



# **Simatic S7 Tia Portal**

## **Erikoisominaisuudet**

Eino Mäkkylä

Opinnäytetyö  
Toukokuu 2013  
Kone- ja tuotantotekniikka  
Kone- ja laiteautomaatio

## TIIVISTELMÄ

Tampereen ammattikorkeakoulu  
Kone- ja tuotantotekniikka  
Kone- ja laiteautomaatio

EINO MÄKKYLÄ  
Simatic S7 Tia Portal  
Erikoisominaisuudet

Opinnäytetyö 35 sivua  
Marraskuu 2013

---

Tämän opinnäytetyön tarkoituksena oli perehtyä Siemens S7 TIA Portal -ohjelmiston uusiin erikoisominaisuuksiin ja tehdä sen perusteelta ohjekirja opetuksen tueksi Tampereen ammattikorkeakoululle. Työ on osana laajempaa käyttöohjekirjastoa uudesta ohjelmistosta.

Opinnäytetyö sisältää ohjeistuksen tärkeisiin aihealueisiin PLC -ohjelmoinnissa, joihin on tullut merkittäviä muutoksia vanhasta Siemens STEP-7 versiosta. Opinnäytetyössä käsitellään muuttujien tallentamista ja eri muuttujien ominaisuuksia, ohjelmarakenteita, tiedostoyksiköitä, organisaatioyksiköitä sekä ohjelman seuranta ja siellä olevien virheiden etsimistä. Jokaisesta aihealueesta on koitettu antaa selkeä ohjeistus kuvien ja tekstin avulla. Oletuksena oli, että lukija tietää ennestään perusasiat PLC -ohjelmoinnista, tai perehtyy käyttöohjekirjaston edellisiin osiin ensin, joissa ne osa-alueet käsitellään.

Opinnäytetyön tekeminen aloitettiin asentamalla ohjelmisto omalle koneelle ja perehtymällä siihen. Seuraavaksi perehdyttiin työssä käsiteltäviin aihealueisiin tarkemmin. Haasteita työn tekemiseen asettaa lähdeaineiston puutteellisuus, joten työssä käytettävät menetelmät ovat lähinnä itse perehtymistä ja siitä aineiston kirjoittaminen, sekä kuvakaappauksen ottaminen.

Lopputuloksena oli eri aihealueita käsittelevä ohjeistus, joihin aiheesta riippuen on perehdytty vähemmän tai enemmän syvällisesti. Ohjeistuksen lopullinen hyöty selviää vasta opetuskäytössä, joten sitä on mahdotonta sanoa etukäteen.

---

## **ABSTRACT**

Tampereen ammattikorkeakoulu  
Tampere University of Applied Sciences  
Degree Programme in Mechanical and Production Engineering  
Option of Machine Automation

EINO MÄKKYLÄ:  
Simatic S7 Tia Portal  
Erikoisominaisuudet

Bachelor's thesis 35 pages  
November 2013

---

The intention of this thesis was to get acquainted with Siemens S7 TIA Portal -software`s new special features and make operating instructions for educational use in the Tampere University Of Applied Sciences. The Thesis is part of the larger operating instruction library about Siemens S7 TIA Portal.

This thesis contains instructions about specific subjects in PLC -programming, which have had notable changes from old Siemens STEP-7 version. This thesis focuses on the characteristics of PLC -tags and how to save them, what are different programs structures, file units and organization units. There are also told about program monitoring and searching of the errors. In every subject there have been made clear instructions, using text and images. Assumption is that that the reader is familiar with the PLC -programming and knows at least basics, before starting to read this, or starts reading from the previous parts of the operating instructions library, where those subjects are taught.

Making of this thesis was started by installing the software to own personal computer and by learning the basics about using it. The next step, was studying the subjects of this work in more detail. One of the challenges of this job, was lack of source material, so the work methods used, were mainly self-studying, reporting about studies and taking screen shots.

The outcome was operating instruction manual covering different subjects, which has been explained more or less in depth depending on the subject. The use of this thesis remains to be seen, because the final results will be known only after usage in teaching.

---

## SISÄLLYS

1	JOHDANTO.....	6
2	Arvojen käsittely ja näyttö .....	7
	2.1 Yleistä .....	7
	2.2 Seuranta ja ohjaus .....	7
	2.3 Kenttälaite .....	7
3	Muuttujat - PLC Tags.....	8
	3.1 Yleistä .....	8
	3.2 Muuttujien lisääminen .....	8
	3.3 PLC -muuttujataulukko – Tag table.....	9
	3.4 Datatyypit.....	9
	3.5 Datatyyppien muunnostoiminnot.....	10
	3.6 Pysyvä muisti - Retentive memory .....	12
	3.6.1 Muistipaikkojen määrittely .....	12
	3.7 Lukujen esitystavat .....	12
	3.7.1 Näyttömuodot.....	13
	3.7.2 Watch tablen luominen .....	13
4	Ohjelmarakenne.....	15
	4.1 Ohjelmarakenteen vaihtoehdot .....	15
	4.2 Ohjelmayksiköt .....	15
	4.2.1 Ohjelmayksikön tyypit .....	15
	4.2.2 Uuden ohjelmayksikön lisääminen .....	16
	4.2.3 Ohjelmakoodin suojaaminen – Know -how protection .....	16
	4.2.4 Ohjelmayksiköiden ominaisuudet.....	17
	4.2.5 Ohjelmaeditorin asetukset.....	17
	4.2.6 Ohjelmayksiköiden kutsuminen.....	19
	4.2.7 Ohjelmayksikön vertailu .....	20
5	Tiedostoyksiköt .....	22
	5.1 Tiedostoyksiköt ja niiden käyttö .....	22
	5.2 Datablokkien luominen .....	22
	5.3 Tiedostoyksikön määrittely.....	23
	5.3.1 Määrittelyn asetukset .....	24
	5.3.2 Tiedostoyksikön muuttujien osoitteet .....	24
	5.4 Muuttujien käsittely tiedostoyksiköissä.....	25
	5.4.1 Perinteinen käsittely .....	25
	5.4.2 Yhdistetty käsittely .....	26
6	Organisaatioyksiköt.....	27

6.1	Yleistä .....	27
6.2	Käynnistys .....	27
6.3	Syklinen ohjelmakäsittely .....	27
6.4	Syklisen ohjelman keskeytys ja prioriteetit .....	27
6.5	Periodinen ohjelmakäsittely .....	28
6.6	Tapahtumaohjattu ohjelmakäsittely .....	29
7	Seuranta ja virheen etsintä.....	30
7.1	Virhetyypit .....	30
7.2	Etsintätyökalut .....	30
7.2.1	Järjestelmän havaitsemat virheet.....	30
7.2.2	Hardware diagnostiikka .....	30
7.2.3	Online&diagnostics.....	30
7.2.4	Diagnoosipuskuri .....	31
7.2.5	Toiminnalliset virheet .....	33
8	YHTEENVETO .....	34
	LÄHTEET.....	35

## 1 JOHDANTO

Tampereen ammattikorkeakoulun koneautomaatiolaboratorio on uudistanut Siemens ohjelmistonsa uudempaan TIA Portal V11 -versioon (Totally Integrated Automation), joka korvaa vanhan STEP-7 -version. Tarkoituksena on tehdä Tampereen ammattikorkeakoululle uuteen ohjelmistoon monipuoliset ja laajat käyttöohjeet, ja tämä opinnäytetyö on osa laajempaa käyttöohjekirjastoa jotka perehtyvät Siemens TIA Portal V11 -ohjelmistoon.

Tästä ohjelmistosta on huonosti tietoa saatavilla ja harva yritys on ottanut sitä vielä täysimittaisesti käyttöön. Tämän lisäksi syynä opinnäytetyön aiheen valintaan oli oma mielenkiinto PLC -ohjelmointia kohtaan. Halusin tehdä opinnäytetyön aiheesta, josta on itselle ja muille hyötyä myöhemmin työelämässä.

Siemens TIA Portal v11 Mahdollistaa monipuolisemman suunnittelun yhden työkalun avulla. Esimerkiksi logiikkojen, käyttöliittymien ja turvaratkaisujen ohjelmointi on yhdistetty, suunnittelu- ja tuotantoprosessit on mahdollista tehdä samalla ohjelmistolla. PLC -ohjelmointi ei merkittävästi eroa vanhasta STEP-7 -versiosta, mutta TIA portal sisältää monipuolisempia ominaisuuksia jotka tehostavat ja helpottavat suunnittelua. Tässä opinnäytetyössä on tarkoitus perehtyä TIA Portal V11 edellä mainittuihin uusiin erikoisominaisuuksiin jotka eroavat vanhasta versiosta. Erityisesti työssä perehdytään muuttujien tallentamiseen sekä eri muuttujien ominaisuuksiin, ohjelmarakenteisiin, tiedostoyksiköihin, organisaatioyksiköihin sekä ohjelman seurantaan ja siellä olevien virheiden etsimiseen. Tavoitteena on antaa selkeät ohjeet uusien toimintojen hyödyntämiseksi, jotta uudesta ohjelmistosta saataisiin kaikki hyöty irti.

Automaatiotekniikka on tulevaisuudessa kasvava teollisuuden ala, josta PLC -suunnittelu, sekä ohjelmistojen monitorointi käyttöönoton yhteydessä, on tärkeä osa-alue. Siemens on johtava ohjelmoitavien logiikkayksiköiden valmistaja, joten lähes jokaisen automaatioinsinöörin on hyvä omata vähintään perustiedot, heidän ohjelmistoista ja laitteista.

## 2 ARVOJEN KÄSITTELY JA NÄYTTÖ

### 2.1 Yleistä

Nykyisillä ohjauslaitteilla digitaalisten arvojen käsittely on entistä suuremmissä merkityksessä ja tulevat vastaan ohjaustekniikan kaikilla osa-alueilla, esimerkiksi prosessiseurannassa ja toimilaitteiden ohjauksessa.

### 2.2 Seuranta ja ohjaus

Prosessiseuranta antaa käyttäjälle jatkuvaa tietoa prosessin, eli koneiden ja laitteiden, tilasta sekä mahdollistaa niiden ohjauksen yleiskuvallisesti ja ymmärrettävässä muodossa. Vanhojen yksinkertaisten anturi- ja toimilaitteiden sijasta nykyään käytössä on enimmäkseen ”älykkäitä” ohjaus- ja valvontalaitteita, kuten operaatiopaneelit ja videovalvomot.

### 2.3 Kenttälaitte

Uusimmat kenttälaitteet, jotka seuraavat tai ohjaavat prosessisuureita, liitetään suoraan kenttäväylään jolloin tietoa siirretään digitaalisina suureina. Analogiliitännät ovat entistä harvinaisempia.

### 3 MUUTTUJAT - PLC TAGS

#### 3.1 Yleistä

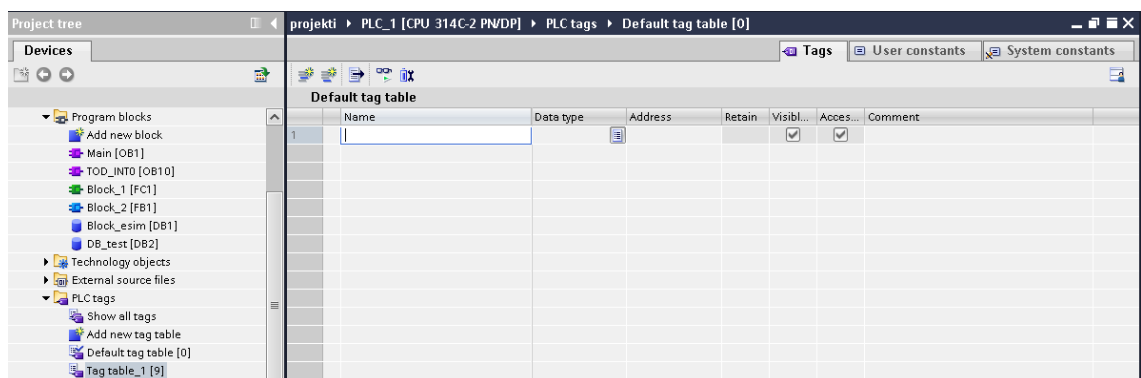
PLC -tagien, eli muuttujien, avulla voidaan I/O tiedoille määrittää muistipaikka ja selkeä nimike.

Kaikilla osoitteilla on sekä absoluuttinen että symbolinen osoite. Absoluuttinen osoite kertoo mihin tavuun ja bittiin kyseinen I/O on asetettu. Symbolinen osoite on itse määritelty nimike tietystä I/O:sta, jota voidaan käyttää myös ohjelmoinnissa, mutta sille on määriteltävä absoluuttinen osoite jotta se voidaan kääntää ja ladata CPU:lle.

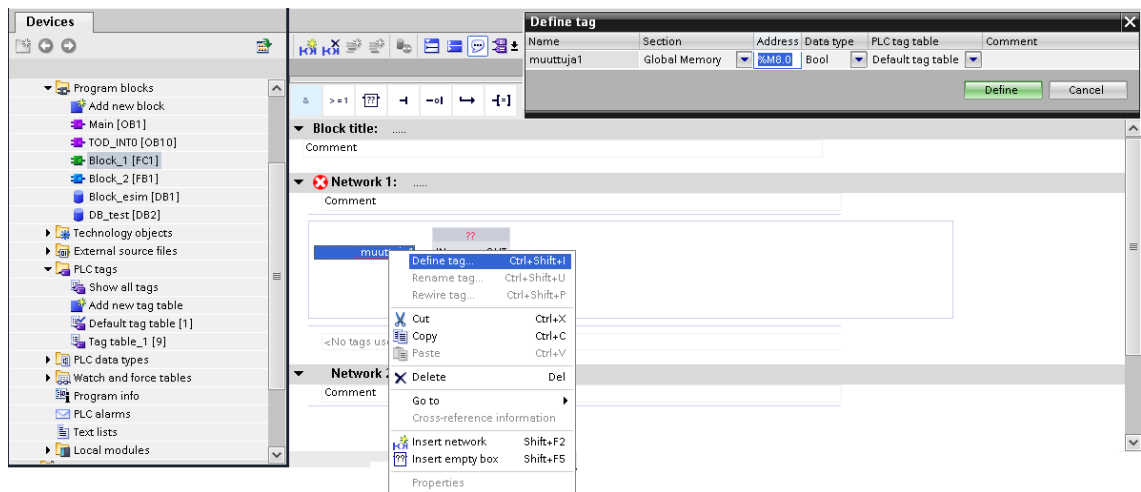
#### 3.2 Muuttujien lisääminen

PLC -tagille määritetään absoluuttinen ja symbolinen osoite, datatyyppi, sekä voidaan lisätä kommentti. Esimerkiksi jollekin anturille voidaan antaa symboliseksi nimeksi Ant1, datatyyppi: BOOL, absoluuttinen osoite: I136.0 ja kommentteihin selventää mikä anturi on kyseessä ja missä se sijaitsee. Tämän jälkeen ohjelmoinnissa voidaan käyttää Ant1 nimikettä eikä tarkkaa muistipaikkaa tarvitse muistaa.

Muuttujien lisääminen onnistuu kirjoittamalla muuttujan symbolinen nimi suoraan ohjelmaan ja määrittelemällä se etukäteen muuttujataulukon (kuva 1.1) tai määrittelemällä se kuvan 1.2 osoittamalla tavalla, ohjelmoinnin yhteydessä.



KUVA 1.1 Tagien lisääminen muuttujataulukon



KUVA 1.2 Tagien lisääminen ohjelmoinnin yhteydessä

### 3.3 PLC -muuttujataulukko – Tag table

Useiden muuttujien hallinnointi ja lisääminen tapahtuu muuttujataulukon avulla. Listaan voidaan lisätä kaikki ohjelmassa tarvittavat muuttujat ja muokata niiden asetuksia.

Muuttujia on mahdollista kopioida koska usein tarvitaan montaa samantyylistä muuttujaa. Kopiointi tapahtuu painamalla fill symbolia valitun solun oikeasta alanurkasta ja vetämällä osoitinta haluttujen solujen yli, hiiren painike alas painettuna. Uudet nimet täydentyvät juoksevilla numeroinnilla ja osoitteet täydentyvät seuraavaan vapaaseen muistipaikkaan, datatyypin huomioiden.

Myös Exceliä on mahdollista käyttää muuttujien editoimisessa. Muuttujien siirto ohjelmien välillä tapahtuu windowssin copy & paste toiminnolla tai Excelin import/Export toiminnolla.

### 3.4 Datatyypit

Datatyypit määrittävät muuttujien ominaisuudet, eli käytännössä tiedon esitystavan ja sallitut rajat. Esimerkiksi BOOL (boolean) tyyppiset muuttujat ovat vain 1-bittisiä, eli 1 tai 0. REAL taas sisältää 32 bittiä ja mahdollistaa suuremmat lukuarvot. Yleisimpiä käytössä olevia datatyyppejä ovat

Bittidatatyypit: BOOL, BYTE, WORD, DWORD

Matemaattiset datatyypit: INT, DINT, REAL

Aikatyytit: TIME

Yhdistetyt datatyypit: ARRAY, STRING

BOOL (boolean). Nämä muuttuja sisältää ainoastaan yhden bitin, jolloin sen arvo on 1 tai 0 (FALSE / TRUE).

BYTE eli tavu sisältää 8 bittiä. Yksittäiset bitit eivät ole merkityksellisiä.

WORD sisältää 16 bittiä (2 tavua). Yksittäiset bitit eivät ole merkityksellisiä.

DWORD (double word) sisältää 32 bittiä (4 tavua). Yksittäiset bitit eivät ole merkityksellisiä.

INT (integer) sisältää 16 bittiä ja on kokonaisluku, eli se ei saa sisältää desimaalipistettä.

INT ja DINT luvuissa voidaan etumerkillä (+/-) määrittää onko lukuarvo positiivinen vai negatiivinen.

DINT (double integer) sisältää 32 bittiä ja on INT tavoin myös kokonaisluku.

REAL (realiluku) sisältää 32 bittiä ja voi sisältää desimaalisia. Tapauksissa joissa on muuttuvia suureita ja jotka sisältävät desimaalisia on ainoa mahdollisuus käyttää reaali-lukuja. Reaalilukujen laskutoimitukset ovat suhteellisesti hitaampia mitä kokonaislukujen joten niitä ei kannata käyttää jos mahdollista.

TIME datatyyppi sisältää 32 bittiä ja käytetään yleisimmin laskureiden ym. yhteydessä ilmoittamaan aikaa. Data on mahdollista esittää päivissä (d), tunneissa (h), minuuteissa (m), sekunteissa (s) ja millisekunteissa (ms). Esim. TIME#24d20h31min20s10ms.

Vastaavia datatyyppejä ovat myös DATE, TIME\_OF\_DAY DATE\_AND\_TIME jne.

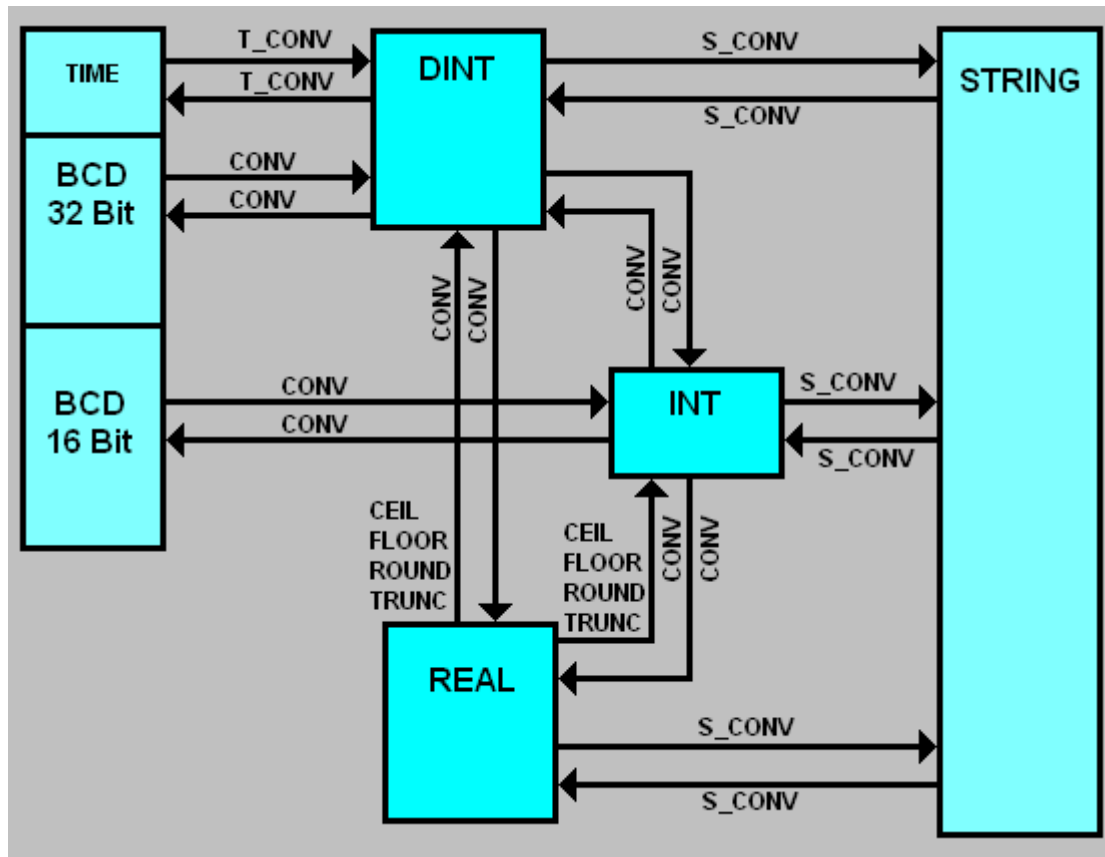
ARRAY (taulukko) bittimäärää ei ole esimääritely, sen pituus on käyttäjän määriteltävissä. Tämän avulla voidaan luoda samankaltaisten datatyyppien kokonaisuuksia.

STRING (merkkijono) pituus on ARRAY tavoin käyttäjän määriteltävissä. STRING datatyyppiä käytetään kun halutaan syöttää pitkiä tekstijonoja. esim ASCII formaatissa.

### 3.5 Datatyyppien muunnostoiminnot

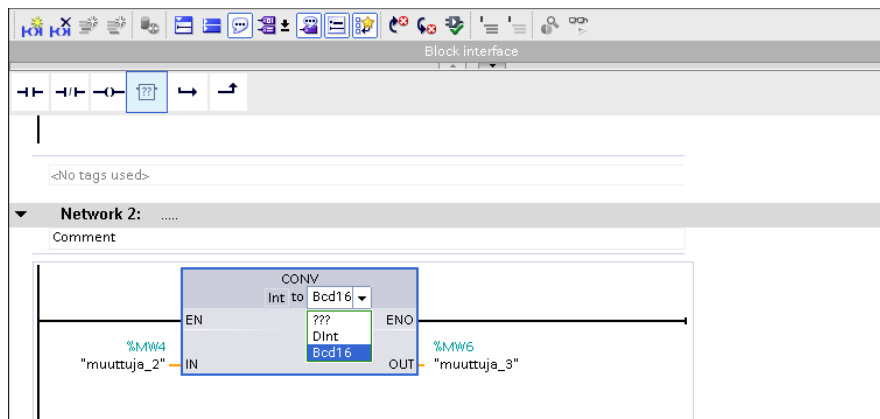
Datatyyppinä on mahdollista muuntaa toiseen muotoon, eri datatyypeiksi. Kaikkia datatyyppinä ei ole kuitenkaan mahdollista muuntaa, eli ”convertoida” keskenään.

Oheisesta kuvasta (KUVA 2.1) käy ilmi mitkä yleisimmin käytetyistä datatyypeistä ovat mahdollisia muuntaa keskenään ja mitä toimintoa muuntamiseen tarvitaan. (BCD = Binary-Coded Decimal, esim. WORD ja DWORD)



KUVA 2.1 Datatyyppien muunnokset (Siemens Oy. Siemens Oy:n verkkosivut 2011a)

Helpon datatyyppien muunnos onnistuu lisäämällä tyhjä laatikko (empty box) ohjelma-kiertoon ja antamalla sen nimeksi esim. ”CONV”, jolloin ohjelma luo laatikosta ”converterin” jonka avulla on mahdollista muuntaa annettu IN muuttuja eri datatyyppien OUT muuttujaksi. Halutut datatypit saadaan valittua laatikon yläosan vetovalikoista. ”CONV” toiminnon lisäksi eri datatyyppien kanssa tarvitaan `S_CONV` ja `T_CONV` toimintoja.



KUVA 2.2 Datatyyppin muuntaminen.

### 3.6 Pysyvä muisti - Retentive memory

Jotta virtakatkosten aikana ei menetettäisi tietoja, voidaan muistipaikkoja, aikoja ja laskureita määrittää pysyvään muistiin. Tämä tapahtuu tallentamalla haluttu tieto pysyvälle muistialueelle. Pysyvän muistialueen tieto säilyy lämpimässä uudelleenkäynnistyksessä (warm restart). Kylmässä uudelleenkäynnistyksessä (cold restart) tieto nollautuu.

#### 3.6.1 Muistipaikkojen määrittely

Muistipaikkojen määrittely tehdään CPU:n parametreistä. Pysyvään muistiin voidaan määrittää haluttu määrä muistipaikkoja, aikapiirejä ja laskureita. Määrittely lähtee tavusta 0. Oletuksena pysyvillä muistipaikoilla ovat muistit MB0-MB15, laskurit C0-C7 ja kaikki aikapiirit ovat nollautuvia.

S7-300/400-laitteilla tiedostot (DB) ovat pysyviä sähkökatkon aikana.

### 3.7 Lukujen esitystavat

”Watch table” ja ”monitoring” toiminnoilla voidaan seurata muuttujien arvoja ohjelman aikana. Jokainen muuttuja voidaan lukea missä tahansa näyttömuodossa, kuitenkin datatyyppistä riippuen on yleensä jokin tietty muoto järkevin. Watch tablessa näyttömuotoa on mahdollista vaihtaa ”display format” valikosta. Watch tablen käyttö on hyödyllistä tilanteissa, joissa halutaan seurata tai pakottaa useita muuttujia, eri puolilta ohjelmaa. Esimerkiksi on mahdollista luoda oma testisivu, jonkin tietyn osa-alueen testaamiseen ohjelmasta.

### 3.7.1 Näyttömuodot

BOOL: Yksittäisen bitin tila (BOOL)

BIN: Yhden tai useamman bitin yksittäinen tila (BYTE, WORD, DWORD)

HEX: Muuttujan sisältö heksadesimaalimuodossa (BYTE, WORD, DWORD)

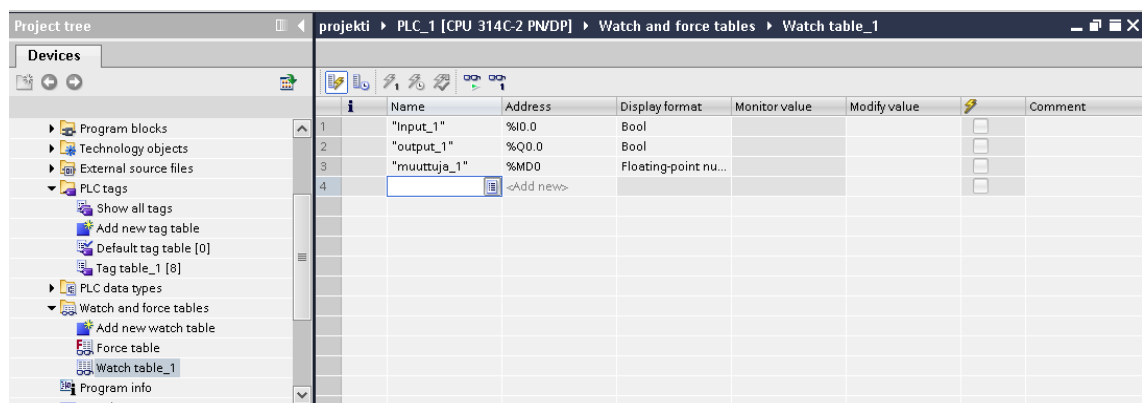
DEC: Muuttujan sisältö desimaalilukuna (INT, DINT)

FLOATING Point: Muuttujan sisältö reaalilukuna (REAL)

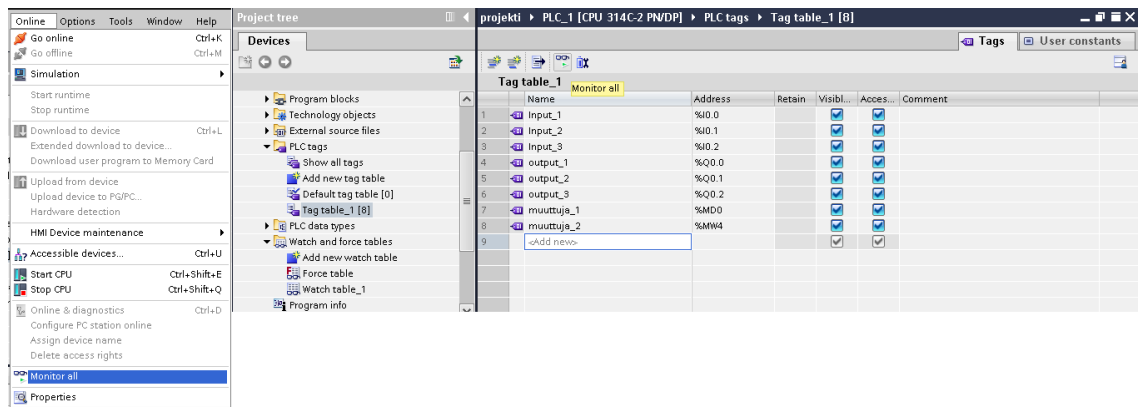
### 3.7.2 Watch tablen luominen

Watch tablen luominen on mahdollista project tree –valikosta josta valitaan ”watch and force tables”, jonka alta on mahdollista luoda uusi watch table. (KUVA 3.1)

Monitorin toiminto on mahdollista aktivoida yläpalkin kautta online valikosta tai esim. suoraan tag tablesta, monitoring ikonista. (KUVA 3.2)



KUVA 3.1 Watch table



## KUVA 3.2 Monitoring

## 4 OHJELMARAKENNE

### 4.1 Ohjelmarakenteen vaihtoehdot

Ohjelmarakenteita on kahta eri vaihtoehtoa, lineaarinen ja rakenteinen ohjelmarakenne. Lineaarinen ohjelma soveltuu pieniin automaatiotehtäviin ja suositellaan vain yksinkertaisissa ohjelmissa. Rakenteinen ohjelma helpottaa monimutkaisia ja vaikeita ohjelmointitehtäviä kun osatehtävät jaotellaan pienempiin alitehtäviin. Alitehtävät esitetään omissa ohjelmaosioissa jotka tunnetaan ohjelmayksikköinä, joista jokainen on itsenäinen osa käyttäjäohjelmaa. Rakenteisen ohjelman etuina on helpompi ymmärrettävyys, hallinta, muutosten toteutus, erillisten ohjelmaosioiden testaus, käyttöönotto ja yksittäisten osioiden standardisointi.

### 4.2 Ohjelmayksiköt

#### 4.2.1 Ohjelmayksikön tyypit

Automaatiojärjestelmät sisältävät toimintayksiköitä, mihin käyttäjäohjelma ja niihin liittyvät tiedot tallennetaan.

Ohjelmayksikkötyypit: Organisaatioyksikkö (OB), Toimintayksikkö (FB), Toiminta (CB) ja Tiedostoyksikkö (DB).

Organisaatioyksikkö muodostaa liittymäpinnan käyttöjärjestelmän ja käyttäjäohjelman välille. Koko ohjelma on mahdollista tallentaa organisaatioyksikköön, jolloin rakennetta kutsutaan lineaariseksi ja käyttöjärjestelmä kutsuu sitä syklisesti.

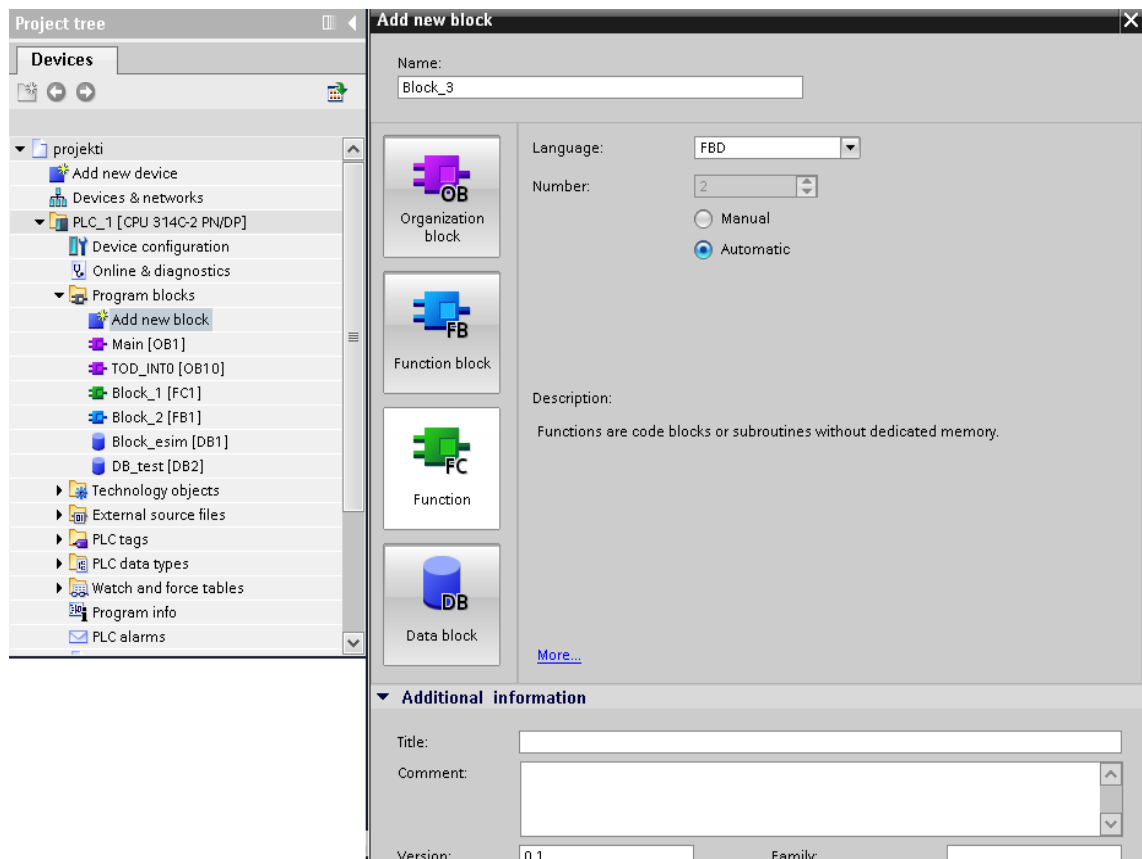
Toimintayksikkö sisältää ohjelman alitoimintoja. Soveltuu usein toistuvaan monimutkaisten alitoimintojen ohjelmointiin.

Toiminnalla on samat ohjelmointimahdollisuudet kuin Toimintayksiköllä, mutta omalla muistialueella jota kutsutaan instanssitiedostoyksiköksi. Soveltuu usein toistuvaan monimutkaisten toimintojen ohjelmointiin.

Tiedostoyksikkö sisältää käyttäjäohjelman data-alueita, jonne käyttäjädata voidaan järjestää hallitusti rakenteiseen järjestykseen.

#### 4.2.2 Uuden ohjelmayksikön lisääminen

Uusi ohjelmayksikkö lisätään CPU:n program blocks -alavalikosta. Uutta ohjelmayksikköä luotaessa sille pitää määrittellä yksikön tyyppi, ohjelmointikieli, symbolinen nimi ja numero. Additional information kohtaan, voidaan laittaa yksityiskohtaista tietoa yksiköstä.



KUVA 4.1 Uuden ohjelmayksikön lisääminen

#### 4.2.3 Ohjelmakoodin suojaaminen – Know-how protection

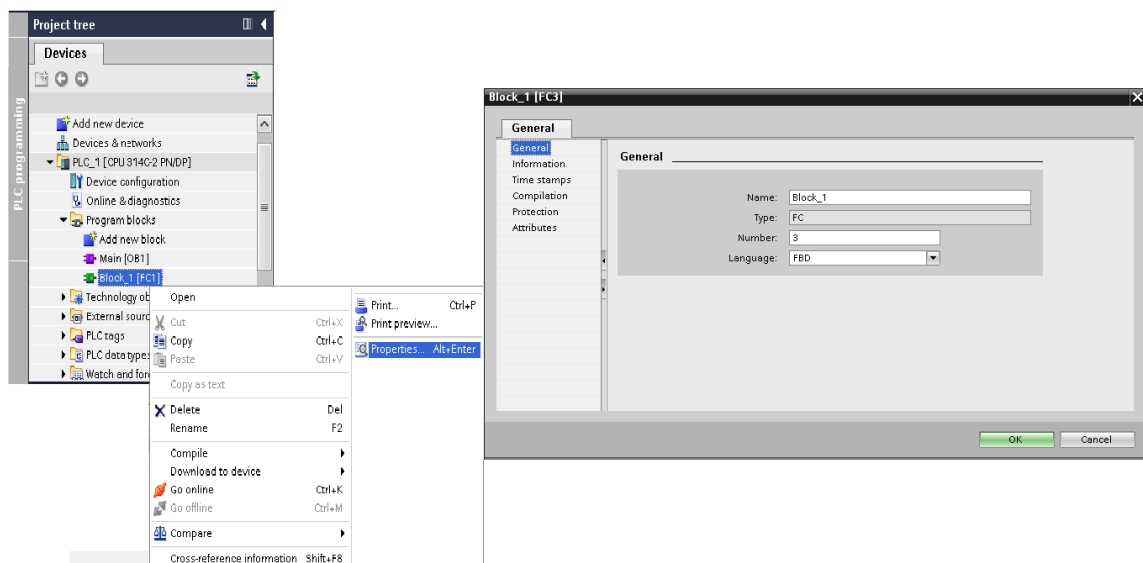
Ohjelmayksiköt voidaan suojata luvattomalta käytöltä salasanalla. Suojatusta yksiköstä on luettavissa ainoastaan: siirtoparametrit Input, output, InOut, Return, yksikön nimi, kommentit, ominaisuudet, ohjelmarakenne ja globaalit muuttujat, ilman tietoa käyttökohdasta.

Seuraavat toimenpiteet voidaan suorittaa: kopiointi ja poistaminen, kutsu ohjelmassa, sekä tarkastus ja päivitys käännöksen yhteydessä.

Salasanasuojattua yksikköä, joka on ladattu laitteeseen, ei ole mahdollista avata vaikka käytettävissä olisi oikea salasana. Jos salasana unohtuu, pääsy suojattuun yksikköön ei ole enää mahdollista.

#### 4.2.4 Ohjelmayksiköiden ominaisuudet

Ohjelmayksikön ominaisuudet saadaan näkyviin projektipuusta kaksoisklikkaamalla haluttua yksikköä, tai painamalla sitä hiiren oikealla napilla ja valitsemalla properties jolloin ominaisuudet avautuvat omaan ikkunaan.

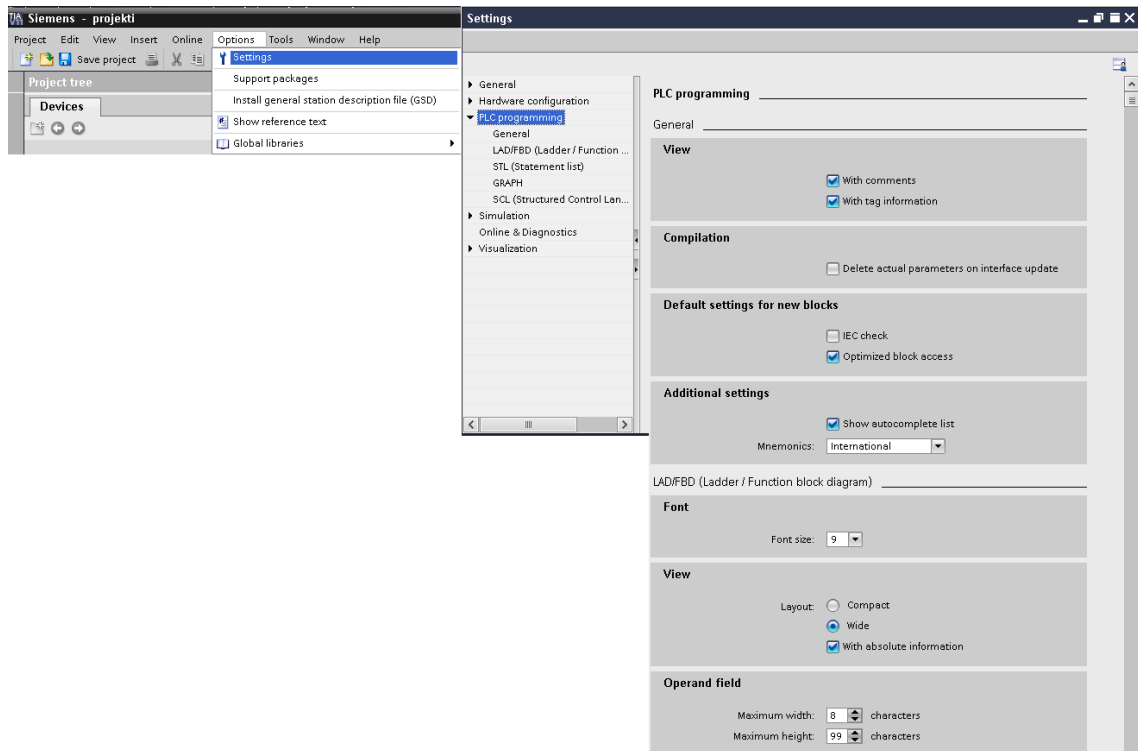


KUVA 4.2 Ohjelmayksiköiden ominaisuudet

Ominaisuuksista on mahdollista muokata seuraavia asioita: yksikön nimi, tyyppi, numerointi, ohjelmointikieli, tunnistamistiedot, aikaleimat, referenssi informaatiot ja suojausten määrittäminen.

#### 4.2.5 Ohjelmaeditorin asetukset

Ohjelmaeditorin asetuksiin päästään valitsemalla yläpalkista options -> settings -> PLC programming.



KUVA 4.3 Ohjelmaeditorin asetukset

”View” valikosta voidaan muokata millaisena ohjelmayksikkö tulee avattaessa esille.

”Compilation”. Kun ”Delete actual parameters on interface update” on valittuna, parametroitavien ohjelmayksikköjen kutsut päivitetään automaattisesti, jos ohjelmayksiköstä on myöhemmin poistettu parametreja.

”Default settings for new blocks”. IEC check: Tarkastetaan että käytettävillä muuttujilla on oikea datatyyppi. Esim. käskyn vaatiessa INT tyyppistä muuttujaa, WORD ei kelpaa, vaikka muuttujien koko on sama 16 bittiä.

”Optimized block access”. Tiedosto- ja ohjelmayksikköiden paikallismuuttujia voidaan käyttää vain symbolisesti. Etuina Optimaalinen muistin varaaminen ja lyhyempi datan haku aika.

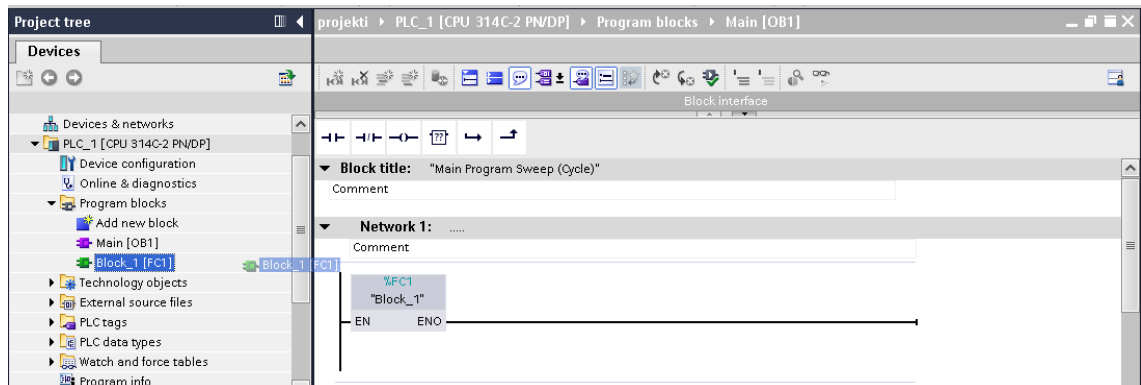
”Additional settings”. ”Mnemonics”: käskykielen valinta. International käyttää yleisesti käytössä olevia lyhenteitä (esim. I = Input), vaihtoehtoisesti voidaan käyttää Saksalaisia lyhenteitä.

”View” (LAD/FBD). Layout: Globaalimuuttujien osoitteiden esittäminen. ”With absolute information” valinnalla saadaan absoluuttiosoitteet näkyville.

”Operand field”. LAD ja FBD käksymboloiden maksimi leveys ja korkeus.

#### 4.2.6 Ohjelmayksikköiden kutsuminen

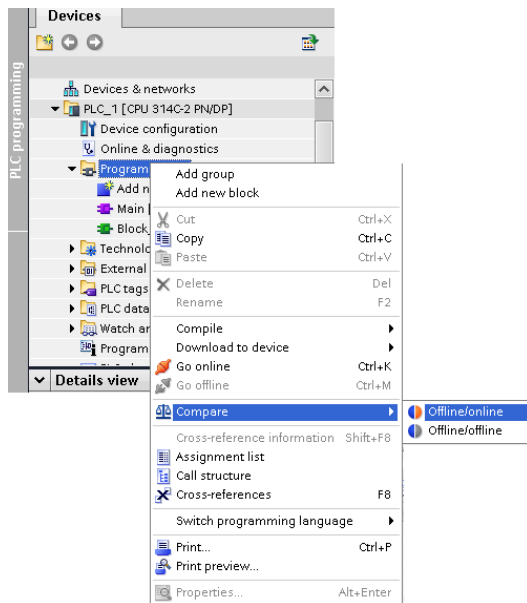
Ohjelmayksikköjä (esim. FC1) on mahdollista kutsua ohjelman sisällä (OB1), jolloin kutsutun yksikön käskyt tulevat ohjelmasuoritukseen. Ohjelmayksikön suorituksen jälkeen ohjelman käsittely palaa kutsuvaan yksikköön (OB1) ja jatkaa seuraavien käskyjen suorittamista. Kutsuttu ohjelmayksikkö tuodaan ohjelmaan vetämällä tai kopioimalla program blocks kansioista.



KUVA 4.4 Ohjelmayksikköiden kutsuminen.

#### 4.2.7 Ohjelmayksikön vertailu




















Ohjelmayksikköjä voidaan vertailla offline/online -välillä. Vertailu suoritetaan kahden ohjelmointilaiteelle tallennetun projektin välillä. Ohjelmayksiköiden vertailun tuloksista saadaan selville esimerkiksi jos alkuperäiseen ohjelmistoon on tehty muutoksia ja mihin yksiköihin muutokset on tehty.



KUVA 4.5 Ohjelmayksikön vertailu

Vertailu vertaa molempien projektien yksiköitä sekä komponentteja toisiinsa, ilmoittaa vertailun tuloksen symboleina ja antaa mahdollisesti tarkempia tietoja details valikossa.

Symbolit:

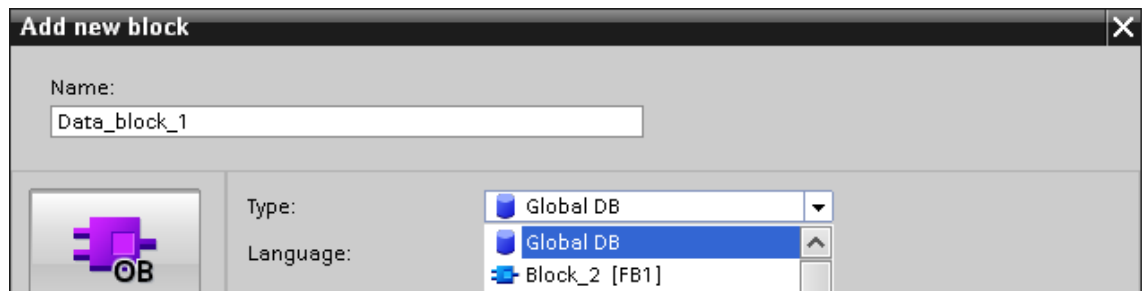
Symbol	Meaning
	Folder contains objects whose online and offline versions are different (only in the project tree)
	Online and offline versions of the object are different
	Object only exists online
	Object only exists offline
	Online and offline versions of the object are the same
	The connection with a CPU is currently being established.
	The CPU is not reachable at the set address.
	The configured CPU and the CPU actually present are of incompatible types.
	On establishment of the online connection to a protected CPU, the password dialog was terminated without specification of the correct password.
	No fault
	Maintenance required
	Maintenance demanded
	Error
	The module or device is disabled.
	The module or the device cannot be reached from the CPU (valid for modules and devices under a CPU).
	Diagnostics data is not available because the current online configuration data is different to the offline configuration data.
	The configured module or device and the module or device actually present are incompatible (valid for modules or devices under a CPU).
	The configured module does not support display of the diagnostics status (valid for modules under a CPU).
	The connection has been established, but the state of the module is currently still being determined.
	The configured module does not support display of the diagnostics status.
	Error in lower-level component: An error is present in at least one lower-level hardware component.

KUVA 4.6 Symbol (Siemens Oy. Siemens Oy:n verkkosivut 2011b)

## 5 TIEDOSTOYKSIKÖT

### 5.1 Tiedostoyksiköt ja niiden käyttö

Program blocks -hakemistoon voidaan luoda uusia tiedostoyksiköitä, tai avata jo projektissa olevia yksiköitä. Tiedostoyksikköjä on kahta eri vaihtoehtoa: Globaali -DB, joka on käyttäjäohjelman kaikkien koodiyksiköiden käytettävissä, sekä oheis -DB, joka on käytettävissä ainoastaan tietyn FB:n tai SFB:n sisällä.



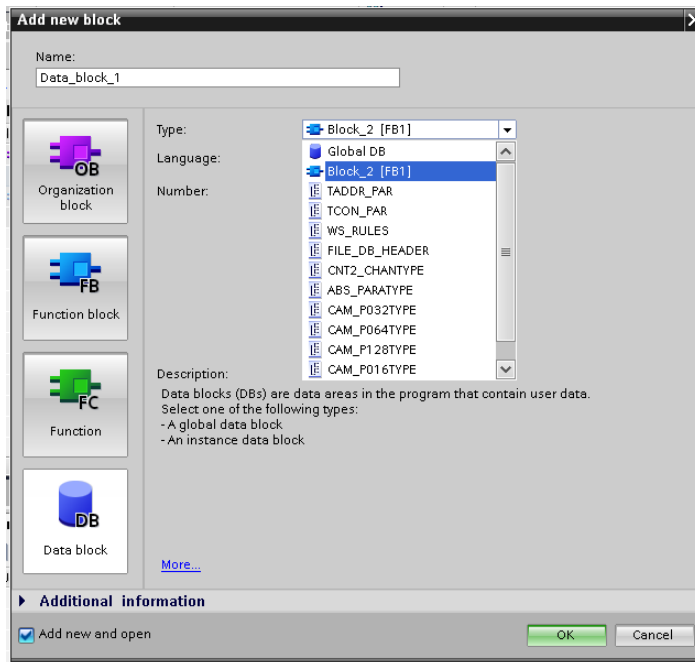
KUVA 5.1 Tiedostoyksiköt

DB -tiedostoyksiköihin on mahdollista tallentaa muuttujia jotka ovat vai tietyn FB -toimintayksikön tai käyttäjäohjelman kaikkien koodiyksiköiden käytettävissä.

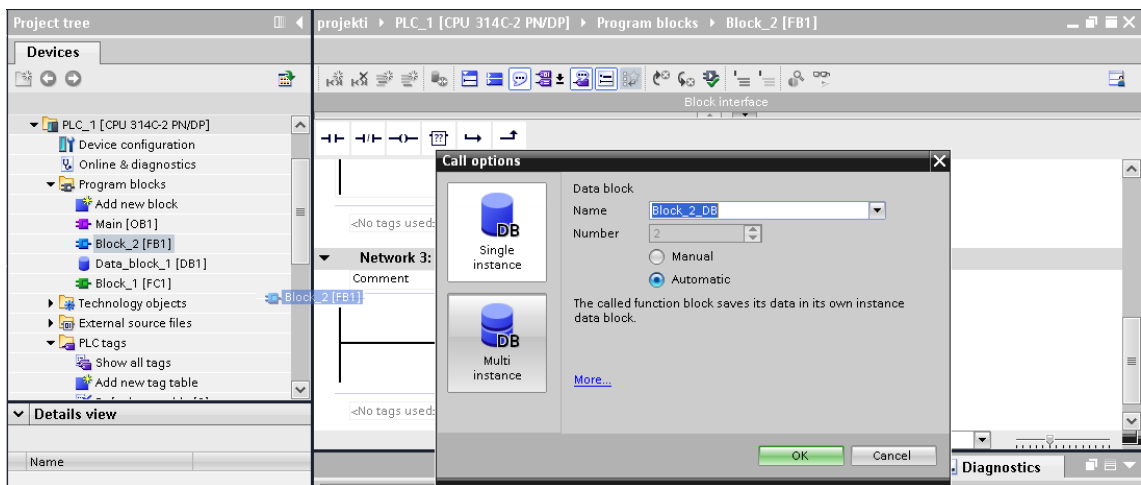
### 5.2 Datablokkien luominen

Datablokkien luominen on mahdollista project tree -valikon kautta ”add new block” toiminnolla. Tiedostoa luotaessa kerrotaan mille FB -yksikölle kyseinen tiedosto kuuluu. Tiedostolle valitaan haluttu vapaa numero ja nimi. (kuva 5.1)

Toinen vaihtoehto on luoda DB -yksikkö FB -yksikön kutsumisen yhteydessä. FB -yksikkö raahataan hiirellä program blocks -valikosta ohjelmaan, jolloin järjestelmä automaattisesti kysyy oheistiedostoa DB. (kuva 5.2)



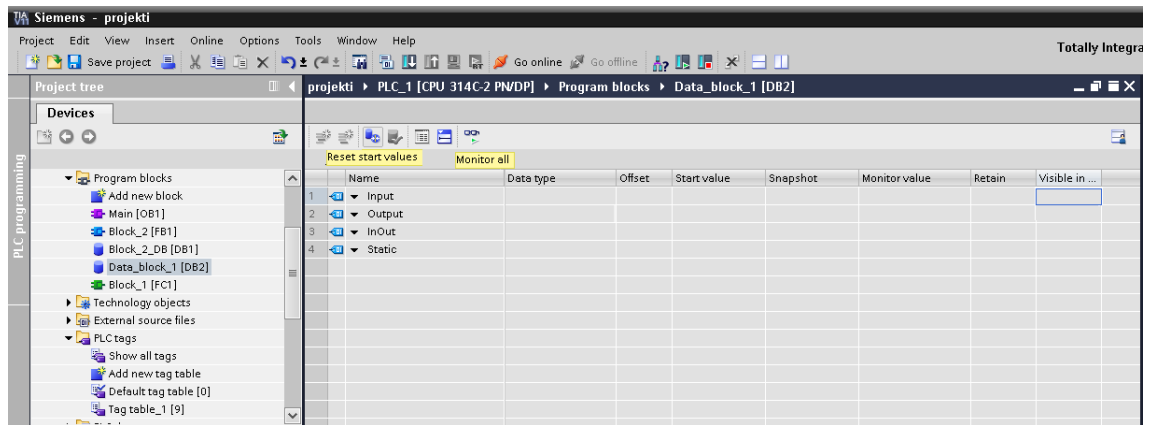
KUVA 5.2 Datablokkien luominen valikon kautta.



KUVA 5.3 Datablokkien luominen kutsumisen yhteydessä.

### 5.3 Tiedostoyksikön määrittely

Uuden muuttujan lisäys onnistuu DB -tiedostoyksikön yläpalkista tai painamalla hiiren oikeaa nappia uudelle riville ja valitsemalla add row/insert row (kuva 5.3) .



KUVA 5.4 Tiedostoyksikön määrittely

### 5.3.1 Määrittelyn asetukset

Tiedostoyksikön sisäiset muuttujat on mahdollista määrittää seuraavien asetusten avulla. ”Visible in HMI”. PLC:n muuttujat joilta tämä valinta puuttuu eivät ole näkyvissä HMI-laitteen muuttujavalinnassa (esim. WinCC).

”Accessible from HMI”. HMI-laite ei voi käsitellä muuttujia joilta puuttuu tämä valinta.

”Retain” määrittelee onko muuttuja tilansa säilyttävä. Retain muuttujat säilyttävät arvonsa perusnollaukseen asti.

”Offset”. Tiedostoyksikön muuttujien absoluuttiosoitteet.

”Start value”. Muuttujien alkuperäiset arvot, voidaan ylikirjoittaa koska tahansa. Alkuarvojen palautus on mahdollista DB yläpalkista ”Reset start values” valinnasta.

”Default value”. Oletusarvoja voidaan määrittellä ainoastaan FB sisällä. Globaali tiedostoyksikön arvoja ei ole mahdollista muuttaa.

”Monitoring value”. Muuttujan sen hetkinen arvo CPU:n työmuistissa

”Snapshot”. Tietyllä ajanhetkellä tallennettu monitoring arvo.

### 5.3.2 Tiedostoyksikön muuttujien osoitteet

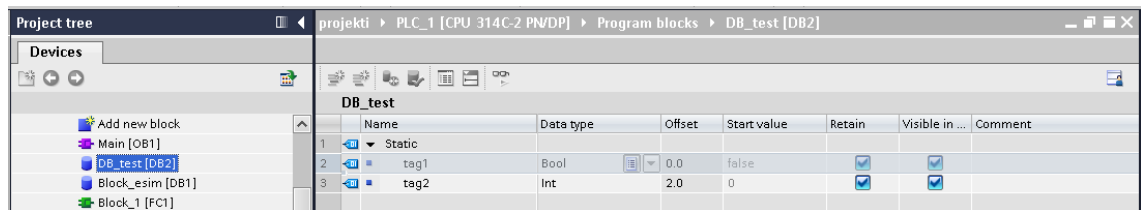
Tiedostoyksikön dataelementtien osoitteenanto tapahtuu tavuittain, kuten merkkereillä. Datatavuja, datasanoja tai kaksoissanoja voidaan ladata (L) ja kopioida (T). Datasanoissa

annetaan ensimmäinen tavuosoite (esim. LDBW 2) operaatioissa ja tästä osoitteesta lähtien ladataan 2 tavua, kaksoissanoissa 4 tavua.

#### 5.4 Muuttujien käsittely tiedostoyksiköissä

Tiedostoyksikköjä käsitellessä tiedostoyksikkö avataan ennen sen sisältämien muuttujien käyttöä. Aiemmin auki ollut tiedostoyksikkö sulkeutuu automaattisesti uuden avautuessa. Avauksen jälkeen yksittäisiä muuttujia voidaan osoittaa tiedostoyksikön sisältä. Avaus voidaan tehdä perinteisellä tai yhdistetyllä käsittelyllä

Muuttujia voidaan osoittaa tiedostoyksikön sisältä: DBX = bitteinä, DBB = tavuina, DBW = sanoina ja DBD = kaksoissanoina.



KUVA 5.5 Muuttujien käsittely.

##### 5.4.1 Perinteinen käsittely

Perinteisessä käsittelyssä tiedostoyksikkö avataan ennen sen sisältämän muuttujan käyttöä. Tiedostoyksikkö voidaan avata absoluutti- tai symbolisella osoitteella, tiedostoyksikön muuttujia pystyy käsittelemään ainoastaan absoluuttisilla osoitteilla, joka hankaloittaa ohjelman lukemista ja virheiden korjaamista.

Network 1: .....				
Comment				
1	OPN	"DB_test"	%DB2	
2	A	%DBX0.0	%DBX0.0	
Network 2: .....				
Comment				
1	OPN	"DB_test"	%DB2	
2	T	%DBW2	%DBW2	
3				

KUVA 5.6 Perinteinen käsittely.

#### 5.4.2 Yhdistetty käsittely

Yhdistetyssä käsittelyssä tiedostoyksikön avaus tehdään samassa käskyssä muuttujan käsittelyn kanssa, jolloin voidaan käyttää absoluuttiosoitteiden lisäksi myös symbolisia osoitteita.

Network 3: .....				
Comment				
1	A	"DB_test".tag1	%DB2.DBX0.0	
2				
3				
Network 4: .....				
Comment				
1	T	"DB_test".tag2	%DB2.DBW2	
2				

KUVA 5.7 Yhdistetty käsittely.

## 6 ORGANISAATIOYKSIKÖT

### 6.1 Yleistä

Organisaatioyksiköt eli ”OB”:t muodostavat liitännäisen käyttöjärjestelmän ja käyttäjäohjelman välille. Koko ohjelma on mahdollista suunnitella yhden organisaatioyksikön (OB1) sisälle mutta vähänkin monimutkaisemmissa ja laajemmissa ohjelmissa se on järkevää jakaa eri yksiköihin.

### 6.2 Käynnistys

Ohjelma suorittaa tietyt rutiinit käynnistyksen yhteydessä. mm. prosessikuvauksen nolaukset, OB100 (101/102) käynnistysohjelman suorituksen ja lähtöjen vapauttamisen, jonka jälkeen siirrytään sykliseen kiertoon .

Käynnistysohjelmat voidaan jakaa OB100, OB 101 ja OB 102. Uuskäynnistyksessä (warm restart) suoritetaan OB100, jälleenkäynnistyksessä (hot restart) OB101 ja kylmäkäynnistyksessä (cold restart) OB102. Jos käynnistystapaa vastaavaa OB:ta ei ole ohjelmoituna, OB suoritus jää välistä ja muut rutiinit suoritetaan samassa järjestyksessä.

Käynnistyksen yhteydessä käynnistysohjelma tekee tarvittavat alustukset ja esim. alustaa kommunikaatiokortit, ko. OB:n paikallisedata päivitetään ja sinne kirjataan mm. tapahtumasyitä ja kutsun kellonaika.

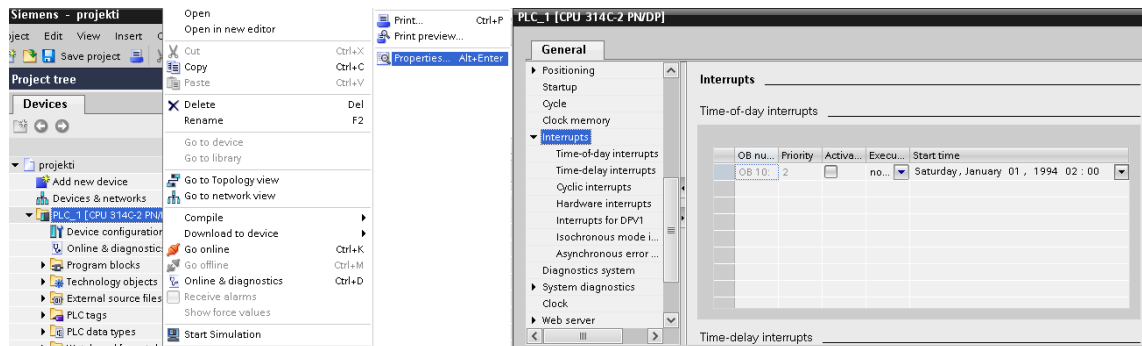
### 6.3 Syklinen ohjelmakäsittely

Syklisessä ohjelmakäsittelyssä OB1:stä kutsutaan jatkuvassa ohjelmakerrossa. Sykliajalla tarkoitetaan ohjelmakiertojen välistä viivettä. Reaktioaika tarkoittaa käyttöjärjestelmän vaatimaa aikaa lisättynä käskyjen käsittelyajalla. Reaktioaika voi nopeimmillaan olla n. sykliajan mittainen tai kestää pisimmillään n. kaksi sykliaikaa.

### 6.4 Syklisen ohjelman keskeytys ja prioriteetit

Syklinen ohjelma on mahdollista keskeyttää tietyn ajan välein, tiettyyn kellonaikaan, määritellystä tapahtumasta riippuen, tai virhetilanteissa. Ohjelmakierrossa OB1:llä on alhaisin prioriteetti(1).OB1 kierto pysähtyy kun jotain toista yksikköä kutsutaan ja ko. yksikön suorittamisen jälkeen ohjelmakierto palautuu OB1 siihen kohtaan missä keskeytys tapahtui.

Isomman prioriteetin omaava yksikkö keskeyttää aina pienemmän yksikön toiminnan. 1 prioriteetti on alin ja 29 korkein. Saman prioriteetin omaavat yksiköt eivät keskeytä toistensa suoritusta vaan ne suoritetaan siinä järjestyksessä missä ne on havaittu. Virhe-OB:n prioriteetit eivät välttämättä aina ole vakioita. Prioriteetti voi olla esim. 28 käynnistyksen aikana ja 26 OB1 suorituksen aikana.



KUVA 6.1 Ohjelma keskeytys.

## 6.5 Periodinen ohjelmakäsittely

Periodinen ohjelmakäsittely tarkoittaa aikakeskeytystä (OB30-38) tai kellokeskeytystä (OB10-17). Aikakeskeytystä käytetään kun halutaan pysäyttää syklinen käsittely ja kutsua jotain ohjelmaa tietyin väliajoin. Kellokeskeytystä ohjelma käydään aina samaan aikaan läpi. esim. joka päivä tiettyyn kellonaikaan.

## 6.6 Tapahtumaohjattu ohjelmakäsittely

Tapahtumaohjatussa ohjelmakäsittelyssä voidaan reagoida nopeasti prosessitapahtumiin käyttämällä prosessikeskeytystä (OB40-47), jolloin syklinen ohjelma keskeytetään määrätystä tapahtumasta.

Viivekeskeytys reagoi myös prosessitapahtumiin mutta toiminnot tapahtuvat viiveellä eivätkä keskeytä kiertoa välittömästi.

Virhekeskeytyksillä (asynkroniset OB80-87, synkroniset OB121/122) voidaan määritellä laitteen käyttäytyminen virhetilanteissa.

## 7 SEURANTA JA VIRHEEN ETSINTÄ

### 7.1 Virhetyypit

Virhetyypit voidaan jakaa kahteen luokkaan. Ensimmäiseen luokkaan kuuluvat järjestelmän havaitsemat virheet, joita ovat korttiviati, oikosulku signaalijohtimissa, syklivalvonta-ajan ylitys ja ohjelmointivirhe. Toiseen luokkaan kuuluvat toiminnalliset virheet. Tähän virheluokkaan lasketaan tapaukset jossa haluttua toimintoa ei tapahdu, tai se ei tapahdu oikein, prosessivika esim. anturissa tai kaapeloinnissa ja ohjelmointivirheet, joita ei huomattu ohjelmoinnissa ja käyttöönotossa.

### 7.2 Etsintätyökalut

#### 7.2.1 Järjestelmän havaitsemat virheet

Järjestelmä estää CPU:n vaihtamisen STOP tilasta RUN tilaan jos se havaitsee virheen, tästä johtuen useimmat havaitut viat joudutaan selvittämään STOP tilassa.

Tässä tapauksessa työkaluja virheenetsintään on Diagnosoipuskuri ja Hardware-Diagnosi. Tulojen ja lähtöjen seuranta ei ole mahdollista koska laitetta ei saada run tilaan.

#### 7.2.2 Hardware diagnostiikka

"Diagnostic hardware"-toiminta löytyy avaamalla projektipuusta "Device configuration" online tilassa. Avautuvasta näkymästä saadaan tietoa moduulien tilasta ja operointimoodista. Painamalla CPU:ta tai I/O korttia hiiren oikella napilla ja valitsemalla "Online & diagnostics", saadaan auki ko. laitteen diagnostiikka puskuri.

#### 7.2.3 Online&diagnostics

Online & diagnostics sisältää tietoja laitteen tiedoista, tilasta ja tapahtumista. Seuraavia tietoja voidaan seurata tältä välilehdeltä.

General: Liitettyjen moduleitten tyyppinumerot ym. tietoa ohjelmiston ja hardwaren versioista.

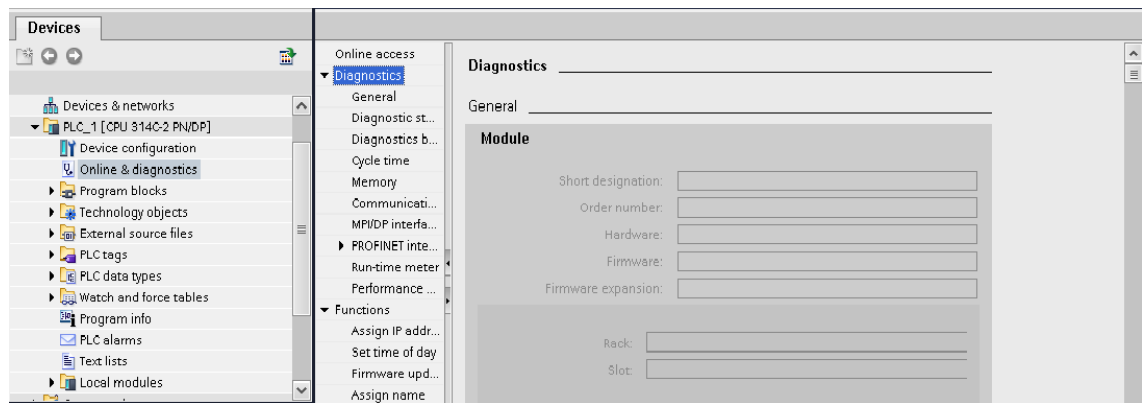
Diagnostics buffer: Diagnostiikkatapahtumat aikajärjestyksessä.

Cycle time: Tietoa valitun syklin valvonta-ajoista ja viimeisimmistä sykliajoista.

Memory: Latausmuistien ja työmuistin koko ja kuormitus.

Communications: Kommunikaatioresurssit ja yhteyksien yleiskatsaus.

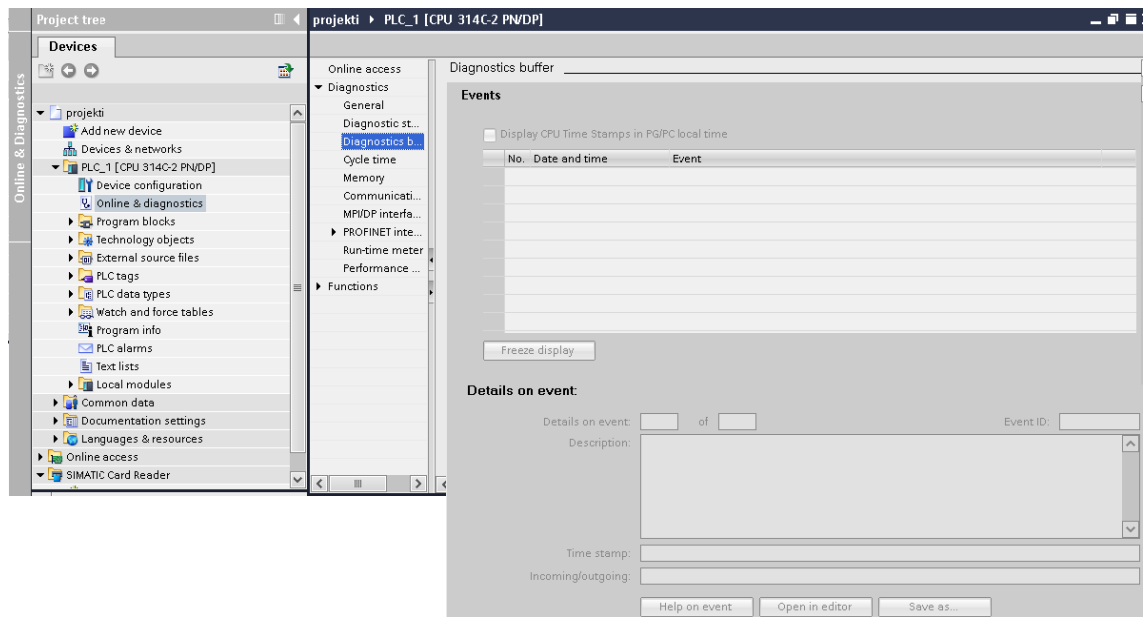
Performance data: Järjestelmäyksiköt (SB), organisaatioyksiköt (OB) ja käytettävissä olevat osoitealueet.



KUVA 7.1 Online&diagnostics.

#### 7.2.4 Diagnoosipuskuri

Diagnoosipuskuriin tallentuvat kaikki laitteessa tapahtuvat virheet, niiden päiväys, kellonaika ja mahdolliset lisätiedot tapahtumista. Puskuri ei nollaannu koskaan mutta laitteesta riippuen sinne mahtuu n. 100 tapahtumaa ja ylimenevät kirjoitetaan vanhimpien tapahtumien päälle. ”Help on event” painikkeesta saadaan lisätietoa valitusta tapahtumasta. ”Open in editor painike”. Jos ohjelmassa on virheellinen käsky tai kohdistettavissa oleva virhe saadaan se avattua tästä painikkeesta.



KUVA 7.2 Diagnoosipuskuri.

### 7.2.5 Toiminnalliset virheet

Toiminnallisten virheiden, joita järjestelmä ei havaitse, selvittäminen jää enemmän käyttäjän vastuulle. Usein kyseessä on kaksoisohjaus, jossa samaa muuttujaa ohjataan kahteen eri tilaan samanaikaisesti tai vajaan kytkennät. Virheen selvittämisessä voi käyttää apuna muuttujien pakko-ohjausta (force), muuttujataulukon seuraamista (watch and force tables) ja ristiviittauksia (cross-reference).

Muuttujataulukon avulla voidaan seurata eri sijainneissa olevia muuttujia samanaikaisesti ja Pakko-ohjaamalla niitä voidaan seurata niiden tilanmuutoksia. Ristiviittauksen avulla nähdään missä kohdissa ohjelmaa kyseistä muuttujaa luetaan ja kirjoitetaan ja sen avulla on helposti selvitettävissä kaksoisohjaukset

## 8 YHTEENVETO

Opinnäytetyön tarkoituksena oli perehtyä Siemens S7 TIA Portal -ohjelmiston uusiin erikoisominaisuuksiin ja tehdä sen perusteelta ohjekirja opetuksen tueksi. Tarkoituksena oli perehtyä tärkeisiin aihealueisiin PLC -ohjelmoinnissa, joihin on tullut merkittäviä muutoksia vanhasta Siemens STEP-7 -versiosta, käydä lävitse esimerkiksi muuttujien tallentamista ja eri muuttujien ominaisuuksia, ohjelmarakenteita, tiedostoyksiköitä, organisatoyksiköitä sekä ohjelman seuranta ja siellä olevien virheiden etsimistä. Jokaisesta aihealueesta oli tarkoitus antaa selkeä ohjeistus kuvien ja tekstin avulla.

Mielestäni asetut tavoitteet, saavutettiin kohtalaisesti. Jokaisesta osa-alueesta, mistä oli ennalta tarkoitus tehdä ohjeistus, saatiin tehtyä jotakin. Suurin osa tekstistä on lisäksi selvitetty kuvien avustuksella.

Vaikka työ ei itsessään ollut haastava, haastavaa oli saada tehtyä siitä opinnäytetyö. Lähdeaineiston puutteellisuus, ohjekirjan rakenteen muokkaaminen opinnäytetyöksi kelpaavaksi ja erillisen PLC -projektin puuttuminen aiheuttivat työhön haasteita. Koska S7 TIA Portal -ohjelmisto on vasta otettu käyttöön monessa paikassa, siitä on hankala hakea tietoa muualta kuin Siemens omilta sivuilta. Osaa saatavilla olevasta informaatiosta ei saa levittää yleiseen jakoon, eikä ole myöskään tarkoituksen mukaista kopioida samoja asioita suoraan Siemensin omilta sivuilta. Rakenteelliset ongelmat opinnäytetyössä aiheuttivat myös päänvaivaa. Koska kyseessä on Ohjekirja ja aihealue käsittelee paljon ns. listausta vaativia, yksittäisiä asioita, niitä oli erittäin hankala saada esitettyä opinnäytetyöksi sopivassa muodossa. Monet kappaleet joissa käsiteltiin esimerkiksi eri datatyyppejä tai niiden ominaisuuksia, olisi ollut helppo kirjoittaa ymmärrettävään muotoon ranskalaisia viivoja apuna käyttäen. Yksi asia mikä olisi helpottanut myös työn tekemistä, olisi ollut valmis laajamittainen PLC -projekti, joka olisi rakennettu tätä ohjelmistoa hyödyntäen. Nyt jouduin tekemään hyvin suppeita testiohjelmiä eri ominaisuuksien selittämistä ja kuvien ottamista varten. Uskon että asiat olisi saanut esitettyä hieman ymmärrettävämmiin ja kuvista selkeämpiä, jos ne olisi testattu perusteellisemmin. Tästä johtuen lopullista arviota työstä joudutaan odottamaan kunnes se otetaan käyttöön opetuksen yhteydessä ja sitä käyttävät hyödykseen oppilaat, jotka eivät vielä ole perillä näistä asioista.

## LÄHTEET

Siemens Oy. Siemens Oy:n verkkosivut 2013a. Product support.

<http://support.automation.siemens.com/ww/llisapi.dll?func=cslib.csinfo&objId=51783574&load=treecontent&lang=en&siteid=cseus&aktprim=0&objaction=csview&extranet=standard&viewreg=WW>

Siemens Oy. Siemens Oy:n verkkosivut 2013b. SCE\_EN\_020-060\_R1209\_Diagnose und Fehlersuche.doc

[http://www.automation.siemens.com/mcms/sce/en/advanced\\_training/training\\_material/TIA-Portal-Module/Documents/SCE\\_EN\\_020-060\\_R1209\\_Diagnose%20und%20Fehlersuche.do](http://www.automation.siemens.com/mcms/sce/en/advanced_training/training_material/TIA-Portal-Module/Documents/SCE_EN_020-060_R1209_Diagnose%20und%20Fehlersuche.do)

