

Tero Salminen

AUTOMAATIOLABORATORION ÄLYKÄS
VALAISTUKSENOHJAUS

Sähkö- ja automaatiotekniikan koulutusohjelma
2019

AUTOMAATIOLABORATORION ÄLYKÄS VALAISTUKSEN OHJAUS

Salminen, Tero
Satakunnan ammattikorkeakoulu
Sähkö- ja automaatiotekniikan koulutusohjelma
Sivumäärä: 47
Liitteitä: 3

Asiasanat: DALI, Beckhoff, TwinCat, EnOcean, Älykäs valaistus

Tässä opinnäytetyössä perehdytään Satakunnan ammattikorkeakoulun (SAMK) Automaatiolaboratorion älykkään valaistuksen ohjauksen käyttöönottoon ja valaistuksen ohjelmoimiseen Beckhoffin logiikalle. Työ toteutettiin kesän 2019 aikana.

Työssä perehdytään projektissa käytettyihin komponentteihin, DALI-valaistukseen ja Beckhoffin DALI-kirjastoon. Työkaluna on Beckhoffin TwinCat 3, jolla ohjelmointi on toteutettu.

Projektissa otettiin käyttöön laboratorioon jo aikaisemmin asennetut DALI-valaisimet ja niitä ohjataan Beckhoffin logiikan avulla. Laboratorion valaisimet jaettiin eri ryhmiin ja ryhmien valaisimia pääsee ohjaamaan langattomien kytkinten kautta tarvittaessa. Valaisimia ohjataan myös ajastimella. Konenäköalueelle tehtiin myös täysin itsenäinen oma alueensa, jota ohjataan täysin erikseen.

INTELLIGENT LIGHT CONTROL FOR THE AUTOMATION LABORATORY

Salminen, Tero

Satakunta University of Applied Sciences

Degree Programme in electrical and automation engineering

December 2019

Number of pages: 47

Appendices: 3

Keywords: DALI, Beckhoff, TwinCat, EnOcean, Intelligent lighting

In this thesis the focus is on commissioning an intelligent light control in the automation laboratory of Satakunta University of Applied Sciences (SAMK) and programming the lights with Beckhoff's PLC. This work was executed on summer 2019.

This thesis describes the components, DALI-lights and Beckhoff's DALI-library used in this project. The programming tool was TwinCat 3.

In this project the DALI-lights that were installed earlier were harnessed. The lights are now controlled by Beckhoff's PLC.

SISÄLLYS

1	JOHDANTO.....	6
2	KIINTEISTÖAUTOMAATIO VALAISTUKSEN OHJAUKSESSA	8
3	DALI	10
3.1	Yleistä tietoa DALI:sta	10
3.2	DALI- / DALI-2 -kaapelointi ja -topologia	10
3.3	DALI- / DALI-2 -laitteiston rakenne	11
3.4	DALI- / DALI-2 -ohjaussignaali	12
4	ENOCEAN.....	14
5	PROJEKTISSA KÄYTETYT KOMPONENTIT	15
5.1	Beckhoff.....	15
5.1.1	TwinCat	15
5.1.2	Beckhoffin fyysiset laitteet.....	15
5.2	Phoenix	17
5.2.1	UNO POWER	17
5.2.2	UNO DC-UPS	17
5.3	Eltako	18
5.3.1	Eltako TF-4FT / 4FT55	18
5.3.2	Eltako FBH65S/12V DC-wg.....	18
5.3.3	KL6583	19
5.4	Helvar.....	20
5.4.1	Digidim 311	21
5.4.2	Digidim 312	21
5.5	Glamox.....	21
5.5.1	Projektissa käytetty Glamoxin valaisin	21
6	PROJEKTIN MÄÄRITTELY	23
7	VALOJEN OHJELMOINTI	25
7.1	Määrittelyt.....	26
7.1.1	DALI-korttien määrittely.....	26
7.1.2	EnOcean-kortin määrittely	28
7.1.3	DALI-laitteiden määrittely	29
7.1.4	DALI-ryhmien määrittely.....	31
7.1.5	Kytkinten määrittely	33
7.1.6	Liiketunnistimien määrittely	35
7.2	Järjestelmäajan hakeminen	37
7.3	Toimilohkojen tekeminen	37

8	PROJEKTIN LOPPUTILANNE.....	42
8.1	Projektin analysointi	42
8.2	Parannukset / Kehitysehdotukset	43
	LÄHTEET	45
	LIITTEET	

1 JOHDANTO

Satakunnan ammattikorkeakoulun (SAMK) automaatiolaboratoriossa on päivitetty valaistus DALI-valoihin ja DALI-yhteensopiviin liiketunnistimiin lukuvuoden 2018-2019 hiihtoloman aikana opiskelijatyönä. Valaisimet ja liiketunnistimet oli siis asennettu jo valmiiksi, joten tämän opinnäytetyön aikana tehtäväksi jäi ottaa valaisimet käyttöön logiikan kautta ja ohjelmoida valot toimimaan halutulla tavalla kesän aikana. Laboratorion parven valaistukset on otettu käyttöön jo aikaisemmin Pauli Valon toimesta.

Työ aloitettiin jatkamalla sähkökeskusta lisäosalla. Tähän lisäosaan asennettiin logiikka ja logiikan varavirtalähde (UPS). Johtimia täytyi tuoda myös viereisestä alkuperäisestä keskuksesta, joten lisäosaan täytyi tehdä läpivientejä, jotta johtimien tuonnit onnistuisi näitisti suojassa. Läpi täytyi tuoda osa valaistuksen DALI-väylistä ja myös virta muuntajalle (230V).

Koska osaan valoista on käytetty hyväksi jo aikaisemmin vedettyjä $3 \times 1,5 \text{ mm}^2$ MMJ-johtoja, on näille valaisimille täytynyt vetää DALI-väylää varten oma $3 \times 1,5 \text{ mm}^2$ MMJ. Muutamit liiketunnistimista on vedetty myös omaan piiriinsä. Tästä syystä joidenkin valaisimien DALI-väylä on voitu vetää suoraan lisäosaan ja toiset valoista on vedetty $5 \times 1,5 \text{ mm}^2$ MMJ:llä alkuperäiseen sähkökeskuksen osaan. Tämän vuoksi nämä on täytynyt vetää keskuksesta toiseen läpivientien kautta. Mikäli valaistus olisi tehty alun perin DALI-yhteensopiviksi, olisi ne tehty heti $5 \times 1,5 \text{ mm}^2$ MMJ:llä, jolloin olisi helppo katsoa, mikä DALI-väylä on millekin valaisimille. Tämä helpottaa myöhempiä DALI-väylien ryhmitystä, jossa valitaan, mitkä valaisimet otetaan millekin väylälle.

Kun johdotukset saatiin vedettyä keskukselle, aloitettiin keskuksen täyttäminen komponenteilla. DIN-kiskoon lisättiin riviliittimiä, jotta kaikilta alueilta saatiin tuotua johtimet DALI-korteille. Laboratorio ja sen parvi on jaoteltu neljään DALI-alueeseen. Yhdeltä alueelta tulevat johtimet yhdistetään riviliittimien avulla sitä vastaavaan DALI-korttiin. Näin tehdään kaikkien alueiden kanssa.

Kytcentöjen valmistuttua alkoi valaistuksen ohjauksen suunnittelu ja ohjelmointi. Kuten edellä kävi ilmi, jäi tämän opinnäytetyön aiheeksi laboratorion alakerran valojen käyttöönotto. Aikaisemmin käyttöönotettujen parven valaisimiin ei tarvinnut puuttua. Tämän vuoksi itse logiikan käyttöönottoa ja korttien määrittelyä ei tarvinnut tehdä kuin näille uusille alakerran alueiden korteille.

2 KIINTEISTÖAUTOMAATIO VALAISTUKSEN OHJAUKSESSA

Kiinteistö- eli rakennusautomaatiota ja varsinkin sen pääjärjestelmää voidaan hyvin pitää koko kiinteistön aivoina. Siihen pyritään liittämään kaikki rakennuksen eri järjestelmät kuten lämmitys-, ilmanvaihto-, ilmastointi- ja valaistusjärjestelmät, sekä muut kiinteistön hallintaan liittyvät järjestelmät kuten turvajärjestelmät, kuten esimerkiksi palohälytys-, murtohälytys- ja kulunvalvontajärjestelmät. Ei ole väliä, mikä loogikka on valittu kiinteistön pääjärjestelmäksi, sillä lähes kaikki tieto kulkee sen kautta. Alajärjestelmä ei kuitenkaan kierrätä tietoa pääjärjestelmään, mikäli toiminta malli on selvä alajärjestelmälle. Tämä tarjoaa etuja, kun kaikki tieto on keskitettynä. Se mahdollistaa tarkkailun ja hallinnan yhden järjestelmän kautta, ja pystytään estämään risteävät ohjaukset esimerkiksi lämmityksessä ja viilennyksessä.

Kiinteistöautomaatiojärjestelmässä on myös se etu, että huollot ovat oikea-aikaisempia eli ei tehdä turhia, liian aikaisia huoltoja ja eikä myöskään liian myöhäisiä korjaustoimenpiteitä. Automaatiojärjestelmistä saadaan paljon enemmän dataa, jolloin voidaan keskittyä oleellisiin toimenpiteisiin. Kiinteistöautomaatio on yleisimmin käytössä institutionaalisilla käyttäjillä, teollisella tai kaupallisella puolella, mutta standardointien kautta yleistyy laajalti myös yksityiselle puolelle.

(Building automation 2019.)

Kiinteistön valaistusta voidaan ohjata suoraan kiinteistöautomaation tai pelkän valaistuksenohjauksen (alajärjestelmä) avulla. Järjestelmän ohjaamisessa lähes kaikki on mahdollista. Sitä voidaan ohjata esimerkiksi ajan, liikkeen, valaistusarvon tai näiden eri yhdistelmien avulla. Valaistuksen ohjaus voidaan myös toteuttaa kenttäväyläpohjaisella digitaalisella valaistusrajapinnalla kuten DALI-väylällä. DALI-väylässä olevista liitäntälaitteista saadaan myös tietoa, mikäli laite on vaurioitunut tai signaali ei etene tai on muuten kadonnut. (Lighting 2019.)

Etuina voidaan pitää taloudellisia säästöjä ja olosuhteiden parantumista. Taloudellisia etuja voidaan saavuttaa suoraan älykkäällä lämmityksellä, mutta myös valaistuksen ohjauksella. Ei tarvita ketään erikseen syyttämään valoja ja sammuttamaan niitä. Suu-

rimpia hyötyjä saadaan julkisissa tiloissa, joihin kokoontuu useita ihmisiä ja ihmismäärät vaihtelevat. Näissä esimerkiksi ilmanvaihto olisi hyvä olla automatisoitu, jolloin ilmanlaatu pysyisi hyvänä, kun ilmanvaihtokoneet saisivat antureilta tietoa tehonsäätöön. Automaatio on myös tehokas ase ilmastonmuutosta vastaan, sillä rakennushankkeessa automaation kustannukset ovat vain muutamia prosentteja, ja sillä voidaan pienentää ratkaisevasti energian tarvetta ja ylläpitokuluja. (Rakennuksetkin saavat hermot ja aivot 2019)

3 DALI

3.1 Yleistä tietoa DALI:sta

DALI-lyhenne tulee sanoista Digital Addressable Lighting Interface. Se on digitaalisesti hallitun valaistuksen protokolla, joka pohjautuu standardiin IEC 62386 valojen ja antureiden välillä. Se helpottaa skaalautuvien ja joustavien valaistuksien luontia. DALI kehitettiin alkujaan vanhan analogisen (0-10V tai 1-10V) ohjaustavan kilpailijaksi ja korvaajaksi. Tavoitteena on valojen ohjauksen täysi digitaalisuus. Standardin tavoitteena on, että eri valmistajien laitteet toimisivat myös tarvittaessa ristiin, vaikka ne olisivat samassa väylässä keskenään (DALI AG 2019).

Vuonna 2014 tuli uusi versio standardista IEC 62386, joka johti DALI-2:n tulemiseen. DiiA (Digital Illumination Interface Alliance) on avoin, maailmanlaajuinen valaistusyritysten järjestö, joka pyrkii DALI-2 -sertifioinnilla parantamaan yhteen toimivuutta ja avoimuutta verrattuna aikaisempiin DALI-järjestelmiin. (Introducing DALI 2019.)

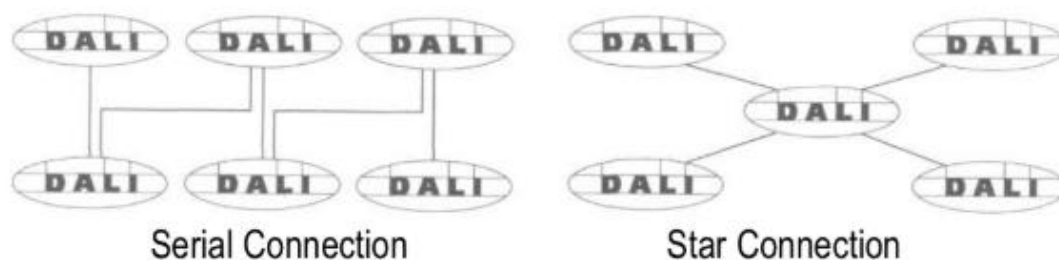
3.2 DALI- / DALI-2 -kaapelointi ja -topologia

Kaapelointi on DALI-laitteiden välillä todella helppoa. Valojen käyttövirta ja tieto saadaan kulkemaan helposti samassa kaapelissa (Kuva 1). Jotkut anturit voivat saada myös käyttövirtansa DALI-väylästä. DALI-väylässä ei tarvitse välittää polarisaatiosta (+/-), eli tieto kulkee, vaikka jossakin olisikin mennyt + ja - signaalit ristiin. DALI-väylän jännitealenema saa olla maksimissaan 2 V. Jännitealeneman maksimista saadaan laskennalliset maksimipituudet, jotka ovat 300 m johtimen ollessa 1,5 mm², 150 m johtimen ollessa 0,75 mm² ja 100 m johtimen ollessa 0,5 mm².



Kuva 1. Havainnollistaminen kaapelista, jossa DALI-väylä ja käyttövirta samassa (TYA670W - KNX/DALI Gateway värilämpötilan säädöllä 2018)

Koska DALI-laitteet nimetään yksitellen omilla lyhyillä osoitteilla, voi Dali-väylässä laitteet olla joko sarjassa, tähtimuodostelmassa tai jopa näiden yhdistelmänä (Kuva 2).



Or in any combination of the two topologies

Kuva 2. Havainnollistaminen DALI-väylän topologiasta (DALI Lighting Control Solutions Explained 2012)

3.3 DALI- / DALI-2 -laitteiston rakenne

Yhdessä DALI -väylässä voi olla korkeintaan 64 ohjauslaitteita (control device) tai 64 valaisinta (control gear), sekä korkeintaan 16 ohjattavaa ryhmää (Group) ja 16 erilaista valaistustilannetta (Scene). Väylä tarvitsee virransyötön, tyypillisesti 250 mA ja 16 V. (DALI-2 systems & product types 2019.)

DALI Groups

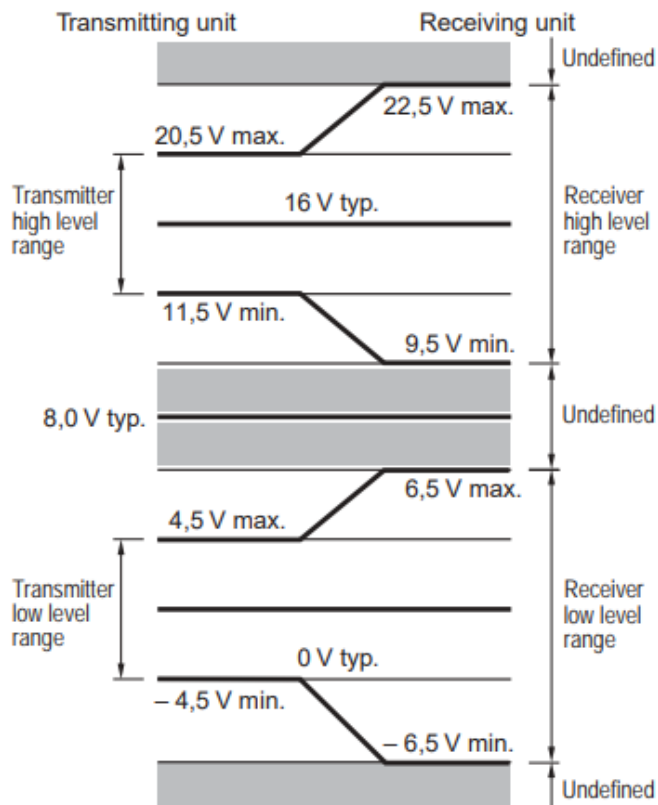
DALI-valaisimet voidaan siis asettaa ryhmiin. Yhdessä DALI-väylässä voi olla 16 eri ryhmää ja jokainen valaisin voidaan asettaa haluttuun ryhmään tai ryhmiin. Valaisin voi olla yhdessä tai useammassa ryhmässä samanaikaisesti, tai voi olla, ettei valaisin kuulu mihinkään ryhmään. Tämän vuoksi on mahdollista asettaa vaikka omakotitalon valaisimet kaikki samaan väylään ja ohjata valoja huonekohtaisesti, mikäli ryhmät on luotu huonekohtaisesti. (luxCONTROL DALI manual 2018.)

DALI Scenes

DALI-valaisimeen voidaan asettaa 16 erilaista sceneä, joissa voidaan asettaa kyseiseen valaisimeen haluttu valaistusarvo. Scenen avulla voidaan asettaa valaisimet asettumaan haluttuihin valaistusarvoihin. Myös himmennykselle voidaan asettaa haluttu nopeus, jolla valaisin saavuttaa asetetun valaistusarvon. Esimerkiksi huoneen 10 eri valaisinta voidaan asettaa kaikki asettumaan 3 sekunnissa haluttuihin valaistusarvoihin. Valaistusarvot asetetaan yksilöllisesti jokaisen valaisimen muistiin, esimerkiksi joka toisen voi asettaa himmeäksi ja joka toisen kirkkaaksi. Valaisimen saadessa käskyn ”Go to scene 1” siirtyy jokainen huoneen valaisin 3 sekunnissa asetettuun tilaan, joka toinen valaisin himmeänä ja joka toinen kirkkaana. (luxCONTROL DALI manual 2018.)

3.4 DALI- / DALI-2 -ohjaussignaali

Ohjaussignaalin nopeus on 1200 bittiä sekunnissa, joka on melko hidas. Tämä parantaa väylän häiriönsietokykyä. Signaalin Low-tilaksi on määritelty lähettävälle puolelle väli -4,5 V - +4,5 V ja vastaanottavalle puolelle väli -6,5 V - +6,5 V. High-tilaksi taas on määritelty lähettävälle puolelle 11,5 V - 20,5 V ja vastaanottavalle 9,5 V - 22,5 V. Nämä myös määrittävät edellä mainitun jännitteenaleneman 2 V (Kuva 3). DALI-väylässä käytetään Manchester-koodia (DALI AG 2001).



Kuva 3. Havainnollistaminen DALI-väylän jänniterajoista (DALI AG 2001)

4 ENOCEAN

EnOcean on patentoidun langattoman energiankeräystekniikan kehittäjä. Yhtiö tuottaa anturiratkaisuja akuttomiin sovelluksiin, joita käytetään rakennusten ja teollisuuden automaatioissa, älykodeissa ja valaistuksen ohjauksessa. EnOcean-tuotteet perustuvat pääosin pienikokoisiin energianmuuntajiin sekä erittäin pienitehoiseen elektroniikkaan ja radiotekniikkaan. EnOcean-moduulit (esim. kytkimet) perustuvat pienois-energiamuuntajiin, jotka muuntavat liike- tai lämpötilaerot sähköenergiaksi. Langattomuus perustuu 868 MHz:n (Euroopassa) radiotaajuudelle, joka on standardin (ISO / IEC 14543-3-1X) mukainen ja osoittautunut luotettavaksi. (Company Profile 2019.)

EnOcean on yksi perustajayhtiöistä EnOcean alliance -järjestössä. EnOcean tarjoaa kehittelemänsä tekniikan ja lisenssit patentoimilleen keksinnöille, kaikille järjestön jäsenille käytettäväksi. (EnOcean 2019.)

5 PROJEKTISSA KÄYTETYT KOMPONENTIT

5.1 Beckhoff

Beckhoff on avoimien automaatiojärjestelmien toimittaja. Järjestelmät pohjautuvat PC-pohjaiseen ohjaustekniikkaan. Tuoteperhe on laaja. Siihen kuuluu kenttäväylä-komponentteja, liikkeenohjaustuotteita, teollisuus-PC:t ja ohjauspaneelija, sekä automaatiosovelluksien ohjelmistoja. Beckhoffin tuotteita voidaan käyttää erillisinä komponentteina, tai niillä voidaan toteuttaa kokonaisia ohjausjärjestelmiä. Beckhoff on maailmanlaajuisesti tunnettu toimittaja, jonka tuotteita käytetään työstökeskusten ohjauksesta kiinteistöjen automaatiosovellusten ohjaamiseen. (New automation technology 2019.)

5.1.1 TwinCat

TwinCat on Beckhoffin automaatio-ohjelmisto, joka voi muuntaa lähes minkä tahansa tietokoneen reaaliaikaiseksi ohjaimeksi. Twincat pystyy korvaamaan tavanomaiset PLC- ja NC- / CNC-ohjaimet sekä käyttölaitteet. TwinCAT suorittaa ohjausohjelmia reaaliajassa, sekä tarjoaa kehitysympäristön ohjelmoinnille, diagnosoinneille ja määrittelyille. Windows-ohjelmat, kuten visualisointiohjelmat tai Office-ohjelmat, voivat käyttää TwinCAT-tietoja Microsoftin rajapintojen kautta tai voivat suorittaa komen-toja. (TwinCAT – PLC and Motion Control on the PC 2019.)

5.1.2 Beckhoffin fyysiset laitteet

C6015

PLC:nä (Programmable Logic Controller) eli ohjelmoitavana logiikkana projektissa toimii Beckhoffin C6015. Se on pienikokoinen ja tehokas teollisuus-PC. Moniytimisellä tuella C6015:ää voidaan käyttää samanaikaisesti tehokkaisiin automaatio-, visualisointi- ja viestintäsovelluksiin. Passiivisesta jäähtytyksestä huolimatta C6015 sopii myös lämpimiin ympäristöihin, aina +55 °C lämpötilaan saakka. Lisäksi asennustilan ollessa vain 82 x 82 x 40 mm, se on pienimpiä tämän hetken teollisuus-PC:itä. C6015

on varustettu 30 Gt:n tallennustilalla ja Windows 10 -käyttöjärjestelmällä. (C6015 – Ultra-compact Industrial PC 2019)

BK1150

BK1150-väylämuunnin yhdistää Ethernet-pohjaisella EtherCatilla elektroniset KL- / KS-päätekortit kenttäväylän tasolla logiikkaan. Yksikkö koostuu väyläkytkimestä (BK1150) ja yhdestä kortista aina 64 korttiin saakka (esim. KL6581). K-väylän jatkeella saadaan maksimissaan 255 korttia yhteen yksikköön. Yksikössä pitää myös olla pääte kortti KL9010. Ylempi rajapinta EtherCat-verkkoon yhdistää kytkimen logiikkaan. Alempaan RJ45-liitäntään voi liittää muita haluttuja EtherCat-laitteita, jotka halutaan samaan verkkoon. (BK1150 – EtherCAT ”Compact” Bus Coupler 2017.)

KL1114

KL1114 on nelikanavainen digitaalitulo-kortti. Se toimii 24 V tasavirtajännitteellä ja indikoi neljän ledin avulla kunkin tulokanavan tilaa. (KL1114 – 4-channel digital Input terminal 24 V DC 2016.)

Tähän korttiin tulee Input-tiedot logiikan varavirtalähteeltä, josta voidaan diagnosoida mahdolliset sähkökatkot ja akun diagnostiikkaa.

KL6581

KL6581 on EnOcean master terminal -kortti. Se voi vastaanottaa antureilta dataa sekä lähettää sitä toimilaitteille. KL6581 on linkki KL6583 EnOcean-lähetin- ja vastaanotinmoduulien välillä. Enintään kahdeksan KL6583-moduulia voidaan kytkeä KL6581 EnOcean -päätelaitteeseen. KL6583-moduulit on ketjutettu CAN-pohjaisella väylällä logiikassa olevaan KL6581-korttiin. Johdin on nelinapainen, joista yhdessä on +24 V, toisessa 0 V, kolmannessa DATA+ ja neljännessä DATA-. Dataväylän enimmäispituus on 500 m. KL6581:seen voidaan maksimissaan liittää 8 kpl KL6583-moduuleja. KL6583-moduulista kerrotaan vielä lisää kappaleessa 5.3.3. (KL6581 – EnOcean master terminal 2019.)

KL6821

KL6821 on DALI- / DALI-2 -master- ja virtalähdekortti. Se mahdollistaa jopa 64 DALI-laitteen kytkemisen. Tämän kortin kautta kulkee kaikki data DALI-laitteisiin.

DALI-laitteiden käyttämiseen ei tarvita muita kortteja. (KL6821 – DALI / DALI 2 multi-master and power supply terminal 2019.)

KL9010

KL9010 on väyläpäätekortti. Sen käyttö on välttämätöntä tiedonsiirron mahdollistamiseen väyläkytkimen ja muiden väyläkorttien välillä. Ilman tätä väylä jää avoimeksi. Jokainen kokoonpano on päätettävä oikeassa päässä väyläpääteellä. Väyläpääteellä ei ole muuta toimintoa tai liitäntämahdollisuutta. (KL9010 – End terminal 2019.)

5.2 Phoenix

Phoenix Contact GmbH & Co on kansainvälisesti kasvava sähköisen liitäntä- ja automaatioteknologian sekä ylijännitesuojusratkaisujen toimittaja. Se tarjoaa yli 60 000 tuotteen valikoiman liittimestä ohjaukseen. Perustettu vuonna 1923. (Yrityksen historia 2019.)

5.2.1 UNO POWER

UNO POWER on virtalähde, josta saadaan 100 W ja 24 V tasavirralla. Se on asennettavissa helposti DIN-kiskoon. Sen koko on vain 55 x 90 x 84 mm ja se toimii hyvin -25°C ... +70°C lämpötiloissa. (Power supply unit - UNO-PS/1AC/24DC/100W 2019.)

5.2.2 UNO DC-UPS

UPS (Uninterruptible Power Supply) tarkoittaa keskeytymättömän virransyötön moduulia. Integroidulla energianvarastoinnilla varustetut UPS-moduulit säästävät erityisesti tilaa, kun UPS-moduuli ja energianvarastointi on samassa kotelossa. UNO DC-UPS:n koko on 10 x 90 x 84 mm ja se painaa kilon. Se toimii lämpötiloissa -15 °C ... + 50 °C. Projektissa tämä UPS varmistaa logiikan virrat sähkökatkokkien varalta, jotta ohjaukset palautuisivat mahdollisimman nopeasti katkoksen jälkeen. Varavirtalähteen akku kestää noin 45 min katkoksen. (Uninterruptible power supply - UNO-UPS/24DC/24DC/60W 2019.)

5.3 Eltako

Eltako on yksi johtavista kytkinratkaisujen kehittäjistä ja toimittajista. Yritys on perustettu vuonna 1949 ja sen nimi tulee saksankielisistä sanoista "Elektrischer Tast-Kontakt", mikä tarkoittaa sähkömekaanista impulssikytkintä (Innovaatioita perinteiden kautta 2019). Eltako on yksi EnOcean-järjestön jäsenistä ja standardin edistäjistä. Eltako on kuulunut järjestöön vuodesta 2009. (Eltako is a promoter of the EnOcean Alliance 2019.)

5.3.1 Eltako TF-4FT / 4FT55

Eltako TF-4FT / 4FT55 on kaksiosainen valkoinen langaton painike. Sen koko on 84 x 84 x 16 mm. Se luo energian painiketta painettaessa. Tämän takia se ei tarvitse johtoja, eikä sillä ole valmiustilakulutusta. Langaton kaksiosainen painike voi lähettää kahdeksan erilaista analysoitavaa signaalia. Painamalla ylhäältä kahta näppäintä eriaikaan tai samaan aikaan (3 eri signaalia) tai alhaalta vastaavanlaisesti kahta näppäintä erikseen tai samaan aikaan. Näillä saadaan aikaan jo kuusi erilaista signaalia. Sekä ylös että alas eripuolilta samaan aikaan painettaessa saadaan kaksi erilaista signaalia. Näin erilaisia signaaleja on kahdeksan. Asennuspohja voidaan ruuvata tasaiseen pohjaan tai liimata seinään, lasiin tai huonekaluun mukana tulevalla kaksipuoleisella taralla. Kytkimessä on myös kiinnityspaikat ruuveille kojerasiaan kiinnittämistä varten. (Tap-radio, Innovatiivinen langaton taloautomaatiojärjestelmä EnOcean teknologialla ja älykoti sovelluksilla 2017.)

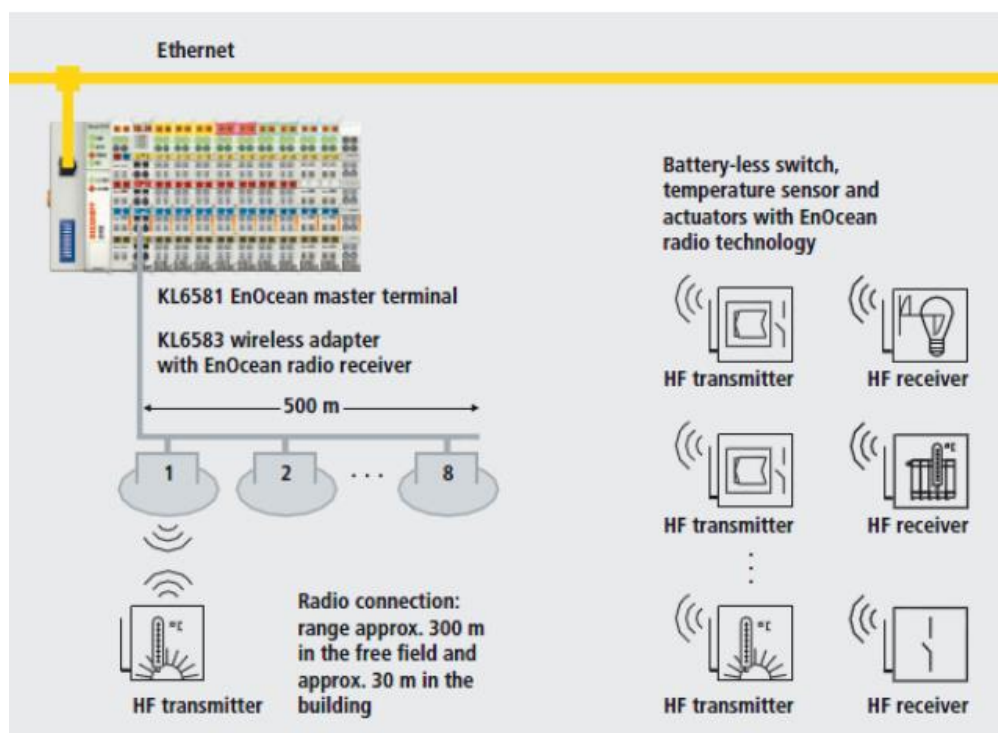
5.3.2 Eltako FBH65S/12V DC-wg

Eltako FBH65S/12V DC-wg on langaton läsnäolo-/valoanturi pinta-asennukseen. Sen koko 84 x 84 x 39 mm. Anturit toimivat kiinteästi asennetuilla aurinkokennoilla tai 12 voltin jännitteellä. Toimitushetkellä laite on täysin lataamaton. Sitä tulee ladata ensimmäisen kerran 12 voltin tasavirtalähteellä 3 minuutin ajan tai aurinkokennoilla 10 tunnin ajan täydessä päivänvalossa.

Havaitessaan läsnäolon lähettää läsnäoloanturi signaalin EnOcean-moduulille. Vasta, kun anturi ei ole minuuttiin havainnut mitään, palautuu se normaalitilaan. (Langaton Eltako-järjestelmä 2017.)

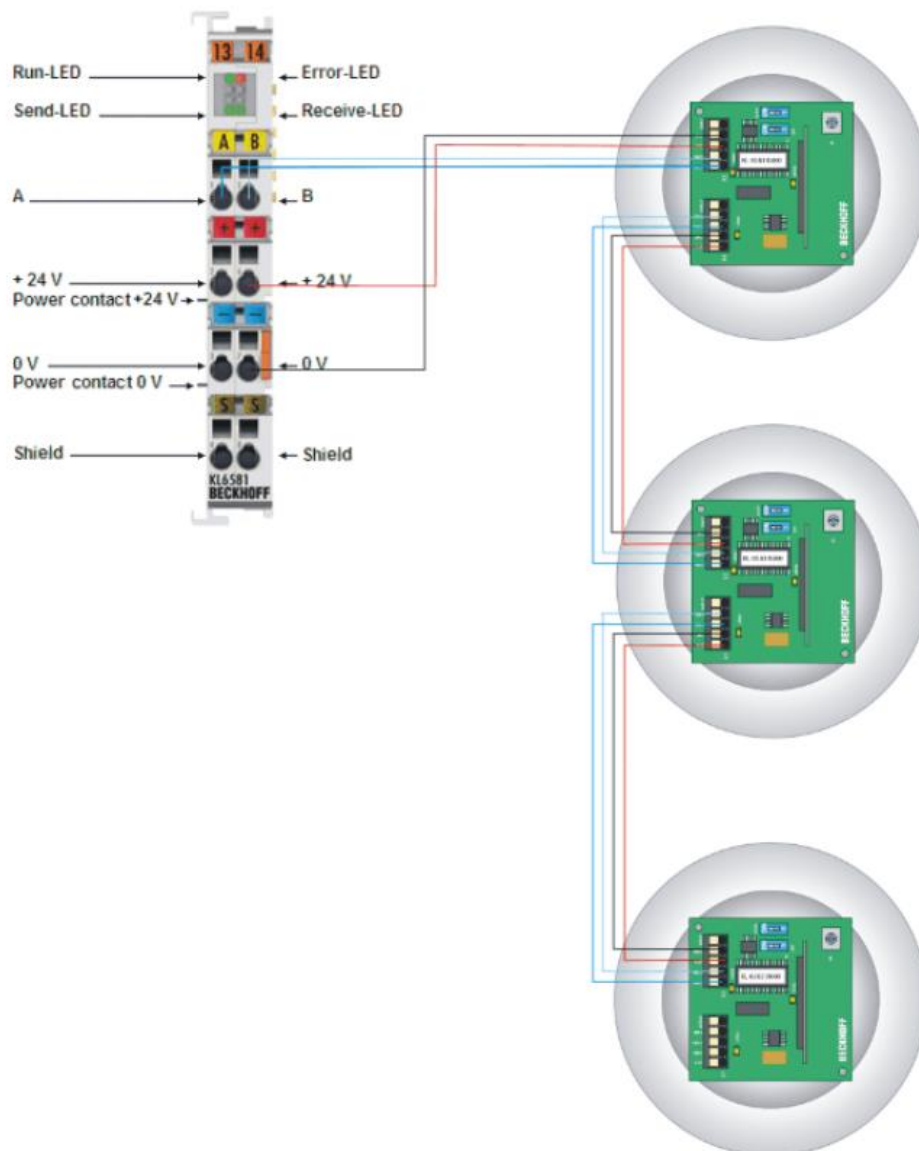
5.3.3 KL6583

KL6583 on Beckhoffin EnOcean-moduuli, joka mahdollistaa EnOcean-tiedonsiirron. Moduuli toimii siis antennina ja mahdollistaa väyläyhteyden (BUS) KL6581-EnOcean-päätelaitteeseen (korttiin). Tiedot siirtyvät päätelaitteeseen kaksijohdin-väylän kautta. Moduuli tarvitsee 24 V jännitteen, joten minimijohdinmäärä tämän toimimiseen on neljän johtiminen kaapeli. Väylän pituus voi olla maksimissaan 500 m ja yhdessä väylässä voi olla jopa kahdeksan KL6583-moduulia (Kuva 4 ja 5.). Teoreettisesti siis voidaan kattaa reilun kilometrin halkaisijaltaan oleva ala. (KL6583 – EnOcean transmitter and receiver 2019.)



Kuva 4. KL6583-moduulin liitântäkuvaus

Cabling example



Kuva 5. Esimerkki KL6583-moduulien liittämisestä sarjaan

5.4 Helvar

Helvar on valaistusratkaisuihin erikoistunut suomalainen yritys, joka on perustettu jo vuonna 1921. Alkujaan yritys perustettiin Suomen (Helsinki) ja Puolan (Varsova) välisen viennin ja tuonnin vuoksi. Tästä myös nimen kerrotaan olevan peräisin (Philip Aminoff on tiiminrakennuksen mestari 1.9.2016). Helvar tarjoaa valaistuksen ohjauslaitteita ja valaisinkomponentteja. Tuotteita voi käyttää joko yhdessä suuressa älykäässä valaistusratkaisussa tai käyttää vain yhtä yksittäistä komponenttia. (Helvar yrityksenä 2019.)

5.4.1 Digidim 311

Digidim 311 on PIR-liiketunnistin, joka voidaan kiinnittää kattoon tai seinään, ja joka vaatii asennuskotelon. Sen avulla voidaan saavuttaa merkittäviä energiansäästöjä, kun sitä käytetään osana DALI-järjestelmää. Sammutusviive on asetettava Helvarin valonohjausohjelmilla: Designer tai Toolbox. (Upotettava PIR-liiketunnistin (311) 2016.)

5.4.2 Digidim 312

Digidim 312 on DALI-multisensori, joka sisältää valoanturin, passiivisen infrapunaliiketunnistimen (PIR) ja infrapunavastaanottimen (kauko-ohjausta varten). Kauko-ohjaimella pystyy tekemään perusohjelmointeja ja säätämään valaistustasoa. Tarkemmat ohjelmoinnit vaativat Helvarin valonohjausohjelman (Designer tai Toolbox). Asennus voidaan tehdä kattoon tai valaistusrakenteisiin. (Digidim 312 DALI Multisensori 2013.)

5.5 Glamox

Glamox on yksi Euroopan johtavista valaisinvalmistajista. Yritys keskittyy toimittamaan korkealaatuisia valaistusratkaisuja julkisiin tiloihin, kuten oppilaitoksiin, toimisto- ja teollisuusrakennuksiin, terveydenhuollon tarpeisiin sekä hotelleihin ja ravintoloihin. (Glamox Oy 2019.)

5.5.1 Projektissa käytetty Glamoxin valaisin

C10-S1 225x1200 LED 6600DALI 840 2XSU

C10-S1 225x1200 LED 6600DALI 840 2XSU on pinta-asennettava tai ripustettava, valkoiseksi maalattu teräsrunkoinen valaisin. Tarkempia tietoja ovat mm:

- IP20
- Valonlähde LED
- Valonvoimakkuus 7091 lm
- Liitäntälaitte DALI

- Värisävy CRI>80, 4000K
- Optiikka SLSU

(C10-S1 2019.)

6 PROJEKTIN MÄÄRITTELY

Tämän opinnäytetyön tavoitteena oli saada SAMK:in automaatiolaboratorioon älykäs valaistus, joka olisi ajastimen, liiketunnistimien ja langattomien kytkimien kanssa ohjattu. Ajastimen haluttiin ohjaavan valoja päälle arkisin ja lauantaisin kello 07:00-18:00 välisellä ajalla. Liiketunnistimien haluttiin taas ohjaavan valoja päälle tämän ajan ulkopuolella niin, että valot olisivat tunnin päällä viimeisestä liikkeen tunnistamisesta tai läsnäolon rekisteröinnistä. Kytкимиä haluttiin kaksi erilaista, ns. siivous- / pääkytkin ja alueellisesti toimiva peruskäyttökytkin. Kun ollaan viikkoajastimen toiminta-ajan ulkopuolella, käyttökytkimen ajastin sammuttaa valot kolmen tunnin kulluttua niiden päälle laittamisesta, jollei alueella ole ollut liikettä viimeiseen puoleen tuntiin. Siivous- / pääkytkimessä on 10 tunnin viive. Tällöin valaisimet sammutetaan, ellei ajastimen pakkokytkenä ole päällä. Näin tapahtuu, jos alueella ei ole käyttökytkimestä laitettu uudelleen valoja päälle tai liikettä ei ole havaittu. Konenäkölaboratorio on jätetty pois ajastimen, kytkinten ja liiketunnistimien vaikutusalueesta. Konenäköalueelle tehdään kokonaan oma visuaalinen käyttöliittymä, jolla voidaan ohjata jokaista konenäön alueen kuutta valoa yksilöllisesti.

Käyttökytkimellä saa valot pois ja päälle sekä säädettyä valojen kirkkautta ylös tai alas. Kytkiminä on Eltakon TF-4FT, jossa on neljä eri painiketta. Vasemmalta puolelta ylhäältä saa valot päälle ja alhaalta saa valot pois. Oikealta puolelta ylhäältä saa lisää kirkkautta ja alhaalta saa vähennettyä kirkkautta. Kirkkauden säätö toimii 100/75/50/25 % -periaatteella, eli kytkimellä pystyy valitsemaan valaistusarvon 250, 230, 210 ja 190. Valaistusarvot eivät korreloi lineaarisesti prosenttiarvoja, vaan ne ovat mittauksilla todettuja. Kirkkautta saa jokaisella painalluksella 20 valaistusarvon yksikköä lisää, kunnes maksimi 250 on saavutettu. Himmennettyä saa samoin 20 valaistusarvon yksikköä per painallus niin kauan, kunnes minimi 190 on saavutettu. Päälle painikkeesta valot syttyvät aina arvoon 230.

Siivous- / pääkytkin toimii hieman eri tavalla. Kumpaa tahansa yläpainiketta painaessa valot syttyvät kirkkaimmalle (arvoon 250) ja jompaa kumpaa alapainiketta painettaessa palautuu edellinen arvo, joka oli valaisimissa ennen kytkimeen koskemista. Täl-

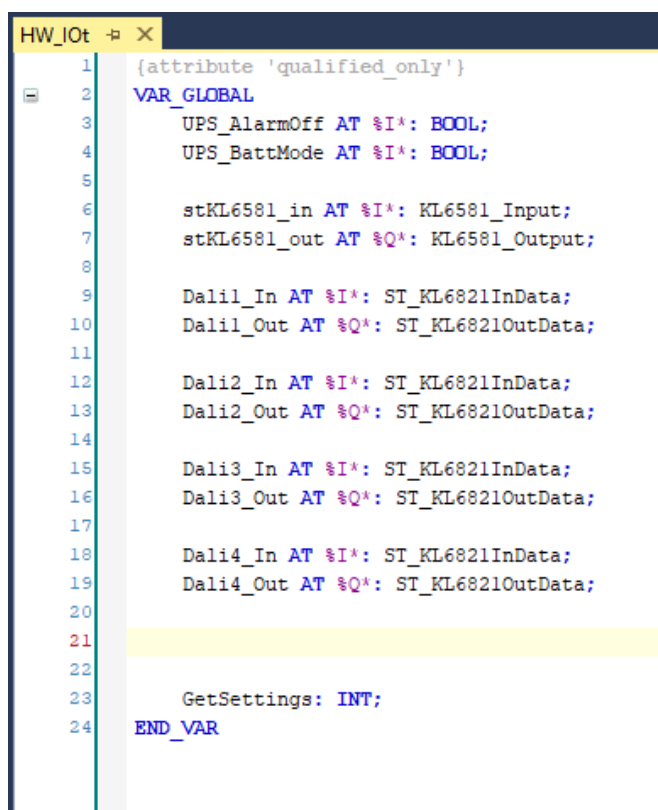
lön valaistus palautuu ns. ”automaatille”. Koko automaatiolaboratorion alakerran valot voidaan sammuttaa, kun jompaa kumpaa alapainiketta pidetään pohjassa vähintään kaksi sekuntia.

Automaatiolaboratorion alakerta on jaettu kolmeen eri DALI-alueeseen, ja parvi on vielä lisäksi omansa. Näin ollen laboratorio on jaettu kokonaisuudessaan neljään eri DALI-alueeseen. Jokaisella alueella on omat määritellyt ryhmänsä, joita voi ohjata mm. käyttökytkimillä. Jokaisella alueella on 4-7 eri ryhmää. Ryhmät on jaoteltu omiksi kokonaisuuksikseen ja helpottamaan tietyn alueen hallintaa. Esimerkiksi automaatioluokan tussitaulun yllä olevat valot ovat oma ryhmänsä, jotta niitä saadaan ohjattua tarvittaessa omassa ryhmässään pois päältä (LIITE1).

7 VALOJEN OHJELMOINTI

Kun ryhmät ja muut toiminnallisuudet oli suunniteltu, aloitettiin valojen ohjelmointi. Tämä täytyi aloittaa määrittelemällä uudet DALI-kortit ja EnOcean-kortti, joiden kautta logiikka kommunikoi antureiden ja toimilaitteiden kanssa. Kun kaikki määrittelyt oli tehty, alkoi vasta itse toiminnallisuuksien ohjelmointi.

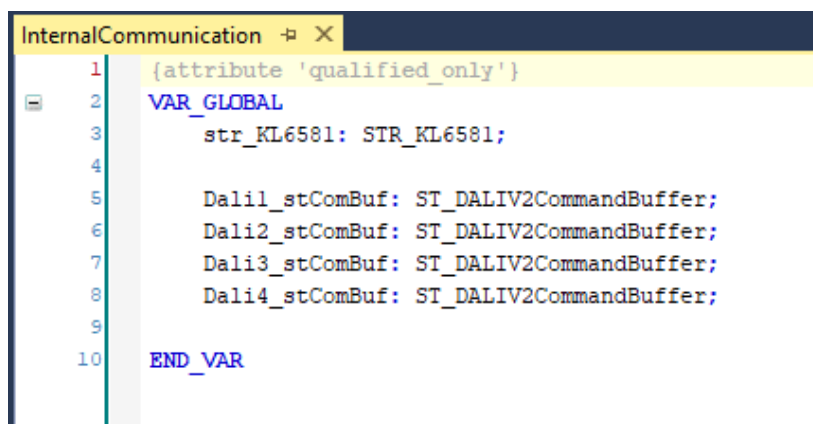
Määrittelyt ja ohjelmoinnit tapahtuivat TwinCat-ohjelmalla, ja Beckhoffin omia DALI- ja EnOcean-kirjastoja käyttämällä. Näistä kirjastoista löytyy kaikki tarvittavat toimilohkot. Muuttujalistoihin täytyi esitellä Hardware-muuttujat ja sisäiset kommunikaatiomuuttujat EnOcean-kortille ja DALI-korteille. Korteilla kulkee tieto molempiin suuntiin, joten piti määrittellä sekä sisään tuleva että ulos kulkeva data. EnOcean-kortti on KL6581 ja DALI-kortti on KL6821 ja näille kaikille on omat esittelynsä (Kuva 6).



```
HW_IOT -p x
1 {attribute 'qualified_only'}
2 VAR_GLOBAL
3     UPS_AlarmOff AT %I*: BOOL;
4     UPS_BattMode AT %I*: BOOL;
5
6     stKL6581_in AT %I*: KL6581_Input;
7     stKL6581_out AT %Q*: KL6581_Output;
8
9     Dali1_In AT %I*: ST_KL6821InData;
10    Dali1_Out AT %Q*: ST_KL6821OutData;
11
12    Dali2_In AT %I*: ST_KL6821InData;
13    Dali2_Out AT %Q*: ST_KL6821OutData;
14
15    Dali3_In AT %I*: ST_KL6821InData;
16    Dali3_Out AT %Q*: ST_KL6821OutData;
17
18    Dali4_In AT %I*: ST_KL6821InData;
19    Dali4_Out AT %Q*: ST_KL6821OutData;
20
21
22
23    GetSettings: INT;
24 END_VAR
```

Kuva 6. Hardware-muuttujien esittely

Tämän jälkeen täytyi esitellä sisäisten kommunikaatioiden muuttujat (Kuva 7).



```
InternalCommunication -# X
1 {attribute 'qualified_only'}
2 VAR_GLOBAL
3     str_KL6581: STR_KL6581;
4
5     Dali1_stComBuf: ST_DALIV2CommandBuffer;
6     Dali2_stComBuf: ST_DALIV2CommandBuffer;
7     Dali3_stComBuf: ST_DALIV2CommandBuffer;
8     Dali4_stComBuf: ST_DALIV2CommandBuffer;
9
10 END_VAR
```

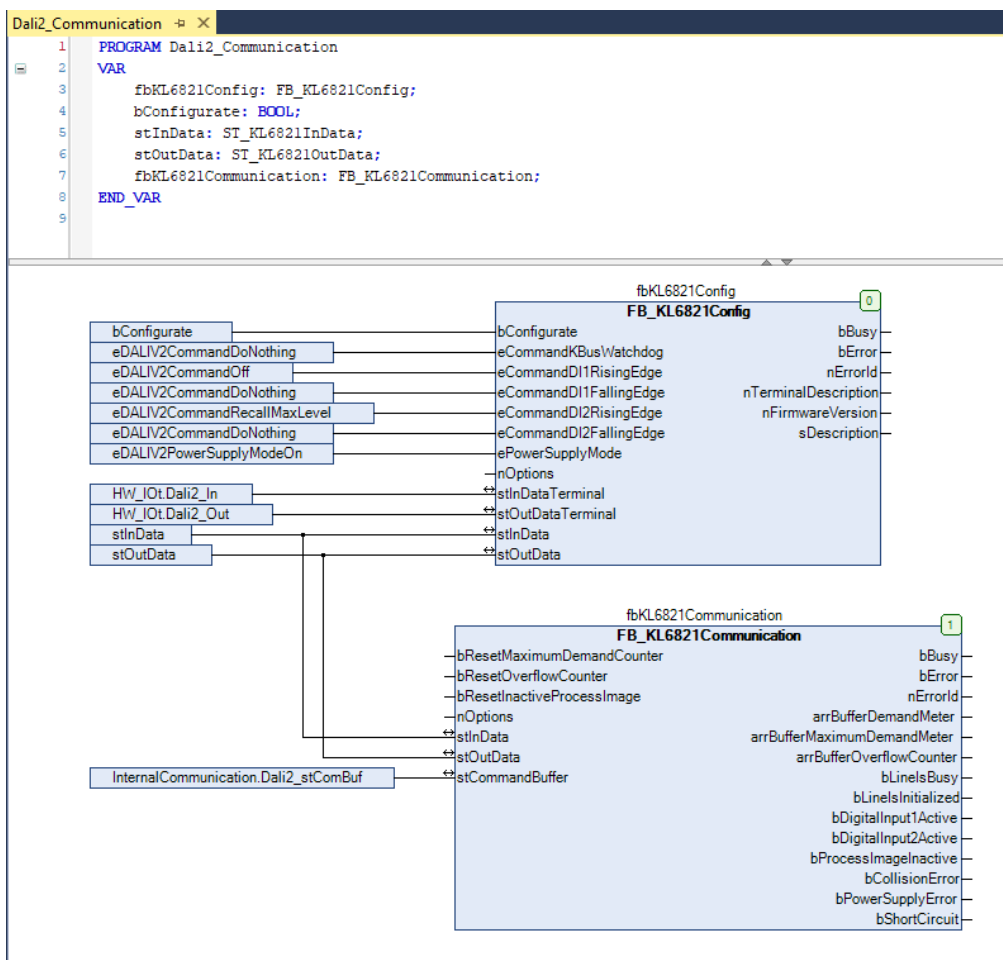
Kuva 7. Sisäisen kommunikaation muuttujien esittely

7.1 Määrittelyt

Seuraavissa kappaleissa kerrotaan käytettyjen korttien, laitteiden ja valaisinryhmien määrittelyistä.

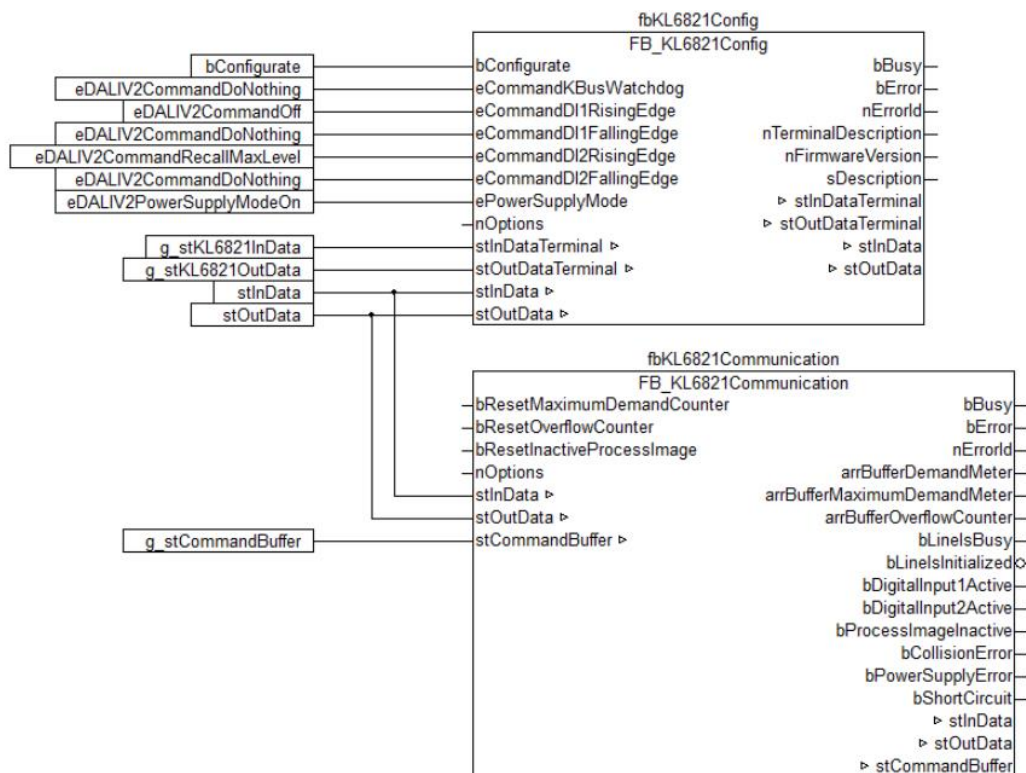
7.1.1 DALI-korttien määrittely

Beckhoffin DALI-kirjastossa on hyvät esimerkit, joiden avulla saadaan kortit määritettyä, jotta saadaan data kulkemaan kyseisellä DALI-väylällä. Määrittely kannattaa tehdä omaksi ohjelmakseen (program) (Kuva 8).



Kuva 8. Esimerkki käytetystä ohjelmasta

DALI-väylän eli -kortin määrittelyyn tarvitaan FB_KL6821Config- ja FB_KL6821Communication-toimilohkoja. Input stCommandBuffer määrittää, mille DALI-väylälle määrittelyt ollaan tekemässä. Tässä tapauksessa tulee huomioida, käytetäänkö esimerkiksi Dali3_stComBuf- vai Dali2_stComBuf-viittauksia. Eli päätetään, kumpi muuttuja asetetaan, jolloin vastaava väylä/kortti määritellään. Alla on esimerkkikuva Beckhoffin sivuilta, josta saa erittäin hyviä ohjeita ja paljon tietoa (Kuva 9). Kuvasta poiketen nimeämiset ovat luonnollisesti erilaiset tässä projektissa, mutta stInDataTerminal-Inputtiin tuleva muuttuja on nimetty HW_IO.Dali1_In. Tämä muuttuja on varattu kortin tuleviin ja lähteviin arvoihin.

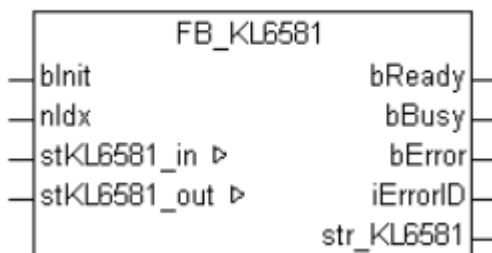


Kuva 9. Beckhoffin esimerkki DALI-kortin määrittelystä (FB_KL6821Config 2019)

7.1.2 EnOcean-kortin määrittely

Myös EnOcean-kortti pitää määrittellä. Tämä tapahtuu samoin periaattein kuin DALI-kortin määrittely. FB_KL6581-toimilohkon avulla saadaan väylä avattua ja aktivoitua datan kulku. Toimilohkoon pitää linkittää tämän projektin HW_IO-muuttujista oikeat muuttujat, jolloin tiedonkulku alkaa, kun toimilohko saa aktivointipyyynnön (Kuva 10.).

FB_KL6581



Kuva 10. Esimerkki EnOcean-kortin määrittelytoimilohkosta (FB_KL6581 2019)

EnOcean-datarakenteessa löytyy seuraavat tiedot (Kuva 11).

STR_KL6581

Internal structure.

About this structure, the block [FB_KL6581\(\)](#) is connected to the read / send function blocks.

```

TYPE STR_KL6581 :
STRUCT
  by_Status   : BYTE;
  by_Node     : BYTE;
  by_ORG      : BYTE;
  ar_DB       : ARRAY[0..3] OF BYTE;
  _Dummy      : BYTE;
  dw_ID       : DWORD;
  ptData      : DWORD;
  iErrorId    : E\_KL6581\_Err;
  by_STATE    : BYTE;
  bError      : BOOL;
  idx         : USINT;
END_STRUCT
END_TYPE

```

Kuva 11. EnOcean-kortin Datarakenne logiikalle (STR_KL6581 2019)

7.1.3 DALI-laitteiden määrittely

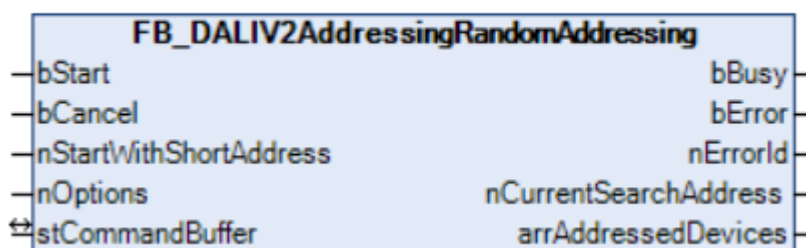
Korttien määrittelyiden jälkeen voidaan aloittaa määrittelemään DALI-laitteita (esimerkiksi valaisimia ja liiketunnistimia). Tässäkin on Beckhoffin DALI-kirjasto jälleen avuksi. Siinä on esitelty kaikki tarvittavat toimilohkot, joilla nämä saa tehtyä. Aina ensimmäiseksi on hyvä lukea, mitä laitteita väylästä löytyy ja mitä tietoja niistä saa. Tässä FB_DALIV2GetSettings-toimilohko on tarpeen. Tämä ei kuitenkaan ole pakollista. Valaisimien alustukseen ja lyhytosoitteiden muuttamiseen tarvitaan toimilohkot FB_DALIV2AddressingRandomAddressing ja FB_DALIV2SwapShortAddress. Seuraavassa on tarkempi esittely toimilohkoista.

FB_DALIV2AddressingRandomAddressing

FB_DALIV2AddressingRandomAddressing-toimilohko (Kuva 12) arpoo väylässä olevat valot uusille lyhytosoitteille sattuman varaiseen järjestykseen. Käyttäjällä itsellään ei ole mahdollisuutta vaikuttaa syntyvään järjestykseen. Toimilohkon Input-tietoihin annetaan lohkonkäynnistys-bitti (bStart), jolla toiminto aloitetaan, sekä nStart-WithShortAddress-tieto, jolla määritellään ensimmäisen valon järjestysnumero.

nOptions-valinnan (Kuva 13) mukaan käyttäjä pystyy valitsemaan, miten haluaa alustaa väylän valaisinlaitteet. Tässäkin toimilohkossa stCommandBufferilla valitaan väylä, jolta valot alustetaan. Toimilohkon Outputeista saadaan tietoa kuten, onko lohko käynnissä (bBusy), tai onko lohkon toiminnassa virheitä (bError) ja jos on, niin millaisia (nErrorId). Toimilohko myös kertoo, missä osoitteessa se etenee käynnissä ollessaan (nCurrentSearchAddress). Samoin lohko ilmoittaa ryhmään tekemistään muutoksista (arrAddressedDevices). (FB_DALIV2AddressingRandomAddressing 2019.)

FB_DALIV2AddressingRandomAddressing



Kuva 12. Esimerkki Beckhoffin DALI-valaisimien alustamiseen vaaditusta toimilohkosta (FB_DALIV2AddressingRandomAddressing 2019)

nOptions: Options for addressing the ballasts (see table). The individual constants must be linked with OR operators.

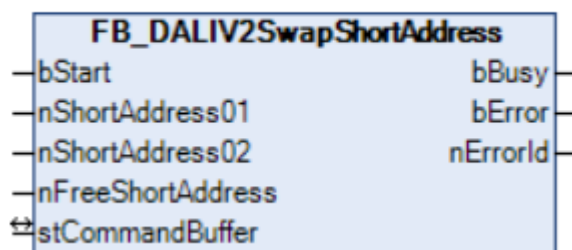
Constant	Description
DALIV2_OPTION_COMPLETE_NEW_INSTALLATION	All ballasts are re-addressed, including ballasts that already have a short address.
DALIV2_OPTION_DELETE_ALL_GROUP_ASSIGNMENTS	Prior to addressing, the group associations are deleted for any ballasts, even those which may not be addressed by the addressing method (see variables GROUP 0-7 and GROUP 8-15).
DALIV2_OPTION_DELETE_ALL_SCENE_ASSIGNMENTS	Prior to addressing, the scenes are deleted for any ballasts, even those which may not be addressed by the addressing method (see variables SCENE 0 to SCENE 15).
DALIV2_OPTION_OPTICAL_FEEDBACK	Prior to addressing, all ballasts are set to MIN LEVEL . Newly addressed ballasts are assigned MAX LEVEL brightness after allocation of the short address.
DALIV2_OPTION_WITHOUT_RANDOMISE	The RANDOMISE command is not called before the addressing sequence. This means that all ballasts retain their existing random address (RANDOM ADDRESS). Only use this option when necessary.

Kuva 13. nOptions-valinnan mahdolliset vaihtoehdot (FB_DALIV2AddressingRandomAddressing 2019)

FB_DALIV2SwapShortAddress

FB_DALIV2SwapShortAddress-toimilohkon (Kuva 14) avulla käyttäjä pystyy muuttamaan alustettujen valojen lyhytosoitteet haluamaansa järjestykseen. Toimilohkon Input-tietoihin annetaan käynnistystieto (bStart), sekä numerot osoitteille (nShortAddress01 ja -02), joiden paikkoja halutaan vaihtaa keskenään. Jälleen pitää olla tarkkana stCommandBufferin suhteen, että tekee muutoksia oikeaan väylään. Tämäkin toimilohko antaa tietoa siitä, onko se käynnissä (bBusy) ja onko virheitä havaittu (bError), sekä mitä virheitä (nErrorId), jos niitä on havaittu. (FB_DALIV2SwapShortAddress 2019.)

FB_DALIV2SwapShortAddress



Kuva 14. Esimerkki toimilohkosta FB_DALIV2SwapShortAddress (FB_DALIV2SwapShortAddress 2019)

7.1.4 DALI-ryhmien määrittely

DALI-ryhmien määrittelyyn tehtiin oma aliohjelma, jota kutsutaan pääohjelmassa. Näin pääohjelma (MAIN) pysyy siistimpänä, eikä käyttäjä tee niin helposti muutoksia valaistusryhmiin vahingossa. Tämä aliohjelma koostuu periaatteessa DALI-kirjastosta ja -ohjeesta löytyvistä valmiista toimilohkoista, FB_DALIV2GetSettings ja FB_DALIV2SetSettings. Ensin asetetaan näihin toimilohkoihin oikea väylä (stCommandBuffer), johon halutaan kohdistaa määrittelyt. Sitten toimilohkoihin määritellään muuttujat, joilla lohkoja pystyy käyttämään. Input nOptions -kohtaan kannattaa valita DALIV2_OPTION_GROUPS, jolloin toinen toimilohko kerää ja toinen määrittelee tiedot, mihin ryhmään kyseinen valo kuuluu. Ryhmittäisiä tehdessä täytyy huomioida, että ryhmät ilmoitetaan binäärimallin avulla. Eli ryhmään 1 liittäminen ilmoitetaan luvulla 1 ja ryhmään 2 liittäminen ilmoitetaan luvulla 2. Jos halutaan valon kuuluvan

molempiin ryhmiin 1 ja 2, liittäminen ilmoitetaan luvulla 3. Mikäli valo halutaan ryhmään 3, 5 ja 6, liitetään se luvulla 52 (Kuva 15.).

		52					
HEX	34						
DEC	52						
OCT	64						
BIN	0011 0100						
	GROUP	QWORD	MS	MT			
0 0 0 0	60	0 0 0 0	56	0 0 0 0	52	0 0 0 0	48
0 0 0 0	44	0 0 0 0	40	0 0 0 0	36	0 0 0 0	32
0 0 0 0	28	0 0 0 0	24	0 0 0 0	20	0 0 0 0	16
0 0 0 0	12	0 0 0 0	8	0 0 1 1	4	0 1 0 0	0

Kuva 15. Laskimella on helppo tarkistaa ilmoitettava luku, kun valitsee ryhmät, joihin haluaa valon liitettävän. Esimerkissä valittu ryhmät 3, 5 ja 6.

For-lausetta käyttäessä ei tarvitse kirjoittaa jokaista valoa erikseen haluttuun ryhmään, vaan saadaan näin helposti tietyt valaisimet tiettyyn ryhmään. Alla oleva kopio koodista selventää ryhmien luomista. Ylempi FOR-lauseke poistaa kaikista valaisimista ryhmät. Alempi FOR-lauseke asettaa valot 12-15 ryhmiin 3, 5 ja 6. Kun tämä ohjelma on tehty, on Dali3 väylän valot 12-15 ryhmissä 3, 5 ja 6. Tämä tarkoittaa, että nämä valot vastaavat, jos jotakin ryhmää näistä kutsutaan tai ohjataan.

```
DaliGetSettings (bStart:=bStartGetSettings,
                bBusy=>bGetSettingBusy,
                nOptions:=DALIV2_OPTION_GROUPS,
                eCommandPriority:=eSetPriority,
                bError=>bGetSettingsError,
                nErrorID=>nGetSettingsErrorId,
                stCommandBuffer:=InternalCommunication.Dali3_stComBuf,
                arrDALIDeviceSettings:=DaliSettingsArray);
IF (bGetSettingsBusy=TRUE) THEN
    bstartGetSettings:= FALSE;
END_IF
```



```

IF (bStartSetSettings=TRUE) THEN
    FOR index:= 0 TO 63 DO
        DaliSettingsArray [index].bPresent:=FALSE;
        DaliSettingsArray [index].nGroups:=0;
    END_FOR
    FOR index:= 12 TO 15 DO
        DaliSettingsArray[index].bPresent:= TRUE;
        DaliSettingsArray[index].nGroups:=52 ;
    END_FOR
END_IF
DaliSetSettings( bStart:=bStartSetSettings,
                bBusy=>bSetSettingsBusy,
                nOptions:=DALIV2_OPTION_GROUPS,
                eCommandPriority:=eSetPriority,
                bError=>bSetSettingsError,
                nErrorId=>bSetSettingsErrorId,
                stCommandbuffer:=internalCommunication.Dali3_stComBuf,
                arrDALIDeviceSettings:=DaliSettingsArray);
IF (bSetSettingsBusy=TRUE) THEN
    bstartSetSettings:= FALSE;
END_IF

```

7.1.5 Kytöinten määrittely

Kun ohjelmaan on määritelty valaisimet ja niiden ryhmät, on aika määrittellä niitä ohjaavia toimilaitteita. Ensimmäiseksi määriteltiin kytkimet, joilla voi ohjata valaisimia. Edellä on mainittu kytkinten olevan Eltakon TF4FT mallin kytkimiä, joiden toiminta perustuu radiotaajuiseen tekniikkaan. Avuksi pitääkin ottaa Beckhoffin Enocean-kirjasto ja -ohjeet, joista löytyy kaikki tarvittavat toimilohkot, jotta saa määriteltyä EnOcean-toimilaitteet toimintaan. Kytkimet lähettävät kaiken tiedon aikaisemmin esiteltyyn KL6583:n kautta.

Seuraavaksi ohjelmaan määritellään toiminnat kytkimien painonapeille. On järkevää tehdä kytkimen toiminnallisuudesta function block (FB), jolloin samaa toiminnallisuutta voi kutsua ohjelmassa moneen eri kohtaan. Myös FB:n sisäisen muistin vuoksi se on kannattavampi valinta kuin pelkkä funktio (FC). FB:n sisällä kannattaa määrittellä muuttujalle oma bitti eri painonapeille (esimerkiksi buttons.1), mikä helpottaa

viittauksia vastaisuudessa. Koska kaikki tieto kulkee KL6583:n kautta, saadaan tarvittavat tiedot myös sisäisestä str_6581 -rakenteesta eli rakenteesta, joka on esitelty kuvassa 11. Täältä luettu tieto kannattaa sijoittaa uuteen muuttujaan, jotta tietoa on helpompi seuraavaksi verrata siihen. Kytkintä määrittäessä luetaan ja tulkitaan siis tavujen arvoja. Jokaisella kytkimen painikkeella 1-4 (Kuva 16) on oma arvonsa, jonka se lähettää eteenpäin. Seuraavaksi täytyy katsoa kaikki esimerkiksi kytkimen painikkeen 1 antamat arvot, kun painike 1 on vaikuttuneena. Kyseisessä kytkimessä buttons-muuttujalla on kolme eri arvoa, joilla ilmaistaan, että painike 1 on painettuna (Kuva 17). Tämä tarkoittaa sitä, että kytkin lähettää eri arvon aina eri variaatiosta, jolla painike on vaikuttuneena, esimerkiksi painikkeiden arvot ovat vaikka 1-4. Kun painiketta yksi vaikutetaan, lähettää kytkin arvon 1, ja kun painiketta kaksi vaikutetaan, lähettää kytkin arvon 2. Mikäli taas painikkeet yksi ja kaksi molemmat ovat vaikuttuneina, kytkin lähettää, vaikka arvon 5, koska arvo kolme on varattu painikkeelle 3. Tämän vuoksi täytyy selvittää kaikilla kolmella tapaa kytkimen lähettämät arvot tilanteissa, joissa kyseinen painike on vaikuttuneena.



Kuva 16. Esimerkki käytetystä kytkimestä ja selitys mitä tarkoitetaan painikkeilla 1-4 (Painike Tap-radio 2019)

```

1
2 IF (InternalCommunication.str_KL6581.dw_ID = EnOceanID) THEN
3     OutputArray:=InternalCommunication.str_KL6581.ar_DB;
4 END_IF
5
6 buttons := 0;
7
8 IF (OutputArray[3] = 48 OR OutputArray[3] = 55 OR OutputArray[3] = 53) THEN
9     buttons.1 := TRUE;
10 ELSE
11     buttons.1 := FALSE;
12 END_IF
13
14 IF (OutputArray[3] = 112 OR OutputArray[3] = 55 OR OutputArray[3] = 23) THEN
15     buttons.2 := TRUE;
16 ELSE
17     buttons.2 := FALSE;
18 END_IF
19
20 IF (OutputArray[3] = 16 OR OutputArray[3] = 21 OR OutputArray[3] = 23) THEN
21     buttons.3 := TRUE;
22 ELSE
23     buttons.3 := FALSE;
24 END_IF
25
26 IF (OutputArray[3] = 80 OR OutputArray[3] = 21 OR OutputArray[3] = 53) THEN
27     buttons.4 := TRUE;
28 ELSE
29     buttons.4 := FALSE;
30 END_IF
31

```

Kuva 17. Esimerkki tehdystä ohjelmasta, jossa määritellään kytkimen näppäimet

7.1.6 Liiketunnistimien määrittely

Kytken määrittelyn jälkeen on vuorossa liiketunnistimien määrittely. Tässä työssä käytetty liiketunnistin on myös samalla radiotaajuisella tekniikalla toimivia kuin käytetyt kytkimet. Näistäkin kaikki tieto kulkee KL6583:n kautta. Jälleen apuna käytetään Beckhoffin EnOcean-kirjastoa ja sen käsikirjaa. Myös liiketunnistimen toiminnallisuudesta kannattaa tehdä FB, sillä näitäkin tarvitaan kokonaisuudessa useampia kuin yhtä.

Toimilohkon FB_Rec_Generic avulla saadaan kaikki vastaanotettu EnOcean-tieto luettua ja tämä on tärkeä toimilohko, jonka avulla pystytään käyttämään liiketunnistimilta tullutta tietoa. Tämän toimilohkon avulla käyttäjän tosin pitää tietää tarkalleen toimilaitteen lähettämä tieto, jotta pystyy tekemään toiminnallisuuden oikein. Tässä

työssä tehdyssä toimilohkossa luetaan tieto sisäisestä tiedonsiirrosta ja sijoitetaan yksilölliseen EnOceanID-toimilohkoon, sekä nimetään tarvittavat muuttujat. Näiden jälkeen määritellään luettu tieto paremmin nimettyihin muuttujiin, jolloin viittaaminen näihin myöhemmin helpottuu. Liiketieto saadaan siis toimilaitteen lähettämästä ensimmäisen tavun toisesta bitistä negaationa. Esimerkkikuva auttaa ymmärtämään liiketunnistimen määrittelyä (Kuva 18.)

```

Eltako_FBH65S* -# X
1  FUNCTION BLOCK Eltako_FBH65S
2  VAR_INPUT
3      EnOceanID: DWORD;
4  END_VAR
5  VAR_OUTPUT
6      newData:BOOL;
7      brightness:WORD;
8      motion:BOOL;
9  END_VAR
10 VAR
11     EnOceanData : FB_Rec_Generic;
12     telegram: ARRAY [0..3] OF BYTE;
13     byNode: BYTE;
14     byState: BYTE;
15    OldData: BOOL;
16     EnOceanType: E_EnOcean_Org;
17     Byte0:BYTE;
18     Byte1:BYTE;
19     Byte2:BYTE;
20     Byte3:BYTE;
21 END_VAR
22
1
2  EnOceanData (str_KL6581:=internalcommunication.str_KL6581, byNode:=0, dw_ID:=EnOceanID,
3      ar_Value=>telegram, by_Node=>byNode, by_STATE=>byState,
4      bReceive=>OldData, EnOceanTyp=>EnOceanType);
5
6  newData := NOT oldData;
7
8  IF (OldData=FALSE) THEN
9      Byte0 := telegram[0];
10     Byte1 := telegram[1];
11     Byte2 := telegram[2];
12     Byte3 := telegram[3];
13
14     motion := NOT (Byte0.1);
15     brightness :=Byte2 * 8;
16 END_IF

```

Kuva 18. Esimerkki liiketunnistimen toimilohkosta

7.2 Järjestelmäajan hakeminen

Valaisimien ohjauksessa tarvitaan myös järjestelmäaikaa, jonka avulla viikkoajastin toimii. Tähän tarvittiin Beckhoffin FB_LocalSystemTime-toimilohkoa. Tämä ei kuitenkaan ole täysin sama kuin Windowsin aika. Se pitääkin synkronoida tietyin väliajoin. Beckhoffin sivuilla on hyvät ohjeet esimerkkeineen, kuinka toimilohkon saa toimimaan. Tässä työssä tehtiin hyvin pitkälti samanlainen kutsu, joka oli esitelty Beckhoffin sivuilla. Erona on ainoastaan tähän projektiin paremmin sopiva harvemmin (10s välein) TwinCat System Logviewiin tulostuva viesti. Tällä ei ole toiminnallisuuteen mitään vaikutusta, vaan se on vain käyttäjälle lisätietona. Synkronointi pidettiin samassa 1 sekunnissa, jolloin periaatteessa Windowsin ja järjestelmän aika on sama. (Kuva 19).

```

SystemTimeSync
1 PROGRAM SystemTimeSync
2
3 VAR
4   fb_Time : FB_LocalSystemTime:= ( bEnable := TRUE, dwCycle := 1 );
5
6   LogTimer : TON := ( IN := TRUE, PT := T#10S );
7
8 END_VAR
9
10 // In the example the FB_LocalSystemTime function block is activated on program startup (rising edge at bEnable input).
11 // Once the time has been synchronised (bValid = TRUE), the PLC writes a message to the TwinCAT System Logview every 500 ms.-->10s)
12 // The internal synchronisation is carried out every second.
13
14 fb_Time();
15
16 LogTimer(IN:= fb_Time.bValid);
17
18 IF logTimer.Q THEN
19   logTimer( IN := FALSE );
20   logTimer( IN := fb_Time.bValid );
21   ADSLOGSTR( ADSLOG_MSGTYPE_HINT OR ADSLOG_MSGTYPE_LOG, 'Local System Time:%s', SYSTEMTIME_TO_STRING(fb_Time.systemTime));
22 END_IF

```

Kuva19. Projektin järjestelmäajan kutsu

7.3 Toimilohkojen tekeminen

Projektissa tehtiin kaksi päätoimilohkoa, joilla ohjataan valaisimia. Ensimmäisellä toimilohkolla ohjataan ajastuksen piirissä olevia valaistusryhmiä koko laboratorion alakerrassa poisluettuna konenäöalue. Toisella toimilohkolla ohjataan vain konenäköalueen valaisimia, mikä on täysin itsenäinen alueensa valonohjauksen kannalta. Konenäköalueelle haluttiin jokainen valaisin erikseen säädettäväksi, koska alueella käytetään paljon kameroita. Siksi on hyvä päästä muuttamaan valaistusta portaattomasti. Konenäköalueen valaisimien ohjaus on vain väliaikainen. Konenäkövalaisimia pystyy

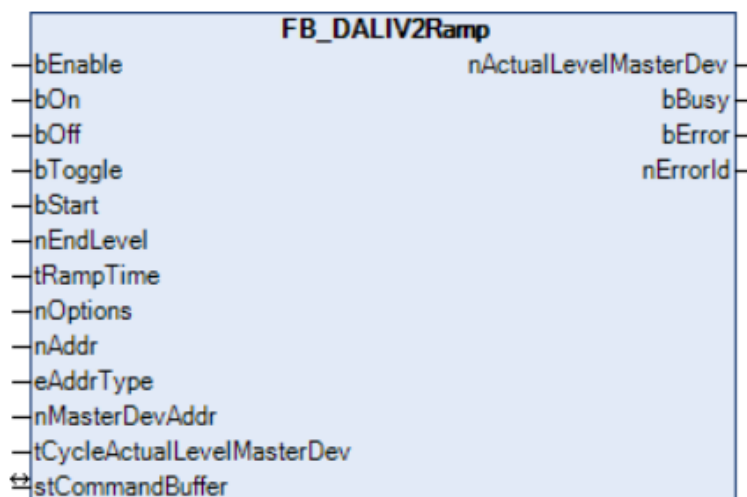
nyt väliaikaisesti kuitenkin ohjaamaan täysin, kunnes saadaan halutut HMI-tarvikkeet. Väliaikainen ohjaus tapahtuu TwinCat:in visualisointipuolelle tehdystä HMI:stä. Ohjauspaneeliin on tehty konenäköalueen jokaiselle valaisimelle oma ohjauksensa ja sieltä näkee myös, mikä on muistipaikan arvo ja millä arvolla mikäkin valaisin valaisee.

Ensimmäiseksi esittelyssä on Beckhoffin DALI-kirjastosta ja Buiding Automation-kirjastosta tärkeimmät toimilohkot, joiden avulla saadaan tehdyissä toimilohkoissa ohjattua valoja halutulla tavalla.

FB_DALIV2Ramp

FB_DALIV2Ramp (Kuva 20.) on toimilohko, joka löytyy Beckhoffin DALI-kirjastosta ja johon kaikki valonohjaus kiteytyy tehdyssä ohjelmassa. Tällä ohjataan valaisimia päälle ja pois, sekä voidaan himmentää niitä. Tällä toimilohkolla saadaan valaisimet portaattomasti siirtymään haluttuun valaistusarvoon halutussa ajassa (tRampTime). Myös sammuttaminen ja sytyttäminen onnistuu hallitusti portaattomasti. Mikäli halutaan sammuttaa tai sytyttää valaisimet portaattomasti, pitää käyttää nEndLevel-arvoja. Käyttämällä bOn- tai bOff-tuloja, syttyy ja sammuu valot ilman liukumaa. nEndLevel-arvoa muuttamalla ja bStart-bitillä saadaan lohko muuttamaan valojen kirkkautta. nEndLevel-arvo voi olla mikä tahansa 0 ja 255 välillä. 0 on sammuksissa ja 255 on kirkkain valaistuksen teho. eAddrType-kohtaan valitaan, mihin lohko vaikuttaa; väylän yksittäisiin valoihin, ryhmiin vai broadcast-tyyppisesti kaikkiin väylän valaisimiin. nAddr-kohtaan valitaan edellä mainitun valinnan suhteen joko valon yksittäinen osoite, tai ryhmän osoite. Lohkoon pitää myös ilmoittaa, mikä on ryhmän master-laite (nMasterDev). Lohko myös ilmoittaa, mikä on master-laitteen valaistusarvo (nActualLevelMasterDev). stCommandBuffer kohtaan taas valitaan, mitä väylää ohjataan. (FB_DALIV2Ramp 2019.)

FB_DALIV2Ramp



Kuva 20. FB_DALIV2Ramp esimerkkikuva lohkoista (FB_DALIV2Ramp 2019)

FB_WeeklyTimeSwitch

FB_WeeklyTimeSwitch (Kuva 21.) on todella kätevä viikkoajastin, jolla voidaan asettaa tulo päälle (bOutput) haluttuina viikonpäivinä. Kerrotaan vain aloitus aika (tSwitchOnTime) ja lopetusaika (tSwitchOffTime). Tämä lohko kuitenkin tarvitsee todellisen kellonajan. Tehdyssä ohjelmassa tätä varten haettiin edellä mainitun järjestelmääjan. Lohkoon voidaan kuitenkin laittaa vain yksi aikaväli, mutta lohkoja voidaan käyttää useampaa samaan aikaan, jolloin voidaan ottaa käyttöön useampiakin aikavälejä. Lohkosta saa myös tiedon nousevasta reunasta toimilohkon Outputin mennessä päälle (bEdgeOn) ja pois päältä (bEdgeOff). Nämä nousevasta ja laskevasta reunasta ilmoittavat signaalit ovat päällä vain yhden syklin, joten resetointia ei tarvita erikseen. (FB_WeeklyTimeSwitch 2019.)

Edellä mainitut toimilohkot ovat tärkeimmät toimilohkot, joita kutsutaan tehdyssä toimilohkossa. Niiden avulla saadaan ajastus ensimmäiseen toimilohkoon ja valojen hallinta molempiin toimilohkoihin

FB_WeeklyTimeSwitch



Kuva 21. Esimerkki Beckhoffin Building Automation Basic-kirjastosta, viikkoajastin toimilohkosta (FB_WeeklyTimeSwitch 2019)

Ensimmäinen toimilohko

Toimilohkon teon ensimmäinen vaihe oli järjestelmäajan kutsuminen. Tämä siksi, että saadaan halutut viikkoajastimet toimimaan oikealla ajalla. Seuraavaksi kutsutaan liiketunnistin-toimilohkoa, jotta voidaan ohjata valoja myös tunnistimella. Sitten esitellään tarvittavat nousevien reunojen tunnistimet eli triggerit. Ne ovat nousevat R_Triggerit (Rising-Trigger) ja laskevat F_Triggerit (Falling-Trigger). R_Triggereitä tarvitaan liiketunnistimessa ja kytkimissä. F_triggereitä tarvitaan taas ajastimien reunojen tunnistamiseen sekä pääkytkimen käyttöön. Sitten vuorossa ovat halutut ajastimet kytkimiin, liiketunnistimiin ja pääkytkimiin. Sitten vasta alkaa ohjelmointi, jonka mukaan halutaan ohjata valoja. Viimeisenä tulee viikkoajastin- ja valonohjaus-toimilohkot. Liitteessä 2 on tarkemmin esitelty ohjelmoitu koodi toimilohkon sisältä.

Toinen toimilohko

Tämä toinen toimilohko on konenäköalueen valoille. Tavoitteena on siis jokaisen valon yksilöllinen ohjaus portaattomasti. Tämä oli helpompi toteuttaa ohjelmallisesti kuin ensimmäinen, sillä DALI-kirjaston toimilohkon lisäksi ei tarvita niin paljo ohjelmoimista lisää. Tarvitaan vain nousevien reunojen triggerit ja muistia varten muutama apumuuttuja, johon voi asettaa haluttu valaistusarvo muistiin. Muistin asetusta varten tarvittiin myös ajastin, jolla tunnistetaan painikkeen pohjassa pito. Uuden arvon

pystyy vaihtamaan pitämällä muistipainiketta 3 sekuntia pohjassa. Liitteessä 3 on tarkemmin ohjelmoitu koodi tämän toimilohkon osalta.

8 PROJEKTIN LOPPUTILANNE

8.1 Projektin analysointi

Projektin alussa oma kokemukseni ohjelmoinnista oli todella vähäistä. Osasin vain sen, mitä koulussa kursseilla ohjatuilla tunneilla on tehty. Alkuun meni aikaa perehtyä itselleni uuteen ohjelmointikieleen, sillä kursseilla ei oltu opetettu Structured Text (ST) -kielellä ohjelmoimista. Aikaa kului myös alussa DALI-väylien liittämiseen logiikkaan. Suurin osan ajasta kului kuitenkin Beckhoffin DALI-, Building Automation- ja EnOcean-kirjastojen lukemiseen. DALI-2 -valaistus Beckhoffin logiikalla ja EnOcean-toimilaitteilla ohjattuna olikin itselleni täysin uusi ja ihmeellinen asia. Internetistä tietoa etsimällä tulinkin päätelleeksi, että tämä kombinaatio on Suomessa vielä melko uusi asia, mutta maailmalla kovassa noususuhdanteessa. Valtaosa DALI-valaistuksista on DALI-1 -versioita ja toteutettu Helvarin ToolBoxin tai KNX:n avulla. Siksi ainoa apuni olikin Beckhoffin omat kirjastot ja käsikirjat. Se ei kuitenkaan ollut huono asia, sillä käsikirjat olivat todella selkeitä ja sisälsivät hyviä esimerkkejä. Projektin suurimpaan ongelmaan kuitenkin törmäsin kesän aikana, kun piti liiketunnistimia alkaa määrittelemään. Kuten jo edellä kerroin, käytössä olivat Helvarin Digidim tunnistimet ja Beckhoffin ja muiden toimittajien DALI-2 -laitteet. Ongelmana olikin, että Digidim-liiketunnistimet eivät olleetkaan DALI-2 -yhteensopivia vaan ovat määriteltävissä vain Helvarin omilla ohjelmointityökaluilla. Tästä johtuen projekti koki pienen takapakin, kun etsin syytä miksi väylässämme olevat valot eivät toimineet kuten testivalaistuslaitteistoni. Varmuuden tähän antoi Beckhoffin nopea asiakaspalvelu, kun pyysin apua ongelmaani.

Koska liiketunnistimia ei päässyt määrittelemään, ajoi niiden antamat käskyt kaiken itse kirjoittamani ohjelman yli. Tämän tiedon saatuani täytyi käydä poistamassa jo asennetut toimilaitteet jokaisesta väylästä erikseen. Haastavaa tästä teki se, että väylissä oli käytetty tähti- ja sarjakytkentää liiketunnistimille ja valoille. Omalle kohdalle tuli hyvinkin selväksi, miksi on tärkeää merkitä johtimet hyvin ja varsinkin, miksi on tärkeää tehdä kytkennät, kuten piirustuksissa on merkitty.

Ohjelmoidessani huomasin huomattavaa edistystä. Ajastuksen piirissä olevaa valojen ohjaustoimilohkoa tehdessäni jo huomasin pieniä parannuksia ja näin ollen teinkin useamman eri version toimilohkosta. Toiminnallisuus kuitenkin pysyi samana eri versioiden välillä. Pari ensimmäistä versiota olivat todella pitkiä ohjelmallisesti ja olivat myös kovin herkkiä virhetiloille. Tämä viimeisin läpäisikin oman testauksen testiympäristössä puhtain paperein ja siirsin ohjelman oikeaan ympäristöön luottavaisin mielin. Tässä kohtaa oli jo otettu liiketunnistimet pois ja edeltävätkin versiot toimivat melko luotettavasti oikeassa ympäristössä.

Kuitenkin liiketunnistinongelma toi mukanaan toisen ongelman. Minulla ei ollut kuin yksi liiketunnistin EnOceanilta testiympäristössäni, ja vain yksi ehjä EnOceanin-ohjain asennettuna valmiiksi parven portaiden ylle. Ongelmana tässä oli se, että kuten edellä olen kertonut, EnOceanin liiketunnistin FBH65S on langaton liiketunnistin, jossa on pieni sisäinen akku. Oletuksena oli, että se saa virtaa tarpeeksi aurinkokennostaan. Tämä kuitenkin oli väärää tietoa, sillä aurinkokenno toimisi ulkona hyvin, mutta ei sisätiloissa, ei ainakaan Automaatiolaboratoriossa. Valaistus, mikä laboratoriossa on, ei riitä lataamaan liiketunnistimen sisäistä akkua. Tästä syystä emme päässeet testaamaan tekemäämme toimilohkoa täydellisesti useamman liiketunnistimen kanssa. Enkä tiedä toimisiko tekemämme ohjelma useamman tunnistimen kanssa. Tästä syystä onkin jäänyt epä tietoisuus ohjelmoinnin täydellisestä toimivuudesta.

8.2 Parannukset / Kehitysehdotukset

Vastaan tuli monia pieniä epäkohtia, jotka toisin tehtynä olisivat helpottanut ja nopeuttanut valaistuksen käyttöönottoa paljon. Esimerkiksi kaapelointien merkitseminen jokaisella jakorasiolla ja keskuksella olisi todella tärkeää. Samoin piirustusten oikeellisuus ja ajantasaisuus on tärkeää. Toimilaitteita tilatessa tulee varmistaa niiden sopivuus. Käyttöönottoa tehdessä taas täytyy varmistaa laitteiden määrä, että laitteita olisi tarpeeksi. Nämä ovat pieniä, ehkä vähäpätöisiltä vaikuttavia asioita, mutta lopputulokseen melko paljon vaikuttavia. Koska tämäkin on ollut projekti, jossa työntekijät on vaihtuneet usein, olisi nämä vaikuttaneet paljon siihen, ettei olisi mennyt aikaa niin sanotusti turhiin asioiden selvityksiin. Mutta toisaalta, kokemuksena tämä on ollut ko-

konaisuudessaan verraton. Itse opin paljon pienien asioiden merkityksistä käyttöön-
otossa. Mieluummin kannattaa tehdä liikaa dokumentointia kuin liian vähän. Toisaalta
laitteiden määrää on vaikea arvioida, varsinkin jos jo asennetut laitteet osoittautuvat
vääränlaisiksi. Tämän vuoksi myös oma testailu valaistuksen ohjauksesta vaikeutui,
kuten myös vaikeutui ohjaamisen saaminen halutun laiseksi tai muuttui lähes mahdot-
tomaksi. Ja kesän aikana ei uusia osiakaan saanut niin nopeasti, että olisi voinut yrittää
korjata tapahtunutta virhettä.

LÄHTEET

BK1150 – EtherCAT ”Compact” Bus Coupler. 2017. Beckhoff Internet-sivusto. Viitattu 14.8.2019. https://www.beckhoff.com/english.asp?bus_terminal/bk1150.htm

Building automation. 2019. Wikipedia Internet-sivusto. Viitattu 23.9.2019. https://en.wikipedia.org/wiki/Building_automation#Lighting

Company Profile. 2019. EnOcean Internet-sivusto. Viitattu 13.8.2019. <https://www.enocean.com/en/about-us/company-profile/>

C10-S1. 2019. Glamox Internet-sivusto. Viitattu 15.8.2019 <https://glamox.com/fi/products/qun35ry/items/c10081631>

C6015 – Ultra-compact Industrial PC. 2019. Beckhoff Internet-sivusto. Viitattu 13.8.2019. <https://www.beckhoff.com/c6015/>

DALI AG. 2001. DALI AG (Digital Addressable Lighting Interface Activity Group). s. 10 ja 17-18. Viitattu 15.8.2019. https://web.archive.org/web/20130627012349/http://www.dali-ag.org/c/manual_gb.pdf

DALI Lighting Control Solutions Explained. 2012. Crestron. Dia 23/91. Viitattu 15.8.2019. <https://www.slideshare.net/crestronhq/dali-lighting-control-solutions-explained>

DALI-2 systems & product types. 2019. Digital Illumination Interface Alliance Internet-sivusto. Viitattu 14.9.2019. <https://www.digitalilluminationinterface.org/systems/>

Digidim 312 DALI Multisensori. 2013. Helvar. Viitattu 15.8.2019. http://www.helvar.com/media/pd/2017/20170301/312_DATASHEET_FI.pdf

Eltako is a promoter of the EnOcean Alliance. 2019. EnOcean-alliance Internet-sivu. Viitattu 14.8.2019. <https://www.enocean-alliance.org/eltako-promoter-enocean-alliance/>

EnOcean. 2019. Wikipedia Internet-sivusto. Viitattu 16.8.2019. <https://en.wikipedia.org/wiki/EnOcean#Alliance>

FB_DALIV2AddressingRandomAddressing. 2019. Beckhoff Internet-sivusto. Viitattu 24.9.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_dali/45035996416601739.html&id=

FB_DALIV2Ramp. 2019. Beckhoff Internet-sivusto. Viitattu 2.10.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_dali/36028797161883787.html&id=

FB_DALIV2SwapShortAddress.2019. Beckhoff Internet-sivusto. Viitattu 24.3.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_dali/45035996416601739.html&id=

FB_KL6581. 2019. Beckhoff Internet-sivusto. Viitattu 22.8.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclibdali/html/tcplclibdaliv2_kl6821config.htm&id=304921586865445113

FB_KL6821Config. 2019. Beckhoff Internet-sivusto. Viitattu 22.8.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclibdali/html/tcplclibdaliv2_kl6821config.htm&id=304921586865445113

FB_WeeklyTimeSwitch. 2019. Beckhoff Internet-sivusto. Viitattu 2.10.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclibbasic/html/TcPlcLibBABasic_WeeklyTimeSwitch.htm&id=

Glamox Oy. 2019. Glamox Internet-sivusto. Viitattu 15.8.2019 <https://glamox.com/fi/about-glamox-luxo>

Helvar yrityksenä. 2019. Helvar Internet-sivusto. Viitattu 15.8.2019. <http://www.helvar.com/fi/yritys/helvar-yrityksena/>

Innovaatioita perinteiden kautta. 2019. Eltako Internet-sivusto. Viitattu 14.8.2019. <https://www.eltako.com/fi/yritys/tietoa-meistae.html>

Introducing DALI. 2019. Digital Illumination Interface Alliance Internet-sivusto. Viitattu 13.8.2019. <https://www.digitalilluminationinterface.org/dali/>

KL1114 – 4-channel digital input terminal 24 V DC. 2016. Beckhoff Internet-sivusto. Viitattu 14.8.2019. https://www.beckhoff.com/english.asp?bus_terminal/kl1114.htm

KL6581 – EnOcean master terminal. 2019. Beckhoff Internet-sivusto. Viitattu 14.8.2019. https://www.beckhoff.com/english.asp?bus_terminal/kl6581.htm

KL6583 – EnOcean transmitter and receiver. 2019. Viitattu 14.8.2019. https://www.beckhoff.com/english.asp?bus_terminal/kl6583.htm

KL6821 – DALI / DALI 2 multi-master and power supply terminal. 2019. Beckhoff Internet-sivusto. Viitattu 14.8.2019. https://www.beckhoff.com/english.asp?bus_terminal/kl6821.htm

KL9010 – End terminal. 2019. Beckhoff Internet-sivusto. Viitattu 14.8.2019. <https://www.beckhoff.com/KL9010/>

Langaton Eltako-järjestelmä. 2017. Eltako. s16/76. Viitattu 14.8.2019. https://www.eltako.com/fileadmin/downloads/fi/prospekte/Langaton_Eltako-jaerjestelmae_fi.pdf

Lighting. 2019. Wikipedia Internet-sivusto. Viitattu 23.9.2019. https://en.wikipedia.org/wiki/Building_automation#Lighting

luxCONTROL DALI manual. 2013. Tridonic. s6/93. Viitattu 27.11.2019. https://www.tridonic.se/se/download/technical/DALI-manual_en.pdf

New automation technology. 2019. Beckhoff Suomi Internet-sivusto. Viitattu 13.8.2019. <https://www.beckhoff.fi/fi/default.htm?beckhoff/default.htm>

Painike Tap-radio. 2019. Sähkönumerot.fi Internet-sivusto. Viitattu 27.11.2019. <https://www.sahkonumerot.fi/2814224>

Philip Aminoff on tiiminrakennuksen mestari. 1.9.2016. Heikki Haapavaara, Kauppalehti Internet-sivusto. Viitattu 15.8.2019. <https://www.kauppalehti.fi/uutiset/philip-aminoff-on-tiiminrakennuksen-mestari/0f17dc72-ccd4-32af-8033-c7dd3f8c2126>

Power supply unit - UNO-PS/1AC/24DC/100W. 2019. Phoenix Contact Internet-sivusto. Viitattu 14.8.2019. <https://www.phoenixcontact.com/>

Rakennuksetkin saavat hermot ja aivot. 2019. Sähköteknisen kaupan liitto Internet-sivusto. Viitattu 23.9.2019. <https://www.stkliitto.fi/viestinta/artikkelit/137-rakennuksetkin-saavat-hermot-ja-aivot.html>

STR_KL6581. 2019. Beckhoff Internet-sivusto. Viitattu 22.8.2019. https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclibenocean/html/TcPlcLibEnOcean_STR_KL6581.htm&id=

Tap-radio, Innovatiivinen langaton taloautomaatiojärjestelmä Enocean teknologialla ja älykoti sovelluksilla. 2017. Eltako. s.19/40. Viitattu 14.8.2019. https://www.eltako.com/fileadmin/downloads/fi/_prospekte/Tap-radio_kuluttaja%20sarja_ammattilaissarja_FIN.pdf

Transport of DALI. 2018. Dignet control systems Internet-sivusto. Viitattu 15.8.2019. <https://www.dignet.net.au/an-introduction-to-dali-2017/>

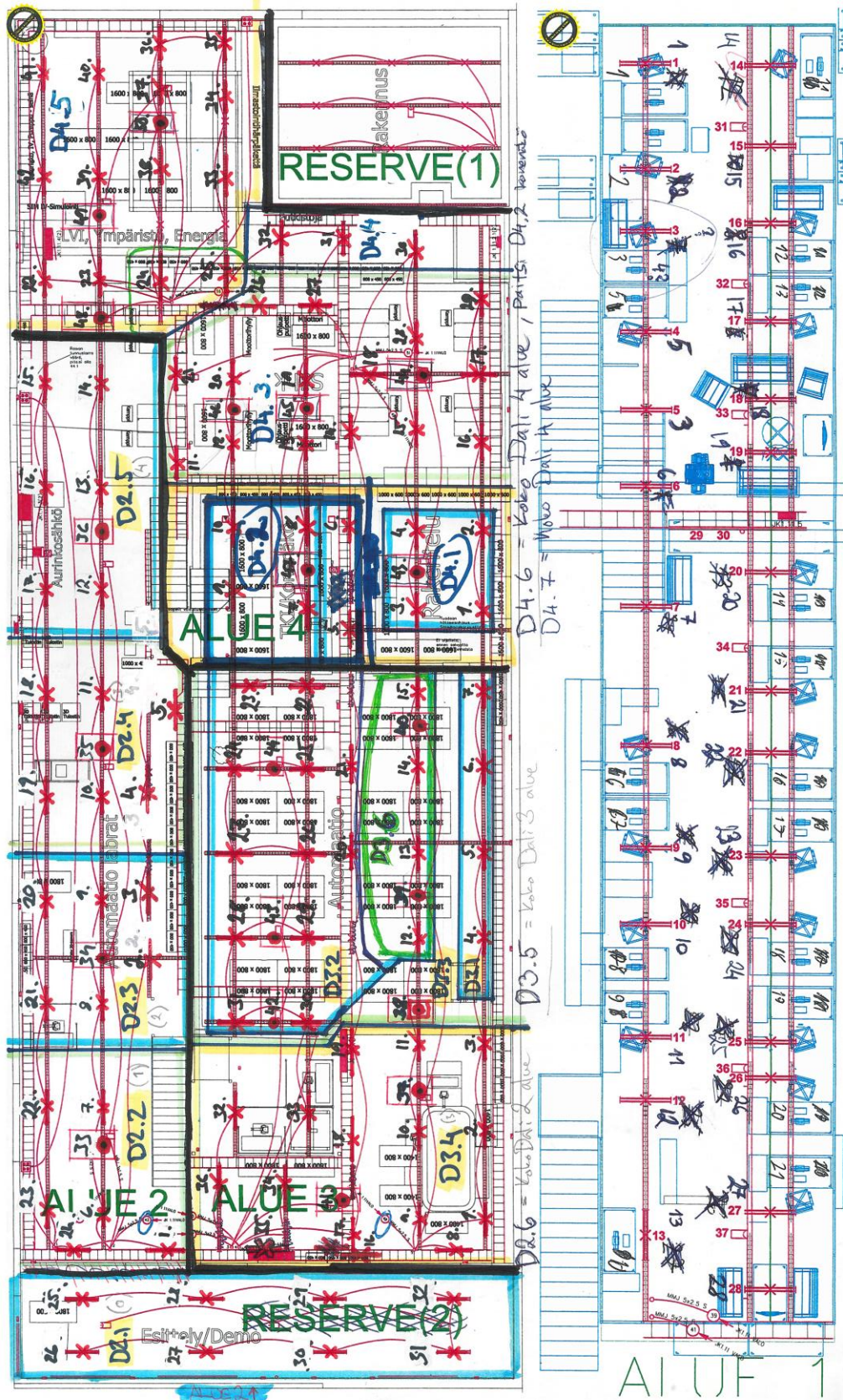
TwinCAT – PLC and Motion Control on the PC. 2019. Beckhoff Internet-sivusto. Viitattu 13.8.2019. <https://www.beckhoff.com/twincat/>

TYA670W - KNX/DALI Gateway värilämpötilan säädöllä. 27.11.2018. UTU Internet-sivusto. Viitattu 15.8.2019. <https://www.utu.eu/tya670w-knxdali-gateway-varilampotilan-saadolla>

Uninterruptible power supply - UNO-UPS/24DC/24DC/60W. 2019. Phoenix Contact Internet-sivusto. Viitattu 14.8.2019. <https://www.phoenixcontact.com/>

Upotettava PIR-liiketunnistin (311). 2016. Helvar. Viitattu 15.8.2019. https://www.helvar.com/media/pd/2017/20170303/311_DATASHEET_FL.pdf

Yrityksen historia. 2019. Phoenix Contact Internet-sivusto. Viitattu 14.8.2019. <https://www.phoenixcontact.com/>



D4.6 = Koko Dait 4 aive / Panssi Dtr. 2 keneräts
D4.7 = Moko Dait 4 aive

D3.5 = Koko Dait 3 aive
D2.6 = Koko Dait 2 aive

AIUE 1

AIUE 2

AIUE 3

AIUE 4

AIUE 5

RESERVE(1)

Valojenohjaukseen käytetty FB väylään 2

Muuttujien esittely:

```

FUNCTION_BLOCK FB_RampSensorTimer_Dali_2
VAR_INPUT
    ActualLevel:BYTE;
    Switch:BYTE; //EnOceanin Buttons
    MasterSwitch:BYTE; //Toinen kytkin (pää)
    MasterSwitchTimeDelay:TIME; //Masterkytkin viive
    EnOceanID_MotionSensor: DWORD; //Liiketunnistimen ID
    Delay_Switch:TIME; //Käyttökytkin viive
    Delay_MotionSensor:TIME; //Liiketunnistimen viive
    Motion_Detected:BOOL; //Havaittu liike
    OnTime:TIME_OF_DAY;
    OffTime:TIME_OF_DAY;
    Addr:BYTE;
    AddrType: E_DALIV2AddrType;
    MasterDevAddr:BYTE;
    RampTime:TIME;
    OutputTimer:BOOL;
END_VAR
VAR_OUTPUT
END_VAR
VAR
    tCurrentTime: DATE_AND_TIME;
    F_TRIG_Timer_Switch: F_TRIG; //Timer_Kytkimen laskeva
    Timer_Switch: TOF; //Timer kytkin
    Sensor: Eltako_FBH65S;
    Timer_MotionSensor: TOF; // Liiketunnistimen ajastin
    F_Trig_MotionSensor: F_TRIG; //liik.tun. laskeva
    R_TRIG_MotionDetected: R_TRIG; //liik.tun. nouseva
    F_Trig_Timer_MasterSwitch: F_TRIG; //pääkytk.laskeva
    Timer_MasterSwitch: TOF; //pääkytkimen ajastin
    WeekTimer: FB_WeeklyTimeSwitch;//viikkoajastin
        OnTimer: BOOL;
        OffTimer: BOOL;
    Lights: FB_DALIV2Ramp; // valojenohjaus
        Start: BOOL;
        EndLevel: BYTE;
        Busy: BOOL;
        Error: BOOL;
        ErrorID: UDINT;
    R_TRIG_1 : R_TRIG; //käyt.kytkimen 1 nouseva reuna
    R_TRIG_2 : R_TRIG;
    R_TRIG_3 : R_TRIG;
    R_TRIG_4 : R_TRIG;
    F_TRIG_Master_1 : F_TRIG; //Pääkytkimen 1 laskeva

```

```

F_TRIG_Master_2 : F_TRIG;
F_TRIG_Master_3 : F_TRIG;
F_TRIG_Master_4 : F_TRIG;
LightOn: BOOL;
LightOff: BOOL;
ExLevel: BYTE;
ThirdLevel: BYTE;
MasterForcedShutDown : TON;
Master_Help:Bool;
END_VAR

```

Toimilohkon koodi:

```

// CurrentTime of System
tCurrentTime:=
SystemTime_TO_DT(TIMESTR:=SystemTimeSync.fb_Time.systemTime;
// Liiketunnistimen liittäminen
Sensor(EnoceanID:=EnOceanID_MotionSensor,motion=>Motion_Detec-
ted); // Lisätään tunnistimien ID:t Inputteihin

// Ajastin Liiketunnistimelle
R_TRIG_MotionDetected(CLK:=Motion_Detected); //nouseva trig-
geri liikkeen havaitsemiosesta liiketunnistimessa
F_Trig_MotionSensor(CLK:=Timer_MotionSensor.Q); // viiveellä
ajastin kuinka monen sekunnin/ minuutin/ tunnin päästä valot sam-
mutuvat
Timer_MotionSensor(IN:=Motion_Detected,
PT:=Delay_MotionSensor); //IN: LiikeHavaittu toimii suoraan oh-
jaamaan TOF timeria koska liiketunnistimen nouseva reuna rese-
toi laskurin ja liiketunnistimen laskeva reuna starttaa lasku-
rin. Eli uusi liike nolaa laskurin

// Triggerit nousevat ja laskevat reunat
R_TRIG_1 (CLK:= Switch.1); //katkaisimen 1 nouseva reuna
R_TRIG_2 (CLK:= Switch.2);
R_TRIG_3 (CLK:= Switch.3);
R_TRIG_4 (CLK:= Switch.4);
F_TRIG_Master_1 (CLK:= MasterSwitch.1); //Pääkytkimen 1 laskeva
F_TRIG_Master_2 (CLK:= MasterSwitch.2);
F_TRIG_Master_3 (CLK:= MasterSwitch.3);
F_TRIG_Master_4 (CLK:= MasterSwitch.4);

```

```

// Ajastin Kytkimelle...
f_TRIG_Timer_Switch (CLK:= Timer_Switch.Q); //Timer_Kytkimen
laskevan reunan triggeri
Timer_Switch (IN:= R_TRIG_1.Q , PT:= Delay_Switch);// Ajastin,
X ajan päästä sammuu kun kytkimestä laitettu valot päälle

// Ajastin "pääkytkin"
F_Trig_Timer_MasterSwitch(CLK:=Timer_MasterSwitch.Q);
Timer_MasterSwitch(IN:= MasterSwitch.1 OR MasterSwitch.2, PT:=
MasterSwitchTimeDelay);

// ViikkoTimer sytyttää valot
IF EndLevel =0 AND OnTimer = TRUE THEN
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 230;
    Start:= TRUE;
END_IF

// Liiketunnistin sytyttää valot
IF endLevel =0 AND R_TRIG_MotionDetected.Q THEN
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 230;
    Start:= TRUE;
END_IF

// viikkoTimer sammuttaa valot
IF OffTimer=TRUE AND (Timer_MotionSensor.Q=FALSE AND
Timer_Switch.Q=FALSE AND Master_Help=FALSE) THEN
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 0;
    Start:= TRUE;
END_IF

```

```

//Liiketunnistin sammuttaa valot
IF F_Trig_MotionSensor.Q=TRUE AND WeekTimer.bOutput=FALSE AND
Timer_Switch.Q=FALSE AND Master_Help=FALSE THEN
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 0;
    Start:= TRUE;
END_IF
// Kytkimen
Ajastin Sammuttaa valot
IF (f_TRIG_Timer_Switch.Q=TRUE AND WeekTimer.bOutput=FALSE AND
Timer_MotionSensor.Q=FALSE AND Motion_Detected=FALSE AND Mas-
ter_Help=FALSE ) THEN
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 0;
    Start:= TRUE;
END_IF

// Kytkinten Ohjaus
IF EndLevel = 0 THEN
    Switch.2 := FALSE;
    Switch.4 := FALSE;
END_IF
IF R_TRIG_1.Q THEN // Katkaisin 1 sytyttää valot 200
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    EndLevel := 230;
    Start:= TRUE;
END_IF
IF R_TRIG_3.Q THEN // Sammuttaa valot
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    EndLevel := 0;
    Start:= TRUE;
END_IF

```

```

IF EndLevel<250 AND R_TRIG_2.Q AND Busy=FALSE THEN
// Nostaa kirkkautta 20, ...busy = false estää tilan siirtymisen
kun valaistuksen taso siirtymässä
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    EndLevel := EndLevel + 20;
    Start:= TRUE;
END_IF
IF EndLevel>190 AND R_TRIG_4.Q AND Busy=FALSE THEN
// himmentää 20, ...busy = false estää tilan siirtymisen kun
valaistuksen taso siirtymässä
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    EndLevel := EndLevel - 20;
    Start:= TRUE;
END_IF

// PääKytkin...
IF F_TRIG_Master_1.Q OR F_TRIG_Master_2.Q THEN
    Master_Help:=TRUE;
    ThirdLevel:=ExLevel;
    ExLevel:=EndLevel;
    EndLevel:= 250;
    Start:= TRUE;
END_IF
IF (F_TRIG_Master_3.Q OR F_TRIG_Master_4.Q OR F_Trig_Timer_Mas-
terSwitch.Q)AND WeekTimer.bOutput=FALSE AND Timer_MotionSen-
sor.Q=FALSE AND Motion_Detected=FALSE AND Timer_Switch.Q=FALSE
AND Master_Help=TRUE THEN
    Master_Help:=FALSE;
    //Timer_MasterSwitch.:=FALSE;
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    EndLevel:=0;
    Start := TRUE;

```

```

ELSIF
(F_TRIG_Master_3.Q OR F_TRIG_Master_4.Q OR F_Trig_Timer_Mas-
terSwitch.Q)AND(WeekTimer.bOutput=TRUE OR Timer_MotionSen-
sor.Q=TRUE OR Motion_Detected=TRUE OR Timer_Switch.Q=TRUE)AND
Master_Help=TRUE THEN
    Master_Help:=FALSE;
    ThirdLevel:=ExLevel;
    ExLevel := EndLevel;
    IF ThirdLevel <> 0 THEN
        EndLevel:=ThirdLevel;
        Start := TRUE;
    ELSIF
        ThirdLevel = 0 THEN
            EndLevel:= ExLevel;
            Start:=TRUE;
    END_IF
END_IF

MasterForcedShutDown (IN:=(MasterSwitch.3 OR MasterSwitch.4),
PT:=T#2S);
    IF MasterForcedShutDown.Q THEN
        ThirdLevel:=ExLevel;
        ExLevel := EndLevel;
        EndLevel:=0;
        Start := TRUE;
    END_IF

// ViikkoTimeri
WeekTimer (bEnable:=TRUE,tCurrentDateTime:=tCurrentTime,
tSwitchOnTime:=OnTime,tSwitchOffTime:=OffTime,
bSunday:=FALSE,bMonday:=TRUE,bTuesday:=TRUE,
bWednesday:=TRUE,bThursday:=TRUE,bFriday:=TRUE,
bSaturday:=TRUE,bOutput=>OutputTimer,
bEdgeOn=>OnTimer,
bEdgeOff=>OffTimer);

```

```
// Valojen Ohjaus
Lights (bOn:=LightOn,bOff:=LightOff,bStart:=Start,
        nEndLevel:=EndLevel,
        nActualLevelMasterDev=>ActualLevel,
        bBusy=>Busy,bError=>Error,nErrorId=>ErrorID ,
        nAddr:=Addr,eAddrType:=AddrType,
        nMasterDevAddr:=MasterDevAddr,tRampTime:=RampTime,
        stCommandBuffer:=
        InternalCommunication.Dali2_stComBuf);

Start := FALSE;
```

Valojenohjaukseen käytetty FB väylään 4 (Konenäöalue)

Muuttujien esittely:

```

FUNCTION_BLOCK FB_Kone_Visu_Dali_4
VAR_INPUT
    Addr: BYTE;
    AddrType: E_DALIV2AddrType;
    MasterDevAddr: BYTE;
END_VAR
VAR_OUTPUT
END_VAR
VAR
    Lights:FB_DALIV2Ramp;           //FB_RAMP...
        EndLevel: BYTE;
        Start: BOOL;
        Busy: BOOL;
        Error: BOOL;
        ErrorID: UDINT;
        LightOn: BOOL;
        LightOff: BOOL;
        ActualLevel: BYTE;
    VisuLevel: BYTE;           //SÄÄDIN VISUSSA
    VisuOn:BOOL;           //Visun ON-näppäin
    VisuOff:BOOL;           //Visun OFF-näppäin
    VisuPlus:BOOL;           //Visun kirkkaus
    VisuMiinus:BOOL;           //Visun himmennys
    VisuMemory:BOOL;           //Visun Muistinäppäin
    VisuMemoryLevel:BYTE; //Muistissaoleva arvo
    MemoryTimer : TON; //ajastin muistille

    R_TRIG_ON:R_TRIG; // nouseva reuna On näpp.
    R_TRIG_OFF:R_TRIG; // NOUSEVA OFF-NÄPP.
    R_TRIG_Plus:R_TRIG; // kirkkauden nouseva
    R_TRIG_miinus:R_TRIG; // himmennyksen nouseva
    F_TRIG_MEMORY:F_TRIG; //Laskeva reuna muistinäpp.
    jotta pystyy asettamaan tarvittaessa uuden arvon muistiin
    Set:BOOL; // Asetakirkkaus bitti
END_VAR

```

Toimilohkon koodi:

```

R_TRIG_ON(CLK:=VisuOn);
R_TRIG_OFF(CLK:=VisuOff);
R_TRIG_Plus(CLK:=VisuPlus);
R_TRIG_miinus(CLK:=VisuMiinus);
F_TRIG_MEMORY(CLK:=VisuMemory);

```



```

IF R_TRIG_ON.Q=TRUE THEN
    EndLevel:=250;
    start:=TRUE;
END_IF

IF R_TRIG_OFF.Q=TRUE THEN
    EndLevel:=0;
    Star:=TRUE;
END_IF

IF EndLevel<250 AND R_TRIG_Plus.Q AND Busy=FALSE THEN
// Nostaa kirkkautta 10, ...busy=false estää tilan siirtymisen
kun valaistuksen taso siirtymässä
    EndLevel := EndLevel + 10;
    Start:= TRUE;
END_IF
IF EndLevel>100 AND R_TRIG_miinus.Q AND Busy=FALSE THEN
// himmentää 50, ...busy = false estää tilan siirtymisen kun
valaistuksen taso siirtymässä;
    EndLevel := EndLevel - 10;
    Start:= TRUE;
END_IF

Memorytimer (IN:= Visumemory, PT:= T#3S);
IF Memorytimer.Q THEN
    VisuMemoryLevel := EndLevel;
END_IF

IF F_TRIG_MEMORY.Q = TRUE THEN
    EndLevel:= VisuMemoryLevel;
    Start:=TRUE;
END_IF
IF Set=TRUE THEN
    Endlevel:= Visulevel;
    start:=TRUE;
END_IF
IF EndLevel <> ActualLevel THEN
    Start:= TRUE;
END_IF

Lights (bOn:=LightOn,bOff:=LightOff,bStart:=Start,
        nEndLevel:=EndLevel,
        nActualLevelMasterDev=>ActualLevel,
        bBusy=>Busy,bError=>Error,nErrorId=>ErrorID,
        nAddr:=Addr,eAddrType:=AddrType,
        nMasterDevAddr:=MasterDevAddr,tRampTime:=T#1S,
        stCommandBuffer:=internalcommunication.Dali4_stComBuf);

    Start := FALSE;

```