

Vili Tiinanen

HOTELLIHUONEEN HUONESÄÄDINSOVELLUS

HOTELLIHUONEEN HUONESÄÄDINSOVELLUS

Vili Tiinanen
Opinnäytetyö
Kevät 2018
Automaatiotekniikan tutkinto-ohjelma
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Automaatiotekniikan tutkinto-ohjelma

Tekijä: Vili Tiinanen
Opinnäytetyön nimi: Hotellihuoneen huonesäädinsovellus
Työn ohjaaja: Tero Hietanen
Työn valmistumislukukausi ja -vuosi: Kevät 2018
Sivumäärä: 41 + 6 liitettä

Tämän insinööriyön tarkoituksena oli luoda hotellihuoneille huonesäädinsovellus. Valmis sovellus vietiin käytäntöön ja sen tarkoitus on toimia n.150 hotellihuoneessa. Työ tehtiin projektinhoitajalle, joka vastaa hotellin rakennusautomaation muusta toteutuksesta. Työn toimeksiantajana on Fidelix Oy.

Työ tehtiin projektinhoitajan antamien suunnitelmien pohjalta. Hotellihuoneessa huonesäädin säättää hotellihuoneen lämpötilaa ohjaamalla puhallinpatterin lämmitys- ja jäähdytysventtiilejä sarjassa. Työn huonesäädin toimii myös osittain I/O:na hotellin valvonta-alakeskukselle. Työssä tehtiin huonesäätimeen käyttöliittymä niin hotellin asiakkaille kuin huoltomiehille.

Työssä tehtiin Modbus-liityntä Chillerin VariPro-ohjainkortille. Multi-24:n ja ohjainkortin kommunikoinnissa ilmeni virheitä ja niitä selvitettiin analysoimalla laitteiden välistä Modbus-liikennettä. Analysoinnin tuloksena laadittiin korjaus, jolla laitteiden välinen kommunikointi saatiin toimimaan.

Insinööriyön tuloksena saatiin valmis huonesäädinsovellus projektinhoitajan käyttöön. Työn aikana löydetty kommunikointivirhe saatettiin myös VariPro-ohjainkortin valmistuttajan tietoisuuteen.

Asiasanat: Fidelix Oy, Multi-24, rakennusautomaatio, säätimet, väylät -- atk, puhaltimet

ABSTRACT

Oulu University of Applied Sciences
Automation technology

Author: Vili Tiinanen

Title of thesis: Personalised HVAC-control in hotel room

Supervisor: Tero Hietanen

Term and year when the thesis was submitted: Spring 2018

Pages: 41 + 6 appendices

This thesis's purpose was to create room controller application for Multi-24 controller. Finished application was implemented in real world, running in 150 hotel rooms. The work covered only room controller part of the project. The rest of the building automation project was carried out by a project executive. The contractor for this job was Fidelix Oy

Work was made according to plans, provided by the project executive. The main process in a hotel room to be controlled, was room temperature. Room temperature control was controlled by combination of Multi-24 controller and a fan convector. Work also consisted of making graphics for the HMI (Human-Machine Interface) of the room controller, which will be operated by the hotel guests and servicemen.

A new modbus communication interface between Multi-24-controller and Chiller's VariPro Control card was made during this thesis. While making this interface, communication errors were encountered. Analyzing the communication between these devices provided the needed fix for the interface.

As a result, this thesis produced a room controller application for the project executive to utilize. The manufacturer of VariPro Control card was advised to check their cards Modbus-communication rules.

Keywords: Fidelix Oy, Multi-24, building automation, regulators, busses (computing), fans

SISÄLLYS

TIIVISTELMÄ	3
ABSTRACT	4
SISÄLLYS	5
SANASTO	6
1 JOHDANTO	7
2 RAKENNUSAUTOMAATIO	8
2.1 Rakennusautomaatiojärjestelmän rakenne	9
2.2 Huonesäädön järjestelmä	10
2.3 Työssä käytetty huonesäädinlaitteisto	12
3 MODBUS-KOMMUNIKAATIO	17
3.1 Modbus-kehysten rakenne	17
3.2 Modbus RTU -tiedonsiirto	18
3.3 Modbus RTU -kehys	18
3.4 Virheiden tarkastus	19
3.5 Modbus-tietomalli	20
3.6 Modbus-funktiokoodit	22
4 TYÖN SUORITUS	24
4.1 Pistetietokanta ja pisteiden luonti	24
4.2 Käyttöliittymä	25
4.3 VAK-liityntä	29
4.4 Sovellusohjelma	30
4.5 Multi-24 -aliväylä	33
4.6 Väylävirheitä	35
4.7 Tulosten tarkastelu	36
5 YHTEENVETO	38
LÄHTEET	39
LIITTEET	41

SANASTO

I/O	Automaatiojärjestelmän tai säätimen tulot ja lähdöt
IEC	International Electrotechnical Commission. Kansainvälinen sähkötekniikan standardointijärjestö.
Modbus	Avoin sarjaliikenneprotokolla
RTU	Modbus-protokollan binäärinen versio
VAK	Valvonta-alakeskus

1 JOHDANTO

Työssä tehtävä sovellus tehtiin Fidelix Oy:n hotelliprojektille. Projektia hoiti projektinhoitaja ja sovellus tehtiin hänen pyynnöstään. Sovellus ladattiin n.150 huonesäätimeen. Sovelluksella on päivittäin vaihtuvia monikansallisia loppukäyttäjiä.

Opinnäytetyön toimeksiantaja Fidelix Oy perustettiin vuonna 2002 kehittämään ja urakoimaan älykkäitä ja luotettavia rakennusautomaatio- sekä turvajärjestelmiä. Vuosien varrella se on kasvanut yhdeksi alan suurimmista toimijoista. (1.)

Rakennusautomaatiojärjestelmissä huonesäädön merkitys on kasvanut viime vuosina. Rakennuksen olosuhteita säädetään, jotta loppukäyttäjällä olisi mahdollisimman viihtyisät oltavat käyttämässään tiloissa. Hajautettu huonesäätö mahdollistaa yksittäisen tilan olosuhteiden säätämisen tilan käyttäjän mieltymysten mukaiseksi. Hotellihuone on hyvä esimerkki tilasta, jossa loppukäyttäjä haluaa vaikuttaa oleskelutilansa olosuhteisiin.

Väyläpohjaisella huonesäädinratkaisulla huonesäätimet saadaan liitettyä muuhun automaatiojärjestelmään. Väyläkommunikaation avulla ylemmältä automaatiotasolta huonesäätimen asetuksia voidaan muuttaa ja valvoa keskitetysti. Jos napin painalluksella alakeskukselta tai valvomosta kaikkien huonesäätimien asetusrvo voidaan yliajaa keskitetysti, on se huomattavasti tehokkaampaa kuin käydä muuttamassa jokainen paikallisesti.

Huonesäätimen käyttöliittymän ulkoasu määrittyy pitkälti loppukäyttäjistä. Kiinteistöhoitajille voidaan tehdä rujompi, mutta monipuolinen käyttöliittymä. Hotellin asiakkaille tulevassa käyttöliittymässä pitää painottaa yksinkertaisuuteen, helposti ymmärrettävyyteen ja mahdolliseen kielimuuriin.

Tehtäessä sovellusta huonesäätimelle on hyvä pitää mielessä, kuinka moni huonesäädin kyseistä sovellusta käyttää. Virheiden selvittäminen kentällä vie huomattavasti enemmän aikaa kuin testipenkissä konttorilla. Huolellinen testaus sovelluksen tekovaiheessa paljastaa sovellusvirheitä ja muita odottamattomia ristiiriitoja. Välttyminen 150 huonesäätimen sovelluksen päivittämiseltä säästää sekä rahaa että aikaa.

2 RAKENNUSAUTOMAATIO

Rakennusautomaatiolla tarkoitetaan rakennuksen erilaisten teknisten toimintojen itsestään tapahtuvaa ohjausta tavalla, jonka käyttäjä on ennalta määritellyt. Talotekniikan automaatiolla pyritään ohjaamaan ja valvomaan kiinteistön toimintaa siten, että saavutetaan mahdollisimman suuri energiatehokkuus, käyttömukavuus ja turvallisuus. (2, s. 9.)

Taloteknisiä järjestelmiä, joita automatiikalla voidaan ohjata, ovat esimerkiksi lämmitys-, valaistus-, valvonta-, hälytys- ja ilmanvaihtojärjestelmät. Laitteistot voivat olla vain yhden järjestelmän ohjaukseen rakennettuja tai useampien järjestelmien kokonaisvaltaiseen hallintaan kykeneviä laajempia keskitettyjä kokonaisuuksia. (3, s. 43.)

Rakennusten olosuhteita säädetään, jotta tiloja käyttävillä ihmisillä olisi mahdollisimman viihtyisä ja terveellinen ympäristö työskentelyyn, opiskeluun, harrastamiseen tai leppäämiseen, oikeastaan mihin tahansa aktiviteettiin tilassa kuin tilassa. Ympäristöministeriön määräyksiin ja ohjeisiin nojaten voidaan todeta, että laadukas sisäilmasto rakentuu lämpötilasta, ilmanlaadusta, ääniolosuhteista ja valaistuksesta. Ilmanvaihtojärjestelmä on myös osa viihtyisää sisäilmastoa ja osaltaan luo edellytykset sen ylläpitämiseen toiminnallaan. (4, s. 5–9.)

Laittevalmistajia ja järjestelmätoimittajia on markkinoilla useita, ja laitteistojen ominaisuudet ja ohjelmoitavuus poikkeavat toisistaan. Laitteistojen kyky kommunikoida keskenään on tärkeää, jotta toteutetut järjestelmät eivät olisi sidottuja vain yhden valmistajan tuotteisiin. Laitteiden yhteensopivuuden tarvetta palvelemaan on luotu erilaisia standardoituja väylätekniikoita ja langattomia tiedonsiirtotekniikoita, joiden avulla erilaiset laitteet pystyvät kommunikoimaan keskenään yhteisellä kielellä. (5, s. 251.)

2.1 Rakennusautomaatiojärjestelmän rakenne

Rakennusautomaatiojärjestelmän rakenne sisältää kolme päätasoa, jotka ovat hallintotaso, automaatiotaso ja kenttätaso. Hallintotasoon kuuluvat keskusvalvomot (kauko-/etävalvomot) ja paikallisvalvomot. Automaatiotasoon kuuluvat alakeskukset I/O-moduuleineen ja kenttätasoon kuuluvat kenttälaitteet (anturit ja toimilaitteet). (5, s. 93.)

Hallintataso

Hallintotaso toimii käyttäjärajapintana järjestelmään päin, joka tarkoittaa paikallisesti kohteessa olevia PC-valvomoita eli automatisoitavan kiinteistön sisällä tai kauko- tai etävalvomossa olevia PC-valvomoita, johon on keskitetty useamman kiinteistön valvonta. (5, s. 93.)

Valvomoon saadaan tiedot erilaisista hälytyksistä. Valvomosta voidaan tarkastella graafisia prosessikuvia sekä tarvittaessa tehdä muutoksia esimerkiksi lämpötilojen asetusarvoihin ja aikaohjelmiin. (5, s. 93.)

Automaatiotaso

Automaatiotason laitteistot muodostavat rakennusautomaatiojärjestelmän toiminnallisen rungon. Automaatiotasolla toimintaa ohjaavat keskusyksiköt, joissa toimivaan ohjelmistoon järjestelmän toiminnot on ohjelmoitu. Kenttätason laitteet välittävät automaatiotasolle tietoa järjestelmien tilasta, ja tämän tiedon perusteella järjestelmä toimii siihen ohjelmoitujen toimintatapojen mukaan ja ohjaa ja säätää kenttätason prosesseja toimilaitteiden avulla. Kenttätason ja automaatiotason välinen toiminta tapahtuu automaattisesti ilman käyttäjän ohjausta. (5, s. 94.)

Automaatiotason laitteita ovat myös kaikki omatoimiset säätömoduulit, esim. huonesäätimet, jotka sisältävät keskusyksikön (suoritin, keskusmuisti) ja kenttälaitteiden kytkentämoduulin samassa paketissa. Tällaiset säätömoduulit voidaan ohjelmoida toimimaan kokonaan itsenäisesti tai vuorovaikutuksessa keskusyksiköiden kanssa. (5, s. 94.)

Kenttätaso

Kenttätasolla rakennusautomaatiojärjestelmä liittyy taloteknisiin järjestelmiin erilaisten kenttälaitteiden esimerkiksi anturien ja toimilaitteiden avulla. Antureilla saadaan erilaisia mittaustietoja esimerkiksi lämpötiloista. Anturit voivat lisäksi kertoa järjestelmästä kyllä/ei-tyyppistä tietoa esimerkiksi siitä, onko jätevesikäivon pinnankorkeus ylittänyt kriittisen rajan. Toimilaitteilla taas ohjataan ja säädetään järjestelmän toimintaa joko portaattomasti tai auki/kiinni-tyyppisesti. (5, s. 95.)

2.2 Huonesäädön järjestelmä

Huonesäädössä automaatiojärjestelmä ohjaa yksittäisen tilan olosuhteita, kuten valaistusta, lämpötilaa tai ilmastointia. Sen käyttöliittymä voi olla toteutettu painikkeilla ja näytöllä tai kosketusnäytöllä. Huonesäätimet voivat toimia kokonaan itsenäisesti tai vuorovaikutuksessa keskusyksiköiden kanssa.

Markkinoilla olevat huonesäätimet voidaan jakaa karkeasti neljään eri ryhmään:

1. Väyläkommunikoiva ja ohjelmoitava
2. Väyläkommunikoiva ja ei-ohjelmoitava
3. Ei-väyläkommunikoiva ja ohjelmoitava
4. Ei-väyläkommunikoiva ja ei-ohjelmoitava.

Väyläkommunikoiva ja ohjelmoitava säädin

Tässä opinnäytetyössä käytetty Fidelixin Multi-24-kenttäohjaimen ja MultiDisplayn yhdistelmä on hyvä esimerkki väyläkommunikoivasta ja vapaasti ohjelmoitavasta huonesäätimestä.

Väyläkommunikoivan huonesäätimen etuna on, että se voidaan liittää muuhun rakennusautomaatiojärjestelmään. Tällöin huonesäätimen parametrejä, mittauksia, hälytyksiä ja säätöjä voidaan valvoa tai muuttaa joko hallintatasolta tai automaatiotasolta. Esimerkkinä hotellihuoneen käyttäjän asetteleman lämpötilan asetusarvon voi palauttaa valvomosta.

Ohjelmitavuus lisää huonesäätimen käyttömahdollisuuksia. Koska huonesäädin ei ole rajoittunut yhteen sovellukseen, sille voidaan esimerkiksi saneerauksessa ohjelmoida uusi käyttötarkoitus.

Vapaan ohjelmitavuuden lisäksi tämän tyyppiset huonesäätimet tarjoavat yleisesti hyvät säätimen sisään- ja ulostulo mahdollisuudet. Multi-24-kenttäohjaimelle voidaan liittää aliväylään kenttätason laitteita.

Väyläkommunikoiva ja ei-ohjelmitava säädin

Väyläkommunikoiva ja ei-ohjelmitava huonesäädin on rajoittunut vain yhteen sovellukseen, joka rajaa sen käyttömahdollisuuksia. Yleisesti yhden sovelluksen huonesäätimet sisältävät vakiodun I/O:n. Väyläkommunikointi mahdollistaa vuorovaikutuksen hallinta ja automaatiotasolle.

Markkinoilta löytyvä esimerkki tälle huonesäädin tyyppille on Fläkt Woods Oy:n STRA-24 (kuva 1). STRA-24 on valmiiksi ohjelmitu huonesäädin lämmityksen ja jäähdytyksen ohjaukseen jälkikäsitteilyjärjestelmissä. Se on valmiiksi ohjelmitu ja tiedonsiirrolla varustettu säädin, joka soveltuu toimistoihin, kouluihin, ostoskeskuksiin, lentoasemille, hotelleihin, sairaaloihin ja muihin vastaaviin tiloihin.
(6.)



KUVA 1. Fläkt Woods STRA-24 (6.)

Ei-väyläkommunikoiva ja ohjelmitava säädin

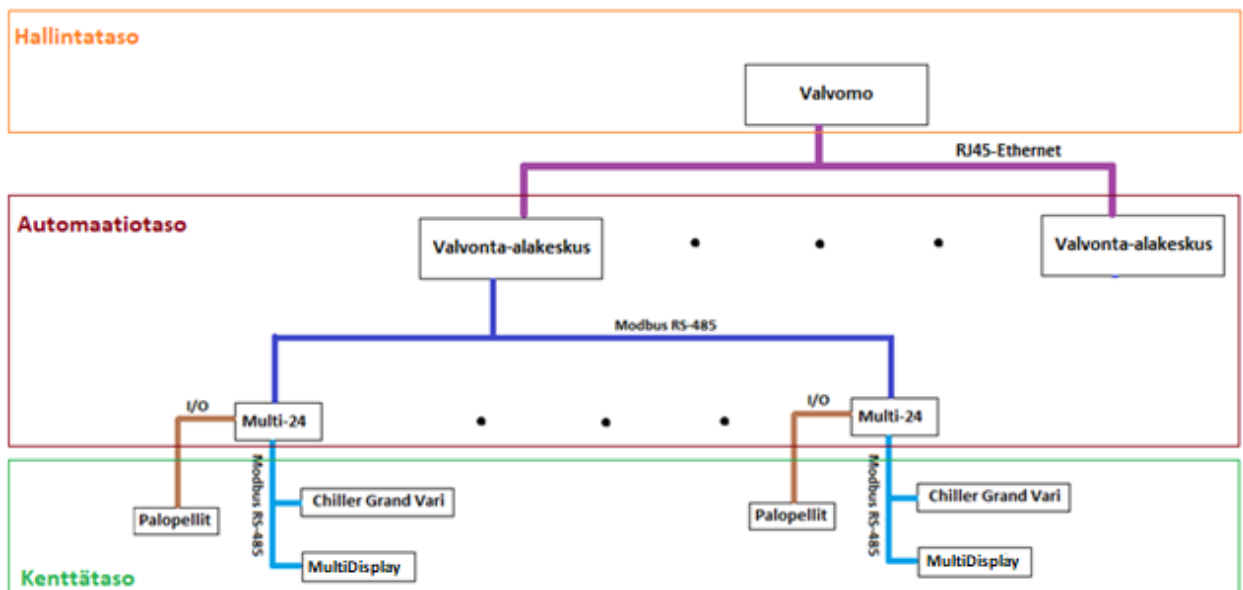
Ei väyläkommunikoivan ja ohjelmitavan huonesäätimen parametrejä ei voida tarkastella tai muuttaa kuin itse huonesäätimeltä paikallisesti. Hallintatason ohjaus ei ole mahdollista. Ohjelmitavuus lisää käyttömahdollisuuksia.

Ei-väyläkommunikoiva ja ei-ohjelmoitava säädin

Ei-väyläkommunikoivan ja ei-ohjelmoitavan huonesäätimien parametointi on työlästä, koska jokainen säädin on käytävä parametroimassa paikallisesti. Säätimen toimintaa ei voida seurata ylemmältä hallintatasolta ja ohjelmoinnin mahdollisuuden puute rajaa käyttömahdollisuuksia.

2.3 Työssä käytetty huonesäädinlaitteisto

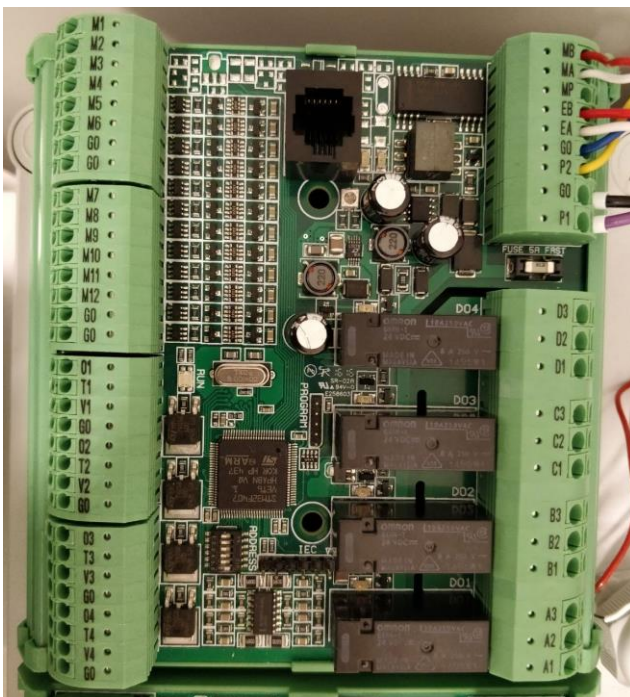
Projektinhoitajan antamien lähtötietojen pohjalta huonesäädinlaitteistoksi oli määriteltä Multi-24-kenttäohjain, MultiDisplay-kosketusnäyttö, Chiller Grand VariPro -puhallinkonvektori sekä kaksi palopeltiä. Multi-24-kenttäohjain koteloitiin hotellihuoneeseen, ja koteloon kaapeloitiin ylemmän tason väylä, jännitesyöttö sekä kentälaitteet. MultiDisplay-näyttö sijoitettiin hotellihuoneen seinälle ja kaapeloitiin Multi-24-kenttäohjaimelle. Puhallinkonvektori asennettiin huoneen katon sisään sekä kaapeloitiin Multi-24-kenttäohjaimelle. Palopellit liitettiin Multi-24:n sisään- ja ulostuloihin. Hierarkkisesti ylemmän väylän kautta Multi-24-säädin toimi osana modulaarista järjestelmää. Tällöin Multi-24-säädin toimi itsenäisesti, mutta vuorovaikutuksessa valvonta-alakeskuksen (myöhemmin VAK) kanssa (kuva 2). Hotellihuoneen kytkentäkuvassa on nähtävillä hotellihuoneen kaapeloinnin ohjeistus (liite 1).



KUVA 2. Periaatekuva järjestelmän väylärakenteesta

Multi-24-kenttäohjain

Säätimen pohjana on Fidelixin kehittämä ja valmistama, itsenäisesti toimiva ja vapaasti ohjelmoitava Multi-24-kenttäohjain (kuva 3). Multi-24 on suunniteltu pienempien automaatiokohteiden toteuttamiseen, kun kohteessa ei ole laajempaa automaatiojärjestelmää tai kohde on muuten järkevämpi toteuttaa omalla säädinlaitteella. Sitä käytetään esimerkiksi kaukolämpöjärjestelmien säätimenä, huonekohtaisen olosuhteidenhallinnan säätimenä ja pienempien ilmanvaihtokoneiden säätimenä. (7.)



KUVA 3. Multi-24-kenttäohjain

Multi-24 voi liittyä esimerkiksi automaatiojärjestelmään Modbus-protokollan mukaisella väyläliitynnällä, jolloin se toimii normaalisti yhtenä väylän laitteena. Tällöin se voi ottaa vastaan ja lähettää esimerkiksi mittaustietoja ja asetuksia automaatiojärjestelmän muiden laitteiden välillä. Lisäksi Multi-24 kykenee toimimaan isäntälaitteena omalle aliväylälleen, johon voidaan liittää väyläliitäntäisiä mittalaitteita tai toimilaitteita. Multi24:n yhteydessä usein käytetty MultiDisplay-kosketusnäyttö liitetään Multi24:ään Modbus-väylällä. KytKentäkuva Multi24:n liittimistä on nähtävissä liitteessä 1. Aliväylän liittimet ovat EA ja EB.

Multi-24-säätimessä on nimensä mukaisesti 24 kappaletta liityntäpisteitä eli niin sanottuja I/O-pisteitä. 12 kappaletta näistä pisteistä on UI-pisteitä (universal input). Näillä pisteillä pystytään vastaanottamaan kärkitietoja eli digitaalisia tuloja avoimella tai suljetulla kärjellä. Tämän lisäksi on mahdollista vastaanottaa pulssimuotoisia tietoja esimerkiksi vesimäärälaskurilta. Analogisia mittauksia voidaan tuoda pisteisiin resistiivisinä mittauksina tai standardimuotoisina jänniteviesteinä tai virtaviesteinä. Jänniteviestin alue on 0–10 V ja virtaviestin alue on 0–20 mA. (8.)

Liityntäpisteiden tyyppin valinta tapahtuu ohjelmallisesti. Ohjelmointivaiheessa luodaan yksi keskeytyssovellus. Tällä tarkoitetaan sovellusta, joka ajetaan ainoastaan tietyn signaalin tullessa. Tässä tapauksessa signaali on laitteiston käynnistys. Toisin sanoen ohjelmassa, joka ajetaan säätimen käynnistyessä, määritetään fyysisten pisteiden tyypit. Keskeytyssovelluksia voi olla määritetty vain yksi. (8.)

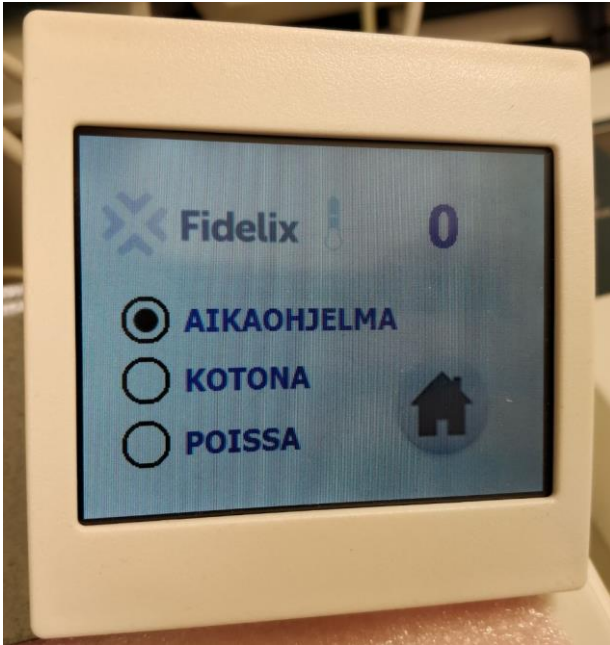
Säätimen ulostuloja on 12, kolmea eri tyyppiä. Siitä löytyy neljä digitaalista ulostuloa. Ulostulot on toteutettu sähkömagneettisilla releillä, joiden virran kesto on 6 ampeeria. Analogisia ulostuloja on kahdeksan. Standardimuotoista 0–10 V syöttäviä ulostuloja on neljä. Loput neljä ulostuloa on pulssinleveysmoduloituja 24 voltin vaihtovirtasyöttäjiä. Sääto on toteutettu triac-komponentein, joiden virran kesto on 1 ampeerin. (8.)

MultiDisplay-kosketusnäyttö

MultiDisplay on Fidelix'in kehittämä ja valmistama pieni kosketusnäyttö, jolla voidaan toteuttaa monipuolisia käyttöliittymiä kaikenlaisille sovelluksille (kuva 11). Näyttö liitetään sitä ohjaavaan laitteeseen Modbus-väylällä. Näyttöä käytetään yleisesti muun muassa huonesäätimien yhteydessä, jolloin näytön avulla käyttäjä voi muuttaa tilan olosuhteita kätevästi suoraan näytöltä ja seurata sen kautta muutosten vaikutusta, mittauksia ja historiatietoja. (9.)

Yleisesti Multi-24-säädin toimii Modbus-väylän isäntänä ja MultiDisplay väylän orjalaitteena. Tällöin MultiDisplay liitetään Multi-24:n aliväylään, kuten muut älyk-

käät väylälaitteet. MultiDisplayn laiteosoite asetetaan suoraan sen käyttöliittymästä. MultiDisplay tunnistaa automaattisesti väylänopeuden ja sen maksimiväylänopeus on 57 600 kbit/s. MultiDisplay voidaan asentaa joko pinta- tai uppoasenteeseen sähkörasiaan. (Kuva 4.) (9.)



KUVA 4. MultiDisplay-kosketusnäyttö

MultiDisplay sisältää 256 kilotavun kiinteän muistin sekä muistikorttipaikan, jonka avulla muistia voi laajentaa mikro-SD-kortilla aina 4 gigatavuun asti. Käyttöliittymän grafiikat ja pisteet tallennetaan näytön muistiin ja näyttöä ohjaavalla laitteella oleva sovellus päivittää pisteiden tietoja väylän kautta.

Chiller Grand VariPro -puhallinpatteri

Hotellihuoneisiin oli määritelty Chiller Oy:n valmistamat puhallinpatterit. Mallina oli erityisesti hotellihuoneisiin, toimistoihin ja vaativiin pientalojen ilmastoitiin sopiva Grand VariPro (kuva 5). Grand Vari puhallinpatterit valmistetaan ja testataan Suomessa (10.).



KUVA 5. Chiller Grand VariPro -puhallinpatteri

Puhallinpatteri säättää huoneen lämpötilaa kierrättämällä huoneilmaa patterin läpi. Patteriin päästetään joko lämmintä tai viileää vettä sen mukaan, halutaanko ilmaa lämmittää vai viilentää. Puhallinpatterin säleikkö on jaettu kahteen osaan: yläosasta ilma puhalletaan huoneeseen ja alaosasta imetään laitteeseen. Raikisilma tuodaan saman puhallussäleikön kautta huoneeseen.

Grand VariPro -puhallinpatterissa oli asennettuna VariPro-ohjainkortti (liite 2), joka mahdollisti puhallinpatterin väyläpohjaisen ohjauksen. Sovelluksen testaamiseksi tilattiin yksi puhallinpatteri Fidelixin Oulun konttorille. Tällöin väyläliityntän toimivuus pystyttiin varmentamaan sovelluksessa jo ennen, kuin sovellus ladataisiin kohteessa oleviin yli 150 huonesäätimeen. Liitteessä 2 on nähtävillä VariPro-ohjainkortin asennusohje, jonka perusteella puhallinpatteri kytkettiin Multi24:n aliväylään. VariPro-ohjainkortin liitin D+ vastasi Multi-24:n aliväylän liittintä EA. VariPro-ohjainkortin liitin D- vastasi Multi-24:n liittintä EB. Multi-24:n aliväylä liittimet näkyvät liitteessä 1 ja VariPro-ohjainkortin väyläliittimet näkyvät liitteessä 2.

3 MODBUS-KOMMUNIKAATIO

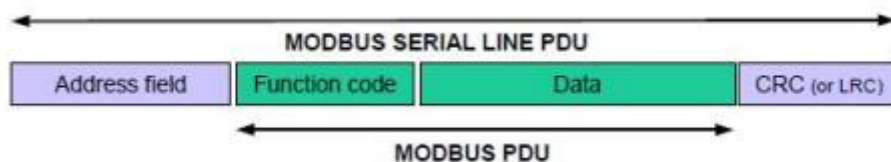
Modbus on Modiconin luoma sarjaliikenneprotokolla, joka alun perin oli tarkoitettu käytettäväksi Modiconin ohjelmoitavien logiikoiden väliseen tiedonsiirtoon. Alun perin Modbus kehitettiin RS-232 (RTU ja ASCII) -sarjaliikenteelle, myöhemmin väylätekniikoiden kehittyessä on mukaan tullut myös RS-485 ja Ethernet (TCP-protokolla). (11.)

Modbus on isäntä-orja- eli Master-Slave-tyyppinen kommunikointiprotokolla, jossa on väylällä yksi isäntä, joka suorittaa kyselyitä tai kirjoittaa dataa tiettyyn orjaan, joiden määrä voi olla jopa 247 yhdessä Modbus-väylässä. Jokaisella orjalla on yksilöllinen ID-osoite tai -numero, jonka avulla isäntä kommunikoi orjan kanssa. Orja lähettää dataa vain isännän pyynnöstä. (11.)

Modbus on lisenssivapaa, joten se on laitevalmistajille vapaasti integroitava kommunikointiprotokolla. Maksuton Modbusin käyttö on tehnyt Modbusista suosittua ja yleisesti käytettyä laiteliityntäprotokollan teollisten elektroniikkalaittevalmistajien keskuudessa. Modbus on muodostunut eräänlaiseksi laitevalmistajien yleisprotokollaksi. (11.)

3.1 Modbus-kehiksen rakenne

Modbus-protokollan data on yksinkertainen kehys (Modbus PDU), joka on riippumaton muista kerroksista. Kehiksen koko rakenne riippuu käytettävästä väylästä tai verkosta. Kuvassa 6 on esitettyä Modbus PDU ja sarjaväyläprotokollan kehys kaikkine kenttineen. (12, s. 8.)



KUVA 6. Modbus RTU -sarjaväylän PDU-kehys (12, s. 8.)

Osoitekenttä sisältää orjan Modbus-osoitteen. Isäntä asettaa osoitekenttään orjan osoitteen, jonka kanssa se haluaa aloittaa keskustelun. Orja vastatessaan

asettaa osoitekenttään oman osoitteensa, jotta isäntä tietää, keneltä vastaus tulee. Funktiokoodin tehtävä on kertoa orjalle tehtävä, joka sen pitää suorittaa. Viimeisenä on virheentarkastuskenttä. Käytettävä virheentarkastustapa riippuu käytettävästä tiedonsiirtotilasta (RTU tai ASCII). (12, s. 8.)

3.2 Modbus RTU -tiedonsiirto

Modbus-sarjaväyläprotokollan "tietokerros" käsittää kaksi erillistä kerrosta: isäntä-orja-protokollan ja tiedonsiirtotilan (RTU tai ASCII). (12, s. 9.)

Valittu tiedonsiirtotila määrittää viestien kehysten sisällön, tiedon pakkauksen ja dekodauksen. Valittu tiedonsiirtotila (RTU tai ASCII) täytyy olla sarjaväylässä sama kaikissa laitteissa. Modbus-standardin mukaan RTU-tila on pakollinen kaikille sarjaliikenteen laitteille ja ASCII-tiedonsiirto valittavissa optiona. (12, s. 12.)

RTU-tiedonsiirrossa kukin 8-bittinen tavu viestissä sisältää kaksi 4 bitin heksadesimaalimerkkiä. Etuna tässä on, että suurempi merkkitiheys mahdollistaa suuremman tiedonsiirtonopeuden verrattuna ASCII-tilaan, jossa lähetetään tavussa kaksi ASCII-merkkiä. (12, s. 12.)

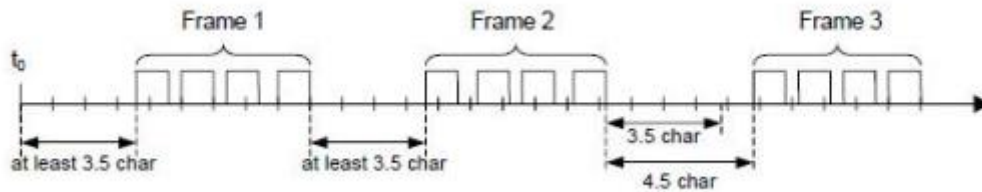
RTU-protokollan jokainen tavu sisältää 1 aloitusbitin, 8 databittiä (vähiten merkitsevin ensin), 1 pariteettibitin ja 1 pysäytysbitin (12, s. 12).

Modbus RTU-protokollan mukaan parillinen pariteetti-tila vaaditaan ja muita tiloja (pariton pariteetti ja ei pariteettia) voidaan myös käyttää. On suositeltavaa, että laitteilla on mahdollista valita myös ei pariteetti-tila, jotta varmistetaan yhteensopivuus muiden Modbus-laitteiden kanssa. Käytettäessä ei pariteetti-tilaa on käytettävä kahta pysäytysbittiä. (12, s. 12.)

3.3 Modbus RTU -kehys

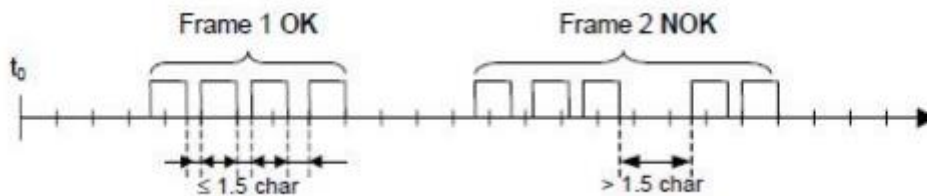
Modbus-viesti on sijoitettu kehykseen, josta tunnetaan alku- ja loppukohtat. Tämä kertoo laitteille, milloin viesti on tullut ja milloin se on loppunut. Lisäksi se mahdollistaa osittaisten virheellisten viestien tunnistamisen. (12, s. 13.)

RTU-tilassa viestikehykset erotetaan vähintään 3,5 merkkikerran hiljaisella jaksolla. Kuvassa 7 on esitettyinä viestikehykset ja niiden välit.



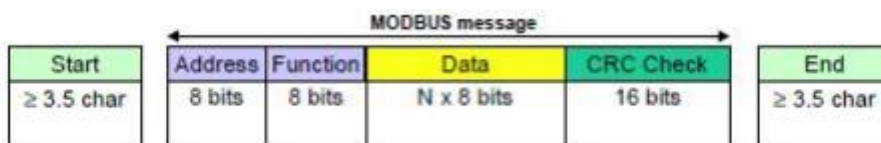
KUVA 7. RTU-viestin merkkikehysten hiljainen väli (12, s. 13.)

Merkkikehys lähetetään jatkuvana virtana merkkejä. Jos merkkikehysten hiljaisen välin merkkien väli on yli 1,5 merkkikertaa, viestikehys hävitetään vastaanottimella. Tämä on esitettyä kuvassa 8. (12, s. 13.)



KUVA 8. Merkkikehysten hiljainen väli (12, s. 13.)

Kuvassa 9 on esitettyä RTU-viestin merkkikehys. Merkkikehys voi olla enintään 256 tavua pitkä. Osoitekenttä ja toimintakoodi ottavat molemmat yhden 8-bittisen tavun. Data-kenttä voi olla maksimissaan 252 8-bittistä tavua ja CRC-tarkistus ottaa kaksi 8-bittistä tavua. (12, s. 13.)



KUVA 9. RTU-viestin merkkikehys (12, s. 13.)

3.4 Virheiden tarkastus

Turvallisuus Modbus-sarjaväylässä perustuu kahdenlaiseen virheentarkasteluun, pariteettitarkastukseen ja kehyksen tarkastukseen (CRC ja LRC). Merkkien tarkkailu ja viestin kehyksen tarkkailu aikaansaadaan sekä isännässä että orjassa. (12, s. 19.)

Isäntä odottaa orjan vastausta viestin aikajakson verran. Jos orja huomaa viestintarkastuksessa virheen, jättää se vastaamatta isännälle. Tällöin syntyy aikakatkaisu isännässä ja syntyy virhe. (12, s. 19.)

Pariteettitarkastus tarkkailee viestin 1-bittien määrää. Jos parillisessa tarkastuksessa ilmenee pariton määrä 1-bittejä, syntyy virhe. Parittomassa tarkastuksessa taas parillisista määristä syntyy virhe. Pariteettitarkastuksen huono puoli on, että se havaitsee virheen vain, jos viestissä on virheellinen määrä bittejä. Kehyksen tarkkailun tyyppi riippuu siitä, onko käytössä Modbus RTU- vai ASCII-tiedonsiirto. RTU-tiedonsiirrossa virheentarkkailu perustuu CRC-menetelmään. ASCII-tiedonsiirrossa virheentarkkailu perustuu LRC-menetelmään. (12, s. 19.)

3.5 Modbus-tietomalli

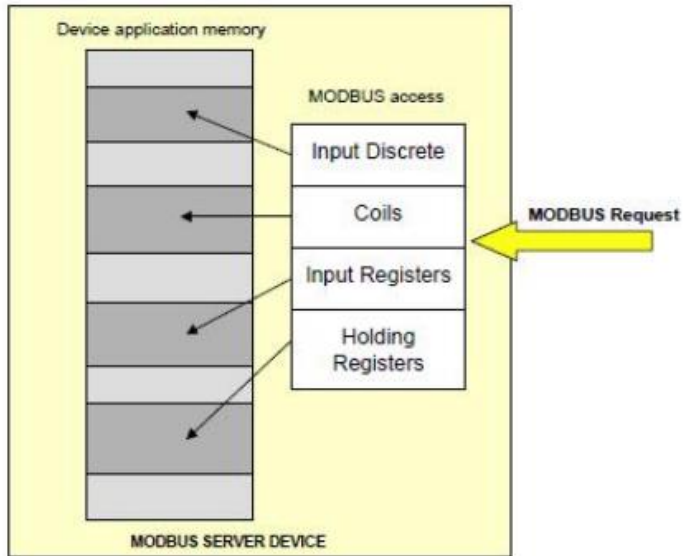
Modbus-tietomalli perustuu taulukoihin. Neljä päätaulukkoa on esitettyinä kuvassa 10.

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

KUVA 10. Modbus päätaulukot (13, s. 6.)

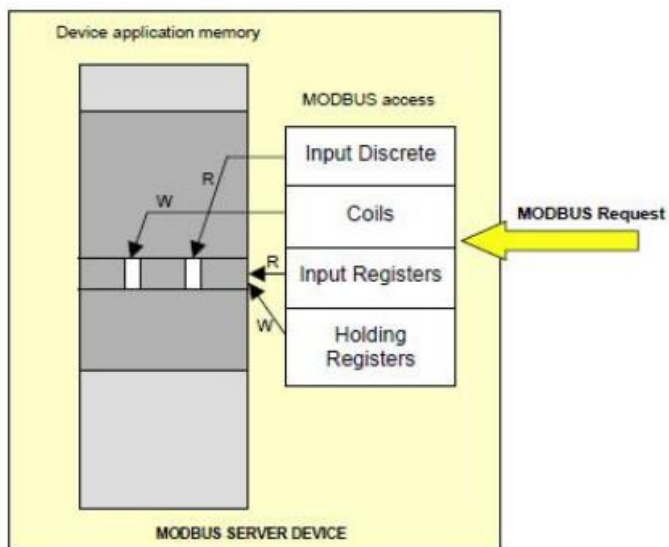
Discretes input eli digitaalisääntulo on yhden bitin mittainen tieto, jota voidaan vain lukea. Coils eli digitaaliulostulot ovat yhden bitin mittainen tieto, jota voidaan lukea ja kirjoittaa. Input register eli analogiasääntulo on kahden tavun mittainen 16-bittinen tieto, jota voidaan lukea. Se voi olla myös neljän tavun mittainen 32-bittinen tieto. Holding register eli analogiaulostulo on kahden tavun mittainen 16-bittinen tieto, jota voidaan lukea ja kirjoittaa. Se voi olla myös neljän tavun mittainen 32-bittinen tieto. On sallittua pitää kaikki neljä taulukkoa päällekkäin. kuvassa 6 ja 7 on esitettyinä kaksi eri tapaa järjestää tieto Modbus-laitteeseen. (13, s. 6.)

Kuvassa 11 jokainen lohko on erillinen, koska tieto ei ole yhteensopiva keskenään. Jokaista lohkoa luetaan omalla funktiolla. (13, s. 6.)



KUVA 11. Modbus-tieto erillisinä lohkoina (13, s. 6.)

Kuvassa 12 on vain yksi lohko. Saman tiedon luku voi olla mahdollista eri funktioiden avulla käyttäen 16-bittistä hakua tai bittinumeroa. Tiedon lukutapa riippuu siitä, luettaanko binääri- vai analogiatietoja. (13, s. 7.)



KUVA 12. Modbus-tieto yhdessä lohkoissa (13, s. 7.)

3.6 Modbus-funktiokoodit

Halutun Modbus-tiedon luku tapahtuu orjista funktiokooodeilla. Ne ovat etumerkitömiä kokonaislukuja ja alkavat nolasta. Funktiokoodit on jaettu kolmeen ryhmään: julkisiin, käyttäjän muokattaviin ja varattuihin funktiokooodeihin. Näistä julkiset koodit ovat ennakkoon määriteltäviä, julkisesti dokumentoituja, modbus.orgin hyväksymiä ja vaatimustenmukaisuus testien läpäisemiä. (13, s.10)

Taulukossa 1 on esitelty yleisesti käytettyjä julkisia funktiokooodeja. Julkisia funktiokooodeja on vielä lisää, mutta tässä työssä esitellään vain muutama yleisemmin käytetty.

TAULUKKO 1. Yleisesti käytetyt julkiset Modbus-funktiokoodit.

Funktiokoodi	Toiminta	Datapisteen tyyppi	Osoite
01 (01 hex)	Luku	Digitaaliulostulo	0xxxx
05 (04 hex)	Kirjoita yksi	Digitaaliulostulo	0xxxx
15 (0F hex)	Kirjoita useampi	Digitaaliulostulo	0xxxx
02 (02 hex)	Luku	Digitaalisääntulo	1xxxx
04 (04 hex)	Luku	Analogiasääntulo	3xxxx
03 (03 hex)	Luku	Analogiaulostulo	4xxxx
06 (06 hex)	Kirjoita yksi	Analogiaulostulo	4xxxx
16 (10 hex)	Kirjoita useampi	Analogiaulostulo	4xxxx

Input registers eli analogiasääntulojen luku tapahtuu funktiokoodilla 04. Analogiasääntulojen osoite alkaa 3:lla. (14, s.14)

Holding registers eli analogiaulostulojen luku tapahtuu funktiokoodilla 03. Yksittäisen rekisterin kirjoitus tapahtuu funktiokoodilla 06 ja useamman rekisterin yhtäaikainen kirjoitus koodilla 16. Analogiaulostulojen osoite alkaa 4:llä. (14, s. 14.)

Discrete inputs eli digitaalisääntulojen luku tapahtuu funktiokoodilla 02. Digitaalisääntulojen osoite alkaa 1:llä. (14, s. 14.)

Coils eli digitaaliulostulojen luku tapahtuu funktiokoodilla 01. Yksittäisen rekisterin kirjoitus tapahtuu funktiokoodilla 05 ja useamman rekisterin yhtäaikainen kirjoitus koodilla 15. Digitaaliulostulojen osoite alkaa 0:lla. (14, s. 15.)

Analogiatiedot voivat olla signed integer, unsigned integer tai float lukuja. Etumerkillisillä luvuilla (Signed integer) saadaan muodostettua 2 tavulla eli 16 bitillä luvut -32 768–32 767. Etumerkittömillä luvuilla (Unsigned integer) saadaan muodostettua 2 tavulla eli 16 bitillä luvut 0–65 535. Liukuluvuilla (FLOAT) voidaan muodostaa luvut $-3,4 \cdot 10^{38}$ – $3,4 \cdot 10^{38}$. (14, s. 15.)

4 TYÖN SUORITUS

Säätimen ohjelmointia aloitettiin suunnittelemaan projektinhoitajan antamien lähtötietojen pohjalta. Lähtötiedoissa mainittiin sovellukselta vaadittu toiminnallisuus. Lopputuloksena saatiin sovellukselle toimintakuvauksen mukainen pääohjelma, VAK-liityntä, MultiDisplay-grafiikka ja Chiller VariPro-liityntä. Sovelluksen kokonaista lähdekoodia tai lähtötietoja ei julkaista kokonaisuudessaan tässä opinnäytetyössä, vaan niistä esitellään vain tarpeelliset osat.

Liitteessä 6 on esitelty periaatteellinen luonnos huonesäädinsovelluksen teosta. Luonnos antaa hyvän kokonaiskuvan sovelluksen luomisen eri osa-alueista. Seuraavissa luvuissa esitellään sovelluksenteon eri vaiheita syvällisemmin.

4.1 Pistetietokanta ja pisteiden luonti

Ennen itse sovellusohjelman luomista, piti määrittellä ohjelman käyttämät muuttujat eli pisteet. Muuttuja voi sisältää esimerkiksi mittauksen tai asetusarvon. Pistesuunnittelu tehtiin Fidelix Oy:n MultiPointTool-työkalulla. Pistetietokannan käsittely, siihen tallentaminen ja tietokannasta luku ovat aiemmin toimineet osana IEC-ohjelmaa, mutta ovat nyt integroidut osaksi Multi-24:n laiteohjelmistoa. Tämä nopeuttaa Multi-24 sovelluksen valmistumista ja on selkeämpi käyttää.

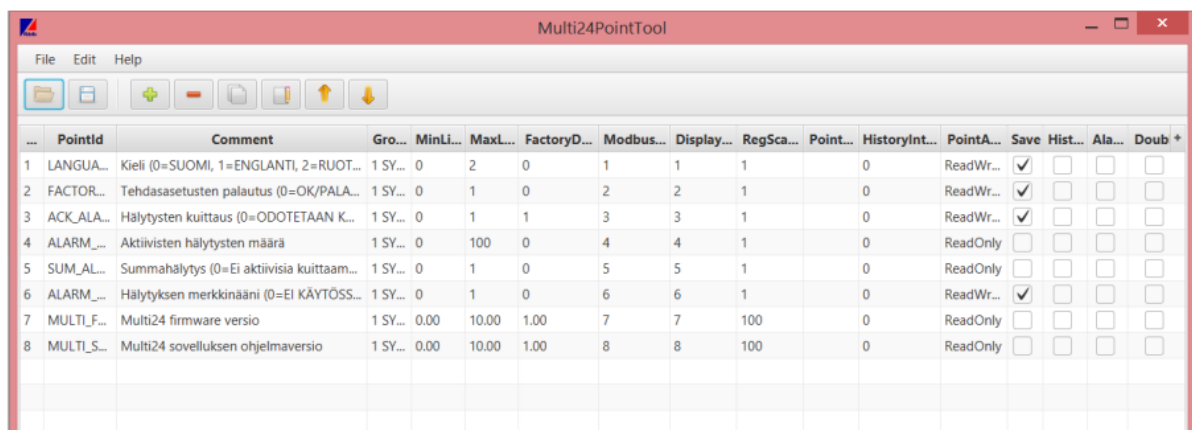
MultiPointTool-ohjelmalla pisteiden luonti tapahtuu lisäämällä pisteitä taulukkoon (kuva 13). Taulukkoon pisteelle määritetään

- pistetunnus (PointId),
- selitysteksti (Comment),
- minimi- ja maksimiarvot (MinLimit, MaxLimit),
- tehdasasetusarvo (FactoryDefault),
- Modbus-rekisterinumero (ModbusRegister),
- mahdollinen näytön käyttämä rekisterinumero (DisplayRegister),
- pisteen skaalauskerroin (RegScaling),
- pisteen yksikkö (PointUnit),
- pisteen historiaan tallennuksen aikaväli (HistoryInterval) ja
- pisteen lukusuunta (PointAccess).

Pisteelle annetaan tietyt "Flag"-muuttujat. Nämä muuttujat määrittävät pisteelle eri ominaisuuksia.

- "Save"-muuttuja määrittää, tallennetaanko pisteen arvo katoamattomaan muistiin.
- "History" -muuttujalla määritetään historiaan tallennus.
- "Alarm"-muuttuja otetaan käyttöön, kun pistettä käytetään hälytyspisteinä.
- "Double"-muuttujaa käytetään, kun pisteen arvo vaatii kahden rekisterin alueen.

Työkalu nopeuttaa pisteiden luomista ja helpottaa niiden hallintaa, sillä työkalu mahdollistaa myös pisteiden massamuokkauksen. Lopuksi, kun pisteet on luotu, ne muunnetaan Multi24:n IEC-ohjelmaan sopivaan muotoon Database.poe -tiedostoksi. IEC-ohjelma käyttää kyseistä tiedostoa pistetietokantanaan, johon kaikkien mittausten luennat tallennetaan ja josta luetaan ohjelmaan pisteiden arvot (kuva 13).



...	Pointid	Comment	Gro...	MinLi...	MaxL...	FactoryD...	Modbus...	Display...	RegSca...	Point...	HistoryInt...	PointA...	Save	Hist...	Ala...	Doub +
1	LANGUA...	Kieli (0=SUOMI, 1=ENGLANTI, 2=RUOT...	1 SY...	0	2	0	1	1	1		0	ReadWr...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	FACTOR...	Tehdasasetusten palautus (0=OK/PALA...	1 SY...	0	1	0	2	2	1		0	ReadWr...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	ACK_ALA...	Hälytysten kuittaus (0=ODOTETAAN K...	1 SY...	0	1	1	3	3	1		0	ReadWr...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	ALARM_...	Aktiivisten hälytysten määrä	1 SY...	0	100	0	4	4	1		0	ReadOnly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	SUM_AL...	Summahälytys (0=Ei aktiivisia kuittaam...	1 SY...	0	1	0	5	5	1		0	ReadOnly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	ALARM_...	Hälytyksen merkinääni (0=EI KÄYTÖSS...	1 SY...	0	1	0	6	6	1		0	ReadWr...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	MULTI_F...	Multi24 firmware versio	1 SY...	0.00	10.00	1.00	7	7	100		0	ReadOnly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	MULTI_S...	Multi24 sovelluksen ohjelmaversio	1 SY...	0.00	10.00	1.00	8	8	100		0	ReadOnly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

KUVA 13. Fidelix Oy:n MultiPointTool-ohjelma

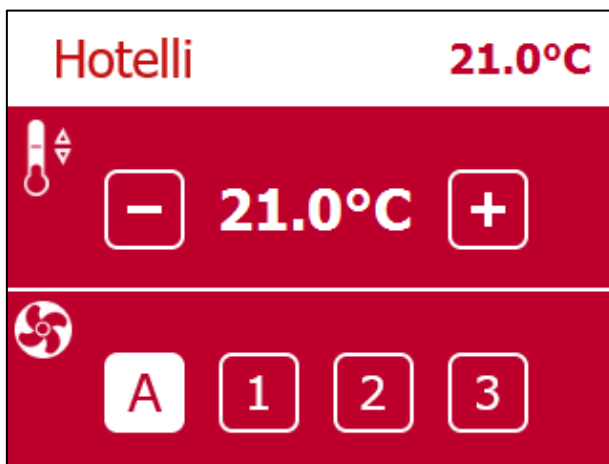
4.2 Käyttöliittymä

Projektinohitaja kertoi esitelleensä huonenäyttöjen malligrafiikkaa tilaajalle. Huonenäyttöjen grafiikan piirtoa lähdettiin toteuttamaan malligrafiikan kaltaiseksi. Grafiikalta tuli löytyä lämpötilan asetusarvo, puhallinsäädöt, hotellin logo ja erilli-

nen huollon asetussivu. Grafiikan kielenä oli englanti. Symbolien käyttö maksimoi, jotta hotellin kansainvälisillä asiakkailla ei tulisi vaikeuksia operoida käytölliittymää.

Grafiikan piirto MultiDisplay-näytölle tehtiin Fidelixin HTMLeditor-sovelluksella. MultiDisplayn grafiikan pistetunnukset voitiin kopioida MultiPointTool-sovelluksesta, ja liittää ne HTMLeditorissa oikeiden symbolien kohdalle. HTMLeditorin kuvat muunnettiin convertterilla MultiDisplaylle sopivaksi.

Pääsivulle piirrettiin hotellin asukkaan käytölliittymä, joten sen tuli olla mahdollisimman yksinkertainen ja helppokäyttöinen. Pääsivu sisälsi asetusarvon asettamisen, puhallinnopeuden asettamisen, linkin huoltoasetuksiin (piilotettu asiakkaalta) sekä huonelämpötilan (kuva 14). Huoltoasetusten linkki piilotettiin laitteen yläkulmaan, jotta hotellin asukas ei vahingossa painaisi siihen. Linkkiin osuessa näyttö kysyy huoltosalasanaa ja oikealla salasanalla päästään huoltoasetuksiin.



KUVA 14. Hotellihuoneen MultiDisplayn pääsivu, hotellin logo vaihdettu "Hotelli"-tekstiin

Huonelämpötilan asetusarvo laskettiin antamalla asetusarvolle perustaso. Perustaso voitiin poikkeuttaa +/-10 pykälää, jossa pykälä 0 vastasi perustaso. Asetusarvolle annettiin celsiusrajat, kuinka paljon perustasosta voitiin poikkeuttaa. Jos poikkeutusraja oli +/- 2 °C ja perustaso 21 °C sekä poikkeutus +1 pykälä, niin laskettu asetusarvo oli 21,2 °C. Jokainen poikkeutuksen pykälä siis vastaisi 0,2 °C:ta.

Pääsivulta asukas pystyi muuttamaan perusasetuksen poikkeutusta. Lasketun asetusarvon laskenta tehtiin ensin Multi-24:n toimintalohkolle, mutta sen huomattiin päivittyvän liian suurella viiveellä MultiDisplay-näytölle. Ongelman ratkaisemiseksi käytettiin MultiDisplay-näytöllä tapahtuvaa laskentaa. Sen avulla lasketun asetusarvon päivitys näytölle saatiin toimimaan nopeasti.

Puhaltimen nopeuden asetus tehtiin tiloilla automaatti, yksi, kaksi ja kolme. Automaattitilassa puhaltimen nopeus määräytyi huonelämpötilan ja asetusarvon erosuureen mukaan. Asetusten yksi, kaksi ja kolme puhallinnopeudet olivat staattisia prosenttilukuja.

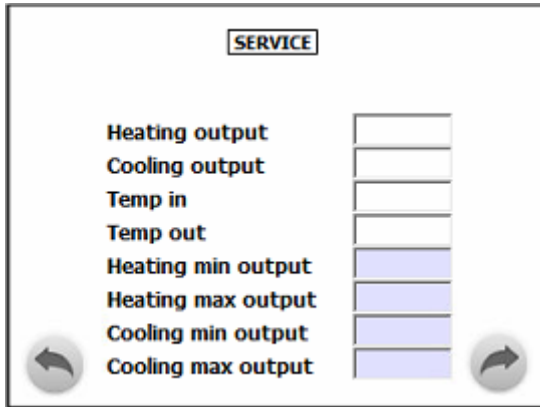
Huollon asetussivulta tuli löytyä taulukossa 2 esitetyt puhallinpatterin rekisterit. Asetussivuista tehtiin mahdollisimman yksinkertaiset, sillä kaikki grafiikat oli tarkoitus ladata MultiDisplayn sisäiseen muistiin. Ladattessa grafiikat sisäiseen muistiin saadaan se etu, että grafiikat voidaan myöhemmin päivittää VAK:sta etänä. Tällöin kuitenkin grafiikoiden muistikokoa joudutaan seuraamaan, sillä MultiDisplayn sisäisen muistin määrä on rajallinen.

TAULUKKO 2. Huollon asetussivun rekisterit

Nimi	Luku/Kirjoitus	Rekisteri
Lämmitysventtiili %	Luku	3x00013
Jäähdytysventtiili %	Luku	3x00012
Min flow lämmitys	Luku/Kirjoitus	4x35233
Max flow lämmitys	Luku/Kirjoitus	4x35234
Min flow jäähdytys	Luku/Kirjoitus	4x35231
Max flow jäähdytys	Luku/Kirjoitus	4x35232
Temp in	Luku	3x00018
Temp out	Luku	3x00019

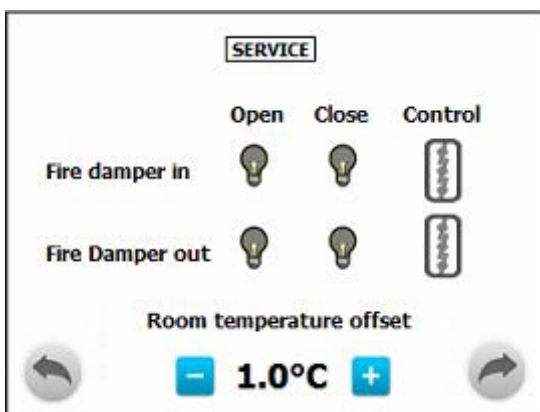
Valmiin huoltovalikon ensimmäiseltä sivulta löytyi taulukon 2 rekisterit. Myöhemmässä toteutusvaiheessa huomattiin, että Min/Max flow -rekisterien kirjoittaminen

ei vaikuttanut venttiilien ajoon halutulla tavalla. Siksi Min/Max flow -rekisterien kirjoitukset korvattiin ohjelmalliselle PID-säätimelle annettavilla minimi- ja maksimiulostuloilla. Tällöin säädin määrittäisi lämmitys- ja jäähdytysventtiilien ulostulojen rajat. Kuvassa 15 on lopullinen huoltoasetusten ensimmäinen sivu. Sinertäviin laatikoihin voidaan asettaa arvot MultiDisplay-näytöltä.



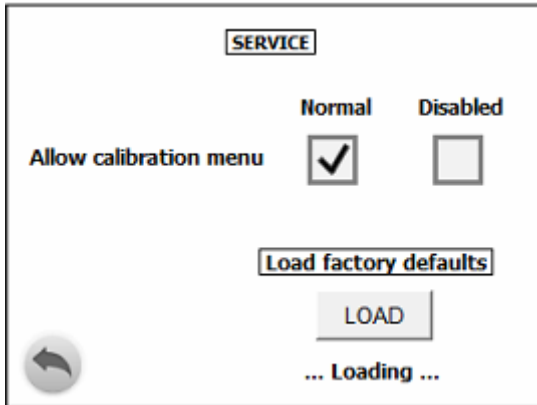
KUVA 15. Huolto-asetusten 1. sivu

Toiselle sivulle (kuva 16) tehtiin tarkasteltavaksi palopeltien indikointien ja ohjauksien tilat. Huonelämpötilamittauksen anturi on integroitu MultiDisplay-näyttöön. Näytön itse tuottama lämpö saattaa nostaa lämpötilalukemaa tai näytön sijainti saattaa aiheuttaa tilanteen, jossa lämpötilanmittaus vääristyy. Tämän vuoksi luotiin huonelämpötilamittaukselle kalibrointimahdollisuus, jossa lämpötilalle voitiin antaa +/-1,0 °C poikkeutusta 0,1 °C:n askelilla.



KUVA 16. Huolto-asetusten 2. sivu

MultiDisplay-näytössä on oletuksena sallittu kalibrointivalikko. Valikkoon pääsee painamalla pitkään näyttöä. Hotellin asiakkaan ei kyseiseen valikkoon tarvitse päästä, mutta huoltohenkilön täytyy. Täten kalibrointivalikko estettiin oletuksena ja sen salliminen tehtiin mahdolliseksi huoltoasetuksista (kuva 17). Huoltoasetuksista voitiin myös palauttaa huonesäätimelle tehdasasetukset.



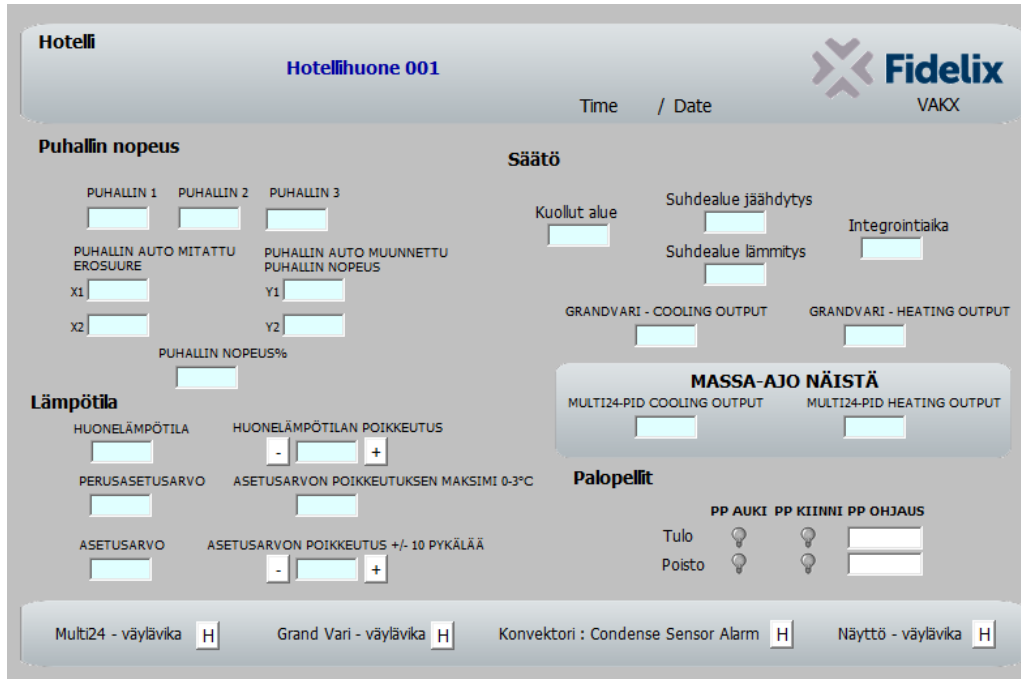
KUVA 17. Huolto-asetusten 3. sivu

4.3 VAK-liityntä

Projektinhoitaja oli määritellyt toimintaselostuksessa kaikki arvot, jotka tuli olla muutettavissa tai luettavissa VAK:lta. VAK:lta tuli löytyä

- puhaltimen auto-asennon puhallinnopeuden muunnostaulukko, joka määrytyi huonelämpötilan ja asetusarvon välisen erosuureen mukaan
- säädön kuolleen alueen koko
- puhallinpatterin kondenssianturi hälytys
- puhaltimen 1-2-3-asentojen arvot
- lämpötilan asetusarvon ylä- ja alaraja
- asetusarvon ohitus
- palopellit
- lämmitys- ja jäähdytysventtiilien massa-ajo
- huonelämpötila, asetusarvo
- puhaltimen nopeus
- jäähdytysventtiilin säätö ja lämmitysventtiilin säätö.

Kaikki projektinhoitajan vaatimukset saatiin täytettyä ja liityntään lisättiin muita tarpeellisia asetuksia, kuten säädön parametrit ja huonelämpötilan poikkeutus. MultiDisplay-näytölle, Multi-24:lle ja puhallinpatterille tehtiin kommunikaatiohälytykset, joista voitiin todeta nopeasti, jos väyläkommunikointi ei toimi (kuva 18).



KUVA 18. Valvonta-alakeskukselle tehdyn liitynnän grafiikka

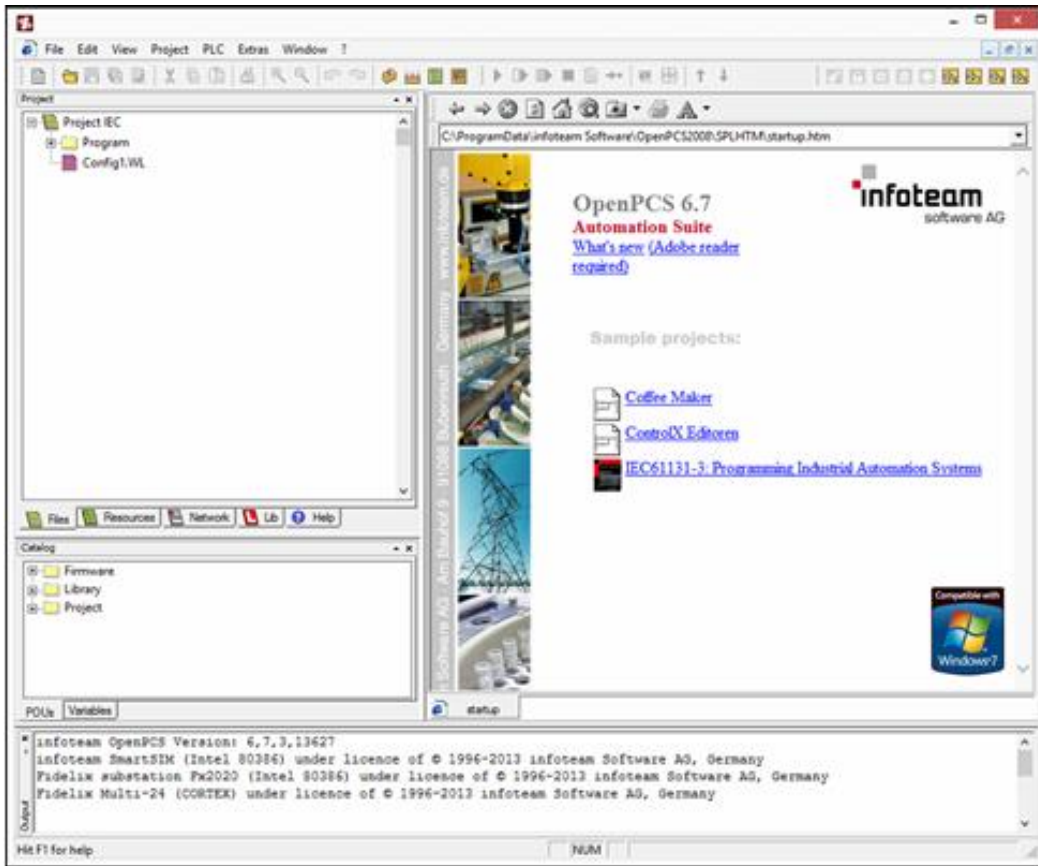
Jokaiseen hotellihuoneeseen tuli kaksi palopeltiä. Jokaisesta palopellistä saatiin kiinni- ja auki-indikointi. Palopeltejä ohjattiin VAK:sta siten, että Multi-24:n tehtäväksi jäi toimia I/O-korttina VAK:lle.

4.4 Sovellusohjelma

Varsinainen säädinsovellus ohjelmoitiin Multi-24:lle Infoteam Software AG:n OpenPCS-ohjelmointityökalulla (kuva 19). Ohjelmointi tapahtuu IEC 61131-3 -standardin määrittämällä Structured Language -tyyppisellä ohjelmointikielellä.

IEC 61131 -standardi on ohjelmoitavien logiikoiden toteutuksia ohjaava standardi ja sen kolmas osio, IEC 61131-3, määrittää ohjelmointiin käytettävät menetelmät, kuten kielet, kielten syntaksin, tietotyypit, ohjelmalohkojen rakenteet ja määrittäykset (15).

Yleisiä rakenteita automaatiosovellusten toteutuksessa ovat erilaiset totuustietojen vertailut, ehtolauseet ja funktiolohkojen kutsuminen tietyillä input-arvoilla.



KUVA 19. OpenPCS-ohjelmointityökalu

Ohjelmointi

Sovellusohjelmointi aloitettiin Multi-24-sovelluksille kootulta mallipohjalta eli templatelta, jossa on valmiita funktiolohkoja ja ohjelmapohjia muun muassa erilaisten hälytysten, säätöjen, käyttöliittymärajapinnan ja mittauksen käsittelyyn. Mallipohjasta karsittiin tarpeettomat ohjelmapohjat pois käytöstä. Ohjelmatiedostoihin ja funktiolohkoihin kuuluu yleensä kaksi osaa, VAR-osa muuttujien määrittelyyn sekä varsinainen ohjelmakoodi. VAR-osassa määritellään ohjelmaan tai funktiolohkoon ulkopuolelta tulevien, sisäisten ja ulospäin annettavien muuttujien sisäinen nimi sekä ohjelman sisällä käytettävät funktiot ja funktiolohkot.

Ohjelma-osassa luetaan ensin tarvittaviin muuttujiin arvot tietokannasta, käsitellään muuttujia ja lopuksi kirjoitetaan muuttujien arvot tietokantaan. Funktiolohkoissa voidaan tehdä muutoksia tietokantaan lohkon sisällä, kuten hälytyksen tilan kirjoittaminen hälytyslohkoissa. Erillisiä ohjelmätiedostoja on tämän säätimen tapauksessa neljä: pääohjelma Application, IOInterface, ModbusDevices ja Databaselnit. Pääohjelmassa Application tehdään mittausten käsittelyä, säädölliset toteutukset ja hälytysten käsittelyä. IOInterfacessa tehdään fyysisen I/O:n käsittely, ModbusDevicessä määritetään kommunikointi Chillerin VariPro-ohjainkortin ja Multi-24:n välillä ja Databaselnitissa luodaan pistetietokanta.

Säätö

Puhallinpatterin säätö tapahtui Multi-24:ään IEC-koodissa, Application-ohjelmatiedostossa. Multi-24:ään on toteutettu laiteohjelmistotasolla PID-säädin-funktio- lohko, joka sopii yleisimpiin säätökohteisiin. Lohkolle annetaan parametreinä mitaustieto, asetusarvo, viritysarvot sekä säädinkohtaiset asetukset.

Lohkolle annettiin parametreinä lämmityksen minimi- ja maksimiarvot (.min2 ja .max2), sekä jäähdytyksen minimi- ja maksimiarvot (.min1 ja .max1). Nämä arvot voitiin asettaa MultiDisplay:n huoltoasetuksista. Säädön suhdealueet (.pband1 ja .pband2), kuollut alue (.deadzone) sekä integrointi-aika (.iTime) saatiin VAK:lta. Lohkolle annettiin kaksi säätöporrasta (.stages) ja säädön tasapainopisteeksi (.balance) laitettiin ensimmäinen porras eli jäähdytys. Mukavuus-parametri (.comfort) poistaa säädöstä integroinnin, jos mittauksen ja jäähdytyksen tai lämmityksen asetusarvon erosuure on pienempi kuin mukavuus-parametrin arvo (kuva 20). Tämä rauhoittaa säätöä poistamalla pienen ulostulon huojunnan, kun mittaus on aivan asetusarvon tuntumassa. Kuvassa 21 on esitelty mukavuus-parametrin ja kuolleen alueen toiminta.

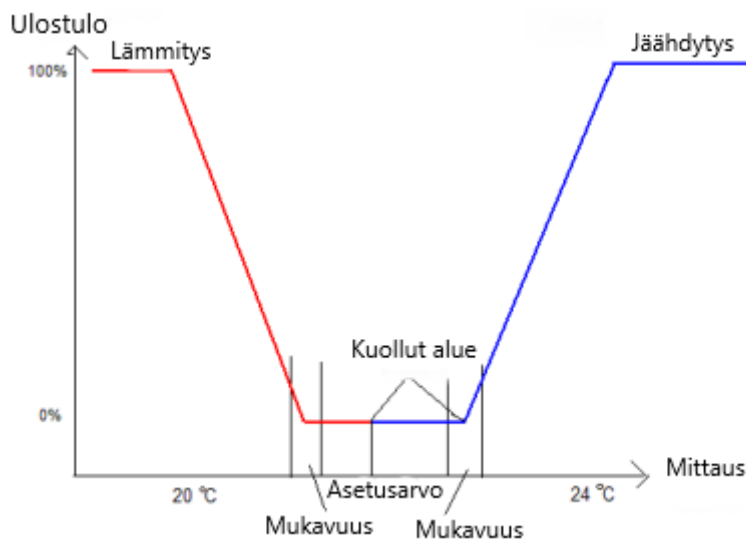

```

VAR
PID_TE: PIDFB;
VAKDeadzone: real;
SetPoint : real;
CoolingOut, HeatingOut: real;
pBandCooling, pBandHeating, RoomTemp: real;
iTimePID: int;
HeatMin, HeatMax, CoolMin, CoolMax : real;
END_VAR

PID_TE.max1 := REAL_TO_INT(CoolMin); (*Jäähdytys on porras 1*)
PID_TE.min1 := REAL_TO_INT(CoolMax); (*Jäähdytys on porras 1*)
PID_TE.max2 := REAL_TO_INT(HeatMax); (*Lämmitys on porras 2*)
PID_TE.min2 := REAL_TO_INT(HeatMin); (*Lämmitys on porras 2*)
PID_TE.pband1 := pBandCooling;
PID_TE.pband2 := pBandHeating;
PID_TE.iTime := iTimePID;
PID_TE.stages := 2; (*Käytössä 2 säätöporrasta*)
PID_TE.balance := 1;
PID_TE.deadzone := VAKDeadzone;
PID_TE.comfort := 0.1;
PID_TE.dTime := 0; (*Ei käytetä derivointia, joten asetetaan se nolnaan*)
PID_TE.measurement := RoomTemp;
PID_TE.setpoint := SetPoint;
PID_TE(); (*Kutsutaan säätöfunktiota*)
CoolingOut := PID_TE.out1; (* Jäähdytyksen ulostulo porras *)
HeatingOut := PID_TE.out2; (* Lämmityksen ulostulo porras *)

```

KUVA 20. Sovelluksessa toteutetun PID-lohkon parametrit



KUVA 21. Kuolleen alueen ja mukavuus-parametrin tarkoitus

4.5 Multi-24 -aliväylä

Multi-24:n aliväylään tuli MultiDisplay-näyttö ja Chillerin Grand VariPro -puhallin-patteri. Kommunikointi MultiDisplayn ja Multi24:n välillä tapahtuu laiteohjelmisto-

tasolla. Multi-24:lle tehtävään pistetietokantaan määritellään pisteet, jotka kommunikoidaan MultiDisplay-näytölle. Puhallinpatterin pisteet ja kommunikointi luotiin puhallinpatterin rekisterilistan mukaan.

Toimintalohko

Chillerin puhallinpatterin ja Multi-24:n liityntä tehtiin erillisenä toimintolohkona Chillerin toimittaman Modbus-rekisterilistan mukaan. Liityntää varten rekisterilistasta poimittiin vain toimintaselostuksessa mainitut rekisterit. Liityntän testausta varten saatiin Chillerin puhallinpatterista demo-kappale, jolla päästiin testaamaan huonesäädinkokoonpanon toimivuutta.

Chillerin Grand VariPro -puhallinpatterin sisäinen toiminta oli käytännössä mysteeri. Toimintaselostusta VariPro-ohjainkortin toiminnasta ei saatu ja toiminnallisuus piti selvittää lähinnä rekisterilistan (liite 3) perusteella.

Kommunikointi

Modbus-kommunikoinnin perusedellytys on tietää laitteiden osoitteet ja väylänopeus, -pariteetti ja pysäytysbitit. Projektinohitaja oli määritelly Chillerille osoitteen 32. Oletuksena Chillerin laiteosoite oli sama kuin sen ID. Jälkikäteen laiteosoite voitiin muuttaa VariPro-kortilla olevilla DIP-kytkimillä. Muutaman yrityksen jälkeen huomattiin, että Chillerin laiteosoite vaihtuu vasta kun sitä käytetään pois päältä. Chillerin oletusväylänopeus oli 38 400bit/s, pariteettia ei ole ja pysäytysbittejä on yksi.

MultiDisplay tukee väylänopeuksia 9 600–115 200 bit/s. Orja käytössä MultiDisplay tunnistaa väylänopeuden automaattisesti, pariteettia ei ole ja pysäytysbittejä on yksi. Multi-24-säätimen aliväylä (liittimet EA ja EB) toimii oletuksena väylänopeudella 57 600 bit/s, ei pariteettia ja pysäytysbittejä on yksi.

Multi-24:n aliväylä konfiguroitiin toimimaan nopeudella 38 400 bit/s, pariteetti ei ole ja pysäytysbitit 1, sillä ne olivat VariPro-ohjainkortin vakioasetukset. Modbus-organisaation ohjeiden mukaan tapauksessa joissa pariteettia ei ole, pysäytysbittien määrän vaaditaan olevan 2. Ohjeiden mukaan pariteetin oletus arvo pitää olla parillinen ja laitteen tulee tukea parillista pariteettia (12, s. 12). Vaikka tämä

on Modbus-organisaation protokollan vaatimus. Monet valmistajat eivät noudata sitä. Ilman pariteettia, 1 pysäytysbitillä toteutettu Modbus-kommunikointi on erinäisten valmistajien laitteiden kesken yleistä.

Testattaessa liityntää huomattiin, että Multi-24:n toimintolohkolla piti käyttää yhtä numeroa pienempää aloitusrekisteriä, kuin Chillerin rekisterilistan indeksi. Laittevalmistajat voivat tehdä omat Modbus-tietolohkonsa laitteen rekistereille ja indeksoida ne haluamallaan tavalla. Tällöin kuitenkin esimerkiksi ensimmäinen digitaalitulostulo rekisterilistan indeksillä 1 tulisi olla luettavissa funktiokoodilla 01 osoitteesta 0.

Ennakointia

Projektinhoitajan toimittamassa hotellihuoneen kytkentäkuvassa MultiDisplay-näyttö ja Chiller Grand VariPro -puhallinpatteri on kytketty virheellisesti tähtitopologiaan (star topology) (liite 4). Modbus RTU tukee virallisesti ainoastaan väylätopologiaa (bus topology). Ennen sovelluksen toimittamista kommunikointi tähtitopologiassa testattiin laitekokoontamalla siltä varalta, että hotellityömaalla johdotus ei mahdollistaisikaan väylätopologian käyttämistä.

4.6 Väylävirheitä

MultiDisplay-näytön ja Chillerin VariPro-ohjainkortin kommunikaatiot toimivat hyvin yksinään, mutta kytkettäessä ne samaan väylään epäonnistuivat molempien kommunikaatiot. Ongelmaa lähdettiin avaamaan taltioimalla väyläliikennettä ja analysoimalla saatua väylälokia (liite 5). Vaikutti siltä, että Chillerin VariPro-ohjainkortin kommunikaatio häiriintyi MultiDisplayn pitkästä datapurskeesta. VariPro-ohjainkortille osoitettuun ensimmäiseen kyselyyn VariPro-ohjainkortti ei vastannut ollenkaan, ja tämä aiheutti kyselylle aikakatkaisun (Timeout) (liite 5, s. 2). Vastaavanlaiset aikakatkaisut ovat nähtävissä myös väylälokin myöhemmissä vaiheissa (liite 5, s. 4, 5). Multi24:n toimintalohkossa aikakatkaisun sekä kyselyn nopeuden (Poll interval) asetusarvoja muutettiin suuremmiksi (Timeout = 0, 8–20 sekuntia ja Poll interval = 1–60 sekuntia), mutta se ei poistanut ongelmaa.

Modbus-protokollan mukaan, jos orjalaite havaitsee virheen sille lähetetyssä kyselyn kehyksessä, se ei vastaa. Jos orja pystyy käsittelemään kyselyn kehysten

ja havaitsee virheen kyselyn muotoilussa tai kyselyn toiminto ei ole sallittu, orjan pitää vastata poikkeusviestillä (12, s. 10). Ensimmäinen kysely VariPro-ohjainkortille datapurskeen jälkeen vastaa oikeanlaista modbus-kehystä ja oletettavasti VariPro-ohjainkortti tulkitsee sen modbus-kyselyksi. Ohjainkortti kuitenkin havaitsee kehyksessä jonkin virheen ja jättää vastaamatta.

Koska ensimmäinen kysely VariPro-ohjainkortille epäonnistui aina, tehtiin Multi-24:n VariPro-liitynnän toimintalohkon alkuun dummy-kysely, johon vastaaminen sai epäonnistua. Dummy-kysely tehoi, sillä vastaus dummy-kyselyyn epäonnistui ja samalla muiden rekistereiden kommunikointi toimi ongelmitta. Tällainen korjaus ei ole suositeltava, mutta tässä tapauksessa ainut vaihtoehto. Tapauksesta ilmoitettiin VariPro-ohjainkortin valmistajalle. Projektilla oli oma aikataulunsa ja tämän korjauksen ansiosta sovellus saatiin toimitettua aikataulussa projektinhoitajalle.

4.7 Tulosten tarkastelu

Taltioitu väylävirhe saattoi johtua kummastakin osapuolesta. Jos virhe johtui Multi-24:stä, se tarkoittaisi, että Multi-24:n laiteohjelmistotasolla toteutettu Modbus-kommunikointilohko olisi viallinen. Multi-24:n aliväylään on kuitenkin liitetty yli 5 vuoden ajan eri valmistajien laitteita ilman vastaavaa ongelmaa.

VariPro-ohjainkortti on suunnattu Chillerin omalle laitteistolle. Ja se varmasti on toiminut ongelmitta tapauksissa, joissa ohjainkortti kommunikoi Chillerin muiden tuotteiden kanssa. VariPro-ohjainkortin kommunikointisäännöistä ei tiedetä mitään, joten vian lähteen etsiminen ohjainkortin puolelta on hankalaa. Tämän vuoksi Chillerille kerrottiin saaduista virheistä, jos yritys haluaa itse tutkia ongelmaa tarkemmin.

Ohjelma, jolla laitteiden välinen kommunikointi aliväylässä taltioitiin, ei taltioi aikaleimoja tarpeeksi luotettavasti. Aikaleimoissa saattoi olla heittoa, jolloin lokissa näkyvät aikakatkaisut eivät välttämättä olleetkaan aikakatkaisuja, vaan Multi-24 saattoi lähettää kaksi kyselyä peräkkäin ilman aikakatkaisun odotusaikaa. Väyläliikenteen taltioiminen toisella ohjelmalla olisi ollut hyvä lisä vertailupohjaksi.

Varmuutta virheen aiheuttajasta ei voida sanoa, sillä molemmilla laitteilla voi olla siihen osuutensa. Käytetty taltiointiohjelma ei tuota täysin varmoja aikaleimoja, joten sen tuottamasta lokista ei voida tehdä lopullisia johtopäätöksiä.

Virheen löydyttyä molempien valmistajien, sekä Fidelixin että Chillerin, olisi hyvä tarkistaa laitteidensa kommunikointisäännöt, jotta vastaavan virheen aiheuttaja selviäisi. Työ tehtiin Fidelixille, joten heillä virhe on jo tiedossa. Chiller Oy ei ole työn toimeksiantaja, vaan hotelliprojektin yksi laitetoimittaja. Chillerille ilmoitettiin tapahtuneesta tilanteesta ja yritys käsittellee asian omalla tavallaan.

5 YHTEENVETO

Työn tarkoituksena oli tehdä sovellus huonesäätimelle, jonka loppusijoituspaikana oli hotellihuone. Sovelluksella ja käyttöliittymällä oli projektinhoitajan asetamat vaatimukset, jotka niiden tuli täyttää. Sovelluksen testaukseen tuli kiinnittää huomiota, sillä jälkeempään korjaaminen tulisi olemaan työlästä ja aikavievää.

Työn tuloksena saatiin valmiit käyttöliittymägrafiikat, huonesäädinsovellus ja alakeskusliityntä, joita projektinhoitaja pystyi käyttämään hotelliprojektissaan. Multi-24:lle valmistunut VariPro-ohjainkortin toimintalohko on nyt käytettävissä muihinkin sovelluksiin.

Hotellihuoneen huonesäätimen käyttöliittymästä saatiin esteettinen ja yksinkertainen. Uskon, että kuka tahansa asiakkaista osaa käyttää sitä ongelmitta. Valvonta-alakeskukselle tehty grafiikka oli vain malli projektinhoitajalle, jonka pohjalta hän pystyi rakentamaan haluamansa näköisen alakeskuksen käyttöliittymän.

Testaukseen panostettiin ja se tuotti toimivan sovelluksen. Kommunikointivirhe Multi-24:n ja VariPro-ohjainkortin välillä pakotti tutkimaan Modbus-kommunikoinnin perusteita. MultiDisplay-näytölle osoitettu datapurske sekoitti VariPro-ohjainkortin kommunikointia. VariPro-ohjainkortti jätti vastaamatta ensimmäiseen sille osoitettuun kyselyyn MultiDisplayn datapurskeen jälkeen. Tämä vastaamatta jättäminen ratkaistiin ”dummy”-kyselyllä, johon vastaus sai epäonnistua. Taltioitu virhetilanne saatettiin myös VariPro-ohjainkortin valmistuttajan tietoisuuteen.

Opinnäytetyö syvensi omaa osaamistani Multi-24:n sovelluksen teossa sekä modbus-liitynnöissä. Työssä käytetyt ohjelmistot ovat nyt minulla hyvin hallussa ja seuraavan sovelluksen tekeminen on jo paljon helpompaa. Työ haastoi terveellä tavalla ja apua sai Fidelix Oy:n työntekijöiltä aina pyydettyäessä. Haluankin kiittää koko Oulun konttoria väkeä tuesta, ja erityisesti tuotekehitysinsinööri Kai Kämäräistä sekä tuotekehitysinsinööri Timo Koskimäkeä. Kiitos myös Oulun aluepäällikölle Arto Nissilälle mahdollisuudesta tehdä tämä opinnäytetyö.

LÄHTEET

1. Fidelix-yrityksenä. 2018. Fidelix. Saatavissa: <https://www.fidelix.fi/fidelix/>. Hakupäivä 2.4.2018.
2. Suomäki, Jorma – Vepsäläinen, Sami. 2013. Talotekniikan automaatio – Käyttäjän opas. Helsinki: Kiinteistöalan kustannus Oy.
3. Harju, Pentti. 2014. Talotekniikan mittauksia, säätöjä ja automatiikkaa. Kouvolaa: Penan Tieto-Opus Ky.
4. Suomen rakentamismääräyskokoelma D2 Rakennusten sisäilmasto ja ilmanvaihto. Määräykset ja ohjeet 2012. 2011. Finlex. Saatavissa: http://www.finlex.fi/data/normit/37187-D2-2012_Suomi.pdf. Hakupäivä 6.4.2018.
5. Härkönen, Pentti – Mikkola, Juhana – Piikkilä, Veijo – Sahala, Antti – Sahlstén, Toivo – Sandström, Börje – Sirviö, Arto – Spangar, Tapani – Sulku, Jukka. 2012. Rakennusautomaatiojärjestelmät. Tietotekniset järjestelmät. Espoo: Sähköinfo Oy.
6. STRA-24. 2018. Fläkt Woods Oy. Saatavissa: <http://www.flaktwoods.fi/products/air-diffusion-/chilled-beams/chilled-beam-controllers/stra-24/>. Hakupäivä 7.3.2018.
7. Multi24 vapaasti ohjelmoitava kenttäsäädin. 2018. Fidelix Oy. Saatavissa: https://www.fidelix.fi/wp-content/uploads/MULTI24_FI.pdf. Hakupäivä 1.4.2018.
8. Multi-24 getting started guide. Fidelix Oy. Sisäinen lähde. Saatavissa: http://support.fidelix.fi/GetFile.aspx?file=Multi24\FXMULTI24_GETTING_STARTED_ENG_V27.pdf. Hakupäivä 21.3.2018.
9. multiDisplay (A) Pieni Modbus-kosketusnäyttö. 2018. Fidelix Oy. Saatavissa: https://www.fidelix.fi/wp-content/uploads/multiDISPLAY_A_FI.pdf. Hakupäivä 21.3.2018.

10. GRAND Vari™ ja GRAND Vari™ PRO -puhallinpatterit. 2018. Chiller Oy.
Saatavissa: <http://www.chiller.fi/grandvari>. Hakupäivä 5.4.2018.
11. FAQ. 2017. Simply Modbus. Saatavissa:
<http://www.simplymodbus.ca/FAQ.htm>. Hakupäivä 2.3.2018.
12. Modbus over serial line. Specification and implementation guide V1.02.
2006. Modbus organization. Saatavissa: http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf. Hakupäivä 2.3.2018.
13. Modbus application protocol specification V1.1b3. 2012. Modbus organization. Saatavissa: http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf. Hakupäivä 2.3.2018.
14. Tikkanen, J. 2013. Modbus-integrointi rakennusautomaatiojärjestelmään. Opinnäytetyö. Tampereen ammattikorkeakoulu. Sähkötekniikan koulutusohjelma. Saatavissa: <http://urn.fi/URN:NBN:fi:amk-201305148469>. Hakupäivä 15.5.2018.
15. Introduction into IEC 61131-3 Programming Languages. 2012. PLCopen. Saatavissa: http://www.plcopen.org/pages/tc1_standards. Hakupäivä 3.4.2018.

LIITTEET

Liite 1 Multi-24 kytkentäkuva

Liite 2 Chiller Grand VariPro kytkentäkuva

Liite 3 Chiller VariPro-ohjainkortin rekisterilista

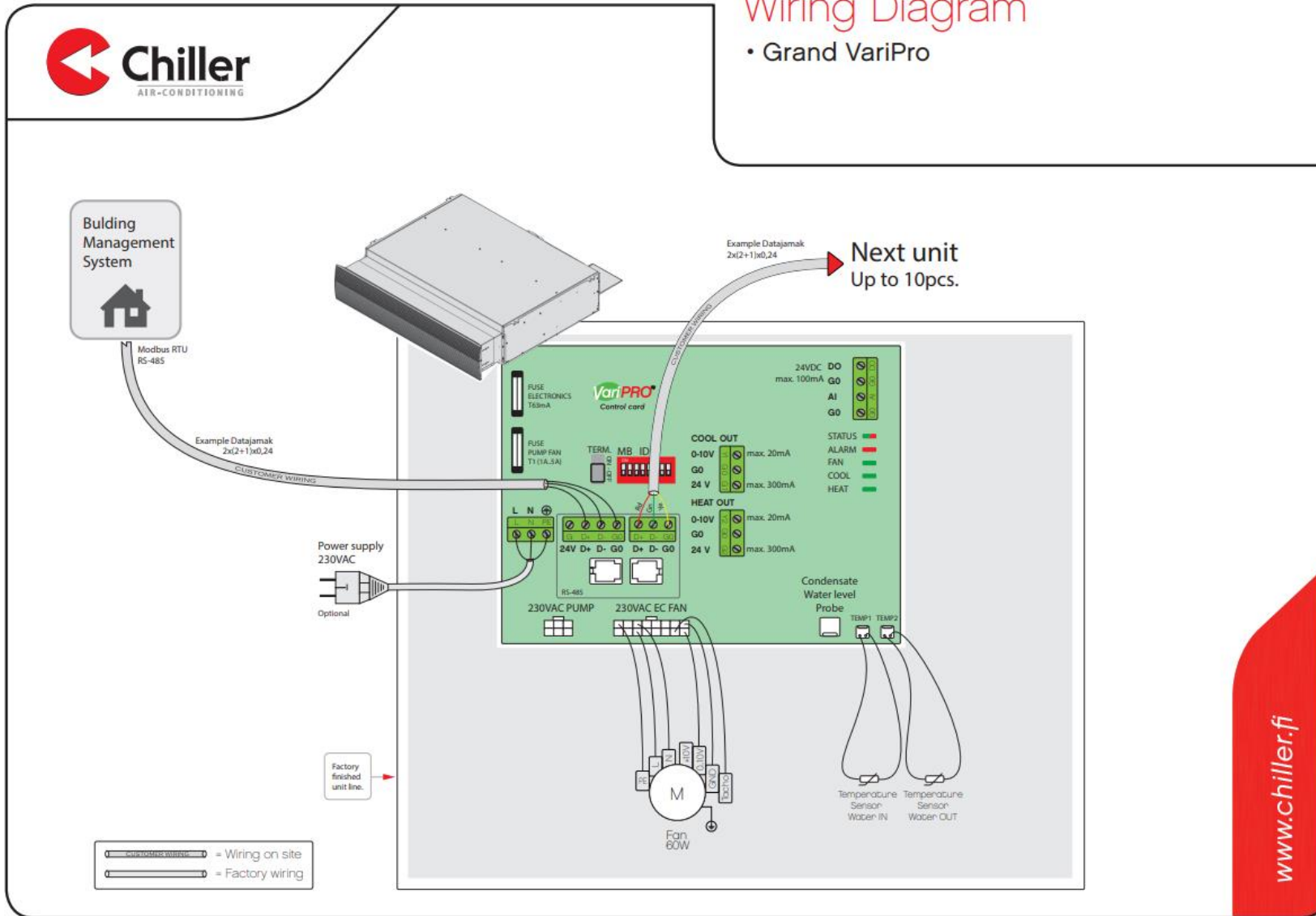
Liite 4 Hotellihuoneen kytkentäkuva

Liite 5 Tallennettu väyläloki

Liite 6 Sovelluksenluonnin periaatteellinen luonnos

Wiring Diagram

• Grand VariPro



Controller card SW 0.33

PLC addressing (base 1)

Description		Min	Max	Unit	Note(s)
Holding register 16 bit integer register R/W					
4x00001	Common status register				Bit0 = Find unit (not used) Bit1 = Force pump ON Bit1-15 = TBD
4x00002	Set fan speed	0	100	%	
4x00003	Set cool output	0	100	%	
4x00004	Set heat output	0	100	%	
4x00005	Set digital output	0	1		
4x00006	System state				User interface send this 1=Stand-by 2=Normal 3=Away 4=Flush (set cool&heat 100%) 10=Test mode (allow cool/heat without fan)
4x00007	CO2 control	0	100	%	Drive heat output if P_1110 = 5
4x00020	Testmode stop	0	1		1=stop this unit while test mode
4x00021	Temperature offset for input 1			x10C	
4x00022	Temperature offset for input 2			x10C	
4x00023	Stand-alone setpoint	0/100	300	x10C	0 = Normal operation, 100-300 = Stand-alone
4x00101	Re set service reminder counter	0	1		1=reset
4x00102	Re set fan ON-time counter	0	1		1=reset
4x00103	Re set pump ON-time counter	0	1		1=reset
4x00104	Set factory parameters	0	1		1=reset
Application parameters (only for seldom use)					
4x35201	Skip startup wizard	0	1		
4x35202	BOX60EL name on UI's screen	0	0		
4x35203	Parameters date Year	2014	2100		
4x35204	Parameters date Month	1	12		
4x35205	Parameters date Day	1	31		
4x35206	Condense pump type 0=NA, 1=KS, 2=SI	0	2		0=NA, 1=KS, 2=SI
4x35207	Cooling valve type	0	4		0=NA, 1=24V-NC, 2=24V-NO, 3=0-10V, 4=10-0V
4x35208	Cooling valve nominal flow value	0	3200	l/h	0=disabled
4x35209	Cooling valve 1% position	1	50	%	
4x35210	Cooling valve 100% position	50	100	%	
4x35211	Heating valve type	0	4		0=NA, 1=24V-NC, 2=24V-NO, 3=0-10V, 4=10-0V, 5=co2 damper
4x35212	Heating valve nominal flow value	0	3200	l/h	0=disabled
4x35213	Heating valve 1% position	1	50	%	
4x35214	Heating valve 100% position	50	100	%	
4x35215	Number of fans	1	2		
4x35216	Fan alarm delay	0	10	min	0=disabled
4x35217	Fan min % value for cooling	0	50	%	
4x35218	Fan max % value for cooling	50	100	%	
4x35219	Air out limit	10	25	°C	
4x35220	DI input type	0	9		1=UI use, 2=cool stop, 3=heat stop, 4=stop cool&heat, 5=stop all, 6=Analog 7=RH, 8=co2, 9=temp
4x35221	DI input polarization	1	2		1=NO 2=NC

4x35222	DO output type	0	2		1=UI use, 2=pump cool, 3=pump heat, 4=DI power out, 5=Cool PWM, 6=Heat PWM, 7=Follow DI, 8=two step heating
4x35223	DO output polarization	1	2		1=NO 2=NC
4x35224	Temp1 on X15	0	4		0=NA, 1=SA 2=EA, 3=SW, 4=EW
4x35225	Temp2 on X16	0	4		0=NA, 1=SA, 2=EA, 3=SW, 4=EW
4x35226	Service reminder intervall, Filter high limit	1000	30000	h	
4x35227	Cool/heat 24V PWM 100% period time	1	120	s	
4x35228	Cool actuator speed	30	600	s	
4x35229	Heat actuator speed	30	600	s	
4x35230	Installation type	1	5		1=C/H, 2=C, 3=H, 4=C&Rad.H, 5=CO
4x35231	Cool min flow value	0	1000	l/h	
4x35232	Cool max flow value	0	3000	l/h	
4x35233	Heat min flow value	0	1000	l/h	
4x35234	Heat max flow value	0	3000	l/h	
4x35235	Consumption at 2V	1	300	W	
4x35236	Consumption at 4V	1	300	W	
4x35237	Consumption at 6V	1	300	W	
4x35238	Consumption at 8V	1	300	W	
4x35239	Consumption at 10V	1	300	W	
4x35240	Consumption pump	0	10	W	
4x35241	Consumption idle	0	10	W	
4x35242	Fan min % value for heating	0	50	%	
4x35243	Fan max % value for heating	50	100	%	
4x35244	Temperature deltaT warning delay	0	10	h	0=OFF
4x35245	Temperature deltaT warning different	1	5	°C	
4x35246	Return water temperature limit's setpoint	0	20	°C	0=OFF
4x35247	deltaT temperature limiter	1	5	°C	
4x35248	Temperature limiter P-band	0	100		
4x35249	Temperature limiter I-area	1	1000	s	
4x35250	Tacho input type	0	4		0=Disable, 1=NO, 2=NC, 3=RPM 1/1, 4=RPM 2/1
4x35251	Volumetric heat capacity	0	10000	J / (dm ³ * K)	
4x35252	SI-pump alarm input polarization	1	2		1=NO 2=NC
4x35253	Change over valve time	80	300	s	
4x35254	Change over polarity	0	1		0 = Normally cool, 1 = Normally heat
4x35255	Condense input alternative use	0	1		0=normal, 1=cool stop
4x35256	Two step heating	0	100	%	
4x35257	CO2 at 0V	0	5000	ppm	
4x35258	CO2 at 10V	0	5000	ppm	
4x35259	CO2 Calibration	-2000	2000		
4x35260	Stand-alone P	1	40	K	
4x35261	Stand-alone I	1	10000	s	
4x35262	Stand-alone D	0	200		
4x35263	Stand-alone Deadzone	1	50	x10C	

Input register 16 bit integer register Read only

3x00001	Component ID	22	22		Every component type has unique ID
3x00002	Application ID	1	1		
3x00003	Map version	1	1		
3x00004	Application state				0=Init, 1=Normal, 2=No cooling, 3=No heating, 4=Stop both, 5=Stop all, 9=Testing, 10=Flush

3x00005	Alarm register 1				Bit0: Condense sensor alarm Bit1: Missing RPM from fan1 Bit2: Missing RPM from fan2 Bit3-10: Not used Bit11: Parameter file Bit12: No communication Bit13: DeltaT warning Bit15: Service reminder
3x00006	Common status register				Reserved for future use
3x00007	Current fan speed output	0	10000	mV	
3x00008	Tachometer fan1	0	3000	rpm	
3x00009	Tachometer fan2	0	3000	rpm	
3x00010	Condensate tank state	0	3		Level
3x00011	Condensate pump status	0	1		
3x00012	Cooling output	0	100	%	
3x00013	Heating output	0	100	%	
3x00014	Digital input configuration	0	9		0 = Not used 1 = General input, UI use 2 = Cool stop 3 = Heat stop 4 = Cool and Heat stop 5 = Stop all 6 = generic analog IN 7 = Analog IN as RH 8 = Analog IN as co2 9 = Analog IN as temperature
3x00015	Digital input status	0	1		0 = Open, 1=Close
3x00016	Digital output configuration	0	7		0 = Not used 1 = UI controls 2 = Circulation pump for cooling 3 = Circulation pump for heating 4 = Power out for Din 5 = Cooling output 6 = Heating output 7 = Follow Din
3x00017	Digital output status	0	1		0 = OFF, 1=ON
3x00018	Temperature 1			x10C	
3x00019	Temperature 2			x10C	
3x00020	Total operation hours			h	32bit value
3x00021					
3x00022	Device hours since last powered up			h	
3x00023	Filter high limit			h	
3x00024	Filter status			h	
3x00025	Service reminder counter			h	
3x00026	Pump up-time counter			h	
3x00027	Fan up-time counter			h	
3x00028	Power consumption	0	1000	W	
3x00029	Energy consumption day	0	1000	kWh	
3x00030	Energy consumption 30d	0	10000	kWh	
3x00031	Energy consumption 360d	0	10000	kWh	
3x00032	Installation type	1	4		1=Cool/heat, 2=Cool, 3=Heat, 4=Cool&Rad.Heat
3x00033	Cooling hours		65535	h	
3x00034	Heating hours		65335	h	
3x00035	Temperature 1 settings				0=NA, 1=SA 2=EA, 3=SW, 4=EW
3x00036	Temperature 2 settings				0=NA, 1=SA, 2=EA, 3=SW, 4=EW
3x00037	Cooling output power	0	65000	W	
3x00038	Cooling energy consumption day	0	1000	kWh	
3x00039	Cooling energy consumption 30d	0	65000	kWh	

3x00040	Cooling energy consumption 360d	0	65000	kWh	
3x00041	Analog IN value	0	10000	mV	
3x00042	Cool flow (calculated, not measured)			l/s	0xFFFF if not available
3x00043	Heat flow (calculated, not measured)			l/s	0xFFFF if not available
3x00044	Heating output power				
3x00045	Heating energy consumption day	0	1000	kWh	
3x00046	Heating energy consumption 30d	0	65000	kWh	
3x00047	Heating energy consumption 360d	0	65000	kWh	
3x00048	Analog input converted value				o/oo RH if 4x35219=7 CO2 ppm if 4x35219=8 0.1C if 4x35219=
3x00049	Condense input alternative use				0 = normal 1 = cool stop
3x00050	Condense input				bit0 = level1 bit1 = level2 bit2 = level3
3x00051	SecondaryAnalogValue				
3x00051	Spare for future change				
3x00100					
3x50001	Component ID				22
3x50002					
3x50017	Component Name				Chiller - CC
3x50018	Application ID				1
3x50019	HW revision				4 = XT1090E (Din support Ain)
3x50020					
3x50021	Serial number				
3x50101	SW version, major				
3x50102	SW version, minor				
3x50103	SW version, SVN build				
3x50104					
3x50149	Software version info				

