

KARELIA-AMMATTIKORKEAKOULU
Talotekniikan koulutus

Joonas Toivanen

ULTRAÄÄNI-ILMAMÄÄRÄSÄÄDIN
LABORATORIOHARJOITUKSISSA

JA SEN HYÖDYNTÄMINEN

Opinnäytetyö
Huhtikuu 2020



OPINNÄYTETYÖ
Huhtikuu 2020
Talotekniikan koulutus

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

Tekijä(t)
Joonas Toivanen

Nimeke
Ultraääni-ilmamääräsäädin ja sen hyödyntäminen laboratorioharjoituksissa

Toimeksiantaja
Karelia-amk

Tiivistelmä

Tämän työn tavoitteena oli selvittää ultraääneen perustuvien ilmamääränsäätimien mittauksen ja säädön toimintaperiaate. Tavoitteena oli myös selvittää, miten niiden ohjelmointi tapahtui Codesys-ohjelmointiympäristössä. Täytyi myös selvittää, miten ultraääni-ilmamääräsäätimiä voitaisiin käyttää hyödyksi laboratorioharjoituksissa.

Ennen tutkimuksen tekoa opinnäytetyössä perehdyttiin tietoon, jonka ymmärrys tarvittiin työssä tutkittavien ultraääni-ilmamääränsäätimien käyttöön. Opinnäytetyössä käsiteltiin myös ultraääni-ilmamääränsäätimien ominaisuuksia ja niiden toimintaperiaate. Opinnäytetyön tutkimusosiossa esitetään ohjelman teko vaiheittain ja sen tekemiseen tarvittava tieto ja menetelmät.

Työn tuloksena saatiin kattava selvitys siitä tiedosta, jota tarvitaan ultraääni-ilmamääränsäätimien käyttöönotossa. Saatiin tehtyä myös ohjelma, jolla voi ohjata molempia ultraääni-ilmamääränsäätimiä tiettyyn suuntaan asetettujen parametrien mukaan. Työtä on mahdollista jatkokehittää täysin automaattisesti toimivaksi järjestelmäksi.

Kieli
suomi

Sivuja	48
Liitteet	4
Liitesivumäärä	12

Asiasanat

ultraääni-ilmamääräsäädin, tiedonsiirto, Modbus



THESIS
April 2020
Degree Programme in Building
Services Engineering

Tikkarinne 9
80200 JOENSUU
FINLAND
+ 358 13 260 600

Author (s)
Joonas Toivanen

Title
Ultrasound Air Volume Controller and Its Utilization in Laboratory Exercises

Commissioned by
Karelia UAS

Abstract

The aim of this thesis was to explain the operating principle of measurement control of ultrasound air volume controllers. The goal was also to clarify how their programming took place at the Codesys programming environment. There was also a need to find out how ultrasound air volume controllers could be utilized in laboratory exercises.

Before conducting the research, the author familiarized with the knowledge that was required to understand the utilizing of the ultrasound air volume controllers which were studied in this thesis. The thesis also deals with the features and operating principle of the ultrasound air volume controllers. In the research part of the thesis, the gradual making of the program is demonstrated and the information and methods which were required to make the program are presented.

The thesis resulted in a comprehensive explanation of the information which is required in utilizing ultrasound air volume controllers. A program was also made whereby both ultrasound air volume controllers can be controlled in a certain direction according to the set parameters. The work can possibly be further developed into a fully automatic system.

Language	Pages	48
Finnish	Appendices	4
	Pages of Appendices	12

Keywords

ultrasound air volume controller, data transfer, Modbus

Sisältö

1	Johdanto	6
2	Ilmanvaihto	6
2.1	Ilmastoinnin ja ilmanvaihdon ero	6
2.2	Ilmanvaihtojärjestelmän tavoitteet ja suunnittelu	7
2.3	Tarpeenmukainen ilmanvaihto.....	8
2.4	Sisäilman epäpuhtaudet	8
3	Tiedonsiirto.....	9
3.1	Kenttäväylät.....	9
3.2	Sarjaliitännät RS-232 ja RS-485.....	10
3.3	Ohjelmoitava logiikka.....	11
3.4	Codesys.....	12
4	Modbus	12
4.1	Yleistä.....	12
4.2	RTU-kehys.....	13
4.3	Dokumentaatio	14
4.4	Datan esitystyyli.....	14
4.5	Yleisimmät funktiokoodit.....	15
4.6	Yleisimmät virhekoodit.....	16
5	Ultraääneen perustuva ilmamääräsäädin.....	17
5.1	Ultraääni ja ilmamääräsäädin	17
5.2	FläktGroup Optivent® Ultra	18
5.3	Lindab Ultralink® FTCU.....	20
6	Soveltava harjoitustyö	20
6.1	Järjestelmän toimintaperiaate.....	20
6.2	Työssä käytetyt laitteet	21
6.3	Harjoitustyön tiedonsiirto	22
6.4	Pääohjelma.....	24
6.5	Alkumäärytykset	26
6.6	Modbus Master.....	27
6.7	Extquery	28
6.8	Response	29
6.9	Kenttälaitteiden parametrisointi	31
6.9.1	Ilmanlaatulähetin ILH-M-RH-N	31
6.9.2	FläktGroupin ultraääni-ilmamääräsäädin.....	31
6.9.3	Lindabin ultraääni-ilmamääräsäädin.....	33
6.10	Kytkenät.....	34
7	Aliohjelma.....	35
7.1	Tehtävä ja rakenne.....	35
7.2	Muuttujien määrittely	36
7.3	Ilmanlaatulähettimen tietojen luku	37
7.4	Optivent Ultra.....	38
7.4.1	Pellin asento ja ilmavirta	38
7.4.2	Ilmamääräsäätimen ohjaus	39
7.5	Ultralink FTCU.....	40
7.5.1	Pellin asento ja ilmavirta.....	40
7.5.2	Ilmamääräsäätimen ohjaus	42
7.6	Aliohjelman lopetus	43
8	Pohdinta ja yhteenveto.....	44
	Lähteet	46

Liitteet

- Liite 1 Modbus Extended Master PRG ohjelmakirjasto
- Liite 2 Ilmanlaatulähetin ILH-M-RH-N Modbus-rekisterit
- Liite 3 FläktGroup Optivent® Ultra kytkennät ja Modbus-rekisterit
- Liite 3 Lindab Ultralink® FTCU kytkennät ja Modbus-rekisterit

1 Johdanto

Opinnäytetyön tarkoituksena oli selvittää, miten ultraääneen perustuva ilmamäärän mittaus ja säätö toimivat. Tavoitteena oli myös selvittää, miten niitä voidaan hyödyntää laboratorioharjoituksissa. Laboratorioharjoituksia varten selvitettiin myös, kuinka työn harjoituksissa hyödynnettävät laitteet ja komponentit kasataan ja kytketään yhteen siten, että järjestelmä toimii halutulla tavalla. Oli myös selvitettävä, miten laitteiden ohjelmointi tapahtuu Codesys-ohjelmointiympäristössä. Lopuksi tehtiin myös pieni ohjelma tarpeenmukaisesta ilmanvaihdosta.

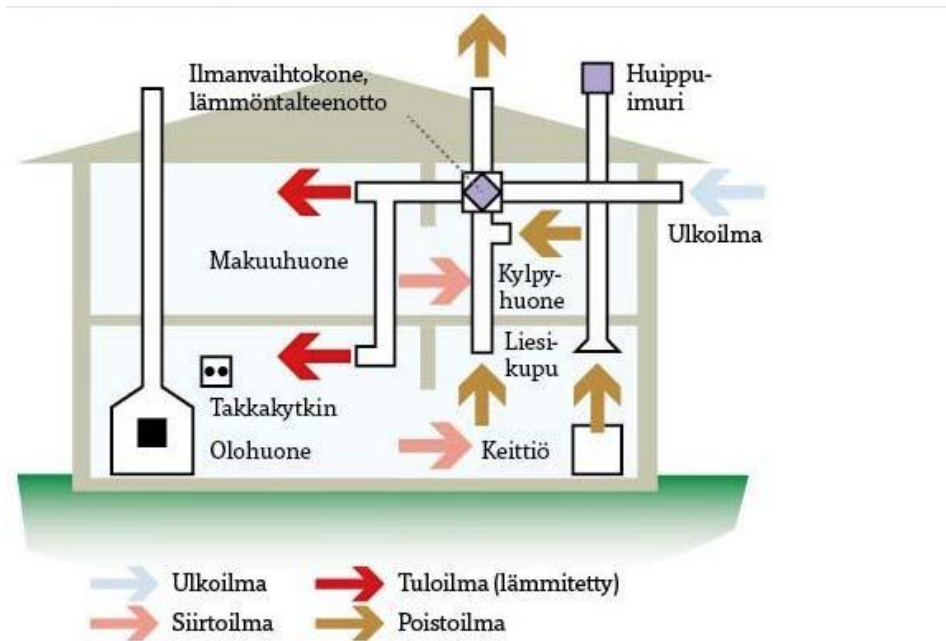
EU:n ilmastotavoitteet määräävät, että tämän vuoden 2020 aikana energiankulutuksen on laskettava 20 %. Energiankulutus on edelleen liian suurta, joten uusia tekniikoita ja parempia ratkaisuja on otettava käyttöön. Etenkin uusien rakennusten energiansäästövaatimukset ovat koventuneet, joten tarpeenmukaisille ilmanvaihtojärjestelmille on kysyntää. [1.]

2 Ilmanvaihto

2.1 Ilmastoinnin ja ilmanvaihdon ero

Ilmastointi ja ilmanvaihto sekoitetaan usein keskenään, vaikka ne eivät tarkoita samaa asiaa. Ilmastointi tarkoittaa sisäilman käsitlemistä eri keinoilla, joita ovat muun muassa jäähdytys ja kuivaus. Uutta ilmaa ilmastointi ei siirrä rakennukseen. Ilmanvaihto tarkoittaa juuri sitä, mitä sana sanoo, eli vaihtaa ilmaa. Ilmaa tuodaan puhtaisiin tiloihin ja viedään likaisista pois. Käytetty ja vanha sisäilma poistuu, ja sen korvaa uusi ja puhdas ilma. [2.]

Koneellinen tulo- ja poistoilmanvaihto (kuva 1) on kaikkein energiantehokkain tapa vaihtaa ilmaa, koska silloin saadaan poistoilman lämpöenergiaa hyväksi käyttäen lämmitettyä tuloilmaa lämmöntalteenotolla. [3.]



Kuva 1. Koneellinen tulo- ja poistoilmanvaihto [3].

2.2 Ilmanvaihtojärjestelmän tavoitteet ja suunnittelu

Ilmanvaihdon on saatava aikaan oleskelutiloissa laadukas sisäilma. Ilmanvaihtojärjestelmän täytyy tuoda rakennukseen riittävä määrä ulkoilmavirtaa ja poistaa sisäilman terveydelle haitalliset aineet, liika kosteus ja epäpuhtaudet, jotka ovat peräisin esimerkiksi ihmisistä tai rakennustuotteista. [4.]

Ilmanvaihtojärjestelmä täytyy suunnitella sellaiseksi, että järjestelmän toimintaan kuuluvat päätoiminnot ovat mitattavissa, ohjattavissa ja seurattavissa. Ilmanvaihtoa voidaan ohjata esimerkiksi lämpötilan, hiilidioksidin tai kosteuden perusteella. Järjestelmän pitää myös kestää koko käyttöiän. Rikkimienemättömyys edellyttää sitä, että järjestelmää käytetään oikein, huolletaan ja kunnossapidetään. Järjestelmä täytyy myös olla kokonaan pysäytettävissä. [4.]

2.3 Tarpeenmukainen ilmanvaihto

Tarpeenmukainen ilmanvaihto tarkoittaa ilmanvaihdon määrän säätämistä automaattisesti asetettujen parametrien mukaan. Se sopii sellaisiin kohteisiin parhaiten, joissa tilan käyttöaste vaihtelee paljon. Tällaisia ovat esimerkiksi tehtaat ja sairaalat. Käyttöaste tarkoittaa sitä, kuinka paljon jotakin tiettyä tilaa käytetään tietyssä ajassa.

Tilassa olevat anturit mittaavat tilan fysikaalisia suureita, kuten esimerkiksi lämpötilaa, CO₂-pitoisuutta ja kosteutta. Mitattujen arvojen perusteella järjestelmä säätää ilmanvaihtoa tilaan sopivaksi järjestelmään asetettujen parametrien mukaan. Ilmanjako tiloihin tapahtuu pääasiassa ilmamääräsäätimien välityksellä järjestelmän ohjausviestin mukaan [5].

Jos halutaan toteuttaa rakennuksen ilmanvaihto tarpeenmukaisesti, on sen toteuttaminen hankalampaa kuin perinteisen järjestelmän. Tarpeenmukaisen ilmanvaihdon suunnittelu on myös suunnittelijalle hieman enemmän työtä teettävämpi. [5] Automaatiikkaan pitää kiinnittää erityisesti huomiota, ja siitä ei ole järkevää tehdä liian monimutkaista, koska yksinkertaisen järjestelmän kokonaisuuden hallinta on helpompaa. Käyttäjän on myös hyvä ymmärtää, miten järjestelmä toimii, eikä pelkästään suunnittelijan. [6]

Automaation tarkoitus on ohjata ja säätää prosesseja, jotta rakennuksen suunnitellut olosuhteet täyttyvät. Keskitettyyn järjestelmään on liitetty usein talotekniikan suurin osa. Järjestelmällä on mahdollista valvoa taloteknistä järjestelmää ja sen toimintatapaa. Nykyiset järjestelmät ilmoittavat tarkasti esimerkiksi kokonaisilmavirran, mutta eivät kuitenkaan sitä, millä tavalla ilma jakautuu rakennuksen eri huonetiloihin. [5.]

2.4 Sisäilman epäpuhtaudet

Hiilidioksidi (CO₂) on sisäilman epäpuhtaus, jota muodostuu ihmisen aineenvaihdunnan seurauksena. Liika CO₂-pitoisuus kiellii usein riittämättömästä

ilmanvaihdesta. Sisäilmassa on liikaa CO₂:ta, jos sen CO₂-pitoisuus on 1 150 ppm (partikkelia miljoonassa) suurempi kuin ulkoilman CO₂-pitoisuus. [7.]

Sisäilman muita epäpuhtauksia CO₂:n lisäksi ovat pienhiukkaset, radon, tupakansavu, kosteusvauriot, asbesti, teolliset mineraalikuidut, VOC¹, formaldehydi, PAH²-yhdisteet ja kasvi- sekä eläinperäiset allergeenit. [8.]

3 Tiedonsiirto

3.1 Kenttäväylät

”Kenttäväylä on digitaalinen ja kaksisuuntainen väyläliityntäinen tiedonsiirtoratkaisu, joka yhdistää älykkäät mittaus- ja ohjauslaitteet, muun automaation, näytöt ja käyttöliittymät.” [9, 28.] Talotekniikassa on monta eri mahdollista väylätekniikkaa. Yleisimmin käytettyjä ovat muun muassa Bacnet, KNX, Modbus ja EnOcean.

Bacnettiä käytetään pääasiassa ohjaamaan LVI-tekniikkaa. KNX:ää käytetään kiinteistötekniikan tiedonsiirtoväylänä, ja sillä voidaan hallita energia- ja käytönhallinnan toimintoja. Modbusia käytetään myös talotekniikassa. EnOcean on tiedonsiirtojärjestelmä, joka toimii radiotaajuudella lyhyitä matkoja. [9, 119–145.] Sitä käytetään kohteissa, missä on mahdotonta vetää kaapeleita [10].

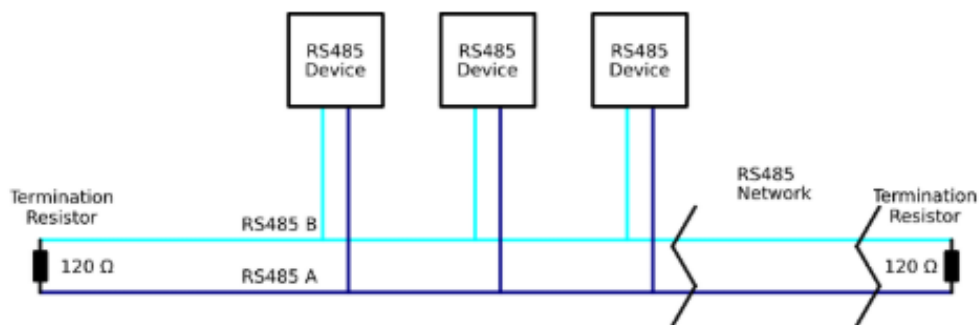
¹ haihtuvat orgaaniset yhdisteet

² polysykliset aromaattiset hiilivedyt

3.2 Sarjaliitännät RS-232 ja RS-485

RS-232 on sarjaliitettä, joka on tarkoitettu kahden laitteen väliseen tietoliikenteeseen. Siinä datan siirto tapahtuu yksi bitti kerrallaan sarjamuodossa. [9, 47.] Ennen kenttäväyliä RS-232-sarjaliitettä käytettiin yhdistämään teollisuuden automaatiojärjestelmät. RS-232 huono puoli oli, että se salli vain kahden laitteen kommunikaation kerrallaan. [11]

RS-485 on sarjaliitettä, jonka rakenne on esitetty kuvassa 2, on RS-232:sen monin verroin parempi seuraaja. Taulukosta 1 voidaan nähdä, että RS-485 kykenee edeltäjänsä RS-232:sta nopeampaan tiedonsiirtoon ja pidempään tiedonsiirtomatkaan.



Kuva 2. RS-485-väylän rakenne [12].

Taulukko 1. RS-232 ja RS-485 välisten ominaisuuksien vertailu [13].

KEY CHARACTERISTICS OF THE RS-232 AND RS-485 SERIAL INTERFACES		
Parameter	RS-232	RS-485
Line configuration	Single-ended	Differential
Mode of operation	Simplex or full duplex	Simplex or half duplex
Maximum cable length	50 feet	4000 feet
Maximum data rate*	20 kbits/s	10 Mbits/s
Typical logic levels	±5 to ±15 V	±1.5 to ±6 V
Minimum receiver input impedance	3 to 7 kΩ	12 kΩ
Receiver sensitivity	±3 V	±200 mV

* Maximum rate at maximum cable length

RS-485:ssä ainoastaan yksi lähetin saa olla kerrallaan aktiivinen. Kaikki muut liitännät ovat suuri-impedanssisessa tilassa ja kuuntelevat sarjavyälän kautta siirrettävää dataa. RS-485-standardin suurin sallittu tiedonsiirtonopeus lyhyillä etäisyyksillä on 10 Mbit/s, mutta käytännössä päästään suotuisissa olosuhteissa jopa 30 Mbit/s. Väyläkaapelin pituus saa olla jopa 1200 metriä, joskin standardissa rajoitetaan tällöin liikennöintiinopeus 90 kilobittiin sekunnissa. RS-485-väylässä käytetään matalia signaalitasoja, jolloin se säteilee hyvin vähän häiriökohinaa ympäristöönsä. Myöskään vastaanottimet eivät ole kohinalle herkkiä differentiaalisen signaloinnin ansiosta. [9, 50.]

3.3 Ohjelmoitava logiikka

Programmable Logic Controller (PLC) eli ohjelmoitava logiikka, on teollisuusautomaatiossa käytetty tietokone (kuva 3). PLC:tä käytetään pohjimmiltaan ohjaamaan teollisuuden automaatiojärjestelmä. Tällä hetkellä ne ovat eri ohjausjärjestelmistä kehittynein ja yksinkertaisin. [14] Pääsyy on se, että tarvittavien johtojen määrä vähentyy huomattavasti, koska PLC:hen on jo valmiiksi asennettu automatiikassa tarvittavat releet, ajastimet ja laskurit ynnä muuta sellaista. [15]

PLC koostuu viidestä eri osuudesta, jotka ovat tulo- ja lähtöalue, keskusyksikkö, ohjelmointityökalu, virtalähde ja muisti. Tuloalue tai tulomoduuili koostuu laitteista kuten kytkimet, tunnistimet ja niin edelleen. Tuloalue on kytketty PLC:hen kiskoa pitkin. Lähtöalue tai -moduuli sisältää kaikki ne laitteet, joita ohjataan erilaisilla tulosignaaleilla esimerkiksi moottorin, lämmittimen tai lampun.

[14.]



Kuva 3. Wago PLC 750-881 [16].

3.4 Codesys

Codesysiä käytetään teollisuuden ohjelmistoalustana. Alustan ydin on IEC-61131-3 mallin ohjelmointityökalu ”CODESYS Development System”. Se tarjoaa käyttäjilleen käytännöllisiä ja integroituja ratkaisuja automaatiosovellusten konfiguraatioon. Käyttäjät voivat joko valita jonkin kontrollerin kattavasta laitekirjastosta tai toteuttaa jonkin valmiin CODESYS SoftPLC:n standardoidusta laitealustasta. [17.]

Laitteiden valmistajat käyttävät Codesysiä toteuttaakseen heidän omia joko ohjelmitavia tai muunneltavia automaatiokomponentteja. Vaihtoehtoihin kuuluu integroidut tuotteet visualisointiin, koordinoitu liikkeenohjaus, yhteys kenttäväylään ja teolliseen internetjärjestelmään. [17.]

4 Modbus

4.1 Yleistä

Modbus on Modiconin vuonna 1979 kehittämä sarjaliikenneprotokolla ohjelmitavia logiikoita varten. Modbus on avoin protokolla, joten eri valmistajat voivat käyttää sitä maksutta laitteidensa kanssa. Siitä on tullut teollisuuden yleisimmin käytetty ja helpoimmin saatavilla oleva protokolla yhdistettäessä teollisuuden elektronisia laitteita. Modbus on Master-Slave järjestelmä, jossa Master-laite ja monet Slave-laitteet, kuten esimerkiksi mittarit ja venttiilit, voidaan liittää yhteen samanaikaisesti. Slave-laitteilla on kaikilla oma yksilöllinen osoitteensa. [18.]

Loogisesti yhdessä Modbus-verkossa voi olla yli 250 laitetta, mutta ei käytännössä, koska RS-485-lähettimek eivät ole tarpeeksi tehokkaita ohjaamaan niin montaa laitetta yhtä aikaa. Käytännössä Modbus-protokolla määrittelee laiterajaksi 32 Slave-laitetta, jota suurin osa RS-485-lähettimeistä tukee. [19.]

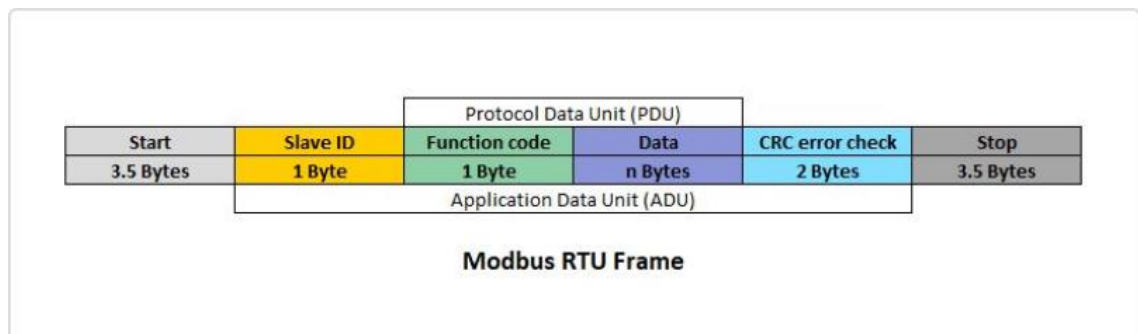
Modbusiin kuuluu kolme kehystä, jotka ovat Modbus ASCII, Modbus RTU ja Modbus TCP/IP. RTU:ta ja ASCII:ta käytetään RS-485-sarjavyölyn päällä ja sen TCP/IP versiota Ethernetin kanssa. [9, 140.] Tässä opinnäytetyössä tarkastellaan pelkästään RTU:ta.

4.2 RTU-kehys

Modbus-viestin RTU-kehys koostuu 6 eri osasta (kuva 4). Se alkaa vähintään 3,5 merkin mittaisella hiljaisella ajanjaksolla, jolloin vastaanottava laite ymmärtää edellisen viestin loppuneen. Tämän jälkeen tulee kahdeksan bitin mittainen osoite, jota seuraa toinen saman pituinen Modbus-funktion komentokoodi, jonka jälkeen tulee data $n \times 8$ bitin mittaisena. [20, 13.]

Sanoman päättää virheetarkastus Cyclic Redundancy Check (CRC). CRC on kahden tavun lisä mahdollisen virheen varalta jokaisen Modbus-viestin lopussa. Jokaista viestin tavua käytetään CRC:n laskemiseen. Myös vastaanottava laite laskee CRC:n ja vertailee sitä lähettävän laitteen CRC:hen. Jos laitteiden välisessä CRC:ssä on edes yhden bitin ero, ovat osapuolten CRC:t erilaiset ja syntyy virhe. [20, 14.]

Sanoman lopetusmerkki on alun tapaan 3,5 merkin hiljainen ajanjakso ja se on myös samalla seuraavan sanoman aloitusmerkki. Edellytys virheetömälle toiminnalle on se, että lähettäessä sanomaa ei saa olla yli 1,5 merkin mittaisia taukoja. [20, 13.]



Kuva 4. Modbus RTU-viestin kehys [21].

4.3 Dokumentaatio

Modbus-dokumentaatio ei ole hyvin standardisoitu. Laitteiden valmistajat dokumentoivat laitteidensa rekisteritietonsa omalla tavallaan. Jotta käyttäjä voisi selvittää, mitä rekisteriä valmistaja todella tarkoittaa, on tehtävä seuraavanlaisia selvitystoimenpiteitä. [19.]

Jos laitteen dokumentaatio ilmoittaa, että haluttu tieto on #1 pitorekisterissä, on se silloin juuri siellä. Kaikkien valmistajien dokumentaatio ei ole kuitenkaan näin selkokielistä. Jos rekisteri on 40001, sijaitsee tieto todellisuudessa pitorekisterissä yksi. Tällainen tiedon esitystapa on peräisin Modiconin ajalta. Kolmas vaihtoehto valmistajilla on kertoa funktiokoodien avulla, mikä rekisteri on kyseessä. Jos tieto ilmoitetaan olevan funktiokoodilla 3 ja 16 rekisterissä yksi, ilmoittaa valmistaja silloin tiedon olevan pitorekisterissä yksi. [19.]

Jos aiemmat selvitystavat eivät tuota tulosta, on selvittävä, tarkoittavatko numerot dokumentaatioissa rekisterinumeroa vai osoitetta. Kenttälaitteen rekisterinumero yksi tarkoittaa kenttälaitteen osoite nollaa, mutta jotkut valmistajat dokumentoivat laitteensa osoitteisiin eikä rekisterinumeroihin. Eli jos kyseessä on osoite, täytyy silloin ilmoitettuun osoitteeseen pitää lisätä yksi, että päästään haluttuun rekisteriin käsiksi. [19.]

4.4 Datan esitystyyli

Tyyli, jolla data esitetään Modbus-viestissä, on peräisin alkuperäisestä standardista ja Modicon logiikan aluetunnuksista. Coil tarkoittaa digitaalista lähtöä tai muistibittiä. Input status tarkoittaa tulobittiä. Register tarkoittaa rekisteriä eli 16-bittistä muistipaikkaa. Rekistereitä on olemassa kahta eri tyyppiä, jotka ovat pito-³ ja tulorekisteri⁴. [22, 11.]

³ Holding register

⁴ Input register

Modbus-protokolla määrittelee rekisterin 16-bittiseksi, mutta datan kirjoittamiseen ja lukemiseen käytetään laajasti laajempaa standardia kuin 16-bittiä. Tyypillisimmät muut datan esitystavat ovat IEE 754 kelluva piste (floating point) ja 32-bittinen kokonaisluku. [19.]

4.5 Yleisimmät funktiokoodit

Modbus-protokollaan on sisällytetty standardoituja funktioita, joilla orjalaitetta voidaan käskä tekemään joku tietty toiminto. Taulukossa 2 on lueteltu yleisimmät funktiokoodit. Funktiokoodit 1–4 ovat lukemiskoodeja ja funktiokoodit 5,6, 15 ja 16 ovat kirjoituskoodeja. Muitakin funktiokoodeja on olemassa, mutta niitä käytetään harvemmin.

Taulukko 2. Yleisimmät funktiokoodit [23].

Decimal	Hexadecimal	Description
01	0x01	Read Coil Status
02	0x02	Read Input Status
03	0x03	Read Holding Registers
04	0x04	Read Internal Registers
05	0x05	Force Single Coil
06	0x06	Preset Single Register
15	0x0F	Force Multiple Coils
16	0x10	Preset Multiple Registers

Koodilla x(01) voidaan lukea yhden orjalaitteessa olevien erillisten digitaalisten lähtöjen ON/OFF tila. Koodilla x(02) tehdään sama asia kuin koodilla x(01), mutta digitaalisille erillisille tuloille [24, 9–10.]

Funktiokoodilla x(03) luetaan orjan pitorekisterien binaarisäily eli lähtö- tai muistisana. Koodilla x(04) tehdään sama asia, mutta kohteena on tulorekisterit esimerkiksi kanavassa vallitseva paine-ero. [24, 11–12.]

Funktiokoodilla x(05) voidaan pakottaa orjalaitteen yhden erillisen lähdön tila ON tai OFF. Koodilla x(06) voidaan asettaa valmiiksi pitorekisteriin joku tietty arvo esimerkiksi pakottaa säätöpelti joko kiinni tai auki. [24, 11–12.]

Koodilla x(15) pakotetaan orjalaitteen jokaisen erillisen lähdön tilaksi joko ON tai OFF. Koodilla x(16) voidaan asettaa valmiiksi arvoja moneen peräkkäiseen pitorekisteriin. [24, 18–19.]

4.6 Yleisimmät virhekoodit

Jos toiminnassa tapahtuu virhe, ilmoittaa järjestelmä silloin siitä virhekoodilla ohjelmassa. Taulukossa 3 on lueteltu yleisimmät virhekoodit. Muitakin virhekoodeja on olemassa, mutta niitä kohtaa harvemmin.

Taulukko 3. Yleisimmät virhekoodit [25].

1	Illegal Function	The function code received in the query is not recognized by the slave or is not allowed by the slave.
2	Illegal Data Address	The data address (register number) received in the query is not an allowed address for the slave, i.e., the register does not exist. If multiple registers were requested, at least one was not permitted.
3	Illegal Data Value	The value contained in the query's data field is not acceptable to the slave.
4	Slave Device Failure	An unrecoverable error occurred while the slave was attempting to perform the requested action
6	Slave Device Busy	The slave is engaged in processing a long-duration command. The master should try again later
10	Gateway Path Unavailable	Specialized use in conjunction with gateways, usually means the gateway is misconfigured or overloaded
11	Gateway Target Device Failed to Respond	Specialized use in conjunction with gateways, indicates no response was received from the target device.

Virhekoodi 1 (Illegal Function) tarkoittaa laitonta funktiota, eli orjalaitteen ei ole sallittua toteuttaa käskettyä funktiokoodia. Se voi myös tarkoittaa, että orjalaite on väärässä tilassa käsitelläkseen haluttua toimintoa. Väärä tila voi tarkoittaa esimerkiksi tilannetta, kun laitetta ei ole konfiguroitu ja sitä pyydetään lähettämään rekisteriarvoja, joita ei vielä ole. [26.]

Virhekoodi 2 (Illegal Data Address) tarkoittaa sitä, että orjalaitteessa ei ole kysyttyä rekisteriosoitetta. Tarkemmin sanottuna referenssinumeron ja siirron pituuden summa ei ole sallittu. Summa saa olla korkeintaan laitteen rekisterien määrän suuruinen. Virhekoodi 3 (Illegal Data Value) on ilmoitus siitä, että kyselyyn sisältyvä datan arvo ei ole sallittu arvo orjalle. Yleensä tämä on ilmoitus monimutkaisen pyynnön rakenteesta olevasta virheestä. [26.]

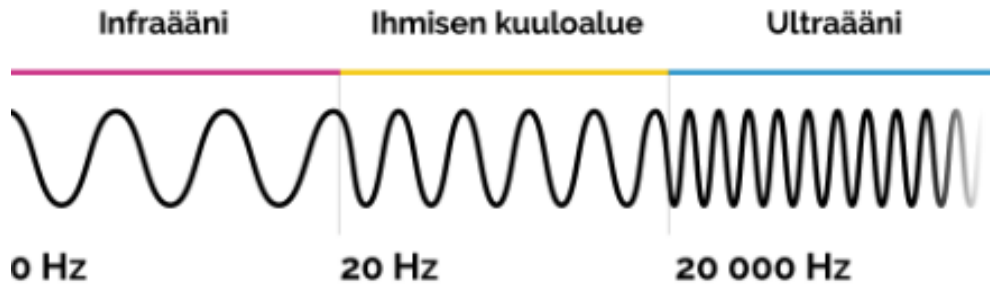
Virhekoodi 4 (Slave Device Failure) on ilmoitus siitä, että orja kohtasi korjaamattoman virheen, kun orja yritti suorittaa pyydetyn toiminnan. Virhekoodi 6 (Slave Device Busy) eli orjalaitteella on edellisen pyynnön suoritus kesken ja se ei kerkeä vastaamaan kysyttyyn pyyntöön. Orjalaite voi suorittaa vain yhtä pyyntöä kerrallaan. [26.]

Virhekoodi 10 (Gateway Path Unavailable) tarkoittaa yleensä, että yhdyskäytävä on määritelty väärin tai sitä on kuormitettu liikaa. Virhekoodi 11 (Gateway Target Device Failed To Respond) tarkoittaa tilannetta, jossa kohdelaitteesta ei saatu vastausta kysyttäessä. Se tarkoittaa yleensä, että laitetta ei ole edes tietyssä verkossa. [26.]

5 Ultraääneen perustuva ilmamääräsäädin

5.1 Ultraääni ja ilmamääräsäädin

Ultraääni on ääntä, joka ylittää ihmisen kuuloalueen 20 MHz rajan (kuva 5). Se on samanlaista ääntä kuin esimerkiksi normaali puheenääni, joten se tarvitsee väliaineen liikkuaakseen eteenpäin. Ultraääni etenee hyvin ilmassa, nesteessä ja kiinteässä aineessa, mutta huonosti kaasussa. [27.]



Kuva 5. Äänen aaltomalli [27].

Ilmamääräsäädin (IMS) on laite, jota käytetään ilman tilavuusvirran mittaamiseen ja säätämiseen [28]. Ilmamääräsäätimiä on olemassa kahdenlaisia: Fyysiseen anturiin perustuva mittaus ja uudempaa tekniikkaa edustava tarkempi ultraäänimittaus, jota tutkitaan tässä opinnäytetyössä.

Uutta tekniikkaa oli tarjolla kahdelta valmistajalta, joita olivat FläktGroup ja Lindab. FläktGroupin tuote on Optivent Ultra® (kuva 6) ja Lindabin Ultralink® FTCU (kuva 7). Tiedonsiirron standardina molemmissa laitteissa käytetään Modbusia, mutta laitteita voidaan myös ohjata analogisesti jänniteviestillä.



Kuva 6. Optivent Ultra® [29].



Kuva 7. Lindab Ultralink® [30, 1].

5.2 FläktGroup Optivent® Ultra

Optivent Ultra® on teknologian tutkimuskeskuksen VTT:n ja Fläkt Woods Oy:n kehittämä uutta tekniikkaa edustava virtaussäädin. Erillisiä säätöpeltejä ei

myöskään tarvita. Uusi innovaatio pitää sisällään ultraäänimittauksen sekä myös ilmavirransäätimen. Virtaussäätimen avulla haluttuun huonetilaan pystytään saamaan täsmälleen oikea ilmamäärä tilan käyttöasteen mukaan, mistä on erittäin paljon hyötyä. Se muun muassa säästää resursseja, parantaa sisäilmailman laatua huomattavasti ja on erittäin energiatehokas ympäristöystävällinen ratkaisu. Uuden virtaussäätimen edut tulevat siitä, että ilmanvaihtojärjestelmässä ei tarvitse olla erillisiä mittalaitteita, jotka muun muassa pölyntyvät helposti, ovat äänekkäitä, tarvitsevat huoltoa säännöllisesti tai ovat tilaa vieviä. [31.]

Virtaussäätimen toimintaperiaate on (kuva 8) on se, että lähettimen signaali lähtee 90 asteen kulmassa ilman virtaussuunnasta katsottuna. Ilman virtaus taivuttaa signaalia. ”Ilman virtaus saa aikaan signaalin taipuman, joka on lähes täysin lineaarisesti riippuvainen ilman virtausnopeudesta.” Vastaanottimilla mitatusta signaalin vaihe-erosta saadaan ratkaistua ilmavirta ja sen nopeus. Signaalin kulkiessa yli koko kanavan häiriölähteiden kuten mm. T-haaran tai mutkan aiheuttama häiriötä ei tarvitse ottaa huomioon. [31.]



Kuva 8. Optivent® Ultran ultraäänitekniikka [32].

5.3 Lindab Ultralink® FTCU

Lindab Ultralink® FTCU on Lindabin versio ultraäänellä toimivasta ilmamääräsäätimestä. Sen toimintaperiaate on samanlainen kuin kilpailijallansa Fläktgroupilla. Sen mittausalue (0,5–20 m/s), ja se pystyy myös mittaamaan ilmavirran nopeudella 0,1 m/s alle 10 % epätarkkuudella. [1.]

Laitteesta on myös olemassa bluetooth-versio, mitä voi ohjata langattomasti älylaitteella. Laitteeseen on myös integroitu Belimon peltimoottori, eli erillistä peltimoottoria ei tarvita. [1.]

6 Soveltava harjoitustyö

6.1 Järjestelmän toimintaperiaate

Kuvassa 9 on nähtävillä järjestelmä, jonka ohjelma manuaalisesti aukaisee, tai sulkee säätöpellin ilmanlaatulähtetimen mittaamien arvojen tai ilmavirran perusteella. Ensiksi ilmanlaatulähtetin mittaa tilan tiedot, sen jälkeen ilmamääräsäädin mittaa läpi menevän ilmavirran ja pellin asennon. Arvojen lukemisen jälkeen laitetaan säätöpellin ohjaus päälle. Ohjaus joko aukaisee, tai sulkee pellin kokonaan asetettujen parametrien mukaan. Pakko-ohjausta ei kuitenkaan tapahdu, jos kumpikaan ehto ei täyty.



Kuva 9. Harjoitustyön järjestelmä.

6.2 Työssä käytetyt laitteet

Työssä käytettiin kahta analogista Ventur TD-160/100 Ecowatt puhallinta (kuva 10). Suoraan puhaltimiin oli yhdistetty molemmat ultraääni-ilmamääränsäätimet. Puhaltimen yksi jälkeen tuli Optivent Ultra ja puhaltimen kaksi jälkeen Ultralink FTCU. Järjestelmään kuului myös Proqualin ilmanlaatulähetin ILH-M-RH-N (kuva 11), jolla mitattiin tilan CO₂-pitoisuutta⁵, lämpötilaa⁶ ja suhteellista kosteutta⁷. Kaikki kentälaitteet paitsi puhaltimet olivat yhdistetty erilliseen 24 V:n jännitelähteeseen, koska logiikan virtalähde ei riittänyt tuottamaan kaikille kentälaitteille tarpeeksi virtaa.

Työn PLC:nä toimi Wagon 750–881. Siihen oli kytketty analoginen lähtökortti 750–559, johon liitettiin molemmat puhaltimet. Lähtökortin jälkeen logiikassa oli

⁵ ppm

⁶ Celsius

⁷ RH %

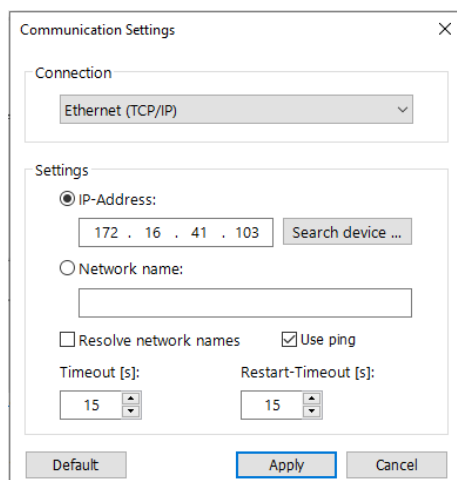
RS-485-sarjaliikennekortti 750–653/003. Ilmanlaatulähetintä sekä molempia ilmamääränsäätimiä ohjattiin Modbus-väylällä, joten ne kaikki liitettiin RS-485-sarjaliikennekorttiin. Viimeisenä logiikassa piti olla vielä end-moduuli.



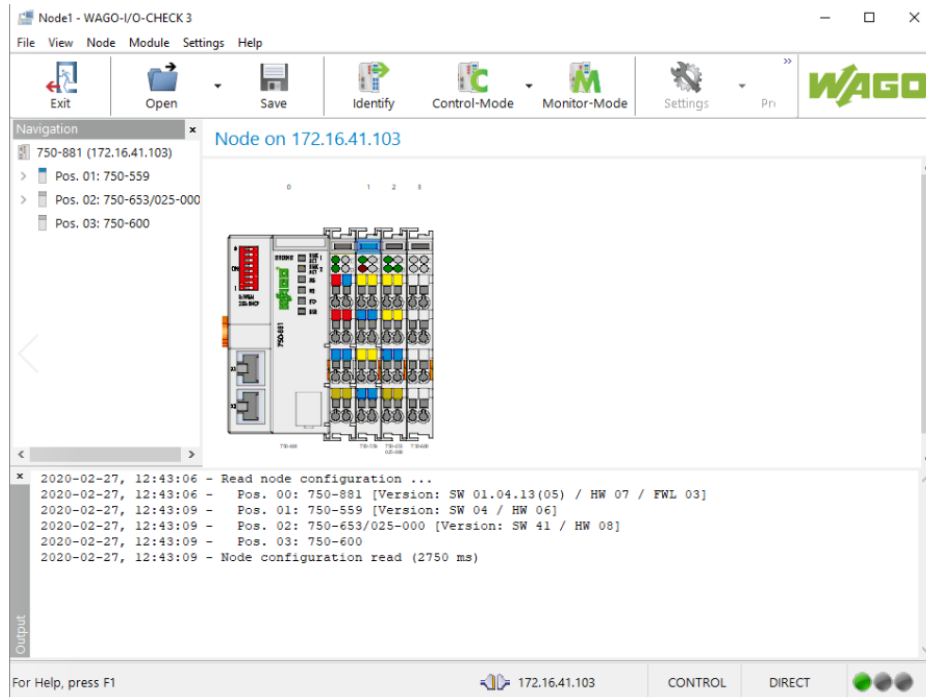
Kuva 10. Ventur TD Ecowatt 160/100 [33]. Kuva 11. Produal ILH-M-RH-N [34].

6.3 Harjoitustyön tiedonsiirto

Tietokoneen ja logiikan välinen tiedonsiirto toteutettiin Ethernetillä, koska Modbus-ohjelmat ovat isoja, jopa yli 30 Mbit. Logiikan yhdistäminen laboratorion experimental-verkkoon oli helppoa. Ensiksi painettiin Wago I/O Check 3 ohjelman Identify-painiketta ja laitettiin Communication Settings -valintaikkunaan kuvassa 12 näkyvät määrittelyt. Jos yhteydenhaku onnistui oikein, löysi tietokone logiikan laboratorion experimental verkosta (kuva 13).

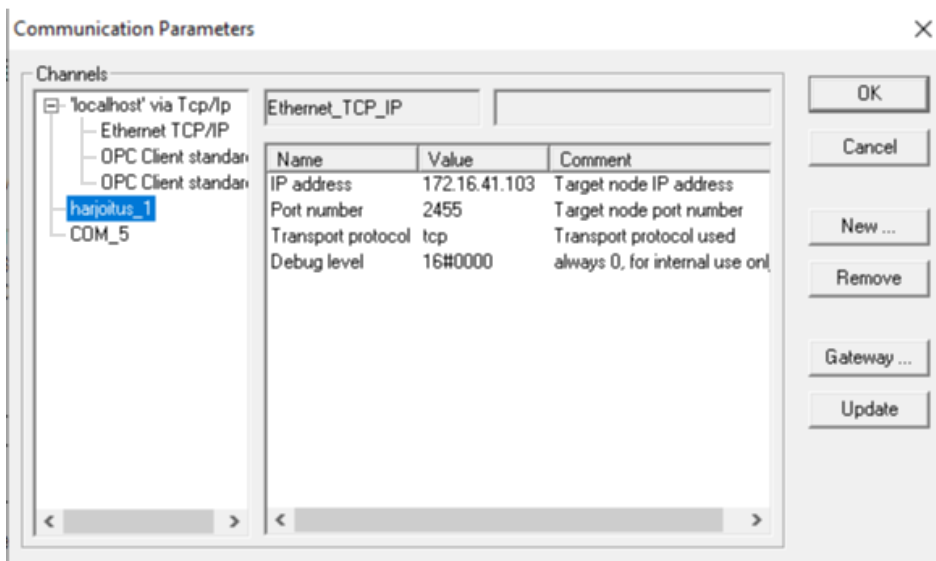


Kuva 12. Yhteysasetusten määrittely.



Kuva 13. Onnistunut yhteys.

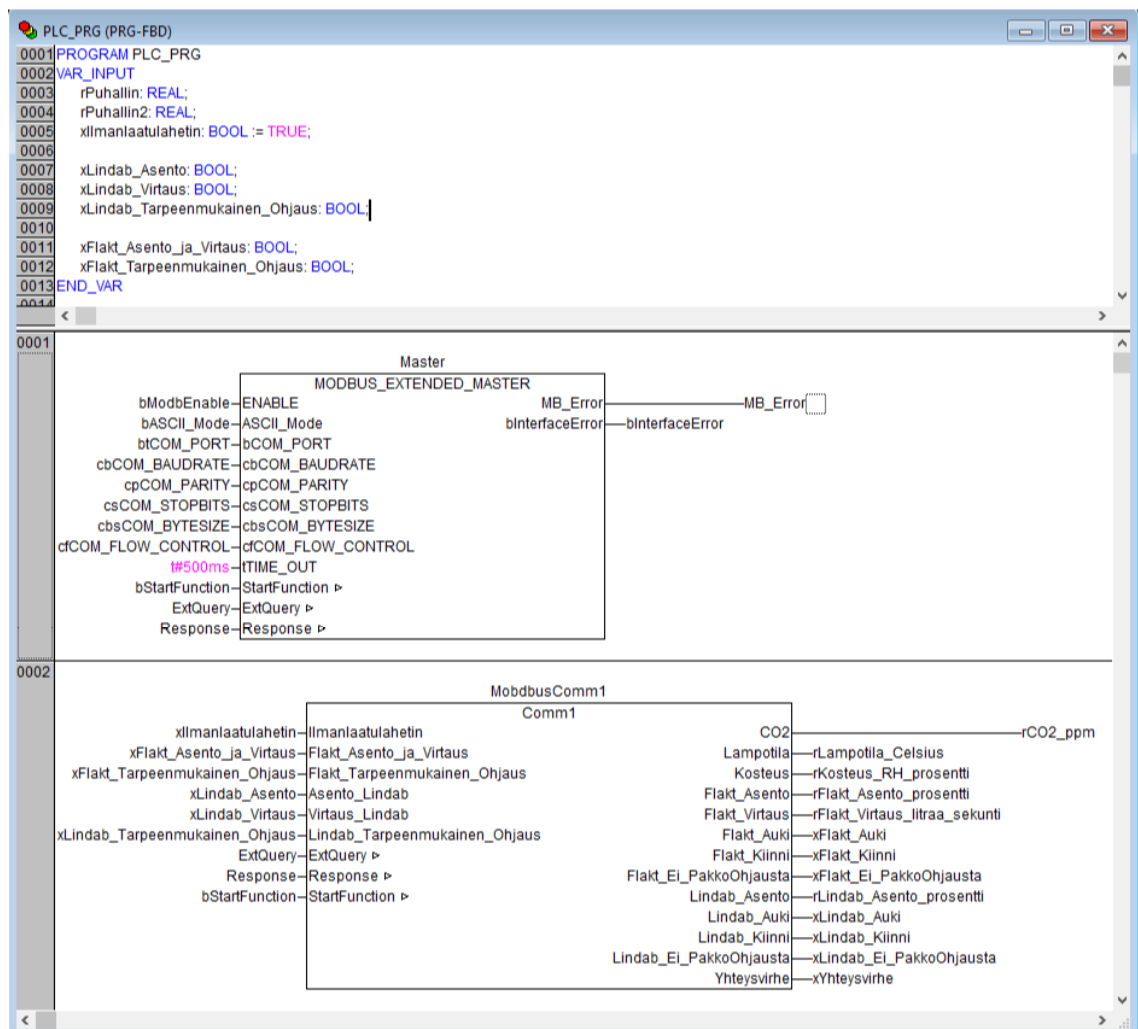
Tämän jälkeen täytyi vielä Codesysin Communication Parameters -alavalikosta käydä määrittämässä yhteystavaksi Ethernet ja kirjoittaa osoitekenttään logiikan osoite (kuva 14). Näiden määrittelyn jälkeen tietokoneen ja logiikan välinen Ethernet-yhteys alkoi toimia.



Kuva 14. Communication Parameters.

6.4 Pääohjelma

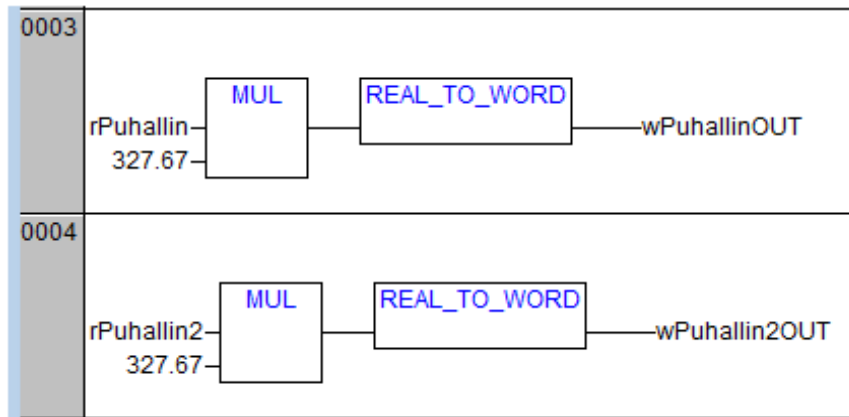
Pääohjelmassa⁸ (kuva 15) on kolme eri kokonaisuutta: Master-laitteen konfiguraatio, aliohjelma⁹ ja puhaltimien ohjausrivit, jotka ovat nähtävissä kuvassa 16. Ohjelman eri muuttujat ja osat ovat jaettu selkeyden vuoksi niin, että puhallinohjaukset ja aliohjelman tulot ovat pääohjelmassa. Aliohjelman lähdöt ovat globaaleissa muuttujissa (kuva 17). Myös Modbus Master -laitteen konfiguraatioasetukset ovat globaaleissa muuttujissa ohjelman selkeyden ja hallittavuuden vuoksi.



Kuva 15. Pääohjelman rakenne.

⁸ PLC_PRG

⁹ Comm1



Kuva 16. Puhaltimien ohjausrivit.

```

Global_Variables
0001 VAR_GLOBAL
0002   ModbusComm1: Comm1;
0003   xYhteysvirhe: BOOL;
0004   (*Ilmanlaatulähetin*)
0005   rCO2_ppm: REAL;
0006   rLampotila_Celsius: REAL;
0007   rKosteus_RH_prosentti: REAL;
0008   (*Fläkt*)
0009   rFlakt_Aseno_prosentti: REAL;
0010   rFlakt_Virtaus_litraa_sekunti: REAL;
0011   xFlakt_Auki: BOOL;
0012   xFlakt_Kiinni: BOOL;
0013   xFlakt_Ei_PakkoOhjausta: BOOL;
0014   (*Lindab*)
0015   rLindab_Aseno_prosentti: REAL;
0016   xLindab_Auki: BOOL;
0017   xLindab_Kiinni: BOOL;
0018   xLindab_Ei_PakkoOhjausta: BOOL;
0019   (*Modbus Master*)
0020   Master: MODBUS_EXTENDED_MASTER;
0021   bModbEnable: BOOL := TRUE;
0022   bASCII_Mode: BOOL := FALSE;
0023   btCOM_PORT: BYTE := 2;
0024   cbCOM_BAUDRATE: COM_BAUDRATE := 960;
0025   cpCOM_PARITY: COM_PARITY := 1;
0026   csCOM_STOPBITS: COM_STOPBITS := 1;
0027   cbsCOM_BYTESIZE: COM_BYTESIZE := 8;
0028   cfCOM_FLOW_CONTROL: COM_FLOW_CONTROL := 4;
0029   bStartFunction: BOOL := FALSE;
0030   ExtQuery: typModbusExtendedQuery;
0031   Response: typModbusResponse;
0032   MB_Error: enumMB_ERROR;
0033   bInterfaceError: BYTE;
0034 END_VAR

```

Kuva 17. Globaalit muuttujat

6.5 Alkumääritykset

Ohjelman logiikkamäärittelyissä valittiin työssä käytettävä logiikka 750–881. POU:ksi (Program Organization Unit) valittiin Program ja POU:n ohjelmointikieleksi FBD (Function Block Diagram). Seuraavaksi määritettiin työssä käytettävät IO-kortit Resources välilehden PLC-Configuration alavalikosta. IO-kortit olivat analoginen lähtökortti 750–559 ja RS-485-sarjaliikennekortti 750–653/000–25. 750–559 lähtökortin ensimmäiseen ja toiseen kanavaan määritettiin molempien puhaltimien ohjauksen globaalit muuttujat¹⁰, jotka olivat molemmat tyyppiä WORD. Muuttujien eteen kannattaa laittaa pieni kirjain kertomaan, mitä tyyppiä kukin muuttuja on, koska se selkeyttää ohjelmointia. Huomioitavaa oli valita täsmälleen samat IO-kortit kuin mitä logiikkaan oli kytketty, koska muuten ohjelmassa tapahtui sisäinen käyttöliittymävirhe¹¹.

Puhallinohjauksille täytyi tehdä ohjelmarivit. Molempien puhaltimien ohjauksille tehtiin muuttujat¹², joilla voi ohjata puhaltimien pyörimisnopeutta. Ensimmäinen muuttuja oli puhaltimen yksi ohjaukseen ja toinen muuttuja puhaltimen kaksi ohjaukseen. Puhaltimia haluttiin ohjata prosenttiasteikolla ja niiden ohjaussignaali oli analoginen 0–10 V, datatyyppiä WORD. WORD on 16-bittinen datatyyppi, joka voi olla joko luonnollinen luku tai kokonaisluku. Puhaltimen ohjaussignaali oli +0-10V, joten signaali piti kertoa luvulla 327.67, että saataisiin luku 32767, joka on luonnollisen luvun maksimiarvo (yhtälö 1). Yksi bitti menee etumerkin kertomiseen, jonka takia luvusta vähennetään yksi. Saatu luku piti myös konvertoida tyypistä REAL muotoon WORD, että puhallinohjaus toimii oikein.

$$2^{15} - 1 = 32767$$

Yhtälö 1. Etumerkillisen WORD-datatyypin luonnollinen maksimiarvo.

¹⁰ wPuhallinOUT ja wPuhallin2OUT

¹¹ bInterfaceError = 15

¹² rPuhallin ja rPuhallin2

6.6 Modbus Master

Modbus Master on järjestelmän isäntälaitte, joka kääntää orjalaitteita toimimaan käyttäjän haluamalla tavalla. Master-laitteen konfiguraatio ja parametrien asetus pitää tehdä ohjelmaan, että järjestelmällä on asetukset, joiden mukaan toimia. Master-laitteen tulot ja lähdöt löytyvät Modbus-ohjelmakirjastosta, jota ei ole valmiiksi mukana Codesysissä, vaan se pitää ladata eri lähteestä erikseen. Kirjasto lisättiin mukaan ohjelmaan Resources-välilehden Library Manager -alavalikosta. Seuraavaksi tehtiin ensimmäiselle ohjelmariville boksi ja kysymysmerkkien päällä F2 painamalla käytiin hakemassa Modbus Master -laitteen graafinen ilmentymä FBD-muodossa [22, 19.].

Seuraavaksi määritettiin Master-laitteen tuloille ja lähdöille ohjelmalliset muuttujat. Tarvittavat muuttujatyypit löytyvät F2 painamalla kysymysmerkkien päällä, jolloin päästään muuttujavalikkoon. Kuvasta 18 nähdään, miten muuttujat pitää määrittellä ja alustaa, että harjoitustyön Master-laitteen konfiguraatio toimii oikein.

```

0017 (*Modbus Master*)
0018 Master: MODBUS_EXTENDED_MASTER;
0019 bModbEnable: BOOL := TRUE;
0020 bASCII_Mode: BOOL := FALSE;
0021 btCOM_PORT: BYTE := 2;
0022 cbCOM_BAUDRATE: COM_BAUDRATE := 960;
0023 cpCOM_PARITY: COM_PARITY := 1;
0024 csCOM_STOPBITS: COM_STOPBITS := 1;
0025 cbsCOM_BYTESIZE: COM_BYTESIZE := 8;
0026 cfCOM_FLOW_CONTROL: COM_FLOW_CONTROL := 4;
0027 bStartFunction: BOOL := FALSE;
0028 ExtQuery: typModbusExtendedQuery;
0029 Response: typModbusResponse;
0030 MB_Error: enumMB_ERROR;
0031 bInterfaceError: BYTE;
0032 END_VAR

```

Kuva 18. Modbus Master muuttujien määrittely.

Jotta Modbus-tiedonsiirto toimii oikein, täytyy järjestelmälle määritellä parametrit, jotka täytyvät olla samat jokaisessa järjestelmään liitettävissä laitteissa. Parametreja ovat siirtonopeus¹³, pariteetti¹⁴ ja pysäytysbitit¹⁵. Siirtonopeudeksi asetettiin 9600, jota vastaa ohjelmassa 960. Pariteetiksi asetettiin 1 eli odd ja pysäytysbiteiksi 1.

Muiden Master-laitteen konfiguraation muuttujien merkitys on seuraava: Modbus-toiminnallisuuden käynnistää sen muuttuja¹⁶. Modbus-protokollan voi myös halutessaan aktivoida toimimaan ASCII-sanomatilassa¹⁷. Sarjakorteille pitää myös määrittää liikenneportti¹⁸, sanoman tavukoko¹⁹ täytyi myös määrittää ja Master-laitteelle täytyy lisäksi kertoa sen viestintätila²⁰, jonka arvo 4 vastaa half-duplexia. Lopuksi Modbusin kysely-kuuntelu toiminto käynnistyy sen käynnistysmuuttujan²¹ arvon ollessa tosi. [22, 20.] Tarkemmat ohjeet ovat liitteessä 1.

6.7 Extquery

Extquery on Master-laitteen lähettämä kysely tietylle kenttälaitteelle. Siinä määritellään kuusi eri käskyn osaa (kuva 19). Ensimmäiseksi kenttälaitteen osoite²². On myös kerrottava, mitä toimintoa²³ Slaven halutaan tekevän. Slaven rekistereistä voidaan joko lukea dataa tai kirjoittaa. Jos halutaan lukea dataa, on määriteltävä ensimmäinen luettava rekisteri²⁴ ja luettavien rekisterien määrä²⁵. Kirjoitettaessa rekistereihin on samalla tavalla määritettävä ensimmäinen

¹³ BAUDRATE

¹⁴ PARITY

¹⁵ STOPBITS

¹⁶ bModbEnable

¹⁷ bASCII_Mode

¹⁸ BtCOM_PORT

¹⁹ cbsCOM_BYTESIZE

²⁰ COM_FLOW_CONTROL

²¹ bStartFunction

²² SlaveAddress

²³ FunctionCode

²⁴ Read_StartAddress

²⁵ Read_Quantity

kirjoitettava rekisteri²⁶ ja niiden määrä²⁷. Kirjoitettaessa rekisteriin dataa pitää myös kertoa, mihin rekisteriin kirjoitetaan ja siihen kirjoitettava jokin tietty arvo²⁸.

ExtQuery	typModbusExtendedQuery	<pre> TYPE typModbusExtendedQuery : STRUCT SlaveAddress : BYTE; FunctionCode : BYTE; Read_StartAddress : UINT; Read_Quantity : UINT; Write_StartAddress: UINT; Write_Quantity : UINT; Write_Data : ARRAY[0..124] OF WORD; END_STRUCT END_TYPE </pre>
----------	------------------------	--

Kuva 19. Extquery-pyyntökäskyn osat [35, 9].

Huomioitavaa on, että laitteen datalehdessä on tarkistettava, että rekisterit ovat olemassa kyseisessä laitteessa, ja että niiden luettava tai kirjoitettava määrä ei ole liian suuri. Esimerkkinä jos laitteen toisessa rekisteristä viidenteen on olemassa dataa, niin että ei aseteta ensimmäistä kirjoitettavaa rekisteriä nollassi tai yritetä sisällyttää lukea myös rekisteriä kuusi, jossa ei sijaitse mitään luettavaa dataa. Seurauksena näistä toimista syntyy Slave-laitteen virhe²⁹, koska yritettiin tehdä jotain sellaista, mitä kyseisessä laitteessa ei ole mahdollista tehdä.

6.8 Response

Response (kuva 20) on Slave-laitteen lähettämä vastaus isännälle toiminnon onnistuessa. Tiedonlähetystoiminnolla³⁰ voidaan valita, mistä rekisteristä halutaan lukea dataa. Vastauksessa näkyy samat arvot kuin kyselyssä, jos toiminto on onnistunut. Virheen³¹ ollessa jotakin muuta kuin nolla, on tapahtunut tiedonsiirrossa jokin virhe. Vastauksessa kaikkien arvojen ollessa nolliä kysely ei

²⁶ Write_StartAddress

²⁷ Write_Quantity

²⁸ Write_Data

²⁹ EXTENDED_SLAVE_ERROR

³⁰ Response.Data

³¹ Response.Error

ole edes mennyt perille Slavelle asti. Arvojen ollessa nolliä on syytä tarkastaa ovatko kytkennät tehty oikein. Jos ne ovat tehty oikein, täytyy seuraavaksi tarkastaa itse koodista, jääkö ohjelma niin sanotusti jumiin jossain vaiheessa. Ohjelman ollessa päällä aliohjelmasta voi tarkastaa ohjelman jäädessä jumiin, että missä vaiheessa jumiutuminen tapahtuu. Kuvasta 21 voidaan nähdä esimerkki, kuinka tiedot näkyvät aliohjelmassa sen ollessa käynnissä, kun tiedonsiirto toimii oikein.

Response	typModbusResponse	<pre> TYPE typModbusResponse : STRUCT Error : WORD; SlaveAddress : BYTE; FunctionCode : BYTE; StartAddress : UINT; Quantity : UINT; Data : ARRAY[0..124] OF WORD; END_STRUCT END_TYPE </pre>
----------	-------------------	--

Kuva 20. Slave-laitteen vastaus [35, 9].

```

0001 IF NOT StartFunction THEN
0002
0003   IF Huoneanturi = TRUE THEN
0004
0005     ExtQuery.SlaveAddress := 2;
0006     ExtQuery.FunctionCode := 4;
0007     ExtQuery.Read_StartAddress := 0;
0008     ExtQuery.Read_Quantity := 3;
0009     ExtQuery.Write_StartAddress := 0;
0010     ExtQuery.Write_Quantity := 0;
0011
0012     IF Response.Error = 0 THEN
0013       IF StartFunction = FALSE THEN
0014
0015         CO2 := Response.Data[0];
0016         Lampotila := (Response.Data[1]/10);
0017         Kosteus := Response.Data[2];
0018
0019       END_IF
0020
0021     ELSE Yhteysvirhe := TRUE;
0022
0023     END_IF

```

StartFunction = TRUE

Huoneanturi = TRUE

ExtQuery.SlaveAddress = 2
 ExtQuery.FunctionCode = 4
 ExtQuery.Read_StartAddress = 0
 ExtQuery.Read_Quantity = 3
 ExtQuery.Write_StartAddress = 0
 ExtQuery.Write_Quantity = 0

Response.Error = 0
 StartFunction = TRUE

CO2 = 452 Response.Data[0] = 452
 Lampotila = 24 Response.Data[1] = 241
 Kosteus = 19 Response.Data[2] = 19

Yhteysvirhe = FALSE

Kuva 21. Toimiva aliohjelma.

6.9 Kenttälaitteiden parametrisointi

6.9.1 Ilmanlaatulähetin ILH-M-RH-N

Ilmanlaatulähettimen parametrien asettamiseen käytetään erillistä Produalin ML-SER Modbus-käyttöönottotyökalua (kuva 22), joka kytketään lähettimen sisällä olevaan väylään. Kannen saa auki esimerkiksi ruuvimeisselillä vääntämällä. Väylään kytkettäessä laite näyttää lähettimen asetukset, ja Communication-välilehdeltä päästään käsiksi tiedonsiirtoasetuksiin.



Kuva 22. Produal ML-SER käyttöönottotyökalu [36].

6.9.2 FläktGroupin ultraääni-ilmamääräsäädin

Fläktin laitteen saadessa virtaa se suoritti ensiksi alustustoiminnon, jonka kesto oli noin minuutin. Laitteen parametrien asettaminen tapahtui kahta säätöruuvia pyörittämällä, joiden käyttöohjeet näkyvät kuvassa 23. Alempi ruuveista on toiminnonvalitsin ja ylempi arvonvalitsin (kuva 24). Adr-kohdassa valittiin ylemmällä ruuvilla laitteen osoitteeksi 3 ja alavalikkoon siirryttäessä numero 3 vastasi tässä tapauksessa harjoitustyössä käytettyjä tiedonsiirtoparametreja. Lopuksi siirrettäessä toiminnonvalitsin takaisin flow-kohtaan, päästiin takaisin alkunäkymään. Laitteessa on myös muita toimintoja esimerkiksi asetusarvon valinta, joiden asettamisohjeet näkyvät liitteessä 2. Huomioitavaa on, että virta piti aina katkaista, jotta tehdyt muutokset tulivat voimaan.

MODBUS-ASETUKSET

Osoite on valittavissa välillä 1–247. Osoite voidaan valita arvonvalitsimella, kun toiminnonvalitsin on Adr-asennossa.

Yhteysasetuksia voidaan muuttaa asettamalla ensin toiminnonvalitsin Adr-asentoon ja kääntämällä sen jälkeen arvonvalitsinta, kunnes näyttöön tulee "2In", jolloin siirrytään alavalikkoon. Tästä on näytöllä merkinä pieni ympyrä.

Liikkuminen Modbus-alavalikossa

Kun toiminnonvalitsin on Adr-asennossa, voidaan tehdä seuraavat yhteysasetukset:

Numero näytössä	EEPROM-arvo	Baudinopeus	Pariteetti	Stop-bitit
1	0	9600	ei	2
2	1	9600	parillinen	1
3	2	9600	pariton	1
4	3	19200	ei	2
5	4	19200	parillinen	1
6	5	19200	pariton	1
7	6	38400	ei	2
8	7	38400	parillinen	1
9	8	38400	pariton	1
10	9	9600	ei	1
11	10	19200	ei	1
12	11	38400	ei	1

Huom: FläktGroup suosittelee asetusten 1–9 käyttöä.

Kuva 23. Modbus-asetusten vaihtamisen käyttöohjeet [37, 10].

KÄYTTÖLIITTYMÄ



TOIMINNOT, YKSITYISKOHTAINEN SELOSTUS

1. Arvonvalitsin

Arvonvalitsimella voidaan muuttaa asetettuja arvoja. Muutos näkyy näytöllä, kun valitsinta on liikutettu $\pm 10^\circ$ asennostaan.

2. Toiminnonvalitsin

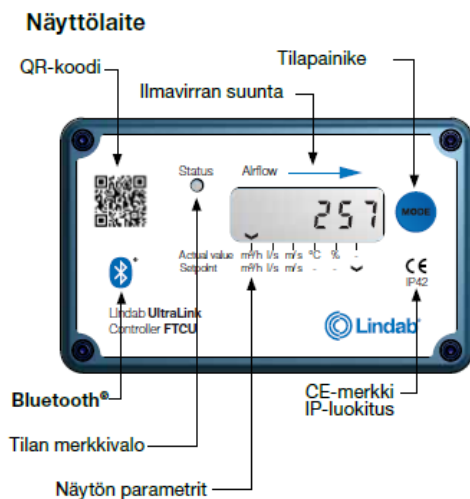
Toiminnonvalitsimella voidaan valita eri toimintoja. Jos mitään toimintoa ei ole valittuna, näytöllä näkyy kolme viivaa (- - -).



Kuva 24. Näyttöpaneeli ja sen käyttöohjeet [37, 3].

6.9.3 Lindabin ultraääni-ilmamääräsäädin

Lindabin laitteen parametrien asettaminen tapahtui näyttölaitteen (kuva 25) tilapainiketta joko lyhyesti tai pitkään painamalla (kuva 26). Ensin painettiin pitkään tilapainiketta, että päästiin alavalikkoon. Seuraavaksi Modbus-väylän tiedonsiirtoasetuskohdassa painettiin uudestaan pitkään nappia, jolloin päästiin muokkaamaan Modbus tiedonsiirtoasetuksia. Haluttiin ohjata laitetta Modbus-väylällä, joten protokollaksi laitettiin Modbus, tiedonsiirtoasetukset harjoitustyön parametrien mukaan ja laitteen osoitteeksi 4. Laitetta ohjattiin väylällä, joten ohjaustavaksi valittiin väyläohjaus. Muita kohtia ei tarvinnut tässä työssä muuttaa. Jotta muutokset tulivat voimaan, piti painaa ensin pitkään nappia Store-kohdassa ja tämän jälkeen katkaista virta. Laitteessa on myös muita toimintoja esimerkiksi asetusarvon valinta, jotka näkyvät liitteessä 4.



Kuva 25. Näyttölaite [30, 3].

Näytön määrittämisvalikon rakenne

Määrittämisvalikko aktivoidaan painamalla näytönläppäpainiketta pitkään (5 sek.). Kun painiketta painetaan pitkään, esiin tulee uusi valikko, jossa on kolme eri vaihtoehtoa:

- Con.Set (Tiedonsiirtoasetukset modbus-väylälle)
- Aln.Set (Analogisen ohjaustavan asetukset)
- Cancel (Peruuta ja palaa infonäyttöön)

Voit selata kolmea vaihtoehtoa painamalla painiketta lyhyesti. Valitse haluamasi vaihtoehto ja siirry valikkorakenteessa alaspäin painamalla pitkään.

Kohdassa Con.Set (väyläasetukset) on käytössä seuraavat vaihtoehdot (selaa painamalla lyhyesti, valitse painamalla pitkään):

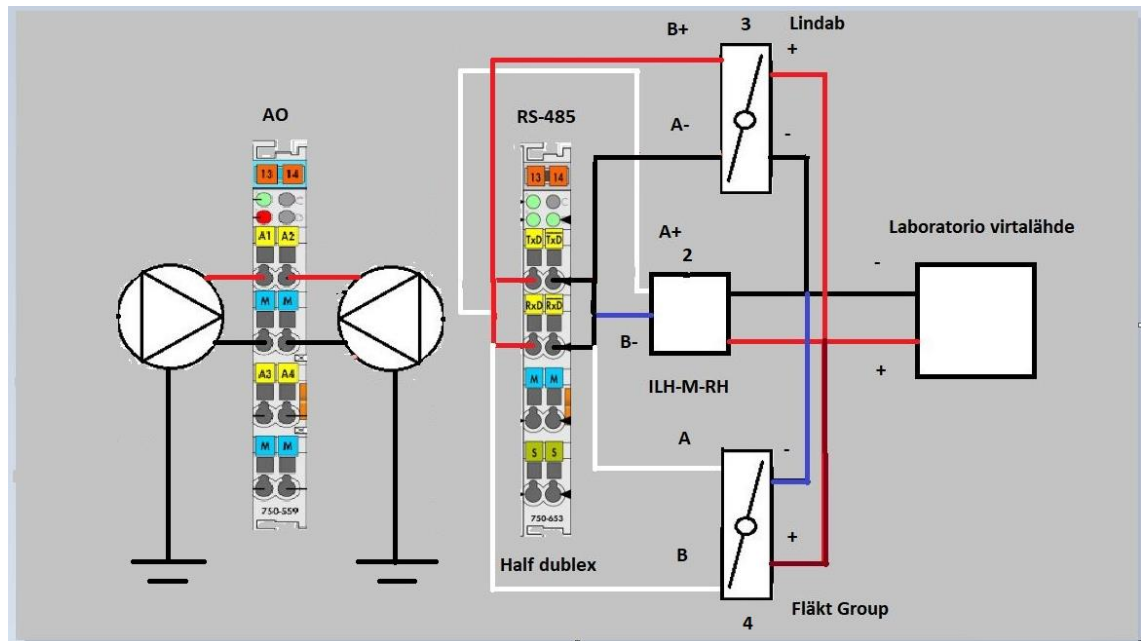
Valikko-merkki	Asetus	Vaihtoehdot	Kuvaus
• Pr.	Protokolla	Pr.PAS Pr.Mod	Pascal-protokolla Modbus
• b.	Baudinopeus	b.9600 b.19200 b.38400 b.76800	Baudinopeus 9600 Baudinopeus 19200 Baudinopeus 38400 Baudinopeus 76800
• bit.	Pysäytysbitit	bit.1 bit.2	1 pysäytysbitti 2 pysäytysbittiiä
• P.	Pariteetti	P.odd P.even P.none	Pariton pariteetti Parillinen pariteetti Pariteetti, ei ole
• Id.	Modbus-osoite	Id.x	Modbus-id (x = arvo) *)
• PLA.	PLA osoite Pascalille	PLA.x	PLA-osoite (x = arvo) *)
• ELA.	ELA osoite Pascalille	ELA.x	ELA-osoite (x = arvo) *)
• Pi.	Pin-koodi	Pi.1111	Oletus = 1111
• Cnt..	Ohjaustapa	Cnt.bus Cnt.Aln	Ohjaus väylältä Ohjaus analogiatulolla
• Store	Tallenna muutokset		Tallentaa muutokset pitkään painettaessa
• Cancel	Peruuta		Peruuta tai ohita muutokset pitkään painamalla

Kuva 26. Lindab IMS valikko-ohjeet [30, 10].

6.10 Kytkennät

Työssä haluttiin saada tiedonsiirto kulkemaan RS-485-sarjaväylä kaksisuuntaisesti eli half-dublexina. Se tapahtui seuraavalla tavalla. Tiedonsiirtomodulin 750-653 TxD + ja – kytketään keskenään yhteen ja samalla tavalla myös RxD + ja -. Molemmat ultraääni IMS:sit ja puhaltimet ja ilmanlaatulähetin kytketään logiikkaan laitteen datalehtien osoittamalla tavalla. Optivent Ultram datalehdessä ei ollut esitetty, kummin päin A ja B piti kytkeä, josta aiheutui ylimääräistä päänvaivaa.

Koko järjestelmän kytkentäkaavio on esitetty kuvassa 27. Kytkentäkaavion johdotukset ovat piirretty johtojen todellisen värin mukaan ja numerot laitteiden päällä tarkoittavat laitteen osoitetta. Fläktin IMS:sin logiikkaan liitettävät johdot olivat laitteen ulkopuolella nipussa, kun taas Lindabin vastineessa johtimien kytkentäpaikat olivat laitteen sisällä, jonka kansi oli kiinni torks-ruuveilla.



Kuva 27. Järjestelmän kytkentäkaavio.

7 Aliohjelma

7.1 Tehtävä ja rakenne

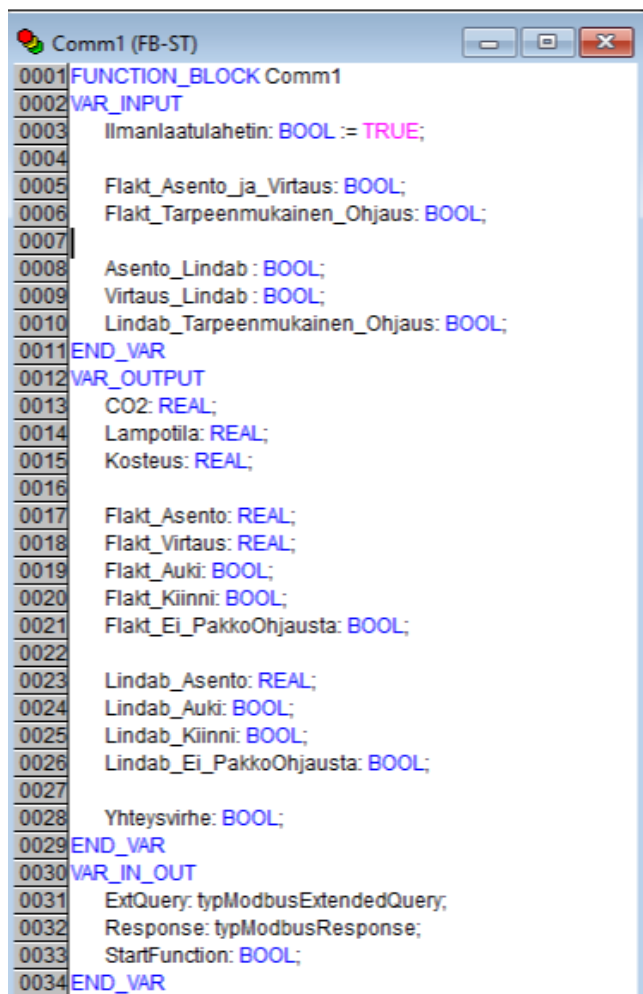
Aliohjelman³² eli Modbus-toiminnallisuuden tarkoitus on taata ohjelman jatkuva suoritus. Modbus-pyyntökäsky ei osaa vaihtaa itsensä osia ja suorittaa kyselyn vain silloin, kun kysely-kuuntelu toiminnon käynnistäjä saa arvon tosi. Pyyntökäskyä on siis autettava, että jatkuva suorittaminen onnistuu. [22, 23.] Tämän vuoksi tehtiin funktioblokki ST-kielellä (Structured Text), joka on tekstipohjainen ohjelmointikieli.

Aliohjelma on jaettu neljään eri toiminnallisuuteen, joiden tulo saadessaan arvon tosi, käynnistää Master-laitteen kyselyn tietyille Slave-laitteelle. Ehtolauseet ovat ilmanlaatulähtetimen tietojen lukeminen, molempien ilmamääräsäätimien ilmavirtauksen ja pellin asennon lukeminen ja kummankin ilmamääräsäätimen ohjaaminen tiettyyn suuntaan ilmanlaatulähtetimen arvojen perusteella.

³² Comm1

7.2 Muuttujien määrittely

Kuvasta 28 nähdään, kuinka koodissa olevat eri muuttujat pitää määrittellä koodia kirjoitettaessa. Tulomuuttujia³³ ovat ne muuttujat, jotka käynnistävät jonkin ohjelmassa olevan toiminallisuuden. Lähtömuuttujia³⁴ ovat ne muuttujat, jotka ovat ohjelman tuloksia. Tuloja ja lähtöjä samaan aikaan³⁵ ovat pyyntökäskey, Slave-laitteen vastaus ja kysely-kuuntelu toiminnon käynnistäjä.



```

0001 FUNCTION_BLOCK Comm1
0002 VAR_INPUT
0003   Ilmanlaatulahetin: BOOL := TRUE;
0004
0005   Flakt_Aseno_ja_Virtaus: BOOL;
0006   Flakt_Tarpeenmukainen_Ohjaus: BOOL;
0007
0008   Asento_Lindab: BOOL;
0009   Virtaus_Lindab: BOOL;
0010   Lindab_Tarpeenmukainen_Ohjaus: BOOL;
0011 END_VAR
0012 VAR_OUTPUT
0013   CO2: REAL;
0014   Lampotila: REAL;
0015   Kosteus: REAL;
0016
0017   Flakt_Aseno: REAL;
0018   Flakt_Virtaus: REAL;
0019   Flakt_Auki: BOOL;
0020   Flakt_Kiinni: BOOL;
0021   Flakt_Ei_PakkoOhjausta: BOOL;
0022
0023   Lindab_Aseno: REAL;
0024   Lindab_Auki: BOOL;
0025   Lindab_Kiinni: BOOL;
0026   Lindab_Ei_PakkoOhjausta: BOOL;
0027
0028   Yhteysvirhe: BOOL;
0029 END_VAR
0030 VAR_IN_OUT
0031   ExtQuery: typModbusExtendedQuery;
0032   Response: typModbusResponse;
0033   StartFunction: BOOL;
0034 END_VAR

```

Kuva 28. Comm1:sen muuttujat.

³³ VAR_INPUT

³⁴ VAR_OUTPUT

³⁵ VAR_IN_OUT

7.3 Ilmanlaatulähettimen tietojen luku

Jos kysely-kuuntelu toiminnon käynnistäjän arvo on epätosi, toteutuu ilmanlaatulähettimen ehtolause (kuva 29), koska kysely-kuuntelu toiminnon käynnistäjä on alustettu arvoon tosi. Master lähettää Slavelle seuraavanlaisen kyselyn. Halutaan tietää, paljonko ilmassa on hiilidioksidia, miten lämmin tilassa on ja kuinka kosteaa. Ilmanlaatulähettimen osoitteeksi on asetettu kaksi. Laitteen datalehdestä (kuva 30) nähdään, että halutut tiedot sijaitsevat rekistereissä 30001–30003 eli tulorekistereissä 1–3. Kaikki muut toiminnot, mitä laitteella voidaan tehdä, on esitetty liitteessä 2.

Ilmanlaatulähettimestä halutaan lukea dataa, eli silloin käytetään tulorekisterien lukemiskoodia neljä ja luettavia rekistereitä on kolme. Kyselyn lähtiessä eteenpäin ilmanlaatulähetin lukee halutut rekisterit isännälle, jos tiedonsiirrossa ei tapahdu virhettä ja kysely-kuuntelu toiminnon käynnistäjä saa arvon epätosi. Muussa tapauksessa muuttuja yhteysvirhe saa arvon tosi, ja tiedonsiirrossa on tapahtunut jokin virhe esimerkiksi virheellinen CRC-tarkistussumma. Lämpötilarekisterin sisältämä tieto 0...500 on jaettu 10, että saataisiin haluttu tieto Celsius-asteina.

```

0001 IF NOT StartFunction THEN
0002
0003     IF Ilmanlaatulähetin = TRUE THEN
0004
0005         ExtQuery.SlaveAddress := 2;
0006         ExtQuery.FunctionCode := 4;
0007         ExtQuery.Read_StartAddress := 0;
0008         ExtQuery.Read_Quantity := 3;
0009         ExtQuery.Write_StartAddress := 0;
0010         ExtQuery.Write_Quantity := 0;
0011
0012         IF Response.Error = 0 THEN
0013             IF StartFunction = FALSE THEN
0014
0015                 CO2 := Response.Data[0];
0016                 Lampotila := (Response.Data[1]/10);
0017                 Kosteus := Response.Data[2];
0018
0019             END_IF
0020
0021         ELSE Yhteysvirhe := TRUE;
0022
0023     END_IF

```

Kuva 29. Ilmanlaatulähettimen koodi.

Register	Parameter description	Data type	Values	Range
30001	VOC measurement (CO ₂ equivalent)	Signed 16	0...10000	0...10000 ppm
30002	Temperature measurement	Signed 16	0...500	0.0...50.0 °C
30003	Humidity measurement	Signed 16	0...1000	0.0...100.0 %

Kuva 30. Ilmanlaatulähettimen rekisterit [34, 11].

7.4 Optivent Ultra

7.4.1 Pellin asento ja ilmavirta

Seuraavana toiminnallisuutena aliohjelmassa Fläktin IMS:stä voidaan lukea säätöpellin asento prosentteina ja säätimen läpi menevä ilmavirta litraa sekunnissa (kuva 31). Asento- ja ilmavirtarekisteri löytyvät laitteen datalehden lopusta (kuva 32). Kaikki muut toiminnot, mitä laitteella voidaan tehdä on esitetty liitteessä 3.

Kompaktisäädin tukee vain pitorekisterien lukua ja kirjoittamista yhteen rekisteriin, joiden funktiokoodit ovat esitetty muodossa 03h ja 06h (kuva 33). Numero edessä tarkoittaa funktiokoodia ja h tarkoittaa pitorekisteriä. Lukeminen aloitetaan rekisteristä 4 ja luettavien rekisterien määrä on myös neljä, koska asentotieto sijaitsee rekisterissä 4 ja ilmavirtatieto rekisterissä 7. Lukemisen onnistuessa asentotieto jaetaan 100, että saadaan pellin asento prosentteina.

```

0025  ELSIF Flakt_Asento_ja_Virtaus = TRUE THEN
0026
0027      ExtQuery.SlaveAddress := 4;
0028      ExtQuery.FunctionCode := 3;
0029      ExtQuery.Read_StartAddress := 4;
0030      ExtQuery.Read_Quantity := 4;
0031      ExtQuery.Write_StartAddress := 0;
0032      ExtQuery.Write_Quantity := 0;
0033
0034
0035      IF Response.Error = 0 THEN
0036          IF StartFunction = FALSE THEN
0037
0038              Flakt_Asento := (Response.Data[0]/100);
0039              Flakt_Virtaus := Response.Data[3];
0040
0041              END_IF
0042
0043          ELSE Yhteysvirhe := TRUE;
0044
0045      END_IF

```

Kuva 31. Fläktin pellin asento ja virtauksen koodi.

5	Suhteellinen asento	4	WORD	0..10 000	Luku	Suhteellisen asennon näyttö (%), 0-100,00	-
6	Absoluuttinen asento	5	WORD	0..65535	Luku	Absoluuttisen asennon näyttö [°]	-
7	Suhteellinen ilmavirta	6	WORD	0..10 000	Luku	Ilmavirran näyttö [%]	-
8	Absoluuttinen ilmavirta	7	WORD	0..65535	Luku	Ilmavirran näyttö [l/s, m ³ /h]	-

Kuva 32. Pellin asennon ja ilmavirran rekisterit. [37, 11].

Funktiokoodit	Nimi	Kuvaus
03h	read hold. register	Laiteparametrien / todellisten arvojen luku (kokonais/liuku)
06h	write single register	Laiteparametrien kirjoitus (yksi sana)

Kuva 33. Optivent Ultram tuetut funktiokoodit. [37, 10].

7.4.2 Ilmamääräsäätimen ohjaus

Kolmantena toiminnallisuutena voidaan ohjata Fläktin säätöpeltiä, joko avautumaan, tai menemään kiinni ilmanlaatulähtetimen arvojen mukaan (kuva 34). Ilmanlaatulähtetimen ja IMS:stä aikaisemmin luetut arvot on tallennettu niiden omiin muuttujiin, joista ohjelma hakee arvot tässä vaiheessa.

Pelti aukeaa pakko-ohjaus rekisterin arvolla 1 (kuva 35), jos CO₂-pitoisuus on yli 500 ppm ja kosteus on yli 25 %, tai ilmavirtaus on alle 5 l/s. Pelti menee kiinni rekisterin arvolla 2, jos CO₂-pitoisuus on alle 475 ppm ja kosteus alle 23 %, tai ilmavirtaus yli 10 l/s. Pakko-ohjausta ei tapahdu rekisterin arvolla 0, jos kumpikaan ehto ei täyty.

```

0046 ELSIF Flakt_Tarpeenmukainen_Ohjaus = TRUE THEN
0047
0048     ExtQuery.SlaveAddress := 4;
0049     ExtQuery.FunctionCode := 6;
0050     ExtQuery.Read_StartAddress := 0;
0051     ExtQuery.Read_Quantity := 0;
0052     ExtQuery.Write_StartAddress := 1;
0053     ExtQuery.Write_Quantity := 1;
0054
0055     IF Response.Error = 0 THEN
0056         IF StartFunction = FALSE THEN
0057
0058             IF (CO2 > 500 AND Kosteus > 25) OR Flakt_Virtaus < 5 THEN
0059
0060                 ExtQuery.Write_Data[0] := 1;
0061                 Flakt_Auki := TRUE;
0062
0063             ELSIF (CO2 < 475 AND Kosteus < 23) OR Flakt_Virtaus > 10 THEN
0064
0065                 ExtQuery.Write_Data[0] := 2;
0066                 Flakt_Kiinni := TRUE;
0067
0068             ELSE
0069
0070                 ExtQuery.Write_Data[0] := 0;
0071                 Flakt_Ei_PakkoOhjausta := TRUE;
0072
0073             END_IF
0074         END_IF
0075
0076     ELSE Yhteysvirhe := TRUE;
0077
0078 END_IF

```

Kuva 34. Fläktin ohjauskoodi.

2	Pakko-ohjaus	1	WORD	0..4	Luku/kirj.	Pakko-ohjauksen asetus 0 = ei pakko-ohjausta 1 = auki 2 = kiinni 3 = Vmin 4 = Vmax 5 = Vbtw	0
---	--------------	---	------	------	------------	---	---

Kuva 35. Fläktin pakko-ohjaus rekisteri. [37, 11].

7.5 Ultralink FTCU

7.5.1 Pellin asento ja Ilmavirta

Lindabin säätöpellin asento voitiin myös lukea ohjelmasta, että nähtiin toimiko pellin ohjaus (kuva 36). Laitteen osoitteeksi oli asetettu 3 ja pellin asentorekisteri sijaitsi rekisterissä 251 (kuva 37). Muut toiminnot, joita laitteella voidaan tehdä, on esitetty liitteessä 4.

```

0046 ELSIF Asento_Lindab = TRUE THEN
0047
0048     ExtQuery.SlaveAddress := 3;
0049     ExtQuery.FunctionCode := 4;
0050     ExtQuery.Read_StartAddress := 250;
0051     ExtQuery.Read_Quantity := 1;
0052     ExtQuery.Write_StartAddress := 0;
0053     ExtQuery.Write_Quantity := 0;
0054
0055     IF Response.Error = 0 THEN
0056         IF StartFunction = FALSE THEN
0057
0058             Lindab_Asento := (Response.Data[0]/10);
0059
0060         END_IF

```

Kuva 36. Pellin asennon koodi.

Damper info										
3x251	X		Damper open in %	Damper actual position in percentage open.	16bit	%	10			RO

Kuva 37. Pellin asentorekisteri. [30, 16].

Ohjaus tapahtuu vain CO₂-pitoisuuden ja kosteuden perusteella, koska Lindabin ilmamääräsäätimen virtaustietoja (kuva 38) ei saatu luettua ohjelmasta, koska ilmapirran tilavuustieto on kerrottu 32-bittisenä FLOAT-datatyypinä, joka vastaa samaa asiaa kuin DWORD eli tupla WORD. Modbus tukee vain 16-bittistä tiedonsiirtoa, joten DWORD-datatyypin pitää muuntaa kahdeksi 16-bittiseksi WORD-datatyypiksi. Muuntaminen tapahtuu lukemalla kaksi peräkkäistä rekisteriä 3x154 ja 3x153 (kuva 39). 3 tarkoittaa tulorekisteriä ja x154 rekisteriä 154. Tällöin DWORD saadaan muutettua kahteen peräkkäiseen WORD-rekisteriin (kuva 40). Kahta WORD:ia ei saatu kuitenkaan muunnettua yhdeksi reaalityyppiä.

3x154	X	X	Air flow in l/s	Air flow in l/s	Float	l/s				RO
-------	---	---	-----------------	-----------------	-------	-----	--	--	--	----

Kuva 38. Lindabin ilmapirtarekisteri. [30, 17].

```

0066 ELSIF Virtaus_Lindab = TRUE THEN
0067
0068   ExtQuery.SlaveAddress := 3;
0069   ExtQuery.FunctionCode := 4;
0070   ExtQuery.Read_StartAddress := 153;
0071   ExtQuery.Read_Quantity := 2;
0072   ExtQuery.Write_StartAddress := 0;
0073   ExtQuery.Write_Quantity := 0;
0074
0075   IF Response.Error = 0 THEN
0076     IF StartFunction = FALSE THEN
0077
0078       Response.Data[0];
0079
0080     END_IF
0081
0082   ELSE Yhteysvirhe := TRUE;
0083
0084   END_IF

```

Kuva 39. Lindabin virtauksen koodi.

```

0037   .....Data[0] = 15693
0038   .....Data[1] = 52772

```

Kuva 40. Kaksi ensimmäistä WORD-rekisteriä.

7.5.2 Ilmamääräsäätimen ohjaus

Viimeisenä toiminnallisuutena voidaan ohjata ilmanlaatulähettimen arvojen mukaan Lindabin säätöpeltiä samalla tavalla (kuva 41) kuin Fläktiä. Säätöpelti aukeaa rekisterin 4x151 arvolla 3 (kuva 42), jos CO₂-pitoisuus on yli 500 ppm ja tilan suhteellinen kosteus yli 25 %. 4 tarkoittaa pitorekisteriä ja x151 rekisteriä 151. Pelti ohjautuu kiinni rekisterin arvolla 4, jos CO₂-pitoisuus on alle 475 ppm ja suhteellinen kosteus on alle 23 %. Pakko-ohjausta ei tapahdu kuitenkaan rekisterin arvolla 0, jos kumpikaan ehto ei täyty.

```

0120 ELSIF Lindab_Tarpeenmukainen_Ohjaus = TRUE THEN
0121
0122     ExtQuery.SlaveAddress := 3;
0123     ExtQuery.FunctionCode := 6;
0124     ExtQuery.Read_StartAddress := 0;
0125     ExtQuery.Read_Quantity := 0;
0126     ExtQuery.Write_StartAddress := 150;
0127     ExtQuery.Write_Quantity := 1;
0128
0129     IF Response.Error = 0 THEN
0130         IF StartFunction = FALSE THEN
0131
0132             IF CO2 > 500 AND Kosteus > 25 THEN
0133
0134                 ExtQuery.Write_Data[0] := 3;
0135                 Lindab_Auki := TRUE;
0136
0137             ELSIF CO2 < 475 AND Kosteus < 23 THEN
0138
0139                 ExtQuery.Write_Data[0] := 4;
0140                 Lindab_Kiinni := TRUE;
0141
0142             ELSE
0143
0144                 ExtQuery.Write_Data[0] := 0;
0145                 Lindab_Ei_PakkoOhjausta := TRUE;
0146
0147             END_IF
0148         END_IF
0149     ELSE Yhteysvirhe := TRUE;
0150 END_IF
0151

```

Kuva 41. Lindabin ohjaukoodi.

4x151	X	Damper Override Conf.	0 = Normal mode; 1 = Override control - Max open; 2 = Override control - Min open; 3 = Override control - 100% open; 4 = Override control - 100% closed	16bit			0	0	4	RW
-------	---	-----------------------	---	-------	--	--	---	---	---	----

Kuva 42. Lindabin pakko-ohjaus rekisteri. [30, 17].

7.6 Aliohjelman lopetus

Aliohjelma päättyy kuvan 43 tavalla. Kun kysely-kuuntelu toiminnon käynnistäjän arvo muutetaan todeksi aliohjelman lopussa, käy aliohjelma tehtyä koodia jatkuvasti läpi. Tällä tavalla mahdollistetaan esimerkiksi ilmanlaatulähettimen tietojen lukeminen jatkuvana, eikä pelkästään juuri sen hetkinen tilanne. Tämä edellyttää kuitenkin sen, että esimerkiksi ilmanlaatulähetintä luettaessa tulomuuttujan Ilmanlaatulähettimen arvon pitää olla tosi. Ilmanlaatulähettimen arvojen lukeminen onnistuu siis jatkuvana, mutta rekisteriin kirjoittaminen, eli ilmamääräsäätimien ohjaus täytyi tehdä aina manuaalisesti.

```

0113     END_IF
0114     END_IF
0115 StartFunction := TRUE;
0116 END_IF
0117
0118

```

Kuva 43. Toiminallisuuden lopetus.

8 Pohdinta ja yhteenveto

Toimeksiantajana oli Karelia-Ammattikorkeakoulu ja se tehtiin oppilaitoksen talotekniikan laboratorioon. Työn tarkoitus oli tutkia ultraääni ilmamääränsäätimiä, että ne voitaisiin ottaa opetuskäyttöön tulevaisuudessa. Työ onkin hyvä apuväline opiskelijoille heidän opiskellessaan kuinka kahden eri valmistajan ultraääni ilmamääränsäätimet pitää kytkeä ja ohjelmoida, että ne toimivat oikein. Opinnäytetyötä tehdessä oppi hakemaan tietoa monesta eri paikasta, lukemaan laitteiden datalehtiä, kärsivällisyyttä ja periksiantamattomuutta. Lindabin laitteesta ei löytynyt paljoa tietoa, minkä takia sitä käsittelevä osuus on melko suppea.

Opinnäytetyön ei ajateltu olevan näin haastava, kuin mitä se lopulta oli. Ensin haasteita tuotti kytkentöjen opettelu, koska laboratoriotyöt tehtiin kursseilla aina ryhmässä. Ryhmässä oli aina joku, joka teki kytkennät nopeasti ajan säästämiseksi, joten niitä ei nähtävästi oppinut vain katsomalla vierestä. Seuraava haaste oli toimivan ohjelman tekeminen. Kävi ilmi, että ohjelman täydellinen onnistuminen olisi vaatinut parempaa ohjelmointiosaamista, kuin mitä oli. Ohjelmointiin kului eniten aikaa koko työssä, koska haluttiin saada aikaan täysin automaattinen järjestelmä. Täysin automaattisesti toimiva järjestelmä haluttiin saada aikaan sen takia, koska sillä tavalla tarpeenmukaisen ilmanvaihdon järjestelmä toimii rakennuksissa.

Opinnäytetyössä onnistuttiin alkuperäisten tavoitteiden mukaisesti lukemaan dataa ilmanlaatulähettimeen ja molempien ultraääni-ilmamääränsäätimien

rekistereistä. Onnistuttiin myös ohjaamaan molempia säätimiä ilmanlaatulähettimeen tilatietojen mukaan. Fläktin säädintä onnistuttiin ohjaamaan myös sen läpi menevän ilmavirran tilavuuden mukaan. Itse ohjelmointi onnistui myös ja saatiin tehtyä Modbus-toiminnallisuuteen toimivat IF-ehtolauseet, joiden sisältämät vaihtoehdot saatiin kaikki toimimaan puhaltamalla ilmanlaatulähettimeen tai Fläktin IMS:sin ollessa kyseessä myös puhaltimen nopeutta vaihtamalla. Opinnäytetyössä ei onnistuttu saada Lindabin ilmamääräsäätimen virtaustietoa halutussa datatyypissä, että myös Lindabin ilmamääräsäädintä olisi voitu ohjata ilmamäärän tilavuuden mukaan Fläktin ilmamääräsäätimen tavoin.

Työtä on mahdollista jatkokehittää paremmilla ohjelmointitaidoilla täysin itsenäiseksi toimivaksi järjestelmäksi, joka säätää ilmamääräsäätimien säätöpeltejä täysin automaattisesti asetettujen parametrien mukaan. Työhön voidaan myös tehdä jatkokehittelynä visualisointi, jota ei tässä työssä tehty. Myös molempien ilmamääräsäätimien lähettämää ilmavirran tilavuuden dataa on mahdollista ohjelmassa verrata keskenään, jos osataan konvertoida Lindabin kahdessa peräkkäisessä WORD-rekisterissä oleva virtaustieto yhden rekisterin reaaliin muotoon.

Lähteet

- 1 Lindab. 2020. Lindab Ultralink® Huoltovapaa ilmavirran mittausta ja säätö. <http://www.lindab.com/fi/pro/pages/ultralink.aspx> 30.1.2020.
- 2 Vallox. Ilmastointi vai ilmanvaihto? https://www.vallox.com/tietoa_ilmanvaihdosta/ilmastointi_vai_ilmanvaihto.html 6.3.2020.
- 3 Hengitysliitto. Ilmanvaihtojärjestelmät <https://www.hengitysliitto.fi/fi/sisailma/ilmanvaihto/ilmanvaihtojarjestelmat> 6.3.2020.
- 4 Talotekniikkainfo. 2020. Sisäilmasto ja ilmanvaihto-opas. <https://www.talotekniikkainfo.fi/sisailmasto-ja-ilmanvaihto-opas/8-ss-ilmanvaihto> 03.02.2020.
- 5 Savitie, I. 2017. Ilmanvaihto on miljoonien eurojen kakku. Talotekniikka-lehti. <https://talotekniikka-lehti.fi/ilmanvaihto-miljoonien-eurojen-kakku/> 30.1.2020.
- 6 Airaksinen H. 2019. Energiankulutuksesta ja ilmanvaihdosta. Äyräväinen. <https://www.ayravainen.fi/2019/06/energiankulutuksesta-ja-ilmanvaihdosta/> 30.1.2020.
- 7 Hengitysliitto. Hiilidioksidi. <https://www.hengitysliitto.fi/fi/sisailma/sisailma-asiat-sisailmaongelmat/kaasumaiset-epapuhautet/hiilidioksidi> 30.1.2020.
- 8 Hengitysliitto. Sisäilman epäpuhtaudet. <https://www.hengitysliitto.fi/fi/sisailma/sisailma-asiat-sisailmaongelmat/kaasumaiset-epapuhautet> 30.1.2020.
- 9 Piikkilä, V. Sahlstén, T. 2017. ST-käsikirja 21 Kiinteistöjen tiedonsiirtoväylät. Sähköinfo Severi. <http://severi.sahkoinfo.fi/item/231> 30.1.2020.
- 10 Wago. 2020. Enocean. <https://www.wago.com/fi/enoccean> 30.1.2020.
- 11 Realpars. 2020. What is fieldbus? <https://realpars.com/fieldbus/> 30.1.2020.
- 12 Sparkfun. Introduction to RS485. <https://learn.sparkfun.com/tutorials/ast-can485-hookup-guide/introduction-to-rs485> 19.3.2020.
- 13 Frenzel, L. 2013. What's the difference between RS-232 and RS-485 serial interfaces? Electronic Design. <https://www.electronicdesign.com/technologies/communications/article/21800966/whats-the-difference-between-the-rs232-and-rs485-serial-interfaces> 30.1.2020.

- 14 Elprocus. 2020. Understanding a Programming Logic Controller (PLC). <https://www.elprocus.com/understanding-a-programming-logic-controller/> 30.1.2020.
- 15 Nelson, J. 2019. Programmable Logic Controllers (PLCs) vs Hard Wiring. Panelshop.com. <https://info.panelshop.com/blog/programmable-logic-controllers-plcs-vs.-hard-wiring> 25.2.2020.
- 16 Wago. 2020. 750-881. <https://www.wago.com/fi/logiikka/ohjain-ether-net/p/750-881> 25.2.2020.
- 17 Codesys. 2020. CODESYS - THE COMPREHENSIVE SOFTWARE SUITE FOR AUTOMATION TECHNOLOGY. <https://www.codesys.com/the-system.html> 30.1.2020.
- 18 Simply Modbus. 2019. Frequently Asked Questions. <https://www.simplymodbus.ca/FAQ.htm> 30.1.2020.
- 19 Control Solutions Minnesota. 2019. Modbus 101 - Introduction to Modbus. https://www.csimn.com/CSI_pages/Modbus101.html 30.1.2020.
- 20 Modbus.org. 2002. MODBUS over Serial Line V1.0. http://www.modbus.org/docs/Modbus_over_serial_line_V1.pdf 30.1.2020.
- 21 Picotech. Modbus® serial protocol decoding. <https://www.picotech.com/library/oscilloscopes/modbus-serial-protocol-decoding> 30.1.2020
- 22 Immonen, S. Mutanen, S. Kananen, A. Berg, T. 2017. Käyttöliittymät ja tiedonsiirto kurssin julkaisematon opintojaksoraportti. 19.3.2020.
- 23 Rutherford, K. 2018. Demystifying Modbus Function Codes. Software Toolbox. <https://blog.softwaretoolbox.com/opc-modbus-function-codes> 30.1.2020.
- 24 M-System. 2020. Modbus Protocol Reference Guide. <https://www.m-system.co.jp/mssenglish/service/emmodbus.pdf> 30.1.2020.
- 25 Schneider Electric. 2018. What do the most common Modbus Exception Error Code's mean? <https://www.se.com/id/en/faqs/FA180028/> 30.1.2020.
- 26 Simply Modbus. 2019. Expection Responses. <http://www.simplymodbus.ca/exceptions.htm> 30.1.2020.

- 27 Oulun Ultra. 2020. Mitä ultraääni on? <https://oulunultra.fi/ultraaani/> 30.1.2020.
- 28 Halton. 2017. PRA- Ilmavirtasäädin. https://www.halton.com/fi_FI/halton/products/-/product/PRA 30.1.2020.
- 29 FläktGroup. OPTIVENT® ULTRA - ULSA. <https://www.flaktgroup.com/fi/products/air-management--atds/air-volume-dampers/variable-air-volume-dampers/optivent-ultra---ulsa/> 24.2.2020
- 30 Lindab. Lindab Ulltralink®-ilmavirtasäädin FTCU tekniset tiedot. https://itsolution.lindab.com/LindabWebProductsDoc/PDF/Documentation/ADS/FIN/Technical/UltraLink-Technical-FTCU_FI.pdf 24.2.2020.
- 31 Expressi. 2017. Innovaatio mullistaa ilmanvaihdon: terveellinen sisäilma ekologisesti ultraäänien avulla. <https://www.epressi.com/tiedotteet/rakentaminen/innovaatio-mullistaa-ilmanvaihdon-terveellinen-sisailma-ekologisesti-ultraaanen-avulla.html> 30.1.2020.
- 32 FläktGroup. OPTIVENT® ULTRA KÄYTTÖOPAS. <http://resources.flaktwoods.com/Perfion/File.aspx?id=590128e7-45dd-4641-9961-8f6b9467f953> 24.2.2020.
- 33 Ventilatory.net. 2016. TD 160/100 Ecowatt. <https://en.ventilatory.net/td-ecowatt-160-100.html> 21.3.2020.
- 34 Pro dual. Ilmanlaatulähetin ILH-M-RH-N. https://www.produal.com/fi/shop/web_concentration_transmitters/sku-1135623 30.1.2020.
- 35 Wago. Modb_I05.lib. https://www.engineering-today.ru/assets/modb_I05-manual.pdf 06.03.2020.
- 36 Pro dual. ML-SER. https://www.produal.com/fi/shop/web_commissioning_tools/sku-1139010 30.1.2020.
- 37 FläktGroup. Modbus-kompaktisäädin 227VMZ-MB. <https://www.flaktgroup.com/api/v1/Documents/c3240cf7-bbc5-4697-ac9c-91e3ce344e4f> 06.03.2020.

Modbus Extended Master PRG
MODBUS_EXTENDED_MASTER

WAGO-I/O-PRO 32 library elements		
Category:	Communication	
Name:	MODBUS_EXTENDED_MASTER	
Type:	Function	Function block X Program
Library name:	Modb_i05.lib	
Used libraries:	Serial_Interface_01.lib SerComm.lib Mod_Com.lib	
Applicable to:	Controller 750-8xx (except 750-812/814/815/816)	
Inputparameter:	Datatype:	Description:
ENABLE	BOOL	Default = TRUE opens the serial port
ASCII_Mode	BOOL	Default = FALSE -> Modbus RTU FALSE -> Modbus RTU TRUE -> Modbus ASCII
bCOM_PORT	BYTE	Port number of serial interface module 16#01 => Service Port 16#02 => 1. Module 750-65X 16#03 => 2. Module 750-65X etc. (up to 64 750-65X modules)
cbCOM_BAUDRATE	COM_BAUDRATE	Attention ! Not all baudrates are supported by all modules. Please check the manual of the used module for your wanted baudrate. BAUD_110 := 11, BAUD_300 := 30, BAUD_600 := 60, BAUD_1200 := 120, BAUD_2400 := 240, BAUD_4800 := 480, BAUD_9600 := 960, BAUD_12000 := 1200, BAUD_14400 := 1440, BAUD_19200 := 1920, BAUD_28800 := 2880, BAUD_38400 := 3840, BAUD_57600 := 5760, BAUD_62500 := 6250, BAUD_115200 := 11520, BAUDRATE_TERMINAL_DEFAULT := 16#7FFF
cbsCOM_BYTESIZE	COM_BYTESIZE	BS_7 :=7, BS_8 :=8, BYTESIZE_TERMINAL_DEFAULT := 16#7FFF

Modbus Extended Master PRG

WAGO-I/O-PRO 32 library elements		
cpCOM_PARITY	COM_PARITY	PARITY_NO:=0, PARITY_ODD:=1, PARITY_EVEN:=2, PARITY_TERMINAL_DEFAULT := 16#7FFF
csCOM_STOPBITS	COM_STOPBITS	STOPBITS_1 := 1, STOPBITS_2 := 2, STOPBITS_TERMINAL_DEFAULT := 16#7FFF
cfCOM_FLOW_CONTROL	COM_FLOW_CONTROL	NO_FLOW_CONTROL :=0, XON_XOFF :=1, RTS_CTS:=2, FULLDUPLEX:=3, HALFDUPLEX:=4, FLOW_CONTROL_TERMINAL_DEFAULT := 16#7FFF
TimeOut	TIME	Maximum delay time
In-/Outparam.:	Datatype:	Description:
StartFunction	BOOL	Set TRUE to start execution of modbus service. The variable will be reset by the funtion block after execution.
ExtQuery	typModbusExtendedQuery	TYPE typModbusExtendedQuery : STRUCT SlaveAddress : BYTE; FunctionCode : BYTE; Read_StartAddress : UINT; Read_Quantity : UINT; Write_StartAddress : UINT; Write_Quantity : UINT; Write_Data : ARRAY[0..124] OF WORD; END_STRUCT END_TYPE
Response	typModbusResponse	TYPE typModbusResponse : STRUCT Error : WORD; SlaveAddress : BYTE; FunctionCode : BYTE; StartAddress : UINT; Quantity : UINT; Data : ARRAY[0..124] OF WORD; END_STRUCT END_TYPE
Ausgangsparameter:	Datentyp:	Kommentar:

Modbus Extended Master PRG

WAGO-I/O-PRO 32 library elements		
MB_Error	enumMB_ERROR	Errorcode TYPE enumMB_ERROR :(MB_NO_ERROR := 16#00, MB_NOT_SUPPORTED_FUNCTION := 16#01, MB_ILLEGAL_DATA := 16#03, MB_EXTENDED_SLAVE_ERROR := 16#90, MB_CRC_ERROR := 16#96, MB_ILLEGAL_NUMBER_OF_POINTS:=16#97, MB_OVERRUN := 16#98, MB_TIME_OUT := 16#99); END_TYPE
<p>Graphic display:</p> <div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">MODBUS_EXTENDED_MASTER</p> <pre> --ENABLE : BOOL --ASCII_Mode : BOOL --bCOM_PORT : BYTE --cbCOM_BAUDRATE : COM_BAUDRATE --cpCOM_PARITY : COM_PARITY --csCOM_STOPBITS : COM_STOPBITS --cbsCOM_BYTE_SIZE : COM_BYTE_SIZE --cfCOM_FLOW_CONTROL : COM_FLOW_CONTROL --TIME_OUT : TIME --StartFunction : BOOL (VAR_IN_OUT) --ExtQuery : typModbusExtendedQuery (VAR_IN_OUT) --Response : typModbusResponse (VAR_IN_OUT) </pre> </div>		
<p>Function description: The function block "MODBUS_MASTER", expands the functionality of a programmable controller as a modbus master (RTU / ASCII).</p> <p>Supported protocols: Modbus RTU Modbus ASCII</p> <p>Supported MODBUS-Services: 0x01 Read Coil Status 0x02 Read Input Status 0x03 Read Holding Register 0x04 Read Input Registers 0x05 Force Single Coil 0x06 Write Single Registers 0x0B Get Comm Event Ctr 0x0F Force Multiple Coils 0x10 Write Multiple Registers 0x17 Read / Write Multiple Registers</p>		
<p>Execute Modbusfunction The user have to place all for the request of the service needed informations at the items of the structure ExtQuery. After this he starts the execution with a rising edge at parameter "StartFunction". The connected variable at this input will be reset by the function block after execution.</p> <p>After execution the user can find the answer inside response structur.</p>		

Modbus Extended Master PRG

Appendix A – Additional Libraries



typModbusExtendedQuery (Datentyp)

WAGO-I/O-PRO 32 library elements		
Category:		
Name:	typModbusExtendedQuery	
Type:	Datatype X	Enumeration
Library name:	Modb_I05.lib	
Structure:		
Element:	Datatype:	Description:
SlaveAddress	BYTE	Address of the requested slave
FunctionCode	BYTE	Modbus-FunctionCode
Read_StartAddress	UINT	target memory address for requested data
Read_Quantity	UINT	quantity of requested data
Write_StartAddress	UINT	target memory address for data to write
Write_Quantity	UINT	quantity of writedata
Write_Data	ARRAY [0..124] OF WORD	data (for write services only)
Header:		
<pre> TYPE typModbusExtendedQuery : STRUCT SlaveAddress : BYTE; (* address of the requested slave *) FunctionCode : BYTE; Read_StartAddress : UINT; Read_Quantity : UINT; Write_StartAddress : UINT; Write_Quantity : UINT; Write_Data : ARRAY[0..124] OF WORD; (* Modbus specifies max. 125 words in one telegram *) END_STRUCT END_TYPE </pre>		

Modbus Extended Master PRG



Appendix A – Additional Libraries

typModbusResponse (Datentyp)

WAGO-I/O-PRO 32 library elements		
Category:		
Name:	typModbusResponse	
Type:	Datatype X	Enumeration
Library name:	Modb_I05.lib	
Structure:		
Element:	Datatype:	Description:
Error	WORD	Error code generated by the slave
SlaveAddress	BYTE	Address of requested slave
FunctionCode	BYTE	Modbus function code
StartAddress	UINT	Target memory address of data
Quantity	UINT	Quantity of data
Data	ARRAY [0..512] OF BYTE	data (for read services only)
Deklaration:		
<pre> TYPE typModbusResponse: STRUCT Error : WORD; SlaveAddress : BYTE; FunctionCode : BYTE; StartAddress : UINT; Quantity : UINT; Data : ARRAY[0..124] OF WORD; (* Modbus specifies max. 125 words in one telegram *) END_STRUCT END_TYPE </pre>		

Produal ILH-M-RH Modbus-rekisterit



KÄYTTÖOHJE
ILH
9 (10)

Modbus-rekisterit

HUOM: Jos yrität kirjoittaa rekisterin alueen ulkopuolella olevan arvon, arvoksi kirjoitetaan lähin hyväksyttävä arvo. Jos esimerkiksi kirjoitat rekisterin 40011 arvoksi 270, rekisterin arvoksi kirjoitetaan 260.

Coils

Register	Parameter description	Data type	Values	Range	Default
1	Y1 output override activation	Bit	0 - 1	0: Off, 1: On	0
2	Y2 output override activation	Bit	0 - 1	0: Off, 1: On	0
3	Y3 output override activation	Bit	0 - 1	0: Off, 1: On	0
4	Y4 output override activation	Bit	0 - 1	0: Off, 1: On	0
5	Relay override activation	Bit	0 - 1	0: Off, 1: On	0
6	Relay override	Bit	0 - 1	0: Off, 1: On	0

Discrete inputs

Register	Parameter description	Data type	Values	Range
10001	Relay status	Bit	0 - 1	0: Off, 1: On

Input registers

Register	Parameter description	Data type	Values	Range
30001	VOC measurement (CO ₂ equivalent)	Signed 16	0...10000	0...10000 ppm
30002	Temperature measurement	Signed 16	0...500	0.0...50.0 °C
30003	Humidity measurement	Signed 16	0...1000	0.0...100.0 %
30004	Y1 output voltage	Unsigned 16	0...1000	0.00...10.00 V
30005	Y2 output voltage	Unsigned 16	0...1000	0.00...10.00 V
30006	Y3 output voltage	Unsigned 16	0...1000	0.00...10.00 V
30007	Y4 output voltage	Unsigned 16	0...1000	0.00...10.00 V
30008	Active potentiometer value	Unsigned 16	0...1000	0.0...100.0 %

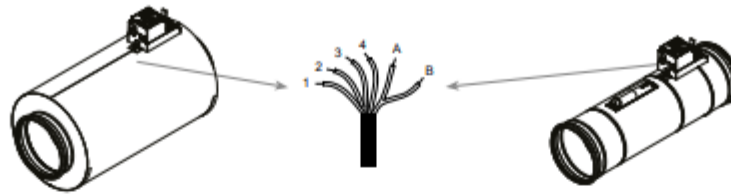
Holding registers

Register	Parameter description	Data type	Values	Range	Default
40001	Y1 output override	Signed 16	0...1000	0...10.00 V	0
40002	Y2 output override	Signed 16	0...1000	0...10.00 V	0
40003	Y3 output override	Signed 16	0...1000	0...10.00 V	0
40004	Y4 output override	Signed 16	0...1000	0...10.00 V	0
40005	VOC measurement tuning (offset)	Signed 16	-200...200	-200...200 ppm	0
40006	Temperature measurement tuning (offset)	Signed 16	-30...30	-3.0...3.0 °C	0
40007	Humidity measurement tuning (offset)	Signed 16	-5...5	-5...5 %	0
40008	Control method	Signed 16	0 - 1	0: P 1: PI	1
40009	Controller output	Signed 16	0 - 1 - 2 - 3 - 4	0: not in use 1: VOC 2: temperature 3: humidity 4: maximum selection	1
40010	Set point, VOC	Signed 16	450...2000	450...2000 ppm	700
40011	Set point, temperature	Signed 16	180...260	18.0...26.0 °C	210
40012	Set point, humidity	Signed 16	0...100	0...100 %	50
40013	Proportional band, VOC	Signed 16	100...2000	100...2000 ppm	500

Optivent Ultra kytkentäohjeet ja Modbus-rekisterit

RAKENNE JA TOIMINTA

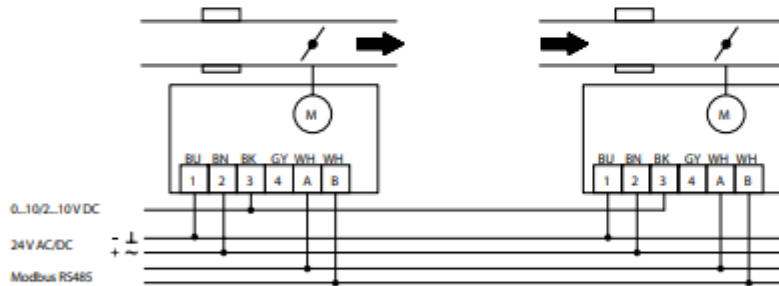
KYTKENTÄ



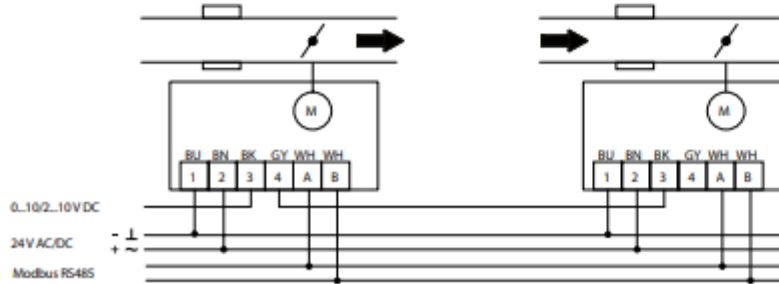
1	2	3	4	A	B
24 VAC - 24 VDC -	24 VAC - 24 VDC +	0..10 V 2..10 V	0..10 V 2..10 V	Modbus RS485	
Käyttöjännite		Ohjaussignaali	Takaisinkyntäsignaali		
sininen (BU)	ruskea (BN)	musta (BK)	harmaa (GY)	valkoinen (WH)	

Mahdolliset kytkentäratkaisut:

Rinnakkaisohjaus



Master-slave (samat koot)



V_{min} = 0
V_{max} = V_{nom}

Optivent Ultra kytkentäohjeet ja Modbus-rekisterit

MODBUS-REKISTERIT

OPERAATIOREKISTERIT

Numero	Nimi	Osoite	Datatyyppi	Alue	Luku/kirj.	Kuvaus	Oletus
1	Asetusarvo	0	WORD	0..10 000	Luku/kirj.	Asetusarvon näyttö/asetus (%) 0-100,00 %, vastaa arvoja Vmin ja Vmax 0 % = Vmin 100 % = Vmax Tämä rekisteri on vain luku -muotoinen, jos rekisteri 122 = 0 tai 3	-
2	Pakko-ohjaus	1	WORD	0..4	Luku/kirj.	Pakko-ohjauksen asetus 0 = ei pakko-ohjausta 1 = auki 2 = kiinni 3 = Vmin 4 = Vmax 5 = vbtw	0
4	Laitetunnus	3	WORD	0..7	Luku	Laitetunnuksen näyttö 0 = - - - 1 = vakiotilailaite 2 = IMS 3 = palopelti 4 = GUAC VAV 5 = GUAC CM 6 = GT 7 = PICB	2
5	Suhteellinen asento	4	WORD	0..10 000	Luku	Suhteellisen asennon näyttö (%), 0-100,00	-
6	Absoluuttinen asento	5	WORD	0..65535	Luku	Absoluuttisen asennon näyttö [°]	-
7	Suhteellinen ilmavirta	6	WORD	0..10 000	Luku	Ilmavirran näyttö [%]	-
8	Absoluuttinen ilmavirta	7	WORD	0..65535	Luku	Ilmavirran näyttö [l/s, m ³ /h]	-
9	Anturin arvo	8	WORD	0..10 000	Luku	Tulosignaalin arvon näyttö [mV] 0 ...10000	-
11	Analoginen lähtö	10	WORD	0..10 000	Luku/kirj.	Analogisen lähdön asetus (mV), 0-10 000 Ks. rekisteri 122	-
13	Lämpötila	12	Etumerkillinen WORD	-5000 ...+30000	Luku	[°C] -50,00 ... +300,00	

Optivent Ultra kytkentäohjeet ja Modbus-rekisterit

HUOLTOREKISTERIT

Näitä arvoja käytetään vain virtaus säätimen käyttöönoton ja tilan valvonnan yhteydessä. Älä kirjoita jatkuvasti arvoja huoltorekistereihin.

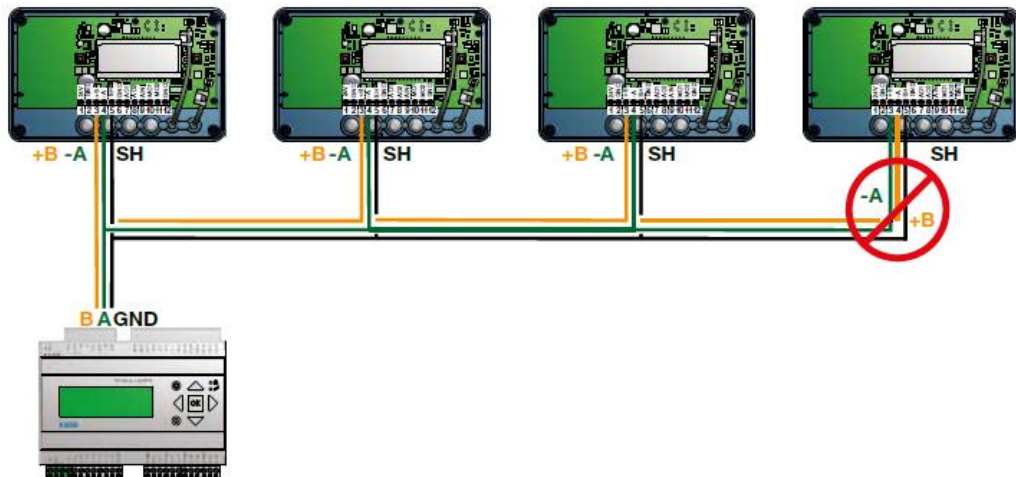
Numero	Nimi	Osoite	Datatyyppi	Alue	Luku/kirj.	Kuvaus	Oletus		
104	Ohjelmistoversio	103	WORD	1..85535	Luku	Ohjelmistoversion näyttö	-		
105	Huoltotiedot	104	WORD	1..85535	Luku	Toimilaitteen tilan näyttö (ks. tilarekisteritaulukko)	-		
106	Suht. min.-arvo	105	WORD	0..10 000	Luku/kirj.	Vmin-arvon asetus [%] 0.. 100.00%	0		
107	Suht. maks.-arvo	106	WORD	0..10 000	Luku/kirj.	Vmax-arvon asetus [%] 0.. 100.00%	10 000		
109	Väylävirheasento	108	WORD		Luku/kirj.	Toimilaitteen asento, kun väylällä ei ole tapahtunut tiedonsiirtoa 120 sekunttiin 0 = väylää ei tarkkailta 1 = kiinni, kun aikakatkaistu tulee 2 = auki, kun aikakatkaistu tulee	0		
119	Suht. väliarvo	118	WORD	0..10 000	Luku/kirj.	Vbtw-arvon asetus [%] 0.. 100.00%	5 000		
120	Absol. väliarvo	119	WORD	0..10 000	Luku/kirj.	Vbtw-arvon asetus [l/s, m ³ /h]	Koosta riippuva		
121	Absol. min.-arvo	120	WORD	0..10 000	Luku/kirj.	Minimiarvon asetus [l/s, m ³ /h]	0		
122	Absol. maks.-arvo	121	WORD	0..10 000	Luku/kirj.	Maksimiarvon asetus [l/s, m ³ /h]	Koosta riippuva		
123	Asetusarvosignaalin ja analogisen lähdön käyttötapa	122	WORD	0..3	Luku/kirj.	Referenssisignaalin ja analogisen lähdön käytettävän asetus	0		
						Rekisterin arvo		Asetusarvo	Lähtösignaalin U arvo
						0		Asetusarvo analogiselta tuloilta Y	Tod. ilmavirta 0 (2)..10V
						1		Asetusarvon kirjoitus rekisterin 0	Tod. ilmavirta 0(2)..10V
						2		Asetusarvon kirjoitus rekisterin 0	Rekisterin 10 kirjoitettu arvo
3	Asetusarvo analogiselta tuloilta Y	Rekisterin 10 kirjoitettu arvo							
131	Modbus-osoitte	130	WORD	1-247	Luku/kirj.	Laitteen osoitteen asetus	1		
132	Anturin tila	131	WORD	0..85535	Luku	Anturin tilan näyttö (ks. anturitilarekisteritaulukko)	-		
133	Asennusasetus	132	WORD	50-150	Luku/kirj.	Asennusasetuksen valinta	100		
202	Ilmavirran yksikkö	201	WORD	0.1	Luku/kirj.	0 = l/s, 1 = m ³ /h	0		
542	Automaattitestausta	541	WORD	0.1	Luku/kirj.	0 = automaattitestausta päällä, 1 = automaattitestausta pois päältä	0		
569	Modbus-yhteysoasetukset	568	BYTE	0..8	Luku/kirj.	Modbus-yhteysoasetusten valinta (ks. Modbus-asetusten taulukko) Huom.: tallennettavat arvot alkavat nollassa.	4		
770	Käyttötapa	769	WORD	0..85535	Luku	Pakko-ohjaustilan näyttö (ks. käyttötaparekisteritaulukko)	-		

Ultralink FTCU kytkentäohjeet ja Modbus-rekisterit

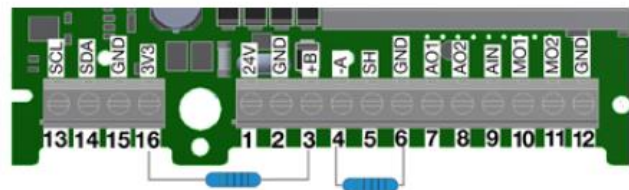
Digitaalinen väyläohjaus

Suosittellemme käyttämään RS485 kierrettyä häiriösuojattua parikaapella. Samaa kaapelia ei saa käyttää virransyöttöön ellei kaapelia ole valmistettu siihen tarkoitukseen.

Liitä Modbus RTU:n A näyttölaitteen -A -liittimeen ja B +B-liittimeen. Useampaa ilmamääräsäädintä liitettäessä väylään on tärkeää kytkeä kaapelit oikein, sillä ristilin kytkeminen katkaisee Modbus-väylän toiminnan. Häiriösuoja RS485-kaapelissa tulee kytkeä muuntajan maallittimestä katkeamatta saman muuntajan takana olevien ilmamääräsäätimien SH-liittimien välille. Jos käytetään useampaa muuntajaa, häiriösuoja tulee katkaista kullakin muuntajalla, jolloin ilmamääräsäätimien SH:t ovat liitetyt ainoastaan oman muuntajansa maahan.



Väylä tulee olla varustettu -A- ja +B-lähtöjen biasoinilla. Jos Modbus master ei ole varustettu biasoinilla, voidaan yksittäinen väylässä oleva ilmamääräsäädin varustaa sopivilla vastuksilla. Käytä 500 - 1000 Ω vastuksia ja liitä yksi vastus -A- ja GND-liittimien välille sekä yksi vastus +B- ja 3V3-liittimien välille. On myös suositeltavaa liittää yksi 120 Ω päätevastus viimeisen ilmamääräsäätimen -A- ja +B-liittimien välille estämään signaaliheijastumat.



Ultralink FTCU kytkentäohjeet ja Modbus-rekisterit

Appendix A – Modbus register

Address :	Modbus register address (3x indicates Input & 4x indicates Holding)
UltraLink® :	Type of UltraLink® where the register is available (Indicated by "x")
Name:	Name of register
Description:	Short description of register.
Data type:	Data type for register (16bit contained in one register, 32bit and float in two consecutive registers).
Unit:	Unit for register value (if any).
Div:	Scale factor for stored value (divide register value with "div" to get correct value).
Default:	Default setting.
Min:	Minimum value allowed for the register.
Max:	Maximum value allowed for the register.
Access:	RO for read only (Input registers) and RW for read and write (Holding registers).

Address	UltraLink®		Name	Description	Datatype	Unit	Div	Default	Min	Max	Access
	Controller	Monitor									
INPUT REGISTERS											
3x008	X	X	Product Nominal Size	Nominal diameter of duct	16bit	mm					RO
3x013	X	X	Unit Status	Current unit status: 0 = Normal mode; 1 = Locating flow; 2 = Override control; 3 = Error; 4 = Control loop regulating; 5 = Angle sensor calibrating	16bit						RO
Flow Info											
3x150	X	X	Velocity in m/s	Velocity in m/s	Float	m/s					RO
3x152	X	X	Air flow in m³/h	Air flow in m³/h	Float	m³/h					RO
3x154	X	X	Air flow in l/s	Air flow in l/s	Float	l/s					RO
Temperature Info											
3x200	X	X	Current temperature in °C	Temperature in degree celsius.	16bit	°C	10				RO
Damper Info											
3x251	X		Damper open in %	Damper actual position in percentage open.	16bit	%	10				RO
3x252	X		Damper motor action	Damper motor action: 0 = Motor stopped. 1 = Motor opening damper 2 = Motor closing damper	16bit						RO
Alarms											
3x400	X	X	Alarm Register 1	Alarms 1-32 - bitwise: 1 = Motor not working. 2 = Angle sensor not working correctly. 3 = Flow setpoint not reached. 4 = Flow measure problems. 5 = Damper is regulating. 6 = Not used. 7 - 31 = Reserved for future use. 32 = Factory data is corrupted.	32bit						RO
Other											
3x500	X	X	Signal amplification	Current signal amplification	16bit			0	3	20	RO

Ultralink FTCU kytkentäohjeet ja Modbus-rekisterit

Address	UltraLink®		Name	Description	Data type	Unit	Div	Default	Min	Max	Access
	Controller	Monitor									
HOLDING REGISTERS											
Communication settings											
4x001	X	X	Communication Id	Modbus address	16bit				1	239	RW
4x002	X	X	RS485 Baud Rate Conf.	Baudrate: 0 – 9600 1 – 19200 2 – 38400 3 – 76800	16bit			1	0	3	RW
4x003	X	X	RS485 Parity Conf.	Parity: 0 – Odd; 1 – Even; 2 – None	16bit			0	0	2	RW
4x004	X	X	RS485 Stop Bit Conf.	Number of stopbits: 1 or 2.	16bit			1	1	2	RW
4x005	X	X	RS485 Protocol Conf.	Protocol: 0 – Modbus; 1 – Not used; 2 – Pascal;	16bit			0	0	2	RW
4x006	X	X	Bluetooth Password	Password which must be provided to pair Bluetooth devices. This password can always be changed from wired connection. From wireless it can only be changed when connection is established using current password.	16bit			1111	0000	9999	RW
4x007	X	X	Bluetooth Enable	Enable Bluetooth Communication 0 – Bluetooth turned off; 1 – Bluetooth turned on;	16bit			1	0	2	RW
4x008	X	X	PLA	ID used for Pascal	16bit				1	239	RW
4x009	X	X	ELA	ID used for Pascal	16bit				1	239	RW
4x010	X	X	Bluetooth TX Power Level	Configure TX Power Level dBm. Accepted values: -40, -20, -16, -12, -8, -4, 0, 2, 3, 4, 5, 6, 7, 8, 9	16bit			0	-40	9	RW
System configuration											
4x070	X		Damper Regulation Conf.	Specifies how damper is regulated: 0 – Regulator turned off 1 – Regulate damper angle 2 – Regulate flow	16bit			2	0	2	RW
4x071	X		Damper Input Conf.	Specifies input to control damper: 0 – Modbus or Pascal 1 – Analog Input	16bit			1	0	1	RW
4x082	X	X	Execute Factory Reset	Factory reset of all parameters. Unit will restart 0 – Do nothing; 1 – Factory Reset	16bit			0	0	1	RW
4x083	X	X	Execute Reboot	Reboot the unit 0 – Do nothing; 1 – Reboot the unit;	16bit			0	0	1	RW
Override configuration											
4x150	X		Damper Override Timeout	Time before returning to normal mode	16bit	min		120	0	600	RW
4x151	X		Damper Override Conf.	0 – Normal mode; 1 – Override control - Max open; 2 – Override control - Min open; 3 – Override control - 100% open; 4 – Override control - 100% closed	16bit			0	0	4	RW
Damper											
4x300	X		Execute Angle Calibration	0 – Do nothing; 1 – Start recalibration of the angle sensor; 2 – Start recalibration when starting up;	16bit			0	0	2	RW
4x302	X		Angle Set Point	Angle setpoint used in normal mode. (Only relevant when 4x070 is set to 1)	16bit	%		0	0	100	RW
4x314	X		Flow Set Point	Flow setpoint used in normal mode. (Only relevant when 4x070 is set to 2)	16bit	l/s		*	0	4700	RW

* = the value depends on the dimension of the product.