

Juha Salmenkangas

## **Prima Power Virtual Machine**

Opinnäytetyö

Kevät 2016

SeAMK Tekniikka

Automaatiotekniikan tutkinto-ohjelma



SEINÄJOEN AMMATTIKORKEAKOULU  
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

SEINÄJOEN AMMATTIKORKEAKOULU

## Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikan yksikkö

Tutkinto-ohjelma: Automaatiotekniikan tutkinto-ohjelma

Suuntautumisvaihtoehto: Koneautomaatio

Tekijä: Juha Salmenkangas

Työn nimi: Prima Power Virtual Machine

Ohjaaja: Jarkko Pakkanen

Vuosi: 2016

Sivumäärä: 50

Liitteiden lukumäärä: -

---

Opinnäytetyön tavoitteena oli rakentaa toimiva yhteys Visual Components 3DCreate -simulointiohjelman ja Beckhoff TwinCAT 3 -ohjauksen välille. Tavoitteena oli yhteyden avulla ohjata Prima Power SGe6 -levyntyöstökoneen 3D-simulointimallia fyysisen SGe6-koneen ohjauksella. Työn toimeksiantajana toimi Kauhavaalla sijaitseva Finn-Power Oy. Virtuaalikonetta tullaan hyödyntämään virtuaalisessa koulutus- ja testausympäristössä.

Työssä oli kartoitettava mitä virtuaalikoneen valmistamiseen vaaditaan. Tarvekartoituksen jälkeen rakennettiin 3D-simulointimalli 3DCreate-ohjelmalla ja tehtiin oikean koneen TwinCAT 3 -ohjaukseen tarvittavat muutokset. Ohjaus yhdistettiin 3D-simulointimalliin 3DCreateen PLC Add-on -lisäosalla.

Tietoa haettiin kirjallisuudesta ja internetistä. Lisäksi hyödynnettiin toimeksiantajan omia dokumentaatioita. Työssä käytettiin apuna koneiden valmiita 3D-malleja sekä oikeiden koneiden PLC-ohjauksia.

Avainsanat: 3D-mallinnus, simulointi, PLC, virtuaalinen käyttöönotto, virtuaalikone

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

## **Thesis abstract**

Faculty: School of Technology

Degree programme: Automation Engineering

Specialisation: Machine Automation

Author: Juha Salmenkangas

Title of thesis: Prima Power Virtual Machine

Supervisor: Jarkko Pakkanen

Year: 2016

Number of pages: 50

Number of appendices: -

---

The purpose of this thesis was to create a functional connection between the Visual Components 3DCreate simulation software and Beckhoff TwinCAT 3 PLC software. The goal was to control the 3D simulation model of the Prima Power SGe6 punching-shearing workcell with the real machine PLC control program. The thesis was commissioned by Finn-Power Oy. The virtual machine will be used in a virtual education and testing environment.

The thesis was started by finding out the requirements for creating the virtual machine. After that the 3D model was created by the 3DCreate software and the necessary modifications were made to the TwinCAT 3 PLC program. The PLC connection was made by the 3DCreate PLC Add-on.

Information was searched from literature and on the Internet. In addition, the company's own documentation was used. The existing 3D models of the machines and the real machine control program were used in this thesis.

Keywords: 3D modeling, simulation, PLC, virtual commissioning, virtual machine

# SISÄLTÖ

Opinnäytetyön tiivistelmä.....	2
Thesis abstract.....	3
SISÄLTÖ.....	4
Kuvioluettelo.....	6
Käytetyt termit ja lyhenteet .....	8
<b>1 JOHDANTO .....</b>	<b>9</b>
1.1 Työn tausta .....	9
1.2 Työn tavoite .....	9
1.3 Työn rakenne .....	10
1.4 Prima Power ja Finn-Power Oy.....	11
<b>2 SIMULOINTI .....</b>	<b>12</b>
2.1 Mitä simulointi on? .....	12
2.2 Miksi simulointi on kannattavaa?.....	12
2.3 Simuloinnin hyödyt ja haitat .....	13
2.4 Visual Components ja 3DCreate .....	14
<b>3 PLC-LOGIIKAT .....</b>	<b>15</b>
3.1 Logiikkaohjausten historiaa.....	15
3.2 Logiikkatyypit.....	15
3.2.1 Ohjelmoitavat logiikat.....	16
3.2.2 Soft-PLC .....	17
3.3 Beckhoff ja Beckhoff TwinCAT 3 XAE.....	18
<b>4 VIRTUAALINEN KÄYTTÖÖNOTTO.....</b>	<b>20</b>
4.1 Mitä on virtuaalinen käyttöönotto?.....	20
4.2 Virtuaalisen käyttöönoton hyödyt .....	21
4.3 Visual Components PLC Add-on.....	22
4.4 Tulus Cell .....	23
4.5 NC Express e <sup>3</sup> .....	24
4.6 Teollisuus 4.0.....	25
<b>5 TYÖN TOTEUTUS.....</b>	<b>27</b>
5.1 Tarvekartoitus .....	27

5.2	Kokoonpano ja laitteistot .....	29
5.2.1	Paikallinen työasema .....	30
5.2.2	Työasema pilvipalvelussa .....	31
5.3	Virtuaalikoneen toimintaperiaate .....	32
5.4	Simulointimallin rakentaminen.....	33
5.4.1	Geometrioiden luominen .....	34
5.4.2	Signaalien luominen.....	35
5.4.3	Parametrien luominen .....	36
5.4.4	PythonScriptin lisääminen.....	36
5.5	Ohjauksen muutokset .....	38
5.6	TwinCAT-ohjauksen kytkeminen simulointimalliin.....	39
5.6.1	PLC-yhteyden luominen.....	40
5.6.2	Ryhmän luominen .....	40
5.7	Virtuaalikoneen testaus .....	43
6	TULOKSET JA JATKOKEHITYS .....	46
7	POHDINTA .....	47
	LÄHTEET .....	48

## Kuvioluettelo

Kuvio 1. Finn-Power Oy Kauhava.....	11
Kuvio 2. Ohjelmoitavan logiikan perustoimintaperiaate.....	17
Kuvio 3. TwinCAT 3 XAE -moduulit .....	19
Kuvio 4. Samalla ohjauksella voidaan ohjata virtuaalikonetta ja oikeaa konetta ...	21
Kuvio 5. Automaatioprojekti ilman virtuaalista käyttöönottoa ja sen kanssa .....	22
Kuvio 6. Tulus Cellin käyttöliittymä.....	23
Kuvio 7. NC Express e <sup>3</sup> -toiminnot.....	24
Kuvio 8. Teollinen Internet kytkee laitteet ja ajoneuvot internetin välityksellä. ....	25
Kuvio 9. Messujen nykyinen tila verrattuna tilaan virtuaalikoneen kanssa .....	27
Kuvio 10. Virtual Demo Center -ständi.....	28
Kuvio 11. Alkuperäinen suunnitelma kokoonpanosta.....	29
Kuvio 12. Virtuaalikoneen kokoonpano.....	30
Kuvio 13. Paikallinen työasema .....	31
Kuvio 14. Työasema pilvipalvelussa .....	32
Kuvio 15. Virtuaalikoneen toimintaperiaate .....	33
Kuvio 16. Geometrian valinta piirrepuusta .....	34
Kuvio 17. Move Sets To Feature -valinta .....	35
Kuvio 18. Signaalien luominen Behavior-välilehdeltä.....	35
Kuvio 19. Parametrien luominen Parameter-välilehdeltä .....	36
Kuvio 20. SGe-peruskoneen PythonScript-koodi.....	37

Kuvio 21. Parametrin arvo ohjaa akselia lineaarisesti.....	37
Kuvio 22. Signaalien reititys.....	38
Kuvio 23. VirtualModel-muuttujalista.....	39
Kuvio 24. TwinCAT RUN-tilassa .....	40
Kuvio 25. PLC-yhteyden luominen Add PLC Connection -painikkeella.....	40
Kuvio 26. Ryhmän luominen muuttujien paritusta varten .....	41
Kuvio 27. Item Configuration -ikkuna .....	41
Kuvio 28. XML-tiedosto muuttujien ja signaalien pareista.....	42
Kuvio 29. Paritetut muuttujat ja signaalit .....	43
Kuvio 30. RUN-tila asetetaan oikeasta yläkulmasta ja PLC-yhteys Connect- painikkeella Actions-välilehdeltä .....	44
Kuvio 31. Tulus Auto Mode -tilassa ja työjono aktiivisena.....	45
Kuvio 32. Active Sheet -välilehdeltä voidaan tarkkailla ohjelman etenemistä .....	45

## Käytetyt termit ja lyhenteet

<b>CAD</b>	Computer-aided design eli tietokoneavusteinen suunnittelu. Käytetään tietokoneella graafiseen 2D- ja 3D-suunnitteluun ja dokumentointiin.
<b>FMS</b>	Flexible manufacturing system. Joustava valmistusjärjestelmä, joka on täysin automatisoitu ja pystyy toimimaan itsenäisesti reagoimalla toiminnassa tapahtuviin muutoksiin.
<b>HMI</b>	Human machine interface. Koneen käyttöliittymä, jonka avulla koneenkäyttäjä käyttää konetta.
<b>I/O</b>	Input/Output. Tulot ja lähdöt tarkoittavat ohjelmoitavan logiikan ulkoisia liitäntöjä, joiden avulla siirretään dataa eri laitteiden välillä.
<b>Nestaus</b>	CAM-ohjelmassa oleva prosessi, jolla pyritään sovittamaan mahdollisimman paljon leikattavia osia yhdelle levyille. Tällä pyritään tehostamaan tuotantoa ja minimoimaan hukkamateriaalin määrää.
<b>PLC</b>	Programmable logic controller eli ohjelmoitava logiikka. Tarkempi kuvaus kappaleessa 3.
<b>Python</b>	Monipuolinen ja tulkattava ohjelmointikieli, jota käytetään esimerkiksi Visual Componentsin 3DCreate-ohjelmassa.
<b>Virtuaalikone</b>	Oikeasta koneesta tehty simuloitu virtuaalinen kaksonen. Ohjataan oikean koneen ohjauksella, joten toimii samalla tavalla.



# 1 JOHDANTO

## 1.1 Työn tausta

Opinnäytetyön aiheena oli valmistaa virtuaalikone, jossa 3D-simulointimallia ohjataan suoraan PLC-logiikalla. Opinnäytetyön toimeksiantajana toimi Kauhavalla sijaitseva Finn-Power Oy. Yritys kuuluu Italialaiseen Prima Industrie -konserniin, joka on yksi maailman johtavista levytyökoneisiin ja -järjestelmiin erikoistuneista yrityksistä. Kauhavalla sijaitsee myös yrityksen Teknologia- ja koulutuskeskus, jossa työskentelee toistakymmentä applikaatiospesialistia ja asiakaskouluttajaa. Työstökoneita koulutetaan pääsääntöisesti asiakkaiden toimitiloissa uusien konetoimitusten yhteydessä tai Kauhavan tehtaalla. Tällä hetkellä testaus- ja koulutusympäristö vaatii käytännössä fyysisen koneen, jolla koneelle tehdyt työstöohjelmat testataan. Virtuaalikoneen avulla testauksen ja koulutuksen voisi järjestää ilman oikeaa konetta. Kauhavalta käsin organisoidaan vuosittain kymmeniä teollisuusmessuja, joissa yrityksen työstökoneita on esillä. Ison tilan tai budjetin puutteen vuoksi oikea kone voitaisiin korvata virtuaalikoneella, jolla koneen toimintaa voitaisiin esitellä.

## 1.2 Työn tavoite

Työn tavoitteena oli selvittää taustat ja tarpeet virtuaalikoneen suunnittelua varten. Suunnittelussa tuli ottaa huomioon ohjelmien rajoitukset, yrityksen tarpeet ja virtuaalikoneen käyttökohteet. Tavoitteena oli luoda virtuaalikone, eli PLC-logiikalla ohjattu 3D-simulointimalli yrityksen valmistamasta SGe6-levyntyöstökoneesta. Yrityksen tavoitteena oli saada täysin oikeaa konetta vastaava virtuaalikone, jota voitaisiin hyödyntää tuotekehityksessä, myynnissä, koulutuksessa ja messuesittelyssä. Tässä opinnäytetyössä tarkastellaankin vain yhtä konetta yrityksen laajasta valikoimasta. Työssä tarkastellaan myös pilvipalvelujen tarjoamia mahdollisuuksia. Pilvipalvelut tarjoaisivat vaihtoehdon paikallisesti asennettaville ohjelmille ja päivityksille.

Ennen varsinaista toteutusta tavoitteena oli toteuttaa interaktiivinen esitys virtuaalikoneen käyttökohteista, hyödyistä sekä toiminnasta. Esitystä käytettiin apuna kyselyssä, jossa kartoitettiin yrityksen Pohjois-Amerikan, Puolan ja Venäjän maajohtajien kiinnostusta virtuaalikoneeseen. Kyselyn tuloksia käytettiin kartoittamaan virtuaalikoneen tarpeellisuutta ja sen kehittämideoita. Tuloksia käytettiin myös arvioimaan tullaanko virtuaalikoneita valmistamaan jatkossa myös muista yrityksen valmistamista koneista tämän opinnäytetyön jälkeen.

Lopullisena tavoitteena oli liittää SGe6-lävistyskoneen TwinCAT 3 -ohjelmalla tehty ohjaus Visual Componentsin 3DCreate-ohjelmassa olevaan 3D-malliin. Ohjauksen liittämisessä käytettiin 3DCreate-ohjelman PLC Add-on -lisäosaa. PLC Add-onin avulla PLC-ohjelmasta voidaan kytkeä tarvittavat muuttujien signaalit suoraan 3D-malliin luotuihin signaaleihin. Näin logiikasta lähtevät signaalit ohjaavat suoraan 3D-mallia. Virtuaalikoneen käyttämiseen vaaditaan myös NC Express e<sup>3</sup>, jolla luodaan työstettävistä kappaleista nestausohjelma ja NC-ohjelma. NC-ohjelmat ajetaan Tulus-ohjelmalla. Tulus ohjaa Twincat 3 -ohjelmaa ja Twincat 3 -ohjelma ohjaa 3DCreatessa olevaa 3D-mallia.

### **1.3 Työn rakenne**

Opinnäytetyö alkaa johdannolla, jossa esitellään työn taustat ja tavoitteet sekä esitellään työn toimeksiantaja, Finn-Power Oy. Teoriaosassa kerrotaan simuloinnista ja sen tarpeellisuudesta, PLC-logiikoista ja Soft-PLC-logiikoista sekä virtuaalisesta käyttönotosta ja Teollisesta Internetistä. Teoriaosassa esitellään myös työssä käytetyt ohjelmat, Visual Components 3DCreate ja PLC Add-on, Beckhoff TwinCAT 3, Tulus Cell ja NC Express e<sup>3</sup>. Työvaiheessa kerrotaan tarvekartoituksesta sekä virtuaalikoneen toiminnasta ja kokoonpanosta. Työvaiheessa kerrotaan myös, kuinka virtuaalikone rakennettiin ja kuinka PLC-yhteys luotiin. Työn lopussa esitellään tulokset ja ideoita jatkokehitystä varten. Tämän jälkeen pohditaan työn onnistumista ja saavutettiinko työn tavoitteet. Lisäksi käydään läpi työssä esiintyneet ongelmat.

## 1.4 Prima Power ja Finn-Power Oy

Kauhavalla vuonna 1969 perustettu Finn-Power Oy on osa Prima Industrie -konsernia ja sen Prima Power -osastoa. Finn-Power Oy saavutti 1970-luvun loppupuolella maailman markkinajohtajuuden sen ensimmäisillä tuotteilla, hydraulisilla letkuliitinpuristimilla. Prima Industrie aloitti toimintansa 3D-laserroboteilla, joilla se nousi johtavaan asemaan 1980-luvulla. Vuonna 1983 Finn-Power Oy esitteli hydraulisen levytyökeskuksen ja siirtyi näin toiminnassaan enemmän levytyötekniikkaan. 1990-luvun alussa Finn-Powerin integroitu lävistys-kulmaleikkuu-konsepti teki maailmanlaajuisen läpimurron joustavien valmistusjärjestelmien keskeisenä työstösoluna. Se tuli osaksi vuonna 1994 esiteltyä Night Train FMS -järjestelmää, joka on edelleen johtavia FMS-konsepteja. Samaan aikaan 1990-luvulla Prima Industrie laajensi toimintaansa 2D-lasermarkkinoille yritysostojen kautta ja jatkoi kasvuaan 2000-luvulla. Vuonna 2008 Prima Industrie hankki Finn-Power Oy:n tytäryhtiöineen. Maaliskuussa 2011 otettiin käyttöön kaikille tuotteille ja palveluille yhteinen tunnus, Prima Power. Prima Powerin Kauhavan yksikkö jatkaa toimintaansa nykyäänkin kuitenkin Finn-Power Oy -nimellä. (Prima Power 2015a.)

Prima Power on vuonna 2016 neljän laser- ja levytyötekniikan kärkiyrityksen joukossa. Prima Powerilla on toimintaa yli 70 maassa ja yhteensä kahdeksan tehdasta Suomessa, Italiassa, USA:ssa ja Kiinassa. Koko Prima Power -konsernissa työskentelee noin 1580 henkilöä, joista noin 330 työskentelee Finn-Power Oy:ssä Kauhavalla. Prima Powerin liikevaihto vuonna 2014 oli 350 miljoonaa euroa ja Finn-Power Oy:n 107 miljoonaa euroa. (Prima Power 2015b.)



Kuvio 1. Finn-Power Oy Kauhava  
(Prima Power 2012).

## 2 SIMULOINTI

### 2.1 Mitä simulointi on?

Simulointi on yksi tehokkaimmista päätöksentekoon vaikuttavista työkaluista tutkitessa monimutkaisia prosesseja ja järjestelmiä. Simulointi mahdollistaa erilaisten toimintojen laskelmia, analysointeja ja arviointeja, joita ei muuten olisi mahdollista toteuttaa. Simuloinnista on tullut lähes korvaamaton menetelmä ongelmanratkaisuun insinöörien, suunnittelijoiden ja johtajien keskuudessa yhä kilpailullisemmassa maailmassa. (Shannon 1998, 7.)

Simulointi tarkoittaa terminä jonkun järjestelmän jäljittelyä. Erilaisia simulointimalleja näkyy päivittäisessä käytössä esimerkiksi säätiedotuksissa, joissa simuloidaan säätilan muutoksia päiviä etukäteen, ja videopeleissä, joissa simuloidaan erilaisia aktiiviteetteja autoilusta kaupunkisuunnitteluun. Simulointien ei aina tarvitse olla tietokonepohjaisia, vaan myös pienoisoratatie ja kauko-ohjattavat veneet ovat tuttuja esimerkkejä fyysisistä simuloinneista. Edellä mainituissa esimerkeissä on otettu huomioon jäljitelmän muuttuminen ajan suhteen, eli ne ovat dynaamisia simulointeja. Niillä on selkeä ero staattisiin simulointeihin, joita on esimerkiksi Las Vegasissa oleva jäljitelmä Pariisin Eiffel-tornista sekä erilaiset CAD-piirustukset tuotantotiloista tai koneista. Staattisissa simuloinneissa jäljitelmän muutosta tarkastellaan tietyllä ajan hetkellä, kun taas dynaamisissa simuloinneissa verrataan jäljitelmän muutosta ajan kuluessa. Tässä työssä keskitytään tietokonepohjaisiin eli dynaamisiin simulointeihin. (Robinson 2004, 2.)

### 2.2 Miksi simulointi on kannattavaa?

Simuloinnin tarvetta voidaan ajatella kolmesta eri näkökulmasta. Ensimmäiseksi mietitään onko simuloinnin käyttökohteen luonne sellainen, että simuloinnista on ylipäätään hyötyä. Simulointia kannattaa käyttää, jos käyttökohteen toiminnoissa on paljon vaihtelevuutta. Simuloinnilla voidaan testata kaikkia vaihtoehtoja, ja tulosten perusteella voidaan valita toimivin vaihtoehto. Simuloimalla voidaan esimerkiksi tes-

tata sairaalan toimivuutta vaihtelemalla potilaiden määrää tai tarkastella tuotantolinjan toimivuutta, kun yksi tuotantosolu lakkaa toimimasta. Toisena huomioonotettava asiana tarkastellaan simuloinnin hyötyjä. Simuloinnin avulla pyritään havainnollistamaan toimintoja visuaalisesti sekä helpottamaan ymmärrystä niiden toiminnasta ja kehittämisestä. Kolmantena huomioonotettavana asiana tarkastellaan simuloinnin mahdollisia haittoja. Täytyy pohtia saadaanko simulointiin käytettävillä resursseilla tarpeeksi suuri hyöty eli onko simulointi kannattavaa. Pohdittavaksi jää myös, saadaanko simuloinnista riittävän laaja ja yksityiskohtainen. (Robinson 2004, 4.)

### **2.3 Simuloinnin hyödyt ja haitat**

Perinteisillä staattisen laskennan menetelmillä on vaikea tarkastella monimutkaisia prosesseja, joissa yhden muuttujan muutos vaikuttaa kokonaisuuden toimivuuteen. Simulointimallia käyttämällä voidaan etukäteen selvittää prosessille optimaaliset parametrit, joilla toiminta on kaikkein tehokkainta ilman kalliita käytännön kokeiluja. Varmistamalla suurten investointien tai muutosten vaikutukset kokonaisuuden toimivuuteen simuloimalla vältetään tekemästä vääriä päätöksiä. Simulointimallia rakennettaessa toimintaprosessit on kartoitettava niin yksityiskohtaisesti, että simulointimallista saataisiin mahdollisimman realistinen. Projektin läpivienti lisää samalla siihen osallistuvien ymmärrystä oman organisaationsa operatiivisesta toiminnasta. (Delfoi 2013.)

Uuden tuotantolinjan tai koneen suunnittelu ja rakentaminen on iso investointi kaikille yritykselle. Jotta investoinnista saataisiin mahdollisimman suuri hyöty, on järkevää linjan suunnitteluvaiheessa valmistaa identtinen simulointimalli, jonka avulla voidaan analysoida sen toimintaa hyvin tarkasti. Simuloinnilla voidaan selvittää tuotannon mahdolliset pullonkaulat ja maksimoida tuottavuus jo ennen tuotannon käynnistämistä. Simuloimalla tehdyt törmäystarkastelut ja ulottuvuustarkastelut pienentävät tuotannossa syntyviä riskejä ja vahinkoja. (Visual Components 2016a.)

Simuloinnin avulla voidaan testata kuinka ja miksi tietyt oletukset ja tapahtumat vaikuttavat järjestelmän toimintaan. Simuloinnin ehdottomiin vahvuuksiin lukeutuu

mahdollisuus säätää ajan kulkua. Sen avulla voidaan sekunneissa nähdä jopa vuosien aikavälillä tapahtuvia toimintojen kehittymisiä ja muutoksia. Ajan kulkua voidaan myös hidastaa tutkittaessa esimerkiksi nopealiikkeisiä koneita tai simuloiteja, joissa tapahtuu monia asioita samaan aikaan. Simuloimalla voidaan myös tutkia uusia ja tuntemattomia tapauksia sekä saada vastauksia ”mitä jos” -kysymyksiin. (Shannon 1998, 7–8.)

Simulointimallinnusta voidaan ajatella taiteena, joka vaatii erikoiskoulutuksen ja tämän vuoksi sen harjoittajien taitotasot vaihtelevat laajasti. Simuloinnista saatavat hyödyt riippuvat siis puhtaasti mallin laadukkuudesta ja ohjelmoijan taidoista. Malliin syötettävien tietojen kerääminen ja simulointimallin rakentaminen on hyvin aikaa vievää, ja simuloinnilla saatavat tulokset ovat joskus hyvin kyseenalaisia. Ongelmana simuloinnin toimivuudessa on, että malli tekee täsmälleen sitä, mitä ohjelmoija on ohjelmoinut. Simulointimalli suorittaa toimintoja simulointiohjelmaan syötettyjen tietojen perusteella, ja suorittaa niiden perusteella mallille ennalta määrättyjä tehtäviä. Se ei osaa siis itse etsiä automaattisesti optimaalisimpia ratkaisuja, vaan ne ovat kiinni ohjelmoijan taidoista. (Shannon 1998, 8.)

## **2.4 Visual Components ja 3DCreate**

Visual Components tarjoaa ohjelmaratkaisuja 3D-simuloinnin eri tarpeisiin. Vuonna 1999 perustettu suomalais-amerikkalainen yritys aloitti toimintansa suunnittelemalla joustavan ja avoimen alustan robottien ja materiaalivirran simulointiin. Tuotteista löytyy nykyään viisi erilaisiin käyttötarkoituksiin soveltuvaa 3D-simulointiohjelmaa. Ohjelmilla voidaan toteuttaa esimerkiksi visuaalinen tehdassimulointi ja tarkka analyysi tuotantolinjan läpimenoajoista. (Visual Components 2016b.)

3DCreate-simulointiohjelma on hyvä valinta suurten tuotantolinjojen ja koneiden simulointiin, kun käytössä on valmiita CAD-malleja. Geometrioita voidaan tuoda suoraan ohjelmaan ja niitä voidaan ohjelman sisällä muokata sekä yhdistellä. Layoutin suunnittelu nopeutuu näin huomattavasti. Yksittäisistä laitemalleista koostettuun layoutkokonaisuuteen voidaan helposti luoda parametreja ja sääntöjä, joilla simuloinnista saadaan toimiva. Ohjelman sisäinen koodaus tapahtuu Python-ohjelmointikielillä. (Visual Components 2016c.)

## 3 PLC-LOGIIKAT

### 3.1 Logiikkaohjausten historiaa

Teollisuudessa ja automaatiossa käytetään yleensä paljon ohjausjärjestelmiä, joissa järjestelmä saa työalueelta kaksitilaista on/off-tyyppistä tietoa. Monet toimilaitteet toimivat myös käyntiin/seis- tai auki/kiinni-tyyppisillä komennoilla. Aiemmin nämä ohjaukset toteutettiin lähes aina releillä, jonka vuoksi järjestelmät olivat monimutkaisia suurine komponenttimäärineen. Ohjelmoitavat logiikat suunniteltiin aluksi autoteollisuutta varten, jossa General Motors esitti vuonna 1968 seuraavat vaatimukset ohjelmoitavalle logiikkaohjaukselle:

1. Laitteen on oltava ohjelmoitava ja se on voitava ohjelmoida uudelleen.
2. Sen on toimittava moitteettomasti teollisuuden konepajoissa.
3. Sen on kestettävä yhdysvaltalaisen vaatimuksen mukaan 120 V:n vaihtosähkösignaaleja, joita annetaan painonapeilla ja rajakytkimillä.
4. Sen lähtöjen tulee kestää sähkömoottoreiden vaatimaa kuormaa sekä käynnistyksissä että jatkuvassa ajossa.
5. Hinnan tulee olla kilpailukykyinen verrattuna jo olemassa oleviin kiinteästi langoitettuihin logiikkalaitteisiin. (Keinänen ym. 2007, 221.)

Ensimmäisenä termin PLC (Programmable Logic Controller) otti käyttöön Bedford Associates, joka myös patentoi keksinnön. Alkuperäinen ajatus logiikkaohjelmoinnista oli, että tietokoneen ohjelmointitaitoja ei tarvittaisi. Nykyisin ohjelmointi tehdään poikkeuksetta tietokoneohjelmilla, ja ohjelman tekijä voi yleensä valita useammasta ohjelmointikielestä itselleen parhaiten sopivan ohjelmointitavan. (Keinänen ym. 2007, 221–222.)

### 3.2 Logiikkatyypit

PLC-laitteiden periaatteellinen ero on niiden valmiudessa suorittaa ohjaustoimintoja. Tikapuuperiaatteella toimivia laitteita kutsutaan askeltaviksi logiikoiksi ja muita

vapaasti ohjelmoitaviksi logiikoiksi. Logiikat voidaan jakaa nykyisin kahteen ryhmään sen perusteella, ovatko ne kiinteällä I/O-määrällä varustettuja vai erilaisista moduuleista kokoonpantavia laitteistoja. Hyvin monimutkaisten ja isojen järjestelmien ohjaamiseen käytetään yleensä modulaarisia logiikkajärjestelmiä, jolloin käyttäjä voi valita tarpeeseensa sopivat erilaiset I/O- ja kenttäväylämoduulit. Kytönteknisesti logiikat voivat olla joko PNP- tai NPN-tyyppisiä. Suomessa ja Euroopassa käytetään yleisesti positiivisesti kytkeytyviä logiikoita, kun taas Japanissa ja USA:ssa NPN-logiikat ovat yleisempiä. Positiivisesti kytkeytyvissä logiikoissa logiikan tulo on nimensä mukainen, koska antureilta saapuva signaali kulkee logiikkaan päin. I/O-piirien tyyppi on tärkeää tietää, koska logiikkaan kytkettävät anturit täytyy valita kytkentätyyppin mukaan. (Keinänen ym. 2007, 222.)

### 3.2.1 Ohjelmoitavat logiikat

PLC eli ohjelmoitava logiikka on pieni mikroprosessorilla varustettu tietokone, joka ohjaa koneiden tai tuotantolinjojen reaaliaikaisia automaatioprosesseja (Keinänen ym. 2007, 212).

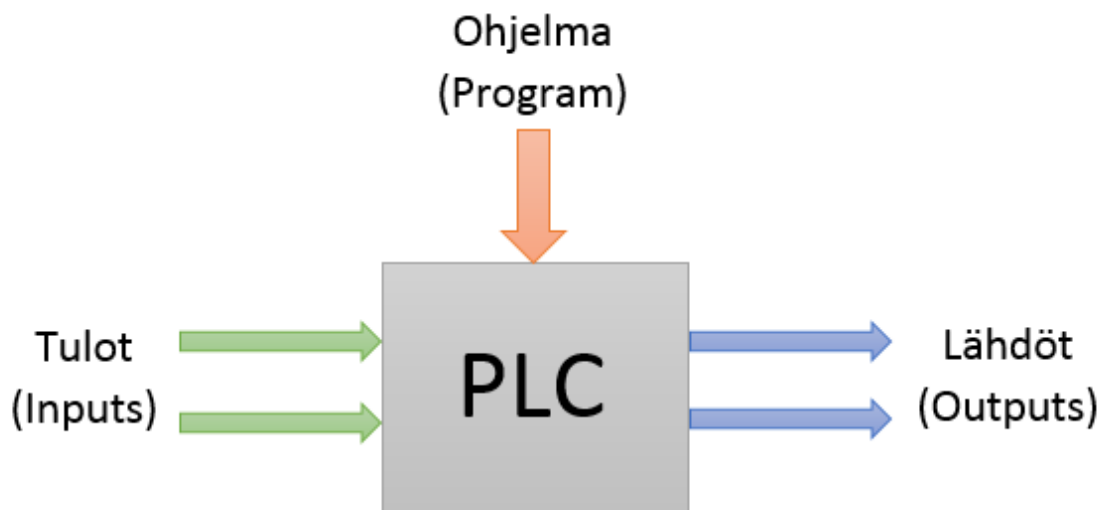
Ohjelmoitava logiikka toimii olennaisena osana ohjelmoitavaa ohjausjärjestelmää. Sen tuloihin kytketään järjestelmän tilaa havainnoivat anturit ja lähestymiskytkimet. Logiikan lähdöt ohjaavat toimilaitteita, kuten sähkömoottoreita, releitä, merkkilamppuja ja magneettiventtiileitä. Ohjelma kirjoitetaan logiikan muistiin, jossa ohjelma valvoo järjestelmän tilaa reaaliaikaisesti. PLC-laitteita kutsutaan myös vapaasti ohjelmoitaviksi logiikoiksi, koska ohjelman kirjoitusjärjestyksellä ei ole merkitystä. Logiikka lukee ohjelmaa kaiken aikaa, joten ohjelman osat voidaan kirjoittaa missä järjestyksessä tahansa. Logiikkaohjelma voidaan ohjelmoida toimimaan myös askel askeleelta -periaatteella. (Keinänen ym. 2007, 223.)

Logiikan muistissa olevaa ohjelmaa selataan PLC-järjestelmässä kiertävästi. Ensin luetaan kaikkien logiikan tulojen ja lähtöjen tila, jonka jälkeen tulokset tallennetaan keskusyksikön I/O-muistiin (kuvio 2). Tämän jälkeen käydään läpi kaikki ohjelma-muistissa olevat ohjelmarivit vuoron perään. Tulos käsitellään ja toteutetaan samassa järjestyksessä, kun logiikka-ohjelmaa luetaan. Logiikat käyttävät ohjelmointikieliä, joiden peruselementit muodostuvat logiikkaporteista sekä käskysanoista,



joilla ohjataan esimerkiksi ajastimia, laskureita ja apumuisteja. Komennot vaihtelevat eri valmistajien ohjelmien mukaan, mutta ohjelmat mahdollistavat yleensä kolme eri ohjelmointiperiaatetta:

1. Käskylista (STL), eli "Statement List"
2. Kosketinkaavio (LAD), eli "Ladder Diagram"
3. Toimintalohko-ohjelmointi (FBD), "Function Block Diagram". (Keinänen ym. 2007, 223.)



Kuvio 2. Ohjelmoitavan logiikan perustoimintaperiaate

### 3.2.2 Soft-PLC

Soft-PLC käyttää käyttöalustana PC:tä. Sen avulla saavutetaan samat asiat ja toiminnot tietokoneympäristössä kuin tavallisella PLC-logiikalla. Verrattuna tavalliseen PLC-logiikkaan Soft-PLC:n etuina ovat matalat kustannukset, tehokkaat yhteydet ja suuri datan käsittelykapasiteetti. (Liang & Li 2011.)

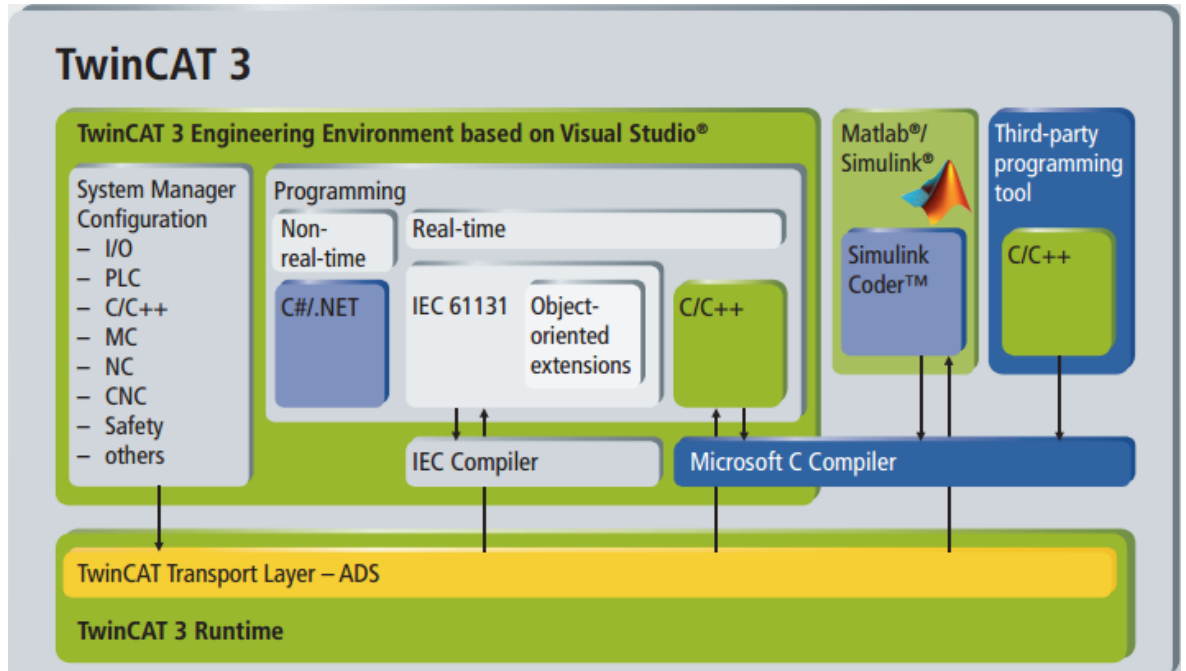
Soft-PLC on tietokoneen mikroprosessoria käyttävä ohjelmallinen logiikka, joka suorittaa logiikkaohjelmaa reaaliaikaisesti niin sanotussa suojatussa tilassa. Logiikkaohjelmien suoritus ei ole siis riippuvainen tietokoneen muiden ohjelmien tai käyttöjärjestelmien tilasta. Soft-PLC-ratkaisuissa kenttälaitteet liitetään järjestelmään kenttäväyläratkaisulla. Soft-PLC:n etuna on prosessissa tarvittavan tietokoneohjauksen ja logiikkaohjauksen toimiminen samassa laitteistossa. Ohjaukset voidaan

liittää toisiinsa ilman erillisiä tiedonsiirtoratkaisuja. (Keinänen ym. 2007, 213.)

### **3.3 Beckhoff ja Beckhoff TwinCAT 3 XAE**

Beckhoff toimittaa PC-pohjaisia avoimia automaatiojärjestelmiä. Sen tuotevalikoimiin kuuluvat muun muassa kenttäväyläkomponentit, liikkeenohjaustuotteet, teollisuus-PC:t, ohjauspaneelit sekä automaatiosovelluksien ohjelmistot, kuten TwinCAT 3. Eri ryhmien tuotteita voidaan integroida täydellisiksi ohjausjärjestelmiksi tai niitä voidaan käyttää erillisinä komponentteina. Beckhoffin Suomen pääkonttori sijaitsee Hyvinkäällä ja haarakonttorit Tampereella, Seinäjoella ja jatkossa Tallinassa. Beckhoff Automation GmbH & Co. KG:n pääkonttori sijaitsee Verlissä, Saksassa. Sinne on keskitetty kaikki keskeisimmät toiminnot. Yhteistyökumppaneita yrityksellä löytyy yli 75 maasta. (Beckhoff 2016.)

TwinCAT 3 XAE -ohjelma on PC-pohjainen automaatio-ohjain, joka integroi reaaliaikaisen ohjauksen PLC-, NC- ja CNC-funktioilla yhdessä ohjelmassa. Ohjelmaa käyttäen ohjelmoidaan kaikki Beckhoffin ohjaimet maailmanlaajuisesti hyväksytyin IEC 61131-3 -standardin mukaisesti. TwinCAT 3 XAE (eXtended Automation Engineering) toimii integroituna osana Microsoftin Visual Studio -ohjelmankehitysympäristössä, joka mahdollistaa IEC 61131-3 -standardin mukaisen PLC-ohjelmoinnin sekä C/C++-ohjelmointikielet. Moduulit voivat lähettää ja vastaanottaa tietoa keskenään riippumatta siitä, millä ohjelmointikielellä ne on kirjoitettu (kuvio 3). TwinCAT System Managerin integrointi kehitysympäristöön mahdollistaa laitteiden konfiguroinnin, parametrisoinnin, ohjelmoinnin ja määrittämisen vain yhdellä ohjelmalla. (Beckhoff 2012.)



Kuvio 3. TwinCAT 3 XAE -moduulit (Beckhoff 2012).

## 4 VIRTUAALINEN KÄYTTÖNOTTO

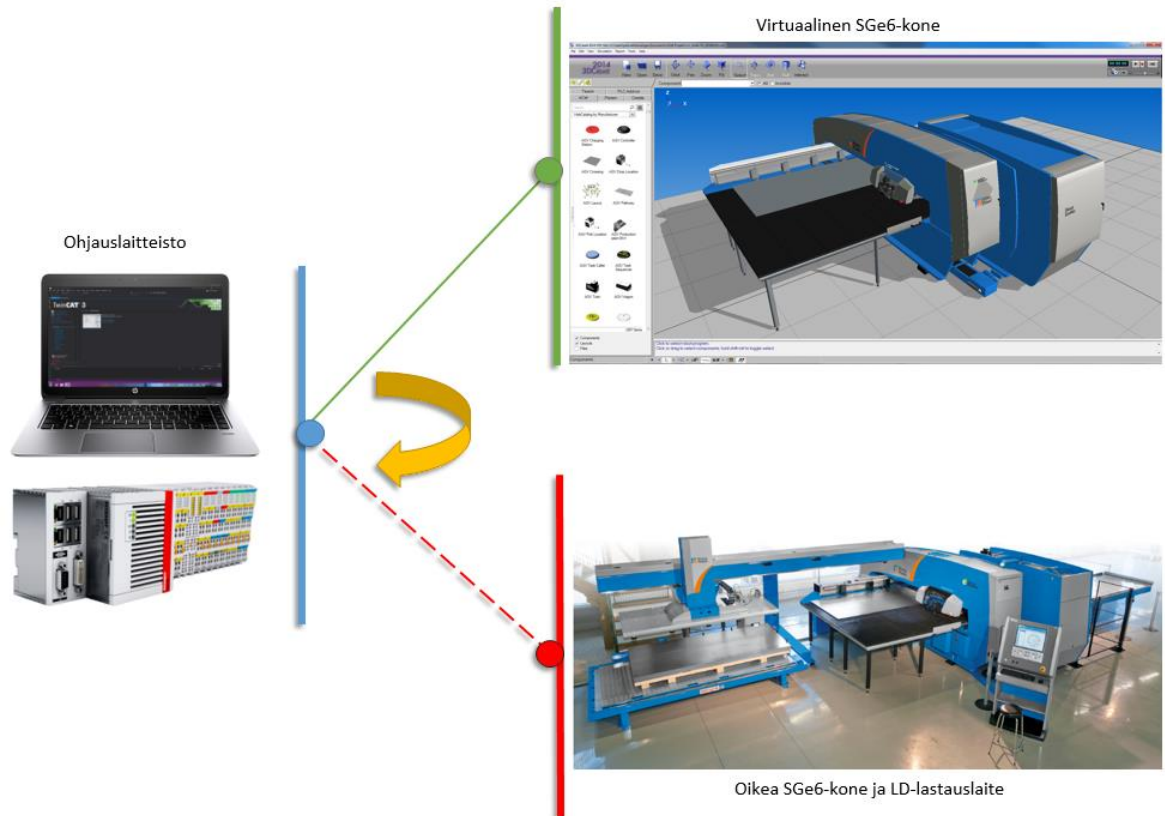
### 4.1 Mitä on virtuaalinen käyttöönotto?

Virtuaalisen käyttöönoton (Virtual Commissioning) perusideana on kytkeä fyysisen koneen PLC- tai HMI-ohjaus virtuaaliseen 3D-malliin (kuvio 4). Näin eri alojen insinöörit voivat työstää samaa mallia suunnittelun ja ohjauksen puolelta. Kun ohjaus kytketään virtuaalimalliin, voidaan PLC-ohjelmia testata virtuaalisesti ennen fyysisen koneen käyttöönottoa. (Liu, Suchold & Diedrich 2012, 132.)

Käyttöönotto on tavallisesti automaatiojärjestelmäprojektien viimeinen vaihe ja se voi viedä jopa 15–20 % kokonaistoimitusajasta. Noin kaksi kolmannesta käyttöönottoon käytetystä ajasta kuluu ohjelmistovirheiden korjaukseen, koska ohjausjärjestelmää päästään yleensä testaamaan kunnolla vasta, kun kaikki laitteet on valmistettu ja liitetty toisiinsa. Tiukkojen aikarajojen vuoksi ohjausten testauksessa kiirehditään turhaan ja ohjausvirheiden vuoksi laitteet voivat kärsiä vahinkoja. Projekteissa voidaan säästää paljon aikaa ja vaivaa, kun nämä ongelmat ratkaistaan ajoissa ja säästetty aika voidaan hyödyntää laitteiden optimointiin ja lyhempiin toimitusaikoihin. (Visual Components 2016d.)

Virtuaalisen käyttöönoton avulla merkittävä osa käyttöönoton työvaiheista voidaan suorittaa ilman tiukkoja aikarajoja projektin aiemmissa vaiheissa. Virtuaalinen käyttöönotto aloitetaan luomalla oikeaa tehdasta tai konetta vastaava simulointimalli. Virtuaalimalli kytketään oikeaan ohjaukseen, jolloin sitä voidaan käyttää ohjauksen testaukseen sekä varmistamaan kokoonpanon ja suunnittelun toimivuus. Mahdolliset virheet voidaan havaita jo suunnitteluvaiheessa, näin ohjaukseen voidaan ajoissa tehdä tarvittavat korjaukset. Virtuaalimalli reagoi ohjauksen lähtöihin ja kytkee päälle tuloja, kuten oikea järjestelmäkin tekisi. Tämä mahdollistaa interaktiivisen 3D-visualisoinnin, jonka avulla suunnittelijat ja asiakkaat näkevät, kuinka järjestelmä oikeasti käyttäytyy, vaikka sitä ei fyysisesti olisi vielä olemassakaan. Interaktiivisesta visualisoinnista on iso etu ohjauksen testauksessa verrattuna yleiseen testaustapaan, jossa ohjelmoija tarkastelee pelkkää koodia sekä tuloja ja lähtöjä on kytkettävä päälle manuaalisesti. Virtuaalisen käyttöönoton tuloksena syntyy laadukkaam-

pia ohjelmia, ja mikä tärkeintä, se on myös hyvin turvallista. Virtuaalisen käyttöönoton avulla säästytään koneisiin ja niiden käyttäjiin kohdistuvilta vahingoilta, joita varsinaisessa testauksessa voisi syntyä. (Visual Components 2016d.)

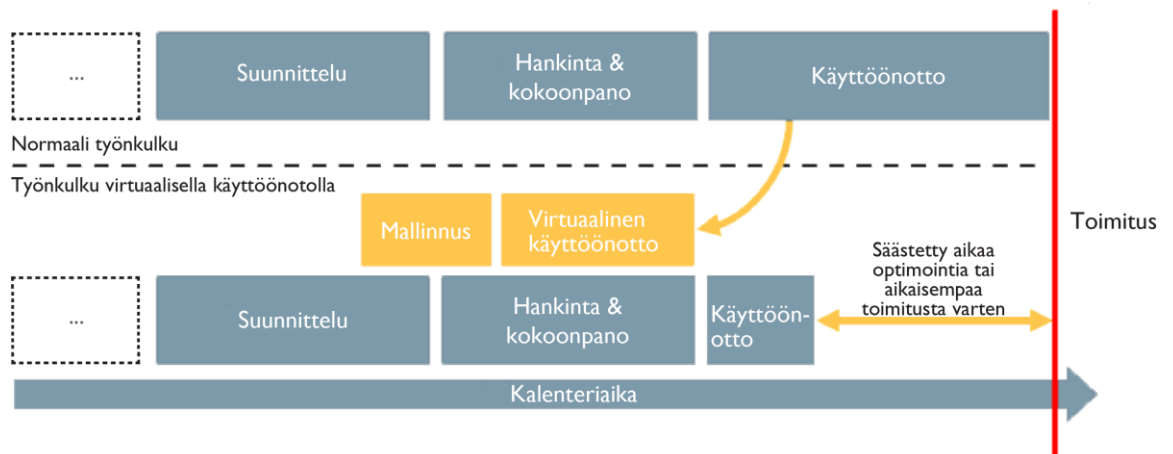


Kuvio 4. Samalla ohjauksella voidaan ohjata virtuaalikonetta ja oikeaa konetta (Perustuu Liu, Suchold & Diedrich 2012, 135).

## 4.2 Virtuaalisen käyttöönoton hyödyt

Virtuaalisella käyttöönotolla on paljon hyötyjä suunnitteluprosessin alusta loppuun (kuvio 5). Suunnittelun kokonaisaikaa saadaan lyhennettyä ja prototyyppien valmistus minimoitua. Suunnittelun aikana huomatu virheet ovat taloudellisesti edullisempia korjata jo suunnitteluvaiheessa, kuin huomata ne vasta käyttöönottovaiheessa. Esimerkkikenttäkokeessa pienellä 17 anturin ja 10 toimilaitteen PLC-järjestelmällä havaittiin, että käytettäessä virtuaalista käyttöönottoa laatu parani 37 %:sta 84 %:iin, kun laatu määriteltiin vaatimusten täyttymisenä. Varsinainen käyttöönottoaika lyheni 75 %:lla ja projektin kokonaisaika 15 %:lla. (Visual Components 2016d.)

Simulointi- ja ohjelmointi-insinöörien välille syntyy tiiviimpää yhteistyötä, kun jo tarjousvaiheessa luodaan yksinkertainen visuaalinen malli tarjotusta järjestelmästä ja tämän jälkeen suunnitteluvaiheessa kehitetään täydellinen simulointimalli. Interaktiivinen 3D-simulointi auttaa ymmärtämään mallin käyttäytymistä paljon paremmin kuin iso pino suunnitteludokumentteja. Simulointimalli toimii visuaalisena esittelytyökaluna myyntitilanteissa. Simulointimallin avulla voidaan seurata vaiheittain projektin etenemistä. Sen avulla pysytään kiinni projektin tavoitteissa ja saadaan järjestelmä toimimaan halutulla tavalla. Toimiva simulointimalli auttaa myös kehittämään uusia ideoita projektin edetessä. (Visual Components 2016d.)



Kuvio 5. Automaatioprojekti ilman virtuaalista käyttöönottoa ja sen kanssa (Perustuu Visual Components 2016d).

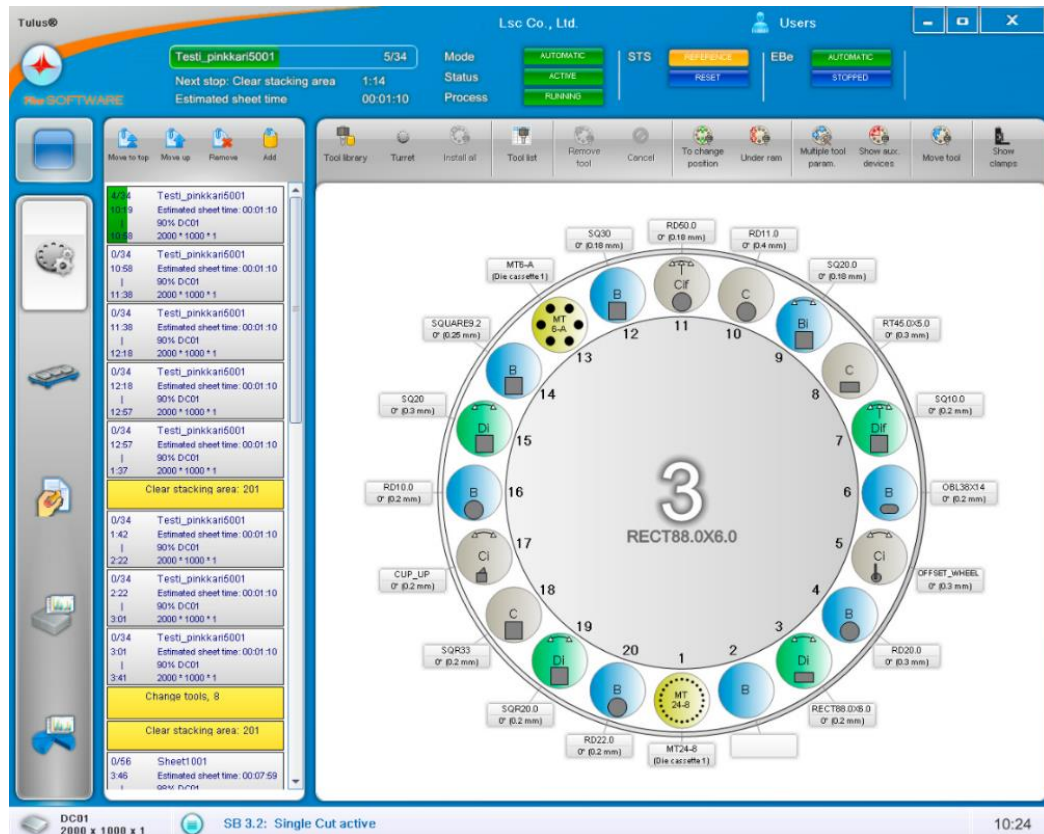
### 4.3 Visual Components PLC Add-on

PLC Add-on on 3DCreaten lisäosa, jolla on mahdollista ohjata simulointimalleja PLC-logiikkalaitteistoilla ja -ohjelmilla reaaliajassa Microsoft-OPC-yhteydellä tai valmistajakohtaisilla PLC-rajapinnoilla. PLC Add-on yhdistää OPC-palvelimessa tai PLC-logiikassa määritellyt muuttujat simulointimalliin luotuihin input- ja output-signaalikomponentteihin. Viestintä logiikan ja simulointimallin välillä on kaksisuuntainen. (Visual Components [Viitattu 15.3.2016].)

#### 4.4 Tulus Cell

Tulus Cell on integroitu ja skaalautuva Prima Powerin ohjelma keskitettyyn valmistuksenhallintaan, oli kyseessä sitten yksittäinen kone tai tehtaantilaajainen prosessi. Tulus ohjaa koneita tehtävillä ja tarjoaa helpon tavan lisätä uusia tilauksia koneen työjonoon. Työjono sisältää kaiken oleellisen konekohtaisen tiedon samassa näkyvässä. Tilauksia voidaan lisätä, poistaa tai järjestää uudelleen tuotannon aikataulujen mukaan myös koneen ollessa käynnissä. Tilauksenhallinta vastaa automaattisesti muutoksiin antaen välittömästi uuden tuotantoaikataulun. Se sisältää myös työkalujen ja materiaalien muutokset sekä arvioidut toimitusajat. (Prima Power 2013.)

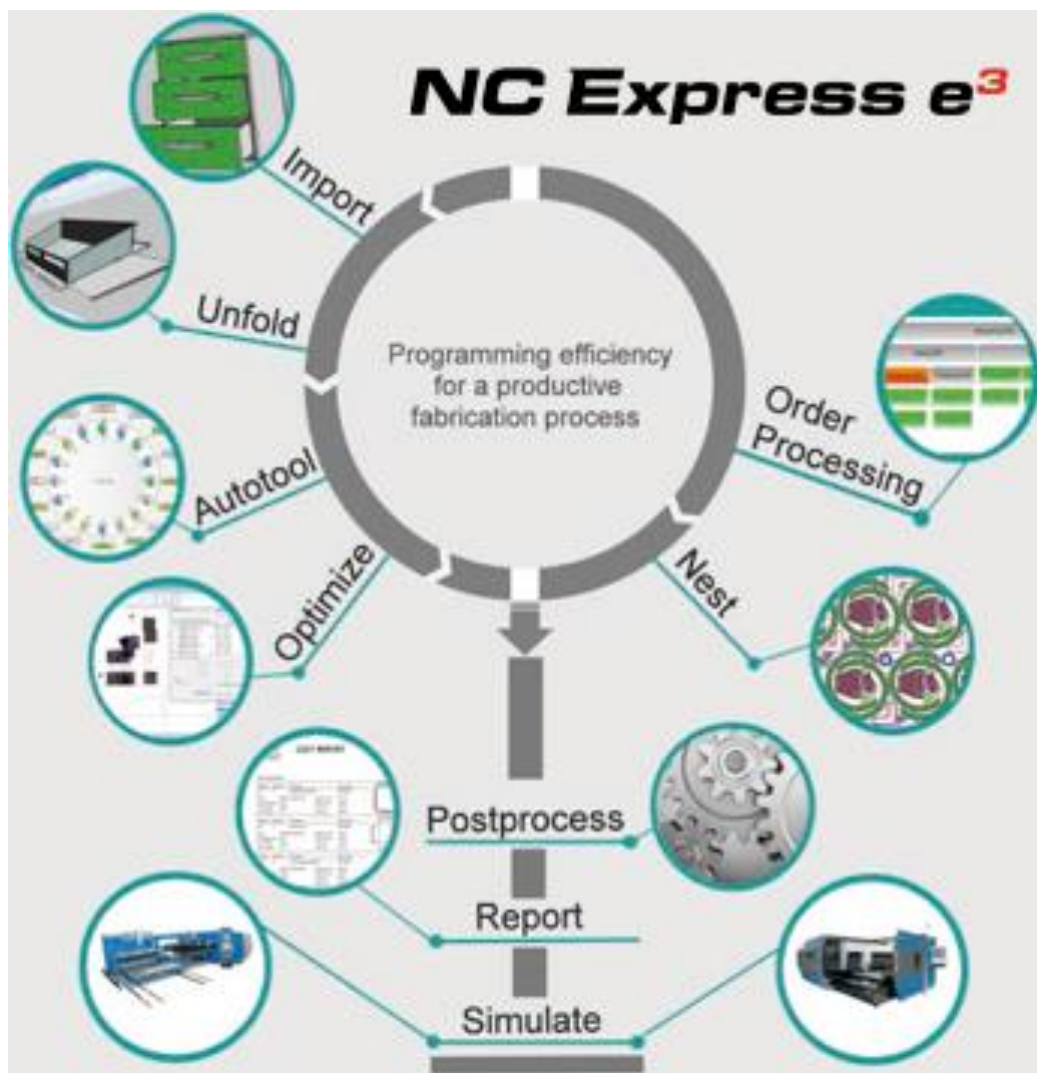
Tulus Cell on samalla koneiden käyttöliittymä, jolla ohjataan koneen toimintoja, työkaluja, työtilauksia ja valmistuneiden osien lajittelua (kuvio 6). Selkeän ja yksinkertaisen käyttöliittymän avulla käyttäjän on helppo käyttää konetta ja muokata sen parametreja. Tulus Cell ja NC Express toimivat keskenään synkronoidusti jakaen työkalujen ja työkalumakasiinin tiedot. (Prima Power 2013.)



Kuvio 6. Tulus Cellin käyttöliittymä (Prima Power 2016b).

#### 4.5 NC Express e<sup>3</sup>

NC Express e<sup>3</sup> on osa Prima Powerin ohjelmaperhettä. Se on suunniteltu toimimaan saumattomasti Prima Powerin koneiden ja Tuluksen kanssa. Ohjelma skaalautuu käyttötarkoituksen mukaan. Se sopii esimerkiksi täysin automaattisen työkalustusohjelman tekemiseen, joka määrittää kaikki kappaleen valmistukseen tarvittavat työkalut, ja missä vaiheessa valmistusta niitä käytetään. NC Express sopii myös vain yhden osan piirtämiseen ja työkalustukseen. Lopputuloksena syntyy automaattisesti optimoitu nestaus- ja NC-ohjelma. NC Expressin automaattisilla toiminnoilla saavutetaan tuotteiden korkeampi tuottavuus, korkeampi luotettavuus, parempi laatu ja lyhemmät läpimenoajat kuin manuaalisilla toiminnoilla (kuvio 7). (Prima Power 2014.)

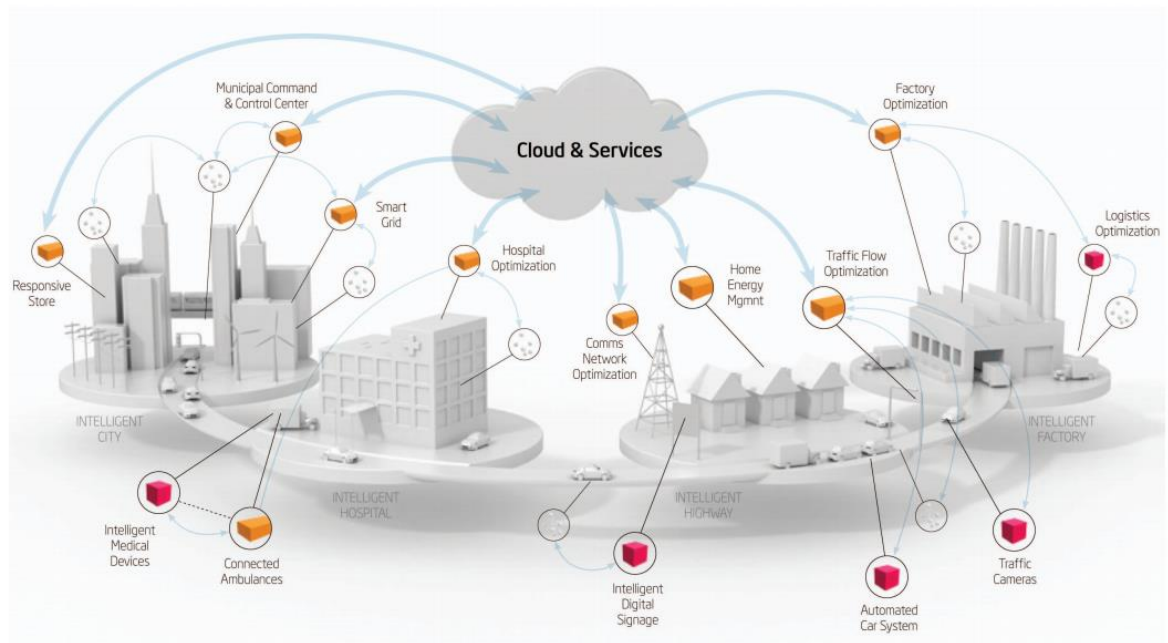


Kuvio 7. NC Express e<sup>3</sup> -toiminnot (Prima Power 2016a).



## 4.6 Teollisuus 4.0

Teollisuus 4.0, teollinen internet, esineiden internet ja big data -termit liittyvät kaikki digitalisaatioon, joka on yleistymässä kaikilla toimialoilla. Teollisuus 4.0 eli teollinen internet on Saksan valtion tukema ja rahoittama hanke, jonka mukaan teollisuus on matkalla kohti neljättä teollista vallankumousta. Ensimmäinen vallankumous käsitti höyrykoneilla ja vesivoimalla valjastetun tuotannon, toinen sähköä käyttävän ja työnjakoon perustuvan massatuotannon ja kolmas vallankumous syntyi elektronii-kan, tietotekniikan ja automaation ansiosta. Neljännen vallankumouksen perustana on älykkäiden esineiden verkko. Kaikki teollisuudessa käytettävät koneet, laitteet, prosessit ja tuotantolaitokset keskustelevat internetin välityksellä toistensa kanssa, ja niitä voidaan ohjata yhteensopivin työkaluin (kuvio 8). Keskeisintä on järjestelmien kyky analysoida kaikkea dataa, jota koneisiin, puolivalmiisiin tuotteisiin ja materiaaleihin upotetuista miniprosessoreista, tallenneyksiköistä, lähettimistä ja antureista saadaan, ja kuinka järjestelmien toimintaa voidaan niiden perusteella muuttaa. Tulevaisuudessa tehtaat ovat niin älykkäitä, että ne voivat optimoida ja kontrolloida tuotantoprosesseja itsenäisesti. (Reset 2014.)



Kuvio 8. Teollinen Internet kytkee laitteet ja ajoneuvot internetin välityksellä. (Lambert [Viitattu 7.4.2016]).

Teollisuus 4.0:n avulla voidaan lyhentää merkittävästi tuotteiden markkinoille tuloaika joustavien ja optimoitujen tuotantoprosessien avulla. Tuotteiden helppo ja kustannustehokas valmistus mahdollistaa erilaisten tuotevaihtoehtojen ja räätälöityjen yksittäisten tuotteiden valmistuksen. Kun tulevaisuudessa teollisuustuotannon koneet ja laitteet viestivät keskenään, logistiikkaketjut tekevät automaattisesti yhteistyötä. Keskeneräiset tuotteet lähettävät niitä valmistaville laitteille tuotteita koskevia tietoja. Koneista ja laitteista saadaan kerättyä paljon dataa, jonka perusteella ennustetaan optimaalinen tuotantopolku. Tuotantopolkua arvioidaan jokaisen tuotantovaiheen jälkeen. (Siemens [Viitattu 6.4.2016].)

Keskeisiä asioita teollisuuden uudella aikakaudella ovat muun muassa uudet ohjelmistot, virtuaalimaailman hyödyntäminen, eri laitteiden ja järjestelmien välillä oleva integraatio sekä datankeräykseen perustuvat uudet palvelut ja liiketoimintamallit. Kilpailukyvyn kannalta elintärkeää tuottavuutta voidaan parantaa parhailla suunnitteluprosesseilla ja mahdollisimman automatisoidulla tuotannolla. (Raunio 2014.)

## 5 TYÖN TOTEUTUS

### 5.1 Tarvekartoitus

Ennen varsinaisen virtuaalikoneen suunnittelua ja valmistusta tehtiin virtuaalikoneen tarvekartoitus, jonka avulla selvitettiin virtuaalikoneen tarpeellisuutta konsernin sisällä. Tarvekartoitukseen kerättiin materiaalia yrityksen sisäistä markkinointia varten. Materiaali toimi samalla myös virtuaalikoneprojektin esittelymateriaalina. Tarvekartoituksen materiaaleilla pyrittiin tiedottamaan uudesta projektista ja esittelemään sen toimintaa. Virtuaalikoneen mahdollisia käyttökohteita voisivat olla esimerkiksi:

1. Messut, joissa on pieni budjetti tai liian vähän tilaa oikeille koneelle.
2. Koneiden käyttö- ja huoltokoulutukset.
3. Teknologia- ja koulutuskeskuksen (TTC, Technology and Training Center) koneiden esittelytyökalu.
4. Esittelytyökalu tytäryhtiöille, joilla ei ole tilaa oikeille koneille.
5. Myynnintuen esittelytyökalu myyntitapahtumissa.

Virtuaalikoneen hyötyjä verrattiin yrityksen nykyiseen tilaan ja niistä luotiin PowerPoint-esitys, jossa verrattiin tilannetta jokaisessa käyttökohteessa (kuvio 9).

**Prima Power**

### 1. Exhibition stand

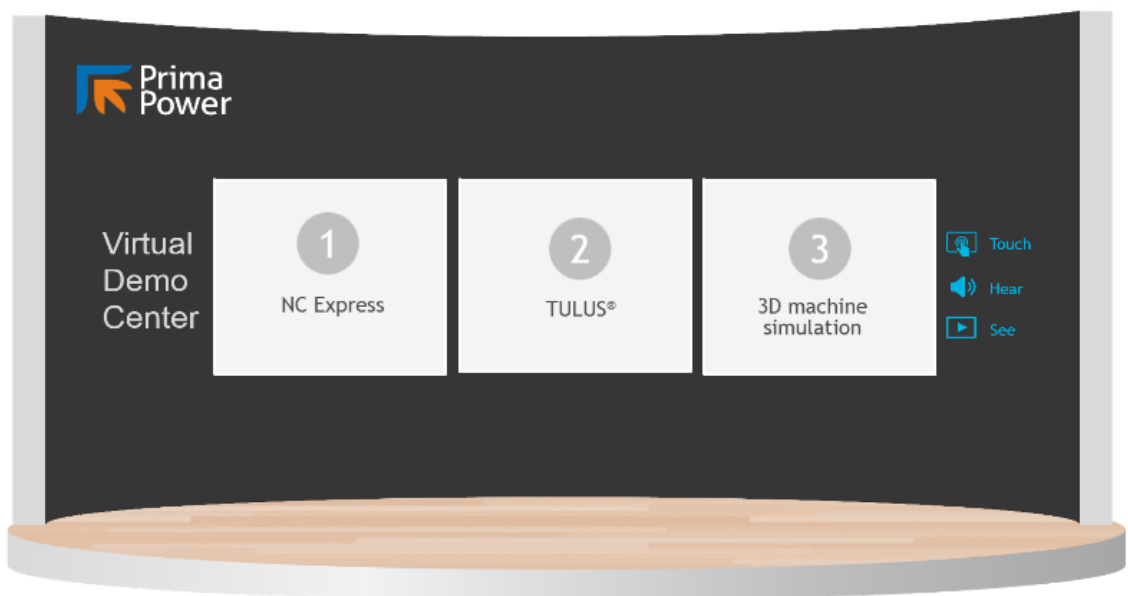
<u>Current situation</u>	<u>Situation with Virtual Machine</u>
<ul style="list-style-type: none"> <li>• Installation group needed -&gt; more people</li> <li>• Time and resources needed for planning and installation -&gt; cuts time from other projects</li> <li>• Hard to demonstrate everything at the same time</li> <li>• Travel and hotel costs</li> <li>• Large space needed</li> <li>• Material costs</li> <li>• Not climate-friendly</li> <li>• Big budget</li> </ul>	<ul style="list-style-type: none"> <li>• No installation group -&gt; less people -&gt; savings</li> <li>• No time needed for installation and planning -&gt; savings</li> <li>• Demonstrate everything in one place at the same time</li> <li>• Less travel and hotel costs -&gt; savings</li> <li>• No space needed</li> <li>• Material savings</li> <li>• Green efficiency</li> <li>• Low budget</li> </ul>

€
€
green means
👍

Kuvio 9. Messujen nykyinen tila verrattuna tilaan virtuaalikoneen kanssa

Esityksellä pyrittiin havainnoimaan, kuinka paljon rahaa ja aikaa voitaisiin säästää käyttämällä virtuaalikonetta. Säästöä syntyisi esimerkiksi käyttöönotto-, matka- ja materiaalikustannuksissa. PowerPoint-esitystä hyödynnettiin Adobe Captivate -ohjelmalla tehdyssä interaktiivisessa esityksessä, jossa virtuaalikoneen käyttökohteiden ja hyötyjen lisäksi havainnollistettiin virtuaalikoneen käyttöön vaadittavia laitteita sekä toimintaperiaatetta kuvin ja videoin. Samalla ideoitiin messuille virtuaalikoneen esittelyä varten Virtual Demo Center -ständi (kuvio 10). Ständissä on jokaiselle virtuaalikoneen käyttämälle ohjelmalle oma näyttönsä, jotta koneenkäyttöä voidaan esitellä alusta alkaen, ja koko prosessi on nähtävillä samalla hetkellä.

Varsinainen tarvekartoitus suoritettiin Yhdysvaltojen, Venäjän sekä Puolan Prima Power -konsernin maajohtajien kanssa. Haastatteluissa esiteltiin ennakkoon tehdyt materiaalit ja käytiin tarkemmin läpi virtuaalikoneen toimintaa sekä vaatimuksia. Haastateltavat eivät saaneet materiaaleja ennakkoon luettaviksi, jotta haastattelun aikana saatiin aito reaktio mahdollisesta kiinnostuksesta. Virtuaalikoneesta oltiin aidosti kiinnostuneita, vaikka virtuaalikonekonsepti nosti esiin epäkohtia ja sen toteuttavuutta epäiltiin. Jo olemassa olevien kehitysideoiden lisäksi saatiin paljon uusia kehitysideoita virtuaalikoneprojektin jatkoa varten.



Kuvio 10. Virtual Demo Center -ständi

## 5.2 Kokoonpano ja laitteistot

Tässä työssä virtuaalikone päätettiin tehdä Prima Powerin TheCOMBI-tuoteperheeseen kuuluvasta Shear Genius SGe6 -koneesta. Koneessa yhdistyvät ohutlevyn lävistys, kulmaleikkuu, automaattinen lastaus ja komponenttien poisto työsolusta. Näiden toimintojen integroinnilla saavutetaan suurempi tuotantokapasiteetti, laatu ja kustannustehokkuus kuin erillisillä laitteilla. Modulaarisen tuoterakenteen avulla SGe6-konetta voidaan käyttää erillisenä koneena tai osana isompaa järjestelmää. Alkuperäisenä kokoonpanona SGe6-koneen lisäksi suunniteltiin kahta modulaarista automaatiolaitetta, LD-lastauslaitetta sekä STS-lajittelu- ja pinoamislaitetta (kuvio 11). Kyseisessä kokoonpanossa olisi ollut kolme TwinCAT 3 -ohjausta, koska jokaisella laitteella on oma ohjaus. Työn alussa selvisi, että Visual Componentsin PLC Add-on ei tue tällä hetkellä kuin yhtä samanaikaista TwinCAT 3 -ohjausta, joten automaatiolaitteet oli jätettävä pois. Kokoonpanoksi jäi siis pelkkä SGe6-kone (kuvio 12).

Työn päätavoitteena oli tehdä paikallinen virtuaalikoneen työasema, jossa kaikki ohjelmat ja toiminnot olisivat samalla PC:llä. Työssä oli tarkoitus myös tarkastella pilvipalvelujen mahdollisuuksia ja selvittää mitä hyötyjä niistä tässä projektissa olisi.



Kuvio 11. Alkuperäinen suunnitelma kokoonpanosta



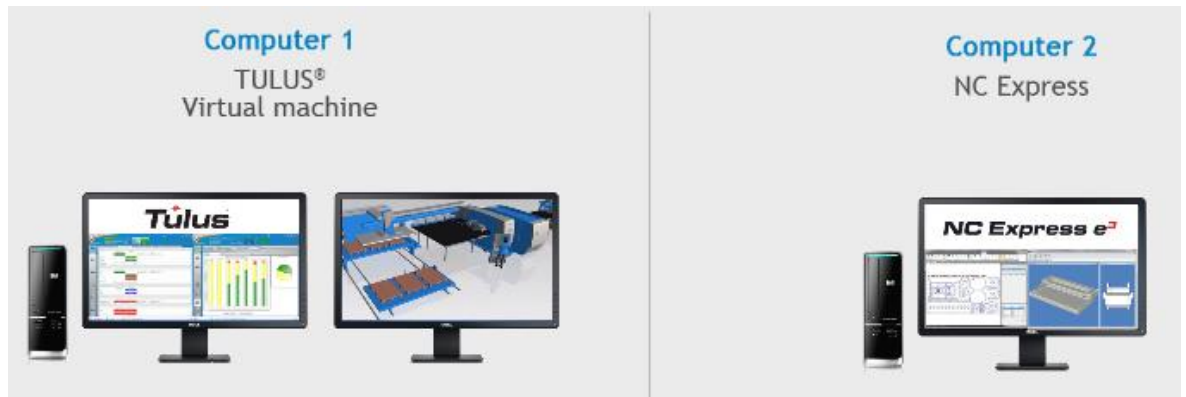
Kuvio 12. Virtuaalikoneen kokoonpano

### 5.2.1 Paikallinen työasema

Paikallinen virtuaalikoneen työasema tarkoittaa, että kaikki tarvittavat ohjelmat on asennettu yhdelle tai useammalle PC:lle ja niitä voidaan käyttää yhtä aikaa (kuvio 13). Virtuaalikoneen käyttöä varten tarvitaan

- 3D-CAD-suunnitteluohjelma, esimerkiksi Siemens Solid Edge
- CAM-ohjelma, NC Express e<sup>3</sup>
- koneen käyttöliittymäohjelma, Tulus Cell
- 3D-simulointiohjelma, Visual Components 3DCreate
- virtuaalisen käyttöönoton ohjelma, Visual Components PLC Add-on

Työn toimeksiantajalta löytyi koneiden 3D-CAD-mallit valmiina, joten aikaa ei kulu-  
nut suurten ja monimutkaisten koneiden mallinnukseen. 3D-CAD-mallit oli tehty  
Siemensin Solid Edge -ohjelmalla ja valmiit CAD-mallit voitiin tuoda suoraan Visual  
Componentsin 3DCreate-simulointiohjelmaan. Toimeksiantaja käyttää koneiden  
ohjaukseen Beckhoffin TwinCAT 3 -ohjelmaa, ja työssä virtuaalikoneen ohjaukseen  
käytettiin olemassa olevien oikeiden koneiden ohjauksia. Virtuaalikoneen testauk-  
sessa käytettiin NC Expressillä tehtyä ohutlevyn nestäusta, jolla voidaan testata vir-  
tuaalikoneen lävistys-, kulmaleikkuu-, työkalunvaihto- ja kynsien otteensiirtotoimin-  
toja.



Kuvio 13. Paikallinen työasema

### 5.2.2 Työasema pilvipalvelussa

Työssä oli tarkoituksena selvittää, millaisia mahdollisuuksia pilvipalvelut tarjoavat ja kuinka niitä voitaisiin tulevaisuudessa hyödyntää. Pilvipalvelujen ideana on luoda siihen tarkoitukseen tehdyillä ohjelmilla virtuaalinen käyttöympäristö, esimerkiksi Amazon Web Services -ohjelmalla. Palvelimille luodaan virtuaalinen tietokone ja luomisen yhteydessä sille asennetaan kaikki tarvittavat ohjelmat (kuvio 14). Tämän jälkeen virtuaalinen tietokone voidaan käynnistää virtuaalisesti sitä tarvittaessa ympäri maailmaa internetin välityksellä. Pilvipalvelut eliminoivat tarpeen paikallisesti asennetuille ohjelmille ja päivityksille, joiden ylläpitäminen on haastavaa. Uusimpien ohjelmaversioiden ja koneiden päivittäminen virtuaaliselle tietokoneelle on helpompaa, koska se on saatavilla internetin välityksellä kaikille, joilla on siihen voimassa olevat käyttöoikeudet.

Pilvipalveluiden käyttö olisi käytännössä jo mahdollista, mutta niitä ei tässä työssä kuitenkaan käytetä. Virtuaalisia tietokoneita on mahdollista tehdä erilaisilla konfiguraatioilla eri käyttäjäryhmille, kun tulevaisuudessa useammasta toimeksiantajan koneesta on tehty virtuaalikone. Myyjille voidaan suunnitella pilvipalveluun yksinkertaisempi käyttöliittymä, josta on helppo valita haluttu virtuaalikone. Testaus- ja tuotekehitys voi käyttää käyttöliittymää, jossa on paljon enemmän valintoja ja mahdollisuuksia. Pilvipalvelut tulevat varmasti yleistymään tulevaisuudessa käytännöllisyyden ja niiden luomien rajattomien mahdollisuuksien myötä.





Kuvio 14. Työasema pilvipalvelussa

### 5.3 Virtuaalikoneen toimintaperiaate

Koko virtuaalikoneen toiminta aloitetaan luomalla valmistettavasta ohutlevytuotteesta 2D- tai 3D-malli esimerkiksi Siemensin Solid Edge -ohjelmalla.

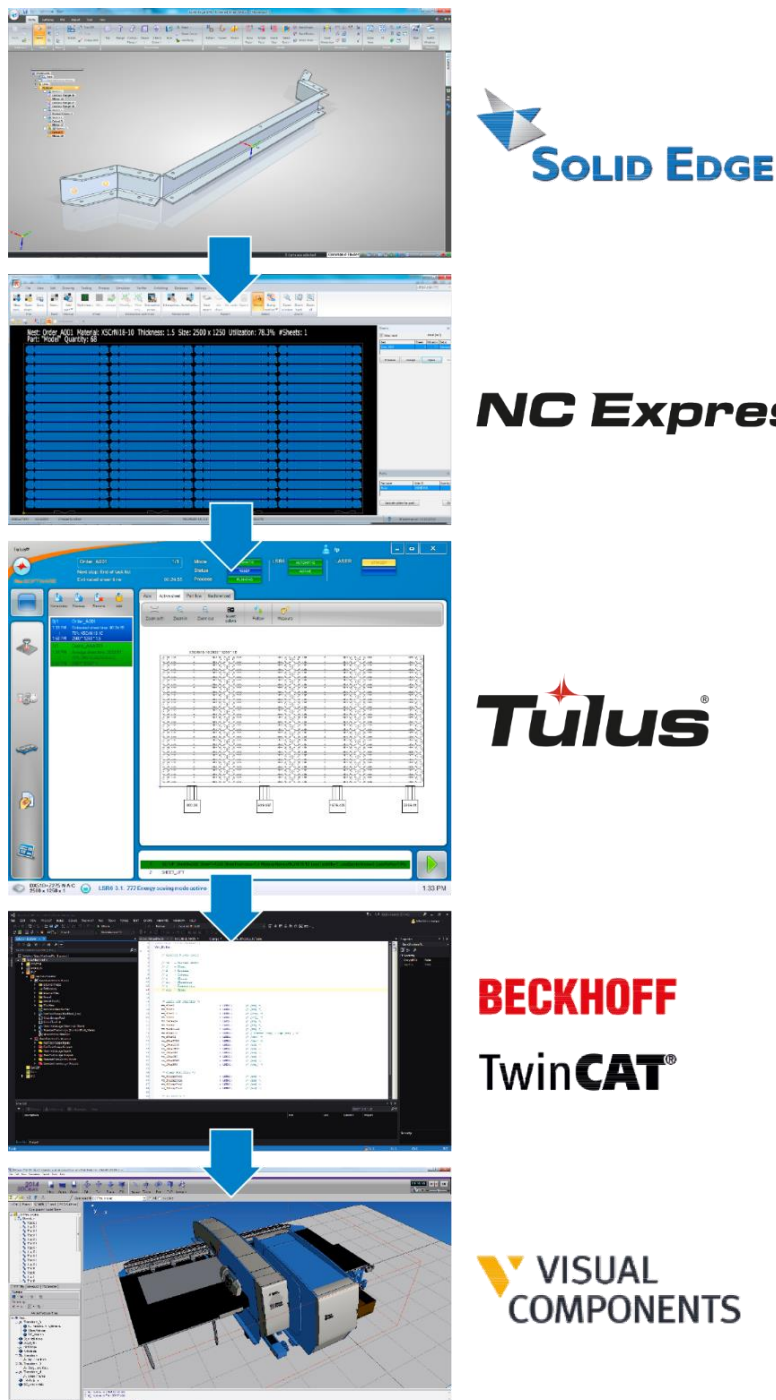
Valmis CAD-malli tuodaan suoraan NC Express e<sup>3</sup>-ohjelmaan, jolla aukilevitetään CAD-mallin yksittäisiä osia tai kokonaisia kokoonpanoja. NC Expressillä määritellään valmistukseen vaadittavat työkalut sekä ohutlevyn nestaus. Nestauksella optimoidaan ohutlevyn hyötykäyttö sovittamalla raaka-ainelevylle mahdollisimman paljon valmistettavia osia. NC Express toimii CAM-ohjelmana, joten se luo myös nestille NC-koodin.

Nestattu levy tuodaan Tulus Cell -ohjelman työjonoon. Ohjelmalla voidaan määrittää tuotettavien levyjen määrä, parametrit sekä mihin valmiit tuotteet lajitellaan. Tulus Cell näyttää jokaiselle levyille arvioidun valmistumisajan sekä työn etenemisen. Tulus ohjaa siis nestauksen mukana tullutta NC-koodia ja kommunikoi sen avulla Beckhoff TwinCATin kanssa.

TwinCATissa oleva oikean koneen ohjaus tekee tarvittavat toiminnot Tuluksesta tulevien tietojen mukaan ja suorittaa virtuaalikoneen ohjauksen. Tarvittavien akseli-muuttujien tiedot siirretään Visual Components 3DCreate -ohjelmaan, jossa on oikeaa konetta vastaava 3D-simulointimalli.

Ensin PLC Add-on -lisäosaan luodaan PLC-yhteys ja tämän jälkeen simulointimalliin luodut signaalimuuttujat kytketään TwinCAT-ohjauksen vastaaviin akselimuuttujiin. Näin TwinCATissa tapahtuvat muutokset ohjaavat virtuaalista simulointimallia, kuten ne normaalisti ohjaisivat oikeaa konetta (kuvio 15).





Kuvio 15. Virtuaalikoneen toimintaperiaate

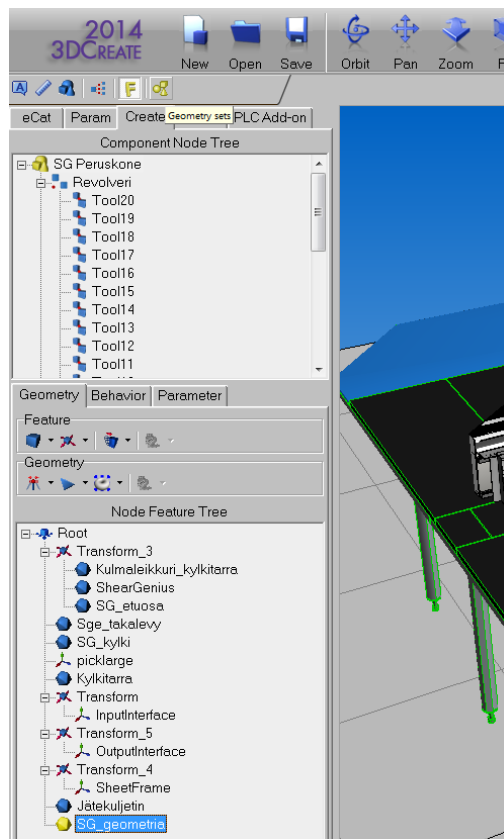
#### 5.4 Simulointimallin rakentaminen

Simulointimallia rakennettaessa malliin oli rakennettava 3D-geometria ja kinematiikanivelet. Malliin oli myös määritettävä signaalit ja parametrit sekä kirjoitettava niitä ohjaamaan PythonScript-koodi. SGe6-koneen CAD-malli jakaantui kahteen koonpanoon, SGe-peruskoneeseen sekä RS-kulmaleikkuriin.

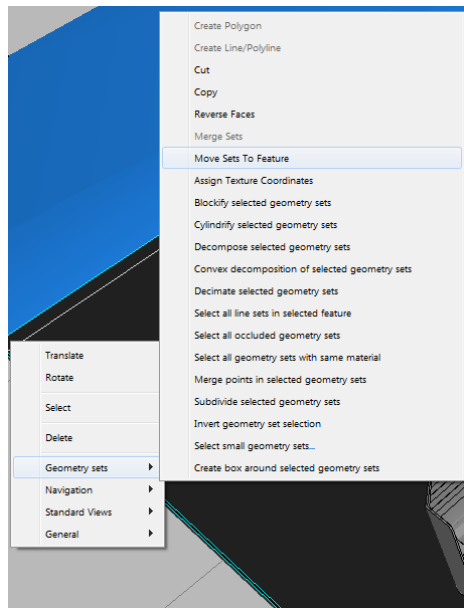
### 5.4.1 Geometrioiden luominen

3DCreateen tuotu CAD-malli oli pilkottava pienempiin osiin, jotta eri komponenteille voitiin luoda piirrepuu, jonka avulla jokaiselle komponentille voitiin määrittää tarkat sijainnit. Piirrepuun avulla komponentteihin voitiin kytkeä niitä liikuttavat muuttujaparametrit. Geometrian luomiseen ja muokkaamiseen liittyvät työkalut löytyvät Create-välilehden alta. Geometry-välilehden alta löytyvät työkalut 3D-mallintamiseen ja niihin liittyvien sääntöjen luomiseen. Saman välilehden alta löytyy myös piirrepuu, johon CAD-mallin kokonaisesta mallista voidaan pilkkoa pienempiä piirteitä.

Piirrepuusta valitaan isompi geometriapiirre ja vaihdetaan valintatyyppiä Geometry sets (kuvio 16). Tämän avulla voidaan valita piirrepuusta valitusta geometriapiirteestä pienempiä piirteitä. Move Sets to Feature -valinta luo valitusta kappaleesta uuden geometriapiirteen, joka voidaan siirtää Create-välilehden yläreunassa olevien geometrialinkkien alle (kuvio 17). Linkkien alle liitetään ne geometriat, joiden halutaan liikkuvan samoilla muuttujilla.



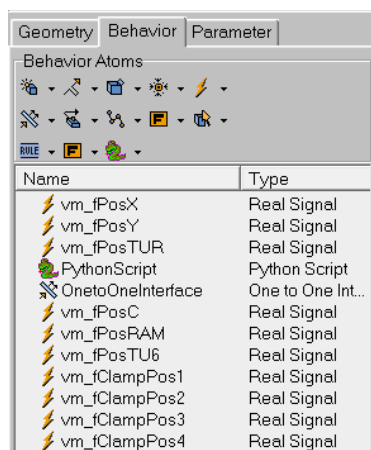
Kuvio 16. Geometrian valinta piirrepuusta



Kuvio 17. Move Sets To Feature -valinta

## 5.4.2 Signaalien luominen

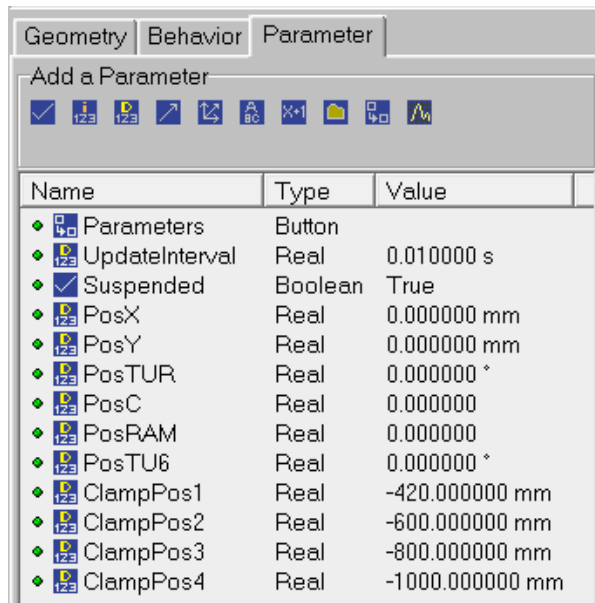
Signaalit luodaan Create-välilehden alta löytyvältä Behavior-välilehdeltä (kuvio 18). Behavior-välilehden alta voidaan lisätä myös erilaisia antureita, reitityssääntöjä ja materiaalinluontikomponentteja. SGe-peruskoneeseen luotiin ainoastaan real-signaaleita, jotka liittyvät esimerkiksi koordinaattipöydän liikkeisiin. RS-kulmaleikkurille luotiin myös integer-signaaleita, jotka liittyvät kitapöydän avaamiseen arvolla 0 pöydän ollessa kiinni ja arvolla 100 pöydän ollessa auki. Pöydän tilaa ilmaisevat myös boolean-signaalit, joiden arvo on joko 0 tai 1. Kaikki signaalit luodaan samoilla TwinCATin muuttujalistalla olevien muuttujien nimillä.



Kuvio 18. Signaalien luominen Behavior-välilehdeltä

### 5.4.3 Parametrien luominen

Parametrit luodaan Create-välilehden alta löytyvältä Parameter-välilehdeeltä (kuvio 19). Parametreja voidaan luoda signaalien tapaan erityyppisiä. SGe-peruskoneen akselimuuttujille luotiin real-parametreja, kun taas RS-kulmaleikkurille luotiin real-parametrien lisäksi myös integer- ja boolean-parametreja. Parametreja voidaan ohjata käsin Param-välilehdeeltä tai käyttää niitä PythonScript-koodissa.



Kuvio 19. Parametrien luominen Parameter-välilehdeeltä

### 5.4.4 PythonScriptin lisääminen

PythonScript-komponentti luodaan Behavior-välilehden alta. Sitä käytetään, kun on tarvetta määrittellä tarkemmin signaalien ja parametrien toimintaa. PythonScript-komponenttia käytettiin SGe-peruskoneessa sekä RS-kulmaleikkurissa. Ensin määriteltiin koodiin `comp=getComponent()`. Tämän avulla `comp`-komennolla voidaan kutsua komponenttien tietoja. Sitten määriteltiin, että TwinCAT-ohjauksessa olevat muuttujat vastaavat 3DCreateen luotuja signaaleita. Esimerkiksi ohjauksessa oleva muuttuja `vm_fPosX` sisältää koordinaattipöydän X-akselin paikkatiedon. Behavior-välilehdelle luotiin real-signaali samalla `vm_fPosX`-nimellä ja PythonScriptin koodilla `vm_fPosX = comp.findBehaviour("vm_fPosX")` ne yhdistetään keskenään (kuvio 20). Samanlainen koodi suoritetaan jokaiselle yhdistettävälle muuttujalle ja signaalille.

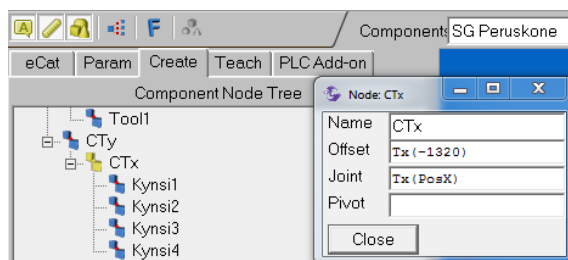
```

Script: SG Peruskone:PythonScript
1 from vcScript import *
2
3 def OnSignal( signal ):
4     comp = getComponent()
5
6     vm_fPosX = comp.findBehaviour("vm_fPosX")
7     vm_fPosY = comp.findBehaviour("vm_fPosY")
8     vm_fPosTUR = comp.findBehaviour("vm_fPosTUR")
9     vm_fPosC = comp.findBehaviour("vm_fPosC")
10    vm_fPosRAM = comp.findBehaviour("vm_fPosRAM")
11    vm_fPosTU6 = comp.findBehaviour("vm_fPosTU6")
12    vm_fPosC2 = comp.findBehaviour("vm_fPosC2")
13    vm_fPosTUR2 = comp.findBehaviour("vm_fPosTUR2")
14    vm_fPosLCUT = comp.findBehaviour("vm_fPosLCUT")
15    vm_fPosLFOC = comp.findBehaviour("vm_fPosLFOC")
16    vm_fPosCEH = comp.findBehaviour("vm_fPosCEH")
17    vm_fPosCEV = comp.findBehaviour("vm_fPosCEV")
18    vm_fPosRAM2 = comp.findBehaviour("vm_fPosRAM2")
19    vm_fPosRRH = comp.findBehaviour("vm_fPosRRH")
20    vm_fClampPos1 = comp.findBehaviour("vm_fClampPos1")
21    vm_fClampPos2 = comp.findBehaviour("vm_fClampPos2")
22    vm_fClampPos3 = comp.findBehaviour("vm_fClampPos3")
23    vm_fClampPos4 = comp.findBehaviour("vm_fClampPos4")
24
25
26    if signal == vm_fPosX:
27        comp.PosX = vm_fPosX.Value
28    elif signal == vm_fPosY:
29        comp.PosY = vm_fPosY.Value
30    elif signal == vm_fPosTUR:
31        comp.PosTUR = vm_fPosTUR.Value
32    elif signal == vm_fPosC:
33        comp.PosC = vm_fPosC.Value
34    elif signal == vm_fPosRAM:
35        comp.PosRAM = vm_fPosRAM.Value
36    elif signal == vm_fPosTU6:
37        comp.PosTU6 = vm_fPosTU6.Value
38    elif signal == vm_fClampPos1:
39        comp.ClampPos1 = vm_fClampPos1.Value
40    elif signal == vm_fClampPos2:
41        comp.ClampPos2 = vm_fClampPos2.Value
42    elif signal == vm_fClampPos3:
43        comp.ClampPos3 = vm_fClampPos3.Value
44    elif signal == vm_fClampPos4:
45        comp.ClampPos4 = vm_fClampPos4.Value
46

```

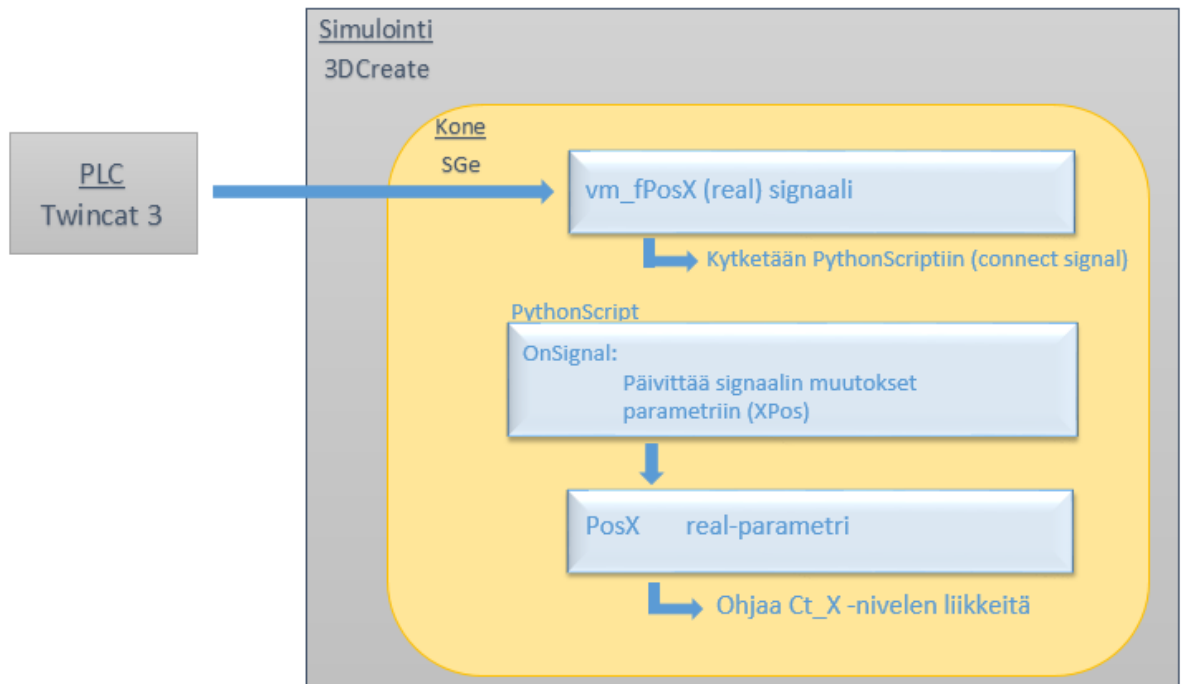
Kuvio 20. SGe-peruskoneen PythonScript-koodi

Koodilla *if signal == vm\_fPosX: Comp.PosX = vm\_fPosX.Value* määriteltiin, että jos ohjaukselta saapuvan *vm\_fPosX*-muuttujan arvo muuttuu, niin se päivittää muutokset *PosX*-parametrin arvoon. Päivitetyllä parametrin arvolla ohjataan suoraan komponenttien piirrepuussa olevaa linkkiä (kuvio 21). Akselin arvoksi määritettiin *Tx(PosX)*, joka tarkoittaa, että *PosX*-parametrin muutokset liikuttavat X-akselia lineaarisesti parametrin arvon mukaan. Samanlainen koodi suoritetaan jokaiselle muuttujalle.



Kuvio 21. Parametrin arvo ohjaa akselia lineaarisesti























Signaalien reitti alkaa TwinCAT-ohjaukseen virtuaalikonetta varten luoduista muuttujista. Ohjauksen muuttujien arvot päivittyvät Tuluksesta tulevan NC-koodin mukaan. 3DCreateen luotiin ohjauksen muuttujien nimen ja tyyppin mukaisia signaaleita, joita ohjataan PythonScript-koodilla. PythonScriptin avulla päivitetään signaalien muutokset parametreihin. Parametrit kytketään linkkeihin ja niiden arvojen muutokset ohjaavat nivelien liikkeitä (kuvio 22).



Kuvio 22. Signaalien reititys

## 5.5 Ohjauksen muutokset

Työssä käytettiin oikean SGe6-koneen ohjausta, joka oli tehty Beckhoff TwinCAT 3 -ohjelmalla. Kun käytettiin oikean koneen olemassa olevaa ohjausta, päätettiin, että ohjaukseen ei tehdä virtuaalikoneen vuoksi muutoksia. Näin virtuaalikoneesta saatiin todenmukainen. Virtuaalikonetta varten luotiin oma GlobalVariable-muuttujalista (kuvio 23). Näitä muuttujia käytettiin rajapintana ohjauksen ja simulointimallin välillä, kun ohjausta kytkettiin simulointimalliin. Erillinen muuttujalista mahdollistaa, että samoja muuttujia voidaan käyttää tulevaisuudessa myös muista koneista tehtävillä virtuaalikoneilla. Ohjauksen ja virtuaalikoneen päivittäminen on myös huomattavasti helpompaa, kun ohjaukseen ei tehty muutoksia.

TwinCAT_Device.BasicMachinePlc.Global_VirtualModel			
Expression	Type	Value	Comment
 vm_fPosX	LREAL	0	[mm]
 vm_fPosY	LREAL	1580	[mm]
 vm_fPosTUR	LREAL	0	[deg]
 vm_fPosC	LREAL	0	[deg]
 vm_fPosRAM	LREAL	0	[mm]
 vm_fPosRS	LREAL	90.5	[mm]
 vm_fPosFORM	LREAL	0	[mm]
 vm_fPosTU6	LREAL	0	[ Turret [deg] ; Tap [mm] ]
 vm_fPosC2	LREAL	0	[deg]
 vm_fPosTUR2	LREAL	0	[deg]
 vm_fPosLCUT	LREAL	0	[mm]
 vm_fPosLFOC	LREAL	0	[mm]
 vm_fPosCEH	LREAL	0	[mm]
 vm_fPosCEV	LREAL	0	[mm]
 vm_fPosRAM2	LREAL	0	[mm]
 vm_fPosRRH	LREAL	0	[deg]
 vm_fClampPos1	LREAL	-420	[mm]
 vm_fClampPos2	LREAL	-600	[mm]
 vm_fClampPos3	LREAL	-800	[mm]
 vm_fClampPos4	LREAL	-1000	[mm]
 vm_nRsTablePos	INT	0	RS Table Position ; 0 = Close ; 100 = Open
 vm_nRsSlideTablePos	INT	0	RS "Slide" Table Position ; 0 = Close ; 100 = Open

Kuvio 23. VirtualModel-muuttujalista

## 5.6 TwinCAT-ohjauksen kytkeminen simulointimalliin

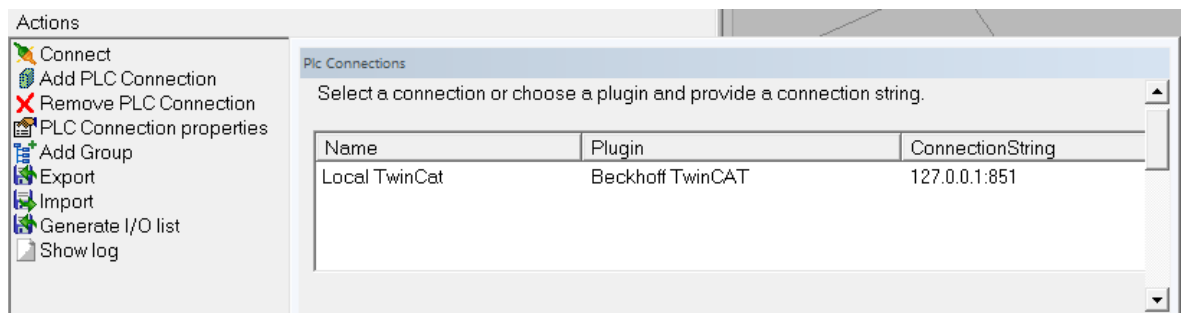
Simulointimalliin luotujen geometrioiden, signaalien, parametrien ja PythonScript-koodien sekä TwinCAT-ohjaukseen tarvittavien muuttujien pohjalta luotiin PLC-yhteys virtuaalikonetta varten. Yhteyden luomisessa käytettiin 3DCreaten PLC Add-on -lisäosaa. PLC Add-onissa on tuki TwinCAT 3 -ohjelmaan, joten se toimi valmiina rajapintana yhteyden luomisessa. Jotta PLC-yhteys voitiin luoda, tuli TwinCAT-ohjauksen olla RUN-tilassa ja PLC Add-onin Connect-tilassa (kuvio 24).

Expression	Type	Value	Comment
vm_fPosX	REAL	202.023800...	[mm]
vm_fPosY	REAL	1292.95955	[mm]
vm_fPosTUR	REAL	180	[deg]
vm_fPosC	REAL	45	[deg]
vm_fPosRAM	REAL	37.6404406...	[mm]
vm_fPosRS	REAL	90.5	[mm]
vm_fPosFORM	REAL	-13	[mm]
vm_fPosTU6	REAL	0	[ Turret [deg] ; Tap [mm] ]
vm_fPosC2	REAL	45	[deg]
vm_fPosTUR2	REAL	0	[deg]
vm_fPosLCUT	REAL	0	[mm]
vm_fPosLFOC	REAL	0	[mm]
vm_fPosCEH	REAL	0	[mm]
vm_fPosCEV	REAL	0	[mm]
vm_fPosRAM2	REAL	0	[mm]
vm_fPosRRH	REAL	0	[deg]
vm_fClampPos1	REAL	-148.283	[mm]
vm_fClampPos2	REAL	-799.699	[mm]
vm_fClampPos3	REAL	-1451.1150...	[mm]
vm_fClampPos4	REAL	-2102.531	[mm]
vm_nRsTablePos	INT	0	RS Table Position ; 0 = Close ; 100 = Open
vm_nRsSlideTablePos	INT	0	RS "Slide" Table Position ; 0 = Close ; 100 = Open

Kuvio 24. TwinCAT RUN-tilassa

### 5.6.1 PLC-yhteyden luominen

PLC-yhteys luotiin PLC Add-onissa Add PLC Connection -painikkeella. Painikkeesta avautuu lista kaikista RUN-tilassa olevista PLC-ohjauksista. Listasta valitaan haluttu ohjaus ja luodaan PLC-yhteys. Tässä työssä valittiin RUN-tilassa oleva Local Twincat -yhteys (kuvio 25).



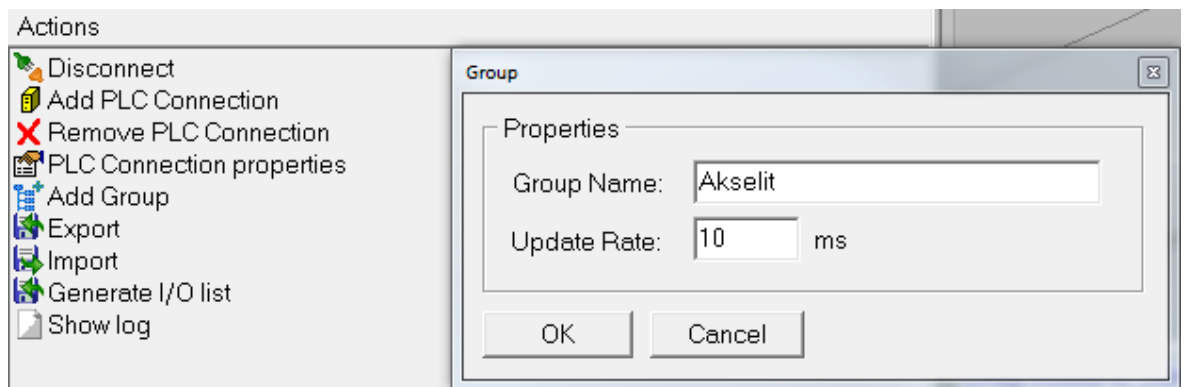
Kuvio 25. PLC-yhteyden luominen Add PLC Connection -painikkeella

### 5.6.2 Ryhmän luominen

Yhteyden luomisen jälkeen luotiin muuttujien yhdistämistä varten ryhmä Add Group -painikkeella. Ryhmä nimettiin Akselit-nimellä ja ryhmän päivitysnopeudeksi asetettiin 10 millisekuntia (kuvio 26). Tämä tarkoittaa, että ryhmän sisällä yhdistettyjen

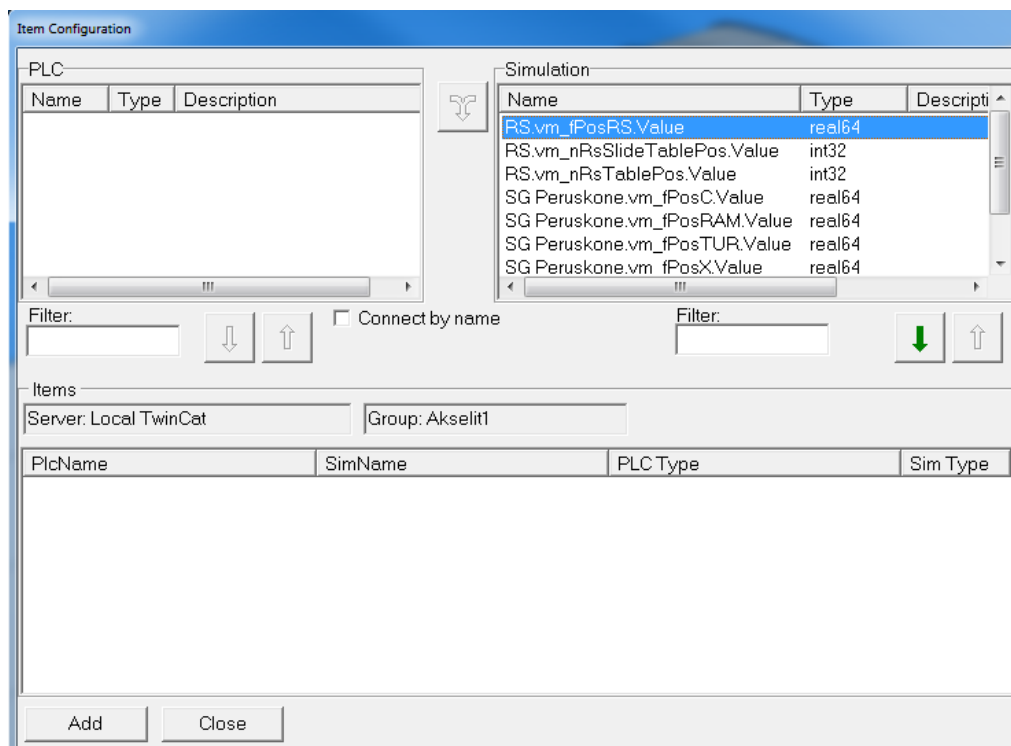


muuttujien arvoja verrataan ja päivitetään TwinCAT-ohjauksen muuttujien arvoihin 10 millisekuntin välein. Lyhyen päivitysajan vuoksi simulointimalli pyörii sulavasti.



Kuvio 26. Ryhmän luominen muuttujien paritusta varten

OK-painikkeesta avautuu Item Configuration -ikkuna (kuvio 27). Ikkunan vasemmassa laidassa PLC-otsikon alla näkyvät kaikki ohjaukselta löytyvät muuttujat. Ikkunan oikeassa laidassa Simulation-otsikon alla näkyvät kaikki simulointimalliin luodut signaalit. Yhdistäminen tapahtuu valitsemalla PLC-listasta ja Simulation-listasta toisiaan vastaavat muuttujat ja painamalla listojen välissä olevaa nuolipainiketta, joka luo parista nimikkeen ikkunan alareunaan.



Kuvio 27. Item Configuration -ikkuna

Item Configuration -ikkunan PLC-lista on tyhjä TwinCAT-ohjauksen suuruuden ja satojen muuttujien vuoksi. Kyseessä oli PLC Add-onissa oleva ohjelmointivirhe, jonka vuoksi listassa ei näy ainuttakaan muuttujaa, jos ohjauksessa on liian suuri määrä muuttujia. Tämän takia paritus täytyi hoitaa manuaalisesti kirjoittamalla halutut parit Notepad++-ohjelmalla XML-tiedostoon (kuvio 28).

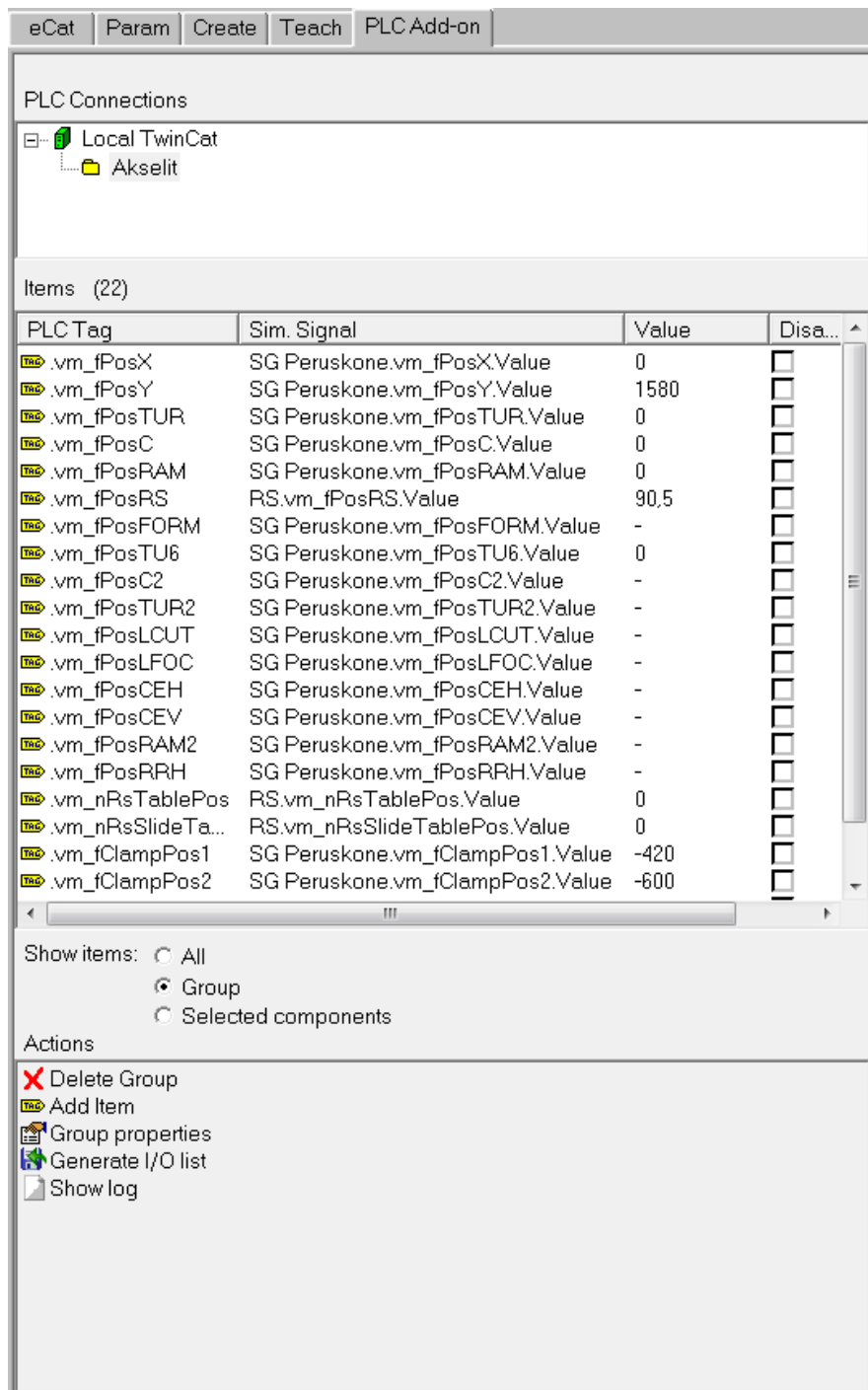
```

1  <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
2  <PlcAddonConfiguration>
3    <Server plugin="Beckhoff TwinCAT" name="Local TwinCat" connectionString="127.0.0.1:851">
4    </Server>
5    <Session/>
6    <Group name="Akselit" updateRate="10">
7      <Item plcItemName="vm_fPosX" vcItemName="SG Peruskone.vm_fPosX.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
8      <Item plcItemName="vm_fPosY" vcItemName="SG Peruskone.vm_fPosY.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
9      <Item plcItemName="vm_fPosTUR" vcItemName="SG Peruskone.vm_fPosTUR.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
10     <Item plcItemName="vm_fPosC" vcItemName="SG Peruskone.vm_fPosC.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
11     <Item plcItemName="vm_fPosRAM" vcItemName="SG Peruskone.vm_fPosRAM.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
12     <Item plcItemName="vm_fPosRS" vcItemName="RS.vm_fPosRS.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
13     <Item plcItemName="vm_fPosRM" vcItemName="SG Peruskone.vm_fPosRM.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
14     <Item plcItemName="vm_fPosTU6" vcItemName="SG Peruskone.vm_fPosTU6.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
15     <Item plcItemName="vm_fPosC2" vcItemName="SG Peruskone.vm_fPosC2.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
16     <Item plcItemName="vm_fPosTUR2" vcItemName="SG Peruskone.vm_fPosTUR2.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
17     <Item plcItemName="vm_fPosLCUT" vcItemName="SG Peruskone.vm_fPosLCUT.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
18     <Item plcItemName="vm_fPosLFOC" vcItemName="SG Peruskone.vm_fPosLFOC.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
19     <Item plcItemName="vm_fPosCEH" vcItemName="SG Peruskone.vm_fPosCEH.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
20     <Item plcItemName="vm_fPosCEV" vcItemName="SG Peruskone.vm_fPosCEV.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
21     <Item plcItemName="vm_fPosRM2" vcItemName="SG Peruskone.vm_fPosRM2.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
22     <Item plcItemName="vm_fPosRR" vcItemName="SG Peruskone.vm_fPosRR.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
23     <Item plcItemName="vm_nRsTablePos" vcItemName="RS.vm_nRsTablePos.Value" accessType="Read/Write" dataType="int16" disabled="false"/>
24     <Item plcItemName="vm_fClampPos1" vcItemName="SG Peruskone.vm_fClampPos1.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
25     <Item plcItemName="vm_fClampPos2" vcItemName="SG Peruskone.vm_fClampPos2.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
26     <Item plcItemName="vm_fClampPos3" vcItemName="SG Peruskone.vm_fClampPos3.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
27     <Item plcItemName="vm_fClampPos4" vcItemName="SG Peruskone.vm_fClampPos4.Value" accessType="Read/Write" dataType="real64" disabled="false"/>
28   </Group>
29 </Session>
30 </PlcAddonConfiguration>

```

Kuvio 28. XML-tiedosto muuttujien ja signaalien pareista

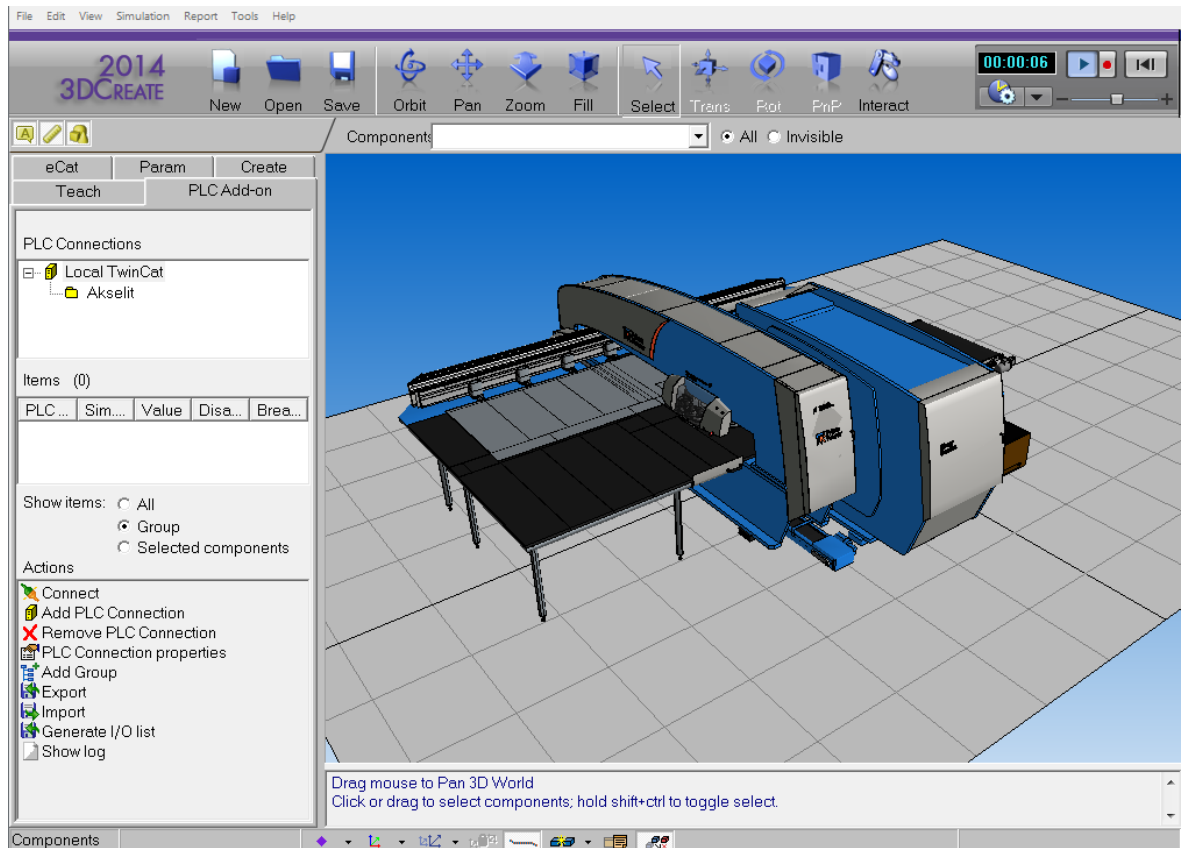
Samalle kommentoriville kirjoitettiin ryhmän, ohjauksessa olevan muuttujan sekä simulointimallin signaalin nimi sekä datatyyppi. Valmis XML-tiedosto tuotiin PLC Add-oniin Import-painikkeella, jonka avulla saatiin ohitettua Item Configuration -ikkunan käyttö. Valmiit parit tulevat näkyviin Items-listan alle (kuvio 29). Menetelmä osoittautui jopa kätevämmäksi kuin normaali paritustapa, sillä Item Configuration -ikkunassa luetaan kaikki mahdolliset muuttujat ohjauksen ja simuloinnin tiedoista, jolloin listat voivat olla hyvinkin pitkiä ja parien löytäminen vaikeaa. Manuaalisesti kirjoittamalla saatiin helposti yhdistettyä vain halutut muuttujat ja signaalit.



Kuvio 29. Paritetut muuttujat ja signaalit

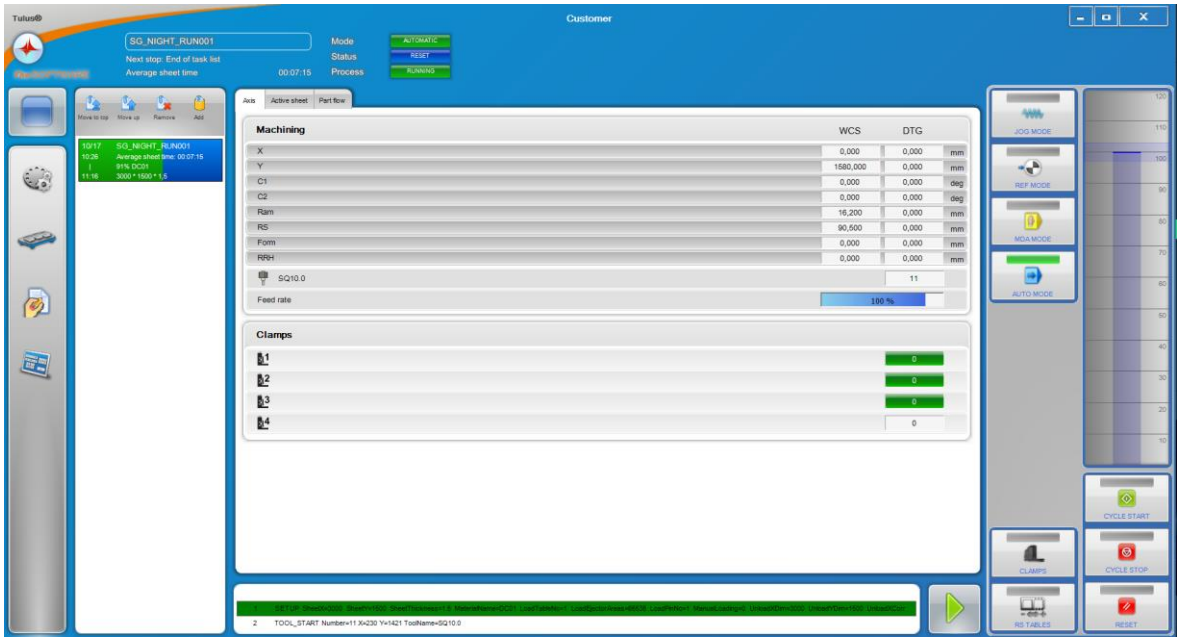
## 5.7 Virtuaalikoneen testaus

Kun muuttujat ja signaalit oli paritettu, voitiin aloittaa virtuaalikoneen testaus. Yhteyden luomista varten 3DCreate oli asetettava RUN-tilaan, jotta simulointimalli liikkuisi ohjauksen ollessa käynnissä. TwinCATin ja PLC Add-onin välille asetettiin yhteys PLC Add-onin Actions-välilehdellä olevalla Connect-painikkeella (kuvio 30).

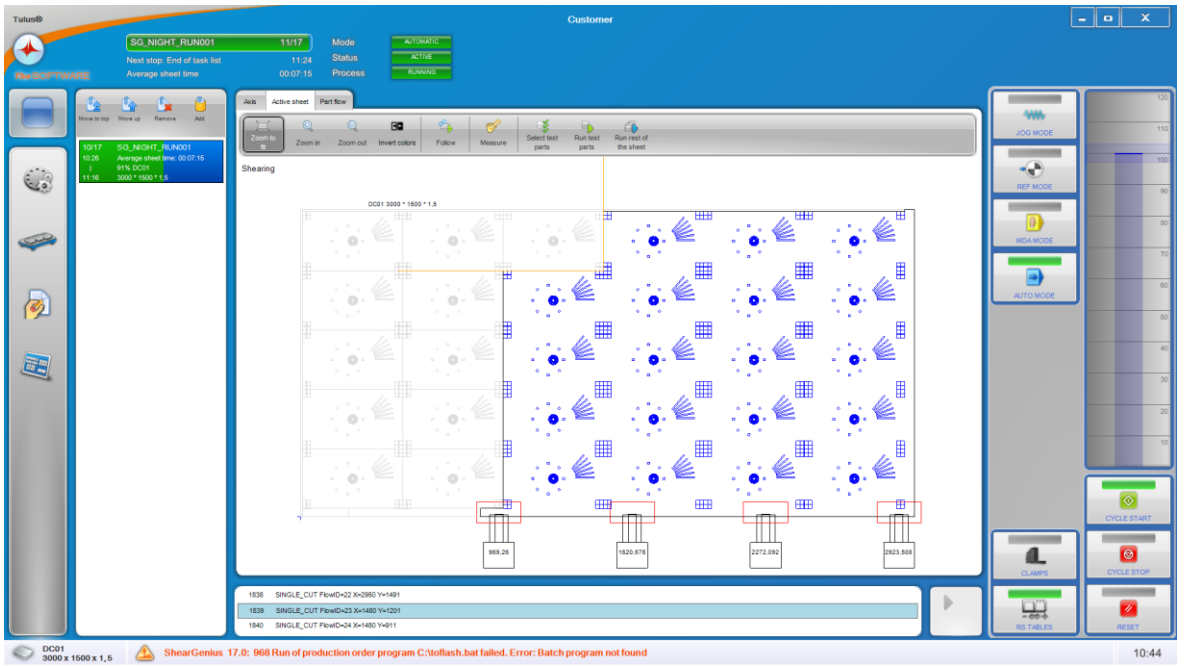


Kuvio 30. RUN-tila asetetaan oikeasta yläkulmasta ja PLC-yhteys Connect-painikkeella Actions-välilehdeltä

Ohjaukselta tulevien muuttujien arvoja voitiin tarkkailla reaaliaikaisesti Items-listalla olevien parien avulla. Virtuaalikoneen testaukseen käytettiin *SG\_NIGHT\_RUN001*-nimistä nestiä, joka tuotiin Tuluksen työjonoon. Sen avulla voitiin tarkkailla, toimivatko kaikki yhdistetyt muuttujat oikealla tavalla. Kone ajettiin ensin Tuluksen Ref Modella referenssiin, jonka jälkeen kytkettiin päälle Auto Mode (kuvio 31). Tämän jälkeen työjonoon lisättiin valmistettavia ohutlevyjä ja käynnistettiin työjono. Kun työjono oli aktivoitu, käynnistettiin kone Cycle Start -painikkeella. Ohjelma alkoi kynsien paikoituksella. Paikoituksen valmistuttua ohutlevy asemoitiin automaattisesti kynsiin ja Clamps-painikkeella laitettiin kynnet kiinni, jolloin ne tarttuivat levyyn. Cycle Start -painiketta uudestaan painamalla varsinainen ohjelma lähti käyntiin ja simulointimalli alkoi liikkumaan Tuluksesta TwinCAT-ohjaukselle menevän NC-koodin mukaan. Simulointimalli liikkui samalla tavalla kuin oikea kone liikkuisi. Ohjelman etenemistä voitiin tarkkailla Tuluksen Active Sheet -välilehdeltä (kuvio 32.), joka näytti ohjelman edetessä jokaisen lävistyksen ja kulmaleikkuun.



Kuvio 31. Tulus Auto Mode -tilassa ja työjono aktiivisena



Kuvio 32. Active Sheet -välilehdeltä voidaan tarkkailla ohjelman etenemistä

## 6 TULOKSET JA JATKOKEHITYS

Työn alussa luotiin virtuaalikoneen tarvekartoitus, jota hyödynnettiin projektin esittelyssä konsernin sisällä. Tarvekartoituksen jälkeen saatiin rakennettua toimiva yhteys kaikkien työssä käytettyjen ohjelmistojen välille ja tärkeimpänä virtuaalinen käyttöympäristö TwinCAT 3 -ohjauksella. Simulointimalliin kytkettiin suurin osa koneen ulkoisesti näkyvissä olevista akseleista. Simulointimallista oli karsittu koneen sisällä olevat komponentit, joten sillä ei voitu vielä testata kaikkia oikean koneen liikkeitä. Työssä tehtiin paikallinen työasema, joka pyörii kannettavalla tietokoneella. Koneelle asennettiin kaikki tarvittavat ohjelmistot ja virtuaalikoneen pyörittämiseen ei tarvita muita laitteita. Virtuaalikone tehtiin Prima Powerin TheCOMBI-tuoteperheeseen kuuluvasta Shear Genius SGe6 -koneesta, joka kytkettiin TwinCAT 3 -ohjaukseen. Simulointimalli tehtiin mahdollisimman kevyeksi karsimalla mallista pienempiä komponentteja. Malliin kytkettiinkin vain näkyvimmit liikkuvat akselit, vielä tässä vaiheessa ei edes tavoiteltu yksityiskohtaisempaa mallia. Opinnäytetyö toimi yrityksessä pilottiprojektina virtuaalikoneiden tekemiselle ja tutkimiselle.

Konsepti osoittautui toimivaksi ja käyttökelpoiseksi yrityksen käytössä, joten virtuaalikoneita tullaan kehittämään myös jatkossa. Kun 3DCreateen saadaan tuki useammalle PLC-ohjaukselle, voidaan virtuaalikoneita tehdä suuremmista konekonfiguraatioista ja useista automaatiolaitteista. Jatkossa on hyvin mahdollista, että jokainen yrityksen kone voidaan löytää omasta virtuaalikonekirjastosta. Työssä tehtyä virtuaalikonetta voi kehittää liittämällä siihen vielä enemmän akselimuuttujia, jolloin siitä saadaan yhä enemmän oikeaa konetta vastaava. Jos virtuaalikonetta käytetään tuotekehitykseen, siihen täytyy liittää kaikki ohjauksen akselimuuttujat, jotta se toimisi kuin oikea kone. Useamman ohjauksen tuki luo mahdollisuuden automaatiolaitteiden modulaarisuuteen, jolloin voidaan helposti lisäillä laitteita ja vertailla niiden toimintoja.

Pilvipalveluita tullaan myös hyödyntämään luomalla virtuaalinen koulutusympäristö. Koulutusympäristön avulla virtuaalikoneita voidaan käyttää etänä kaikkialta ja se helpottaa varsinkin koneiden koulutusta. Koulutus voidaan hoitaa osittain internetin välityksellä ilman oikeaa konetta.

## 7 POHDINTA

Työn tavoitteet saavutettiin niille asetetuilla määreillä. Tarvekartoituksen avulla voitiin esittää virtuaalikoneen toimintaa ja saatiin hyviä kehitysideoita. Työn päätavoite saavutettiin luomalla yhteys 3D-simulointimallin ja PLC-ohjauksen välille.

Virtuaalikoneen käyttötarkoitukset ja projektin suunta muuttui projektin edetessä. Alun perin virtuaalikonetta rakennettiin messu- ja myyntitarkoituksiin visuaaliseksi esittelyvälineeksi korvaamaan oikea kone. Tämä suuntaus muuttui, koska malli ei ole tarpeeksi yksityiskohtainen esittelykäyttöön. Yksityiskohtainen simulointimalli on hyvin raskas ja ei välttämättä toimisi kunnolla. Virtuaalikonetta lähdettiinkin viemään koulutus- ja testauskäyttöön. Koneiden koulutus tapahtuu asiakkaille tavallisesti vasta, kun koneiden käyttöönotto on jo suoritettu asiakkaalla. Virtuaalikoneella koneen ja ohjelmistojen peruskoulutus voidaan aloittaa asiakkaille samalla, kun oikeita koneita ollaan vasta käyttöönottamassa. Tämä lyhentää huomattavasti koulutusaikaa, kun koulutettavilla on jo oikeille koneille päästessään hyvä käsitys koneiden toiminnasta, eikä tällöin kulu aikaa perusasioiden opettamiseen. Virtuaalikone lyhentää myös ohjelmistotestausta, kun ohjelmaa ei tarvitse lähteä testaamaan oikeilla koneilla.

Haasteita tuotti 3DCreaten PLC Add-on lisäosa, jossa ei ollut tukea usealle PLC-ohjaukselle, jonka vuoksi ei voitu rakentaa isompia järjestelmiä. Ohjauksen suuri määrä muuttujia tuotti myös hankaluuksia, kun PLC Add-on ei pystynyt niitä käsittelemään. Ongelmaan löytyi loppujen lopuksi ratkaisu ja kytkeminen onnistui hyvin, kun ohjaukseen ja simulointimalliin oli tehty tarvittavat valmistelut.

Konsepti saatiin toimimaan odotetulla tavalla. Ohjelmistojen yhteensopivuus paranee, kunhan 3DCreatesta ja PLC Add-onista saadaan päivitykset, jotka korjaavat niissä esiintyneet virheet.

Virtuaalikoneen kuluja tulee tietokoneista, ohjelmistolisensseistä ja mahdollisista pilvipalveluista. Virtuaalikoneella säästöjä syntyy koulutuksessa, tuotekehityksessä ja testauksessa säästetystä ajasta, matkustuskulujen vähenemisestä, materiaalikustannusten vähenemisestä. Virtuaalikoneet tehostavat yrityksen toimintaa huomattavasti.

## LÄHTEET

- Beckhoff. 2012. TwinCAT 3 - eXtended Automation (XA). [PDF-tiedosto]. Beckhoff Automation GmbH. [Viitattu 1.4.2016]. Saatavissa: [http://download.beckhoff.com/download/document/catalog/Beckhoff\\_TwinCAT3\\_042012\\_e.pdf](http://download.beckhoff.com/download/document/catalog/Beckhoff_TwinCAT3_042012_e.pdf)
- Beckhoff. 2016. New Automation Technology. [WWW-dokumentti]. Beckhoff Automation GmbH. [Viitattu 1.4.2016]. Saatavissa: <http://www.beckhoff.fi/>
- Delfoi. 2013. Toiminnan kehittäminen ja simulointi ennen rakentamista. [PDF-tiedosto]. Delfoi Oy. [Viitattu 28.3.2016]. Saatavissa: [https://tapahtumat.tekes.fi/uploads/077e2249/Jaatinen\\_Ossi\\_delfoi-3552.pdf](https://tapahtumat.tekes.fi/uploads/077e2249/Jaatinen_Ossi_delfoi-3552.pdf)
- Keinänen, T. Kärkkäinen, P. Lähetkangas, M. & Sumujärvi, M. 2007. Automaatiojärjestelmien logiikat ja ohjaustekniikat. Helsinki: WSOY Oppimateriaalit Oy.
- Liang, Q. Li, L. 2011. The Study of Soft PLC Running System. [Verkoartikkeli]. Procedia Engineering, 2011, Vol.15, 1234–1238. [Viitattu 19.3.2016]. Saatavissa: <http://www.sciencedirect.com/science/article/pii/S1877705811017292>
- Liu, Z. Suchold, N. & Diedrich, C. 2012. Virtual Commissioning of Automated Systems. Teoksessa: F. Kongoli. (Toim.) Automation. InTech, 132–148.
- Lambert, J. Ei päiväystä. Dana Carvey, Industrial Internet, and the Internet of Things. [Valokuva]. MoneyTech Search Group. [Viitattu 7.4.2016]. Saatavissa: <http://moneytechsearch.com/wp-content/uploads/2015/05/Internet-of-things-lq.jpg>
- Prima Power. 2012. Prima Power premises, Kauhava. [Valokuva]. PRIMA INDUSTRIE S.p.A. [Viitattu 1.4.2016]. Saatavana Prima Power Photo Bank -tietokannasta. Vaatii käyttöoikeuden.
- Prima Power. 2013. Prima Power Tulus Cell. [Prima Powerin sisäinen materiaali]. Saatavana Prima Power The BOX -tietokannasta. Vaatii käyttöoikeuden.
- Prima Power. 2014. NC Express e<sup>3</sup> Programming efficiency for a productive fabrication process. PRIMA INDUSTRIE S.p.A. Ohjelmistoesite.
- Prima Power. 2015a. Our history. [WWW-dokumentti]. PRIMA INDUSTRIE S.p.A. [Viitattu 14.3.2016]. Saatavissa: <http://www.primapower.com/fi/about-us/history/>



- Prima Power. 2015b. Prima Power - Next to You. PRIMA INDUSTRIE S.p.A. Yritysesite.
- Prima Power. 2016a. NC Express e<sup>3</sup>. [Valokuva]. PRIMA INDUSTRIE S.p.A. [Viitattu 7.4.2016]. Saatavana Prima Power The BOX -tietokannasta. Vaatii käyttöoikeuden.
- Prima Power. 2016b. Tulus<sup>®</sup> Cell. [Valokuva]. PRIMA INDUSTRIE S.p.A. [Viitattu 7.4.2016]. Saatavana Prima Power The BOX -tietokannasta. Vaatii käyttöoikeuden.
- Raunio, H. 2014. Teollisuus 4.0 - "Suomen oltava kilpailukykyinen vaihtoehto, kun teollisuuden paluumuutto Aasiasta Eurooppaan alkaa". [Verkkojulkaisu]. Tekniikka & Talous. [Viitattu 6.4.2016]. Saatavissa: <http://www.tekniikkatalous.fi/tekniikka/2014-09-27/Teollisuus-4.0---Suomen-oltava-kilpailukykyinen-vaihtoehto-kun-teollisuuden-paluumuutto-Aasiasta-Eurooppaan-alkaa-3255926.html>
- Reset. 2014. Teollisuus 4.0 on pian täällä. [Verkkolehtiartikkeli]. Reset-lehti. [Viitattu 6.4.2016]. Saatavissa: <http://reset-lehti.fi/teollisuus-4-0-pian-taalla/>
- Robinson, S. 2004. Simulation: The Practice of Model Development and Use. Chichester: John Wiley & Sons Ltd.
- Siemens. Ei päiväystä. Teollisuus 4.0. [WWW-dokumentti]. Siemens AG. [Viitattu 6.4.2016]. Saatavissa: [http://www.siemens.fi/fi/industry/teollisuus/teollisuus\\_4\\_0.htm](http://www.siemens.fi/fi/industry/teollisuus/teollisuus_4_0.htm)
- Shannon, R. 1998. Introduction to the art and science of simulation. [WWW-dokumentti]. Proceedings of the 1998 Winter Simulation Conference 13-16.1.1998. [Viitattu 29.3.2016]. Saatavissa: <http://cecs.wright.edu/~fciarall/ISE195/Readings/ShannonSimulationART.pdf>
- Visual Components. 2016a. Engineering tool. [WWW-dokumentti]. Visual Components Oy. [Viitattu 29.3.2016]. Saatavissa: <http://www.visualcomponents.com/solutions/engineering-tool/>
- Visual Components. 2016b. About us. [WWW-dokumentti]. Visual Components Oy. [Viitattu 15.3.2016]. Saatavissa: <http://www.visualcomponents.com/about-us/>
- Visual Components. 2016c. 3DCreate. [WWW-dokumentti]. Visual Components Oy. [Viitattu 15.3.2016]. Saatavissa: <http://www.visualcomponents.com/products/3dcreate/>

Visual Components. 2016d. Increasing control software quality with virtual commissioning. [WWW-dokumentti]. Visual Components Oy. [Viitattu 5.4.2016]. Saatavissa: <http://www.visualcomponents.com/insights/articles/increasing-control-software-quality-with-virtual-commissioning/>

Visual Components. Ei päiväystä. PLC Add-on Manual. [PDF-tiedosto]. Visual Components Oy. [Viitattu 15.3.2016]. Saatavissa: <http://download.visualcomponents.net/elib/2009/Brochures/PLCAdd-on%20Manual.pdf>