

Iiro Kauppinen

**MANIPULAATTORISOLUN 3D-MALLIN SIMULOINTI JA VIRTU-
AALINEN KÄYTTÖÖNOTTO**

**Opinnäytetyö
CENTRIA-AMMATTIKORKEAKOULU
Tieto- ja viestintäteknikan koulutusohjelma
Maaliskuu 2020**

TIIVISTELMÄ OPINNÄYTETYÖSTÄ

Centria-ammattikorkeakoulu	Aika Maaliskuu 2020	Tekijä/tekijät Iiro Kauppinen
Koulutusohjelma Tieto- ja viestintätekniikka		
Työn nimi MANIPULAATTORISOLUN 3D-MALLIN SIMULOINTI JA VIRTUAALINEN KÄYTTÖÖNOTTO		
Työn ohjaaja Hannu Ala-Pönttiö	Sivumäärä 36 + 0	
Työelämäohjaaja Tuomo Käsäkangas		
<p>Opinnäytetyössä tutkittiin virtuaalisen suunnittelun vaikutusta automaatiolaitteiston käyttöönoton avuksi. Tavoitteena oli saada tehtyä virtuaalinen käyttöönotto manipulaattorisolulle, joka toimi pullojen pakkausoluna. Pulloja ja kennoja pakattiin päällekkäin kerroksittain lavojen päälle. Virtuaalinen käyttöönotto toteutettiin, koska manipulaattorin sijoituskohta oli kriittinen tuotannon kannalta ja pitkää tuotantokatkoa ei haluttu syntyvän. Tästä syystä haluttiin mahdollisimman tarkasti tietää jo suunnittelu- vaiheessa, kuinka manipulaattori toimisi.</p> <p>Manipulaattorisolua ohjattiin Siemensin ohjelmoitavan logiikan avulla. Manipulaattori itsessään sisälsi neljä servomootoria, paineilmasylintereitä sekä anturointeja. Muita solussa olevia koneita olivat lavojen purkausasema, kennojen syöttökuljetinlaitteisto, pullojen syöttöpöytä, lavojen siirtovaunu sekä pakkausaseman kuljetinlaitteisto. Opinnäytetyöhön kuuluivat logiikan ohjelman suunnittelu sekä manipulaattorin 3D-mallinnus virtuaaliseksi malliksi. Virtuaaliseen malliin luotiin fysikaalisia ilmiötä, joiden avulla selvitettiin manipulaattorisolun toimintaa. 3D-mallin suunnittelusta vastasi 3D-suunnittelutoimisto. Mallinnuksessa ja virtuaalisessa käyttöönotossa käytettiin Siemensin NX-ohjelmistoa, SIMIT-simulointityökalua sekä TIA Portal-suunnittelutyökalua. NX-ohjelmistosta hyödynsin Mechatronics Concept Designer -sovellusta. Ohjelmoitavaksi logiikaksi valikoitui Siemensin 1500-sarjan logiikkayksikkö. Logiikan simulointi toteutettiin Siemensin PLCSIM Advanced-simulaattorilla ja rajapinta simulaattorin sekä 3D-mallinnuksen välille luotiin SIMIT-ohjelmalla.</p> <p>Työssä oli tarkoitus selvittää, kuinka hyödyllistä virtuaalinen käyttöönotto on ja millaisiin tuloksiin sillä päästään. Opinnäytetyössä kartoitettiin resurssien tarvetta suunnittelutyössä, eli suunnittelutyön mahdollista kasvua ennen koneen käyttöönottoa sekä simuloinnin vaikutusta todelliseen käyttöönottoaikaan. Näiden avulla selvitettiin, paljonko projektin suunnittelun vaatii aikaa, mitä etuja virtuaalisella käyttöönotolla saavutetaan sekä miten virtuaalista käyttöönottoa saataisiin hyödynnettyä tulevaisuudessa mahdollisimman hyvin.</p>		
Asiasanat 3D-malli, digitaalinen kaksonen, käyttöönotto, manipulaattori, rajapinta, virtuaalinen suunnittelu.		

ABSTRACT

Centria University of Applied Sciences	Date March 2020	Author Iiro Kauppinen
Degree programme Information and communication technology		
Name of thesis 3D MODEL SIMULATION AND VIRTUAL COMMISSIONING OF MANIPULATOR CELL		
Instructor Hannu Ala-Pönttiö	Pages 36 + 0	
Supervisor Tuomo Käsäkangas		
<p>This thesis work researched the benefits of virtual planning in the commissioning of automation machines. The aim of the research was to implement virtual commissioning to a manipulator cell which packed bottles and package cells on the pallets. The bottles and cells were packed in layers on the pallet. The reason to made virtual commissioning to this manipulator cell was that the cell was very critical point of the production flow of this factory and long break of material flow would have caused a dramatic cost to the factory. To avoid the long break possible fail situations were identified by simulating the process before the actual commissioning.</p> <p>The manipulator cell was controlled by Siemens logic controller. The manipulator itself had four servo axis, pressure cylinders and sensors. Other machines of the cell were pallets unloading station, packing cells feeding station, bottles feeding desk, pallets transport machine and packing stations conveyor equipment. The thesis included programming of the machines and virtual commissioning of the cell. Virtual commissioning included linking physical attributes to the 3D model to create virtual model. The 3D model was created by 3D designing office.</p> <p>We used Siemens NX software platform and Mechatronics Concept Designer application was used to model the physical phenomena of the digital twin. Siemens 1500 series logic unit was selected for the logical unit. The simulation of the logic was handled by Siemens PLCSIM Advanced simulator. The interface between NX software and PLC simulator was created by SIMIT which is a simulation platform tool, created by Siemens.</p> <p>The purpose of the thesis was to discover how useful virtual commissioning is and what kind of results it will bring. This thesis also examined the need of resources to make a digital twin and the possible increase of designing work to create the simulation environment. Also, it was investigated how virtual commissioning will ease the real commissioning situation. With these results it would be possible to utilize virtual commissioning in the future.</p>		

<p>Key words 3D model, digital twin, commissioning, interface, manipulator, virtual designing.</p>

KÄSITTEIDEN MÄÄRITTELY

3D-malli – Kolmiulotteinen mallinnus.

Hardware – Automaatiosuunnittelussa hardwarella viitataan automaatiojärjestelmässä käytettyihin automaatiolaitteisiin.

HMI – Human Machine Interface, käyttöliittymä koneen ja ihmisen välillä.

I/O – Inputs / Outputs, kuvaa informaation siirtoon liittyvää viestintää tietoteknistenlaitteiden välillä.

MCD – Mechatronics Concept Designer, sovellusmoduuli NX-ohjelmaan, jolla voidaan mallintaa fyysikaalisia ilmiöitä 3D-malliin.

NX – Siemensin valmistama PLM-suunnitteluohjelmisto.

OPC UA – Open Platform Communication Unified Architecture, teollisuuden automaatoratkaisuihin käytettävä standardisoitu kommunikointirajapinta.

PLC – Programmable Logic Controller, ohjelmoitava logiikka.

PLM – Product Lifecycle Management, tuotteen elinkaaren hallinta.

PROFIBUS – Teollisuuteen tarkoitettu standardisoitu kenttäväylä.

PROFINET – Teollisuuteen tarkoitettu standardisoitu teollisuuden ethernet-tekniikka.

PROFIsafe – PROFINET:n ja PROFIBUS:n rinnalla toimiva turvallistamiseen standardisoitu tekniikka.

Shared Memory – Ohjelmien kesken jaettua muistia.

TIIVISTELMÄ
ABSTRACT
KÄSITTEIDEN MÄÄRITTELY
SISÄLLYS

1 JOHDANTO	1
2 VIRTUAALINEN SUUNNITTELU	3
3 3D-MALLIN SIMULOINTI	6
4 NX-OHJELMISTO SEKÄ MCD-SOVELLUKSEN KÄYTTÖ	9
4.1 Lisenssipalvelu	9
4.2 Ohjelman ominaisuudet	10
4.3 MCD-objektien ominaisuuksien määrittäminen	13
4.4 MCD-signaalien konfigurointi	16
4.5 SIMIT-ohjelmisto	17
5 PROJEKTIN VAIHEET	19
5.1 Projektin määrittely	21
5.2 Projektin suunnittelu	22
5.3 Simulointimallin toteutus	24
5.4 Virtuaalinen käyttöönotto	26
5.5 Optimointi	29
6 YHTEENVETO	32
LÄHTEET	35
LIITTEET	
KUVAT	
KUVA 1. Virtuaalisen suunnittelun eri vaiheet PLM-järjestelmässä	3
KUVA 2. Virtuaalisen suunnittelun tuoma etu projektissa	5
KUVA 3. Kustannukset ajan funktiona projektin eri vaiheissa	7
KUVA 4. SPLM lisenssi serverin toiminta	10
KUVA 5. MCD:n vasen valikko	12
KUVA 6. MCD:n yläpalkin vasenosa	12
KUVA 7. MCD:n yläpalkin oikeaosa	13
KUVA 8. Rigid Body -työkalu	13
KUVA 9. Position Control -työkalu	14
KUVA 10. MCD:n Gantt-kaavio	16
KUVA 11. SIMIT, telegram 1 määrittely	18
KUVA 12. Automaatiosuunnittelun elinkaarimalli	19
KUVA 13. Automaatiosuunnittelun tasot	20
KUVA 14 S7-GRAPH-ohjelmointikieli	23
KUVA 15. SCL-ohjelmointikieli	24
KUVA 16. Manipulaattorisolun 3D-malli	25
KUVA 17. Koneturvallisuuden liittyviä turvallisuusstandardeja	26
KUVA 18. Ohjelmiston turvallisuuselinkaaren yksinkertaistettu V-malli	29
KUVA 19. Manipulaattorin parametointiin tarkoitettu ohjausikkuna	30

KUVA 20. Manipulaattorin kehittynyt ohjausikkuna.....	31
---	----

TAULUKOT

TAULUKKO 1. Virheiden korjauskustannukset projektin eri vaiheissa	7
TAULUKKO 2. Ulkoisten signaaleiden protokollat ja käytettävät datatyypit	17

1 JOHDANTO

Miten virtuaalista suunnittelua voidaan hyödyntää automaatiolaitteen käyttöönottilanteessa? Saadanko käyttöönottoaikaa lyhennettyä asiakkaan luona ja paljoko suunnittelutyön määrä muuttuu suunnitteluvaiheessa? Virtuaalisen käyttöönoton perusideana pidetään virtuaalisen tehtaan tai sen osan liittämistä todelliseen ohjaimeen, kuten PLC (Programmable Logic Controller) tai HMI (Human Machine Interface) laitteeseen. Tällöin suunnittelijat eri tiimeistä voivat työskennellä yhdessä ja kehittää laitteistoa. Laitteiston toiminta voidaan varmentaa jo etukäteen virtuaalisesti, ja tämä parantaa projektin etenemistä verrattuna tavalliseen suunnittelutilanteeseen. Virtuaalisen käyttöönoton avulla havaitaan virheet aikaisemmin ja niistä keskustelu on helpompaa. (Liu, Suchold, & Diedrich 2012.)

Teknologia kehittyy jatkuvasti, ja viime aikoina kehitys on tuntunut nopealta. Teknologian kehitys on tuonut esiin virtuaalisen ympäristön, jota yhä enemmän käytetään suunnittelun tukena. Tämä on tarjonnut mahdollisuuden ajatella tuotteiden kehittämistä uudella näkökulmalla. Tänä päivänä kun puhutaan tuoteprosesseista, puhutaan koko tuotteen elinkaaren tarvekartoituksesta uudelleen kierrätykseen. Tuotetiedon ja tuotteen elinkaaren hallintajärjestelmiä kutsutaan PLM-järjestelmiksi. (Ahola ym. 2011.)

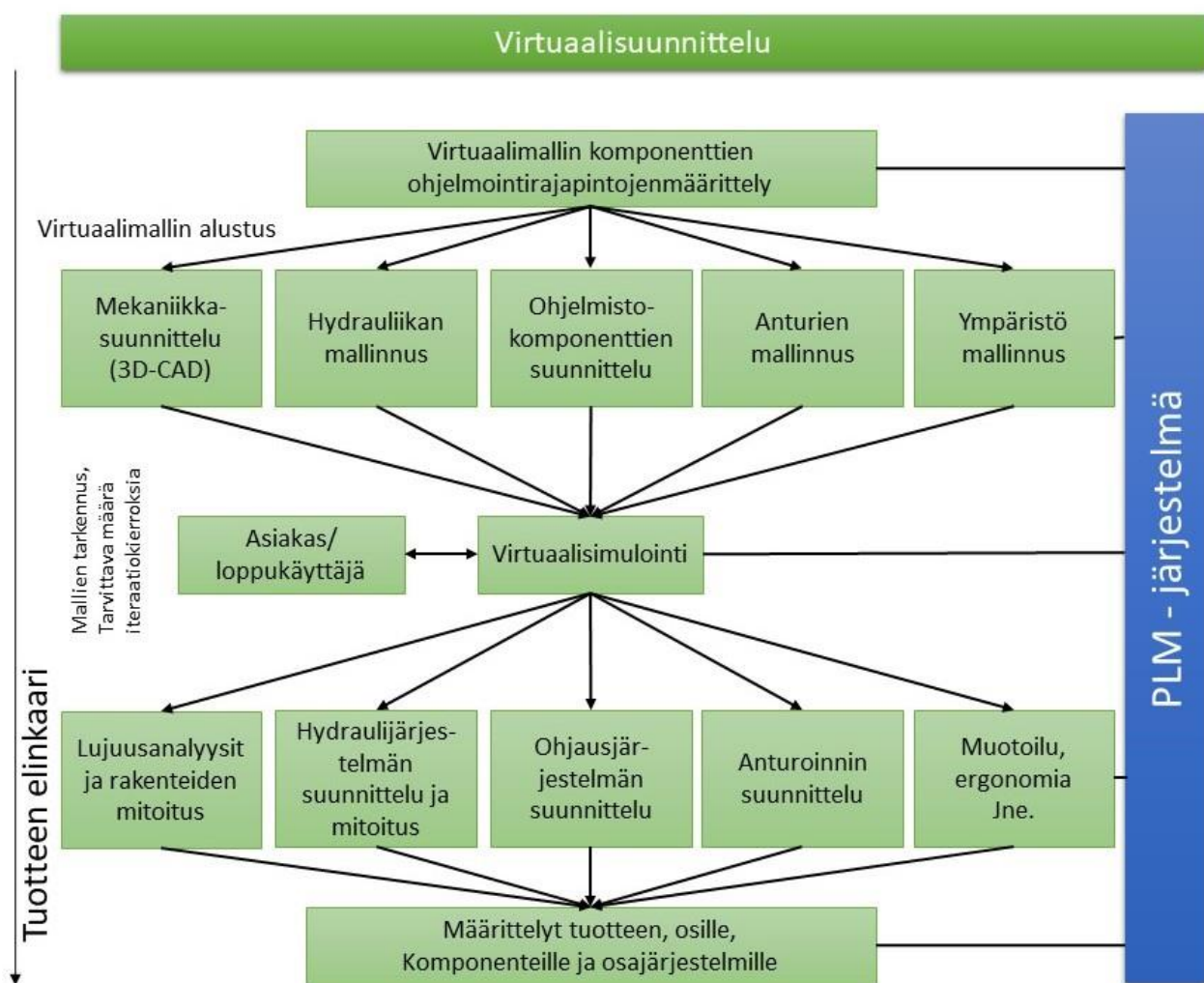
PLM-järjestelmät eli tuotteen elinkaaren hallintajärjestelmät liittyvät virtuaalisuunnitteluun läheisesti. Nykyään järjestelmiin integroidaan useita sovelluksia, jolloin tuoteprosesseja voidaan tutkia monesta eri näkökulmasta. Tällä pyritään tuotteiden hallintaan luonnosvaiheesta aina tuotteiden kehitys- sekä logistiikkaratkaisuihin saakka. (Silvennoinen 2016.)

Kuinka PLM-järjestelmät näkyvät NX-ohjelmistossa? NX-suunnitteluohjelmisto sisältää useita tuotteen suunnitteluun tarkoitettuja sovelluksia. Tuotetta voidaan suunnitella mekaanisen, sähköistyksen, automatisoinnin sekä prosessin näkökulmasta. Tulevaisuudessa pyritään siihen, että suunnittelua voitaisiin tehdä usean eri vaiheen kanssa päällekkäin virtuaalimallin avulla, jolloin tuoteprosessin kehittämisvaiheen ajankäyttö saataisiin maksimoitua. Kun tuotetta voidaan tutkia virtuaalimallissa kehityksen aikana, saadaan osa käyttöönottoajasta siirrettyä tuotteen kehitysvaiheeseen. Tällöin saataisiin säästettyä aikaa käyttöönottilanteesta tai tuotteen markkinoille viemisestä, jotka yleensä ovat kriittisiä asiakkaan kannalta. (Merhof 2018.)

Opinnäytetyö toteutettiin Apex Automationin toiveesta tutustua virtuaaliseen mallintamiseen sekä simuloinnin kehittämiseen. Työllä tutkitaan mahdollista hyötyä virtuaalisen käyttöönoton vaikutuksesta automaatioprojekteille. Apex Automation on Kokkolassa ja Vaasassa sijaitseva automaatioalan insinööri-toimisto, joka tuottaa suunnittelu- ja asiantuntijapalveluita asiakkailleen (Apex Automation Oy 2020).

2 VIRTUAALINEN SUUNNITTELU

Mitä virtuaalisuunnittelulla tarkoitetaan? Virtuaalisuunnittelulla mallinnetaan teknisiä järjestelmiä tietokoneohjelmistojen avulla. Ohjelmistoilla luodaan fysikaalisia ominaisuuksia järjestelmiin, jotta ilmiöitä voidaan tutkia. Mallista pyritään havaitsemaan suunnitteluvirheitä toistamalla simulointeja useita kertoja. Olisi hyvä, että suunnittelulle olisi asetettu tavoite, jotta ongelmien määrittäminen voitaisiin keskittää mahdollisimman tarkasti. Kuvassa 1, jossa esitetään virtuaalisen suunnittelun vaiheet PLM-järjestelmässä. (Ahola ym. 2011.)



KUVA 1. Virtuaalisen suunnittelun eri vaiheet PLM-järjestelmässä. (mukaillen Ahola ym. 2011)

Virtuaalisuunnittelulla pyritään nostamaan suunnittelun laatua sekä nopeuttamaan tuotekehitysprosessia ja karttamaan kustannuksista aiheutuvia riskejä. Jotta edellä mainittuihin tavoitteisiin päästäisiin, on tärkeää, että kaikki tuotekehitysprosessissa olevat sidosryhmät (asiakas, loppukäyttäjä, myynti, tuotanto

jne.) ja suunnittelutiimit panostaisivat virtuaalisuunnitteluun jo projektin alkuvaiheesta saakka. Kolmiulotteinen prototyyppi luo mielenkiintoa ja keskustelua eri sidosryhmien välille. Siksi olisi tärkeää, että sitä esitettäisiin koko tuotekehitysorganisaation kesken eikä ainoastaan simulointiosastolle. Tällöin havaittaisiin tarkemmin myös eri puutteita, kun prototyyppi olisi laajemmin esillä eri tuotekehitysaluilla. (Ahola, ym. 2011.)

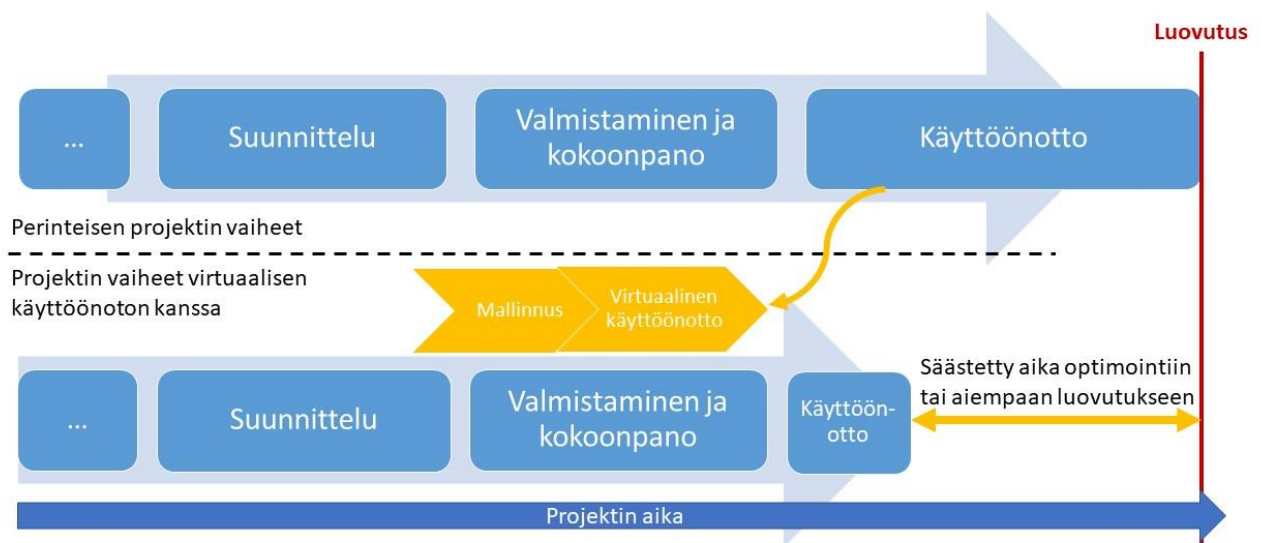
Visual Components on automaatio suunnitteluun erikoistunut yritys, ja sen mukaan käyttöönottilanne vie yleensä koko projektin ajasta noin 15–20 %. Yleensä käyttöönotossa jopa kaksi kolmasosaa ajasta tuhlautuu ohjelmavirheiden korjaamiseen, koska ohjelma päästään testaamaan kunnolla vasta kun kone on saatu mekaanisesti asennettua ja koottua. Tämä johtaa siihen, että käyttöönottilanteessa joudutaan usein työskentelemään kovan paineen alla, kun projekti pitäisi saada mahdollisimman nopeasti valmiiksi. Tästä seuraa yleensä suurempi riski tehdä virheitä. Jos käyttöönottoa voitaisiin tehdä jo suunnitteluvaiheessa, saataisiin kriittistä aikaa vähennettyä käyttöönottilanteesta. Tällöin voitaisiin keskittyä käyttöönottilanteessa enemmän laitteen optimointiin kuin vikojen paikantamiseen. (Visual Components 2016.)

Virtuaalinen käyttöönotto mahdollistaa osan käyttöönotosta siirtämisen jo suunnitteluvaiheeseen. Tämä tapahtuu simuloimalla 3D-mallia todellisen koneen tai prosessin sijasta. Virtuaalista 3D-mallia ohjataan todellisella ohjainyksiköllä, jota käytetään projektissa todellisuudessakin. Tämä mahdollistaa virheiden tunnistamisen jo aikaisemmassa vaiheessa. On myös mahdollista käyttää niin sanottua Hybrid commissioning -menetelmää, jossa todellisen koneen rinnalle tehdään virtuaalinen mallinnus. Tämän avulla voidaan muutokset luoda ensin simuloimalla ja siirtää sen jälkeen vaiheittain todelliseen maailmaan. (Visual Components 2016.)

Virtuaalisella käyttöönotolla lisätään myös ohjelmoijan tietoisuutta laitteistosta mallinnuksen avulla. Yleensä ohjelmoijat saavat ainoastaan toimintakuvauksen laitteistosta ja tekevät sen perusteella ohjelman laitteistolle. Mallinnuksen avulla ohjelmoija näkee laitteiston todellisuudessa, ja se avartaa laitteiston toimintaa huomattavasti enemmän. Tällöin voidaan huomata kriittisiä asioita, joita muuten olisi hankalampi havainnoida. Tämä lisää siten ohjelman laatua ja parantaa ohjelman turvallisuutta. (Visual Components 2016.)

Mallinnuksen tekemisessä on myös muita hyötyjä. Sillä saadaan luotua huomattavasti tarkemman näköinen kuva projektista muille projektissa työskenteleville tiimeille kuin dokumenttien pohjalta esitetyllä mallilla. Tämä johtaa parempien saavutusten toteutumisiin, laitteiston käyttäytymisen hahmottamiseen

ja informaation laatuun asiakkaalle. Kuvassa 2 vertaillaan perinteisen suunnittelun ja virtuaalisen suunnittelun tuomaa etua. (Visual Components 2016.)



KUVA 2. Virtuaalisen suunnittelun tuoma etu projektissa. (mukaiillen Visual Components 2016)

3 3D-MALLIN SIMULOINTI

Virtuaalista suunnittelua hyödyntävässä suunnitteluprosessissa, jossa tarkoituksena on simuloida mallinnus, on mallinuksen oltava mukana tuotekehityksen alusta saakka. Tavoitteena on, että mallia kehitettäisiin projektin edetessä kohti toteutettavissa olevaa kokonaisuutta. Malli koostuu useista eri komponenteista, joista vastaavat yleensä eri osa-alueet. Mallinnusta tehdessä on suositeltavaa, että kokonaisuus kasattaisiin osakokonaisuuksista, jotta suunnittelu sujuisi joustavammin ja niiden hallinta olisi yksinkertaisempaa kuin yhden suuren kokonaisuuden. Tällaista mallinnustapaa kutsutaan modulaarisiksi mallintamiseksi. Modulaarisen mallintamisen avulla kyetään myös tarvittaessa hankkimaan osamallinuksia yrityksen ulkopuolisilta tekijöiltä. PLM-järjestelmistä voidaan hallita osamallinuksia sekä niiden tiedonkäsittelyä. (Ahola ym. 2011.)

Osamallien yhteensopivuus eri rajapintojen välillä tulee varmistaa ennen niiden toteuttamista. Ennen simuloinnin aloittamista tulee osamallit yhdistää yhdeksi kokonaisuudeksi, johon liitytään ohjausjärjestelmällä. Tuotetta suunnitellessa kannattaa harkita, tekeekö kaiken itse vai hankkiiko osan palveluista alihankintana. (Ahola ym. 2011.)

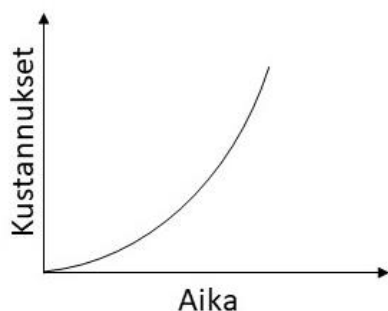
3D-mallia voidaan virtuaalisen suunnittelun ohella kutsua digitaalseksi kaksosiksi (Digital Twin). Tällä tarkoitetaan tuotetta, jolla on olemassa todellinen malli reaali maailmassa ja virtuaalinen malli virtuaalitodellisuudessa. Virtuaalisen ja todellisen mallin välillä tulee olla jonkin asteen yhteys, jolla informaatiota siirretään niiden välillä. Tavoitteena on, että kaksosen avulla voitaisiin työskennellä molempien mallien kanssa yhtäaikaisesti. Digitaalisen kaksosen avulla saavutetaan nopeampi vasteaika muutoksille ja sitä voidaan muokata etäältä, esimerkiksi toiselta puolelta maailmaa. Tämä mahdollistaa globaalien tuen projektille eikä sido kaikkia projektiin kuuluvia lähelle projektin kehittämispaikkaa. Kaksosen avulla voidaan luoda mallille raja-arvoja, joita voidaan virtuaali maailmassa testata välittömästi parametrejä muuntamalla. Parametrejä voivat olla esimerkiksi liikkeen pituus tai nopeus. (Rosen, Von Wichert, Lo & Bettenhausen 2015, 567–572.)

Simulointia on syytä tarkastella myös kustannusten näkökulmasta. Hamblingin, Morganin, Samaroon, Thompsonin sekä Williamsin teoksessa (2007, 7-27) Morgan on tutkinut projektille aiheutuvia kustannuksia projektin eri vaiheissa. Hän on taulukoinut kustannukset seuraavasti.

TAULUKKO 1. Virheiden korjauskustannukset projektin eri vaiheissa. (mukaillen Hambling ym. 2007, 7-27.)

Projektin vaihe, jolloin vika tunnistetaan	Suhteellinen kustannus
Projektin määrittely	1 USD
Ohjelmointi	10 USD
Ohjelman testaus	100 USD
Järjestelmän testaus	1 000 USD
Hyväksyntätestaus	10 000 USD
Järjestelmä käytössä	100 000 USD

Kustannukset voidaan kuvata myös kuvan 3 mukaan. Kuvasta huomataan, että mitä pidemmälle projekti on edennyt, sitä enemmän muutoksista aiheutuu kustannuksia projektille.



KUVA 3. Kustannukset ajan funktiona projektin eri vaiheissa. (mukaillen Hambling ym. 2007, 7-27.)

Tästä syystä Morganin mukaan olisi erittäin tärkeää pystyä testaamaan tuote jo hyvissä ajoin. Mitä myöhemmin testaus aloitetaan, sitä vähemmäksi jäljellä oleva aika jää. Vikojen paikantaminen on yksi testaamisen tarkoituksista. Testaamisella tulisi pyrkiä etsimään mahdollisimman paljon virheitä, jotta viat paikannettaisiin aikaisessa vaiheessa. Testaaminen tulisi myös lähtökohtaisesti keskittää sovelluksen päätehtävään. Esimerkkinä voidaan pitää Morganin esittämää verkkokaupan nettisivujen suunnittelun testaus. Sivustoja kehittäessä olisi tärkeää tarkastaa salasanojen toimivuus sekä maksujen suorittamiseen liittyvät asiat. Voisiko esimerkiksi maksun suorittaa väärennetyllä luottokortilla. (Hambling ym. 2007, 7-27.)

Ohjelmoitavaa logiikkaa simuloidessa tulisi ottaa huomioon, ettei suunnitteluvaiheessa muokkaisi ohjelmaa simulointia varten vaan ohjelman muokkaus simulointia varten tulisi tehdä erillisellä systeemillä esimerkiksi erillisen sovelluksen avulla. Tällöin olisi varauduttu, ettei käyttöönottilanteessa ohjelmaan

ole jäänyt simulointia varten tehtyjä muutoksia. Pahimmissa tapauksessa on simulointia varten ohitettu turvaohjelman ehtoja ja ohitukset ovat voimassa vielä käyttöönottotilanteessa. Tämä voisi aiheuttaa huomattavan turvallisuusrisikin. Siemensiltä löytyy muun muassa SIMIT-työkalu, johon voidaan liittää esimerkiksi taajuusmuuttajien telegrammeja. Tuolloin simuloitu logiikka ymmärtää, että siihen on kytketty taajuusmuuttaja ja itse ohjelmaan ei tarvitse tehdä erillistä laskentaa tai määrittelyä simulointia varten. (DI-VIRTCOM 2018.)

Siemensin PLCSIM Advanced-simulaattoriin on mahdollista liittää erilaisia apusovelluksia. Tämänkaltaisen sovellus on muun muassa Crosslink-sovellus. Se on vapaasti ladattava Siemensin ohjelma, jolla voidaan yhdistää kaksi PLCSIM Advanced-instanssia keskenään siten, että ohjelma kääntää logiikoiden I/O:n ristiin. Tällä tarkoitetaan, että inputit kääntyvät toisen logiikan outputteihin ja vastaavasti outputit kääntyvät toisen logiikan inputteihin. Näin voidaan luoda simulointiympäristö toiseen logiikkaan ja toisessa logiikassa voidaan pitää itse projektin ohjelma koskemattomana. (SIMATIC S7 PLCSIM Advanced: Crosslink API 2018.)

PLCSIM Advanced-simulaattoriin on myös mahdollista kehittää itse omia simulointisovelluksia käyttäen Microsoftin Visual studio -ohjelmointityökalua. Siemensin sivuilta on mahdollista ladata esimerkkiprojekti, jossa esitellään, kuinka valmiiksi luotu simulointiympäristö toimii. Valmiin mallin pohjalta on helpompi lähteä kehittämään omia sovelluksia simulointikäyttöön. (SIMATIC S7-PLCSIM Advanced: Co-Simulation via API 2017.)

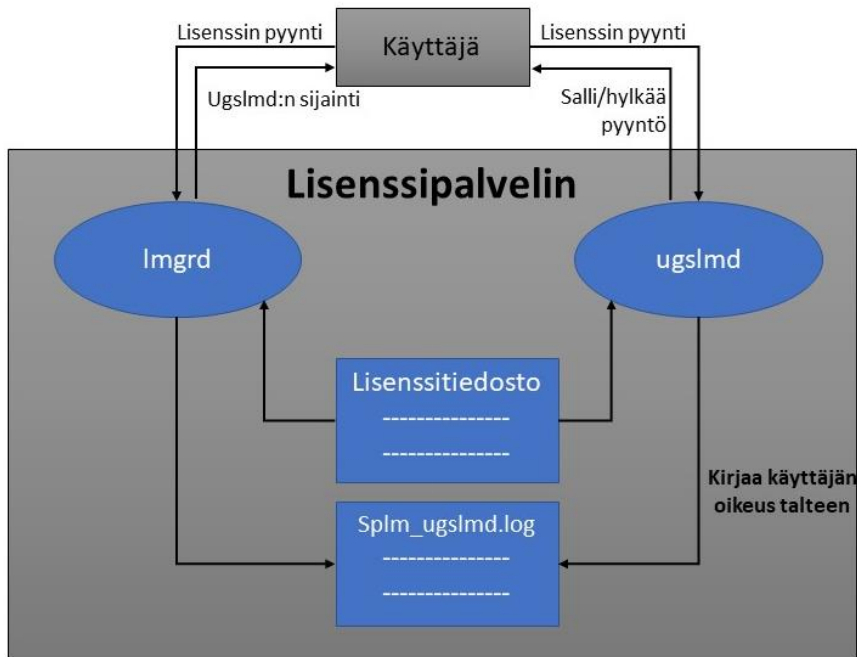
4 NX-OHJELMISTO SEKÄ MCD-SOVELLUKSEN KÄYTTÖ

NX on Siemensin valmistama suunnitteluun ja laskentaan liittyvä ohjelmisto. Ohjelmistolla pyritään saavuttamaan keskiarvoltaan noin 30 % etu tuotteiden, koneiden sekä tuotantolinjojen luovutusajassa. (NG 2019.) Se voidaan jaotella kolmeen eri pääryhmään: suunnitteluun, simulointiin sekä valmistukseen. Suunnitteluosio sisältää tuotteen konseptoinnin, dokumentoinnin ja mallintamisen. Simuloinnilla voidaan testata mallin rakennetta kokoonpanossa ja varmistaa osien yhteensopivuus. Simuloinnilla voidaan laskea myös kappaleiden lujuuslaskentaa sekä virtaus- ja mekaanisia liikelaskelmia. Valmistusosio sisältää laaduntarkastuksen ja koneistus- ja työstöohjaukseen liittyvät työkalut. NX-ohjelma kykenee avaamaan myös muilla CAD-sovelluksilla tehtyjä 3D-malleja. (NX 2020.)

MCD on sovellus, joka on tarkoitettu 3D-mallien simulointiin. MCD-ohjelmiston avulla testaaminen voidaan aloittaa hyvin aikaisessa vaiheessa suunnittelua. Sen avulla kyetään suorittamaan suunnittelu- vaiheessa oleville koneille optimointeja. NX-ohjelmisto sisältää itsessään useita eri sovellusmoduuleita. Sen avulla voidaan luoda 3D-malleja, sähkömekaanista suunnittelua, putkistoja, erilaisia analyysejä jne. NX-ohjelmistoa käytetään muun muassa opetuskäytössä, autoteollisuudessa, kuluttajatuotteissa ja konepajateollisuudessa. (Ideal PLM 2020 & NX 2020.)

4.1 Lisenssipalvelu

Siemensin NX-ohjelmisto vaatii toimiakseen ohjelmistolisenssin. Lisenssejä toimittaa Suomessa Ideal PLM. Lisenssejä varten tulee asentaa lisenssimanageri, jolla lisenssi aktivoidaan. Samaa lisenssimanageria voidaan käyttää koko NX-ohjelmiston sekä Teamcenterin ja Tecnomatixen ohjelmistojen lisenssien aktivointiin. Lisenssejä on kahdenlaisia, kelluvia (floating) ja MAC-osoitteeseen sidottuja (node locked) NX- ja TX-lisenssejä. USB-Donglen kanssa käytettävä kiinteä lisenssi ei vaadi lisenssipalvelua. Lisenssimanageri voidaan asentaa tietokoneelle, jossa on NX:lle tuettu käyttöjärjestelmä. Lisenssimanageri asennetaan yleensä serverille, josta verkon kautta käyttäjät käyttävät lisenssejä. Lisenssit ovat sidottuina laitteen Composite-ID-lukuun (CID). Lisenssejä hallinnoidaan Siemens PLM software licensing toolsin sekä LMToolsin avulla. Lisenssiserverin toiminta on kuvattu kuvassa 4. Kuvassa lisenssiserverille on aktivoitu lisenssitiedosto, jota käyttäjä kyselee omalta tietokoneeltaan. (Siemens PLM 2016.)



KUVA 4. SPLM-lisenssiserverin toiminta. (mukaiillen Siemens PLM 2016)

4.2 Ohjelman ominaisuudet

MCD-sovelluksen ideana on luoda virtuaalinen maailma. Virtuaaliseen maailmaan luodaan realistiset fysiikat. 3D-mallia ohjataan PLC:n tai MCD:ssä määritettyjen operointien avulla ja testataan mallin toimintaa. Ohjelman avulla voidaan suorittaa optimointeja simulointimallille. MCD-sovellusta voidaan näin ollen käyttää digitaalisen kaksosen tekoon, sillä ohjelmistoon voidaan liittää reaali maailmassa olevan koneen toiminnot. Simulointikokoonpanoa, jossa esiintyy jokin todellinen laite, kutsutaan Hardware in the loop nimellä, lyhennettynä HiL. Tämä tarkoittaa, että virtuaaliseen mallintamiseen on liitettyä fyysinen ohjain, esimerkiksi oikea logiikka. Mikäli kaikki testauksessa olevat komponentit ovat virtuaalisia, käytetään termiä Software in the loop (SiL). Ohjelmiston rajapintana voidaan käyttää useita eri rajapintoja. Yleisimmät rajapinnat ovat OPC UA, Shared Memory, Matlab sekä Siemensin PLCSIM Advanced. (DI-VIRTCOM 2018.)

MCD-sovellus käyttää toimiakseen Nvidia PhysX -fysiikkamoottoria. Fysiikkamoottorin avulla mahdollistetaan kappaleiden fyysisten ominaisuuksien toiminta. Tämän avulla kyetään muokkaamaan simuloinnin nopeutta. Hidastamalla nopeutta voidaan tarkemmin kiinnittää huomiota laitteen toimintaan tai nopeuttamalla voidaan lyhyemmässä ajassa tehdä muun muassa kapasiteettitestejä, jolloin säästetään huomattava määrä aikaa reaali maailmaan nähden. (NX 2020 & NX help 2019.) Seuraavaksi käydään läpi MCD-sovelluksen eri työkaluja sekä sovellusnäkyä.

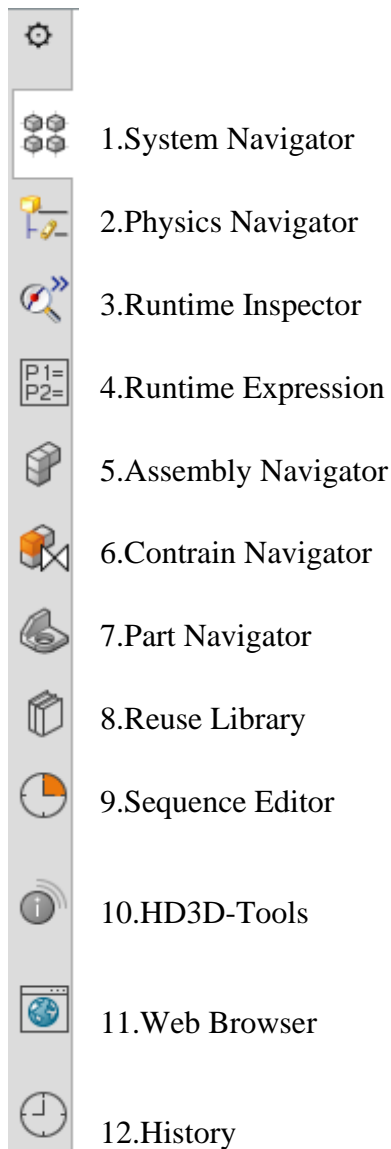
Kuvassa ylhäältä löytyy erilaisia työkaluja mallintamiseen (KUVA 6 & KUVA 7). Työkalut on jaoteltu ryhmittäin eri välilehdille sekä osastoihin. Vasemmasta laidasta löytyy valikko, johon on tuotu seuraavia toimintoja:

1. System Navigator. Työkalulla on tarkoitus tuoda esille suunnittelussa olevan laitteiston vaatimukset, käytännöllisyys ja looginen toiminta. Mikäli käytössä on Teamcenter, voidaan kyseiset projektin tiedot tuoda sieltä NX-sovellukseen. System Navigatoriin voidaan tallentaa tiedostot .XML-tiedostoiksi. System Navigatorista on hyvä seurata, tuleeko vaatimukset suoritettua ja onko toiminta haluttu.
2. Physics Navigator. Tarkoituksena on luoda mallille realistiset fysikaaliset ominaisuudet. Muodostetaan objekteista kiinteitä kappaleita, joilla on massa, sekä luodaan liikkeitä ja liitoksia kappaleiden välille. Signaalien käsittely tapahtuu myös tällä työkalulla.
3. Runtime Inspector. Käytetään diagnostiikkatyökaluna, mikäli halutaan seurata jonkin tietyn kappaleen ominaisuuksia. Voidaan seurata kaikkia kappaleeseen kohdistuvia parametrejä yhtäaikaaisesti. Hyvä työkalu ongelmien ja vikatilojen selvittämiseen. Työkalulla voidaan myös luoda snapshot, jolloin mallin simuloinnin käynnistyminen alkaa siitä ajanhetkestä, kun snapshot on luotu. Muita ominaisuuksia ovat trendin piirto sekä simuloinnin tallennus.
4. Runtime Expression -työkalun avulla voidaan luoda laskennallisia kaavoja ja määritellä niiden avulla siteitä eri parametrien välille. Kytetään muokkaamaan parametrejä esimerkiksi skaalamaan jokin arvo haluttuun muotoon.
5. Assembly Navigator. Täällä voidaan muokata kokoonpanoa sekä valita eri kokonaisuuksia aktiiviseksi. Ennen fysiikoiden mallintamista on hyvä luoda 3D-mallille kokoonpano, joka vastaa liikkeitä.
6. Constraint Navigator. Tähän tulevat esille kaikki mallissa olevat sidokset kappaleiden välillä. Sidokset pitävät kappaleet yhdistettynä toisiinsa sidospinnan suhteen.
7. Part Navigator. Sisältää kaikki yhden kappaleen ominaisuudet. Siellä voidaan muuttaa kappaleen ominaisuuksia sekä tarkastella kappaletta tai mallia eri kuvakulmista.
8. Reuse Library on työkalu, jolla voidaan luoda objektikirjastoja. Kirjastoihin tallennetaan tehtyjä kappaleita, joita voidaan myöhemmin käyttää Reuse Library -toiminnon avulla. Tämä on hyvä työkalu, jos samoja kappaleita käytetään usein.
9. Sequence Editorilla voidaan luoda sekvenssi, johon voidaan liittää mallin eri liikkeitä. Tällöin voidaan simuloida mallia ilman ulkoisia ohjauksia. Sekvenssi luodaan operaatioiden avulla. Operaatioihin liitetään objekti sekä toiminto. Operaatiota voidaan käyttää myös kappaleeseen tarttumisen suorittamiseksi.

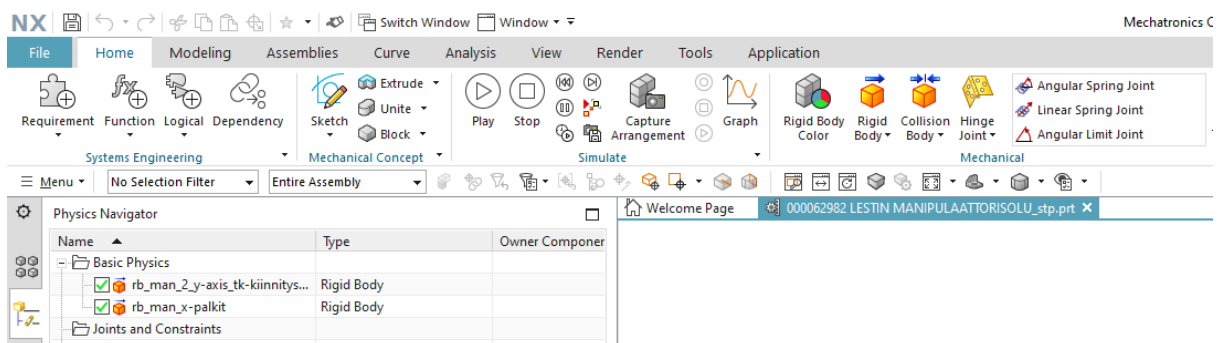
10. HD3D-Tools -työkalun avulla voidaan luoda visuaalista informaatiota 3D-malliin.

11. Web Browser. Internet selain. Kotisivuna toimii NX help, josta löytyy apuja mallinukseen.

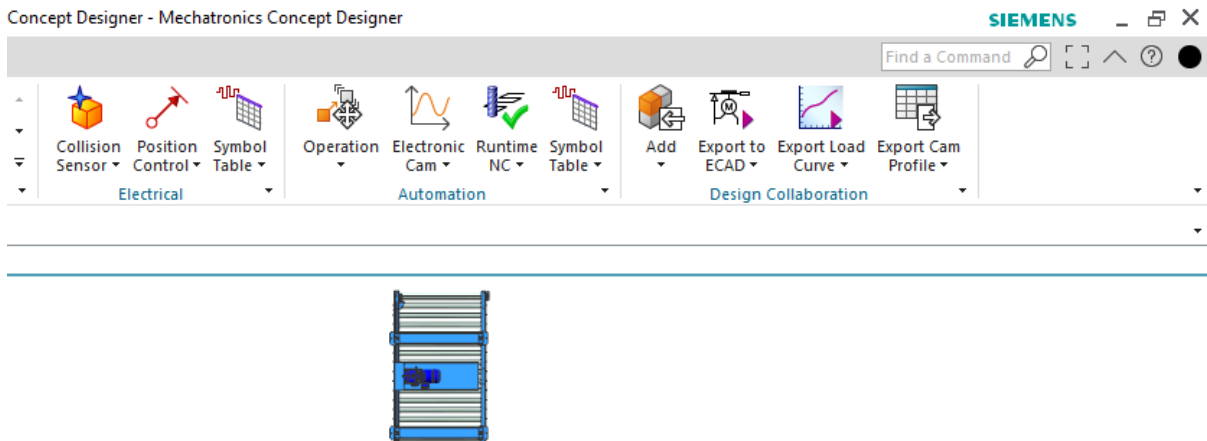
12. History. Voidaan tarkastella, mitä aikaisemmin on tehty.



KUVA 5. MCD:n vasen valikko.



KUVA 6. MCD:n yläpalkin vasen osa.



KUVA 7. MCD:n yläpalkin oikea osa.

4.3 MCD-objektien ominaisuuksien määrittäminen

MCD-sovelluksessa fysiikat luodaan käyttämällä erilaisia työkaluja. Tärkein työkalu fysiikoiden luomiseen on Rigid Body -työkalu. Sen avulla luodaan kappaleelle massa ja määritetään kappaleeseen kohdistuvat voimat. Rigid Body -työkalua käytetään kaikkiin kiinteisiin kappaleisiin, joita halutaan liikuttaa. Mikäli kappale määritetään Rigid Body -työkalulla reagoi se maanvetovoimaan ja putoaa alaspäin. Tämä voidaan estää esimerkiksi tekemällä liitos kappaleelle. (NX help 2019.)



KUVA 8. Rigid Body -työkalu.

Liitoksia on olemassa useita erilaisia. Liitosten määrittely riippuu halutun liikkeen määrittämisestä. Jos kappale halutaan kiinnittää toiseen kappaleeseen kiinteästi, käytetään Fixed Joint -työkalua. Tällä työkalulla voidaan kiinnittää kappaleita ilmaan tai tehdä ns. lattia. Mikäli mallissa halutaan yhdistellä muun muassa hammasrattaita akseleihin, tehdään ne myös Fixed Joint -työkalulla. Muita liitoksia ovat esimerkiksi Hinge Joint, Sliding Joint ja Ball Joint. MCD:stä ei tällä hetkellä löydy työkalua kappaleen tarttumista varten. Tarttuminen voidaan kuitenkin toteuttaa käyttämällä Fixed Joint -työkalua sekä luomalla operointi tarttumiseen sekä irrotukseen. Tätä ei kuitenkaan vaadita, jos kappale on mekaanisesti kiinni toisessa pinnassa esimerkiksi leukojen avulla. (NX help 2019.)

Jos mallinnuksessa halutaan siirtää kappaleita paikasta toiseen, tulee mallille luoda materiaalivirta. Materiaalivirta luodaan Object Source ja Object Sink -työkaluilla. Object Source -työkalulla luodaan materiaalien generointipiste, johon määritettyä objekti generoituu joko ajan, tai muun määrittelyn perusteella. Object Sink -työkalulla poistetaan generoitu objekti mallista. Malliin ei ole syytä jättää turhia objekteja, koska ne aiheuttavat turhaa laskentaa mallinnukselle. (NX help 2019.)

Materiaalivirtojen objektit tulee määrittää kiinteiksi kappaleiksi, ja niille tulee määrittää törmäävät pinnat. Törmäävillä pinnoilla tarkoitetaan pintoja, jotka törmätessään eivät voi mennä toisistaan läpi. Tämä määrittely toteutetaan Collision Body -työkalulla. Kun kappaleille määritetään pintoja, on syytä kiinnittää huomiota siihen, kuinka tarkat rajat kappaleen pinnalle määritetään. Jos kappale on monimutkainen, voi yhdellä pinnalla tehty kappaleen pinta muodostua raskaaksi laskennan kannalta ja aiheuttaa pahimmassa tapauksessa suorituskykyyn ongelmia. Pintojen muodoiksi voi valita esimerkkinä box-, sphere- tai mesh-määrittelyn. Näistä määrittelyistä mesh piirtää kappaleelle yksityiskohtaisen pinnan, mikä vaatii eniten laskentatehoa. (NX help 2019.)

Pintoja määrittäessä voidaan vaikuttaa pintojen kitkakertoimiin, jotta mallinuksesta saataisiin mahdollisimman todennukainen. Kitkakertoimet määritetään luomalla materiaali ja määrittämällä materiaalin parametrit. Tämän jälkeen materiaali linkitetään törmäävän pinnan määrittelyssä pinnan materiaaliksi. Kaikki kuljettimet, jotka kuljettavat kappaleita, tulee määrittää törmäviksi pinnoiksi. Tällöin kappaleet pysyvät kuljettimen päällä eivätkä tipahda kuljettimen läpi avaruuteen. Kuljettimia luodessa tulee ne lisäksi määrittellä Transport Surface -työkalulla, jolla tehdään kuljettimen pinta kappaleita liikuttavaksi. (NX help 2019.)

Kun kappaleeseen on määritetty ominaisuudet, voidaan niihin lisätä liikkeitä. Liikkeet luodaan liitospintojen ja halutun muuttujan mukaan, esimerkiksi vektorin tai toisen pinnan. Ohjauksiin käytetään kahta erilaista työkalua, nopeuden- sekä asennonohjaukseen. Asennonohjauksen luomiseen käytetään Position Control -työkalua (kuva 9). Nopeudenohjaustyökalua käytetään muun muassa kuljettimien kappaleiden siirtopinnan määrittämisessä. (NX help 2019.)

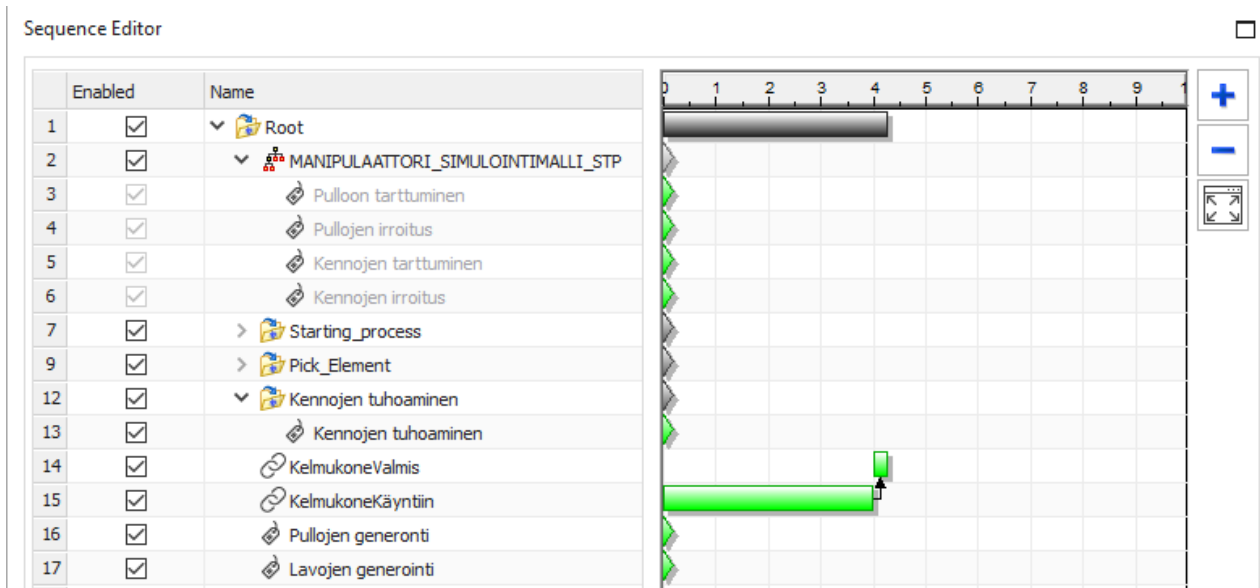


KUVA 9. Position Control -työkalu.

Seuraavaksi voidaan määrittellä järjestelmässä olevat anturit. Anturityyppejä on useita, mutta tyypillisimpänä käytetään Collision Sensor -työkalua. Tällä tarkoitetaan esimerkiksi valoverhoa tai mekaanista rajaa, joka kappaleen törmätessä pintaan aiheuttaa tilamuutoksen objektin parametrissa. Törmäysanturia voi käyttää myös simulointimallin sisäisissä toiminnoissa, kuten operaatioiden määrittämisessä. Operaatioilla voidaan muuttaa halutulla hetkellä jonkin objektin parametrejä. Esimerkiksi objektien generoinnin aktivointi voidaan suorittaa törmäysanturin ja operaation avulla. Jos törmäysanturi tunnistaa objektin, uusi objekti generoituu simulointimalliin. (NX help 2019.)

Simulointimalli voidaan rakentaa täysin pelkän MCD:n varaan, jolloin ulkoista ohjausjärjestelmää ei vaadita. Tämä on suuri etu laitevalmistelijoille, jotka haluavat testata laitteiston mekaanista toimintaa eivätkä niinkään puuttua ohjelmointityöhön. Tällainen konfiguraatio voidaan toteuttaa operaatioita luomalla. Operointien avulla aktivoidaan määritettyjä liikkeitä sarjana. Operaatiota voidaan muokata Sequence Editor -välilehdellä. Operaatiot näkyvät Gantt-kaaviona, josta voidaan seurata operaatioiden suoritusjärjestystä. (NX help 2019.)

Gantt-kaavio on yksi käytetyimmistä tavoista kuvata eri aktiviteetteja tai tapahtumia. Kaaviossa vasemmalla kuvataan kaavioon liittyvät tapahtumat ja ylhäällä esitetään aikaskaalaus. Tapahtumien kulku esitetään kaaviossa palkkeina siten, että niille määräytyy aikaskaalauksen mukaan aloitus- ja lopetuspiste. Tapahtumat esiintyvät allekkain palkkeina, joista on helppo seurata, mikäli päällekkäisiä tapahtumia syntyy. Kaaviosta selviävät myös tapahtuman kestoon liittyvät määritteet selkeästi. Gantt-kaavio soveltuu erinomaisesti projektien hallintaa ja aikataulutukseen. Siihen voidaan lisätä myös lisämääritteitä, muun muassa tekstikenttiä, joihin voidaan tarkentaa tapahtumien sisältöä. (Gantt 2020.) Kuvassa 10 on esitetty MCD:ssä käytetty Gantt-kaavion malli, josta näkyy operaatioiden esiintymisjärjestys.



KUVA 10. MCD:n Gantt-kaavio.

4.4 MCD-signaalien konfigurointi

Virtuaalisessa ympäristössä olevilla komponenteilla on erilaisia parametrejä, joita luetaan järjestelmästä ulos jonkin rajapinnan avulla. Jotta parametrejä voidaan lukea, tulee niistä luoda signaali Signal adapter -työkalulla. Signaalit linkitetään luomalla sisäiset muuttujat parametreille, jotka siirretään luotuihin signaaleihin kaavojen määrittelyssä. Työkalua käytetään mallissa olevien signaalien käsittelyyn ja liikkeiden määrittämiseen. Signaaleja varten tulee luoda järjestelmään Symbol Table, johon kaikki mallissa olevat signaalit rekisteröidään. Signaaleita luotaessa määritetään niiden parametrit. Parametreihin määritellään, esiintyykö signaali MCD:stä lähtevänä vai tulevana signaalina sekä signaalin datatyyppi. Datatyyppinä oletuksena on käytössä boolean-muuttuja, mutta se voidaan muuttaa kokonaisluvuksi tai liukuluvuksi, jos signaali linkitetään analogiseen arvoon. Työkalun avulla voidaan tehdä muun muassa laskentaa kuljettimen vetopyörän vaikutuksesta kuljetinmattoon. Tällöin lasketaan vetopyörän halkaisijan avulla, kuinka nopeasti kuljetin liikuttaa kappaletta kuljettimen matolla. Tällaiset laskennat määritetään kaavoihin. (NX help 2019.)

Signaalit liitetään simulointimallin suunnittelun loppuvaiheessa todellisiin ohjattaviin signaaleihin, esimerkiksi logiikan I/O-alueeseen, tai mikäli käytetään ainoastaan MCD-työkalua, voidaan ohjausten komennot toteuttaa Sequence Editorin avulla, Gantt-kuvaajaan piirrettyjen ohjausten perusteella. Tällöin simulointimalli voidaan täysin rakentaa MCD:n avulla. Signaalien käsittely ulkopuoliseen ohjausjärjestelmään toteutetaan External Signal Configuration -työkalun avulla. Sillä valitaan, mitä protokollaa halutaan käyttää signaalien tulkitsemiseen. MCD:hen voi liittää signaaleita alla olevan taulukko 2:n mukaisesti. Taulukkoon on merkattu myös käytettävissä olevat datatypit. (NX help 2019.)

TAULUKKO 2. Ulkoisten signaaleiden protokollat ja käytettävät datatyypit. (mukaihen NX help 2019.)

PROTOKOLLAT	DATATYYPIT
MATLAB	Bool (1 bit)
OPC DA	Byte (1 byte)
OPC UA	Word (2 byte)
PLCSIM Advanced	Int (2 byte)
Profinet	DWord (4 byte)
SHM	DInt (4 byte)
TCP	Real (4 byte)
UDP	DReal (8 byte)

Kun signaalit on päivitetty MCD:hen, käytetään Signal Mapping -työkalua. Tällä linkitetään MCD:ssä luotu signaali ulkoisesta järjestelmästä tuotuun signaaliin. Signaalien linkityksen voi suorittaa myös automaattisesti, jolloin samannimiset signaalit linkittyvät keskenään. Tämä toiminto nopeuttaa linkittämistä huomattavan suuresti. (NX help 2019.)

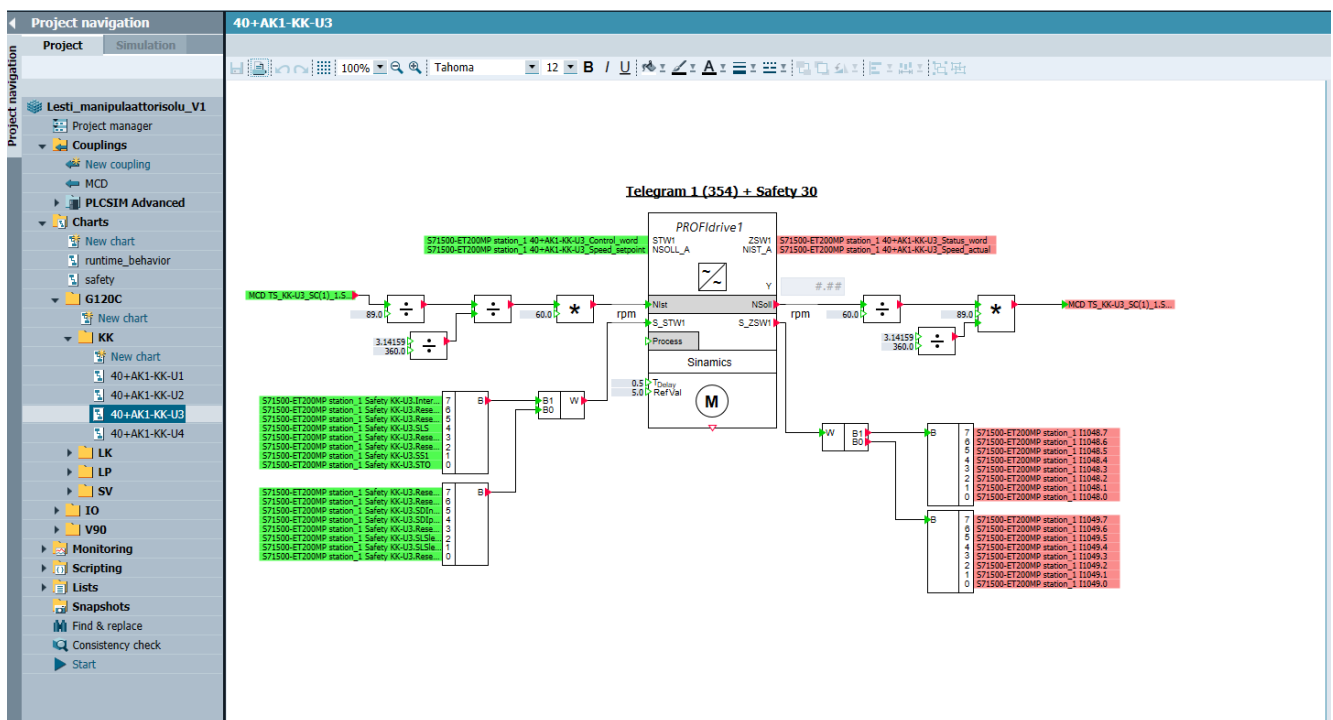
Kun kaikki signaalit on linkitetty, testataan signaalit läpi ja aloitetaan 3D-mallin simulointi. Simulointi käynnistetään Play-painikkeesta ja voidaan pysäyttää Stop-painikkeesta. Jos simulointi halutaan käynnistää, joltain halutulta ajanhetkeltä kesken simuloinnin, voidaan simuloinnin aikana suorittaa Snapshot-toiminto, jolloin kyseistä hetkestä voidaan jatkaa simulointia. Tämä voi joissain tilanteissa nopeuttaa simulointia huomattavasti. Simulointi voidaan käynnistää uudelleen Runtime Inspector -työkalusta Snapshot-välilehden alta. (NX help 2019.)

4.5 SIMIT-ohjelmisto

SIMIT-ohjelmisto on Siemensin kehittämä simulointiympäristön luomiseen tarkoitettu sovellus. SIMIT-ohjelmalla voidaan luoda rajapintoja eri simulointiympäristöjen välille tai luoda ympäristö itse SIMIT-ohjelmaan. Sovelluksella on tarkoituksena toimia simulointialustana virtuaalista käyttöönottoa varten. Se sisältää valmiita kirjastoja, joita voidaan hyödyntää simulointimallin kehittämisessä. Simulointimalliin voidaan lisätä myös analyysimittauksia, jotta saadaan kerättyä dataa mallin toiminnasta. (SIMIT Simulation Platform 2019.)

Tässä opinnäytetyössä ei ollut alkuun tarkoitusta käyttää SIMIT-ohjelmistoa, sillä ei koettu sen olevan tärkeä osa digitaalisen kaksosen luomista, koska NX:stä löytyi itsessään rajapinta PLCSIM Advance-diin. Opinnäytetyöprosessin aikana törmättiin kuitenkin suorituskykyongelmiin NX:n ja PLCSIM Advanced-ohjelmiston välillä, minkä vuoksi jouduttiin ottaa käyttöön myös SIMIT-ohjelmiston. Sitä käytäessä saatiin simulointiin tuotua etuja muun muassa taajuusmuuttajien simulointiin. SIMIT-ohjelmistossa on olemassa valmiita kirjastoja, joten moottorien käyttäjien simulointi saatiin toteutettua mahdollisimman todenmukaiseksi. Valmiiden kirjastojen ja kehitettyjen lohkojen avulla kyettiin diagnosoimaan taajuusmuuttajien sekä servovahvistimien dataa laajasti. Niistä voitiin tutkia telegrammien bittien tilojen vaihtelua niin tavallisen kuin PROFIsafe sanoman osalta. (SIMIT Simulation Platform 2019.)

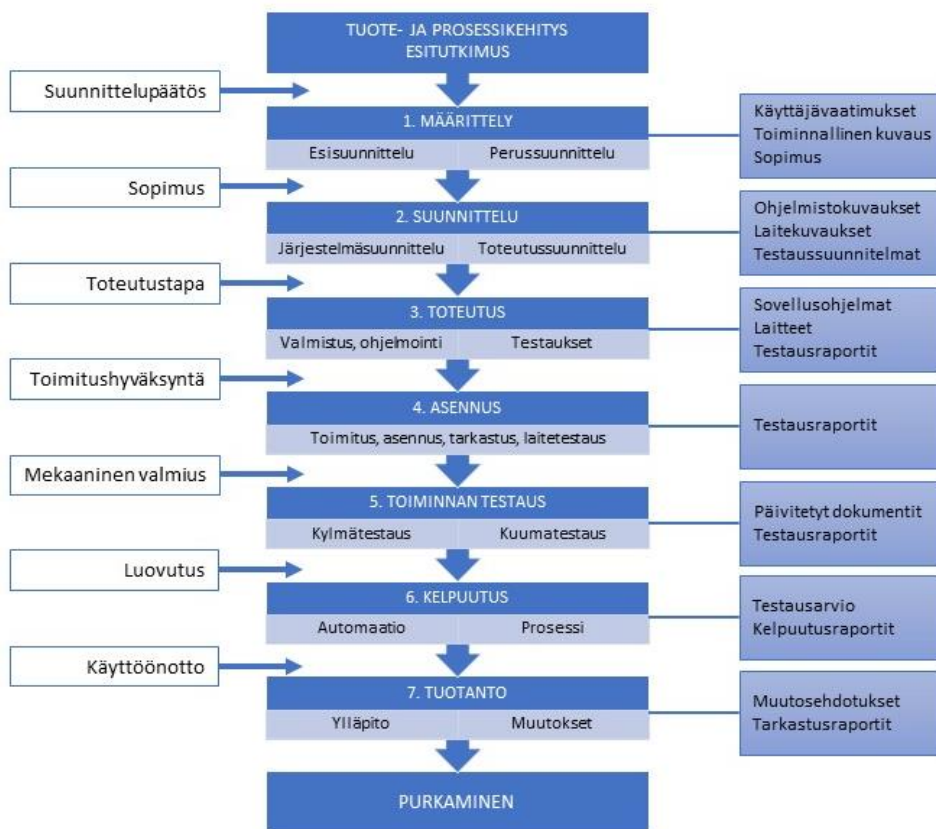
SIMIT-ohjelmaan lisättiin taajuusmuuttajien kiihdytys- ja hidastusrampit valmiiden lohkojen avulla. Lohkojen avulla saatiin malli käyttäytymään mahdollisimman todenmukaiseksi. Rampit olisi voinut myös luoda TIA Portaaliin, mutta tällöin projektia olisi joutunut muokkaamaan simulointia varten liikaa. Kuvassa 11 on esimerkki SIMIT:ssä käytetystä taajuusmuuttajan simulointiympäristöstä.



KUVA 11. SIMIT, telegram 1 määrittely.

5 PROJEKTIN VAIHEET

Tässä luvussa kerron opinnäytetyöprosessin kulusta ja siitä, mitä kaikkea projektiin kuului. Käyn läpi, miten projekti eteni ja mitä kaikkea virtuaalista käyttöönottoa varten tehtiin. Projektin etenemiseen hyödynsin automaatio suunnittelun elinkaarimallia (KUVA 12), joka käsittelee automaatiojärjestelmän elinkaaren vaiheet sekä niin sanotut etapit ja tärkeimmät tulokset. Vaikka malli kuvaa kokonaisen automaatiojärjestelmän elinkaarimallia, koin mallin olevan riittävän tarkka manipulaattorisolun suunnitteluprosessin tukena.



KUVA 12. Automaatiosuunnittelun elinkaarimalli. (mukaiillen Suomen Automaatioseura ry 2007).

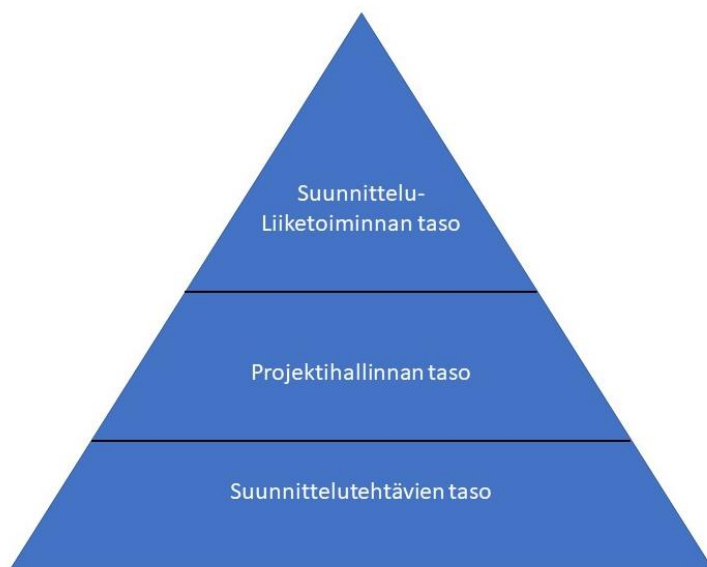
Vaikka edellä mainitsin, että käytin elinkaarimallia suunnitteluprosessin tukena, ei se kuvaa täysin, mitä suunnittelija tekee projektissa. Tehtävät saattavat olla vaikeita hahmotella johonkin tiettyyn elinkaarimallin vaiheeseen, koska ne saattavat sisältää useita vaiheita samanaikaisesti. Jotta oleelliset asiat saataisiin mahdollisimman tarkasti hahmoteltua, tulisi suunnittelua tarkastella eri tarkkuustasolla sekä monesta näkökulmasta. Jos yhteen kuvaukseen kerätään kaikki siihen liittyvä tieto, paisuu kuvaus monimutkaiseksi. Tämän vuoksi on kuvaus jaoteltava eri toimijoiden mukaan siten, ettei kaikkien tarvitse tietää kaikkia yksityiskohtia. Projektipäällikön tehtävät poikkeavat suuresti ohjelmoijan tehtävistä,

minkä vuoksi heidän ei tarvitse yksityiskohtaisesti tietää toistensa tarpeita. Riittää, että heillä on riittävä käsitys toistensa toimintatavoista. Tähän tapaan tarkastella projektin vaiheita on keksitty tarkkuustasot, jotka jaottelevat vaiheen tehtävät eri toimijoille (KUVA 13). (Suomen Automaatioseura ry 2007.)

Ylimpänä mallissa kuvataan suunnitteluliiketoiminnan tasoa. Se kattaa liiketoiminnan kannattavuuden sekä toimintaprosessin kehittämisen. Tässä tasossa suunnittelua voidaan kuvata elinkaarimalleina, jotka paloittelevat suunnitteluprosessin sarjaksi erilaisiin vaiheisiin sekä tehtäväkokonaisuuksiin. Organisaatioiden, asiakkuuksien, alihankintaketjujen sekä projektitoiminnan kannattavuuden hallitseminen sisältyvät tähän tasolle. (Suomen Automaatioseura ry 2007.)

Seuraavassa tasossa tarkastellaan projektihallinnan tasoa, joka kattaa yksittäisen projektin suunnitteluprosessin ilmentymänä. Projektissa saattaa esiintyä poikkeamia, mutta se pyrkii noudattelemaan määritettyä suunnitteluprosessia. Tavoitteena on projektikohtaisesti tehdä määrittely asioista, joita ei olla kuvattu suunnitteluprosessissa. (Suomen Automaatioseura ry 2007.)

Alinta tasoa pidetään suunnittelijoiden henkilökohtaisten työskentely- ja ajatustapojen tasona, johon sisältyvät myös yksittäiset suunnittelutehtävät. Työskentelytapoja on haastavaa alkaa mallintamaan, mutta on hyvä tiedostaa, että suunnittelijoilla on erilaisia työtapoja ja tästä syystä olisi suositeltavaa luoda yhteisiä toimintamalleja, joilla voitaisiin tehostaa suunnittelijoiden työskentelyä. (Suomen Automaatioseura ry 2007.)



KUVA 13. Automaatiosuunnittelun tasot. (mukailten Suomen Automaatioseura ry 2007).

5.1 Projektin määrittely

Opinnäytetyöprosessi alkoi tutustumalla Siemensin NX-ohjelmistoon. Alkuun valitsimme yritykseltä tietokoneen, johon ohjelmistot asennettaisiin. Päädyimme asentamaan ohjelmiston tietokoneelle, jossa oli mahdollisimman hyvät tekniset ominaisuudet. Lisäksi tietokoneen tuli olla yrityksen sisäisessä verkossa, jotta siitä kyettäisiin tekemään lisenssipalvelin yrityksen verkkoon. Tarkoituksena oli, että tietokoneelle asennettaisiin NX-suunnitteluohjelmisto sekä lisenssipalvelut. Muut käyttäjät voisivat asentaa omille tietokoneilleen MCD-Player-ohjelman, jolla voitaisiin simuloida valmista virtuaalista mallinusta. Ohjelmistot ladattiin Siemensin ja IDEAL PLM-yrityksien kautta. Ohjelmistojen asentaminen aloitettiin lisenssipalvelusta. Kun lisenssipalvelin oli saatu toimimaan toivotulla tavalla, siirryttiin asentamaan NX-ohjelmistoa sekä MCD-Designer-sovellusta. Kun oli todettu ohjelmistot toimiviksi, aloitettiin projektiin tutustuminen.

Projektin tarkoituksena oli korvata vanhentunut robotti, joka pakkasi pulloja ja pakkauskennoja kuormalavojen päälle. Kohteessa haluttiin lisätä pakkauskapasiteettia. Aiemmin oli todettua, että olemassa olleella robotilla ei kyetty saavuttamaan haluttua kapasiteettia vaan kohteessa vaadittiin päivitystä. Vastaavia pakkauskohteita löytyy asiakkaalta useampi, minkä vuoksi haluttiin suunnitella kone, jota voitaisiin hyödyntää myös mahdollisesti muissa kohteissa. Tämän vuoksi aloimme suunnitella kohteeseen manipulaattoria, jota voitaisiin myöhemmässä vaiheessa muokata halutun kohteen mukaan.

Manipulaattorilla haluttiin pakata pulloja eurolavojen päälle ja siirrellä pakkauskennoja pullokerrosten väliin. Lavoille suunniteltiin oma siirtovaunu ja lavanpurkausjärjestelmä, joilla siirrettiin lavoja manipulaattorin pakkausasemaan. Manipulaattoriin suunniteltiin neljä servoakselia, joilla ohjattiin manipulaattorin kahta työkalua x- ja z-akselien suuntaan. Työkalut toteutettiin paineilma-avusteisilla sylintereillä sekä tarttujilla.

Ennen projektin varsinaisen suunnittelun alkamista tuli projektia varten määrittellä I/O-luettelo. I/O-luetteloon määritettiin kaikille anturoinneille, ohjauksille sekä laitteille positiotunnukset. Positiotunnusten lisäksi listaan määritettiin laitteiden tyypit, IP-osoitteet, kuvaukset, osoitealueet logiikassa sekä datatyytit. I/O-luettelon avulla projektin hallitseminen selkeytyy, kun kaikki laitteet on listattu ja positioitu. Luettelosta on apua myös projektin käyttöönotossa ja muutosten hallinnassa.

5.2 Projektin suunnittelu

Varsinaisen projektin automaatio suunnittelu tehtiin TIA Portal -työkalulla. TIA Portal on Siemensin kehittämä automaatio suunnitteluun tarkoitettu ohjelmisto. Ohjelmistolla määritetään projektissa käytettävät automaatiolaitteet ja mahdolliset kenttäväylät tai automaatioverkot. TIA Portal-sovelluksella voidaan suunnitella muun muassa logiikkaohjelmia, operointipaneelien käyttöliittymiä ja PC-pohjaisia valvomo-ohjelmia. (Totally Integrated Automation Portal 2020.)

Automaatio suunnittelu aloitettiin määrittelemällä hardwaren projektiin. Hardware määrittää automaatiojärjestelmässä käytetyt laitteistoratkaisut. Automaatiojärjestelmässä olevia laitteita voivat olla esimerkiksi logiikat, operointipaneelit sekä PC-valvomot. Hardwarea voidaan kutsua myös nimellä laitteistokuvaus. Automaatiolaitteiden, automaatioverkkojen sekä kenttäväylien määrittäminen suoritetaan laitteistokuvauksessa. (Ajo ym. 2001.)

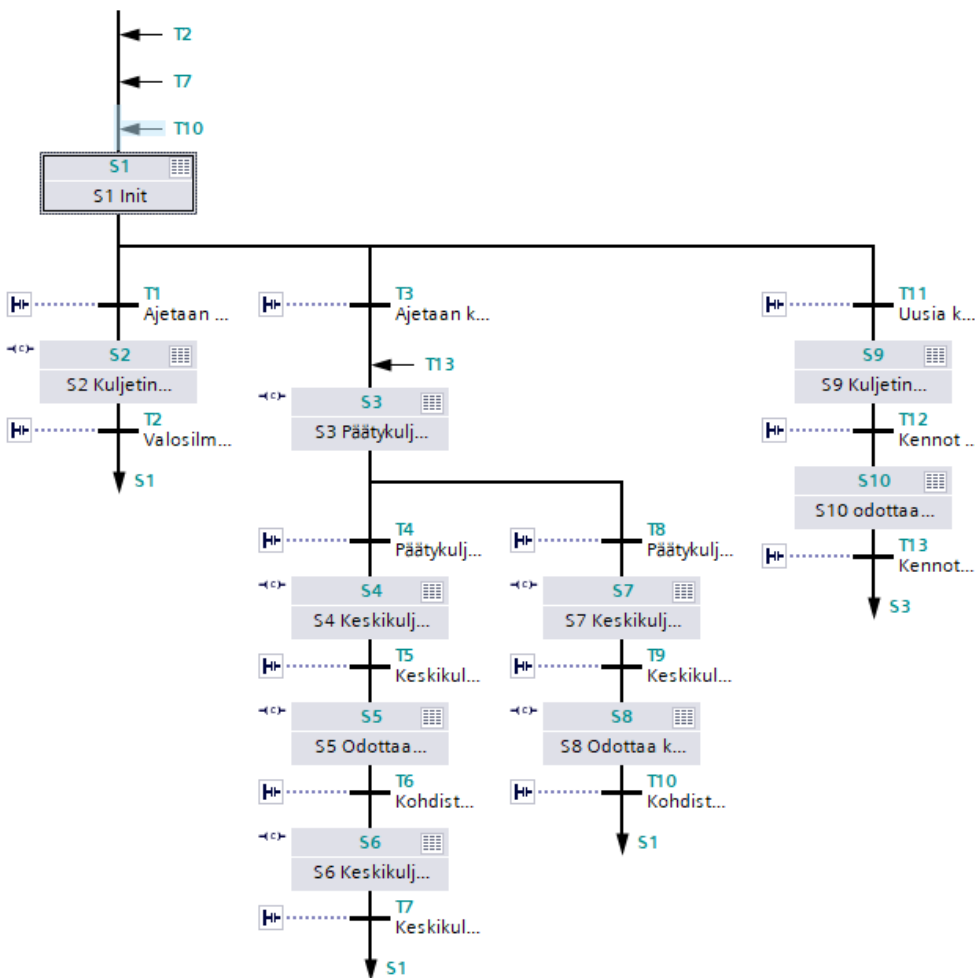
Alkuun hardwareen lisättiin logiikka, hajautettua I/O:ta, neljä V90-servovahvistinta sekä muutamia G120C-taajuusmuuttajia. Akseleiden toiminta toteutettiin käyttämällä teknologiaobjekteja. Tämä tehtiin sen vuoksi, että teknologiaobjektit voidaan asettaa simulointitilaan, jolloin simuloinnin suorittaminen on yksinkertaisempaa solulle. G120C-taajuusmuuttajien simulointi toteutettiin lisäämällä simulointilabitti taajuusmuuttajien ohjauslohkoihin. Ohjauslohkon sisällä bitillä määritettiin taajuusmuuttajan tilasana kuvaamaan taajuusmuuttajan valmistilaa. Tämä toteutettiin, jotta ulkoista liityntää taajuusmuuttajien rajapintaan ei tarvinnut suunnitella. Yksi tapa tehdä taajuusmuuttajien rajapinta kuntoon ilman erillistä simulointibittiä olisi ollut käyttää Siemensin Crosslink-työkalua. Jätin Crosslink-ohjelman käyttämättä, koska koin saavani pidettyä ohjelman riittävän yksinkertaisena ilman erillistä simulointiohjelmaa. Simulointibittiä ohjattiin MCD:n avulla simuloinnin käynnistyttyä. Tällöin varmistettiin, ettei todellisessa tilanteessa simulointitila jää vahingossa aktiiviseksi.

Myöhemmässä vaiheessa projektia jouduttiin kuitenkin tekemään ulkoinen kättely taajuusmuuttajien simulointia varten, jolloin ohjauslohkoon luotu simulointitila jäi käyttämättömäksi. Tämä johtui MCD:n ja PLCSIM Advancedin välisestä suorituskykyongelmasta. Näin ollen taajuusmuuttajien simulointi suoritettiin SIMIT-ohjelmistoa hyödyntäen.

Logiikan ohjelmointi toteutettiin jakamalla solun toiminnallisuudet omiin osioihin. Jokainen solunosa sisälsi yhteisten parametrien lisäksi oman datalohkon parametreille. Laskennoille, lukituksille, sekvenssille ja ohjauksille luotiin jokaiselle omat lohkot, jotta ohjelma pysyisi mahdollisimman yksinkertaisena

ja olisi helposti muokattavissa. Mikäli samankaltaisia toiminnallisuuksia tulisi vastaan tulevaisuudessa, olisi kokonaisuus lähes valmiina. Myöskin, jos jokin toiminnallisuus poistuisi ja tilalle tulisi toinen, olisi ohjelma helposti muokattavissa. Koska toiminnallisuudet jaoteltiin omina laitteina, eivät häiriötilanteet keskeyttäisi muiden sekvensseiden toimintaa vaan kaikki osa-alueet toimisivat omina.

Sekvenssien ohjelmointiin käytettiin S7-GRAPH- ja LAD-ohjelmointikieliä. S7-GRAPH ohjelmointikieli (KUVA 14) kuvaa visuaalisesti sekvenssin toimintaa puumaisena kaaviona. Tämä antaa hyvää visuaalista palautetta ohjelmoijalle. Akseleiden ohjauksissa ja laskennoissa käytettiin SCL-ohjelmointikieltä (KUVA 15), koska sillä on joustavaa ja tilaa säästävää tehdä toistuvia ohjelmarakenteita. Alla esimerkit S7-GRAPH- sekä SCL-ohjelmointikielistä. (Siemens GRAPH 2018 & SIMATIC S7-SCL V5.3 for S7-300/400 2005.)



KUVA 14. S7-GRAPH-ohjelmointikieli.

```

134 REGION PULLOJEN POSITIOIDEN MÄÄRITYS
135 // Lasketaan pullojen hakuun liittyvät positiot Z-akselille
136 //Määritetään pullojen positiot
137 //
138 FOR "60 Parametrit"."Pullojen määrittely".For_Integer := 0 TO "60 Parametrit"."Pullojen määrittely".For_Integer - 1
139
140 // "60 Parametrit"."Pullojen määrittely".PulloKerrokset_LK["60 Parametrit"."Pullojen määrittely".For_Integer]
141
142 "60 Parametrit"."Pullojen määrittely".PulloKerrokset_LK["60 Parametrit"."Pullojen määrittely".For_Integer] :=
143 (#Alinkerros_K - "60 Parametrit"."Kennojen määrittely".KennonKorkeus - "60 Parametrit"."Pullojen määrittely".For_Integer) *
144 "60 Parametrit"."Pullojen määrittely".For_Integer;
145
146
147 IF "60 Parametrit"."Pullojen määrittely".PulloKerrokset_LK["60 Parametrit"."Pullojen määrittely".For_Integer] > 0
148 "60 Parametrit"."Pullojen määrittely".PulloKerrokset_LK["60 Parametrit"."Pullojen määrittely".For_Integer] :=
149 "60 Parametrit"."Pullojen määrittely".PulloKerrokset_LK["60 Parametrit"."Pullojen määrittely".For_Integer] - 1;
150
151 END_IF;
152
153 END_FOR;
154 END_REGION

```

KUVA 15. SCL-ohjelmointikieli.

5.3 Simulointimallin toteutus

Logiikkaohjelman edetessä käytettiin välillä aikaa simulointimallin mallintamiseen. Mallintaminen aloitettiin kokoonpanon järjestämisellä toiminnallisuuksiin, jotta liikkeiden määrittäminen olisi yksinkertaisempaa. Kokoonpanon järjestämisen jälkeen aloitettiin kiinteiden kappaleiden määrittäminen Rigid Body -työkalun avulla. Tein yhden osakokonaisuuden kerrallaan valmiiksi, jotta pystyin keskittymään paremmin kyseisen laitteen toiminnallisuuksiin. Kun laitteen liikkuvat objektit oli saatu määriteltyä kiinteiksi, määritettiin törmäävät pinnat Collision Body -työkalulla. Törmääviksi pinnoiksi määriteltiin objektien pintoja, jotka liikkuvat jonkin tason päällä tai estivät objektien liikkeitä, esimerkiksi seinät. Pinnan materiaali voitiin määrittää kitkakertoimien vuoksi, jolloin simulointimalliin saatiin määritettyä objekteille todenmukaisia liikkeitä. Mallissa luotiin kuormalavojen pinnoille oma materiaali.

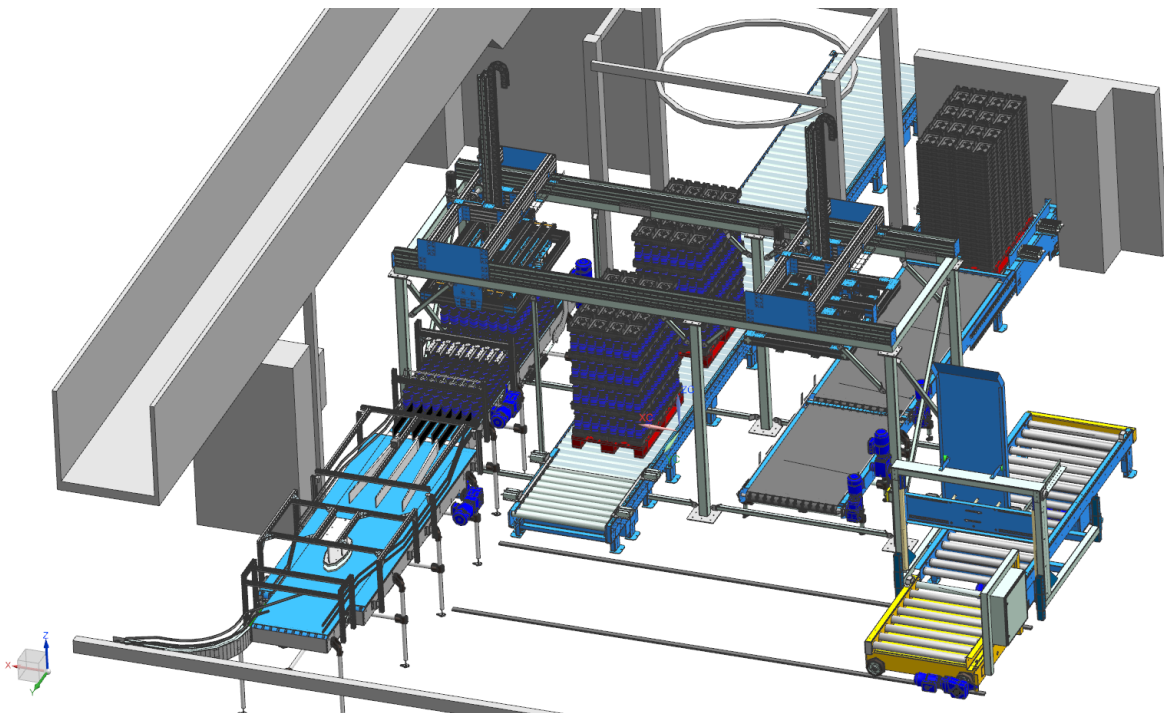
Seuraavaksi laitteille määriteltiin liikkeitä. Sylinteriohjauksissa käytettiin Sliding Joint- sekä Position Control -työkalua. Sylinteriliikkeitä määrittäessä tärkeäksi työkaluksi koin mittastyökalun. Sen avulla voitiin mitata sylinteriliikkeiden pituus, jotta liike olisi mahdollisimman todenmukainen. Liikkeen pituus määriteltiin Signal Adapter -työkalulla. Päätin hyödyntää IF-ELSE-komentoja liikkeiden määrittämisessä sylintereille.

Liikkeiden valmistuttua määritettiin laitteiden anturoinnit Collision Sensor -työkalulla sekä luotiin tarvittavat operaatiot objektien tarttumisia ja irrotuksia varten. Tämän valmistuttua linkitettiin signaalit

Signal Adapterin avulla. Sylinterien anturointien määrittystä en kaikissa tilanteissa kokenut tärkeäksi vaan linkitin sylinterin ohjauskäskyn suoraan auki- ja kiinnirajatietoon. Joissain tilanteissa kuitenkin sylinterin tilatieto oli aikakriittinen, minkä vuoksi loin anturit kuvaamaan niiden tilaa. Tämän jälkeen yhdistettiin PLCSIM Advanced-instanssi MCD-sovellukseen ja linkitettiin signaalit keskenään Signal mapping -työkalulla. Kun rajapinta oli valmis, aloitin virtuaalisen käyttöönoton manipulaattorisolulle.

Simulointimallista jätettiin pois pullojen syöttöpöydän simulointi. Simulointia ei tehty, koska havaittiin simulointimallin olevan hyvin raskas tietokoneelle ja jatkuva pullojen generointi simulointimalliin olisi aiheuttanut valtavan kuormituksen tietokoneelle. Pullojen simulointi suoritettiin mallissa generoimalla valmis erä pulloja syöttöpöydän pätyyn, jolloin manipulaattori kykeni hakemaan pullot syöttöpöydältä. Pullojen generoinnissa hyödynnettiin MCD-sovelluksessa olevaa operaatiotyökalua. Operaatiossa tutkittiin manipulaattorin pullotyökalun pullojen tarttuvien paineilman ohjausta. Mikäli ohjaus meni päälle, liipaisi MCD pullojen Object Source -objektin aktiiviseksi, jolloin pullot generoituivat syöttöpöydän hihnalle.

Manipulaattorisolun 3D-malli on esitetty kuvassa 16. Kuvassa näkyvät kaikki solussa olevat koneet sekä pakkauksessa esiintyvä materiaali. Solun malli ei rakenteellisesti vastaa täysin todellista ympäristöä, mutta malli on koneiden osalta tarkka simulointia varten.

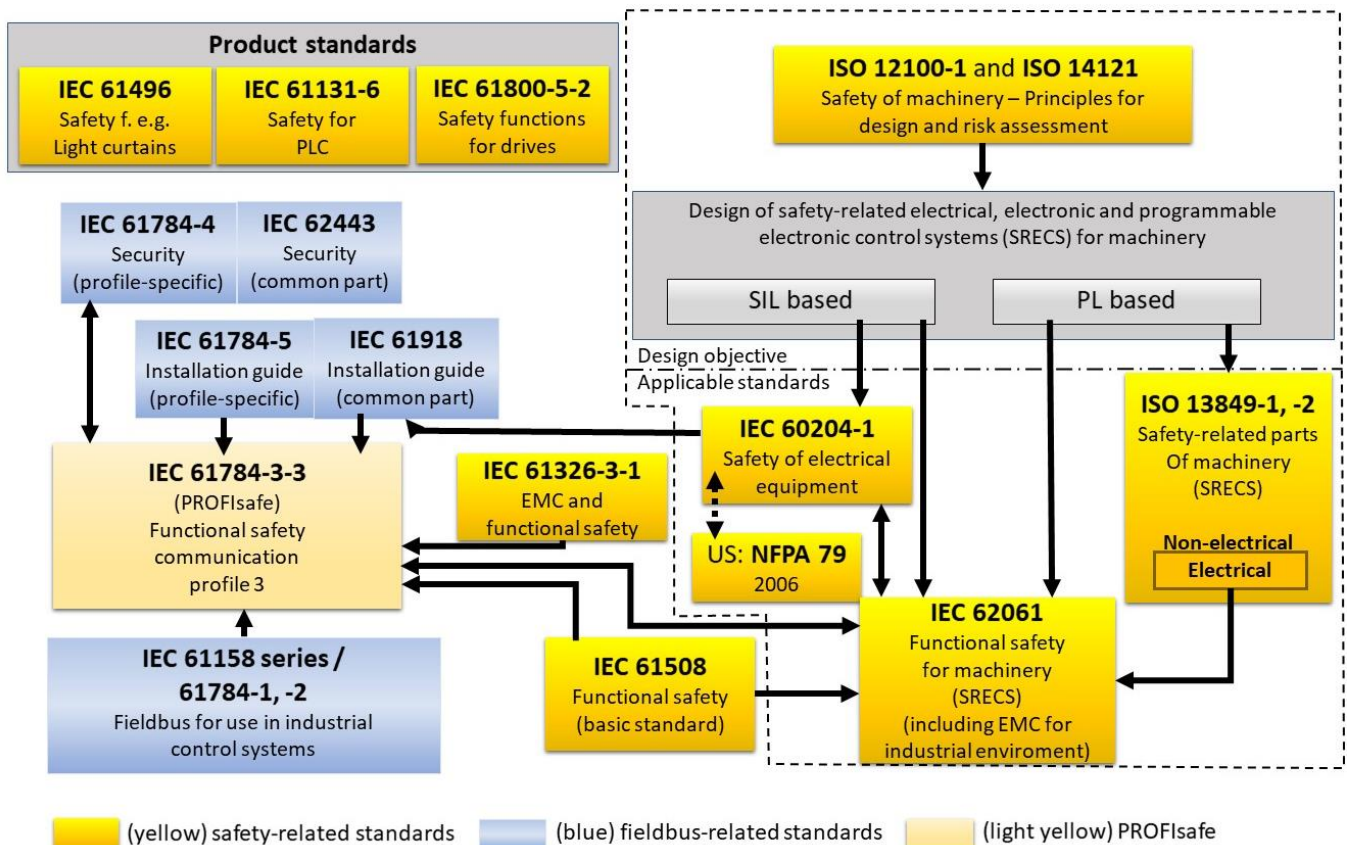


KUVA 16. Manipulaattorisolun 3D-malli.

5.4 Virtuaalinen käyttöönotto

Ensimmäinen huomio, joka virtuaalisen käyttöönoton alussa ilmeni, oli turvatoimintojen simulointi. Käytössä oli Siemensin F-sarjan CPU, mikä mahdollisti turvaohjelman luomisen logiikkaohjelman rinnalle PLC:hen. Turvaominaisuuksia tässä vaiheessa projektia olivat turva-alue sekä hätäseis-piirit. Jotta logiikkaohjelman turvaohjelma saataisiin kuitattua turvalliseen tilaan, tuli MCD:hen luoda signaalit hätäseis-painikkeesta sekä turva-alueesta. Simulointimallia simuloidessa signaaleita voitiin manipuloida ja täten voitiin testata myös hätäseispiirin ja turva-alueen toiminta. SIMIT-ohjelmiston lisäys projektiin kasvatti turvaominaisuuksien diagnosointia ja simulointia. Hätäseis-piirien lisäksi turvaominaisuuksia voitiin hyödyntää G120C-taajuusmuuttajissa, joissa hyödynnettiin PROFIsafe-tekniikkaa.

PROFIsafe-tekniikalla tarkoitetaan tiedonsiirtoprofiilia, joka toimii PROFIBUS- tai PROFINET-väylän rinnalla. Se on sertifioitu turvallistamiseen tarkoitettu väylätekniikkaratkaisu, jolla kyetään yhteensopivien turvatoimintojen toteuttamiseen. Tekniikka on standardisoitu IEC 61158 ja IEC 61784-1/-2 standardien mukaiseksi. Se otettiin ensimmäistä kertaa käyttöön vuonna 1999. IFA (Institut für Arbeitsschutz) sekä TÜV (Technischer Überwachungsverein) ovat hyväksyneet PROFIsafe-tekniikan käytön. Kuvassa 17 on kuvattu koneautomaatioon liittyviä turvallisuusstandardeja. (PROFIsafe System Description 2010.)



KUVA 17. Koneturvallisuuteen liittyviä turvallisuusstandardeja. (mukaiillen PROFIsafe System Description 2010)

Ennen turvatoimintojen testausta suoritettiin simulointimallille I/O-testi. I/O-testissä varmistettiin, että signaalien linkitys on tapahtunut oikein ja kaikki anturit ja ohjaukset ovat kytkettyinä oikein. Signaalien tilojen vaihtelua tarkasteltiin TIA Portalista online-näkymästä sekä MCD:stä runtime inspectorista. I/O-testaus suoritettiin I/O-luettelon pohjalta, jolloin varmistettiin, että kaikki signaalit käytiin läpi.

Moottoreiden testaus suoritettiin käynnistämällä moottori HMI-paneelistä ja tutkimalla taajuusmuuttajan diagnostiikkaa. Jos SIMIT-ohjelmistossa taajuusmuuttajien telegrammien tilasanat ja ohjaussanat vastasivat logiikassa esiintyviä sanoja, voitiin todeta kommunikoinnin toimivan, ja moottorin käyttäytymän kuten suunniteltiin. Ennen SIMIT-ohjelman käyttöönottamista moottorit testattiin lähettämällä ohjauskomento käynnistää moottorit, minkä jälkeen tutkittiin MCD:ssä, pyöriikö haluttu moottori ja miltä moottorin nopeus vaikuttaa simulointimallissa.

Kun turvatoiminnot olivat kunnossa, testattiin niiden toiminta simuloinnin aikana. Hätäseis voitiin manipuloida lauenneeksi MCD:stä vaihtamalla hätäseisbitin tila false-tilaan. Tämä aiheutti hätäpysäytystoiminnon logiikassa, mikä pysäytti kaikki moottorit ja liikkeet, jotka olivat aktiivisena. Se myös aiheutti hälytyksen HMI-paneelin hälytyslistaan ja lukitsi moottorien uudelleenkäynnistyksen. Paneeliin oli lisäksi määriteltynä pop-up-ikkunat hätäseis ja turva-alueen hälytyksestä. Näiden toiminta voitiin myös tarkistaa simuloinnin yhteydessä.

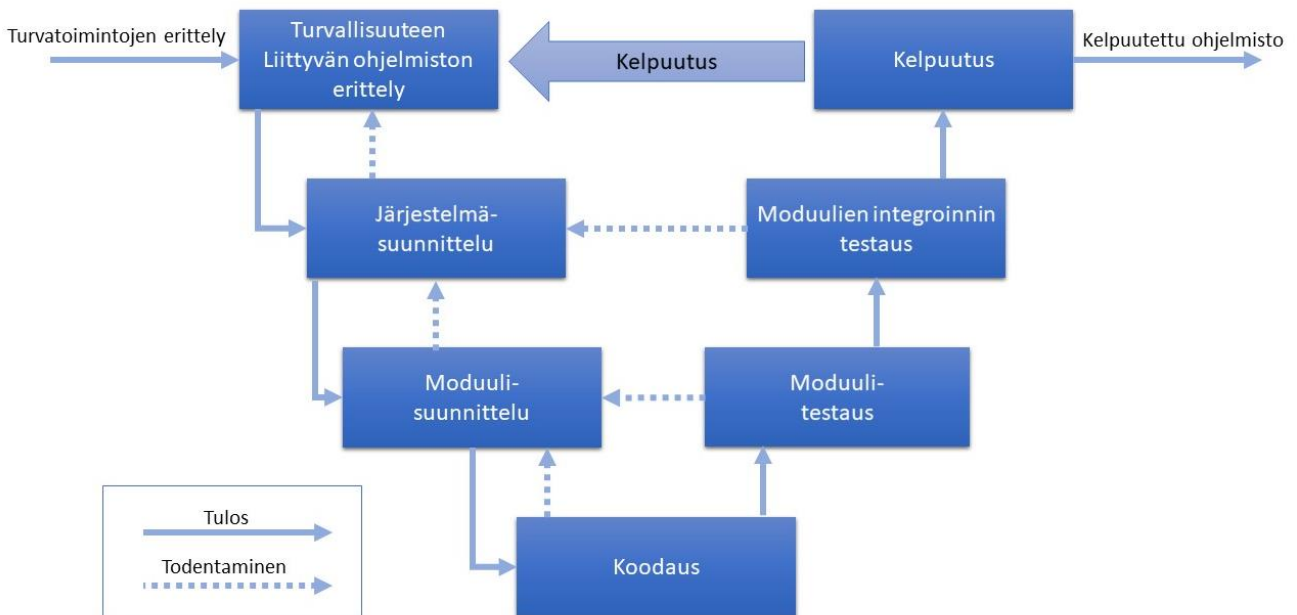
Edellä olleiden testausten jälkeen siirryttiin logiikkaohjelman testaamiseen. Logiikkaohjelman testaus suoritettiin laitteistokohtaisesti yksi kerrallaan. Alkuun testattiin laitteen sekvenssi, mikä kertoi, onko laitteen toiminta sellainen kuin suunnitteluvaiheessa oli määritelty. Automaatioseuran prosessimallin mukaan määrittelyt saattavat kuitenkin projektin edetessä elää, minkä vuoksi määrittelyjä tulisi tarkentaa ja täsmentää mahdolliset poikkeustilanteiden hallinnat sekä hälytysten käsittely ja käyttöliittymien hallinta. Määrittely tulisi suorittaa suurista kokonaisuuksista alkaen ja päättää yksityiskohtaisiin kuvauksiin. Tällöin testaaminen voitaisiin suorittaa suunnitteluvaatimusten mukaan, jolloin laitteiston luovutusta varten tarvittavat testaukset on suoritettu. (Suomen Automaatioseura ry 2007.)

Sekvenssien testauksien yhteydessä suoritettiin laitteistojen lukitusten ja hälytysten toiminnan testauksia. Manipulaattorisolun laitteistojen ohjelmoinnissa noudatettiin Suomen konedirektiivin vaatimuksia

ohjausjärjestelmälle. Lukituksina manipulaattorissa toimi muun muassa vastakkaisten akseleiden liikkeiden lukitus, jos akseleilla lähestyttiin törmäyspistettä. Ohjelmallisilla lukituksilla ehkäistiin koneen mekaanista rikkoontumista pysäyttämällä laite ennen kriittisen toiminnan tapahtumista. Ohjelmallisilla lukituksilla voidaan myös ehkäistä henkilöihin kohdistuvia vaara tai uhkatilanteita, mutta henkilöihin liittyvät suojaustoiminnot tulee toteuttaa turvallisuusstandardin IEC 61508 mukaisesti. (SFS-EN 62061, 10)

Ohjelmalliset lukitukset on esitetty SFS-EN ISO 13849 ja SFS-EN 62061 koneturvallisuuden liittyvissä standardeissa. Standardin SFS-EN 62061 mukaan koneiden turvallistamiseen käytetään useissa tapauksissa sähköistä ohjausjärjestelmää, joka vähentää riskejä koneessa. Tavallisesti koneeseen on asennettu suojaus, mikä avatessa sallii henkilön kulun vaara-alueelle ja lähettää signaalin ohjausjärjestelmälle, jotta vaarallinen toiminta koneessa loppuu. Riskeistä, jotka aiheutuvat, mikäli sähköinen ohjausjärjestelmä vikaantuu, on standardissa esitetty vaatimuksia ja menetelmiä, joilla niitä koitetaan minimoida. Turvallistamiselle löytyy eheystasoja, jotka tulee asettaa jokaiselle turvallisuuden liittyvälle ohjaustoiminnolle. Sähköinen ohjausjärjestelmä tulee olla lisäksi turvallisuuden kelpuutettu. (SFS-EN 62061, 10)

Standardissa SFS-EN ISO 13849 on esitetty turvallisuuden liittyvät ohjausjärjestelmien osat koneturvallisuuden liittyen. Ohjelmiston turvallisuusvaatimuksissa kerrotaan, että ohjelmisto tulee suunnitella siten, että se kattaa turvallisuuden liittyvät toimenpiteet koko elinkaarensa aikana. Tällöin laitteen vikaantumiset eivät saa vaikuttaa laitteen turvallisuuteen. Alla olevassa kuvassa on esitetty koneen elinkaaren aikana kuuluvat suositellut toimenpiteet. (SFS-EN ISO 13849-1, 29-30.)

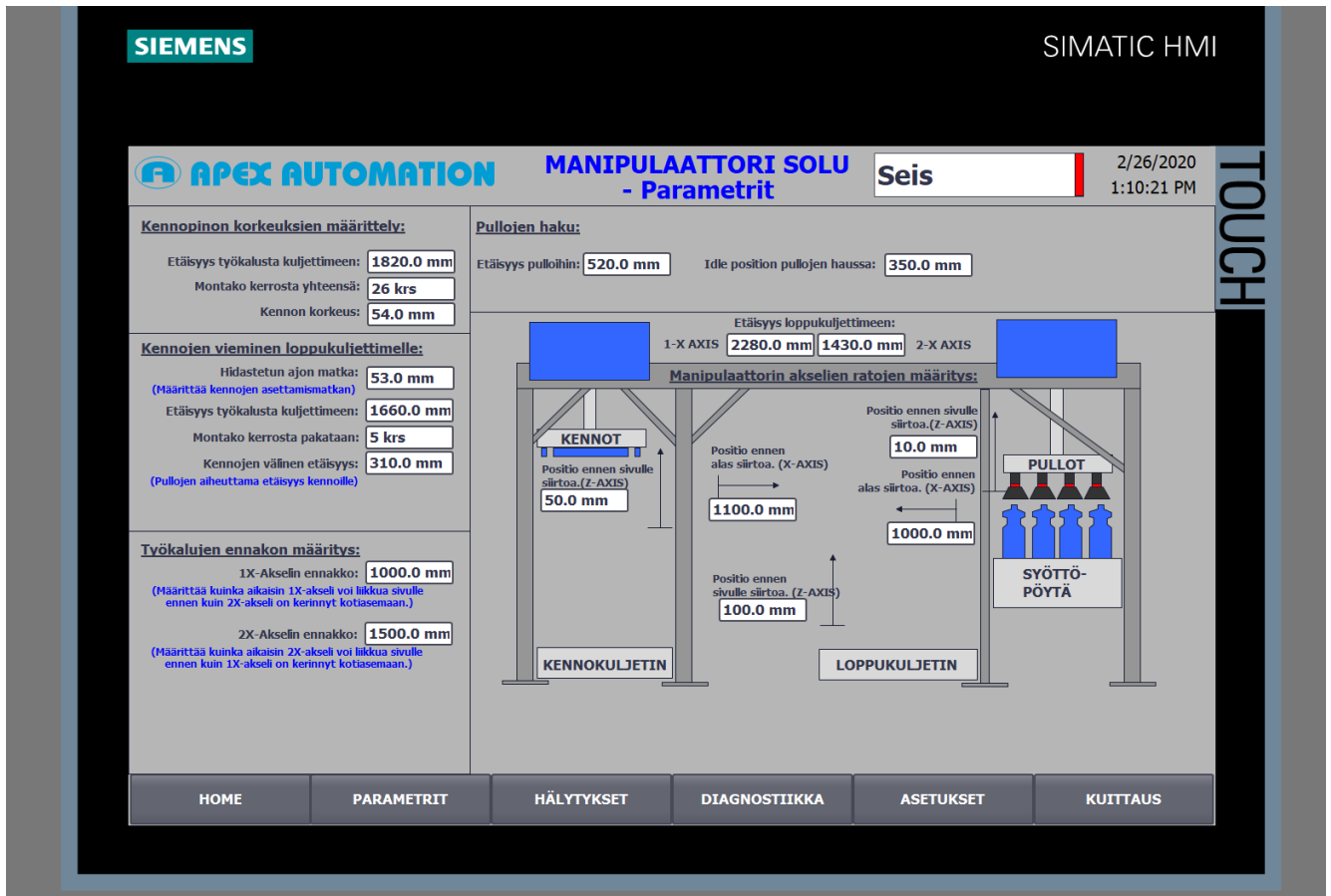


KUVA 18. Ohjelmiston turvallisuuselinkaaren yksinkertaistettu V-malli. (mukaien SFS-EN ISO 13849-1)

Manipulaattorisolun turvallistamiseksi suunniteltiin turva-alue, joka esti henkilön kulun vaaralliselle alueelle koneen ollessa käynnissä. Turva-alueelle kulkemista varten käyttäjän tuli pysäyttää kone operointipaneelilta. Turva-alueelta palatessaan henkilön tulee kuitata itsensä poistuneeksi alueelta. Käyttäjän tulee itse huolehtia, ettei alueella ole muita henkilöitä ennen kuitaamista. Tämän jälkeen koneet voidaan käynnistää uudelleen.

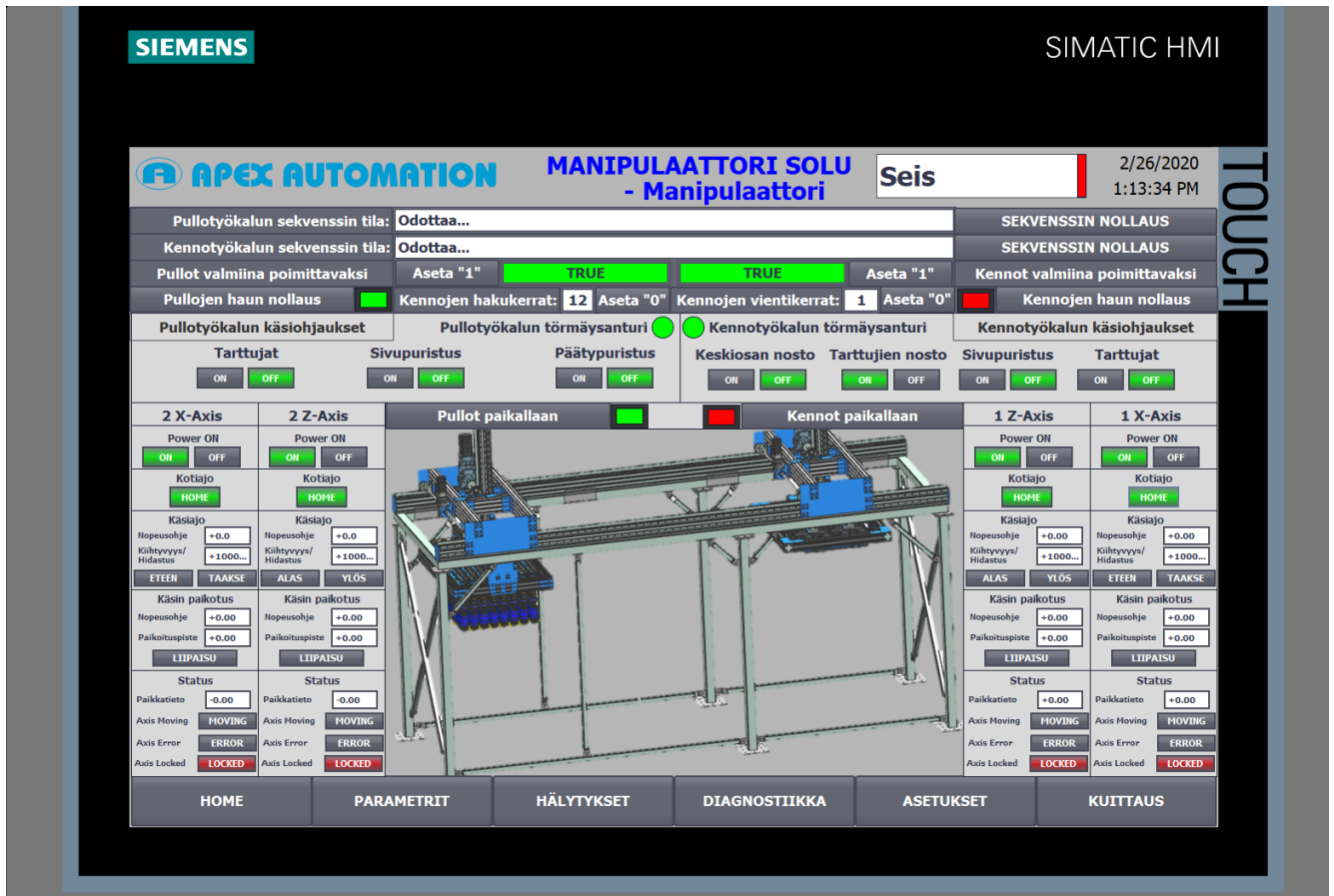
5.5 Optimointi

Virtuaaliseen käyttöönottoon kuului laitteiston optimointi, sillä asiakkaan kanssa oli sovittu minimikapasiteetti, joka laitteiston tuli pakata tuntia kohden. Jotta kapasiteetti saatiin maksimoitua, tuli koneiden toiminnot optimoida. Optimointiin kuului manipulaattorin akseleiden liikkeiden optimointi, jolloin manipulaattori kykeni tekemään yhtäaikaista liikkeitä eri akseleilla. Manipulaattorille lisättiin operointipaneeliin parametrisivu, josta käyttäjä pystyi hallitsemaan akseleiden lukituksia ja ennakoajoja, joilla voitiin nopeuttaa tehtävien suoritusta. Parametrisivu oli kuitenkin asetettu salasanan taakse, johon ainoastaan osaavalla henkilöstöllä oli pääsyoikeus. Tällä vältettiin virheellisten parametrien asettamista koneelle. Kuvassa 19 on esitetty manipulaattorille suunniteltu parametrisivu, josta parametrejä pystyi muuttamaan.



KUVA 19. Manipulaattorin parametointiin tarkoitettu ohjausikkuna.

Parametrisivun lisäksi manipulaattorille tehtiin kaksi ohjaussivua ohjauspaneeliin. Toinen sivuista oli yksinkertaistettu, ja sillä koneen käyttäjä pystyi toimimaan konetta käyttäessä. Toinen sivuista taas oli kehittyneempi ohjaussivu, jossa voitiin puuttua sekvenssien toimintaan sekä ohjata manipulaattoria koneen ollessa käsiäjoitilassa. Tämä sivu oli varustettuna salasanalla, jottei koneen tavallisessa käytössä operaattori muuttaisi sekvenssin asetuksia. Kuvassa 20 on esitelty manipulaattorin ohjaukseen tarkoitettu kehittyneempi ohjausikkuna.



KUVA 20. Manipulaattorin kehittynyt ohjausikkuna.

Manipulaattorin optimoinnin lisäksi solusta optimoitiin myös muut koneet. Lavanpinontakoneen sekä siirtovaunun toiminnassa oli tärkeää, että manipulaattorin pakkaukselle ei koidu viiveitä lavojen syötössä. Tämän vuoksi lavoja ajettiin puskuriin valmiiksi manipulaattorille. Lavanpinoajalle mahtui kerrallaan useampi lava, joita se automaattisesti purki manipulaattorille päin, kun lavat tulivat pakkauksesta valmiiksi. Lavanpinoajalle lavat kuljetetaan trukilla, josta koneen käyttäjän tulee huolehtia.

Kennokuljetinlaitteiston tuli huolehtia siitä, että manipulaattorilla riitti pakkausasemassa kennoja. Kone optimoitiin siten, että sillä kyettiin siirtämään kennopaketteja puskuriin pakkausaseman eteen. Kennojen loputtua pakkausasemasta kuljetinlaitteisto siirsi uuden kennopaketin pakkausasemaan, jotta pakkaus kykeni jatkumaan keskeytyksettä.

6 YHTEENVETO

Opinnäytetyö tehtiin Apex Automation Oy:n toiveesta tutkia logiikkaohjelman simulointia 3D-mallinnuksen avulla. Ohjelmistoja varten oli lisenssit hankittu jo ennen opinnäytetyön alkamista. Yrityksellä ei ollut aikaisempaa kokemusta 3D-mallin simuloinnista, vaikkakin logiikkaohjelmien simuloinnista osaamista löytyy. Yrityksellä oli aikaisempaa kokemusta PLCSIM Advancedin käytöstä, mikä auttoi ohjelman käytössä opinnäytetyössä.

NX-ohjelmiston asentaminen ja käyttöönotto sujui sulavasti hyvien asennusdokumenttien ansiosta. Lisenssipalvelimen käyttöönotossa ongelmia koitui serveritietokoneen palomuurin asetuksista, jotka estivät lisenssien pyynnit käyttäjiltä verkossa. Palomuurin asetusten muuttaminen korjasi ongelman, minkä jälkeen käyttäjät pystyivät ottamaan lisenssin itselleen haltuun serveriltä. MCD-sovellusta varten suoritettiin Siemensin Sitrain virtuaalisen käyttöönottoon liittyvän koulutuksen, joka auttoi virtuaalisessa suunnittelussa. Koulutus antoi hyvät lähtötiedot opinnäytetyölle sekä auttoi projektin aloituksessa. Opinnäytetyön aikana Siemensiltä sai hyvin tukea, jos MCD:n tai SIMIT:n käytössä ilmeni ongelmia.

Virtuaalinen suunnittelu mallinnuksen avulla auttoi automaatio suunnittelijan näkökulmasta huomattavasti hahmottamaan koneen toimintaa ja täten helpotti ohjelmointia. MCD:llä tehty mallinnus vaati kuitenkin huomattavan suuren määrän aikaa, jotta malli oli riittävän hyvä virtuaalisen käyttöönoton aloittamiseen. Itse koin opinnäytetyön aikana, että MCD:llä saa tehtyä pienen mallin nopeasti, jos mallissa ei ole paljoa yksityiskohtia ja simuloitavat liikkeet ovat yksinkertaisia. Mallin ollessa monimutkainen ja sisältäessä päällekkäisiä laskentaketjuja, muuttuu simulointimalli erittäin raskaaksi. Tällöin mallinnukseen tulee panostaa enemmän, jolloin aikaa mallinnukseen menee merkittävästi enemmän. Tämä koitui ongelmaksi opinnäytetyön aikana. Simuloitavaa mallia ei saatu rakennettua riittävän yksinkertaiseksi, mikä johti suorituskykyongelmiin.

Suorituskykyongelma saatiin kuitenkin hallintaan, kun otimme käyttöön SIMIT-työkalun MCD:n rinnalle. Ongelma vaikutti liittyvän MCD:n ja PLCSIM Advancedin väliseen tiedon synkronointiin. Lisäämällä SIMIT-ohjelman toteutettiin ohjelmien välinen synkronointi SIMIT:ssä. SIMIT-ohjelma toimi näin ollen masterina tiedonsiirrolle simuloitavan logiikan ja 3D-mallin välillä. Kyseinen ongelma on kuitenkin merkittävä, sillä MCD:tä markkinoidaan toimivaksi PLCSIM Advanced -ohjelman kanssa il-

man SIMIT-ohjelmistoa. Näin ei kuitenkaan tässä opinnäytetyössä voitu toimia. Ongelmaa ei oltu tietojeni mukaan havaittu Siemensin puolesta aikaisemmin, vaan ongelma ilmeni vasta tämän työn seurauksena.

Kokonaisuudessaan suunnitteluun kuluva aika en voi suoraan arvioida, koska todellista käyttöönottoa manipulaattorisolulle ei ennätetty tehdä ennen opinnäytetyön valmistumista. Lähtökohtaisesti kuitenkin virtuaalisesta käyttöönotosta jäi positiivinen kuva, sillä sen avulla saatiin paikannettua monia virheitä ohjelmasta ja laitteen toiminta voitiin katsoa realistiseksi. En itse kuitenkaan ole täysin samaa mieltä Visual Components-yrityksen kanssa siitä, että virtuaalinen käyttöönotto vähentäisi projektin suunnitteluun vaadittavaa aikaa. Todellista käyttöönottoa virtuaalinen käyttöönotto tulee todennäköisesti merkittävästi helpottamaan ja nopeuttamaan. Asiakkaalle koituvia kuluja virtuaalinen käyttöönotto tulee hyvin todennäköisesti laskemaan, sillä käyttöönottoaikaa saadaan lyhennettyä. Virtuaalisen käyttöönoton avulla asiakas voi myös minimoida tuotantokatkoksen, kun laitteisto on testattu jo etukäteen.

Visual Components oli arvioinut, että projektin suunnittelussa säästetään huomattava määrä aikaa, jos projektissa hyödynnetään virtuaalista käyttöönottoa. Yritykset, jotka toimivat laitevalmistajina, kokevat hyödyn varmasti suurena, sillä tuote voidaan testata ennen asiakkaalle luovutusta. Pelkän automaatio-suunnittelutoimiston perspektiivistä voidaan katsoa tilannetta hiukan eri näkökulmasta, koska tuotteet ovat yleensä uniikkeja eikä tuotetta ole välttämättä tarkoitus kehittää luovutuksen jälkeen. Tuotteiden kehittämiseen NX-ohjelmisto vaikutti olevan kuitenkin erittäin potentiaalinen, sillä siitä löytyy lukuisia mahdollisuuksia kerätä dataa kehitettävästä mallista ja malli voidaan jalostaa todella todenmukaiseksi.

Suunnitteluun vaadittava aika riippuu paljon mallinnettavasta koneesta tai prosessista. Siemensin koulutusmateriaalissa virtuaaliseen käyttöönottoon on esitetty kuvaaja, jossa näytetään suunnitteluun kuluva aika tehdyn mallinnuksen suhteen. Kuvaajan mukaan noin kahdeksankymmentä prosenttia tehdystä mallinnuksessa saavutetaan kahdessakymmenessä prosentissa projektin mallinnukseen kuluvasta ajasta, kun taas loput kahdeksankymmentä prosenttia ajasta menee mallin hienosäätöön. (DI-VIRTCOM 2018.) Jos mallinnusta ei ole tarvetta tehdä koko koneesta tai prosessista, voidaan mallintaa ainoastaan osa, jolloin aikaa ei mene läheskään yhtä paljon kuin koko systeemin mallinnukseen. Tällaisissa tapauksissa virtuaalisesta käyttöönotosta saadaan mielestäni suurin hyöty pienimmällä määrällä työtä.

Opinnäytetyössä huomasin, että SIMIT-ohjelmiston avulla mallinnuksen teossa saadaan säästettyä huomattava määrä aikaa, sillä erillistä signaaleiden käsittelyä ei tarvitse määritellä MCD:ssä. SIMIT osaa

lukea MCD:n tietokannasta objekteja ilman, että niistä olisi määritetty erillisiä signaaleita. Tällöin esimerkiksi anturin määrittäminen vaatii ainoastaan Collision Sensor -määrittelyn MCD:hen. SIMIT:ssä signaalien käsittelyminen voidaan tehdä Microsoftin Excel-työkalulla, jolloin voidaan hyödyntää massasuunnittelua. Tämä nopeuttaa simulointimallin mallintamista. Miinuksena SIMIT:ssä on, ettei sen avulla voida lukea logiikasta datalohkojen tai muistialueen muuttujia. SIMIT lukee ainoastaan logiikan tuloja ja lähtöjä. Jos mallinnuksessa halutaan käyttää apumuuttujia, tulee ne siirtää logiikan lähdöissä MCD:lle. Tästä voi johtua, että ohjelmaan joudutaan lisäämään simulointia varten ylimääräisiä muuttujia, jotka sotkevat ohjelmaa. Tällaista menettelyä voi tarvita kuitenkin ainoastaan poikkeustilanteissa, minkä vuoksi ongelmaa ei voida pitää suurena.

SIMIT-ohjelmisto tuo myös kattavan määrän diagnostiikkaa ja parantaa käyttöjen simulointia valmiiden lohkokirjastojen avulla. Koen että, mikäli tulevaisuudessa halutaan hyödyntää virtuaalista käyttöönottoa Siemens laiteympäristössä, olisi suositeltavaa käyttää SIMIT-ohjelmistoa MCD:n rinnalla. Mielestäni MCD-sovellus on toiminnallisuuksiltaan monipuolinen ohjelmisto simulointiympäristön luomiseen, ja sen avulla simuloinnilla saavutetaan hyötyjä todellista käyttöönottoa varten.

LÄHTEET

- Ahola, J. M. 2011. Moniteknisen tuotteen virtuaalisuunnittelun konsepti. Julkaisija: VTT. Saatavissa: <https://docplayer.fi/4015549-Moniteknisen-tuotteen-virtuaalisuunnittelun-konsepti-tyokoneiden-tuote-tiedonhallinta-seminaari-30-3-2011-jari-m-ahola-vtt.html>. Viitattu 8.10.2019.
- Ahola, J.M., Hovila, J., Karhunen, E., Nevala, K., Nevala, T. & Schäfer, T. 2011. Moniteknisen pien-sarjatuotteen digitaalinen tuoteprosessi. Julkaisija: VTT. Saatavissa: <https://docplayer.fi/12272317-Moniteknisen-piensarjatuotteen-digitaalinen-tuoteprosessi.html>. Viitattu 26.9.2019.
- Ajo, R., Hakonen, S., Harju, H., Järvi, J., Kaskes, K., Lenardic, E., Niukkanen, E., Nurminen, T., Ritala, P., Tolppanen, M. & Tommila, T. 2001. Laatu automaatioissa. Julkaisija: Suomen automaatioseura Ry. Saatavissa: <https://www.automatioseura.fi/site/assets/files/1367/laatuautomaatioissa.pdf>. Viitattu 28.2.2020.
- Apex Automation Oy 2020. Yritys. Saatavissa: <https://www.apexautomation.fi/>. Viitattu 13.3.2020.
- DI-VIRTCOM. 2018. SITRAIN - Siemens. Virtuaalinen käyttöönottokoulutusmateriaali. Viitattu 28.2.2020. Materiaali ei julkisesti saatavilla.
- Gantt. 2020. What is a Gantt Chart? Saatavissa: <https://www.gantt.com/>. Viitattu 22.1.2020.
- Hambling, B., Morgan, P., Samaroo, A., Thompson, G. & Williams, P. 2007. Software Testing. BCS. Saatavissa: <https://www.bcs.org/upload/pdf/software-testing-chapter1.pdf>. Viitattu 14.10.2019.
- Ideal PLM. 2020. Mechatronics Concept Designer. Saatavissa: https://ideal.fi/uPage/Mechatronics_Concept_Designer. Viitattu 27.2.2020.
- Liu, Z., Suchold, N. & Diedrich, C. 2012. Virtual Commissioning of Automated Systems. (Eds.) Institut für Automation und Kommunikation (ifak), Magdeburg, Germany. Saatavissa: http://cdn.intechopen.com/pdfs/37992/InTech-Virtual_commissioning_of_automated_systems.pdf. Viitattu 8.10.2019.
- Merhof, J. 2018. Mechatronics Concept Designer. Julkaisija: Siemens. Saatavissa: https://www.plm-europe.org/admin/presentations/2018/2216_PLMEurope_30.10.18-08-00_JOCHEN-MERHOF_SIEMENS_PLM_SOFTWARE_mechatronics_concept_designer.pdf. Viitattu 8.10.2019.
- NG 2019. NX Mechatronic Concept Designer (MCD). Saatavissa: <https://www.ng.engineering/en-PL/detail/nx-mechatronic-concept-designer-mcd-33.html>. Viitattu 14.10.2019.
- NX help. 2019. NX help -sivusto. Siemens. Saatavissa: https://docs.plm.automation.siemens.com/toc/nx/1847/nx_help#uid:index. Viitattu 17.10.2019.
- NX 2020. NX-esittelysivusto. Siemens. Saatavissa: <https://www.plm.automation.siemens.com/global/en/products/nx/>. Viitattu 27.2.2020.
- PROFIsafe System Description. 2010. Automation. Saatavissa: https://www.automation.com/pdf_articles/profinet/PROFIsafe_system_description_v_2010_English.pdf. Viitattu 25.2.2020.

- Rosen, R., Von Wichert, G., Lo, G. & Bettenhausen, K.D., 2015 About the importance of autonomy and digital twins for the future of manufacturing. IFAC-PapersOnLine. Saatavissa: <https://www.sciencedirect.com/science/article/pii/S2405896315003808>. Viitattu 14.10.2019.
- SFS-EN 62061. Koneturvallisuus. Turvallisuuteen liittyvien sähköisten, elektronisten ja ohjelmoitavien elektronisten ohjausjärjestelmien toiminnallinen turvallisuus. 2010. Helsinki: Suomen standardisoimisliitto SFS.
- SFS-EN ISO 13849-1. Koneturvallisuus. Koneturvallisuus. Turvallisuuteen liittyvät ohjausjärjestelmien osat. Osa 1: Yleiset suunnitteluperiaatteet. 2016. Helsinki: Suomen standardisoimisliitto SFS.
- Siemens GRAPH. 2018. Automation of Sequential Processes with GRAPH in the TIA Portal for S7-1500. Saatavissa: https://cache.industry.siemens.com/dl/files/822/109759822/att_963255/v3/109759822_GRAPH_TIAPortal_V15_V1.0_en.pdf. Viitattu 27.2.2020.
- Siemens PLM. 2016. Licensing user guide. Saatavissa: https://docs.plm.automation.siemens.com/data_services/resources/nx/11.0.1/nx_help/custom/en_US/ugslicensing/SPLMLicensing_user_guide.pdf. Viitattu 27.2.2020.
- Silvennoinen, P. 2016. Tuotetiedon hallinta: PIM, PLM ja PDM – mitä ne oikeasti tarkoittavat? Saatavissa: <https://www.crasman.fi/blogi/tuotetiedon-hallinta-pim-plm-ja-pdm-mit%C3%A4-ne-oikeasti-tarkoittavat>. Viitattu 8.10.2019.
- SIMATIC S7 PLCSIM Advanced: Crosslink API. 2018. Siemens. Saatavissa: <https://support.industry.siemens.com/cs/document/109754823/digitalization-with-tia-portal%3A-virtual-commissioning-by-basic-behavior-modeling-with-simatic-s7-plcsim-advanced?dti=0&lc=en-WW>. Viitattu 27.2.2020.
- SIMATIC S7-PLCSIM Advanced: Co-Simulation via API. 2017. Siemens. Saatavissa: <https://support.industry.siemens.com/cs/document/109739660/simatic-s7%E2%80%91plcsim-advanced%3A-co%E2%80%91simulation-via-api?dti=0&lc=en-WW>. Viitattu 27.2.2020.
- SIMATIC S7-SCL V5.3 for S7-300/400. 2005. Siemens. Saatavissa: https://cache.industry.siemens.com/dl/files/793/5581793/att_66783/v1/SCL_e.pdf. Viitattu 27.2.2020.
- SIMIT Simulation Platform. 2019. Siemens. Saatavissa: <https://assets.new.siemens.com/siemens/assets/api/uuid:4ce72796-3757-493c-8afe-e1fefb988025/version:1565169240/simitsimulationplatform-datasheet2019-1en-144.pdf>. Viitattu 22.1.2020.
- Suomen Automaatioseura ry. 2007. Automaatio suunnittelun prosessimalli. Helsinki. Saatavissa: https://www.automaatioseura.fi/site/assets/files/1367/automaatio suunnittelun_prosessimalli.pdf. Viitattu 1.11.2019.
- Totally Integrated Automation Portal. 2020. Siemens. Saatavissa: <https://new.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html>. Viitattu 25.3.2020.
- Visual Components. 2016. What is virtual commissioning? Saatavissa: <https://www.visualcomponents.com/insights/articles/increasing-control-software-quality-with-virtual-commissioning/>. Viitattu 8.10.2019.