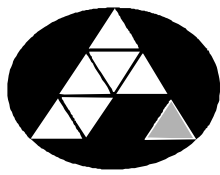


POHJOIS-KARJALAN AMMATTIKORKEAKOULU
Tietotekniikan koulutusohjelma

Jarno Romppanen

KELAAJALAITTEEN AUTOMATISOINTI

Opinnäytetyö
Syyskuu 2011



POHJOIS-KARJALAN
AMMATTIKORKEAKOULU

OPINNÄYTETYÖ
Syyskuu 2011
Tietotekniikan koulutusohjelma

Karjalankatu 3
80200 JOENSUU
p. (013) 260 6800

Tekijä
Jarno Romppanen

Nimeke
Kelaajalaitteen automatisointi

Toimeksiantaja
Pohjois-Karjalan ammattikorkeakoulu

Tiivistelmä

Tämän opinnäytetyön ensisijaisena tarkoituksena oli toteuttaa Pohjois-Karjalan ammattikorkeakoululla olevalle kelaajaprosessille käyttöliittymä ja sekvenssiohjelma. Käyttöliittymä toteutettiin paneelin lisäksi PC-valvomolle. Tämän lisäksi oli tarkoitus toteuttaa ohjaukset servo- ja taajuusmuuttajamoottoreille. Tulevaisuudessa kelaajalaitteen olisi tarkoitus olla opetuskäytössä.

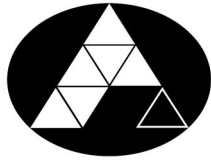
Kyseessä on toiminnallinen opinnäytetyö, joka piti sisällään suunnittelua ja toteutusta. Käyttöliittymän suunnittelun pohjana oli Siemensin toteuttama esimerkkiprojekti taajuusmuuttajaohjaimelle. Tätä muokkaamalla saatiin toteutettua lohko ja käyttöliittymäikkunat myös servo-ohjaimelle. Käytettävyyttä ja hahmolakia pyrittiin noudattamaan käyttöliittymiä tehdessä.

Päätavoitteet eli sekvenssin ja käyttöliittymien toteutus saavutettiin. Servon ja taajuusmuuttajan ohjauksien osalta tavoitteita ei täysin saavutettu, koska toinen servo-ohjain ei toiminut, joten tämä rajattiin pois. Prosessilla on vielä jatkokehitys aiheita, mutta sitä voidaan jo demonstroida oppilaille. Ongelmatilanteet veivät opinnäytetyössä yllättävänkin paljon aikaa, mutta onneksi näistä selvittiin osittain omin avuin ja osittain ohjaajan avustuksella.

Kieli
suomi

Sivuja 40
Liitteet 10
Liitesivumäärä 14

Asiasanat
käyttöliittymä, sekvenssi, servo, taajuusmuuttaja



NORTH KARELIA
UNIVERSITY OF APPLIED SCIENCES

THESIS
September 2011
**Degree Programme in Information
Technology**
Karjalankatu 3
FIN 80200 JOENSUU
FINLAND
Tel. 358–13-260 6800

Author
Jarno Romppanen

Title
Automation of Reeling Machine

Commissioned by
North Karelia University of Applied Sciences

Abstract

The purpose of this thesis was to plan and produce user interfaces and a sequence program for a reeling machine. User interfaces were produced to touch screen panel and PC. Another aim of this work was to implement control for servo motor and frequency changer. The device will be used for teaching purposes in North Karelia University of Applied Sciences.

Siemens's example project for frequency changer was used as source in planning the user interface. Making some changes to this project, a block was implemented and the user interface for the servo controller, too. Obeying usability and Gestalt grouping laws was important when planning the interfaces.

The main goals, which were sequence program and user interfaces, were reached in this thesis. As for the control for the servo and frequency changer, the goals were not fully reached because the other servo controller did not work so it had to be left out. The process has still some further development subjects, but it can, however, be demonstrated for students. Solving the problems took surprisingly much time, but none of those problems were too bad to be solved by myself or with the help from the supervisor.

Language
Finnish

Pages 40
Appendices 10
Pages of Appendices 14

Keywords
frequency changer, sequence, servo, user interface.

Sisältö

Tiivistelmä

Abstract

1	Johdanto	6
2	Kelaajaprosessi.....	7
2.1	Järjestelmäkaavio ja laitteisto	7
2.2	Laitteistorakenne	9
2.3	Prosessin väylät.....	10
2.3.1	Profibus DP.....	10
2.3.2	AS-interface.....	10
2.3.3	Ethernet	11
3	Taajuusmuuttajan ohjaus.....	12
3.1	Taajuusmuuttaja	12
3.2	Moottorin arvot.....	12
3.3	Taajuusmuuttajaohjaimen parametointi.....	13
3.4	Taajuusmuuttajalohko.....	14
4	Servon ohjaus.....	17
4.1	Servo	17
4.2	Servolohkon toiminta	17
5	Sekvenssiohjaus.....	18
5.1	Oletetut käyttäjät.....	18
5.2	Sekvenssin suunnittelu	19
5.3	Sekvenssin toteutus.....	19
5.4	Testauksen jälkeiset muutokset.....	21
6	Käyttöliittymän teoria.....	22
6.1	Lähtökohta	22
6.2	Hahmolait.....	22
6.3	Selkeys ja hahmotettavuus	23
6.4	Luettavuus	24
6.5	Siemensin tekemän piirinäytön tarkastelu.....	24
7	Käyttöliittymien suunnittelu	27
7.1	Paneelin käyttöliittymäikkunat.....	27
7.2	PC:n käyttöliittymäikkunat.....	31
8	Käyttöliittymien toteutus	33
8.1	WinCC Flexible	33
8.2	Paneelin käyttöliittymän toteutus	35
8.3	PC-käyttöliittymän toteutus	35
8.4	Testaus.....	36
9	Pohdinta.....	37
	Lähteet.....	39

Liitteet

Liite 1	Taajuusmuuttajalohko
Liite 2	Servolohko
Liite 3	Toimintasekvenssi
Liite 4	Sekvenssin tila- ja ohjausbitit
Liite 5	Sekvenssi-ohjelma
Liite 6	Paneelin käyttöliittymä

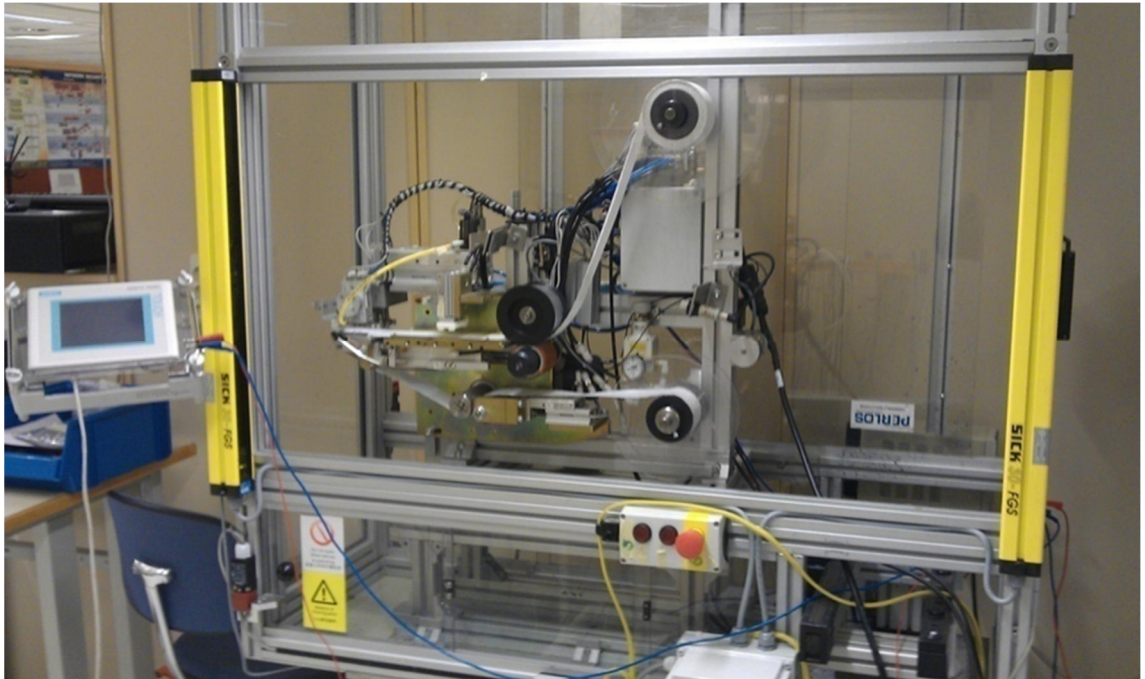
Liite 7	Faceplaten näkymä yleisnäkymässä
Liite 8	PC-valvomon käyttöliittymä
Liite 9	PC-valvomon taajuusmuuttajan välilehdet
Liite 10	PC-valvomon servon välilehdet

1 Johdanto

Työn ensisijaisena tavoitteena oli toteuttaa Pohjois-Karjalan ammattikorkeakoulun laboratoriotiloissa olevalle kelaajaprosessille käyttöliittymä ja sekvenssiohjelma. Käyttöliittymän toteutin kosketuspaneelin lisäksi PC-valvomolle. Tämän lisäksi tavoitteena oli toteuttaa ohjaus prosessissa oleville moottoreille. Tämänkin opinnäytetyön jälkeen prosessia olisi tarkoitus jatkokehittää. Seuraava suurempi jatkokehitys työ on enemmänkin säätötekniikkaan liittyvä, jotta prosessi saadaan toimimaan mahdollisimman prosessimaisesti. Tarkoituksena olisi, että prosessia voitaisiin tulevaisuudessa käyttää eri opetuskäytöissä.

Koulu on saanut kelaajaprosessin (kuva 1) opetustarkoitukseen Kontiolahden kunnassa ennen toimineelta Perlos Oyj -nimiseltä yritykseltä. Kyseinen prosessi on ollut osa suurempaa linjastokokonaisuutta, joka on tuottanut erilaisia kännykkätuotteita. Prosessin alkuperäinen tarkoitus oli kiinnittää tarra matkapuhelimeen. Tällä hetkellä prosessi on pysynyt alkuperään nähden samana vain mekaniikan osalta, ohjauspuolen muutokset on tehty palvelemaan paremmin automaatiotekniikan opetusta. Prosessi tarjoaa erittäin monipuoliset mahdollisuudet opettaa eri osa-alueita automaatiosta, koska se sisältää erilaisia väylätekniikoita, moottorin ohjausratkaisuja ja turvateknistä puolta. Tämän lisäksi se antaa oppilaalle erinomaisen kuvan siitä, minkälaisilla ratkaisuilla automaatiolinjasto voidaan toteuttaa.

Aikaisempi opinnäytetyön tekijä Tuomo Lavikainen (2011) oli onnistuneesti rakentanut prosessin liitännät ja tehnyt ohjauspuolen muutokset, sekä myös toteuttanut kummallekin servo-ohjaimelle hajautetun IO-yksikön, joka on mielestäni hyvä ratkaisu toimiessaan. Prosessissa on lukuinen määrä turvatekniikkaa; Lavikainen (2011) oli toteuttanut eri kytkennät turvalaitteille ja tehnyt Asimon-ohjelman. Työn laajuuden vuoksi turvapuolen liittäminen tähän opinnäytetyöhön jätettiin pois.

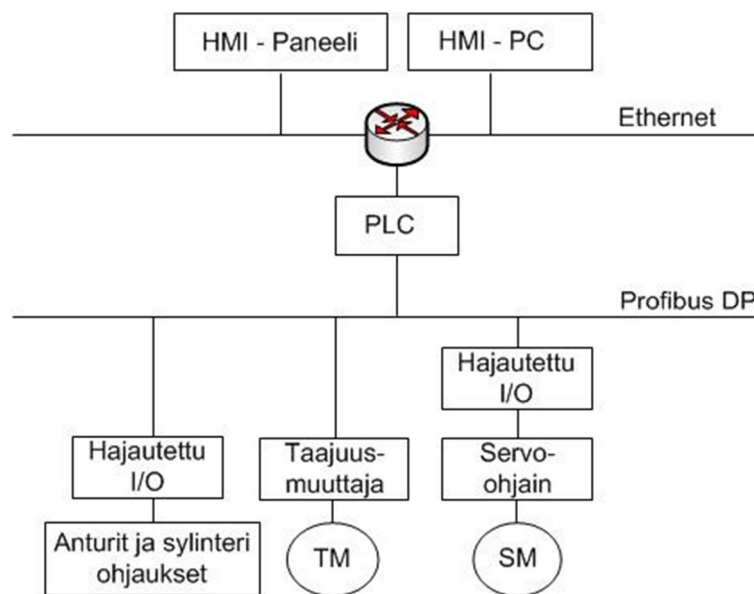


Kuva 1. Kelaajaprosessi.

2 Kelaajaprosessi

2.1 Järjestelmäkaavio ja laitteisto

Kuvasta 2 näkee järjestelmäkaavion, jossa TM osoittaa taajuusmuuttajamoottoria ja SM servomoottoria. Logiikka (PLC, programmable logic controller) ja käyttäliittymien (HMI, human machine interface) välinen kommunikointi tapahtuu Ethernet-yhteyttä käyttäen. Logiikan, taajuusmuuttajaohjaimen ja servo-ohjaimen välinen kommunikointi on toteutettu Profibusia käyttäen.



Kuva 2. Prosessin järjestelmäkaavio.

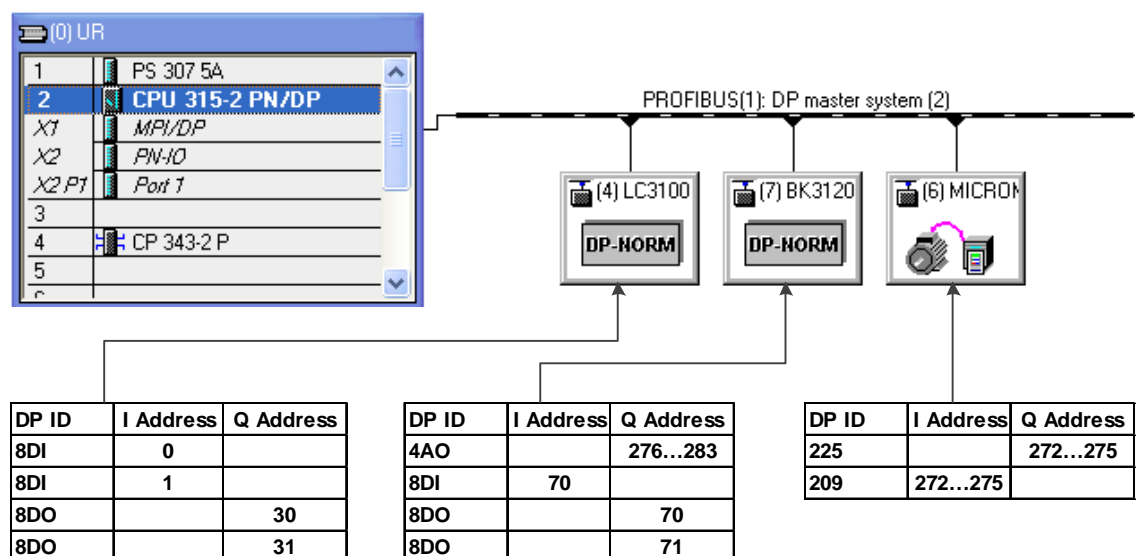
Prosessi sisältää suuren määrän erilaisia laitteistoja: sylintereitä, raja-antureita, erilaisia painelaitteistoja ynnä muuta. Laitteistot, johon itse jouduin perehtymään tarkemmin ovat erillään prosessista: logiikka, taajuusmuuttajaohjain ja moottori, servo-ohjain ja moottori sekä kosketusnäyttöpaneeli.

Tässä työssä on käytetty logiikkana Siemensin CPU315-2 PN/DP -logiikkaa. Tämä logiikka valittiin siksi, koska siinä oli Ethernet-yhteys mahdollisuus. Taajuusmuuttajana on Siemensin valmistama Micromaster 440-ohjain ja moottorina Minimotorin MC145-mallin moottori. Servo-ohjaimena toimii Omronin R88D-WT01H-ohjain, joka kommunikoi logiikan kanssa hajautetun IO-yksikön välityksellä. Hajautettu IO-yksikkö on rakennettu Bechhoffin valmistamista moduuleista. Servomoottorina toimii Yaskawan SGMAH-mallinen moottori. Kosketusnäyttöpaneelina on Siemensin valmistama TP177B-paneeli. Järjestelmäkaaviosta puuttuu toinen servo-ohjain ja – moottori, mutta sen toimimattomuuden vuoksi rajattiin se tämän työn ulkopuolelle jo opinnäytetyön aloitusvaiheessa. Selkeyden vuoksi turvapuoli on jätetty järjestelmäkaaviosta pois ja se on kuvattu tarkemmin luvussa 2.3.2.

2.2 Laitteistorakenne

Prosessin hajautettu IO-moduuli sisältää seuraavanlaiset tila- ja ohjaukortit: neljä digitaalista tulokorttia (KL1114, neljä tuloa per kortti), joilla luetaan prosessin tilatietoja ja kolme digitaalista lähtökorttia (KL2134, neljä lähtöä per kortti), joiden tehtävänä on hoitaa ohjauspuoli. Servon tila- ja ohjaukortit ovat seuraavanlaiset: analoginen lähtökortti hoitaa nopeus- ja momenttiohjeen asetusarvojen syötöt servo-ohjaimelle (KL4404, kaksi lähtöä per kortti); kolme digitaalista tulokorttia (2 x KL1114 & KL2134), jotka hoitavat tilatietojen lukemisen ja neljä digitaalista lähtökorttia hoitaa ohjauspuolen (KL2622, kaksi lähtöä per kortti). Taajuusmuuttajan tila- ja ohjaukortit on sisäänrakennettu.

Logiikan konfigurointi tapahtuu Step7-ohjelman avulla ja ohjelman konfigurointi ikkunan kautta. Kuvasta 3 näkee toteutetun hardware-konfiguraatio. Hajautetut IO-moduulit täytyi tuoda ohjelmaan GSD-tiedostona, jonka saa ladattua valmistajan kotisivuilta. IO-korteista LC3100 sisältää prosessin hajautetut kortit ja servon hajautettu IO-korttina on BK3120, joka mahdollistaa analogiohjaukset. Step 7:ssa hajautettuihin IO-moduuleihin täytyy määrittää kortit. Taajuusmuuttajaan ei tarvinnut määrittää kortteja, koska ne on integroitu moduuliin.



Kuva 3. Hardware-konfiguraatio.

2.3 Prosessin väylät

2.3.1 Profibus DP

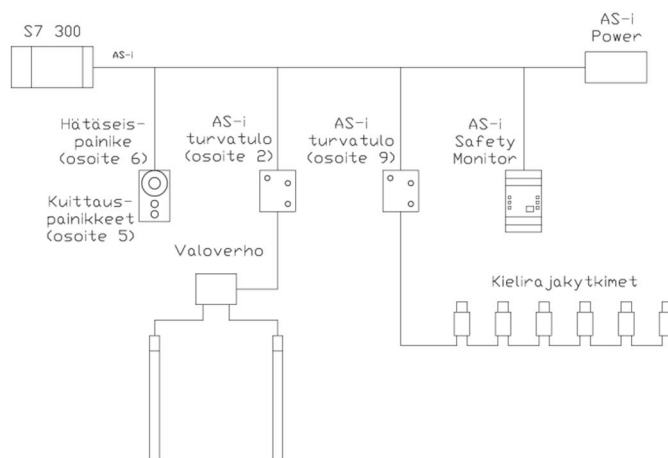
Profibus on avoin kenttäväylästandardi, joka on toimittajasta riippumaton. Profibus on hyvin yleisesti käytössä teollisuusautomaatiossa. Profibus-standardi EN 50170 takaa avoimuuden ja riippumattomuuden. Profibus DP on yksi kolmesta Profibus-tuoteperheen tyypistä, muut ovat Profibus PA ja Profibus FMS. Profibus DP on tehty nopeaa tiedonsiirtoa varten, ja sen erityisenä tarkoituksena on kommunikoida automaatiojärjestelmän ja hajautetun laitetaso välillä. Rinnakkaiskaapelointi, jossa on käytössä jännite- tai virtaviestejä, voidaan korvata Profibus DP – väylällä (Vacon 2006). Profibus DP-väylässä on yksi isäntä, joka etsii väylässä olevia orjalaitteita loogisessa renkaassa. Orjien lukumäärä voi olla maksimissaan 122, mutta yhdessä segmentissä korkeintaan 32. Liikennöinti nopeus riippuu valitusta nopeudesta, mutta standardin mukainen maksiminopeus on 12 Mb/s (Niiranen 1999, 280).

Prosessin IO-yksikkö, taajuusmuuttajaohjain, servon IO-yksikkö ja logiikka kommunikoivat keskenään Profibus-kenttäväylää käyttäen. Prosessin Profibus-korttina toimii Bechhoffin LC3100-kortti ja servon Profibus-korttina BK3120-kortti. Näiden osoite annetaan kääntämällä kiertokytkimestä. Taajuusmuuttajaohjaimen Profibus-osoite syötetään ohjelmallisesti (katso luku 3.3).

2.3.2 AS-interface

Turvapuolen kommunikointi on toteutettu AS-interface–väylää (AS-i) ja AS-i safety at Work -teknologiaa käyttäen. AS-i on avoin kenttäväylä, joka perustuu EN 50295-standardiin. AS-i isäntälaitte tekee yhteyden verkon ja ylemmän tason ohjausjärjestelmän välille. Se pystyy itsenäisesti suoriutumaan tiedonhallinnasta käyttäen AS-i -kaapelia. Orjien tehtävänä on kerätä tietoa ja antaa tämä tieto isäntälaitteelle. Isäntälaitte lähettää kyselyitä määritettyihin orjalaitteisiin ja kerää näin orjalaitteiden keräämät tiedot (Becker 2009, 33–35). Orjien lukumäärä voi olla maksimissaan 64, tulojen 256 ja lähtöjen 192. AS-i siirtonopeus on 167 kbit/s ja vasteaika 10 millisekuntia (AS-interface 2011)

Isäntälaitteena toimii logiikkaan liitetty AS-i-kortti, joka kyselee tietoja orjalaitteilta. Turvamonitorin (AS-i safety monitor) tehtävänä on tarkkailla väylässä kulkevaa dataa. Turvapuolen IO-laitteina toimivat: ovikytkimet, valoverho, hätäseis ja kuittauspainike. Nämä laitteet on liitetty AS-i moduuleihin, joiden kautta tieto annetaan turvamonitorille. Järjestelmäkaavio ja IO-laitteiden osoitteet näkyvät kuvassa 4. Osoitteiden määrittämisestä on tarkemmin Lavikaisen opinnäytetyössä (2011). Laitteistolle osoitteen määrittäminen tapahtuu AS-i-ohjelmointi työkalun avulla.



Kuva 4. Turvalaitteiden järjestelmäkaavio (Lavikainen 2011, Liite 3, 4).

2.3.3 Ethernet

PC, logiikka ja paneeli (TP177B) kommunikoivat toistensa kanssa Ethernet-väylää käyttäen. Tässä työssä reitittimenä käytin A-linkin RR-24 ADSL-reitintä. Osoitteet laitoin alkamaan osoitteesta 192.168.0.1 lähtien. Laitteiden osoitteet ovat taulukko 1:n mukaiset.

Taulukko 1. IP-osoitetaulukko

Laite	IP-osoite	Aliverkon peite	Yhdyskäytävä
Reititin	192.168.0.1	255.255.255.0	192.168.0.1
PC	192.168.0.2	255.255.255.0	192.168.0.1
Logiikka	192.168.0.3	255.255.255.0	192.168.0.1
Paneeli	192.168.0.4	255.255.255.0	192.168.0.1

A-linkin reitittimen konfigurointi onnistui helposti pikaohjeen mukaisesti (A-LINK 2007). Ensiksi laite resetoidaan laitteen takana olevasta kytkimestä, jotta laitteisto palautuu tehdasasetuksille. Seuraavaksi mennään selaimella osoitteeseen 10.0.0.2. Käyttäjänimi on oletuksena admin ja salasana password. Tämän jälkeen valikoissa edetään seuraavanlaisesti: Setup-valikosta edetään LAN Configuration-valikkoon, josta valitaan Configure. Täällä määritetään reititin käyttämään kiinteää IP-osoitetta, joka on siis tässä tapauksessa 192.168.0.1. Samasta valikosta määritetään mistä osoitteesta alkaen IP-osoite määritetään. Lopuksi tallennetaan tehdyt muutokset menemällä Tools-valikon kautta System Commands- valikkoon ja valitaan Save all.

Logiikan Ethernet-osoitteen määrittäminen tapahtuu hardware-konfiguraatiosta valitsemalla PN/IO:n properties-valikon auki. Valikossa määritetään IP- ja aliverkon peite, tämän lisäksi määritettiin logiikka käyttämään reititintä ja syötetään tämän osoite. Paneelin ja PC:n osoitteet määritetään omista ohjauspaneelin verkkoasetuksista.

3 Taajuusmuuttajan ohjaus

3.1 Taajuusmuuttaja

Taajuusmuuttajia on kahta tyyppiä: suoria ja välipiirillisiä. Suoran tyyppisissä taajuusmuuttajissa vaihtosähköverkon sähkö jaetaan puolijohdekytkimillä halutunlaisiksi jännitteiksi ja taajuiseksi vaihtosähköksi. Välipiirillinen toimintaperiaate on, että ensiksi sähkö muutetaan tasasähköksi ja sen jälkeen vaihtosähköksi (Niiranen, 1999, 48). Taajuusmuuttajan suurin etu on energian säästö, joka tulee siitä kun moottoria käytetään tarpeen mukaisella nopeudella, näin hukka on mahdollisimman pieni (Rossiter 2011).

3.2 Moottorin arvot

Ennen kuin pystyin parametroimaan taajuusmuuttajaa, täytyi tietää moottorin arvot. Työssä käytettiin Minimotor valmistajan moottoria ja sen arvot löytyivät

moottorin arvokilvestä ja moottorin manuaalista (Minimotor 1999), jotka olivat seuraavanlaiset:

- taajuus: 50 Hz
- virta: 0,32 A
- jännite: 230 V
- moottorin pyörimisnopeus: 1400 r/min
- moottorin teho: 18 W.

Kuitenkin $\cos\varphi$:tä ei löytynyt arvokilvestä eikä manuaalista, joten laskin moottorin tiedetyillä arvoilla ja tehokaavalla suurpiirteisen $\cos\varphi$:n (kaava 1).

$$\cos\varphi = \frac{P}{UI} = \frac{18W}{230V \cdot 0,32A} \approx 0,25 \quad (1)$$

Jossa	P	=teho
	U	=jännite
	I	=virta.

3.3 Taajuusmuuttajaohjaimen parametointi

Moottorin arvojen selvittyä, voitiin lähteä parametroimaan taajuusmuuttajaa. Taajuusmuuttaja ohjaimena työssä käytettiin Siemensin Micromaster 440 ohjainta. Parametointi onnistuu kyseiselle ohjaimelle suhteellisen vaivattomasti käyttäen valmistajan tekemää manuaalia (Siemens 2001). Parametroinnin voi tehdä joko ohjelmallisesti Starter-ohjelmalla tai vaihtoehtoisesti BOP-paneelilla, valitsin jälkimmäisen vaihtoehdon. Ohjaimessa on satoja parametrejä, mutta ohjaimesta löytyy myös pikakäyttöönotto, joka tässä tapauksessa riitti hyvin. Aluksi palautin ohjaimen tehdasasetuksille, tämä onnistuu asettamalla P0010-parametri arvoon 30 ja tämän jälkeen parametri P0970 ykköseksi. Seuraavaksi tein pikakäyttöön otton asettamalla parametrin P0010 ykköseksi ja syötin ohjaimelle seuraavanlaiset arvot:

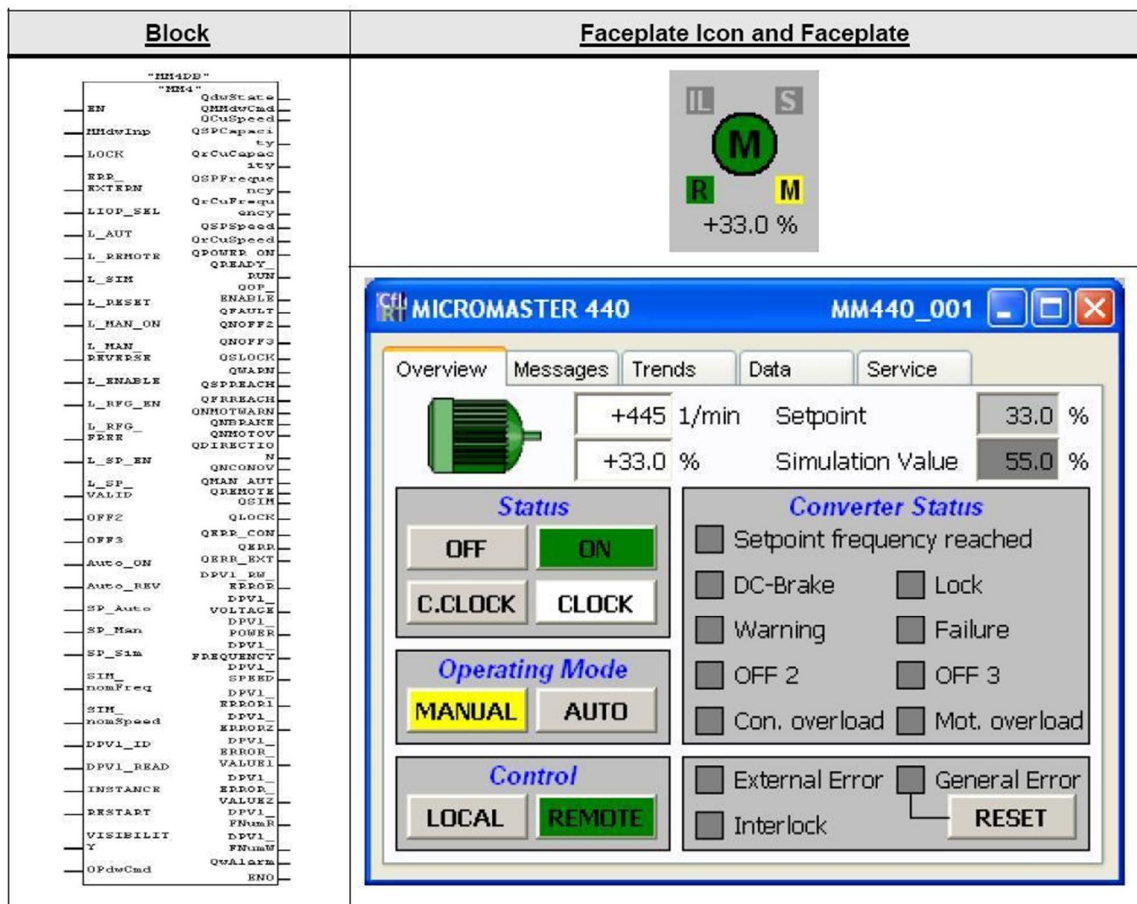
- P0304 = 230 (V)
- P0305 = 0.32 (A)
- P0307 = 0.018 (kW)

- P0308 = 0.25 (cos phi)
- P0310 = 50 (Hz)
- P0311 = 1400 (r/min)
- P0700 = 2 (BOP-paneelin käyttöönotto)
- P1080 = 0 (minimi taajuus Hz)
- P1082 = 50 (maksimi taajuus Hz)
- P3900 = 1 (syötettyjen parametrien tallennus).

Pikakäyttöönotossa ei syötetty profibus-osoitetta, joten se täytyi syöttää tämän jälkeen parametrissa P0918, osoitteeksi määritettiin kuusi. Tämän lisäksi profibus-väylälle täytyy määrittää baund-nopeus valikosta P2010.

3.4 Taajuusmuuttajalohko

Siemens on toteuttanut taajuusmuuttajalle esimerkkilohkon (kuva 5), josta näkyy miten taajuusmuuttaja ratkaisu voidaan toteuttaa. Lohkon lisäksi Siemensin projekti sisältää käyttöliittymän osalta tilaikkunan ja piirinäytön. Lohkon ja käyttöliittymän tarkoituksena on, että kopioituvuus nopeuttaisi työskentelyä. Käyttäessä samaa käyttöliittymäobjektia (jatkossa faceplate) useammassa paikassa, tarvitsee tehdä muutoksia vain yhteen faceplaten ja nämä muutokset tulevat voimaan kaikissa faceplatessa esim. jos käytössä on useita samanlaisia taajuusmuuttajia ja halutaan näiden samankaltaisuus taata, nopeuttaa faceplate-ominaisuus huomattavasti työskentelyä.



Kuva 5. Taajuusmuuttajalohko ja sen käyttöliittymäobjektit (Siemens 2010a).

Lohko tarjoaa seuraavat ohjaus ominaisuudet: käynnistyksen/pysäytyksen, asetusarvojen syötöt, suunnanvaihdon, ohjaustavan valinnan (manual/auto) ja ohjauspaikan valinnan (local/remote). Manual -ohjauksella asetusarvo syötetään manuaalisesti ja auto-ohjauksella asetusarvo syötetään ulkoisesta lähteestä. Paikallisesti ohjaus (local) tapahtuu laitteen lähellä olevista kytkimistä. Etäohjaus (remote) tapahtuu ulkoisesta käyttöliittymästä. Tarkemmin piirinäytön välilehtien ominaisuuksista on luvussa 6.5.

Lohkon toimintaan perehtyminen oli järjestyksessään tärkeimpiä perehtymisiä. Lohko on erittäin laaja ja se käsittää useita riviä ohjelmallista koodia. Lohkon toiminta on toteutettu siten, että lohkoa voidaan ohjata joko paikallisella signaaleilla tai ohjaussanalla. Ohjaussana on ensisijaisesti tarkoitettu käyttöliittymän ohjaukseen, joten se on tässä työssä tärkeämpi. Ohjaussanan ja tilasanan bittien merkitys näkyy kuvissa 6-7.

OPdhwCmd		Bitti
Signaali		31
		30
		29
		28
		27
		26
		25
	RESET	24
		23
		22
	SIMULATION	21
	PROCESS	20
	REMOTE	19
	LOCAL	18
	AUTO	17
	MANUAL	16
		15
		14
		13
		12
		11
		10
		9
	DPV1_REQ	8
		7
		6
		5
		4
	REVERSE	3
	FORWARD	2
	START	1
	STOP	0

Kuva 6. Taajuusmuuttaja lohkon ohjaussanan bittien merkitys.

QdwsState		Bitti
Signaali		31
	DPV1_RW_ERROR	30
		29
		28
	LOCK	27
	QLOCK	26
	QERR_EXT	25
	QERR	24
		23
		22
	QFAULT	21
	QWARN	20
		19
	QSIM	18
	QREMOTE	17
	QMAN_AUT	16
	QONCONOV	15
	QNMOTOV	14
	QNBRAKE	13
	QNMOTWARN	12
	QNOFF3	11
	QNOFF2	10
	QFRREACH	9
	QSRREACH	8
	OP_REMOTE	7
	OP_REVERSE	6
	OP_ON	5
	QREADY_RUN	4
	QSLOCK	3
	QDIRECTION	2
	QOP_ENABLE	1
	QPOWER_ON	0

Kuva 7. Taajuusmuuttaja lohkon tilasana bittien merkitys.

Huomioitavaa on se, että ohjaussana ei toimi jos ohjaustavaksi on valittu paikallinen ohjaus (Liop_sel = 1). Tämän lisäksi bitit 0-3 toimivat vain kun ohjaus on manuaali ohjauksella. Ohjaussanaa ja auto-ohjausta käyttäessä täytyy käyttää erillistä käynnistysbittiä, jotta moottorin saa päälle ja pysäytettyä. Tarkemmin lohkontoiminnasta voi lukea Siemensin tekemästä laajasta manuaalista (Siemens 2010a).

Liitteestä 1 näkee Step7-ohjelmassa olevan lohkon kokonaisuudessaan. Lohkoon täytyy tuoda laiteosoitteet taajuusmuuttajaohjaimen tuloon MMdwlInp ja lähtöön QMMdhwCmd. Tämän lisäksi DPV1_ID-kenttään täytyi tuoda taajuusmuuttajan diagnostiikka-osoite heksamuodossa, tiedot saadaan luettua hardware-konfiguraatiosta. Tässä tapauksessa diagnostiikka-osoite oli 2044, joka on heksana 7FC.

4 Servon ohjaus

4.1 Servo

Yleistyttyään servot ovat tuoneet älyä mekaanisten komponenttien ohjaukseen. Servojen toiminnan tarkoitus on tehdä mekaaninen liike sitä ohjaavan järjestelmän antaman ohjeen mukaisesti. Servon etuna on se, että liike voi olla suoran liikkeen lisäksi kiertyvää. Takaisinkytkennät ovat pakollisia, jotta voidaan suorittaa ohjauksen asettamia ohjeita. Servon signaalit voidaan jakaa kolmeen ryhmään: ohjaus-, häiriö ja lähtösignaaleihin. Nämä ovat tekijöitä, jotka vaikuttavat servon toimintaan. (Fonselius, Rinkinen & Vilenius. 1998, 7,11)

Servo-ohjaimen parametointi onnistuu joko ohjaimesta itsestään tai vaihtoehtoisesti ohjelmallisesti. Servoa ei tarvinnut lähteä parametroidaan, vaan se oli valmiiksi parametroidu Lavikaisen (2011) toimesta. Tutustuin kuitenkin tärkeimpiin servon parametreihin SigmaWin+ -ohjelmalla ja tein listan tämän hetkisistä parametreista prosessin kansioon.

4.2 Servolohkon toiminta

Taajuusmuuttajalohko toimii pohjana servolohkelle. Tämä ei kuitenkaan suoraan toimi servo-ohjaimella vaan ohjaajan oli tehnyt lohkon tarvittavat muutokset. Servon ohjauksena ja tilasana ovat kuvien 8 ja 9 mukaiset.

OPdrvCmd	Bitti
Signaali	31
	30
	29
	28
	27
	26
	25
	24
RESET	23
TORQUE	22
SPEED	21
SIMULATION	20
PROCESS	19
REMOTE	18
LOCAL	17
AUTO	16
MANUAL	15
	14
	13
	12
	11
	10
	9
	8
	7
	6
	5
	4
	3
	2
START	1
STOP	0

Kuva 8. Servolohkon ohjauksena bittien merkitys.

QdrvState	Bitti
	31
	30
	29
	28
LOCK	27
QLOCK	26
QERR_EXT	25
QERR	24
	23
	22
QFAULT	21
	20
	19
OSIM	18
OREMOTE	17
QMAN_AUT	16
	15
	14
	13
	12
	11
	10
	9
QSPREACH	8
OP_REMOTE	7
OP_REVERSE	6
S_ON	5
QREADY	4
QSLACK	3
QDIRECTION	2
QOP_ENABLE	1
QPOWER_ON	0

Kuva 9. Servolohkon tilassanan bittien merkitys.

Suurimpana muutoksena taajuusmuuttajan lohkon verrattuna tilassanassa on se, että servolohkosta ei tule yhtä paljon hälytystietoja mitä taajuusmuuttajasta. Ohjauksenaakin on melko samanlainen mitä taajuusmuuttajan ohjauksena, poikkeavuutena on kuitenkin se, että servolohkossa ei ole suunnanvaihtoa, eikä DPV1-lukua. Lisäyksenä ohjauksenaan on lisätty nopeus- ja momenttiohjeen vaihto.

Liitteestä 2 näkee Step7-ohjelmassa olevan servolohkon kokonaisuudessaan. Servolohkon tuloihin ja lähtöihin täytyy tuoda seuraavat laiteosoitteet: tuloliitäntä (ServoSB), moottori päällä (S_ON), ohjauksenaan valinta (C_SEL), hälytyksen kuittaus (ALM_RST), nopeusohje (QCuSpeed) ja momenttiohje (QCuTorque).

5 Sekvenssiohjaus

5.1 Oletetut käyttäjät

Ennen suunnittelun aloittamista täytyi määrittää oletetut käyttäjät. Täytyi miettiä, mitkä voisivat olla mahdolliset käyttäjäryhmät prosessille, jotta tämä helpottaisi määrittämään tietynlaiset käyttöliittymät eri ryhmille. Tulon siihen johtopäätökseen, että riittää kun käyttäjäryhmiä on kaksi: ajajat ja huolto. Eniten prosessia käyttävät ajajat, joten tämä pitää määrittää pääkäyttäjärühmäksi. Ajajat-ryhmälle täytyisi kuitenkin näkyä vain prosessin ajoon liittyvät ohjaukset. Huoltoryhmän pitää päästä kaikkiin mahdollisiin ohjauksiin.

5.2 Sekvenssin suunnittelu

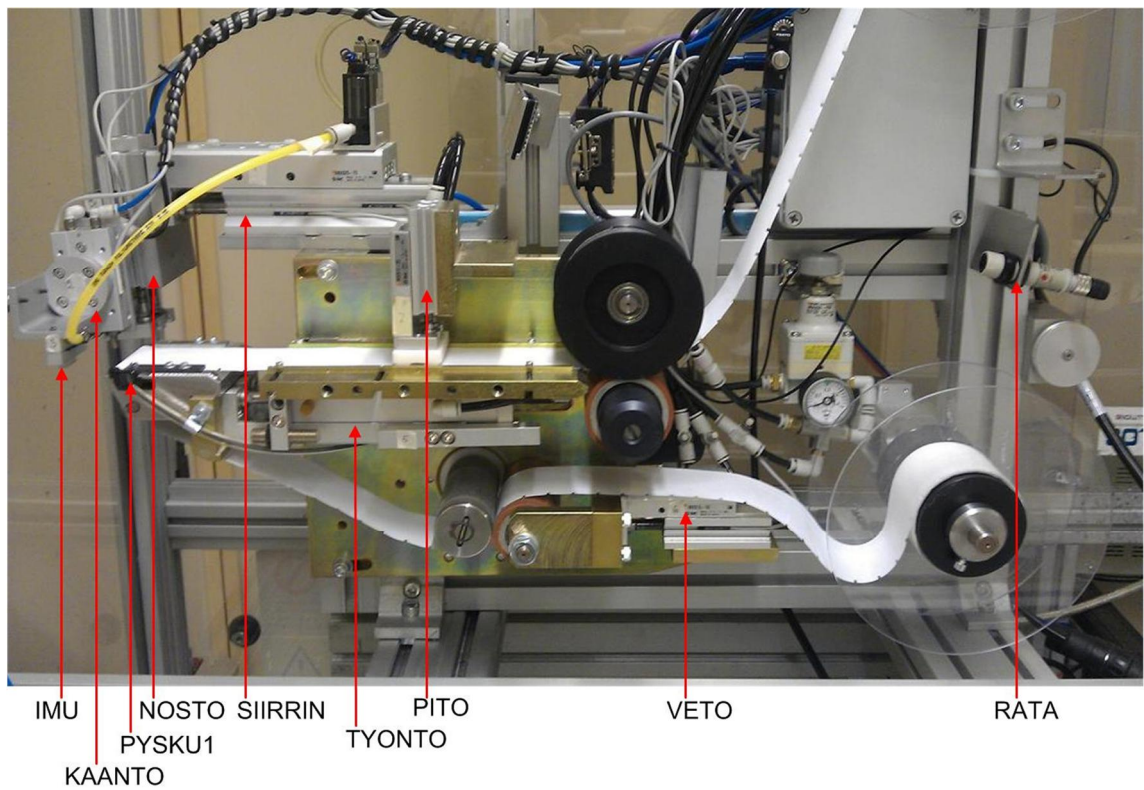
Sekvenssin suunnittelun pohjana toimi toimeksiantajan antama pääpiirteinen kirjallinen toimintasekvenssi. Tässä oli selvitetty lähinnä järjestys siitä, missä järjestyksessä sylinterit liikkuvat. Siinä ei ollut huomioitu pysäytyskuitua eikä myöskään moottorin toimintaa. Kun olin perehtynyt prosessiin ja moottoreiden toimintaan, lähdin miettimään koko kokonaisuutta.

Lähtökohtaisesti määrittelimme toimeksiantajan kanssa, että perustilassa moottorit ovat auto-ohjauksella. Tämä siitä syystä, että prosessin oletettujen ajajien (luku 6.1) ei tarvitse asettaa moottoreiden nopeutta vaan ne ovat ennalta asetetut. Täytyy ottaa myös huomioon se, että manuaali ohjaukseen siirtyessä sekvenssi ei asettaisi sitä välittömästi takaisin auto-ohjaukselle.

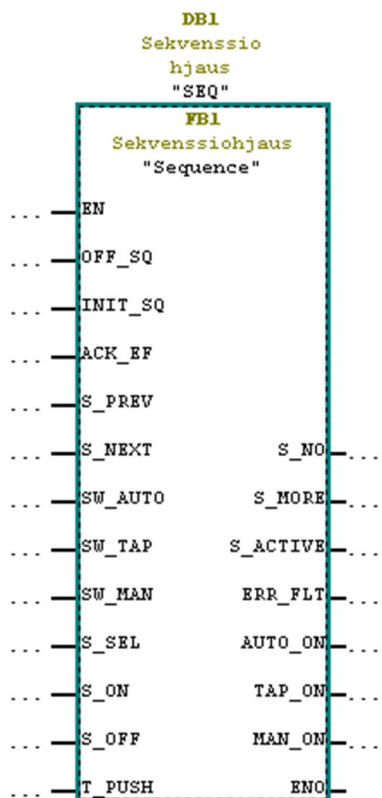
Moottoreiden lohossa on kaksi tilaa, joiden täytyy olla päällä, jotta moottori on auto-ohjauksella. Tämä helpottaa sekvenssin toteutusta; näin sekvenssin voi asettaa toisen bitin ja käyttöliittymästä voidaan asettaa toinen bitti. Niinpä kun siirrytään auto-ohjaukselta manuaali ohjaukseen, sekvenssi ei sotke tätä, koska toinen bitti voi olla päällä ilman, että se yksin asettaisi auto-ohjausta päälle. Pysäytyskuitujen tehtävä on se, että sen tilanvaihdot vaikuttavat moottoreiden pysäytyksiin ja käynnistykseen. Näiden tietojen valossa täydensin annettua toimintasekvenssiä (liite 3.)

5.3 Sekvenssin toteutus

Sekvenssin toteutusta varten täytyi listata suunnittelussa esitettyjen ohjaus- ja tilabitien osoitteet, jotka ovat liitteen 4. mukaiset. Ohjausbittien osoitteet on nimetty niin, että niiden tunnistettavuus helpottuisi. Kuvaan 10 on merkitty mitä milläkin nimellä tarkoitetaan. Tilabitit on nimetty samalla tavalla, mutta perään on laitettu kirjain osoittamaan, missä asennossa sylinteri on.



Kuva 10. Bittien merkitys.



Kuva 11. Sekvenssin lohko.

Sekvenssin toteutus oli nopeaa ja helppoa hyvän pohjasuunnittelun ansiosta. Kuvasta 11. näkee Step7:ssa olevan sekvenssin lohkon ja liitteestä 5. näkee Step7:ssa olevan sekvenssi-toteutuksen kokonaisuudessaan. Sekvenssissä on kolme valinnaista ohjausta: auto, manual ja tap. Auto-ohjauksessa sekvenssi käy automaattisesti askeleita järjestyksessä, tässä tapauksessa ykkösestä kolmeentoista. Manuaali ohjauksella voi syöttää halutun askeleen ja aktivoi-da/deaktivoida sen. Tap-ohjauksella edetään painalluksella, askel kerrallaan. Sekvenssissä ohjaus- ja tilabittien merkitys on seuraavanlainen:

- OFF_SQ: sekvenssin pysäytys.
- INIT_SQ: sekvenssin käynnistys.
- ACK_EF: virheen kuittaus.
- S_PREV: edelliseen askeleeseen siirto (manual/auto).
- S_NEXT: seuraavaan askeleeseen siirto (manual/auto).
- SW_AUTO: auto-ohjaus päälle.
- SW_TAP: tap-ohjaus päälle.
- SW_MAN: manual-ohjaus päälle.
- S_SEL: askeleen syöttäminen kokonaislukuna (manual).
- S_ON: syötetyn askeleen aktivointi (manual).
- S_OFF: syötetyn askeleen deaktivointi (manual).
- T_PUSH: askel tap-ohjauksessa.
- S_NO: tämän hetkinen syötetty askel.
- S_ACTIVE: askel aktiivinen.
- ERR_FLT: sekvenssi virhetilassa.
- AUTO_ON: auto-ohjaus päällä.
- TAP_ON: tap-ohjaus päällä.
- MAN_ON: manuaali ohjaus päällä.

5.4 Testauksen jälkeiset muutokset

Sekvenssin testauksessa tuli muutamia muutoksia suunnitelmaan. Ensimmäinen muutos oli se, että sama pystykuitu piti huomioida kahdessa kohtaa, koska

toinen pystykuitu ei ollut toiminnassa; askel kahdessa (nolla) ja askel kolmessa (yksi). Toinen muutos oli se, että taajuusmuuttajamoottoria ei tarvitse pysäyttää, vaan se voi olla koko ajan päällä. Tähän on syynä se, että taajuusmuuttajamoottorin käynnistys on prosessin kannalta hidasta.

6 Käyttöliittymän teoria

6.1 Lähtökohta

Alkuasetelmat käyttöliittymän suunnittelussa ovat sellaiset, että käyttöliittymä toteutetaan kahdelle laitteistolle, ensisijaisesti Siemensin Touch TP177B-paneelille, mutta myös PC:lle. Tämä vaatii tiettyjä muutoksia, mutta kokonaisuudessa nämä pyritään pitämään samanlaisina. Käyttöliittymäohjelmaksi määritettiin jo toimeksiannossa WinCC flexible. Tämä siitä syystä, että se integroituu logiikan ja näyttöpaneelin kanssa hyvin yhteen, koska sekä logiikka, että paneeli ovat Siemensin valmistamia.

Siemens tarjoaa Micromasterille (taajuusmuuttajaohjain) valmiin käyttöliittymäikkunan, jota muokkaamalla saadaan taajuusmuuttaja- ja servo-ohjaimelle hyvin samankaltaiset ikkunat. Poikkeuksena tässä on kuitenkin huomioitava se, että servo-ohjaimelta ei saada aivan yhtä kattavasti tietoa mitä taajuusmuuttajalta. Tämän lisäksi huomioitava on se, että paneeli ei tue Siemensin toteuttamaa faceplate-käyttöliittymää.

6.2 Hahmolait

Käyttöliittymän suunnittelussa yhtenä tärkeänä tekijänä pidetään hahmolakien merkitystä. Yksittäisistä ärsykkeistä muodostuu ihmisen havaintojärjestelmään yksi iso kokonaisuus. Kohteen tuttuus, sekä ihmisen odotukset määrittävät sen, miten nopeasti ihminen pystyy luokittelemaan ja tunnistamaan kohteen. Oikeanlaisella ryhmittelyllä saadaan luotua laajempi kokonaisuus (Sinkkonen, Kuoppala, Parkkinen, Vastamäki 2002, 102).

Läheisyys ja samanlaisuus

Lähekkäiset ja samanlaiset kohteet mielletään yhteenkuuluviksi ja täten ne nähdään myös ryhmänä (Sinkkonen ym. 2002, 102). Läheisyyden huomiointi käyttöliittymä toteutuksessa toteutuu siten, että ohjauspainikkeet sijaitsevat saman etäisyyden päässä toisistaan. Samanlaisuus toteutuu siinä, että ohjauspainikkeet ja tilavalot ovat samankokoisia ja värisiä.

Valiomuotoisuus

Valiomuotoisuudella tarkoitetaan sitä, että kohteet mielletään mahdollisimman yksinkertaisina ja valiomuotoisina (Sinkkonen ym. 2002, 102). Käyttöliittymässä valiomuotoisuus näkyy siinä, että painikkeet noudattavat normaalia neliskulmaista yleisen painikkeen kaavaa ja tilavalot ympyrää/neliötä.

Yhteenliittyminen

Yhteenliittymisellä tarkoitetaan sitä, että kohteet jotka ovat kiinni toisissaan, pidetään samaan ryhmään kuuluvina (Sinkkonen ym. 2002, 104). Käyttöliittymässä yhteenliittyminen näkyy siinä, että tilavalot ovat painikkeiden vieressä.

Sulkeutuvuus

Sulkeutuvuudessa ihminen näkee jonkin tilan eroteltuna alueena. Alueen rajat ovat ärsykeitä, jotka muodostavat alueesta yhteenkuuluvan. (Sinkkonen ym. 2002, 104). Käyttöliittymissä sulkeutuvuus näkyy niin, että tilatiedot on eroteltu ohjauksista rajalla. Tämän lisäksi rajalla on eroteltu eri ohjaukset.

6.3 Selkeys ja hahmotettavuus

Selkeyteen kannattaa panostaa mahdollisimman kattavasti. Käyttöliittymä voi sisältää paljonkin informaatiota ilman, että se menettää selkeyttään. Selkeys määrittyy siitä, miten hyvin ihmisen aivot hahmottavat katsomansa kohteen. Jäsentämisessä on periaatteita, joita hyvän käyttöliittymän täytyisi noudattaa. Länsimäisellä ihmisellä katse kulkee vasemmasta ylänurkasta oikeaan alanurkaan. Tämän johdosta tärkein informaatio on hyvä sijoittaa vasemmalle ylös ja

vähemmän tärkeämpi oikealle alas. (Suomen automaatioseura ry, 2010, 103-104)

Toteutuvassa käyttöliittymässä sijoittelu tulee olemaan niin, että start- ja stop-painikkeet sijoittuvat vasemmalle mahdollisimman ylös, koska ne ovat ehdottomasti tärkeimmät ohjaukset. Tilatiedot sen sijaan sijoitetaan oikealle puolelle käyttöliittymää, koska ne ovat vähemmän tärkeitä tekijöitä.

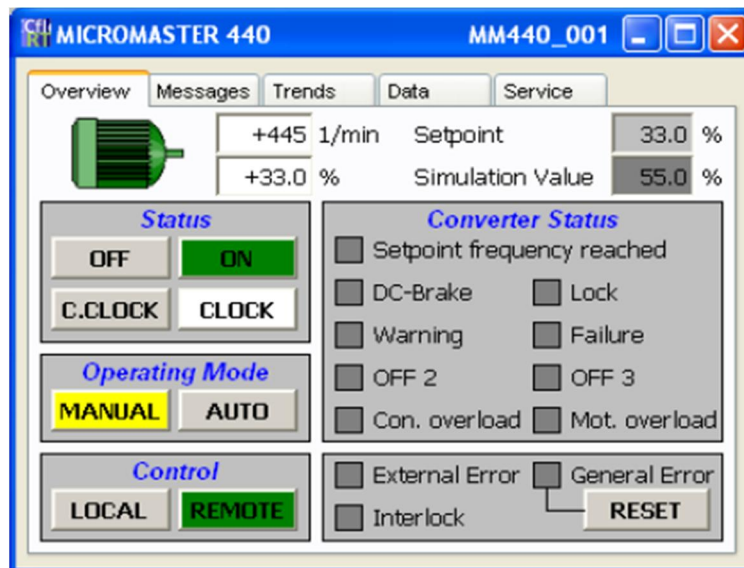
6.4 Luettavuus

Selkeää käyttöliittymää tavoiteltaessa täytyy sen olla luettavuudeltaan hyvä. Luettavuuteen vaikuttavia tekijöitä ovat tekstin koko, muoto, värit ja kontrasti taustaan katsottuna. Tekstin kokoa valittaessa on huomioitava käyttäjän katseluetäisyys ja mietittävä onko se koko ajan sama vai saattaako katseluetäisyys muuttua. Fontiksi on hyvä valita yleisesti käytössä olevat, esim. Arial taiTahoma. Värien käyttö täytyy olla hillittyä ja on suosittava vaaleita värejä. ISO 9241-12 mukaan värejä ei saisi olla mustan ja valkoisen lisäksi kun maksimissaan kuusi. Tämä yksinkertaiseksi siksi, että ihminen ei pysty muistamaan tätä enempää värejä. (Suomen automaatioseura ry, 2010, 106-107)

Luettavuus on huomioitu värien osalta siten, että käyttöliittymissä on käytössä värejä yhteensä neljä: punainen, vihreä, harmaa ja keltainen (pois luettuna musta ja valkoinen). Fontiksi valitsen Tahoman, koska näyttöpaneeli (TP 177B) ei tue muita yleisiä fontteja, samankaltaisuuden vuoksi valitsin PC-valvomon fontiksi myös Tahoman. Tekstin koon valitsen katseluetäisyyden mukaan. Näyttöpaneelissa (TP 177B) se tulee olemaan katseluetäisyyden ja tilan takia PC-valvomoa pienempi.

6.5 Siemensin tekemän piirinäytön tarkastelu

Siemens tarjoaa Micromaster-ohjaimelle valmiin ohjausblokin lisäksi valmiin käyttöliittymän, joten toimeksiantaja katsoi, että tätä olisi hyvä käyttää taajuusmuuttajan ohjaamiseen. Tätä pystyi myös hyödyntämään servon käyttöliittymässä, tehden vain tiettyjä muutoksia. Tarkastelin kyseisen käyttöliittymän käytettävyyttä, ennen kuin lähdin miettimään tarvittavien muutoksien tekemistä.



Kuva 12. Taajuusmuuttajan ohjaisikkuna (Siemens 2010a).

Kyseisessä käyttöliittymässä on kuvan 12 mukaisesti viisi välilehteä. Ensimmäisellä välilehdellä (Overview) on moottorin ohjaus ja tilatiedot. Messages-välilehdessä on yleistä tietoa moottorin tilasta eli tähän välilehteen tulostuu esim. tieto jos ohjaustapaa on vaihdettu. Trends-välilehden alle piirtyy trendikäyrä asetetuista arvoista, tässä tapauksessa asetusravosta (setpoint) ja todellisesta prosentuaalisesta nopeudesta. Data-välilehteen tulee tiedot ohjaimen syötetyistä moottorin arvoista, nämä tiedot käyttöliittymä lukee DPV1-kentistä. Service-välilehdestä voi ohjaimen asettaa simulointi-ajoon.

Käyttöliittymän suunnittelija on ottanut hyvin huomioon lähes kaikki käytettävyyteen liittyvät asiat. Käyttöliittymä sisältää paljon informaatiota, mutta siitä huolimatta se ei sotke selkeyttä. Samankaltaisuuteen on päästy hyvin pitämällä painikkeet ja teksti samankokoisina. Sulkeutuvuus on toteutettu hyvin, rajaamalla eri ohjaukset ja tilatiedot toisistaan.

Ainoana ongelmana näen värien käytön. Ensinnäkin värit on sidottu painikkeisiin, mikä johtuu todennäköisesti siitä, että näin yritetään säästää tilaa, jotta se sopii myös paneelikäyttöön. Tämä mielestäni häiritsee jossain määrin hahmotettavuutta. Tämän lisäksi eri tiloille on annettu neljää eri väriä. Käyntitilalle (on) ja etäohjaukselle (remote) on annettu vihreä, manuaali ajotilalle keltainen sekä pyörimissuunnalle (c.clock/clock) ja pysäytystilalle (off) valkoinen ja hälytystiloil-

le punainen. Mielestäni hahmotettavuuden kannalta kannattaisi käyttää vain kolmea eri väriä: vihreää, keltaista ja punaista. Vaikka määrällisesti nykyinen ei riko sitä ohjenuoraa, että kuutta väriä enempää ei saa käyttää. Kuitenkin mielestäni pyörimissuuntaa ja pysäytystilaa ei tarvitse ilmoittaa eri väreillä, vaan siinäkin voisi käyttää keltaista. Viimeinen huomio värien käytössä on etäohjauksen väritys, joka on vihreä. Mielestäni tämän voisi olla keltainen, koska vihreä osoittaa enemmän käyntitilaa. Tällä tavalla saataisiin luotua paremmin hahmotettava käyttöliittymä, joten nämä muutokset aion toteuttaa käyttöliittymään muiden muutosten lisäksi.

Tämän lisäksi Siemens on toteuttanut pienen tilaikkunan, joka on kuvan 13 kaltainen. Tilaikkuna näyttää kuvan 14 mukaiset tiedot. Muutoksia täytyy tehdä vain servon tilaikkunaan. Siihen täytyy tuoda toinen nopeustieto ja merkitä ne erottavuuden takia lyhenteillä; V (Nopeus) ja T (momentti). Tämän lisäksi, koska servon simulointia ja interlock-ohjausta ei ole, niin ne voi poistaa servon pienestä tilaikkunasta.



Kuva 13. Taajuusmuuttajan tilaikkuna (Siemens 2010a).

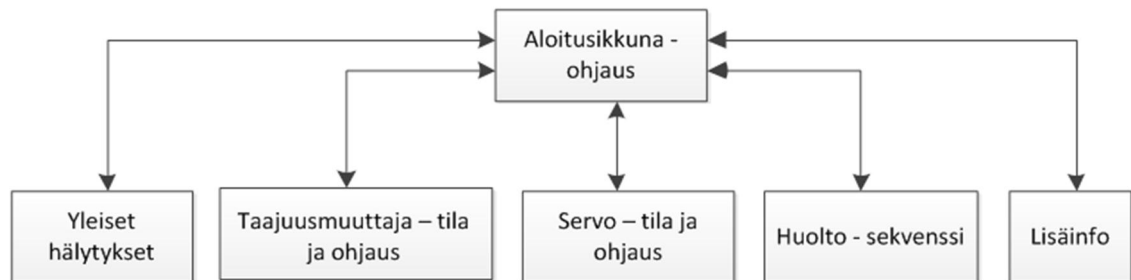
State	Symbol
Local	L
Remote	R
Manual	M
Automatic	A
Simulation ON	S
Simulation OFF	S
Interlock OFF	IL
Interlock ON	IL
Interlock triggered	IL

Kuva 14. Taajuusmuuttaja tilaikkunan tilojen merkitykset (Siemens 2010a).

7 Käyttöliittymien suunnittelu

7.1 Paneelin käyttöliittymäikkunat

Aluksi täytyi paneelin hahmotelmassa ottaa huomioon eri käyttäjäryhmät, mitkä on lueteltu luvussa 5.1. Tämän jälkeen hahmottelin pääpiirteittäin eri ikkunaryhmät, joista tuli kuvan 15 kaltaiset. Ikkunaryhmät tulevat eri välilehtiin ja käyttäjäryhmiä ajatellen täytyy kolmeen välilehteen olla pääsy vain huoltoryhmälle. Nämä välilehdet ovat: taajuusmuuttaja-, servo- ja huolto-ikkuna. Välilehdet ja yläpalkki löytyvät WinCC Flexible-ohjelmasta valmiina, joten otin nämä mukaan jo suunnittelukuvia piirtäessä.



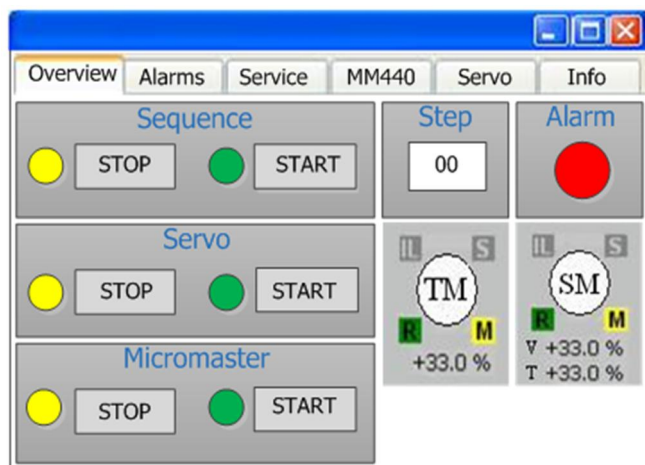
Kuva 15. Yleisnäkymä.

Ohjausikkuna

Ohjausikkuna (overview) on aloitusikkuna paneelissa, joka on räätälöity enemänkin ajaja-ryhmälle. Ikkuna sisältää kaikki oleelliset ohjaukset mitä prosessin ajaja tarvitsee. Tällöin taajuusmuuttaja- ja servomoottori ovat auto-ohjauksella. Auto-ohjauksella asetusrivot on ennalta asetettu sellaiseksi, että kelaaja pyörii mahdollisimman optimaalisesti. Aloitusikkuna sisältää seuraavat elementit:

- sekvenssin käynnistys- ja pysäytystoiminnot
- taajuusmuuttajamoottorin käynnistys- ja pysäytystoiminnot
- servomoottorin käynnistys- ja pysäytystoiminnot
- yleisen hälytystilavalon
- taajuusmuuttaja- ja servo moottoreiden tilan.

Elementtien hahmottelussa täytyi ottaa jo valmiiksi huomioon paneelin koko, joka on 113 mm x 85 mm. Hahmoteltaessa käyttöliittymää Microsoft Officen Visio-ohjelmalla tein rajauksen myös tähän ohjelmaan niin, että suhde oli 1:1, tämä helpotti hahmottamista. Tämän lisäksi täytyi ottaa jo hahmottamisvaiheessa huomioon käytettävyyteen liittyvät asiat, joista on kirjoitettu luvussa 6. Tarkemmin hahmoteltu ohjausikkuna näkyy kuvasta 16.



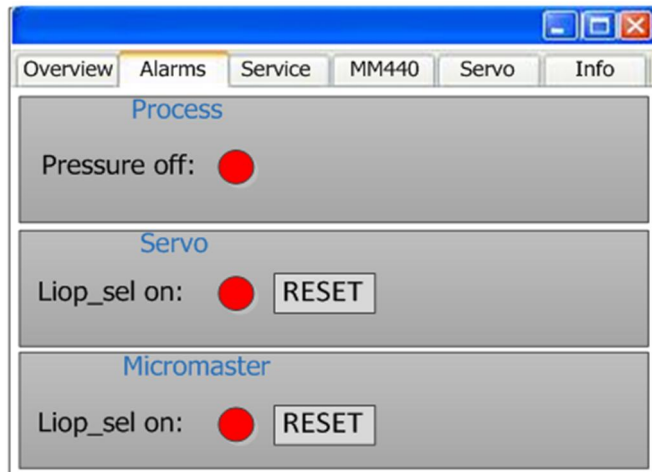
Kuva 16. Ohjausikkuna.

Tilatietojen värit tulevat olemaan valkoiset, kun kyseinen bitti on nolla ja kuvan mukaiset, kun se on ykkönen. Taajuusmuuttaja- ja servo-ohjaimen tilaikkunat ovat osittain Siemensin valmistamat, joten toin ne suoraan hahmotelmaan tehden vain muutokset keskiöön ja laittamalla servomoottorin tilaikkunaan kaksi nopeustietoa ja lyhenteen V (Nopeus) ja T (Momentti), nimeämällä moottorit TM:ksi (taajuusmuuttajamoottori) ja SM:ksi (servomoottori).

Hälytysikkuna

Hälytysikkunaan on tarkoitus tuoda vain yleiset hälytykset. Moottoreilla on omat hälytykset omissa ikkunoissaan, joten niitä ei tarvitse tähän ikkunaan tuoda. Ikkunaan on tuotu seuraavanlaiset hälytykset: paine on kiinni, moottoreiden ohjaus on paikallisohjauksella (Liop_sel = 1). Tämä on tärkeä tuoda käyttöliittymään, koska jos kyseinen bitti on päällä, ei ohjauksena ole toiminnassa, vaan ohjaus toimii vain logiikka-ohjelmasta käsin. Tuomalla tämän tiedon hälytysik-

kunaan, ei tarvitse mennä logiikkaohjelmaan, saadaksesen ohjaussanan takaisin toimintaan. Hälytysikkuna tulee olemaan kuvan 17 mukainen.



Kuva 17. Hälytysikkuna.

Taajuusmuuttaja ikkuna

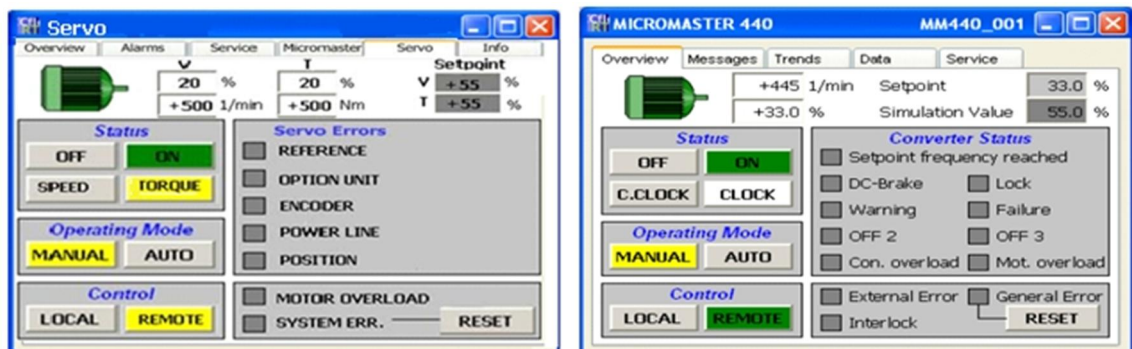
Taajuusmuuttaja ikkuna tulee olemaan hyvin samanlainen mitä Siemens on tehnyt. Poikkeuksena on vain se, että teen käyttöliittymään luvussa 6.5 määrittelemäni värimuutokset. Tämän lisäksi paneeliin jää taajuusmuuttajasta vain aloitusikkuna (overview-välilehti). Tämä johtuu siitä, että paneelin tila on hyvin rajallinen. Taajuusmuuttajasta en piirtänyt hahmotelmaa, koska sen toteutus tapahtuu muutoksena Siemensin toteuttamaan käyttöliittymään. Välilehti tulee olemaan servo- ja huoltoikkunan tavoin salasanalla suojattu.

Servoikkuna

Servoikkuna tulee myös olemaan melko samanlainen mitä taajuusmuuttajan ikkuna paitsi, että siihen tulee enemmän muutoksia. Kuvassa 18 näkyy hahmotelma tulevista muutoksista alkuperäiseen taajuusmuuttaja ikkunaan nähden. Muutokset tulevat olemaan seuraavanlaiset:

- pyörintäsuunnan ohjaus (C.Clock/Clock) muuttuu toimintatavan valinnaksi (Speed/Torque)
- asetusarvo (Setpoint) muuttuu nopeussäädön asetusarvon syötöksi (Setpoint V)

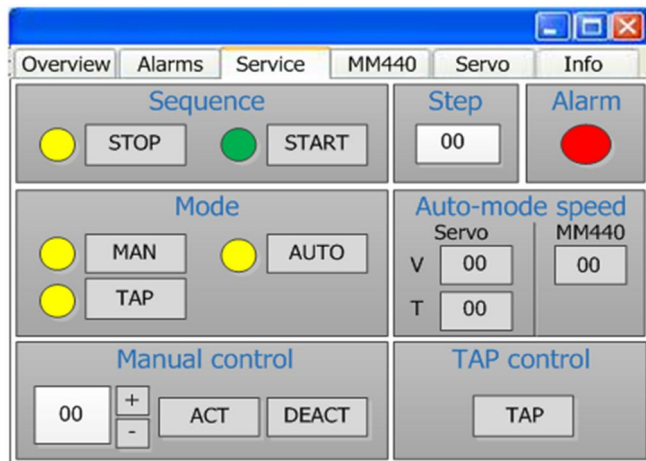
- simulointi syöttö (Simulation value) muuttuu momenttisäädön asetustarvon syötöksi (Setpoint T)
- kaksi lisäkenttää: momenttinopeus (%) ja pyörintänopeus (Nm)
- hälytystiedot/tilatiedot muuttuvat kokonaan. Servosta saadaan seitsemän virheilmoitus, jotka ovat: reference error, option unit error, encoder error, position error, power line error, servo overload ja system error.



Kuva 18. Servoikkunan muutokset.

Paneelin huoltoikkuna

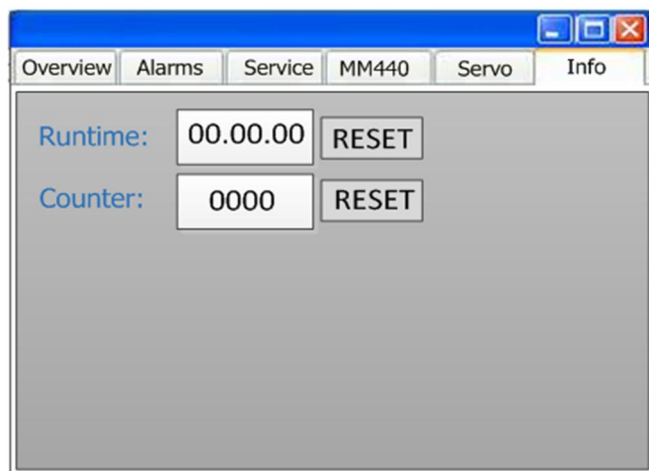
Huoltoikkunasta (kuva 19) huoltohenkilökunta pääsee muuttamaan sekvenssin toimintatapaa ja ajamaan sekvenssiä manuaali- ja tap-ohjauksella. Tämän lisäksi moottoreiden nopeuksia voi muuttaa tästä valikosta ohjauksen ollessa auto-ohjauksella. Tämä on laitettu tähän siksi, että alkuperäisessä taajuusmuuttaja ikkunassa ei ollut tälle arvon syöttöä ja sen lisääminen olisi tilanpuutteen takia erittäin hankalaa.



Kuva 19. Huoltoikkuna.

Paneelin infoikkuna

Info-ikkunan tarkoitus on antaa ajallisista ja määrällistä informaatiota prosessista. Sisältäen seuraavat tiedot prosessista: käyntiaika ja rainan laskuri. Kuvassa 20 näkyy info-ikkunan hahmotelma. Tähän ikkunaan voidaan myöhemmin lisätä lisää infoa, jos sille nähdään tarvetta.



Kuva 20. Info-ikkuna.

7.2 PC:n käyttöliittymäikkunat

PC-käyttöliittymän hahmottelu ja suunnittelu on jossain määrin helpompi, koska kaikki elementit mahtuvat yhteen ikkunaan. Tärkeintä siis on lähinnä elementtien oikeanlainen sijoittelu eli käytettävyyden ajattelua. Elementit ovat hyvin pit-

käلتi samoja mitä paneelissa, joten kopioituvuus helpottaa suunnittelua ja toteutusta. Tämän lisäksi PC-valvomo tukee faceplaten tekoa, joka helpottaa huomattavasti liikutettavuutta ja kopiointia. PC:n näytön resoluutio on 1280 x 1024 pikseliä, joka täytyy ottaa huomioon suunnittelu vaiheessa. Aluksi täytyi kuitenkin miettiä, että tarvitseeko paneelissa käytettäviin ikkunoihin tehdä muutoksia, vai pystyisikö ne kopioimaan sellaisenaan.

Ohjausikkuna

PC-valvomon ohjausikkunaan tulee lähinnä karsivia muutoksia paneelin nähdessä. Moottoreiden tilaikkunat ovat tässä ikkunassa turhat, koska ne näkyvät muutenkin. Kaikki muut välilehdet voi karsia pois paitsi service-välilehden. Muut välilehdissä olevat tiedot tulevat näkyviin pääsivulla. Service-ikkuna on hyvä pitää välilehden alla, koska kyseiseen välilehteen pääsemiseksi vaaditaan salasana.

Moottoreiden ikkunat

Ajajille näkyy moottoreiden tiedoista vain pienet tilaikkunat, koska muu ei ole tarpeellista tietoa heille. Huoltoryhmä pääsee pienistä tilaikkunoista suurempaan kokonaisuuteen klikkaamalla tätä ikkunaa ja syöttämällä määritetyn salasanan. Servo ja taajuusmuuttaja tilaikkunoihin tehdyt muutokset (luvussa 7.1) ovat muuten samankaltaiset, mutta PC-valvomoon voidaan tuoda myös muut alkuperäisessä käyttöliittymässä olevat välilehdet. Lisämuutoksena näihin ikkunoihin tulee se, että ohjauspainikkeiden viereen tulee tilavalot selkeyttämään kokonaisuutta. Tämän lisäksi service-välilehden alle tulee lisäyksenä auto-ohjauksen nopeudensäätö. Servomoottorin ikkunan välilehteen tulee muutoksia niin, että service-ikkunan alta jää pois simulointi ohjaus. Tämän lisäksi data-ikkuna jää kokonaan pois, koska servolta ei saa luettua DPV1-tietoja, niin kuin taajuusmuuttajaohjaimelta.

Muut tiedot

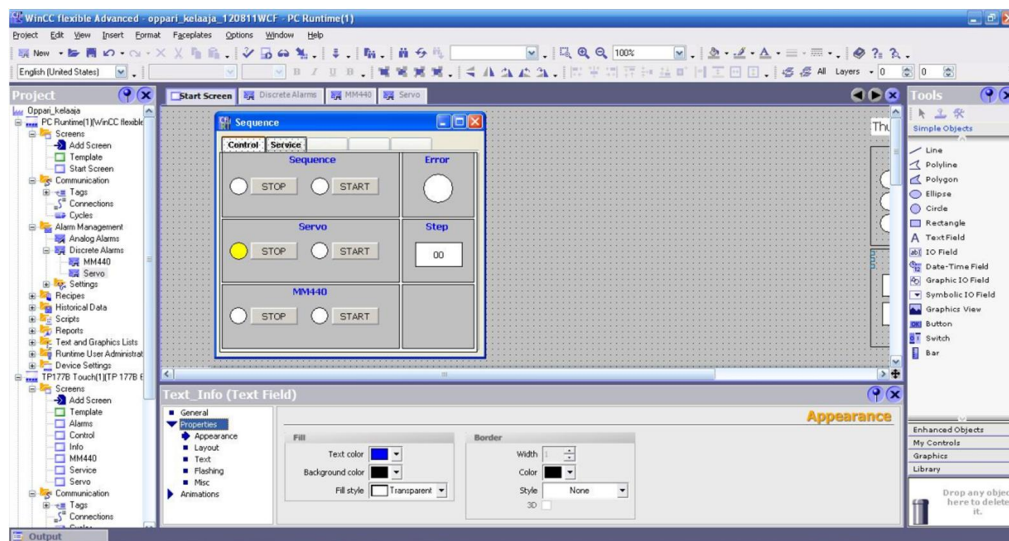
PC-valvomoon ei tule erillisiä ikkunoita yleisille hälytyksille ja muille informaatioille, vaan ne sijoittuvat ikkunan oikeaan reunaan. Eri tiedot rajataan erilleen toisistaan, jotta ne helpottavat hahmottavuutta. Tämän lisäksi poikkeuksena

paneeliin nähden, tähän yhteyteen laitetaan huoltoryhmän uloskirjautuminen ja runtime-tilasta poistuminen.

8 Käyttöliittymien toteutus

8.1 WinCC Flexible

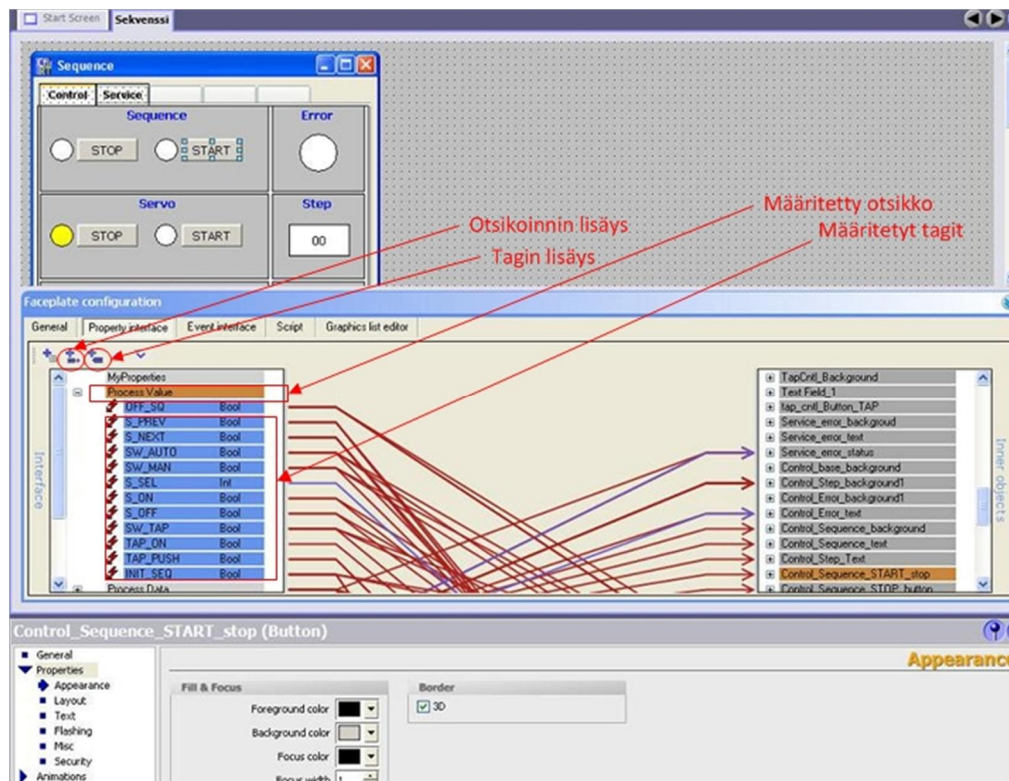
Käyttöliittymien toteutus tapahtui WinCC Flexible 2008 SP2-ohjelmalla. WinCC Flexible on Siemensin kehittämä graafinen ohjelmointiympäristö. Se tukee OPC-rajapintaa ja vaatii toimiakseen Windows-käyttöjärjestelmän. WinCC Flexible sisältää valmiita kirjastoja, jotka helpottavat käyttöliittymien luomista. Liitettävyyks Siemensin logiikoihin on todella yksinkertaista ja helppoa, mutta myös muihin logiikoihin se on mahdollista esim. Omron ja Mitsubishi (Siemens 2010b).



Kuva 21. WinCC Flexiblen yleisnäkymä.

Kuvassa 21 näkyy WinCC Flexiblen yleisnäkymä. Vasemmalta puolelta project-ikkunasta päästään liikkumaan eri valikoissa. Keskelle aukeaa project-ikkunasta valittu ikkuna/valikko ja tässä ikkunassa tapahtuu varsinainen käyttöliittymän teko. Keskelle alas aukeaa valitun objektin properties-ikkuna, josta päästään muokkaamaan valittua objektia. Oikealla puolella on ohjelmointityökalut ja WinCC Flexiblen kirjasto.

Faceplate-ominaisuus on omassa työssäni tärkein perehdyttävä ominaisuus, joten siihen perehtyminen oli laajaa. Aivan ensimmäinen vaihe faceplatea tehtäessä on määrittää faceplaten muuttujat (tagit). Muuttujat täytyy tuoda WinCC Flexblessä määritetyistä muuttujista. Faceplatea tehtäessä kannattaa tehdä otsikointi erottamaan eri muuttujia esim. tila- ja ohjaustiedot. Kuvasta 22 näkyy faceplaten kokonäkymä muokkausikkunassa, kuvaan on lisätty myös nuolet osoittamaan mistä otsikoinnit ja muuttujat määritetään ja minne ne määrittyvät. Ohjelma yhdistää automaattiset nuolet muuttujien ja objektien välille muuttujien vastapuolelle (oikealle), kun määrittää objektille tehtävän. Tästä syystä on tärkeää, että jokaisen objektin nimeää niin, että sen tunnistaa. Tämä siitä syystä, että jos ohjelma antaa virheilmoituksen tarkistusvaiheessa, niin virheellisen objektin voi paikantaa helpommin. Kun määritelmä on valmis, näyttää yleisnäkymä liitteen 7 kaltaiselta, missä täytyy määrittää muuttujat joita tarvitaan faceplatesa. Yksityiskohtaisemmat faceplate teko-ohjeet löytyvät Siemensin support-sivustolta (Siemens 2009).



Kuva 22. Faceplaten muokkaus näkymä.

8.2 Paneelin käyttöliittymän toteutus

Paneelin käyttöliittymän toteutuksessa ei pystynyt käyttämään faceplatea, koska faceplate on tuettu vasta TP270-sarjan tuotteista eteenpäin. Tästä syystä paneelin jokaisesta välilehden informaatioista täytyi tehdä oma ikkunansa. Vaikka objektien päällekkäisyyden olisi pystynyt toteuttamaan antamalla eri välilehtien objekteille eri layer-numerot, siitä huolimatta päädyin mainittuun ratkaisuun.

Toteutus oli helppo tehdä, koska hahmotelmat siitä, mitä painikkeiden pitää tehdä, olivat selkeät hyvän suunnittelun ansioista. Ikkunoiden yläpalkki ja välilehti palkki olivat valmiina WinCC Flexiblen kirjastoissa. Välilehti oli kuitenkin vaan viiden välilehtipaikan kokoinen, joten jouduin kuvankäsittelyohjelmalla muokkaamaan sitä niin, että sain kuudennenkin välilehden siihen. Toteutuneet käyttöliittymät näkyvät liitteessä 6.

Huoltoryhmälle täytyi määrittää salasana, jotta he pääsisivät kolmeen suojattuun välilehteen (luku 7.1). Käyttöliittymä kysyy salasanaa, kun välilehden yritetään avata. Käyttäjänimeksi ja salasanaksi määritin nyt oletuksena ABC. Paneelissa ei ole erillistä uloskirjautumista huoltoryhmälle, vaan paneeli kirjautuu automaattisesti ulos, kun paneelia ei ole käytetty kahteen minuuttiin. Huoltohenkilölle jää vastuu siitä, että siirtyy lähtiessään sellaiseen ikkunaan, joka on ajajallekin sallittu.

8.3 PC-käyttöliittymän toteutus

PC-käyttöliittymä koostuu yhdestä isosta ikkunasta, jossa on kolme faceplatea: sekvenssi-, taajuusmuuttaja- ja servo-ohjaus. Tämän lisäksi on vielä infoikkuna, josta ei ole tehty erillistä faceplatea, koska en nähnyt sen olevan tarpeellista. Taajuusmuuttajan ja servon toteutus tapahtui lähinnä vain muokkamalla valmista faceplatea. Sekvenssille täytyi tehdä täysin uusi faceplate.

Faceplaten teko on helppoa ja se helpottaa huomattavasti siirrettävyyttä ja kopiaitavuutta. Faceplaten voi tehdä kahdella tavalla, joko toteuttamalla käyttöliittymän valmiiksi yleisnäkyvässä ja tekemällä tästä faceplaten tai sitten toteuttamalla kaiken faceplaten muokkausikkunassa. Jälkimmäinen tapa on kuitenkin

helpompi vaihtoehto, koska tällöin kaikki objektit rakentuvat suoraan faceplanteen, toisessa tavassa voi käydä niin, että jokin objekti jää valikoimatta ja tällöin se jää faceplaten ulkopuolelle.

Moottoreiden tiedot messages-välilehteen tulevat WinCC Flexiblessä määritetyistä hälytyksistä. Hälytykset määritetään "DiscreteAlarms"-kohdasta, jossa määritetään muuttujat ja bitit, mistä hälytys tulee. Tämän lisäksi tekstien alkuun on laitettava käyttöliittymään määritetty instance-teksti (Esim. MM440_001: Motor is ON). Tällä tavalla käyttöliittymä tunnistaa ja sijoittaa hälytyksen oikeaan messages-kenttään. Tässä tapauksessa moottoreiden hälytykset tulevat "QwAlarms-muuttujista". Servolla on hälytyksiä vähemmän, koska tiedon saanti määrä on rajallisempi.

Huoltohenkilölle on luotu oletuksena sama käyttäjänimi ja salasana mitä paneelin yhteydessä (luku 8.2). PC-valvomoon on tuotu myös mahdollisuus huoltohenkilöille kirjautua ulos. Ohjelma kuitenkin kirjautuu ulos, jos PC:tä ei käytetä viiteen minuuttiin. Samalla tavalla kun paneelissa, huoltohenkilöllä on vastuu käytön jälkeen sulkea sellaiset ikkunat, jotka ovat ajajilta estettyjä. Toteutetut käyttöliittymät näkyvät liitteissä 8-10.

8.4 Testaus

Testaus toteutui monella eri tapaa. Ensiksi testasin taajuusmuuttajan ohjauksen ja tilatietojen näytön kaikissa mahdollisissa eri tilanteissa. Tämän jälkeen servon samalla tapaa. Moottorien toiminnan testauksen jälkeen, testasin sekvenssin toiminnan aluksi auto-ohjauksella ja tämän jälkeen moottoreiden tilaa vaihdellen. Tämän testauksen toteutin molemmille käyttöliittymille samalla tapaa käyttäen testauslistaa joka sisälsi kaikki ohjauksen – ja tilatiedot. Testauksessa ilmeni joitakin väärin määriteltyjä tilatietoja, mutta ei mitään suurempaa, mikä olisi vaatinut muutoksia ohjelmaan.

9 Pohdinta

Kokonaisuutta ajatellen tavoitteissa ei onnistuttu täydellisesti. Aikataulu petti osaksi ongelmatilanteiden takia ja osaksi liian tiukan aikataulun takia. Aikataulun siirto ei ollut onneksi kuitenkaan ongelma kummallekaan osapuolelle, joten sitä ei voida asettaa kovin suureksi haitaksi kokonaisuudelle. Kuitenkin keskeistä tavoitetta ajatellen eli käyttöliittymän toteutusta, voidaan sanoa, että tulokset saavutettiin.

Käyttöliittymien toteutuksia ajatellen mielestäni se täyttää käyttöliittymän suunnittelussa asetetut tavoitteet, eli se ei juuri riko pahasti käytettävyyttä eikä hahmolakia. Samankaltaisuus kuitenkin rikkoutui jonkin verran paneelin ja PC:n välillä, niiden koko eron vuoksi. Mielestäni ikkunat ovat hyvin samankaltaisia, joten se ei ole kovin suuri poikkeus, että servon ja taajuusmuuttaja ikkunan piirinäytöt ovat erillään. Pitää ottaa huomioon se, että PC-käyttöliittymään on tuotu enemmän tietoa näistä moottoreista. Näiden tietojen vieminen paneelille olisi aiheuttanut suurta tilaongelmaa, josta olisi seurannut vain käytettävyys ja selkeys puutteita. Tämän lisäksi jos PC-käyttöliittymästä olisi tehty täsmälleen samanlainen mitä paneelin käyttöliittymästä, olisi tämä ollut todella tilanhukkaan heittämistä.

Opinnäytetyönä kyseinen työ oli haastava ja mielenkiintoinen. Työ asetti monia haasteita, mutta niistä selvittiin. Haasteiden selvittämisestä oli opin kannalta erittäin paljon hyötyä. Työn suunnitelmallisuus auttoi työn etenemisessä, vaikka alkuperäistä aikataulua täytyi siirtää. Ohjaajan apu ongelmatilanteissa oli aina saatavilla kysyttäessä, joten tämä antoi työn teolle tietynlaista turvaa.

Prosessilla on monia jatkokehitys mahdollisuuksia. Lavikainen (2011) on luetellut omat jatkokehitysideansa opinnäytetyössään, jotka painottuvat säätötekniisiin asioihin. Turvapuolen liittäminen logiikkaan on ensimmäinen jatkokehityksen aihe. Tämän lisäksi käyttöliittymän jatkokehitystä on ainakin sen verran, että käyttöliittymästä voisi tehdä etäohjattavan. WinCC Flexiblessä tähän on erittäin hyvät mahdollisuudet. Toisen servo-ohjaimen liittäminen prosessiin mukaan on kanssa jatkokehityksen paikka, tällä hetkellä kyseinen servo ei toimi jostain syystä. Servon liittäminen käyttöliittymään ei kuitenkaan aiheuta suurta työtä,

koska se voidaan kopioida täysin toisesta tämän hetkisestä servon faceplateesta. Paneelin sijoittelu saattaa aiheuttaa tilanpuutteen takia pientä pohdiskelua, mutta sekään ei ole täysin ylitsepääsemätöntä. Paneelista voidaan rajata pois esim. info-kenttä, jossa on prosessin toimivuuden kannalta epäolennaista tietoa. PC-valvomoon toisen servo-ohjaimen lisääminen ei ole tilankaan puolesta mikään este.

WinCC Flexible ohjelman runsas käyttö tässä työssä nosti ohjelman henkilökohtaista osaamista hyvälle tasolle. Tämän lisäksi käyttöliittymien teorian läpikäynti kehitti käytettävyyttä ajattelua. Opiskelu aikanani tutustuin laajemmin WinCC Flexiblen lisäksi kahteen muuhun käyttöliittymäsovellukseen: InTouchin ja ClearScadän. InTouchin kokemus jäi näistä vähäisemmäksi siinä määrin, että sillä en toteuttanut mitään laajempaa kokonaisuutta. Tämän takia vertailukohta sillä muihin ei ole aivan optimaalinen. ClearScadalla toteutin projektikurssissa kiinteistönohjaukselle käyttöliittymän, joka oli melko laajakokonaisuus. Vertaillen näitä kolmea käyttöliittymää olen sitä mieltä, että WinCC Flexible on ainakin Siemens omia tuotteita käytettäessä mielestäni paras vaihtoehto. WinCC Flexiblen heikkona puolena on se, että se vaatii PC:ltä melko paljon tehoa, varsinkin tehtäessä faceplate-käyttöliittymää. Kuitenkin juuri tämä faceplate-ominaisuus on parempi mitä ClearScadalla ja InTouchilla, joissa kyseinen toiminta rajoittuu siihen, että objektit linkittyvät yhdeksi objektiksi, ei muuta. Voinkin todeta, että lähtökohta eli käytössä olevat laitteistot ja tarvittavat käyttöliittymä ominaisuudet määrittävät hyvin pitkälti, mikä käyttöliittymäsovellus on paras valinta.

Lähteet

A-LINK. 2007. Roadrunner 24 manual.

<ftp://ftp.a-link.com/rr24i/3.7.1/Quick.pdf> [1.7.2011]

AS-interface. 2011. Facts.

<http://as-interface.net/System/Facts> [1.9.2011]

Becker R. 2009. Automation is easy – with AS-interface. Frankfurt: Henrich Druck + Medien GmbH.

Fonselius J., Rinkinen J., Vilenius M., 1998. Servotekniikka. Helsinki: Oy Edita Ab.

Kuutti W. 2003. Käytettävyyden suunnittelu ja arviointi. Helsinki: Talentum.

Lavikainen T. 2011. Kelaajalaitteen suunnittelu ja toteutus opetusympäristöksi. Joensuu: Pohjois-Karjalan ammattikorkeakoulu, tietotekniikan koulutusohjelma.

Minimotor. 1999. Worm Gear Motor – MC.

<http://www.minimotor.com/downloads/pdf/MC.pdf> [3.5.2011]

Niiranen J. 1999. Sähkömoottorikäytön digitaalinen ohjaus. Helsinki: Otatieto.

Rossiter S. 2011. Benefits of Variable Speed Drives.

http://www.emcsolutions.com/article_vfd_benefits.html [1.8.2011]

Siemens. 2001. Micromaster 440 – Parameter list.

http://cache.automation.siemens.com/dnl/TU3Nzg3AAAA_9294860_HB/MM440_PLI_1201_en.pdf [2.5.2011]

Siemens. 2009. How do you create a WinCC flexible faceplate?

<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&objId=23427291&nodeid0=16502685&load=treecontent&lang=en&siteid=cseus&aktprim=0&objaction=csview&extranet=standard&viewreg=WW> [2.9.2011]

Siemens. 2010a. Sample Blocks for STEP 7 and WinCC flexible.

http://support.automation.siemens.com/WW/llisapi.dll/csfetch/36435784/36435784_Documentation_e.zip?func=cslib.csFetch&nodeid=39297489 [1.5.2011]

Siemens. 2010b. SIMATIC WinCC flexible - Flexibility in any HMI applications – from the Micro Panel to the PC.

http://www.automation.siemens.com/salesmaterial-as/brochure/en/brochure_simatic-wincc-flexible_en.pdf [15.7.2011]

Sinkkonen I., Kuoppala H., Parkkinen J. & Vastamäki R. 2002. Käytettävyyden psykologia. Helsinki: Edita.

Suomen Automaatioseura ry. 2010. Valvomo – Suunnittelun periaatteet ja käytännöt. Helsinki: Suomen Automaatioseura ry.

Vacon. 2006. Käyttäjänkäsikirja – NX-taajuusmuuttajat – Profibus DP optiokortit.

<http://www.vacon.fi/File.aspx?id=466328&ext=pdf&routing=396771&webid=396774&name=UD01142A> [15.7.2011]

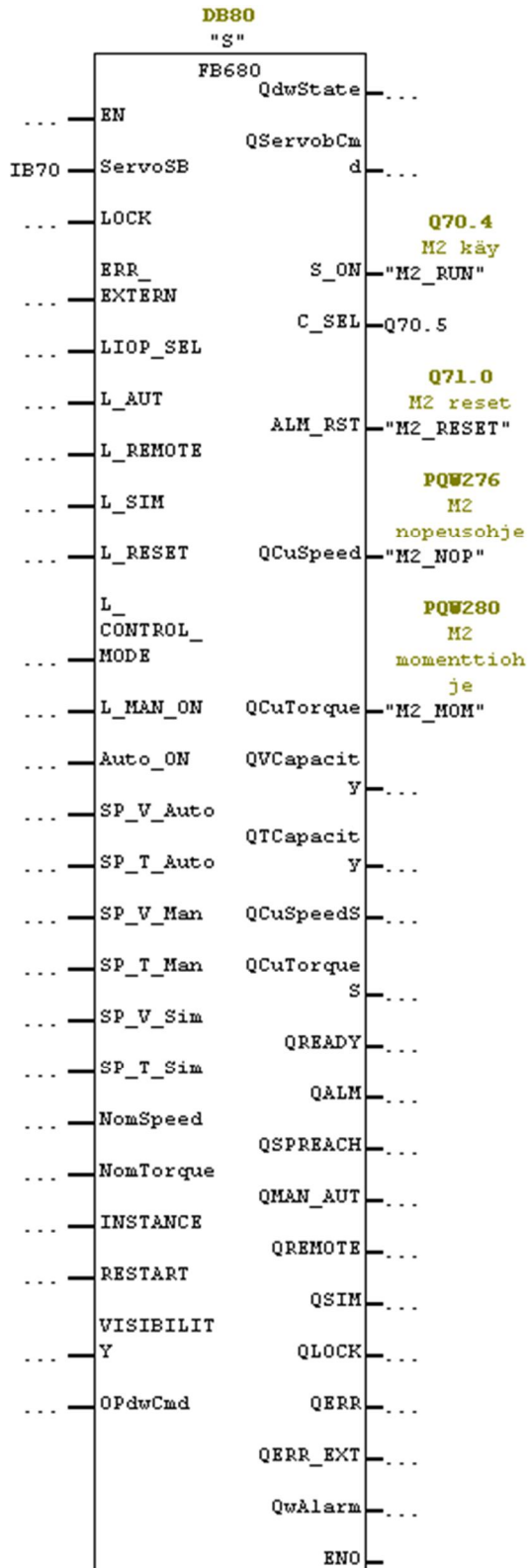
Taajuusmuuttajalohko

```

DB68
Instance
database
of "Motor"
"M"
FB681
Interface FB for a
MicroMaster 440
"MM4"
QdwState ...
... EN QMMdwCmd PQD272
PID272 MMdwInp QCuSpeed ...
... LOCK QSPCapaci
... ERR_ ty ...
... EXTERN QrCuCapac
... LIOP_SEL ity ...
... L_AUT QSPFreque
... L_REMOTE ncy ...
... L_SIM QrCuFrequ
... L_RESET QSPSpeed ...
... L_MAN_ON QrCuSpeed ...
... L_MAN_ QPOWER_ON ...
... REVERSE QREADY_
... L_ENABLE RUN ...
... L_RFC_EN QOP_
... L_RFC_ ENABLE ...
... FREE QFAULT ...
... L_SP_EN QNOFF2 ...
... L_SP_ QNOFF3 ...
... VALID QSLOCK ...
... OFF2 QWARN ...
... OFF3 QSPREACH ...
... Auto_ON QFRREACH ...
... Auto_REV QNMOTWARN ...
... SP_Auto QNBRAKE ...
... SP_Man QNMOTOV ...
... SP_Sim QDIRECTIO
... SIM_ N ...
... nomFreq QNCONOV ...
... SIM_ QMAN_AUT ...
... nomSpeed
#16#7FC DPV1_ID QREMOTE ...
... DPV1_READ QSIM ...
... INSTANCE QLOCK ...
M50.0 RESTART QERR_CON ...
... VISIBILIT QERR ...
... Y QERR_EXT ...
... OPdwCmd DPV1_RV
... ERROR ...
... DPV1_
... VOLTAGE ...
... DPV1_
... POWER ...
... DPV1_
... FREQUENCY ...
... DPV1_
... SPEED ...
... DPV1_
... ERROR1 ...
... DPV1_
... ERROR2 ...
... DPV1_
... ERROR_
... VALUE1 ...
... DPV1_
... ERROR_
... VALUE2 ...
... DPV1_
... FNumR ...
... DPV1_
... FNumW ...
... QvAlarm ...
... ENO

```

Servolohko



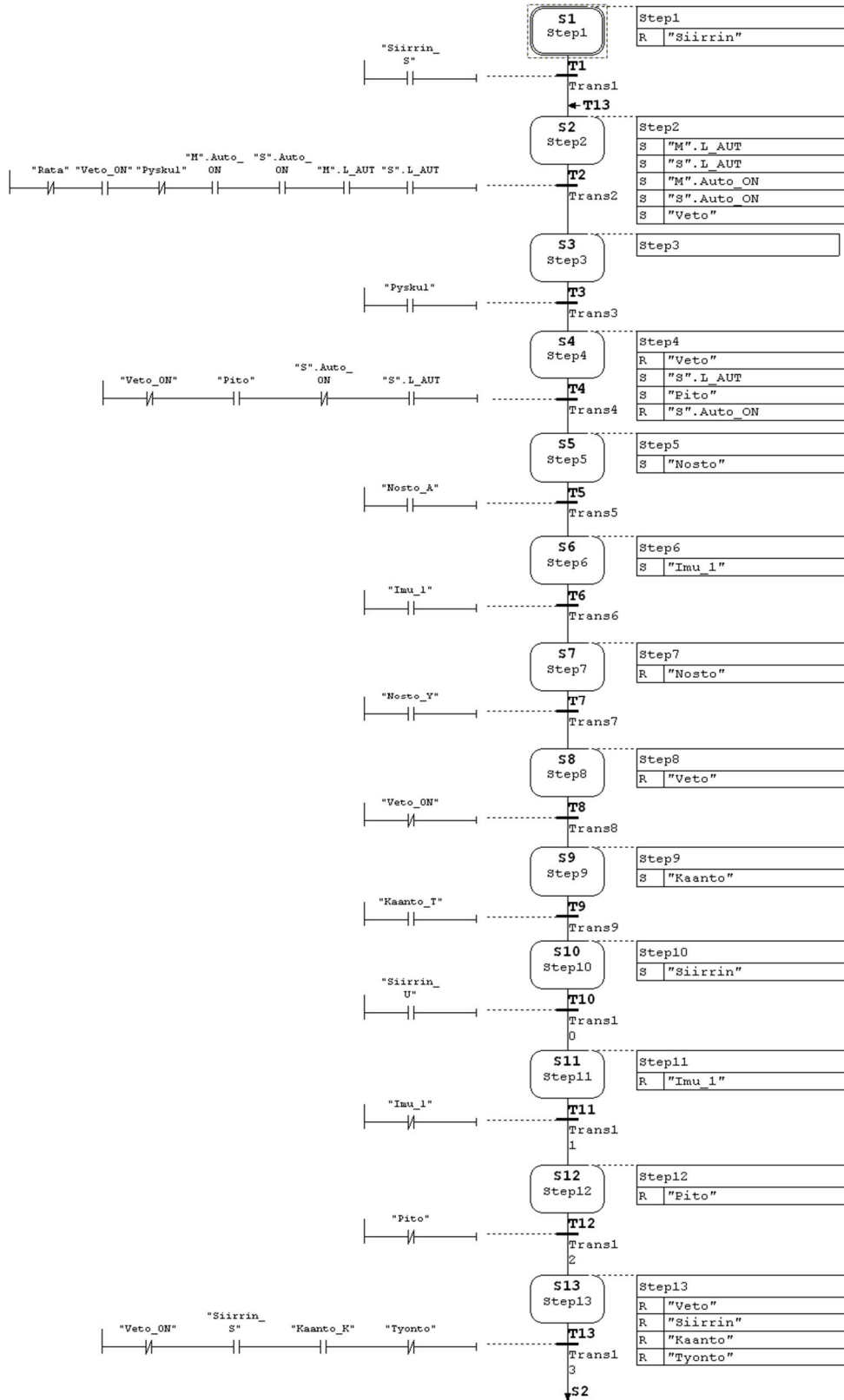
Toimintasekvenssi

1. Perusasema
 - Laite asetetaan perusasemaan
2. Veto päälle
 - Veto pannaan päälle, jolloin raina liikkuu.
 - Taajuusmuuttaja moottori päälle
 - Taajuusmuuttajan auto-moodi päälle
 - Servomoottori päälle
 - Servon auto-moodi päälle
3. Pysäytyskuitu
 - Pysäytyskuitu tunnistaa pysäytyskohdan.
4. Jarru päälle
 - Jarru pannaan päälle, jolloin raina ei liiku
 - Veto pois.
 - Servomoottori pois päältä
5. Nostin alas
 - Nostin alas eli haetaan tarra
6. Imu päälle
 - Imu päälle, jolloin tarra kiinnittyy nostolaitteeseen.
7. Nostin ylös
8. Veto sisään
 - Vetolaite siirretään sisäasentoon
9. Kääntö
10. Siirto
 - Siirto, jolloin tarra siirretään kohdelaitteeseen.
11. Imu pois
 - Imu pois, jolloin tarra irtoaa ja kiinnittyy kohdelaitteeseen
 - Veto ulos
 - Vetolaite siirretään ulkoasentoon.
 - Työnnin siirretään sisäasentoon
12. Jarru pois
 - Jarru vapautetaan.
13. Toisto, vaiheeseen 2
 - Uusi työkierto vaiheesta 2.

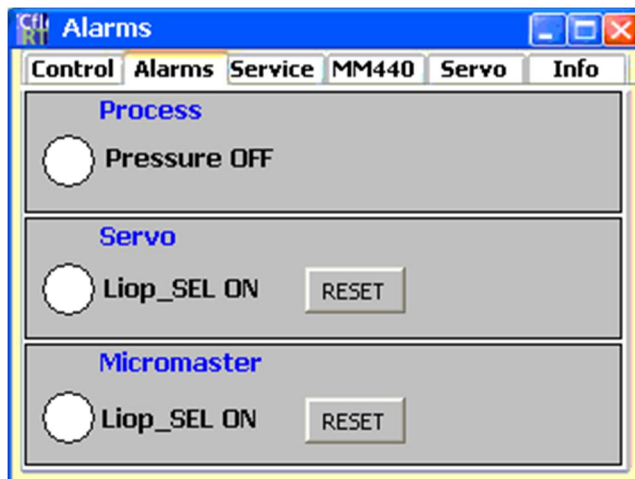
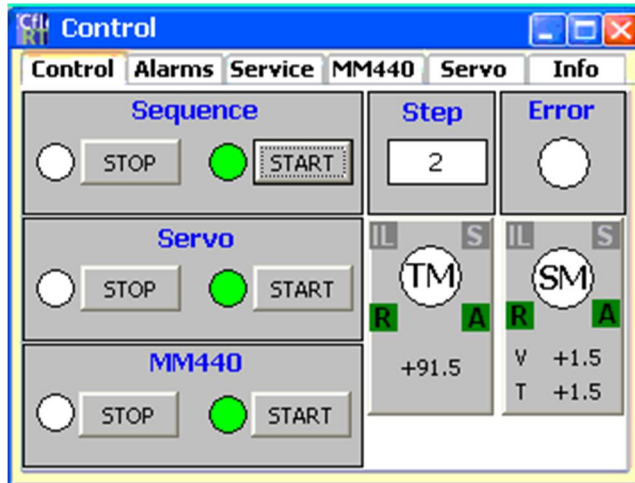
Sekvenssin tila- ja ohjausbitit

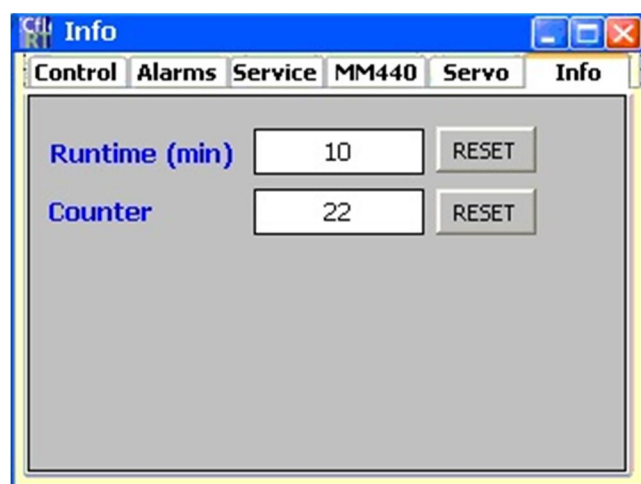
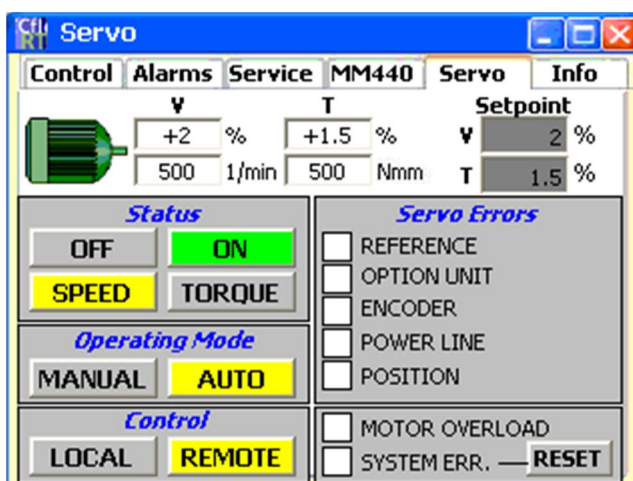
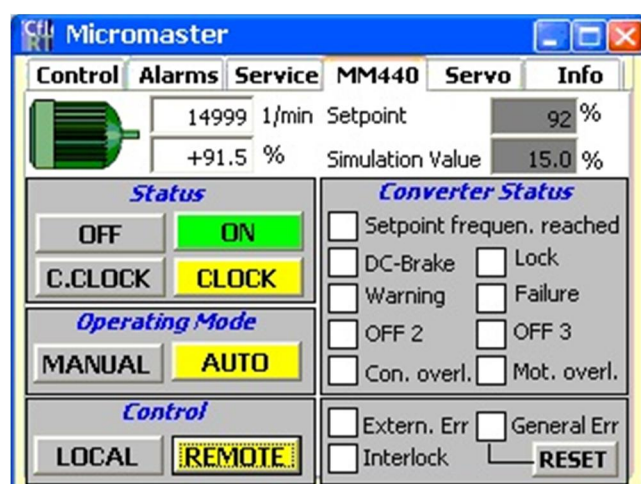
	Address	Symbol	Display format	Status value	Modify value
1	//Tulot				
2	I 0.0	"Nosto_Y"	BOOL		
3	I 0.1	"Nosto_A"	BOOL		
4	I 0.2	"Siirrin_S"	BOOL		
5	I 0.3	"Siirrin_U"	BOOL		
6	I 0.4	"Kaanto_K"	BOOL		
7	I 0.5	"Kaanto_T"	BOOL		
8	I 0.6	"Tyonto_S"	BOOL		
9	I 0.7	"Tyonto_U"	BOOL		
10	I 1.0	"Veto_ON"	BOOL		
11	I 1.1	"Pysku1"	BOOL		
12	I 1.2	"Rata"	BOOL		
13	I 1.3	"Nauha"	BOOL		
14	I 1.4	"Pysku2"	BOOL		
15	I 1.5	"Paine"	BOOL		
16	I 1.6	"Suodatin"	BOOL		
17	I 1.7	"Res 1"	BOOL		
18	//Lähdöt				
19	Q 30.0	"Nosto"	BOOL		
20	Q 30.1	"Siirrin"	BOOL		
21	Q 30.2	"Kaanto"	BOOL		
22	Q 30.3	"Imu_1"	BOOL		
23	Q 30.4	"Imu_2"	BOOL		
24	Q 30.5	"Tyonto"	BOOL		
25	Q 30.6	"Veto"	BOOL		
26	Q 30.7	"Pito"	BOOL		
27	Q 31.0	"Kelaus"	BOOL		
28					
29	//Moottorit				
30	DB80.DBX 1.3	"S".L_AUT	BOOL		
31	DB80.DBX 2.1	"S".Auto_ON	BOOL		
32	DB68.DBX 4.3	"M".L_AUT	BOOL		
33	DB68.DBX 6.0	"M".Auto_ON	BOOL		
34					

Sekvenssiohjelman



Paneelin käyttöliittymä

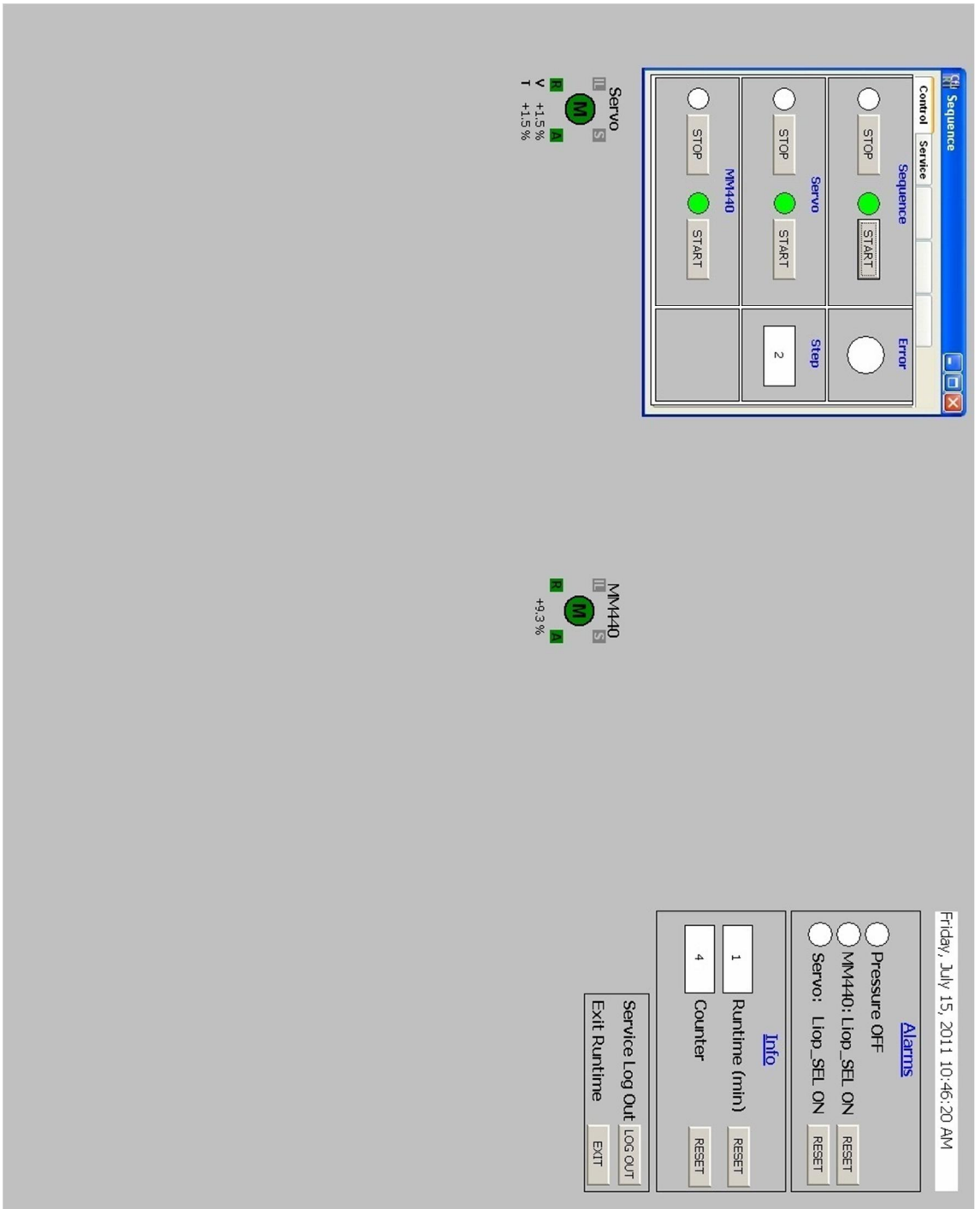




Faceplaten näkymä yleisnäkymässä

Process Value		
OFF_SEQ	DB1.OFF_SEQ ▼	1 s
S_PREV	DB1.S_PREV ▼	1 s
S_NEXT	DB1.S_NEXT ▼	1 s
SW_AUTO	DB1.SW_AUTO ▼	1 s
SW_MAN	DB1.SW_MAN ▼	1 s
S_SEL	DB1.S_SEL ▼	1 s
S_ON	DB1.S_ON ▼	1 s
S_OFF	DB1.S_OFF ▼	1 s
SW_TAP	DB1.SW_TAP ▼	1 s
TAP_ON	DB1.TAP_ON ▼	1 s
TAP_PUSH	DB1.T_PUSH ▼	1 s
INIT_SEQ	DB1.INIT_SEQ ▼	1 s
Process Data		
S_NO	DB1.S_NO ▼	1 s
S_ACTIVE	DB1.S_ACTIVE ▼	1 s
AUTO_ON	DB1.AUTO_ON ▼	1 s
MAN_ON	DB1.MAN_ON ▼	1 s
ERROR	DB1.ERR_FLT ▼	1 s
Motors Control		
Servo_Cmd	Servo.OPdwCmd ▼	100 ms
MM440_Cmd	MM4DB.OPdwCm ▼	100 ms

PC-valvomon käyttöliittymä



Sequence

Control Service

Sequence	<input type="radio"/> STOP <input checked="" type="radio"/> START	Error	<input type="radio"/>
Servo	<input type="radio"/> STOP <input checked="" type="radio"/> START	Step	<input type="text" value="2"/>
MM440	<input type="radio"/> STOP <input checked="" type="radio"/> START		

Servo

M S

V +1.5 %
T +1.5 %

Servo_001

Overview Messages Trends Service

V +500 1/min
T +1.5 %

Service: Sepoint V +500 1/min, Sepoint T +1.5 Nmm, Sepoint V 1.5 %, Sepoint T 1.5 %

Status: OFF, SPEED, TORQUE, ON

Operating Mode: MANUAL, AUTO

Control: LOCAL, REMOTE

Servo Errors: Reference, Option unit, Encoder, Power line, Position, Motor Overload, System Error

Friday, July 15, 2011 11:00:07 AM

Alarms

Pressure OFF

MM440: Llop_SEL ON

Servo: Llop_SEL ON

Info

Runtime (min)

Counter

Service Log Out

Exit Runtime

MM440

M S

+58.5 %

MM440_001

Overview Messages Trends Service Data

V 9577 1/min
T +58.5 %

Service: Sepoint 91.6 %, Simulation Value 1510 %

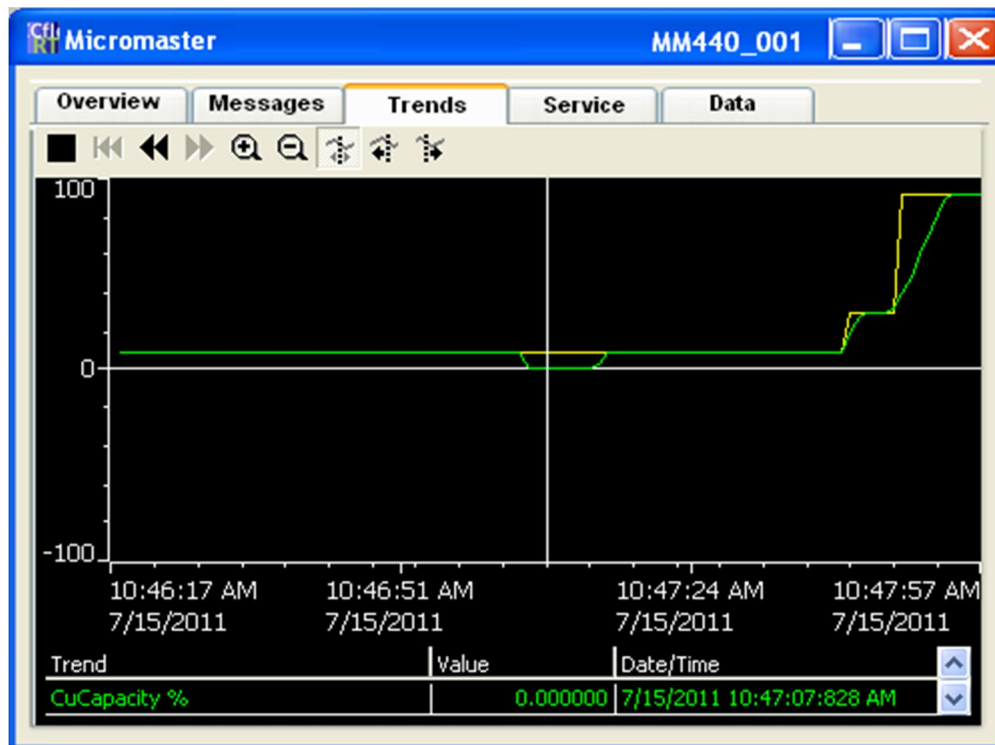
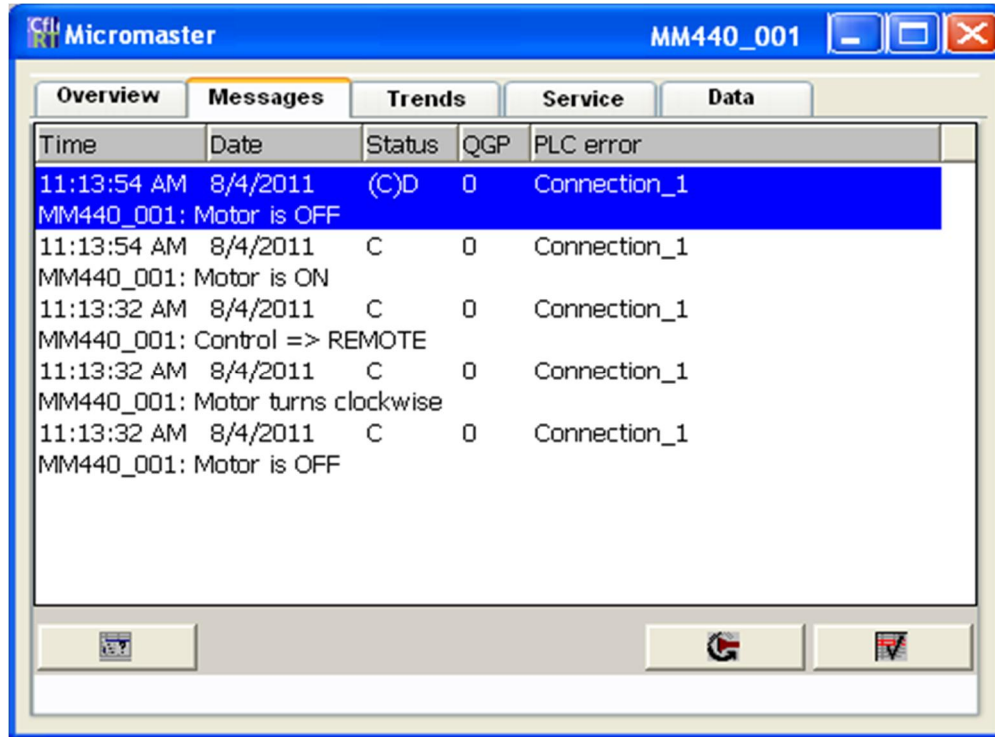
Converter Status: Sepoint frequency/reached, DC-Brake, Lock, Warning, Failure, OFF 2, OFF 3, Con. overload, Mot. overload

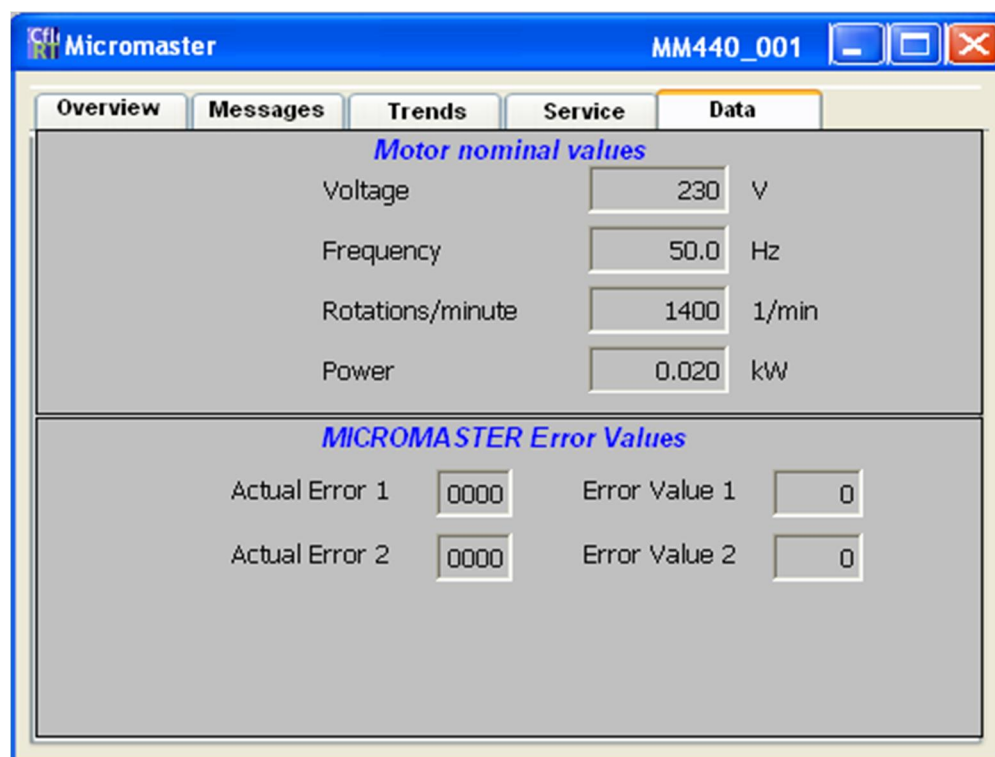
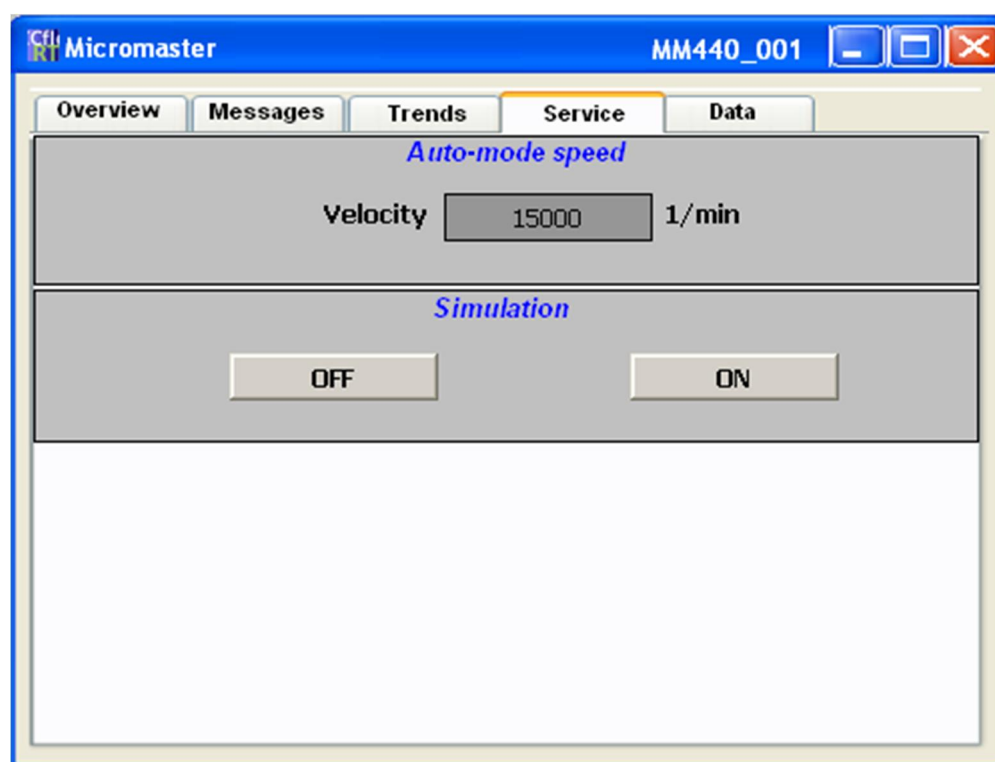
Operating Mode: MANUAL, AUTO

Control: LOCAL, REMOTE

Errors: External Error, Interlock, General Error

PC-valvomon taajuusmuuttajan välilehdet





PC-valvomon servon välilehdet

