



SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Kristo Atanassov

Logiikkakaaviomallit prosessiautomaation suunnittelun apuna

Opinnäytetyö
Kevät 2023

Insinööri (AMK), Automaatiotekniikka



SEINÄJOEN AMMATTIKORKEAKOULU

Opinnäytetyön tiivistelmä

Tutkinto-ohjelma: Insinööri (AMK), Automaatiotekniikka

Suuntautumisvaihtoehto: Koneautomaatio

Tekijä: Kristo Atanassov

Työn nimi: Logiikkakaaviomallit prosessiautomaation suunnittelun apuna

Ohjaaja: Matti Panula

Vuosi:2023

Sivumäärä:46

Tässä opinnäytetyössä esitetään logiikkakaavioiden luomisen periaatteet prosessiteollisuuden asiakastöille. Asiakastyön avulla myös tutkittiin mahdollisia tapoja, joilla voitiin alkaa luoda toimintatapakuvausta uudelle työntekijälle, kun tämä perehdytetään logiikkakaavioiden luomiseen.

Logiikkakaaviot ovat graafisia piirustuksia automaatioprosessin toiminnasta. Tämän työn logiikkakaaviot luotiin asiakastyönä tilattua automatisoitua prosessia varten, ja käytännön työesimerkeissä käytettiin BricsCAD-ohjelmaa. Näiden työssä luotujen logiikkakaavioiden tarkoituksena oli toimia mallina, jonka pohjalta kolmannen osapuolen ohjelmoija voi luoda automaatio-ohjelman IEC 61131-3 -standardissa mainittuja ohjelmointitapoja käyttäen. Logiikkakaavioiden luomista nopeuttamaan rakennettiin objektikirjasto, johon kerättiin käytetyimmät logiikkaportit sekä kokoelma toimilohkoja ja funktiolohkoja. Objektikirjastoa voidaan myöhemmin käyttää hyödyksi muissakin logiikkakaavioita sisältävissä projekteissa.

Asiakastyön valmistuttua luotiin yleinen toimintatapaohjeistus logiikkakaavioiden luomisesta yrityksen sisäiseen käyttöön. Toimintatapakuvausten lopullista luomista ei ole esitetty opinnäytetyössä, koska kyseinen kuvaus saattaa sisältää luottamuksellista tietoa asiakkaasta.

¹ Asiasanat: BricsCAD, logiikkakaavio, toimilohko

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Thesis abstract

Degree programme: Automation Engineering

Specialisation: Machine Automation

Author/s: Kristo Atanassov

Title of thesis: The usage of logical diagrams in process automation

Supervisor(s): Matti Panula

Year:2023

Number of pages:42

The thesis studied the main principles of creating logical diagrams for the customer work of process automation industry. The thesis also gave the opportunity to study the creation of functional descriptions and be well informed about logical diagrams at the same time.

Logical diagrams are graphical drawings of the functioning of an automation process. In the thesis all logical diagrams were created for an automation process client. The actual program was made by using a CAD software called BricsCAD. The actual logical diagrams were meant to be used as models, based on which a programmer can create an automation process according to the programming methods mentioned in the IEC-61131-3 standard. To speed up the creation of logical diagrams an object library was created. It included the most often used logical ports and function blocks. Later, the object library can be used in other logical diagram projects.

After finishing the work for the client, the thesis was utilized as a basis for a procedure manual needed for creating logical diagrams for the use of the client of the thesis. The actual procedure manual was not included in the thesis since it is confidential.

¹ Keywords: BricsCAD, logical diagram, Function block

SISÄLTÖ

Opinnäytetyön tiivistelmä	2
Thesis abstract	3
SISÄLTÖ	3
Kuva-, kuvio- ja taulukkuuettelo	6
Käytetyt termit ja lyhenteet.....	8
1 JOHDANTO	9
1.1 Työn tausta	9
1.2 Työn tavoitteet.....	9
1.3 Työn rakenne	10
1.4 PCS-Engineering Oy	10
2 AUTOMAATIOSUUNNITTELUN DOKUMENTIT	11
2.1 PI-kaavio	11
2.2 Toimintakuvaus	13
2.3 Logiikkakaavio.....	14
2.4 IO-lista	16
3 KÄYTETYT TYÖKALUT	17
3.1 CAD-Ohjelma	17
3.2 Tekstinkäsittely.....	18
3.3 Objekti kirjasto BricsCAD:iin.....	20
4 TYÖN TOTEUTUS	23
4.1 Lähtötiedot	23
4.2 Sekvenssit.....	23
4.3 Toimilohkot.....	25
4.3.1 Moottorilohko	25
4.3.2 Venttiililohko.....	30
4.3.3 Ohjausloho	33
4.3.4 Mittausloho.....	37
4.3.5 Laskuri	40

4.4 Työn viimeistely.....	42
5 TULOKSET	43
6 YHTEENVETO JA POHDINTA.....	45
LÄHTEET	46

Kuva-, kuvio- ja taulukkoluetelo

Kuva 1. Esimerkki PI-kaaviosta .	12
Kuva 2. IEEE Std 91/91a standardin logiikkakaavioiden ilmaisutapa	15
Kuva 3. IEC-60617-12 standardin logiikkakaavioiden ilmaisutapa.....	15
Kuva 4. Esimerkki IO-listasta	16
Kuva 5. BricsCAD käyttöympäristö	18
Kuva 6. Batte-käyttöliittymä	19
Kuva 7. Objektikirjasto	20
Kuva 8. Esimerkki sekvenssin askeleesta	24
Kuva 9. Moottorin toimilohko.....	27
Kuva 10. Moottorin käynnistäminen ja pysäyttäminen	29
Kuva 11. Piiriin lisätyt pysähtymisen aiheuttavat ehdot.....	29
Kuva 12. Moottorin valmis logiikkakaavio	30
Kuva 13. Venttiilin toimilohko	31
Kuva 14. Venttiilin avautuminen ehdoilla	32
Kuva 15. Valmis venttiilin logiikkakaavio.....	33
Kuva 16. Ohjauksen toimilohko	34
Kuva 17: Esimerkkitoteutus ohjauslohkon käytöstä	36
Kuva 18: Mittauksen toimilohko	37
Kuva 19: Valmis mittauksen logiikkakaavio	39
Kuva 20: Laskurin toimilohko	40

Kuva 21: Esimerkki laskurin käytöstä	41
--	----

Käytetyt termit ja lyhenteet

LAD	Ladder Diagram
FBD	Function Block Diagram. Kaavio, joka kuvastaa ohjelman toimintoa toimilohkoilla.
SFC	Sequential Function Charts. Visuaalinen ohjelmointikieli, joka perustuu enemmän ohjelman askeleiden luomiseen kuin yksittäisiin lohkoihin.
ST	Structured Text. Perinteisempi tekstipohjainen PLC-ohjelmointikieli
IL	Instruction List. Askelmaiseen ohjelmointitapaan perustuva ohjelmointikieli.
Binäärinen signaali	Signaali, jonka arvo on joko 1 tai 0.
Analoginen signaali	Signaali, jonka arvo voi muuttua asetettujen arvojen mukaisesti
IEC 61131-3	Standardi, jossa määritellään viisi käytetyintä ohjelmointikieltä sekä datatyypit, joita käytetään PLC-ohjelmoinnissa.
NC	Normaalisti kiinni oleva piiri tai toimilaite
NO	Normaalisti auki oleva piiri tai toimilaite

1 JOHDANTO

1.1 Työn tausta

Prosessiteollisuuden automaatioprojekteissa luodaan ensin toimintakuvaus sekä laajemmasta tehdaskokonaisuudesta tai sen osa-alueista, kuten esimerkiksi yksittäisestä toimilaitteesta. Toimintakuvaus voi olla siis dokumentti toimilaitteesta tai käytettävän ohjelman askeleesta. Toimintakuvaus kuvaa toimilaitteen signaalit, toimintaehdot sekä ohjausmahdollisuudet, joita toimilaitteen käyttäjällä tulee olla. Tämän toimintakuvauksen pohjalta luodaan logiikkakaavio itse automaatio-ohjelman luomista varten. Logiikkakaavio kuvastaa ohjelman toimintaa graafisessa muodossa, ja sen pohjalta ohjelmaa luovan henkilön on mahdollista kääntää kaavioihin piirretty ohjelma toimivaksi toteutukseksi valitsemaansa ohjelmaan.

1.2 Työn tavoitteet

Työn tavoitteena on esitellä lukijalle logiikkakaaviot sekä se, mihin ja miten niiden luominen perustuu sekä miten logiikkakaavioita käytetään. Tämän tavoitteen saavuttamiseksi työssä on esitelty logiikkakaaviot sekä keskeisimmät toimilohkot, joita logiikkakaaviotyössä käytetään. Työn jälkeen lukijan tulisi ymmärtää logiikkakaavioissa käytettyjen toimilohkojen toimintatapa sekä käyttökohde. Tehdyn työn pohjalta lähdettiin luomaan toimintatapaohjetta uudelle työntekijälle, jotta voitaisiin nopeuttaa perehdytysprosessia, ja säästää näin yrityksen resursseja.

1.3 Työn rakenne

Luku kaksi sisältää työn teoriaosuuden, jossa käydään läpi automaattisuunnittelun keskeisimmät dokumentit. Luvussa kolme käsitellään työkaluja, jotka ovat hyödyksi logiikkakaavioiden luomisessa yleisellä tasolla. Luvussa neljä esitellään objektikirjasto, joka luotiin kaavioiden luomisen sujuvuutta parantamaan. Luvussa neljä esitellään myös logiikkakaaviot sekvenssien sekä toimilaitteiden toimilohkojen suhteen. Luvussa viisi käydään läpi työn tulokset. Luvussa kuusi käydään läpi pohdinta ja yhteenveto.

1.4 PCS-Engineering Oy

PCS-Engineering Oy on Seinäjoella, Oulussa, Jyväskylässä sekä Rovaniemellä toimiva suomalainen yritys (PCS-Engineering, i.a.). PCS-Engineering (Project Consulting Services) tarjoaa nimensä mukaisesti konsultointipalveluita pääsääntöisesti prosessiteollisuudessa. Näihin palveluihin sisältyvät ohjelmistosuunnittelu, käyttöönotto sekä projektisuunnittelu. PCS-Engineeringillä on ollut projekteja lukuisille eri teknologia-alalla toimiville yrityksille, esim. Valmet, VEO, Outokumpu sekä SSAB.

2 AUTOMAATIOSUUNNITTELUN DOKUMENTIT

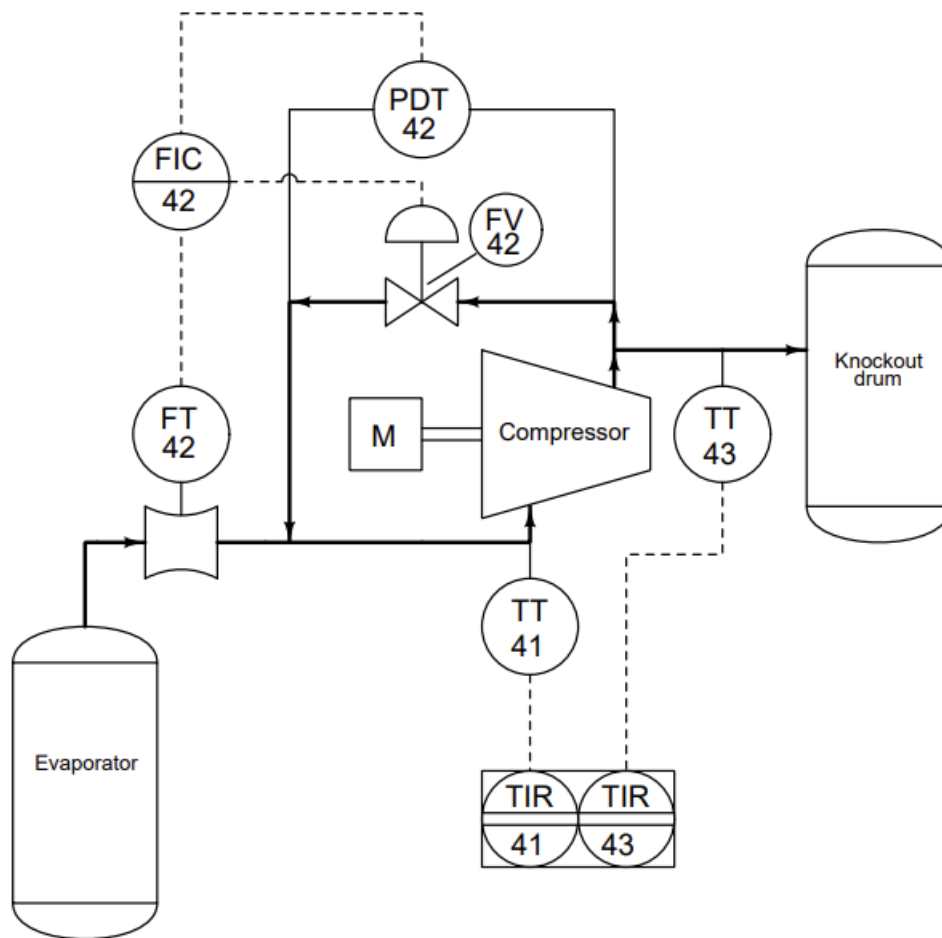
Automaatiosuunnittelun dokumentointi kattaa kaikki automaatioprojektisuunnittelussa tarvittavat tiedot, jotka tarvitaan automatisoidun prosessin toteuttamiseen. Automaatiosuunnittelun dokumentointi kattaa PI-kaaviot, toimintakuvaukset, logiikkakaaviot ja IO-listat. Mikään näistä ei ole standardoitu dokumentti ilmaisutavaltaan, mutta niiden sisällössä on jo standardoituja elementtejä. Näitä standardisoituja elementtejä ovat esimerkiksi logiikkakaaviossa olevat toimilohkot ja funktiolohkot. Tämä standardisointi edesauttaa dokumenttien yhtenäistä ilmaisutapaa sekä käyttöä prosessiteollisuudessa. Näiden dokumenttien luominen ja säilyttäminen on välttämätöntä koko prosessin toteutuksen luomiseksi, ylläpitämiseksi ja päivittämiseksi.

2.1 PI-kaavio

PI-kaavio (putkitus ja instrumentointi) on ensimmäisiä automaatiosuunnittelussa luotavia dokumentteja. PI-kaavio on automaation perussuunnittelun yksi tärkeimmistä dokumenteista, sillä siitä saadaan selville prosessin putkisto. PI-kaavio on laitoksen putkistoa ja instrumentointia kuvaava piirustus, josta selviää toimilaitteet, joita kokonaisuudessa on käytetty (Kalani, 2002, s. 7).

PI-kaavion tarkoituksena on antaa kuva prosessin teknillisistä ratkaisuista sekä esittää putkien ja muiden kuljetusteiden kulku prosessin kokonaisuudessa. PI-kaavion avulla yleensä aloitetaan myös toimintakuvauksen luominen toimilaitteille. Samoin sitä voidaan hyödyntää myös kustannusarvion laadinnassa prosessista sekä siitä saadaan materiaaliluetteloon tiedot.

PI-kaavioiden piirustuksessa noudatetaan standardeja ISO 10628 ja ISO 14617. Standardien tarkoituksena on määrittää symbolit toimilaitteille, joita käytetään kemikaaliteollisuudessa (ISO 10628), sekä luoda kirjasto, jonka pohjalta voidaan hakea symbolit käytettäväksi teknisissä sovelluksissa (ISO 14617) (International Organization for Standardization (ISO), 2004, s. 6); ISO 2014, s. 11). Näistä standardisoiduista objekteista nähdään esimerkki kuviossa 1.



Kuvio 1. Esimerkki PI-kaaviosta (Kuphaldt, 2022, s. 562).

2.2 Toimintakuvaus

Toimintakuvaus on automaatio suunnittelun dokumenteista luomisjärjestyksessä toinen, koska se luodaan PI-kaavion pohjalta. Toimintakuvauksen luominen on välttämätöntä, kun kyseessä on automatisoitu kokonaisuus. Toimintakuvauksen avulla ohjelmaa luovat henkilöt tietävät, mitä asiakas haluaa sekä millaista toteutusta nämä alkavat luomaan kuvauksessa mainittujen ehtojen mukaan.

Toimintakuvaus on dokumentti toimilaitteesta tai sekvenssin askeleesta, jota ohjelmassa käytetään (Realpars, i.a). Toimintakuvauksessa on listattu signaalit, joita toimilaitteeseen tulee, ehdot, joilla toimilaite toimii, sekä mahdolliset ohjausmahdollisuudet, joita toimilaitteen käyttäjällä tulee olla. Toimintakuvaukseen on myös selostettu sekvenssin toiminta sekä ehdot, joilla sekvenssi etenee askeleessa seuraavaan. Toimintakuvauksen avulla luodaan myöhemmin myös logiikkakaaviot. Alla on kirjallinen esimerkki ultrafilteripumpun toimintakuvauksesta. Viitattua kuvasta selviää ehdot, joilla pumppu pysähtyy, sekä ehdot, jotka ohjaavat pumpun käynnistymistä. Näitä ehtoja ovat sekvenssistä tulevat käskyt sekä pysähtymisen aiheuttavat rajamittaukset, jotka on liitetty pumppuun tulevana signaaleina. Toimintakuvauksessa isolla kirjoitettu AND ja OR kuvastavat ehtoja signaaleille. Alla on esitetty kirjallinen toimintakuvaus:

Ultrafilter feed pump

Function

- MANUAL-mode
 - Can be set freely from display
- AUTO-mode
 - Controlled by ultrafilter feed pump sequences.

Interlocks

- Interlocks stop the pump when:
 - Ultrafilter has not been running for over 30 seconds OR
 - Feed tank outlet is open AND Feed tank level is below LL limit OR
 - Washing water tank outlet is open AND washing water tank level is below LL limit OR
 - Permeate tank outlet is open AND permeate tank level is below LL limit OR
 - Sealing water flow low alarm has been on for over 5 seconds

ALARMS

- According to project motor template

2.3 Logiikkakaavio

Logiikkakaaviot luodaan prosessin toimintakuvauksen pohjalta. Logiikkakaavio on graafinen piirustus, joka kuvastaa prosessin ohjauksen ohjelman vaiheita tulo- ja lähtösignaalien kanssa (Battikhas, 2017, s. 10). Logiikkakaavio voi myös kuvastaa yksittäisen toimilaitteen toimilohkoa, joka sisältää ehdot, joilla toimilaite käynnistyy tai pysähtyy. Logiikkakaavio ei välttämättä ole standardin määrittelemä dokumentti, vaan logiikkakaavio voi olla esimerkiksi käsin tehty esimerkki ohjelman toiminnasta. Logiikkakaavion etu on, että se kuvastaa toimilaitteen tai sekvenssin toimintaa riippumatta käytetystä ohjelmointikielestä tai ohjelmointitavasta. Logiikkakaavion tarkoituksena on siis, että sen pohjalta voidaan luoda automaatio-ohjelma hyödyntäen jotakin standardissa IEC 61131-3 määritettyä ohjelmointitapaa. Lyhkäisyydessään logiikkakaavio on graafinen ja yleiskäyttöisempi versio toimintakuvauksesta, joka ei ole riippuvainen kuvauksen kielestä tai esitystavasta.

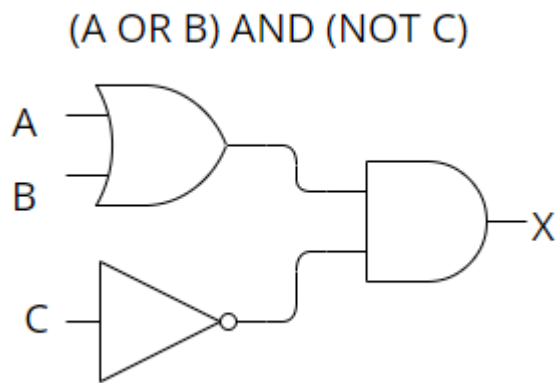
IEC-61131-3

Standardi IEC 61131-3 määrittää viisi automaatio-ohjelmoinnissa käytetyintä ohjelmointikieltä (Hanssen, 2015, s. 200). Näistä ohjelmointikielistä kolme ovat graafisia ja kaksi tekstipohjaisia. Nämä ohjelmointikielet ovat **LAD** Ladder Diagram, **SFC** Sequential Function Charts, **FBD** Function Block Diagram, **ST** Structured Text sekä **IL** Instruction List.

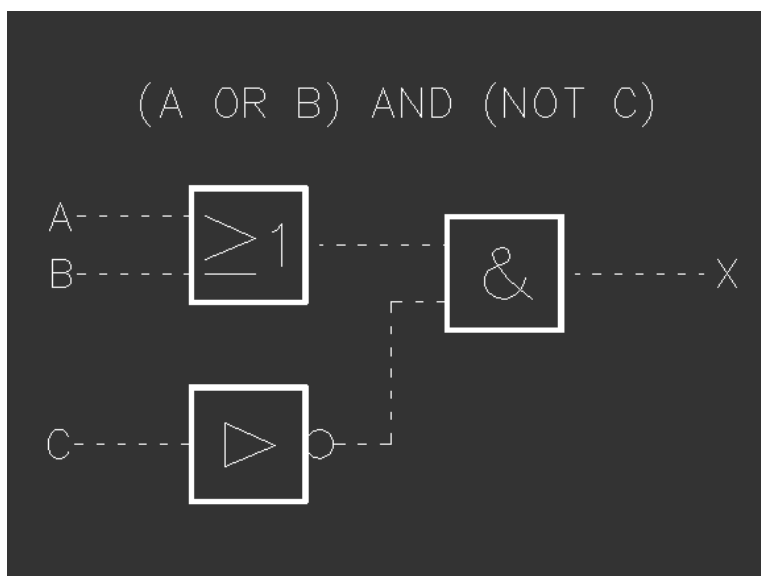
Objektien ilmaisu logiikkakaaviossa

Tapa, jolla logiikkakaaviossa ilmaistaan funktiot, riippuu kaavion käyttökohteesta. Prosessiteollisuuden ja yleisen elektroniikan logiikkakaavioiden funktiolohkot eroavat ilmaisutavaltaan toisistaan huomattavasti. Elektroniikassa yleisesti käytettävien funktioiden ilmaisussa suositaan standardia IEEE Std 91/91a-1991 (Al-Alaoui, 2008, s. 268). Prosessiteollisuudessa käytettyjen objektien ilmaisussa suositaan standardia IEC-60617-12.

Elektroniikassa käytetty logiikkakaavio esitetään kuviossa 2. Prosessiteollisuudessa suositusta logiikkakaavion esitystavasta näytetään esimerkki kuviossa 3.



Kuvio 1. IEEE Std 91/91a standardin logiikkakaavioiden ilmaisutapa



Kuvio 3. IEC-60617-12 standardin logiikkakaavioiden ilmaisutapa

2.4 IO-lista

IO-lista on listaus prosessin eri toimilaitteiden positiotunnuksista (Automationforum, i.a.) Nämä tunnuksot kuvastavat toimilaitteen tulevaa sekä lähtevää signaalia (Input / Output). IO-listassa on myös määritetty analogisten signaalien mitta-alue, mittauksen tyyppi sekä käytetyn mittauksen yksikkö.

IO-listat eivät myöskään ole standardisoituja dokumentteja. Tästä huolimatta IO-lista on yksi tärkeimmistä dokumenteista automaatio-ohjelmoinnin suhteen. IO-listassa voidaan myös määrittää osoitteet, joita toteutuksen toimilaitteet ohjelmoitavassa logiikassa tulevat käyttämään. IO-listassa voi myös olla merkitty toimilaitteen sijainti, tunnus PI-kaaviossa sekä toimilaitteen tyyppi. Kuviossa 4. on esimerkki kuvitteellisesta IO-listasta.

FIELD PLC				DCS			
Tag	Comms Tag from PKG	Communication Tag to DCS	IO Address	Data Type	Communication Tag	IO Address	Description
RWT-UGAL		RWT-UGAL-X0	%Q0.0	BOOL	RWT-UGAL-X0	I0.0	RWTP urgent alarm
MCF-1-11011		MCF-1-11011-X1	%Q1.0	BOOL	MCF-1-11011-X1	I1.0	DAF-1 Feed Pump 1 Motor Running
		MCF-1-11011-X2	%Q1.1	BOOL	MCF-1-11011-X2	I1.1	DAF-1 Feed Pump 1 Motor Tripped
MCF-2-11012		MCF-2-11012-X3	%Q1.2	BOOL	MCF-2-11012-X3	I1.2	DAF-1 Feed Pump 2 Motor Running
		MCF-2-11012-X4	%Q1.3	BOOL	MCF-2-11012-X4	I1.3	DAF-1 Feed Pump 2 Motor Tripped
SSS-1-30031		SSS-1-30031-X5	%Q1.4	BOOL	SSS-1-30031-X5	I1.4	DAF-1 Surface Sludge Scaper 1 Motor Running
		SSS-1-30031-X6	%Q1.5	BOOL	SSS-1-30031-X6	I1.5	DAF-1 Surface Sludge Scaper 1 Motor Tripped
SLP-1-40041		SLP-1-40041-X7	%Q1.6	BOOL	SLP-1-40041-X7	I1.6	DAF-1 Sludge Pump 1 Motor Running
		SLP-1-40041-X8	%Q1.7	BOOL	SLP-1-40041-X8	I1.7	DAF-1 Sludge Pump 1 Motor Tripped
SLP-1-50051		SLP-1-50051-X9	%Q2.0	BOOL	3000-XU-30013-X-14	I2.0	DAF-1 Sludge Pump 2 Motor Running
		SLP-1-50051-X10	%Q2.1	BOOL	3000-XU-30013-X-15	I2.1	DAF-1 Sludge Pump 2 Motor Tripped

Kuvio 4. Esimerkki IO-listasta

3 KÄYTETTY TYÖKALUT

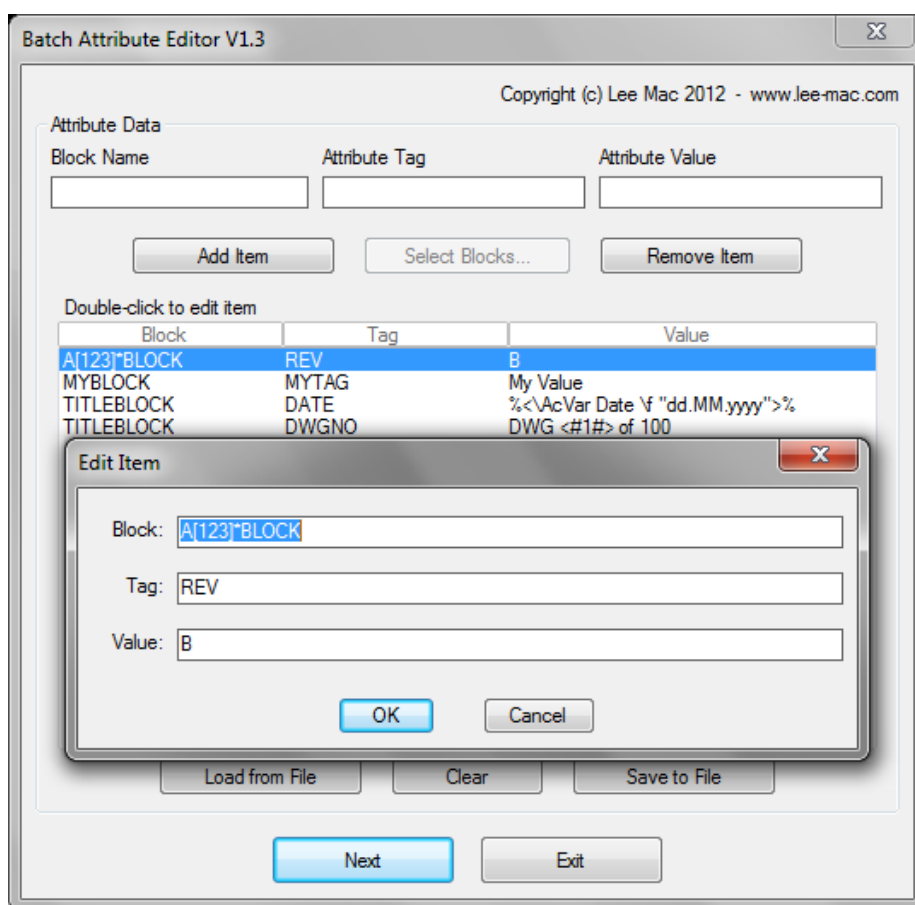
3.1 CAD-Ohjelma

CAD (Computer-Aided Design) on tietokoneelle luotu suunnittelutyökalu, jonka avulla voidaan luoda erinäisiä piirustuksia, joita käytetään tekniikan alalla. Yleisin tiedostotyyppi, jota CAD-piirustuksissa suositetaan, on dwg. Kyseistä tiedostoa tukevat lukuisat suositut CAD-ohjelmat, joten ohjelmien välinen tiedostojen vaihtaminen on mahdollista. Tämän dwg-tiedoston suosio perustuu sen mahdollisuuteen sisältää teknillisen piirustuksen kaikki tarvittavat tiedot kaksi- tai kolmiulotteisena sekä myös sen helpon muokattavuuden vuoksi. Logiikkakaavioiden luomiseen on mahdollista käyttää mitä tahansa piirto-ohjelmaa, mutta CAD-ohjelmat ovat suosituimpia tekniikan alalla.

BricsCAD on Bricsys nv:n luoma ohjelma, jolla voidaan luoda 2D-piirustuksia tai 3D-malleja, pohjapiirustuksia sekä logiikkakaavioita (Bricsys NV i.a.). BricsCAD tukee lukuisia yhteisön luomia työkaluja, jotka mahdollistavat työn sulavuuden parantamisen. Tässä työssä BricsCAD-ohjelmaa on käytetty sekvenssien sekä toimilohkojen suunnitteluun toimintakuvausten pohjalta.

BricsCAD toimii pääsääntöisenä suunnittelutyökaluna logiikkakaavioille tässä opinnäytetyössä. BricsCAD:ia päädyttiin käyttämään tämän suhteellisen edullisen lisenssihinnan vuoksi verrattuna muihin CAD-ohjelmiin sekä mahdollisuudesta käyttää yhteisön luomia työkaluja.

Batch Attribute Editorilla on myös mahdollista luoda Excel-lista, johon on listattu kaikki muokattavaksi merkatut tekstikentät. Excel-listaan generoituihin tekstikenttiin voidaan muuttaa haluttu teksti, joka tullaan vaihtamaan kaavioihin, jotka muunnetaan Batten avulla. Muokatut tekstikentät sisältävä Excel-lista voidaan ladata takaisin Batteen, minkä jälkeen ohjelma toteuttaa halutut muutokset listauksen mukaisesti.

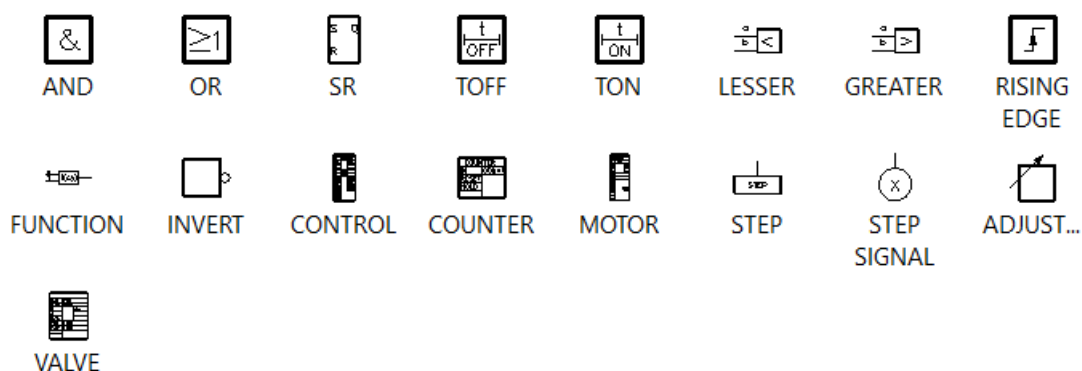


Kuvio 6. Batte-käyttöliittymä

3.3 Objektikirjasto BricsCAD:iin

Logiikkakaavioiden luomista varten tehtiin työkalukirjasto BricsCAD:iin josta voitiin hakea objekteja logiikkakaavioiden luomiseen. Työkalukirjaston avulla saatiin luotua yhtenäinen toimintamalli objektien käytölle. Objektikirjaston käytöllä varmistettiin tasainen laatu koko työprosessin aikana täten välttyen mahdollisilta aikaa vieviltä korjauksilta. Työkalukirjasto myös nopeutti uusien logiikkakaavioiden luomista, koska objekteja ei tarvinnut hakea jo valmiista kaavioista tai alkaa luomaan uudelleen jokaista uutta logiikkakaaviota varten.

Kuviossa 7 on esitelty vain tämän työn esimerkkikaavioissa käytetyt objektit, mutta varsinainen objektikirjasto, joka luotiin työtä varten, on tätäkin laajempi.



Kuvio 7. Objektikirjasto

Kuviossa 7 on esitelty objektikirjasto, jossa on kaikki esimerkkikaavioissa käytetyt objektit. Näiden objektien toiminta on seuraavanlainen:

- **AND**

Kaikkien lohkon tulevien signaalien tulee olla aktiivisia, ennen kuin lohko päästää signaalin lävitse.

- **OR**

Vähintään yhden lohkon tulevan signaalin tulee olla aktiivinen, ennen kuin lohko päästää signaalin lävitse.

- **SR**

Lähtö Q asetetaan aktiiviseksi, kun tuloon S tuleva signaali on aktiivinen. Lähtö Q nollataan, kun tuloon R tuleva signaali on aktiivinen.

- **TOFF**

Lohko sulkee lähdön tälle asetetun ajan jälkeen.

- **TON**

Lohko avaa lähdön tälle asetetun ajan jälkeen.

- **LESSER**

Arvon A tulee olla pienempi kuin arvon B, ennen kuin lohko päästää signaalin lävitse.

- **GREATER**

Arvon A tulee olla isompi kuin arvon B, ennen kuin lohko päästää signaalin lävitse.

- **RISING EDGE**

Lohkon lähtö on aktiivinen, kun signaalin arvo nousee arvosta 0 arvoon 1.

- **INVERT**

Tuleva signaali käännetään vastakohtaiseksi alkuperäiseen signaaliin verrattuna.

– **STEP**

Kuvastaa toteutettavaa askelta.

– **STEP SIGNAL**

Toteutuneesta askeleesta tuleva "Valmis"-signaali, jonka avulla voidaan suorittaa ehtoja tai käynnistää seuraava askel.

– **ADJUST**

Kuvastaa piiriin ulkopuolelta tulevaa signaalia tai arvoa, esimerkiksi ruudulta asetettua arvoa.

4 TYÖN TOTEUTUS

4.1 Lähtötiedot

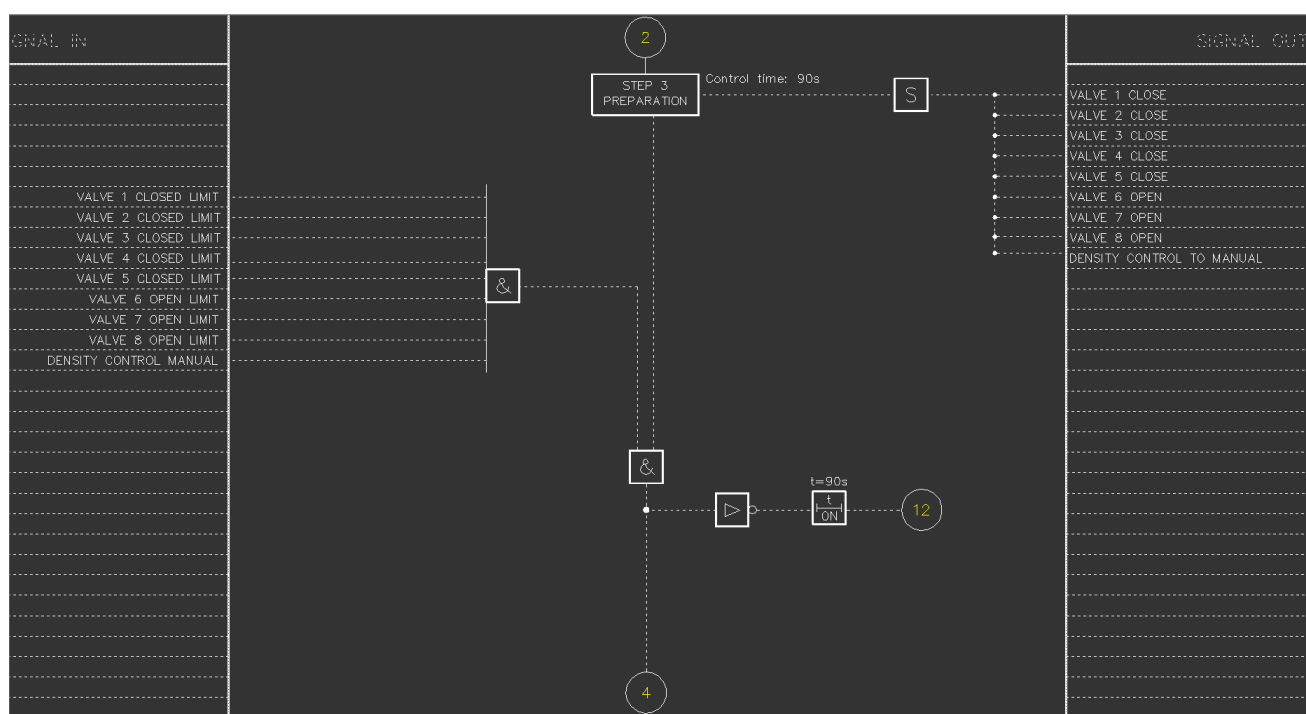
Työn keskeisenä aiheena oli itse logiikkakaavioiden luominen. Logiikkakaavioiden luominen aloitettiin perehtymällä asiakkaan antamaan toimintakuvaukseen, josta selvisi, miten toimilaitteiden tulisi toimia, kun ne ovat ohjelman tai sekvenssin ohjauksen alaisena. Asiakkaan PI-kaaviosta selvitettiin yksittäisten toimilaitteiden käyttötarkoitus, jos tämä ei tullut toimintakuvauksesta selväksi. IO-listan avulla selvitettiin logiikkakaavioissa käytettävien toimilaitteiden nimet sekä osoitteet, esimerkiksi venttiilin auki- ja kiinni-signaalin osoite.

4.2 Sekvenssit

Sekvenssin luomiseen tarvittavat tiedot saatiin toimintakuvauksista. Työssä oli kaksi pääkoh-
taa, joista ensimmäinen oli sekvenssikaaviot. Sekvenssikaavioiden toimintakuvaus on yleensä listattu ensimmäisenä koko prosessin toimintakuvauksessa. Sekvenssikaavion tarkoitus on kuvastaa tehtaan eri prosessien toimintaa askeleittain. Sekvenssikaavioiden pohjalta voi pääohjelman tekijä alkaa luomaan ohjelmaa laitoksen kokonaisprosessia varten. Nämä sekvenssikaaviot sisältävät ehdot, joiden perusteella siirrytään seuraavaan askeleeseen, sekä myös ehdot, joilla ohjelma pysähtyy, mikäli ilmenee virhetilanne toimilaitteessa tai prosessin askeleessa.

Sekvenssikaavion yksittäisen askeleen toteutustapa on seuraavanlainen: Kun askel alkaa, ohjataan toimilaitteet niille ohjelmoituun tilaan, esimerkiksi tietty ryhmä venttiilejä auki tai kiinni. Seuraavaan askeleeseen siirrytään, kun toimilaitteet ovat saavuttaneet niille asetetut ehdot. Mikäli toimilaitteet eivät lähetä vaadittuja tilatietoja askeleelle asetettuun määräaikaan mennessä, menee ohjelma vikatilaan ja sekvenssi kokonaisuutena pysäytetään. Kun sekvenssin vikatila nollataan, ohjelma on pysähtynyt ja valmis aloittamaan sekvenssin alusta ensimmäisestä askeleesta. Kuviossa 8 on esitetty sekvenssin askeleen logiikkakaavio.

Askeleen toiminta on seuraavanlainen: Kun askel 2 on valmis, aloitetaan askel 3, suljetaan venttiilit 1–5, aukaistaan venttiilit 6–8 sekä asetetaan nesteen tiheydenohjaus manuaalitilaan. Kun venttiilit ovat saavuttaneet asetetut auki- ja kiinni -rajat sekä askeleen käyntisignaali on aktiivinen, siirtyy ohjelma seuraavaan askeleeseen. Mikäli askel ei saavuta asetettuja ehtoja 90 sekunnin sisällä, menee sekvenssi vikatilaan ja ohjelmakierto pysäytetään.



Kuvio 8. Esimerkki sekvenssin askeleesta

4.3 Toimilohkot

Toimilohkojen luominen oli asiakastyön toinen pääkohta. Toimilohkot kuvastavat toimilaitetta sekä siihen tulevia ja lähteviä signaalia mahdollisten ehtojen kanssa. Toimilohkot voivat olla valmiista kirjastosta haettuja tai sitten täysin itse tehtyjä yksittäistä projektia varten. Toimilohkopiirustuksen tarkoituksena on, että henkilö, joka luo ohjelmaa logiikkakaavioiden perusteella, saa ymmärryksen toimilaitteen toiminnasta sille määrätyillä ehdoilla sekä eri lähdöistä, joita toimilaite saattaa ohjata.

Toimilohkoja ohjaavat sekvenssin eri askeleet. Nämä askeleet antavat toimilohkolle tiedon, milloin käynnistyä, pysähtyä tai mihin tilaan toimilaite asetetaan. Toimilohkoja voivat myös ohjata erilliset sille asetetut ehdot, mikäli toimilohko on asetettu automaattitilaan. Yksi ehto voi olla esimerkiksi säiliön nesteen pinnan korkeus. Tässä työssä käytetyt toimilohkot ovat moottori-, venttiili-, ohjaus-, mittaus- sekä laskuritoimilohko, joista on tarkempi kuvaus seuraavissa kappaleissa.

4.3.1 Moottorilohko

Moottorilohko sisältää kyseisen lohkon ohjaukseen liittyvät tulevat sekä lähtevät signaalit. Yleensä moottorilla on jotkin ehdot, joiden mukaan moottori käynnistyy, pysähtyy sekä menee manuaali- tai automaattiohjaukseen. Moottorin toimintakuvauksesta selviävät ehdot, joihin moottorin toiminta perustuu. Toimintakuvauksesta myös selviää, jos moottorin kuuluu pysähtyä tiettyjen ehtojen mukaisesti, esimerkiksi jos säiliön nesteen pinta on liian alhainen. Moottorin tiedot ilmaistaan käyttöliittymässä, josta näkee moottorin käyntitiedot sekä mahdolliset hälytykset.

Moottorin mahdollisia hälytyksiä ovat

- **paluutietovika**

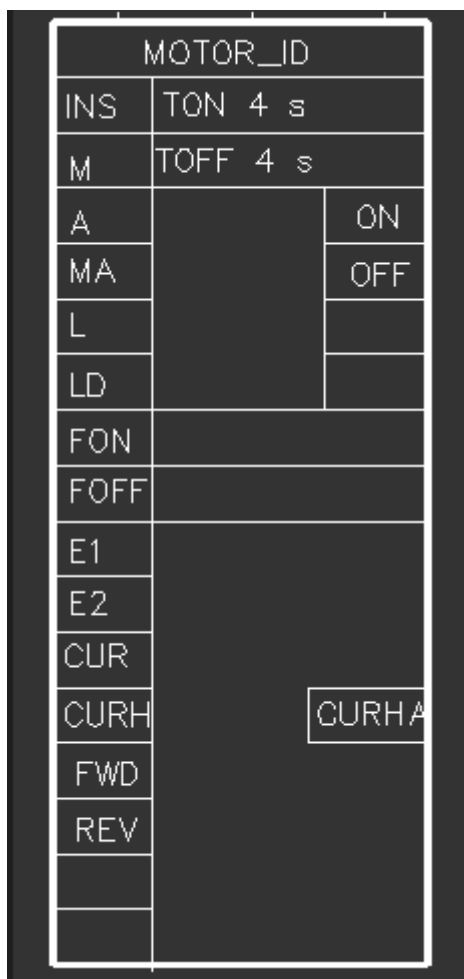
Mikäli moottorilta ei saada käyntitietoa asetettuun aikaan mennessä, menee moottori vikatilaan.

- **ylivirta**

Mikäli moottori havaitsee, että moottorin käyttöjännite nousee yli asetetun rajan, menee moottori vikatilaan.

- **ehdollinen vikatila**

Mikäli toimintakuvauksessa mainitut käynnistysehdot eivät täyty, menee moottori vikatilaan. Ehdollinen vikatilanne voi olla esimerkiksi tilanne, jossa moottori ei käynnisty säiliön käynnistysrajan ylittymisen vuoksi.



Kuvio 9. Moottorin toimilohko

Kuviossa 9 näkyy moottorin/pumpun toimilohko. Siinä olevat tulot ja lähdöt tarkoittavat seuraavaa:

- **INS** Moottorin käyntitieto
- **M** Manual, moottori käynnistyy manuaaltilassa
- **A** Auto, moottori käynnistyy automaattitilassa
- **M/A** M/A, Manual/Auto-ohjauksen valinta
- **L** Local, ohjaus asetetaan käsikäyttöön
- **FON** Forced On, moottori pakotetaan käyntiin
- **FOFF** Forced Off, moottori pakotetaan pysähtymään
- **E1** External input 1, ulkoinen tulo 1
- **E2** External input 2, ulkoinen tulo 2
- **CUR** Current, moottorin virta

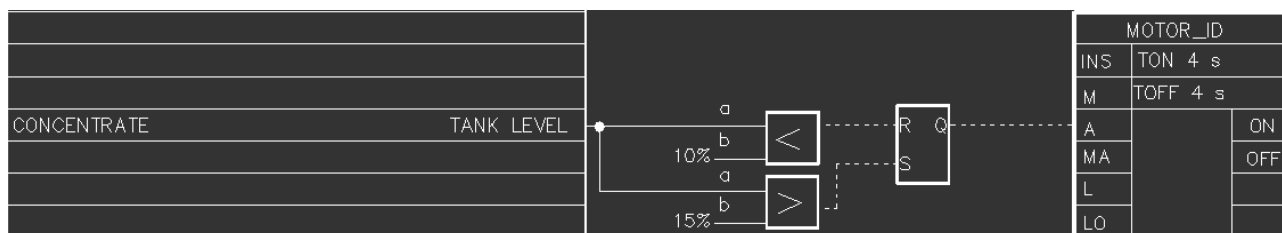
- **CURH** Current High, virran yläraja
- **CURHA** Current High Alarm, virran ylärajan hälytys
- **ON** Ohjaus asetetaan päälle
- **OFF** Ohjaus asetetaan pois päältä
- **TON** Time To On, aika, jonka jälkeen moottori käynnistyy
- **TOFF** Time To Off, aika, jonka jälkeen moottori pysähtyy.

Moottorilohkon käyttö. Tässä asiakastyössä oli useita moottorilohkoja, jotka luotiin toimintakuvauksen pohjalta. Yksi näistä toimintakuvauksen moottoreista oli konsentraattisäiliön sekoittamiseen käytetty moottori. Moottori oli kytketty lapoihin, jotka sekoittavat konsentraattisäiliössä olevaa nestettä.

Työssä käytetyn moottorilohkon toimintaehdot olivat seuraavanlaiset: Moottorin käynnistymisehtoina oli, että konsentraattisäiliön nesteen pinta on vähintään 15 % tai yli. Moottorin pysähtymisehtona oli, että konsentraattisäiliön nesteen pinta on alle 10 %. Moottori ilmoittaa virhetilasta hälytyksellä, jos konsentraattisäiliön pinta on ollut yli 15 %, mutta moottori ei ole käynnistynyt 30 sekunnin sisällä.

Kun lähdettiin luomaan konsentraattisäiliön moottorin toimilohkoa, luotiin ensimmäisenä logiikka, jonka mukaan moottori käynnistyy ja pysähtyy toimintakuvauksen pohjalta.

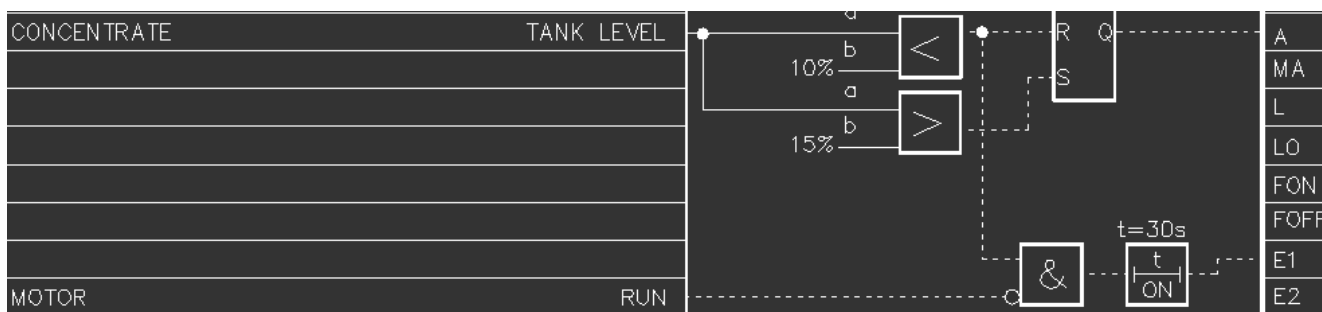
Tämä logiikka käynnistää moottorin, kun moottori on automaattilla ja säiliön nesteen pinta on yli 15 %, ja pysäyttää moottorin, kun moottori on automaattilla ja pinta on alle 10 %. Käynnistäminen ja pysäyttäminen toteutettiin SR-kytkimellä (Set/Reset). Kun kytkimen SET-tuloon (S) tuleva signaali on aktiivinen, asettaa kytkin moottorin käyntitilaan. Kun kytkimen RESET-tuloon (R) tuleva signaali on aktiivinen, kytkin nollaantuu ja moottori pysähtyy. Tämä toiminto esitetään Kuviossa 10.



Kuvio 2. Moottorin käynnistäminen ja pysäyttäminen

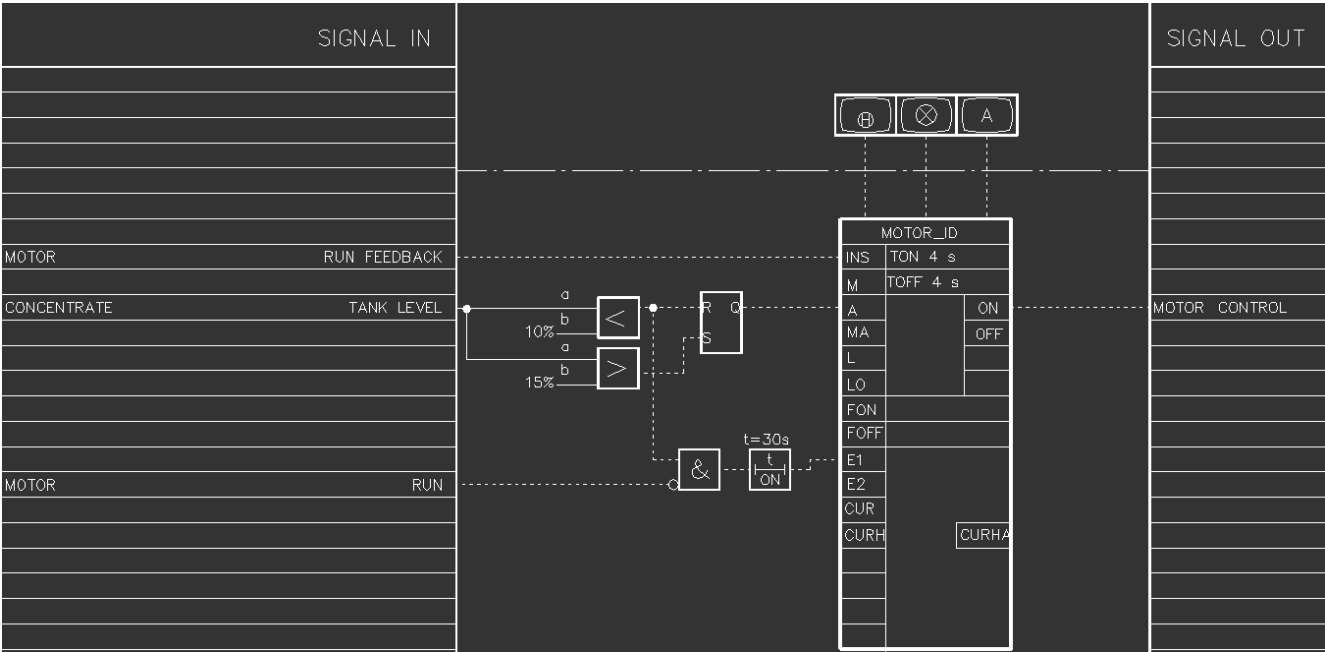
Kun saatiin luotua käynnistyminen ja pysähtyminen moottorille, siirryttiin luomaan logiikkaa, jonka perusteella moottori pysähtyy, jos ilmenee vikatila toimintakuvauksessa saneltujen ehtojen mukaisesti. Toimintakuvaukseen kirjattiin ehdot, joilla moottorin tulee pysähtyä ja ilmoittaa vikatilasta.

Toimintakuvauksessa mainittuna pysähtymisen ehtona oli, että jos konsentraattisäiliön pinnan ollessa alle 10 % ja, moottorin toiminta, ei ole käynnistynyt 30 sekunnin sisällä, moottorin toiminta pysähtyy ja moottori lähettää vikailmoituksen. Ehdot liitettiin tuloon E1. Kyseisen tulon funktion voi ohjelman tekijä määrittää itse. Tässä tapauksessa E1-tulo toimi vikatilakäskynä moottorille. Tämä toiminto on esitetty kuviossa 11.



Kuvio 3. Piiriin lisätyt pysähtymisen aiheuttavat ehdot

Kuviossa 12 on esitelty valmis logiikkakaavio moottorilohkolle. Valmiista kaaviosta selviää, millä ehdoilla lohkon ohjaama toimilaite käynnistyy ja pysähtyy, sekä ehdot, jotka aiheuttavat virhetilan. Kaaviossa näkyy liitettynä tuloon INS-signaali "MOTOR RUN FEEDBACK". Tämä tieto kuvastaa moottorin käyntitietoa. Jos moottori ei saa käyntitietoa, aiheuttaa tämä virhetilan. Myös ON-lähtöön on liitetty signaali "MOTOR CONTROL", joka kuvastaa toimilaitteelle lähtevää ohjaussignaalia.



Kuviossa 4. Moottorin valmis logiikkakaavio

4.3.2 Venttiililohko

Venttiililohkoa käytetään, kun halutaan ohjata venttiiliä automaatti- tai manuaalillassa käyttöliittymän tai ohjelman avulla. Venttiileille myös voidaan asettaa tiettyjä ehtoja, jolloin venttiili avautuu tai sulkeutuu. Näitä ehtoja voivat olla esimerkiksi, paine, lämpötila tai pinnan taso. Jos ohjattava venttiili ei saavuta auki- tai kiinnirajatietoja niille asetettujen aikojen kuluessa, venttiili menee vikatilaan ja vikatilasta lähetetään ilmoitus valvomoon.

Venttiililohkoa voidaan myös käyttää NC- sekä NO-tyyppisten venttiilien ohjaukseen. Venttiililohkoa käytetään venttiileissä, joiden ohjaus on binäärinen, eli venttiili menee pelkästään auki tai kiinni. Säädetäviä venttiilejä ohjataan ohjauslohkolla.

EV-0000		
SON	TON	25 s
SOFF	TOFF	25 s
M		ON
A		
M/A		
FON	TYPE	
FOFF	TYPE	
TON		
TOFF		

Kuvio 5. Venttiilin toimilohko

Kuviossa 13 näkyy venttiilin toimilohko. Siinä olevat tulot ja lähdöt tarkoittavat seuraavaa:

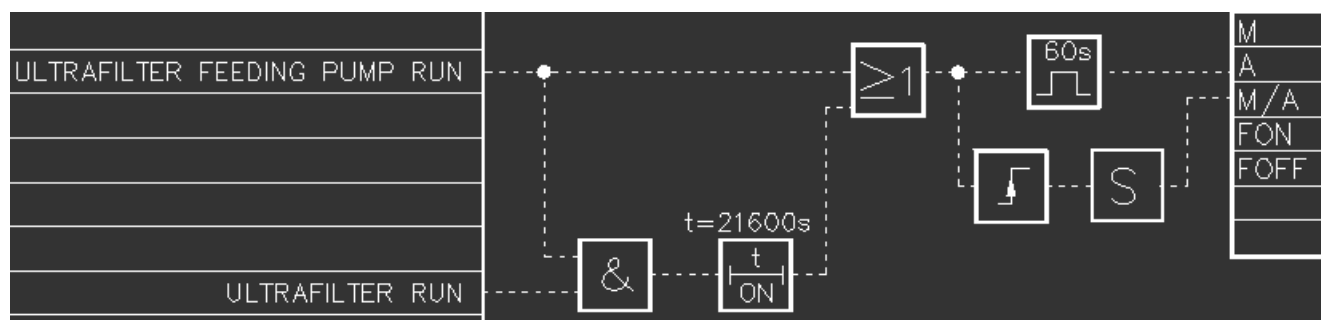
- **SON** Set On, Venttiilin auki-raja
- **SOFF** Set Off, venttiilin kiinni-raja
- **M** Manual, venttiilin ohjaus asetetaan manuaalitilaan

- **A** Automatic, venttiilin ohjaus asetetaan automaattiseen tilaan
- **M/A** Manual/Auto, ohjauksen valinta
- **FON** Force On, pakotettu käynnistys
- **FOFF** Force Off, pakotettu sammutus
- **ON** Venttiili on aktiivinen

Venttiililohkon käyttö. Työssä yksi venttiilin käyttökohde oli ultrafiltterissä käytetty ilmanpoistoverttiili. Venttiili voidaan aukaista manuaalisesti joko käyttöliittymästä tai sille asetetuilla ehdoilla. Venttiilin toimintaehdot olivat seuraavanlaiset.

Ilmanpoistoverttiiliin tulee avautua 60 sekunniksi, mikäli ultrafiltterin pumppu käynnistyy tai ultrafiltterin syöttöpumppu sekä itse ultrafiltteri ovat olleet käynnissä yli kuusi tuntia. Ilmanpoistoverttiili hälyttää vikatilanteesta käyttöliittymään, mikäli venttiili ei saavuta auki- tai kiinni-asentoa ohjelman asetuksen mukaan.

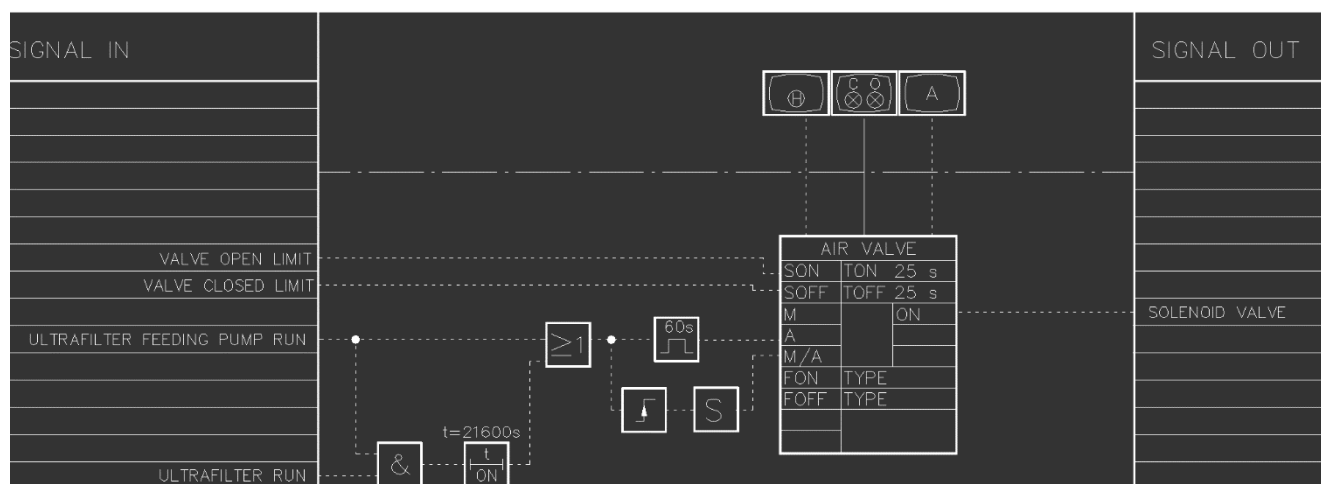
Kun lähdettiin luomaan venttiilille logiikkakaaviota, tehtiin ensimmäisenä logiikka, jonka perusteella venttiilin ohjaus asetetaan manuaali- tai automaattitilaan. Venttiili asetetaan automaattitilaan ja venttiili aukeaa, mikäli ultrafiltterin syöttöpumppu oli käynnissä. Vaihtoehtoisesti jos ultrafiltterin syöttöpumppu sekä ultrafiltteri on ollut käynnissä yli kuusi tuntia, asetetaan venttiilin ohjaus automaattitilaan ja venttiili aukaistaan. Venttiili sulkeutuu, mikäli jokin edellä mainituista ehdoista ei täyty. Tämä toiminto esitetään kuviossa 14.



Kuvio 14. Venttiilin avautuminen ehdoilla

Kuviossa 15 on esitelty valmis logiikkakaavio ultrafiltterin ilmanpoistoverttiilille. Valmiista kaaviosta selviää ehdot, joilla venttiili avautuu sekä sulkeutuu. Valmiista kaaviosta myös näkee tulot SON sekä SOFF, joihin tulee signaalit VALVE OPEN LIMIT sekä VALVE

CLOSED LIMIT, jotka kuvastavat venttiilin auki- ja kiinnirajoja. Mikäli venttiilille on annettu käsky avautua tai sulkeutua ja venttiili ei saavuta vaadittavaa rajaa tälle asetettuun aikaan mennessä, ilmoittaa lohko virhetilasta. ON-lähtöön liitetty signaali SOLENOID VALVE kuvastaa toimilaitteelle lähtevää ohjaussignaalia.



Kuvio 6. Valmis venttiilin logiikkakaavio

4.3.3 Ohjauslohko

Ohjauslohko säättää toimilaitteiden arvoja, esim. moottorin nopeuksia ja säätöventtiilin avautumista. Ohjauslohko on mahdollista asettaa sekä automaattiohjaukseen että manuaaliseen ohjaukseen. Ohjauksen ollessa manuaalillassa voi käyttäjä asettaa ohjausarvot manuaalisesti. Ohjaimen toimilohkoon voidaan asettaa ylä- ja alarajat, joiden perusteella voidaan lähettää hälytys tai huomautus valvomoon. Asetetut hälytykset voidaan myös piilottaa, kun tietyt ehdot täyttyvät, esimerkiksi kun jokin moottori ei ole toiminnassa. Lähtevää ohjaussignaalia on mahdollista lineaarisesti supistaa, jolloin nopeasti tapahtuvat muutokset voidaan vaimentaa ja saadaan tasaisempi toiminta ohjattavalle toimilaitteelle.

LIC—0000			
ME	CONDIR 1		
SP1		CON	
SP2		SPA	
ISP		E	
MA			
FM	TYPE		
FC	TYPE		
FCIN			
MEHH			MEHHA
MEH			MEHA
MEL			MELA
MELL			MELLA
DEVL			DEVA
LMASK			
HMASK			
DMASK			
DTIME			

Kuvio 7. Ohjauksen toimilohko

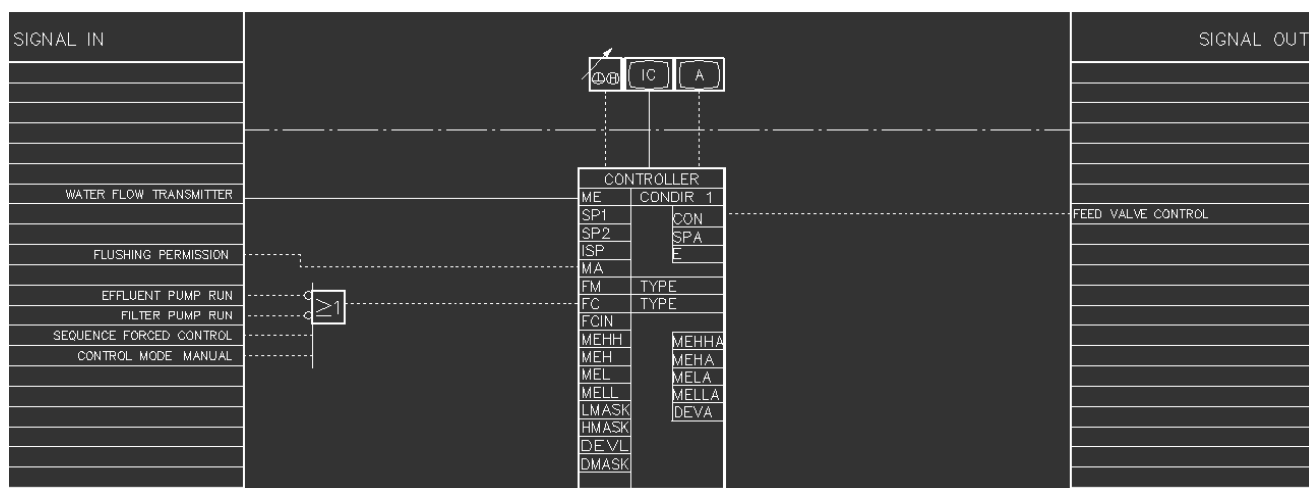
Kuviossa 16 näkyy ohjaustoimilohko. Siinä olevat tulot ja lähdöt tarkoittavat seuraavaa:

- **ME** Measured Value Input
- **SP1** Setpoint 1 local
- **SP2** Setpoint 2 remote
- **ISP** Setpoint selection
- **MA** Manual / Auto control mode
- **FM** Forced Manual Control
- **FC** Forced Control
- **FCIN** Forced Control Input
- **MEHH** Measured value Higher High alarm limit
- **MEH** Measured value High alarm limit
- **MEL** Measured value Low alarm limit
- **MELL** Measured value Lower Low alarm limit
- **DEVL** Deviation Limit +- between measured value input and active setpoint
- **LMASK** Low alarm mask
- **HMASK** High alarm mask
- **DMASK** Deviation alarm mask
- **CON** Control
- **SPA** Active setpoint value
- **MEHHA** Measured value higher high limit alarm
- **MEHA** Measured value high limit alarm
- **MELA** Measured value low limit alarm
- **MELLA** Measured value lower low limit alarm
- **DEVA** Deviation alarm
- **DTIME** Deviation time until deviation alarm is activated.

Ohjauslohkon käyttö. Työssä yksi käyttökohteista ohjauslohkolle oli jäteveden virtausta ohjaavan venttiilin säätely. Ohjauslohkon toimintaehdot olivat seuraavanlaiset.

Käyttäjä voi asettaa ohjatulle venttiilille asetusrvon valvomosta, jota ohjauslohkon ohjaama toimilaite pyrkii seuraamaan mahdollisimman tarkasti. Ohjauslohko menee automaattitilaan, mikäli käyttäjä asettaa ohjauksen säätöluvan aktiiviseksi valvomosta. Ohjauslohko sulkee venttiilin, mikäli jätevesipumppu ei ole käynnissä, jäteveden filteröintipumppu ei ole käynnissä, ohjauslohko on manuaalitilassa tai jätevesiohjauksen sekvenssi sammuttaa pumpun. Ohjauslohko hälyttää, mikäli virtauksen mittauksessa ilmenee virhe.

Kuviossa 17 on esitelty ohjauslohko yllä mainitulla toimintaperiaatteella. Kuviossa esitetyssä toimilohkossa pysäytyksen aiheuttavat ehdot on kiinnitetty FC-tuloon. Kun tulo FC on aktiivinen, asetetaan ohjauslohko pakotetun ohjauksen alaisuuteen. Kun lohko on pakotetussa ohjaustilassa, on lohko ohjelmoitu sulkemaan virtausta säätelevä venttiili.



Kuvio 8: Esimerkkiteutus ohjauslohkon käytöstä

4.3.4 Mittauslohko

Mittauslohkoa käytetään silloin, kun mitattavalla arvolla ei tarvitse olla erillistä ohjausta. Mittausarvo yleensä toimii ehtona toimilaitteen käynnistymiselle tai sekvenssin askeleen etenemiselle. Mittauslohkoa on mahdollista mukauttaa mitattavaan kohteeseen ja laajentaa näin mitattavien kohtien määrää. Mitattavaa arvoa voidaan suodattaa tilanteissa, joissa mitattava arvo saattaa heittelehtiä vaikeuttaen mitattavan arvon tulkitsemista. Mitattaville arvoille voidaan asettaa rajat, joiden ylittyminen tai alittuminen lähettää hälytyksen tai varoituksen valvomoon. Lähetettävät hälytykset voidaan myös piilottaa, mikäli jokin ehto ei täyty ja hälytys olisi tällöin aiheeton. Mitattavia kohteita voivat olla esimerkiksi paine, lämpötila tai säiliön pinnan taso. Mitattava arvo voi myös perustua laskennalliseen kaavaan, jonka tulos viedään mittauslohkoon.

MEASURE	
AV	
HH	MEHH
H	MEH
L	MEL
LL	MELL
HMASK	
LMASK	

Kuvio 9. Mittauksen toimilohko

Kuviossa 18 näkyy mittauslohkko. Siinä olevat tulot ja lähdöt tarkoittavat seuraavaa:

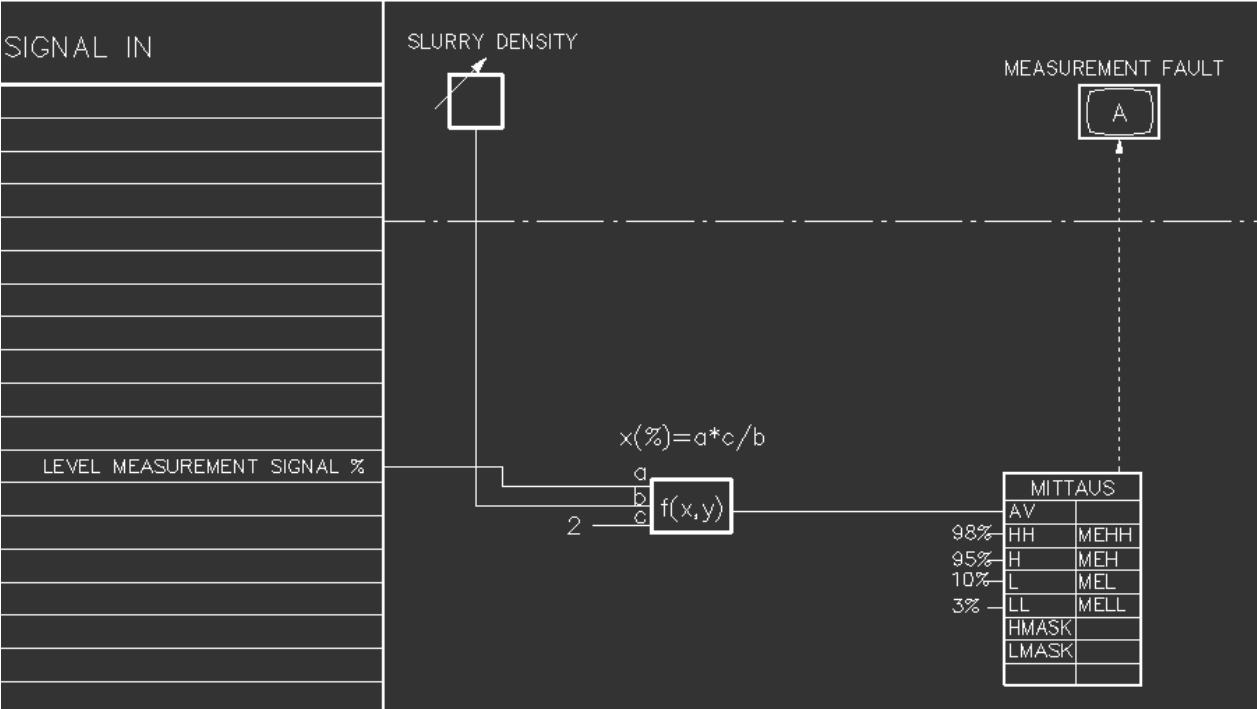
- **AV** Value of the measured analog signal
- **HH** Value of higher high limit alarm
- **H** Value of high limit alarm
- **L** Value of lower limit alarm
- **LL** Value of lower low limit alarm
- **HMASK** HH, H alarm mask
- **LMASK** LL, L alarm mask
- **MEHH** Measured value Higher High limit alarm
- **MEH** Measured value High limit alarm
- **MEL** Measured value Low limit alarm
- **MELL** Measured value Lower Low limit alarm.

Mittauslohkon käyttö. Yksi mittauslohkon käyttökohde tässä työssä oli nestesäiliön pinnan mittaus. Mitattava arvo perustui kaavaan, jossa huomioitiin nesteeseen sekoitettavan lisäaineen tiheys. Lisäaineen tiheys syötettiin näytöltä käsin. Mittauslohkon luominen oli yksinkertaista, koska ainoa tieto joka mittauslohkoon tarvittiin, oli prosentti, jolla pinnantason hälytykset aktivoituvat. Mittauslohkolta ainoa lähtevä tieto oli ruudulle viety hälytystieto, mikäli pinnantason hälytys oli aktiivinen.

Mittausarvo laskettiin kaavalla $x (\%) = a * c / b$, jossa

- **a** = säiliön pintamittaus
- **b** = näytöltä syötetty lisäaineen tiheys arvo
- **c** = mittausanturin kalibroitukorjaus, jonka arvo oli 2.

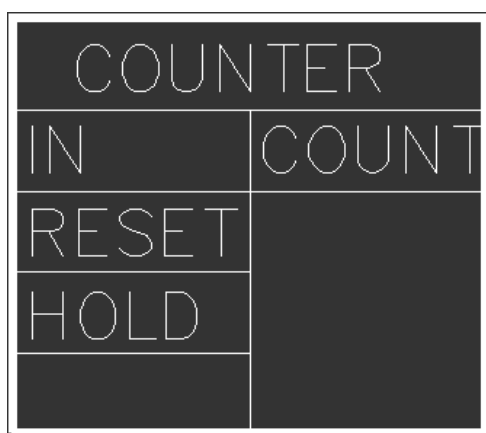
Kuviossa 19 on esitetty valmis logiikkakaavio, jossa käytetään mittauslohkoa. Kuviossa esitettyyn laskennalliseen funktiolohkoon tuodaan näytöltä käyttäjän asettama lisäaineen tiheysarvo. Funktiolohkoon myös tuodaan säiliön pintamittaus ja valmiiksi asetettu kalibroitikorjauksen arvo.



Kuvio 10. Valmis mittauksen logiikkakaavio

4.3.5 Laskuri

Laskurilohkoa käytetään, kun halutaan laskea tietyn signaalin ilmoittamaa toimintaa, esimerkiksi venttiilin aukeamiskerrat. Laskurilohkolla laskettua arvoa voidaan verrata toiseen arvoon ja näin voidaan luoda ehtoja muille toimilohkoille. Laskurilohkolle ei yleensä luoda erillistä kaaviota, vaan se voidaan integroida lohkoa käyttävään kaavioon.



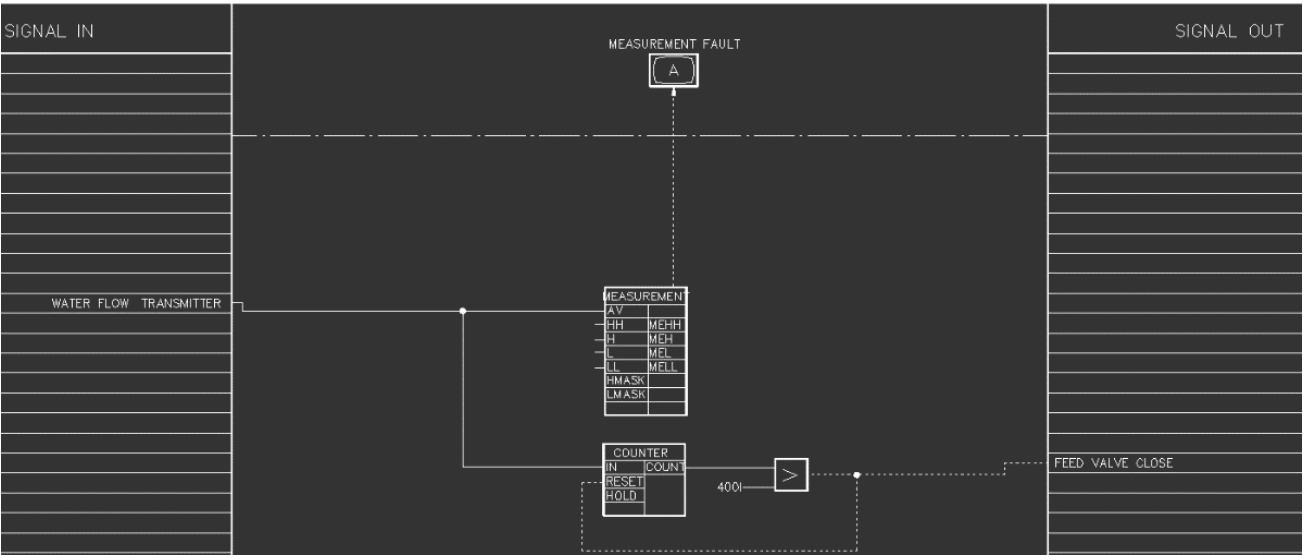
Kuvio 20. Laskurin toimilohko

Kuviossa 20 näkyy laskurin toimilohko. Siinä olevat tulot ja lähdöt tarkoittavat seuraavaa:

- **IN** Tuleva signaali, jonka arvoa lasketaan
- **RESET** Nollaa laskurin, kun tulo on aktiivinen
- **HOLD** Kun tulo on aktiivinen, laskettu arvo pysyy muuttumattomana
- **COUNT** Laskettu arvo.

Laskurilohkon käyttö. Esimerkkikäyttökohde laskentalohkolle oli putkistossa virranneen nesteen määrän mittaus. Mitattava arvo viedään laskurille, ja kun mitattava arvo yrittää 400 litran rajan, suljetaan veden syöttöventtiili. Laskuri nollataan, kun ylitetään 400 litran raja.

Tämä toiminto on esitelty kuviossa 21.



Kuvio 21. Esimerkki laskurin käytöstä

4.4 Työn viimeistely

Lähtötilanteessa asiakkaalla oli PI-kaaviot, toimintakuvaukset sekä IO-listat, mutta asiakkaalta puuttuivat logiikkakaaviot, joiden pohjalta asiakas voisi helpommin luoda ohjelman automatisoitua prosessia varten.

Työn lopputuloksena asiakkaalla oli useasivuinen listaus logiikkakaavioista, jotka olivat graafinen kuvaus prosessin toiminnasta sekä sen toimilaitteiden yksittäisistä toiminnoista prosessin kuluessa. Tämä moniosainen kokonaisuus sisälsi sekvenssien askeleiden logiikkakaaviot ja kaikkien toimilaitteiden logiikkakaaviot, joissa oli kuvattu toimilaitteen toimilohko toimintaperiaatteineen ja -ehtoineen. Asiakkaan listauksessa oli useita toimilaitteita, joilla kuitenkin oli sama toimilohko, kuten esimerkiksi moottoritoimilaitteilla oli yhteinen moottorilohko.

Kun valmiit logiikkakaaviot on yhdistetty yhdeksi kokonaisuudeksi, luodaan kyseisistä kaavioista listaus pdf-muodossa, joka lähetetään asiakkaalle hyväksyntää varten. Kun asiakas on hyväksynyt listauksen mahdollisten muutoksien jälkeen, voi asiakas tulostaa logiikkakaaviot paperisena versiona mahdollista arkistointia varten. Näihin valmiisiin logiikkakaavioihin voidaan palata, kun halutaan selvittää jonkin toimilaitteen toimintaperiaate tai sekvenssin toimintatapa mahdollisia tulevia muutoksia tai huoltoja varten.

Tekstinkäsittely

Tekstinkäsittelyohjelmalla voidaan korjata logiikkakaavioista toistuvasti puuttuvat tekstikentät tai signaalityypit. Esimerkiksi kun valmiita logiikkakaavioita on yli 100 kappaletta ja jokaiseen joudutaan lisäämään asiakkaan nimi tai kaavion käyttökohde, voidaan työ tms. toteuttaa helposti yllä mainitulla tekstinkäsittelyohjelmalla. Tämä tekstin korjaus jätettiin jokaisen logiikkakaaviokokonaisuuden loppuvaiheeseen, koska prioriteettina oli itse ohjelman tai toimilaitteen toiminnan visualisointi logiikkakaavioon ja se, että voitaisiin työstää kaikki logiikkakaaviot kerralla.

5 TULOKSET

Työn tuloksena saatiin asiakkaan toimintakuvauksen pohjalta luotua logiikkakaaviokokonaisuus. Nämä logiikkakaaviot sisälsivät sekvenssikaaviot sekä toimilohkokaaviot, joiden pohjalta pystytään luomaan automaatio-ohjelma haluttua IEC 61131-3 -standardissa määritettyä ohjelmointimenetelmää käyttäen. Logiikkakaavioita lähetettiin asiakkaalle tarkistukseen aina, kun saatiin määrätyn alueen logiikkakaaviot valmiiksi. Näillä tarkistuskierroksilla asiakkaan oli mahdollista ilmoittaa mahdollisista halutuista muutoksista logiikkakaavioihin sekä logiikkakaavioihin. Samalla varmistettiin että asiakas on yleisesti tyytyväinen logiikkakaavioiden laatuun. Kun asiakas ilmaisi olevansa tyytyväinen luotuihin logiikkakaavioihin, voitiin merkitä tietylle alueelle luodut logiikkakaaviot valmiiksi.

Kun asiakkaalle tehtiin logiikkakaaviokokonaisuus, huomattiin työprosessin aikana, että logiikkakaavioiden luominen saattaa olla uudelle työntekijälle haasteellista ja aikaa vievää ilman asianmukaista perehdytystä ja ohjeistusta. Myös kaavioissa ilmenevät elementit, esimerkiksi objektit, saattavat olla tuntemattomia uudelle työntekijälle.

Tämän asiakastyön aikana tehdyn tutkimuksen pohjalta luotiin työpaikalle yleisen tason toimintatapaohjeistus logiikkakaavioiden luomisen tueksi. Asiasta keskusteltiin esimiesten kanssa ja todettiin, että ohjeistuksen tuli olla riittävän kattava, mutta samalla selkeän yksinkertainen ja siihen piti voida lisätä osia tai komponentteja myöhemminkin, koska jokainen asiakasprojekti voi vaatia pieniä muutoksia. Ohjeistuksessa on esitelty keskeisimmät toimilohkot, joita yleisimmin käytetään prosessiteollisuuden logiikkakaavioissa. Näistä toimilohkoista on ohjeistuksessa kuvattu niiden toimintatavat sekä selostettu lohkojen tulojen ja lähtöjen toiminta sekä käyttötarkoitus. Jos uudella työntekijällä tulee epäselvyyksiä lohkojen toiminnan suhteen, voi hän aina palata ohjeistukseen ja hakea siitä esimerkkiä lohkojen toiminnasta ja käyttötavasta. Ohjeistuksessa on myös esitelty tavat, joilla voidaan luoda esimerkillinen toteutus sekvenssikaaviosta vaiheittain. Ohjeistuksessa on esimerkki toimintakuvauksesta sekä tämän kuvauksen pohjalta luodusta sekvenssikaaviosta, jonka jokaisesta kohdasta on kuvallinen viittaus valmiiseen sekvenssikaavioon.

Toimintatapaohjeistuksen etuna on, että jatkossa logiikkakaavioiden luominen on yksinkertaisempaa sekä sujuvampaa uuden työntekijän kannalta ja että niiden tekijät noudattavat

samaa toimintatapaa ja käyttävät samoja objekteja. Noudattamalla yhteistä toimintatapaa voidaan taata tasainen laatu kaavioiden luomisessa.

6 YHTEENVETO JA POHDINTA

Tämän opinnäytetyön tarkoituksena oli esittää logiikkakaavioiden luomisen teoriapohja, niiden sisältö ja käyttökohteet. Logiikkakaavioiden tarkoitus on antaa kolmannen osapuolen ohjelmoijalle malli, jonka pohjalta tämä voi alkaa luomaan varsinaista automaatio-ohjelmaa IEC 61131-3 -standardissa lueteltuja ohjelmointitapoja käyttäen.

Kun logiikkakaavioita lähdettiin luomaan, jouduttiin uusille työntekijöille perehdyttämään henkilökohtaisesti logiikkakaavioiden tarkoitus sekä esittämään logiikkakaavioiden luomistavat. Tämän takia varsinaisen asiakastyön aikana tutkittiin, mitä asioita toimintatapaohjeistuksen tulisi sisältää.

Hyvän toimintatapaohjeen tulisi olla sellainen, että se kuvaa, millaiseen automaatioprosessiin toimintatapaohjeistus toteutetaan, ketkä ovat ohjeen käyttäjät, kuinka hyvin kyseessä oleva käyttäjä tuntee prosessin ja minkälainen ohjeistuksen esitystavan tulisi olla, että kaikenlaiset käyttäjät saavat sen avulla samanlaisen lopputuloksen. Omassa tutkimustyössäni seurasin näitä periaatteita, ja toimintatapaohjetta alettiin luoda logiikkakaavioista sekä niiden keskeisimmistä asioista. Tämä toimintatapaohje sisältää perehdytyksen logiikkakaavioihin ja esityksen erilaisista logiikkakaavioista sekä näiden sisällöstä. Ohjeistuksen avulla uusi työntekijä voi alkaa työstämään logiikkakaavioita ja samalla perehtyä ohjeistukseen, mikäli tällä työntekijällä tulee epäselvyyksiä logiikkakaavion luomisen suhteen.

Toimintatapaohje tulee ainoastaan yrityksen sisäiseen käyttöön, koska esimerkkikaaviot, joita ohjeistuksessa saatetaan käyttää, voi sisältää luottamuksellista tietoa. Tämän takia opinnäytetyössä ei ole esitelty varsinaisen toimintatapaohjeistuksen sisältöä.

Työtä varten luotu objektiokirjasto tallennettiin myöhempiä logiikkakaavioprojekteja varten. Koska kirjaston kuvakkeet ovat .dwg-muodossa, voidaan kirjastoa käyttää muissakin CAD-ohjelmissa. Kirjastoa tullaan laajentamaan, mikäli tulevaisuudessa projekteissa ilmenee uusia funktioita tai objekteja, jotka voisivat olla hyödyksi yleisellä tasolla. Objektiokirjastosta on myös mahdollista luoda erilaisia versioita tiettyä asiakasryhmää tai työtä varten.

LÄHTEET

Automationforum. (i.a.). *I/O List*. Haettu 7.2.2023. <https://automationforum.co/i-o-list/>

Batch Attribute Editor. (i.a.). Batch attribute editor. Haettu 7.2.2023. <http://www.le-mac.com/batte.html>

Battikha, N. E. (2017) *The condensed handbook of measurement and control*, (4. P.) ISA

Bricsys (i.a.). What is BricsCAD?. Haettu 7.2.2023. <https://www.bricsys.com/fi-fi>

Kalani, G (2002) *Industrial Process Control: Advances and Applications*. Gulf Professional Publishing.

Hanssen, D. H. (2015). *Programmable logic controllers: a practical approach to IEC 61131-3 using CODESYS*.

International Organization for Standardization (ISO). (2004). *Diagrams for the chemical and petrochemical industry. Part 1: Specification of diagrams (ISO 10628-1:2014)*.

International Organization for Standardization (ISO). (2015). *Kaavioissa käytettävät piirrosmerkit. Osa 5: Mittaus- ja ohjauslaitteet (ISO 14617-5:2015)*.

Kuphaldt, T. (26.2.2022). *Lessons In Industrial Instrumentation*. [Valokuva].

PCS-Engineering (i.a.). *Prosessiteollisuuden laadukas ja luotettava kumppani – Reliable Partner*. Haettu 28.1.2023. <https://pcs-engineering.fi>

Realpars.com (i.a.). *What is a functional design specification (FDS)?* Haettu 31.1.2023 <https://realpars.com/fds/>