



Karelia-ammattikorkeakoulu
Insinööri, talotekniikka (AMK)

Verhomoottorin ohjauksen toteuttaminen PLC-ohjelmoinnilla

Santeri Turunen

Opinnäytetyö, toukokuu 2025

www.karelia.fi



OPINNÄYTETYÖ
Toukokuu 2025
Talotekniikan koulutus

Tikkarinne 9
80200 JOENSUU
+358 13 260 600 (vaihde)

Tekijä
Santeri Turunen

Nimeke
Verhomoottorin ohjauksen toteuttaminen PLC-ohjelmoinnilla

Toimeksiantaja
Karelia-ammattikorkeakoulu

Tiivistelmä

Opinnäytetyön aiheena oli suunnitella ja toteuttaa verhomoottorinohjaus Beckhoff Twincat -ohjelmalla. Tavoitteena työssä oli nykyisen verhomoottorinohjausjärjestelmän nykyaikaistaminen, sekä demolaitteiston saaminen hyötykäyttöön. Työn toimeksiantajana toimi Karelia-ammattikorkeakoulu. Ohjelman alustava käyttötarkoitus on toimeksiantajan laboratoriossa opetuskäyttö.

Verhomoottorinohjauksen suunnittelun lähtökohtana oli aikaisempi järjestelmä ja sen toiminnot. Näiden pohjalta piti tutkia, mitkä ohjelmistokirjastot ja toimintolohkot soveltuvat parhaiten tavoiteltujen toimintojen toteutukseen. Lohkojen valinnan jälkeen aloitettiin varsinainen ohjelmointiosuus, ja ohjelmoinnin aikainen vianselvitys. Työn viimeisessä vaiheessa suoritettiin ohjelman testaaminen käyttötapatestauksena.

Valmiissa verhomoottorinohjausohjelmassa on kaikki toiminnallisuudet, jotka opinnäytetyön alkuvaiheessa määriteltiin ja ne testattiin toimiviksi. Johtopäätöksenä todetaan, että ohjelma soveltuu hyvin opetuskäyttöön ja muutamilla lisäyksillä, sekä parannuksilla se saataisiin ohjaamaan verhoja Karelia-ammattikorkeakoulun Wärtsilä kampuksen D-siipeen.

Kieli
suomi

Sivuja 28
Liitteet 4
Liitesivumäärä 12

Asiasanat
ohjelmoitava logiikkaohjain (PLC), verhomoottori, rakennusautomaatio



THESIS
May 2025
Degree Programme in Building Services
Engineering

Tikkarinne 9
80200 JOENSUU
FINLAND
+358 13 260 600 (switchboard)

Author
Santeri Turunen

Title
Implementing Curtain Motor Control with PLC Programming

Commissioned by
Karelia University of Applied Sciences

Abstract

The purpose of the thesis was to design and implement a curtain motor control with the Beckhoff TwinCAT program. The aim of the thesis was to modernize the current curtain motor control system, and to utilize already existing demo equipment provided by the client. The thesis was commissioned by Karelia University of Applied Sciences. The preliminary purpose of the program is educational use in the client's laboratory.

The starting point for the design of the curtain motor control was the earlier system and its functions. Based on these, it was necessary to examine which software libraries and function blocks are best suited for the implementation of the desired functions. After selection of function blocks, the actual programming part began. This included troubleshooting during programming. The final stage of the work was to perform testing of the program as use case testing.

The complete curtain engine control program has all the functionalities that were defined in the early stage of the thesis, and these were tested to work. The conclusion is that the program is well suited for teaching use and with a few additions, and improvements, it would be able to control the curtains of the D-wing of the Wärtsilä campus of Karelia University of Applied Sciences.

Language
Finnish

Pages 28
Appendices 4
Pages of Appendices 12

Keywords
programmable logic controller (PLC), curtain motor, building automation

Sisältö

1	Johdanto	5
2	Ohjauksen tekninen tausta	6
2.1	Rakennusautomaatiojärjestelmät yleisesti	6
2.2	Tietoturva automaatiojärjestelmissä	6
2.3	Opinnäytetyön laitteisto	8
2.3.1	Beckhoff-demopöytä	8
2.3.2	Beckhoff CX5140-logiikka	9
2.3.3	Digitaaliset IO-kortit	9
2.3.4	Analogiset IO-kortit	10
2.3.5	Verhomoottori Somfy	10
2.4	Beckhoff TwinCat 3	11
2.5	Standardi IEC 61131-3	12
2.6	Automaatiojärjestelmän toiminnanvarmistaminen	12
2.7	Aurinkosuojaus lyhyesti	13
3	Opinnäytetyön kulku	14
3.1	Opinnäytetyö menetelmät	14
3.2	Projektisuunnitelma	14
3.3	Aikaisempi Wago-järjestelmä	15
3.4	Ohjauksen suunnittelu ja toteutus TwinCat:issä	16
3.4.1	Toimintalohkojen valinta	17
3.5	Haasteet työn aikana	18
3.6	Ohjelman toiminnan varmistaminen	19
4	Ohjelman kuvaus	21
4.1	Pääohjauslohkon toiminta	21
4.2	Synkronointi lohko	22
4.3	Antureiden skaalaus ja muuttaminen	23
4.4	Lämmityksen ja jäähdytyksen avustus	24
4.5	Viihtyvyyden olosuhteet arkipäivä ja yö	24
4.6	Viikonlopun aikainen ohjaus	25
4.7	Automaattiohjauksen prioriteettilohko	25
4.8	Kellonaika	25
4.9	Web- käyttöliittymä	26
5	Pohdinta	26
5.1	Parannusideat ja jatkokehitys	27
	Lähteet	28

Liitteet

Liite 1	Ohjelma
Liite 2	Toiminnan tarkastamisen pöytäkirja
Liite 3	Web-käyttöliittymä
Liite 4	Toimintakaavio

1 Johdanto

Opinnäytetyön aiheena on suunnitella ja toteuttaa verhomoottorinohjaus toimeksiantajalla olemassa olevalle Beckhoff-laitteistolle. Kyseistä verhomoottorinohjaus ohjelmaa on tarkoitus käyttää aluksi opetuskäytössä. Myöhemmässä vaiheessa verhomoottorinohjaus on mahdollista siirtää Karelia-ammattikorkeakoulun kampuksen D-siiven aulatilaa ohjaamaan siellä olevia moottoroituja verhoja. Tavoitteena on, että ohjauksesta tulisi nykyaikaisempi ja se olisi myös mahdollista tulevaisuudessa siirtää Microsoftin tarjoamaan Azure pilvipalveluun osaksi muita automaatiojärjestelmiä. Aihe on Karelia-ammattikorkeakoulun ehdottama. Työlle lähtökohtana toimi Karelian Wärtsilä kampuksen aulatilassa oleva edellinen järjestelmä, joka on toteutettu Wagon järjestelmällä. Myös tämä aikaisempi järjestelmä on toteutettu opinnäytetyönä.

Vaikka työn tavoitteissa ei suoraan huomioida energiatehokkuutta, se liittyy silti kiinteästi rakennusautomaatiojärjestelmiin. Rakennusautomaatiojärjestelmillä pyritään helpottamaan rakennusten ylläpitämistä, mutta yksi niiden tarkoitus on energiatehokkuuden lisääminen. Aihe on erittäin mielenkiintoinen, koska rakennetun ympäristömme energiatehokkuuden merkitys on kehittymässä joka päivä entistä tärkeämmäksi osaksi rakentamista ja kiinteistöjen ylläpitämistä varten. Kaikkein energiatehokkaimmaksi kiinteistöt saadaan fiksuilla ja edistyksellisillä ratkaisuilla ja yksi näistä ratkaisuista on taloteknisten järjestelmien automatisointi. Hyvin toteutetulla verhomoottorin ohjauksella saadaan hyödynnettyä auringon tuottama lämpö silloin, kun sitä tarvitaan, mutta sillä saadaan myös ehkäistyä tilojen yllämmitystä ja täten pienennettyä jäähdyttämiseen tarvittavan energian määrää. Oikein toteutettuna automatisoiduilla verhoilla voidaan parantaa myös tilojen viihtyvyyttä huomattavasti.

2 Ohjauksen tekninen tausta

2.1 Rakennusautomaatiojärjestelmät yleisesti

Rakennusautomaatiolla tarkoitetaan järjestelmää tai järjestelmäkokonaisuuksia, joilla automatisoidaan erilaisia toimintoja rakennuksissa.

Rakennusautomaatiolla ohjataan ja valvotaan nykyaikaisissa rakennuksissa niiden lämmitys- ja ilmanvaihtojärjestelmiä, jäähdytysjärjestelmiä, sekä valaistusjärjestelmiä. (Meriläinen 2024.)

Rakennusautomaatio mahdollistaa sen, että rakennusten ylläpitämisestä tulee helpompaa kuin aikaisemmin ja kiinteistöjä pystytään hallitsemaan mahdollisesti etänä. Rakennusautomaatiolla saadaan toteutettua entistä tehokkaampia ja tarkempia järjestelmiä. Samalla ne voidaan monesti myös yhdistää yhteen kokonaisuuteen mistä voidaan hallita kiinteistön tai rakennuksen kaikkia järjestelmiä samasta paikasta. (Meriläinen 2024.)

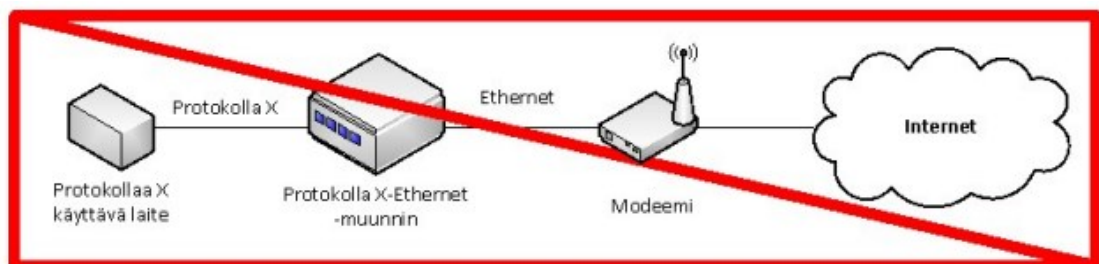
Nykyaikaiset rakennusautomaatiot Voidaan myös yhdistää pilveen, jolloin ei ole välttämätöntä käyttää perinteistä valvonta-alakeskusta. Näin ollen myös järjestelmästä saatavaa dataa voidaan käytännössä tallentaa todella suurilla määrillä verrattuna perinteisiin järjestelmiin. Tämä mahdollistaa datan perusteella tehtävät raportit ja erilaiset analyysit mitä rakennuksen loppukäyttäjä voi halutessaan hyödyntää.

2.2 Tietoturva automaatiojärjestelmissä

Tietoturva rakennusten automaatiojärjestelmissä on vielä valitettavasti liian usein todella huonosti toteutettu tai sitä ei ole toteutettu lainkaan. (Hursti 2016) Kyberturvallisuuskeskus tekee vuosittain tutkimuksia löytääkseen rakennus- ja teollisuuskäytössä olevia suojaamattomia laitteita. Tarkoituksena tiedottaa

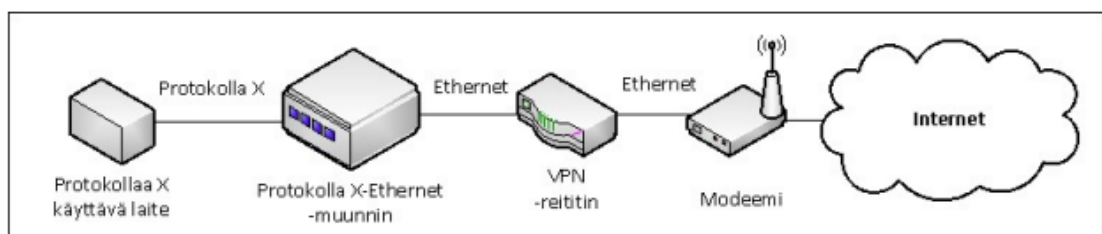
suojaamattomien laitteiden omistajille, jotta tietoturvaa saataisiin parannettua automaatiojärjestelmissä. (Kyberturvallisuuskeskus 2020)

Usein laitteistojen suojaamattomuus johtuu siitä, että laitteiston loppukäyttäjällä ei ole riittävää tietoa ja kiinnostusta laitteiden suojaamista varten, ja ajatellaan tämän vain olevan turha lisäkustannus kokonaisuuteen. Tämän seurauksena laite saatetaan kytkeä internettiin ilman suojausta. Alla olevassa kuvassa 1 esitetään laitteen kytkeminen internettiin, mutta tällä tavalla toteutettuna yhteys on käytännössä täysin ilman minkäänlaista suojaa, näin se on altis erilaisille hyökkäyksille internetin välityksellä.



Kuva 1. Esimerkki miten laitteita ei kytketä internettiin (kyberturvallisuuskeskus 2020)

Kuvassa 2. esitetään yksi malli, miten laitteen voi kytkeä turvallisesti internettiin. Kuvassa suojattu yhteys toteutetaan VPN-reitittimen avulla. Myös yleisemmin tietokoneista tunnettuja palomureja voidaan käyttää suojaamaan rakennusautomaatiojärjestelmien yhteyksiä.



Kuva 2. Automaatiolaitteen turvallinen kytkeminen internettiin (kyberturvallisuuskeskus 2020)

Vuonna 2020 toteutetussa kartoituksessa kyberturvallisuuskeskus tarkasti 1280 verkkoavaruutta ja noin 13 miljoonaa IP-osoitetta. Rakennusautomaation osalta

näistä verkkoavaruuksista ja IP-osoitteista löytyi n. 800 suojaamatonta laitetta. (kyberturvallisuuskeskus 2020) Uhkana rakennusautomaation suojaamattomuuden osalta ei välttämättä ole se, että joku ulkoinen taho pääsee käsiksi talon teknisiin järjestelmiin ja aiheuttaa kiinteistölle vahinkoa. Todellinen uhka on, että laitetta käytetään esimerkiksi osana palvelunestohyökkäystä. Monissa rakennusautomaation laitteissa on esimerkiksi mahdollisuus lähettää- ja vastaanottaa tekstiviestejä, jotka voivat tiettyihin paikkoihin lähetettynä olla todella kalliita. (Kyberturvallisuuskeskus 2020) Usein laitteistojen suojaamattomuus johtuu siitä, että laitteiston loppukäyttäjällä ei ole riittävästi tietoa ja kiinnostusta laitteiden suojaamista varten, ja ajatellaan tämän vain olevan turha lisäkustannus kokonaisuuteen.

Rakennusautomaatiojärjestelmien tietoturvan voi jaotella neljään osaan:

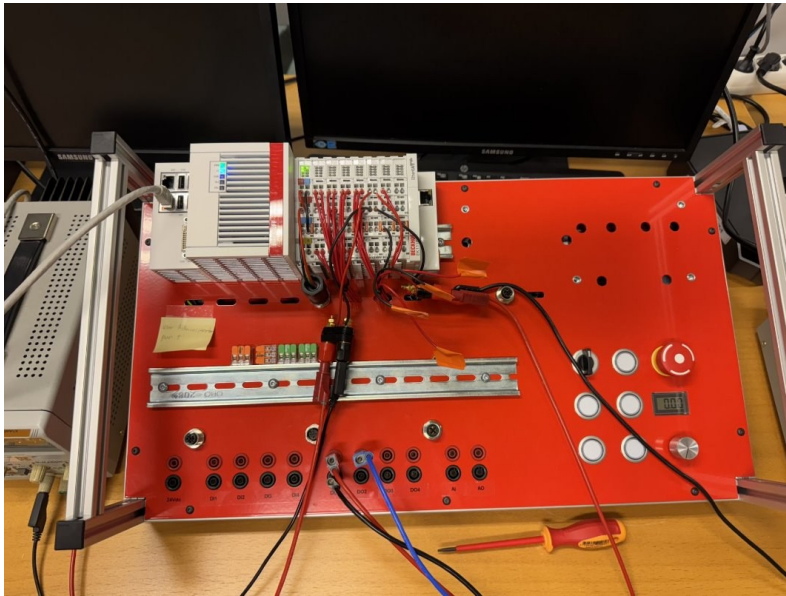
1. Valvomo PC:n liitäntä verkon yli tietoturvallisesti
2. Valvomo PC:n tietoturva, Windows päivitykset ja virusturva
3. Rakennusautomaatiojärjestelmän valvomo-ohjelmiston tietoturva
4. Rakennusautomaatiojärjestelmän valvomo-ohjelmiston käyttäjäpolitiikka

(Hursti 2016.)

2.3 Opinnäytetyön laitteisto

2.3.1 Beckhoff-demopöytä

Ohjelman testaamisvaiheessa hyödynnetään Karelia-ammattikorkeakoululta jo valmiiksi löytyvää Beckhoff-demopöytää. Demopöytä koostuu CX5140-logiikasta sekä yhteensä 2 kpl digital input -kortteja, 2 kappaletta digital output - kortteja, 1 kappale analog input -kortteja, 1 kappale analog output -kortti, sekä 1 Ethernet lisäosakortti. Tämän lisäksi korteille oli valmiiksi kytketty hätäseis - painike, 5 kappaletta painonappeja, yksi keinukytkin ja yksi pontetiometri, joka oli kytketty näyttöön. Alla olevassa kuvassa esitettynä Beckhoff-demolaitteisto ja testauksen aikaiset kytkennät.



Kuva 3. Beckhoff-demolaitteisto

2.3.2 Beckhoff CX5140-logiikka

Logiikassa on neljäytiminen Intel prosessori, jonka kellotaajuus on 1.9 GHz. Prosessori mahdollistaa logiikan moniydinteknologian käyttämisen osana yhdistettyjä laitteistoverkkoja. (Beckhoff Automation 2025) Logiikassa on valmiina kaksi Ethernet-porttia, näitä voidaan lisämoduuleilla tarpeen mukaan lisätä, sekä neljä USB- porttia ja myös DVI-I liitännä näyttöä varten. Logiikan käyttöjärjestelmä on Windows 7, mutta se tukee myös uudempia Windows versioita. (Beckhoff Automation 2025)

2.3.3 Digitaaliset IO-kortit

Digitaaliset tulo- ja lähtökortit ovat kortteja, joiden avulla logiikalle saadaan mahdollisuus lukea ja kirjoittaa digitaalisia signaaleja. Logiikalle saadaan näiden kautta tuotua ulkoisia laitteita, jotka käyttävät digitaalisia signaaleja toimiakseen, nämä laitteet antavat binääriarvo signaaleja logiikalle. Digital output korttiin kytkettävä laite voi siis olla esimerkiksi valoanturi tai liiketunnistin. (Optimal Controls and Building Analytics Inc 2025)

Demopöydän digitaaliset kortit ovat Beckhoff EL1004(DI), EL1008 (DI), EL2004 (DO) ja EL2008 (DO). Korttien viimeinen numero kertoo kortin kanavien määrän. Jokaisen kortin toimintajännite on 24VDC.

2.3.4 Analogiset IO-kortit

Analogisten tulo- ja lähtökorttien ansiosta logiikalle saadaan mahdollisuus kytkeä laitteita, jotka antavat muuttuvaa mittaustulosta, esimerkiksi kosteus, lämpötila ja paineanturit. Analogisilla lähtökorteilla voidaan myös ohjata laitteiden nopeutta tai asentoa ja tästä hyvä esimerkki on lämpimän käyttöveden säätöventtiili. (Roberts 2019.)

Demopöydän analogisia kortteja ovat Beckhoff EL3102 (AI), EL4132 (AO), molempien korttien käyttöjännite on -10 ja +10 V välillä, tämä muunnetaan digitaaliseen muotoon 16 bitiksi.

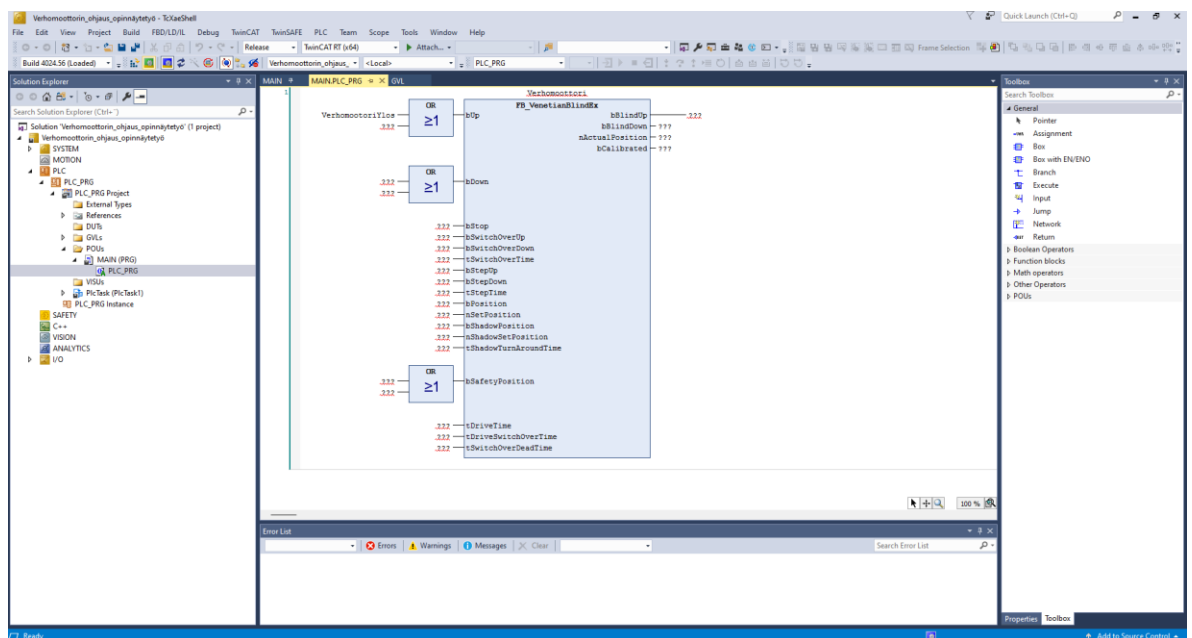
2.3.5 Verhomoottori Somfy

Ohjelmiston testaamista varten käytettiin koulun laboratoriotilasta löytyvää Somfy LT50- verhomoottoria. Moottori käyttää yksivaihevirtaa 230 V, 50 Hz. Verhomoottorin pyörimissuunta voidaan määrittää kytkennöillä, eli moottorinpyörimissuunta määritellään johtojen kytkemisellä. Verhomoottorissa on sisäiset rajakytkimet, joiden avulla voidaan määrittää verhon liikkeille rajat. (Somfy tekninen liite 2025.)

2.4 Beckhoff TwinCat 3

Beckhoff TwinCat 3- ohjelmisto on tarkoitettu rakennusautomaatiosovellusten tehokkaaseen ja yksinkertaiseen suunnitteluun. Ohjelmaan on kehitetty todella laajasti eri ohjelmistotyökaluja ja kirjastoja, joiden ansiosta ohjelmalla voidaan toteuttaa automaatiosovellustensuunnittelua rakennus kuin myös teollisuuden käyttötarkoituksiin. (Beckhoff Automation 2024.) Ohjelma perustuu standardiin IEC 61131-3, jossa on määritetty ohjelmointikielet, joista tässä työssä käytetään function block diagram (FBD) ohjelmointia. Sen etuja ovat sen graafinen lähestymistapa, joka tekee loogisen rakenteen helpommin ymmärrettävämmäksi, joten suunnittelu ja vianhaku on helpompaa. Yksi sen suurimmista eduista on kuitenkin sen uudelleen käytettävyys, koska suunnittelijat pystyvät hyödyntämään valmiita funktiolohkoja. (Control Engineering 2007)

Tärkein ohjelmistokirjasto opinnäytetyön kannalta on building automation basics, tämä ohjelmistokirjasto sisältää kaiken tarpeellisen rakennusautomaatiojärjestelmän sovellussuunnitteluun. Kirjasto on jaettu 7 osaan: energian hallinta, julkisivu, suodatin toiminnot, valaistus, tilanteiden hallinta, signaalien käsittely, sekä aikakytkimet. (Beckhoff Automation 2024)



Kuva 4. Beckhoff TwinCat 3- ohjelmiston näkymä

2.5 Standardi IEC 61131-3

Tämä standardi on osa laajempaa sarjaa IEC 61131, joka käsittelee laajemmin teollisuuden automaatiojärjestelmien standardeja. Standardi määrittää ohjelmoitavien logiikoiden ohjelmointikielet, sekä niiden käyttämisen automaatiojärjestelmissä. IEC 61131-3 keskittyy erityisesti viiteen eri ohjelmointikielen, joita käytetään ohjelmoitavien logiikoiden ohjelmoimiseen:

1. Structured Text (ST) - Tekstillinen
2. Ladder Diagram (LD) – Graafinen
3. Instruction List (IL) - Tekstillinen
4. Sequential Function (SFC) - Graafinen
5. Function Block Diagram (FBD) – Graafinen

(SFS-EN 61131-3.)

Standardissa määritetään myös ohjelmoinnissa käytettävät datatyypit ja muuttujat, jotka helpottavat ohjelman suunnittelua, toteuttamista ja sen myöhempää muokkaamista. Standardissa on myös määritetty Program Organization Unit (POU), joka on ohjelman rakenne.

2.6 Automaatiojärjestelmän toiminnanvarmistaminen

Ennen kuin rakennusautomaatio järjestelmä voidaan ottaa käyttöön tai luovuttaa loppukäyttäjälle sen toiminta pitää varmistaa tekemällä ohjeistuksen mukaiset tarkastukset. (ST-käsikirja 17 2018, 160)

ST-käsikirjan mukaan ainakin seuraavat tarkastukset täytyy suorittaa ennen järjestelmän luovuttamista

- Asennustapatarkastukset
- Järjestelmäkohtaiset urakoitsijan tarkastukset
- teknisten järjestelmien ja laitteiden käyttöönotot laitetoimittajien kanssa
- rakennuttajan toimintakokeet
- LVIAS-järjestelmien yhteiskoekäyttö
- Sähkökatkotesti

- RAU-järjestelmän valvomo-ohjelmiston toimintojen tarkastus
(ST-käsikirja 17 2018, 160)

2.7 Aurinkosuojaus lyhyesti

Aurinkosuojauksen peruseräite on estää sisätiloihin lankeavaa auringon säteilyä. Yleisesti ottaen auringon säteily voidaan jakaa kolmeen osaan: suora säteily, hajasäteily ja heijastunut säteily. (RT 07-11300 2018, 1)

Aurinkosuojauksella tavoitellaan auringonsäteilyn negatiivista vaikutusta kiinteistön energiankulutukseen ja sisäolosuhteisiin (RT 07-11300 2018, 2)
Aurinkosuojaus voidaan toteuttaa hyödyntämällä esimerkiksi kohteen läheisyydessä olevaa kasvustoa kuten pensaita ja puustoa, ympärillä olevia rakennuksia, rakenteellisia aurinkosuojauksen tapoja, esimerkiksi lipat ja räystäät ja oikeanlaiset ikkunan valitsemiset, tähän voidaan myös sisällyttää jo suunnitteluvaiheessa mietityt tilojen sijoittelu ja suuntaaminen sekä manuaaliset tai automatisoidut verhot ja markiisit. (RT 07-11300 2018, 2)

Aurinkosuojauksella on iso merkitys kiinteistön energiankulutukseen, oikein toteutettuna sillä voidaan estää jäähdytystarpeen vähentämistä ja myös toisinpäin varastoida energiaa vaipan ulkopuolisiin lasitettuihin tiloihin. Joten nykyaikaisessa rakentamisessa on todella tärkeää panostaa jo suunnitteluvaiheessa aurinkosuojaukseen, näin välttää myös lisäkustannuksilta, joita voi tulla kesällä, kun varsinkin asuinkiinteistöt lämpenevät todella paljon.

3 Opinnäytetyön kulku

3.1 Opinnäytetyö menetelmät

Opinnäytetyö on luonteeltaan toiminnallinen ja näin tavoitteen määrittäminen ei ollut vaikeaa. Tavoitteena on luoda toimiva ohjaus verhomootoreille käyttäen Beckhoff- logiikkaa, sekä heidän tarjoamaa TwinCat 3- ohjelmointiympäristöä. Verhomootorin ohjauksen tulee olla sellainen, että se voidaan jatkossa liittää pilvipalveluun osaksi muuta Karelia-ammattikorkeakoulun rakennusautomaatiota. Vaikeampi osuus on rajata opinnäytetyön aihe riittävän pieneksi, koska verhomootorin ohjaukseen liittyy hyvin paljon erilaisia asioita, joilla on suuri merkitys ohjauksen toimivuuteen. Ohjauksen voi myös perustaa todella moneen erilaiseen asiaan, jolloin on todella vaativaa saada aikaiseksi mahdollisimman hyvä ohjaus, joka toimisi niin opetuskäytössä kuin myös todellisuudessa D-siiven aulatilassa ohjaamassa verhoja.

Verhomootoreita varten tässä opinnäytetyössä suunnitellaan ja toteutetaan ohjelma, jossa on lämmityksen ja jäähdytyksen tukemisen toiminto ja aikataulupohjainen ohjaus. Tällä pyritään siihen, että verhojen ohjaus toimii arkisin normaaleina aukioloaikoina. Samalla pyritään parantamaan tilojen viihtyvyyttä, ottaen huomioon lämpötilan sekä valoisuuden. Eli päivisin ja öisin olisi omat ohjausparametrit, samoin kuin arkisin ja viikonloppuisin ohjaus toimisi eri tavalla. Verhoon halutaan mahdollisuus ohjata sitä käsin ylös ja alas käyttämällä painonappia, tämän lisäksi sovellukseen halutaan siivous- ja huoltokytin, joka ohittaa kaiken muun verhon ohjauksen määritetyksi ajaksi, jotta saadaan tehtyä siivous tai korjaustoimenpiteitä turvallisesti.

3.2 Projektisuunnitelma

Opinnäytetyön prosessi alkoi perehtymällä jo olemassa olevaan järjestelmään ja sen toimintoihin. Tarkoituksena oli sisäistää nykyinen järjestelmä ja alkaa miettiä mitä vanhasta järjestelmästä voi hyödyntää. Lisäksi pohdittiin mitä

ominaisuuksia järjestelmään halutaan, jotta saavutetaan haluttu lopputulos työn valmistuessa.

Seuraavaksi tehtiin opinnäytetyösuunnitelma ja se toimitettiin opinto-ohjaajalle. Tämän jälkeen työ hyväksyttiin ja määrättiin työlle ohjaaja. Tämän jälkeen varsinainen työ alkoi ja piti perehtyä jo olemassa olevaan verhomoottorinohjaus- ohjelmaan, jotta pystyttiin hahmottamaan millainen ohjelma uudessa Beckhoff- ympäristössä toteutetaan.

Aluksi ideana oli toteuttaa työ kokonaisuudessaan etänä, eli myös ohjelman testaamisen olisi voinut toteuttaa etänä. Laitteistoon olisi pitänyt luoda turvallinen etäyhteys käyttämällä VPN verkkoa. Tämän lisäksi verholle olisi pitänyt löytää pysyvä paikka, jotta siihen olisi voitu asentaa valvontakamera kuvaamaan, näin verhoa olisi pystynyt seuraamaan sen etätestauksen aikana. Tämä osoittautui kuitenkin liian haasteelliseksi ja pelkästään etäyhteyden turvallinen muodostaminen ei ollut mahdollista, joten päädyttiin ratkaisuun, että ohjelman suunnittelu ja rakentaminen tapahtui etänä, mutta testaaminen käytiin tekemässä kampuksen laboratoriotilassa.

3.3 Aikaisempi Wago-järjestelmä

Aikaisempi järjestelmä on toteutettu opinnäytetyönä ja sen tekijöinä ovat Pärkkinen ja Tarnanen, opinnäytetyön nimike on Auringonvalon ja -lämmönhallinta kiinteistössä. Opinnäytetyö on valmistunut vuonna 2019. Työn lähtötiedot ja ohjelman suunnittelun pohjana hyödynnettiin edellisen verhomoottorin ohjauksen tekijöiden opinnäytetyötä, automaation toimintakaaviota sekä lähdekoodia. Toimintakaavion avulla sai nopeasti käsityksen vanhan järjestelmän toiminnasta, ja näin pystyin nopeasti alkaa suunnitella oman ohjelmani toteutusta. Edellinen järjestelmä oli rakennettu käyttämällä Wagon logiikkaa, ja ohjelma oli tehty käyttämällä heidän ohjelmointiympäristöänsä eCockpit, joka perustuu myös standardiin IEC31131-3. (Pärkkinen & Tarnanen. 2019) Uudesta ohjelmasta löytyy siis samat toiminnallisuudet kuin aikaisemmasta järjestelmästä nämä esitetty alapuolella.

1. Siivouskytkin
2. Kalibrointiajo
3. Käsiohjaus
4. Lämmityksen ja jäähdytyksen avustus -automaatiikka
5. Viihtyvyyys ja mukavuus olosuhteet -automaatiikka.

(Pärkkinen & Tarnanen. 2019)

Vanhasta ohjelmasta hyödynsin ohjelman suunnittelussa sen rakennetta, ja suunnittelun alku oli tästä syystä nopeampaa. Ohjelma on rakennettu niin, että on yksi ohjaava päälohko, joka vie käskyt verhomootoreille. Päälohkon alapuolella jokaiselle toiminnolle on oma lohko. (Pärkkinen & Tarnanen. 2019) Aikaisemmassa järjestelmässä on myös ollut käytössä Trace toiminto, jolla on pystytty lokittamaan mittausten dataa ja pystytty piirtämään niistä graafeja ja näiden avulla optimoimaan järjestelmän toimintaa datan perusteella (Pärkkinen & Tarnanen. 2019) Uudessa järjestelmässä ei ole tämä ominaisuus käytössä, mutta sille on tuki ja mahdollisuus lisätä se sinne.

3.4 Ohjauksen suunnittelu ja toteutus TwinCat:issä

Lähtökohtana ohjelman suunnittelussa ja toteutuksessa oli, että se toteutetaan noudattamalla standardin IEC 61131-3 määrittämällä ohjelmointi kielellä, sekä sen määräämillä datatyypeillä. Tämän lisäksi ohjelman ulkoasun tulisi olla siisti vaikkei tätä suoranaisesti määritellä standardissa. Ohjelman ulkoasun siisteyteen vaikuttaa olennaisesti muuttujien nimeäminen ja se, että muuttujien tyypit ovat samassa linjassa.

```
VAR
    //Verhomoottorin ohjainlohko
    VerhomoottorinOhjainLohko:          FB_VenetianBlindEx;
    Kellonaika:                        FB_HVACGetSystemTime;
    usiVerhonAsetus:                   USINT;
    AsentoNollaus:                     rs;
    Synkronointi:                      rs;
```

Kuva 5. muuttujien nimeämisen ulkoasu

Twincat -ohjelmistossa projekti aloitettiin luomalla projekti, sekä lisättiin projektiin tarpeelliset ohjelmistokirjastot, jotta voidaan hyödyntää ohjelmistokirjastoista valmiita toimintalohkoja. Tärkeimmät ohjelmistokirjastot vakioden lisäksi olivat Building Automation, Building Automation Basic, HVAC ja visuaalisaatio elementit, joita käytettiin Web-visun rakentamisessa.

Ohjelman suunnittelun pohjana oli luontevaa käyttää edellisestä Wagon järjestelmästä perusrakennetta. Eli yksi lohko, joka tekee ohjaamisen ja sen alapuolella toiminnoille omat lohkot, jotka yhdistetään prioriteettikytkimen avulla yhdeksi muuttujaksi. Kun ohjelman peruseriaate oli saatu selville, lähdin etsimään oikeita ominaisuuksia omaavia toimintolohkoja.

3.4.1 Toimintalohkojen valinta

Ensimmäinen valinta oli verhomoottoriohjainlohko, bechhoff tarjoaa building automation kirjastossa neljää erilaista ohjauslohkoa, jotka ovat FB_RoofWindow, FB_VenetianBlind, FB_VenetianBlindEX ja FB_VenetianblindEX1Switch. Näistä kaksi ensimmäistä eivät soveltuneet tämän työn kannalta ohjaaviksi lohkoiksi, koska toinen oli kattoikkunan varjostukseen ja toisessa ominaisuudet olivat yksinkertaisesti liian suppeat. Valittavaksi jäi siis kahden väliltä, joissa ainoa ero oli tulo bSwitch, jota ei ollut FB_VenetianBlindEX- lohkoissa. Sen tarkoitus oli pystyä kääntämään verhon ohjaus ylös tai alas nappia painamalla, kun tätä toimintoa ei tarvittu oli valinta loppujen lopuksi helppo.

Seuraavaksi vuorossa oli valittavana lämmitystä ja jäähdytystä tukeva lohko, sekä aikatauluihin perustuvan ohjauksen lohko. Aluksi vaikutti siltä, että parhaat vaihtoehdot näihin olisivat FB_BA_ThermoAutomatic ja FB_BA_Facade_TwilightAutomatic, ja ohjelmaa ehdin näiden lohkojen päälle jo luoda. Viimeisiä lohkoja eli viikonloppu ohjausta ja prioriteettikytkintä etsiessäni huomasin, että kun viikonlopun aikaiseen ohjaukseen tarkoitettu lohko on FB_BAR alkuinen, niin FB_BA alkuiset lohkot eivät sovellu näiden kanssa

yhteen, koska prioriteettilohko hyväksyy vain ainoastaan FB_BAR tyyppin ohjelmalohkoja. Tämän takia jouduin vielä myöhemmässä vaiheessa vaihtamaan FB_BA_ThermoAutomatic lohkon FB_BAR_SunblindThermoAutomatic lohkoon ja FB_BA_Facade_TwilightAutomatic lohkon FB_BARSunblindTwilightAutomatic lohkoon. Näin ollen ohjelmassa käytetyt lohkot olivat siis seuraavat:

FB_VenetianBlindEX
FB_BARSunblindThermoAutomatic
FB_BARSunblindTwilightAutomatic
FB_BARSunblindEvent
FB_BAR_SunblindPrioritySwitch
FB_HVACGetSystemTime

taulukko 1. työssä käytetyt toimintolohkot

3.5 Haasteet työn aikana

Ensimmäisenä testauspäivänä oletettiin, että testitietokone ja demolaitteisto voitaisiin yhdistää suoraan liittämällä laitteet Ethernet-kaapelilla ja muuttamalla testitietokoneen IP-osoite samaan osoiteavaruuteen logiikan kanssa.

Testitietokoneelta kuitenkin puuttui oikea versio Beckhoff TwinCAT -sovelluksesta, minkä seurauksena yhteyden muodostamisessa kesti odotettua kauemmin, ja ensimmäisestä testauspäivästä kului puolet pelkästään yhteyden muodostamiseen.

Kun yhteys saatiin lopulta muodostettua laitteiden välille ja IO-korttien lukeminen logiikasta onnistui, päästiin muuttujien linkittämiseen. Linkityksen yhteydessä havaittiin ongelma ohjelmassa, sillä muuttujien linkittäminen logiikan painonappeihin ei onnistunut. Tämän vuoksi jouduttiin luomaan uusi projekti ja lisäämään olemassa oleva ohjelma manuaalisesti siihen.

Linkittämisen yhteydessä huomattiin myös, että demopöytä oli alun perin kytketty virheellisesti. Potentiometri oli liitetty vain demopöydän diginäyttöön, mutta ei muihin komponentteihin. Vaikka arvo muuttui diginäytössä potentiometriä pyöritettäessä, se ei siirtynyt eteenpäin mihinkään. Tämä aiheutti lisähaastetta, sillä tarkoituksena oli käyttää potentiometriä simuloimaan ohjelmassa käytettyä anturia. Ongelma ratkaistiin käyttämällä jännitelähteitä anturien simulointiin.

Ensimmäiselle testikerralle oli varattu kaksi kokonaista päivää, mutta alkuvaiheen haasteiden vuoksi ne eivät riittäneet ohjelman täydelliseen testaamiseen. Kuitenkin saatiin tarvittavat havainnot, joiden avulla korjausta vaativat kohdat pystyttiin tunnistamaan.

Koulutuksen ansiosta onnistuttiin luomaan testausympäristö myös kotitietokoneelle, jossa ohjelman simulointi oli mahdollista. Tämäkin osoittautui haastavaksi ja vaati BIOS-asetusten muuttamista. Simulointia varten Twincatin käyttöön piti varata erikseen kaksi ydintä kuudesta.

Työn suurimpana hidasteena oli puutteellinen osaaminen Beckhoffin järjestelmistä, ja paljon aikaa kului yksittäisten ohjelmalohkojen toiminnan ymmärtämiseen. Esimerkiksi lohkoissa FB_BARSunblindThermoAutomatic, FB_BARSunblindTwilightAutomatic ja FB_BARSunblindEvent on syöte eDataSecurityType, jonka arvon täytyy olla 1 eikä 0. Ilman tätä lohko ei toimi, vaikka sen aktivoitumisehdot muutoin täyttyvät. Tämän yksinkertaisen yksityiskohdan selvittämiseen kului kymmeniä tunteja, koska vikaa etsittiin aluksi kaikkialta muualta. Lopulta ratkaisu löytyi laitevalmistajalta.

3.6 Ohjelman toiminnan varmistaminen

Ohjelmalle ei tarvinnut tehdä niin kattavaa toiminnan varmistamista kuin mitä luvussa 2.7 on käsitelty, koska ohjelma tulee alustavasti käyttöön vain opetuskäyttöön. Ohjelman toiminnan varmistaminen tehtiin koulun kampuksella paikan päällä käyttämällä laboratorioissa olevaa verhomootoria, sekä aikaisemmin luvussa 2.3.1 kuvattua Beckhoff-demopöytää. Ohjelman

testaaminen jouduttiin haasteiden takia tekemään 2 osassa, ja ensimmäiset kaksi testauspäivää käytettiin demolaitteiston kasaamiseen, testitietokoneen ja demologiikan yhteyden muodostamiseen ja muiden haasteiden selvittämiseen. Toisen testipäivän jälkeen verhojen ohjaus toimi ainoastaan painonappien kautta ja tältä kerralta sain kaiken tarvittavan tiedon ohjelman korjaamiseen. korjauksien ja lisäyksien jälkeen, kun pääsin vihdoin uudelleen testaamaan ohjelmaa, aikaa ei kulunut kuin hetki ja ohjelma oli jo käynnissä simulointi tilassa.

Ohjelmiston testaamisen päätin suorittaa tekemällä simulaatioita erilaisista käyttötilanteista.

1. Ohjelman käynnistys
2. Painonappien toiminta
3. Kalibroinnin toiminta
4. Lämmitys & Jäähdytys
5. Aikaohjelmat
6. Siivouskytkin

Taulukko 2. Testatut ominaisuudet

Ensimmäisenä testinä oli laitteiston käynnistäminen ja tarkoituksena oli testata, lähteekö ohjelma automaattisesti ohjaamaan, kun se käynnistyy. Testin aikana paljastui, että vasta toinen ohjaukskäsky aktivoi verhomoottoriohjauksen. Syy tälle toiminnalle ei selvinnyt ja se on nyt ohjelman ominaisuus. Seuraavaksi testasin käsikytkimien, eli painonappien toimintaa. Painonapit ylös- ja alas toimivat jokaisella kerralla, kun niitä testin aikana painoin. Kolmantena testinä verhomoottorin kalibrointiajo, verhomoottorin ohjauslohko, synkronoi itsensä aina kun se saavuttaa toisen ääripään, testauksen aikana kuitenkin havaitsin, että ohjaukseen voi tulla epätarkkuutta ilman erillistä kalibrointilohkoa, joten testin jälkeen lisäsin ohjelmaan erillisen kalibrointiajon, jonka toiminta on kuvattu tarkemmin luvussa 4.2.

Seuraavana testauksen kohteena oli lämmitys- ja jäähdytyslohkon toiminta, tätä testasin simuloimalla valoisuus- ja lämpötila-antureita jännitelähteillä ja muutin ne oikeaan muotoon ohjelmassa muunnoslohkojen avulla. Molemmat

ominaisuudet toimivat halutulla tavalla. Lohkossa on myös niin sanottu passiivinen tila, kun ei ole tarvetta lämmittää tai jäähdyttää verho pidetään siinä asennossa mihin se on ennen tätä jäänyt. Aikaohjelmien testaaminen testattiin manipuloimalla ajanhakulohkon antamaa aikaa manuaalisesti. Viimeiseksi testasin vielä siivouskytkimen toimintaa, sen testaaminen tapahtui kääntämällä siivouskytkin aktiiviseksi ja toteamalla, että mikään muu ohjaus ei tapahtunut tuon kytkimen ollessa aktiivinen. Kirjasin kaikki havainnot ylös ja täytin myöhemmin tiedot toiminnanvarmistus pöytäkirjaan, liite 2.

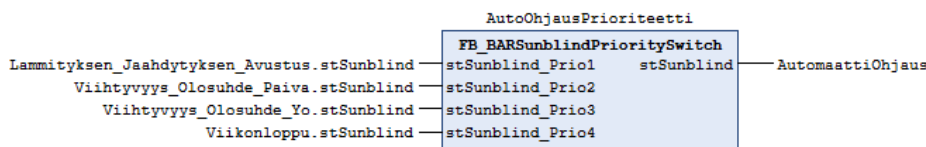
4 Ohjelman kuvaus

Ohjelma on toteutettu noudattamalla standardin IEC 31131-3 määrittämiä ohjeistuksia ja hyviä muuttujien nimeämiskäytäntöjä. Ohjelmointikielenä on käytetty Function Block Diagram- ohjelmointia, eli samaa kieltä kuin aikaisemmassakin järjestelmässä. Ohjelman lähdekoodi on esitetty liitteissä. (liite 1.) Ohjelman toimintaperiaatteena on, että yksi lohko toimii pääohjauslohkona ja tämän alla toiminnot on ohjelmoitu omina lohkoinaan ja ne tuodaan prioriteettikytkimen kautta pääohjauslohkolle. Ohjelman toiminta on kuvattu tarkemmin liitteessä 4 toimintakaavio.

4.1 Pääohjauslohkon toiminta

Pääohjauslohkon tarkoitus on toimia varsinaisena ohjaavana osana ohjelmistossa, mikä lähettää käskyn eteenpäin verhomoottorille. Pääohjauslohkona toimii FB_VenetianBlindEX, joka valikoitui ominaisuuksiensa vuoksi ohjaamaan verhomoottoreita. Lohkon tuloissa bSwitchOverUp ja bSwitchOverDown on kytketty manuaalisen ohjauksen muuttujat GVL_IO_Mapping_bManuaalinenOhjausYlos ja GVL_IO_Mapping_bManuaalinenOhjausAlas. Tämä mahdollistaa verhon liikuttamisen ylös- ja alas painonapeilla.

Lohkon tuloa bPosition käytetään automaattiohjauksien muuttujille, eli yhteensä 4 eri muuttujaa, jotka näkyvät kuvassa 6.



Kuva 6. Prioriteettikytkin

Nämä muuttujat tuovat käskyn moottorille aktivoida oikean prioriteetin verhon positio. Lohkon toiminnan vuoksi eri prioriteettien toiminta varmistettiin nollaamalla bPosition- tulo TRUE → FALSE aina kun edellisen käskyn kertoma positio verhossa on saavutettu, ilman tulon nollaamista lohko ei ota uutta käskyä vastaan, joten ilman tätä lisäystä verhon ohjaus ei toimisi oikein. Lohkon tulo nSetPosition kertoo tämänhetkisen käskyn prosentuaalisen asentotiedon.

bSafetyPosition tuloon määriteltiin napilla aktivoitava siivoustila, jonka ollessa aktiivisena verhot ajetaan ylös ja verhoa ei pysty liikuttamaan mihinkään suuntaan. Siivouskytkin pitää siis aina siivouksen loputtua palauttaa takaisin OFF- asentoon, että verhomoottorin automaattinen ohjaus toimii. Tähän tuloon määritettiin myös erillinen verhomoottorin synkronointi.

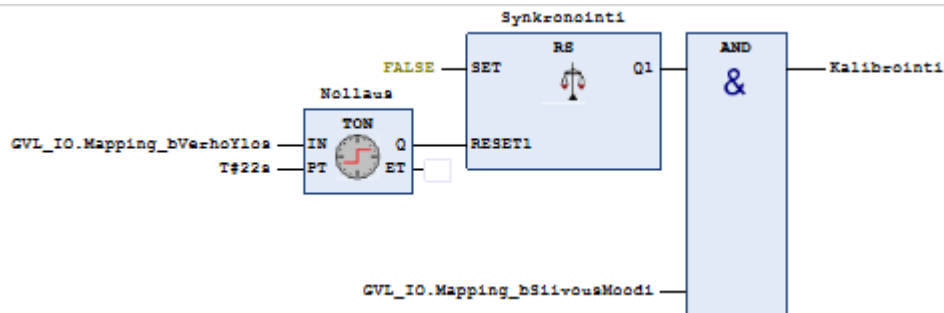
Lohkolla on myös omat tulot asetuksille esimerkiksi verhon ajoaika, sekä suunnan muutoksessa käytettävä viive, jolla voidaan pidentää verhomoottorin käyttöikää huomattavasti.

Lohkon lähdöissä on vain ja ainoastaan indikoinnit verhon alas / ylös liikkumiselle, näille muuttujille annettiin nimeksi GVL_IO_Mapping_bVerhoAlas ja GVL_IO_Mapping_bVerhoYlos, näiden lisäksi verhon varsinainen tilatieto, sekä lähtö, joka indikoi verhon kalibroinnin tilan.

4.2 Synkronointi lohko

Verhomoottorin ohjauslohkoon on valmiiksi koodattu itsensä referointi, joka synkronoi verhon tilatiedon aina kun lohko antaa moottorille komennon toiseen

ääripäähän. Tämä toiminto ei kuitenkaan takaa riittävän tarkkaa ohjausta verhomoottorille, testaamisen aikana havaittiin, että ohjaus voi mennä sekaisin ja ilman erillistä verhomoottorin synkronointia kokonainen päivä voi mennä ohjelmalta, ennen kuin toinen ääripää saavutetaan ja synkronointi tapahtuu.



Kuva 7. Verhomoottorin synkronointilohko

Ohjelmaan rakennettiin verhomoottorille erillinen synkronointilohko, jota käytetään bSafetyPosition- tulon kautta, eli synkronoinnin aikana muita toimintoja ei voi toteuttaa. TON- lohkon IN- tuloon määritettiin verhon ylös ajaminen ja PT- tuloon verhojen ajoaika. Näin varmistetaan tarkempi ja parempi ohjaus.

4.3 Antureiden skaalaus ja muuttaminen

Verhojen ohjauksessa käytettäville 2 anturille piti rakentaa samanlaiset skaalaukset kuin aikaisemmassakin ohjelmassa, että niiden lähettämät saatiin näkymään ohjelmassa oikeassa muodossa. Muuttujat GVL_IO_Mapping_rAurinkoAnturi ja GVL_IO_Mapping_rSisaLampotila_Anturi skaalataan oikein ja niistä saadaan muuttujat T_C_aurinko ja T_C_Sisa. Tämän jälkeen voidaan toteuttaa laskenta, jossa vähennetään sisälämpötila, auringonvalosta ja kerrotaan saatu luku 15, lopputuloksena saadaan aurinkoenergia watteina.

4.4 Lämmityksen ja jäähdytyksen avustus

Lämmityksen & jäähdytyksen tukemisen lohkon tarkoituksena on määrittää verholle parametrit, joiden mukaan se sään ja olosuhteiden mukaan tukee joko tilan jäähdyttämistä tai sen lämmittämistä. Lohko on aktiivinen ainoastaan silloin kun tila on tyhjä, eli lohkon tulo bPresence: = FALSE, ja ulkona on riittävä auringonpaiste. Myös tämä lohko vaatii toimiakseen ajastimen, joka resetoit tulon bEnable, että saadaan mahdollinen uusi käsky menemään lohkolle läpi. Lohko aktivoituu, kun kirkkaus ulkona ylittää 75 luxia ja pysyy yli tuon 20 sekunnin ajan ja vastaavasti se menee ei aktiiviseksi, kun valoisuus laskee 74 luxiin ja pysyy siellä 15 sekunnin ajan. Lämmityksen tukemisen tilanteessa verhot ajetaan 25 % auki ja jäähdytys tilanteessa verhot ajetaan 75 % auki.

4.5 Viihtyvyyden olosuhteet arkipäivä ja yö

Molemmissa olosuhte ohjauksissa käytetään FB_BARSunblindTwilightAutomatic- lohkoa, jolla saadaan toteutettua olosuhteohjaus, joka toimii, kun ulkona on riittävän valoisaa ja aikataulukyttimeen määritetty aika toteutuu. Lohkon tuloon bEnable kytketään FB_WeeklyTimeSwitch- lohko, joka määrittää milloin lohko aktivoituu. Päiväaikataulu on aktiivisena maanantaista perjantaihin 07:00-19:00 ja vastaavasti yöaikataulu on aktiivinen 19:00-07:00. Molemmissa tapauksissa valoisuusanturi liitetään tuloon uiBrightness. Molemmissa tapauksissa käytettiin myös apuna OR- ja NOT lohkoja, koska tämä lohko ei voi olla yhtä aikaa aktiivinen silloin kun lämmityksen- jäähdytyksentuki tai siivouskytkin on aktiivisena.

Päivä- tai yö olosuhteen ollessa aktiivinen lohko aktivoituu, kun valoisuus laskee alle 176 luxiin ja pysyy siellä yli 60 sekuntia. Kun valoisuus menee yli 176 ja pysyy siellä yli 20 sekuntia, lohko deaktivoituu. Kun päiväolosuhde on aktiivinen niin verho ajetaan 75 % ja yöllä vastaavasti verho ajetaan 50 %. Lohkon lähtö bActive kertoo molemmissa tapauksissa lohkon aktiivisuuden.

4.6 Viikonlopun aikainen ohjaus

Viikonlopun aikainen ohjaus toteutettiin lohkolla FB_BARSunblindEvent ja lohkon tuloon bActivate kytketään viikkoaikataulukytkin, joka on aktiivisena viikonloppuisin. Verholle annetaan viikonlopun aikana ohjauskäskyksi 100 % eli verho ajetaan täysin alas viikonlopun aikana.

4.7 Automaattiohjauksen prioriteettilohko

Jokaisella yllä mainituilla toimintalohkoilla on lähtö stSunblind, joka on string tyyppinen ja se pitää sisällään lohkon ohjaavat toiminnot. Toiminnoille pitää siis asettaa erillisesti prioriteetit, joiden mukaan lohko FB_BAR_SunblindPrioritySwitch lähettää verhomoottorinohjainlohkolle oikean käskyn prioriteettien mukaisesti. Lohkoon on määritetty toiminnot seuraavassa prioriteettijärjestyksessä:

1. Lämmityksen ja Jäähdytyksen avustus
2. Viihtyvyyden olosuhde päivä
3. Viihtyvyyden olosuhde yö
4. Viikonloppu

Tältä lohkolta saadaan lähdöstä stSunblind automaattiohjaus, joka on kytketty verhomoottorinohjainlohkoon tuloon bPosition. Manuaaliset käskyt ovat prioriteetiltaan korkeammat kuin automaattiohjauksen.

4.8 Kellonaika

Ohjelman aikatauluohjauksille tarvittiin saada järjestelmästä aika, FB_HVACGetSystemTime- lohkolla ei ole siis muuta toiminnallisuutta kuin ajan hakeminen, lohko on aina aktiivinen ja se hakee käytössä olevan järjestelmän ajan ja tämän jälkeen aikaa voidaan käyttää aikatauluohjauksissa.

4.9 Web- käyttöliittymä

Viimeisenä vaiheena ohjelmaan lisättiin webkäyttöliittymä (liite 3), jonka kautta on mahdollista ajaa verhoa ylös ja alas, sekä muuttaa lämmityksen, jäähdytyksen, päivätilan ja yö tilan verhon prosentuaalista asetusta.

Painonappien liikkumissuunta ilmaistaan web- käyttöliittymässä väreillä. Vihreä valo tarkoittaa verhon liikettä ylöspäin ja keltainen valo verhon liikettä alaspäin. Siivouskytkimelle ei määritetty nappia vaan pelkästään punainen indikointivalo, kun tila on aktiivinen. Verhon tilatiedolle laitettiin myös oma indikaattori, jotta verhon tilaa voi seurata käyttöliittymän kautta. Webkäyttöliittymälle ei tehty mahdollisuutta etäohjaukseen, vaan se toimii ainoastaan sisäisestä verkosta.

5 Pohdinta

Työn aiheena oli suunnitella ja toteuttaa verhomoottorinohjaus Beckhoff-järjestelmällä. Tavoitteena oli nykyaikaistaa vanhaa järjestelmää ja samalla saada hyötykäyttöön käyttämättömänä oleva demolaitteisto. Ohjelman sisällöllisenä tavoitteena oli saada säilytettyä ainakin samat toiminnallisuudet kuin vanhassa järjestelmässä. Ohjelma todettiin soveltuvaksi erinomaisesti opetuskäyttöön, ja pienellä jatkokehityksellä se voitaisiin siirtää mihin tahansa koulutilaan tai julkiseen tilaan ohjaamaan moottoroituja verhoja. Työ on kokonaisuutena erittäin opettavainen, ja se on hyödyllinen tulevaisuutta ajatellen.

Ohjelmoinnin katsotaan olevan jatkuvasti kasvava osa rakennettua ympäristöä, ja sen hallintaa pidetään työelämässä hyödyllisenä taitona. Projektin alkuvaiheessa jouduttiin käyttämään paljon aikaa Beckhoff:in laitteisiin ja ohjelmointiympäristöön perehtymiseen, ennen kuin tarvittavat asiat, kuten ohjelmointikirjastot, löydettiin. Tämä huomioon ottaen projektin aikatauluttaminen olisi pitänyt tehdä paremmin, jotta työ olisi saatu valmiiksi nopeammalla aikataululla.

5.1 Parannusideat ja jatkokehitys

Lämmitys & Jäähdytys Lohkon alkuperäinen käyttötarkoitus on yhdistää siihen ilmanvaihdon & jäähdytyksen toiminta. (raja-arvot jäähdytykselle & lämmitykselle), nämä voidaan myöhemmin muuttaa ohjelmaan täsmäämään, kun ohjelmalla halutaan toteuttaa varsinainen ohjaus Karelia ammattikorkeakoulun D-siivessä. Esimerkiksi samalla kun verhojen ohjaus viedään Microsoft Azure pilvipalveluun osaksi muuta rakennusautomaatiota.

Jotkin ohjelman lohkot olisi voinut valita paremmin tai toteuttaa ohjelman hieman eri tavalla, nyt esimerkiksi lämmityksen ja jäähdytyksen tukemiseen käytettyyn lohkoon piti tehdä sen tuloon bEnable nollaava toiminto, että se osaa lähettää oikean verhon tilatiedon verhomoottorinohjain lohkolle.

Kun verhomoottorinohjaus siirretään myöhemmin käyttöön Karelia-ammattikorkeakoulun D-siiven aulatilaan, laitteisto poikkeaa hieman opinnäytetyössä käytetystä demolaitteistosta. Tämän lisäksi ohjelmaan olisi hyvä lisätä toiminnallisuus, jolla saadaan verhot lukittua ala-asentoon.

Lähteet

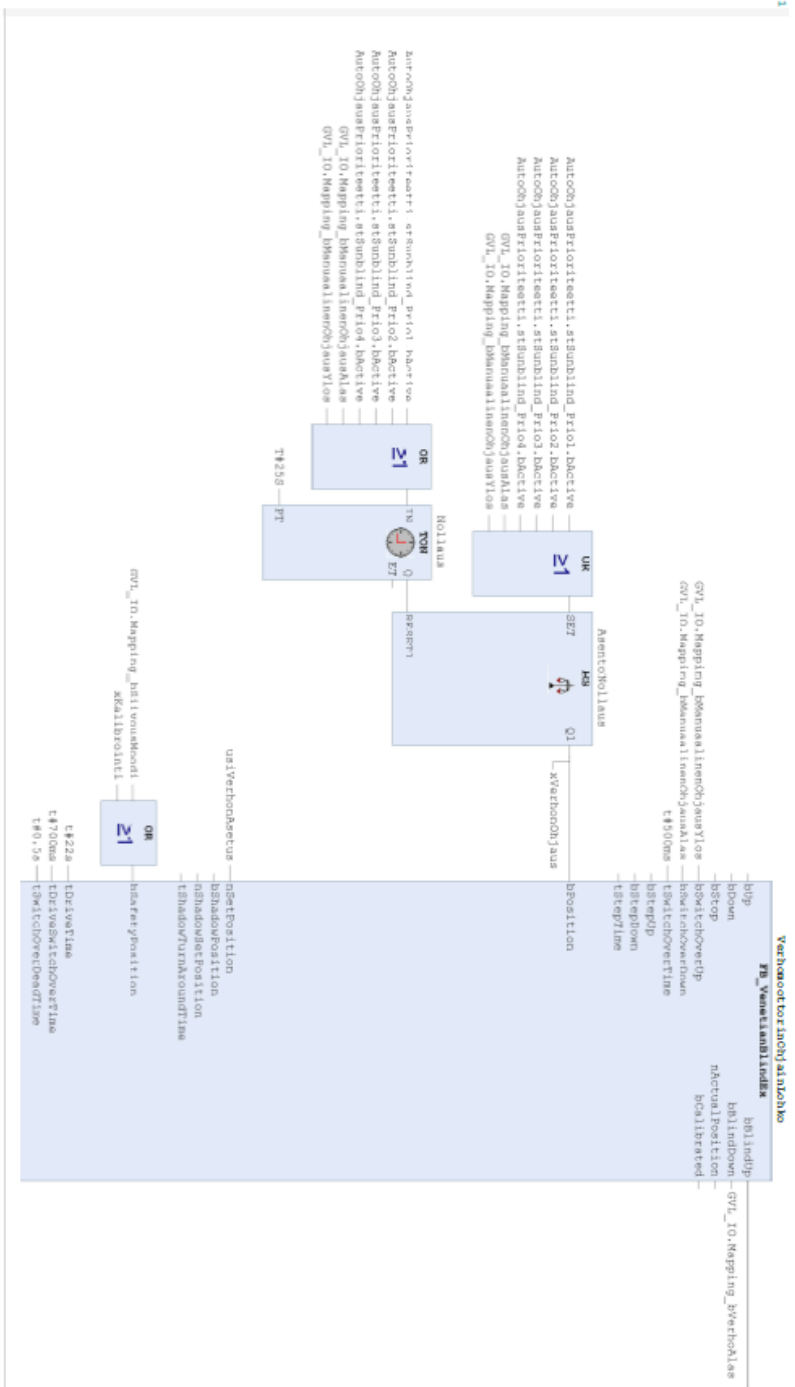
- Beckhoff Automation. 2025. Information System.
https://infosys.beckhoff.com/english.php?content=../content/1033/tcplcib_tc2_standard/74406539.html&id=4250055966004209897.
 10.11.2024.
- Control Engineering. 2007. Function block diagrams.
<https://www.controleng.com/function-block-diagrams/>. 23.11.2024
- Hursti, R. 2016. Miten Tietoturva liittyy rakennusautomaatioon?
<https://energistarakentamista.com/2016/05/25/rakennusautomaation-tietoturva/>. 19.7.2024.
- Kyberturvallisuuskeskus. 2021. Noin 1000 Automaatiolaitetta on yhä suojaamattomana suomalaisissa verkoissa.
<https://www.kyberturvallisuuskeskus.fi/fi/ajankohtaista/noin-1000-automaatiolaitetta-yha-suojaamatta-suomalaisissa-verkoissa>.
 19.7.2024.
- Kyberturvallisuuskeskus. 2020. Automaatiolaitetekartoitus.
https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/file/automaatiolaitetekartoitus_2020.pdf. 19.7.2024
- Meriläinen J. 2024. Rakennusautomaatio tutuksi – Mitä on rakennusautomaatio?
<https://www.buildwise.fi/rakennusautomaatio-tutuksi/>. 23.11.2024.
- Optimal Controls and Building Analytics Inc. 2025. What you need to know about building automation system.
<https://optimalcontrols.ai/blog/what-you-need-to-know-about-building-automation-system-bas>. 12.3.2025.
- Pärkkinen, P. Tarnanen, T. 2019. Auringonvalon ja -lämmönhallinta kiinteistössä. Karelia-ammattikorkeakoulu. Rakennus- ja yhdyskuntatekniikan koulutus. opinnäytetyö.
https://www.theseus.fi/bitstream/handle/10024/169789/Tarnanen_Tomi_Pärkinen_Petteri.pdf?sequence=2&isAllowed=y. 8.4.2025.
- Pärkkinen, P. Tarnanen, T. 2019. Verhomoottoriohjelman lähdekoodi.
- Roberts, B. 2019. What is building automation (temperature controls, energy management)? <https://www.mycontrolsolutions.com/building-automation-controls/>. 20.8.2024
- RT 07-11300. 2018. Aurinkosuojaus. Rakennustietosäätiö.
- SFS-EN 61131-3. 2013. Programmable controllers – Part 3: Programming languages. Suomen Standardisoimisliitto SFS.
- Somfy. Tekninenliite. 2024
- ST-Käsikirja 17. 2025. Rakennusautomaatiojärjestelmät. Sähkötieto ry.


```

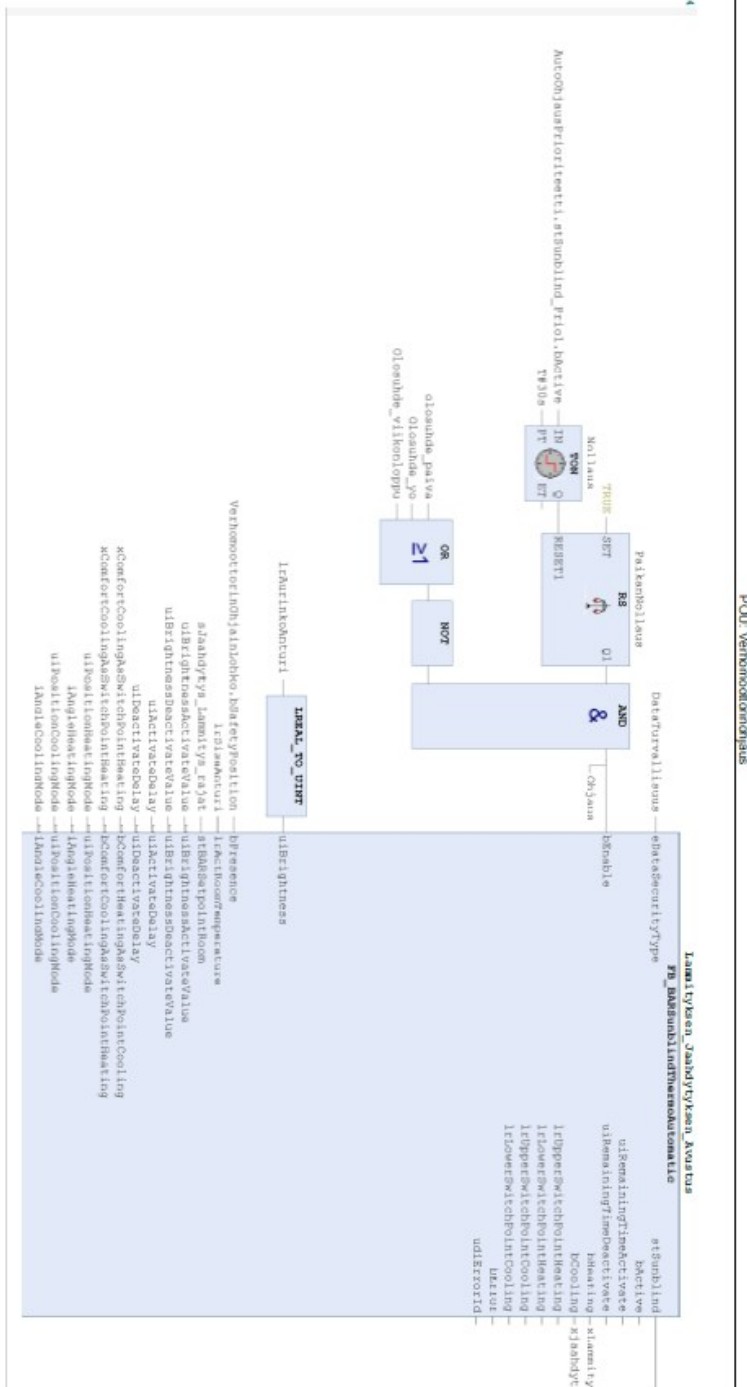
56 //Kõigi eelise lühikood :
57 ulfoos_11000001_lisipole :
58 //Kõigi eelise lühikood :
59 Dekarveerilissuur :
60
61 //Alkarsin lühikood 4 70
62 Palva_Alkarsin_u :
63 //Al_katavulu :
64 Vihni_yyyr_a_010_sade_Palva :
65 ulfoos_sade_010_sade_Palva :
66 ulfoos_sade_010_sade_Palva :
67 ulfoos_sade_010_sade_Palva :
68 ulfoos_sade_010_sade_Palva :
69 ulfoos_sade_010_sade_Palva :
70 ulfoos_sade_010_sade_Palva :
71 ulfoos_sade_010_sade_Palva :
72 ulfoos_sade_010_sade_Palva :
73 ulfoos_sade_010_sade_Palva :
74 ulfoos_sade_010_sade_Palva :
75 ulfoos_sade_010_sade_Palva :
76 ulfoos_sade_010_sade_Palva :
77 ulfoos_sade_010_sade_Palva :
78 ulfoos_sade_010_sade_Palva :
79 ulfoos_sade_010_sade_Palva :
80 ulfoos_sade_010_sade_Palva :
81 ulfoos_sade_010_sade_Palva :
82 ulfoos_sade_010_sade_Palva :
83 ulfoos_sade_010_sade_Palva :
84 ulfoos_sade_010_sade_Palva :
85
86 // Vilkonilgpn
87 koloo_010_sade_Palva :
88 ulfoos_sade_010_sade_Palva :
89 ulfoos_sade_010_sade_Palva :
90 ulfoos_sade_010_sade_Palva :
91 // Olgutuan Fzooltsedle
92 Autokõigutuan_010_sade_Palva :
93 Autokõigutuan_010_sade_Palva :
94 Villo_loppu :
95
96 // Palva_LightAutokõigutuan_010_sade_Palva :
97 // Palva_LightAutokõigutuan_010_sade_Palva :
98 // Palva_LightAutokõigutuan_010_sade_Palva :
99 // Palva_LightAutokõigutuan_010_sade_Palva :
100 // Palva_LightAutokõigutuan_010_sade_Palva :
101

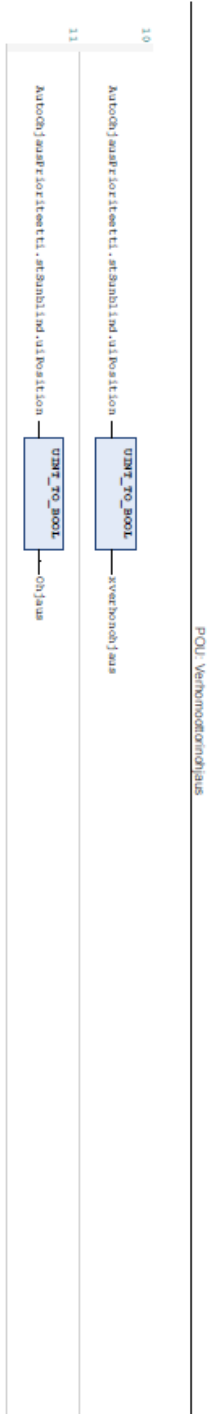
```

POU Verhoonotindus









Web-käyttöliittymä

