



Beckhoffin PLC:n käyttöönoton ohjeistus opetusympäristössä

Simo Sinkkonen

Opinnäytetyö
Tammikuu 2015
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Konetekniikka
Kone- ja laiteautomaatio

SIMO SINKKONEN:

Beckhoffin PLC:n käyttöönoton ohjeistus opetusympäristössä
Työn tilaaja TAMK:n konelaboratorio Seppo Mäkelä

Opinnäytetyö 47 sivua, joista liitteitä 24 sivua
Tammikuu 2015

EtherCAT on Beckhoffin kehittämä kenttäväyläratkaisu, jossa hyödynnetään Ethernet-protokollan tekniikkaa automaatiojärjestelmän vaatimilla ehdoilla. Teollisuus-etherne-tissä yhdistyvät yksinkertainen kaapelointi sekä verkkorakenteiden helppo laajennetta-vuus ja teollisuusympäristöön suunnitellut tukevampitekoiset verkko-komponentit, jotka kestävät teollisuuden vaativampia olosuhteita ja kriteerejä.

Tämä opinnäytetyö tehtiin opetuskäyttöön Tampereen ammattikorkeakoulun Kone- ja laiteautomaatio -suuntautumisvaihtoehtoa varten. Aiheena oli laatia ohjeistus siitä, miten Beckhoffin PLC:n saa konfiguroitua EtherCAT-menetelmää ja Beckhoffin Windows-pohjaista TwinCAT-ohejelmistoa apunaan käyttäen. Työn alussa perehdytään automaation perusosa-alueisiin ja siellä hyödynnettäviin työn kannalta olennaisiin laitteisiin. Työn loppupuolella käsitellään Ethernet-verkkoteknologian peruskäsitteitä. Työn tuloksena syntyi ohje Beckhoffin PLC:n konfiguroimiseksi, joka on työn liitteenä.

Avainsanat: Beckhoff, TWINCat, PLC

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Mechanical and Production Engineering
Machine Automation

SIMO SINKKONEN:

Beckhoffs PLCs configuration in teaching environment

Thesis Supervisor Supervisor Laboratory Seppo Mäkelä
Bachelor's thesis 47 pages, appendices 24 pages
October 2014

EtherCat is fieldbus settlement developed by Beckhoff in which ethernet –protocoll's technic is utilized by automation system's terms. Industrial Ethernet combines simple cabling and easy expandibility of web structures and web components designed for factory environments so that they endure more demanding conditions and criteria.

This thesis was made for teaching purposes for Tampere University of Applied Sciences's Machine Automation –orientation. The topic was to compile guidelines about how to configurate Beckhoff's PLC with EtherCAT-method and with the help of Beckhoff's Windows-based TwinCAT-program. In the beginning of the thesis we are familiarized with basic principles of automation, programming, PLC's principle of operation and structure, and the basics of Ethernet web technology. The end of thesis deals with properties of EtherCAT's and Siemens' corresponding technologies. In the end there is attached a guideline for configuration of Beckhoff's PLC.

Keywords: Beckhoff, TWINCat, PLC

SISÄLLYS

1	Johdanto.....	5
2	Automaatio	6
3	Ohjelmoitava logiikka	7
3.1	Ohjelmoitavan logiikan peruskomponentit.....	7
3.2	Ohjelmointi	11
4	Teollisuus-ethernet	13
	Ethernetkehys	14
5	Kenttäväyläratkaisut	16
5.1	EtherCAT.....	16
5.2	Profinet.....	17
6	Ohjeen laatiminen.....	19
7	Yhteenveto.....	21
	LÄHTEET.....	22
	LIITTEET	23
	Liite 1. Beckhoffin PLC:n käyttöönotto-ohje.....	23

1 Johdanto

Tämän opinnäytetyön tavoitteena oli luoda ohjeistus Beckhoffin ohjelmoitavan logiikan käyttöönottamiseksi opetusympäristössä Tampereen ammattikorkeakoulun kone- ja laiteautomaatiotekniikan opiskelijoille. Lähtökohtana oli luoda jo aikaisemmin automaatiota ja ohjausjärjestelmiä opiskelleille selkeä ja helposti seurattava ohjeistus.

Beckhoffin PC-pohjainen TwinCAT -automaatio -ohjelmisto yhdessä heidän oman automaatiotekniikan kanssa soveltuvat erinomaisesti korkeaa dynamiikkaa vaativiin erilaisiin asemointisovelluksiin. TwinCAT -ohjelmisto on hyvin kehittynyt ja monipuolinen logiikkaohjelma, joka kuitenkin vaatii perehtymistä sen sujuvasti hallitsemiseksi.

Työn alussa (luvut 2-5) perehdytään automaation perusosa-alueisiin, ohjelmointiin, PLC:n toimintaperiaatteeseen ja rakenteeseen, sekä Ethernet-verkkoteknologian perusteisiin. Työn loppupuolella käsitellään Ether-CAT:n ja Siemensin vastaavan teknologian ominaisuuksia. Lopuksi liitteenä on työn tuloksena syntynyt ohjeistus Beckhoffin PLC:n konfiguroimiseksi.

2 Automaatio

Automaatio on jatkuvasti kasvava teollisuudesta lähtöisin oleva tekniikan osa-alue, joka näkyy tänä päivänä jo jokaisen suomalaisen arkipäiväisessä elämässä. Kovin moni ei ajattele lähikauppaan astuessaan, että automaattisesti avautuvat liukuovet ovat osa automaation tuottamaa helpotusta ja mukavuutta. Automaation nopea kehitys on muuttanut monia toimintamalleja ja sen kehityksen odotetaan vain kasvavan merkittävästi jatkossakin. Usein ajatellaan, että automaatio syrjäyttää ihmisiä työnteosta. Todellisuudessa kuitenkin automaatiota suunnitellaan aina ihmistä varten. Sen avulla pyritään esimerkiksi vapauttamaan ihminen yksitoikkoisesta tai ihmiselle vaarallisesta työtehtävästä. Automaatio sovelluksesta riippuen tuottaa tarkempaa tietoa ja pystyy työskentelemään vaativissa ja tarkkuutta vaativissa tehtävissä nykypäivänä jo huomattavasti tarkemmin kuin ihminen.

Aluksi automaation pohjana oli tehostaa työntekijän tekemälle työlle asetettuja tavoitteita, valvoa ympäristöä ja niiden pohjalta valita toimenpiteitä, jotka edistävät tuotannon tehokkuutta. Ongelmana alkuvaiheessa oli kuitenkin se mitä voitiin automatisoida ja mitä ei. Tämä ongelma määräsi ihmisen ja automaation välisen tehtäväjaon. (Wahlström 2006, 1-2)

Nykypäivän automaatiossa on mahdollisuuksia melkein pärajattomasti jatkuvan kehityksen ansiosta. Yhä kasvavassa määrin suuret automaatioalan yritykset, kuten esimerkiksi Siemens, Beckhoff ja ABB kehittävät tuotteitaan jatkuvasti monipuolisemmiksi ominaisuuksiltaan, jotta ne voisivat vastata suureen kysyntään ja teollisuuden asettamiin haasteisiin tuotannon tehostamiseksi. Toisaalta myös jatkuva kilpailu pakottaa yrityksiä panostamaan tuotekehitykseen ja näin automaatiotekniikan kehitys on ollut viimeisten vuosien aikana räjähdysmäistä.

3 Ohjelmoitava logiikka

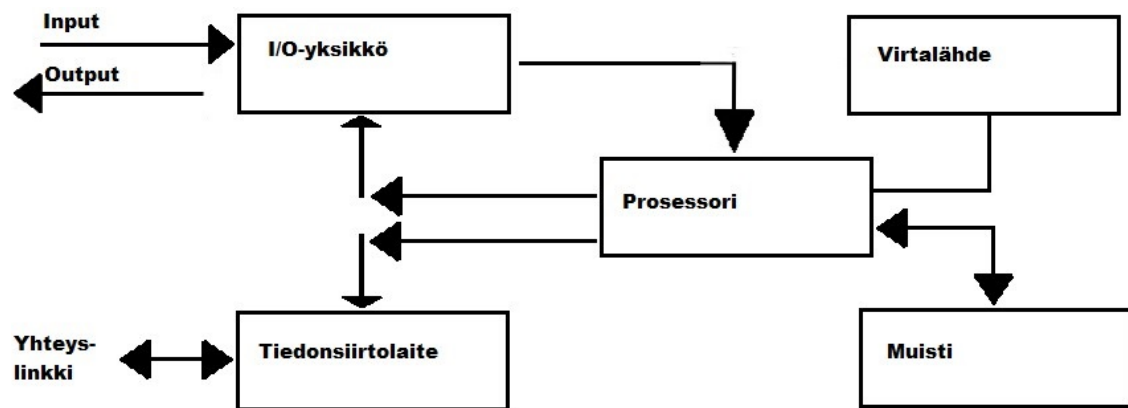
Ohjelmoitavat logiikat eli PLC:t (Programmable Logic Controller) suunniteltiin alun perin korvaamaan rele-pohjaiset ohjausjärjestelmät ja kiinteästi kytketyt puolijohdelogiikat. Tekniikan kehittyessä tästä järjestelmästä on kuitenkin tullut paljon monimutkaisempi ja tehokkaampi tapa ohjata erilaisia automatisoituja laitteita. Nykytekniikalla yhdellä keskusyksiköllä pystytään korvaamaan jopa satoja vanhanaikaisia releitä. Lisäksi ne ovat tehokkaampia ja toimintavarmuudeltaan parempia. Kaiken perustana ohjelmoitavissa logiikoissa on niiden kyky tarkastella sisääntulojen (input) tilaa ja vastata siihen logiikkaan ajetun ohjelman haluamalla tavalla, kuten käynnistää haluttu toiminto tai toimilaite. Näitä toimintoja tai toimilaitteita logiikka ohjaa ulostuloilla (output). Tätä tapahtumaa kutsutaan ohjauslogiikaksi. (Hughes 2005, 2–5)

Henkilökohtaisten tietokoneiden eli pc:n kehitys ja yleistyminen ovat johtaneet siihen, että yhä useammin Windows-pohjainen pc (personal computer) on korvannut PLC:n käytön säätö- ja ohjelmointisovellusten käytössä. Niiden räjähdysmäisesti kehittyneet ominaisuudet ja halventunut hintataso ovat suurimmat yksittäiset syyt tähän ilmiöön. Pc:n usein paljon tehokkaampi prosessori, suurempi muistikapasiteetti ja monet erilaiset Windows-käyttöjärjestelmässä toimivat ohjelmointi- ja valvontaohjelmat ovat usein helppokäyttöisiä. Jos aikaisemmin tarvittiin hyvin asiaan perehtynyt automaatioalan asiantuntija, niin nykyään pc:n avulla muidenkin on helpompi seurata, valvoa ja jopa tehdä muutoksia logiikan ohjelmistoon. Tähän on myös suuresti vaikuttanut se, että lähes jokaisella on kotitaloudessaan oma tietokone, jolloin niiden käyttö on tuttua. (Hughes 2005, 24–25)

3.1 Ohjelmoitavan logiikan peruskomponentit

Riippumatta PLC:n koosta, rakenteesta, hinnasta tai monimutkaisuudesta kaikki ne sisältävät samat komponentit ja toiminnalliset ominaisuudet. PLC tyypillisesti rakentuu seuraavista laitteiston osista: prosessori, sisään- ja ulostulolaitteisto, muisti, virtalähde sekä tiedonsiirtoportista tai -laitteesta. PLC tarvitsee myös ohjelmointilaitteen ja siihen sopivan ohjelmiston, jotka voidaan laskea kuuluvaksi peruskomponentteihin, vaikka niiden

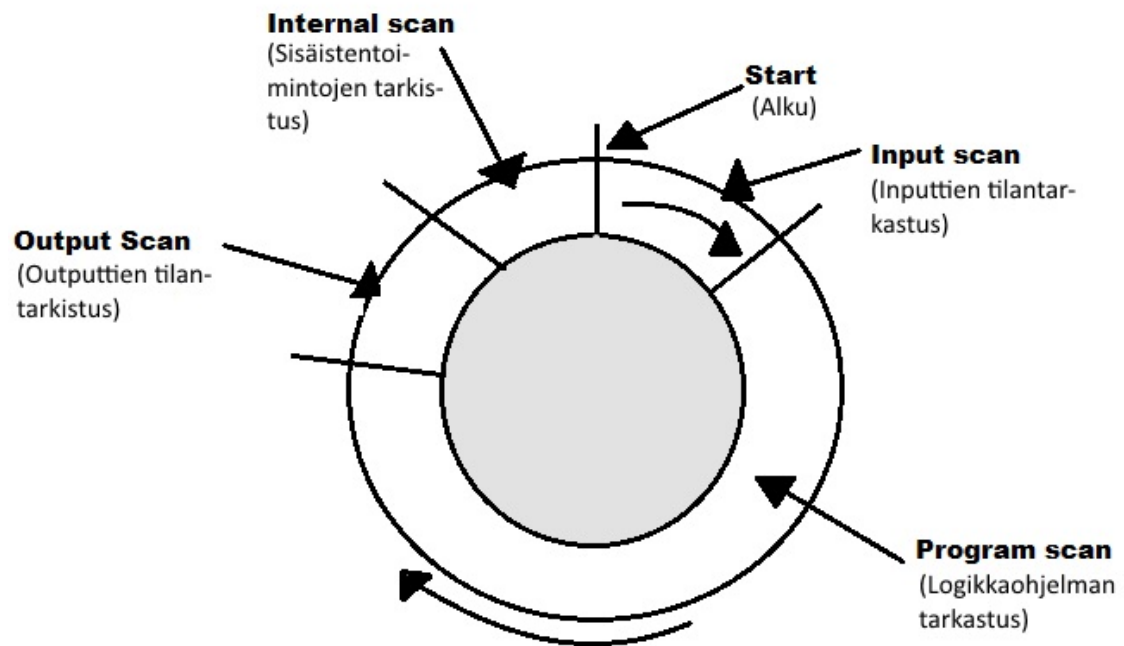
luonne eroaakin muusta laitteistosta. Ohjelmointilaitteena toimii usein vain siihen tarkoitukseen oleva tietokone. Kuviossa 1 esitetään yksinkertaisimmillaan PLC:n sisällä tapahtuvaa tiedonsiirto toimintaa lohkokaaavion avulla. Seuraavissa kappaleissa käsitellään PLC:n komponenttien toimintaa tarkemmin.



Kuvio 1: PLC:n sisäinen liikenne

Prosessori

Prosessori muodostuu yhdestä tai useammasta standardisoidusta tai kustomoidusta mikroprosessorista ja muista integroiduista mikropiireistä, jotka suorittavat loogisia, ohjaavia tai muistin toimintoja ohjelmitavassa logiikassa. Prosessori lukee inputteja ja suorittaa niistä saamansa tiedon perusteella sovellusohjelmalla, tämän työn tapauksessa Twin-CAT:n avulla tehty ohjelma, erilaisia laskelmia ja kontrolloi outputteja niiden mukaisesti. Prosessorin tehtävänä on kontrolloida toimintakiertoa eli ohjelmakiertoa. Toimintakierto rakentuu sarjasta toimintoja, jotka suoritetaan peräkkäin ja toistuvasti siten, että loppuun päästyään aloittaa prosessori kierron alusta. Kuvassa 2 on kuvattu tavallista toimintakiertoa prosessorissa. Toimintakierron kulku saattaa kuitenkin vaihdella eri valmistajan automaatio-ohjelmistosta riippuen. (Hughes 2005, 4–6)



Kuvio 2: Toimintakierto

Toimintakierros alkaa input scan (inputtien tarkistus) -vaiheesta, jossa keskusyksikkö etsii mahdollisia kytkettyjä laitteita input-porteista sekä tarkastelee niiden tilaa ja toimintaa. Kun niiden tilat on käyty läpi, tallennetaan ne inputtien väliaikaismuistiin tai niille erillisesti tarkoitettuun muistipaikkaan. Seuraavaksi program scan (logiikkaohjelman tarkistus) -vaiheessa prosessori lukee sille tallennetusta logiikkaohjelmasta määritelmät ja ehdot käyttäen hyväksi väliaikaismuistissa olevia tietoja inputtien tilasta ja toiminnasta. Tämän jälkeen se määrittää mitkä outputeista mahdollisesti aktivoidaan eli niiden läpi syötetään jännite ja tallentaa tämän tiedon outputtien väliaikaismuistiin. Output scan (outputtien tarkistus) -vaiheessa PLC aktivoi tai deaktivoi niitä ulostuloja joiden tiedot tallennettiin edellisessä vaiheessa. Viimeisessä vaiheessa toimintakiertoa, internal scan (sisäisten toimintojen tarkistus) -vaiheessa, prosessori suorittaa erilaisia ylläpitoon liittyviä toimintoja, kuten yhteyksiä muihin siihen mahdollisesti kytkettyihin logiikkoihin ja sen ohjelmointilaitteeseen.

Toimintakierto kestää tavallisesti 1-25 millisekuntia. Kiertoon kuluva aika riippuu kuitenkin suuresti käyttäjän tekemän ohjelman monimutkaisuudesta. Mitä monimutkaisempi ja suurempi ohjelma on, sitä kauemmin program scan -vaihe kestää, jopa 250 millisekuntia. Muut toimintakierron vaiheet ovat yleensä lyhyitä verrattuna program scan -vaiheeseen. Kun kierto on mennyt loppuun, alkaa heti uusi kierto. (Hughes 2005, 4–6)

Input- ja outputyksikkö

Input- ja outputyksiköstä käytetään yleisesti nimeä I/O-yksikkö, joka tulee yksinkertaisesti sanojen input ja output ensimmäisistä kirjaimista. I/O – järjestelmästä käytetään myös nimitystä tulot ja lähdöt. I/O-tieto on signaali, joka saadaan niihin kytketyistä antureista tai muista järjestelmään kytketyistä valvontalaitteistoista tai niiden osista. (PLC I/O Devices, 2013)

I/O-yksikkö koostuu useammista tulo- ja lähtöporteista. Tuloporttien avulla logiikka saa tietoa järjestelmän tilasta ja lähtöporttien kautta se voi ohjata järjestelmää. Tulo- ja lähtöporttien signaalit voivat olla joko digitaalisia tai analogisia. Digitaalisten signaalien ilmaisemiseksi käytetään usein jännitettä tai virtaa. Alun perin ohjelmoitavissa logiikoissa oli vain digitaalisia liitäntöjä. Tällä tavalla tietty suureen alue tulkitaan 0-tilaksi ja toinen 1-tilaksi eli ne ovat joko päällä tai pois. Jos esimerkiksi logiikka käyttää 24 V jännitettä, niin alle 2 V jännite tulkitaan nolaksi ja yli 22 V jännite ykköseksi. Analogiset signaalit välittävät kaikki toiminta-alueen arvot ääripäiden väliltä. Mitattavan signaalin arvo voi olla mitä tahansa valitulla mitta-alueella, analogiapostin tyypistä riippuen esimerkiksi 4–20 mA, +10 V tai +5 V. On myös olemassa erilaisia vastuskortteja, eli joihin voidaan kytkeä suoraan anturi, joka antaa signaalia resistanssi muodossa. Näin tuloportti osaa tulkita tulevaa signaalia suoraan, eikä tarvita erillistä muunninta muuntamaan signaali ensin virraksi tai jännitteeksi. (Hughes 2005, 6–11)

Muisti

Muistia käytetään varastoimaan PLC:n käyttöjärjestelmää. Siihen tallennettu tieto määrittää kuinka sisään- ja ulostulot toimivat. Muisteja on monia eri tyyppisiä, mutta ne voidaan aina jakaa kahteen eri ryhmään: haihtuvaan ja pysyvään. Näiden kahden erona on muistin säilyminen, kun sähköä ei ole kytketty. Haihtuvaan muistiin tallennetut ohjelmat ja tiedot pyyhkiytyvät sähkökatkon aikana. Tämän takia haihtuvaa tyyppiä olevat muistit on usein varusteltu varapatterilla tai jollain muulla varajärjestelmällä. Pysyvässä muistissa ei sähkökatkon aikana tapahdu tyhjenemistä ja se ei siten tarvitse varajärjestelmää. PLC:n muisti voidaan jakaa myös kahteen alueeseen. Toiseen alueeseen käyttäjä pääsee omalta päätteeltään tekemään muutoksia ohjelmiin, laskureihin ja tulkitsemaan esimerkiksi vikatilaa sattua kertynyttä dataa. Toisen alueen ohjelmoinnista puolestaan vastaa

kunkin PLC:n valmistaja. Tämä muistin osa ohjaa ja käyttää käyttöjärjestelmää, sekä erilaisia toimintoja, kuten esimerkiksi sisääntulolinjojen datan tarkkailu ja ulostulojen kontrollointi. (Hughes 2005, 12)

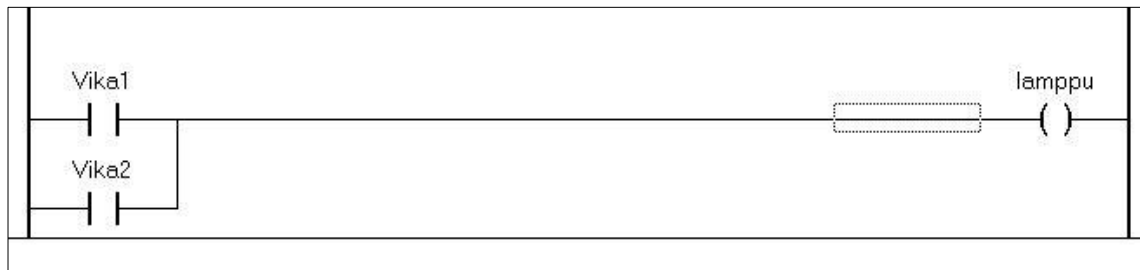
Virtalähde ja Tiedonsiirtolaitteisto

Virtalähde voi olla integroituna prosessorin, muistin sekä sisään- ja ulostuloyksikön kanssa yhdeksi kokonaisuudeksi tai se voi olla erillisenä yksikkönä, joka on yhdistetty kaapeloinnilla muihin yksikköihin. Sen tehtävänä on muuntaa vaihtovirtaa tasavirraksi. Kun automaatiojärjestelmässä on useita sisään- ja ulostuloyksikköjä, tarvitaan usein lisävirtalähde täyttämään kasvanutta tehon kulutusta. Tiedonsiirtolaitteiston tai -portin päätehtävänä on kommunikoida ohjelmointiin ja valvontaan käytettävän tietokoneen kanssa, jotta sillä voidaan muuttaa ja valvoa automaatiojärjestelmän toimintoja. Lisäksi mikäli ohjausjärjestelmässä on kaksi porttia tai enemmän, käytetään lisänä olevia portteja muodostamaan yhteys usean eri käyttöjärjestelmän välille, jotta ohjausjärjestelmää voidaan etäkäyttää ja valvoa yhtäaikaisesti keskitetysti muiden järjestelmien kanssa. Lisäksi näin mahdollistetaan kaapelivian sattuessa asennustapa, jossa järjestelmät ovat linkitetty piireittäin ja näin yhden kaapelin katkeaminen ei vielä kaada koko automaatiojärjestelmää ja näin aiheuta käyttökatkoa. (Hughes, 2005, 13–14)

3.2 Ohjelmointi

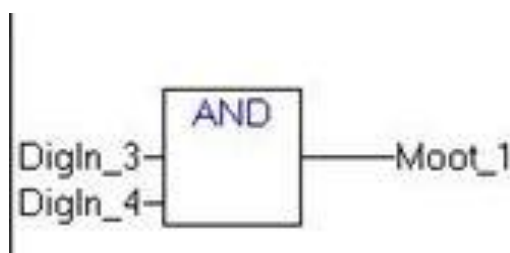
Ohjelmoitavat logiikat tarvitsevat toimiakseen logiikkaohjelman, joka niihin kirjoitetaan erityisesti sitä varten suunnitetuilla ja kehitetyillä ohjelmilla. Tässä työssä käytettävä ohjelmalla on Beckhoffin kehittämä TwinCAT. Jokaisella komponenttivalmistajalla on oma ohjelmistonsa logiikkaohjelman tekemiseen ja sen edelleen automaatiojärjestelmään siirtämiseen. Ohjelmaa käytetään tavallisella tietokoneella. Useat valmistajat kuitenkin tarjoavat kannettavia tietokoneita osana automaatiojärjestelmää, kun sellaisen hankintaa suunnitellaan ja toteutetaan teollisuuden yrityksiin. Tämä kannettavatietokone on tarkoitettu pääasiassa logiikkaohjelman toiminnan seurantaan tai päivittämiseen. Ohjelmointiin on kehitetty erilaisia ohjelmointitapoja. Tikapuuohjelmoinnin tapa yleistyi logiikoiden kehityksen alussa. Sen idea lainattiin suoraan relekaavioista, jota muokattiin hieman sen käytön ja tulkitsemisen helpottamiseksi. Tämän tyyppisessä kaaviossa on ohjelmoitavat releet kytketty kahden virtakiskon väliin ja ohjelma etenee vasemmalta oikealle. Näiden

kahden kiskon väliin sijoitetut symbolit kuvaavat releen kärkiä, jotka ovat yksinkertaisimmillaan kiinni tai auki. Jonossa ne muodostavat sarjan ja mikäli virta pääsee kulkemaan vasemmalta kiskolta oikealle, suorittaa logiikka määrätyn toiminnon aktivoimalla ulostulon. Kuviossa 3 on esitetty tikapuutavan ohjelmointia yksinkertaisella tavalla. Vika1 ja Vika2 kuvaavat releitä ja Lamppu toimilaitetta, yksinkertaisimmillaan esimerkiksi hehkulamppua. (Hughes, 2005 14–19; Relay-ladder-logic, 2013)



Kuvio 3: Tikapuukaavio

Myöhemmin, kun teollisuudessa käytettävä laitteisto on kehittynyt, on myös tikapuu-kaavion avulla tapahtuvan ohjelmoinnin rinnalle kehitetty lukuisia muita ohjelmointikieliä. Monista vaihtoehdoista yleisimmistä on kehittynyt FBD-malli (Function Block Diagram), jota on kutsuttu myös graafiseksi malliksi. Nimitys johtuu sen tavasta esittää ohjelma toimintalohkomallina, jossa kytkennät lohkojen välillä suoritetaan automaattisesti tai ne niin sanotusti piirretään eli yhdistetään viivalla, joka toimii vastaavasti kuin sähköjohto lampun ja virtalähteen välillä. Sen käyttö perustuu standardoituihin lohkoelementteihin, joita käyttäjä voi vapaasti valita ohjelman FBD-kirjastosta. Näitä lohkoja pystyy myös yhdistelemään ja muokkaamaan niissä sijaitsevien inputtien ja outputtien määrää. Kuviossa 4 on esitetty yksinkertainen JA-lohko, jossa on kaksi input- (DigIn_3 ja DigIn_4) ja yksi output-liitäntä (Moot_1). Tämän ohjelmointitavan monipuolisuus on tehnyt eri ohjelmavalmistajien kirjastoon lähes rajattoman määrän erilaisia toimintalohkoja joiden tarkoitus vaihtelee perinteisestä JA-lohkosta monimutkaisimpiin laskureihin ja niiden yhdistelmiin. (Hughes 2005, 14 – 20)



Kuvio 4: FBD-malli

4 Teollisuus-ethernet

Ethernet on verkkoteknologia, johon internet perustuu. Se tarjoaa monia mahdollisuuksia maailmanlaajuiseen verkostoitumiseen. Nimitys ethernet tulee alun perin tiedonsiirtoon käytetystä koaksiaalikaapelista, jota kutsuttiin eetteriksi (ether). Se kehitettiin jo 1970-luvun puolivälissä Yhdysvalloissa, kun Xerox esitteli konseptin ethernetistä, jossa yli 100 päälaitetta pystyi kommunikoimaan verkossa silloin suurella 3 Mb/s tiedonsiirtonopeudella keskenään ilman aiempaa tietämystä toisistaan. Ethernet on verkkoteknologia, joka ei ota kantaa sen päällä toimiviin protokolleihin, vaan se on vain tiedonsiirtoyhteys, jossa tieto kulkee. Nykypäivänä teollisuudessa toimistoympäristöjen intranetin sekä internetin tarjoamia mahdollisuuksia hyödynnetään monella tapaa myös tuotannossa ja prosessiautomaatiossa. Tästä esimerkkejä ovat muun muassa käyttöasteen, työtilanteen ja -järjestyksen seuraaminen sekä esimerkiksi huollon tarpeesta ilmoittaminen. Edellä mainittujen hyötyjen ansiosta tekniikkaa on kehitetty ja se on yleistynyt viime vuosina. Teollisuus-Ethernetissä yhdistyvät yksinkertainen kaapelointi sekä verkkorakenteiden helppo laajennettavuus ja teollisuusympäristöön suunnitellut tukevampitekoiset verkkokomponentit. Verkkokomponentit ovat järeämpiä, jotta ne kestävät teollisuuden vaativampia olosuhteita. Näitä ovat esimerkiksi vaativammat ympäristön ja tuotannon olosuhteet, kuten pöly ja lika, vesitiiveys sekä lämpötilanvaihtelut. Ethernetin käyttö automaatioprosessin ohjauksessa asettaa myös protokollalle lisävaateita toimistossa käytettäviin ethernetiin verrattuna. Toisin kuin toimistokäytössä, automaatiiossa tiedon pitää kulkea deterministisesti, eli ennustavasti, ja sen pitää olla mahdollisimman reaaliaikaista. Kun toimistomaailmassa riittää, että tietopaketti on perillä sekunneissa, on automaation aikavaade millisekunteja. Tähän ei yksin riitä se, että tiedonsiirtonopeus nostetaan mahdollisimman suureksi. Sen lisäksi on myös tärkeää, että dataliikennettä kontrolloidaan tavalla, joka pystyy takaamaan aikaviiveen datan siirrossa riittävän pienenä. Teollisuus-ethernet onkin yhä tärkeämmällä sijalla nykyaikaisessa teollisuusautomaatiossa. Ammattilaisen suunnittelema teollisuus-ethernet mahdollistaa tiedon kulkemisen automaatiiossa mukana olevien laitteiden ja järjestelmien välillä. Nykyaikainen teollisuus-ethernet myös pystyy vastaamaan kaikkiin teollisuusautomaation asettamiin haasteisiin. (Kallionpää 2010, 58; Siemen Teollisuus-ethernet 2014)

Kenttäväylät ja niiden suorituskyky ei ole enää kuumin puheenaihe kun puhutaan teollisuuden Ethernet-sovelluksista, mutta täytyy muistaa, että ne ovat vain yksi sovelluskohde. Kenttäväylät alkavat olla viimeisiä kohteita, joissa Ethernetin käyttö ei ole vielä standardikäytäntö, mutta muussa teollisuuden tiedonsiirrossa kenttätasolla Ethernet on jo vakiinnuttanut asemansa. Ethernet on jo ulotettu ohjaustasolle ja myös kaikenlainen varsinainen säätöpiirin ulkopuolella tapahtuva tiedonsiirto, kuten antureilta saatava mittausdata tuotannonhallintaa varten on jo yleensä toteutettu Ethernet tekniikalla ja jopa mahdollisesti WLAN-laitteita hyödyntäen. (Silvola 2006, 25–26)

Teollisuudessa käytettävää WLANia kutsutaan IWLAN:ksi (Industrial WLAN), teollisuuden langaton lähiverkko. Kupari ja valokuitukaapelien sijaan langaton tiedonsiirto tapahtuu radioaalloilla. WLAN-standardissa ei ole tarvittavia määräyksiä teollisuusympäristön vaatimuksiin. Tämä johtuu osaltaan siitä, että tuotanto- ja automaatiotasolla on lukuisia sähkömagneettisten häiriöiden lähteitä. Toisaalta syynä ovat automaation tiukat vaatimukset reaaliaikaisuudelle ja tiedonsiirron deterministisyydelle. IWLAN:iin onkin jouduttu tekemään parannuksia, jotta se täyttäisi automaation tiukat vaatimukset. Nykyään teollisuuden langaton tiedonsiirto käyttää useita erilaisia langattomia verkkoja. Näitä voivat olla esimerkiksi WLAN-, GSM- tai GPRS-yhteys. Niiden käyttö on mahdollista, koska ne toimivat eri taajuuksalueilla. Myös langattoman verkon tietoturvariskit on otettu huomioon nykyaikaisissa langattomissa yhteyksissä. Lisäksi käytettäviä komponentteja on muokattu järeämmiksi, jotta ne kestäisivät teollisuuden vaatimat olosuhteet. (Siemens Teollisuus WLAN, 2014)

Ethernetkehys

Nykyisin Ethernet-nimellä ei enää viitata yhteen standardiin, vaan Ethernet on enemminkin usean lähiverkkostandardin sarja. Alkuperäisen standardin ympärille on määritelty ja määritellään yhä lisää erilaisia täydentäviä standardeja. Näiden standardien myötä ethernet hyödyntää uusia teknologioita, tukien muun muassa uusia, entistäkin suurempia data-nopeuksia sekä vaihtoehtoisia fyysisiä kaapelointeja. (Silvola 2006, 20)

Ethernet-verkot ovat pakettikytkennäisiä. Tällä tarkoitetaan sitä, että verkossa liikkuva data lähetetään itsenäisissä paketeissa. Ethernetissä näitä paketteja kutsutaan kehyksiksi. Käytännössä kehykset ovat jono tiettyssä järjestyksessä olevia datapaketteja eli dataokteetteja. Ethernet-kehys rakentuu kuudesta pääosasta: tahdistusosa, kohdeosoite, lähdeosoite,

tyyppi, data ja tarkistussumma eli CRC (Cyclic Redundancy Check). Tahdistusosalla kehyksen siirto tahdistetaan muiden toimijoiden kanssa samaan tahtiin. Tätä edeltää pieni tauko, jonka aikana mahdollistetaan muiden toimijoiden kehyksen lähetys. Kohde- ja lähteosoitteet sisältävät toimilaitteiden fyysiset MAC-osoitteet eli laitteen verkkosovittimen yksilöllinen osoite. Tyypikentässä ilmoitetaan mille protokollalle kehyksen data ohjataan, esimerkiksi ohjataan se IEEE 802.3 standardille vai johonkin muulle standardipohjalle. (Silvola 2006, 20–24)

5 Kenttäväyläratkaisut

Automaatiojärjestelmien kenttäväylässä eli tiedonsiirtojärjestelyissä on monia erilaisia ja toisistaan poikkeavia ratkaisuja. Automaatiojärjestelmät ovat käyttäneet sisäiseen tiedonsiirtoonsa joko valmistajakohtaisia tai standardisoituja väyliä kuten esimerkiksi PROFIBUS:ta, ASiA ja CANia. Näiden väylien tiedonsiirtonopeudet ovat tyypillisesti merkittävästi pienempiä kuin tietokoneverkkojen nopeudet. Automaatiioväylien erityisenä vaatimuksena on riittävän pieni ja tunnettu viive. Ohjausten ja säätöjen täsmällisen toiminnan edellytyksenä on mittaus- ja ohjaustietojen perillemeno juuri tietyllä hetkellä. (Kallionpää 2010, 10)

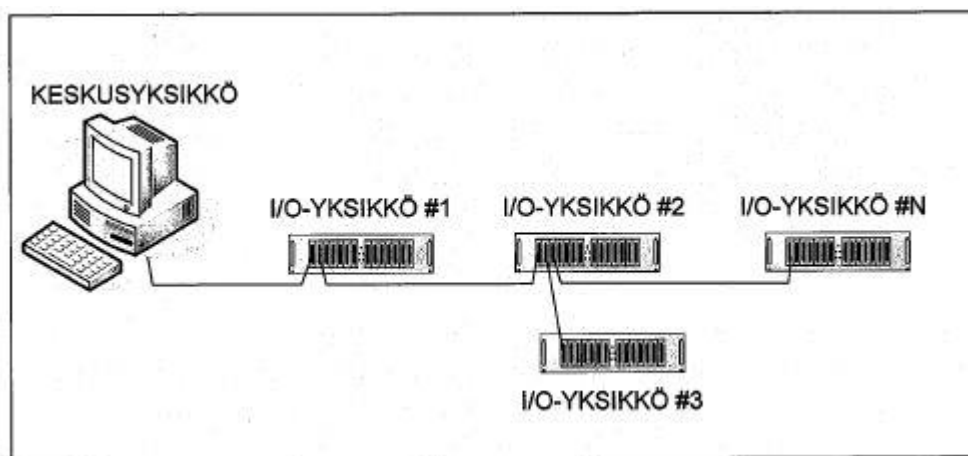
Tässä työssä tarkastelun kohteena olivat vain Beckhoffin EtherCAT ja Siemensin Profinet ethernet-tyyppiseen tiedonsiirtoon perustuvat kenttäväyläratkaisut. Niiden eroja ja ominaisuuksia käsitellään seuraavissa kappaleissa tarkemmin.

5.1 EtherCAT

EtherCAT (Ethernet for Control Automation Technology) on alun perin Beckhoff Automation GmbH:n (www.beckhoff.com) kehittämä kenttäväyläprotokolla ja sille on myöhemmin perustettu oma EtherCAT Technology Group:n (<http://www.ethercat.org>) organisaatio, jonka tehtävänä on yhdessä kehittää ja edesauttaa tämän tekniikan yleistymistä. EtherCAT:ssa automaatiojärjestelmän keskusyksikkö keskustelelee liitäntäyksiköiden (I/O-moduuli) kanssa käyttämällä Ethernet-kehysrakennetta.

EtherCAT:n tavassa ohjausjärjestelmän keskusyksikkö on MASTER-roolissa ja I/O-moduuli toimii tiedonsiirron kannalta katsottuna SLAVE-roolissa. Master-yksikkö määrää kaiken tiedonsiirron ja kehys kulkee kaikkien slave-yksiköiden kautta. Viimeinen slave-yksikkö lähettää kehys takaisin ja kun se saapuu master-yksikölle, niin master-yksikkö voi olla varma, että sanoma on saavuttanut kaikki siihen kytketyt yksiköt. Yksittäisen slave-yksikön aiheuttama viive kehys käsittelyssä on vain muutama nanosekunti ja tänä aikana slave-yksikkö lukee sille tarkoitetun tiedon ja kirjoittaa oman tietonsa kehykseen. (Kallionpää 2010, 21, 58)

Beckhoffin ja EtherCATn kehittämää tiedonsiirtokehystä voisi verrata junaan, jossa jokaiselle I/O-moduulin tiedoille on varattu oma vaununsa. Näin jokainen I/O-moduuli on ikään kuin oma asemansa, missä tietoa, eli tavuja ja bittejä, voisi verrata matkustajiin. Juna ei kuitenkaan pysähdy jokaisella asemalla vaan se suorittaa niin sanottujen matkustajien vaihdon lennosta ja säästää näin arvokasta aikaa ja pienentää toimintaviivettä. Junan saavutettua viimeinen I/O-moduulin, palaa se takaisin keskusyksikölle. Paluu matkalla ei tietokehystä enää muuteta eikä sitä myöskään tarkkailla. Keskusyksikkö voi myös lähettää tietokehyksiä samaan aikaan kun edelliset kehykset ovat paluumatkalla. EtherCATn käyttämää kenttäväylärakennetta on havainnollistettu kuviossa 5.

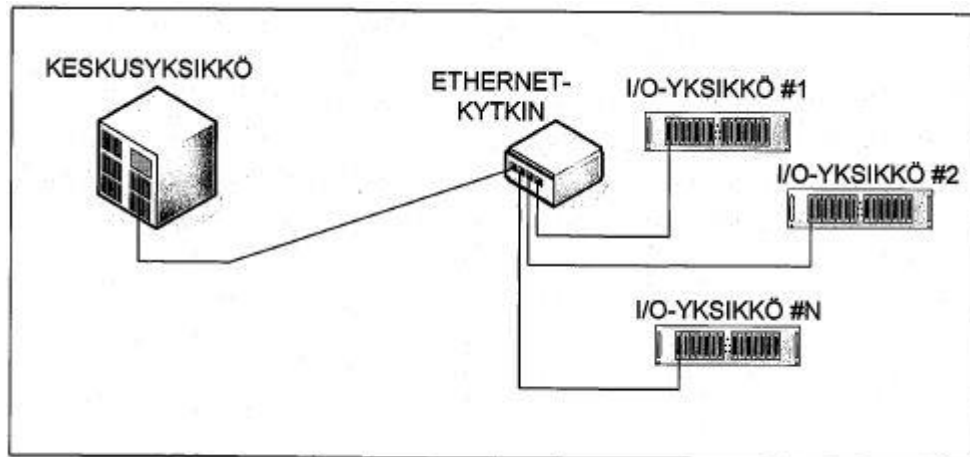


Kuvio 5: EtherCAT-järjestelmän topologinen rakenne

5.2 Profinet

Profinetin kehittäjänä ovat toimineet Siemens (www.siemens.fi) ja Profibus User Organization (www.profibus.com). Se poikkeaa toiminnaltaan EtherCATiin verrattuna siten, että siinä I/O-yksiköt eivät liikennöi keskenään vaan kaikki liikenne tapahtuu keskusyksikön ja I/O-yksiköisen välillä. Jokaisella I/O-yksiköllä on siis oma osoitteensa, joille keskusyksikkö lähettää oman tiedonsiirtokehyksensä. Tätä toimintamallia on havainnollistettu kuviossa 6. Myös merkittävä ero on miten tiedonsiirto rakentuu keskusyksikön ja I/O-yksikön välillä. Profinetissä on myös jaettu reaaliaikaisuus kahteen ryhmään. Reaaliaikaiseen (Real Time, RT) ja isokroonisesti reaaliaikaiseen (Isochronous Real-Time, IRT). Profinet käyttää optimoitua IRT-kanavaa sille tiedolle mikä on priorisoitu viiveeksi korkeintaan 1 millisekunti, kuten liikkeenohjaussovelluksiin ja muihin hyvin pieniä reaktioaikoja vaativiin sovelluksiin. Näin se pystyy vastaamaan myös laitteiden korkeisiin

vaatimuksiin kenttäväylä tasolla. Lisäksi Profinetissä RT-luokittelu on jaettu kolmeen eri ryhmään niiden reaaliaikaisuuskriteerien perusteella. Nämä ryhmät ovat yksinkertaisesti RT1-3. Keskusyksikön ja I/O-laitteiden väliseen liikenteeseen, niin sanottuun alemman prioriteetin datan siirtoon, jonka aika viiveeksi on määritelty 1-10 millisekuntia, Profinet käyttää RT-kanavaa. (Kallionpää 2010, 21–22; Siemens Profinet)



Kuvio 6: Profinet-järjestelmän topologinen rakenne

6 Ohjeen laatiminen

Ennen ohjeen työstämistä pyrin tutustumaan jo olemassa oleviin ohjeisiin PLC:n konfiguroimiseen liittyen. Beckhoffin Twin CAT System Manager ja PLC Project Controller vaativat perehtymistä, ennen kuin niiden käyttö on sujuvaa. Ohjeen laajuutta ei rajoitettu sivumääriin työn alkuvaiheilla. Sovelsinkin vapaasti omaa harkintaa kuinka tarkasti asiat ohjeessa esitetään. Pyrin kuitenkin siihen, että ohje olisi selkeä ja havainnollistava. Tähän vaikutti suuresti kuvien käytön määrä. Monissa lukemissani ohjeissa kuvia on ollut liian vähän tai niiden havainnollistava vaikutus on epämääräinen ja epäselvä. Kävimme asiasta työnohjaajaan Seppo Mäkelän kanssa pohdintaa ja tulimme siihen lopputulokseen, että ohjeen on parempi olla liian yksityiskohtainen kuin suppea.

Monessa lukemassani ohjeessa määriteltiin ensin yhteys käytettävään logiikkaan ja samalla skannattiin jo olemassa olevat input- ja outputkortit, ja vasta tämän jälkeen aloitettiin itse ohjelman mallintaminen. Valmiissa ohjeissa oli myös usein kuvien lisäksi vain mahdollisimman tyypistetyillä lauseilla kerrottu kuinka mikäkin asia tehdään. Joissakin ohjeissa myös hypittiin asiasta toiseen ja niiden seuraaminen oli hankalaa. Lähtökohtana omassa ohjeessani pyrin havainnollistamaan tekstin ja kuvien mitkä eri vaiheissa olisi tarkoitus tehdä ja kuinka se myös helposti onnistuu. Ohjeen seurattavuus ja selkeä eteneminen olisi myös erittäin tärkeää. Tulin myös siihen lopputulokseen, että yhteyden muodostaminen ei ole tarpeellista ensimmäiseksi, kun aloittaa työskentelyn Beckhoffin PLC:n kanssa. Automaatiosuunnittelun kannalta on toki tärkeää alkuvaiheessa tietää millaista in- ja output tietoa tarvitsee käsitellä ja millaisen PLC-yksikön kanssa toimitaan, jotta osaa valita oikeanlaisen ohjelmapohjan ja määrittää muuttujatiedot jo valmiiksi oikein, esimerkiksi analogia- ja digitaalityyppeihin liittyen.

Tekemässäni ohjeessa projekti aloitetaan sillä perusteella mistä on luontaisempaa ja selkeää lähteä automaatio-ohjelmaa rakentamaan. Ensimmäiseksi luodaan PLC-projekti. Siinä määritellään halutut toiminnot, ehdot, muuttujat ja lopuksi se käännetään oikeaan muotoon logiikan ymmärrettäväksi. Kun PLC-projekti on saatu valmiiksi, siirrytään TwinCAT System Managerin puolelle, jossa linkitetään projektiin luotuja muuttujia fyysisiin tuloihin ja lähtöihin. Tämä tapahtuma poikkeaa jokseenkin paljon esimerkiksi Siemensin ohjelmien tavasta, jossa muuttujien osoitteet määritellään jo ohjelmanluonti vaiheessa. Beckhoffin ohjelmapaktilla on yksinkertaista ja sujuvaa luoda erilaisia toimintoja, vaikka alkuun se tuntuukin kankealta ja monimutkaiselta. Ohjelma on luotu älykkääksi ja

siinä on monia automatisoituja toimintoja, jotka helpottavat käytössä oleviin laitteisiin yhteyden muodostamista ja niihin liitettyjen I/O-korttien löytämistä. Lisäksi varsinkin FBD-ohjelmoinnin avuksi on luotu erilaisia pikanäppäinyhdistelmiä, joiden avulla ohjelmointia pystyy nopeuttamaan ja helpottamaan merkittävästi. Näiden pikanäppäimien hyödyntämisen oppii jo parin tunnin käyttökokemuksella. Tämän opinnäytetyön tuloksena syntynyt ohje Beckhoffin PLC:n konfiguroimiseksi löytyy liitteenä työn lopusta.

7 Yhteenveto

Tämän opinnäytetyön lopputuloksena ja tavoitteena oli mahdollistaa Beckhoffin PLC:n käyttöönnotto eli konfiguraatio Tampereen ammattikorkeakoulun opetusympäristössä EtherCAT-mentelmää käyttäen. Lisäksi työssä oli tarkoitus antaa perustietoa ohjelmoimiseen, teknologiaan ja automaation peruseriaatteisiin liittyen. Ohje oli kohtalaisen helppo tehdä, vaikka joitakin ongelmia myös sitä tehdessä ilmeni. Ennalta ajateltuna Ethernetiä hyödyntävä yhteyden muodotustapa pitäisi olla helpohkoa ja yksinkertaista. Työn edessä kuitenkin tuli selväksi, että juuri Ethernetin kanssa toimiessa olisi myös verkkoteknologian ja sen toimintaperiaatteen tunteminen suureksi hyödyksi ongelmatilanteita ratkaistaessa.

Lopputuloksena syntynyt ohje on kattava, yksityiskohtainen ja antaa mahdollisuuden tulevaisuudessa teettää erilaisia automaatioharjoituksia opiskelijoille. Ohjeen avulla vältetään aikaa vievä perehtyminen laitteen konfirmoimiseen ja ohjeen avulla mahdollistetaan näin ripeämmän työskentelyn itse harjoitustehtävän kanssa. Ohje tehtiin osaksi melko pikkutarkaksi. Tämä mahdollistaa jossain määrin myös kokemattoman käyttäjän harjoitusten tekemisen ja aloittamisen.

Jatkossa ohjeen rakennetta ja sisältöä tulisi kehittää, kun sen hyödyistä ja ongelmista on kattavaa palautetta. Ohje on kuitenkin oma näkemykseni siitä miten Beckhoffin TwinCAT -ohjelmaan pääsee perehtymään ilman erillistä kurssia ohjelman käytöstä. Yksittäisen palautteen pohjalta voisi joitain muutoksia tehdä, mutta kun ohjetta on käyttänyt tarpeeksi moni opiskelija, tulisi paremmin esille ne asiat mitkä aiheuttavat epäselvyyttä tai kehitystä.

LÄHTEET

Hughes, T. 2005. Programmable Controller

Kallionpää, R. 2012. Automaatiojärjestelmien tietoliikenne. Tampereen ammattikorkeakoulu. Ylemmänasteen opinnäytetyö.

Silvola, R. 2006. Reaaliaikaiset teollisuus-Ethernet -ratkaisut automaatiojärjestelmissä. Tampereen Teknillinen Yliopisto, Automaatiotekniikan osasto. Diplomityö.

Wahlström, B. 2006. Automaatio ja ihminen VTT, Suomen automaatioseura, Control Systems konferenssi 2006, Tampere. <http://www.automatioseura.fi/>

Siemens 2014 a, Profinet. Luettu 10.9.2014 http://www.siemens.fi/pool/products/industry/iadt_is/tuotteet/automaatiotekniikka/teollinen_tiedonsiirto/profinet/br_profinet.pdf

Siemens 2014 b, Teollisuus-ethernet. Luettu 5.3.2014. http://www.siemens.fi/fi/industry/teollisuuden_tuotteet_ja_ratkaisut/tuotesivut/automaatiotekniikka/teollinen_tiedonsiirto_esim_profinet/teollisuus_ethernet.htm

PLC-IO-Devices 2014. Luettu 10.3.2014. <http://www.plctutor.com/plc-io-devices.html>

LIITTEET

Liite 1. Beckhoffin PLC:n käyttöönotto-ohje