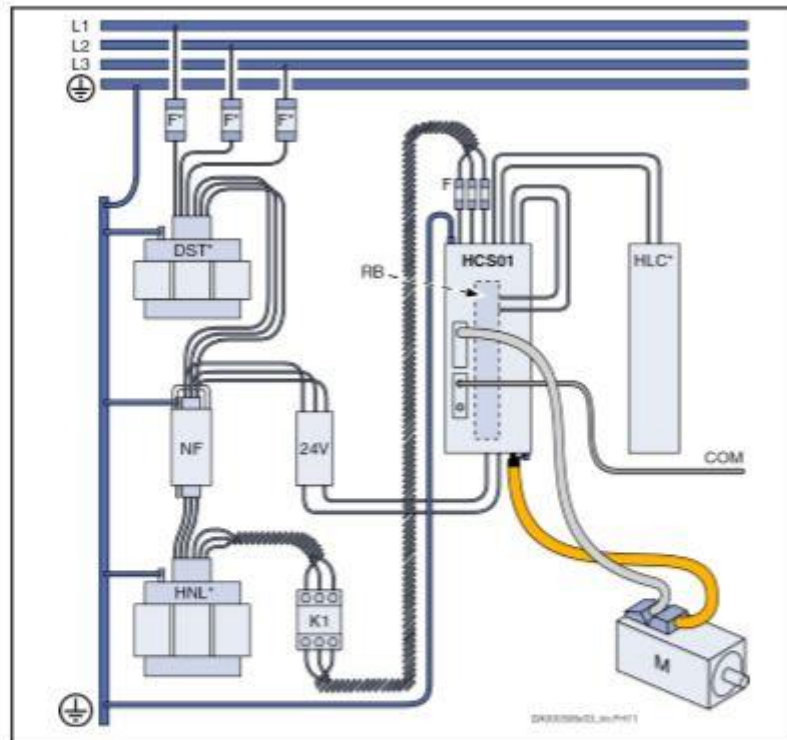
Työssä käytetään IndraDrive Cs HCS01.1E-W0008-A-03 -servovahvistimia. Pääsyöttö vahvistimille on 3 x AC 200-500 V, joten syötön jännitealue on Suomen jänniteverkkoon soveltuva. Vahvistimet tarvitsevat lisäksi 24 V:n tasajännitesyötön ohjauselektronikalle. Vahvistin ottaa syötöstä 2,5 A:n virran. Vahvistin on suunniteltu 2,7 A:n tasaiselle moottorikuormalle, mutta maksimivirta voi olla jopa 8 A hetkellisesti. Moottorilähdön jännitealue on 3 x 0-500 V ja taajuusalue 0-1200 Hz.



Kuva 16. IndraDrive Cs -vahvistin.

HCS01-sarjan servovahvistimen syötön edessä suositellaan käytettäväksi kuristinta HNL01. Tämä varmistaa korkeamman ja tasaisemman tehon vahvistimille ja kuristimet vähentävät harmonisia aaltoja syötössä, minkä takia häiriöt järjestelmässä vähentyvät. (Boschrexroth [Viitattu 15.5.2019].)

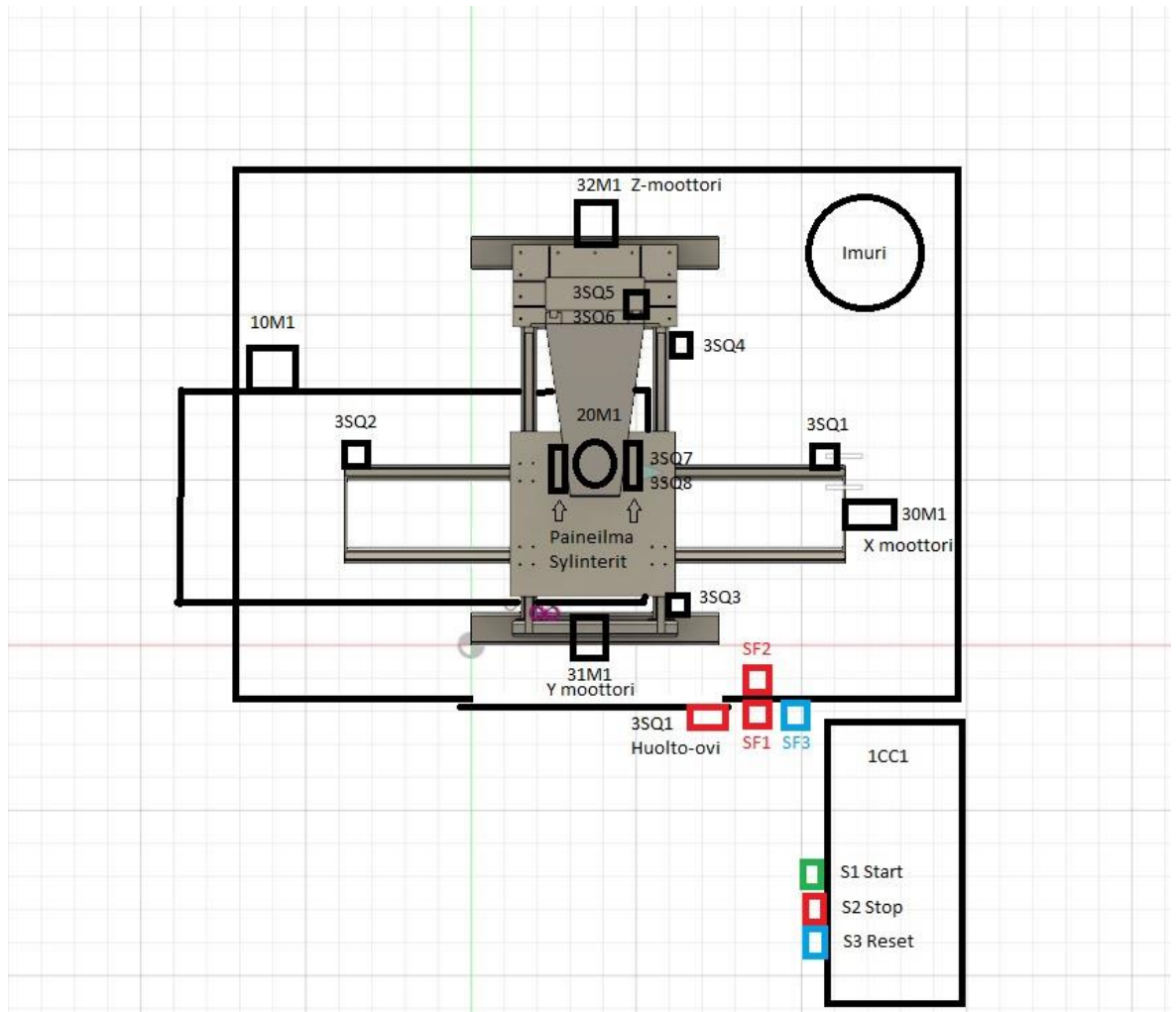
Wiring Diagram



*	Optional
24V	Control voltage supply
HLC	DC bus capacitor unit (for devices with DC bus connection)
COM	Communication
DST	Autotransformer
F	Fuses
HCS01	Converter
NF	Mains filter
HNL	Mains choke
K1	External mains contactor
M	Motor
RB	Braking resistor (at the back of the drive controller)
Fig. 5-1:	Drive System Rexroth IndraDrive Cs

Kuva 17. IndraDrive Cs HCS01 -kytkentäesimerkki (Boschrexroth 11.3.2013).

5.5 Koneen toimintakuvaus automaatio suunnittelua varten



Kuva 18. CNC-jyrsimen layout automaatio suunnittelua varten.

Jokaiselle akselille tulee kaksi päätyrajaa, joista toista käytetään myös kotirajana. Kaikki rajat on kytketty Beckhoffin 2A3_DI-tulokorttiin.

Karamoottorina toimii kolmevaiheinen moottori 20M1, jota ohjataan taajuusmuuttajalla 20U1. Taajuusmuuttajan kierrosnopeutta ohjataan analogisignaalilla kortin 2A9_AO kautta. Muut ohjaussignaalit ovat lähtökortilla 2A5_DO. o_SpindleForward, o_SpindleReverse, o_SpindleReset.

Koneen alla on jyrsinlastuille tarkoitettu jätekuljetin, joka siirtää lastut koneen suo-
jauksien ulkopuolelle. Kuljetinta ohjataan lähdöllä o_ScrapConveyorON.

Jyrsintapin ympärillä on paineilmasylintereillä laskeva lastusuoja, josta on tarkoitus myös imeä lastua/pölyä imurilla. Sylintereiden rajat 3SQ7 ja 3SQ8 on kytketty 2A3_DI-tulokorttiin. Imurille on varattuna pistorasia sähkökaapissa, jonka toimintaa voidaan ohjata lähtökortin 2A5_DO: o_AspiratorON lähdöllä.

Koneen ympärille tulee suojaseinät, joissa on käyntiovi koneen sisälle. Hätäseisnappeja on kaksi kappaletta SF1 ja SF2, toinen on suojauksien ulkopuolella ja toinen sisäpuolella. SF1-napissa on myös hätäseispiirin kuitausnappi. HS-piirissä käytetään Pilz-turvareleitä, joissa on 0,5 sekunnin päästöhidastus. Koneelle annetaan 0.5 sekuntia aikaa pysäyttää liikkeet hallitusti, ennen kuin syöttö katkaistaan.

Käyntioivessa on turvalukko 3SQ1, joka saadaan lukittua lähtökortin 2A5_DO: o_CloseDoorLock lähdöllä. Kun kone on toimintakunnossa, ovi lukitaan ohjauksen kautta. Koneen käyttöliittymään tulee tehdä lupapyyntö-nappi, jonka avulla voidaan kulkea koneen sisälle hallitusti. Ovilukko avautuu, jos painetaan HS-nappia, koska ohjauksen lähtöpuolen tehonsyöttökortista 2A4_PWR katkaistaan jännite.

5.6 Hätäseispiiri ja Sistema-riskianalyysi

Sistema on saksalaisen työturvallisuus- ja työterveyslaitoksen IFA:n kehittämä ilmainen ohjelmisto koneturvallisuuden tason arviointiin ja testaamiseen. Sistema tarkoittaa "Safety Integrity Software Tool for the Evaluation of Machine Applications". Ohjelmisto noudattaa standardia ISO 13849-1, joka on kansainvälinen koneturvallisuuden standardi. (IFA [Viitattu 15.5.2019].)

Sistema-ohjelmistoon on tarkoitus lisätä kaikki koneessa käytettävät turvalaitekomponentit ja määritellä jokaisen komponentin suoritustaso PL. PL-luokkia on yhteensä viisi a, b, c, d, e, joista a tarkoittaa matalimman turvallisuustason vaatimusta ja e tarkoittaa suurinta turvallisuustason vaatimusta. Koneessa olevat turvalaitteet voidaan lisätä ohjelmistoon joko käsin komponenttivalmistajan tietojen perusteella tai komponentit voidaan lisätä valmistajan tekemän kirjaston mukaan. IFA:n sivuilta löytyy lista valmistajista, jotka tukevat suoraan komponenttikirjaston lisäämistä.

Valmiin komponenttikirjaston käyttö poistaa virhelyöntien mahdollisuuden, kun komponentin suoritustasoa määritellään. Ohjelmisto laskee kaikkien turvalaitteiden kokonaissuoritustason PL-luokan.

Ohjelmistossa on työkalu, jolla voidaan määrittellä koneen turvalaitteiden vaatima suoritustaso PLr. Ohjelmiston laskema turvalaitteiden suoritustason PL-luokan tulee siis olla suurempi tai yhtä suuri kuin koneeseen käsin määritelty suoritustason PLr-taso.

Seuraavaksi selvitetään CNC-jyrsintyöhön tehty Sistema-riskianalyysin PLr-tason määrittely ja määrittelyyn liittyviä muuttujia.

Ensimmäisenä Sistemassa määritellään kyseisen koneen turvalaitteilta vaadittava suoritustaso PLr. PLr-tason määrittelyssä voidaan käyttää kuvassa 19 näkyvää riskigraafia. Turvallisuustason määrittelyssä pitää muistaa, että koneen turvallisuutta arvioidaan ilman turvalaitteita. Määrittelyssä käytetään apuna standardissa SFS-EN ISO 13849-1 löytyviä määrittelyohjeita.

Turvatoiminto

Dokumentaatio PLr PL Alajärjestelmät

Syötä PLr-arvo suoraan
 Määritä PLr-taso riskigraafista

Vaadittava suoritustaso:

Vamman vakavuus (S)

- S1 Lievä (tavallisesti palautuva vamma)
- S2 Vakava (tavallisesti palautumaton vamma tai kuolema)

Taajuus ja/tai altistumisaika vaaralle (F)

- F1 Harvoin tai joskus ja/tai altistumisaika on lyhyt
- F2 Usein tai jatkuvasti ja/tai altistumisaika on pitkä

Mahdollisuus välttää vaaraa tai rajoittaa vahinkoa (P)

- P1 Mahdollista tietyissä olosuhteissa
- P2 Tuskin mahdollista

The risk graph on the left shows a tree structure with nodes S1, S2, F1, F2, P1, and P2. The path leading to S2, F2, and P1 is highlighted in red, corresponding to the selected options in the right-hand list. The final result 'd' is shown in a box at the top right of the graph area.

Kuva 19. Sistema suoritustason riskigraafi.

PLr-tason ensimmäinen määritelmä on vamman vakavuusluokka, joka voi olla S1 tai S2. S1-luokka tulee valita, jos kyseessä on pienet ruhjeet tai lievät vammat. S2-luokka valitaan, jos kyseessä on vakavat vammat, irtileikkautuminen tai kuolema. (SFS-EN ISO 13849-1 2015, 53.) Tähän työhön on valittu S2-luokka, koska koneessa käytetään erilaisia jyrsintyökaluja, joilla tulevat vammat voivat olla vakavia. Myös koneen liikkuvien osien väliin joutuminen on vaarallista.

F-luokassa määritellään, kuinka usein voidaan altistua vaaralle ja mikä on tämän vaaralle altistumisen kesto, vaihtoehtoina ovat joko F1- tai F2-luokat. Jos henkilö joutuu olemaan toistuvasti vaaralle altistuneena, määritellään luokaksi F2. (SFS-EN ISO 13849-1 2015, 53-54.) Tähän työhön on valittu F2-luokka, koska koneistettavia kappaleita joudutaan vaihtamaan CNC-jyrsimeen sekä jyrsintyökalua joudutaan vaihtamaan käsin.

P-luokassa määritellään, voidaanko vaarallista tapahtumaa välttää ja mikä on sen esiintymistodennäköisyys, vaihtoehtoina ovat joko P1- tai P2-luokat. P1-luokka voidaan valita ainoastaan, jos vaarallinen tilanne on todella mahdollista välttää tai sen vaikutusta voidaan merkittävästi vähentää. (SFS-EN ISO 13849-1 2015, 54.) Tähän työhön on valittu P1-luokka, koska vaarallinen tapahtuma voidaan välttää koneen oikeaoppisella käytöllä.

Näiden syötettyjen tietojen perusteella ohjelma näyttää turvalaitteilta vaadittavan suoritustason PLr. Tässä työssä määriteltyjen luokkien perusteella vaadittava suoritustaso PLr on d.

Kun PLr-taso on määritelty, voidaan Sistemaan syöttää turvalaitekomponentit ja näille jokaiselle syötetään kyseisen komponentin suoritustaso. Komponenttivalmistajien sivuilta löytyvät tarvittavat tiedot suoritustason määrittelyyn. Esimerkkinä on kuvassa 20 näkyvä Pilz-turvarele. Turvareleen PL-taso on määritelty valmistajan ohjekirjassa PL e-luokaksi.

Safety characteristic data



NOTICE

You must comply with the safety-related characteristic data in order to achieve the required safety level for your plant/machine.

Operating Mode	EN ISO 13849-1: 2015	EN ISO 13849-1: 2015	EN 62061 SIL CL	EN 62061 PFH _D [1/h]	IEC 61511 SIL	IEC 61511 PFD	EN ISO 13849-1: 2015 T _M [year]
PL	Category						
–	PL e	Cat. 4	SIL CL 3	2,31E-09	SIL 3	2,03E-06	20

All the units used within a safety function must be considered when calculating the safety characteristic data.

Kuva 20. Turvareleen PL-taso valmistajan ohjekirjasta (PILZ 27.6.2017).

Kun kaikki komponentit on lisätty listalle ja näille on määriteltä suoritusastot, ohjelma laskee syötettyjen arvojen perusteella kaikkien turvalaitteiden suoritusastan eli PL-arvon.

Asiayhteys	
Hätäseis Piiri	
PLr	d
PL	d
PFHD [1/h]	1,6E-7

Kuva 21. Sistema riskianalysin lopputulos.

Työssä käytetyillä turvalaitteilla PL-arvoksi muodostui d. Sisteman riskigraafin mukaan, vaadittava suoritusasto PLr tulisi olla vähintään d, joten koneen turvalaitteet ovat tarpeeksi turvalliset kyseiseen laitteeseen.

6 AUTOMAATIOSUUNNITTELU

Tässä osiossa käsitellään kahden erilaisen koneenohjauksen käyttöönottoa sekä automaatisuunnittelun yleisiä ohjelmointikieliä.

6.1 IEC 61131-3 ja PLCopen

IEC 61131-3 on kansainvälinen standardi, joka käsittelee ohjelmointikieltä. Standardin tarkoituksena on muodostaa alalla oleville valmistajille yhtenäinen ohjelmointikieli, mutta standardissa hyväksytään myös valmistajien omien ohjelmointisääntöjen käyttöä. Standardissa käsitellään viittä erilaista ohjelmointikieltä. (Sands & Verhappen 2018, Chapter 16.)

IEC 61131-3 -standardin mukaiset ohjelmointikielet ovat:

- Structured Text (ST)
- Instruction List (IL)
- Function Block Diagram (FBD)
- Ladder Diagram (LD)
- Sequential Function Chart (SFC). (Sands & Verhappen 2018, Chapter 16.)

PLCopen on kansainvälinen organisaatio, jonka tarkoituksena on kehittää ohjelmitaville logiikoille yhtenäisiä ohjelmointisääntöjä ja toimintalohkoja. PLCopen noudattaa IEC 61131-3 -standardin mukaisia sääntöjä ja on kehittänyt esimerkiksi toimintalohkoja liikkeenohjauksiin ja turvatoimintoihin. (PLCopen [Viitattu 25.5.2019].)

6.2 Grbl v1.1

Grbl on Arduino-pohjainen avoimen lähdekoodiin koneenohjausjärjestelmä. Ohjausta voidaan käyttää kolmeakselisen koneen ohjauksessa ja ohjelmisto toimii kai-

kissa Arduinoissa, joissa on a 328p mikroprosessori (Uno, nano). Ohjaus on suunniteltu askelmoottorikäyttöön, Grbl pystyy tuottamaan 30 kHz ohjaussignaalia. Tämä tarkoittaa, että askelmoottoreille voidaan syöttää 30 000 pulssia sekunnissa. (Github [Viitattu 28.4.2019].)

Grbl-järjestelmässä on tuki DIN 66025 -pohjaiseen G- ja M-koodiin. Makrotoiminnoille ja muuttujille ei ole tukea. Tarkoituksena on, että nämä käsiteltäisiin käyttöliittymän puolella. Jokaisesta Grbl-versiosta löytyy lista, jossa esitellään tuetut komennot. (Github [Viitattu: 28.4.2019].)

Ohjauksessa on kiihtyvyydenhallintajärjestelmä, joka lukee NC-ohjelmasta 16:sta liikekomentoa ennen niiden suorittamista ja suunnittelee nopeudet näiden mukaan. Tämä mahdollistaa sulavat liikkeenohjaukset. (Github [Viitattu 28.4.2019].)

Tällä hetkellä version pääkehittäjänä toimii Sungeun Jeon, nykyinen ohjaus on kehitetty Simen Svale Skogsrudin vuonna 2009 aloittamasta Grbl-projektista. Grbl on ilmainen ohjelmisto ja se on julkaistu GPLv3-lisenssin alla. (Github [Viitattu 28.4.2019].)

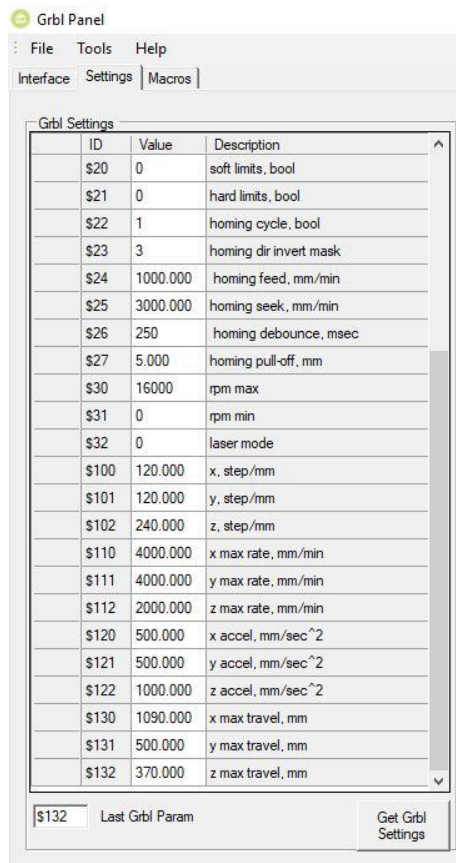
6.2.1 Grbl-ohjauksen käyttöönotto

Grbl-ohjauksen käyttöönotto aloitetaan ohjelmiston latauksella Arduinolle. Ohjelmiston lataukseen löytyy ohjeistukset Grbl-ohjauksen sivustolta. Lataus voidaan suorittaa käyttämällä Arduinon omaa Arduino IDE -ohjelmistoa, jota myös suositellaan käytettäväksi.

Grbl-ohjelmiston koodista löytyy config.h-tiedosto, jota muokkaamalla voidaan vaikuttaa koneen toimintoihin. Tiedoston muokkaaminen ei ole tarpeellista, jos Grbl-ohjausta halutaan käyttää vakioasetuksilla. Yleisempiä asetuksia voidaan muokata myös ohjelmiston latauksen jälkeen. Config.h-tiedostossa määritellään esimerkiksi akselien kotiinajon toimintajärjestys. Tiedostossa on kommentoituna kaikki muutettavat parametrit, joten ohjauksen muokkaaminen koneeseen on helppoa. Tiedoston muokkaaminen tulisi tehdä ennen kuin Grbl-ohjausta ladataan Arduinolle.

Kun config.h-tiedosto on muokattu, voidaan Grbl-ohjauksen koodi ladata Arduinolle käyttämällä Arduino IDE -ohjelmistoa. On suositeltavaa käyttää aina uusinta versiota lataukseen.

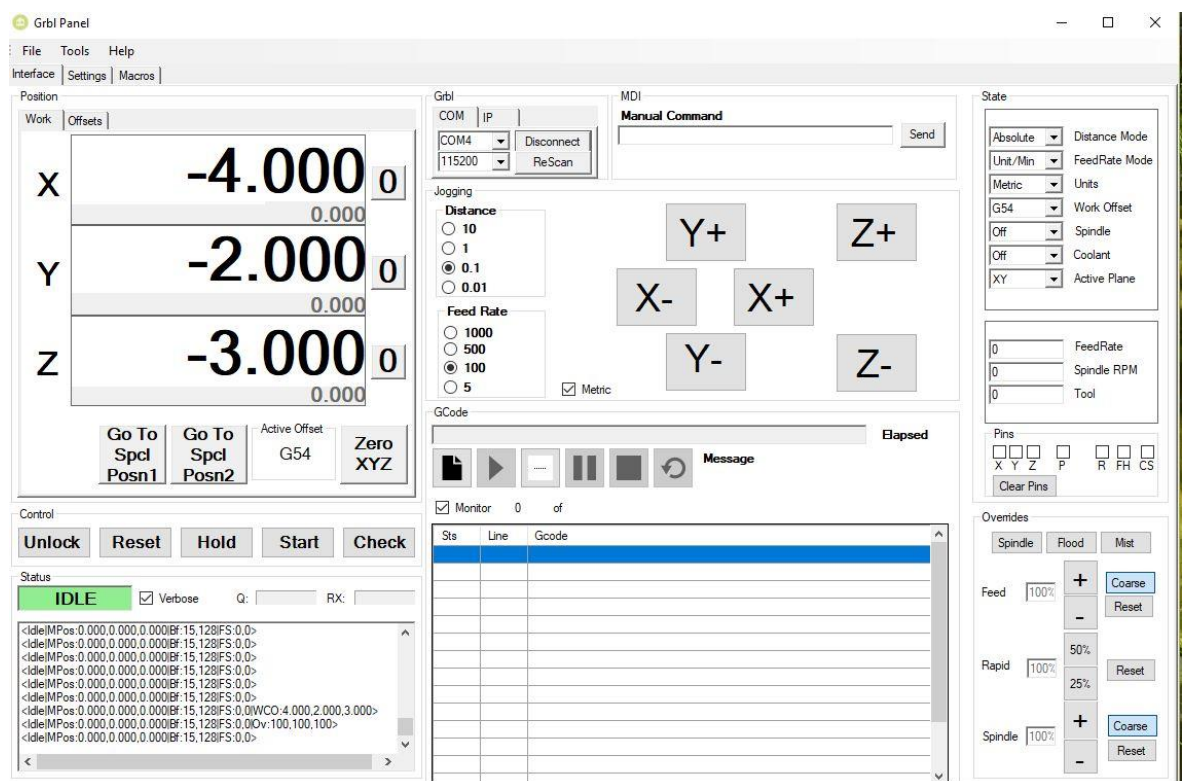
Kun Grbl-ohjauksen koodi on ladattu Arduinolle, voidaan aloittaa koneen asetusten määrittely. Asetusten määrittely on helpointa suorittaa Grbl-ohjaukselle suunniteltujen käyttöliittymien avulla. Käyttöliittymistä löytyy yleensä asetukset-välilehti, jolla koneen asetusten määrittely on helpointa suorittaa. Tärkeimpinä asetuksina on koneen akselien välityssuhteen määrittely, joka on esimerkiksi X-akselille parametri 100. Tällä parametrilla kerrotaan ohjaukselle, kuinka monta pulssia askelmoottorille pitää syöttää, jotta akseli on liikkunut yhden millimetrin. Muita asetuksia ovat esimerkiksi akselien maksiminopeus, kiihtyvyys ja liikealue.



Kuva 22. Grbl-ohjauksen asetusten määrittely Grbl Panel -ohjelmiston avulla.

6.2.2 Grbl-ohjauksen käyttöliittymä

Grbl-ohjaukselle on saatavilla monia valmiita käyttöliittymiä. Tämä on yksi avoimen lähdekoodin aikaansaamia hyviä puolia. Työssä käytettiin GrblPanel-ohjelmistoa. GrblPanel-ohjelmiston ylläpitäjänä toimii Gerrit Visser. Ohjelmistosta löytyy CNC-koneen ohjaukseen tarvittavat toiminnot. GrblPanel-ohjelmalla voidaan ajaa akseleita käsikäytöllä, ladata ohjaukselle NC-ohjelmia, nollata kappaleen työkoordinaattistoja sekä muuttaa Grbl-ohjauksen asetuksia. (Github [Viitattu 1.5.2019].)



Kuva 23. Grbl Panel -käyttöliittymän ulkonäkö (Github [Viitattu 1.5.2019]).

6.3 Beckhoff TwinCAT 3

Beckhoffin valmistama TwinCat 3 -ohjelmisto on tietokonepohjainen reaaliajassa toimiva koneautomaatio-ohjausjärjestelmä. Reaaliajassa toimiva ohjaus tarkoittaa tässä tapauksessa, että ohjelmoinnissa suoritettavia tehtäviä käsitellään järjestelmällisesti ohjelmointilohko kerrallaan. Ohjelmakoodin kiertoaika voi olla jopa 50 us, eli ohjelmakoodi luetaan ja käsitellään uudestaan 50 us:n välein. Kiertoajan pituus

vaikuttaa tietokoneen prosessorin kuormaan, aikaa suurentamalla prosessori-kuorma pienenee. Tämä asia on hyvä muistaa, jos tietokoneessa on vain rajallisesti tehoa. TwinCat 3 tukee myös useamman ytimen käyttöä, joten eri tehtäviä voidaan jakaa ytimien kesken. TwinCat 3 noudattaa edellisen versionsa lailla IEC 61131-3 -standardia ja ohjelmistolla on PLC Open Motion Control hyväksyntä. (Beckhoff 2012.)

Yksi suurempia kehitysaskelaita on kehitysympäristön lisääminen Microsoftin Visual Studio laajennusosaksi. Tämä tarkoittaa, että TwinCat 3 toimii Visual Studio pohjalla ja mahdollistaa näin myös korkeampien ohjelmointikielien käyttöä reaaliaika-ohjauksessa, joita ovat esimerkiksi C, C++, Matlab ja Simulink. (Beckhoff 2012.)

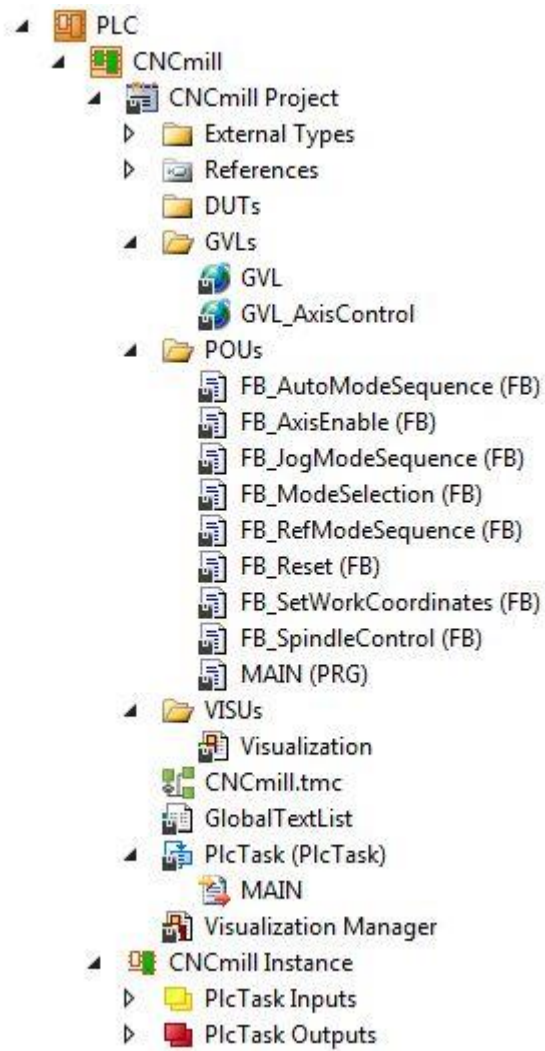
Vanhojen TwinCat 2 -projektien kääntö uudeksi TwinCat 3 -projektiksi onnistuu suoraan ohjelmistolla (Beckhoff 2012).

6.3.1 TC3 PLC

PLC-ohjelmakoodissa käytetään IEC 61131-3 -standardin mukaista ohjelmointikieltä. Opinnäytetyössä on käytetty Structured Text (ST) -ohjelmointikieltä. PLC-ohjelma koostuu MAIN-pääohjelmasta, jossa kutsutaan haluttuja aliohjelmaa (POU). PLC-ohjelmakoodin puolella voidaan suorittaa esimerkiksi toimilaitteiden ohjauksia, nämä ohjaussignaalit voidaan yhdistää TwinCat I/O -puolella fyysisiin lähtöihin, joihin toimilaitte on kytkettynä.

Akselien liikkeenohjaus suoritetaan TwinCat Motion Control -lohkossa, jota käsitellään myöhemmin, mutta PLC-ohjelmakoodin puolella kuitenkin kutsutaan ja hallinnoidaan toimintalohkoja, jotka ohjaavat Motion Control -lohkossa olevia akseleita.

Opinnäytetyössä oleva PLC-lohko näyttää kuvan 24 mukaiselta.



Kuva 24. PLC-lohkon näkymä.

6.3.2 TC3 I/O

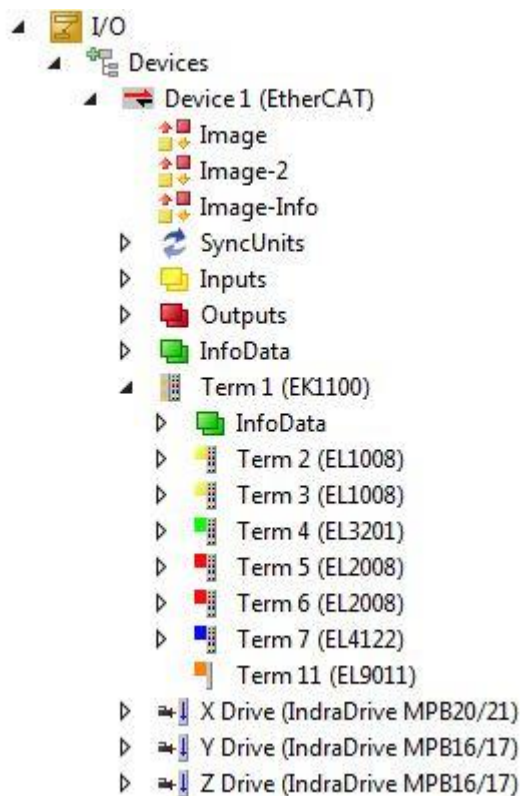
TwinCat 3 I/O -lohkossa muodostetaan yhteys koneessa oleviin sähköisiin komponentteihin. Yhteyden muodostukseen käytetään väylää. TwinCatissä on tuki seuraaville väylille.

- EtherCat
- Lightbus
- Profibus Dp
- Interbus

- CANopen
- SERCOS interface
- DeviceNet
- EtherNet
- USB. (Beckhoff 2012.)

I/O toimii rajapintana sähköisten komponenttien ja PLC-ohjelmakoodin välillä.

Työssä tietokoneen verkkokortti toimii EtherCAT-kenttäväylä-masterina, josta ensimmäisenä väyläkaapeli kulkee EK1100 EtherCAT -reitittimeen. Reitittimessä ovat kiinni tulo- ja lähtökortit, joissa on sisäänrakennettu E-Bus-väylä. EtherCAT-kenttäväylä kulkee myös Bosh Rexrothin IndraDrive CS -servovahvistimiin.

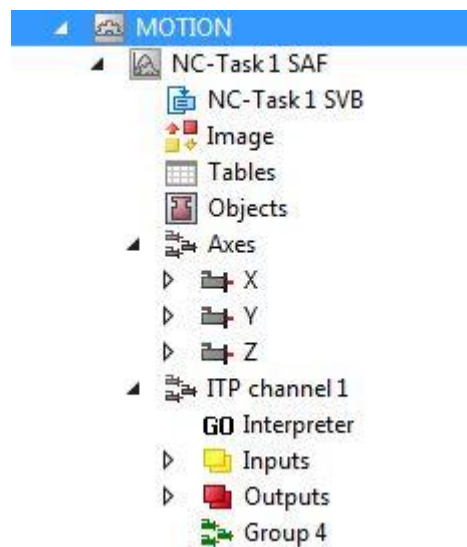


Kuva 25. I/O-lohkon näkymä.

TwinCat I/O-lohkossa on mahdollista seurata sähköisien komponenttien online-tilaa ja tätä kautta on mahdollista pakottaa tuloja/lähtöjä päälle. Tämä toiminto voi olla tarpeellinen, kun laitteistoa halutaan testata.

6.3.3 TC3 Motion control

Liikkeenohjaus voidaan toteuttaa eri tavalla, mutta tässä työssä on käsitelty vain työssä tarvitsemia toimintoja, joita ovat NC PTP ja NC I.



Kuva 26. Motion-lohkon näkymä.

NC PTP (point to point) -toiminto on yksinkertainen liikkeen ohjaus. Akselien liike voidaan suorittaa PLC Open Motion Control -lohkojen avulla PLC-ohjelmakoodissa. Liike tapahtuu alkupisteestä loppupisteeseen, mutta liikerataa näiden pisteiden välillä ei valvota. (Beckhoff [Viitattu 13.4.2019].)

NC I (interpolated motion) -toiminnolla voidaan ohjata useampaa akselia yhdenaikaisesti ja synkronissa toisiinsa nähden. Tämä liikkeenohjaus on välttämätöntä, jos tarkoituksena on muodostaa koneella esimerkiksi pyöreitä tai lineaarisia liikeratoja. Akselien interpolointi ohjelmoidaan pääasiassa DIN 66025 -standardin mukaisella NC-koodilla, jonka interpolaattori kääntää liikkeen ohjauksen komennoiksi. Ohjelmointia tehdään PLC-ohjelmakoodin kautta, haluttuja toimintoja kutsutaan silloin, kun niitä tarvitaan. (Beckhoff [Viitattu 14.4.2019].)

Työssä käytetään PTP-toimintoa akselien käsinajoon. Käsinajo suoritetaan valmiilla FB_JOG-toimilohkoilla, joita käytetään FB_JogModeSequence-aliohjelmassa. FB_JogModeSequence-koodissa valvotaan ajettavan akselin valintaa sekä sen liikkuttamista käsinajolla.

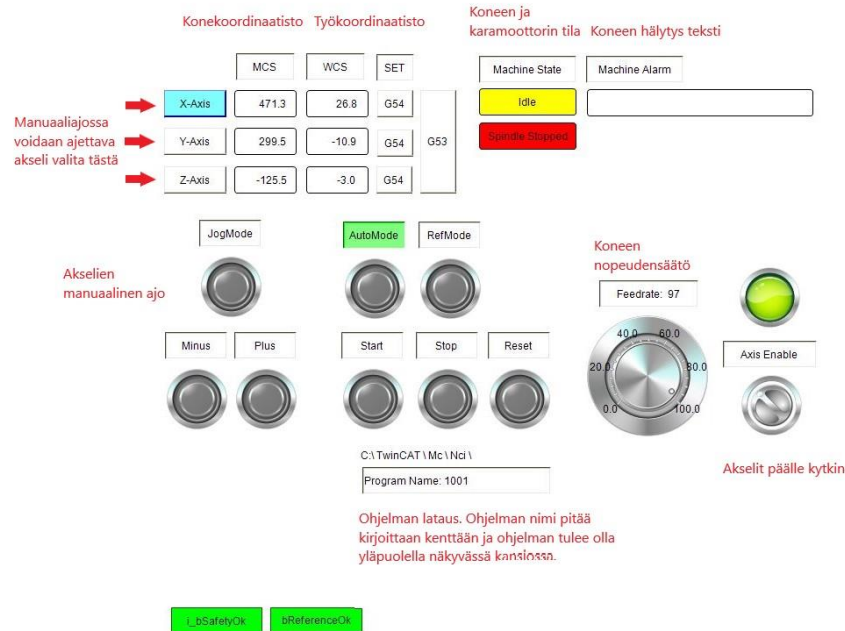
NC I -interpolaattoria käytetään työssä automaattiseen ajoon. Interpolaattoriin syötetään NC-koodia, jonka komentoja koneen akselit seuraavat. PLC-koodissa olevassa FB_AutoModeSequence-aliohjelmassa suoritetaan akselien interpoloinnin vaatimat toimenpiteet. Tässä aliohjelmassa yhdistetään ajettavat akselit, luetaan mahdollinen kappalekoordinaatisto, ladataan NC-ohjelma interpolaattoriin ja suoritetaan NC-ohjelma.

G54-kappalekoordinaatistoa ja M-koodeja joudutaan käsittelemään erikseen PLC-koodin puolella. TwinCat 3 -interpolaattori ymmärtää suoraan sinne lähetetyt G-koodit, mutta M-koodit joudutaan käsittelemään itse.

Työssä kappalekoordinaatiston hallinta toimii niin, että akselit ajetaan työstettävän kappaleen origo-pisteeseen käsinajolla ja käyttöliittymästä painetaan G54-kappalekoordinaatiston asetuspainiketta. Tämä toiminta kirjoittaa konekoordinaattipoikkeaman muistiin ja tämä luetaan FB_AutoModeSequence-aliohjelmassa, ennen kuin automaattiohjelmaa aloitetaan ajamaan. G54-kappalekoordinaatiston hallinta tapahtuu FB_SetWorkCoordinates-aliohjelmassa.

M-koodien käsittely tapahtuu myös FB_AutoModeSequence-aliohjelmassa. M-koodien käsittely suoritetaan valmiilla ltpGetHskMFunc-toimilohkolla, joka poimii NC-koodista M-koodin tunnuksen ja tätä vertailemalla pystytään tekemään haluttuja toimintoja esimerkiksi M3-komennon tullessa käynnistetään jyrsimen karamoottori myötäpäivään, M4-komennolla vastapäivään ja M5-komennolla pysäytetään karamoottori.

6.3.4 TC3-koneen käyttöliittymä



Kuva 27. TwinCat 3 visualization -käyttöliittymä.

Työssä CNC-jyrsimen käyttöliittymä on tehty TwinCatin omalla visualization-toiminnolla. Tämä sen takia, että koneen testaus on saatu tehtyä nopeammin. Myöhemmin on tarkoituksena tehdä käyttöliittymä käyttäen Microsoftin Visual Studiota.

Käyttöliittymästä löytyy koneen testaukseen tarvittavat toiminnot. Käyttöliittymästä näkee X-, Y- ja Z- akselien konekoordinaatiston ja sen vierellä työkoordinaatiston, joka voidaan nollata G54 Set -napilla. Koneen tämänhetkisen tilan näkee Machine Staten alapuolelta ja sen vierellä näkyvät koneen hälytykset.

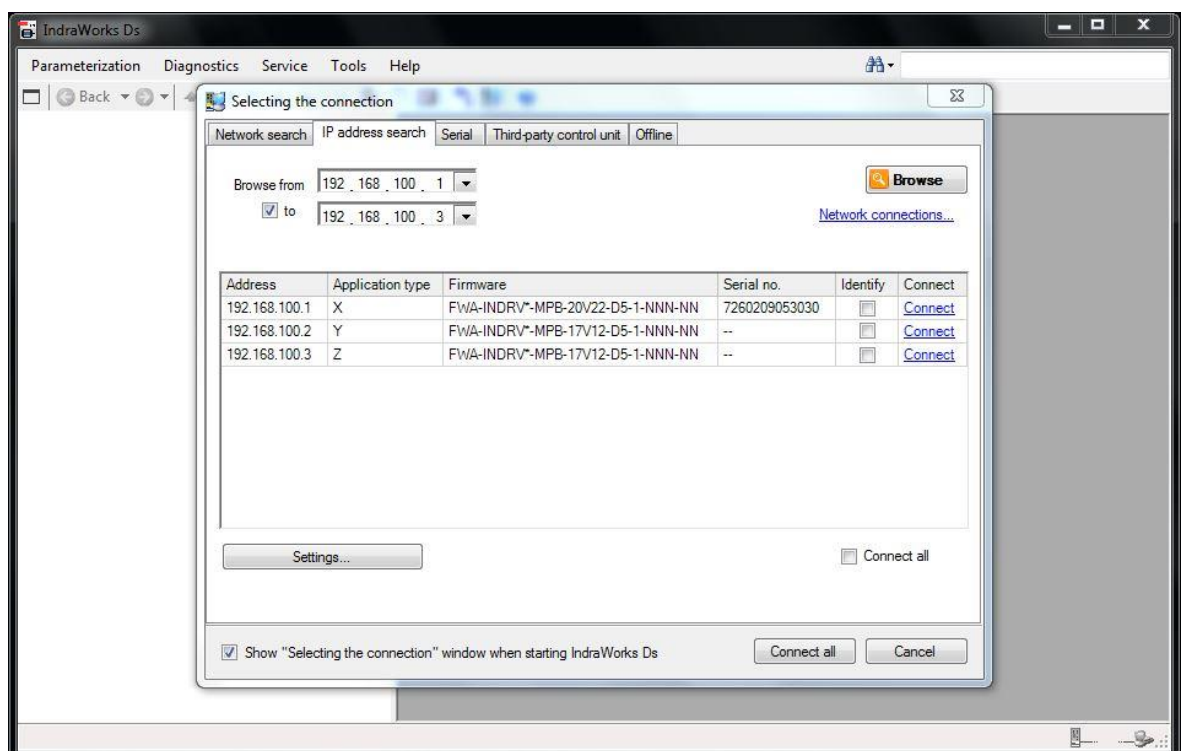
Kone voidaan vaihtaa käsikäyttötilaan JogMode-painikkeella ja tämän jälkeen voidaan valita liikuteltava akseli konekoordinaatiston vasemmalta puolelta. NC-ohjelmien ajo toimii, kun automaattitila on valittu AutoMode-painikkeella. Ajettava NC-ohjelma tulee siirtää C:\TwinCat\Mc\Nci-kansioon. Tämä ohjelman nimi pitää kirjoittaa Program Name -kenttään. RefMode-painikkeella saadaan kone referenssitilaan,

jossa koneen akselit voidaan ajaa hallitusti opetusrajoille ja konekoordinaatti kirjoitetaan uudestaan nollassa.

Axis Enable -kytkimellä annetaan ohjaukselle lupa laittaa akselit ajovalmiuteen. Feedrate-potentiometrillä voidaan valita koneen ajonopeus prosentteina maksiminopeudesta. Reset-painike nolaa koneen hälytykset ja meneillään olevat komennot. Stop-painike pysäyttää koneen liikkeen. Start-painike aloittaa NC-ohjelman automaattisen ajon.

6.4 IndraDrive CS -vahvistimien ja -servomoottorien käyttöönotto

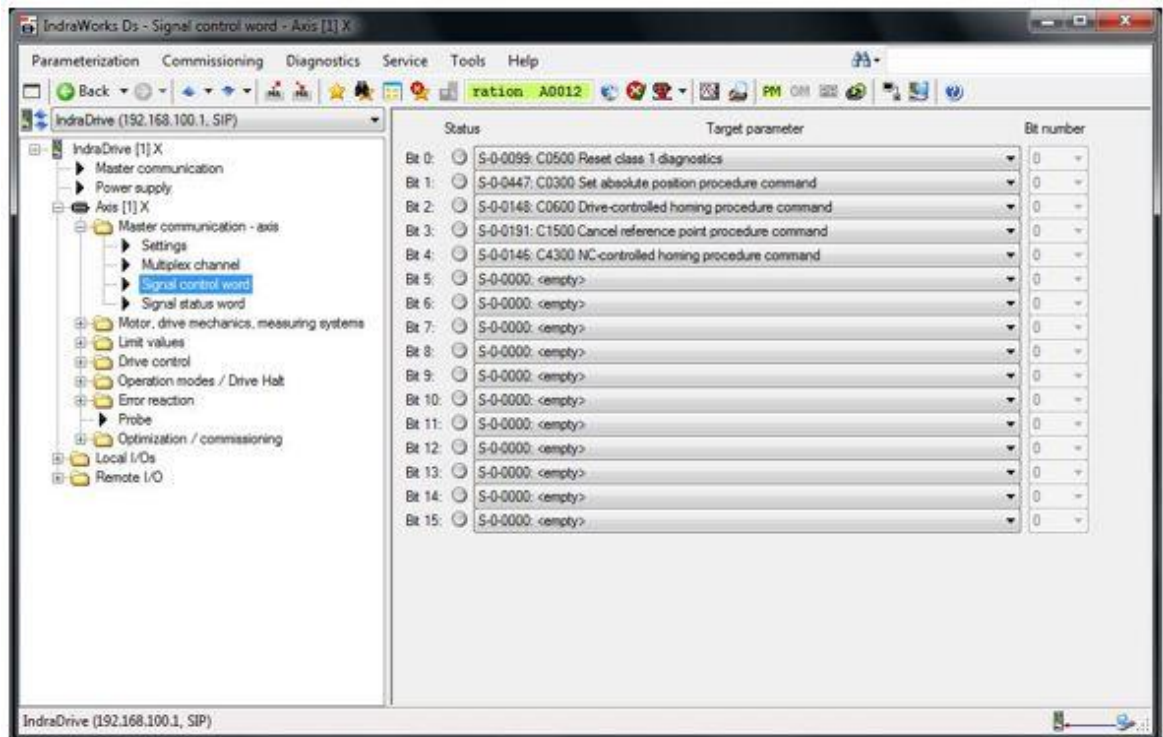
Vahvistimille on määritelty TwinCatin puolella IP-osoitteet, joihin IndraWorks-ohjelmistolla voidaan ottaa yhteys. Kuvasta 28 voidaan nähdään koneessa olevien vahvistimien yhteyden muodostus IP-osoitteen kautta.



Kuva 28. Yhteyden muodostaminen IndraDrive Cs -vahvistimille.

Kun yhteys on muodostettu vahvistimeen, pitää sieltä tarkastaa muutamia asioita. Vahvistimen Signal Control Word -asetuksissa määritellään, minkälaisia komentoja

PLC-ohjelmakoodin puolelta voidaan vahvistimille lähettää. Kuvasta 29 näkyy X-akselin vahvistimen asetukset, sekä PLC-ohjelmakoodin puolelta nollaussignaalin lähetys vahvistimelle, jos Reset-painiketta painetaan.



```
// Drive reset
// Sends signal to the Drives and resets possible alarms
out_XaxisSignalControlWord = 0 .0 FALSE := i_bResetPB FALSE;
out_YaxisSignalControlWord = 0 .0 FALSE := i_bResetPB FALSE;
out_ZaxisSignalControlWord = 0 .0 FALSE := i_bResetPB FALSE;
```

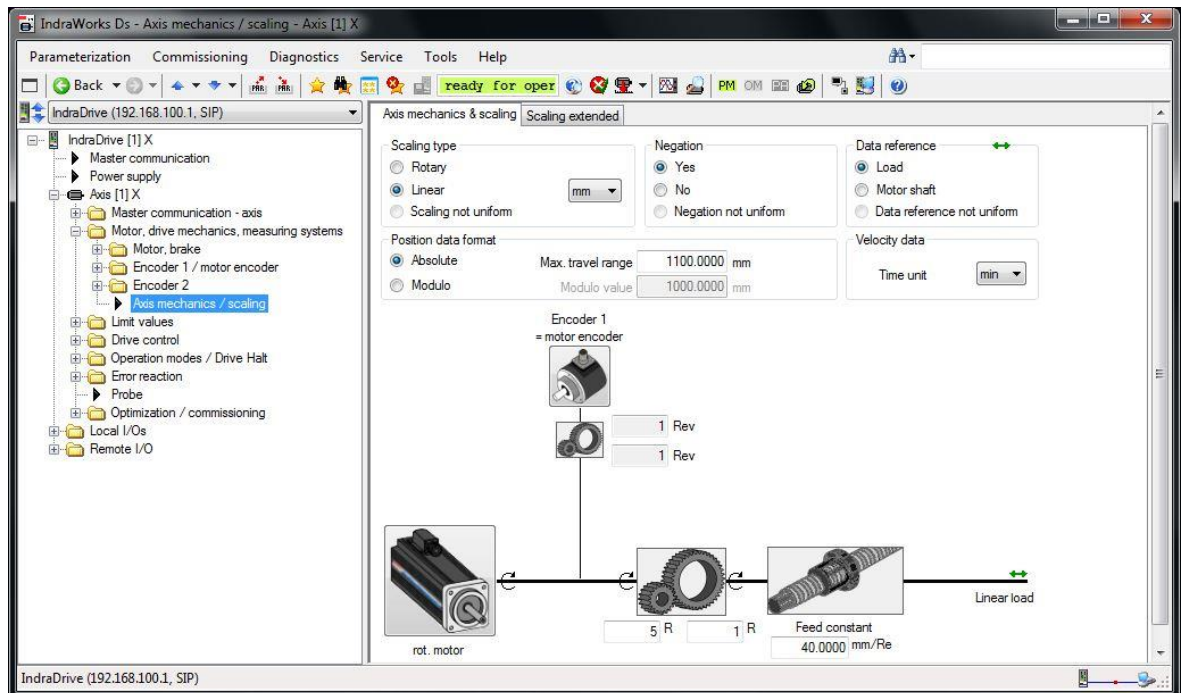
Kuva 29. Kommunikointi vahvistimien ja ohjauksen välillä.

Esimerkkinä PLC-ohjelmakoodissa oleva Signal Control Word: Bit 0 käynnistää vahvistimen puolella C0500-komennon, joka nolaa vahvistimessa olevat hälytykset. Tätä nollauskomentoa voidaan nyt ohjata PLC-ohjelmakoodin puolelta lähettämällä Bit 0 -signaali vahvistimelle.

Vahvistimen Signal Status Word -asetuksia ei ole muokattu, mutta tätä kautta vahvistimilta lähtee TwinCat-ohjauksen puolelle esimerkiksi akselien paikkatieto.

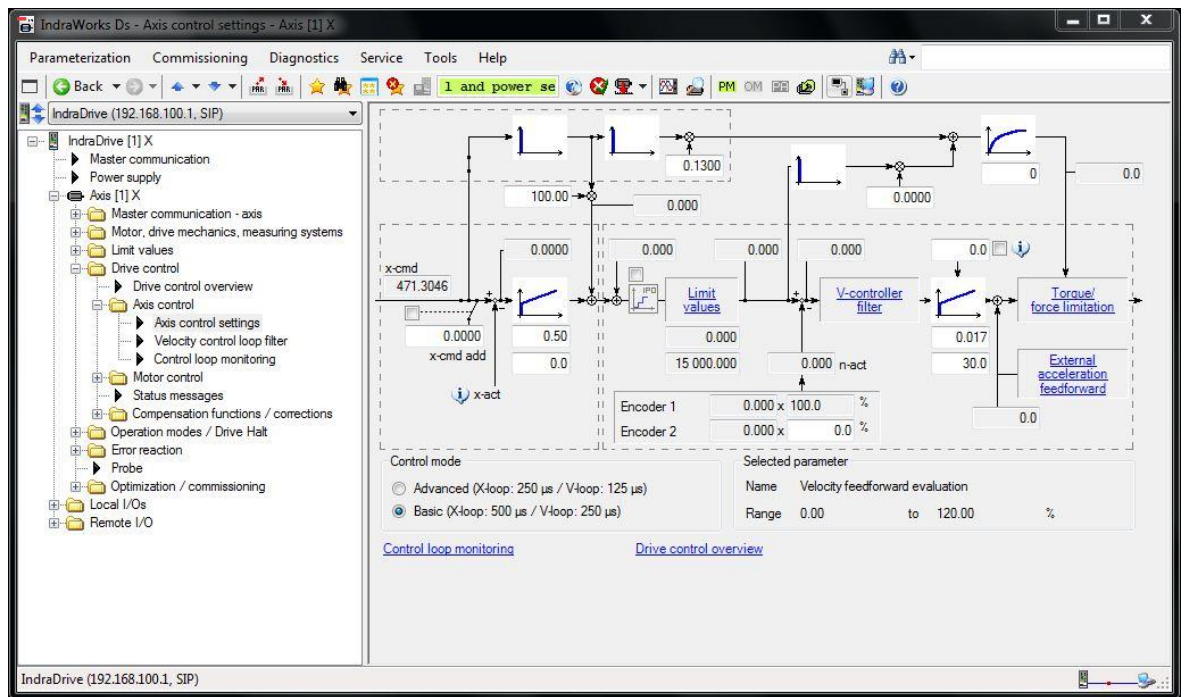
Akselien mekaniikan ja skaalauksen säädöt pitää myös määrittellä. Axis mechanics / Scaling -asetuksissa on määriteltä, että akseli on lineaarisesti liikkuva ja paikkatieto

ilmoitetaan absoluuttisesti. Akselin maksimiliikealue on 1100 mm. Moottorin ja kuularuuvin välityssuhde on 5:1, kuularuuvin nousu on 40 mm/kierros. Nämä arvot vaaditaan, että moottorissa olevan pulssianturin tieto voidaan käsitellä oikein ja akselin liike vastaa todellista liikettä. Työn asetukset näkyvät kuvassa 30.



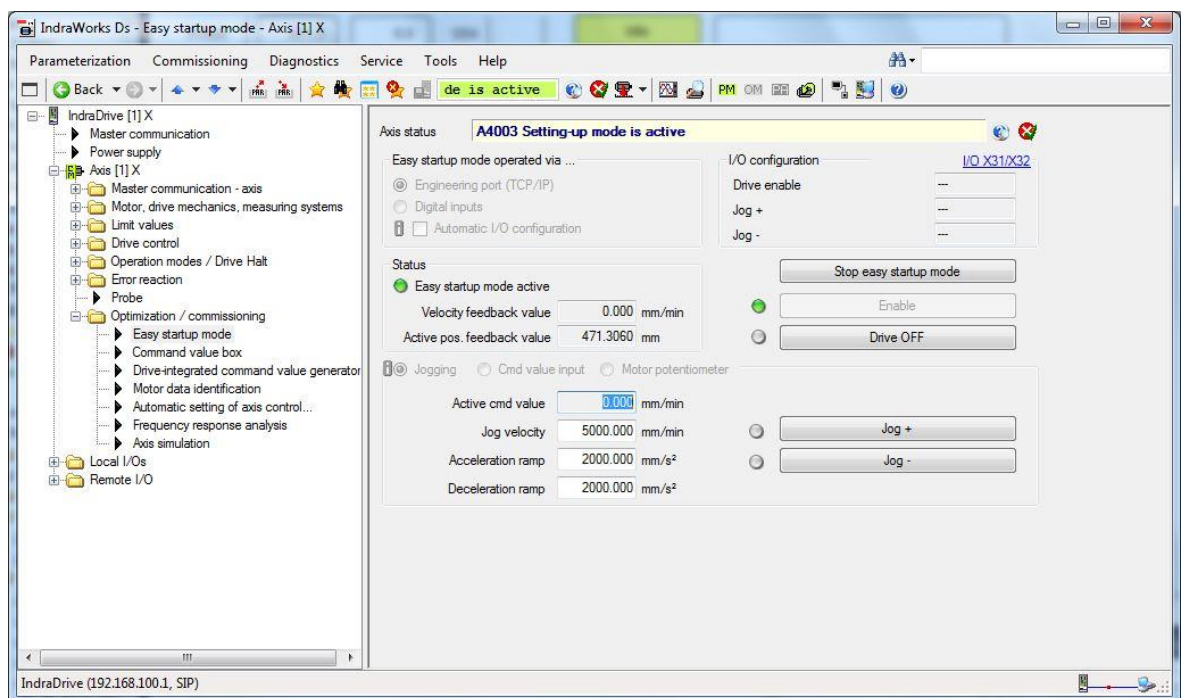
Kuva 30. Akselien mekaniikan ja skaalauksen asetukset.

Axis control setting -asetuksissa säädetään moottorin säätöpiiriä. Säättöpiiri-asetukset vaikuttavat moottorin käyttäytymiseen ja reagointiaikaan. IndraWorks-ohjelmistossa on automaattinen toiminto näiden arvojen saamiseksi.



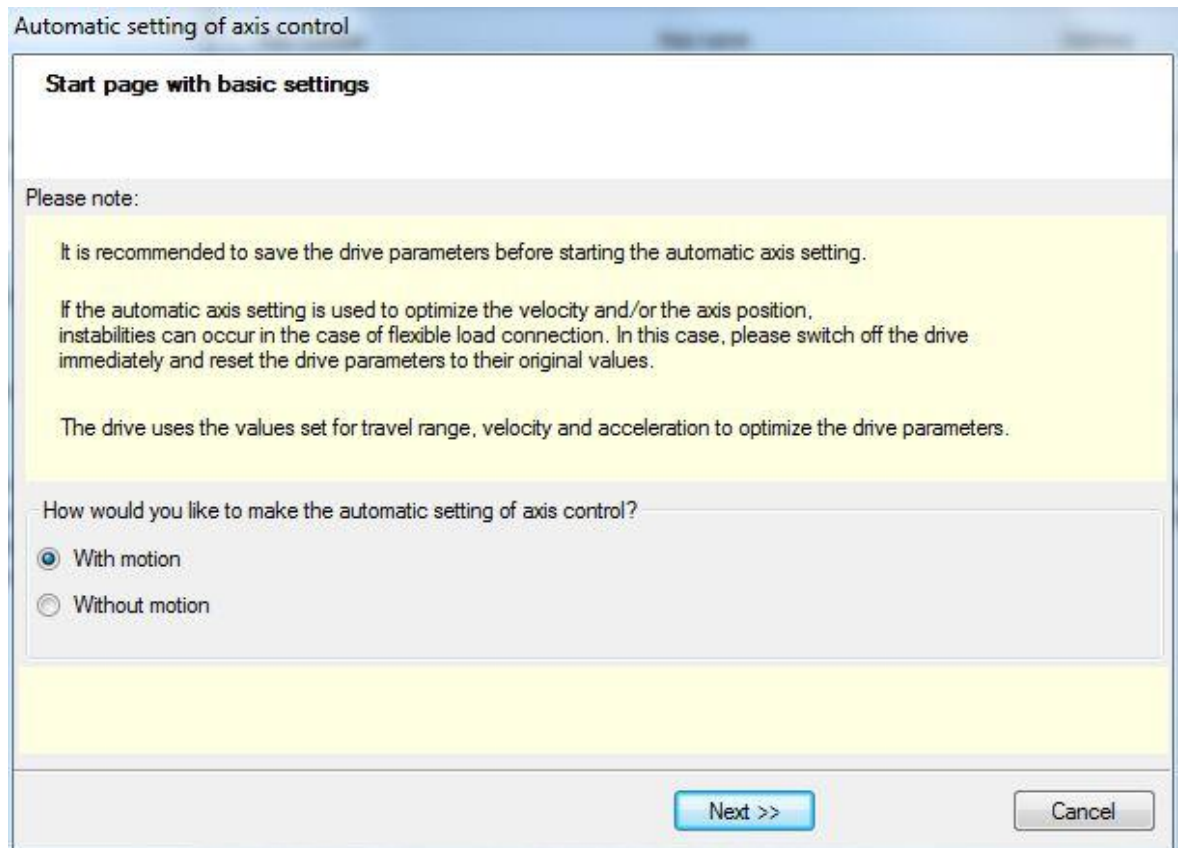
Kuva 31. Akselien säätöpiiriasetukset.

Säätöpiirin automaattisen toiminnon aloittaminen vaatii, että vahvistin laitetaan Easy Startup -tilaan. Tämän toiminnon saa aktivoitua Optimization / commissionin -asetusten alta. Easy Startup mode asettaa vahvistimen ajokuntoon testausta varten.



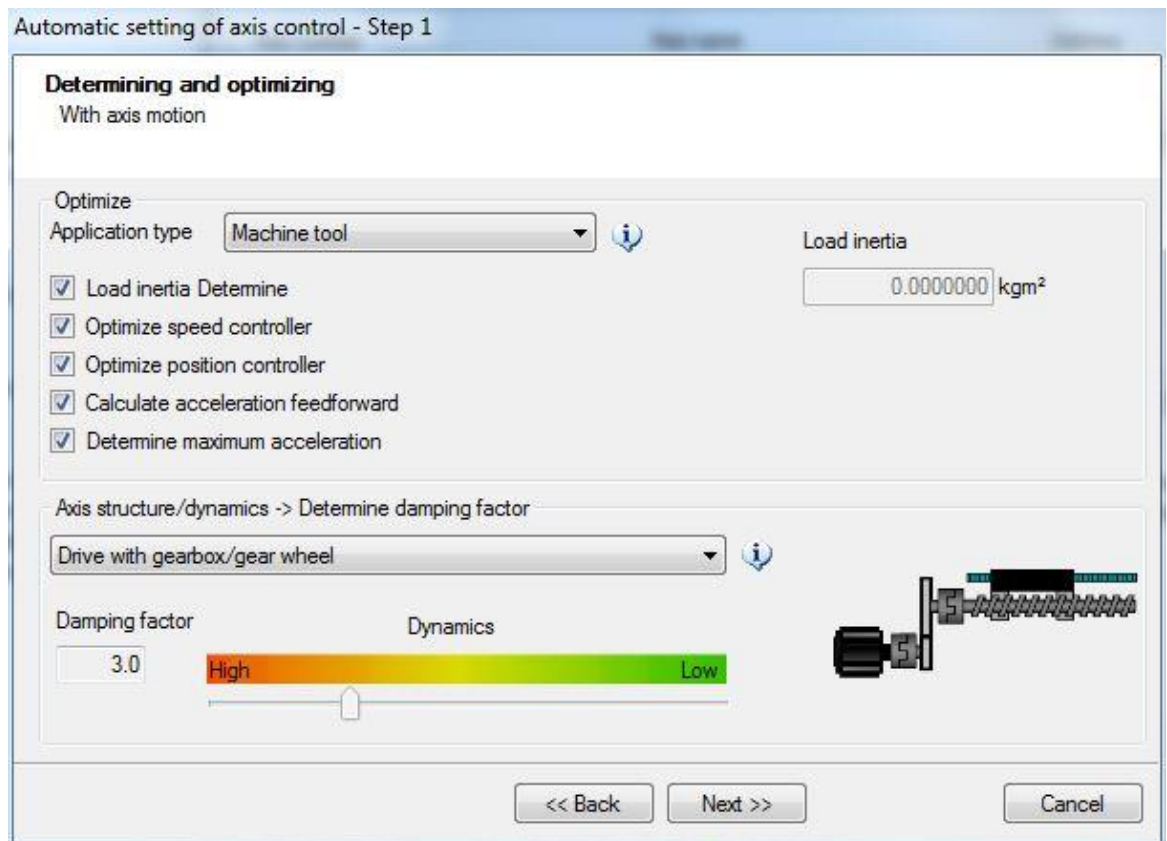
Kuva 32. Servovahvistimen asettaminen Easy Startup -tilaan.

Kun vahvistin on Easy Startup -moodissa, voidaan Automatic setting of axis control -valinnalla aloittaa akselin säätöpiiriasetusten automaattinen määrittely. Testauksen voi suorittaa niin, että kone liikkuu tai on paikallaan. Testaus suoritettiin koneen liikkeen avulla.



Kuva 33. Akselin säätöpiirin automaattisen määrittelyn aloittaminen.

Seuraavalla välilehdellä voidaan valita, mitä arvoja halutaan testata. Kuvassa 34 kaikki testit on valittu ja akselin rakenteeseen on valittu hihnavälitys.



Kuva 34. Akselin säätöpiirin automaattisen määrittelyn ensimmäinen vaihe.

Liikkeen maksimiarvoksi asetettiin 200 mm. Koneen liikkeen nopeudeksi määriteltiin 5000 mm/min ja kiihtyvyydeksi 1000 mm/s². Tämän jälkeen Start optimization -painikkeella kone aloittaa 200 mm:n suuruisen liikkeen edestakaisin ja lisäksi automaattisesti määrittelee akselin säätöpiiriarvot, jotka tulevat heti voimaan Finish-painikkeella.

Automatic setting of axis control - Step 2

Determining and optimizing - Expert mode

With axis motion

Travel profile

Specify the travel limits
 Specify the travel range

Oscillating motion

Travel distance: 200 mm

End position neg.: 286.1610 mm

End position pos.: 312.9280 mm

Actual position: 299.5446 mm

Specify the range in which the axis can move freely when the command is executed. The lower or upper limit is automatically calculated according to the current position and the specified travel range.

Attention: To execute the command, the minimum travel range has to be 26.6667 mm!

<< Back Next >> Cancel

Kuva 35. Akselin säätöpiirin automaattisen määrittelyn toinen vaihe.

Automatic setting of axis control - Step 3

Calculate and show result

With axis motion

Start optimization

Drive enable

Clear error

Drive optimization, velocity: 5000.000 mm/min

Feedrate override: 100.00 %

Drive optimization, acceleration: 1000.000 mm/s²

Result

bxtdn	Name	Value (before)	Value (after)	Unit
P-0-4010.0.0	Load inertia	0.0000589	0.0000644	kgm ²
S-0-0100.0.0	Velocity loop proportional gain	0.042	0.044	Nm/(rad/s)
S-0-0101.0.0	Velocity loop integral action time	10.5	10.5	ms
P-0-0004.0.0	Velocity loop smoothing time constant	300	300	us
S-0-0104.0.0	Position loop Kv-factor	1.42	1.42	1000/min
S-0-0348.0.0	Acceleration feedforward gain	0.0889	0.0944	mNm/(rad/s ²)
P-0-0168.0.0	Maximum acceleration to be parameterized	51521.968	48518.620	mm/s ²

<Back> does not mean that the controller parameters are reset!

<< Back Finish Cancel

Kuva 36. Akselin säätöpiirin automaattisen määrittelyn kolmas ja viimeinen vaihe.

7 POHDINTA

Opinnäytetyön tuloksena syntyi toimiva CNC-jyrsin, jonka tarkkuus ylitti omat vaatimukset. Koneella on mahdollista myös koneistaa terästä, mutta rungossa oli havaittavissa värinää. Rungon täyttäminen esimerkiksi hiekalla voisi vaimentaa teräksen koneistamisesta johtuvia värinöitä, koska rungon massaa saisi näin kasvatettua.

Grbl-ohjaus osoittautui erittäin toimintakykyiseksi ja tämän käyttöönotto oli helppoa. Ohjauksesta oli kytkentäesimerkkejä ja ohjeistuksia hyvin saatavilla. Tämä oli ensimmäinen kerta, kun tämän työn tekijä käytti askelmoottoreita liikkeenohjaukseen ja ainoana huonona puolena tuli askelmoottoreista tulevaa värinää. Värinää on kuitenkin mahdollista pienentää muuttamalla askelmoottorihjaimen mikroaskellusasetuksia. Mikroaskellusasetuksen kasvattaminen pienentää yhden askeleen aikaansaamaa liikettä, joten askelmoottorin sisällä oleva hammastus ei muodosta niin suurta värinää. Mikroaskelluksen kasvattaminen pienentää askelmoottorin vääntöä, mutta kasvattaa samalla koneen tarkkuutta.

TwinCat 3 -ohjauksen kanssa oli enemmän haasteita. Tämä ohjaus ei ole hintansa puolesta harrastajien käytössä, minkä takia ongelmatilanteista oli hankala löytää verkosta apua. Ainoa tiedonlähde oli oikeastansa Beckhoffin kotisivut, sieltä etsittiin tietoa. Suurimmat haasteet olivat TwinCatin ja IndraDrive-servovahvistimien välisessä kommunikoinnissa, johon löydettiin lopulta apuja valmistajien ohjeista. Yksi servomoottori oli liian uusi, joten vanhan mallinen servovahvistin ei tunnistanut moottoria. Uudemmallalla vahvistimella saatiin moottori toimimaan. TwinCat-ohjaukseen piti itse ohjelmoida koneenohjaus ja käyttöliittymä, joten tämä toi lisää haastetta projektiin. Toisaalta tämä mahdollistaa sen, että koneen saa toimimaan juuri niin kuin itse sen haluaa toimivan. Tulevaisuudessa koneen virallinen käyttöliittymä olisi tarkoitus tehdä Microsoftin Visual Studio -ohjelmistolla.

Projekti oli kokonaisuudessaan erittäin opettavainen ja sain haluamaani kokemusta koneensuunnittelusta. Autodesk Fusion 360 -ohjelmiston käytön suoritin itseopiskeluna ja tämän ohjelmiston CAM-ominaisuus tulee tarpeeseen tulevia koneistusprojekteja ajatellen.

LÄHTEET

- Autodesk. Ei päiväystä. Fusion 360. [Verkkosivu]. [Viitattu 10.5.2019]. Saatavana: <https://www.autodesk.com/products/fusion-360/>
- Beckhoff. 2012. TwinCAT 3: eXtended Automation (XA). [Verkkójulkaisu]. [Viitattu 12.04.2019]. Saatavana: https://download.beckhoff.com/download/document/catalog/Beckhoff_TwinCAT3_042012_e.pdf
- Beckhoff. Ei päiväystä. TwinCAT NC PTP: Point-to-point axis positioning. [Verkkójulkaisu]. [Viitattu 13.04.2019]. Saatavana: https://download.beckhoff.com/download/Document/Catalog/Main_Catalog/english/separate-pages/TwinCAT/TwinCAT_NC_PTP.pdf
- Beckhoff. Ei päiväystä. TwinCAT NC I. [Verkkójulkaisu]. [Viitattu 14.04.2019]. Saatavana: https://download.beckhoff.com/download/Document/Catalog/Main_Catalog/english/separate-pages/TwinCAT/TwinCAT_NC_I.pdf
- Boschrexroth. Ei päiväystä. IndraDrive Cs HCS01. [Verkkosivu]. [Viitattu 14.5.2019]. Saatavana: <https://www.boschrexroth.com/en/xc/products/product-groups/electric-drives-and-controls/servo-drives/compact-converter/hcs01>
- Boschrexroth. Ei päiväystä. Mains choke. [Verkkosivu]. [Viitattu 15.5.2019]. Saatavana: <https://www.boschrexroth.com/en/xc/products/direct-entry?cat=Electric-Drives-and-Controls-Catalog&language=&m=XC&u=si&o=portal&p=p790617>
- Boschrexroth. 11.3.2013. Rexroth IndraDrive Drive Controllers Power Sections HCS01: Operating instructions. [Verkkójulkaisu]. [Viitattu 16.5.2019]. Saatavana: https://md.boschrexroth.com/modules/BRMV2PDFDownload-internet.dll/R911339012_01.pdf?db=brmv2&lvid=1171535&mvid=14006&clid=20&sid=C7439809BA597039F37F38D496D27305.borex-tc&sch=M&id=14006,20,1171535
- Bralla, James G. 2007. Handbook of Manufacturing Processes - How Products, Components and Materials are Made. [Verkkokirja]. Saatavana Knovel-palvelusta.
- Eplan. Ei päiväystä. Eplan platform. [Verkkosivu]. [Viitattu 12.5.2019]. Saatavana: <https://www.eplan.fi/fi/ratkaisut/tuotteen-yleiskuvaus/eplan-platform/>
- Github. Ei päiväystä. Grbl. [Verkkosivu]. [Viitattu 28.4.2019]. Saatavana: <https://github.com/gnea/grbl>
- Github. Ei päiväystä. Grbl wiki. [Verkkosivu]. [Viitattu 29.4.2019]. Saatavana: <https://github.com/gnea/grbl/wiki>

- Github. Ei päiväystä. Compiling Grbl. [Verkkosivu]. [Viitattu 30.4.2019]. Saatavana: <https://github.com/gnea/grbl/wiki/Compiling-Grbl>
- Github. Ei päiväystä. Grbl Panel. [Verkkosivu]. [Viitattu 1.5.2019]. Saatavana: <https://github.com/Gerritv/Grbl-Panel/wiki>
- Github. Ei päiväystä. Connecting Grbl. [Verkkosivu]. [Viitattu 2.5.2019]. Saatavana: <https://github.com/gnea/grbl/wiki/Connecting-Grbl>
- Hiwin. Ei päiväystä. Linear guideways. [Verkkosivu]. [Viitattu 2.5.2019]. Saatavana: https://www.hiwin.com/pdf/linear_guideways.pdf
- Hiwin. Ei päiväystä. Ballscrews. [Verkkosivu]. [Viitattu 3.5.2019]. Saatavana: <https://www.hiwin.com/pdf/ballscrews.pdf>
- IFA. Ei päiväystä. Sistema: Safety Integrity Software Tool for the Evaluation of Machine Applications. [Verkkosivu]. [Viitattu 15.5.2019]. Saatavana: <https://www.dguv.de/ifa/praxishilfen/practical-solutions-machine-safety/software-sistema/index.jsp>
- Kandray, Daniel E. 2010. Programmable Automation Technologies - An Introduction to CNC, Robotics and PLCs. [Verkkokirja]. Saatavana Knovel-palvelusta.
- Maaranen, K. 2012. Koneistus. Helsinki: Sanoma Pro Oy.
- Opetushallitus. Ei päiväystä. NC ohjelman rakenne ja NC koodit. [Verkkosivu]. [Viitattu: 25.4.2019]. Saatavana: <http://www03.edu.fi/oppimateriaalit/nctekniikka/NCkoodit.html>
- PILZ. 27.6.2017. PNOZ X2.8P. [Verkkosivu]. [Viitattu 31.5.2019]. Saatavana: https://www.pilz.com/download/open/PNOZ_X2_8P_Operat_Manual_1004082-EN-19.pdf
- Pikkarainen, E. & Mustonen, M. 2010. Numeerisesti ohjatut työstökoneet. 2. uud. p. Tampere: Juvenesprint Oy.
- PLCopen. Ei päiväystä. PLCopen [Verkkosivu]. [Viitattu 25.5.2019]. Saatavana: <https://www.plcopen.org/>
- Sands, Nicholas P. & Verhappen, Ian. 2018. Guide to the Automation Body of Knowledge (3rd Edition) [Verkkokirja]. Saatavana Knovel-palvelusta.
- Siemens. Ei päiväystä. [Verkkosivu]. [Viitattu 25.5.2019]. Saatavana: https://cache.industry.siemens.com/dl/files/039/28739039/att_54134/v1/BASsl_0108_fi.pdf

SFS-EN ISO 13849-1. 2015. Koneturvallisuus: Turvallisuuteen liittyvät ohjausjärjestelmien osat: Osa 1: Yleiset suunnitteluperiaatteet. Helsinki: Suomen Standardisoimisliitto.

Vesämäki, H. 2014. Lastuavan työstön nc-ohjelmointi. 4. uud. p. Tampere: Tammerprint Oy.

Wantmotor. 13.5.2016. Stepper Motor Basics. [Verkkosivu]. [Viitattu 28.4.2019]. Saatavana: <http://www.wantmotor.com/article/20160513/484.html>

LIITTEET

Liite 1. Keskeisimmät sähkökuvat TwinCat 3 ohjaukseen

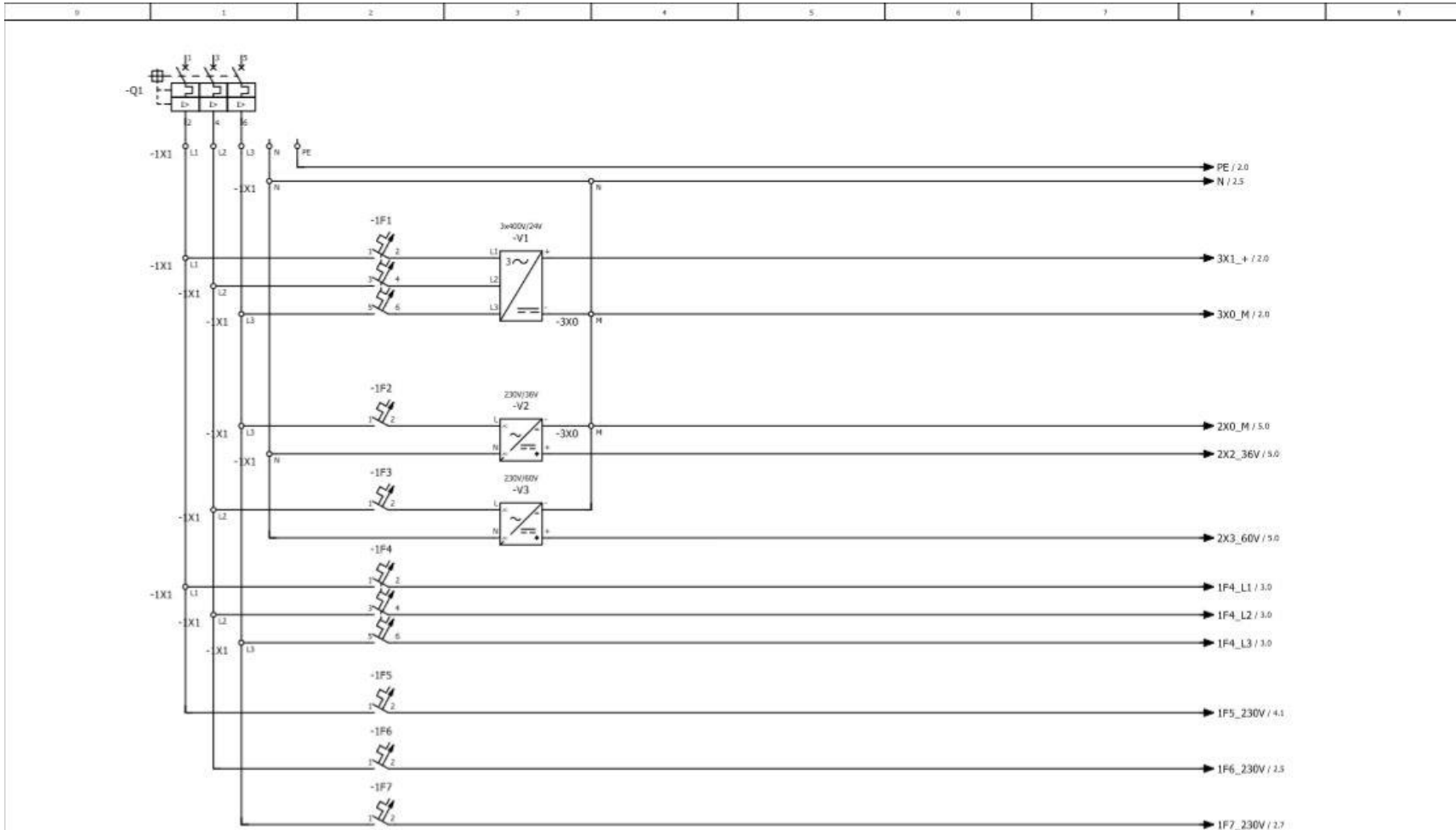
Liite 2. Piirikaaviokuva askelmoottoreista ja askelmoottoriohjaimista

Liite 3. CNC-jyrsimen TwinCat 3 MAIN-ohjelmakoodi

Liite 4. CNC-jyrsimen TwinCat 3 FB_SetWorkCoordinates-ohjelmakoodi

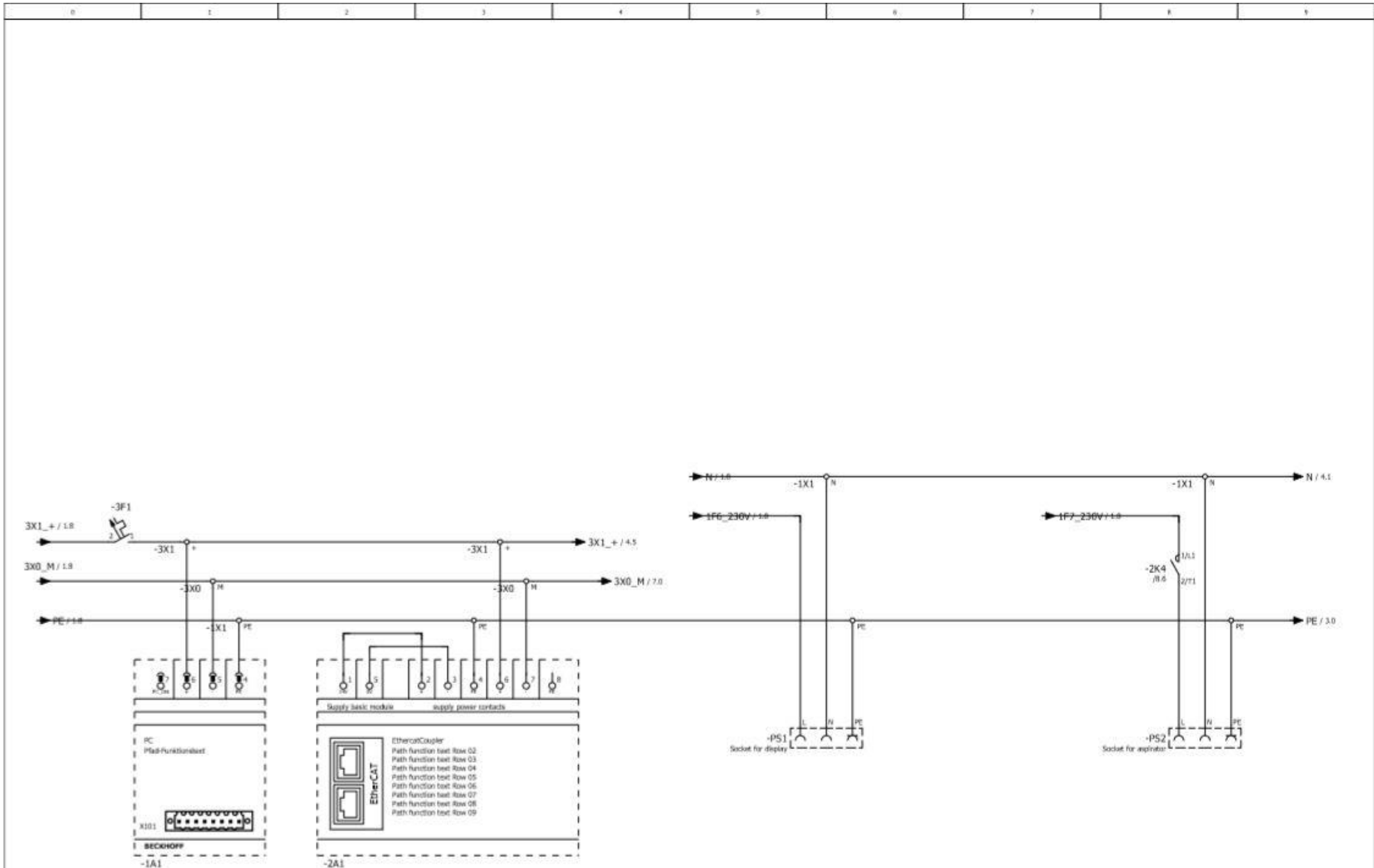
Liite 5. CNC-jyrsimen TwinCat 3 FB_AutoModeSequence-ohjelmakoodi

Liite 1. Keskeisimmät sähkökuvat TwinCat 3 ohjaukseen

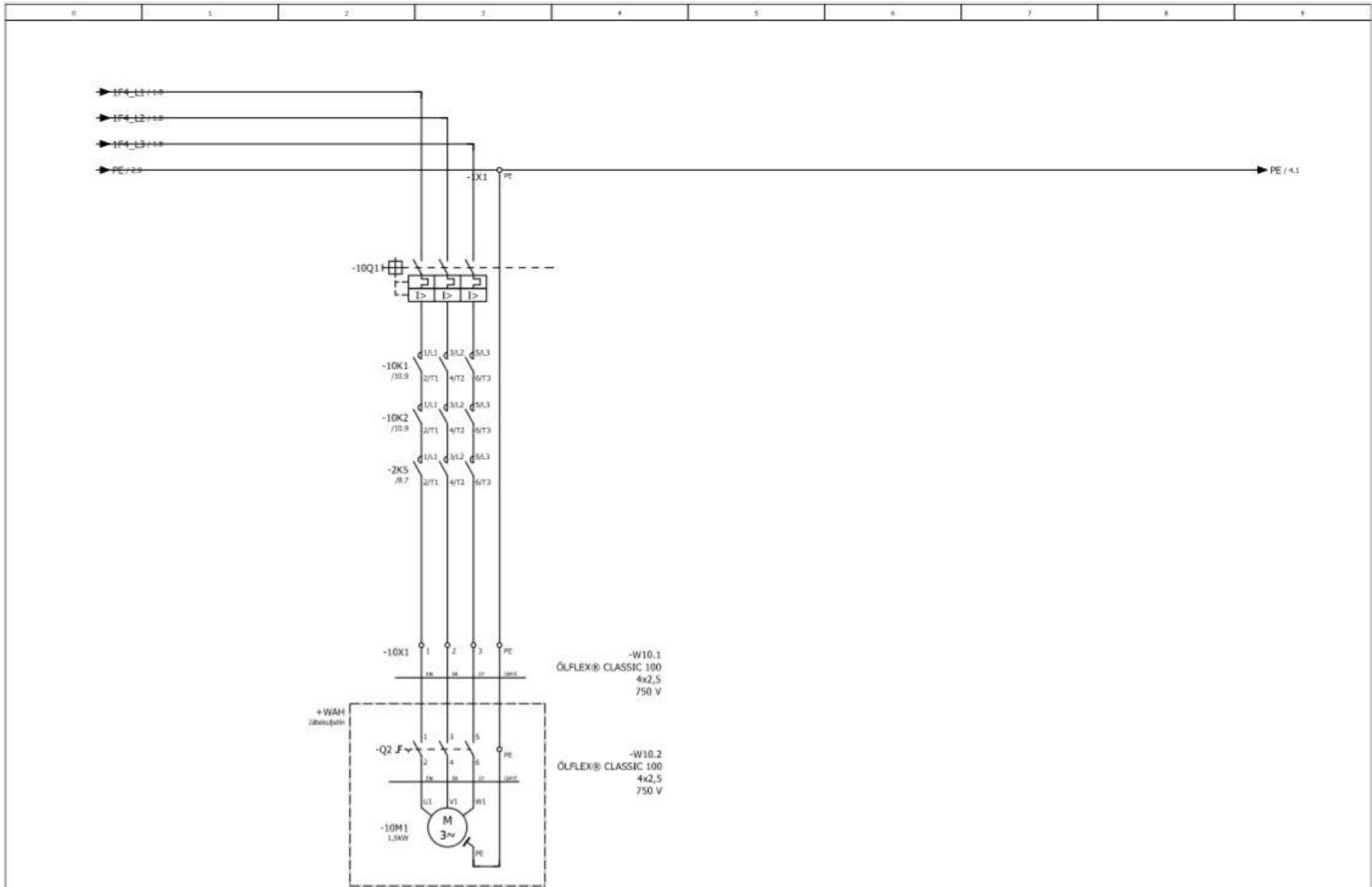


AMK Tekniikka

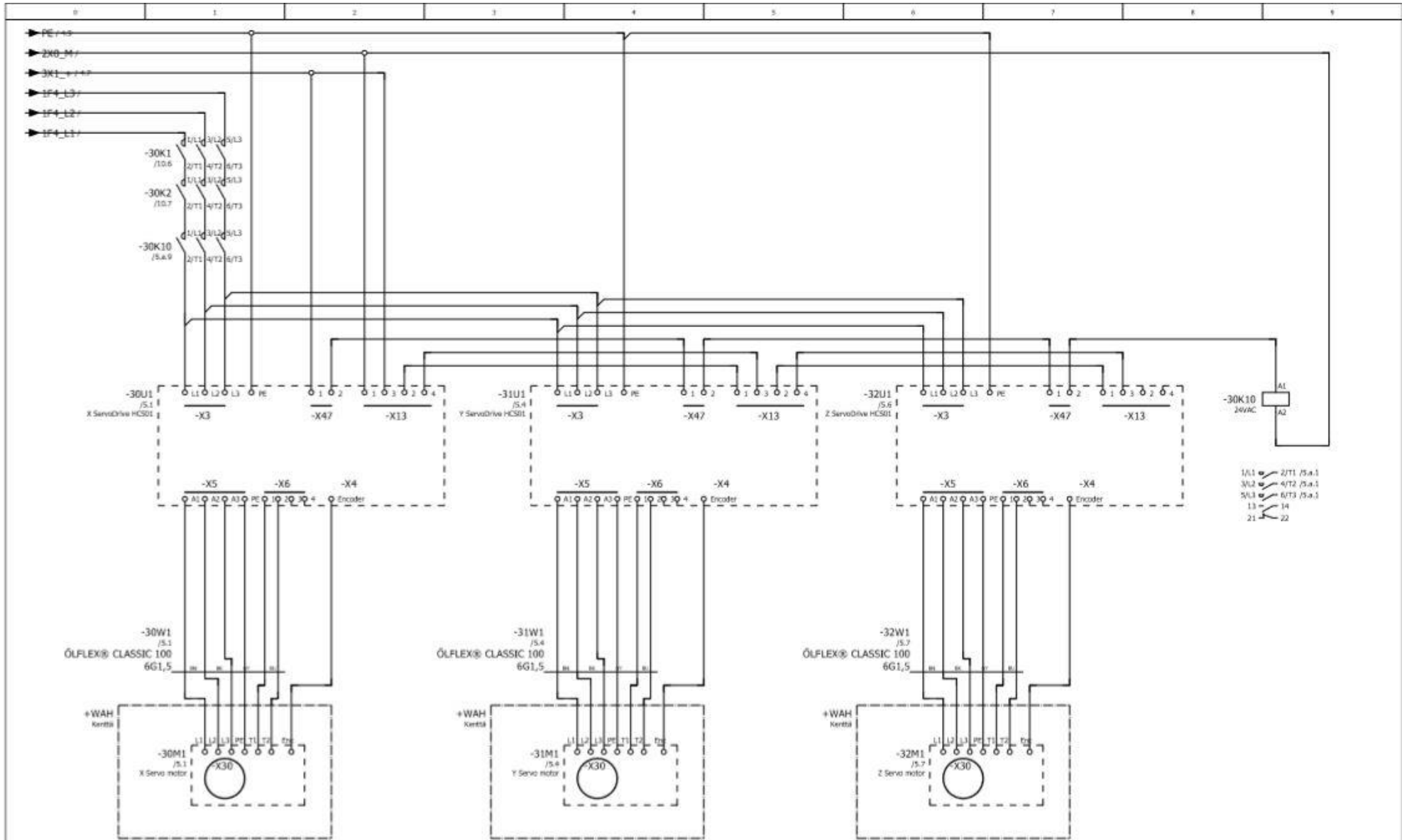
Date: 8.3.2019		JPU		JAPU Oy		Sähkönsyöttö		= 60130	
Ei: K1401261		Project template with identification structure in accordance with IEC standard: Page structure		with higher-level function and mounting location		ITC_r001		+ ICC1	
Appr:		Replacement of:		Replaced by:				Page: 1	
Modification:		Original:						Page: 57 / 70	



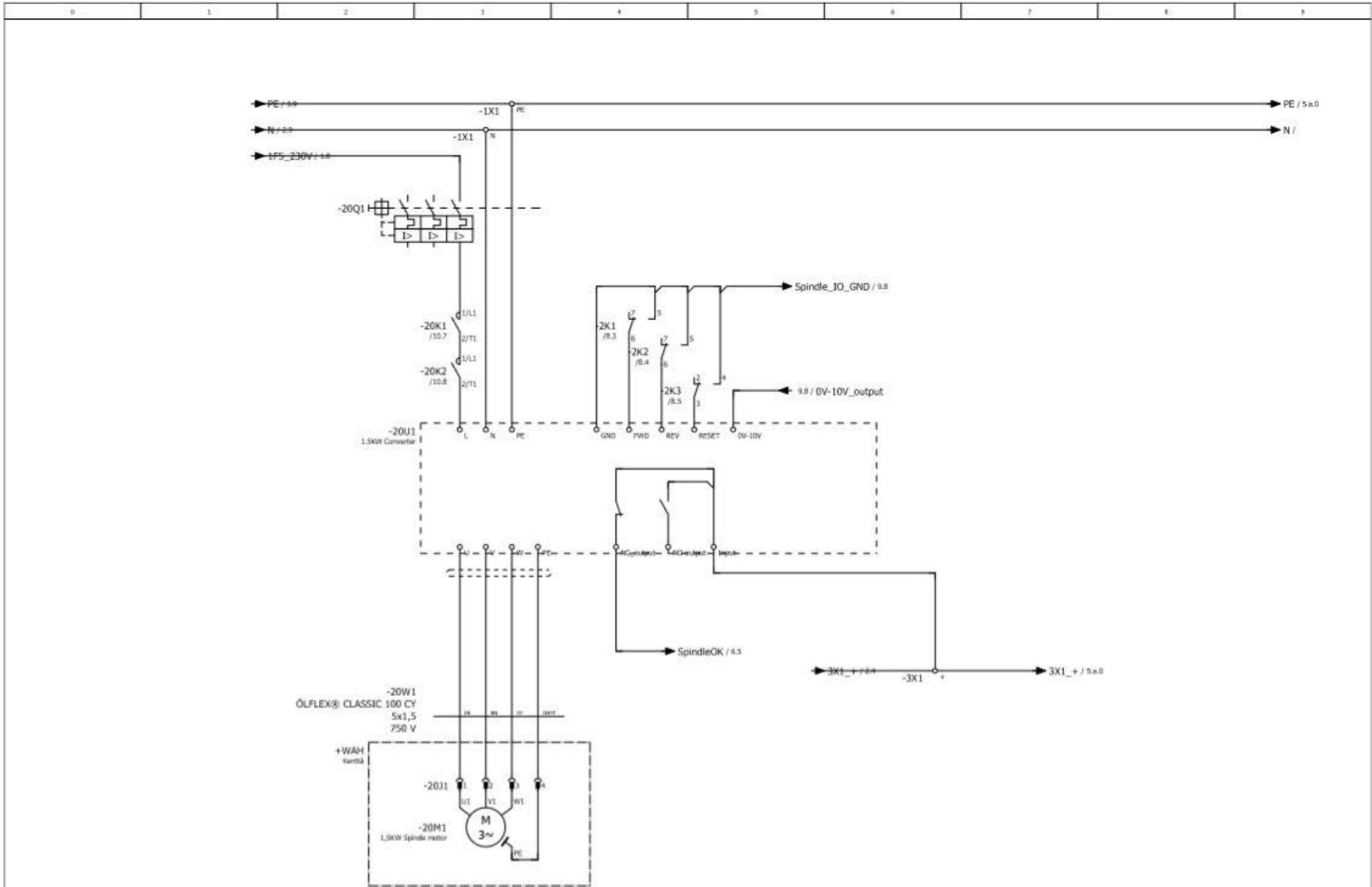
Date	7.3.2019	JPU	JAPU Oy	Sähkönsyöttö	= 60130
SB	K3401261				= ICC1
Author		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	ITC_tpe001
					Page 2
					Page 38 / 70



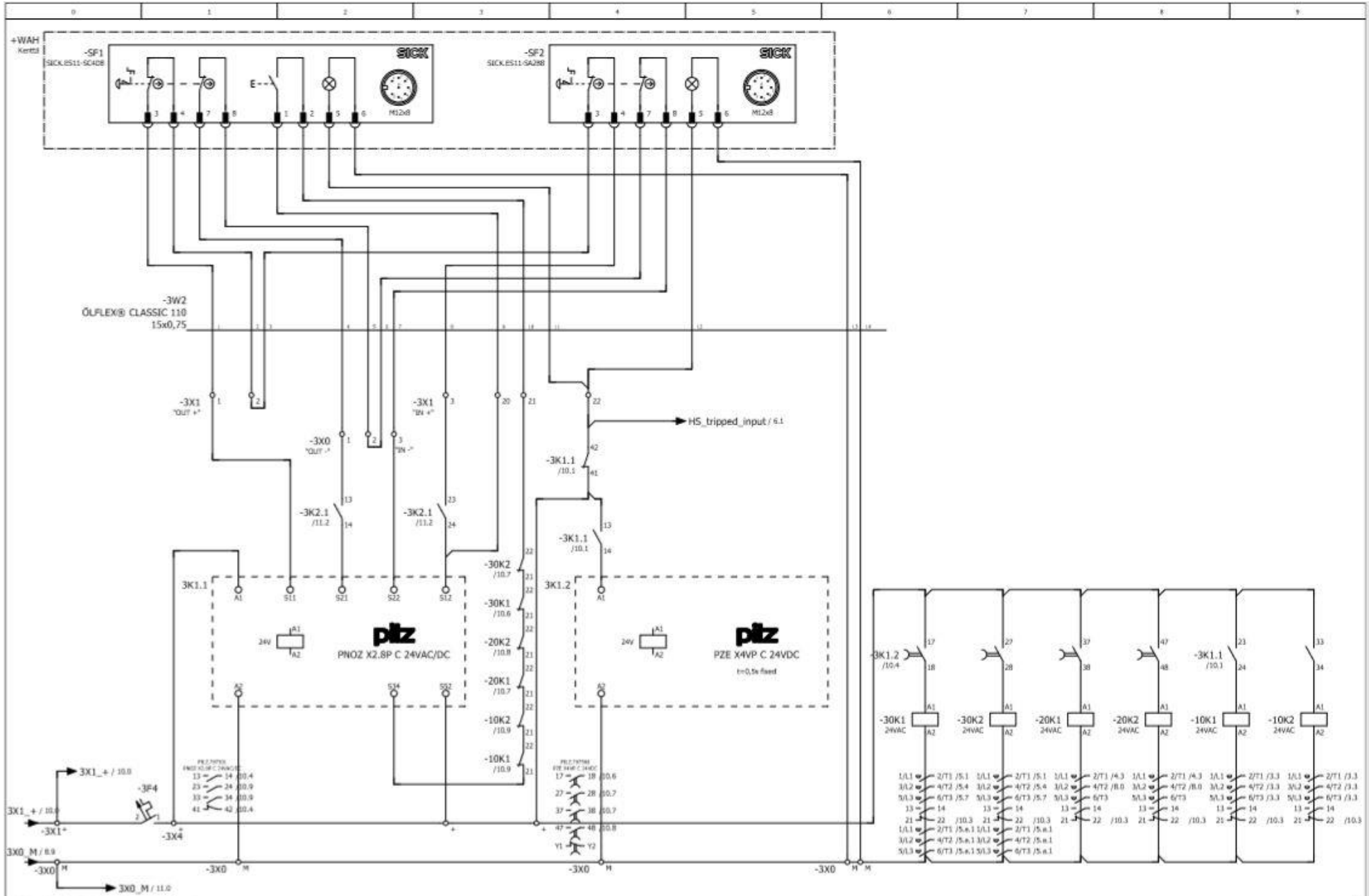
Date	3.3.2019	JPU	JAPU Oy	Pääkaavio Kuljetin	= 60150
Ed	K1401261				= 1001
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	IBC_tyd001
					Page 3
					Page 30 / 70



Date	27.5.2019	JPU	JAPU Oy	Pääkaavio Servomootorit	= 60130
Id	414021261				= ICC1
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	ITC_tpd01
					Page 5 of 62 / 70

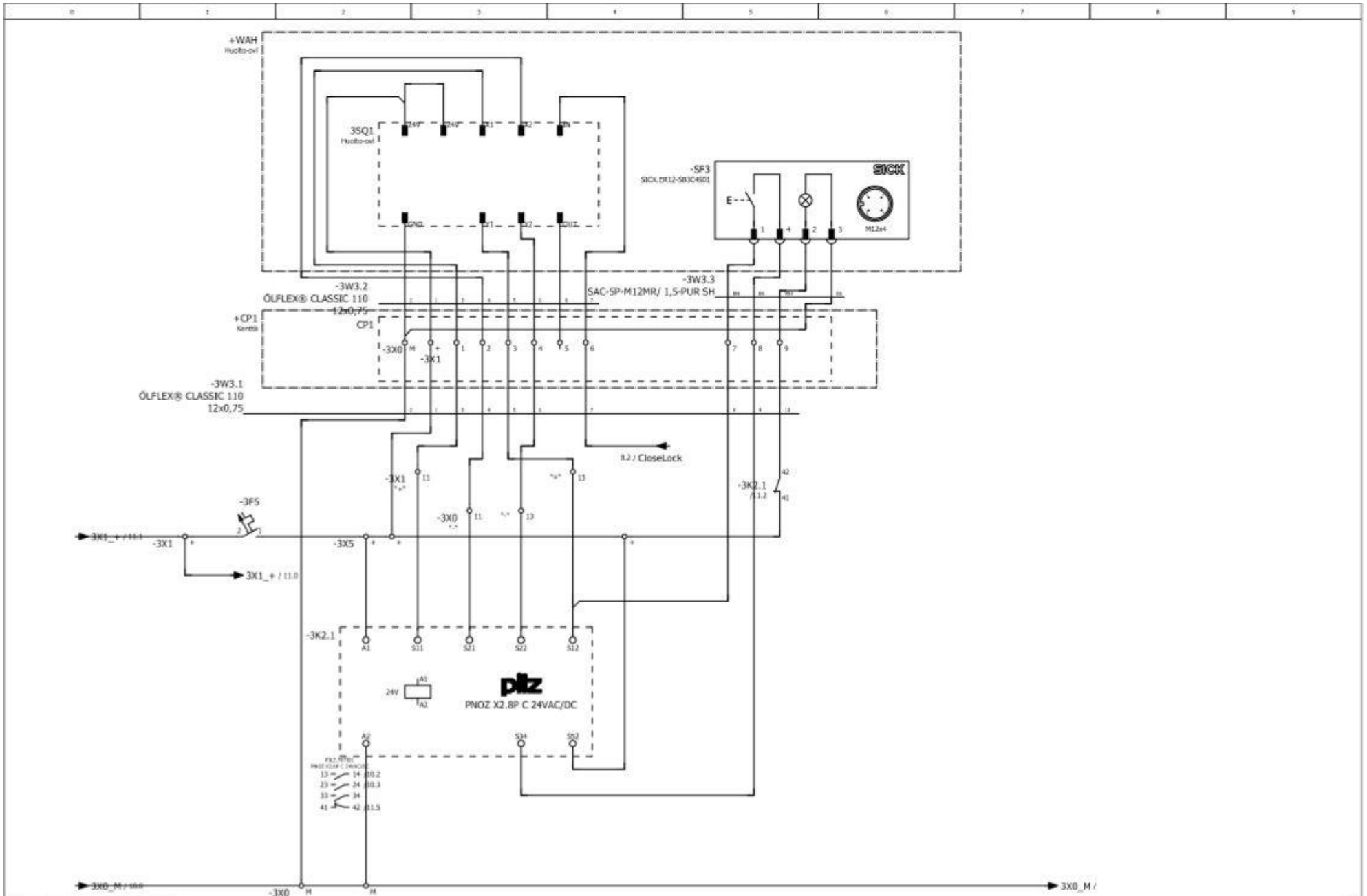


Date	27.5.2019	JPU	JAPU Oy	Pääkaavio Kara moottori	= 60150
SS	K1401261				= ICC1
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	ITC_jp001 Page 4
					Page 60 / 70

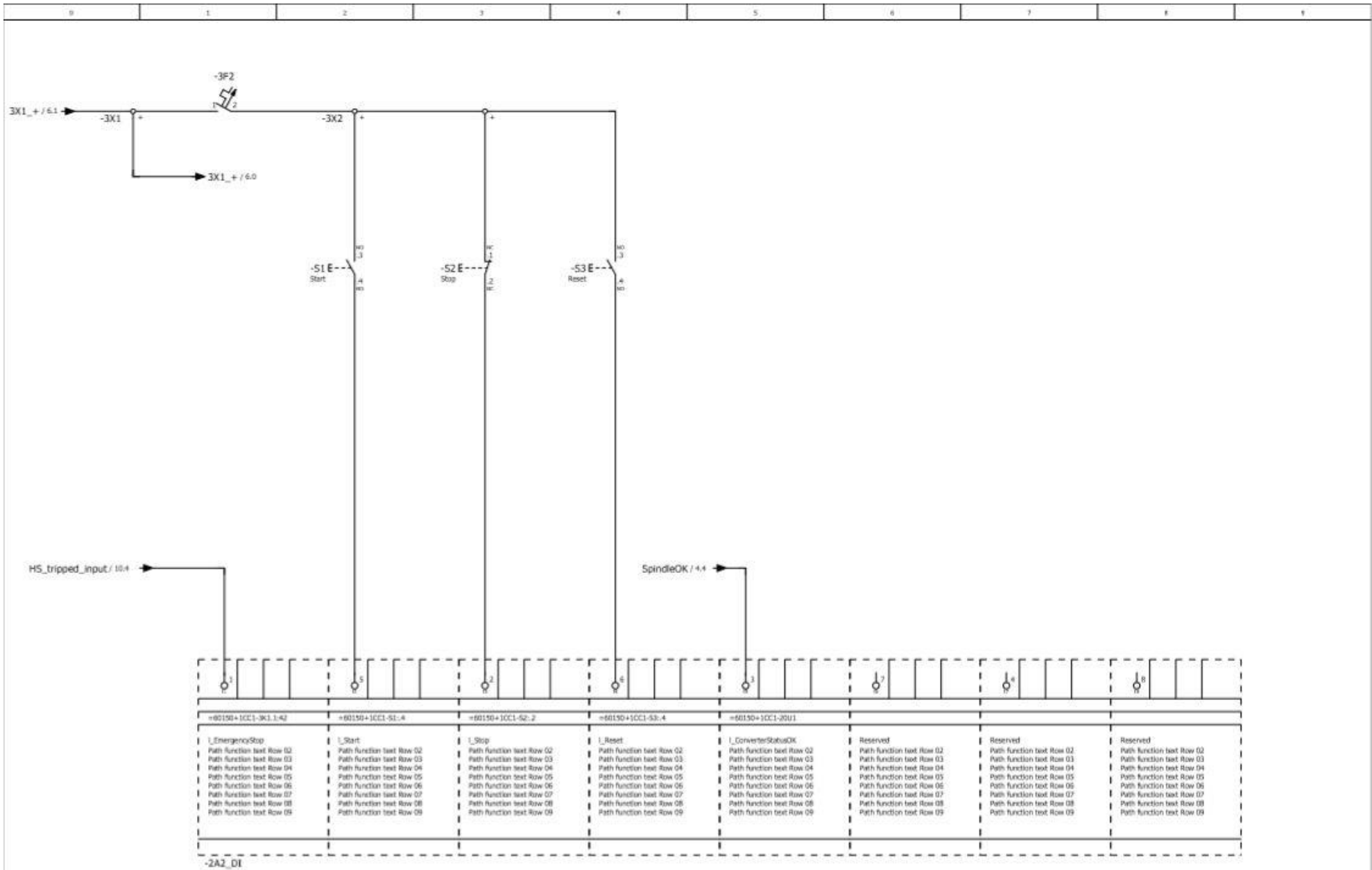


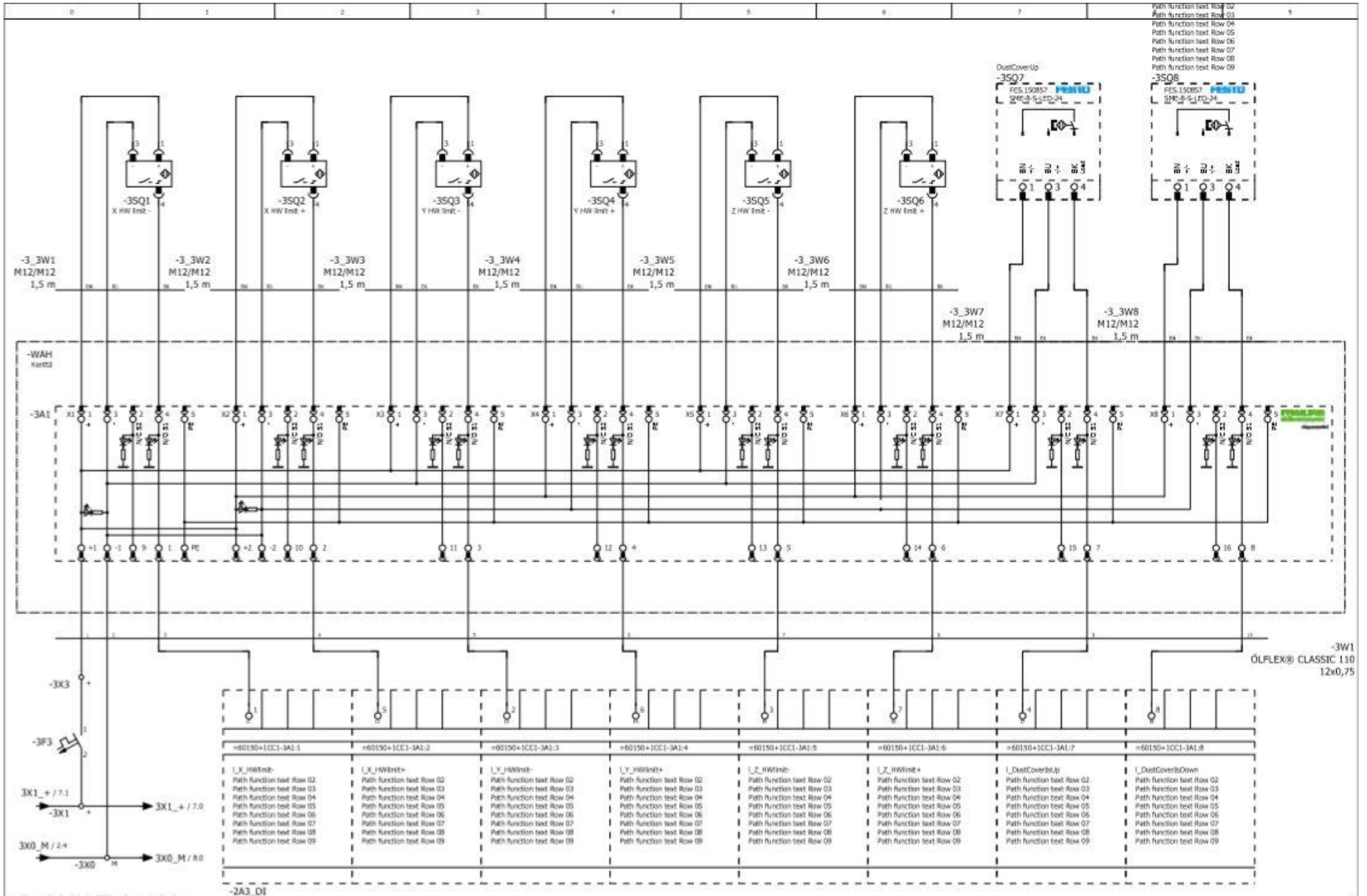
9 SoAMK Tekniikka

Date	7.3.2019	JPU	IAPU Oy	HS turvapiiri	= 60180
Si	K1401261				+ 1CC1
Appr		Project template with identification structure in accordance with IEC standard; Page structure with higher-level function and mounting location			Page 10
Modification	Date	Name	Original	Replaced by	Page 67 / 70



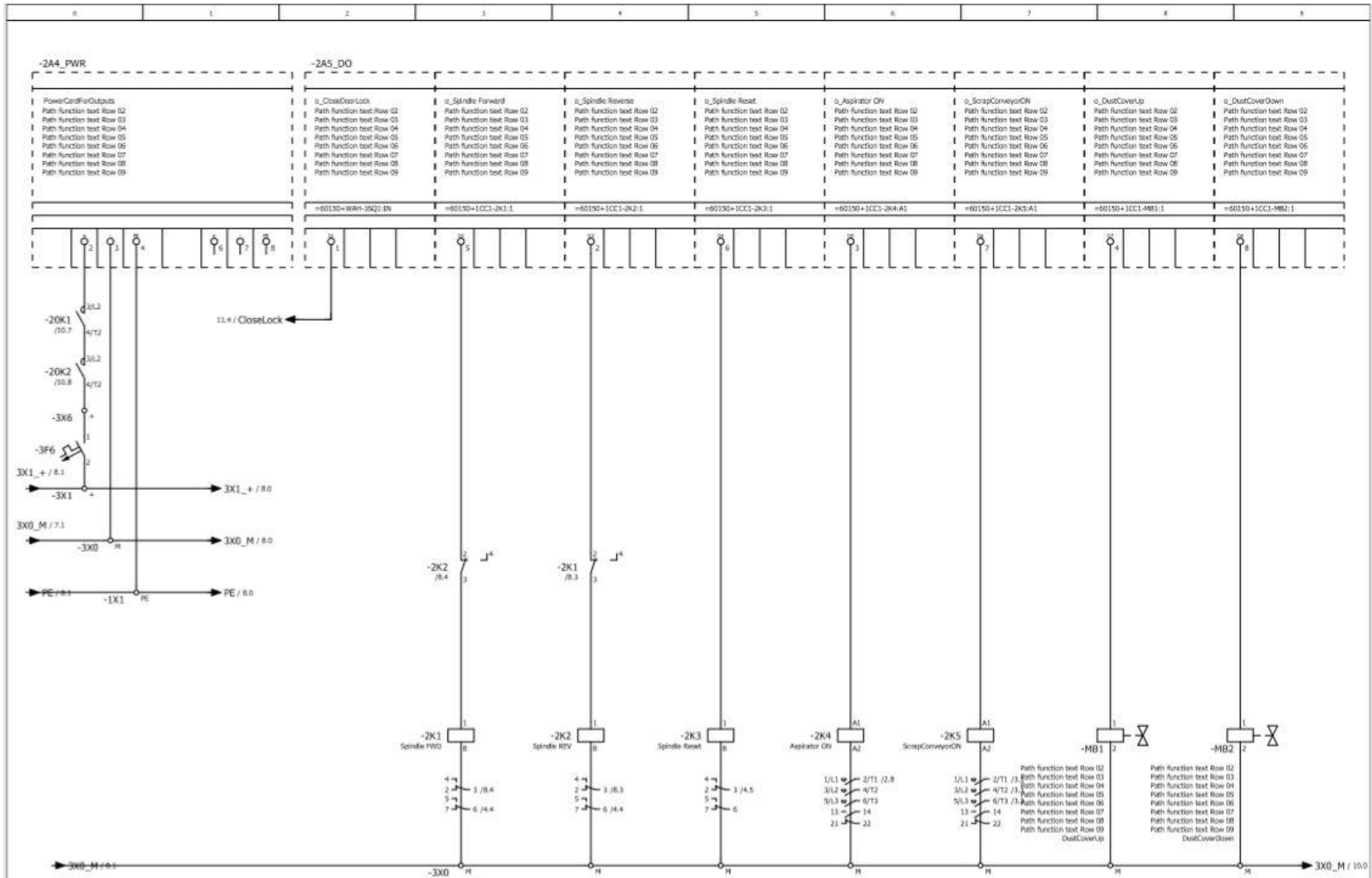
Date	8.3.2019	JPU	JAPU Oy	Huolto-ovi turvapiri	= 60L3D
Ed	K1401261				+ 1CC1
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	ITC_tsp001
					Page 51
					Page 68 / 70





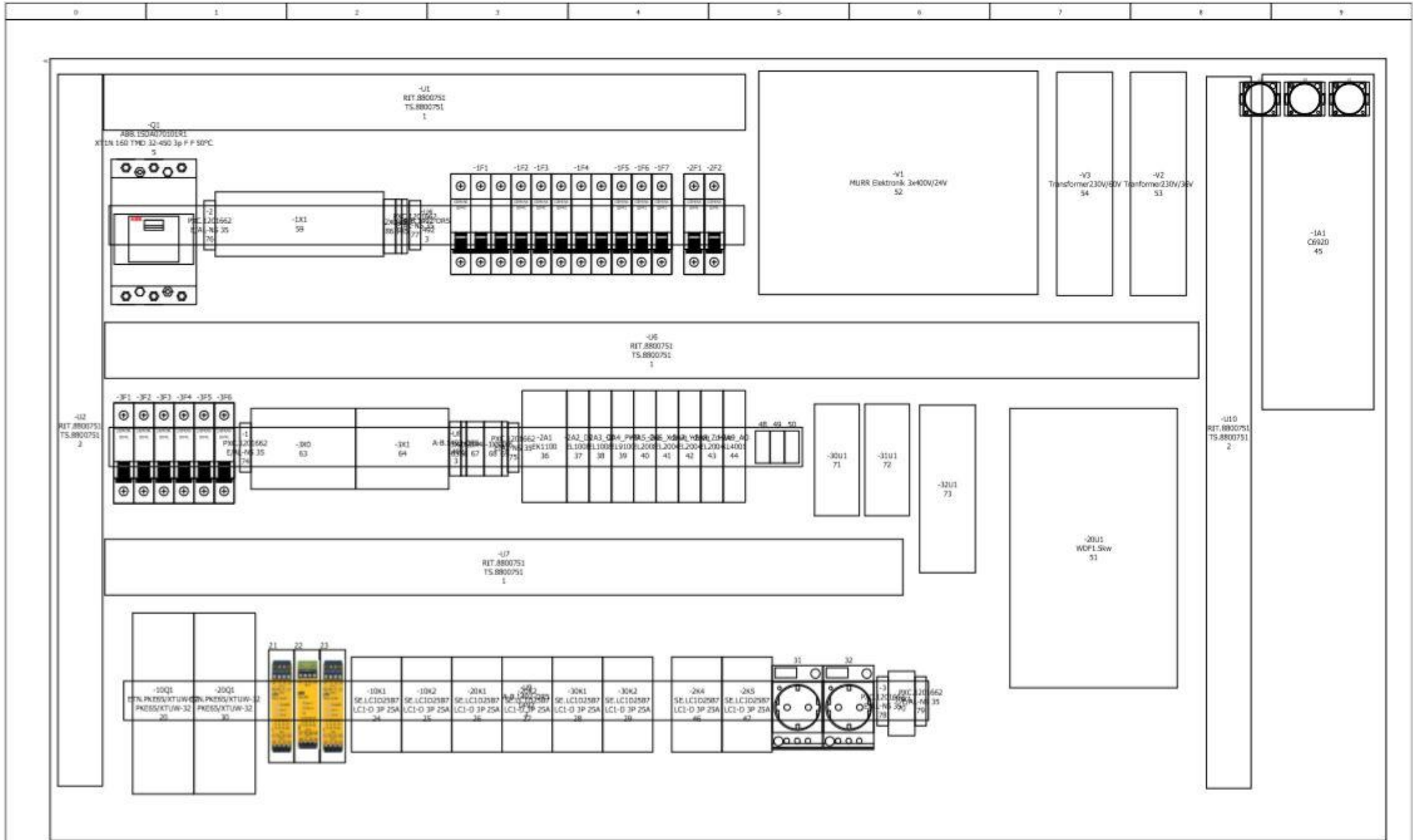
6 SoAMK Tekniikka

Date	8.3.2019	JPU	JAPU Oy	Piirikaavio Input Kenttä	= 60130
SB	K140281				= ICC1
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			Page 7
Modification	Date	Name	Original	Replaced by	Page 64 / 70



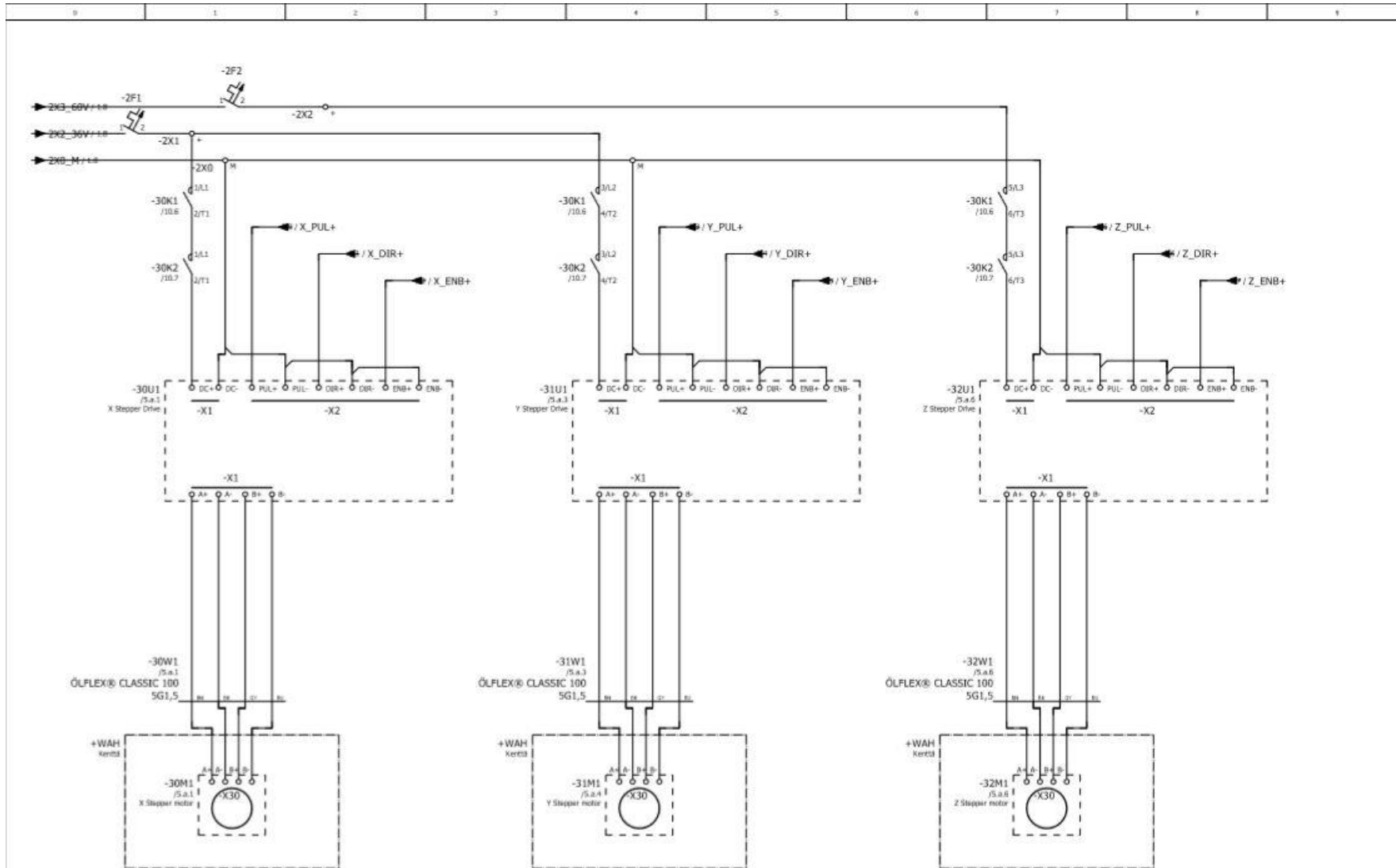
7 SoAMK Tekniikka

			Date	7.3.2019	JPU	JAPU Oy	Piirikaavio Output	= 60150	9
			Ed.	K1401261	Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			= 1CC1	
			Appr.					ITC_tpf001	Page 8
Modification	Date	Name	Original	Replacement of	Replaced by				Page 65 / 70



Date	27.5.2019	JPU	JAPU Oy	Keskuslayout ICC1	= 60150
Ohj.	4.1401261				+ ICC1
Appr.		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replaced by	IEC_14001
					Page 60 / 70

Liite 2. Piirikaaviokuva askelmoottoreista ja askelmoottoriohjaimista



Date	27.5.2019	JPU	JAPU Oy	Pääkaavio Askelmoottorit	= 60190
SB	A1401261				= 1CC1
Appr		Project template with identification structure in accordance with IEC standard: Page structure with higher-level function and mounting location			
Modification	Date	Name	Original	Replacement of	Replaced by
					ITC_tpk001
					Page 5
					Page 61 / 70

Liite 3. CNC-jyrsimen TwinCat 3 MAIN-ohjelmakoodi

```
PROGRAM MAIN
```

```
VAR
```

```

fbEnableAxes           :FB_AxisEnable;
fbModeSelection        :FB_ModeSelection;
fbRefModeSequence     :FB_RefModeSequence;
fbJogModeSequence     :FB_JogModeSequence;
fbAutoModeSequence    :FB_AutoModeSequence;
fbSetWorkCoordinates  :FB_SetWorkCoordinates;
fbReset               :FB_Reset;
fbSpindleControl      :FB_SpindleControl;

bRefModeBusy          :BOOL;
bJogModeBusy          :BOOL;
bAutoModeBusy         :BOOL;

bRefModeError         :BOOL;
bJogModeError         :BOOL;
bAutoModeError        :BOOL;

```

```
END_VAR
```

```
// CNCmill V1.1
```

```
//=====
```

```
// Axis reset
```

```
fbReset();
```

```
// Enable all axis if Enable axis switch is ON and safety is OK
```

```
fbEnableAxes(bEnableAxes := i_bEnableAxesSW AND i_bSafetyOK);
```

```
// Auto, Jog or Ref Mode selection
```

```
fbModeSelection();
```

```
// Reference mode sequence, this machine has absolute encoders so referencing is not used
```

```
// but FB is still checking that absolut encoders are in reference
```

```
fbRefModeSequence(
  bExecute := i_bCycleStartPB AND bRefMode AND i_bSafetyOK AND NOT bMachineStateError,
  bBusy=> bRefModeBusy,
  bError=> bRefModeError,
  bDone=> );
```

```
// Jog Mode sequence
```

```
fbJogModeSequence(
  bExecute := bJogMode AND i_bSafetyOK, //AND NOT bMachineStateError,
  bBusy=> bJogModeBusy,
  bError=> bJogModeError,
  bDone=> );
```

```
// Sets and calculate work coordinates
```

```
fbSetWorkCoordinates();
```

```
// Setting interpolation feedrate
```

```
ItpSetOverridePercent(fOverridePercent := fUserOverride , sPlcToNci:=out_stPlcToItp);
```

```
// Auto Mode sequence. 3 axis interpolation.
```

```
fbAutoModeSequence(
  bExecute := i_bCycleStartPB AND bAutoMode AND i_bSafetyOK AND bAllAxesReady AND NOT bMachineStateError,
  bBusy=> bAutoModeBusy,
  bError=> bAutoModeError,
  bDone=> bNcProgramDone);
```

```
// Spindle motor control
```

```
fbSpindleControl();
```

```
// Machine State info
```

```
bMachineStateBusy := bRefModeBusy OR bJogModeBusy OR bAutoModeBusy;
```

```
bMachineStateError := bRefModeError OR bJogModeError OR bAutoModeError;
```

Liite 4. CNC-jyrsimen TwinCat 3 FB_SetWorkCoordinates-ohjelmakoodi

```

FUNCTION_BLOCK FB_SetWorkCoordinates

VAR_INPUT
END_VAR
VAR_OUTPUT
END_VAR
VAR
    fbReadG54offset          : ItpReadZeroShiftEx;
    fbResetG54offset        : ItpSetZeroShiftNullEx;
END_VAR

// Write G54 X offset to ItpZeroShiftEx function block
IF bWriteXorigin THEN
    fShiftX := Xaxis.NcToPlc.ActPos;
    out_G54ZeroShift.fShiftX := fShiftX;
END_IF

// Write G54 Y offset to ItpZeroShiftEx function block
IF bWriteYorigin THEN
    fShiftY := Yaxis.NcToPlc.ActPos;
    out_G54ZeroShift.fShiftY := fShiftY;
END_IF

// Write G54 Z offset to ItpZeroShiftEx function block
IF bWriteZorigin THEN
    fShiftZ := Zaxis.NcToPlc.ActPos;
    out_G54ZeroShift.fShiftZ := fShiftZ;
END_IF

// Resets G54 offset values from interpreter
fbResetG54offset(
    bExecute:= bResetG54offset,
    tTimeout:= ,
    sNciToPlc:= in_stItpToPlc,
    bBusy => ,
    bErr => ,
    nErrId => );

// Reads G54 Offset values
fbReadG54offset(
    bExecute:= bReadG54offset,
    nZsNo:= 54,
    tTimeout:= ,
    sNciToPlc:= in_stItpToPlc,
    sZeroShiftDesc:= out_G54ZeroShift,
    bBusy => ,
    bErr => ,
    nErrId => );

// Calculate Work coordinates
XWorkCoordinate := Xaxis.NcToPlc.ActPos - fShiftX;
YWorkCoordinate := Yaxis.NcToPlc.ActPos - fShiftY;
ZWorkCoordinate := Zaxis.NcToPlc.ActPos - fShiftZ;

```

Liite 5. CNC-jyrsimen TwinCat 3 FB_AutoModeSequence-ohjelmakoodi

```

FUNCTION_BLOCK FB_AutoModeSequence

VAR_INPUT
    bExecute          : BOOL;
END_VAR
VAR_OUTPUT
    bBusy             : BOOL;
    bError            : BOOL;
    bDone             : BOOL;
END_VAR
VAR_IN_OUT
END_VAR
VAR
    nState            : UDINT      := 0;
                                // Auto cycle current state
    nErrorState       : UDINT;
                                // Error state number

    fbWriteG54offset  : ItpWriteZeroShiftEx; // G54 offsets to interpreter
    fbBuildGroup      : CfgBuildExt3DGroup;  // Build 3D interpolation group
    fbLoadGCode       : ItpLoadProgEx;      // Load G-code to interpreter
    fbStartGCode      : ItpStartStopEx;     // Execute G-code program
    fbClearGroup      : CfgReconfigGroup;   // Clear 3D interpolation group
    fbConfirmHsk      : ItpConfirmHsk;     // Execute M-function
    fbStop            : RS;

    fUserOverrideOld  : LREAL;
    nInterpreterState : UDINT      := 0;    // Interpreter state
    bConfirmHsk       : BOOL       := FALSE; // Confirm current M-function
    sPrgName          : STRING(255); // Program file name
END_VAR

// Reset State machine
IF i_bResetPB THEN
    bBusy      := FALSE;
    bDone      := FALSE;
    bError     := FALSE;
    nState     := 0;
    nErrorState := nState;
END_IF

// If CycleStop is ON set feedrate to 0
fbStop(
    SET:= i_bCycleStopPB,
    RESET:= i_bCycleStartPB OR i_bResetPB,
    Q1=> bCycleStop);
IF bCycleStop THEN
    fUserOverride := fUserOverrideOld;
    fUserOverride := 0;
    MsgMachineState := 'Machine Stopped';
ELSE
    fUserOverrideOld := fUserOverride;
END_IF

// Automatic program cycle
// First step is to set machine state to busy
CASE nState OF
0:
    IF bExecute THEN
        bBusy      := TRUE;
        bDone      := FALSE;
        bError     := FALSE;
        nErrorState := nState;
        MsgMachineState := 'Start the program';
        nState      := 10;
    END_IF
END_CASE

```

```

// Build 3D interpolation group
fbBuildGroup(
    bExecute:=TRUE,
    nGroupId:=ItpGetGroupId(sNciToPlc:=in_stItpToPlc) ,
    nXAxisId:=Xaxis.NcToPlc.AxisId,
    nYAxisId:=Yaxis.NcToPlc.AxisId,
    nZAxisId:=Zaxis.NcToPlc.AxisId,
    tTimeout:= ,
    bBusy=> ,
    bErr=> ,
    nErrId=> );
IF NOT fbBuildGroup.bBusy THEN
    IF NOT fbBuildGroup.bErr THEN
        nState := 15;
    ELSE
        (* error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        MsgAlarm :='Error when building interpolation group';
        nState := 9999;
    END_IF
    fbBuildGroup(bExecute:=FALSE);
END_IF
15:
//Write G54 offsets to interpreter
fbWriteG54offset(
    bExecute:= TRUE,
    nZsNo:= 54,
    tTimeout:= ,
    sNciToPlc:= in_stItpToPlc,
    sZeroShiftDesc:= out_G54ZeroShift,
    bBusy => ,
    bErr => ,
    nErrId => );
IF NOT fbWriteG54offset.bBusy THEN
    IF NOT fbWriteG54offset.bErr THEN
        nState := 20;
    ELSE
        (* error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        MsgAlarm :='Error when writing G54 offsets';
        nState := 9999;
    END_IF
    fbWriteG54offset(
        bExecute:=FALSE,
        sNciToPlc:=in_stItpToPlc,
        sZeroShiftDesc:= out_G54ZeroShift,);
END_IF
20:
// Load g-code file
fbLoadGCode(
    sNciToPlc:=in_stItpToPlc,
    bExecute:=TRUE,
    sPrg:= sPrgName,
    nLength:= INT_TO_UDINT(LEN(sPrgName)),
    tTimeout:= ,
    bBusy=> ,
    bErr=> ,
    nErrId=> );
IF NOT fbLoadGCode.bBusy THEN
    IF NOT fbLoadGCode.bErr THEN
        nState := 30;
    ELSE
        (* error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        MsgAlarm :='Error when loading g-code file';
        nState := 9999;
    END_IF
    fbLoadGCode(sNciToPlc:=in_stItpToPlc, bExecute:=FALSE);
END_IF

```

```

30: // Check the status of the interpreter. It has to be in ready state, in case of loading the program suc-
    // cessfully.
    nInterpreterState := ItpGetStateInterpreter(sNciToPlc:=in_stItpToPlc);
    IF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_READY THEN
        nState := 40;
    ELSE
        (* error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        MsgAlarm := 'Interpreter is not ready';
        nState := 9999;
    END_IF

40: // Start g-code file
    fbStartGCode(
        bStart:=TRUE,
        bStop:=FALSE,
        tTimeout:= ,
        sNciToPlc:= in_stItpToPlc,
        bBusy=> ,
        bErr=> ,
        nErrId=> );
    IF NOT fbStartGCode.bBusy THEN
        IF NOT fbStartGCode.bErr THEN
            nState := 50;
        ELSE
            (* error handling *)
            nErrorState := nState;
            bBusy := FALSE;
            bError := TRUE;
            MsgAlarm := 'Error when starting program';
            nState := 9999;
        END_IF
        fbStartGCode( bStart:=FALSE, sNciToPlc:= in_stItpToPlc );
    END_IF

50: // check state, again - we are at least not in ready state for several ticks
    // this is to ensure that we don't indicate program has finished, before we have started
    nInterpreterState := ItpGetStateInterpreter(sNciToPlc:=in_stItpToPlc);
    IF nInterpreterState <> Tc2_NCI.NCI_INTERPRETER_READY THEN
        // Program is running
        MsgMachineState := 'Program Running';
        nState := 60;
    END_IF

60: nInterpreterState := ItpGetStateInterpreter(sNciToPlc:=in_stItpToPlc);
    IF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_READY THEN
        // program has finished
        MsgMachineState := 'Program Finished';
        nState := 70;
    ELSIF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_ABORTED THEN
        // a run-time error occurred - this could be a lag error or something else...
        // error handling
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        MsgAlarm := 'Program interrupted';
        nState := 9999;
    END_IF

70: // program has finished
    // clear 3D interpolation group
    fbClearGroup(
        bExecute:=TRUE,
        nGroupId:=ItpGetGroupId(sNciToPlc:=in_stItpToPlc) ,
        tTimeout:= ,
        bBusy=> ,
        bErr=> ,
        nErrId=> );
    IF NOT fbClearGroup.bBusy THEN
        IF NOT fbClearGroup.bErr THEN
            nState := 80;
        ELSE
            (* error handling *)

```

```

                                nErrorState := nState;
                                bBusy       := FALSE;
                                bError      := TRUE;
                                MsgAlarm   := 'Error when clearing interpolation group';
                                nState     := 9999;
                                END_IF
                                fbClearGroup(bExecute:=FALSE);
                                END_IF
80:
    bDone       := TRUE;
    bBusy      := FALSE;
    IF NOT bExecute THEN
        nState   := 0;
    END_IF
9999:
    // error state
    IF NOT bExecute THEN
        nState   := 0;
        bError   := FALSE;
    END_IF
END_CASE

(***) M-Function handling (***)
// M-functions are handled outside of state machine
// Reads M-function from G-code
IF ItpIsHskMFunc(in_stItpToPlc) AND NOT fbConfirmHsk.bBusy THEN
    ItpGetHskMFunc(in_stItpToPlc);

    // IF M3-function detected start Spindle CW
    IF in_stItpToPlc.HskMFuncNo = 3 THEN
        o_SpindleCW := TRUE;
        MsgSpindleState := 'Spindle CW';
        bConfirmHsk   := TRUE;
    ELSE
        o_SpindleCW := FALSE;
    END_IF

    // If M4-function detected starts Spindle CCW
    IF in_stItpToPlc.HskMFuncNo = 4 THEN
        o_SpindleCCW := TRUE;
        MsgSpindleState := 'Spindle CCW';
        bConfirmHsk   := TRUE;
    ELSE
        o_SpindleCCW := FALSE;
    END_IF

    // If M5-fuction detected Stops Spindle
    IF in_stItpToPlc.HskMFuncNo = 5 THEN
        o_SpindleCW := FALSE;
        o_SpindleCCW := FALSE;
        MsgSpindleState := 'Spindle Stopped';
        bConfirmHsk   := TRUE;
    END_IF

ELSE
    bConfirmHsk := FALSE;
END_IF

// Execute M-function
fbConfirmHsk(
    bExecute:=bConfirmHsk ,
    sNciToPlc:=in_stItpToPlc ,
    sPlcToNci:=out_stPlcToItp ,
    bBusy=> ,
    bErr=> ,
    nErrId=> );

// Reading current block number
IF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_RUNNING THEN
bCurrentBlockNumber := ItpGetBlockNumber(sNciToPlc:= in_stItpToPlc);
END_IF

```