

Jarno Vähäylkkä

**POLTTOMOOTTORIGENERAATTORIN ETÄVALVOMON
SUUNNITTELU**

POLTTOMOOTTORIGENERAATTORIN ETÄVALVOMON SUUNNITTELU

Jarno Vähäylkkä
Opinnäytetyö
Kevät 2014
Automaatiotekniikan koulutusohjelma
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Automaatiotekniikan koulutusohjelma

Tekijä: Jarno Vähäylkkä

Opinnäytetyön nimi: Polttomoottorigeneraattorin etävalvomon suunnittelu

Työn ohjaaja: Tero Hietanen

Työn valmistuslukukausi ja -vuosi: Kevät 2014

Sivumäärä: 44

Tässä opinnäytetyössä oli tavoitteena suunnitella ja toteuttaa polttomoottorigeneraattorin tiedonsiirto Fidelixin alakeskukselle ja siitä Webvision-valvomo-ohjelmistolle luettavaan muotoon. Työssä myös suunniteltiin ja toteutettiin graafinen käyttöliittymä polttomoottorigeneraattorille Webvision-valvomo-ohjelmistoon. Työ tehtiin Oulun ammattikorkeakoulun energialaboratorioon. Työssä oli tarkoituksena toteuttaa tiedonsiirto käyttämällä lähiverkkoa, jolloin polttomoottorigeneraattorikonttia voidaan liikuttaa paikasta toiseen, kun konttiin ei ole vedettyä ylimääräisiä kaapeleita.

Mittaustietojen siirtäminen lähiverkon välityksellä osoittautui työläämmäksi tavaksi, kuin mittaustietojen siirtäminen suoraan alakeskukselle Modbus RTU -väylätekniikkaa hyödyntäen lopulta oli. Se kuitenkin valittiin toteutustavaksi, jotta välttyttiin useampien kaapeleiden vetämiseltä konttiin.

Opinnäytetyössä saavutettiin sille asetetut tavoitteet, sillä tiedonsiirron toteutus saatiin tehtyä hyödyntämällä koulun lähiverkkoa. Tiedonsiirrossa CAN-väylältä tulevat mittaustiedot muutetaan Modbus TCP -protokollaan, jonka avulla tieto saadaan siirrettyä lähiverkkoon ja siitä edelleen alakeskukselle. Myös toimiva käyttöliittymä saatiin toteutettua polttomoottorin mittaustiedoille

Asiasanat: automaatio, Fidelix, Webvision, kenttäväylät, Modbus, Canbus, tiedonsiirto

ALKULAUSE

Insinööri työ tehtiin Oulun ammattikorkeakoulun energiatekniikan laboratoriolle. Työn valvojana toimi Erkki Kylmänen energiatekniikan laboratorion johtajana ja ohjaajana opettajana tekniikan yksiköstä lehtori Tero Hietanen.

Kiitän Oulun ammattikorkeakoulun projektisuunnittelijaa Erkki Kylmästä opin-
näytetyömahdollisuuden antamisesta ja laitteiden liittämistä alakeskukseen
antaneesta avusta Fidelixin huoltoinsinööriä Ville Karhumaata sekä ohjauksesta
lehtori Tero Hietasta.

Oulussa 24.4.2014

Jarno Vähäylikkä

SISÄLLYS

TIIVISTELMÄ	3
ALKULAUSE	4
SISÄLLYS	5
TERMIT JA LYHENTEET	7
1 JOHDANTO	9
2 SÄHKÖN JA LÄMMÖN YHTEISTUOTTO	11
2.1 Sähkön tuotto projektissa	11
2.2 Pääkomponentit CHP-laitoksessa	13
2.2.1 Turbiini	13
2.2.2 Mäntämoottori	13
2.2.3 Generaattori	14
2.2.4 Lämmönvaihdin	14
2.2.5 Sytytin	14
3 TIEDONSIIRTO	16
3.1 Tiedonsiirron historia	16
3.2 Tiedonsiirto automaatiassa	17
3.3 CAN-väylä	18
3.4 Modbus-väylä	20
3.4.1 Modbus TCP/IP	21
3.4.2 Modbus ASCII	22
3.4.3 Modbus RTU	22
4 TYÖN SUORITUS	23
4.1 Laitteiden tilaus	24
4.2 Fidelix alakeskus Fx2025A	25
4.3 Tiedonsiirto työssä	26
4.4 Väylämuunnin	27
4.5 Muuntimen konfigurointi	29
4.6 IEC-ohjelma	32
4.7 Webvision	34

4.8 Käyttöliittymä	37
5 YHTEENVETO	40
LÄHTEET	42

TERMIT JA LYHENTEET

ACK	Acknowledgement, CAN-viestin kuittauskenttä
ASCII	American Standard Code for Information Interchange, 128 merkkipaikan laajuinen tietokoneiden merkkistö
CAN	Controller Area Network, tiedonsiirtoväylä
CHP	Combined heat and power, Sähkön ja lämmön yhteistuotto
CRC	Cyclic Redundancy Check, CAN-viestin tarkistussumma
EOF	End of frame, CAN-viestin lopetuskenttä
Ethernet	Pakettipohjainen lähiverkkoratkaisu, joka on yleisin ja käytetyin lähiverkkoratkaisu.
IDE	Identifier, CAN-viestin kehyksen tunnistekenttä
IEC	International Electrotechnical Commission, kansainvälinen sähköalan standardointiorganisaatio
IFS	Inter-frame separator, kehykset toisistaan erottava 3-bittinen kenttä
IP	Internet Protocol, TCP/IP-mallin Internet-kerroksen protokolla
ISO	International Organization for Standardisation, kansainvälinen standardisoimisjärjestö
LRC	Longitudinal Redundancy Check, tarkistussumma

Modbus	Sarjaliikenneprotokolla, jonka on kehittänyt Modicon vuonna 1979
NRZ	Non-return to zero, binäärikoodaus
OSI	Open Systems Interconnection Reference Model, tiedonsiirto-protokollien kuvays seitsemässä kerroksessa
PDU	Protocol Data Unit, pakettien sisältämä kehys, joka määrittelee paketin sisältämän tiedon
RTR	Remote Transmission Request, CAN-viestin kenttä, joka ilmaisee onko kyseessä kysely vai tietokehys
RTU	Remote Terminal Unit, väyläpohjainen tiedonsiirto-protokolla
SOF	Start of frame, CAN-viestin aloituskenttä
TCP	Transmission Control Protocol, tietoliikenneprotokolla tietokoneiden väliseen keskusteluun

1 JOHDANTO

Opinnäytetyön aiheena oli Oulun ammattikorkeakoululle tehty polttomoottorigeneraattorin tiedonsiirto alakeskukselle ja käyttöliittymän suunnittelu. Tiedonsiirrossa siirretään antureiden suorittamat mittaukset polttomoottorigeneraattorilta Fidelixin alakeskukselle Fx2025A. Tiedonsiirron lisäksi työhön sisältyi alakeskuksen liittäminen Webvision-valvomo-ohjelmistoon sekä käyttöliittymän suunnittelu ja toteuttaminen polttomoottorigeneraattorista Webvision-valvomo-ohjelmistoon.

Polttomoottorigeneraattoriprojekti oli osana koulun sähköverkkoon liitettävien, sähköenergiaa tuottavien laitteiden mittaustietojen tiedonsiirtoa. Muita hankkeeseen liittyviä laitteita olivat aurinkopaneelit ja maalämpöpumppu, mutta nämä eivät sisältyneet opinnäytetyöhön. Polttomoottorigeneraattorikontti on rahtikontin kokoinen metallikontti, johon polttomoottori ja generaattori on sijoitettuna. Kontti sijaitsee ammattikorkeakoulussa autolaboratorion takapihalla. Energialaboratoriolta saadun opinnäytetyöaiheen tiedonsiirtoa koskeva osuus katettiin suunnittelemalla polttomoottorigeneraattorin ja alakeskuksen välinen tiedonsiirtoyhteys.

Etäluettavien mittaustietojen tarkastelu oli tärkeä osa projektia, sillä tarkoituksena oli tietää, vastaavatko luetut tiedot odotuksia, ja mahdollisissa ongelmatilanteissa selvittää näiden perusteella ongelman sijaintia. Lisäksi Fidelixin Webvision-valvomo-ohjelmiston mittaushistoriasta pystytään tarkastelemaan pidemmällä aikavälillä saatuja mittaustietoja seurantaan ja analysointia varten.

Työn tavoitteena oli saada siirrettyä antureiden tekemät mittaustiedot polttomoottorigeneraattorilta Fidelixin Webvision-valvomo-ohjelmistoon luettavaksi, siten ettei polttomoottorigeneraattorikontiin vedettäisi enempää kaapeleita.

Työssä oli tarkoituksena myös saada mittaustiedot näkymään Webvision valvomo-ohjelmistossa sekä käyttöliittymässä. Käyttöliittymän suunnitteleminen ja toteutus oli myös osana opinnäytetyötä. Käyttöliittymässä tuli olla suuntaa antava

kuva polttomoottorigeneraattorista reaaliaikaisten mittaustietojen kanssa, josta saadaan helposti luettua mittaustiedot kuvan havainnollistamasta paikasta selityksineen.

2 SÄHKÖN JA LÄMMÖN YHTEISTUOTTO

Opinnäytetyöhön liittyvässä polttomoottorigeneraattorikontissa on käytössä polttomoottori ja generaattori, joilla voidaan tuottaa sähköenergiaa koulun sähköverkkoon. Teoriaosuudessa käsitellään yleisesti sähkön ja lämmön yhteistuottoa, joka liittyy hyvin koulun polttomoottorigeneraattorin teoriaan.

2.1 Sähkön tuotto projektissa

Sähkön ja lämmön yhteistuotolla (CHP) saadaan yhtäaikaisesti otettua talteen tuotettua sähkö- ja lämpöenergiaa. Sähkön ja lämmön yhteistuotto on erittäin tehokas energian tuottamismuoto, joka voi saavuttaa primäärienergian säästön noin 40-prosenttisesti verrattuna erikseen ostettuun sähköenergiaan sähköverkosta ja kaasulämmittimellä paikallisesti tuotettuun lämpöön. (1, s. 24–25.)

Yleisesti tuotettu sähköenergia käytetään laitoksessa, jossa polttomoottori sijaitsee ja tuotetun sähköenergian ylijäämä välitetään sähkön jakeluverkkoon ja alijäämä otetaan sähkön jakeluverkosta. Prosessissa tuotettua lämpöä käytetään joko tehtaan prosesseihin tai tilojen lämmittämiseen tai vaihtoehtoisesti siirretään lämmönjakeluun lähialueelle. (1, s. 24–25.)

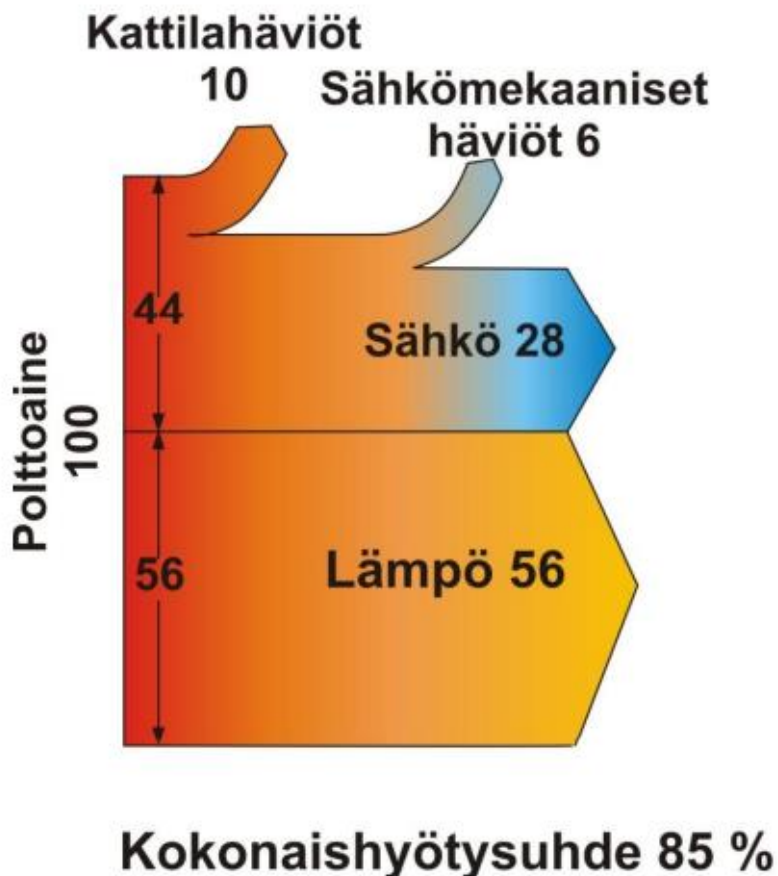
Höyryn lämpötilaa nostetaan käyttämällä kaasuturbiinin pakokaasuja hukkalämpökattilan läpi. Höyry syötetään joko vastapaineella tai jaettavalla höyryturbiinilla. Hyödynnettävä lämpö saadaan otettua talteen höyryturbiinilta. CHP-järjestelmän avulla yhdistetään eri vaiheissa 80–90 % polttoaineen energiasta, joka saadaan hyödynnettyä sähköenergiaksi tai käytettäväksi lämmöksi. (Kuva 1.)

CHP-järjestelmien monimutkaisuuden ja pääomakustannusten takia järjestelmät sopivat suurille sähköenergiaa ja lämpökuormia vaativille järjestelmille, kuten kunnille ja suurille teollisuuslaitoksille. (1, s. 24–25.)

Suuremmilla moottoreilla (noin 500 kW) tulee taloudellisemmaksi siirtää pakokaasut hukkalämpökattilalle, sillä pakokaasun lämpötila voi olla jopa 350–500 °C:ssa. Jäähdytysvaipasta ja voiteluöljystä saatava lämpöenergia on tyypillisesti

70-80 °C. Polttoaineena käytetään yleensä maakaasua, johon voidaan lisätä pieniä määriä polttoöljyä parantamaan palamista tai joissakin tapauksissa bio-kaasua jäteveden käsittelylaitoksilta. Tyypillisiä käyttökohteita ovat vapaa-ajan keskuksset, hotellit, sairaalat, koulut ja teollisuusprosessit. (1, s. 24–25.)

Tyypillisesti CHP-yksiköitä käytetään täyttämään sähköenergiatarpeet laitoksella, eikä ylijäämää tavoitella lähetettäväksi sähköverkkoon. On tavallista, että CHP-yksiköt voidaan ohjata vastaamaan haluttua lämpökuormaa kaukolämpöjärjestelmissä, jolloin lämmöntuotto vastaa vaadittua ympäristön lämpötilaa. Vaihtoehtoisesti yksiköitä voidaan ohjata vastaamaan haluttua sähkökuormaa tuottoaikalla ja kaikki alijäämä lämmöntuotossa hankitaan lisälähteestä. (1, s. 24–25.)



KUVA 1. CHP:n kokonaishyötysuhde (2, s. 3)

2.2 Pääkomponentit CHP-laitoksessa

Pääkomponentit yleisesti CHP-laitoksessa ovat

- turbiini
- mäntämoottori
- generaattori
- lämmönvaihdin
- sytytin

Opinnäytetyöhön liittyvässä sähköenergian tuottamisprosessissa ei kuitenkaan ollut käytössä kuin mäntämoottori ja generaattori. Prosessin tarkoituksena on ainoastaan tuottaa sähköenergiaa.

2.2.1 Turbiini

Kaasuturbiinin toiminnassa puristin puristaa ilmaa polttokammioihin, jolloin polttoaine palaa suurella paineella ja pakokaasut pyörittävät turbiinin siipiä, jolloin turbiinin keskellä oleva akseli pyörittää generaattoria. Kaasuturbiinin toiminnassa ilma päästetään jatkuvalla virtauksella puristimelle, joka on moderneissa turbiineissa aksiaalista virtausta. Painetta nostetaan kahden tai kolmen puristimen läpi, jotka ovat itsenäisiä ja toisistaan riippumattomia, joissa jokaisessa on oma turbiinilevynsä saman suuntaisilla akseleilla. Näin jokainen turbiinilevy voi pyöriä omalla nopeudellansa. (3, s. 198.)

2.2.2 Mäntämoottori

Nelitahtimoottorissa ensimmäisessä vaiheessa mäntä ottaa polttoainetta ja ilmaa sylinteriin, jonka jälkeen toisessa vaiheessa mäntä puristaa seoksen ja sytyttää sen joko sytyttimellä tai kasvavan paineen avulla. Kolmannessa vaiheessa, kun polttoaine laajenee palaessaan, se liikuttaa mäntää. Männän liikkeen kasvaessa saadaan liike-energiaa pyöriväksi liikkeeksi kiertokangen ja kampiakselin avulla. Neljännessä vaiheessa palamistuotteet poistetaan sylin-

teristä. Pyörivän liikkeen ansiosta kampiakselilla voidaan tuottaa liikettä generaattoriin roottorille. (3, s. 204.)

2.2.3 Generaattori

Generaattorissa on kaksi osaa, staattori ja roottori, joiden liikkeen ansiosta saadaan tuotettua sähköenergiaa. Staattori on paikallaan oleva osa, johon on kääritty käämi, ja pyöritettäessä magneettista roottoria staattorin sisällä indusoituu käämeihin sähkövirtaa. Näin ollen saadaan tuotettu sähköenergia siirrettyä käämiltä eteenpäin hyötykäyttöön. (4, s. 14.)

Synkronisen generaattorin toiminta perustuu Faradayn elektromagneettiseen induktiolakiin. Näin ollen generaattorin staattorin täytyy vaihtaa magneettista napaa vastustaen liikettä. Yleensä staattori on paikallaan ja roottori liikkuu. (4, s. 14.)

2.2.4 Lämmönvaihdin

Lämmönvaihtimella saadaan siirrettyä polttomoottorin tuottamaa lämpöenergiaa hyödylliseen käyttöön. Lämmönvaihtimen toiminta perustuu lämpötilaeroihin, jolloin kuumasta polttomoottorista siirtyy lämpöenergiaa lämmönvaihtimelle. (4, s. 14.)

Lämmönvaihdin kiinnitetään esimerkiksi pakokaasuputken ympärille, jolloin kuuma putki lämmittää lämmönvaihtimessa olevaa nestettä. Lämmönvaihtimessa oleva kuuma neste sekoittuu ympärillä olevan nesteen kanssa yrittäen tavoittaa lämpötilatasapainon. (4, s. 14.)

2.2.5 Sytytin

Palaminen normaalissa kaasuturbiinissa on jatkuva prosessi, jossa polttoaine palaa ilman kanssa. Sytyttimen antamaa kipinää tarvitaan aloittamaan palamisprosessi, jolloin sytyttimen pitää kyetä antamaan kipinä ilman ukoista apua. Mäntämoottorissa tapahtuva polttoaineen palaminen vaatii kipinän sytyttämään

polttoaineen moottorissa aina, kun mäntä on vetänyt kammion täyteen polttoainetta. (4, s. 15.)

3 TIEDONSIIRTO

Työssä siirrettiin mittaustietoja CAN-väylää pitkin Can to Modbus TCP -muuntimelle, josta tiedot siirrettiin lähiverkon kautta alakeskukselle.

Teoriaosuudessa käydään läpi tiedonsiirron historiaa ja kerrotaan tarkemmin työssä käytetyistä tiedonsiirtomenetelmistä.

Tiedonsiirto on tiedon siirtämisestä pisteestä pisteeseen tai useampaan pisteeseen kerralla. Signaali on elektromagneettista, kuten jännite, radioaalto tai infrapunasignaali. (5.)

3.1 Tiedonsiirron historia

Ensimmäisenä binäärisenä tiedonsiirtona voidaan olettaa tapahtuneen yli 2000 vuotta sitten, kun kreikkalaiset käyttivät tiedonsiirtoon soihtujen luomaa valokeilaa ja roomalaiset käyttivät siipiopastimia. Moderni historia digitaalisessa tiedonsiirrossa alkoi 1830-luvulla, kun sähköistä lennätintä alettiin käyttämään kaupallisesti tiedonsiirrossa. Tapoja oli keksittynä yli 60, kuinka lähettää viestejä, mutta Samuel Morse keksi elektromagneettiseen lennättimen, joka sitten hyväksyttiin universaaliksi tavaksi lähettää viestejä. Morsen alkuperäisessä tavassa oli listattuna sanoja paperille ja pulssien määrä kertoi, kuinka mones sana oli kyseessä paperilla. Myöhemmin kuitenkin Alfred Vailin avulla tämä tapa korvattiin pisteillä ja viivoilla, jolloin voitiin lähettää monimutkaisempia viestejäkin helpommalla. (6, s. 1–2.)

Puhelimen käyttö tiedonsiirrossa alkoi 1876, kun Alexander Graham Bell lähetti ensimmäisen lauseen assistentilleen. Ensimmäiset puhelit olivat alkeellisia vastaanottimia ja lähettimiä, mutta eivät molempia. 1880-luvulla puhelin alkoi hallitsemaan tiedonsiirtoa ja analogista tiedonsiirtoa kasvaen eniten käytetyimmäksi tiedonsiirtotavaksi. 1880 Bell demonstroi lähetystä valon avulla, mutta tätä tapaa ei silloin voitu alkaa käyttämään, koska ei ollut luotettavia valon lähteitä. 1970-luvulla, kun laserit ja optiset kaapelit olivat jo keksittynä, alkoivat

ensimmäiset testilähetykset digitaalisessa tiedonsiirrossa valon avulla, jonka jälkeen optinen tiedonsiirto alkoi kasvamaan rajusti. (6, s. 2–3.)

Radion kehittäminen alkoi 1800-luvun loppupuolella, kun italialainen Guglielmo Marconi alkoi kehittämään langatonta tiedonsiirtoa. Mukana oli useita keksijöitä ympäri maailmaa, mutta Marconin hakemaa patenttia pidetään ensimmäisenä radiopatenttina, minkä vuoksi Marconia pidetään radion keksijänä. Vuonna 1899 Marconi lähetti ensimmäisen viestin radioaalloilla Englannin kanaalin yli ja kahta vuotta myöhemmin Atlantin valtameren yli. (6, s. 3.)

Satelliittien käyttäminen tiedonsiirrossa alkoi 1962 kun ensimmäinen satelliitti lähetettiin avaruuteen. Tässä satelliitissa oli käytössä aktiiviset toistimet. Vuoteen 1970 mennessä kansainvälinen tietoliikennesatelliittiyhtiö INTELSAT oli saavuttanut maailmanlaajuisen telekommunikaatioverkon. (6, s. 4.)

Internetin kehittäminen alkoi 1950-luvun loppupuolella, kun Yhdysvallat halusi kehittää informaatioteknologiaa, joka voisi selvitä ydiniskusta. Vuonna 1969 kehitettiin Internetin edeltäjä ARPANET, joka toimi kolmen yliopiston ja yhden tutkimuslaitoksen välillä. Tämä mahdollisti viestien välittämisen verkon avulla. 1990-luvulla Internet kaupallistettiin ja siirrettiin operaattoreiden hoidettavaksi, minkä jälkeen Internetin suosio on kasvanut räjähdysmäisesti. (7.)

3.2 Tiedonsiirto automaatiassa

Automaatiassa tiedonsiirtoa voidaan toteuttaa langoitetulla automaatiolla, point-to-point-yhteydellä, kenttäväylillä, lähiverkolla ja langattomilla tiedonsiirtoyhteyksillä (8, s. 13).

- Langoitetussa automaatiassa sähköiset signaalit kuljettavat mittausviestit ja ohjausviestit antureilta automaatiojärjestelmän mittaus/ohjauskortille omalla johtimella. Tämä on perinteinen ja vieläkin hyvin yleinen tapa toteuttaa tiedonsiirtoa, vaikkakin se on hankala asentaa ja vaikeaa tehdä muutoksia. (8, s. 13.)

- Point-to-point-yhteydessä käytetään asynkronista sarjaväyläyhteyttä kahden laitteen välillä. (8, s. 13.)
- Lähiverkkoa käytetään yleisesti tehdastasolla tietokoneiden ja tietokonejärjestelmien väliseen kommunikointiin. Lähiverkkoon on helppo kytkeä ja poistaa laitteita ilman, että muut lähiverkossa olevat laitteet häiriintyvät. (8, s. 13.)
- Langattomassa tiedonsiirrossa käytetään langattomia lähettämiä ja vastaanottimia tiedonsiirtoon laitteelta toiselle. Tämä tapa vastaa langatonta point-to-point tiedonsiirtoa. (8, s. 13.)
- Kenttäväylät ovat tehokas tapa siirtää tietoa laitteilta toisille. Rakenteeltaan kenttäväylät ovat digitaalisia, kaksisuuntaisilla väyläliitynnöillä olevia tiedonsiirtoratkaisuja, joita yhdistää älykkäät mittaus- ja ohjauslaitteet. Kenttäväyläratkaisut perustuvat tehokkaaseen tietoliikenteeseen ja älykkäisiin kenttälaitteisiin. Prosessiautomaatiossa luotettavuudella on tärkeä merkitys, joten esimerkiksi antureiden ja toimilaitteiden toiminnan jatkuva seuranta on välttämätöntä. (9, s. 32)

Kenttäväylät tuovat mukanaan hajautetun arkkitehtuurin, jossa huomattava osa toiminnoista siirtyy kenttälaitteiden suoritettavaksi. Yksittäisillä väylään liitetyillä laitteilla on valmius toteuttaa useita sellaisia perustoimintoja, jotka vielä nykyisinkin hoidetaan järjestelmässä. (10.) Seuraavaksi tutustutaan kahteen yleiseen kenttäväyläratkaisuun, CAN-väylään ja Modbus-väylään.

3.3 CAN-väylä

CAN-väylä on automaatiioväylä, jota on kehitetty vuodesta 1983 lähtien, jolloin ajoneuvojen elektroniikkaan suuntautunut yritys Bosch uudisti ajoneuvojen tarvitsemia johdotuksia yksinkertaisemmiksi. Tällöin ei ollut saatavilla protokollia, jotka olisivat täyttäneet vaatimukset, joten yritys päätti suunnitella uuden väylän tiedonsiirtoa varten. CAN-väylä julkaistiin vuonna 1986, jolloin väylää alettiin käyttää ajoneuvoissa, koneissa ja teollisuuslaitteissa. Nykyään lähes kaikissa Euroopassa valmistetuissa ajoneuvoissa käytetään CAN-väylää. (11.)

CAN-väylässä kaikki liikenne välitetään kaikille moduuleille ja jokaisella viestillä on sanomatunniste, jonka perusteella vastaanottava moduuli päättää kuuluuko viesti kyseiselle laitteelle. Tämän ansiosta sama viesti voidaan vastaanottaa useammalla laitteella, esimerkiksi autossa nopeusmittaus voidaan lähettää samanaikaisesti nopeusmittarille ja vakionopeudensädölle. Väylän monitorointi on helppoa, koska ylimääräinen moduuli voidaan lisätä väylälle ilman, että sillä olisi mitään vaikutusta muihin toimintoihin. Sanomatunniste on myös samalla viestin prioriteetti, jolloin kahta viestiä yhtäaikaisesti lähetettäessä alemmalla prioriteetilla oleva luopuu lähetyksestä, kunnes suuremmalla prioriteetilla ollut viesti on lähetetty. (11.)

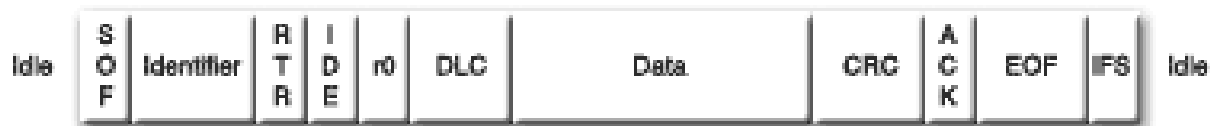
CAN-protokolla määrittelee siirtokerroksen OSI-mallissa ja osan fyysisestä kerroksesta. (Kuva 2.) Kansainvälinen standardisointijärjestö (ISO) määrittelee standardin, joka sisällyttää CAN-väylän määritelmän osaksi fyysistä kerrosta, sillä fyysinen signalisointi sisältää koodauksen ja dekodeuksen (Non Return to Zero, NRZ) sekä bittien ajoituksen ja synkronoinnin. (12.)



KUVA 2. OSI-malli (mukaillen 13)

CAN-protokolla tukee kahdenlaisia kehysformaatteja. Ainoa merkittävä ero näissä formaateissa on tunnisteiden pituus. Lyhyemmässä CAN-väylän kehystunnisteessa pituus on 11 bittiä (CAN 2.0A) ja pidemmässä kehystunnisteessa 29 bittiä (CAN 2.0B). (12.)

Kehyksessä ensimmäistä bittiä kutsutaan aloitusbitiksi (Start of Frame, SOF), jota seuraa tunnistekenttä (Identifier), jossa määritellään viestin prioriteetti. Tämän jälkeen seuraa RTR-bitti, jota käytetään erottamaan datakehys ja datan pyyntökehys toisistaan. CAN 2.0B -kehystunnisteessa arbitration-kenttä sisältää RTR-bitin ja 29 tunnistebittiä. Näihin tunnistebitteihin kuuluu IDE, jonka arvo on 1. Datan pituus (DLC) käytetään osoittamaan datan pituutta data-kentässä. Kehyksen eheys tarkistetaan CRC-tarkistussummalla. Kuittauskenttä (ACK) on kahden bitin mittainen kuittauskenttä, joka koostuu kahdesta bitistä ACK slotista ja ACK delimiteristä. Viestikehyksen vastaanottaneet solmut kirjoittavat tähän dominantin bitin. EOF-kenttä on lopetuskenttä, joka on seitsemän bitin mittainen, jossa kaikki seitsemän bittiä ovat resessiivisiä. IFS on kolmen bitin mittainen kenttä, joka erottaa kehykset toisistaan. (12.) Kuvassa 3 on näytetty kenttien paikat kehyksessä.



KUVA 3. CAN-kehys (12)

CAN-väylä käyttää NRZ-linjakoodausta, jossa 1-bitit esitetään yhtenä merkitseväänä tilana ja 0-bitit toisena merkitseväänä tilana. NRZ-koodauksessa bitit eivät palaa johonkin perustilaan, jolloin signaali voi pysyä pitkään samana. (14.)

3.4 Modbus-väylä

Modbus on vuonna 1979 julkaistu tiedonsiirtoprotokollaperhe, joka on alun perin ohjelmoitavien logiikoiden liittämiseen tarkoitettu avoimeen arkkitehtuurin perustava väylä. Modbus mahdollistaa isäntä-orja- tai asiakas-palvelin-kommunikoinnin älykkäiden laitteiden välillä. (15.)

Modbus-protokollaa käytetään laajasti teollisuuden sovellusten lisäksi rakennuskohteissa, energian optimointijärjestelmissä, automaatiojärjestelmissä, pitkän matkan tiedonsiirrossa ja ohjauspaneelien yhdistämisessä. Modbus on myös

ideaali protokolla RTU-menetelmille, jossa langatonta kommunikointia vaaditaan. Tästä syystä Modbusia käytetään myös lukemattomissa kaasu- ja öljyaseilla. Modbus on edullinen tapa liittää eri valmistajien laitteita keskenään valmiilla yhteisellä protokollalla. Modbusista on olemassa kolme erilaista tiedonsiirtoprotokollaa, ASCII, RTU ja TCP. (15.)

3.4.1 Modbus TCP/IP

Modbus TCP/IP on yleinen tietoliikenneprotokolla internetissä ja se on joukko kerroksellisia protokollia, jotka tarjoavat luotettavan tiedonsiirtomekanismin koneesta toiseen. Yksinkertaistettuna Modbus TCP/IP on Modbus RTU TCP/IP-liitynnällä, joka mahdollistaa tiedonsiirron lähiverkossa sopivien laitteiden välillä. Modbus TCP/IP yhdistää fyysisen verkon ja verkkotyöskentelyn TCP/IP-standardilla. (15.)

Lähiverkosta on tullut de facto -standardi yritysten järjestelmiin, joten siitä on myös tullut standardi tehtaiden verkkoihin. Lähiverkko ei ole uutta tekniikkaa, vaan se on kasvanut pisteeseen, jossa kulut verkon liittämistä ovat pienentyneet kilpailemaan muiden kenttäväylien kanssa. (15.)

Modbus TCP/IP:stä on tullut kaikkialla läsnäoleva, koska se on avoin, yksinkertainen, edullinen kehittää ja koska se tarvitsee vähän laitteita tukemaan sitä. Markkinoilla on olemassa useita satoja Modbus TCP/IP-laitteita ja uusia kehitetään joka vuosi lisää. Modbus TCP/IP:tä käytetään informaation vaihtoon laitteiden välillä, monitorointiin ja ohjelmoimaan laitteita. Sitä käytetään myös hallinnoimaan jaettuja I/O-laitteita ja se on valmistajien suosittelu protokolla laitteisiin. (15.)

Modbus TCP/IP:n huomattavia etuja ovat yksinkertaisuus, tavallisen lähiverkon käyttö, saatavuus monille laitteille ja avoimuus, jolloin se ei tarvitse lisenssiä tai mitään muitakaan maksuja (15).

3.4.2 Modbus ASCII

Modbus ASCII on puhtaasti tekstipohjainen ja ihmisten ymmärrettävässä muodossa. Siinä on käytössä 256 erilaista merkkiä. ASCII-muodossa viesti lähetetään 8-bittisenä tavuna, jossa on kaksi ASCII-merkkiä. (16.)

Tiedon siirtämiseen Modbus ASCII -formaattissa käytetään sarjaliikennettä. ASCII-formaatissa käytetään LRC-tarkistussummaa tietojen oikeellisuuden tarkistamiseen, joka ei kuitenkaan ole kovin tehokas virheidentarkistukseen. ASCII-viestit ovat yksinkertaisempia ja helpommin ymmärrettävissä, mutta niiden hyötysuhde on alhainen verrattuna muihin Modbus-protokolliin. (16.)

3.4.3 Modbus RTU

Modbus RTU on yleisesti käytetty ja avoin sarjaprotokolla isäntä-orja -arkkitehtuurilla, jota käytetään laajasti rakennusten hallintajärjestelmissä ja teollisuuden automaatiojärjestelmissä. Laajan suosion takana on Modbus RTU:n helppokäyttöisyys ja avoimuus. Modbus RTU käyttää binäärikoodausta ja tiedon eheyden tarkistukseen CRC -tarkistussummaa. Helppokäyttöisyyden ansiosta Modbus RTU on paljon yksinkertaisempi kytkeä laitteisiin, kuin uudemmat protokollat ja tämän ansiosta Modbus RTU hallinnoi markkinoita. Yksinkertaisuuden ansiosta 16-bittistä Modbus RTU:n rekisterirakennetta voidaan käyttää liukulukujen, taulujen, ASCII-tekstien, jonojen ja muiden tietojen lähettämiseen. Modbus RTU:lla on binääriesitys laitteiden välisen pakettien vaihdon (PDU) verkon välillä, joka on riippumaton taustalla olevista viestikeroista. (17.)

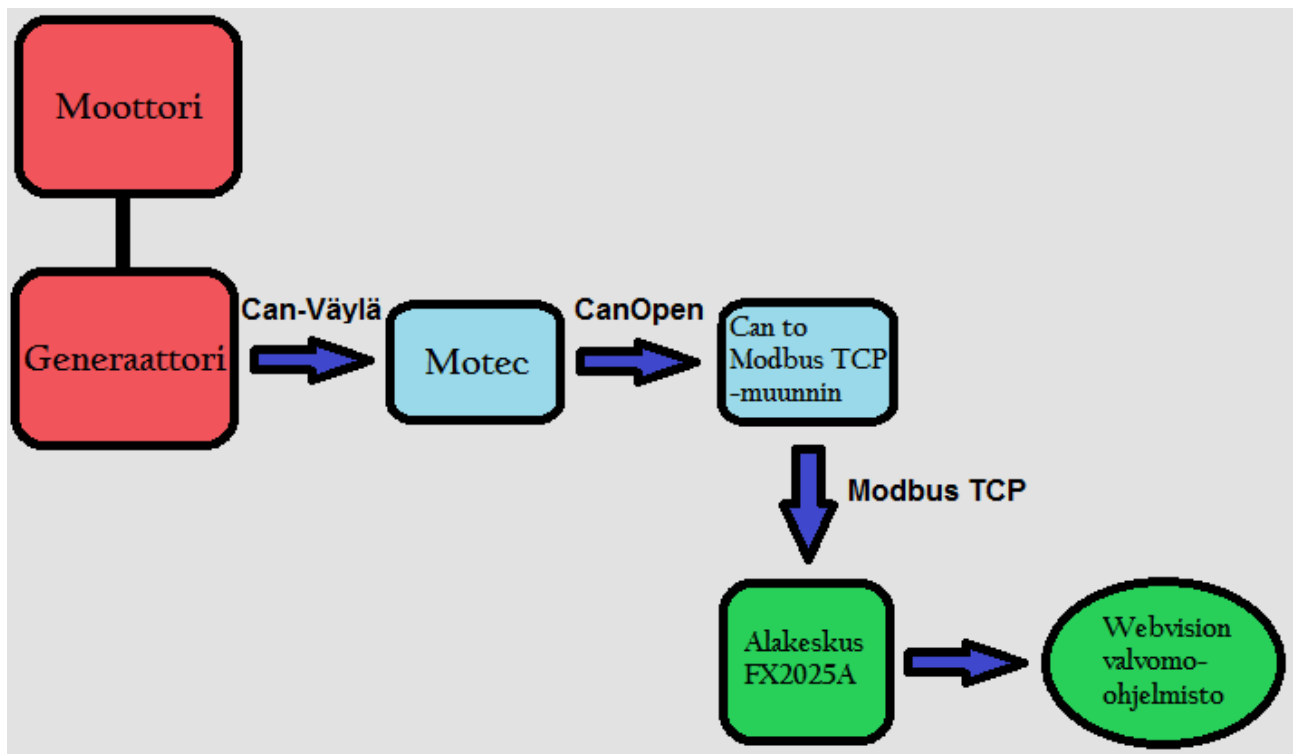
Modbus-väylää pidetään OSI-mallissa sovelluserroksena, joka tarjoaa isäntä-orja -viestinnän yhteen kytkettyjen laitteiden välillä. Isäntälaitte kysyy yhdeltä tai useammalta orjalaitteelta tietoja. Tämä tarkoittaa sitä, ettei orjalaitte voi lähettää tietoja ilman että siltä kysytään. Isäntälaitte kirjoittaa orjalaitteen rekisteriin tietoja ja lukee tietoja rekisteristä. (17.)

4 TYÖN SUORITUS

Polttomootorissa ja generaattorissa on useita antureita, joilla mitataan laitteiden toimintaa ja saadaan tuotetun sähköenergian määrä. Antureilla mitataan polttomootorin polttoaineen syötön määrä ja venttiilin asento syötölle, ilmansyöttö polttomootorille ja venttiilin asento, jäähdyttimen lämpötila, polttomootorin öljynpaine, pakosarjan paine, polttomootorin kierrosnopeus ja pakokaasun lämpötila. Generaattorista mitataan generaattorin syöttöteho ja syöttövirta.

Saadut mittaustiedot siirrettiin CAN-väyliä pitkin moottorinohjain Motecille, jossa mittaustiedot yhdistetään CANopen-väylälle, jolloin kaikki mittaustiedot saatiin siirrettyä samaa johdinparia pitkin eteenpäin muuntimelle. Can to Modbus TCP -muuntimella muunnettiin CAN-väylä Modbus TCP -väylälle, jolloin mittaustiedot saatiin siirrettyä alakeskukselle lähiverkkoa pitkin alakeskuksen ymmärrettävään muotoon. Muuntimeen määritettiin erikseen muunnettavat kehykset CAN-väylältä Modbus TCP -väylälle. Alakeskukselle tehtiin IEC-ohjelma, jonka avulla saatiin alakeskus ymmärtämään haettavat Modbus TCP protokollan mukaiset kehykset muuntimelta. Kun mittaustiedot saatiin siirrettyä alakeskukselle luettavaan muotoon voitiin ne lukea myös Webvision valvomo-ohjelmistolla ja käyttöliittymällä.

Kuvassa 4 on esitelty tiedonsiirron rakenne moottorilta Webvision-valvomo-ohjelmistolle.



KUVA 4. Tiedonsiirto

4.1 Laitteiden tilaus

Polttomoottorigeneraattorilta tulevat mittaustiedot käyttivät kenttäväylänä CAN-väylää, josta tiedonsiirto piti toteuttaa Fidelixin alakeskukselle Fx2025A, joka pystyy ainoastaan vastaanottamaan Modbus RTU- ja Modbus TCP -väyläliikennettä. Modbus RTU -väylää ei voitu käyttää, koska polttomoottorigeneraattorikontille ei haluttu vetää ylimääräisiä kaapeleita, ja useamman laitteen käyttäminen lisää mahdollisuuksia laitteiden rikkoontumiselle. Lisäksi useamman laitteen konfiguroinnissa olisi saattanut kestää huomattavasti kauemmin. Lisäksi keskustellessani Fidelixin huoltoinsinöörin kanssa päädyimme, siihen CAN-väylän muuntaminen Modbus TCP:hen on kannattavampi vaihtoehto, sillä IEC-ohjelman tekemiseen löytyy hyvin ohjeita Fidelixin ohjesivustolta. Päädyin toteuttamaan tiedonsiirron muuntamalla väyläliikenteen Modbus TCP -protokollan mukaiseksi. Muuntaminen CAN-väylältä Modbus TCP -väylälle toteutettiin käyttämällä ADFweb.com S.r.l.:stä tilattua CAN To Modbus TCP

Slave
-muunninta.

4.2 Fidelix alakeskus Fx2025A

Opinnäytetyössä käytettiin Fidelix Fx2025A -alakeskusta (kuva 5), joka on vapaasti ohjelmoitava alakeskus, johon voidaan kytkeä Modbus RTU-485 ja RS-232 -sarjaväylää käyttäviä moduuleita. Lisäksi voidaan käyttää Ethernet-väylää, jolloin alakeskukselle lisätään IEC-ohjelma, joka muuntaa tulevan Modbus TCP -liikenteen lähiverkosta alakeskukselle ymmärrettävään muotoon. Alakeskuksen käyttöjärjestelmä perustuu teollisuus-PC:hen ja Windows CE - käyttöjärjestelmään, jota käytetään selainpohjaisella liitynnällä, joka tekee alakeskuksen käytöstä helppoa. Alakeskusta voidaan käyttää myös 10,4”-n kosketusnäytöllä. Alakeskuksen näyttöön on integroitu läsnäolotunnistin, joka sammuttaa ja käynnistää näytön automaattisesti, jolloin näytön elinikää saadaan pidennettyä huomattavasti. (18.)



KUVA 5. Fx2025A-automaatiojärjestelmä (18)

Alakeskus kytkettiin Webvision-valvomo-ohjelmistoon käyttämällä Fidelix FdxOnline-ohjelmaa, joka hoitaa kommunikoinnin alakeskuksille. Ohjelmaan määritettiin alakeskuksen nimi, IP-osoite ja portit, jolloin taustalla pyörivä FdxOnlineService-ohjelma hoitaa alakeskukselle kommunikointia itsenäisesti. Kuvassa 6 nähdään alakeskukselle FdxOnline-ohjelmaan määritetyt asetukset.

OPETUSSALKKU		
▶	IPADDRESS	193.167.101.142
	WEBPORT	80
	WEBVISIONPORT	1235
	DEFAULTUSER	
	DEFAULTPASSWORD	
	SHOWENERGY	<input type="checkbox"/>
	ENERGYGROUP	
	SHOWCALENDAR	<input type="checkbox"/>
	CALENDARNAME	
	TIMEOUT	
	USERS	
	GROUPS	

KUVA 6. Alakeskuksen asetukset Webvision valvomo-ohjelmistolle

4.3 Tiedonsiirto työssä

Opinnäytetyössä päätettiin käyttää tiedonsiirtoon Ethernet-yhteyttä, jolloin välttyttiin useamman kaapelin vetämiseltä polttomoottorigeneraattorille. Tällä tavalla saadaan säilytettyä generaattorin liikuteltavuus, jolloin polttomoottorigeneraattorikonttia voidaan liikuttaa tarvittaessa paikasta toiseen ilman suurempia muutoksia tiedonsiirron toteutukseen.

Polttomoottorigeneraattorilla on useita eri antureita mittaamassa tietoja, joilla tarkkaillaan moottorin ja generaattorin toimintaa käytön aikana. Moottorin ja generaattorin mittaustietojen siirrossa käytetään CAN-väylätekniikkaa, joka

mahdollistaa useiden mittaustietojen siirtämisen samassa kehyksessä. Polttomootorigeneraattorilta CAN-väylältä tulevat mittaustiedot kerätään CANopen-väyläteknikalla, josta mittaustulokset vietään CAN to Modbus TCP Slave -muuntimelle, jonka avulla tieto saadaan muunnettua Modbus TCP -protokollaan.

CAN-viestejä ei voitu viedä suoraan Fidelixin alakeskukselle sillä alakeskusta ei ole ohjelmoitu ymmärtämään CAN-väylän viestejä. Näin ollen viestit täytyi muuntaa Modbus TCP -protokollan mukaisiksi viesteiksi. Alakeskuksella Modbus TCP -viestit voitiin lukea käyttämällä IEC-ohjelmaa, jolloin alakeskus ymmärsi viestit ja ne saatiin luettavaan muotoon.

4.4 Väylämuunnin

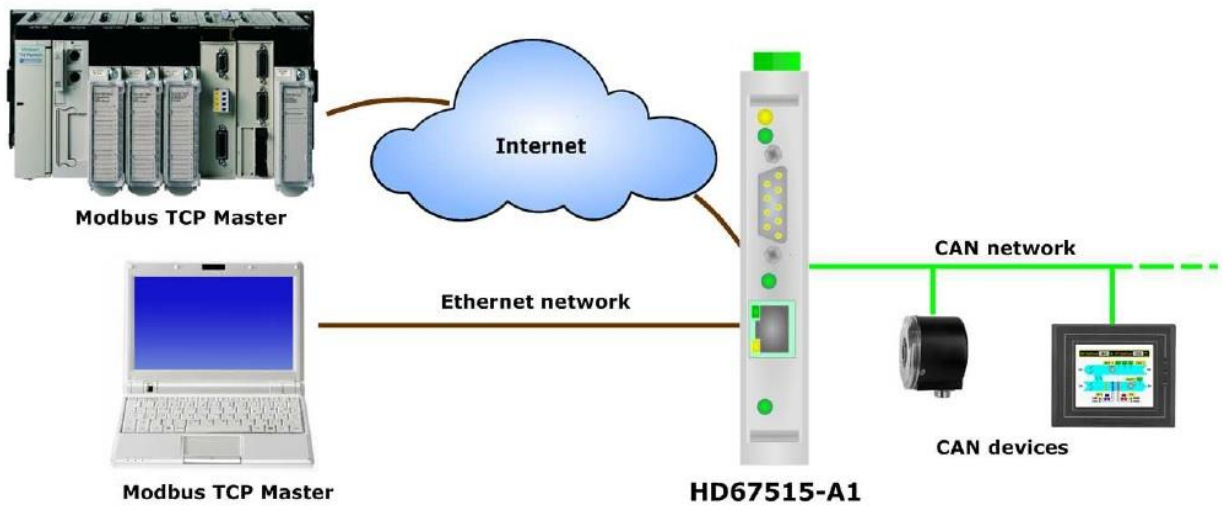
Mittausviestien muuntamiseen käytettiin tilattua Can To Modbus TCP Slave -muunninta, jolla saadaan CAN-väylältä tulevien mittauslaitteiden mittaustietoviestit muunnettua Modbus TCP -protokollan mukaisiksi, jolloin ne voidaan lukea lähiverkon kautta. Kuvassa 7 on esiteltyä yhteysesimerkki tiedonsiirron toteutukseen ja kuvassa 8 nähdään muuntimen liityntöjen selitykset.

Muuntimen asetuksia päästään muuttamaan kytkemällä tietokone muuntimen COM-porttiin. Muuntimella on seuraavanlaisia ominaisuuksia:

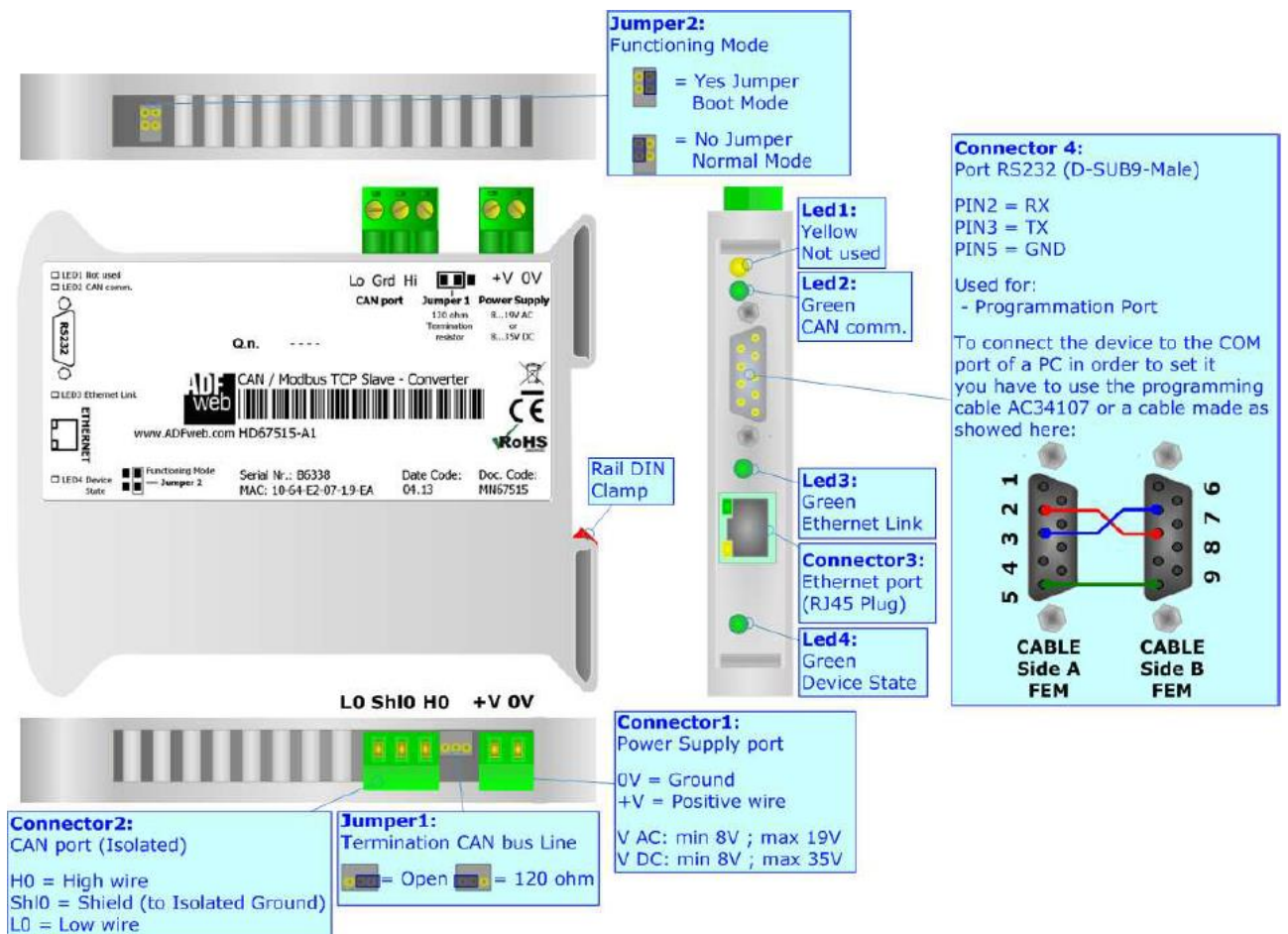
- kaksisuuntainen tiedonsiirto CAN-verkon ja ModBus-verkon välillä
- sähköinen eristys väylien välillä
- voi lukea CAN-kehiksen ModBus-sanasta
- voi kirjoittaa CAN-kehiksen ModBus-sanasta
- voidaan kiinnittää DIN-kiskoon
- käyttölämpötila -40 °C – +85 °C.

Laite voi toimia vaihtojännitteellä AC 8 V – 19 V tai tasajännitteellä 8 V – 35 V.

(19.)



KUVA 7. Yhteysesimerkki (16)



KUVA 8. Liityntä (16)

4.5 Muuntimen konfigurointi

Muunnin konfiguroidaan valmistajan kehittämällä SW67515-ohjelmalla. Muunnin kytketään tietokoneeseen käyttämällä RS232 COM -kaapelia. SW67515-ohjelmalla määritetään laitteelle yhteysparametrit, CAN-kehysten ID-tunnisteet ja lähetettävät Modbus TCP -kehykset.

Yhteysasetuksilla määritellään muuntimen CAN-väylältä vastaanottamien viestien pituus (CAN Bus 2.0A) ja käytetty tiedonsiirtonopeus, joka on tässä työssä 1000. Yhteysasetuksiin määriteltiin laitteelle lähiverkkoasetukset, jotka on hankittu ammattikorkeakoulun IT-tuesta. (Kuva 9.)

SET COMMUNICATION

CAN Bus

Baud rate: 1000

CAN Bus 2.0A (CobID 11Bit)
 CAN Bus 2.0B (CobID 29Bit)

TimeOut (s): 3

Ethernet

IP Address: 193 . 167 . 101 . 222

SubNet Mask: 255 . 255 . 255 . 0

Default Gateway
 193 . 167 . 101 . 254

Port: 502

Fast StratUp (only 100Mb/s)

KUVA 9. Yhteysasetukset

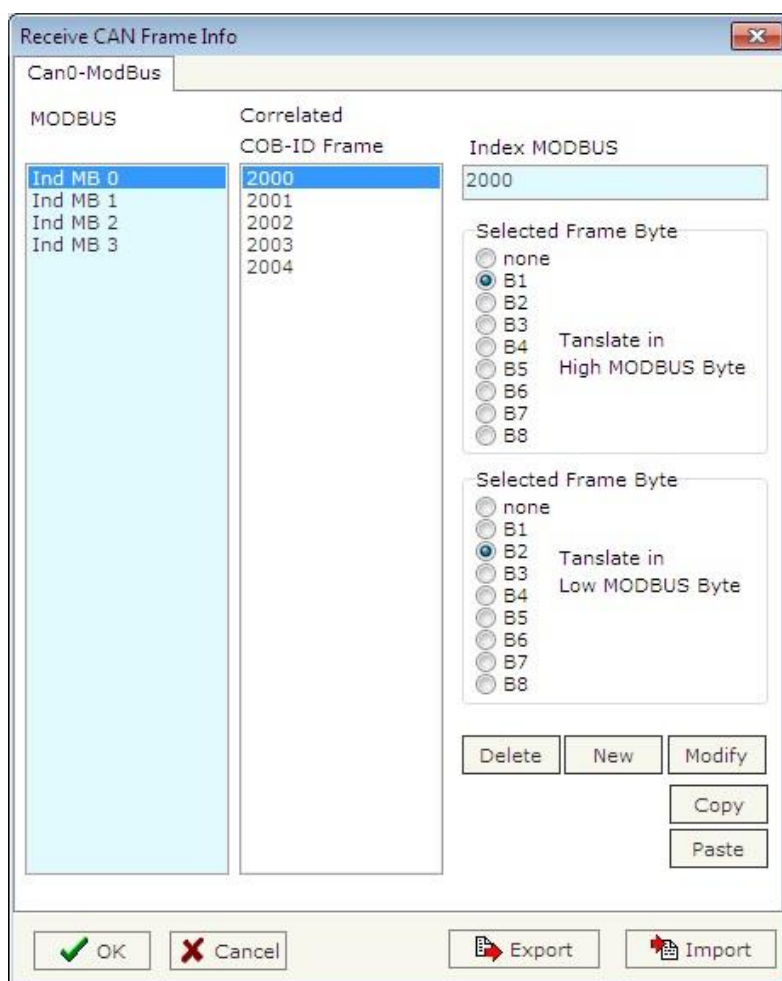
Ohjelmaan määritellään CAN-kehukset, joissa mittausviestit saapuvat. Poltto-moottorigeneraattorin CAN-asetuksiin oli määritelty kehukset 2000–2004, jotka sitten määriteltiin myös muuntimen asetuksiin. Kehysten pituudeksi oli määritelty 8, jolloin jokainen kehys sisältää 4 eri mittautietoa. (Kuva 10.)

Receive CAN Frame

N	CobID	Dimension	TimeOut	Mnemonic
1	2000	8	<input type="checkbox"/>	
2	2001	8	<input type="checkbox"/>	
3	2002	8	<input type="checkbox"/>	
4	2003	8	<input type="checkbox"/>	
5	2004	8	<input type="checkbox"/>	
6			<input type="checkbox"/>	

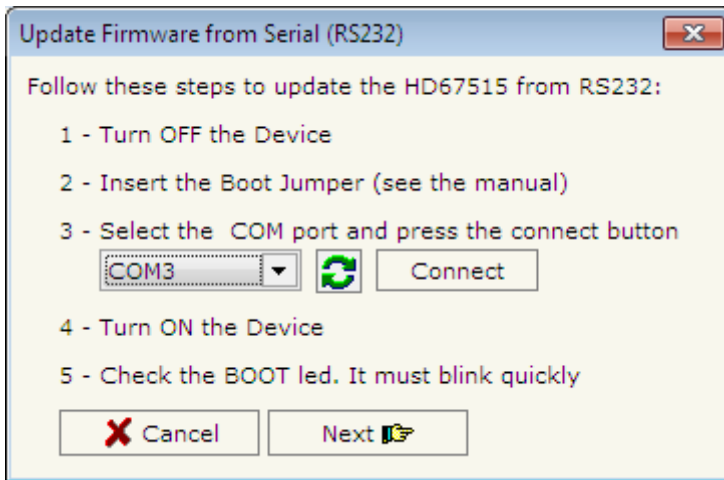
KUVA 10. Määritetyt CAN-kehukset

Jokaiselle kehyksen sisältämälle neljälle mittaustiedolle määritettiin Modbus index, jonka perusteella alakeskus voi hakea tietyn viestin muuntimelta. Valituilla kehyksen biteillä määritetään missä kohtaa kehystä haluttu mittaustieto sijaitsee. (Kuva 11.)

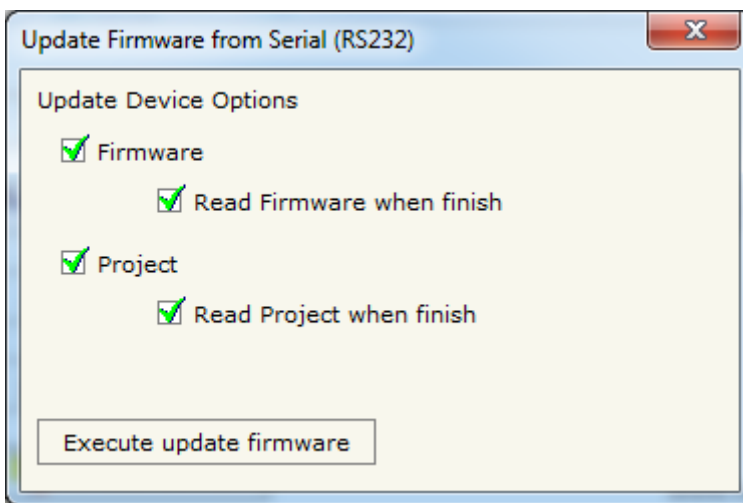


KUVA 11. Kehykseen määritetyt mittaustiedot

Kun yhteysasetukset ja kehysasetukset on määritelty ladattiin ohjelma muuntimelle. Muuntimen asetuksia päivittäessä voitiin päivittää myös muuntimen firmware eli laiteohjelmisto. (Kuva 12 ja kuva 13.)



KUVA 12. Asetusten päivittäminen



KUVA 13. Firmwaren päivittäminen

4.6 IEC-ohjelma

Alakeskuksen sovelluksen tekemiseen ei minulla ollut minkäänlaista kokemusta, jolloin lähdin perehtymään sovelluksen tekemiseen käyttämällä Fidelixin tukisivustolla olevia esimerkkiohjelmia.

Alakeskukselle on luotu Infoteamin OpenPCS -ohjelmalla IEC-ohjelma, jonka avulla pystytään lukemaan Modbus TCP-viestejä. OpenPCS automaatio-ohjelma edustaa IEC61131-1 yhteensopivaa ohjelmointiympäristöä. Ohjelmalla voidaan yhdistää saumattomasti kaikki CANopen-palvelut ja ominaisuudet verkosta käteviin funktiolohkoihin asti kaikkiin CANopen-palveluihin. (20.)

OpenPCS-ohjelman ominaisuudet:

- täydellinen CANopen-integrointi IEC-ohjelmointiympäristöön
- useiden ohjelmointikielien yhdistäminen yhteen projektiin
- integroitu offline-simulaattori
- yhteys PLC:hen RS232, CAN-väylällä tai lähiverkon kautta
- laajat debuggaus-ominaisuudet (20.)

Valmis ohjelma ladataan alakeskukselle, jolloin se alkaa välittömästi käsitellä määriteltyjä mittaustietoja.

Muuntimen rekistereiden lukemiseen väylällä käytettiin toimilohko-ohjelmaa, joka on yleisesti käytetty logiikkaohjelmoinnissa. Ohjelmaan ei ole erikseen määritetty rekisteriarvoja, joista tiedot luetaan vaan toimilohkokirjastossa on määriteltynä aloitusrekisteri, josta rekisterien lukeminen alkaa.

Kuvassa 14 on esiteltynä työssä käytetty toimilohko-ohjelma, jossa tiedot luettiin id_holding_reg -rekistereistä ja suoritetaan mittauservolle joko kerroin tai jakaminen, jolloin mittaustiedot saadaan näkymään järkevässä muodossa.

```
(* luetaan holding rekisterit *)
HLS34 (Send:=0, Module:=modbus_address, StartRegister:=2000, RegisterType:=3, Port:=serial_port);
if HLS34.datavalid = 1 then
  HLS34.Reg0 := HoldingRegF(id_String := id_holding_reg_02, RegValue := HLS34.Reg0, rDivider := 100.0);
  HLS34.Reg1 := HoldingRegF(id_String := id_holding_reg_03, RegValue := HLS34.Reg1, rDivider := 10.0);
  HLS34.Reg2 := HoldingRegF(id_String := id_holding_reg_04, RegValue := HLS34.Reg2, rMultiplier := 1.0);
  HLS34.Reg3 := HoldingRegF(id_String := id_holding_reg_05, RegValue := HLS34.Reg3, rMultiplier := 1.0);
  HLS34.Reg4 := HoldingRegF(id_String := id_holding_reg_06, RegValue := HLS34.Reg4, rDivider := 1000.0);
  HLS34.Reg5 := HoldingRegF(id_String := id_holding_reg_07, RegValue := HLS34.Reg5, rMultiplier := 1.0);
  HLS34.Reg6 := HoldingRegF(id_String := id_holding_reg_08, RegValue := HLS34.Reg6, rDivider := 10.0);
  HLS34.Reg7 := HoldingRegF(id_String := id_holding_reg_09, RegValue := HLS34.Reg7, rMultiplier := 100.0);
  HLS34.Reg8 := HoldingRegF(id_String := id_holding_reg_10, RegValue := HLS34.Reg8, rDivider := 10.0);
  HLS34.Reg9 := HoldingRegF(id_String := id_holding_reg_11, RegValue := HLS34.Reg9, rDivider := 10.0);
  HLS34.Reg10 := HoldingRegF(id_String := id_holding_reg_12, RegValue := HLS34.Reg10, rDivider := 10.0);
  HLS34.Reg11 := HoldingRegF(id_String := id_holding_reg_13, RegValue := HLS34.Reg11, rMultiplier := 1.0);
  HLS34.Reg12 := HoldingRegF(id_String := id_holding_reg_14, RegValue := HLS34.Reg12, rMultiplier := 1.0);
  HLS34.Reg13 := HoldingRegF(id_String := id_holding_reg_15, RegValue := HLS34.Reg13, rMultiplier := 1.0);
  HLS34.Reg14 := HoldingRegF(id_String := id_holding_reg_16, RegValue := HLS34.Reg14, rMultiplier := 1.0);
  HLS34.Reg15 := HoldingRegF(id_String := id_holding_reg_17, RegValue := HLS34.Reg15, rMultiplier := 1.0);
  HLS34.Reg16 := HoldingRegF(id_String := id_holding_reg_18, RegValue := HLS34.Reg16, rMultiplier := 1.0);
  HLS34.Reg17 := HoldingRegF(id_String := id_holding_reg_19, RegValue := HLS34.Reg17, rMultiplier := 1.0);
  HLS34.Reg18 := HoldingRegF(id_String := id_holding_reg_20, RegValue := HLS34.Reg18, rMultiplier := 1.0);
  HLS34.Reg19 := HoldingRegF(id_String := id_holding_reg_21, RegValue := HLS34.Reg19, rMultiplier := 1.0);
end_if;
```

KUVA 14. Toimilohko-ohjelma.

4.7 Webvision

Webvision on valvomo-ohjelmisto, jolla voidaan hallita Fidelixin rakennusautomaatiojärjestelmiä. Ohjelmalla pystytään keräämään tietoja useista alakeskuksista ja hallitsemaan keskitetysti etänä toisistaan sijaitsevia alakeskuksia.


Käyttöliittymän pystyy rakentamaan käyttäjälle sopivaksi, jolloin halutut tiedot saadaan näkyväksi toivotulla tavalla. Itse käyttöliittymä ei tarvitse erillistä ohjelmistoa vaan toimii internet-selaimella. (21.)


Ominaisuudet:


- Useiden alakeskusten historiatietojen, lokien ja energiaraporttien sekä prosessikuvien katselu
- Helppokäyttöinen selainpohjainen käyttöliittymä
- Mahdollisuus kolmannen osapuolen tietojärjestelmien integrointi (OPC, ODBC, XML, Access ja webservice)
- SQL-historiatietokanta
- Monipuolinen raportointi
- Usean yhtäaikaisen käyttäjän tuki (21.)


Webvisionista alakeskukselle määriteltiin Modbus TCP -portti porttiin 6 (kuva 15), josta viestejä luetaan

Hälytykset


 Finnish


 English


 Swedish


 Dutch

Asetukset

Portit

Porttien asetukset

Portti 0	MBUS			
Portti 1	Modeemi		<input type="text" value="9600"/>	<input type="button" value="v"/>
Portti 3	<input style="width: 100px;" type="button" value="v"/>	<input style="width: 30px;" type="button" value="v"/>	<input type="text" value="N82"/>	<input style="width: 30px;" type="button" value="v"/>
Portti 4	<input style="width: 100px;" type="button" value="v"/>	<input style="width: 30px;" type="button" value="v"/>	<input type="text" value="N82"/>	<input style="width: 30px;" type="button" value="v"/>
Portti 5	<input style="width: 100px;" type="button" value="v"/>	<input style="width: 30px;" type="button" value="v"/>	<input type="text" value="N82"/>	<input style="width: 30px;" type="button" value="v"/>
Portti 6	<input style="width: 100px;" type="button" value="v"/>	Lähetysviive (ms) <input type="text" value="0"/>		
Portti 7	<input style="width: 100px;" type="button" value="v"/>			
Portti 8	<input style="width: 100px;" type="button" value="v"/>			
Portti 9	<input style="width: 100px;" type="button" value="v"/>			
Portti 10	<input style="width: 100px;" type="button" value="v"/>			

M-BUS lähetysviive (ms)

Debug GSM modem

KUVA 15. Porttien asetukset Webvisionissa

Modbuslaitteet välilehdestä määritetään porttien asetuksiin laitettu porttinumero, Modbuslaitteen osoite, rekisterien tyyppi, aloitusrekisteri ja rekisterien määrä (kuva 16). Lisäksi määriteltiin Canbus to Modbus TCP -muuntimelle annettu IP-osoite ja portti.

Hälytykset

Ma 24.03.2014 15:09:13 Modbus laitteet Päivitä

Ohjelmointi

06.002 3=HOLDING 2000 20 Port=6 Kommunikaatio Ok Errors 0.02%

Modbuslaitteet

Lisää

Korjaa

Poista

Porttinumero 6 Modbuslaitteen osoite 2

Rekisterien tyyppi	3=HOLDING REGS (READ=3,WRITE=16)	TCP/IP moduli	<input checked="" type="checkbox"/>
Aloitusrekisteri	2000	IP osoite	193.167.101.222
Rekisterien määrä	20	IP portti	502
Debug	<input type="checkbox"/>	Vain lähetys	<input type="checkbox"/>

Sanomia yhteensä	601110
Virheitä nyt	0
Virheitä enimmillään	16
Virheitä yhteensä	130
Viimeisin virhekoodi	Timeout
Vastaanoton odotusaika (ms)	300

Tyhjennä kaikkien modulien virhelaskurit

KUVA 16. Modbuslaitteiden asetukset Webvisionissa

Mittaukset välilehdestä määritellään mittauspisteet, jotka on määritelty alakeskuksen IEC-ohjelmassa. Tunnuksen perusteella ohjelma osaa hakea sitä vastaavan mittautiedon ja näyttää sen mittautuloksen. Mittauspisteisiin lisättiin kuvaava teksti selkeyttämään mistä mittauksesta on kyse. (Kuva 17.)

Hälytykset

Ma 24.03.2014 15:13:11 Mittauspisteet Päivitä

Ohjelmointi

Suodatus

Mittaukset

Lisää

Korjaa

Poista

01_2000_FM Mittauspiste	Akun Jännite	13.1 V Ohjelma
01_2001_FM Mittauspiste	Jäähdyttimen lämpötila	23.5 °C Ohjelma
01_2002_FM Mittauspiste	Moottorin kierrosnopeus	0.0 RPM Auto
01_2003_FM Mittauspiste	Moottorin öljynpaine	924.0 kPa Ohjelma
01_2004_FM Mittauspiste	Pakokaasun Lambda-arvo	0.0 La Auto
01_2005_FM Mittauspiste	Pakosarjan paine	988.0 kPa Ohjelma
01_2006_FM Mittauspiste	Generaattorin suhteväite	0.0 A Auto

Tunnus 01_2000_FM Teksti Akun Jännite Taso: Katselu 0

Moduli 00.000 Piste 0 Taso: Käsihjaus 0

Taso: Ohjelmointi 0

Globaalipiste Asetusarvo

Laskuri Analoginen

Yksikkö V Näytteenottoväli (sek) 60

Desimaaleja 1 Toleranssi 0.1

Muunnostaulukko Korjaus 0 Aikavakio (sek) 0

Kuva Avaa Auto

KUVA 17. Mittaukset-välilehti

4.8 Käyttöliittymä

Käyttöliittymä on liityntä, jonka kautta ihminen voi olla vuorovaikutuksessa koneen kanssa. Tavoitteena käyttöliittymällä on parantaa ja tehostaa vuorovai-
kutusta käyttäjän ja koneen kanssa. Käyttöliittymän antama palaute helpottaa
operaattorin toimintaa päätöksiä tehdessä. (22.)

Mittaustietojen tarkasteluun vaadittiin käyttöliittymä, joka luotiin käyttämällä
Fidelix Oy:n FdxHtmlEdit-ohjelmaa, joka on suunniteltu Fidelixin Webvision-val-
vomo-ohjelmiston grafiikkakuvien luontiin. Ohjelmaa käytettiin luomaan sivu,
jossa näytetään polttomoottorigeneraattorilta tulevat mittauksiedot sekä suuntaa
antava kuva polttomoottorigeneraattorista. Grafiikkakuvat luotiin käyttämällä
sovelluksen omia kuvia ja lisäämällä itse tehtyjä kuvia projektin WWW-
hakemiston Symbols-alihakemistoon.

Pistetunnus

Painonappi
 Numerokenttä
 Pylväsnäyttö
 Teksti
 Kuva
 Symboli

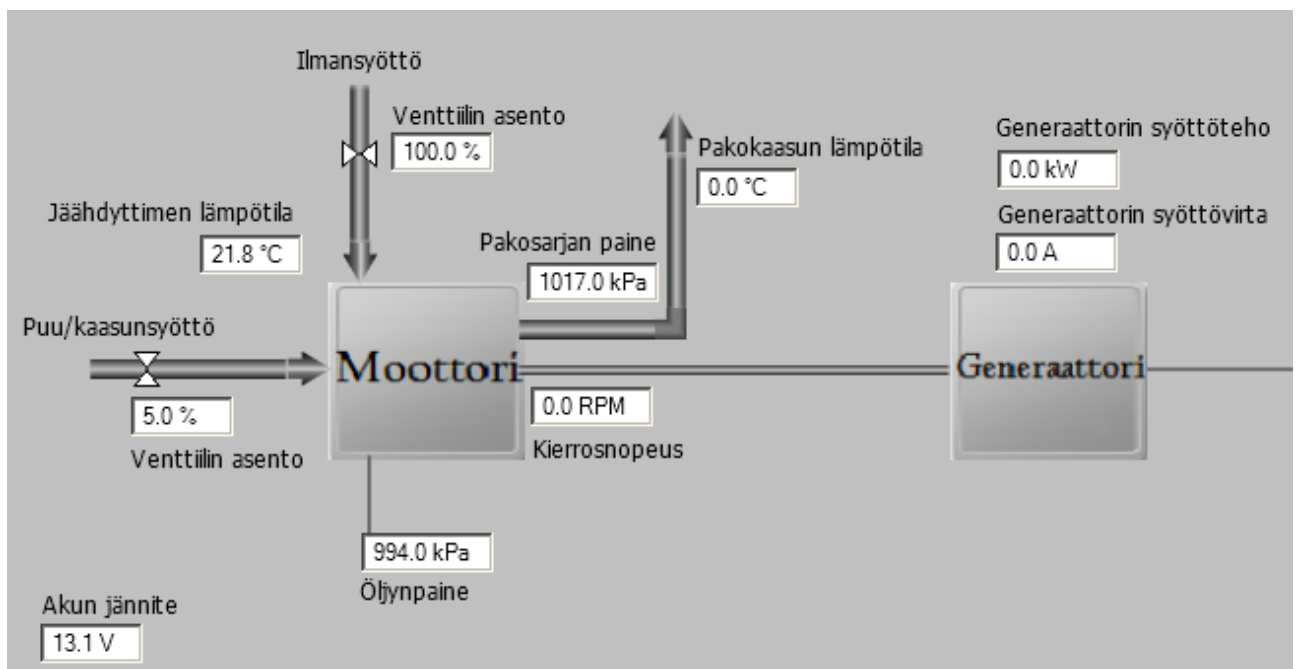
Säädön asetusarvo

Taustaväri
 Edustaväri
 CSS Luokka
 Käyttäjätaso

X Y
 Leveys Korkeus
 Näytä info

KUVA 18. Pistetunnuksen linkittäminen objektiin

Jokaiselle näytettävälle mittaustiedolle täytyi määritellä sama pistetunnus, joka on myös määriteltynä alakeskukselle kyseessä olevalle mittaustiedolle (kuva 18). Tällä menetelmällä mittaustiedot saatiin näkymään suoraan kuvassa reaaliajassa (kuva 19).



KUVA 19. Polttomoottorigeneraattorin grafiikka

Grafiikkakuvien valmistuttua käyttöliittymä tallennettiin alakeskukselle, jolloin pienen viiveen jälkeen käyttöliittymä oli nähtävissä Webvision valvomo-ohjelmistossa.

5 YHTEENVETO

Työssä oli tarkoituksena toteuttaa tiedonsiirtoa polttomoottorigeneraattorilta alakeskukselle ja suunnitella käyttöliittymä polttomoottorigeneraattorille Webvision valvomo-ohjelmistoon. Tähän kuului tutkia tarkoitukseen sopiva tapa siirtää antureiden mittaustiedot ilman ylimääräisten kaapeleiden vetämistä polttomoottorigeneraattorille, sekä toteuttaa selkeä käyttöliittymäkuva polttomoottorigeneraattorista Webvision-valvomo-ohjelmistoon.

Tiedonsiirto saatiin toteutettua vaaditulla tavalla reaaliaikaisesti ja luotettavasti toimivaksi ilman, että uusia kaapeleita vedettiin polttomoottorigeneraattorikontille. Tämä vastaa tiedonsiirrolle asetettuja vaatimuksia. Polttomoottorigeneraattoriin on suunnitteilla lisätä antureita, joiden mittaustietoja tulee myös siirtää alakeskukselle. Uusien CAN-väylien lisääminen ei ole kuitenkaan kovin yksinkertaista, sillä tähän vaaditaan muuntimen asetusten päivittämistä vastaanottamaan ja lähettämään uusilla kehyksillä, IEC-ohjelman päivittämistä vastaanottamaan uusia Modbus TCP-kehyksiä ja Webvisioniin uusien mittauspisteiden lisäämistä. Itse suosittelisin käyttämään CAN-kehyksiä, jotka ovat jo määriteltynä muuntimeen, IEC-ohjelmaan ja Webvision-valvomo-ohjelmistoon, sillä kuudessa eri mittauspaikassa on laitettuna CAN-väylä lähettämään nollaa.

Käyttöliittymä on suunniteltuna vastaamaan asetettuja odotuksia. Graafinen kuva näyttää yksinkertaistetun kuvan polttomoottorigeneraattorista mittaustietoineen. Mittaustiedot päivittyvät reaaliajassa ja sijaitsevat kuvassa niille ominaisilla paikoilla selitysteksteineen.

Loppujen lopuksi työ kokonaisuudessaan vastaa asetettuja vaatimuksia monien ongelmien jälkeen. Olisin itse vielä halunnut liittää loputkin anturit siirtämään mittaustietoja alakeskukselle, mutta näitä antureita ei ollut vielä kytkettynä polttomoottorigeneraattoriin.

Opinnäytetyötä tehdessäni opettelín paljon uusia ja hyödyllisiä asioita tulevaisuutta varten. IEC-ohjelman oppiminen oli erittäin mielenkiintoinen ja haastava,

ja sen opettelemiseen menikin aikaa. Alakeskuksen ja valvomo-ohjelmiston käyttäminen toi mielenkiintoisia haasteita työn suorittamiseen, mutta opittuani käyttämään niitä alkoi työ sujua erinomaisesti. Työtä tehdessäni perehdyin enemmän CAN-väylän ja Modbus-väylän toimintaan ja niiden rakenteeseen.

LÄHTEET

1. Kirschen, Daniel – Jenkins, Nick – Crossley, Peter – Allan, Ron – Strbac, Goran 2000. Embedded Generation. Lontoo: The Institution of Electrical Engineers.
2. Flyktman, Martti 2013. Sähkön ja lämmön yhteistuotanto biomassasta. VTT. Saatavissa: http://www.vtt.fi/files/news/2013/13062013/06_Sahkon_ja_lammon_yhteistuotanto.pdf. Hakupäivä 17.4.2014.
3. Milton, Brian 1995. Thermodynamics, Combustion and Engines. Lontoo: Chapman & Hall.
4. Fritzsche, Mario 2005. Combined Heat and Power. Saatavissa: http://www.mariofritzsche.de/resources/Assesment+Thermo_1.2.pdf. Hakupäivä 22.4.2014.
5. Huang, Yih 2006. Communication Theory. George Mason University. Saatavissa: <http://www.cs.gmu.edu/~huangyih/455/comm.pdf>. Hakupäivä 3.5.2014
6. Smith, David 2003. Digital Transmission Systems. New York: Springer.
7. Bogren, Sue – Erlinger, Kevin – Hari, Jan – Walleghen, Pam Van 1999. History of the Internet. University of Illinois. Saatavissa: <http://education.illinois.edu/wp/commercialism/history-of-the-internet.htm>. Hakupäivä 22.4.2014
8. Peltola, Jukka 2013. Tiedonsiirto – Kenttäväylät. Espoo: Aalto yliopisto, sähkötekniikan ja automaation laitos. Saatavissa: https://noppa.aalto.fi/noppa/kurssi/elec-c1210/luennot/ELEC-C1210_luentokalvot.1501_automatio_1_s13_tiedonsiirto_ja_kenttavaylat.pdf. Hakupäivä 22.4.2014
9. Piikkilä, Veijo 2006. Kiinteistöjen tiedonsiirtoväylät. Espoo: Sähköinfo Oy.

10. Uuspää, Pentti 1998. Kenttäväylä ja standardointi. Automaatioväylä nro. 8. Saatavissa: <http://www.automaatiovayla.fi/index.php/lehti/lehtiarkisto/25-lehdet-1998/98-lehti-8-1998>. Hakupäivä 3.4.2014.
11. CAN history. 2014. CAN in Automation. Saatavissa: <http://www.can-cia.de/index.php?id=161>. Hakupäivä 1.4.2014.
12. CAN protocol. 2014. CAN in Automation. Saatavissa: <http://www.can-cia.de/index.php?id=systemdesign-can-protocol>. Hakupäivä 16.4.2014
13. OSI-malli. 2013. Wikipedia. Saatavissa: <http://fi.wikipedia.org/wiki/OSI-malli>. Hakupäivä 21.4.2014
14. CAN physical layer. 2014. CAN in Automation. Saatavissa: <http://www.can-cia.de/index.php?id=systemdesign-can-physicallayer>. Hakupäivä 16.4.2014
15. Modbus FAQ: About the Protocol. 2014. Modbus Organization. Saatavissa: <http://www.modbus.org/faq.php>. Hakupäivä 5.4.2014.
16. What's the difference between Modbus ASCII and Modbus RTU. 2008. ProSoft Technology. Saatavissa: <http://www.prosoft-technology.com/kb/article.php?id=382>. Hakupäivä 4.4.2014.
17. Modbus RTU Overview. 2009. Real Time Automation. Saatavissa: <http://www.rtaautomation.com/modbusrtu/>. Hakupäivä 1.4.2014.
18. Fx2025A Digital Controller. Fidelix Oy. Saatavissa: http://www.fidelix.fi/documents/tuki/FX2025A_EN.pdf. Hakupäivä 27.3.2014.
19. CAN / Modbus TCP Slave - Converter. 2013. ADFweb.com S.r.l. Saatavissa: http://www.adfweb.com/download/filefold/Can_Modbus_Server_TCP_MN67515_ENG.pdf. Hakupäivä 25.3.2014.
20. OpenPCS Automation Suite. 2014. SYS TEC electronic GmbH. Saatavissa: <http://www.systec-electronic.com/en/products/automation->

[components/development-and-configuration-tools/openpcs-iec-61131-3-automation-suite](#). Hakupäivä 25.3.2014.

21. Webvision valvomo-ohjelmisto. Fidelix Oy. Saatavissa:

http://www.fidelix.fi/documents/tuki/webVision_FI.pdf. Hakupäivä: 27.3.2014

22. Margaret, Rouse 2005. User interface (UI). TechTarget. Saatavissa:

<http://searchsoa.techtarget.com/definition/user-interface>. Hakupäivä
2.5.2014