

Miika Haapala

VAIHTOEHTOISET ETHERCAT MASTER -TOTEUTUKSET

VAIHTOEHTOISET ETHERCAT MASTER -TOTEUTUKSET

Miika Haapala
Opinnäytetyö
Kevät 2021
Sähkö- ja automaatiotekniikka
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Sähkö- ja automaatiotekniikka, Automaatiotekniikka

Tekijä: Miika Haapala
Opinnäytetyön nimi: Vaihtoehtoiset EtherCAT Master -toteutukset
Työn ohjaaja: Satu Vähänikkilä, Esa Harju
Työn valmistumislukukausi ja -vuosi: Kevät 2021
Sivumäärä: 54

Tämän opinnäytetyön tarkoituksena oli tutkia vaihtoehtoja nykyisen toimittajan kehitysympäristölle, PLC-tuotteelle ja EtherCAT-väyläohjaimelle. PLC-tuotteita käytetään automaatiojärjestelmien ohjaamiseen. EtherCAT-väyläohjaimia käytetään I/O-laitteiden tiedonsiirtoon. Tavoitteena oli tutkia ja vertailla muutamaa potentiaalista vaihtoehtoa. Vaihtoehtoista haluttiin tutkia, löytyisikö jokin yrityksen käyttöön sopivampi ratkaisu.

Tutkimiseen kuului tarkastelun kohteena olevien EtherCAT Mastereiden tuotetietojen tutkiminen ja niiden asennus. Jokaisella EtherCAT Masterilla toteutettiin yksinkertainen esimerkkiohjelma, että nähtäisiin, miten tiedonsiirron järjestäminen onnistuu niiden avulla.

Työssä saatiin suoritettua tutkiminen ja vertailu kolmelle EtherCAT Masterille ja työn tavoitteet täyttyivät. Tutkimisen kohteena olevista EtherCAT Mastereista saatiin tietoa, mikä niistä saattaisi sopia yrityksen tarkoituksiin parhaiten. Jokaisella tutkimisen kohteena olevalla EtherCAT Masterilla saatiin tehtyä esimerkkiohjelma, joka luki ja kirjoitti I/O-dataa.

Asiasanat:

PLC, EtherCAT, KINGSTAR, CODESYS, Acontis

ABSTRACT

Oulu University of Applied Sciences
Degree Programme in Electrical and Automation Engineering, Option of Automation

Author: Miika Haapala
Title of thesis: Alternative EtherCAT Master Implementations
Supervisor: Satu Vähänikkilä, Esa Harju
Term and year when the thesis was submitted: Spring 2021
Number of pages: 54

The purpose of this thesis was to study alternatives for development environment, PLC-product and EtherCAT Master of current supplier. PLC-products are used to control automation systems. EtherCAT Masters are used to move data to I/O-devices. The target was to study and compare a few potential options. It was to be studied if any of the options was better for the company.

Studying was composed of researching information and installing the EtherCAT Masters. An example program was made with each of the EtherCAT Masters.

The studying and comparing was completed and the purpose of this work was fulfilled. Information was gained about the EtherCAT Masters and their suitability for the purposes of the company. An example program, which wrote and read I/O-data, was made with each of the EtherCAT Masters.

Keywords:

PLC, EtherCAT, KINGSTAR, CODESYS, Acontis

SISÄLLYS

1	JOHDANTO	6
2	ETHERCAT	7
3	KÄYTÖSSÄ OLEVA LAITTEISTO	10
3.1	Aaeon Up Squared	10
3.2	Beckhoff C6015-0010	10
3.3	EK1100	10
3.4	EL1008	11
3.5	EL2008	12
4	TARKASTELTAVIEN ETHERCAT MASTEREIDEN VALINTA	13
4.1	Valitut EtherCAT Masterit	14
4.2	Lisenssiehdot	15
4.2.1	GPL	15
4.2.2	LGPL	15
4.2.3	BSD	15
4.2.4	MIT	16
5	KINGSTAR ETHERCAT MASTER	17
5.1	KINGSTARin asennus	19
5.2	KINGSTARin käyttö	24
5.3	Yhteenvedo KINGSTARista	31
6	ACONTIS ETHERCAT MASTER	32
6.1	Acontisin asennus	35
6.2	Acontisin käyttö	37
6.3	Yhteenvedo Acontiksestä	40
7	CODESYS ETHERCAT MASTER	41
7.1	CODESYSin asennus	42
7.2	CODESYSin käyttö	45
7.3	Yhteenvedo CODESYSistä	51
8	YHTEENVETO	52
	LÄHTEET	53

1 JOHDANTO

Tässä työssä etsittiin ja tutkittiin vaihtoehtoja nykyisen toimittajan kehitysympäristölle, PLC-tuotteelle ja EtherCAT-väyläohjaimelle. PLC-tuotteita käytetään automaatiojärjestelmien ohjaamiseen. Ne ovat ohjelmoitavia ohjainlaitteita. Tässä työssä käytetyt PLC-tuotteet olivat SoftPLC-tuotteita, eli ne toimivat tietokoneella. Myös työssä käytetyt EtherCAT Masterit olivat tietokoneella ajettavia ohjelmia. Työ tehtiin Orbis Systems Oy:lle. Kyseisten vaihtoehtojen ominaisuuksia vertailtiin, niiden käyttöönottoa kokeiltiin ja niillä kokeiltiin väyläliikenteen muodostamista.

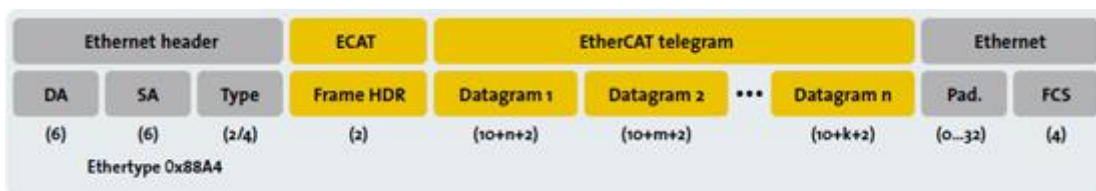
Vaihtoehtoisten EtherCAT Mastereiden valintaan kohdistui tiettyjä kriteerejä. Valittavan EtherCAT Masterin tulisi mielellään olla käytettävissä Linuxin kanssa ja se saisi olla kaupallinen tuote tukipalveluiden turvaamiseksi. Valinnassa käytettiin myös muita kriteerejä. Valittavan EtherCAT Masterin tulisi olla käytettävissä sarjatuotannossa ja sitä käytettäessä kehityksen sekä käyttöönoton tulisi olla mahdollisimman kustannustehokasta.

2 ETHERCAT

EtherCAT on reaaliaikainen teollisuuskäyttöön tarkoitettu Ethernet-pohjainen väyläratkaisu. Sen on alun perin kehittänyt Beckhoff. EtherCAT-protokolla on määritelty standardissa IEC61158. Pääfokus EtherCAT:in kehityksessä on ollut lyhyet syklijajat ja tarkka synkronisaatio. EtherCAT julkaistiin alun perin vuonna 2003. (1.)

EtherCAT-pohjainen laitteisto koostuu master-laitteesta, joka ohjaa väyläkommunikaatiota ja slave-laitteita. Viestintä EtherCAT:ssa toimii siten, että master-laite lähettää viestin, joka kulkee slave-laitteiden läpi. Kun viesti kulkee slave-laitteiden läpi, slave-laitteet muokkaavat viestiä sen kulkiessa slave-laitteelta seuraavalle. Kun väylän viimeisenä oleva laite tunnistaa, että sen jälkeen ei ole enää lisää laitteita, se lähettää väylän läpi kulkeneen viestin takaisin master-laitteelle. (1.)

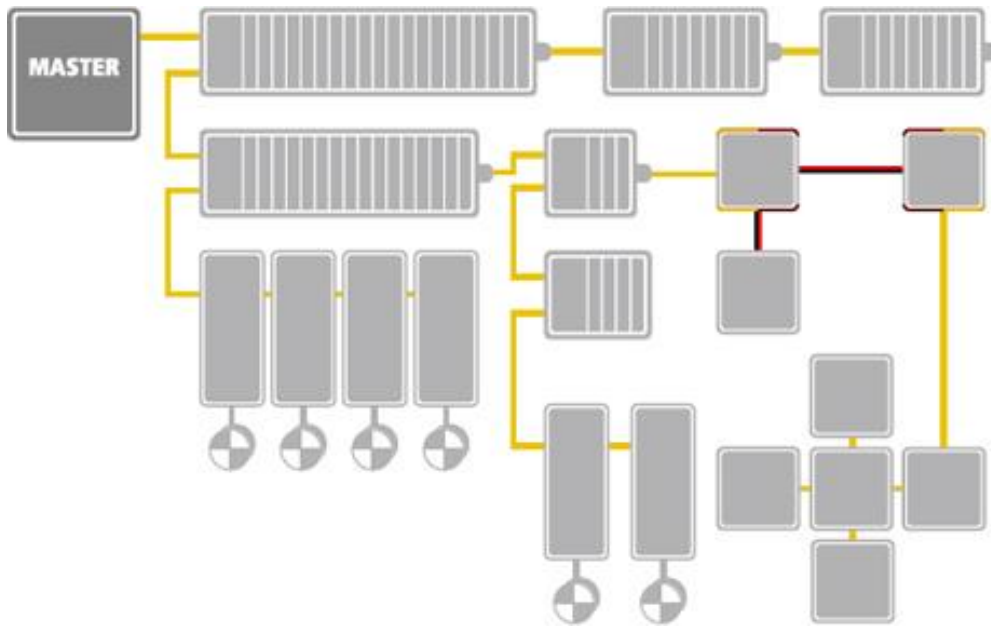
EtherCAT käyttää viestintään standardia Ethernet-kehystä. Kehyksen sisältö on kuvattu kuvassa 1. Käynnistyksen yhteydessä master-laite konfiguroi slave-laitteiden käyttämän prosessidatan. Kullekin slave-laitteelle on allokoituna oma osionsa prosessidatasta. Kun EtherCAT-kehys kulkee slave-laitteen läpi, se pystyy lukemaan tai muokkaamaan itselleen allokoitua osiota prosessidatasta. (1.)



KUVA 1. EtherCAT-kehysten sisältö

Master-laite pystyy myös kommunikoimaan slave-laitteiden kanssa suoraan. Tätä käytetään käynnistyksen yhteydessä määrittelemään väylän topologia. Verkon konfiguroinnin jälkeen master-laite voi asettaa kullekin slave-laitteelle kiinteän osoitteen ja käyttää sitä suoraan kommunikointiin slave-laitteen kanssa. Master-laitteen kommunikointia suoraan slave-laitteelle kutsutaan mailbox-kommunikaatioksi. (1.)

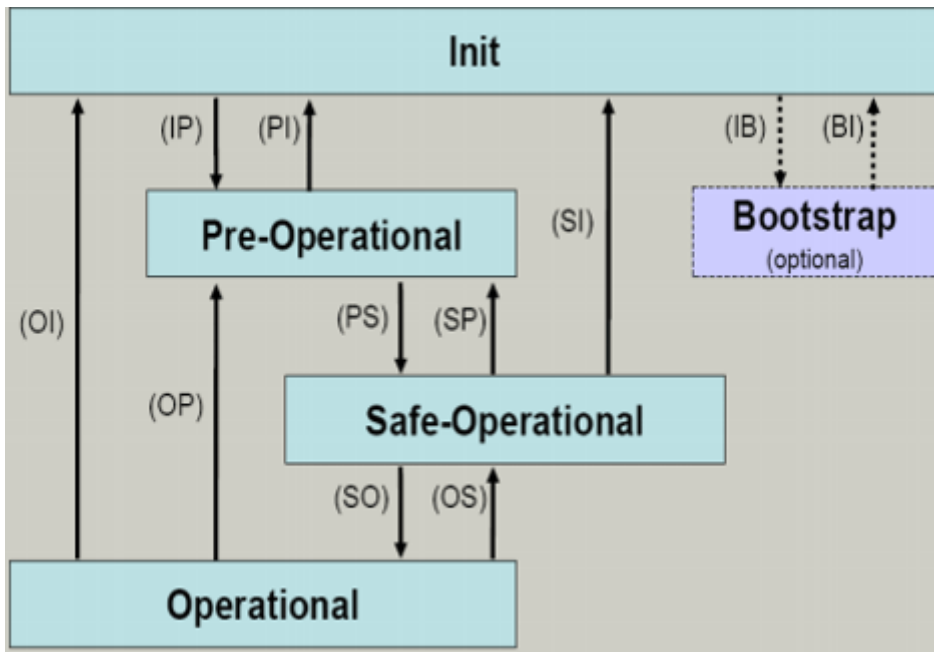
EtherCAT-väylän topologia voi olla käytännössä minkä muotoinen tahansa. Topologia voi olla suora topologia, puutopologia taikka tähtitopologia. EtherCAT-väylän kanssa voidaan myös käyttää rengastopologiaa, joka tuo kaapelointiin samalla kahdennuksen. EtherCAT-väylän topologia voi olla myös yhdistelmä kaikkia edellä mainittuja topologioita. Esimerkkitopologia on esitetty kuvassa 2. (1.)



KUVA 2. Mahdollinen EtherCAT-väylän topologia

EtherCAT Slavella voi olla neljä eri tilaa, jotka muodostavat tilakoneen. Mahdolliset tilat ovat Init, Pre-Operational, Safe-Operational ja Operational. Näiden tilojen muodostama tilakone on esitetty kuvassa 3. Tilanmuutokset voivat kulkea vain kuvan nuolien osoittamien reittien mukaisesti. EtherCAT-väylässä master-laite ohjaa slave-laitteiden tilaa. Kun slave-laite käynnistyy, se saavuttaa Init-tilan. Init-tilassa master-laite asettaa slave-laitteelle mailbox-kommunikaation päälle. Kun Init-tilasta siirrytään Pre-Operational-tilaan, master-laite tarkastaa, onko slave-laitteen mailbox initialisoitu oikein. Pre-Operational-tilassa on mahdollista käydä mailbox-kommunikaatiota, mutta prosessidatakommunikaatio ei ole mahdollista. Pre-Operational-tilasta siirrytään seuraavaksi Safe-Operational-tilaan. Ennen Safe-Operational-tilaan siirtymistä slave-laite kopioi sisääntulojen datan muistiinsa. Safe-Operational-tilassa sekä mailbox-kommunikaatio että prosessidatakommunikaatio ovat mahdollisia. Kuitenkin I/O-laitteiden tapauksessa slave-laitteen ulostulot ovat turvallisessa tilassa. Esimerkiksi binääristen ulostulokorttien tapauksessa ulostulot ovat jännitteettömiä. Ennen kuin master-laite voi kytkeä slave-laitteen Operational-tilaan, tulee slave-laitteen lähettää validia dataa. Slave-laitteen saapuessa Operational-tilaan se kytkee output-datansa fyysisiin liitäntöihin.

Tällöin slave-laite on täysin käyttövalmis ja se kykenee mailbox-kommunikointiin ja prosessidatakommunikointiin. (2.)



KUVA 3. EtherCAT-tilakone

3 KÄYTÖSSÄ OLEVA LAITTEISTO

Tämän työn käytössä oli saatavilla useita tietokoneita sekä tarvittava määrä I/O-laitteita. Tietokoneiksi valikoituivat Aaeonin Up Squared yhden piirilevyn tietokone ja Beckhoffin C6015-0010. I/O-laitteita valittiin sen mukaan, että EtherCAT Mastereita testattaessa voitaisiin testata, miten niillä onnistuu binäärisen I/O:n ohjaus. Täten I/O-korteiksi valikoituivat 8-kanavainen sisääntuloterminaali EL1008 ja 8-kanavainen ulostuloterminaali EL2008. Näiden yhdistämiseen EtherCAT-väylään käytettiin väyläyhdistintä EK1100. Molemmista tietokoneista löytyi kaksi Ethernet-porttia, joten ne voitaisiin tarvittaessa yhdistää sekä EtherCAT-väylään sekä toiseen verkkoon, josta voitaisiin esimerkiksi ottaa etäyhteys ohjauslaitteeseen.

3.1 Aaeon Up Squared

Aaeon Up Squared on neliytimisen suorittimen sisältävä sulautettu yhden piirilevyn tietokone. Se sisältää kaksi Ethernet porttia, yhden HDMI-portin, yhden DP-portin sekä kaksi USB-porttia. Tallennustilana se käyttää eMMC-muistia. Aaeon Up Squaredin käyttämä Ethernet-piiri on Realtek RTL8111G-CG. Sitä on saatavilla useilla erilaisilla muistikonfiguraatioilla sekä myös suorittimen valintaan voi vaikuttaa. (3.)

3.2 Beckhoff C6015-0010

Beckhoffin C6015-0010 on pienikoteloinen ja passiivisesti jäähdytetty teollisuus-pc. Se voidaan kiinnittää käyttöpaikkaansa useilla eri tavoilla. Sen suoritin voi sisältää yksi, kaksi tai neljä ydintä. Sitä on myös saatavilla useilla eri muistikonfiguraatioilla. Beckhoffin C6015-0010 sisältää kaksi Ethernet porttia, kaksi USB-porttia sekä yhden DP-portin. Sen käyttämä Ethernet-piiri on Intel I210. (4.)

3.3 EK1100

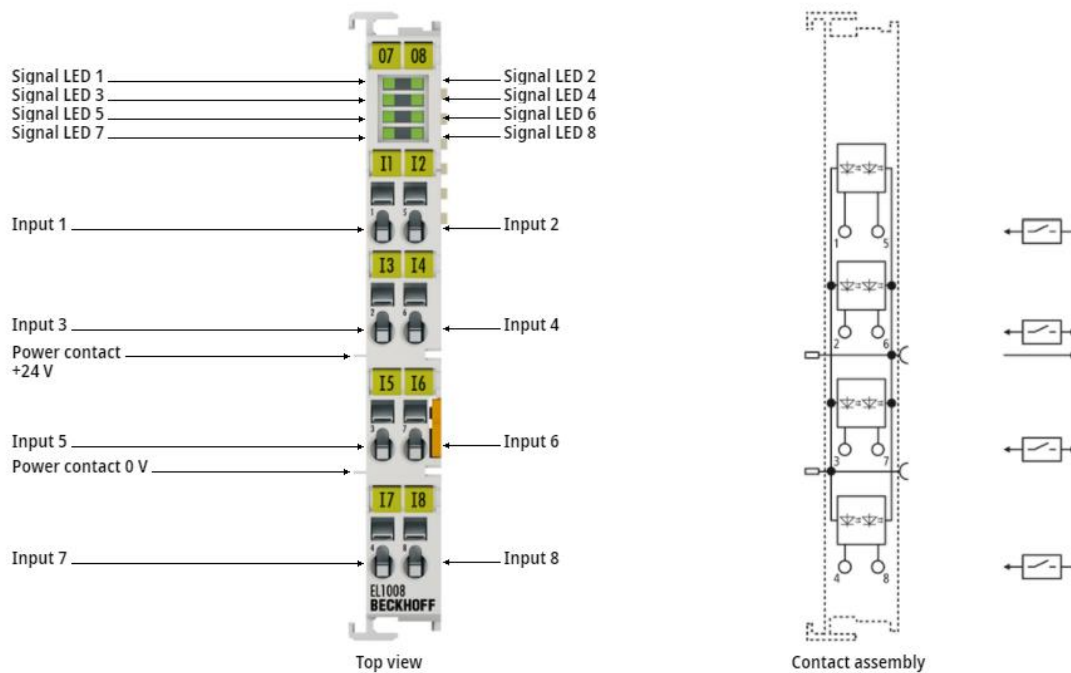
EK1100 on väyläyhdistin, jota käytetään yhdistämään I/O-laitteita EtherCAT-väylään. Se sisältää kaksi Ethernet-porttia, joista toista käytetään sisään tulevalle liikenteelle ja toista ulosmenevälle liikenteelle. Sille ei ole määritetty maksimimäärää, kuinka monta I/O-laitetta siihen voidaan kytkeä

mutta siihen kytketyt I/O-laitteet saavat käyttää enintään 2000 mA:n virran. Sen terminaaleihin tulee kytkeä 24 V:n jännite. (5.)

3.4 EL1008

EL1008 on 8-kanavainen binäärinen sisääntuloterminaali. Se saa jännitesyöttönsä kyljessään olevista kontakteista. Se tulee kytkeä väyläyhdistimeen, joka kytkee sen väylään. Sisään tulevien signaalien jännitetason tulee olla 11 - 30 V. Se sisältää 8 lediä, jotka kertovat kanavien kulloisenkin tilan. (6.)

EL1008:n kontaktit on esitetty kuvassa 4. Kuvasta näkee, että sisääntuloliittimien lisäksi sen kyljessä on jännitesyöttökontaktit, joista terminaali saa käyttöjännitteensä.

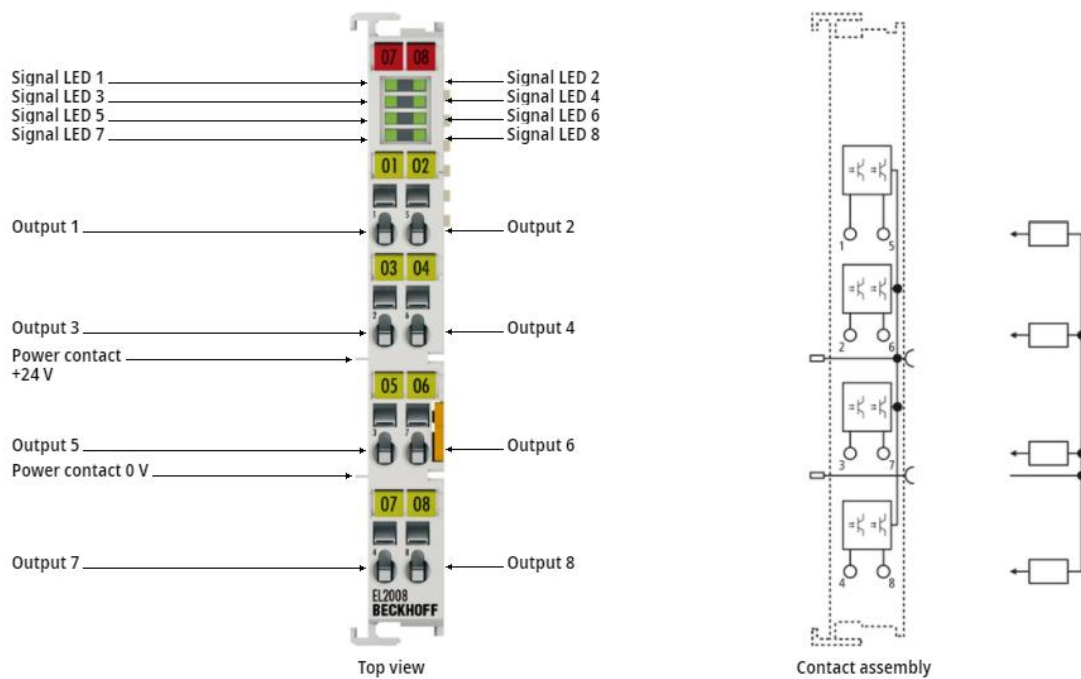


KUVA 4. EL1008

3.5 EL2008

EL2008 on 8-kanavainen binäärinen ulostuloterminaali. Se saa jännitesyöttönsä kyljessään olevista kontakteista. Se tulee kytkeä väyläyhdistimeen, joka kytkee sen väylään. Ulosmenevien signaalien jännitetaso on 24 V. Se sisältää 8 lediä, jotka kertovat kanavien kulloisenkin tilan. (7.)

EL2008:n kontaktit on esitetty kuvassa 5. Kuvasta näkee, että ulostuloliittimien lisäksi sen kyljessä on jännitesyöttökontaktit, joista terminaali saa käyttöjännitteensä.



KUVA 5. EL2008

4 TARKASTELTAVIEN ETHERCAT MASTEREIDEN VALINTA

Tässä työssä otettiin esitarkasteluun useita eri EtherCAT Mastereita. Beckhoff karsittiin pois, koska siitä ei haluttu tässä työssä saada kokemusta. EtherCAT Masterin valinnassa haluttiin painottaa Linux-yhteensopivuutta. Tähän oli syynä erityisesti se, että Linuxia käytettäessä voitaisiin ohjauslaitteen kovalevystä ottaa levykuva, jota voitaisiin käyttää tulevien laitteiden käyttöönotossa hyväksi. Windowsia käyttävillä EtherCAT Mastereilla tämä ei olisi mahdollista Windowsin lisensoinnin takia.

Valinnassa haluttiin myös painottaa sitä, että valittavat EtherCAT Masterit olisivat kaupallisia tuotteita. Ilmaisisissa avoimen lähdekoodin ratkaisuissa tukipalvelut ovat yleensä olemattomia ja järjestelmän kehitystä saatettaisiin kääntää yhtäkkisesti suuntaan, joka ei olisi käyttäjälle toivottavaa. Pahimmassa tapauksessa avoimen lähdekoodin ratkaisua käytettäessä voitaisiin joutua ylläpitämään kopiota lähdekoodista, johon päivitetäisiin vain halutut muutokset. Monissa tapauksissa tämä saattaisi olla haluttukin menetelmä mutta monelle yritykselle se vain tarkoittaisi lisätyötä, jota ei ole järkevää tehdä.

Valinnassa huomioitiin myös kehitysympäristön olemassaolo ja PLC-tuotteen olemassaolo. Monien vaihtoehtojen kohdalla näitä kumpaakaan ei ollut, mikä tarkoittaisi sitä, että ne jouduttaisiin kehittämään itse. Tämä taas veisi huomattavan paljon kehitysresursseja.

Lisäksi huomionarvoista on se, miten helposti ladatut sovellukset ovat päivitettävissä. Erityisesti huomioidaan se, tarvittaisiinko ladattujen sovellusten päivittämiseen kehitysympäristöä vai voitaisiinko se hoitaa vain kopioimalla uudet binääritiedostot ohjauslaitteelle.

Huomionarvoista olisi myös se, tukeeko tarkasteltava EtherCAT Master turvaominaisuuksia. Tässä tapauksessa tarkastellaan, tukeeko EtherCAT Master EtherCAT:in laajennusta nimeltä FsoE. Tapauksessa, jossa se ei tukisi, ei sen avulla voitaisi toteuttaa projekteja, joissa tarvittaisiin turvaominaisuuksia.

Taulukkoon 1 on listattu EtherCAT Mastereita, joista tarkasteluun päätyneet toteutukset valittiin. EtherCAT Mastereista on listattu oleellimmat tiedot eli niiden tukemat alustat ja se, miten niiden

ohjelmointi tulisi hoitaa. Myös järjestelmän kaupallisuus taikka mahdollinen avoimen lähdekoodin lisenssi on merkitty.

TAULUKKO 1. Esivalittuja EtherCAT Mastereita

Tuote	Käyttöjärjestelmä	Ohjelmointi	Kaupallisuus
Acontis EC-Master	Linux/Windows	C API, C#, Python	Kaupallinen
KINGSTAR	Windows	IEC 61131-3 yhteensopiva ohjelmointiympäristö, PLCopen Labview C/C++	Kaupallinen
CODESYS	Linux / Windows	IEC 61131-3 yhteensopiva ohjelmointiympäristö	Kaupallinen
Etherlab ja IgH master	Linux / Windows	C API/Simulink	GPLv2/GPLv3
Beckhoff	Windows	IEC 61131-3 yhteensopiva ohjelmointiympäristö, PLCopen	Kaupallinen
SOEM	Linux / Windows	C API	GPLv2

4.1 Valitut EtherCAT Masterit

Koska Beckhoffin lisäksi esivalittujen EtherCAT Mastereiden listalla oli vain kolme kaupallista toteutusta, päätettiin kaikki ne kolme valita tarkastelun alle. Osassa niissä oli piirteitä, jotka eivät tyydyttäneet halutun EtherCAT Masterin kriteerejä, mutta kolmea kappaletta pidettiin työhön sopivana määränä. Valituiksi siis päätyivät Acontis EC-Master, KINGSTAR ja CODESYS.

4.2 Lisenssiehdot

Ohjelmistotuotteen valinnassa on myös hyvä ottaa huomioon lisenssiasiat. Kaikissa ohjelmistotuotteissa lisenssiehdot määrittävät mihin ja miten tuotetta saa käyttää. Kaupallisissa tuotteissa lisenssiehdot saattavat olla mitä tahansa ja ne eivät välttämättä noudata mitään yleisiä lisenssiehtoja. Avoimen lähdekoodin tuotteissa taas on muutamia yleisesti käytössä olevia lisenssejä, joista kehittäjä valitsee tuotteelleen sopivimman.

4.2.1 GPL

GPL-lisenssi on yleisin käytössä oleva avoimen lähdekoodin lisenssi. Yksi tärkeimmistä ehdoista on se, että kaikki ohjelmistotuotteet, jotka käyttävät GPL-lisenssoitua ohjelmakoodia, ovat velvoitettuja julkaisemaan ohjelmansa lähdekoodin. GPL sallii ohjelmakoodin vapaan käytön, muokkauksen ja uudelleenlevityksen. Ehtona on vain se, että lähdekoodi tulee aina julkaista.

4.2.2 LGPL

LGPL-lisenssi vastaa muuten GPL-lisenssiä mutta LGPL-lisenssi sallii ohjelmakoodin dynaamisen linkittämisen muilla lisensseillä lisensoidun lähdekoodin kanssa. Tämä tarkoittaa sitä, että ohjelmistoprojektin, joka käyttää LGPL-lisenssoitua ohjelmakoodia, ei välttämättä tarvitse julkaista kaikkea ohjelmakoodiaan.

4.2.3 BSD

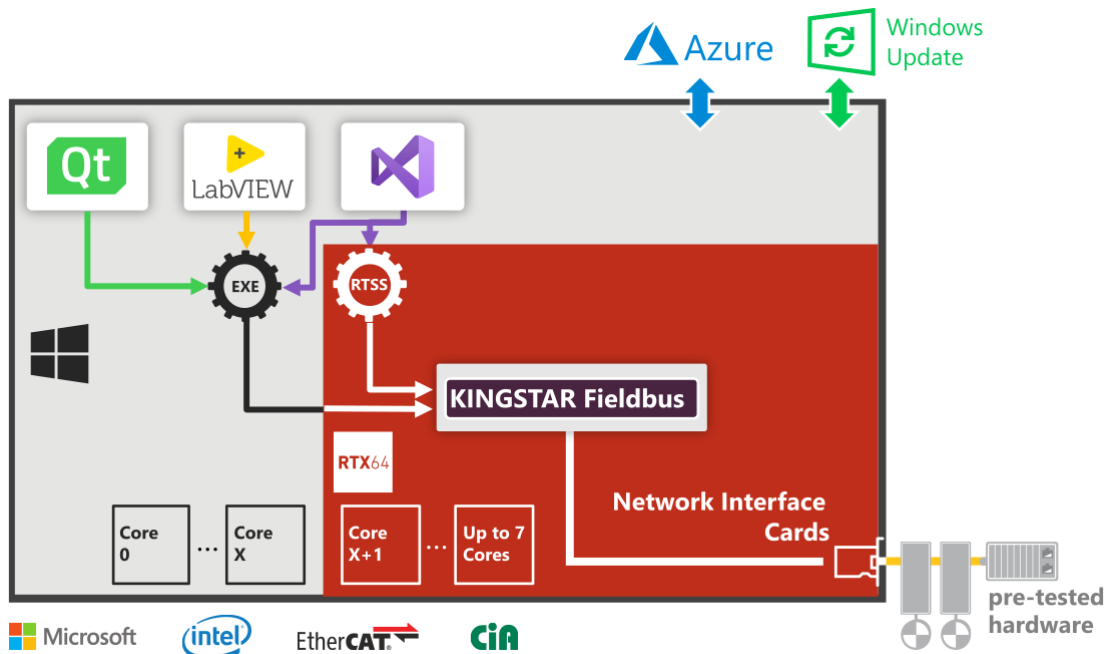
BSD-lisenssi sallii sen alla lisensoidulle ohjelmakoodille tehtävän aivan mitä tahansa, kunhan käytettyyn ohjelmakoodiin liitetyt lisenssiehdot säilytetään. Tämän ansiosta BSD-lisenssoitua ohjelmakoodia voidaan käyttää, muokata ja uudelleen levittää ilman että lähdekoodia tarvitsee julkaista.

4.2.4 MIT

MIT-lisenssi vastaa suurimmilta osin BSD-lisenssiä. Se eroaa BSD-lisenssistä lähinnä siten, että se explisiittisesti sallii tiettyjä asioita. Nämä asiat ovat kuitenkin sallittuja myös BSD-lisensoidun ohjelmakoodin kanssa.

5 KINGSTAR ETHERCAT MASTER

KINGSTAR on IntervalZeron toteuttama EtherCAT Master. KINGSTARin alustana on tarkoitus käyttää Windows 10 -käyttöjärjestelmää. EtherCAT Masterin lisäksi se myös sisältää IntervalZeron reaaliaika-Windows-toteutuksen, SoftPLC-tuotteen, konfiguraatio- ja diagnostiikkatyökaluja ollen näin täysi paketti EtherCAT-väylää käyttävien laitteiden ohjaukseen. KINGSTARin arkkitehtuuri on esitetty kuvassa 6.



KUVA 6. KINGSTARin arkkitehtuuri

KINGSTARin mukana tuleva IntervalZero RTX64 on Windowsin reaaliaikalaajennus, jonka avulla tietokone kykenee toimimaan reaaliaikaisemmin. Tämä on I/O-laitteiden ohjauksessa erityisen tärkeää.

KINGSTARin EtherCAT Master tukee seuraavia EtherCAT-väylän ominaisuuksia (8, 2 - 6):

- Process Data Exchange
- Network Configuration
- Mailbox Support
- CAN application layer over EtherCAT (CoE)
- Ethernet over EtherCAT (EoE)
- File access over EtherCAT (FoE)
- Servodrive-Profile over EtherCAT (SoE)
- ADS over EtherCAT (AoE)
- Vendor over EtherCAT (VoE)
- Synchronization with Distributed Clock (DC)
- Slave-to-Slave Communication
- Hot Connect.

Edellä olevasta listasta kuitenkin huomataan, että FsoE eli Safety over EtherCAT puuttuu kyseiseltä listalta. Näin ollen KINGSTARin EtherCAT Masteria ei voida käyttää sovelluksiin, joissa tarvitaan turva-automaatiota. Tämä on valitettava puute, jonka takia KINGSTAR mahdollisesti karsiutuisi vaihtoehtojen joukosta monen projektin osalta.

KINGSTARin mukana tulee SoftPLC-tuote nimeltään LogicLab. LogicLab on IEC61131-3 yhteensopiva ja se tukeekin kaikkia kyseisen standardin määrittämiä ohjelmointikieliä. LogicLabissa on laaja määrä erilaisia ominaisuuksia, kuten että sen koodin suoritusta voidaan seurata reaaliaikaisesti, siinä voidaan asettaa breakpointteja ja siinä voidaan seurata useiden muuttujien tilaa graafisesti. LogicLabin mukana tulevat myös PLCopenin standardisoimat ja määrittelemät liikeohjauksien ohjelmointiin käytettävät funktioblokit. LogicLabissa voidaan myös ohjelmoidun sovelluksen lähdekoodit tallentaa salattuna ohjauslaitteelle. (9.)

KINGSTAR sisältää myös konfiguraatiotyökalun, jonka avulla voidaan konfiguroida väylään liitetyjä I/O-kortteja. Konfiguraatiotyökalun avulla voidaan myös konfiguroida EtherCAT Masteriin liittyviä parametreja ja sen avulla voidaan ottaa vaihtoehtoisia ominaisuuksia käyttöön kuten Hot Connect. Sen kautta voidaan myös määrittää EtherCAT Masterin sykli aika. Myös servo-ohjaimien kaikki parametrit ovat säädettävissä kyseisen konfiguraatiotyökalun avulla.

KINGSTARin alustana on Windows 10 ja se asettaa tiettyjä vaatimuksia tietokoneelle, johon KINGSTAR asennetaan. Käytössä tulee olla moniydinprosessori ja asennettavan laitteen Ethernet-piiriin tulee olla yhteensopiva KINGSTARin kanssa. Taulukossa 2 on esitetty KINGSTARin kanssa yhteensopivat Ethernet-piirit (10, 1 - 2).

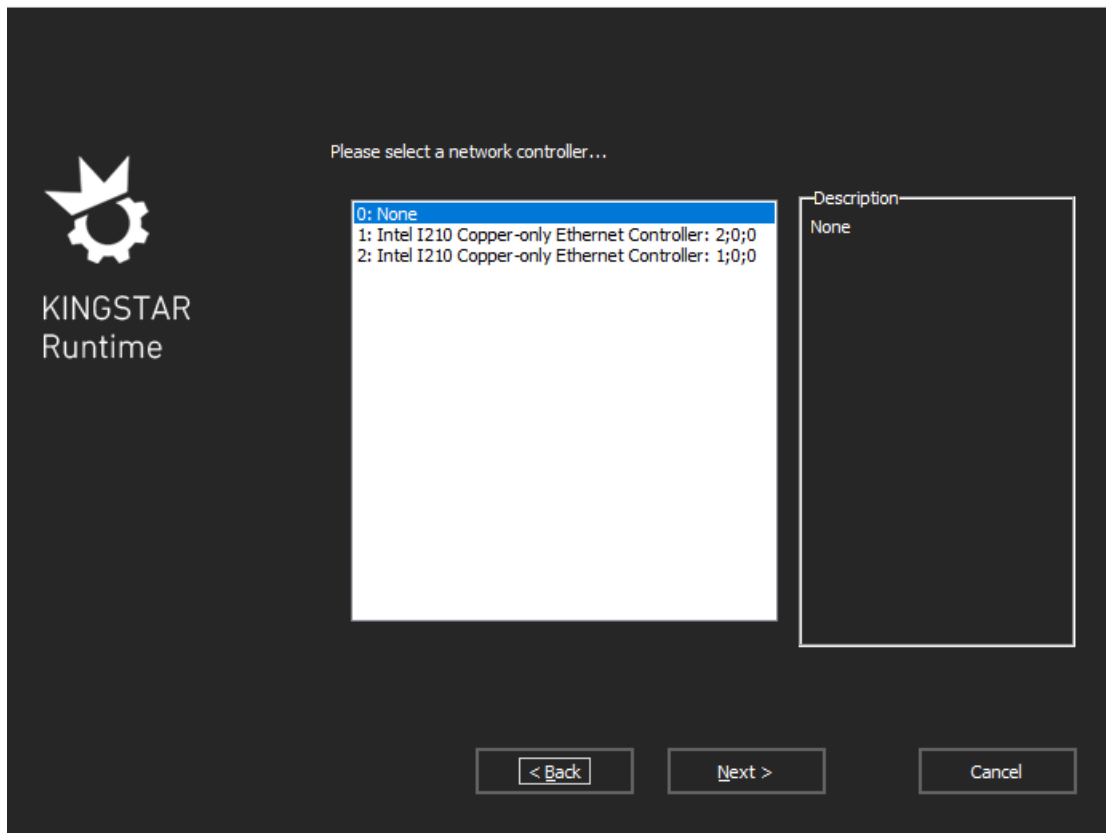
TAULUKKO 2. KINGSTARin kanssa yhteensopivat Ethernet-piirit

Vendor ID	Device ID	Kuvaus	KINGSTAR:in vähimmäisversio
0x8086	0x1533	Intel® I210 T1 Copper-only Ethernet Controller	1.0
0x8086	0x1521	Intel® I350 Quad/Dual PCIe Copper Ethernet Controller	1.0
0x8086	0x1523	Intel® I350 Quad/Dual PCIe 1000BASE-KX/BX Ethernet Controller	1.0
0x8086	0x1539	Intel® I211-AT Ethernet Controller	2.0
0x8086	0x157B	Intel® I210 Flash-less Copper-only Ethernet Controller	3.0
0x8086	0x10D3	Intel® 82574L Gigabit Ethernet Controller	1.0
0x8086	0x10F6	Intel 82574LA Ethernet Controller	3.1.2

Tässä työssä testaukseen valittiin laitteistoksi Beckhoffin C6015-0010 tietokone, jonka Ethernet-piiri on Intel I210. Näin ollen kyseinen tietokone on KINGSTARin kanssa yhteensopiva.

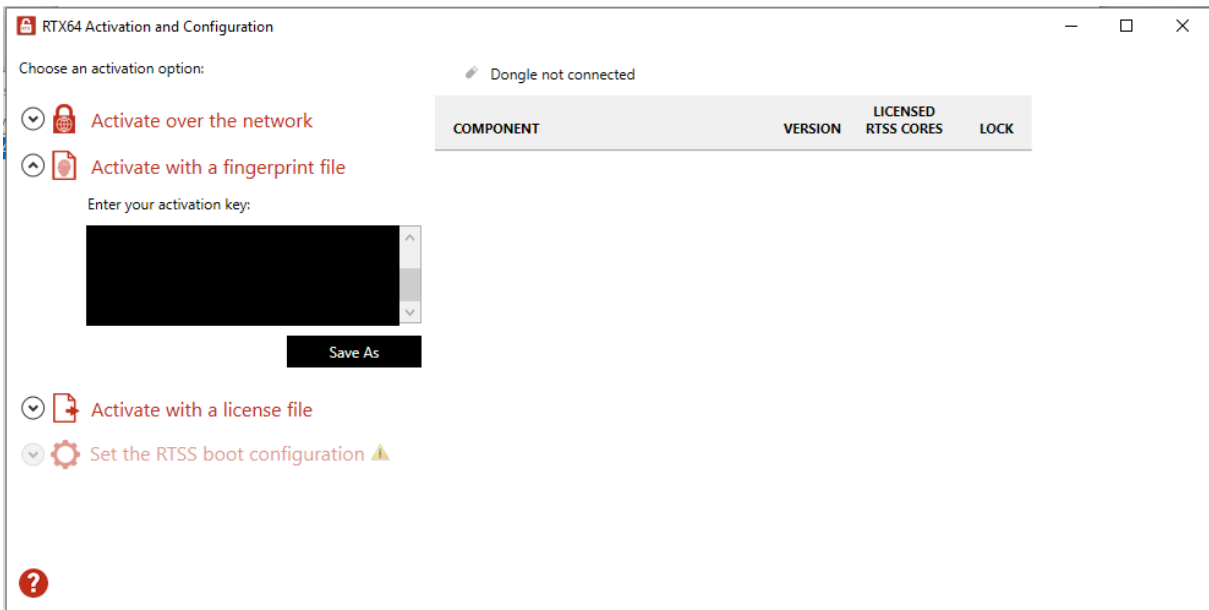
5.1 KINGSTARin asennus

KINGSTARin asennus aloitetaan siirtämällä KINGSTARin runtimen asennuspaketti tietokoneelle, jota halutaan käyttää ohjauslaitteena. Kyseisen asennuspaketin suoritus on kuten normaalin Windowsiin asennettavan ohjelman asennuspaketin suoritus, paitsi että asennuspaketti kysyy, mitä Ethernet-piiriä käytetään EtherCAT Masteria varten. Tämä valinta on esitetty kuvassa 7. Tässä tapauksessa valikosta valitaan kohta 1. Runtimen asennuksen jälkeen tuli asentaa LogicLab erillisen asennuspaketin avulla. LogicLabin asennuksesta ei jäänyt mitään mainittavaa.



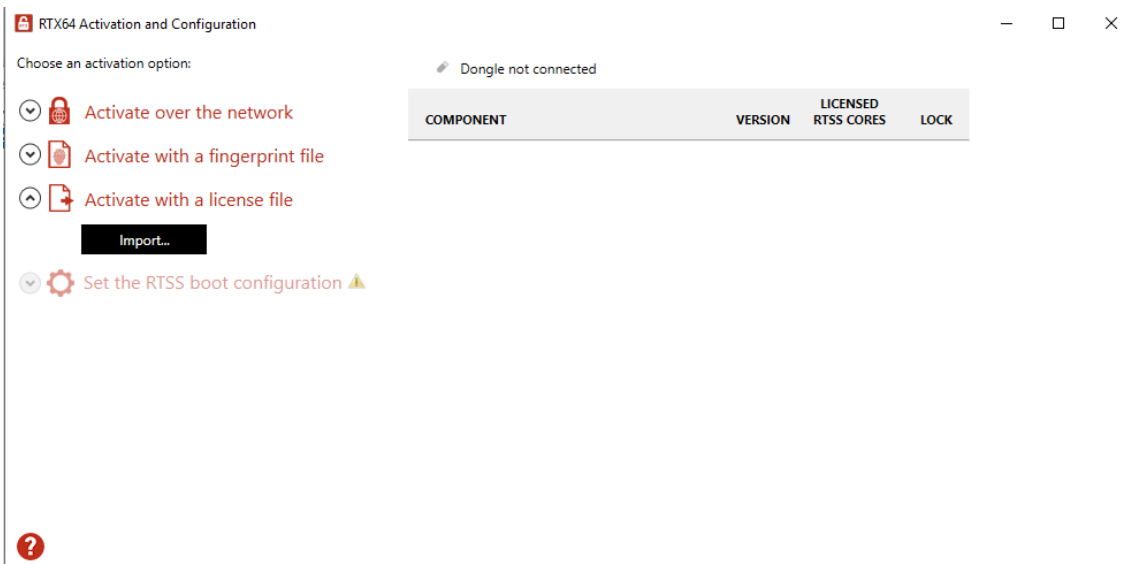
KUVA 7. EtherCAT masterina käytettävän Ethernet-piirin valinta

Tuotteiden asennuksen jälkeen pitää lisensoida asennetut tuotteet. Tämä on mahdollista, vaikka käytettävässä laitteessa ei ole internet-yhteyttä käytettävissä. Saatu lisenssikoodi syötetään kuvassa 8 näkyvään laatikkoon ja luodaan lisensoinninpyyntötiedosto painamalla Save As -nappia.

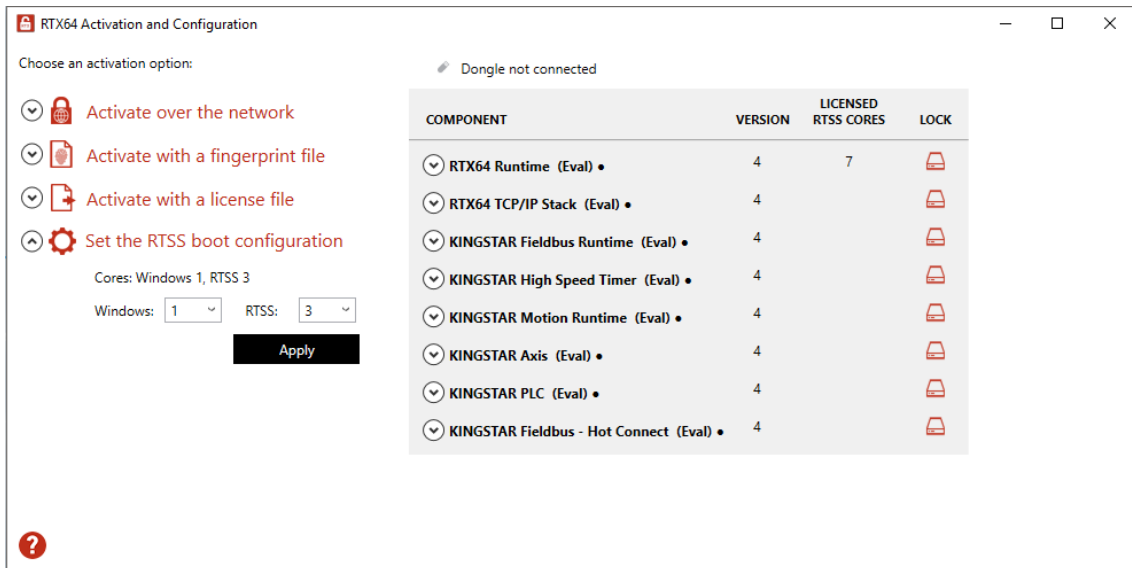


KUVA 8. Lisenssiavaimen syöttö

Lisensoinninpyyntötiedoston luomisen jälkeen tulee mennä verkkosivulle <http://activation.intervalzero.com/> ja syöttää saatu lisensoinninpyyntötiedosto kyseiselle sivulle. Tämän tekeminen antaa ladata sivustolta lisenssitiedoston, jonka voi siirtää ohjauslaitteena käytettävälle laitteelle, jolloin lisenssitiedosto voidaan syöttää kuvan 9 näyttämässä näkymässä painamalla Import-nappia. Tämän jälkeen ohjelma näyttää käytössä olevat lisenssit ja pyytää asettamaan käynnistyskonfiguraation painamalla kuvassa 10 näkyvää Apply-nappia.

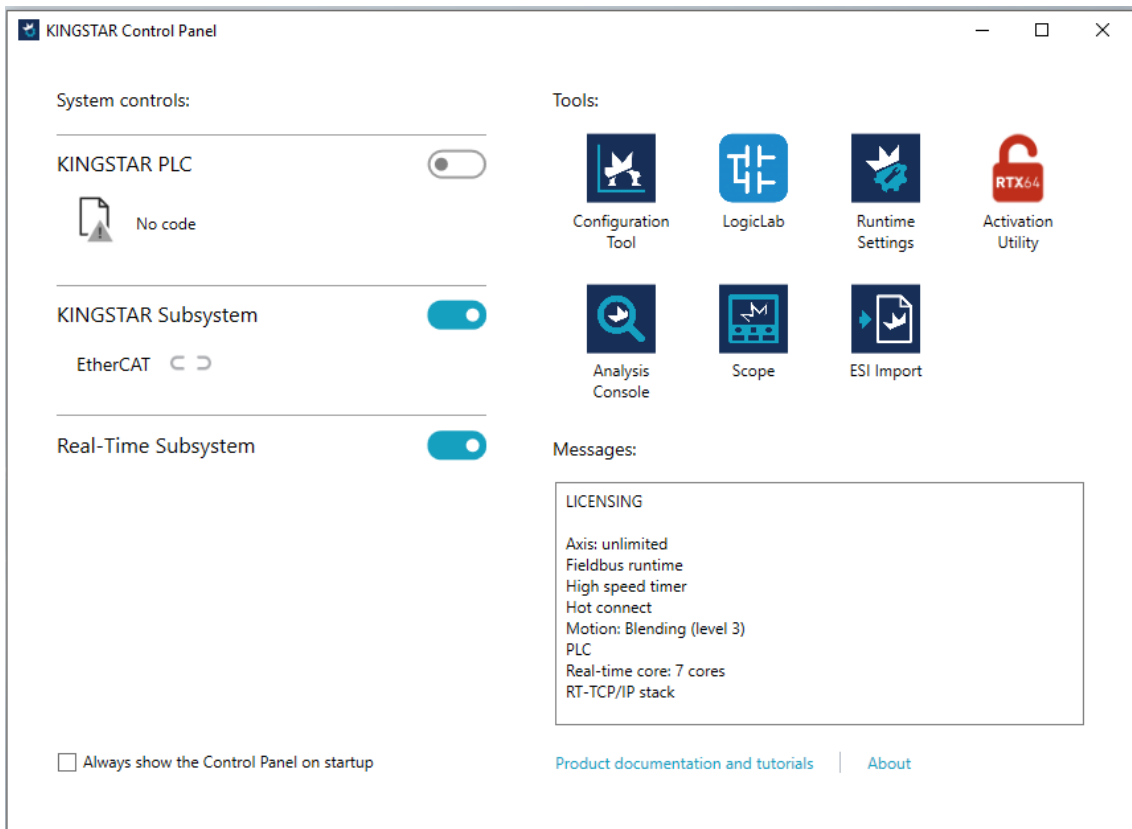


KUVA 9. Lisenssitiedoston syöttäminen



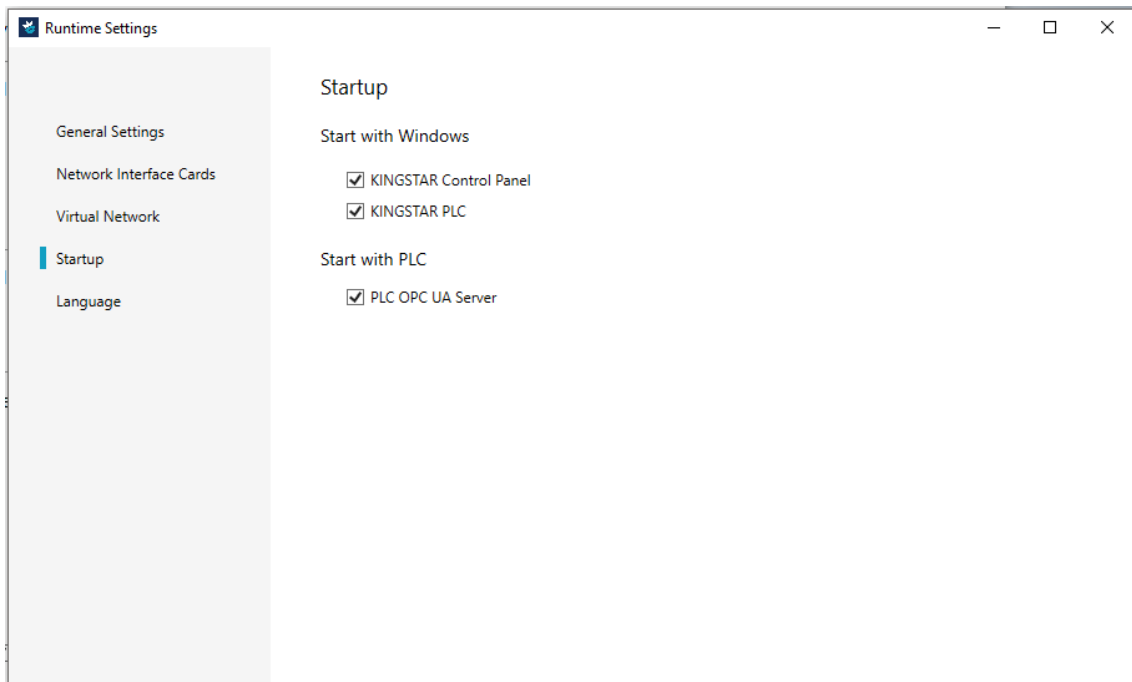
KUVA 10. Käynnistyskonfiguraation asettaminen

Tämän jälkeen järjestelmä käynnistetään uudelleen ja käynnistyessään se lataa IntervalZero RTX64:n avulla muokatun version Windows 10 -käyttöjärjestelmästä. Käynnistymisen jälkeen aukeaa kuvassa 11 näkyvä konfiguraatiopaneeli, josta voidaan käynnistää ja sulkea järjestelmän eri osia ja jossa on myös pikakuvakkeet saatavilla oleviin työkaluihin.



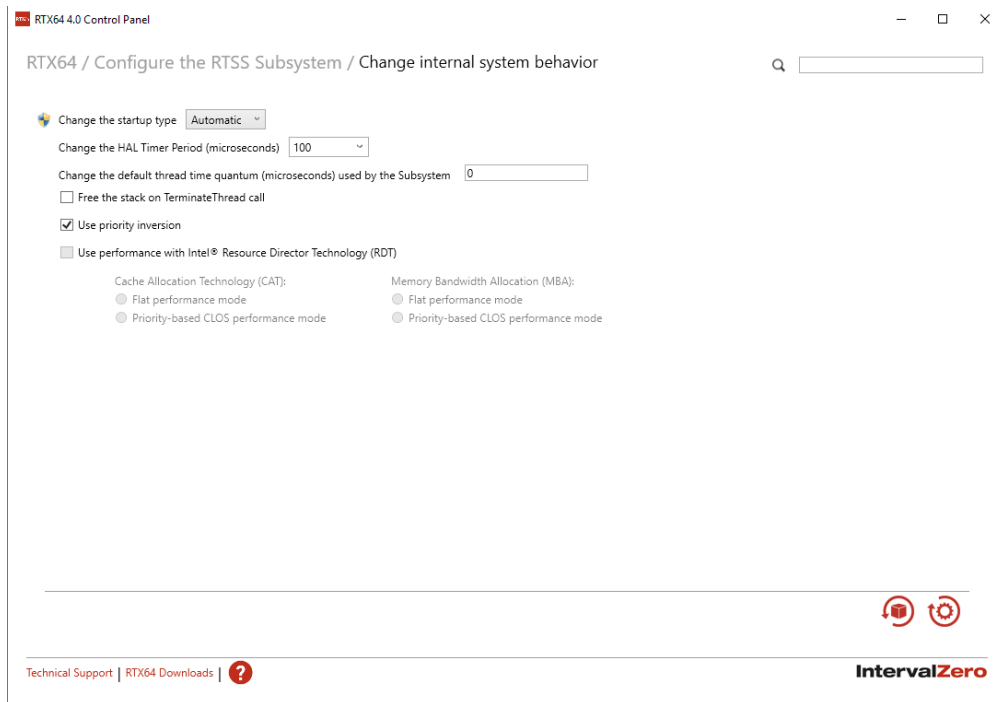
KUVA 11. Konfiguraatiopaneeli

Seuraavana oli aika asettaa asetukset niin, että KINGSTARin sisältämät palvelut käynnistyvät automaattisesti tietokoneen käynnistyessä. Ensimmäisenä tuli avata Runtime Settings KINGSTAR Control -paneelista ja siirtyä siinä välilehdelle Startup. Kyseinen välilehti on kuvattu kuvassa 12. Kyseisessä välilehdessä valittiin, että kaikki valittavissa olevat palvelut käynnistetään tietokoneen käynnistyessä.



KUVA 12. PLC:n automaattisen käynnistyksen valinta

Seuraavana tuli avata RTX64 Control -paneeli ja valita sieltä, että reaaliaikainen sivujärjestelmä käynnistyy automaattisesti käynnistyksen yhteydessä. Tämä valinta löytyy polusta RTX64 / Configure the RTSS Subsystem / Change internal system behavior. Kyseisessä polussa tuli valita, että RTSS Subsystem käynnistyy automaattisesti. Tämä valinta on kuvattu kuvassa 13.



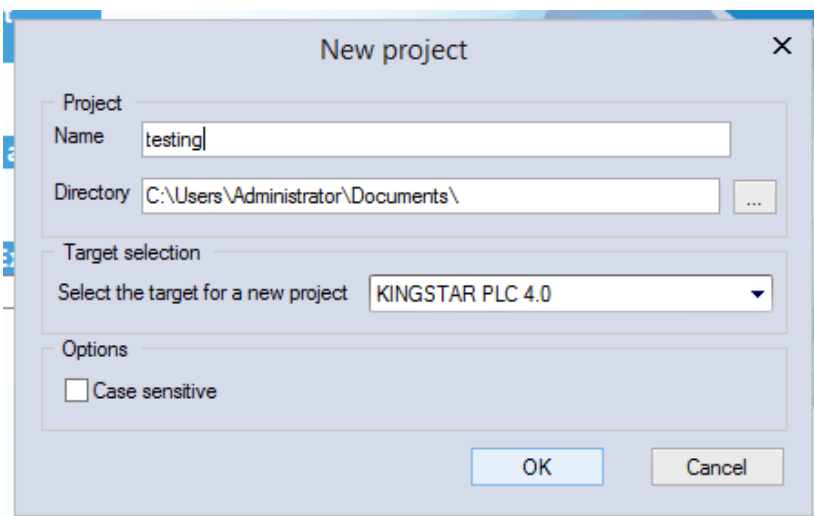
KUVA 13. RTSS Subsystemin automaattinen käynnistys

5.2 KINGSTARin käyttö

KINGSTARilla ohjelmointi aloitetaan avaamalla KINGSTAR Control paneelista LogicLab. Sen auetessa aukeaa kuvassa 14 näkyvä näkymä. Ensimmäisenä tuli luoda uusi projekti painamalla New project -nappia, jolloin aukeaa kuvassa 15 näkyvä ponnahdusikkuna, jossa voidaan valita projektin tallennuskansio.

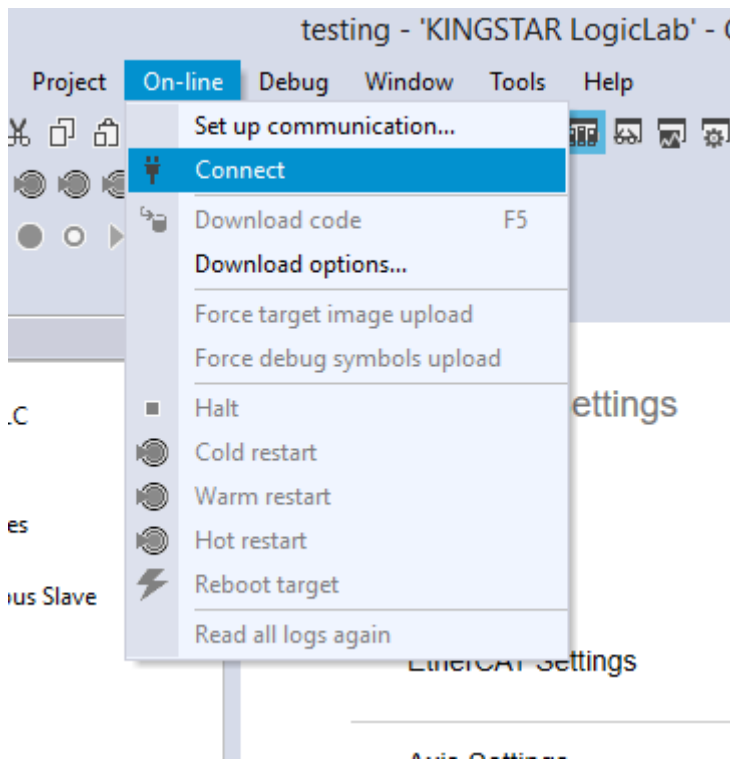


KUVA 14. LogicLab

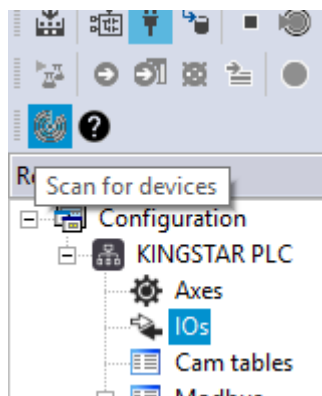


KUVA 15. Projektin tallennuskansion sijainti

Seuraavana tuli yhdistää LogicLab järjestelmään. Tämä hoituu painamalla kuvassa 16 näkyvää nappia ohjelman yläpalkista. Tämän jälkeen voidaan hakea väylästä väylässä olevat laitteet ja luoda niitä varten muuttujat PLC-projektiin. Laitteiden haku tapahtuu painamalla kuvassa 17 näkyvää nappia.

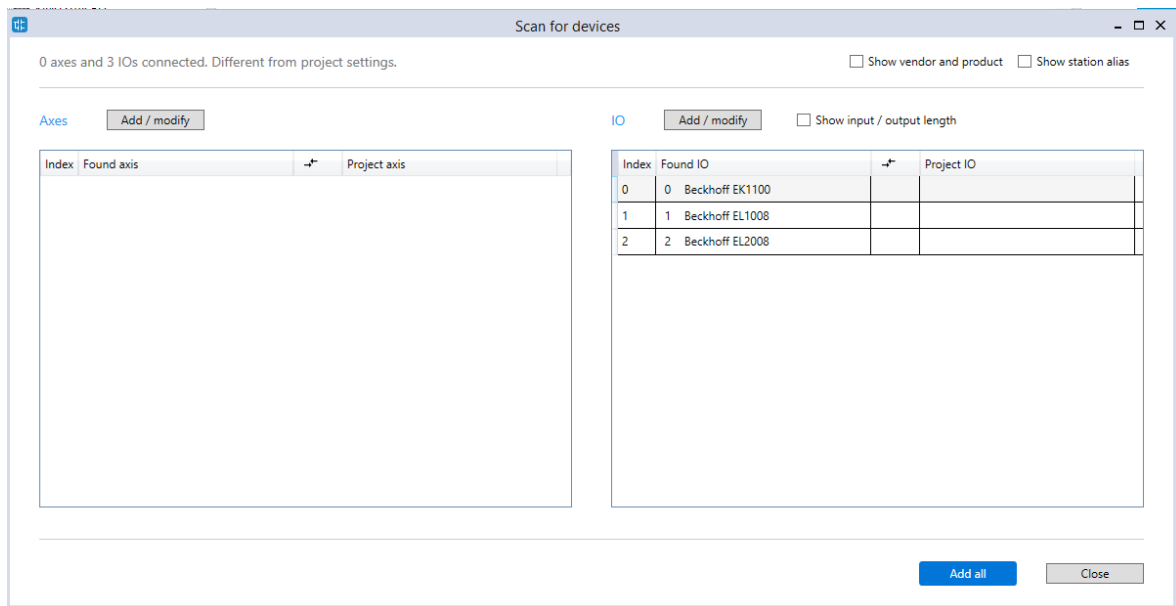


KUVA 16. Logiclabin yhdistäminen järjestelmään



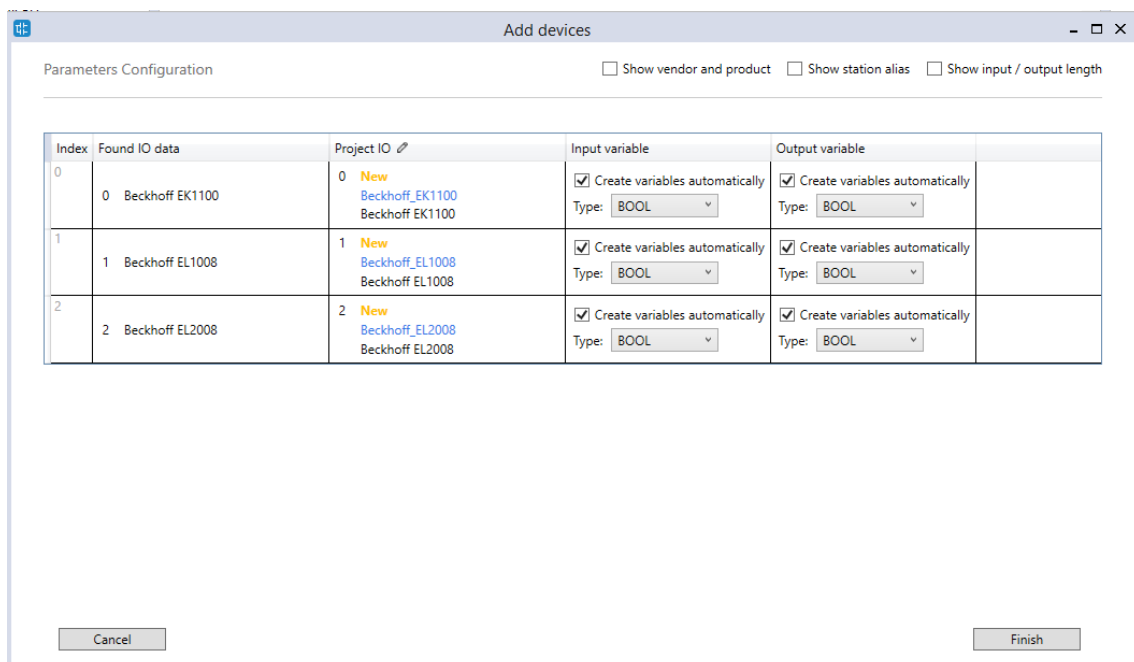
KUVA 17. Laitteiden haku väylästä

Scan for devices -napin painalluksen jälkeen aukeaa ikkuna, jossa näkyy mitä I/O-laitteita väylästä on löytynyt. Tämä ikkuna näkyy kuvassa 18. Tässä tapauksessa väylästä löytyi suoraan se laitteisto, joka oli käytössä, joten voitiin painaa Add all -nappia.



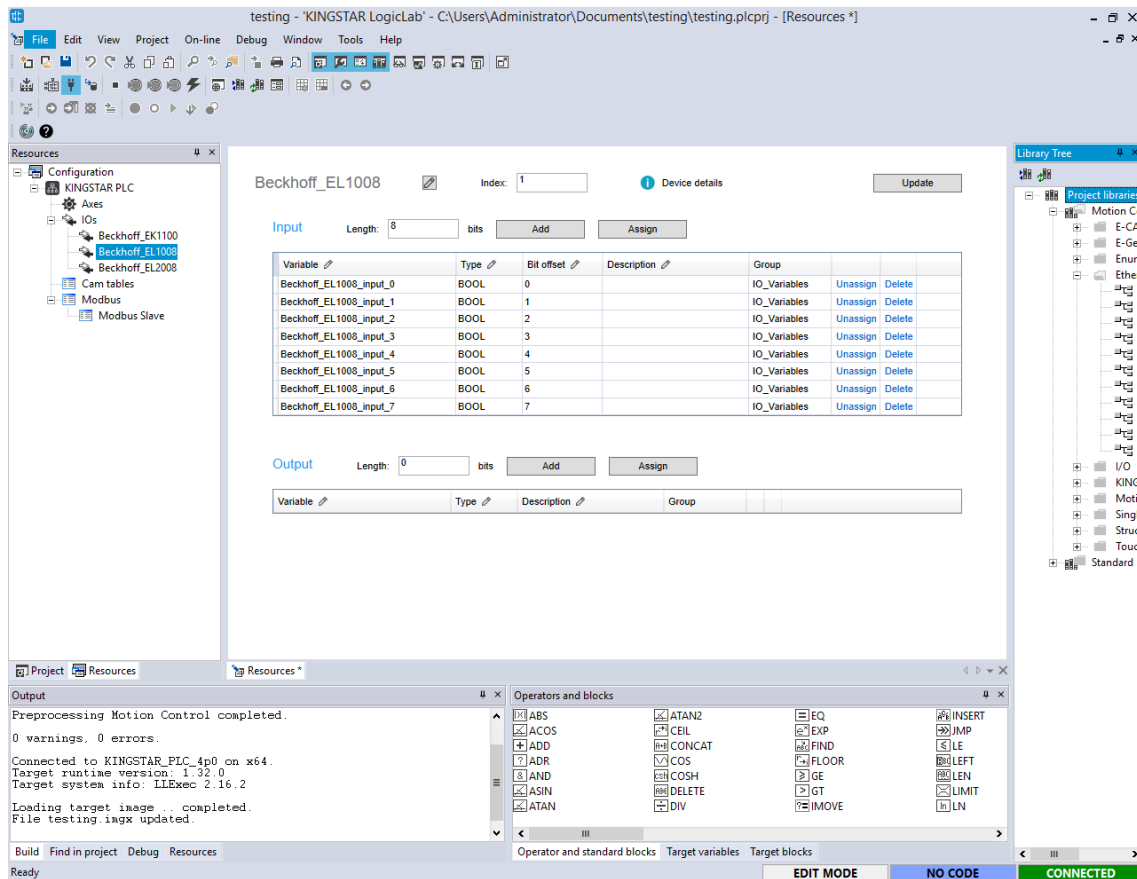
KUVA 18. Väylästä löytyneet laitteet

Seuraavana aukeaa Add devices -niminen ikkuna, jossa voidaan valita, luodaanko väylästä löytyneitä laitteita varten automaattisesti muuttujat. Tämä ikkuna näkyy kuvassa 19. Tässä tapauksessa muuttujien automaattinen luominen oli juuri sitä mitä haluttiin, joten voitiin painaa Finish-nappia.



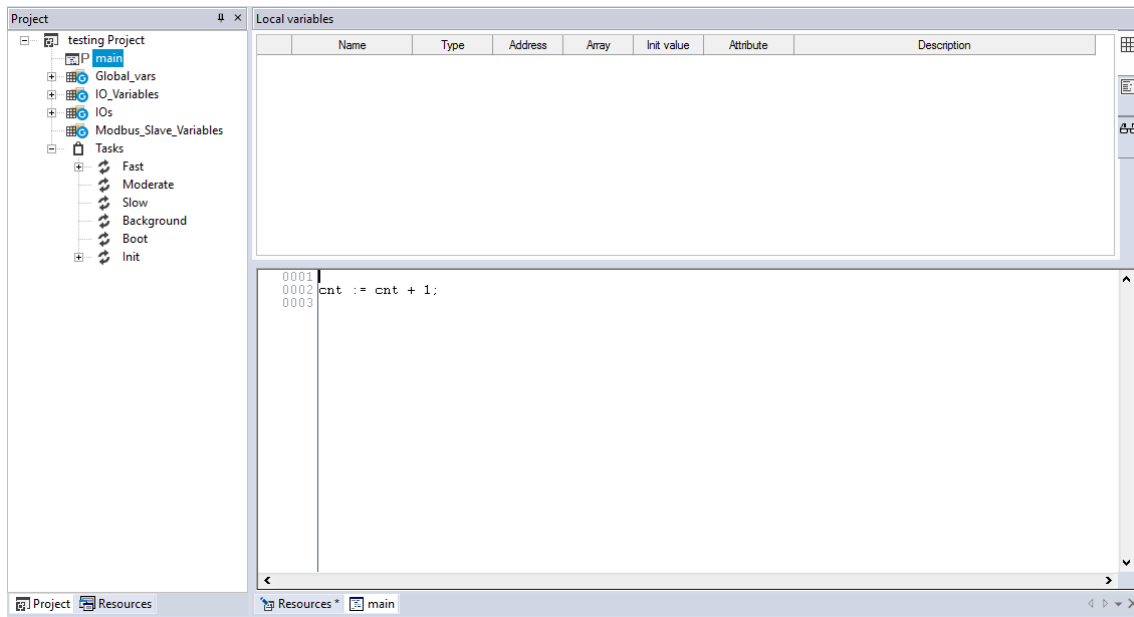
KUVA 19. Muuttujien automaattinen luonti

Tämän jälkeen voitiin tarkastella ruudussa vasemmalla olevan puurakenteen IOs-osiota ja huomattiin, että halutut muuttujat todellakin olivat ilmestyneet automaattisesti projektiin. Kyseiset muuttujat olivat jaoteltuina korttikohtaisesti IOs-puurakenteen alapuolelle. Niille oli myös automaattisesti annettu nimet. Kyseinen näkymä on kuvattu kuvassa 20.



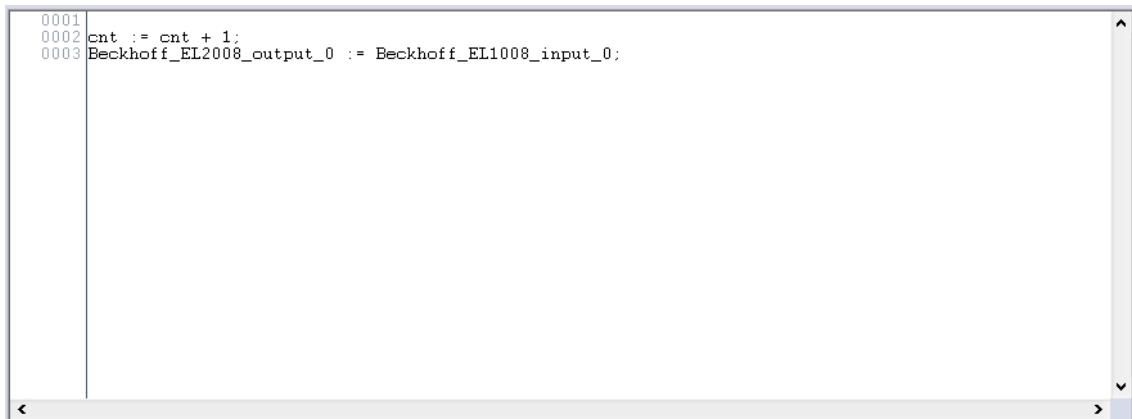
KUVA 20. Automaattisesti lisätyt muuttujat

Seuraavana oli aika kokeilla ohjelmointia LogicLabin kautta, joten siirryttiin Resources-välilehdeltä Project-välilehdelle. Siellä olivat kaikki projektin lähdekooditiedostot listattuna. Uuden projektin luonnissa sinne oli myös lisätty main-niminen lähdekooditiedosto, joka on projektin ylätason ohjelma. Project-välilehti sekä automaattisesti luotu main-ohjelma näkyvät kuvassa 21.



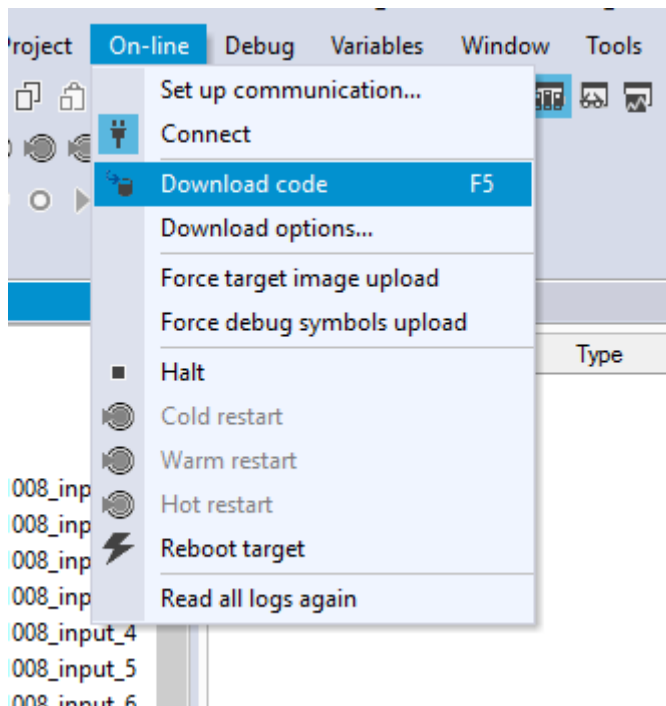
KUVA 21. Automaattisesti luotu main-ohjelma

Automaattisesti luotua main-ohjelmaa päätettiin muokata. Tässä työssä haluttiin kokeilla binäärisen I/O:n lukua ja kirjoitusta käyttäen saatavilla olevia I/O-kortteja EL1008 ja EL2008. Yksinkertainen ohjelma, jossa luetaan sisään tulevaa dataa ja kirjoitetaan ulosmenevää dataa, oli sellainen ohjelma, jossa sisään tuleva data kirjoitetaan suoraan ulostulokortille. Tällainen ohjelma onnistui kirjoittamalla yksi rivi koodia. Main-ohjelmaan tehdyt muokkaukset on kuvattu kuvassa 22.



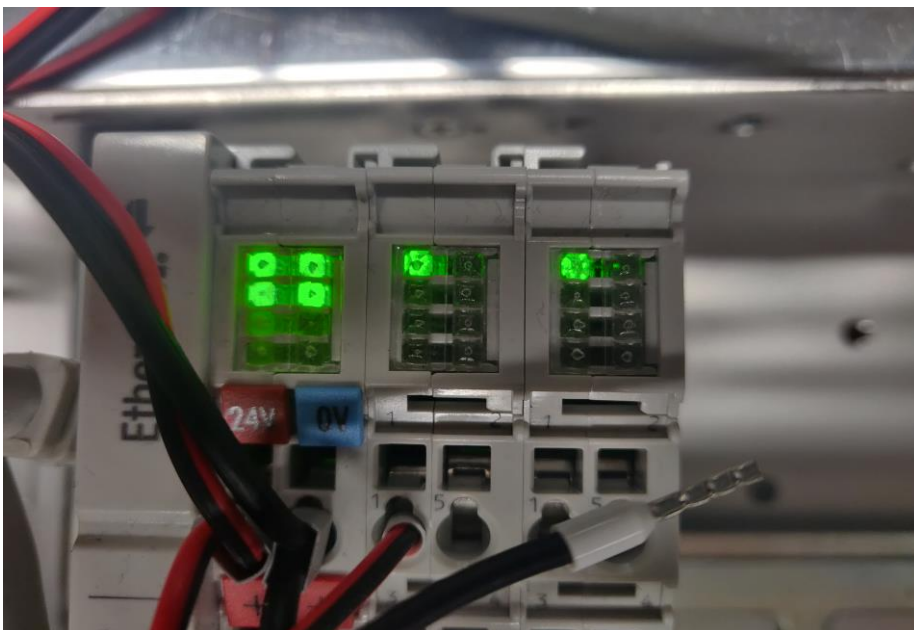
KUVA 22. Sisääntulon kirjoitus ulostuloon

Tämän jälkeen kirjoitettu ohjelma tulisi ladata PLC:lle. Tämä hoituu painamalla Download code -nappia yläpalkin On-line-valikosta. Kyseinen nappi on esitetty kuvassa 23.



KUVA 23. Ohjelman lataus PLC:lle

Kyseisen napin painalluksen jälkeen ohjelma latautui PLC:lle ja sitä voitiin alkaa testaamaan. Ohjauslaitteistoon kytketyille EL1008-kortille kytkettiin ensimmäiseen kanavaan 24 V:n jännite, jolloin kyseisen kanavan tilaa kuvaava LED syttyi. Myös kortilla EL2008 oleva ensimmäisen kanavan tilaa kuvaava LED syttyi, joten ohjelma onnistuneesti kopioi EL1008:n ensimmäisen kanavan tilan EL2008:n ensimmäiseen kanavaan. LED:ien tilat ovat kuvattuna kuvassa 24.



KUVA 24. Ohjelman onnistunut suoritus

Tämän jälkeen huomattiin, että EtherCAT-yhteys ei vielä lähtenyt automaattisesti käyntiin, kun laitteisto käynnistettiin. Selvitystyön jälkeen huomattiin, että se tulisi hoitaa PLC:lle ladattavan ohjelman kautta. Esimerkkiohjelmasta saatiin tarkasteltua, mihin muuttujiin tulisi kirjoittaa, että tämä tapahtuisi. Muokkaukset tehtiin kirjoitettuun ohjelmaan ja kyseiset muokkaukset havaittiin toimiviksi. Muokkaukset ovat esitetty kuvassa 25.

```
0001 IF first = TRUE THEN
0002     ksStart := TRUE;
0003 END_IF;
0004 IF ksInitialized = TRUE THEN
0005     first := FALSE;
0006     ksStart := FALSE;
0007 END_IF;
0008
0009 cnt := cnt + 1;
0010 Beckhoff_EL2008_output_0 := Beckhoff_EL1008_input_0;
0011
```

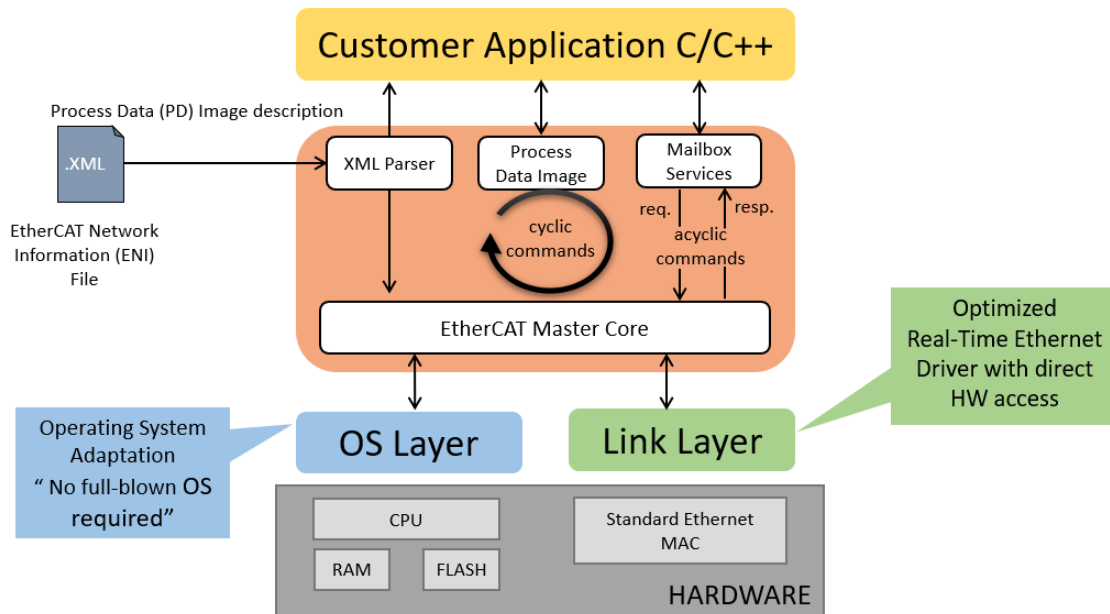
KUVA 25. EtherCAT-yhteyden automaattinen käynnistys

5.3 Yhteenveto KINGSTARista

KINGSTARia käyttäessä huomattiin, että sen asennus ja käyttö on hyvinkin suoraviivaista. Valitettavasti tämän työn aikana ei testattu servo-ohjaimien käyttöä mutta sitä mahdollisuutta kuitenkin tutkittiin ja sen huomattiin olevan hyvinkin helppoa. Erityisesti PLCopen-liikeohjauskirjastojen läsnäolo huomattiin erityisen hyväksi ominaisuudeksi. Positiivisena voidaan myös pitää sitä, että laitteiston lisenssin aktivointi voidaan hoitaa ilman että ohjauslaitteessa on internet-yhteyttä. KINGSTARista löytyi suurin osa tarvittavista ominaisuuksista ja sen hyväksi todetun ohjelmointiympäristön olemassaolon takia sitä käyttämällä kehitystyötä voidaan kuvitella mukavaksi prosessiksi. Huonona puolena KINGSTAR ei tue turva-automaatiota, joka karsii KINGSTARin käytön kaikkien niiden projektien osalta, joissa on vaatimuksia turva-automaation osalta. Valitettavasti KINGSTAR toimii ainoastaan Windows-ympäristössä, joten valmisteltua ohjauslaitteen levykuvaa ei voida kopioida ilman Windowsin lisenssiin liittyviä ongelmia.

6 ACONTIS ETHERCAT MASTER

EC-Master on Acontis technologiesin kehittämä EtherCAT Master. Kyseinen tuote ei sisällä mitään muuta kuin EtherCAT Masterin ja Ethernet-ajurin, siihen ei siis kuulu minkäänlaista PLC-tuotetta. EC-Masterin arkkitehtuuri on kuvattu kuvassa 26.



KUVA 26. EC-Masterin arkkitehtuuri

EC-Master EtherCAT Masteria käytettäessä sovelluskehitys tapahtuu C tai C++ -kieltä käyttäen. Myös C#:lla ja Pythonilla ohjelmointi onnistuu. Kehitystyössä tulee myös huomioida se, että koska Acontis ei sisällä PLC-tuotetta, tulee PLC käytännössä kehittää itse.

EC-Masterin mukana tulee useita esimerkkisovelluksia, jollaista tässäkin työssä hyödynnettiin. Esimerkkisovellus initialisoi väylän ja hoitaa väyläliikenteen jättäen vain datan muokkaamiseen liittyvän ohjelmakoodin kirjoittamisen käyttäjän huoleksi. Esimerkkisovellus siis huolehtii suurimmilta osin siitä, mitä PLC normaalisti tekisi.

Acontisilta on myös saatavilla EC-Engineer-niminen työkalu, jota tässäkin työssä käytettiin. Sillä pystyy luomaan väyläkonfiguraation, jota voidaan käyttää hyödyksi EC-Masteria käytettäessä. Myös

I/O-tuotteiden parametroidin voi hoitaa EC-Engineerin kautta. EC-Engineer toimii ainoastaan Windowsilla. (11.)

Acontisin EC-Master toimii seuraavilla alustoilla (12):

- Linux
- Windows 7/8/10
- VxWorks
- QNX
- RTX
- INTime
- Integrity
- Xenomai
- On Time RTOS-32
- RTEMS
- FreeRTOS
- eCos
- TI-RTOS
- T-Kernel

Tässä työssä alustaksi valittiin Linux. Lisäksi Linuxiin päätettiin asentaa RT-päivitetty käyttöjärjestelmäydin.

EC-Master tukee seuraavia EtherCAT:in ominaisuuksia (13):

- CAN application layer over EtherCAT (CoE)
- Ethernet over EtherCAT (EoE)
- File access over EtherCAT (FoE)
- Servodrive-Profile over EtherCAT (SoE)
- ADS over EtherCAT (AoE)
- Vendor over EtherCAT (VoE)
- Synchronization with Distributed Clock (DC)
- Slave-to-Slave Communication
- Cable Redundancy
- Hot Connect
- TCP-Server and Remote API
- EoE Endpoint

Edellä mainittujen ominaisuuksien lisäksi EC-Master tukee myös FsoE:ta joten EC-Masteria voidaan käyttää niissä sovellutuksissa, joissa tarvitaan turva-automaatio -ominaisuuksia.

Tässä työssä laitteistoksi EC-Masterille valittiin Aaeon Up Squared tietokone. Siihen päätettiin asentaa Ubuntu 18.04 ja RT-päivitetty käyttöjärjestelmäydin. Kyseinen laite valittiin tähän työhön alustaksi, koska siinä on EC-Masterin kanssa yhteensopiva Ethernet-piiri.

EC-Masterin kanssa Linuxilla yhteensopivat Ethernet-piirit ovat seuraavat (14):

- Intel Pro/1000
- Realtek 8111/8168/8169
- Beckhoff CX2000, CX5000, CX5100, etc.
- Texas Instruments AM335x, AM437x, AM57x, CPSW
- NXP PowerPC, eTSEC
- NXP i.MX6, i.MX7, i.MX8
- Altera Cyclone V SoC, Synopsis, DW3504
- Xilinx Zynq-7000, UltraScale+, GEM
- STMicroelectronics , STM32MP1

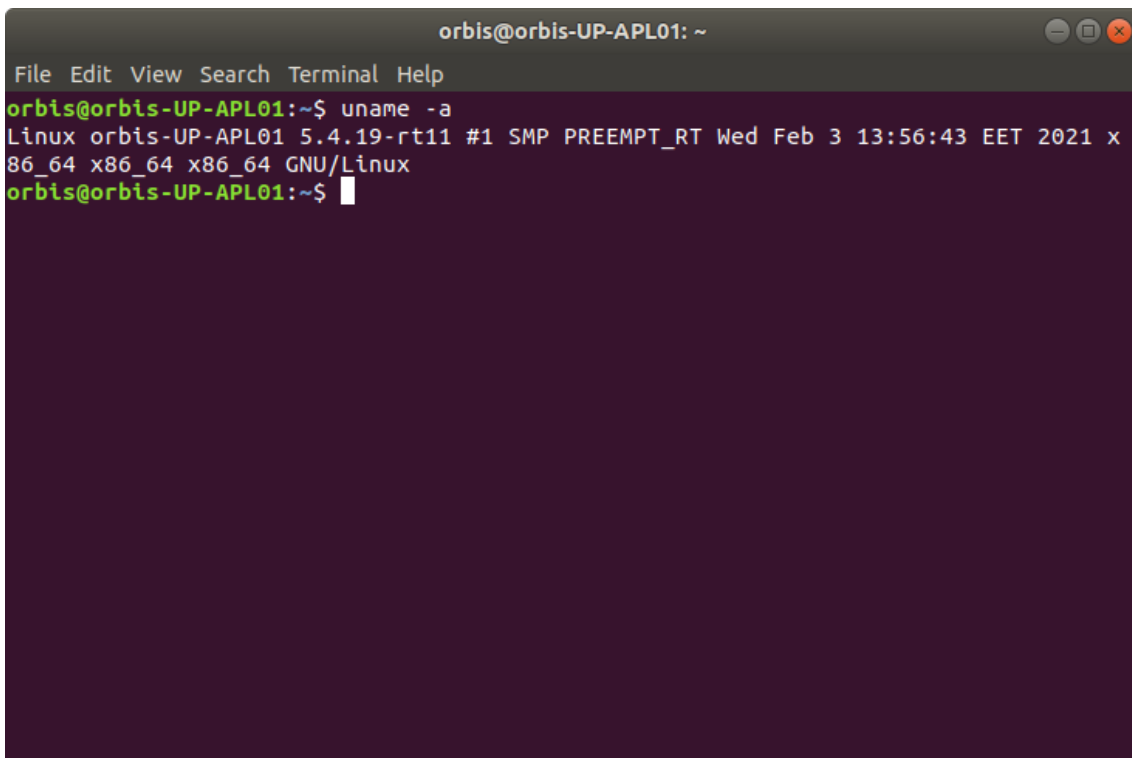
6.1 Acontisin asennus

Ensimmäisenä käytössä olevaan laitteistoon asennettiin Ubuntu 18.04. Sen asentamisen jälkeen päätettiin päivittää Linuxin käyttöjärjestelmäydin reaaliaikayhteensopivaksi. Tämä tapahtui kuvassa 27 esitettyjen käskyjen avulla.

```
1 installing kernel:
2 cd ~
3 mkdir kernel
4 cd kernel
5 sudo apt install build-essential git libssl-dev libelf-dev libncurses-dev
6 wget https://mirrors.edge.kernel.org/pub/linux/kernel/v5.x/linux-5.4.19.tar.xz
7 wget https://mirrors.edge.kernel.org/pub/linux/kernel/projects/rt/5.4/older/patch-5.4.19-rt11.patch.xz
8 xz -cd linux-5.4.19.tar.xz | tar xvf -
9 cd linux-5.4.19
10 xzcat ../patch-5.4.19-rt11.patch.xz | patch -p1
11 cp /boot/config-5.4.0-42-generic .config
12 sudo apt install flex bison
13 make oldconfig
14 make -j8 deb-pkg
15 sudo dpkg -i ../linux-headers-5.4.19-rt11_5.4.19-rt11-1_amd64.deb
16 sudo dpkg -i ../linux-image-5.4.19-rt11_5.4.19-rt11-1_amd64.deb
17 sudo dpkg -i ../linux-libc-dev_5.4.19-rt11-1_amd64.deb
```

KUVA 27. Reaaliaikakernelin asennukseen vaadittavat käskyt

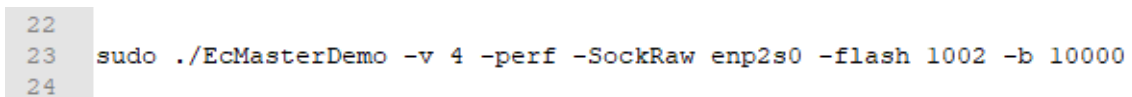
Aluksi ladattiin paketinhallintatyökalulla apt käyttöjärjestelmäytimen asentamiseen ja kääntämiseen tarvittavat työkalut. Tämän jälkeen ladattiin valittu käyttöjärjestelmäytimen versio. Myös käyttöjärjestelmäytimen reaaliaikalaajennus ladattiin. Tämän jälkeen nämä molemmat purettiin ja käyttöjärjestelmäytimen tiedostot päivitettiin reaaliaikalaajennuksen tiedostoilla. Tämän jälkeen käyttöjärjestelmäytimen sisältämät ominaisuudet konfiguroitiin. Sitten käyttöjärjestelmäydin käännettiin. Tähän työvaiheeseen kannattaa varata aikaa useita tunteja, sillä kääntämistyöhön kului aikaa noin kolme tuntia käytössä olevalla laitteistolla. Kääntämisen jälkeen käyttöjärjestelmäydin asennettiin ja järjestelmä käynnistettiin uudelleen. Uudelleenkäynnistyksen jälkeen käytössä oleva käyttöjärjestelmäydin tarkistettiin kuvan 28 mukaisesti. Tulosteesta olevasta PREEMPT_RT-tekstistä tunnistaa, että reaaliaikalaajennus on käytössä.

A terminal window titled 'orbis@orbis-UP-APL01: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'uname -a' and its output: 'Linux orbis-UP-APL01 5.4.19-rt11 #1 SMP PREEMPT_RT Wed Feb 3 13:56:43 EET 2021 x86_64 x86_64 x86_64 GNU/Linux'. The prompt 'orbis@orbis-UP-APL01:~\$' is visible at the end of the output.

```
orbis@orbis-UP-APL01: ~
File Edit View Search Terminal Help
orbis@orbis-UP-APL01:~$ uname -a
Linux orbis-UP-APL01 5.4.19-rt11 #1 SMP PREEMPT_RT Wed Feb 3 13:56:43 EET 2021 x
86_64 x86_64 x86_64 GNU/Linux
orbis@orbis-UP-APL01:~$
```

KUVA 28. Käytössä oleva käyttöjärjestelmäydin

Tämän jälkeen Acontisin toimittamaa esimerkisovellusta testattiin. Sitä on myös mahdollista kokeilla asentamatta Acontisin Ethernet-ajuria ja näin päätettiin tässä vaiheessa tehdä. Tämä onnistui ajamalla komentokehoteessa kuvassa 29 näkyvä käsky. Tulosteesta huomattiin, että EtherCAT-Master lähti käyntiin, se sai yhteyden laitteistoon ja aloitti kommunikaation laitteiston kanssa.

A snippet of a terminal window showing three lines of text. The first line is '22', the second is '23 sudo ./EcMasterDemo -v 4 -perf -SockRaw enp2s0 -flash 1002 -b 10000', and the third is '24'.

```
22
23 sudo ./EcMasterDemo -v 4 -perf -SockRaw enp2s0 -flash 1002 -b 10000
24
```

KUVA 29. EC-Mastein kokeileminen ilman Ethernet-ajurin asennusta

Seuraavaksi päätettiin asentaa Acontisin Ethernet-ajuri. Tämä tapahtui kuvassa 30 näkyvien käskyjen avulla. Ensimmäisenä Ethernet-piirin käyttöjärjestelmäytimen mukana tuleva ajuri otettiin pois käytöstä. Tämän jälkeen järjestelmä käynnistettiin uudelleen. Sitten Acontisin Ethernet-ajuri käännettiin käyttöjärjestelmäydintä vasten ja kääntämällä saatu moduuli ladattiin käyttöjärjestelmäyttimeen.

```

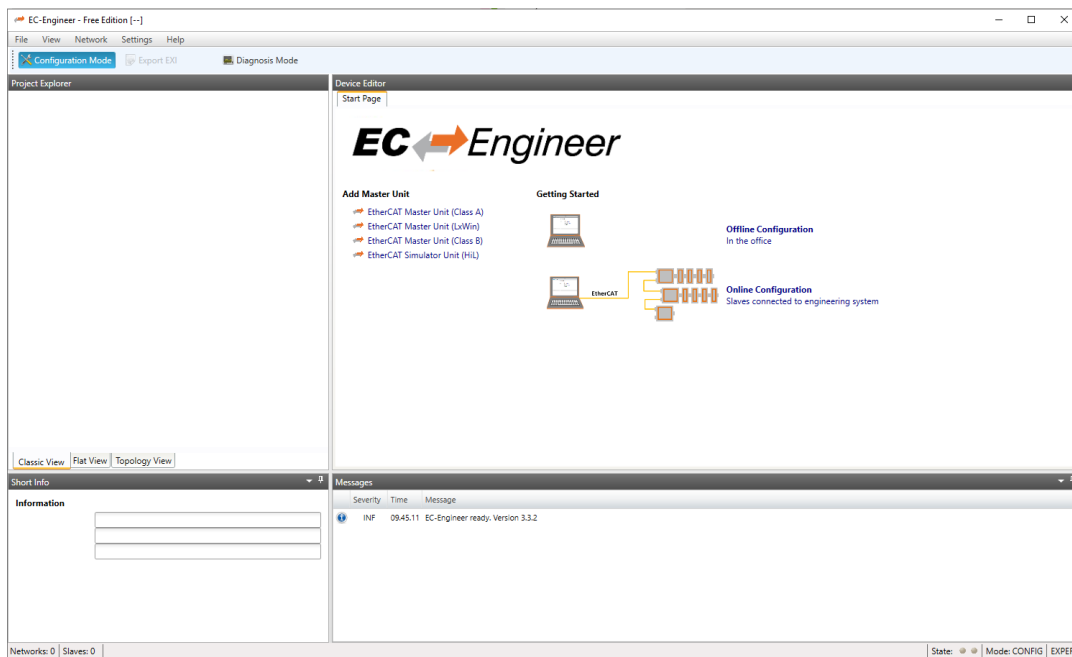
27 echo blacklist r8169 | sudo tee -a /etc/modprobe.d/blacklist.conf
28 sudo update-initramfs -k all -u
29 sudo reboot
30 cd ~/EC-Master-V3.0-Linux_x64-Eval/Sources/LinkOsLayer/Linux/atemsys/
31 make modules
32 sudo insmod atemsys.ko

```

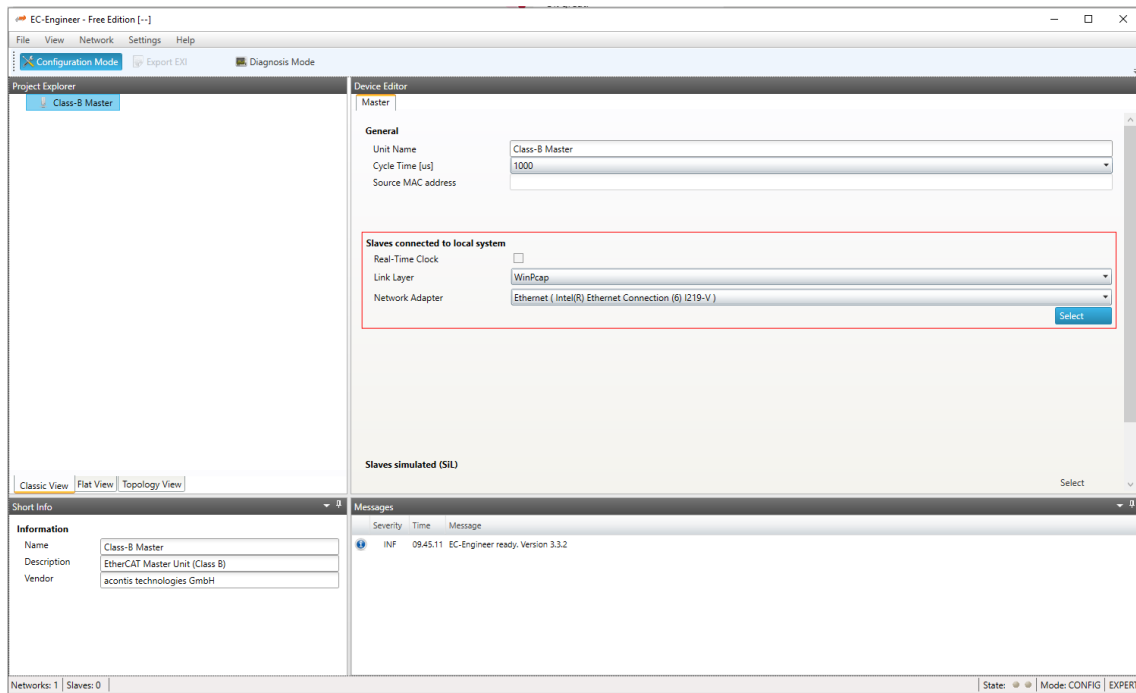
KUVA 30. Acontiksen Ethernet-ajurin asentaminen

6.2 Acontisin käyttö

Ensimmäisenä ennen EC-Masterin käyttöä tuli luoda väyläkonfiguraatio EC-Engineerillä. Tämä tapahtui kytkemällä I/O-laitteisto Windowsia käyttävään tietokoneeseen. Kyseisellä tietokoneella avattiin EC-Engineer ja painettiin kuvassa 31 näkyvää Online Configuration -nappia. Tämän jälkeen valittiin alasvetovalikosta EtherCAT Master Unit (Class B) ja painettiin OK-nappia. Tämän jälkeen aukeavasta näkymästä, joka näkyy kuvassa 32, painettiin Select-nappia.

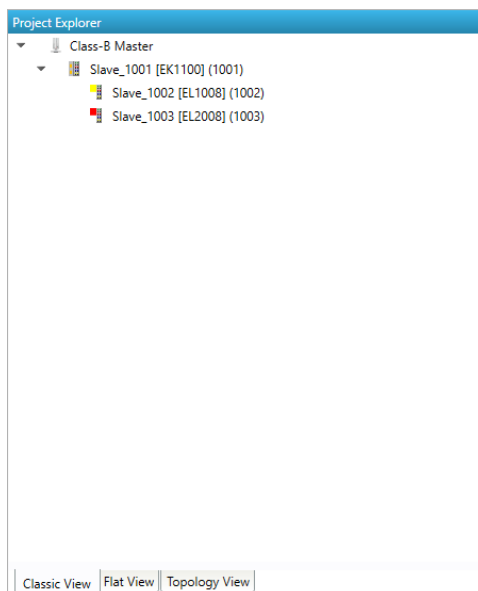


KUVA 31. EC-Engineerin aloitusnäyttö



KUVA 32. Verkkoadapterin valinta

Verkkoadapterin valinnan jälkeen ohjelma hakee väylästä löytyvät laitteet. Tässä tapauksessa löytyvät laitteet näkyvät kuvassa 33. Kun oli tarkistettu, että väylästä löytyneet laitteet olivat oikeita, voitiin painaa Export EXI -nappia ja tallentaa väyläkonfiguraation sisältävä EXI-tiedosto tietokoneelle. Tämän jälkeen kyseinen EXI-tiedosto siirrettiin ohjauslaitteelle.



KUVA 33. Väylästä löytyneet laitteet

Seuraavaksi voitiin alkaa tekemään testisovellusta käyttäen hyödyksi EC-Masterin mukana tulevaa esimerkkiohjelmia. Esimerkkiohjelmassa on merkattuna selkeä kohta, johon ohjelman käyttäjä voi lisätä koodia, joka muokkaa prosessidataa. Tässä tapauksessa haluttiin vain testata, että datan lukeminen I/O-korteilta ja datan kirjoittaminen I/O-korteille toimii, joten tehtiin sovellus, joka vain kopioi EL1008 sisääntulokortin ensimmäisen kanavan tilan EL2008 ulostulokortin ensimmäiseen kanavaan. Esimerkkisovelluksessa käytettiin apuna EC-Masterin mukana tulevaa EC_COPYBITS-funktiota, jolla voi kopioida halutun määrän bittejä osoitteesta toiseen. Esimerkkisovelluksen muokatusta funktiosta tuli kuvan 34 mukainen.

```

/*****
**
\brief demo application working process data function.

This function is called in every cycle after the the master stack is started.

*/
static EC_T_DWORD myAppWorkpd(T_DEMO_THREAD_PARAM* pDemoThreadParam,
    EC_T_BYTE* pbyPDIn, /* [in] pointer to process data input buffer */
    EC_T_BYTE* pbyPDOut /* [in] pointer to process data output buffer */
)
{
    EC_UNREFPARM(pbyPDIn);
    T_DEMO_FLASH_DATA* pFlashData = &pDemoThreadParam->FlashData;
    EC_COPYBITS(pbyPDOut, 0, pbyPDIn, 0, 1);

    return EC_E_NOERROR;
}

```

KUVA 34. Esimerkkisovellukseen lisätty itse muokattu funktio

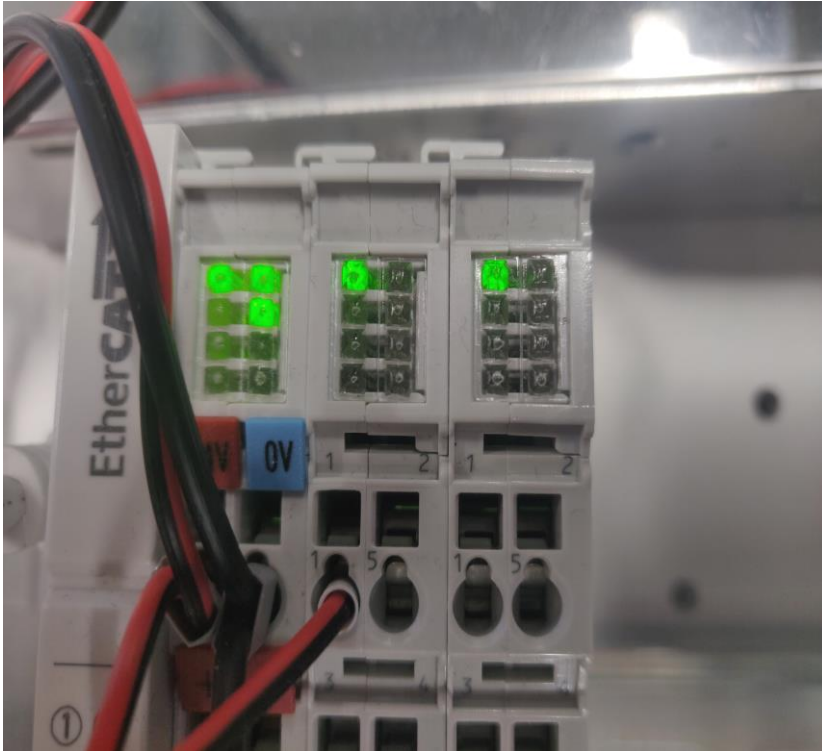
Esimerkkiohjelma käännettiin tämän jälkeen ja se käynnistettiin kuvassa 35 näkyvällä komennolla. Ohjelman tulosteesta huomattiin, että ohjelma käynnistyy oikein, se saa yhteyden väylälaitteisiin ja aloittaa niiden kanssa kommunikaation. Ohjelman toimivuus testattiin kytkemällä EL1008 sisääntulokortin ensimmäiseen kanavaan 24 V:n jännite, jolloin EL2008 ulostulokortin ensimmäisen kanavan tilaa kuvaava LED syttyi palamaan. Tämän näkee kuvasta 36.

```

41
42 sudo ./demo -v 4 -perf -rtl8169 1 1 -b 10000 -f ~/exi.xml
43

```

KUVA 35. Esimerkkiohjelman käynnistävä komento



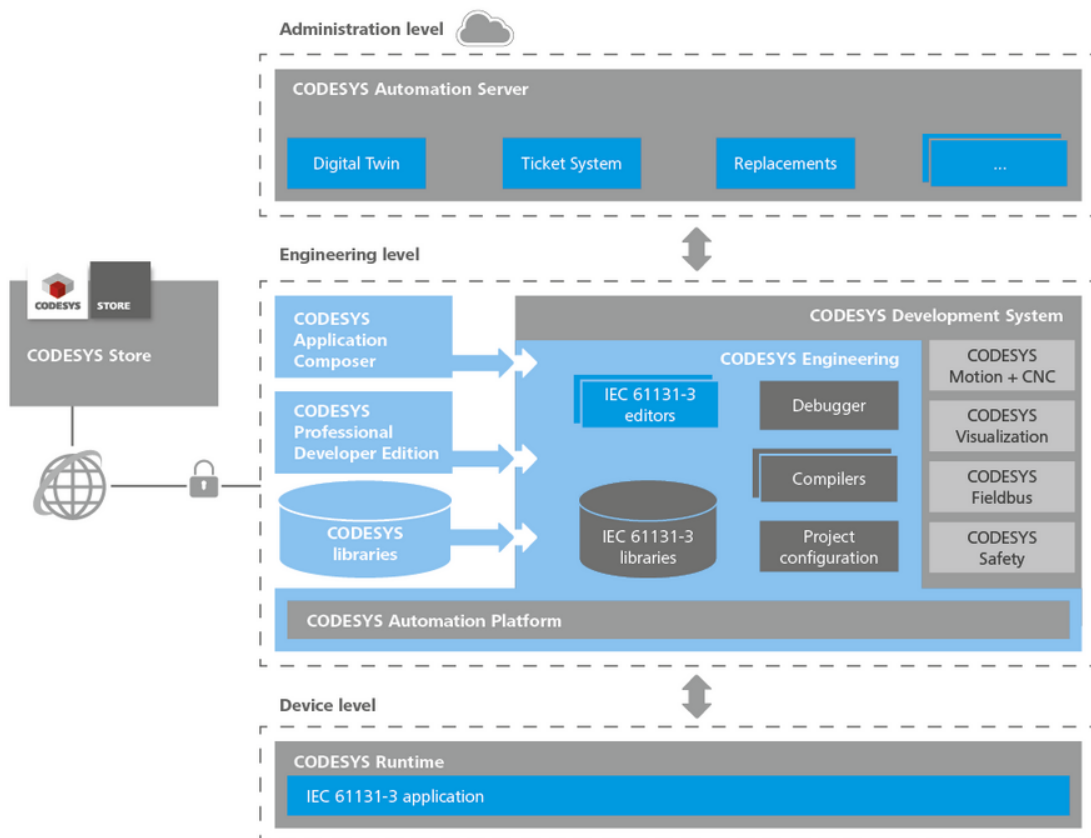
KUVA 36. Oikein toimivan ohjelman toteaminen

6.3 Yhteenveto Acontiksesta

EC-Masterista löytyi sekä hyviä että huonoja puolia. Erityisesti kehitystyön raskaus tulee huomioida EC-Masterin kanssa. Tämä johtuu siitä, että EC-Master ei tarjoa kehitysympäristöä ja PLC-tuotetta, joten näiden normaalisti hoitamat roolit tulee ohjelmoida itse. EC-Master tukee myös FsoE:ta, joten sen kanssa voi toteuttaa turva-automaatiota vaativia projekteja. Hyvänä puolena EC-Master pyörii Linuxin päällä, joten valmistellusta käyttöjärjestelmälevystä voidaan ottaa levykuva, jota voidaan hyödyntää laitteiden käyttöönotossa.

7 CODESYS ETHERCAT MASTER

CODESYS on tuote, joka sisältää EtherCAT Masterin lisäksi paljon muutakin. Se sisältää täyden IEC61131-3 yhteensopivan kehitysympäristön, tuen monille muille väyläprotokollille, konfigurointi- ja diagnostiikkatyökaluja sekä SoftPLC-tuotteen. Voidaankin sanoa, että se sisältää kaiken mitä tarvitaan automaatiojärjestelmien ohjelmointiin sekä ajamiseen. CODESYSin arkkitehtuuri on kuvattu kuvassa 37.



KUVA 37. CODESYSin arkkitehtuuri.

CODESYSin mukana itsessään ei tule käyttöjärjestelmän reaaliaikalaajennusta, joten esimerkiksi Linuxilla on suotavaa asentaa jokin reaaliaikalaajennus. CODESYS toimiikin useilla eri alustoilla kuten Windows ja Linux.

Linuxilla CODESYS toimii minkä tahansa verkkopiirin kanssa ja Windowsilla se toimii seuraavien verkkopiirien kanssa (15):

- Realtek 8139
- Realtek 8169
- Intel EtherExpressPro1000
- Intel EtherExpress PRO/100.

CODESYS tukee useita EtherCAT:in ominaisuuksia mukaan lukien FsoE:ta, joka tarkoittaa, että sitä voidaan käyttää sovelluksissa, joissa tarvitaan turva-automaatiota. CODESYSin tukee muun muassa seuraavia EtherCAT:in ominaisuuksia (15):

- CAN application layer over EtherCAT
- Ethernet over EtherCAT
- Servodrive over EtherCAT
- Safety over EtherCAT
- File over EtherCAT
- Vendor over EtherCAT.

Tässä työssä valittiin CODESYSin kanssa alustaksi Aaeon Up Squared -tietokone ja käyttöjärjestelmäksi Ubuntu 18.04. Ubuntuun asennettiin reaaliaikalaajennus.

7.1 CODESYSin asennus

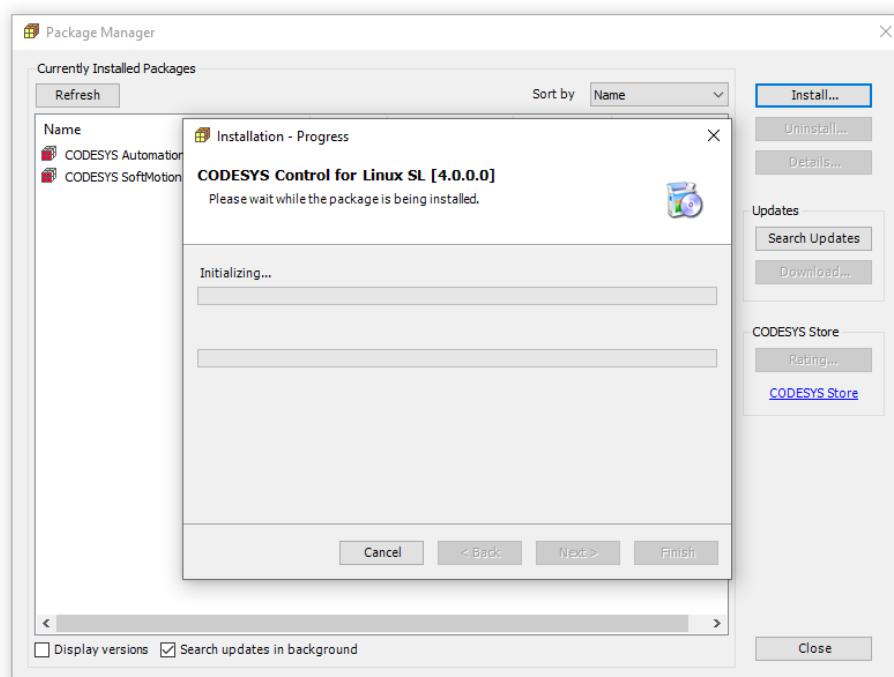
Tämä osio tehtiin samalla alustalla kuin Acontisin EC-Master. Näin ollen tässä ei käydä läpi käyttöjärjestelmätimen RT-päivitystä. On kuitenkin huomattava, että käyttöjärjestelmätimen RT-päivitys on tarpeellista myös CODESYSin kanssa.

CODESYSin asennus aloitettiin lataamalla CODESYSin kehitysympäristö ohjelmointityökaluna käytettävälle tietokoneelle. Lisäksi ohjauslaitteena käytettävälle tietokoneelle asennettiin muutamia tarvittavia paketteja. Pakettien asennukseen ja käyttöönottoon liittyvät komennot on esitetty kuvassa 38.

```
3
4 sudo apt install openssh-server
5 sudo systemctl enable ssh
6 sudo systemctl start ssh
7
```

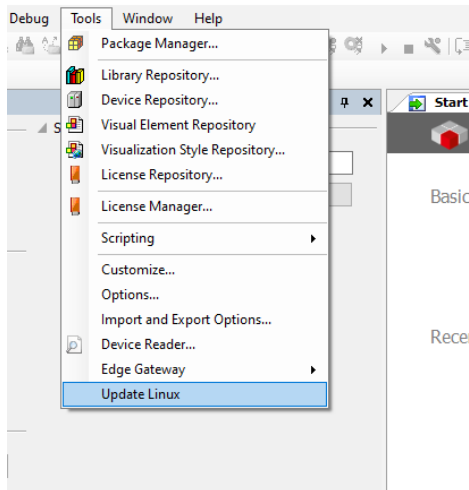
KUVA 38. Ohjauslaitteiston valmistelu

Tämän jälkeen CODESYSin ohjelmointiympäristö käynnistettiin ja siihen asennettiin Package Manageria käyttäen Linux-laitteiden ohjaamiseen tarkoitettu paketti. Tämä on esitetty kuvassa 39.

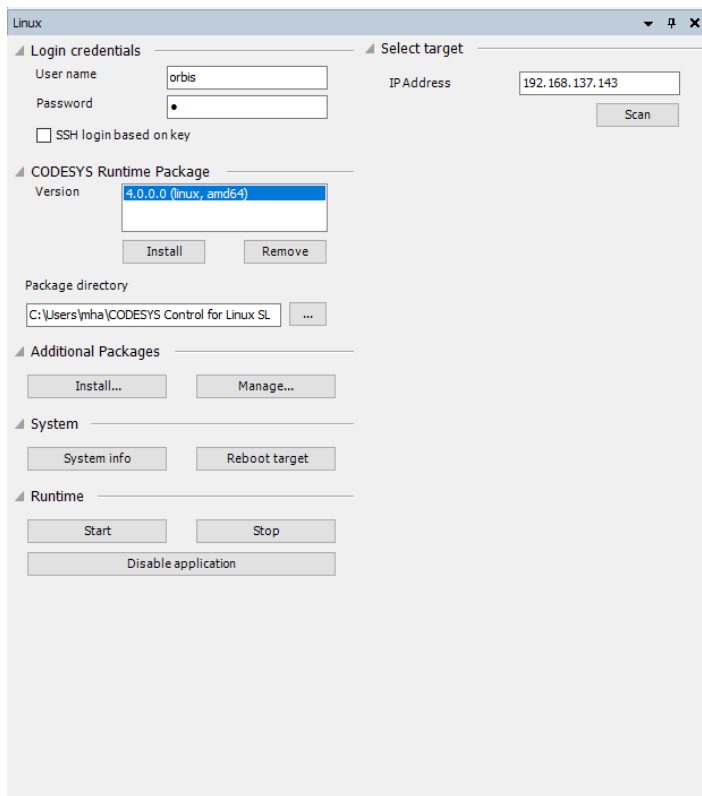


KUVA 39. Linux-paketin asentaminen

Tämän jälkeen avattiin Update Linux -työkalu kuvassa 40 näkyvästä paikasta. Ruudulle aukesi kuvassa 41 näkyvä näkymä. Siihen syötettiin ohjauslaitteen sisäänkirjautumistiedot sekä IP-osoite. Tämän jälkeen CODESYS Runtime Package asennettiin painamalla Install-nappia. Seuraavaksi runtime käynnistettiin painamalla Start-nappia.

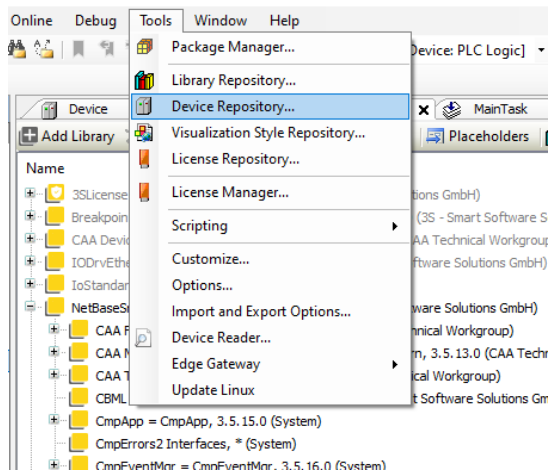


KUVA 40. Update Linux -työkalun sijainti

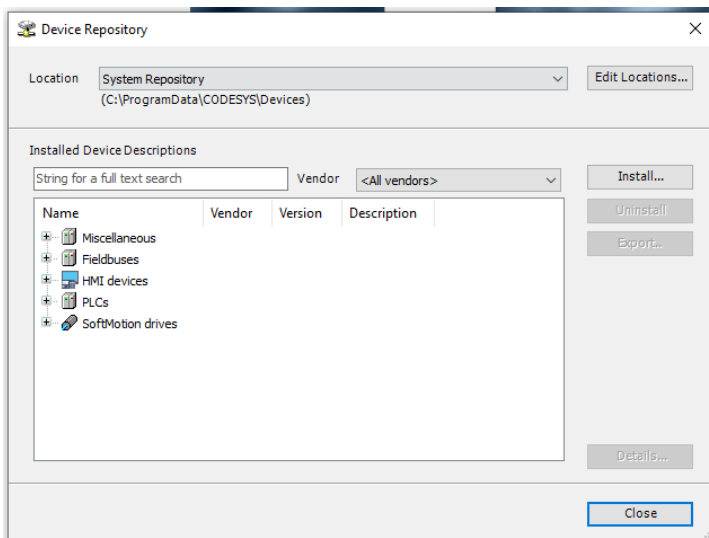


KUVA 41. Runtimeen asennus ohjauslaitteistoon

Seuraavaksi CODESYSiin lisätiin laitetiedot sisältävät ESI-tiedostot käyttämällä Device Repository -työkalua, joka löytyy kuvan 42 osoittamasta paikasta. Device Repository on esitetty kuvassa 43. Siellä Install-nappia painamalla pääsee valitsemaan oikeat halutut ESI-tiedostot.



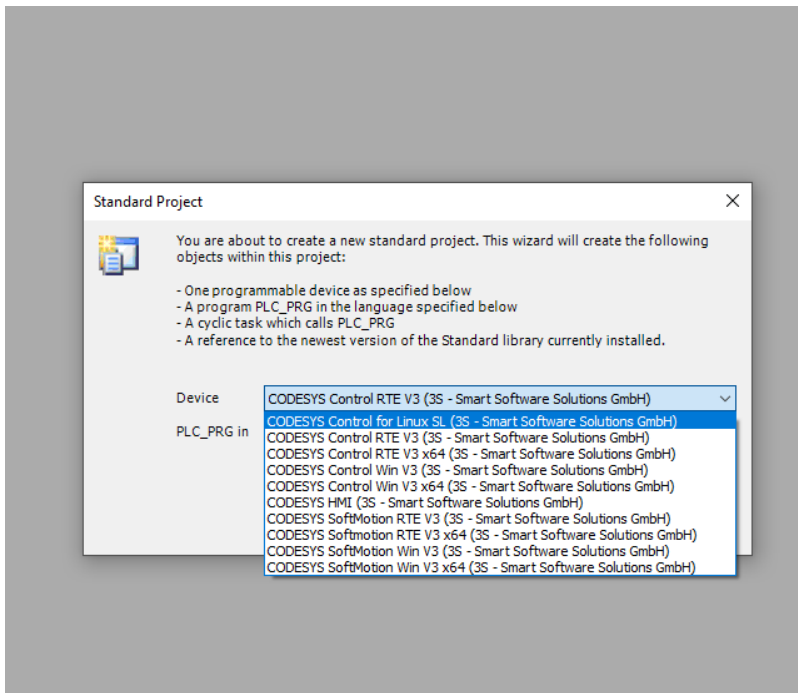
KUVA 42. Device Repositoryn sijainti



KUVA 43. Device Repository

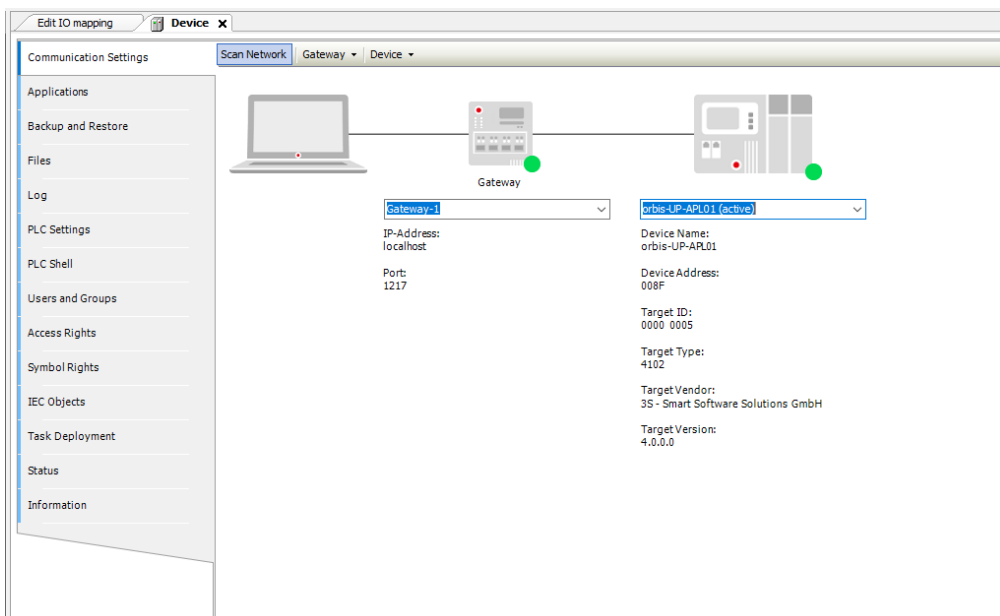
7.2 CODESYSin käyttö

CODESYSin käyttö aloitettiin luomalla uusi projekti. Projektityypiksi valittiin Standard Project. Projektin luonnin jälkeen valittiin kuvassa 44 näkyvä laitetyyppi.



KUVA 44. Valittu laitetyyppi

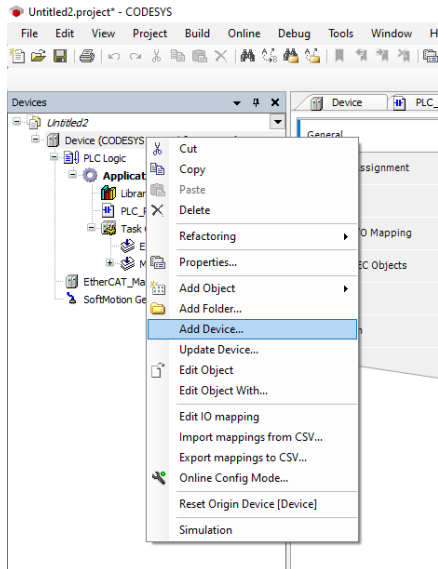
Seuraavaksi valittiin käytettävä ohjauslaite. Tämä tapahtui avaamalla Device-välilehti ja painamalla Scan Network. Löytynyt laite ja sen tiedot ilmestyivät kuvan 45 näkymän oikeaan laitaan.



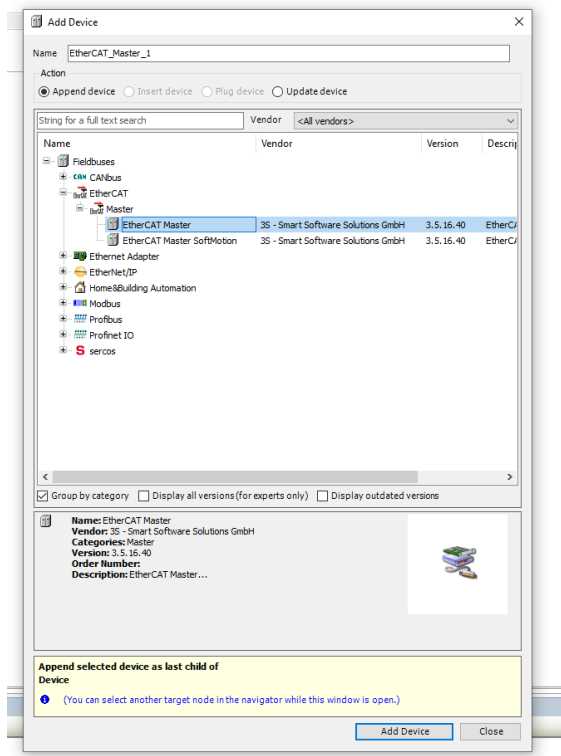
KUVA 45. Ohjauslaitteen valinta

Ohjauslaitteen valinnan jälkeen tuli projektiin lisätä EtherCAT Master. Tämä tapahtui avaamalla Add Device -ikkuna kuvan 46 osoittamasta paikasta. Add Device -ikkunassa tuli etsiä valikosta

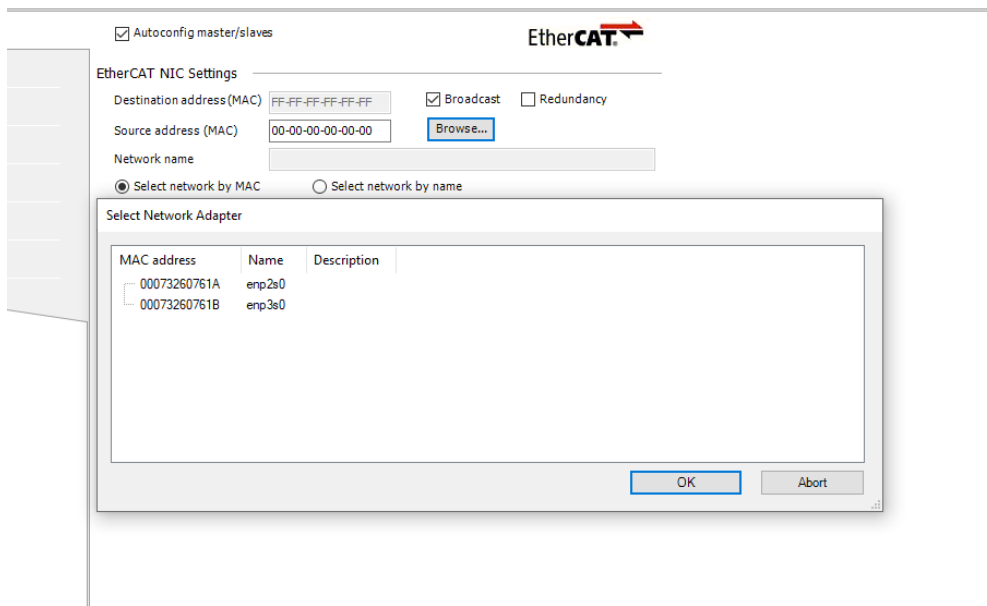
EtherCAT Master ja painaa Add Device -nappia. Tämän jälkeen projektiin ilmestyi EtherCAT Masteri. Add Device ikkuna on kuvattu kuvassa 47. Tämän jälkeen EtherCAT Masterille tuli valita oikea verkkokortti kuvassa 48 näkyvästä ikkunasta.



KUVA 46. Add Device -ikkunan sijainti

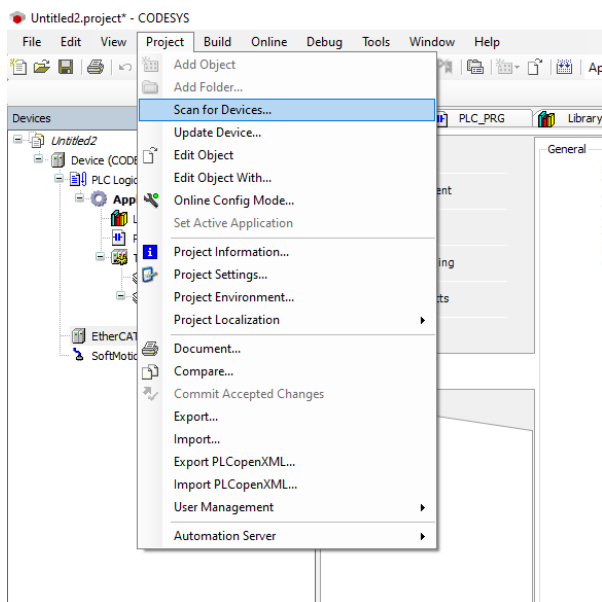


KUVA 47. Add Device -ikkuna

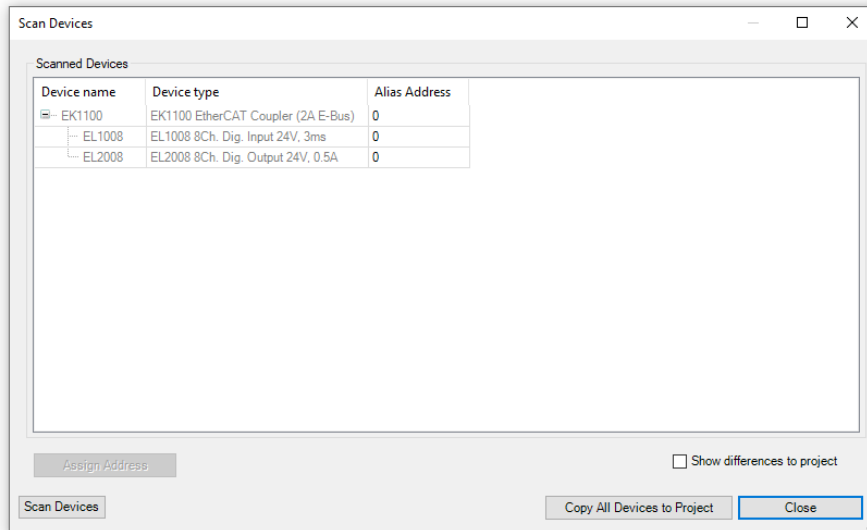


KUVA 48. Verkkokortin valinta

Verkkokortin valinnan jälkeen voitiin käyttää Scan Devices -työkalua kuvan 49 osoittamasta paikasta. Scan Devices -työkalussa avautui kuvan 50 mukainen näkymä. Sen avulla lisätään väylästä löytyneet EtherCAT-laitteet projektiin. Työkalussa tarvitsi vain painaa Copy All Devices to Project -nappia ja väylästä löytyneet laitteet lisättiin projektiin.

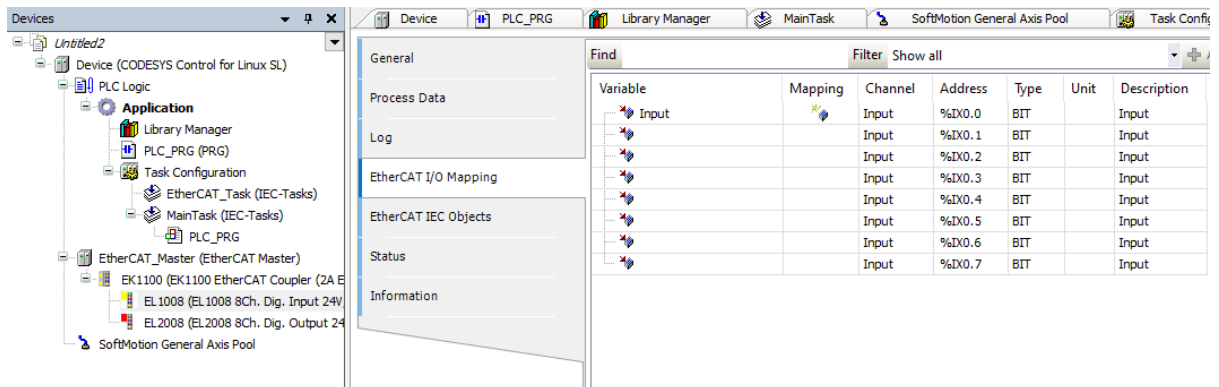


KUVA 49 Scan Devices -työkalun sijainti

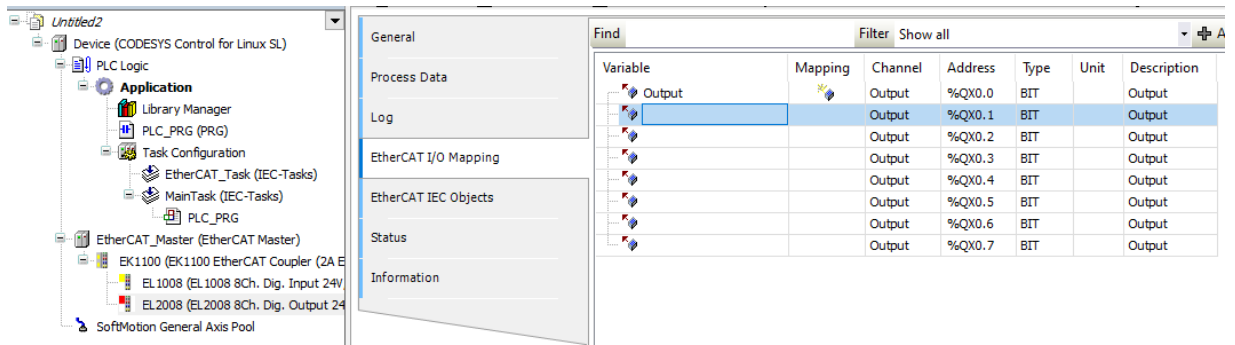


KUVA 50. Scan Devices -työkalu

I/O-laitteiden lisäämisen jälkeen EL1008-sisääntulokortin ensimmäiselle kanavalle luotiin Input-niminen muuttuja ja EL2008-ulostulokortin ensimmäiselle kanavalle luotiin Output-niminen muuttuja. Tarkoituksena oli luoda yksinkertainen mahdollinen ohjelma, jolla voitaisiin testata, saadaanko sisääntuloja luettua ja ulostuloja kirjoitettua. Päätettiin siis kirjoittaa ohjelma, joka kopioi Input-muuttujan tilan suoraan Output-muuttujaan. Muuttujien luonnit on esitetty kuvissa 51 ja 52.

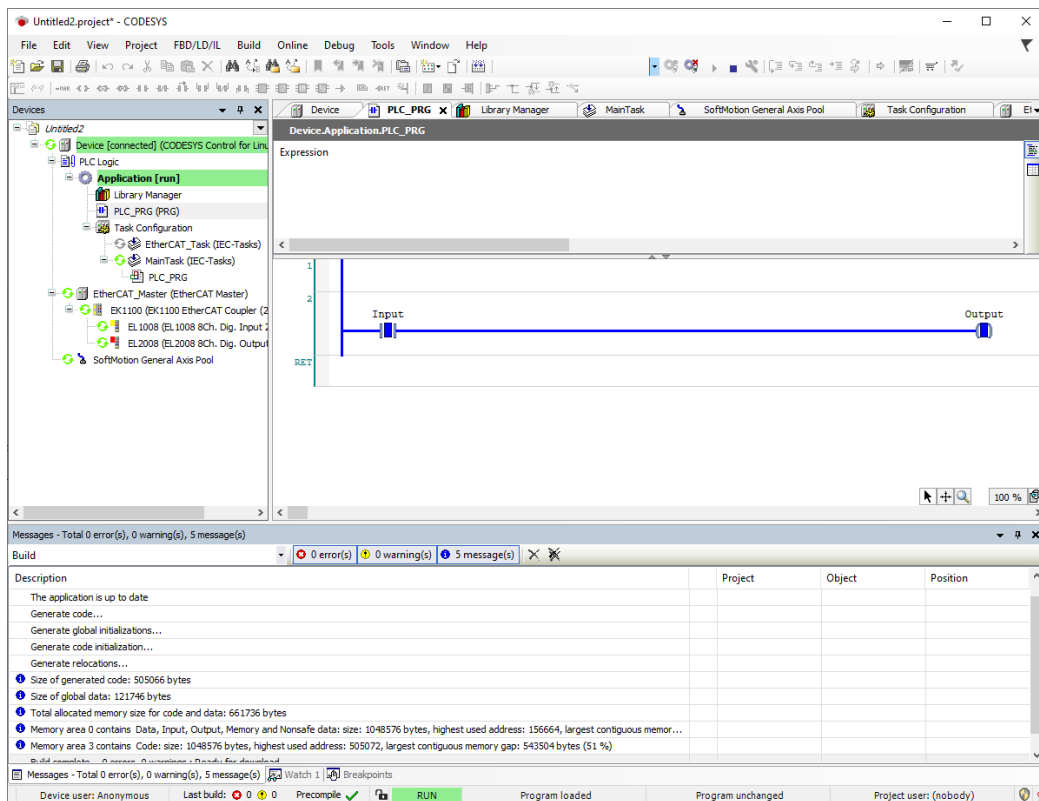


KUVA 51. Input-muuttujan luominen



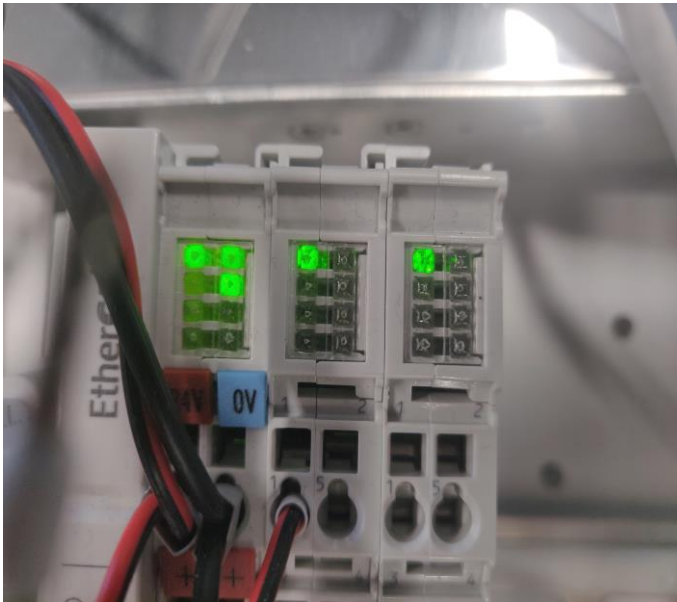
KUVA 52. Output-muuttujan luominen

Seuraavaksi oli aika ladata projekti ohjauslaitteelle. Tämä tapahtui painamalla Build-nappia, jonka jälkeen painettiin ylävalikosta Online -> Login. Tässä vaiheessa ohjelma kysyy, halutaanko sovellus ladata ohjauslaitteelle. Kun on vastattu myöntävästi, sovelluksen suoritus alkaa. Ohjelman suoritus on kuvattu kuvassa 53.



KUVA 53. Testiohjelma

Seuraavaksi tarkistettiin, palaako EL1008-sisääntulokortin ensimmäisen kanavan tilaa kuvaava LED samaan aikaan EL2008-ulostulokortin ensimmäisen kanavan tilaa kuvaavan LED:n kanssa. Tulokset tästä näkyvät kuvassa 54. Uudelleen käynnistäessä ohjauslaite aloitti sovelluksen suorituksen automaattisesti.



KUVA 54. Sovelluksen toimivuuden tarkistus

7.3 Yhteenveto CODESYSistä

CODESYSin hyvänä puolena voidaan pitää pitkälle kehittyneitä kehitysympäristöä ja sitä, että se tukee useita eri väyläprotokollia. Myös se, että sitä voidaan käyttää sekä Windowsin että Linuxin kanssa, on hyvä puoli. Linuxin kanssa käytettäessä voidaan käyttöjärjestelmälevystä ottaa levykuva, jota voidaan käyttää tulevien käyttöönottojen apuna. CODESYS tukee myös FsoE:ta, joten sitä voidaan käyttää projekteissa, jotka vaativat turva-automaatiota. Huonona puolena voidaan pitää sitä, että Linuxille CODESYS ei tarjoa Ethernet-piirikohtaisia ajureita vaan Linuxilla käytettäessä käytetään yleistä ajuria, joka toimii kaikkien Ethernet-piirien kanssa. Tällöin järjestelmän reaaliaikaisuus saattaa kärsiä, mikä voi vaikuttaa varsinkin servo-ohjauksiin.

8 YHTEENVETO

Työn teon aikana tehtiin useita huomioita vertailun kohteena olevista EtherCAT Mastereista. Jos tarkastellaan alkuvaatimuksia, niin sekä CODESYS että Acontis täyttävät yhden tärkeimmistä alkuvaatimuksista. Molemmat niistä toimivat Linuxilla, joten niiden kanssa voidaan tehdä käyttöönottoja kopioimalla levykuva toimivasta laitteesta.

Kaikki kolme testaukseen päätynyttä tuotetta olivat kaupallisia tuotteita. Tämän valinnan takana oli erityisesti se, että yleensä kaupallisiin tuotteisiin on saatavilla tukipalveluja. Valitettavasti tämän työn aikana yhdenkään vertailun kohteena olleen valmistajan tukipalveluja ei päästy hyödyntämään, joten niistä ei voitu tehdä vertailua.

Kaksi vertailun kohteena olevista tuotteista sisältää kehitysympäristön ja PLC-tuotteen. Nämä ovat KINGSTAR ja CODESYS. Molemmissa näistä oli hyväksi koettu kehitysympäristö. Tämä voidaan ehdottomasti laskea näiden tuotteiden eduksi EC-Masteriin verrattuna. EC-Masteria käyttäessä tulisi nämä ohjelmiston osa-alueet kehittää itse. Tällöin toki saataisiin juuri sellainen ratkaisu kuin halutaan, mutta se kuluttaisi huomattavan määrän kehitysresursseja.

Tarkastelun kohteena olevista tuotteista sekä CODESYS että EC-Master tukevat FsoE:ta, joten niitä voitaisiin käyttää projekteissa, joissa tarvitaan turva-automaatiota. Tarve turva-automaatiolle toki riippuu projektin luonteesta, joten tuen puute ei mahdollisesti käyttökohteen mukaan ole ongelma.

Ainoastaan EC-Masterilla tehdyt sovellukset voitaisiin päivittää kopioimalla binääritiedostot vanhojen tilalle. Sekä CODESYSillä että KINGSTARilla ohjelman lataamiseen ohjaislaitteelle tarvittaisiin kehitysympäristö. Näiden tapauksissa tämä aiheuttaisi sen, että sovelluksen päivittäjällä tulisi olla käytössään sovelluksen lähdekoodit, mikä ei välttämättä olisi haluttu tilanne. Sellaisissa tapauksissa, joissa sovelluksen lähdekoodia ei haluta jakaa laitteiston käyttäjälle, tulisi laitteiston valmistajan itse käydä päivittämässä sovellus.

LÄHTEET

1. EtherCAT Technology Group. EtherCAT – the Ethernet Fieldbus. Hakupäivä 09.02.2021
<https://www.ethercat.org/en/technology.html>
2. Beckhoff. EtherCAT State Machine. Hakupäivä 09.02.2021.
https://infosys.beckhoff.com/english.php?content=../content/1033/ax5000_usermanual/html/Bt_EcBasics_EcStateMachine.htm&id=
3. UP. Up Squared Series. Hakupäivä 09.02.2021. <https://up-board.org/upsquared/specifications/>
4. Beckhoff Automation 2021. C6015-0010 | Ultra-compact industrial PC. Hakupäivä 09.02.2021. <https://www.beckhoff.com/fi-fi/products/ipc/pcs/c60xx-ultra-compact-industrial-pcs/c6015-0010.html>
5. Beckhoff Automation 2021. EK1100 | EtherCAT Coupler. Hakupäivä 09.02.2021.
<https://www.beckhoff.com/fi-fi/products/i-o/ethercat-terminals/ek1xxx-bk1xx0-ethercat-coupler/ek1100.html>
6. Beckhoff Automation 2021. EL1008 | 8-channel digital input terminal 24 V DC, 3 ms. Hakupäivä 09.02.2021. <https://www.beckhoff.com/fi-fi/products/i-o/ethercat-terminals/el1xxx-digital-input/el1008.html>
7. Beckhoff Automation 2021. EL2008 | 8-channel digital output terminal 24 V DC, 0.5 A. Hakupäivä 09.02.2021. <https://www.beckhoff.com/en-us/products/i-o/ethercat-terminals/el2xxx-digital-output/el2008.html>
8. KINGSTAR. KINGSTAR Support for EtherCAT Technology Group Standards. Hakupäivä 09.02.2021. <https://kingstar.com/library/content/KINGSTAR-Support-for-ETG-Standards2.pdf>
9. KINGSTAR 2021. KINGSTAR PLC. Hakupäivä 09.02.2021.
<https://kingstar.com/products/soft-plc/>
10. KINGSTAR 2020. SUPPORTED NETWORK INTERFACE CARDS. Hakupäivä 09.02.2021. <https://kingstar.com/library/documentation/3.8docs/KINGSTAR-Supported-NIC.pdf>
11. Acontis technologies. EtherCAT® Network Configuration, Diagnosis, and Monitoring. Hakupäivä 09.02.2021. <https://www.acontis.com/en/ethercat-configuration.html>

12. Acontis technologies. EtherCAT Master Stack. Hakupäivä 09.02.2021.
<https://www.acontis.com/en/ethercat-master.html>
13. Acontis technologies. Features according to ETG.1500 Master Classes. Hakupäivä 09.02.2021. <https://www.acontis.com/en/features.html>
14. Acontis technologies. EtherCAT Master Software on Linux. Hakupäivä 09.02.2021.
<https://www.acontis.com/en/ethercat-master-linux.html>
15. CODESYS. CODESYS ETHERCAT MASTER. Hakupäivä 09.02.2021.
<https://www.codesys.com/products/codesys-fieldbus/industrial-ethernet/ethercat.html>