

Esa Laakso

# Fidelix Genius –säätimen automaattinen testausjärjestelmä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Automaatiotekniikka

Insinöörityö

9.4.2015

Tekijä(t)	Esa Laakso
Otsikko	Fidelix Genius –säätimen automaattinen testausjärjestelmä
Sivumäärä	31 sivua + 1 liitettä
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Automaatiotekniikka
Suuntautumisvaihtoehto	Tietoliikennetekniikka
Ohjaaja(t)	Opettaja Markku Inkinen Tuotekehitys diplomi-insinööri Tero Ritari
<p>Tämän insinööryön tarkoituksen oli tuottaa automaattinen kaukolämmönsäätimen testausjärjestelmä Fidelix Oy:n tuotteelle, joka on nimeltään Genius -kaukolämmönsäädin. Työn kahtena osana oli tuottaa suunnitelma toiminnalle ja toteutus laitteistolle. Tarkoituksena oli saada vakio testausjärjestelmä, jota käytetään aina kun säätimen ohjelmaan tehdään muutoksia.</p> <p>Testauslaitteistona käytettiin Fidelix Oy:n rakennusautomaatioon käytettäviä tuotteita ja se koostuu ohjelmoitavasta logiikasta sekä muutamasta I/O-moduuleista. Työ toteutettiin kokonaan Fidelix Oy:n tiloissa ja yrityksen omilla laitteistoilla. Tässä työssä on käytetty logiikan toteuttamiseen Fidelix:n omia ohjelmallisia työkaluja sekä Infoteam Oy:n OpenPCS ohjelmointityökalua.</p> <p>Testausjärjestelmää toteutettaessa on pyritty myös tuottamaan käytettäviä ohjelmallisia työkaluja, kuten uusia funktiota tai ohjelmia tulevaisuuden käyttöön, sekä on pyritty pitämään ohjelma helposti muokattavana jatkoa varten. Työssä tarvittavat tiedot kerättiin Fidelix Oy:n henkilökunnalta, pääasiassa tuotekehitysosastolta.</p>	
Avainsanat	testausjärjestelmä, rakennusautomaatio, kaukolämmönsäädin

Author(s)	First name Last name
Title	Fidelix Genius –controllers automatic testing system
Number of Pages	31 pages + 1 appendices
Degree	Bachelor of Engineering
Degree Programme	Automation Engineering
Specialisation option	Telecommunication Engineering
Instructor(s)	Teacher Markku Inkinen R&D department, Master of Engineering Tero Ritari
<p>The purpose of this project was to create an automatic testing system for a product of Fidelix Ltd. named Genius which works as a District Heating Controller. The project comprised of two parts, creating a plan for a testing system and implementing the plan. The main purpose of this project was to create a testing protocol that is administered every time a new software version of the controller is released.</p> <p>The testing system was made with Fidelix's own devices that are normally used in building automation and it consists of a Programmable Logic Controller and few I/O-modules The entire project was done in Fidelix's own premises and with their own devices. The programming tools used for the programmable logic during the production of the project consisted of Fidelix's own software tools and Infoteam Ltd's programming software.</p> <p>Another goal for this project was to create new software tools for use in programming, such as new functions or programs for future use, as well as make the program easily changeable if required. The information required for this project was collected mainly from Fidelix's staff, mostly the R&amp;D department.</p>	
Keywords	testing system, building automation, district heating controller

## Sisällys

### Lyhenteet

1	Johdanto	1
2	Kiinteistö- ja rakennusautomaatio yleisesti	1
2.1	Kiinteistön kaukolämmön lämmönjako	2
3	Fidelix Oy	2
4	Genius säädin	3
4.1	Yleisesti	3
4.2	Käyttökohteita	4
5	Genius-säätimen ominaisuuksia	6
5.1	Mittaukset	6
5.2	Säätöventtiilit	6
5.3	Pumppujen ohjaus	6
5.4	Hälytykset	7
6	Testauslaitteisto ja kytkennät	7
6.1	Logiikkaohjain Fx-2025A	8
6.2	IO-modulit ja kytkentäkaaviot	8
6.2.1	COMBI-36 ja AO-8	8
6.2.2	Multi-24-moduuli ja Genius-säädin	9
6.3	Kytkenät testerissä	9
7	Käytetyt ohjelmatyökalut	10
7.1	Infoteam OpenPCS	10
7.2	Fidelix HTMLeditor	11
8	IEC 61311-3 –ohjelmointistandardi	12
8.1	Structured Text (ST) –ohjelmointikieli	13
8.2	Functionblock	13
9	Testivaiheet ja niiden ohjelmoinnin toteuttaminen	14

9.1	Normaalisimulaatio	15
9.2	Oletusasetuksien tarkistus	15
9.3	Ulkoanturin hajoaminen	15
9.4	Verkostojen lämpötilojen säätö	16
9.5	Verkostojen menoanturin hajoaminen	17
9.6	Verkostojen pumpun ristiriita	17
9.7	Lämmitysverkostojen paineiden kärkitieto	18
9.8	Verkoston pumpun kärkitieto	18
9.9	Antureiden muunnostaulukot	19
9.10	Huonekompensointi	20
10	Käyttöliittymä	20
10.1	Käyttöliittymä yleisesti	20
10.2	Painikkeet	21
10.3	Tilatekstit	22
10.4	Muut asiat käyttöliittymässä,	22
11	Testilaitteiston testaus	23
11.1	Testivaiheiden suoritus normaalisti	23
11.1.1	Oletusasetuksien tarkistuksen testaus	23
11.1.2	Verkostojen säätöjen testaus	24
11.1.3	Verkostojen menoantureiden rikkoutumisen testaus	24
11.1.4	Pumppujen ristiriidat	24
11.1.5	Paineen kärkitiedot ja pumppujen kärkitiedot	24
11.1.6	Muunnostaulukkojen testaus	24
11.1.7	Huonekompensoinnin testaus	25
11.2	Testauksen häirintä ja sen tulkinta	25
11.2.1	Oletusarvojen testaus	25
11.2.2	Ulkoanturin rikkoutuminen	25
11.2.3	Verkostojen säätöjen testaus	26
11.2.4	Verkostojen menoanturien rikkoutuminen	26
11.2.5	Pumppujen ristiriidat	26
11.2.6	Paineiden kärkitietojen testaus	27
11.2.7	Pumppujen kärkitietojen testaus	27
11.2.8	Antureiden muunnostaulukkojen testaus	27
11.2.9	Huonekompensoinnin testaus	27
11.3	Löydetyt viat testerin normaalisuorituksissa	28
12	Lopputulos	28

12.1	Testivaiheiden arviointi ja mahdolliset parannukset	29
12.1.1	Normaalisimulaatio	29
12.1.2	Oletusasetuksien tarkistus	29
12.1.3	Verkostojen säädöt sekä menoantureiden rikkoutuminen	29
12.1.4	Muunnostaulukkojen testaus	30
	Lähteet	31
	Liitteet	
	Liite 1. Testausvaiheiden toteutuksesta malli IEC -muodossa	

## Lyhenteet

PLC	Programmable Logic Controller, eli ohjelmoitava logiikka, jota käytetään automaation toteuttamiseen
IO	Input ja Output, eli sisään- ja ulostulot, käytännössä mittaukset ja ohjaukset. Yleinen käsite digitaalisille ja analogisille sisään- ja ulostuloille
DI	Digital Input, eli digitaalinen sisääntulo tai tilatieto siitä, onko jokin päällä tai pois.
DO	Digital Output, eli digitaalinen ulostulo tai ohjaus päälle tai pois.
AI	Analog Input, eli analoginen sisääntulo, eli mittaustieto jatkuvalta alueelta eikä pelkästään tilatieto.
AO	Analog Output, eli analoginen ulostulo, eli ohjaus, joka voi olla jatkuvalta alueella eikä pelkästään päällä tai pois.
IEC	Ohjelmointistandardi automaatiojärjestelmille, tarkoittaa tässä tapauksessa IEC 61131-3 -standardia
ST	Structured Text -ohjelmointikieli, määritelty IEC 61131-3 standardissa
LV	Lämminvesi tai käyttövesi lämmönjaossa
LVK	Lämminvesikierto, käyttöveden paluuvesi
KV	Kylmävesi, kylmänveden tulo jakeluverkostosta

## 1 Johdanto

Insinöörityön tavoitteena oli tehdä testilaitteisto Fidelix Oy:n Genius-säätimelle. Fidelix Oy:n tarkoitus oli saada ennakoiva testaus, jolla varmistetaan säätimen toimivuus jo ennen säätimen käyttöönottoa. Testilaitteistolla on tarkoitus varmistaa Genius-säätimen tarkoituksen mukainen toiminta, kun säätimestä ollaan julkaisemassa uusi versio.

Genius-säädin on Fidelix Oy:n oma tuote, joka on tarkoitettu lämmönjaon säätöön kiinteistöissä. Genius-säädintä käytetään kiinteistöissä, joissa on kaksi tai kolme lämmönjakopiiriä. Testilaitteisto on tehty kokonaan Fidelix Oy:n omista tuotteista.

Testilaitteisto simuloi oikeaa säätötilannetta säätimelle ja tulkitsee tilannetta lukemalla vaaditut tiedot säätimeltä. Testilaitteiston käyttäjälle on tehty käyttöliittymä, josta käyttäjä voi suorittaa koko testin askelmaisesti vaiheissa tai tietyn yksittäisen halutun vaiheen. Testin käynnistyessä tai testivaihetta suorittaessa PLC jatkaa simulointia, mutta lisää simulaation tilanteen, jonka jälkeen PLC seuraa tekeekö säädin niin kuin sen pitäisi.

## 2 Kiinteistö- ja rakennusautomaatio yleisesti

Kiinteistö- ja rakennusautomaatiolla on tarkoitus automatisoida LVI-, ilmastointi- ja sähköjärjestelmiä kiinteistöissä. 1900-luvun alkupuolella kiinteistöjen lämpöä ja painetta ohjattiin manuaalisesti tulkitsemalla mittareita ja käsin säätämällä esimerkiksi virtauksia [1, s. 23]. Kiinteistöautomaation on tarkoitus kehittyä jatkuvasti toimivammaksi ja energiatehokkaammaksi järjestelmäksi kiristyneiden vaatimusten mukaisesti [1, s. 49].

Automaatiotekniikan avulla kiinteistöjen lämmityksen, ilmastoinnin ja sähköisten toimilaitteiden käyttö voidaan automatisoida ja varmistaa niiden oikea toiminta. Internetin kehityksen ansiosta kiinteistöautomaatio on kehittynyt valvottaviksi sekä ohjattavaksi etäältä [1, s. 25]. Olennainen osa järjestelmän valvontaa on tieto vikatilanteista, joita kutsutaan hälytyksiksi. Hälytykset ilmoittavat järjestelmän valvojalle, jos esimerkiksi ilmastointilaitteen puhallin on rikki [1, s. 225].



## 2.1 Kiinteistön kaukolämmön lämmönjako

Kiinteistön kaukolämmön lämmönjako tarkoittaa yksinkertaisesti sitä, että kiinteistössä on lämmönjakokeskus, jolla säädetään esimerkiksi patteriverkoston menevän veden lämpötilaa. Useimmat suomalaiset kiinteistöt käyttävät kaukolämpöä [2, s. 1]. Kaukolämpö siirretään vesikiertona kaukolämpövoimalalta kiinteistöön, josta se lähes aina siirretään lämmönsiirtimen kautta lämmitysverkkoon [2, s. 271]. Genius-säädintä käytetään vain kaukolämmöllä lämmitettävissä kiinteistöissä.

Kaukolämmön lämmönjaon tärkeimmät osat ovat ensiöpuoli ja toisiopuoli. Ensiöpuolella tarkoitetaan kaukolämmön kiertoa ja toisiopuolella lämmitysverkoston kiertovettä, esimerkiksi patteriverkoston. Ensiö- ja toisiopuolta yhdistää lähes aina lämmönsiirrin, jotta kaukolämpöverkoston ei vaikuta lämmitysverkoston vuodot tai syöpymiset, eikä lämmitysverkoston tarvitse tehdä kestävään kaukolämpöverkoston paineita [2, s. 271].

Kaukolämmöstä siirtyvää lämpöenergiaa säädetään muuttamalla kaukolämmön virtausta lämmönsiirtimen läpi. Virtausta muutetaan sulkemalla ja avaamalla kaukolämpöventtiileitä.

## 3 Fidelix Oy

Fidelix Oy:n päätoimi on toteuttaa suunnittelijan suunnitteleman järjestelmän automaatiourakka. Urakoinnin lisäksi yritys on kiinteistöautomaation laitevalmistaja. Fidelix Oy:n tuotteisiin kuuluu ohjauslogiikoita, -keskuksia, mittaus- sekä ohjausmoduuleita, säätölaitteita, käyttöpaneeleita sekä antureita.

## 4 Genius säädin

### 4.1 Yleisesti

Genius-säädin on kehitetty lämmönjaon säätämiseen. Säädin on Fidelix Oy:n kehittämä laite ja se pohjautuu yrityksen toiseen tuotteeseen, joka on nimeltään Multi-24. Multi-24 on IEC-ohjelmoitava tuote, joka on kehitetty ohjelmoitavaksi säätimeksi ja sitä käytetään yleisesti esimerkiksi huonesäätimenä (Kuva 1). Multi-24 sisältää siis ohjelman, joka ohjaa kaukolämmönsäädintä. Tämä ohjelmoitu versio Multi-24:stä on tuotteistettu Genius-lämmönjakosäätimeksi. Säätimen mukana tulee myös 3,5” kosketusnäyttö, josta näkee lämmönjakokeskuksen tärkeimmät tiedot, kuten lämpötilamittaukset sekä pumppujen tilat (Kuva 2).



Kuva 1. Kuva Multi-24 -säätimestä kalustettuna Geniusiksi ja siihen on liitetty GSM-modeemi



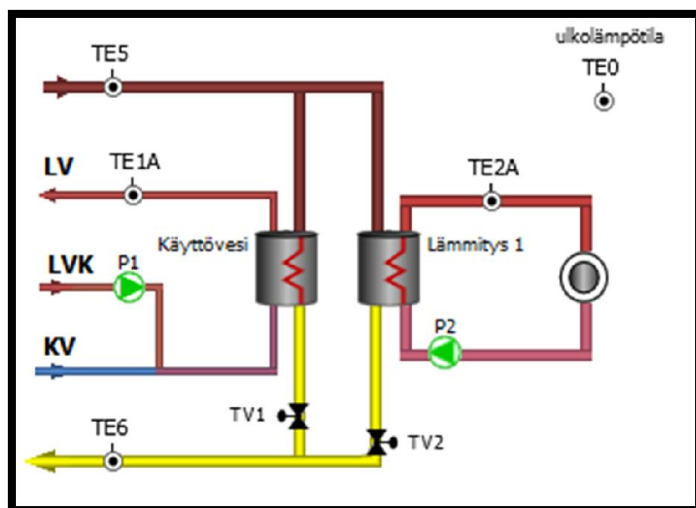
Kuva 2. Kuva valmiista tuotteesta, jota käytetään lämmönjakokeskuksissa.

## 4.2 Käyttökohteita

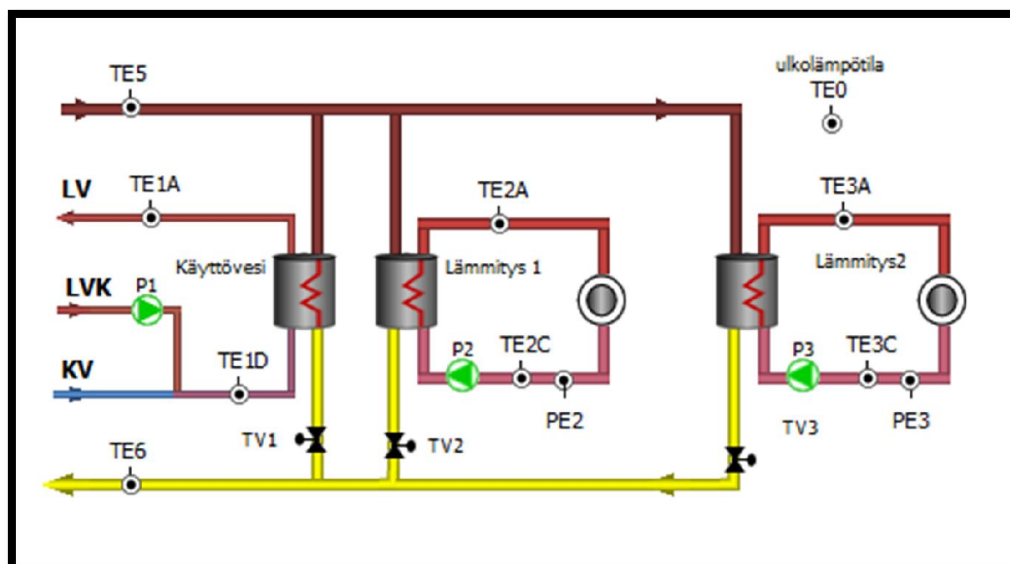
Genius-säädintä käytetään tällä hetkellä ainoastaan lämmönjaon säätämiseksi ja tulevaisuudessa. Geniusta voidaan käyttää ohjaamaan lämmönjakoverkostoa, joihin kuuluu kaksi tai kolme lämmönjakopiiriä. Lämmönjakopiirit jaotellaan yleensä käyttöveteen, lämmitysverkosto 1:een ja lämmitysverkosto 2:een, joista lämmitysverkostot voivat olla esimerkiksi ilmanvaihdon lämmityspiiri tai lattialämmityspiiri.

Geniuksen myytävät versiot on jaoteltu myös isoihin ja pieniin talokeskuksiin, joilla tarkoitetaan lämmityspiirien laajuutta. Toisin sanoen geniuksesta on tällä hetkellä neljä myytävää versiota, 2-piirinen pientalokeskus, 2-piirinen isotalokeskus, 3-piirinen pientalokeskus ja 3-piirinen isotalokeskus. Pientalo- ja isotalokeskukset ovat tarkoitettu niemiensä mukaisesti erikokoisille kiinteistöille, pientalokeskus on tarkoitettu yleisesti omakotitaloille tai pienille rivitaloille ja isotalokeskus on tarkoitettu pienille kerrostaloille, joihin

ei ole kannattavaa rakentaa monipuolista automaatiojärjestelmää. Pientalo- ja isotalokeskukset eroavat toisistaan mittaustietojen perusteella. Pientalokeskuksessa ei ole ol- lenkaan painemittauksia eikä pumpuille indikoiteja. Kuva 3 ja Kuva 4 on esitetty 2- ja 3- piiristen pientalokeskusten piirikaaviot tehtynä Fidelixin HTML-editorilla (ks. Kappale 7.2).



Kuva 3. Kaksi-piirisen pientalokeskuksen piirikaavio



Kuva 4. Kolme-piirisen pientalokeskuksen piirikaavio

## 5 Genius-säätimen ominaisuuksia

Genius-säätimen toiminta perustuu sen fyysiseen kokoonpanoon sekä siihen ennalta ohjelmoituihin toimintoihin. Tässä kappaleessa kuvataan sitä, miten Genius-säädin toimii ja miten sitä voidaan muokata toimimaan erilaisissa tilanteissa.

### 5.1 Mittaukset

Geniuksessa on mittauspisteitä, joita käsitellään pakollisina, jotta säädin toimii sekä valinnaisia mittauksia, joista saadaan tietoa tai lisää ominaisuuksia esimerkiksi energian säästöön. Pakollisia mittauksia ovat piirien menolämpötilat sekä paikallinen ulkolämpötila. Valinnaisia mittauksia ovat kaukolämmön meno- ja paluulämpötilat, kaikkien piirien paluulämpötilat sekä lämmitysverkostojen paineet. Mittauksia simuloidaan testerissä AO-pisteiden avulla.

### 5.2 Säätoventtiilit

Lämmönjaon säädössä olennainen osa on säätoventtiilit ja niiden ohjaukset. Venttiilit ovat lämmönsiirtimen ensiöpuolella (ks. kappale 2.1). Säädin vertaa piirin asetusarvoa, eli haluttua menoveden lämpötilaa todelliseen menoveden mittaukseen ja muuttaa venttiilin asentoa.

### 5.3 Pumppujen ohjaus

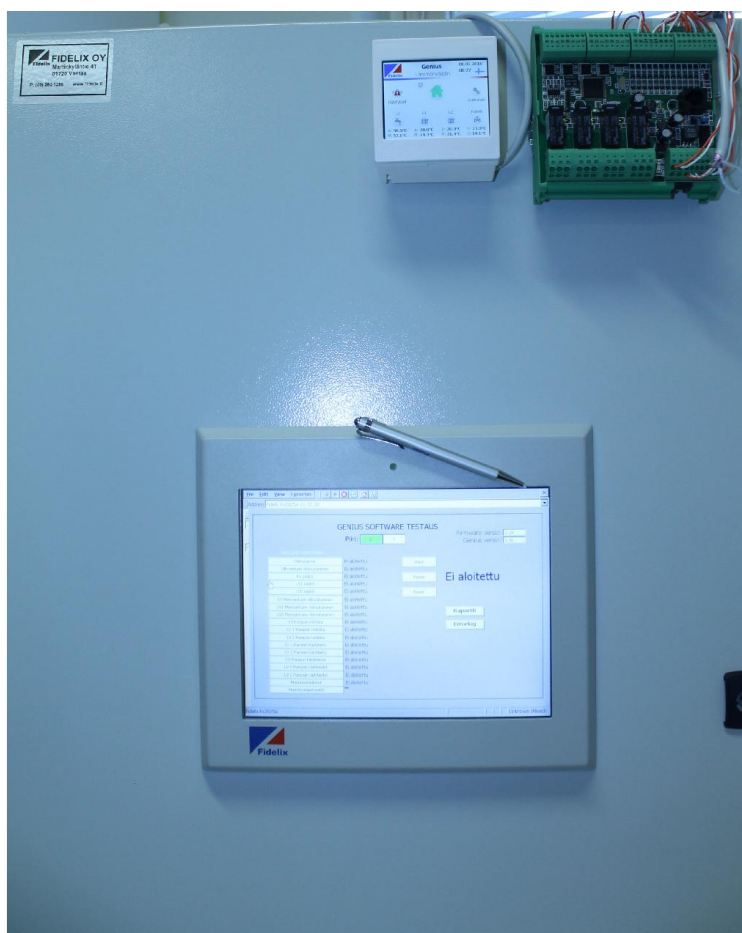
Säädin ohjaa pumppuja päälle tai pois, mutta ei niiden pyörimisnopeutta. Pumppujen toimintatila varmistetaan myös pumpulta tulevalla kärkitiedolla, jotta tiedetään pumpun oikeasti olevan toiminnassa tai päällä. Kärkitiedon saa myös pois käytöstä säätimeltä tai sitä voidaan käsitellä hälytystietona tarvittaessa.

## 5.4 Hälytykset

Kaikista virhetilanteista, jotka voivat vaikuttaa säätimen toimintaan, saadaan hälytys. Jos esimerkiksi verkostojen lämpötilat ovat yli tai alle tietyn kriittisen rajan, tulee hälytys joka näkyy näytöllä. Toinen vaihtoehto on liittää Geniukseen GSM-modeemi, joka lähettää tekstiviestinä aina kun hälytys aktivoituu tai poistuu.

## 6 Testauslaitteisto ja kytkennät

Testauslaitteisto koostuu täysin Fidelix Oy:n omista tuotteista. Testauslogiikkana toimii Fx-2025A-keskusyksikkö. Mittauksia sekä tilatietoja simuloidaan fyysisesti käyttämällä idelixin IO-moduuleita, eli yhtä COMBI-36 -moduulia ja kahta AO-8 -moduulia (Kuva 5).



Kuva 5. Testauslaitteiston kaapin ovi, oikealla ylhäällä on Genius –säädin ja sen näyttö ja keskellä on logiikkaohjain Fx-2025A. Säädin on laitettu kaapin etuosaan, jotta sen voi vaihtaa ja näytöltä voi katsoa, mitä testeri oikeasti tekee säätimelle.

Kaikki moduulit sekä Genius-säädin kommunikoivat logiikan kanssa Modbus-väylän kautta, eli niitä voidaan ohjata ja niiltä voidaan lukea tietoja RS485-standardin mukaisesti. Testauslaitteistossa käytetään kahta AO-8-moduulia, koska testauslaitteiston pitää olla kykenevä testaamaan 2- ja 3-piiriset säätimet ilman kytkentämuutoksia..

## 6.1 Logiikkaohjain Fx-2025A

Fidelixin Fx-2025A on Fidelixin oma PLC eli logiikkaohjain. Fx2025A on Windows CE:hen rakennettu kosketusnäytöllä ja usealla sarjaportilla varustettu logiikkaohjain. Laitetta voidaan ohjelmoida käyttämällä IEC-61131-3-ohjelmointistandardin mukaisia ohjelmointikieliä. Testeri on kokonaan ohjelmoitu käyttämällä tätä IEC-ohjelmointistandardia.

## 6.2 IO-modulit ja kytkentäkaaviot

### 6.2.1 COMBI-36 ja AO-8

COMBI-36-moduuli on yhdistetty IO-moduuli, jossa on 36 eri IO-pistettä. COMBI-36 siis koostuu

- 12:sta DI-pisteestä
- 8:sta DO-pisteestä
- 8:sta AI-pisteestä
- 8:sta AO-pisteestä.

AO-8-moduuli on moduuli, jossa on kahdeksan eri analogista ulostulopistettä. AO-8 on siis kohdistettu vain yhteen ulostulotyyppiin.

### 6.2.2 Multi-24-moduuli ja Genius-säädin

Genius-säätimen kokoonpano perustuu Fidelix Oy:n Multi-24-moduliin, johon on tehty ohjelma toimimaan kaukolämmönsäätimenä ja lisätty paikallinen näyttö. Multi-24-moduli voi toimia omana PLC-logiikkana, mutta muut aiemmin mainitut modulit eivät. Multi-24:ssä IO koostuu

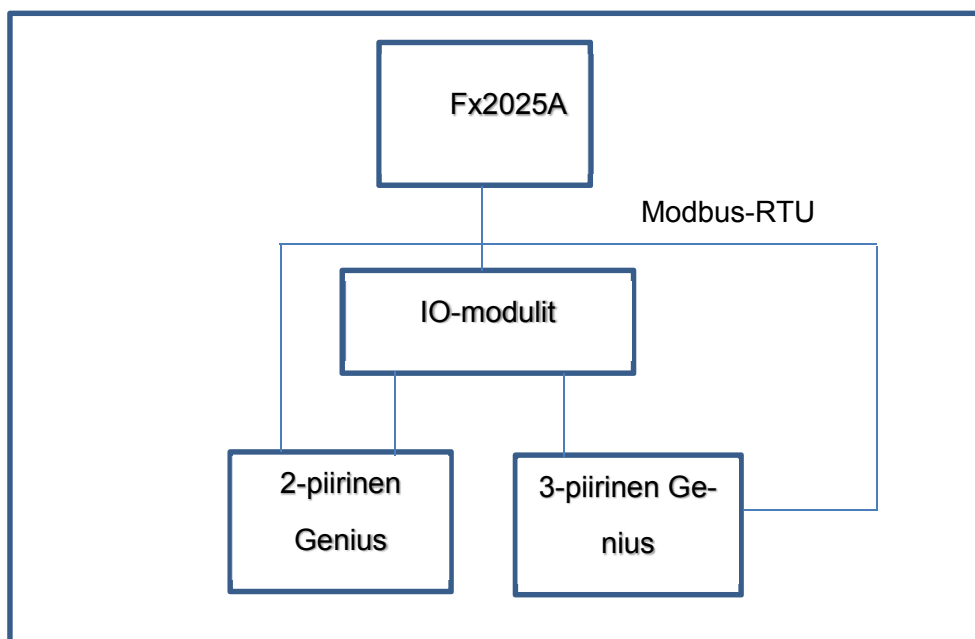
- 12:sta ohjelmallisesti valittavasta AI tai DI-pisteestä
- 4:stä DO-pisteestä
- 8:sta AO-pisteestä.

Genius-säätimeen on näistä vaihtoehdoista ennalta määritetty mihin kytketään mikäkin sisääntulo ja ulostulo.

### 6.3 Kytkenät testerissä

Genius on kytketty IO-moduuleihin, jotta voidaan simuloida tarvittuja mittauksia. Yksi COMBI-36 ei riitä kaikkien mittausten simulointiin, joten kokoonpanoon on lisätty kaksi AO-8:a. Tähän kokoonpanoon on varattu tila kaksi- ja kolmepiirisen geniuksen kytkentöihin. Kaikkia moduuleita sekä joitain Geniuksen asetusarvoja ohjataan Modbus-väylän kautta. Kuvassa 6.2 on esitetty kytkentäkaavio testausjärjestelmälle.



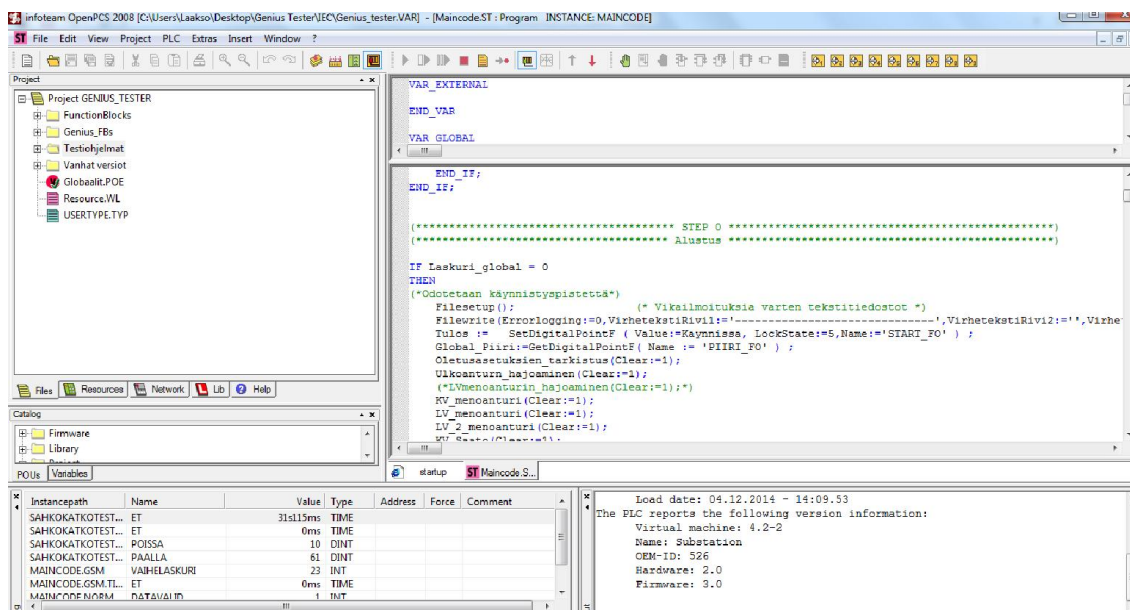


Kuva 6. Testausjärjestelmän kytkentäkaavio

## 7 Käytetyt ohjelmatyökalut

### 7.1 Infoteam OpenPCS

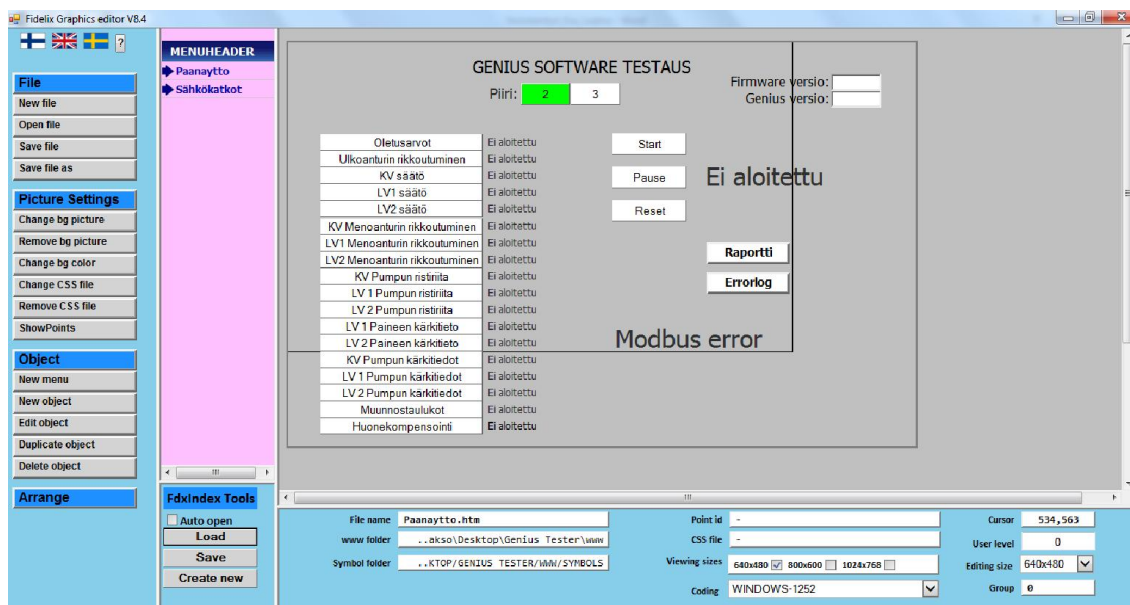
Infoteam OpenPCS-ohjelmointityökalua käytettiin tuottamaan testaukseen logiikkaohjelma IEC-61131-3 standardin mukaisesti (Kappale 8). Tällä työkalulla on myös tehty ohjelma Genius-säätimeen. Infoteam OpenPCS:llä pystytään tuottamaan ohjelma sekä syöttämään se Fx-2025A logiikkaohjaimen lähiverkossa. Tämän lisäksi työkalulla voidaan seurata ohjelman muuttujien tiloja, kun ohjelma on käynnissä logiikkaohjaimessa. Kuva 7 on esitetty OpenPCS.



Kuva 7. Infoteam OpenPCS.

## 7.2 Fidelix HTMLeditor

Fidelix HTML-editor on nimensä mukaisesti HTML –editointityökalu, jota käytetään tuottamaan käyttöliittymä Fidelixin logiikkaohjaimiin. HTML-editor tuottaa HTML-muotoisen verkkosivutiedoston, jota logiikkaohjaimen web-server käyttää arvojen ja automaatiojärjestelmän esittämiseen. Kuva 8 on viimeisimmästä Genius-testerin käyttöliittymän editointia.



Kuva 8. Genius-testerin käyttöliittymän viimeisin versio avattuna.

## 8 IEC 61311-3 –ohjelmointistandardi

IEC 61311-3-ohjelmointistandardi on PLC-logiikoissa käytettävä ohjelmointityyli. Sen on luotu nopeuttamaan ohjelmoinnin toteutusta ja parantamaan ohjelman yleistä laatua [3, xi]. Verrattuna muihin ohjelmointikieliin, IEC 61311-3:n mukaiset kielet eivät kehity yhtä nopeasti, vaan sen käytön on tarkoituksena olla virheetöntä [3, xi]. IEC 61311-3:n kuuluu 5 eri ohjelmointikieltä, joita käytetään eri PLC-valmistajien määritysten mukaisesti. Nämä kielet ovat Structured Text, Function Block Diagram, Ladder Diagram, Instruction List ja Sequential Function Chart [3, s. 117]. Fidelixin PLC:tä, eli fx2025A:ta ohjelmoidaan Structured Text-kielillä.

Jokainen IEC 61131-3:lla tuotettu ohjelma voidaan jakaa erillisiin suoritettaviin ohjelmayksiköihin, (englanniksi Program Organisation Unit, POU). Näitä ohjelmia suoritetaan PLC-laitevalmistajan määrittelyn mukaisesti joko jatkuvana toistona tai tietynä ajankohdana, esimerkiksi kun PLC käynnistetään.

IEC 61311:n yksi ominaisuus on se, että kun valmista ohjelmaa suoritetaan, se ei suorita ohjelmariviä kerrallaan, eikä se muokkaa muuttujien arvoa ns. reaaliaikaisesti, vaan koko koodi käydään läpi ja lopputuloksena on viimeisimmät toteutuneet määrittelyt. Tällä oli

suuri vaikutus testausohjelmaa toteutettaessa, koska ohjelmaa ei voinut kirjoittaa suoraan reaaliaikaiseksi, vaan piti suorittaa ohjelmaa uudestaan kun halusi tehdä ajasta riippuvaista ohjelmaa.

### 8.1 Structured Text (ST) –ohjelmointikieli

Structured Text-ohjelmointikieli on tekstimuotoista ohjelmointia. Sen tarkoituksena on yksinkertainen arvojen käsittely sekä monipuoliset aritmeettiset laskutoimitukset [3, s. 118]. Tässä ohjelmointikielessä jokainen muuttujien hallinta hoidetaan tekemällä muuttujan määrittelmä POU:n alkuun ja sen toiminta POU:n loppuun [3, s. 69]. Kuva 9 Kuva 8 on esimerkki, jossa luodaan kokonaislukumuuttujat x1 ja x2. Muuttujaan x1 laitetaan määrittelyn yhteydessä arvo 7 ja x2:n lasketaan toteutuksessa x1:n neliö.

```
VAR  
x1 : int := 7;  
x2 : int;  
END VAR
```

```
x2:=x1*x1;
```

Kuva 9. Yksinkertainen esimerkki ST-ohjelmointikielen käytöstä.

### 8.2 Functionblock

Functionblockit ovat tärkeä osa IEC 61131-3:a, sillä niiden avulla tietyt toistuvat ohjelmarakenteet saadaan yksinkertaistettua ja hoidettua yhdellä ohjelmapätkällä [3, s. 223]. Functionblockeja kutsutaan ohjelmoijan määrittämässä kohdassa ja se suorittaa halutun loogisen operaation. Functionblockille voidaan määrittää sisääntulo- ja ulostuloparametrejä, joita käytetään Functionblockin toteutuksessa [3, s. 225]. Niissä voidaan käyttää kaikkia IEC 61131-3:n määrittämiä ohjelmointikieliä.

## 9 Testivaiheet ja niiden ohjelmoinnin toteuttaminen

Geniuksen testausohjelma tuotettiin testivaihe kerrallaan. Alussa piti ensin kokeilla Geniuksen ja Fidelixin PLC:n rajapintaa Modbus-väylän kautta, jotta tiedetään, miten simuloinnin täytyy tapahtua. Ratkaisuksi todettiin vaiheittainen ohjelmointi, joka tuotettiin IEC-61131-3-ohjelmointistandardin mukaisella ST-koodikielellä. Testauksen suorittamiseen todettiin hyväksi vaiheittainen testaustyö, eli odotetaan edellisen testausvaiheen päättymistä ennen kuin aloitetaan uuden suorittaminen. Tämä todettiin sopivaksi, kun kaikkia asioita ei voi, eikä välttämättä haluta testata samanaikaisesti. Tässä kappaleessa testausvaiheet on selitetty niiden toteutusjärjestyksessä testerissä, ei ohjelmointijärjestyksessä.

Jokainen testausvaihe on oma Functionblock IEC-koodissa, jotta koodin tulkitseminen on jälkeempään yksinkertaisempaa. Lisäksi koodiin tehtiin pohjalle neljä functionblockia, joita käytetään toistuvasti testausten aikana ja joihin testerin koodi sekä toiminta perustuvat. Nämä functionblockit ovat säätimeltä luku, sinne kirjoitus, raportin alustus sekä raportin kirjoitus. Liitteessä 1 on esitetty tyyli, jolla testauslaitteisto on ohjelmoitu.

Jokaisessa testauksen vaiheessa kirjoitetaan kahteen PLC:ssä olevaan tekstitiedostoon raportti testaustuloksesta. Tämä helpottaa paikantamaan mahdollisen puutteen säätimen ohjelmassa. Toinen raportti on vain testituloksen ilmoittaminen, eli esimerkiksi ”Testivaihe on OK” tai ”Testivaihe on hylätty”. Toinen raportti on virheloki, jossa kerrotaan tarkemmin, mikä vaihe meni pieleen testatessa, esimerkiksi ”Säätötestin ensimmäisen vaiheen säätötulos ei muuttunut”.

Säätimen mittauksia simuloidaan syöttämällä IO-moduulien AO-pisteillä jännitettä säätimen AI-mittauspisteisiin. Lämpötilamittauksissa 1,0V-3,3V vastaa jotakin vastusarvoa, jota säätimeen kytkettävät lämpötilavastukset olisivat. Indikoiteja eli tilatietoja simuloidaan käyttämällä DO-pisteitä IO-moduuleissa ja kytkemällä ne säätimen DI-pisteisiin. IO-moduuleista on selitetty enemmän kappaleessa 6.

## 9.1 Normaalisimulaatio

Normaalisimulaation pitää nimensä mukaisesti simuloida normaalitilannetta säätimelle. Tämä ei ole testausvaihe, vaan säätöjen testausta helpottava toiminto. Sen tarkoituksen on muuttaa kaikkien verkostojen menovesien lämpötilaa asetusarvojen molemmin puolin. Tämän seurauksena säädin ohjaa venttiileitä auki tai kiinni, pitäen ne tietyllä alueella ja valmiina säätöjen testausta varten. Normaalisimulaatiota käytetään useaan kertaan testausten aikana, ennen kuin testaus alkaa ja ennen jokaisen verkoston säätöjen testausta.

Normaalisimulaatioon on tehty toiminto jotta säätöjen testauksen aikana verkostojen venttiilit eivät pääse 100 % tai 0 %, jolloin säätöjä ei voida testata ilman pysäytyksiä. Tämä toteutetaan simuloimalla asetusarvosta suuresti eroavaa lämpötilaa, noin 20 °C:n erotusta asetusarvon molemmin puolin. Tämän seurauksena säädin yrittää ohjata nopeammin auki tai kiinni PI-säätimen ominaisuuksien mukaisesti. Tällä pidetään venttiilit 30 % - 70 % auki, joka on todettu riittäväksi säätöjen testausta varten.

## 9.2 Oletusasetuksien tarkistus

Oletusasetusten tarkistus testaa, että geniukseen asetetut vakioparametrit ovat ne, mitkä niiden pitääkin olla. Näihin parametreihin kuuluvat kaikki geniuksen toimintaan vaaditut sekä erityistoimintojen asetukset. Koodi on yksinkertaisesti pelkkiä ehtolauseita jokaiselle parametrille. Jokaisesta virheestä kirjoitetaan rivi raporttiin ja kerrotaan modbus -rekisterin tunnus, jotta voidaan nähdä, mikä asetusparametri ei ole niin kuin pitäisi.

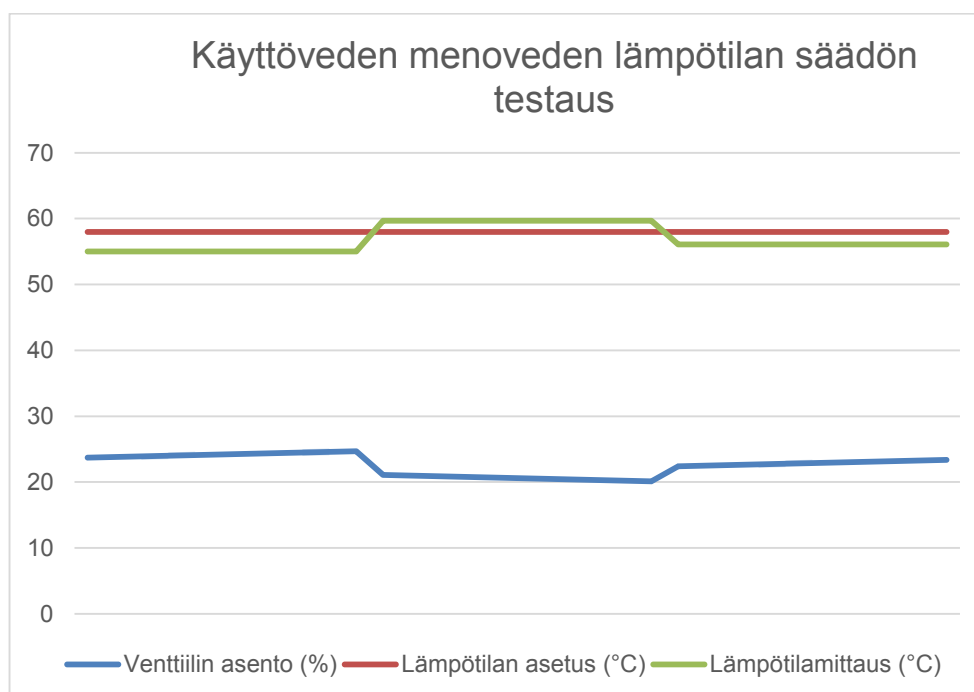
## 9.3 Ulkoanturin hajoaminen

Jos käy niin, että Geniukseen liitetty ulkoanturi hajoaa, eli ei anna mitään lämpötila-arvoa tai on yli järkevästä asteikosta, geniuksen täytyy muuttaa lämmitysverkostojen asetusarvo ja ohjata vesi säätökäyrän keskimmäiseen pisteeseen. Testauksessa siis simuloidaan ääretöntä vastusarvoa, jolloin säädin tulkitsee anturin olevan rikki. Tämän jälkeen tarkistetaan, onko asetusarvo yhtä kuin keskimäinen säätöpiste.

## 9.4 Verkostojen lämpötilojen säätö

Säätöjen testaus on testerin tärkein osa. Tällä taataan, että säädin suorittaa juuri oikein sen päätarkoituksen, eli lämmitys tai käyttövesiverkostojen lämmönsäädön. Testauksessa simuloidaan aina kyseisen piirin menoveden anturin lämpötilaa asetusravon molemmin puolin. Jokaisen verkoston säätöjen toiminta voitaisiin tarkastaa samanaikaisesti, mutta testeriä tuottaessa todettiin, että on kuitenkin varmempaa, että jokainen verkosto testataan erikseen. Kaikkien verkostojen säätöjen testaus toimii samalla tavalla.

Ennen testausta suoritetaan kappaleessa 6.1 mainittu normaalisimulaatio, jottei verkoston venttiili ole eikä mene täysin auki tai kiinni kesken testin. Kun venttiili on sopivassa kohdassa, alkaa testin ensimmäinen vaihe. Ensin muutetaan testattavan verkoston menolämpötilaa joitakin asteita asetusravon alapuolelle. Tämän seurauksena, säädin pyrkii avaamaan toimilaitetta ja lämmittämään verkostoa. Testeri lukee venttiilin asennon kolmena eri ajankohtana ja tarkistaa, että se on liikuttanut toimilaitetta oikeaan suuntaan. Tämän jälkeen muutetaan lämpötilaa asetusravon yläpuolelle ja tehdään samat asiat toisinpäin.



## 9.5 Verkostojen menoanturin hajoaminen

Geniukseen on tehty varotoiminta sille varalle, että menoanturi on hajonnut. Säädin pyrkii tässä tilanteessa säätämään verkoston lämpötilaa paluuvesianturin lämpötilan mukaisesti. Käyttövesiverkoston kiertovesi pyritään pitämään 45 °C:ssa ja lämmitysverkostojen paluuvesi 25 °C:ssa. Testerillä siis tarkistetaan, että säädin tekee samat asiat kuin normaalisäädössä (kappale 6.4), mutta menolämpötilan sijaan, muutetaan kyseisen verkoston paluulämpötilaa tiettyyn suuntaan.

## 9.6 Verkostojen pumpun ristiriita

Jos jonkin verkoston pumpun ohjaus tai indikointi (tilatieto) poikkeavat toisistaan, Geniuksesta tulee hälytys. Testeritarkistaa, että näin oikeasti tapahtuu. Jokaisen verkoston pumppu toimii samalla tavalla ja tämän testin jokainen vaihe on samanlainen jokaiselle verkostolle.

Ensimmäisessä vaiheessa varmistetaan, että pumppujen ohjaus ja indikointi ovat kummatkin päällä. Seuraavaksi otetaan testattavalta pumpulta indikointi pois päältä ja odotetaan noin 12 sekuntia, että pumpun ristiriitahälytys käynnistyy. Tämän jälkeen tarkistetaan hälytyksen aktiivisuus, ohjataan pumpun indikointi takaisin päälle ja kuitataan hälytykset ja tarkistetaan, että kaikki on normaalitilassa.

Testerin toisessa vaiheessa tehdään hälytyksen kuittauksen hälytyksen poistumisen tarkistus toisinpäin. Eli siis aktivoidaan ristiriitahälytys, tarkistetaan sen aktiivisuus ja toisin kuin aiemmassa vaiheessa, nyt kuitataan hälytys ensin ja vasta sitten aktivoidaan indikointi. Tekemällä testaus kahdella eri tavalla nähdään, että hälytys on juuri siinä tilassa kuin pitääkin.



## 9.7 Lämmitysverkostojen paineiden kärkitieto

Tietyissä käyttökohteissa ei käytetä painelähetintä ollenkaan, vaan pelkästään painekyt-kintä, jossa sulkeutuva tai avautuva kosketin vastaa paineen ylä- tai alarajahälytystä. Testausvaiheen teki osittain haastavaksi se, että painelähettimeen ei kytketty DO-pis-tettä COMBI-kortilta, vaan AO-pistettä, josta ei saada suoraan varsinaista kärkitietoa, vaan kärkitieto pitää simuloida jännitteellä. Syöttämällä AO-pisteellä 3,3V:n jännite saa-daan aikaiseksi avautunut kosketin ja laskemalla jännite 0V:n saadaan sulkeutunut kos-ketin. Käyttövesiverkostossa ei ole mahdollisuutta käyttää painelähetintä ollenkaan. Kummankin lämmitysverkoston paineiden kärkitiedot toimivat samalla tavalla, joten testi on samanlainen molemmille lämmitysverkostoille.

Testauksen ensimmäisessä vaiheessa muutetaan painelähetin NO-kosketintiedoksi ja tarkistetaan, että säädin tulkitsee painemittauksen kärkitietona eikä mittauksena. Kun tila on muuttunut, simuloidaan painemittaukselle sulkeutuvan koskettimen tila. Tämän jäl-keen hälytyksen pitäisi aktivoitua. Seuraavaksi tarkistetaan, että hälytys on aktiivinen ja muutetaan koskettimen simuloitu tila takaisin avautuneeksi ja tarkistetaan, että hälytys ei ole enää aktiivinen ja kuitataan se pois.

Testauksen toisessa vaiheessa suoritetaan sama kuin edellisessä kappaleessa, mutta NC-kosketintietona. Ensin muutetaan siis painelähetin NC-kosketin -tiedoksi, varmiste-taan sen aktiivisuus, ohjataan mittauspisteeseen avautuvan koskettimen tila (3,3V), tar-kistetaan hälytyksen aktivoituminen, muutetaan koskettimen tila takaisin sulkeutuneeksi ja kuitataan hälytys pois.

## 9.8 Verkoston pumpun kärkitieto

Joissakin käyttökohteissa pumpuilta ei saada indikointia (tilatietoa), vaan pumpuilta tulee kärkitietona ainoastaan hälytys, joka kertoo, jos kysesisessä pumpussa on jokin vika. Kärkitiedot testataan samalla tavalla kuin paineiden kärkitiedot, paitsi että ne ovat jo kyt-ketty COMBI-kortin DO-pisteisiin, joten ei tarvitse simuloida kärkitietoa jännitteellä, vaan voidaan käyttää oikeaa kärkitietoa. Jokaisella verkostolla on oma pumppunsa, joten testi suoritetaan samalla tavalla jokaiselle verkostolle.

Testauksen ensimmäisessä vaiheessa muutetaan pumpun kärkitieto NO-kosketintiedoksi ja tarkistetaan, että säädin tulkitsee kärkitiedon hälytyksenä eikä indikointina. Kun tila on muuttunut, ohjataan kärkitieto kiinni. Tämän jälkeen hälytyksen pitäisi aktivoitua. Seuraavaksi tarkistetaan, että hälytys on aktiivinen ja muutetaan koskettimen tila takaisin avautuneeksi ja tarkistetaan, että hälytys ei ole enää aktiivinen ja kuitataan se pois.

Testauksen toisessa vaiheessa suoritetaan sama kuin edellisessä kappaleessa, mutta NC-kosketintietona. Ensin muutetaan siis painelähetin NC-kosketintiedoksi, varmistetaan sen aktiivisuus, ohjataan kärkitieto avautuvan koskettimen tilaan, tarkistetaan hälytyksen aktivoituminen, muutetaan koskettimen tila takaisin sulkeutuneeksi ja kuitataan hälytys pois.

### 9.9 Antureiden muunnostaulukot

Jokaisella mittauksella on oma anturityyppinsä ja muunnostaulukolla valitaan, millä laskukaavalla vastusarvo muutetaan lämpötilaksi. Testeri syöttää COMBI-kortin AO-pisteellä tietyn jännitearvon jokaiselle mittauspisteelle ja tarkistaa, että mittaustulos on tietyn alueen sisällä. Geniuksen tukemat anturielementit ja niiden testausjärjestys on NTC20, NTC10, Pt1000, Ni1000Lg. Mitään anturityyppiä ei testata samanaikaisesti kahden tai useamman mittauspisteen kesken, vaan ne testataan portaittain vaihtamalla anturityyppiä kun edellinen kierros on tarkistettu. Kuva 10 on hahmotelma testausmetodista käyttövesi- ja lämmitysverkoston mittauspisteistä.

	Ensimmäinen vaihe				Toinen vaihe				Kolmas vaihe			
	NTC10	NTC20	Pt1000	Ni1000lg	NTC10	NTC20	Pt1000	Ni1000lg	NTC10	NTC20	Pt1000	Ni1000lg
KV meno	X					X					X	
KV kierto		X					X					X
LV meno			X					X	X			
LV paluu				X	X					X		

Kuva 10. Kuvaus muunnostaulukojen testausmetodista

## 9.10 Huonekompensointi

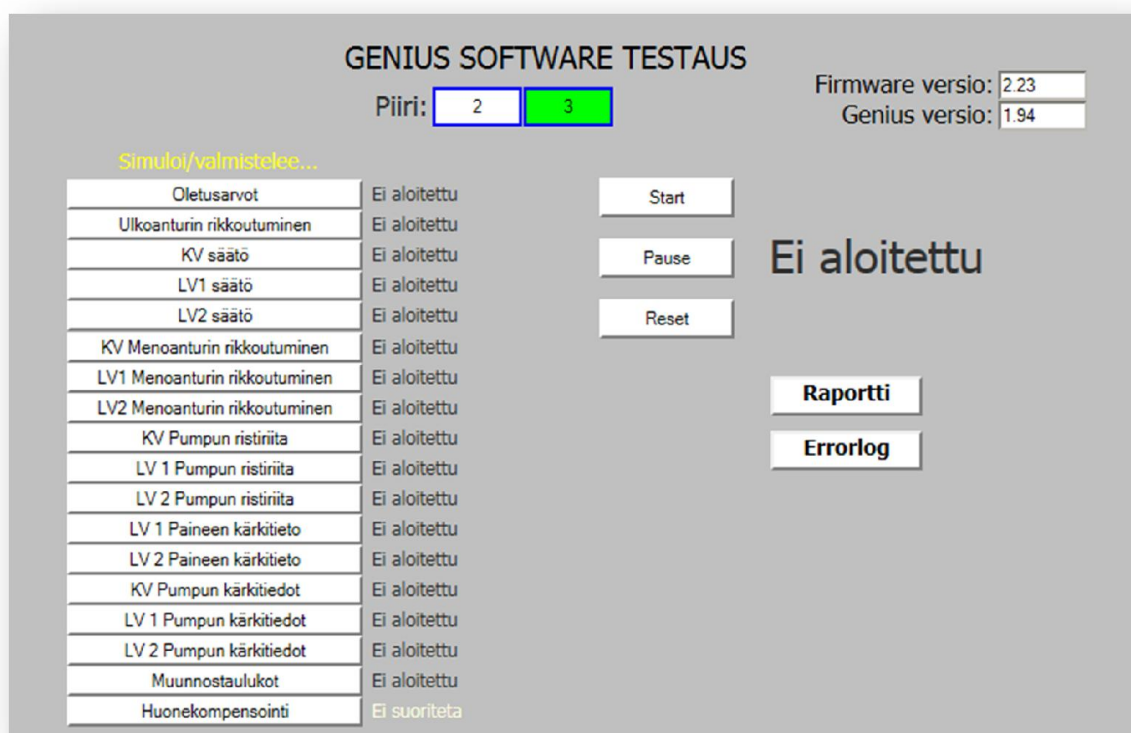
Geniuksessa on mahdollista asettaa ensimmäiselle lämmitysverkostolle huonelämpötilaan suhteutettu kompensointi. Tämän avulla saadaan parempi takaisinkytkentä lämmityksen vaikutuksesta ja saadaan paremmin tarvetta vastaava säätö sekä säästetään mahdollisesti energiaa.

Ensimmäisessä vaiheessa otetaan huonekompensointi käyttöön ja tarkistetaan, että se on päällä. Toisessa vaiheessa asetetaan huonelämpö huonekompensoinnin asetusarvon yläpuolelle. Tämän seurauksena säädin laskee lämmitysverkoston menoveden asetusarvoa, jotta huoneen lämpötila ei nouse. Testeri tarkistaa, että menoveden asetusarvo on nyt tietyllä alueella. Testauksen kolmannessa vaiheessa huonelämpötilaa lasketaan ja sen vaikutus tarkistetaan.

## 10 Käyttöliittymä

### 10.1 Käyttöliittymä yleisesti

Testausjärjestelmän käyttöliittymä on Fidelixin järjestelmään tehty HTML-pohjainen yksinkertainen kokonaisuus. Käyttöliittymän graafinen osuus on toteutettu Fidelix HTML-editor-työkalulla. Jokaiselle muuttuvalle grafiikalle, kuten tilateksteille ja napeille on määritetty pistetunnus, joka taas on linkitetty tietyn IO-modulin tiettyyn pisteeseen. Fidelixin automaatiojärjestelmiin verrattuna testerissä on hyvin pieni ja yksinkertainen käyttöliittymä. Tavallisessa kiinteistöautomaatiokohteissa voi olla useita sivuja käyttöliittymässä, mutta testerissä ei tarvitse tällä hetkellä kuin yhden sivun. Kuva 10 on esitettyä viimeisen version käyttöliittymä.



Kuva 11. Viimeisimmän version käyttöliittymä

## 10.2 Painikkeet

Käyttöliittymässä on pääasiassa kolme eri käyttötarkoitusta vastaavia painikkeita. Testauksen kannalta tärkeimmät painikkeet ovat ”Start”, ”Stop” ja ”Pause” ja lisäksi käyttöliittymässä valitaan, kuinka monipiirinen genius on testauksessa. Jokaiselle testausvaiheelle on lisäksi oma painike, joilla voidaan suorittaa yksittäinen vaihe.

”Start”-painike käynnistää koko testauksen, tai jatkaa sitä siitä kohdasta, johon se on pysäytetty ”pause”-painikkeella. ”Start”-painike ei tee mitään, jos jokin yksittäisen testausvaiheen painike on painettuna. ”Pause”-painike pysäyttää testerin siihen vaiheeseen mihin se jäänyt. ”Reset”-painike asettaa testerin nollatilaan, jolloin testaus voidaan aloittaa alusta.



Kuva 12. Testauksen kokonaisuuden hallinnan painikkeet

### 10.3 Tilatekstit

Käyttöliittymään on tehty tilatekstejä, jotta käyttäjä tietää mitä testeri tekee. Käyttöliittymään on tehty koko testille sekä jokaiselle yksittäiselle testivaiheelle teksti, jossa lukee, missä tilassa mikäkin vaihe on. Tilatekstissä lukee joko "Ei aloitettu", "Suorittaa...", "Hyväksytty", "Hylätty" tai "Pause", mikä kuvastaa tilatietoa kyseisestä testivaiheesta tai koko testistä. Kuva 13 on esitetty tilatekstien ulkonäöt.



Kuva 13. Kuva testerin käyttämistä tilateksteistä.

### 10.4 Muut asiat käyttöliittymässä,

Päätoimintojen lisäksi testerissä on muutama muu asia helpottamaan käyttäjää. Jos testeri suorittaa simulaatiota valmistellakseen tiettyä testivaihetta, testivaiheiden yläpuolella

lukee ”Simuloi/Valmistele...”. Lisäksi grafiikan oikeassa yläkulmassa lukee säätimen versiotiedot, jotta käyttäjä näkee testaavansa mitä pitääkin. Käyttöliittymässä on myös linkit raportteihin, jotka on mainittu kappaleessa 8.

## 11 Testilaitteiston testaus

Testilaitteisto piti käydä useaan kertaan läpi ja varmistaa sen toiminta. Jokaisen testausvaiheen jälkeen kyseinen vaihe testattiin heti läpi ja katsottiin sen toiminta. Infoteamin ohjelmiston sekä testattavan laitteiston läsnäolon ansiosta jokaisen vaiheen testauksen pystyi toteamaan helposti paikallisesti.

Ohjelman vikojen takia saatiin joskus eri tuloksia suorittamalla yksi testivaihe tai koko testaus. Esimerkiksi pumppujen ja paineiden kärkitietojen testaukset hyväksyttiin joka kerta, kun ne suoritettiin yksittäin, mutta jos jotain suoritettiin niitä ennen, testeri saattoi antaa hylätyn. Tämä johtui ohjelmointivirheestä, jossa kärkitietojen testauksen alusta puuttui viive, joka odotti säätimen muuttavan asetukset oikeaksi. Tässä kappaleessa ei mainita jokaista yksittäistä virhettä tai ongelmia jokaisessa testivaiheessa, vaan suurimmat ongelmat, jotka tulivat ilmi testilaitteistoa testattaessa ja arvioidessa.

### 11.1 Testivaiheiden suoritus normaalisti

Jokaista testivaihetta suoritettiin ja testattiin useaan kertaan ja niissä oli ja on edelleen jotain parannettavaa. Tässä kappaleessa on yhteenveto testerin testauksesta testerin viimeiselle versiolle.

#### 11.1.1 Oletusasetuksien tarkistuksen testaus

Oletusasetuksien tarkistus toimi hyvin, se on vain vertailu kymmeniä eri asetusarvoja, joten ainoa asia mikä tässä voisi yleisesti epäonnistua, olisi se, että testattavaa säädintä ei ole palautettu tehdasasetuksille.

### 11.1.2 Verkostojen säätöjen testaus

Verkostojen säädössä on ollut paljon pieniä ongelmia. Käyttövesiverkosto ja lämmitysverkostot säätävät eri tavoin. Esimerkiksi käyttövesiverkoston tarvitaan herkemmin reagoivaa säätöä, joten suorittaessa testiä lämmitysverkostojen menovesien lämpötiloilla täytyy olla isompi erotus asetusarvoon nähden. Lisää ilmenneitä ongelmia on kerrottu kappaleessa 9.3. Kaikkien vikojen korjauksen jälkeen tämän testausvaiheen voitiin todeta toimivan hyvin.

### 11.1.3 Verkostojen menoantureiden rikkoutumisen testaus

Verkostojen menoantureiden rikkoutumisen testauksessa oli samanlaisia ongelmia kuin normaaleissa säädöissä mainittu kappaleessa 10.1.2. Tästä testausvaiheesta saatiin kuitenkin hyvin toimiva ilman virheitä.

### 11.1.4 Pumppujen ristiriidat

Ristiriitoja testatessa ei tullut mitään ongelmia enää testivaiheessa. Kaikki toimi niin kuin pitääkin.

### 11.1.5 Paineen kärkitiedot ja pumppujen kärkitiedot

Näissä testausvaiheissa oli yksi yhteinen ongelma, joka häiritsi pitkään. Ohjelmasta oli unohtunut viive, joka odottaa säätimen muuttavan asetusarvojen tyyppiä kärkitiedoksi. Tämän seurauksena säädin ei joko muuttanut asetusta kärkitiedoksi tai sitten sillä tuli niin monta virhettä siitä, että jokin muu virhe sai aikaan hylätyn. Kun tämä saatiin korjattua, säätö toimi hyvin.

### 11.1.6 Muunnostaulukkojen testaus

Muunnostaulukkojen testauksessa toimi hyvin, lukuun ottamatta tarkkuusvirhettä Pt1000- ja Ni1000lg-anturityyppien muunnostaulukoilla. Tämä ongelma huomattiin vahingossa, kun mittaustulosta simuloivan AO-pisteen liitin oli huonosti kiinni, vaikka jos vaihdettiin testattavaa säädintä. Tästä seurasi se, että näiden muunnostaulukkojen tes-

tausvaiheessa sallittu alue oli väärin. Tämä johtuu siitä, että muunnostaulukon mittaus-tulos on suuri pienellä jännitealueella. Jos esimerkiksi mittauspisteeseen syötetty jännite poikkesi 0.01 voltia, mittaus-tulos poikkesi 5 astetta. Huonon kontaktin takia, mittauspisteessä oli suurempi jännite ja se ei enää toiminut samalla tavalla. Lisäksi tähän vaiheeseen vaikuttava tekijä on se, että AO -pisteet ei ole suunniteltu niin tarkkoihin jännitealueisiin. Näiden testaukseen täytyi lisätä suurempi hyväksyttävä mittausalue, jotta käyttäjän ei tarvitse muuttaa aluetta riippuen testattavasta säätimestä. Paras korjaus tälle olisi tehdä suhteellinen laskuri käyttämällä toista muunnostaulukkoa.

#### 11.1.7 Huonekompensoinnin testaus

Huonekompensoinnin testauksessa ei tullut mitään ongelmia. Lämmitysverkosto 1:n asetusarvo muuttui aina oikeaan suuntaan ja oikeassa suhteessa, kun huonekompensointi otettiin käyttöön ja muutettiin huonelämpötilaa.

#### 11.2 Testauksen häirintä ja sen tulkinta

Kappaleessa 7 mainituille testivaiheille tehtiin kyseiseen testivaiheeseen vaikuttavia ongelmia, jotta nähdään, reagoiko kyseinen testivaihe oikein ja ettei tehty ongelma vaikuta mihinkään muuhun testivaiheeseen. Aina kun tehtiin häiriö, suoritettiin koko testi läpi, jotta nähdään sen vaikutus muihinkin testivaiheisiin, kuin mihin häiriö on tarkoitettu. Tämän avulla saatiin testattua odottamattomiakin vaikutuksia, joita ei ohjelman tuottajana välttämättä huomaa.

##### 11.2.1 Oletusarvojen testaus

Tätä testatessa muutettiin yhtä säätimen asetusta, jolloin tämä testivaihe hylätään. Tässä asiassa testeri toimi niin kuin pitääkin, eli testeri hylkäsi aina säätimen, koska jokin ennalta määritellyistä asetusarvoista poikkesi normaalista arvostaan.

##### 11.2.2 Ulkoanturin rikkoutuminen

Tälle testivaiheelle ei ollut hylkäävää tekijää, jonka olisi voinut tuottaa muuttamalla säätimen tai testerin ohjelmaa. Toteaminen tapahtui lukemalla asetusarvo suoraan näytöltä.



Tehdasasetuksilla asetusarvo olisi ollut 28 °C, joka on säätökäyrän keskimäinen arvo. Muuttamalla säätökäyrän keskimäistä arvoa nähtiin, että säädin aina otti sen arvon, eli tämä kohta hyväksytään vaikka kyseistä arvoa muutetaan. Tämä lopputulos oli juuri oikea testerin testauksen kannalta.

### 11.2.3 Verkostojen säätöjen testaus

Verkostojen toiminta voidaan lopettaa ottamalla pumppujen tilatiedot pois päältä tai muuttamalla verkoston menoveden asetusarvoa, jolloin verkosto ei toimi enää samalla tavalla ja testerit antaa hylätyn. Tässä vaiheessa tuli myös todettua testerin ongelma käyttää verkostojen menovesien eri asetusarvoa, kuin mikä on asetettu tehdasasetuksiin. Testerit muuttaa mittausarvoa ainoastaan tehdasasetuksissa määritetyn menoveden asetusarvon molemmin puolin, vaikka se voisi tarkistaa asetusarvon ja laskea uudet mittausarvot, joita se simuloi.

### 11.2.4 Verkostojen menoanturien rikkoutuminen

Irrottamalla paluuvesianturin kyseisestä verkostosta säädin yrittää yhä säätää paluuvettä, mutta sen lämpötila ei muutu aina tarpeen tullen. Tällöin säädin pitää yhden tietyn suunnan, johon se ohjaa venttiiliä. Testerit antaa hylätyn, kun venttiili ei muuta asentoa haluttuun suuntaan. Tässä vaiheessa tuli ilmi virhe koodissa, joka tarkisti venttiilin tuloksen väärin ja testerit antoi aina hyväksytyin riippumatta siitä, että oliko kaikki kunnossa, kunhan venttiili asento on vain muuttunut johonkin suuntaan. Testauksen ehto oli että kunhan venttiilin asento on eri kuin edellisellä tarkistuskierröksellä. Tämä virhe oli ollut myös säätöjen testauksessa, mutta siitä se oli poistettu jo paljon aiemmin.

### 11.2.5 Pumppujen ristiriidat

Asettamalla testattavan pumpun indikointi pysyvästi tiettyyn tilaan: pois tai päälle. Hälytys ei joko mennyt koskaan päälle tai hälytys ei koskaan poistunut, riippuen testivaiheesta. Testaamalla tämä vaihe nähtiin, että testerit tulkitsee hälytysten tilat oikein. Testaus toimii juuri niin kuin sen kuuluukin, eli testerit antaa hylätyn jossain vaiheessa.

### 11.2.6 Paineiden kärkitietojen testaus

Irrottamalla testattavaan verkostoon kytketyn painemittauksen, riippuen valitusta kärkitiedon tyypistä, hälytys ei koskaan käynnistynyt tai sammunut, jolloin testeri antoi hylätyn kyseisestä vaiheesta. Samalla tavalla, kuin kappaleessa 6.1.5 pumppujen ristiriitojen testivaiheen testauksessa, suorittamalla tämä testi nähdään, että painekärkitietojen hälytykset tulkitaan oikein. Testivaihe suoriutui juuri niin kuin pitikin.

### 11.2.7 Pumppujen kärkitietojen testaus

Samalla tavalla kuin ristiriitahälytysten testauksen häirinnässä, indikointi voidaan asettaa tiettyyn tilaan, jolloin hälytys ei koskaan käynnistynyt tai sammunut ja testeri antaa tästä hylätyn. Testausta toteutettaessa huomattiin sen toimivan niin kuin pitääkin. Testausohjelma antaa aina hylätyn, koska hälytys ei ole oikeassa vaiheessa aktiivinen tai epäaktiivinen.

### 11.2.8 Antureiden muunnostaulukkojen testaus

Ottamalla mikä tahansa mittaustieto irti, tämä vaihe pitäisi tulla hylätyksi missä tahansa testauksen vaiheessa. Testattaessa tätä vaihetta, tuli ilmi kaksi ongelmaa. Ensimmäinen ongelmista oli virhe koodissa, jossa annettu mittaustuloksen alue oli huono ja se sopi irrotetun anturin mittaustuloksen arvoon tietyllä muunnostaulukolla. Tässä vaiheessa testeri silti hylkäsi testivaiheen. Toinen ongelma oli se että simuloitaessa Ni1000lg- ja Pt1000-mittaelementtejä, mittaustulos vaihtui uudella säätimellä. Mittaustulo jännitteen nähden oli siis liian herkkä.

### 11.2.9 Huonekompensoinnin testaus

Huonekompensoinnin testausta pystyi häiritsemään ottamalla mittauksen pois tai muuttamalla kompensoinnin suhteita. Jos otti mittauksen pois, mittauspiste antoi arvon  $-50^{\circ}\text{C}$ , jolloin lämmitysverkoston asetusarvo nousi huonekompensointiin määritettyyn maksimiin. Jos taas muutti huonekompensoinnin vaikutusta asetuksista, lämmitysverkoston menoveden asetusarvo meni eri alueelle kokonaan ja testeri hylkäsi sen. Näistä voitiin päätellä, että testeri toimii niin kuin kuuluukin.

### 11.3 Löydetyt viat testerin normaalisuorituksissa

Testerä tehdessä tuli tehtyä useita virheitä koodissa. Osa vioista tuli huomattua heti kun vaihetta ensimmäistä kertaa testattaessa tai kun testausta suoritettiin kokonaisuudessaan. Tässä kappaleessa kuvataan koodivikoja, jotka ovat enemmän ja vähemmän haitallisia ja vaikeita korjata sekä osaamiseen liittyviä ongelmia IEC-koodin tuottamisessa.

## 12 Lopputulos

Geniuksen testauslaitteistoa tuotettaessa ilmeni useampi ongelma ja niitä ratkaistaessa tuli opittua hyvin paljon monesta asiasta. Tässä kappaleessa arvioidaan työn tulos ja todetaan tämän toimivuus. Kuva 14olevassa taulukossa on oma arviointi testausvaiheista asteikolla 1-5 ja kappaleessa on myös kommentoitu vaiheista, joissa voisi olla jotain parannettavaa.

Suoritettava vaihe	Suorituksen arviointi
Normaalisimulaatio	4
Oletusasetuksien tarkistus	4
Ulkoanturin rikkoutumistesti	5
Verkostojen säädöt	3
Pumppujen ristiriidat	3
Paineiden ja pumppujen kärkitiedot	3
Muunnostaulukkojen testaus	3

Kuva 14. Taulukko testausvaiheiden suorituksen arvioinnista

Kokonaissuorituksena testilaitteistosta tuli käytännöllinen järjestelmä, joka säästää aikaa testauslaitteiston käyttäjältä, kun testataan uutta versiota Geniuksesta. Testauslaitteistoon voisi lisätä erillisiä vaiheita, joita ei tiettyjen viiveiden takia haluta suorittaa aivan joka kerta.

## 12.1 Testivaiheiden arviointi ja mahdolliset parannukset

### 12.1.1 Normaalisimulaatio

Normaalisimulaatio toimii hyvin, kun säätimen elektroniset ominaisuudet eivät muutu oleellisesti. Ongelmana voi olla mittausarvon muuttuminen, vaikka ohjelmallisesti simuloidaan samaa jännitettä. Syöttäessä tiettyä jännitettä, mittauksen arvo voi heittää vähän, jonka seurauksen normaalisimulaation simuloima mittaustulos on aina asetusarvon yhdellä puolella. Tämä on hyvin pieni ongelma simulaation tuottaman suuremman eron ansiosta, joka suoritetaan venttiilin ollessa alle 30 % tai yli 70 %. Parempi tapa suorittaa simulointi voisi olla mittaustuloksen skaalaus jollain metodilla ja tehdä asetusarvoa lähenevä simulointi.

### 12.1.2 Oletusasetuksien tarkistus

Oletusasetuksien tarkistus toimii hyvin, mutta siinä voisi olla parempi tapa määrittää normaaliarvot. Tällä hetkellä tarkistettavien arvojen vertaukset ovat määritetty koodiin pysyvästi, mutta ne voisi olla määritettävissä. Syynä on se, että jos säätimeen tehdään uusi ohjelma ja siinä on poikkeavia asetusarvoja tai paremmetrejä, pitää testeriin tehdä muutokset tähän koodiin. Toinen vaihtoehto on tarkistaa, että arvot eivät ole tyhjä, eli nolli, mutta silloin ei oteta huomioon asetuksia, joilla on minimi- tai maksimi-arvot. Niin kauan, kun arvot pysyvät samana, testausvaihe toimii hyvin ja ongelmitta ja on ajallisesti nopein kaikista testivaiheista.

### 12.1.3 Verkostojen säädöt sekä menoantureiden rikkoutuminen

Verkostojen säädöt toimivat tarpeeksi hyvin. Tämä testausvaihe voisi olla myös parempi, jos käytettäisiin ennalta määritettyjen mittausarvojen sijaan loogista erotusta, joka määritetään tämän testausvaiheen alussa. Toinen parannus voisi olla, että testeri tarkistaa säädön hyvyden ja mittaa sen käyttämän ajan ja tulkitsee, toimiiko säätö optimaalisesti

testausympäristössä. Kuitenkin testausvaiheella näkee selvästi, ohjaavatko venttiilit oikeaan suuntaan vai eivät ja etteivät venttiilit ohjaa liian hitaasti.

Verkostojen menoantureiden rikkoutumisen testaus toimii yhtä hyvin ja samoilla pienillä puutteilla kuin perussäädötkin. Ne pohjautuvat samaan koodiin, joten niissä ei ole juurikaan eroja.

#### 12.1.4 Muunnostaulukkojen testaus

Muunnostaulukkojen testaus toimii, mutta siinä on liikaa riippuvuutta aina yksilöllisen säätimen elektroniikkaan. Tietyillä säätimillä Pt1000- ja Ni1000lg-muunnostaulukot voivat poiketa kymmenillä asteilla verrattuna toiseen säätimeen. Tämä vaikeuttaa testausta, koska testeri käyttää tiettyä hyväksyttyä aluetta mittaustulokselle, jolloin eri säädin saattaa olla kunnossa, mutta ei testerin mukaan ole hyväksytty.

## Lähteet

- 1 Sähkötieto ry                      2012                      Rakennusautomaatiojärjestelmät                      Es-  
poo: Sähköinfo Oy                      Luettu 8.7.2014
- 2 Suomen LVI-liitto ry                      2001                      Rakennusten lämmitys                      Kouvola: Olli Sep-  
pänen                      Luettu 9.7.2014
- 3 The Institution of Engineering and Technology 2007                      Programming In-  
dustrial Control System Using IEC 1131-3                      UK: R.W. Lewis                      Luettu  
4.1.2015

(Käytä jompaakumpaa yllä olevista lähteiden luettelointitavoista.)

## Testausvaiheiden toteutuksesta malli IEC –muodossa

(\*Testausvaiheen muuttujat\*)

VAR\_INPUT

Clear : int;

END\_VAR

VAR\_OUTPUT

Ready: int;

END\_VAR

VAR

allowNext : BOOL;

Vaihelaskuri : int;

odotusviive : ton;

odotus : int;

(\*Testausvaiheissa on myös muita muuttujia, kuten luku/kirjoitus sekä raportinkirjoitus functionblockit.\*)

(\*Jokaisella testausvaiheella on myös yksilöllisiä muuttujia\*)

END\_VAR

(\*Itse testaus\*)

IF Vaihelaskuri = 0 THEN

Ready:=1; (\*Ready on 1, jolloin testaus on käynnissä \*)

(\*Ready-muuttujan arvo kirjoitetaan myös käyttöliittymän tilateksteihin\*)

(\*Tehdään alustus ja valmistellaan testeriä sekä geniusta\*)

END\_IF;

IF Vaihelaskuri = 1 THEN

odotusviive.IN:=(odotus = 1);

odotusviive.PT:=DINT\_TO\_TIME(3\*1000);

odotusviive();

odotus:=1;

(\*Odotetaan kolme sekuntia, ennen kuin tehdään mitään\*)

```
IF odotusviive.Q = TRUE THEN
```

```
    odotus:=0; (*Tämän jälkeen tämä vaihe voidaan suorittaa uudestaan*)
```

```
    (*Tehdään tarvittavat tarkistukset*)
```

```
    IF NOT_OK THEN
```

```
        Retrylaskuri:=Retrylaskuri+1;
```

```
        (*Kirjoitetaan virhe vikalokiin*)
```

```
        (*Joissain testauksissa hypätään yksi vaihe taaksepäin*)
```

```
    ELSE
```

```
        allowNext:=TRUE;
```

```
    END_IF;
```

```
END_IF;
```

```
END_IF;
```

```
(*Suoritetaan monta samalla periaatteella koodattua vaihetta, kunnes tullaan viimeiseen vaiheeseen*)
```

```
IF Vaihelaskuri = 8 THEN
```

```
(*Tehdään viiveet ja tarkistukset jne.*)
```

```
    IF NOT_OK THEN
```

```
        Retrylaskuri:=Retrylaskuri+1;
```

```
        (*Kirjoitetaan virhe vikalokiin*)
```

```
    ELSE
```

```
        Ready:=2;
```

```
        (*Kun Ready-muuttujan arvo on 2, se tarkoittaa että testivaihe on hyväksytty*)
```

```
    END_IF;
```

```
END_IF;
```

```
IF allowNext = TRUE THEN
```

```
    Vaihelaskuri:=Vaihelaskuri+1;
```

```
    allowNext := FALSE;
```

```
(*Tällä tehdään varsinainen testausvaiheen vaihto. Vaihdon voisi myös manuaalisesti tekemällä muuttujan "Vaihelaskuri" inkrementointi erikseen jokaisessa kohdassa, mutta
```



tuottamisvaiheessa vaiheiden määrä saattoi lisääntyä, vähentyä tai niiden järjestys muuttua, joten oli helpompi tuottaa yksi yhteinen ehto ja muuttuja tämän asian toteuttamiseen.\*)

END\_IF;

IF Retrylaskuri >= 5 THEN

(\*Jos uudelleenyrityksiä on ollut liikaa, testaus lopetetaan kesken ja jatketaan eteenpäin\*)

(\*Kirjoitetaan lokiin, että testausvaihe ei ollut OK \*)

(\*Palautetaan säädin normaaliarvoihinsa\*)

END\_IF;

IF Ready = 2 AND Retrylaskuri < 5 THEN

(\*Kirjoitetaan lokiin, että testaus on OK \*)

(\*Palautetaan säädin normaaliarvoihinsa\*)

END\_IF;

IF Clear = 1 THEN

(\*Tyhjennetään kaikkien muuttujien arvot\*)

(\*Tämä tehdään vain, kun halutaan valmistautua uuteen testaukseen \*)

END\_IF;