



VAASAN AMMATTIKORKEAKOULU  
VASA YRKESHÖGSKOLA  
UNIVERSITY OF APPLIED SCIENCES

Henri Lepistö

# TOIMILAITTEEN OHJAUS LOGIIKALLA

Tekniikka ja liikenne  
2015

## TIIVISTELMÄ

Tekijä	Henri Lepistö
Opinnäytetyön nimi	Toimilaitteen ohjaus logiikalla
Vuosi	2015
Kieli	suomi
Sivumäärä	63 + 2 liitettä
Ohjaaja	Mika Alatalo ja Juha Nieminen

---

Tämä opinnäytetyö tehtiin Vaasan Sähköasennus Oy:n toimesta. Työssä suunniteltiin pohjaratkaisu teollisuuslaitteistolle, joka toimii automaatiojärjestelmällä. Lisäksi työssä käytiin läpi eri vaiheita, mitä kuuluu automaatiojärjestelmän suunnitteluun.

Lähtökohtana oli suunnitella automaatio kuvauksen perusteella toimiva pohjaratkaisu toimilaitteen käytölle. Työssä valittiin käyttötarkoitukseen sopiva logiikkaohjain ja muut komponentit ja näille piirrettiin piirikaaviot. Logiikkaan ohjelmoitiin toimiva järjestelmä ja käyttöliittymä, jonka avulla seurattiin toimintaa. Järjestelmän toiminta toteutettiin esimerkki konfiguraation avulla.

Lopputuloksena syntyi toimiva automaatiojärjestelmän avulla toimiva teollisuuslaitteisto. Koska kyseessä oli pohjaratkaisu, sitä tullaan tulevaisuudessa muokkaamaan käyttökohteen mukaan sopivaksi.

---

Avainsanat	automaatio, piirikaavio, logiikkaohjain, ohjelmointi, käyttöliittymä
------------	--

## ABSTRACT

Author	Henri Lepistö
Title	PLC for an Actuator
Year	2015
Language	Finnish
Pages	63 + 2 Appendices
Name of Supervisor	Mika Alatalo and Juha Nieminen

---

This thesis was made by Vaasan Sähköasennus Oy. In this thesis the layout was designed for an industrial device, which is operated by an automation system. Also various steps included in the designing of an automation system were gone through.

The starting point was to design a workable basic solution for an actuator based on the description of automation. A suitable PLC and other components for this purpose were chosen and the circuit diagram was drawn. The system was programmed as well as a user interface to follow the operation. Finally, the operation of the system was implemented using an example configuration.

As a result we have a working industrial device operated by an automation system. Because this is only a basic solution, it can be modified to suit other application in future.

---

Keywords	Automation, circuit diagram, PLC , programming, user interface
----------	--

## SISÄLLYS

TIIVISTELMÄ

ABSTRACT

KUVIO- JA TAULUKKOLUETTELO

LYHENTEET

LIITELUETTELO

1	JOHDANTO.....	10
2	VAASAN SÄHKÖASENNUS OY - VASEL .....	11
	2.1 Vaasan Sähköasennus Oy lyhyesti.....	11
	2.2 Vaasan Sähköasennus Oy:n palvelut .....	11
3	OHJELMOITAVA LOGIIKKA .....	12
	3.1 Historia.....	12
	3.2 Perusominaisuudet .....	13
	3.2.1 I/O-liitännät .....	13
	3.2.2 Nopea ja käytännönläheinen ohjelmointi.....	14
	3.2.3 Käyttövarmuus, hinta ja tiedonkeruu .....	14
	3.3 Ohjelmoitavien logiikkojen rakenne .....	14
4	AUTOMAATION KUVAUS .....	16
	4.1 Hyvä tietää suunnittelussa.....	16
	4.2 Järjestelmän kuvaus .....	17
5	LOGIIKOIDEN VERTAILU .....	18
6	LAITTEIDEN VALINTA.....	20
7	KÄYTTÖLIITTYMÄ .....	22
	7.1 Käyttöliittymän tarkoitus .....	22
	7.2 Käyttöliittymän yleistiedot.....	22
	7.3 Ominaisuudet .....	23
8	PIIRIKAAVIOT .....	24
	8.1 Yleistä piirikaavioista .....	24
	8.2 Piirikaavioiden piirtäminen.....	25
9	OHJELMOINTI JA KONFIGURAATIO.....	29

9.1 Yleistietoa toimilohkojen luomisesta.....	29
9.1.1 Moottoripiiri.....	29
9.1.2 Venttiilipiiri.....	37
9.1.3 Taajuusmuuttajapiiri .....	43
9.1.4 Analogiatulojen muuttaminen.....	52
10 OHJELMAN LUOMINEN TOIMILOHKOILLE .....	54
10.1 Toimilohkojen ohjelmointi .....	54
10.2 Taajuusmuuttajan ohjelmointi .....	56
10.3 Käyttöliittymän luominen .....	57
10.4 Testaus .....	59
10.4.1 Yleistä testauksesta .....	59
10.4.2 Laitteiston testaaminen.....	59
11 YHTEENVETO .....	60
LÄHTEET.....	61
LIITTEET	

## KUVIO- JA TAULUKKOLUETTELO

<b>Kuvio 1.</b> Logiikan kehitysaskeleita. /3/ .....	13
<b>Kuvio 2.</b> CPU:n sijoittuminen automaatiojärjestelmissä. /3/.....	15
<b>Kuvio 3.</b> Automaation kuvaus. ....	17
<b>Kuvio 4.</b> Valittu CPU Siemens Simatic S7-1200 .....	21
<b>Kuvio 5.</b> Siemens KTP 700 Basic. /7/ .....	23
<b>Kuvio 6.</b> Moniviivaesitys vasemmalla ja yksiviivaesitys oikealla. ....	25
<b>Kuvio 7.</b> Koordinaatiomerkinnot .....	25
<b>Kuvio 8.</b> Logiikan kytkentäpisteet /9/ .....	27
<b>Kuvio 9.</b> Taajuusmuuttajan I/O – kortin kytkentäpisteet. /10/ .....	28
<b>Kuvio 10.</b> Moottoripiirin toimilohko .....	30
<b>Kuvio 11.</b> Apukoskettimien indikointi .....	31
<b>Kuvio 12.</b> Lukituspiirin ja paikallisohjauksen toteutus .....	31
<b>Kuvio 13.</b> Manuaalitilan valinta ja manuaalitilan valvomotilan resetointi. ....	32
<b>Kuvio 14.</b> Automaattitilan valinta ja automaattitilan valvomotilan resetointi. ....	33
<b>Kuvio 15.</b> Moottorin start-komento ja manuaali ohjauksen valvomotilan resetointi. ....	34
<b>Kuvio 16.</b> Moottorin stop – komento ja manuaalisen valvomotilan resetointi. ...	35
<b>Kuvio 17.</b> Moottorin kontaktorin valvonta ja ohjauksen pitopiiri. ....	36
<b>Kuvio 18.</b> Moottorin vika paineen muutoksesta. ....	37
<b>Kuvio 19.</b> Venttiilipiiri .....	37
<b>Kuvio 20.</b> Venttiilin lukituspiiri ja paikallisohjauksen valinta. ....	38
<b>Kuvio 21.</b> Venttiilin manuaalitilan valinta ja valvomo-ohjauksen resetointi. ....	39
<b>Kuvio 22.</b> Venttiilin automaattitilan valinta ja valvomo-ohjauksen resetointi. ....	39
<b>Kuvio 23.</b> Venttiilin avaaminen ja valvomotilan resetointi. ....	40
<b>Kuvio 24.</b> Venttiilin kiinni toiminto ja valvomotilan resetointi. ....	41
<b>Kuvio 25.</b> Venttiilin ohjauksen pitopiiri, kiinniohjaus ja rajojen ristiriita. ....	42
<b>Kuvio 26.</b> Venttiilin auki ja kiinni indikoinnit sekä vika toiminto. ....	43
<b>Kuvio 27.</b> Taajuusmuuttajan lohko. ....	44

<b>Kuvio 28.</b> Vasemmanpuoleiset bitit ovat taajuusmuuttajan status wordiä ja control wordit oikeanpuoleisia bittejä. ....	45
<b>Kuvio 29.</b> Sw_addr -muuttujan purkaminen biteiksi.....	46
<b>Kuvio 30.</b> Bittien pakkaaminen cw_addr -muuttujaan.....	48
<b>Kuvio 31.</b> Manuaalitilan valinta. ....	49
<b>Kuvio 32.</b> Automaattitilan valinta. ....	49
<b>Kuvio 33.</b> Start –reunan toteutus ja valvomotilan resetointi. ....	50
<b>Kuvio 34.</b> Stop –reuna ja valvomotilan resetointi. ....	51
<b>Kuvio 35.</b> Taajuuden asetus lohkoon ja taajuusohje manuaalitilassa.....	52
<b>Kuvio 36.</b> Paine- ja virtausanturit fb –lohkoina.....	53
<b>Kuvio 37.</b> Standardiviestin muuttaminen reaalityyppiseksi.....	53
<b>Kuvio 38.</b> Paine – ja virtausanturin muutokset reaali – tyyppi. ....	54
<b>Kuvio 39.</b> Moottorille asetetut toiminnot. ....	55
<b>Kuvio 40.</b> Venttiilille asetetut toiminnot. ....	55
<b>Kuvio 41.</b> Järjestelmän laitteet network viewissä.....	56
<b>Kuvio 42.</b> Taajuusmuuttajan toiminnot. ....	57
<b>Kuvio 43.</b> Näkymä käyttöliittymästä. ....	58
<b>Taulukko 1.</b> Ohjattavat komponentit .....	21

**LYHENTEET**

PLC	Programmable Logic Controller	ohjelmoitava logiikka
I/O	Input/Output	tulo/lähtö
CPU	Central Processing Unit	keskussuoritin
FM	Function Module	toimintomoduuli
DI	Digital Input	digitaalitulo
AI	Analog Input	analogiatulo
DO	Digital Output	digitaalilähtö
AO	Analog Output	analogialähtö
RAM	Random Access Memory	keskusmuisti
PII	Process Input Image	tulorekisteri
PQI	Process Output Image	lähtörekisteri
TFT	Thin Film Transistor	ohutkalvotransistori
FB	Function Block	toimilohko



**LIITELUETTELO**

**LIITE 1.** Moottori - ja taajuusmuuttajalähdöt.

**LIITE 2.** Logiikan kytkentäkaavio.

## 1 JOHDANTO

Opinnäytetyö on tehty Vaasan ammattikorkeakoulussa syksyllä 2014. Tilaajana toimi Vaasan Sähköasennus Oy – Vasel. Työssä suunniteltiin pohjaratkaisu toimilaitteelle, jota ohjattaisiin automaatiojärjestelmällä.

Opinnäytetyön tarkoituksena oli suunnitella pohjaratkaisu logiikalla ohjattaville moottori- ja pumppukäyttöille, jota voitaisiin käyttää tulevaisuudessa erilaisissa teollisuuslaitteistoissa ja muokata kohteeseen sopivaksi järjestelmäksi.

Työn tavoitteena oli ohjata logiikan avulla moottoria ja taajuusmuuttaja ohjattua pumppua, jotka saavat ohjausarvot antureilta myös venttiilin ohjaus tapahtui ohjelmoitavan logiikan avulla. Työssä vertailtiin Siemensin ja Beckhoffin tuotteiden yleisiä ominaisuuksia ja muut komponentit valittiin mahdolliselle toimilaitteelle ja logiikalle sopiviksi. Työssä suunniteltiin piirikaaviot ja logiikan ohjelmointi sekä valittiin sopiva käyttöliittymä. Käyttöliittymälle suunniteltiin selkeä ja helppolu-  
kuinen toimintaympäristö. Lopuksi työ toteutettiin esimerkki konfiguraation avul-  
la.

## **2 VAASAN SÄHKÖASENNUS OY - VASEL**

### **2.1 Vaasan Sähköasennus Oy lyhyesti**

Vaasan sähköasennus Oy on vuonna 2001 perustettu sähköurakointiliike ja toiminta sijoittuu pääasiassa Vaasaan ja sen lähikuntiin. Toimintaan sisältyy suunnittelun, asennuksen, huollon ja tarvikkeiden myynnin lisäksi myös ilmalämpöpumppujen myynti, asennus ja huolto. /1/

### **2.2 Vaasan Sähköasennus Oy:n palvelut**

Vaasan Sähköasennus Oy:n palveluihin kuuluvat sähkösuunnitelmat ja – työt, edellä mainittu ilmalämpöpumppujen asennus ja myös pienet kaivuutyöt kuten kaapeliojat ja salaojat.

Suunnittelu sisältää rakennusten sähkö- ja valaistussuunnittelun, taloautomaatiojärjestelmien suunnittelun sekä teollisuuden komponentti- ja järjestelmäsuunnittelun.

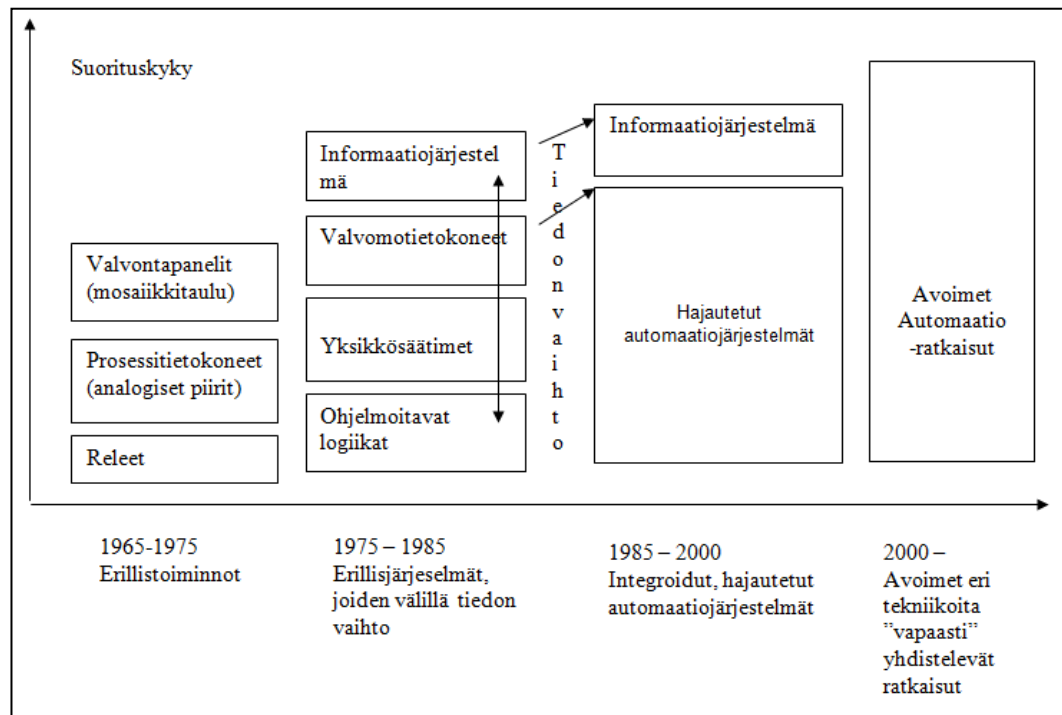
Sähkötyöt sisältävät sähköasennukset ja -huollon sekä seuraavia erilaisten järjestelmien asennuksia: ATK- ja antenniverkot, taloautomaatiojärjestelmän KNX, valaistuksenohjausjärjestelmät esim. DALI ja DMX, ylijännitesuojaukset sekä induktiosilmukat ja vahvistimet. /2/

## 3 OHJELMOITAVA LOGIIKKA

### 3.1 Historia

Relelogiikoilla tehtiin ensimmäiset automaatiototeutukset eli logiikka toteutettiin mekaanisten releiden avulla. Elektroniikan kehittyessä 1960-luvun alussa syntyivät ensimmäiset elektroniset logiikkajärjestelmät. Toteutus tapahtui langoittamalla elektronisia kombinaatiopiirejä ja niitä kutsuttiinkin kytkentöjen mukaisesti, langoitetuksi logiikoiksi. Vaikka integrointiaste paranikin huomattavasti, niin halutun toiminnan aikaansaaminen vaati edelleen käsin suoritettavaa johdotusta, minkä vuoksi muutostyöt ja dokumentointi olivat kuluttavaa. PLC:t eli ohjelmoitavat logiikat tulivat markkinoille 1969. Liikkeenlepanon suoritti Yhdysvaltojen autoteollisuus, joka edellytti joustavampia tapoja toteuttaa automaatiota.

Aluksi ohjelmoitavilla logiikoilla ohjattiin toimilaitteita, kuten moottoria, venttiilejä ja lamppuja. Nykyisin ohjelmoitavilla logiikoilla ohjataan yhä enemmän prosesseja ja kerätään mahdollista tietoa prosessin kulusta kuten hälytyksiä, indikoimiteja ja mittauksia sekä ohjataan prosessin toimintoja. Lisäksi, jos prosessissa on useita eri ohjelmoitavia logiikoita, voidaan nämä yhdistää samaksi kokonaisuudeksi ja jakaa prosessointikuormaa usealle eri keskusyksikölle. Kuviossa 1 on kuvattu logiikan eri kehitysaskelaita 1960-luvulta tähän päivään saakka. /3/



**Kuvio 1.** Logiikan kehitysaskeleita. /3/

## 3.2 Perusominaisuudet

Logiikoiden edistyessä alkoivat myös muut ominaisuudet kehittyä kuten liitännämahdollisuudet, nopeus ja käyttövarmuus. Samassa ajassa myös tiedonkeruu kehittyi ja logiikoita saatiin liitettyä keskenään toisiinsa. Myös jatkuva kehittyminen on saanut aikaan sen, että hinta tasoa on saatu alemmaksi. /3/

### 3.2.1 I/O-liitännät

I/O-liitännät (Input/Output) ovat logiikoiden tyypillisin ominaisuus, tällä toiminnolla suoritetaan tilatietojen ja mittausten välitys ohjelmaan ja takaisin. Nämä ovat juuri niitä ominaisuuksia, joita varten logiikka on kehitetty ja näiden välitys on tehty mahdollisimman yksinkertaiseksi komponentin ja ohjelman välillä. /3/

### 3.2.2 Nopea ja käytännönläheinen ohjelmointi

Kun tiedetään ohjattavan kohteen toiminta, on tarvittavan ohjelman tekeminen suhteellisen helppoa. Ohjelman rakenne vastaa ohjattavan kohteen rakennetta. Ohjelmien muuttaminen ja dokumentointi on nopeaa, mikä tekee laajankin toiminnan mahdolliseksi. Yleisempiä ohjelmointikieliä ovat basic- ja c-kieli, jotka toimivat yleisempien käytössä olevien tietokoneiden kanssa. Suurilla valmistajilla on oma optimoitu prosessorinsa bitti – operaatioiden suorittamiseen ja tämän ansioista logiikat ovat nopeita, joille tyypillinen reaktioaika on alle 10 ms. Jos haluaa nopeampaa reaktioaikoja voi käyttää tavallisimpia CPU:ta (Central Processing Unit) ja/tai FM (Function Module) -erikoiskortteja, joissa on oma prosessori nopeuttamassa toimintoja. /3/

### 3.2.3 Käyttövarmuus, hinta ja tiedonkeruu

Logiikka on suunniteltu kestävämpään teollisuusympäristön vaatimukset ja sen vuoksi logiikat ovat käyttövarmempia, kuin mekaaniset releet ja jopa luotettavampia kuin tietokoneet. Logiikoiden hinnat alkavat n. 100 eurosta ja yhdellä logiikalla voidaan korvata kymmeniä releitä. Logiikat voidaan kytkeä väylän avulla toisiinsa, jotta ne voivat kerätä tietoa toisiltaan. Samaan väylään voidaan kiinnittää myös ylempällä tasolla toimiva tietokone, jolloin logiikat toimivat tiedonkeruuyksikkönä. /3/

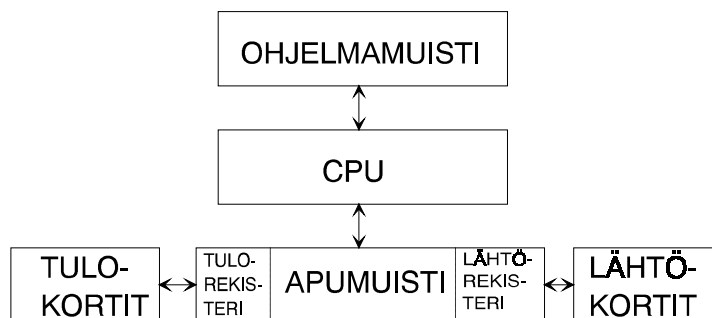
## 3.3 Ohjelmoitavien logiikkojen rakenne

Ohjelmoitavat logiikat koostuvat erilaisista korteista, muisteista sekä keskusyksikkö CPU:sta. Korteja ovat tulo- ja lähtökortit, joita on sekä digitaalisia että analogisia. Tulokortit eli DI (Digital Input)- tai AI (Analog Input) -kortit välittävät prosessin toiminnasta jännite- ja virtasignaaleja CPU:lle. Lähtökortit DO (Digital Output- tai AO (Analog Output) -kortit välittävät ohjauskäskyt toimilaitteille ja soveltavat ohjaussignaalien jännite- ja virtatasot toimilaitteille. Ohjelmamuistina käytetään RAM-muistia, jonne tallennetaan käyttäjän ohjelmat ja tämän koko riippuu CPU:n tyypistä.

Järjestelmämuistiin (System Memory) sisältyy digitaalitulojen PII (Process Input Image) –muisti ja lähtöjen PQI (Process Output Image) –muisti. PII-muistilta lue-  
taan tilat ohjelman suorituksen aikana, jonne ne siirretään ennen ohjelmakierron  
alkua. Ennen ohjelmakiertoa PQI-muistiin tallennetaan lähtökanavien tilat ja kier-  
ron lopussa ne siirretään DO-kortille. Tila säilyy DO -kortilla niin kauan kunnes  
tulee uusi ohjaus. Lisäksi järjestelmämuistiin sisältyy seuraavia ominaisuuksia:

- apumuisti, jota voidaan käyttää nimen mukaisesti apuna ohjelman muo-  
dostamisessa
- ajastimet joita käytetään, että saadaan tiettyyn aikaan toiminta suoritettua
- laskurit jotka laskevat joko ylöspäin tai alaspäin
- latausmuistia käytetään siihen, että sinne tallennetaan käyttäjän ohjelma  
ladattavaksi logiikan käynnistyessä.

CPU on keskusyksikkö joka suorittaa apumuistiin tallennettujen arvojen väliset  
lukitukset ja laskutoimitukset. CPU:n sijoittuminen automaatiojärjestelmässä on  
kuvattu kuvioon 2. /3/



**Kuvio 2.** CPU:n sijoittuminen automaatiojärjestelmissä. /3/

## 4 AUTOMAATION KUVAUS

### 4.1 Hyvä tietää suunnittelussa

Automaattisesti toimivan koneen suunnittelu lähtee liikkeelle käyttäjäliittymien ja toimintavaatimusten selvittämisestä. Yleisesti automaatio pyritään pitämään mahdollisimman yksinkertaisena ja selkeänä, mutta järjestelmän toteutus kuitenkin monimutkaistuu, kun siihen lisätään erilaisia turvallisuustoimintoja. Suunnittelu-projekti voi olla uusi laitteiston suunnittelu tai vanhan koneen automatisointi. Useimpien automaattisten koneiden ohjaukseen käytetään tietokonepohjaisia järjestelmiä ja kiinteitä ohjauslogiikoita. /4/

Kohteen suunnittelussa on hyvä olla selkeä jäsenyys, jotta tilaajan ja suunnittelijan yhteistyö olisi sujuvaa. Suunnittelun tulos on aineisto, jossa suunnittelijat tuottavat ja käyttävät lähtötietona tätä aineistoa ja tuotteen käyttäjät hyödyntävät ja täydentävät suunnittelijoiden tietoja. Saatu tieto voidaan jakaa kolmeen alueeseen: vaatimuksiin, toiminnallisiin kuvauksiin ja toteutuksen kuvauksiin.

Kokonaisen järjestelmän suunnittelu on hyvä aloittaa vaatimusten määrittelystä. Asiakas kuvaa ulospäin näkyvän toiminnon ja kertoo mitä teknisiä ominaisuuksia asiakas haluaa. Sisäistä toimintaa eli logiikan rakennetta ei kerrota ja muille toteutuksille on hyvä jättää mahdollisimman paljon vaihtoehtoja. Suunnittelussa pitää ottaa huomioon jo valmiiksi määritellyt rajoitukset.

Toiminnallisessa kuvauksessa määritellään millaisilla tietorakenteilla ja toiminoilla vaatimukset voidaan toteuttaa. Vaikka kuvaus painottuu loogiseen tasoon, siinä myös kuvataan karkeasti toteutuksen rakennetta. Automaatiotoimituksissa tämä on tiedossa, koska tämän usein tekee toimittaja. Toiminnallinen kuvaus on asiakkaan ja toimittajan sopimuksen perusta, joka pitää olla molempien ymmärrettävissä.

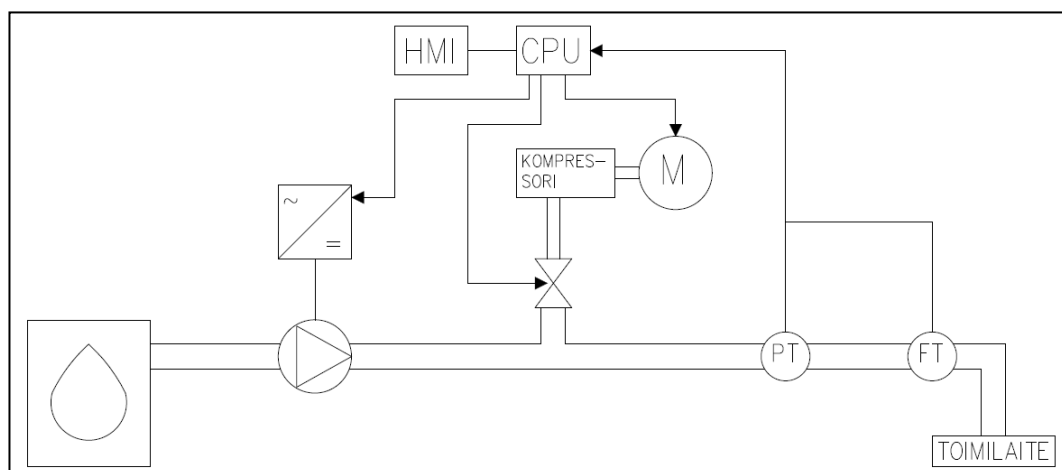
Toteutuksen kuvauksessa käsitellään laitteita ja ohjelmistoja. Tämä kuvaus on luontevin arkielämässä ja tämä painottuu toteutustason yksityiskohtiin. Kohteet



voivat olla tuotantolaitos, ohjausjärjestelmä, koneet jne. nämä ovat järjestelmiä, joilla on tietty käyttötarkoitus ja jotka muodostuvat olioista kuten laitteista, ihmisistä ja ohjelmistoista. Järjestelmä toteuttaa tarvittavat toiminnot ja se muodostuu pienimmistä toiminnoista ja lopulta komponenteista, joten esimerkiksi koko tehdas voi olla järjestelmä. Järjestelmän rajausta riippuu näkökulmasta, esimerkiksi tietokone on järjestelmä. Siinä eri käyttäjillä on eri sovelluksia ja ne ovat niiden käyttäjien järjestelmiä. Joten tällaisessa tapauksessa järjestelmät voivat mennä päällekkäin eri näkökulmien vuoksi. /5/

#### 4.2 Järjestelmän kuvaus

Automaatiojärjestelmällä on tarkoitus ohjata mahdollista korkeapainekoneikkoa. Järjestelmässä paine- ja virtausanturin mukaan ohjattaisiin moottorikäyttöä ja taajuusmuuttaja ohjattua pumppua. Moottorikäyttöön liitettiin kompressori, jolla tuotettaisiin painetta järjestelmään solenoidiventtiilin kautta ja tuotettua painetta säädettäisiin paineanturin arvoilla. Taajuusmuuttaja ohjatulla pumpulla säädettäisiin nesteen määrä toimilaitteeseen ja virtauksen määrää seurattaisiin virtausanturilta saatavilta arvoilta. Koko järjestelmää ohjattiin kosketusnäytöllisellä käyttöliittymällä. Järjestelmän kuvaus on kuvattu kuvioon 3.



**Kuvio 3.** Automaation kuvaus.

## 5 LOGIIKOIDEN VERTAILU

Tässä luvussa otettiin selvää miten Siemensin S7-1200 ja Beckhoffin CX9020 loogiikat toimivat eri ominaisuuksien perusteella. Tietoja selvitettiin yhtiöiden Internet-sivujen kautta sekä puhelimitse.

Ensimmäiseksi tarkasteltiin tapahtumien, pääasiassa hälytyksien ja vikojen, rekisteröintiä muistiin ja niiden selaamista. Siemensissä rekisteröinti tapahtuu TIA Portal -ohjelmiston avulla tai paneeliin ohjelmoiduilla muistipaikoilla. Paneeliin saatiin myös aikaleima, josta nähtiin milloin tapahtuman on kirjattu. Beckhoffin osalta laitteisto hälytykset tulivat näkyviin twincat-ohjelmiston sisällä ja tarvittaessa ne voitiin tuoda Indusoft-käyttöliittymään, jossa ne näytettiin taulukkona ja tapahtuma historian voitiin tallentaa erilaisiin tietokantoihin tai tiedostoihin.

Seuraavaksi tiedusteltiin pystyykö käyttöliittymiin tekemään käyttäjäkohtaisia toimintoja, joita pystyi tarkastelemaan vain tietyt henkilöt. Siemensin ja Beckhoffin tuotteissa voitiin tehdä käyttäjäkohtaiset liittymät salasanan avulla. Myös käyttäjien tekemät toiminnot eri komponentteihin voitiin selata ohjelmoinnin avulla.

Ohjelmointiin ja käyttöliittymän tekemiseen Siemensin osalta tarvittiin TIA Portal -ohjelmiston, jossa on kaikki komponentit mitä tarvittiin koko tuotantoketjun automaation suunnitteluun ja prosessointiin. Ohjelmaan sisältyi mm. ohjelmistojen suunnittelu, käyttöliittymien tekeminen sekä taajuusmuuttaja- ja valvomo -ohjelmistot. Kyseinen ohjelmisto toimii uudemmilla Windows alustoilla ja tabletteja varten ohjelmisto kaupoissa on valmiita sovelluksia. Beckhoffin ohjelmiston voitiin toteuttaa Twincat 2-ohjelmistolla, jolla voitiin hallita projektien ohjelmoinnin ja konfiguraatiot. Twincat 2-ohjelmisto on tarkoitettu plc -ohjelmointiin, digitaalisten ja analogisten tulojen ja lähtöjen ohjelmointiin. Twincat toimii myös uusimmissa Windows - alustoissa. Tabletteja varten on indusoft-ohjelmisto, jolla

voitiin tehdä erilaisia käyttöliittymiä ja sitä voidaan käyttää tabletissa selaimen kautta.

Tabletin käyttäminen onnistui molemmissa logiikoissa vakiona kuuluvan ethernet-kaapeliominaisuuden avulla. Siemensiin on saatavilla erillinen WLAN-komponentti ja bechhoffiin riittää, kun kytkee morkkulan kiinni laitteeseen. Siemensillä on tabletteja varten omat sovellukset ios- ja android alustoille. Beckhoffille vastaavia tuotteita ei ole löytynyt kirjoittamisen hetkellä. Molemmilta valmistajilta saatiin myös kiinteät näyttöpaneelit, joihin on valmiina tai niihin voitiin tehdä tarpeen mukaan komponenttien ”laatikot”.

Tarvittaessa suurempia teollisuus kokonaisuuksia varten käytettiin väyläliityntöjä. Molemmilta valmistajilla oli yleisimpiä liityntöjä, näitä on joko vakiona laitteissa tai niitä voitiin lisätä erillisillä korteilla. Siemensillä yleisempiä liityntöjä ovat ethernet, profinet, profibus, modbus ja RS-väylät. Beckhoffilta saatiin näitä samoja liityntöjä sekä lisäksi etherCAT, canopen ja deviceNet.

## 6 LAITTEIDEN VALINTA

Tässä luvussa kerrotaan millaisiin laitevalintoihin päädyttiin niin logiikan, taajuusmuuttajan ja muiden komponenttien osalta.

Siemensin logiikkaan päädyttiin heidän suuren ohjelmisto palveluiden ja uuden TIA Portal -ohjelmiston vuoksi. Myös maailmanlaajuisuus sekä helppokäyttöisyys, että maailmalla kehuttu uusi käyttöjärjestelmä puolsivat valintaa. Helposti toteutuvat ohjausjärjestelmät kuten I/O-liittynät ja kompaktit ratkaisut väyläominaisuuksiin olivat eduksi Siemensin valintaan. Valmiina olevat helppokäyttöiset käyttöliittymät sekä komponentit tableteille, kuten android ja ios -alustoille vahvistivat myös valintaa Siemensiin. Muut laitteet kuten moottori, paine- ja virtausanturit ja venttiilit valittiin siten että ne ovat yhteensopivia toimimaan Siemensin logiikan kanssa ja mahdollisen toimilaitteen kanssa.

Varsinaiseksi logiikaksi valittiin Siemens Simatic S7-1200 sarjan CPU:n, tarkemmin CPU 1212C AC/DC/Rly (**Kuvio 4.**), jossa on 8 digitaalituloa ja 6 digitaalilähtöä ja analogia tuloja on 2 kpl. Myös yhdelle 24 voltin tasajännite lähdölle oli mahdollisuus. CPU:n oli sisälletty liityntä profinet-väylälle ja käyttöjännitte otettiin 230 voltin verkosta. Käyttöliittymäksi valittiin Siemens KTP 700 Basic, jonka ominaisuuksista kerrotaan enemmän luvussa 7. Taajuusmuuttajaksi valittiin Vaccon 10 -sarjan 1-vaiheinen taajuusmuuttaja, jonka teho on 0.75 kW. Lisäksi taajuusmuuttajalle valittiin OPTE9-lisäoptiokortti, jossa oli kaksi profinet-liityntää. Tällä saatiin kommunikointi toimimaan taajuusmuuttajan, CPU:n ja käyttöpaneelin välillä, tämän vuoksi ylimääräiset kytkimet voitiin jättää pois. Työssä käytettiin taulukon 1 mukaisia komponentteja.

**Taulukko 1.** Ohjattavat komponentit

Moottori	3-vaihe	1,1 kW	1500 rpm	
Taajuusmuuttaja	1 – vaihe	0,75 kW		
Pumppu	Taajuusmuuttaja-ohjaus	1 – 10 V	0,5 kW	100 l / min
Solenoidiventtiili	24 VDC			
Paineanturi	0 – 6 bar			
Virtausanturi	0 - 100 l/min			

**Kuvio 4.** Valittu CPU Siemens Simatic S7-1200

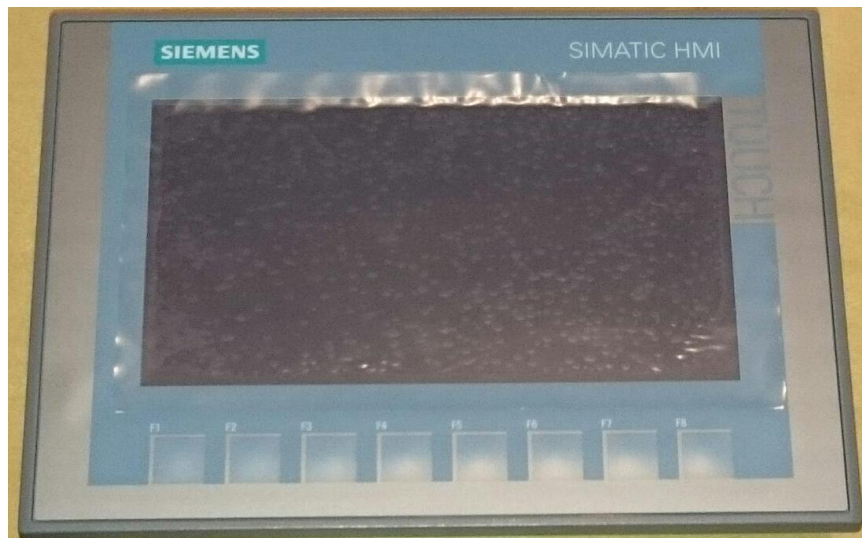
## **7 KÄYTTÖLIITTYMÄ**

### **7.1 Käyttöliittymän tarkoitus**

Käyttöliittymä on operaattorin ikkuna valvottavaan kohteeseen. Nykyään käyttöliittymä mielletään pelkäksi tietotekniikaksi, mutta siihen kuuluu myös ohjauskytkimet, merkkilamput sekä näkö- ja kuuloyhteys. Käyttöliittymä siis muodostuu fyysisistä laitteista ja ohjelmistosta, jotka tarjoavat automaatiojärjestelmän näkyvää osaa käyttäjälle eli informaation sisältöä ja toimintaa. Automaatiojärjestelmän käyttöliittymä muodostuu valvomoon ja/tai kentälle sijoitetuista käyttöliittymä laitteista kuten näppäimistöistä ja näyttöpaneelistä sekä tarvittavasta ohjelmistosta. Nykyisillä käyttöliittymillä voidaan hallita useita eri järjestelmiä, mutta vielä tarvitaan perinteisiä ratkaisuja kuten hätä-seis-painikkeita ja merkkilamppuja. /6/

### **7.2 Käyttöliittymän yleistiedot**

Käyttöliittymäksi valittiin Siemensin näyttöpaneeli KTP 700-Basic, joka on esitetty kuviossa 5. KTP 700-Basic:ssä on 7 tuuman TFT (Thin Film Transistor) – kosketusnäyttö ja näytössä on 8 toiminto painiketta. Tarkkuus on 800x480 pikseliä ja näytön käyttömuisti on 10 MB. Näytössä on yksi portti profinetille sekä usb:lle. /7/



**Kuvio 5.** Siemens KTP 700 Basic. /7/

### 7.3 Ominaisuudet

Näyttö voitiin konfiguroida WinCC-ohjelmalla, joka kuuluu TIA Portal -ohjelmistoon. Näyttö sisältää 800 valmiiksi ohjelmoitua ”laatikko” -paneelia, joihin pystyttiin ohjelmoimaan käyttäjän omia toimintoja, näytössä oli myös mahdollisuus erilaisiin grafiikka kuviin ja käyrästöihin. Näyttöön saatiin ohjelmoitua tuhansia erilaisia viestejä, jotka voivat olla esim. vikoja tai hälytyksiä, myös arkistointi kuului näytön ominaisuuksiin. Usb-portin avulla näyttöön saatiin liitettyä näppäimistö tai hiiri ja lisäksi usb-tikkuun voitiin tallentaa arkistoja. /7/.

## 8 PIIRIKAAVIOT

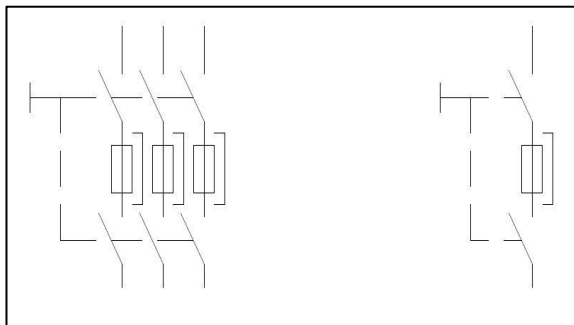
### 8.1 Yleistä piirikaavioista

Piirikaaviossa esitetään laitteen tai laiteyhdistelmän sähköiset piirit käyttäen apuna piirrosmerkkejä, numero- tai kirjainmerkintöjä, kirjallisia selityksiä sekä taulukoita. Se antaa tietoa johdotuspiirustusten ja -taulukoiden laatimiseen, koestukseen vianetsintään ja muuhun käyttöön liittyvään toimintaan. Komponentit ja merkit pyritään sijoittamaan piirustukseen siten, että ne ovat helposti tunnistettavissa ja piirikaavion ”lukeminen” on helppoa ja ymmärrettävää.

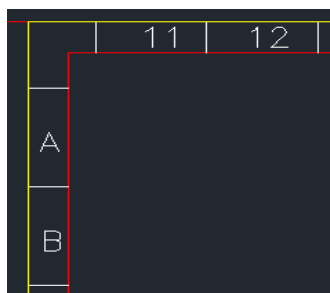
Nykyisen piirikaaviot piirretään tietokoneella, joista yleisempiä piirustusohjelmia ovat autocad ja cads. Piirtäminen tapahtuu aina liittimestä liittimeen ja mahdolliset haaroitukset liittimestä, kuten todellisuudessa, eikä keskeltä johdinta. Jos piirikaavion tulee paljon risteileviä johdotuksia, ne katkaistaan ja merkitään viitteellä selkeästi johtimen molemmat päät.

Piirikaavion laatimista voidaan helpottaa yksinkertaistamalla jo selviä piirejä ja niiden osia piirtämällä yksiviivaesityksen, jos se on riittävä kyseiseen esitykseen (**Kuvio 6.**). Jos piirin toiminnan ymmärtämisen kannalta on tarpeen esittää muissa piirustuksissa esitettyjä yhteisiä piirejä, ne voidaan esittää yksinkertaistettuna piirikaaviossa varustaen ne viitteellä ja piirtää tarkempaan esitykseen. Piirrosmerkin sijainnin voi helpoimmin ilmaista koordinaation (**Kuvio 7.**) avulla, joka on piirustuksen reunoilla. /8/





**Kuvio 6.** Moniviivaesitys vasemmalla ja yksiviivaesitys oikealla.



**Kuvio 7.** Koordinaatiomerkinnot

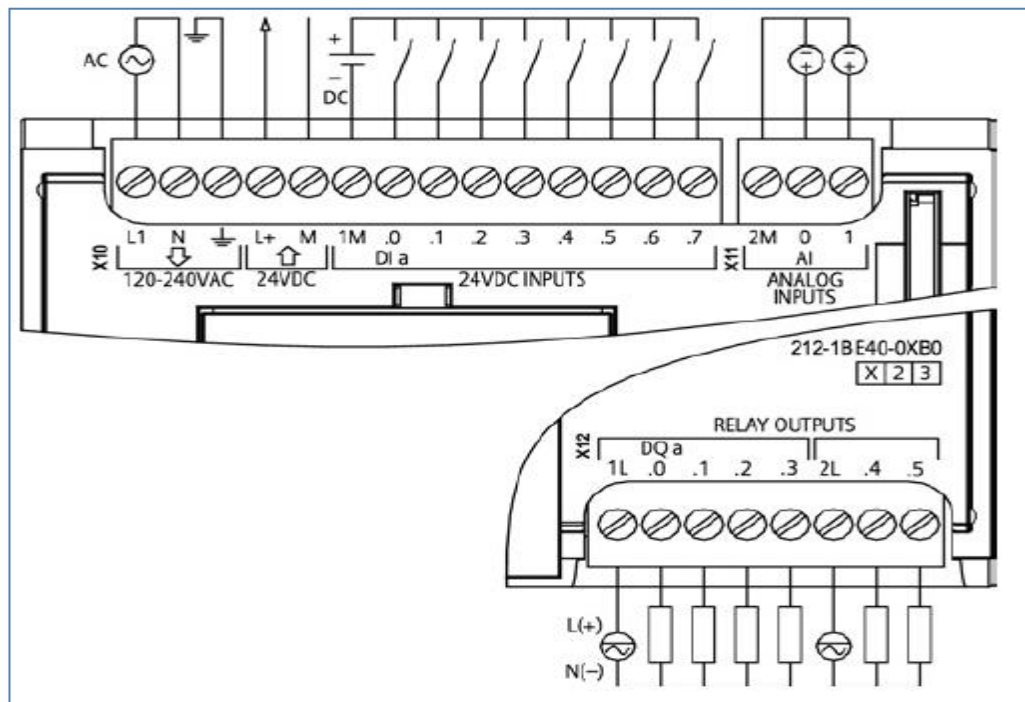
## 8.2 Piirikaavioiden piirtäminen

Piirikaavio piirrettiin autocad -ohjelmaa käyttäen. Piirikaavio piirrettiin kahdelle eri sivulle, jotta tämä selkeyttäisi ymmärtämistä logiikkaan tulevien kytkentöjen ja pääpiirikytkentöjen välille. Piirikaavioon piirrettiin suora moottorilähtö ja tälle hätä-seis-painikkeen kytkennät, pumpulle taajuusmuuttaja ohjaus ja logiikan CPU:lle venttiilin, antureiden, käyttöliittymän sekä muiden tarvittavien komponenttien kytkennät.


Suoraan moottorilähtöön piirrettiin ensimmäiseksi 400 V ja 50 Hz kolmivaihekiskosto, josta piirrettiin komponentit seuraavassa järjestyksessä johdonsuojakatkaisija F01, kontaktori K01, lämpörele B01, turvakytin Q02 ja moottori M1. Myös moottorille M1 asennettiin hätä-seis-painike katkaisemaan kaikki vaiheet

turvakytkimen Q02 jälkeen. Häätä-seis-painikkeen kytkennät on kuvattu sivun oikealle puolelle. Taajuusmuuttaja ohjatulle pumpulle M2 piirrettiin johdonsuojakatkaisija F02 ja 1-10V ohjausjännite A1 liittimeen vakiona olevalle I/O-kortille, jonka liittymät tarkistettiin Vaconin manuaalista (**Kuvio 9.**). Taajuusmuuttajalle lisättiin profinettiä varten lisäoptiokortti OPTE9, jotta taajuusmuuttajan ja CPU:n välinen kommunikointi saatiin suoritettua. (LIITE 1).

Logiikan kytkentä pisteet tarkastettiin Siemensin S7-1200 manuaalista (**Kuvio 8.**) ja kuvion avulla tarkastettiin mihin liityntäpisteisiin komponentit asennettiin. Havaittiin että logiikka tarvitsee 230 voltin käyttöjännitteen, joka liitettiin verkosta johdonsuojakatkaisijan F12 kautta logiikalle ja samasta kaapelista liitettiin logiikalle maadoitus. Digitaalituloihin DI.0 ja DI.1 lisättiin moottorille tulevat turvakytkin Q02 ja häätä-sesi-painike S11, jotta turvallisuus toiminnot saatiin välitettyä logiikalle. Moottorin kontaktori K2 käy-stop -toiminta asetettiin digitaalilähtöön DQ.0. Paineanturi PT1 ja virtausanturi FT1 kytkettiin logiikan analogia tuloihin, jotka sijaitsivat CPU:n riviliittimessä X11. Lisäksi kuvaan piirrettiin muuntaja T10 ja tälle johdonsuojakatkaisija F11, jotta antureille saataisiin 24 voltin käyttöjännite. Solenoidiventtiilin ohjaus piirrettiin DO.5:een. Käyttöliittymän jännite saatiin CPU:n 24 voltin syötöstä ja tämä kytkettiin taajuusmuuttajaan profinetkaapelilla (LIITE 2).



**Kuvio 8.** Logiikan kytkentäpisteet /9/



Liitin	Signaali	Tehdasasetus	Kuvaus
1	+10Vref	Ohjearvojännite	Maksimikuorma 10 mA
2	AI1	Analogiatulo 1	Taajuusohje <sup>P1</sup>
3	GND	I/O-signaali maa	
6	24Vout	24V lähtö DI:lle	±20 %, maks. k. 50 mA
7	GND	I/O-signaali maa	
8	DI1	Digitaalitulo 1	Käy eteen <sup>P1</sup>
9	DI2	Digitaalitulo 2	Käy taakse <sup>P1</sup>
10	DI3	Digitaalitulo 3	Vian kuittaus <sup>P1</sup>
A	A	RS485 signaali A	Kenttäv.yhteys
B	B	RS485, signaali B	Kenttäv.yhteys
4	AI2	Analogiatulo 2	PI hetkellisarvo <sup>P1</sup>
5	GND	I/O-signaali maa	
13	GND	I/O-signaali maa	
14	DI4	Digitaalitulo 4	Esiasetusnopeus B0 <sup>P1</sup>
15	DI5	Digitaalitulo 5	Esiasetettu nopeus B1 <sup>P1</sup>
16	DI6	Digitaalitulo 6	Ulkoisen vika <sup>P1</sup>
18	AO	Analogialähtö	Lähtötaajuus <sup>P1</sup>
20	DO	Digitaalisignaali lähtö	Aktiivinen = VAL-MIS <sup>P1</sup>
22	RO1 NO	Relelähtö 1	Aktiivinen = KÄY <sup>P1</sup>
23	RO1 CM		
24	RO2 NC	Relelähtö 2	Aktiivinen = VIKKA <sup>P1</sup>
25	RO2 CM		
26	RO2 NO		

**Kuvio 9.** Taajuusmuuttajan I/O-kortin kytkentäpisteet. /10/

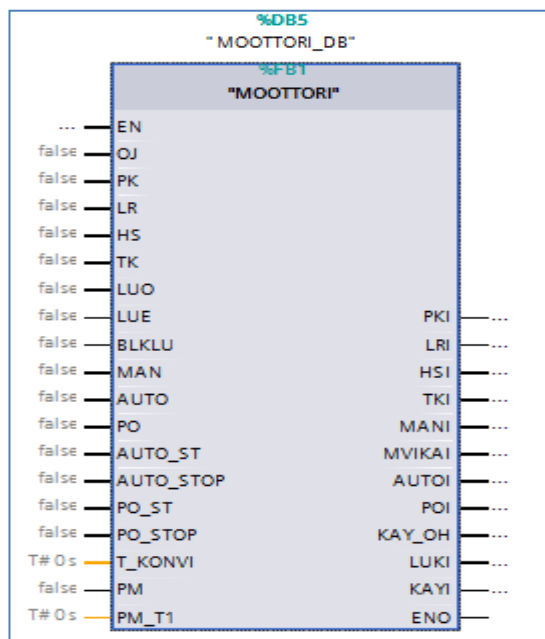
## 9 OHJELMOINTI JA KONFIGURAATIO

### 9.1 Yleistietoa toimilohkojen luomisesta

Ohjelmoitavan logiikan peruskäskyinä ovat loogiset porttipiirit. Näistä rakennetaan toimilaiteohjaukset eli oliot kuten moottorilähdöt, venttiiliohjaukset jne. Olioiden käytöstä on hyötyä sillä ohjauskokonaisuudet on testattu toimiviksi. Kaikki toimilaiteohjaukset toimivat samalla tavoin ohjelmoijasta riippumatta. Muutokset ovat helppoja, koska yleensä vain muutetaan I/O-signaaleja tai parametrien muutoksilla, jolloin ydintoimintoja ei tarvitse muuttaa. Valvomo toiminnot voidaan rakentaa vakioksi, jolloin valvomon ohjelmointi jää vähäiseksi. Haittapuolia ovat käyttämättömät toiminnot, koska näille varatut apumuistipaikat jäävät käyttämättä ja ohjelman kiertoaika pitenee, koska oliossa suoritetaan myös ylimääräiset toiminnot. Kiertoaikasta aiheutuva haitta on vain 1 sekunnin luokkaa, joten tästä ei suuresti ole näkyvää haittaa laitteiston toiminnassa. /11/

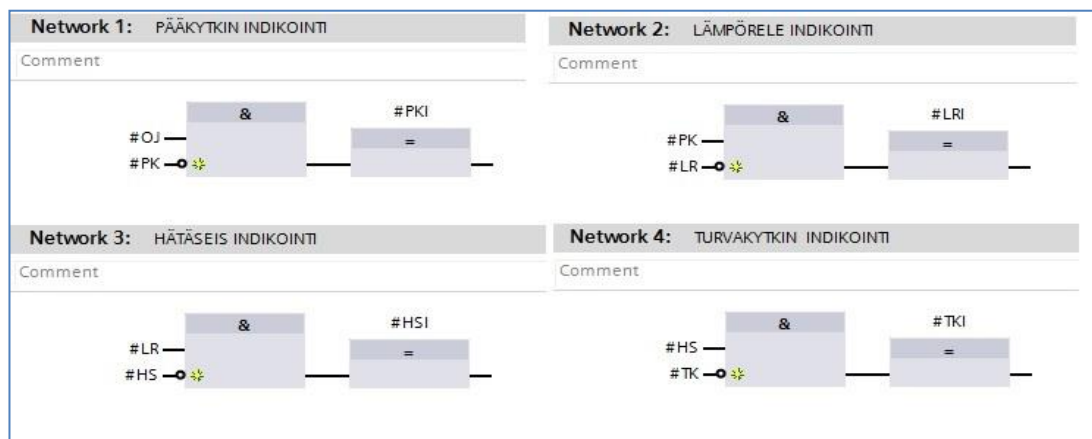
#### 9.1.1 Moottoripiiri

Ohjelman luominen aloitettiin tekemällä uusi projekti ja tälle valittiin PLC:ksi hankkimamme CPU. Ohjelmassa luotiin toimilaitteille sekä antureille omat toimilohkot FB-lohkoihin (Function Block). Ensimmäiseksi luotiin moottoripiirin toiminta FB-lohkoon. (**Kuvio 10.**)



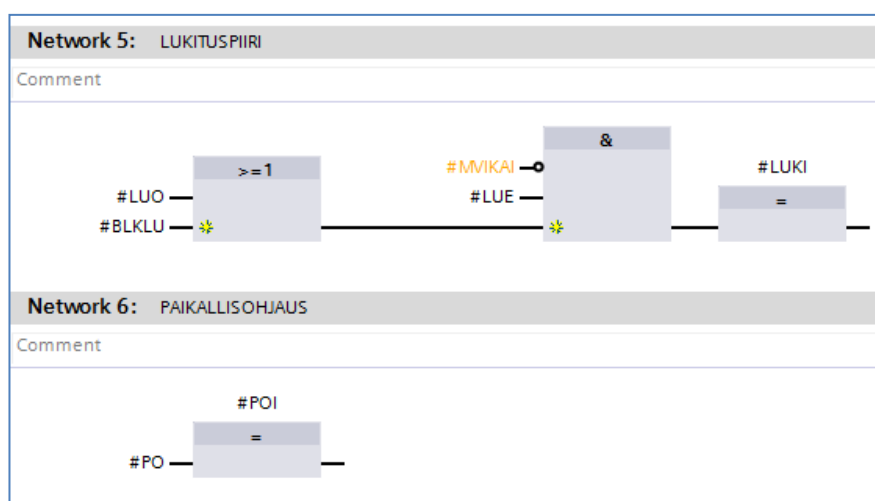
**Kuvio 10.** Moottoripiirin toimilohko

Moottorille tulevien pääkytkimen, lämpöreleen, hätä-seis-painikkeen ja turvakyt-  
kimen indikoinnit toteutettiin networkeissa 1 – 4. Näillä toiminnoilla estettiin  
moottorin käyttö jos jokin toteutui. Esimerkiksi jos hätä-seis-painiketta ei oltu  
painettu ja turvakytin oli 0-tilassa, niin turvakytimen indikointi (TKI) toteutui.  
Tällä toiminnolla annettiin vika- ja stop-käsäy moottorille network 17:sta. (**Kuvio  
11**).



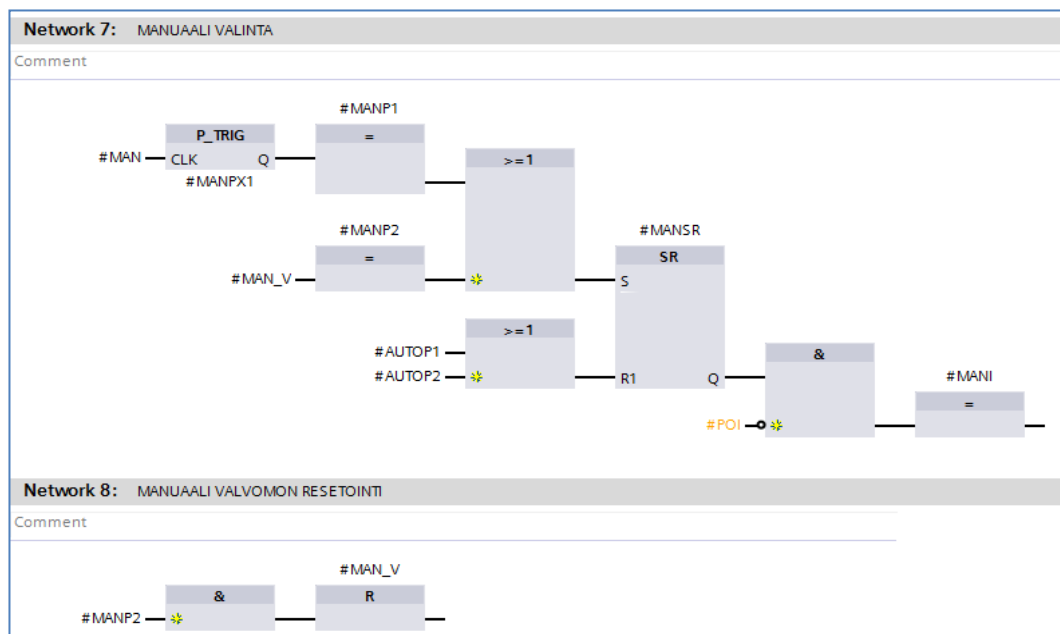
**Kuvio 11.** Apukoskettimien indikointi

Lukituspiiri toteuttiin network 5:ssä, siinä lukitus saa indikoinnin, kun moottorissa ei ole vikaa ja ehdoton lukitus sekä jompikumpi tai molemmat ohitettavista lukituksista ovat voimassa. Lukituspiirillä estettiin laitteen tahaton käyttö. Paikallisohjauksen indikointi tulee suoraan paikallisohjauksen käskystä, se toteutettiin network 6:ssä. (**Kuvio 12.**)



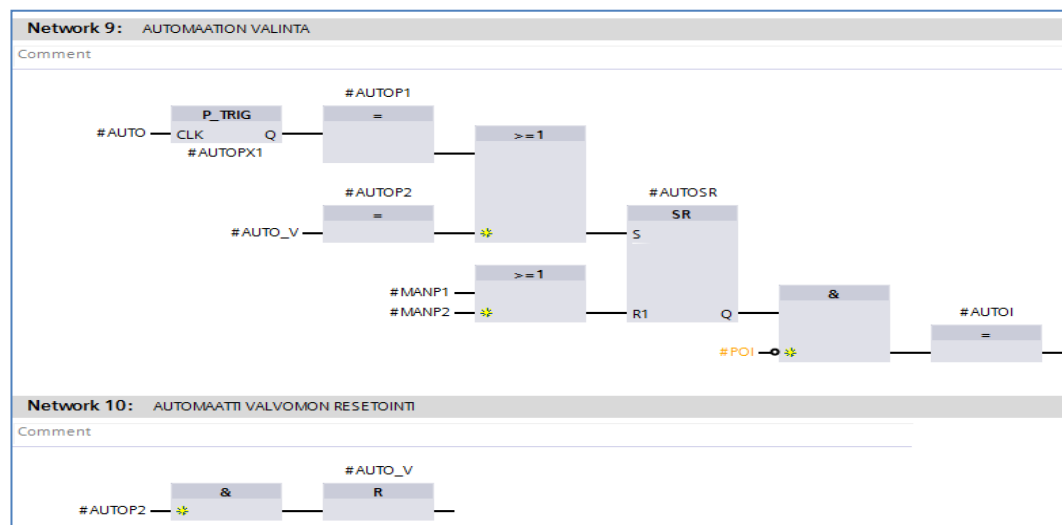
**Kuvio 12.** Lukituspiirin ja paikallisohjauksen toteutus

Network 7 – 10:ssä luotiin manuaali- ja automaatiotoimintojen valinnat. Valinnat toimivat täysin samalla tavalla, mutta manuaalin valinta tapahtui manuaalikomennoilla ja vastaavasti automaatin valinta automaattikomennoilla. Toisen valinnalla nollattiin toisen valinta ja paikallishjauksen indikointi täytyi olla ”epätosi”, jotta ehdot täyttyisivät. Molempien valvomotilojen eli käyttöliittymä valinnan resetoitiin sitä vastaavassa pulssissa, jotka ovat networkeissä 8 ja 10. Manuaalitalan valinta ja valvomokäskyn on kuvattu kuvioon 13 ja vastaava automaattitalan valinta on kuviossa 14.



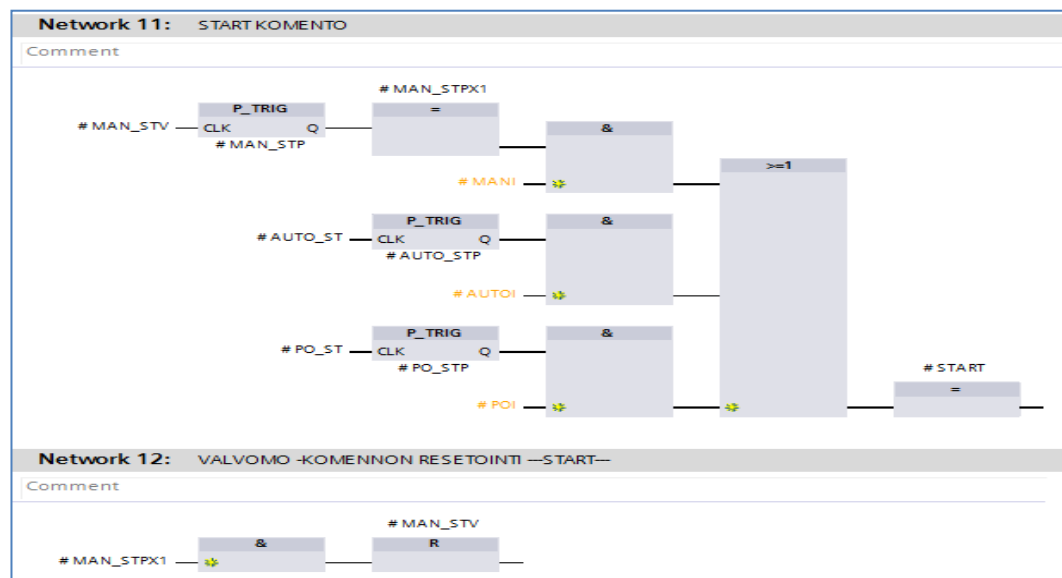
**Kuvio 13.** Manuaalitalan valinta ja manuaalitalan valvomotilan resetointi.





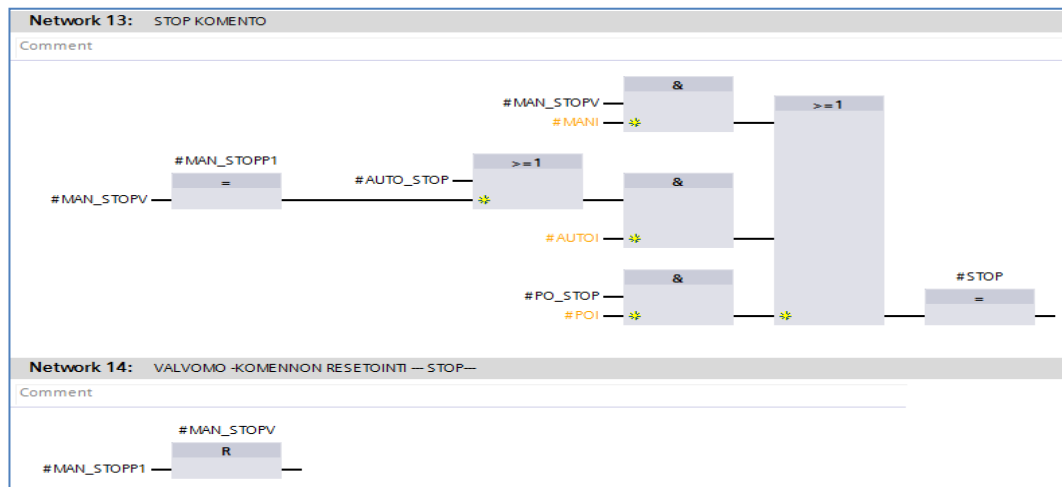
**Kuvio 14.** Automaattitilan valinta ja automaattitilan valvomotilan resetointi.

Moottorin start-komento annettiin manuaalisesti, automaattisesti tai paikallisohjauksen avulla ja jokaisen käytön kohdalla tuli olla vastaava indikointi voimassa. Manuaalitilan valvomon start-komento resetoitiin network 12:sta (**Kuvio 15**).



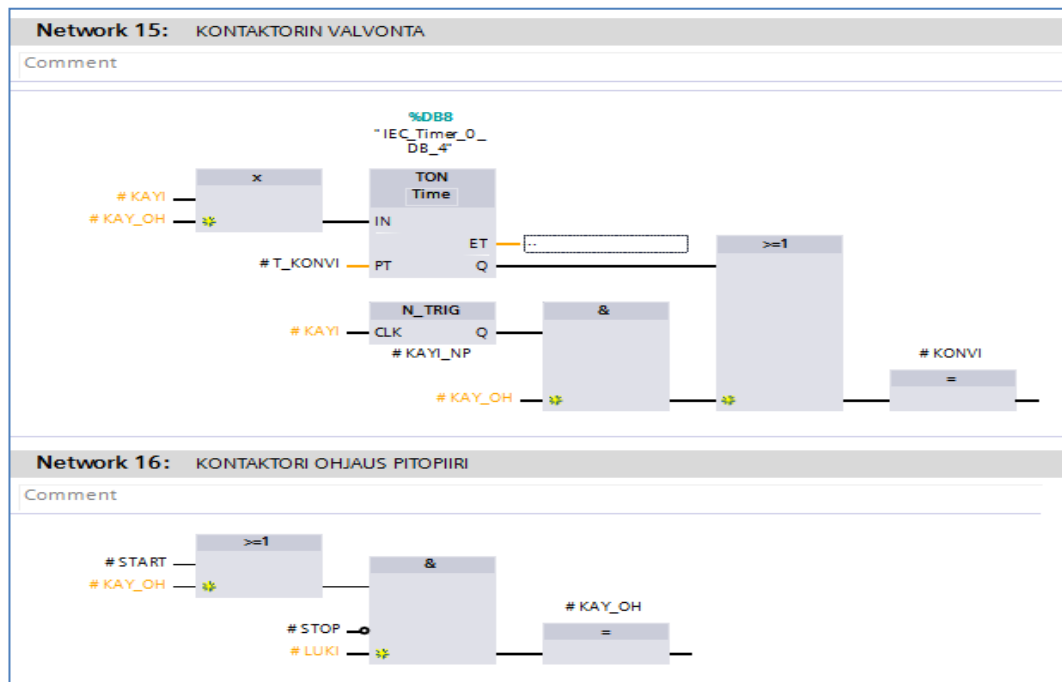
**Kuvio 15.** Moottorin start-komento ja manuaali ohjauksen valvomotilan resetointi.

Network 13:ssa moottori sai stop-komennon joka oli samankaltainen kuin start-komento, mutta moottori saatiin myös pysäytettyä manuaalitulassa vaikka automaattitila oli valittu. Manuaalin valvomotila resetoitiin kuten aikaisemminkin (**Kuvio 16.**).



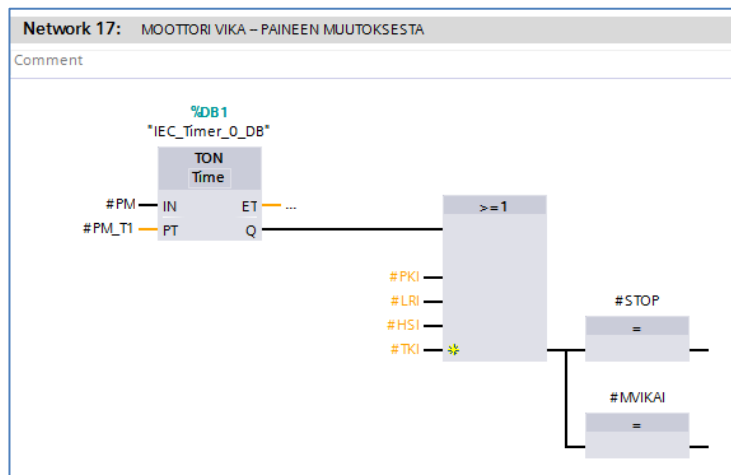
**Kuvio 16.** Moottorin stop – komento ja manuaalisen valvomotilan resetointi.

Network 15:ssa toteutettiin kontaktorin valvonta, jossa kontaktorin vikailmoitus sai arvon jos käy indikoinnin negatiivinen pulssi ja moottori käy-ohjearvo toteutuivat. Vika toteutui vain jos jompikumpi käy-indikoinnista tai käy-ohjearvosta toteutui. Network 16:ssa ohjelmoitiin kontaktorille pitopiiri, jossa start-käskyllä käynnistettiin piiri ja käy-käskyn ohjearvolla ylläpidettiin piiri tosi-tilassa. Pitopiiri myös vaati sen että lukitukset olivat kunnossa ja stop-komento ei ollut voimassa (**Kuvio 17.**).



**Kuvio 17.** Moottorin kontaktorin valvonta ja ohjauksen pitopiiri.

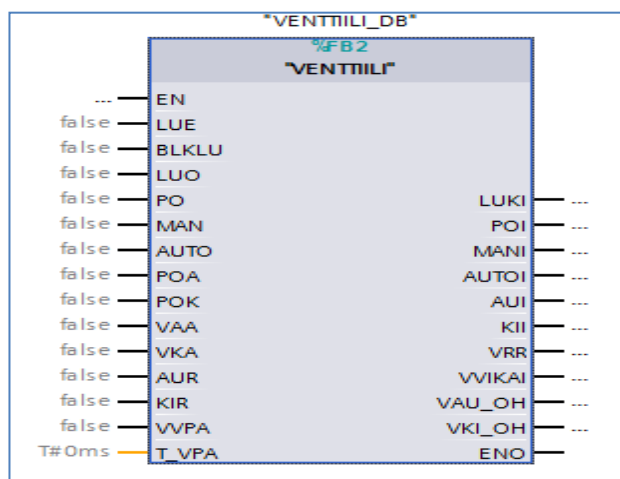
Network 17:ssa moottorille luotiin vian antava toiminto. Toiminnossa moottorille asetettiin vika komento ja toiminta keskeytettiin kun paine muuttui asetetun ajan aikana tai networkeissä 1 – 4 (**Kuvio 11.**) jokin komento tuli toteen (**Kuvio 18.**).



**Kuvio 18.** Moottorin vika paineen muutoksesta.

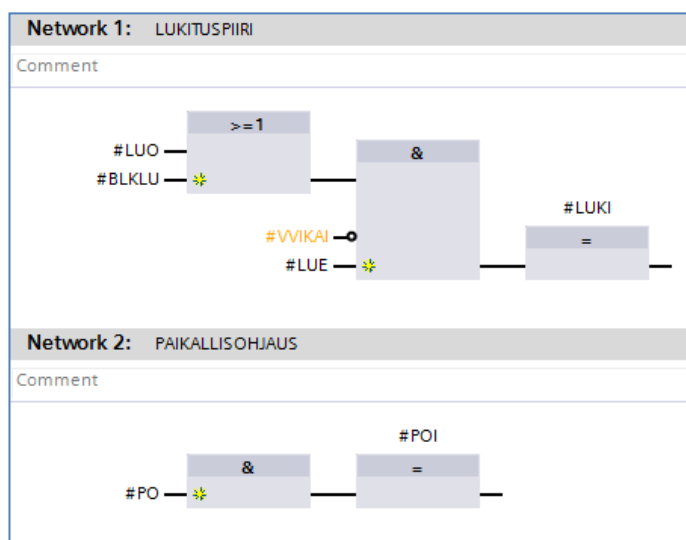
### 9.1.2 Venttiilipiiri

Venttiilipiiri luotiin moottoripiirin toimintojen pohjalta, mutta toimintoja muutettiin venttiilin ominaisuuksien mukaan. Logiikasta nähtynä venttiili näytti kuvion 19 mukaiselta.



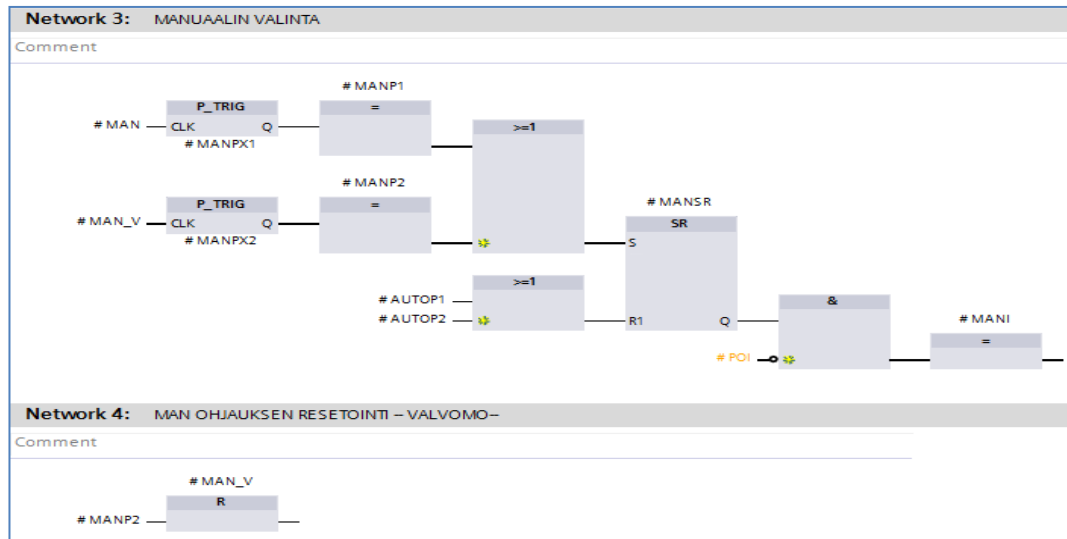
**Kuvio 19.** Venttiilipiiri

Networkeissa 1 ja 2 toteutettiin venttiilin lukituspiiri ja paikallishjauksen toiminto noudattivat samaa kaavaa kuin moottoripiirissä. Myös venttiilin tapauksessa lukituspiirillä estettiin laitteen tahaton käyttö (**Kuvio 20.**).

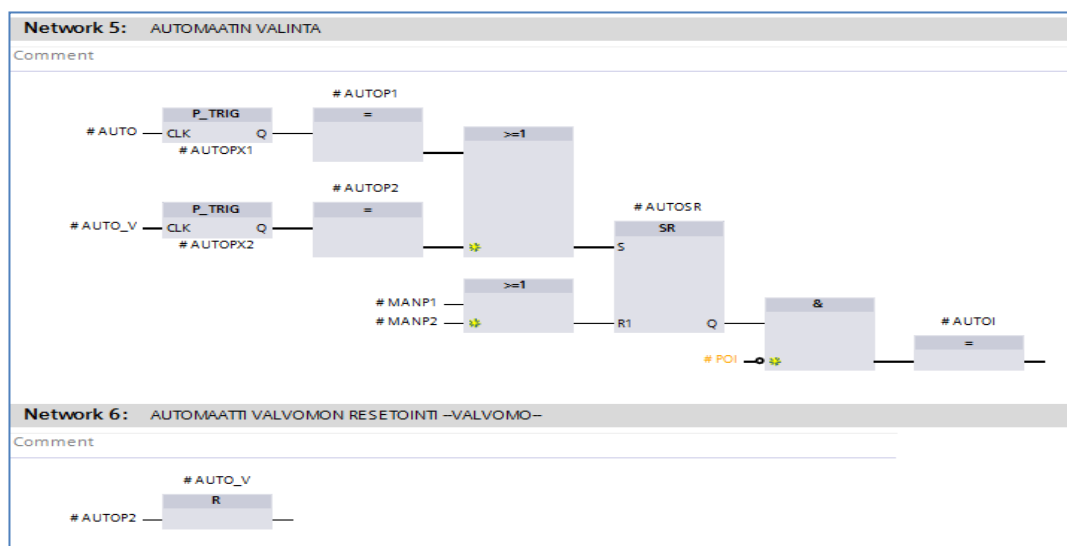


**Kuvio 20.** Venttiilin lukituspiiri ja paikallishjauksen valinta.

Venttiilipiirin manuaalin ja automaattitilojen indikoinnit olivat samanlaisia kuin moottoripiirissä. Manuaalin valinta tapahtui manuaalitalan valinnalla tai valvomotilan komennolla ja piirin nollautui automaattitalan valinnalla ja vastaavasti automaattitala nollautui manuaalitalan valinnalla. Molemmissa tuli olla paikallishjaus epätosi-tilassa, jotta toiminto saatiin suoritettua. Valvomotilojen resetointi toteutettiin samalla tavalla kuin moottoripiirissä (**Kuviot 21. - 22.**).

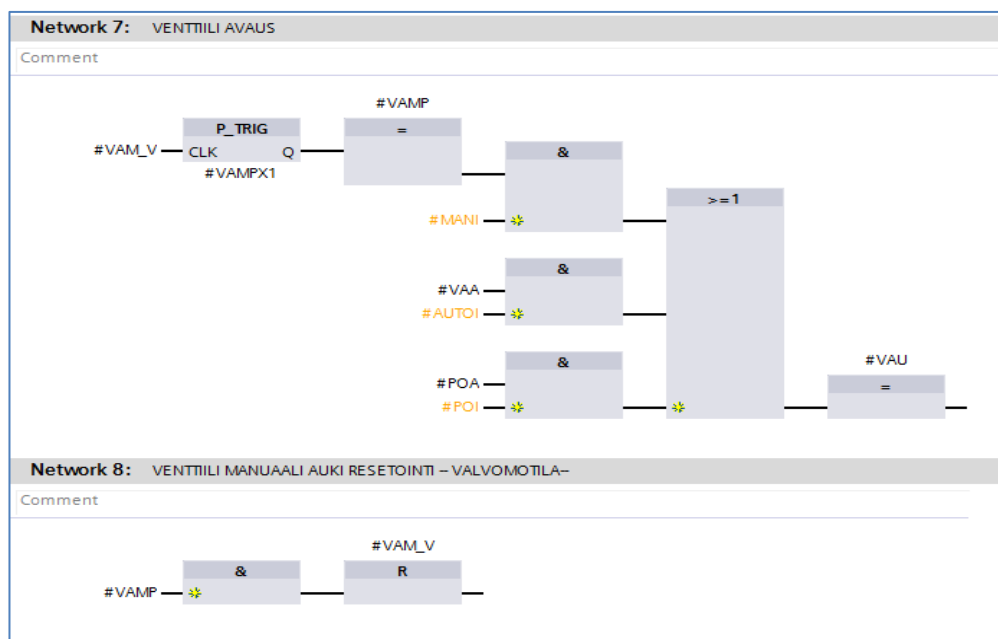


**Kuvio 21.** Venttiilin manuaalitilan valinta ja valvomo-ohjauksen resetointi.



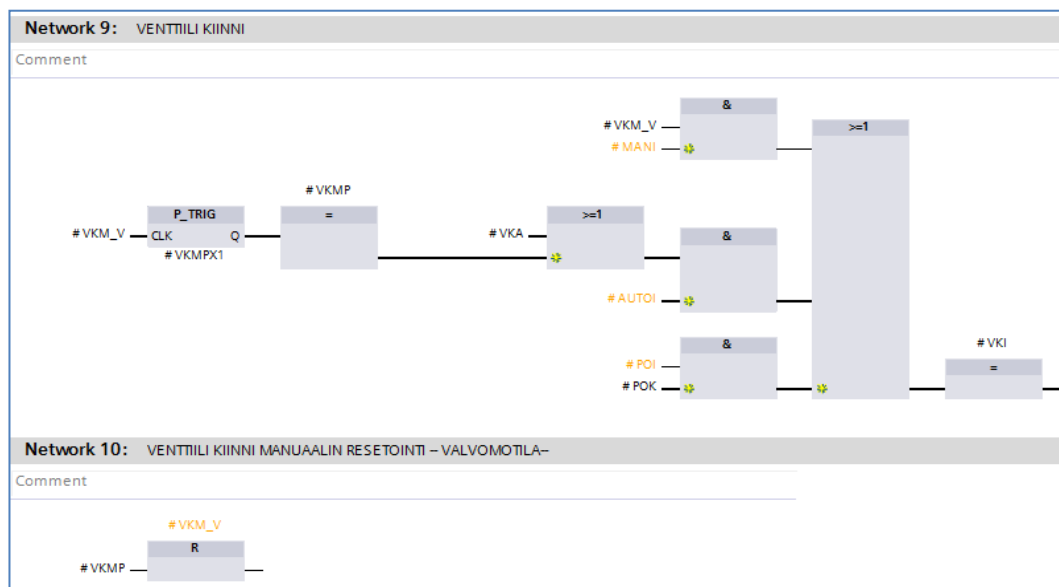
**Kuvio 22.** Venttiilin automaattitilan valinta ja valvomo-ohjauksen resetointi.

Venttiilin avaaminen toteutettiin network 7:ssä. Venttiilin avaaminen tapahtui samanlaisilla komennoilla kuin moottorin käynnistäminen eli vastaavan avauskomenton kanssa tuli olla valitun tilan indikointi voimassa. Jälleen manuaali valvomotilan resetointi toteutettiin seuraavassa networkissa eli tässä tapauksessa network 8:ssa. Venttiilin kiinni toiminta eli sulkeminen ja valvomo komennon resetointi luotiin networkeissa 9 ja 10. Toiminnot olivat samanlaisia kuin avaamisessa, mutta venttiilin ollessa automaatti tilassa se voitiin myös sulkea manuaalisti (Kuviot 23. - 24.).



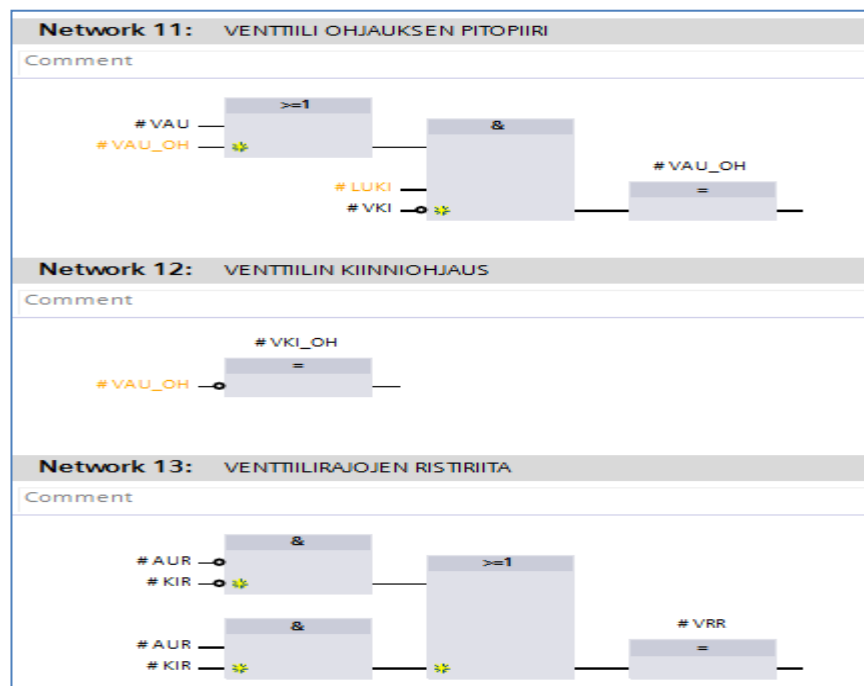
**Kuvio 23.** Venttiilin avaaminen ja valvomotilan resetointi.





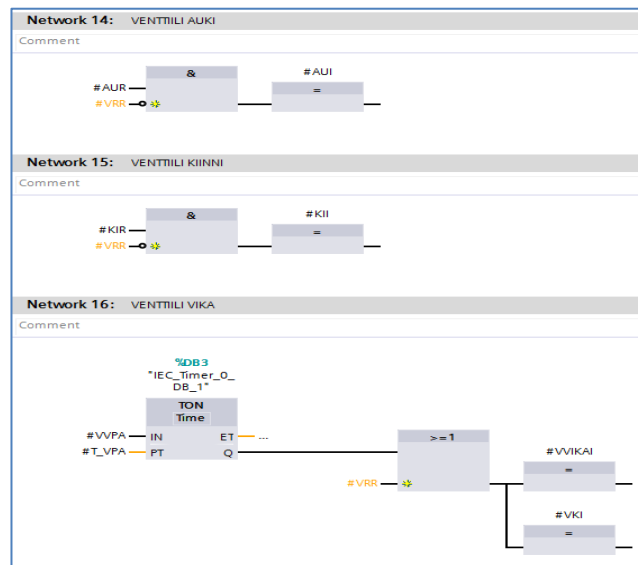
**Kuvio 24.** Venttiilin kiinni toiminto ja valvomotilan resetointi.

Network 11:ssä toteutettiin venttiilin aukiohjauksen pitopiiri joka toimi samalla periaatteella kuin moottorin käy-ohjauksen pitopiiri. Käsky annettiin auki komenolla jonka jälkeen venttiili auki ohjaus-komento piti voimassa piirin toteutuksen. Toiminnan edellytyksiin kuului, että lukitusten tuli olla kunnossa ja venttiili ei ole saanut kiinni käskyä. Network 12:ssä venttiilin kiinni ohjaus toteutettiin auki ohjauksen negatiolla. Network 13:sta luotiin venttiilin auki ja kiinni rajojen ristiriita, jolla oli tarkoitus ilmoittaa tilasta, että venttiili ei voi olla auki ja kiinni yhtäaikaan (**Kuvio 25.**).



**Kuvio 25.** Venttiilin ohjauksen pitopiiri, kiinniohjaus ja rajojen ristiriita.

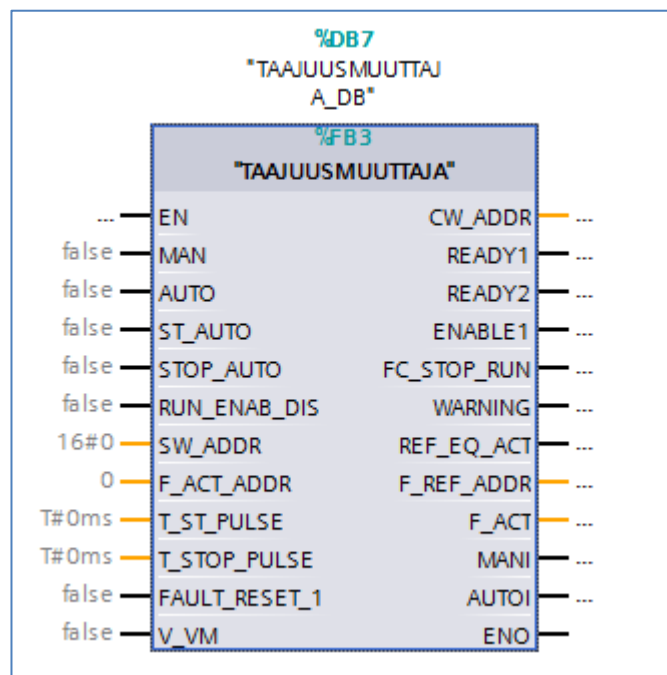
Venttiilin auki ja kiinni indikoinnit toteutettiin networkissa 14 ja 15. Auki indikoinnissa tuli olla auki raja tosi-tilassa ja rajojen ristiriita tuli olla epätosi-tilassa, jotta indikointi toteutuisi. Vastaavasti kiinni indikoinnissa tuli olla kiinni raja voimassa ja rajojen ristiriita epätosi-tilassa. Network 16:sta luotiin venttiilin vika ilmoitus paineen muuttuessa tietyssä ajassa tai venttiilin rajojen ristiriita toteutui. Toiminto ilmoitti venttiilin vian ja venttiili meni kiinni tilaan (**Kuvio 26**).



**Kuvio 26.** Venttiilin auki ja kiinni indikoinnit sekä vika toiminto.

### 9.1.3 Taajuusmuuttajapiiri

Taajuusmuuttajan ohjaus tehtiin huomattavasti erilaisemmaksi kuin moottorin tai venttiilin toiminta. Ainoastaan ohjauspaikan valinta, kuten automaatin tai manuaalin ja start- ja stop-signaali olivat samantyyllisiä käskyjä kuin venttiilissä ja moottorissa. Taajuusmuuttajan lohko on kuvattu kuvioon 27.



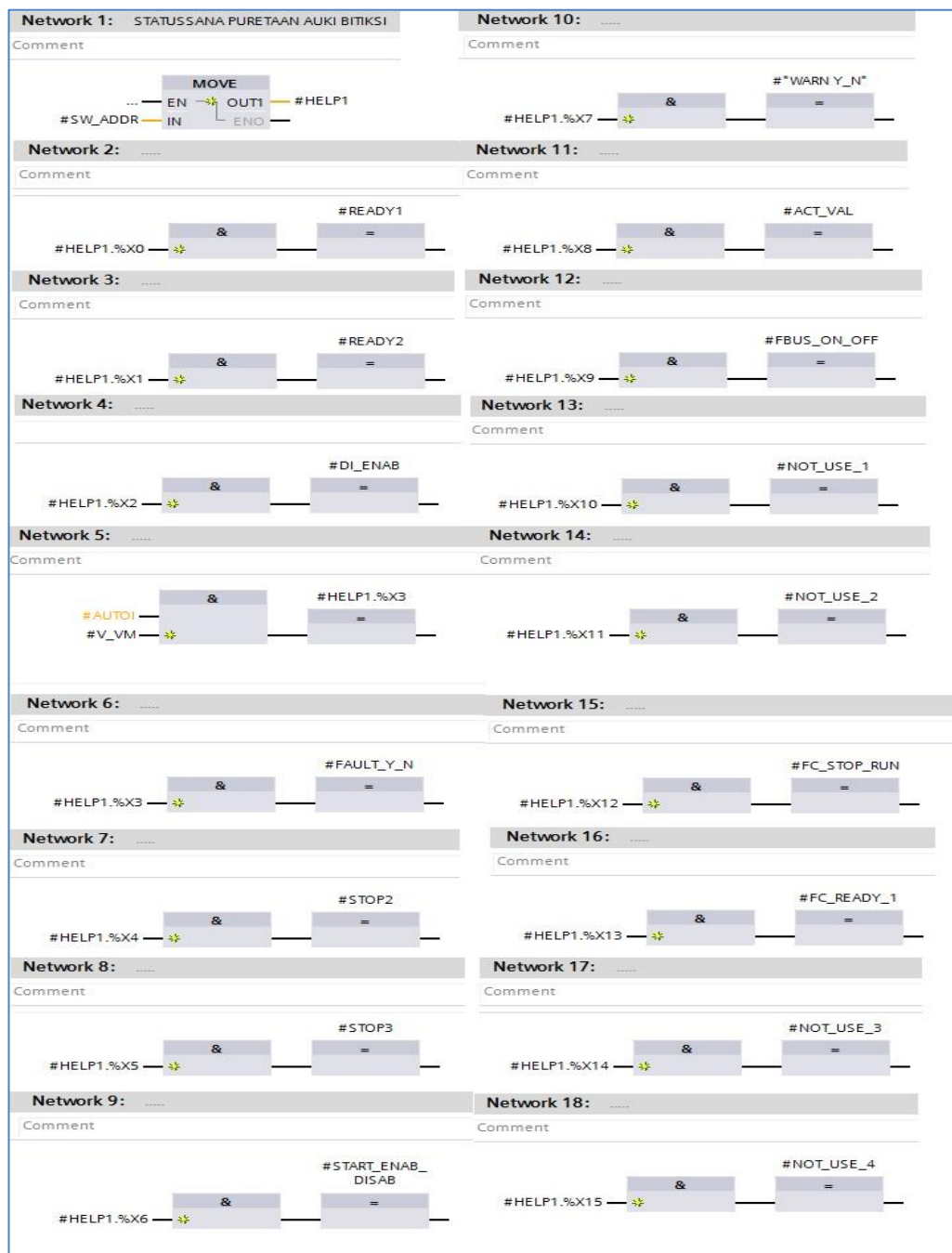
**Kuvio 27.** Taajuusmuuttajan lohko.

Ennen kuin aloitettiin luomaan taajuusmuuttajalohkoa, täytyi vaconin manuaalista tarkistaa mitä bittejä haluttiin ohjata ja lukea. Manuaalista havaitsimme mitä toimintoa ohjattiin tai tilaa luettiin kunkin bitin kohdalla. Lisäksi havaittiin mitä kuvaukset tarkoittivat arvoilla 0 ja 1 (**Kuvio 28.**). /12/

Bit	Description		Bit	Description	
	Value = 0	Value = 1		Value = 0	Value = 1
0	Not Ready (initial)	READY 1 **	0	STOP 1 (by ramp)	ON 1
1	Not Ready	READY 2 **	1	STOP 2 (by coast)	ON 2
2	DISABLE	ENABLE **	2	STOP 3 (by ramp)	ON 3
3	NO FAULT	FAULT ACTIVE *	3	RUN DISABLE	ENABLE
4	STOP 2	NO STOP 2 **	4	No action	START
5	STOP 3	NO STOP 3 **	5	No action	START
6	START ENABLE	START DISABLE **	6	No action	START
7	No Warning	Warning *	7	No action	FAULT RESET (0 ->1)
8	Reference ≠ Actual value	Reference = Actual value *	8	No action	No action
9	Fieldbus control OFF	Fieldbus control ON *	9	No action	No action
10	Not used	Not used	10	Disable fieldbus control	Enable fieldbus control
11	Not used	Not used	11	Fieldbus DIN1=OFF	Fieldbus DIN1=ON
12	FC stopped	Running *	12	Fieldbus DIN2=OFF	Fieldbus DIN2=ON
13	FC not ready	FC ready *	13	Fieldbus DIN3=OFF	Fieldbus DIN3=ON
14	Not used	Not used	14	Fieldbus DIN4=OFF	Fieldbus DIN4=ON
15	Not used	Not used	15	Fieldbus DIN5=OFF	Fieldbus DIN5=ON

**Kuvio 28.** Vasemmanpuoleiset bitit ovat taajuusmuuttajan status wordiä ja control wordit oikeanpuoleisia bittejä.

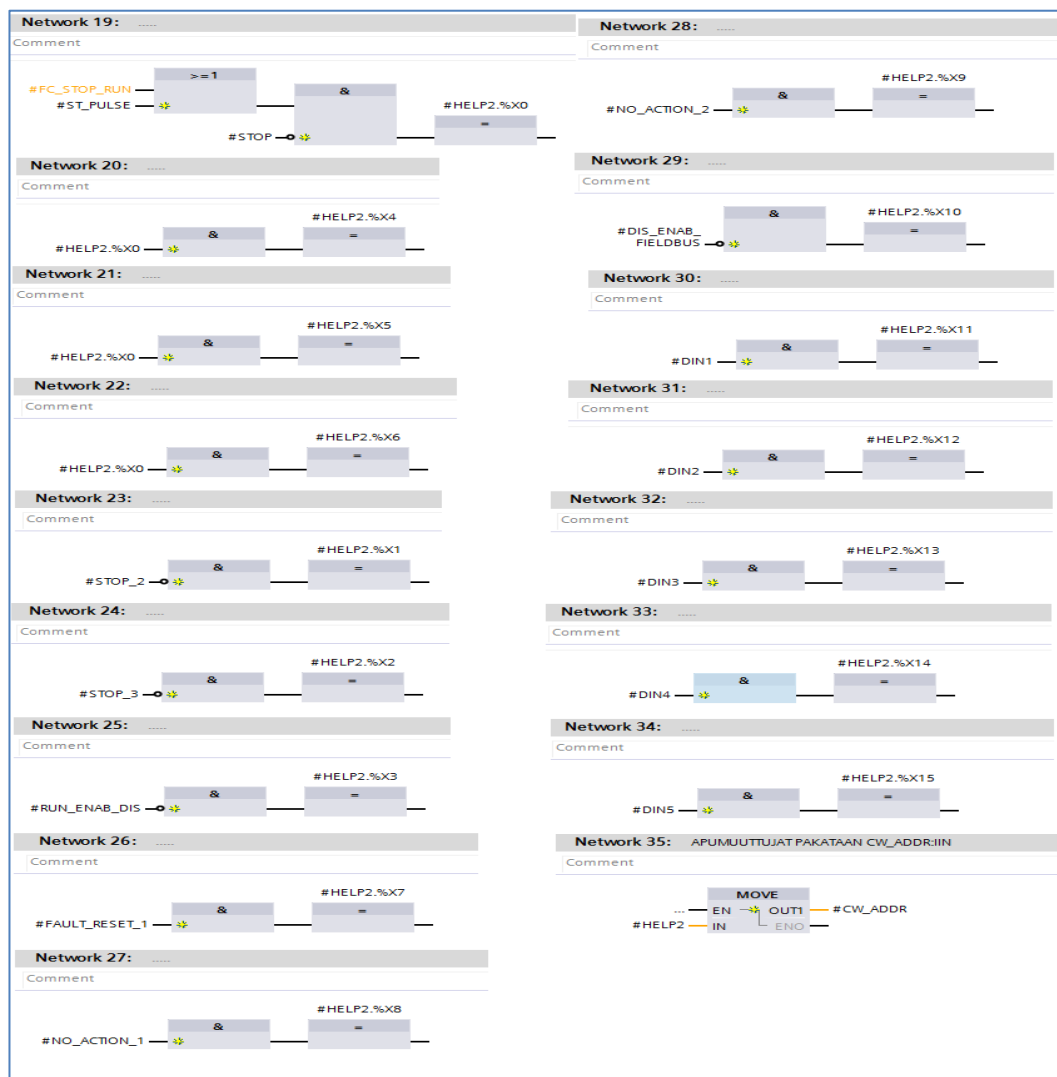
Taajuusmuuttajalta tulevan status word purettiin biteiksi apumuuttujan help1 avulla. Jokaiselle bitille luotiin oman network, jotta tämä helpottaisi bittien toimintojen seuraamista ja mahdollista muokkaamista. Toimintojen siirto oli helppoa, esimerkiksi taajuusmuuttajan bittiin 0 viitattiin #help1.%x0 komennolla. Network 5:ssa luotiin toiminto jossa taajuusmuuttaja saa fault\_y\_n ilmoituksen eli vika tilan, kun automaatti tila oli valittuna ja tila v\_vm eli vika virtauksen muutoksesta oli voimassa. Tilojen toiminnot vastasivat kuvion 28 status wordien kuvauksia. **(Kuvio 29.)**



**Kuvio 29.** Sw\_addr -muuttujan purkaminen biteiksi.

Network:ssä 19. – 35. taajuusmuuttajan ohjaus käskyt pakattiin apumuuttujan help2 avulla muuttujaan cw\_addr. Toiminnot vastasivat kuvion 28 control word

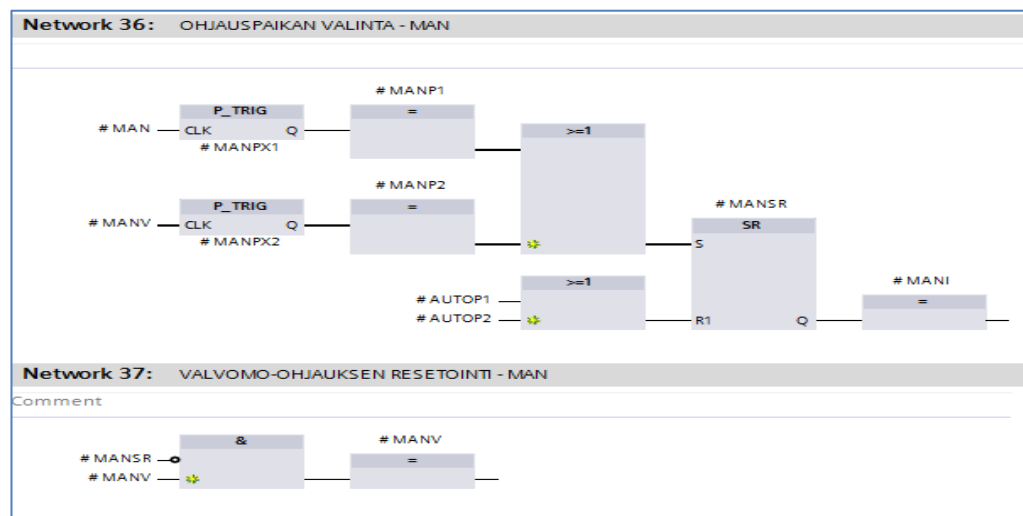
komentoja. Jokaiselle bitille luotiin oman network kuten sw\_addr:n tapauksessa ja taajuusmuuttajan toimintojen bitteihin viitattiin komennoilla #help2.%x. Network 19 luotiin pitopiiri käy –toiminnolle, joka sais toiminto käskyn taajuusmuuttajan ollessa käytilassa (FC\_STOP\_RUN) tai start-pulssista kun stop-tila ei ollut voimassa. Tämän toiminnon toteutuessa ohjattiin myös bittejä 4 – 6 (**Kuvio 30.**).



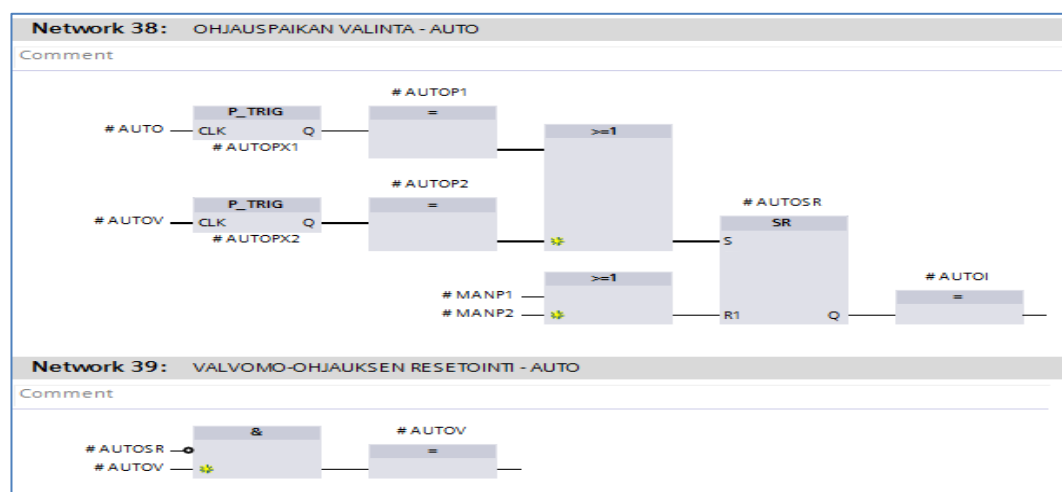
**Kuvio 30.** Bittien pakkaaminen cw\_addr -muuttujaan.

Manuaali- ja automaattitilan valinta toimii samalla periaatteella kuin moottori- ja venttiilipiireissä. Molemmissa tapauksissa pystyi valitsemaan valvomotilan ja toiminnot nollasivat toisensa kun valittiin toinen tila. Valvomo-ohjauksen resetointi toteutettiin hieman erillä tavalla, mutta tulokseksi saatiin samanlainen toiminta kuin moottori- ja venttiilipiireissä (**Kuvio 30. - 31.**)





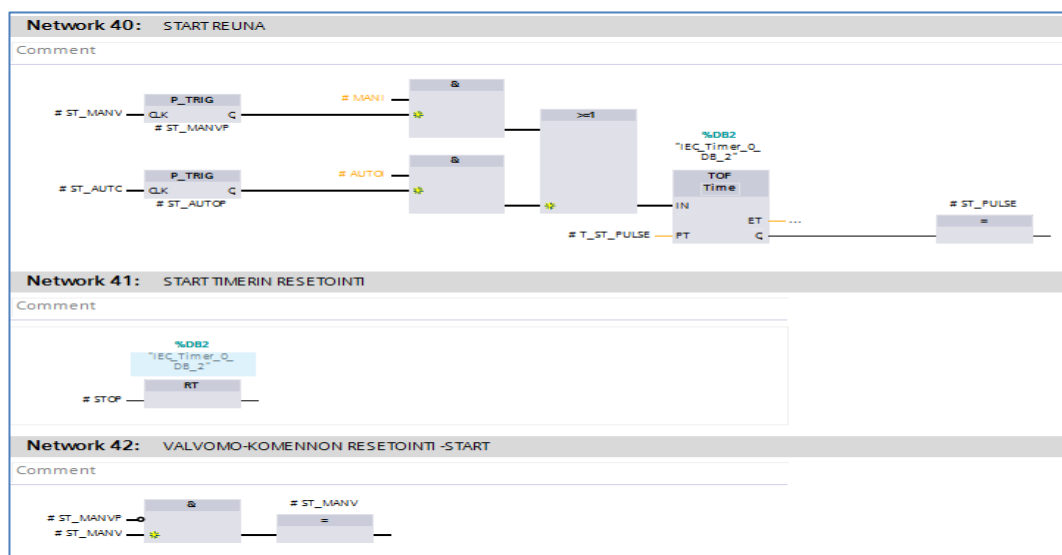
**Kuvio 31.** Manuaalitilan valinta.



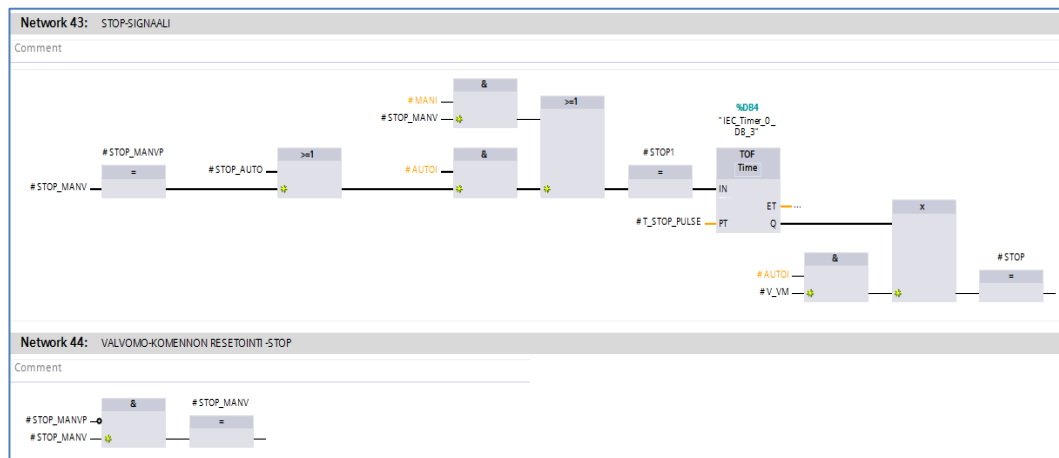
**Kuvio 32.** Automaattitilan valinta.

Network 40 luotiin start-reuna taajuusmuuttajalle. Start-käskey annettiin joko manuaalisesti tai automaattisesti ja molemmissa piti olla vastaava indikointi voimassa, että toiminto hyväksyttäisiin. Start-käskyn manuaalitilan valvomo-ohjaus rese-

toitiin samalla tavalla kuin muut valvomo-ohjaukset taajuusmuuttajapiirissä. Networkissä 41 start-piirin timer DB2 resetointi tapahtui stop-komennolla. Tällä toiminnolla saatiin pysäytettyä start-käskeä, jos havaittiin ongelma kohta ennen kuin aika oli asetetussa ajassa. Stop-reunan toiminta toteutettiin samalla tavalla kuin start, mutta automaattitilassa voitiin myös antaa manuaalisesti stop-käskeä. Lisäksi stop-käskeä voitiin toteuttaa, kun auto-indikointi ja V\_VM eli vika-käskeä olivat voimassa. Jälleen valvomo-ohjauksen resetointi tapahtui vastaavalla tavalla, kuin muut resetoinnit (**Kuvio 33. ja 34.**).

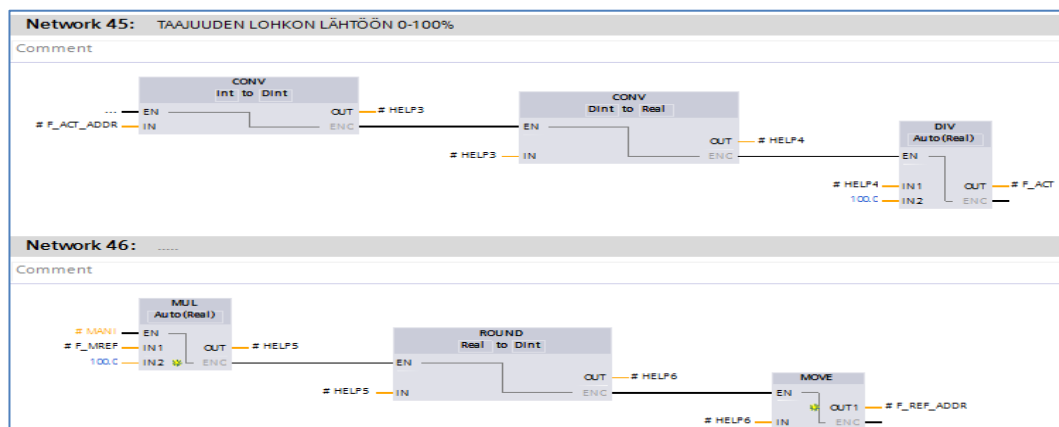


**Kuvio 33.** Start –reunan toteutus ja valvomotilan resetointi.



**Kuvio 34.** Stop –reuna ja valvomotilan resetointi.

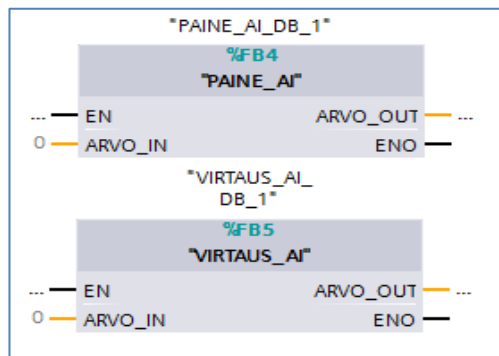
Network 45:ssä ohjattiin taajuuden arvo lähtöön välille 0 – 100 %. Network 46:ssa annettiin taajuusohje 0 – 100 % F\_MREF kohtaan, joka kirjoitettiin käyttöliittymästä ja väylälle ohjeeksi 0 – 10 000 ja tämä arvo kirjoitettaisiin lähtöön F\_REF\_ADDR. Koska vastaava automaattitilaa ei tultu tekemään, niin taajuusohje annettiin vain manuaalasti (**Kuvio 35**).



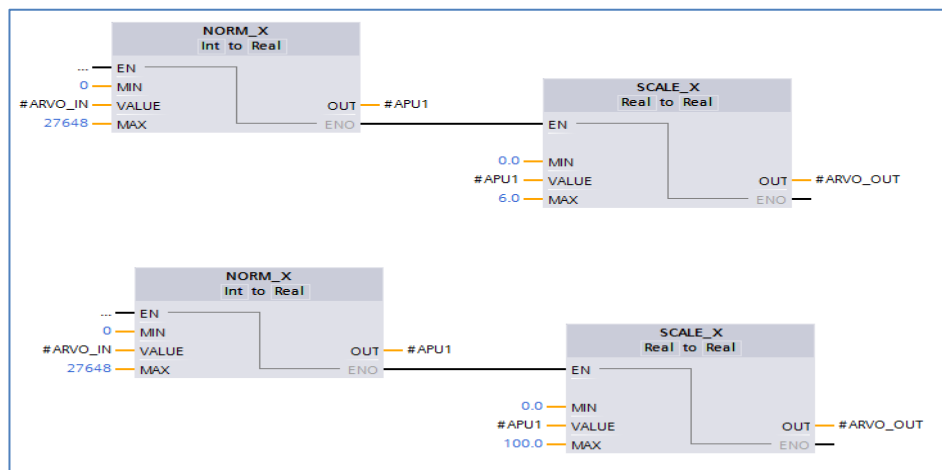
**Kuvio 35.** Taajuuden asetus lohkon ja taajuusohje manuaalitulassa.

#### 9.1.4 Analogiatulosten muuttaminen

Analogiatulot paine- ja virtausanturilta muutettiin reaalityypiksi fb –lohkoissa. Standardiviesti 0 – 27 648 asetettiin näyttämään painearvoa siten, että 0 tarkoitti 0 bar:a ja 27 648 6 bar:a. Vastaavasti virtaus asetettiin että 0 tarkoitti 0 l/min ja 27 648 tarkoitti 100 l/min. Muuttujassa annettiin arvo 0 – 27 648 int -datatyypinä ja tämä muutettiin reaalityypiksi asetetulle välille 0 – 6 ja 0 – 100 (**Kuviot 36. - 37.**).



**Kuvio 36.** Paine- ja virtausanturit fb –lohkoina.

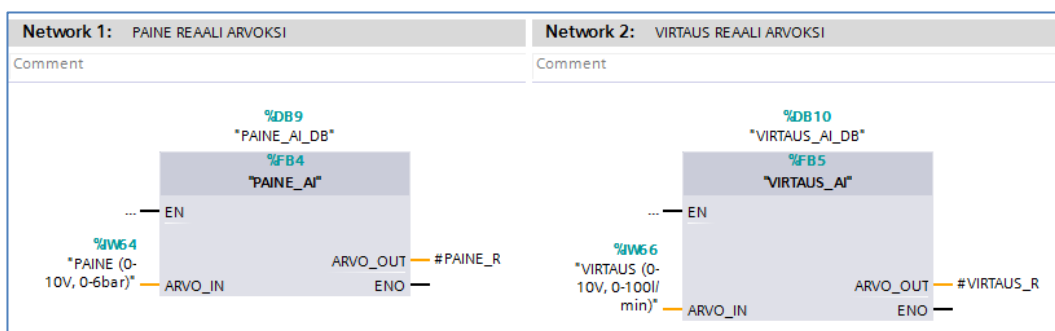


**Kuvio 37.** Standardiviestin muuttaminen reaaliluvuksi.

## 10 OHJELMAN LUOMINEN TOIMILOHKOILLE

### 10.1 Toimilohkojen ohjelmointi

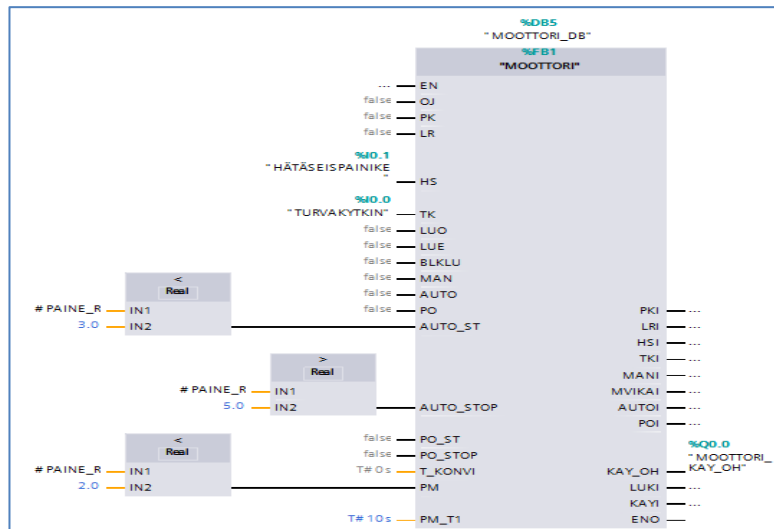
Toimilohkojen valmistumisen jälkeen aloimme luoda ohjelmaa toimilaitteille. Loimme OB-lohkon (Organization Block) ja tähän lohkoon luotiin varsinainen ohjelma miten laitteet toimisivat eri tilanteissa. Ensimmäiseksi muutimme analogitulot eli paine- ja virtausarvot reaali luvuksi apumuuttujiin PAINE\_R ja VIRTAUS\_R (**Kuvio38**).



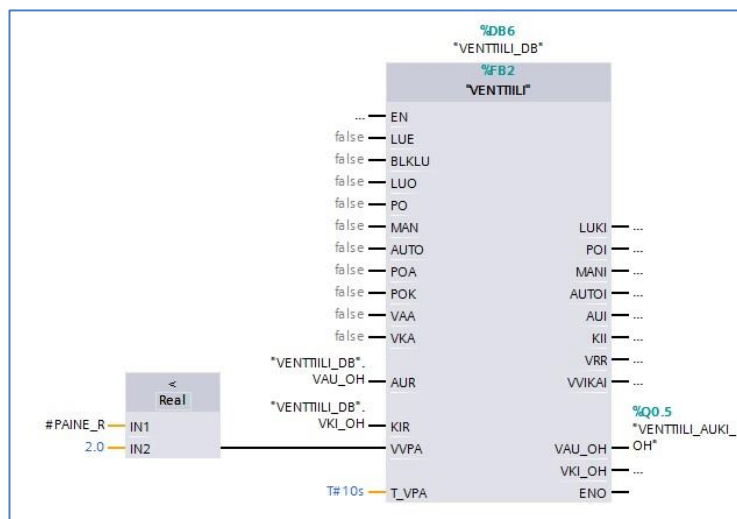
**Kuvio 38.** Paine – ja virtausanturin muutokset reaali – luvuksi.

Seuraavaksi loimme moottorille eli kompressorille toiminnan. Turvallisuus toiminnot, jotka lisättiin jo piirikaavioissa, nämä asetettiin oikeille paikoille, koska ulkopuolelta annetut toiminnot pitivät olla kunnossa ennen kuin moottoria voitiin käyttää logiikkaohjaimen kautta. Manuaalitulassa moottori pyörii niin kauan kunnes annettiin stop-käskey tai paine tippui 10 sekunniksi alle 2 barin. Vastaavasti automaattitulassa moottori käynnistyi kun paine oli alle 3 baria ja pysähtyi kun paine ylitti 5 baria. Jos paine tippui 10 sekunniksi alle 2 barin, niin moottori pysähtyi ja vika ilmoitus tuli näyttöpaneeliin. Kun moottorille annettiin käy-käskey, niin CPU:n Q0.0:ssa syttyi merkkivalo. Venttiilille asetettiin auki- ja kiinniraja, jotka annettiin auki- ja kiinni-ohjauksen tiloista. Vika käskey asetettiin venttiilille samoin kuin moottorille eli käskey saatiin kun paine tippui alle 2 barin 10 sekunnin

ajaksi. Venttiilin annettiin auki-ohjaus syttyi merkkivalo CPU:n Q0.5:ssä (Kuvio 39. – 40.).



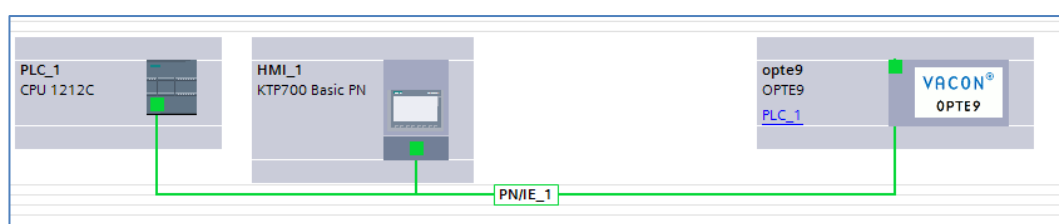
Kuvio 39. Moottorille asetetut toiminnot.



Kuvio 40. Venttiilille asetetut toiminnot.

## 10.2 Taajuusmuuttajan ohjelmointi

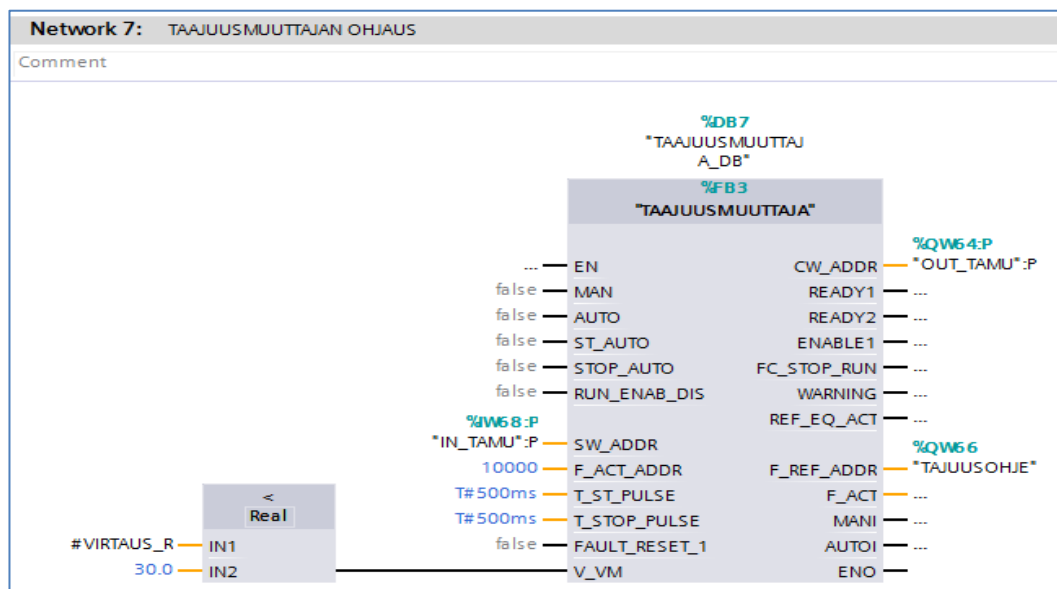
Jotta taajuusmuuttaja voitiin käyttää, jouduttiin tälle lataamaan profinet-kortin tiedosto. Kortin valitsemisen lisäksi tälle valittiin telegram-tiedosto, jotta tieto saatiin välitettyä logiikalle. Taajuusmuuttajalle tulevat input- ja output-osoitteet tarkistettiin profinet-kortin asetuksista. Järjestelmän laitteiden näkymä network viewissä on kuvattu kuvioon 41.



**Kuvio 41.** Järjestelmän laitteet network viewissä.

Profinet-kortin input word (IW68) asetettiin SW\_ADDR:ään, josta taajuusmuuttajalohko lukee tila-arvot. Vastaavasti CW\_ADDR:ään asetettiin output word (QW64), jolla ohjattiin taajuusmuuttajaa. Start- ja stop-pulsseille annettiin toiminta-ajaksi 500 ms ja taajuusmuuttajan ohjearvo 10 000 asetettiin F\_ACT\_ADDR:ään. Taajuusmuuttajan taajuusohje-arvo asetettiin QW66:een, tällä ohjattiin pumpun pyörisnopeutta (**Kuvio 42.**).





**Kuvio 42.** Taajuusmuuttajan toiminnot.

### 10.3 Käyttöliittymän luominen

Käyttöliittymä luotiin Siemensin omalla WinCC -ohjelmistolla. Käyttöliittymä luotiin yksinkertaiseksi ja helpolukaiseksi, jossa oli selkeästi tarvittavat toiminnot esillä pumpulle, kompressorille ja venttiilille.

Kaikkiin komponentteihin luotiin automaatin- ja manuaalin valintapainikkeet ja näiden indikoinnit, jotka näkyivät siten että valittu toiminto näkyi vihreällä ja ei käytössä ollut toiminto punaisella. Taajuusmuuttajalle ja moottorille tehtiin start- ja stop -painikkeet sekä venttiilille kiinni- ja auki -painikkeet, myös näille luotiin indikoinnit. Indikoinneissa noudatettiin samanlaista kaavaa kuin tilojen valinnoissa. Esimerkiksi kun moottori tai taajuusmuuttaja sai start-käskyn, laatikon väri vaihtui vihreäksi ja teksti muuttui kay-tilaan. Vastaavasti stop-käskyllä tuli punai-

nen väri ja ja stop-teksti. Painearvon määrä asetettiin samaan osaan kuin kompressorin toiminta.

Pumpulle luotiin myös säätöpainikkeet, josta saatiin säädettyä pumpun kierrosnopeutta, joka näkyi prosentteina. Plus-painikkeesta taajuusmuuttajan nopeus kasvoi 2-prosenttiyksikköä ja miinus-painikkeesta vähennettiin sama yksikkömäärä. Virtauksen määrä asetettiin pumpun toimintojen kanssa samaan osioon.

Kaikille toimilaitteille luotiin vian antava ilmoitus, siten että vika-tilan tulesa voimaan alkoi vian ilmoittava ”laatikko” vilkkua keltaisena. Näiden lisäksi kompressorille ja venttiilille lisättiin lukitusten on-off-toiminto, jotta näytöltä tapahtuvat toiminnot saatiin välitettyä toimilaitteelle lukitusten ollessa on-tilassa (**Kuvio 42.**).



**Kuvio 43.** Näkymä käyttöliittymästä.

## **10.4 Testaus**

### **10.4.1 Yleistä testauksesta**

Ohjelmien testauksella etsitään suunnitelmallisesti toimintojen virheitä ja tähän luetaan käytettävyyden, luotettavuuden, suorituskyvyn yläpidettävyyden varmistaminen. Umpimähkäisesti testattuja ohjelmia tai sen osia ei lueta testaukseksi. Testausmenetelmät voivat vaihdella yrityksen, projektin tai henkilön mukaan, tämän vuoksi testaukseen on syytä kiinnittää erityistä huomiota. Testaus voidaan jakaa eri alueisiin kuten rakenteelliseen, käytön näkökulman tai toiminnalliseen testaukseen.

Rakenteellisen testauksen kohteena ovat komponentit ja muut rajapinnat, käytön näkökulma tarkoittaa nimen mukaisesti käyttäjäkohtaisia toimintoja ja toiminnallisessa testauksessa tarkistetaan, että halutut toiminnot suorittavat annetut käskyt. Testausta olisi aina hyvä käsitellä kokonaisena prosessina, mutta pienet ja yksinkertaiset prosessit ovat poikkeustapauksia. Testaus voidaan suorittaa toimittajan tiloissa, mutta kokonaisen järjestelmän testaaminen olisi tärkeää suorittaa lopullisessa sijoituspaikassa, jotta koko järjestelmän toimivuutta voidaan tarkastella. /5/

### **10.4.2 Laitteiston testaaminen**

Kun toimilaitteiden ohjelmointi saatiin valmiiksi, tehtiin kaikille laitteille myös käyttöpaneelille testaukset. Testauksessa tarkistettiin, että toimilaitteiden lohkot ja näille tehdyt ohjelmoinnit toimivat moitteettomasti. Käyttöpaneelin osalta tarkistettiin, että valvomolta annettavat komennot ja näiden indikoinnit toimivat halutulla tavalla, myös taajuusmuuttajan säätöpainikkeet testattiin toimiviksi. Lisäksi tarkistettiin, että virtaus- ja paineanturit antavat halutut arvot näytölle.

## 11 YHTEENVETO

Insinööriyössä suunniteltiin pohjaratkaisu automaatiolla ohjattavalle toimilaitteelle. Projektissa toteutettiin helposti muokattavissa olevia ratkaisuja ja koska työn on vasta pohjaratkaisu, sitä tullaan varmasti muuttamaan tulevaisuudessa käyttökohteen mukaan.

Opinnäytetyön tekeminen aloitettiin automaation kuvauksella, jossa kuvattiin mahdollinen toimilaite ja tämän toiminta. Tämän jälkeen projektiin valittiin sopiva logiikkaohjain, taajuusmuuttaja sekä muut toimilaitteet. Valintojen jälkeen piirikaavioiden piirtäminen sujui odotetusti koulussa piirrettyjen harjoitusten perusteella. Myös ohjelmointi sujui koulusta saatujen oppien mukaisesti, vaikka muutamia ongelma kohtia havaittiin työn edetessä. Ongelmat painoutuivat uuteen TIA Portal –ohjelmistoon, koska muutamissa toiminnoissa pienet lisät sekä muutokset vanhaan versioon tuottivat päänvaivaa.

Kokonaisuudessaan opinnäytetyöstä saatiin kattavasti tietoa mitä kuuluu automaatiojärjestelmän suunnitteluun. Suurin osa ajasta kului logiikan ohjelmointiin, sillä tästä ei vielä ollut riittävästi kokemusta. Mutta tämä projekti on tulevaisuudessa hyvänä pohjana, kun ajatellaan mahdollisia työtehtäviä.

## LÄHTEET

/5/ Asmala H, Koskinen K, Koskela M, Mätäsniemi T, Soini A, Strömman M, Tommila T & Valkonen J. 2005. Automaatiosovellusten ohjelmistokehitys: Suunnittelun työtavat, välineet ja sovellusarkkitehtuuri. 1. painos. Helsinki. Paino-merkki Oy. Viitattu 7.11.2014

/4/ Fonselius J, Pekkola K, Selosmaa S, Ström M & Välimaa T. 1999. Automaatiolaitteet. 1.-2. painos. Helsinki. Oy Edita Ab. Viitattu 6.11.2014

/6/ Heimbürger H, Markkanen P, Norros L, Paunonen H, Savioja P, Sundquist M & Tommila T. 2010. Valvomo: Suunnittelun periaatteet ja käytännöt. 2.painos. Helsinki. Suomen Automaatioseura ry. Viitattu 3.11.2014

/8/ Lågland H, Ala-Toppari J, Sulkumäki H, Raivio T, Laitinen M, Leivo P, Sulonen R, Örn L, Koskinen O, Lehtelä P, Virtanen V & Jumpponen E. 1992. Sähköasennukset 2. 2.painos. Espoo. Sähköurakoitsijaliiton Koulutus ja Kustannus Oy. Viitattu 27.10.2014

/3/ Nieminen, J./ Ylivainio, M. 2011, Ohjelmoitavat logiikat luentomateriaali. Vaasan ammattikorkeakoulu. Viitattu 2.6.2014

[www.cc.puv.fi/~my/KURSSIMATERIAALIT/OHLO\\_K13/b\\_TEORIAA%20JA%20OHJEITA/OILu1\\_26.10.11.doc](http://www.cc.puv.fi/~my/KURSSIMATERIAALIT/OHLO_K13/b_TEORIAA%20JA%20OHJEITA/OILu1_26.10.11.doc)

/11/ Nieminen J. Rakenteellinen ohjelmointi. Vaasan ammattikorkeakoulu. Viitattu 17.1.2015

[www.cc.puv.fi/~jun/Da/Ohlo\\_Rakenteellinen%20ohjelmointi.doc](http://www.cc.puv.fi/~jun/Da/Ohlo_Rakenteellinen%20ohjelmointi.doc)

/7/ Siemensin verkkosivut. Viitattu 21.10.2014

[https://www.automation.siemens.com/salesmaterial-as/brochure/en/brochure\\_panels\\_en.pdf](https://www.automation.siemens.com/salesmaterial-as/brochure/en/brochure_panels_en.pdf)

/9/ S7-1200 System manual. Viitattu 1.11.2014.

<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=6ES7212-1BE40-0XB0&caller=view>

/12/ Vacon NX OPTC Profinet IO Board User Manual. Viitattu 17.1.2015.  
[http://www.vacon.com/ImageVaultFiles/id\\_3124/cf\\_2/Vacon-NX-OPTCP-Profinet-IO-Board-User-Manual-DPD00.PDF?635223813622430000](http://www.vacon.com/ImageVaultFiles/id_3124/cf_2/Vacon-NX-OPTCP-Profinet-IO-Board-User-Manual-DPD00.PDF?635223813622430000)

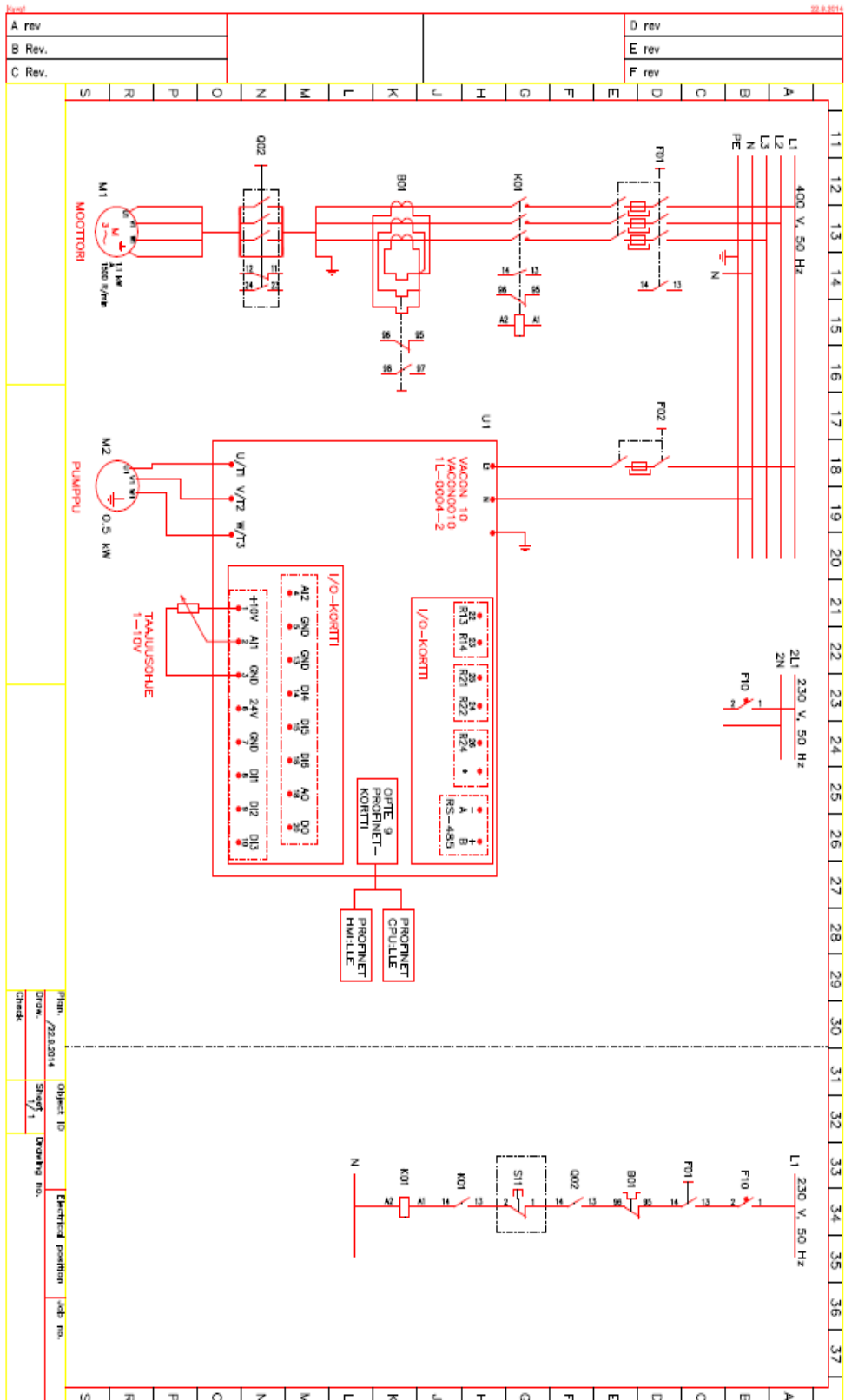
/10/ Vacon 10 Complete manual. Viitattu 27.11.2014. Vacon-10-Complete-Manual-DPD01318F1-FI.PDF

[http://www.vacon.com/ImageVaultFiles/id\\_3124/cf\\_2/Vacon-NX-OPTCP-Profinet-IO-Board-User-Manual-DPD00.PDF?635223813622430000](http://www.vacon.com/ImageVaultFiles/id_3124/cf_2/Vacon-NX-OPTCP-Profinet-IO-Board-User-Manual-DPD00.PDF?635223813622430000)

/1/ Vasel Vaasan Sähköasennus Oy. Viitattu 22.5.2014  
[www.sahkoasennus.com/web/](http://www.sahkoasennus.com/web/)

/2/ Vasel Vaasan Sähköasennus Oy. Viitattu 22.5.2014  
<http://www.sahkoasennus.com/web/palvelut>

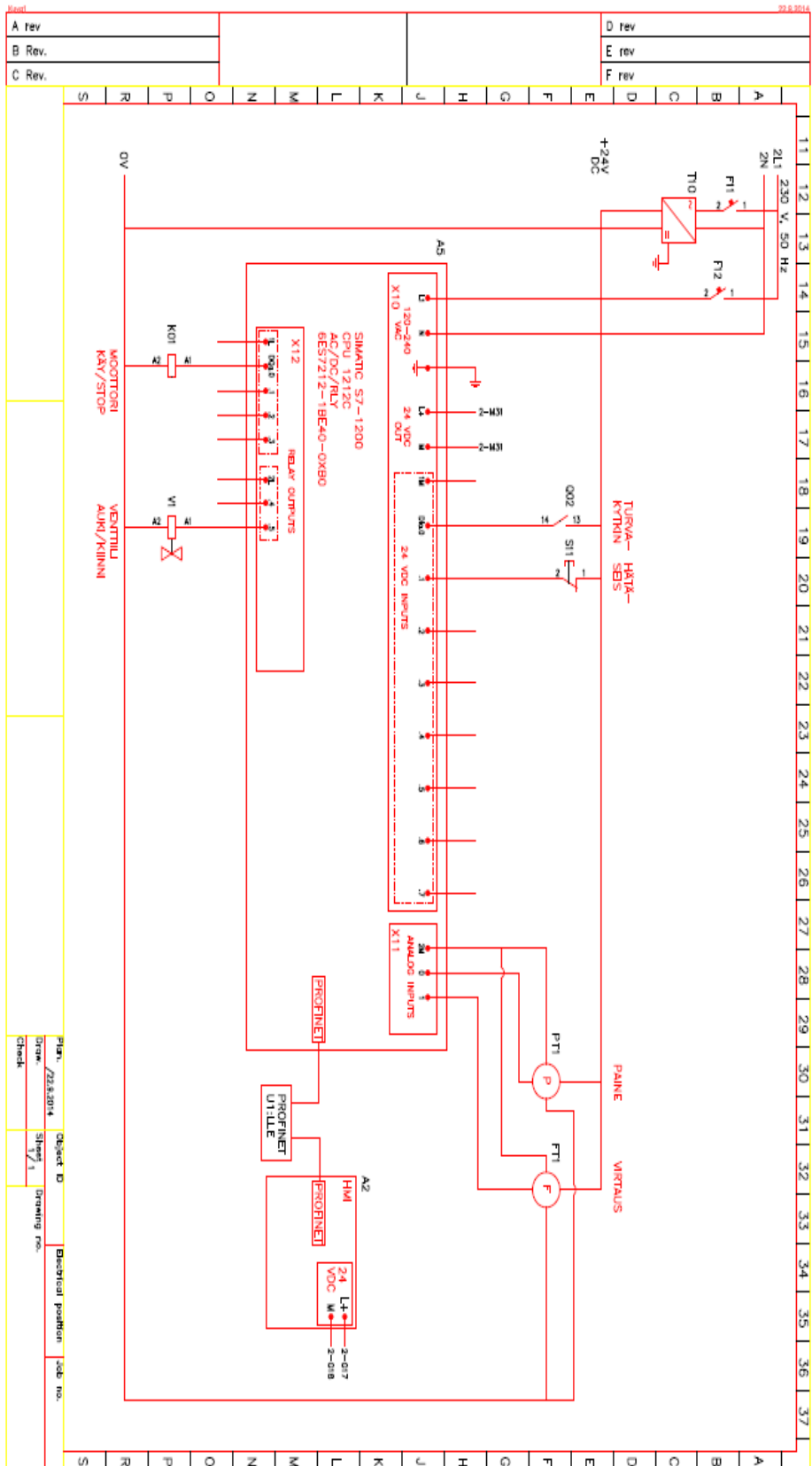
# LIITE 1



A rev	D rev
B Rev.	E rev
C Rev.	F rev

Plan / 22.2014	Object ID	Electrical position	Job no.
Draw.	Sheet / 1		
Check	Drawing no.		

# LIITE2



Plan: /22.5.2014  
 Draw: /1/1  
 Object ID: /1/1  
 Drawing no.: /1/1  
 Electrical position: /1/1  
 Job no.: /1/1

A Rev.		D Rev.	
B Rev.		E Rev.	
C Rev.		F Rev.	