

Niko Raitaluoma

# Servomoottorisovellutus käyttöönoton harjoitteluun

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Kone- ja tuotantotekniikka

Insinööriytyö

8.5.2016

Tekijä Otsikko	Niko Raitaluoma Servomoottorisovellutus käyttöönoton harjoitteluun
Sivumäärä Aika	28 sivua 8.5.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Kone- ja tuotantotekniikka
Suuntautumisvaihtoehto	Koneautomaatio
Ohjaaja	Lehtori Heikki Paavilainen
<p>Tämä insinööri työ tehtiin Metropolia Ammattikorkeakoululle. Työssä suunniteltiin, käyttöön otettiin ja ohjelmoitiin yksinkertainen ympäristö servomoottorin virittämisen harjoittelemiseen. Servomoottori viritetään valmiiksi tyhjälle akselille ja opiskelijat harjoittelevat virittämistä kuormalla. Servon ohjaukseen laadittiin graafinen käyttöliittymä ja scope virityksen tarkasteluun.</p> <p>Kokoonpano suunniteltiin jo olemassa olevan Metropolian servomoottorin ympärille. Beckhoffin valmistama kokoonpano kiinnitettiin alustalle ja virtajohtojen liitännöitä parannettiin. Moottorille rakennettiin oma kiinnike, jolla se saatiin tukevasti kiinnitettyä alustaan. Kuormaksi sorvattiin teräksinen ympyrälevy.</p> <p>Työssä opittiin servomoottorin virittämisestä, logiikkaohjelmoinnista sekä Beckhoffin Twin-CAT3-ohjelman käyttöä.</p>	
Avainsanat	Beckhoff, servomoottori, viritys, PLC

Author Title	Niko Raitaluoma Servo Motor Application for the Training of Introduction
Number of Pages Date	28 pages 8 May 2016
Degree	Bachelor of Engineering
Degree Programme	Mechanical Engineering
Specialisation option	Machine Automation
Instructor	Heikki Paavilainen, Senior Lecturer
<p>This Bachelor's thesis was assigned by Helsinki Metropolia University of Applied Sciences. The objective of the thesis was to design, introduce and program a simple environment to train the tuning of a servo motor. The servo motor is tuned for an empty axis and students are able to practice tuning on the loaded axis. Firstly, a graphical user interface was programmed to control the servo. Secondly, a scope was added to view the tuning.</p> <p>To start with, topic-related literature was examined. The assembly was designed around the existing servo motor of Metropolia UAS. The servo assembly made by Beckhoff Automation was installed on a platform. The couplings were enhanced by joining the power cords of two power supplies with connection strips. In addition, a protective covering was made around the servo motor to prevent accidents. The program was created with Beckhoff Automation's TwinCAT3 software using the function block diagram programming language.</p> <p>As a result, a good environment was made for practicing the tuning of the servo motor. Everything needed for tuning is installed in the program and the assembly is easy to connect to the computer using the Ethernet cable. The scopes are linked to appropriate variables and the program is intelligible to use.</p>	
Keywords	Beckhoff Automation, servo motor, tuning, PLC

# Sisällys

## Lyhenteet

1	Johdanto	1
2	Servotekniikka	2
2.1	Historia	2
2.2	Toimintaperiaate	2
2.3	Anturit	3
2.3.1	Inkrementtianturi	4
2.3.2	Absoluuttianturi	4
3	Ohjelmoitava logiikka	5
3.1	Historia	5
3.2	TwinCAT3	6
3.3	Ohjelmointikielet	6
3.3.1	Strukturoitu teksti (Structured Text, ST)	6
3.3.2	Käskylista (Instruction List, IL)	6
3.3.3	Funktioblokkiohjelmointi (Function Block Diagram, FBD)	7
3.3.4	Relekaavio-ohjelmointi (Ladder Diagram, LD)	7
3.3.5	Sekvessikaavio-ohjelmointi (Sequential Function Chart, SFC)	7
4	Laitteisto	8
4.1	Moottori	8
4.2	Servokäytöt	9
4.3	Virransyöttö	11
4.4	Väylä	12
5	Ohjelma ja käyttöliittymä	13
5.1	Ohjelma	13
5.2	Käyttöliittymä	18

6	Viritys	19
6.1	Servon säätimet	19
6.1.1	P-säädin	19
6.1.2	PI-säädin	20
6.1.3	PID-säädin	20
6.2	Työssä käytetyn servon viritys	20
7	Kokoonpano	22
7.1	Suunnitelma	22
7.2	Toteutus	23
8	Yhteenveto	25
	Lähteet	26
	Kuvalähteet	28

## Lyhenteet

EtherCat	Ethernet Control Automation Technology, Ethernetiin perustuva kenttäväylä teollisuuteen.
TwinCAT	The Windows Control and Automation Technology, ohjelma jolla servon logiikkaohjelmointi tuotettiin.
OCT	One Cable Technology, yhden kaapelin teknologia. Servo moottorille tulee vain yksi kaapeli, jota pitkin myös takaisinkytkentä toimii.
PLC	Programmable Logic Controller, ohjelmoitava logiikka.
Scope	TwinCATissa käytetyn virtuaalisen oskilloskoopin nimitys.

## 1 Johdanto

Insinööriyössä suunniteltiin servomootorilaitteisto, jolla voi harjoitella servomootorin käyttöönottoa. Tavoitteena oli saada kokoonpanoon kuuluvat osat kätevästi yhdelle helposti liikuteltavalle alustalle. Servokäyttö ja virtalähteet olivat jo valmiiksi asennettuna C-kiskoon, johon lisättiin riviliittimiä sähköjohtojen yhdistämiseksi. C-kisko ja moottori sijoitettiin alustalle. Moottorille rakennettiin oma kiinnike.

Servomootorille haluttiin helppokäyttöinen käyttöliittymä, jolla sitä voidaan ohjata sekä scope, jolla akselin paikkaa voidaan seurata sen virittämiseksi. Käyttöliittymä toteutettiin Beckhoff Automationin TwinCAT3 sovelluksella. Työssä perehdyttiin servotekniikkaan, PLC-ohjelmointiin ja servomootorin viritykseen.

## 2 Servotekniikka

### 2.1 Historia

Sana servo tulee latinan orjaa tarkoittavasta sanasta. Tämä kuvaa hyvin servo järjestelmien toimintaa, koska servot ottavat ohjeita muualta ja toimivat sen mukaisesti. Ensimmäisenä servona voidaan pitää James Wattin höyrykonetta. Se toimi servon periaatteiden mukaisesti. Wattin höyrykoneessa oli erillinen kondensaatiokammio, johon virtasi vettä höyryn noustessa mäntään sylinterin päässä. Suljettu systeemi korjasi itseään, kunnes haluttu asema palasi ja sykli alkoi uudestaan. (1; 2.)

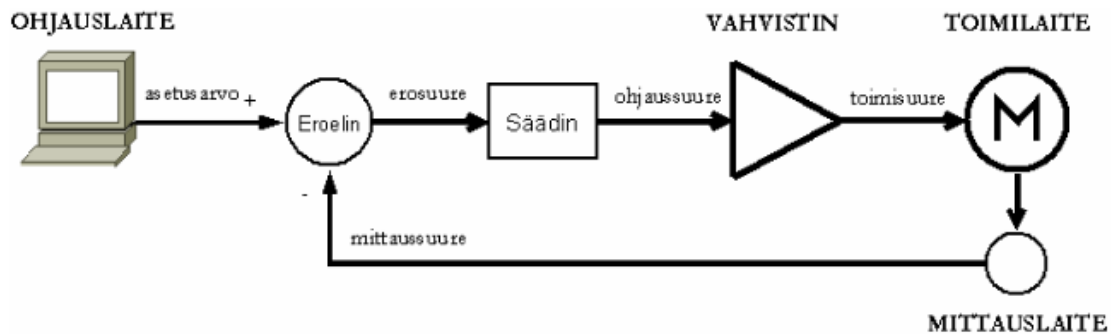
Servotekniikka alkoi kehittyä kuitenkin teollisuuteen vasta Toisen maailmansodan jälkeen. Toisessa maailmansodassa huomattiin servotekniikan edut ilmatorjunnassa, jossa tarvittiin nopeita ja tarkkoja liikkeitä. Siitä eteenpäin servotekniikan kehitys on ollut huijaa ja tekniikkaa onkin nykyisin käytössä lähes kaikessa automatisoidussa teollisuudessa. (1; 2.)

### 2.2 Toimintaperiaate

Servomootorilla tarkoitetaan hydraulii- tai sähkömootoria, jota ohjataan elektronisen servo-ohjaimen avulla haluttuun arvoon. Servomottoorin ohjaus perustuu takaisinkytkentään. Tämä tarkoittaa sitä, että servo-ohjain vastaanottaa servomootorilta asematiedon, vertaa sitä haluttuun asemaan ja ohjaa servomootorin pyörimistä saadun tiedon perusteella. Servokäytöstä puhuttaessa tarkoitetaan yhdessä moottoria ja asema- sekä nopeusanturia. (3; 4.)



Kuva 1 havainnollistaa servojärjestelmän toimintaa. Ohjauslaite on esimerkiksi PC, jolla käyttäjä ilmoittaa haluamansa aseman tai nopeuden. Eroelin vertaa mittauslaitteen antamaa arvoa PC:ltä annettuun arvoon ja säädin muokkaa sitä erotuksen mukaan. Vahvistin muuttaa tulevan arvon toimilaitteen ymmärtämäksi. Kun asetusarvo ja mittauslaitteen mittaama arvo on sama, laite on halutussa paikassa tai nopeudessa. (3; 4.)



Kuva 1. Servojärjestelmä

Tässä työssä käytössä olevassa moottorissa on sisäinen absoluuttianturi, jota hyödynnetään virityksessä. Virittäminen on tärkeä osa servomoottorin toimintaa. Servomoottori tulee virittää aina kuhunkin sovellutukseen erikseen, jotta se toimii mahdollisimman nopeasti ja tarkasti. (3; 4.)

### 2.3 Anturit

Servokäytössä anturit ovat tärkeä osa toimivaa kokonaisuutta, jotta saadaan tieto toimilaitteen asemasta ja nopeudesta. Tähän tarkoitukseen kaksitilainen anturi ei ole enää riittävä, vaan tarvitaan antureita aseman ja nopeuden mittaamiseen. Antureiden kehittyminen ja niiden helppo kytkentä ohjausjärjestelmiin ovat osasyynä digitaalisten antureiden yleistymiseen. Servokäytössä tärkeimpiä antureita ovat inkrementtianturit ja absoluuttianturit. (5; 6.)

### 2.3.1 Inkrementtianturi

Inkrementtianturi eli pulssianturi on yksinkertainen anturityyppi, joka mittaa kääntymiskulmaa. Sen toimintaperiaatteena on valoa lähettävän LEDin ja valoa vastaanottavan diodin välissä oleva optinen pulssikiekko, joka on kiinnitetty akseliin. Pulssikiekossa on valoa läpi päästäviä ja läpäisemättömiä viivoja, joiden lukumäärä määräytyy anturin resoluution mukaan. Kiekon pyöriessä valo vilkkuu läpäisemättömien viivojen ansiosta ja tämä analoginen signaali vahvistetaan ja muutetaan suorakulmaiseksi kanttiaalloksi ulostuloon. Anturin pyörimissuunnan määrittämiseksi kiekolla on kaksi keskenään 90° vaihesiirrossa olevaa viivakehää. Näin saadaan aikaan kaksi erillistä kanttiaaltoa, A- ja B-kanavat. Varsinainen asematieto lasketaan A kanavan pulssirivistä ja B kanavan pulssirivi ilmaisee pyörimissuunnan. Inkrementtianturi kadottaa asematietonsa, mikäli jännite katoaa esimerkiksi sähkökatkoksen takia. (5; 6.)

### 2.3.2 Absoluuttianturi

Absoluuttianturin peruseriaate pulssien muodostamisessa on sama kuin inkrementtiantureilla. Kuitenkin pelkkien viivojen sijasta pulssikiekolle on värjätty tiettyä asentoa ilmaiseva binäärikoodi usealle sisäkkäiselle kehälle. Yleisimpiä koodeja ovat luonnollinen binäärikoodi, Gray-koodi sekä binääridesimaalikoodi. Pulssianturit jaetaan toimintatansa mukaan kolmeen ryhmään: yksikierroksisiin, monikierroksisiin ja kenttäväyläantureihin, jotka voidaan ohjelmoida joko yksi- tai monikerroksisiksi. (5;6.)

Yksikierroksiset absoluuttianturit mittaavat vain kääntymiskulmaa. Kierroksen jälkeen koodi alkaa uudelleen nolasta. Vähäisen tietomäärän ansiosta joissain tapauksissa voidaan käyttää rinnakkaismuotoista tiedonsiirtoa. Silloin jokaista asentokoodin bittiä vastaa yksi johdin. (5; 6.)

Monikierroksinen absoluuttianturi toimii kuten yksikierros-absoluuttianturi, mutta siinä on myös erillinen kierrosten laskenta. Niissä on resoluution bittimäärän lisäksi varattu omat bitit kierrosten laskemiselle. (5; 6.)

Kenttäväyläanturit sopivat nimensä mukaisesti kenttäväyläasennuksiin. Kenttäväylät sopivat erinomaisesti absoluuttianturien kytkentään suuren tiedonvälityskapasiteettinsa ansiosta. (5; 6)

### 3 Ohjelmoitava logiikka

#### 3.1 Historia

Ensimmäiset PLC:t eli ohjelmoitavat logiikat esiteltiin 1960-luvun lopussa. Idea uudelleenlaiseen ohjausjärjestelmään tuli amerikkalaisen General Motorsin insinööreiltä. Pääsyyinä ohjelmoitavien logiikoiden kehitykseen oli pyrkimys päästä eroon monimutkaisista ja hintavista releohjausjärjestelmistä. Uuden keksinnön oli myös toteutettava General Motorsin tiukat vaatimukset:

- laajentaa staattisten virtapiirien hyödyt 90 %:iin yhtiön koneista
- vähentää ohjausongelmien aiheuttamaa häiriöaikaa koneissa, olla helposti ylläpidettävä ja ohjelmoitava releohjauksesta tutulla ohjelmointikielellä
- komponenttien tuli olla helposti vaihdettavissa ja laajennettavissa tulevaisuutta varten
- toimia teollisuusympäristössä eli kestää pölyä, kosteutta ja värähtelyä.

Prototyypin rakentamiseen kilpailutettiin neljä yhtiötä, jotka olivat Allen-Bradley, Digital Equipment Corporation, Century Detroit ja Bedford Associates. Bedford Associaten viisi työntekijää olivat jo pyynnön aikoihin kehittämässä vaatimukseen sopivaa ohjainta. Työntekijät perustivat oman yrityksen Modiconin, joka toimi yhteistyössä Bedfordin kanssa Modicon 084-ohjaimen parissa. (7; 8.)

Bedford ja Modicon voittivat kilpailutuksen Modicon 084:llä. Se koostui kolmesta eri komponentista; prosessoritaulusta, muistista ja logiikan ratkaisutaulusta. Koska Modicon 084 oli ensimmäinen uudennlainen ohjelmoitava logiikka-järjestelmä, se oli erittäin karu. Siinä ei ollut virtakytkintä ollenkaan ja jäähdytys toimi ainoastaan johtumisen avulla. (7; 8.)

## 3.2 TwinCAT3

TwinCAT3 (The Windows Control and Automation Technology) on Beckhoffin luoma ohjelmisto, jolla minkä tahansa PC-järjestelmän saa muunnettua reaaliaikaiseksi PLC-, NC- tai CNC-kontrolleriksi. Se mahdollistaa sovellusten ohjelmoinnin ja kokoonpanon yhdellä ohjelmalla. Ohjelmisto on integroitu Microsoftin Visual Studion kanssa, ja sen ansiosta ohjelmitavaa logiikkaa ja erilaisia visualisointeja ohjelmille voi tehdä samassa ohjelmassa. Lisäksi ohjelmistoon voi yhdistää Matlab- tai Simulink-sovelluksissa tehtyjä simulaatioita ja funktioita. (9; 10.)

TwinCAT3 tukee kansainvälistä IEC 61131-3-standardia ja sitä kautta sillä on PLCOpen-sertifikaatio. IEC 61131-3-standardi sisältää viisi erilaista graafista ja tekstipohjaista ohjelmointikieltä: Structured Text (ST), Instruction List (IL), Function Block Diagram (FBD), Ladder Diagram (LD) and Sequential Function Chart (SFC). Näiden lisäksi TwinCAT3 tukee myös C/C++ kieliä ohjelmoinnissa. (9; 10.)

## 3.3 Ohjelmointikielien

### 3.3.1 Strukturoitu teksti (Structured Text, ST)

On tekstimuotoinen ohjelmointikieli, jolla on graafisia ohjelmointikieliä yksinkertaisempaa tehdä monimutkaisia laskentaoperaatioita tai ohjelmasilmuksia. Se sisältää modernin ohjelmointikielen tärkeitä elementtejä kuten IF-ehtolauseita, CASE-rakennetta sekä FOR- ja WHILE-silmukkarakenteita. Ohjelman tulkinta on kuitenkin graafisia ohjelmointikieliä hankalampaa, ellei koodin tekijä laita kommentteja koodiin selittämään ratkaisuja. (11; 12; 13.)

### 3.3.2 Käskylista (Instruction List, IL)

Käskylista-ohjelmointikieli on yksinkertainen ja rajoittunut. Yksinkertaisia logiikkaohjauksia ja laskentoja on mahdollista tehdä tiiviisti, monimutkaisempia asioita on hankalampi toteuttaa. Käskylistan käskyt muistuttavat rele- tai logiikkakaavion ohjelmointirakennetta, sillä käskyt koostuvat käskystä ja siihen liittyvästä operandista. (11; 12; 13.)

### 3.3.3 Funktioblokkiohjelmointi (Function Block Diagram, FBD)

Tässä työssä servon ohjelmointi toteutettiin funktioblokkiohjelmoinnilla. Se on yleinen graafinen logiikkaohjelmointikieli. Usein funktioblokkeja käytetään apuna esimerkiksi strukturoidun tekstin kanssa, jolloin pääohjelmalla voidaan kutsua aliohjelmaan tehtyjä funktioblokkeja. Funktioblokkiohjelmoinnin suurimpana etuna ovat helposti nähtävillä olevat in- ja outputit eli sisään ja ulos menevät muuttujat. TwinCATiin on saatavilla monia erilaisia kirjastoja, joilla löytyy funktioblokkeja moneen eri tarkoitukseen. (11; 12; 13.)

### 3.3.4 Relekaavio-ohjelmointi (Ladder Diagram, LD)

On selkeä ohjelmointikieli, joka juontaa juurensa aikaan, jolloin sovelluksissa käytettiin releohjausta. Silmukoiden tekeminen relekaavio-ohjelmoinnilla on erittäin hankalaa, ja jo useamman ehdon sisällyttäminen yhteen virtapiiriin kadottaa selkeyden. (11; 12; 13.)

### 3.3.5 Sekvenssikaavio-ohjelmointi (Sequential Function Chart, SFC)

Sekvenssikaavio-ohjelmoinnissa funktiot sijoittuvat askelketjumaisesti ja koodi siirtyy askeleesta toiseen niiden välissä näkyvien siirtymäehtojen mukaisesti. Kovin hankalia ohjelmia ei ole mahdollista tehdä sekvenssikaaviolla, mutta yksinkertaisiin askelohjauksiin se soveltuu helppolukuisuutensa vuoksi. (11; 12; 13.)

## 4 Laitteisto

Tässä kokoonpanossa käytettiin Metropoliasta valmiiksi löytyvää servokäyttöä. Nykyinen järjestelmä on tehty niin, että ohjainosat ja virtalähteet ovat kiinni metallikiskossa ja moottori on erikseen.

### 4.1 Moottori

Työssä käytetty servomoottori on Beckhoffin valmistama AM81xx-sarjaan kuuluva AM8112-0F20 (kuva 2). Moottorissa on OCT for power and feedback, joka tarkoittaa sitä, että takaisinkytkentä tapahtuu moottorin johdon avulla. Tällöin erilliselle takaisinkytkennälle ei ole tarvetta. Absoluuttianturi mittaa asemaa moottorissa 4096 kierroksen alueella 18 bitin resoluutiolla. Servomoottorin vääntömomentti on 0.38 Nm, teho 0.17 kW ja pyörimisnopeus 4 500 rpm. AM8112 on yhteensopiva käytössä olevan EtherCAT-terminaalin EL7211 kanssa. (14.)



Kuva 2. AM8112-0F20

## 4.2 Servokäytöt

Laitteisto yhdistyy tietokoneeseen Beckhoffin CX9020-moduulin kautta (kuva 3). Se on kompakti CPU-moduuli 1 GHz:n ARM Cortex-A8-suorittimella, kahdella microSD muistikorttipaikalla ja 1 GB DDR RAM-muistilla. Moduulissa on integroitu yhteys Beckhoffin I/O-järjestelmiin. Siinä on monia erilaisia väylävaihtoehtoja, esimerkiksi Ethernet, USB ja PROFIBUS. Tässä tapauksessa yhteys tietokoneeseen hoidetaan Ethernet-yhteydellä. Moduulin käyttöjärjestelmä on Microsoft Windows Embedded Compact 7. TwinCat3 osaa muuttaa ohjelmointikoodin suoraan kyseiseen käyttöjärjestelmään sopivaksi. (15.)



Kuva 3. CX9020 (1.)

Servomootorin ohjaukseen käytettiin CX9020-moduuliin liitettyä Beckhoffin servomootoriterminaalia EL7211 (kuva 4). Kyseisessä EtherCAT-väylän terminaalissa on integroitu resolveria tukeva rajapinta, ja se tarjoaa korkealaatuista suoritustehoa pienessä koossa AM81xx-sarjan servomootoreille. Terminaalin nopea ohjausteknologia perustuu PI-nopeusohjaukseen, ja se tukee nopeita ja dynaamisia asemointitehtäviä. (16.)



Kuva 4. EL7211 (2.)



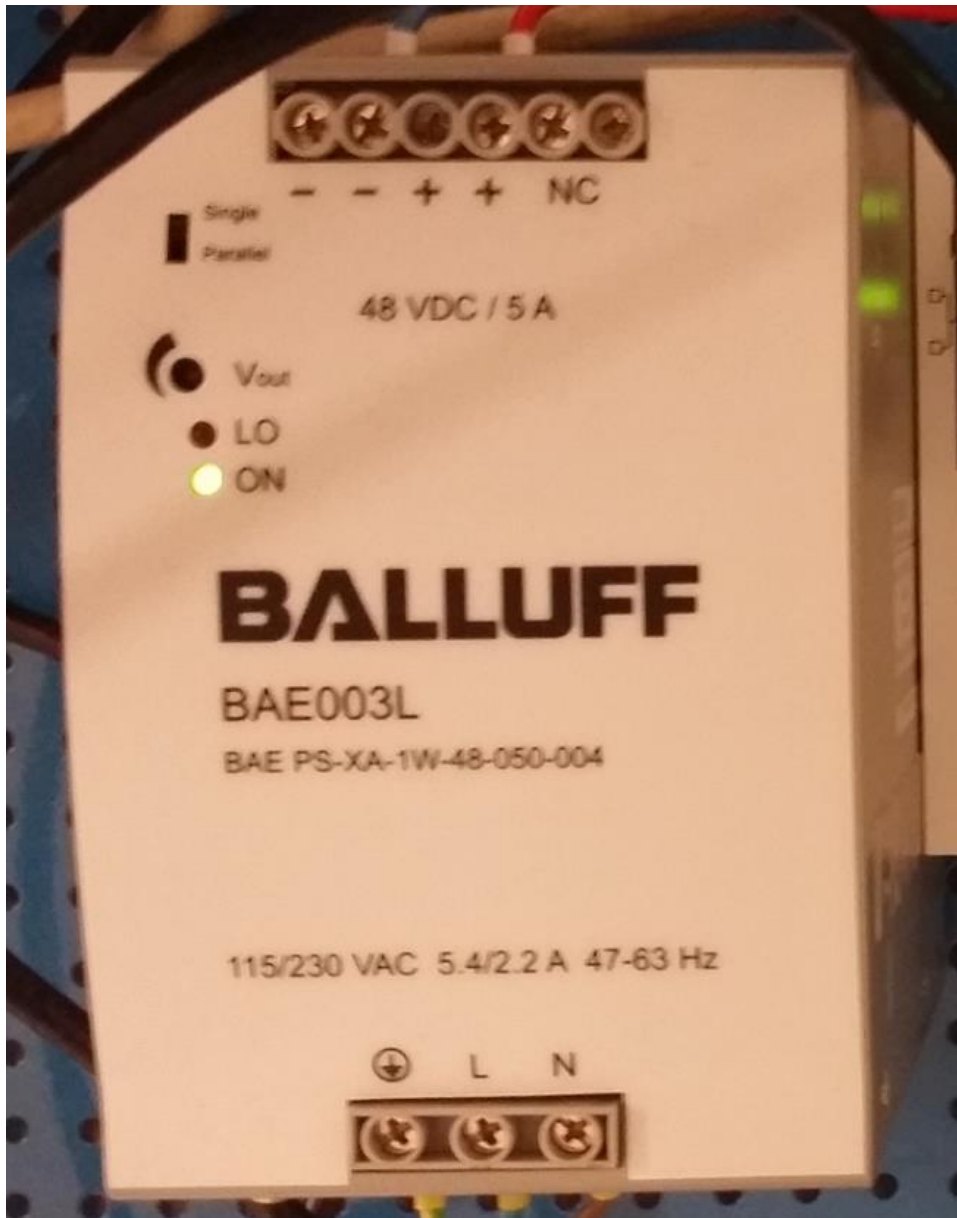
### 4.3 Virransyöttö

CX9020-moduulin virtalähteenä käytettiin BALLUFFin valmistamaa BAE0005-virtalähdettä (kuva 5). Kyseinen virtalähde antaa moduulille tarvittavan 24 V:n tasajännitteen ja 2,5 A:n virran.



Kuva 5. BALLUFF BAE0005

Servomootorin virtalähteenä käytettiin BALLUFFin BAE003L virtalähdettä (kuva 6). BAE003L:stä saadaan 48 V:n tasajännite ja 5 A:n virta.



Kuva 6. BALLUFF BAE003L

#### 4.4 Väylä

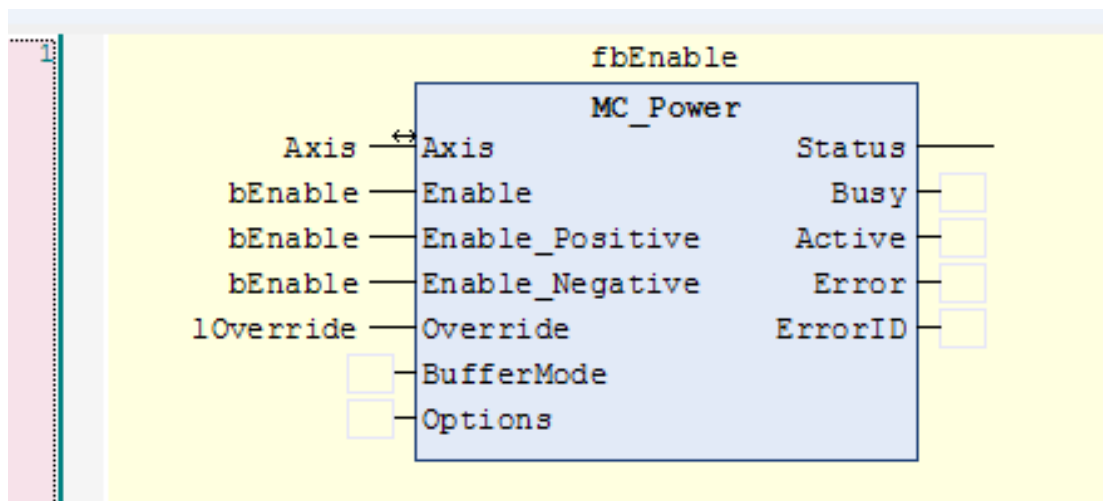
Työssä käytettiin EtherCAT-kenttäväyläprotokollaa. Se on alun perin huhtikuussa 2003 Beckhoffin teollisuuskäyttöön suunnittelema Ethernetiin perustuva väyläprotokolla. EtherCAT kuuluu IEC61158-standardiin ja se sopii automaatiotekniikan sovelluksiin. Sen hyviin ominaisuuksiin kuuluvat pienet vasteajat ja edulliset kustannukset. Lisäksi siinä esiintyy vain vähäistä viiveen vaihtelua, mikä helpottaa tarkkoja synkronointeja. (9.)

## 5 Ohjelma ja käyttöliittymä

Työhön päätettiin toteuttaa yksinkertainen manuaaliohjelma ja käyttöliittymä, jolla servomootoria voidaan ajaa haluttuun paikkaan tai halutulla nopeudella. Näin mahdollistetaan nopeus- ja paikkasäätoisen virityksen harjoittelu. Ohjelmointi ja käyttöliittymä toteutettiin Beckhoffin TwinCat3-ohjelmistolla.

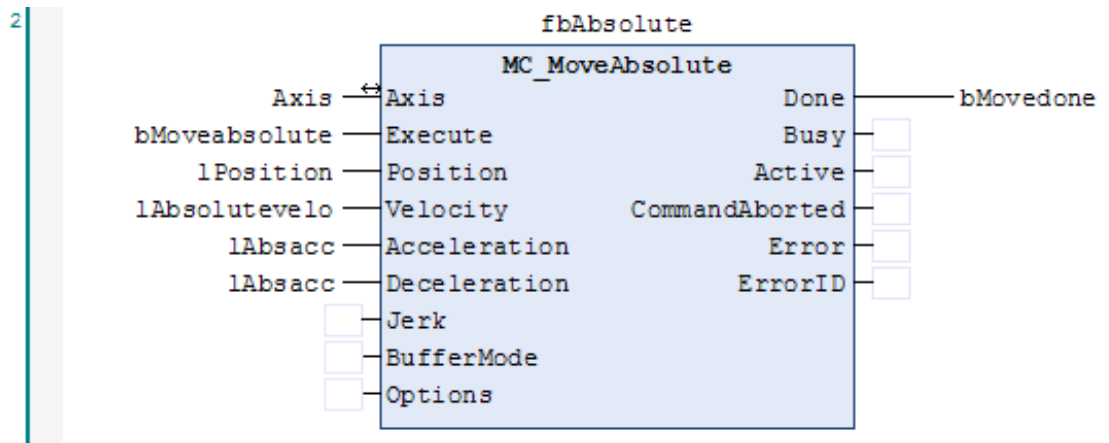
### 5.1 Ohjelma

Ohjelma toteutettiin yksinkertaisesti funktioblokeilla. MC\_Power-blokki (kuva 7) käynnistää järjestelmän. Blokissa oleva lOverride-muuttuja saa arvoja väliltä 0 - 100 ja käytännössä tämä tarkoittaa sitä, kuinka suurella teholla (0 - 100%) servomootori toimii. Mikäli MC\_Power ei ole aktiivinen, mikään muu funktioblokki ei toimi.



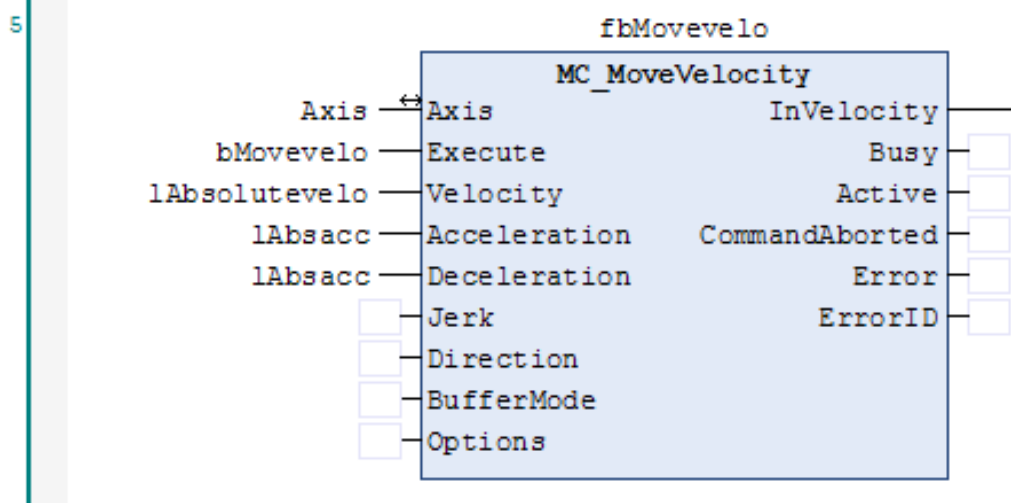
Kuva 7. MC\_Power

MC\_MoveAbsolute (kuva 8) antaa servomootorille käskyn liikkua tiettyyn asemaan. Asema määräytyy IPosition-muuttujan mukaan. Lisäksi käyttäjän on mahdollista määrittää nopeus, jolla asemaan ajetaan, muuttujalla IAbsolutevelo, sekä kiihtyvyys ja hidastuvuus muuttujalla IAbsacc.



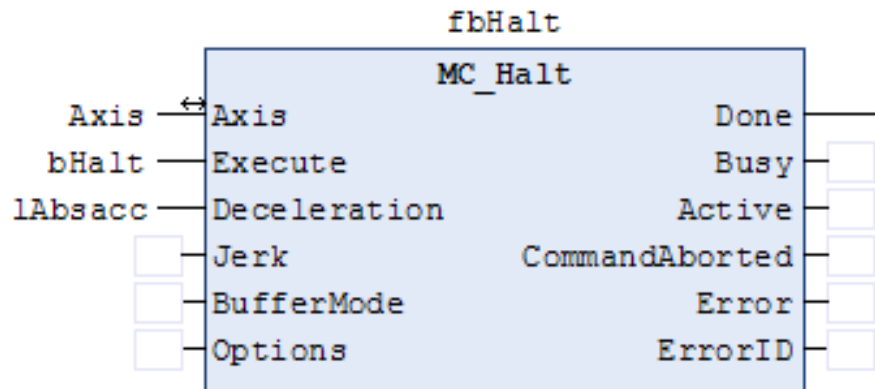
Kuva 8. MC\_MoveAbsolute

MC\_MoveVelocity blokin (kuva 9) avulla servomootoria voi ajaa tietyllä nopeudella ilman paikan määrittystä. Samoin kuin aiemmassa blokissa IAbsolutevelo määrittää nopeuden ja IAbsacc määrittää kiihtyvyyden ja hidastuvuuden.



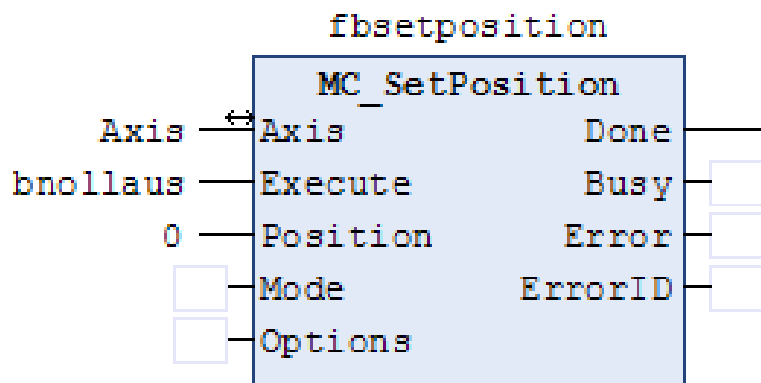
Kuva 9. MC\_MoveVelocity

MC\_Halt-funktioblokki (kuva 10) on pysäytys. Kun käyttäjä ajaa esimerkiksi MC\_MoveVelocity-blokillä tietyllä nopeudella, MC\_Halt on keino pysäyttää moottorin liike. Tässä hidastuvuus määräytyy lAbsacc-muuttujan arvon mukaan.



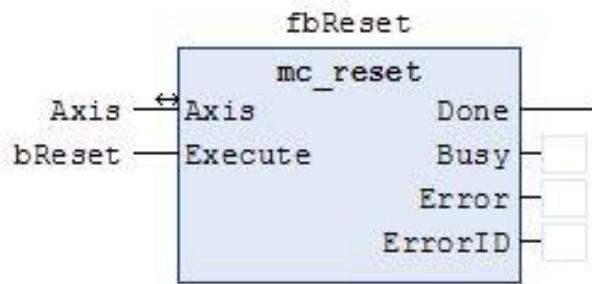
Kuva 10. MC\_Halt

MC\_SetPosition-blokillä (kuva 11) voidaan määrätä servomoottorin sen hetkinen asema tietyksi asemaksi. Tässä tapauksessa kyseistä blokkia käytetään paikan nollaamiseen, jotta paikka voidaan määrätä nolaksi ilman, että servomoottori täytyy aina ajaa nolakohtaan. Tämä helpottaa scopen käyttöä mahdollistaen ajamisen aloittamisen nolasta. Varsinaisissa servoratkaisuissa paikkaa ei näin kuulu nolata.



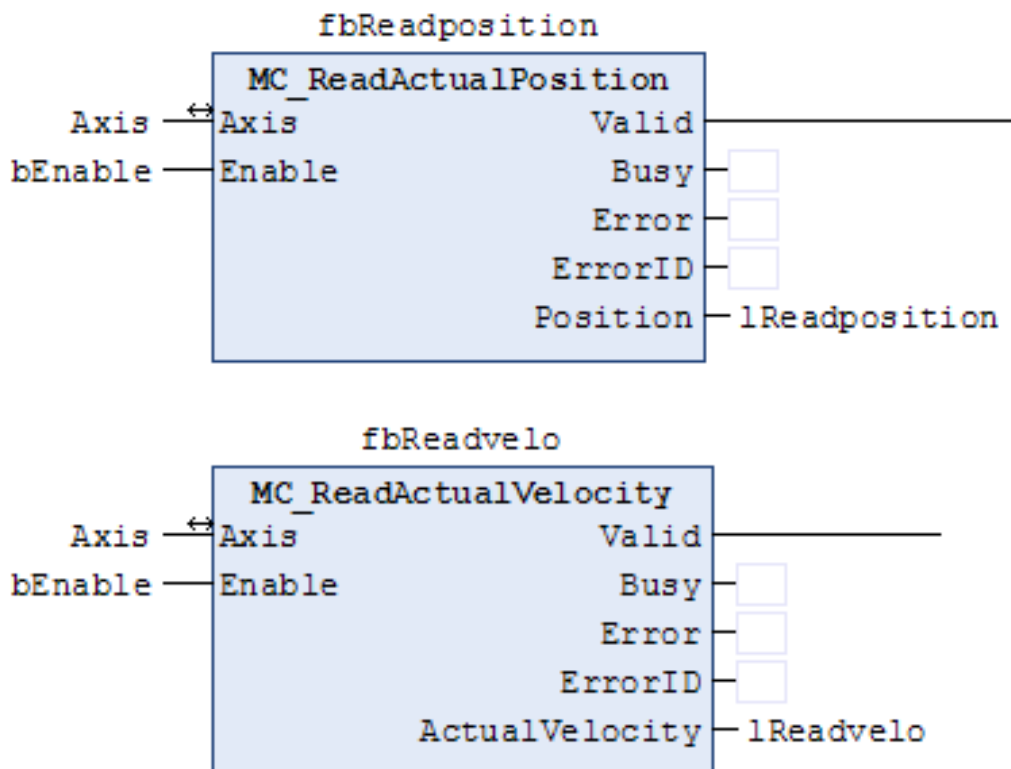
Kuva 11. MC\_SetPosition

Kuvassa 12 näkyvällä MC\_Reset-blokilla voidaan kuitata virheilmoitukset.



Kuva 12. MC\_Reset

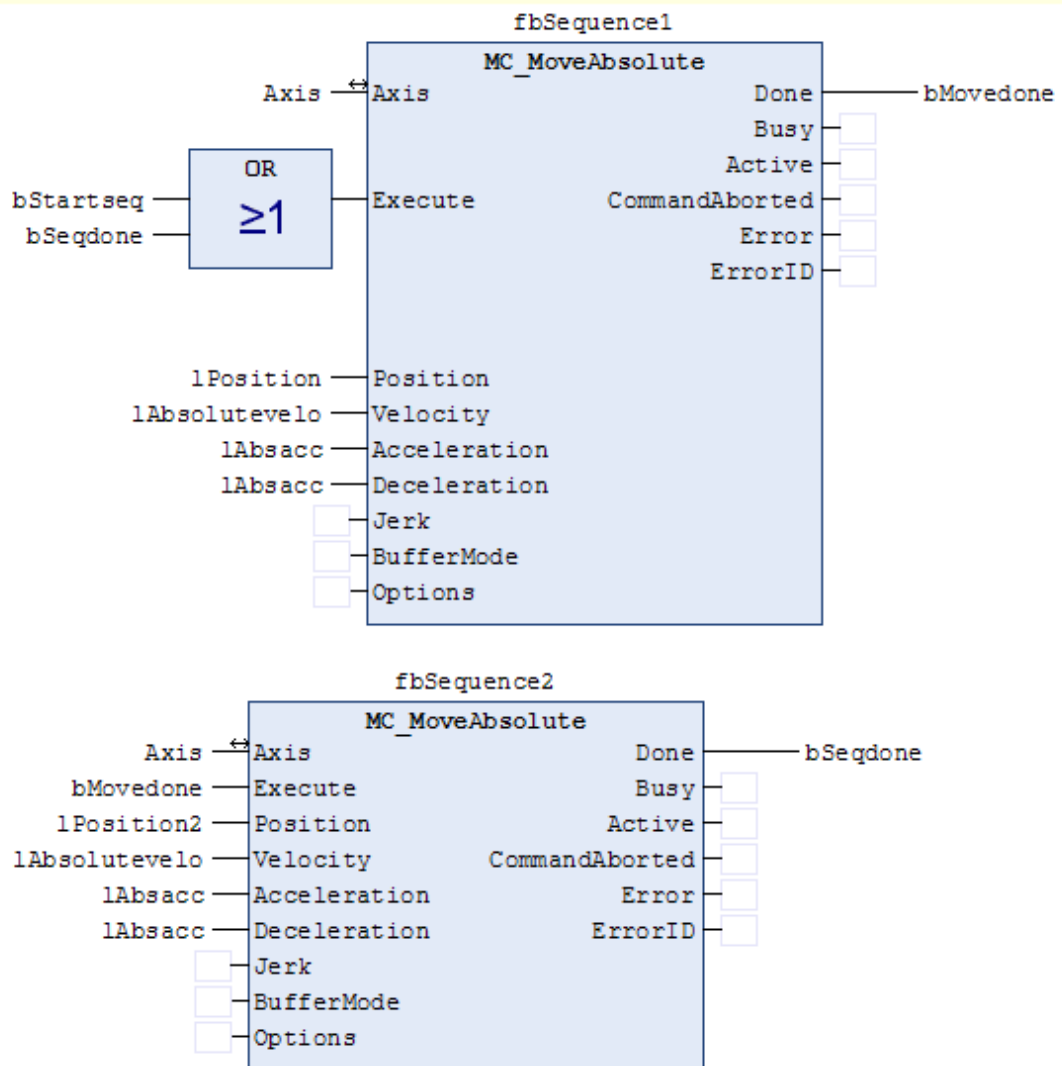
Käyttöliittymään tulevat paikka- ja nopeustiedot saadaan MC\_ReadActualPosition- ja MC\_ReadActualVelocity-funktioblokeilla (kuva 13). Blokit aktivoituvat, kun MC\_Power-blokki on aktiivisena. Ne lukevat moottorin absoluuttianturin antamat tiedot ja paikka ja nopeus saadaan luettua lReadposition- ja lReadvelo-muuttujista.



Kuva 13. MC\_ReadActualPosition ja MC\_ReadActualVelocity

Ohjelmaan tehtiin vielä erikseen kuvassa 14 näkyvä funktioblokkipari, jolla voi ajaa mootoria sekvenssinä. Käyttäjä määrittää paikat muuttujiin IPosition sekä IPosition2, jolloin servon saa ajamaan näiden asemien väliä. Sekvenssi on toteutettu kahdella MoveAbsolute-blokillla.

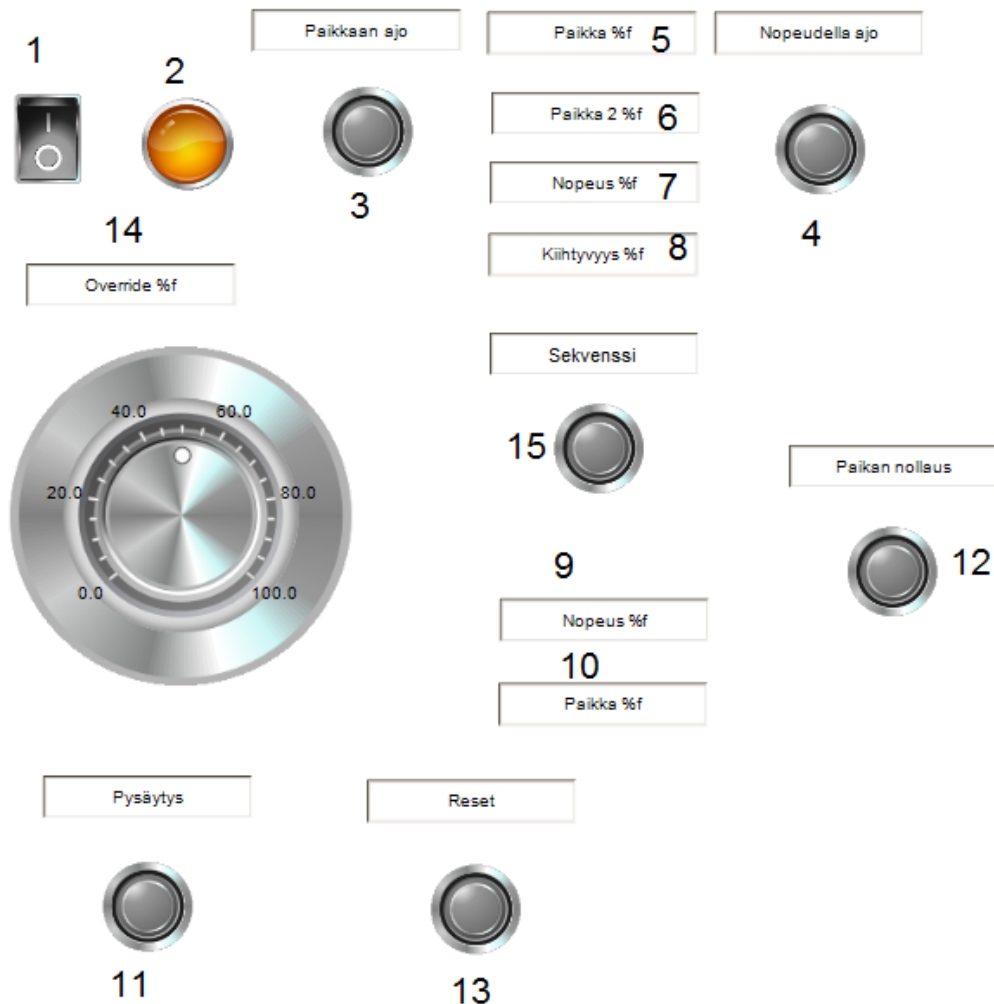
Ensimmäisen blokin voi aktivoida sekvenssin aloituspainikkeella. Toinen blokki aktivoituu, kun ensimmäiseen blokkiin määritelty asema on saavutettu. Tämän jälkeen ensimmäinen blokki aktivoituu uudelleen, kun toisessa blokissa määritelty asema on saavutettu. Nopeus ja kiihtyvyys voidaan myös määrittää erikseen, jolloin sekvenssiajo vastaa lähes virityksessä käytettävää velo step sequence-funktiota.



Kuva 14. Sekvenssi funktioblokki pari

## 5.2 Käyttöliittymä

Tarkoituksena oli saada aikaan helppokäyttöinen käyttöliittymä, jolla servomoottoria voidaan ajaa ja samalla scopella tarkkailla viritystä. Kuva 15 havainnollistaa käyttöliittymän painikkeita.



**Kuva 15. Käyttöliittymä**

MC\_Power-blokki eli järjestelmän kytkeminen toimintaan tapahtuu napista 1. Merkkivalo (2) ilmoittaa, mikäli järjestelmä on päällä. Paikkaan ajo eli MC\_MoveAbsolute-blokin execute on kytketty nappiin 3. Nopeudella ajo tapahtuu napista 4. Sekvenssijonon saa kytkeä päälle napista 15. Sekvenssi ja nopeudella ajo voidaan pysäyttää ainoastaan pysäytysnapilla.

Käyttäjä voi määrittää halutun aseman, johon ajetaan kohdasta 5, ja toisen aseman sekvenssijonon varten kohdasta 6. Kun käyttäjä painaa tekstikenttää, avautuu numeronäyttö, josta voi määrittää haluamansa aseman. Näihin kirjoitetut luvut ovat muuttujan IPosition



ja IPosition 2 arvot MC\_MoveAbsolute-blokeissa. Samalla tavalla toimivat tekstikentät 7 ja 8, nämä määrittävät muuttujien IAbsolutevelo ja IAbsacc arvot.

Overridea eli moottorin tehoa väliltä 0 - 100% voidaan muuttaa säätimen 14 avulla, joka on kytketty MC\_Power:in IOverride-muuttujaan. Säätimen yläpuolella oleva tekstikenttä ilmoittaa valitun overriden. Servomoottorin hetkellinen paikka ja nopeus näkyvät tekstikentissä 9 ja 10. Nämä on kytketty IReadposition- ja IReadvelo-muuttujiin.

Nappi 11 ohjaa MC\_Halt-blokkia, jolla servomoottorin liikkeen voi pysäyttää. Paikan nollaus tapahtuu napista 12, joka on yhdistetty MC\_SetPosition-funktioblokkiin. Nappi 13 eli reset on yhdistetty MC\_Reset-blokkiin ja sillä voidaan nollata mahdolliset virheilmoitukset.

## 6 Viritys

Servomoottorien viritys tapahtuu muuttamalla niissä olevien säätimien parametreja. P- ja PI-säätimelliset servot voidaan virittää kokeellisesti, mutta PID-säätimien virityksessä on hyvä käyttää esimerkiksi Ziegler-Nicholsin värähtelymenetelmää.

### 6.1 Servon säätimet

Yleisimmät servojen käytössä olevat säätimet ovat P-, PI- ja PID-säätimiä. Säätimien tarkoitus on vähentää prosessissa olevien häiriötekijöiden vaikutusta. Säätojärjestelmän pyrkimys on pitää eresignaali lähellä nollaa, jotta järjestelmä automaattisesti korjaisi häiriöiden muutokset. Erosignaalin vahvistuksen määrä vaikuttaa säädön tarkkuuteen. Signaalin vahvistusta ei voi kasvattaa äärettömästi, koska se aiheuttaa värähtelyjä ja epävakautta. (4; 6.)

#### 6.1.1 P-säädin

P-säätimen vahvistus  $K_p$  kasvattaa tai vähentää nopeutta virheen suhteessa. Liian suuri vahvistusarvo saa järjestelmän värähtelemään, kun taas liian pieni arvo voi aiheuttaa suurta jättämää. (4; 6.)

### 6.1.2 PI-säädin

PI-säädin koostuu kahdesta osiosta, aiemmin esitellystä P-säätimestä sekä siihen liitetystä I-termistä. I-termiä  $K_i$  kutsutaan integrointivakioksi ja se muuttaa nopeutta samassa suhteessa kuin virheiden yhteenlaskettu summa ajan funktiona. Kun erosuure on nolasta poikkeava, ohjesuure  $P$  ajaa eroa korjaavaan suuntaan. Samalla I-termin vaikutuksesta ero pienenee nolaa kohti. PI-säädin toimii kuten P-säädin, mikäli I-termi on suuri. Suuri I-termi kuitenkin lisää värähtelyä ja epästabiilisuutta poistaen pysyvän poikkeaman. (4; 6.)

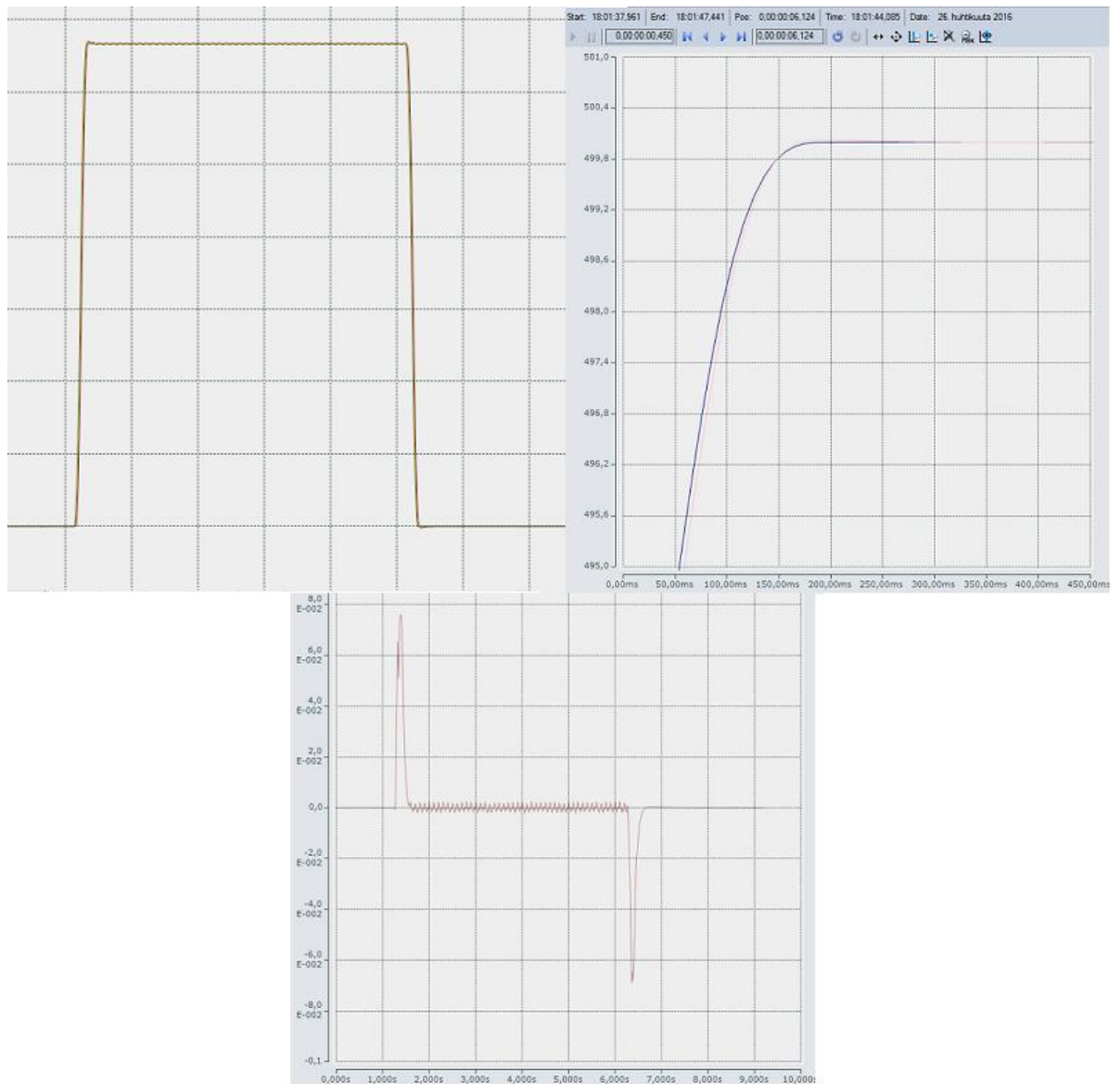
### 6.1.3 PID-säädin

PID-säädin on yleinen esimerkiksi NC-ohjaimien asemasäätimissä. Siinä on P- ja I-termin lisäksi derivointivakio D-termi. D-termin tehtävä on muuttaa nopeutta asemavirheen muutoksen funktiona ja sillä tavoin vakauttaa systeemiä. Suuri D-termi tasapainottaa, nopeuttaa ja vähentää värähtelyä. Se on kuitenkin herkkä kohinalle ja viiveille. PID-säätimen etuna verrattuna PI-säätöön on nopea reagointi muuttuvaan erosuureeseen, mitä kutsutaan ennakoivaksi säädöksi. PID-säädintä voi käyttää P-säätimenä, PI-säätimenä tai PD-säätimenä asettamalla ei halutun termin nolaksi. (4; 6.)

## 6.2 Työssä käytetyn servon viritys

Tässä työssä servo viritettiin nopeus- ja paikkaohjattuna kokeellisesti. Tämä tarkoittaa sitä, että aluksi paikkaohjaimen  $K_p$ -arvo asetettiin nolaksi. Keskityttiin aluksi säätämään vain nopeusohjainta. Kiihtyvyys asetettiin mahdollisimman suureksi, jotta servo saavuttaa tavoitenopeuden hetkessä. Servo laitettiin pyörimään sekvenssitoiminnolla kahden eri paikan välillä tietyllä nopeudella. Sen jälkeen säädettiin nopeusohjaimen P- ja I-termit kohdalleen.

Kun servo saatiin viritettyä nopeusohjattuna, siirryttiin virittämään paikkaohjainta. Servoa ajettiin paikkaan ajo-liikkeellä ja säädettiin paikkaohjaimen P-säädintä. Kuvassa 16 vasemmalla näkyy nopeusohjattuna, kuinka punainen moottorin nopeus seuraa asetettua vihreää nopeutta. Nopeus saavutetaan tarkasti, mutta siinä esiintyy hieman värinää. Oikealla näkyy paikkasäädön onnistuminen. Vaaleanpunainen servon paikka asetuu melko hyvin halutun sinisen arvon kohdalle. Alemmassa kuvassa näkyy vielä asema ero halutun ja saavutetun aseman välillä aikaa kohti.

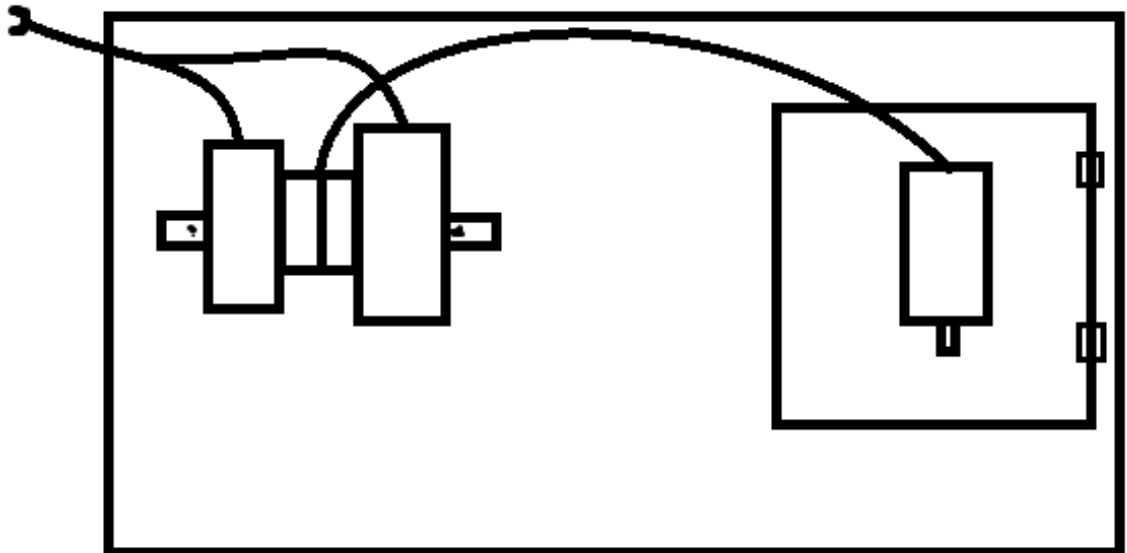


Kuva 16. Virituksen tulokset

## 7 Kokoonpano

### 7.1 Suunnitelma

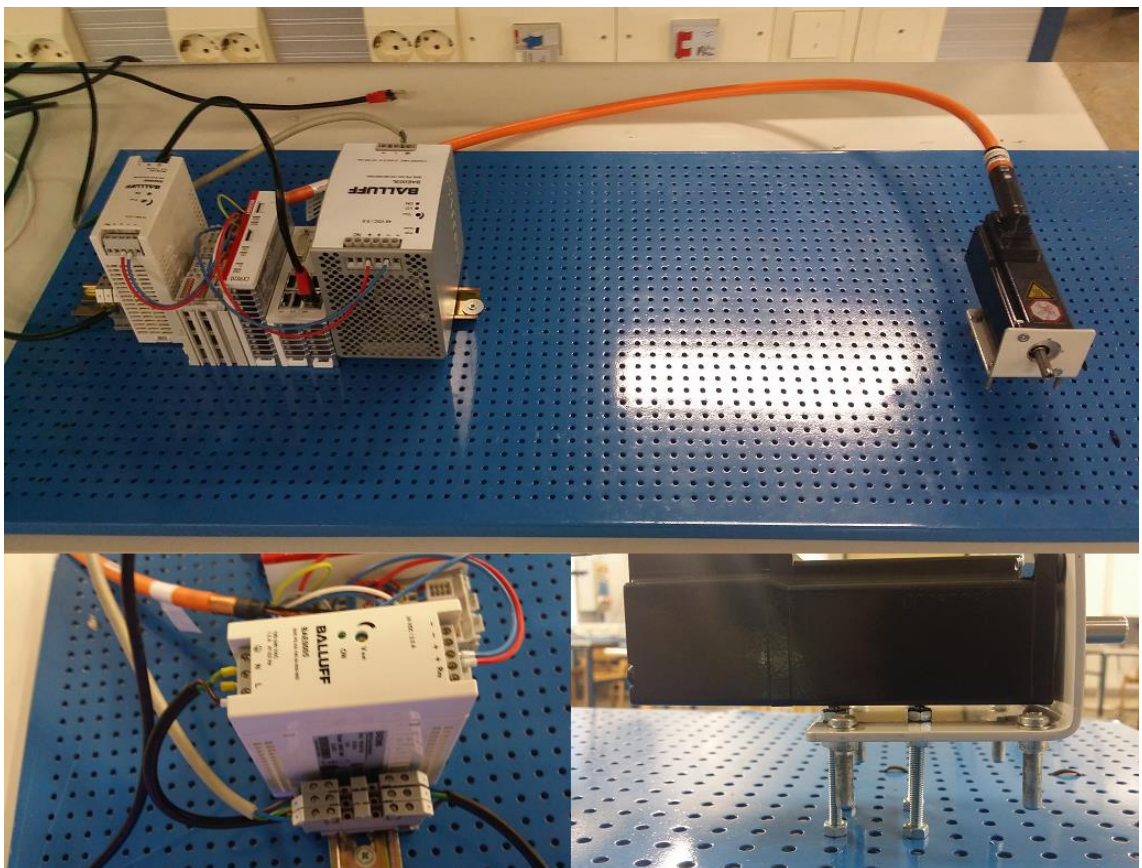
Kokoonpanossa haluttiin saada kaikki rakenteeseen kuuluvat osat samalle alustalle kiinni. Aiemmin virtalähteet ja servokortit olivat yhdellä c-kiskolla ja moottori vain johdon päässä. Tarkoituksena oli asentaa c-kisko kiinni alustaan ja moottorille oma jalusta, jotta kuormalla on mahdollisuus pyöriä. Virtalähteistä lähti kaksi virtajohtoa, jotka haluttiin yhdistää yhdeksi. Moottorille tuleva kuorma päätettiin sorvata koululla ja tilata sille sovite, jotta sen saa akseliin tiukasti kiinni. Moottorin ympärille haluttiin rakentaa suoja läpinäkyvästä polykarbonaatista, jotta vahinkoja ei tapahtuisi, mikäli kuorma irtoaisi vauhdissa. Kuvassa 17 on karkea suunnitelmakuva toteutuksesta. Suoja laitettaisiin kiinni sara-noilla, jotta kuorman lisääminen on helppoa.



Kuva 17. Suunnitelman piirros

## 7.2 Toteutus

Kokoonpano sijoitettiin metalliselle alustalle, johon se saatiin hyvin kiinni. C-kisko sijoitettiin toiselle puolelle alustaa ja moottori vastakkaiselle puolelle. Virtalähteiden johdot yhdistettiin kolmella riviliittimellä ja ne asennettiin jo olemassa olevaan c-kiskoon niin, että sähköiskujen vaaraa ei ole. Moottorin jalustana käytettiin koululta löytynyttä Boschin pneumaattisen painonapin kehikkoa. Siihen porattiin reiät moottorin kiinnitykseen sekä kehikon kiinnittämiseksi alustaan. Moottorin ja kehikon toisiinsa liittäneisiin ruuveihin laitettiin väliin kumiset prikot vähentämään värinää alustassa. Kuvassa 18 ylimpänä näkyy koko järjestelmä ilman kuormaa. Vasemmassa kuvassa näkyvät riviliittimet ja sähköjohdot liittäntä. Oikealla lähikuva moottorin kiinnityksestä. Suojaa ei rakennettu, koska koululle ei ehditty tilaamaan polykarbonaatilevyjä.



Kuva 18. Valmis asennus

Kuvassa 19 näkyvä kuorma tehtiin itse sorvaamalla teräksestä. Sen ulkohalkaisija on 100 mm, sisähalkaisija 15 mm ja paksuus 12 mm. Kuorma saadaan akseliin kiinni tukevasti SKS:ltä tilatulla ETP-MINI 8-kiinnitysholkilla. Kiinnitysholkin sisähalkaisija valittiin moottorin akselin halkaisijan (8mm) mukaan. Ulkohalkaisija valitulla holkillä on 15 mm, joten kuormaksi haluttu kappale sorvattiin sen mukaan. Kiinnitysholkin hitausmomentti  $J$  on  $3,3e^{-6} \text{ kgm}^2$ . Kuorman hitausmomentti saadaan laskemalla yhtälö

$$J = \frac{1}{2} * mr^2$$

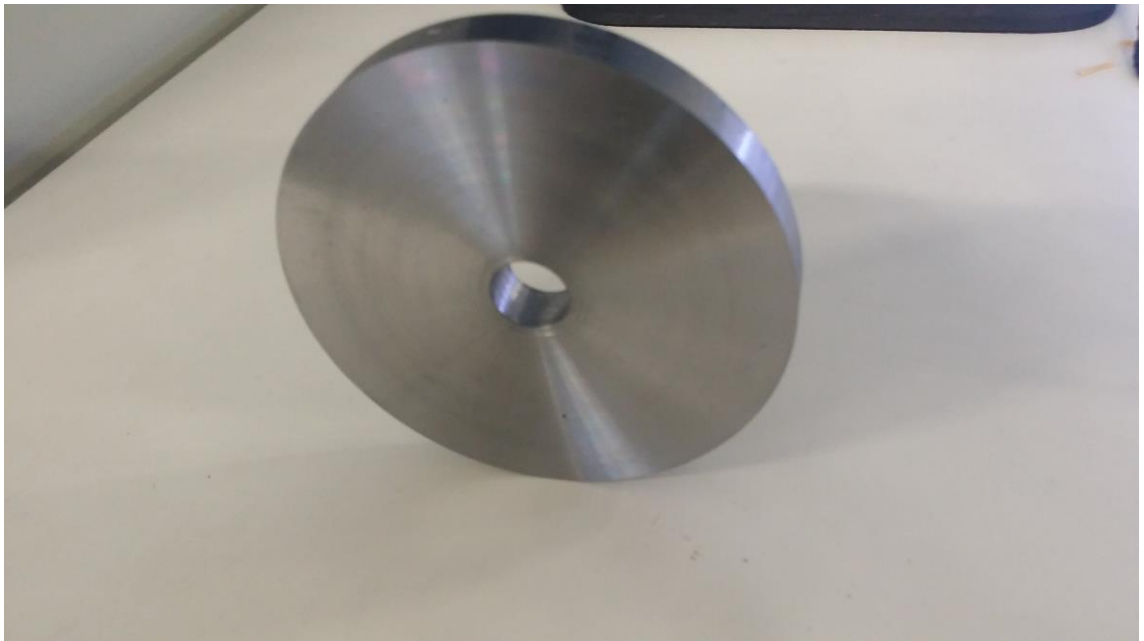
jossa  $m$  ja  $r$  ovat kuorman massa ja säde.

Kuorman massa  $m$  on 723 g, joten hitausmomentti  $J$  on  $9,0375e^{-4} \text{ kgm}^2$ . Holkin ja kuorman yhteiseksi hitausmomentiksi moottorin akselille saadaan siis  $9,0705e^{-4} \text{ kgm}^2$ . Moottorin nimellinen kulmakihtyvyys  $\alpha$  kuorman kanssa saadaan laskettua kaavalla

$$M = J * \alpha$$

jossa  $M$  on moottorin nimellisvääntömomentti ja  $J$  akselilla oleva hitausmomentti.

Moottorin nimellismomentin ollessa 0.38 Nm kulmakihtyvyydeksi  $\alpha$  saadaan  $420 \frac{\text{rad}}{\text{s}^2}$ .



Kuva 19. Kuormakiekkö

## 8 Yhteenveto

Servokäyttö saatiin sijoitettua hyvin alustalle. Johtojen yhdistäminen onnistui siististi ja nyt servon käyttöön tarvitaan vain yksi virtajohto sekä Ethernet-johto tietokoneeseen yhdistämiseksi. Moottorille saatiin rakennettua hyvä kiinnityspala, johon se saatiin tiukasti kiinni. Moottorin ja kiinnityspalan väliin on mahdollista lisätä esimerkiksi ohutta kumimattoa, jolla mahdollista tärinää on helppo vaimentaa.

Moottorin ympärille tarkoitettua suojaa ei ehditty rakentamaan osien puutteen vuoksi. Sen voi kuitenkin helposti lisätä nykyiseen kokoonpanoon, koska sille varattiin tila kiinnitykseen moottorin ympäriltä.

Luotiin PLC-ohjelma sekä graafinen käyttöliittymä servomoottorin ohjaukseen. Lisäksi asennettiin valmiiksi scope oikeilla asetuksilla ja viritettiin akseli tyhjänä. Akseliin tuli irrotettava kuorma, jonka avulla virittämistä voi harjoitella.

Ohjelma saatiin toimimaan moitteettomasti. Sitä on helppo käyttää myös muihin soveluksiin monipuolisten ajomahdollisuuksiensa ansiosta. Lisäksi graafista käyttöliittymää voi käyttää suoraan toisiin ohjelmiin, mikäli ohjelmissa käyttää samannimisiä muuttujia.

## Lähteet

- 1 First Servomechanism. Verkkodokumentti. <http://rexroth-us.com/blog/the-first-servomechanism/> Luettu 4.5.2016.
- 2 Mustajärvi Pasi. 2013. *Projektioppimisympäristö servomoottoreille*. Opinnäytetyö. Tampereen ammattikorkeakoulu. Kone- ja tuotantotekniikka.
- 3 Vainikainen Emma. 2015. *Opetuskäyttöön tulevan servokäytön kokoonpanon suunnittelu ja käyttöönotto*. Insinööriyö. Metropolia ammattikorkeakoulu. Kone- ja tuotantotekniikka.
- 4 Hanhisalo Harri. 2008. *Servomoottorin ohjain*. Insinööriyö. Kajaanin ammattikorkeakoulu. Tekniikan ja liikenteen ala.
- 5 Pulssianturien teoriaa. Verkkodokumentti. [http://www.oem.fi/Tuotteet/Anturi/Pulssianturit/Yleista/Pulssianturien\\_teoriala/825723-526144.html](http://www.oem.fi/Tuotteet/Anturi/Pulssianturit/Yleista/Pulssianturien_teoriala/825723-526144.html) Luettu 4.5.2016
- 6 Servojärjestelmän viritys. Verkkodokumentti. [http://automation.tkk.fi/attach/AS-0-2230/Labratyo4\\_2008.pdf](http://automation.tkk.fi/attach/AS-0-2230/Labratyo4_2008.pdf) Luettu 4.5.2016.
- 7 Mannermaa Jukka. 2012. *Pakkauskoneen käyttöliittymän suunnittelu ja käyttöönotto*. Oulun seudun ammattikorkeakoulu. Automaatiotekniikan koulutusohjelma.
- 8 History of Control History of PLC and DCS. Verkkodokumentti. [http://www.control.lth.se/media/Education/DoctorateProgram/2012/HistoryOfControl/Vanessa\\_Alfred\\_report.pdf](http://www.control.lth.se/media/Education/DoctorateProgram/2012/HistoryOfControl/Vanessa_Alfred_report.pdf) Luettu 4.5.2016.
- 9 TwinCAT3 eXtended Automation. Verkkodokumentti. [ftp://ftp.beckhoff.com/Document/Catalog/Beckhoff\\_TwinCAT3\\_042012\\_e.pdf](ftp://ftp.beckhoff.com/Document/Catalog/Beckhoff_TwinCAT3_042012_e.pdf) Luettu 3.5.2016.
- 10 Introduction into IEC 61131-1 General Information. Verkkodokumentti. [http://www.plcopen.org/pages/tc1\\_standards/iec\\_61131\\_1/](http://www.plcopen.org/pages/tc1_standards/iec_61131_1/) Luettu 3.5.2016.
- 11 Automaatiotekniikan laboraatiot 2010-2011. Verkkodokumentti. <http://miniweb.lpt.fi/automaatio/opetus/tiedotteet/labrat/mek/Loglab15.htm> Luettu 6.5.2016.
- 12 CX-One ja logiikkaohjelmointi. Verkkodokumentti. [https://www.myomron.com/downloads/9.Local%20Material/Finnish/CX-One%20ja%20logiikkaohjelmointi%202009\\_2.pdf](https://www.myomron.com/downloads/9.Local%20Material/Finnish/CX-One%20ja%20logiikkaohjelmointi%202009_2.pdf) Luettu 6.5.2016.
- 13 Kokkonen Juho. 2009. *Arkittamon hylynkäsittelyn automaatiojärjestelmän modernisointi*. Kemi-Tornion ammattikorkeakoulu. Automaatiotekniikka.
- 14 AM8112 Servomotor 0.38 Nm. Verkkodokumentti. [http://www.beckhoff.be/english.asp?drive\\_technology/am8112.htm](http://www.beckhoff.be/english.asp?drive_technology/am8112.htm) Luettu 5.5.2016.



- 15 CX9020 Basic CPU module. Verkkodokumentti. [http://www.beckhoff.co.jp/english.asp?embedded\\_pc/cx9020.htm](http://www.beckhoff.co.jp/english.asp?embedded_pc/cx9020.htm)/ Luettu 5.5.2016.
- 16 EL7211 Servomotor terminal 50 V DC, 4.5 A. Verkkodokumentti. <http://www.beckhoff.com/english.asp?EtherCAT/el7211.htm> Luettu 5.5.2016.

**Kuvalähteet**

1. [http://www.beckhoff.co.jp/english.asp?embedded\\_pc/cx9020.htm](http://www.beckhoff.co.jp/english.asp?embedded_pc/cx9020.htm)
2. <http://www.pddnet.com/sites/pddnet.com/files/pdBeckhoff0729.jpg>