

Fidelix-ohjelmointiopas

Vili Korhonen

Opinnäytetyö
Marraskuu 2013

Automaatiotekniikan koulutusohjelma
Teknologiayksikkö



Tekijä(t) Korhonen, Vili	Julkaisun laji Opinnäytetyö	Päivämäärä 20.11.2013
	Sivumäärä 53	Julkaisun kieli Suomi
	Luottamuksellisuus () saakka	Verkkojulkaisulupa myönnetty (X)
Työn nimi Fidelix-automaatiojärjestelmän ohjelmointiopas		
Koulutusohjelma Automaatiotekniikan koulutusohjelma		
Työn ohjaaja(t) Ström, Markku ja Flyktman, Teppo		
Toimeksiantaja(t) LVI-Elektro Graf Oy		
Tiivistelmä <p>Opinnäytetyön aiheena on Fidelix-automaatiojärjestelmän ohjelmointioppaan luominen. Työn tilaajana toimii LVI-Elektro Oy. Yrityksen ydinosaminen keskittyy rakennusautomaatioon, sekä energi-anhallintaan automatiikan avulla.</p> <p>Kyseisessä oppaassa käydään läpi, kuinka Fidelix-automaatiojärjestelmään tehdään toimiva logiikka alusta alkaen. Työ sisältää pisteiden luomisen PointGen-työkalulla, grafiikkakuvien tekemisen FxHtmlEditorilla, logiikan ohjelmoimisen OpenPCS-ohjelmointityökalulla ja FX-2025A-säätimen lopullisen ohjelmoinnin.</p> <p>Kun työ on lopullisesti valmis, käytetään sitä yritykseen palkattujen ohjelmoijien koulutukseen/täydennyskoulutukseen sekä ulkopuolelta tilattavien koulutusten pitämiseen. Opinnäytetyötä testattiin koehenkilöllä yrityksestä. Koehenkilö oli aikaisemmin ollut tekemisissä rakennusautomaation kanssa, mutta ei ollut perehtynyt Fidelixin laitteisiin.</p> <p>Opinnäytetyötä tullaan kehittämään koehenkilöltä saadun palautteen perusteella, sekä tulevien koulutustilaisuuksista kerätyn palautteen mukaan. Ohjelmistoversioihin tulee jatkuvasti lisää päivityksiä ja opinnäytetyö on tarkoitus myös niiden osalta pitää ajan tasalla.</p>		
Avainsanat (asiasanat) Rakennusautomaatio, Fidelix, OpenPCS, FxHtmlEditor, Fx2025A		
Muut tiedot		



Author(s) KORHONEN, Vili	Type of publication Bachelor's Thesis	Date 20.11.2013
	Pages 53	Language Finnish
	Confidential () Until	Permission for web publication (X)
Title Programming manual for Fidelix automationsystem		
Degree Programme Degree programme of Automation engineering		
Tutor(s) Ström, Markku and Flyktman, Teppo		
Assigned by LVI-Elektro Oy		
Abstract <p>The subject of this thesis is the creation of a programming manual for Fidelix automationsystem. This manual was ordered by LVI-Elektro Oy. Main business of LVI-Elektro Oy is building automation and energy management by automation solutions.</p> <p>This manual will instruct the user how to create a working PLC for the Fidelix automationcontroller. The manual based on the thesis includes the creation of points by PointGen and the creation of graphic pictures by FxHtmlEditor. The manual based on the thesis will also include programming by OpenPCS-software and the final programming of the Fx-2025A controller.</p> <p>When the thesis work is done, the manual based on it will be used to educate and to train new programmers for the company, as well as to provide on-demands training to other companies. The solution based on the thesis were tested with a junior programmer of the company. The programmer has previously been involved in building automation but was not familiar with Fidelix-controllers.</p> <p>The thesis will be developed based on the feedback from the programmer, as well as from the future training sessions. The software versions from Fidelix-controllers and programming software will be constantly upgraded. The manual based on the thesis will also have upgrades every time there are some major upgrades from Fidelix.</p>		
Keywords Building automation, Fidelix, OpenPCS, FxHtmlEditor, Fx2025A		
Miscellaneous		

SISÄLTÖ

1	JOHDANTO	5
1.1	Toimeksiantaja.....	5
1.2	Projektin tavoitteet.....	5
1.3	Projektin vaatimukset.....	5
2	TEORIA.....	6
2.1	Ohjelmointi	6
2.2	IEC 61131-3-standardi	6
2.2.1	Datatyypit.....	6
2.2.2	Ohjelmointikielet.....	7
3	AUTOMAATIOJÄRJESTELMÄN OHJELMOINTI	9
3.1	Projektin aloitus.....	9
3.2	Pisteiden luominen, PointGen.....	9
3.2.1	PointGen, johdanto	9
3.2.2	Pisteiden luominen PointListillä	9
3.2.3	PointList.....	10
3.2.4	Pisteiden lataaminen säätimelle	12
3.3	Grafiikkakuvien luominen, Fidelix HtmlEditor.....	12
3.3.1	Päänäkymä	12
3.3.2	Uusi tiedosto/Avaa tiedosto/Tallenna tiedosto-valikko	13
3.3.3	Alkio-valikko	13
3.4	OpenPCS-ohjelmointityökalu	21
3.4.1	Yleistä	21
3.4.2	Uuden projektin luominen	21
3.4.3	Ohjelmointi.....	26

3.4.4	Ohjelman lataaminen säätimeen	31
3.5	Säädin FX2025A	34
3.5.1	Aloitussivu	34
3.5.2	Grafiikkakuvat	36
3.5.3	Käyttäjätila.....	37
3.5.4	Ohjelmointi.....	39
3.5.5	Historia	50
4	POHDINTA	52
4.1	Tulokset	52
4.1.1	Kehitettävät osa-alueet.....	53
4.1.2	Työn onnistuminen	54
	LÄHTEET.....	54

KUVIOT

Kuvio 1.	PointTable.....	10
Kuvio 2.	PointList, Hälytykset	11
Kuvio 3.	PointList, Digitaalitulot	11
Kuvio 4.	PointList, Ohjaukset.....	11
Kuvio 5.	PointList, Mittaukset	11
Kuvio 6.	PointList, Säädot.....	11
Kuvio 7.	PointList, Asetusarvot.....	11
Kuvio 8.	PointList, Aikaohjelmat.....	11
Kuvio 9.	Fidelix HTML editor	12
Kuvio 10.	FxHTMLEditor, Uusi Tiedosto/Avaa tiedosto/Tallenna tiedosto	13
Kuvio 11.	FxHTMLEditor, Alkiot.....	14
Kuvio 12.	FxHTMLEditor, Pistetunnus	15
Kuvio 13.	FxHTMLEditor, Numerokenttä	16

Kuvio 14. FxHTMLEditor, Pylväsnäyttö	17
Kuvio 15. FxHTMLEditor, Teksti.....	18
Kuvio 16. FxHTMLEditor, Kuva	19
Kuvio 17. FxHTMLEditor, Symboli	20
Kuvio 18. OpenPCS-työkalu	21
Kuvio 19. OpenPCS, uusi projekti	22
Kuvio 20. OpenPCS, Luo projekti.....	23
Kuvio 21. OpenPCS, Kansion luonti	23
Kuvio 22. OpenPCS, Ohjelmaosion luonti	24
Kuvio 23. OpenPCS, Ohjelmaosion näkymä	25
Kuvio 24. OpenPCS, Aktiiviset resurssit.....	25
Kuvio 25. OpenPCS, Muuttujien haku	27
Kuvio 26. OpenPCS, Säästöpuistin haku	28
Kuvio 27. OpenPCS, Var.....	28
Kuvio 28. OpenPCS, IF-lause.....	31
Kuvio 29. OpenPCS, Syntax Check.....	32
Kuvio 30. OpenPCS, Resource Properties	33
Kuvio 31. OpenPCS, Connections	33
Kuvio 32. OpenPCS, TCP Settings	33
Kuvio 33. OpenPCS, Online.....	34
Kuvio 34. Fidelix, Aloitusnäkymä	35
Kuvio 35. Fidelix, Ylempi alavetovalikko	36
Kuvio 36. Fidelix, Grafiikkakuva.....	37
Kuvio 37. Fidelix, Indikoinnit	38
Kuvio 38. Fidelix, Moduulit.....	38
Kuvio 39. Fidelix, Tapahtumaloki	38
Kuvio 40. Fidelix, Digitaalitulon ohjelmointi	40
Kuvio 41. Fidelix, Ohjauksen ohjelmointi	41
Kuvio 42. Fidelix, Analogialähdön ohjelmointi	41
Kuvio 43. Fidelix, Mittauksien ohjelmointi.....	42
Kuvio 44. Fidelix, Muunnostaulukoiden ohjelmointi	43

Kuvio 45. Fidelix, Aikaohjelminen ohjelmointi	44
Kuvio 46. Fidelix, Hälytyspisteen ohjelmointi 1	45
Kuvio 47. Fidelix, Hälytyspisteen ohjelmointi 2	45
Kuvio 48. Fidelix, Hälytyspisteen ohjelmointi, Rajahälytys	46
Kuvio 49. Fidelix, Hälytyspisteen ohjelmointi, Ristiriitahälytys.....	46
Kuvio 50. Fidelix, Hälytyspisteen ohjelmointi, Modulivika	47
Kuvio 51. Fidelix, Hälytysryhmät	47
Kuvio 52. Fidelix, Hälytyslaitteet	48
Kuvio 53. Fidelix, Tilatekstit.....	49
Kuvio 54. Fidelix, Modulien ohjelmointi	49
Kuvio 55. Fidelix, Historia	50
Kuvio 56. Fidelix, Historiaryhmät	51
Kuvio 57. Fidelix, Historian piirto	52

1 JOHDANTO

1.1 Toimeksiantaja

LVI-Elektro Oy on Jyväskylässä toimiva yritys, joka perustettiin 1983. Yrityksen päätoimialana on rakennusautomaatio ja siihen liittyvät järjestelmät. Yritys myös markkinoi €LVIS-energiansäästöprojekteja, joissa tarkoitus on pienentää rakennusautomaatoratkaisuilla energian kulutusta.

Vuonna 2012 yrityksen liikevaihto oli noin 682 000 euroa. Vuonna 2013 yrityksessä työskenteli seitsemän henkilöä, joista kaksi ohjelmoinnin parissa.

1.2 Projektin tavoitteet

Projektin tarkoitus on tuottaa yrityksen tekemän Fidelix-koulutuksen tueksi ohjelmointiopas. Ohjelmointioppaan päätarkoitus on toimia uusille ohjelmoijille oppaana, josta he voivat tarkistaa ja palauttaa muistiinsa koulutuksen jälkeen unohtuneita asioita. Opas toimii myös vanhemmille ohjelmoijille täydennyskoulutuksena.

1.3 Projektin vaatimukset

Projektissa luotava opas testataan yrityksen uuden ohjelmoijan koulutuksen yhteydessä. Opas luovutetaan koulutettavalle nopean koulutuksen jälkeen ja ohjelmoijan täytyy pystyä läpiviemään projektin ohjelmointiosuuden ilman vaikeuksia. Samalla löydetään oppaan puutteet, jotka täydennetään seuraavaan opasversioon.

2 TEORIA

2.1 Ohjelmointi

Suurin osa nykyään käytettävistä ohjelmista on valmiita ohjelmia, kuten kuvankäsittelyohjelma Adobe Photoshop ja tekstinkäsittelyohjelma Microsoft Word. Ohjelmointia tarvitaan laajentamaan jo valmiiksi olemassa olevia ohjelmia tai luomaan täysin uusia. Esimerkiksi Microsoft Word sisältää ohjelmointimahdollisuuden, jolla voi tehdä ohjelmasta itselleen soveltuvan. Ohjelmoija voi esimerkiksi ratkaista jonkin vaikeammankin ongelman tekemällä siitä ohjelman, johon hän syöttää muuttujat ja saa vastauksen. Kun ohjelmoija on kerran tehnyt ohjelman valmiiksi, voi hän käyttää tätä samaa ohjelmaa useampaan kertaan. Vähäisin muutoksin ohjelmaa voi käyttää myös samantyyppisten ongelmien ratkaisuun. (Peltomäki, 1999.)

2.2 IEC 61131-3-standardi

1990-luvulla International Electrotechnical Commission (IEC) alkoi kehittää IEC 61131-standardia ohjelmoitaville logiikoille. IEC 61131-3 on tuon standardin kolmas osa, joka keskittyy ohjelmoitavien logiikoiden ohjelmointiympäristöön ja ohjelmointikieliin. (InTech, 2012.) IEC 61131-3 on Euroopassa yleisesti käytetty standardi. Nykyään standardi on leviämässä ja ottamassa jalansijaa myös muualla maailmassa. Standardi määrittelee ohjelmoitaville logiikoille ohjelmointikielien sekä käytettävät datatyypit.

2.2.1 Datatyypit

IEC 61131-3 standardi määrittelee ohjelmoitaville logiikoille datatyypit. Standardissa määritetyt datan tyypit ovat boolean (totuusarvo), integer (kokonaisluku), real (reaaliluku), byte (bitti), word (sana), date (päivämäärä), time-of-day (aika) ja string

(merkkijono). Datan tyyppejä määritetään ainoastaan ohjelmassa käytettäville muuttujille. Näille muuttujille voi määritellä useita tyyppejä. Tällä ennaltaehkäistään virheitä. Esimerkiksi päivämäärän jakamista kokonaisluvulla laskutoimituksessa ei sallita. (PLCopen, 2013.)

2.2.2 Ohjelmointikielet

IEC 61131-3-standardissa määritellään syntaksi, eli lauseoppi, viidelle standardiin sisällytylle ohjelmointikielelle. Tämä tarjoaa ohjelmoijalle liikkumavaraa ohjelmointikielien välillä, koska erilaiset ohjelmat ja ongelmat voivat tarvita erilaisia ohjelmointikieliä. (Real Time Automation, 2009.)

2.2.2.1 *Structured Text (ST)*

Structured Text, eli strukturoitu teksti, on nimensä mukaisesti tekstipohjainen ohjelmointikieli. Ohjelma muistuttaa hyvin läheisesti Pascal- ja C-kieltä, ja näistä tutut elementit, kuten IF-ELSE- ja CASE OF -lauseet ovat tärkeimpiä käskyjä myös Structure Textissä. (PLCopen, 2013.) Structured Textin avulla pystytään luomaan monimutkaisimmat ohjelmat viidestä IEC 61131-3-standardin kielestä, ja sitä käytetään myös Fidelix-säätimen ohjelmoinnissa.

2.2.2.2 *Instruction List (IL)*

Instruction List, eli käskylista, on toinen standardin tekstipohjaisista ohjelmointikielistä. Ohjelmointikieli muistuttaa Assembly-kieltä. Instruction List on käsky pohjainen ohjelmointikieli, joka toimii eurooppalaisena vastineena Ladder Diagramille. (PLCopen, 2013.)

2.2.2.3 Ladder Diagram (LD)

Ladder Diagram, eli tikas diagrammi, on graafinen ohjelmointikieli, jonka ulkoasu muistuttaa Yhdysvalloissa kehitettyä releohjauksien piirikaaviota. Ohjelma koostuu pääosin koskettimista, joita ohjataan erilaisilla ohjelmalohkoilla. Ohjelma on yksinkertainen ja helppolukuinen. Se soveltuu siten yksinkertaisiin ohjelmiin. (PLCopen, 2013.)

2.2.2.4 Function Block Diagram (FBD)

Function Block Diagram, eli logiikkakaavio-ohjelmointi, on myös graafinen ohjelmointikieli. Se rakentuu valmiista funktioista, jotka esiintyvät kielessä ohjelmapalikkoina. Funktiot ovat normaalisti erilaisia loogisia toimintoja kuten and, or ja add. Ohjelmointikieli sisältää myös edistyneempiä toimintoja, kuten laskurit ja viiveajastimet. (PLCopen, 2013.)

2.2.2.5 Sequential Function Charts (SFC)

Sequential Function Chartia, eli peräkkäisten funktioiden kaavio, käytetään sekvenssiohjelmointiin. Se itsessään ei ole ohjelmointikieli vaan sillä pikemminkin hallitaan niitä ja niiden suoritusjärjestystä. SFC koostuu sekvenssistä tai monista sekvensseistä, jotka sisältävät askeleessa tehtävän toiminnan ja siirtoehdon seuraavaan askeleeseen. Toiminta voi olla esimerkiksi jokin muulla ohjelmointikielellä luotu ohjelma tai toiminto. (PLCopen, 2013.)

3 AUTOMAATIOJÄRJESTELMÄN OHJELMOINTI

3.1 Projektin aloitus

Projektissa käytettäväksi järjestelmäksi valittiin Fidelixin tuotevalikoimasta Fx-2025A-säädin. Toinen vaihtoehto Fidelix-järjestelmään olisi ollut Fidelixin Fx-Spider, mutta koska kyse on ohjelmoinnin perusoppaasta, käytetään enemmän projekteissa käytettyä Fx-2025A-säädintä.

Ohjelmoinnissa käytettävät ohjelmistot ohjelmoijan täytyy hakea Fidelixin tukisivuilta (support.fidelix.fi). Tunnukset kyseisille sivuille antaa Fidelix. Sivuilta tulee ladata seuraavat ohjelmat: FxHmlEditor, OpenPCS 5.x0 sekä PointGen. HTMLEditorilla luodaan graafinen käyttöliittymä, OpenPCS-ohjelmaa käytetään PLC-ohjelmointiin ja Excel-pohjaista PointGenia pisteiden luomiseen.

3.2 Pisteiden luominen, PointGen

3.2.1 PointGen, johdanto

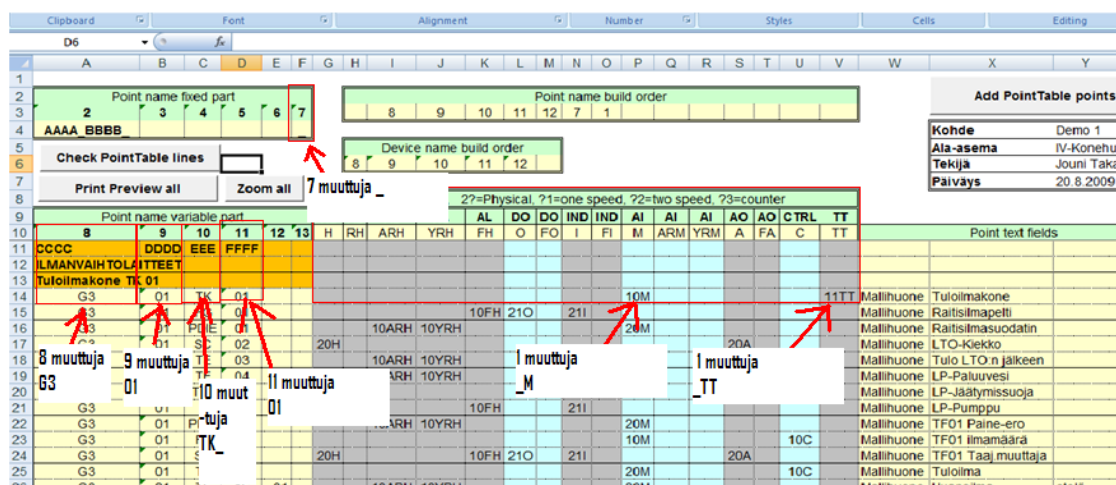
PointGen on Excel-pohjainen pisteidenluomiseen tarkoitettu ohjelma. Pisteet luodaan Excelissä ja muutetaan .txt-muotoon. Tämän jälkeen .txt-tiedosto siirretään säätimeen ja sisällytetään pisteet ohjelmaan.

PointGenin käyttöä aloittaessa on hyvä olla ns. puhdas pohja, jolle voidaan aloittaa luomisprosessi. Myös toisesta työstä kopioitua PointGen-tiedostoa voidaan käyttää, mutta tällöin virheiden tekoon on huomattavasti suurempi mahdollisuus.

3.2.2 Pisteiden luominen PointListillä

PointGenin PointList-välilehdellä luodaan pisteen tunnus. Pisteen tunnus määräytyy ”Point name build orderin”, eli pistetunnuksen rakennusjärjestyksen mukaan, joka näkyy kuviossa 1 keskellä ylhäällä. Kuviossa 1 pisteen tunnus määräytyy muuttujien

8, 9, 10, 11, 12, 7 ja 1 mukaan. Taulukossa vasemmalla ovat kaikki muut muuttujat paitsi muuttuja 1. Esimerkiksi kuvion 1 mukaisen ensimmäisen rivin pisteen alkuosan nimeksi tulisi G301TK01_. Kuviossa 1 G3 on muuttuja 8, 01 on 9, TK on 10, 01 on 11 ja _ on 7. Muuttuja 12 on tyhjänä, joten sitä ei sisällytetä pisteen nimeen. Muuttuja 1 on aina pisteen lopputunnus. Lopputunnus pisteelle määritellään keskellä taulukkoa. Kuviossa 1 ensimmäisen rivin piste on saanut kaksi lopputunnusta, _M ja _TT, jotka tarkoittavat mittauksia ja aikaohjelmaa. Piste pystytään myös määrittelemään ohjaukseksi (DO tai Digital Output), hälytykseksi (AL tai Alarm), digitaalituloiksi (IND tai Digital Input), säätöpisteeksi (AO tai Analog Output) tai asetusarvoksi (CTRL tai Control Point). Ohjelmoija itse tekee tarvittavat pisteet ohjelmaan joka kerta uudestaan tarpeen mukaan. Kun kaikki pisteet on lisätty taulukkoon, painetaan ”Add PointTable points to PointList”-painiketta oikeasta yläkulmasta. Tällöin kaikkien luotujen piste-tunnusten nimet generoidaan PointList-välilehdelle ja niille annetaan modulyyppi. Esimerkiksi digitaalitulot menevät IND-osaan.



Kuvio 1. PointTable

3.2.3 PointList

PointGenin PointList-välilehdellä asetellaan fyysisille pisteille osoitteet sijoittamalla kysymysmerkkien tilalle modulinumero ja kanavanumero. Esimerkiksi kuviossa 3 digitaalituloja on kiinnitetty modulin 24 kanaviin 1–4. Lehdellä myös pystyy muokkaamaan pisteen ominaisuuksia, kuten päälle- ja poismenoviivettä. Kun pisteille on määritelty osoitteet, ne tarkistetaan ”Check Module and Point number of Points”-

painikkeesta. Tämä tarkistaa, että kaikki pisteet on määrätty johonkin osoitteeseen ja että niissä ei ole esimerkiksi päällekkäisyyksiä. Jos tarkistus menee läpi, voidaan pisteet lisätä moduleihin painikkeella ”Add Modules”. Moduleiden lisäyksen jälkeen pistelistat ovat tulostettavissa seuraavilta välilehdiltä. Esim Mod_AI-välilehdessä on AI-moduliin liitetyt pisteet.

1	Version	Delete points	Count points								
2	123									Check Module and Point number of Point	
3											
4	SaveAndExit	Delete modules	Zoom all							Add PointTable points to PointList	
5	AL										
6	Name	Text	Port	Module	Point	OnDelay	OffDelay	Open	Contact 0/1	StateText	Priority
7	Write DefaultValues into this line										
8	G103TE42_YRH	IV-Verkosto Paluuvesi	3	0	0	1	5	0	0	HÄLYTYS	5
9	G103TE42_ARH	IV-Verkosto Paluuvesi	3	0	0	1	5	0	0	HÄLYTYS	5
10	G103TE41_YRH	IV-Verkosto Menovesi	3	0	0	1	5	0	0	HÄLYTYS	5

Kuvio 2. PointList, Hälytykset

69	IND										
70	Name	Text	Port	Module	Point	OnDelay	OffDelay	Open	Contact 0/1	StateText	TriState 0/1
71	Write DefaultValues into this line										
72	G103P01_I	IV-Verkosto Pumppu	3	24	1	0	5	0	0	SEIS_KÄY	0
73	G102P01_I	IV-Verkosto Pumppu	3	24	2	0	5	0	0	SEIS_KÄY	0
74	G101P01_I	Patteriverkosto Pumppu	3	24	3	0	5	0	0	SEIS_KÄY	0
75	G201P01_I	Käyttövesiverk. pumppu	3	24	4	0	5	0	0	SEIS_KÄY	0

Kuvio 3. PointList, Digitaalitulos

82	G301FG01_I	Mallihuone Raitisilmapelti	3	28	3	0	5	0	0	SEIS_KÄY	0
83	DO										
84	Name	Text	Port	Module	Point	OnDelay	OffDelay	Open	Contact 0/1	StateText	TriState 0/1
85	Write DefaultValues into this line										
86	G103P01_O	IV-Verkosto Pumppu	3	21	1	0	0	0	0	SEIS_KÄY	0
87	G102P01_O	IV-Verkosto Pumppu	3	21	2	0	0	0	0	SEIS_KÄY	0
88	G101P01_O	Patteriverkosto Pumppu	3	21	3	0	0	0	0	SEIS_KÄY	0
89	G301FE02_O	Mallihuone DE01 Tied. muuttaja	3	24	4	0	0	0	0	SEIS_KÄY	0

Kuvio 4. PointList, Ohjaukset

95	AI										
96	Name	Text	Port	Module	Point	Analog-1	Scaling	Min Pulse	Offset	Lookup Table	Unit
97	Write DefaultValues into this line										
98	G201FO03_M	Käyttövesiverk. Kylmävesimäärä	3	0	0	1	1	10	0		°C
99	G301FE21_M	Mallihuone PF01 ilmamäärä	3	0	0	1	1	10	0		°C
100	G301FE08_M	Mallihuone TF01 ilmamäärä	3	0	0	1	1	10	0		°C

Kuvio 5. PointList, Mittaukset

36	AO										
37	Name	Text	Port	Module	Point	Min Voltage	Max Voltage	Time Table	Sample Time		
38	Write DefaultValues into this line										
39	G103FV42_A	IV-Verkosto Venttiili	3	23	1	0	10		60		
40	G102FV01_02_A	IV-Verkosto Venttiili, iso kvs	3	23	2	0	10		60		
41	G102FV01_01_A	IV-Verkosto Venttiili, pieni kvs	3	23	3	0	10		60		

Kuvio 6. PointList, Säädöt

148	CTRL								
149	Name	Text	Module	Point					
150	Write DefaultValues into this line								
151	G103TE41_C	IV-Verkosto Menovesi	0	0					
152	G103PDE01_C	IV-Verkosto Paine-ero	0	0					
153	G102TE41_C	IV-Verkosto Menovesi	0	0					

Kuvio 7. PointList, Asetusarvot

164	TT					
165	Name	Text	Module	Point	StateText	
166	Write DefaultValues into this line					
167	G103P01_TT	IV-Verkosto Pumppu	0	0	SEIS_KÄY	
168	G102P01_TT	IV-Verkosto Pumppu	0	0	SEIS_KÄY	
169	G101P01_TT	Patteriverkosto Pumppu	0	0	SEIS_KÄY	

Kuvio 8. PointList, Aikaohjelmat

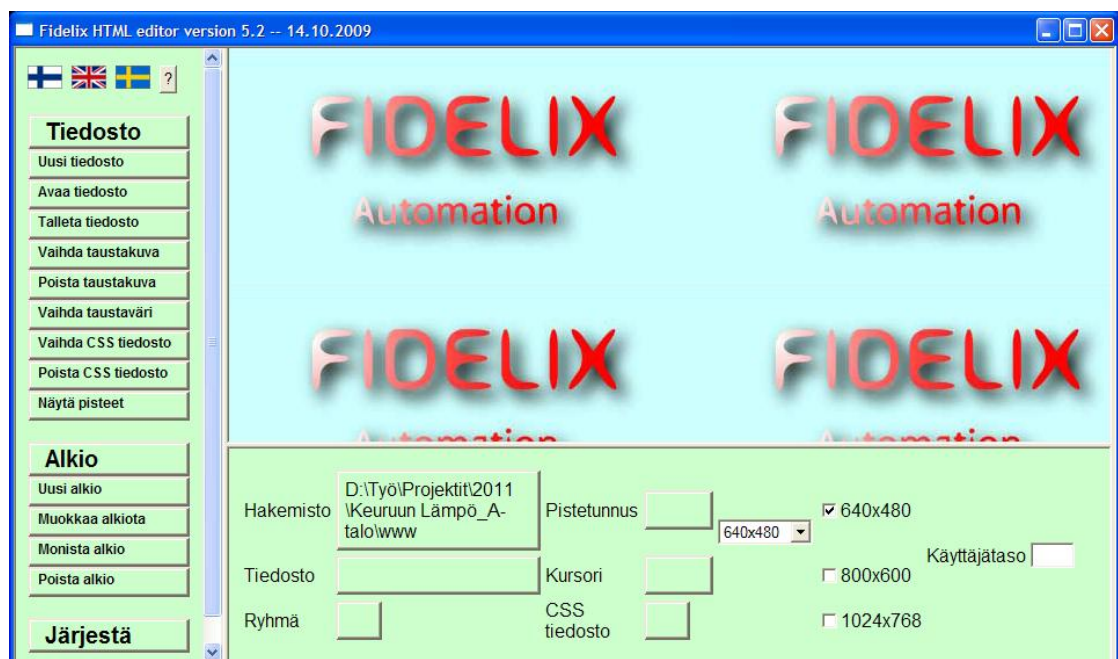
3.2.4 Pisteiden lataaminen säätimelle

Kun PointGenistä on poistuttu painamalla ”Save & Exit”-painiketta, luo se kaksi .txt-tiedostoa kansioon, jossa PointGen sijaitsee. Näistä _v123.txt-päätteinen tiedosto siirretään säätimeen käyttämällä ftp-palvelinta. Kun tiedosto on siirretty säätimelle, muodostetaan säätimeen telnet-yhteys esimerkiksi suorittamalla komentorivillä käsky ”telnet <säätimen ip-osoite>”. Yhteyden luominen vaatii käyttäjätunnuksen (fx2020) ja salasanan (demo). Kun telnet-yhteys on muodostettu, komennolla ”importpoints <.txt tiedoston polku, esim. HDisk/Fidelix/xxx.txt>” ladataan pisteet teksti-tiedostosta säätimeen. Telnet-yhteys terminoidaan tämän jälkeen komennolla ”exit”.

3.3 Grafiikkakuvien luominen, Fidelix HtmlEditor

3.3.1 Päänäkymä

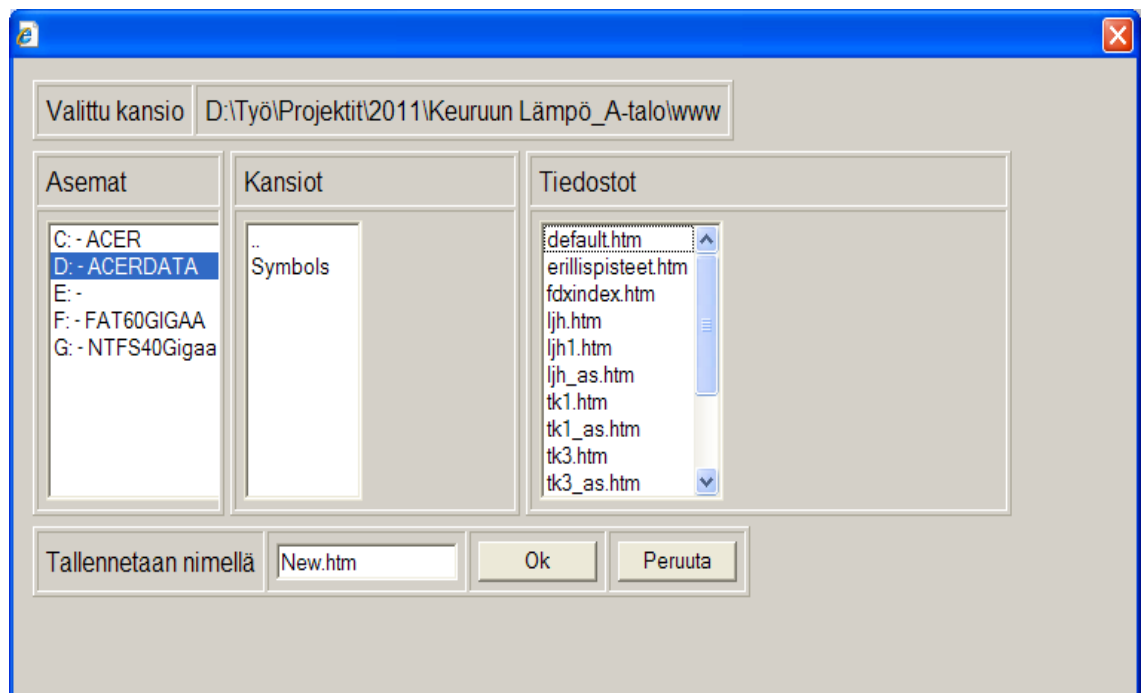
Kun ohjelma käynnistetään, avautuu kuvion 9 mukainen päänäkymä. Vasemmalla sivupalkissa sijaitsevat tärkeimmät toiminnot, ja isoon ikkunaan luodaan säätimelle tallentuvat kuvat.



Kuvio 9. Fidelix HTML editor

3.3.2 Uusi tiedosto/Avaa tiedosto/Tallenna tiedosto-valikko

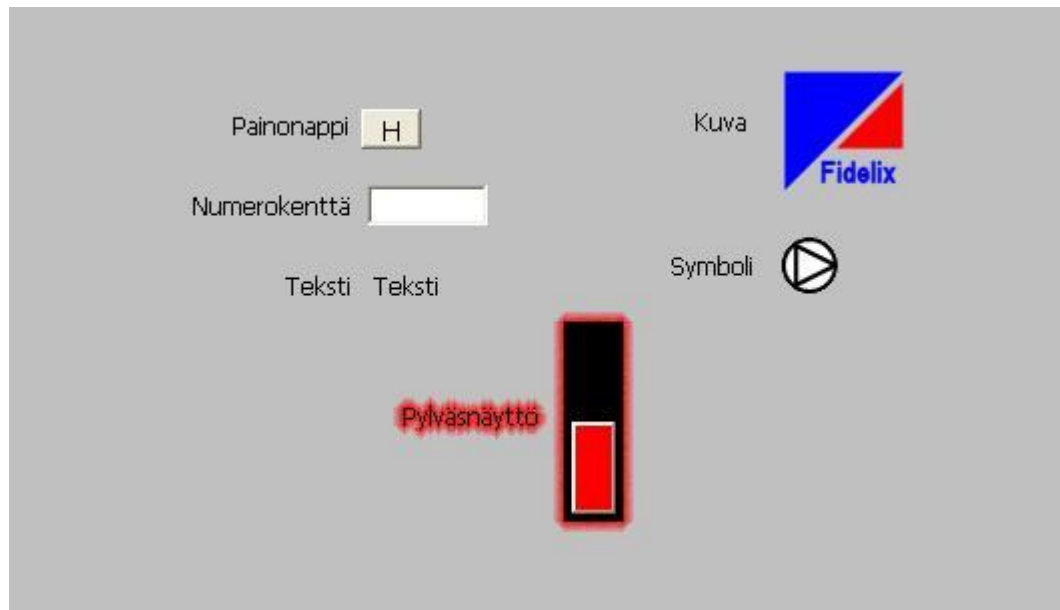
”Uusi tiedosto”-, ”Avaa tiedosto”- tai ”Tallenna tiedosto”-painiketta painamalla avautuu kuvion 10 kaltainen näkymä. Uutta tiedostoa luotaessa valitaan kansio, johon tiedosto tallennetaan, sekä luodaan uuden tiedoston nimi. Tallennettaessa tiedostoa valitaan kansio, johon tiedosto tallennetaan (oletuksena on kansio, josta tiedosto avattiin). Kun tiedostoa avataan uudestaan, se valitaan kansioista, johon se on tallennettu.



Kuvio 10. FxHTMLEditor, Uusi Tiedosto/Avaa tiedosto/Tallenna tiedosto

3.3.3 Alkio-valikko

Alkio-valikosta voidaan luoda, muokata, monistaa sekä poistaa alkioita. Kaikki kuviossa 11 olevat objektit ovat alkioita.

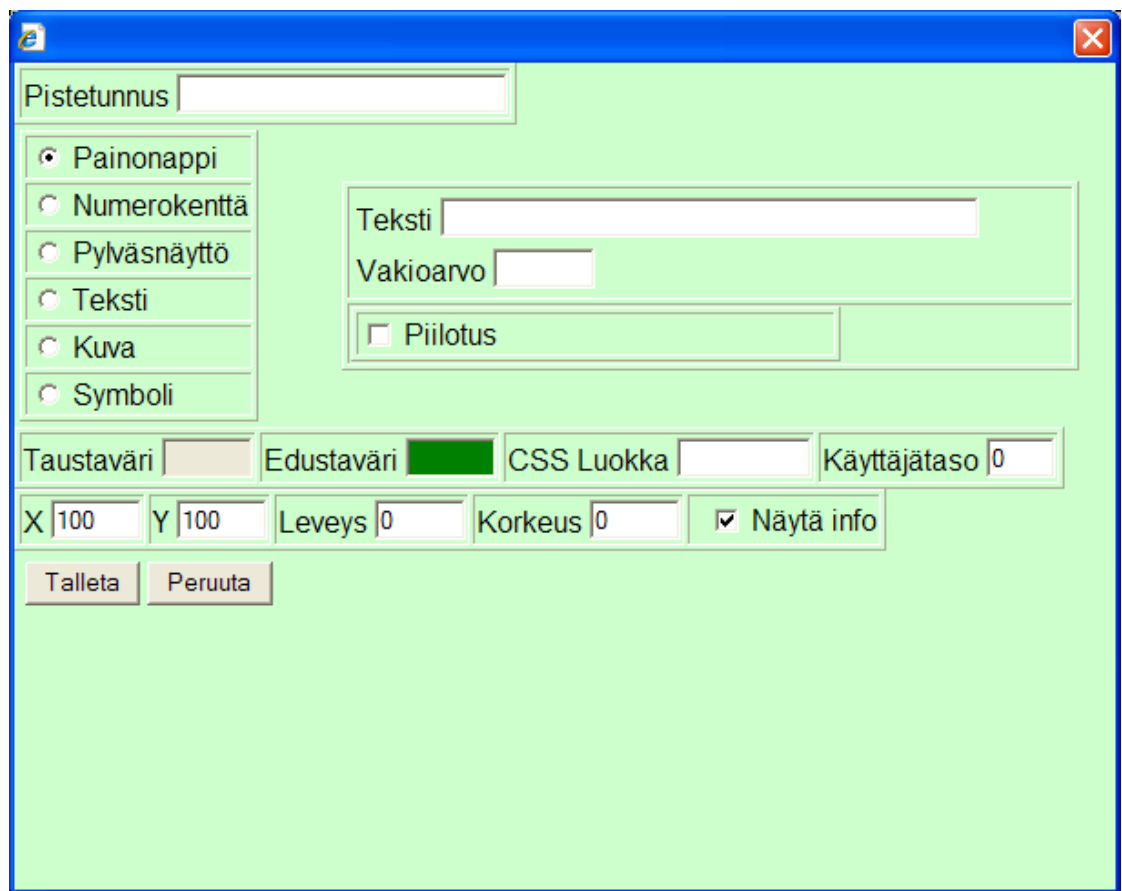


Kuvio 11. FxHTMLEditor, Alkiot

Alkion luominen aloitetaan painamalla ”Uusi alkio”-painiketta, jonka jälkeen avautuu kuvion 12 näkymä.

3.3.3.1 Painonappi

Kuvion 12 pistetunnus-kohdassa määritellään alkion pistetunnus, esim. LQH_PV01_TE01_ARH. Vasemmalla olevasta sarakkeesta valitaan alkion tyyppi. Painonappi-alkiota tehdessä tulee määrittää teksti, joka on painonapin päällä, alkion vakioarvo (oletus 0) sekä se, onko alkio piilotettu (näkyvää vain, jos alkion pistetunnus on 1). Kun painonapin tunnus on 0, on painonappi taustavärin värinen. Kun painonapin tunnus on 1, on painonappi edustavärin värinen. Kun painonappi-alkio on valmis, painetaan ”Talleta”, jolloin alkio tulee näkyviin. Myöhemmin alkiota voi muokata joko tuplaklikkaamalla alkiota tai valitsemalla alkion oikealla hiirenpainikkeella ja painamalla ”Muokkaa”-painiketta.

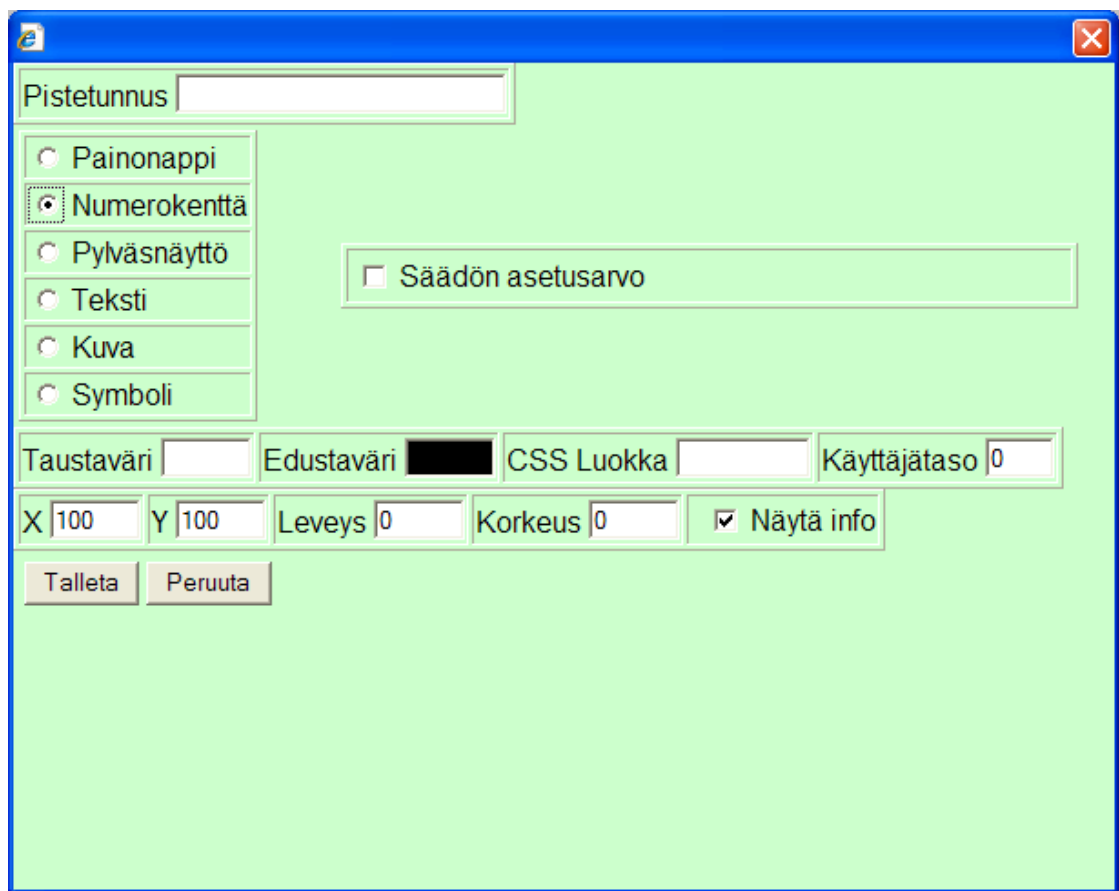


Kuvio 12. FxHTMLEditor, Pistetunnus

3.3.3.2 Numerokenttä

Kuvion 13 pistetunnus-kohdassa määritellään alkion tunnus, esim.

LJH_PV01_TE01_M. Numerokenttä-alkiota tehdessä ei tarvitse tehdä muita toimenpiteitä, jos numerokenttä ei ole säätöpiste. Jos numerokenttä on säätöpiste (pisteen lopputunnus on _S), valitaan "Säädön asetusarvo"-valinta. Jos sitä ei valita, näyttää numerokenttä säädön tilan (aktiivinen/pois käytöstä). Taustaväri määrittää numerokentän värin ja edustaväri numerokentän tekstin värin. Kun alkio on valmis, painetaan "Talleta", ja alkio tulee näkyviin. Myöhemmin alkioita voi muokata joko tuplakklikkaamalla alkioita tai valitsemalla alkion oikealla hiirenpainikkeella ja painamalla "Muokkaa"-painiketta.

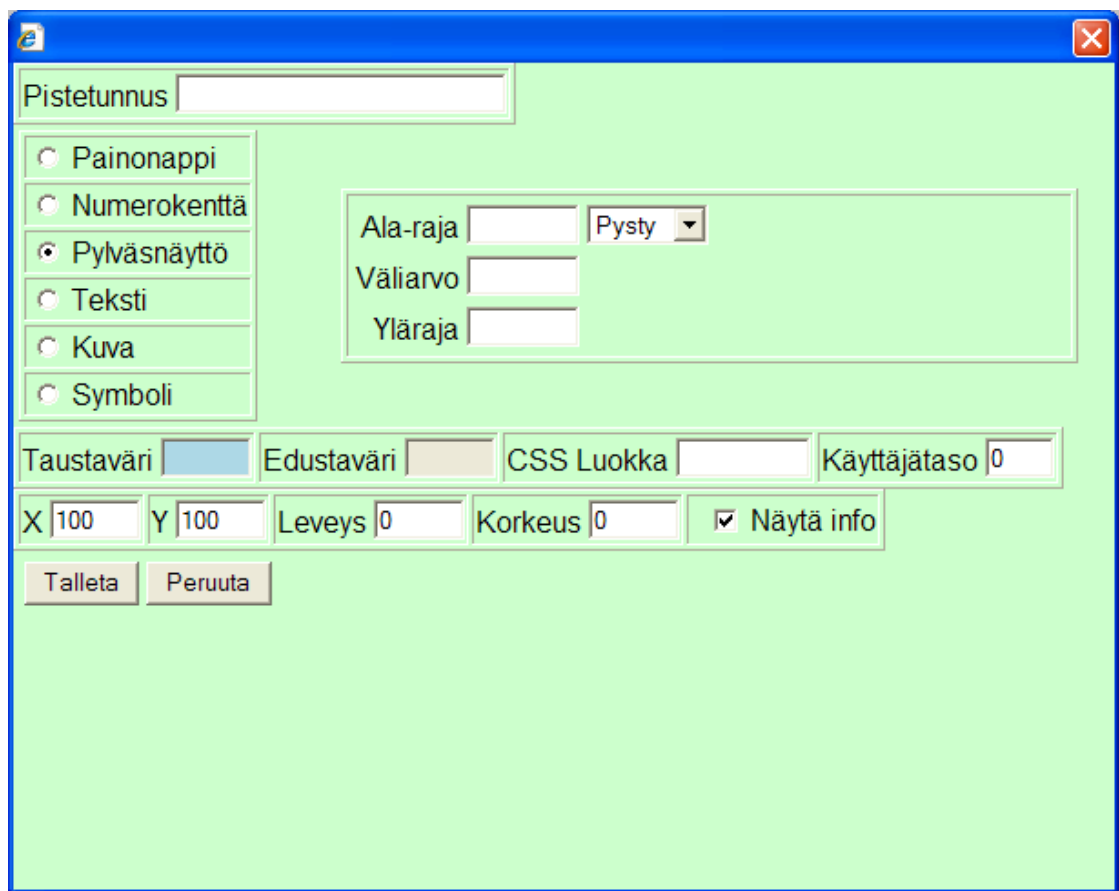


Kuvio 13. FxHTMLEditor, Numerokenttä

3.3.3.3 Pylväsnäyttö

Kuvion 14 pistetunnus-kohdassa määritellään alkion tunnus, esim.

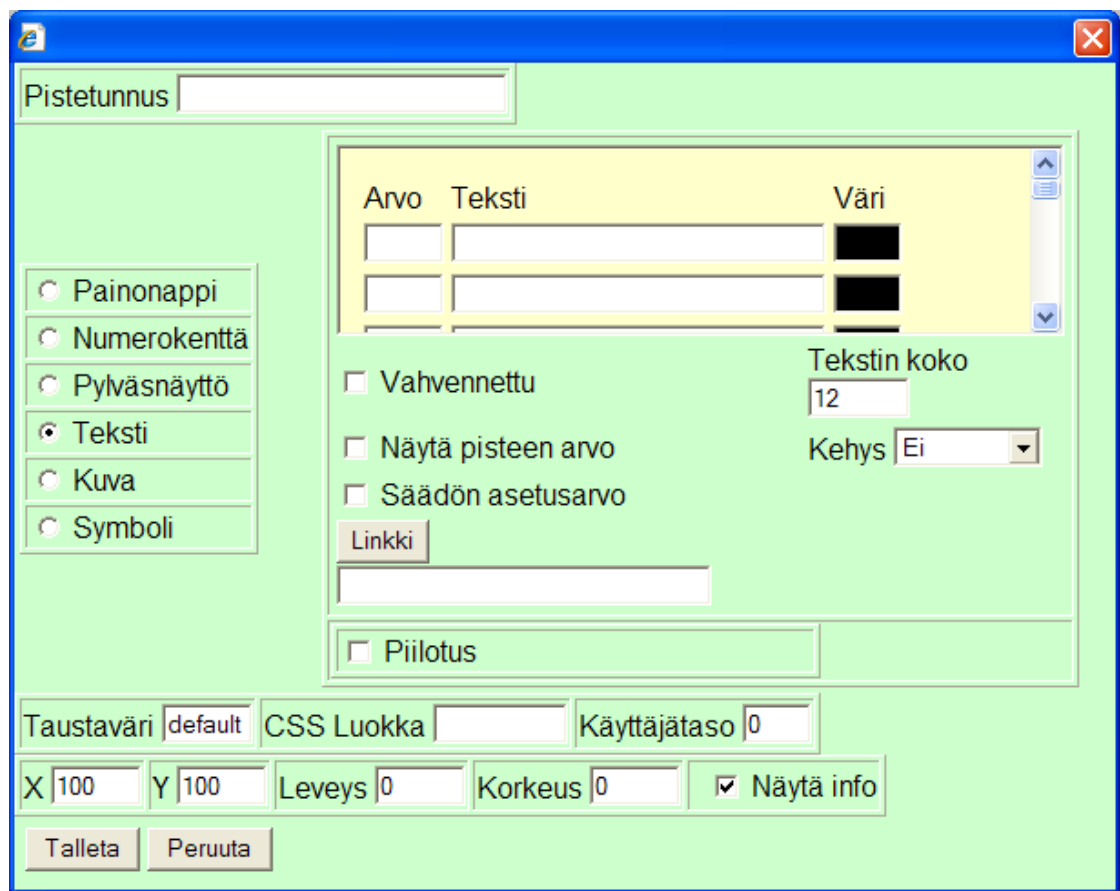
LJH_PV01_TV01_A. Pylväsnäyttö-alkiota tehdessä alkion tulee määrittellä sen alin arvo, pylvään keskiosan arvo sekä ylin arvo. Taustaväri määrittää pylväsnäytön taustan ja edustaväri sen täyttövärin. Kun alkio on valmis, painetaan "Talleta", ja alkio tulee näkyviin. Myöhemmin alkioita voi muokata joko tuplaklikkaamalla alkioita tai valitsemalla alkion oikealla hiirenpainikkeella ja painamalla "Muokkaa"-painiketta.



Kuvio 14. FxHTMLEditor, Pylväs näyttö

3.3.3.4 Teksti

Kuvion 15 pistetunnus-kohdassa voidaan määrittellä alkion tunnus, esim. TK1_FI, jos halutaan tunnuksen muuttavan tekstiä eri arvoilla. Teksti-alkiota tehdessä pisteelle tulee määrittellä, minkä tekstin alkio milläkin pisteen arvolla näyttää. Jos alkion ei ole määriteltä tunnusta, jätetään arvo-kohta tyhjäksi ja kirjoitetaan haluttu teksti ensimmäiseen tekstisarakkeeseen. Tekstin ulkoasua (fontti, kehykset) voidaan myös muokata halutunlaisiksi. Jos tekstiä painamalla halutaan siirtyä toiseen kuvaan, määritellään tekstille linkki, jonka nimenä on halutun kuvan nimi (esim. ljh.html). Kun alkio on valmis, painetaan "Talleta", ja alkio tulee näkyviin. Myöhemmin alkion voi muokata joko tuplaklikkaamalla alkion tai valitsemalla alkion oikealla hiirenpainikkeella ja painamalla "Muokkaa"-painiketta.



Kuvio 15. FxHTMLEditor, Teksti

3.3.3.4.1 Poikkeustapauksia

Jos halutaan hakea päivämäärä ja aika säätimeltä ja esittää se kuvassa, voidaan käyttää hyväksi teksti-alkiota. Teksti-alkion ensimmäiseen riviin kirjoitetaan "Date" tai "Time" sen mukaan, kumpi alkio halutaan luoda. Date tarkoittaa päivää ja Time tarkoittaa aikaa. Kaikki muu jätetään tyhjäksi. Tekstiä voidaan vahventaa ja fonttia muuttaa, jos halutaan ajan tai päivämäärän näkyvän paremmin.

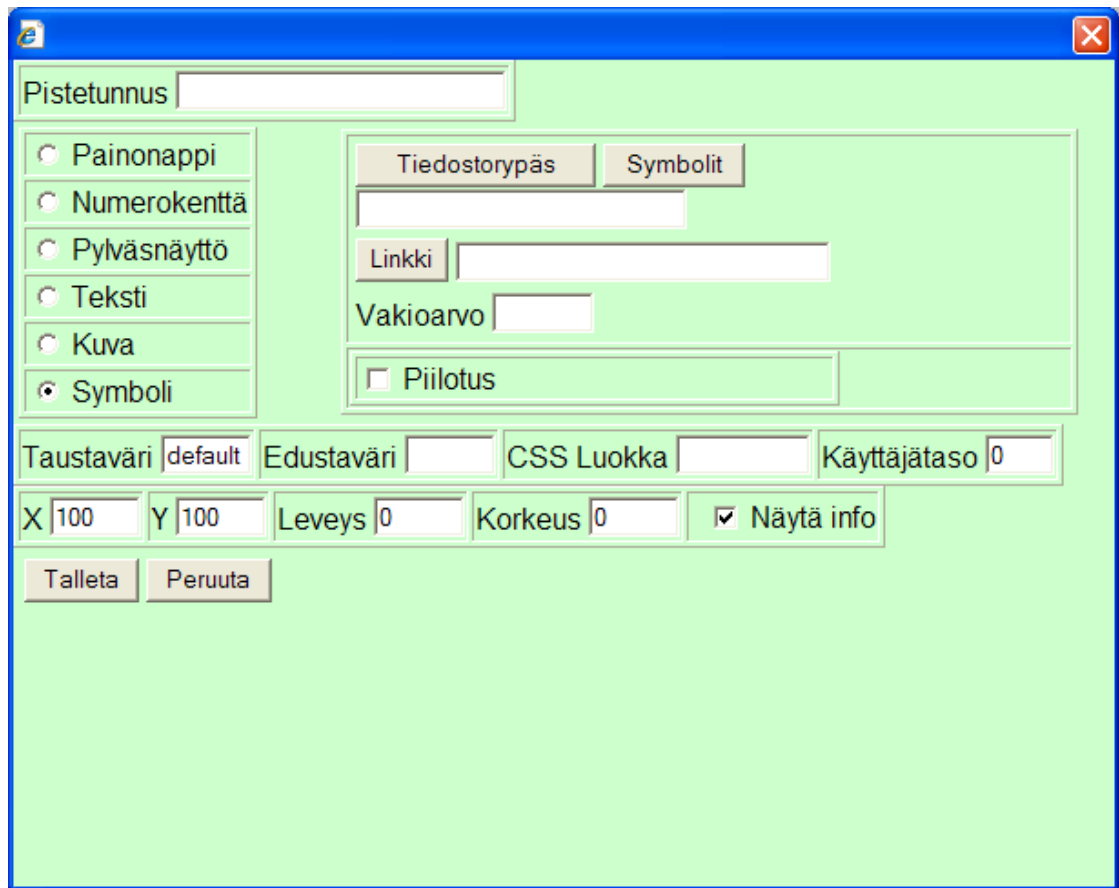
3.3.3.5 Kuva

Grafiikka-kuvaan voidaan lisätä kuvatiedostoja. Kuvatiedoston pitää olla muotoa "nimi-0-640/800/1024.jpg/png/gif". Kuvatiedostoa voidaan esimerkiksi käyttää taustakuvana tai kuvatiedostolle voidaan määrittää tunnus sekä linkkaus johonkin toiseen kuvaan samalla tapaa kuin tekstille.

Kuvio 16. FxHTMLEditor, Kuva

3.3.3.6 Symboli

Kuvio 17 pistetunnus-kohdassa määritellään alkion tunnus, esim. TK1_TF01_I. Symboli-alkiota tehdessä tulee alkion määrittely sen symboli. Symbolit haetaan Fidelixin symbolikirjastosta painamalla "Symbolit"-painiketta. Symbolikirjaston on oltava samassa kansiossa kuin kuvan, jota ollaan tekemässä. Myös omia symboleita voi tehdä. Symbolille voidaan tehdä linkkaus samoin kuin tekstille ja kuvalle. Symboli voidaan myös piilottaa (eli se näkyy vain, kun symbolin pistetunnuksen arvo on 1). Kun alkio on valmis, painetaan "Talleta", ja alkio tulee näkyviin. Myöhemmin alkion voi muokata joko tuplaklikkaamalla alkion tai valitsemalla alkion oikealla hiirenpainikkeella ja painamalla "Muokkaa"-painiketta.



Kuvio 17. FxHTMLEditor, Symboli

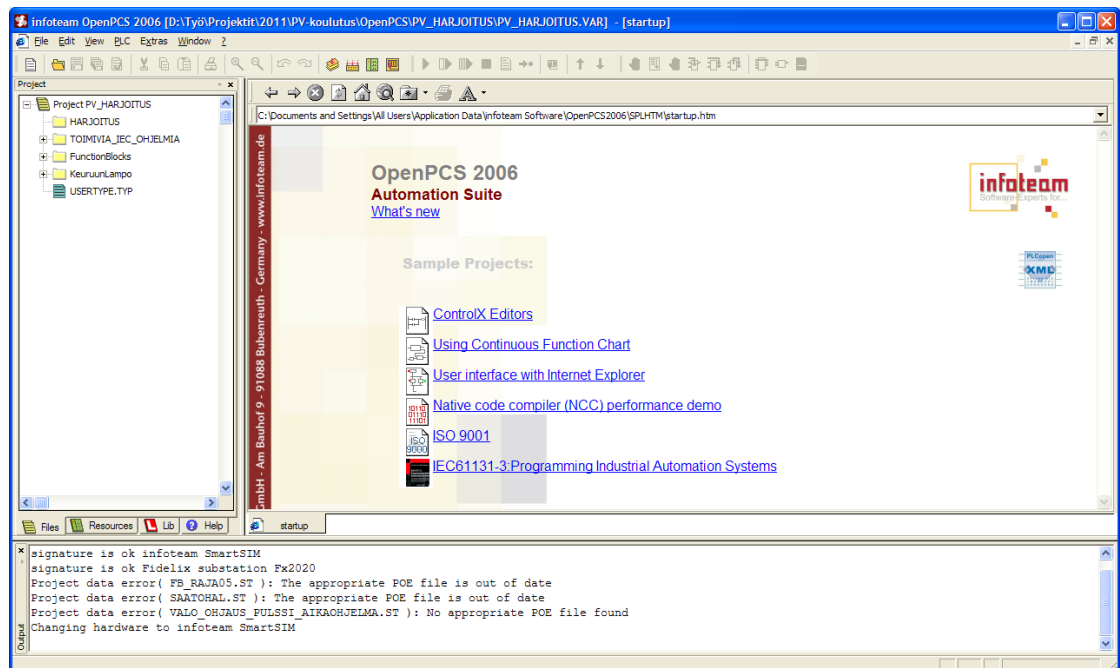
3.3.3.7 Kuvan rakentaminen

Kuva luodaan lisäämällä alkioita ja rakentamalla niistä kokonaisuus. Ohjelmassa toimivat Ctrl+C- ja Ctrl+V-toiminnot. Jokaista alkiota ei siis tarvitse erikseen luoda, vaan niitä voidaan kopioida ja muokata halutunlaisiksi. Alkioita voidaan liikuttaa nuolinäppäimillä pikseli kerrallaan ja Shift+nuolinäppäimellä viisi pikseliä kerrallaan. Alkioita pystytään raahaamaan ympäri kuvaa myös hiiren avulla. Jokaisen alkion kokoa voidaan muuttaa halutuksi alkion ominaisuuksien alimmalla rivillä. Toiminto on erittäin hyödyllinen esim. ilmastointikanavan luomisessa, kun kanavan leveyttä ja pituutta muutetaan. Toinen tapa on pitää Ctrl-näppäin pohjassa ja muuttaa alkion kokoa pikseli kerrallaan tai painaa Ctrl+Shift ja muuttaa kokoa viisi pikseliä kerrallaan. Lopuksi, kun kuva on valmis, voidaan tarvittaessa vielä muokata sen resoluutiota.

3.4 OpenPCS-ohjelmointityökalu

3.4.1 Yleistä

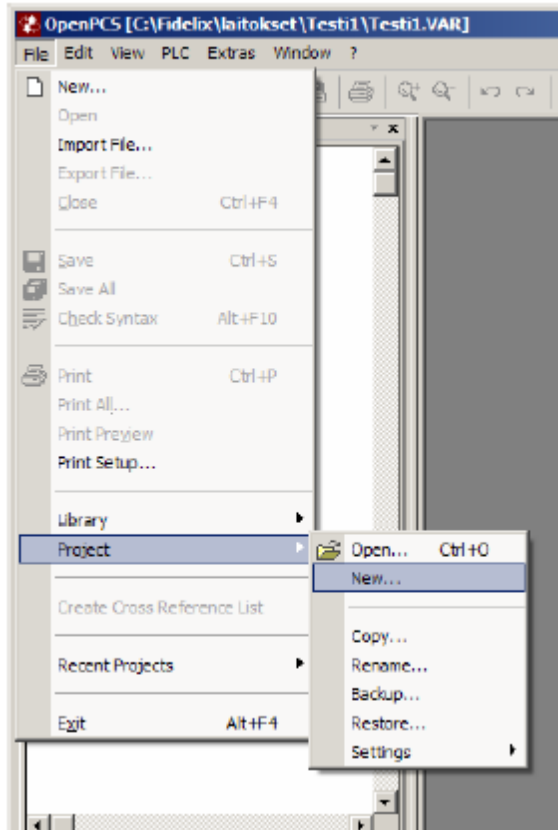
OpenPCS on IEC 61131-3-standardin pohjalta luotu ohjelmointityökalu. IEC 61131-3-standardin mukainen tekstipohjainen ohjelmointikieli Structured Text on hyvin samankaltainen kuin yleisimmät tekstipohjaiset ohjelmointikielet, kuten C++. Näin ollen kielen ymmärtäminen ei ole kovin vaikeaa, jos käyttäjä osaa esimerkiksi C++-kielen perusteet. Fidelixin säädin ohjelmoidaan käyttäen Structured Text ohjelmointikieltä. Tässä oppaassa käydään läpi ohjelmointityökalun peruskäyttö ja IEC-kielen peruskäytöt, kuten muuttujien hakeminen säätimeltä ja IF-lause.



Kuvio 18. OpenPCS-työkalu

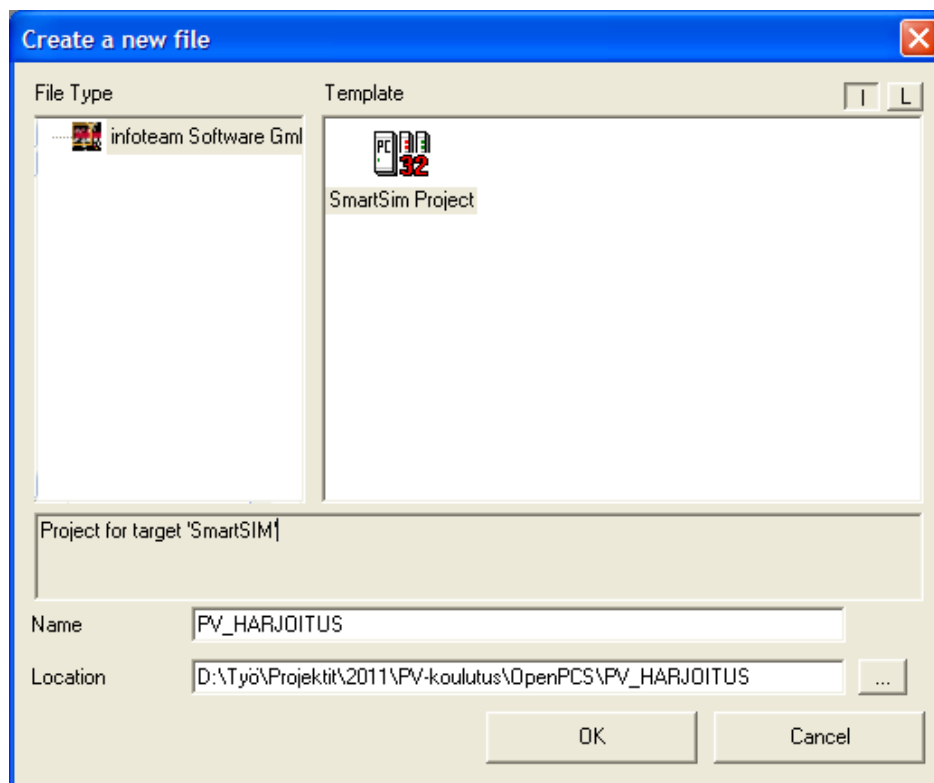
3.4.2 Uuden projektin luominen

Uusi projekti luodaan avaamalla OpenPCS-ohjelmointityökalu ja valitsemalla kuvion 19 esimerkin mukaisesti yläpalkista File → Project → New...



Kuvio 19. OpenPCS, uusi projekti

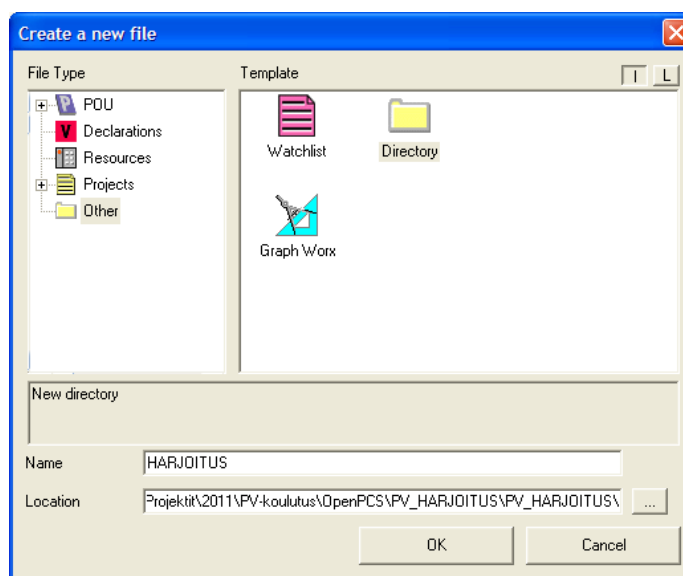
Avautuvasta ikkunasta valitaan projektille nimi ja tallennuskansio. Lopuksi painetaan "OK"-painiketta, ja uusi projekti on luotu.



Kuvio 20. OpenPCS, Luo projekti

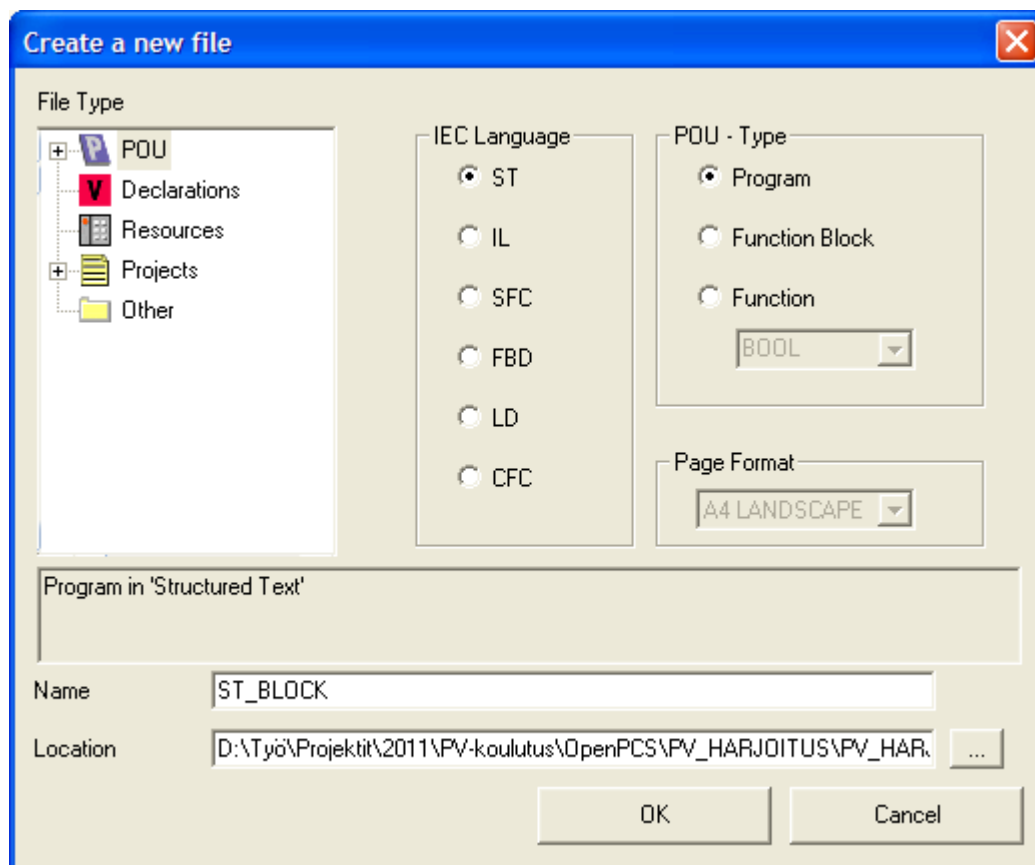
3.4.2.1 Uuden tiedoston luominen

Yläpalkista File → New luodaan kaikki ohjelmointiin liittyvät toiminnot, kuten ohjelmointiosiot, -kansiot ja -funktiot.



Kuvio 21. OpenPCS, Kansion luonti

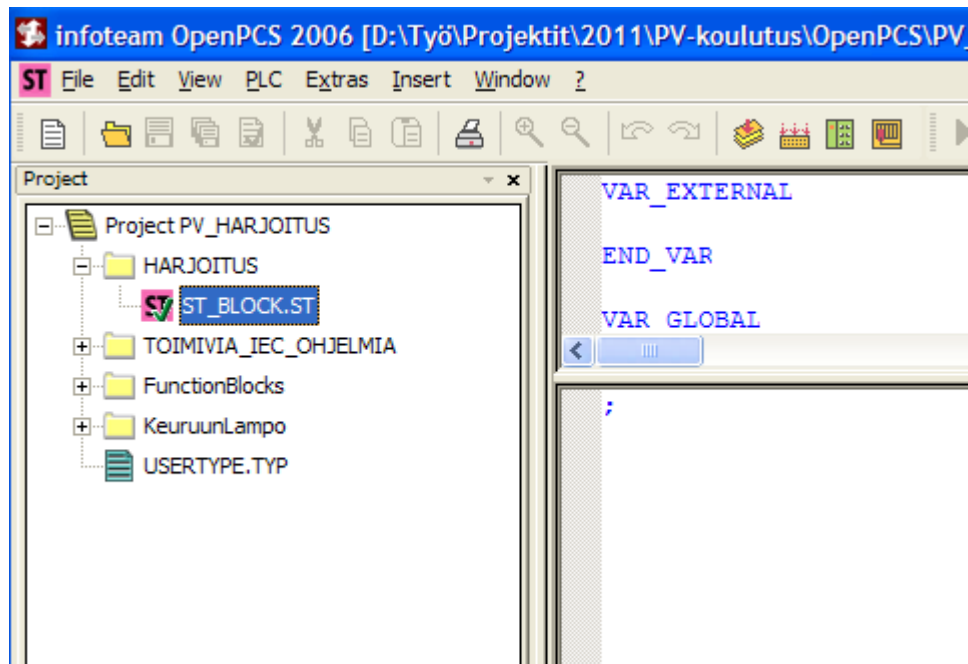
Kun tehdään uutta kansiota, johon voi laittaa ohjelmia, valitaan Filetype-valikosta kuvion 21 mukaan "Other" ja sieltä "Directory". Sen jälkeen valitaan kansiolle nimi ja paikka johon se luodaan. Lopuksi painetaan "OK" ja uusi kansio on luotu projektille.



Kuvio 22. OpenPCS, Ohjelmaosion luonti

Kun halutaan luoda ohjelmaosio, valitaan samasta valikosta (File Type) POU ja tehdään kuvion 22 mukaiset valinnat. IEC Languageksi, eli ohjelmointikieleksi, valitaan ST sekä POU Typeksi Program. Tämän jälkeen valitaan ohjelmaosion nimi ja sen tallennuskansio. Tallennuskansioksi on suositeltavaa valita Projekti-kansion sisällä oleva kansio, joka luotiin kohdassa 3.4.2.1. Tämän jälkeen ensimmäinen kansio sekä ohjelmaosio on luotu.

Ohjelmaan pystytään myös lisäämään aikaisemmin tehtyjä kansioita sekä ohjelmaosioita. Se voidaan tehdä sulkemalla ohjelma (kun projekti on tehty) ja kopioimalla vanhat ohjelmakansiot uuden projektin kansioon.

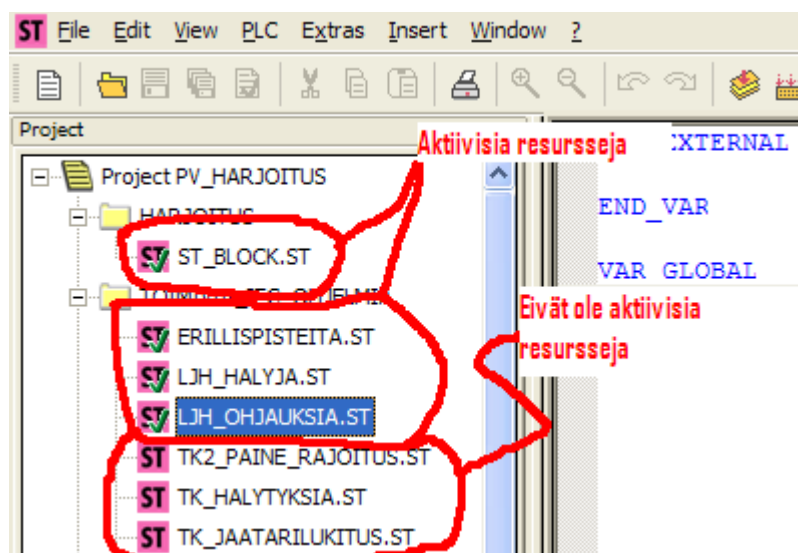


Kuvio 23. OpenPCS, Ohjelmaosion näkymä

Yksittäisiä ohjelmia voidaan myös lisätä kopiaamalla ohjelman .ST-, .bak- ja .POU-tiedostot kohdekansioon.

3.4.2.2 Aktiiviset resurssit

Aktiiviset resurssit tarkoittavat ohjelmaosioita, jotka ajetaan säätimeen.



Kuvio 24. OpenPCS, Aktiiviset resurssit

Resurssit valitaan aktiivisiksi hiiren oikealla painikkeella ja valitsemalla "Link To Active Resources". Jos ne halutaan poistaa resursseista, tehdään sama toimenpide.

3.4.3 Ohjelmointi

Kun ohjelmointi aloitetaan, ensimmäiseksi haetaan muuttujat, eli pisteet, säätimestä ohjelmaan. Säätimessä on pääasiassa kahta muuttujatyyppiä, Int- ja Real-muuttujia, joita käytetään. Int-muuttujat ovat kokonaislukuja (0/1/2...), esimerkiksi aikaohjelmia ja digitaalituloja eli indikoiteja. Real-muuttujat ovat reaalilukuja eli muuttujia, joissa on jokin arvo (0.00), esimerkiksi lämpötiloja. Muita määrittäjiä ovat esimerkiksi viiveet (dint). Kaikkien ohjelmaan tuotujen pisteiden tyyppi on määriteltävä. Tässä osiossa käydään läpi, kuinka pisteiden arvot haetaan ohjelmaan sekä kuinka niitä käytetään ohjelmassa. Tämän lisäksi käydään läpi IF-ELSE-lause.

3.4.3.1 Pisteen arvon lukeminen tietokannasta

Kun halutaan lukea jonkin pisteen arvo säätimeltä ohjelmaan, täytyy se hakea jollakin seuraavista komennoista:

IntMuuttuja := **GetDigitalPointF(Name := 'PisteTunnusSäätimellä')** ;

- Käsky palauttaa pisteen arvon Int-tyyppinä.

RealMuuttuja := **GetAnalogPointF(Name := 'PisteTunnusSäätimellä')** ;

- Käsky palauttaa pisteen arvon Real-tyyppinä.

Tässä on huomattava, että yllä olevien kahden funktion ero on palautettavan arvon tyyppi. Int on kokonaisluku (digitaalinen) ja Real on reaaliluku (analoginen).

realMuuttuja:= **GetLimitF(LimitNumber:=x, Name := 'PisteTunnus')**

- Käsky palauttaa tietyn mittauspisteen raja-arvon. Raja-arvo on jollekin mittaukselle ominainen arvo, jota voi käyttää esimerkiksi hälytysrajana.

RealMuuttuja := **GetConversionTableF(Value := 10.60, Name := 'TaulukonNimi')** ;

- Käskyllä haetaan muunnostaulukko säätimeltä. Value on muunnostaulukon muutettava arvo. Value on aina Reap-tyyppinen. "RealMuuttuja" on muunnostaulukossa muutettu arvo. Valuena pystytään käyttämään esim. ulkolämpötilaa, jolloin RealMuuttuja on aina ulkolämpötilan arvo muutettuna muunnostaulukon arvoon.

(* Luetaan säätöpiste *)

```
Control_Muuttuja( Name := 'SäätoPisteenTunnus' );
```

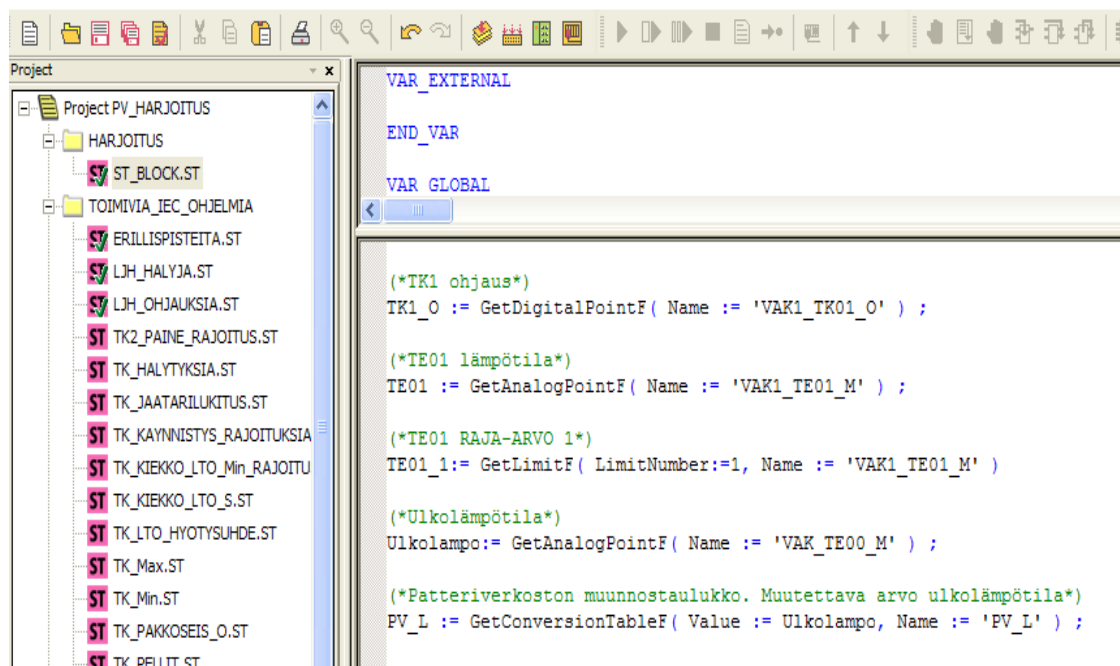
(* Luetaan säätöpisteen 1. portaan arvo *)

```
SaatoArvo_Muuttuja := Control_Muuttuja.stage1 ;
```

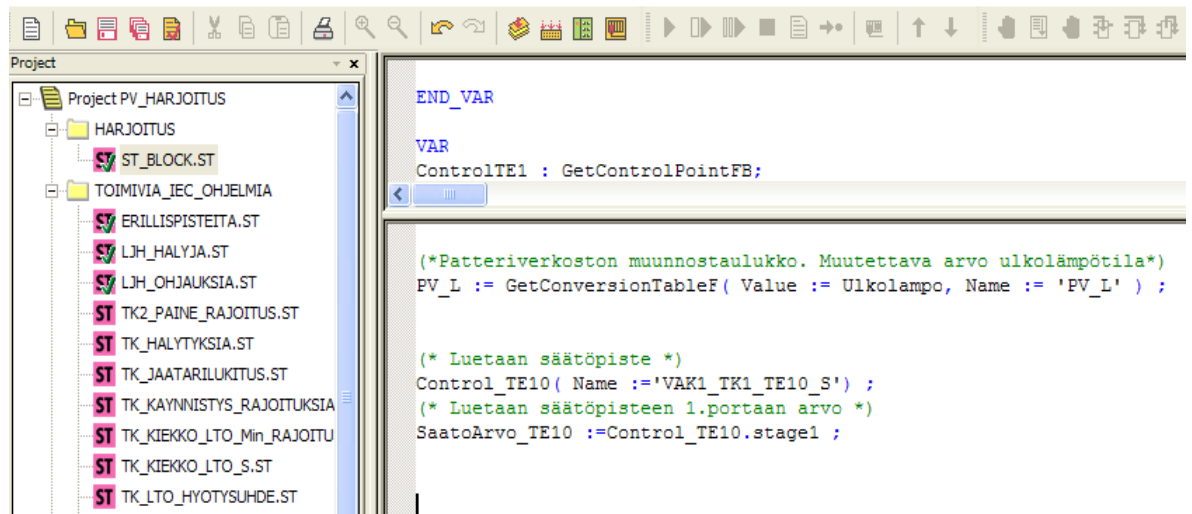
Tällä tavoin luetaan säätöpisteen ensimmäisen säätöportaan arvo (stage1). Säätöpiste eroaa muista siinä, että säätöpisteenpisteen määrittely on ohjelman sisäinen FB-kutsu. SaatoArvo_Muuttuja on normaali reaalityyppi.

3.4.3.2 Pisteen hakeminen ohjelmassa

Kuviossa 25 näytetään, miltä kohdassa 3.4.3.1 mainitut komennot näyttävät ohjelmassa. Lisäksi kuviossa 26 käydään vielä erillisenä säätöpisteenhaku ohjelmaan.



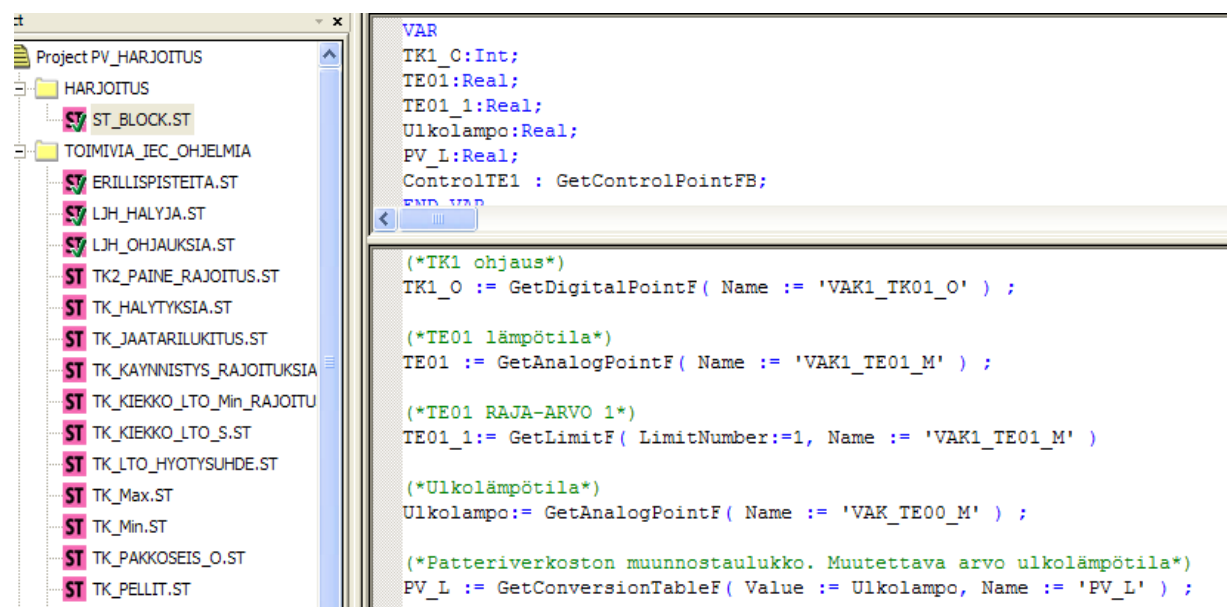
Kuvio 25. OpenPCS, Muuttujien haku



Kuvio 26. OpenPCS, Säätöpisteen haku

3.4.3.3 Variable table

Jokainen muuttuja täytyy lisätä OpenPCS-ohjelmointityökalussa erilliseen variable table -osiin. Variable table on muuttujataulukko, jossa muuttujille annetaan datatyyppi. Jos muuttuja on Int-tyyppinen (kokonaisluku), täytyy muuttujan loppupääte olla :Int;, ja jos se on Real-tyyppinen (reaaliluku) täytyy loppupääte olla :real;. Kuviossa 27 havainnollistetaan, miten variable table -osiota hallinnoidaan.



Kuvio 27. OpenPCS, Var

Kuviosta 27 nähdään, että jokainen muuttuja täytyy lisätä sivun yläosassa olevaan variable table -osiin. Tässä tapauksessa vain TK1_O on Int-tyyppinen muuttuja. Myös esimerkiksi digitaaliset tulot ovat Int-tyyppisiä. Samantyyppiset muuttujat voi-

daan myös laittaa peräkkäin. Tällöin ne on erotettava pilkulla ja :int; tai :real; tulee laittaa rivin perään.

3.4.3.4 IF-lause

IF-lauseella pystytään tekemään suurin osa ohjelmoinnista, ja se on yksi tärkeimmistä lauseista muissakin ohjelmointikielissä. IF-lause on nimensä mukaisesti jos-lause, eli jos määrätyt ehdot täyttyvät, tapahtuu tietty asia. Seuraavassa esimerkissä käytetään yksinkertaista IV-kone-ohjausta. _T-pääte on aikaohjelma, _O ohjaus ja _H hälytys.

(* OHJELMA TK1 *)

If TK1_T = 1 and TZA_H = 0 then

TULOS:=SetDigitalPointF(Value:=1,LockState:=1,Name:='VAK1_TK1_O');

else

TULOS:=SetDigitalPointF(Value:=0,LockState:=1,Name:='VAK1_TK1_O');

End_if;

Esimerkistä selviää, että sana "if" aloittaa IF-lauseen. Tämän jälkeen määritellään ehdot. Tässä tapauksessa ehtoina ovat, että TK1-aikaohjelma on päällä ja jäätymsuoja TZA ei ole aktiivinen. Then-sana on aina pakollinen ehtolauseen lopussa. Seuraavaksi muuttuja TULOS kirjoittaa arvon 1 TK1-ohjaukseen, eli kone lähtee päälle. Sana "else" tarkoittaa, että jos ehtolause ei toteudu, tapahtuu sanan jälkeinen TULOS. Tässä tapauksessa, jos ehtolause ei toteudu, on TK1 ohjaus pois päältä. End_if; päättää ehtolauseen. Ehtolauseita voi olla myös useampia, kuten seuraavassa esimerkissä:

(* IV-pumpun ohjaus *)

If TE00 < IV_KAY or IV_AIKA = 1 then

TULOS_4 := SetDigitalPointF(Value:=1,LockState:=1,Name:='VAK1_LJH_102_P01_O');

else If (TE00 > IV_SEIS and IV_AIKA = 0) then


```
TULOS_4 := SetDigitalPointF(Value:=0,LockState:=1,Name:='VAK1_LJH_102_P01_O');
```

```
End_if;
```

```
End_if;
```

Tässä lauseessa ELSE IF -lause indikoi toista IF-lausetta. IF-lause täytyy lopettaa niin moneen End_if;-päätöslauseeseen kuin lauseessa on IF-sanoja (esimerkissä on kaksi: IF ja ELSE IF). IF-lauseita voi laittaa monia yhteen rakenteeseen.

3.4.3.4.1 IF-lauseen ehtolause

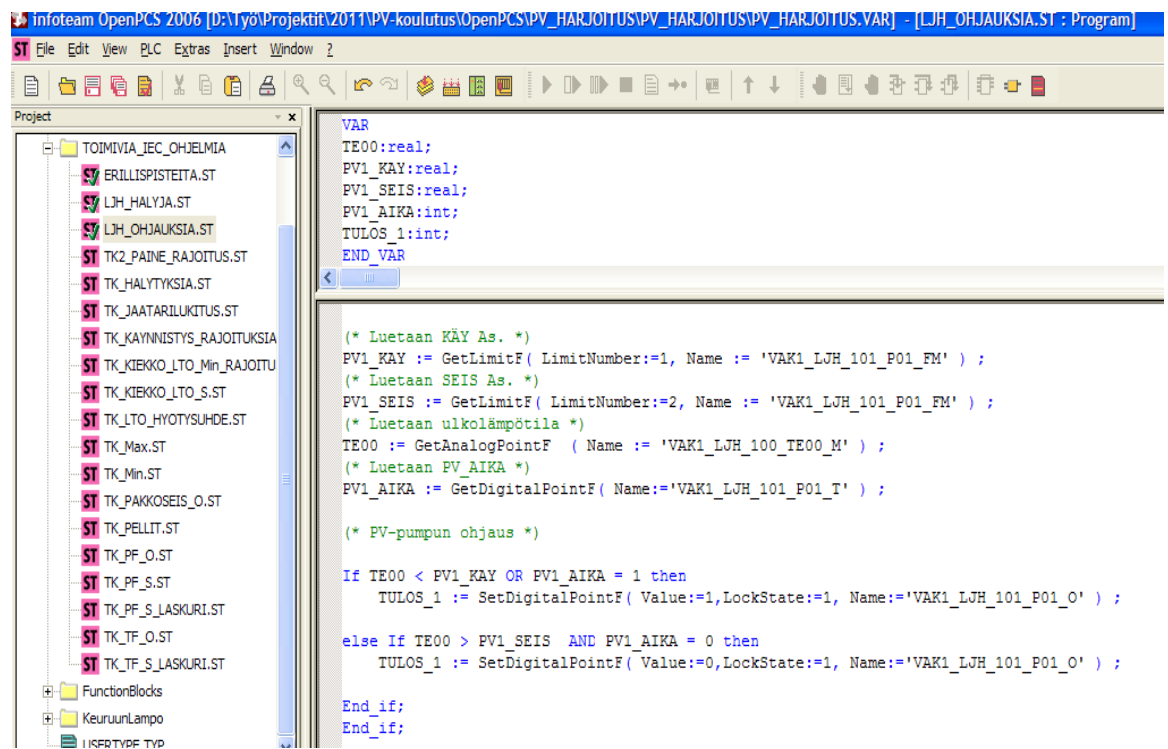
IF-lause on ehtolause, joka suoritetaan vain tietyin ehdoin. Ehtolauseessa on mahdollista tehdä normaaleja laskutoimituksia sekä vertailuja merkein: suurempi kuin, pienempi kuin, yhtä suuri kuin ja eri suuri kuin. Lisäksi jos halutaan asettaa lisää ehtoja, tarvitaan toimintoja AND, OR ja XOR. Seuraavassa on esimerkkejä ehtolauseista:

```
If TK1_T =1 and TZA_H = 0 then
```

```
If (TE00 > 5.00 and JK6_H = 0) or TE00 > 15.00 then
```

```
If TE00 < IV_KAY_RAJA or IV_AIKA = 1 then
```

Samat ehtolauseet toimivat myös ELSE IF -lauseen ehtolauseina.

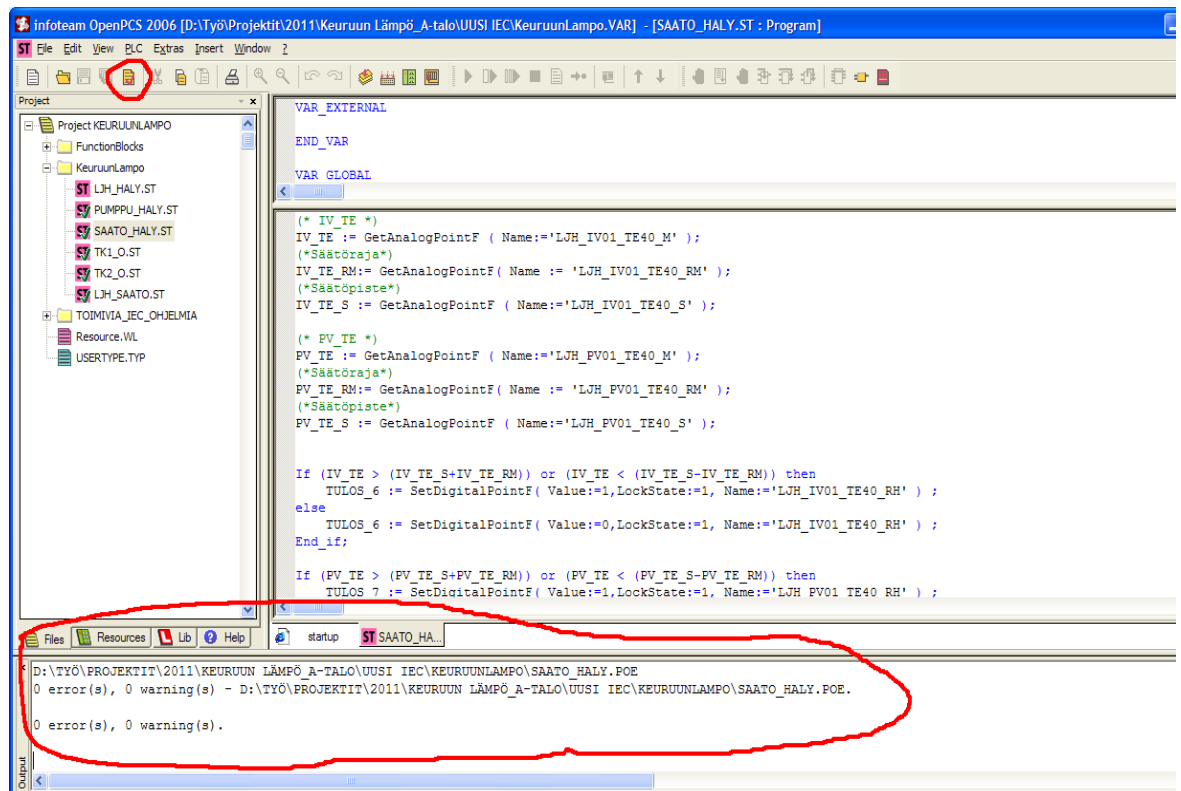


Kuvio 28. OpenPCS, IF-lause

Kuviossa 28 on täydellinen IF-lause ja ELSE IF -lause ehtolauseineen ja muuttujineen. Kuvion ohjelmassa tarkastellaan ulkolämpötilaa ja patteriverkoston pumpun ulkolämpötilarajaa. Jos ulkolämpötila on pienempi kuin pumpun käy-raja, pumppu saa käyntiluvan. Jos taas ulkolämpötila on suurempi kuin pumpun seis-raja, pumppu pysähtyy. Pumpulla on myös aikaohjelma, joka pyörittää pumppua riippumatta ulkolämpötilan arvosta.

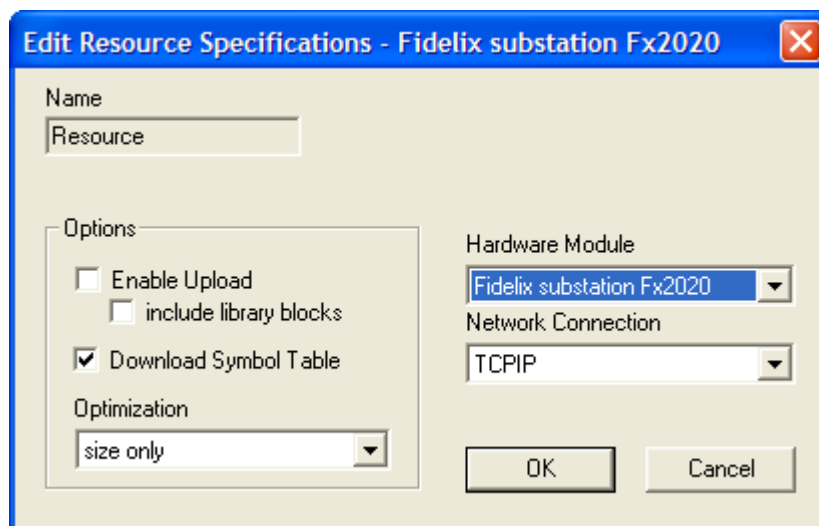
3.4.4 Ohjelman lataaminen säätimeen

Kun ohjelma on valmis, sen voi ladata säätimeen. Ensin kuitenkin tehdään syntax check, eli tarkastetaan ohjelman lauseoppi virheiden varalta. Jos virheitä löytyy, ne täytyy korjata, ennen kuin OpenPCS antaa ladata ohjelman säätimeen.



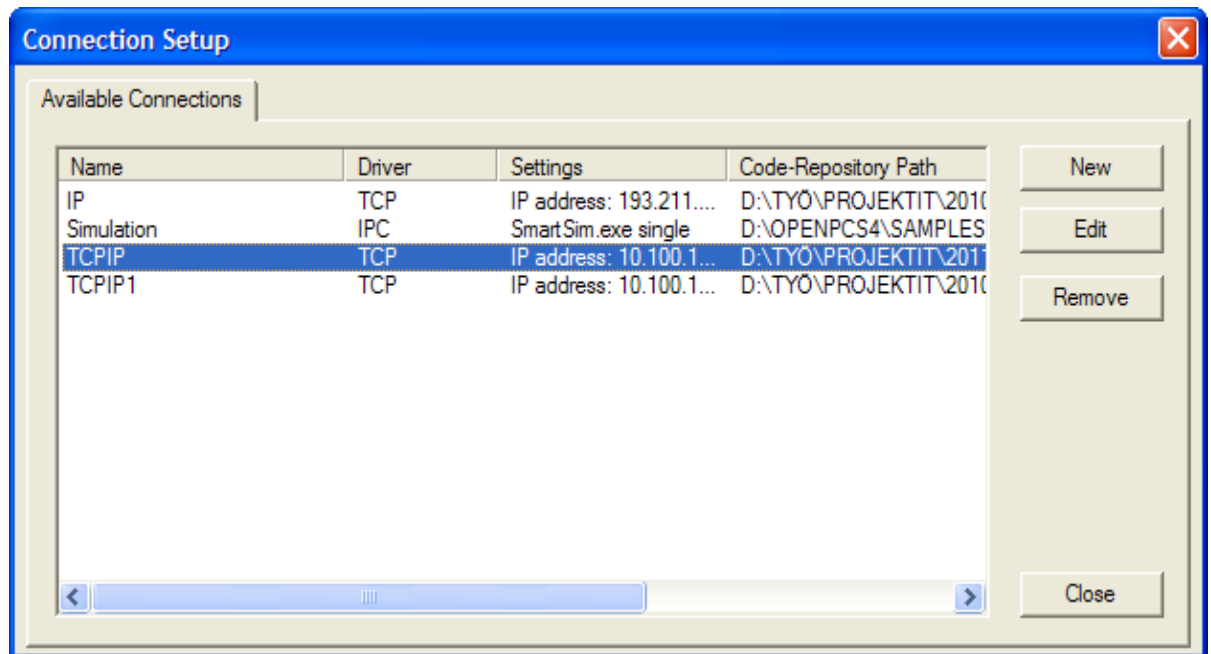
Kuvio 29. OpenPCS, Syntax Check

Kuviossa 29 on ympyröity vasemmasta yläkulmasta ”syntax check”-painike. Painike etsii ohjelmasta virheitä ja tulostaa ne sen jälkeen ala-ikkunaan. Ala-ikkunasta nähdään, onko virheitä tullut. Jos virheitä on tullut, niitä voidaan selata f4-painikkeella, jolloin osoitin hyppää ohjelmassa virhekohtaan. Kuviossa 29 niitä ei ole. Tämän jälkeen ylävalikon PLC-valikosta kohdasta ”Resource Properties” valitaan kuvion 30 mukaiset valinnat: moduliksi Fidelix substation Fx2020 ja Network Connectioniksi TCPIP.



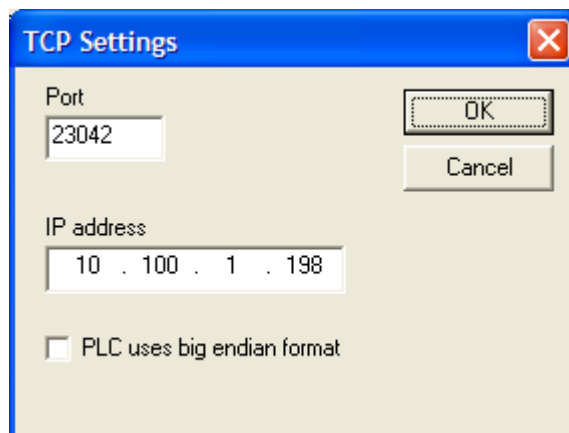
Kuvio 30. OpenPCS, Resource Properties

Kun nämä valinnat on tehty, valitaan ylävalikon PLC-valikosta "Connections"-painike.

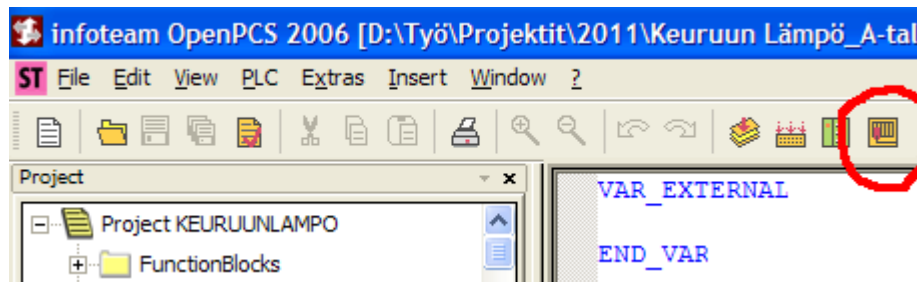


Kuvio 31. OpenPCS, Connections

Sen jälkeen valitaan listalta TCPIP, joka valittiin myös aikaisemmassa Resource Properties -ikkunassa. Jos sitä ei ole, se luodaan. Kun TCPIP on valittu, painetaan "Edit"-painiketta, ja seuraavasta ikkunasta valitaan "Settings". "TCP Settingsiin" laitetaan säätimen IP-osoite sekä valitaan portti, joka on aina 23042. Tämän jälkeen ohjelma on valmis ladattavaksi. Ennen ohjelman latausta säätimeen tarvitsee olla yhteys.



Kuvio 32. OpenPCS, TCP Settings



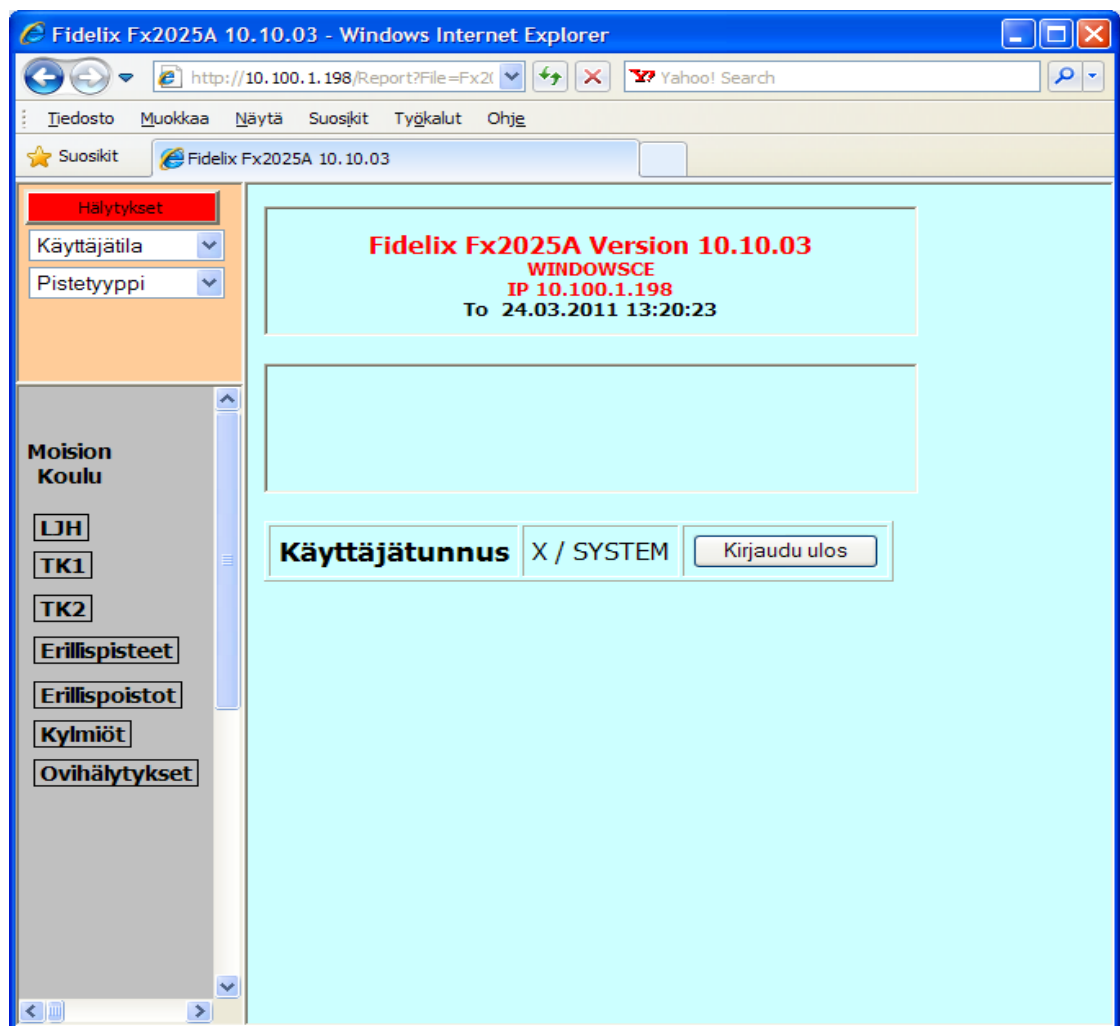
Kuvio 33. OpenPCS, Online

Painamalla kuviossa 33 ympyröityä painiketta siirrytään online-tilaan. Online-tilassa avataan yhteys ohjelmasta säätimeen. Kun yhteys on muodostettu, ja ohjelmaa on muutettu viime latauksen jälkeen, OpenPCS kysyy: haluatko ladata ohjelman. Valitaan ”Kyllä”. Jos kaikki on tehty oikein tähän mennessä, alkaa ohjelma latautua säätimeen. Säädintä ei tarvitse pysäyttää muutosten tapahtumiseksi. Muutokset otetaan käyttöön painamalla online-tilan oikealla puolella olevaa coldstart -painiketta (play-kuvio). Online-tilasta päästään pois painamalla uudestaan online-tila painiketta.

3.5 Säädin FX2025A

3.5.1 Aloitusivu

Sisäänkirjautuminen tapahtuu kirjoittamalla säätimen IP-osoite Internet Explorerin osoiteriville, minkä jälkeen selain pyytää käyttäjätunnusta ja salasanaa. Sisäänkirjautuksen jälkeen eteen avautuu aloitusivu. Aloitusivu tulee aina ensimmäisenä sisäänkirjautumisen jälkeen eteen, jos säätimelle ei ole määritetty jotakin toista ikkunaa etusivuksi. Aloitusvivulle pääsee aina ylemmästä alavetovalikosta valitsemalla ”Aloitus”. Vasemmalta vetovalikoista voidaan navigoida haluttuun paikkaan tai painaa linkkiä haluttuun FxHtmlEditorilla luotuun sivuun. Aloitusvivulla näkyy säätimen versio, säätimen IP-osoite, aika ja päivämäärä sekä se, millä käyttäjätunnuksella ja oikeuksilla käyttäjä on kirjautunut sisään.



Kuvio 34. Fidelix, Aloitus sivu

3.5.1.1 Ylempi alavetovalikko

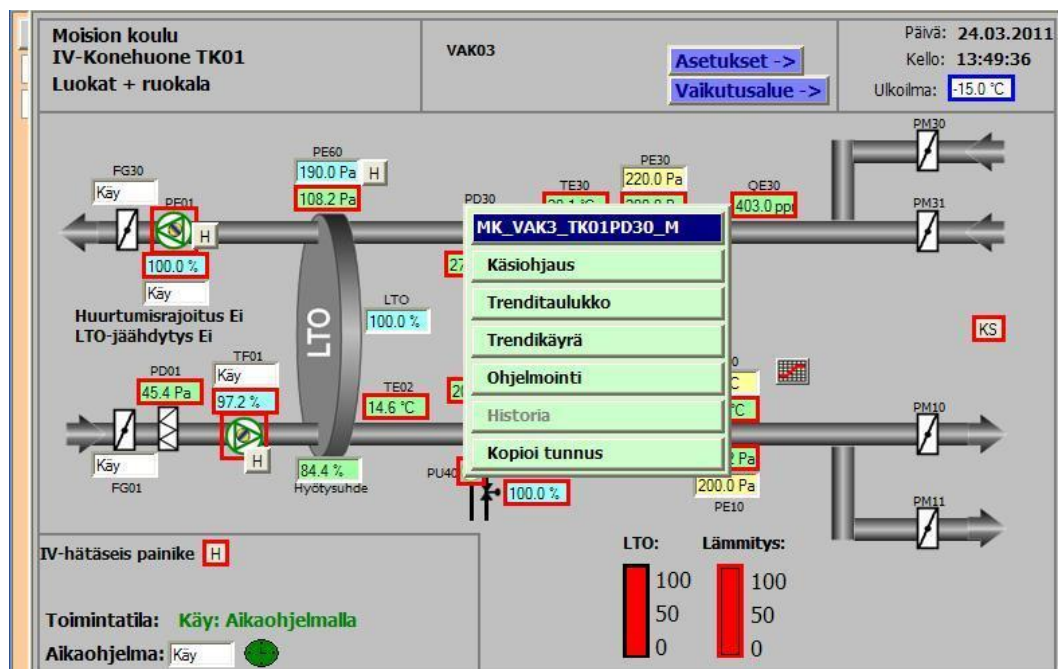
Ylemmässä alavetovalikossa ovat valittavissa kohdat Käyttäjätila, Ohjelmointi, Asetukset, Turvatoiminnot, Historia, Kulutus seuranta ja Aloitus. Käyttäjätila-kohdassa päästään katsomaan säätimeen tehtyä ohjelmaa, grafiikkaa ja pisteitä. Ohjelmointi-välilehden kautta pystytään hallitsemaan pisteitä sekä luomaan uusia pisteitä. Ohjelmointi-välilehden kautta päästään myös hallitsemaan käyttäjätietoja, hälytyksiä ja moduleita. Asetukset-välilehden kautta pystytään hallitsemaan säätimen ominaisuuksia, esim. IP-osoitetta, portteja, palomuuria ja järjestelmätietoja. Historia-välilehden kautta päästään historiaseurantaan, jossa seurataan sinne tallennettujen trendi-pisteiden arvoja ja kerätään pisteiden käyttäytymisestä tietoa. Kulutus seuranta seuraa esim. energiankulutusta. Aloituksen kautta päästään takaisin aloitussivulle.

Asetukset	Käyttäjätila	Ohjelmointi
Asetusryhmä	Pistetyyppi	Pistetyyppi
Asetusryhmä	Pistetyyppi	Pistetyyppi
Sähköposti	Asetusarvot	Indikoinnit
OPC pisteet	Indikoinnit	Ohjaukset
Portit	Ohjaukset	Mittaukset
IP osoitteet	Mittaukset	AO pisteet
Kello	AO pisteet	Muunn.Taulukot
Palomuri	Muunn.Taul.	Aikaohjelmat
Osoitemuunnos	Aikaohjelmat	Säätöpisteet
Järj.tiedot	Säätöpisteet	Hälytykset
Järj.asetukset	Hälytykset	Hälytysryhmät
Synkronointi	Kaikki	Hälytyslaitteet
Lokiasetukset	Modulit	Hälytysohjeet
SMS asetukset	Kalenteri	Tilatekstit
Amadeus asetukset	Tapahtumaloki	Modulit
	Jäätymisvaara	Modbuslaitteet
		Ouman EH105
		Ouman EH203
		Käyttäjryhmät
		Käyttäjät
		Jäätymisvaara
		Moduliversiot

Kuvio 35. Fidelix, Ylempi alavetovalikko

3.5.2 Grafiikkakuvat

Grafiikkakuvaan päästään klikkaamalla vasemmalla olevia linkkejä (esim. TK1). Nämä kuvat luodaan FxHtmlEditorilla. Grafiikkakuvassa pystytään katselemaan ja hallinnoimaan säätimen hallitsemissa prosesseissa.



Kuvio 36. Fidelix, Grafiikkakuva

Kun painetaan jotakin alkioita, johon on kiinnitetty pistetunnus, esim. lämpötilaa, avautuu kuvion 36 mukainen näkymä. Käsiohjauksella pystytään muuttamaan käsin pisteen arvoa, esim. lämpötilaa, venttiilin asentoa tai jotakin ohjausta pakotetusti päälle. Trenditaulukko ja trendikäyrä tarkoittavat samaa asiaa, ja niistä pystytään seuraamaan pisteen käyttäytymistä edellisen kahden tunnin ajalta. Trenditaulukko näyttää pisteen arvoja taulukkomuodossa ja trendikäyrä piirtää käyrän siitä, miten piste on käyttäytynyt. ”Kopioi tunnus”-kohdasta pisteen tunnus voidaan kopioida leikepöydälle. Ohjelmointipainikkeesta päästään suoraan pisteen ohjelmointivälilehdelle, jossa pisteen arvoja pystyy muokkaamaan, esim. lisäämään desimaaleja lämpötilaan tai muuttamaan mittauksen yksikköä.

3.5.3 Käyttäjätila

Käyttäjätilassa pystytään selailemaan pisteitä tyyppin mukaan, esimerkiksi valitsemalla alemmasta alavetovalikosta indikoinnit, jolloin säädin listaa kaikki digitaaliset tulot ja niiden arvot. Käyttäjätilassa pystytään myös tarkastelemaan, mitä pisteitä on kytketty mihinkin moduliin. Käyttäjätilan kalenterista näkee kalenterin ja käyttäjätilan tapahtumalokista sen, mitä muutoksia käyttäjät ovat säätimeen tehneet.

Hälytykset		
Käyttäjätila	Indikoinnit	Suodatus <input type="checkbox"/>
Moision Koulu		
To 24.03.2011 14:22:56 Indikointipisteet Päivitä		
MK_VAK2_LJ01PU40_I Indikointipiste Modulivika 3.28	Katilaverkoston pumppu	Käy Auto
MK_VAK2_LL01PU40_I Indikointipiste Modulivika 3.28	Lattialämmitysverkoston pumppu	Käy Auto
MK_VAK2_LV01PU40_I Indikointipiste Modulivika 3.28	Käyttövesiverkosto pumppu	Käy Auto
MK_VAK2_OJ01PO01_I Indikointipiste	Palo-ovi	Auki Auto

Kuvio 37. Fidelix, Indikoinnit

Hälytykset			
Käyttäjätila	Modulit		
Moision Koulu			
Moduli 03.001	Tyyppi DI	Versio V0.00	
Piste	Tunnus	Teksti	Liittimet
1	MK_VAK3_TK01PM10_H	Palopelti	1,2
2	MK_VAK3_TK01PM11_H	Palopelti	3,4
3	MK_VAK3_TK01PM30_H	Palopelti	5,6
4	MK_VAK3_TK01PM31_H	Palopelti	7,8
5	MK_VAK3_TK02PM10_H	Palopelti	9,10
6	MK_VAK3_TK02PM11_H	Palopelti	11,12

Kuvio 38. Fidelix, Modulit

Hälytykset					
Käyttäjätila	Tapahtumaloki				
Moision Koulu					
LJH	TK1	TK2			
Alku-aika	Loppu-aika	Käyttäjä	Tunnus		
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	Päivitä	
			<input checked="" type="checkbox"/> Piilota pisteteksti		
Tapahtumahistoria					
Aika	Käyttäjä	Tapahtuma	Tunnus	Yleistä	
24.03.2011 13:44:28	X (10.100.1.199)	Muutettu piste	MK_VAK2_PV01TE40_M	49.7 °C Auto -> 60.0 °C Käsin	
24.03.2011 13:44:08	X (10.100.1.199)	Hälytyksen kuittaus	MK_VAK2_PV01TE40_RH		

Kuvio 39. Fidelix, Tapahtumaloki

3.5.4 Ohjelmointi

Ohjelmointi-sivun alavalikoista nähdään eri pisteiden ja laitteiden ohjelmointimahdollisuudet ja se, miten ne on ohjelmoitu. Jokaisesta pisteohjelmointilehdestä löytyy pisteen tunnus, teksti, pisteen tasot (millä oikeuksilla piste nähdään ja millä oikeuksilla sitä voidaan ohjata ja ohjelmoida) ja se, minkä modulin missä kanavassa se sijaitsee fyysisesti. Fiktiivisten pisteiden moduli ja piste on aina 0, koska niitä ei ole fyysisesti kytketty mihinkään.

3.5.4.1 Indikointi

Indikointi-välilehdeltä saadaan näkyviin kaikkien digitaalisten tulojen ohjelmoinnit. Globaalipiste tarkoittaa, että tämä piste on haettu muusta säätimestä. Käynnistysviiveenä on se aika, jonka jälkeen piste aktivoituu, kun se on saanut signaalin käynnistyä (viive). Pysäytysviiveellä on samanlainen toiminto kuin käynnistysviiveellä, mutta käänteisenä. Jos piste on kolmitilainen, se tarkoittaa, että pisteessä on kolme tilaa. Kolmitilainen piste vie kaksi perättäistä pistettä modulissa, eli piste, joka valitaan, ja tästä seuraava piste. Kuvion 40 tapauksessa ohjaus veisi pisteet 2 ja 3 kolmitilaisena. Avautuva kosketin (NC=Normally Closed) tarkoittaa pistettä, joka aktivoituu, kun signaalin tulo katkeaa. Normaalisti digitaalitulo aktivoituu, kun piste saa signaalin. Tilatekstinä yleisesti käytetään SEIS_KÄY, mutta jos digitaalinen tulo tai ohjaus on kolmitilainen, siinä voi olla esimerkiksi SEIS_HIDAS_NOPEA. Tilateksti tarkoittaa tekstiä, jolla piste ilmoittaa, missä tilassa se milloinkin on. Kuva tarkoittaa kuvaa, jossa piste sijaitsee. Säädin tekee kuvalinkkauksen automaattisesti grafiikkakuvien perusteella.

MK_VAK2_OJ01PO01_I Indikointipiste	Palo-ovi	Auki Auto	
---------------------------------------	----------	-----------	--

Tunnus	MK_VAK2_LJ01PU40_I	Teksti	Katilaverkoston pumppu	Taso: Katselu	0
Moduli	03.028	Piste	2	Taso: Käsiohjaus	0
				Taso: Ohjelmointi	0

<input type="checkbox"/> Globaalipiste	
Käynnistyviive (sek)	0
Pysäytysviive (sek)	0
Kolmitilainen	<input type="checkbox"/>
Avautuva kosketin	<input type="checkbox"/>
Tilateksti	SEIS_KÄY
Kuva	<input type="button" value="Avaa"/> <input type="text" value="MK_ljh1.htm"/> <input checked="" type="checkbox"/> Auto
Automaattinen kuvan avaus	<input type="checkbox"/>

Kuvio 40. Fidelix, Digitaalitulon ohjelmointi

3.5.4.2 Ohjaukset

Ohjauksissa ominaisuudet ovat lähes samat kuin indikoinneissa paria poikkeusta lukuun ottamatta. Jos halutaan, että aikaohjelma ohjaa ohjausta, täytyy aikaohjelma liittää ohjauspisteeseen. Modulin oletusarvo tarkoittaa arvoa, jonka ohjaus ottaa esim. uudelleenkäynnistyksen yhteydessä. "Vanha arvo" on yleisesti käytetty oletusarvona, ja vain harvoissa poikkeustapauksissa käytetään jotain muuta.

MK_VAK2_OJ01SO01_O Ohjauspiste	Sisäovi	Auki Auto	
Tunnus	MK_VAK2_LJ01FV40_O	Teksti	Kattilan maakaasuventtiili
Moduli	00.000	Piste	0
		Taso: Katselu	0
		Taso: Käsihjaus	0
		Taso: Ohjelmointi	0
<input type="checkbox"/> Globaalipiste			
Käynnistyviive (sek)			0
Pysäytysviive (sek)			0
Aikaohjelma			
Kolmitilainen			<input type="checkbox"/>
Tilateksti			KIINNI_AUKI
Modulin oletusarvo			Vanha arvo
Kuva	<input type="button" value="Avaa"/>	MK_ljh1.htm	Auto <input checked="" type="checkbox"/>

Kuvio 41. Fidelix, Ohjauksen ohjelmointi

3.5.4.3 Analogiaohjauspisteet

Jos halutaan, että analogiaohjauspiste on aktiivinen vain tietynä aikana, liitetään siihen aikaohjelma. Muuten oletuksena aikaohjelma jätetään pois. Minimijännite ja maksimijännite tarkoittavat sitä, mitä jännitettä analogiaohjauspiste lähettää. Kuvion 42 tapauksessa 100 prosenttia on 10 voltia. Näytteenottoväli taas on vakiona 60 sekuntia. Tämä vaikuttaa trendiin otettujen pisteiden tiheyteen. Toleranssi tarkoittaa sitä, kuinka tarkasti jännitettä lähetetään ulos. Kuvion 42 tapauksessa jännitettä lähetetään 0.1 voltin tarkkuudella.

MK_VAK3_TK02PK01_A Analogiaohjauspiste	Poistopuhallin	49.0 % Ohjelma	
Tunnus	MK_VAK2_LJ01KA01_A	Teksti	Kaasukattila
Moduli	03.031	Piste	2
		Taso: Katselu	0
		Taso: Käsihjaus	0
		Taso: Ohjelmointi	0
<input type="checkbox"/> Globaalipiste			
Aikaohjelma			
Minimijännite	0.0	Maksimijännite	10.0
Näytteenottoväli (sek)	60	Toleranssi	0.1
Kuva	<input type="button" value="Avaa"/>	MK_ljh1.htm	Auto <input checked="" type="checkbox"/>
Modulin oletusarvo			Vanha arvo

Kuvio 42. Fidelix, Analogialähdön ohjelmointi

3.5.4.4 Mittaukset

Mittauksia on kahdenlaisia. Laskurit ovat pulssimittareita, esim. vesimittareita. Analogiset ovat mittareita, joista saadaan tietyllä lämpötilalla jokin analoginen arvo. Yksikkönä lämpötilalle on celcius ja näytteenottovälinä on 60 sekuntia. Näytteenottoväli voi olla myös lyhyempi. Desimaaleja voidaan lisätä ja poistaa sen mukaan, kuinka tarkka mittauksen pitää olla näytössä. Korjauksella voidaan muuttaa mittauksen arvoa koko mittausalueella, eli jos halutaan nostaa tai laskea mittaustulosta, voidaan korjaus-kohtaan sijoittaa korjausarvo. Raja-arvoille taas on monta eri käyttötarkoitusta. Niitä voidaan ohjelmoidessa käyttää rajoittavina tekijöinä, asetusarvoina tai hälytysrajoina, jotka asetetaan hälytyksien ohjelmointivälilehdestä.

MK_VAK2_LL01TE40_M Mittauspiste Modulivika 3.30	Lattialämitys menolämpö	31.8 °C Auto	
	3 - SäätöHälyRaja	5.0 °C Käsini	

Tunnus	MK_VAK2_LL01TE40_M	Teksti	Lattialämitys menolämpö	Taso: Katselu	0
Moduli	03.030	Piste	7	Taso: Käsiohjaus	0
				Taso: Ohjelmointi	0

Globaalipiste Asetusarvo
 Laskuri Analoginen

Yksikkö °C Näytteenottoväli (sek) 60
 Desimaaleja 1 Toleranssi 0.1
 Muunnostaulukko NTC20 Korjaus 0 Aikavakio (sek) 0
 Kuva MK_ljh.htm Auto

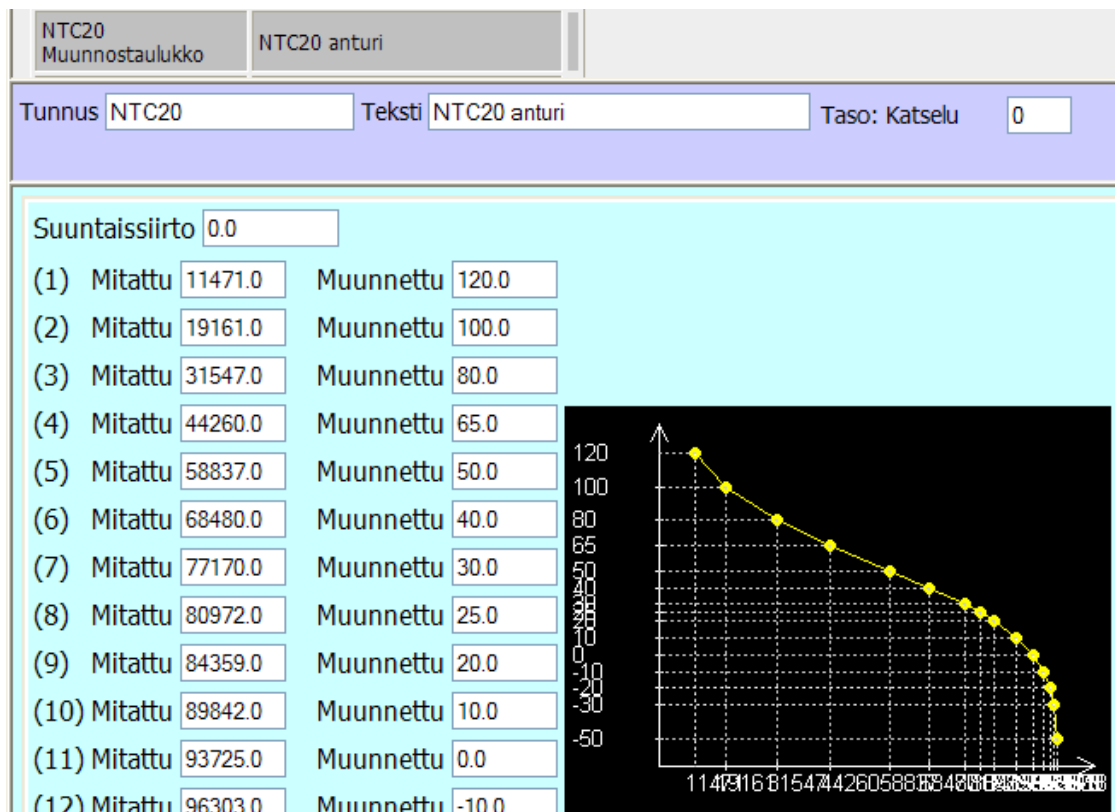
Raja-arvo 1 Nimi Arvo 0 Käsini
 Raja-arvo 2 Nimi Arvo 0 Käsini
 Raja-arvo 3 Nimi SäätöHälyRaja Arvo 5 Käsini
 Raja-arvo 4 Nimi Arvo 0 Käsini

Kuvio 43. Fidelix, Mittauksien ohjelmointi

3.5.4.5 Muunnostaulukot

Muunnostaulukkoja on eri mittarityypeille. Kuvion 43 anturi on NTC20-anturi. Yleensä NTC10-anturit ovat yleisiä lämpötilamittauksissa. Mitauksissa esiintyy myös esim. 0–10 voltia → 0–2 000 yksikköä muunnostaulukoita, eli 10 voltia on 2 000 yksikköä. Painemittareissa tällaiset muunnostaulukot ovat yleisiä.

Muunnostaulukoissa muunnetaan jokin arvo (yleensä mitattu arvo) joksikin toiseksi arvoksi. Kuvassa 44 on NTC20-muunnostaulukko. Kun saadaan mitattu vastusarvo lämpötila-anturilta, täytyy se muuntaa luettavaan muotoon celcius -asteiksi. 0–10 voltia mittauksissa 10 voltia on mitattuna 100 000 yksikköä. Esimerkiksi 0–10 voltia → 0–2 000 muutettuna muunnostaulukkaan olisi Mitattu-kohdan arvo 100 000, muunnettuna 2 000.



Kuvio 44. Fidelix, Muunnostaulukoiden ohjelmointi

3.5.4.6 Aikaohjelmat

Aikaohjelmia tarvitaan esimerkiksi tuloilmakoneisiin ja pumppujen kesäpyöräytyksiin. Aikaohjelmassa tilatekstillä voi olla useampia asentoja (SEIS_HIDAS_NOPEA). Tiloja

on maksimissaan kuusi. ”Kopioi”-painikkeella voidaan esim. kopioida maanantaipäivä kaikkiin päiviin, jolloin jokaista päivää ei tarvitse määritellä erikseen.

MK_VAK3_TK01_T Aikaohjelma	Tuloilmakone	Käy Auto	
Tunnus	MK_VAK2_OJ01UO01_	Teksti	Oven aikaohjelma
Tilateksti	KIINNI_AUKI	Käytä erikoispäiviä	Kyllä <input type="button" value="Kopioi"/>
Ma	07:55 Auki	15:15 Kiinni	
Ti	07:55 Auki	15:15 Kiinni	
	07:55	15:15	

Kuvio 45. Fidelix, Aikaohjelminen ohjelmointi

3.5.4.7 Hälytykset

Hälytysviive ja poistumisviive tarkoittavat samaa kuin käynnistysviive ja pysäytysviive indikoinneissa (luku 3.5.4.1). Prioriteetti on hälytyksen tärkeys. Avautuvaa kosketinta käytetään esim. paloilmoitinkeskuksessa, joissa hälytys tulee, kun signaali katkeaa. ”Tallenna lokiin”-kohta tarkoittaa, että hälytyksen aiheuttamat hälytykset löytyvät myöhemmin hälytyslokista. Hälytysryhmällä tarkoitetaan sitä, kenelle hälytys menee hälytyksen sattuessa. Hälytysryhmä voidaan myös jättää tyhjäksi, jos halutaan hälytyksen pysyvän säätimessä. Hälytyksiä varten on luotu hälytys-tilateksti, joka ilmoittaa tekstinä hälytyksen tilan (normaali tai hälytys).

MK_VAK2_SJ01VV01_RH Hälytyspiste Prioriteetti:2	Lastauslaiturin sulatus	Normaali Auto Kuitattu	Ryhmä: VIKA
--	-------------------------	---------------------------	-------------

Tunnus	MK_VAK2_PV01TE40_F	Teksti	Patteriverkosto menolämpö	Taso: Katselu	0
Moduli	00.000	Piste	0	Taso: Käsiöhaus	0
				Taso: Ohjelmointi	0

Globaalipiste

Hälytysviive (sek)	1800
Poistumisviive (sek)	0
Prioriteetti	2
Alerta hälytysnumero	1
Avautuva kosketin	<input type="checkbox"/>
Tallenna lokiin	<input checked="" type="checkbox"/>
Hälytysryhmä	HUOLTO
Tilateksti	HÄLYTYS

Kuvio 46. Fidelix, Hälytyspisteen ohjelmointi 1

Tilateksti	HÄLYTYS						
Kuva	Avaa	MK_jjh.htm	Auto	<input checked="" type="checkbox"/>			
Normaali	<input checked="" type="radio"/>	Rajahälytys	<input type="radio"/>	Ristiriitahälytys	<input type="radio"/>	Modulivikahälytys	<input type="radio"/>
Estopisteen tunnus		Estotila	0				
Hälytysohje	Ryhmän mukaan						

Kuvio 47. Fidelix, Hälytyspisteen ohjelmointi 2

Säätimessä pystytään ohjelmoimaan hälytyksiä lähes samoin kuin IEC-koodissa. Kuviossa 47 näkyy, mitä vaihtoehtoja hälytyspisteen ohjelmoinnille on. Normaali hälytys tarkoittaa IEC-koodissa tehtyä hälytystä tai fyysistä hälytystä. Rajahälytys tarkoittaa mittauksille ohjelmoitavaa hälytystä, joka aktivoituu, jos pistetunnus (asetettava) on pienempi kuin (tai suurempi kuin) sen mittauspisteen raja-arvo (katso kuviosta 48 mittauksien raja-arvo). Raja-arvon jälkeen voidaan laittaa JA/TAI saman mittauksen

suurempi/pienempi toinen raja-arvo. Kuollut alue tarkoittaa aluetta, jolloin hälytys pysyy aktiivisena, vaikka mittaus on sallitulla alueella hälytyksen jälkeen. Jos estopiste on yhtä suuri kuin estotila, hälytystä ei tapahdu (pois käytöstä, jos alue jätetään tyhjäksi).

The screenshot shows the configuration interface for a limit alarm (Rajahälytys) in the Fidelix system. The interface is light blue and contains the following elements:

- Four radio buttons for alarm types: Normaalii (unselected), Rajahälytys (selected), Ristiriitahälytys (unselected), and Modulivikahälytys (unselected).
- A text input field for "Pistetunnus" (alarm point ID).
- A dropdown menu set to "Pienempi kuin" (smaller than), followed by a "Raja" (limit) dropdown set to "1" and a "..." dropdown.
- A text input field for "Kuollut alue +/-" (dead zone) set to "0.5", and a checked checkbox for "Nollaa mittaus" (zero measurement).
- A section containing a text input for "Estopisteen tunnus" (bypass point ID) and a text input for "Estotila" (bypass level) set to "0".
- A dropdown menu for "Hälytysohje" (alarm instruction) set to "Ryhmän mukaan" (according to group).

Kuvio 48. Fidelix, Hälytyspisteen ohjelmointi, Rajahälytys

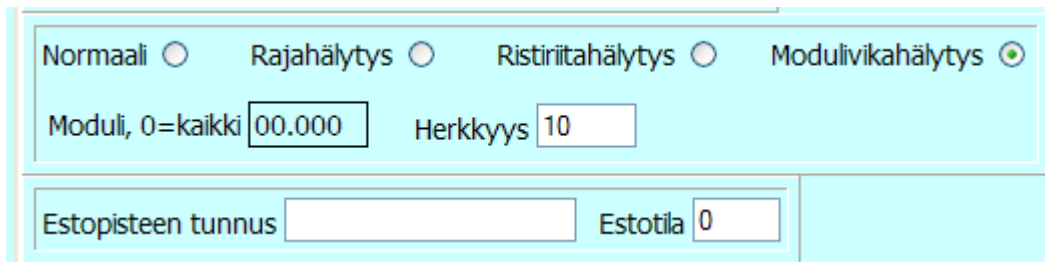
Ristiriitahälytys tulee, kun kaksi aseteltua pistetunnusta, esim. puhaltimen ohjaus ja käyntitila, eivät ole yhtä suuria. Pistetunnukset ovat aseteltavia.

The screenshot shows the configuration interface for a conflict alarm (Ristiriitahälytys) in the Fidelix system. The interface is light blue and contains the following elements:

- Four radio buttons for alarm types: Normaalii (unselected), Rajahälytys (unselected), Ristiriitahälytys (selected), and Modulivikahälytys (unselected).
- Two text input fields for "Pistetunnus" (alarm point ID).
- A section containing a text input for "Estopisteen tunnus" (bypass point ID) and a text input for "Estotila" (bypass level) set to "0".

Kuvio 49. Fidelix, Hälytyspisteen ohjelmointi, Ristiriitahälytys

Modulivikahälytys on hälytys, joka aktivoituu, jos jokin moduli antaa virhesanomia yhtäjaksoisesti herkkyyden verran (kuviossa 50 herkkyys on 10). Tällöin hälytyspiste hälyttää, jos määriteltyyn moduliin ei saada yhteyttä. Se tarkoittaa yleensä sitä, että väylä on jostakin syystä poikki tai moduli on rikkoutunut.



Normaali Rajahälytys Ristiriitähälytys Modulivikahälytys

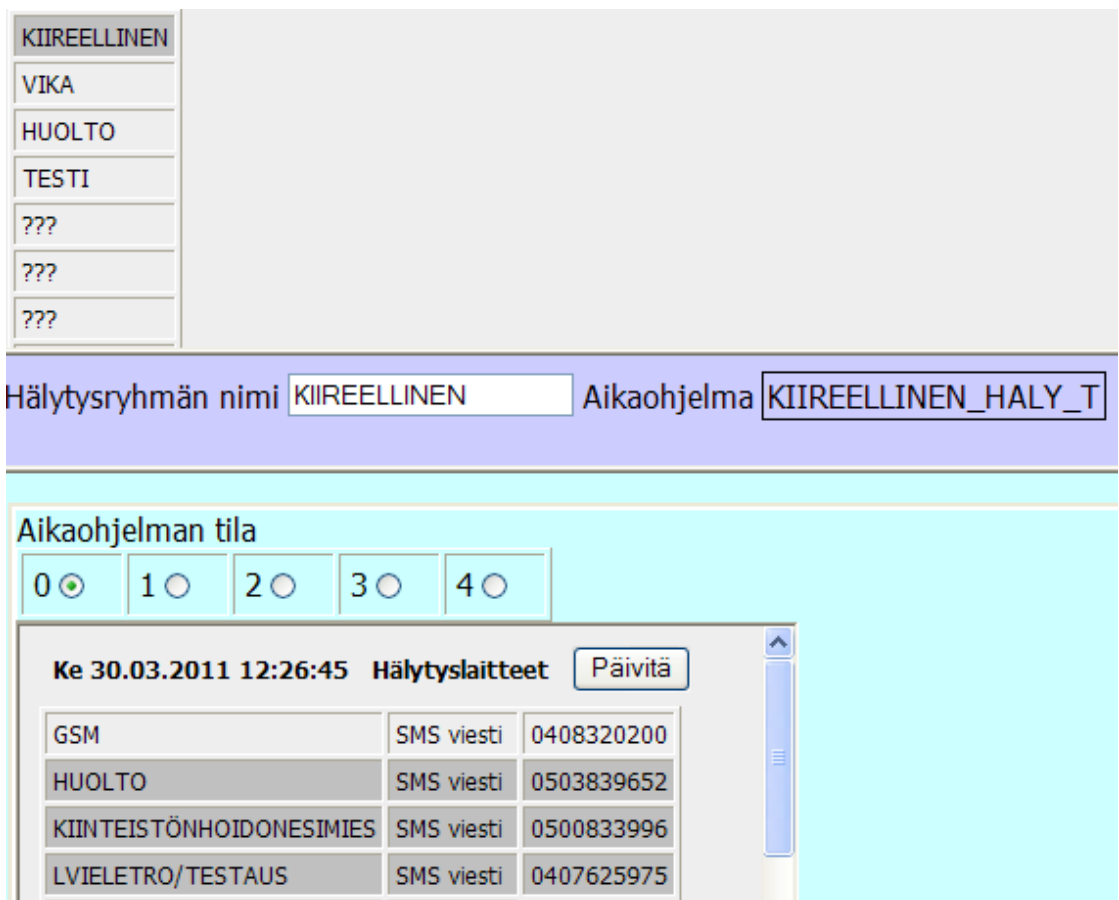
Moduli, 0=kaikki Herkkyyys

Estopisteen tunnus Estotila

Kuvio 50. Fidelix, Hälytyspisteen ohjelmointi, Modulivika

3.5.4.8 Hälytysryhmät

Hälytysryhmä voidaan asettaa hälytyspisteelle, jos piste halutaan jatkohälytyksiin. Ensin valitaan haluttu hälytysryhmä. Sen jälkeen valitaan hälytyslaitteista, mihin puhelinnumeroihin hälytyksen halutaan menevän. Tummanharmaa valinta on jatkohälytyksen siirtymisen merkki kyseiselle hälytyslaitteelle. Hälytysryhmälle voidaan myös määrittää oma aikaohjelma. Jos aikaohjelma on määriteltynä ja valittuna, hälytykset siirtyvät vain sen sallimissa rajoissa.



KIIREELLINEN
VIKA
HUOLTO
TESTI
???
???
???

Hälytysryhmän nimi Aikaohjelma

Aikaohjelman tila

0 1 2 3 4

Ke 30.03.2011 12:26:45 Hälytyslaitteet

GSM	SMS viesti	0408320200
HUOLTO	SMS viesti	0503839652
KIINTEISTÖNHOIDONESIMIES	SMS viesti	0500833996
LVIELETRO/TESTAUS	SMS viesti	0407625975

Kuvio 51. Fidelix, Hälytysryhmät

3.5.4.9 Hälytyslaitteet

Hälytyslaitteelle annetaan nimi ja tyyppi. Useimmin käytetty hälytyslaitteen tyyppi on GSM-tekstiviesti hälytysrobotin kautta. Kuvion 52 Puhelinnumero-kohtaan asetetaan kohdenumero ja lähetettävistä tapahtumista valitaan ne, joista halutaan antaa jatko-hälytys. Esimerkiksi Tila 1 tarkoittaa hälytyksen aktivoitumista.

Ke 30.03.2011 12:31:27 Hälytyslaitteet Päivitä

GSM	SMS viesti	0408320200
HUOLTO	SMS viesti	0503839652
KIINTEISTÖNHOIDONESIMIES	SMS viesti	0500833996
LVIELETRO/TESTAUS	SMS viesti	0407625975
??	Sähköposti	
??	Sähköposti	
??	Sähköposti	

Hälytyslaitteen nimi Tyyppi

Puhelinnumero

SMS komennot

Valitse lähetettävät tapahtumat

Kuittaus Lukitustila

Tila 0 Tila 1 Tila 2 Tila 3 Tila 4

Tila 6 Tila 7 Tila 8 Tila 9 Tila 10

Kuvio 52. Fidelix, Hälytyslaitteet

3.5.4.10 Tilatekstit

Tilatekstit ovat digitaalisten pisteiden eli digitaalitulojen ja ohjauksien indikointitekstejä. Esimerkiksi aikaohjelmalla voi olla kolme tilaa: seis, hidas ja nopea. Uusia tilatekstejä on mahdollista luoda mielensä mukaan. Tilatekstejä voidaan kiinnittää pisteisiin, joiden tilateksti vaihtuu niiden saamien tilojen mukaan

Ke 30.03.2011 12:33:49 Tilatekstit Päivitä

SEIS_KÄY
SEIS_HIDAS_NOPEA
KIINNI_AUKI
HÄLYTYS
PÄIVYSTYS
POIS_PÄÄLLÄ
??

Tekstiryhmän nimi SEIS_HIDAS_NOPEA

Arvo	0	Teksti	Seis
Arvo	1	Teksti	Hidas
Arvo	2	Teksti	Nopea

Kuvio 53. Fidelix, Tilatekstit

3.5.4.11 Moduilit

Moduilit-sivulta näkee modulien tilan. Tällä sivulla myös määritellään uudet moduilit. Moduleita ei tarvitse etsiä käsityönä, vaan säädin voidaan laittaa suorittamaan haku itse painamalla "Etsi moduilit"-painiketta. Tällöin säädin hakee kaikki väylässä kiinni olevat moduilit ja päivittää ne listaan.

Portti.Moduli 03.010 Tyyppi DO Versio V0.00 Kommunikaatiovirhe 8-Channel Digital Output Module

Porttinumero 3 Modulin osoite 2 Tyyppi DO

Kierrosaika (ms)	610	TCP/UDP moduli	<input type="checkbox"/>
Sanomia	1364388		
Virheitä nyt	543064		
Virheitä enimmillään	2419648		
Virheitä yhteensä	3198418	Debug	<input type="checkbox"/>
Vastaanoton odotusaika, ms	80		
Oletusarvon viive, sek	120	Vahtikoiran viive, sek	0

Tyhjennä kaikkien modulien virhelaskurit

Etsi moduilit

Kuvio 54. Fidelix, Modulien ohjelmointi

3.5.5 Historia

Historia-välilehdellä pystytään tallentamaan pisteiden trenditietoja pidemmäksi aikaa kuin säädin normaalisti tallentaa. Historia-sivuille syötetään pistetunnus sekä määritellään näytteenottoväli ja näytteiden maksimilukumäärä. Tämän jälkeen tallennetaan piste, ja säädin alkaa automaattisesti tallentaa pisteen tietoja tietokantaan.

Pistetunnus	Pisteen teksti	Energia mittaus	Näytteenotto väli	Näytteet maks.	Päivät maks.	T
MK_VAK2_KYL1TE20_M	Pakastin D71	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL2TE20_M	Liha D72	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL3TE20_M	Vihannes D73	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL4TE20_M	Maito D67	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL5TE20_M	Kylmäkeititti D70	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL6TE20_M	Valmisruoka D69	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL7TE20_M	Pakkauskylmä	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL8TE20_M	Pakastin D63	Ei	1 Tuntia	17568	--	0
MK_VAK2_LJ01KA01_A	Kaasukattila	Ei	1 Minuuttia	5000	--	0

Kuvio 55. Fidelix, Historia

3.5.5.1 Historiaryhmän luominen

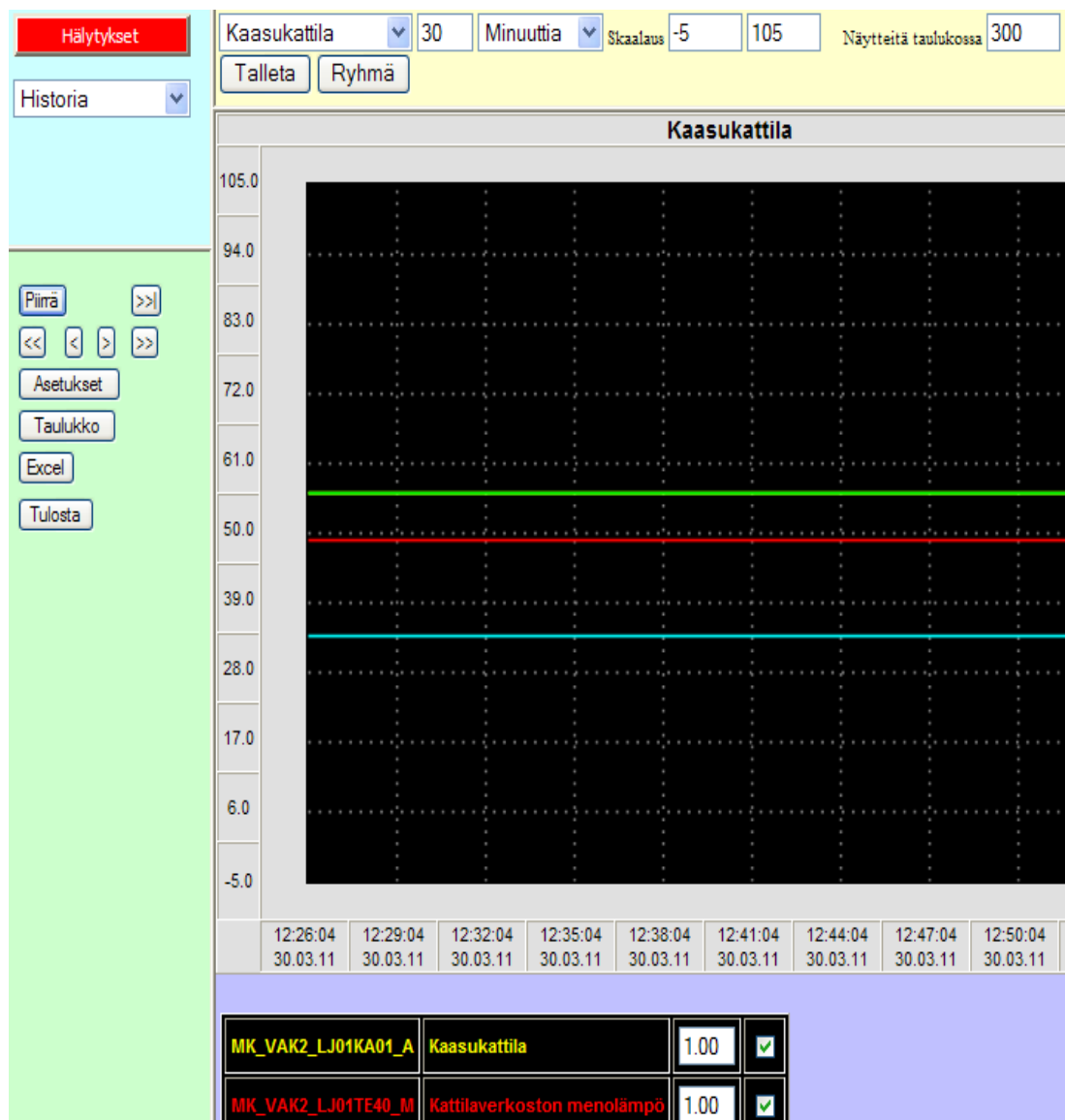
Jotta pisteiden tietoja voidaan katsella, ne täytyy kiinnittää johonkin ryhmään painamalla "Ryhmä"-painiketta. Kuvassa 56 on esimerkiksi annettu ryhmälle nimi ja ryhmään on lisätty venttiili, säätöpiste ja kaksi mittausta. Jotta pisteitä voidaan lisätä ryhmään, on ensin valittava pisteen paikka, ja sen jälkeen paikkaan piste alemmasta ikkunasta. Pisteitä voi myös vaihtaa ja poistaa. Lopuksi painetaan "Uusi ryhmä"-painiketta.

Pistetunnus	Pisteen teksti	Energia mittaus	Näytteenotto väli	Näytteet maks.	Päivät maks.	T
MK_VAK2_KYL1TE20_M	Pakastin D71	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL2TE20_M	Liha D72	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL3TE20_M	Vihannes D73	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL4TE20_M	Maito D67	Ei	1 Tuntia	17568	--	0
MK_VAK2_KYL5TE20_M	Kylmäkeitin D70	Ei	1 Tuntia	17568	--	0

Kuvio 56. Fidelix, Historiaryhmit

3.5.5.2 Piirto

Kun ryhmä halutaan piirtää, valitaan ryhmä yläriviltä ja painetaan "Piirrä"-painiketta. Yläriviltä valitaan myös piirtoaika (kuinka pitkältä ajanjaksolta halutaan näytteet) sekä skaalaus. Taulukon alapuolelta nähdään selitykset viivoille. Jos halutaan tarkastella vain osaa ryhmän pisteistä, otetaan rasti näytteen pistetunnuksen vierestä pois ja painetaan "Piirrä"-painiketta uudestaan.



Kuvio 57. Fidelix, Historian piirto

4 POHDINTA

4.1 Tulokset

Opinnäytetyön soveltuvuutta koulutuskäyttöön testattiin yrityksen nuoremmalla ohjelmoijalla, joka ei juuri ollut perehtynyt ohjelmointityökaluihin. Tarkoituksena hänellä oli ohjelmoida ohjelmointiprojekti alusta alkaen itsenäisesti lyhyen koulutussuuden jälkeen, jossa käytiin läpi Fidelix-säätimen ohjelmointiin tarvittavat työ-

kalut. Nuorempi ohjelmoija onnistui projektissa hyvin. Hän tarvitsi vain vähän apua projektin aloituksessa, pisteiden luomisessa (PointGen) ja kuvien piirrosta (FxHtmlEditor). Ohjelmointityökalu kerrattiin läpi kerran ohjelmointiin perehtymisen jälkeen. Säädinohjelmoinnissa hän tarvitsi hieman apua tästä oppaasta jääneiden ominaisuuksien selventämiseen. Häneltä saatu palaute tiivistetään seuraavissa kappaleissa ja niiden perusteella opinnäytetyön aiheena ollutta Fidelix-ohjelmointiopasta päivitetään.

4.1.1 Kehitettävät osa-alueet

Seuraavassa luettelossa käydään läpi kohdehenkilöltä tulleet kehitysehdotukset:

- Koehenkilö kaipasi enemmän esimerkkejä valmiista projekteista opinnäytetyöhön.
- Säädinohjelmoinnista oli jäänyt pois säätöpisteen ohjelmoinnin läpikäyminen.
- OpenPCS-ohjelmaan tarvitaan laajempaa perehdyttämistä ja lisää ohjelmia esimerkkeihin. Oppaassa olisi käytävä lävitse useampia osakokonaisuuksia ja lisättävä uusia ohjelmia, kuten viiveohjelma.
- Opinnäytetyön kuvat voisi päivittää uudempia projekteja kuvaavaksi. Tällä hetkellä kuvat ovat vuoden 2010–2012 projekteista, joten ne eivät täysin vastaa viimeisimpiä ohjelmistoversioita.

Opinnäytetyön kehitysehdotuksien perusteella parannetaan työn käytännön osuutta. Tarkoitus on myös muokata opinnäytetyö enemmän demo-projektin kaltaiseksi, eli työssä luotaisiin projekti. Opas olisi voinut alun alkaen voinut olla projektimuotoinen, mutta joitakin osioita olisi saattanut jäädä projektimuotoisesta oppaasta pois. Nykyisellään oppaassa käydään läpi pääpiirteittäin ohjelmoinnin vaiheet, mutta koulutettavalla saattaa olla vaikea tehdä projektia oppaan perusteella ilman koulutusta, jos opas ei ole projektimuotoinen. Nämä kehitysehdotukset otetaan huomioon ja ne sisällytetään oppaan seuraavaan versioon. Seuraavaan versioon myös päivitetään kuvat niiltä osin kun ne ovat vanhentuneita.

Jatkossa oppaan kehittämiseksi sitä käytetään LVI-Elektro Oy:n nuoremmalla ohjelmoijalla uudestaan, ja tästä kerätään palaute. Opinnäytetyötä on tarkoitus myös kehittää aina kun tulee jostakin ohjelmistosta tai Fidelixin säätimestä uusi versio.

4.1.2 Työn onnistuminen

Kokonaisuutena opinnäytetyö täyttää alussa määritellyt vaatimukset. Ohjelmointiopas oli opinnäytetyönä haastava. Kun itse lukee opasta, on sen tyyli ajoittain töksähtelevä, ja se on välillä melko listamainen. Kuitenkin oppaan tarkoitus on olla tukemassa ohjelmointia ikään kuin tarkistuslistana (jos jokin asia unohtuu), ja tässä opas onnistuu. Tulevaisuudessa opasta laajennetaan palautteen mukaan ja lista- maista selostusta pyritään muokkaamaan sujuvampaan muotoon.

LÄHTEET

InTech, 2012. IEC 61131-3 industrial control programming standard advancements.

Viitattu 9.11.2013

<http://www.isa.org/InTechTemplate.cfm?template=/ContentManagement/ContentDisplay.cfm&ContentID=91467>

Peltomäki, J. 1999. KODAA!, Helsinki, Teknolit Oy

PLCopen. IEC 61131-3: A standard programming resource. Viitattu 8.11.2013

http://www.plcopen.org/pages/pc2_training/downloads/downloads/new_intro_iec.doc

Real Time Automation, 2009. Viitattu 8.11.2013

<http://www.rtaautomation.com/iec61131-3/>