



Sampsa Snellman

Konttisekoittimen automaatiojärjestelmän modernisointi

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Sähkö- ja automaatiotekniikka

Insinöörityö

27.10.2024

Tiivistelmä

Tekijä: Sampsa Snellman
Otsikko: Konttisekoittimen automaatiojärjestelmän modernisointi
Sivumäärä: 54 sivua + 4 liitettä
Aika: 27.10.2024

Tutkinto: Insinööri (AMK)
Tutkinto-ohjelma: Sähkö- ja automaatiotekniikka
Ammatillinen pääaine: Automaatiotekniikka
Ohjaajat: Automaatioinsinööri Olli Suonpää
Lehtori Reijo Leinonen

Insinööriyön tarkoituksena oli toteuttaa konttisekoittimen automaatiojärjestelmän modernisointi lääketeollisuuden yritykselle Orion Oyj. Työn tavoitteena oli valita uudet automaatiojärjestelmän komponentit ja toteuttaa ohjelma konttisekoittimen uudelle PLC:lle ja operointipaneelille, jotka käyttöönoton jälkeen mahdollistavat luotettavamman, turvallisemman ja kustannustehokkaamman prosessilaitteen.

Siemensin SIMATIC S7-1500 -sarjan ohjelmoitava logiikka (PLC) CPU 1515-2 PN löytyi Orionin varastosta, jonka yhteensopivuutta konttisekoittimen vaatimuksille täytyi vain tarkastella. Tarkasteltu PLC valittiin, ja lisäksi uudeksi HMI-paneeliksi (SIMATIC HMI TP1200 Comfort) valittiin kooltaan merkittävästi suurempi kuin vanha malli. Muutkin Siemensin komponentit valittiin yhteensopivuus edellä.

PLC:n ohjelma ja HMI-paneelin käyttöliittymä toteutettiin Siemensin TIA Portal -ohjelmistossa, jossa vanha ohjelma migroitiin uudelle PLC:lle ja käyttöliittymä nykyaikaistettiin. TIA Portal -ohjelmistolla toteutettiin myös reseptien hallintajärjestelmä, joka mahdollistaa automaattisen sekoitusparametrien asettamisen tuotekohtaisesti tuotetai eränumeron perusteella. Lisäksi TIA Portal -ohjelmistolla tehtiin valmiudet konttisekoittimen liittämiseksi Orionin tehdasjärjestelmiin, jotka ovat käyttäjienhallintapanelin, verkkolevyt, etäyhteys operointipaneelille ja varmennettu aikapalvelin.

Lopuksi ohjelman kaikki toiminnot lukuun ottamatta tehdasjärjestelmäliitoksia testattiin onnistuneesti käyttäen Siemensin S7-PLCSIM-simulointityökalua ja ohjelmallisesti toteutettua simulointiympäristöä. Tehdasjärjestelmäliitokset tullaan kuitenkin testaamaan ennen varsinaista käyttöönottoa fyysisellä PLC:llä ja operointipaneelilla. Insinööriyön tuloksena syntyi valmis ohjelma, joka ladataan konttisekoittimen uudelle PLC:lle sen käyttöönoton yhteydessä vuonna 2025 kesällä.

Avainsanat: ohjelmoitavat logiikat, käyttöliittymät, automaatiojärjestelmät, automaatio, ohjelmointi

Tämän insinööriyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla. Insinööriyössä on käytetty OpenAI:n ChatGPT:n versioita 4o kieliasun tarkistamisessa ja viimeistelyssä. Lisäksi kyseistä tekoälyä on käytetty tiivistelmän kääntämisessä englanninkieliseksi ja alt-tekstien generoimisessa. Insinööriyöntekijällä on kaikki vastuu työn sisällöstä.

Abstract

Author: Sampsa Snellman
Title: Modernization of the Container Mixer Automation System
Number of Pages: 54 pages + 4 appendices
Date: 27 October 2024

Degree: Bachelor of Engineering
Degree Programme: Electrical and Automation Engineering
Professional Major: Automation Engineering
Supervisors: Olli Suonpää, Automation Engineer
Reijo Leinonen, Senior Lecturer

The purpose of the engineering thesis work was to modernize the automation system of a container mixer for the pharmaceutical company Orion Oyj. The objective was to select new components for the automation system and develop a program for the container mixer's new PLC and HMI panel, enabling a more reliable, safer, and cost-efficient machine after the implementation.

A Siemens SIMATIC S7-1500 series programmable logic controller (PLC) CPU 1515-2 PN was found in Orion's inventory, and its compatibility with the container mixer's requirements was evaluated. The PLC was selected, and a significantly larger HMI panel (SIMATIC HMI TP1200 Comfort) was chosen than the previous model. Other Siemens components were selected based on compatibility.

The PLC program and HMI panel interface were developed using Siemens TIA Portal software, where the old program was migrated to the new PLC, and the user interface was modernized. A recipe management system was also implemented with TIA Portal software, allowing automatic setting of mixing parameters for each product based on its product or batch number. Additionally, preparations were made for integrating the container mixer into Orion's factory systems, including user management server, network drives, remote access to the HMI panel, and a verified time server.

Finally, all functionalities of the program, except for the factory system connections, were successfully tested using Siemens' S7-PLCSIM simulation tool and a software-based simulation environment. The factory system connections will be tested before commissioning with the physical PLC and HMI panel. The result of the engineering thesis work is a commission ready program, which will be loaded onto the container mixer's new PLC during its commissioning in the summer of 2025.

Keywords: programmable logic controllers, user interfaces, automation systems, automation, programming

Sisällys

Lyhenteet

1	Johdanto	1
2	Konttisekoitin ja sen automaatio	2
2.1	Yleiskuvaus	2
2.2	Toiminnallinen kuvaus	3
2.3	Automaatiojärjestelmä	5
2.3.1	Väyläyhteydet	5
2.3.2	Ohjelmitava logiikka	7
2.3.3	I/O-moduulit	7
2.3.4	Taajuusmuuttaja	9
2.3.5	HMI-paneeli	9
2.3.6	Ohjelmisto	10
2.4	Lääketeollisuuden vaatimukset automaatiolle	11
3	Perustelu automaatiojärjestelmän modernisoinnille	12
3.1	Komponenttien elinkaari	12
3.2	Sekoitusreseptien hallintajärjestelmä	14
3.3	Yhteys tehdasjärjestelmiin	14
3.4	Operointipaneelin käyttöliittymän nykyaikaistaminen	16
4	Komponenttien valinta	16
4.1	Ohjelmitava logiikka	18
4.2	I/O-hajautus	18
4.3	Taajuusmuuttaja	19
4.4	HMI-paneeli	21
5	Konttisekoittimen automaatiojärjestelmän ohjelman toteutus	22
5.1	S7-1500-sarjan logiikan ohjelmointi	23
5.1.1	Vanhan ohjelman migraatio	23
5.1.2	HW-konfiguraatio ja verkotus	27
5.2	HMI TP1200 Comfort -paneelin käyttöliittymän toteutus	28
5.2.1	Pohjasivu	30
5.2.2	Pääsivu	30

5.2.3	Prosessisivu	33
5.2.4	Muut sivut	35
5.3	Reseptien hallintajärjestelmän toteutus	36
5.4	Automaatiojärjestelmän liittäminen tehdasjärjestelmiin	42
5.5	Simulointiympäristön luominen ja ohjelman testaaminen	47
6	Käyttöönottoon valmistautuminen ja työn tulokset	51
7	Yhteenveto	54
	Lähteet	55

Liitteet

Liite 1: Ohjelman kääntämisen raportti logiikan migraation jälkeen

Liite 2: Viivakoodin luentaan liittyvien toimintojen toteuttaminen VBScript-ohjelmointikielellä TIA Portal -ohjelmistossa

Liite 3: Eräluettelon käsittelyn toteuttaminen SCL-ohjelmointikielellä TIA Portal -ohjelmistossa

Liite 4: Kehikon käännön testaaminen simuloinnilla prosessisivulta seurattuna

Lyhenteet

AI: *Analog Input*. Analoginen tulo.

AO: *Analog Output*. Analoginen lähtö.

DI: *Digital Input*. Digitaalinen tulo.

DO: *Digital Output*. Digitaalinen lähtö.

GMP: *Good Manufacturing Practices*. Hyvät tuotantotavat.

HMI: Human-Machine Interface. Ihmisen ja koneen rajapinta.

HW: *Hardware*. Laitteisto.

I/O: *Input/Output*. Tulo/lähtö.

MPI: *Multi-point interface protocol*. Tiedonsiirto-protokolla.

PLC: *Programmable Logic Controller*. Ohjelmoitava logiikka.

TIA Portal: *Totally Integrated Automation Portal*. Siemensin ohjelmisto, joka on tarkoitettu esimerkiksi logiikoiden ohjelmointiin.

VNC: *Virtual Network Computing*. Etäyhteysprotokolla.

1 Johdanto

Teknologian kehittyessä vanhojen prosessilaitteiden automaatiojärjestelmät eivät tue uusimpia ominaisuuksia, joilla voidaan parantaa laitteen turvallisuutta, tuottavuutta, luotettavuutta ja alentaa ylläpitokustannuksia. Tämän vuoksi vanhemmat prosessilaitteistot tarvitsevat automaatiojärjestelmän modernisointia. Käytännössä laitteiston automaatiojärjestelmän aivot eli ohjelmoitava logiikka ja operointipaneelien käyttöliittymät sekä niiden ohjelmistot päivitetään nykyaikaiseksi, mikä mahdollistaa uusien ominaisuuksien käyttöönoton. Lisäksi automaatiojärjestelmään liittyvät komponentit päivitetään sopivaksi uuden ohjelmoitavan logiikan kanssa, ja muidenkin komponenttien päivittämistä tarkastellaan varaosasaatavuuden perusteella.

Insinööriyön tarkoituksena on toteuttaa konttisekoittimen automaatiojärjestelmän modernisointi lääketeollisuuden yritykselle Orion Oyj. Konttisekoittimella sekoitetaan kontissa olevaa lääkemassaa pyörittämällä konttia. Konttisekoitinta ohjataan yhdellä operointipaneelilla, johon täytyy manuaalisesti syöttää jokaisen eri tuotteen sekoitusparametrit reseptin mukaisesti. Modernisoinnissa toteutetaan reseptienhallintajärjestelmä, joka mahdollistaa tuotekohtaisten sekoitusparametrien automaattisen asettamisen tuote- tai eränumeron perusteella. Automaatiojärjestelmää ei ole liitettyä mihinkään Orionin tehdasjärjestelmiin, joten modernisoinnin yhteydessä konttisekoittimelle luodaan yhteys käyttäjienhallintapalvelimeen, varmennettuun aikapalvelimeen, verkkolevyille ja etäyhteys sovelukseen. Konttisekoittimen automaatiojärjestelmän ohjelmoitavassa logiikassa oleva ohjelma päivitetään uuteen logiikkaan sopivaksi, joten siihen yhteydessä olevat muutkin komponentit täytyy myös päivittää. Lisäksi operointipaneeli päivitetään uudempaan ja suurempaan malliin, sekä sen käyttöliittymä nykyaikaistetaan.

Konttisekoittimen modernisoinnin suunnittelu on tehty, joten insinööriyössä keskitytään modernisoinnin toteutusvaiheeseen. Insinööriyön aihealueeseen kuuluu uusien komponenttien valinta ja uuden ohjelman sekä käyttöliittymän

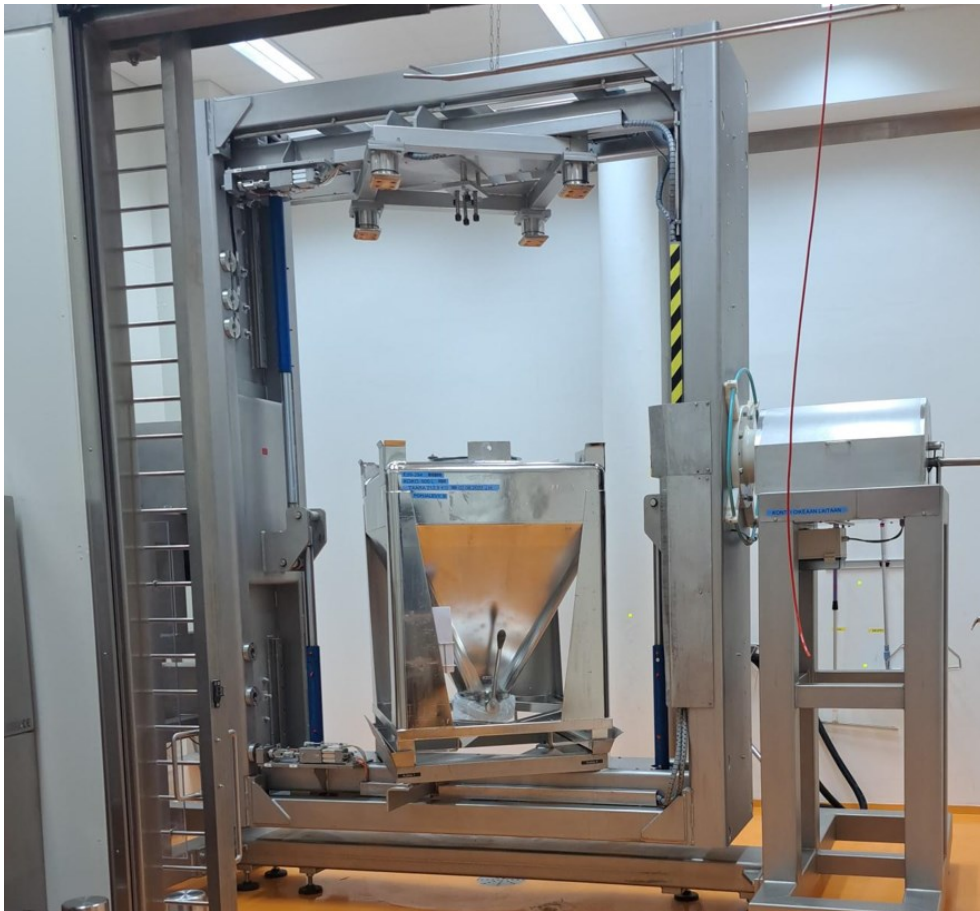
toteuttaminen Siemensin TIA Portal -ohjelmointityökalulla. Konttisekoittimen automaatiojärjestelmän liittämisen tehdasjärjestelmiin voi tehdä vain käyttöönotossa, joten insinööriyössä tehdään tarvittavat ohjelmalliset alkuvalmistelut sitä varten. Käyttöönotto ei kuulu insinööriyön aihealueeseen. Ohjelman toimivuus testataan simuloimalla ilman fyysisiä komponentteja. Insinööriyön tuloksena syntyy käyttöönottovalmis ohjelma, joka sisältää operointipaneelin käyttöliittymän, PLC:n ohjelman, reseptien hallintajärjestelmän ja valmiuden yhdistyä tehdasjärjestelmiin.

2 Konttisekoitin ja sen automaatio

2.1 Yleiskuvaus

Modernisoinnin kohteena on vuonna 2009 valmistettu konttisekoitin. Konttisekoittimella sekoitetaan lääkemassaa sisältävää konttia 360 astetta pystysuuntaisesti. Konttisekoittimella voi sekoittaa neljää eri konttikokoa, jotka ovat 2100 litraa, 1500 litraa, 800 litraa ja 300 litraa. Pienimmällä kontilla täytyy käyttää adapteria, jotta sitä pystyy turvallisesti sekoittamaan. Tämän vuoksi automaatiojärjestelmä tunnistaa vain kolme eri konttikokoa. (1.)

Sekoitusprosessi suoritetaan seuraavanlaisesti: Operaattori tuo kontin välitilaan, jossa hän skannaa kontin pinnassa olevan tuotteen ja eränumeron viivakoodit. Tämän jälkeen kontti siirretään laitteen kehikkoon, minkä jälkeen operaattori poistuu huoneesta ja sulkee lukollisen turvaportin. Seuraavaksi operaattori syöttää tuotteen sekoitusreseptin mukaiset parametrit manuaalisesti operointipaneelille ja käynnistää sekoitusohjelman. Ensimmäiseksi ohjausjärjestelmä puristaa moottoriohjattujen kelkkojen avulla kontin tukevasti kiinni kehikkoon. Tämän jälkeen laite pyörittää lääkemassaa sisältävää konttia 360 astetta pystysuuntaisesti. Sekoitusohjelman loputtua operaattori avaa turvaportin ja vie kontin seuraavaan lääkevalmistuksen prosessilaitteeseen. Kuvassa 1 näkyy konttisekoittimen kehikko, johon on lastattu 800 litran kontti. (1.)

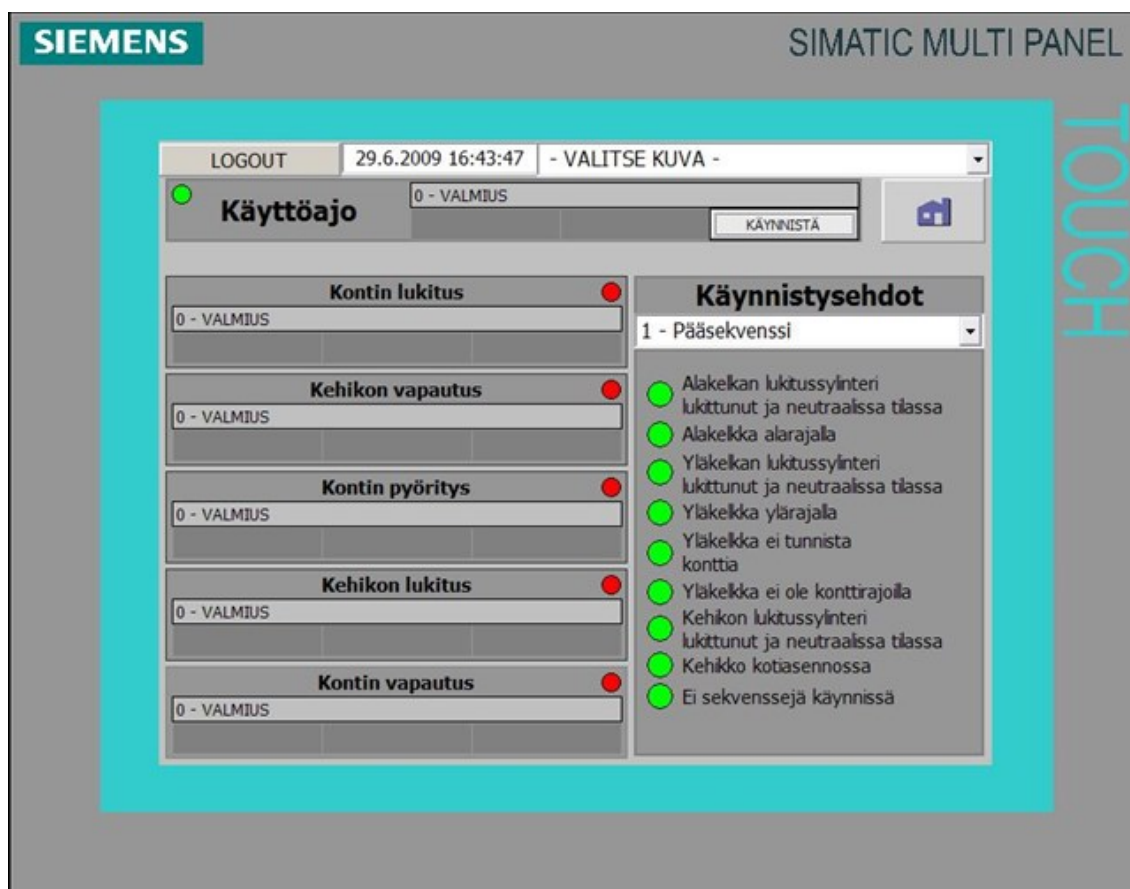


Kuva 1. Konttisekoitin, jonka kehikkoon on lastattu kontti.

2.2 Toiminnallinen kuvaus

Konttisekoitinta voidaan ajaa automaatti- tai manuaalijolla. Automaattijossa sekoitusohjelma toimii sekvenssimäisesti eli laite suorittaa toimintoja askel kerrallaan. Laitteessa on pääsekvenssi ja sen alapuolella on osasekvenssejä. Sekoitusohjelman käynnistäminen edellyttää aloitusehtojen täyttymistä, kuten esimerkiksi "Yläkelkka ei tunnista konttia" -ehdon täyttymistä (Kuva 2). Pääsekvenssi ohjaa ja valvoo osasekvenssejä. Askeleet toimivat pää- ja osasekvensien sisällä, ja askeleiden sisällä tapahtuu varsinainen toimilaitteiden ohjaus. Pääsekvenssin yksi askelista voi olla esimerkiksi osasekvenssin käynnistäminen ja sen valvominen. Yksi osasekvenssin askeleista voi olla taas esimerkiksi moottorin käynnistäminen.

Askeleessa tehtävien toimintojen täytyy onnistua määritetyn ajan sisällä, jotta voidaan siirtyä seuraavaan askeleeseen. Muuten ohjelma ilmoittaa virheestä ja menee keskeytystilaan. Kuvassa 2 on operointipaneelin näkymä, jossa näkyvät pääsekvenssin käynnistysehdot ja osasekvenssien nimet. Lisäksi operaattori voi käynnistää tuotantoajan kyseiseltä sivulta. Manuaaliajolla vain laitehuoltaja tai Admin-käyttäjä voi ohjata kaikkia toimilaitteita vapaasti, mutta kontin pyörittämiseen tarkoitettua moottoria ei voi käyttää, elleivät turvaportti ja kelkat ole lukittuna. (2.)



Kuva 2. Operointipaneelissa näkyvä sekvenssinäkymä (2).

Kontin tulee olla tukevasti kiinni kehikossa ja turvaportin täytyy olla suljettuna ennen sekoitusohjelman käynnistämistä. Ennen sekoituksen käynnistämistä operaattori syöttää manuaalisesti operointipaneelille sekoitusreseptin mukaiset parametrit, kuten kehikon pyöritysnopeuden ja -ajan. Operaattorin käynnistäessä sekoitusohjelman laite lukitsee automaattisesti turvaportin ja puristaa ylä-

ja alakelkan kiinni konttiin käyttäen moottoriohjattuja sähköisiä sylintereitä. Lisäksi laitteessa olevat paineilmaohjatut sylinterit lukitsevat kelkat paikoilleen. Laite tunnistaa kontin koon etäisyyslaserin avulla ja osaa tämän tiedon perusteella siirtää sekä lukita kelkat oikeaan kohtaan. (2; 3.)

Konttia ja kehikkoa pyörittää 22 kW:n kartiopyörävaihdemoottori, joka on myös oikosulkumoottori. Sen nopeutta säädetään taajuusmuuttajalla, ja kierrosnopeuden pystyy säätämään välille 1–20 kierrosta minuutissa. Hätäseis-painikkeesta tai operointipaneelista voidaan keskeyttää tuotantoajo milloin tahansa. Konttia pyöritetään sille määritetyn pyörimisajan verran, minkä jälkeen laite ajaa itsensä takaisin alkuasentoon. Pyörivän kehikon akselissa oleva paikoitusanturi tunnistaa kontin kulman maata vasten ja auttaa laitetta palautumaan takaisin alkuasentoon. (3; 2.)

Operointipaneelissa on myös sivu aktiivisille hälytyksille sekä hälytyshistoria. Operointipaneelin käyttöliittymä ilmoittaa hälytyksellä, jos toimilaitte tai anturi hajoaa. Hälytyksen vakavuuden perusteella automaatiojärjestelmä tekee konttisekoittimelle tarvittavat turvatoimet, kuten kontin pyöriksen pysäyttämisen. (2.)

2.3 Automaatiojärjestelmä

Tässä luvussa käsitellään konttisekoittimen automaatiojärjestelmän rakennetta ja siihen kuuluvia laitteiden toiminnallisuuksia ennen modernisaatiota. Automaatiojärjestelmä luo yhtenäisen kokonaisuuden konttisekoittimen ohjaukseen ja valvontaan. Lisäksi käsitellään ohjelmistot, joilla on toteutettu ohjelmoitavan logiikan ohjelma ja käyttöliittymän rakentaminen.

2.3.1 Väyläyhteydet

Väyläyhteyksillä tarkoitetaan kommunikaatiota automaatiojärjestelmän ja kenttälaitteiden välillä. Väylissä tieto kulkee digitaalisessa muodossa, mikä mahdollistaa useiden eri laitteiden ohjaamisen ja valvomisen yhdellä kaapelilla.

Konttisekoittimen automaatiojärjestelmässä käytetään kolmea eri väyläprotokollaa, jotka ovat PROFIBUS, PROFINET ja MPI. (4; 5.)

PROFIBUS on yksi ensimmäisistä teollisuudessa käytetyistä väyläprotokollista, joka mahdollistaa kaksisuuntaisen liikenteen kenttälaitteiden ja master-yksikön välillä. Käytännössä master-yksikkö ohjaa tai pyytää tietoa kenttälaitteelta. (6.) Konttisekoittimen automaatiojärjestelmässä ohjelmoitava logiikka toimii master-yksikkönä. Ohjelmoitavan logiikan lisäksi PROFIBUS-väylään on liitetty kehikkoa pyörittävän moottorin taajuusmuuttaja ja paikoitusanturi. (5).

PROFINET-väyläprotokolla tuli markkinoille PROFIBUS-väyläprotokollaa myöhemmin. PROFINET käyttää tyypillistä Ethernet-teknologiaa, joka on yleisesti käytössä kaikkialla. PROFINET-väyläprotokollaa voidaan hyödyntää järjestelmän ja kenttälaitteiden väliseen yhteyteen, mutta myös järjestelmien väliseen yhteyteen. Tämä tarjoaa laajan sovellettavuuden eri käyttökohteisiin.

Konttisekoittimen muutostyön yhteydessä vuonna 2016 sen pyörivään kehiköön lisättiin WLAN-verkkokortti, joka on liitetty PROFINET-yhteydellä tulo- ja lähtömoduuleihin. Kyseinen WLAN-verkkokortti on langattomasti yhteydessä WLAN-tukiasemaan, joka sijaitsee pyörivän kehikön ulkopuolella. WLAN-tukiasema on taas yhteydessä PROFINET-kaapelilla erilliseen kommunikaatiomoduliin, joka sijaitsee PLC:n kyljessä. Muutoksella voitiin vähentää kaapelien määrää laitteisiin, jotka sijaitsevat pyörivässä kehikossa (6; 5.)

MPI eli Multi-Point Interface on Siemensin kehittämä väyläprotokolla, jota käytetään erityisesti vanhojen S7-300- ja S7-400-sarjan logiikoiden ja insinööriyöasemien väliseen tiedonsiirtoon. MPI-väyläprotokollaa voidaan käyttää myös yksinkertaisten paneelien ja logiikoiden väliseen tiedonsiirtoon. Insinööriyöasemalla on mahdollista tarkastella ohjelman reaaliaikaista tilaa ja tehdä siihen tarvittavia muutoksia. Uudemmissa logiikoilla kyseinen teknologia on korvattu PROFINET-väyläprotokollalla. Konttisekoittimessa automaatiojärjestelmän ohjelmoitava logiikka on liitetty MPI-yhteydellä operointipaneeliin. (7; 5.)

2.3.2 Ohjelmoitava logiikka

Ohjelmoitava logiikka eli PLC (Programmable Logic Controller) on automaatiojärjestelmän aivot ja muistuttaa rakenteeltaan minitietokonetta. PLC vastaanottaa tietoa antureilta tai rajakytkimiltä ja ohjaa toimilaitteita saamansa tiedon perusteella. PLC:n toiminnalliset ominaisuudet määräytyvät siihen ohjelmoidun ohjelman mukaan, ja ohjelmointi tehdään erityisesti tätä varten suunnitelluilla ohjelmointityökaluilla. Tyypillisesti PLC koostuu keskusprosessointiyksiköstä (CPU) ja kommunikaatiomodulistista. Riippuen PLC:n tyypistä, se voi sisältää myös tulo- tai lähtömoduuleita. (8.)

Konttisekoittimen automaatiojärjestelmässä käytetään logiikkana Siemensin S7-300-sarjan ohjelmoitavaa logiikkaa, jonka tyyppi on CPU 315-2DP. PLC:n kommunikaatiomoduli tukee MPI- ja PROFIBUS-väyläprotokollaa. Kuitenkin konttisekoittimen muutostyön yhteydessä vuonna 2016 täytyi hankkia erillinen kommunikaatiomoduli, joka tukee myös PROFINET-väyläprotokollaa. (5; 9.)

2.3.3 I/O-moduulit

I/O-moduuleilla tarkoitetaan tulo- ja lähtömoduuleita, joihin anturit ja toimilaitteet liitetään. Näiden moduulien kautta PLC pystyy ohjaamaan toimilaitteita tai vastaanottamaan tietoa antureilta. I/O-moduuleita on erityyppisiä, kuten AI-moduuli (analoginen tulomoduli), AO-moduuli (analoginen lähtömoduuli), DI-moduuli (digitaalinen tulomoduli) ja DO-moduuli (digitaalinen lähtömoduuli). (10.) Yhtein I/O-moduulin voidaan kytkeä useita toimilaitteita tai antureita riippuen moduulin tyypistä.

AI-moduuli vastaanottaa antureilta tietoa sähköisessä muodossa esimerkiksi 4–20 mA signaalina. Tämä sähköinen signaali skaalautuu lineaarisesti mitattavan suureen mukaan. Esimerkiksi, jos lämpötila-anturi pystyy mittamaan lämpötilaa 0–100 C ° väliltä ja lämpötila on nykyhetkellä 50 C °, niin tällöin anturi antaa ulos 12mA signaalia, joka on puolessa välissä mitta-alueella. Useimmiten anturit eivät itsessään anna mA signaalia, mutta niihin on kytkettynä laite, joka

pystyy tämän muunnoksen tekemään. Kyseisen muuntimen pystyy myös hankimaan erillisenä laitteena. AI-moduuli muuntaa taas tämän sähköisen signaalin digitaalseksi arvoksi, jolloin PLC pystyy tulkitsemaan anturin antamaa mittausarvoa. (11, s. 33–34.)

AO-moduuli ohjaa toimilaitteita sähköisesti esimerkiksi 4–20 mA signaalilla (11, s. 56–57). Esimerkiksi säätöventtiili voidaan liittää AO-moduuliin, jolloin venttiilin asentoa voidaan säätää asteittain. Säätöventtiili voidaan avata osittain tai kokonaan riippuen PLC:n lähettämästä arvosta. AO-moduuli muuttaa PLC:n lähettämän digitaalisen arvon sähköiseksi, jolloin säätöventtiili pystyy tulkitsemaan ohjauksikäskyä.

DI-moduuli vastaanottaa tietoa kaksivaiheisesti ja DO-moduuli ohjaa kaksivaiheisesti, eli joko ”päällä” tai ”pois”. DI-moduuliin voidaan liittää esimerkiksi painonappi, jolloin mahdolliset arvot ovat 1 tai 0 eli nappi on joko painettuna tai ei. DO-moduulin kautta PLC voi ohjata esimerkiksi venttiiliä siten, että venttiili avataan tai suljetaan täysin. (10.)

Konttisekoittimen automaatiojärjestelmän PLC:n kylkeen on liitettynä yksi DI-, DO- ja AI-moduuli. Lisäksi eriyttynä PLC:stä on I/O-hajautus, johon on liitetty kaksi DO-moduulia ja yksi DI-moduuli. I/O-hajautuksella tarkoitetaan I/O-moduuleita, jotka ei ole sijoitettu suoraan logiikkaan kiinni, vaan ne kommunikoivat PLC:n kanssa väyläyhteyden kautta. Tämä mahdollistaa IO-moduuleiden sijoittamisen lähelle antureita ja toimilaitteita, jolloin kaapelien pituudet pienentyvät. AI-moduuliin on kytketty vain laseretäisyysanturi, jota käytetään kontin koon tunnistamiselle. DI-moduuleihin on kytketty sylinterin asentotietoja, induktiivisia rajakytkimiä ja turvarajoja. DO-moduuleihin on kytketty magneettiventtiileitä, joilla ohjataan sylintereitä. IO-hajautus on liitetty PROFINET-yhteydelle WLAN-verkkokorttiin, kuten luvussa 2.3.1 on kuvattu. (5; 12.)

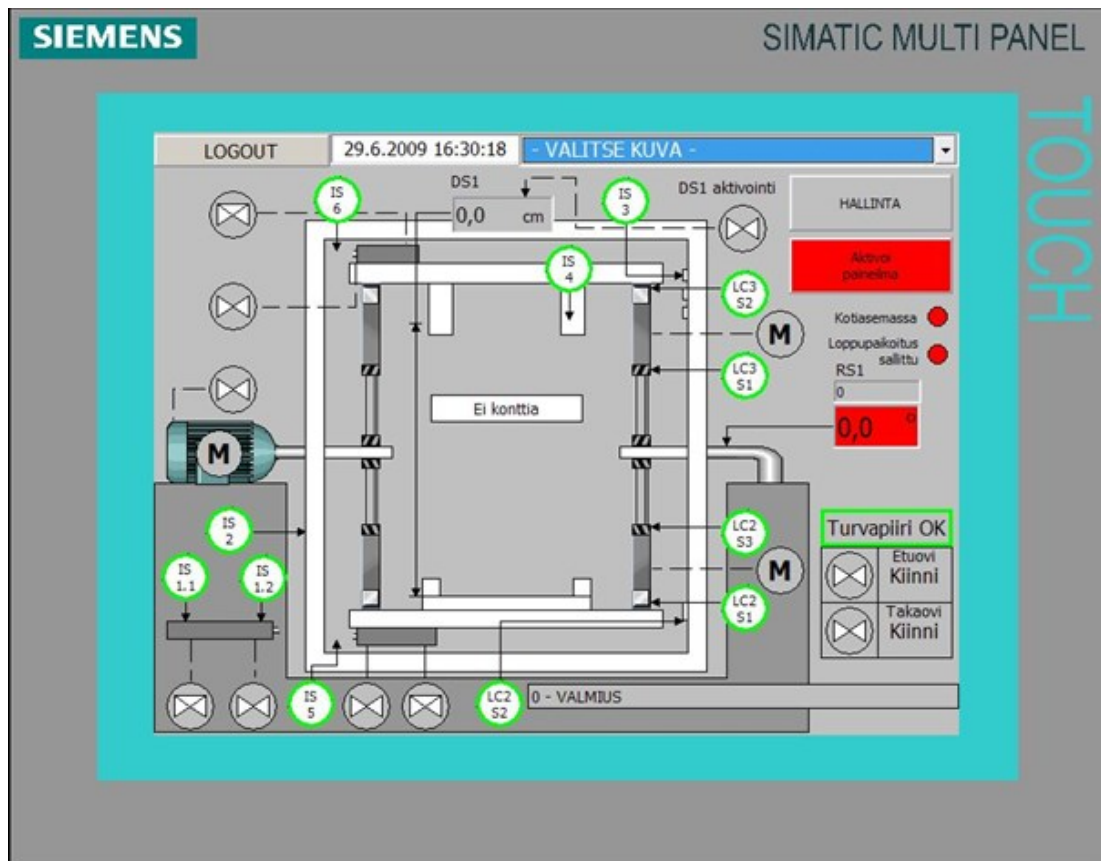
2.3.4 Taajuusmuuttaja

Taajuusmuuttajilla voidaan ohjata vaihtovirtamoottoreita eri nopeuksilla muuttamalla vaihtovirran taajuutta. Konttisekoittimen kehikkoa ja konttia pyörittävää 22 kW:sta moottoria ohjataan Siemensin G120-sarjan taajuusmuuttajalla. G120-sarjan taajuusmuuttaja koostuu kolmesta osasta, jotka ovat tehoyksikkö, ohjausyksikkö ja operointipaneeli. Tehoyksikkö on moottorin virtalähde, joten tehoyksikkö täytyy valita moottorin tehon tarpeen mukaan. Ohjausyksikkö on taajuusmuuttajan aivot ja se täytyy valita tehoyksikköön sopivaksi sekä moottorin käyttötarkoituksen mukaisesti. Ohjausyksikkö säätelee tehoyksikön läpi kulkevaa vaihtovirran taajuutta. Lisäksi ohjausyksikölle voidaan asettaa tuhansia eri parametreja, joilla voidaan saada lisäominaisuuksia taajuusmuuttajan ohjaukseen tai vian diagnosointiin. Konttisekoittimessa ohjausyksikkö on liitettyinä logiikkaan käyttäen PROFIBUS-väylää. Taajuusmuuttajan operointipaneeli on yksinkertainen näyttö, jonka avulla voidaan selata ja asettaa parametreja sekä käyttää apuna vikojen diagnosoinnissa ilman erillistä ohjelmistoa. (13; 5; 14.)

2.3.5 HMI-paneeli

Human-Machine Interface (HMI) tarkoittaa ihmisen ja koneen rajapintaa. Se on käyttöliittymä, jonka avulla käyttäjä voi seurata ja ohjata prosessilaitetta. HMI-laitteena voidaan käyttää joko tietokonetta tai erillistä paneelia. (15.)

Konttisekoittimessa käytetään Siemensin kahdeksantuumaista kosketusnäyttöpaneelia, jonka avulla ohjataan ja valvotaan konttisekoitinta. Paneelille on ladattu ohjelmointityökalulla rakennettu käyttöliittymä, joka sisältää prosessilaitteen ohjaukseen tarvittavat toimet. Kuvassa 3 on esitetty operointipaneelin prosessisivu, josta käyttäjä näkee toimilaitteiden ja anturien tilatiedot. (5; 2.)



Kuva 3. Operointipaneelin prosessisivu (2).

2.3.6 Ohjelmisto

Konttisekoittimen automaatiojärjestelmän PLC on ohjelmoitu käyttäen Siemensin Simatic Manager STEP 7 V5.5 -ohjelmointityökalua. Simatic Manager on Siemensin kehittämä ohjelmointityökalu ohjelmoitaville S7-300- ja S7-400-sarjan logiikoille. Konttisekoittimen PLC:n ohjelma on toteutettu pääosin Ladder-ohjelmointikielellä, joka on visuaalinen ohjelmointikieli ja muistuttaa rakenteeltaan sähköpiirikaaviota. (16; 17, s. 21, 196.) Ladder-ohjelmointikieli on yleinen näkymä erityisesti vanhemmilla logiikoilla (PLC), sillä ennen logiikoita kaikki monimutkaiset ohjaukset suunniteltiin piirikaavioilla ja toteutettiin releohjauksilla. Tämän takia piirikaavioiden piirtämisestä saattoi olla helpompaa siirtyä ohjelmoimaan logiikoita Ladder-ohjelmointikielellä.

Konttisekoittimen operointipaneelin käyttöliittymä on toteutettu käyttäen WinCC Flexible 2008 -ohjelmistoa. WinCC on Siemensin kehittämä alusta teollisuudessa käytettäville käyttöliittymille sekä niiden suunnitteluun ja toteuttamiseen. WinCC mahdollistaa esimerkiksi prosessien visualisoinnin ja ohjaamisen, hälytysten esittämisen näytöllä sekä mittautustietojen keräämisen sisäiselle tallennuslevylle. Lisäksi ohjelmistolla voidaan tuottaa raportteja esimerkiksi tuotantoerän jälkeen. WinCC Flexible 2008 -ohjelmisto koostuu kahdesta osasta: WinCC Flexible Runtime, joka toimii käyttöliittymän alustana, ja WinCC Flexible Engineering System, jolla käyttöliittymä konfiguroidaan ja rakennetaan. (16; 18, s. 19–20.)

2.4 Lääketeollisuuden vaatimukset automaatiolle

Konttisekoittimen automaatiojärjestelmän modernisointi toteutetaan yritykselle Orion Oyj. Orion on kansainvälinen lääketeollisuuden yritys, joka tuottaa ja kehittää lääkkeitä sekä ihmisille että eläimille. Orionilla työskentelee yli 3500 henkilöä ympäri maailmaa. (19.)

Orionilla noudatetaan GMP:tä (Good Manufacturing Practice), joka tarkoittaa hyviä tuotantotapoja. GMP on Euroopan komissiossa määritelty ohjeistus, jonka tavoitteena on varmistaa potilasturvallisuus ja tuotteiden laatu lääketeollisuudessa. GMP kattaa kaikki lääketuotannon osa-alueet, henkilöstön koulutuksista laitteiden luotettavuuteen. GMP:n keskeinen periaate on tarkka dokumentointi, jotta lääkkeisiin liittyvän tiedon katoaminen tai muuttuminen voidaan estää kaikissa tuotannon vaiheissa. (20, s. 12.) Automatisoidulla prosessilaitteella tämä voi tarkoittaa esimerkiksi sitä, että jokainen operointipaneelilla tehty toiminto ja sen suorittaja tallentuu laitteen lokiin, eikä tietoja voi muokata tai poistaa.

Siemens on kehittänyt WinCC-ohjelmistoille Audit Trail -toiminnon, joka tukee GMP-vaatimusten täyttymistä. Tämä toiminto helpottaa jokaisen HMI-paneelille kirjautuneen henkilön toimintojen seuraamista. Audit Trail -toiminnosta löytyy myös ominaisuus, jolla esimerkiksi kriittisten laiteparametrien muuttamisen yhteydessä järjestelmä pyytää kuittauksen, ja vasta kuittauksen jälkeen

järjestelmä toteuttaa parametrin muuttamisen. Lisäksi Audit Trail -toiminnon asetuksista voidaan määrittää, että kuittauksen lisäksi käyttäjän tulee syöttää käyttäjätunnus ja salasana, jotta järjestelmä hyväksyy muutoksen. (21.)

3 Perustelu automaatiojärjestelmän modernisoinnille

Konttisekoittimen modernisoinnille täytyy olla perusteltu tarve. Yleisesti ottaen laitteen ikä on merkittävä tekijä, kun arvioidaan potentiaalisia modernisointikohteita, mutta lopullinen päätös modernisoinnista edellyttää tarkempia perusteluja. Seuraavissa luvuissa (3.1–3.5) kerrotaan tarkemmin, miksi konttisekoittimen modernisointi on ajankohtainen ja tarpeellinen.

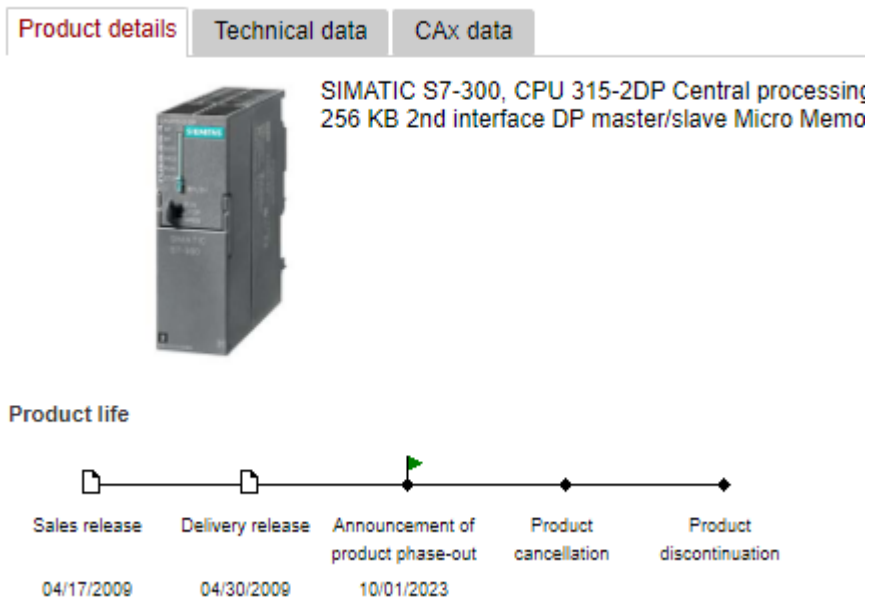
3.1 Komponenttien elinkaari

Komponenttien elinkaarella tarkoitetaan ajanjaksoa tuotteen julkaisusta sen valmistuksen lopettamiseen (22). Modernisointia tehdessä on tärkeää miettiä komponenttien elinkaarta, koska tällöin välttytään tulevaisuudessa yllättäviltä komponenttihankinnoilta ja niiden saatavuusongelmilta. Hyvä komponenttien elinkaa- ren hallinta näkyy positiivisesti laitteen ylläpitokustannuksissa.

Konttisekoittimen automaatiojärjestelmään kuuluu lähinnä Siemensin tuotteita, joten komponenttien elinkaaria tarkastellaan Siemensin verkkosivuja käyttäen. Voi myös neuvotella Siemensin kanssa, jos tarvitsee tarkempaa tietoa komponentin elinkaaren vaiheesta. Siemensin tuotteiden elinkaari koostuu viidestä vaiheesta, joista ensimmäinen on tuotteen myymisen aloittaminen (Sales release). Toinen vaihe on tuotteen tuottaminen (Delivery release) eli tällöin tuotetta valmistetaan aktiivisesti ja sitä saa tilattua jälleenmyyjiltä normaalisti. Kolmannessa vaiheessa ilmoitetaan tuotteen valmistuksen lopettamisesta lähitulevaisuudessa (Announcement of product phase-out), ja tuotteen mainostaminen lopetetaan myös. Neljäs vaihe on tuotteen aktiivisen valmistuksen lopettaminen (Product cancellation). Tämä tarkoittaa tuotteen siirtymistä varaosaksi, jolloin sitä saa ostettua vain tilaustuotteena. Lisäksi tuote on kalliimpi, koska tuote joudutaan erikseen valmistamaan. Viides ja viimeinen vaihe on tuotteen

valmistuksen lopettaminen kokonaan (Product discontinuation). Tuotteen elinkaari on päätynyt ja sille ei saa enää tuotetukea Siemensiltä. Tyypillinen elinkaari Siemensin tuotteilla on 22 vuotta. Kuvassa 4 esitetään konttisekoittimen automaatiojärjestelmässä olevan PLC:n elinkaari ja missä vaiheessa se on elinkaarta. (22.)

CPU315-2DP, 256 KB



Kuva 4. Konttisekoittimen PLC:n elinkaari esitettynä. PLC:n elinkaari on vaiheessa kolme (9).

Yleisesti aktiivisessa vaiheessa (toinen vaihe) olevaa komponenttia ei tarvitse uusida, ellei järjestelmän uusi PLC ole yhteensopiva kyseisen komponentin kanssa. Konttisekoittimen modernisoinnissa tarkastellaan niiden komponenttien elinkaarta, jotka ovat jollain tasolla liitoksissa PLC:n ohjelmaan. Muut komponentit, kuten releet, ovat yksinkertaisia sähköohjattuja kytkimiä, joten niiden tilalle voi elinkaaren päättymisen jälkeen vaihtaa uudemman tai eri merkkisen mallin ilman ohjelmamuutoksia.

3.2 Sekoitusreseptien hallintajärjestelmä

Jokaisella tuotteella on oma sekoitusresepti, jonka mukaan konttia pyöritetään. Tuotteen sekoitusresepti määrittää, millä nopeudella ja kuinka kauan kontti pyörii. Nykyisessä järjestelmässä operaattorin täytyy etsiä tuotetta vastaava sekoitusresepti Excel-taulukosta ja syöttää sen sisältämät parametrit laitteelle, mikä vie merkittävästi aikaa. (1.) Lisäksi inhimillisten virheiden takia on mahdollista, että syötetään väärät sekoitusparametrit.

Konttisekoittimen modernisoinnin yhteydessä järjestelmään voidaan lisätä Siemensin tarjoama reseptien hallintajärjestelmä. Hallintajärjestelmään syötetään kaikki reseptit, ja se tallentaa ne automaattisesti. Järjestelmä voidaan ohjelmoida etsimään tuotekoodia vastaava sekoitusresepti ja asettamaan sekoitusparametrit automaattisesti laitteelle, mikä säästää operaattorin aikaa ja tehostaa tuotantoa. (23, s. 5.)

3.3 Yhteys tehdasjärjestelmiin

Tehtaan digitalisointi ja tiukentuneet tiedonhallintavaatimukset ovat perusteltuja syytä konttisekoittimen modernisoinnille. Konttisekoittimen automaatiojärjestelmä ei ole tällä hetkellä kytkettynä Orionin tehdasverkkoon, ja vanha PLC ei tue liittämistä tehdasjärjestelmiin. Edellä kerrotaan, mihin järjestelmiin konttisekoitin on syytä liittää ja mitä hyötyä niistä on.

Tehtaan verkkolevyt

Konttisekoittimen automaatiojärjestelmän liittäminen tehdasverkkoon mahdollistaa tehtaan verkkolevyillä olevien kansiodien jakamisen operointipaneelin käyttöön. Tämän myötä esimerkiksi hälytyshistoria- ja Audit Trail -tiedoston pystyy suoraan siirtämään tai kopioimaan tehtaan verkkolevyille. Lisäksi sekoituksen jälkeen tulostettava paperinen raportti voidaan jatkossa tulostaa PDF-tiedostona suoraan tehtaan verkkolevyille. GMP paranee huomattavasti, kun kriittinen tieto

säilytetään verkkolevyillä, jolloin operointipaneelin tai paikallismuistin hajotessa ei häviä tietoa.

Varmennettu aikapalvelin

Nykyisessä toteutuksessa kellonaika täytyy asettaa manuaalisesti. Ei voida siis varmuudella sanoa, että näyttääkö kello oikein, koska aikaa ei synkronisoida automaattisesti mistään, ja lisäksi täytyy ottaa huomioon kesä- ja talviajat. Uudemmat Siemensin PLC:t tarjoavat mahdollisuutta hakea kellonaika NTP-aikapalvelimelta. NTP (Network Time Protocol) on internetprotokolla, jonka avulla kellon aika haetaan verkon ylitse NTP-palvelimelta. (2; 24; 25.)

Käyttäjienhallintapalvelin

Konttisekoittimessa on käytössä vain paikalliset käyttäjätunnukset eli käyttäjiä voidaan lisätä ja poistaa vain operointipaneelilta käsin. Tämä vaikeuttaa oikeuksien hallinnointia käyttäjien suuren määrän vuoksi, minkä takia konttisekoittimen peruskäyttö ei vaadi kirjautumista tunnuksilla. (2.) Modernisoinnin yhteydessä konttisekoitin liitetään Orionin käyttäjienhallintapalvelimeen, jolloin jokaisen käyttäjän on kirjaututtava henkilökohtaisilla tunnuksillaan operointipaneelille.

Etäyhteys operointipaneelille

Konttisekoittimen operointipaneelin ei ole mahdollista ottaa etäyhteyttä, koska se ei ole yhdistetty verkkoon. Siemensin nykyaikaisilla operointipaneeleilla pystyy ottamaan käyttöön Smart-palvelimen. Tämä mahdollistaa VNC-etäyhteyden ottamisen esimerkiksi toimistolta suoraan laitteen operointipaneelille. Virtual Network Computing eli VNC on yksinkertainen protokolla, jolla pystyy ottamaan etäyhteyden laitteisiin, jotka tukevat kyseistä protokollaa. VNC tarvitsee toimiakseen VNC Viewer -sovelluksen, joka asennetaan laitteelle, jolta halutaan luoda etäyhteys. Lisäksi täytyy asentaa VNC-palvelin laitteelle, johon halutaan luoda etäyhteys. Siemensin tarjoama Smart-palvelin pystyy toimimaan VNC-palvelimenä. (26, s. 2; 27.) Etäyhteyden ottaminen operointipaneelille voi auttaa

esimerkiksi vikadiagnoosin tekemisessä etänä, jolloin huoltaja näkee etukäteen, että mikä komponentti on vaihdettava.

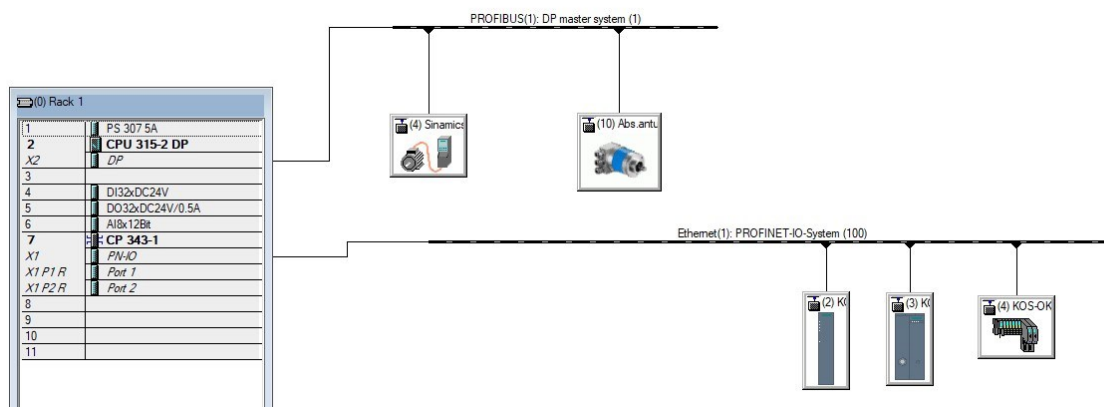
3.4 Operointipaneelin käyttöliittymän nykyaikaistaminen

Automaatiojärjestelmän modernisoinnin yhteydessä on hyvä tilaisuus parantaa operointipaneelin käyttöliittymän ulkoasua ja tuoda uusia toimintoja, jotka helpottavat ja tehostavat konttisekoittimen käyttöä. Uudelle operointipaneelille voidaan tuoda esimerkiksi enemmän diagnosointityökaluja, trendit mittauksista ja reseptien selailu. Pelkästään tämä uudistus ei riitä modernisoinnin perustelulle, mutta käyttöliittymä täytyy joka tapauksessa tehdä uudelle pohjalle, jotta se sopii uuteen operointipaneeliin. Lisäksi käyttöliittymään täytyy tehdä merkittäviä muutoksia, jotta luvuissa 3.2 ja 3.3 käsitellyt ominaisuudet voidaan ottaa käyttöön.

4 Komponenttien valinta

Oikeiden komponenttien valinta on keskeinen osa modernisointia, jotta valitut komponentit tukevat haluttuja ominaisuuksia ja ovat yhteensopivia muiden komponenttien kanssa. Komponenttien valinnassa on huomioitava Orionin varaston varaosasaatavuus, koska on helpompaa tarkastella komponentin sopivuutta konttisekoittimelle, kuin lähteä etsimään tuhansista saman mallisista komponenteista sopivinta. Lisäksi varaosien hallinta pysyy parempana, koska eri prosessilaitteilla on samoja komponentteja, joten ei tarvitse erikseen ostaa varaosaa laitteelle. (28.)

Ensimmäiseksi on selvitettävä, mitkä komponentit on yhdistetty automaatiojärjestelmään ennestään. Tämä onnistuu esimerkiksi avaamalla vanhan konttisekoittimen Simatic Manager -projektitiedoston HW-konfiguraatiosivun. Kuvassa 5 esitetään Simatic Managerin laitteiston konfiguraatiosivu (HW-konfiguraatio), josta tarkat laitetyypit voidaan tarkastella klikkaamalla niiden kuvakkeita. Lisäksi kyseisellä sivulla voidaan määrittää, mitkä fyysiset laitteet ovat liitettyinä automaatiojärjestelmään.



Kuva 5. Simatic Manager -ohjelmistossa näkyvä laitteiden konfiguraatiosivu.

Toiseksi on selvitettävä, missä vaiheessa elinkaarta ovat automaatiojärjestelmään kuuluvat komponentit. Siemensin verkkosivuilta löytyy sivun 13 kuvan 4 mukainen sivu, jossa voidaan tarkastella komponentin elinkaaren vaihetta (9). Lähtökohtaisesti aktiivista komponenttia, eli elinkaaren vaiheessa kaksi olevaa komponenttia ei tarvitse uusia (luku 3.1).

Kolmanneksi on valittava vanhan komponentin tilalle korvaava komponentti. Siemensin verkkosivuilta tarjotaan monille komponenteille korvaavia tuotteita. Uutta komponenttia valittaessa on otettava huomioon, että ovatko muut komponentit yhteensopivia keskenään, kuten mitä väyläyhteyksiä käytetään ja tukevatko ne uusia ominaisuuksia. Joissain tapauksissa tarvitaan erillisiä komponentteja, jotta automaatiojärjestelmän tiedonsiirto saadaan toimimaan. Lisäksi komponentteja valittaessa voi tarkastella myös automaatiojärjestelmään kuulumattomia komponentteja. Esimerkiksi vanhemmissa toteutuksissa taajuusmuuttajat saattavat olla kytkettynä logiikan I/O-moduuleihin monella eri johdolla, jolloin ne voidaan modernisoinnin yhteydessä liittää väyläjärjestelmään. Tämä vähentää kaapeloinnin ja I/O-kanavien määrää, sekä lisäksi mahdollistaa taajuusmuuttajasta laajemman tiedonkeruun, esimerkiksi vian selvitykseen. Luvuissa 4.1–4.4 esiintyvät komponentit on päätetty vaihdettavan, ja muut komponentit pysyvät ennallaan.

4.1 Ohjelmoitava logiikka

PLC täytyy vaihtaa, koska sen elinkaari on vaiheessa kolme, ja nykyinen PLC ei tue kaikkia ominaisuuksia, joita halutaan lisätä modernisoinnin yhteydessä (9). Siemens ei tarjoa suoraan korvaavaa tuotetta tilalle, koska PLC:n vaihtaminen eri malliin tarvitsee aina ohjelmamuutoksia. Siemens tarjoaa kuitenkin samalla tai paremmalla suorituskyvyllä S7-1500-sarjan PLC:tä tilalle. Lisäksi S7-300-sarjan PLC:n vanhan ohjelman voi muuntaa tietyllä ohjelmointityökalulla S7-1500-sarjan logiikalle sopivaksi (luku 5.2).

Orionin varastosta löytyy S7-1500-sarjan PLC, tyyppiä CPU 1515-2 PN, joten tarkastellaan kyseisen PLC:n soveltuvuutta nykyisen PLC:n (CPU 315-2 DP) tilalle. CPU 1515-2 on selvästi nopeampi kuin nykyinen PLC ja pystyy käsittelemään suurempia tietomääriä. CPU 1515-2 sisältää kaksi verkkokorttia, mikä mahdollistaa sen liittämisen kahteen eri verkkoalueeseen. Näin ollen tehdasjärjestelmät ja väylälaitteet voidaan erottaa toisistaan ja liittää eri verkkoalueille. CPU 1515-2:sta löytyy PROFINET-liitäntä, mutta PROFIBUS-liitäntä puuttuu. Tämän vuoksi PLC:n lisäksi on hankittava erillinen kommunikaatiomoduli, johon voidaan liittää PROFINET- ja PROFIBUS-kaapelit. (29; 30.) Konttisekoittimen automaatiojärjestelmässä ei tarvitse käyttää älykkäitä toimintoja tai se ei vaadi merkittävästi tehoa PLC:tä, joten todennäköisesti kaikki tai suurin osa S7-1500-sarjan PLC:stä kävisi nykyisen tilalle.

4.2 I/O-hajautus

Vanhan S7-300-sarjan PLC:n kyljessä on kiinni I/O-moduuleita, joten niiden tilalle täytyy hankkia S7-1500-sarjan PLC:hen sopivat I/O-moduulit. Alun perin oli ajatuksena hankkia nimenomaan S7-1500-sarjan PLC:lle tarkoitettuja I/O-moduuleita ja kiinnittää ne suoraan PLC:n kylkeen, mutta automaatiopäällikkö Tomi Pitkäniemen kanssa käyty keskustelu toi uusia näkökulmia I/O-moduulien hankintaan. Päätetään hankkia I/O-hajautus, joka liitetään PROFINET-väylään, eikä suoraan S7-1500-sarjan PLC:n kylkeen. Tämä mahdollistaa PLC:n ja I/O-

hajautuksen vaihtamisen tulevaisuudessa eri malliin ilman, että ne ovat riippuvaisia yhteensopivuudesta keskenään. (28.)

Päädytään valitsemaan Siemensin Simatic ET 200SP -hajautus, koska niitä on käytössä paljon Orionilla, ja konttisekoittimen muutostyössä vuonna 2016 on lisätty niitä WLAN-hajautukseen. Lisäksi niitä on helppo huoltaa, koska I/O-moduulin rikkoutuessa sen voi vaihtaa nopeasti ilman työkaluja sähköt päällä. ET 200SP -hajautuksen I/O-moduuleille täytyy myös erikseen hankkia kytkentäpohjat, johon antureiden tai toimilaitteiden johdot kytketään. ET 200SP -hajautukseen täytyy liittää myös yksi kommunikaatiomodulaari, johon liitetään PROFINET-kaapelit. Kyseisen kommunikaatiomodulaarin kylkeen voi liittää I/O-moduulien kytkentäpohjat pikaliitoksilla. (31.) Kommunikaatiomodulaariin liitettävien I/O-moduulien tyypit täytyy valita toimilaitteiden, antureiden, rajakytkimien ja painikkeiden mukaisesti. Laitteen I/O-luettelosta selviää I/O-tyypit ja niiden määrät. Esimerkiksi konttisekoittimen laseranturi on analoginen tulotieto ja niitä on vain yksi kappale, joten täytyy hankkia yksi ET 200SP AI 2x 2–4-wire eli kaksi kanavainen analoginen tulomodulaari (32). Yleisesti modernisoinneissa uudistus tehdään olemassa oleville laitteistoille, joten tulojen ja lähtöjen tyypit sekä niiden määrät löytyvät laitteen I/O-luettelosta.

4.3 Taajuusmuuttaja

Konttisekoittimen kehikkoa pyörittävän moottorin taajuusmuuttajan tehoyksikkö (Sinamics PM250-IP20-FSD-A-400V 22KW) on aktiivinen tuote, joten sitä ei ole tarvetta vaihtaa (33). Sen sijaan taajuusmuuttajan ohjausyksikkö (Sinamics G120 Control Unit 240S DP-F) ja operointipaneeli (Sinamics G110 Basic Operator Panel) eivät ole aktiivisia tuotteita, joten ne täytyy hankkia erikseen (34; 35.) Ohjausyksikköä valittaessa täytyy ottaa huomioon, että se on yhteensopiva olemassa olevan tehoyksikön kanssa, ja valitun ohjausyksikön täytyy sisältää ainakin samat ominaisuudet tai enemmän ominaisuuksia, kuin vanha ohjausyksikkö. Lisäksi täytyy varmistaa, että ohjausyksikkö on mahdollista liittää PROFIBUS-väylään ja että siinä on tarpeelliset turvatoiminnot.

Ohjausyksiköitä on kolme perusmallia, jotka sopivat nykyiseen tehoyksikköön: CU 230P-2, CU 240E-2 ja CU 250S-2. CU 230P-2 on suunniteltu erityisesti pumpuille, tuulettimille ja kompressoreille. CU 240E-2 on näistä yleiskäyttöisin ja sopii moniin eri sovelluksiin, kuten sekoittimiin ja kuljetinhihnoin. CU 250S-2 on tarkoitettu laitteisiin, joissa on paljon nopeuden vaihtelua tai säätöä, esimerkiksi nostimiin. (36, luku. 9.2.) Konttisekoittimessa ei ole paljon nopeuden vaihtelua, joten näistä ohjausyksiköistä CU 240E-2 on sopivin. Tämän jälkeen täytyy valita tarkempi tyyppi valitun ohjausyksikön perusmallista. Ohjausyksiköstä täytyy löytyä liitettävyyttä PROFIBUS-väylään ja tarvittavat turvaominaisuudet.

Konttisekoittimeen tehdyn riskiarvioinnin perusteella nykyiset turvaominaisuudet täyttävät vaatimukset, joten valitaan samat turvaominaisuudet kuin vanhassa ohjausyksikössä. Nämä ovat Safe Torque Off (STO) ja Safe Stop 1 (SS1). STO-toiminto estää hätäpysäytyksen tapahtuessa kaiken virransyötön moottorille ja estää sen uudelleenkäynnistymisen, kunnes STO-toiminto on kuitattu. STO-toiminnon voi toteuttaa myös turvakäyttöön tarkoitetuilla kontaktoreilla. SS1-toiminto puolestaan monitoroi esimerkiksi ohjausyksikölle saapuvia hätäseispainikkeiden johtoja. Yleensä hätäseispainiketta ohjataan kahdella johtimella, jotta toisen johtimen katketessa painike toimisi edelleen. SS1-toiminto vertaa kahden johtimien sähkönsyöttöä keskenään, ja tilanteissa, joissa johtimia ohjataan eri aikaan tai toisesta johtimesta putoaa sähkö, niin taajuusmuuttaja antaa virheilmoituksen ja laukaisee STO-toiminnon. Näin ollen taajuusmuuttajan ohjausyksiköksi valitaan Sinamics CU 240E-2 DP-F. (36, luku. 9.4–9.6.)

Viimeisenä täytyy valita taajuusmuuttajan operointipaneeli. Operointipaneeleita on kahta eri mallia: Älykäs operointipaneeli 2 (IOP-2) ja perusoperointipaneeli 2 (BOP-2). Lisäksi on olemassa operointipaneelin tilalle asennettava WLAN-verkkokortti, jolloin paneeliin voidaan ottaa yhteys verkkoselaimen kautta. Operointipaneelin valinnalla ei ole merkitystä taajuusmuuttajan toiminnan kannalta, mutta sen avulla voidaan esimerkiksi ohjata taajuusmuuttajaa käsiohjauksella testaus-tilanteissa tai asettaa parametreja ilman ohjelmointityökaluja. Älykkäässä operointipaneelissa on suurempi näyttö ja siinä on enemmän ominaisuuksia esimerkiksi vian diagnosointiin kuin perusoperointipaneelissa. (36, luku. 9.105–

9.112.) Varmuuden vuoksi valitaan älykäs operointipaneeli siltä varalta, että sen ominaisuuksia tarvitaan tulevaisuudessa, ja lisäksi suurempaa näyttöä on mieltyvämpi katsoa. Hintaero ei ole merkittävä älykkään operointipaneelin ja perusoperointipaneelin välillä.

4.4 HMI-paneeli

Vanha HMI-paneeli eli operointipaneeli, jolla konttisekoittimen toimintoja ohjataan, ei ole enää aktiivinen tuote, eikä se tue uusia ominaisuuksia, joten sen tilalle täytyy hankkia uusi paneeli. Siemensillä on tarjolla neljä eri paneelimalia: Basic-paneeli, Comfort-paneeli, Key-paneeli ja Mobile-paneeli. Basic-paneeli tarjoaa nimensä mukaisesti perusominaisuuksia, kuten visuaalisten painonappien tai tulo-/lähtökenttien (I/O field) lisäämisen näytölle. Comfort-paneeli on Basic-paneelistä kehittyneempi versio ja mahdollistaa samat toiminnot kuin Basic-paneeli, mutta sen lisäksi sillä voidaan esimerkiksi tulostaa raportteja, ottaa etäyhteys (Smartclient) HMI-paneeliin, käyttää ponnahtusikkunoita sekä käyttää Audit Trail -ominaisuutta. Comfort-paneelin tekninen suorituskyky on parempi, kuin Basic-paneelilla. Key-paneeli on painonappipaneeli, joten siinä ei ole näyttöä lainkaan. Laitteen käyttäjä siis ohjaa laitetta painonappien avulla ja seuraa laitetta led-valojen avulla. Mobile-paneeli on langaton paneeli, jota voidaan käyttää liikuttaessa, esimerkiksi käsikäyttöisen nostimen ohjaamiseen. Kyseisen paneelin tekninen suorituskyky vaihtelee mallin mukaan, ja se sijoittuu yleensä Basic- ja Comfort-paneelin välille. Basic- ja Comfort-paneeleista on saatavilla myös Unified-versio, joiden käyttöliittymä rakennetaan eri ohjelmistolla kuin perinteisten paneelien. (37; 38, luku 2.)

Konttisekoittimen modernisoinnin yhteydessä täytyy lisätä Audit trail -ominaisuus, jotta GMP paranee. Ainoastaan Comfort- ja Mobile-paneeleissa on tämä ominaisuus. Lisäksi Comfort-paneelissa on monia muitakin ominaisuuksia, jotka voivat olla hyödyllisiä toteuttaessa modernisointia. Vanha operointipaneeli sijaitsee kiinteästi ohjauskeskuksen ovelta, joten näiden syiden perusteella valitaan Comfort-paneeli. Seuraavaksi täytyy valita paneelin koko. Vanha konttisekoittimen operointipaneeli on 8 tuumainen eli hyvin pieni. Päätetään hankkia 12

tuumainen operointipaneeli, jolloin näytölle mahtuu enemmän tietoa, ja se on käyttökäytävällisempi, kun painonapit ja muut graafiset elementit näkyvät paremmin.

Comfort-paneeleista on saatavilla eri tyyppisiä esimerkiksi ulkokäyttöön, ja joihinkin tyyppisiin on saatavilla fyysiset näppäimet näytön lisäksi. Comfort PRO -paneelin ja perusmallin erona on se, että Comfort PRO -paneelin takaosassa on sama suojausluokitus kuin etuosassa, mikä mahdollistaa sen asentamisen ilman koteloa. Perusmalli sen sijaan tarvitsee asennukseen kotelon. Operointipaneeli tullaan asentamaan sisätiloihin upotettuna ohjauskaappiin, eikä fyysisille painonapeille ole tarvetta, joten valitaan konttisekoittimen operointipaneeliksi SIMATIC HMI TP1200 Comfort. (38, luku 2.)

5 Konttisekoittimen automaatiojärjestelmän ohjelman toteutus

Konttisekoittimen automaatiojärjestelmän ohjelma toteutetaan kokonaisuudessaan TIA Portal -ohjelmistolla. TIA Portal (Totally Integrated Automation Portal) on ohjelmisto, joka on suunniteltu pääosin Siemensin S7-1200- ja S7-1500-sarjan logiikoiden (PLC) ohjelmointiin. Käytännössä TIA Portal on Simatic Manager -ohjelmiston uudempi versio. Yksi TIA Portal -ohjelmiston merkittävimmistä eroista Simatic Manager -ohjelmistoon on se, että TIA Portal -ohjelmiston projektipuhun voidaan integroida esimerkiksi Siemensin taajuusmuuttajat ja HMI-paneeleit. Tämä mahdollistaa taajuusmuuttajien parametroidin ja HMI-paneeleiden käyttöliittymän rakentamisen suoraan TIA Portal -projektin sisällä, eikä erillisiä ohjelmistoja tarvita. (39.) Konttisekoittimen uuden S7-1500-sarjan logiikan ohjelmointi, uuden operointipaneelin käyttöliittymän rakentaminen ja uuden taajuusmuuttajan ohjausyksikön parametroidi toteutetaan TIA Portal V18 -ohjelmistolla.

5.1 S7-1500-sarjan logiikan ohjelmointi

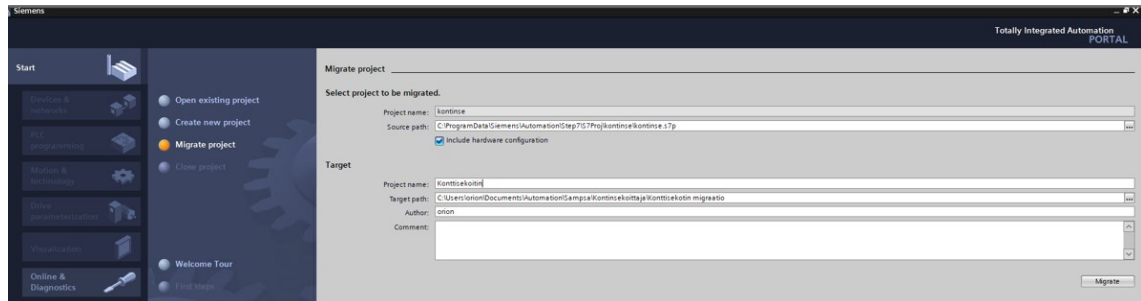
5.1.1 Vanhan ohjelman migraatio

Ohjelman migraatiolla tarkoitetaan sitä, että Simatic Manager -ohjelmistolla toteutetut S7-300- ja S7-400-sarjan logiikoiden (PLC) projektitiedostot voidaan muuttaa TIA Portal -projektitiedostoiksi. Varsinainen migraatio toteutetaan TIA Portal -ohjelmistossa. Lisäksi migroidun TIA Portal -projektitiedoston sisällä voi vaihtaa kyseisten logiikoiden mallin S7-1500-sarjan logiikaksi, jolloin TIA Portal -ohjelmisto toteuttaa tarvittavat ohjelmamuutokset. (40, s. 5.) Tämä voi olla erityisen hyödyllistä silloin, kun kohdelaitteen toiminallisuus ei muutu merkittävästi tai kohdelaitteen uudelleen ohjelmointi olisi erittäin työlästä. Haittapuolena on se, että TIA Portal -ohjelmisto ei osaa optimoida S7-300- tai S7-400-sarjan logiikoiden ohjelmarakennetta S7-1500-sarjan logiikalle täysin sopivaksi, jolloin ohjelma voi toimia jopa aiempaa hitaammin (40, s. 55). Lisäksi joitakin kyseisten logiikoiden ohjelmoinnissa käytettyjä ohjelmalohkoja TIA Portal -ohjelmisto ei osaa suoraan muuttaa S7-1500-sarjan logiikalle sopivaksi (41, s. 24–31). Näiden korjaaminen vaatii manuaalisesti tehtäviä ohjelmamuutoksia.

Ennen kuin migraatio voidaan toteuttaa, täytyy tehdä tietyt esivalmistelut. Ensinnäkin Simatic Manager -ohjelmisto on asennettava samalle tietokoneelle, johon on asennettu TIA Portal -ohjelmisto. Koneelle täytyy myös asentaa vaihtoehtoiset ohjelmat, joita on käytetty konttisekoittimen Simatic Manager -projektitiedoston toteuttamisessa. (40, s. 41–42.) Lisäksi konttisekoittimen projektitiedosto pitää vielä kääntää Simatic Manager -ohjelmistossa mahdollisten virheiden varalta (40, s. 44). Tämän jälkeen migraatio voidaan toteuttaa TIA Portal -ohjelmistossa.

Migraatio tehdään TIA Portal -ohjelmiston Start-sivun Migrate project -välilehdellä valitsemalla migroitava Simatic Manager -projektitiedosto ja kansio, johon uusi migroitu TIA Portal -projektitiedosto tallennetaan. Lisäksi voidaan valita, että halutaanko tuoda Simatic Manager -projektitiedostosta sinne määritettyjen fyysisten laitteiden asetukset eli HW-konfiguraatio (Hardware configuration).

(40, s. 44–50.) HW-konfiguraatiossa määritetään, mitä väylälaitteita tai logiikoita projektissa käytetään. Kuvassa 6 näkyy TIA Portal -ohjelmistossa sivu, jolla Simatic Manager -projektitiedosto migroidaan TIA Portal -projektitiedostoksi.



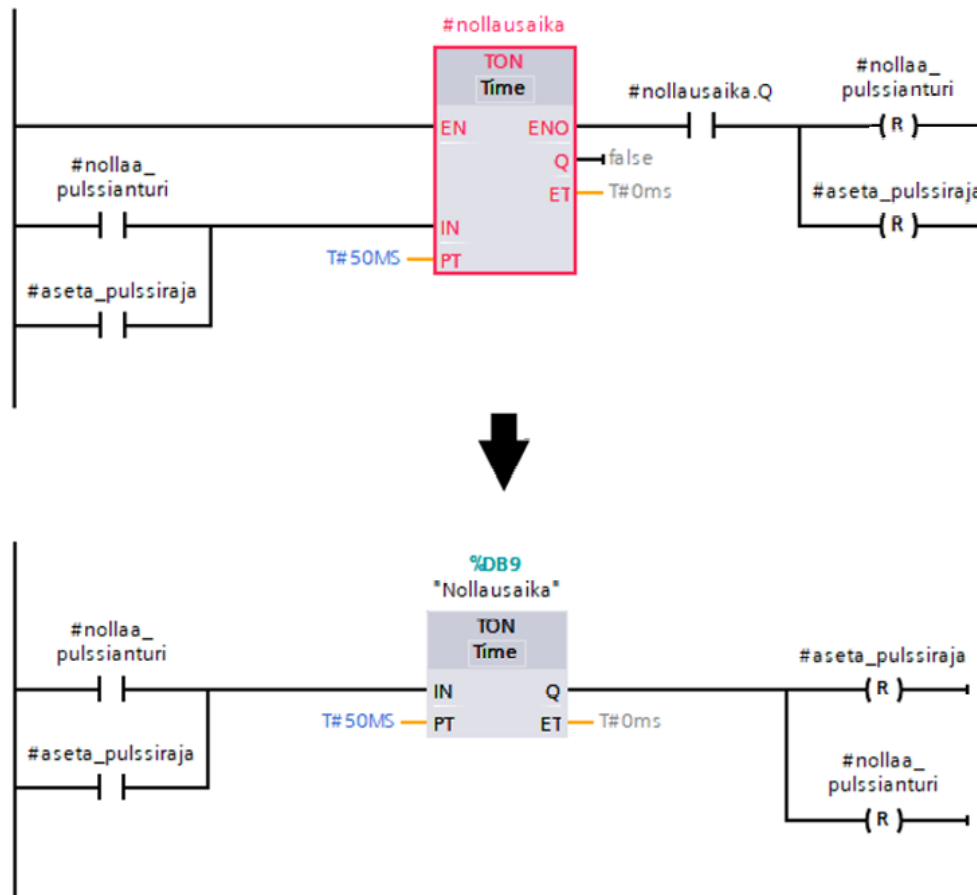
Kuva 6. Migraation toteuttamissivu TIA Portal -ohjelmistossa.

Migraation jälkeen siirrytään uuden TIA Portal -projektitiedoston projektinäkömään, jossa näkyy tapahtumaloki, joka kuvaa migraation onnistumisen. Migraatio suoritettiin ilman virheitä, mutta varoituksia ilmeni. Virheet estävät migraation toteutumisen, kun taas varoitukset ilmoittavat poikkeamasta, mutta eivät estä migraatiota toteutumista onnistuneesti. Insinööriyön kirjoittajan kokemuksen perusteella varoitukset eivät yleensä ole vakavia ja ne ovat helposti korjattavissa. Projektinäköymässä S7-300-sarjan logiikan ohjelma käännettiin ilman virheitä. Kääntämisen jälkeen S7-300-sarjan logiikka voidaan muuttaa S7-1500-sarjan logiikaksi. Tämä voidaan tehdä HW-konfiguraatiosivulla klikkaamalla S7-300-sarjan logiikkaa ja valitsemalla vaihtoehto "Migrate to S7-1500". (40, s. 33–37.) Ohjelma käytännössä luo uuden logiikan projektiin, joka ei ole yhteydessä mihinkään muuhun laitteeseen. S7-300-sarjan logiikka jäi vielä projektiin, joten se pitää itse poistaa. Insinööriyön kirjoittaja ei suosittele vanhan logiikan poistamista, ennen kuin S7-1500-sarjan logiikka on konfiguroitu vastaamaan mahdollisimman tarkkaa S7-300-sarjan logiikan asetuksia. Hyvä tapa on myös tallentaa projekti uudella nimellä, jolloin sen voi avata viiteprojektina varsinaiselle projektille.

Migraation jälkeen S7-1500-sarjan logiikan ohjelma käännetään, jotta mahdollisista varoituksista ja virheistä tulee raportti. Liitteessä 1 on raportti ohjelman

kääntämisestä, jossa näkyy ilmoitus 69:stä virheestä ja 20:stä varoituksesta. Kääntämisen yhteydessä ilmenneet virheet estävät ohjelman lataamisen logiikalle. Kaikkia S7-300-sarjan logiikan ohjelmoinnissa käytettyjä toimintalohkoja TIA Portal -ohjelmisto ei osaa suoraan muuntaa S7-1500-sarjan logiikalle sopivaksi, koska kyseisiä toimintalohkoja ei voi käyttää uudella logiikalla tai niiden käyttötapa eroaa merkittävästi (42, s. 24–31). Lisäksi joidenkin muuttujien tietotyypimuunnokset ilmoittivat virheestä, ja S7-1500-sarjan logiikan ohjelmassa ei sallita tietyissä toimintalohkoissa tyhjiä tuloja ja lähtöjä raportin mukaan (liite 1).

Seuraavaksi käydään läpi yhden virheen korjaaminen, joka näkyy liitteen 1 kuvassa 1 nimellä "RS1.ohjaus (FB100)". Kuvassa 7 ylhäällä on S7-300-sarjan logiikan ohjelmoinnissa käytetty TON-lohko (ON Delay Timer). TON-lohkolla voidaan esimerkiksi ohjata venttiiliä viiveellä, eli se toimii ajastettuna porttina, joka avautuu, kun tietty aika on kulunut. S7-300-sarjan logiikassa käytetyllä TON-lohkolla on kaksi tuloa eri käyttötarkoitukseen. Toisella tulolla (EN) TON-lohko otetaan käyttöön ja toisella tulolla (IN) käynnistetään ajastin. S7-1500-sarjan logiikan ohjelmoinnissa käytetyllä TON-lohkolla on vain tulo ajastimen käynnistämiseksi, joten lohko on aina käytössä. Kuvassa 7 alapuolella näkyy S7-1500-sarjan logiikan ohjelmoinnissa käytetty TON-lohko ja korjattu koodi. Ohjelma-
muutoksia tehdessä täytyy olla tarkkana, ettei ohjelman toiminnallisuus muutu millään tavalla.



Kuva 7. LAD-ohjelmointikielellä tehty migraation jälkeisen virheen korjaus. Ylemmän koodin toimintalohkon punainen ääriiviiva ilmoittaa virheestä.

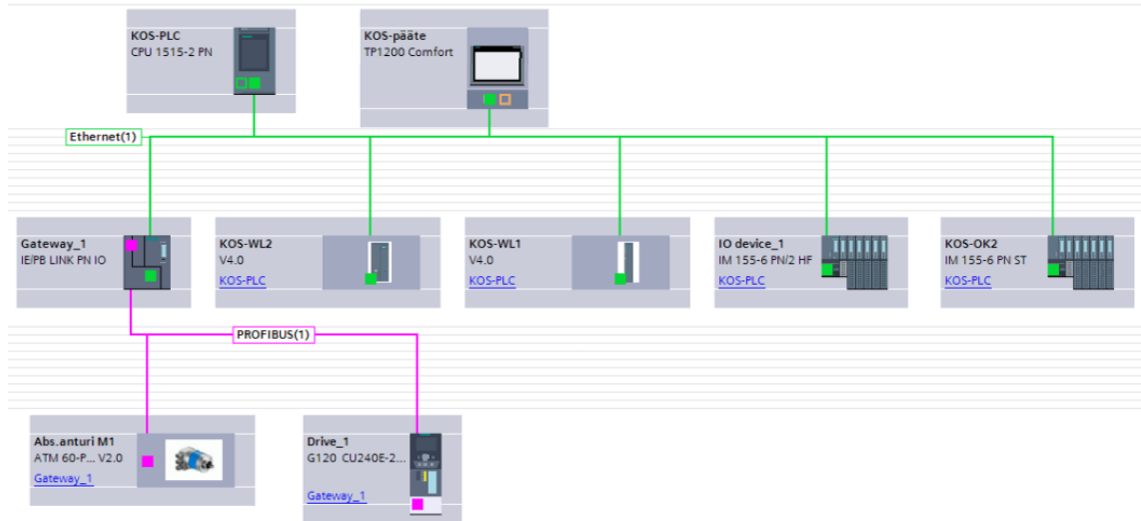
Ohjelmaan ei tehdä konttisekoittimen mekaaniseen toimintaan vaikuttavia muutoksia, mutta jokainen pienikin ohjelmamuutos vaatii ohjelman testaamisen. Vaikka ohjelma ei antaisi virheitä tai varoituksia kääntämisen jälkeen, korjaustapa voi silti muuttaa ohjelman toiminnallisuutta ja toimia epätoivotulla tavalla. Ensimmäiseksi täytyy kuitenkin korjata kaikki virheet, jotta ohjelma voidaan ladata virtuaaliselle logiikalle. Ohjelma voidaan testata käyttämällä TIA Portal -ohjelmiston simulointityökalua, jossa ohjelma ladataan virtuaaliselle logiikalle, ja sen tulosta voidaan manipuloida virtuaalisesti (42, s. 221). Konttisekoittimisessa on useita ominaisuuksia, joita ohjataan käyttöliittymän avulla, joten perusteellinen testaaminen kannattaa suorittaa käyttöliittymän valmistumisen jälkeen. Luvussa 5.5 käydään läpi konttisekoittimen toimintojen simuloinnin toteutus.

5.1.2 HW-konfiguraatio ja verkotus

HW-konfiguraatiolla ja verkotuksella tarkoitetaan automaatiojärjestelmässä käytettävien fyysisten komponenttien määrittämistä TIA Portal -projektiin ja niiden verkottamista toimimaan yhdessä esimerkiksi PROFINET-väylällä. Lisäksi komponenttien tarkemmat asetukset määritellään HW-konfiguraatiosivulla. (43, s. 40.) Migraation yhteydessä HW-konfiguraatio tuotiin uuteen TIA Portal -projektiin automaattisesti, mutta vanhoilla laitteilla. Tämän vuoksi luvussa 4 valitut laitteet on määritettävä projektiin erikseen.

Ensimmäiseksi poimitaan oikeat laitteen TIA Portal -ohjelmiston laitetietokannasta HW-konfiguraatiosivulle. Suurin osa Siemensin automaatiojärjestelmään liitettävistä komponenteista löytyy valmiiksi ohjelman tietokannasta, joten erillisiä tiedostoja ei tarvitse ladata. Tämän jälkeen PROFINET-laitteet liitetään ali-verkkoon ja niille määritetään yksilölliset IP-osoitteet. PROFIBUS-laitteet määritetään omaan DB-verkkoon, ja niille määritetään yksilölliset DB-osoitteet. (43, s. 9–105.) Osoitteiden määrittämisen jälkeen voidaan määrittellä tarkemmat parametrit ja I/O-osoitteet laitteille. Konttisekoittimen TIA Portal -projektissa laiteparametrien ja osoitteiden asettelu on suoraviivaista I/O-moduuleille ja logiikalle, koska vanhoista I/O-moduuleista ja logiikasta voi ottaa mallia.

Konttisekoittimen vanhan paneelin käyttöliittymän WinCC Flexible -projekti on aikaisemmin migroitu TIA Portal -projektiksi, joten sen voi avata viiteprojektina ja liittää konttisekoittimen TIA Portal -projektiin. Vanhan taajuusmuuttajan ohjausyksikön parametreja ei voi suoraan siirtää uudelle ohjausyksikölle, joten näiden asettamisessa täytyy olla tarkkana. Vanhan taajuusmuuttajan parametrit täytyy ensiksi ottaa talteen vanhasta ohjelmasta, ja tämän jälkeen uusi taajuusmuuttaja voidaan parametroida manuaalisesti vastaamaan vanhaa taajuusmuuttajaa TIA Portal -ohjelmistossa. Kuvassa 8 näkyy TIA Portal -projektin HW-konfiguraatiosivu, jossa kaikki laitteet ovat verkotettu ja parametroitu.



Kuva 8. Konttisekoittimen TIA Portal -projektissa määritetty väyläverkotus.

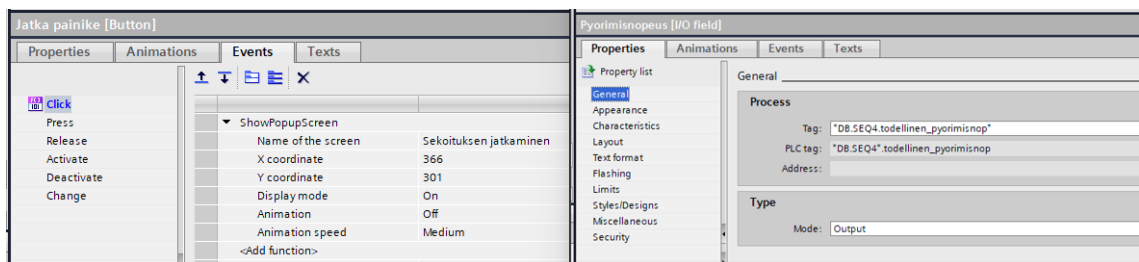
5.2 HMI TP1200 Comfort -paneelin käyttöliittymän toteutus

Uuden HMI-paneelin käyttöliittymä toteutetaan myös TIA Portal -ohjelmistolla. TIA Portal -ohjelmistoon on integroitu WinCC-pohjaisen käyttöliittymän rakentamiseen tarkoitettuja työkaluja. Nämä työkalut saadaan käyttöön lisäämällä HW-konfiguraatiossa jokin HMI-paneeli tai PC-järjestelmä. Lisäksi näiden työkalujen käyttö vaatii erillisen lisenssin. Käyttöliittymän rakentaminen saattaa olla erittäin työlästä, joten usein käytetään muita TIA Portal -projekteja viitteenä, josta pystyy kopioida valmiita näyttöelementtejä (Screen Object) tai jopa kokonaisia sivuja, joita voidaan muokata omaan projektiin sopivaksi (44).

Uuden käyttöliittymän rakentamiseen käytetään pohjana aikaisemmin migroitua TIA Portal -projektiä, joka liitetään varsinaiseen konttisekoittimen TIA Portal -projektiin. Vanhan käyttöliittymän TIA Portal -projektiin ei ole tehty muutoksia migraation jälkeen, joten se näyttää samalta kuin tällä hetkellä käytössä oleva käyttöliittymä. Käyttöliittymään tulee paljon muutoksia, joten on helpompaa aloittaa tyhjältä sivulta ja siirtää tarvittaessa vanhan käyttöliittymän näyttöelementteihin määritettyjä muuttujia uusiin näyttöelementteihin. Näyttöelementeillä (Screen Object) tarkoitetaan esimerkiksi näytössä olevia painonappeja tai tulo-/lähtökenttiä (I/O-field) (45, s. 2054). Näyttöelementtiin määritetyllä muuttujalla

tarkoitetaan, esimerkiksi kyseisen muuttujan arvon näyttämistä käyttöliittymässä tulo-/lähtökentän (I/O-field) avulla.

Käyttöliittymän perustoimintojen toteuttaminen on yksinkertaista eikä vaadi ohjelmointitaitoja. Näyttöelementti lisätään halutulle sivulle vetämällä se työkalupalkista, ja sen ulkomuotoa tai asetuksia voidaan muokata klikkaamalla näyttöelementtiä ja valitsemalla "Properties". Kuvassa 9 oikealla on tulo-/lähtökentän (I/O-field) asetukset-sivu, jossa on määritetty muuttuja, joka näyttää käyttöliittymässä kehikon pyörimisnopeuden. Lisäksi tulo-/lähtökentälle voi määrittää myös toiminnon, jolloin operointipaneelilta käsin voidaan syöttää arvo ja ohjelma siirtää arvon näyttöelementtiin määritettyyn muuttujaan. Kuvassa 9 vasemmalla näkyy painonapin tapahtumasetukset, jossa määritetään, mitä tapahtuu, kun painiketta klikataan. Painiketta klikkaamalla avautuu ponnahdusikkuna, jossa kysytään varmistusta sekoituksen jatkamiselle.



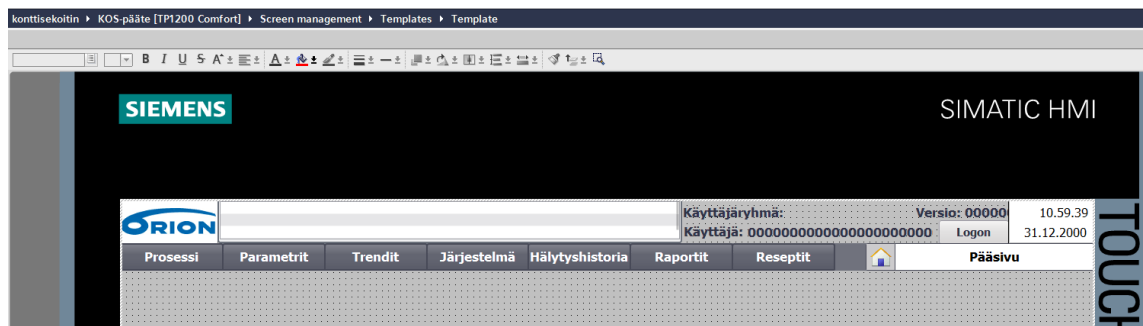
Kuva 9. Painonapin ja tulo-/lähtökentän asetukset TIA Portal -ohjelmistossa.

Migroidussa käyttöliittymässä hälytykset, HMI-muuttujat ovat säilyneet, ja niihin ei tule merkittäviä muutoksia, joten niitä ei tarvitse määrittää uudelleen. Lisäksi raportointitoiminnot migroitui uudelle käyttöliittymälle. Tämän takia tässä luvussa keskitytään enemmän sivujen rakentamiseen. Ainoastaan Audit trail -asetukset täytyy määrittää vanhoille ja uusille HMI-muuttujille, joiden muuttamisesta halutaan jättää jälki Audit trail -tiedostoon. Näitä ovat HMI-muuttujat, joissa säilytetään esimerkiksi laiteparametreja.

5.2.1 Pohjasivu

Käyttöliittymän pohjasivulla tarkoitetaan sivua, joka toimii jokaisen muun sivun pohjana. Tämä mahdollistaa olennaisten tietojen näyttämisen sekä sivujen välisen navigoinnin kaikilla sivuilla. Kuvassa 2 ja 3 näkyy vanhan käyttöliittymän pää- ja prosessisivu, joiden yläreunassa näkyy pohjasivu, joka näkyy kaikilla sivuilla. Kyseisellä pohjasivulla on vain ulos-/sisäänkirjautumispainike, päivämäärä ja aika sekä sivujen välinen navigointi.

Uudistetun konttisekoittimen käyttäjävaatimuksien (46) mukaan pohjasivulla täytyy näkyä kirjautuneen käyttäjän tiedot, kirjautumispainike, päivämäärä ja aika, aktiiviset hälytykset ja ohjelmaversio. Käyttäjävaatimukset voivat olla eri laitteilla hyvin samankaltaisia, joten pohjasivu kopioitiin toisesta Orionin prosessilaitteen TIA Portal -projektista (47) ja muokattiin vastaamaan uuden konttisekoittimen käyttöliittymän tarpeita. Kuvassa 10 esitetään valmis pohjasivu käyttöliittymäeditorissa, josta löytyvät kaikki käyttäjävaatimuksissa (46) vaaditut ominaisuudet, sekä lisättiin Orionin logo ja järkevä sivujen välinen navigointi.



Kuva 10. Valmis pohjasivu toteutettuna TIA Portal -ohjelmiston käyttöliittymäeditorilla.

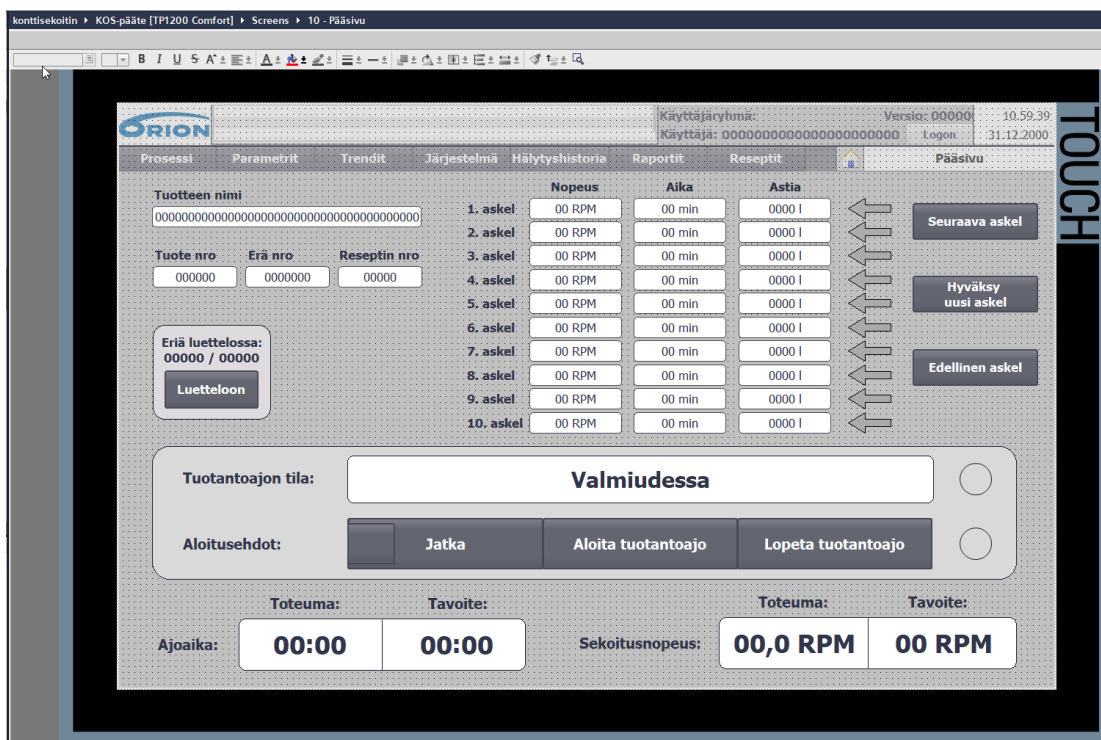
5.2.2 Pääsivu

Pääsivulla tapahtuu konttisekoittimen tuotantoajon hallinta ja sen seuraaminen. Vanhan käyttöliittymän pääsivulla on tietoa tai ominaisuuksia, joita operaattorin ei tarvitse nähdä tai käyttää, kuten osasekvenssien käyttö ja niiden aloitusehtojen tarkastelu (kuva 2). Lisäksi sekoituksen edistymisen seuraaminen on

hankalaa, koska oleelliset tiedot, kuten sekoitusnopeus ja kulunut aika, ovat hyvin pienellä fontilla vanhassa käyttöliittymässä.

Uuden käyttöliittymän pääsivu tehdään kokonaan uudelle pohjalle, koska uusi pääsivu tulee eroamaan huomattavasti vanhasta. Vanhan käyttöliittymän pääsivu jätetään kuitenkin talteen toistaiseksi, jotta sen näyttöelementteihin määritettyjä muuttujia voidaan käyttää hyödyksi uudella pääsivulla. Käyttäjävaatimusten (46) mukaisesti pääsivulle täytyy lisätä tietoja käytetystä reseptistä, tuotetiedot, eränumero, ajoparametrit, toteuma ajotiedoista, tuotannon käynnistys ehdot ja tilatieto sekoituksen tilasta, kuten esimerkiksi onko konttisekoitin keskeytyneenä tai lukitseeko se konttia.

Kuvassa 12 on valmis pääsivu käyttöliittymäeditorissa, josta löytyvät edellisessä kappaleessa mainitut vaatimukset. Lisäksi kuvasta löytyvät painike, jota painamalla pääsee katselmoimaan eräluetteloa sekä painikkeet sekoitusaskeleen ohittamiseen. Näyttöelementtejä voidaan myös määrittää näkymättömäksi tiettyjen ehtojen perusteella (45, s. 222–223). Pääsivulla on paljon näyttöelementtejä, joita ei tarvitse näyttää kuin tietyissä tilanteissa, esimerkiksi silloin, kun tuotantoajo on käynnissä, joten tuotantoajon aloituspainikkeen näyttäminen ei ole enää tarpeen. Sama koskee tuotteen sekoitusaskeleiden tietoja ja sekoitusaskeleiden ohituspainikkeita. Esimerkiksi sekoitusaskeleiden ohituspainikkeet näkyvät vain työnjohdolle ja pääkäyttäjälle. Operaattorilla ei ole oikeuksia ohittaa askelta, joten painikkeiden ei tarvitse näkyä operaattorille. Tämän lisäksi tuotteiden sekoitusaskeleiden määrä vaihtelee, joten on järkevämpää näyttää käyttöliittymässä vain sen verran reseptiparametreja, kuin tuotteella on sekoitusaskeleita. Käytännössä, jos tuotteella on neljä sekoitusaskelta, niin operointipaneelilla näkyy neljä riviä tuotteen sekoitusaskeleen reseptiparametreja. Kuvassa 12 näkyvistä nuolista vain yksi näkyy kerralla operointipaneelilla ja se osoittaa sitä sekoitusaskelta, jonka ajoparametreja käytetään, kun aloitetaan kontin sekoittaminen.



Kuva 11. Valmis pääsivu toteutettuna TIA Portal -ohjelmiston käyttöliittymäeditorilla.

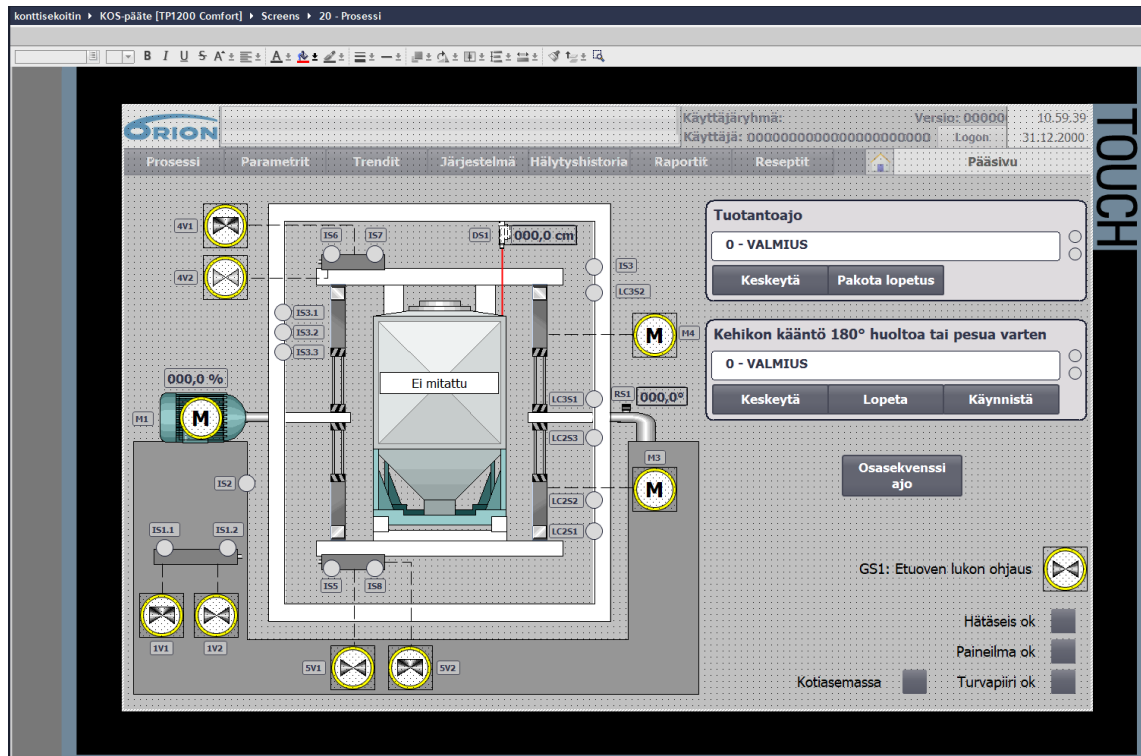
Tilan säästämiseksi tuotannon aloitusehdoissa käytetään väri-indikointia. Käytännössä näyttöelementin ollessa vihreä, tuotantoajo voidaan aloittaa, ja sen ollessa punainen, tuotantoajoa ei voida aloittaa. Halutessaan käyttäjä voi klikata kyseistä painiketta, jolloin avautuu ponnahdusikkuna. Ponnahdusikkunassa voi tarkastella tarkemmin aloitusehdoja samalla tavalla, kuin vanhan käyttöliittymän pääsivulla (kuva 2). Ponnahdusikkunoita käytetään myös silloin, kun halutaan varmistaa jonkin toiminnon aloittaminen, esimerkiksi varsinaisen sekoittamisen aloitus. TIA Portal -ohjelmiston käyttöliittymäeditori mahdollistaa ponnahdusikkunoiden luomisen, joiden avautuminen määritetään esimerkiksi painiketta painaessa, kuten kuvassa 9 on esitelty (45, s. 63–66). Tuotantoajon aloittamisen jälkeen siirrytään viivakoodinluku-sivulle, jossa resepti haetaan viivakoodinlukijalla luetun tuotenumeron tai eränumeron perusteella. Reseptin löytyessä siirrytään takaisin pääsivulle, jonka jälkeen varsinaisen sekoittamisen voi aloittaa, kunhan aloitusehdot ovat täyttyneet. Reseptien hakemisen toteutusta käsitellään tarkemmin luvussa 5.3.

5.2.3 Prosessisivu

Prosessisivulla voidaan nähdä konttisekoittimen jokaisen anturin ja toimilaitteen tilatieto. Lisäksi sivulla voidaan ohjata manuaalisesti jokaista toimilaitetta erikseen. Sivua käytetään pääasiassa huoltojen yhteydessä ja prosessin seuraamiseen. Kuvassa 3 on prosessisivu vanhassa käyttöliittymässä.

Uuden käyttöliittymän prosessisivun toiminnot ja nähtävät tiedot eivät tule muuttumaan merkittävästi vanhaan verrattuna, koska uuden konttisekoittimen käyttäjävaatimuksien (45) mukaan nykyisestä toteutuksesta löytyvät kaikki tarvittavat ominaisuudet. Kuitenkin sivun ulkoasu päivitetään modernimmaksi, jotta se sopii yhteen muiden käyttöliittymän sivujen kanssa. Lisäksi toiminto, jolla konttisekoittimen kehikko käännetään 180° huollon tai pesun ajaksi, siirretään prosessisivulle. Kyseinen toiminto käynnistettiin vanhassa käyttöliittymässä erilliseltä sivulta, joten järkevämpää olisi, että kyseistä toimintoa käytettäisiin prosessisivulla, jotta huollon yhteydessä voidaan nähdä laitteiden tilatiedot (1).

Prosessisivun toteuttaminen on yksinkertaista, koska näyttöelementtejä voidaan kopioida suoraan ilman isoja muokkauksia uudelle prosessisivulle sivulta, johon on liitetty vanhan käyttöliittymän prosessisivu. Näitä ovat esimerkiksi näyttöelementit, jotka luovat yhdessä konttisekoittimesta 2D-mallinnuksen, kuten kuvasta 12 voidaan nähdä. Muiden näyttöelementtien muuttajat ja asetukset siirretään uusiin näyttöelementteihin, jotka toteuttavat saman toiminnon, mutta näyttävät erilaiselta. Kuvien 12 ja 3 käyttöliittymien pääsivuja voidaan verrata keskenään, jolloin voidaan nähdä selvä visuaalinen ero niiden välillä. Lisäksi kehikon kääntöä varten lisätään täysin uudet näyttöelementit, kuten painikkeet kehikon pyörittämisen hallitsemiseen sekä tilatieto siitä, mitä parhaillaan tapahtuu. Kuvassa 12 on valmis prosessisivu toteutettuna TIA Portal -ohjelmiston käyttöliittymäeditorilla. Kuva on otettu käyttöliittymäeditorissa, joten todellisuudessa se näyttää erilaiselta varsinaisella operointipaneelilla.



Kuva 12. Uuden käyttöliittymän valmis prosessisivu TIA Portal -ohjelmiston käyttöliittymäeditorissa.

Käyttöliittymän prosessisivulle lisättiin myös näyttöelementit, joiden avulla voidaan seurata tuotantoajoa ja tarvittaessa keskeyttää tai pakottaa tuotantoajon lopetus. Tuotantoajon pakotetulla lopettamisella tarkoitetaan toimintoa, jolla voidaan pakottaa sekvenssien lopettaminen, jolloin konttisekoitin ei suorita enää mitään toimintoja ja täytyy paikoittaa manuaalisesti kotiasemaan. Normaali tuotantoajon lopettamisen painike käynnistää kehikon paikoitustoiminnon, jolloin konttisekoitin ajaa itsensä kotiasemaan. Sekvenssin pakotettu lopetus on tarpeen esimerkiksi toimilaitteen vikatilanteissa, jonka takia paikoittamista ei voida suorittaa. Tuotantoajoa ei ole tarkoitettu käynnistettäväksi prosessisivulta, joten sille ei ole laitettu painiketta kyseiselle sivulle.

Vanhassa käyttöliittymässä osasekvenssien hallinta toteutettiin pääsivulla, mutta uuden käyttöliittymän pääsivulla ei ole niille tilaa, eikä niitä ole tarpeen hallita siellä. Tämän vuoksi osasekvenssien hallitsemiseen tehdään oma sivu,

johon pääsee prosessisivulta painamalla painiketta, joka näkyy kuvassa 12. Käytännössä osasekvenssien hallintasivu toteutetaan kopioimalla valmis prosessisivu uudelle sivulle, mutta kopioidulla sivulla monistetaan kehikon käännön hallintalaatikko osasekvenssien määrän verran, ja tuotantoajon hallintalaatikko poistetaan. Näin osasekvenssien hallintaa tarkoitetut muuttujat voidaan siirtää monistettuihin laatikoihin. Tämän ratkaisun myötä osasekvenssejä voidaan hallita erilliseltä sivulta, ja lisäksi voidaan nähdä kaikkien toimilaitteiden sekä antureiden tilatiedot samaan aikaan, mikä voi auttaa testaamisessa ja vian selvityksessä.

5.2.4 Muut sivut

Käyttöliittymän rakentaminen on hyvin työlästä, ja samoja toteutustapoja toistetaan eri sivuilla käyttöliittymää rakennettaessa. Tämän vuoksi muiden sivujen toteutusta ei käsitellä tarkemmin. Muut sivut kuitenkin pääosin rakennetaan samoilla periaatteilla kuin luvuissa 5.2.1–5.2.3 esiintyvät sivut. Alla on luettelo muista uuden käyttöliittymän sivuista, jotka toteutetaan ja lyhyt selitys niiden käyttötarkoituksesta.

- **Reseptit-sivulla** voidaan selailta, tallentaa tai muokata tuotteiden reseptejä. TIA Portal -ohjelmistossa on käytettävissä kehittyneempi näyttöelementti, joka on tarkoitettu juuri näiden toimintojen suorittamiseen (45, s. 1128–1129).
- **Raportit-sivulla** voidaan selata PDF-raportteja ajetuista eristä, jotka on tallennettu verkkolevylle. TIA Portal -ohjelmistolla on tarjota kehittyneempi näyttöelementti, jolla pääsee käsiksi PDF-tiedostoihin (45, s. 447).
- **Hälytyshistoriasivulla** voidaan selata menneitä ja aktiivisia hälytyksiä. TIA Portal -ohjelmistossa on käytettävissä kehittyneempi näyttöelementti, joka on tarkoitettu näiden toimintojen suorittamiseen (45, s. 432). Lisäksi

näyttöelementtiin kuuluu ominaisuus, joka tallentaa paikallisesti hälytykset luettavaan CSV-tiedostoon (45, s. 902).

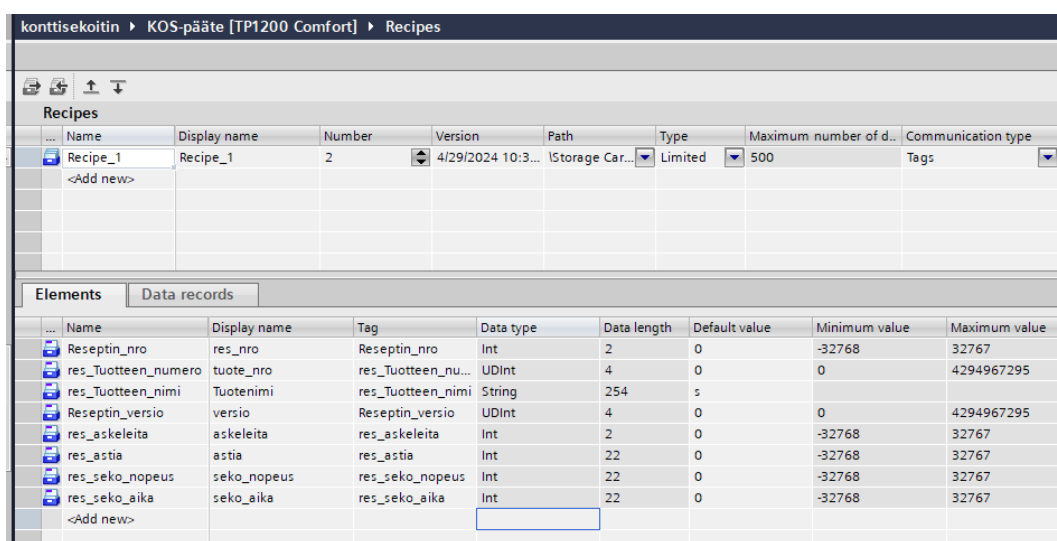
- **Järjestelmäsiivu** sivu koostuu useasta välilehdestä, joilla voidaan tehdä järjestelmän hallintaa liittyviä toimia. Kyseisiltä välilehdiltä löytyy esimerkiksi toiminnot automaatiojärjestelmän sisäiseen diagnostiikkaan, käyttöliittymän alas ajamiseen, varmuuskopioiden hallintaa ja näytön kalibrointiin.
- **Trendit-sivulla** voidaan seurata moottorin pyörimisnopeutta ja sen asetusarvoa trendikäyrän muodossa määritetyiltä aikaväliltä. TIA Portal -ohjelmistossa on valmiina kehittyneempi näyttöelementti, joka piirtää käyrää muuttujan muutoksien mukaisesti (45, s. 415).
- **Parametrit-sivulla** sivulla voidaan asettaa parametreja liittyen konttisekoittimen mekaaniseen toimintaan, esimerkiksi millä pyörimisnopeudella moottori paikoittaa kehikon kotiasemaan tai millä raja-arvoilla laseranturi tunnistaa ison kontin. Konttisekoittimen mekaaninen toiminta ei muutu modernisoinnin yhteydessä, joten parametrit pidetään ennallaan uudessa toteutuksessa.

5.3 Reseptien hallintajärjestelmän toteutus

Konttisekoittimen suurin ohjelmallinen muutos on reseptien hallintajärjestelmä. Modernisoinnin jälkeen operaattorin tarvitsee vain lukea viivakoodit kontista, ja järjestelmä hakee automaattisesti oikean sekoitusreseptin tuotteelle. Tuotteen reseptissä on kolme sekoitusparametria: sekoitusaika, kontin pyöritysnopeus ja kontin koko. Reseptien hallintajärjestelmän toteuttamisesta tekee haastavampaa se, että tuotteella saattaa olla enintään kymmenen sekoitusaskelta, eli samaa tuotetta saatetaan sekoittaa kymmenellä eri kerralla. Tuotteen sekoittamisen jälkeen tuote siirtyy seuraavaan prosessivaiheeseen, jonka jälkeen se saattaa tulla sekoitettavaksi uudelleen, mutta sekoitusparametrit voivat olla seuraavassa sekoituksessa täysin eri arvoilla. Esimerkiksi tuotteen reseptillä voi olla

10 eri arvoa sekoitusnopeudelle, joista yhtä käytetään varsinaisessa sekoituksessa. Lisäksi sekoitusaskeleiden välissä sekoitetaan muitakin tuotteita, joten järjestelmän täytyy sekoituksen jälkeen seurata jokaista erää, joilla ei ole viimeistä sekoituaskelta suoritettu. Järjestelmään täytyy myös pystyä lisäämään uusia reseptejä suoraan operointipaneelilta. (46.)

TIA Portal -ohjelmistossa on jonkin verran työkaluja reseptien hallintajärjestelmän toteuttamiseen, mutta se vaatii silti paljon insinööriä (45, s. 1106). Ensimmäiseksi määritetään muuttujat, joihin ladataan väliaikaisesti haettu resepti. Tämän jälkeen voidaan määrittää reseptin rakenne reseptien määrittämissivun elementit-välilehdellä (Elements) kuvan 13 mukaisesti, eli käytännössä sekoitusparametrit ja tunnisteet, joilla tuote tunnistetaan. Aikaisemmin määritetyt muuttujat yhdistetään reseptielementteihin, jotta järjestelmän sisäänrakennettu reseptien hallintajärjestelmä voi ladata reseptin tiedot niihin. (23, s. 15–28.)



Name	Display name	Tag	Data type	Data length	Default value	Minimum value	Maximum value
Reseptin_nro	res_nro	Reseptin_nro	Int	2	0	-32768	32767
res_Tuotteen_numero	tuote_nro	res_Tuotteen_nu...	UDInt	4	0	0	4294967295
res_Tuotteen_nimi	Tuotenimi	res_Tuotteen_nimi	String	254	s		
Reseptin_versio	versio	Reseptin_versio	UDInt	4	0	0	4294967295
res_askeleita	askeleita	res_askeleita	Int	2	0	-32768	32767
res_astia	astia	res_astia	Int	22	0	-32768	32767
res_seko_nopeus	seko_nopeus	res_seko_nopeus	Int	22	0	-32768	32767
res_seko_aika	seko_aika	res_seko_aika	Int	22	0	-32768	32767

Kuva 13. Reseptirakenteen määrittäminen TIA Portal -ohjelmistolla.

Reseptisivun Data records -välilehdellä voidaan määrittää tuotekohtaiset sekoitusparametrit ja tunnisteet eli varsinaiset reseptit (23, s. 28–30). Tässä projektissa ne kuitenkin ladataan suoraan paneelille CSV-tiedostosta ImportDataRecord-toiminnon avulla (45, s. 1634–1635). Testausta varten TIA Portal -projektiin voidaan määrittää reseptejä kyseiselle välilehdelle.

TIA Portal -ohjelmistossa on valmiita toimintoja, joilla voidaan lukea resepti sen järjestysnumeron perusteella ja siirtää arvot reseptielementtien määrätyille muuttujille. Haasteena on se, että järjestelmän täytyy osata hakea automaattisesti oikea resepti pelkän tuotenumeron perusteella. Tuotenumero on määritetty elementiksi ja sen perusteella ei voida suoraan lukea reseptiä. TIA Portal -ohjelmistossa on mahdollista käyttää tapahtumapohjaista VB Script -ohjelmointikieltä suorittamaan toimintoja WinCC-käyttöliittymässä, joita ei ole mahdollista tehdä perustoiminnoilla (48).

VB Script -ohjelmointikielellä voidaan luoda VB-funktioita, joita kutsutaan toisessa VB-funktiossa tai tapahtumalla, esimerkiksi painonapilla. VB-funktioon kirjoitettu koodi ajetaan kutsuttaessa kerran, joten se ei tarjoa syklistä tiedon siirtoa, kuten PLC. VB Script -ohjelmointikieli tukee hyvin for-silmukoita, joten esimerkiksi reseptin lukeminen voidaan toistaa useita kertoja yhdellä tapahtumalla. (48.) Normaalisti resepti voidaan lukea vain kerran, jos se määritetään kuvan 9 vasemman puolen mukaisesti.

Koodiesimerkissä 1 on luotu VB-funktio, joka mahdollistaa usean reseptin lukemisen yhdellä tapahtumalla. Koodissa LoadDataRecord-toiminto lataa sille määritetyn reseptin arvot reseptielementtien muuttujiin sen järjestysnumeron perusteella. Tämän jälkeen viivakoodilla syötettyä tuotenumeroa verrataan juuri luetun reseptin tuotenumeroon. Jos viivakoodin tuotenumero täsmää reseptin tuotenumeroon, niin hypätään pois silmukasta, muuten siirrytään seuraavan reseptin lukemiseen. Load Data Record -toiminto toimii epäsynkronoidusti muun koodin kanssa, eli käytännössä yksittäisessä for-silmukassa ohjelma ei odota toiminnon päättymistä vaan yrittää lukea seuraavan reseptin ennen kuin edellinen on luettu. Insinööriyön kirjoittaja havaitsi tämän ongelman valvomon järjestelmähälytyksissä, joissa ilmoitettiin siitä, että ohjelma ei voi lukea reseptiä, koska reseptinlukutoiminto on jo käynnissä. Tämän vuoksi for-silmukan sisälle lisättiin toinen silmukka, jossa pysytään niin kauan, kunnes LoadDataRecord-toiminto on valmis. (48.)

```

For ReseptiNro = 1 To SmartTags("MaksimiMaaraResepteja")
    SmartTags("ResLukemisenStatus") = 2
    j=SmartTags("ResLukemisenStatus ")
    touhuuaa=0

    Do While j = 2
        If touhuuaa=0 Then
            LoadDataRecord 2, ReseptiNro,"ResLukemisenStatus"
            touhuuaa = 1
        End If

        j=SmartTags("ResLukemisenStatus")
    Loop

    If j = 4 Then

        If SmartTags("res_Tuotteen_numero") =
            SmartTags("EtsiNumero") Then
            osuma = ReseptiNro
            SmartTags("EtsittyReseptinNumero") = osuma

            SmartTags("Etsitty_Tuotteen_nimi") =
            SmartTags("res_Tuotteen_nimi")
            Exit For
        End If
    End If
Next

```

Esimerkkikoodi 1. VBScript -ohjelmointikielellä tehty VB-funktio, jolla haetaan tuotenumeron perusteella oikea resepti tuotteelle.

Liitteen 2 mukaisessa VB-funktiossa kutsutaan koodiesimerkin 1 VB-funktiota nimeltä "VB_EtsiResepti". Liitteessä 2 on toteutettu viivakoodin lukemiseen tarvittavat toiminnot sekä reseptin ja erän löytämisen tilatietojen näyttäminen ope-
rintipaneelilla. Kyseistä VB-funktiota kutsutaan viivakoodinlukusivulla useita kertoja, esimerkiksi sivun auetessa ensimmäisen kerran ja seuraavan kerran, kun eränumero syötetään. Liitteessä 2 on myös muita yksityiskohtaisempia toimintoja, joita ei käsitellä tässä insinööriyössä. (48.) Liitteen 2 koodin ja esi-
merkkikoodin 1 kirjoittamisessa on käytetty apuna Orionin toista TIA Portal -projektiä (48), jossa on toteutettuna reseptien hallintajärjestelmä.

Konttisekoittimella samaa erää voidaan sekoittaa useita kertoja eri sekoituspa-
rametreilla, joten tätä varten täytyy rakentaa eräluettelo, joka tallentaa erän tie-
dot, kun se ensimmäistä kertaa saapuu laitteelle. Sekoitusaskeleet suoritetaan
aina järjestyksessä, joten jokaisen onnistuneen sekoituksen jälkeen, järjestel-
män täytyy valita seuraavan sekoitusaskeleen mukaiset parametrit. Luettelossa

täytyy siis säilyttää eränumeron lisäksi tiedot erän seuraavasta sekoitusaskeleesta, erän kokonaissekoitusaskeleiden määrästä, reseptinumerosta ja erän aloituksen päiväyksestä. Reseptinumerona käytetään reseptin järjestysnumeroa, joten sen perusteella resepti voidaan suoraan lukea. Tämän takia jokaisen reseptin läpikäynti ja tuotenumeron etsiminen ei ole tarpeen, kuin tilanteissa, joissa erä saapuu ensimmäistä kertaa laitteelle. Lisäksi ohjelman täytyy poistaa erä luettelosta automaattisesti, kun kaikki tuotteen sekoitusaskeleet on suoritettu.

Luettelon tietojen säilyttäminen esimerkiksi sähkökatkoksen aikaan ei ole mahdollista HMI-muuttujalla tehdyssä luettelossa. Lisäksi HMI-muuttujaa ei voida määrittää luetteloksi, joka voisi säilyttää useita eri tietotyyppisiä samassa alkiossa. Alkiolla tarkoitetaan luettelossa olevaa paikkaa, jossa tiedot sijaitsevat, esimerkiksi Excel-ohjelmiston luettelossa yksittäisen erän tiedot voisivat sijaita rivillä neljä. Tietotyyppillä taas tarkoitetaan esimerkiksi kokonaislukua, joka TIA Portal -ohjelmistossa on lyhennetty sanalla `int` (integer) (49, s. 245). TIA Portal -ohjelmiston logiikan ohjelmointiominaisuuksista löytyy työkalut näiden ongelmien ratkaisemiseen, joten luettelon käsittely ja säilyttäminen toteutetaan logiikalla.

Luettelon tietoja tullaan säilyttämään TIA Portal -ohjelmiston tarjoamassa tietolohkossa (DataBlock). Tietolohkojen arvot voidaan määrittää säilyttämään haihtumattomassa hajasaantimuistissa (Non-Volatile Random Access Memory), joka kykenee säilyttämään tietoa ilman sähköä (43, s. 1059–1060). Asetus määritetään tietolohkon muuttujien määrittämällä sarakkeessa, jonka otsikossa lukee "Retain" (49, s. 8076–8077). TIA Portal -ohjelmistolla voidaan myös luoda käyttäjän omia tietotyyppisiä (User-defined data type) eli käytännössä yhden muuttujan rakenne voi koostua useasta alimuuttujasta, joilla kaikilla on eri tietotyyppi (49, s. 274–276). Konttisekoittimen TIA Portal -projektissa määritetään tietotyyppi, joka sisältää viisi alimuuttujaa kolmella eri tietotyypillä. Tämän jälkeen kuvan 14 mukaisesti lisätään tietolohkoon uusi muuttuja, joka määritellään luetteloksi ja sen tietotyyppiksi määritellään aikaisemmin luotu tietotyyppi. Tällä ratkaisulla voidaan säilyttää useita eri arvoja ja tietotyyppisiä yhden muuttujan alla.

	Name	Data type	Start value
1	Input		
2	Output		
3	InOut		
4	Static		
5	Eratietokanta	Array["Eranseuranta_RecipeMemory_ArrayMin"..."Eranseuranta_RecipeMemory_ArrayMax"] of "Recipedatatype"	
6	Eratietokanta[1]	"Recipedatatype"	
7	Batch	UDInt	1
8	TotalMixes	Int	0
9	Date	Date_And_Time	DT# 1990-01-01-0
10	CurrentMix	Int	0
11	RecipeNro	Int	0
12	Eratietokanta[2]	"Recipedatatype"	
13	Eratietokanta[3]	"Recipedatatype"	
14	Eratietokanta[4]	"Recipedatatype"	
15	Eratietokanta[5]	"Recipedatatype"	

Kuva 14. TIA Portal -ohjelmistossa tietolohkoon määritetty muuttuja luetteloksi, jolla seurataan keskeneräisiä tuotantoeräiä.

Luettelon käsittelyyn valitaan lausepohjainen PLC-ohjelmointikieli nimeltä Structured Control Language (SCL). SCL on korkeamman tason ohjelmointikieli, joka sopii hyvin esimerkiksi luetteloiden käsittelyyn. (49, s. 8385.) Sillä voi tehdä esimerkiksi for-silmukan, jonka voi ohjelmoida käymään läpi jokaisen arvon luettelossa ja etsiä tiettyä arvoa, kuten eränumeroa. Esimerkkikoodissa 2 on toteutettu viivakoodinlukijalla luetun eränumeron etsiminen luettelosta. Siinä käydään läpi jokainen luettelon alkio, kunnes haettu eränumero löytyy. Kun eränumero on löytynyt, sen arvo kopioidaan toiseen muuttujaan ja se näytetään operointipaneelilla. Lisäksi reseptinumero ja seuraava sekoitusaskel viedään HMI-paneelille, jossa liitteessä 2 oleva VB-funktio käyttää kyseisiä arvoja vaiheessa kolme, jotta se osaa hakea oikean reseptin ja valita oikean sekoitusaskeleen.

```

IF "Eränseuranta".HaeEra AND "Eränseuranta".EraNro <> 0 THEN

    FOR #i := "Eränseuranta_RecipeMemory_ArrayMin" TO
"Eränseuranta_RecipeMemory_ArrayMax" DO

        IF "Eränseuranta".EraNro = "DB.Eratietokanta".Eratieto-
kanta[#i].Batch THEN
            "Eränseuranta".Eräntiedot.Batch := "DB.Eratietokanta".Era-
tietokanta[#i].Batch;
            "Eränseuranta".NextStep := "DB.Eratietokanta".Eratieto-
kanta[#i].CurrentMix;
            "Eränseuranta".ResepetiNro := "DB.Eratietokanta".Eratieto-
kanta[#i].RecipeNro;

            "Eränseuranta".Eraloityti := TRUE;

            EXIT;
        END_IF;
    END_FOR;

    "Eränseuranta".HaeEra := FALSE;

END_IF;

```

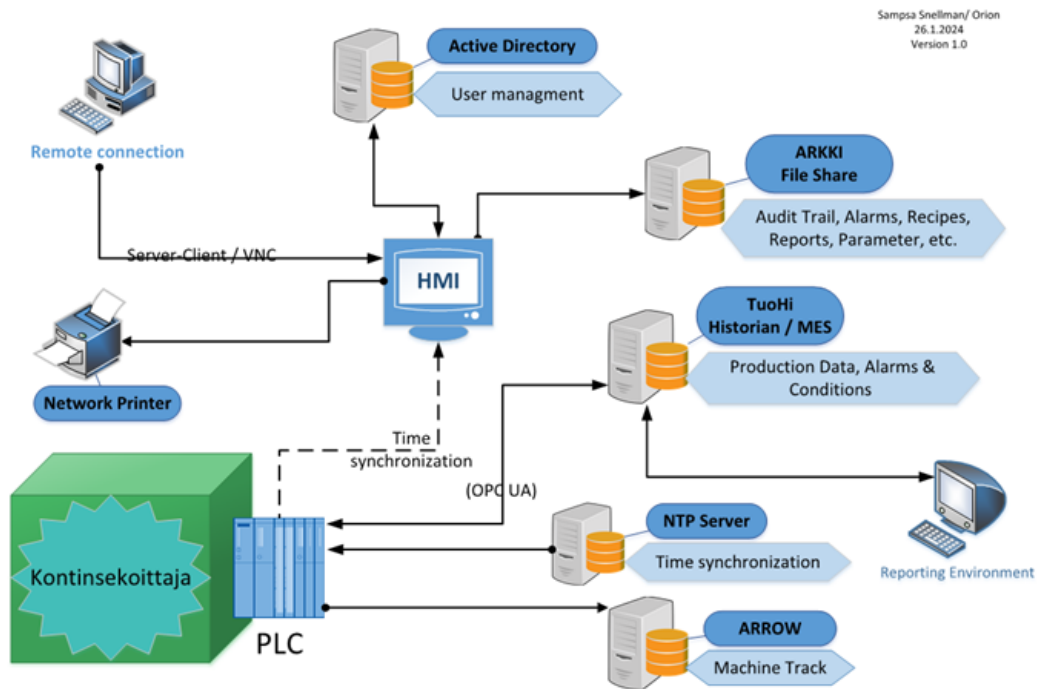
Esimerkkikoodi 2. SCL-ohjelmointikielellä toteutettu eränumeron etsiminen luettelosta.

Oikean eränumeron etsimisen lisäksi täytyy toteuttaa SCL-ohjelmointikielellä uuden erän lisääminen luetteloon ja erän poistaminen luettelosta, kun kaikki sekoitusaskeleet on suoritettu, sekä erän seuraavaan sekoitusaskeleen päivittäminen, kun askel on suoritettu onnistuneesti. Liitteessä 3 on esitetty nämä ominaisuudet SCL-ohjelmointikielellä toteutettuna. Lisäksi täytyy vielä toteuttaa eräluettelon katselmointi ja hallinta operointipaneelilta sekä eräluettelon tallentaminen tiedostoksi ja sen vieminen verkkolevylle.

5.4 Automaatiojärjestelmän liittäminen tehdasjärjestelmiin

Yksi modernisoinnin perusteluista on konttisekoittimen liittäminen Orionin sisäverkkoon, joka mahdollistaa eri tehdasjärjestelmien käyttöönoton. Konttisekoittimen automaatiojärjestelmän integrointi muiden järjestelmien kanssa vaatii toimenpiteitä TIA Portal -ohjelmistossa sekä niissä järjestelmissä, johon konttisekoitin liitetään. Tässä insinööriyössä käydään vain TIA Portal -ohjelmistossa tehtävät toimenpiteet. Kuvassa 15 esitetään, mihin tehdasjärjestelmiin

konttisekoittin liitetään. Kuvassa näkyvien TuoHi -ja ARROW-järjestelmien integraatiota konttisekoittimen automaatiojärjestelmään ei käsitellä tässä työssä.



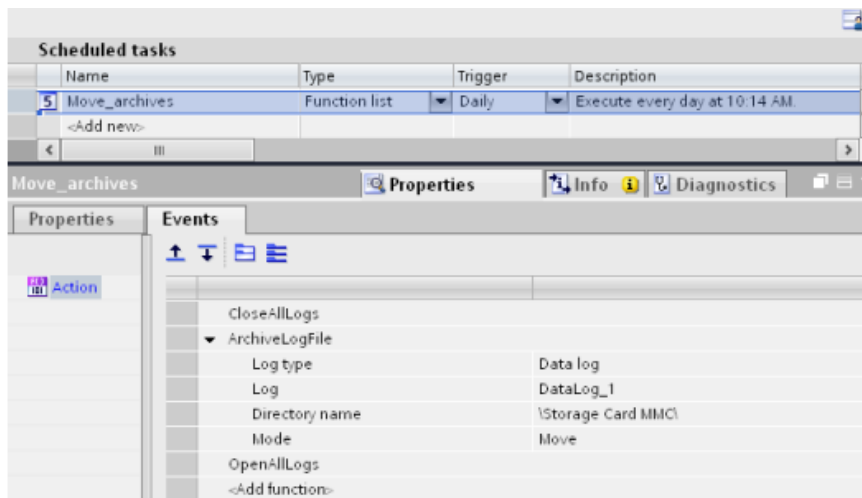
Kuva 15. Suunnitelma tehdasjärjestelmäliitoksista konttisekoittimen logiikan ja operointipaneelin kanssa.

Jaetut verkkolevyt

Konttisekoittimen automaatiojärjestelmän tulee generoida useita tiedostoja, jotka tallennetaan verkkolevyille. Käyttäjävaatimusten (46) mukaan järjestelmän on tallennettava verkkolevyille hälytyshistoria, Audit Trail -tiedosto, sekoitusraportti sekä eräluettelon ja reseptien varmuuskopio. TIA Portal -ohjelmistossa nämä tiedostot voidaan luoda ohjelmiston tarjoamilla valmiilla toiminnoilla, lukuun ottamatta eräluettelon varmuuskopiota, jonka luominen on toteutettava ohjelmallisesti, esimerkiksi VB Script -ohjelmointikielellä (45, s. 898, 1046, 1141, 1287).

Audit Trail - ja hälytyshistoriatiedostot ovat CSV-lokitiedostoja eli käytännössä HMI-paneeli lisää jatkuvasti tietoa niihin, jolloin ne ovat koko ajan käytössä.

Pahimmillaan käytössä olevan tiedoston siirto tai kopiointi voi johtaa sen korrumpoitumiseen ja tehdä sen käyttökelvottomaksi, joten tiedoston siirto tai kopiointi on toteutettava kolmivaiheisesti. Ensimmäiseksi on määritettävä CloseAllLogs-toiminto käynnistettäväksi tapahtuman yhteydessä, esimerkiksi tietty kellonaika tai painonapin painallus. Tämä toiminto sulkee kaikki lokitiedostot, jolloin HMI-paneeli ei voi enää käyttää niitä. Seuraavaksi samaan tapahtumaan määritetään ArchiveLogFile-toiminto, joka toteuttaa tiedoston siirron tai kopioinnin. Toiminnon parametreihin määritetään lokitiedoston tyyppi (esimerkiksi hälytykset tai mittaukset), lokitiedoston nimi sekä kansio polku, jonne lokitiedosto siirretään tai kopioidaan, ja lisäksi valitaan, halutaanko siirtää tai kopioida tiedosto. Lopuksi samaan tapahtumaan lisätään OpenAllLogs-toiminto, joka avaa tiedostot uudelleen. Järjestelmä tunnistaa, että onko tiedoston siirto tai kopiointi vielä kesken, joten tiedostot avataan uudestaan vasta tämän toiminnon jälkeen. Kuvassa 16 on esimerkki TIA Portal -ohjelmiston käyttöohjeesta, kuinka tämän voi toteuttaa käytännössä. (45, s. 1543–1545.)



Kuva 16. TIA Portal -ohjelmiston käyttöohjeessa esitetty esimerkkitapa, kuinka tiedoston siirron voisi toteuttaa (45, s. 1545).

Tiedoston siirto tai kopiointi on suositeltavaa toteuttaa yhdelle tiedostolle kerrallaan, sillä insinööriyön tekijän kokemuksen mukaan HMI-paneelit eivät kykene suorittamaan useita toimintoja samanaikaisesti luotettavasti. Audit Trail -tiedosto on määritettävä siirrettäväksi, koska muuten rikottaisiin Yhdysvaltain

elintarvike- ja lääkeviraston (FDA) ohjeistusta päällekkäisen tiedon luomisesta (45, s. 639). Hälytyshistoriatiedosto taas määritetään kopioitavaksi, jotta lokitiedostoa voidaan selata suoraan operointipaneelilta. Tiedostojen tallentamisella verkko- tai paikalliskansioille ei ole eroa, joten riittää, että ArchiveLogFile-toiminnon parametreihin määritetään suoraan verkkopolku. On kuitenkin huomioitava, että fyysisen paneelin verkkoasetuksissa on määritettävä nimipalvelimen (DNS-palvelin) IP-osoite, jotta paneeli osaa hakea nimen perusteella oikean verkkopolun.

Sekoitusraportin PDF-tiedoston tallentaminen verkkolevylle onnistuu määrittämällä fyysisen paneelin tulostinasetuksista tulostuksen kohdekansioiksi verkkopolun, jolloin paneeli tulostaa PrintRaport-toiminnon avulla suoraan verkkokansioon. Ennen tätä on kuitenkin siirrettävä PDF-tulostinohjain (Printer Driver PDF V1.5) paneelille. (50, s. 7–8.) Reseptien varmuuskopio otetaan ExportDataRecord-toiminolla, jonka voi käynnistää esimerkiksi järjestelmätapahtuma, joka ilmoittaa onnistuneesta reseptin tallennuksesta. Toiminnon parametreihin voidaan asettaa verkkopolku, jolloin reseptit tallennetaan suoraan verkkolevylle. Eräluettelon varmuuskopioinnin siirron toteuttaminen on hieman monimutkaisempaa, koska TIA Portal -ohjelmistossa ei ole sille valmiita toimintoja, joita voisi liittää käynnistymään tapahtuman yhteydessä. Tämän vuoksi käytetään VB Script -ohjelmointikieltä. Eräluettelon varmuuskopioinnin siirto verkkolevylle ja palautus sieltä toteutetaan käyttämällä VB Script -kirjastoa nimeltä ”File System Controls” (51).

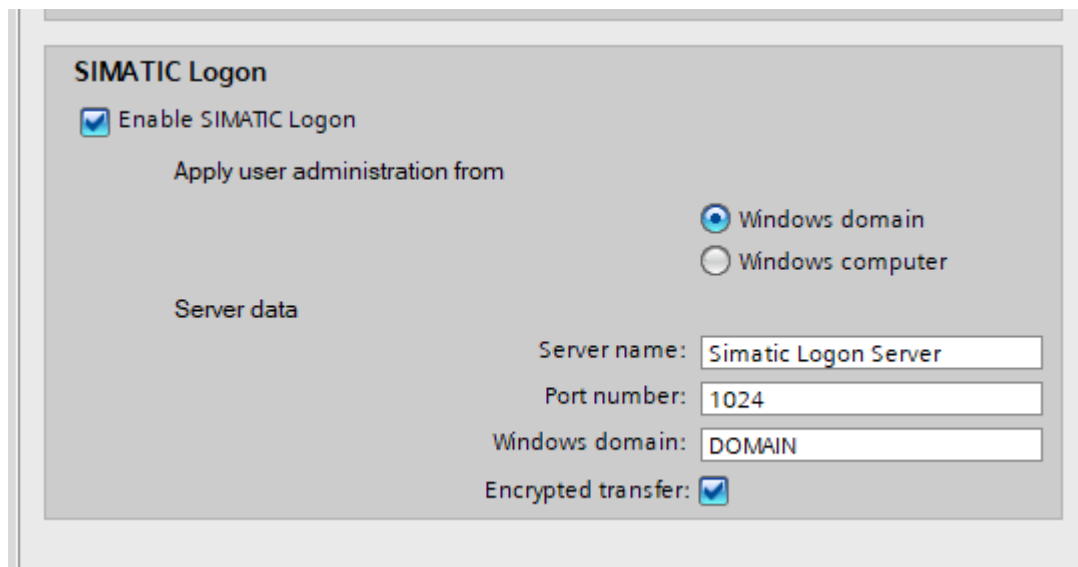
Varmennettu aikapalvelin

Konttisekoittimen liittäminen varmennettuun aikapalvelimeen tehdään TIA Portal -ohjelmiston PLC:n HW-konfigurointi sivulla. Konfigurointisivun välilehdellä nimeltä ”Time-of-day synchronization” määritetään enintään neljän eri NTP-palvelimen IP-osoite. Orionilla on omia NTP-palvelimia, joten niiden IP-osoitteet asetetaan niihin. Useamman palvelimen liittäminen parantaa kellonajan luotettavuutta ja tarkkuutta verrattuna yhden palvelimen käyttöön. Lisäksi on

määritettävä toisella välilehdellä kesäajan siirron ajankohta sekä Suomen aika-
vyöhyke. (25.)

Käyttäjienhallintapalvelin

Operointipaneelia ei voida suoraan yhdistää Orionin käyttäjienhallintapalvelimeen. Ensimmäiseksi operointipaneelin täytyy yhdistyä Siemensin SIMATIC Logon -palvelimeen, joka on yhteydessä esimerkiksi Windowsin käyttäjienhallintapalvelimeen. Orionilla on asennettuna useita SIMATIC Logon -palvelimia, mutta TIA Portal -ohjelmiston HMI-paneelin asetuksista täytyy ainoastaan määrittää yhden palvelimen tiedot kuvan 17 mukaisesti. (45, s. 1343–1347.)



Kuva 17. SIMATIC Logon -palvelimeen yhdistämisen määrittely HMI-paneelin asetuksista. Orionin SIMATIC logon -palvelimen tiedot on muutettu kuvassa tietoturvasyistä.

Etäyhteys operointipaneelille

Smart-palvelimen aktivointi tapahtuu TIA Portal -ohjelmiston HMI-paneelin asetuksista. Tämän avulla jokainen käyttäjä, jolla on verkkoyhteys operointipaneelin Smart-palvelimeen voi käyttää sitä etänä. Fyysisen operointipaneelin asetuksista pystyy asettamaan salasanan etäkäytölle, jotta estetään haitallinen etäkäyttö operointipaneelilla. (26.) On myös tärkeää huomioida, että itse

operointipaneelin käyttö vaatii sisäänkirjautumisen henkilökohtaisilla tunnuksilla, joten haitallisia toimenpiteitä ei voida suorittaa laitteella, vaikka etäyhteyden saisi luotua laitteen käyttöön kouluttamaton henkilö.

5.5 Simulointiympäristön luominen ja ohjelman testaaminen

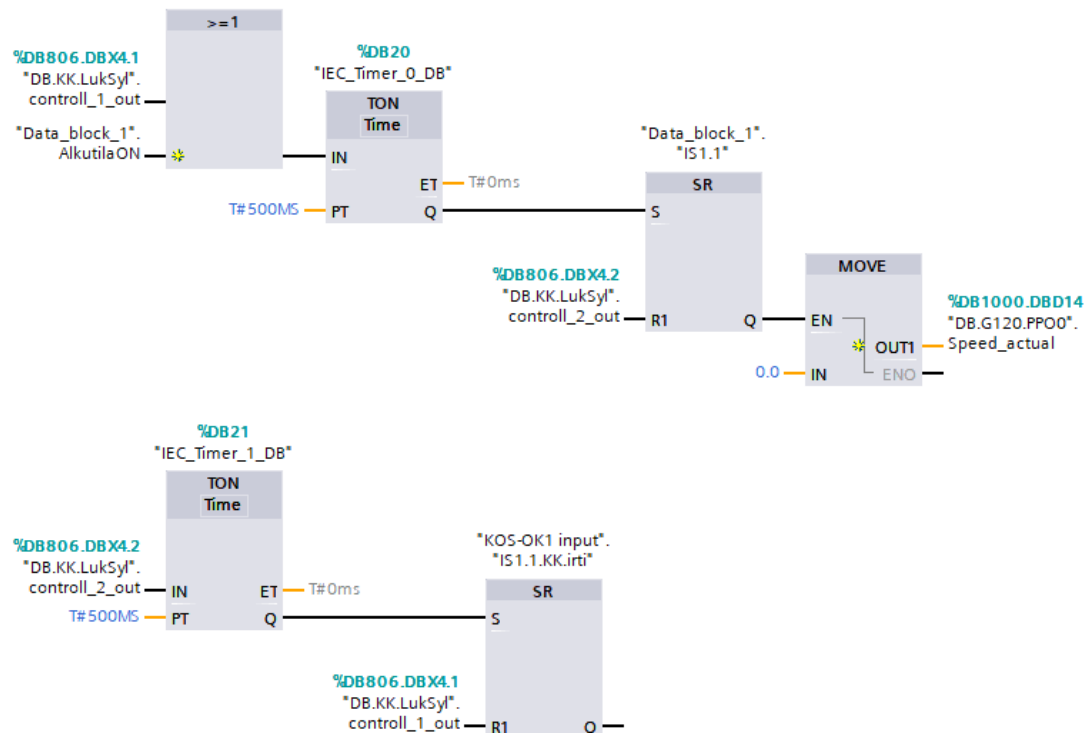
Testaaminen on äärimmäisen tärkeää koko ohjelmiston toteutusprosessin aikana, koska ohjelmistoon on lisätty useita uusia toimintoja, ja yksikin pieni kirjoitusvirhe voi merkittävästi muuttaa ohjelmiston toimintaa. On järkevää testata jokainen uusi toiminto erikseen heti niiden toteutuksen jälkeen, koska ohjelman testaaminen vasta lopussa voi paljastaa suuren määrän virheitä useassa eri toiminnossa, jolloin niiden korjaaminen saattaa olla erittäin työlästä ja vaikeaa. Virallinen testaaminen kuitenkin suoritetaan vasta ohjelman ollessa kokonaan valmis. Ensin ohjelma testataan simuloimalla hyvissä ajoin, ja lopullinen perusteellinen testaaminen suoritetaan fyysisellä konttisekoittimella käyttöönoton yhteydessä. Käyttöönoton aikainen lopullinen testaus ei kuulu tämän insinööriyön aihealueeseen, joten tässä työssä käsitellään vain simuloitun testauksen toteutus.

Simuloituun testaamiseen käytetään Siemensin tarjoamaa S7-PLCSIM V18 -ohjelmaa, joka on integroitu TIA Portal -ohjelmistoon. Ohjelma mahdollistaa TIA Portal -projektin lataamisen virtuaaliselle logiikalle ja HW-konfiguraatiossa logiikkaan liitettyjen AI- tai DI-moduulien kanaviin liitettyjen muuttujien arvojen manipuloinnin. Lisäksi virtuaalisen HMI-paneelin toimintoja voidaan simuloida TIA Portal -ohjelmistossa, jolloin se on yhteydessä virtuaaliseen logiikkaan. (42, s. 221.)

Ongelmana on, että esimerkiksi sylinterien testaaminen virtuaalisesti sekvenssien kanssa on hankalaa, koska venttiilin ohjauksen jälkeen sylinterin auki- tai kiinniraja pitäisi aktivoitua, muuten ohjelma ilmoittaa laiteviasta. Simulointiohjelmalla voidaan manipuloimaan auki- tai kiinnirajaa, mutta silloin ohjelman testaaminen olisi hyvin manuaalista ja työlästä. Ongelman ratkaisemiseksi voidaan luoda erillinen ohjelmakoodi, jossa manipuloidaan ohjelmallisesti tiettyä

tulokanavaa, jonka tilaa toimilaite muuttaisi fyysisessä ympäristössä. TIA Portal -ohjelmistossa ei ole mahdollista ohjelmallisesti muuttaa tulokanavan tilaa, joten täytyy käyttää tietolohkoa (Datablock) apuna. Tietolohkon muuttujiin voidaan ohjelmallisesti siirtää arvoja ja muuttujien arvoja voidaan käyttää jonkin asian ohjaamiseen eli käytännössä tietolohkon muuttuja voi toimia testaamisen ajan DI-moduulin tulokanavana.

Ennen käyttöönottoa on erittäin tärkeää, että kaikki simulointiin liittyvät koodit poistetaan kokonaan ohjelmasta, koska simulointitoimintojen käyttö fyysisellä laitteella voi johtaa vakaviin vahinkoihin. Tästä syystä simulointiin liittyvät ohjelmalohkot täytyy eritellä muista ohjelmalohkoista, ja niiden poistaminen ohjelmasta ei saa vaikuttaa muuhun ohjelmaan. Kuvassa 18 on toteutettuna kehikon lukitsemisen simulointi ohjelmallisesti FBD-ohjelmointikielellä TIA Portal -ohjelmistossa. Kuvan yläosassa ohjataan magneettiventtiiliä, jolloin kuvitteellisen rajakytkimen tilatieto aktivoituu. Lukitsemisen simuloinnissa käytetään SR-lohkoa, joka pitää rajakytkimen tilatietoa aktiivisena, vaikka venttiiliä ei enää ohjata. Tämä johtuu siitä, että fyysisessä ympäristössä venttiilillä ohjataan paineilmaa sylinterille, jotta se voi lukita kehikon. Tämän jälkeen paineilman syöttö lopetetaan kyseiselle sylinterille, ja toiselta magneettiventtiililtä ohjataan paineilmaa samalle sylinterille varmistamaan kehikon lukitus. Sillä ei ole rajakytkimiä, joten sitä ei tarvitse ohjelmallisesti simuloida. Kuvan 18 ajastinlohkoilla lisätään simulointiin todenmukaisuutta. Lisäksi kehikkoa pyörittävän moottorin simuloitu nopeus nollataan kehikon lukitsemisen yhteydessä. Kuvan alaosassa toteutetaan kehikon lukituksen avaus samalla periaatteella kuin lukitseminen.



Kuva 18. Kehikon lukitsemisen simuloinnin toteuttaminen ohjelmallisesti FBD-ohjelmointikielellä.

Lähes kaikki konttisekoittimen toiminnallisuudet voidaan testata, kun kaikkien toimilaitteiden ja rajakytkimien simulointi on toteutettu ohjelmallisesti. Kuvassa 19 testataan konttisekoittimen tuotantoajon toimintaa TIA Portal -ohjelmiston HMI-paneelin simulointityökalulla ja ohjelmallisesti luodun simulointiympäristön avulla. Kuvassa näkyy pääsivu, jossa on käynnissä eräluettelosta löytyvän erän sekoitusvaihe. Liitteen 4 kuvassa 1 testataan kehikon kääntöä 180° huollon tai pesun ajaksi prosessisivulta seurattuna.

The screenshot displays the TIA Portal interface for testing a production process. The top navigation bar includes the Orion logo and user information: Käyttäjärühmä: Pääkäyttäjä, Käyttäjä: Admin, Versio: 11.38.49, and Logoff 4.8.2024. The main menu contains Prosessi, Parametrit, Trendit, Järjestelmä, Häilytyshistoria, Raportit, Reseptit, and Pääsivu.

The recipe details are as follows:

Step	Nopeus	Aika	Astia
1. askel	20 RPM	1 min	2100 l
2. askel	17 RPM	5 min	2100 l
3. askel	10 RPM	1 min	2100 l
4. askel	15 RPM	2 min	2100 l
5. askel	7 RPM	1 min	2100 l

The production status is 'Tuotantoajon tila: Sekoitetaan konttia'. The current speed is 14,7 RPM and the target speed is 15 RPM. The current time is 00:44 and the target time is 02:00. A button 'Keskeytä sekoitus' is visible.

Kuva 19. Tuotantoajon testaaminen TIA Portal -ohjelmiston simulointityökalulla.

Insinööriyön viimeinen vaihe on konttisekoittimen testaaminen simuloimalla. Kaikki ohjelmistossa toteutetut ominaisuudet testattiin onnistuneesti lukuun ottamatta tehdasjärjestelmäyhteyksiä. Lopullinen testaaminen sujui ongelmitta, koska uusia ominaisuuksia oli testattu perusteellisesti jo toteutusvaiheen aikana. Toteutusvaiheen testauksessa ilmeni monia ongelmia, mutta ne olivat helposti ratkaistavissa, koska ohjelmaan tehtiin pieniä muutoksia kerrallaan. Migraation jälkeinen testaaminen ja virheiden korjaaminen osoittautui haastavimmaksi, koska virheiden vuoksi ohjelmaa ei voitu ladata virtuaaliselle logikalle. Tehdasjärjestelmäyhteydet voidaan testata ennen käyttöönottoa fyysisellä PLC:llä ja HMI-paneelilla, kunhan uudet tilatut komponentit ovat saapuneet tehtaalle.

6 Käyttöönottoon valmistautuminen ja työn tulokset

Insinööriyön tarkoituksena oli valita uudet komponentit ja toteuttaa käyttöönottovalmis ohjelma modernisoinnin tarpeessa olevalle konttisekoittimelle. Tavoite saavutettiin onnistuneesti, sillä kaikki suunnitellut ominaisuudet saatiin toteutettua. Modernisointi ei kuitenkaan pääty tähän, vaan seuraavaksi aloitetaan valmistautuminen käyttöönottoon, joka on suunniteltu kesälle 2025. Tämä sisältää dokumentaation päivittämisen ajan tasalle sekä automaatiojärjestelmän fyysisten komponenttien testaamisen automaatiolaboratoriossa, mukaan lukien tehdasjärjestelmäyhteyksien testaukset. Päivitettäviä dokumentteja ovat muun muassa sähköpiirustukset, ohjelmistokuvaus, käyttöohjeet, I/O-luettelo, laiteluettelo, väyläkaavio, riskiarvio sekä monet muut Orionin omat dokumentit (52). Lisäksi käyttöönottoa varten on myös laadittava suunnitelma lopulliselle testaukselle.

Konttisekoittimen modernisoinnin toteutus koostui useista eri vaiheista ja osa-alueista, joista jokaisen suorittaminen saavutti toivotun lopputuloksen. Yhdessä näiden onnistuminen muodosti yhtenäisen toimivan kokonaisuuden. Seuraavaksi tarkastellaan yksityiskohtaisemmin kunkin osa-alueen tuloksia.

Komponenttien valintaprosessia helpotti se, että Orionin varastosta löytyi mahdollisesti sopivia komponentteja, joiden soveltuvuutta konttisekoittimeen tuli arvioida. Tämän myötä Orionin varaosa tilanne paranee, kun samat komponentit käyvät monen eri laitteen automaatiojärjestelmään. Oikeiden komponenttivalintojen myötä uusi toteutettu ohjelma on yhteensopiva automaatiojärjestelmän kanssa. Lisäksi operointipaneelin valinnalla oli suuri merkitys, sillä valittiin selvästi aiempaa suurempi paneeli, joka parantaa merkittävästi käyttökokemusta.

Orionilla kokeiltiin ensimmäistä kertaa vanhan ohjelman migraatiota uudelle PLC:lle. Migraation etuna oli se, että ohjelmaa ei tarvinnut kirjoittaa kokonaan uudelleen. Aluksi ajateltiin, että valmis ohjelma syntyy vain nappulaa painamalla ilman ongelmia. Tämä ei kuitenkaan pitänyt paikkaansa, sillä migraation jälkeen ohjelmaan jäi 69 virhettä korjattavaksi. Ohjelman ymmärtäminen vei myös

aikaa, sillä insinööriyön tekijä ei ollut alkuperäisen ohjelman laatija, ja lisäksi yksi siinä käytetty ohjelmointikieli (ST-ohjelmointikieli) ei ollut täysin tuttu. Migraation tekeminen on kannattavaa, jos tekijällä on vahva ohjelmalukutaito eri PLC-ohjelmointikielillä, ja erityisesti silloin, kun laitteen toiminnallisuus ei muutu merkittävästi. Suurten ohjelmamuutosten yhdistäminen vanhaan koodiin voi tehdä siitä sekavan ja työläämmin toteutettavaksi.

Vanhasta käyttöliittymästä oli olemassa migroitu TIA Portal -projekti, mutta lähes kaikki näyttösivut jouduttiin rakentamaan tyhjältä pohjalta, koska uuteen käyttöliittymään tuli paljon uusia ominaisuuksia, ja vanhoja ominaisuuksia siirrettiin muihin sivuihin tai poistettiin kokonaan käyttöliittymästä. Migroitu käyttöliittymä kuitenkin helpotti työtä siinä mielessä, että hälytykset ja HMI-muuttujat oli valmiiksi määritelty, joten muuttujien siirtäminen vanhoista näyttöelementeistä uusiin oli suoraviivaista.

Uuden käyttöliittymän toteutuksen laajuus yllätti ja sen toteuttaminen vaati eniten aikaa. Työtä helpotti hieman se, että käytössä oli muita Orionin TIA Portal -projekteja, joita voitiin käyttää viiteprojekteina. Uuden käyttöliittymän merkittävimmät parannukset ja uudet ominaisuudet ovat muun muassa parannettu navigointi sivujen välillä, parempi diagnostiikan hallinta, erien seuranta, selkeämpi tuotantoajon seuranta, reseptien tallentaminen, raporttien katselu, mittausarvojen tarkastelu trendimuodossa sekä käyttöliittymän nykyaikaistaminen visuaalisesti.

Reseptien hallintajärjestelmä on täysin uusi ominaisuus konttisekoittimella ja se täytyi toteuttaa puhtaalta pöydältä. Toteuttamista helpotti se, että käytössä oli muita TIA Portal -projekteja viitteenä, joissa reseptien hallintajärjestelmä oli jo toteutettu. Ominaisuuden toteuttaminen vaati paljon VB Script -ohjelmointikielen käyttämistä, sillä TIA Portal -ohjelmistosta ei löytynyt valmiita toiminnallisuuksia vaadittujen ominaisuuksien toteuttamiseksi. Suuri osa ajasta kuluikin kyseisen ohjelmointikielen opiskeluun. Käyttöänon jälkeen uusi reseptien hallintajärjestelmä vapauttaa operaattoreilta aikaa muihin työtehtäviin ja vähentää inhimillisten virheiden mahdollisuutta.

Erien seuraamisjärjestelmää ei ollut valmiiksi toteutettuna missään muussa Orionin TIA Portal -projektissa, joten se täytyi suunnitella ja toteuttaa kokonaan ilman TIA Portal -viiteprojekteja. Käyttäjän itse määrittämistä tehdyistä muuttujista (User-defined data type) oli merkittävä hyöty, koska niiden avulla voitiin pakata koko eräluettelo yhden muuttujan sisälle. Käyttöönoton jälkeen järjestelmä tallentaa uudet erät luetteloon ja seuraa niiden sekoitusaskeleita, jonka myötä järjestelmä osaa hakea aina oikean sekoitusaskeleen parametrit reseptistä.

Konttisekoittimen liittäminen tehdasjärjestelmiin tapahtuu käyttöönotossa, joten TIA Portal -ohjelmistossa tehtiin valmistelevat toimenpiteet, joiden avulla konttisekoittimen liittäminen tehdasjärjestelmiin onnistuu vaivattomasti. Suurin osa työstä oli asetusten määrittämistä, mikä sujui nopeasti, lukuun ottamatta eräluettelon siirtoa, jonka toteuttaminen vaati VB Script -ohjelmointikielen osaamista.

Riskinä on, että kaikki toiminnot eivät välttämättä toimi oikein, koska tehdasjärjestelmiin liittäminen ei voida testata ilman fyysistä paneelia ja ohjelmoitavaa logiikkaa. Suurinta osaa toiminnoista ei tarvinnut ohjelmoida, vaan asetukset voitiin määrittää suoraan käyttöohjeiden (45) mukaisesti, mikä vähentää virheiden riskiä. Ongelmat voivat kuitenkin johtua myös tehdasjärjestelmien määryksistä. Näitä toimintoja on mahdollista esitestata ennen varsinaista käyttöönottoa, jos fyysisen paneelin ja logiikan voi yhdistää tehdasverkkoon. Käyttöönoton jälkeen konttisekoittimen kriittiset tiedot tallennetaan verkkolevyille, ja laitteistoa voivat käyttää vain koulutetut käyttäjät henkilökohtaisilla tunnuksilla. Näin konttisekoittimen GMP paranee.

Konttisekoittimen simuloinnin toteuttaminen ohjelmallisesti osoittautui erittäin hyödylliseksi uusien toimintojen testaamisessa. Vaikka sen toteuttamiseen kului jonkin verran aikaa, niin sen toteuttamiseen kuluneen ajan sai nopeasti takaisin, sillä ongelmakohtat voitiin testata nopeasti useita kertoja, jonka myötä oli helpompi selvittää, mistä ongelma johtui. Lisäksi voitiin toteuttaa laaja-alaisempi simuloitu lopputestaus, mikä vähentää varsinaisessa käyttöönotossa ilmeneviä

mahdollisia ongelmia. Konttisekoittimen simuloinnista voisi olla hyötyä myös tulevaisuudessa, esimerkiksi operaattorien kouluttamista varten.

7 Yhteenveto

Insinööriyössä toteutettiin konttisekoittimen automaatiojärjestelmän modernisointi. Modernisointi oli tarpeen, koska automaatiojärjestelmän komponentit ovat saapuneet elinkaarensa päähän, ja lisäksi konttisekoittimeen haluttiin lisätä uusia ominaisuuksia, joilla laitteen luotettavuus, turvallisuus ja kustannustehokkuus paranee. Työssä käsiteltiin uuden ohjelman toteutus ja uusien automaatio-komponenttien valintaprosessi. Ohjelman toteuttaminen, joka sisälsi reseptien hallintajärjestelmän, operointipaneelin käyttöliittymän, PLC:n ohjelman ja tehdasjärjestelmäliitokset, onnistui kokonaisuudessaan. Ohjelman testaaminen simuloimalla onnistui, lukuun ottamatta tehdasjärjestelmäliitoksia, jotka voidaan testata vasta fyysisen HMI-paneelin ja PLC:n ollessa käytettävissä.

Haastavinta oli migraation jälkeisten virheiden korjaaminen, sillä ohjelmaa ei voitu testata yhden virheen korjaamisen jälkeen, vaan kaikki virheet oli korjattava ennen kuin ohjelma voitiin ladata virtuaaliselle PLC:lle. Eniten aikaa vei käyttöliittymän rakentaminen, koska sen toteuttaminen aloitettiin lähes tyhjältä pohjalta.

Insinööriyö sisälsi useita eri osa-alueita, joitten toteuttaminen vaati paljon opiskelua ja selvitystyötä. Työstä saatu kokemus luo hyvän pohjan tuleville modernisointiprojekteille. Lisäksi tavoitteena oli insinööriyön tekijän kirjoittaa raportti, jota opiskelijan tai insinöörin on helppo seurata ja soveltaa omissa projekteissaan. Insinööriyössä käsitellään sekä TIA Portal -ohjelmiston perustoimintoja, että hieman monimutkaisempia ominaisuuksia, joten raportti sopii automaatioinsinööreille, alaa opiskeleville tai muuten aiheesta kiinnostuneille.

Lähteet

- 1 WI 30968, käyttö ja puhdistus: konttisekoitin Finname. 2022. Yrityksen sisäinen aineisto. Orion Oyj.
- 2 Konttisekoitin, operointipäätteen käyttöohje. 2016. Yrityksen sisäinen aineisto. Orion Oyj.
- 3 KONTTISEKOITTIMEN TOIMINTAKUVAUS. Yrityksen sisäinen aineisto. Orion Oyj.
- 4 What is an industrial bus? 2019. Verkkoaineisto. Sumcab. <<https://sumcab.com/en/noticias/what-industrial-bus>>. Luettu 19.3.2024.
- 5 2T2 KOS laiteluettelo. Yrityksen sisäinen aineisto. Orion Oyj.
- 6 Bowne, Michael. 2020. THE DIFFERENCE BETWEEN PROFIBUS AND PROFINET. Verkkoaineisto. PI North America. <<https://us.PROFINET.com/the-difference-between-PROFIBUS-and-PROFINET/>>. 10.6.2020. Luettu 20.3.2024.
- 7 Communication with SIMATIC. 2006. Verkkoaineisto. Siemens. <https://cache.industry.siemens.com/dl/files/686/1254686/att_46478/v1/S7komm_e.pdf>. Luettu 17.3.2024.
- 8 What Is a Programmable Logic Controller (PLC)? 2021. Verkkoaineisto. Polycase. <<https://www.polycase.com/techtalk/electronics-tips/what-is-a-programmable-logic-controller.html>>. Luettu 16.3.2024.
- 9 6ES7315-2AH14-0AB0. Verkkoaineisto. Siemens. <<https://support.industry.siemens.com/cs/pd/155410?pti=td&dl=en&lc=en-FI>>. Luettu 17.3.2024.
- 10 What is IO Module in PLC: A Comprehensive Overview. Verkkoaineisto. SonnePower. <<https://www.sonnepowercontrol.com/company-news/what-is-io-module-in-plc-a-comprehensive-overview.html>>. Luettu 17.3.2024.
- 11 Types of I/O Modules. Opintomateriaali. Arab Academy for Science, Technology and Maritime Transport.
- 12 Cox, Shawn. 2022. Distributed I/O delivers. Verkkoaineisto. Control Design. <<https://www.controldesign.com/connections/i-o/article/11288083/distributed-i-o-delivers>>. Luettu 17.3.2024.

- 13 What is Frequency Converter? How it works? Verkkoaineisto. Gozuk. <<http://www.frequencyinverter.org/what-is-frequency-converter-how-it-works-631601.html>>. Luettu 19.3.2024.
- 14 What you need to know about SINAMICS G120. Verkkoaineisto. Siemens. <<https://www.siemens.com/us/en/products/drives/sinamics-electric-drives/low-voltage-drives/standard-performance-drives/sinamics-g120.html>>. Luettu 19.3.2024.
- 15 Leinonen, Miikka. 2022. LÄMPÖLAITOKSEN AUTOMAATIOJÄRJESTELMÄN MODERNISOINTI. Insinööriyö. Lapin Ammattikorkeakoulu. Theseus-tietokanta.
- 16 OHJELMISTOKUVAUS. 2016. Yrityksen sisäinen aineisto. Orion Oyj.
- 17 Programming with STEP 7. 2010. Verkkoaineisto. Siemens. <https://cache.industry.siemens.com/dl/files/825/109751825/att_933142/v1/STEP_7_-_Programming_with_STEP_7.pdf>. Luettu 24.3.2024.
- 18 WinCC flexible 2008. 2008. Verkkoaineisto. Siemens. <<https://docs.rs-online.com/d132/0900766b80dbac59.pdf>>. Luettu 24.3.2024.
- 19 About Orion. Verkkoaineisto. Orion Oyj. <<https://www.orion.fi/en/about-orion/>>. Luettu 14.2.2024.
- 20 Saarinen, Tytti. 2023. Laatuksikirjan päivittäminen. Insinööriyö. Turun ammattikorkeakoulu. Theseus-tietokanta.
- 21 Edge2Cloud Desigo Optic User Guide. 2022. Verkkoaineisto. Siemens. <<https://sid.siemens.com/r/A6V13068902.>> 21.11.2022. Luettu 24.3.2024.
- 22 Description of the product lifecycle for SIMATIC HMI products (PLM). 2021. Verkkoaineisto. Siemens. <[https://support.industry.siemens.com/cs/document/91688124/description-of-the-product-lifecycle-for-simatic-hmi-products-\(plm\)?dti=0&lc=en-US.](https://support.industry.siemens.com/cs/document/91688124/description-of-the-product-lifecycle-for-simatic-hmi-products-(plm)?dti=0&lc=en-US.)> 1.12.2021. Luettu 27.3.2024.
- 23 Creating a Standard Recipe View and Recipe Screen. 2021. Käyttöohje. Siemens.
- 24 Time synchronization with an online NTP server for SIMATIC S7-1500 Controller. 2021. Verkkoaineisto. Siemens. <https://cache.industry.siemens.com/dl/files/548/109800548/att_1075570/v1/109800548_Time_synchronization_with_an_online_NTP_server_en.pdf.> Luettu 29.3.2024.

- 25 Kirwan, Paul. 2022. Network Time Protocol (NTP). Verkkoaineisto. Tech-Target. <<https://www.techtarget.com/searchnetworking/definition/Network-Time-Protocol>>. Luettu 29.3.2024.
- 26 Scott, Henry. 2020. Siemens Comfort Panel Sma@rtServer Setup. Verkkoaineisto. C&E Advanced Technologies. <<https://www.ceadvancedtech.com/files/resources/downloads/ComfortPanelSmartServer-Setup.pdf>>. 22.4.2020. Luettu 3.4.2024.
- 27 Virtual Network Computing. 1998. Verkkoaineisto. ORL. <<https://web.mit.edu/cdsdev/src/howitworks.html>>. Luettu 18.2.2024.
- 28 Pitkäniemi, Tomi. 2023. Automation Manager, Design and Development, Orion Oyj, Espoo. Keskusteltu 5.3.2024.
- 29 Suonpää, Olli. 2023. Automation Engineer, Design and Development, Orion Oyj, Espoo. Keskusteltu 11.3.2024.
- 30 Data sheet, 6ES7515-2AM02-0AB0. 2024. Tietolehtiö. Siemens. 17.3.2024.
- 31 SIMATIC ET 200SP. Verkkoaineisto. Siemens. <<https://mall.industry.siemens.com/mall/en/ww/catalog/products/10144488?tree=CatalogTree>>. Luettu 27.4.2024.
- 32 SIMATIC ET200SP Analog input module AI 2xI 2-/4-wire ST (6ES7134-6GB00-0BA1) Manual. 2018. Käyttöohje. Siemens.
- 33 6SL3225-0BE32-2AA0. Verkkoaineisto. Siemens. <<https://mall.industry.siemens.com/mall/en/FI/Catalog/Product/?mlfb=6SL3225-0BE32-2AA0&SiepCountryCode=FI>>. Luettu 29.4.2024.
- 34 6SL3244-0BA21-1PA0. Verkkoaineisto. Siemens. <<https://support.industry.siemens.com/cs/pd/40129?pti=pi&dl=en&lc=en-FI>>. Luettu 29.4.2024.
- 35 6SL3255-0AA00-4BA1. Verkkoaineisto. Siemens. <<https://support.industry.siemens.com/cs/pd/107764?pti=pi&dl=en&lc=en-FI>>. Luettu 29.4.2024.
- 36 Motion Control Drives, SINAMICS Converters for Single-Axis Drives Built-In Units. 2024. Tuoteluettelo. Siemens.
- 37 6AV6643-0CB01-1AX1. Verkkoaineisto. Siemens. <<https://mall.industry.siemens.com/mall/en/ww/Catalog/Product/6AV6643-0CB01-1AX1>>. Luettu 1.5.2024.

- 38 Human Machine Interface Systems/ PC-based Automation. 2023. Tuoteluettelo. Siemens.
- 39 Lins, Letácio. The Complete Practical Guide to Siemens Tia Portal Programming. Verkkoaineisto. <<https://www.solisplc.com/tutorials/a-practical-guide-to-siemens-tia-portal-programming>>. Luettu 18.2.2024.
- 40 Guide for Migrating SIMATIC S7-300/S7-400 to SIMATIC S7-1500 and TIA Portal. 2017. Käyttöohje. Siemens.
- 41 STEP 7 and WinCC Engineering V18, Migrating projects and programs. 2022. Käyttöohje. Siemens.
- 42 STEP 7 and WinCC Engineering V18, Using online and diagnostics functions. 2022. Käyttöohje. Siemens.
- 43 STEP 7 and WinCC Engineering V18, Editing devices and networks. 2022. Käyttöohje. Siemens.
- 44 Jalkanen, Jussi. 2024. Senior Automation Manager, Process and Machine automation. Orion Oyj, Espoo. Keskusteltu 18.7.2024.
- 45 STEP 7 and WinCC Engineering V18, WinCC (TIA Portal) Visualizing processes. 2022. Käyttöohje. Siemens.
- 46 Konttisekoittaja URS 2 11 070. 2024. Yrityksen sisäinen aineisto. Orion Oyj.
- 47 Pölynpoisto, TIA Portal -projekti. 2024. Yrityksen sisäinen aineisto. Orion Oyj. Ohjelmaversio 1.4.
- 48 Servolift, TIA Portal -projekti. 2023. Yrityksen sisäinen aineisto. Orion Oyj. Ohjelmaversio 230216.
- 49 STEP 7 and WinCC Engineering V18, Programming the PLC. 2022. Käyttöohje. Siemens.
- 50 Printing with SIMATIC HMI Operator Panels. 2021. Käyttöohje. Siemens.
- 51 VB Script Wince File System Controls. VB Script -kirjasto.
- 52 Espoon konttisekoittimen dokumenttillistaus. Yrityksen sisäinen aineisto. Orion Oyj.

Ohjelman kääntämisen raportti logiikan migraation jälkeen

Compiling finished (errors: 69; warnings: 20)

!	Path	Description
✖	PLC_1	
✖	Program blocks	
✔	parametrit (FB11)	Block was successfully compiled.
✔	DB.parametrit (DB11)	Block was successfully compiled.
⚠	DS1.mittaus (FB12)	
✔	DB.DS1_mittaus (DB12)	Block was successfully compiled.
✔	ET200SP input (DB15)	Block was successfully compiled.
✔	subtract_deg (FC122)	Block was successfully compiled.
✔	add_deg (FC123)	Block was successfully compiled.
✔	deg_is_in_region (FC124)	Block was successfully compiled.
✔	deg_diff (FC125)	Block was successfully compiled.
✔	kontin kulman paikoitus...	Block was successfully compiled.
✔	kontin kulman haku (DB...	Block was successfully compiled.
✔	FAP (ONO_3 FUNCTION)...	Block was successfully compiled.
✔	DB.AK.LukSyl (DB808)	Block was successfully compiled.
✔	DB.YK.LukSyl (DB810)	Block was successfully compiled.
✔	DB.G120 (DB800)	Block was successfully compiled.
✔	DB.Etuovi (DB801)	Block was successfully compiled.
✔	DB.Takaovi (DB802)	Block was successfully compiled.
✔	DB.YK.LinSylinteri (DB8...	Block was successfully compiled.
✔	DB.KK.LukSyl.aktivoi (D...	Block was successfully compiled.
✔	DB.KK.LukSyl (DB806)	Block was successfully compiled.
✔	DB.AK.LukSyl.aktivoi (D...	Block was successfully compiled.
✔	DB.YK.LukSyl.aktivoi (D...	Block was successfully compiled.
✔	DB.DS1_aktivointi (DB8...	Block was successfully compiled.
✔	DB.AK.LinSylinteri (DB8...	Block was successfully compiled.
✔	ET200SP output (DB14)	Block was successfully compiled.
✔	HW.paineilma (FB59)	Block was successfully compiled.
✔	DB.HW.Paineilma (DB59)	Block was successfully compiled.
✖	RS1.ohjaus (FB100)	
✖	Network 12	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 12	A preceding Boolean logic operation is missing.
✔	DB.RS1.ohjaus (DB100)	Block was successfully compiled.
✔	FAP (AME FUNCTION) (F...	Block was successfully compiled.
✔	DB.Analog.RIKKI-PIW32...	Block was successfully compiled.
✔	DB.Analog.DS1.etäisyys ...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW32...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW32...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW32...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW33...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW33...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW33...	Block was successfully compiled.
✔	DB.Analog.VARA-PIW33...	Block was successfully compiled.
⚠	OP1 ohjaus (FB81)	
✔	DB.OP1 (DB81)	Block was successfully compiled.
✔	OP.sivunHaku (FC11)	Block was successfully compiled.
✔	ohjelman alkurutiinit (F...	Block was successfully compiled.
✔	ohjelman loppurutiinit (...)	Block was successfully compiled.

Kuva 1. Ohjelman kääntämisen raportti migraation jälkeen TIA Portal -ohjelmistossa.

	▼ SEQ7 (FB7)	
	Network 8	The address is not occupied by a tag.
	Network 14	The address specified for "S71500" is incomplete.
	Network 45	The operand required at the input or output is missing or has an incorrect data type.
	Network 45	The operand required at the input or output is missing or has an incorrect data type.
	Network 45	The operand required at the input or output is missing or has an incorrect data type.
	Network 45	The operand required at the input or output is missing or has an incorrect data type.
	Network 45	Block 'TON' is not supported by the CPU or library version you are using.
	Network 45	Block 'TON' is not supported by the CPU or library version you are using.
	Network 45	A preceding Boolean logic operation is missing.
	DB.SEQ7 (DB7)	Block was successfully compiled.
	▼ SEQ6 (FB6)	
	Network 6	The address is not occupied by a tag.
	Network 14	The address specified for "S71500" is incomplete.
	Network 36	The operand required at the input or output is missing or has an incorrect data type.
	Network 36	The operand required at the input or output is missing or has an incorrect data type.
	Network 36	The operand required at the input or output is missing or has an incorrect data type.
	Network 36	The operand required at the input or output is missing or has an incorrect data type.
	Network 36	Block 'TON' is not supported by the CPU or library version you are using.
	Network 36	Block 'TON' is not supported by the CPU or library version you are using.
	Network 36	A preceding Boolean logic operation is missing.
	DB.SEQ6 (DB6)	Block was successfully compiled.
	▼ SEQ5 (FB5)	
	Network 6	The address is not occupied by a tag.
	Network 16	The address specified for "S71500" is incomplete.
	Network 28	The operand required at the input or output is missing or has an incorrect data type.
	Network 28	The operand required at the input or output is missing or has an incorrect data type.
	Network 28	The operand required at the input or output is missing or has an incorrect data type.
	Network 28	The operand required at the input or output is missing or has an incorrect data type.
	Network 28	Block 'TON' is not supported by the CPU or library version you are using.
	Network 28	Block 'TON' is not supported by the CPU or library version you are using.
	Network 28	A preceding Boolean logic operation is missing.
	DB.SEQ5 (DB5)	Block was successfully compiled.
	report_data (FB50)	Block was successfully compiled.
	DB.reportdata (DB50)	Block was successfully compiled.
	▼ FAP (MICROMASTER PPO...	
	Network 1	The address specified for "S71500" is incomplete.
	Network 2	The address specified for "S71500" is incomplete.
	Network 5	The address specified for "S71500" is incomplete.
	Network 8	The address specified for "S71500" is incomplete.
	Network 11	The address specified for "S71500" is incomplete.
	DB.G120.PPO0 (DB1000)	Block was successfully compiled.
	▶ RAMP (FC51)	

Kuva 2. Ohjelman kääntämisen raportti migraation jälkeen TIA Portal -ohjelmistossa.

✖	▼ SEQ4 (FB4)	
⚠	Network 7	The address is not occupied by a tag.
✖	Network 16	The address specified for "S71500" is incomplete.
✖	Network 43	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 43	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 43	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 43	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 43	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 43	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 43	A preceding Boolean logic operation is missing.
✔	DB.SEQ4 (DB4)	Block was successfully compiled.
✖	▼ SEQ3 (FB3)	
⚠	Network 6	The address is not occupied by a tag.
✖	Network 14	The address specified for "S71500" is incomplete.
✖	Network 26	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 26	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 26	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 26	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 26	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 26	A preceding Boolean logic operation is missing.
✔	DB.SEQ3 (DB3)	Block was successfully compiled.
✖	▼ SEQ2 (FB2)	
⚠	Network 7	The address is not occupied by a tag.
✖	Network 15	The address specified for "S71500" is incomplete.
✖	Network 38	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 38	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 38	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 38	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 38	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 38	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 38	A preceding Boolean logic operation is missing.
✔	DB.SEQ2 (DB2)	Block was successfully compiled.
✖	▼ SEQ1 (FB1)	
⚠	Network 11	The address is not occupied by a tag.
✖	Network 17	The address specified for "S71500" is incomplete.
✖	Network 58	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 58	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 58	The operand required at the input or output is missing or has an incorrect data type.
✖	Network 58	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 58	Block 'TON' is not supported by the CPU or library version you are using.
✖	Network 58	A preceding Boolean logic operation is missing.
✔	DB.SEQ1 (DB1)	Block was successfully compiled.
✖	▼ DP: SYSTEMdiag DP (FB...	
✖	Network 1	Block 'RDSYSST' is not supported by the CPU or library version you are using.

Kuva 3. Ohjelman kääntämisen raportti migraation jälkeen TIA Portal -ohjelmistossa.

✓	DB.DP: DP DIAGNOSTIIK...	Block was successfully compiled.
✓	Muut: CLOCK (FB54)	Block was successfully compiled.
✓	DB.Muut: PLC_clock (DB...	Block was successfully compiled.
✗	▼ Muut: Aikaohjaus (FB55)	
✗	Network 2	Data type Date_And_Time cannot be implicitly converted to data type Int.
✓	DB.Muut: Aikaohjaus (D...	Block was successfully compiled.
✓	FAP (AI SCALE FUNCTIO...	Block was successfully compiled.
⚠	▶ FAP (SIWA-U IO FUNCTI...	
✓	FAP (PID TYPE FUNCTIO...	Block was successfully compiled.
✓	PID-750 (DB750)	Block was successfully compiled.
✓	DB.Muut: Versionhallint...	Block was successfully compiled.
✓	HelperF.FILTER 10 % (FC...	Block was successfully compiled.
✓	- mittaus ja säätö - (FC...	Block was successfully compiled.
✓	- laitekutsut - (FC100)	Block was successfully compiled.
✓	- laitelukitukset - (FC1...	Block was successfully compiled.
✓	raw_to_deg (FC120)	Block was successfully compiled.
✓	deg_to_raw (FC121)	Block was successfully compiled.
✓	OB: Cycle Execution (O...	Block was successfully compiled.
✓	OB: ToD Interrupt 0 (OB...	Block was successfully compiled.
✗	▼ OB: AikakriittisetOhj. (O...	
✗	Network 8	Block 'PNIO_RECV' is not supported by the CPU or library version you are using.
✗	Network 13	Block 'PNIO_SEND' is not supported by the CPU or library version you are using.
✓	OB: Cycle Time Fault (O...	Block was successfully compiled.
✓	OB: I/O Point Fault 1 (O...	Block was successfully compiled.
✓	OB: Loss of Rack Fault (...)	Block was successfully compiled.
✓	OB: Complete Restart (O..)	Block was successfully compiled.
✓	OB: Programming Error ...	Block was successfully compiled.
✓	OB: MOD_ERR (OB122)	Block was successfully compiled.
✓	Helper.NoiseFilter (FB99)	Block was successfully compiled.
✓	HW.esikäsitteily (FB56)	Block was successfully compiled.
✓	DB.HW.Esikäsittely (DB56)	Block was successfully compiled.
✓	Helper.WaitStateSequen..	Block was successfully compiled.
✓	HW.Apufunktiot (FB57)	Block was successfully compiled.
✓	DB.HW.Apufunktiot (DB...	Block was successfully compiled.
✗	▼ Helper.InputLock (FB58)	
✗	Network 8	Block 'SAVE' is not supported by the CPU or library version you are using.
✗	▼ - input-lukitukset - (FB...	
✗	Network 58	The address specified for "S71500" is incomplete.
✓	DB.inputLukitukset (DB...	Block was successfully compiled.
⚠	▼ General warnings	
⚠		Inputs or outputs are used that do not exist in the configured hardware.
✗		Compiling finished (errors: 69; warnings: 20)

Kuva 4. Ohjelman kääntämisen raportti migraation jälkeen TIA Portal -ohjelmistossa.

Viivakoodin luentaan liittyvien toimintojen toteuttaminen VBScript-ohjelmointikielellä TIA Portal -ohjelmistossa

```
' tämä ohjelma etenee vaiheittain
' kun viivakoodinluku avataan, vaihe saa arvon 0 ja sitä
'kasvatetaan vaiheiden edetessä
Dim vaihe,recipenro, i, IOFIELD, minnro, maxnro, valittunimi, valittu-
resepti, valittuastia, astiaX, astiaR, astiavirhe, Currentstep, j,
touhuua
vaihe = SmartTags("BARvaihe")

*****VAIHE1
'Tähän vaiheeseen siirrytään, kun ollaan luettu eränumero.
'Eränumero tarkastetaan ja syötetään PLC:n muuttujaan, jossa ohjelma
'tarkastaa, että
'löytyykö samanlaista eränumeroa luettelosta
If vaihe = 1 Then
    ' nyt on luettu eränumero, tarkastellaan sen järkevyyttä
    minnro = 1000000
    maxnro = 9999999
    If SmartTags("BAReranro") < minnro Or SmartTags("BAReranro") >
maxnro Then
        SmartTags("BAReratarkistus") = True
        SmartTags("Erastatus") = 0

        Set IOFIELD = HmiRuntime.Screens("11 - Viivakoodinluku").Scree-
nItems("Eränumero")
        IOFIELD.Activate

    Else
        SmartTags("HaeEra") = True
        SmartTags("BARvaihe") = 2
        SmartTags("BAReratarkistus") = False
        SmartTags("BARTuotenimi") = "***LUE TUOTENUMERO**"
        SmartTags("Eränsuranta_EraNro") = SmartTags("BAReranro")

        If SmartTags("CurrentUserGroup") < 2 Then
            SetScreenKeyboardMode hmiOff

    Else
        SetScreenKeyboardMode hmiOn

End If

    Set IOFIELD = HmiRuntime.Screens("11 - Viivakood-
inluku").ScreenItems("Tuotenumero")
    IOFIELD.Activate

End If
End If

*****VAIHE4
'Tässä vaiheessa etsitään oikea askel tuotteelle ja siirretään
```

```
'reseptin mukaiset arvot PLC:lle ja näytetään kaikki reseptin mukaiset
'arvot paneelilla
    'Tämä vaihe ajetaan, kun painetaan
'hyväksy tuotetiedot" -painiketta
If vaihe = 4 Then

    SmartTags("Eränumero_tuotanto") =SmartTags("BAReranro")
    SmartTags("NextStepHMI") = SmartTags("Eränseuranta_Currentstep")

    'Tässä funktiossa siirretään reseptin parametrit PLC:n muuttujiin
    Call VB_Tuotantoon_xnew()

    'Tässä funktiossa valitaan oikea sekoitusaskel erälle
    Call VB_ValitseAskel()

End If

'*****VAIHE 2
'Tämä vaihe käynnistetään vain, jos eränumeroa ei löytynyt luettelosta
'Vaiheessa etsitään oikea resepti tuotteelle, joka luettiin
'viivakoodinlukijalla
If vaihe = 2 And SmartTags("Eränseuranta_Eraaeiloytynyt") Then

    'varsinainen etsintä vain kerran

If SmartTags("EtsittyResepti_LKM") < 1 Then

    ' Tuotenumero on syötetty tehdään alkutarkistus onko koodi järke-
vällä alueella
    minnro = 100000
    maxnro = 999999
    If SmartTags("BARTuotenro") < minnro Or SmartTags("BARTuotenro") >
maxnro Then
        SmartTags("BARTuotetarkistus") = True
        SmartTags("BARTuotenimi") = "***LUE UUDESTAAN**"

        ' tällä scriptillä tarkoitus saada viivakoodin TUOTENUMERO
'kenttä aktiiviseksi
        Set IOFIELD = HmiRuntime.Screens("11 - Viivakood-
inluku").ScreenItems("Tuotenumero")
        IOFIELD.Activate

    Else
        SmartTags("BARTuotetarkistus") = False
        ' Jos viivakoodilla saatiin sallittu tuotenumero, etsitään
'tuotenumeron reseptiä
        SmartTags("EtsiNumero") = SmartTags("BARTuotenro")
        SmartTags("BARTuotenimi") = "Etsitään..." "..touhua.."

        'Kutsutaan funktiota, joka lukee kaikki reseptit ja etsii
'oikean reseptin tuotteelle
        Call VB_EtsiResepti

        SmartTags("BARTuotenimi") = SmartTags("Etsitty_Tuotteen_nimi")
        SmartTags("Eränseuranta_Currentstep") = 1
        SmartTags("NextStep") = 1
```

```

        SmartTags("EtsittyReseptinNumeroX") = SmartTags("EtsittyRe-
septinNumero")
        SmartTags("Etsitty_Tuotteen_nimiX") = SmartTags("Et-
sitty_Tuotteen_nimi")

        SmartTags("BARvaihe") = 3
        vaihe = SmartTags("BARvaihe")

    Else
        SmartTags("Erastatus") = 4
        SmartTags("BARTuotenimi") = "***LUE TUOTENUMERO**"
        SmartTags("BARvaihe") = 2

        ' tällä scriptillä tarkoitus saada viivakoodin TUOTENUMERO
'kenttä aktiiviseksi
        Set IOFIELD = HmiRuntime.Screens("11 - Viivakood-
inluku").ScreenItems("Tuotenumero")
        IOFIELD.Activate
        End IF
    End If

End If

'*****Vaihe 3
'Tähän vaiheeseen hypätään suoraan, jos eränumero löydettiin
'luettelosta, muuten tullaan kakkosvaiheen kautta
'Tässä vaiheessa luetaan resepti, jos eränumero löytyi eräluettelosta
'ja näytetään tuotenimi paneelilla

If vaihe = 3 Or (SmartTags("BARvaihe") = 2 And SmartTags("Erastatus")
= 2) Then
    ' resepteistä on etsitty tuotenumeroa vastaava tuotenimi
    ' nyt tarkistetaan sen onnistuminen
    SmartTags("Eränseuranta_Eraloytyi") = False

    If SmartTags("Erastatus") = 2 Then

        SmartTags("EtsittyReseptinNumeroX") = SmartTags("Eränseu-
ranta_ResepetiNro")
        End If

        SmartTags("resreadstatus")=2 ' laitetaan luennan statuksen arvoksi 2
        j=SmartTags("resreadstatus")

        touhuua=0 ' tällä varmistetaan että reseptinluku komento saa
'työrauhan
        Do While j = 2
            If touhuua=0 Then
                LoadDataRecord 2, SmartTags("EtsittyReseptinNu-
meroX"),"resreadstatus" 'j 'resreadstatus
                touhuua = 1
            End If
            j=SmartTags("resreadstatus")
        Loop

    If SmartTags("Erastatus") = 2 Then

```

```
SmartTags("Etsitty_Tuotteen_nimiX") = SmartTags("res_Tuot-
teen_nimi")
SmartTags("EtsittyReseptinNumeroX") = SmartTags("Eränseuranta_Re-
sepetiNro")
SmartTags("BARTuotenro") = SmartTags("res_Tuotteen_numero")

End If

SmartTags("BARTuotenimi") = SmartTags("Etsitty_Tuotteen_nimiX")

If SmartTags("EtsittyReseptinNumeroX") > 0 Then
  'resepti löytyi, näytetään tuotenimi ruudulla ja jatketaan seuraa-
vaan vaiheeseen
  SmartTags("BARvaihe") = 4
  SmartTags("Erastatus") = 3

Else
  SmartTags("BARTuotenimi") = "***LUE TUOTENUMERO**"
  SmartTags("BARvaihe") = 2

End If
End If

'*****VAIHE0
If vaihe = 0 Then
  ' alustetaan näytöllä näytettävät tiedot alkutilaan
  SmartTags("BAReranro") = 0
  SmartTags("BARTuotenro") = 0
  SmartTags("Erastatus") = 0
  SmartTags("EtsittyResepti_valinta") = 0
  SmartTags("EtsittyResepti_LKM") = 0
  SmartTags("BARTuotetarkistus") = False
  SmartTags("BARTuotenimi") = ""
  SmartTags("EtsittyReseptinNumero1") = 0
  SmartTags("BAReratarkistus") = False
  SmartTags("BARvaihe") = 1
  SmartTags("Eränseuranta_Eraaeiloytynyt") = False
  SmartTags("Eränseuranta_Eraloytyi") = False
  SmartTags("Eränseuranta_Currentstep") = 0
  SmartTags("NextStep") = 0
  SmartTags("NextStepHMI") = 0
  SmartTags("Eränseuranta_ResepetiNro") = 0

  'Poistetaan virtuaalinäppäimistön käyttö operaattoritunnuksilla
  If SmartTags("CurrentUserGroup") < 2 Then
    SetScreenKeyboardMode hmiOff

  Else
    SetScreenKeyboardMode hmiOn

  End If

  ' tällä scriptillä tarkoitus saada viivakoodin eränumero kenttä
  'aktiiviseksi
  Set IOFIELD = HmiRuntime.Screens("11 - Viivakood-
inluku").ScreenItems("Eranumero")
  IOFIELD.Activate
End If
```

Eräluettelon käsittelyn toteuttaminen SCL-ohjelmointikielellä TIA Portal -ohjelmistossa

```
//Kun resepti hyväksytään, niin ohjelma lisää erän tiedot luetteloon,
//jos erää ei löydy ennestään sieltä
IF "Eränseuranta".LisaaEraLuetteloon AND "Eränseuranta".Eraaeilloytynyt
AND "DB.ReseptiOtsikko"."ErA(x)".ID.Tuota_batch > 999999 THEN

    //Lisätään eräntiedot paikallismuuttujaan
    #Temp_eratietokanta.Batch := "DB.Resep-
tiOtsikko"."ErA(x)".ID.Tuota_batch;
    #Temp_eratietokanta.TotalMixes := "DB.Resep-
tiOtsikko"."ErA(x)".RES.Askel_lkm;
    #Temp_eratietokanta.CurrentMix := "Eränseuranta".NextStep;
    #Temp_eratietokanta.RecipeNro := "DB.Resep-
tiOtsikko"."ErA(x)".RES.Res_nro;
    #Temp_eratietokanta."Date" := "Eränseuranta".PLCtime;

    //Kutsutaan funktiota, jossa hoidetaan erän lisäys luetteloon ja
//argumenttina annetaan eräntiedot
    "FC.erän lisäys luetteloon" (#Temp_eratietokanta);

    "Eränseuranta".Eraaeilloytynyt := FALSE;
    "Eränseuranta".LisaaEraLuetteloon := FALSE;

END_IF;

IF "DB.SEQ4".#Ajonkesto.Q AND "DB.SEQ1".STEP20 AND "DB.SEQ4".STEP8
THEN
    "Eränseuranta".Apubitti := TRUE;

END_IF;

IF "Eränseuranta".Apubitti AND "DB.SEQ4".STEP9 THEN

    FOR #k := "Eränseuranta_RecipeMemory_ArrayMin" TO
"Eränseuranta_RecipeMemory_ArrayMax" DO

        //Kun kontin sekoittaminen on päätynyt onnistuneesti, niin
//luettelossa currentmix-muuttuja + 1, jolloin seuraavan
        //kerran kun ajetaan samaa erää, niin laite valitsee
//seuraavan sekoitusaskeleen mukaiset reseptiparametrit.
        IF "DB.Eratietokanta".Eratietokanta[#k].Batch = "DB.Resep-
tiOtsikko"."ErA(x)".ID.Tuota_batch THEN
            "DB.Eratietokanta".Eratietokanta[#k].CurrentMix :=
"DB.Eratietokanta".Eratietokanta[#k].CurrentMix + 1;

            "Eränseuranta".PopUp_statuspaattynysekoitus := 1;

            //Kun erän kaikki askeleet on suoritettu, niin erä
//poistetaan luettelosta
            IF "DB.Eratietokanta".Eratietokanta[#k].TotalMixes <
"DB.Eratietokanta".Eratietokanta[#k].CurrentMix THEN
                "Eränseuranta".PopUp_statuspaattynysekoitus := 2;
                "Eränseuranta".PoistaEraLuettelosta := TRUE;
```

```
        //Kutsutaan eränpoistamis funktiota. Argumenttina
//annetaan eränumero, joka halutaan poistaa tietokannasta
        "PoistaEraluetteloista" ("DB.Resep-
tiOtsikko"."ErA(x)".ID.Tuota_batch);
        END_IF;

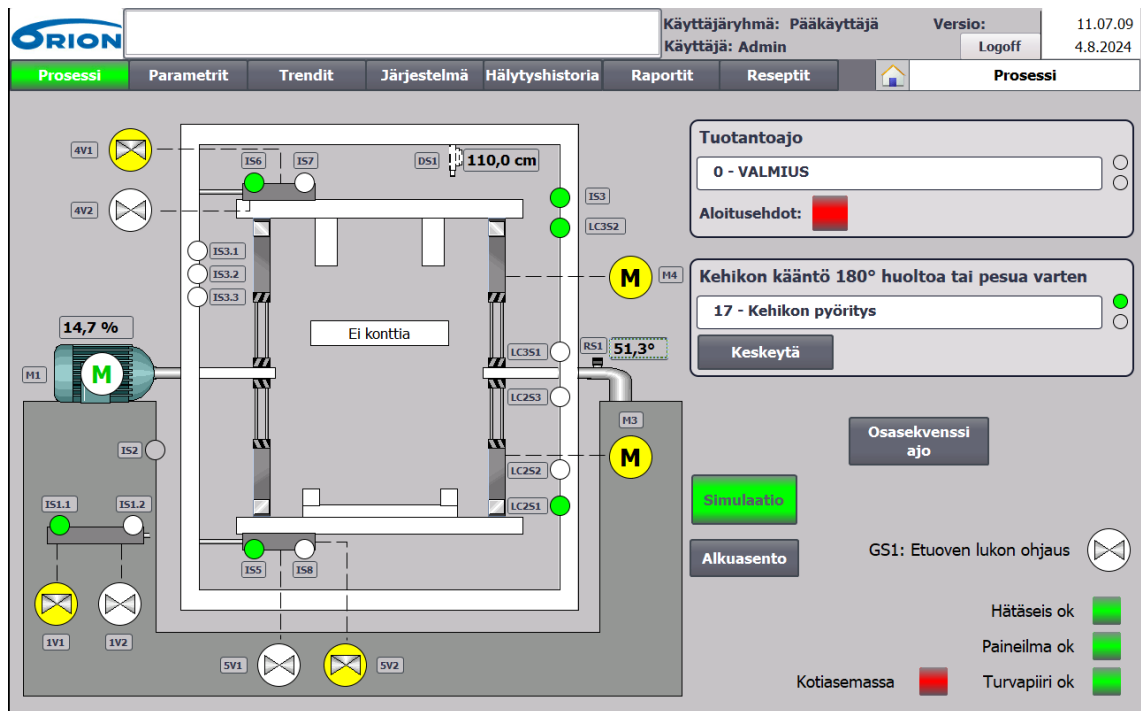
        EXIT;

    END_IF;

END_FOR;

"Eränseuranta".Apubitti := FALSE;
END_IF;
```

Kehikon käännön testaaminen simuloinnilla prosessisivulta seurattuna



Kuva 1. Kehikon käännön testaaminen simulointiympäristössä prosessisivulta seurattuna.