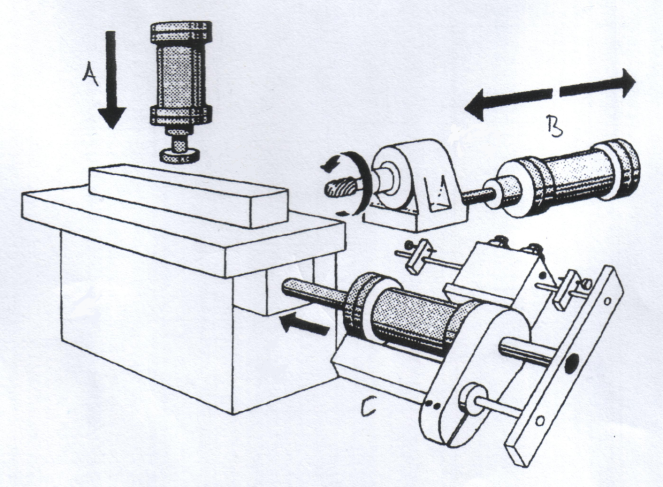
**Uporaba orodij za programiranje industrijskih krmilnikov**

**Primer enostavnega projekta:**

Avtomatizirajte glavna in podajalna gibanja lesnega frezalnega stroja. Projekt pripravite za izvedbo s pnevmatskim krmiljenjem ter za izvedbo z električnim krmiljenjem v relejski in prosto programirljivi tehniki s pomočjo logičnega modula LOGO! in krmilnika Simatic S7-314C 2PN-DP.

**1. Tehnološka skica**



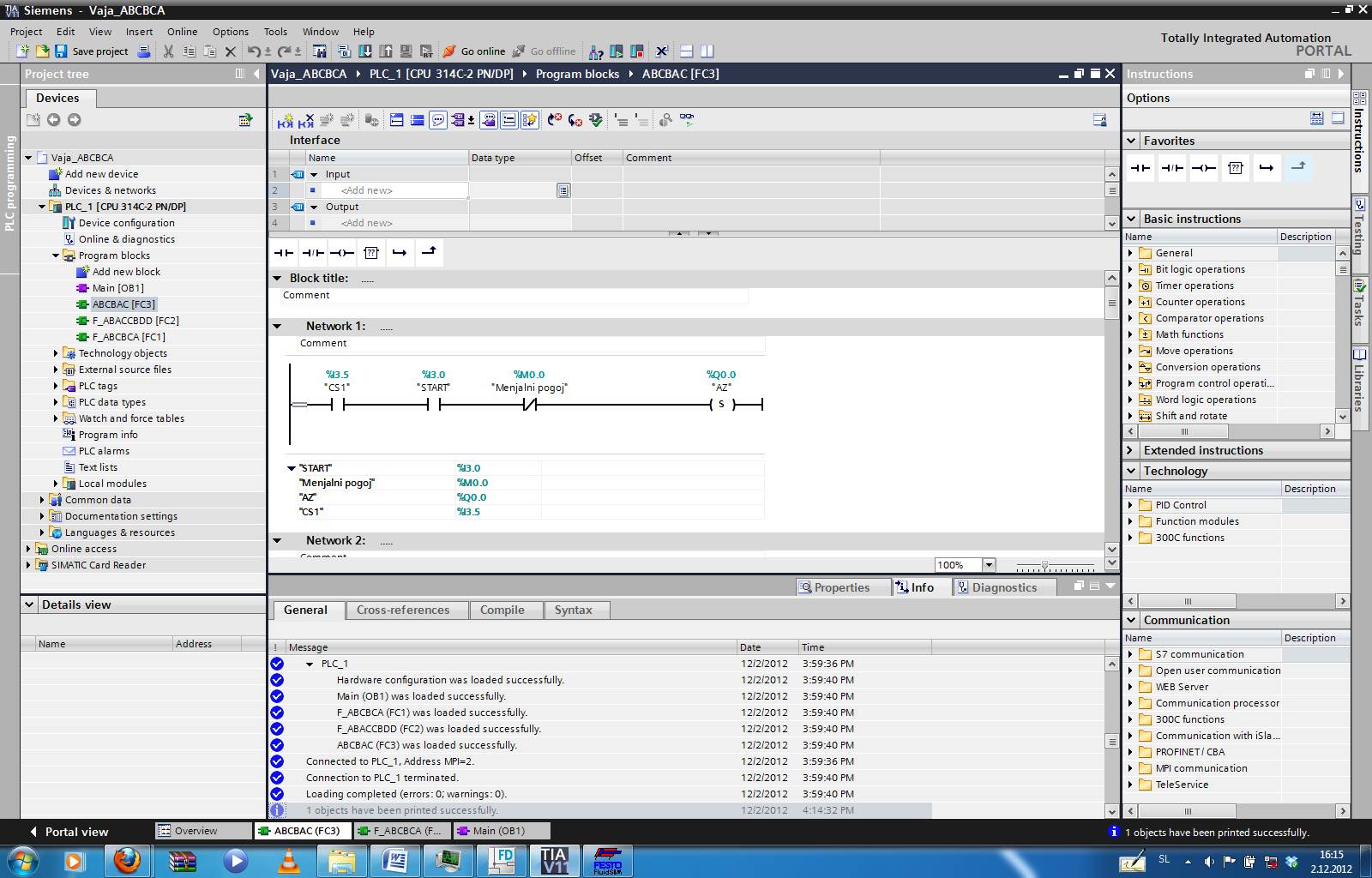
**TIA PORTAL V11**

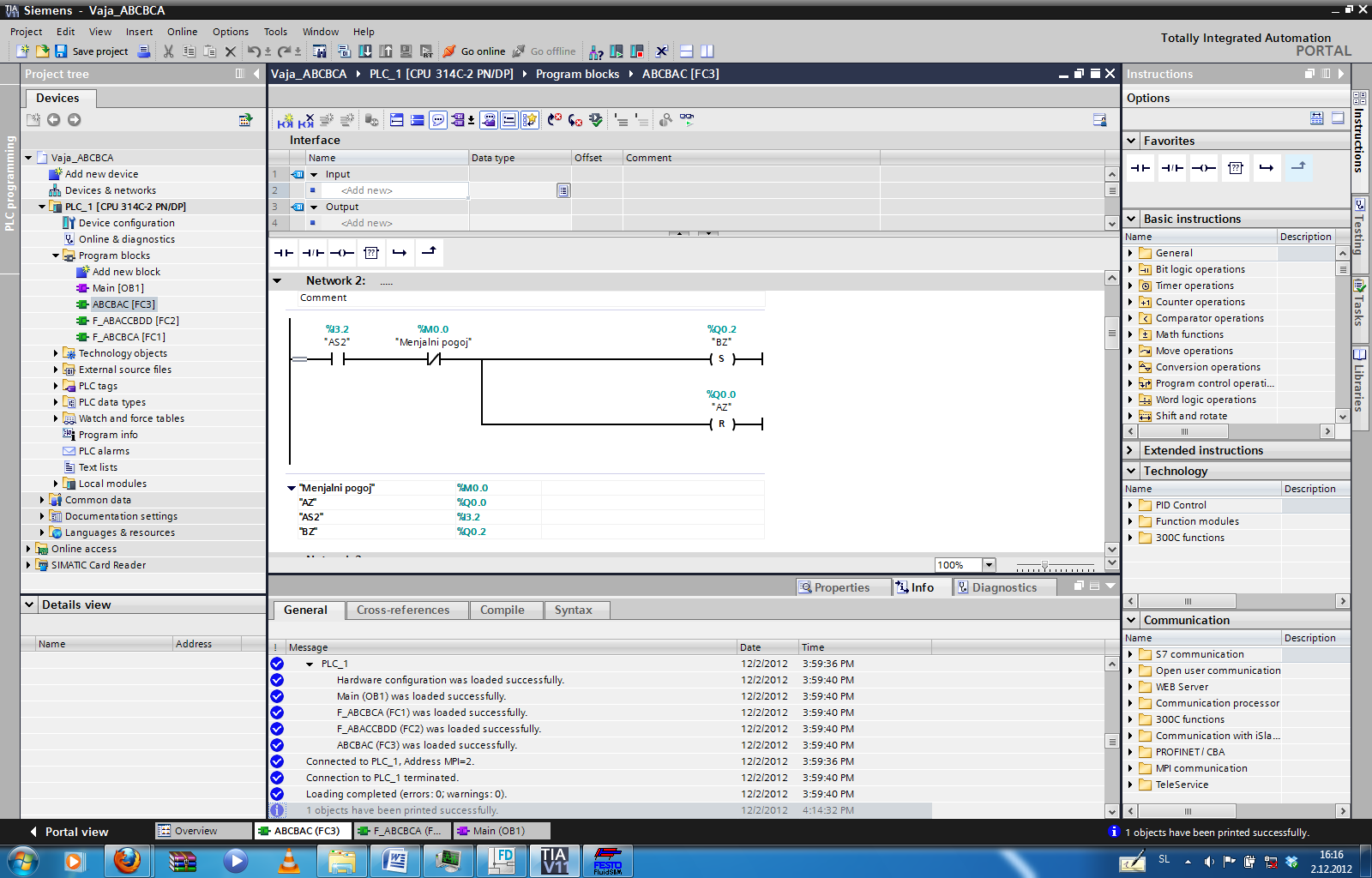
1. **Konfiguracija krmilnega sistema**

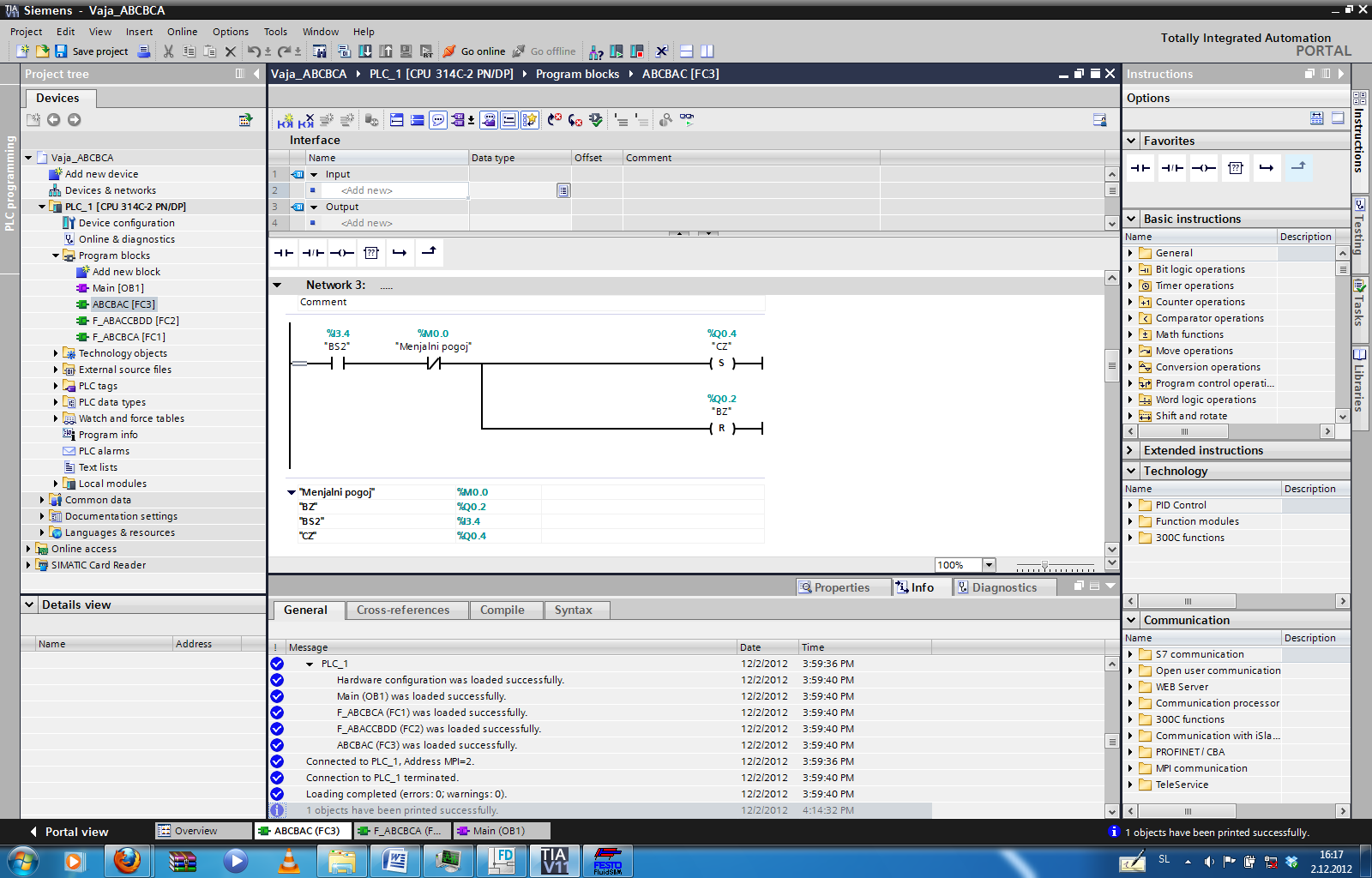
Po zagonu TIA portala kreiramo nov projekt: vnesemo ime projekta (**Frezalni\_stroj**), določimo mapo, v katero se bo projekt shranil in potrdimo s **Create**. Na naslednjem oknu izberemo **Configure a device**, da določimo komponente našega krmilnega sistema. Na naslednjem oknu v naš sistem najprej vstavimo novo CPU enoto krmilnika. Kliknemo na **Add new device**, iz seznama izberemo naš krmilnik **SIMATIC S7-300 -> PLC -> CPU 314C-2 PN/DP ->** **6ES7, 314-6EH04-0AB0** ter kliknemo na **Add**. Ker gre za kompaktno izvedbo CPU enote, ima ta že vgrajene vhodne/izhodne priključke, in sicer AI5/AO2 ter 24DI/16DO. V spodnjem oknu lahko izberemo zavihek **Properties**, kjer lahko spremenimo začetne naslove teh priključkov. Izbrani CPU enoti nato dodamo napajalnik; na desni strani kliknemo na **PS** ter izberemo **PS 307 5A -> 6ES7 307-1EA01-0AA0**, ter modul s 16 vhodnimi priključki, stikali, s katerimi lahko simuliramo delovanje digitalnih tipal; **DI -> DI16 x 24VDC -> 6ES7 321-1BH02-0AA0**. Po opravljeni konfiguraciji na levi strani okna z desno tipko miške kliknemo na naš **PLC\_1[CPU 314C-2 PN/DP]** ter poženemo prevajanje **Compile** nastavljene hardverske konfiguracije.

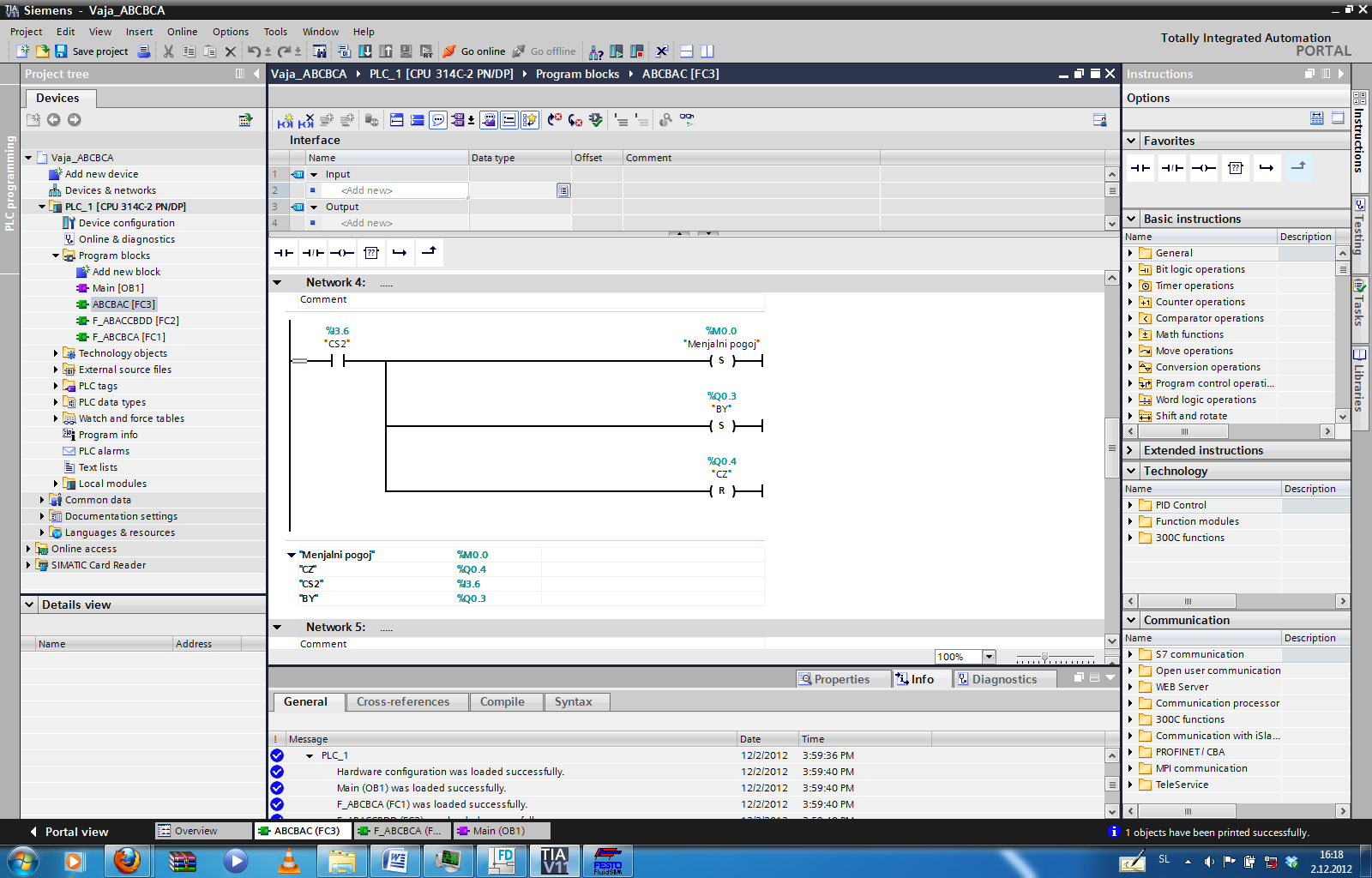
1. **Programiranje**

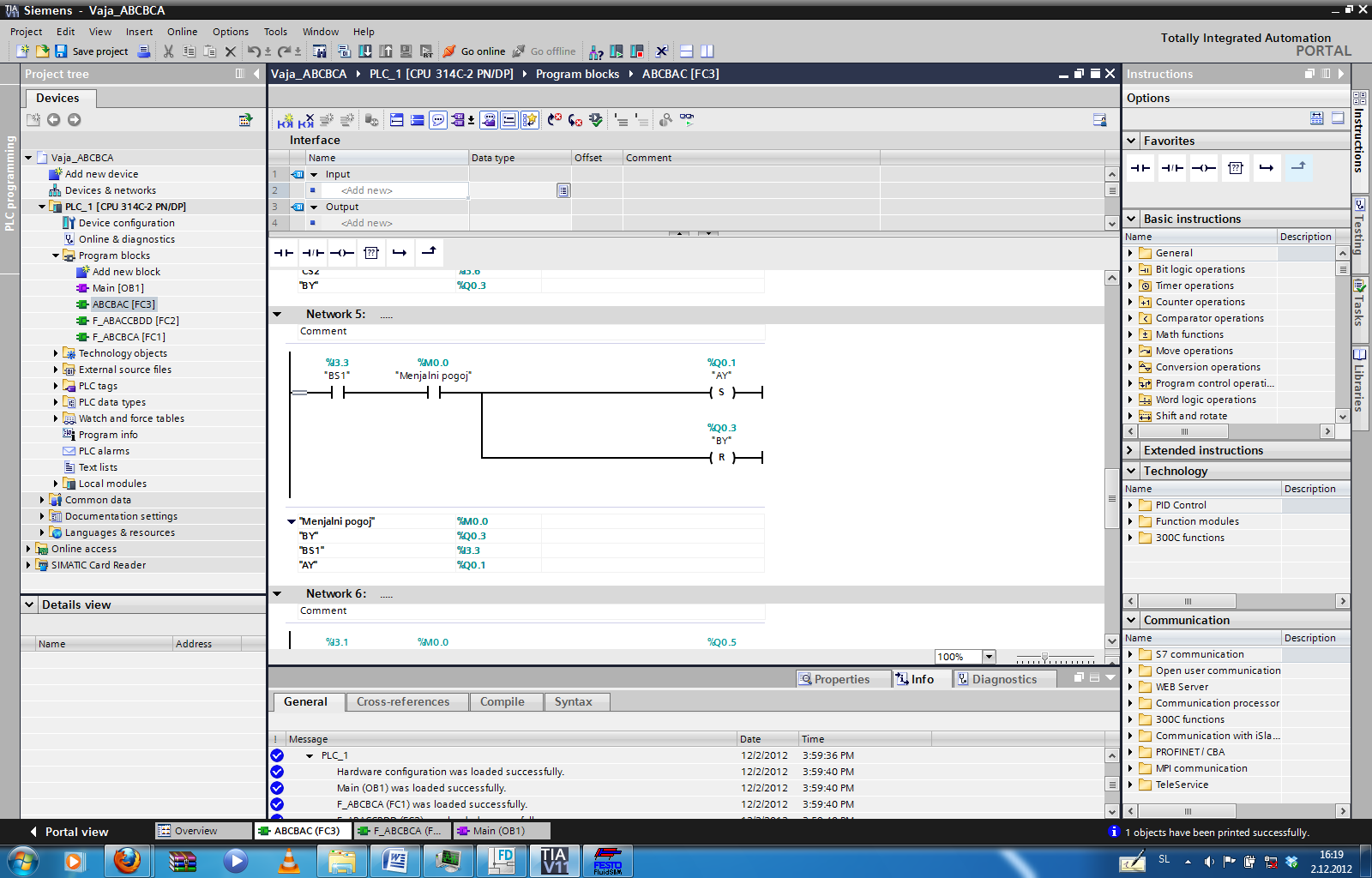
Na levi strani odpremo zavihek našega PLC-ja **PLC\_1[CPU 314C-2 PN/DP]** ter znotraj tega še zavihek **Program blocks**, kjer dvakrat kliknemo na **Add new block**. V naslednjem oknu najprej izberemo **Function**, nato določimo ime funkcije **F1\_Frezanje** ter to potrdimo z **OK**. Na levi strani v drevesu programskih blokov smo sedaj poleg glavnega organizacijskega bloka dobili še našo funkcijo, ki se je hkrati tudi odprla za pisanje. Sedaj najprej z dvoklikom odpremo blok **Main [OB1]** ter vanj v prvi tokokrog povlečemo našo funkcijo. S tem smo dosegli, da se bo v vsakem ciklu programa izvršila tudi naša funkcija. V spodnji vrstici sedaj ponovno odpremo našo funkcijo, v katero začnemo pisati podprogram za frezalni stroj.

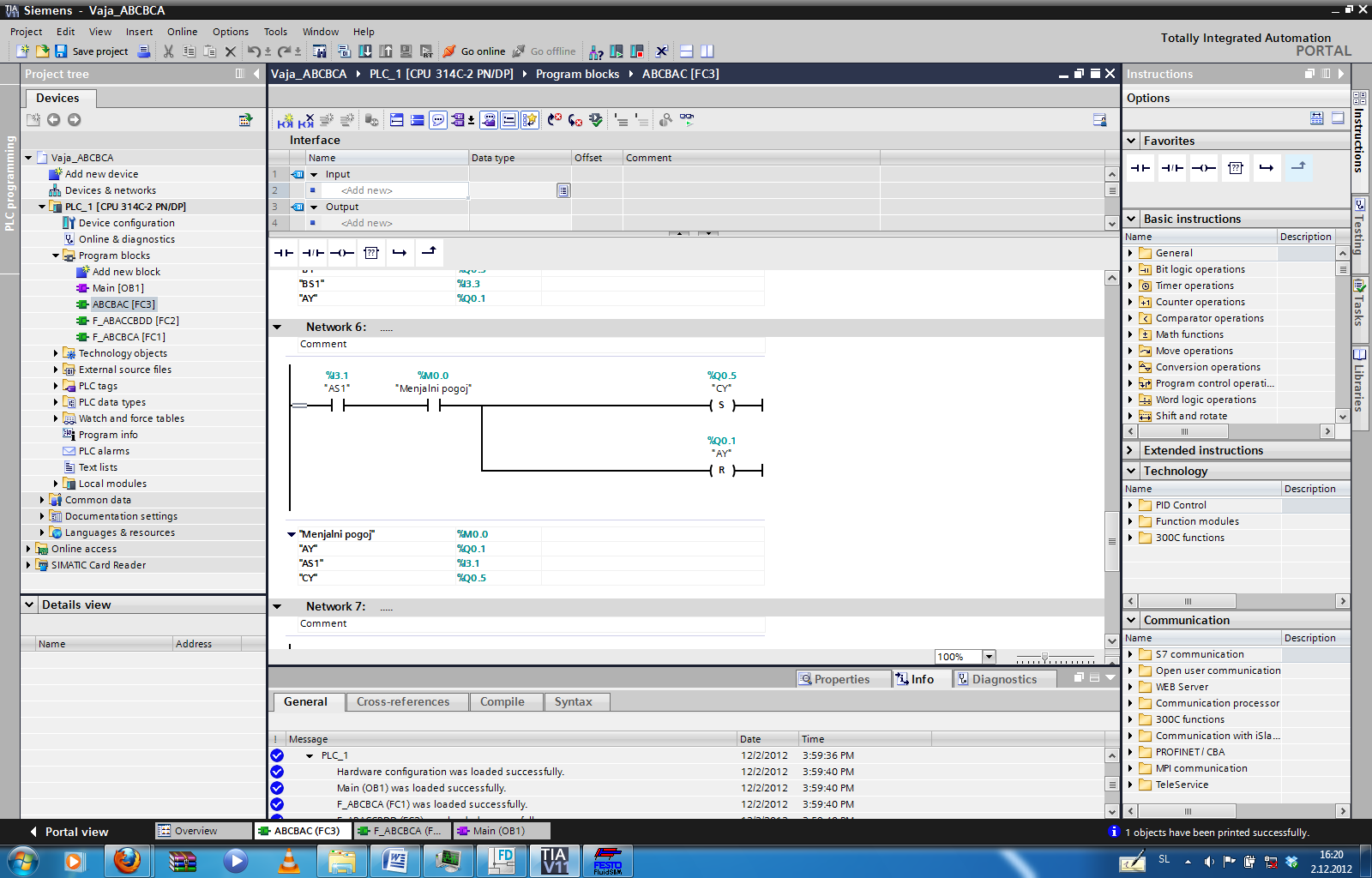


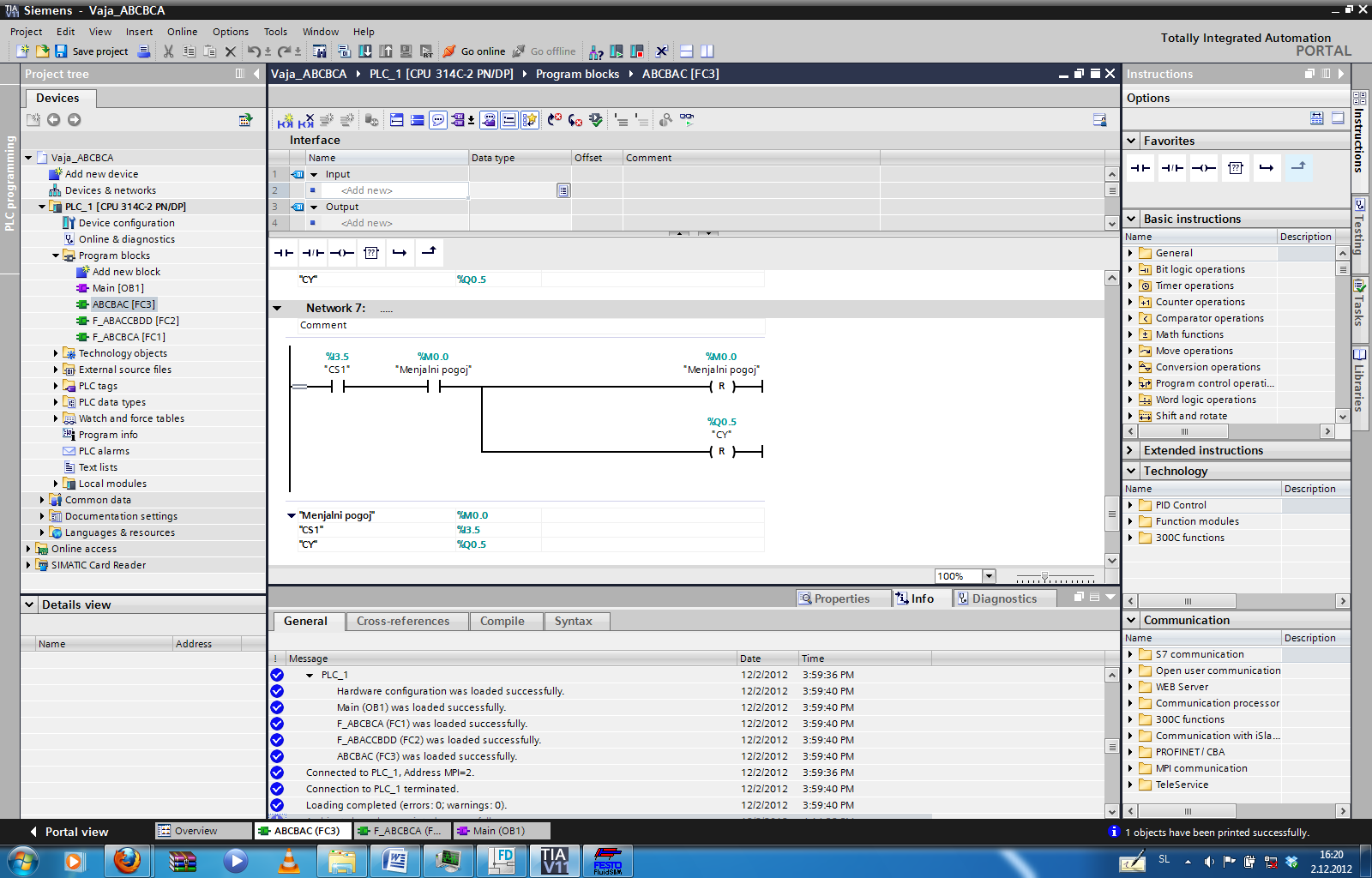












1. **Testiranje in simulacija**

Ko je program napisan, sledi prevajanje programa na podoben način kot pri konfiguraciji sistema; na levi strani okna z desno tipko miške kliknemo na **PLC\_1[CPU 314C-2 PN/DP]** ter poženemo prevajanje **Compile -> All**. Če program nima napak, ga lahko testiramo tako, da ga naložimo na krmilnik ali pa ga testiramo v simulatorju.

Pri simulacije se program naloži v virtualni krmilnik, začetek simulacije pa sprožimo tako, da na levi strani okna z desno tipko miške kliknemo na **PLC\_1[CPU 314C-2 PN/DP]** ter poženemo **Start Simulation**. Po zagonu simulacije se torej najprej naloži virtualni krmilnik, v katerega je potem potrebno naložiti še naš program. Zato se nam hkrati odpre okno **Extended download to device**, kjer moramo enako kot pri nalaganju na realni krmilnik izbrati tip programirnega vmesnika **MPI**. Ko nam program javi, da je uspešno povezan z (virtualnim) krmilnikom, potrdimo nalaganje programa s pritiskom na gumb **Load**. Po opravljenem nalaganju programa odpremo glavni vmesnik virtualnega krmilnika ter v okno vključimo vhodne in izhodne priključke oziroma spremenljivke programa. Sledi vklop krmilnika ter testiranje delovanja programa.

