



JANI RAINTOLA

Indusoft ohjauslohkot

SÄHKÖ- JA AUTOMAATIOTEKNIIKAN TUTKINTO-OH-
JELMA
2022

Tekijä(t) Raintola, Jani	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä 08/2022
	Sivumäärä 33	Julkaisun kieli Suomi
Julkaisun nimi Indusoft ohjauslohkot		
Tutkinto-ohjelma Sähkö- ja automaatiotekniikka		
Tiivistelmä <p>Opinnäytetyön tavoitteena oli luoda AFRY Finland Oy:lle indusoft web studio valvomo-ohjelmistoon valmiita ohjauslohkoja, joita voitaisiin käyttää jatkossa uusissa automaatioprojekteissa.</p> <p>Tehdyt ohjauslohkot valittiin yrityksen automaatioprojekteissa yleisimmin käytettävistä komponenteista. Ohjauslohkot tehtiin suora- ja taajuusmuuttajaohjatulle moottorille, automaattiselle venttiilille ja analogiselle mittalaitteelle. Ohjauslohkojen tarkoitus on sujuvoittaa käyttöliittymän tekemistä sekä vähentää siihen käytettävää aikaa. Valmiilla ohjauslohkoilla tehtäessä saadaan uusia toimilaitteita lisättyä projektiin helposti, eikä jokaiselle laitteelle tarvitse erikseen määrittää omia parametreja.</p> <p>Ohjauslohkoille tehtiin myös käyttöohjeet, jossa on ohjeistettu ohjauslohkojen lisääminen uuteen projektiin, sekä esitetty miten ohjauslohkon saa liitettyä automaatio-ohjelmistoon.</p>		
Avainsanat Käyttöliittymät, Indusoft Web Studio		

Author(s) Raintola, Jani	Type of Publication Bachelor's thesis	Date 08/2022
	Number of pages 33	Language of publication: Finnish
Title of publication Indusoft control blocks		
Degree programme Electrical- and automation technology		
Abstract The purpose of the thesis was to create ready control blocks with Indusoft web studio-user interface software for AFRY Finland Oy, which could be used in new automation projects in the future. Created control blocks were chosen from the most used components in the company's automation projects. The control blocks were made for a direct and frequency converter controlled motors, automatic valve and for analog measurement. The purpose of the control blocks is to facilitate the creation of user interface and reduce the time spent on it. By using ready-made control blocks, it is easy to add new devices to a new project, and it is not necessary to define each parameter separately. Instructions were also made for the control blocks, which includes instructions for adding control blocks to a new project, as well as presented how to connect the control block to the automation software.		
Keywords User interfaces, Indusoft Web Studio		

SISÄLLYS

1 JOHDANTO.....	6
2 TYÖN TOIMEKSIANTAJA	6
3 TYÖN AIHEEN RAJAUS.....	7
4 INDUSOFT OHJELMISTO.....	7
5 KÄYTTÖLIITTYMÄN OHJAUSLOHKOT.....	8
5.1 Ohjauslohkon pop up -ikkuna.....	8
5.2 Ohjauslohkojen tagit	10
5.3 Suorakäyttöinen moottori	11
5.4 Taajuusmuuttajaohjattu moottori	12
5.5 Automaattiventtiili	12
5.6 Analoginen mittalaite	12
5.7 Ohjauslohkon komennot.....	12
5.7.1 Käsiäjon painike.....	15
5.7.2 Automaattiajon painike	15
5.7.3 Moottorin käynnistys- ja venttiilin avauspainike.....	16
5.7.4 Moottorin pysäytys- ja venttiilin sulkemispainike.....	17
5.7.5 Ikkunan sulkemispainike	17
5.7.6 Text data linkitykset.....	18
5.8 Ohjauslohkon tilatiedot	19
6 KÄYTTÖOHJEET	22
6.1 Ohjausikkunan avaaminen symbolista	25
6.2 Yhdistäminen logiikkaan.....	27
7 YHTEENVETO.....	28
LÄHTEET	

KÄYTETYT LYHENTEET

IWS Indusoft Web Studio

Tag Merkki / tunniste

1 JOHDANTO

Opinnäytetyön tarkoituksena oli suunnitella Indusoft Web Studio, v7.1-ohjelmalla valmiit ohjauslohkot asiakasyrityksen yleisimmin käytettäviin komponentteihin. Ohjauslohkot tehtiin niin suora- kuin taajuusmuuttajaohjatuille moottoreille, automaattiventtiileille, sekä mittalaitteille. Valmiita ohjauslohkoja voidaan jatkossa käyttää osana käyttöliittymän suunnittelua ja toteutusta. Ohjauslohkojen toiminta haluttiin toteuttaa yleisillä tageilla, joita käyttämällä voidaan saada ohjelman tarvitsema lisenssi kokoa pienemmäksi, ja tätä kautta mahdollisesti säästää lisenssin kustannuksissa.

Käyttöliittymä, tässä tapauksessa ohjauslohkot, ovat rajapinta käyttäjän ja ohjelmiston välillä. Hyvin tehty käyttöliittymä on selkeä ja helppokäyttöinen, josta jää käyttäjälle positiivinen käyttäjäkokemus. Hyvän käyttöliittymän tekeminen vaatii suunnittelijalta visuaalista osaamista ja näkemystä.

2 TYÖN TOIMEKSIANTAJA

AFRY on yksi Euroopan johtavista konsultointi- ja suunnitteluyhtiöistä. AFRY on perustettu helmikuussa 2019, kun Ruotsissa perustettu ÅF osti Pöyryn. Suomessa AFRY työllistää noin 2800 henkilöä 28 eri paikkakunnalla. Maailmanlaajuisesti työntekijöitä on noin 17000, 50 eri maassa. Pääkonttori sijaitsee Tukholmassa.

Liiketoiminta jakautuu teollisiin ja digitaalisiin ratkaisuihin, konsultointiin, rakennettuun ympäristöön, sekä energia- ja prosessiteollisuuteen. Vastuullisuus ja kestävät ratkaisut ovat osa kaikkea yrityksen liiketoimintaa. (AFRY Oy [www](http://www.afry.com)-sivut 2022)

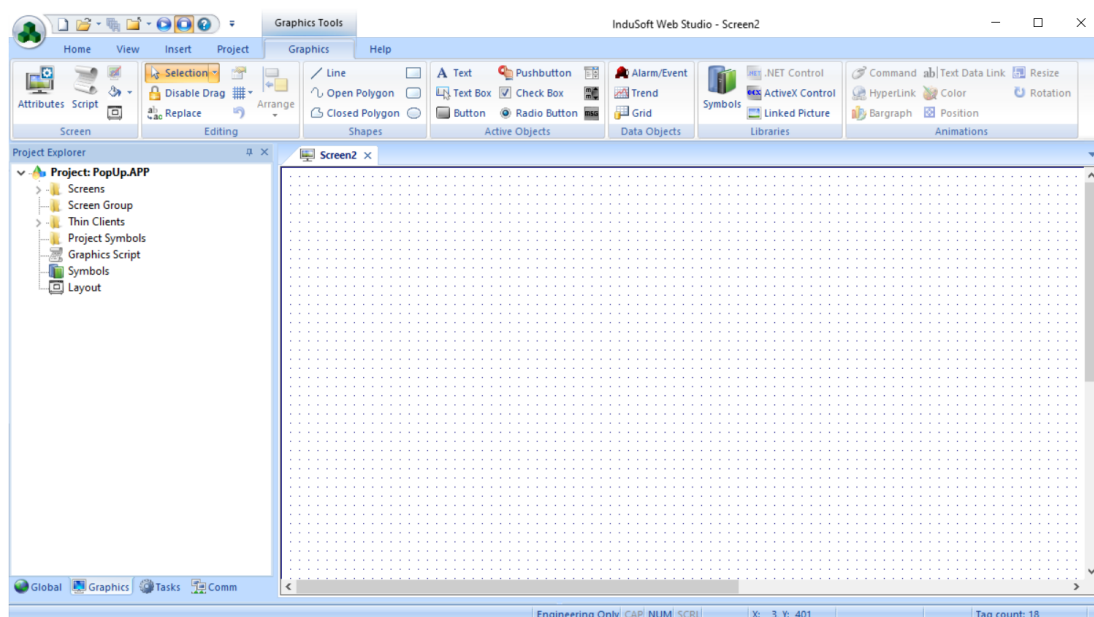
3 TYÖN AIHEEN RAJAUS

Työhön sisällytettiin IWS ohjelmalla tehtävät lohkot, sisältäen omat ohjauslohkot suora- ja taajuusmuuttajaohjatuille moottoreille, automaattiventtiilille ja mittalaitteelle. Työssä kerrotaan ja havainnollistetaan yksityiskohtaisesti, miten mikäkin toiminnallisuus ja ohjaus on tehty. Työhön kuului myös käyttöohjeet, joiden avulla käyttäjä pystyy tuomaan valmiita ohjauslohkoja uuteen projektiin, sekä määrittelemään ohjauslohkojen toiminnan kannalta välttämättömät tagit ja ohjaukset.

IWS ohjelmalla tehtyjen ohjauslohkojen on tarkoitus toimia yhdessä automaatio-ohjelman kanssa. Työn tuloksena syntyneitä ohjauslohkoja tullaan käyttämään Siemensin Tia Portal -ohjelmistoympäristössä. Logiikkapuolen lohkot ja ohjelmointi jätettiin oppinnäytetyön aihe rajauksen ulkopuolelle.

4 INDUSOFT OHJELMISTO

Indusoft Web Studio (Kuva 1), nykyisin uudelta nimeltään AVEVA Edge, on ympäri maailmaa käytössä oleva käyttöliittymäohjelmisto. IWS on lisenssilainainen ohjelmisto, jossa lisenssi määrittää muun muassa ohjelmassa käytössä olevien tagien määrän. Tässä työssä keskityttiin samalla tekemään ohjelman ohjauslohkot mahdollisimman pienellä tagien määrällä.



Kuva 1. IWS ohjelman päänäkymä.

IWS käyttöliittymä on yhteensopiva seuraavien käyttöjärjestelmien kanssa: Windows 7,8 ja 10, Windows Server 2008, 2012 ja 2016, Windows Embedded Standard 7 ja 8, Windows 10 IoT Enterprise, sekä Windows Embedded Compact 6, 7 ja 2013 versiot.

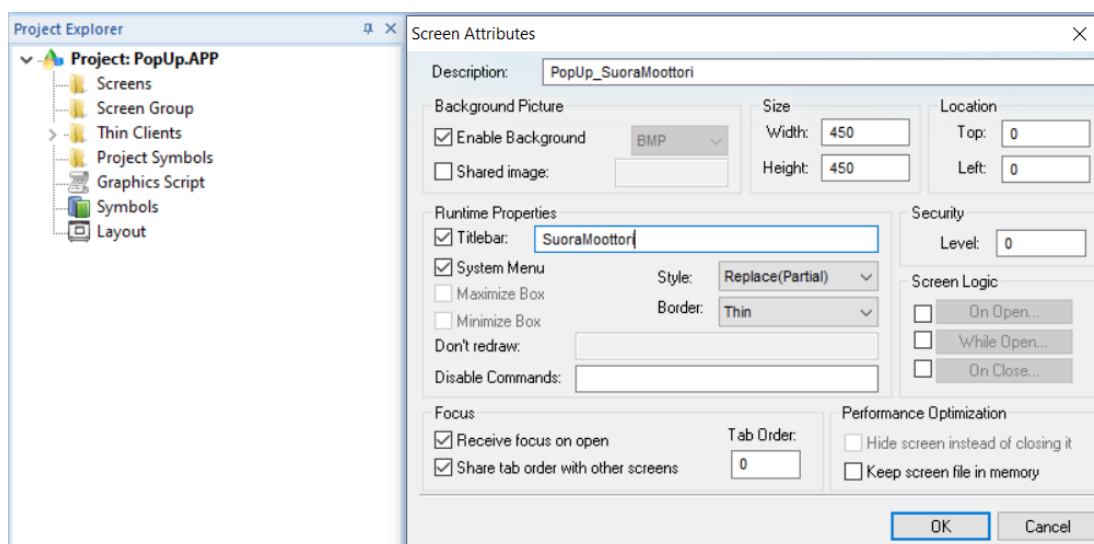
5 KÄYTTÖLIITTYMÄN OHJAUSLOHKOT

Ohjauslohkojen toteutus on tehty silmällä pitäen niiden mahdollisimman yksinkertaista käyttöä, selkeät ja järjestelmälliset komennot, sekä käytetty maltillisia taustavärejä. Ohjauslohkoihin on myös lisätty väritehosteita kertomaan kyseisen komponentin oleellista tilatiedoista, kuten esimerkiksi päällä/pois-tilatieto. Rakenteessa on otettu huomioon myös lohkon mahdollinen muokkaaminen, eli ylimääräiset toiminnot voidaan poistaa kokonaan näkyvistä niin, että poisto ei vaikuta muiden ohjausten toimintaan.

5.1 Ohjauslohkon pop up -ikkuna

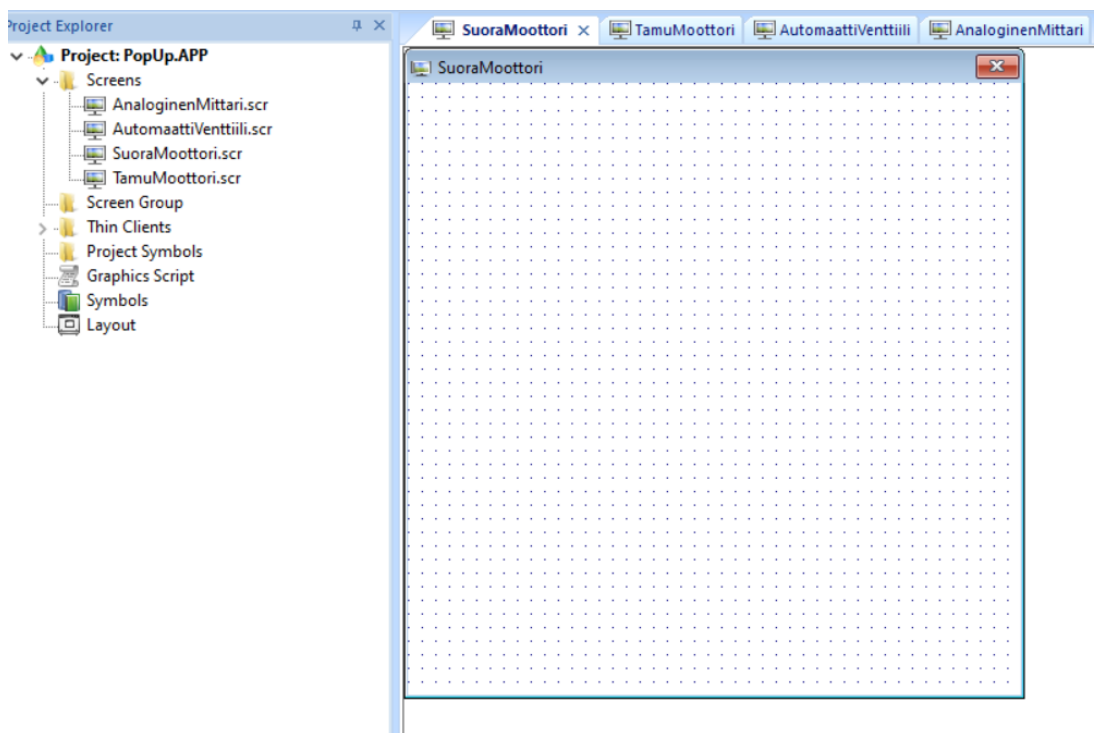
Kaikki ohjauslohkojen näkymät määritettiin ”Home”-välilehdeltä, projekti -kansion alta, kohdasta ”Screens” → ”Insert”. Ohjelma avaa ”Screen attributes”-ikkunan, jossa

saadaan määritettyä ikkunan tarkemmat tiedot (Kuva 2). Ikkunan tietoja saa myös jälkikäteen muokattua sopivammaksi ”Graphics”-välilehdeltä, kohdasta ”Attributes”. Esimerkiksi ikkunan todellinen koko on helpompi määrittää jälkikäteen, kun kaikki tarvittavat toiminnot ja kuvakkeet on lisätty ja ikkuna voidaan rajata oikean koiseksi. Ohjauslohkojen ikkunoiden koko määritetään pikseleinä ”Size”-valikosta syöttämällä halutut korkeus- ja leveysarvot. Location ”Top” määrittää ikkunan sijainnin X-akselin mukaan pystysuunnassa ja ”Left” Y-akselin mukaan poikittaissuunnassa, eli 0-arvoilla ikkuna aukeaa origosta, vasempaan yläkulmaan. ”Runtime Properties” valikosta saadaan määritettyä avautuvan ikkunan käytönaikaisia asetuksia, kuten lisätä ikkunaan yläpalkki otsikolla ja sulkemismahdollisuudella, tai määrittää voiko ikkunan kokoa muokata kulmasta raahaamalla.



Kuva 2. Näkymä uuden ikkunan parametreistä.

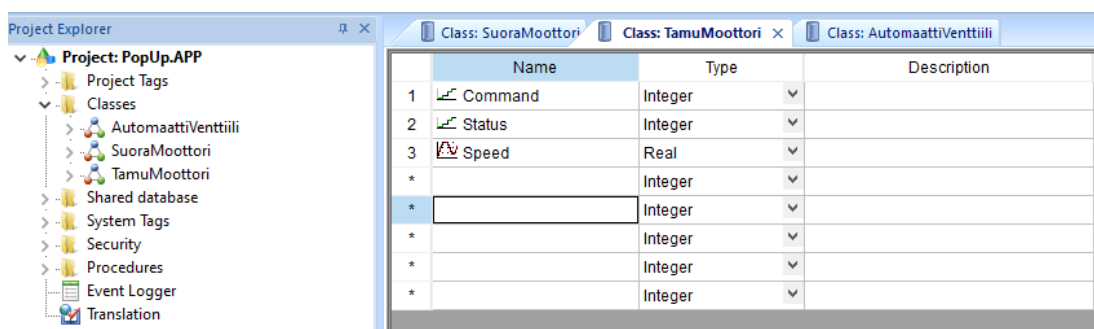
Valmiiksi määritetyt ohjauslohkojen pohjat ovat avattavissa ja muokattavissa ”Screens” kansion alta (Kuva 3).



Kuva 3. Näkymä luoduista ikkunoista.

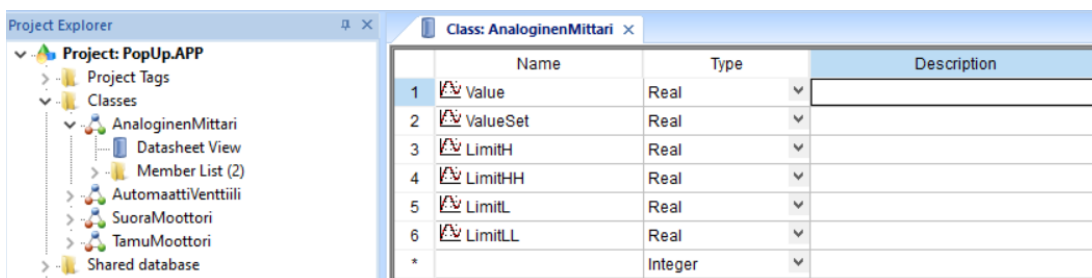
5.2 Ohjauslohkojen tagit

Luodaan ohjauslohkoille tagi ”luokat”, joihin lisätään moottoreille ja automaattiventtiilille integer-datatyypin komento ja tilatiedot, sekä taajuusmuuttajaohjatulle moottorille vielä lisäksi real-datatyypinen nopeudensäätö (Kuva 4).



Kuva 4. Tagi luokkien luonti.

Analogiselle mittalaitteelle lisätään real-tyyppin tagit mittauksen arvolle, syötettävälle halutulle arvolle ja mittauksen ala- ja ylärajat varoituksille ja hälytyksille (Kuva 5).



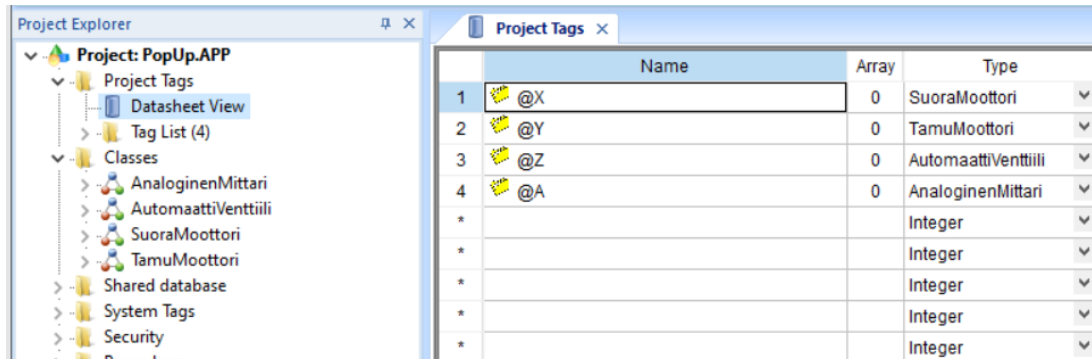
The screenshot shows the Project Explorer on the left with the 'Project: PopUp.APP' tree. Under 'Classes', 'AnaloginenMittari' is expanded, showing 'Datasheet View', 'Member List (2)', 'AutomaattiVenttiili', 'SuoraMoottori', and 'TamuMoottori'. The main window displays the 'Class: AnaloginenMittari' with a table of tags.

	Name	Type	Description
1	Value	Real	
2	ValueSet	Real	
3	LimitH	Real	
4	LimitHH	Real	
5	LimitL	Real	
6	LimitLL	Real	
*		Integer	

Kuva 5. Analogisen mittarin tagit.

Projektiin liitettäessä uusi laite, esimerkiksi automaattiventtiili, se voidaan lisätä luotuun ”AutomaattiVenttiili”-luokkaan, ja saa tätä kautta sille määritetyt tagit. Luokkia käyttämällä säästetään aikaa, kun jokaiselle yksittäiselle laitteelle ei tarvitse määrittää erikseen omia tageja.

Luodaan projektin tageihin yleinen tagi moottoreille, automaattiventtiilille ja mittalaitteelle, jotka liitetään aikaisemmin tehtyihin luokkiin, näin saadaan yksilöityä avattavan laitteen ikkuna, vaikka samassa luokassa olisi useampia laitteita. @-merkillä saadaan tehtyä yleinen tagi. Nimetään suoramoottorin tagiksi X, taajuusmuuttajaohjatun moottorin tagiksi Y, automaattiventtiilin tagiksi Z ja mittalaitteen tagiksi A (Kuva 6).



The screenshot shows the Project Explorer on the left with the 'Project: PopUp.APP' tree. Under 'Project Tags', 'Tag List (4)' is expanded, showing '@X', '@Y', '@Z', and '@A'. The main window displays the 'Project Tags' with a table of the tags.

	Name	Array	Type
1	@X	0	SuoraMoottori
2	@Y	0	TamuMoottori
3	@Z	0	AutomaattiVenttiili
4	@A	0	AnaloginenMittari
*			Integer
*			Integer
*			Integer
*			Integer
*			Integer

Kuva 6. Projektin yleiset tagit

5.3 Suorakäyttöinen moottori

Suorakäyttöisen moottorin ohjauslohkoon tarvitaan painikkeet moottorin tilan asettamiseen automaatti- tai käsiajolle, sekä käynnistys- ja pysäytyspainikkeet, joilla on mahdollisuus ohjata moottoria sen ollessa käsiajotilassa. Ohjauslohkoon tehdään myös erillinen painike ikkunan sulkemiseen. Tilatiedot otetaan automaatti- ja käsiajosta, sekä vika-, lukitus- ja turvakytkimen asennosta. Tilatiedot ja moottorin symboli korostetaan väreillä kertomaan laitteen tilasta.

5.4 Taajuusmuuttajaohjattu moottori

Taajuusmuuttajaohjatun moottorin ohjauslohko toteutetaan muuten vastaavasti kuin suorakäyttöisen moottorin, mutta ohjauslohkoon lisätään vielä mahdollisuus säätää moottorin pyörimisnopeutta. Moottorin pyörimisnopeus ilmoitetaan yleensä taajuuden yksiköllä, Hz, mutta lohossa käytetään pyörimisnopeudelle %-yksikköä, joka on helpommin tulkittavissa syöttämällä nopeus asteikolla 0-100.

5.5 Automaattiventtiili

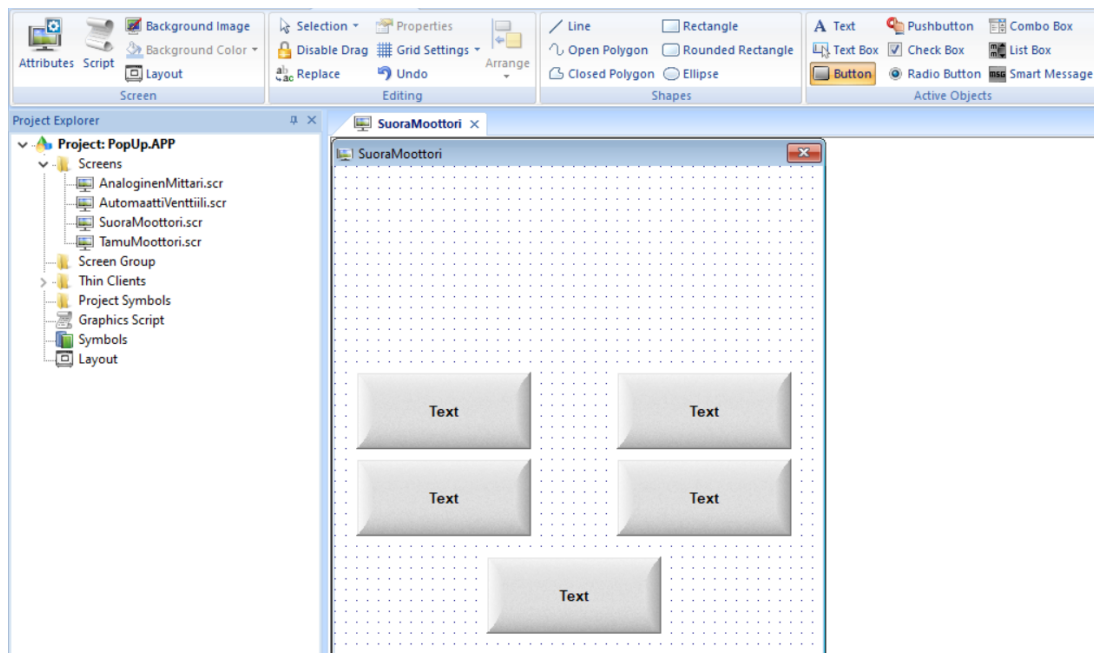
Automaattiventtiilin ohjauslohkoon otetaan tilatiedot venttiilin tilasta; lukitus-, vika-, käsiajo- ja automaattiajo tilatiedot. Ohjauspainikkeet tehdään automaattiajolle ja käsiajolle, sekä käsiajon painikkeet venttiilin avaamiseen ja sulkemiseen.

5.6 Analoginen mittalaite

Analogisen mittalaitteen ohjauslohkoon tehdään näkymä, josta käy ilmi mittalaitteen mittaama arvo. Ohjaukseen tehdään mahdollisuus syöttää haluttu mitta-arvo, eli esimerkiksi virtausmittarille voidaan asettaa pyydetty virtaus tai pintamittaukselle haluttu pinnankorkeus. Ohjauslohkosta on myös mahdollisuus määrittää mittaukselle hälytys- ja varoitusrajojen arvot.

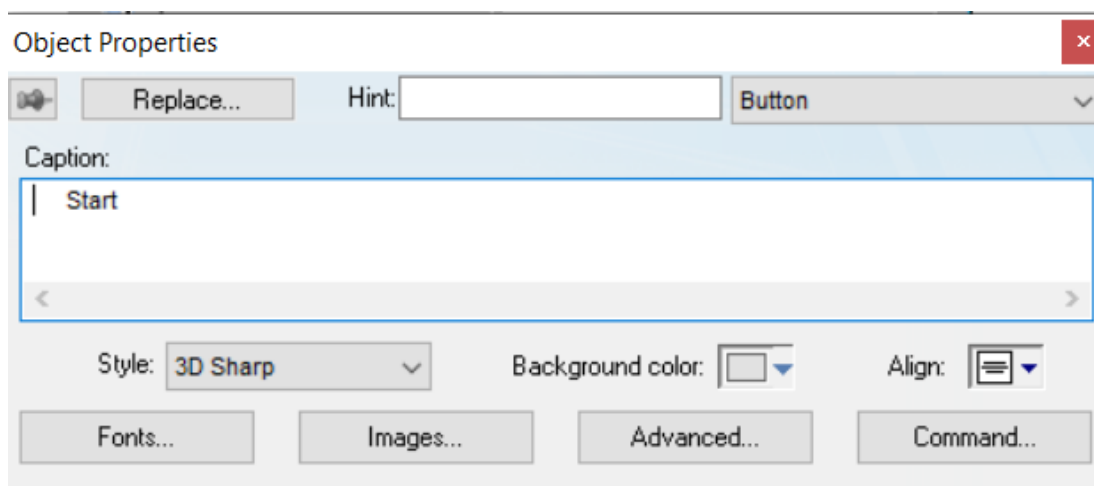
5.7 Ohjauslohkon komennot

Ohjauslohkon komentopainikkeet saadaan tehtyä ”Graphics”-välilehdeltä, valitsemalla Active Objects-valikosta ”Button” ja lisätään niitä tarvittava määrä lohkoihin. Esimerkiksi suorakäyttöisen moottorin ohjauslohkoon tarvitaan painikkeet käsi- ja automaattiajolle, käynnistykselle, pysäytykselle ja ohjauslohkon sulkemiseen (Kuva 7).



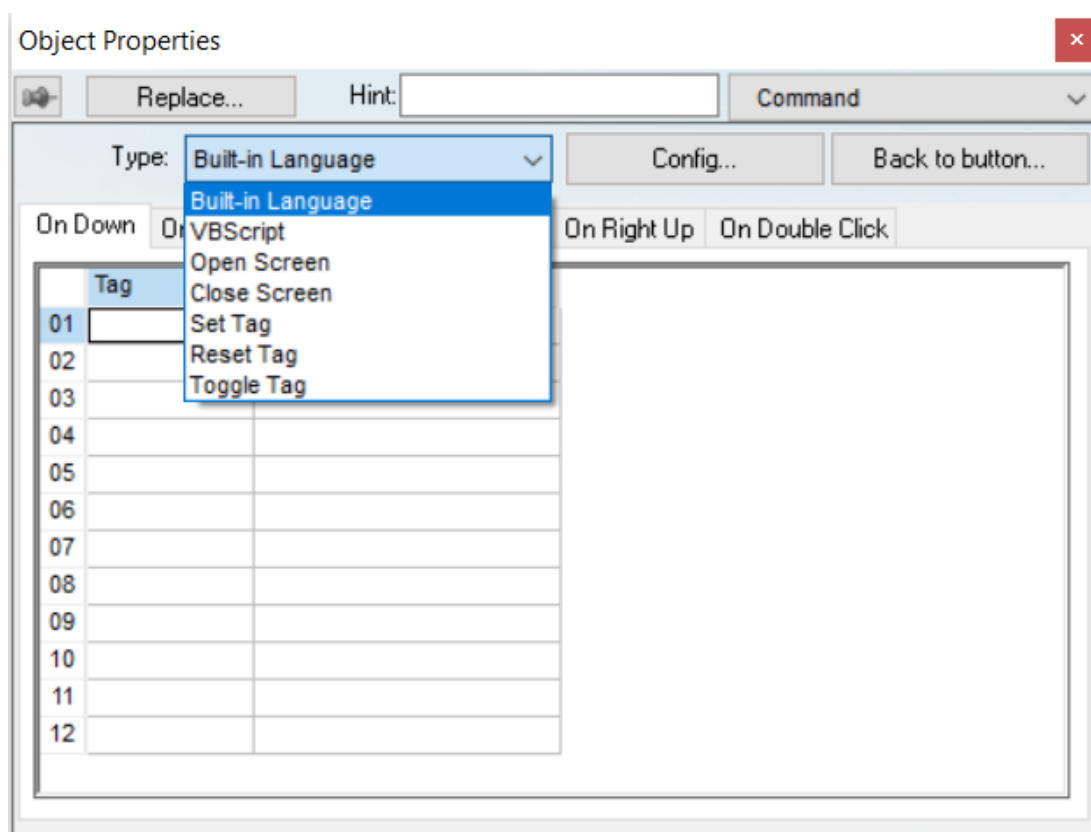
Kuva 7. Painikkeet komentojen antamiselle.

Painikkeen tiedot saadaan esille kaksoispainamalla painiketta, josta pääsee muokkaamaan painikkeessa olevaa tekstiä (Kuva 8).



Kuva 8. Komentopainikkeen tiedot.

Painikkeisiin saadaan lisättyä komennot ”Graphics”-välilehdeltä, valitsemalla Animations-valikosta ”Command”, ja sitten kaksoispainamalla haluttua painiketta. Haluttu komento määritetään avautuvaan ”Object Properties”-valikkoon (Kuva 9).



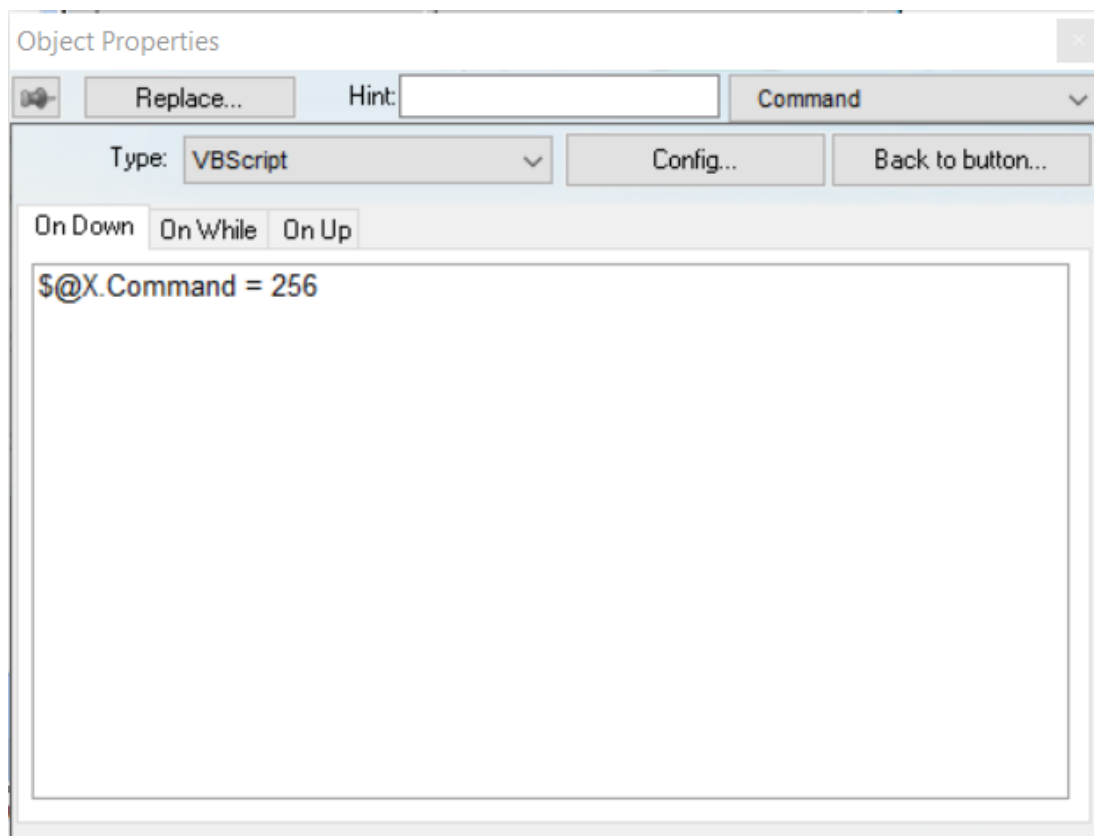
Kuva 9. Komennon luominen.

Painikkeen komento määritetään Type-valikosta halutulla tyyllillä. Built-in Language, jolla voi määrittää useampia scripteja. VBScript, johon voi kirjoittaa halutut tapahtumat VBScript-kielellä. Open ja Close Screen vaihtoehdoilla voidaan määrittää painalluksesta avautuva tai sulkeutuva ikkuna. Set Tag asetuksella painallus muuttaa ohjatun tagin arvoksi 1 ja Reset Tag asetuksella painallus muuttaa ohjatun tagin arvoksi 0. Toggle Tag muuttaa jokaisella painalluksen ohjatun tagin arvoa 0 ja 1 välillä.

Tässä työssä ohjauslohkojen komennot toteutetaan VBScript-kielellä, joilla ohjataan TIA Portaliin luotavia ”PLC data type”-luokkia. Datatyyppinä käytetään Wordia, ja komennot määritetään aloittamalla jälkimmäisestä tavusta, eli arvosta 256 seuraavasti: manuaaliajo ($2^8 = 256$), automaattiajo ($2^9 = 512$), käsiajon käynnistys ($2^{10} = 1024$), käsiajon pysäytys ($2^{11} = 2048$).

5.7.1 Käsiäjon painike

Nimetään käsiäjopainikkeen nimeksi ”Käsiäjo”, ja lisätään painikkeeseen komento. \$-symboli tekstin alkuun, jonka jälkeen viittaus laitteen yleiseen tagiin, esimerkiksi suo-raohjatun moottorin komennoissa ”@X.”, jonka jälkeen ”Command” tagi moottoreille luodusta luokasta (Kuva 10).

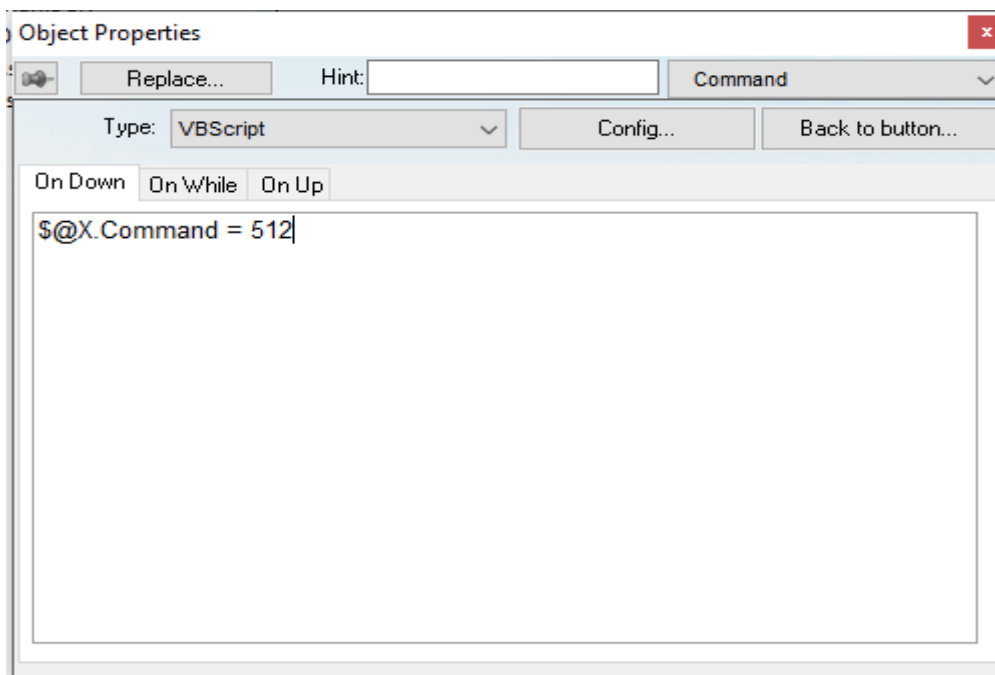


Kuva 10. Käsiäjo painikkeen komento.

Painikkeet saadaan luotua samalla tavalla taajuusmuuttajaohjatulle moottorille ja automaattiventtiilille, vaihtamalla vain yleisen tagin oikeaksi. Nyt painiketta painamalla wordin 8. bitti muuttuu arvosta 0 arvoon 1.

5.7.2 Automaattiajon painike

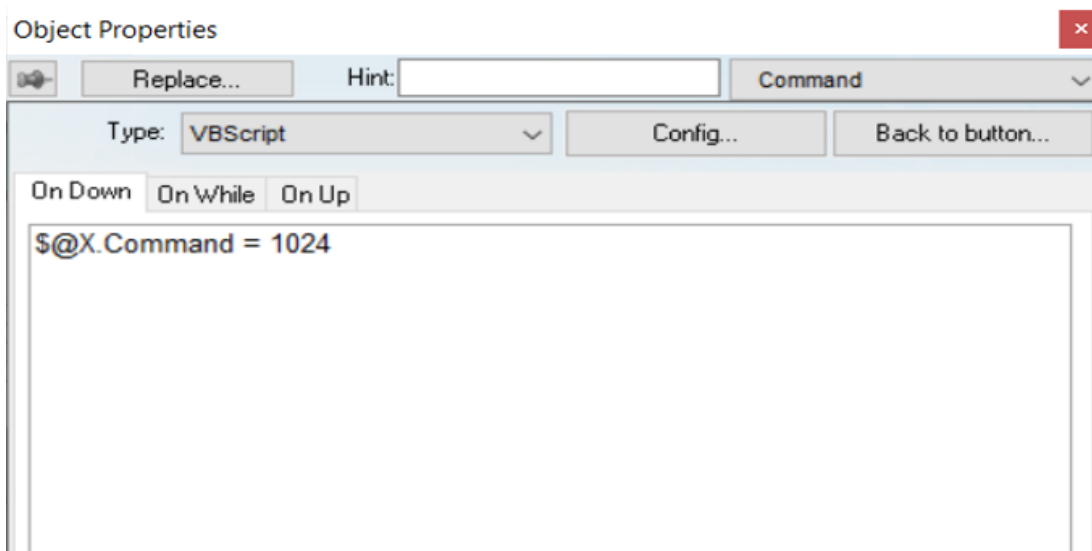
Nimetään automaattiajon painikkeen nimeksi ”Auto”, ja lisätään painikkeeseen komento edellisen komennon mukaisesti, vaihtamalla ohjattava bitti määritetyn mukaisesti (Kuva 11).



Kuva 11. Automaattiajon painikkeen komento.

5.7.3 Moottorin käynnistys- ja venttiilin avauspainike

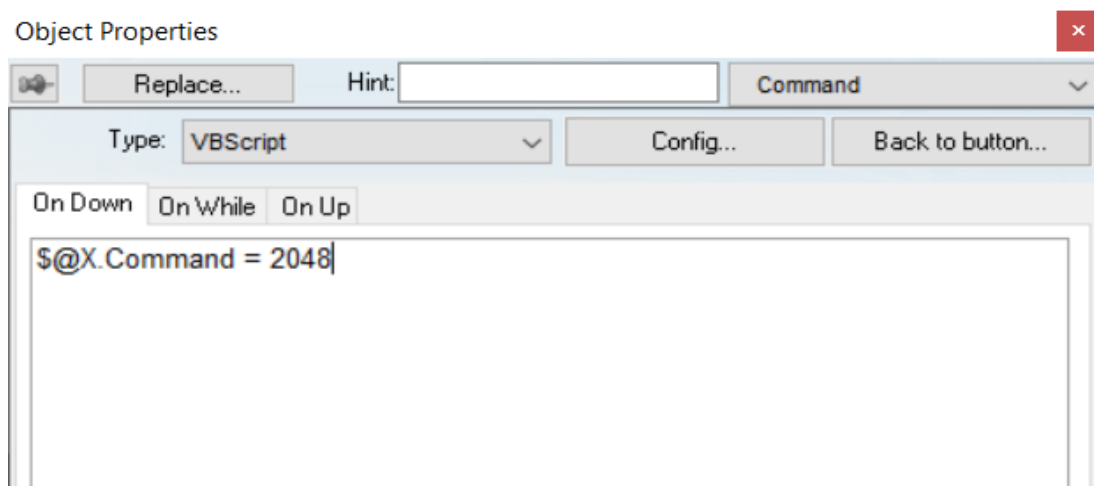
Nimetään painikkeen nimeksi moottorien ohjaukseen ”Start” ja venttiilin ohjaukseen ”Open”, ja lisätään painikkeeseen komento ohjaamaan seuraavaa bittiä (Kuva 12).



Kuva 12. Käynnistys- ja avauspainikkeen komento.

5.7.4 Moottorin pysäytys- ja venttiilin sulkemispainike

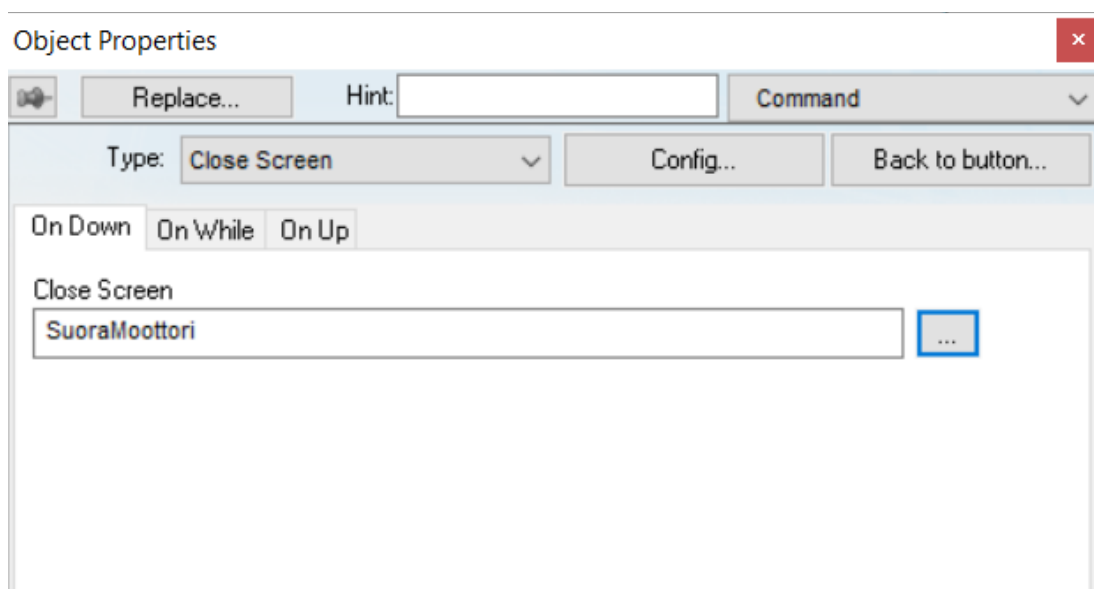
Nimetään painikkeen nimeksi moottorin ohjaukseen ”Stop” ja venttiilin ohjaukseen ”Close”, ja lisätään painikkeeseen komento ohjaamaan seuraavaa bittiä (Kuva 13).



Kuva 13. Pysäytys- ja sulkemispainikkeen komento.

5.7.5 Ikkunan sulkemispainike

Nimetään sulkemispainikkeen nimeksi ”Exit”. Lisätään painikkeeseen ”Close Screen”-tyypin komento, ja linkitetään siihen ikkuna, jossa painike sijaitsee ja halutaan sulkeutuvan (Kuva 14).

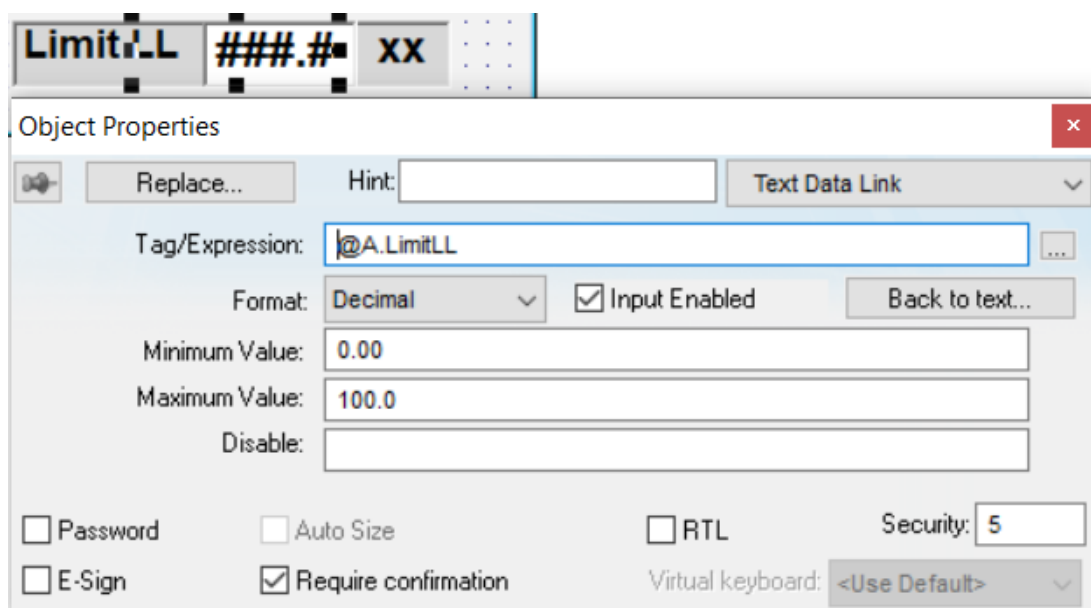


Kuva 14. Komento ikkunan sulkemiseen.

5.7.6 Text data linkitykset

Real-datatyypin muotoilu tehdään IWS ohjelmalla #-merkeillä. Lisätään tarvittava määrä #-merkkejä niin, että yksi merkki vastaa yhtä luettavaa numeroa, eli esimerkiksi jos luettava arvo on välillä 0-100, tarvitaan merkinnäksi ”###”. Desimaaliluvut saadaan luettua pisteellä erotettuna, muotoa ”##.#”.

Tehdään yllä mainitulla tavalla taajuusmuuttajaohjatun moottorin nopeudensäätö ja analogisen mittarin mittaukset ”Active Objects”-valikosta löytyvällä tekstityökalulla ”Text”, jolla saadaan kirjoitettua #-merkkejä haluttu määrä. Lisätään merkit muodossa ”###.#”, jolla saadaan katettua suurin osa mittauksista, ilman erillistä tekstin muok-kausta. Näytettäväksi mittausväliksi tulee näin 0.0 – 999.9. Valitsemalla hiiren painal-luksella merkit, ”Animations”-valikon ”Text Data Link”-painike tulee aktiiviseksi. Text Data Linkillä saadaan lisättyä luettava tagi ja määritettyä sen ominaisuuksia (Kuva 15).

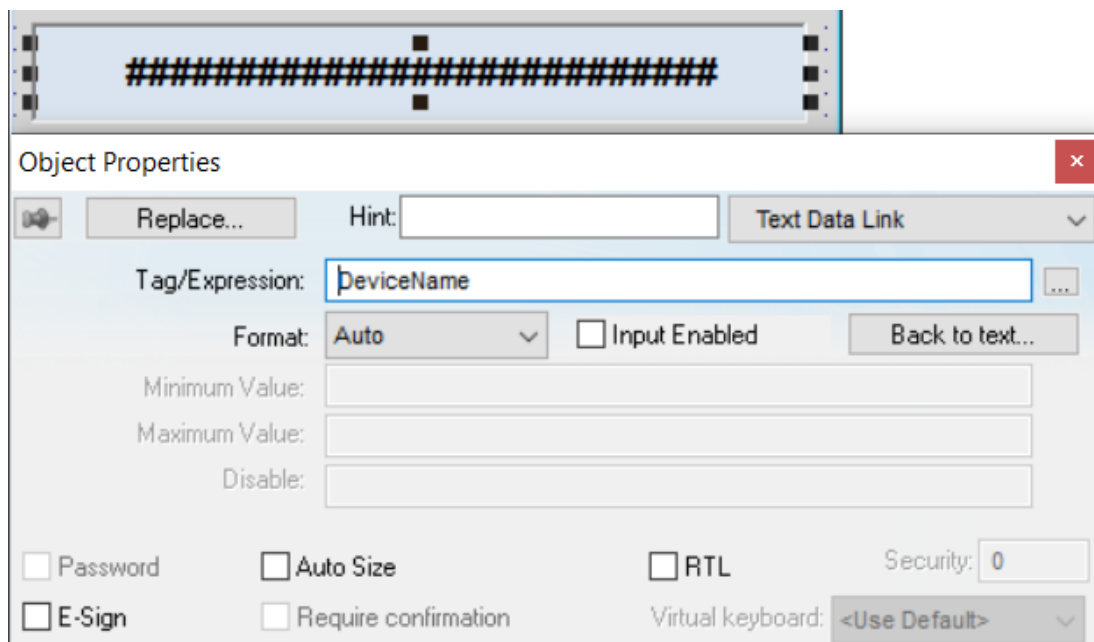


Kuva 15. Teksti datan linkitys.

Kuvassa 15 ”Tag/Expression” kohtaan on määritetty analogisen mittauksen alarajan hälytys, linkittämällä analogisen mittarin luokkaan (@A) määritetty sille vastaava tagi, muodossa luokka.tagi = @A.LimitLL. Format -valikosta saadaan määritettyä tekstinmuoto (Decimal, hexadecimal, binary tai auto), valikon vieressä on valintaruutu, ”Input Enabled”, jonka aktivoiminen mahdollistaa arvon syöttämisen. Syötettävää arvoa voidaan rajata minimi ja maksimi arvoilla, jolloin ohjelma ei hyväksy asteikon

ulkupuolelta syötettäviä arvoja, esimerkiksi kuvan 15 asetuksilla ohjelma hyväksyy syötettävät arvot välillä 0-100. Disable sarakkeeseen voidaan määrittää tagi, jonka ollessa aktiivisena ei kyseistä arvoa pysty muokkaamaan. Arvon muokkaaminen on myös mahdollista suojata salasanalla tai rajoittaa oikeudet vain tietyille ohjelman käyttäjille. Valitsemalla ”Require confirmation” valintaruudun aktiiviseksi, ohjelma kysyy arvoa muokattaessa vielä vahvistuksen ennen arvon muuttamista.

Lisätään jokaiselle ohjauslohkolle otsikoksi tarvittava määrä #-merkkejä, ja lisätään projektin ”Global”-valikosta projektin tageihin ”DeviceName” niminen string data-tyyppinen tagi. Text data linkkiä käyttäen liitetään se jokaisen ohjauslohkon otsikon #-merkkeihin (Kuva 16).

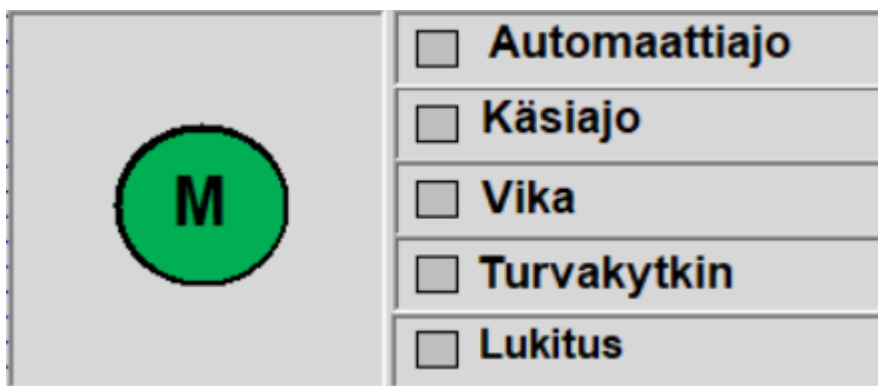


Kuva 16. Ohjauslohkon otsikon linkitys.

5.8 Ohjauslohkon tilatiedot

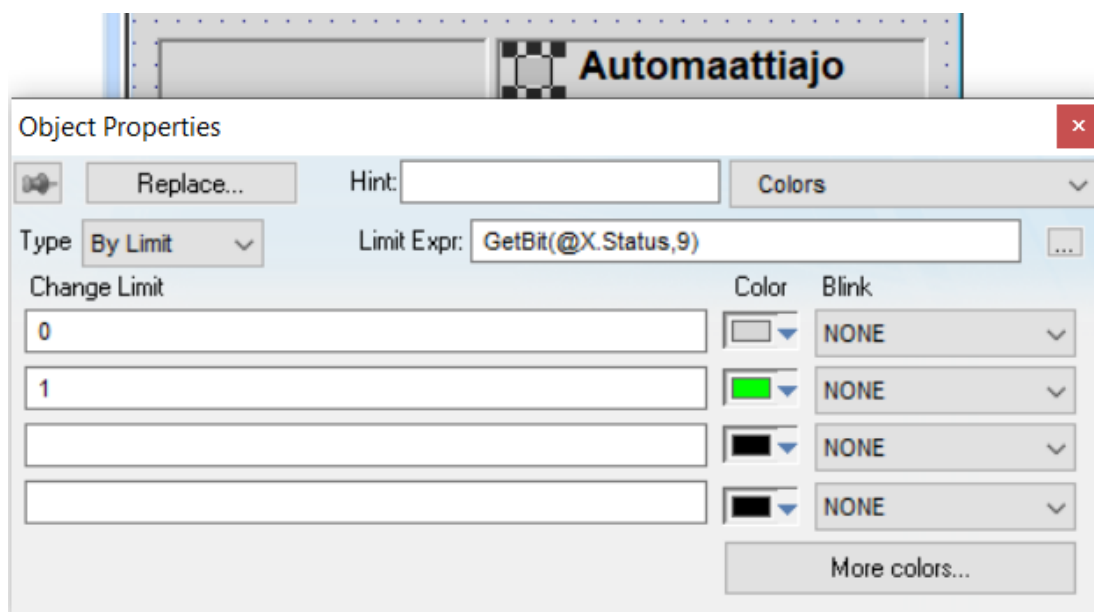
Tilatiedot IWS ohjelmaan saadaan luettua GetBit-komennolla. Tilatiedot korostetaan väreillä niin, että ne ovat mahdollisimman selkeästi tulkittavissa. Automaatti- tai käsiajon ollessa aktiivisena, korostetaan väri vihreällä. Vika-, lukitus- ja turvakytkimen tilatiedot korostetaan aktiivisena punaisella. Ei aktiivisessa tilassa pohjavärinä on harmaa, paitsi turvakytkimen tila korostetaan keltaisella. Moottorin käyntitieto, sekä venttiilin auki tieto näytetään vihreällä värillä, erikseen piirrettyssä symbolissa.

Tilatiedoille piirrettiin omat laatikot, josta käy ilmi tekstimuodossa mitä sillä näytetään, sekä oma neliö värитеhostetta varten (Kuva 17).



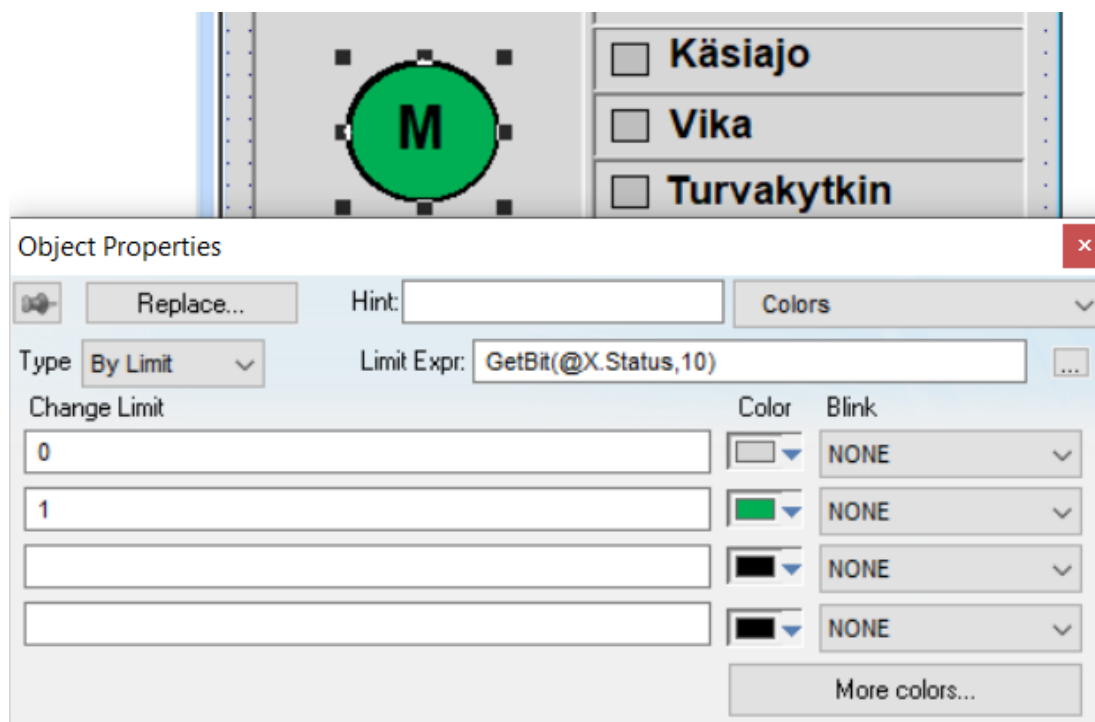
Kuva 17. Näkymä ohjauslohkon tilatiedoista.

Värитеhosteet saadaan lisättyä valitsemalla Animations-välilehdeltä Color, jonka jälkeen painamalla kohdetta johon tehosteet halutaan lisätä. Linkitykset tehdään kaavalla `GetBit(@X.Status,y)`, jossa "@X." kuvaa aikaisemmin määritettyä vakiota, esimerkiksi suoraohjatusta moottorista, "Status" teksti vakioon määritetyn tagin mukaisesti, jonka jälkeen pilkulla erotettuna y, eli bitin numero, jota halutaan lukea. Automaattiajon komentoon määritettiin 512 (2^9), eli automaattiajon tilatietoa luettaessa perään laitetaan bitin numero 9. Bitin arvon ollessa 0 väriksi laitetaan harmaa, ja bitin arvon ollessa 1 väriksi laitetaan vihreä (Kuva 18).



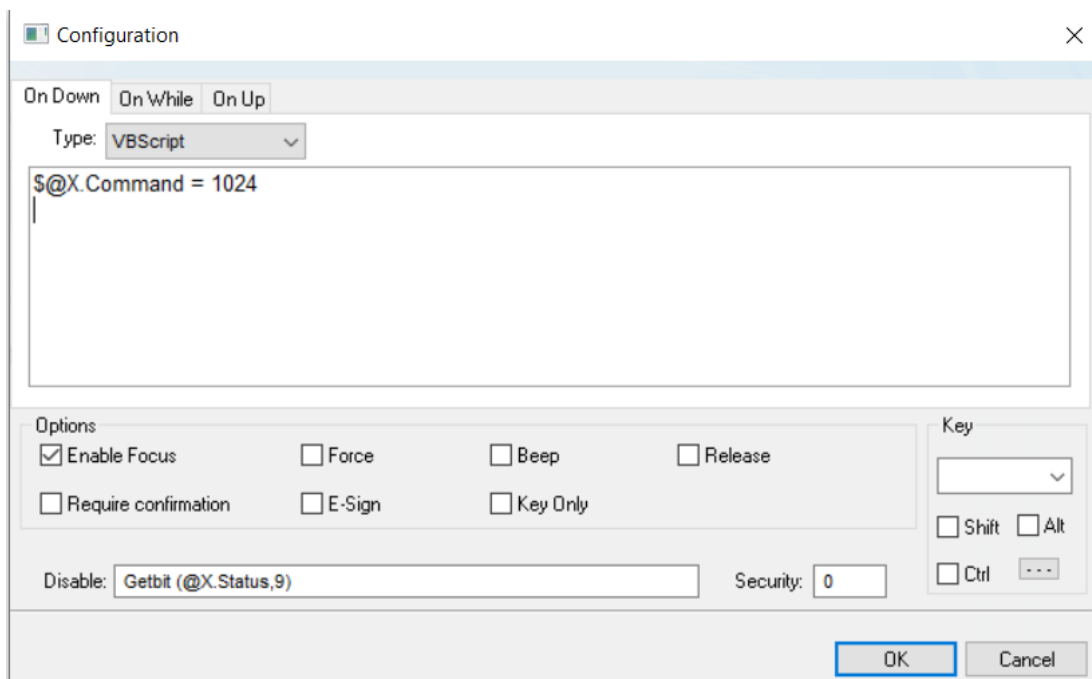
Kuva 18. Automaattiajon tilatiedon lukeminen.

Käsiajon tilatieto otetaan vastaavasti bitistä 8. Turvakytkimen, vian- ja lukituksen tilatiedot otetaan vapaista biteistä, 13,14 ja 15, joita voidaan ohjata logiikassa, mutta tilatiedot saadaan silti näkymään käyttöliittymässä. Piirretyille moottorin ja venttiilin symboleille tilatieto otetaan käynti- ja auki tilatiedoista, bitti numerosta 10 (Kuva 19).



Kuva 19. Moottorin käyntitiedon linkitys.

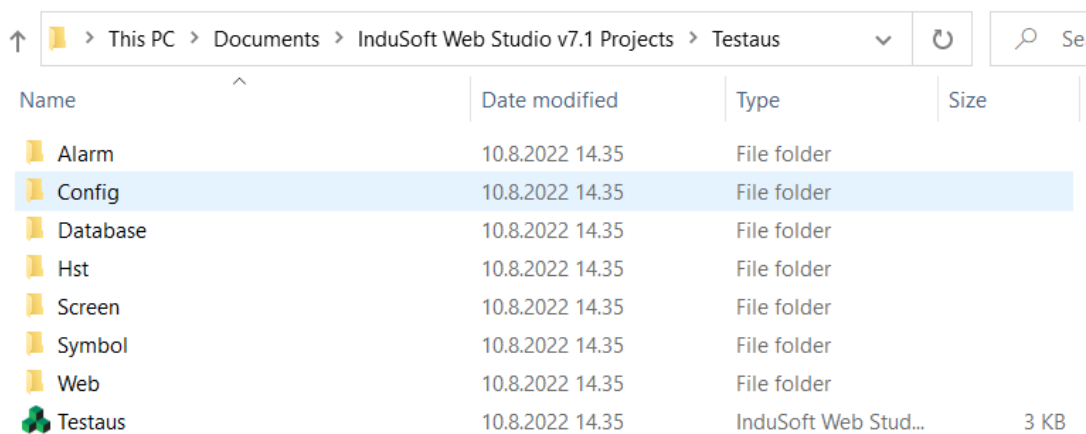
Tilatiedoilla voidaan myös estää moottorin käynnistys- ja pysäytyspainikkeiden sekä venttiilin avaus- ja sulkemispainikkeiden käyttö laitteen ollessa automaattiajolla. Avataan painikkeen Command-valikosta ”Config”, ja määritetään Disable-sarakkeeseen automaattiajon bitti. Näin saadaan estettyä käsiajo, kun laite on automaattiajolla (Kuva 20).



Kuva 20. Käsiäjon estäminen automaattitilassa.

6 KÄYTTÖOHJEET

Käyttöohjeet sisältävät opastuksen valmiiksi luotujen ohjauslohkojen käyttöönottamiseen uuteen projektiin. Luodaan uusi projekti IWS ohjelman päävalikosta, kohdasta ”new”. Annetaan projektille nimi, ja määritetään projektin tallennuskansio, sekä näytön koko pikseleinä. Kun uusi projekti on luotu, avataan projektinkansio paikasta, johon se luontivaiheessa määritettiin tallennettavaksi (Kuva 21).



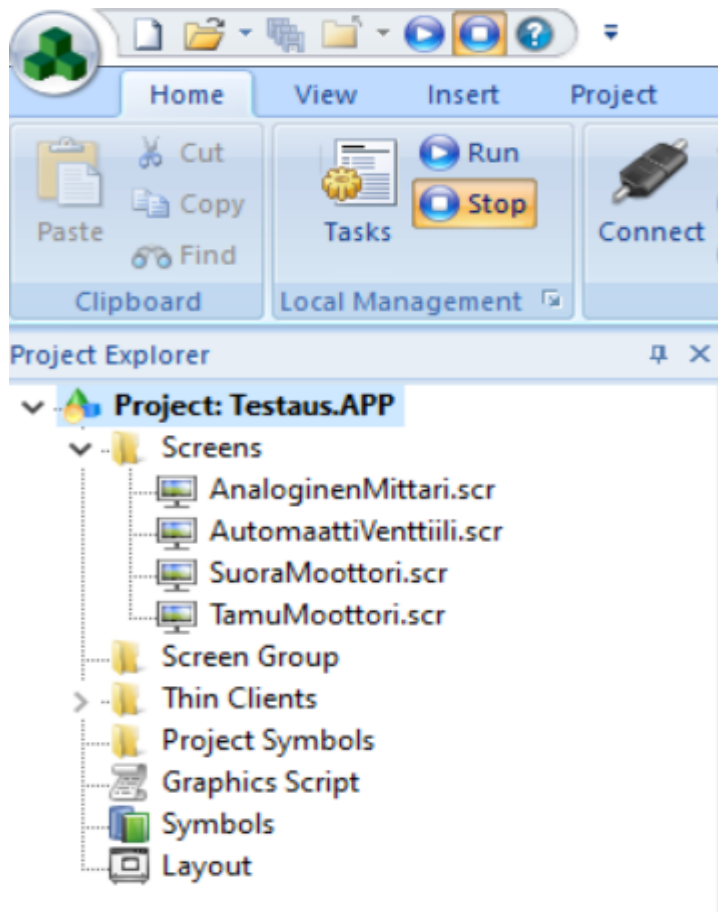
Kuva 21. Projektin kansio.

Tehdyt ohjauslohkot saadaan tuotua uuteen projektiin kopioimalla ne erikseen tallennetusta kansioista. Avataan uuden projektin kansioista, kansio nimeltä ”Screen”, ja liitetään kopioidut ohjauslohkot sinne (Kuva 22).



Kuva 22. Ohjauslohkojen lisääminen uuteen projektiin.

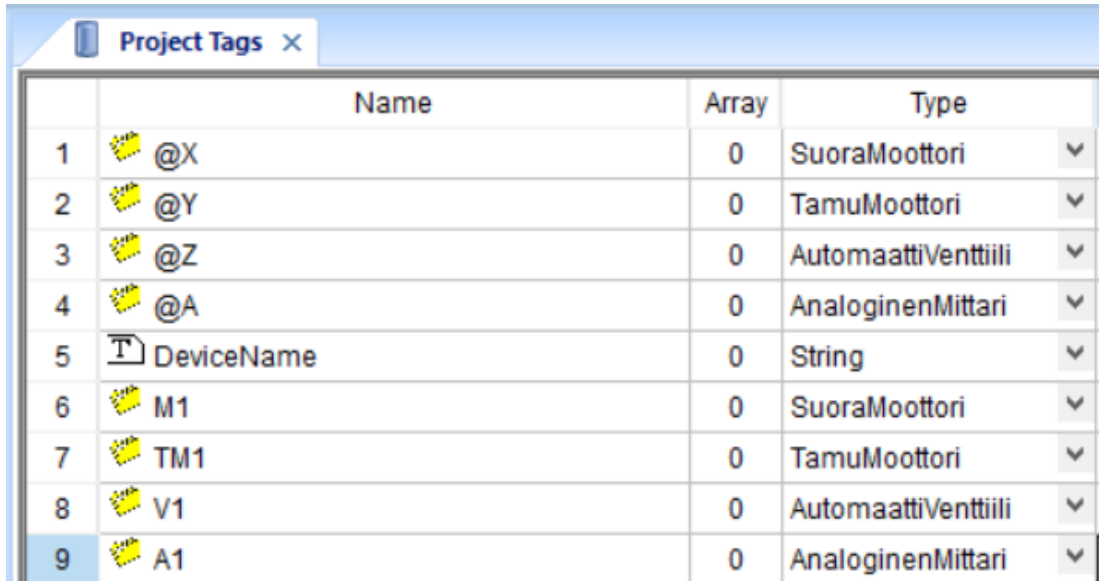
Tämän jälkeen ohjauslohkot tulevat näkyville projektin Screens-valikon alle, ja ovat käytettävissä uudessa projektissa (Kuva 23).



Kuva 23. Ohjauslohkot tuotu uuteen projektiin.

Ohjauslohkon toiminnan kuvaamista varten luodaan uusi ikkuna nimeltä ”Main”, joka tavallisesti voisi olla projektin päänäkymä. Lisätään päänäkymään symbolit, kuvaamaan suoraikäyttöistä moottoria, taajuusmuuttajaohjattua moottoria, automaattiventtiiliä ja analogista mittalaitetta. Lisätyt symbolit, joihin ohjauslohkojen linkitykset tehdään, voidaan tehdä joko piirtämällä tai käyttää IWS ohjelman kirjastosta löytyviä valmiita symboleja, ”Libraries”-valikon kohdasta ”Symbols”. Tässä esimerkki projektissa käytetään suoraikäyttöisen moottorin nimenä M1, taajuusmuuttajaohjatun moottorin nimenä TM1, automaattiventtiilin nimenä V1 ja analogisen mittalaitteen nimenä A1.

Lisätään projektiin tagiluokat ja yleiset tagit jokaiselle ohjauslohkolle, kohdan 4.2 mukaisesti. Lisätään projektin tageihin ”DeviceName”-tagi, sekä äsken päänäkymään lisätyt M1, TM1, V1 ja A1, ja valitaan ”type”-valikosta niille kuuluva luokka (Kuva 24).

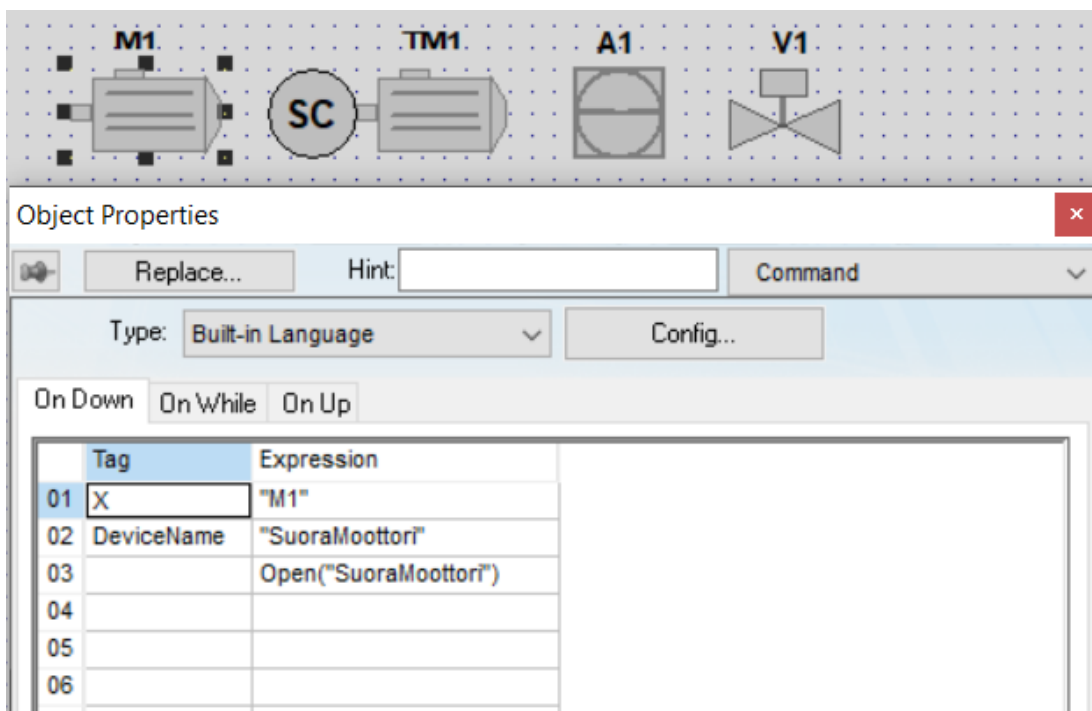


	Name	Array	Type
1	@X	0	SuoraMoottori
2	@Y	0	TamuMoottori
3	@Z	0	AutomaattiVentiili
4	@A	0	AnaloginenMittari
5	DeviceName	0	String
6	M1	0	SuoraMoottori
7	TM1	0	TamuMoottori
8	V1	0	AutomaattiVentiili
9	A1	0	AnaloginenMittari

Kuva 24. Projektin tagit.

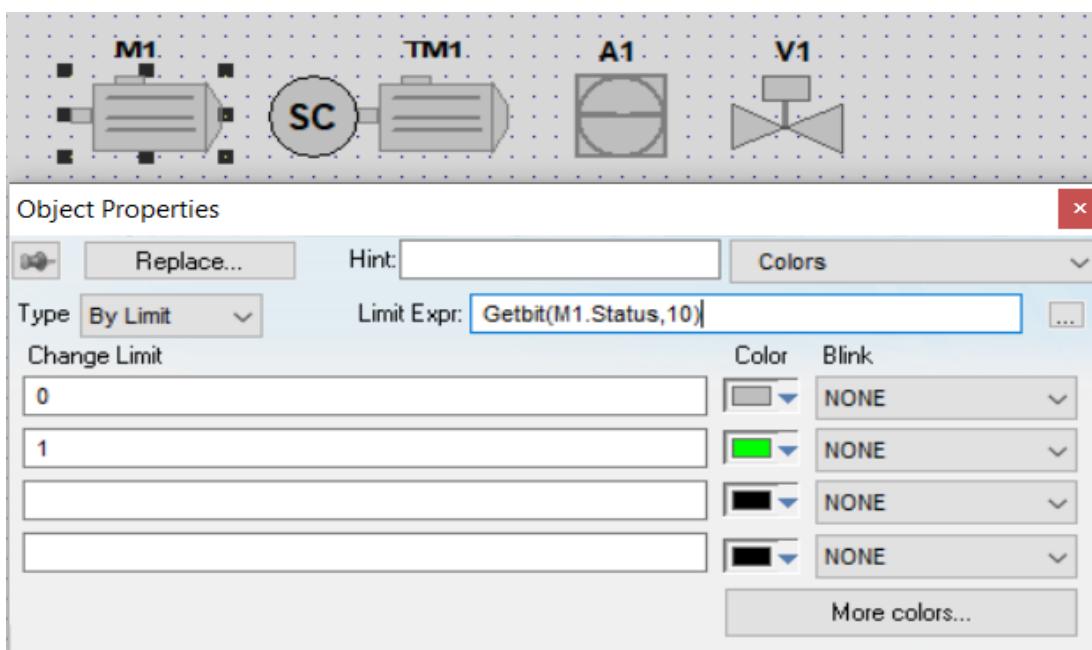
6.1 Ohjausikkunan avaaminen symbolista

Lisätään komento päänäkymään lisätyille symboleille. Komento tehdään ”Built-in Language”-tyylillä, jossa vasempaan sarakkeeseen määritetään tagit, ja oikeanpuoleiseen sarakkeeseen tapahtuma. Ensimmäiselle riville määritetään avattavan ohjauslohon yleinen tagi ilman @-merkkiä (X,Y,Z tai A), ja tapahtumapuolelle ”-merkkien väliin kirjoitettuna yksilöidään laite, jonka ohjausikkuna halutaan avata, eli esimerkiksi suoraohjatun moottorin avauksessa käytetään merkintänä ”M1”. Merkinässä käytettävä tagi pitää olla määritetty projektin tageihin oikeaan luokkaan, jotta ohjelma osaa hakea sille oikeat tiedot. Seuraavalle riville määritetään ”DeviceName”-tagi, jonka arvo määritetään tapahtumapuolelle ”-merkkien väliin, tyylillä ”SuoraMoottori”. Kun tarvittavat tiedot on määritetty, lisätään viimeiseksi riviksi tapahtumapuolelle komento, avaamaan haluttu ohjausikkuna, käyttäen ”-merkkien sisällä nimeä, joka ikkunalle on määritetty, muotoa ”Open(”SuoraMoottori”), kuvan 25 mukaisesti.



Kuva 25. Ohjausikkunan avaaminen symbolista.

Symboleihin voidaan lisätä myös väritehosteita kertomaan laitteen tilatiedoista. Lisätään moottoreihin käyntitieto ja venttiiliin auki-asennon tilatieto korostumaan vihreällä värillä. Color-tehostetta käyttämällä saadaan määritettyä symboleihin luettavan laitteen tilatieto Getbit-komennolla, muotoa "Getbit(laite.Status,luettava bitti)" (Kuva 26).



Kuva 26. Symbolin tilatiedon korostaminen värillä.

6.2 Yhdistäminen logiikkaan

IWS ohjelman ohjauslohkot saadaan kommunikoidaan automaatio-ohjelman kanssa, tässä tapauksessa Siemensin Tia Portalin, SITIA:n ajurin avulla. Tagit määritetään projektin ”Comm”-valikon Drivers kansioista, avaamalla SITIA:n tagi taulukon. ”Tag Name”-sarakkeeseen syötetään IWS ohjelmassa käytettävä tagi, ”Station”-sarakkeeseen syötetään IP-osoite ja ”I/O Address”-sarakkeeseen syötetään linkitettävä tagi logiikasta (Kuva 27). Moottoreilta ja automaattiventtiileiltä linkitykseen tarvitaan tila-tieto (laite.Status) ja komentotieto (laite.Command), sekä taajuusmuuttajaohjatulta moottorilta vielä lisäksi nopeus (laite.Speed). Analoginen mittalaite tarvitsee tagin jokaiselle käytössä olevalle parametrille.

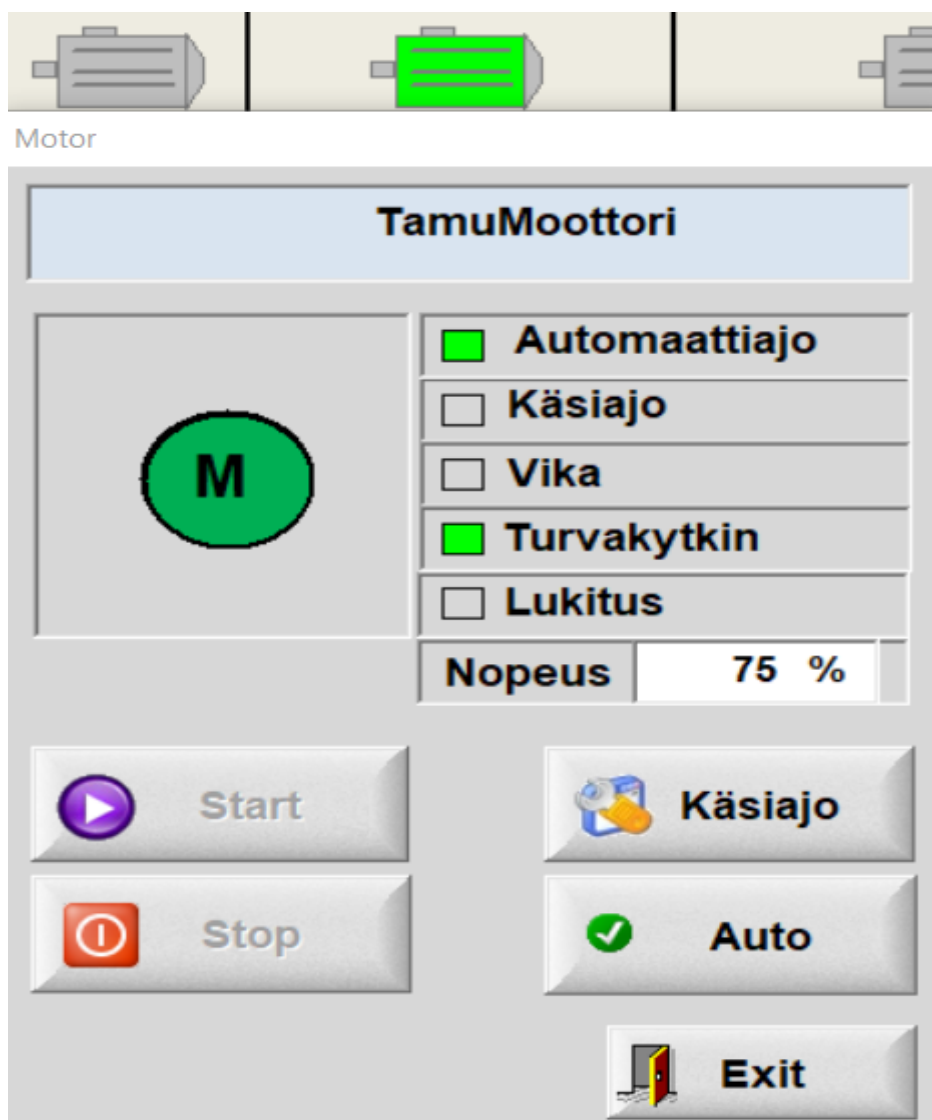
	Tag Name	Station	I/O Address	Action	Scan
286	M1.Status	192.168.0.1	dbPLC>HMI.M1_Status	Read+Write	Always
287	M1.Command	192.168.0.1	dbHMI>PLC.M1_Command	Read+Write	Always
288	TM1.Status	192.168.0.1	dbPLC>HMI.TM1_Status	Read+Write	Always
289	TM1.Command	192.168.0.1	dbHMI>PLC.TM1_Command	Read+Write	Always
290	TM1.Speed	192.168.0.1	dbHMI>PLC.TM1sp	Read+Write	Always
291	V1.Status	192.168.0.1	dbPLC>HMI.V1_Status	Read+Write	Always
292	V1.Command	192.168.0.1	dbHMI>PLC.V1_Command	Read+Write	Always
293	A1.Value	192.168.0.1	dbHMI>PLC.A1value	Read+Write	Always
294	A1.ValueSet	192.168.0.1	dbHMI>PLC.A1set	Read+Write	Always
295	A1.LimitH	192.168.0.1	dbHMI>PLC.A1_H	Read+Write	Always
296	A1.LimitHH	192.168.0.1	dbHMI>PLC.A1_HH	Read+Write	Always
297	A1.LimitL	192.168.0.1	dbHMI>PLC.A1_L	Read+Write	Always
298	A1.LimitLL	192.168.0.1	dbHMI>PLC.A1_LL	Read+Write	Always

Kuva 27. Malli IWS tagien yhdistämisestä logiikkaan.

7 YHTEENVETO

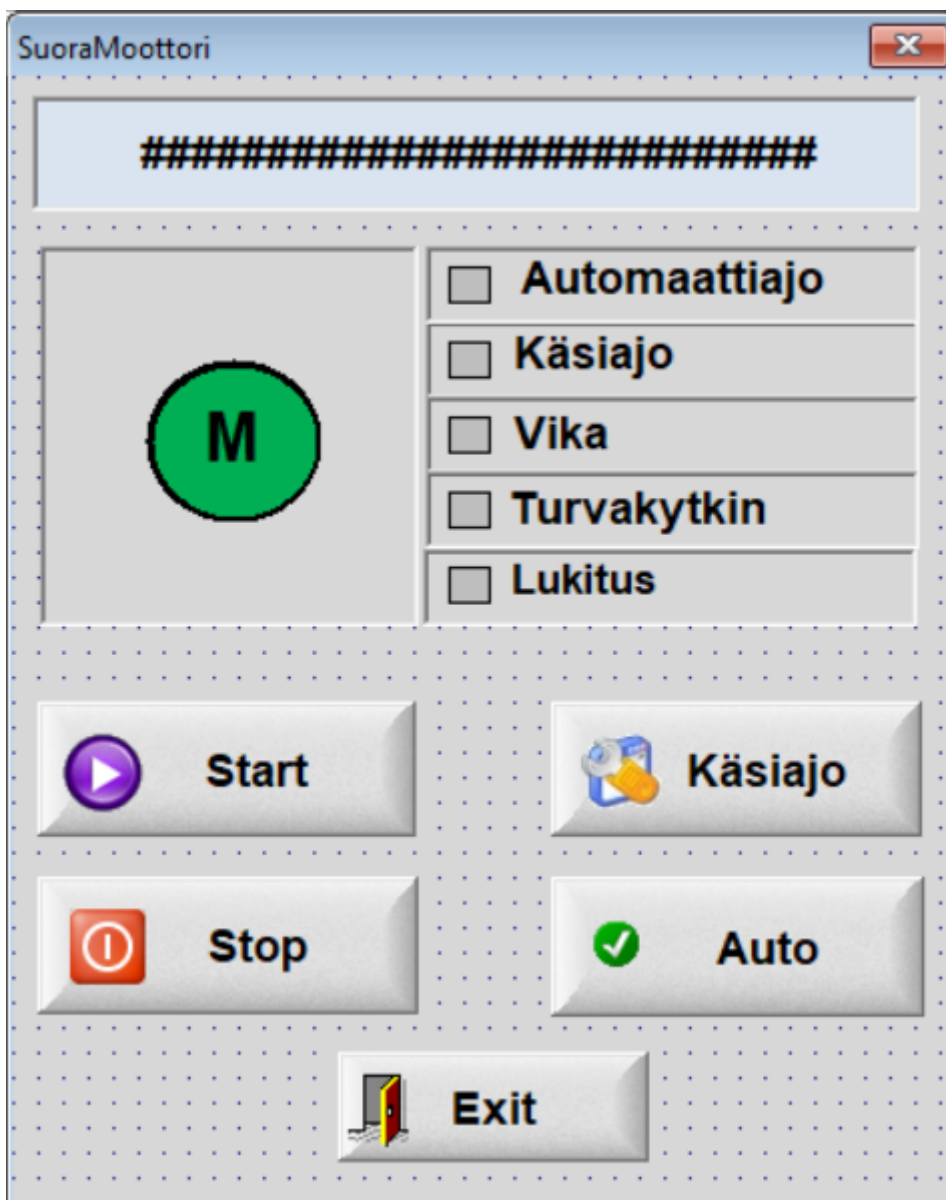
IWS ohjelmaympäristö oli minulle täysin tuntematon ennen opinnäytetyön aloittamista. Työn avulla opin käyttämään ohjelmistoa, sekä pääsin perehtymään sen toimintaan yksityiskohtaisemmin. Ohjauslohkojen tekeminen antoi minulle hyvät edellytykset toimia IWS ohjelman parissa, sekä osaamisen soveltaa työssä käytettyjä menetelmiä myös jatkossa.

Pääsin myös testaamaan IWS ohjelmalla luotuja ohjauslohkoja Siemensin Tia Portal automaatio-ohjelman kanssa. Ohjauslohkojen toiminta saatiin testattua ja todettua toimivaksi. Kuvassa 28 taajuusmuuttajaohjatun moottorin ohjauslohko käytönaikana yhdistettynä Tia Portaliin.



Kuva 28. Taajuusmuuttajaohjatun moottorin ohjauslohkon näkymä.

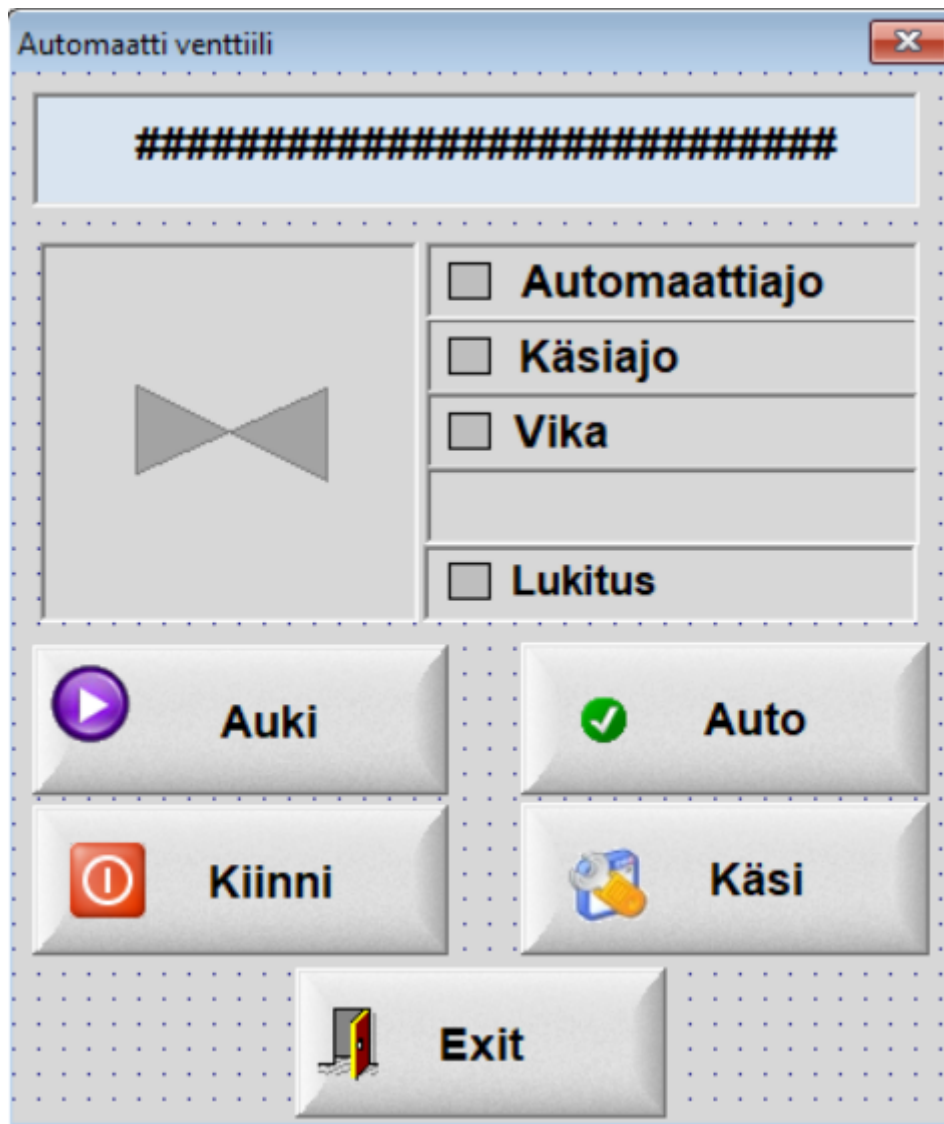
Työn lopputuloksena syntyi valmiit ohjauslohkot suorakäyttöiselle moottorille (Kuva 29), taajuusmuuttajaohjatulle moottorille (Kuva 30), automaattiventtiilille (Kuva 31) ja analogiselle mittalaitteelle (Kuva 32).



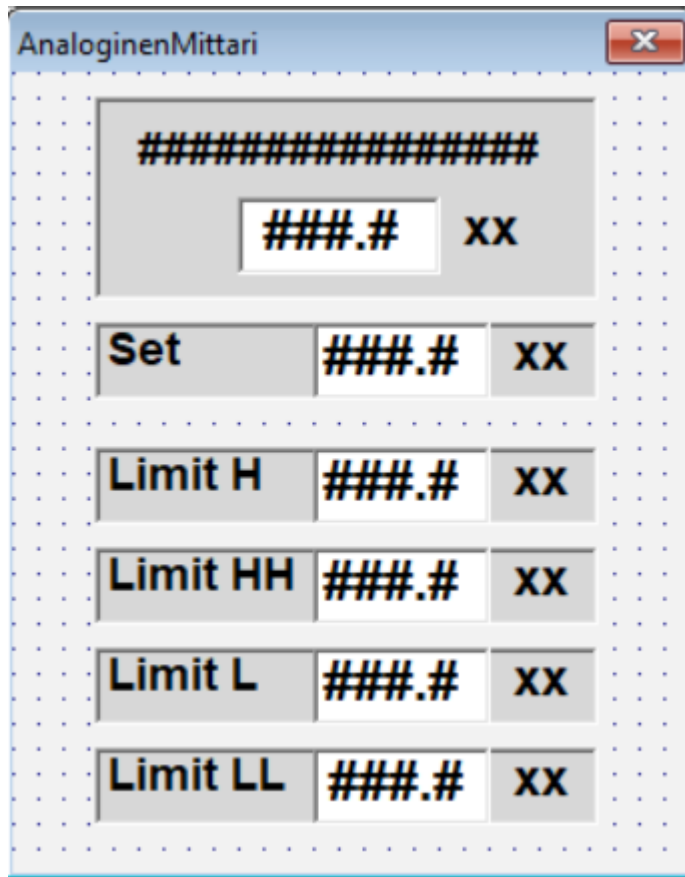
Kuva 29. Suorakäyttöisen moottorin valmis ohjauslohko.



Kuva 30. Taajuusmuuttajaohjatun moottorin valmis ohjauslohko.



Kuva 31. Automaattisen venttiilin valmis ohjauslohko.



Kuva 32. Analogisen mittarin valmis ohjauslohko.

LÄHTEET

AFRY Oy www-sivut 2022. Tietoa meistä. Viitattu 28.05.2022

<https://afry.com/fi-fi/tietoa-meista>

[http://download.indusoft.com/81.2.0/IWS+v8.1%2BSP2+Technical+Reference+\(LT\).pdf](http://download.indusoft.com/81.2.0/IWS+v8.1%2BSP2+Technical+Reference+(LT).pdf)