



Mia Manninen

## **JATKUVATOIMISEN MINIPILOT-RIKASTAMON KÄYTTÖLIITTY- MÄ**

# **JATKUVATOIMISEN MINIPILOT-RIKASTAMON KÄYTTÖLIITTY- MÄ**

Mia Manninen  
Opinnäytetyö  
Kevät 2013  
Automaatiotekniikan koulutusohjelma  
Oulun seudun ammattikorkeakoulu

# TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu  
Automaatiotekniikan koulutusohjelma

---

Tekijä: Mia Manninen

Opinnäytetyön nimi: Jatkuvatoimisen minipilot-rikastamon käyttöliittymä

Työn ohjaaja: Timo Heikkinen

Työn valmistumislukukausi ja -vuosi: K 2013      Sivumäärä: 38 + 12 liitettä

---

Työ tehtiin Oulu Mining Schoolille (OMS), joka on Oulun yliopistossa toimintansa aloittanut vuorialan koulutus- ja tutkimusverkosto. Työssä suunniteltiin ja toteutettiin rakenteilla olevan minipilot-kokoluokan jatkuvatoimisen rikastamon automaation käyttöliittymä. Työn tarkoituksena oli toteuttaa prosessinohjauksen käyttöliittymä rikastamolle ottaen huomioon tulevien muutos- ja lisästarpeiden helppo integroitavuus järjestelmään.

Valvomon käyttöliittymä toteutettiin Schneider Electricin Vijeo Citect -ohjelmalla. Lisäksi työssä konfiguroitiin historiatietokanta ja raportointikantana käytettävä tuotannonohjausjärjestelmä.

Työn tuloksena toteutettiin rikastamon käyttöliittymä. Lisäksi laadittiin selkeät ohjeet käyttöliittymän ja historia- sekä raportointikantojen ylläpitämiseksi ja laajentamiseksi.

---

Asiasanat: valvomot, käyttöliittymät, rikastamot, automaatiojärjestelmät

## ALKULAUSE

Haluan kiittää Oulu Mining Schoolin projektipäällikkö Ilkka Hynystä mahdollisuudesta toteuttaa lopputyöni minipilot-hankkeen parissa. Lisäksi haluan kiittää Ilkkaa suuresta luottamuksesta työtäni kohtaan ja saamastani suuresta arvostuksesta.

Haluaisin myös lausua suuret kiitokset korvaamattomasta avusta ja tuesta kaikille Schneider Electricin työntekijöille, joihin projektin aikana tutustuin. On ollut ilo saada työskennellä teidän kanssanne.

Oulussa 9.1.2013

Mia Manninen

# SISÄLLYS

TIIVISTELMÄ	3
ALKULAUSE	4
SISÄLLYS	5
1 JOHDANTO	7
2 VALVOMOSUUNNITTELU	8
2.1 Prosessinohjauksen tehtävätyypit	8
2.2 Konseptisuunnittelu	9
2.3 Näyttösivujen suunnittelu	9
2.4 Hälytyksien suunnittelu	11
2.5 Käyttöönotto	12
3 TUTKIMUS- JA TUOTANTOLAITTEISTON SUUNNITTELUN EROT	14
3.1 Automaatioaste	14
3.2 Laatu ja toimitusvarmuus	15
3.3 Toiminnanohjaus	16
3.4 Tuotanto	17
3.5 Varotoimenpiteet	18
3.6 Dokumentointi	19
4 TYÖN TOTEUTUS	20
4.1 Esiselvitykset	20
4.2 Käyttöliittymän toteutus	20
4.2.1 Lähtötiedot	20
4.2.2 Suunnittelu	21
4.2.3 Grafiikan suunnittelu	22
4.2.4 Ohjelmointi	22
4.2.5 Erikoisnäytöt	24
4.3 Testaukset ja käyttöönotto	25
5 TOTEUTETTU KÄYTTÖLIITTYMÄ	28
5.1 Valikkorakenne	28
5.2 Vaahdotuskennostot ja letkulinjat	29
5.3 Moottorit	30
5.4 Hätäseis-toiminto	33

6 YHTEENVETO	34
LÄHTEET	37
LIITTEET	38

# 1 JOHDANTO

Oulu Mining School (OMS) on vuonna 2007 Oulun yliopistossa toimintansa aloittanut vuorialan koulutus- ja tutkimusverkosto, jonka ydintoimijoita ovat geotieteiden laitos, prosessi- ja ympäristötekniikan osasto sekä täydentävien opintojen keskus. Kansainvälisesti korkeatasoisena, tiedekuntien ja tutkimuslaitosten rajat ylittävänä verkostona Oulu Mining School vastaa vuorialalle erikoistuvien geologien ja diplomi-insinöörien peruskoulutuksesta sekä alan täydennyskoulutuksesta. (1.)

Oulu Mining Schoolin yhteyteen rakennettava koerikastamo on maailman ensimmäinen yliopistoympäristössä toimiva rikastamo. Jatkuvatoinen minipilotmittakaavainen tutkimus- ja tuotantolaitos mahdollistaa rikastusprosessin koulutuksen, tutkimuksen ja tuotekehityksen. Laitoksen yhteyteen liittyvään analyysilaboratorioon on sijoitettu erilaisia mineralogiseen tutkimukseen tarkoitettuja analysointilaitteita, kuten röntgenfluoresenssiin perustuva XRF.

Minipilotin prosessilaitteisto on skaalattu mittakaavassa 1:5000 Pyhäsalmen malmikaivoksen rikastusprosessista. Malmimäärä, jota laitoksessa käsitellään, on 20–50 kg/h, ja aluksi tuotetaan kupari- ja sinkkirikastetta.

Tässä insinööriyössä suunnitellaan ja toteutetaan minipilot-rikastamon automaatiojärjestelmään liittyvä valvomo.

## 2 VALVOMOSUUNNITTELU

Valvomosuunnittelu liitetään usein huonetilan fyysiseen suunnitteluun, mutta se kattaa myös tuotannon hallitsemiseksi tarvittavat toiminnot. Sen lisäksi valvomossa hoidetaan myös muita tuotannon vaatimia tehtäviä. Valvomosuunnitteluun kuuluvat valvomotilojen ja kalustuksen suunnittelu sekä käyttöliittymien sisällön ja toiminnan määrittely. (2, s. 35, 123.)

Valvomosuunnittelu jakautuu kolmeen rinnakkaiseen osa-alueeseen eli rakenteelliseen suunnitteluun (valvomotilat, ohjauspaikat ja olosuhteet), käyttöliittymien suunnitteluun (näytöt, hälytykset, vuorovaikutus) ja valvomotyön suunnitteluun (organisaatio, koulutus, ohjeistus). (2, s. 142, 151.)

Tässä työssä keskitytään käyttöliittymän suunnitteluun ja valvomon rakenteellinen suunnittelu on rajattu kokonaan työn ulkopuolelle. Lisäksi toteutetaan valvomotyön suunnitteluun kuuluva ohjeistus ja operaattoreiden koulutus.

Automaatiojärjestelmän käyttöliittymä muodostuu valvomoon ja kentälle sijoitetuista käyttöliittymälaitteista, kuten näppäimistöistä ja näytöistä. Se on automaatiojärjestelmän käyttäjälle näkyvä osa ja tarjoaa käyttäjän tarvitsemaa informaatiota ja vaikutusmahdollisuuksia. Käyttöliittymään kuuluvat myös perinteisemmät ohjauskytkimet ja merkkilamput sekä mahdollinen näkö- ja kuuloyhteys prosessitilaan. Käyttöliittymä on myös näkymä automaatiojärjestelmään, jossa olennaista on informaatioisisältö ja toiminta, ei toteutustekniikka. Yleensä käyttöliittymä kuuluu automaatiojärjestelmän toimitukseen. (2, s. 21.)

### 2.1 Prosessinohjauksen tehtävätyypit

Prosessin ohjaukseen liittyviä primääritehtäviä ovat prosessiin kohdistuvat työtehtävät eli valvonta ja monitorointi, ohjaukset, suunnitellut toimenpiteet (ylös- ja alasajot) sekä häiriötilanteiden hallinta. Normaalisissa tuotannossa operaattorilla on toiminnan tukena ohjeet siitä, kuinka jokin ajotilanne hoidetaan. (2, s. 47, 174.)



Sekundääritehtäviksi luetaan työvälineiden hallinta, erityisesti automaatio- ja informaatiojärjestelmien hallinta ja operointi, informaation ylläpito ja luotettavuuden tarkistaminen. Sekundääritehtävien pitää olla mahdollisimman yksinkertaisia ja näkymättömiä, mutta niiden hallinta vaatii aina jonkin verran osaamista. Näiden tehtävien muutokset aiheuttavat rutiinien hajoamista ja epävarmuutta. (2, s. 47, 51, 174.)

## **2.2 Konseptisuunnittelu**

Konseptisuunnittelun tehtävänä on selvittää tuotteelle asetetut vaatimukset. Se on tuotekustannusten kannalta kriittisin vaihe yhdessä järjestelmätason suunnittelun kanssa, koska sen aikana tehdyt valinnat ja ratkaisut määrittelevät suurimman osan tuotteen minimikustannuksista. (3, s. 254.)

Käyttöliittymäkonseptin lähtökohtia on kaksi perustapausta: uuden valvomon suunnittelu ja vanhan uudistaminen tai laajentaminen. Jos kyseessä on vanhan valvomon uudistaminen, on vanhan käyttöliittymän suunnitteluperusteet otettava mahdollisuuksien mukaan huomioon uuden suunnittelussa. Tämä on erityisen tärkeää, jos olemassa olevan käyttöliittymän rinnalle tuodaan uuden järjestelmän käyttöliittymä. (2, s. 132, 144, 146, 150.)

Käyttöliittymäkonseptissa määritellään käyttäjien ja automaatiojärjestelmän välinen vuorovaikutusperiaate. Konseptissa määritellään toiminnallinen perusrakenne ja suunnittelussa sovellettavat tyyliohjeet eli informaation koodaustavat, värit ja symbolit, operoinnit ja näyttösivutyypit. (2, s. 124, 141–142, 145.)

## **2.3 Näyttösivujen suunnittelu**

Käyttöliittymä toimii kaksisuuntaisena vuorovaikutuskanavana järjestelmän eli prosessin ja käyttäjän välillä. Tyypillisenä vuorovaikutuskanava toimii prosessitietoa esittävä näyttösivu. Näyttösivun tarkoitus on esittää prosessin tilanne ja mahdollistaa sen ohjaus. Näyttösivu voi perustua prosessin toiminnalliseen jaoteluun (prosessikaavionäyttö) tai se voidaan suunnitella johonkin yksittäiseen tilanteeseen tai tehtävään (tehtävänäyttö). Erilaisten näyttösivujen määrittely ja suunnittelu on erittäin tärkeä osa käyttöliittymäsuunnittelua. (2, s. 25–28.)

Prosessia esittävän tiedon jakaminen eri näyttösivuille on tärkeä tehtävä. Tehtävien suorittaminen on helpompaa, jos sivuja joudutaan selaamaan mahdollisimman vähän, mutta toisaalta operaattorin on helposti löydettävä tarvitsemansa tieto. Yleisin jakoperuste on prosessin toiminnalliset kokonaisuudet, joiden pohjalta syntyvät prosessikaavionäytöt. Valvottava prosessi jaetaan valmistuksen kannalta järkeviin kokonaisuuksiin siten, että yleisvalvonnan, yksityiskohtaisen valvonnan ja operoinnin vaatimukset voidaan täyttää. Prosessikaavionäytöjen tarkkuustaso voi vaihdella karkeasta prosessin yleiskuvasta yksityiskohtaiseen toimintoja esittävään tarkkuuteen. (2, s. 170–171, 176.)

Toinen tärkeä sivujen jakoperuste ovat usein toistuvat ja ohjeistetut tehtävät, kuten ajotapamuutokset sekä käynnistykset ja pysäytykset. Näytöillä esitetään tehtävien suorittamisessa tarvittava informaatio ja ohjausmahdollisuus. (2, s.171.)

Tavallisimpia näyttösivutyyppejä ovat

- prosessikaavionäyttö, jota käytetään prosessin tilan seuraamiseen ja prosessin ohjaamiseen
- tehtävänäyttö, joka on suunniteltu tietyn tehtävän suorittamiseen ja ohjausten vaikutusten seuraamiseen
- piirinäyttö, joka on tarkoitettu yhden säätöpiirin, mittauksen tai laitteen yksityiskohtaisten tietojen esittämiseen
- trendinäyttö, yhden tai useamman suureen historiatieto käyrällä esitettynä
- hälytyslista, hälytyksien tilan esittäminen tekstinä sekä niiden kuitaaminen
- raporttinäyttö erilaisia yhteenvetoja ja analyysejä varten
- sekvenssinäyttö, joka esittää sekvenssiohjelman tilat ja askeleet sekä mahdollistaa sekvenssiohjelman etenemisen seurannan
- kunnonvalvontanäyttö, joka sisältää laitteiden kunnonvalvontaa liittyvää tietoa. (2, s. 30.)

Näyttösivujen ohella oleellisia suunnittelukohteita ovat tapahtumat. Tapahtumassa automaatiojärjestelmä tuo tietoa prosessista käyttäjälle eli aloitteen teki-

jänä on automaatiojärjestelmä. Tapahtumia ovat mm. hälytykset ja ilmoitukset. Hälytys kertoo prosessin poikkeavasta tilasta, kun taas ilmoitus on normaaliin toimintaan liittyvä tiedonanto. Lisäksi käyttöliittymiin sisällytetään tavallisesti raportointiominaisuuksia, päiväkirjoja sekä ohjeita ja reseptejä. (2, s. 31–32, 181–182.)

Prosessin hallinnan kannalta tärkeät näyttötyypit, prosessikaavionaäytöt ja tehtävänäytöt, vaativat hyvin yksityiskohtaista suunnittelua. Osa näyttösivuista, kuten trendinäytöt ja diagnostiikkänäytöt, vaativat vain vähän sivukohtaista suunnittelua. (2, s. 169–170.)

## **2.4 Hälytyksien suunnittelu**

Hälytysten suunnittelu on vaativa tehtävä. Prosessien valvonta perustuu paljolti poikkeamien seurantaan, eli prosessin tilaa seurataan aktiivisesti vain yleisellä tasolla ja poikkeamat ilmoitetaan hälytyksillä. Hälytys edellyttää operaattorilta toimenpidettä tai vähintään huomiota. Operaattorin pitää tietää, mitä hälytyksen ilmoittama poikkeama tarkoittaa ja miten tilanteessa tulee toimia. Hälytyksille määritellään kiireellisyysluokat, joita voi olla 2–4 prosessin luonteesta riippuen. Yleisimmin hälytykset esitetään tapahtumalistana, jossa hälytyksestä kerrotaan aikaleima, tilatieto sekä häiriön syy. (2, s. 147–148, 174–175.)

Turvallisuuskriittiset toiminnot voivat vaatia erityistä huomiota käyttöliittymien suunnittelussa. Joissakin turvallisuuskriittisissä tehtävissä toteutetaan ns. turvakäyttöliittymä tavanomaisen käyttöliittymän rinnalle. Toimintaohjeet kriittisessä tilanteessa voivat olla hyvinkin tärkeitä ja niiden suunnittelu ja sijoittelu näytöille on otettava tällöin huomioon. (2, s. 146, 171; 4, s. 24.)

Tämän työ teoriaosuudessa ei tarkemmin perehdytä vakiintuneisiin tapoihin prosessin ja muiden kohteiden informaation koodaustavoista, kuten esimerkiksi värien ja symbolien käyttö. Työn suorituksessa käyttöliittymän ja näytön graafisen suunnittelun periaatteet otetaan ilman muuta tarkasti huomioon.

## 2.5 Käyttöönotto

Tehdastestistä käytetään nimitystä FAT (Factory Acceptance Test). Siinä pyritään testaamaan sovelluskonfiguraatio mahdollisimman pitkälle jo ennen laitteiston toimittamista kohdepaikalle ja havaitsemaan toteutuksen virheet ennen käyttöönottoa. Testissä simuloidaan mittaukset ja ohjaukset testaamalla jokainen I/O sekä koetetaan ristikytkentä simuloimalla kentältä tulevat signaalit oikeykennöillä tai mA-simulaattorilla. (2, s. 188; 4, s. 8, 15.)

Käyttöönotto suoritetaan testaamalla laitteisto paikan päällä esimerkiksi vesiajossa. Testauksesta käytetään nimitystä SAT (Site Acceptance Test). Käyttöönottotestissä voidaan testata suuri osa mittaus- ja ohjauspiireistä. (2, s. 189.)

Käyttöönottoa voi vielä seurata ns. suorituskykytesti, jossa prosessia ajetaan normaalioloissa ja äärioloissa. Tarkoituksena on varmistaa prosessin toimintavarmuus kaikissa tilanteissa. Suorituskykytestienkin jälkeen laitteistoa voidaan joutua korjaamaan ja muuttamaan. (4, s. 17.)

Valvomon käyttöönottoa suunniteltaessa on otettava huomioon mm.

- testauskohteet ja -menettelyt
- mahdollisesti ilmenevien puutteiden käsittely ja korjaus
- kaikkien ohjauspaikkojen testaus ja niiden välisen työnjaon ja tiedonkulun toimiminen
- käyttöliittymän liitynnät ulkoisiin järjestelmiin, jotka on testattava
- mittauksien ja muiden tietojen näkyminen oikein kaikilla näytöillä ja sivuilla
- hälytysrajat, jotka on pyrittävä asettamaan heti oikein tai ainakin turvallisiksi
- hälytyksien toimivuus ja niiden näkyminen näytöillä sekä äänimerkit
- järjestelmän suorituskyvyn arviointi, esim. hälytysruuhkissa. (2, s. 191.)

Valvomoarvioinnin kohteita ovat mm.

- yhteensopivuus eli tiedon esitystavan ja operaattorin oletetun suorituksen tulee olla sopuoinnussa ihmisen suorituskyvyn kanssa

- informaation esitys, onko tieto esitetty helposti ymmärrettävässä muodossa ja annetaanko operoinnista selkeää palautetta
- tilannetietoisuus ja hallittavuus eli operaattori ymmärtää prosessin tilan ja sen kehityssuunnan
- ymmärrettävyys eli kuinka asiaan perehtymätön käyttäjä kykenee käyttämään järjestelmää
- automaatiojärjestelmän kyky pitää prosessisuureet sallituissa rajoissa, erityisesti turvallisuuteen liittyvissä toiminnoissa. (2, s. 198.)

## 3 TUTKIMUS- JA TUOTANTOLAITTEISTON SUUNNITTELUN EROT

Tuotantolaitoksen konseptointi ja suunnitteluvaihe sekä turvallisuusanalyysit ovat erittäin tärkeitä ja aikaavieviä vaiheita. Laitoksen työvaiheet suunnitellaan huolella, sillä muutosten tekeminen valmiiseen laitokseen on erittäin kallista ja vie paljon aikaa.

Tutkimuslaitos, joka on luonteeltaan suunniteltakin enemmän modulaariseksi kokonaisuudeksi, sallii enemmän vapauksia suunnittelun yksityiskohtien suhteen. Lopullinen laitoksen kokoonpano ei ole välttämättä edes selvillä, kun laitteiston rakentaminen aloitetaan.

### 3.1 Automaatioaste

Automaatio- ja valvomosuunnittelun keskeinen asia on sopivan automaatioasteen löytäminen ja toteuttaminen eli toimintojen jako automaatiojärjestelmän ja ihmisen välillä. Automaatioasteella on suuri merkitys työn kokonaisluonteen ja prosessin dynaamisten ilmiöiden oppimisen kannalta. (2, s. 70, 135–136.)

Jokaista tehtävää tulee arvioida suhteessa ihmisen ja järjestelmän kykyyn suorittaa tehtävän vaatimuksista. Arvioitavia piirteitä ovat vaadittu reagointi- ja suoritus aika, toistuvuus, tarkkuus, monimutkaisuus ja turvallisuusvaikutukset. Arvioinnissa on otettava huomioon myös henkilöiden koulutustaso ja kokemus. Tehtävät jaetaan arvioinnin perusteella kategorioihin seuraavasti:

- tehtävät, jotka on automatisoitava
- tehtävät, jotka olisi parempi automatisoida
- tehtävät, jotka sopivat parhaiten ihmiselle
- tehtävät, jotka jaetaan automaation ja ihmisen kesken. (2, s. 135–136.)

Normaalisti rutiininomaiset sekä muistamista ja tarkkuutta vaativat tehtävät automatisoidaan, jolloin prosessin kokonaistilanteen valvonta ja hallinta sekä monimutkaisten poikkeustilanteiden hoitaminen jää operaattorin tehtäväksi.

Automaatioaste vaikuttaa kuitenkin suoraan siihen, miten aktiiviseksi operaattorin rooli työssä muodostuu. Aktiivisesti ohjaukseen osallistuvat ovat paremmin perillä prosessitilanteen kehittymisestä kuin passiivisemmin monitorointia vaativissa ohjauksissa. Korkean automaation sovelluksissa onkin vaarana, että ihminen menettää tuntuman valvottavan prosessin toimintaan. (2, s. 71, 136.)

Automaatioaste vaikuttaa myös käyttöliittymän suunnitteluun sen mukaan ohjaatanko yksittäisiä laitteita vai prosessikokonaisuuksia esimerkiksi sekvensseillä. Automaatioaste on yleensä korkea. Laitoksen operaattoreiden määrä on minimoitu, joten laitoksen ohjaamiseen tarvitaan paljon mittauksia, säätöjä ja automaattisia toimintoja. Normaalit toiminnot suunnitellaan siten, että yksi henkilö pystyy suorittamaan mahdollisimman ison osan laitoksen toiminnoista. Poikkeustilanteissa, kuten kylmäkäynnistys, vaaditaan enemmän toimenpiteitä ja henkilöstöäkin on enemmän. (4, s. 10–11, 23.)

Automaatioaste on tutkimuslaitteistossa varsin matala verrattuna prosessituotantoon isoissa laitoksissa. Tällä halutaan varmistaa ennen kaikkea mahdollisuus prosessin muunneltavuuteen ja tarvittaessa mahdollisuus ohjata prosessia poikkeavasti. Lisäksi operaattorien ymmärrys prosessin sisäisestä toiminnasta ja vaikutusmekanismeista kasvaa.

Tutkimuslaitteiston automaatiopiirissä on näin ollen vähän toimilohkoja, koska varsinaisia säätöjäkin on vähän. Automaatiopiiri koostuukin pääasiallisesti laitteiston analogia- ja digitaalimittauksista sekä ohjauksista.

Tutkimuslaitteisto pyritään suunnittelemaan siten, että mahdollisimman monen parametrin muuttaminen on mahdollisimman yksinkertaista. Näin laitteistolla pystytään tekemään monipuolista ja vaihtelevaa tutkimus- ja kehitystyötä. Toisaalta prosessiolosuhteiden vakiona pitäminen on erittäin tärkeää, koska mitä vakioidummassa olosuhteissa prosessi toimii, sitä luotettavampia ovat tutkimustulokset.

### **3.2 Laatu ja toimitusvarmuus**

Tuotteen kannalta katsottuna tekninen laatu tarkoittaa tuotteen virheettömyyttä ja asiakaslaatu tarkoittaa tuotteen sopivuutta käyttötarkoitukseensa. Tuotanto-

toiminnassa laatu ja toimitusvarmuus ovat erittäin tärkeitä seikkoja. Laatu tarkoittaa sekä tuotteen että itse prosessin virheettömyyttä. (3, s. 141.)

Prosessin vaihtelua (virheitä) aiheuttaa kaksi erilaista syytyyppiä. Erityissyöt ovat sattumanvaraisia ulkoisia häiriötekijöitä ja yleiset syyt ovat prosessin sisäisestä vaihtelusta johtuvia prosessin normaalitilan ominaisuuksia. Yleisten syiden poistaminen prosessista vaatii prosessin suunnittelemista uudelleen ja sen sisäisten mekanismien parempaa ymmärtämistä. (3, s. 145–147.)

Tuotantoprosessista pyritään poistamaan kaikki mahdolliset virhelähteet, koska ne aiheuttavat kustannuksia ja häiriötilanteita suunniteltuun tuotantoon. Toiminnassa ilmenevät virheet heijastuvat usein ongelmoina toimitusvarmuudessa. (5, s. 334.)

Erätuotantotyyppisessä toiminnassa erityisesti asetusajkojen lyhentäminen on olennaista kustannusten ja toimitusvarmuuden kannalta. Tutkimuslaitteiston käyttö onkin luonteeltaan lähempänä erä- kuin massatuotantoa, vaikka jatkuva-toimista prosessia mallinnettaisiinkin. Kuitenkaan tutkimustoiminnan kustannusten ääretön hallinta ja häiriöiden poistaminen tuotannosta eivät ole samalla tavalla keskeisiä kuin tuotantoprosessissa. Sen sijaan siinä voidaan nimenomaan tutkia erilaisten häiriöiden ja virheiden vaikutusta tuotantoon ja toimitusvarmuuteen.

### **3.3 Toiminnanohjaus**

Toiminnanohjauksella tarkoitetaan yrityksen eri toimintojen ja tehtävien välistä jokapäiväistä ohjausta, toteutusta ja valvontaa. Sen kohteena ovat yrityksen perustoiminnot, kuten hankinta, varastointi, tuotanto, jakelu, myynti ja laskutus. Toiminnanohjauksen keskeisimmät tavoitteet ovat kapasiteetin korkea kuormitusaste, vaihto-omaisuuden minimointi, toimituskyky ja läpimenoaika. (3, s. 128, 374, 379.)

Toiminnanohjauksen tavoitteiden saavuttamista voidaan usein kehittää parhaiten tuotantojärjestelmän ominaispiirteitä kehittämällä, koska tuotantojärjestelmän ominaisuuksilla on merkittävä yhteys tuotannon tehokkuuteen, ohjaukseen liittyviin tehtäviin ja ongelmiin sekä käytössä oleviin ohjausperiaatteisiin ja



-menetelmiin. Tuotannon läpimenoaikojen lyhentäminen on yksi tehokkaimmista toiminnanohjauksen kehittämismenetelmistä. (3, s. 382.)

Tuotantojärjestelmän ohjattavuus kuvaa järjestelmän kykyä reagoida ohjausmuuttujiin. Ne voidaan jakaa kahteen kategoriaan: sisäisiin ja ulkoiisiin. Sisäisiä tekijöitä ovat mm. tuotantomuoto, läpimenoaika, valmistuserien suuruus, yksikön koko, tuotevariaatioiden määrä ja työvaiheiden määrä. Ulkoiisiin tekijöihin kuuluvat mm. markkinat, asiakkaat, toimittajat, menekin ennustettavuus, toimitusaika jousto ja asiakaskohtaiset tuotevaatimukset. (6, s. 21–22.)

Tuotekehitykseen ja tutkimukseen keskittyvässä toiminnassa tuotantojärjestelmän ominaisuuksia pyritään kehittämään keskittymällä sisäisten tekijöiden vaikutusten tutkimiseen. Ulkoiset tekijät sanelevat enemmänkin tuotekehitystoiminnan taloudellisen kannattavuuden, mutta onnistuessaan tuotekehitys voi muuttaa myös ulkoisia tekijöitä ja parantaa siten toiminnanohjauksen tehokkuutta. Systemaattisella tuotekehityksellä voidaankin saavuttaa useita liiketoiminnan kannalta merkittäviä etuja (3, s. 249).

### **3.4 Tuotanto**

Tuotannolle asetettuja yleisiä tavoitteita ovat mm. alhaiset valmistuskustannukset, virheettömyys, toimituskyky sekä joustavuus eli kyky vastata kysynnän vaihteluihin (3, s. 61).

Tuotannon kilpailulähtöiset tavoitteet ovat kustannustehokkuus, laatu, aika ja joustavuus. Tuotantoprosessin läpimenoaikaa pyritään lyhentämään, koska se tehostaa prosesseja ja parantaa toiminnan laatua ja pienentää kustannuksia. Joustavuus käsittää nopeutta ja kustannustehokkuutta, jolla tuotantoprosessia voidaan muuttaa, kuten volyymijoustavuus, tuotevariaatiot, uusien tuotteiden ja teknologien käyttöönotto. (5, s. 335–336.)

Prosessituotannossa toiminta on jatkuvaa kehittämistä ja toiminnan parantamista. Massatuotannossa optimaalisia ajo-olosuhteita pyritään pitämään yllä mahdollisimman pitkään, jolloin tuotannon käynnistykset ja lopetukset ovat poikkeustilanteita (5, s. 331).

Tutkimustuotannossa lähestytään erätuotannon piirteitä. Prosessiajo voidaan keskeyttää, kun optimitilanne on saatu tallennettua. Toiminta on lyhytaikaisempaa ja ylös- ja alasajoja tapahtuu paljon ja samaa tuotantokoneistoa käytetään useiden tuotteiden valmistamiseen.

Prosessituotannon huoltoseisokit ja laitteiden vaihtamiset minimoidaan ja suunnitellaan hyvissä ajoin etukäteen. Tutkimustoiminnassa tuotantolaitteita vaihdetaan linjalle usein, jolloin automaatio-ohjaukselle ja valvomo-ohjaukselle asetetaan suuret jousto- ja muuntautumisvaatimukset. Tämä vaikuttaa osaltaan myös siihen, kuinka paljon automaatio-ohjelmaan voidaan ylipäättään rakentaa säätöpiirejä ja muita automaattisia ohjauksia.

### **3.5 Varotoimenpiteet**

Tuotevastuu ja -turvallisuus ovat keskeisiä tuotekehityksessä huomioon otettavia seikkoja (3, s. 255). Operaattoreiden tilannetietoisuus prosessin tilasta voi olla häiriintynyt tai virheellinen tai heidän huomionsa kohde voi olla kaventunut. Tämä on tyypillistä ihmisen päätöksenteossa, mutta se voi teollisuusympäristössä aiheuttaa häiriöitä ja onnettomuuksia. Käyttöliittymäsuunnittelulla voidaan parantaa operaattoreiden tilannetietoisuutta. (2, s. 70–71.)

Vaikkei toimintaa poikkeustilanteissa voida kokonaan automatisoida, on suojauksilla ja lukituksilla sekä niiden laadukkaalla suunnittelulla pidettävä vaaratilanteet minimissä. Varotoimenpiteet ovatkin prosessiteollisuudessa äärimmäisen tärkeitä, mutta valitettavasti normaalissa toimintatilanteessa ne saattavat jäädä tekemättä, koska tuntuvat turhilta tai tehokkuuspaineet muuttavat sovittuja turvallisuusrutiineja ja toimintatapoja. Virheellisen toiminnan estämiseen voidaan vaikuttaa valvomoratkaisuilla, jolloin voidaan esimerkiksi sekvensseillä helpottaa varmistusten ja muiden turvallisuutta edistävien rutiinien toteutumista. (2, s. 50.)

Lukitukset ja hälytykset on usein minimoitu tutkimustoiminnassa, koska poikkeamia tapahtuu paljon, osittain myös tarkoituksellisesti, jolloin normaalit tuotannon rajat eivät sovellu sellaisinaan tämäntyyppiseen toimintaan. Työ- ja kone-

turvallisuus täytyy ilman muuta ottaa huomioon, kun prosessin ajoon liittyviä lukituksia vähennetään.

### **3.6 Dokumentointi**

Tietojärjestelmien merkitys yrityksen toiminnanohjauksessa ja tietojenhallinnassa on jatkuvasti kasvanut. Teollisuusyrityksessä käsiteltävät tietomäärät ovat suuret. Tavallisimmin tietojärjestelmiä käytetään materiaali- ja kapasiteetin hallintaan sekä kustannuslaskentaan. (5, s. 407, 410–411.)

Tuotetiedon määrän kasvu voi aiheuttaa ongelmia. Tietoon voi olla hankala päästä käsiksi. Lisäksi tiedosta voi olla useita keskenään ristiriitaisia versioita ja tiedon luotettavuus voi olla heikko. Tuotetiedon hallintajärjestelmien avulla näihin ongelmiin voidaan hakea ratkaisuja. Tuotetiedon hallinta kattaa dokumenttien hallinnan, työkulun ja muutosten hallinnan, tuoterakenteiden ja konfiguraatioiden hallinnan. (3, s. 260.)

Toiminnan keskeisimmät erityispiirteet ja haasteet aiheutuvat yksittäisten projektien ainutlaatuisuudesta. Lisäksi tyypillisesti tuotekehitysvaihe tuottaa paljon puoliteknisiä dokumentteja. (3, s. 245).

Tiedonkeruun merkitys, tiedon käyttö ja mittapisteiden määrä korostuvat tutkimuksessa, jossa tärkeää on nimenomaan ajoaikaisten parametrien tallettaminen ja niiden myöhempi mallintaminen. Tietoa tuotetaan valtavat määrät ja tiedon keruu sekä erityisesti käsittely asettavat omat vaatimuksensa.

## 4 TYÖN TOTEUTUS

### 4.1 Esiselvitykset

Työn aluksi perehdyttiin Schneider Electricin Vijeo Citect- ja Unity Pro -ohjelmistoihin. Vijeo Citect on Client/Server -tyyppinen valvomo-ohjelmisto ja Unity Pro on automaatio-ohjelmisto, jolla logiikka ohjelmoidaan.

Tutustuttiin kaivoksien ja rikastamon toimintaan sekä rikastusprosessiin ja yksittäisten laitteiden toimintaan. Lisäksi selvitettiin rikastamoissa käytettyjen termien englanninkieliset nimitykset, sillä käyttöliittymä toteutettiin englanninkielisenä. (7; 8.)

Endominesin Tuupovaaran kultakaivoksella tutustuttiin nykyaikaisiin prosessinäyttöihin ja kerättiin hyviä huomioita prosessin esittämisestä valvomossa.

Vierailulla Geologian Tutkimuskeskuksella (GTK) päästiin tutustumaan minimipilot-rikastuslaitteistoon ja sen toimintaan. Prosessilaitteisto oli muunneltava modulaarinen kokonaisuus. Täällä nähtiin käytännössä, kuinka tutkimusympäristössä asiat toteutetaan eri lailla kuin tuotannonohjauksessa. Prosessiohjaukset oli toteutettu paikallisesti ja valvomonäytöllä oli ainoastaan PH-näyttö ja hienosäätö. PH:n hienosäätö olikin ainoa säätöpiirillä toteutettu ohjaus. Prosessin hälytykset oli karsittu minimiin ja lukituksia ei käytetty.

### 4.2 Käyttöliittymän toteutus

#### 4.2.1 Lähtötiedot

Valvomosuunnittelun lähtötietoina oli käytettävissä PI-kaavio (liite 1) sekä varsin varhaisessa vaiheessa automaatiojärjestelmän yleiskaavio (liite 2). Lähtötietomäärittelynä kaikki ohjaukset toteutetaan manuaalisesti, eikä säätöpiirejä käytetä. Moottoreiden käynnistykset ja pysäytykset toteutetaan manuaalisesti, sekvenssejä ei käytetä. Moottoreita ajetaan vain eteenpäin, jolloin taakseohjausta ei tarvita. Järjestelmää ohjataan automaatiojärjestelmällä ja manuaaliohjaukset tapahtuvat valvomosta järjestelmän käyttöliittymän kautta.

Käyttöliittymän grafiikalle ei esitetty vaatimuksia. Erityisesti vaahdotuskennojen toiminnan esittämiseksi tutkittiin teollisuudessa käytössä olevia erilaisia vaahdotuskennostojen kuvakkeita. (7, s. 417; 7, liitteet 1–6; 9; 10.)

#### 4.2.2 Suunnittelu

Rikastusprosessi jaettiin kolmeen toiminnalliseen kokonaisuuteen: jauhatus, kuparipiiri ja sinkkipiiri.

Työssä käytettyjä näyttösivutyyppejä ovat prosessikaavionäyttö, tehtävänäyttö, trendinäyttö ja hälytyslista. Jatkossa, kunhan prosessin automatisointia lisätään, tarvittavia näyttötyyppejä tulevat olemaan ainakin piirinäyttö PH:n hienosäätöön sekä sekvenssinäyttö käynnistyksille.

Raporttinäyttöjä ei toteutettu valvomon käyttöliittymään, koska niistä huolehtii Ampla. Ampla on Schneider Electricin tuotannonohjausjärjestelmä. Amplaan viedään prosessiraportit, analyysilaboratoriossa tuotetut aineanalyysit sekä päiväkirjamaisia huomioita ja ohjeita.

Prosessikaavionäytöissä käytettiin useampaa tarkkuustasoa esitettävän tiedon määrän mukaan. Koko prosessia esittävässä prosessinäytössä on esitetty vain laitteet ja virtaukset hyvin yleisellä tasolla (liite 3). Jauhatuspiirissä on toteutettu suoraan taso kaksi, jossa esitetään virtausmittareiden näyttämä ja moottoreiden pyörintänopeudet (liite 4). Tämä valittiin toteutustavaksi, koska jauhatuspiirissä on hyvin vähän laitteita ja yksityiskohtaisemmatkin tiedot mahtuvat esitettäväksi yhdellä tasolla.

Kupari- ja sinkkipiireissä toteutettiin tasot yksi ja kaksi, yleisemmän tiedon esittäminen (liitteet 5–6) ja yksityiskohtaisemman tiedon esittäminen (liitteet 7–8). Tasolla yksi esitetään laitteet ja yleiset materiaalivirtaukset, taso kaksi sisältää mittareiden näyttämät, moottoreiden pyörintänopeudet sekä kemikaalipumppujen virtaukset.

Jokaiselta tasolta voidaan haluttaessa ohjata moottoreiden toimintoja, mutta tietoja ei näytetä suoraan käyttöliittymässä tasolla yksi, jotta valvonnan pääasiallinen huomio voidaan kiinnittää laitteiden toimintaan. Näin pyrittiin varmistamaan

maan, että operaattori voi tarpeen mukaan valita sopivan näyttöhierarkiatason, joko yleisvalvonta tai yksityiskohtaisempi valvonta.

### **4.2.3 Grafiikan suunnittelu**

Prosessikaavion ja PI-kaavion esittäminen valvomonäytöillä vaati erittäin paljon yksinkertaistamista ja monta piirroskierrosta. Valvomokuva ei vastaa todellista fyysistä kokoonpanoa, vaan prosessi on pyritty esittämään näytöillä mahdollisimman ymmärrettävästi.

Värien määrittelyssä käytiin useita testauskierroksia varsinkin letkulinjojen ja vaahdotuskennostojen esittämisessä. Kemikaalilinjat ja prosessivesilinjat esitetään teollisuudessa vakiintuneen käytännön mukaan omilla väreillään.

Letkulinjojen virtaussuuntien osoittaminen piti suunnitella huolella, koska ohjelmassa ei ole valmiina mahdollisuutta osoittaa virtaussuntaa putkilinjan päässä olevalla virtaussuntaa osoittavalla nuolella. Niinpä pumput päätettiin sijoittaa letkulinjan alkupäähän ja vaahdotuskennostojen tulo- ja lähtölinjat sijoitettiin näytöillä vakioituihin kohtiin, vaikka todellisuudessa kennojen tulo- ja lähtölinjojen paikat vaihtelevat jonkin verran.

Toisaalta vaahdotuskennostojen sijoittelulla pyrittiin osoittamaan fyysisesti toisiinsa liittyvien kennostojen yhteys. Kahden erityyppisen vaahdotuskennoston rakenne on esitetty samalla piirrosteknisellä tavalla.

Vijeo Citectissa olevia valmiita kuvakkeita, kuten pumppuja, säiliöitä ja mootto-reita, käytettiin mahdollisimman paljon.

### **4.2.4 Ohjelmointi**

Logiikan ja käyttöliittymän rajapinnassa käytettiin OPC Factory Serveriä (OFS). Se toimii kommunikaatorajapintana valvomo-ohjelmiston ja ohjelmoitavan logiikan välillä. Ratkaisu mahdollistaa käyttöliittymän ohjelmoinnissa logiikan muuttujien loogisten nimien käytön muistipaikkojen sijaan.

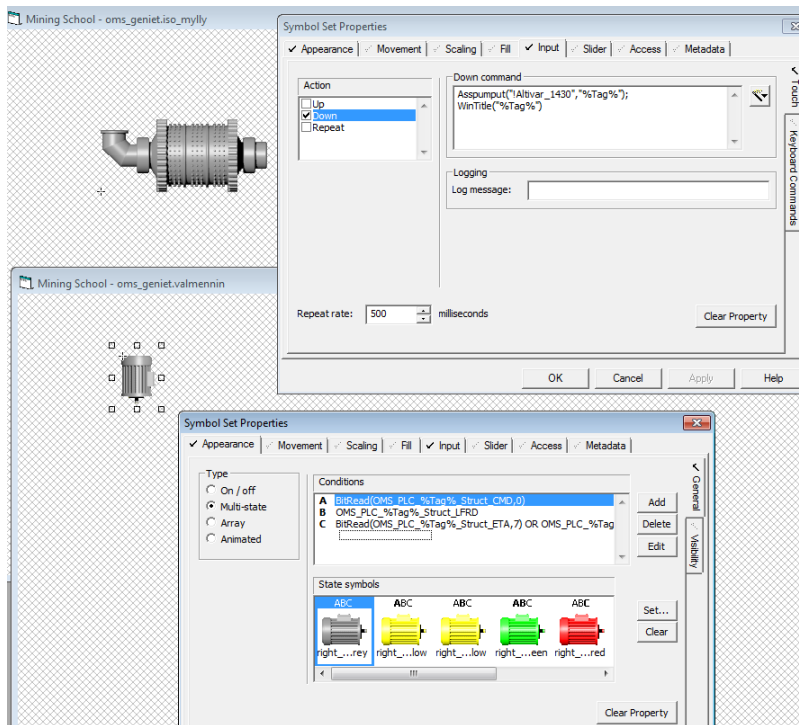
Logiikkaohjelmoinnissa käytettyjen struktuurien muuttujanimien selvittely vaati tutustumista logiikan toimintaan toimilohkotasolla. Struktuurien muuttujia käyte-

tään struktuurin sisäisen muuttujanimen avulla, ja juuri tätä muuttujaa käyttöliittymässä luetaan ja kirjoitetaan.

Vijeo Citectin ohjelmoinnissa käytetään Cicode-ohjelmointikieltä. Käyttöliittymän ohjelmoinnissa käytetään Cicode-koodia esimerkiksi näyttöjen ja niissä esitettävien tietojen määrittelyyn. Sitä käytetään luettaessa ja kirjoitettaessa logiikan WORD-muotoa olevaa muuttujaa. Tässä projektissa ATV32-taajuusmuuttajien komentosana (CMD) ja tilasana (ETA) ovat WORD-muotoa. Cicode-koodia käyttäen käyttöliittymästä kirjoitetaan ja luetaan WORD-sanan bittejä.

Moottoreiden ohjausikkunat toteutettiin ponnahdusikkunoina, jotka tehdään Vijeo Citect -ohjelmassa ns. supergenienä. Unity Prolla ohjelmoidun logiikan muuttujanimet ovat projektissa niin pitkät, että suurin osa näistä supergenie-toteutuksista vaatii käyttöliittymää ohjelmoitaessa cicode-koodin käyttöä. Cicode-koodin avulla ponnahdusikkunalle välitetään sen tarvitsemat tiedot eli moottorin ohjausikkunan tiedot.

Genie on uudelleen käytettävä symboliryhmä, ja usein käytettyjä toimilaitteita voidaan helposti kopioida käyttämällä näitä symboliryhmiä uudelleen (kuva 1).



KUVA 1. Genien luonti

Geniessä laitteen ominaisuudet, kuten käytetty kuvake, tilojen värit jne., määritellään tallennettavaan symboliin. Samalla määritellään toimilaitteen ohjaamiseen käytetty ponnahtusikkuna. Symbolia käytettäessä sille määritellään todellisen laitteen muuttujanimi.

Suurimmalle osalle laitteista onkin luotu genie, ja näin ollen uusien samanlaisten laitteiden lisääminen valvomoon on helppoa ja yksinkertaista. Laiteluettelossa on luetteloitu käytetyt geniet, supergeniet ja niihin liittyvät Cicode-koodit (liite 9).

#### 4.2.5 Erikoisnäytöt

Hälytyssivujen luontiin käytettiin Vijeo Citect -ohjelmassa valmiina olevia hälytysikkunoiden pohjia. Hälytyksessä käytettävät muuttujat, hälytyksen nimi ja selitys piti määrittellä hälytysmuuttujalistaan. Muuten käytetty pohja huolehti hälytyssivujen toteuttamisesta.

Hälytyksissä on vain yksi taso, koska hälytykset koostuvat ainoastaan laitteiston vikaantumisen hälytyksistä, oikosulkusuojista ja moottorisuojista. Säätepii-



reihin kuuluvia poikkeamahälytyksiä laitoksessa ei luonnollisestikaan ole. PH:n seurantaan olisi voinut suunnitella poikkeamahälytyksen, mutta sitä ei vaatimuslistalla ollut ja se on näin ollen toteuttamatta.

Trendinäyttöä (liite 10) käytetään valmentimien PH:n ja massavirtauksen vaihtelujen havainnollistamiseen ja seurantaan. Vijeo Citect mahdollistaa kaksi tapaa tehdä trendinäyttö: trendipiirros ja Process Analyst -työkalu. Työssä käytettiin Process Analyst -työkalua, koska se tarjoaa laajemmat mahdollisuudet trendien käsittelyyn. Projektissa trendiä piirretään valituista muuttujista koko ajan, ei vain näytön ollessa esillä. Sen lisäksi näyttö on määritelty avautumaan automaattisesti, kun sivu valitaan. Nämä ominaisuudet määriteltiin cicode-koodin avulla Process Analyst -näyttöihin.

Prosessin valvonnan helpottamiseksi linjastossa on neljä Pelco-kameraa, jotka kuvaavat laitteistoa eri paikoista. Yksi kameroista kuvaa linjastoa kokonaisuudessaan, yksi siiloa ylhäältäpäin ja kaksi vaahdotuskennostoja. Pelco-kameroita ohjataan Vijeo Citectissa olevalla lisäohjelmalla.

Valvomon käyttöliittymässä ei ole raporttisivuja, koska raportointi tehdään Amp-lan kautta. Ampla on tuotannonohjaukseen tarkoitettu ohjelma, jolla voidaan tuottaa raportteja ja tunnuslukuja prosessista. Prosessista mitattavat tiedot siirretään Amplaan historiatietokannan kautta. Historiatietokannan parametointi suoritettiin työn aikana.

Turvallisuustoimintoina prosessialissa on kolme hätäseis-painiketta ja lisäksi valvomon ruudulla on hätäseis-mahdollisuus. Painiketta painamalla kaikki laitteet pysähtyvät ja vaativat valvomosta tehtävän kuittauksen, ennen kuin laitteiden uusi käynnistyminen on mahdollista. Hätäseis-toiminto on toteutettu laitoksessa logiikan tasolla.

### **4.3 Testaukset ja käyttöönotto**

Käyttöliittymän toiminnan testaamiseen käytettiin aluksi simulointia. Vijeo Citect käyttöliittymä ja Unity Pro -logiikkaohjelma liitettiin toisiinsa OFS-serverin kautta. Näin ollen pystyttiin testaamaan käytettyjen luku- ja kirjoitusstruktuurimuuttujien nimien oikeellisuudet. Moottorien ja pumppujen tilatietojen muutokset voitiin tes-

tata simuloimalla niitä logiikassa. Tässä vaiheessa saatiin selvitettyä lopullisesti, millä muuttujanimellä logiikan struktuureja luetaan ja kirjoitetaan.

Trendinäyttöjen piirrostoimintoa ei voitu testata tässä vaiheessa. Myöskään Pelco-kameroita ei voitu testata simuloimalla.

Taajuusmuuttajat konfigurointiin ja koekäytettiin kaapeloinnin ja pyörimissuunnan varmistamiseksi, kun sähköistykset niihin oli tehty. ATV32-taajuusmuuttajiin ja niiden parametrintiohjelmaan saatiin koulutus, jolloin niiden parametointi oli hyvin helppoa.

Varsinaiseen testausvaiheeseen päästiin, kun kaikki asennukset oli suoritettu ja automaatiojärjestelmä toimintakunnossa. Tässä vaiheessa logiikan toiminta testattiin ja tehtiin tarvittavat muutokset toimintoihin. Käyttöliittymäohjaukset toimivat simuloinnin jälkeen varsin suoraviivaisesti, mutta paljon muutoksia vaadittiin valvomon toimintaan liittyvään logiikkaan. Tässä vaiheessa siis huomattiin, ettei jokin suunniteltu tiedon esittämistapa ollutkaan hyvä eikä aukoton, vaan jouduttiin etsimään paremmin toimivia vaihtoehtoja. Moottoreiden tilatietoa kuvaavien värityksien määrittäminen oli haasteellista, koska mitään vaatimusmäärittelyä ei tiedoille ollut.

Käyttöliittymään liittyvät letkulinjojen kuvaukset, jotka oli tehty PI-kaavioiden perusteella, osoittautuivat käytännössä mahdottomiksi toteuttaa. Vaahdotuskoneiden letkuttaminen PI-kaavion osoittamalla tavalla oli käytännössä mahdotonta ja jouduttiin toteuttamaan kokonaan eri tavalla. Letkupumppujen ominaisuus, kolme toimielintä yhdessä pumpussa, asettaa putkisuunnittelulle melkoiset haasteet, kun otetaan huomioon erilaiset pyörimisnopeustarpeet prosessin eri vaiheissa.

Pumppujen ja virtauksien osoittamiseksi käyttöliittymän näytöt piti suunnitella tässä vaiheessa uusiksi. Prosessissa oli todellisuudessa enemmän pumppuja kuin PI-kaaviossa oli esitetty ja käytännössä pumppuja joudutaan vielä lisäämään prosessiin. Niinpä suunniteltua pumppujen esitystapaa jouduttiin muuttamaan. Näytön selkeyden säilyttämiseksi virtaukset esitetään edelleenkin yhdellä letkulinjalla, vaikka todellisuudessa niitä on huomattavasti enemmän. Prosessin

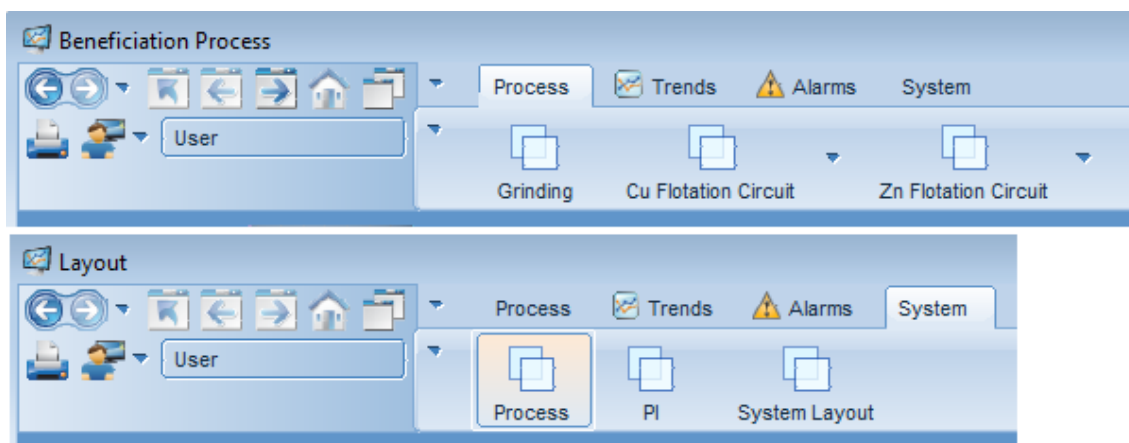
pumput kuvattiin sen sijaan todellisissa määrissä, mutta säilyttäen esityslogiikka. Pumppujen paikat määräytyivät suunnittelukäytännön perusteella, jolla pyrittiin yksiselitteisesti ja selkeästi kuvaamaan letkulinjan virtaussuunta sekä yksinkertaistamaan vaahdotuskennostojen kuvakkeita. Pumppujen yksilöintitiedot muuttuivat tässä vaiheessa alkuperäistä suunnitelmaa yksinkertaisemmiksi.

## 5 TOTEUTETTU KÄYTTÖLIITTYMÄ

Muutamiin käyttöliittymän yksityiskohtiin ja toteutustapoihin paneudutaan seuraavassa tarkemmin. Käyttöliittymän ja historiatietokannan ohjeissa on selostettu tarkemmin yksityiskohtien toteutusta ja ohjelmointi (liitteet 11–12).

### 5.1 Valikkorakenne

Käyttöliittymään kuuluvat perustoiminnot rakentuvat Vijeo Citect -ohjelmassa automaattisesti. Tällaisia toimintoja ovat mm. valikkorakenne ja sisäänkirjautumisikkuna. Kuvassa 2 on valmiiksi muokattu valikkorakenne. Ylimmällä tasolla ovat valinnat Process, Trends, Alarms ja System.

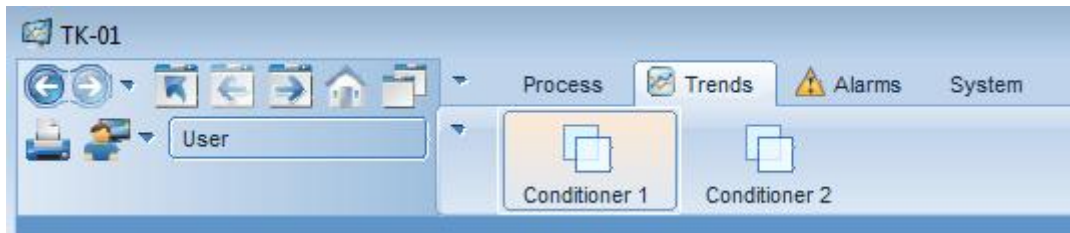


KUVA 2. Otsikkorakenne

Process-välilehdellä näyttösivun kuvana on koko prosessin yleiskuva (liite 3). Lisäksi aukeaa toisen tason valikko: jauhatus, kuparipiiri ja sinkkiapiiri. Kupari- ja sinkkipiirin tarkemman alatason näyttösivut saadaan edelleen valitsemalla valikkorakenteessa kolmantena olevat tasot.

System-välilehdellä on toisella tasolla valittavana Process, PI ja System Layout. Process-näyttösivu esittää rikastamon kolmiulotteisen mallikuvan, PI-näyttösivulla on PI-kaavio (liite 1) ja System Layout esittää järjestelmäkaavion (liite 2).

Kullakin valikkorakenteen tasolla voidaan liikkua nuolikuvakkeilla oikealle ja vasemmalle. Alemmalta tasolta pääsee ylemmälle tasolle nuolikuvakkeen avulla. Tämä toiminto on toteutettu kupari- ja sinkkipiirin tasolta kolme tasolle kaksi siirryttäessä. Trendinäytöllä ylemmällä tasolla on trendikuvaan liittyvä prosessikuva (kuva 3).



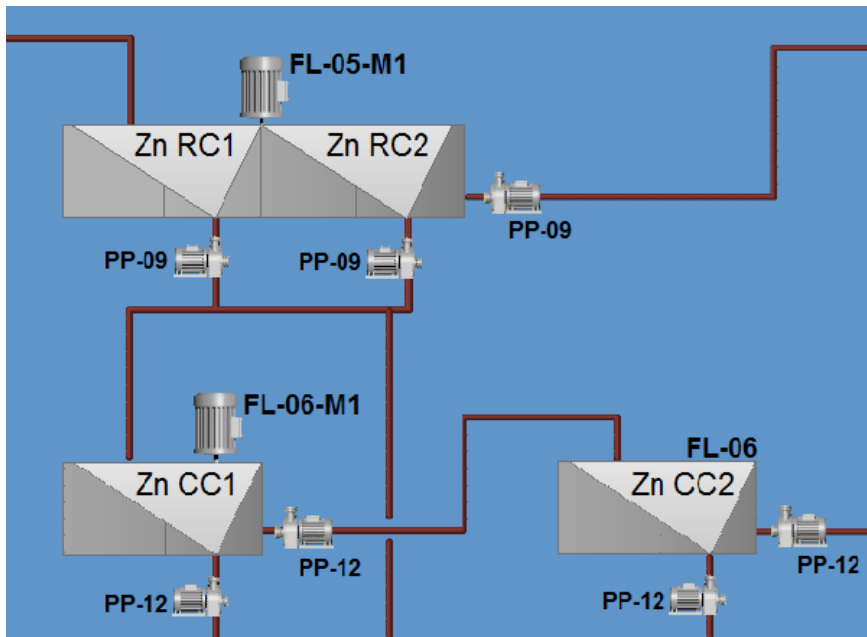
*KUVA 3. Menurakenteessa ylöspäin siirtyminen*

Näin ollen valmentimen yksi trendinäytöstä siirrytään kuparipiiriin ja valmentimen kaksi trendinäytöstä vastaavasti sinkkipiiriin.

## **5.2 Vaahdotuskennostot ja letkulinjat**

Erilaisten vaahdotuskennostojen rakenteet esitettiin yhtenäisellä esitystavalla, vaikka fyysisesti kennostot ovat erilaisia. Esityksessä korostettiin tulo- ja lähtölinjojen sijaintia selkeyden lisäämiseksi. Ne sijaitsevat aina kennostossa samassa paikassa ja ovat näytöllä samassa tasossa toisiinsa nähden (kuva 4).

Kuvassa 4 olevat vaahdotuskennostot FL-05 ja FL-06 ovat toiminnaltaan ja rakenteeltaan erilaiset. Kennosto FL-05 on läpivirtauskennosto, jossa on vain yksi alitelähtö kun taas kennostosta FL-06 alitelähtöjä on kaksi. Kennostojen syöttöpisteet, joita FL-05:ssä on yksi ja FL-06:ssa on kaksi kappaletta eri paikoissa, on esitetty selkeyden vuoksi samassa kohdassa kennoa. Lisäksi vaaleammalla esitetty ylite ja tummemmalla oleva alite poistuvat kennoista aina samalta kohdalta.



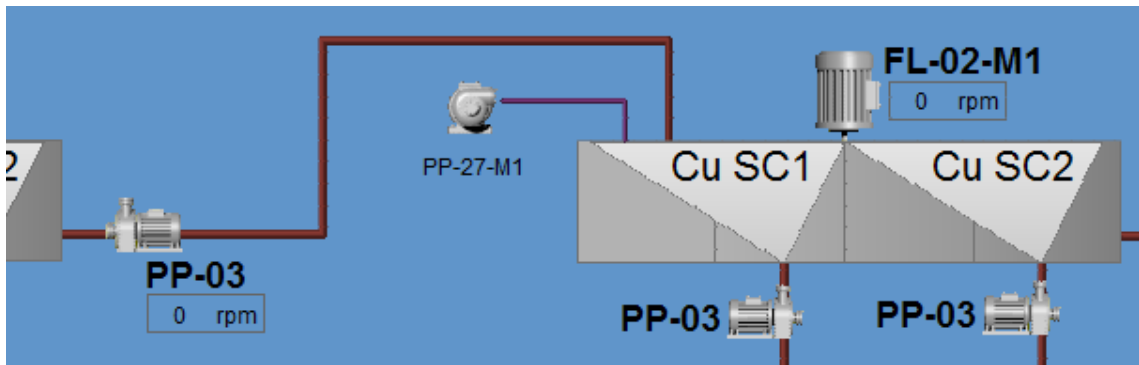
*KUVA 4. Yksityiskohtavaahdotus kennostoista*

Kennosto FL-06 on jaettu näytöllä kahteen osaan, joiden yhteenkuuluvuutta korostetaan läheisyydellä ja nimeämällä. Lisäksi esitystapa korostaa letkulinjojen virtaussuuntia. Poistuvan virtauksen letkulinjan alussa on pumppu.

Kuvassa 4 nähdään, kuinka letkulinjat on osoitettu yhdellä viivalla, vaikka todellisuudessa kennoston FL-06 ensimmäisen kennon syötteeseen tulee kolme letkulinjaa.

### 5.3 Moottorit

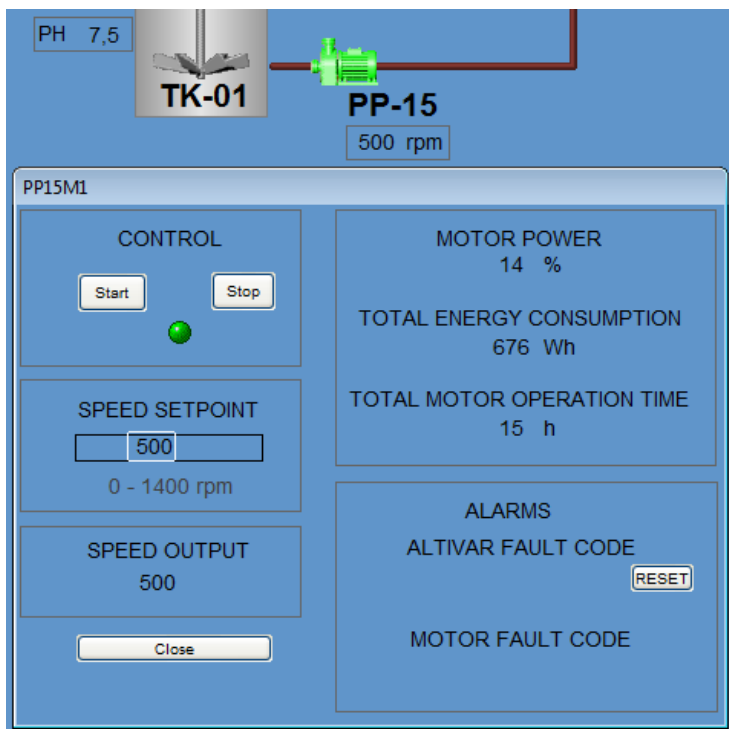
Kuvassa 4 näkyvä pumppu PP-09 on saman pumpun eri toimilohkoja ja niiden esitysmuoto yleistason kuvassa. Tarkemman tason kuvassa letkupumpun ominaisuutta on korostettu kierrosluvun esitystavalla, kuten kuvassa 5.



KUVA 5. Pumppujen esitystapa

Pumpun kierrosluku esitetään vain kerran, vaikka pumppu esiintyykin linjastossa kolmessa kohdassa. Jokaisesta pumppukuvakkeesta voi avata pumppunohjausikkunan, mutta ikkuna avautuu vain kerran yhtä pumppua kohti.

Pumppua kaksoisklikattaessa avautuu ponnahdusikkuna, joka toimii pumpun ohjausikkunana (kuva 6).

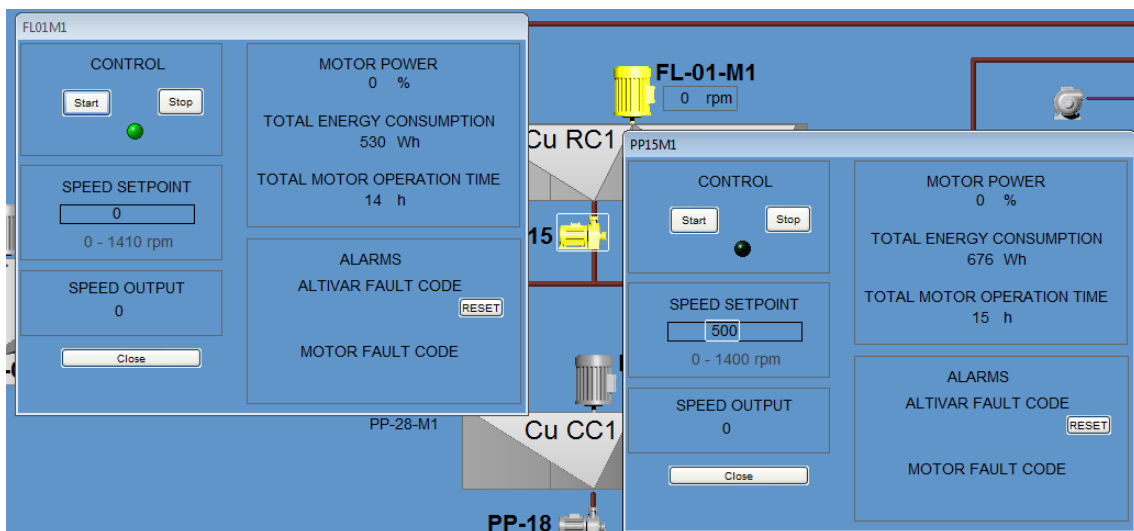


KUVA 6. Moottorin ohjausikkuna

Moottoreiden (pumppujen, sekoittimien, myllyjen ja luokittimen) ohjausikkunat ovat samanlaiset. Ohjausikkunasta moottoreille annetaan Start- tai Stop-käskey sekä pyörintänopeus. Ohjausikkuna osoittaa asetetun pyörintänopeuden, todellisen nopeuden, moottorin energiankulutustietoja ja mahdolliset vikatiedot. Vikatiedot voidaan kuitata RESET-painikkeella. Moottorin vikakoodi osoittaa, jos vikavirtasuoja tai moottorisuojakytkin on avoinna.

Ohjausikkuna voidaan siirtää näytöllä haluttuun paikkaan ja se suljetaan Close-näppäimellä.

Moottoreiden tilatiedot osoitetaan näytöllä värikoodeilla. Käytössä on neljä mahdollista tilaa (kuva 7).



*KUVA 7. Moottorin tilatieto*

Keltaisella kuvakkeella varustetulle moottorille on asetettu logiikalle joko käyntikäskey tai pyörintänopeus. Pyöriäkseen moottori tarvitsee kummankin tiedon, jolloin väri vaihtuu vihreäksi. Keltaisen tarkoitus onkin osoittaa, ettei moottori käy tilanteessa, jossa operaattori on epähuomiossa antanut vain toisen tiedon.

Moottorikuvake on punainen vikakoodin ollessa päällä ja harmaa, kun moottori ei ole toiminnassa eikä mitään asetusarvoa ole annettu.



Pienen kuulamylyn ohjausikkuna poikkeaa muista moottoreista. Myllyn pyörintä pysähtyy, jos suojaovi avataan. Avattu ovi muuttaa myllyn symbolin punaiseksi ja sama tieto ilmoitetaan ohjausikkunassa. Toiminnan esto on toteutettu logiikkatasolla. Mylly jatkaa toimintaa oven sulkeutuessa.

#### 5.4 Hätäseis-toiminto

Valvomossa jokaisella näytöllä esitetään hätäseis-toiminnon tilatieto (kuva 8) oikeassa yläkulmassa.



*KUVA 8. Hätäseis*

Kuvassa 8 ensimmäinen vaihe esittää prosessin normaalia toimintatilaa, jolloin näkyvissä on valvomon hätäseis-painike ja kuitauspainike. Toisessa vaiheessa jotain prosessialissa olevista kolmesta painikkeesta on painettu ja hälytystieto on aktivoitunut. Painettu hätäseis-painike on kuitattava, ennen kuin valvomon kuitauspainikkeella voidaan asettaa logiikka normaalitilaan. Kolmannessa vaiheessa valvomon näytöllä oleva hätäseis-painike on aktivoitu. Se tulee kuitata, ennen kuin logiikan normaalitilaan palauttaminen on mahdollista.

## 6 YHTEENVETO

Työssä toteutettiin graafinen käyttöliittymä Oulun yliopiston minipilot-rikastamon automaatiojärjestelmään. Käyttöliittymästä pyrittiin tekemään mahdollisimman havainnollinen ja helppolukuinen. Valvomon käytettävyyks onkin hyvä ja muunneltavuus on otettu huomioon toteuttamalla käyttöliittymän laiteyksiköt uudelleen käytettävillä symboliryhmillä. Niiden avulla on mahdollista nopeasti ja helposti lisätä valvomokuvaan laitteita ja linkittää ne oikeisiin logiikan muuttujiin.

Työn lähtökohdissa tutustuttiin yliopistolla olemassa olevien automaatiojärjestelmien käyttöliittymiin, vaikka kyseessä onkin uuden laitoksen toteuttaminen. Tarkoituksena oli nivoa toteutettava minipilot-rikastamo olemassa olevaan muotokieleeseen. Toisaalta erittäin tärkeä ilmaisumuodon määrittäjä oli rikastusteollisuus ja siellä vallitsevat käytössä olevat prosessin toiminnan ja tilan ilmaisutavat. Tarkoituksena oli antaa opiskelijoille suora miellelyhtymä teollisuuskäytössä oleviin rikastamon käyttöliittymiin.

Laitokselle ei ollut olemassa käyttöliittymäkonseptia, jossa olisi määritelty käyttäjien ja automaatiojärjestelmän välinen vuorovaikutusperiaate, toiminnallinen perusrakenne ja suunnittelussa sovellettavat tyyliohjeet. Näistä lähtökohdista muodostettiin eräänlainen suunnittelukonsepti, jossa otettiin huomioon teollisuudessa käytössä olevat vakiintuneet tavat.

Käyttöliittymän toiminnan ja tehtävien suunnittelua vaikeutti automaatiojärjestelmän suunnitteluun oleellisesti liittyvän prosessin toimintakuvausten puuttuminen. Tästä johtuen logiikan ja käyttöliittymän toimintaperiaatteet perustuvat suunnittelijoiden näkemyksiin. Laitteiden ohjaukset ja esitystavat on suunniteltu mahdollisimman sopiviksi käyttöönsä. Tämän vuoksi suuriakin muutoksia toteutettiin vielä testausvaiheen jälkeen, kun laitosta ajettiin pitempiaikaisesti. Kuvausten puute siis hankaloitti ja hidasti projektin etenemistä.

Rikastamon tutkimuksellisen luonteen vuoksi prosessin ohjauksen primääritehtävät ovat tavanomaiseen teollisuusprosessiin verrattuna vähemmän vakiintuneita. Lisäksi prosessin ominaisuuksien vuoksi puuttuvat säätöpiirit asettavat operaattorille tavanomaista vaihtelevammat työtehtävät ja prosessin ohjaami-

nen perustuu pitkälti kokemukseen eikä niinkään valmiiden ohjeiden noudattamiseen.

Sekundääritehtävien hallinta ja operointi aiheuttavat varsin suuren osan operaattorin työtehtävistä. Varsinkin uusien laitteiden lisäykset prosessiin ja olemassa olevien muutokset teettävät töitä sekä logiikan, käyttöliittymän että raportoinnin tasolla. Tätä työtä helpottamaan tehtiin varsin laajat ohjeet käyttöliittymän ylläpitämisestä ja muuttamisesta (liite 11) sekä historiatietokannan ja Amp-lan päivittämisestä (liite 12). Myös käyttöliittymässä käytetyt symboliryhmät helpottavat suuresti prosessilaitteiden lisäämistä ja muuttamista valvomoon.

Käyttöliittymän ohjeessa on kuvaus järjestelmätoteutuksesta sekä Vijeo Citect-ohjelman käytöstä laitoksen operaattoreille. Ohjeessa kuvataan kuvakkeiden käyttäminen ja selostetaan cicode-koodien merkitys sekä käyttötarkoitukset. Ohjeiden avulla operaattorin on tarkoitus pienen perehtymisen jälkeen pystyä hallinnoimaan sekä käyttöliittymää että siihen liittyviä historia- ja raportointikantoja.

Käyttöönottoa ja testausta pyrittiin varmistamaan ja helpottamaan vierailemalla automaatiotoimittajan tiloissa tekemässä FAT-testejä. Tarkoituksena oli tutustua automaatiojärjestelmään ja testata automaatiokaapin johdotukset. Harmiksemme analogiatuloja ja -lähtöjä ei voitu koestaa kunnolla. Lopullisessa käyttöönotossa ilmenikin sitten, etteivät analogiatulot ja -lähdöt toimineet.

Käyttöönotossa löytyi varsin paljon muutostarpeita, joiden selvittely vei osittain kauankin aikaa. Lisäksi käyttöönoton aikana ilmeni, ettei PI-kaavioon piirrettyä pumppujakoa voitaisi käyttää, vaan letkutus ja pumput jouduttiin suunnittelemaan uudelleen. PI-kaavion merkitys kutistuikin näin ollen lähinnä virtauskaaviksi. Käyttöliittymän pumppujen esitykset suunniteltiin tässä vaiheessa uudelleen. Sinkkipiirin letkupumppupositiot eivät ole vielä tässä vaiheessa lopulliset. Letkupumppuja joudutaan vielä lisäämään prosessiin ja tuomaan niiden ohjaukset käyttöliittymään.

Käyttöliittymän tulevat kehitystarpeet liittyvät logiikan kehittämiseen. Käyttöönotettava mahdollinen sekvenssi vaatii valvomolle omat erityissivunsa. Lisäksi

Unity Pro mahdollistaa sekvenssin laitesisällön vapaan määrittelyn manuaalisesti valvomosta, jolloin käyttöliittymässä täytyy olla valmius tietojen syöttämiseen sekvenssille. PH:n hienosäätöön mahdollisesti tuleva säätöpiiri täytyy konfiguroida myös käyttöliittymään.

Letkutuksien esittämiseen näyttösiivuilla joudutaan hyvin todennäköisesti miettimään vaihtoehtoisia tapoja. Näyttösiivuja voitaisiin toteuttaa erilaisilla letkutuksilla ajoreseptin mukaan. Lisäksi voisi suunnitella näyttösiivun, jossa ei ole ollenkaan letkutuksia. Kennostojen käyttötarkoitus ja järjestys ilmaistaisiin niihin manuaalisesti lisättävällä kennostotunnuksella

Työn aikana tutustuttiin kaivosteollisuuteen ja perehdyttiin rikastamon toimintaan. Työssä saatiin toteutettua hyvin toimiva käyttöliittymä Oulu Mining Schoolin minipilot-rikastamolle.

## LÄHTEET

1. Oulu Mining School. 2012. Saatavissa:  
<http://www.oulumining.fi/fi/koulutus/mika-on-oulu-mining-school.html>. Haku-päivä 10.8.2012.
2. Heimbürger, Harri – Markkanen, Pentti – Norros, Leena – Paunonen, Hannu – Savioja, Paula – Sundquist, Matti – Tommila, Teemu 2010. Valvomo, Suunnittelun periaatteet ja käytännöt. Helsinki: Suomen Automaatioseura ry.
3. Lehtonen, Juha-Matti 2004. Tuotantotalous. Porvoo: Werner Söderström Osakeyhtiö.
4. Oksanen, Kari 2011. Valvomosuunnittelun periaatteet ja toteutustavat. Oulu: Oulun yliopisto, prosessi- ja ympäristötekniikan osasto. Diplomityö.
5. Uusi-Rauva, Erkki – Haverila, Matti - Kouri, Ilkka 1999. Teollisuustalous. 3. painos. Tampere: Infacs johtamistekniikka Oy.
6. Piironen, Markku 2009. Tuotannon tehostaminen ABB:n System AC - tulosyksikön Durra-linjalla. Helsinki: Metropolia Ammattikorkeakoulu, tuotantotalous. Insinööriyö.
7. Hukki, R.T 1964. Mineraalien hienonnus ja rikastus. Keuruu: Teknillisten tieteiden akatemia.
8. Mankinen, Antti 2012. Pyhäsalmen kaivoksen rikastusprosessin vaahdotus-kemikaalit ja rikkipiirin kartoitus. Oulu: Oulun yliopisto, prosessi- ja ympäristötekniikan osasto. Kandidaatintyö.
9. Controle de Revisão de Telas sinóticas 2009. Julkaisematon dokumentti.
10. Pyhäsalmen kaivoksen valvomokuvat 2011. Julkaisematon dokumentti.

## **LIITTEET**

Liite 1 PI-kaavio

Liite 2 Järjestelmäkaavio

Liite 3 Rikastusprosessi

Liite 4 Jauhatus

Liite 5 Kuparipiiri Taso 1

Liite 6 Sinkkipiiri Taso 1

Liite 7 Kuparipiiri Taso 2

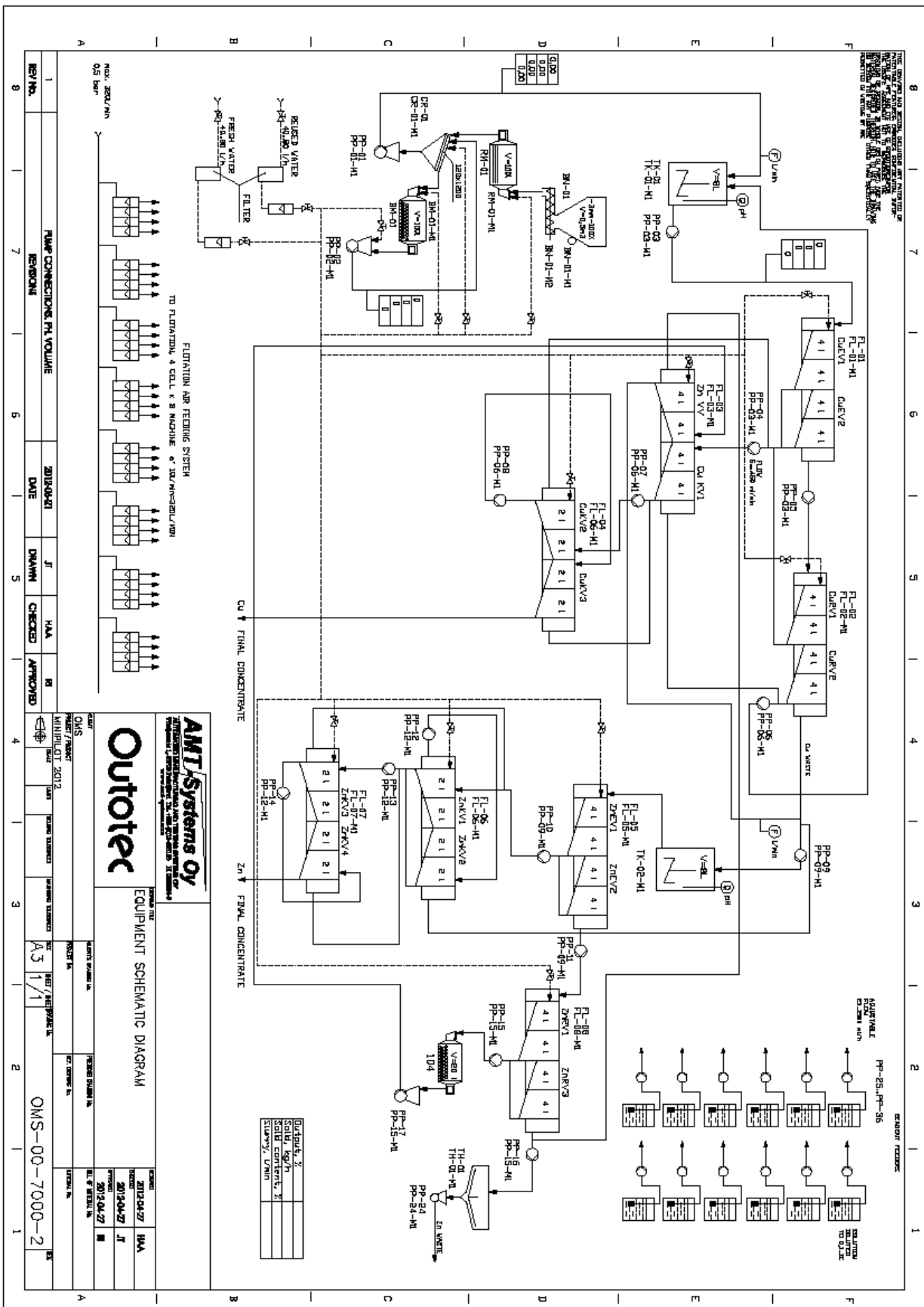
Liite 8 Sinkkipiiri Taso 2

Liite 9 Laitepositioluettelo

Liite 10 Trendinäyttö

Liite 11 Käyttöliittymäohje

Liite 12 Historiatietokantaohje



REV. NO.	1	REVISION	2012/04/27	DATE	JT	DRAWN	MAA	CHECKED	APPROVED
NAME CONNECTIONS IN VOLUME									
TO FLUOTATION & COLL. X & S MACHINE & 200 L/h/200 L/h/200 L/h									
FLOTATION AIR FEEDING SYSTEM									
TO FLUOTATION & COLL. X & S MACHINE & 200 L/h/200 L/h/200 L/h									

**AMT Systems Oy**  
 OYJÄRJESTYKSEN KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS

**Outotec**

EQUIPMENT SCHEMATIC DIAGRAM

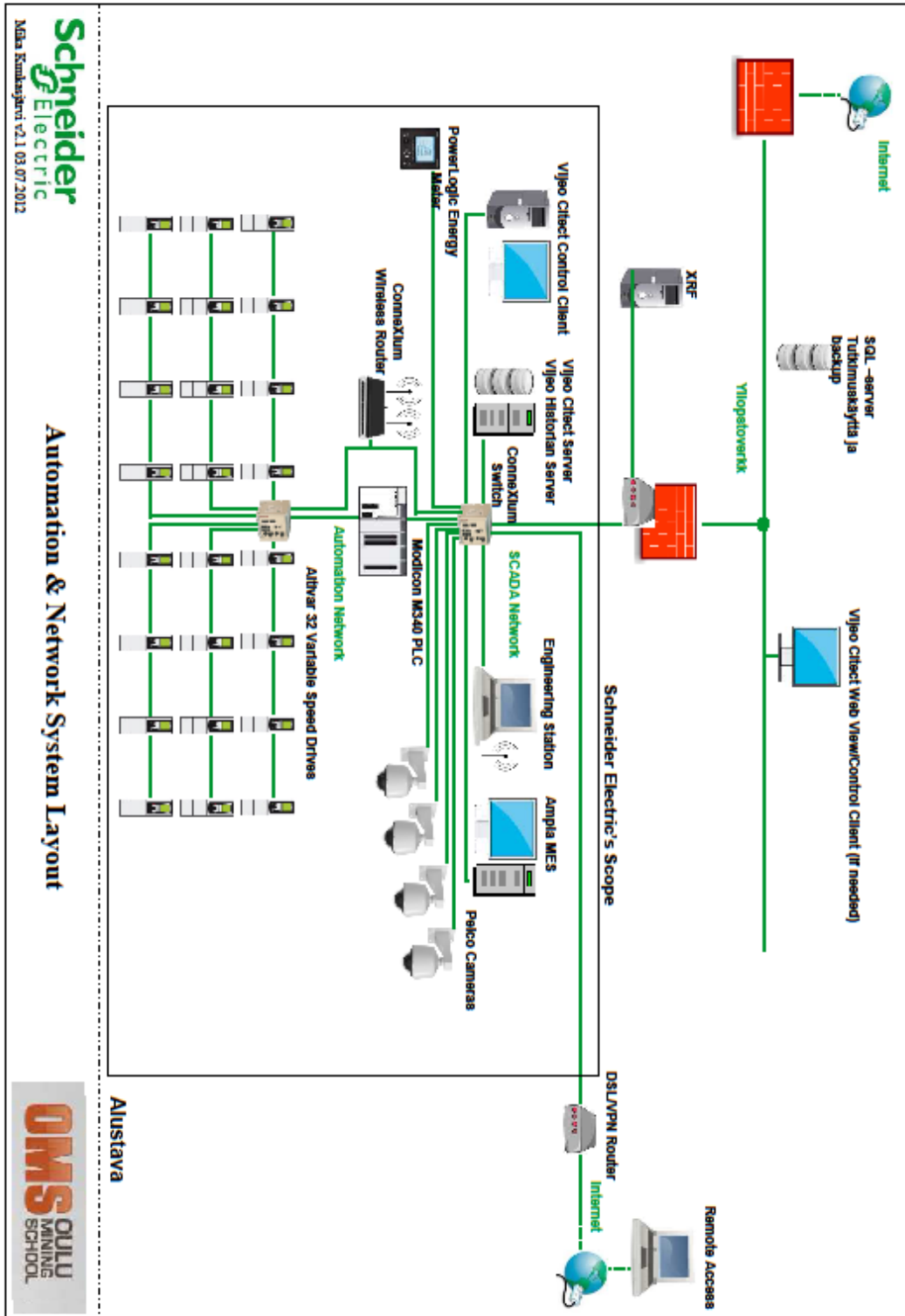
AMT Systems Oy  
 OYJÄRJESTYKSEN KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS

AMT Systems Oy  
 OYJÄRJESTYKSEN KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS  
 KÄYTTÖOIKEUS KÄYTTÖOIKEUS

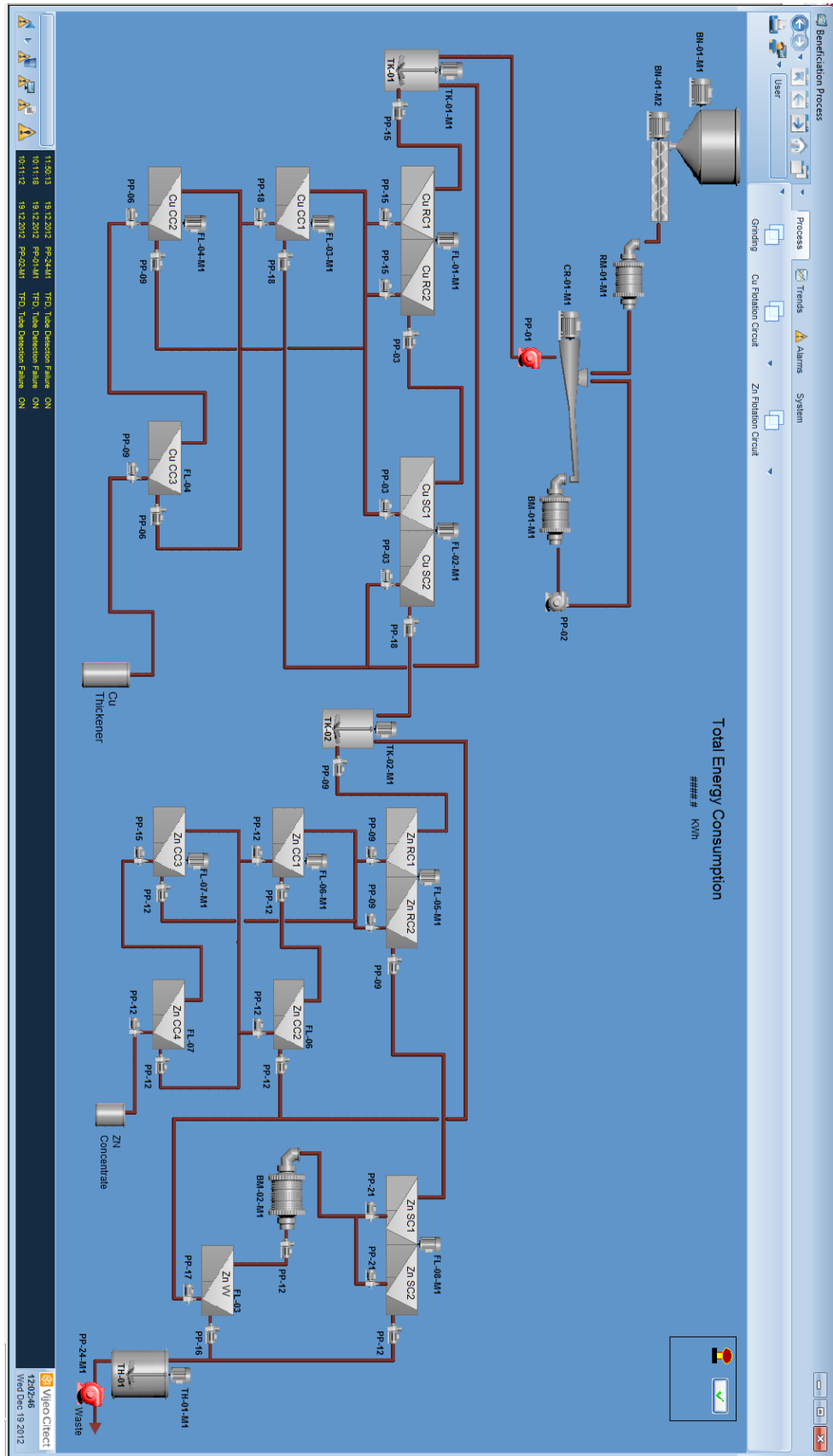
OUTPUT 1	2012/04/27	JMA
OUTPUT 2	2012/04/27	JT
OUTPUT 3	2012/04/27	M
OUTPUT 4	2012/04/27	M
OUTPUT 5	2012/04/27	M
OUTPUT 6	2012/04/27	M
OUTPUT 7	2012/04/27	M
OUTPUT 8	2012/04/27	M
OUTPUT 9	2012/04/27	M
OUTPUT 10	2012/04/27	M

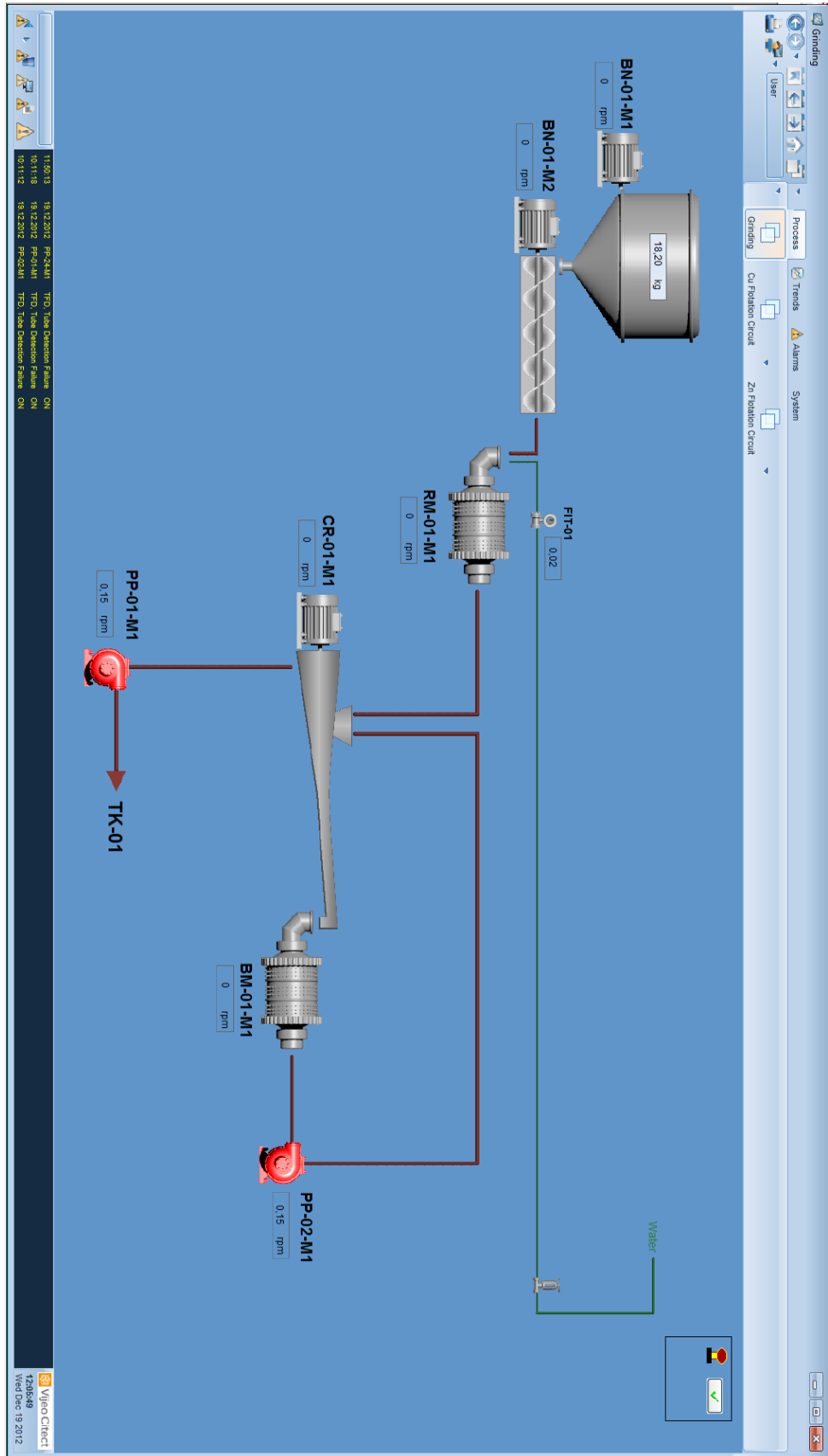
KNOWN RECORDS

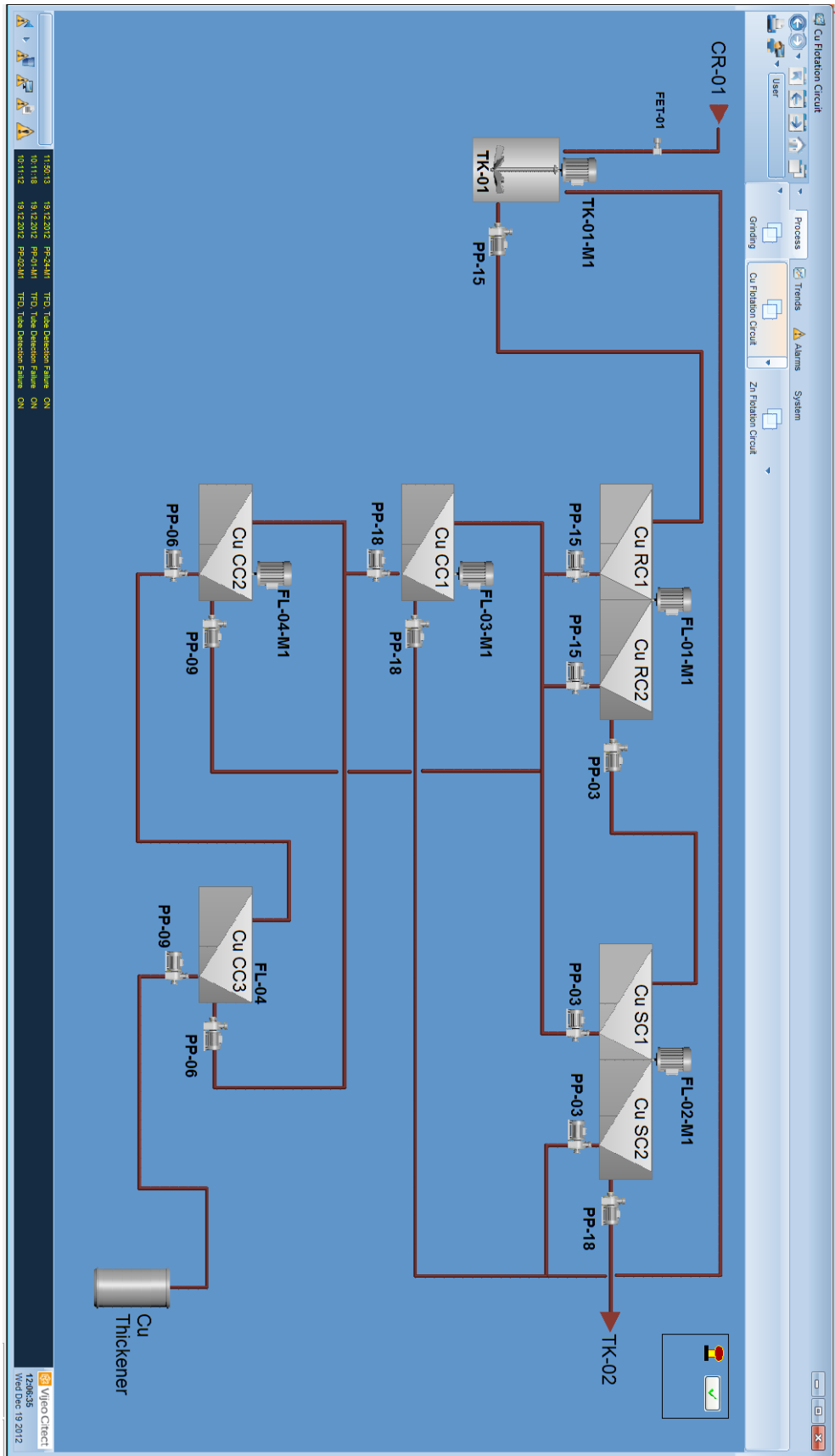
PP-05	PP-06	PP-07	PP-08	PP-09	PP-10	PP-11	PP-12	PP-13	PP-14	PP-15	PP-16	PP-17	PP-18	PP-19	PP-20	PP-21	PP-22	PP-23	PP-24
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

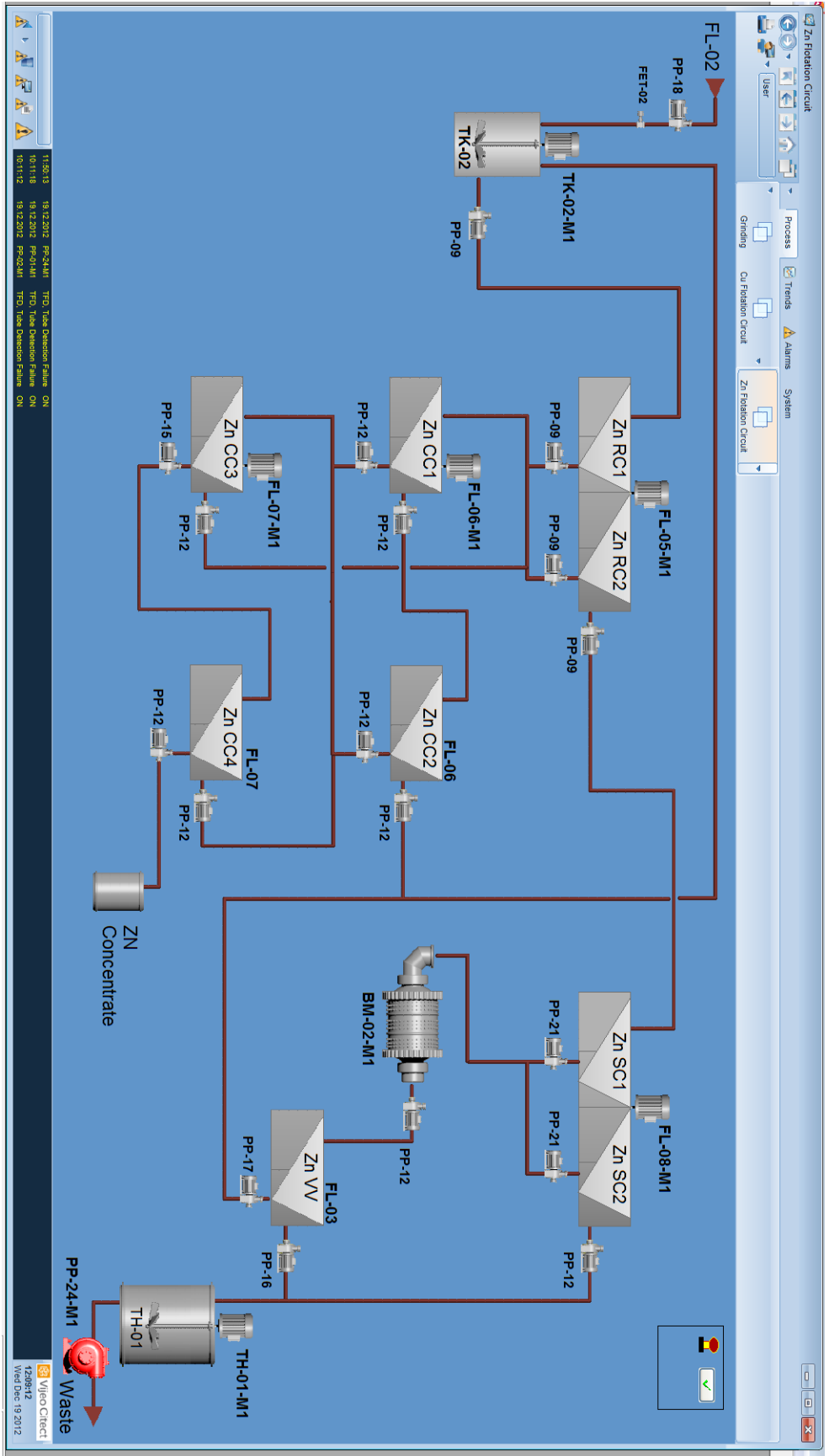


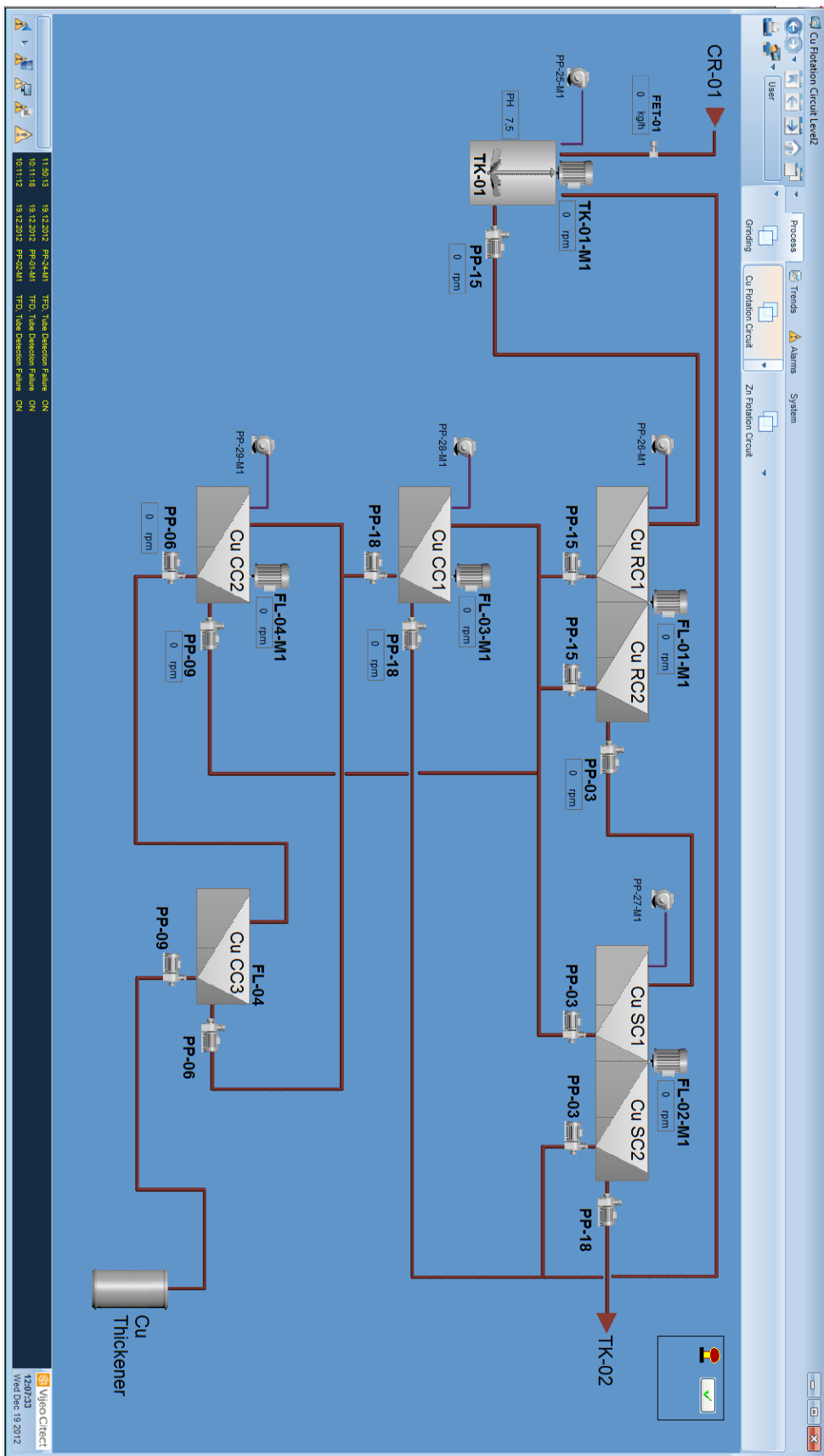


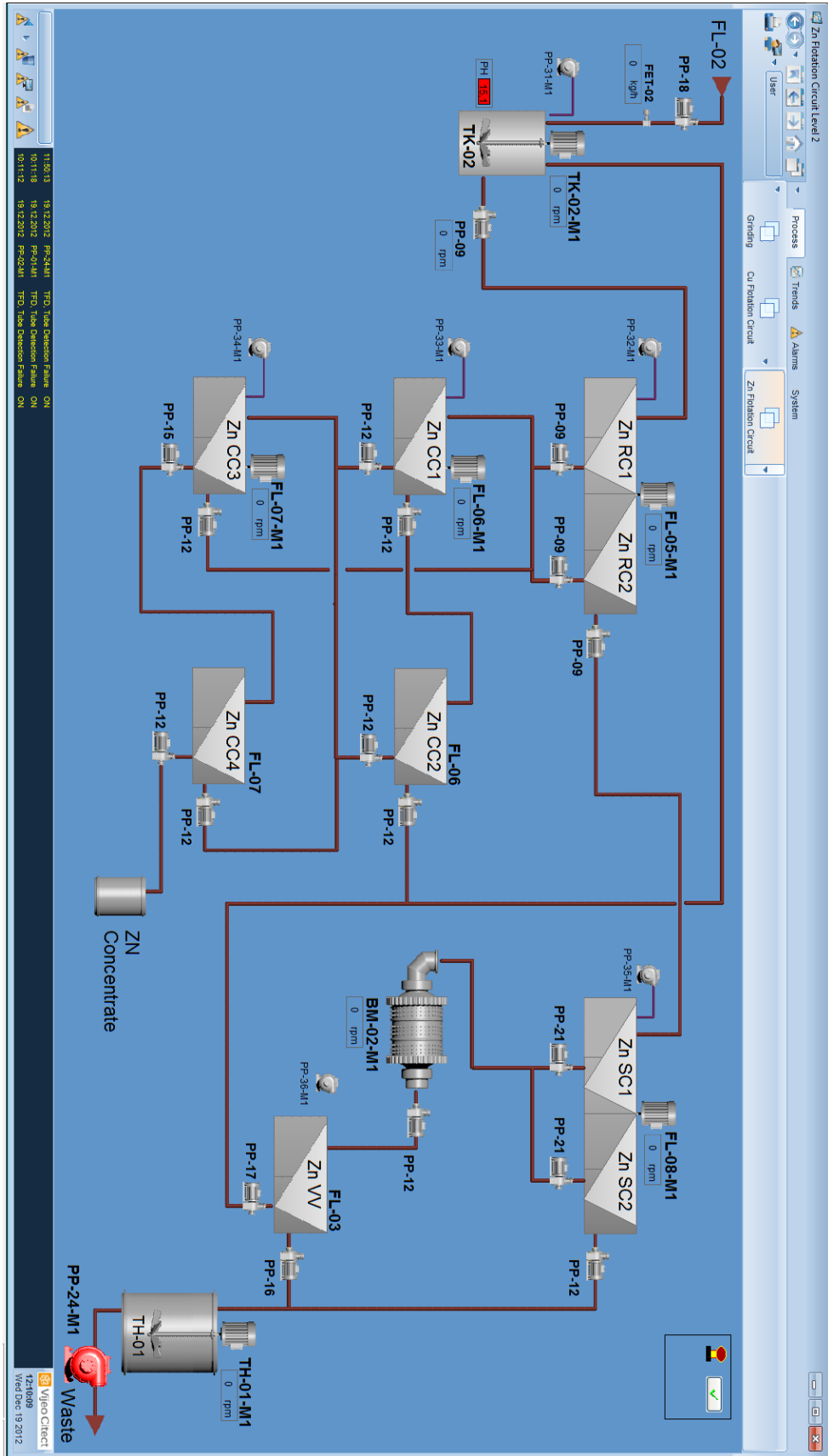






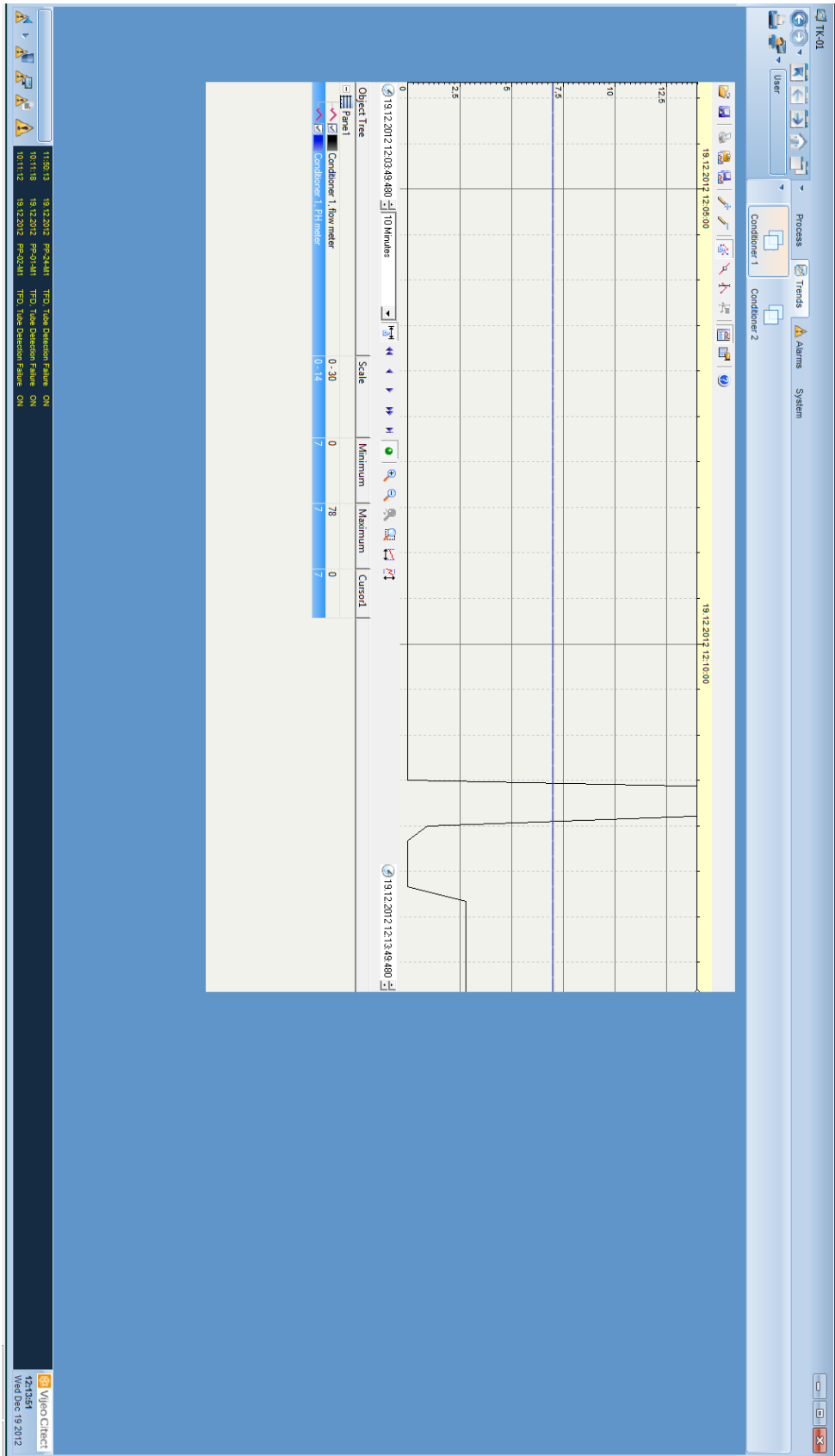






Taulukossa on lueteltu laitepositioon liittyvä genie, supergenie (ponnahdusikkuna) ja mahdollinen cicode-koodi ja cicode-koodin funktionimi.

Positio	Genie	Ponnahdusikkuna	Cicode-koodi	Cicode-funktio
BN-01-M1	-	!Altivar_3000	pumppujen_ohjaus	Asspumpput
BN-01-M2	-	!Altivar_910	pumppujen_ohjaus	Asspumpput
RM-01-M1	iso_mylly	!Altivar_1430	pumppujen_ohjaus	Asspumpput
CR-01-M1	-	!Altivar_1430	pumppujen_ohjaus	Asspumpput
BM-01-M1	iso_mylly	!Altivar_1430	pumppujen_ohjaus	Asspumpput
TK-01-M1				
TK-02-M1	valmennin	!Altivar_1380	pumppujen_ohjaus	Asspumpput
FL-01-M1 - FL-08-M1	vaahdotus_moottori	!Altivar_1410	pumppujen_ohjaus	Asspumpput
PP-03-M1 06/09/12/ 15/18/21	letkupumppu	!Altivar_pumput	pumppujen_ohjaus	Asspumpput
PP-25-M1 - PP-36-M1	kemikaalipumppu	!KP	-	-
PP-01-M1 02/24	siirto_pumppu	!SP	siirto_pumppu_ohjaus	Asssiirtopumppu
TH-01-M1	-	!Altivar_910	pumppujen_ohjaus	Asspumpput
BM-02-M1	-	!PMylly	pieni_mylly_ohjaus	Asspienimylly





## VIJEO CITECT

### 1 LISENSSIT

Vijeo Citect -lisenssi eräännyy xxx. Lisenssi täytyy uusina Schneiderin tuotetuen kautta. Vijeo Citect -oppilaitoslisenssi on RUN-tilassa 8h, jonka jälkeen virtuaalinen serveri täytyy käynnistää uudelleen.

### 2 JÄRJESTELMÄTIEDOT JA KÄYTTÄJÄTUNNUKSET

Täydellinen järjestelmätietojen ja käyttäjätunnusten Excel-lista löytyy tietokoneelta.

### 3 VIRTUAALINEN SERVERI

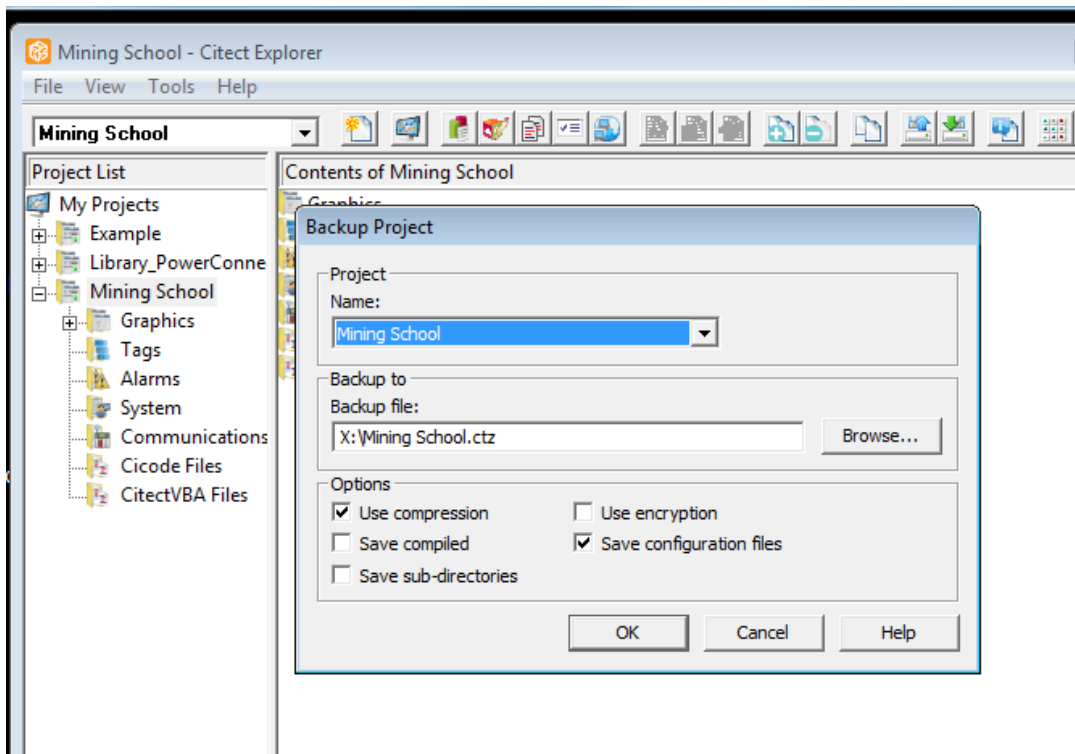
Vijeo Citect virtuaaliserveri on IP-osoitteessa xxx.

OPC Factory Serveri (OFS) tulee jättää päälle, sitä ei saa sammuttaa. Virtuaaliserveri tulee myös aina jättää päälle, eli siitä poistutaan oikean yläkulman ruksilla, ei koskaan loggautua ulos.

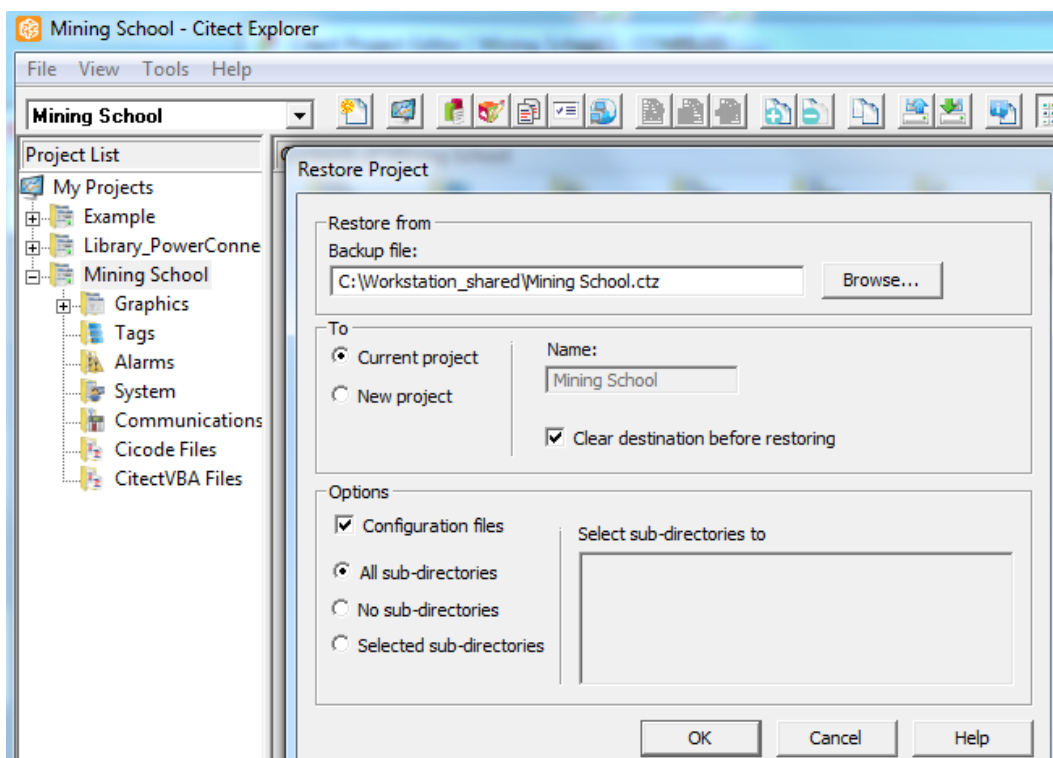
### 4 VIJEO CITECT PROJEKTIN SIIRTO PÖYTÄKONEELLE

Varsinainen kehitysympäristö on virtuaaliserverillä, jota voi käyttää pöytäkoneelta tai kannettavalta. SCADAa voidaan ajaa RUN-tilassa myös virtuaaliselta, mutta normaaliajoja varten SCADA täytyy siirtää virtuaaliselta pöytäkoneelle.

Virtuaalisella Citect Explorerista ajetaan Backup-komento, jossa tallennuspaikkana on oltava X:\ levy. Kyseinen levy on pöytäkoneen jaettu C-asema.



Pöytäkoneella Restore komennolla projekti tulee tallettaa C:\ asemalle nykyiseen projektiin.



## KAIKKI SEURAAVA TEHDÄÄN VIRTUAALISELLA!

### 5 LOGIIKAN MUUTTUJIEN TUONTI SCADAAN

Unityssä pitää aina olla Data Dictionary aktiivisena, valvomo ei toimi muuten!

Jos Unityllä on luotu uusia muuttujia tai muutettu olemassa olevien nimiä, niiden tägit täytyy tuoda Citect-ohjelmaan Citect Explorerista löytyvällä Refresh Linked Tags -komennolla, ennenkuin niitä voi käyttää Citectin puolella. Lisäksi OFS serveri täytyy käynnistää uudelleen! (Jos muuttujanimi ei vaihdu, vaikka osoittava muistipaikka muuttuisikin tätä ei tarvitse tehdä.)

Jos muuttuja on trendi-muuttuja, joka halutaan Process Analyysin avulla trendi näyttöön tulee ko. muuttuja linkata Citect Project Editorissa Trend Tag -muuttujaksi. Vastaavasti hälytysmuuttuja linkataan Alarm -muuttujaksi.

### 6 VIJEO CITECTILLA TEHDYT CICODE-KOODIT

#### 6.1 Word to Bits -cicodekoodi

Tätä käytetään koska Unity käyttää ATV32-taajuusmuuttajien komentosanaa (CMD) ja tilasanaa (ETA) WORD-tyyppisenä. Citectissa täytyy kuitenkin kirjoittaa ja lukea bittejä.

Tälle cicode-koodille ei tarvitse tehdä mitään eikä sitä missään erikseen varsinaisesti kutsuta.

#### 6.2 Genie cicode-koodi

Tämän avulla viedään muuttujat ponnahdusikkunalle (supergenieen), jonka kautta laitteiden ohjaaminen tapahtuu. Tätä tarvitaan, koska muuttujanimet ovat logiikassa niin pitkät. Esim. kemikaalipumppujen supergeniestä näkyy toteutus, jossa cicode-koodia ei tarvitse käyttää.

```

FUNCTION Asspumput (STRING sSuperGenie, STRING ATV)
STRING Tag1Value;
//Ohjelma tagien syöttämiseen pop-up ikkunalle
//ATV32 taaajuusmuuttajien ohjaukseen

Ass(-2, 1, "+ATV+" , 0);
Ass(-2, 2, "OMS_PLC_"+ATV+"_Struct_ETI", 0);
Ass(-2, 3, "OMS_PLC_"+ATV+"_RD_ATV32_VSD_Struct_CMD", 0);
Ass(-2, 4, "OMS_PLC_"+ATV+"_RD_ATV32_VSD_Struct_LFRD", 0);
Ass(-2, 5, "OMS_PLC_"+ATV+"_Struct_RFRD", 0);
Ass(-2, 6, "OMS_PLC_"+ATV+"_Struct_OPR", 0);
Ass(-2, 7, "OMS_PLC_"+ATV+"_Struct_APH", 0);
Ass(-2, 8, "OMS_PLC_"+ATV+"_Struct_RTH", 0);
Ass(-2, 9, "OMS_PLC_"+ATV+"_Struct_ALARMS_LFT", 0);
Ass(-2, 10, "OMS_PLC_"+ATV+"_Struct_ALARMS_MCC", 0);
Ass(-2, 11, "OMS_PLC_"+ATV+"_Struct_ALARMS_FLD", 0);

IF AssWin(sSuperGenie, 80, 550, 1+8+512) <> 0 THEN
RETURN 269;
ELSE
RETURN 0;
END
END

FUNCTION Asspieniimly (STRING sSuperGenie, STRING ATV)
STRING Tag1Value;
//Ohjelma tagien syöttämiseen pop-up ikkunalle
//pienen kuulamyllyn ohjaukseen

Ass(-2, 1, "OMS_PLC_"+ATV+"_Struct_ALARMS_"+ATV+"_TK_ovi", 0);
Ass(-2, 2, "OMS_PLC_"+ATV+"_Struct_ETI", 0);
Ass(-2, 3, "OMS_PLC_"+ATV+"_RD_ATV32_VSD_Struct_CMD", 0);
Ass(-2, 4, "OMS_PLC_"+ATV+"_RD_ATV32_VSD_Struct_LFRD", 0);
Ass(-2, 5, "OMS_PLC_"+ATV+"_Struct_RFRD", 0);
Ass(-2, 6, "OMS_PLC_"+ATV+"_Struct_OPR", 0);
Ass(-2, 7, "OMS_PLC_"+ATV+"_Struct_APH", 0);
Ass(-2, 8, "OMS_PLC_"+ATV+"_Struct_RTH", 0);
Ass(-2, 9, "OMS_PLC_"+ATV+"_Struct_ALARMS_LFT", 0);
Ass(-2, 10, "OMS_PLC_"+ATV+"_Struct_ALARMS_MCC", 0);
Ass(-2, 11, "OMS_PLC_"+ATV+"_Struct_ALARMS_FLD", 0);

IF AssWin(sSuperGenie, 80, 550, 1+8+512) <> 0 THEN
RETURN 269;
ELSE
RETURN 0;
END
END

FUNCTION Asssiirtopumppu (STRING sSuperGenie, STRING ATV)
STRING Tag1Value;
//Ohjelma tagien syöttämiseen pop-up ikkunalle
//isojen P1, P2, P24 ohjaukseen

Ass(-2, 1, "OMS_PLC_"+ATV+"_RD_Out_Struct_TAMU_OHJAUS", 0);
Ass(-2, 2, "OMS_PLC_"+ATV+"_RD_Out_Struct_OHJADS", 0);
Ass(-2, 3, "OMS_PLC_"+ATV+"_Struct_TAKAISINKYTTENTIA", 0);
Ass(-2, 4, "OMS_PLC_"+ATV+"_Struct_HALTYKSET_HALTYSI", 0);
Ass(-2, 5, "OMS_PLC_"+ATV+"_Struct_HALTYKSET_FLD", 0);
Ass(-2, 6, "OMS_PLC_"+ATV+"_Struct_HALTYKSET_MCC", 0);

IF AssWin(sSuperGenie, 80, 550, 1+8+512) <> 0 THEN
RETURN 269;
ELSE
RETURN 0;
END
END

```

Asspumput kertoo nimen, jolla funktiota kutsutaan. STRING määrittelee tiedot, jotka tuodaan tähän koodiin. Tässä tapauksessa kaksi tietoa: avautuvan supergenien nimi ja muutuja.

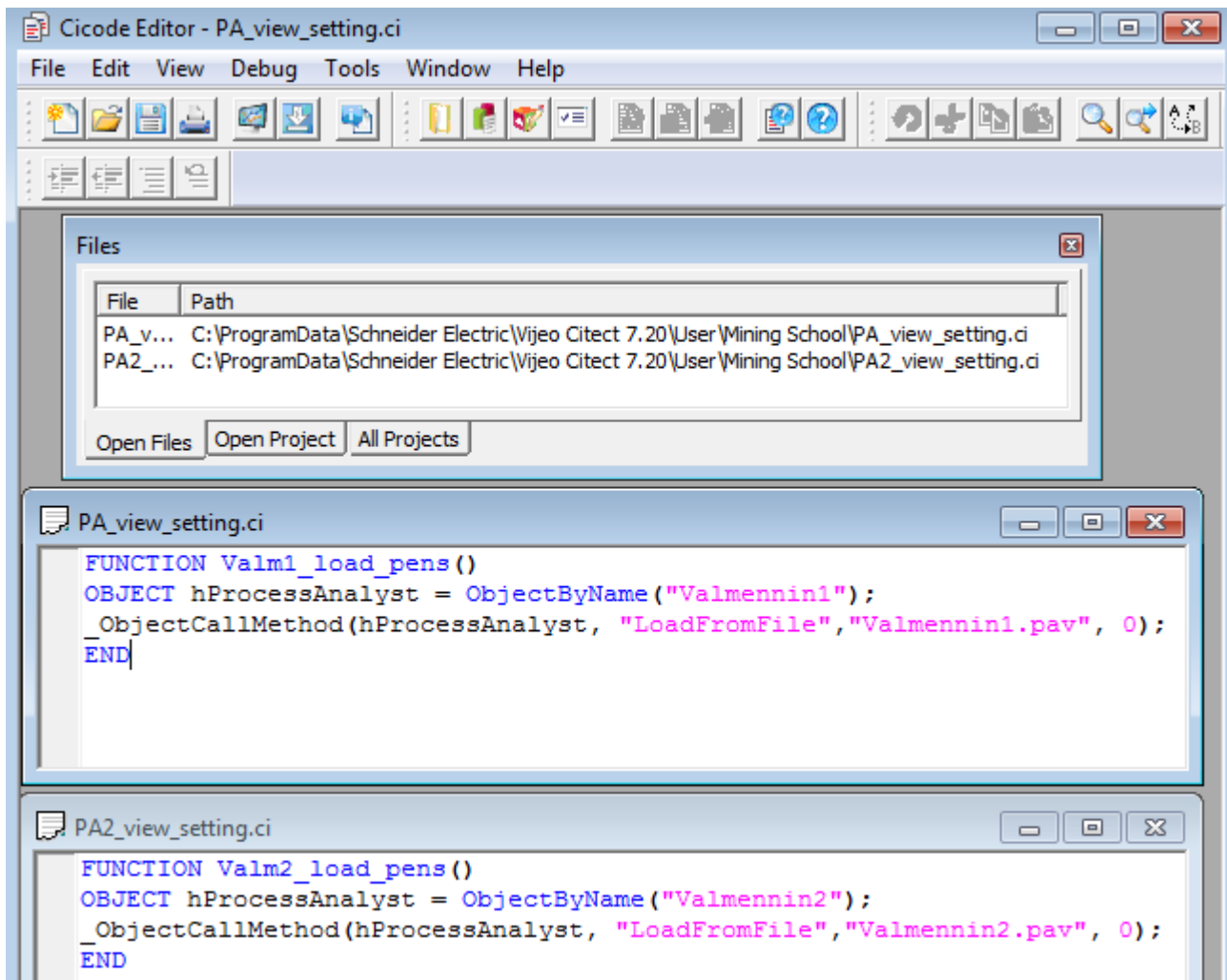
80, 550 määrittelee pixeleinä mihin kohtaan näyttöruutua ponnahdusikkuna avautuu.

1+8+512 määrittelee ikkunan ominaisuudet eli avautuu vain kerran ja ikkunan kokoa ei voi muuttaa. Cicode helpistä löytyy selostukset ja lisää vaihtoehtoja.

### 6.3 Process Analyst cicode-koodi

Trendinäytöt on toteutettu ns. Process Analystin avulla, koska sen ominaisuudet ovat laajemmat kuin pelkän trendin.

Tämän koodin avulla trendinäyttö on heti valmiina SCADAssa, eikä vaadi erikseen käynnistämistä. Jokainen trendisivu tarvitsee oman koodin.



## 7 GENIE JA SUPERGENIE

Genie on uudelleen käytettävä symboliryhmä ja supergenie on ponnahdusikkuna. Kumpikin toimii itsenäisesti.

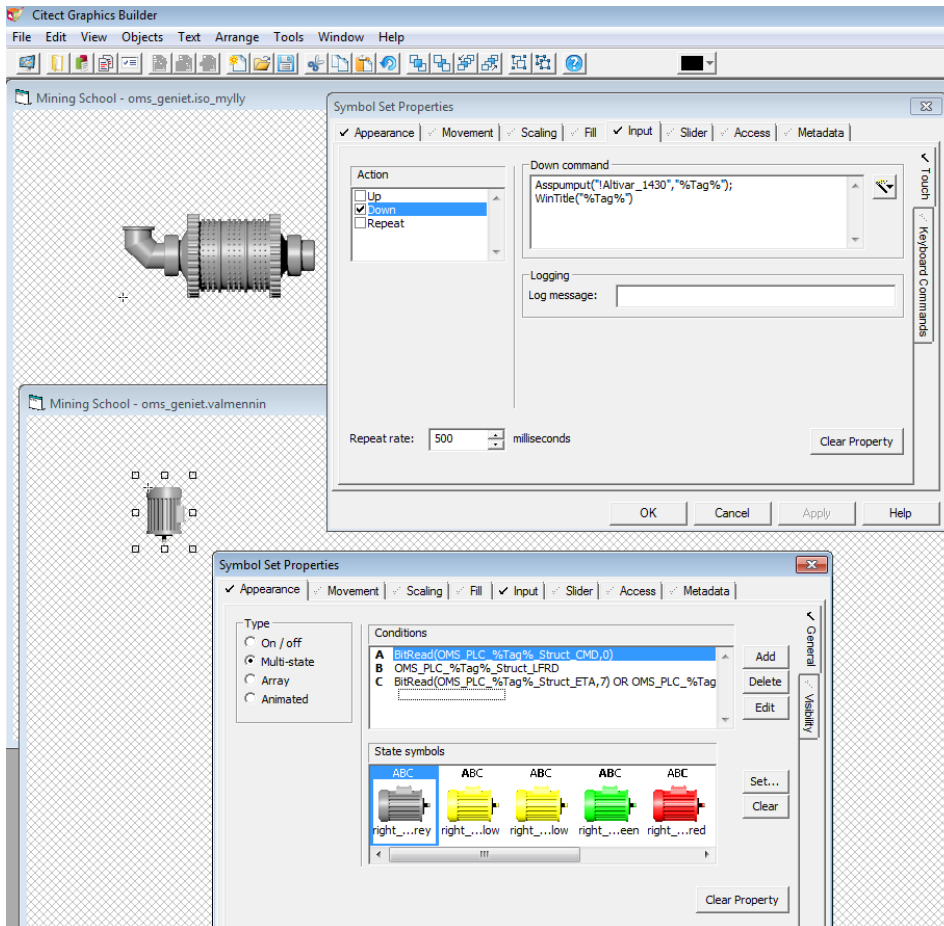
Taulukossa on lueteltu käytetyt genit, supergeniet ja niihin liittyvät cicode-koodit.

Positio	Genie	Ponnahdus- ikkuna	Cicode-koodi	Cicode -funktio
BN-01-M1	-	!Altivar_3000	pumppujen_ohjaus	Asspumpput
BN-01-M2	-	!Altivar_910	pumppujen_ohjaus	Asspumpput
RM-01-M1	iso_mylly	!Altivar_1430	pumppujen_ohjaus	Asspumpput
CR-01-M1	-	!Altivar_1430	pumppujen_ohjaus	Asspumpput
BM-01-M1	iso_mylly	!Altivar_1430	pumppujen_ohjaus	Asspumpput
TK-01-M1 TK-02-M1	valmennin	!Altivar_1380	pumppujen_ohjaus	Asspumpput
FL-01-M1 - FL-08-M1	vaahdotus_moottori	!Altivar_1410	pumppujen_ohjaus	Asspumpput
PP-03-M1 06/09/12/15/ 18/21	letkupumppu	!Altivar_pumpput	pumppujen_ohjaus	Asspumpput
PP-25-M1 - PP-36-M1	kemikaalipumppu	!KP	-	-
PP-01-M1 02/24	siirto_pumppu	!SP	siir- to_pumppu_ohjaus	Asssiirto- pumppu
TH-01-M1	-	!Altivar_910	pumppujen_ohjaus	Asspumpput
BM-02-M1	-	!PMylly	pieni_mylly_ohjaus	Ass- pienimylly

Graphic Builderissä kuvaan sijoitetun valmiin genien tiedot saa näkyviin klikkaamalla kuvaketta CTRL+ALT+kaksoisklikkaus.

Geniet luodaan Graphics Builderissä avaamalla uusi sivupohja (New > Genie). Valitaan haluttu Symbol Set (eli laitteen kuvake) ja määritellään sille halutut tiedot.

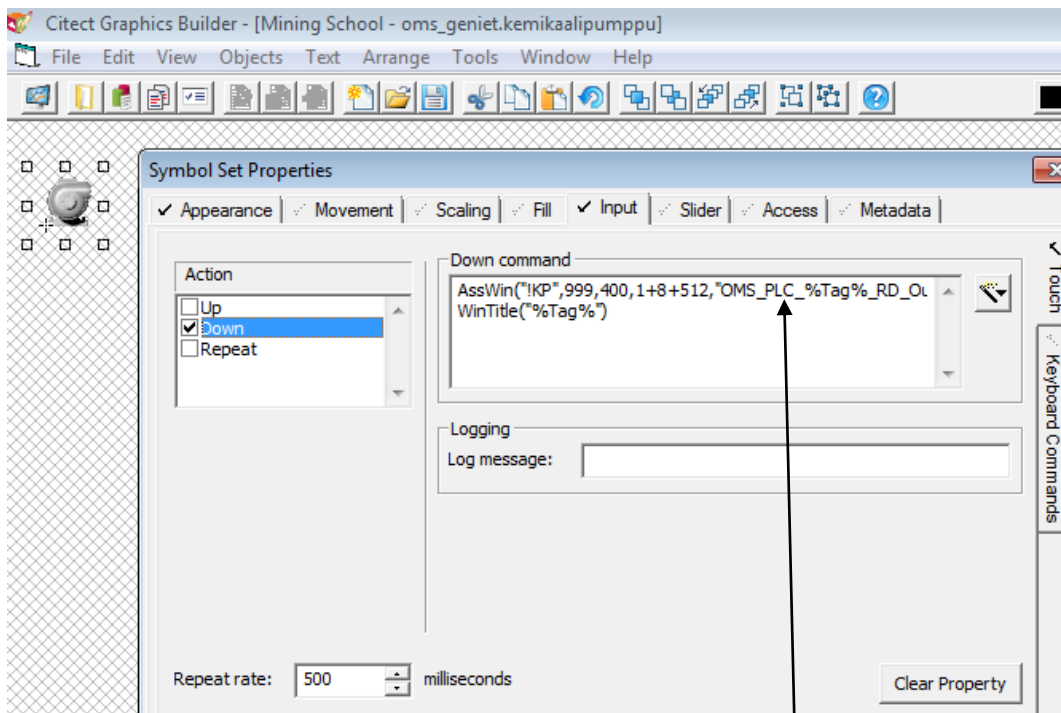
Input Down määrittelee mitä tapahtuu kun kuvaketta painetaan hiirellä. Tässä tapauksessa kutsutaan cicode-funktiota nimeltä Asspumpput, jolle viedään tietoina avautuvan ponnahdusikkunan nimi ja muuttujanimi. WinTitle -toiminto laittaa ponnahdusikkunan osoitteeksi muuttujanimen. Muuttujanimi on muoto %Tag%, koska ohjelma osaa kysyä %-muotoisen tiedon. Tieto syötetään genietä käytettäessä, esim. PP01M1.



Appearance välilehdelle syötetään kuvakkeen värimäärittelyt ja mitä muuttujia luetaan. Koska tästä muodostetaan yleiskuvake, jossa kulloinkin laitenumero kysytään kuvaketta käyttöönotettaessa, niin muuttujanimen laiteosa on korvattu %Tag% stringillä.

BitRead(x,y) eli x on luettava muuttujanimi ja y on luettava bitti (0,1,2,3...).

Jos genietä muodostettaessa ei käytetä cicode-koodia niin vastaavat tiedot syötetään suoraan genieen sen sijaan että kutsuttaisiin cicodea. Syötettäessä käytetään funktiota AssWin, jossa avautuvan ponnahdusikkunanimi, ikkunan pixelit näytöllä, ikkunan määrittelyt sekä vietävien muuttujien nimet (tässä tapauksessa käytetty taas %Tag% muotoa koska kyseessä uudelleen käytettävä genie). Kemikaalipumpujen on toteutus tällainen.

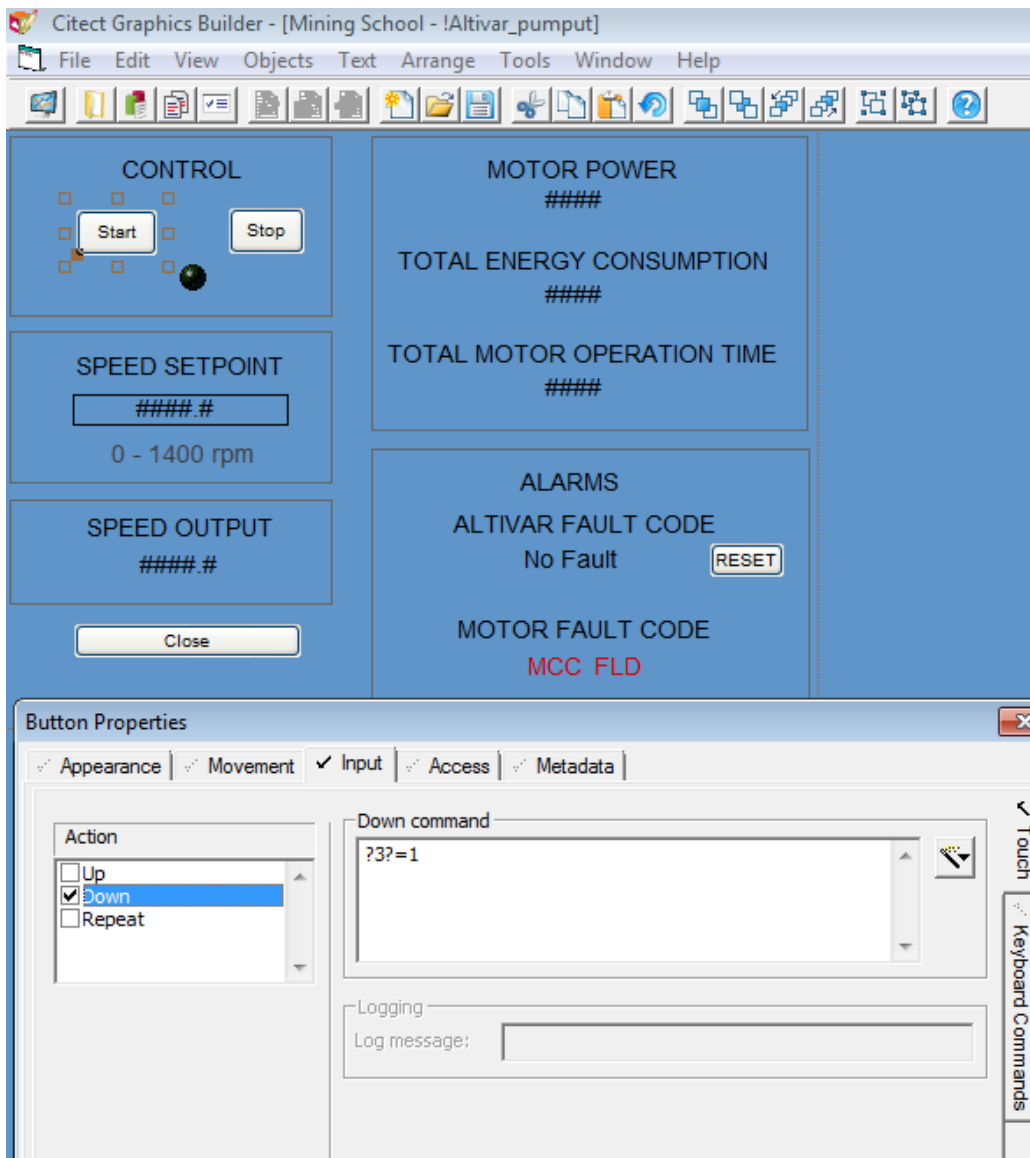


Supergenie eli ponnausikkuna luodaan Graphics Builderissä avaamalla uusi sivupohja New > Page > Template Style : Blank. Piirretään haluttu ikkuna ja määritellään sille käytettävät muuttujat, jotka käytettäessä Cicode-koodia tulevat koodista ja on numeroitu 1, 2... Numeron merkitys löytyy siis koodista.

Jos Cicode-koodia ei käytetä (kuten kemikaalipumput) niin numerolla viitataan Symbol Set Properties valinnalla määritelyihin muuttujiin.

Supergenie tallennetaan nimellä !sivunnimi, jotta ohjelma osaa tunnistaa ko. sivun nimen omaan supergenieksi.



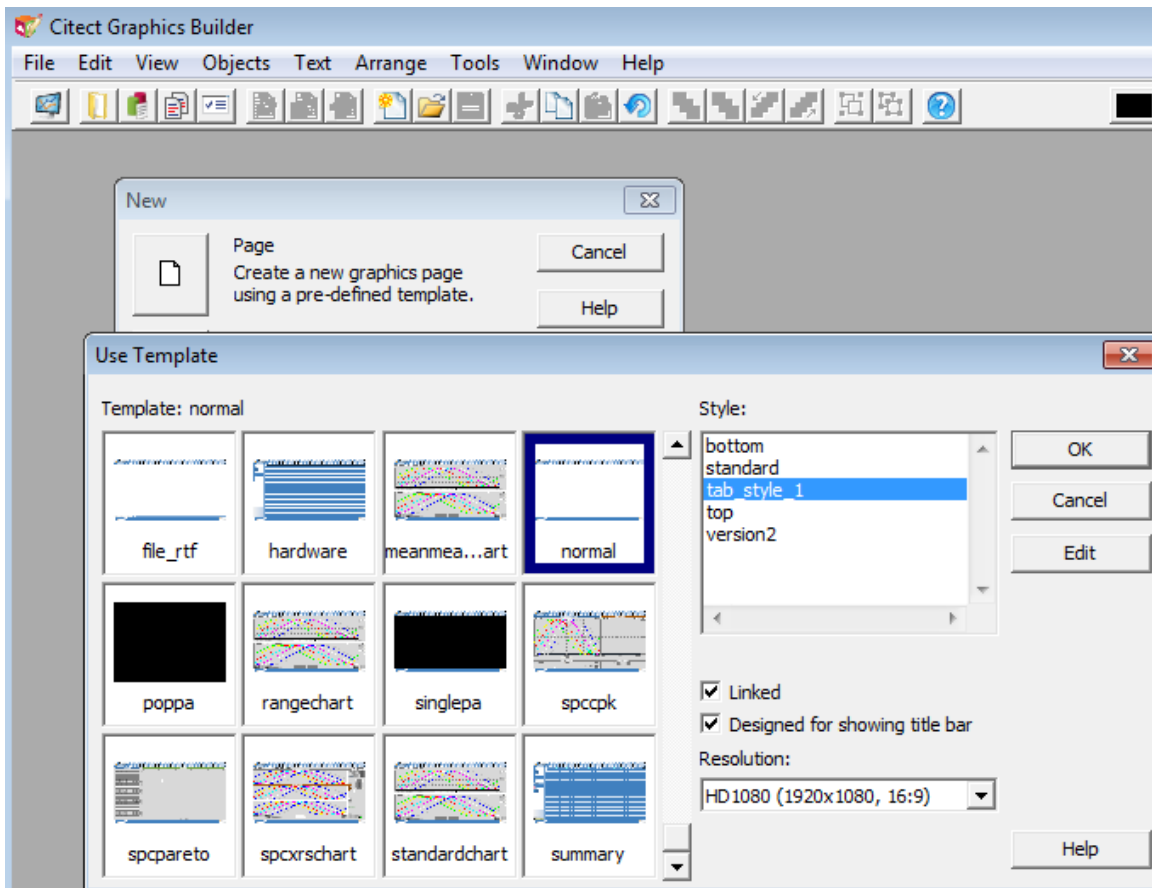


Esimerkissä painettaessa Start-nappia kirjoitetaan muuttujan numero 3 arvoksi yksi.

Ponnahdusikkunan koko määritellään File > Properties > View Area.

## 8 UUSI NÄYTTÖSIVU

Luotaessa uusi näyttösivu valitse seuraavat asetukset:



Sivun nimi, joka näkyy RUN-moodissa vasemmassa yläreunassa määritellään Graphic Builderissa File > Properties > Window Title.

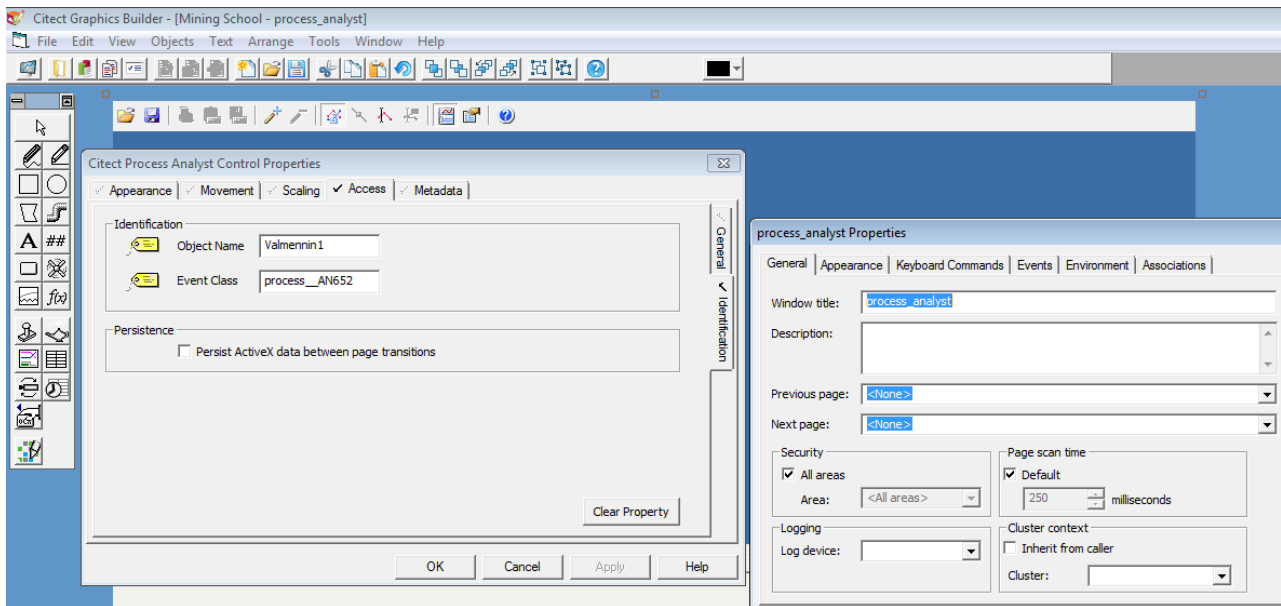
## 9 PROCESS ANALYST

Trendinäytöt on muodostettu Process Analystilla. Jotta trendi avautuisi heti sivulle mentäessä täytyy käyttää cicode-koodia.

Aluksi täytyy määrittellä Graphic Builderissa PA objektille sen nimi, valitsemalla hiiren oikealla Properties.

Sen jälkeen määritellään cicode-koodiin funktionimi ja äsken luoto objektin nimi sekä Load from File, jossa määritellään minkä niminen tiedosto avataan sivulle mentäessä.

Sitten jälleen Graphic Builderista määritellään File > Properties > Events > On Page Shown kutsuttavan funktion nimi, joka määriteltiin ciodessa.



Sitten Citect RUN-moodiin ja määritellään kynät (muuttujat), joita halutaan näyttää ja tallennetaan tämä näkymä sillä nimellä, joka cicode-koodissa määriteltiin kohtaan Load from File.

Jos haluttua muuttujaa ei löydy RUN-moodissa Process Analyst -ikkunasta täytyy käydä määrittelemässä ko. muuttuja Citect Project Editorissa > Tags > Trend Tags.

## 10 HÄLYTYKSIEN MÄÄRITTELY

Uudet hälytykset täytyy määritellä Citect Project Editorissa > Alarm > Digital Alarm.

Alarm Tag Name täytyy olla joka hälytykselle uniikki ja erikoismerkkejä ei voi käyttää eikä välejä.

## 11 SCADASSA NÄKYVÄ VALIKKORAKENNE

Valikkorakenne, joka näkyy SCADAssa RUN-moodissa määritellään Citect Project Editorissa > System > Menu Configuration.

## 12 PROJEKTIN TIETOJEN JÄRJESTÄMINEN

Jos Citect Project Editorissa on tehty paljon muutoksia hälytyksiin, tageihin tai valikkorakenteeseen (lisätty, poistettu, muutettu) on hyvä jossain vaiheessa ajaa Citect Project Editorissa > File > Pack. Tämä komento poistaa lopullisesti tiedot ja ”korjaa” rakenteen.

## 13 RUN-MOODI

Saadaksesi sivut näkymään RUN-moodissa täytyy Citect Project Editorissa ajaa Compile ja sen jälkeen Run. Tai pikatoimintona tallennettuasi Graphic Builder sivun valitse F5. Se tekee käännöksen ja ajaa projektin.

Saadaksesi Builderissa tehdyt muutokset näkymään RUN-moodissa täytyy ajoikkuna ensin sulkea ja avata sitten uudelleen.

Citect ajaa sen projektin, joka on Citect Explorerissa valittu aktiiviseksi!

Voidaksesi tehdä mitään ajoikkunassa täytyy kirjautua sisään!

## 14 KÄYTTÄJÄRYHMÄT

Voidaksesi tehdä mitään ajoikkunassa täytyy kirjautua sisään!

Käyttäjien ja oikeuksien määrittely tapahtuu Citect Project Editorissa > Systems > Users ja Roles.

Käyttäjänimi: xxx, Salasana: xxx voi tehdä normaalit operointitoimenpiteet prosessille. Tämä on se käyttäjänimi, jota normaalisti käytetään. Sillä on roolissa määritellyt oikeudet, joita on hyvin minimaalinen määrä.

Käyttäjänimi: xxx, Salasana: xxx. Tälle on roolissa määritelty tason 1 oikeudet.

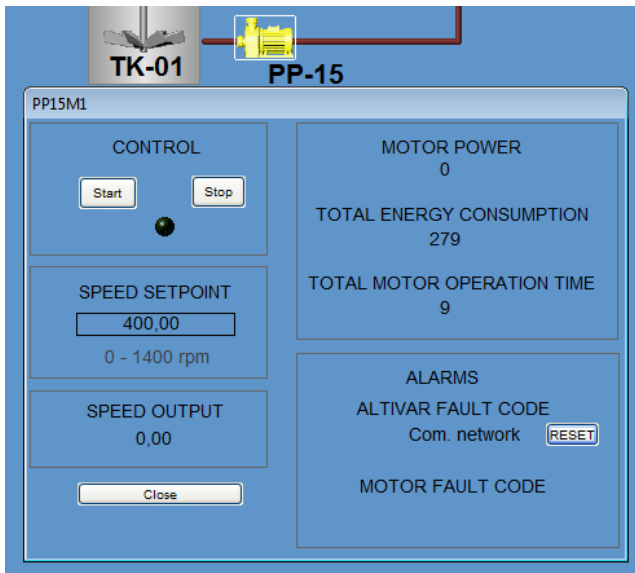
Käyttäjänimi: xxx, salasana: xxx. Tällä on roolissa täydet oikeudet (1..8).

## 15 ATV32-MOOTTOREIDEN AJAMINEN

Start-napin painallus asettaa moottorille käyntikäskyn, jos käsky kirjautuu logiikkaan oikein mustamerkkivalo vaihtuu vihreäksi.

Speed Setpoint kohtaan kirjoitetaan nopeusohje, jonka jälkeen moottori käynnistyy, jos käyntikäsky on annettu. Pelkkä nopeusohje ei käynnistä moottoria.

Pumppukuvake on keltainen, jos käyntikäsky tai nopeusohje on annettu. Harmaa, jos kumpaakaan ei ole määritetty ja punainen, jos toiminnassa on joku häiriö. Häiriö voi olla taajuusmuuttajassa tai sitten moottorisuojakytkin tai turvakytkin on kytketty pois päältä.



Jos moottori ei käynnisty paina Reset-nappia, taajuusmuuttaja voi olla tilassa jossa se ei ota käskyjä vastaan. Myös taajuusmuuttajan viat täytyy kuitata Reset-napilla. Reset-nappia voi joutua painamaan useammin, tärkeää on että hiiren painiketta pidetään riittävän kauan pohjassa, jotta kuittaustieto ehditään kirjoittaa logiikalle.

## VIJEO HISTORIAN

### 1 LISENSSIT

Vijeo Historian lisenssi eräänny xx. Ampla lisenssi eräänny xx. Lisenssit täytyy uusia Schneiderin tuotetuen kautta.

### 2 VIRTUAALISET SERVERIT

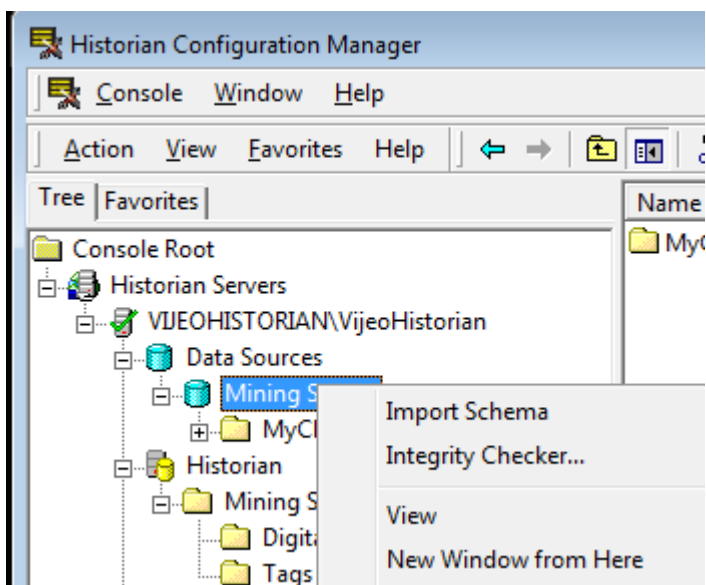
Vijeo Historian virtuaalinen serveri IP-osoite xxx

Virtuaali serveri tulee aina jättää päälle, eli siitä poistutaan oikean yläkulman ruksilla, ei koskaan loggaututa ulos.

Ampla virtuaalinen serveri IP-osoite xxx.

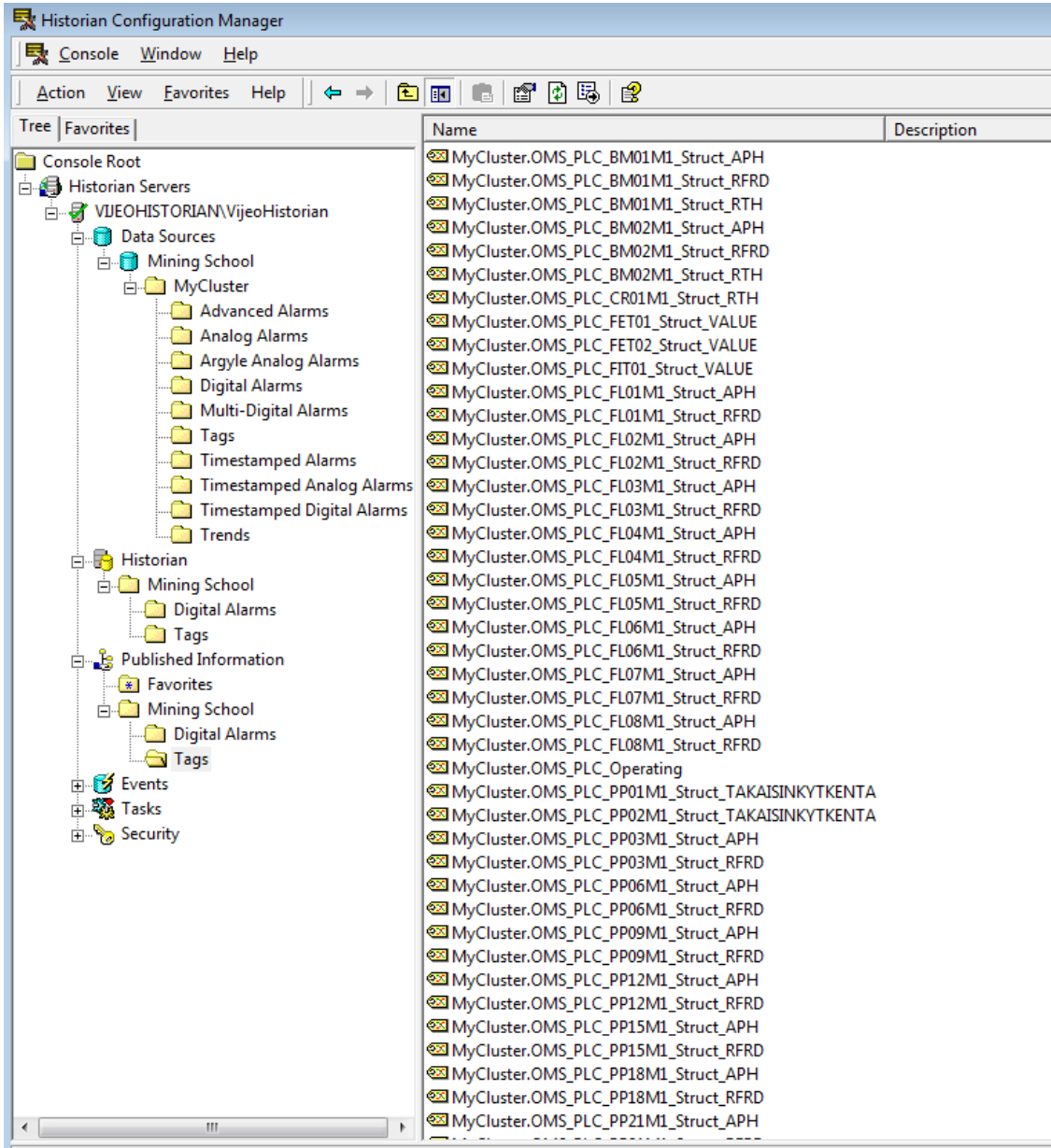
### 3 TAGIEN TUONTI VIJEO HISTORIAAN

Uudet muuttujat saa Historianiin valitsemalla Historian Configuration Managerissa Import Schema -toiminnon.



#### 4 TIEDON JULKAISU HISTORIASSA JA VIENTI AMPLAAN

Configuration Managerissa halutut tagit pitää siirtää Data Sourcesta Historianiin ja sieltä Published Informationiin, jotta tieto olisi Amplan käytettävissä.



Lisäksi pitää käynnistää Server Data Manager uudelleen valitsemalla Data Service – Stop > Data Service – Start. Tämän jälkeen tagien pitäisi näkyä Amplassa.

