

Toimilohko- ja symbolikirjasto TIA-Portaaliin

Opinnäytetyö

Jesse Laajisto

Tammikuu 2013

Automaatiotekniikka
Tekniikan ja liikenteen ala





Tekijä(t) LAAJISTO, Jesse	Julkaisun laji Opinnäytetyö	Päivämäärä 19.1.2013
	Sivumäärä 29	Julkaisun kieli Suomi
		Verkojulkaisulupa myönnetty (X)
Työn nimi Toimilohko- ja symbolikirjasto TIA-Portaaliin		
Koulutusohjelma Automaatiotekniikan koulutusohjelma		
Työn ohjaaja(t) SELOSMAA, Seppo		
Toimeksiantaja(t) JEEC Oy PEKKANEN, Jarmo		
Tiivistelmä <p>Opinnäytetyön päätavoitteena oli luoda toimilohko- ja symbolikirjasto Siemens TIA-Portaaliin, joka vähentää suunnittelijan työtä nopean määrittelyn ja parametroidin sekä sulautettujen toimintojen avulla. Kirjastoa tulla käyttämään Siemensin ohjelmoitaville logiikoille toteutetuissa sovelluksissa. Opinnäytetyön tilaajana oli jyväskyläläinen suunnittelu- ja konsultointialan yritys JEEC Oy, jonka ydinosajärjestelmään kuuluu automaatio-, sähkö- ja instrumentointisuunnittelu.</p> <p>TIA-Portaalilla toteutettuun kirjastoon sisällytettiin yleisimpiä toimilohkoja kuten moottori-, venttiili-, mittaus- ja säädintoimilohko. Opinnäytetyössä käydään läpi edellä mainittujen toimilohkojen valmistusprosessia ja niiden ominaisuuksia. Toimilohkokirjaston lisäksi toteutettiin symbolikirjasto, joka sisältää näyttösymbolit WinCC-käyttöliittymään. Toimilohkoa kohden valmistettiin kolme piirinäyttöä, jotka jakautuivat laitteen monitorointiin, ohjaukseen ja parametroidiin. Piirinäytöt tuli toteuttaa siten, että yhden osoitteen avulla määritetään kaikki symboliin ja toimilohkon välillä tapahtuva kommunikointi.</p> <p>Täysin saumattomasti toimivat toimilohkot ja symbolit vähentivät automaatio-sovelluksen valmistukseen kuluva aikaa merkittävästi. Kehitystyö on tuonut toimeksiantajalle lisää kilpailukykyä ja mahdollisuuden myös tarjota asiakkaalle informatiivinen käyttöliittymä mahdollisimman kustannustehokkaasti.</p>		
Avainsanat (asiasanat) TIA-Portal, WinCC, Siemens, Step7, toimilohkokirjasto, symbolikirjasto, ohjelmointi		
Muut tiedot Liihteenä venttiiliin ja mittauksen laajennetut piirinäytöt, 1 sivu.		



Author(s) LAAJISTO, Jesse	Type of publication Bachelor's Thesis	Date 19.1.2013
	Pages 29	Language Finnish
	Confidential () Until	Permission for web publication (X)
Title Function and symbol library to TIA-Portal		
Automation technology		
Tutor(s) SELOSMAA, Seppo		
Assigned by JEEC Oy PEKKANEN, Jarmo		
Abstract <p>The purpose of this bachelor's thesis is to create a function and symbol library for Siemens TIA-Portal software, which reduces programming time with integrated functions and easy setup. The library will be used when programming Siemens logic controllers. This thesis was assigned by JEEC Ltd, which offers automation engineering and consultant services.</p> <p>The library consists of the most common function blocks such as motor, valve, measurement and control. This thesis goes through the manufacturing process of the function blocks and their features. The symbol library was produced for the WinCC user interface. The function blocks have three different loop windows which divide into three groups; monitoring, controlling and parameterization. Communication between function block and symbol was arranged with one address, so setting up communication is fast.</p> <p>Software manufacturing time is reduced significantly using the symbol and functional block library. The development of the library has increased JEEC's competitiveness as it enables the selling of control systems with a user friendly user interface.</p>		
Keyword TIA-Portal, WinCC, Siemens, Step7, function library, symbol library, software development		
Miscellaneous Screenshots of valve and measurement loop windows as attachments, 1 page.		

Sisältö

Kuviot.....	2
Sanasto	3
1 Toimeksiantaja ja tavoitteet	4
1.1 Toimeksiantaja.....	4
1.2 Opinnäytetyön tavoite	4
1.3 Henkilökohtaiset tavoitteet.....	5
2 Opinnäytetyön tietoperusta.....	5
2.1 Ohjelmoitavalla logiikalla toteutettava automatisointiprojekti	5
2.2 Simatic S5.....	6
2.3 Simatic S7.....	7
2.4 WinCC Flexible	8
2.5 TIA-Portaali.....	9
2.6 Standardi ohjelmakielet.....	10
2.6.1 Function block diagram.....	10
2.6.2 Ladder Diagram	10
2.6.3 Structure text	11
2.6.4 Instruction List.....	11
2.6.5 Sequential Function Chart	11
3 Toteutus	13
3.1 Toimilohkot	13
3.1.1 Hälytykset.....	14
3.1.2 Lukitukset.....	15
3.2 Moottori.....	16
3.2.1 Suoraohjaus	16
3.2.2 Taajuusmuuttajaohjaus.....	17
3.2.3 Hälytykset.....	17
3.3 On/off -venttiili.....	17
3.3.1 Ohjaus	17
3.3.2 Hälytykset.....	18
3.4 Säädin.....	19
3.4.1 Ohjaus	19
3.4.2 Hälytykset.....	19
3.4.3 Lukitukset	20
3.5 Analogiamittaus	20
3.5.1 Vikatilanne	20
3.5.2 Hälytykset.....	21
3.6 Käyttöliittymä	21

3.6.1	Näyttösymbolit	22
3.6.2	Monitorointi- ja ohjauspiirinäyttö.....	22
3.6.3	Laajennettu piirinäyttö	24
4	Tulokset	26
4.1	Saavutettu etu	26
4.2	Kehityskohteet	27
5	Pohdinta	28
	Lähteet	29
	Liitteet	31
	Liite 1. Venttiilin laajennettu piirinäyttö	31
	Liite 2. Mittauksen laajennettu piirinäyttö	31

Kuviot

KUVIO 1.	Mittaustoimilohkosta generoituja datablockeja.....	14
KUVIO 2.	Moottorin laajennettu piirinäyttö.....	16
KUVIO 3.	Moottori symboli	23
KUVIO 4.	Moottorin operointinäkö	23
KUVIO 5.	Säätimen laajennettu piirinäyttö	25

Sanasto

HMI	Human Machine Interface, koneen ja ihmisen välinen rajapinta, esimerkiksi operointipaneeli.
Käyttöliittymä	Järjestelmä, jonka kautta hallitaan konetta.
Metso DNA	Dynamic Network of Applications, Metson automaatiojärjestelmä.
MS-DOS	Microsoft Disk Operating System, teksti pohjainen käyttöjärjestelmä.
PLC	Programmable Logic Controller, ohjelmoitava logiikka.
SCADA	Supervisory Control And Data Acquisition, PC-pohjainen hallinta- ja tiedonkeruuohjelmisto.
Step7	Siemensin logiikoille suunniteltu ohjelmointityökalu.
TIA-Portaali	Totally integrated automation, Siemensin ohjelmointityökalu logiikoille, käyttöliittymille, taajuusmuuttajille ja turvajärjestelmille.
WinCC	Siemensin käyttöliittymätyökalu.

1 Toimeksiantaja ja tavoitteet

1.1 Toimeksiantaja

Opinnäytetyön tilaajana oli jyväskyläläinen suunnittelu- ja konsultointialan yritys JEEC Oy. Yrityksen ydinosamisalue on automaatio, jota tukee sähkö- ja instrumentointisuunnittelupalvelut (JEEC Oy, 2010). Yritys on perustettu vuonna 2009 ja se työllistää vakituisesti kymmenen henkilöä. Nuoresta iästään huolimatta yritys sisältää vuosikymmenien kokemusta sähkö- ja automaatio-suunnittelusta.

1.2 Opinnäytetyön tavoite

Opinnäytetyön päätavoitteena oli luoda toimilohko- ja symbolikirjasto, joka nopeuttaa suunnittelijan työtä nopean määrittelyn ja parametroinnin avulla. Lisäksi toimilohkoihin haluttiin sisällyttää automaattisesti generoituvat hälytysviestit, jotta välttyttäisiin rutiinin omaisilta työvaiheilta. Toimilohkojen lisäksi tuli toteuttaa vastaavat piirinäytöt operointipaneelille, jotta operaattorilla olisi mahdollisuus hallita ja monitoroida prosessin tai koneen toimintaa.

Kehitystyön myötä sovellussuunnittelu TIA-ympäristössä tehostuu: projektin keskeisimmät toimilohkot ja näyttösymbolit löytyvät kirjastossa, eikä niitä tarvitse jokaisessa projektissa luoda erikseen uudelleen.

1.3 Henkilökohtaiset tavoitteet

Opinnäytetyöprosessin tavoitteenani oli oppia suunnittelutoimiston käytänteitä ja toimintamalleja. Tämä tarkoittaa sitä, että opinnäytetyöprojektin jälkeen kykenen tulkitsemaan sovellussuunnittelun tukena esiintyviä dokumentteja, hakemaan tietoa, käyttämään TIA-portaalin sovellussuunnittelutyökäytäntöä ja toteuttaa sen avulla sovelluksia itsenäisesti. Toissijaisena tavoitteena oli tutustua automaatioalalla työskenteleviin ihmisiin, nähdä automaatioinsinöörin työtä käytännössä ja oppia työelämän käytäntöjä. Näitä oppeja kykenen soveltamaan tekemässäni työssä, joka antaa minulle huomattavaa etua kiristyneillä automaatioalan työmarkkinoilla.

2 Opinnäytetyön tietoperusta

2.1 Ohjelmoitavalla logiikalla toteutettava automatisointiprojekti

Automatisoinnin tarkoituksena on saada jokin prosessi tai prosessin osa toimimaan automaattisesti, ilman käyttäjän jatkuvaa valvontaa tai puuttumista prosessin kulkuun (Keinänen, Kärkinen, Metso & Putkonen 2001, 248). Ohjelmoitavalla logiikalla toteutettava automatisointiprojekti edellyttää kattavaa lähtötietoaineistoa, joiden mukaisesti projektia toteutetaan. Ihanteellisessa tilanteessa sovellussuunnittelijalla on käytettävissä seuraavanlaisia lähtötietoja:

- prosessi- ja instrumentointikaavio
- piiri- ja laiteluettelo
- toimintakaaviot

- toimintakuvaukset
- lukituskaaviot
- sekvenssi- ja ajokaaviot
- hälytys- ja raporttimäärittelyt

Yleisesti suunnittelutiedot kuitenkin muuttuvat sovellussuunnitteluprojektin aikana, ja myös uusia projektiin vaikuttavia tietoja ilmenee suunnitteluvaiheen aikana (Tolonen, 2010, 5.)

2.2 Simatic S5

Simatic S5 –tuoteperhe ilmestyi vuonna 1979. Yli 30 vuotta käytössä ollutta S5-ohjausjärjestelmää esiintyy edelleen Suomen teollisuudessa, vaikka järjestelmä on vähitellen siirtymässä syrjään. Paineita S5-järjestelmän uusimiselle on aiheuttanut kasvaneet ylläpito- ja huoltokustannukset, järjestelmän epävakaas, heikko suorituskyky, huono varaosien saatavuus ja vanhenunut ohjelmointiympäristö. S5-tuoteperheen logiikat ovat ohjelmoitavissa Step5 työkalulla, joka toimi 6.5 versiosta lähtien MS-DOS-käyttöjärjestelmässä. (Siemens Oy. Kotisivut. Simatic S5. 2013, Wallenius, 2012, 11.)

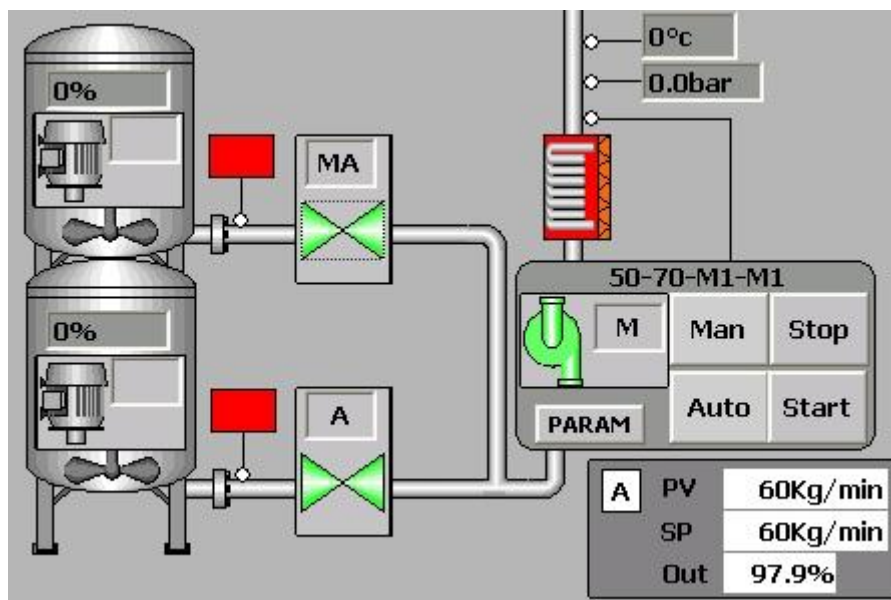
2.3 Simatic S7

Simatic S7 –tuoteperhe ilmestyi markkinoille vuonna 1995. Ilmestymisen jälkeen tuoteperhe nousi Siemensin suosituimmaksi logiikkasarjaksi yhdessä ohjelmointityökalun kanssa. Simatic Step7 –ohjelmointiympäristössä on suuria uudistuksia verrattuna edeltäjäänsä Step5 –työkaluun. Uudistuksista merkittävin oli siirtyminen MS-DOS:sta Windows-pohjaiseen käyttöliittymään. Käyttöliittymävaihdon myötä suunnittelussa pystyttiin ensimmäistä kertaa käyttämään näppäimistöä ja hiirtä. (Wallenius, 2012, 12)

Simatic S7 –tuoteperheen oli tarkoituksena syrjäyttää vanhentunut Simatic S5 –tuoteperhe. Siirtymistä S5-ohjausjärjestelmästä S7-ohjausjärjestelmään on helpotettu Step7-työkalusta löytyvällä konvertointi-työkalulla, jonka avulla voidaan Step5 –työkalun avulla toteutettu projekti kääntää Step7 -projektiksi. Mikäli ohjattavaa prosessia tai laitetta ei muuteta oleellisesti järjestelmän uusimisen yhteydessä, on mahdollista käyttää valmiita johdinsarjoja. Valmiita johdinsarjoja käyttämällä saadaan S5-korttien etupistoke kytkettyä suoraan vastaavaan S7-300- tai S7-400-sarjan I/O- korttiin. Näiden ominaisuuksien myötä järjestelmän vaihdos on erittäin vaivatonta ja tuotantoon aiheutettu katkos pysyy mahdollisimman lyhyenä. (Siemens Oy. Kotisivut. Simatic S5. 2013.)

2.4 WinCC Flexible

WinCC Flexible –ohjelmistolla toteutettu HMI-käyttöliittymä visualisoidaan operointipaneelilla tai tietokonevalvomolla. ”HMI” tulee sanasta human machine interface ja se tarkoittaa ihmisen ja koneen välistä rajapintaa. HMI-käyttöliittymällä ohjataan tavallisesti yksittäistä konetta, joka saattaa olla osana suurempaa automaatiojärjestelmää, kuten esimerkiksi Siemens PCS7:tä tai Metso DNA:ta. SCADA-työkalussa on yleensä huomattavasti enemmän valmiita ominaisuuksia ja toimintoja kuin opinnäytetyössä käytetyssä HMI-työkalussa. TIA-portaalin myötä päivittynyt WinCC Flexiblen hyvinä puolina voidaan pitää sovellusnäkyä, joka on samanlainen kaikille pienistä paneelista SCADA-järjestelmään (TIA Portal Suunnitelun uusi aikakausi 2012, 3).



Kuva 1. Esimerkki WinCC-valvomosta

2.5 TIA-Portaali

Siemens TIA-Portaali –lyhenne tulee englannin sanoista totally integrated automation, mikä tarkoittaa erillisten ohjelmien integrointia yhden ohjelman alle. TIA-Portaali eroaa edeltäjästään siten, että samalla työkalulla voidaan ohjelmoida logiikat, käyttöliittymät, turvaratkaisut sekä lähitulevaisuudessa määritellä moottorit ja taajusmuuttajat. Integroinnin myötä työkalun käytettävyys on merkittävästi kohentunut. TIA-Portaali pitää sisällään aikaisemmin mainitut Step7 –työkalun ja WinCC-työkalun, joiden käyttöliittymää on kehitetty integroinnin yhteydessä. TIA-Portaalilla on mahdollista ohjelmoida kaikkia S7-sarjan logiikoita sekä uusia S7-1200- ja S7-1500-sarjan logiikoita, joiden ohjelmoiminen vanhalla Step7 –työkalulla ei enään ole mahdollista. (TIA-Portal Suunnitelun uusi aikakausi 2012, 1).

TIA-Portaalin migraatio-ominaisuudella voidaan muuntaa Step7:llä toteutettu projekti TIA-Portaalin projektiksi, joten siirtyminen Step7 –ympäristöstä TIA-Portaaliin on helppoa.

2.6 Standardoidut ohjelmakielet

TIA-Portaali noudattaa automaatio suunnittelustandardia SFS-EN 61131-3, jossa määritellään ohjelmoitavissa logiikoissa käytettävä ohjelmointikieli sekä tekstimuotoisina että graafisina esitysmuotoina. Standardi on suunnattu tuotekehittäjille kuten Siemensille, jotka tekevät ohjelmointikieliä ohjausjärjestelmiinsä. Standardissa määriteltyjä ohjelmointikieliä ovat FBD (Function Block Diagram), LAD (Ladder Diagram), ST (Structure Text), IL (Instruction List) ja SFC (Sequential Function Chart). (SFSedu. Kotisivut. SFS-Käsikirja 631-2, Kuusisto, 2008, 11).

2.6.1 Function block diagram

Function block diagram eli toimintalohko-ohjelmointi rakentuu suorakaiteen muotoisista toiminnoista, joissa toiminta määräytyy suorakaiteen sisällä olevan symbolin mukaan. Tyypillisiä toimintoja ovat erilaiset loogiset toiminnot kuten and ja or sekä enemmän toiminallisuutta sisältävät kiikut, ajastimet ja laskurit. Toimilohko-ohjelmoinnin etuna on se, että toiminta voidaan esittää tiiviissä ja havainnollisessa muodossa. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 121).

2.6.2 Ladder Diagram

Ladder diagram tarkoittaa suomeksi tikapuukaavio- tai relekaavio-ohjelmointikieltä. Ladder Diagram on visuaalinen ohjelmointikieli, jonka ulkoasu on periytynyt releohjauksien piirikaavioista. Ohjelmointikieli koostuu avautuvista ja sulkeutuvista koskettimista, joiden rinnalla toimii erilaisia toimilohkoja: ajastimia, kiikkuja ja laskureita. Tikapuukaavio on hyvä ohjelmoin-

tikieli yksinkertaisten ohjauksien toteutuksessa, koska koodia on helppo tuottaa ja tulkita jälkikäteen. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 119).

2.6.3 Structure text

Structure text eli rakenteellinen teksti on yksi viidestä IEC 61131-3- standardin määrittelemästä ohjelmointikielestä. Ohjelmakieli muistuttaa Pascal- ja C-kieltä ja sitä käytetään tilanteissa, joissa ei voida käyttää graafisia ohjelmointikieliä kuten FBD:ia ja LAD:ia. Structure text -kielellä on hyvin nopeaa tuottaa sovellusta, mutta sen tulkitseminen ja opettelu ovat huomattavasti vaikeampia kuin graafisten ohjelmointikielien. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 123).

2.6.4 Instruction List

Instruction list eli käskylista on ohjelmakieli, joka muistuttaa Assembly-kieltä. Ohjelmointikieli koostuu käskyistä sekä niihin liittyvistä operandeista ja kommentteista. Siemens on kehittänyt oman variaation käskylistaohjelmointikielestä, joka on nimeltään STL eli statement list. (Okkonen, 2012, 10 & 11).

2.6.5 Sequential Function Chart

Sequential function chartia käytetään askeltavien ohjauksien ohjelmointiin eli sekvenssiohjelmointiin. Sekvenssi koostuu askelista, jotka sisältää toiminnan

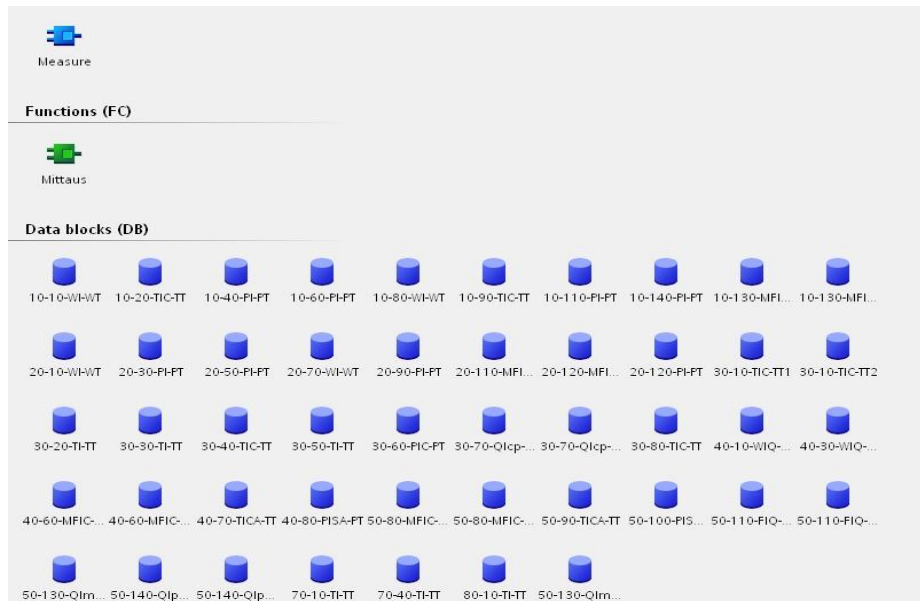
ja siirtoehdon seuraavaan askeleeseen. Sekvenssin suorittama toiminta voidaan ohjelmoida halutulla ohjelmointikielellä. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 123).

3 Toteutus

3.1 Toimilohkot

Tehtävänä oli tuottaa toimilohkokirjasto projekteissa yleisimmin esiintyvälle laitteistoille. Toimilohko on valmis logiikkasovelluksen osa, joka suorittaa tietyn toiminnon, esimerkiksi moottorilähdön ohjaamisen ja valvonnan tai säätoälogitmin (Rajala 2010, 13). Tämä on eduksi tilanteissa, joissa halutaan luoda ohjelmarunko esimerkiksi 30 samanlaiselle toiminnalle. Työmäärä olisi valtava tilanteessa, jossa jokaiselle toiminnalle tulisi ohjelmoida erikseen oma ohjelma. Ohjelmointivirheiden korjaus- ja muutostyöt ovat helppoja toteuttaa toimilohkoihin, joissa tehdyt muutokset siirtyvät automaattisesti siitä generoituihin datablockeihin. Tämä itsestään tuo huomattavaa etua erikseen ohjelmoituun malliin. Toimilohkokirjasto tulee tehostamaan sovellussuunnittelua tulevaisuudessa, jolloin vältytään toimilohkojen valmistamiselta ja työaika voidaan keskittää projektin toteuttamiseen.

Toimilohkoihin tuli sisällyttää mahdollisimman kattavasti ominaisuuksia, joiden ohjenuorana toimi toimeksiantajan laatima lista. Toimeksiantajan laatiman listan lisäksi tarkastelun kohteena olivat automaatiojärjestelmissä esiintyvät toimilohkot ja niiden ominaisuudet. Automaatiojärjestelmissä käytettyjä ratkaisuja sovellettiin PLC-toimilohkoihin tarpeen ja toteutettavuuden mukaan.



KUVIO 1. Mittaustoimilohkosta generoituja datablockeja

3.1.1 Hälytykset

Toimilohkoihin haluttiin sisältää automaattisesti generoituvat hälytysviestit, joiden avulla perustavan laatuisen hälytyksen, kuten esimerkiksi mittauksen ylärajan hälytysviesti, syntyisivät automaattisesti datablockin generoitumisen yhteydessä. Automaattisesti generoituvien hälytysviestien myötä suunnittelijalta säästyy huomattavasti aikaa, sillä rutiiniin omaisten hälytysviestien tekeminen jää kokonaan pois.

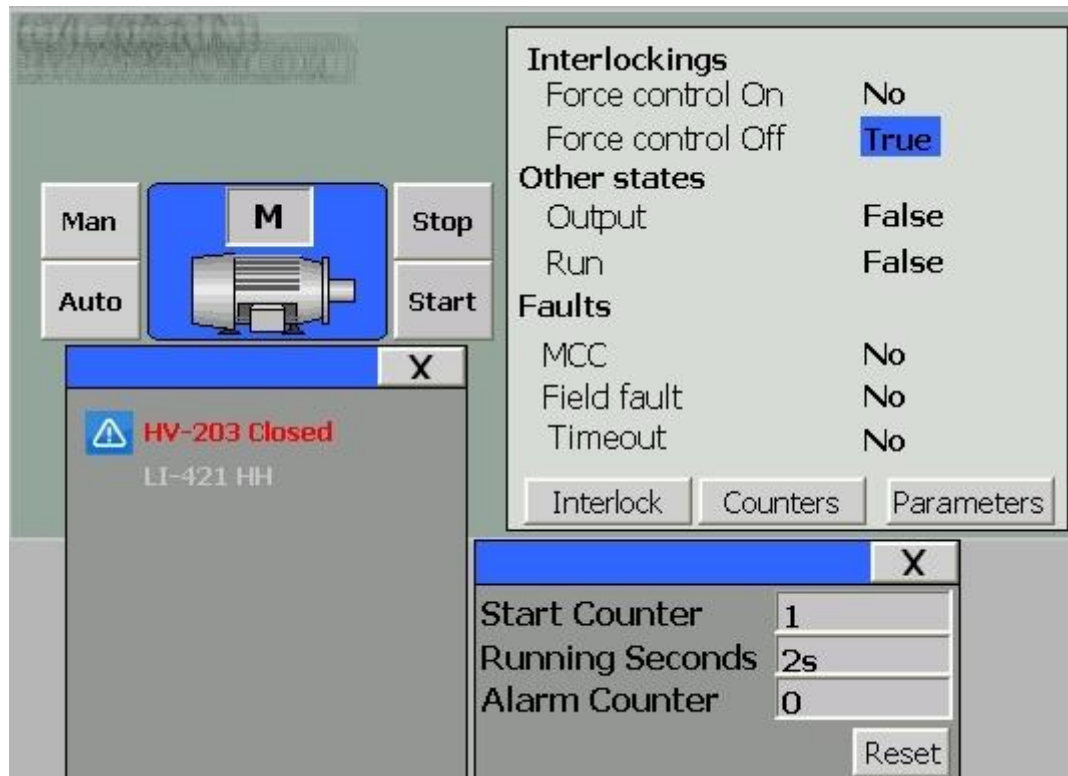
Hälytysviestit saatiin toteutettua Siemensin ohjelmakirjastosta löytyvän alarm_D- tai alarm_DQ-toimilohkojen avulla. Yhtä hälytystoimilohkoa käytti hyväkseen useampi hälytystilanne, jonka avulla hälytyslistassa määriteltäviä tietoja saatiin vähennettyä. Yhdessä hälytysviestissä saatiin ilmaistua mittauk-

sen HH, H, L, LL, oikosulku ja viallinen virtapiiri. Kuuden erillisen hälytysviestin sijasta saatiin kaikki tiivistettyä yhden hälytysviestin alle. Mittauspiirin position generoiminen hälytysviestiin ei onnistunut automaattisesti, vaan positio syötetään manuaalisesti hälytyslistaan. Asian selvittämisessä tukeuduttiin Siemensin tekniseen tukeen. Vastauksessaan he kertoivat, että instanssi DB:n lisääminen AlarmS/D-ilmoitukseen ei vielä toistaiseksi onnistu TIA V11:lla.

3.1.2 Lukitukset

Hyvän ohjelmointitavan mukaisesti toimilohkoille tuli toteuttaa myös lukitusmahdollisuudet. Lukituksilla pyritään estämään laitteen toiminta tilanteissa, jossa se voi aiheuttaa vahinkoa ympäristölleen tai itselleen: moottorin kiviakäynti tai säiliön ylitäyttö estetään lukitusten avulla. Lukitukset toteutetaan vain laitteille, joita voidaan ohjata. Esimerkiksi mittauslohko suorittaa vain mittauksia, eikä sitä voida ohjata valvomosta käsin, joten sille ei valmisteta lukituksia.

Laitteet voidaan pääsääntöisesti lukita auki, kiinni, käyntiin tai seistilaan. Lisäksi parametroidin avulla voidaan määrittää mihin tilaan laite jätetään lukituksen jälkeen. Lukittaviin toimilohkoihin sisällytettiin siepparilohko, jonka tehtävä on ilmoittaa mikä lukituksista kytkeytyi ensimmäisenä päälle: tämä helpottaa vian paikantamisessa tilanteessa, jossa lukituksia on paljon ja halutaan selvittää lukituksen aiheuttaja nopeasti. Ensimmäinen lukitus ilmaistaan sinisellä huomiomerkillä ja päällä olevat lukitukset korostetaan punaisella tekstillä (Ks. Kuvio2).



KUVIO 2. Moottorin laajennettu piirinäyttö

3.2 Moottori

3.2.1 Suoraohjaus

Moottorin suora ohjaus onnistuu pelkän moottoritoimilohkon avulla. Moottoria voidaan ohjata manuaalisesti operointipanelista tai automaattitoiminnon avulla esimerkiksi sekvenssistä. Toimilohkoon on sisällytetty automaattitoiminnaisuus, joka asettaa lähdön pysähdystilaan, jos virheitä ilmenee moottorilähdön toiminnassa. Näiden toimintojen avulla estetään moottorin yllättävä käynnistyminen virhetilanteen jälkeen, mikä lisää laitteen turvallisuutta.

3.2.2 Taajuusmuuttajaohjaus

Taajuusmuuttajaohjaus tapahtuu kahden erillisen toimilohkon ja näyttösymbolin myötä. Toinen toimilohkoista on moottoritoimilohko, jota käytetään suoraohjauksessa. Toista toimilohkoista käytetään taajuuden asetuksessa. Taajuustoimilohko perii lukitustiedot moottoritoimilohkolta, minkä avulla vältetään lukitusten uudelleen määrittelyltä. Lukitustilanteessa taajuustoimilohko siirtää FCNTR-arvon taajuusmuuttajalle. Taajuudenasetustoimilohkon manuaali- ja auto-ohjaus tapahtuvat vastaavanlaisesti kuin moottoritoimilohkonkin.

3.2.3 Hälytykset

Moottoritoimilohkolla on seuraavanlaisia virhetilanteita: moottorin paluutietovika, moottorikeskusvika ja turvakytkinvika. Paluutietovika laukeaa päälle, mikäli paluutietoa ei tule parametroidun ajan sisällä ohjauksen jälkeen. Moottorikeskus- ja turvakytkinvika ilmenevät, jos paluutietoa ei saada kyseiseltä laitteelta. Vikatilanteet voidaan parametroida pois käytöstä mikäli turvakytkin-, moottorikeskus- tai paluutietoa ei ole saatavilla. Edellä mainitut vikatilanteet ilmaistaan HMI-laitteen hälytysviestilistassa, jonne viestit generoituvat automaattisesti.

3.3 On/off -venttiili

3.3.1 Ohjaus

Venttiilitoimilohkon manuaali- ja auto-ohjaus toimii samankaltaisesti kuin moottoritoimilohkonkin. Mikäli rajatietoja ei saada auki tai kiinni ohjauksen

jälkeen parametroidun ajan kuluessa, venttiili asetetaan parametrien mukaan auki tai kiinni. Samaa toimilohkoa voidaan käyttää NO- ja NC-tyyppisten venttiilien ohjauksissa. On myös mahdollista, että venttiilillä ei ole olemassa rajakytkimiä, jonka takia venttiilille valmistetaan simulointi simuloimaan rajatietoja. Simulaatio voidaan parametroida käyttöön venttiilitoimilohkossa.

3.3.2 Hälytykset

Venttiilille on vakioitu paluutietohälytys, joka tarkoittaa tilaa, jossa venttiili ei ole saavuttanut auki- tai kiinnirajaa ohjauksen jälkeen hälytysajan kuluttua. Hälytysaika on parametriarvo, joka tarkoittaa sitä, kuinka kauan venttiili odottaa toimielimen ilmaantumista rajakytkimelle ennen hälytyksen antamista.

3.4 Säädin

Säädin toimii yhteistyössä mittauslohkon kanssa ja säätimelle kuuluva mittauslohko osoitetaan parametriarvona säätimellä. Mittauslohko osoitetaan datablockin numeroinnin mukaan esimerkiksi DB7. Tämän avulla säästetään aikaa konfiguroinnissa, kun kaikkia tietoja ei tarvitse osoittaa erikseen säätimelle.

3.4.1 Ohjaus

Säädintä voidaan ohjata manuaali- ja auto-tilassa. Manuaalitulassa voidaan valvomosta käsin asettaa säätimen ohjausarvoksi esimerkiksi ”venttiilin avauma 70%”. Automaattitulassa voidaan asettaa säätimelle tavoitearvo eli setpoint, johon kyseistä mittausarvoa tavoitellaan. Esimerkkitapauksessa virtaus voidaan asettaa 100 litraa minuuttia kohden, jolloin säädin alkaa etsiä avautumaa, jossa tavoiteltu virtaus saavutetaan. Säätimelle voidaan asettaa kahdenlaista setpointeja: local setpoint, joka mahdollistaa setpointin asettamisen manuaalisesti käyttöliittymästä tai remote setpoint, joka saadaan automaattitoiminnolta, kuten sekvenssiltä tai laskentalohkolta. Lähtevää ohjausta on mahdollista rampittaa prosenttia sekuntia kohden –periaattella, jonka avulla saadaan vaimennettua nopeasti tapahtuvat muutokset säätimen lähdessä.

3.4.2 Hälytykset

Säätölohko toimii yhdessä mittauslohkon kanssa, joten mittauksessa ilmenevät hälytykset siirtyvät säätimelle. Mittauksen ja säätimen laajennetussa piiri-

näytössä näkyvät samat hälytykset, mutta ne ovat mahdollista kuitata vain mittauksen piirinäytöltä tai hälytyslistalta.

3.4.3 Lukitukset

Säätimellä on olemassa kaksi erillistä lukitusnastaa, joista aiheutuvat erilaiset toiminnat. Ensimmäinen lukitusnasta asettaa säätimen manuaalitilaan ja tämän lisäksi ottaa käyttöönsä FMAN-parametriin syötetyön arvon. Tämä on hyödyllinen tilanteissa, joissa halutaan nopeasti reagoida lukitukseen. Toinen käytössä oleva lukituksista asettaa säätimen automaatille ja asettaa setpointiksi FAUT-arvon, jonka jälkeen säädin aloittaa tavoitearvoon siirtymisen.

3.5 Analogiamittaus

Mittauslohko on hyvin laaja ja mukautuva toimilohko. Mittauslohko on laajasti parametroitavissa, kuten esimerkiksi mitta-alueen skaalaus, käytettävissä olevat hälytykset, mittauksen suodatus ja suodatuksen vahvuus. Mittauksen suodatus on erittäin käytännöllinen tilanteissa, jossa mittausarvo heittelee vaikeuttaen näin mittauksen tulkintaa. Käytännössä suodatus tasoittaa mittausarvoa poistaen siitä tulkintaa väärentävän heittelyn.

3.5.1 Vikatilanne

Mittauslohko osaa ilmoittaa vikatilanteista, joissa mittauspiiri on poikki tai se on oikosulussa. Edellä mainituissa tilanteissa mittauslohko lähettää valvomopäätelle hälytysviestin, jonka voi kuitata joko piirinäytöstä tai hälytyslistalta.

Hälytys generoituu automaattisesti, kun mittauslohko otetaan käyttöön, joten sen parametointiin ei kulu automaatio suunnittelijalta aikaa.

3.5.2 Hälytykset

Mittauslohko sisältää viisi erillistä hälytystä: higher high, high, low, lower low ja measurement alarm. Kaikki mitta-arvoon liittyvät hälytykset ovat parametroitavissa päälle tai pois sekä lisäksi niihin liittyvät raja-arvot ovat täysin käyttäjän aseteltavissa myös käyttöliittymästä käsin. Hälytys generoituu automaattisesti toimilohkoa käytettäessä.

3.6 Käyttöliittymä

Käyttöliittymät luovat rajapinnan ihmisen ja koneen välille. Tämän rajapinnan avulla käyttäjä voi monitoroida ja hallita sekä prosessia että konetta. Aikaisemmin rajapinta oli toteutettu ohjauspulpeteilla, joissa oli kytkimiä, peukalopyöriä, merkkilamppuja ja erillaisia osoittimia. (Fonselius, Pekkola, Selosmaa, Ström & Välimaa 1996, 190). Kosketusnäytöt ja PC-valvomot on syrjäyttänyt ohjauspulpetit ohjainlaitteena. Käyttöpaneelien ja PC-valvomoiden etuna ohjauspulpetteihin verrattuna on monitoroinnin selkeys, runsas tiedonmäärä, ohjattavuus, toteutuksen helppous ja muunneltavuus. Opinnäytetyössäni käytettiin WinCC flexible –työkalua, joka on suunniteltu näyttöpaneelien käyttöliittymän tekoon. WINCC flexible –työkalulla toteutettavasta käyttöliittymästä voidaan käyttää nimitystä HMI-käyttöliittymä.

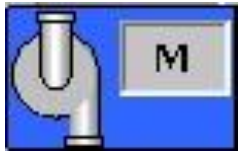
3.6.1 Näyttösymbolit

Toimilohkojen valmistuttua siirryttiin tekemään niitä vastaavia piirinäyttöjä operointipaneelille. Piirinäytöt tuli toteuttaa siten, että yhden osoitteen avulla määritetään kaikki symboliin liittyvät indikoinnit ja ohjaukset. Vaatimuksena oli myös toteuttaa symboleista kolme eri laajuista piirinäyttöä. Suppein piirisymboli osoittaa laitteen tilaa, joka sisältää tiedot käynnistä, viasta, lukituksesta ja ohjauspaikasta (Ks. Kuvio3.). Piirisymboli on laajennettavissa painamalla symbolia sormella tai hiirellä klikkaamalla, jonka jälkeen saadaan operointinäkyvä, jossa monitoroinnin lisäksi on mahdollista ohjata laitetta (Ks. Kuvio4.). Laajennettu piirinäyttö antaa edellä mainittujen ominaisuuksien lisäksi paremetroita kaikki laitteeseen liittyvät ominaisuudet: esimerkkinä mitauksen minimi- ja maksimiarvot (Ks. Kuvio2.). Laajennettu piirinäyttö saadaan avattua piirinäytön param-painikkeesta.

3.6.2 Monitorointi- ja ohjauspiirinäyttö

Suppea ja laajennettu piirinäyttö valmistettiin käyttämällä ohjelman faceplate-ominaisuutta. Faceplate yhdistää valitut yksittäiset objektit yhdeksi kokonaisuudeksi ja mahdollistaa niiden käsittelyn yksittäisenä objektina. Kun symbolin indikoinnit ja ohjauspainikkeet oltiin ryhmitelty yhdeksi kokonaisuudeksi, täytyi myös niille menevät tagit ryhmittää myös yhdeksi kokonaisuudeksi. Datatyyppin sai toteutettua ohjelmiston omalla työkalulla, jonne listattiin kaikki faceplatessa esiintyvät tagit ja niille kuuluvat tietotyypit. Datatypen valmistuttua se otettiin käyttöön faceplatessa, jonka jälkeen faceplate oli yhdistettävissä toimilohkon avulla luotuun datablockiin.

Faceplaten suurin etu on se, että kertamäärittelyn jälkeen se on käytettävissä aina uudestaan ja uudestaan. Lisäksi se vähentää suunnittelijan työmäärää, sillä sen käyttöönotto ja paremetrointi on yhden osoitteen määrittelyn jälkeen valmis. Lisäksi piirinäytön mukauttaminen jälkikäteen on helppa, koska yhden faceplaten muokkaaminen muokkaa samalla kerralla kaikki vastaavanlaiset faceplatet samanlaisiksi.



KUVIO 3. Moottori symboli



KUVIO 4. Moottorin operointinäkyvä

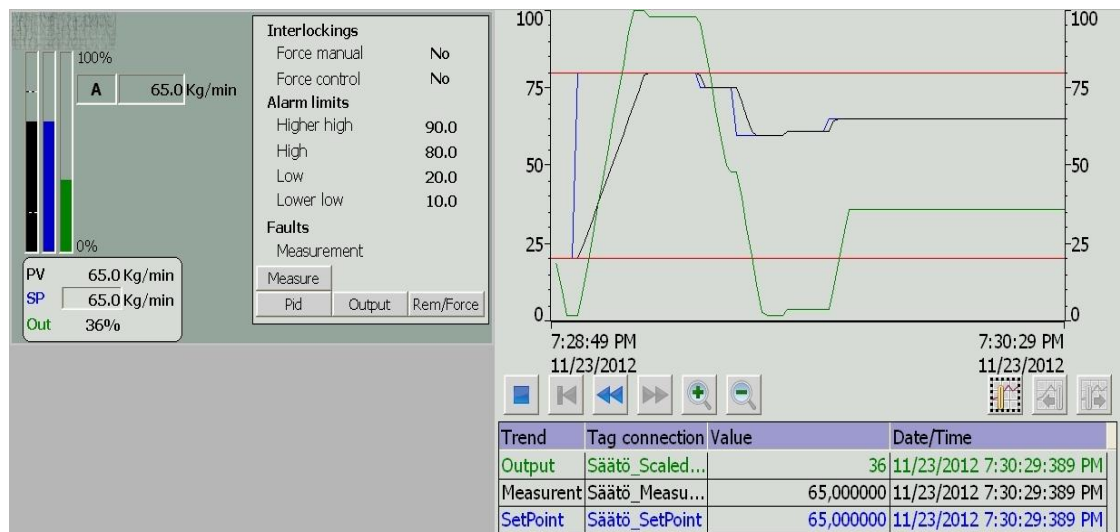
3.6.3 Laajennettu piirinäyttö

Jokaiselle valvomossa käytetylle piirinäytölle valmistettiin sitä vastaava laajennettu piirinäyttö. Laajennetussa piirinäytössä on mahdollista monitoroida ja ohjata haluttua laitetta, lisäksi myös laitteiden parametointi onnistuu laajennetusta piirinäytöstä käsin. Tavallisia parametroitavia tietoja on mm. käytössä olevat hälytykset, hälytysrajat, mittauksen ylä- ja alaraja sekä häiriönsuodatuksen vahvuus. Parametointi vaatii, että käyttöliittymään kirjautuneella käyttäjällä on oikeudet muuttaa parametriarvoja.

Koska käyttöliittymän ja logiikan välistä tiedonvaihtoa on yhdessä laajennetussa piirinäytössä erittäin runsaasti, ei ole enää järkevää tai tehokasta käyttää toimilohkoon sidottua piirinäyttöä kuten suppeassa piirinäytössä liiallisen tiedonvaihdon takia. Suunnitelmana oli käyttää yhtä yhteistä laajennettua piirinäyttöä kaikkien vastaavanlaisien toimilohkojen kesken. Tämä onnistuu käyttämällä multiplexing-ominaisuutta. Multiplexingiä hyväksikäyttäen luettavan datablockin numeroa voidaan vaihtaa käyttöliittymästä käsin. Suppeaan piirinäyttöön on lisätty param-painike, jota painamalla avautuu laajennettu piirinäyttö ja luettavan datablockin numero vaihtuu suppean piirinäytön mukaiseksi. Valinta olisi mahdollista tehdä myös tekstilistalta, josta valitsemalla avautuu position tai nimen mukainen laajennettu piirinäyttö.

Käyttämällä multiplexingiä powertageja säästyy huomattava määrä. Power-tag tarkoittaa käyttöliittymän ja logiikan välillä siirettävää tietoa. Yksi siirettävä bitti kuluttaa yhtä paljon powertageja kuin yksi siirettävä word. Tämän takia yksittäiset bitit on järkevää pakata toimilohkoissa yhden byten tai wordin sisälle.

Laajennettu piirinäyttö on uudelleen käytettävissä ilman erillisiä määrittelyjä, mutta uudelleen käyttö vaatii sitä vastaavan symbolipöydän kopioimisen yhdessä näyttöikkunan kanssa. Laajennettu piirinäyttö tekee automaatiosovelluksesta entistä joustavamman, koska PLC voidaan parametroida ilman erillistä tietokonetta. Tämä lisää tehokkuutta prosessin käyttöönottovaiheessa ja käytön aikana.



KUVIO 5. Säätimen laajennettu piirinäyttö

4 Tulokset

4.1 Saavutettu etu

Kehitystyön tuloksena syntynyt toimilohkokirjasto otettiin välittömästi käyttöön valmistumisen jälkeen. Kirjaston käyttöönottoa helpotti käyttöohjeet, joka kertoi suunnittelijalle tärkeää tietoa toimilohkoille ja symboleille määriteltävistä asoista.

Täysin saumattomasti toimivat toimilohkot ja symbolit vähensivät automaattiosovelluksen valmistukseen kuluvaan aikaan merkittävästi. Symbolin ja toimilohkon välinen kommunikointi järjestetään yhden osoitteen avulla, minkä johdosta määrittelyyn kuuluva aika saatiin minimoitua. Automaattisesti generoituvat hälytysviestit osoittivat hyödyllisyytensä ensimmäisessä projektissa, jossa välttyttiin sadoilta hälytysviestien teolta. Suunnittelun tehokkuutta lisäsi myös laajennetut piirinäytöt, jonka johdosta symboleille ei tarvinnut tehdä enään erillisiä lukitus- ja parametointi-ikkunoita.

Symbolien uudistuneet piirinäytöt tuovat merkittävästi lisää visuaalisuutta ja informaatiota operoivalle käyttäjälle: edustavan visuaalisen ilmeensä takia tuotetta on helppoa markkinoida asiakkaalle. Prosessin tai koneen hallittavuus kohenee myös uudistuksien myötä. Kehitystyö on tuonut toimeksiantajalle lisää kilpailukykyä ja mahdollisuuden myös tuottaa asiakkaalle hyvin informatiivinen käyttöliittymä mahdollisimman kustannustehokkaasti.

4.2 Kehityskohteet

Toimilohkokirjaston luominen ja kehitystyö on ollut erittäin haastava, mutta opettavainen kokemus. Kovan työn ja tutkimisen jälkeen sain luotua ohjelmistokirjaston, joka pitää sisällään projekteissa yleisimmin käytettävät toimilohkot. Vaikka ohjelmistokirjaston luomiseen käytettiin huomattavasti aikaa, jäi siihen myös kehittämisen varaa. Opinnäytetyön jälkeen hankitun ammatillisen kokemuksen myötä tulee enemmän näkemystä siihen, kuinka toimilohkoja olisi hyvä kehittää ja parantaa jatkossa.

Toimilohko- ja symbolikirjasto otettiin nopeasti käyttöön valmistumisen jälkeen, minkä johdosta kehityskohteita ilmenee käyttökokemusten perusteella. Symbolien kehitystyötä on hyvä suorittaa jatkossa kuuntelemalla operaattoreiden mielipiteitä käyttöliittymän symboleista ja niiden toiminnasta. Operaattorit näkevät symboleiden puutteet ja osaavat siten antaa rakentavia symboleiden kehitysideoita, sillä he työskentelevät käyttöliittymän kanssa lähes päivittäin.

Tulevaisuuden kehityskohteita kirjastolle tulee olemaan toimilohkokirjaston laajentaminen, joka tapahtuu vähitellen, kun tarpeen tullen luodut toimilohkot siirretään valmistumisen jälkeen kirjastoon uusiokäytettäväksi. Lisäksi tarkoituksena on olemassa olevien toimilohkojen ominaisuuksien lisääminen ja piirinäyttöjen mukauttaminen käyttäjäkokemusten pohjalta.

5 Pohdinta

Opinnäytetyöprojekti on ollut ammatikorkeakoulun merkittävin vaihe oppimisen kannalta. Oppia on tullut paljon käytännön sekä teorian saralla. Käytännön taitoja on erityisesti kertynyt TIA-Portaalin käytön myötä, minkä johdosta ohjelmalla toteutettavat projektit onnistuvat jo hyvin itsenäisesti. Opinnäytetyön alkuvaiheissa TIA-Portaali oli minulle täysin tuntematon työkalu, voisin todeta, että asetetut tavoitteet saavutettiin ohjelman käytön osalta. TIA-Portaalin osaaminen on erittäin arvokas taito: sen käyttö on nykyään vahvassa nousussa ja sen osaamista arvostetaan.

Teoreettista osaamista kehitti tutustuminen automaatiojärjestelmissä esiintyviin toimilohkoihin sekä niiden ominaisuuksiin ja käyttötilanteisiin. Haastavin vaihe opinnäytetyössä oli määrittää tarvittavat ominaisuudet ja toiminnot ohjelmoitavalle logiikalle toteutettaville toimilohkoille. Määrittelyn teki haastavaksi oma kokemattomuuteni sovellussuunnittelualalla: oli erittäin vaikea kehittää tuotetta, josta ei ole aikaisempaa käyttökokemusta tai mielikuvaa käyttötilanteista. Kaikesta huolimatta opinnäytetyölle asetetut toiminnalliset ja henkiset tavoitteet saavutettiin erinomaisesti.

Opinnäytetyön aikana opitut ammattimaiset työskentelytavat, tiedot ja käytännöt ovat antaneet minulle merkittävästi valmiuksia tulevalle insinöörin uralleni, johon suhtaudun hyvin luottavaisin ja odottavaisin mielin.

Lähteet

Fonselius, J., Pekkola, K., Selosmaa, S., Ström, M., & Välimaa, T. 1996. Auto-
maatiolaitteet. 1. p. Helsinki: Edita. Viitattu 26.11.2012.

JEEC Oy. Kotisivut. 2010. Viitattu 2.12.2012. <http://jeec.fi/>

Keinänen, T., Kärkinen, P., Metso, T. & Putkonen, K. 2001. Logiikat ja ohjaus-
järjestelmät. 1. p. Helsinki: Wsoy. Viitattu 25.11.2012.

Kuusisto, H. 2008. Ohjelmoitavaiden logiikoiden kartoitus. Viitattu 31.1.2013
<https://publications.theseus.fi/bitstream/handle/10024/735/Kuusisto%20Ari.pdf?sequence=1>

Ohjelmoitavat logiikat simatic S5. Siemens Oy kotisivut. 2013. Viitattu
13.1.2013.
http://www.siemens.fi/fi/industry/teollisuuden_tuotteet_ja_ratkaisut/tuotesivut/automaatiotekniikka/ohjelmoitavat_logiikat_simatic/s5.htm

Okkonen, Ville. 2012. Lämpölaitoksen ohjausjärjestelmän modernisointi. Vii-
tattu 31.1.2013.
https://publications.theseus.fi/bitstream/handle/10024/43691/Okkonen_Ville.pdf?sequence=1

Rajala-Rahko, V. 2010. Toimilohkokirjasto kattilalaitoksen ohjaamiseen. Viitat-
tu 15.11.2012.
http://publications.theseus.fi/bitstream/handle/10024/16766/Rajala-Rahko_Ville.pdf?sequence=1

SFSedu. Kotisivut 2013. SFS-Käsikirja 631-2. Viitattu 31.1.2013
<http://www.sfsedu.fi/www/fi/oppilaitoskasikirjat/Automaatio/631-2.php>

Siemensin tuote tuki. 2012. Instanssi DB:n nimen lisäys AlarmS/D ilmoituk-
seen. Sähköpostiviesti 14.9.2012. Vastaanottaja J. Laajisto. Siemensin tuki-
pyyntöön vataus.

TIA Portal suunnitelun uusi aikakausi. Siemens TIA-portal esite. 2012. Viitattu
25.11.2012.
http://www.siemens.fi/pool/products/industry/iadt_is/tuotteet/tia_portal/tia_portal_esite.pdf

Tolonen, J. 2010. Symbolikirjastojen luonti WinCC -ympäristöön. Viitattu 2.12.2012.

https://publications.theseus.fi/bitstream/handle/10024/21109/Jaakko_Tolonen.pdf?sequence=1

Wallenius,H. 2012. Siemens TIA-Portal V11 logiikkaohjelmoinnin perusteet. Viitattu 19.1.2013.

http://publications.theseus.fi/bitstream/handle/10024/48033/Wallenius_Henri.pdf?sequence=1

Liitteet

Liite 1. Venttiilin laajennettu piirinäyttö

Interlockings	Value
Force control Open	No
Force control Close	No

Other states	Value
Limit Switches	Open & Close
Output	True
Open limit	True
Close limit	False

Faults	Value
Timeout	No

Buttons: Counters, Parameters

Liite 2. Mittauksen laajennettu piirinäyttö

Alarm limits	Value
Higher high	90.0
High	80.0
Low	20.0
Lower low	10.0

Faults	Value
Measurement	
Fault	
Hysteresis	0.0
Filter	0

Trend	Tag connection	Value	Date/Time
Measurent	Scaled Value	#####	11/27/2012 3:48:40:359 PM