



TAMPEREEN
AMMATTIKORKEAKOULU

KOMPRESSORIYKSIKÖN OHJAUSJÄRJES- TELMÄN JA KÄYTTÖLIITTYMÄN SUUNNIT- TELU

Juha-Petteri Suominen

Opinnäytetyö
Huhtikuu 2016
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Kone- ja tuotantotekniikka
Kone- ja laiteautomaatio

SUOMINEN, JUHA-PETTERI:

Kompressoriyksikön ohjausjärjestelmän ja käyttöliittymän suunnittelu

Opinnäytetyö 38 sivua, joista liitteitä 2 sivua
Huhtikuu 2016

Opinnäytetyö tehtiin osana Tampereen Konepajat Oy:n tuotekehitysprojektia, jonka tarkoituksena oli kehittää suurpaineekompressoriyksikkö rakennusteollisuuden tarpeisiin. Työn tavoitteena oli suunnitella laitteelle ohjausjärjestelmä sekä käyttöliittymä, jonka avulla ruuvikompressoria, dieselmoottoria sekä muita käytettyjä komponentteja saadaan ohjattua ja seurattua halutulla tavalla. Työssä keskityttiin logiikkaohjelmointiin automaatioväylän laitteille.

Työn tuloksena saatiin määritettyä tarvittavat komponentit laitteelle sekä toteutettua logiikkaohjelmaa, jonka avulla laitteen suoritusarvoja pystytään monitoroimaan sekä muuttamaan. Oleellisena osana työtä oli dieselmoottorin ohjausyksikön tietojen käsittely sekä niiden järkevän esitystavan toteuttaminen. Toteutetun käyttöliittymän kautta esitetään käyttäjän kannalta tärkeät moottorin suoritusarvot kuten kierrosluku ja polttoainemäärä. Tämän lisäksi näytöllä ilmaistaan mahdolliset vikailmaisut ja sen kautta voidaan lukea kaikki moottorin ohjausyksiköltä tulevat vikakoodit.

Aikataulullisista syistä toteutettua ohjelmaa ei päästy testaamaan tämän työn puitteissa valmiilla laitteella, mutta kehitystyötä tullaan jatkamaan tuotekehitysprosessin loppuun saakka. Laitteen valmistuttua ohjausjärjestelmää tullaan varmasti jatkokehittämään testitulosten ja käyttökokemusten pohjalta.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Mechanical and Production Engineering
Machine Automation

SUOMINEN, JUHA-PETTERI:

Control System and User Interface Design for a Compressor Unit

Bachelor's thesis 38 pages, appendices 2 pages

April 2016

This thesis was made as part of Tampere Machinery Ltd's research and development project that aims to develop high pressure compressor unit for construction industry's needs. The aim of this thesis was to design control system and user interface to control screw compressor and diesel engine. The work focused on logic programming for automation bus devices.

During this project, components such as sensors were selected and logic program was made to control and observe performance of the engine and screw compressor. A big part of this project was to process the large amount of data that engine management system provides. User interface was designed so that the user can easily observe the critical engine parameters. Also all the fault indications and fault codes are displayed on the user interface.

The schedule of this project didn't give chance to test the designed logic program with functioning motor and compressor so all the testing was done with simulation. Development of the control system will continue until the end of the project. The program will also, most likely, need further development after testing and collecting user feedback.

Key words: research and development, logic programming, compressor

SISÄLLYS

1	JOHDANTO.....	6
2	CAN -VÄYLÄ	7
2.1	Esittely	7
2.2	Fyysinen rakenne	7
2.3	Tiedonsiirto	9
2.3.1	Data- ja etäkehys	10
2.3.2	Virhekehys	11
2.3.3	Ylikuormituskehys	12
2.4	Ylemmän tason protokollat.....	12
2.4.1	CANopen	12
2.4.2	SAE J1939.....	13
3	CODESYS.....	14
3.1	IEC 61131-3.....	14
3.1.1	Program organisation unit (POU)	15
3.1.2	Function block diagram (FBD)	16
3.1.3	Ladder diagram (LD)	17
3.1.4	Instruction list (IL)	17
3.1.5	Structured text (ST).....	18
3.1.6	Sequential function charts (SFC)	19
4	SUUNNITTELU	21
4.1	Suunnittelun lähtökohta	21
4.1.1	IFM electronics	21
4.1.2	Scania DC16	23
4.2	Ohjausjärjestelmän suunnittelu.....	24
4.2.1	Scania EMS8.....	26
4.2.2	Hätäpysäytys	26
4.3	Käyttöliittymän suunnittelu	27
5	TOTEUTUS	30
5.1	Ohjausjärjestelmä.....	30
5.2	Käyttöliittymä	31
6	POHDINTA.....	35
	LÄHTEET.....	36
	LIITTEET	37
	Liite 1. Käyttöliittymän varoitussivut	37
	Liite 2. Ruuvien pyörimisnopeuden suhde ilmamäärään.....	38

ERITYISSANASTO tai LYHENTEET JA TERMIT (valitse jompikumpi)

ACK	acknowledgement
CAN	controller area network
CiA	can in automation
Codesys	controller development system
CRC	cyclic redundancy field
EMS	engine management system
EOF	end of frame
IFS	interframe space
J1939	ylemmän tason CAN -standardi
PLC	programmable logic controller
RTR	remote transmission request
SOF	start of frame

1 JOHDANTO

Opinnäytetyön tarkoituksena oli suunnitella kehitysvaiheessa olevalle kompressoriyksikölle ohjausjärjestelmä sekä käyttöliittymä. Työ on osa Tampereen Konepajat Oy:n tuotekehitysprojektia, jonka tarkoituksena on suunnitella ja valmistaa suurpaineekompressoriyksikkö rakennusteollisuuden tarpeisiin.

Projektille oli ennalta määritetty käytettävä ruuvikompressorin ilman tuoton sekä paineen tarpeen pohjalta, myös kompressorin käyttövoimana toimiva Scanian dieselmoottori oli valittu ennalta. Tämän lisäksi logiikkatoimittajaksi oli valikoitunut aikaisempien käyttökokemusten pohjalta saksalainen IFM electronics gmbh. Muiden komponenttien osalta valinnat tehtiin työn edetessä.

Tiedonsiirto Scanian dieselmoottorin ohjausyksikön ja ohjausjärjestelmän logiikan välillä tapahtuu CAN -väylää käyttäen J1939 protokollan avulla. Suurimmaksi haasteeksi työssä koettiin moottorin ohjausyksikön suuren tietomäärän käsittely sekä järkevä esitystapa käyttöliittymässä. Suunniteltavaa kompressoriyksikköä tullaan käyttämään ulkona vaihtelevissa sääolosuhteissa, joten käyttöliittymän täytyy olla selkeä luettavuudeltaan sekä käytettävissä myös hanskat kädessä.

Työssä käsitellään CAN -väylän toimintaa sekä Codesys -ohjelmointiympäristöä, jonka avulla valitut IFM electronics:n logiikat ohjelmoidaan. Codesys -ohjelmointiympäristön toiminta pohjautuu kansainväliseen IEC 61131-3 standardiin, jossa käsitellään PLC -ohjelmointiin liittyviä asioita. Tämän lisäksi työssä otetaan kantaa muihin koneen suunnitteluun liittyvissä asioissa kuten hätäpysäytyspiirin rakenteeseen ja siihen mitä vaatimuksia konedirektiivi ja standardit niistä antavat.

2 CAN -VÄYLÄ

2.1 Esittely

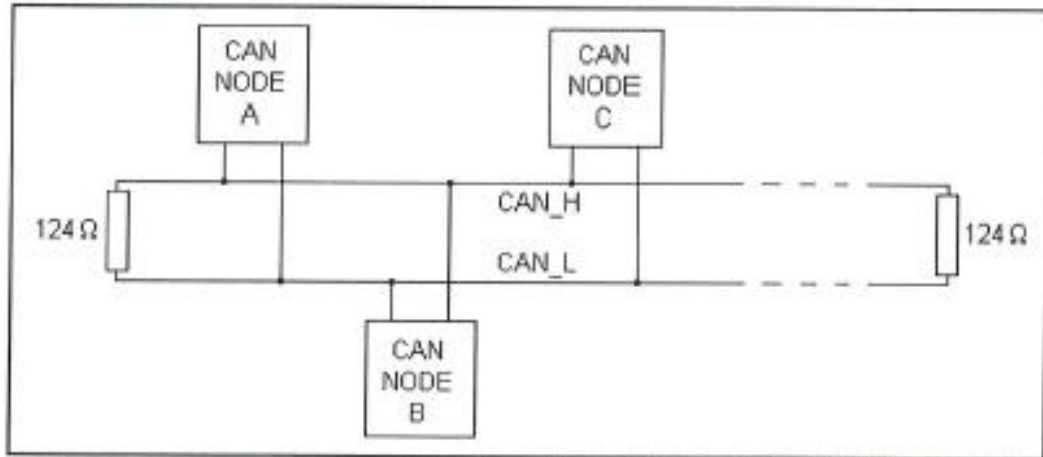
CAN -väylä on sarjaväylä, joka on alun perin kehitetty ajoneuvojen ohjausjärjestelmän käyttöön 1980 -luvun alussa. Protokollaa kuvaava standardi ISO 11898 on julkaistu vuonna 1993. (Farsi & Barbosa 2000, 41–42.) Nykypäivänä CAN -väylän käyttö on levinnyt laaja-alaisesti autoteollisuudesta muun muassa meri-, lääke- ja koneteollisuuden käyttöön. Laaja levinneisyys ja käyttö takaa laitteiden halvan hinnan sekä hyvän saatavuuden. (Voss 2005, 9–10.)

CAN -väylä on usean isännän (multi-master) väylä, jolla tarkoitetaan, että usea väylälle liitetty laite voi yrittää milloin tahansa lähettää tietoa väylälle. CAN -väylä on suunniteltu erityisesti sovelluksiin jossa tarvitaan nopeita vasteaikoja, luotettavaa virheentunnistusta sekä nopeaa virheiden korjausta. (Voss 2005, 2–3.)

Vuonna 1992 perustettiin CAN in automation (CiA) yhteisö, jonka tehtävänä oli kerätä yhteen ja julkaista tietoa CAN -väylän toiminnasta ja kehityksestä. Tällä hetkellä yhteisöön kuuluu yli 600 yritystä ympäri maailmaa. CiA toimii yhteistyössä standardointi organisaatioiden kanssa kehittääkseen CAN -väylään liittyviä standardeja jäsentensä parannusehdotusten pohjalta. (CAN in automation 2016.)

2.2 Fyysinen rakenne

CAN -väylän rakenne koostuu tyypillisesti parikaapelista, jotka ovat nimetty CAN_L ja CAN_H kuten kuvassa 1 on esitetty. Parikaapelin kummassakin päässä on päätevastukset, jotka ovat resistanssiltaan 100–130 ohmia. Päätevastuksia tarvitaan, jotta väylällä liikkuva signaali ei heijastu kaapelien päästä takasin väylälle. (Voss 2005, 132–133.)



KUVA 1. CAN -väylän rakenne (Farsi & Barbosa 2000, 43)

Väylän pituus tulisi pitää mahdollisimman lyhyenä ja yksinkertaisena, sillä maksimi tiedonsiirtonopeus väylällä riippuu väylän pituudesta. Väylän maksimi tiedonsiirtonopeus on 1 Mbit/s, jotta tähän tiedonsiirtonopeuteen päästään tulisi väylän pituuden olla alle 40 metriä. Väylän pituutta voidaan kasvattaa huomattavasti alentamalla tiedonsiirtonopeutta. (Voss 2005, 134.)

Kaapeleiden osalta CAN -väylä ei vaadi tavallisissa olosuhteissa käytettäväksi vaipalla suojattuja kaapeleita, sillä parikaapeleihin indusoituu sama jännite, jolloin kaapeleiden välinen jännite-ero pysyy samana ja tiedonsiirto ei häiriinny. Joissakin sovelluksissa parikaapelin lisäksi vaaditaan maadoituskaapeli. CAN in automation yhteisön julkaisussa DS-102 suositellaan käytettäväksi liittiminä kuvan 2 mukaista 9 -pinnistä D-Sub liittintä. (Voss 2005, 141.)



KUVA 2. D-sub liittin (Voss 2005, 141–142)

2.3 Tiedonsiirto

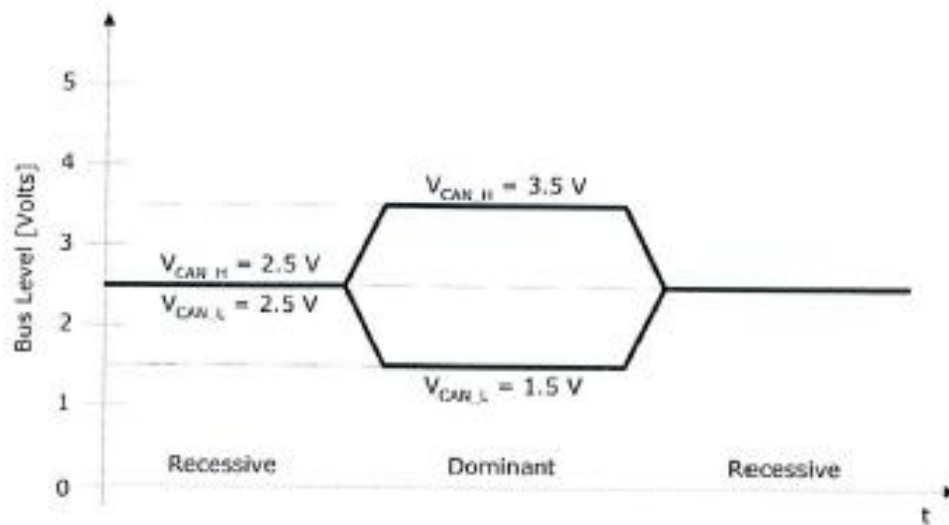
CAN on niin sanottu usean isännän (multi-master) väylä, joka tarkoittaa sitä, että kaikilla väylän laitteilla on samat oikeudet viestien lähettämässä. Jokainen laite voi lähettää tietoa väylälle aina kun väylällä ei ole muuta liikennettä. Yksittäistä väylälle lähetettävää viestiä kutsutaan viestikehykseksi, joita CAN -protokolla tukee neljää erilaista:

- datakehys = tietoa sisältävä viesti
- etäkehys = pyyntö toista laitetta lähettämään haluttua dataa väylälle
- virhekehys = ilmoitus virhetilasta
- ylikuormituskehys = ilmoitus laitteen ylikuormittumisesta. (Voss 2005, 14.)

Usean isännän väylä luo mahdollisuuden tilanteelle, jossa useampi kuin yksi laite yrittää lähettää tietoa väylälle samanaikaisesti. Tällainen tilanne on ratkaistu sillä, että viestikehyksen alkuun on lisätty tieto viestin prioriteetista. Mikäli väylän laite huomaa omaa viestitään lähettäessä, että toinen laite lähettää viestiä, joka sisältää isomman prioriteetin, keskeyttää pienemmällä prioriteetilla lähettävä laite viestin lähettämisen ja siirtyy kuuntelemaan toista viestiä. Väylän vapauduttua laite yrittää lähettää viestiä uudestaan. (Voss 2005, 14–15.)

CAN -väylälle lähetettyjä viestejä ei vahvisteta vastaanotetuksi millään tavalla, koska tämä ruuhkauttaisi turhaan väylän liikennettä. Viestit suodatetaan siten, että vain ne laitteet, joille viesti on tarkoitettu käyvät viestin sisältämän datan läpi, mutta kaikki verkon laitteet tarkastavat viestin oikeellisuuden. Mikäli jokin laitteista huomaa virheen viestissä, se ilmoittaa siitä väylälle ja alkuperäisen viestin lähettäjä lähettää väylälle virhekehysviestin. Väylän laitteet laskevat lähetettyjä ja vastaanotettuja virhekehysviestejä ja mikäli tietty laite ruuhkauttaa väylää toistuvilla virhekehysillä voi se kytkeä itsensä pois toiminnasta. (Voss 2005, 18–19.)

Tiedonsiirtoon väylällä käytetään kahta signaalitasoa, dominantti ja resessiivinen. Signaalitasot määräytyvät CAN_L ja CAN_H kaapeleiden jännitetasojen erotuksesta kuten kuvassa 3 on esitetty. Jännite-eron käyttäminen CAN_L ja CAN_H kaapeleiden välillä signaalitasojen määrittelemiseksi antaa hyvän sietokyvyn sähkömagneettisille häiriöille, sillä kumpaankin kaapeliin indusoituu sama jännite, jolloin jännite-ero pysyy samana. (Voss 2005, 136–137.)



KUVA 3. Väylän jännitetasot (Voss 2005, 135)

2.3.1 Data- ja etäkehys

Datakehystä käytetään tiedon lähettämiseen CAN -väylälle, kun taas etäkehystä käytetään datan pyytämiseen joltain väylän laitteelta. Data- ja etäkehysten rakenne on hyvin samanlainen, yksinkertaisesti etäkehys on datakehys ilman datakenttää. Kuvan 4 mukaisesti rakenteellisesti datakehykseen kuuluvat seuraavat osat:

- SOF (start of frame) = aloitusbitti, joka ilmaisee kehyksen alkua
- arbitration field = tunnistusosa, johon kuuluu viestin tunnus sekä RTR (remote transmission request) bitti, joka määrittää onko lähetettävä kehys data- vai etäkehys
- control field = käytetään määrittämään datakentän sekä viestitunnuksen pituudet.
- data field = datakenttä, johon on sisällytetty kaikki kehyksessä siirrettävä varsinaisen tieto
- CRC field = tarkastussumma
- ACK field = tarkastussumman vahvistus
- EOF (end of frame) = kehyksen loppua ilmaiseva kenttä
- IFS = viive ennen seuraavaa kehystä. (Voss 2005, 36–37.)



KUVA 4. Datakehysten rakenne (Voss 2005, 34)

Kokonaisuudessaan datakehys on pituudeltaan 47 – 111 bittiä, riippuen datakentän pituudesta, joka voi olla 0 – 8 tavua pitkä. Suurimmalla 1 Mbit/s tiedonsiirtonopeudella tämä tarkoittaa sitä, että yhden datakehysten lähettämiseen aikaa kuluu enintään 111 µs. Tunnistusosassa resessiivinen RTR bitti ilmoittaa, että lähetettävä kehys on etäkehys, jolloin datakenttää ei lähetetä. (Voss 2005, 44–45.)

2.3.2 Virhekehys

Virhekehysten avulla varmistetaan, että virheellisen data- tai etäkehysten tietoa ei käsitellä, tämä suoritetaan aiheuttamalla CAN -standardin vastainen toimenpide tiedonsiirrossa. CAN -standardi sallii vain viiden peräkkäisen samassa tilassa olevan bitin lähettämisen kehyksen alun ja CRC -kentän välissä. Virhekehyksessä virheilmaisimena (error flag) lähetetään vähintään kuusi peräkkäistä dominanttia bittiä, jolloin jokainen väylän laite huomaa rikkeen. (Voss 2005, 57.)

Virheilmaisimen lisäksi sen jälkeen virhekehyksessä lähetetään virhe-erotin (error delimiter), joka on kahdeksan resessiivisen bitin jono. Virheestä ilmoittanut laite lähettää ensimmäisenä resessiivisen bitin väylälle ja jatkaa lähettämistä niin kauan kunnes väylä muuttuu resessiiviseksi. Väylä muuttuu resessiiviseksi vasta siinä vaiheessa kun viimeisenkin laite on saanut lähetettyä virheilmaisimen väylälle ja alkaa lähettämään virhe-erotinta. Kyseistä toimintoa käytetään, jotta väylän laitteet saadaan synkronoitua samaan tilaan ja uutta kehystä päästään lähettämään väylälle. (Voss 2005, 44–45.)

2.3.3 Ylikuormituskehys

Ylikuormituskehys on hyvin samanlainen kuin virhekehys, mutta toisin kuin virhekehyyksessä ylikuormituskehys ei aiheuta edellisen kehyksen uudelleenlähtämistä. Virhekehyyksen tapaan ylikuormituskehys sisältää kuuden bitin mittaisen virheilmaisimen sekä kahdeksan bitin mittaisen virhe-erottimen. (Voss 2005, 66.)

Väylän laite voi lähettää ylikuormituskehyyksen data- tai etäkehyyksen välissä viivyttyäkseen seuraavaa kehystä. Ylikuormituskehys alkaa aina IFS -kentän ensimmäisen bitin kohdalla. Enimmillään kaksi ylikuormituskehystä voidaan lähettää peräkkäin seuraavan data- tai etäkehyyksen viivästyttämiseksi. (Voss 2005, 67.)

2.4 Ylemmän tason protokollat

CAN on erittäin tehokas ja nopea väyläteknikka ajoneuvoissa ja muissa pienissä sovelluksissa, mutta pelkästään se ei sovellu koneautomaation monimutkaisiin sovelluksiin. Tästä syystä on kehitetty ylemmän tason protokollia kuten J1939 työkoneiden käyttöön, CANopen teollisuuden koneiden ohjaamiseen sekä DeviceNet tehdasautomaation käyttöön. Tavallinen CAN -standardin viesti tukee maksimissaan kahdeksan tavun pituisia datapaketteja kun taas ylemmän tason protokollat tukevat äärettömän pitkiä. Toinen merkittävä ominaisuus ylemmän tason standardeissa on niiden kyky konfiguroida laitteet isäntä/orja (master/slave) periaatteella. (Voss 2008, 14.)

2.4.1 CANopen

CANopen on CAN in automation yhteisön kehittämä ylemmän tason protokolla. Alun perin CANopen on kehitetty liikeohjausta vaativien konesovellusten käyttöön, tänä päivänä sitä käytetään laajasti työkoneiden, lääketieteen sekä meriteollisuuden sovellusten ohjaamiseen. Protokollan tarkoituksena on tarjota standardoituja CANopen laitteita, joita ei tarvitse konfiguroida erikseen ja tällä tavoin helpottaa suunnittelijan ja käyttäjän työtä. (CAN in automation 2016.)

2.4.2 SAE J1939

J1939 on SAE:n (society of automotive engineers) kehittämä CAN -standardiin pohjautuva ylemmän tason protokolla, joka on kehitetty erityisesti teollisuuden työkoneiden sekä metsäkoneiden käyttöön (Voss 2008, 1). Tavallisesta CAN -standardista poiketen J1939 käyttää 11 bitin tunnistekentän sijaan laajennettua 29 bitin tunnistekenttää, joka sallii 2048 erilaisen tunnisteen sijaan 2^{29} erilaista tunnistetta. Yksinkertaiseen CAN -väylään riittää hyvin tavallisen tunnistekentän ominaisuudet, mutta monimutkaisten nykyaikaisten laitteiden kuten metsäkoneen kohdalla tarvitaan laajennettua tunnistekenttää. (Voss 2008, 5.)

3 CODESYS

CODESYS on saksalaisen 3S-Smart Software Solutions GmbH:n kehittämä ohjelmisto teollisuuden automaatiojärjestelmien suunnitteluun. Ohjelmointi CODESYS ohjelmistolla pohjautuu vahvasti IEC 61131-3 standardiin ja se tukee kaikkia viittä standardissa määriteltyä ohjelmointikieltä. Ohjelmistossa on myös kattava visualisointityökalu käyttöliittymän suunnitteluun. (CODESYS 2016.)

Ohjelmistosta on saatavilla kaksi versiota, V3 ja V2. Uudet päivitykset ja lisäosat tuodaan ainoastaan uudempaan V3, mutta monet PLC -valmistajat käyttävät vielä vanhempaa versiota laitteidensa ohjelmointiin. Vanhemmalle versiolle suunniteltuja laitteita ei voida ohjelmoida uudemmalla ohjelmistoversiolla ja tästä syystä vanhempaa versiota ylläpidetään ja korjataan valmistajan toimesta vuoden 2019 loppuun saakka. (CODESYS 2016.)

3.1 IEC 61131-3

IEC 61131 on kansainvälinen standardi, johon on kerätty yhteen PLC -järjestelmien vaatimukset. Standardi sisältää laitteiden vaatimukset fyysisen rakenteen sekä ohjelmoimisen kannalta. (John & Tiegelkamp 2010, 12.)

IEC 61131-3 osio standardista antaa suosituksia PLC -laitteistojen ohjelmoimisesta. Standardi määrittelee valtavan määrän ohjelmointiin liittyviä yksityiskohtia, joten ei voida olettaa, että järjestelmät toteuttavat täysin standardin vaatimukset. PLC -valmistajien on dokumentoitava sekä pystyttävä esittämään, miltä osin heidän ohjelmistonsa täyttävät tämän standardin vaatimukset. Tätä varten standardi sisältää toiminnallisuustaulukoita, jotka täyttämällä pystytään todentamaan kuinka hyvin luotu ohjelmisto täyttää standardin vaatimukset. (John & Tiegelkamp 2010, 12.)

Käyttäjät työskentelevät usein samanaikaisesti eri laitevalmistajien PLC -järjestelmien parissa. Usein tämä on merkinnyt sitä, että käyttäjän tulee opiskella jokaisen ohjelmiston käyttö alusta lähtien. IEC 61131-3 -yhteensopivien ohjelmistojen perus ohjelmointiperi-

aatteet ovat aina samat, joten käyttäjä tarvitsee koulutuksen vain valmistajan erityissovel-
luksiin. (John & Tiegelkamp 2010, 134.) Seuraavissa kappaleissa on käsitelty yleisiä stan-
dardin määrittelemiä ohjelmiston rakenteita sekä ohjelmointikieliä.

3.1.1 Program organisation unit (POU)

IEC 61131-3 standardi kutsuu lohkoja, joista ohjelmat ja projektit koostuvat program or-
ganisation unit:ksi. POU voi olla tyypiltään ohjelma, toiminto, tai toimintolohko. Nämä
kolme lohkotyyppiä eroavat toisistaan seuraavasti:

- toiminto = lohko, jolle voidaan asettaa parametrit, mutta sillä ei ole lohkon sisäisiä
muuttujia. Mikäli sisääntuloparametreja ei muuteta, lohko antaa aina saman ulos-
tulon
- toimintolohko = lohko, jolle voidaan asettaa parametrit sekä sisäisiä muuttujia.
Ulostulon tilaan vaikuttaa sisääntuloparametrien lisäksi lohkon sisäisten muuttu-
jien tila
- ohjelma = Lohko, joka toimii pääohjelmana. kaikki ohjelman muuttujat, joille on
asetettu fyysinen osoite, esimerkiksi PLC:n sisään- ja ulostulot on ilmoitettava
tässä lohkoissa tai sen yläpuolella. (John & Tiegelkamp 2010, 30–31.)

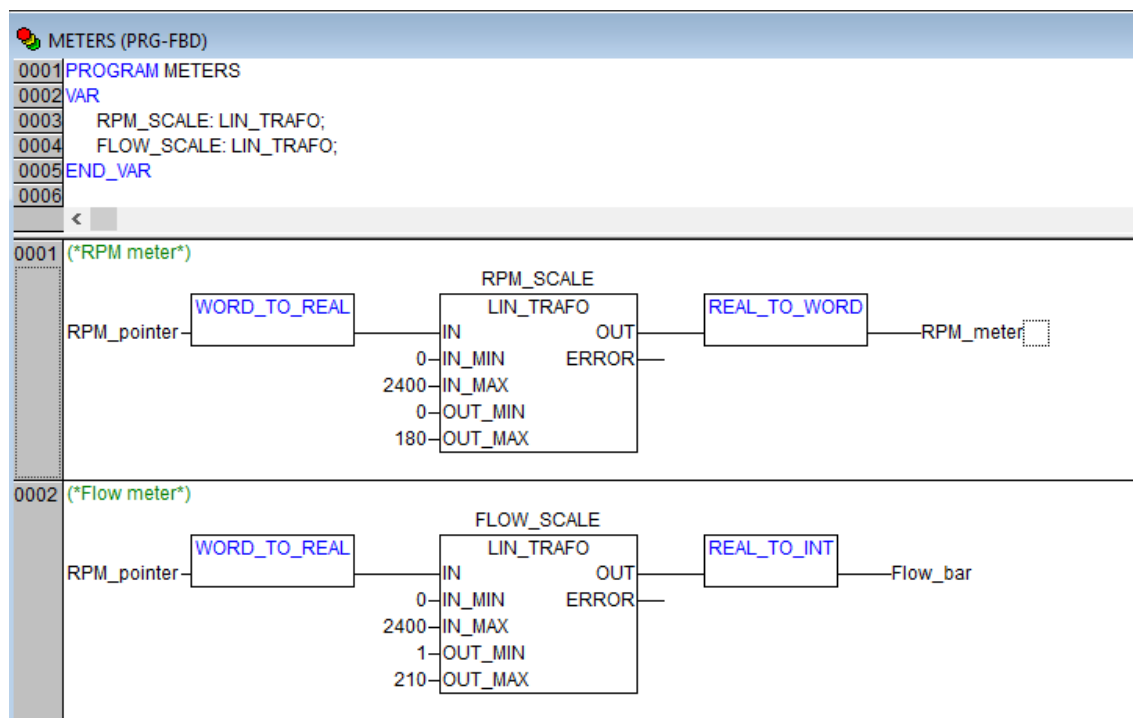
POU on yksittäinen lohko, joka voidaan kääntää itsenäisesti ottamatta huomioon ohjel-
man muita osioita. Mikäli POU:n sisällä kutsutaan muita lohkoja, niin silloin kääntämi-
seen tarvitaan pääsy kutsuttaviin lohkoihin. Yhdistelemällä lohkoja yhdeksi kokonaisu-
deksi saadaan luotua ohjelma. (John & Tiegelkamp 2010, 31.)

Rakenteellisesti POU koostuu nimiosiosta, ilmoitusosiosta sekä koodiosiosta. Ni-
miosiosta selviää lohkon tyyppi sekä sen nimi. Ilmoitusosiossa määritellään kaikki loh-
kossa käytettävät muuttujat sekä ovatko ne käytössä vain lohkon sisällä vai koko ohjel-
massa. Koodiosiossa on esitetty itse ohjelmakoodi valitulla ohjelmointikielellä. (John &
Tiegelkamp 2010, 32.)

3.1.2 Function block diagram (FBD)

Function block diagram eli toimintalohkokaavio on graafinen ohjelmointikieli. Toimintalohkokaavio ohjelmointikielenä on lähtöisin signaalinkäsittelyn alalta, missä kokonaislukujen sekä liukulukujen käsittely on tärkeää. Nykyään toimintalohkokaavio on hyvin laajasti käytetty ohjelmointikieli teollisuuden logiikkaohjaimissa. (John & Tiegelkamp 2010, 134.)

Toimintalohkokaaviossa ohjelman koodiosio on jaettu verkkoihin ohjelman kulun ja luettavuuden helpottamiseksi. Jokainen verkko on numeroitu yksilöllisesti ja mikäli väliin lisätään uusi verkko ohjelmistot osaavat päivittää muiden verkkojen tunnuksen. Tunnukset helpottavat oikean verkon löytämistä esimerkiksi virhetilanteissa. (John & Tiegelkamp 2010, 134-135) Kuvassa 5 on esitetty yksinkertainen ohjelmakoodin pätkä toimintalohkokaavion avulla.

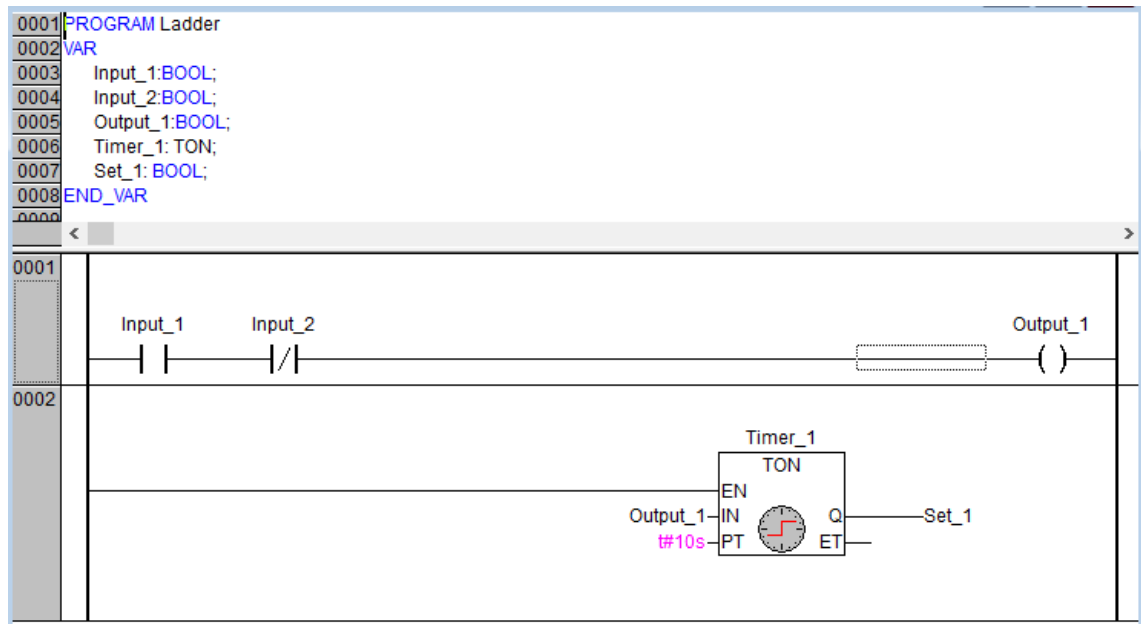


KUVA 5. Toimintalohkokaavio

Ohjelmakoodi luodaan verkkoihin graafisten toimintalohkojen avulla, jotka yhdistetään toisiinsa viivojen avulla. Kuten kuvasta 5 huomataan, niin lohkojen tuloihin voidaan tuoda muuttujia, vakioita tai vaihtoehtoisesti jättää tyhjäksi.

3.1.3 Ladder diagram (LD)

Ladder diagram eli tikapuukaavio on lähtöisin sähkötekniikan relekytkennöistä ja siinä kuvataan signaalin kulkua POU:n verkon läpi vasemmalta oikealle. Ohjelmointikieli on pääsääntöisesti suunniteltu boolean tyyppisille tosi / epätosi -signaaleille. (John & Tiegelkamp 2010, 147.)



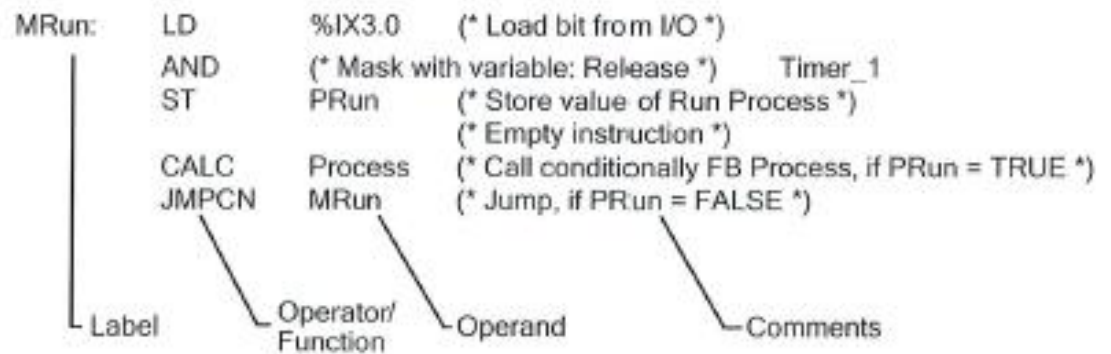
KUVA 6. Tikapuukaavio

Tikapuukaaviossa ohjelman verkoissa kulkee niin sanottu virtakaapeli vasemmalta oikealle. Signaalin kulkua ohjataan koskettimilla ja niiden loogista tilaa vaihtamalla. (John & Tiegelkamp 2010, 147.) Kuvassa 6 on esitetty yksinkertainen ohjelma tikapuukaavion avulla. Input_1 ollessa loogisessa tilassa 1 ja Input_2 ollessa loogisessa tilassa 0 signaali pääsee kulkemaan ulostuloon Output_1 ja asettamaan sen arvoon 1. Output_1 aktivoituessa käynnistyy verkossa 2 sijaitseva ajastin ja 10 sekunnin kuluttua ajastin asettaa muuttujan Set_1 arvoon 1.

3.1.4 Instruction list (IL)

Instruction list eli komentolista on nimensä mukaan lista tekstipohjaisia komentoja, joista muodostuu suoritettava ohjelmaosio. Komentolistassa yksi komento esitetään yhden rivin

pituisena ja se muodostuu otsikosta, toiminnosta, muuttujasta sekä kommentista. Kommentit sekä otsikko eivät ole pakollisia, mutta helpottavat ohjelman lukua. Otsikoiden avulla mahdollistetaan, myös hyppiminen ohjelman eri kohtiin sitä suorittaessa. Toimintokentässä ilmaistaan toiminto, joka halutaan suorittaa muuttujalle. (John & Tiegelkamp 2010, 100–101.) Kuvassa 7 on esitetty esimerkkikomentoja komentolistalla.



KUVA 7. IL malli (John & Tiegelkamp 2010, 101)

3.1.5 Structured text (ST)

Structured text eli strukturoitu teksti on komentolistan tapaan tekstipohjainen ohjelmointikieli, jota voidaan verrata C tai PASCAL ohjelmointikielen. ST ohjelma koostuu lausekkeista, joiden avulla lasketaan ja asetetaan arvoja muuttujille sekä kontrolloidaan komentojen kulkua. (John & Tiegelkamp 2010, 116.) Taulukossa 1 on esitetty yleisimpiä komentoja structured text ohjelmointikielillä.

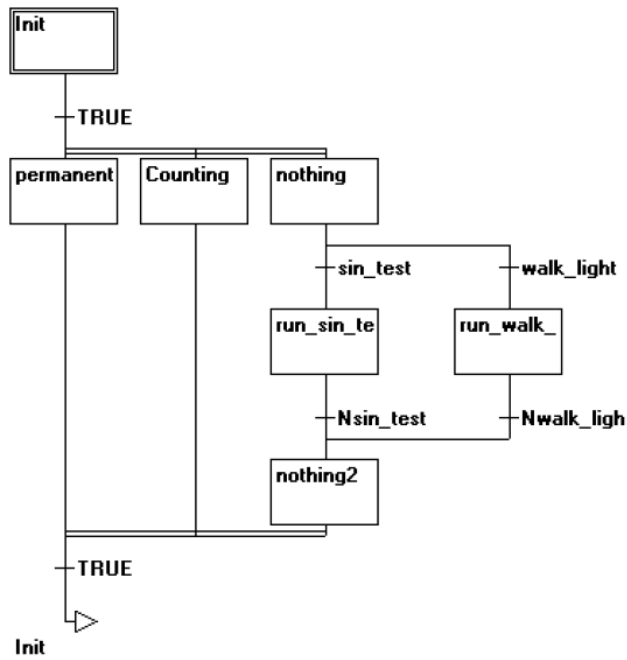
TAULUKKO 1. Structured text komentoja

Komento	Esimerkki	Selitys
:=	a:=9;	Muuttuja a asetetaan arvoon 9
IF	IF a= 9 THEN b:= 8; END_IF;	Mikäli a saa arvon 9 asetetaan b arvoon 8
CASE	CASE a OF 1: c:= 2; 2: c:= 3; 3: c:= 4; ELSE g:= 0; END_CASE;	Muuttujan c tila vaihtuu muuttuja a:n tilan mukaan
WHILE	WHILE a > 5 d:= FALSE; END_WHILE;	Kun a on suurempi kuin 5 muuttuja d on epätosi

ST ohjelmointikielessä lausekkeet erotetaan toisistaan puolipisteellä. Toisin kuin komentolistassa komennot voivat jatkua usealle riville tai useampi lauseke voidaan kirjoittaa samalle riville. (John & Tiegelkamp 2010, 116.) Tekstipohjaiset ohjelmointikielet kuten ST ovat hyvin tarkkoja koodin asettelusta ja merkintöjen oikeellisuudesta. Virhetilanteissa pitkistä ohjelmista voi olla hyvinkin haastavaa löytää virheitä, tätä seikkaa helpottaakseen Codesys -ohjelmointiympäristöön on kehitetty tarkastusmahdollisuus, joka ilmoittaa koodissa olevat virheet rivin tarkkuudella.

3.1.6 Sequential function charts (SFC)

Sequential function chart eli vuokaavio-ohjelmointi on kehitetty monimutkaisten ohjelmien kulunhallintaan. Sen avulla voidaan ohjelma jakaa pienempiin osiin ja hallita suoritusmenetelmien kulkua. SFC:n avulla on mahdollista suunnitella rinnakkaisia ja peräkkäisiä prosesseja. (John & Tiegelkamp 2010, 169.) Kuvassa 8 on esimerkki SFC ohjelmasta.



KUVA 8. Sequential function chart (User manual for... 2007, 28)

SFC on pääsääntöisesti graafinen ohjelmointikieli vaikka myös tekstipohjainen rakenne on määritetty IEC 61131-3 standardissa. Ohjelmaa tehdessä ohjelmalohkot on kuitenkin ohjelmitava jollakin neljästä aiemmissä kappaleissa esitetystä ohjelmointikielestä. SFC sopii parhaiten prosessiin, jossa edetään askelittain osiosta toiseen, kuten esimerkiksi pesukoneen pesuohjelma. (John & Tiegelkamp 2010, 147.)

4 SUUNNITTELU

4.1 Suunnittelun lähtökohta

Suunniteltavan kompressorisyksikön pääasiallisena käyttötarkoituksena on tuottaa paineilmaa maalämpökaivojen porauksessa käytettävälle porayksikölle. Käyttötilanteessa kompressorisyksikkö sijaitsee etäällä porayksiköstä, jolloin poraamista valvova henkilö ei ole jatkuvasti kompressorin välittömässä läheisyydessä vaan hoitaa porausta porayksikön luota. Tämän takia laitteen ohjausjärjestelmä tulee suunnitella siten, että sen suoritusarvoja ei jouduta muuttamaan kesken poraustapahtuman, vaan ohjausjärjestelmä valvoo laitteen toimintaa ja ilmoittaa varoitusvalojen avulla mahdollisista vioista. Vakavien viikkojen osalta ohjausjärjestelmä tulee suunnitella siten, että laite sammuttaa itsensä vian ilmetessä.

Projektille oli määritelty ennen tämän työn aloittamista toiminnallisia vaatimuksia sekä käytettäviä komponentteja asiakkaiden toiveiden ja käyttökokemusten pohjalta. Ohjausjärjestelmän kannalta tärkein ennalta määritelty asia oli logiikkatoimittajaksi valikoitunut IFM electronics gmbh. Muita tämän työn kannalta oleellisia ennalta määriteltyjä komponentteja olivat GHH RAND:n valmistama ruuvikompressori sekä sille käyttövoiman antava Scaniaan dieselmoottori. Toiminnallisia vaatimuksia laitteelle oli

- helppokäyttöisyys
- helppo ja nopea huollettavuus
- edullisuus kilpailijoihin nähden
- luotettava toiminta
- varaosien hyvä saatavuus.

4.1.1 IFM electronics

IFM electronics on 1969 perustettu saksalainen automaatioteknologian valmistaja, jonka laaja tuotevalikoima kattaa standardiratkaisujen lisäksi myös monien yksittäisten teollisuudenalojen erikoisvaatimukset. Tunnettuja tuotemerkkejä ovat ”efector” tunnistus-, diagnostiikka ja identifiointijärjestelmille sekä ”ecomat” väylä- ja ohjausjärjestelmille. (IFM electronics 2016.)

Ohjausjärjestelmässä käytettäviksi laitteiksi oli valikoitunut ecomatmobile basic sarjan pienlogiikka, joka on liikkuvan kaluston käyttöön suunniteltu moduulirakenteinen, helpokäyttöinen ja edullinen ohjausjärjestelmä. Käyttöliittymänä käytettiin CR0452 -näytölaitetta, joka on varustettu 4,3” värinäytöllä, kuudella vapaasti ohjelmoitavalla funktiopainikkeella sekä keinukytkimellä kuten kuvassa 9 on esitetty. Neljä näytön alareunassa olevaa funktiopainiketta on nimetty vasemmalta oikealle F1–F4 ja kaksi pienempää painiketta on esivalinnaltaan OK ja ESC painikkeet. Näytön suojausluokka on IP67, joten sitä voidaan käyttää olosuhteissa, joissa se joutuu altistumaan vedelle ja lialle. (IFM electronics 2016.)



KUVA 9. CR0452 (IFM electronics 2016)

PLC:ksi valittiin näytön kanssa samaa sarjaa edustava kuvassa 10 esitetty CR0403 Basic Controller, jossa on 12 tuloa, 12 lähtöä sekä kaksi CAN-liitäntää. Kummatkin CAN-liitännät tukevat sekä CANopen, että J1939 standardia. Tulot voidaan konfiguroida tarpeen mukaan käsittelemään digitali-, virta-, jännite-, resistanssi- tai taajuussignaalia.

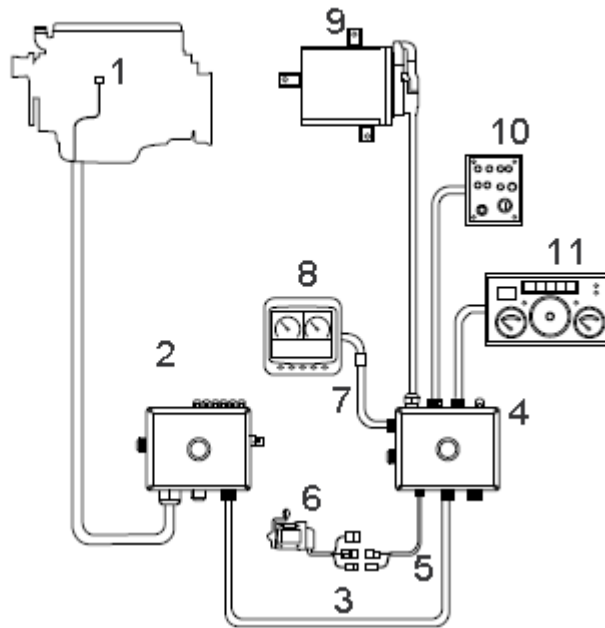


KUVA 10. CR0403 (IFM electronics 2016)

4.1.2 Scania DC16

DC16 kuuluu työn tekohetkellä Scanian uusimpaan Stage IV -vaatimukset täyttävään teollisuusmoottorisarjaan, jotka ovat kehitetty kuorma-autojen ja linja-autojen Scania Euro 6 -sarjan moottoreiden pohjalta (Scania-teollisuusmoottorit 2016). Scania oli valikoitunut moottoritoimittajaksi tätä projektia varten sen laajan huoltoverkoston sekä hyvien käyttäjäkokemusten perusteella.

Sähköjärjestelmän osalta Scaniaalta on saatavilla moottoriin oma kuvan 11 mukainen perusjärjestelmä, jonka avulla moottoria pystytään käyttämään ja valvomaan ilman muita ulkoisia laitteita.



Teollisuusmoottorien perusjärjestelmä ja liitetyt Scania-yksiköt

1. Moottorin ohjausyksikkö
2. Pääkytkentärasia
3. Liitäntäjohto: 2, 8, 12 tai 24 m
4. Koordinaattorin kytkentärasia
5. Liitäntäkaapeli: 2 tai 8 m
6. Kaasupoljinanturi
7. Liitäntäkaapeli: 2 m
8. Scanian digitaalinäyttö
9. Koordinaattori
10. Ohjauspaneeli
11. Analoginen kojetaulu

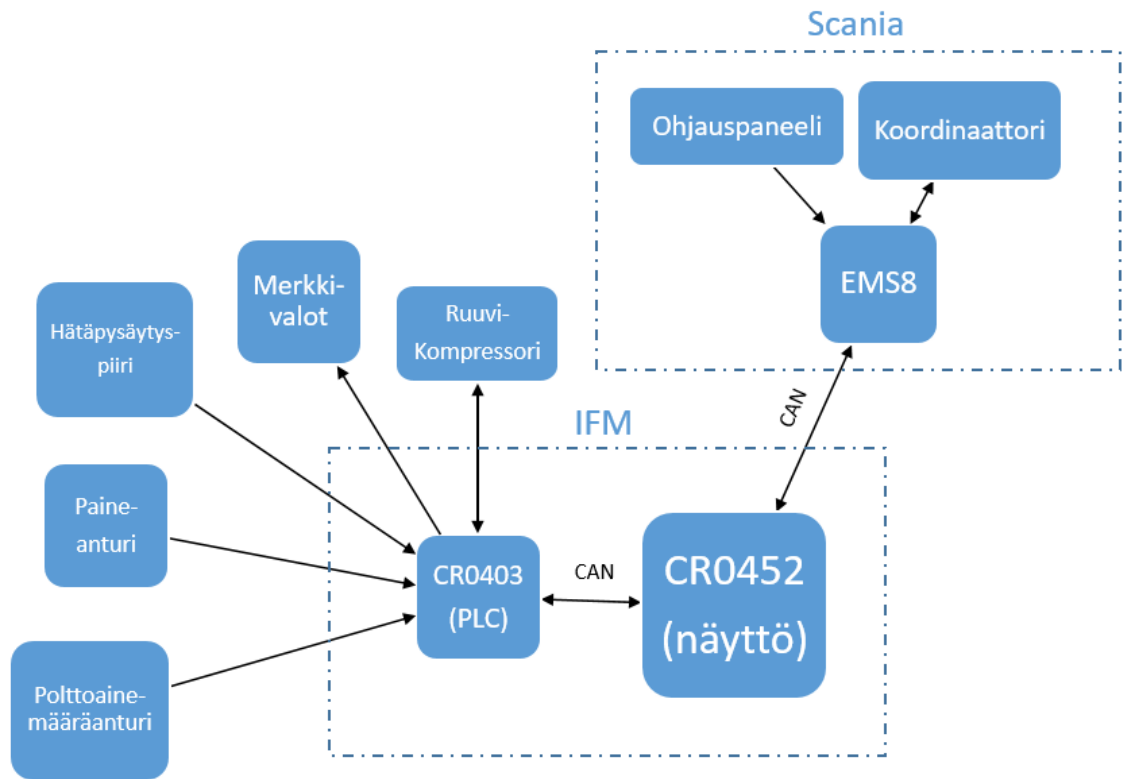
KUVA 11. Scania teollisuusmoottorin perusjärjestelmä

Vaihtoehtoisesti asiakas voi suunnitella ja toteuttaa itse koko ohjausjärjestelmän moottorille, jolloin Scania toimittaa ainoastaan asiakasliittymän liittimen moottorin ohjausyksikölle. Tähän projektiin valittiin osittainen Scanian perusjärjestelmä, johon kuuluu pääkytkentärasia, koordinaattorin kytkentärasia, koordinaattori, liitäntäjohdot sekä ohjauspaneeli.

4.2 Ohjausjärjestelmän suunnittelu

Ohjausjärjestelmän suunnittelu aloitettiin määrittelemällä komponentit ja laitteet, joiden ympärille ohjausjärjestelmää lähdetäisiin rakentamaan. Kuviossa 1 on havainnollistettu

järjestelmään liitettäviä laitteita sekä niiden keskinäistä kommunikointia. Pääosin tiedonsiirto ohjausjärjestelmässä tapahtuu näytön CR0452 ja Scanian EMS8 ohjausyksikön välillä.



KUVIO 1. Tiedonsiirto järjestelmässä

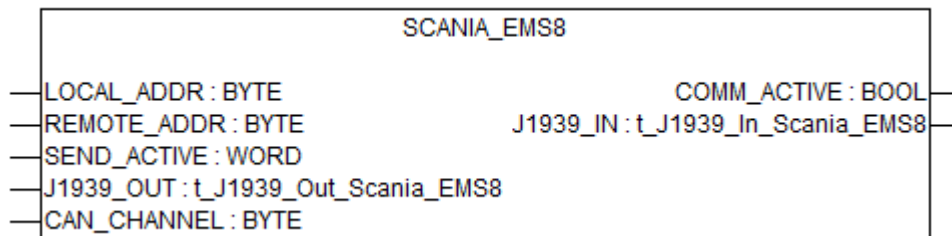
CAN -väylällä keskustelevia kuviossa 1 esiintyvistä lohkoista on ainoastaan PLC, näyttö sekä moottorin ohjausyksikkö EMS8. Muut PLC:n kanssa keskustelevat komponentit liitetään PLC:n sisään- ja ulostuloihin ja konfiguroidaan niiden käyttämän signaalityyppin mukaisesti.

IFM:n näytön ja PLC:n pohjalla toimivan ohjelman suunnittelussa käytettiin Codesys -ohjelmistoa, jonka pohjalle IFM on tehnyt jokaiselle laitteelle konfiguraatiot sekä mallin ohjelmakoodista. Näytön ohjelman suunnitteluun käytettiin IFM:n mallia, johon oli tehty valmiit ohjelmalohkot sivujen vaihdolle sekä konfiguroinnit näytön painikkeille.

4.2.1 Scania EMS8

EMS8 on käytetyn Scanian dieselmoottorin ohjausyksikkö, jonka kautta moottorin toimintoja ohjataan. Ohjausjärjestelmän liittäminen ohjausyksikköön tapahtuu CAN -väylän avulla moottorin kyljessä sijaitsevan asiakasliittynnän avulla. Scania toimittaa moottorin asennusoppaiden mukana tarkat selostukset jokaisesta CAN -viestistä, joka ohjausyksikköön voidaan lähettää tai sieltä vastaanottaa.

Ohjausjärjestelmän suunnittelua helpotti erittäin paljon se, että IFM on luonut valmiin toimintalohkon Codesys -ohjelmointiympäristöön EMS8 ohjausyksikölle. Kuvassa 12 esitetyn toimintalohkon avulla saadaan luettua ja kirjoitettua tietoa ohjausyksiköstä.



KUVA 12. EMS8 toimintalohko

4.2.2 Häätäpysäytys

Konedirektiivissä määritetään, että jokaisessa koneessa on oltava vähintään yksi hätäpysäytyslaite (Konedirektiivi 2016). Suunnittelun lähtökohta hätäpysäytyslaitteiden osalta oli, että laitteen kylkeen käyttöpaneelin läheisyyteen sijoitetaan yksi hätäpysäytyspainike sekä selvitetään mahdollisuutta langattoman hätäpysäytyskauko-ohjaimen toteuttamisesta. Kauko-ohjain koettiin hyödylliseksi, sillä konetta käytettäessä käyttäjä ei ole jatkuvasti koneen välittömässä läheisyydessä, jolloin vikatilanteessa kauko-ohjaimen käyttö vähentäisi riskiä vahingon sattumiselle. Toinen syy kauko-ohjaimen käytölle oli mahdollinen paineilmaletkun rikkoutuminen, jolloin letku pääsisi piiskamaisesti heilumaan, estäen pääsyn koneelle ja hätäpysäytyspainikkeen painamisen.

Standardissa SFS-EN ISO 13850 käsitellään hätäpysäytyksen ominaisuuksia ja vaatimuksia suunnittelun näkökulmasta. Standardin mukaan hätäpysäytyksen tarkoituksena on estää vaaratilanteen syntyminen, joka johtuu ihmisten tai koneen odottamattomasta

vaarallisesta toiminnasta. Häätäpysäytystoiminto on suunniteltava siten, että se menee prioriteetiltaan kaikkien ohjaustoimenpiteiden edelle, kuitenkin vaikuttamatta muiden suojaustoimintojen toimintaan. Toiminto ei myöskään saa aiheuttaa muita vaaroja. (SFS-EN ISO 13850, 7–8.)

Tämän työn osalta ongelmaksi hätäpysäytyksessä osoittautui muiden vaarojen estäminen hätäpysäytystä painettaessa, sillä moottorin äkkinäinen pysäyttäminen normaalista toimintanopeudesta aiheuttaa kompressorille virhetilanteen, jolloin on mahdollista, että kompressori ”sylkäisee” voiteluun käytettävät öljyt imukanavan kautta ulos kompressorista. Tilanteen estämiseksi hätäpysäytystä painettaessa moottori on pysäytettävä hallitusti kierroslukua laskemalla ennen pysäytyskäskeyä.

4.3 Käyttöliittymän suunnittelu

Laitteen käyttöliittymänä toimii osiossa 4.1.1 esitelty IFM electronics:n CR0452 näyttöyksikkö sekä Scanian oma käyttöpaneeli. Käyttöliittymän suunnittelussa ensisijaisiksi ominaisuuksiksi muodostui helppokäyttöisyys ja selkeys. Laitetta tullaan käyttämään ulkona vaihtelevissa olosuhteissa, jolloin käyttöpaneelia täytyy pystyä käyttämään myös hanskat kädessä. Tästä syystä näyttöyksiköksi valikoitui perinteinen fyysisillä painikkeilla varustettu näyttö nykyaikaisten kosketusnäyttöjen sijasta.

Scanian käyttöpaneeli, joka on esitetty kuvassa 11 numerolla 10, on osa Scanian perusjärjestelmää ja sitä käytetään moottorin käynnistämiseen, sammuttamiseen sekä kierrosluvun säätämiseen. Paneeli sisältää:

- käynnistyskytkimen avaimella
- sammutuspainikkeen
- kaksi ohjelmoitavaa kytkintä kierrosluvun esiasetuksille
- kytkimen kierrosluvun portaattomalle säädölle.

Käyttöliittymän tehtävänä on esittää käyttäjälle moottorin ja kompressorin oleelliset suoritusarvot sekä varoittaa mahdollisista vioista. Näiden ominaisuuksien lisäksi käyttöliittymän kautta tulisi pystyä säätämään moottorin kierroslukua halutulle tasolle. Käyttöliittymän suunnitteluun liittyvissä asioissa kuten visuaaliseen merkinantoon on annettu sään-

nöksiä standardissa SFS-EN 60073. Standardin tavoitteena on helpottaa prosessin ohjausta ja valvontaa sekä tämän kautta parantaa henkilöiden, omaisuuden ja ympäröivän alueen turvallisuutta (SFS-EN 60073, 12).






Näyttöyksikön käyttöliittymän suunnittelussa tulee huomioida visuaalisten elementtien ja erityisesti varoitusten ja huomautusten osalta oikeiden värien ja muotojen käyttö, jotta niiden merkitys on helposti ja nopeasti ymmärrettävissä. Standardissa SFS-EN 60073 on esitetty taulukon 2 mukaiset yleisperiaatteet värien valinnassa kun niitä käytetään informaation esittämiseen. Käytettävien värien on erotuttava toisistaan ja taustasta kontrastieron avulla, erityisesti turvallisuuteen liittyvissä merkityksissä värien on oltava kirkkaita ja erottuvia (SFS-EN 60073, 24).

TAULUKKO 2. Värien merkitys (SFS-EN 60073, 22)

Värit	Merkitys		
	Henkilöiden tai ympäristön turvallisuus	Prosessin olosuhteet	Laitteen tila
PUNAINEN	Vaara	Hätätilanne	Viallinen
KELTAINEN	Varoitus/huomio	Epänormaali	Epänormaali
VIHREÄ	Turvallinen	Normaali	Normaali
SININEN	Merkitsee pakollisuutta		
VALKOINEN HARMAA MUSTA	Mitään erityistä merkitystä ei anneta		

Kuvioiden osalta muodot ovat määritelty samaan tapaan kuin värit. Näkyvissä koodeissa muoto voi toimia yksinään pääkoodina tai vaihtoehtoisesti täydentämään pääkoodia esimerkiksi estämään henkilöiden puutteellisesta värinäöstä aiheutuvia virheitä (SFS-EN 60073, 24). Muotojen osalta SFS-EN 60073 määrittää yleisperiaatteet taulukon 3 mukaisesti.

TAULUKKO 3. Muotojen merkitys (SFS-EN 60073, 26)

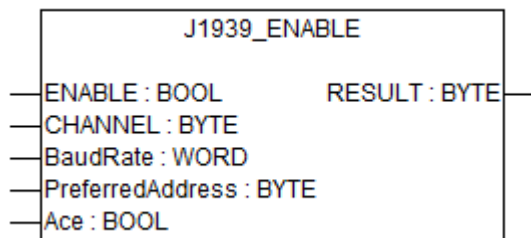
Muoto	Merkitys		
	Henkilöiden tai ympäristön turvallisuus	Prosessin olosuhteet	Laitteen tila
 ¹⁾	Vaara	Hätätilanne	Viallinen
 ¹⁾	Varoitus/Huomio	Epänormaali	Epänormaali
 ¹⁾	Turvallinen	Normaali	Normaali
	Pakollinen merkitys		
	Mitään erityistä merkitystä ei anneta		
HUOM. Laitteen tai prosessin tila olisi koodattava standardisarjoihin IEC 60417, IEC 60617 sekä ISO 7000 perustuvien merkein tai kuvatunnuksin.			
¹⁾ Ainoastaan turvallisuuteen liittyvissä sovellutuksissa on muoto kehystettävä leveällä reunuksella.			

Merkinannossa on huomioitava, että suunniteltavaa laitetta käytettäessä käyttäjä ei ole jatkuvasti laitteen välittömässä läheisyydessä. Tämä tilanne johtaa siihen, että pelkällä näytössä ilmoitettavalla merkillä ei välttämättä saada kiinnitettyä käyttäjän huomiota. Tilanteen ratkaisemiseksi laitteen katolle tai kylkeen on lisättävä oranssi sekä punainen valo vikojen ilmaisuun. Lisäksi vakavien moottori- ja kompressorivikojen, kuten alhaisen öljynpaineen sattuessa, moottorin ohjausyksikkö on ohjelmoitu siten, että moottori sammuu automaattisesti vian aktivoituessa.

5 TOTEUTUS

5.1 Ohjausjärjestelmä

Ohjausjärjestelmän toteutus keskittyi moottorin ohjausyksikön viestien sekä PLC:n tulo- ja lähtötietojen käsittelyyn. Jotta moottorin ohjausyksikön tietoja voidaan käsitellä, täytyi ohjausjärjestelmän laitteet asettaa keskustelemaan ohjausyksikön tukemalla J1939 protokollalla. Protokollan asettaminen tapahtuu IFM:n luomalla valmiilla toimintalohkolla joka on esitetty kuvassa 13.



KUVA 13. Toimintalohko J1939 asettamiseksi

Oikean CAN -protokollan asettamisen jälkeen moottorin ohjausyksikön tietoja voitiin lukea ja sinne voitiin kirjoittaa sanomia kuvassa 12 esitetyn toimintalohkon avulla. Tässä työssä suurin osa käsiteltävistä viesteistä on moottorin ohjausyksikön tulotietoja suoritusarvoista sekä vikailmaisuuksista. Tulotiedot käsiteltiin siten, että niillä voidaan ohjata käyttöliittymän haluttuja elementtejä.

Moottorin ohjausyksikköön kirjoitettavista viesteistä hyvänä esimerkkinä on osiossa 4.2.2 esitetty ongelma liian nopeasta hätäpysäytyksestä, joka on ratkaistu siten, että hätäpysäytyspainikkeen aktivoimisesta ohjausjärjestelmä tiputtaa ensin hallitusti kierrosluvun tyhjäkäynnille, jonka jälkeen annetaan pysäytyskäsky moottorille. Standardi SFS-EN ISO 13850 antaa mahdollisuuden hallitulle pysäytysnopeudelle, mikäli muutoin aiheutuu muita riskejä koneen käyttäjälle (SFS-EN ISO 13850, 8).

PLC:n lähtöjä ja tuloja on tässä revisioissa käytössä yhteensä kaksitoista, joista kaksi on lähtöjä ja kymmenen tuloja. Tuloja käyttävät paineanturi, polttoainemääräanturi, hätäpysäytyspiiri sekä kompressorin anturitiedot. Lähtöjä käyttää punainen ja oranssi varoi-

tusvalo. Tulot ja lähdöt ovat asetettu tukemaan anturin tai valon signaalityyppiä, sille tarkoitetun toimintalohkona avulla. Polttoainemääräanturin tyyppiä ei ollut määritelty työn tekohetkellä johtuen siitä, että polttoainetankin suunnittelua ei ollut saatu vielä päätökseen. Anturille on kuitenkin määritelty oma tulo, johon se tullaan kytkemään ja skaalamaan signaalin tyyppin mukaan mittarille sopivaksi.

Kompressorilta on saatavilla anturitietoja paineista sekä lämpötiloista ilman ja voiteluöljyn osalta. Anturitietoja käytetään kompressorin hallittuun pysäyttämiseen raja-arvojen ylittyessä. Raja-arvon ylittyessä viat ilmaistaan moottorin vikakoodien tapaan käyttöliittymässä.

5.2 Käyttöliittymä

Käyttöliittymän toteutus keskittyi vahvasti IFM:n näytön ohjelmointiin ja visuaalisten elementtien asettamiseen. Scanian ohjauspaneelin konfigurointi rajoittuu kahden kytkimen kierrosluvun esiasetuksen asettamiseen. Kytkimet konfiguroidaan siten, että toinen kytkin asettaa moottorin tyhjäkäynnille ja toinen asettaa kierrosluvun ruuvikompressorin maksimipaineen asetukselle, joka on noin 1850 RPM.

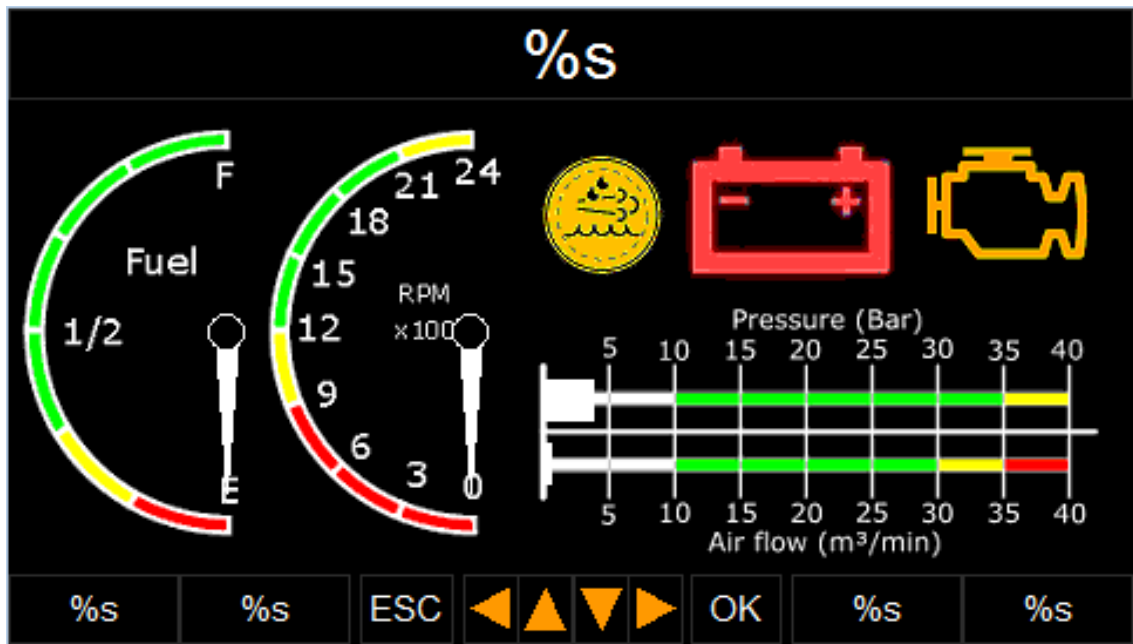
CR0452 näyttöyksikkö ohjelmoitiin siten, että käyttäjällä on valittavissa kolme eri sivua käyttötilanteen mukaan. Pääsivulla näkyy perusnäkyvä, jossa on esitetty laitteen yleiset suoritusarvot (kuva 14). Käyttäjälle on näkyvillä seuraavat elementit:

- polttoainemäärä
- moottorin kierrosluku
- Ilmanpaine
- ilman virtausmäärä.



KUVA 14. Näytön perusnäkyvä

Kuvassa 14 näytön alareunan tekstit ”MOTOR” ja ”DIAG” kuvaavat näyttöyksikössä olevien painikkeiden toimintoja ja tässä kyseisessä tilanteessa painikkeiden avulla voidaan siirtyä sivulta toiselle. Kuvassa esitetään tilannetta, jossa ohjausjärjestelmä on kytketty päälle ja yhtäkään vikaa ei ole aktiivisena. Kuvassa 15 on esitetty kaikki perusnäkyvän elementit. Normaalisessa käyttötilanteessa piilossa olevia vikailmoituksia näyttöön on asetettu alhaisen ureamäärän ilmaisu, generaattorin virheen ilmaisu sekä moottorin vikavalvonta, joka syttyy, mikäli moottorilta tulee yksi tai useampi aktiivinen vikakoodi. Alhaisen ureamäärän ilmaisu on koodattu siten, että varoitus on päällä jatkuvasti, kun urean taso on alle 20 %, vilkkuu hitaasti tason ollessa alle 10 % ja vilkkuu nopeasti säiliön ollessa tyhjä. Kyseinen koodaustapa vaaditaan USA:n markkinoille myytävälle tier 4 hyväksynnän moottoreille, mutta vaikka tässä työssä käytettävällä moottorilla ei ole tier 4 hyväksyntää, koettiin kyseinen koodaustapa hyväksi ureatason ilmaisemiselle.

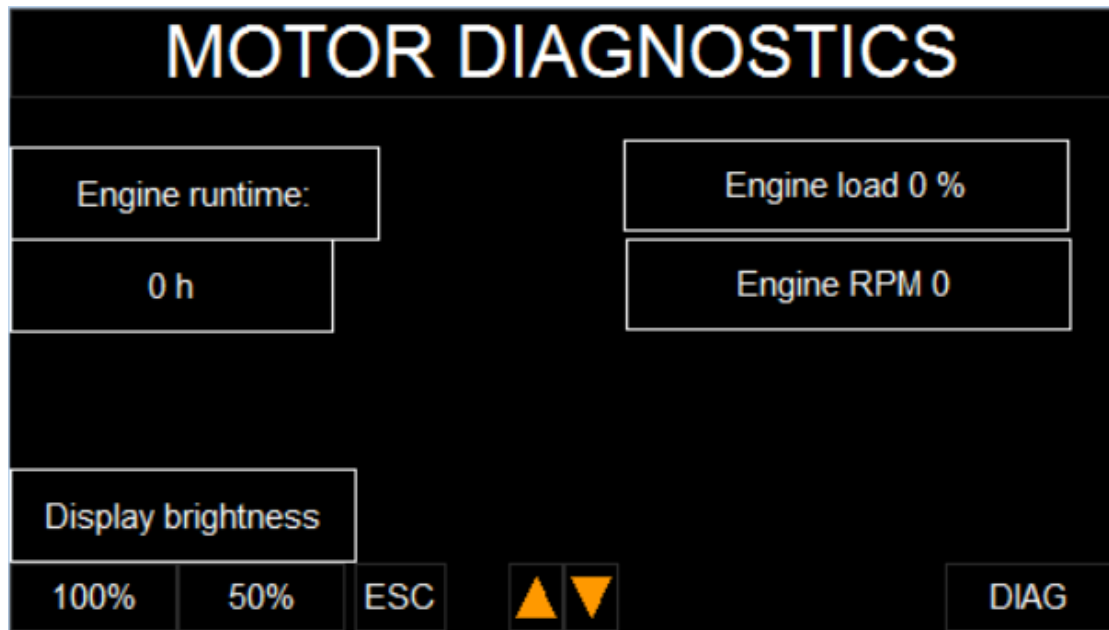


KUVA 15. Näytön perusnäkökulma vikailmaisuuilla

Punainen akun kuva, joka ilmaisee generaattorin vikaa, on ainoa vakavan vian ilmaisu perusnäytössä. Muille moottorin vakaville vioille, joita on korkean jäähdytysnesteen lämpötila ja alhainen öljynpaine on tehty koko näytön kokoinen varoitus vikojen vakavuuden vuoksi. Koko sivun varoitukset sekä ilmoitus siitä, että hätäpysäytyspainike on painettu, esitetään liitteessä 1.

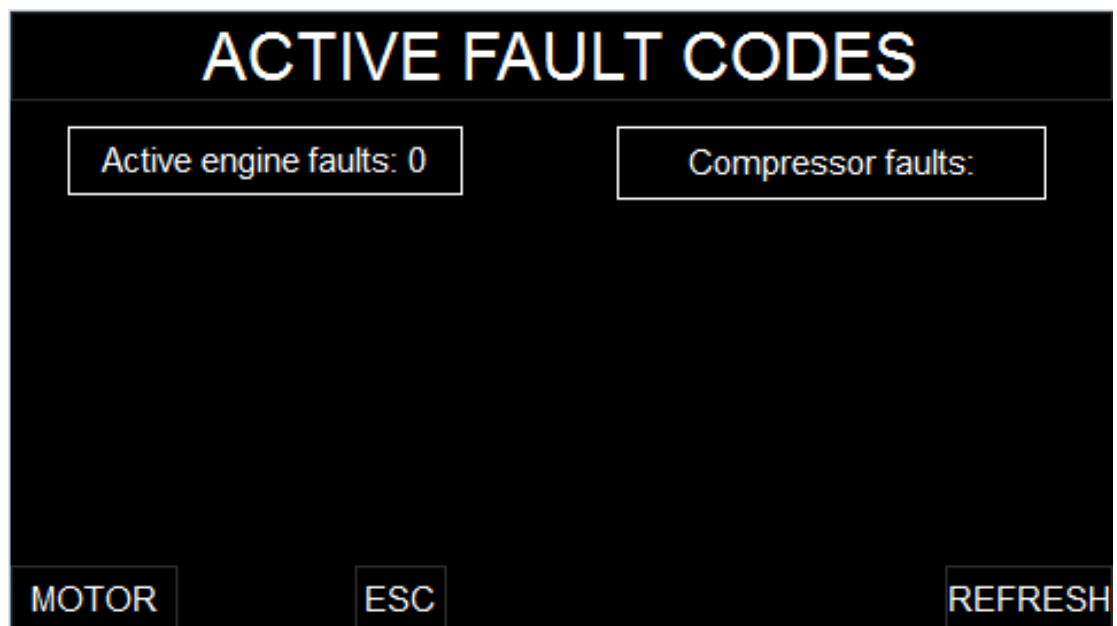
Yhdistetyssä paineen ja virtauksen mittarissa painetieto saadaan anturilta, joka lähettää signaalin milliampeeritietona. Virtausta ei mitata anturilla tässä sovelluksessa vaan mittarin antama tieto on skaalattu ruuvien pyörimisnopeudesta. Pyörimisnopeuden suhde ilman virtausmäärään on esitetty liitteessä 2.

Toinen käyttäjän valittavan oleva sivu on varattu moottorin tiedoille. Sivun päätarkoituksena on näyttää käyttäjälle moottorin kokonaiskäyntiaika, jotta käyttäjä tietää milloin moottorille täytyy suorittaa huolto. Sivun tyhjään tilaan voidaan lisätä käyttökokemusten perusteella tarpeelliseksi koettuja tietoja moottorista. Tässä revisioissa sivulle on lisätty moottorin kuormitusta sekä kierroslukua ilmaisevat kentät kuvan 16 mukaisesti. Moottorin tietojen lisäksi tältä sivulta voidaan säätää näytön taustavalon kirkkautta.



KUVA 16. Moottorinäyttö

Kolmas käyttäjän valittavissa oleva sivu on varattu aktiivisille vikakoodeille (kuva 17). Mikäli päänäkymässä esitetty moottorin vikavallo syttyy, voi käyttäjä tarkistaa tältä sivulta moottorilta tulevat aktiiviset vikakoodit sekä kompressorin raja-arvojen ylitykset. Moottorin aktiiviset vikakoodit ilmoitetaan numerosarjana, jonka avulla käyttäjä voi tarkistaa moottorin ohjekirjasta vikakoodin selostuksen. Sivulla näytetään moottorin aktiivisten vikakoodien määrä sekä maksimissaan viisi vikakoodia. Näytön vikakoodit voi päivittää painamalla F4 -painiketta. Kompressorilta tulee tietoa seitsemältä eri anturilta ja mikäli raja-arvot ylitetään, niin näytöllä ilmaistaan miltä anturilta tieto tulee.



KUVA 17. Vikanäyttö

6 POHDINTA

Opinnäytetyön tuloksena saatiin kehitettyä suunniteltavalle laitteelle ohjausjärjestelmän ohjelmapohja, jonka kehittämistä on helppo jatkaa testaus- ja käyttökokemusten pohjalta. Aikataulullisista syistä johtuen ohjelmaa ei valitettavasti päästy testaamaan toimintakunnossa olevalla laitteella, vaan kaikki tehty testaus perustuu Codesys -ohjelmointiympäristön simulointiominaisuuksiin.

Aikataulullisesti työ eteni johdonmukaisesti eteenpäin kevään ajan hieman hitaan alun jälkeen. Työn jatkuvan etenemisen takasi se, että Tampereen Konepaja Oy oli aktiivisesti mukana kehityksessä ja auttoivat aina ongelmien esiintyessä. Joidenkin komponenttivalintojen osalta aikataulu venyi tai niitä ei päästy ollenkaan tekemään tämän työn puitteissa, johtuen muun suunnittelutyön tilasta. Myös Scanian moottorin optiona valittavat osat varmistuivat melko myöhään, joka aiheutti pieniä muutoksia ohjausjärjestelmään ja käyttöliittymään. Kehitetty ohjausjärjestelmä on kuitenkin pyritty toteuttamaan siten, että jokainen ohjelmalohko on kommentoitu selkeästi, jotta ohjelman muokkaaminen ja korjaaminen olisi mahdollisimman helppoa ja vaivatonta.

Työssä käytetyt Codesys -ohjelmointiympäristö ja CAN -väylä ovat hyvin laajasti käytössä teollisuuden eri osa-alueilla, joten työstä saatu osaaminen tulee varmasti olemaan arvokasta tulevaisuudessa. Myös kokemus projektityöskentelystä antaa edellytyksiä nykyaikaiseen insinööriyöhön.

Ohjausjärjestelmän kehitystä tullaan jatkamaan siihen saakka kunnes ensimmäinen valmis tuote saadaan aikaiseksi. Jatkokehitysmahdollisuuksia tämän työn pohjalta tulee varmasti käyttökokemusten pohjalta, kun laite saadaan valmiiksi ja sitä päästään testaamaan. Tulevien asiakkaiden toiveiden ja kehitysehdotusten pohjalta ohjausjärjestelmässä ja käyttöliittymässä on myös mahdollisuus jatkokehitykselle.

LÄHTEET

CAN in automation. luettu 4.3.2016. <http://www.can-cia.org/>

Codesys. luettu 3.3.2016. https://www.codesys.com/#_

Farsi, M. & Barbosa, M. 2010. CANopen Implementation applications to industrial networks. Great Britain: SRP Ltd.

IFM electronics. luettu 15.1.2016. <https://www.ifm.com/ifmfin/web/home.htm>

John, K. & Tiegelkamp, M. 2010. IEC 61131-3: Programming Industrial Automation System. 2nd Edition. Germany: Springer

Konedirektiivi. luettu 17.3.2016. <http://eur-lex.europa.eu/legal-content/fi/TXT/?uri=CELEX:32006L0042>

Scania-teollisuusmoottorit, luettu 21.2.2016. <http://www.scania.fi/engines/industrial-engines/scania-industrial-engines-ready-2014/>

SFS-EN ISO 13850 Koneturvallisuus. Hätäpysäytys. Suunnitteluperiaatteet. 17.3.2016. Suomen Standardisoimisliitto SFS ry. SFS Verkkokauppa. <https://online.sfs.fi.elib.tamk.fi/fi/index/tuotteet/SFS/CENISO/ID2/1/404432.html.stx>

SFS ISO 60073 Ihmisen ja koneen välisen rajapinnan perus- ja turvallisuusperiaatteet. Merkinantolaitteiden ja ohjaimien koodaus. 18.3.2016. Suomen Standardisoimisliitto SFS ry. SFS Verkkokauppa. <https://online.sfs.fi.elib.tamk.fi/fi/index/tuotteet/SFSsahko/CENELEC/ID2/6/10397.html.stx>

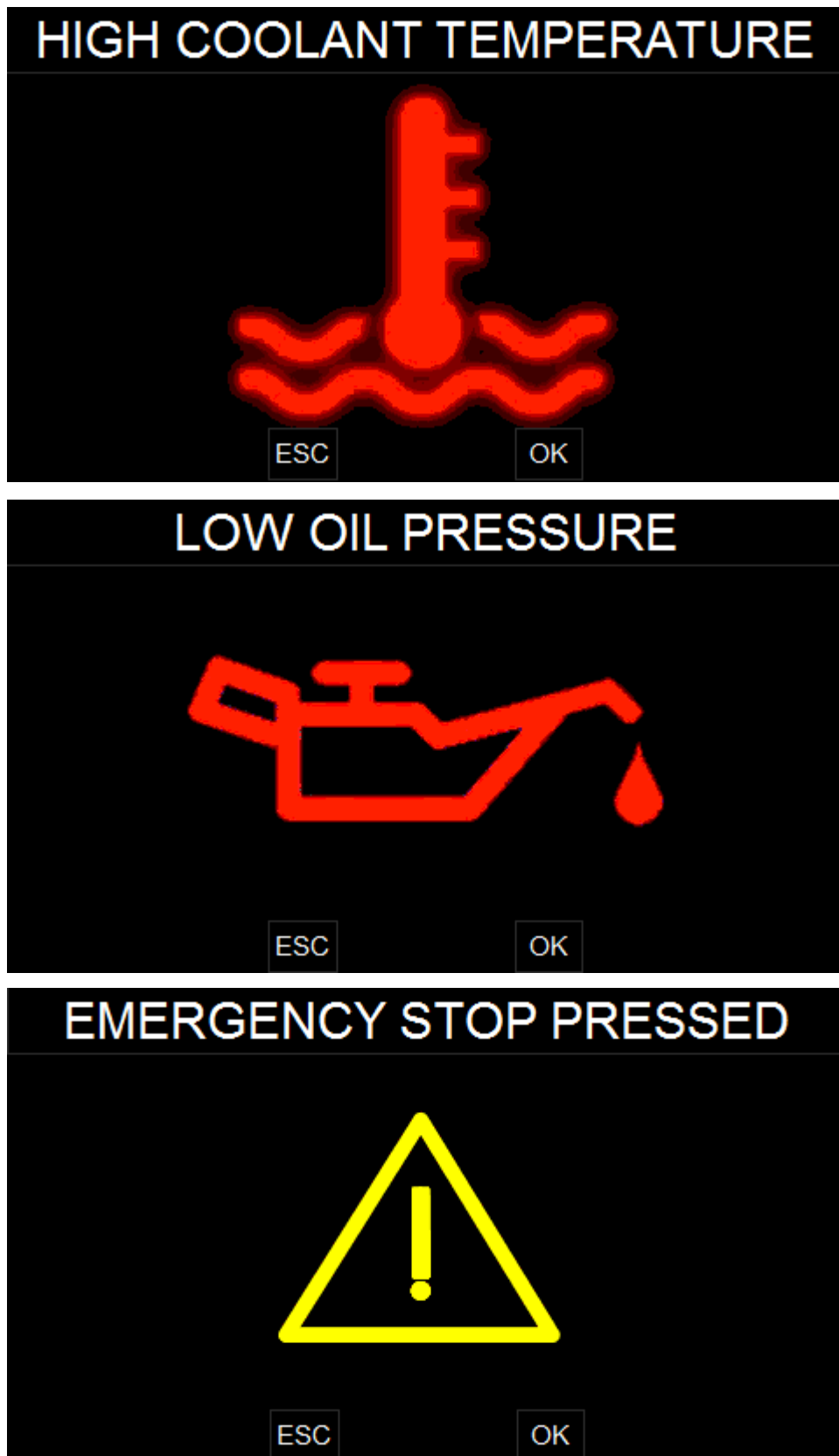
Voss, W. 2005. A Comprehensive Guide to Controller Area Network. 2nd Edition. USA: Copperhill Media Corporation

Voss, W. 2008. A Comprehensive Guide to J1939. USA: Copperhill Media Corporation

3S – Smart Software Solution GmbH. 2007. User Manual for PLC Programming with CoDeSys 2.3

LIITTEET

Liite 1. Käyttöliittymän varoitussivut



Liite 2. Ruuvin pyörimisnopeuden suhde ilmamäärään

