

KARELIA-AMMATTIKORKEAKOULU
Sähkötekniikan koulutusohjelma

Harri Karppinen

ILMASTOINTIKONEEN VALVOMO

Opinnäytetyö
Toukokuu 2016



OPINNÄYTETYÖ
Toukokuu 2016
Sähkötekniikan koulutusohjelma

Karjalankatu 3
80220 JOENSUU
+358 13 260600

Tekijä(t)
Harri Karppinen

Nimeke
Ilmastointikoneen valvomo

Toimeksiantaja
Karelia-amk

Tiivistelmä

Opinnäytetyön aiheena oli suunnitella ja toteuttaa Karelia-ammattikorkeakoululle opetus-
käyttöön rakennettuun ilmastointikoneeseen valvomo-ohjelmisto ja yhdistää se ohjelmoi-
tavalle logiikalle OPC -palvelimen kautta. Lisäksi ilmastointikoneen jäähdytysjärjestel-
mälle piti lisätä valmistajan ohjeiden mukaiset päälle - pois kierto rajoitukset, ja estot.
Rajoitukset ja estot tarvittiin, ettei jäähdytystä voida käynnistää jos ilmastointikoneen pu-
haltimet eivät ole päällä ja ulkoilmapellit eivät ole auki.

Opinnäytetyö toteutettiin tutkimalla ilmastointikoneiden valvomo-ohjelmien konventioita ja
ohjelmia ja tutustumalla muihin tarvittaviin materiaaleihin. Opinnäytetyössä tarvittavia oh-
jelmia olivat CoDeSys -ohjelmointiympäristö, CoDeSys OPC Server ja Wonderware In-
dusoft Web Studio.

Työn ansiosta ilmastointikoneita on helpompi käyttää opiskelijoiden harjoituksissa, sillä
selkeä valvomo-ohjelmisto helpottaa osaltaan ilmastointikoneeseen tutustumista. Tehty
materiaali yhteyden muodostamisesta OPC-palvelimen välityksellä helpottaa vastaavien
projektien läpivientiä tulevaisuudessa. Tehdyt muutokset ilmastointikoneen ohjelmaan es-
tävät jäähdytysjärjestelmän ylikuumentumisen vääränlaisen käytön seurauksena.

Kieli
suomi

Sivuja 33
Liitteet 3

Asiasanat

Ilmastointikone, käyttöliittymä, OPC



THESIS
May 2016
Degree Programme in
Electrical Engineering
Karjalankatu 3
80220 JOENSUU
FINLAND
+358 13 260600

Author (s)
Harri Karppinen

Title
Supervisory Control and Data Acquisition System for Air Conditioner

Commissioned by
Karelia University of Applied Sciences

Abstract

The subject of this thesis was to design and implement SCADA for HVAC unit at Karelia University of Applied Sciences. SCADA program should be connected to PLC via OPC – server. In addition, safety measures were added to PLC program of the HVAC cooling system to meet the cooling unit manufacturer’s instructions of on/off cycle rate. Terms were also added to prevent the starting of cooling system if the fans are not operating and dampers are not opened to prevent overheating of the cooling system.

The starting point for the thesis was studying the conventions of HVAC HMI’s and programs that are used, from literature, other available materials and manuals from developers. Programs that were used in this thesis are CoDeSys, CoDeSys OPC server as OPC server and Wonderware Indusoft Web Studio as SCADA program.

As a result of this thesis, the HVAC is easier to implement in students’ laboratory assignments. Clear and easy to understand SCADA helps getting started to understand the operation of the HVAC unit. The material which was made about the OPC server communication helps implementing the OPC server to other project in the future. The changes that were made on hte HVAC program will prevent damages to the cooling unit from misuse.

Language

Finnish

Pages 33

Appendices 3

Keywords

SCADA, HVAC, OPC

Sisältö

1	Johdanto	6
2	Käyttöliittymän suunnittelu	7
2.1	Suunnittelun lähtökohdat	7
2.2	Käytettävyys	9
3	OPC-standardi	10
4	Jäähdytysjärjestelmä ilmastointikoneessa	12
5	Tavoitteet	13
6	Käytetyt ohjelmat	13
6.1	Ohjelmien valinta	13
6.2	CodeSys OPC Server	14
6.3	Wonderware Indusoft Web Studio 8.0	16
7	Toteutus	18
7.1	OPC-yhteys	18
7.2	Käyttöliittymä	20
7.2.1	Symboleiden luominen Indusoft Web Studiossa	21
7.2.2	Päänäkymä	23
7.2.3	Trendikäyrät	25
7.2.4	Hälytykset	26
7.2.5	Ongelmat	29
7.3	Ilmastointikoneen ohjausohjelmaan tehdyt muutokset	29
8	Tulokset	31
9	Pohdinta	32
	Lähteet	33

Liitteet

- Liite 1 Ilmastointijärjestelmän ilmavirtojen värit
- Liite 2 Ilmastointilaitteiden ja varusteiden piirrosmerkkejä
- Liite 3 CoDeSys OPC Server määrittely

Käsitteet, lyhenteet ja määritelmät

Client	Asiakasohjelma
HMI	Human-Machine Interface, käyttöliittymä.
IWS	Indusoft Web Studio on kehitetty erityisesti SCADA HMI sovelluksien, näyttötaulujen sekä OEE käyttöliittymien kehittämissympäristöksi ja suorituksen aikaiseksi ohjelmaksi
Käyttöliittymä	Laitteen tai ohjelmiston osa jota kautta käyttäjä käyttää tuotetta
Localhost	Tietokoneverkoissa localhostilla tarkoitetaan tietokoneita jota ollaan käyttämässä.
LTO	Ilmastointilaitteissa oleva lämmönalteenotto.
PLC	Programmable logic controller, Ohjelmoitava logiikka,
OEE	Overall Equipment Effectiveness mittaa prosessin tehokkuutta.
OPC	<u>OLE</u> (<u>O</u> bject <u>L</u> inking and <u>E</u> mbedding) for <u>P</u> rocess <u>C</u> ontrol on OPC Foundationin ylläpitämä ja kehittämä avoimen tiedonsiirron standardi.
SCADA	Supervisory Control And Data Acquisition, suomeksi valvomo-ohjelmisto
Tag	Tagille ei ole vakiintunut suomessa mitään yleisesti käytettävää nimeä, nimitys riippuu käyttökohteesta. Tagilla voidaan tarkoittaa muun muassa muuttujaa, tai sen nimeä.
TCP/IP	Transmission Control Protocol / Internet Protocol, on tiedonsiirrossa käytetty protokolla, jossa IP-protokolla vastaa päätelaitteiden osoittamisesta ja pakettien reitittämisestä verkossa, ja TCP-protokolla vastaa kahden päätelaitteen välisestä tiedonsiirtoyhteydestä, pakettien järjestämisestä sekä hävinneiden pakettien uudelleen lähettämisestä

1 Johdanto

Sain tehtäväksi Karelia-ammattikorkeakoululta suunnitella ja toteuttaa rakenteilla olevaan taloautomaatiolaboratorion KAIR harjoitusilmastointikoneeseen valvomo-ohjelmiston, sen yhdistämisen ilmastointikoneen ohjelmoitavalle logiikalle ja käsikäytölle turvarajojen lisäämisen jäähdytyskompressorille. Opinnäytetyöni sisältää kolme pienempää kokonaisuutta. Ilmastointikoneen rakentaminen on aloitettu keväällä 2015, ja sen mekaaninen rakenne on jo valmiina. Ilmastointikoneetta olisi tarkoitus käyttää apuna opetuksessa ja rakennusautomaatiolaboratorion harjoituksissa.

Valvomo-ohjelman tulisi olla selkeä ja tarkoitukseen soveltuva, jotta sitä voitaisiin käyttää hyödyksi opetuksessa ja opiskelijoiden harjoituksissa. Sen tulisi myös olla alalla vallitsevien valvomo-ohjelmistojen konventioiden mukainen.

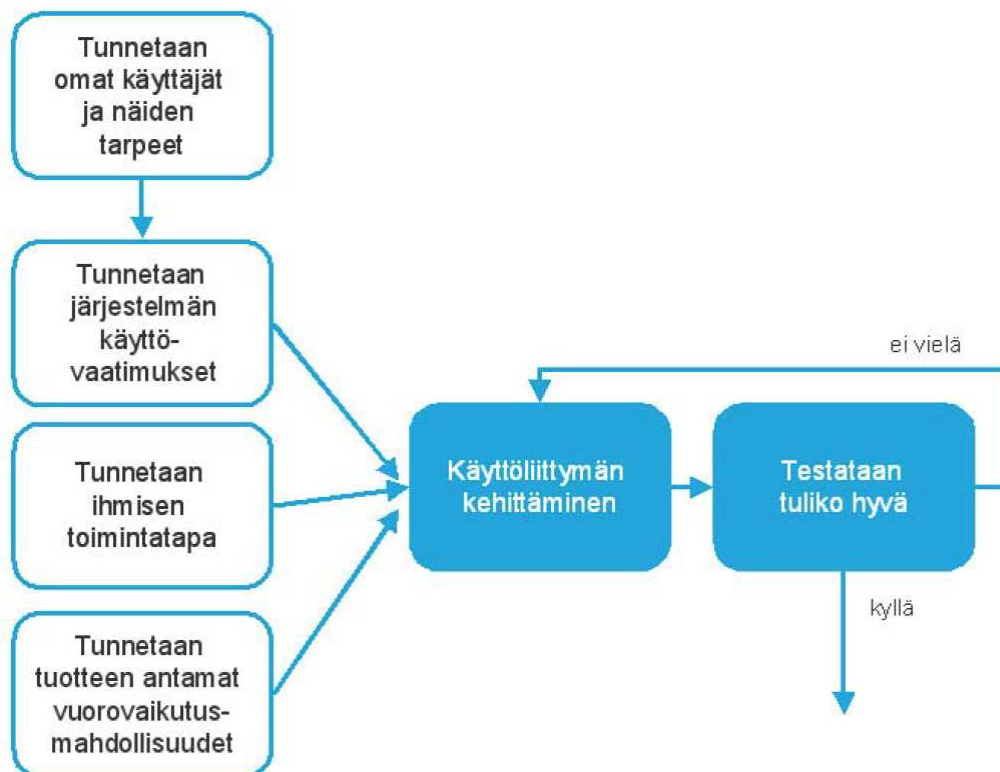
Samalla myös esitettiin toive, että kommunikointi ilmastointikoneen ja valvomo-ohjelmiston välillä toteutettaisiin OPC -palvelimen kautta. Tutustuminen ja käyttöohjeen tekeminen OPC -standardin mukaiselle kommunikoinnille auttaa tulevaisuudessa muissa projekteissa, joissa halutaan käyttää kyseistä kommunikointistandardia.

Ilmastointikoneen käsikäytöllä tarvittavat turvarajat koskevat jäähdytyskompressorimoottori yksikköä, ettei sitä saada ylikuumenemaan ja pahimmassa tapauksessa vahingoittumaan käsikäytöllä.

2 Käyttöliittymän suunnittelu

2.1 Suunnittelun lähtökohdat

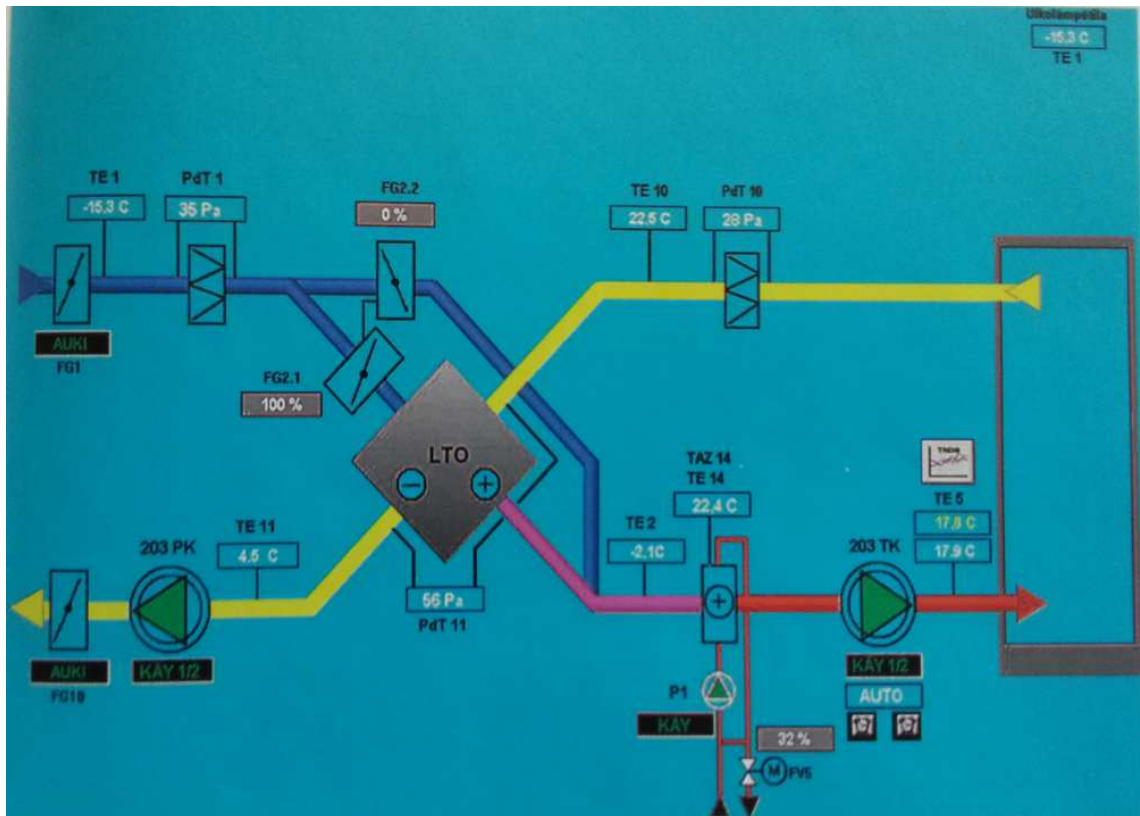
Kun tehdään käyttöliittymää kohteeseen, olisi jo suunnitteluvaiheessa hyvä tiedostaa käyttäjä ja näiden tarpeet. Muita käyttöliittymän tai valvontaohjelman suunnitteluun vaikuttavia asioita ovat muun muassa käyttöliittymälle asetetut vaatimukset, käytettävyystiedot, ohjeet ja standardit. (Sinkkonen, Kuoppala, Parkkinen, Vastamäki 2009, 11.) Kuvassa 1 prosessin eteneminen on kuvattu kaaviona.



Kuva 1. Käyttöliittymän kehityksen prosessi (Sinkkonen ym. 2009, 11).

Käyttöliittymää suunniteltaessa on myös hyvä ottaa huomioon, kuinka ihminen oppii käyttämään uutta laitetta ja kuinka hän muistaa kuinka tuotetta käytetään. Hyvällä käyttöliittymällä voidaan auttaa ihmistä selviytymään alun haasteista pa-

remmin. Tähän voidaan vaikuttaa selkeällä ja intuitiivisella ulkoasulla, komponenttien sijoittelulla, mitä palautetta ohjelma antaa käyttäjälle ja pohtimalla ihmisen ymmärrystä ja ohjelmiston vuorovaikutusta. (Sinkkonen ym. 2009, 10.) Kuvassa 2 on tyypillinen ilmastointikoneen valvontanäkymä.



Kuva 2. Tyypillinen konventioiden mukainen ilmastointijärjestelmän valvontanäkymä (Sähkötieto ry 2008, 93).

Uutta suunniteltaessa tulisi myös ymmärtää, etteivät kaikki asiat välttämättä ole loppukäyttäjälle aivan yhtä itsestään selviä kuin pitkään asian kanssa työskennelleelle asiantuntijalle. Tähän samaan asiaan törmätään koulumaailmassa päivittäin; kun opettaja on saattanut opettaa asiaa jopa kymmeniä vuosia, voi hänellä olla vaikeuksia samaistua oppilaiden asemaan joille asia on entuudestaan täysin tuntematon. Usein joku ymmärtämisen kannalta olennainen seikka itsestään selvyytensä saattaa vain jäädä kertomatta. Ihmiselle on tyypillistä unohtaa kuinka paljon he ovat tehneet jonkin oppimisen eteen työtä, ja silloin voi tuskastua kun asiat eivät ole toisille yhtä itsestään selviä (Sinkkonen ym. 2009, 16).

2.2 Käytettävyys

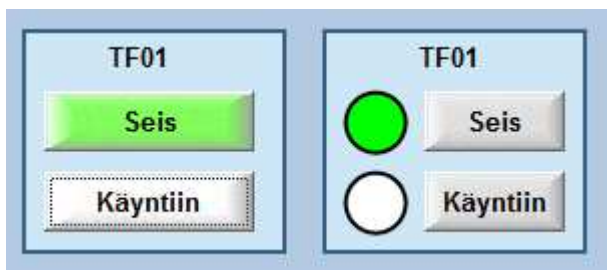
Käytettävyyttä määritellään ISO 9241-11 -standardissa. Standardissa ei ole määritelty käytettävyydelle mitään tarkkoja ominaisuuksia, mutta se määrittelee käytettävyuden seuraavasti: "Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" (The international Organization for standardization 1998). Eli vapaasti suomennettuna: Missä määrin tuotetta tietty käyttäjä voi käyttää saavuttamaan tietyn tavoitteen tehokkaasti ja tyytyväisenä tietyssä yhteydessä. Käyttöliittymän käytettävyuden kannalta olennaisia ovat: selkeys, hahmotettavuus, tehokkuus ja tarkoituksenmukaisuus.

Nykyään pyritään, että käyttäjä osaisi käyttää keskeisiltä osiltaan käyttöliittymää ilman koulutusta tai ohjeiden lukemista. Perehdytyksellä voidaan parantaa käytön tehokkuutta, mutta perustoiminnot tulisi pystyä hallitsemaan omatoimisesti. (Suomen automaatioseura ry 2010, 111.)

Värit ovat tehokas tekijä parantamaan nähdyn hahmottamista, mutta väreillä voidaan myös rikkoa elementtien hyvällä sijoittelulla saatu ryhmittely. Näkymää on helpompi lukea jos värejä on käytetty harkiten ja johdonmukaisesti. Käytettyjen värien tulisi olla hillittyjä eli tulisi välttää liian voimakkaita värejä, sillä on vaarana että niiden käyttö saa näytöstä sekavan. Tässä tapauksessa puhutaankin niin sanotusta Las Vegas -ilmiöstä, jossa kaiken korostaminen johtaa siihen, ettei lopulta mikään korostu. Hyvänä esimerkkinä voimakkaan värin käytöstä voitaisiin mainita esimerkiksi punaisen värin käyttö ilmaisemaan poikkeusta tai pysähtynyttä tilaa. Tässä tilanteessa voimakkaalla värillä saadaan vahvistettua viestin perille menoa. Värien käyttöä voi rajoittaa prosessissa totutut tai tiettyihin tilanteisiin standardoidut värit, joilla voidaan ilmaista jotain määrättyä ainetta tai toimintoa putkistossa. (Suomen automaatioseura ry 2010, 105.) Ilmastoinnissa käytetyt ilmapirtujen värit on esitetty liitteessä 1.

Kuvassa 3 on esimerkit hyvästä ja huonosta ratkaisusta toteuttaa pumpun käynnistämistä ja sammuttamista. Vasemman puoleinen esimerkki on huono, sillä siinä on laitteen tilan ilmaisu yhdistettynä painikkeisiin vihreällä ja harmaalla värillä. Painikkeen tulisi kertoa mitä tapahtuu sitä painettaessa, mutta käyttäjälle

kerrotaan ristiriitaista informaatiota. Vihreä viittaa yleensä käyntiin ja tekstissä sanotaan seis. Painikkeiden ja tilan tulisi olla erillisiä, kuten oikeanpuoleisessa suositellussa esimerkissä on tehty. (Suomen automaatioseura ry 2010, 111.)



Kuva 3. Kuvassa on esimerkit huonosta ratkaisusta vasemmalla ja suositellusta ratkaisusta oikealla.

3 OPC-standardi

OPC on OPC Foundationin ylläpitämä ja kehittämä alustasta riippumaton yhteensopivuusstandardi teollisuuden ja muiden toimialojen automaation tarpeisiin. OPC tulee sanoista OLE (Object Linking and EMBEDding) for Process Control. OPC on avoin standardi automaatiolaitteen ja ulkopuolisen sovelluksen välillä. Standardi julkaistiin vuonna 1996 ja sen tarkoituksena on mahdollistaa yhteys erikoisprotokollien, kuten Modbus ja Profibus, välillä standardoidun rajapinnan avulla. (OPC Foundation 2016a.)

OPC-standardiin perustuva yhteys mahdollistaa kommunikoinnin ohjelmoitavan logiikan ja ulkopuolisen sovelluksen välillä. OPC pohjautuu Microsoftin COM/DCOM-standardiin jota on laajennettu automaatioprosessien kommunikoinnin tarpeisiin, jotta sillä voitaisiin lukea ja kirjoittaa dataa ohjelmoitavalle logiikalle. Tyypillisiä OPC -asiakasohjelmia ovat prosessin visualisointiin ja prosessin arvojen tallentamiseen liittyvät ohjelmat. (3S-Smart Software Solutions GmbH. 2006, 3.)

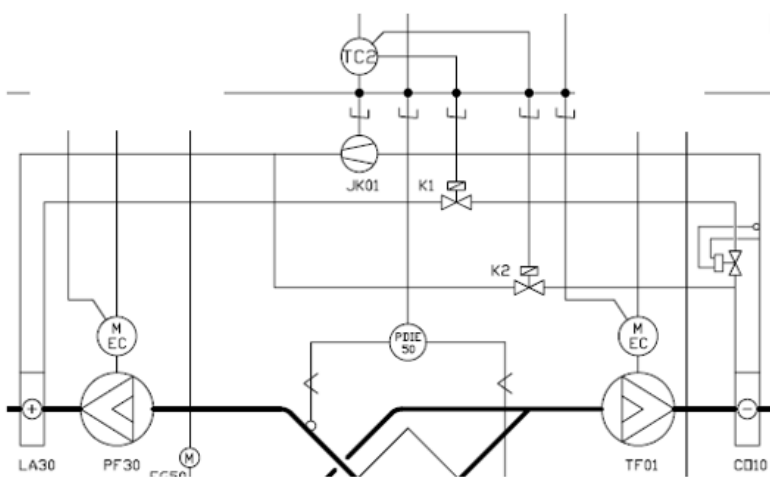
OPC-spesifikaatio on useampia määrittelyjä. Tunnetuimmat näistä OPC määrittelyistä ovat DA (Data Access), A&E (Alarms and Events) ja HDA (Historical Data Access). DA määrittely on tarkoitettu tosiaikaisen prosessidatan siirtoon prosessilaitteissa ja ohjausjärjestelmissä, A&E on puolestaan tarkoitettu hälytys ja tapahtumatietojen välittämiseen ja HDA on tarkoitettu historiatietojen siirtoon. (Suomen automaatioseura ry. 2016.)

Vuonna 2008 OPC Foundation julkaisi uuden määrittelyn OPC:lle. OPC UA (Unified Architecture) on alustasta riippumaton palvelulähtöinen arkkitehtuuri, jossa yhdistetään kaikki aiemmat OPC määritelmät yhdeksi laajennettavaksi kehukseksi. Alustariippumattomuuden ja muiden vanhojen OPC määritelmien yhdistämisen lisäksi uudessa määritelmässä on parannettu tietoturva muun muassa mahdollistamalla tiedonsiirron salaus (encryption) ja oikeaksi tunnistaminen (authentication). UA:n myötä on nyt myös parannettu määritelmän laajennettavuutta mahdollistamalla uusien ominaisuuksien lisääminen, siten etteivät ne vaikuta jo olemassa oleviin sovelluksiin. (OPC Foundation 2016b.)

4 Jäähdytysjärjestelmä ilmastointikoneessa

Yleensä ilmastoinnin jäähdytys toteutetaan kiertoprosessilla. Kiertoprosessissa kylmän tekeminen perustuu suljetussa järjestelmässä kiertävään kylmäaineen höyrystymiseen ja lauhduttamiseen (Silvan, Kauppila & Kaappola 2014, 244). Jäähdytysjärjestelmä ilmastointikoneessa muodostuu kompressorista, lauhduttimesta ja höyrystimestä.

Kompressorissa kylmäaineen paine kohotetaan ennen lauhdutinta, lauhduttimen jälkeen paine alennetaan paisuntaventtiilillä ennen kuin kylmäaine ohjataan höyrystimelle. Höyrystimessä kylmäainevirtaan sitoutunut lämpö on suurempi kuin kompressorin vaatima työ, täten höyrystimestä poistuva lämpövirta on höyrystimessä sitoutuneen lämmön ja kompressorityön summa. (Seppänen 1996, 308.) Kuvassa 4 on osakuva KAIR ilmastointikoneen prosessikaaviosta rajattuna jäähdytysjärjestelmään.



Kuva 4. Jäähdytysjärjestelmän prosessikaavio. LA30 on lauhdutin, CO10 on höyrystin ja JK01 on kompressor.

5 Tavoitteet

Opinnäytetyön tavoitteena on kehittää opiskelijoiden käyttöön ilmastointikoneen käyttöliittymä ja valvomo-ohjelma. Halutaan myös varmistaa ettei ilmastointikoneita voida vahingoittaa käsikäytöllä, joten sen ohjausohjelmaan joudutaan tekemään pieniä muutoksia. Valvomo-ohjelman avulla opiskelijat voivat tutustua ilmastointikoneen toimintaan ja nähdä eri parametrien vaikutus ilmastointikoneen toimivuuteen. Tavoitteena oli myös tutkia OPC-palvelimen käyttöä kommunikoinnissa järjestelmän ja valvomo-ohjelmiston välillä ja lopuksi tehdä käyttöohjeet OPC-palvelimen käyttöönotosta ja muista vaadituista toimenpiteistä.

Tavoitteisiin tulisi päästä mahdollisimman edullisesti, joten valittujen ratkaisujen ja ohjelmistojen tulisi olla sellaisia jotka löytyvät jo valmiiksi koululta tai sellaisia joihin on saatavilla ilmainen lisenssi opetus- ja opiskelukäyttöön. Alustavasti sovittiin että opinnäytetyön tulisi olla valmis huhtikuussa 2016.

6 Käytetyt ohjelmat

6.1 Ohjelmien valinta

Käytettävät ohjelmat valikoituivat sen mukaan mitä koululla on käytössä tai joihin on mahdollista saada opetus- tai opiskelijakäyttöön ilmainen lisenssi. Koululla olevassa KAIR harjoitteluilmastointikoneessa ilmastointikoneen ohjaus on hoidettu WAGO 750-881 ohjelmoitavalla logiikalla, joten luonnollisesti tätä kautta valikoitui käyttöön CoDeSys v2,3 -ohjelmointiympäristö ja CoDeSys OPC Server v2,3 -palvelin. Ohjelmat löytyivät jo koululta valmiina.

Valvomo-ohjelmistoksi valikoitui Wonderware Indusoft Web Studio v8,0, johon oli mahdollista saada ilmainen, vuoden kerrallaan voimassa oleva educational lisenssi. Web Studiossa lisenssit on jaoteltu tagien lukumäärän mukaan ja lisens-

siksi valittiin 150 tagin lisenssi. OPC-yhteyden muodostamiseen tutustuessa käytin myös internetistä ilmaiseksi löytynyttä Matrikon Data Manageria varmistamaan että yhteys OPC palvelimelle toimii ja muuttujat listautuvat oikein.

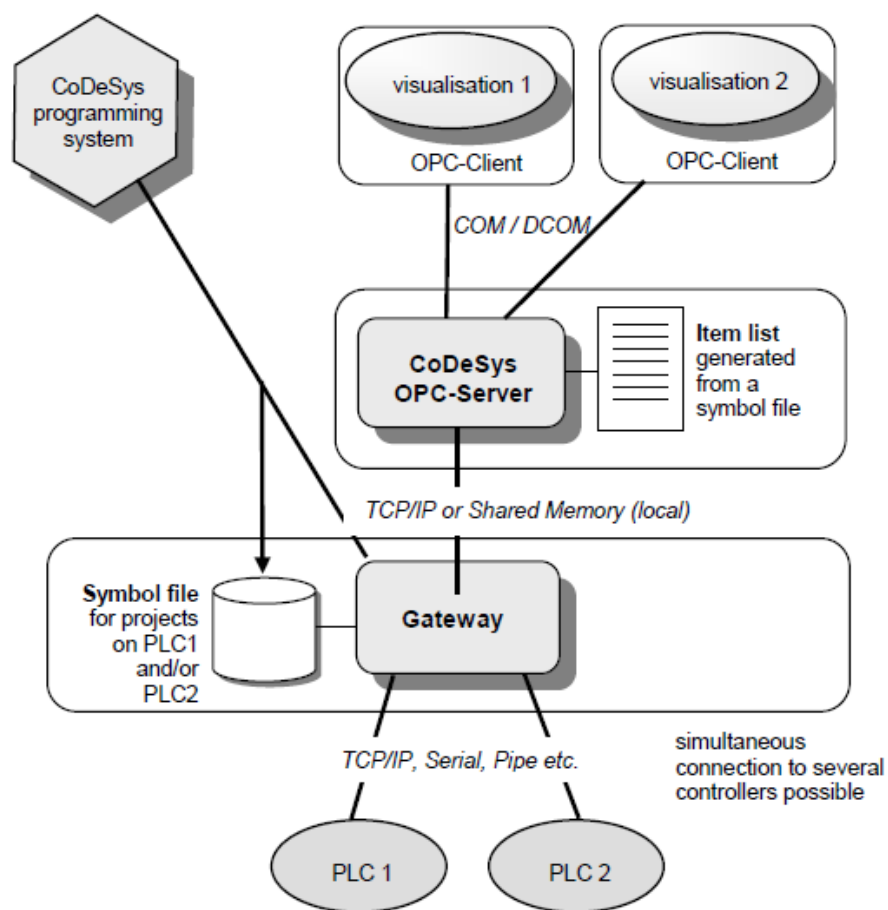
6.2 CodeSys OPC Server

CoDeSys OPC Server on 3S-Smart Software Solutions:in kehittämä ja ylläpitämä OPC-standardin mukainen palvelin, joka mahdollistaa IEC 61131-3 mukaisen prosessidatan lukemisen ja kirjoittamisen ohjelmoitavalta logiikan ja prosessin välillä. Ohjelman uusin versio kirjoitushetkellä on versio 3. (3S-Smart Software Solutions GmbH 2016.)

Kuvassa 5 on havainnollistettu CodeSys OPC-palvelimen kommunikointia ohjelmoitavan logiikan ja asiakasohjelman välillä sekä sen arkkitehtuuri. Opinnäyttyöni tapauksessa ohjelmoitava logiikka on TCP/IP protokollalla yhdistettynä CoDeSys Gatewayn kautta CoDeSys OPC palvelimelle, joka vuorostaan kommunikoi asiakasohjelman kanssa.

OPC-palvelin ei ole aktiivinen koko aikaa, vaan käynnistyy yhteyden muodostamisen yhteydessä ohjelmoitavan logiikan ja asiakasohjelman välillä. OPC-palvelin ilmoittaa asiakasohjelmalle aina kun muuttujan tila tai arvo muuttuu. (3S-Smart Software Solutions GmbH 2006, 3.)

Ohjelmaa ohjelmoitavalle logiikalle siirrettäessä tulee siirtää myös projektin symbolitiedosto, josta luodaan OPC-palvelimelle lista käytössä olevista muuttujista. Yhteyden muodostamista ja symbolitiedoston määrittelyä on käsitelty enemmän liitteessä 3.



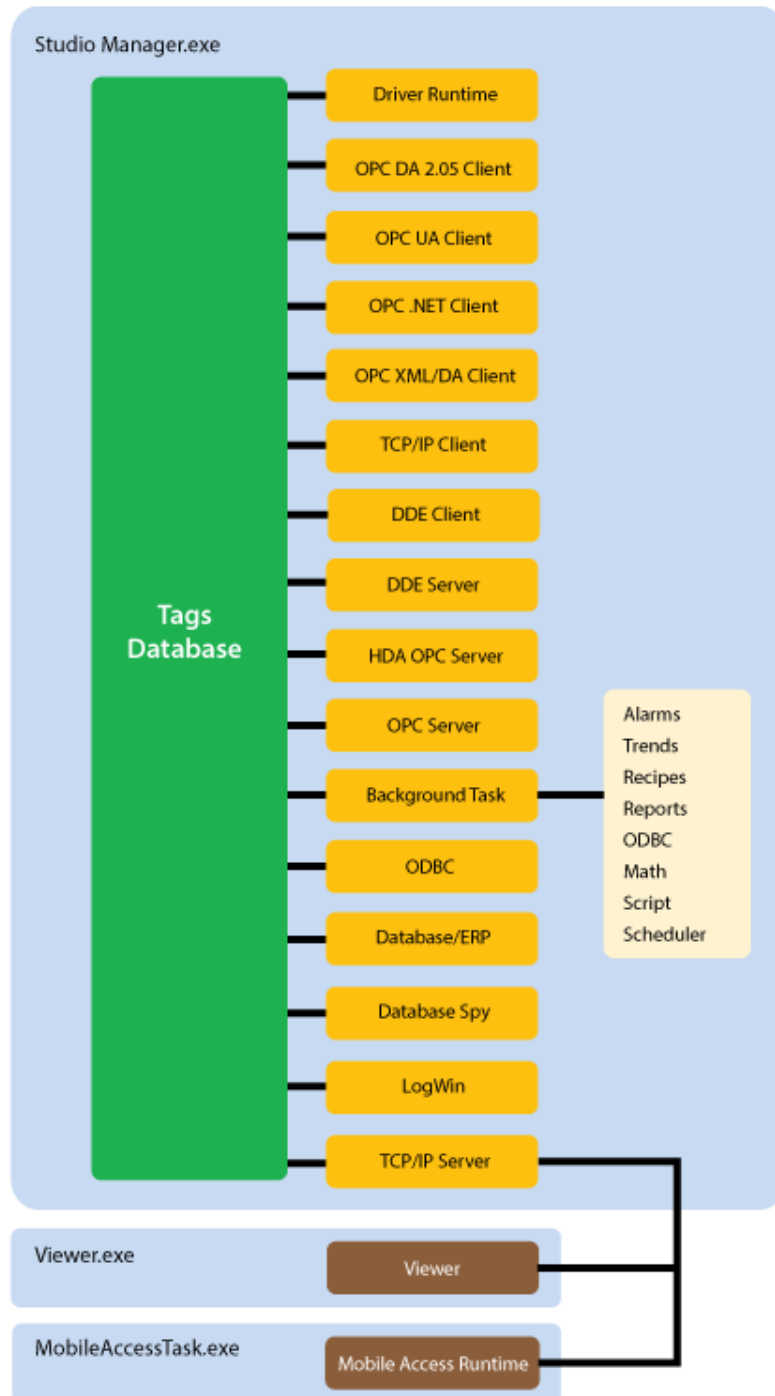
Kuva 5. CodeSys OPC - serveri V2.0 arkkitehtuuri (3S-Smart Software Solutions GmbH. 2006, 3).

CoDeSys OPC Server 2,0 ei tue OPC UA-määritelmää, joten sitä kautta ei päästy nauttimaan uuden määritelmän tuomista eduista. Uudemmassa CoDeSys OPC Server v.3,0 OPC UA on jo tuettuna, mutta kyseistä ohjelmaa ei ollut koululla käytössä.

6.3 Wonderware Indusoft Web Studio 8.0

Indusoft Web Studio, lyhennetään jatkossa IWS, on kehitetty erityisesti SCADA HMI -sovelluksien, näyttötaulujen sekä OEE käyttöliittymien kehittämissympäristöksi ja suorituksen aikaiseksi ohjelmaksi (Schneider Electric Software 2016a). Ohjelmassa on panostettu helppokäyttöisyyteen ja integroinnin helppouteen eri järjestelmiin. Ohjelmasta löytyy kommunikointiajurit yleisesti käytössä olevalle laitteelle ja laitevalmistajalle valmiiksi. Ohjelmassa myös on tuki yleisimmille eri OPC määrittelyille: OPC DA 2.05, OPC UA, OPC .Net ja OPC XML/DA. Helppokäyttöisyyttä ja ohjelmaan tutustumista on edelleen helpotettu kattavilla manuaaleilla ja ohjeilla jotka löytyvät vaivattomasti valmistajan sivuilta sekä video-, että PDF -muodoissa. Valmistajan sivuilta on myös useita esimerkkiohjelmia eri tarpeisiin.

Tagit ovat IWS -projektin ydin. Kuvassa 6 on esitetty Projektin suorituksen aikainen arkkitehtuuri, ja siitä näkee kuinka kaikki runtime-toiminta liittyy osana Tags Databaseen. IWS:ssa tagilla tarkoitetaan yksinkertaistetusti muuttujien nimiä, vaikka todellisuudessa se on hieman laajempi käsite, joita käytetään lukemaan ja varastoimaan muilta laitteilta saatua dataa. IWS:ssa tag:it on jaoteltu kolmeen ryhmään: Projekti muuttujat (Project Tags), jaetun tietokannan muuttujat (Shared Database Tags) ja systeemi muuttujat (System Tags). (Schneider Electric Software 2016b.)



Kuva 6. Projektin runtimearkkitehtuuri Windows ympäristössä (Schneider Electric Software 2016b).

Project tagit ovat muuttujia, jotka käsittävät ruudulla olevat muuttujat, muuttujat joita luetaan ja kirjoitetaan yhdistetyiltä laitteilta, control tagit ovat muuttujia joita käytetään laskutoimituksissa. Shared Database Tagit ovat muuttujia, jotka on luotu PC-pohjaisella hallintaohjelmalla ja sen jälkeen tuotu Web Studioon. Näitä shared database tageja ei voida muuttaa Web Studiossa vaan muutokset tulee

suorittaa PC-hallintaohjelmassa, jonka jälkeen ne voidaan uudelleen tuoda Web Studion muuttujatietokantaan. Kolmas muuttuja ryhmä Web Studiossa system tagit, joilla tarkoitetaan muuttujia jotka ovat ennalta määriteltäviä muuttujia tai funktioita, kuten aikaan ja päivämäärään liittyvät funktiot. Suurin osa system tageista ovat vain luettavia eli näitä muuttujia ei voi poistaa tai muokata tietokannassa. (Schneider Electric Software 2016c.)

7 Toteutus

Keväällä 2015 koululle oli rakennettu uusiin rakennusautomaatiolaboratorion tiloihin harjoittelu- ja opetuskäyttöön soveltuva ilmastointikone. Ilmastointikoneelle tarvittiin selkeä ja helppokäyttöinen valvonta- ja käyttöympäristö opiskelijoille. Yhteys ohjelmoitavan logiikan ja valvomo-ohjelmiston välillä haluttiin toteutettavan OPC-palvelimen kautta, sillä koululla on muutama projekti joissa tekemääni selvitystyötä voidaan käyttää hyödyksi.

Haluttiin myös varmistua, että ilmastointikoneen jäähdytyksen väärällä käytöllä sitä ei saatettaisi rikkoa. Tämä vaatii, että tutustuisin valmistajan ohjeisiin ja suosituksiin muun muassa kompressorin käyttökiertoajoista. Käytössäni oli KAIR ilmastointikoneen konekortti ja dokumentit, joita oli käytetty ilmastointikonetta rakennettaessa. Työn toteutukselle ja tavoille minulle annettiin vapaat kädet.

7.1 OPC-yhteys

OPC -palvelimeen ja sen kautta yhteydenmuodostamiseen tutustumisen aloitin tutustumalla käytössä olevien ohjelmien manuaaleihin ja lukemalla OPC:sta yleisesti OPC Foundationin ja Suomen automaatioseuran verkkosivuilta. Heti alussa suurimman ongelman aiheutti CoDeSys OPC Server manuaali, sillä sitä on viimeksi päivitetty vuonna 2006. Ohjeet on tehty Windows 98 -käyttöjärjestelmälle,

joten useat vaiheet asennuksen ja rekisterihallinnan osilta olivat nykypäivän näkökulmasta virheellisiä.

Alkuun pääsemiseksi sain apua vanhasta projektiraportista, jossa oli käsitelty tarvittavia muutoksia CoDeSys PLC -ohjelmaan, jotta datan lähettäminen ja vastaanottaminen olisi mahdollista ohjelmoitavalla logiikalla. Raportissa yhteys oli muodostettu WAGO ohjelmoitavan logiikan ja Python-ohjelman välille. Tärkeimpänä muutoksena ohjelmaan oli symbolitiedoston luominen, joka pitää sisällään käytettävissä olevat muuttujat. Tiedosto ladataan CoDeSys Gatewaylle ja tiedostosta luotu lista muuttujista ohjelmoitavalle logiikalle ohjelmaa siirrettäessä.

Kun periaate kommunikointiin oli selvinnyt, aloin testaamaan sitä yksinkertaisella ohjelmalla jossa oli vain yksi sisääntulo ja yksi ulostulo. Ohjelmaan tehtyjen määrittelyjen ja ohjelman ohjelmoitavalle logiikalle siirtämisen jälkeen ei asiakasohjelmalla (client) käytetty Indusoftin Web Studio kuitenkaan löytänyt hakukonetta, jolla OPC-palvelin olisi Indusoftin ohjeiden mukaan pitänyt löytyä, joten päätin etsiä internetistä jonkin ilmaisen asiakasohjelman yhteyden testaukseen. Pienen tutkiskelun jälkeen päädyin Matrikonin OPC Data Manageriin. OPC Data Manager löysi heti automaattisesti määrittelemäni localhost OPC-palvelimen ja ohjelmassa luodut muuttujat. Täten selvisi, että määrittelyt oli tehty oikein ja ongelma on Indusoft Web Studiossa.

Seuraavaksi aloin selvittämään, miksi IWS ei löytänyt OPC-palvelinta. Lukemalla keskustelufoorumeita löysin muutamia tapauksia, joissa törmättiin samaan ongelmaan, mutta ei mitään syytä tai ratkaisua siihen. IWS:n ilmoittama virhekoodikin veti vesiperän. Lopulta löysin IWS:sta Import Wizardin, jolla pystyy tuomaan ohjelmaan OPC Server Databasen. Tätä kautta muuttujien tuominen valvomo-ohjelmaan onnistui, minkä jälkeen aiemmin epäonnistunut tapa myös alkoi toimimaan.

Kun pienimuotoisessa ympäristössä tehdyt testit yhteyden muodostamiseksi olivat onnistuneet, tuli vuoroon tehdä määrittelyt KAIR ilmastointikoneen ohjelmaan. Symbolitiedoston lähettäminen WAGO 750-881 ohjelmoitavalle logiikalle ei sujunut ongelmitta, sillä tiedostoja siirrettäessä CoDeSys väitti, että siirrettävien tiedostojen yhteiskoko olisi ollut liian suuri ohjelmoitavan logiikan muistiin mahtuakseen. Kontrollerin manuaalin mukaan tämä ei pitänyt paikkaansa sillä

muistia laitteesta löytyy datalle 512 kilotavua ja ohjelmalle 1 024 kilotavua. Siirrettävällä symbolitiedostolla oli kokoa 127 010 tavua ja ohjelmalla oli kokoa 311 284 tavua, joten tilan olisi pitänyt riittää hyvin käytettävässä ohjelmoitavassa loogikassa. Ongelmien vuoksi päätin vaihtaa hyllystä löytyvän vastaavan WAGO 750-881 ohjelmoitavan logiikan tilalle ja homma onnistui sen jälkeen ilman ongelmia. Yhteyden muodostaminen on käsitelty vaihe vaiheelta liitteessä 3 CoDeSys OPC Server.

7.2 Käyttöliittymä

Projektin alussa pyrin keräämään kirjallisuutta, jossa käsitellään ilmastointikoneita ja niiden käyttöliittymää. Opettajalla oli muutama ehdotus käyttöliittymä-ohjelmistolle, joista toisen hylkäsin ensimetreillä sillä valitsemassani Indusoft Web Studiossa käytettävyyys ja dokumentointi oli tehty esimerkillisen hyvin. Indusoftilla on muun muassa Youtubessa ja heidän omilla kotisivuillaan tarjolla kaikki tarvittava materiaali ja opetusvideoita eri ominaisuuksille esimerkkeineen.

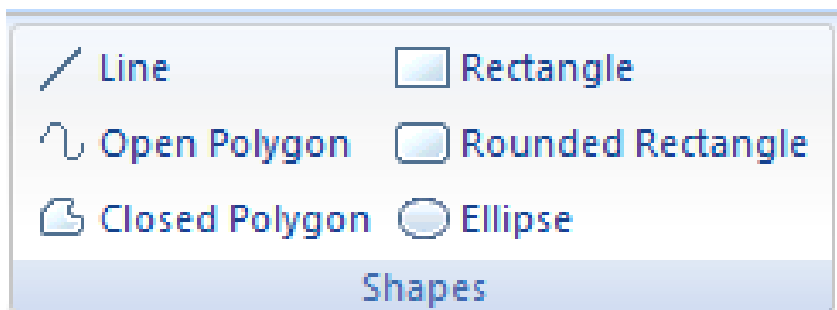
Aloin hoitaa opiskelijalisenssin hankkimista itsenäisesti, mutta lopulta se kilpistyi lisenssiehdon vaatimukseen, että sopimuksen tulee olla koulun ja Indusoftin välinen käyttöpaikan perusteella. Lisenssi on konesidonnainen eikä sitä voida siirtää toiselle koneelle ja täten jokaiselle tietokoneelle tulee hankkia lisenssi erikseen. Tämän tajuttuani siirsin lisenssiasioden hoidon koulun tietohallinnolle.

Vaikka käyttöliittymän suunnittelu ei ollut opinnäytetyöni pääasia, vei sen suunnittelu ja toteutus valtaosan ajasta. Pyrin tekemään käyttöliittymästä mahdollisimman selkeän ja helppokäyttöisen. Tähän päämäärään auttoi kokemus vastaavantalaisesta projektista, jossa tein ammattikoululle vesiprosessin käyttöliittymän kosketusnäytölle, ja tutustuminen alalla vallitseviin muihin ilmastointikoneiden käyttöliittymiin ja konventioihin. Alan kirjallisuudesta esimerkiksi Esa Sandbergin toimittamasta Sisäilmasto ja ilmastointijärjestelmät, löysin oikeat piirrosmerkit ja putkien ja kanavien värit. Standardien mukaiset ilmastointikoneiden valvontaohjelmistojen piirrosmerkit ja kanavien värit on listattu liitteissä 1 ja 2.

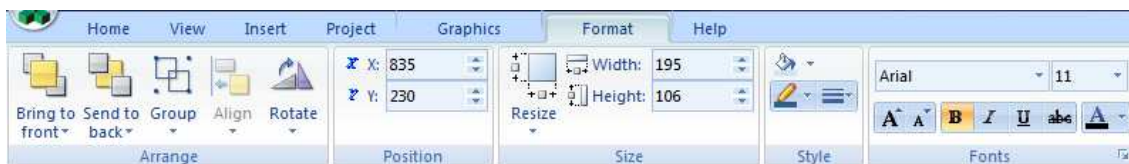
7.2.1 Symboleiden luominen Indusoft Web Studiossa

Ohjelmassa tulee valmiita symboleita eri käyttötarkoituksiin. Valmiina löytyy muun muassa yleiskäyttöisiä prosessisymboleita, liukukytкимиä ja merkkivaloja. Totesin hyvin nopeasti etteivät valmiit symbolit soveltuneet tähän projektiin. Symboleiden luominen annetuilla työkaluilla on helppoa ja yksinkertaista. Suurimpana puutteena ohjelmassa ei voida kiertää symbolia kuin 90°:teen kulmissa. Tämän takia jouduin sulkua- tai säätöpeltien symbolin piirtämään kahteen kertaan eri kulmissa.

Ensin symboli piirretään käytössä olevilla perusmuodoilla. Käytettävissä olevat perusmuodot joita käytetään piirtämiseen näkyvät kuvassa 7. Symboleihin voidaan myös lisätä värejä ja kirjoitusta. Symbolin eri elementtien attribuutteja ja paikkaa voidaan muokata kuvan 8 työkaluilla. Symbolien attribuuttien muokkausnäkyminen on kuvassa 8. Jos symboli muodostuu useammasta elementistä, niin voidaan elementit yhdistää yhdeksi kokonaisuudeksi valitsemalla kaikki halutut elementit ja oikean hiiren napin valikosta valitsemalla group.

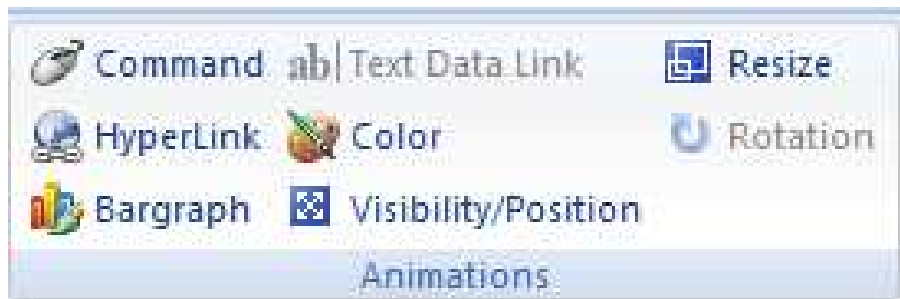


Kuva 7. Indusoft Web Studiossa käytettävissä olevat piirustustyökalut.



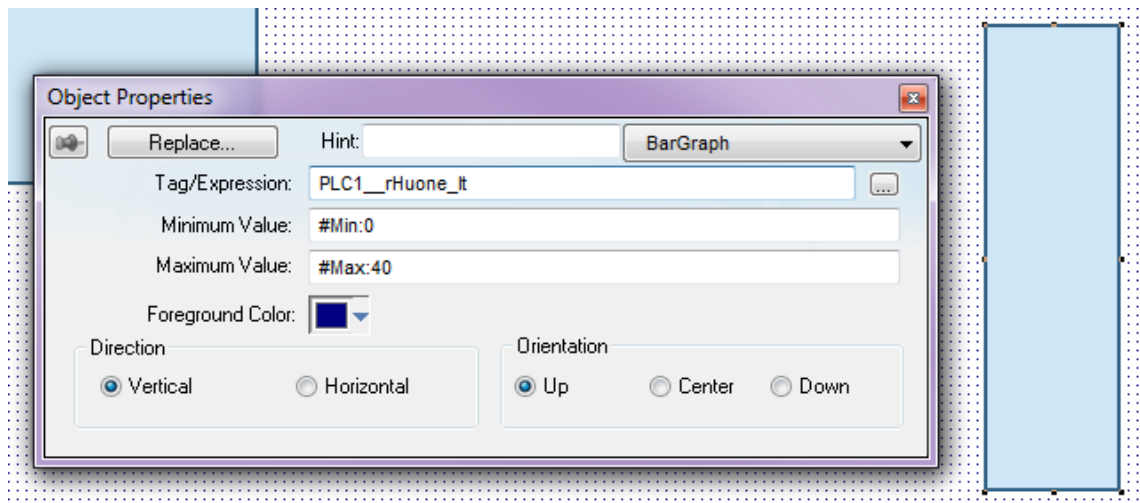
Kuva 8. Symbolin elementtejä ja paikkaa pystytään muokkaamaan aktivoimalla elementti ja format -välilehdestä muuttamalla attribuutteja.

Kun symboli on saatu haluttuun ulkomuotoon pitää sille valita tarvittavat ominaisuudet tai toiminnollisuus. Valittavissa (kuva 9) on muun muassa Command, jonka alta symboli voidaan määrittää painonapiksi, tai halutessa ajamaan jokin haluttu VBScripti. Bargraph määrittää kuvion palkiksi joka vaihtaa kokoa muuttujan arvon mukaan halutulla välillä. Color, visibility/position ja resize animaatiolla saadaan symboli vaihtamaan väriä, muuttumaan näkymättömäksi/näkyväksi, vaihtamaan paikkaa tai kokoaan määritellyn muuttujan mukaan.



Kuva 9. Symbolille valinnaisia toimintoja ja ominaisuuksia.

Kun symbolille halutut animaatiot on valittu, pääsee niiden ominaisuuksia muuttamaan kaksoisnapauttamalla tehtyä symbolia. Kuvassa 10 on esimerkkinä tehty huonelämpötilan mukaan muuttuva palkki ja sen asetukset.

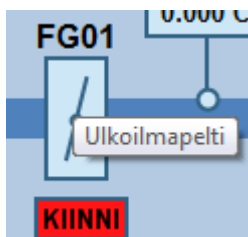


Kuva 10. Esimerkki on tehdystä symbolista ohjelmassa ja sen asetuksista.

7.2.2 Päänäkymä

Käyttöliittymää suunniteltaessa pyrin luomaan siitä mahdollisimman selkeän, sillä tulevat käyttäjät ovat opiskelijoita joille ilmastointilaitteen toiminta ei ole täysin tuttua, mutta tilannetta vaikeuttaa toimintojen määrä. Prosessissa pystytään ohjaamaan useita komponentteja, joten näille komponenteille piti lisätä oma ohjauksensa. Komponenttien määrä ruudulla heikentää käyttöliittymän toimintaa pelkästään valvontatyökaluna, mutta katsoin ratkaisun kuitenkin paremmaksi kuin toisiin ikkunoihin piilotetut toiminnot opetuksen ja ilmastointikoneen tutustumisen näkökulmasta, sillä nyt näkee suoraan mitä komponenttia pystytään ohjaamaan halutessa.

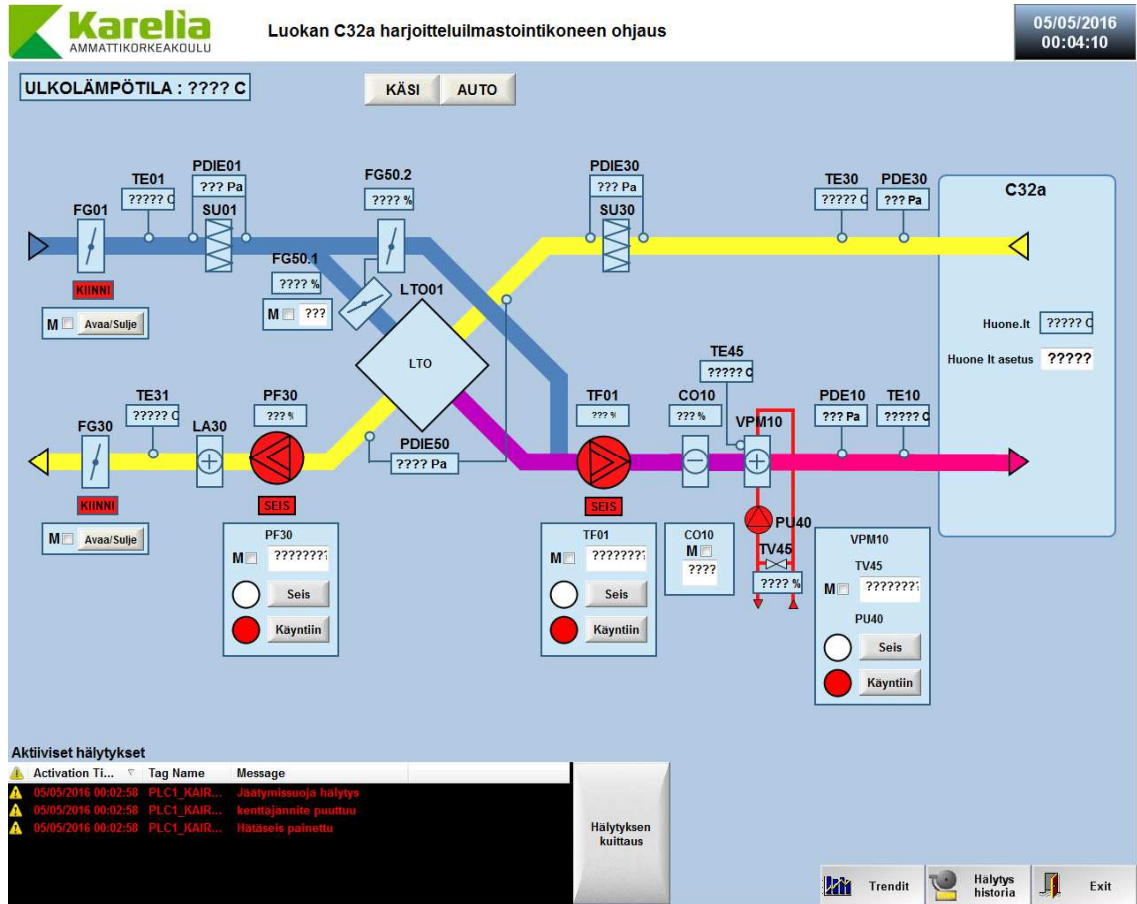
Prosessin ymmärtämistä helpottamaan loin jokaiselle ruudulla näkyvälle komponentille pienen kuvauksen, joka tulee esiin kun hiiren osoitin viedään komponentin päälle (kuva 11). Kehystin myös aina yhteen komponenttiin liittyvät ohjaukset yhdeksi kokonaisuudeksi. Kokonaisuuksien kehystämällä saavutetaan parempi ryhmittely ja myös selkeämpi kuva siitä mitkä ohjaukset liittyvät toisiinsa. Esimerkiksi tulopuhaltimen TF01 käsikäytön valinta, nopeusohjeen määrittely ja päälle/pois -painonapit ovat kehystetty yhdeksi kokonaisuudeksi.



Kuva 11. Esimerkki: FG01 komponentin kuvaus.

Halusin parantaa käytettävyyttä erottamalla eri pohjaväreillä ohjaukseen käytettävät muokattavat- ja pelkästään prosessidatan lukemisen käytettävät kentät. Muokattavat kentät ovat valkealla- ja vain luettavissa olevat kentät vaaleansinisellä pohjalla, sillä näin on helpompi yhdellä silmäyksellä erottaa ne toisistaan.

Pohjaväriksi näytölle valitsin hillityn vaalean harmaansinisen, josta eriväriset komponentit ja symbolit tulevat hyvin esiin rasittamatta silmiä pitkässäkään katselussa. Kuvassa 12 on kuvakaappaus valvontaohjelmiston päänäkymästä.

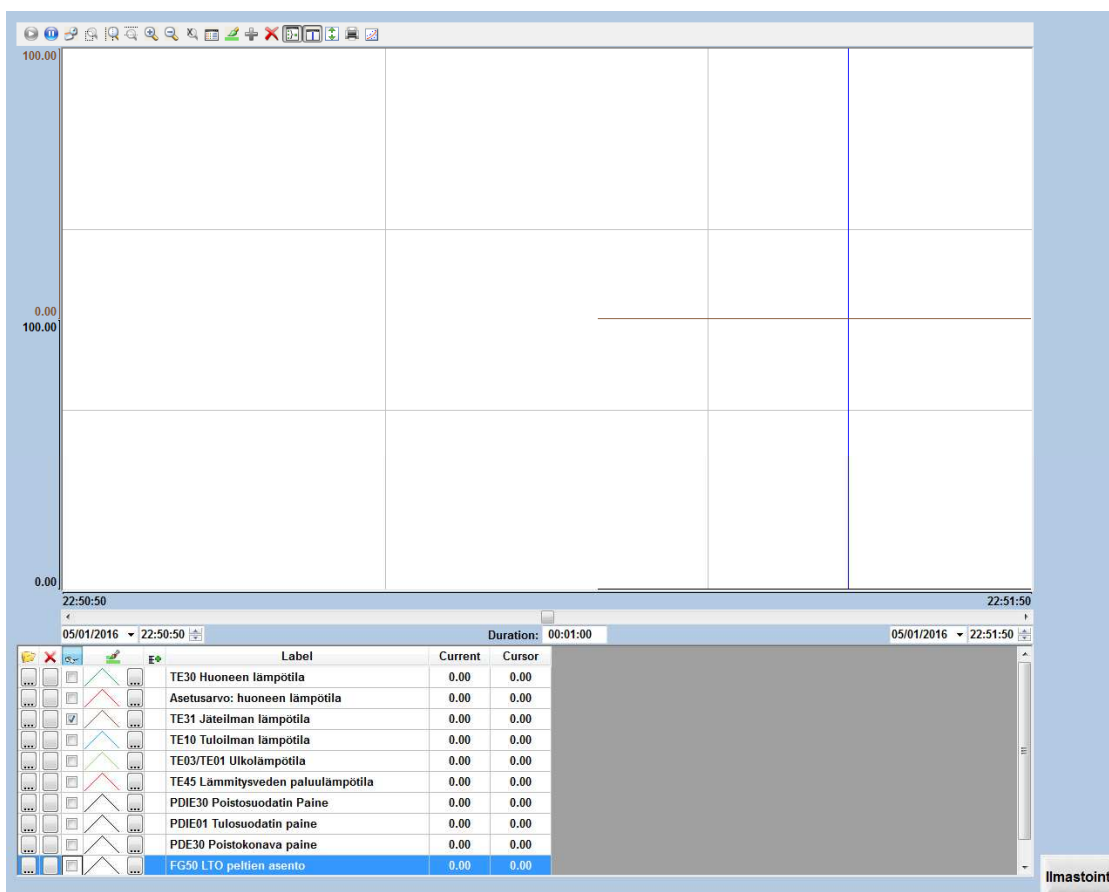


Kuva 12. Valvomo-ohjelman päänäkymä.

Kuvasta 12 huomaa, että olen poikennut lämmöntalteenoton jälkeisessä ilmastointikanavassa standardin mukaisesta väryksestä. Päädyin tähän ratkaisuun siksi, että halusin korostaa violetilla käytössä olevan lämmöntalteenoton aikaansaamaa ilman lämpenemistä. Samaa tehokeinoa on käytetty muutamassa kuvassa, joita käytin oman työni pohjalla perusideana. Violetti väri sopii tähän paikkaan hyvin, sillä se on kylmän sinisen ja lämpimän punaisen sekoitus, eikä sitä ole varattu mihinkään muuhunkaan käyttötarkoitukseen.

7.2.3 Trendikäyrät

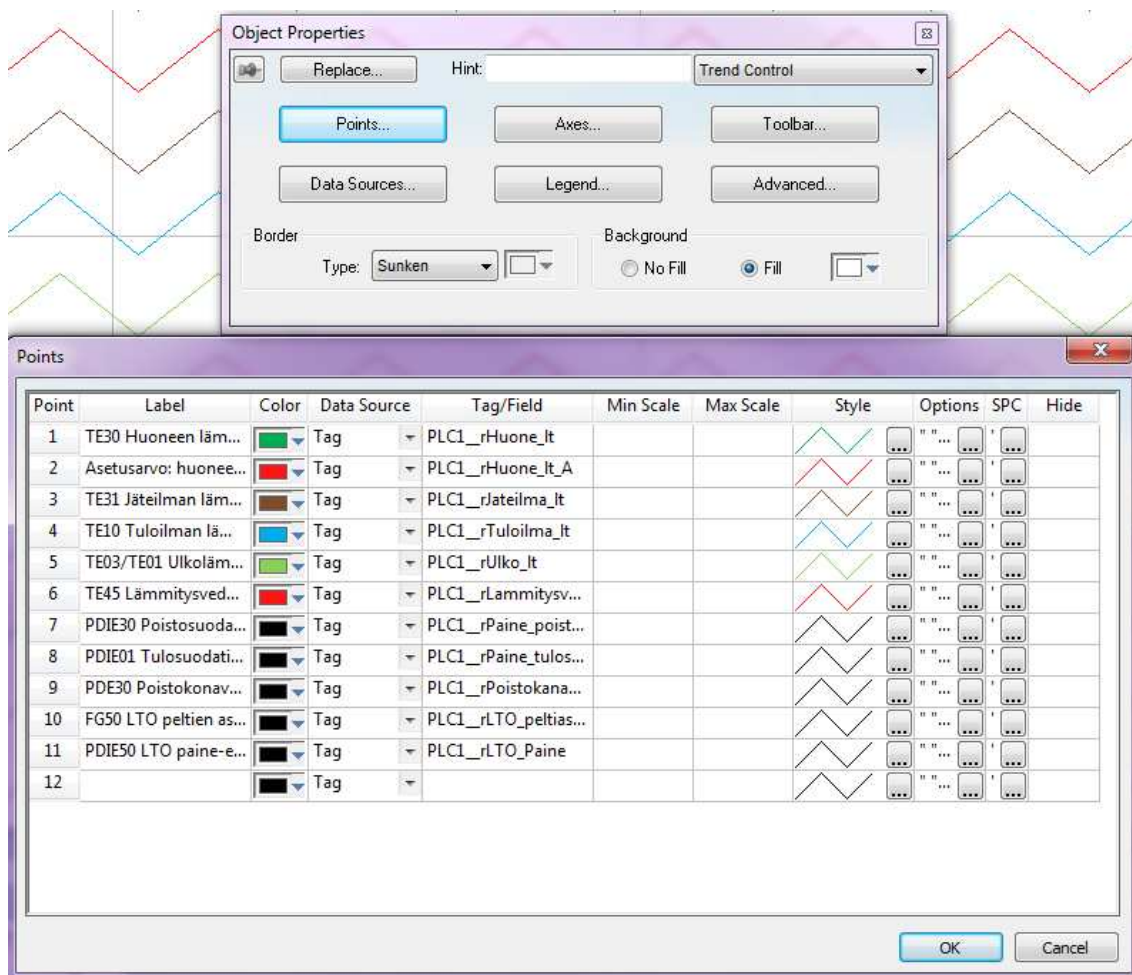
Web Studiosta löytyi oletuksena käyttökelpoinen työkalu trendien seurantaan ja tallentamiseen. Yksinkertaisimmillaan trendikäyrän saa luotua valitsemalla Graphic välilehdestä Data Objects alta Trend ja viemällä se ruudulle. Samasta paikasta löytyy myös näkymä vikaistorialle, jota tarvitaan hieman myöhemmin. Trendinäkylässä pystytään listalta valitsemaan halutut kohteet ja tarpeen mukaan vaihtamaan kuvaajan väriä. Tarkan arvon tietyllä hetkellä saa näkyviin viemällä kursori haluttuun kohtaan trendinäytöllä. Kuvassa 13 on ruutukaappaus valmiista trendinäkyvästä.



Kuva 13. Yleiskuva trendit-ikkunasta.

Trendit näkymän toimintakuntoonlaitto vaati ainoastaan muuttujien määrittelyn ja niiden nimeämisen ymmärrettävämpään muotoon. Kuvassa 14 on tehty määrittelyt kaikessa yksinkertaisuudessaan. Halutuille trendeille on määritetty nimi (label), kuvaajan väri, data lähde, seurattava muuttuja ja trendikäyrän tyyli. Trendille

voitaisiin myös määrittellä ylä- ja ala raja-arvot, mutta tässä työssä niitä en katsonut tarpeelliseksi niitä määrittellä, sillä automaattinen skaalaus toimi riittävän hyvin.



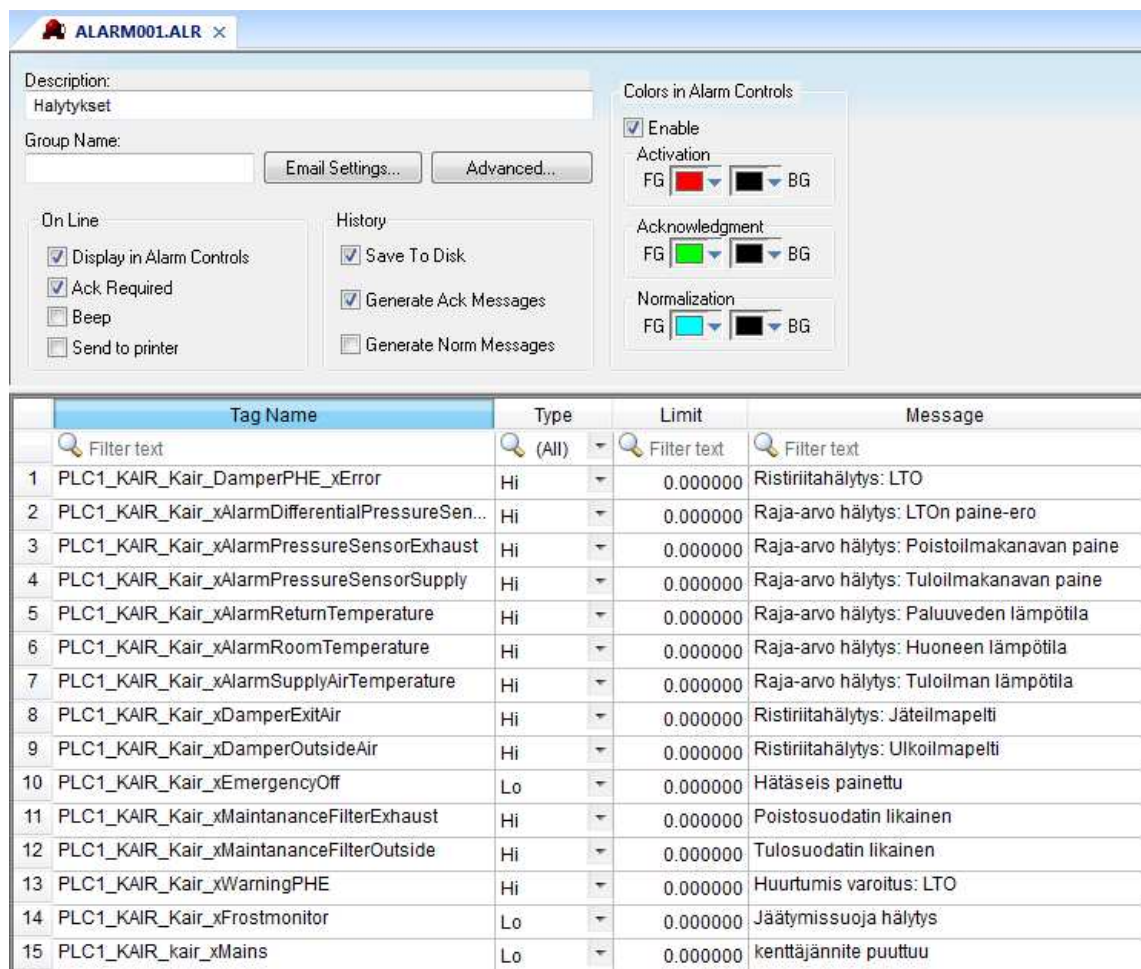
Kuva 14. Trendit näkymän vaatimat määrittelyt.

7.2.4 Hälytykset

Indusoft Web Studiossa taskeista pystytään määrittelemään tarvittavat hälytykset (Alarms) ja muuttujat joita seurataan. Hälytyksen aktivointi voidaan triggeroida muuttujasignaalin muuttuessa hälytystyyppien mukaan: HiHi, Hi, Lo, LoLo, muuttumisnopeuden ja poikkeaman mukaan sekä positiiviseen että negatiiviseen suuntaan. Hälytyksissä voidaan myös määrittää teksti joka näytetään hälytystapahtuman sattuessa ja hälytyksen prioriteetti, sekä halutaanko hälytyksille saada

kuittaus ja/tai äänimerkki. Jos hälytykset tahdotaan tallentaa, voidaan hälytyshistoria tallentaa tiedostoon myöhempää tarkastelua varten automaattisesti valitsemalla History - Save To Disk aktiiviseksi. Hälytykset voidaan myös värikoodata hälytysnäyttöä varten. Hälytyksille voidaan määrittellä värit kolmeen eri tapaukseen: aktiiviset hälytykset, kuitatut hälytykset ja normalisoituneet hälytykset.

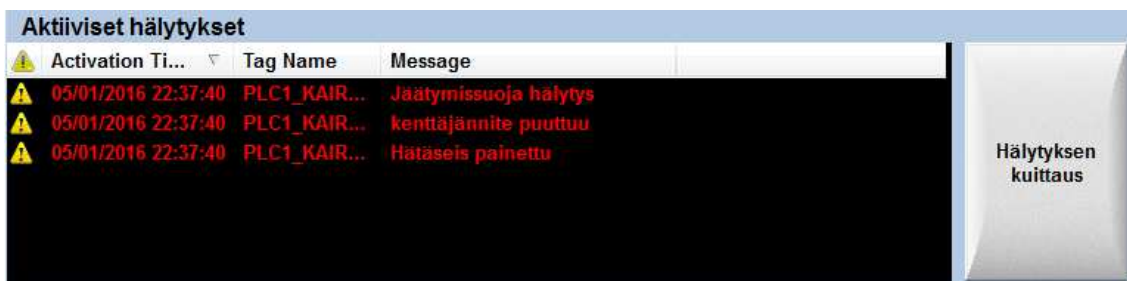
Tässä projektissa hälytykset ja niiden tyypit prioriteetteineen tuli suoraan Co-DeSys PLC -ohjelmasta. Ohjelmassa on määritelty yhteensä viisitoista tapahtumaa jotka voivat aiheuttaa hälytykset. Kuvassa 15 alla näkyvät käyttöliittymään luodut hälytykset ja niiden määrittelyt. Samat viisitoista hälytystä ja ehtoa on määritelty myös ilmastointikoneen ohjausohjelmassa ohjelmoitavalle logiikalle.



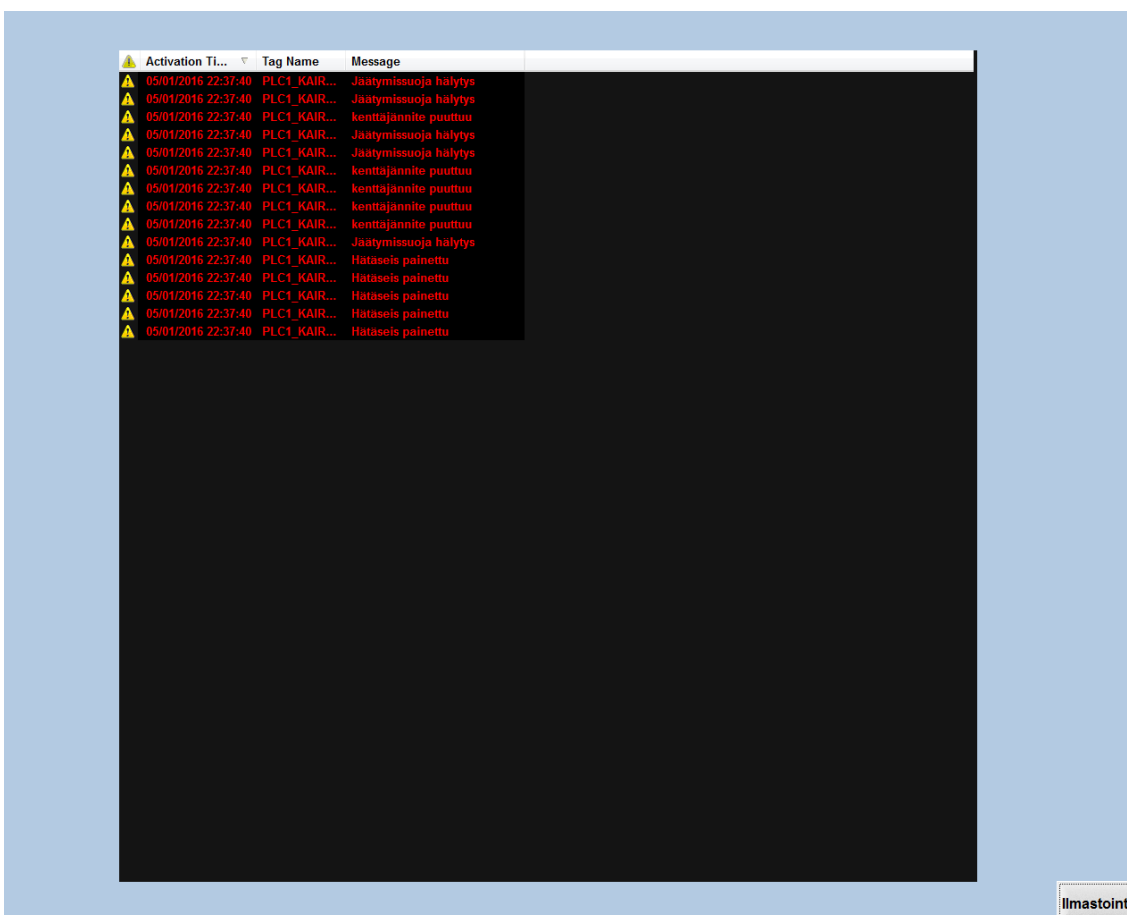
	Tag Name	Type	Limit	Message
	Filter text	(All)	Filter text	Filter text
1	PLC1_KAIR_Kair_DamperPHE_xError	Hi	0.000000	Ristiriitahälytys: LTO
2	PLC1_KAIR_Kair_xAlarmDifferentialPressureSen...	Hi	0.000000	Raja-arvo hälytys: LTO:n paine-ero
3	PLC1_KAIR_Kair_xAlarmPressureSensorExhaust	Hi	0.000000	Raja-arvo hälytys: Poistoilmakanavan paine
4	PLC1_KAIR_Kair_xAlarmPressureSensorSupply	Hi	0.000000	Raja-arvo hälytys: Tuloilmakanavan paine
5	PLC1_KAIR_Kair_xAlarmReturnTemperature	Hi	0.000000	Raja-arvo hälytys: Paluuveden lämpötila
6	PLC1_KAIR_Kair_xAlarmRoomTemperature	Hi	0.000000	Raja-arvo hälytys: Huoneen lämpötila
7	PLC1_KAIR_Kair_xAlarmSupplyAirTemperature	Hi	0.000000	Raja-arvo hälytys: Tuloilman lämpötila
8	PLC1_KAIR_Kair_xDamperExitAir	Hi	0.000000	Ristiriitahälytys: Jäteilmapelti
9	PLC1_KAIR_Kair_xDamperOutsideAir	Hi	0.000000	Ristiriitahälytys: Ulkoilmapelti
10	PLC1_KAIR_Kair_xEmergencyOff	Lo	0.000000	Hätäseis painettu
11	PLC1_KAIR_Kair_xMaintenanceFilterExhaust	Hi	0.000000	Poistosuodatin likainen
12	PLC1_KAIR_Kair_xMaintenanceFilterOutside	Hi	0.000000	Tulosuodatin likainen
13	PLC1_KAIR_Kair_xWarningPHE	Hi	0.000000	Huurtumis varoitus: LTO
14	PLC1_KAIR_Kair_xFrostmonitor	Lo	0.000000	Jäätymissuoja hälytys
15	PLC1_KAIR_kair_xMains	Lo	0.000000	kenttäjännite puuttuu

Kuva 15. Hälytykset task Indusoft Web Studio ohjelmassa.

Kuvassa 16 on aktiiviset hälytykset näkymä ja hälytyksen kuittauspainike valvontaohjelman päänäkymässä, ja kuvassa 17 on vikahistoria ikkuna valvontaohjelmassa. Pohjaväriksi on valittu musta, jolla saavutetaan suurempi kontrasti muihin komponentteihin nähden.



Kuva 16. Aktiivisena olevat hälytykset näkyvät myös päänäkymässä.



Kuva 17. Vikahistoria ikkunassa ensin on aika, seuraavaksi muuttuja ja viimeiseksi haluttu viesti vikatilanteesta.

7.2.5 Ongelmat

Lopussa suurimmaksi ja myös ylitsepääsemättömäksi ongelmaksi muodostui valittu ilmainen IWS Educational -lisenssi, sillä lisenssiä tilatessa me emme osanneet arvioida oikein, kuinka paljon muuttujia lopulta olisi tarvittu. Lisenssin sisältämät 150 muuttujaa lopulta pakottivat minut luopumaan osasta valvontaohjelmiston toiminnollisuuksia, kuten aikaohjelman säätämisen suoraan tekemästäni käyttöliittymästä. Pelkästään aikaohjelman tekeminen kokonaisuudessaan olisi tarvinnut vajaan 50 kappaletta muuttujia. Perustoiminnallisuuden jälkeen käytettävissä olevia muuttujia oli käytössä enää 40 kappaletta. Nyt Ilmastointikoneen aikaohjelma määrittäminen pitää suorittaa CoDeSys ohjelmointiympäristön oman visualisoinnin kautta, mikä sotii kaikki yhdessä paikassa ajattelua vastaan.

7.3 Ilmastointikoneen ohjausohjelmaan tehdyt muutokset

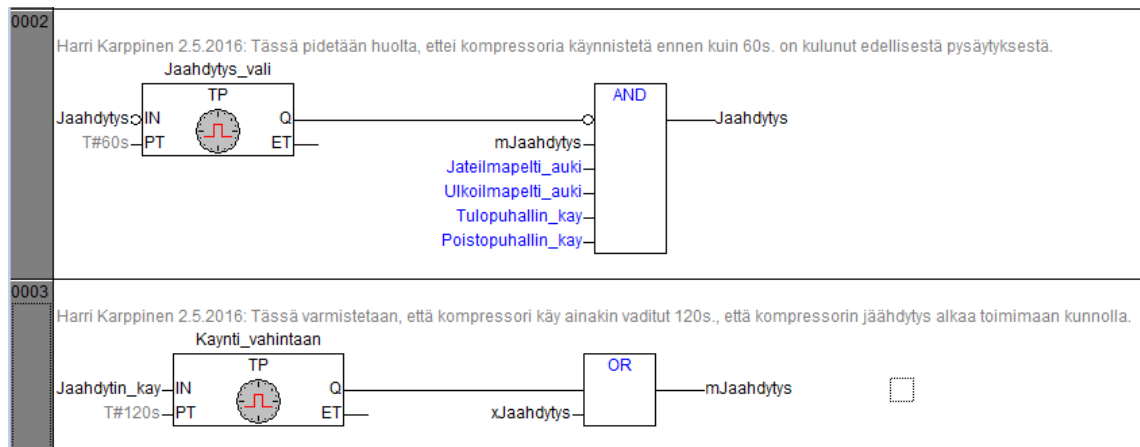
Ilmastointikoneeseen tarvittiin käsikäyttöä varten lisää turvallisuutta. Käsikäytöllä ilmastointikoneessa ei ole automatiikkaa joten on esimerkiksi mahdollista asettaa jäähdytyskompressori päälle tilanteessa jossa ulkoilmapellit ovat kiinni ja puhaltimekset eivät ole päällä. Tästä voi seurata kompressori-moottoriyksikön vaurioituminen ylikuumenemisen seurauksena. Kompressorin moottorissa on ylikuumenemisen ja ylivirran varalle bi-metallikytkin joka laukeaa ylikuumenemisen tai ylivirran seurauksena. Bi-metallikytkin palautuu normaaliasentoon jäähdyttyään automaattisesti (Danfoss 2015, 26).

Ohjelmasta löytyi jo esto, ettei jäähdytystä saanut uudelleen päälle jos edellisestä pysäytyksestä oli kulunut alle kolme minuuttia. Jäähdytyskompressorin valmistajan mukaan suositeltu kierto olisi, että kompressori-moottori yksikkö olisi päällä ainakin kaksi minuuttia kerralla jotta yksikön jäähdytys alkaisi toimimaan ja että yksikköä ei voitaisi käynnistää ainakaan minuuttiin edellisestä seisauksesta. Valmistaja myös varoittaa että ei ole suositeltavaa, että jäähdytysjärjestelmä käyn-

nistettäisiin yli kaksitoista kertaa tunnin aikana, sillä kahtatoista käynnistystä suurempi määrä voi vähentää moottori-kompressorin yksikön käyttöikä (Danfoss 2015, 35).

Jotta jäähdytyskompressorin olisi suojattu myös käsikäytöllä ylikuumentumisesta, lisäksi ohjelmaan ehdot, että jäähdytystä ei voi laittaa päälle elleivät ulkoilmapellit ole auki ja puhaltimet ole jo päällä (kuva 18).

Lisäykset ohjelmaan ovat yksinkertaisuudessaan yksi uusi muuttuja, kaksi uutta ajastinta ja JA- ja TAI -portit. Ohjelman yksinkertaisuuden ja ajastimien ominaisuuksien vuoksi ei jäähdytystä saada päälle prosessin ensimmäisen minuutin päällä olon aikana. Arvelisin ettei tästä muodostu todellista ongelmaa missään tapauksessa. Kuvassa neljä on ohjelmaan tehdyt uudet verkot.



Kuva 18. Ilmastointikoneen ohjelmaan tehdyt uudet verkot, joilla estetään valmistajan ohjeiden vastainen käyttö.

8 Tulokset

Asetettuihin tavoitteisiin päästiin melkoisen hyvin, mutta parannettavaakin tosin jäi. Kommunikointi valvontaohjelman ja ohjelmoitavan logiikan välillä toimii OPC-palvelimen kautta ja valvomo-ohjelma on toimiva ja alan konventioiden mukainen. Uuden käyttäjän prosessin ymmärtämistä helpotetaan muun muassa ryhmittelemällä yhteen liittyvät ohjaukset kokonaisuuksiksi, jota on tehostettu edelleen niiden kehystämällä, ja lisäämällä pieni kuvaus jokaiselle komponentille prosessissa. Käyttäjäkokeuksia valvomo-ohjelman toiminnollisuudesta ja ulkoasusta tarvittaisiin vielä lisää, mutta aikataulun tiukkuus ei mahdollistanut niiden keräämistä siinä määrin missä olisin halunnut. Muutaman käyttäjäkokeuksen perusteella sain hyviä korjausehdotuksia, jotka osaltaan paransivat lopputulosta. Saadut parannusehdotukset koskivat muun muassa symboleiden selkeämpää ryhmittelyä ja ruudulla näkyvien mittaustietojen päivitysnopeutta, jotta saavutettaisiin niiden parempi luettavuus.

Valvomo-ohjelmalla pystytään ohjaamaan ja valvomaan yksittäisten komponenttien tilaa, seuraamaan mittaus- ja tilatietoja antureilta ja komponenteilta sekä päänäkyvässä että trendinäkyvässä. Prosessiin määritellyt hälytykset viesteineen näkyvät ohjelmassa ja viat pystytään kuittaamaan ohjelman kautta. Vikaistoriaa pystytään valvomo-ohjelmassa tarkkailemaan omassa ikkunassa.

Lopulta projektiin valitun valvomo-ohjelman lisenssin jouduin toteamaan hieman liian rajoittuneeksi. Lisenssin sisältämät 150 muuttujaa olivat auttamattomasti liian vähän, jotta olisin voinut toteuttaa valvomo-ohjelmaan kaikki tahtomani ominaisuudet. Lisenssi hankittiin hyvissä ajoin ennen kuin aloitin opinnäytetyön tekemisen, eikä tarvittavien muuttujien määrää osattu ennakoita vielä siinä vaiheessa. Valvomo-ohjelman ohjaukset, trendit ja hälytykset söivät yhteensä 109 muuttujaa, mutta aikaohjelman lisääminen ilmastointikoneen automaattitilalle olisi vaatinut vajaat 50 muuttujaa lisää. Tämä olisi nostanut muuttujien määrän yli käytettävissä olevien 150:n.

Ohjelmoitavan logiikan ohjelmaan tehdyt muutokset toimivat halutulla tavalla. Ohjelmassa on estetty jäähdytyksen liian nopea päälle – pois -kierto kompressorin

valmistajan ohjeiden mukaisesti ja ohjelmaan on tehty lukitukset, ettei jäähdystystä voi käynnistää elleivät pellit ole auki ja puhallinmoottorit päällä.

9 Pohdinta

Opinnäytetyötäni voidaan suoraan hyödyntää opetuksessa ja opiskelijoiden ilmastointikoneeseen liittyvissä harjoituksissa. Valvomo-ohjelman ansiosta opiskelijoiden on helpompi ymmärtää mitä tapahtuu koneen sisällä sillä he pystyvät näkemään mittaustiedot ja peltien asennot suoraan ohjelmasta. Ilmastointikoneen ohjausohjelmaan tekemieni muutoksien ansiosta ilmastointikoneen käyttö on turvallisempaa, sillä sitä ei jäähdytyksen osalta voida ajaa tilaan jossa se voisi vahingoittua. Tekemäni työ OPC-palvelimen kanssa helpottaa tulevaisuudessa ja olemassa olevissa projekteissa, joissa käytetään OPC-standardin mukaista kommunikointia.

Työn aikana tuli vastaan useita mielenkiintoisia kohteita, joita olisi ollut mukava lähteä tutkimaan lisää. Valvomo-ohjelmaksi valitussa Indusoft Web Studiossa olisi mahdollista laajentaa sitä web käyttöliittymällä. Web käyttöliittymällä voidaan ottaa yhteyttä valvomo-ohjelmaan internetin yli ja valvoa ja ohjata ilmastointikoneita etänä. IWS mahdollistaisi samanlaisen yhteyden muodostamisen myös mobiililaitteilla. Nämä helpottaisivat ohjelman käyttöä opetuksessa entisestään, sillä tällöin ei oltaisi enää sidottuna yhteen fyysiseen paikkaan.







Valvomo-ohjelmiston lisenssiä pykälällä korotettaessa käytössä oleva kokonaisuuttujamäärä nousisi jo kolmeensataan, joka mahdollistaisi jo aikaohjelman ja PID-, kanavapaine ynnä muiden sellaisten asetusten määrittämisen itse valvomo-ohjelmasta. Nyt ne täytyy määrittellä CoDeSys -ohjelmointiympäristössä.

Tulevaisuudessa ohjelmaa voidaan kehittää jatkossa käyttäjäpalautteen mukaan lisää ja pohtia; täytyykö joitakin osia valvomo-ohjelmasta muuttaa, jotta siitä tulisi parempi. Yksi asia jota kannattaisi miettiä tulevaisuudessa, on se tulisiko ohjelmaan liittää myös muita valvontakohteita.

Lähteet

- 3S-Smart Software Solutions Gmbh. 2006. CoDeSys OPC-Server V2.0 Installation and usage. Saksa: 3S-Smart Software Solutions Gmbh.
- 3S-Smart Software Solutions Gmbh. 2016. Products. OPC Server. 3S-Smart <https://www.codesys.com/products/codesys-runtime/opc-server.html> 4.5.2016.
- Danfoss. 2015. Commercial Compressors. Compressors for refrigeration. Danfoss scroll compressors. Documents. Guideline Danfoss scroll MLM MLZ series –GB –US. <http://www.ra.danfoss.com/TechnicalInfo/Literature/Manuals/17/frcc.pc.015.a7.22-mlm-mlz-apr2015.pdf> 8.5.2016.
- The international organization for standardization. 1998. Search. Iso 9241-11:1998 Ergonomic requirement for office work with visual display terminals (VDTs) – Part 11: Guidance on usability. <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-1:v1:en> 8.5.2016.
- OPC Foundation. 2016a. About. What is OPC?. <https://opcfoundation.org/about/what-is-opc/> 24.4.2016.
- OPC Foundation. 2016b. About. OPC Technologies. Unified Architecture. <https://opcfoundation.org/about/opc-technologies/opc-ua/> 1.5.2016.
- Schneider Electric Software. 2016a. Products & Downloads. HMI Software. Indusoft Web Studio. <http://www.indusoft.com/Products-Downloads/HMI-Software/InduSoft-Web-Studio> 4.5.2016.
- Schneider Electric Software. 2016b. Wonderware Indusoft Web Studio 8.0 Technical Reference. Internal structure and data flow. Ranska: Schneider Electric Software.
- Schneider Electric Software. 2016c. Wonderware Indusoft Web Studio 8.0 Technical Reference. About Tags and the Project Database. Ranska: Schneider Electric Software.
- Seppänen O 1996. Ilmastointitekniikka ja sisäilmasto. Helsinki: Suomen LVI-yhdistysten liitto.
- Silvan, J, Kauppila, K, Kaappola, E. 2014. Jäähdytyslaitteet. Teoksessa Sandberg, E. (toim.) Sisäilmasto ja ilmastointijärjestelmät Ilmastointitekniikka 1. Helsinki: Talotekniikka-Julkaisut Oy, 241–285.
- Sinkkonen I, Kuoppala H, Parkkinen J, Vastamäki R 2009. Käytettävyyden psykologia. Helsinki: Adage Oy.
- Suomen automaatioseura. 2010. Valvomo: suunnittelun periaatteet ja käytännöt. Helsinki: Suomen automaatioseura.
- Suomen automaatioseura ry. 2016. Toiminta. OPC-toimikunta. <http://www.automatioseura.fi/index/toiminta.php?id=1032&sivu=opcesittely> 1.5.2016.
- Sähkötieto ry. 2008. ST-käsikirja 22 Kiinteistöjen valvomojärjestelmät. Espoo: Sähköinfo.
- Sähkötieto ry. 2012. ST-käsikirja 17 Rakennusautomaatiojärjestelmät. Espoo: Sähköinfo.

Ilmastointijärjestelmän ilmavirtojen värit

-  Tuloilma
-  Poistoilma
-  Likainen poistoilma
-  Ulkoilma
-  Jäteilma ulos
-  Siirtoilma, Kierrätysilma

(Sandberg Esa Sisäilmasto ja ilmastointijärjestelmät s.405 soveltaen)

Ilmastointilaitteiden ja varusteiden piirrosmerkkejä

Suodatin



Lämmitin



Jäähdytin



Lämmöntalteenotin



Sulku- tai Sätöpelti



Palopelti



Puhallin, yleismerkki



Puhallin, 2- nopeuksinen



Puhallin, jatkuvasäätäinen



Pumppu, yleismerkki

(ST Käsikirja 17 (2012) s.267 mukailten)

KARELIA-AMMATTIKORKEAKOULU
Sähkötekniikan koulutusohjelma

Harri Karppinen

CoDeSys OPC palvelimen määrittely

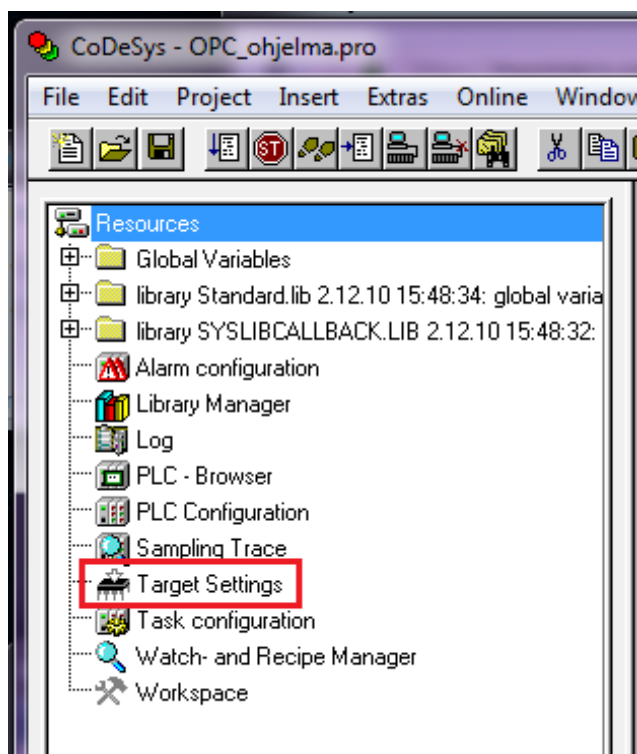
Ohje
Huhtikuu 2016

Johdanto

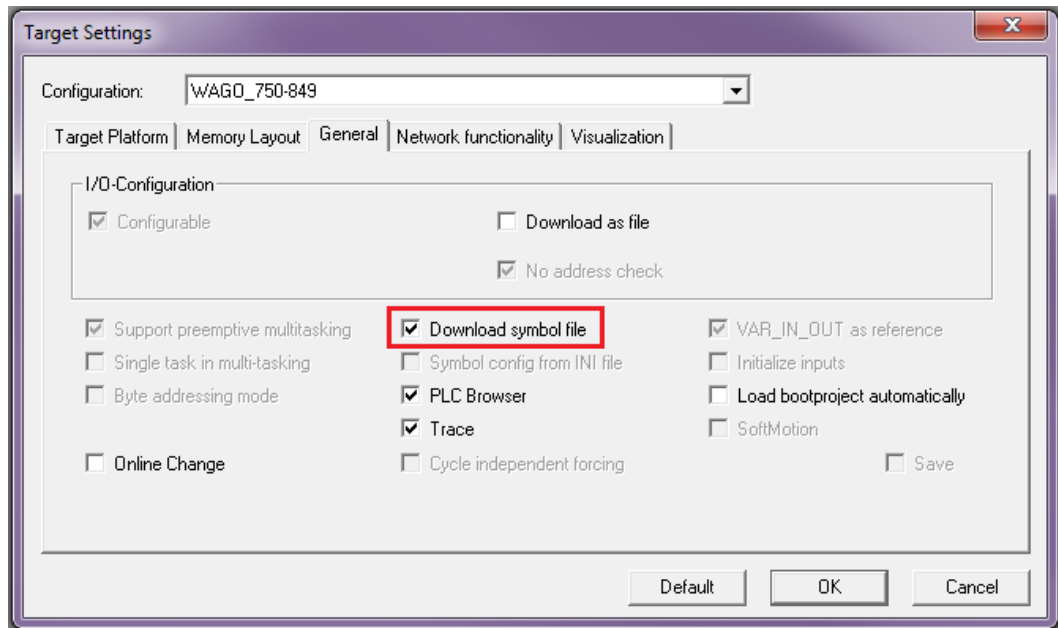
Tässä ohjeessa käsitellään CoDeSys OPC-serverin v.2,3 konfigurointi kohta kohalta toimimaan Wago-PLC ja client-ohjelman välillä. Tässä ohjeessa oletetaan, että osaat jo ohjelmoida CoDeSys ohjelmointiympäristössä, sekä määrittää yhteysasetukset TCP/IP protokollalla.

CoDeSys 2,3 konfigurointi

CoDeSys- ohjelmassa ensin valitaan Resources > Target Settings > General > Download symbol file.

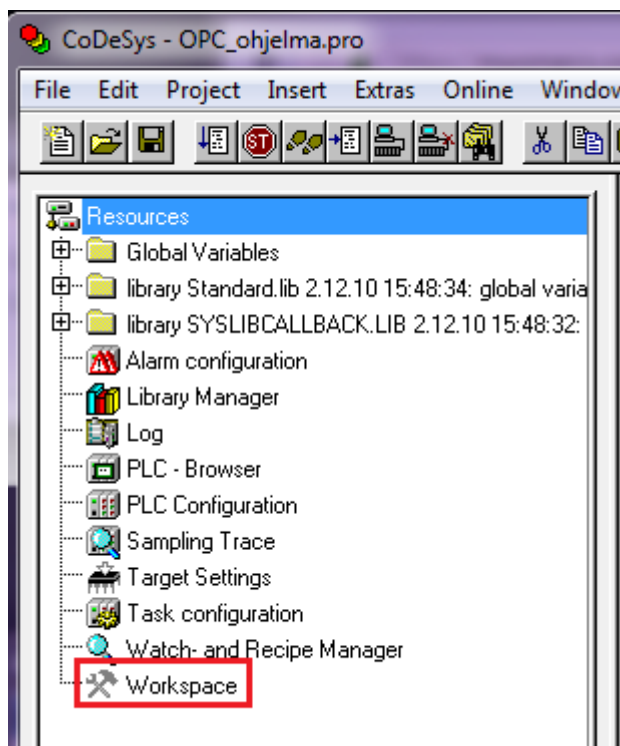


Kuva 1. Target Settings löytyy Resources välilehdeltä



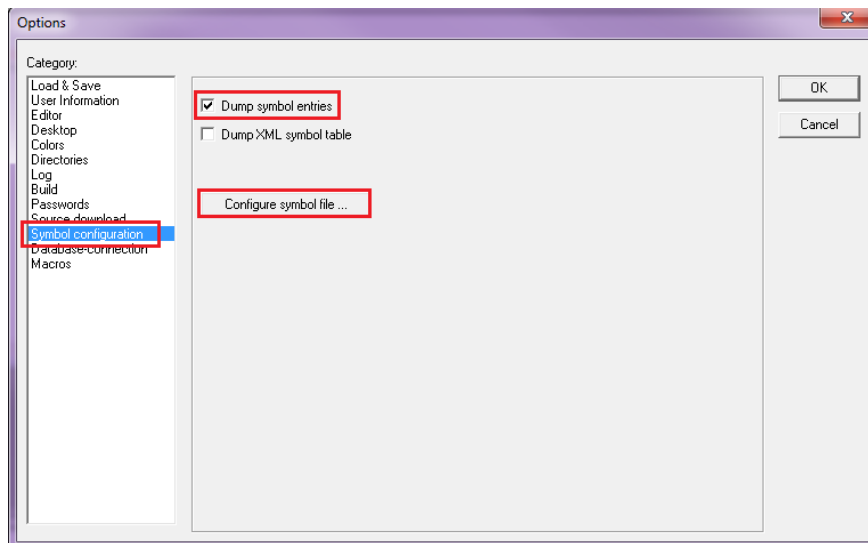
Kuva 2. Tarkasta, että Download symbol file on valittu

Seuraavaksi määritellään Symbolien asetukset Resources > Workspace > Symbol configuration.



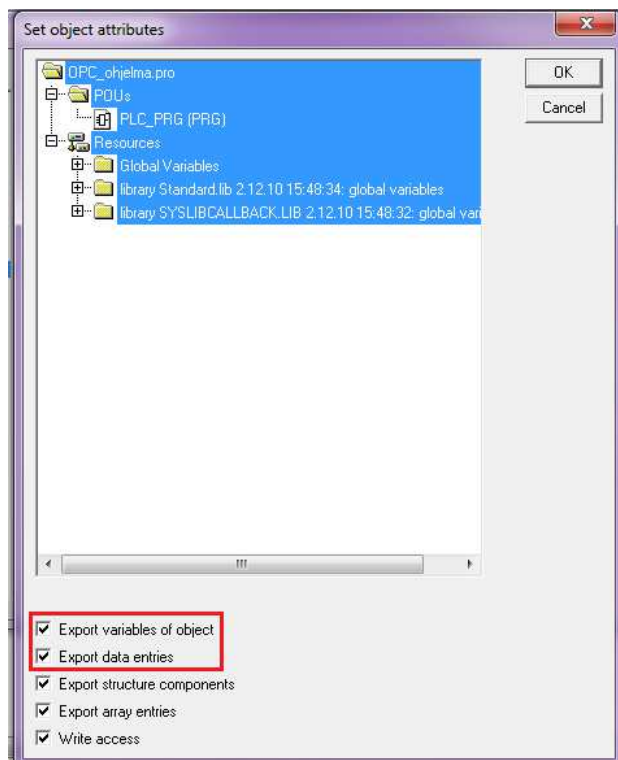
Kuva 3. Workspace löytyy Resources välilehdeltä

Varmista, että Dump symbol entries on valittu ja tämän jälkeen paina Configure symbol file painiketta.



Kuva 4. Symbol Configuration

Set object attributes ikkunassa varmista, että Export variables of objects ja Export data entries ovat valittu eikä valinta ole harmaana. Hakemistopuusta pystyt valitsemaan haluamasi komponentit joidenka haluat tuotavan OPC serverille.



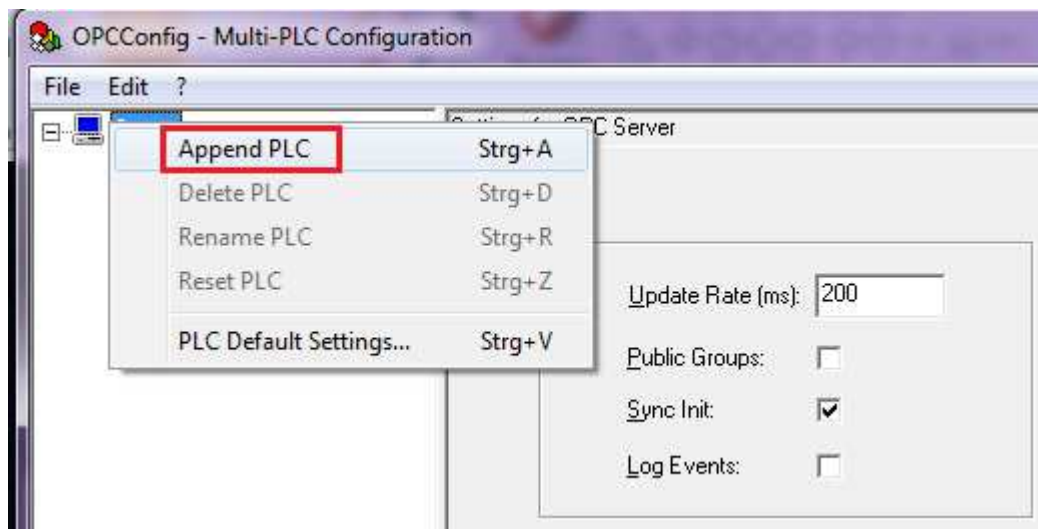
Kuva 5. Set object attributes ikkuna

Määritä yhteysasetukset normaalisti TCP/IP kommunikointia varten Online – Communication Parameters valikosta. Siirrä ohjelma Wago – PLC:lle.

CoDeSys OPC Configurator

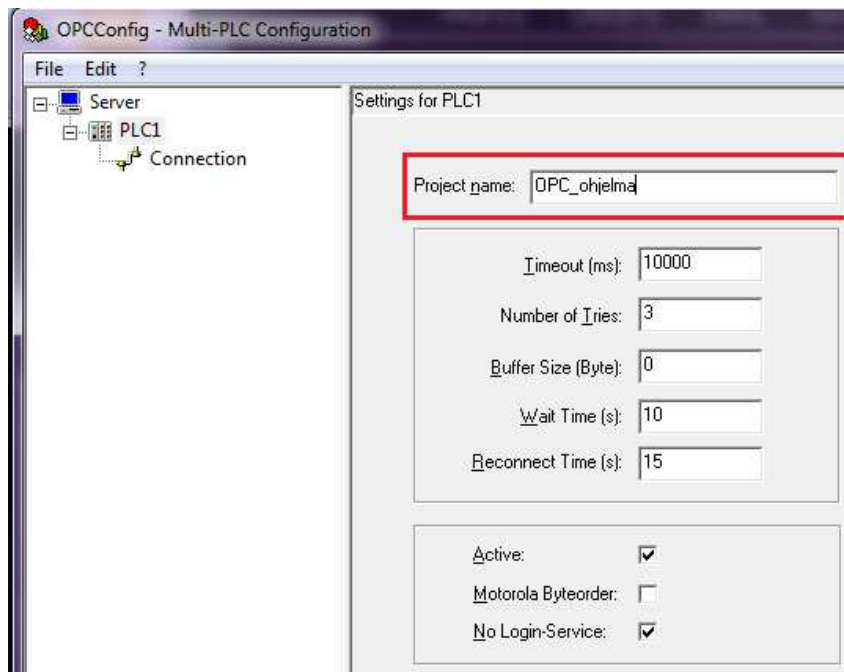
Avaa CoDeSys OPC Configurator. Ohjelma löytyy Windowsin käynnistävalikosta Wago Software > CoDeSys > Communication alta.

Avautuvassa ikkunassa valitse Server vasemmasta laidasta ja hiiren oikealla napilla avautuvasta valikosta lisää uusi PLC "Append PLC". Loput asetukset voit tässä näkymässä jättää ennalleen.



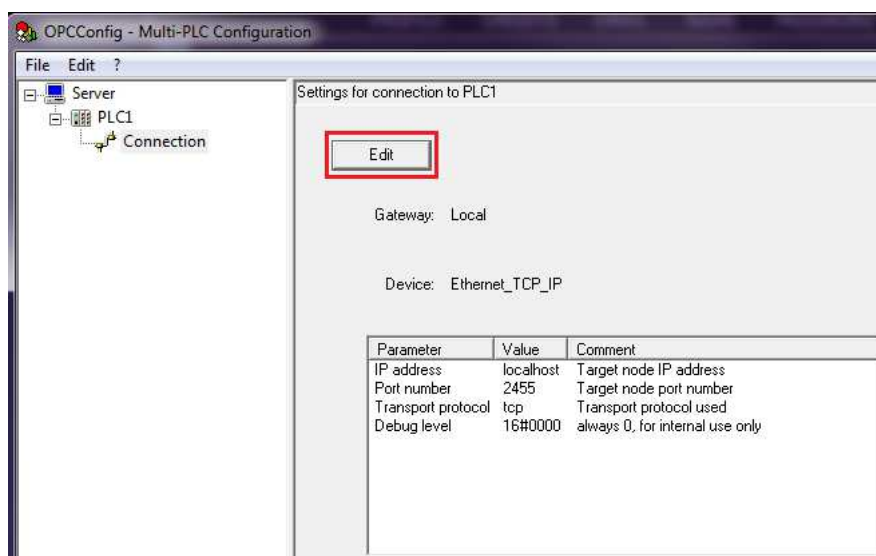
Kuva 6. Uuden PLC:n lisääminen OPCConfig- ohjelmassa

Nyt luodussa PLC1 määritä Project name. Kirjoita tähän Wago- PLC:lle siirretyn projektin nimi.



Kuva 7. Varmista, että Project name on sama kuin kontrollerilla olevan projektin nimi

Connections alta valitse Edit, ja avautuvasta ikkunasta valitse Wago – kontrollerille määritelty yhteys. Tallenna tekemäsi määrittäykset ja tämän jälkeen ohjelman voi sulkea.



Kuva 8. Yhteyden määrittäminen

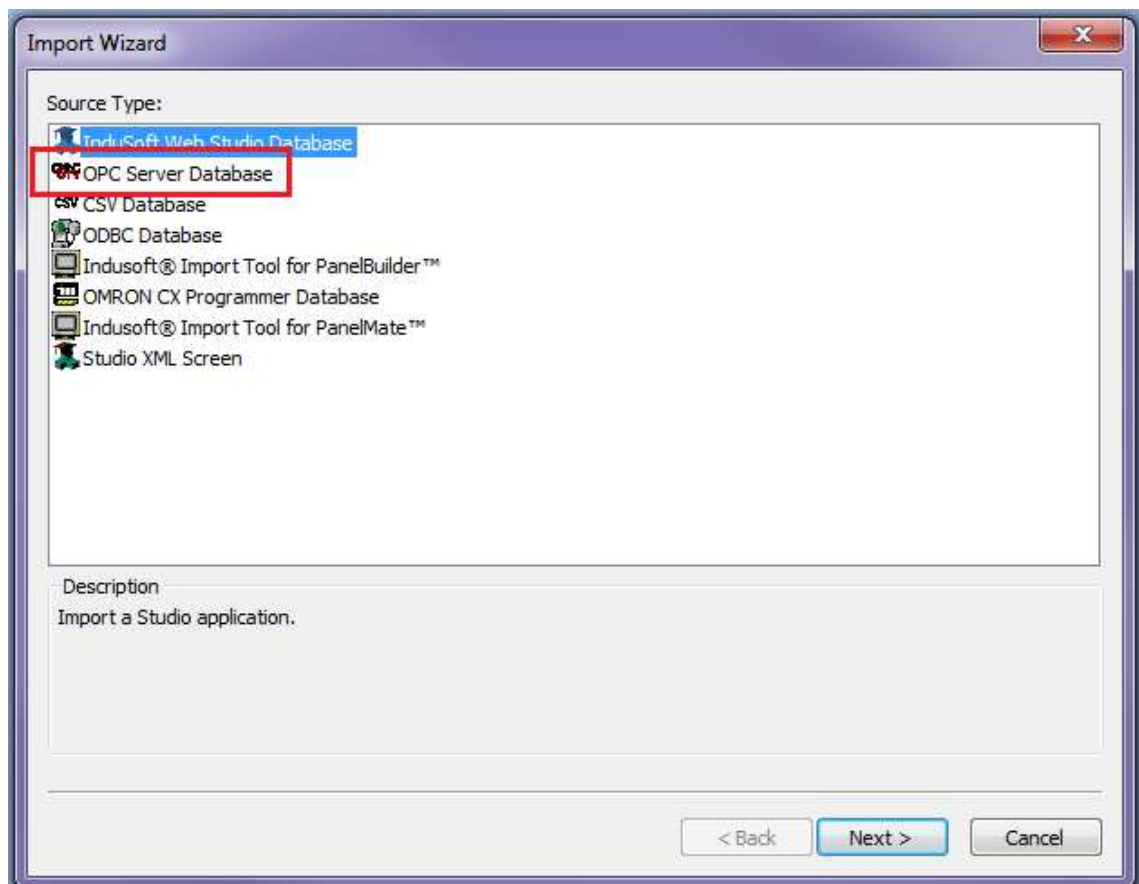
OPC serverin lisääminen client ohjelmaan

Client-ohjelmana tässä ohjeessa käytän Wonderware Indusoft Web Studio 8,0. Yksinkertaisimmillaan yhdistäminen onnistuu, kun client ja OPC server ovat samalla tietokoneella. Indusoft Web Studiossa Home välilehdeltä valitse Tools > Import Wizard.



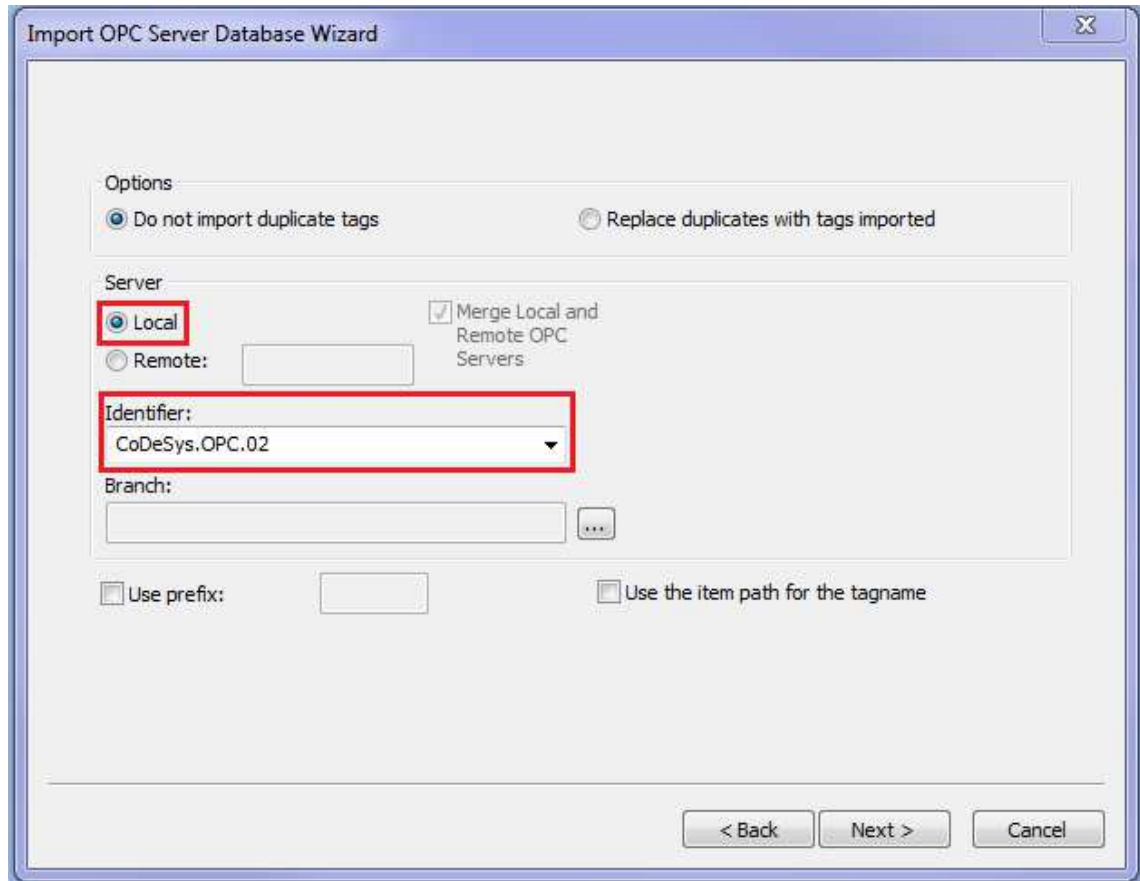
Kuva 9. Import Wizard

Avautuvasta ikkunasta valitse OPC Server Database ja paina next.



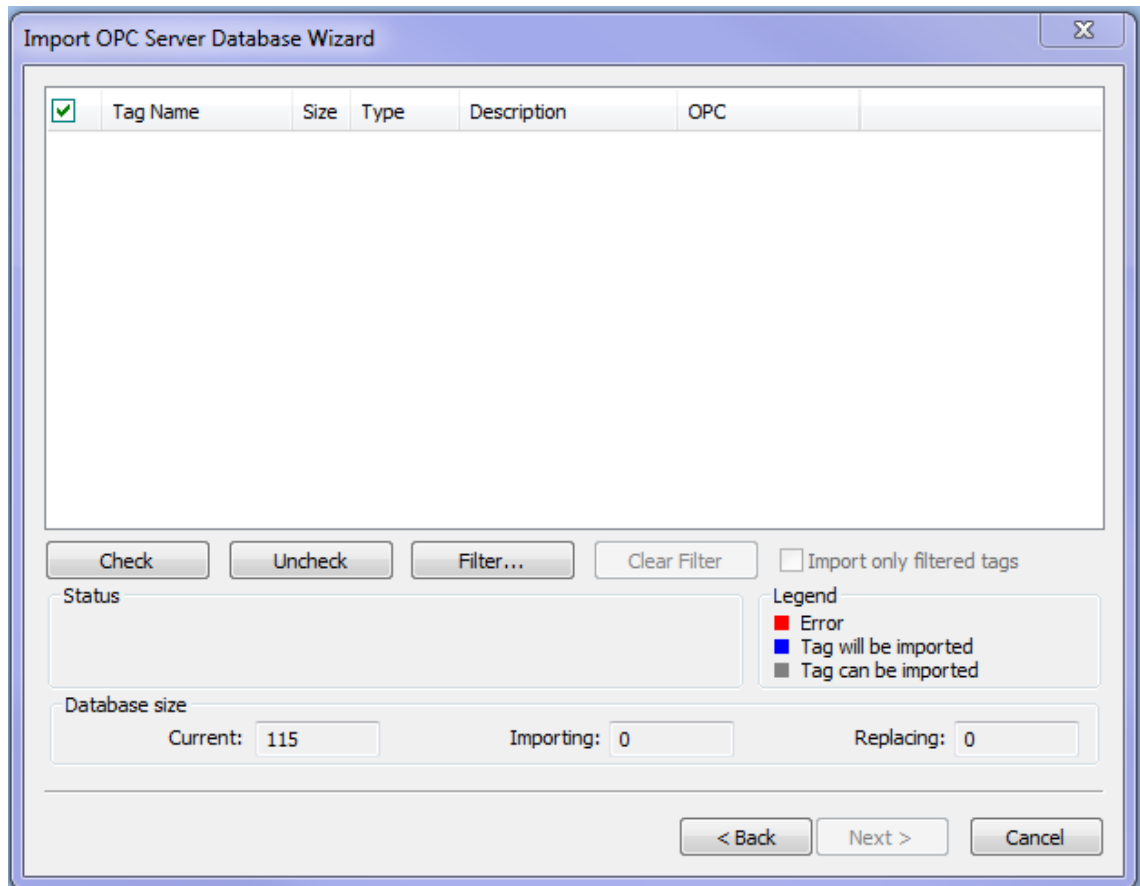
Kuva 10. Import Wizard ikkuna

Import OPC Server Database Wizard ikkunassa valitse Serveriksi local ja etsi Identifiers alta käyttämäsi OPC serveri. Tässä tapauksessa se on Co-DeSys.OPC.02 ja tämän jälkeen paina next.



Kuva 11. Import OPC Server Database Wizard ikkuna

Avautuvassa ikkunassa pystyt valitsemaan haluamasi tagit jotka tahdot tuoda ohjelmaan.



Kuva 12. Tässä ikkunassa voit listalta valita haluamasi tag:it jotka haluat tuoda ohjelmaan.

Valittuasi haluamasi tagit Import Wizard luo Indusoft Web Studio Project Explorerin Comm välilehteen uuden OPC DA 2.05 > Imported tags. Tässä näkyvässä pystyt muokkaamaan muun muassa Read Update Rate:a ja määritellyn yhteyden nimeä. Uusien tagien tuominen tälle listalle ei vaadi import wizardin uudelleen ajamista. Riittää, että tupla napautat ensimmäistä tyhjää kohtaa Item listalla, jolloin avautuu näkymä OPC serverillä olevista muuttujista. Aina kun lisäät uusia muuttujia OPC serveriltä pitää myös määrittää sille Tag Name ja muuttujan tyyppi Web Studiassa.

Project Explorer

Project: KAIR_ilmastointikone.APP

- Drivers
 - OPC DA 2.05
 - 1: Imported tags
 - 2: Hälytykset
 - OPC UA
 - Connections
 - Redundancy Group
 - OPC .Net
 - OPC XML/DA
 - TCP/IP
 - DDE

OPCCL001.OPC

Description: Imported tags Server Identifier: CoDeSys.OPC.02 Disable:

Read Update Rate (ms): 2000 Percent Deadband: Status:

Remote Server Name: Browse... Read before writing
 Read after writing
 Accept Tag Name in the Item column

	Tag Name	Item	Scan
	<input type="text"/> Filter text	<input type="text"/> Filter text	<input type="text"/> (All) ▼
1	PLC1__Automaatti	PLC1:Automaatti	Always ▼
2	PLC1__Hata_seis	PLC1:Hata_seis	Always ▼
3	PLC1__Huonelt_A	PLC1:Huonelt_A	Always ▼
4	PLC1__Jaahdytin_kay	PLC1:Jaahdytin_kay	Always ▼
5	PLC1__Jaahdytys	PLC1:Jaahdytys	Always ▼
6	PLC1__Jaahdytys_O	PLC1:Jaahdytys_O	Always ▼
7	PLC1__Jaatymissuoja	PLC1:Jaatymissuoja	Always ▼
8	PLC1__Jateilma_It	PLC1:Jateilma_It	Always ▼
9	PLC1__Jateilmapelti	PLC1:Jateilmapelti	Always ▼
10	PLC1__Jateilmapelti_auki	PLC1:Jateilmapelti_auki	Always ▼
11	PLC1__Jateilmapelti_kiinni	PLC1:Jateilmapelti_kiinni	Always ▼
12	PLC1__Kenttajannite	PLC1:Kenttajannite	Always ▼
13	PLC1__Kuittaus	PLC1:Kuittaus	Always ▼
14	PLC1__Lammityspumppu	PLC1:Lammityspumppu	Always ▼
15	PLC1__Lammityspumppu_k...	PLC1:Lammityspumppu_kay	Always ▼
16	PLC1__Lammitysventtiili_M	PLC1:Lammitysventtiili_M	Always ▼
17	PLC1__Lammitysventtiili_O	PLC1:Lammitysventtiili_O	Always ▼
18	PLC1__Lammitysvesi_It	PLC1:Lammitysvesi_It	Always ▼
19	PLC1__LTO_paine	PLC1:LTO_paine	Always ▼
20	PLC1__LTO_pelti_M	PLC1:LTO_pelti_M	Always ▼

Kuva 13. Esimerkki importatuista tag:eista.