



Petri Waara

MODBUS-KNX-RAJAPINTA

MODBUS-KNX-RAJAPINTA

Petri Waara
Opinnäytetyö
Kevät 2012
Automaatiotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Automaatiotekniikka, Projektointi

Tekijä: Petri Waara

Opinnäytetyön nimi: Modbus–KNX-rajapinta

Työn ohjaaja: Heikki Kurki

Työn valmistumislukukausi ja -vuosi: Kevät 2012

Sivumäärä: 26 + 15

Tämän insinöörityön aiheena oli tutkia Modbus- ja KNX-väylien välistä rajapintaa. Tavoitteina oli selvittää millaisilla välineillä rajapinta oli muodostettavissa ja toteuttaa rajapinta Fidelix-järjestelmän kanssa. Tilaajalle tehtiin ohjelmapohja ja ohjeet, joita voidaan käyttää tulevilla projekteilla helpottamaan käyttöönottoa.

Insinöörityö päädyttiin toteuttamaan Intesis Software SL:n IntesisBox-yhdyskäytävälaitteella. Yhdyskäytävälaitteeseen konfiguroidaan KNX-väylän ryhmäosoitteet niiltä osin, joita halutaan ohjata, sekä Modbus-väylän rekisterinumerot Fidelix-järjestelmän käyttöön. Fidelix-järjestelmä kommunikoi yhdyskäytävälaitteen kautta KNX-väylän kanssa ja kirjoittaa tarvittaessa käyttäjän valinnan mukaan KNX-väylälle halutun valojenohjauksen tilan.

Materiaalin hankinnan yhteydessä selvisi, että Intesis Software SL on ainoa tällaisen yhdyskäytävälaitteen toimittaja. Rajapinnan muodostaminen TCP/IP-protokollan avulla olisi varmasti ollut mahdollista, mutta tässä työssä siihen ei otettu kantaa. Insinöörityön tuloksena tilaajayritys sai käyttöönsä modulaarisen ohjelman, jolla voidaan ottaa käyttöön halutut KNX-väyläominaisuudet, esimerkiksi valojenohjaus tai huonelämpötilan säätö.

Asiasanat:

Fidelix, Modbus, KNX, EIB, Intesis, automaatio

ALKULAUSE

Haluan kiittää vaimoani tämän työn loppuun saattamisesta. Keinoina käytettiin uhkailua, kiristystä sekä lahjontaa.

Oulussa 24.5.2012

Petri Waara

SISÄLLYS

TIIVISTELMÄ.....	3
ALKULAUSE.....	4
SISÄLLYS.....	5
LYHENTEET.....	6
1 JOHDANTO.....	7
2 MODBUS-VÄYLÄ FIDELIX-JÄRJESTELMÄSSÄ.....	9
3 KNX-VÄYLÄ.....	11
4 INTESISBOX-YHDYSKÄYTÄVÄLAITE.....	13
4.1 Uuden projektin luonti.....	14
4.2 Tiedonsiirtoasetukset.....	14
4.3 KNX-väylän ryhmäosoitteet ja Modbus-väylän rekisterit.....	15
4.4 Tietojen lataus Intesis-yhdyskäytävälaitteelle.....	17
4.5 Yhteyksien testaaminen.....	19
5 FIDELIX-ALAKESKUKSEN OHJELMOINTI.....	20
5.1 Lähtötilanne.....	20
5.2 Ohjelman soveltaminen.....	20
5.3 Alakeskuksen Modbus-laitteet.....	21
6 ASENNUKSET JA TESTAUKSET.....	23
7 YHTEENVETO.....	25
LÄHTEET.....	26
LIITTEET	
Liite 1 Pelkistetty järjestelmäkaavio	
Liite 2 Yleisohjelma	
Liite 3 Kohteen ohjelma	

LYHENTEET

b/s	Tiedonsiirtonopeus, bittiä sekunnissa
CRC	Cyclic Redundancy check, tiedonsiirron virheen tarkistus
CSMA/CA	Carrier Sense Multiple Access With Collision Avoidance, tietoliikenteen siirtotien varausjärjestelmä
EIB	European Installation Bus, kuuluu KNX:ään
I/O	INPUT/OUTPUT, järjestelmän tulot ja lähdöt
IV-KONE	Ilmanvaihtokone
KNX	Väylä, joka täyttää standardien ISO/IEC (14543), CENELEC (EN50090) ja CEN (13321) vaatimukset
OpenPCS	Infoteam ohjelmointityökalu, käyttää standardia IEC 61131-3
RS232/485	Sarjaväyläliityntä
RTU	Remote Terminal Unit, väyläpohjainen tiedonsiirtoprotokolla
USB-RS485	Tiedonsiirtokaapeli, jossa toisessa päässä USB-liitin ja toisessa RS485-sarjaväyläliitin

1 JOHDANTO

Opinnäytetyö tehtiin Fidelix Oy:lle. Opinnäytetyö oli osa työprojektina olevan lomamökin taloteknisten järjestelmien automatisointia. Automatisointiin kuului muun muassa toimilaitteiden asennukset ja kytkennät, valvonta-alakeskuksen kytkentä ja ohjelmointi sekä Fidelix-järjestelmän käyttöönotto.

Kohteeseen tehtiin pakettityyppinen IV-koneen aikaohjelmaohjaus, lattialämmityksen säätö, palo- ja turvajärjestelmät sekä KNX-järjestelmän avulla toteutettu valojenohjaus. Normaalitilanteessa KNX-väylän valojenohjauksen liitäntä toiseen järjestelmään on toteutettu käyttäen perinteistä I/O:ta. Kohteeseen oli suunniteltu valojen ohjaus käyttäen Fidelix-alakeskuksen Modbus-väylää. Opinnäytetyöni oli muodostaa toimiva ratkaisu Modbus- ja KNX-väylien välille.

Väyläpohjaisen valojenohjauksen etuja ovat kaapelointikustannusten pienentyminen sekä valvonta-alakeskuksen relelähtöjen säästö. Väyläohjaus helpottaa mahdollisia muutostöitä, koska ei tarvitse tutkia kaapeleita, vaan voidaan suoraan väylältä muuttaa haluttu valoryhmä toimimaan halutulla tavalla.

Yksinkertaisuudessaan opinnäytetyöhön tarvittiin väyläsovittimella yhdistetyt KNX- ja Modbus-väylät, alakeskuksen väyläkytkentä sekä ohjelmat. KNX-järjestelmän puolella tehtiin valoryhmäohjelmoinnit sekä valojen ja katkaisijoiden asennukset ja kytkennät.

Opinnäytetyön aihe tuntui sopivan haastavalta ja erittäin mielenkiintoiselta. Se poikkesi huomattavasti aikaisemmista työtehtävistä, joissa käytettiin valmiita Fidelixin pääkonttorilla tehtyjä väyläohjelmia. Yleensä valmista väyläohjelmaa käytetään taajuusmuuttajakäyttöjen käyttöönotossa.

Tällainen tehtävä oli todella mukava ja antoisa. Vantaalta tuli tähänkin väyläohjelma, mutta siinä oli jossain virhe, sillä IntesisBox-yhdyskäytävälaitteen

konfigurointiohjelma LinkBoxMB antoi virheilmoituksen testattaessa väylän toimivuutta.

Työ eteni seuraavien vaiheiden mukaisesti:

1. KNX- ja Modbus-väyliä tutkittiin valmiiden dokumenttien sekä Internetin lähteiden avulla.
2. Ohjelma täydennettiin ja testattiin KNX-demo-laukulla.
3. Ohjelma muutettiin soveltumaan kohteeseen.
4. Kohteeseen tehtiin asennukset sekä testattiin Fidelix-järjestelmän toiminnallisuus.
5. Suoritettiin yhteistestit KNX-väylän kanssa.

2 MODBUS-VÄYLÄ FIDELIX-JÄRJESTELMÄSSÄ

Fidelix-järjestelmä käyttää Modbus-väylää alakeskuksen ja I/O-moduulien välisessä kommunikoinnissa. Tiedonsiirtonopeutena käytetään yleensä 57600 bittiä sekunnissa, mutta vaihtoehtoina löytyy 9600, 19200 sekä 38400 b/s. (1.)

Fidelix-järjestelmässä käytetään RTU-tyyppistä Modbus-väylää, jossa 8-bittinen tavu koostuu kahdesta 4-bittisestä heksadesimaaliluvusta. Tieto lähtee pakettina, jossa on aloitus- ja lopetuskohdat, jolloin vastaanottava laite tietää aloittaa ja lopettaa lähetyksen oikein. Lisäksi tällä tavalla saadaan rikkinäisestä paketista virhetieto. (2, s. 16.)

Modbus-väylältä luetaan tietoa seuraavilla rekisterityypeillä:

- 01 Read Coil Status
- 02 Read Input Status
- 03 Read Holding Registers
- 04 Read Input Registers
- 05 Force Single Coil
- 06 Preset Single Register
- 07 Read Exception Status. (2, s. 33.)

Fidelix-järjestelmässä käytetään näistä kuutta ensimmäistä lukemaan ja kirjoittamaan väylällä tietoa, esimerkiksi huonelämpötilamittaus tai jäähdytyslaitteen antama tilatieto. Työssä käytettiin pelkästään '03 Read Holding Registers' -rekisterityyppiä ohjaamaan valoryhmiä. (1.)

Modbus-väylässä laitteille jaetaan osoitteet väliltä 1–255. Portteja on käytettävissä 9, joten laitteita saadaan yhteen alakeskukseen jopa 2 295 kappaletta. Fidelix-järjestelmässä luodaan Modbus-laite, jonka tietoihin lisätään laiteosoite, portti, aloitusrekisteri ja peräkkäisten rekistereiden lukumäärä.

Tämän jälkeen rekistereiden tiedot ovat luettavissa OpenPCS-ohjelmalla, minkä jälkeen tiedot voidaan tulostaa grafiikalle käyttäjän tulkittavaksi. (1.)

Muiden kuin Fidelix-järjestelmän omien Modbus-laitteiden tiedon käsittelyyn joudutaan luomaan erillinen ohjelmallinen Modbus-laite. Laitteelle annetaan syötteiksi väylän porttinumero, fyysisen laitteen Modbus-osoite, aloitusrekisteri ja luettujen rekistereiden lukumäärä sekä käytetty rekisterityyppi. Tämän jälkeen väylän tiedot ovat luettavissa ohjelmalla.

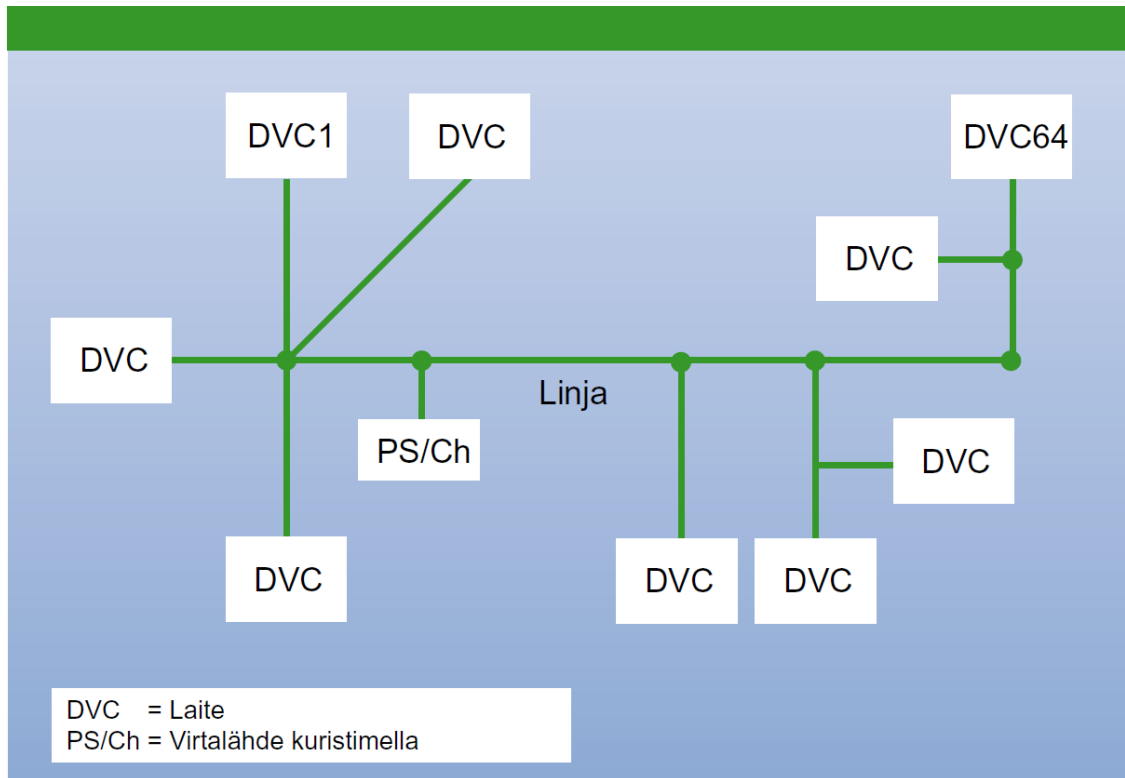
3 KNX-VÄYLÄ

KNX-väylä on kehitetty 1990-luvun alussa takaamaan sähköasennusten muun muassa turvallisuuden ja mukavuuden kasvavat vaatimukset. EIB-väylätekniikka muodostaa nykyisen KNX-järjestelmän ytimen. (3, s. 10.)

KNX-järjestelmässä voidaan käyttää kolmea eri tiedonsiirtotapaa: väyläkaapelia, sähköverkkoa ja/tai radioverkkoa. Yleisin näistä on väyläkaapeli, jossa käytetään kierrettyä parikaapelia tiedon ja käyttöjännitteen siirtämisessä toimilaitteille. Suomessa harvinaista sähköverkon kautta tapahtuvaa tiedonsiirtoa voidaan miettiä saneerauskohteissa, joissa kaapeleiden veto saattaa tuoda ongelmia. Tällöin siis käytetään olemassa olevia pienjännitekaapeleita. Tiedonsiirtoa sähköverkossa käytetään varsin yleisesti esimerkiksi Saksassa. Radioverkon kautta kulkevaa tiedonsiirtoa käytetään myös saneerauksissa ja tiloissa, joissa kaapelin veto ei onnistu. (3, s. 25.)

Käytettäessä kierrettyä parikaapelia tiedonsiirtotapana voidaan tiedonsiirtonopeutena käyttää 9 600 bittiä sekunnissa kun taas käytettäessä voimalinjaa tiedonsiirtonopeudeksi saadaan vain 1200 bittiä sekunnissa. Radioverkoissa voidaan käyttää 16 384:ää bittiä sekunnissa. (4, s. 2–3.)

KNX-väylähierarkia koostuu linjoista ja alueista. Linjat voivat käyttää esimerkiksi tähtiverkkotopologiaa (kuva 1) ja linjat yhdistetään alueiksi suorilla väylillä. Rengasverkkotopologiaa ei käytetä, vaikka tällä tavalla saataisiin estettyä tiedonsiirron katkeaminen väyläkaapelin katketessa. (3, s. 25.)



KUVA 1. KNX-väylän linja

Tiedot liikkuvat väylässä peräkkäin. Tästä johtuen linjassa liikkuu vain yksi tieto kerrallaan. Väylässä käytetään CSMA/CA-yhteysmenetelmää, joka tarkoittaa siirtotien varausmenetelmää. Siinä laite lähettää ennen varsinaista tietoa varaussignaalin, joka varaa väylän omaan käyttöönsä estäen törmäykset. (3, s. 29.)

4 INTESISBOX-YHDYSKÄYTÄVÄLAITE

Tässä työssä pyrittiin saamaan kaksi eri rekisterikantaa käyttävää väylää kommunikoimaan keskenään (liite 1). Tällaiseen tarvitaan aina joko ohjelmointia tai laitteistoa. Tässä työssä käytettiin Fidelix-järjestelmän ja KNX-valojenohjausjärjestelmän välissä Intesis Software SL:n IntesisBox-yhdyskäytävälaitetta (kuva 2). Laite kommunikoi molempien väylien kanssa sille konfiguroitujen asetusten mukaan ja välittää tiedon toisesta väylästä toiselle.

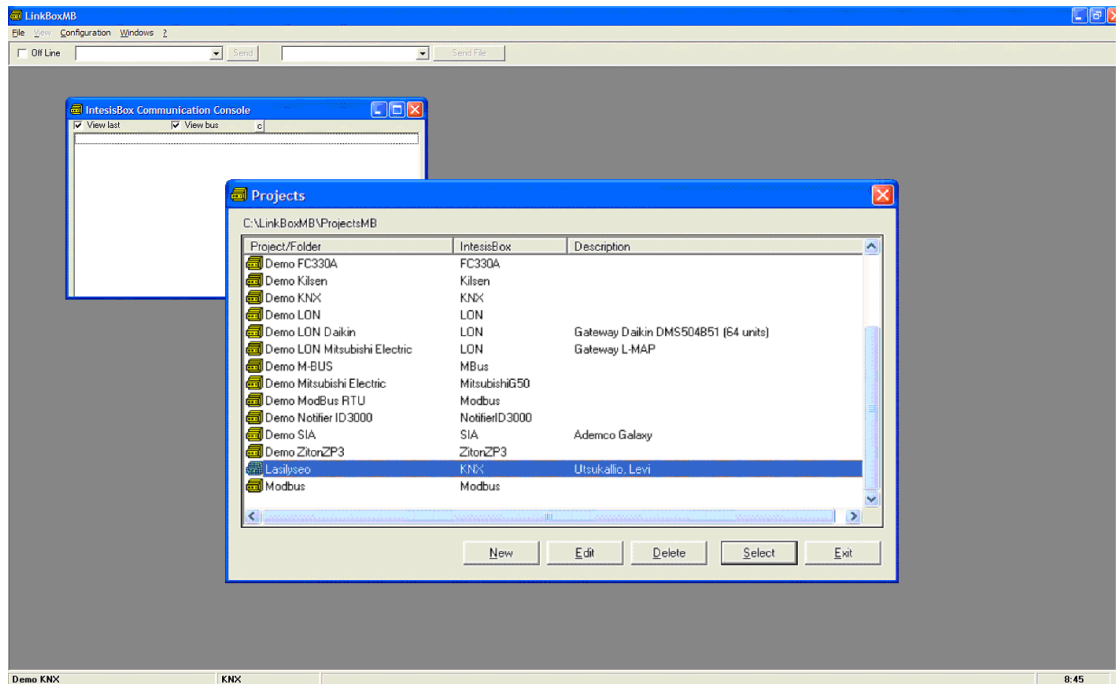
IntesisBox-yhdyskäytävälaitteen ohjelma, LinkBoxMB, on hyvin yksinkertainen käyttää ja konfiguroida, vaikkakin konfigurointi on työlästä, sillä jokainen piste pitää erikseen luoda ja valita sen ominaisuudet. Pienellä kehityksellä Intesis Software SL voisi helpottaa pisteiden luontia, esimerkiksi toteuttamalla kopiointi/liittämisominaisuuden tai jopa mahdollistaa Excel-taulukko-ohjelmassa luodun taulukon tuonnin konfigurointiohjelmaan.



KUVA 2. IntesisBox-yhdyskäytävälaitte

4.1 Uuden projektin luonti

Projekti luodaan File/Projects-valikon kautta. Kuvassa 3 näkyy tehdasvalmisteisia demo-ohjelmia monille eri väylille. Alimpana on Vantaalla tehty Modbus-projekti ja sen yläpuolella kohteeseen tehty Lasilyseo-projekti.

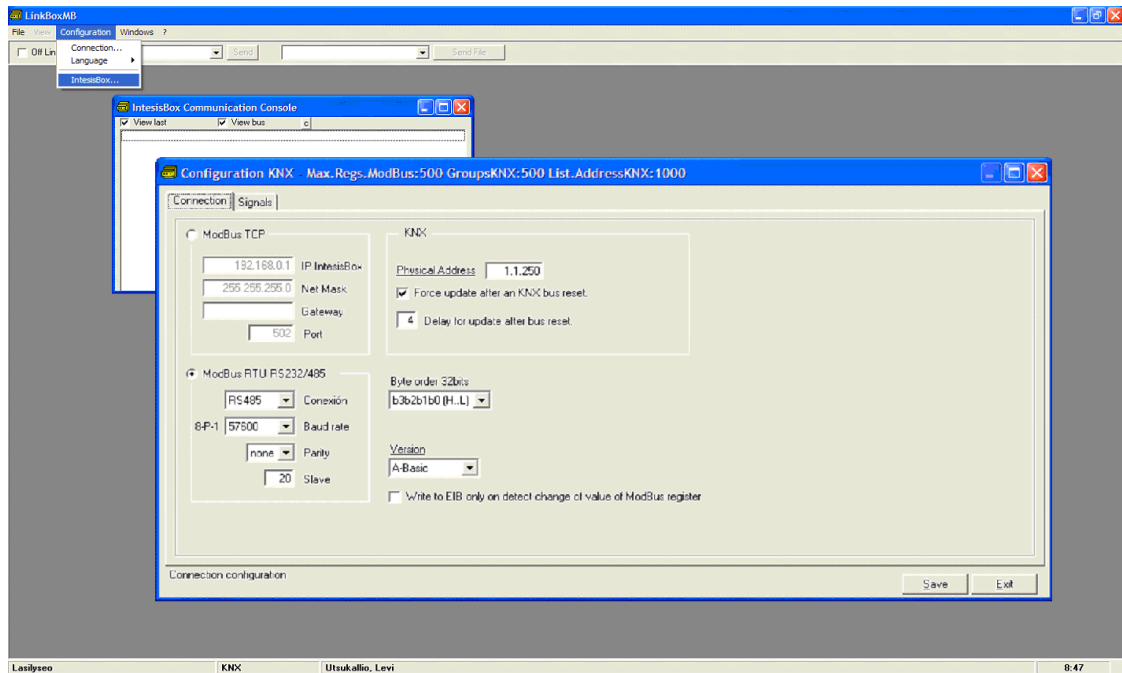


KUVA 3. LinkBoxMB -ohjelman Projekti-ikkuna

4.2 Tiedonsiirtoasetukset

Projektin luonnin jälkeen konfiguroidaan LinkBoxMB-ohjelma vastaamaan kohteen tietoja. Valikosta valitaan Configuration/IntesisBox..., jolloin avautuu ikkuna (kuva 4). Valitaan kohta 'ModBus RTU RS232/485', minkä jälkeen ensimmäisestä alasetusvalikko valitaan RS485 sekä toisesta 57 600 b/s tiedonsiirtonopeudeksi, sillä Fidelix-järjestelmän Modbus-väylä käyttää näitä asetuksia tehdasasetuksena. Pariteetiksi valitaan 'none', sillä rekisteridatan sisällä on jo kaksi tavua CRC-tarkistussummaa. Alimmaiseen kohtaan valitaan mikä tahansa laiteosoite väliltä 1–255. Osoite ei saa olla sama kuin Fidelix-järjestelmän omat moduuliosoitteet tai minkään muun Modbus-laitteen, jos nämä ovat kytketty samaan porttiin.

Yhdyskäytävälaitteen väyläksi varattiin porttinumero 5 ja Fidelix-järjestelmän omat moduulit olivat porttinumerossa 3. Kohteessa oli myös Ouman 203/GT porttinumerossa 5, mutta siinä käytettiin Modbus-osoitetta 9, jolloin välttyttiin porttien väliseltä päällekkäisyyksiltä. Nyt voitiin tallentaa muutokset konfigurointiohjelmaan.



KUVA 4. LinkBoxMB tiedonsiirtoasetukset

4.3 KNX-väylän ryhmäosoitteet ja Modbus-väylän rekisterit

Seuraavana työvaiheena on konfiguroinnin työläin vaihe, jos hallittavia pisteitä on paljon. Kohteeseen oli tarkoitus tehdä kaksi releohjausta, joista toinen oli ulkovalojen ja toinen mainosvalojen ohjausryhmä. Tätä varten KNX:ään oli tehty ryhmäosoitteet 1/0/0 ja 1/1/0. Ryhmäosoitteet teki KNX-laitetoimittaja UTU Powel. Fidelix-järjestelmään tehtiin näitä ryhmäosoitteita vastaavat Modbus-väylän pisteet 100 ja 101. Valmiiseen kohteeseen tehtiin vielä varaukseksi 13 lisäryhmää, jotta myöhemmin voidaan tarvittaessa käyttöönottaa lisäpisteet helpommin etäyhteydellä.

Kuvassa 5 kahdeksan ensimmäistä saraketta on tarkoitettu KNX-väylän tietoihin. Näistä EIS-sarakkeesta voidaan valita tiedolle tyyppi. 1-Switching (1

bit) tarkoittaa tietoa joka on yhden bitin kokoinen, eli 0 tai 1. Tätä käytetään sekä esimerkiksi valojen ohjauksiin että erilaisten tilatietojen ilmaisemiseen. Muita vaihtoehtoja, joita tässä työssä ei käytetty, löytyy muun muassa 4 bitin himmennintieto sekä 16-bittinen liukuluku mittauksia ja asetuksia varten.

GROUP-sarakkeeseen kirjoitetaan se ryhmäosoite, joka on KNX-väylällä tälle ryhmälle ohjelmoitu. Sarake Format tarkoittaa tiedon muotoa, mikä lähetetään Modbus-väylään. Nyrkkisääntönä on '1-16 bits unsigned' kaikissa ohjauksissa, tilatiedoissa sekä himmentimissä. '3-16 bits signed * 10' käytetään mittausten ja asetusten kirjoittamisessa.

Point-sarakkeessa valitaan rekisteri, jossa tieto on. Nämä on syytä olla numerojärjestyksessä helpottamaan myöhemmän Modbus-laitekannan konfigurointia Fidelix-järjestelmässä. R/W-sarake kertoo sen, kirjoittaako vai lukeeko väylä vai tekeekö se molempia. Pelkkä R tarkoittaa, että vain IntesisBox-yhdyskäytävälaite välittää KNX-väylältä tulevan tiedon Modbus-väylän luettavaksi. W tarkoittaa, että IntesisBox kirjoittaa KNX-väylälle Modbus-väylältä tiedon. R/W:n tapauksessa Modbus-väylä voi kirjoittaa ja lukea KNX-väylää. Näin ollen Fidelix-järjestelmän väyläohjaus jää voimaan.

LinkBoxMB-konfiguroinnista voidaan valita, että KNX-väylälle kirjoitetaan vain, jos Modbus-väylällä tapahtuu tilan muutos. Viimeinen sarake ilmoittaa, onko kyseinen piste käytössä vai ei.

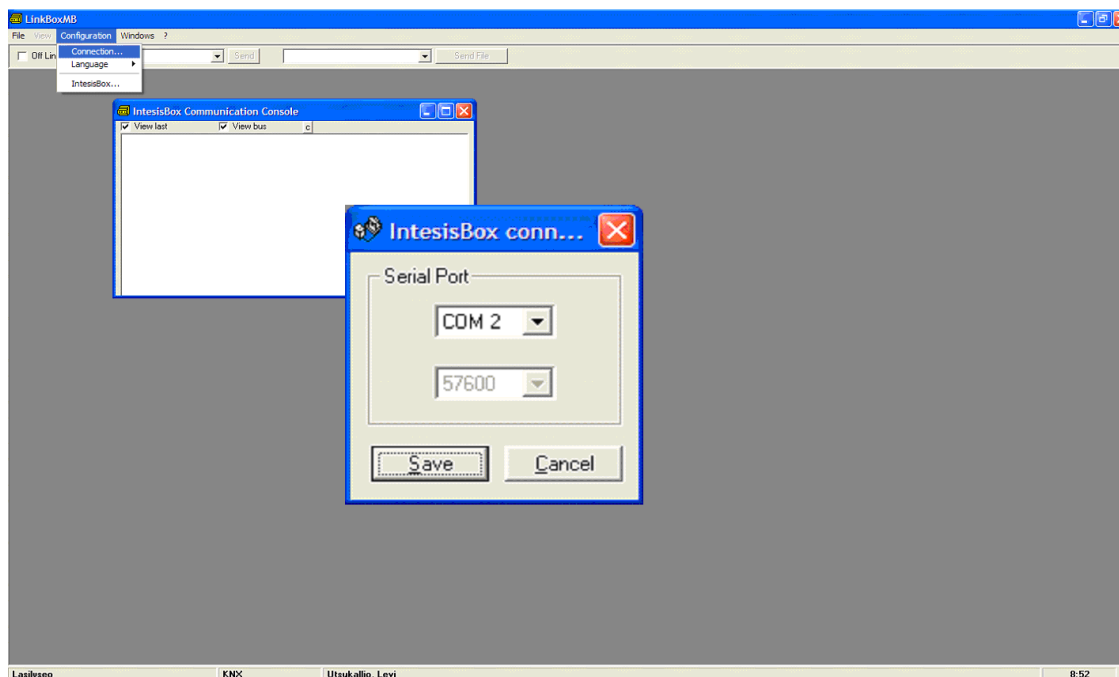
#	Data type	Description	EIS	Group	Listening addresses	R	W	T	U	Format	Point	R/W	Active
1	0-Error co	Error com. KNX								1-16 bits unsigned	1		1-Yes
2	1-Data	Ryhmä 12	1 - Switching (1 bit)	4/2/1		R	W	T		1-16 bits unsigned	113	1-RW	1-Yes
3	1-Data	Ryhmä 13	1 - Switching (1 bit)	4/1/0		R	W	T		1-16 bits unsigned	114	1-RW	1-Yes
4	1-Data	Ryhmä 11	1 - Switching (1 bit)	4/0/0		R	W	T		1-16 bits unsigned	112	1-RW	1-Yes
5	1-Data	Ryhmä 10	1 - Switching (1 bit)	3/1/1		R	W	T		1-16 bits unsigned	111	1-RW	1-Yes
6	1-Data	Ryhmä 9	1 - Switching (1 bit)	3/1/0		R	W	T		1-16 bits unsigned	110	1-RW	1-Yes
7	1-Data	Ryhmä 8	1 - Switching (1 bit)	3/0/0		R	W	T		1-16 bits unsigned	109	1-RW	1-Yes
8	1-Data	Ryhmä 7	1 - Switching (1 bit)	2/2/1		R	W	T		1-16 bits unsigned	108	1-RW	1-Yes
9	1-Data	Ryhmä 6	1 - Switching (1 bit)	2/1/2		R	W	T		1-16 bits unsigned	107	1-RW	1-Yes
10	1-Data	Ryhmä 5	1 - Switching (1 bit)	2/0/1		R	W	T		1-16 bits unsigned	106	1-RW	1-Yes
11	1-Data	Ryhmä 4	1 - Switching (1 bit)	2/2/2		R	W	T		1-16 bits unsigned	105	1-RW	1-Yes
12	1-Data	Ryhmä 3	1 - Switching (1 bit)	2/1/1		R	W	T		1-16 bits unsigned	104	1-RW	1-Yes
13	1-Data	Ryhmä 1	1 - Switching (1 bit)	2/1/0		R	W	T		1-16 bits unsigned	102	1-RW	1-Yes
14	1-Data	Ryhmä 2	1 - Switching (1 bit)	2/0/0		R	W	T		1-16 bits unsigned	103	1-RW	1-Yes
15	1-Data	Ulkovoalo ryhmä	1 - Switching (1 bit)	1/0/0		R	W	T		1-16 bits unsigned	100	1-RW	1-Yes
16	1-Data	Mainosvalot	1 - Switching (1 bit)	1/1/0		R	W	T		1-16 bits unsigned	101	1-RW	1-Yes

KUVA 5. Kohteen KNX-ryhmäosoitteet ja Modbus-väylän rekisterit

Näillä asetuksilla voidaan yksiselitteisesti osoittaa KNX-järjestelmän ryhmäosoitteellinen tieto, joka on luettavissa Modbus-väylältä. Tarkemmin sanottuna se luetaan käyttämällä ohjelmallista Modbus-laitetta, jonka rekisteri on määritelty LinkBoxMB-konfigurointiin samalle riville kuin ryhmäosoite. Tieto on haettavissa grafiikalle käyttäjän tulkittavaksi käyttämällä samoja Modbus-laitteita kuten edellä.

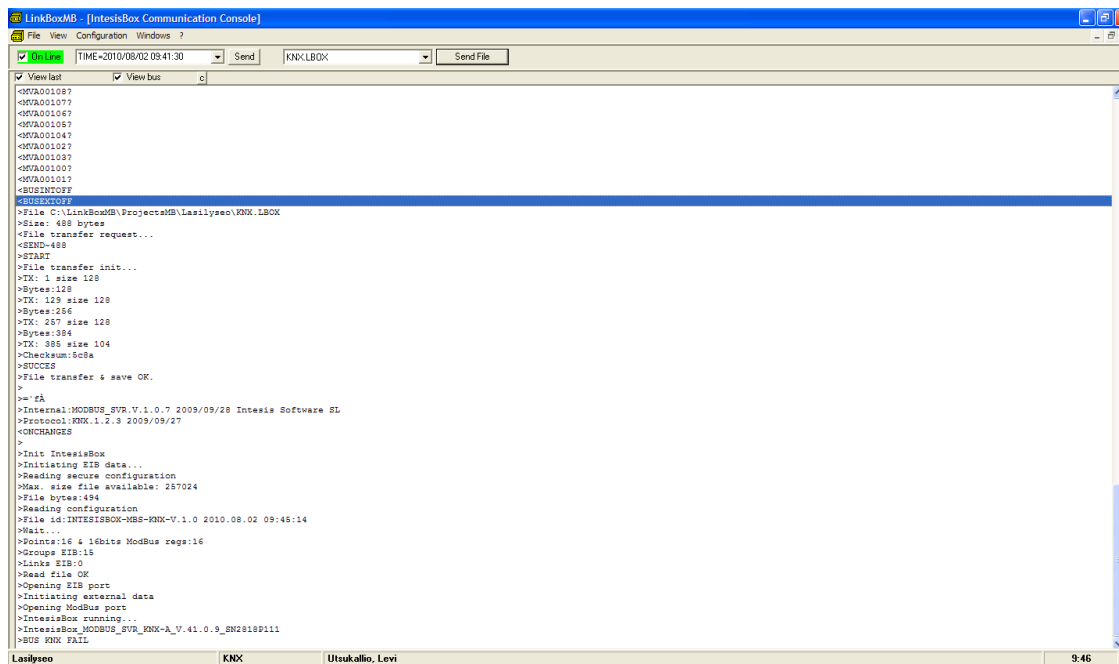
4.4 Tietojen lataus Intesis-yhdyskäytävälaitteelle

Kun projekti on konfiguroitu kohdetta vastaavaksi, on aika ladata tiedot IntesisBox-yhdyskäytävälaitteeseen. Tämä tapahtuu kytkemällä USB-RS485-kaapeli tietokoneen, jossa LinkBoxMB-ohjelma on, ja IntesisBox-yhdyskäytävälaitteen välille. Konfigurointiohjelman valikosta valitaan Configuration/Connection.. ja valitaan USB-kaapelille oikea portti (kuva 6). Windowsin ohjauspaneelin kautta pitää käydä varmistamassa, että portti ja tiedonsiirtonopeus vastaavat LinkBoxMB-valintoja.



KUVA 6. USB COM-portti yhteyden luontia varten

Asetukset ladataan laitteeseen painamalla 'Off Line' -painiketta, jolloin ohjelma alkaa ladata asetuksia laitteeseen. LinkBoxMB-ohjelman 'Communication Console' -ikkunasta voi seurata asetusten siirtymistä ja siirron valmistumista. (kuva 7).



KUVA 7. Tiedonsiirron seurantaikkuna

4.5 Yhteyksien testaaminen

Edellä mainittujen toimenpiteiden jälkeen yhdyskäytävälaite kommunikoi molempien väylien kanssa. Väylien liikennettä voidaan seurata valitsemalla View-valikosta joko Modbus- tai KNX-väylä. Nämä ikkunat tosin näyttävät liikenteen heksadesimaalilukuna, joten enemmän informaatiota antaa Signals Viewer-ikkuna. (kuva 8). Tämän ikkunan Value-sarakkeesta nähdään kyseisen pisteen tila. Samasta sarakkeesta voidaan pakottaa piste haluttuun tilaan. Yhdyskäytävälaite tulkitsee sen niin, että Modbus-väylältä tulee käsky vaihtaa tilaa. (5, s. 5–28.)

#	Signal	EIS	Group	Listening addresses	RW/TU	Format	Point R/W	Value
1	Error com. KNX					1-16 bits unsigned	1-OR	0
2	Ryhmä 12	1 - Switching (1 bit)	4/2/1		RW/T	1-16 bits unsigned	113-1-RW	0
3	Ryhmä 13	1 - Switching (1 bit)	4/1/0		RW/T	1-16 bits unsigned	114-1-RW	0
4	Ryhmä 11	1 - Switching (1 bit)	4/0/0		RW/T	1-16 bits unsigned	112-1-RW	0
5	Ryhmä 10	1 - Switching (1 bit)	3/1/1		RW/T	1-16 bits unsigned	111-1-RW	0
6	Ryhmä 9	1 - Switching (1 bit)	3/1/0		RW/T	1-16 bits unsigned	110-1-RW	0
7	Ryhmä 8	1 - Switching (1 bit)	3/0/0		RW/T	1-16 bits unsigned	109-1-RW	0
8	Ryhmä 7	1 - Switching (1 bit)	2/2/1		RW/T	1-16 bits unsigned	108-1-RW	0
9	Ryhmä 6	1 - Switching (1 bit)	2/1/2		RW/T	1-16 bits unsigned	107-1-RW	0
10	Ryhmä 5	1 - Switching (1 bit)	2/0/1		RW/T	1-16 bits unsigned	106-1-RW	0
11	Ryhmä 4	1 - Switching (1 bit)	2/2/2		RW/T	1-16 bits unsigned	105-1-RW	0
12	Ryhmä 3	1 - Switching (1 bit)	2/1/1		RW/T	1-16 bits unsigned	104-1-RW	0
13	Kotona	1 - Switching (1 bit)	1/2/0		RW/T	1-16 bits unsigned	102-1-RW	1
14	Poissa	1 - Switching (1 bit)	1/3/0		RW/T	1-16 bits unsigned	103-1-RW	1
15	Ulkovalo ryhmä	1 - Switching (1 bit)	1/0/0		RW/T	1-16 bits unsigned	100-1-RW	1
16	Mainsvalot	1 - Switching (1 bit)	1/1/0		RW/T	1-16 bits unsigned	101-1-RW	1

KUVA 8. Ohjauksien tilatiedot sekä KNX- ja Modbus-väylän rekisteriliikenne

5 FIDELIX-ALAKESKUKSEN OHJELMOINTI

Fidelix-alakeskuksen ohjelmointi tapahtuu Infoteamin OpenPCS ohjelmointityökalulla. Työkalu käyttää standardina IEC 61131-3:sen mukaista ohjelmointikieltä. Ohjelman syntaksi muistuttaa C-kieltä.

5.1 Lähtötilanne

Modbus–KNX-rajapintaa oli jo tutkittu hieman Vantaalla yhteistyössä KNX-laitetoimittaja ABB:n kanssa. Fidelix-järjestelmän käsiohjaus oli antanut IntesisBox-yhdyskäytävälaitteessa virheilmauksen, eikä ohjaus näin ollen mennyt KNX-väylälle asti. IntesisBox-yhdyskäytävälaitteen konfigurointiasetuksissa oli jotakin väärin. Ongelmaa kartoitettiin IntesisBox-yhdyskäytävälaitteen toimittaneen yrityksen henkilökunnan avulla. Virheilmauksen syyksi selvisi se, että rekisterit olivat menneet sekaisin. Ohjelmassa yritettiin kirjoittaa tietoa 'vain luku' -rekisteriin. Ongelma poistui järjestelemällä rekisterit LinkBoxMB-ohjelmassa niin, että peräkkäisinä rekistereinä oli kirjoitettavat rekisterit ja 'vain luku' -rekisterit.

5.2 Ohjelman soveltaminen

Ongelman ratkettua päästiin itse kohteen ohjelmien tekoon. Vantaalla tehtyä väyläohjelmaa muokattiin niin, että se soveltui kohteeseen. Samalla tehtiin ohjelmasta sellaista, että se soveltui jatkossa ohjelmajohdaksi muihin kohteisiin, jossa tarvitsi vain määritellä kohdekohtaiset pisteet Fidelix-järjestelmässä (liite 2) ja konfiguroida IntesisBox-yhdyskäytävälaite KNX-ryhmäpisteillä.

OpenPCS-ohjelmoinnissa käytetään Fidelix Oy:n tekemiä kirjastoja. Väylä-ohjelmoinnit suoritetaan käyttämällä toimilohko-ohjelmia, joihin syötetään tietoja. Toimilohko käsittelee tietoja ohjelmoidulla tavalla ja antaa ulostulona käsitellyn tiedon. Tässä tapauksessa IntesisBox-yhdyskäytävälaitteen Modbus-

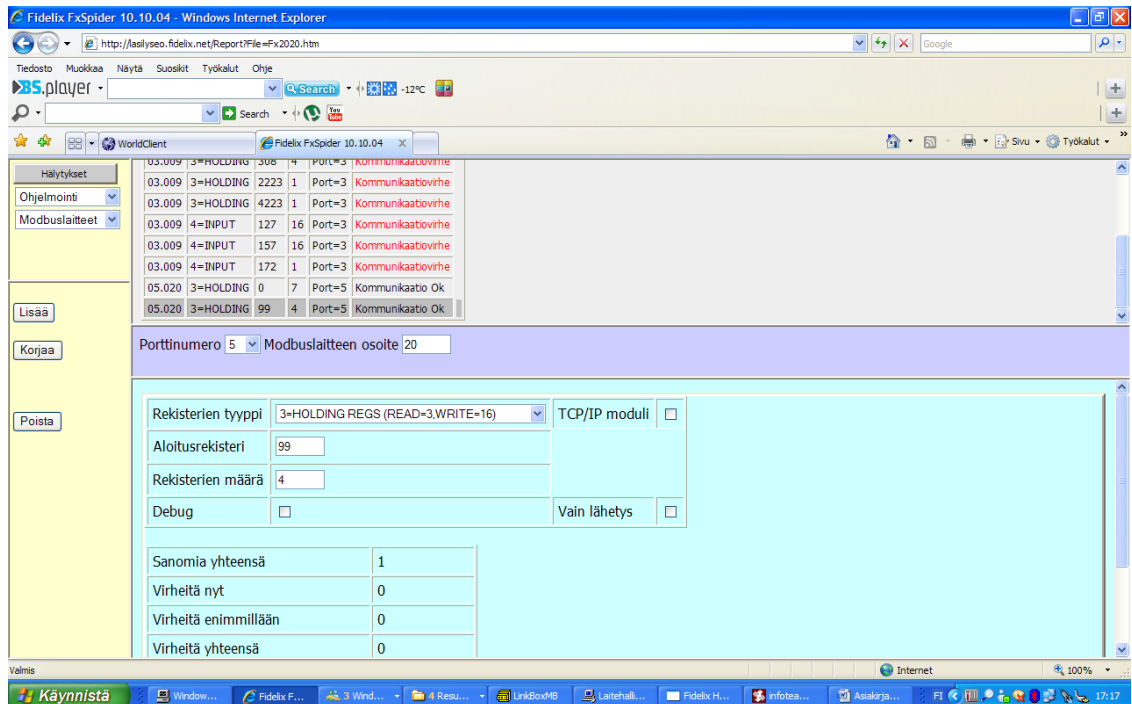
väylältä tuleva tieto syötetään 'FXIEC_FB_KNX_VO1.ST' -toimilohkolle, joka käsittelee KNX-rekisterin tiedon ja tallentaa tiedon ymmärrettävässä muodossa Fidelix-järjestelmän omalle rekisterille. Varsinaisessa ohjelmassa, 'KNX_MODBUS.ST', haetaan tiedot rekisteristä ja tallennetaan pisteelle (liite 3). Pisteiden tila löytyy myös alakeskuksen graafiselta näytöltä, josta käyttäjä voi lukea sen. Pistettä voidaan myös ohjata näytöltä.

5.3 Alakeskuksen Modbus-laitteet

Kun ohjelmat ja asetukset on saatu ladattua laitteisiin, voidaan siirtyä Fidelix-järjestelmän rekisterien hallintaan. Käytettäessä väylätietoja ohjelmasta joudutaan luomaan Modbus-laitteet kyseisille rekisteritiedoille. Kuvassa 10 on luotu KNX-väylää varten kaksi erillistä Modbus-laitetta:

- 05.020 3= HOLDING 0 7 Port=5 Kommunikaatio OK
- 05.020 3= HOLDING 99 4 Port=5 Kommunikaatio OK.

Ensimmäiset kaksi numerosarjaa merkitsevät porttia ja IntesisBox-yhdyskäytävälaitteen Modbus-osoitetta, seuraavaksi on ilmoitettu tietotyyppi, sen jälkeen aloitusrekisterin numero ja sitten montako peräkkäistä rekisteriä luetaan. Sen jälkeen on uudestaan portti selkokielellä sekä Modbus-väylän liikenteen tila.



KUVA 10. Fidelix-järjestelmän Modbuslaitteet

Rekisteristä 0 lähtien luettavat 7 rekisteriä ovat IntesisBox-yhdyskäytävälaitteen omia rekistereitä, joista voidaan lukea esimerkiksi IntesisBox-yhdyskäytävälaitteen päiväys ja kellonaika. Yhdestä näistä rekistereistä on tieto KNX-väylän liikenteen toimivuudesta, josta voidaan esimerkiksi ottaa hälytys Fidelix-järjestelmään jatkohälytystä varten. Työn kannalta tärkein rekisteriryhmä alkaa rekisteristä 99 ja jatkuu neljän rekisterin verran. Eli näissä on KNX-ryhmäohjaukset, mainosvalot, ulkovaloryhmä sekä kaksi jälkeempään lisättyä ryhmäohjausta, 'kotona' ja 'poissa'.

Väylän toimivuus nähdään suoraan viimeisestä sarakkeesta, jossa lukee kommunikaatio OK. Tämä ei vielä tarkoita, että ohjelma toimisi, vaan pitää myös tarkastella sanomien liikennettä. Kuvasta nähdään, että sanomia on yhteensä yksi, eikä siinä ole virhettä, joten ohjelma toimii.

6 ASENNUKSET JA TESTAUKSET

Testausten ja muiden taloautomaatio-ohjelmien teon jälkeen siirryttiin työkohteeseen. Fidelix-alakeskuksen kytkennät oli tehnyt aliurakoitsijayritys. Valvonta-alakeskukseen ladattiin toimistolla tehdyt ja testatut ohjelmat ja sen jälkeen kytkettiin KNX- ja Modbus-väylä tuotuun IntesisBox-yhdyskäytävälaitteen liittimiin. IntesisBox-yhdyskäytävälaite konfiguroitiin KNX-ohjelmoijalta saaduilla ryhmäosoitteilla, minkä jälkeen valmistauduttiin yhteistestiin.

Ensimmäisen testin jälkeen vain toista valoryhmistä voitiin ohjata Fidelix-järjestelmästä. Vika löytyi ongelmaa miettimällä ja testaamalla eri vaihtoehtoja. Vika oli siinä, että Fidelix-järjestelmän Modbus-laitteen aloitusrekisteri oli 100 ja LinkBoxMB-ohjelmassa oli rekisteri 100. Fidelix-järjestelmässä rekisteri 0 vastaa Modbus-väylässä rekisteriä 1, joten muuttamalla rekisterin alkamaan rekisteristä 99 saatiin molemmat ryhmäohjaukset toimimaan.

Fidelix-järjestelmän ohjattaessa valoryhmän tilaa toiseen KNX-väylä pakotti hetken päästä valoryhmän tilan takaisin alkuperäiseen, ja tällöin vaihtui tila myös Fidelix-järjestelmässä takaisin. Ongelma poistui, kun pakotettiin Modbus-väylältä tila jäämään sekunniksi haluttuun tilaan, tällöin KNX-väylä ehti lukea tiedon ennen kuin itse kirjoitti sen päälle.

Loppuaika käytettiin käyttöliittymän hiomiseen sellaiseksi, että käyttäjä osaisi hyödyntää sitä mahdollisimman hyvin. Lopputulokseksi syntyi grafiikka (kuva 11), jossa näytettiin KNX-väylän tila sekä vieressä käyttäjän haluama tila. Tiloja määriteltiin 3, jotka olivat:

- valot pois päältä
- valot päälle
- valot KNX-automaattiohjauksessa.

Tilassa 'valot pois päältä' lähetetään KNX-väylälle bitti nolla jatkuvana ja vastaavasti tilassa 'valot päälle' bitti yksi jatkuvasti. Näin ollen vaikka käyttäjä painaisi vastaavaa fyysistä painiketta kohteessa, ei mitään tapahtuisi. Kun taas tila on automaattiohjauksessa, toimii KNX-ohjaus kuten se on tarkoitettu.

	KNX tila	Modbus-ohjaus
Mainosvalot	1	KNX-Automaatilla
Ulkovalot	0	Valot pois
Kotona	0	Poissa
Poissa	1	Poissa

KUVA 11. Graafinen käyttöliittymä KNX-väylän ohjauksille

7 YHTEENVETO

Opinnäytetyön tarkoituksena oli löytää rajapinta Modbus- ja KNX-väylien välille Fidelix-järjestelmässä. Työ toteutettiin käyttämällä Intesis Software SL:n IntesisBox-yhdyskäytävälaitetta.

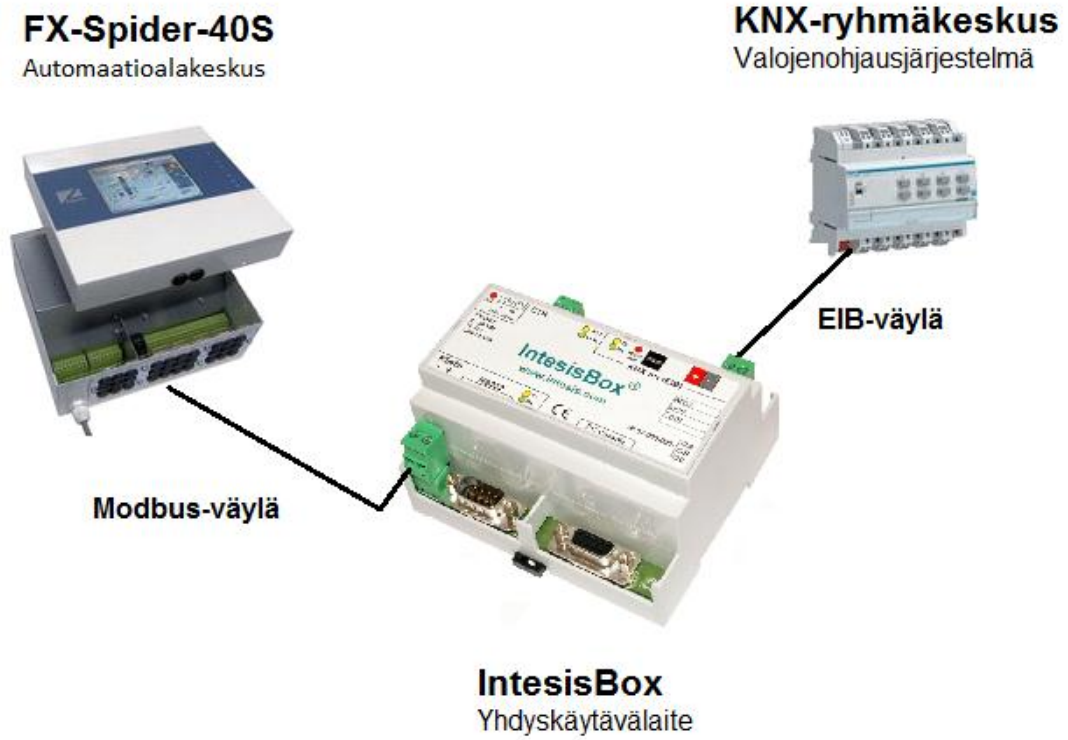
Työn suorituksen ensimmäinen vaihe oli IntesisBox-yhdyskäytävälaitteen konfigurointi vastaamaan Modbus-väylän rekistereitä ja KNX-väylän ryhmäosoitteita. Tässä kohdassa piti muistaa huomioida Fidelix-järjestelmän ja IntesisBox-yhdyskäytävälaitteen rekisterinumeromerkintöjen ero. Fidelix-järjestelmän Modbus-laitteeksi on merkittävä aina yksi pienempi rekisterinumero kuin vastaava IntesisBox-yhdyskäytävälaitteessa. Seuraavaksi ohjelmoitiin Fidelix-järjestelmä, jotta KNX-väylän tiedot saadaan käyttäjän ohjattavaksi. Samaa ohjelmaa käytettiin tilaajayrityksen pyytämän yleisohjelman pohjaksi. Lopuksi testattiin väylän toimivuus siten että, ohjattiin Fidelix-järjestelmän grafiikalta valoryhmän tilaa ja seurattiin KNX-keskuksen releiden toimintaa. Vastaavasti valoryhmän tilaa ohjattiin KNX-kytkimestä ja seurattiin tilan vaihtumista Fidelix-järjestelmän grafiikalta. Testit osoittivat Modbus- ja KNX-väylien toimivan moitteettomasti ja työ voitiin viimeistellä hiomalla Fidelix-järjestelmän grafiikka helppolukuisemmaksi.

Työssä onnistuttiin ohjaamaan Fidelix-järjestelmästä KNX-järjestelmän valoryhmiä väylän kautta perinteisen I/O:n sijaan. Tämä säästää automaatiojärjestelmän I/O:ta sekä kaapelointia KNX-keskuksen ja valvonta-alakeskuksen välillä. Ainakin osa mahdollisista muutostöistä voidaan tehdä vain ja ainoastaan konfiguroimalla uudelleen IntesisBox-yhdyskäytävälaite ja muuttamalla Fidelix-ohjelmaa vastaavasti.

Opinnäytetyön tavoitteet saavutettiin, sillä tämän työn ohella työstettiin Fidelix Oy:lle yleisohjelma, jolla voidaan helposti ja nopeasti ottaa Modbus–KNX-rajapinta käyttöön. Käyttöönottaján tarvitsee vain määritellä Fidelix-järjestelmään ohjatut tai luetut KNX-väylän pisteet sekä konfiguroida IntesisBox-yhdyskäytävälaite vastaamaan väylien rekistereitä.

LÄHTEET

1. Programming manual version 10.52. Sisäinen dokumentti. Fidelix Oy.
2. Modicon Modbus Protocol Reference Guide. 1996. Modicon Inc. Saatavissa: http://modbus.org/docs/PI_MBUS_300.pdf. Hakupäivä 29.4.2012.
3. Schneider, Godehart W – Tschischka, Walter – Heinje, Torsten 2010. Käsikirja asuntojen ja rakennusten ohjauksiin - KNX Peruseriaatteet. Suom. Piikkilä, Veijo – Liukku, Harri – Parviainen, Kari. KNX Finland Ry. Espoo.
4. Implementation of the KNX Standard. 2006. Weinzierl Engineering GmbH. Saatavissa: http://www.weinzierl.de/download/references/KNX_Impl_E_2006_11_02.pdf. Hakupäivä 12.5.2012.
5. IntesisBox Modbus Server - KNX datasheet v10 r12 eng. 2009. Datalehti. Intesis Software SL.



KNX toimilohkon muuttujien esittely

(* ----- *)

VAR_EXTERNAL
END_VAR

VAR_INPUT

modbus_address : int;
Port : int;

```
id_holding_reg_100 : string (30) := "";
id_holding_reg_101 : string (30) := "";
id_holding_reg_102 : string (30) := "";
id_holding_reg_103 : string (30) := "";
id_holding_reg_104 : string (30) := "";
id_holding_reg_105 : string (30) := "";
id_holding_reg_106 : string (30) := "";
id_holding_reg_107 : string (30) := "";
id_holding_reg_108 : string (30) := "";
id_holding_reg_109 : string (30) := "";
id_holding_reg_110 : string (30) := "";
id_holding_reg_111 : string (30) := "";
id_holding_reg_112 : string (30) := "";
id_holding_reg_113 : string (30) := "";
id_holding_reg_114 : string (30) := "";
id_holding_reg_115 : string (30) := "";
id_holding_reg_116 : string (30) := "";
id_holding_reg_117 : string (30) := "";
(* jne *)
```

```
id_holding_reg_200 : string (30) := "";
id_holding_reg_201 : string (30) := "";
id_holding_reg_202 : string (30) := "";
id_holding_reg_203 : string (30) := "";
id_holding_reg_204 : string (30) := "";
id_holding_reg_205 : string (30) := "";
id_holding_reg_206 : string (30) := "";
id_holding_reg_207 : string (30) := "";
id_holding_reg_208 : string (30) := "";
id_holding_reg_209 : string (30) := "";
id_holding_reg_210 : string (30) := "";
id_holding_reg_211 : string (30) := "";
id_holding_reg_212 : string (30) := "";
id_holding_reg_213 : string (30) := "";
id_holding_reg_214 : string (30) := "";
id_holding_reg_215 : string (30) := "";
id_holding_reg_216 : string (30) := "";
```

```
id_holding_reg_217 : string (30) := "";
(* jne *)
```

```
id_coil_300 : string (30) := "";
id_coil_301 : string (30) := "";
id_coil_302 : string (30) := "";
id_coil_303 : string (30) := "";
id_coil_304 : string (30) := "";
id_coil_305 : string (30) := "";
id_coil_306 : string (30) := "";
id_coil_307 : string (30) := "";
id_coil_308 : string (30) := "";
id_coil_309 : string (30) := "";
id_coil_310 : string (30) := "";
id_coil_311 : string (30) := "";
id_coil_312 : string (30) := "";
id_coil_313 : string (30) := "";
id_coil_314 : string (30) := "";
id_coil_315 : string (30) := "";
id_coil_316 : string (30) := "";
id_coil_317 : string (30) := "";
id_coil_318 : string (30) := "";
id_coil_319 : string (30) := "";
id_coil_320 : string (30) := "";
id_coil_321 : string (30) := "";
id_coil_322 : string (30) := "";
(* jne *)
```

```
id_coil_400 : string (30) := "";
id_coil_401 : string (30) := "";
id_coil_402 : string (30) := "";
id_coil_403 : string (30) := "";
id_coil_404 : string (30) := "";
id_coil_405 : string (30) := "";
id_coil_406 : string (30) := "";
id_coil_407 : string (30) := "";
id_coil_408 : string (30) := "";
id_coil_409 : string (30) := "";
id_coil_410 : string (30) := "";
id_coil_411 : string (30) := "";
id_coil_412 : string (30) := "";
id_coil_413 : string (30) := "";
id_coil_414 : string (30) := "";
id_coil_415 : string (30) := "";
id_coil_416 : string (30) := "";
id_coil_417 : string (30) := "";
id_coil_418 : string (30) := "";
id_coil_419 : string (30) := "";
```

```

        id_coil_420 : string (30) := "";
        id_coil_421 : string (30) := "";
        id_coil_422 : string (30) := "";
        (* jne *)

        id_modbus_alarm : string (30) := "";
END_VAR

VAR_OUTPUT
END_VAR

VAR
        udiResult : udint;
        KNX : GenericModbus642FB;
        Result : int;
        ohjaus : int;
        speed : dint;
        swap_FB : BYTE_SWAP;
        word_temp : word;
        inverted : word;
        original, byte1, byte2 : word;
END_VAR

VAR CONSTANT
END_VAR

(* ----- *)

KNX toimilohko-ohjelma
(* ----- *)

KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=0,
RegisterType:=3);

if KNX.datavalid = 1 then
        udiResult := DoubleHoldingRegF(id_String:=id_date5,
        HighRegValue:=KNX.Reg4, LowRegValue:=KNX.Reg5,
        rMultiplier:=1.0);
        UdiResult := DoubleHoldingRegF(id_String:=id_time7,
        HighRegValue:=KNX.Reg6, LowRegValue:=KNX.Reg7,
        rMultiplier:=1.0);
end_if;

KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=99,
RegisterType:=3);

if KNX.datavalid = 1 then

```

```
KNX.Reg1 := HoldingRegF(id_String := id_holding_reg_100,
RegValue := KNX.Reg1, rMultiplier := 10.0);
KNX.Reg2 := HoldingRegF(id_String := id_holding_reg_101,
RegValue := KNX.Reg2, rMultiplier := 10.0);
KNX.Reg3 := HoldingRegF(id_String := id_holding_reg_102,
RegValue := KNX.Reg3, rMultiplier := 10.0);
KNX.Reg4 := HoldingRegF(id_String := id_holding_reg_103,
RegValue := KNX.Reg4, rMultiplier := 10.0);
KNX.Reg5 := HoldingRegF(id_String := id_holding_reg_104,
RegValue := KNX.Reg5, rMultiplier := 10.0);
KNX.Reg6 := HoldingRegF(id_String := id_holding_reg_105,
RegValue := KNX.Reg6, rMultiplier := 10.0);
KNX.Reg7 := HoldingRegF(id_String := id_holding_reg_106,
RegValue := KNX.Reg7, rMultiplier := 10.0);
KNX.Reg8 := HoldingRegF(id_String := id_holding_reg_107,
RegValue := KNX.Reg8, rMultiplier := 10.0);
KNX.Reg9 := HoldingRegF(id_String := id_holding_reg_108,
RegValue := KNX.Reg9, rMultiplier := 10.0);
KNX.Reg10 := HoldingRegF(id_String := id_holding_reg_109,
RegValue := KNX.Reg10, rMultiplier := 10.0);
KNX.Reg11 := HoldingRegF(id_String := id_holding_reg_110,
RegValue := KNX.Reg11, rMultiplier := 10.0);
KNX.Reg12 := HoldingRegF(id_String := id_holding_reg_111,
RegValue := KNX.Reg12, rMultiplier := 10.0);
KNX.Reg13 := HoldingRegF(id_String := id_holding_reg_112,
RegValue := KNX.Reg13, rMultiplier := 10.0);
KNX.Reg14 := HoldingRegF(id_String := id_holding_reg_113,
RegValue := KNX.Reg14, rMultiplier := 10.0);
KNX.Reg15 := HoldingRegF(id_String := id_holding_reg_114,
RegValue := KNX.Reg15, rMultiplier := 10.0);
KNX.Reg16 := HoldingRegF(id_String := id_holding_reg_115,
RegValue := KNX.Reg16, rMultiplier := 10.0);
KNX.Reg17 := HoldingRegF(id_String := id_holding_reg_116,
RegValue := KNX.Reg17, rMultiplier := 10.0);
KNX.Reg18 := HoldingRegF(id_String := id_holding_reg_117,
RegValue := KNX.Reg18, rMultiplier := 10.0);
KNX (Send:=2);

end_if;

KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=199,
RegisterType:=3);

if KNX.datavalid = 1 then
    KNX.Reg1 := HoldingRegF(id_String := id_holding_reg_200,
    RegValue := KNX.Reg1, rMultiplier := 10.0);
    KNX.Reg2 := HoldingRegF(id_String := id_holding_reg_201,
    RegValue := KNX.Reg2, rMultiplier := 10.0);
    KNX.Reg3 := HoldingRegF(id_String := id_holding_reg_202,
```

```
    RegValue := KNX.Reg3, rMultiplier := 10.0);
    KNX.Reg4 := HoldingRegF(id_String := id_holding_reg_203,
    RegValue := KNX.Reg4, rMultiplier := 10.0);
    KNX.Reg5 := HoldingRegF(id_String := id_holding_reg_204,
    RegValue := KNX.Reg5, rMultiplier := 10.0);
    KNX.Reg6 := HoldingRegF(id_String := id_holding_reg_205,
    RegValue := KNX.Reg6, rMultiplier := 10.0);
    KNX.Reg7 := HoldingRegF(id_String := id_holding_reg_206,
    RegValue := KNX.Reg7, rMultiplier := 10.0);
    KNX.Reg8 := HoldingRegF(id_String := id_holding_reg_207,
    RegValue := KNX.Reg8, rMultiplier := 10.0);
    KNX.Reg9 := HoldingRegF(id_String := id_holding_reg_208,
    RegValue := KNX.Reg9, rMultiplier := 10.0);
    KNX.Reg10 := HoldingRegF(id_String := id_holding_reg_209,
    RegValue := KNX.Reg10, rMultiplier := 10.0);
    KNX.Reg11 := HoldingRegF(id_String := id_holding_reg_210,
    RegValue := KNX.Reg11, rMultiplier := 10.0);
    KNX.Reg12 := HoldingRegF(id_String := id_holding_reg_211,
    RegValue := KNX.Reg12, rMultiplier := 10.0);
    KNX.Reg13 := HoldingRegF(id_String := id_holding_reg_212,
    RegValue := KNX.Reg13, rMultiplier := 10.0);
    KNX.Reg14 := HoldingRegF(id_String := id_holding_reg_213,
    RegValue := KNX.Reg14, rMultiplier := 10.0);
    KNX.Reg15 := HoldingRegF(id_String := id_holding_reg_214,
    RegValue := KNX.Reg15, rMultiplier := 10.0);
    KNX.Reg16 := HoldingRegF(id_String := id_holding_reg_215,
    RegValue := KNX.Reg16, rMultiplier := 10.0);
    KNX.Reg17 := HoldingRegF(id_String := id_holding_reg_216,
    RegValue := KNX.Reg17, rMultiplier := 10.0);
    KNX.Reg18 := HoldingRegF(id_String := id_holding_reg_217,
    RegValue := KNX.Reg18, rMultiplier := 10.0);
    KNX (Send:=2);
end_if;
KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=299,
RegisterType:=3);

if KNX.datavalid = 1 then
    KNX.Reg0 := HoldingRegF(id_String := id_coil_300, RegValue :=
    KNX.Reg0, rMultiplier := 1.0);
    KNX.Reg1 := HoldingRegF(id_String := id_coil_301, RegValue :=
    KNX.Reg1, rMultiplier := 1.0);
    KNX.Reg2 := HoldingRegF(id_String := id_coil_302, RegValue :=
    KNX.Reg2, rMultiplier := 1.0);
    KNX.Reg3 := HoldingRegF(id_String := id_coil_303, RegValue :=
    KNX.Reg3, rMultiplier := 1.0);
    KNX.Reg4 := HoldingRegF(id_String := id_coil_304, RegValue :=
    KNX.Reg4, rMultiplier := 1.0);
    KNX.Reg5 := HoldingRegF(id_String := id_coil_305, RegValue :=
```



```
KNX.Reg5, rMultiplier := 1.0);
KNX.Reg6 := HoldingRegF(id_String := id_coil_306, RegValue :=
KNX.Reg6, rMultiplier := 1.0);
KNX.Reg7 := HoldingRegF(id_String := id_coil_307, RegValue :=
KNX.Reg7, rMultiplier := 1.0);
KNX.Reg8 := HoldingRegF(id_String := id_coil_308, RegValue :=
KNX.Reg8, rMultiplier := 1.0);
KNX.Reg9 := HoldingRegF(id_String := id_coil_309, RegValue :=
KNX.Reg9, rMultiplier := 1.0);
KNX.Reg10 := HoldingRegF(id_String := id_coil_310, RegValue :=
KNX.Reg10, rMultiplier := 1.0);
KNX.Reg11 := HoldingRegF(id_String := id_coil_311, RegValue :=
KNX.Reg11, rMultiplier := 1.0);
KNX.Reg12 := HoldingRegF(id_String := id_coil_312, RegValue :=
KNX.Reg12, rMultiplier := 1.0);
KNX.Reg13 := HoldingRegF(id_String := id_coil_313, RegValue :=
KNX.Reg13, rMultiplier := 1.0);
KNX.Reg14 := HoldingRegF(id_String := id_coil_314, RegValue :=
KNX.Reg14, rMultiplier := 1.0);
KNX.Reg15 := HoldingRegF(id_String := id_coil_315, RegValue :=
KNX.Reg15, rMultiplier := 1.0);
KNX.Reg16 := HoldingRegF(id_String := id_coil_316, RegValue :=
KNX.Reg16, rMultiplier := 1.0);
KNX.Reg17 := HoldingRegF(id_String := id_coil_317, RegValue :=
KNX.Reg17, rMultiplier := 1.0);
KNX.Reg18 := HoldingRegF(id_String := id_coil_318, RegValue :=
KNX.Reg18, rMultiplier := 1.0);
KNX.Reg19 := HoldingRegF(id_String := id_coil_319, RegValue :=
KNX.Reg19, rMultiplier := 1.0);
KNX.Reg20 := HoldingRegF(id_String := id_coil_320, RegValue :=
KNX.Reg20, rMultiplier := 1.0);
KNX.Reg21 := HoldingRegF(id_String := id_coil_321, RegValue :=
KNX.Reg21, rMultiplier := 1.0);
KNX.Reg22 := HoldingRegF(id_String := id_coil_322, RegValue :=
KNX.Reg22, rMultiplier := 1.0);
KNX (Send:=2);
end_if;

KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=399,
RegisterType:=3);

if KNX.datavalid = 1 then
KNX.Reg0 := HoldingRegF(id_String := id_coil_400, RegValue :=
KNX.Reg0, rMultiplier := 1.0);
KNX.Reg1 := HoldingRegF(id_String := id_coil_401, RegValue :=
KNX.Reg1, rMultiplier := 1.0);
KNX.Reg2 := HoldingRegF(id_String := id_coil_402, RegValue :=
KNX.Reg2, rMultiplier := 1.0);
```

```
KNX.Reg3 := HoldingRegF(id_String := id_coil_403, RegValue :=
KNX.Reg3, rMultiplier := 1.0);
KNX.Reg4 := HoldingRegF(id_String := id_coil_404, RegValue :=
KNX.Reg4, rMultiplier := 1.0);
KNX.Reg5 := HoldingRegF(id_String := id_coil_405, RegValue :=
KNX.Reg5, rMultiplier := 1.0);
KNX.Reg6 := HoldingRegF(id_String := id_coil_406, RegValue :=
KNX.Reg6, rMultiplier := 1.0);
KNX.Reg7 := HoldingRegF(id_String := id_coil_407, RegValue :=
KNX.Reg7, rMultiplier := 1.0);
KNX.Reg8 := HoldingRegF(id_String := id_coil_408, RegValue :=
KNX.Reg8, rMultiplier := 1.0);
KNX.Reg9 := HoldingRegF(id_String := id_coil_409, RegValue :=
KNX.Reg9, rMultiplier := 1.0);
KNX.Reg10 := HoldingRegF(id_String := id_coil_410, RegValue :=
KNX.Reg10, rMultiplier := 1.0);
KNX.Reg11 := HoldingRegF(id_String := id_coil_411, RegValue :=
KNX.Reg11, rMultiplier := 1.0);
KNX.Reg12 := HoldingRegF(id_String := id_coil_412, RegValue :=
KNX.Reg12, rMultiplier := 1.0);
KNX.Reg13 := HoldingRegF(id_String := id_coil_413, RegValue :=
KNX.Reg13, rMultiplier := 1.0);
KNX.Reg14 := HoldingRegF(id_String := id_coil_414, RegValue :=
KNX.Reg14, rMultiplier := 1.0);
KNX.Reg15 := HoldingRegF(id_String := id_coil_415, RegValue :=
KNX.Reg15, rMultiplier := 1.0);
KNX.Reg16 := HoldingRegF(id_String := id_coil_416, RegValue :=
KNX.Reg16, rMultiplier := 1.0);
KNX.Reg17 := HoldingRegF(id_String := id_coil_417, RegValue :=
KNX.Reg17, rMultiplier := 1.0);
KNX.Reg18 := HoldingRegF(id_String := id_coil_418, RegValue :=
KNX.Reg18, rMultiplier := 1.0);
KNX.Reg19 := HoldingRegF(id_String := id_coil_419, RegValue :=
KNX.Reg19, rMultiplier := 1.0);
KNX.Reg20 := HoldingRegF(id_String := id_coil_420, RegValue :=
KNX.Reg20, rMultiplier := 1.0);
KNX.Reg21 := HoldingRegF(id_String := id_coil_421, RegValue :=
KNX.Reg21, rMultiplier := 1.0);
KNX.Reg22 := HoldingRegF(id_String := id_coil_422, RegValue :=
KNX.Reg22, rMultiplier := 1.0);
KNX (Send:=2);
end_if;
```

(* Tarkistetaan kommunikaation toimivuus ja asetetaan hälytyspisteen tila *)

```
if KNX.datavalid = 0 then
```

```
    Result := SetDigitalPointF(
    Value:=BOOL_TO_INT(GetSystemStatusF( Mode:=1,
```

```
        iParameter:=modbus_address, rParameter:=0.0 ) > 50.0),
        LockState:=1, Name:=id_modbus_alarm );
else      Result := SetDigitalPointF(
        Value:=BOOL_TO_INT(GetSystemStatusF( Mode:=0,
        iParameter:=modbus_address, rParameter:=0.0 ) < 50.0),
        LockState:=1, Name:=id_modbus_alarm );
end_if;
```

```
(* ----- *)
```

KNX toimilohkon kutsuohjelman esittely

```
(* ----- *)
```

VAR

```
        KNX : FXIEC_FB_KNX_V01_1;
        Result : int;
```

END_VAR

```
(* ----- *)
```

KNX toimilohkon kutsuohjelma

```
(* ----- *)
```

KNX (modbus_address := 20, (* Modbus-osoite *))

port := 5, (* Sarjaportin numero *)

(* modbuslaite rekisterit 99-148, käytä esim mittauksia *)

```
        id_holding_reg_100 := 'MITTAUS1_M',
        id_holding_reg_101 := "",
        id_holding_reg_102 := "",
        id_holding_reg_103 := "",
        id_holding_reg_104 := "",
        id_holding_reg_105 := "",
        id_holding_reg_106 := "",
        id_holding_reg_107 := "",
        Id_holding_reg_108 := "",
        id_holding_reg_109 := "",
        id_holding_reg_110 := "",
        id_holding_reg_111 := "",
        id_holding_reg_112 := "",
        id_holding_reg_113 := "",
        id_holding_reg_114 := "",
        id_holding_reg_115 := "",
```

```
id_holding_reg_116 := "  
id_holding_reg_117 := "  
(* JNE *)
```

(* modbuslaite rekisterit 199-248 käytä esim asetuksia *)

```
id_holding_reg_200 := 'ASETUS1_S',  
id_holding_reg_201 := "  
id_holding_reg_202 := "  
id_holding_reg_203 := "  
id_holding_reg_204 := "  
id_holding_reg_205 := "  
id_holding_reg_206 := "  
id_holding_reg_207 := "  
Id_holding_reg_208 := "  
id_holding_reg_209 := "  
id_holding_reg_210 := "  
id_holding_reg_211 := "  
id_holding_reg_212 := "  
id_holding_reg_213 := "  
id_holding_reg_214 := "  
id_holding_reg_215 := "  
id_holding_reg_216 := "  
id_holding_reg_217 := "  
(* JNE *)
```

(* modbuslaite rekisterit 299-348 käytä esim indikoiteja *)

```
id_coil_300 := 'TILATIETO1_I',  
id_coil_301 := "  
id_coil_302 := "  
id_coil_303 := "  
id_coil_304 := "  
id_coil_305 := "  
id_coil_306 := "  
id_coil_307 := "  
id_coil_308 := "  
id_coil_309 := "  
id_coil_310 := "  
id_coil_311 := "  
id_coil_312 := "  
id_coil_313 := "  
id_coil_314 := "  
id_coil_315 := "  
id_coil_316 := "  
id_coil_317 := "  
id_coil_318 := "  
id_coil_319 := "
```

```
id_coil_320 := "",
id_coil_321 := "",
id_coil_322 := "",
(* JNE ... *)
```

(* modbuslaite rekisterit 399-448 käytä rele ohjauksia *)

```
id_coil_400 := 'OHJAUS1_O',
id_coil_401 := "",
id_coil_402 := "",
id_coil_403 := "",
id_coil_404 := "",
id_coil_405 := "",
id_coil_406 := "",
id_coil_407 := "",
id_coil_408 := "",
id_coil_409 := "",
id_coil_410 := "",
id_coil_411 := "",
id_coil_412 := "",
id_coil_413 := "",
id_coil_414 := "",
id_coil_415 := "",
id_coil_416 := "",
id_coil_417 := "",
id_coil_418 := "",
id_coil_419 := "",
id_coil_420 := "",
id_coil_421 := "",
id_coil_422 := "",
(* JNE ... *)
```

(*Kommunikaatiohälytys *)

```
id_modbus_alarm := 'KNX_MODDBUS_ALARM_H');
```

(* ----- *)

Toimilohkon muuttujien esittely

(* ----- *)

VAR_EXTERNAL
END_VAR

VAR_INPUT

modbus_address : int;
Port : int;
id_coil_1 : string (30) := "
id_coil_2 : string (30) := "
id_coil_3 : string (30) := "
id_coil_4 : string (30) := "
id_modbus_alarm : string (30) := "

END_VAR

VAR_OUTPUT
END_VAR

VAR

udiResult : udint;
KNX : GenericModbus642FB;
Result : int;
ohjaus : int;
speed : dint;
swap_FB : BYTE_SWAP;
word_temp : word;
inverted : word;
original, byte1, byte2 : word;

END_VAR

VAR CONSTANT
END_VAR

(* ----- *)

Toimilohko-ohjelma

(* ----- *)

KNX (Send:=0, Port:=PORT, Module:=modbus_address, StartRegister:=99,
RegisterType:=3);

if KNX.datavalid = 1 then

KNX.Reg0 := HoldingRegF(id_String := id_coil_1, RegValue :=

```

        KNX.Reg0, rMultiplier := 1.0);
        KNX.Reg1 := HoldingRegF(id_String := id_coil_2, RegValue :=
        KNX.Reg1, rMultiplier := 1.0);
        KNX.Reg2 := HoldingRegF(id_String := id_coil_3, RegValue :=
        KNX.Reg2, rMultiplier := 1.0);
        KNX.Reg3 := HoldingRegF(id_String := id_coil_4, RegValue :=
        KNX.Reg3, rMultiplier := 1.0);
        KNX (Send:=2);
end_if;

```

```

(* Tarkistetaan kommunikaation toimivuus ja asetetaan hälytyspisteen tila *)
Result := SetDigitalPointF( Value:=BOOL_TO_INT(GetSystemStatusF(
Mode:=1, iParameter:=modbus_address, rParameter:=0.0 ) > 30.0),
LockState:=1, Name:=id_modbus_alarm );

```

```

(* ----- *)

```

Toimilohkon kutsuohjelman muuttujien esittely

```

(* ----- *)

```

```

VAR_EXTERNAL
END_VAR

```

```

VAR_GLOBAL
END_VAR

```

```

VAR
        KNX : FXIEC_FB_KNX_V01;
        Result : int;
END_VAR

```

```

(* ----- *)

```

Toimilohkon kutsuohjelma

```

(* ----- *)

```

```

KNX
        (modbus_address := 20, (* Modbus-osoite *)
        port := 5, (* Sarjaportin numero *)
        id_coil_1 := 'VALO1_O',
        id_coil_2 := 'VALO2_O',
        id_coil_3 := 'KOTONA_O',
        id_coil_4 := 'POISSA_O',

        (*Kommunikaatiohälytys *)

```

```
id_modbus_alarm := 'VLT_MODDBUS_ALARM_H');
```

```
(* ----- *)
```

IEC-ohjelman muuttujien esittely

```
(* ----- *)
```

```
VAR_EXTERNAL  
END_VAR
```

```
VAR_GLOBAL  
END_VAR
```

```
VAR  
    aika,tila,ohj,Tulos,vanha:int;  
    apu1, apu2:int;  
    FB_Viive_PA1:TON;  
    FB_Viive_PA2:TON;  
    FB_Viive_PA3:TON;  
    FB_Viive_PA4:TON;  
END_VAR
```

```
(* ----- *)
```

IEC-ohjelma (väylän tiedot grafiikkaan / grafiikasta ohjaus väylään)

```
(* ----- *)
```

```
aika := GetDigitalPointF (Name :='VALO1_T'); (* 0 = kiinni, 1 = auki, 2 =  
automaatilla *)
```

```
tila := GetDigitalPointF (Name :='VALO1_O');
```

```
if aika = 1 AND aika <> vanha then
```

```
    Tulos := SetDigitalPointF (Value := 1, LockState := 2, Name :=  
    'VALO1_O');  
    apu1 := 1;
```

```
end_if;
```

```
FB_Viive_PA1.IN:=(apu1 = 1);
```

```
FB_Viive_PA1.PT:=DINT_TO_TIME(1000*1);
```

```
FB_Viive_PA1();
```

```
if FB_Viive_PA1.Q = true then
```

```
    Tulos := SetDigitalPointF (Value := 1, LockState := 3,  
    Name := 'VALO1_O');
```



```
        apu1 := 0;
end_if;

if aika = 0 AND aika <> vanha then
    Tulos := SetDigitalPointF (Value := 0, LockState := 2, Name :=
        'VALO1_O');
    apu2 := 1;
end_if;

FB_Viive_PA2.IN:=(apu2 = 1);
FB_Viive_PA2.PT:=DINT_TO_TIME(1000*1);
FB_Viive_PA2();

if FB_Viive_PA2.Q = true then
    Tulos := SetDigitalPointF (Value := 0, LockState := 3,
        Name :='VALO1_O');
    apu2 := 0;
end_if;

vanha := aika;
```