



Tuomas Rahkola

MODBUS-PROFIBUS-RAJAPINTA

MODBUS–PROFIBUS-RAJAPINTA

Tuomas Rahkola
Opinnäytetyö
Kevät 2013
Automaatiotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Automaatiotekniikan koulutusohjelma

Tekijä: Tuomas Rahkola
Opinnäytetyön nimi: Modbus–Profibus-rajapinta
Työn ohjaaja: Tero Hietanen
Työn valmistumislukukausi ja -vuosi: Kevät 2013
Sivumäärä: 50 + 8 liitettä

Työ tehtiin Oulun Fidelix Oy:lle. Lähtökohtana opinnäytetyölle oli Stora Enson Uruguayihin rakennuttamalle tehtaalle toimitettavien tuloilmakoneiden tilaajan vaatimus liittynästä, joka tuli toteuttaa Fidelixin alakeskuksiin Profibus-väylän kautta. Työn tarkoituksena oli selvittää, voitaisiinko valitulla yhdyskäytävälaitteella toteuttaa tietojen siirto Modbus–Profibus-rajapinnan yli tehtaalla käytettävään Honeywellin automaatiojärjestelmään. Mikäli valittu yhdyskäytävä osoittautuisi tarkoitukseen sopivaksi, tuli tilaajalle tuottaa englanninkieliset ohjeet laitteiden käyttöönottoa varten.

Koska laitteistoa ei vielä työn aloitusvaiheessa ollut määritelty tai tilattu, aloitettiin työ paneutumalla väylätekniikoiden teoriaan sekä koululta löytyviin laitteistoihin, joita myöhemmässä vaiheessa tultaisiin vielä tarvitsemaan. Kun käytettävä yhdyskäytävä myöhemmässä vaiheessa oli päätetty ja se saapui, päästiin tutustumaan sen käyttöohjeisiin ja mukana tuleviin ohjelmistoihin sekä liittämään sitä tekniikan yksikön automaatiolaboratorioon toteutettuun testauslaitteistoon.

Työn tuloksena syntyi englanninkielinen ohjekirja tilaajalle ja saatiin varmuus siitä, että liittynä on toteutettavissa valitulla yhdyskäytävälaitteella. Selvitystyö onnistui ja tehdyn dokumentaation perusteella on tulevaisuudessa mahdollista toteuttaa vastaavanlaisia ratkaisuja vaivattomammin.

Asiasanat:
Modbus, Profibus, kenttäväylät, Fidelix, Simatic, kiinteistöautomaatio

ALKULAUSE

Insinöörityö on tehty kiinteistöautomaatioon erikoistuneelle Fidelix Oy:lle. Työn valvojana toimi projektinjohtaja Janne Tolonen Fidelix Oy:stä ja ohjaavana opettajana lehtori Tero Hietanen Oulun seudun ammattikorkeakoulun tekniikan yksiköstä.

Kiitän Fidelixin Oulun seudun aluepäällikkö Arto Nissilää opinnäytetyömahdollisuuden antamisesta. Avusta eri työvaiheissa haluan kiittää Tero Hietasta sekä työn loppuvaiheessa alakeskuksen konfiguroinnissa auttanutta Fidelixin huoltosinööri Ville Karhumaata. Suuri kiitos myös työni kieliasun tarkastaneelle lehtori Tuula Hopeavuorelle.

Oulussa 13.5.2013

Tuomas Rahkola

SISÄLLYS

TIIVISTELMÄ	3
ALKULAUSE	4
SISÄLLYS	5
1 JOHDANTO	9
2 TIETOLIIKENNEPROTOKOLLAT	11
2.1 OSI-kerrosarkkitehtuuri	11
2.2 TCP/IP-protokollaperhe	13
2.2.1 TCP	15
2.2.2 UDP	16
3 KENTTÄVÄYLÄT	17
3.1 Topologiat	17
3.2 Profibus DP	19
3.2.1 Perusta	20
3.2.2 Toimintaperiaate	21
3.2.3 Protokollaversiot	21
3.2.4 Master-laitetyypit	22
3.2.5 Viestikehykset	23
3.2.6 Asennus	23
3.2.7 Kaapelointi	24
3.2.8 Segmentointi	25
3.2.9 Terminointi	25
3.2.10 Maadoitus	26
3.3 Modbus/TCP	27
3.3.1 Perusta	27
3.3.2 Toimintaperiaate	29
4 KÄYTETYT LAITTEET JA OHJELMAT	32
4.1 ICP DAS GW-7553 -yhdyskäytävä	32
4.2 Fidelix FX-2025 -alakeskus	33
4.3 Simatic Manager	34
4.4 GW-7553-laitteohjelmat	35
5 TYÖN SUORITUS	36

5.1 Projektin luonti Simatic Managerilla	36
5.2 Yhteyden testaus virtuaalisella master-laitteella	40
5.3 Fidelix alakeskuksen toiminnan testaus	42
5.4 Laitteiston valmistelu lähetystä varten	44
6 POHDINTA	46
LÄHTEET	48
LIITTEET	51

TERMIT JA LYHENTEET

ADU	Application data unit
ASCII	American standard code for information interchange
DNS	Domain name system
FMS	Field bus message specification
GSD	General station description
I/O	Input/Output
IP	Internet protocol
ISO	International organization for standardization
LSB	Least significant bit
MAC	Media access control
MBAP	Modbus Application Header
MSB	Most significant bit
Olio	Ohjelmiston perusyksikkö, joka sisältää joukon loogisesti yhteenkuuluvaa tietoa ja toiminnallisuutta
OSI	Open systems interconnection
PDU	Protocol data unit
PI-kaavio	Prosessin instrumentointikaavio
PLC	Programmable logic controller
Profibus	Process field bus
RS-485	Differentiaalinen sarjaväylä, johon voi liittyä useita väylälaitteita samanaikaisesti

RTU	Remote terminal unit
SAP	Service access point
Standardi	Organisaation esittämä määritelmä siitä, miten jokin asia tulisi tehdä
TCP	Transmission control protocol
UDP	User datagram protocol

1 JOHDANTO

Fidelix Oy on vuonna 2002 perustettu, kotimainen rakennusautomaatio- ja turvajärjestelmiä kehittävä ja myyvä yritys. Voimakkaasti kasvavan yrityksen pääkonttori sijaitsee Vantaalla ja aluekonttoreita on Turussa, Tampereella, Oulussa, Joensuussa, Vaasassa sekä Kokkolassa. Fidelixin palveluksessa Suomessa on noin 100 henkilöä ja liikevaihto oli vuonna 2012 noin 12 miljoonaa euroa. Merkittävä osa yrityksen tuotteista menee vientiin. Edistykselliset ratkaisut ja palvelukonseptit perustuvat avoimuuteen ja standardoituihin tekniikoihin. (1.)

Kiinteistöautomaatio, eli rakennusautomaatio määritellään erilaisiksi automaattisiksi säätö-, valvonta-, ohjaus- ja hälytystoiminnoiksi, joiden avulla hallitaan kiinteistöjen LVIS-prosesseja (lämpö, vesi, ilma, sähkö). Tällaisia prosesseja ovat ilmanvaihto, valaistus, lämmitys, kulunvalvonta ja turvatoiminnot. Kiinteistöautomaatio on eroteltu omaksi ryhmäkseen erilleen prosessiautomaatiosta, sillä kiinteistöjen valvonta- ja säätökohteet eroavat paljon teollisuudesta löytyvistä. Aikaisemmin kiinteistöautomaatiojärjestelmiä oli pääosin vain suuremmissa kiinteistöissä, mutta nykyään niitä on runsaasti myös pienemmissä kohteissa, kuten omakotitaloissa ja vapaa-ajan asunnoissa. Kiinteistöautomaatiojärjestelmällä pyritään luomaan tiloihin halutunlaiset olosuhteet ja ominaisuudet. Optimaalisten olosuhteiden saavuttaminen on mahdollista, kun automaatiojärjestelmä on asennettu ja säädetty oikealla tavalla. Tällöin myös investoinnit alkavat maksaa itseään takaisin lähinnä kulutuksen optimoinnin kautta ja takaisinmaksuaika voi jäädä jopa muutaman vuoden mittaiseksi. (2, s. 9.)

Helmikuussa 2013 tiedustelin Fidelixin Oulun seudun aluepäällikkö Arto Nissilältä mahdollisia opinnäytteen aiheita yrityksessä. Arto lupasi ottaa yhteyttä, mikäli sopiva aihe tulisi tarjolle. Onnekseni Fidelix sai tilauksen viidestä tuloilmakoneesta Uruguayihin rakennettavalle Stora Enson sellutehtaalle. Tilauksen yhteydessä oli vaatimus kolmen tuloilmakoneen mittaus- ja tilatietojen tuomisesta Fidelixin alakeskuksilta tehtaan Honeywell-automaatiojärjestelmään Profibus-väylän kautta.

Työn varsinaisena tarkoituksena oli selvittää, voitaisiinko valitulla yhdyskäytävä-laitteella toteuttaa tietojen siirto Fidelixin alakeskukselta Modbus–Profibus-rajapinnan yli tehtaalla käytettävään automaatiojärjestelmään. Vastaavanlaisia ratkaisuja tullaan tarvitsemaan myös tulevaisuuden kohteissa. Tämän vuoksi työn tuli sisältää kattava dokumentaatio myöhempien testausten ja käyttöönottojen tueksi. Laitteiston toiminnan testaukseen oli käytettävissä Oulun seudun ammattikorkeakoulun tekniikan yksikön automaatiolaboratoriosta löytyvä Siemens S7-400 -sarjan ohjelmoitava logiikka, josta löytyi tarvittu Profibus-väylä.

2 TIETOLIIKENNEPROTOKOLLAT

Protokolla on säännöstö, jota käyttämällä kaksi saman protokollan toteuttavaa osapuolta voi suorittaa protokollan määrittelemiä tehtäviään niin, että kumpikin ymmärtää toisiaan. Sitä voisi verrata luonnolliseen kieleen kuten suomi tai englanti, jolla ihmiset keskustelevat keskenään. (3, s. 14; 4, s. 14.)

Tässä luvussa käydään lävitse työn kannalta oleellimmat protokollat. Protokoliin paneudutaan sillä tarkkuudella kuin varsinaisen työn suoritus sitä vaati. Lisäksi tarkastellaan arkkitehtuuria, jonka pohjalle ne rakentuvat.

2.1 OSI-kerrosarkkitehtuuri

Vuonna 1977 ISO (International organization for standardization) asetti komitean, jonka tehtäväksi annettiin tietoliikenteelle toimivan toimintamallin luominen. Ideana oli luoda laitevalmistajille ja verkon käyttäjille ympäristö, jonka avulla järjestelmät kykenisivät kommunikoimaan sujuvasti keskenään. Näin päästäisiin lukuisten erilaisten ja eri tarpeisiin kehitettyjen verkkojen yhteensovittamisesta eroon. Kuten tietoliikennemaailmassa usein on tapana monimutkaisen kokonaisuuden luonnissa, käytettiin pohjaideana suunnittelussa kerroksiin jaettua arkkitehtuuria. Vuosien työn tuloksena syntyi OSI-viitemalli (Open systems interconnection), jonka ISO hyväksyi vuonna 1983. Viitemalli muodosti perustan sille, miten laitteita liitetään toisiinsa hajautetuissa tietojärjestelmissä. (3, s. 18–19; 5, s. 7–8.)

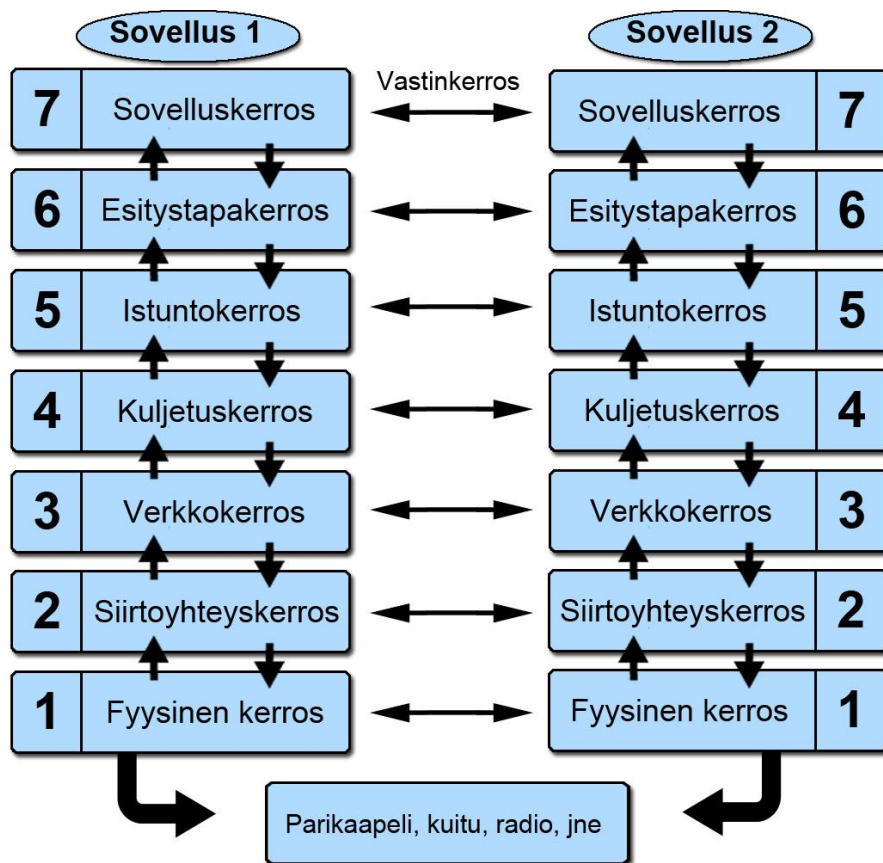
OSI-viitemalli on jaettu seitsemään erilliseen prosessointikerrokseen. Kerrosajattelun mukaisesti jokainen kerros tarjoaa palveluita ylemmälle kerrokselle käyttäen alemmalta kerrokseltaan saamiaan palveluja. Ne siis rakentuvat alempana olevan kerroksen toimintojen varaan. Kerrokset on erotettu toisistaan rajoilla, joita sanotaan rajapinnoiksi. Kaikki kyselyt kerroksesta toiseen välittyvät rajapintojen kautta. Rajapinnoilla sijaitsee SAP-palvelupisteitä (Service access point), joissa kerrosten välinen kommunikointi on määritelty. Kerrosten oliot käyttävät tai tarjoavat palveluita näiden pisteiden välityksellä ylemmälle ja alemmalle kerrokselle. Yläkerrokset palvelevat pääasiassa käyttäjätoimintoja ja sovellusohjelmistoja, alimmat neljä kerrosta muodostavat varsinaiset tietoliik-

kennepalvelut. Kenttäväylästandardin piiriin kuuluvat OSI-viitemallin kerrokset 1, 2 ja 7. OSI-viitemallin kerrokset kuvauksineen on esitetty kuvassa 1. (2, s. 83–84.)

<u>n:o</u>	<u>Nimitys</u>	<u>Kuvaus</u>
7	Sovelluskerros	Ikkuna, jonka kautta sovellukset ja niiden prosessit, kuten sähköpostiohjelmat ja sähköpostin lähetyksprosessit kiinnittyvät verkkoon.
6	Esitystapakerros	Huolehtii siirrettävien tietojen kuvaamisesta. Kompressoii, kryptaa ja suorittaa tiedon esitystapa- ja koodimuutokset.
5	Istuntokerros	Huolehtii istunnoista, tämä tarkoittaa käytännössä viestintäsuhteiden hallintaa ja tahdistustoimintojen tarjoamista.
4	Kuljetuskerros	Tehtävänä vuonhallinta sekä virheiden korjaus. Huolehtii siis luotettavasta tiedonsiirrosta laitteiden välillä.
3	Verkkokerros	Päätehtävänä reititys. Valitsee reitin, jota pitkin data kuljetetaan lähettävältä laitteelta vastaanottavalle laitteelle.
2	Siirtoyhteykskerros	Suorittaa sanomien siirron ja valvonnan solmupisteiden välillä. Huolehtii siirtoyhteyksien loogisesta rakentamisesta ja niiden purusta.
1	Fyysinen kerros	Huolehtii fyysisellä tasolla tiedonsiirrosta tarjoamalla sähköiset ja mekaaniset ominaisuudet sekä menetelmät tätä varten.

KUVA 1. OSI-viitemalli (5, s. 7–11)

OSI-viitemallin kerrokset on aseteltu niin, että ne vaikuttavat käyttäytyvän kuin voisivat kommunikoida toisen laitteen itseään vastaavan kerroksen kanssa suoraan välittämättä alemmista kerroksista (kuva 2). Todellisuudessa jokaisen viestikehyksen on kuljettava vähintäänkin lähettävän laitteen fyysisen kerroksen kautta vastaanottavan laitteen fyysiselle kerrokselle ja siitä eteenpäin kehyksestä löytyvän osoitteen perusteella vastaavalle kerrokselle kuin kerros, jolta se alunperin lähti. (5, s. 7–11.)



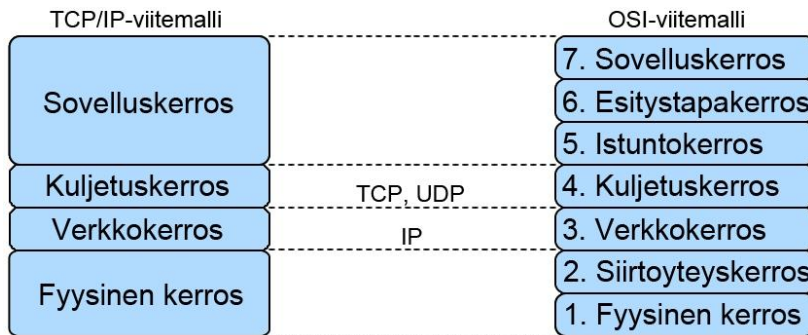
KUVA 2. OSI-viitemallin kerrosten väliset suhteet (2, s. 82)

2.2 TCP/IP-protokollaperhe

TCP/IP on tiedonsiirtokäytäntö, jonka kehitti alunperin Yhdysvaltain puolustusministeriö ARPANET:n (Advanced research projects agency network) kehitystyön yhteydessä. Kehitystyö aloitettiin 1960-luvulla ja se jatkui 1980-luvulle saakka, jolloin TCP/IP vakiinnutti oman asemansa. Nykyään verkkotekniikoiden yhtenäistyessä aletaan siirtyä kohti yhtä suurta monipalveluverkkoa. Tälle tulevaisuuden verkolle varten otettavana runkona on TCP/IP-protokollaperhe. Kantavana voimana sillä on sen yleisyys, avoimuus ja erinomainen käytettävyys erilaisissa laite- ja ohjelmistoarkkitehtuureissa. (6, s. 561.)

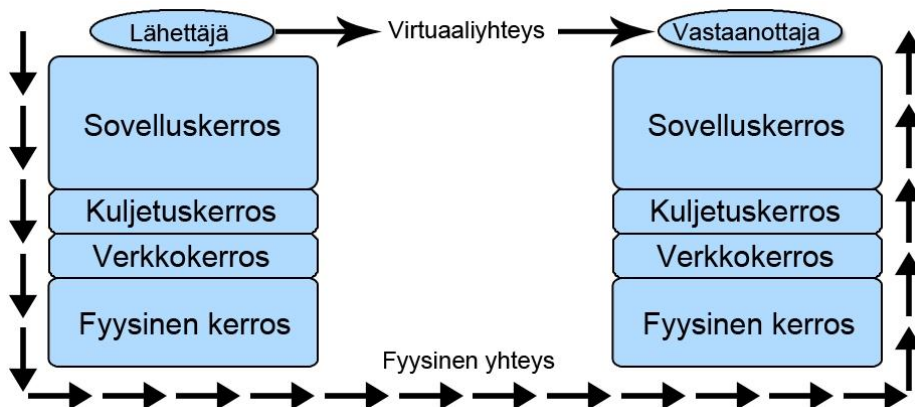
TCP/IP on hyvin samankaltainen OSI-viitemallin kanssa, mutta se käyttää seitsemän kerroksen sijasta neljää. Kukin kerros sisältää yhden tai useamman OSI-kerroksen toiminnallisuudet. Myös jakoa viiteen kerrokseen käytetään, mutta kuvassa 3 esitetty nelikerrosmalli on yleisempi. TCP/IP soveltuu käytettäväksi kiinteiden toimistolähiverkkojen lisäksi langattomissa ympäristöissä, automaa-

tioverkoissa sekä kaikkialla, missä on toteutettavissa TCP/IP-protokollaohjelmisto eli protokollapino. (3, s. 13–14.)



KUVA 3. TCP/IP-viitemallin sijoittuminen OSI-kerroksille (7, s. 8)

TCP/IP ei määrittele, mitä välitettävä data tarkoittaa tai miten sitä tulkitaan. Lähetettäessä tietoa viitemallin alempi kerros lisää ylemmältä kerrokselta vastaanottamansa tietosisällön omaan viestikehykseensä. Jokaisen viestikehyksen alkuun lisätään ylhäältä alaspäin siirryttäessä otsikkoon joko porttinumero, IP- tai Ethernet-osoite sen mukaan, mikä kerros kulloinkin tietoa välittää. Vastaanotettaessa tietoa viestikehys siirtyy otsikoiden sisältämien osoitteiden opastamana ylöspäin kerroksilla. Kukin kerros poistaa vuorollaan oman kehysrakenteensa ja välittää jäljelle jäävän tietosisällön ylemmälle kerrokselle. Jokainen kerros lähettäjän pinossa kommunikoi omalla viestikehyksen osallaan vastaanottajan vastaavassa pinossa samasta kohtaa löytyvän kerroksen kanssa. Kuva 4 havainnollistaa viestikehysten liikkeitä TCP/IP-pinoissa. (7, s. 10; 8, s. 247–250.)



KUVA 4. TCP/IP-pinojen toiminta niiden kommunikoidessa (7, s. 10)

IP (Internet protocol) toimii perustana koko TCP/IP-protokollaperheelle, sillä kaikki muut perheenjäsenet toimivat sen päällä. Protokollan toimiminen toisen protokollan päällä tarkoittaa, että ylemmällä tasolla toimivien protokollien viestit lähetetään alemman tason protokollan viestikehysten sisällä. (4, s. 14.)

IP:n päätehtävänä on mahdollistaa viestintä laitteelta toiselle monimutkaisissa, lukuisten itsenäisten verkkojen muodostamisissa kokonaisuuksissa. Tämä tapahtuu lisäämällä vastaanottajan ja lähettäjän IP-osoitteet viestikehyksiin, jotka muodostavat yhdessä kuljetustasolta saatujen porttinumeroiden kanssa yhteydet yksilöivät socketit. Kehykset reititetään protokollan parhaaksi katsomaa reittiä pitkin vastaanottajalle huomioiden kullekin paketille asetetut prioriteetit. IP:n alla voi rajoituksetta käyttää mitä tahansa datan siirtämiseen soveltuvaa protokollaa. (8, s. 249.)

IPv4 (Internet protocol version 4), joka on neljäs versio IP:stä, on nykyään vielä yleisesti käytössä, mutta sen puutteiden vuoksi ollaan hitaasti siirtymässä uudempaan IPv6-versioon (Internet protocol version 6). Version 4 mukainen osoite muodostuu neljästä tavusta eli 32 bitistä. Osoitteiden lyhydestä johtuu IPv4:n suurin puute eli osoitteiden määrän riittämättömyys. 32-bittisellä osoitteella voidaan ilmaista noin 4,3 miljardia osoitetta. Käytännössä luku on vielä huomattavasti pienempi, mikä johtuu osoiteavaruuden tehottomasta käytöstä. Seuraajaksi kehitetty IPv6-protokolla käyttää 128-bittisiä osoitekenttää. Näin saadaan muodostettua 2^{128} uniikkia osoitetta. (9, s. 215–216.)

2.2.1 TCP

TCP (Transmission control protocol) on TCP/IP-pinon kuljetuskerroksella sijaitseva yhteydellinen protokolla, joka kykenee ainoastaan point-to-point-muotoiseen yhteyteen. Se huolehtii luotettavan yhteyden muodostamisesta kahden eri laitteissa sijaitsevan ohjelman välille. Osapuolten kommunikoidessa TCP-protokollalla tulee niiden aluksi avata TCP-yhteys ja tiedonsiirron loputtua erikseen sulkea se. TCP varmistaa virheettömän liikennöinnin lisäämällä viestikehyksiin järjestysnumeron ja tarkisteen sekä jäämällä odottamaan kuittausta jokaisesta lähetetystä kehyksestä. Mikäli vastausta ei saada asetetussa aikamääreessä tai vastaanottajan kuittauksessaan ilmoittama vastaanotetun paketin

koko ei täsmää lähetetyn paketin kokoa, huolehtii TCP uudelleenlähetyksestä. TCP-protokolla valvoo aktiivisesti verkon tilaa ja suorittaa vuon hallintaa verkon kuormituksen perusteella. (3, s. 166–167; 4, s. 29.)

TCP toimii IP-protokollan päällä eli sen ei tarvitse huolehtia viestin välittämisestä oikealle laitteelle. Protokolla käyttää lähde- ja kohdeosoitteina porttinumeroita. Samassa laitteessa sijaitsevilla TCP-protokollaa käyttävillä ohjelmissa on aina oma yksilöllinen portti, jota se käyttää kommunikoinnissa. Koska TCP-yhteys avataan ohjelmien välille, voidaan jokainen yhteys yksilöidä socketin eli laitteen IP-osoitteen ja sen portin avulla, jota ohjelma laitteessa käyttää. Ohjelmat voivat käyttää useita portteja samanaikaisesti ja yhtä porttia useamman yhteyden avaamiseen. Porttinumero on kokonaisluku väliltä 0–65 535. Ensimmäiset 1024 porttia on varattu ennalta määrättyyn tarkoitukseen. TCP-viestikehyksen otsikossa ei määritellä tiedon laatua ja tästä johtuen osaa porteista käytetään vain tietyn tyyppisen tiedon välitykseen. Portti 502 on varattu Modbus/TCP:tä varten. (4, s. 29–30, s. 360.)

2.2.2 UDP

UDP (User datagram protocol) tarjoaa yksinkertaisen, epäluotettavan ja yhteydettömän palvelun kuljetuskerroksella. Protokollalta puuttuu kokonaan vuonohjaus ja pakettien uudelleenlähetykset. Jos pakettien uudelleenlähetykset halutaan toteuttaa, täytyy se tehdä sovellustasolla. UDP soveltuu hyvin syklisiin sovelluksiin, joissa vastaanottajan ja lähettäjän tulee pysyä synkronoituina, mutta muutamien sanomien hukkuminen matkalla ei ole vakavaa. Tällaisia verkkoja löytyy paljon automaatiotekniikan sovelluksista, joissa tieto vanhenee pian, eikä uudelleenlähetyksille ole aikaa. (3, s. 156–157.)

UDP-kehysten pituus on 8 tavua. Lähettäjän ja vastaanottajan porttitietojen lisäksi siitä löytyy ainoastaan kenttä tarkistussummalle ja pituudelle ennen vaihtelevan pituista datakenttää. UDP-protokolla käyttää portteja samalla tavalla kuin TCP-protokolla ja myös niiden määrä on sama. (3, s. 156–157.)

3 KENTTÄVÄYLÄT

Kenttäväylällä tarkoitetaan digitaalista, sarjaväyläistä ja kaksisuuntaista verkkoa, joka yhdistää älykkäät mittaus- ja ohjauslaitteet muuhun automaatiolaitteistoon yhdellä tiedonsiirtokaapelilla. Se on pitkälle hajautettu automaation toteutusympäristö, jossa osa järjestelmän toiminnoista voidaan siirtää älykkäille kenttäinstrumenteille. Siirrettäviä toimintoja ovat esimerkiksi mittaustietojen suodatus, skaalaus ja muunnokset. Kaksisuuntainen digitaalinen tiedonsiirto on tuonut mukanaan uuden sukupolven diagnostiikkajärjestelmät, jotka helpottavat siirtymistä korjaavasta kunnossapidosta ennakoivaan. (10, s. 7; 2, s. 33.)

Väylässä master-laite ja sille alisteiset hajautetut slave-laitteet viestivät käyttäen sarjamuotoista liikennettä. Master-laite valvoo väylän liikennettä ja slave-laite voi käyttää väylää viestimiseen vain silloin, kun se saa master-laitteelta siihen erillisen käskyn. (11, s. 141.)

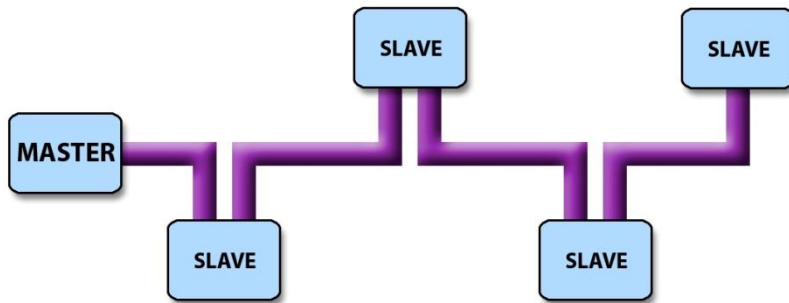
Kenttäväyläjärjestelmästä on laskettu saatavan merkittäviä etuja pienempien investointikustannusten sekä parantuneen prosessin hallinnan ja ylläpidon vuoksi. Verrattaessa perinteistä automaatiojärjestelmää kenttäväyläratkaisuun voidaan havaita selviä eroja esimerkiksi tarvittavassa kaapeloinnissa ja liitännöissä sekä järjestelmätason laitteiden määrässä. Kenttäväyläjärjestelmä vähentää kaapelointia ja kytkentäpisteiden sekä kytkentäliityntöjen määrää parhaassa tapauksessa jopa 60–70 prosenttia. Ratkaisu vähentää myös liityntäelektroniikkaa ja kenttätason yläpuolista laskentakapasiteetin tarvetta. (2, s. 33–35.)

3.1 Topologiat

Tässä luvussa puhutaan loogisen topologian sijaan fyysisestä topologiasta, jonka saa selville seuraamalla väylän kaapeleita. Topologiavaihtoehdot kenttäväylälle ovat linja-, tähti-, rengas- ja puumuotoinen sekä näiden yhdistelmät. Topologian nimitys riippuu siitä, kuinka kaapelointi on tehty suhteessa master-laitteeseen ja kuinka slave-laitteet on siihen kytketty. Väyläkaapelina käytetään yleisimmin kierrettyä parikaapelia. Muita käytettyjä kaapelityyppejä ovat optinen ja koaksiaalikaapeli. Jotta väylästä saataisiin mahdollisimman tehokas ja talou-

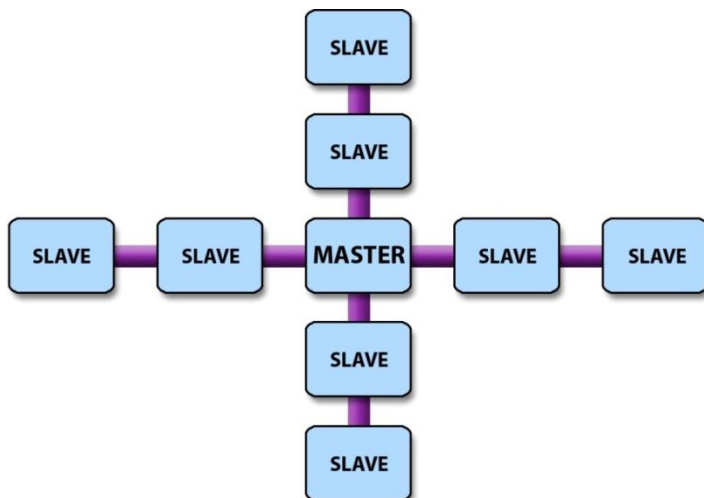
dellinen, on yhteyden oltava niin suora kuin mahdollista ja kaapelia tulee käyttää mahdollisimman vähän. (2, s. 71.)

Linjamuotoisessa kenttäväylässä laitteet on kytketty yhteen kaapeliin, joka kiertää laitteelta laitteelle (kuva 5).



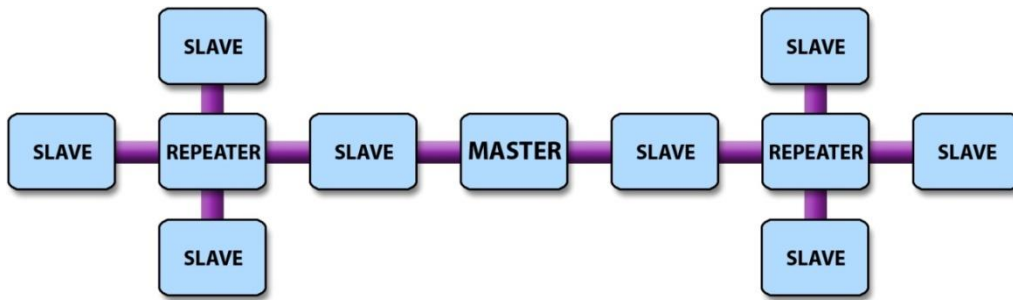
KUVA 5. Linjamuotoinen topologia

Tähtityyppinen kenttäväylä eroaa linjamuotoisesta siten, että master-laitteelta lähtee useita linjoja (kuva 6).



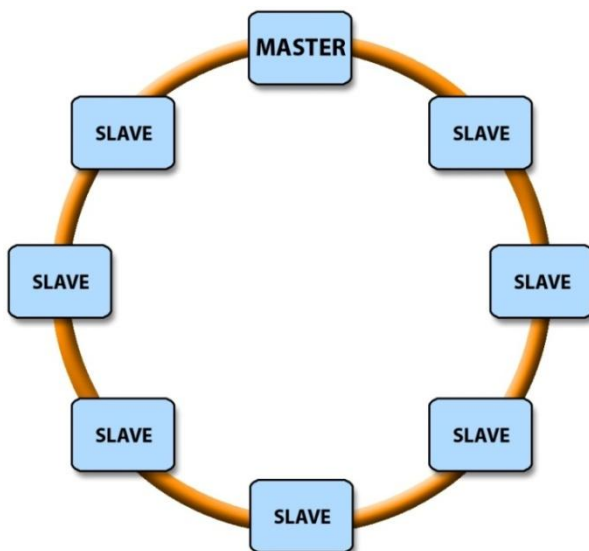
KUVA 6. Tähtityyppinen topologia

Puutyypisessä kenttäväylässä kaapelointi haarautuu puun oksien tapaan. Haarautuminen on mahdollista saada aikaan käyttämällä esimerkiksi repeate-reita eli toistimia (kuva 7).



KUVA 7. Puutyypinen topologia

Rengastyypisessä kenttäväylässä kaapelointi kulkee katkeamattomana ketju-
na aloituspisteestä jokaisen kenttälaitteen kautta takaisin samaan lähtöpisteeseen (kuva 8).



KUVA 8. Rengastyypinen topologia

3.2 Profibus DP

Profibus DP:tä (Decentralized peripherals) käytetään nopean, aikakriittisen tiedon välitykseen sekä laaja-alaisiin, monimutkaisiin kommunikointitehtäviin. Sovelluskohteita löytyy teollisuus- ja prosessiautomaatio sekä rakennusalalta. (12.)

3.2.1 Perusta

Saksalaiset yritykset ja alan yhteisöt kehittivät Profibusin (Process field bus) 1980-luvulla. Ensimmäisen kerran Profibus esiteltiin vuonna 1989. Nykyään Profibus on yleisin, etenkin Euroopassa hyvin suosittu kokoelma kenttäväylä-protokollia. Se on standardiin EN 50170 perustuva, toimittajasta riippumaton ja avoin protokollaperhe. Tunnetuimpia protokollia ovat Profibus DP, Profibus PA (Process automation) ja Profibus FMS (Field bus message specification). Tilaa- jien on mahdollista yhdistellä protokollia tarpeidensa mukaan. Profibus määrittelee sarjamoitoiselle kenttäväyläjärjestelmälle tekniset ja toiminnalliset ominai- suudet, joilla hajautettujen digitaalisten laitteiden yhteenkytkentä voidaan toteut- ta. Toimittajasta riippumattomuus näkyy siten, että väylään voidaan liittää eri valmistajien laitteita ja ne kykenevät kommunikoimaan keskenään. Vuoden 2012 loppuun mennessä oli asennettu yli 43,6 miljoonaa Profibus-laitetta, joista 7,5 miljoonaa prosessiteollisuudessa. (13; 12.)

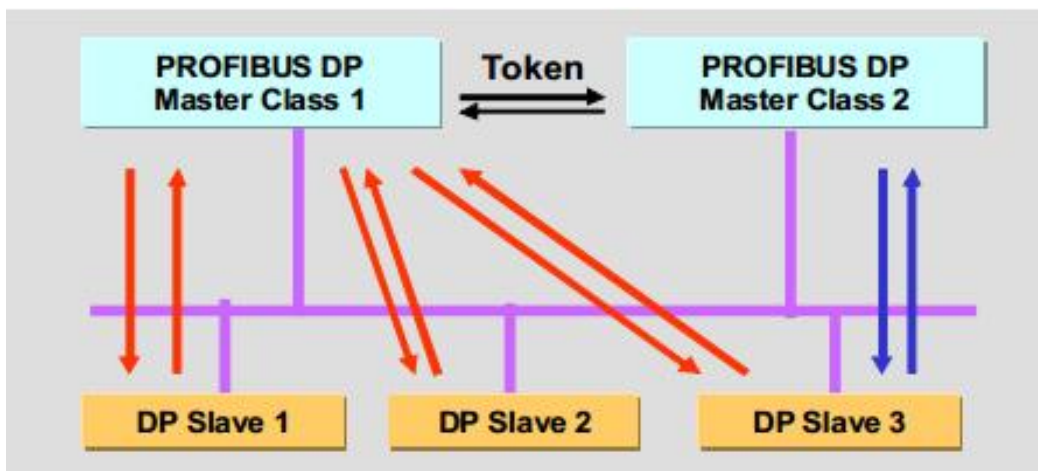
Profibus-standardi määrittelee OSI-viitemallista fyysisen, siirto- ja sovellusker- roksen (kuva 9). Fyysinen kerros määrittelee väylän fyysisen toteutuksen. Siir- toyhteyskerros jakaa väylää tarvitseville laitteille hallintaoikeudet väylän käyt- töön ja sovelluskerros huolehtii datan oikeasta muodosta, kuittauksista ja aika- taulutuksesta.

OSI-malli		PROFIBUS	
7	Sovelluskerros	DP-V0,-V1,-V2	7
6	Esitystapakerros		6
5	Istunterkerros		5
4	Kuljetuskerros		4
3	Verkkokerros		3
2	Siirtoyhteyskerros	Väylän hallinta	2
1	Fyysinen kerros	Sähköinen, Langaton, Optinen	1

KUVA 9. OSI-viitemalli ja Profibusin sijoittuminen sen kerroksille (14, s. 3)

3.2.2 Toimintaperiaate

Profibus DP erottaa laitteet master- ja slave-laitteiksi. Järjestelmässä voi olla monta masteria, jotka hallinnoivat väylällä tapahtuvaa tiedonsiirtoa. Kunkin masterin on ennen väylälle lähettämistä hankittava siihen vaadittava tunniste. Väylällä käytetään token passing -menetelmää, jossa se master, jolla token-tunniste kulloinkin on, voi käyttää väylää. Lähetysten jälkeen token siirtyy seuraavalle master-laitteelle syklisesti (kuva 10). Slave-laitteita ovat anturit, toimilaitteet ja lähettimet. Niillä ei ole väylälle pääsyoikeutta, vaan ne ainoastaan lähettävät kuittaussanomiam saamistaan viesteistä tai master-laitteen pyynnöstä välittävät sille halutut tiedot, poikkeuksena uusi Profibus DP:n protokollaversio DP-V2. Kaikille Profibus-väylän laitteille on määritetty oma osoitteensa, jolloin lähetetty sanoma voidaan osoittaa oikealle laitteelle. Tuettuna ovat osoitteet väliltä 0–127. Osoitteet 0, 126 ja 127 on varattu erityiskäyttöön. (15, s. 36–38; 16, s. 21.)

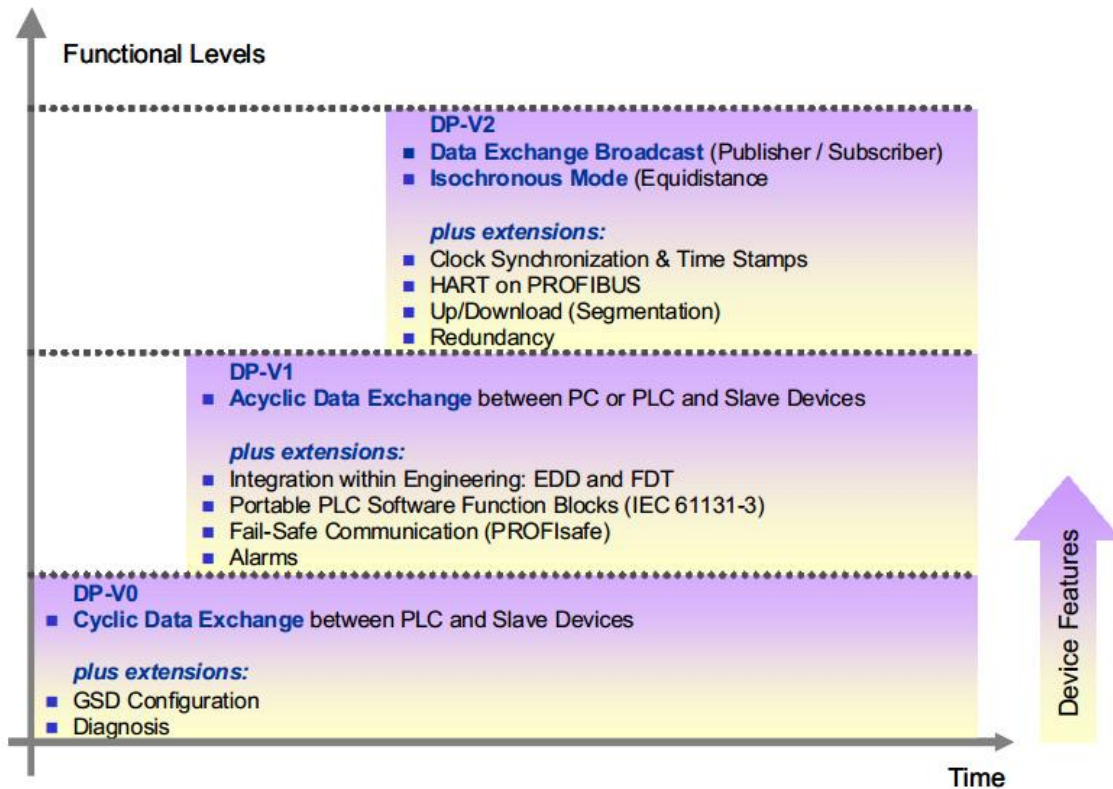


KUVA 10. Token passing -menetelmän toimintaperiaate (16, s. 21)

3.2.3 Protokollaversiot

Profibus DP:stä on olemassa kolme protokollaversiota: DP-V0, DP-V1 ja DP-V2 (kuva 11). DP-V0 tukee master–slave-peruskommunikointia. Se määrittelee perustoiminnallisuudet, kuten syklisen datan siirron maksimissaan 126 solmulle. DP-V1 sisältää lisäominaisuuksia, joita ovat esimerkiksi ajon aikainen laitteiden vaihto, parametointi sekä kalibrointi ja kyky asykliseen tiedonsiirtoon. Syklinen tiedonsiirto on jatkuvaa mittaustiedon tai ohjaussignaalin välittämistä. Asyklista

tiedonsiirtoa tapahtuu vain tarvittaessa. Se koostuu yleensä tapahtumista, joita ovat esimerkiksi hälytykset ja konfigurointi. DP-V2 on uusin ja kattavin versio DP-väylästä. Se mahdollistaa slave–slave-kommunikoinnin, jossa slave-laitteet kommunikoivat suoraan toisilleen. (14, s. 9–10.)



KUVA 11. Profibus DP:n protokollaversiot ominaisuuksineen (14, s. 10)

3.2.4 Master-laitetyypit

Master-laitteita löytyy kahdentyyppisiä, DPM1 (DP master class 1) ja DPM2 (DP master class 2). Yleensä master-laite on DPM1. DPM1:llä on oikeus aktiiviseen väylälle liittymiseen ja se voi lukea sisääntuloja ja kirjoittaa ulostuloihin. DPM2-tyyppin isäntälaitteesta löytyy DPM1:n ominaisuuksien lisäksi suunnitteluun ja käyttöönottoon tarvittavia ominaisuuksia, kuten diagnostiikka, konfigurointi sekä osoitteen anto. DPM1-tyyppin master-laite käyttää syklistä ja DPM2-tyyppin asyklisiä tiedonsiirtoa. DPM2:ta käytetään yleensä väylän käyttöönotossa ja analysoinnissa, joten sen ei tarvitse olla pysyvästi kytkettynä väylään. Useat laitekonfigurointityökalut sisältävät DPM2-toimintoja osittain. (14, s. 10–11.)

3.2.5 Viestikehykset

Viestikehysten rakenne Profibus DP -väylälle määritellään standardissa IEC 61158. Datan siirtoa, uusien asemien etsintää sekä väylän käyttöoikeuden siirtoa varten löytyy omat kehysrakenteensa. Siirrettävän datan pituus on kiinteästi 8 tavua tai vaihtoehtoisesti datan pituus voi vaihdella välillä 1–246 tavua. Kuvassa 12 on esitetty kiinteän ja datamäärältään vaihtelevan viestikehyksen muoto. (17, s. 17.)

Muuttuva datan määrä

Selite	SD2	LE	LEr	SD2	DA	SA	FC	DSAP	SSAP	DU	FCS	ED
Pituus	tavu	tavu	tavu	tavu	tavu	tavu	tavu	tavu	tavu	1-246 tavua	tavu	tavu

Kiinteä datan määrä

Selite	SD3	DA	SA	FC	DU	FCS	ED
Pituus	tavu	tavu	tavu	tavu	8 tavua	tavu	tavu

- SD = Start Delimitter, alkuerotin (SD2 = 0x68, SD3 = 0xA2)
- LE = Length, siirrettävän datan pituus (sisältää: DA, SA, FC, DSAP, SSAP)
- LEr = Length repetition, siirrettävän datan pituus toistettuna
- DA = Destination Address, kohteen osoite
- SA = Source Address, lähettäjän osoite
- FC = Function Code, funktiokoodi
- DSAP = Destination Service Access Point, kohteen portti
- SSAP = Source Service Access Point, lähettäjän portti
- DU = Data Unit, datakenttä (hyötydata)
- FCS = Frame Checking Sequence, kehyksen tarkistussekvenssi
- ED = End Delimitter, loppuerotin (0x16)

KUVA 12. SD2- ja SD3-tyypin Profibus-viestikehykset (17, s. 17)

Kun lasketaan yhteen viestikehyksen eri osien sisältämät tavut, saadaan selville kehyksen sisältämien tavujen kokonaismäärä. Perinteisen määritelmän mukaan tavuun sisältyy 8 bittiä; Profibus-väylällä jokaisen tavun eli kahdeksan bitin siirtämiseen tarvitaan kuitenkin 11 bittiä. Siirrettävän tavun alkuun lisätään aloitusbitti, databittien perään tulee pariteettibitti ja lopetusbitti. (17, s. 15.)

3.2.6 Asennus

Profibus-väylä on erittäin helppoa asentaa väärin. Yleisimmät ongelmat liittyvät puutteelliseen maadoitukseen sekä väylän väärin toteutettuun terminointiin. Kun muistaa muutamat perussäännöt, onnistuu väylän kokoaminen ja käyttöönotto

paljon kivuttomammin. Kenttäväylätopologiaksi on valittavissa kaikki topologiat luvussa mainituista.

3.2.7 Kaapelointi

Profibus DP perustuu sähköisesti RS-485-standardiin. RS-485:ssä käytettävän kierretyn parikaapelin ja differentiaalisen signaloinnin vuoksi kumoutuvat yhteismuotoiset häiriöt väylässä. RS-485:n etuja ovat hinta, nopeus ja hyvä häiriönsieto. Väylää rakennettaessa on käytössä myös optinen ja langaton tiedonsiirto muuntimia käyttämällä. Kullakin siirtotavalla on omat etunsa ja heikkoutensa.

Kaapelin maksimipituus riippuu vaaditusta siirtonopeudesta ja kaapelityypistä. Pituus vaihtelee 100 metristä kymmeneen kilometriin. Taulukkoon 1 on listattu suurin RS-485-kaapelilla saavutettava tiedonsiirtonopeus segmenttien eri pituuksilla. Taulukoissa 2 ja 3 on esitelty sähköisiä signaaleja käyttävien kupari-kaapeleiden sekä optisten tiedonsiirtokaapelien tekniset tiedot sekä rajoitukset. (16, s. 26–28.)

TAULUKKO 1. Tiedonsiirtonopeuden riippuvuus segmentin pituudesta (18, s. 53)

Transmission rate [kbits/s]	maximum Transmission distance [m]
9.6	1200
19.2	1200
45.45	1200
93.75	1200
187.5	1000
500	400
1500	200
3000	100
6000	100
12000	100

TAULUKKO 2. Kuparikaapelit (16, s. 26)

	RS - 485 PROFIBUS DP	MBP PROFIBUS PA	MBP- IS PROFIBUS PA
Baud rate	9.6 ... 12.000 kBit/s	31.25 kBit/sec	31.25 kBit/sec
Devices/segment (max.)	32	32	32
Devices/segment (typic.)		14 ... 20	4 ... 6
Cable length max.	1200	1900 m	1000 m
Spur line length max.		120 m	60 m

TAULUKKO 3. Optiset kaapelit (16, s. 28)

Fiber type	Core diameter [μm]	Transmission range
Multi-mode glass fiber	62,5 / 125	2 - 3 km
Single-mode glass fiber	9 / 125	> 15 km
Plastic fiber	980 / 1000	Up to 100 m
HCS® fiber	200 / 230	Approx. 500 m

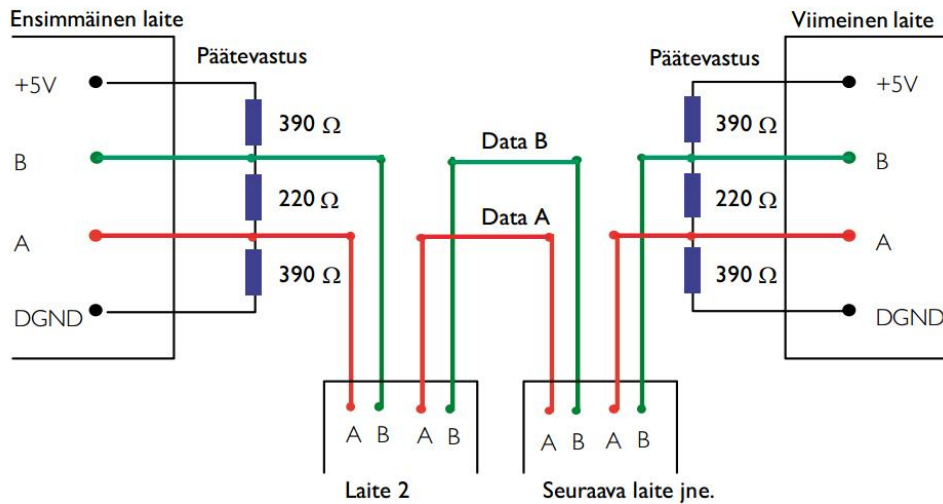
3.2.8 Segmentointi

Mikäli väylälaitteita on runsaasti, on väylä jaettava segmentteihin, jotka liitetään yhteen toistimilla. Segmenttiä kohden lisättävien laitteiden määrän ylärajana pidetään yleisesti 32:tä laitetta. Todellisuudessa laitteiden määrä on kuitenkin lähes poikkeuksetta pienempi. Väylään laitteita voidaan liittää enintään 126 kappaletta. Toistimia käytetään myös galvaaniseen erotukseen ja väylän enimmäispituuden kasvattamiseen. Toistin vahvistaa väylän signaalit alkuperäiseen voimakkuuteensa sekä poistaa syntyneitä häiriöitä. Samassa väylässä käytettävien toistimien enimmäismäärä riippuu niiden tyypistä. (18, s. 112–114.)

3.2.9 Terminointi

Väylän luotettavan toiminnan takaamiseksi on väyläsegmenttien molempiin päihin kytkettävä päätevastus. Päätevastuksien tarkoituksena on estää häiriöt ja heijastukset. RS-485-väylän päätevastus koostuu kolmesta vastuksesta (kuva 13). Terminointi tapahtuu käytettävällä väyläliittimellä tai aktiivisen päätevastuksen avulla. Yleisesti käytössä olevasta 9-napaisesta D-liittimestä (kuva 14) löytyy sisäinen päätevastuskytkentä ja se saadaan aktiiviseksi liittimestä löytyvällä

On/Off-kytkimellä. Toisin kuin aktiivinen päätevastus, kaapelin D-liittimen päätevastus ei tarvitse erillistä jännitesyöttöä, vaan se on otettu sisäisessä kytkennässä huomioon. (19, s. 1–3.)



KUVA 13. Profibus-väylän terminointi molemmista päistä kolmella vastuksella (19, s. 2)



KUVA 14. 9-napaisen D-liittimen kytkentäohje ja PG-liitin, eli läpiviennillinen D-liitin (20, s. 10)

3.2.10 Maadoitus

Kunnollisen toimintavarmuuden saavuttamiseksi Profibus DP -väylä tulee maadoittaa hyvin. Maadoituksen tarkoitus on johdattaa ulkoiset häiriöt pois väyläkaapelista. Jokaisella laitteella ja aina, kun kaapelilla mennään kaapin sisään tai ulos, tulisi kaapelin vaippa maadoittaa (kuva 15). Tarvittaessa voidaan asentaa

myös lisämaadoituksia. Vaipan maadoitus tulee toteuttaa niin, että vaikka kaapeli olisi irti laitteesta, on vaippa silti maadoitettu. Profibus DP -väylän kaapeleita asennettaessa tulee muistaa, ettei niitä saa vetää yhdessä sähkösyöttöjen kanssa. Esimerkiksi etäisyys kategorია III:n (< 400 V) kaapeloinnista on oltava vähintään 0,2 m ja kategorია IV:n kaapeleista yli 0,5 m. Poikkeuksena häiriövaapaat optiset väyläkaapelit (19, s. 1–3.)



KUVA 15. Molemmista päistä maadoitettu Profibus-kaapeli (18, s. 166)

3.3 Modbus/TCP

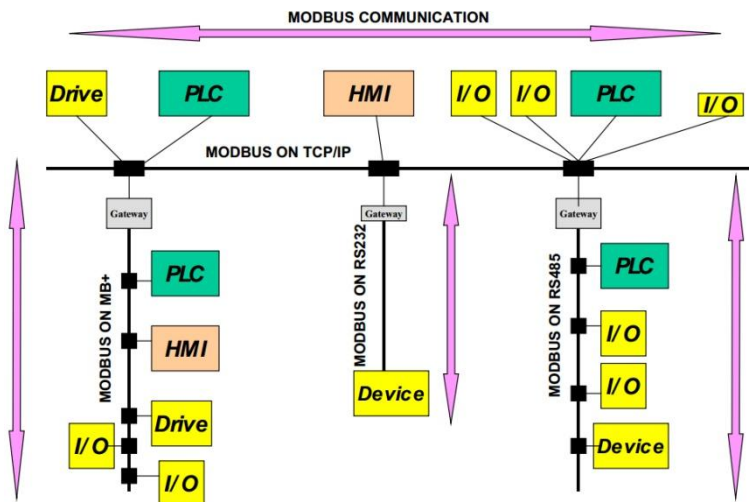
Modbus/TCP on TCP/IP-pohjainen muunnos alkuperäisestä Modbusista ja se toimii tiedonsiirtoprotokollana Ethernet-verkoissa. Viestikehysten välitykseen protokollassa käytetään TCP:tä. Varsinaisen datan sisältävä Modbus-kehys kapseloidaan TCP-kehysten sisälle. (21, s. 4–6.)

3.3.1 Perusta

Modbus on Modiconin, nykyisen Schneider Electricin, vuonna 1979 julkaisema protokollaperhe, joka alun perin kehitettiin väyläksi ohjelmoitavien logiikoiden liittämiseksi toisiinsa. Modbus perustuu avoimeen arkkitehtuuriin ja sitä käyttäviä laitteita voi kuka tahansa valmistaa ilman erillistä korvausta protokollan kehittäjille. Modbusia käytetään laajasti teollisuuden lisäksi rakennuskohteissa ja ohjauspaneelien yhdistämisessä. (2, s. 243–245; 21, s. 2.)

Modbusin kolme yleisimmin käytettyä sovellusta ovat Modbus/RTU (Remote Terminal Unit), Modbus/ASCII (American Standard Code for Information Interchange) ja Modbus/TCP. RTU ja ASCII versiot Modbusista toimivat yleisimmin RS-485-sarjaväylän päällä ja TCP versio Ethernet-verkossa. Kaikkien edellä mainittujen sovellusten liittäminen toisiinsa on helposti toteutettavissa käytännöllä yhdyskäytäviä (kuva 16). Yhteensovittamisen mahdollistaa se, että kaikki

käyttävät samaa Modbus-standardissa määriteltyä PDU-kehystä (Protocol data unit), jonka ne kapseloivat oman viestikehysrakenteensa sisälle. Yhdyskäytävät vain purkavat viestikehysten ja kapseloivat PDU-kehukset uudelleen. (22.)



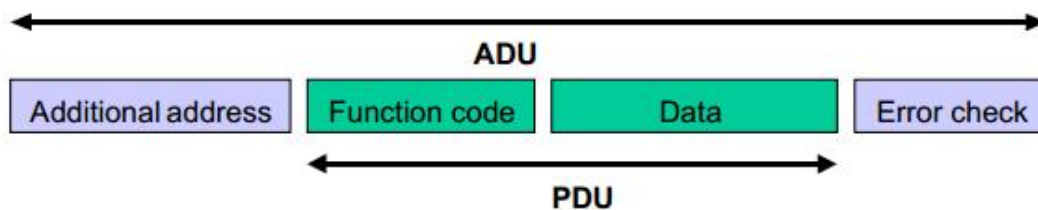
KUVA 16. Usean Modbus-sovelluksen liittäminen yhteen yhdyskäytävillä (23, s. 3)

Modbusin yleisimmät fyysisen kerroksen toteutukseen käytettävät tekniikat ovat RS-232, RS-485 sekä Ethernet-verkko. Muita käytettävissä olevia tekniikoita ovat esimerkiksi valokuitu ja langattomat yhteydet. Poikkeuksena on Modbus+, joka määrittelee oman fyysisen kerroksensa. Ethernet-verkon käyttö on muodostumassa suosituimmaksi vaihtoehdoksi. (22; 23, s. 3.)

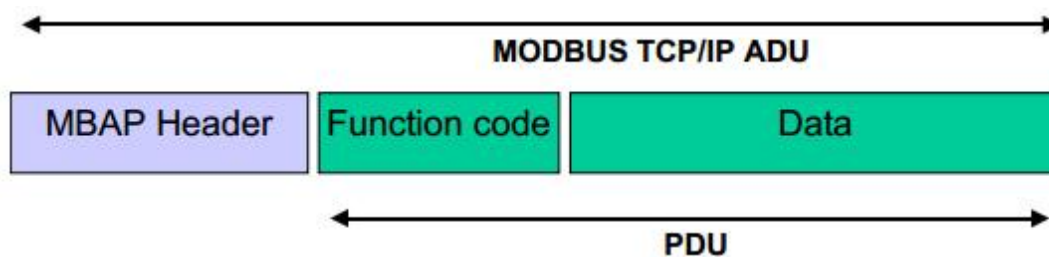
Modbus-protokolla kattaa OSI-viitemallin kerrokset 1, 2 ja 7. Se toimii half duplex -periaatteella, eli liikennöinti on vuorottelevaa lähetystä ja vastaanottoa. Sarjaporttiversioiden yhteydessä käytetään keskustelevalle laitteille nimityksiä master ja slave sekä Ethernet-versiota käytettäessä client ja server. Selkeyden vuoksi seuraavissa luvuissa puhutaan myös Modbus/TCP:n yhteydessä master- ja slave-laitteista. Sarjaliikenteisessä versiossa master suorittaa väylän kyselyt. Yleensä mitta- tai ohjainlaitteita olevat slave-laitteet vain vastailevat niille osoitettuihin kyselyihin. Ethernet-käytössä sekä master- että slave-laite voi kirjoittaa kyselyitä verkkoon, mutta yleensä kyselyt hoitaa yksi master. Harvemmin käytetty Modbus+ mahdollistaa usean masterin väylällä token passing -menetelmällä. (2, s. 244; 21, s. 2.)

3.3.2 Toimintaperiaate

Modbus/TCP:ssä käytetään pohjana viestikehyksille Modbus-standardissa määriteltyä PDU-kehystä. Toisin kuin sarjaliikenneversioissa Modbus/TCP ei lisää viestikehyksiin osoitekenttää eikä virheentarkistussummaa, vaan MBAP-otsikkolohkon (Modbus application header). MBAP pitää sisällään viestin järjestysnumeron, protokollatunnuksen, pituuden sekä laitenumeron. Virheentarkistussummaa ei tarvita, koska protokollassa käytetään TCP/IP-pinon kuljetuskerroksen toimintavarmaa yhteydellistä TCP:tä. Kuva 17 on normaalista sarjaliikenneviestinnässä käytettävästä ADU (Application data unit) viestikehyksestä ja kuva 18 Modbus/TCP:n ADU viestikehyksestä. (21, s. 4–6.)



KUVA 17. Sarjaliikenteisen Modbus-sovelluksen ADU-viestikehys (21, s. 4)

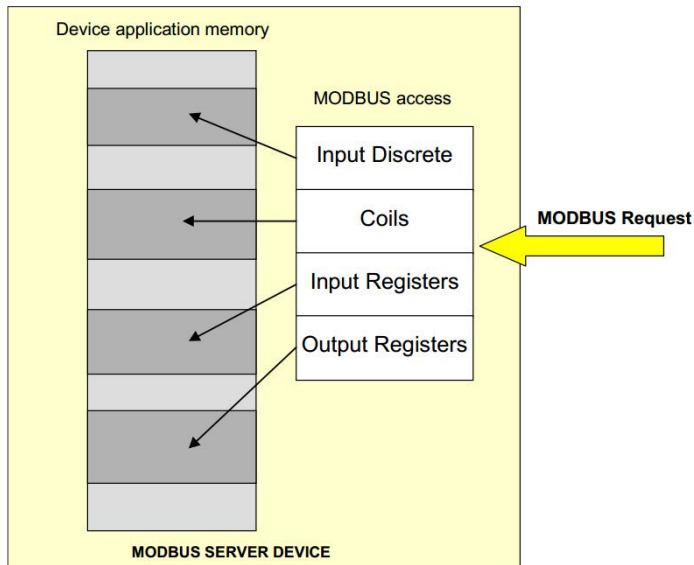


KUVA 18. Modbus/TCP:n ADU-viestikehys (21, s. 4)

Modbus/TCP:ssä, kuten muissakin Modbusin sovelluksissa, käytetään neljää erilaista rekisterityyppiä. Rekisterityypit ovat discrete input, coils, input registers sekä holding registers. Kahdessa ensimmäisessä rekisterissä jokaiseen muistiinpaikkaan kirjoitetaan yhden bitin mittainen data ja jälkimmäisissä 16 bitin mittaiset sanat (taulukko 4). Jokaisen rekisterityypille luku- ja kirjoitusominaisuuksia varten löytyy omat funktiokoodinsa. Kuten tutkittavassa yhdyskäytävässäkin, on yleisin tapa Modbus-laitteen rekisterien sijoittelulle luoda jokaiselle tyyppille oma moduulinsa (kuva 19). (21, s. 7–8.)

TAULUKKO 4. Modbus/TCP:n rekisterityypit (21, s. 8)

Primary tables	Object type	Type of	Comments
Discretes Input	Single bit	Read-Only	This type of data can be provided by an I/O system.
Coils	Single bit	Read-Write	This type of data can be alterable by an application program.
Input Registers	16-bit word	Read-Only	This type of data can be provided by an I/O system
Holding Registers	16-bit word	Read-Write	This type of data can be alterable by an application program.



KUVA 19. Modbus/TCP:n rekisterimoduulien sijoittuminen laitemuistiin (21, s. 8)

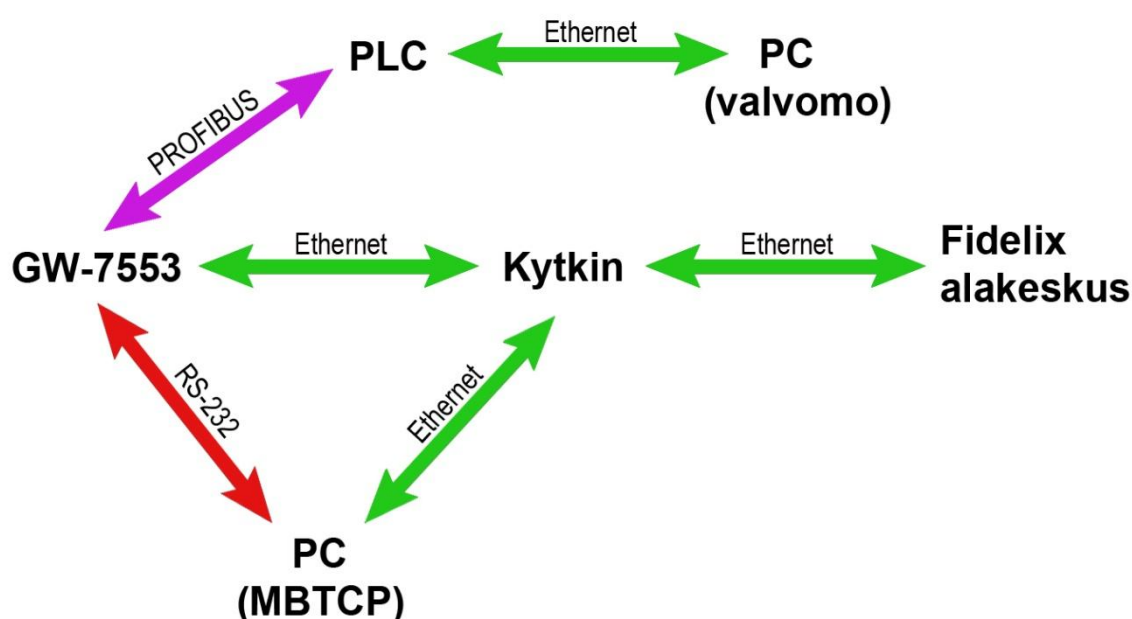
Modbus-standardi määrittelee kolmentyyppisiä funktiokoodoja. Julkiset funktiokoodit ovat hyvin määriteltäviä, dokumentoituja ja Modbus.orgin hyväksymiä (taulukko 5). Niitä kutsutaan kooditunnuksilla 1–127. Muun muassa kaikki yhdyskäytävän tukemat funktiokoodit lukeutuvat julkisiin funktiokodeihin. Loput kaksi harvemmin käytettyä tyyppiä ovat käyttäjän määrittelemät sekä varatut funktiokoodit. (23, s. 10–11.)

TAULUKKO 5. Modbusin julkiset funktiokoodit (23, s. 11)

				Function Codes			
				code	Sub code	(hex)	Section
Data Access	Bit access	Physical Discrete Inputs	Read Discrete Inputs	02		02	6.2
			Read Coils	01		01	6.1
		Internal Bits Or Physical coils	Write Single Coil	05		05	6.5
			Write Multiple Coils	15		0F	6.11
	16 bits access	Physical Input Registers	Read Input Register	04		04	6.4
			Read Holding Registers	03		03	6.3
		Internal Registers Or Physical Output Registers	Write Single Register	06		06	6.6
			Write Multiple Registers	16		10	6.12
			Read/Write Multiple Registers	23		17	6.17
			Mask Write Register	22		16	6.16
		Read FIFO queue	24		18	6.18	
		File record access	Read File record	20		14	6.14
	Write File record		21		15	6.15	
	Diagnostics	Read Exception status	07		07	6.7	
		Diagnostic	08	00-18,20	08	6.8	
		Get Com event counter	11		0B	6.9	
		Get Com Event Log	12		0C	6.10	
		Report Server ID	17		11	6.13	
		Read device Identification	43	14	2B	6.21	
	Other	Encapsulated Interface Transport	43	13,14	2B	6.19	
CANopen General Reference		43	13	2B	6.20		

4 KÄYTETYT LAITTEET JA OHJELMAT

Työn suoritukseen tarvittiin monia laitteita, jotka saatiin sovitettua yhteen lukuisien ohjelmistojen ja apulaitteiden avulla (kuva 20). Ethernetin kytkentäpisteiden lisäämiseksi täytyi laitteistoon lisätä kytkin. Kytkin on laite, joka pystyy välittämään eri lähdeporteista eri kohdeportteihin menevää liikennettä useiden liityntöjensä välillä samanaikaisesti. Kytkimen taustaväylät ovat siis tyypillisesti sellaisia, ettei törmäyksiä tapahdu, vaikka portista toiseen kulkevan liikenteet menisivätkin ristiin. Kytkin toimii siis tähtipisteenä Ethernet-verkon laitteille. (3, s. 30.)



KUVA 20. Työn suoritukseen käytetyn laitteiston kytkentöjen malli

4.1 ICP DAS GW-7553 -yhdyskäytävä

Yhdyskäytävä on laite tai ohjelma, jolla yhdistetään eri protokollia käyttäviä järjestelmiä toisiinsa. Käytännössä yhdyskäytävä purkaa protokollan X mukaisen paketin ja kokoaa uuden protokollaa Y vastaavan paketin. (3, s. 31; 2, s. 265.)

Profibus DP ja Modbus/TCP ovat laajalti käytettyjä protokollia tehdas-, prosessi- ja rakennusautomaatiojärjestelmissä. GW-7553 (kuva 21) toimii yhdyskäytävänä Profibus DP- ja Modbus/TCP-protokollien välillä. GW-7553:a käyttämällä on helppoa lisätä Modbus-laitteet osaksi Profibus-väylää. Modbus-laitteet voivat

vaihtaa dataa Profibus-master-laitteen kanssa, käyttäen yhdyskäytävän tukemia Modbus-funktiokoodoja (taulukko 6). GW-7553-yhdyskäytävä on suunniteltu erityisesti slave-laitteeksi Profibus-väylään, mutta Modbus-väylällä se voi toimia sekä slave- että master-laitteena. Etäkäytön helpottamiseksi yhdyskäytävään on lisätty myös mahdollisuus selainpohjaiseen asetusten muuttamiseen.



KUVA 21. GW-7553-yhdyskäytävä (24, s. 1)

TAULUKKO 6. GW-7553:n tukemat Modbus-funktiokoodit (24, s. 5)

Code	Name	Description
01	Read Coil Status	Read the ON/OFF status of discrete outputs in the slave
02	Read Input Status	Read the ON/OFF status of discrete inputs in the slave
03	Read Holding Registers	Read the binary contents of holding registers in the slave
04	Read Input Registers	Read the binary contents of input registers in the slave
05	Force Single Coil	Write a single output to either ON or OFF in the slave
06	Preset Single Register	Write an integer value into a single register in the slave
15	Force Multi. Coils	Write each coil in the sequence of coils to either ON or OFF in the slave
16	Preset Multi. Registers	Write a block of contiguous registers in the slave

4.2 Fidelix FX-2025 -alakeskus

Vapaasti ohjelmoitava FX-2025-alakeskus (kuva 22) perustuu teollisuus-PC:hen. Käyttöjärjestelmänä käytetään Windows CE:tä. FX-2025:n käyttöliittymä on selainpohjainen ja helppokäyttöinen. Alakeskusta voi käyttää joko paikallisesti 10,4-tuumaiselta kosketusnäytöltä tai verkkoselaimella etäyhteyden kautta. Näytön eliniän pidentämiseksi on siihen integroitu läsnäoloilmaisin, joka sammuttaa ja käynnistää näytön automaattisesti. Näytön yläpuolelta löytyvä

merkkivalo ilmoittaa alakeskuksen hälytystilanteen. Tarkemmat tekniset tiedot alakeskuksesta löytyvät taulukosta 7. (25, s. 1.)



KUVA 22. Fidelix FX-2025 -alakeskus

TAULUKKO 7. Fidelix FX-2025 -alakeskuksen tekniset tiedot (25)

Tekniset tiedot

Syöttöjännite/virta:	24 VDC
Käyttölämpötila:	0 °C – 50 °C
Käyttöolosuhteet:	Max 95 %RH, ei kondensivettä
Koko:	290 mm x 235 mm x 60 mm
Paino:	3,0 kg
Suojausluokka:	IP21
Liitynnät:	2x Ethernet RJ-45, RS-485 Modbus, RS-232, 2x USB PS-2, VGA (ulkoinen näyttö)
RAM-muisti:	128 MB
Flash-muisti:	128 MB (industrial)
Proessori:	AMD GX2 333 MHz
Merkkivalo hälytyksille:	vilkkuva punainen = kuitaamaton hälytys kiinteä punainen = kuitattu hälytys vihreä = ei hälytyksiä
Tilauuskoodit:	FX-2025-BASIC (= CPU-yksikkö ilman näyttöä) FX-2025-10.4 (= CPU-yksikkö 10.4" kosketusnäytön kanssa)

4.3 Simatic Manager

Simatic Manager on suunnitteluohjelma. Manager sitoo yhteen työkalut, joita prosessinohjausjärjestelmässä tarvitaan. Manager on modulaarinen ja sitä on helppo muokata tarpeen mukaan lisäämällä siihen kirjastoja tai ohjelmia. Manageriin on integroitu laaja työkalupaketti, jossa käsitellään, arkistoidaan ja doku-

mentoidaan kaikki projektiin liittyvät asiat. Yhdessä ohjelmassa päivitetty tieto päivittyy automaattisesti myös muihin ohjelmiin. Tämä on yhteisen tietokannan ansiota. (26.)

Kirjastot sisältävät valmiita toimilohkoja, symboleja ja faceplateja. Faceplatet ovat PC-valvomossa omiin ikkunoihinsa aukeavia työkaluja, joilla voi monitoroida sekä ohjata yksittäisen komponentin kuten pumpun toimintoja. HW-Configilla voi konfiguroida ja parametrisoida laitteita, kuten automaattiosysteemejä, viestintäkomponentteja ja I/O:ita. ”Process object view” tarjoaa näkymän luotuihin toimintalohkoihin ja hierarkiakuvaan tehtaasta. CFC:ssä (Continuous function chart) esiohjelmoidut blokit liitetään graafisesti toisiinsa automaatiologiikan mukaisesti. CFC tarjoaa hyvän mahdollisuuden toimintojen testaamiseen. SFC-ohjelma (Sequential function chart) tarjoaa perustyökalut sekvenssin tekemiseen. (26.)

4.4 GW-7553-laiteohjelmat

Laitteen mukana toimitetaan kirjava joukko ohjelmia, joista useita tarvitaan käyttöönotossa. TCP-lähetysten etsimiseen sekä yhdyskäytävän laiteohjelmiston päivitykseen käytetään MiniOS7 Utility -ohjelmaa. IP-asetukset laitteeseen ladataan Profibus/Modbus Gateway Utility -ohjelmalla. Virtuaalinen master-laite liikennöinnin testausta varten modbus-väylälle luodaan Modbus Utility -ohjelmasta löytyvällä MBTCP-työkalulla. Kuvankaappaukset sekä tarkemmat ohjeet ohjelmien käytöstä löytyvät liitteestä 2.

5 TYÖN SUORITUS

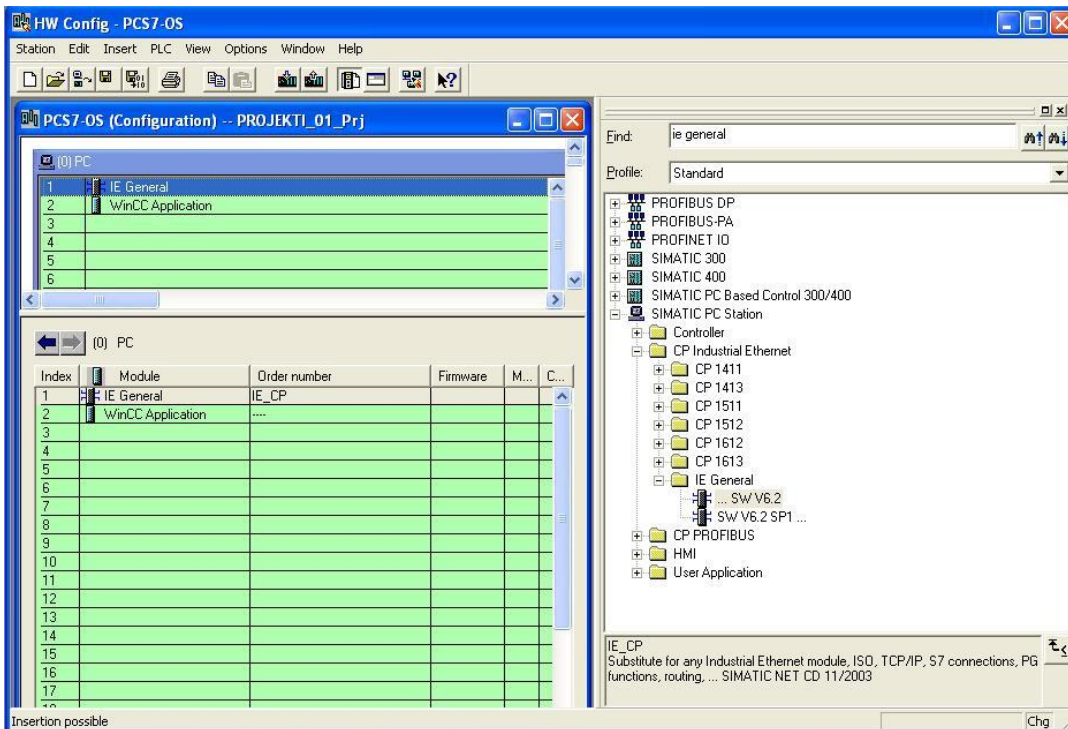
Aluksi, kun yhdyskäytävälaitetta ei vielä ollut päätetty, keskityttiin testausympäristön rakentamiseen. Parhaiten toiminnan testaukseen soveltui tekniikan yksikön automaatiolaboratoriosta löytyvä Siemens S7-400 -sarjan ohjelmoitava logiikka (kuva 23), jota käytetään Simatic Manager -ohjelmistolla.



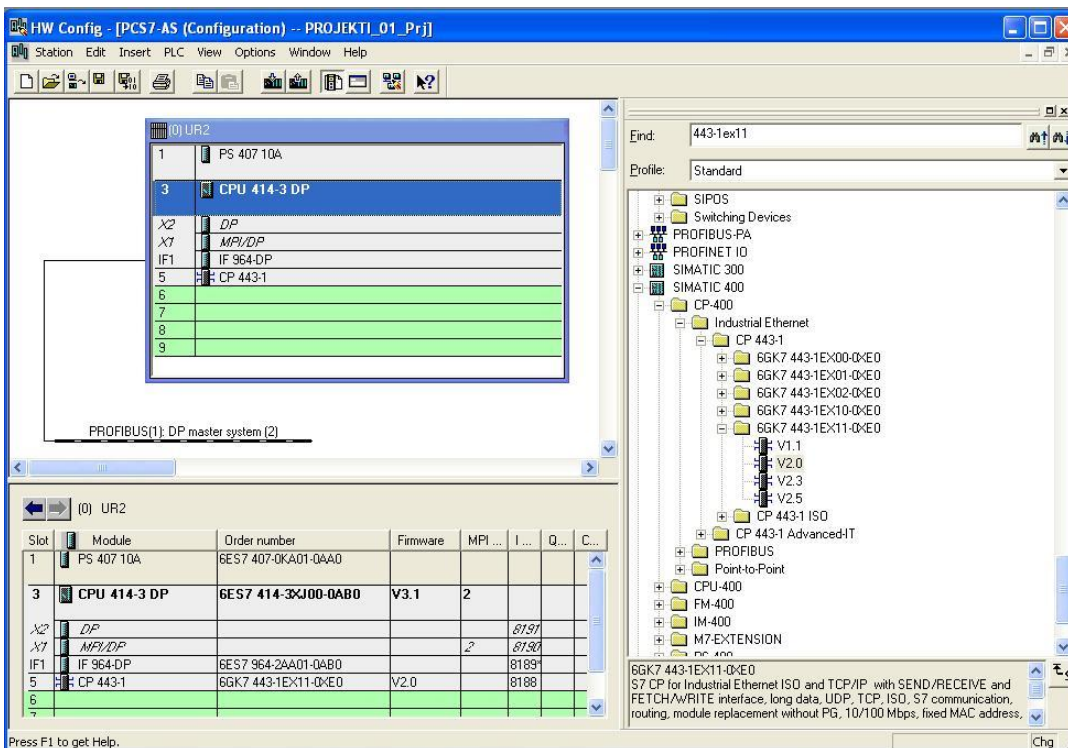
KUVA 23. Siemens S7-400 -sarjan ohjelmoitava logiikka moduuleineen

5.1 Projektin luonti Simatic Managerilla

Projekti luotiin Simatic Managerilla, johon ensin määritettiin käytettävä CPU-moduuli (Central processing unit) ja valittiin projektille nimi. PC-valvomon HW-Configiin lisättiin Ethernet-moduuli, jotta liikennöinti PLC:n kanssa olisi mahdollista Ethernet-verkon ylitse (kuva 24). Lisääminen tapahtuu raahaamalla haluttu laiteajuri kursorilla sille tarkoitettuun paikkaan. Seuraavaksi rakennettiin PLC:n HW-Config-työkalulla (kuva 25) kokoonpano, joka vastaa automaatiolaboratoriosta löytyvää Siemensin laitteistoa. Koska laitteisto on modulaarinen, täytyy laitteita lisätä useita. Moduulien lisäämistä helpottaa niiden kannesta löytyvä sarjanumero, jonka perusteella ne voidaan hakutyökalulla etsiä kirjastosta. Laitteita lisättäessä täytyi niiden asetuksiin määrittää esimerkiksi PC-valvomon ja PLC:n Ethernet-moduulin MAC-osoitteet (Media access control) sekä käytettävät väylät ja niiden nopeudet.



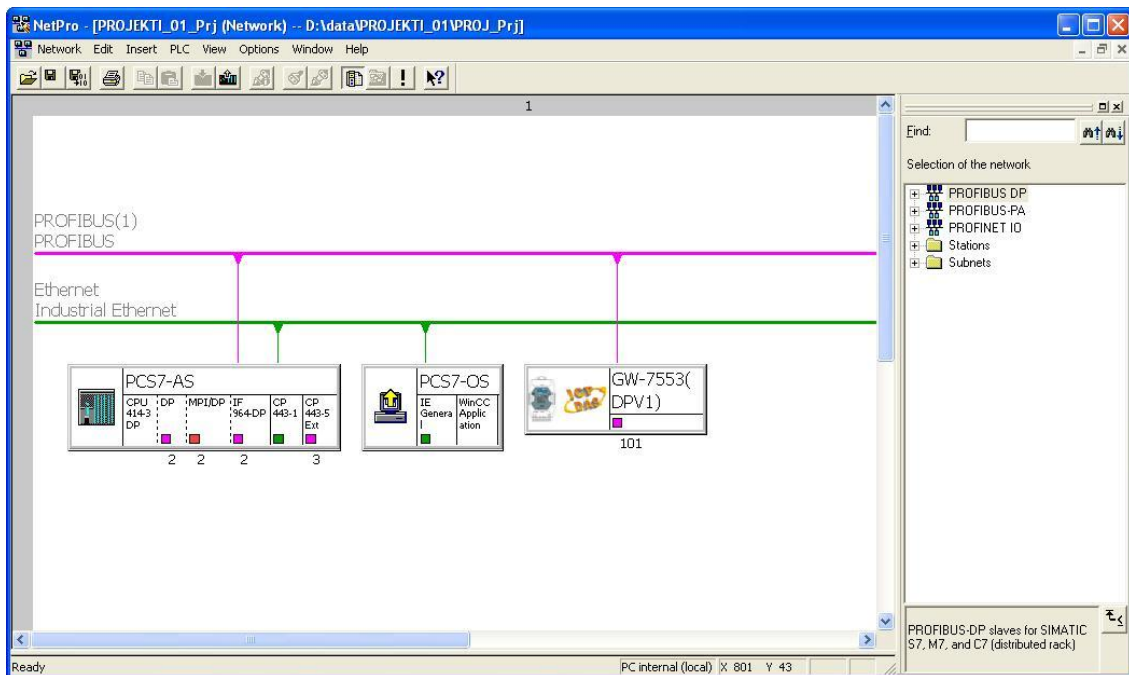
KUVA 24. PC-valvomon HW-Config



KUVA 25. PLC:n HW-Config

Kun laitteiston ajurit oli lisätty, tarkistettiin väyläliitynnät NetPro-työkalulla (kuva 26). NetPro luo visuaalisen kuvan projektin topologiasta. Selkeään kuvaan on

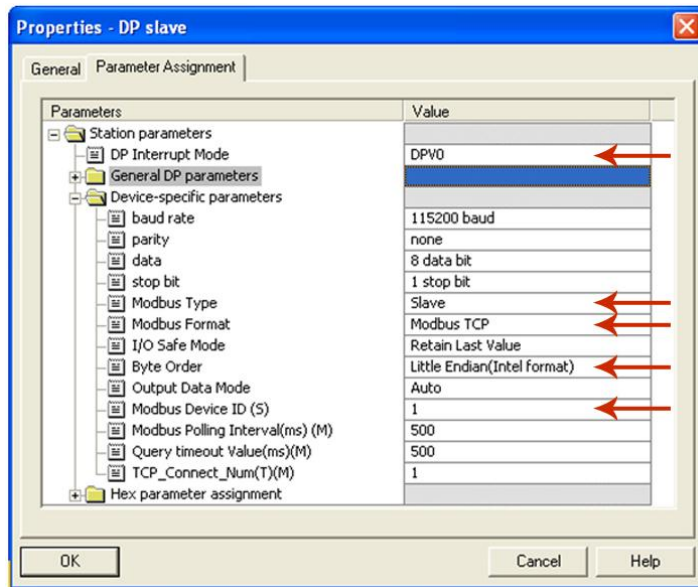
helppoa tehdä vaadittavat muutokset ja korjaukset. Korjausten jälkeen pystytään väylälle lisäämään esimerkiksi käytettävät kenttälaitteet, toistimet ja muunnitimet. Laitteiden lisääminen tapahtuu lataamalla kirjastoon kunkin laitteen valmistajan tarjoama GSD-tiedosto (General station description). Kirjastoon lisäyksen jälkeen voidaan laite asettaa väylälle. Tärkeimpänä asetuksena laitteelle oikean toiminnan varmistamiseksi voisi pitää yksilöllistä joko ohjelmallisesti tai DIP-kytkimellä asetettua Profibus-osoitetta.



KUVA 26. NetPro ja luodun projektin topologia

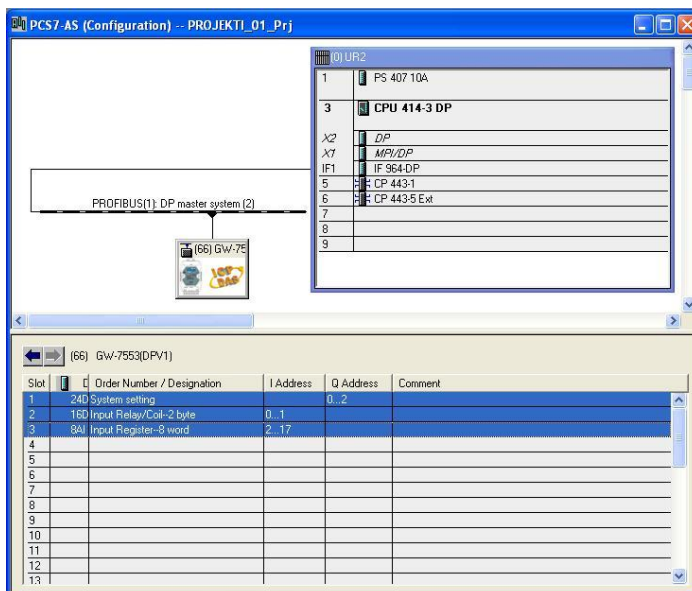
Projektin oltua jo jonkin aikaa valmiina saatiin selvyys käytettävästä yhdyskäytävästä. Valituksi tuli ICP DAS:n valmistama GW-7553. Laitteelle on oma GSD-tiedosto, eli sen lisääminen väylälle on vaivatonta. Laite näkyy väylälle lisättynä kuvassa 26. Yhdyskäytävän Profibus-osoitteeksi on määritetty 101.

Koska jokainen käyttäjä tarvitsee yhdyskäytävää omanlaiseensa tarkoitukseen, täytyy laitteesta löytyä kattavat mahdollisuudet laitekohtaisten asetusten määrittämiseen. Asetukset määritetään väylälle lisätyn laitetiedoston ominaisuuksista (kuva 27).



KUVA 27. GW-7553:n laitekohtaisten asetusten oleelliset kohdat

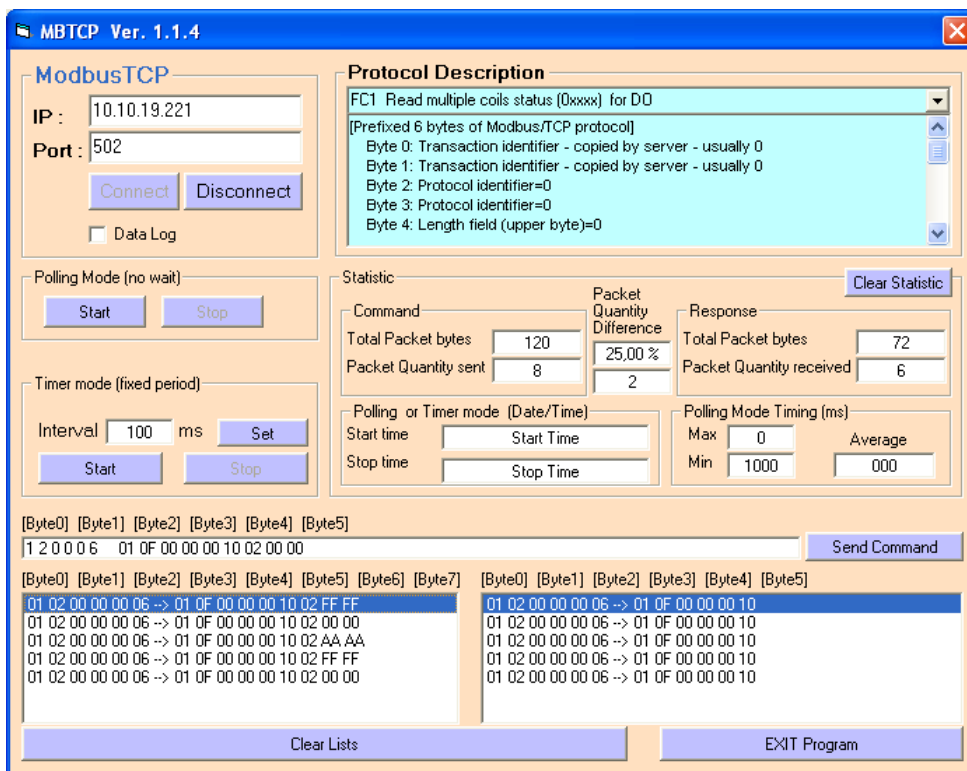
Vaatumukset tarvittaville rekisterityypeille ja niiden pituuksille vaihtelevat käyttökohteen mukaan. Tästä johtuen mahdollisuus valita vain tarvittavat rekisterit on tärkeä ominaisuus työn selkeyden kannalta (kuva 28). Moduulit lisätään laitteen alle PLC:n HW-Configissa valitsemalla ne laitekirjastosta. Moduulit löytyvät hierarkiapuusta yhdyskäytävän alta.



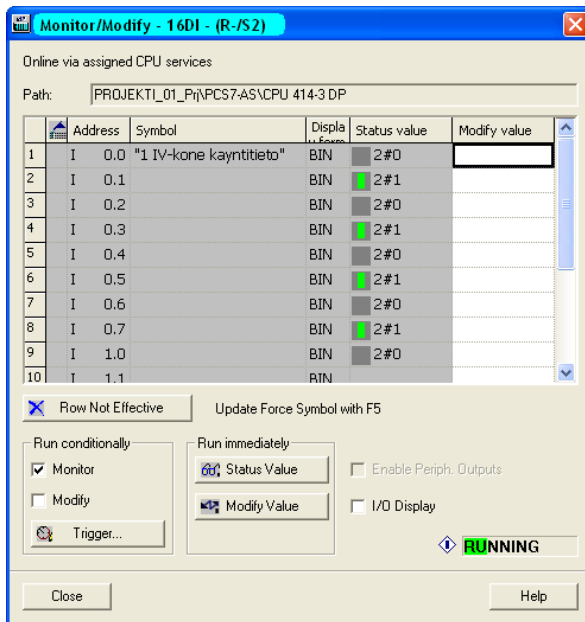
KUVA 28. Yhdyskäytävän rekisterimoduulien lisäys

5.2 Yhteyden testaus virtuaalisella master-laitteella

Yhteys testattiin luomalla kuvan 20 mukaiset kytkennät laitteiden välille. PC-valvomona toimineessa koneessa monitoroitiin yhdyskäytävään ladattujen rekisterien tiloja. MBTCP-ohjelmaa pyörittäneellä PC:llä luotiin virtuaalinen master-laite Modbus-väylälle (kuva 29). Väylälle lähetettiin Ethernetin välityksellä Modbus-viestikehyksiä, jotka GW-7553 muunsi ja välitti Profibus-väylälle. Yhteyden toimiessa sanomiin tuli vastaus. Mikäli viestikehyksessä oli käsketty muuttamaan yksittäisen bitin tai kokonaisen 16-bittisen rekisterin tilaa, näkyi muutos myös PC-valvomossa (kuva 30).



KUVA 29. Virtuaalinen Modbus-master



KUVA 30. Rekisterien monitorointi PC-valvomolla

Kuvassa 30 joka toinen bitti on aktivoitu. Tämä tapahtui lähettämällä virtuaaliselta master-laitteelta komento "01 0F 00 00 00 10 02 AA AA". Komento koostuu yhdeksästä tavusta, joista kukin on esitetty kahdella heksaluvulla. Vasemmalta lukien ensimmäinen tavu määrittelee vastaanottajan Modbus-osoitteen, tässä tapauksessa 01 hex = 01 dec. Toinen tavu on funktiokoodin numero 0F hex = 15 dec (force multiple coils). Seuraavat kaksi tavua määrittävät sen, mistä kohtaa rekisteriä lähdetään kirjoittamaan 0000 hex = 0 dec (kohdasta 0). Kun kirjoituksen aloituskohta on selvillä, kerrotaan kahdella tavulla se, kuinka monen bitin mittainen data on tarkoitus kirjoittaa 0010 hex = 16 dec (16 bittiä). Kun kirjoitettavien bittien määrä on selvillä, kerrotaan yhdellä tavulla datan pituus tavuina 02 hex = 2 dec (2 tavua). Viimeiseksi lisätään kirjoitettava data, jonka määrä riippuu aikaisemmista määrytyksistä. Esimerkitapauksessa lähetetään data AAAA, joka on binäärisessä muodossa 1010101010101010. Tätä binäärimuotoa kutsutaan bittimaskiksi. Vähiten merkitsevä LSB-bitti (Least significant bit) löytyy jonon oikeasta reunasta. LSB-bitti kirjoitetaan ensimmäisenä rekisterin aloituskohtaan 0 (address 0.0). Loput bitit kirjoitetaan järjestyksessä aina MSB-bittiin (Most significant bit) saakka, joka tulee paikkaan 16 (address 1.7). Vastauksena onnistuneesta lähetyksestä saa Modbus-master-laite kaikuna saman viestikehyksen ilman kirjoitettua dataa ja sen pituutta tavuina.

Ohjelmoitaessa tulee ottaa huomioon, että PDU:ssa rekisterien paikkojen numerointi lähtee 0:sta ja Modbus-laitteissa olevat rekisteripaikat alkavat paikasta 1. Käytössä on siis yhden numeron poikkeama.

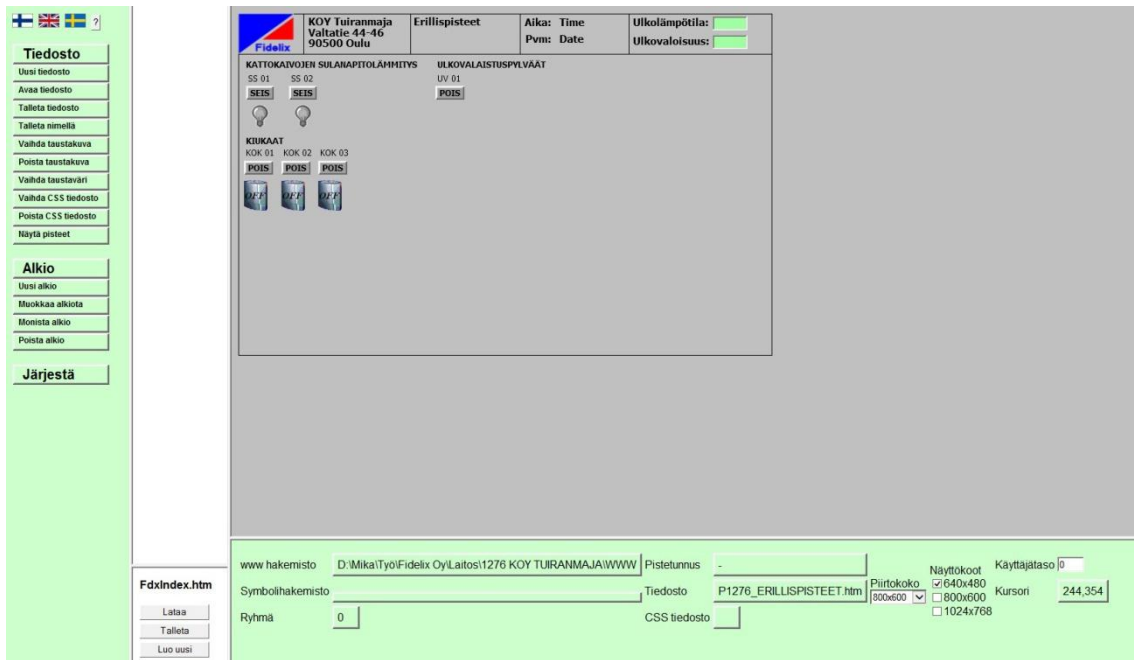
Koska yhdyskäytävän tulee käyttökohteessaan ainoastaan kirjoittaa Profibus-väylälle, testattiin sen tukemat Modbus-funktiokoodit 5, 6, 15 ja 16, jotka liittyvät datan kirjoitukseen. Testattujen funktiokoodien tarkempi kuvaus löytyy taulukosta 6.

5.3 Fidelix alakeskuksen toiminnan testaus

Kun oli saatu varmuus yhdyskäytävän toimivuudesta ja tiedettiin se, mitä yhteyden muodostaminen vaati, aloitettiin alakeskuksen testaus. Ensimmäin määritettiin IP-asetukset (liite 3) sekä portti Modbus/TCP-liikennöintiä varten (liite 4). Jokainen käytettävä yhdyskäytävän rekisterimoduuli määritettiin omana Modbus-laitteena (liite 5). Yksittäinen rekisteri on mahdollista jakaa useampaan osaan.

Modbus-laitteita eli rekisterimoduuleja lisättäessä tuli tietää yhdyskäytävän Modbus-osoite, rekisterin tyyppi, kirjoituksen aloituspaikka, kirjoitusalueen pituus, yhdyskäytävän IP-osoite sekä lähettäjän ja vastaanottajan portit. Mikäli asetukset olivat kohdallaan, ilmoitti alakeskus tästä kirjoittamalla Modbus-laitteen statukseen "Communication OK" (liite 5).

Onnistuneen liikennöinnin varmistuttua asennettiin Fidelixin henkilökunnan kanssa yhteistyössä alakeskuksen lopulliset grafiikat, jotka luotiin Fidelixin HTML-editorilla (FdxHtmlEdit) (kuva 31). Grafiikkakuvat luodaan suoraan htm-muotoon, jotta ne toimisivat selainpohjaisessa ympäristössä. Grafiikkaeditorissa luotuun projektiin lisättiin halutut kuvat symbolikirjastosta. Kuhunkin kuvaan liitettiin siihen kuuluvat pistetunnukset sekä muut halutut ominaisuudet. Pistetunnukset selviävät kunkin laitteiston PI-kaaviosta (liite 6). Kuva alakeskuksiin käyttöön tulleesta käyttöliittymästä löytyy liitteestä 7.



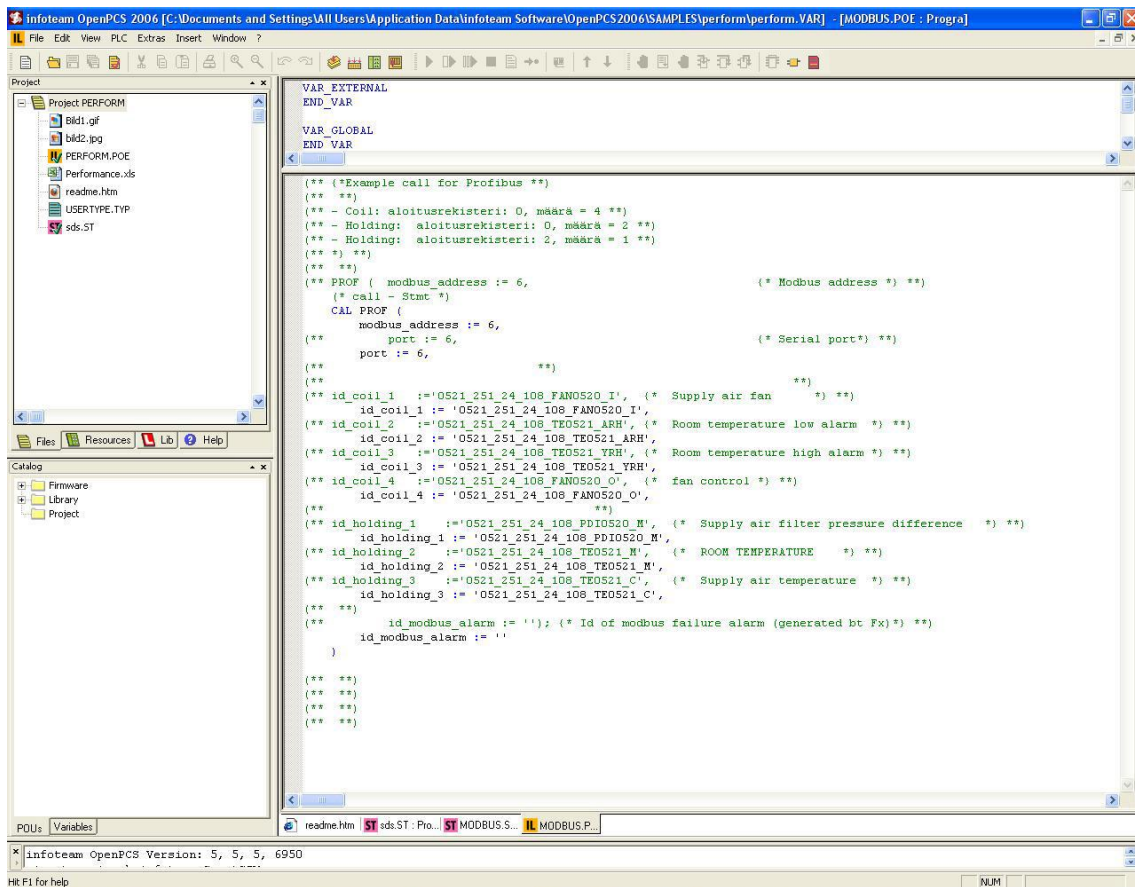
KUVA 31. Fidelixin HTML-editori grafiikkakuvien luomiseen

Fidelixillä on laaja kirjasto valmiita ohjelmapohjia ohjelmointia varten. Kirjastoa käyttämällä projektinhoitajan on helppoa valita tarvitsemansa pohjat ja muokata ne omiin tarpeisiinsa sopiviksi määrittelemällä pisteiden määrät ja osoitteet PI-kaavion mukaisesti. Valmiit pohjat ovat hyvä lähtökohta ohjelmoinnille, jos ei ole siihen erityisen hyvin perehtynyt. Yleisesti käytössä olevat ratkaisut pitävät ohjelmat yhdenmukaisina ja ne nopeuttavat työn etenemistä huomattavasti.

Käytettävän ohjelman sekä funktiolohkojen luominen ei ollut aivan mutkatonta. Ongelmia tuotti se, että yhdyskäytävän kanssa keskustelevalle alakeskukselle tuli ainoastaan kirjoittaa yhdyskäytävän rekistereihin, kun yleensä alakeskus ainoastaan lukee tietoja kenttälaitteilta ja antaa niiden perusteella ohjauskomentoja. Vaadittavat koodin pätkät kuitenkin lopulta löydettiin aikaisemmista projekteista ja mittapisteiden tietojen kirjoitusta rekistereihin päästiin testaamaan. Päänvai-vaai aiheutti pitkään tavujen saapuminen rekisteriin väärässä järjestyksessä, mutta yhdyskäytävän asetuksia hieman muuttelemalla saatiin kirjoitus lopulta onnistumaan sekä 16 että 1 bitin mittaisiin rekisteripaikkoihin.

Sovellusohjelmointiin käytettiin Infoteam OpenPCS -ohjelmointityökalua (kuva 32). Ohjelmalla pystytään luomaan sovelluksia ohjelmoitaville logikoille, ha-

jautetuille järjestelmille ja etäkäytettäville laitteille. Käyttöliittymän vasemmasta laidasta löytyy projektin hierarkiapuu, jonka alta ohjelmat sekä funktiolohkot. Projektin hierarkiapuun alta löytyy funktiokirjasto. Oikea puoli on jaettu kahteen osaan, joista ylempään tehdään ohjelmien muuttujien esittelyt ja määrittelyt. Alempaan kenttään tulee varsinainen ohjelmakoodi. Ohjelman alalaita sisältää ilmoitusosan, jossa näkyy esimerkiksi ohjelmaa testattaessa virheilmoitukset kuvauksineen. Työssä käytettiin ehtolauseisiin perustuvaa ST-tyyppiä (Structured text) ohjelmointia. ST on kehitetty erityisesti ohjelmoitavia logiikoita varten. Esimerkki luodusta funktiolohkosta liitteessä 8. Liitteen 8 funktiota kutsutaan, kun halutaan kirjoittaa 16 bittiä pitkä mittaus tieto sanamuotoiseen rekisteriin.



KUVA 32. OpenPCS:n ohjelmointiympäristö

5.4 Laitteiston valmistelu lähetystä varten

Uruguayihin ei lähde ketään paikan päälle suorittamaan käyttöönottoa, vaan se tehdään laitteistoon väliaikaisesti asennettavan 3G-modeemin välityksellä. Modeemilla saadaan yhteys alakeskusten sekä yhdyskäytävien selainpohjaiseen

käyttöliittymään. Tämän vuoksi laitteisto tuli konfiguroida mahdollisimman valmiiksi ennen sen lähetystä tilaajalle. Kaikkiin yhdyskäytäviin asennettiin uusin laiteohjelmisto ja IP-asetukset määritettiin Fidelixiltä saatujen ohjeiden mukaisesti. Koska alakeskukset ja niiden ohjaamat laitteistot ovat identtisiä ja ne toimivat toisistaan erillään, määritettiin kaikkiin kolmeen yhdyskäytävään samat asetukset.

Niistä toimenpiteistä, jotka on mahdollista suorittaa vasta, kun laitteisto liitetään tilaajan Honeywell-automaatiojärjestelmään, valmistettiin kirjalliset ohjeet (liite 2). Ohjekirjasta löytyy oma lukunsa GW-7553:n lisäämiseksi väylälle. Lisäksi laiteohjelmiston päivitykselle ja IP-asetusten määrittämiselle löytyy omat lukunsa.

6 POHDINTA

Opinnäytetyön tarkoituksena oli tutkia Fidelix Oy:lle Profibus–Modbus-liitynnän toteutusmahdollisuudet ICP DAS GW-7553 -yhdyskäytävällä. Työhön kuului testauslaitteiston toteutus ja konfigurointi tekniikan yksikön automaatiolaboratorioon, yhdyskäytävän testaus virtuaalisella Modbus-master-laitteella sekä lopullinen rajapintatestaus FX-2025-alakeskuksella. Mikäli liityntä todettaisiin toimivaksi, tuli Uruguayihin lähtevän laitteiston mukaan valmistaa ohjekirja yhdyskäytävien käyttöönottoa varten.

Aihe oli haastava ja lopulta myös mielenkiintoinen. Alkutaipaleella opinnäytetyön aihe tuntui vieraalta ja vaikeasti lähestyttävältä, mutta tietotason karttuessa asiat selkenivät ja osa-alueet alkoivat hitaasti hahmottua. Tutkittava asia oli itseni lisäksi täysin tuntematon myös Fidelixin henkilökunnalle. Esimerkiksi alakeskukseen tarvittavan ohjelman luominen työn loppuvaiheessa oli pitkälti yritykseen ja erehdykseen perustuvaa työskentelyä. Haastavuutta lisäsivät myös erittäin tiukka aikataulu, avun saannin vaikeus työskenneltäessä koululla sekä se, ettei käytettävissä ollut Honeywellin ohjelmoitavaa logiikkaa Profibus-liitynnällä.

Työn aikana eteen tuli monenlaisia haasteita ja ongelmia. Yhtenä suurimpana ongelmana olivat yhdyskäytävän valmistajan manuaalit, joissa oli puutteita sekä virheellisiä ohjeita muun muassa laiteohjelmiston päivityksestä. Koska firmware asennettiin puutteellisesti työn alkuvaiheessa, ei selainpohjainen käyttöliittymä toiminut ollenkaan aivan viime metreille saakka. Tämän ja aikataulun tiukkuuden vuoksi ei etäkäyttöä ehditty testaamaan ollenkaan. Myös alakeskuksen Ethernet-portit lakkailivat toimimasta eikä aina voinut olla varma siitä, oliko konfigurointi epäonnistunut vai oliko kyseessä ohi menevä häiriö, joka kuittaantuisi laitteiston uudelleenkäynnistyksellä.

Varmuus yhdyskäytävän toiminnasta kuitenkin saatiin ja kaikki vaaditut toiminnot onnistuttiin toteuttamaan alakeskuksen ja PLC:n välillä. Käyttöönottoa helpottamaan valmistui 15-sivuinen englanninkielinen ohjekirja (liite 2).

Ennen työn aloitusta eivät opinnäytetyöni osa-alueet olleet itselleni tuttuja ja juuri siksi aihe oli niin opettavainen. Tällä tietopohjalla on hyvä jatkaa tutustumista kenttäväyliin ja niihin liittyviin laitteisiin.

LÄHTEET

1. Fidelix Oy 2013. Yleisesite. Saatavissa:
http://www.fidelix.fi/documents/FI/Fidelix_Yleisesite.pdf. Hakupäivä 9.4.2013.
2. Piikkilä, Veijo – Sahlstén, Toivo 2006. ST-Käsikirja 21. Kiinteistöjen tiedon-
siirtoväylät. Espoo: Sähköinfo Oy.
3. Kaario, Kimmo 2002. TCP/IP-verkot. Jyväskylä: Docendo Finland Oy.
4. Ruohonen, Mika 2002. Tietoturva. Jyväskylä: Docendo Finland Oy.
5. Granlund, Kaj 2007. Tietoliikenne. Jyväskylä: WSOYpro/Docendo-tuotteet.
6. Jaakohuhta, Hannu – Lahtinen, Tapani 1997. Tietoliikenneverkot. Tehokäyt-
täjän opas. Espoo: Suomen Atk-kustannus Oy.
7. Introduction to Modbus TCP/IP 2005. Saatavissa:
[http://www.acromag.com/sites/default/files/Acromag_Intro_ModbusTCP_765
A.pdf](http://www.acromag.com/sites/default/files/Acromag_Intro_ModbusTCP_765_A.pdf). Hakupäivä 18.4.2013.
8. Microsoft 2000. Verkkoteliikka+. Helsinki: Oy Edita Ab.
9. Hakala, Mika – Vainio, Mika 2005. Tietoverkon rakentaminen. Jyväskylä:
Docendo Finland Oy.
10. Systeemitekniikan laboratoriotyöt. Harjoitustyö n:o 6. Johtokyvyn mittaus ja
säätö. Saatavissa: <http://cc oulu.fi/~posyswww/opiskelu/sytelabrat/tyo6.pdf>.
Hakupäivä 7.4.2013.
11. Johansson, Erik 2000. Säätö- ja mittaustekniikka. Iisalmi: IS-VET.
12. Profibus ajaa vaconeja. Saatavissa:
<http://old.vacon.com/Default.aspx?Id=461144>. Hakupäivä 22.4.2013.
13. Profibus overview 2013. Saatavissa:
<http://www.profibus.com/technology/profibus/overview/>. Hakupäivä 4.5.2013.

14. Profibus System Description 2013. Saatavissa:
<http://www.profibus.com/nc/download/technical-descriptions-books/downloads/profibus-technology-and-application-system-description/display/>. Hakupäivä 22.4.2013.
15. Pyykkö, Tero 2005. Profibus VS. perinteinen signaalin siirto. Automaatioväylä nro 1. s. 36–38.
16. Basic Slide Set 2013. Saatavissa:
<http://www.profibus.com/download/presentations-logos/>. Hakupäivä 21.4.2013.
17. Introduction to Profibus DP 2002. Saatavissa:
http://www.diit.unict.it/users/scava/dispense/II_270/ProfibusIntroduction.pdf.
Hakupäivä 22.4.2013.
18. Installation Guideline for Planning 2010. Saatavissa:
<http://www.profibus.com/download/installation-guide/>. Hakupäivä 21.4.2013.
19. Aarrelampi, Hannu 2009. Kaapelia väylään. Saatavissa:
http://www.cc.puv.fi/~ot/ISA0604%20Datasiirron%20perusteet/Profibus/Pro maint_8-09_25-27.pdf. Hakupäivä 22.4.2013.
20. Hietanen, Tero 2009. Profibus väyläanalyysi. Saatavissa:
http://www.tekniikka.oamk.fi/~terohi/Labrat/C_analyysi.doc. Hakupäivä 22.4.2013.
21. Modbus messaging on TCP/IP implementation guide V1.0b 2006. Saatavissa:
http://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf. Hakupäivä 17.4.2013.
22. Modbus 2013. Saatavissa: <http://en.wikipedia.org/wiki/Modbus>. Hakupäivä 23.4.2013.

23. Modbus application protocol specification V1.1b3 2012. Saatavissa:
http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf.
Hakupäivä 17.4.2013.
24. GW-7553 Profibus/Modbus TCP Gateway User's Manual 2012. Saatavissa:
http://www.icpdas-europe.com/uploads/tx_v4icpdascatalog/128086/web/icpdas/manual-gw-7553-b_cr.pdf. Hakupäivä 29.4.2013.
25. FX-2025 Alakeskus 2007. Saatavissa:
http://www.fidelix.fi/documents/tuki/Fidelix_FX2025_Fin.pdf. Hakupäivä 25.4.2013.
26. Kuusisto, Teemu. Siemens Simatic PCS 7– ohjelmiston toiminnot. Saatavissa:
<http://aut-bscw.hut.fi/pub/bscw.cgi/d252105/PCS7%20toiminnot.pdf>.
Hakupäivä 28.4.2013.

LIITTEET

Liite 1. Lähtötietomuistio

Liite 2. Käyttöohjeet yhdyskäytävän käyttöönottoon

Liite 3. Kuvankaappaus alakeskuksen IP-asetuksista

Liite 4. Kuvankaappaus alakeskuksen porttiasetuksista

Liite 5. Kuvankaappaus alakeskuksen Modbus-laite -asetuksista

Liite 6. Tuloilmakoneen PI-kaavio

Liite 7. Kuvankaappaus alakeskuksen graafisesta käyttöliittymästä

Liite 8. Esimerkki kirjoitusta varten luodusta funktiolohkosta

OULUN SEUDUN
AMMATTIKORKEAKOULU



TEKNIKAN YKSIKKÖ
KOTIKANTIE 1, 90250 OULU
www.oamk.fi

LÄHTÖTIETOMUISTIO

Työn tiedot	Tekijä ¹ Tommas Laakkola +9ratu0@students.oamk.fi 0505288773	Tilaaaja ² Fidelix Oy
Tilaaajan yhdyshenkilö ja yhteystiedot ³ Janne Tolonen 050 910 1271		
Työn nimi ⁴ Modbus - Profibus - rajapinta		
Työn kuvaus ⁵ Selvitystyö siitä miten fidelixin modbus-väylää käyttävät alakeskukset saadaan liitettyä ICP DAS:in valmistamalla gw-752-yhdyskäytävälaitteella profibusväylään.		
Työn tavoitteet ⁶ Tuottaa sellainen dokumentaatio, jolla vastaavan rajapintaliitoksen käyttöönotto onnistuu ensi kerralla vaivattomammin.		
Tavoiteaikataulu ⁷ Työllä on kiire ja selvitykset yhteensopivuudesta pyritään tekemään mahdollisimman pian riippuen miten käytettävien laitteiden valmistajat pystyvät ne toimittamaan.		
Päiväys ja allekirjoitukset ⁸ 01/03/13 Tekijän allekirjoitus <i>Tommas Laakkola</i>		01/03/13 Tilaaajan allekirjoitus <i>Janne Tolonen</i>
<ol style="list-style-type: none"> 1. Tekijän nimi, puhelinnumero ja sähköpostiosoite. 2. Työn teettävän yrityksen virallinen nimi. 3. Sen henkilön nimi ja yhteystiedot, joka yrityksessä valvoo työn suoritusta. 4. Työn nimi voi olla tässä vaiheessa työnimi, jota myöhemmin tarkennetaan. 5. Työ kuvataan lyhyesti. Siinä esitetään muun muassa työn tausta, lähtötilanne ja työssä ratkaistavat ongelmat. 6. Esitetään lyhyesti ja selvästi työn tavoitteet. 7. Esitetään projektin tavoiteaikataulu. Silloin, kun työllä on välitavoitteita, myös ne merkitään aikatauluun. Tavoiteaikataulun ja oppilaitoksen yleisaikataulun perusteella tekijä laatii oman aikataulunsa. 8. Lähtötietomuuisto päivätään ja sen allekirjoittavat tekijä ja tilaaajan yhdyshenkilö. 		

Using ICP DAS GW-7553 gateway with Fidelix FX-2025 substation

IP settings have been set and the latest firmware was installed before shipping the gateways. Only GSD file needs to be installed and configured to make Modbus–Profibus connection to work.



2013



CONTENTS:

1. Hardware specifications of the device	2
2. Layout of the system	3
3. Installing and configuring the GSD file	4
4. Updating new firmware to gateway.....	8
5. Configuring IP settings of the GW-7553	12
6. Troubleshooting.....	15

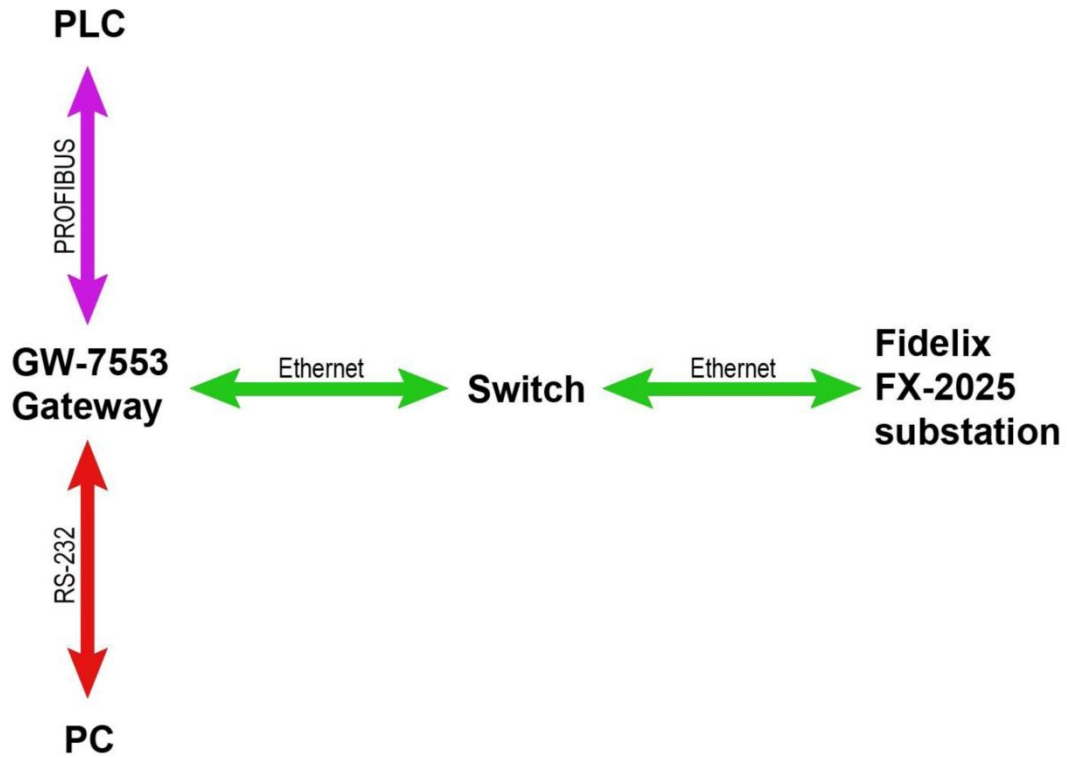
1. HARDWARE SPECIFICATIONS OF THE DEVICE

Hardware	
CPU	80186, 80 MHz or compatible
SRAM/Flash/EEPROM	512 KB / 512 KB / 16 KB
Watchdog	Watchdog IC CPU built-in
ESD Protection	4 kV class A
PROFIBUS Interface	
Controller	Profichip VPC3+C
Transceiver	ADI ADM2486
Connector	9-pin female D-Sub
Baud Rate (bps)	9.6 k, 19.2 k, 45.45 k, 93.75 k, 187.5 k, 500 k, 1.5 M, 3 M, 6 M, 12 M
Transmission Distance (m)	Depend on baud rate (for example, max. 1200 m at 9.6 kbps)
Isolation	3000 V _{DC} for DC-to-DC, 2500 Vrms for bus-to-logic
Protocol	DP-V0 & DP-V1
UART Interface	
COM	RS-232 (For updating firmware and configuration)
COM Connector	5-pin screwed terminal block (TxD, RxD, RTS, CTS, GND)
Baud Rate (bps)	2400, 4800, 9600, 19200, 38400, 57600, 115200
Data bit	7, 8
Stop bit	1, 2
Parity	None, Even, Odd
Ethernet Interface	
Controller	10/100Base-TX Ethernet Controller (Auto-negotiating, Auto_MDIX)
Connector	RJ-45 with LED indicator
LED	
Round LED	PWR LED, RUN LED, ERR LED
Power	
Power supply	Unregulated +10 ~ +30 V _{DC}
Protection	Power reverse polarity protection, Over-voltage brown-out protection
Power Consumption	2.5 W
Mechanism	
Installation	DIN-Rail
Dimensions	72mm x 33mm x 119mm (W x L x H)
Environment	
Operating Temp.	-25 ~ 75 °C
Storage Temp.	-30 ~ 85°C
Humidity	5 ~ 95% RH, non-condensing

PICTURE 1. Specifications of the device

2. LAYOUT OF THE SYSTEM

GW-7553 is a Profibus slave and Modbus slave device. RS-232 connection is only needed when updating firmware or changing the IP settings of the GW-7553.



PICTURE 2. Layout of the system

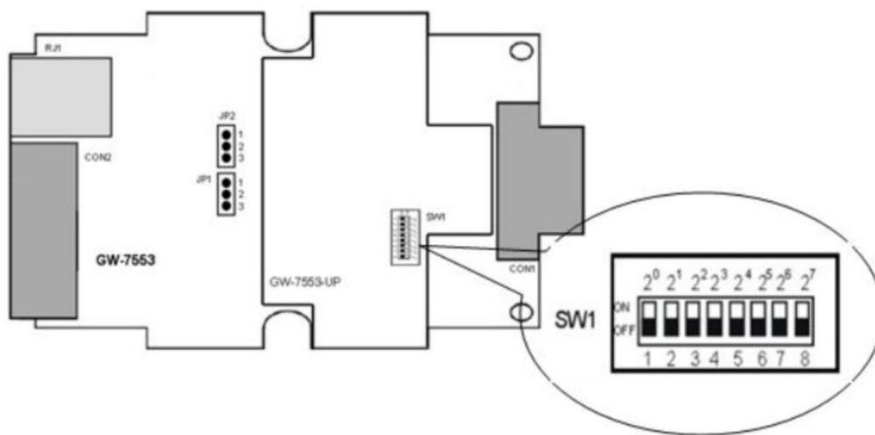
3. INSTALLING AND CONFIGURING THE GSD FILE

GSD file of the GW-7553 can be downloaded at:

http://ftp.icpdas.com/pub/cd/fieldbus_cd/profibus_gateway/gw-7553/gsd/

Example pictures used in this part of the instruction manual are from Simatic PCS7 but same principle works with all manufacturers.

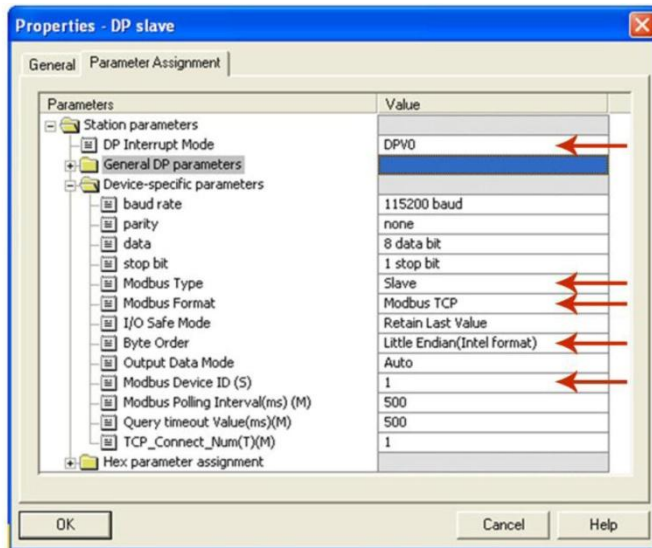
1. Profibus address of GW-7553 can be set by using the DIP switch. DIP switch can be found by opening the cover of GW-7553 (picture 3). The DIP switch covers a range from 0 to 255. Valid address range of a Profibus station spans from 0 to 126.



PICTURE 3. Locating the DIP switch

2. First install the GSD file to your hardware library and add it to your bus using same Profibus address that was set with the DIP switch. GSD file of GW-7553 contains several settings (picture 4), ones that need to be modified from their default values are:

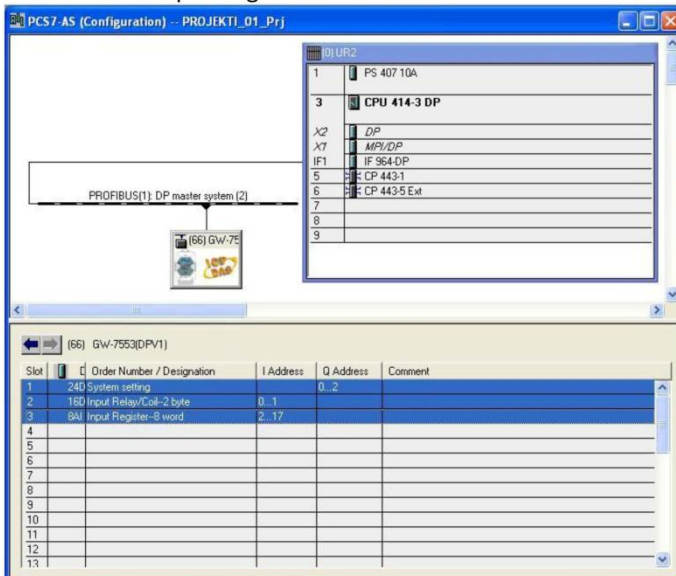
- DP Interrupt Mode: from DPV0 to same version that is used in GW-7553's bus.
- Modbus Type: from Master to Slave
- Modbus Format: from Modbus RTU to Modbus TCP
- Byte order: from Little Endian(Intel format) to Big Endian(Motorola Format)
- Modbus Device ID: from 1 to 6



PICTURE 4. GSD settings

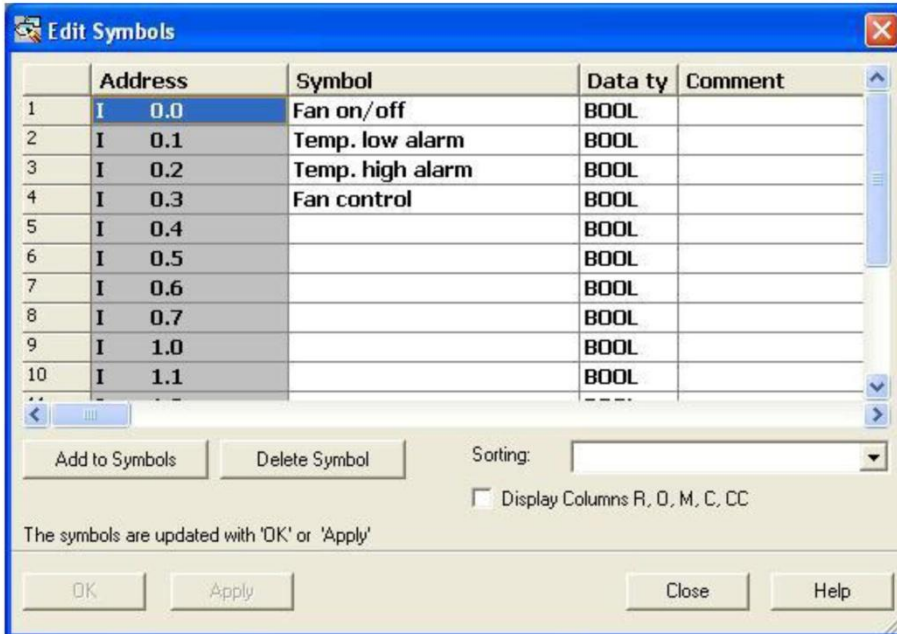
3. You need to add three modules that will be uploaded to GW-7553. Modules are in hardware library under the GW-7553's GSD file. Modules needed and the slots used for them are shown in picture 5.

- Slot 1: System setting
- Slot 2: Input Relay/Coil--2 byte
- Slot 3: Input Register--8 word

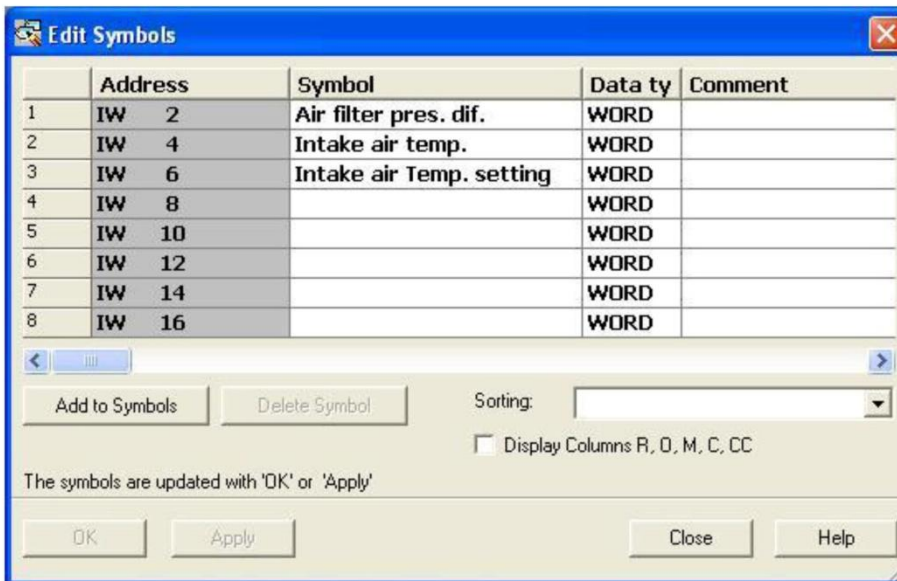


PICTURE 5. Adding modules to gateways slots

4. Values from Fidelix substation will be written in following order to modules (picture 6 & 7). Because values are always written to modules in same order "Address" field can be determined by user.

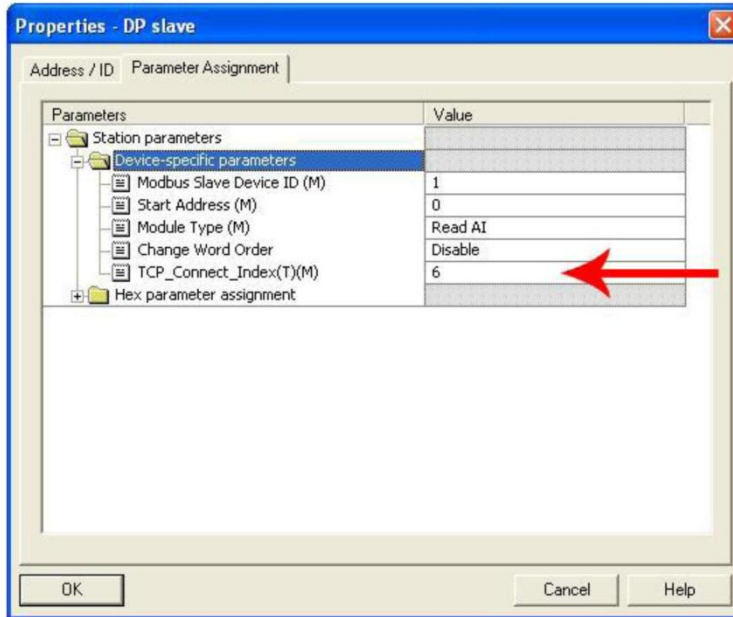


PICTURE 6. Input Relay/Coil--2 byte

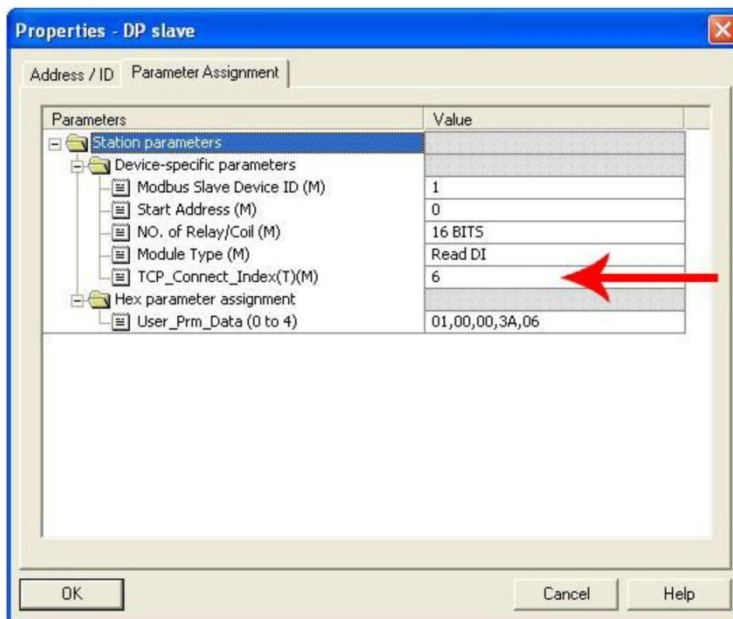


PICTURE 7. Input Register--8 word

5. Modules in slot one and two need their settings to be changed. In both TCP_Connect_Index(T)(M) needs to be changed from 1 to 6 (picture 8 & 9).



PICTURE 8. Input Register--8 word properties



PICTURE 9. Input Relay/Coil--2 byte properties

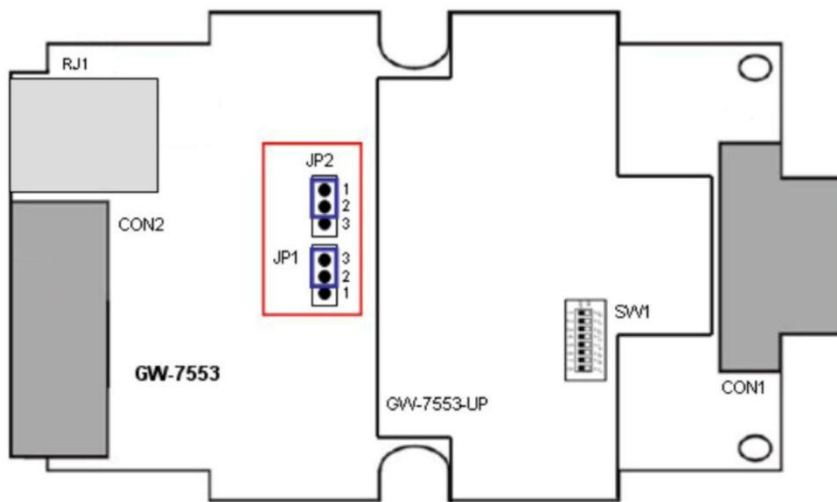
Now GW-7553 gateway is ready to be downloaded into PLC

4. UPDATING NEW FIRMWARE TO GATEWAY

Newest firmware can be found at:

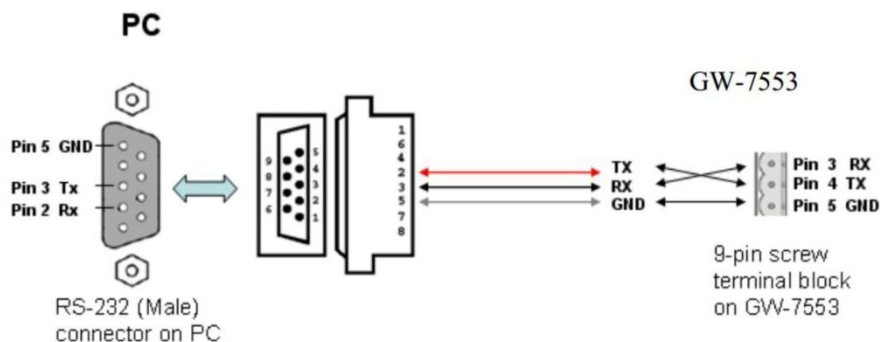
ftp://ftp.icpdas.com.tw/pub/cd/fieldbus_cd/profibus/gateway/gw-7553/firmware/

1. Turn the power off. Open the cover to access two jumpers inside the gateway (picture 10). Close the initial pin 2, 3 (JP1) and flash protection pin 1, 2 (JP2) with jumper.



PICTURE 10. Setting the jumpers for updating new firmware

2. Connect RS-232 cable between PC's com port and gateway (picture 11). Turn the power back on.

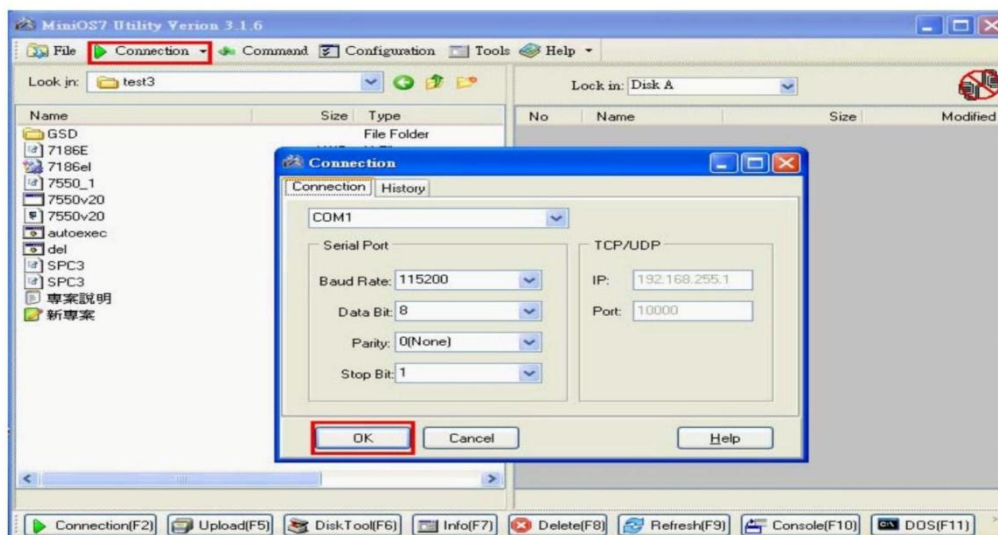


PICTURE 11. Pinout for the 9-pin Female D-sub and 3-wire RS-232 cable

3. Run MiniOS7 utility. Software can be downloaded at:

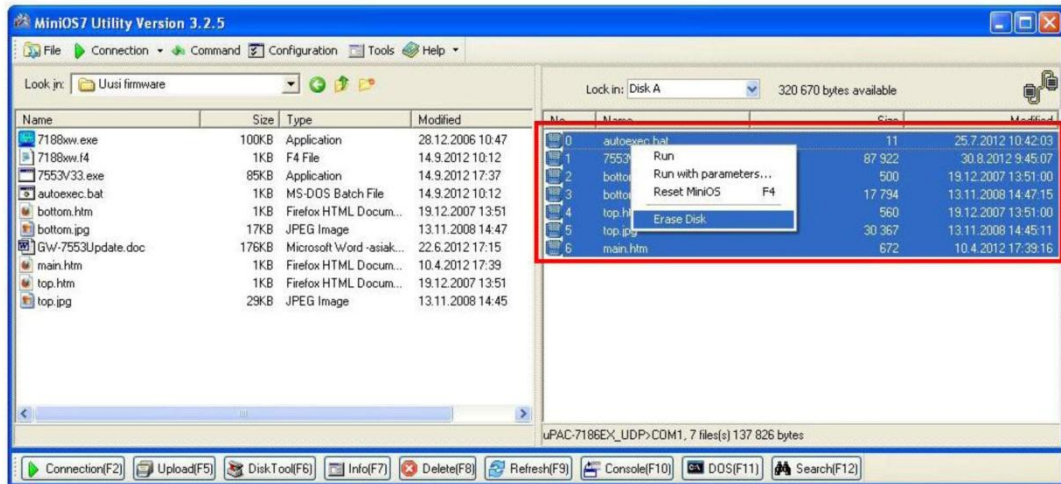
http://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/minios7/utility/minios7_utility/

4. Connect MiniOS7 Utility and the GW-7553 by clicking the connection button. Com port and "Serial Port" settings need to be set before connection can be established (picture 12). Set the com port to use same port you connected your RS-232 cable at. Rest of the settings need to match ones made in GSD setup. Press "OK".



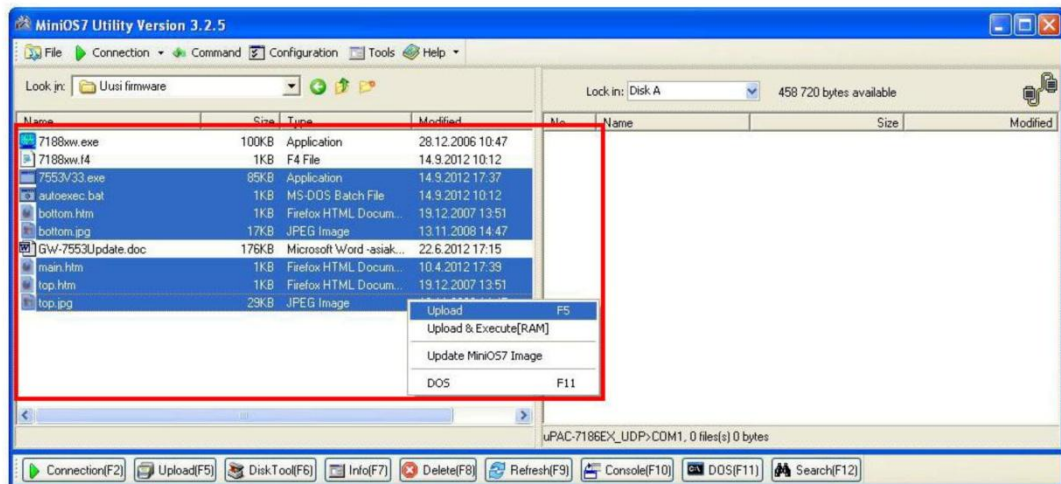
PICTURE 12. MiniOS7 Connection

5. Right side browser is used to show files inside the gateway. Erase old 7553vxx.exe & autoexec.bat from gateway (picture 13).



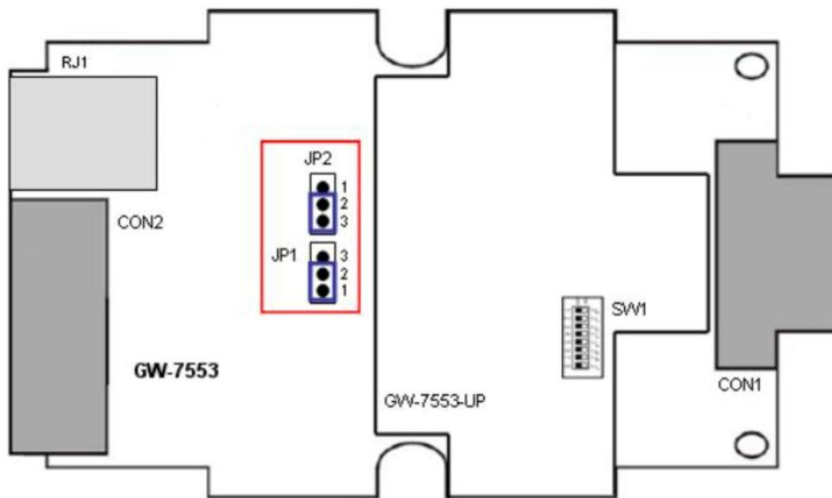
PICTURE 13. Removing old firmware in MiniOS7

6. Use left side browser to locate new firmware files in your PC. You will need 7 files shown in picture 14. Select them and upload.



PICTURE 14. Uploading new firmware to gateway in MiniOS7

7. After all files have been uploaded close, the program and turn the power off from the gateway. Close the initial pin 2, 3 (JP1) and flash protection pin 1, 2 (JP2) with jumper (picture 15). Close the cover of GW-7553.



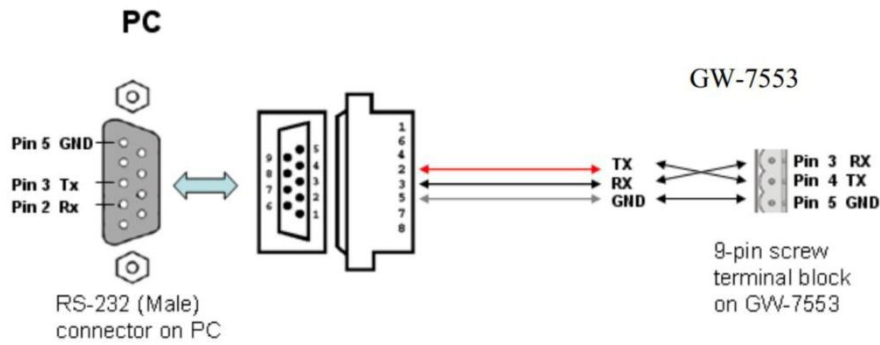
PICTURE 15. Setting the jumpers for normal use

New firmware has now been installed

5. CONFIGURING IP SETTINGS OF THE GW-7553

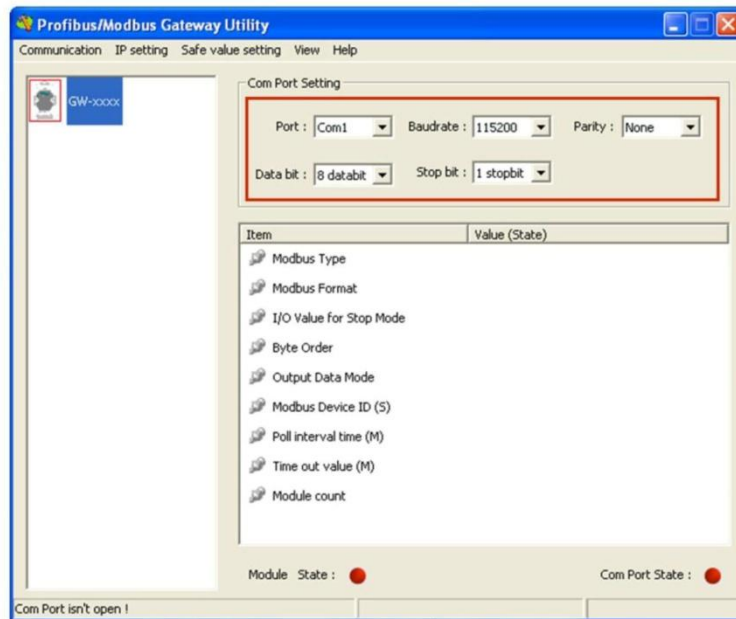
Configuring IP settings is done with Profibus/Modbus Gateway Utility. Utility can be downloaded at:

ftp://ftp.icpdas.com/pub/cd/fieldbus_cd/profibus/gateway/gw-7553/utilities/



PICTURE 16. Pinout for the 9-pin Female D-sub and 3-wire RS-232 cable

1. Run the utility program. Set the com port to use same port you connected your RS-232 cable at. Baudrate, parity etc. must be set the same way they are set in the GSD setup.



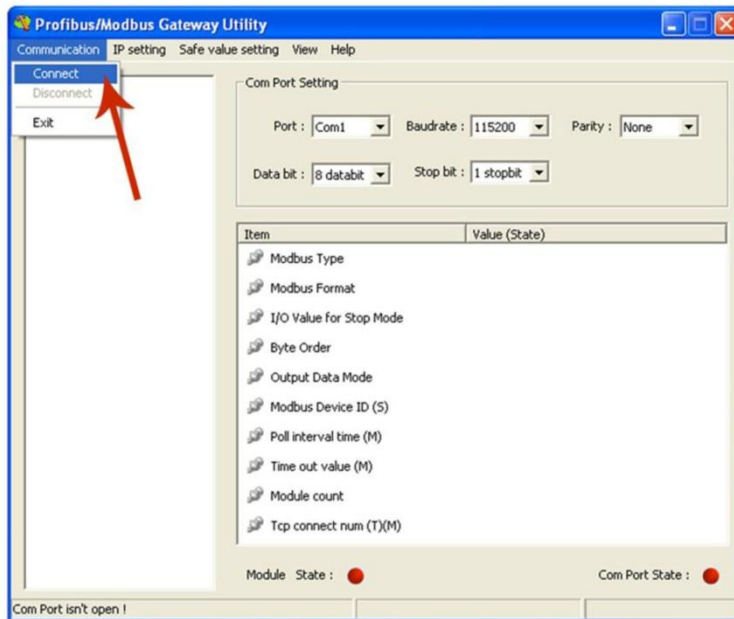
PICTURE 17. Configuring settings in the utility program for connection

2. Before the connection, make sure that the RS-232 cable has been connected between the PC and the Gateway (picture 16). RUN LED of the GW-7553 needs to be on and the switch in back of the GW-7553 must be set at setting mode (picture 18).



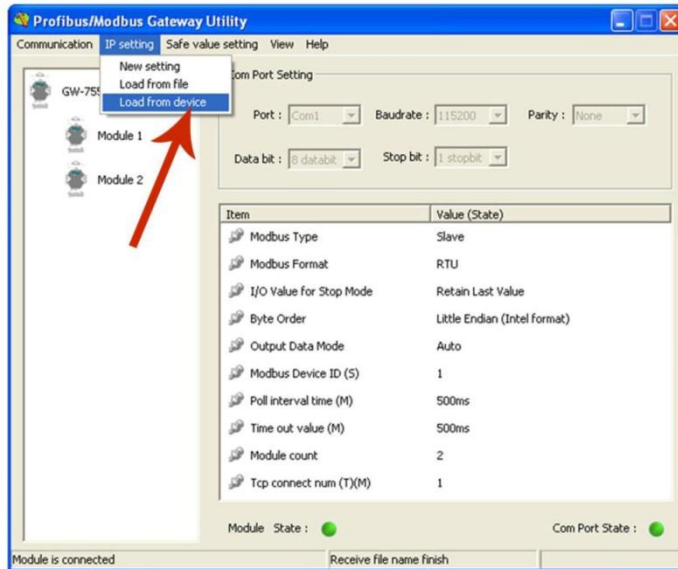
PICTURE 18. Run LED and Normal/Setting switch

3. Now go to "Communication" and select "Connect" (picture 19).



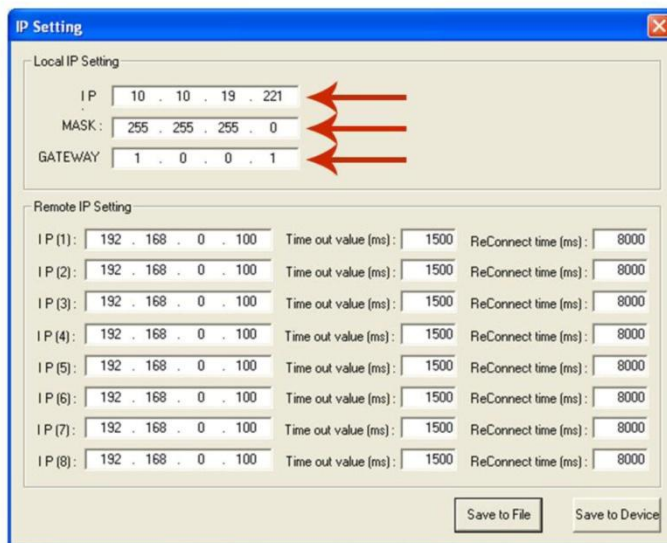
PICTURE 19. Making the connection in utility program

4. If everything was set correctly module and com port state indicators should turn green. Next select "IP setting" and "Load from device" (picture 20).



PICTURE 20. Loading IP settings from gateway

5. In our case only local IP settings need to be changed. After correct IP, mask, and gateway have been set press the "Save to Device" button (picture 21). Close the utility program.



PICTURE 21. Saving new IP settings to gateway

IP settings have now been set

6. TROUBLESHOOTING

Input data is coming in but byte order is wrong.





1. Try changing the GSD-setting.

Byte order: from Big Endian(Motorola Format) to Little Endian(Intel format).

2. There is also "Change Word Order" setting in each module.

Default setting is disabled but enabling it could also resolve the problem.

Alarms

 Finnish  English  Swedish  Dutch

Settings

IP addresses

Change

IP addresses

Ethernet 2 - Active IP address:

Automatic address (DHCP)

Default IP-address:

Default gateway:

Name server (DNS):

Ethernet 1 - Active IP address: 10.10.19.222

Automatic address (DHCP)

Default IP-address:

Default gateway:

Name server (DNS):

Local IP-address:

History server IP address:



Remote server IP address:



Domain login used

Public IP address: Updated ???

Activate changed settings now

Alarms

 Finnish  Swedish

 English  Dutch

Settings

Ports

Change

Port settings

Port 0	MBUS									
Port 1	Modem									
Port 3	Modbus		9600	N82	57600	Send delay (ms)	0			
Port 4	Free			N82	19200					
Port 5	Free			N82	19200					
Port 6	Modbus TCP					Send delay (ms)	250			
Port 7	Free									
Port 8	Free									
Port 9	Free									
Port 10	Free									

M-BUS send delay (ms) 5000

Debug GSM modem

Alarms
Programming
Modbus devices

Fr 19.04.2013 10:01:17 Modbus devices Update

06.006	1=COILS	0	1	Port=6	Communication Ok
06.006	3=HOLDING	0	2	Port=6	Communication Ok

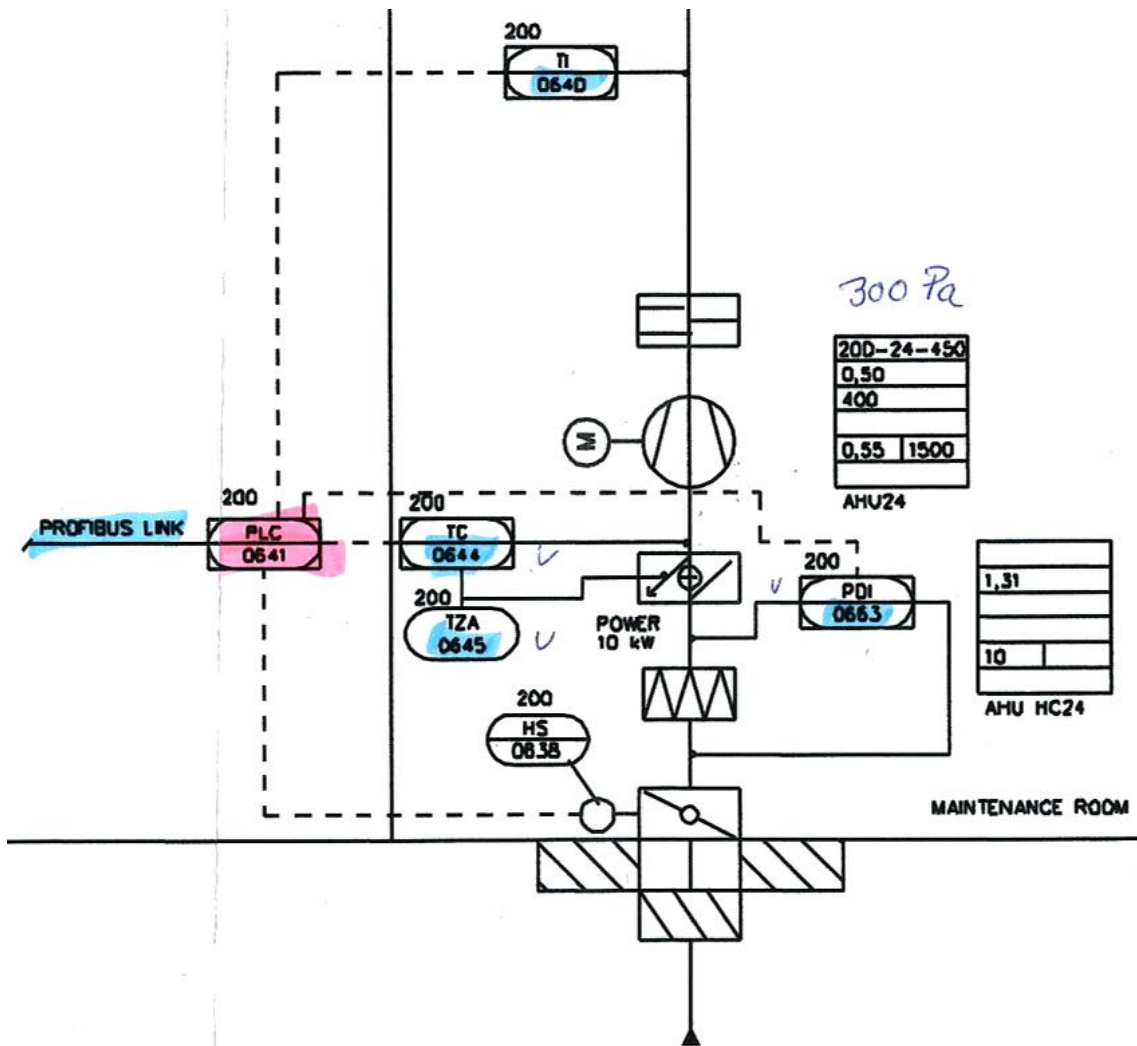
Port number 6 Address of modbus device 6


Register type	3=HOLDING REGS (READ=3,WRITE=16)	TCP/IP module	<input checked="" type="checkbox"/>
Start register	0	IP address	10.10.19.221
Register count	2	IP port	502
Debug	<input type="checkbox"/>	Send only	<input type="checkbox"/>

Messages total	42
Errors now	0
Errors max	0
Errors total	0
Last error code	Ok
Receive timeout (ms)	500

Clear error counters

Add Change Delete

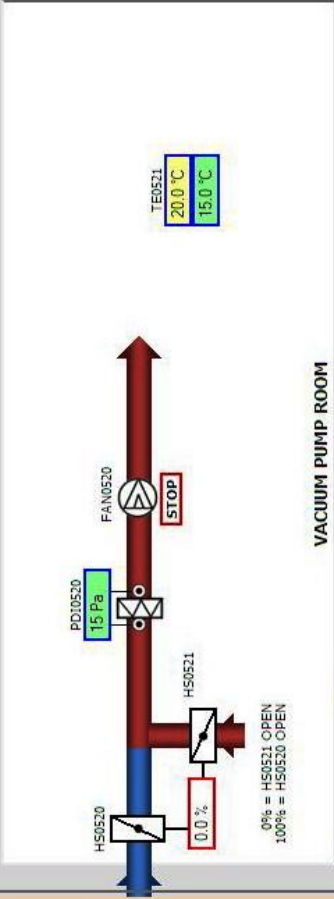




Montes del Plata
Stora Enso

PLC-0521
Evaporation

Date: 19.04.2013
Time: 10:08:09



VACUUM PUMP ROOM

0% = H50521 OPEN
100% = H50520 OPEN

■ = MEASUREMENT ■ = SETPOINT VALUE □ = ANALOG OUTPUT VALUE

ADDITIONAL INFORMATION:

Emergency stop

Fire alarm

STATUS: STOP: Timetable

TIMETABLE: STOP

MAIN PAGE
SETTINGS

Change point value

Cancel
Auto
Change

0521_251_24_108_PDI0520_M

Supply air filter pressure difference

1	2	3	15
4	5	6	
7	8	9	
0	.	+/-	C

```
(* Zero division not allowed *)
rMultiply := rMultiplier;
if (rMultiply = 0.0) then
    rMultiply := 1.0;
end_if;

rFX2020 := GetAnalogPointF(Name:=id_String);
(* WORD_TO_INT conversion needed because of possible negative values *)
iReg:=WORD_TO_INT(RegValue);
rReg:=INT_TO_REAL(iReg) / rMultiply;

IF id_String <> '' THEN

    (* The value ALWAYS FROM FX *)
    rReg := rFX2020;

END_IF;

(* REAL_TO_INT conversion needed because of possible negative values *)
iReg := REAL_TO_INT(rReg * rMultiply);
HoldingRegF_Write := INT_TO_WORD(iReg);
```