

LAMPPU - TEHTÄVÄNANTO

Sisällysluettelo

Pisteytys:	1
Tehtävä:.....	1
Ohjeet:.....	1
Odotettu ulostulo	6
Reflektointi.....	6

Pisteytys:

- Max 5p
 - Ohjelma toimii (3p)
 - Reflektointi (2p)

Tehtävä:

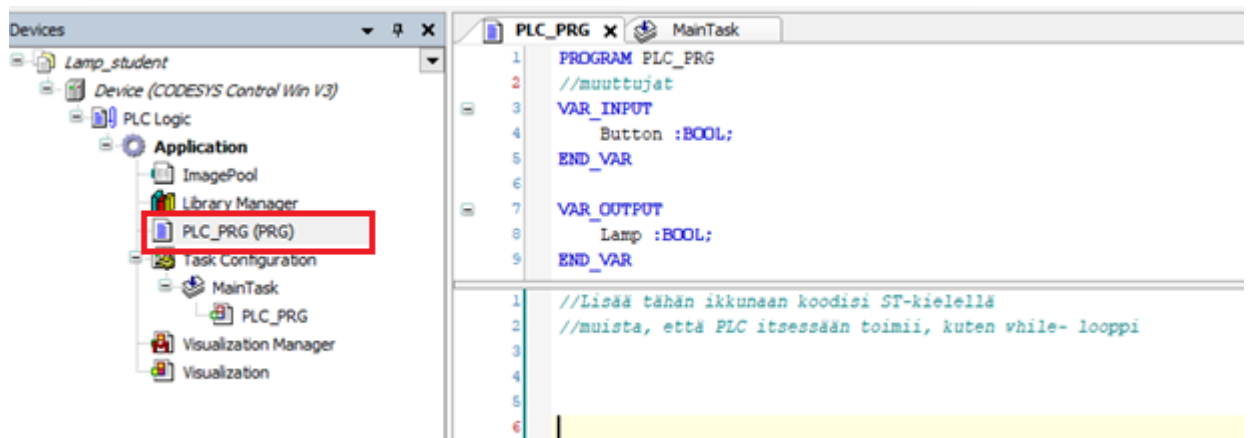
Tehtävässä pääset tutustumaan Codesys-ohjelmistoon tekemällä yksinkertaisen ohjelman. Ohjelman tarkoitus on ohjata lampun tilaa (ulostulo, output) virtuaalisella napilla (sisääntulo, input). Ohjelman logiikka on mahdollisimman triviaali: jos nappi on painettu, laitetaan virta päälle, muuten laitetaan virta pois.

INPUT	OUTPUT
1	1
0	0

Ohjeet:

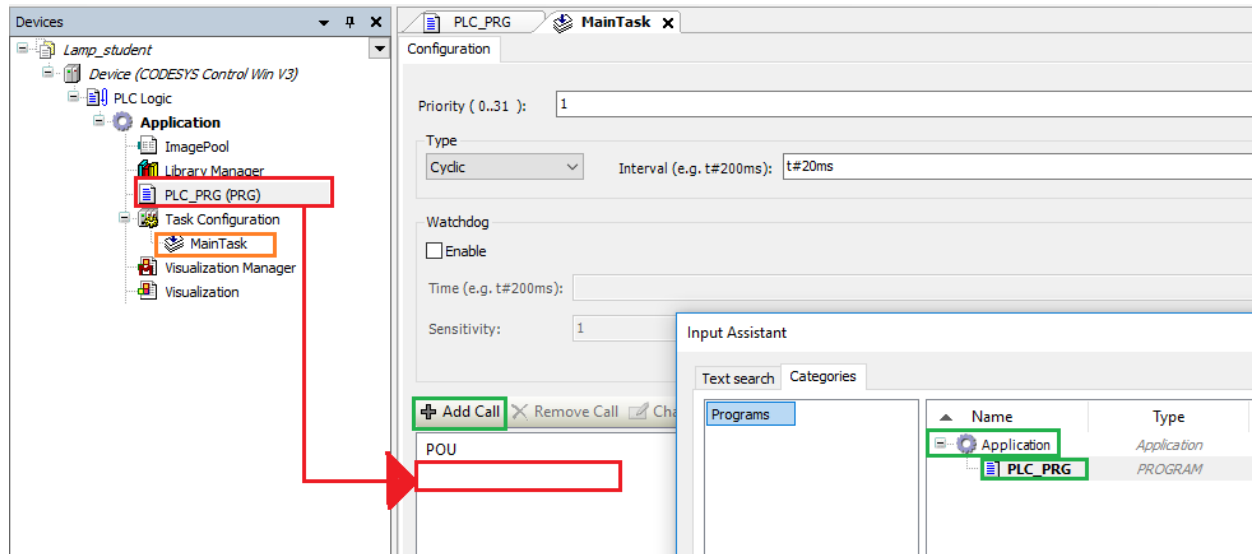
1. Lataa kurssin MyCourses-sivulta lamppu projekti ja avaa se Codesys-ohjelmistolla.

2. Erittäin todennäköisesti käyttämäsi ohjelmistoversio on uudempi, kuin MyCourses sivulta saatava pohja, jolloin ohjelmisto herjaa puuttuvista kirjastoista. Ota esille MyCourses sivulta löytyvä ” Ohjeita Codesys ongelmatilanteisiin” ja toimi sen ohjeiden mukaan.
3. Avaa PLC_PRG-ohjelma vasemmalla olevasta Devices-ikkunasta. Mikäli välilehti ei aukea, siirry kohtaan 6 – kirjaudu sisään virtuaaliselle PLC:lle ja tuplaklikkaa uudestaan PLC_PRG(PRG). PLC_PRG:n pitäisi nyt aueta. Loggaa ulos ja jatka kohdasta 4.
4. PLC_PRG:n ikkuna on jaettu kahteen osaan. Huomaa, että ohjelmassa on valmiiksi luotu BOOL muuttujat alla olevan kuvan mukaisesti (Kuva 1). Codesys-ohjelmistossa yläikkunassa määritetään kyseisen ohjelman lokaalit muuttujat. Varsinainen ohjelma kirjoitetaan alaikkunassa (ohjeiden kohta 6).



Kuva 1: PLC_PRG(PRG) - muuttujat ja koodi

5. Seuraavaksi varmistetaan, että ohjelma on suoritusjonossa (Kuva 2). Avaa MainTask-välilehti (kuvassa **ORANSSI**). Ohjelman "PLC_PRG(PRG)" voi raahata POU:n alle (kuvassa **PUNAINEN**). Vaihtoehtoisesti voitte lisätä ohjelman painamalla MainTaskin Add Call ja avautuvasta ikkunasta Applicationin alta PLC_PRG (kuvassa **VIHREÄ**).



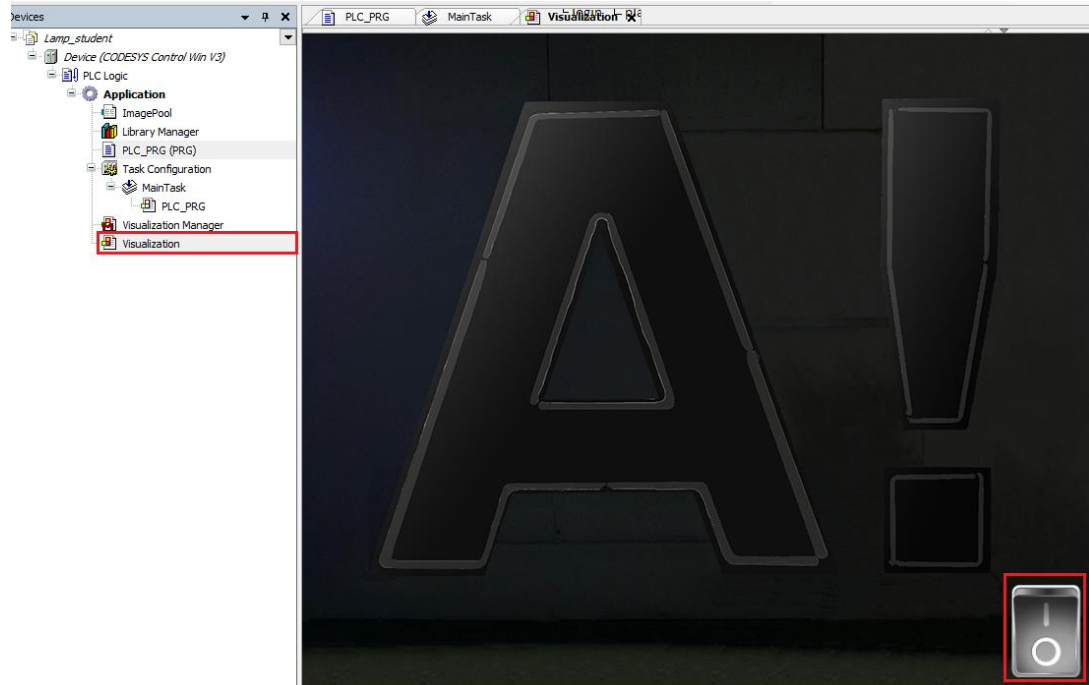
Kuva 2: Ohjelma laitetaan suoritusjonoon

6. Seuraavaksi kirjoitetaan itse ohjelmakoodi (PLC_PRG). Ei ole olemassa yhtä oikeaa ratkaisua, mutta suosittelemme tässä tapauksessa ST-kielen käyttöä. Tutustu ST kielen yleiseen ohjeeseen MyCourses-sivuilla.
7. Tallenna koodi. Tallennettu koodi tulee nyt "buildata"/KÄÄNTÄÄ Tämän voi tehdä ylävalikon Build välilehdellä (Alt + b peräkkäin TAI F11). Codesys ohjelmisto "buildaa" ohjelman automaattisesti myös kirjautuessa virtuaaliseen PLC-laitteeseen. Mikäli ohjelmaa ei voida buildata, codesys varoittaa tästä erikseen.
8. Kirjautu virtuaaliselle PLC-laitteelle ylävalikon Online välilehdeltä (Alt + o peräkkäin TAI Alt + F8 yhtäikaa). Kun olet kirjautunut, suorita kylmäresetointi (reset cold) ylävalikon Online- välilehdeltä. Tällöin ohjelma aloittaa toimintansa alusta. Käynnistä sovellus ylävalikon Debug välilehdeltä (Alt + d peräkkäin TAI F5). Voit myös käyttää kuvassa 3 olevia pikakomentoja.



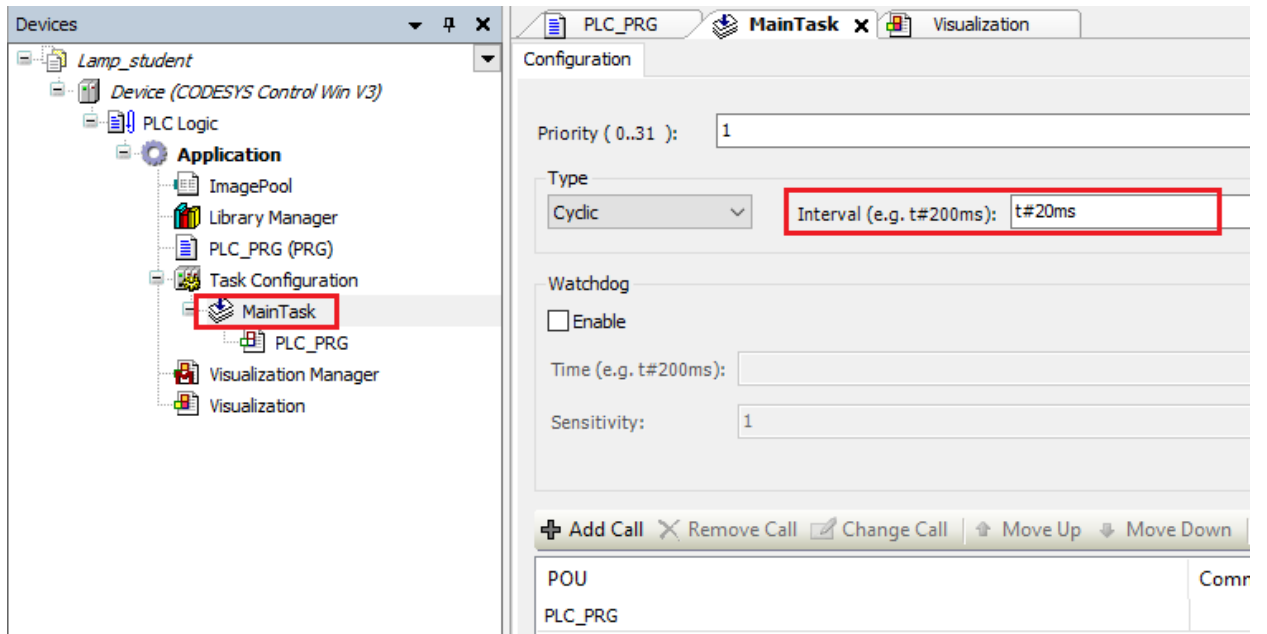
Kuva 3: Pikakomennot

9. Avaa visualization-välilehti ja kokeile toimiiko lamppu halutulla tavalla, painamalla keinukytkintä (kuva 4).



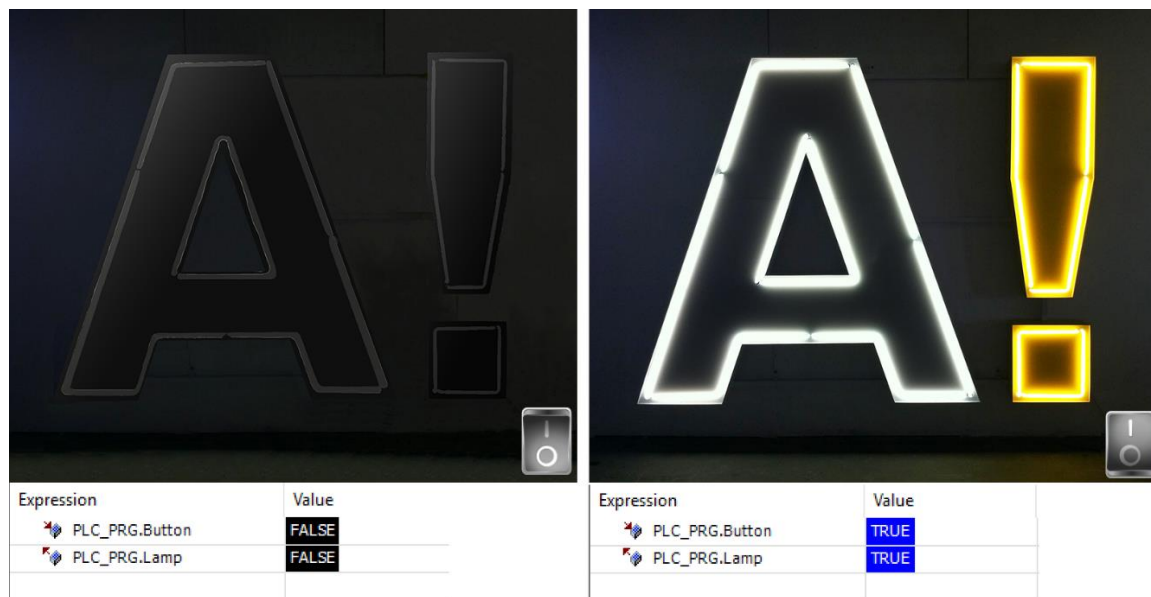
Kuva 4: Visualisointi

10. Pysäytä sovellus neliön muotoisesta stop-napista ja katkaise yhteys PLC laitteistoon login napin vieressä olevalla logout-napilla (kuva 3). Ollessasi yhteydessä PLC-laitteistoon ohjelmakoodin editointi ei ole sallittu.
11. Muuta syklin intervallin arvoksi 2000 ms (ks. Kuva 5) ja aja ohjelma uudelleen ja kiinnitä huomiota kuinka kauan kuluu siitä, kun nappia painetaan siihen, että valo syttyy. Valmistaudu selittämään lampun käyttäytyminen ja pohdi mikä olisi sopiva syklin intervalli. Kokeile ehdotustasi.



Kuva 5: Syklin intervallin muuttaminen

Odotettu ulostulo



Kuva 6: Odotettu ulostulo

Reflektointi

1. Miten syklin intervallin arvon muuttaminen vaikutti lampun käyttöön ja miksi? Mikä olisi sopiva arvo ja miksi? Miksi jokin arvo ei mahdollisesti ole sopiva?
2. Suunnittele ajankäyttöä koko kurssin näkökulmasta parisi kanssa. Huomio, että
 - Meillä on kerran viikossa kahden tunnin ohjattu harjoitus, joten harjoitusaikojen ulkopuolellakin pitää työskennellä.
 - o Ohjelmointitehtävien mitoituksessa on lähdetty siitä, että niitä tehdään myös viikkoharjoituksen ulkopuolella.
 - o Etusivun kuvan pisteytys antaa käsityksen harjoitusten työmäärästä ja osa harjoituksista ei ole pakollisia.
 - Kurssilaisten taustoissa ja ohjelmointikokemuksissa on suuria eroja, joten kurssi on organisoitu niin että kukin pari voi edetä omaan tahtiin, mutta tämä edellyttää ajankäytön suunnittelua.

- Melkein kaikki tehtävät sisältävät lukutehtävän, jonka voi ja kannattaa tehdä halutessaan yksin kotona. Myös teoria ja tehtävänanto kannattaa lukea enne harjoitusaikaa.
 - o Anturitaulu-tehtävän voi halutessaan tehdä yksin – se joka tapauksessa tehdään viikoittaisen harjoitusajan ulkopuolella. Anturitaulu demotaan kuitenkin harjoitusaikana.
- **Mieti parisi kanssa, miten organisoitte työskentelyn ja valmistaudu kertomaan siitä harjoitusten vetäjälle suullisesti.**
 - o Määrittäkää tavoite arvosanalla - tarkastakaa arvosteluperusteet
 - o Määrittäkää mitkä kaikki tehtävät on suoritettava täysillä pisteillä, jotta tavoitteeseen päästään
 - Osa arvioinnista perustuu aina tehtävän reflektointiin. Älä kopioi naapuriryhmän ratkaisua
 - o Pohtikaa, mitkä tehtävät on oltava suoritettuna ja millä viikolla, jotta tavoitearvosanan saavuttaminen on realistista
 - Muistakaa, että harjoitustilaisuudessa on vain yksi assistentti ja useampi autettava. Et voi siis olettaa, että assistentti on heti saatavilla.
 - o Kirjallinen aikataulutus voi auttaa hahmottamaan työkuormaa