УНИВЕРЗИТЕТ У БАЊОЈ ЛУЦИ
UNIVERSITY OF BANJA LUKA
МАШИНСКИ ФАКУЛТЕТ
FACULTY OF MECHANICAL ENGINEERING

Бојан З. Кнежевић
Алмир Османовић

ПРОГРАМАБИЛНИ ЛОГИЧКИ КОНТРОЛЕРИ

ПРАКТИКУМ

Бања Лука, 2026.

УНИВЕРЗИТЕТ У БАЊОЈ ЛУЦИ
МАШИНСКИ ФАКУЛТЕТ БАЊА ЛУКА

Бојан З. Кнежевић
Алмир Османовић

ПРОГРАМАБИЛНИ ЛОГИЧКИ КОНТРОЛЕРИ

ПРАКТИКУМ

Бања Лука, 2026.

Аутори
Проф. др Бојан З. Кнежевић
Проф. др Алмир Османовић

Наслов
Програмабилни логички контролери – практикум

Издавачи
Универзитет у Бањој Луци, Универзитетски град, Булевар војводе Петра Бојовића 1А
Машински факултет Универзитета у Бањој Луци, Булевар војводе Степе Степановића 71

Рецезенти
Проф. др Срђан Савић, ванредни професор, Факултет техничких наука Нови Сад
Проф. др Јасмин Халиловић, ванредни професор, Машински факултет Тузла

Аутор прелома и графичке обраде
Проф. др Бојан З. Кнежевић

Графички дизајн
Проф. др Бојан З. Кнежевић

Лектор:
Наташа Бојевић Ћускић

Одобрено одлуком Сената Универзитета у Бањој Луци, број 02/04-3.2506-69/26 од 17.9.2026.
Бања Лука, 2026.

ISBN: 978-99997-45-44-4
DOI: https://doi.org/10.63356/978-99997-45-44-4
Програмабилни логички контролери – практикум © 2026, Б. З. Кнежевић, А. Османовић. Ово дјело је лиценцирано под међународном лиценцом CC BY-NC-ND 4.0.
Копија лиценце на линку: https://creativecommons.org/licenses/by-nc-nd/4.0/

CIP - Каталогизација у публикацији
Народна и универзитетска библиотека
Републике Српске, Бања Лука

004.384(075.8)8076)(0.034.2)

КНЕЖЕВИЋ, Бојан, 1982-
Програмабилни логички контролери [Elektronski izvor] : практикум / Бојан З. Кнежевић, Алмир Османовић. - Онлајн изд. - Ел. књига. - Бања Лука : Универзитет у Бањој Луци : Машински факултет, 2026

Системски захтјеви нису наведени. - Начин приступа (URL): https://blupress.unibl.org/mf/catalog/book/191. - Насл. са насл. екрана. - Опис извора дана 23.9.2026. - Ел. публикација у ПДФ формату опсега 123 стр. - Библиографија: стр. 121-123.

ISBN 978-99997-45-44-4

COBISS.RS-ID 144821505

САДРЖАЈ

1. Предговор 1
2. Опис хардверске конфигурације радне станице 2
2.1. Програмабилни логички контролер 4
2.1.1. Сабирница 6
2.1.2. Напојна јединица 7
2.1.3. Процесорски модул 9
2.1.4. Модул дигиталних улаза 10
2.1.5. Модул дигиталних излаза 12
2.1.6. Модул аналогних улаза и излаза 13
2.1.7. Модул брзог бројача 15
2.2. Фреквенцијски регулатор 17
2.3. Командна табла 20
2.4. Сензори 28
2.4.1. Индуктивни сензор 28
2.4.2. Капацитивни сензор 31
2.4.3. Оптички сензор 33
2.5. Трофазни асинхрони мотор 36
2.6. Чвориште за *CANopen* комуникацију 36
2.7. Заштитни уређај диференцијалне струје 37
3. Опис софтверске платформе 39
3.1. Адресе и промјенљиве 39
3.2. Типови промјенљивих 43
3.3. Конверзија промјенљивих 44
3.4. Програмска платформа *Unity Pro XL* 45
3.5. Програмски језици ПЛК-а Модикон М340 58
3.5.1. Структурирани текст (*ST*) 58
3.5.2. Листа инструкција (*IL*) 59
3.5.3. Секвенцијални функцијски дијаграм (*SFC*) 59
3.5.4. Љествичасти дијаграм (*Ladder*) 60
3.5.5. Функцијски блок дијаграм (*FBD*) 61

3.6. Опис блокова у програмском језику *FBD* 62
3.6.1. Функцијски блокови логичких операција 64
3.6.2. Функцијски блокови поређења 70
3.6.3. Тајмери 73
3.6.4. Бројачи 77
3.6.5. Функцијски блокови основних математичких операција 80
3.6.6. Функцијски блокови сложених математичких операција 82
3.6.7. Функцијски блокови за конверзију промјенљивих 84
3.7. Повезивање ПЛК-а и рачунара 85
3.7.1. Повезивање ПЛК-а и рачунара путем *USB* комуникације 86
3.7.2. Повезивање ПЛК-а и рачунара путем *Ethernet-a* 87
3.8. Отпремање програма на ПЛК 90
4. Управљање фреквенцијским регулатором 93
4.1. Управљање задавањем референце директно на фреквенцијском регулатору 94
4.2. Управљање задавањем референце фреквенцијском регулатору помоћу екстерног потенциометра 96
4.3. Управљање фреквенцијским регулатором путем дигиталних и аналогних улаза 96
4.3.1. Управљање фреквенцијским регулатором путем дигиталних улаза 96
4.3.2. Управљање фреквенцијским регулатором путем аналогних улаза 98
4.4. Управљање фреквенцијским регулатором помоћу *CANopen* комуникације 99
4.5. Повезивање програмабилног логичког контролера и фреквенцијског регулатора 100
4.5.1. Директно повезивање 100
4.5.2. Повезивање *CANopen* протоколом 101
5. Примјери коришћења радних станица 113
5.1. Припремни задатак 1 113
5.2. Припремни задатак 2 113
5.3. Припремни задатак 3 113
5.4. Припремни задатак 4 114
5.5. Припремни задатак 5 114
5.6. Припремни задатак 6 114

5.7. Припремни задатак 7 115
5.8. Припремни задатак 8 116
5.9. Управљање брзином асинхроног мотора помоћу *CANopen* протокола 116
5.10. Мјерење броја обртаја вратила мотора помоћу индуктивног сензора 117
6. Литература 121

1. ПРЕДГОВОР

Машински факултет Универзитета у Бањој Луци је кроз пројекат Ерасмус+ КА107 који се реализује са Техничким универзитетом у Келну, а уз пресудну и свесрдну помоћ и залагање проф. др Хасана Смајића, добио професионалну опрему из области индустријске аутоматизације, произвођача Шнајдер Електрик (енг. *Schneider Electric*). Опрема је добијена у компонентама које су искоришћене да се од њих реализују дидактичке апаратуре за наставу из предмета Програмабилни логички контролери али и у оквиру других предмета као што су Програмирање и примјена ПЛК и Аутоматизација и визуализација процеса. Добијена опрема повезана је у једну цјелину, односно формиране су пројектоване дидактичке апаратуре.

Да би се студенти лакше упознали са апаратурама као комплексним системима и били спремни за извођење вјежби у лабораторији, припремљен је овај практикум. Практикум даје детаљна упутства за извођење вјежби из предмета Програмабилни логички контролери који се изводи као обавезни предмет на првом циклусу студија на студијском програму мехатроника Машинског факултета Универзитета у Бањој Луци.

Осим Предговора на почетку и Литературе на крају, практикум је организован у пет глава.

У другој глави под називом „Опис хардверске конфигурације радне станице“ дат је детаљан опис свих компоненти које сачињавају радну станицу. Осим битнијих физичких карактеристика компоненти, објашњен је и начин повезивања компоненти са другим компатибилним уређајима. Друга глава подијељена је на седам поглавља како би практикум био прегледнији и лакши за коришћење.

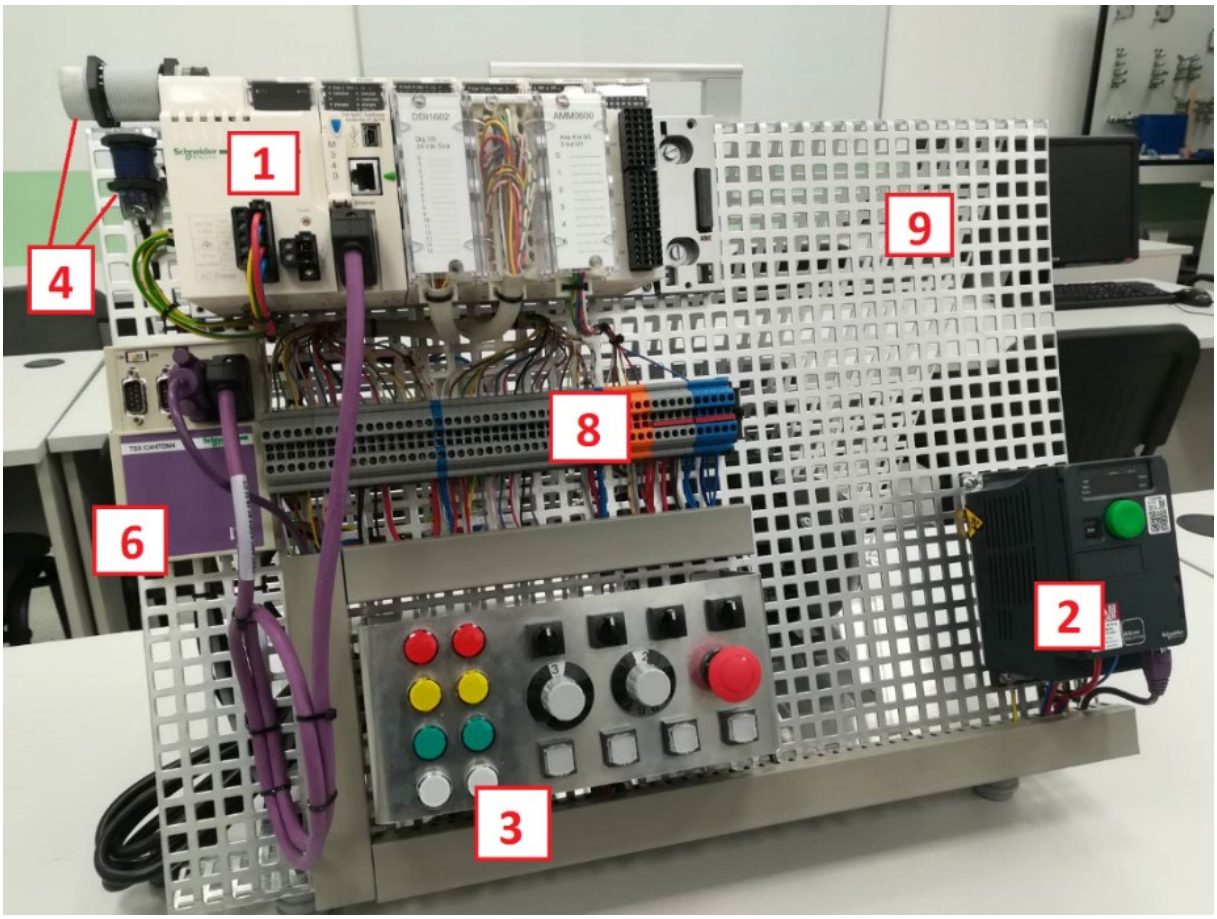
У трећој глави практикума под називом „Опис софтверске платформе“ дат је детаљан опис софтверског пакета *Unity Pro XL* као припадајућег развојног окружења. Познавањем могућности, односно спецификација софтвера за програмирање програмабилног логичког контролера, значајно се олакшавају поступци писања програма, те се на тај начин повећава и ефикасност инжењерског рада. Дат је преглед свих програмских језика, а детаљно је описан функцијски блок дијаграм. Седмо поглавље описује процедуру повезивања рачунара са инсталираним софтвером и ПЛК и то путем два могућа интерфејса.

Четврта глава описује фреквенцијски регулатор и начине повезивања и управљања са погоном који је саставни дио апаратуре.

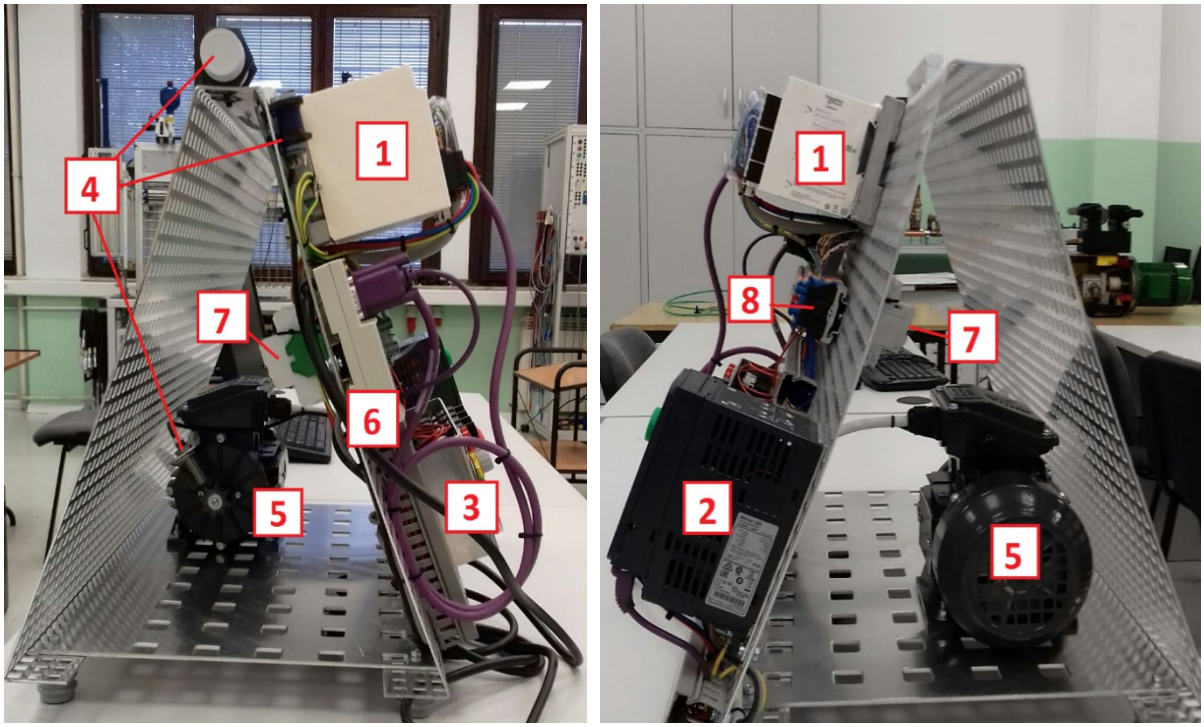
У петој глави дати су примјери који служе за увођење студената у вјежбе и приказују функционалност цјелокупног система.

2. ОПИС ХАРДВЕРСКЕ КОНФИГУРАЦИЈЕ РАДНЕ СТАНИЦЕ

Радне станице се састоје од компоненти које су монтиране на алуминијумску конструкцију са ручком за лакше преношење. Све компоненте су монтиране са једне стране радне станице док је друга страна слободна за даља проширења. На фотографијама са Сл. 1а и Сл. 1б приказана је радна станица са нумерисаним компонентама које ће бити детаљно описане.



Слика 1а – Изглед радне станице



Слика 1б – Изглед радне станице

Компоненте које чине радну станицу су (Сл. 1а и Сл. 1б):

- програмабилни логички контролер са дигиталним и аналогним улазно/излазним модулима (1),
- фреквенцијски регулатор (2),
- командна табла (3),
- сензори (4),
- трофазни асинхрони мотор (5),
- чвориште за *CANopen* комуникацију (6),
- заштитни уређај диференцијалне струје (7),
- редне стезаљке (8),
- алуминијумско кућиште (9).

2.1. Програмабилни логички контролер

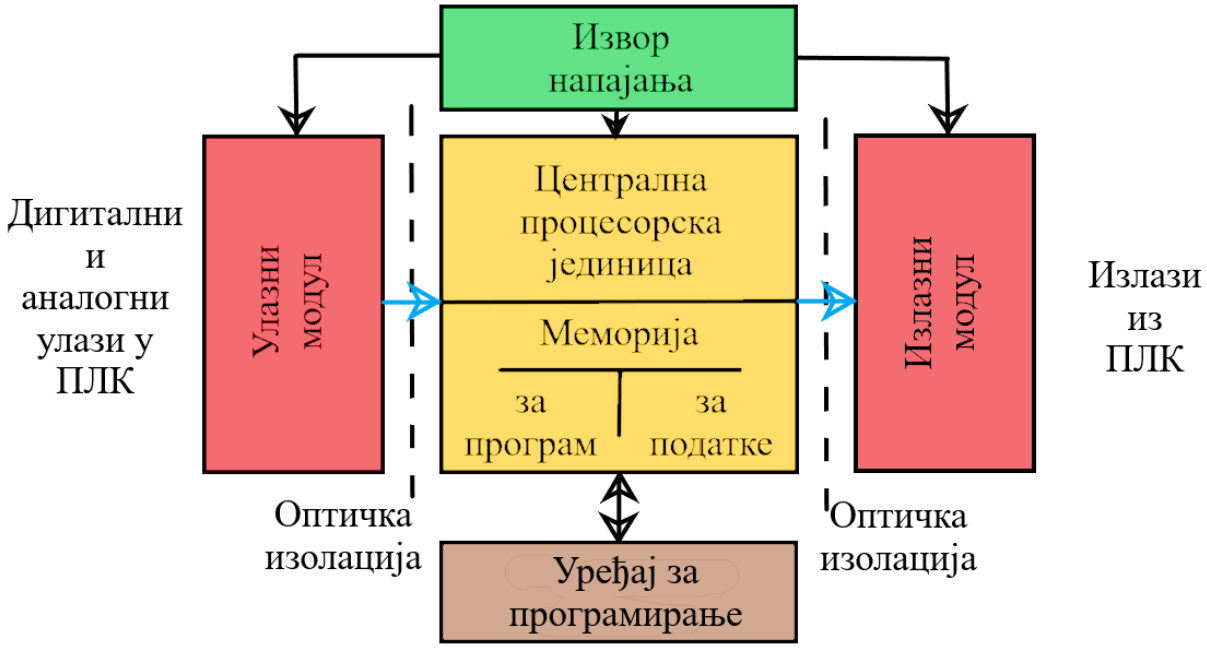
Прије почетка описивања конкретног програмабилног логичког контролера, потребно је најприје дати дефиницију као и улогу програмабилних логичких контролера у аутоматизованим системима.

Програмабилни логички контролер (ПЛК) представља специјализовани рачунар који се користи за контролу машина и процеса. ПЛК користи програмабилну меморију у коју се смјештају одређене инструкције (наредбе), као и специјалне функције које се користе за пребројавање, укључивање/искључивање, мјерење времена, секвенцирање, те аритметичко-логичке функције. Сама потреба за развијањем ПЛК-а почела је из разлога комплексности индустријских аутоматизованих система. Развијањем ПЛК-а знатно се смањила хардверска конфигурација у смислу ожичења, и повећала флексибилност самог система којим се управља, на начин да се врло лако врше измјене улазно/излазних уређаја и измјене програма рада система. Веома је важно нагласити да се ПЛК врло ефикасно користи у индустријским условима (повишене температуре, прашина, радиоактивне материје итд.).

Развој програмабилних логичких контролера почео је шездесетих година двадесетог вијека, те је задатак био да се развије уређај који нуди исту функционалност као релејни управљачки систем. Предности система који користе програмабилне логичке контролере су:

- флексибилност,
- брже вријеме одзива,
- једноставније ожичење система,
- смањење покретних дијелова (систем је на вишем степену интеграције),
- модуларни дизајн (огледа се у већој једноставности отклањања кварова, те додавања, односно проширивања система),
- лакше управљање сложенијим системима,
- софистициране наредбе и,
- нижа цијена.

Да би се лакше разумјела функција програмабилног логичког контролера у системима, потребно је најприје ПЛК подијелити на цјелине, што је приказано на Сл. 2.



Слика 2 – Блоковска структура ПЛК-а

На улазни модул доводе се дигитални и аналогни сигнали (сигнали са релеја, тастера, прекидача, сензора који дају дигиталне или аналогне сигнале), који се прилагођавају централној процесорској јединици и шаљу ка њој уз галванско раздвајање.

Централна процесорска јединица (ЦПЈ) представља основну управљачку цјелину програмабилног логичког контролера. Састоји се од микропроцесора који имплементира логику, те врши контролу и комуникацију између свих модула који су у саставу ПЛК-а. Микропроцесор врши управљање излазним уређајима извршавајући кориснички програм, а на основу добијених улазних информација о стању система.

Излазни модул има задатак да излазне сигнале микропроцесор прилагоди за даље коришћење у систему (покретање релеја, електромагнетних вентила, свјетлосних индикатора, мотора итд.) уз галванско раздвајање.

Као уређај за програмирање, најчешће се користи рачунар (енгл. *PC*[1]). Сваки произвођач ПЛК-а користи своје програмско окружење за писање програма као и конфигурацију ПЛК и комуникационе мреже. Софтвер пружа могућности кориснику да

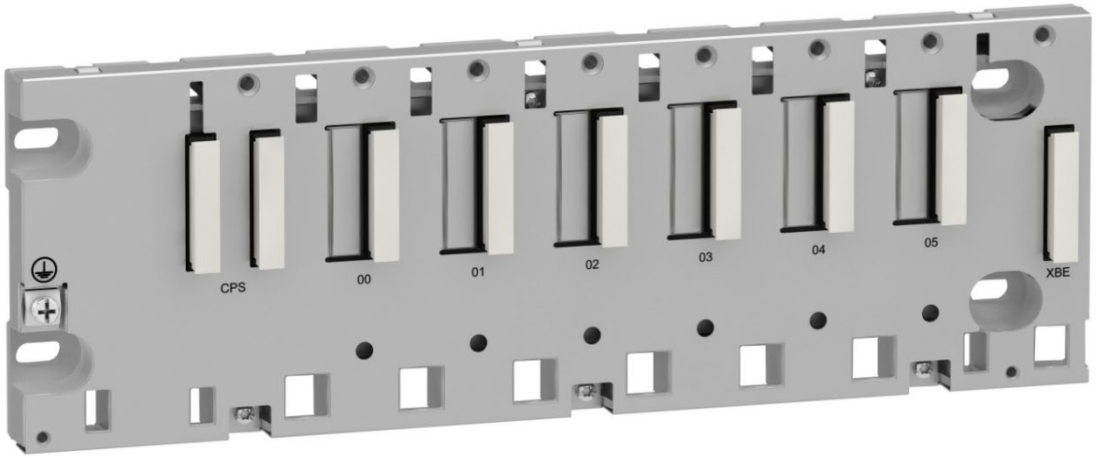
[1] *PC* - *Personal Computer*, у случају рачунара који је под *Windows* оперативним системом.

креира, мијења или сачува написани програм, као и да га путем комуникације ПЛК-*PC* пребаци на програмабилни логички контролер.

Након базног упознавања са појмовима који су у вези са програмабилним логичким контролером, може се приступити описивању конкретног ПЛК-а произвођача Шнајдер Електрик.

2.1.1. Сабирница

Сабирница или рек (енгл. *Rack*) представља плочу на којој се формира ПЛК монтирањем свих потребних модула. Број и врста модула који формирају ПЛК зависи од конкретне примјене. Сабирнице произвођача Шнајдер Електрик се састоје од четири, шест, осам, десет или дванаест лежишта, односно слотова (енгл. *Slot*). Конкретан модел сабирнице која је уграђена на радној станици је *BMXXBP0600* (Сл. 3). Сабирница се на алуминијумско кућиште повезује путем вијака (четири M5 вијка). Потрошња сабирнице износи 1.5 W.



Слика 3 – Сабирница BMXXBP0600

На конекторе *CPS* (Сл. 3) уграђује се модул за напајање, док се конектор *XBE* користи за даље проширивање сабирнице другим сабирницама. Распоред модула на сабирници која је уграђена на радној станици, приказан је на Сл. 4.

Слика 4 – Распоред постављања модула на сабирници

2.1.2. Напојна јединица

Напојна јединица (Сл. 5) има задатак да снабдијева све модуле прикључене на сабирницу једносмјерним напоном. У мањим системима који користе ПЛК, напојна јединица може да врши напајање и неких других уређаја које систем користи. Конкретан модел напојне јединице уграђене на сабирницу је *BMXCPS2000*, произвођача Шнајдер Електрик. Изглед напојног модула дат је на Сл. 5.

Напонска јединица претвара наизмјенични улазни напон у једносмјерни, односно на излазу даје једносмјерни напон од 24 V. Карактеристике напојног модула су:

- улазни напон од 100 до 240 V,
- минималан улазни напон 85 V,
- максималан улазни напон 264 V,
- фреквенција напајања 50 и 60 Hz,
- максимална снага дисипације 8.5 W,
- струја од 0.45 A и снага од 10.8 W при излазном једносмјерном напону од 24 V за напајање сензора,

- струја од 0.7 A и снага од 16.8 W при излазном једносмјерном напону од 24 V за напајање улазно/излазних модула и централне процесорске јединице,
- струја од 2.5 A и снага од 8.3 W при излазном једносмјерном напону од 3.3 V за напајање логике улазно/излазних модула,
- радна температура од 0 до 60°C.

Слика 5 – Напојна јединица BMXCPS2000

На предњој страни напојног модула налазе се индикатори са зеленим лампицама које имају улогу сигнализације. Једна сигнална лампица свјетлосним сигналом приказује да ли модул ради коректно, односно да ли је у оперативном стању и да ли је номиналан напон на сабирници (зелени свјетлосни сигнал поред текста OK), док друга лампица потврђује адекватну вриједност једносмјерног напона којим се напајају сензори (24 V). Са Сл. 5 може се уочити да напојна јединица посједује два конектора као и тастер *RESET*. Лијеви конектор има пет прикључака, чија је намјена сљедећа:

- прикључци 24 V и 0 V користе за напајање свих потрошача који користе стандардни једносмјерни напон од 24 V (сензори и неки модули ПЛК)
- прикључак *PE* представља уземљење,

- прикључци *N* и *L* представљају нулу и фазу и на њих се доводи наизмјенични монофазни напон који се доводи са заштитног уређаја диференцијалне струје.

Десни конектор представља прекидач који може да да информацију програмабилном логичком контролеру уколико је дошло грешке на напојном модулу. При улазном монофазном наизмјеничном напону од 100 до 240 V, напојни модул врши стабилизацију излазног напона на 24 V.

2.1.3. Процесорски модул

Процесорски модул са централном процесорском јединицом има задатак да управља свим модулима и свим процесима који се дешавају у програмабилном логичком контролеру, да врши комуникацију са свим модулима, комуникацију са другим ПЛК-има и уређајима помоћу којих се врши програмирање. Да би се успјешно извеле све аритметичко-логичке операције, те добијени резултати чували и користили за даљи рад ПЛК-а, потребно је да у склопу процесорске јединице постоји меморија. За складиштење података углавном се користи *EEPROM*[2], док се као радна меморија користи *RAM*[3]. Процесорски модул који се користи на радним станицама је модел *BMXP3420302* произвођача Шнајдер Електрик (Сл. 6).

Позиција процесорског модула на сабирници је одмах након напојне јединице, односно на нултом лежишту. Уграђени процесорски модул подржава до 1024 улазно/излазна дигитална уређаја, као и до 256 аналогних улазно/излазних уређаја. Веома је важно нагласити да модул посједује три врсте комуникације и то:

- *USB* комуникација (пренос података до 12Mbit/s),
- *CANopen*[4] комуникацију (двије увијене парице са заштитом, пренос података до 1Mbit/s),
- *Ethernet TCP/IP RJ45*.

---

[2] *EEPROM* представља меморију за трајно складиштење података који се могу брисати и поново уписивати. Скраћеница *EEPROM* настала је од енглеских ријечи *Electrical Erasable Read-Only Memory.* Предност *EEPROM* огледа се у технологији која пружа могућност кориснику да брише, односно мијења само дијелове података записане у меморији. Битна карактеристика *EEPROM* је да подаци остају сачувани и при нестанку напајања меморије.

[3] *RAM* (енгл. *Random Accsess Memory*) представља меморију која се користи за писање и читање података. За разлику од *EEPROM*-а, *RAM* је меморија чији се подаци бришу када дође до прекида напајања- Због велике брзине приступа подацима, односно њиховог брисања и уписивања, *RAM* се користи као радна меморија.

[4] *CANopen* је комуникациони протокол. Базиран је на *CAN* (*Controll Area Network*) протоколу. Настао је као потреба комуникације у уграђеним системима. Комуникација се врши између мастера и једног или више *Slave* уређаја.

Слика 6 – Процесорски модул BMXP3420302

Коришћењем *CANopen* комуникације процесорски модул у улози мастера може да оствари комуникацију са 63 слејв (енг. *Slave*) уређаја. Као радна меморија користи се *RAM* величине 4 MB, док се за складиштење података користи интерна меморија величине 256 kB. Процесорски модул подржава екстерну меморију, односно посједује лежиште за екстерну меморијску картицу типа СД. За напајање се користи једносмјерни напон од 24 V, који се доводи са сабирнице. Модул посједује десет лампица (четири зелене, пет црвених и једна жута) које се користе са сигнализацију (Сл. 6).

2.1.4. Модул дигиталних улаза

На улазни модул спојени су сензори и сигнали других уређаја који ПЛК-у дају информације о стању система којим се управља. Код модуларних система, као што је систем који се налази на радној станици, могуће је прикључити више улазних јединица, а број могућих улаза (дигиталних/аналогних) зависи од процесорског модула.

Модул дигиталних улаза уграђен на радну станицу је *BMXDDI1602*, произвођача Шнајдер Електрик (Сл. 7).

Слика 7 – Модул дигиталних улаза BMXDDI1602

Карактеристике модула дигиталних улаза *BMXDDI1602* су:

- напон улаза 24 V,
- струја улаза 3.5 mA,
- напон напајања од 19 до 30 V,
- регистровање логичке јединице за улазни напон већи од 11 V,
- регистровање логичке јединице за улазну струју већу од 2 mA,
- регистровање логичке нуле за улазни напон мањи од 5 V,
- регистровање логичке нуле за улазну струју мању од 1.5 mA,
- снага дисипације 2.5 W,

- потрошња струје од 76 mA за 3.3 V једносмјерног напона,
- заштита од инверзног поларитета,
- радна температура од 0 до 60°C.

Позиција модула дигиталних улаза на сабирници је одмах поред процесорског модула (прво лежиште). Дигитални модул посједује 16 дигиталних улаза. Да би модул правилно функционисао потребно је довести екстерни једносмјерни напон од 24 V са конектора на напојној јединици.

На модул дигиталних улаза повезују се компоненте које имају дигиталне (бинарне) излазе, а то су углавном прекидачи, тастери (који се налазе на контролној табли), поједини сензори, итд.

При повезивању дигиталних улаза на улазни модул, искоришћено је укупно дванаест улаза, од чега су:

- четири прекидача,
- четири тастера са контактима NO,
- три сензора (индуктивни, капацитивни и оптички),
- нужни стоп тастер са контактима NC.

Сигналним лампицама се може пратити тренутно стање улазних сигнала (логичка 0 или логичка 1), а што може бити корисно за дијагностику рада прикључених сензора.

2.1.5. Модул дигиталних излаза

Задатак модула дигиталних излаза је да обезбиједи покретање одређених спољашњих уређаја. Модул дигиталних излаза уграђен у радне станице је *BMXDDO1602* произвођача Шнајдер Електрик (Сл. 8). Положај модула на сабирници је одмах након модула дигиталних улаза (друго лежиште).

Карактеристике модула дигиталних излаза *BMXDDO1602* су:

- напон излаза од 19 до 30 V,
- струја излаза 0.5 A,
- струја по једном каналу 0.625 A,
- максимална струја по модулу 10 A,
- снага дисипације 4 W,
- потрошња струје 79 mA при 3.3 V,
- радна температура од 0 до 60°C.

Модул дигиталних излаза посједује шеснаест дигиталних излаза. На врху модула налази се индикатор са сигналним лампицама. Као и код модула дигиталних улаза, индикатори могу да помогну у дијагностици рада прикључених компоненти. Могуће је лоцирати грешку, да ли је у програму, односно ПЛК-у, или је у прикљученој компоненти. Од укупно шеснаест дигиталних излаза, искоришћено је десет. Излази су повезани са сигналним лампицама на контролној табли (осам укупно), а два излаза су спојена на дигиталне улазе фреквенцијског регулатора.

Слика 8 – Модул дигиталних излаза BMXDDO1602

2.1.6. Модул аналогних улаза и излаза

Како сам назив каже, програмабилни логички контролери у основи су намијењени раду са логичким функцијама. Логичке функције као промјенљиве имају бинарне вриједности, нула или један. Развојем аутоматизације појавила се потреба за мјерењем величина, у употребу су ушли сензори са аналогним излазима, а ПЛК-и су добили аналогне улазе као и излазе за пропорционално управљање.

Аналогни сигнали могу бити напонски и струјни и најчешће су у стандардним индустријским опсезима од 0 до 10 V, односно од 0 до 20 mA и од 4 до 20 mA. Опсег од 4 до 20 mA има предност јер је њиме могуће детектовати прекид проводника или квар прикљученог сензора или актуатора. Модел уграђеног модула аналогних улаза и излаза произвођача Шнајдер Електрик је *BMXAMM0600* (Сл. 9).

Слика 9 – Модул аналогних улаза и излаза BMXAMM0600

Карактеристике модула аналогних улаза и излаза *BMXAMM0600* су:

- четири аналогна улаза,
- улазни сигнал може бити
  напонски:
  - од -10 до 10 V (резолуција од 14 бита),
  - од 0 до 10 V (резолуција од 13 бита),
  - од 0 до 5 V (резолуција од 12 бита) и

од 1 до 5 V (резолуција од 12 бита)

или струјни:

од 0 до 20 mA (резолуција од 12 бита) и

од 4 до 20 mA (резолуција од 12 бита),

- два аналогна излаза,
- излазни сигнал може бити

  напонски:

  од -10 до 10 V (резолуција од 12 бита)

  или струјни:

  од 0 до 20 mA (резолуција од 11 бита) и

  од 4 до 20 mA (резолуција од 11 бита)),
- максимална потрошња снаге је 3.2 W за 24 V једносмјерног напона,
- радна температура од 0 до 60°C.

Модул посједује четири аналогна улаза и два аналогна излаза. На аналогне улазе прикључени су потенциометри са контролне табле Један од излаза повезан је са аналогним улазом фреквенцијског регулатора, а други излаз је слободан.

2.1.7. Модул брзог бројача

Позиција модула брзог бројача налази се након модула аналогних улаза и излаза (четврто лежиште). Уграђени модул брзог бројача је двоканални, произвођача Шнајдер Електрик, а тачна ознака модела је *BMXEHC0200* (Сл. 10).

Карактеристике модула брзог бројача *BMXEHC0200* су:

- два канала,
- максимална фреквенција бројања је 60000 Hz,
- шест улаза (три брза и три помоћна улаза),
- једносмјерни улазни напон 24 V,
- два излаза,
- излазни напон 24 V,
- више функција бројања (бројање кроз петљу, бројање уназад, мјерење временских периода, бројање догађаја, мјерење фреквенције),
- вријеме циклуса 1 ms,
- максималан улазни напон 30 V,

- улазна струја 2 mA на 11 V,
- логичка јединица при улазном напону од 11 до 30 V или при струји већој од 6 mA,
- логичка нула при улазном напону мањем од 5 V и струји мањој од 1.5 mA,
- максимална излазна струја 2 A по модулу и 0.5 A по излазу,
- минималан излазни напон 19.2 V,
- максималан излазни напон 30 V,
- радна температура од 0 до 60°C.

Слика 10 – Модул брзог бројача BMXEHC0200

Користи се у случајевима када се вриједност улазног сигнала мијења великом брзином, односно када је програмски скен превише дуготрајан да изброји све промјене на улазу, због обима програма. На радној станици нису спојени уређају на модул брзог бројача. Са предње стране модула уочавају се три конектора (Сл. 10). Прва два конектора означена са *CH0* и *CH1* имају по шеснаест прикључака и користе се за повезивање

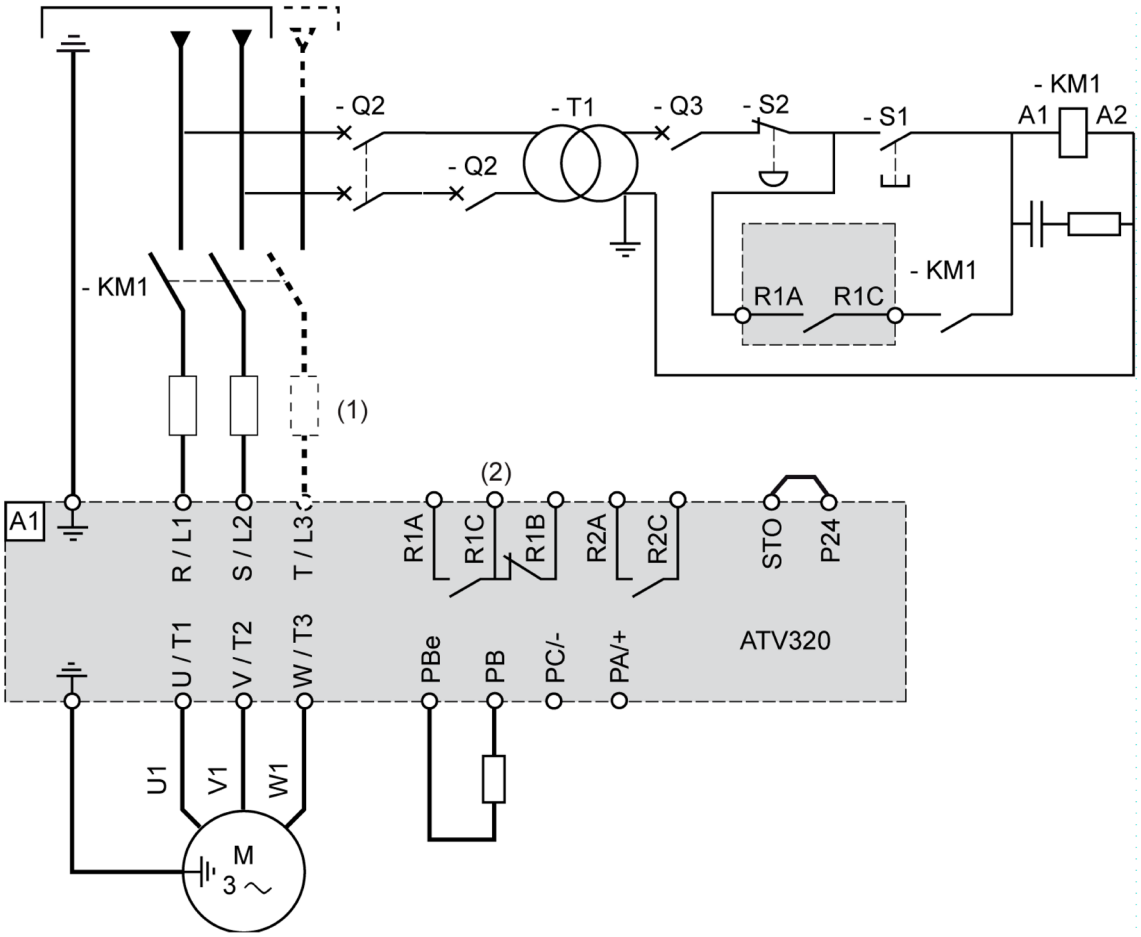
сензора на бројаче (канале) 0 и 1, док се трећи конектор користи за повезивање помоћних улаза и за напајање сензора (посједује десет прикључака). На модул брзог бројача могу се прикључити двожични и трожични сензори, односно детектори присуства, као и инкрементални енкодери. Напајање модула брзог бројача врши се кроз рек.

2.2. Фреквенцијски регулатор

Фреквенцијски регулатор се користи за управљање бројем обртаја вратила електромотора. Модел фреквенцијског регулатора уграђен на радној станици је *ATV320U02M2C* (Сл. 11), произвођача Шнајдер Електрик. Напајање фреквенцијског регулатора врши се довођењем наизмјеничног напона 230 V на конекторе *RL1* и *SL2*. Мотор који је прикључен на фреквенцијски регулатор је асинхрони трофазни мотор снаге 0.18 kW, тип *Drehstrom – Asynchron Motor VDE 0530*. Мотор је прикључен на конекторе *UT1,VT2* и *WT3* фреквенцијског регулатора. Шема повезивања енергетског дијела фреквенцијског регулатора приказана је на Сл. 12.

Слика 11 – Фреквенцијски регулатор ATV320U02M2C

За повезивање са другим уређајима фреквенцијски регулатор посједује два комуникациона протокола, односно *Modbus* и *CANopen* протокол. На радној станици *CANopen* протокол је искоришћен за комуникацију са програмабилним логичким контролером.

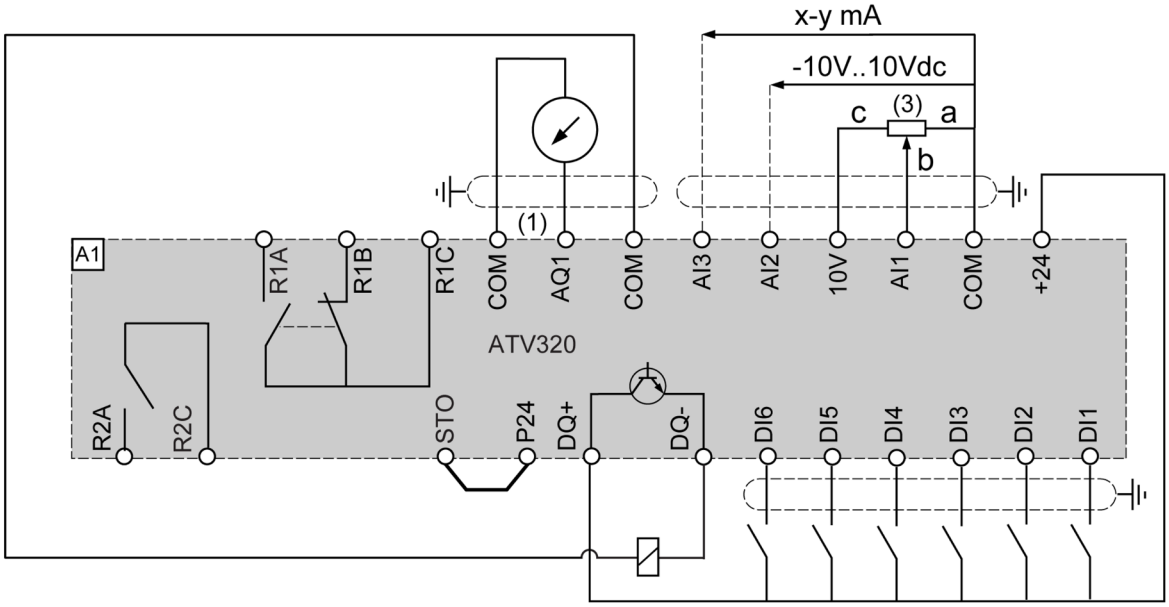


Слика 12 – Шема повезивања фреквенцијског регулатора

За остваривање спољњег управљања фреквенцијски регулатор има:

- седам дигиталних улаза,
- три дигитална излаза,
- три аналогна улаза и
- један аналогни излаз.

Шема прикључења улаза и излаза фреквенцијског регулатора приказана је на Сл. 13.



Слика 13 – Шема прикључења улаза и излаза фреквенцијског регулатора

Дигитални улази *DI1, DI2, DI3, DI4* и *DI6* користе се за довођење сигнала са сензора са дигиталним излазима (једносмјерни напон од 24 V).

Дигитални улаз *DI5* је брзи дигитални улаз за сигнале фреквенције до 30 kHz, 24 V.

Дигитални улаз *STO* (енгл. *Safe Torque Off*) користи се као заштитни улаз који искључује напајање мотора.

Аналогни улази означени су са *AI1, AI2* и *AI3* (Сл. 13).

Аналогни улаз *AI1* користи се за повезивање екстерног потенциометра на фреквенцијски регулатор. Клизач потенциометра повезује се на прикључак *AI1*, док се друга два краја потенциометра повезују на прикључке 10 V и COM (Сл. 13).

Аналогни улаз *AI2* је напонски аналогни улаз за опсег сигнала од -10 до 10 V у односу на COM (Сл. 13).

Аналогни улаз *AI3* је струјни аналогни улаз за опсег сигнала од 0 до 20 mA или од 4 до 20 mA.

Карактеристике фреквенцијског регулатора *ATV320U02M2C* су:

- улазни напон од 200 до 240 V,
- максимална улазна струја 3.4 A

- максимална излазна струја 6.9 A,
- максималан излазни напон 240 V,
- максимална излазна фреквенција 599 Hz,
- намијењен за мотор номиналне снаге 0.18 kW,
- представља слејв уређај за *CANopen* комуникацију,
- могућност управљања мотором у сва четири квадранта,
- посједује рампе за убрзавање и успоравање.

Фреквенцијски регулатор ради на принципу импулсно-ширинске модулације (енгл. *Pulse Width Modulation*) која му омогућава да прикљученим мотором управља промјеном напона и фреквенције због чега се често користи термин *VFC* (енгл. *Voltage Frequency Control*).

Од фреквенције напона напајања зависи број обртаја мотора. Принцип управљања фреквенцијом који се код асинхроних мотора користи подразумијева и промјену напона и то тако да однос напона и фреквенције напајања мотора увијек остаје исти. Корисник или ПЛК задају фреквенцију, а напон одређује сам регулатор.

Постоји више начина задавања референтне фреквенције напона напајања мотора на коришћеном фреквенцијском регулатору, а то су:

- локално управљање на фреквенцијском регулатору задавањем фреквенције,
- управљање путем серијске комуникације,
- управљање посредством *CANopen* протокола,
- управљање путем дигиталних и/или аналогних улаза фреквенцијског регулатора.

Детаљни описи задавања референтне фреквенције напона напајања мотора описани су у четвртој глави, чији је назив „Управљање фреквенцијским регулатором“.

2.3. Командна табла

Командна табла има улогу да врши импровизацију реалног окружења, односно реалног система којим се управља. Изглед контролне табле приказан је на Сл. 14.

Командна табла се састоји од:

- четири прекидача,
- четири тастера,
- осам сигналних лампица,
- два потенциометра и

- једног гљивастог прекидача, нужни СТОП.

Тастери и прекидачи се могу посматрати као сензори или уређаји за задавање команди са дигиталним излазима и то са нормално отвореним контактима (енгл. *NO*). Њиховим активирањем на дигиталне улазе ПЛК-а доводе се једносмјерни напони од 24 V, што представља логичку јединицу. Тастери имају и свјетлосну индикацију бијеле боје која је повезана тако да је све вријеме укључена док је радна станица прикључена на напајање.

Уколико се ради о прекидачима, они задржавају своју вриједност све док се не врате у почетно стање (искључено), док тастери дају логичку јединицу само за вријеме када су притиснути.

Нужни СТОП је прекидач са нормално затвореним контактима (енгл. *NC*). Нормално затворен контакт подразумијева да ће давати на излазу логичку јединицу док год не дође до његовог активирања.

Лампице имају улогу свјетлосних индикатора. Прикључене су на дигиталне излазе ПЛК-а. Састоје се од носача и свјетлећег дијела који је у овом случају *LED* диода.

На контролној табли налазе се два потенциометра. Улога потенциометара је да остваре симулацију промјене неке аналогне величине која се доводи на аналогне улазе ПЛК-а ради мјерења.

Слика 14 – Командна табла

У Табели 1 налази се списак компоненти које се налазе на контролној табли.

Табела 1 – Списак коришћених компоненти у саставу контролне табле

Назив компоненте, енгл.	Модел	Опис компоненте	Количина
Red pilot light head	*ZB6AV4*	Црвени носач *LED* диоде	2
Red light block	*ZB6EB4B*	Црвена *LED* диода	2
Green pilot light head	*ZB6AV3*	Зелени носач *LED* диоде	2
Green light block	*ZB6EB3B*	Зелена *LED* диода	2
Yellow pilot light head	*ZB6AV5*	Жути носач *LED* диоде	2
Yellow light block	*ZB6EB5B*	Жута *LED* диода	2
White pilot light head	*ZB6AV1*	Бијели носач *LED* диоде	2
White light block	*ZB6EB1B*	Бијела *LED* диода	6
White pushbutton	*ZB6DE1*	Бијели тастер	4
Body fixing collar	*ZB6Y009*	Носач *NO* и *NC* контаката	8
Single contact block NO	*ZB6E1B*	Контактни тастер *NO*	8
Single contact block NC	*ZB6E2B*	Контактни тастер *NC*	1
Black square selector switch	*ZB6CD25*	Троположајни прекидач	4
Emergency STOP pushbutton	*XB6AS8349B*	Гљивасти прекидач за нужни стоп	1
Potentiometar 2.2kOhm	*PE30L0FR222MAB*	Потенциометар 2.2 kΩ	2

Све компоненте из Таб. 1 су произвођача Шнајдер Електрик осим потенциометара. Са Сл. 14 види се распоред компоненти на контролној табли. Монтажом *LED* диода (Сл. 15) у тијело лампице (Сл. 16) формирају се лампице које се користе као свјетлосни индикатори.

Слика 15 – LED диоде: црвене боје ZB6EB4B, зелене боје ZB6EB3B, жуте боје ZB6EB5B, бијеле боје ZB6EB1B

а) б)

в) г)

Слика 16 – Тијело лампице: а) Црвено тијело лампице ZB6AV4, б) Зелено тијело лампице ZB6AV3, в) Жуто тијело лампице ZB6AV5, г) Бијело тијело лампице ZB6AV1са навртком

Из Таб. 1 уочава се да је уграђено шест бијелих *LED* диода. Двије бијеле *LED* диоде искоришћене су за лампице, док су четири *LED* диоде искоришћене за свјетлосну индикацију на тастерима. Као што је претходно описано, прекидачи и тастери су типа *NO*, док је тастер нужни стоп типа *NC*. Прекидач, тастер и нужни СТОП приказани су на Сл. 17. Све три компоненте су модуларне и састоје се од носача контаката и контаката засебно што се види из Таб. 1 и на Сл. 18.

а)

б)

в)

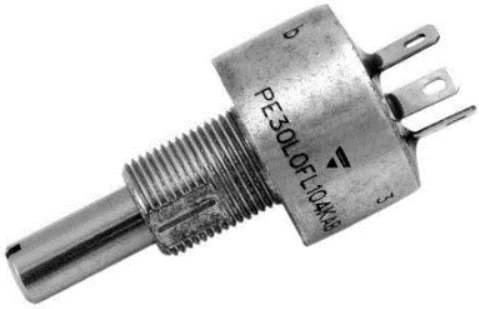
Слика 17 – Улазне компоненте на контролној табли: а) Троположајни прекидач ZB6CD25, б) Бијели тастер ZB6DE1, в) Нужни СТОП тастер XB6AS8349B

а)

б)

Слика 18 – Компоненте са контролне табле: а) Нормално отворени контакти ZB6E1B, односно нормално затворени контакти ZB6E2B, б) Носач отворених и затворених контаката ZB6Y009

На контролној табли уграђена су два потенциометра чија је ознака модела *PE30L0FR222MAB* (Сл. 19). Потенциометри симулирају мјерење у процесу. Задатак им је да омогуће промјену напона у опсегу од 0 до 10 V који је стандардан за напонске аналогне улазе ПЛК-а. С обзиром да на радним станицама постоји једносмјерни напон једино у вриједности од 24 V, потенциометри не могу бити директно прикључени на ПЛК.



Слика 19 – Потенциометар PE30L0FR222MAB

Да би се из напона напајања добио неопходни напонски опсег, коришћен је напонски раздјелник.

Карактеристике потенциометра *PE30L0FR222MAB* су:

- отпорност 2.2 kΩ,
- максимална снага 3 W (на 70°C),
- максималан радни напон 81.2 V,
- максимална струја 37 mA.

Из карактеристика потенциометра, закључује се да се потенциометар може повезати на једносмјерни напон од 24 V. На Сл. 20 приказана је шема повезивања потенциометра са ПЛК-ом.

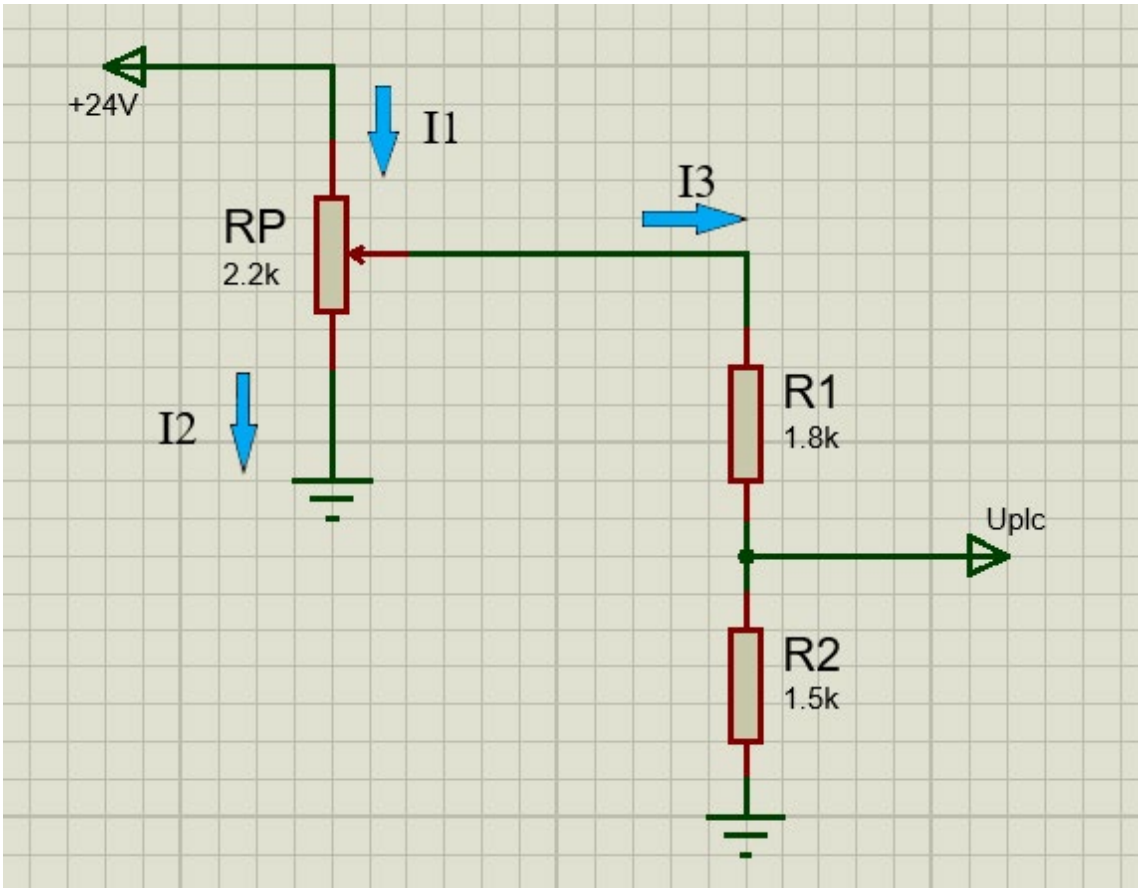
Да би се одредили параметри отпорника у колу тако да се на излазу добије жељени напон потребно је израчунати све карактеристичне величине. Однос отпорности отпорника напонског раздјелника пропорционалан је односу улазног и излазног напона:

$$\frac{R_1}{R_2} + 1 = \frac{24}{10}. \tag{1}$$

Користећи стандардне отпорнике није могуће добити тражени омјер из (1) али је могуће приближан. Усвајамо отпорнике стандардних вриједности: R_1=1.8 kΩ и R_2=1.5 kΩ.

Укупна, односно еквивалентна отпорност кола (Сл. 20) је:

$$R_e = \frac{R_p(R_1+R_2)}{R_p+R_1+R_2} = \frac{2.2\cdot 3.3}{2.2+3.3} = \frac{7.26}{5.5} = 1.32\ k\Omega. \tag{2}$$



Слика 20 – Формирани напонски раздјелник

Струја I_1 рачуна се као количник напона и еквивалентне отпорности:

$$I_1 = \frac{24\,V}{1320\,\Omega} = 0.01818\,A = 18.18\,mA. \tag{3}$$

Струја I_2 рачуна се као:

$$I_2 = \frac{24\,V}{2200\,\Omega} = 0.0109\,A = 10.9\,mA. \tag{4}$$

Струја I_3 представља разлику струја I_1 и I_2:

$$I_3 = I_1 - I_2 = 0.01818A - 0.0109A = 0.00728\,A = 7.28mA. \tag{5}$$

Напон који се шаље према ПЛК-у је:

$$U_{plc} = R_2 \cdot I_3 = 1500\,\Omega \cdot 0.00728\,A = 10.92\,V. \tag{6}$$

Напон на отпорнику R_1 рачуна се као:

$$U_{R1} = 24V - U_{plc} = 13.08\,V. \tag{7}$$

Снаге отпорника се рачунају као:

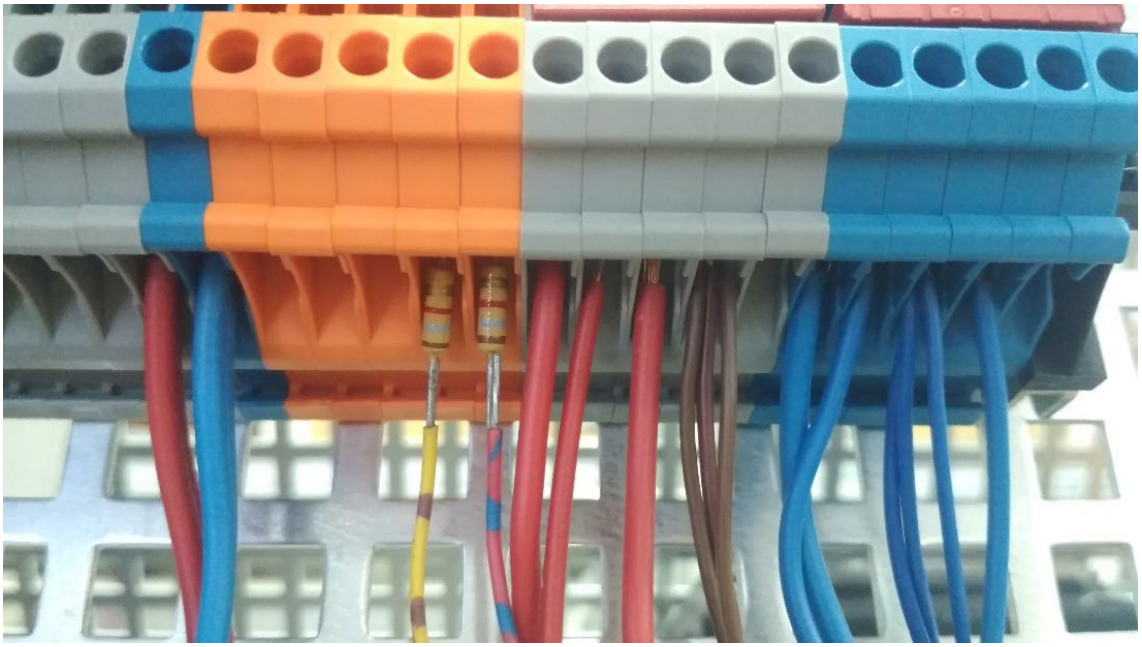
$$P_p = I_2^2 \cdot R_p = (0.0109\,A)^2 \cdot 2200\,\Omega = 0.261\,W. \tag{8}$$

$$P_1 = I_3^2 \cdot R_1 = (0.00728\,A)^2 \cdot 1800\,\Omega = 0.095\,W. \tag{9}$$

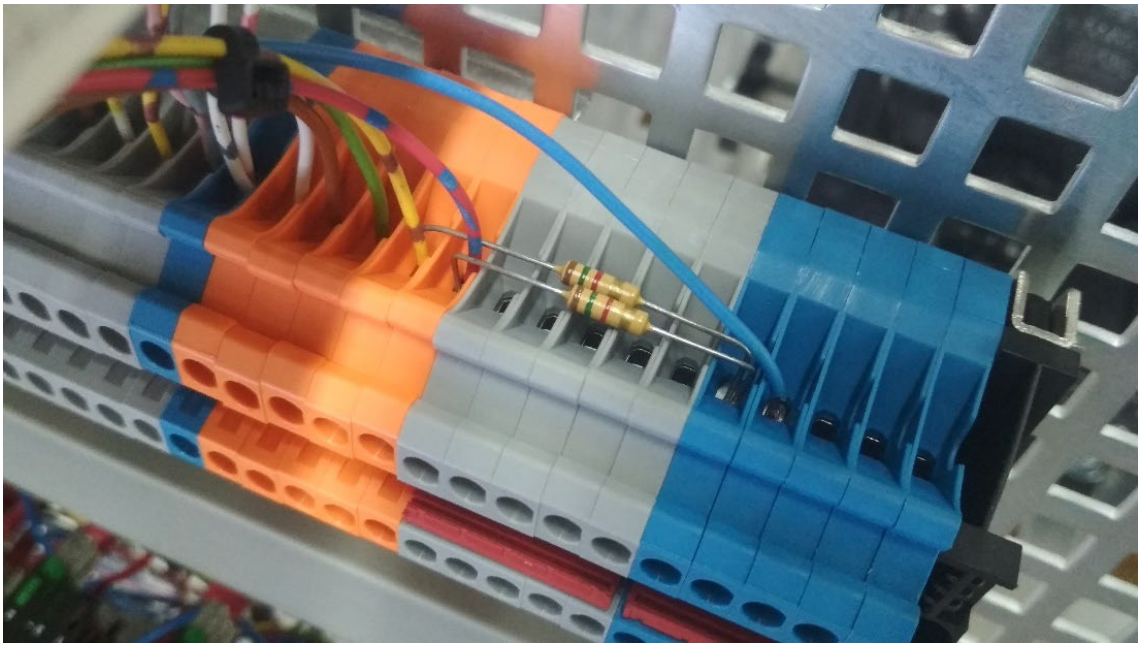
$$P_2 = I_3^2 \cdot R_2 = (0.00728\,A)^2 \cdot 1500\,\Omega = 0.079\,W. \tag{10}$$

Прорачун изнад односи се када је клизач потенциометра потпуно отворен, односно када је отпорност потенциометра 2200 **Ω**. У овом случају напон на напонском раздјелнику (отпорници R_1 и R_2) је 24 V.

На основу прорачуна, на радној станици уграђени су потребни отпорници који сачињавају напонски раздјелник, како је приказано на Сл. 21а и Сл. 21б.



Слика 21а – Уграђени отпорници на радној станици



Слика 21б – Уграђени отпорници на радној станици

2.4. Сензори

Сензори су врло битни за аутоматизоване системе јер на основу информација које сензори дају ПЛК добија информације о стању процеса којим управља. У зависности од врсте и типа сензора, излази сензора могу да буду дигитални (бинарни, прекидачки) или аналогни. Аналогни излази сензора могу бити напон или струја, у стандардним опсезима. Сензори уграђени на радној станици су:

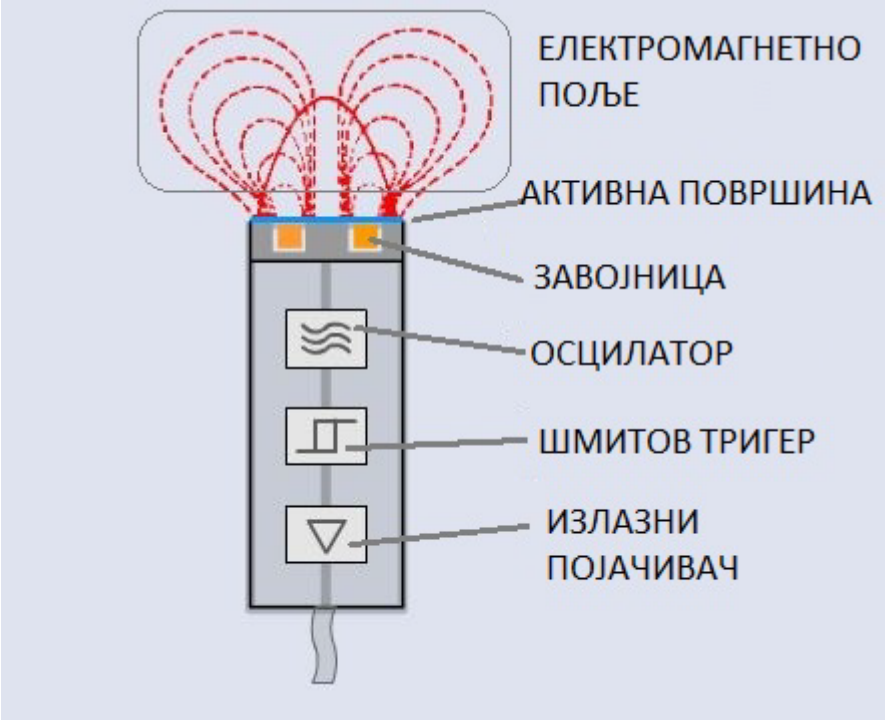
- индуктивни сензор,
- капацитивни сензор и
- оптички сензор.

Сва три наведена сензора представљају детекторе присуства јер су им излази дигитални.

2.4.1. Индуктивни сензор

Индуктивни сензор користи се за детекцију присуства металних објеката на малим растојањима. Принцип рада индуктивног сензора заснива се на промјени индуктивности

у зависности од промјене електромагнетне отпорности. На Сл. 22 приказана је блок шема индуктивног детектора присуства.



Слика 22 – Блок шема индуктивног сензора

Са Сл. 22 уочава се осцилатор, који се састоји од индуктивне завојнице коју сачињава велики број бакарних проводника, од кондензатора који се користи за складиштење наелектрисања и од извора енергије који се користи за побуђивање осцилатора. Кондензатор и завојница су уграђени тако да обезбиједе самоодрживо осциловање, односно генерисање синусног сигнала константне фреквенције. Осцилатор се напаја електричном енергијом из интерног извора једносмјерног напона.

Осцилације стварају електромагнетно поље на активној површини сензора. Када се метални објекат нађе у електромагнетном пољу, он утиче на смањење амплитуде осцилација. Амплитуда осцилација смањује се до те мјере да се активира коло које се налази испред осцилатора, односно Шмитов тригер. Шмитов тригер има задатак да на излазу да сигнал, све док се објекат налази у електромагнетном пољу сензора.

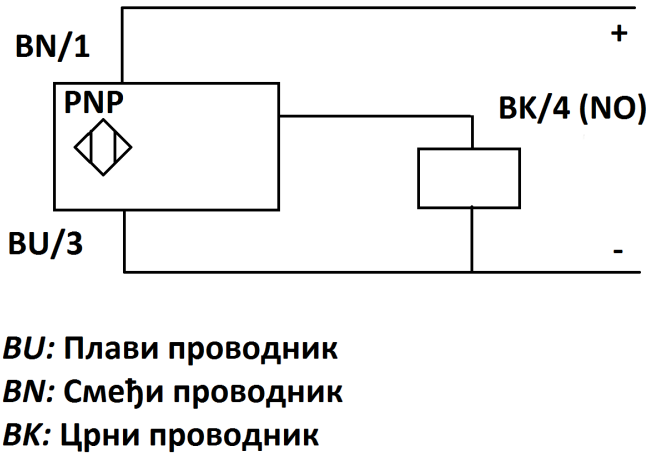
Излазни појачивач, односно излазно коло, користи се за појачање излазног сигнала са Шмитовог тригера, те је дигитални сигнал са сензора спреман за даљу употребу.

Индуктивни детектор присуства, уграђен на радну станицу, је *XS612B1PAL2* произвођача Шнајдер Електрик (Сл. 23).

Слика 23 – Индуктивни детектор присуства XS612B1PAL2

Индуктивни сензори могу бити различитих облика, а уграђени индуктивни сензор је цилиндричног облика са метричким навојем за монтажу. Кућиште је направљено од нерђајућег челика са свјетлосном индикацијом детекције објекта.

Из сензора су изведена три проводника. Два проводника се користе за напајање сензора, односно један је спојен на једносмјерни напон од 24 V док је други спојен на 0 V. Трећи проводник спаја се директно на модул дигиталних улаза ПЛК-а (Сл. 24).



Слика 24 – Шема повезивања индуктивног сензора на радној станици

Положај индуктивног сензора на радној станици је поред вратила асинхроног мотора. На излазно вратило асинхроног мотора постављен је ланчаник са осам зуба. Индуктивни сензор постављен је тако да може да региструје пролаз зуба. Обртањем вратила електромотора долази до ротације и самог ланчаника, те испред индуктивног сензора

долази до сталне смјене зуба и шупљина између њих што проузрокује сталну промјену на излазу из индуктивног сензора.

Када се испред сензора нађе зуб ланчаника, излаз из сензора ће бити 24 V док проласком шупљине испред индуктивног сензора, излаз са сензора ће бити 0 V. ПЛК може да искористи формирану серију импулса за прорачун брзине или положаја вратила асинхроног мотора. Ланчаник има осам зуба, што значи да за један пуни обртај вратила електромотора узрокује осам дигиталних импулса које изброји индуктивни сензор.

Индуктивни сензор *XS612B1PAL2* садржи једну жуту сигналну лампицу која се укључује када сензор шаље логичку јединицу на модул дигиталних улаза. Карактеристике индуктивног сензора *XS612B1PAL2* су:

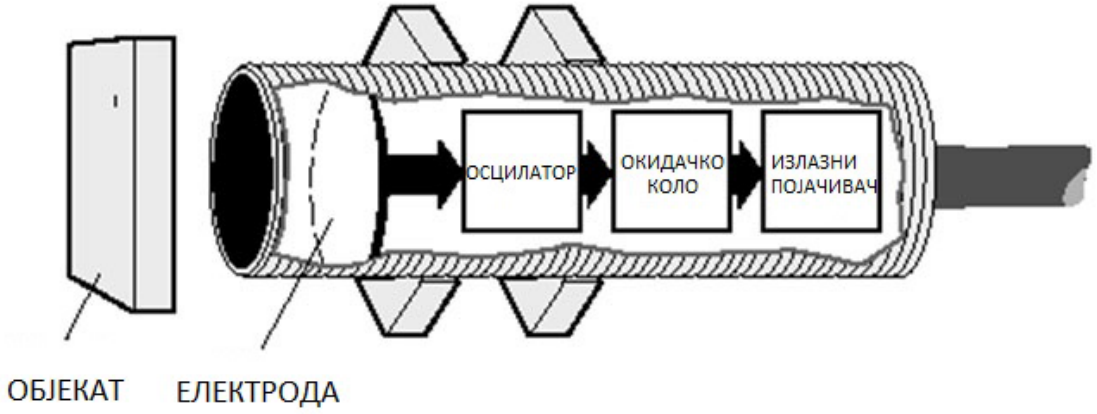
- напон напајања (једносмјерни) од 12 до 36 V уз заштиту од обрнутог поларитета,
- дигитални излазни сигнал,
- три проводника за повезивање,
- домет детекције објеката на растојању од 4 mm,
- максимална фреквенција промјена 2500 Hz,
- максимално вријеме одзива 0.2 ms,
- радна температура од -25 до 70°C.

2.4.2. Капацитивни сензор

Капацитивни сензор користи се за детекцију готово свих објеката, за разлику од индуктивног сензора који се користи само за детекцију металних објеката. Принцип рада капацитивног детектора присуства заснива се на промјени капацитивности која настаје као посљедица детекције објекта. Основни елемент капацитивног сензора је кондензатор. Између два проводна елемента, налази се диелектрик и на тај начин се формира кондензатор.

На Сл. 25 приказана је блок шема капацитивног детектора присуства.

Једну електроду кондензатора чини електрода која се налази на површини сензора, док другу електроду представља објекат који се детектује. Унутрашња електрода је спојена на осцилаторно коло. За разлику од индуктивног сензора који је генерисао електромагнетно поље, капацитивни сензор ради на принципу електричног поља. Када дође до детекције објекта, јавља се капацитивност између објекта и унутрашње електроде која се налази у сензору, а амплитуда осцилација расте све док не достигне задану вриједност на окидачком колу (Шмитов тригер). Окидачко коло шаље сигнал на излазни појачивач и на тај начин дигитални излазни сигнал са сензора је спреман за даље коришћење.



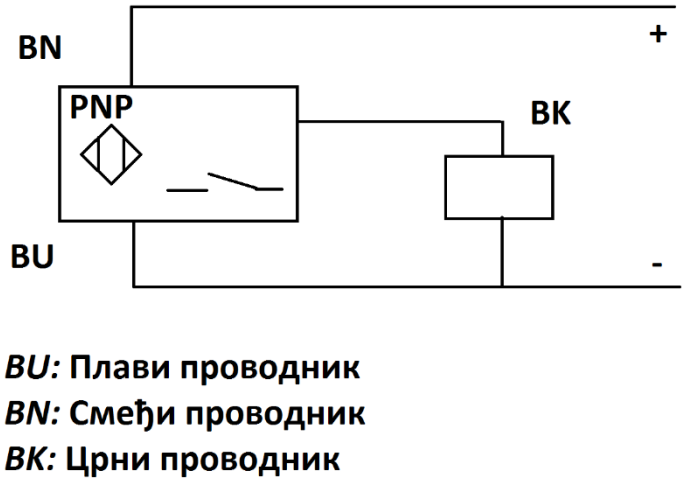
Слика 25 – Блок шема капацитивног детектора присуства

Капацитивни детектори присуства најчешће имају могућност подешавања нивоа осјетљивости, односно дистанце на којој се објекат може детектовати. У ту сврху, најчешће се користи потенциометар који се налази на сензору.

Капацитивни детектор присуства постављен на радној станици је *XT230A1PAL2,* произвођача Шнајдер Електик (Сл. 26). Положај сензора је на врху радне станице. Капацитивни детектор присуства посједује три извода. Два проводника се повезују на једносмјерни напон 24 V, односно 0 V, док се трећи излаз са сензора повезује директно на модул дигиталних улаза ПЛК-а (Сл. 27).

Слика 26 – Капацитивни детектор присуства XT230A1PAL2

Сензор је цилиндричног облика, а тијело сензора је направљено од пластике. За разлику од индуктивног сензора, капацитивни сензор има већи домет за неометану детекцију објекта.



Слика 27 – Шема повезивања капацитивног сензора на радној станици

Карактеристике капацитивног сензора *XT230A1PAL2* су:

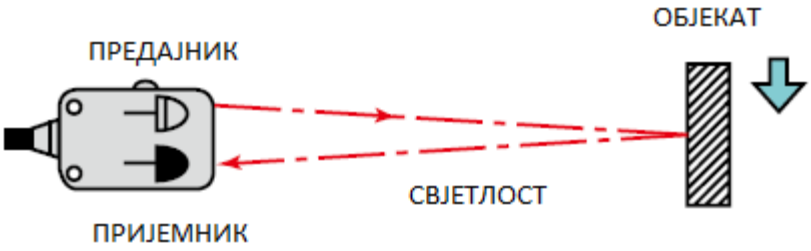
- напон напајања (једносмјерни) од 10 до 30 V уз заштиту од обрнутог поларитета,
- дигитални излазни сигнал,
- три проводника за повезивање,
- домет детекције објеката на растојању од 15 mm,
- максимална фреквенција промјена 150 Hz,
- максимално вријеме одзива 10 ms,
- радна температура од -10 до 60°C.

На сензору се налази жута сигнална лампица која се укључује када дође до детекције екстерног објекта.

2.4.3. Оптички сензор

Оптички детектори присуства користе се за детекцију објеката на одређеној даљини коришћењем свјетлосног снопа. Користе се за детекцију свих непрозирних објеката. Оптички сензор се састоји од пријемника и предајника свјетлости, као и потребних

електронских кола која обезбјеђују рад сензора. Предајник је најчешће ласерска диода или *LED* диода, док се као пријемник користи фото-диода или фото-транзистор. Електронска кола имају задатак да обезбиједе правилно напајање оптичког сензора и да обезбиједе појачање излазног сигнала. Свјетлост коју емитује предајник може бити видљива или инфрацрвена. На Сл. 28 приказан је принцип рада оптичког детектора присуства који ради на принципу дифузије свјетлости.



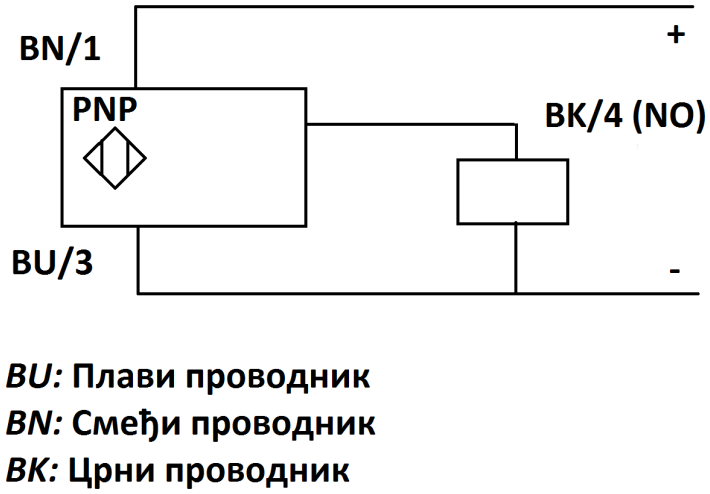
Слика 28 – Принцип рада оптичког детектора присуства

У сензору су смјештени и пријемник и предајник. Објекат који је детектован користи се као рефлектор, односно одбија дио снопа свјетлости и на тај начин се врши његова детекција путем оптичког детектора присуства.

Уграђени оптички сензор на радној станици је *XUB4APANL2*, произвођача Шнајдер Електрик (Сл. 29).

Слика 29 – Оптички детектор присуства XUB4APANL2

Положај оптичког детектора присуства је одмах поред напојне јединице ПЛК-а. Усмјерен је ка горе да се при раду са радним станицама не би нехотично ометао свјетлосни сноп и изазивала непредвиђена реакција ПЛК. Оптички сензор посједује три извода, од којих се два користе за напајање сензора (24 V и 0 V), док се трећи проводник користи за повезивање са дигиталним излазима (Сл. 30).



Слика 30 – Шема повезивања оптичког сензора на радној станици

Оптички сензор је цилиндричног облика израђен од пластике. Излаз са сензора је дигитална вриједност, односно са сензора се на модул дигиталних улаза шаље дигитални сигнал када дође до детекције објекта, те се при детекцији објекта укључује жута сигнална лампица која се налази на сензору.

Карактеристике оптичког сензора *XUB4APANL2* су:

- напон напајања (једносмјерни) од 10 до 36 V уз заштиту од обрнутог поларитета,
- дигитални излазни сигнал,
- три проводника за повезивање,
- домет детекције објеката на растојању до 100 mm,
- максимална фреквенција промјена 500 Hz,
- максимално вријеме одзива 1 ms,
- радна температура од -25 до 55°C.

2.5. Трофазни асинхрони мотор

У аутоматизованим системима као актуатори често се користе електромотори. У зависности од потребе система користе се мотори на једносмјерну, на наизмјеничну струју као и корачни електромотори.

Када се посматрају мотори на наизмјеничну струју (*AC* мотори), постоје два типа, асинхрони и синхрони. Назив асинхрони мотор потиче из разлога јер су код асинхроног мотора неусаглашене брзине ротора и брзина обртног магнетног поља. Асинхрони мотор може да буде једнофазни и трофазни. Трофазни намотај се повезује у звијезду или троугао у зависности од напона напајања.

Предност асинхроних мотора је ниска цијена, лако одржавање, поузданост, и лака регулација ако се користи у комбинацији са фреквенцијским регулатором, док релативно лош однос снаге и тежине представља ману.

На радну станицу монтиран је трофазни асинхрони мотор *Drehstrom – Asynchron Motor VDE 0530*, а његови параметри при фреквенцији напојне мреже од 50 Hz су:

- номинална снага 0.18 kW,
- номиналан број обртаја 2710 o/min,
- номинални напон напајања 230/400 V,
- номинална струја 0.95/0.55 A,
- могућност спајања у звијезду и троугао.

2.6. Чвориште за *CANopen* комуникацију

Један од начина повезивања програмабилног логичког контролера са фреквенцијским регулатором јесте путем *CANopen* протокола. Да би се веза између ова два уређаја остварила користи се чвориште. *CANopen* чвориште уграђено на радне станице је модел *TSXCANTDM4,* произвођача Шнајдер Електрик (Сл. 31).

Чвориште на себи садржи четири *SUB-D9* конектора од којих су два искоришћена за повезивање ПЛК-а и фреквенцијског регулатора. На први конектор (са десне стране) повезан је програмабилни логички контролер, док је на други конектор повезан фреквенцијски регулатор. Поред четири конектора на *CANopen* чворишту налази се и прекидач који може да буде у два стања, односно укључен и искључен и који се користи да омогући, односно онемогући *CANopen* комуникацију.

На дну модула налази извод који се повезује са уземљењем.

Слика 31 – Чвориште за CANopen комуникацију TSXCANTDM4

2.7. Заштитни уређај диференцијалне струје

У унутрашњости алуминијумског кућишта радне станице налази се ЗУДС[5] (*RCCB* склопка[6]). ЗУДС представља струјни сензор који мјери диференцијалну струју и врши аутоматски прекид струјног кола уколико дође до квара односно, ако измјерена диференцијална струја премаши дозвољену вриједност. ЗУДС ради на принципу Кирхофовог закона, односно на принципу да улазна струја потрошача мора бити једнака излазној струји из потрошача. Уређај мјери разлику струја кроз фазни и нулти проводник. У случају када потрошач ради исправно, струја кроз фазни проводник једнака је струји кроз нулти (неутрални) проводник па је диференцијална струја једнака нули.

Када дође до квара на потрошачу (спој дијелова под напоном са металном конструкцијом потрошача), струја кроз фазни проводник биће већа од струје кроз нулти проводник јер ће један дио струје тећи према земљи кроз заштитни проводник уземљења. ЗУДС детектује разлику у струјама и прекида струјни круг.

Људски организам може бити изложен јачини струје од 30 mA без посљедица. ЗУДС који има задатак заштиту човјека од струјног удара подешен је да искључује струјни круг

[5] ЗУДС склопка – Заштитни уређај диференцијалне струје

[6] *RCCB* склопка – енгл. *Residual Current Circuit Breaker*

при детекцији јачине диференцијалне струје управо од 30 mA. Поред заштите људи, ЗУДС може да се користи за заштиту од пожара или заштиту електромотора. Тада се користи ЗУДС који реагује на јачину диференцијалне струје од 300 mA.

Заштитни уређај диференцијалне струје може бити намијењен за монофазне и трофазне инсталације.

Заштитни уређај диференцијалне струје који је уграђен на радну станицу за заштиту од струјног удара је *A9Z21225*, произвођача Шнајдер Електрик (Сл. 32).

Слика 32 – Заштитни уређај диференцијалне струје A9Z21125

Карактеристике заштитног уређаја диференцијалне струје *A9Z21225* су:

- два пола, тј. монофазни,
- номинална струја 25 A (струја при којој ЗУДС може правилно функционисати за номинални напон при номиналној фреквенцији),
- струја кратког споја 10 kA (струја коју ЗУДС може да поднесе без оштећења у временском интервалу од 1 s),
- осјетљивост на промјену диференцијалне струје од 30 mA,
- тренутно вријеме реаговања,
- радни напон (наизмјенични) од 220 до 240 V,
- фреквенција мреже 50 Hz,
- радна температура од -20 до 65°C.

3. ОПИС СОФТВЕРСКЕ ПЛАТФОРМЕ

Након детаљног описа хардверске конфигурације радне станице, потребно је извршити упознавање и дати детаљну анализу софтвера. Како би радна станица могла да функционише према потребама и захтјевима корисника, важно је упознати се са програмским окружењем које се користи за конфигурацију, програмирање и мониторинг ПЛК-а. Програмска платформа која се користи за ПЛК-а серије *Modicon* M340 произвођача Шнајдер Електрик је *Unity Pro XL*[7]. Прије писања програма у софтверу *Unity Pro XL* потребно се упознати са базним појмовима везаним за програмирање.

3.1. Адресе и промјенљиве

У претходној глави описан је начин повезивања уређаја са улазима и излазима на програмабилном логичком контролеру. Уколико се ради о дигиталном улазу или излазу, њихове вриједности могу бити логичка нула или логичка јединица, те у меморији они заузимају само један бит.

Сваки улаз, односно излаз, повезан је са одређеним конектором ПЛК-а којем припада одређена адреса у меморији. Свака адреса почиње знаком „%“, те овај знак означава почетак адресе. Након знака „%“ долази словна ознака која означава да ли се ради о улазу или излазу. Ознаке за улазе, односно излазе су сљедеће:

- *I* – уколико се у адреси након знака „%“ налази знак „*I*“, то означава да се ради о дигиталном улазу,
- *Q* – уколико се у адреси након знака „%“ налази знак „*Q*“, ради се о дигиталном излазу,

Послије словне ознаке пишу се три броја која дефинишу сљедеће:

- Први број – адреса сабирнице, представља редни број сабирнице. На радној станици постоји само једна сабирница, те је овај број увијек „0“ у случају конкретне радне станице,
- Други број – адреса модула, означава на којем мјесту на сабирници се налази модул у чијем саставу је адресирани улаз, односно излаз. У опису хардверске структуре радне станице дат је распоред модула на сабирници, па сходно томе и адресирање модула је једнако редном броју положаја,
- Трећи број – адреса улаза, односно излаза, означава редни број улаза, односно излаза (конектора) на модулу.

[7] Послије верзије 13 зове се *EcoStructure Control Expert*.

За лакше разумијевање адресирања дат је сљедећи примјер:

$$\%I0.1.2. \tag{11}$$

Из (11) се види да се ради о адреси дигиталног улаза ПЛК-а који се налази на нултој сабирници, на модулу који је у првом лежишту (процесорски модул је у нултом), на другом конектору.

Током програмирања често постоји потреба да се позивају улази, или излази навођењем њихове адресе. То може бити заморно, а и захтјева од програмера да памти који сензор или актуатор је на којој адреси. Много је лакше ако се улазима, односно излазима додјеле имена која асоцирају на прикључени уређај.

Да би се то остварило уводи се појам промјенљиве. Промјенљиве представљају вриједности улаза и излаза које се мијењају у времену и свака промјенљива има своју адресу и име које јој додјељује програмер.

У Табелама 2 и 3 дате су адресе и називи промјенљивих написаних у сладу са стварним дигиталним улазима и излазима на ПЛК-у радне станице.

Табела 2 – Адресе дигиталних улаза

Прикључак	Адреса	Назив промјенљиве	Тип промјенљиве
0	%I0.1.0	Прекидач_1	*EBOOL*
1	%I0.1.1	Прекидач_2	*EBOOL*
2	%I0.1.2	Прекидач_3	*EBOOL*
3	%I0.1.3	Прекидач_4	*EBOOL*
4	%I0.1.4	Тастер_1	*EBOOL*
5	%I0.1.5	Тастер_2	*EBOOL*
6	%I0.1.6	Тастер_3	*EBOOL*
7	%I0.1.7	Тастер_4	*EBOOL*
8	%I0.1.8	Оптички_сензор	*EBOOL*
9	%I0.1.9	Индуктивни_сензор	*EBOOL*
10	%I0.1.10	Капацитивни_сензор	*EBOOL*
11	%I0.1.11	Нужни_стоп	*EBOOL*

Из Табеле 2. се види да наведена адреса из (11) представља трећи прекидач на контролној табли. На модул дигиталних улаза спојено је укупно дванаест уређаја, што се може видјети из табеле, а на располагању остају још четири слободна улаза.

Табела 3 – Адресе дигиталних излаза

Прикључак	Адреса	Назив промјенљиве	Тип промјенљиве
0	%Q0.2.0	Мотор_лијево	*EBOOL*
1	%Q0.2.1	Мотор_десно	*EBOOL*
2	%Q0.2.2	Црвена_1	*EBOOL*
3	%Q0.2.3	Жута_1	*EBOOL*
4	%Q0.2.4	Зелена_1	*EBOOL*
5	%Q0.2.5	Бијела_1	*EBOOL*
6	%Q0.2.6	Бијела_2	*EBOOL*
7	%Q0.2.7	Зелена_2	*EBOOL*
8	%Q0.2.8	Жута_2	*EBOOL*
9	%Q0.2.9	Црвена_2	*EBOOL*

Као што се види из Табеле 3. искоришћено је укупно десет конектора модула дигиталних излаза те на располагању остаје још шест слободних излаза.

До сада су описани начини адресирања промјенљивих које су типа бит, као што су дигитални улази и излази. Аналогне величине не могу да се опишу једним битом него са више њих зависно од резолуције. Најчешће је потребно два или четири бајта за репрезентацију аналогних величина.

Да би процесор знао колико ће простора резервисати за поједину промјенљиву, у адреси промјенљиве треба навести и тип промјенљиве. Тип промјенљиве се наводи словом иза слова које одређује да ли се ради о улазу или излазу. О ознакама за остале типове детаљније касније. Аналогни улази и излази су промјенљиве типа ријеч (енгл. *Word*) па се њихово адресирање записује у облику:

- *IW* – аналогни улаз и
- *QW* – аналогни излаз.

У Табели 4 дате су адресе и називи промјенљивих написаних у сладу са стварним аналогним улазима и излазима на ПЛК-у радне станице.

Табела 4 – Адресе аналогних улаза и излаза

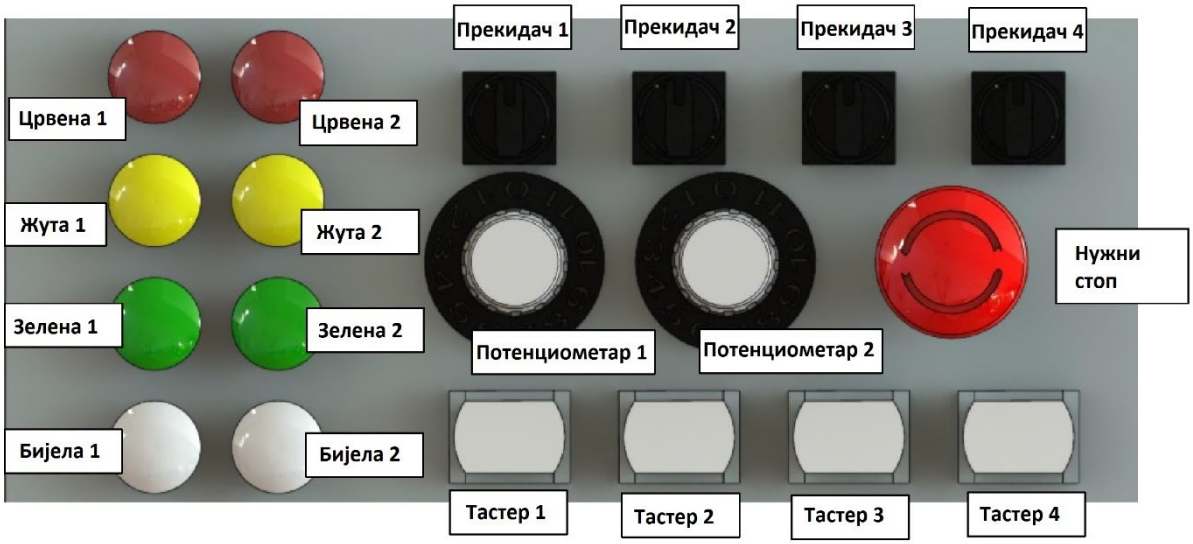
Прикључак	Адреса	Назив промјенљиве	Тип промјенљиве
0	%IW0.3.0	Аналогни_улаз	*INT*
1	%IW0.3.1	Потенциометар_1	*INT*
2	%IW0.3.2	Потенциометар_2	*INT*
4	%QW0.3.4	Аналогни_излаз	*INT*

Из Таб. 4. се уочава да постоје само два искоришћена аналогна улаза, на прикључцима 1 и 2, а прикључци 0 и 3 остају слободни.

На радној станици постоји један аналогни излаз који је повезан за аналогним улазом фреквенцијског регулатора. Адреса аналогног излаза је %QW0.3.4, те остаје још један слободан аналогни излаз који се може искористити, а његова адреса је %QW0.3.5.

Треба напоменути да промјенљиве аналогних улаза и излаза које се не користе у програму не треба формирати јер могу изазвати нефункционисање програма.

На Сл. 33. приказана је командна табла са називима промјенљивих као у Таб. 2, 3 и 4.



Слика 33 – Командна табла са ознакама

Често је потребно да се поред промјенљивих које припадају улазима и излазима, креирају и интерне промјенљиве. Интерне промјенљиве нису ни улази ни излази него мјеста у меморији на која смјештамо промјенљиве које егзистирају само у програму и служе за привремено или трајно памћење неких стања или вриједности. Адресирају се исто као улази и излази осим што умјесто словне ознаке „*I*“, односно „*Q*“ пише се „*M*“. Име им задаје програмер сходно њиховој улози у програму. Могу бити у два типа и то:

- %M<i> (користи се за креирање интерне промјенљиве типа *BOOL* (индекс *i* представља одређен број, примјер: %M1)) и
- %MW<i> (користи се за креирање интерне промјенљиве типа *WORD* (индекс *i* представља одређен број, примјер: %MW10)).

3.2. Типови промјенљивих

Из Таб. 2, 3 и 4. уочава се у четвртој колони тип промјенљиве. Модикон М340 ПЛК подржава сљедеће типове промјенљивих:

- *BOOL* (бит, користи се за памћење само једног бита који може бити или логичка нула или логичка јединица),
- *EBOOL* (користи се за памћење стања једног бита као и *BOOL,* са разликом да може садржати информацију да ли се ради о узлазној или силазној ивици при преласку из стања логичке нуле у стање логичке јединице, што може бити врло значајно при програмирању. Такође има могућност да се може вршити форсирање бита путем софтвера),
- *BYTE* (бајт, представља осам бита. Користи се за представљање цјелобројних података у опсегу од -128 до 127),
- *INT* (интеџер, промјенљива која у меморији заузима два бајта, односно шеснаест бита. Могу представити цјелобројне вриједности у опсегу од -32768 до 32767),
- *DINT* (дупли интеџер, користи се за меморисање цјелобројних података уколико је опсег који нуди тип податка *INT* недовољан. Тип промјенљиве *DINT* заузима простор у меморији од 32 бита, 4 бајта. Користи се представљање цјелобројних вриједности у опсегу од -2147483648 до 2147483647),
- *UINT* (позитивни интеџер, тип промјенљиве настао од енглеских ријечи *Unsigned Integer*. Тип промјенљиве *UINT* користи се за представљање цјелобројних података у вриједности од 0 до 65535, односно користи се само за позитивне цјелобројне вриједности. У меморији заузима простор од 16 бита, 2 бајта),
- *UDINT* (дупли позитивни интеџер, тип промјенљиве *UDINT* користи се за меморисање цјелобројних вриједности од 0 до 4294967295. У меморији заузима простор од 32 бита, 4 бајта),
- *REAL* (реални број, користи се за меморисање бројева са децималном тачком. У зависности од система, зависи прецизност *REAL* промјенљиве. У меморији заузима простор од 32 бита, 4 бајта),
- *WORD* (ријеч, тип промјенљиве који заузима простор у меморији од 16 бита, 2 бајта),
- *DWORD* (дупла ријеч, у меморији заузима вриједност од 32 бита, 4 бајта),
- *STRING* (текст, тип промјенљиве који се користи за чување текста у меморији. У принципу, тип промјенљиве *STRING* представља низ знакова. Низ знакова односи се на *ASCII* карактере, те је за сваки уписани карактер потребно 8 бита меморије. Из разлога што се промјенљиве *STRING* чувају у динамичкој меморији, веома је

битно имати представу колико знакова се уноси. Максималан број знакова који се могу унијети у *STRING* је 65535),

- *TIME* (вријеме, тип промјењљиве који се користи за меморисање времена. Вријеме је скаларна величина па уз своју бројчану вриједност има и јединицу, те је то разлог зашто се овај тип промјењљиве користи за меморисање времена, а не типови промјењљиве као што су *INT* и *REAL.* Синтакса уписивања овог типа промјењљиве у софтвер који се користи за програмирање ПЛК-а је „t#3s“, што конкретно у овом случају представља вриједност трајања временског интервала од три секунде),
- *DATE* (датум, представља тип промјењљиве који се користи за чување датума. У меморији заузима простор од 32 бита, 4 бајта. Шеснаест нижих бита користи се за меморисање дана и мјесеца у години, док се виших шеснаест бита користи за меморисање године),
- *DT* (датум и вријеме, представља тип промјењљиве за меморисање датума и времена. У меморији заузима простор од 64 бита, 8 бајтова),
- *TOD* (сати, тип промјењљиве настао од енглеских ријечи *Time Of the Day.* Користи се за меморисање времена. Осам најнижих битова је неискоришћено, односно ништа се не уписује у њих. Осталих 24 бита се користе за упис секунди, минута и дана).

3.3. Конверзија промјењљивих

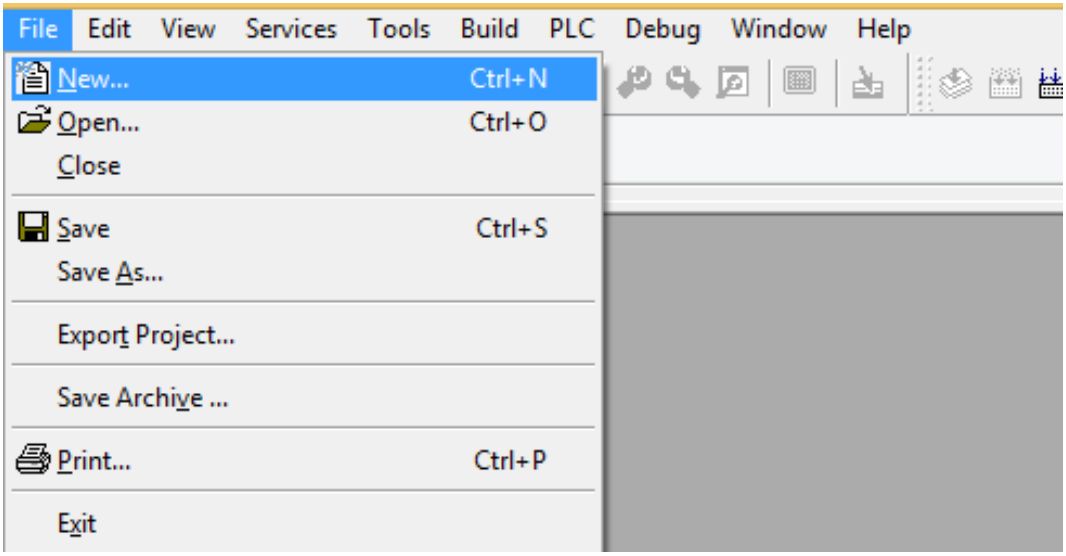
Програмски пакет *Unity Pro XL* пружа могућност претварања промјењљивих једног типа у други, односно жељени тип. Прије почетка претварања промјењљиве једног типа у промјењљиву другог типа, потребно је познавати колико који тип промјењљиве заузима простора у меморији. При одређеном претварању типова промјењљивих може да дође до губитка података усљед различите битске дужине промјењљиве у коју је уписан податак. Неки од примјера претварања једног типа података у други тип су:

- *REAL* у *INT* (претварање података из *REAL* у *INT* врши се на начин да се цјелобројни дио из *REAL* записује у *INT* док се децимални дио одбацује),
- *INT* у *REAL* (претварање је директно, гдје је цијели податак из *INT* цјелобројна вриједност, те се такав уписује у *REAL*),
- *INT, BYTE, DINT, WORD, DINT, UINT, UDINT, TIME, DWORD* или *REAL* у *BOOL* (претварање се врши на начин да уколико су подаци типа *INT, BYTE, DINT, WORD, DINT, UINT, UDINT, TIME, DWORD* или *REAL* већи од 0, они се у типу податка *BOOL* записују као логичка јединица, а ако су једнаки 0, онда се записују као логичка нула),

- *BYTE* у *DWORD, WORD, INT, DINT, UINT, UDINT, REAL* или *TIME* (претварање се врши на начин да се вриједност података типа *BYTE* препише у најниже бите типова података у које се претвара, док се сва у сва остала слободна мјеста, односно више битове, уписују логичке нуле),
- *TIME* у *INT* (приликом претварања из *TIME* у *INT* треба обратити пажњу да је јединица за вријеме у типу *TIME* секунда, док је јединица у типу *INT* милисекунда).

3.4. Програмска платформа *Unity Pro XL*

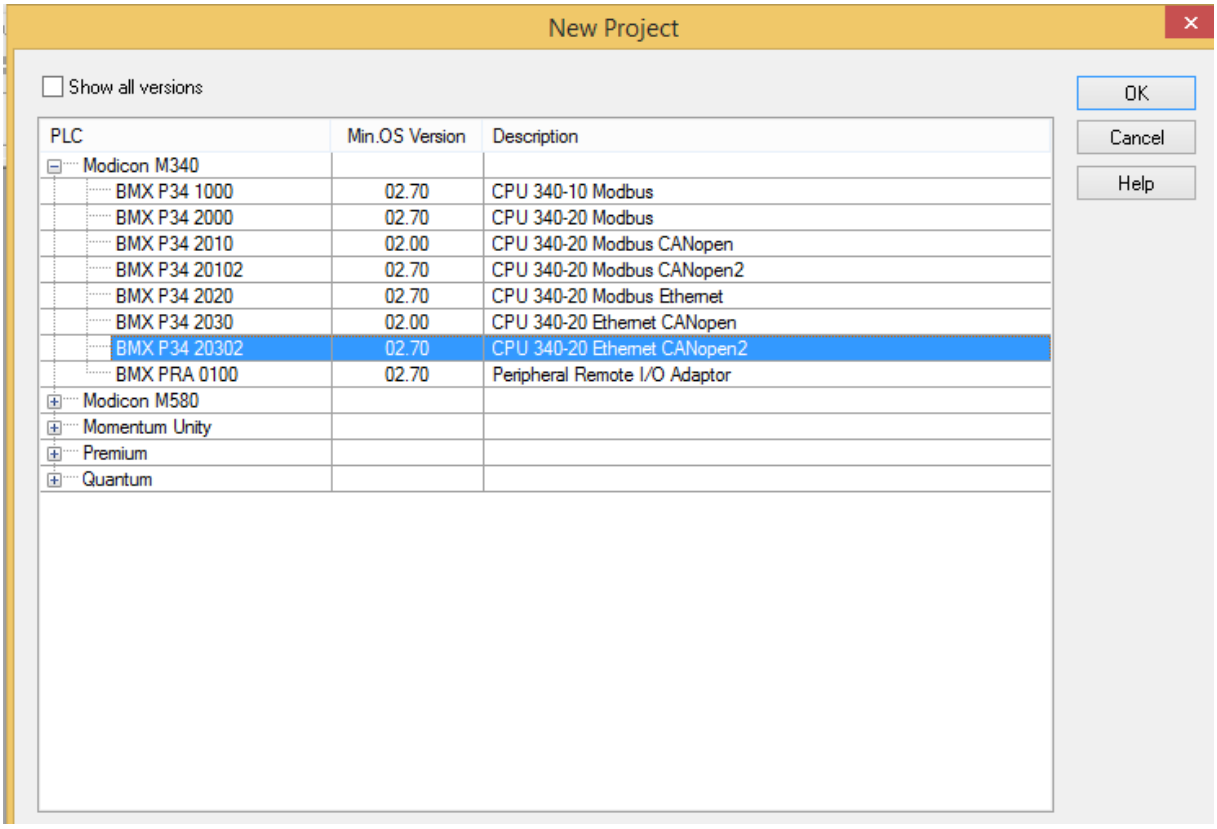
Након упознавања са адресама, промјенљивим, типовима промјенљивих, као и основним типовима конверзија између типова података, потребно се детаљно упознати са софтвером који се користи за програмирање програмабилног логичког контролера. Програмска платформа која се користи за програмирање ПЛК-а произвођача Шнајдер Електрик је *Unity Pro XL*[8]. Након правилне инсталације софтвера и покретањем истог, потребно је креирати нови пројекат (Сл. 34).



Слика 34 – Креирање новог пројекта у софтверу Unity Pro XL

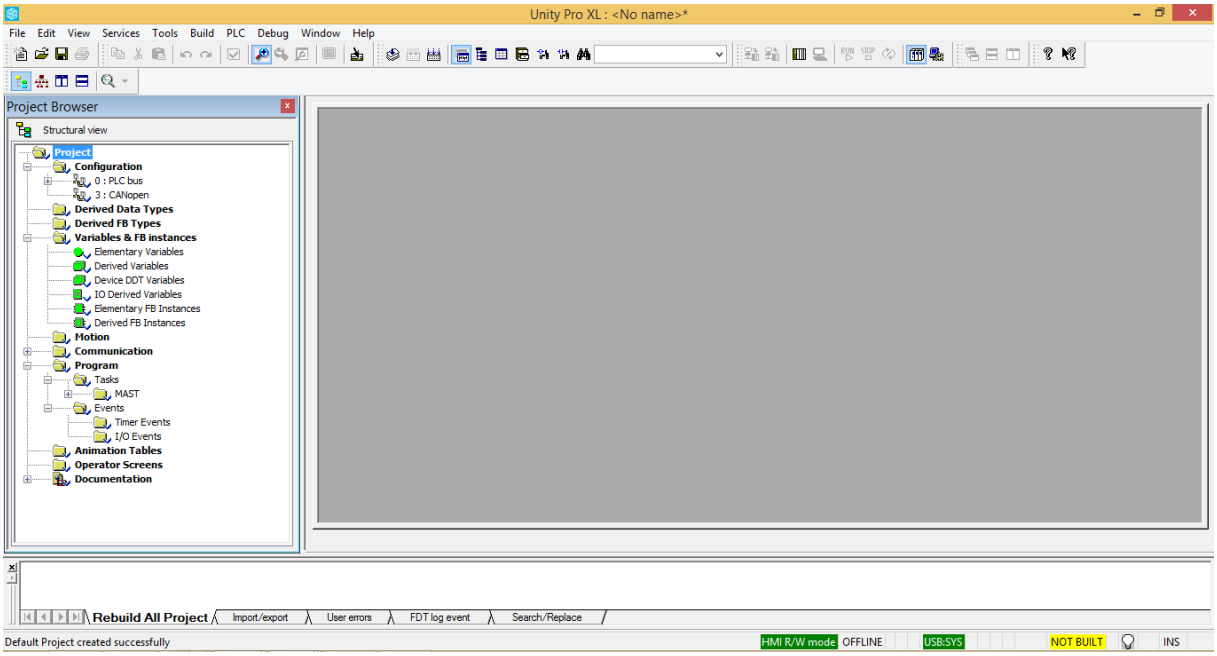
[8] Постоји више верзија софтверске платформе *Unity Pro*. Верзије су означене словима *S, M, L, XL, XLS* и у завиности од верзије кориснику се пружају различите могућности. Најмања софтверска платоформа је *Unity Pro S* и нуди најмање могућности. Могућности се огледају у приступу библиотекама које нуди софтверска платформа, као и приступу програмским језицима.

Након отварања опције *New*, софтвер тражи да се одабере одговарајућа верзија процесорског модула, што је у случају ове радне станице *BMXP3420302* (Сл. 35).



Слика 35 – Одабир централне процесорске јединице при креирању новог пројекта

Након одабира *BMXP3420302,* врши се потврда одабира *CPU* на *OK*. Описаним поступком направљен је нови пројекат и отвара се сљедећи прозор приказан на Сл. 36.

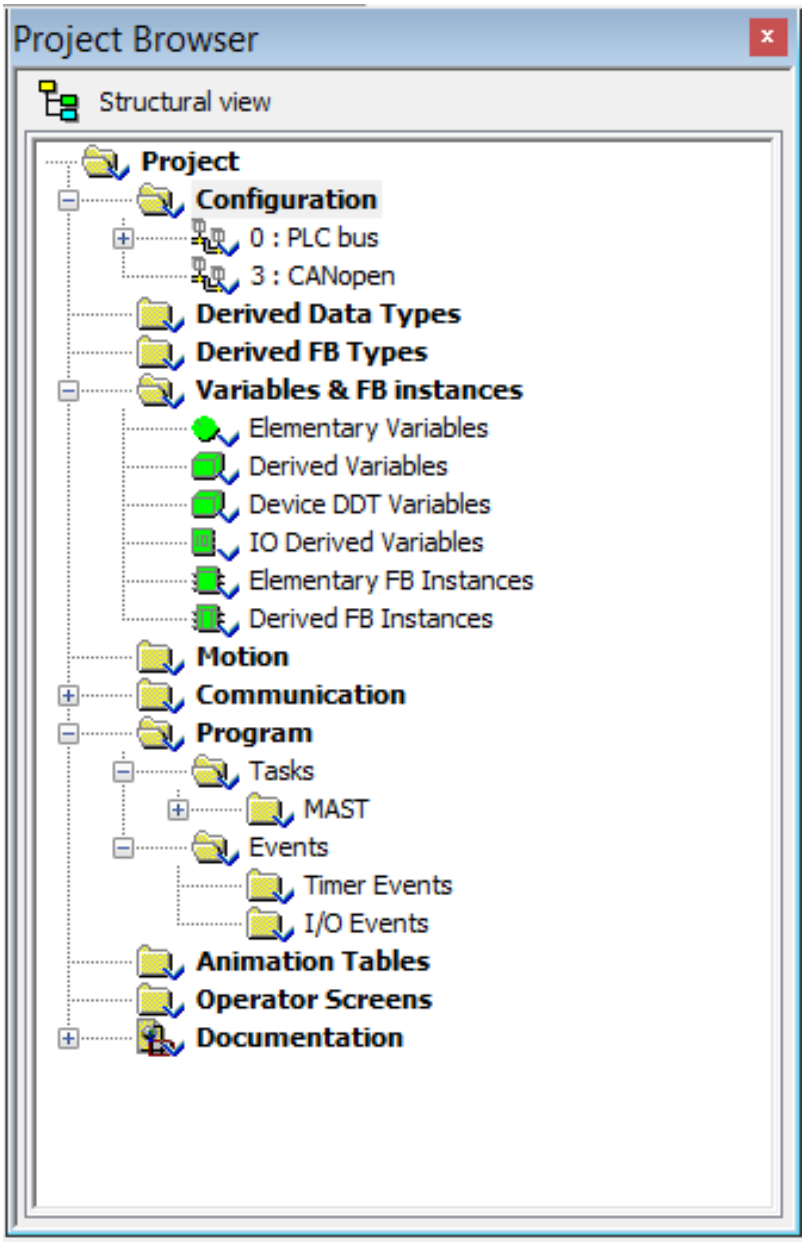


Слика 36 – Изглед прозора након креирања новог пројекта

Одабран је коришћени процесорски модул. Да би се могао писати програм, комплетан ПЛК мора бити конфигурисан у софтверу и то онако како је и у стварности конфигурисан. Софтверска конфигурација је неопходна јер се њоме остварује усаглашеност програма са хардвером. Конфигурација ПЛК-а се наставља додавањем осталих коришћених модула.

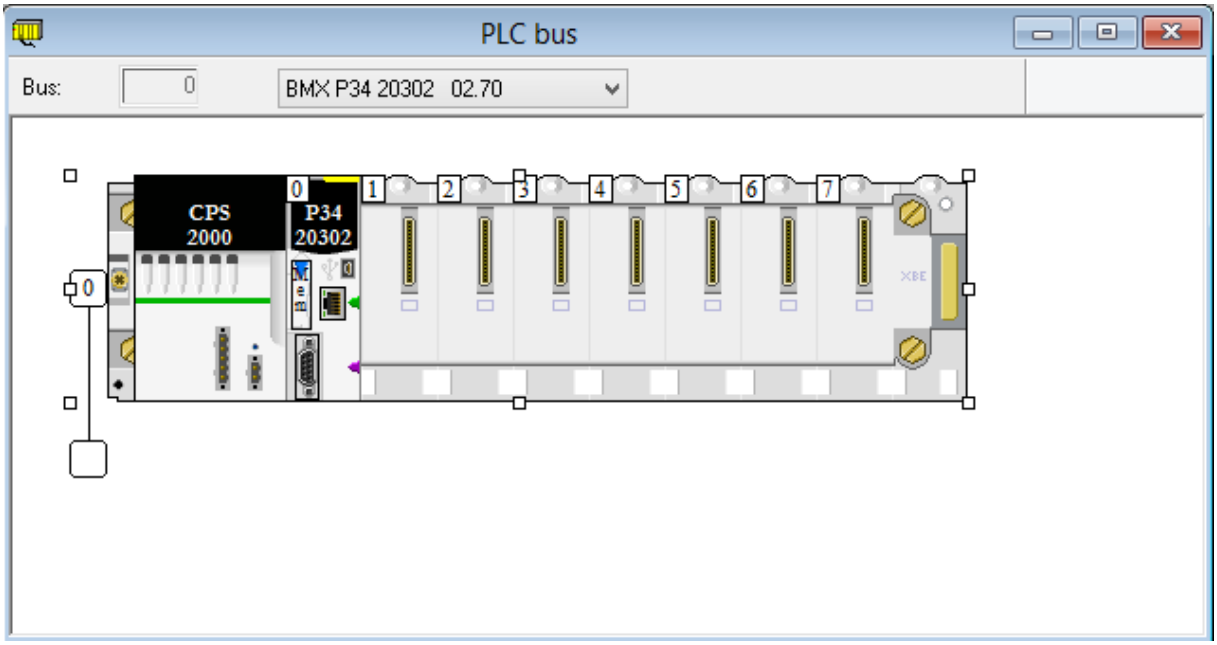
На сабирници се налазе модул дигиталних улаза (прво лежиште), модул дигиталних улаза (друго лежиште), модул аналогних улаза и излаза (треће лежиште), те модул брзог бројача (четврто лежиште). Софтверска конфигурација ПЛК-а реализује се у графичком окружењу које у потпуности одговара изгледу хардвера.

На лијевој страни Сл. 36 налази се *Project Browser,* у којем се налази поље *Configuration* (Сл. 37).



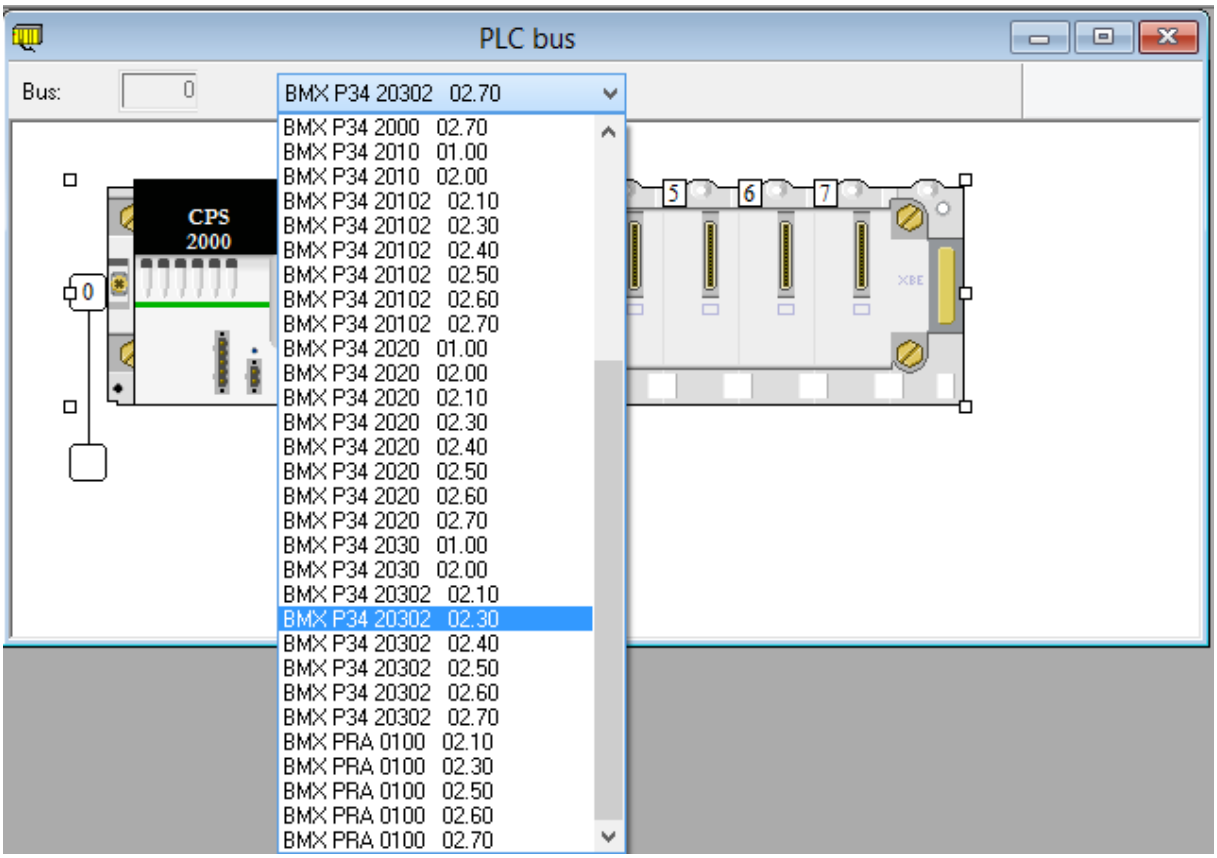
Слика 37 – Приказ опција у Project Browser-u

Отварањем поља *Configuration* отвара се нови прозор приказан на Сл. 38. На сабирници се тренутно налазе два модула, односно напојна јединица *CPS 2000* и процесорски модул *P34 20302*, који се налази на нултом лежишту. На лежишта један, два, три и четири потребно је додати одговарајуће модуле редослиједом којим су постављени хардверски на радној станици. Прије одабира модула, потребно је изабрати одговарајућу верзију фирмвера која се бира на падајућем менију приказаном на Сл. 39.



Слика 38 – Конфигурација ПЛК-а у софтверу Unity Pro XL

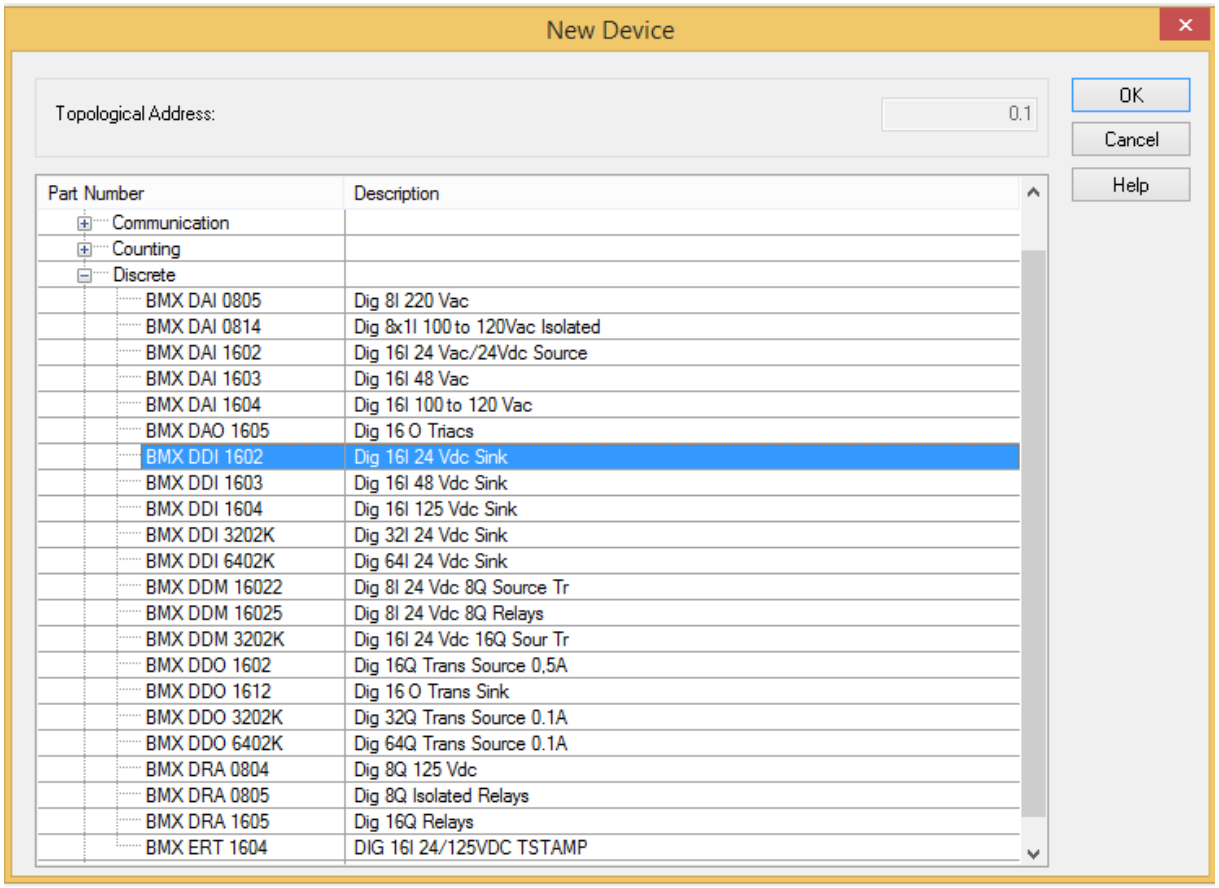
Тренутна верзија фирмвера који се користи је 02.70.



Слика 39 – Избор верзије фирмвера који се користи при програмирању ПЛК-а

На прво лежиште потребно је додати модул дигиталних улаза. Двоструким лијевим кликом миша на лежиште на позицији 1, отвара се нови прозор, те је потребно је навести који модул се жели додати. У прозору су модули груписани по функцији ради лакшег проналажења модула. Бира се група *Discrete* јер је модул дигиталних улаза. Из већ претходно описане хардверске структуре радне станице, познато је да је у првом лежишту модул типа *BMXDDI1602* (Сл. 40).

Потврда одабира модула потврђује се опцијом *OK*.



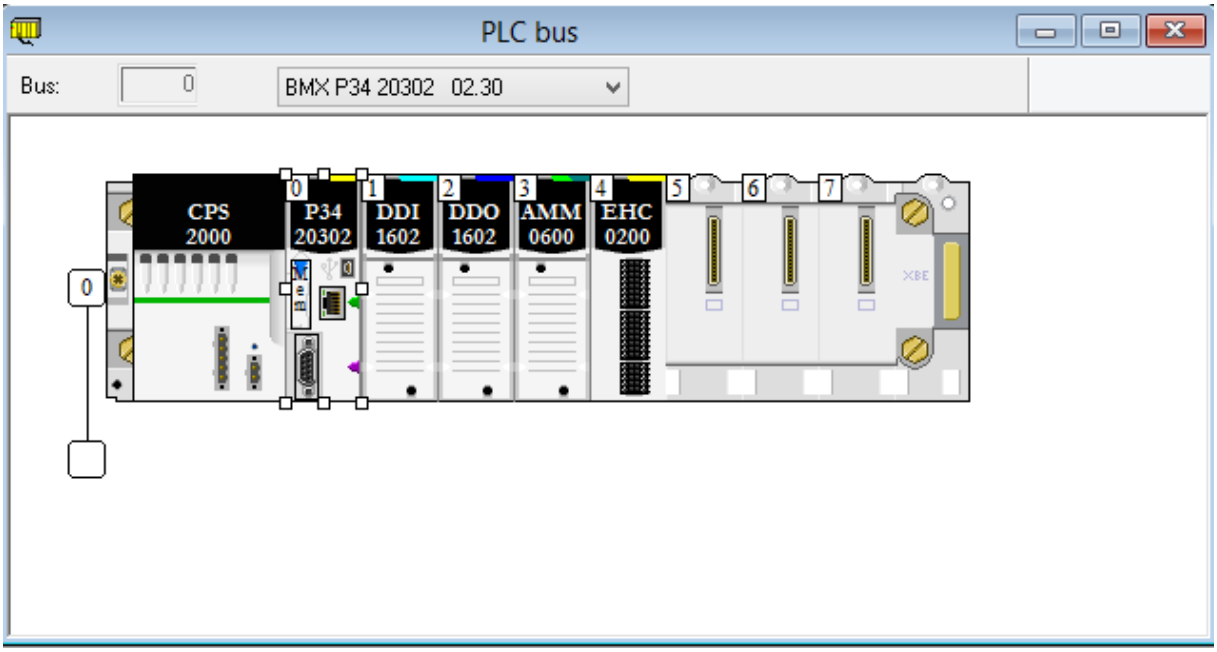
Слика 40 – Избор модула дигиталних улаза BMXDDI1602

Додавање модула дигиталних излаза врши се на сличан начин. Једина разлика је што се на лежиште два поставља дигитални излазни модул *BMXDDO1602.*

Постављање модула аналогних улаза и излаза на треће лежиште врши се на начин да се одабере група *Analog*, те се бира модел *BMXAMM0600.*

На четврто лежиште потребно је додати модул брзог бројача. Модел брзог бројача *BMXEHC0200* налази се у групи *Counting*. Уколико се појави проблем у вези са модулом брзог бројача, модул је потребно уклонити са сабирнице, јер се не користи.

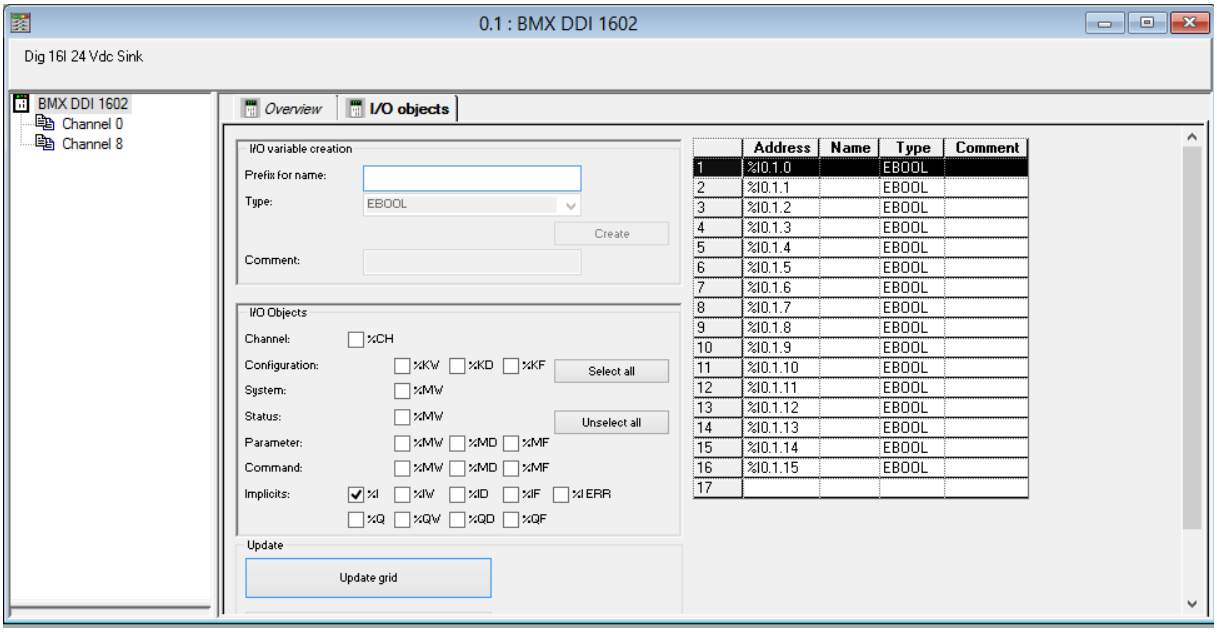
Када се заврши процес конфигурације, прозор за конфигурацију изгледа као на Сл. 41.



Слика 41 – Изглед сабирнице ПЛК-а у софтверу Unity Pro XL након завршене конфигурације

Сљедећи корак је додјељивање адреса улазима и излазима на свим модулима. Додјељивање адреса почиње двоструким лијевим кликом на одговарајући модул који се налази на сабирници у конфигуратору (Сл. 41).

Отвара се прозор приказан на Сл. 42 и потребно је одабрати картицу *BMXDDI1602* на лијевој страни прозора.



Слика 42 – Адресирање улазних уређаја на модулу дигиталних улаза BMXDDI1602

Након тога потребно је одабрати картицу *I/O objects*. Већ је описано да адресе дигиталних улаза почињу са знаковима „%*I*“, те како би добили приступ адресама улаза на модулу потребно је означити квадратић поред којег пише „%*I*“ (Сл. 42) и потом освјежити табелу притиском на *Upgrade grid.*

Освјежена табела приказана на десној страни Сл. 42, гдје су наведене адресе свих улаза који постоје на модулу и њихови типови. Сваком улазу може да се додијели име којим се у програму може позивати умјесто навођења комплетне адресе. Имена се уписују у колони *Name.*

Имена промјенљивих могу да буду произвољна, али најбоље је задати имена која ће олакшати програмирање. Примјер тога је да је на адреси *%I0.1.0* повезан прекидач 1, те је најбоље дати име наведеној промјенљивој *Prekidac1*. Након задавања имена свим промјенљивим, односно улазима, добија се табела приказана на Сл. 43.

Исти поступак треба поновити за модул дигиталних излаза. Једина разлика је у томе што се прије опције *Update grid*, означава квадратић који поред свог имена има „*%Q*“. Након додјељивања имена промјенљивима, табела излаза на модулу дигиталних излаза ће имати изглед као на Сл. 44.

	Address	Name	Type	Comment
1	%I0.1.0	Prekidac1	EBOOL	
2	%I0.1.1	Prekidac2	EBOOL	
3	%I0.1.2	Prekidac3	EBOOL	
4	%I0.1.3	Prekidac4	EBOOL	
5	%I0.1.4	Taster1	EBOOL	
6	%I0.1.5	Taster2	EBOOL	
7	%I0.1.6	Taster3	EBOOL	
8	%I0.1.7	Taster4	EBOOL	
9	%I0.1.8	Opt_senzor	EBOOL	
10	%I0.1.9	Ind_senzor	EBOOL	
11	%I0.1.10	Kap_senzor	EBOOL	
12	%I0.1.11	NuzniStop	EBOOL	
13	%I0.1.12		EBOOL	
14	%I0.1.13		EBOOL	
15	%I0.1.14		EBOOL	
16	%I0.1.15		EBOOL	
17				

Слика 43 – Табела улазних промјенљивих са адресама, именом и типом

	Address	Name	Type	Comment
1	%Q0.2.0	Motor_Lijevo	EBOOL	
2	%Q0.2.1	Motor_Desno	EBOOL	
3	%Q0.2.2	Crvena1	EBOOL	
4	%Q0.2.3	Zuta1	EBOOL	
5	%Q0.2.4	Zelena1	EBOOL	
6	%Q0.2.5	Bijela1	EBOOL	
7	%Q0.2.6	Bijela2	EBOOL	
8	%Q0.2.7	Zelena2	EBOOL	
9	%Q0.2.8	Zuta2	EBOOL	
10	%Q0.2.9	Crvena2	EBOOL	
11	%Q0.2.10		EBOOL	
12	%Q0.2.11		EBOOL	
13	%Q0.2.12		EBOOL	
14	%Q0.2.13		EBOOL	
15	%Q0.2.14		EBOOL	
16	%Q0.2.15		EBOOL	
17				

Слика 44 – Табела излазних промјенљивих са адресама, именом и типом

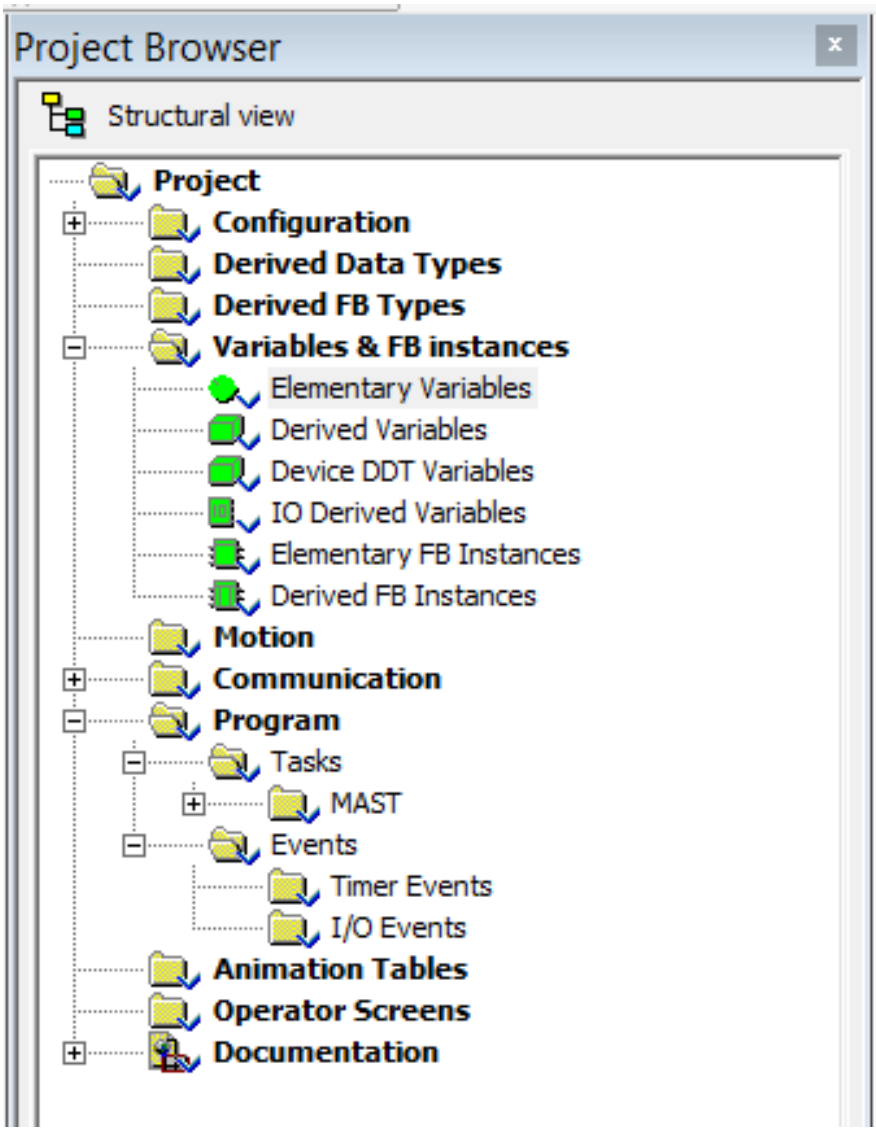
Поступак за аналогне улазе и излазе је сличан као и за дигиталне улазе и излазе само што се прије опције *Update grid*, означи квадратић поред којег пише „*%IW*“ и „*%QW*“. Након промјене назива промјенљивих, табела аналогних улаза и излаза изгледа као на Сл. 45.

	Address	Name	Type	Comment
2	%IW0.3.0	Analogni_ulaz	INT	
3	%IW0.3.0.2		INT	
4	%IW0.3.0.3		INT	
5	%QW0.3.0		INT	
6	%QW0.3.0.1		INT	
7	%IW0.3.1	Potenciometar1	INT	
8	%IW0.3.1.1		INT	
9	%IW0.3.1.2		INT	
10	%IW0.3.1.3		INT	
11	%QW0.3.1		INT	
12	%QW0.3.1.1		INT	
13	%IW0.3.2	Potenciometar2	INT	
14	%IW0.3.2.1		INT	
15	%IW0.3.2.2		INT	
16	%IW0.3.2.3		INT	
17	%QW0.3.2		INT	
18	%QW0.3.2.1		INT	
19	%IW0.3.3		INT	
20	%IW0.3.3.1		INT	
21	%IW0.3.3.2		INT	
22	%IW0.3.3.3		INT	
23	%QW0.3.3		INT	
24	%QW0.3.3.1		INT	
25	%IW0.3.4		INT	
26	%IW0.3.4.1		INT	
27	%QW0.3.4	Analogni_Izlaz	INT	
28	%QW0.3.4.1		INT	

Слика 45 – Табела излазних промјенљивих са адресама, именом и типом

За модул брзог бројача није потребно уносити називе промјенљивих у табелу јер ниједан уређај није спојен на модул брзог бројача.

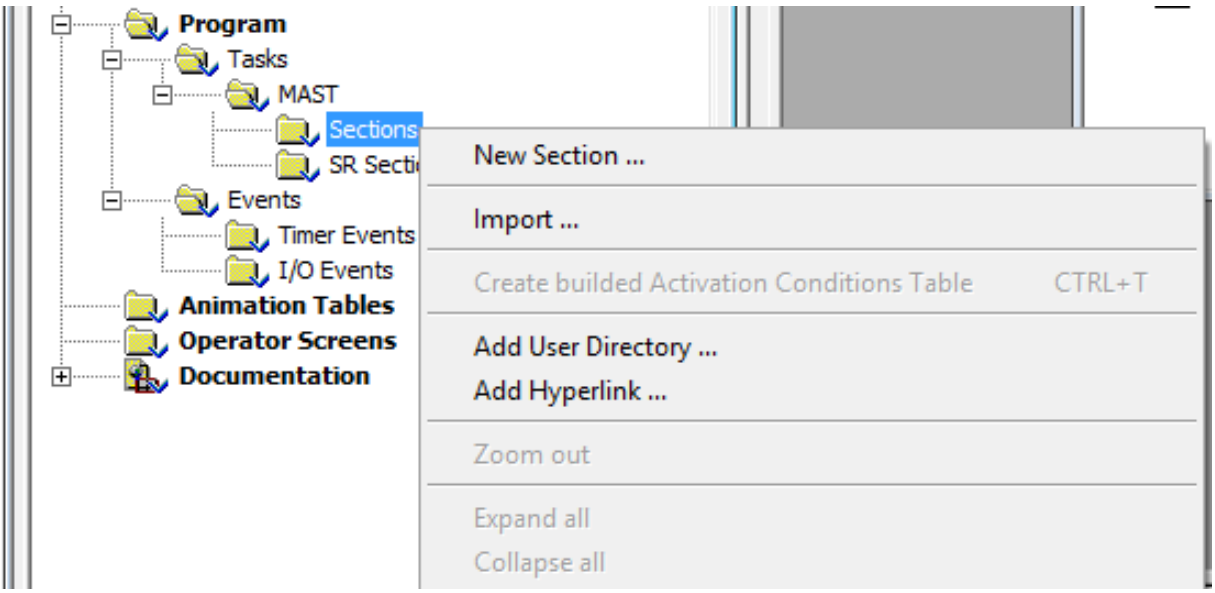
Списку свих промјенљивих се може приступити у *Project Browseru* на картици *Variables & FB instances*, двоструким лијевим кликом миша на *Elementary Variables* (Сл. 46).



Слика 46 – Приступ списку дефинисаних промјенљивих

Приликом покретања картице *Elementary Variables,* отвара се табела свих унесених промјенљивих, те је могуће додавати нове промјенљиве или мијењати постојеће (назив промјенљиве, тип промјенљиве, додавање коментара итд.).

Након софтверске конфигурације ПЛК-а и адресирања улаза и излаза, може се започети са писањем програма. Лијевим кликом миша на картицу *Program*, затим *Tasks*, те након тога *MAST*, добија се приступ картици *Sections*. Писање програма почиње креирањем новог програма, десним кликом миша на опцију *Sections*, а затим избором опцијом *New Section*, као што је приказано на Сл. 47.

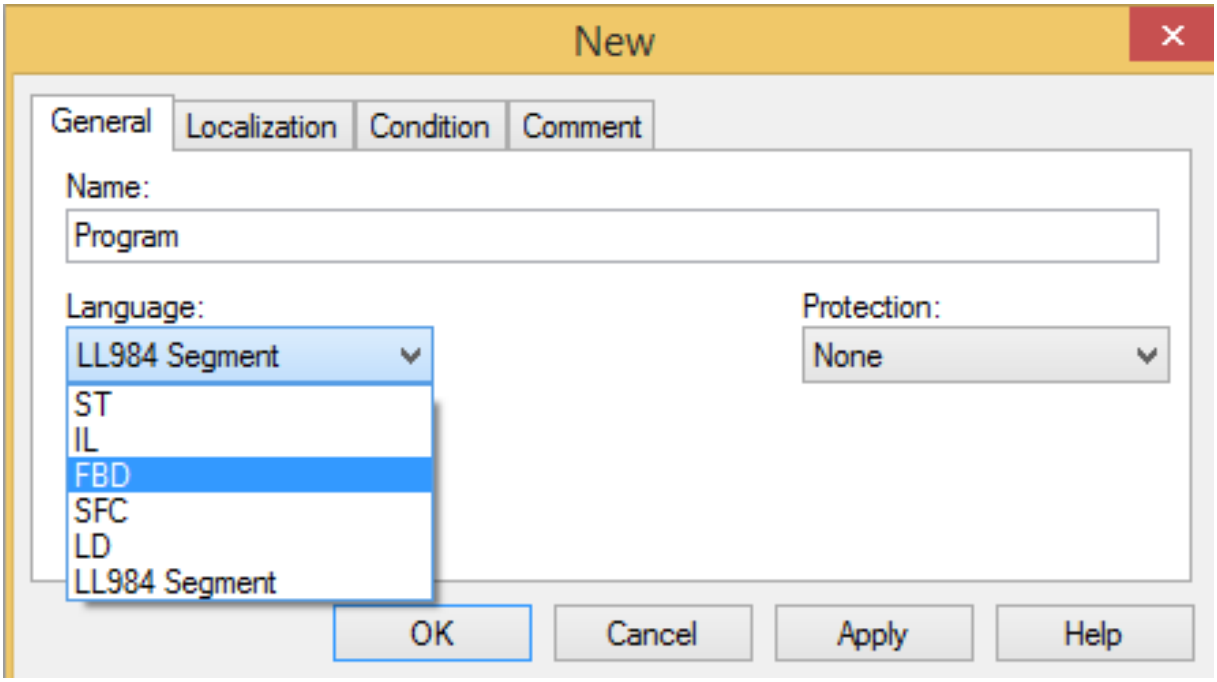


Слика 47 – Креирање новог програма у софтверу Unity Pro XL

Како би се генерисао нови програм потребно је задати жељено име програму и одабрати језик у којем ће програм бити писан (Сл. 48). У поље *Name* уписује се име програма, док се на менију *Language* бира програмски језик.

Софтверска платформа *Unity Pro XL* нуди програмирање у више програмских језика. Који програмски језици могу да се користе описано је у наредној глави.

У оквиру једног пројекта корисник може да направи више програма одвојених у различите цјелине (енгл. *Sections*), како би себи олакшао прегледност програма. Програмске цјелине могу бити писане у различитим програмским језицима.



Слика 48 – Креирање новог програма

3.5. Програмски језици ПЛК-а Модикон М340

Пет програмских језика који су дефинисани стандардом IEC 61131-3 и који могу да се користе за програмирање ПЛК-а Модикон М340 су:

- Структурирани текст (енгл. *Structured Text, ST*),
- Листа наредби (енгл. *Instruction List, IL*),
- Функцијски блок дијаграм (енгл. *Function Block Diagram, FBD*),
- Љествичасти дијаграм (енгл. *Ladder Diagram, LD*),
- Секвенцијални функцијски дијаграм (енгл. *Sequential Function Chart, SFC*),
- *LL984 Segment* (специфичан за Шнајдер Електрик, није дефинисан стандардом IEC 61131-3).

3.5.1. Структурирани текст (*ST*)

Структурирани текст (*ST*) представља један од пет програмских језика за програмирање ПЛК-а по стандарду *IEC-61131-3*. Писање програма је слично програмирању у програмском језику *Pascal*. Инструкције се пишу једна иза друге, те се

на тај начин и извршавају. Структурирани текст као програмски језик у софтверу *Unity Pro XL* омогућава коришћење итеративних петљи, условних петљи, као и различитих функција укључујући математичке.

Примјер једне линије програма написаног у *ST* програмском језику је:

$$Vrijednost = SQRT_{REAL(Vrijednost)}.$$

Наведена наредба израчунава квадратни коријен промјенљиве *Vrijednost* чији је тип *REAL* и уписује је на исту адресу.

3.5.2. Листа инструкција (*IL*)

Листа инструкција је програмски језик који је сличан асемблерском језику. Свака инструкција се пише у новој линији кода. Инструкција се састоји од једног оператора и након њега низа операнада. По стандарду *IEC-61131-3* постоји неколико врста оператора, а неки од њих су:

- *LD* (користи се за уписивање операнда у регистар који се користи за уписивање резултата),
- *LDN* (користи се за уписивање негиране вриједности операнда у регистар који се користи за уписивање резултата),
- *AND* (користи се као оператор логичке *I* функције),
- *OR* (оператор логичке *ILI* функције), итд.

Примјер инструкције у програмском језику *IL* дат је:

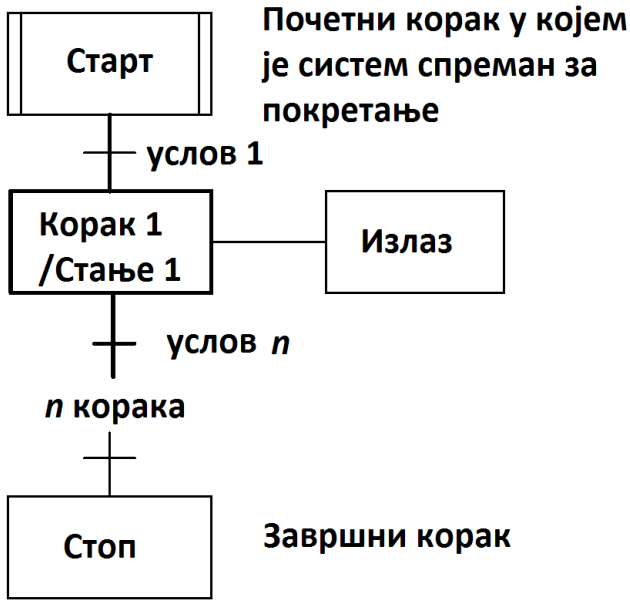
$$upisi\ operand{:}\ LD\ C\ (*\ upisi\ operand\ C\ *).$$

Први дио инструкције прије оператора *LD* представља назив инструкције (енгл. *Label*). Оператор *LD* је оператор за уписивање операнда у регистар, док је *C* операнд који се уписује. На крају инструкције налази се коментар који се може користити као додатно појашњење и који се не компајлира.

3.5.3. Секвенцијални функцијски дијаграм (*SFC*)

Секвенцијални функцијски дијаграм (*SFC*) је програмски језик намијењен програмирању секвенцијалних управљања. Графички је језик и подсјећа на дијаграме тока у програмирању, а заправо је прилагођен приказивању и управљању корацима у процесу којим се управља. Процес управљања системом почиње одређеним условом који представља почетак извршавања циклуса, након тога слиједи произвољан број услова који ће омогућити управљање системом у другом кораку. За прелазак у сваки наредни корак задаје се одређени услов који треба бити испуњен. На крају секвенце налази се посљедњи корак који коришћењем одређеног услова може да врати управљање у неки

од претходних корака. Кораци су представљени правоугаоницима, док се симбол почетног корака разликује од осталих блокова. Начин коришћења *SFC* приказан је на Сл. 49.



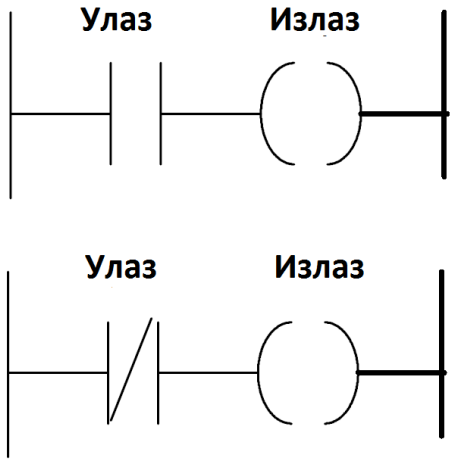
Слика 49 – Структура секвенцијалног функцијског дијаграма (SFC)

3.5.4. Љествичасти дијаграм (*Ladder*)

Љествичасти дијаграм као начин програмирања настао је да би олакшао инжењерима који посједују знање из пројектовања релејних шема, да што лакше напишу програмски код. Коришћењем програмских симбола (разних врста тајмера, бројача, итд.) креира се програм.

Љествичаста структура подијељена је на произвољан број линија (рангова). Да би један ранг био валидан, са лијеве стране постављају се улазне величине, односно услови који требају бити испуњени да би се извршила одређена акција. Са десне стране се постављају излази тј. акције и сваки ранг мора имати бар један или више акција.

Код у програмском језику *LD* се чита одозго према доље, те свака линија се чита са лијеве стране према десној. Када је ПЛК у оперативном стању, он чита комплетан љествичасти дијаграм и мијења управљање уколико је дошло до испуњавања одређених услова. Могуће је поставити исти улазни/излазни уређај у различитим ранговима љествичасте структуре. На Сл. 50 приказан је примјер једноставног програма.



Слика 50 – Примјер програма у програмском језику LD

У првом случају (Сл. 50) излаз ће имати стање логичке нуле све док је стање улаза логичка нула. За улаз се користи нормално отворени контакт чија вриједност постаје логичка јединица тек када је испуњен одређен услов. Конкретно, на радној станици то може бити један од прекидача који када се активира, услов на улазу постаје испуњен, односно долази до промјене сигнала са логичке нуле на логичку јединицу. Све док је улазни услов испуњен, односно док је на дигиталном улазу логичка јединица, излаз ће бити активан.

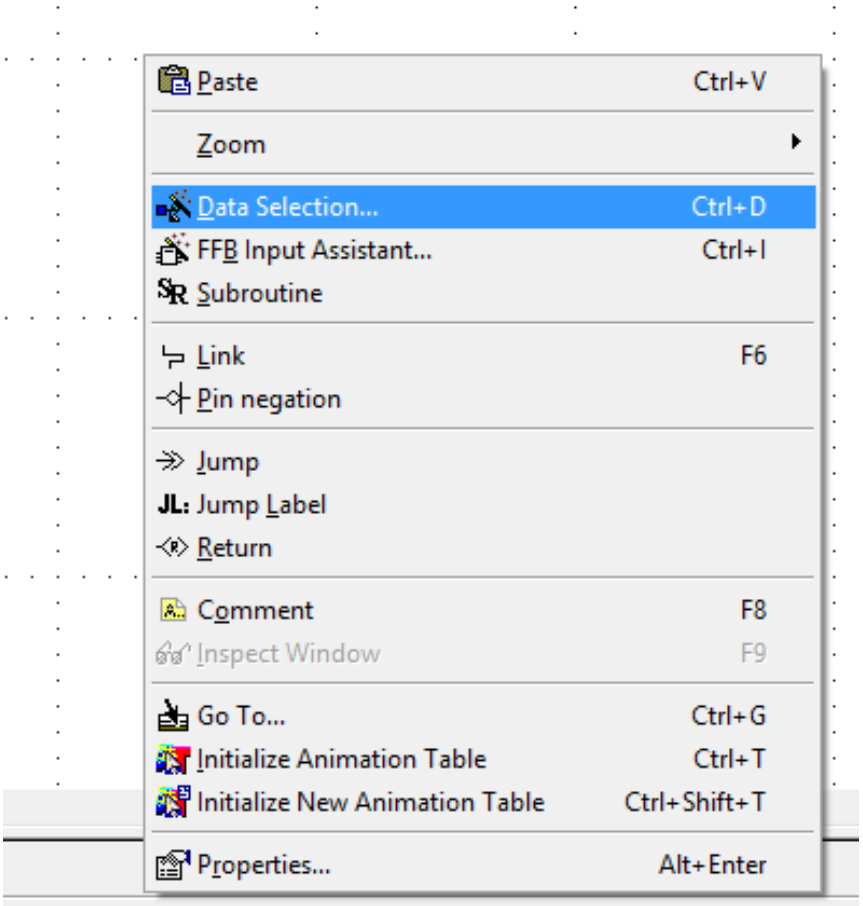
У другом случају се на улазу користи нормално затворени контакт, а примјер нормално затвореног контакта на радној станици је нужни стоп. Излаз ће бити активан све док не дође до промјене улазног сигнала са логичке јединице на логичку нулу. Поред приказаних симбола на Сл. 50, за писање програма могу се користити тајмери, бројачи, компаратори итд.

3.5.5. Функцијски блок дијаграм (*FBD*)

Функцијски блок дијаграм представља графички начин програмирања. Улазне и излазне промјенљиве се повезују на улазе, односно излазе блокова. Могуће је повезивати блокове, односно излазе једног блока повезати за улазима другог блока. Конекције, односно линије којима се блокови повезују су оријентисане, односно сигнали се преносе са лијева на десно. Битно је нагласити да уколико се повезују блокови, конекција између њих преноси сигнал одређеног типа, те излаз једног блока мора имати исти тип промјенљиве као улаз другог блока.

3.6. Опис блокова у програмском језику *FBD*

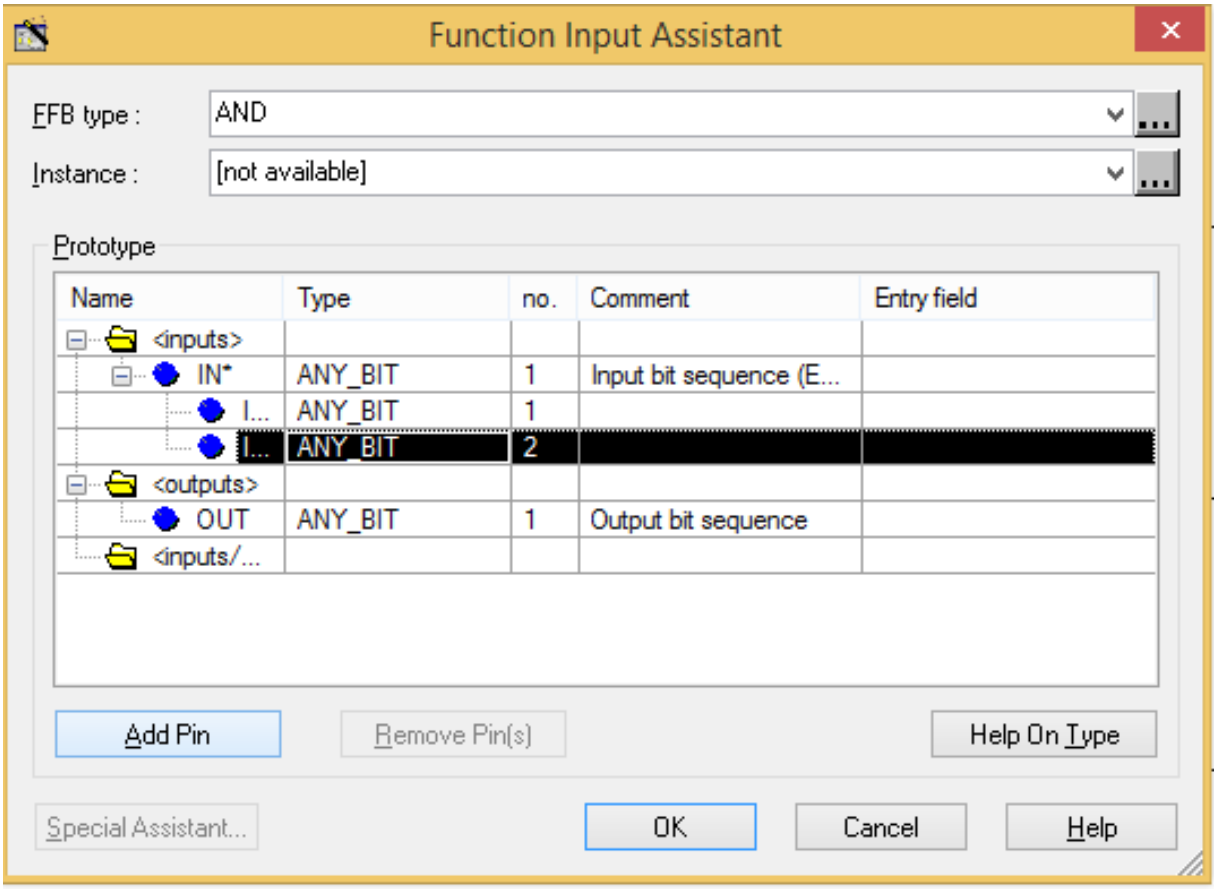
У овом поглављу детаљно ће бити описан програмски језик Функцијски блок дијаграм, односно биће описан сваки функцијски блок појединачно. Уз опис сваког блока дат је сликовит примјер чија се функционалност може провјерити на радној станици. Уколико је одабран програмски језик *FBD*, унос блокова може да се изврши на два начина. Један од начина је приказан на Сл. 51.



Слика 51 – Унос блока путем опције Data Selection

Притиском десног клика миша на површину предвиђену за програмирање отвара се понуђени мени. Након избора опције *Data Selection* потребно је унијети име блока. Након уноса имена, одабир блока потврђује се тастером *Enter*, те се блок поставља на радну површину.

Други начин уноса блокова на радну површину је избором опције *FFB Input Assistant* у истом менију. Отвара се нови прозор (Сл. 52) који олакшава тражење жељеног блока и даје прегледније информације о блоку.



Слика 52 – Унос блокова на радну површину путем FBB Input Assistant опције

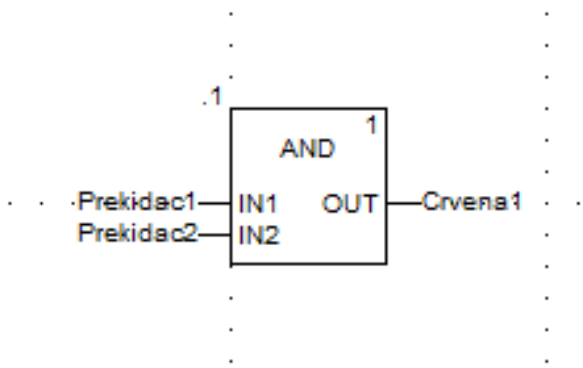
У поље *FBB Type* уноси се назив блока који се жели користити. Могуће је додавати, односно брисати одређене улазе са блока. Сваки блок када се постави на радну површину предвиђену за програмирање има своје име и на њему пише шта је функција појединих прикључака. Генерално, сваки блок има прикључке и то са лијеве стране улазе, а са десне излазе. Промјенљиве се придружују блоковима на начин да се двоструким кликом на одређени прикључак изабере жељена промјенљива чији тип мора одговарати типу прикључка.

3.6.1. Функцијски блокови логичких операција

Блоковима логичких операција припадају блокови:

- *AND* (логичко И),
- *OR* (логичко ИЛИ),
- *RS* (блок са доминантним ресетом),
- *SR* (блок са доминантним сетом),
- *F_TRIG* (блок за детекцију силазне ивице),
- *R_TRIG* (блок за детекцију узлазне ивице),
- *MOVE* (блок са уписивање вриједности),
- *XOR* (ексклузивно ИЛИ),
- *NOT* (блок негације),
- *SET* (блок за сетовање) и
- *RESET* (блок за ресетовање).

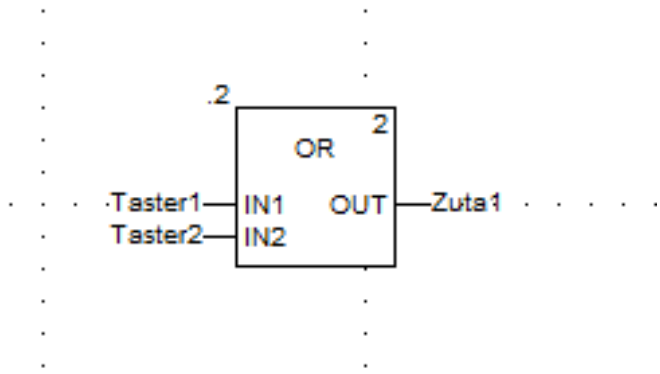
Блок *AND* може да садржи произвољан број улаза и само један излаз. И улази и излаз из блока су типа *BOOL*. Излаз ће бити логичка јединица само у случају када су сви улази истовремено логичка јединица (Сл. 53).



Слика 53 – Примјер коришћења функцијског блока AND

На примјеру са Сл. 53 црвена лампица ће бити активирана само у случају ако су активирани и прекидач 1 и прекидач 2.

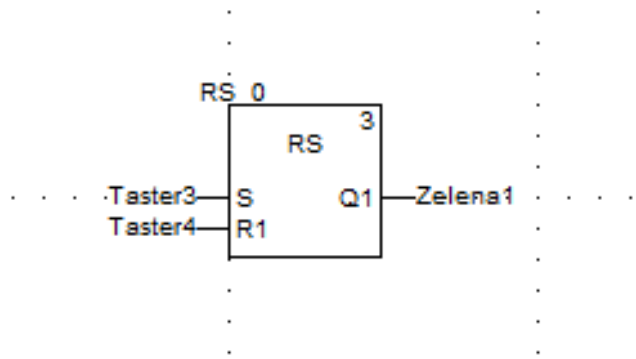
Блок *OR* може да садржи произвољан број улаза и само један излаз. Излаз ће бити логичка јединица у случају да је било који од улаза логичка јединица (Сл. 54). Улазне вриједности као и излаз су типа *BOOL*.



Слика 54 – Примјер коришћења функцијског блока OR

На примјеру са Сл. 54 жута лампица ће бити активирана у случају да је притиснут било који тастер или оба тастера истовремено.

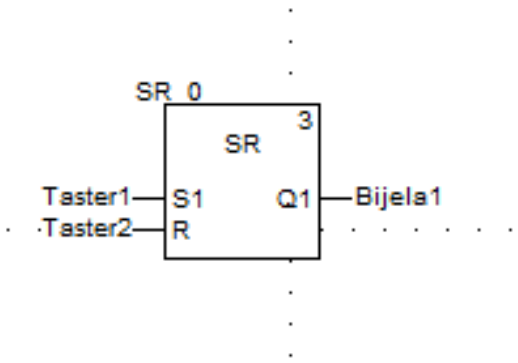
Блок *RS* садржи улазе сет, ресет и излаз. Све промјенљиве које се користе су типа *BOOL*. Блок *RS* има доминантан ресет (Сл. 55).



Слика 55 – Примјер коришћења функцијског блока RS

У примјеру са Сл. 55 довођењем логичке јединице на улаз *S* (коришћен је трећи тастер), зелена лампица ће бити активирана све до тренутка када се на улаз *R* (четврти тастер) не доведе логичка јединица. Уколико се на оба улаза истовремено доведе логичка јединица, излаз ће бити постављен на логичку нулу, јер функцијски блок *RS* има доминантан ресет.

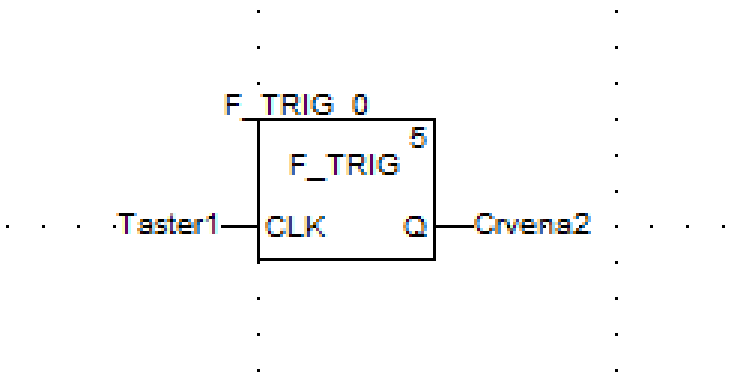
Блок *SR* има исту функцију као и *RS* са разликом да је сет доминантан (Сл. 56). Све промјенљиве су такође типа *BOOL*.



Слика 56 – Примјер коришћења функцијског блока SR

Притиском на први тастер, односно довођењем логичке јединице на улаз *S1* излаз из блока постаје логичка јединица (бијела лампица је активна). Уколико дође до промјене стања првог тастера са логичке јединице на логичку нулу, излаз и даље остаје логичка јединица. Бијела лампица престаје да емитује свјетлост у тренутку када се притисне други тастер, односно када се на улаз *R* доведе логичка јединица. Уколико су оба улаза логичка јединица, излаз остаје логичка јединица јер је улаз сет доминантан.

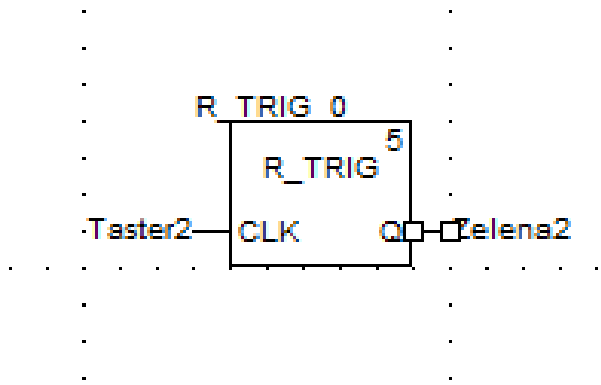
Функцијски блок *F_TRIG* поставља излаз на логичку јединицу уколико се на улазу детектује промјена са логичке јединице на логичку нулу. Излаз остаје активан само у једном програмском циклусу, те се може искористити као окидач за неки други функцијски блок ако су повезани. Примјер коришћења функцијског блока *F _TRIG* дат је на Сл. 57.



Слика 57 – Примјер коришћења функцијског блока F_TRIG

На примјеру са Сл. 57 приликом отпуштања првог тастера са контролне табле долази до промјене на улазу блока из стања логичке јединице на стање логичке нуле. Излаз постаје активан само у једном програмском циклусу.

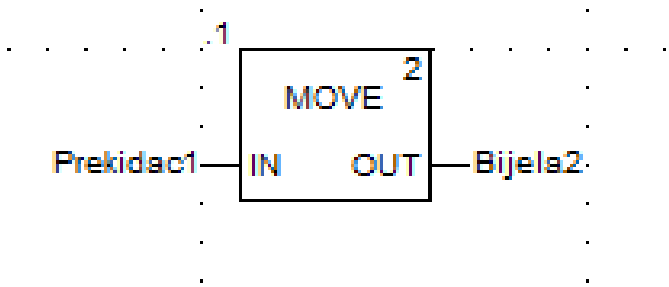
Функцијски блок *R_TRIG* поставља излаз на логичку јединицу уколико се на улазу детектује промјена са логичке нуле на логичку јединицу. Излаз остаје активан само у једном програмском циклусу, те се може искористити као окидач за неки други функцијски блок ако су повезани. Примјер коришћења функцијског блока *R_TRIG* дат је на Сл. 58.



Слика 58 – Примјер коришћења функцијског блока R_TRIG

На примјеру са Сл. 58 приликом притиска другог тастера на контролној табли долази до промјене из стања логичке нуле на стање логичке јединице. Излаз постаје краткотрајно активан, односно само у једном програмском циклусу.

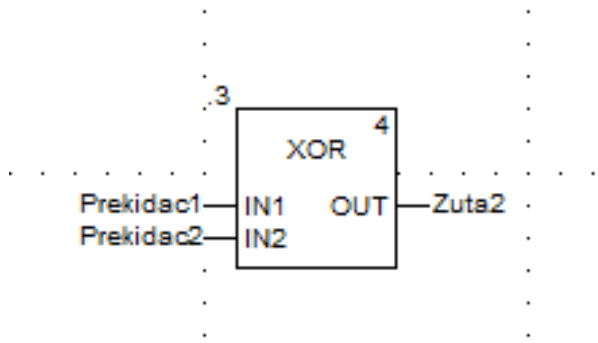
Функцијски блок *MOVE* користи се за упис вриједности улазне промјенљиве на адресу излазне промјенљиве (Сл. 59).



Слика 59 – Примјер коришћења функцијског блока MOVE

Уколико је прекидач на стању логичке јединице, бијела лампица ће да се укључи. Функцијски блок *MOVE* користи се за било коју врсту података, а не само за *BOOL*. Треба обратити пажњу да су улазна и излазна промјенљива истог типа.

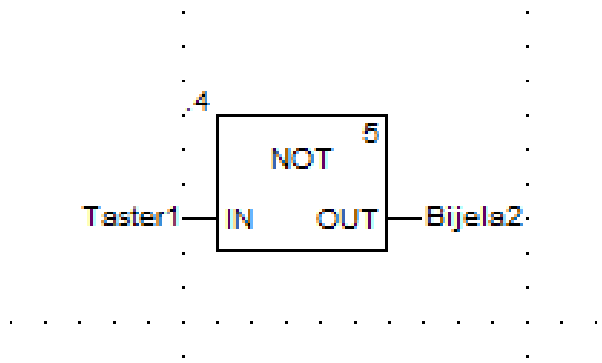
Функцијски блок *XOR* има више улаза и један излаз. Улазне и излазне промјенљиве су типа *BOOL*. Излаз има стање логичке јединице само у случају када су улази на различитим логичким нивоима (Сл. 60).



Слика 60 – Примјер коришћења функцијског блока XOR

На Сл. 60 жута лампица ће бити активирана само уколико је један прекидач укључен, а други искључен.

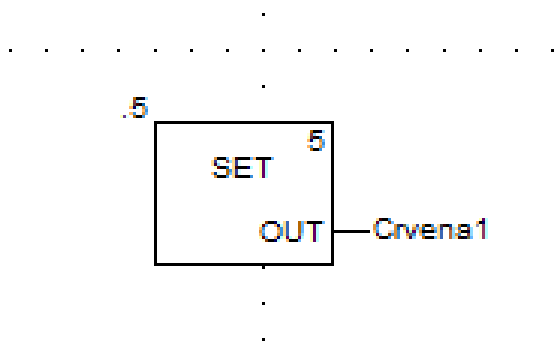
Функцијски блок *NOT* на излазу даје комплементирану вриједност улаза (Сл. 61). Има један улаз и један излаз и они морају бити истог типа. Користи се за комплементирање улазне промјенљиве бит по бит, а може да се користи за промјенљиве типа *BOOL, BYTE, WORD* и *DWORD.*



Слика 61 – Примјер коришћења функцијског блока NOT

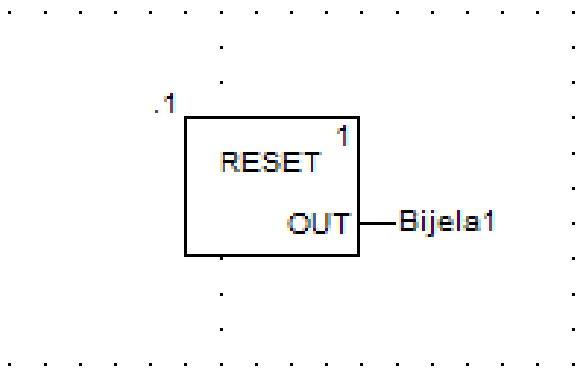
На Сл. 61 уколико је тастер притиснут, лампица ће бити искључена и обрнуто.

Функцијски блок *SET* користи се за постављање излазне промјенљиве на логичку јединицу. Примјер коришћења блока *SET* приказан је на Сл. 62.



Слика 62 – Примјер коришћења функцијског блока SET

Функцијски блок *RESET* користи се за постављање излазне вриједности на логичку нулу. Излаз је типа *BOOL,* као и у случају коришћења функцијског блока *SET*. Примјер коришћења функцијског блока *RESET* приказан је на Сл. 63.



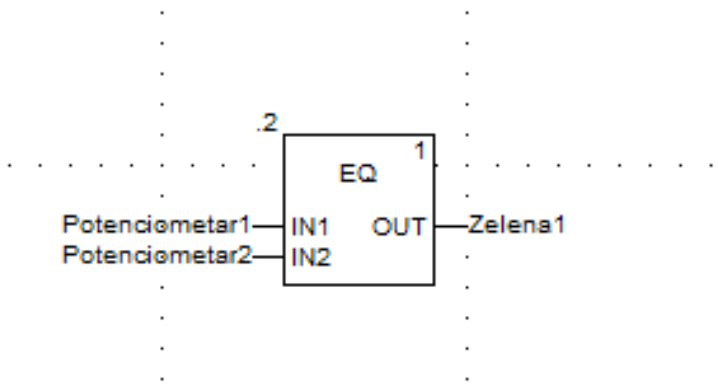
Слика 63 – Примјер коришћења функцијског блока RESET

3.6.2. Функцијски блокови поређења

У функцијске блокове поређења се убрајају блокови:

- *EQ* енгл. *Equal To* (једнако),
- *GE* енгл. *Greater Than Or Equal To* (веће или једнако),
- *GT* енгл. *Greater Than* (веће од),
- *LE* енгл. *Less Than Or Equal To* (мање или једнако),
- *LT* енгл. *Less Than* (мање од) и
- *NE* енгл. *Not Equal To* (неједнако).

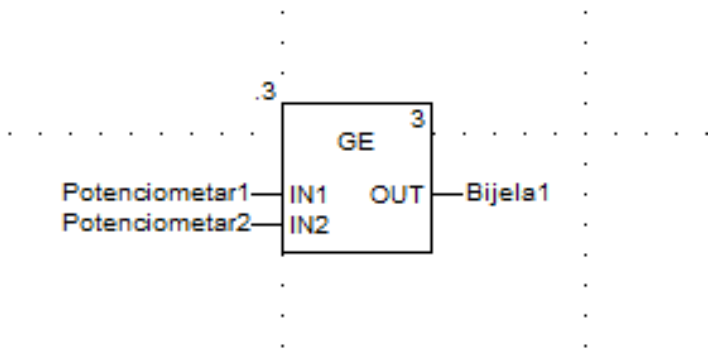
Функцијски блок поређења *EQ* има вриједност излаза логичку јединицу уколико улази имају исту вриједност. Излаз из функцијског блока је типа *BOOL*, док улази могу бити било којег типа али обавезно истог. Број улазних вриједности се може проширити на максимално 32 улазна сигнала. Примјер коришћења функцијског блока *EQ* приказан је на Сл. 64.



Слика 64 – Примјер коришћења функцијског блока поређења EQ

На примјеру приказаном на Сл. 64 зелена лампица ће бити активна само у случају када су вриједности потенциометара једнаке.

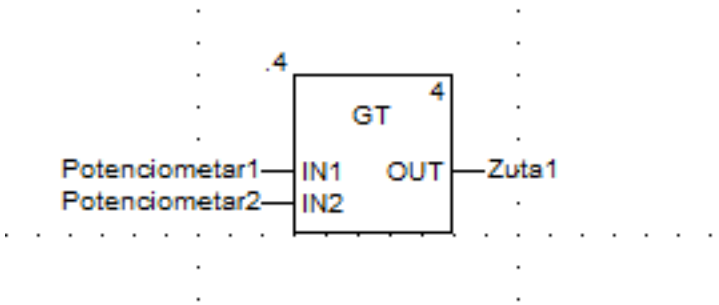
Функцијски блок поређења *GE* на излазу даје логичку јединицу уколико је улаз *IN1* већи или једнак од улаза *IN2* (Сл. 65). Типови промјенљивих улазних величина морају бити истог типа, а број могућих улаза је 32.Улази се пореде секвенцијално, односно први улаз се пореди са сваким сљедећим улазом.



Слика 65 – Примјер коришћења функцијског блока поређења GE

На примјеру приказаном на Сл. 65 бијела лампица ће бити активна док год је вриједност потенциометра 1 већа или једнака од вриједности потенциометра 2.

Функцијски блок поређења *GT* на излазу даје логичку јединицу уколико је улазни доведени сигнал на првом улазу већи од улазног сигнала на другом или осталим улазима. Број улаза је ограничен на 32, а улазне величине морају бити истог типа. Принцип функционисања блока *GT* ради на принципу поређења улазних вриједности секвенцијално, односно први улаз мора бити већи од свих улаза како би на излазу била логичка јединица (Сл. 66).

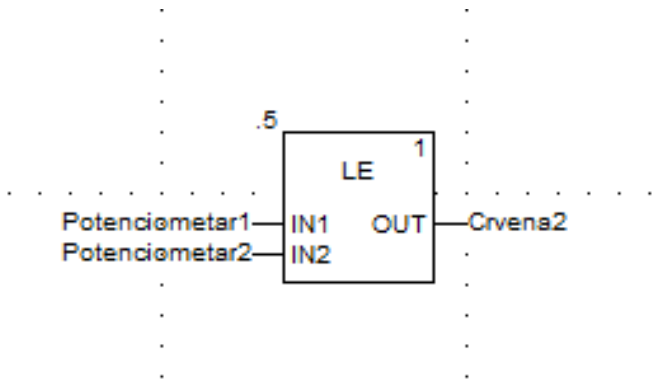


Слика 66 – Примјер коришћења функцијског блока поређења GT

На примјеру приказаном на Сл. 66 жута лампица ће бити активна све док је вриједност потенциометра 1 већа од вриједности потенциометра 2.

Функцијски блок поређења *LE* на излазу даје логичку јединицу уколико је вриједност улаза на прикључку *IN1* мања или једнака вриједности на осталим улазним

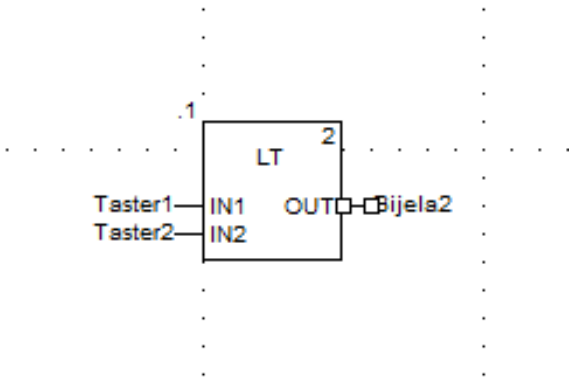
прикључцима. Улазне промјенљиве морају бити истог типа, а тип излазне промјенљиве је *BOOL*. Примјер коришћења функцијског блока *LE* поређења приказан је на Сл. 67.



Слика 67 – Примјер коришћења функцијског блока поређења LE

На примјеру приказаном на Сл. 67 црвена лампица ће бити активна само уколико је вриједност потенциометра 1 мања или једнака вриједности потенциометра 2.

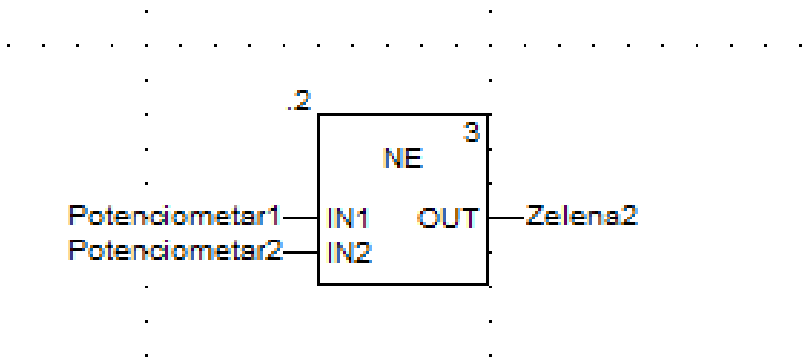
Функцијски блок *LT* врши поређење улазних промјенљивих истог типа. Излаз постаје логичка јединица уколико је вриједност улазног сигнала на првом улазу (*IN1*) мања од свих осталих улазних вриједности. Максималан број улазних вриједности је 32. У примјеру на Сл. 68 пореде се улазне вриједности типа *BOOL*.



Слика 68 – Примјер коришћења функцијског блока поређења LT

Из примјера са Сл. 68 врши се поређење улазних вриједности. Да би излаз био логичка јединица, други тастер мора бити активиран (логичка јединица), а први тастер неактиван (логичка нула).

Посљедњи функцијски блок из групе функцијских блокова поређења је блок *NE*. Функцијски блок поређења *NE* на излазу из блока даје логичку јединицу уколико улазне промјенљиве имају различите вриједности. Улази морају бити истог типа, а може их бити максимално 32. Примјер коришћења функцијског блока поређења *NE* дат је на Сл. 69.



Слика 69 – Примјер коришћења функцијског блока поређења NE

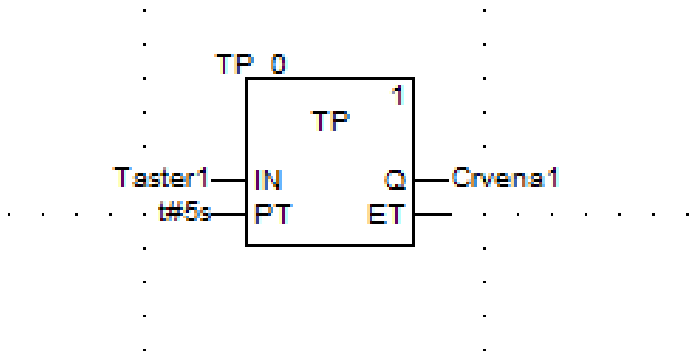
На примјеру приказаном на Сл. 69 зелена лампица ће бити активна, све док су вриједности потенциометра 1 и потенциометра 2 различите.

3.6.3. Тајмери

Тајмери су функцијски блокови који се користе за мјерење времена. Функцијски блокови који обављају функцију тајмера су:

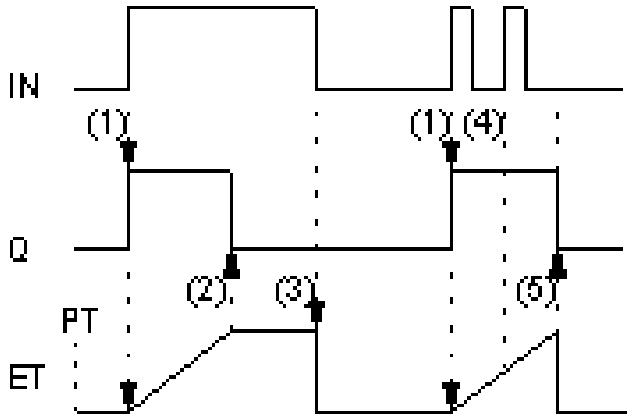
- TP енгл. *Pulse* (пулс тајмер),
- TON енгл. *On Delay* (кашњење при укључењу) и
- TOF енгл. *Off Delay* (кашњење при искључењу).

Функцијски блок *TP* користи се као генератор пулса жељеног трајања. На улазу у функцијски блок *TP* налази се улаз који је типа *BOOL* и прикључак *PT* у који се уноси жељено вријеме трајања пулса. На излазу из блока налази се излаз типа *BOOL* и промјенљива *ET* која садржи тренутну вриједност тајмера. На Сл. 70 приказан је примјер коришћења тајмера *TP*.



Слика 70 – Примјер коришћења тајмера TP

У тренутку када дође до промјене улазне вриједности са стања логичке нуле на стање логичке јединице, излаз постаје логичка јединица све до истека подешеног времена које је задано на прикључку *PT*. Тип промјенљивих *PT* и *ET* су типа *TIME*. На примјеру са Сл. 70. притиском првог тастера на контролној табли, црвена лампица постаје активна пет секунди. На Сл. 71 приказан је временски дијаграм улаза и излаза тајмера *TP*.

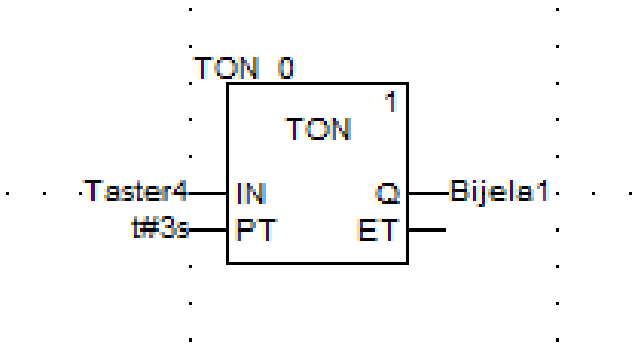


Слика 71 – Временски дијаграм тајмера TP

Са Сл. 71 се уочава да ће излаз имати стање логичке јединице све док вријеме *ET* не достигне вриједност *PT,* независно да ли се на улазу поново доведе логичка јединица или не. Уколико је тастер један на високом логичком нивоу и након проласка задатог времена

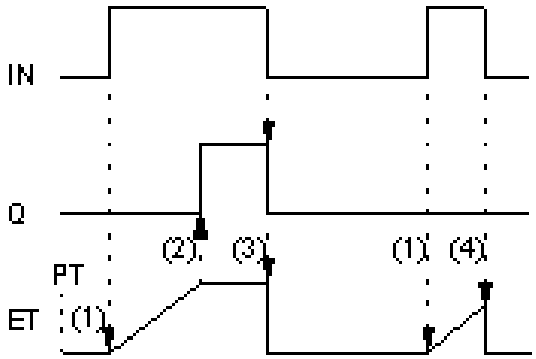
PT, излаз ће се вратити на логичку нулу, а промјенљива *ET* ће задржавати своју вриједност све док улаз не промијени вриједност са логичке јединице на логичку нулу.

Други функцијски блок који се користи као тајмер је *TON.* Тајмер *TON* се користи као тајмер који касни при укључењу. Као и код претходног тајмера *TP*, тајмер има један улаз типа *BOOL* који се користи за стартовање тајмера. Жељено вријеме се задаје на улазу *PT.* Примјер коришћења тајмера *TON* приказан је на Сл. 72, док је временски дијаграм приказан на Сл 73.



Слика 72 – Примјер коришћења тајмера TON

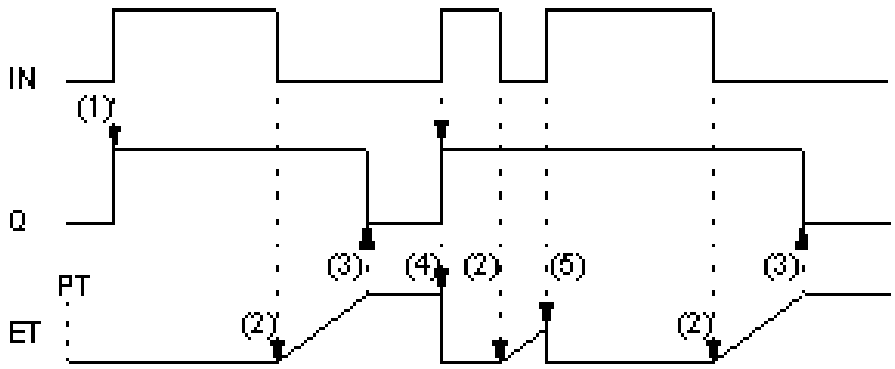
За примјер са Сл. 72 као улаз коришћен је четврти тастер на контролној табли, те је подешено вријеме 3 s послије којег долази до укључења излаза.



Слика 73 – Временски дијаграм тајмера TON

Анализирајући временски дијаграм са Сл. 73 види се да промјеном стања улаза са логичке нуле на логичку јединицу, долази до почетка бројања времена на излазу *ET*. Када вријеме измјерено на *ET* достигне вриједност времена задатог на улазној страни блока *PT*, излаз мијења своје стање на логичку јединицу. Излаз је активан све до тренутка када улаз промијени своје стање на логичку нулу. Уколико вријеме трајања улазног импулса није довољно дуго да вријеме *ET* достигне задано вријеме *PT*, излаз из блока се неће активирати. Примјеном анализе временског дијаграма на примјер са Сл. 72 закључује се да дужина трајања логичке јединице на улазу у блок *TON*, мора бити дуже од три секунде, како би се активирала бијела лампица. Такође, бијела лампица ће бити укључена са кашњењем од три секунде у односу на команду на улазу.

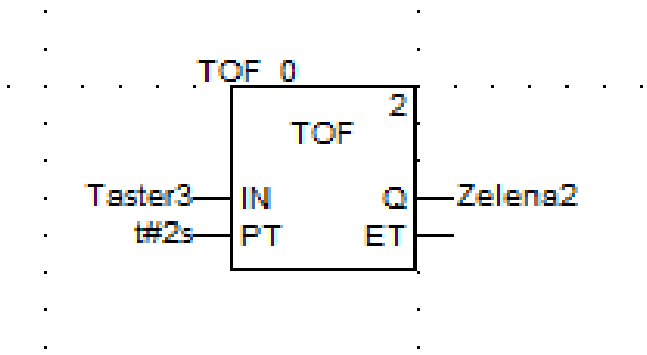
Посљедњи функцијски блок који припада тајмерима је *TOF*. Користи се као тајмер са кашњењем на искључењу. Улази и излази из тајмера *TOF* имају исте типове промјенљивих као и тајмери *TON* и *TP*. Временски дијаграм *TOF* функцијског блока приказан је на Сл. 74.



Слика 74 – Временски дијаграм тајмера TOF

Са временског дијаграма приказаног на Сл. 74 може се закључити да довођењем логичке јединице на улазу блока, излаз тајмера постаје логичка јединица. У тренутку преласка улазног сигнала са вриједности логичке јединице на логичку нулу, излаз и даље остаје активиран, а почиње мјерење времена на излазу *ET*. У тренутку када се избројано вријеме *ET* изједначи са временом *PT*, излаз са тајмера мијења своју вриједност на логичку нулу, а вријеме *ET* задржава измјерену вриједност времена све док не дође до поновног укључења улазног сигнала.

У случају да се након преласка улазног сигнала са логичке јединице на логичку нулу не стигну изједначити избројано вријеме *ET* и задато вријеме *PT*, а у међувремену се поново деси промјена улазне вриједности на логичку јединицу, улаз ће остати активан, а вријеме *ET* ће се ресетовати. Излаз ће промијенити вриједност логичке јединице у логичку нулу, тек када вријеме *ET* достигне задато вријеме *PT*. Примјер коришћења тајмера *TOF* приказан је на Сл. 75.



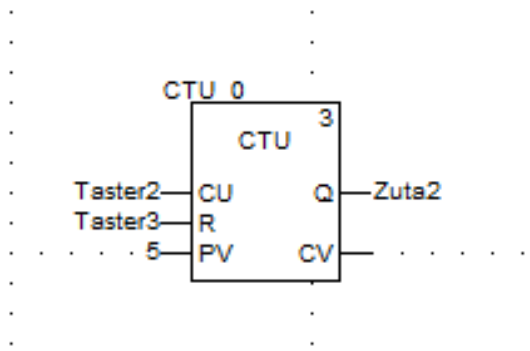
Слика 75 – Примјер коришћења тајмера TOF

Притиском трећег тастера на контролној табли укључиће се зелена лампица која остаје укључена двије секунде од отпуштања тастера. Лампица ће радити како је описано под условом да се поновна активација тастера не деси у краћем временском интервалу од задате двије секунде.

3.6.4. Бројачи

- У функцијске блокове који обављају функцију бројања, односно бројачи (енгл. *Counters*) убрајају се:
- *CTU* енгл. *Counter Up* (бројач унапријед),
- *CTD* енгл. *Counter Down* (бројач уназад),
- *CTUD* енгл. *Counter Up Down* (бројач унапријед и уназад).

Функцијски блок *CTU* је бројач унапријед. Примјер коришћења бројача *CTU* приказан је на Сл. 76.



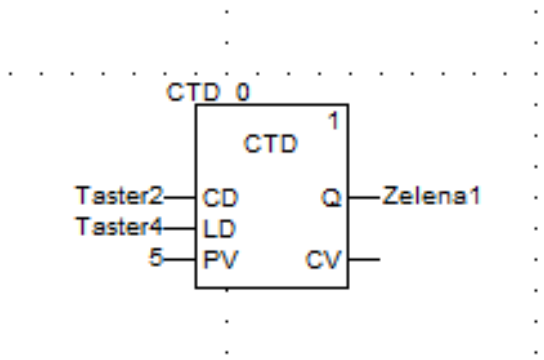
Слика 76 – Примјер коришћења бројача CTU

Улази *CU* и *R*, као и излаз *Q* су типа *BOOL*. Прикључци *PV* и *CV* су типа *INT*. Бројач на улазу *CU* региструје промјену логичке нуле на логичку јединицу, те за сваки избројану промјену на улазу, повећава вриједност на излазу *CV* за један (инкрементира). Када вриједност *CV* достигне вриједност задату на *PV*, излаз *Q* се поставља на логичку јединицу.

Промјена стања са логичке нуле на логичку јединицу на улазу *R* (ресет) долази до враћања вриједности *CV* на нулу као и излаза *Q*.

У примјеру са Сл. 76 притиском другог тастера пет пута долази до активирања жуте лампице. Излаз бити активан све док се тастером три на контролној табли не ресетује бројач. Притискањем другог тастера више од пет пута, вриједност промјенљиве *CV* ће се повећавати, а жута лампица остаје активна док се бројач не ресетује.

Бројач *CTD* је бројач уназад. Примјер коришћења бројача уназад приказан је на Сл. 77.

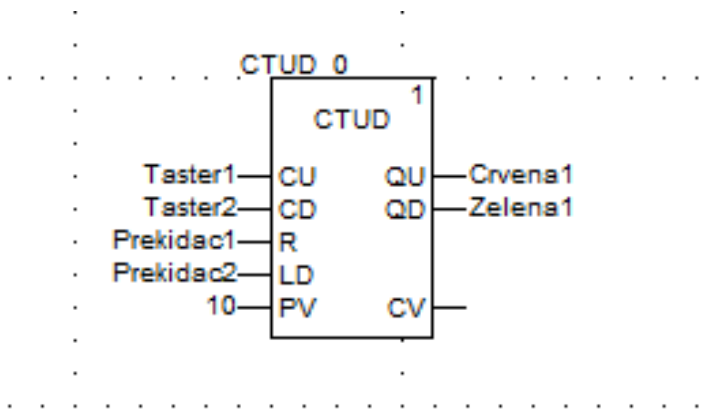


Слика 77 – Примјер коришћења бројача CTD

Улази *CD* и *LD* као и излаз *Q* су типа *BOOL*. Промјенљиве типа *PV* и *CV* су типа *INT*. Регистровањем логичке јединице на улазу *LD* вриједност из *PV* шаље се на излаз *CV*. Сваком промјеном са логичке нуле на логичку јединицу улаза *CD* долази до умањења вриједности *CV* за један (декрементира). Када вриједност *CV* постане једнака или мања од нуле, излаз *Q* постаје логичка јединица.

На примјеру са Сл. 77 подешена је вриједност пет на улаз *PV*. Активирањем четвртог тастера са контролне табле долази до уписивања вриједности пет у *CV*. Активирањем другог тастера пет пута, вриједност *CV* смањује се на нулу, те се активира зелена лампица. Притискањем другог тастера више од пет пута, вриједност промјенљиве *CV* наставља да се умањује, а зелена лампица остаје активна све док се на прикључак *LD* не доведе логичка јединица.

Посљедњи функцијски блок који ради ако бројач је бројач који омогућава бројање и уназад и унапријед, односно *CTUD*. Примјер коришћења бројача *CTUD* приказан је на Сл. 78.



Слика 78 – Примјер коришћења бројача CTUD

Бројач *CTUD* представља комбинацију претходно описана два бројача. Све улазне, односно излазне вриједности су типа *BOOL*, осим *PV* и *CV* које су типа *INT*. У примјеру на Сл. 78 у вриједност *PV* унесена је вриједност десет. Окретањем другог прекидача на контролној табли врши се пренос вриједност са *PV* на *CV*. Прекидач један користи се за ресет промјенљиве *CV*. Први тастер се користи за инкрементирање *CV*, док се други тастер користи за декрементирање *CV*.

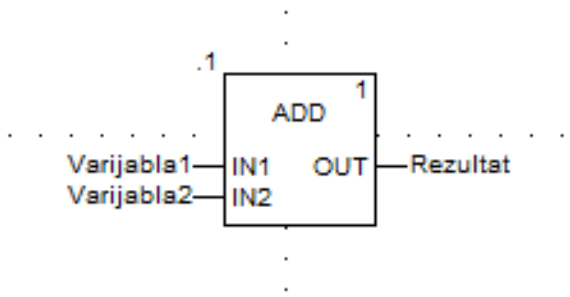
Када вриједност *CV* достигне вриједност *PV* долази до активирања излаза *QU*, док се излаз *QD* активира у тренутку када се вриједност *CV* смањи на нулу, или ако је мања од нуле. Уколико дође до истовременог довођења логичке јединице на улазе *R* и *LD*, доминантан је улаз *R*.

3.6.5. Функцијски блокови основних математичких операција

У функцијске блокове основних математичких операција убрајају се:

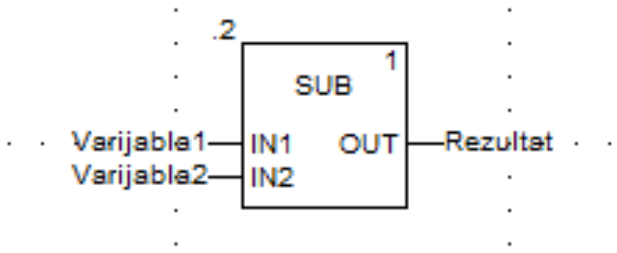
- ADD енгл. *Addition* (сабирање),
- SUB енгл. *Substraction* (одузимање),
- MUL енгл. *Multiplication* (множење),
- DIV енгл. *Divison* (дијељење),
- ABS енгл. *Absolute value computation* (апсолутна вриједност),
- NEG енгл. *Negation* (негација).

Функцијски блок *ADD* користи се за операцију сабирања. Улази као и излаз морају да буду истог типа (*INT, DINT, UDINT, UINT* или *REAL*). Максималан број улаза је 32. На Сл. 79 приказан је функцијски блок *ADD.*



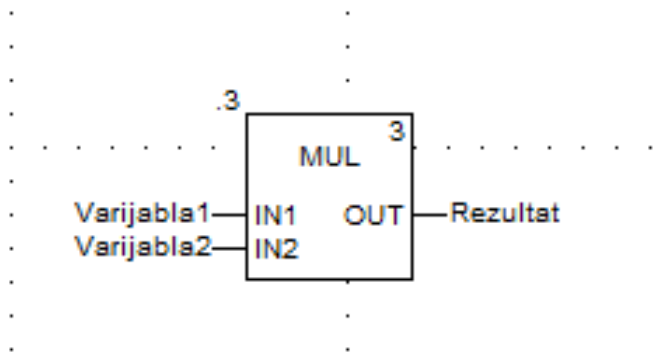
Слика 79 – Функцијски блок ADD

Функцијски блок *SUB* користи се за математичку операцију одузимања. На Сл. 80 приказан је функцијски блок *SUB.* Све вриједности коришћене за одузимање морају имати исти тип податка.



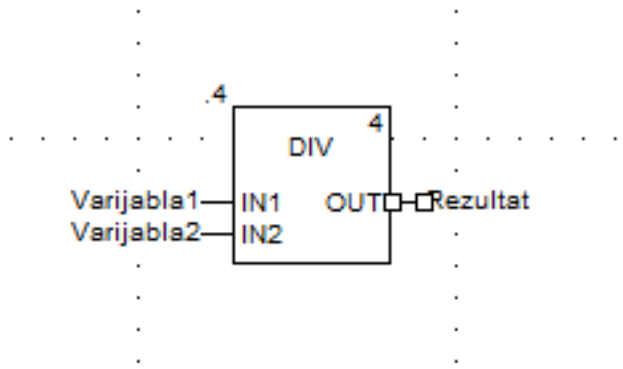
Слика 80 – Функцијски блок SUB

Функцијски блок *MUL* користи се за математичку операцију множења (Сл. 81). Максималан број улаза је 32.



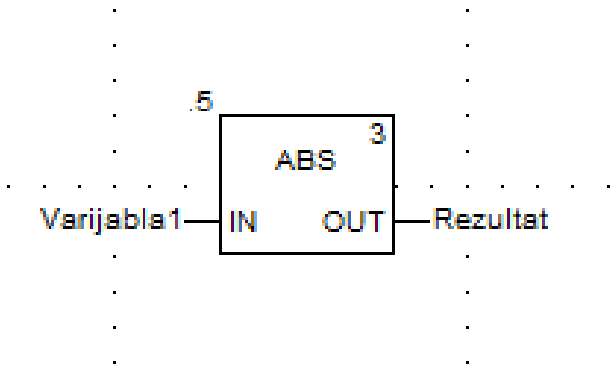
Слика 81 – Функцијски блок MUL

Задатак функцијског блока *DIV* (Сл. 82) је дијељење двије вриједности које се доведу на улаз у блок. На излазу је резултат математичке операције истог типа као и улази.



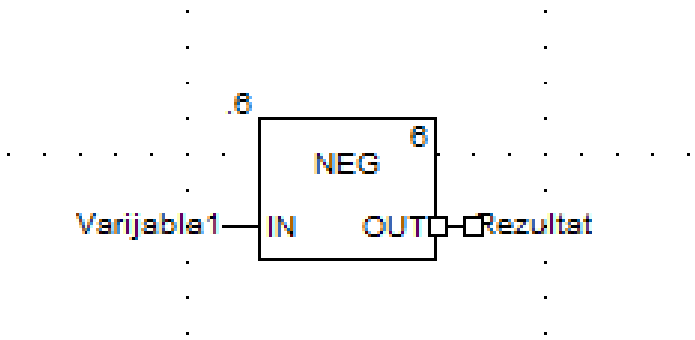
Слика 82 – Функцијски блок DIV

Блок *ABS* користи се за рачунање апсолутне вриједности улазне промјенљиве (Сл. 83). Тип податка на излазу је мора бити идентичан типу податка на улазу.



Слика 83 – Функцијски блок ABS

Посљедњи функцијски блок основних математичких операција је блок *NEG* и користи се за добијање негативне вриједности улазне промјенљиве (Сл. 84). Тип податка на излазу мора бити идентичан типу податка који се доведе на улазу.



Слика 84 – Функцијски блок NEG

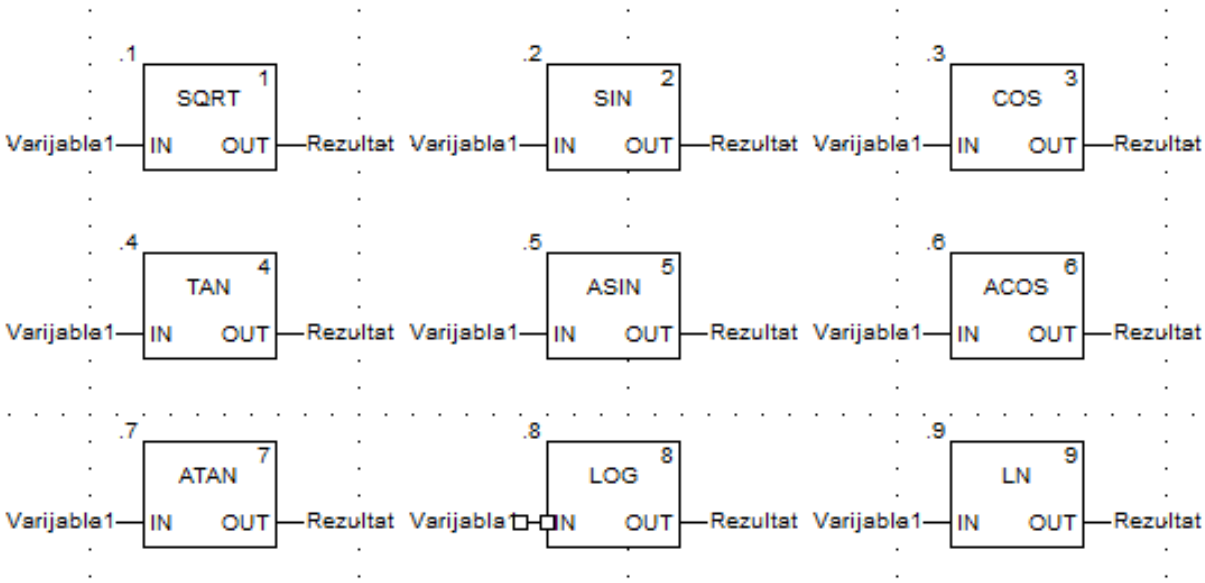
3.6.6. Функцијски блокови сложених математичких операција

У функцијске блокове сложених математичких функција убрајају се блокови:

- *SQRT* енгл. *Square root* (квандратни коријен),
- *SIN* енгл. *Sine* (синус),
- *COS* енгл. *Cosine* (косинус),
- *TAN* енгл. *Tangent* (тангенс),
- *ASIN* енгл. *Arc Sine* (аркус синус),

- *ACOS* енгл. *Arc Cosine* (аркус косинус),
- *ATAN* енгл. *Arc Tangent* (аркус тангенс),
- *LOG* енгл. *Base 10 Logarithm* (логаритам базе 10) и
- *LN* енгл. *Natural Logarithm* (природни логаритам).

Наведени блокови имају један улаз и један излаз. Функцијски блокови сложених математичких операција обављају математичке функције у складу са називом блока, те из тог разлога неће бити детаљно описани у овом поглављу. На Сл. 85 приказани су сви блокови који се користе за обављање сложених математичких операција у програмском језику *FBD*.



Слика 85 – Примјер коришћења блокова сложених математичких операција

Потребно је нагласити основна правила при коришћењу блокова сложених математичких функција.

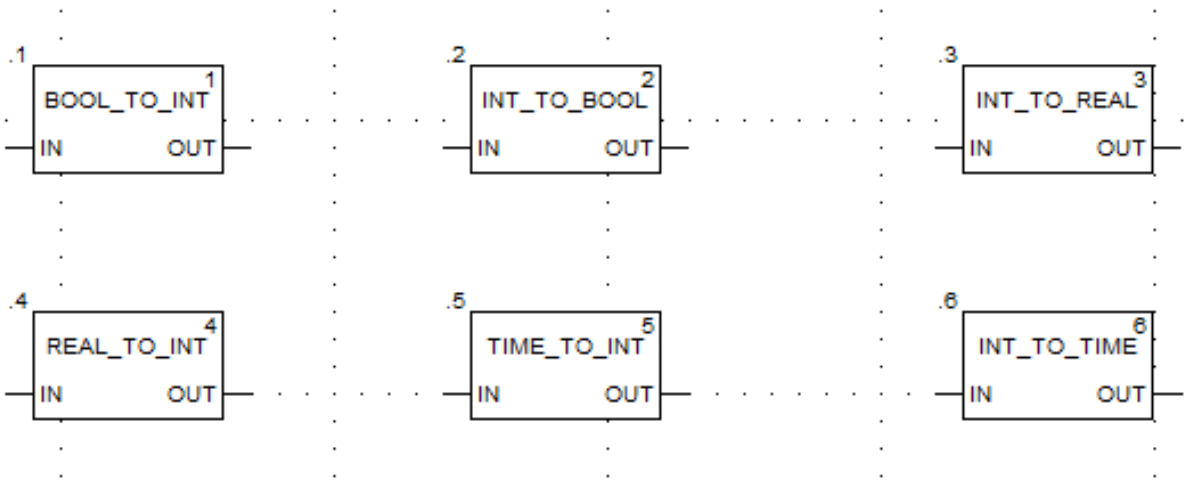
Функцијски блок *SQRT* може се користити за промјенљиве типа *INT, DINT* и *REAL*. Излаз из блока мора имати исти тип промјенљиве као улазна промјенљива.

Улазна промјенљива функцијских блокова тригонометријских функција *SIN, COS, TAN, ASIN, ACOS* и *ATAN* је угао у радијанима.

Улазне и излазне промјенљиве су најчешће типа *REAL*, јер се извршавање набројаних математичких операција углавном врши за вриједности са децималном тачком.

3.6.7. Функцијски блокови за конверзију промјенљивих

Конверзија између различитих типова промјенљивих описана је у поглављу 3.3 под називом *„Конверзија промјенљивих“*. Правила по којима се врше конверзије детаљно су описане, тако да ће на Сл. 86 бити само приказани функцијски блокови који се најчешће користе.

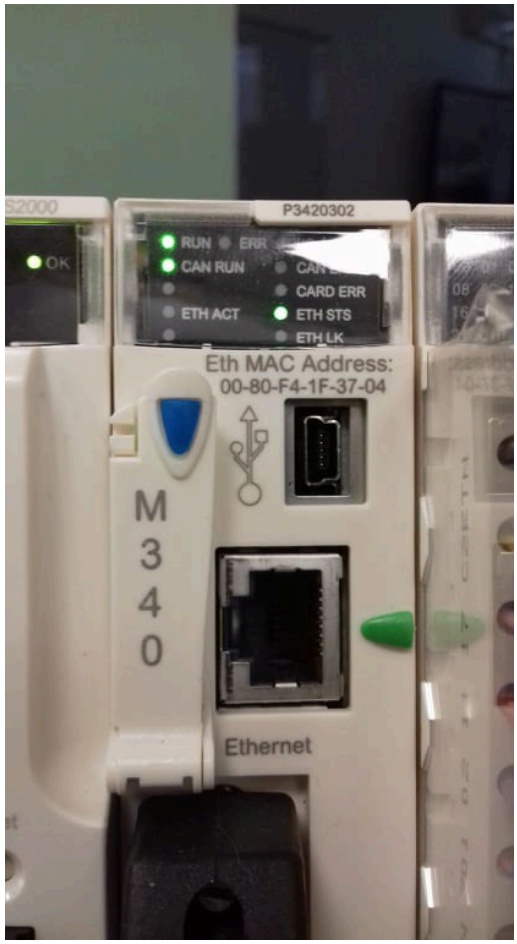


Слика 86 – Функцијски блокови за конверзију промјенљивих

Поред описаних функцијских блокова постоје и специјални функцијски блокови који се користе при повезивању ПЛК-а са фреквенцијским регулатором путем *CANopen* комуникације, као и функцијски блокови који се користе за управљање фреквенцијским регулатором, односно асинхроним мотором. Детаљан опис поменутих блокова биће дат у поглављу о повезивању ПЛК-а и фреквенцијског регулатора.

3.7. Повезивање ПЛК-а и рачунара

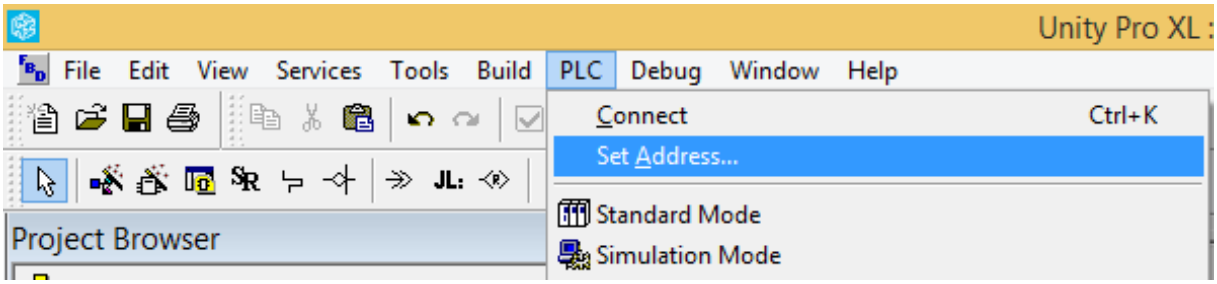
По завршетку писања програма, програм је потребно отпремити на програмабилни логички контролер. Уграђени процесорски модул пружа кориснику повезивање са рачунаром на два начина, а то су *USB* и *TCP IP* протокол (путем *Ethernet-а*). На Сл. 87 приказани су предвиђени портови на процесорском модулу за повезивање са рачунаром, као и са свим осталим уређајима који могу да остваре комуникацију са ПЛК-ом на неки од два наведена начина.



Слика 87 – Комуникациони портови на процесорском модулу ПЛК-а који се могу користити за повезивање са рачунаром

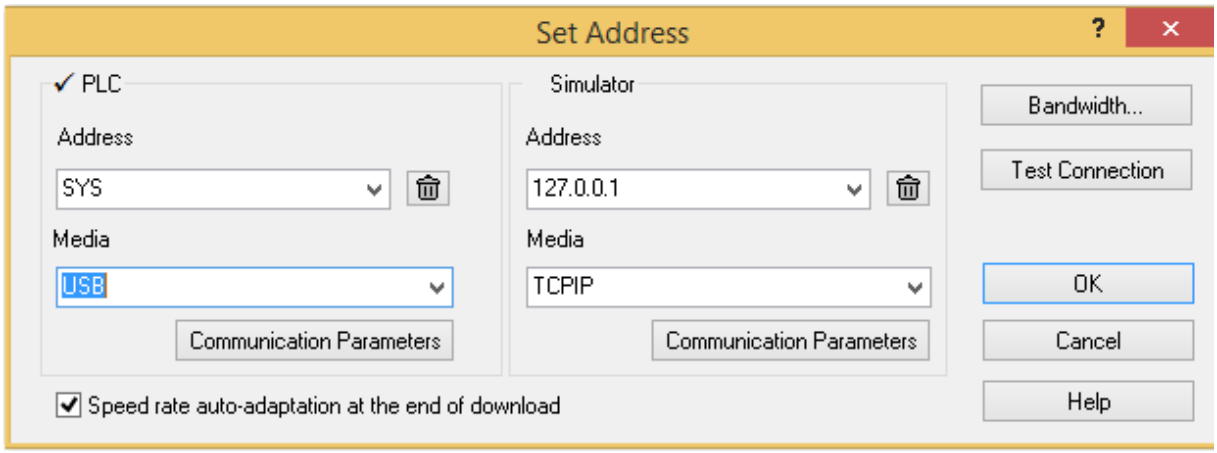
3.7.1. Повезивање ПЛК-а и рачунара путем *USB* комуникације

ПЛК и рачунар потребно је повезати одговарајућим каблом предвиђеним за *USB* протокол. Да би комуникација била могућа, потребно је софтверски поставити одређене параметре који дефинишу комуникацију између два уређаја. У софтверу *Unity Pro XL* на картици *PLC* потребно је одабрати *Set Address* (Сл. 88).



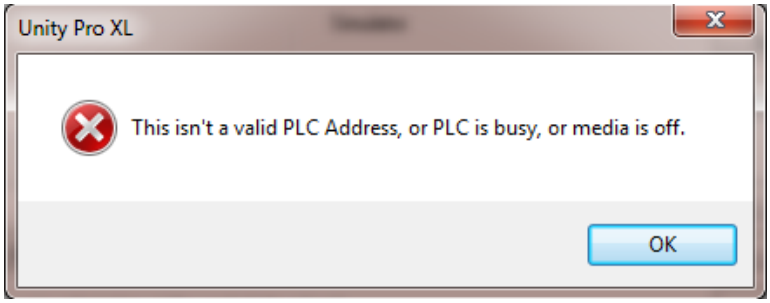
Слика 88 – Подешавање комуникације ПЛК-а и рачунара

Одабиром *Set Address* отвара се прозор као на Сл. 89. У пољу *Address* бира се *SYS,* док се у пољу *Media* (лијева страна прозора) бира *USB*. Сљедећи корак је опција *Test Connection.*



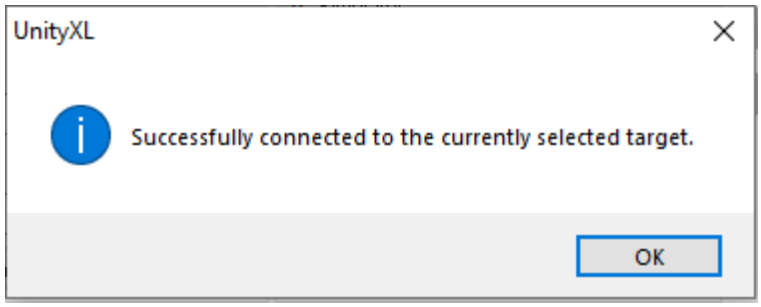
Слика 89 – Унос параметара за USB комуникацију

Уколико нису правилно одабрани комуникациони параметри, или нису правилно инсталирани сви *USB* драјвери, исписаће се порука приказана на Сл. 90. Драјвери могу да се нађу на линку https://schneider-electric.box.com/s/2g6x5er43rxdalczqc3uqhvt52rueku7.



Слика. 90 – Грешка комуникације између ПЛК-а и рачунара

Уколико су правилно инсталирани сви *USB* драјвери који се могу преузети са официјалног сајта произвођача Шнајдер Електрик, након одабира опције *Test Connection,* исписаће се порука приказана на Сл. 91.



Слика 91 – Потврда успостављене комуникације између ПЛК-а и рачунара

Након добијања потврде од стране софтвера *Unity Pro XL* да је комуникација омогућена, потребно је на картици *PLC* (Сл. 88) одабрати опцију *Connect*, те је процес повезивања рачунара и ПЛК-а завршен.

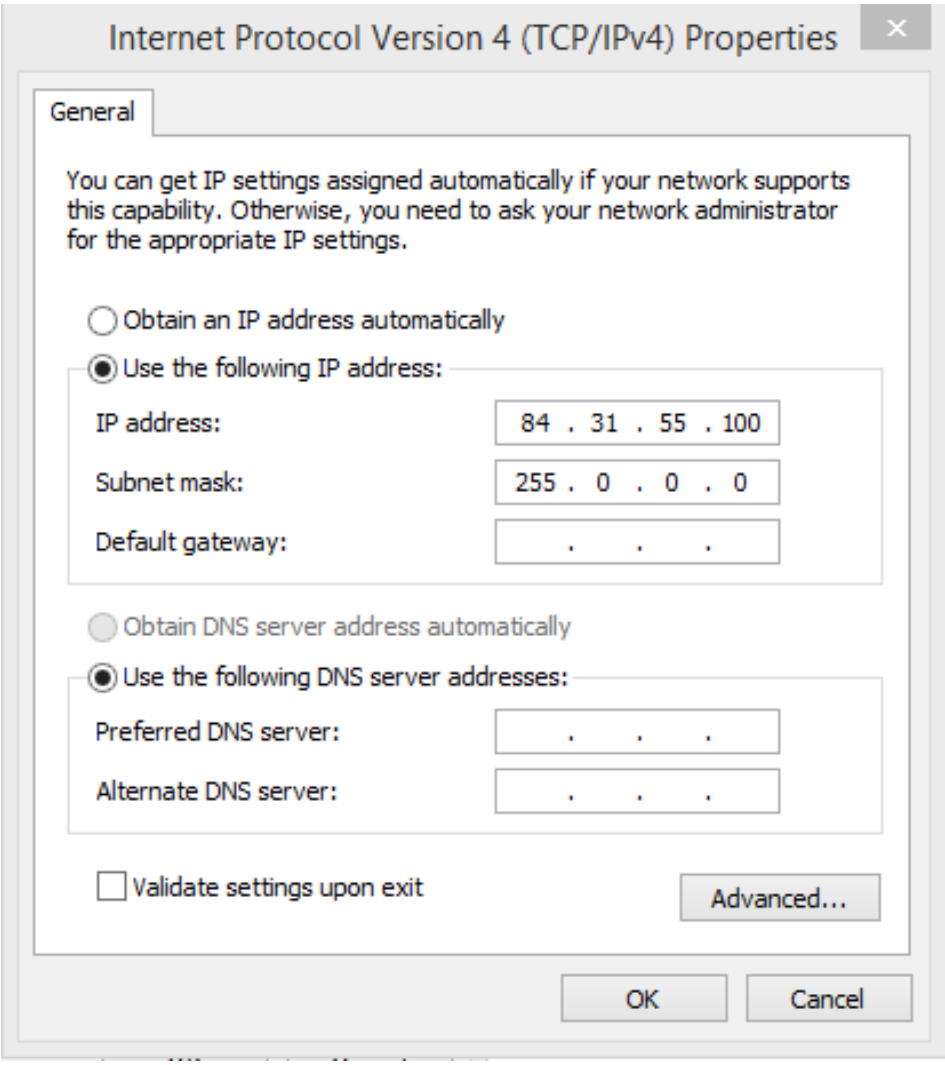
3.7.2. Повезивање ПЛК-а и рачунара путем *Ethernet-а*

ПЛК и рачунар потребно је повезати одговарајућим каблом предвиђеним за Етернет комуникацију. Софтверско подешавање се врши избором опције *Set Address* у картици *PLC* (Сл. 88). У пољу *Media* (лијева страна прозора) бира се *TCPIP*, што представља *Ethernet* комуникацију, док се у пољу *Adress* уписује *IP* адреса ПЛК-а (Сл 93).

IP адреса ПЛК-а се добија из његове *MAC* адресе која је записана на процесорском модулу (Сл. 87). За ПЛК Модикон М340 *IP* адреса је формата „84.X.X.X“. Број 84 је увијек фиксан за дати процесорски модул, док се сљедећа три броја адресе добијају претварањем посљедња три броја *MAC* адресе из хексадецималног у декадни бројни систем. На примјер, ако је *MAC* адреса ПЛК-а 00-80-F4-1F-37-04 потребно је претворити

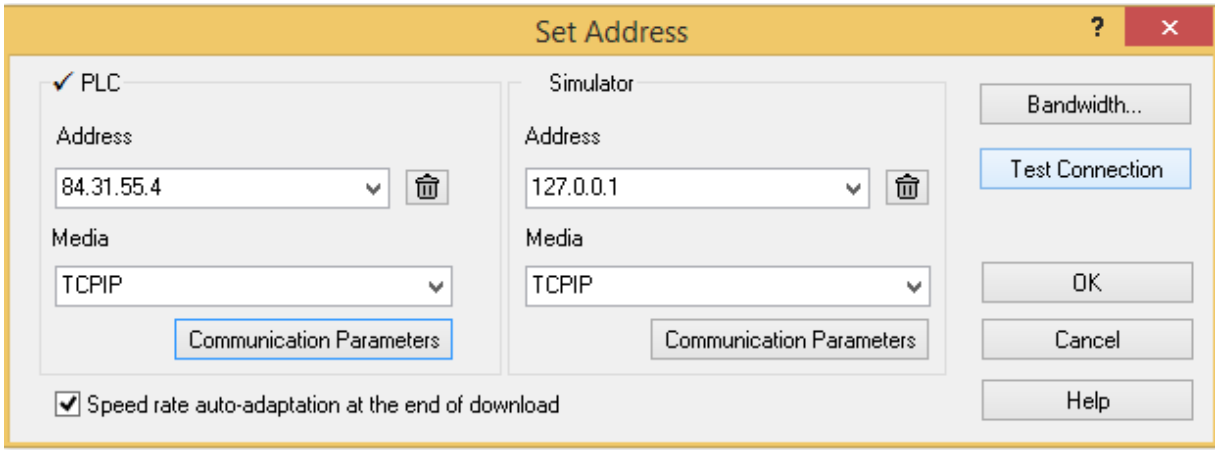
хексадецималне вриједности 1F.37.04 у децимални облик. Након конверзије посљедња три броја адресе постају 31.55.4. па је *IP* адреса ПЛК-а 84.31.55.4.

Да би комуникација између ПЛК-а и рачунара била могућа, потребно је промијенити статичку *IP* адресу рачунара. Статичка *IP* адреса рачунара треба да садржи прва три броја *IP* адресе ПЛК-а, док посљедњи број мора бити различит од посљедњег броја *IP* адресе ПЛК-а (у овом случају 4). За претходни примјер статичка *IP* адреса може бити 84.31.55.100 те је унос исте приказан на Сл. 92.



Слика 92 – Уношење статичке IP адресе рачунара

Након уношења статичке *IP* адресе рачунара, те постављања параметара конекције као на Сл. 93, врши се тестирање конекције опцијом *Test Connection.*

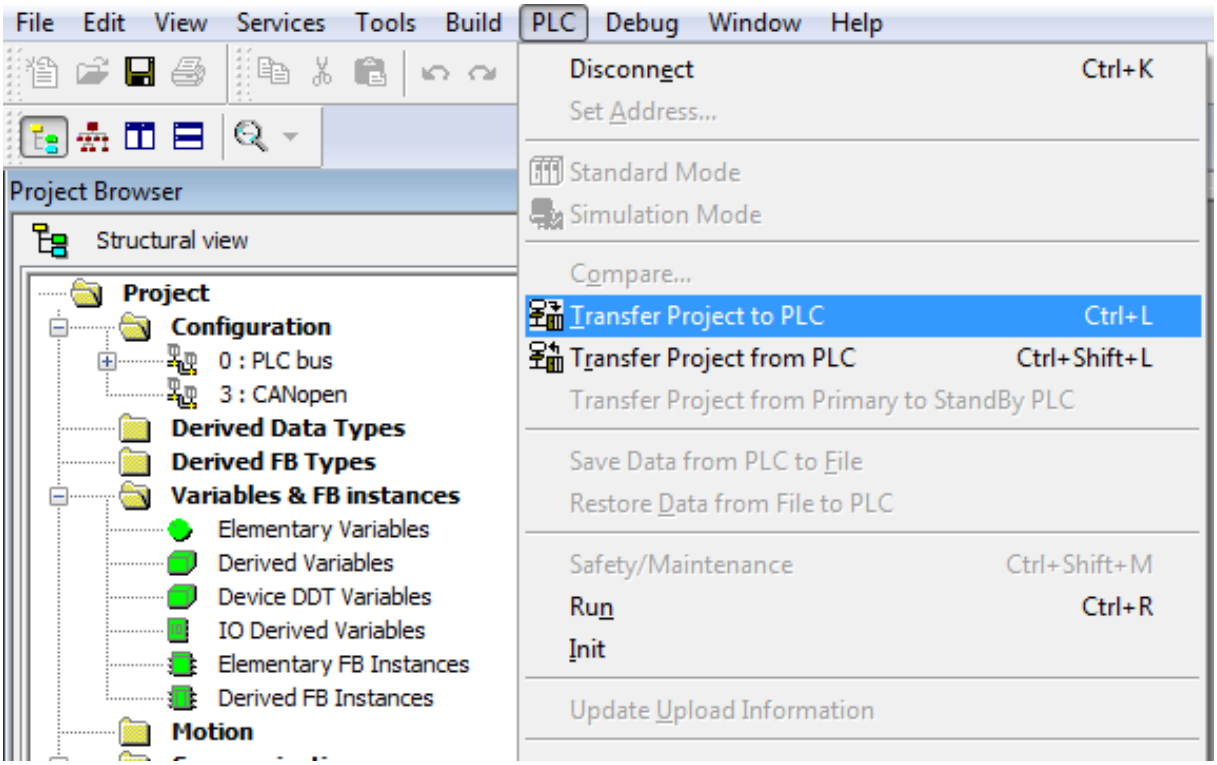


Слика 93 – Параметри TCP IP конекције

Након одабира опције *Test Connection* отворићe се прозор приказан на Сл. 91. Након успјешног тестирања комуникације опцијом *Test Connection,* потребно је повезати ПЛК са рачунаром коришћењем опције *Connect* која се налази на картици *PLC* (Сл. 88).

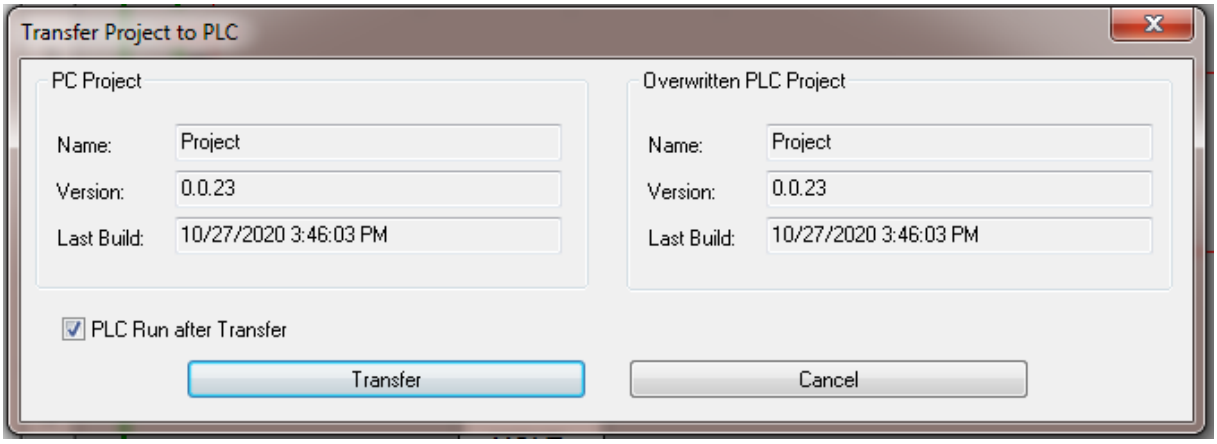
3.8. Отпремање програма на ПЛК

Након написаног програма и избора типа комуникације која се успоставља са ПЛК-ом, потребно је извршити отпремање програма на ПЛК. Трансфер програма се врши бирањем опције *Transfer project to PLC*, на картици *PLC* (Сл. 94).



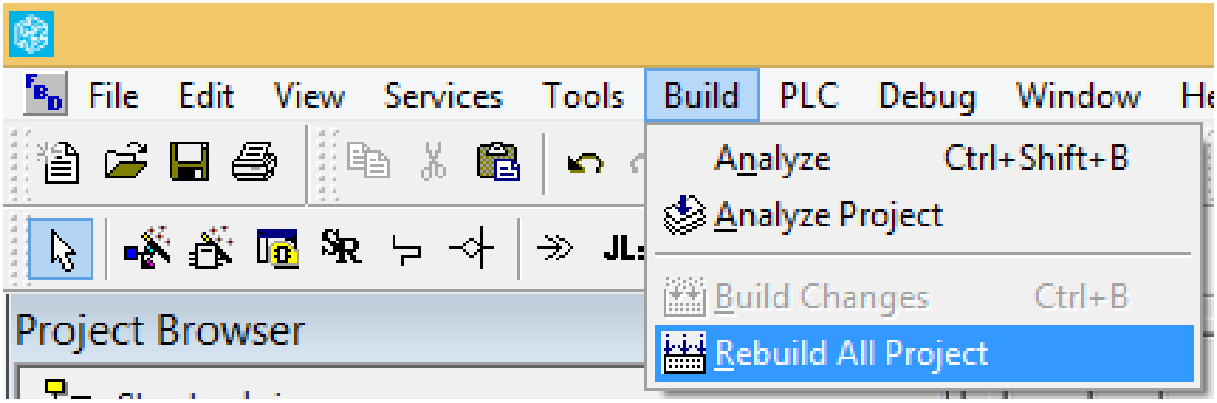
Слика 94 – Отпремање програма на ПЛК

Одабиром опције *Transfer Project to PLC*, отвара се нови прозор на којем треба означити опцију *Run PLC after transfer* (што омогућава да ПЛК почне са извршавањем програма одмах по завршетку отпремања), те одабрати опцију *Transfer* (Сл. 95).



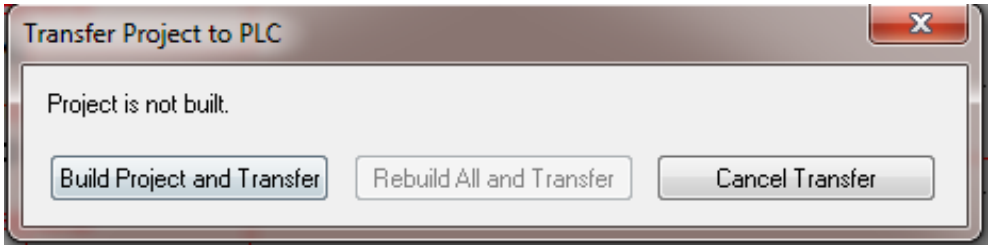
Слика 95 – Отпремање програма на ПЛК са опцијом покретања ПЛК-а након извршеног отпремања

Уколико претходно није извршена изградња програма опцијом *Build* (Сл. 96), потребно је изабрати опцију *Rebuild all and transfer* или *Build Project and Transfer* (Сл. 97).



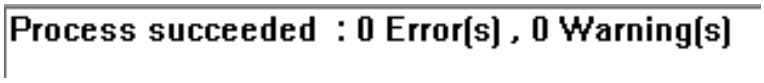
Слика 96 – Изградња свих дијелова пројекта

Након одабира опције *Rebuild all and Transfer* или *Build Project and Transfer*, написани програм ће бити отпремљен на ПЛК. Уколико опција *Run PLC after transfer* није претходно означена, то може накнадно да изврши опцијом *Run* која се налази на траци алата.



Слика 97 – Одабир опције изградње програма током отпремања програма на ПЛК

Приликом изградње програма, постоји могућност провјере евентуалних грешака или упозорења која могу да се односе на одређене дијелове програма. Уколико нема грешака ни упозорења, на статусној линији биће приказан текст као на Сл. 98.



Слика 98 – Извјештај софтвера Unity Pro XL након успјешне изградње програма

4. УПРАВЉАЊЕ ФРЕКВЕНЦИЈСКИМ РЕГУЛАТОРОМ

У другој глави практикума описан је принцип рада, као и улога фреквенцијског регулатора. Како управљати асинхроним мотором посредством фреквенцијског регулатора омогућено је на неколико начина који су описани у наредним поглављима.

Без обзира који начин управљања се користи, претходно је потребно унијети параметре мотора којим се управља у фреквенцијски регулатор. При првом укључивању фреквенцијског регулатора, на дисплеју се исписује *rdY.* На фреквенцијском регулатору налази се тастер *ESC* који се користи за излазак из појединих ставки у менију фреквенцијског регулатора. Поред тастера *ESC*, на фреквенцијском регулатору налази се зелени точкић са тастером (Сл. 99). Окретањем точкића врши кретање кроз мени по вертикали и селектовање одређених опција у менију, а притиском на точкић врши се избор селектоване ставке менија.

Слика 99 – Фреквенцијски регулатор

Унос параметара мотора врши се зеленим точкићем са тастером, избором сљедеће опције *ConF -> FuLL ->Sin.* Одабиром опције *Sin*, уносе се сљедећи параметри:

- *bFr* (стандардна фреквенција (може да буде 50 Hz или 60 Hz) мотора која директно утиче на поједине параметре мотора (на максималну фреквенцију (*HSP*), номинални напон мотора (*uNs*), номиналну фреквенцију мотора (*FrS*)), уноси се у херцима (50 Hz)),
- *nPr* (номинална снага мотора (0.18 kW)),
- *uNs* (номинални напон мотора (230 V)),
- *nCr* (номинална струја мотора (0.95 A)),
- *FrS* (номинална фреквенција мотора (50 Hz)),
- *nSp* (номинална брзина мотора (2710 o/min)),
- *ACC* (вријеме убрзавања мотора (3 s)),
- *dEC* (вријеме успоравања мотора (3 s)),
- *LSP* (фреквенција мотора при задавању минималне референце (0 Hz)),
- *HSP* (фреквенција мотора при задавању максималне референце (50 Hz)).

Сви параметри мотора очитавају се са називне плочице асинхроног мотора.

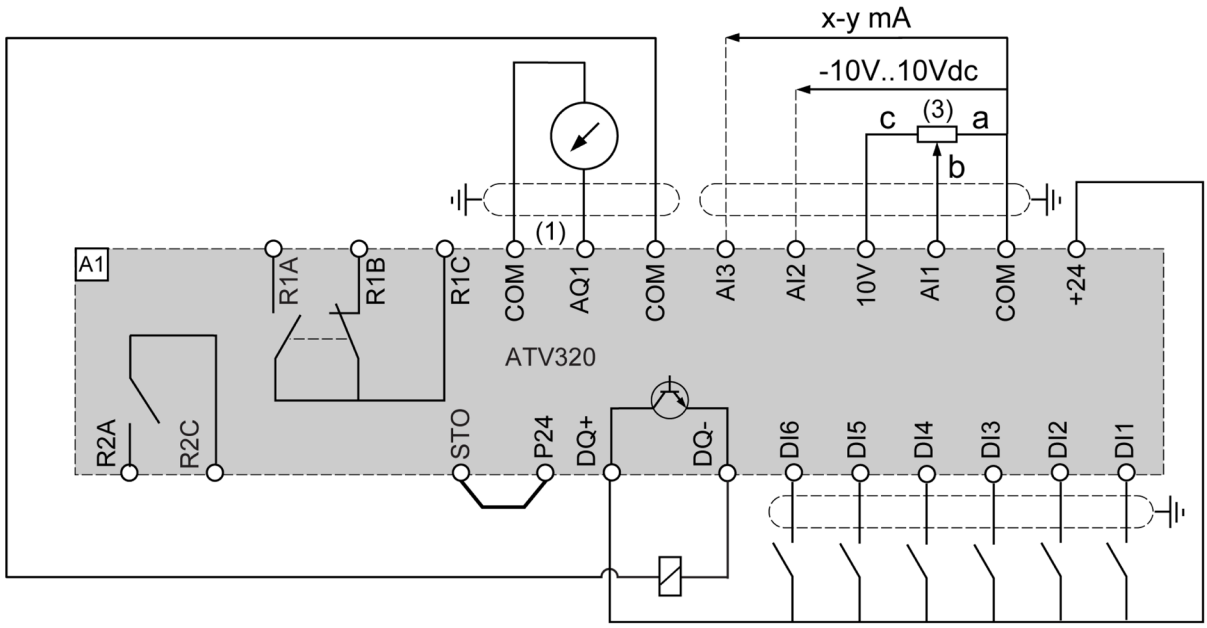
4.1. Управљање задавањем референце директно на фреквенцијском регулатору

Један од начина управљања асинхроним мотором је задавањем референце броја обртаја директно на фреквенцијском регулатору. Потребно је у менију изабрати опцију која омогућава задавање референце директно са фреквенцијског регулатора коришћењем зеленог точкића са тастером на сљедећи начин.

У менију одабрати сљедеће параметре *ConF- > FuLL -> CtL -> CHCF -> Sep* и вратити се на почетни мени користећи *ESC* тастер. Поред подешавања претходне опције гдје је подешен профил *CHCF* са *Sin* (команда и референца нису раздвојене) на *SEP* (команда и референца су раздвојене), потребно је поставити да је референца виртуелни аналогни улаз, односно *A1u1.*

Подешавање виртуелне референце врши се на сљедећи начин *ConF -> FuLL -> CtL -> Fr1 -> A1u1.* Избором опције за задавање виртуелне референце, брзина мотора се може директно задавати путем зеленог точкића са тастером.

Након задавања референце, фреквенцијском регулатору је потребно задати команду за старт, односно на прикључак дигиталног улаза *DI1* потребно је довести једносмјерни напон од 24 V (Сл. 100), логичка јединица.



Слика 100 – Шема повезивања улаза фреквенцијског регулатора

Логичка јединица на *DI1* фреквенцијског регулатора команда је за обртање вратила мотора у једном смјеру, а логичка јединица на *DI2* је команда за обртање вратила мотора у другом смјеру.

4.2. Управљање задавањем референце фреквенцијском регулатору помоћу екстерног потенциометра

Број обртаја мотора, односно задавање фреквенције, може се вршити путем екстерног потенциометра.

Прво је потребно повезати потенциометар на одговарајуће прикључке фреквенцијског регулатора. Један крај потенциометра повезује се на прикључак 10 V, други крај на 0 V, док се клизач потенциометра повезује на аналогни улаз, прикључак *AI1* (Сл. 100).

Након правилног повезивања екстерног потенциометра са фреквенцијским регулатором, потребно је подесити параметре регулатора за ту врсту управљања. Подешавање се врши избором сљедећих команди: *ConF -> FuLL -> CtL -> Fr1 -> A11.* Одабиром опције *A11* изабрано је екстерно управљање брзином мотора путем аналогног сигнала. Да би се мотор покренуо, потребно је довести 24 V једносмјерног напона на *DI1,* уколико се жели обртање вратила мотора на једну страну, односно довођењем 24 V једносмјерног напона на *DI2* за окретање вратила мотора у супротну страну (Сл. 100).

4.3. Управљање фреквенцијским регулатором путем дигиталних и аналогних улаза

Управљање бројем обртаја вратила асинхроног мотора може се вршити путем дигиталних и аналогних улаза фреквенцијског регулатора. Приликом описивања фреквенцијског регулатора, објашњено је да је на фреквенцијски регулатор могуће довести седам дигиталних улаза као и три аналогна улаза.

4.3.1. Управљање фреквенцијским регулатором путем дигиталних улаза

Дигитални улази *DI1* и *DI2* предвиђени су покретање мотора. Фреквенцијски регулатор дозвољава довођење четири дигитална улазна сигнала за које се могу дефинисати жељене брзине на фреквенцијском регулатору. Такође, могуће је дефинисати брзине уколико је активно више дигиталних улаза истовремено. Како би било омогућено управљање путем дигиталних улаза, потребно је конфигурисати фреквенцијски регулатор за ту намјену. Подешавање на фреквенцијском регулатору је сљедеће *ConF -> FuLL ->FUn ->P55 -> P52* (*P52, P54, P58, P516*).

У конфигурацији параметара *P52, P54, P58, P516*, бирају се конкретни дигитални улази фреквенцијског регулатора. Дигитални улази означени су са *LI<i>*, гдје *<i>* представља број који може бити 3,4,5 и 6 (примјер *LI3,* односи се на дигитални улаз *DI3* (Сл. 100)). Након уношења свих жељених дигиталних улаза фреквенцијског регулатора, може се приступити задавању предефинисаних брзина.

Брзине се дефинишу на сљедећи начин: *ConF -> FuLL ->FUn ->P55 -> SP2* (*SP2, SP3, SP4...*). Брзина се задаје подешавањем фреквенције напона напајања у херцима. Максималан број предефинисаних брзина зависи од броја дигиталних улаза. За максималан број од четири дигитална улаза, могуће је дефинисати 15 различитих предефинисаних брзина (Сл. 101).

16 брзина LI (PS16) 4 улаза	**8 брзина LI (PS8)** 3 улаза	**4 брзине LI (PS4)** 2 улаза	**2 LI (PS2)** 1 улаз	**Референца брзине**
0	0	0	0	Референца
0	0	0	1	SP2
0	0	1	0	SP3
0	0	1	1	SP4
0	1	0	0	SP5
0	1	0	1	SP6
0	1	1	0	SP7
0	1	1	1	SP8
1	0	0	0	SP9
1	0	0	1	SP10
1	0	1	0	SP11
1	0	1	1	SP12
1	1	0	0	SP13
1	1	0	1	SP14
1	1	1	0	SP15
1	1	1	1	SP16

Слика 101 – Број могућих предефинисаних брзина

За коришћење само једног дигиталног улаза фреквенцијског регулатора, могуће је дефинисати само једну брзину, односно *SP2*. Коришћењем два дигитална улаза (на примјер *DI3* и *DI4*) могуће је дефинисати три предефинисане брзине, односно *SP2*, *SP3* и *SP4* (представља предефинисану брзину уколико су активна оба дигитална улаза истовремено). Исти принцип вриједи за довођење три и четири дигитална улаза, што се види са Сл. 101.

Након дефинисања жељених брзина и довођењем логичких јединица на дигиталне улазе фреквенцијског регулатора, извршено је управљање асинхроним мотором путем дигиталних улаза. Важно је напоменути да је за покретање мотора потребно довести логичку јединицу и на дигиталне улазе који се користе за старт мотора (*DI1* и *DI2*).

4.3.2. Управљање фреквенцијским регулатором путем аналогних улаза

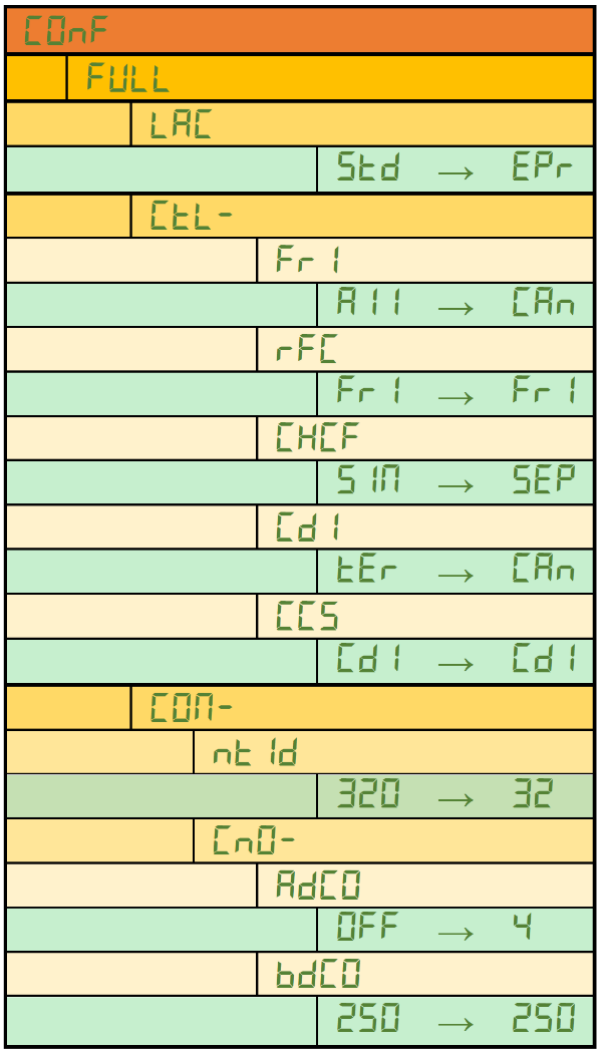
Поред управљања дигиталним улазима фреквенцијског регулатора, асинхроним мотором се може управљати и аналогним улазима фреквенцијског регулатора. Фреквенцијски регулатор има три предвиђена прикључка за управљање екстерним аналогним сигналима. Прикључак *AI1* користи се за повезивање екстерног потенциометра. Прикључак *AI2* користи се за довођење улазног аналогног напонског сигнала чија се вриједност креће у опсегу од -10 до 10 V. Трећи прикључак који се може користити за довођење аналогног сигнала је прикључак *AI3.* Прикључак *AI3* користи се за довођење струјног улазног аналогног сигнала у опсегу од 0 до 20 mA, или у опсегу од 4 до 20 mA. Управљање, односно задавање референце путем екстерног потенциометра је већ описано, те остаје да се опише управљање асинхроним мотором путем аналогних улаза фреквенцијског регулатора *AI2* ,односно *AI3.*

Уколико се жели управљати фреквенцијским регулатором путем аналогног напонског сигнала, потребно је конфигурисати фреквенцијски регулатор на сљедећи начин *ConF -> FuLL -> CtL -> Fr1 -> AI2. CHCF* профил треба бити подешен на *SEP* како би била раздвојена референца и команда, а подешавање *CHCF* профила врши се на сљедећи начин *ConF -> FuLL -> CtL -> CHCF -> SEP*. Довођењем аналогног напонског сигнала, као и дигиталног улаза *DI1* или *DI2*, омогућено је управљање фреквенцијским регулатором путем аналогног напонског сигнала.

Уколико се фреквенцијским регулатором жели управљати путем аналогног струјног сигнала, потребно је конфигурисати фреквенцијски регулатор на сљедећи начин *ConF -> FuLL -> CtL -> Fr1 -> AI3.* Профил *CHCF* треба бити подешен на *SEP,* као што је претходно описано. Подешавање минималне и максималне вриједности улазне струје врши се параметрима *CrL3* и *CrH3.* Параметар *CrL3* односи се на минималну вриједност струје, док се параметар *CrH3* односи на максималну вриједност улазне струје. У зависности од изабраног струјног опсега, потребно је извршити подешавање поменутих параметара. Параметри се подешавају на сљедећи начин *ConF -> FuLL -> I_O->CrL3* (*CrH3*). Довођењем аналогног струјног сигнала, као и дигиталног улаза *DI1* или *DI2*, врши се управљање фреквенцијским регулатором путем аналогног струјног сигнала.

4.4. Управљање фреквенцијским регулатором помоћу *CANopen* комуникације

За повезивање са другим уређајима фреквенцијски регулатор посједује два комуникациона протокола, *Modbus* и *CANopen* протокол. На фреквенцијском регулатору могу се подесити параметри за *CANopen* комуникацију (Сл. 102).



Слика 102 – Подешавање параметара фреквентног регулатора за CANopen комуникацију

4.5. Повезивање програмабилног логичког контролера и фреквенцијског регулатора

На радној станици ПЛК је повезан са фреквенцијским регулатором на два начина, директно и путем *CANopen* протокола.

4.5.1. Директно повезивање

Директно повезивање ПЛК-а и фреквенцијског регулатора подразумијева повезивање дигиталних и аналогних излаза програмабилног логичког контролера са дигиталним и аналогним улазима фреквенцијског регулатора.

Два дигитална излаза са ПЛК-а повезана су на улазе *DI1* и *DI2* фреквенцијског регулатора.

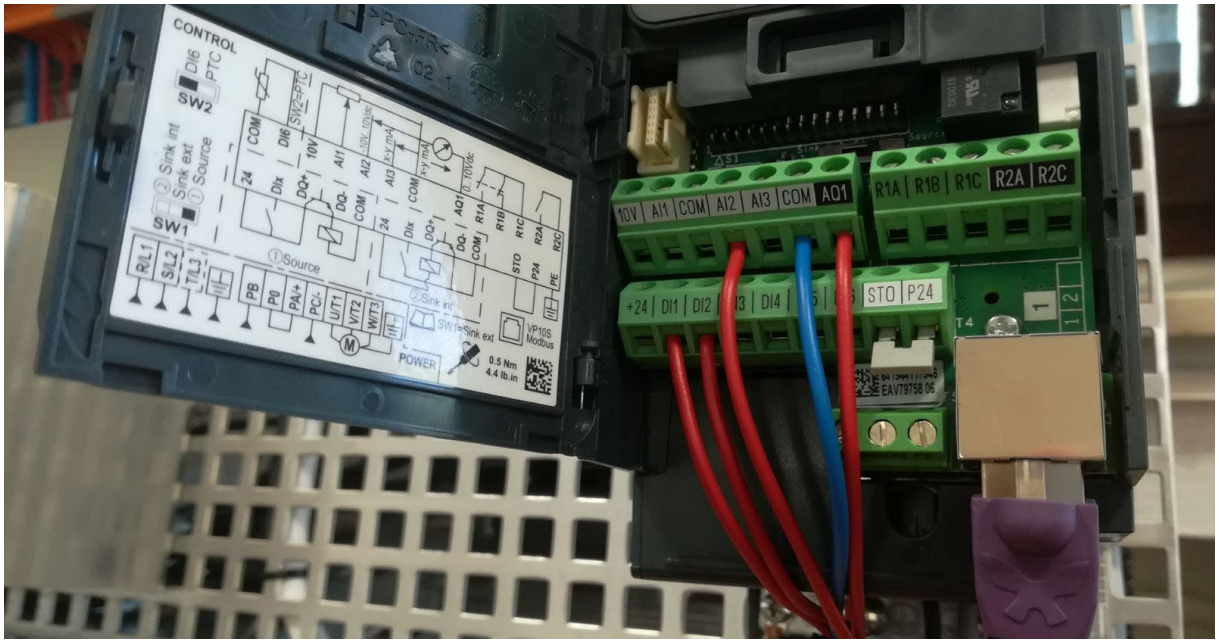
На аналогни улаз *AI2* фреквенцијског регулатора повезан је један аналогни излаз са програмабилног логичког контролера.

Аналогни излаз *AQ1* фреквенцијског регулатора повезан је на аналогни улаз ПЛК. У подешавањима регулатора могу се изабрати различити облици излазног сигнала: струја, напон, фреквенција, обртни момент и други. Ако се изабере фреквенција напона напајања мотора као информација, аналогни излаз има вриједност 5 V при фреквенцији од 50 Hz.

У Таб. 5, приказани су дигитални и аналогни интерфејси ПЛК-а искоришћени за директно повезивање са фреквенцијским регулатором. На Сл. 103 приказани су спојеви дигиталних и аналогних улаза и излаза фреквенцијског регулатора.

Табела 5 – Дигитални и аналогни излази искоришћени за директно повезивање ПЛК-а и фреквенцијског регулатора

Излази/улази са ПЛК-а		Улази/излази фреквенцијског регулатора	
Модул	Адреса	Прикључак	Тип
BMXDDO1602	%Q0.2.0	DI1	Дигитални улаз
BMXDDO1602	%Q0.2.1	DI2	Дигитални улаз
BMXAMM600	%QW0.3.4	AI2	Аналогни улаз
BMXAMM600	%IW0.3.0	AQ1	Аналогни излаз

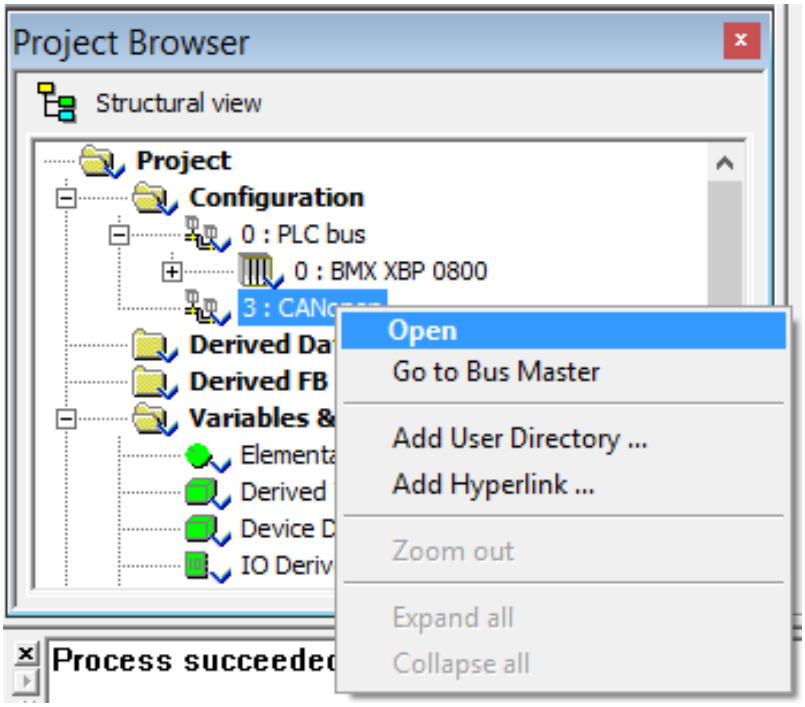


Слика 103 – Директно повезивање ПЛК-а и фреквенцијског регулатора

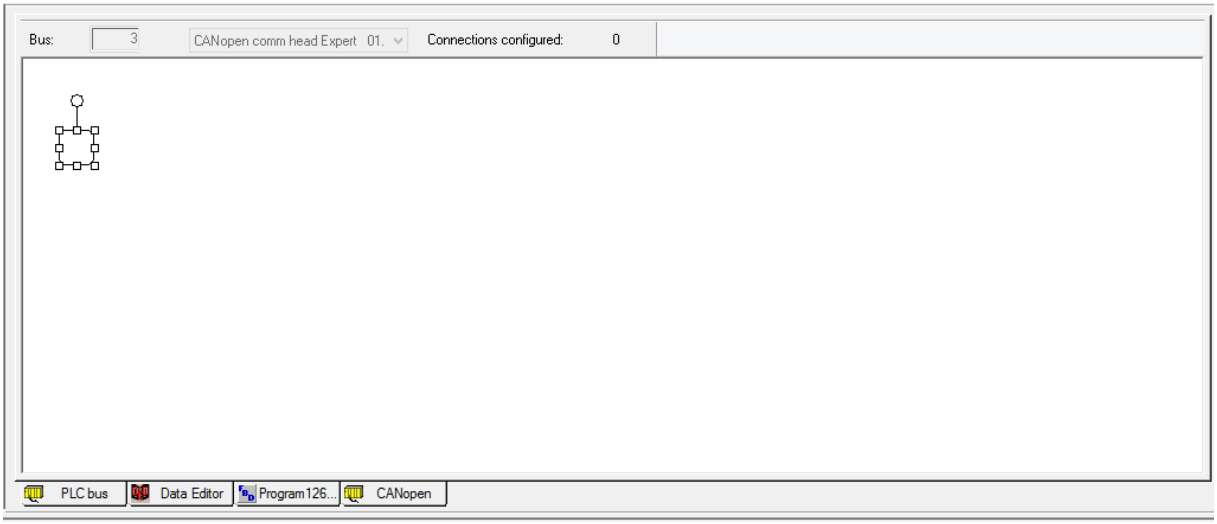
4.5.2. Повезивање *CANopen* протоколом

На радној страници фреквенцијски регулатор је поред директног повезивања, повезан и преко *CANopen* протокола са програмабилним логичким контролером. Након подешених параметара *CANopen* комуникације директно на фреквенцијском регулатору (Сл. 102), потребно је и софтверски извршити повезивање фреквенцијског регулатора са програмабилним логичким контролером. Софтверско повезивање ПЛК-а и фреквенцијског регулатора почиње отварањем картице *CANopen* на падајућем менију у *Project Browseru* софтверске платформе *Unity Pro XL* (Сл. 104).

Након отварања *CANopen* опције у менију са Сл. 104 отвара се прозор приказан на Сл 105.

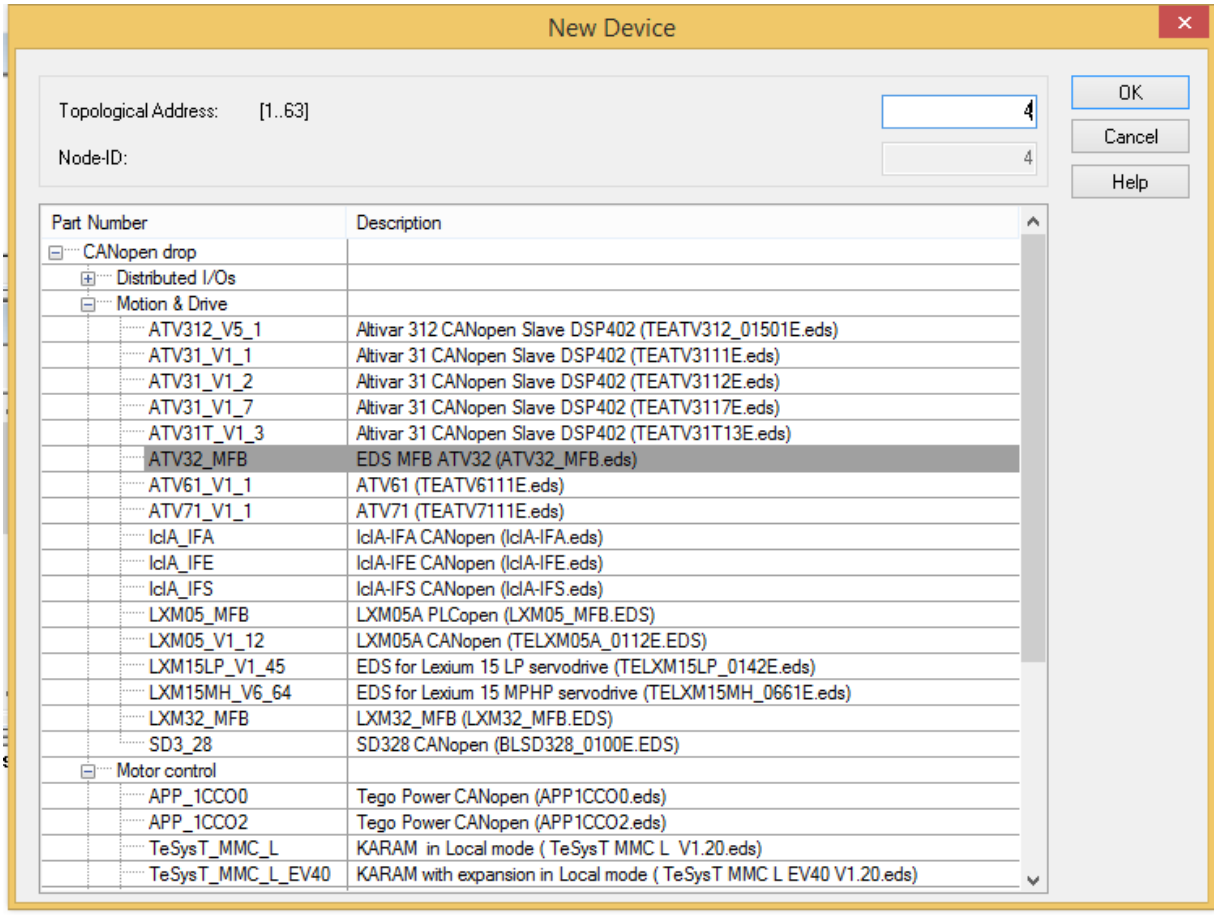


Слика 104 – Подешавање CANopen комуникације у софтверу Unity Pro XL



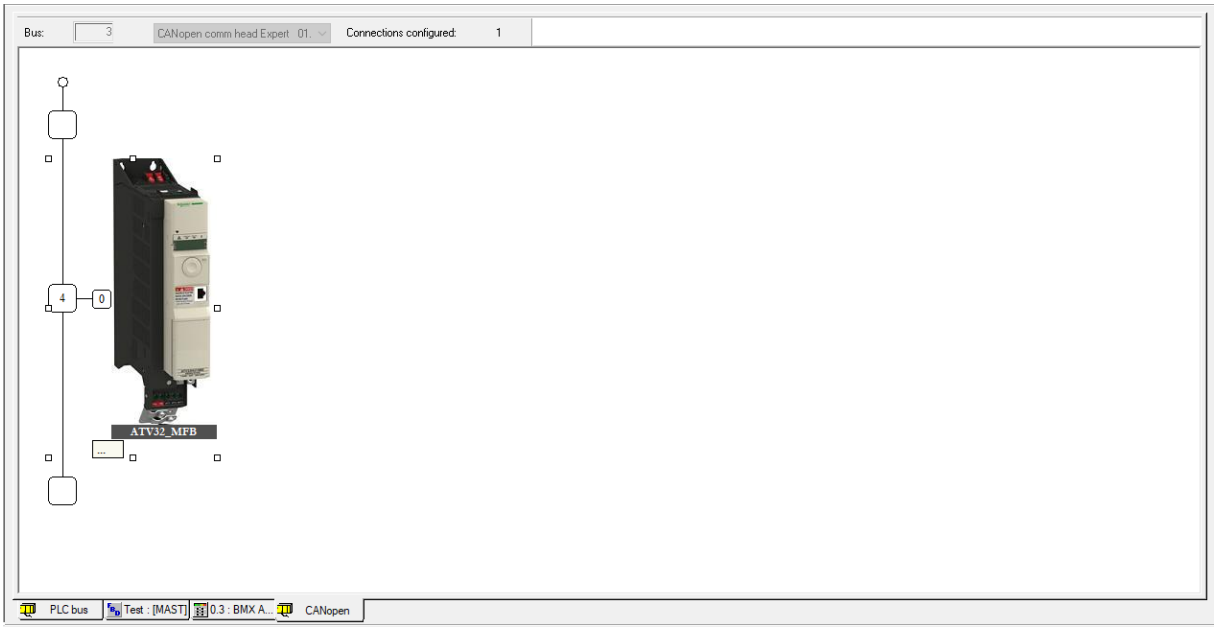
Слика 105 – Прозор за подешавање уређаја који се повезују са ПЛК-ом путем CANopen комуникације

Двоструким кликом миша на бијели квадрат приказан на Сл. 105, отвара се нови прозор који нуди одабир уређаја који су повезани са програмабилним логичким контролером путем *CANopen* комуникације. Уређај који се бира је *ATV32_MFB* који се налази на картици *Motion&Drive* (Сл. 106). Као адреса регулатора уноси се број од 0 до 63, јер је то број могућих слејв уређаја који подржава ПЛК-а, те се унос потврђује опцијом *OK*.

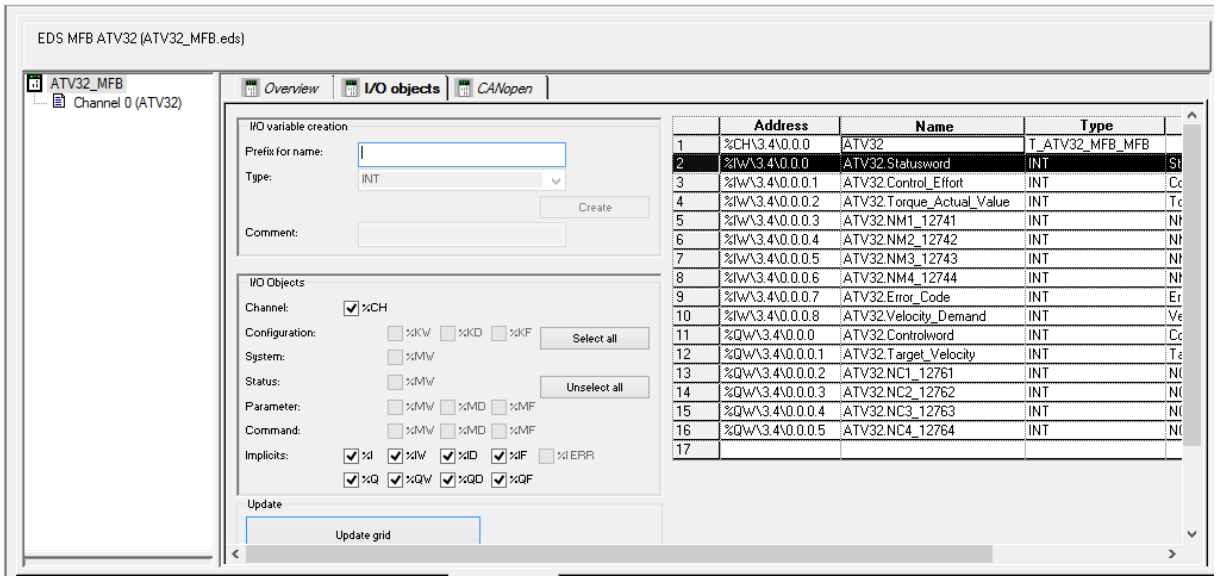


Слика 106 – Избор фреквенцијског регулатора са којим ће ПЛК да оствари комуникацију

Након одабира фреквенцијског регулатора, потребно је учитати све потребне промјенљиве везане за фреквенцијски регулатор. Да би се промјенљиве учитале, потребно је двоструким кликом миша на постављени фреквенцијски регулатор (Сл. 107) отворити прозор приказан на Сл. 108.



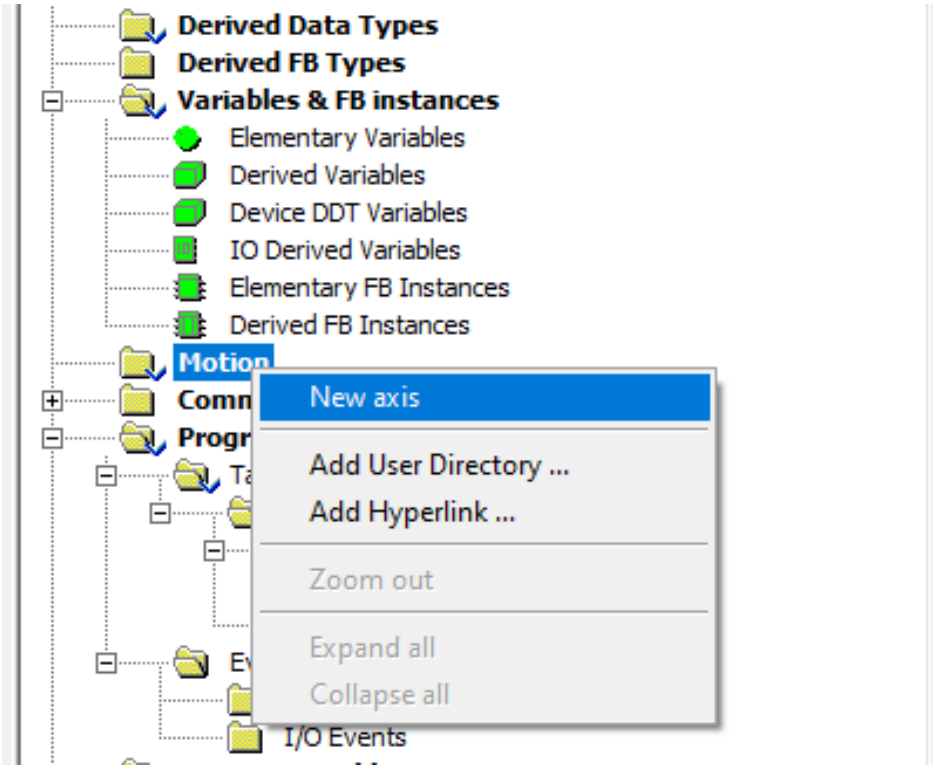
Слика 107 – Изабрани фреквенцијски регулатор



Слика 108 – Подешавање промјенљивих фреквенцијског регулатора

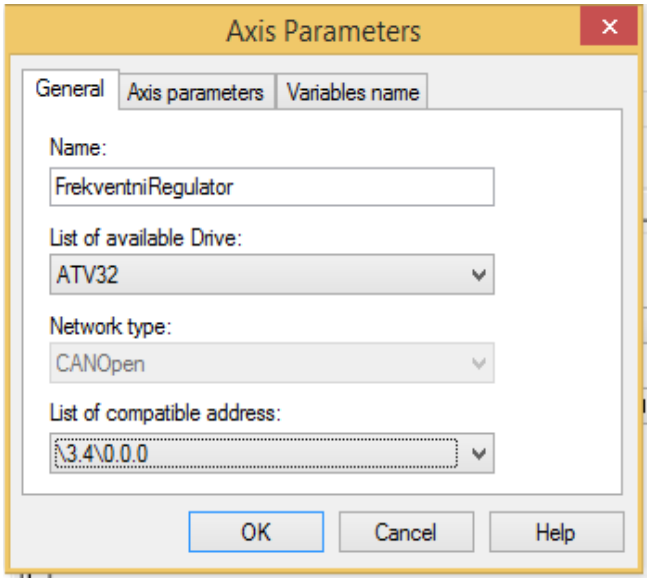
Подешавање промјенљивих фреквенцијског регулатора врши се на картици *I/O objects*, након што се лијевим кликом миша кликне на *ATV32_MFB* на лијевој страни прозора са Сл. 108. Потребно је одабрати опцију *Select all* како би се означили сви улазно/излазни објекти, а промјенљиве учитати опцијом *Update grid.* Након учитавања списка промјенљивих, на мјесту прве промјенљиве чија је адреса *%CH/3.4/0.0.0* потребно је унијети име *ATV32.* Притиском тастера *Enter* учитавају се додатне промјенљиве које су потребне за управљање фреквенцијским регулатором, односно асинхроним мотором.

Након конфигурације комуникације између ПЛК-а и фреквенцијског регулатора, потребно је направити у софтверу осу мотора. Оса мотора (енгл. *Axis*) креира се у *Project Browser*-у на картици *Motion.* Десним кликом миша на картицу *Motion* бира се опција *New Axis* Сл. 109.



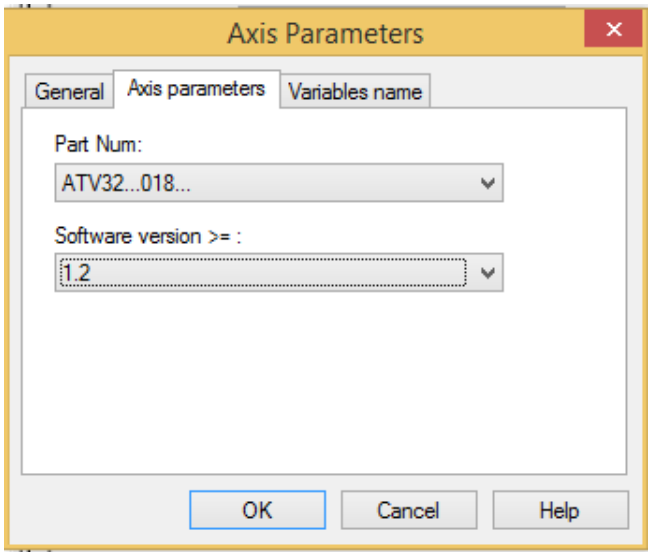
Слика 109 – Креирање нове осе мотора

Након одабира опције *New Axis* потребно је поставити параметре осе. Прозор који се отвара има три картице. На првој картици *General* уноси се име фреквенцијског регулатора као и његова адреса, као што је приказано на Сл. 110.



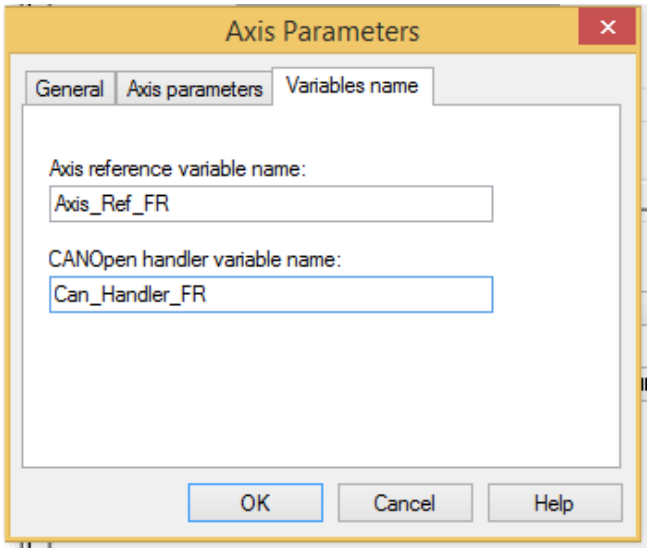
Слика 110 – Задавање имена и одабир адресе фреквенцијског регулатора

У другој картици под називом *Axis parametres* уносе се подаци као на Сл. 111. Врши се избор конкретног ATV32 фреквенцијског регулатора и верзије његовог софтвера (фирмвера).



Слика 111 – Одабир конкретног ATV32 фреквенцијског регулатора и верзије софтвера

У трећој картици за одређивање параметара осе, под називом *Variable name* уносе се називи промјенљивих као на Сл. 112.



Слика 112 – Унос имена промјенљивих осе

Описаним поступком завршена је конфигурација комуникације ПЛК-а и фреквенцијског регулатора. Сада се може приступити изради управљачког програма.

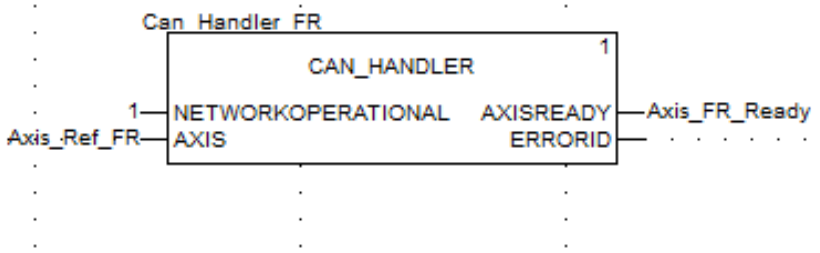
Поред описаних функцијских блокова из поглавља 3.6 под називом *„Опис блокова у програмском језику FBD“* постоји група функцијских блокова који се користи за управљање мотором, као и за остваривање *CANopen* комуникације. У функцијске блокове који се користе за *CANopen* комуникацију као и за управљање асинхроним мотором убрајају се:

- CAN_HANDLER,
- MC_POWER,
- MC_STOP,
- MC_MOVEVELOCITY,
- MC_TORQUECONTROL,
- MC_MOVEABSOLUTE,
- MC_READACTUALVELOCITY,
- MC_READACTUALTORQUE,
- MC_RESET.

Функцијски блок *CAN_HANDLER* користи се за провјеру комуникације између уређаја повезаних комуникацијом и конфигурисаних у софтверу *Unity Pro XL.* На улазној страни функцијски блок посједује два улаза.

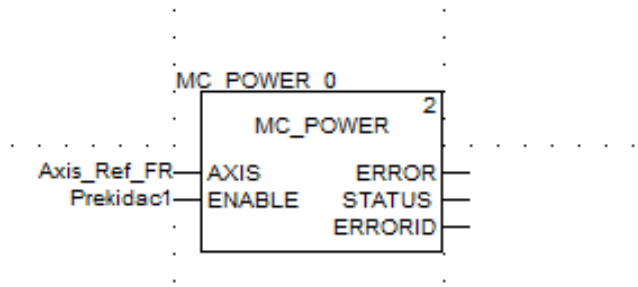
На првом улазу (*NETWORKOPERATIONAL*) уписује се логичка јединица, односно вриједност која представља исправност комуникације између фреквенцијског регулатора и ПЛК-а. На улаз *AXIS* уноси се назив креиране осе.

На излазу из функцијског блока *CAN_HANDLER* налази се излаз *AXISREADY.* На излаз се поставља промјенљива типа *BOOL* и она има вриједност логичке јединице уколико је *CANopen* комуникација успостављена. Излаз *ERRORID* је промјенљива типа *INT* и представља код грешке уколико до ње дође при успостављању *CANopen* комуникације. Значење кода грешке се може наћи у опцији *Help* у софтверу *Unity Pro XL.* Примјер коришћења *CAN_HANDLER* блока приказан је на Сл. 113.



Слика 113 – Функцијски блок CAN_HANDLER

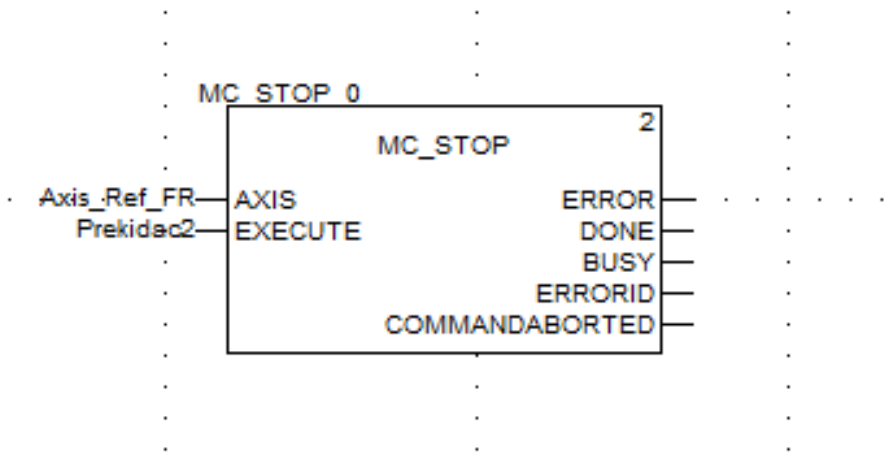
Функцијски блок *MC_POWER* користи се за укључивање фреквенцијског регулатора након успостављања *CANopen* комуникације. На Сл. 114 приказан је функцијски блок *MC_POWER.*



Слика 114 – Функцијски блок MC_POWER

На улазу у функцијски блок налази се улаз *AXIS* на који се доводи назив претходно креиране осе. Улаз *ENABLE* је типа *BOOL* и користи се за активирање блока, уколико се на улаз доведе логичка јединица. У случају са Сл. 114, на улаз *ENABLE* доведен је дигитални улаз, односно први прекидач. На излазу *ERROR* могу бити два стања, логичка нула или логичка јединица. Уколико је на излазу логичка јединица, дошло је до грешке у функцијском блоку, односно за тачно објашњење грешке, чита се број са излаза *ERRORID* који је типа *INT*. Излаз *STATUS* треба бити логичка јединица уколико функцијски блок нема грешку.

Функцијски блок *MC_STOP* користи се за заустављање асинхроног мотора путем фреквенцијског регулатора. Као и блок *MC_POWER* и блок *MC_STOP* има два улаза. Један се користи за повезивање са осом, док се други користи за активирање блока. Функцијски блок *MC_STOP* приказан је на Сл. 115.

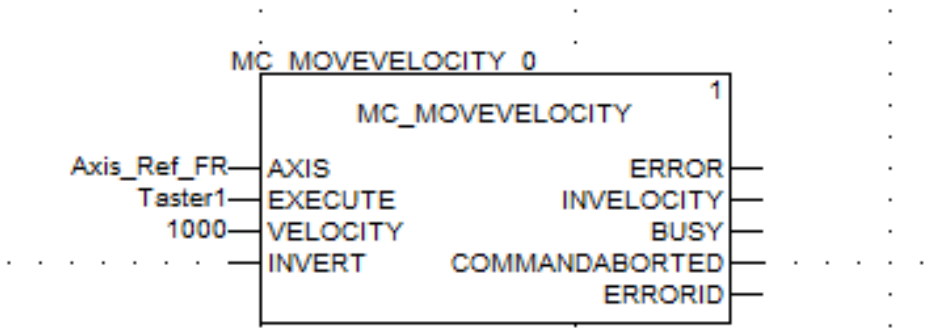


Слика 115 – Функцијски блок MC_STOP

Функцијски блок *MC_STOP* има више излаза. Излази *ERROR* и *ERRORID* претходно су описани.

У тренутку када дође до активирања блока, односно у тренутку када је вриједност обртања осе нула, излаз *DONE* постаје логичка јединица. Излаз *BUSY* је типа *BOOL*, и има вриједност логичке јединице све док се функцијски блок извршава. Излаз *COMMANDABORTED* има вриједност логичке јединице уколико је неки други функцијски блок преузео контролу управљања.

Функцијски блок *MC_MOVEVELOCITY* користи се за регулацију броја обртаја мотора (Сл. 116).



Слика 116 – Функцијски блок MC_MOVEVELOCITY

На улаз *VELOCITY* уноси се брзина у броју обртаја по минути. Улазом *EXECUTE* активира се блок. У случају са Сл. 116 као команда на улазу блока користи се тастер, односно довођењем логичке јединице на улаз.

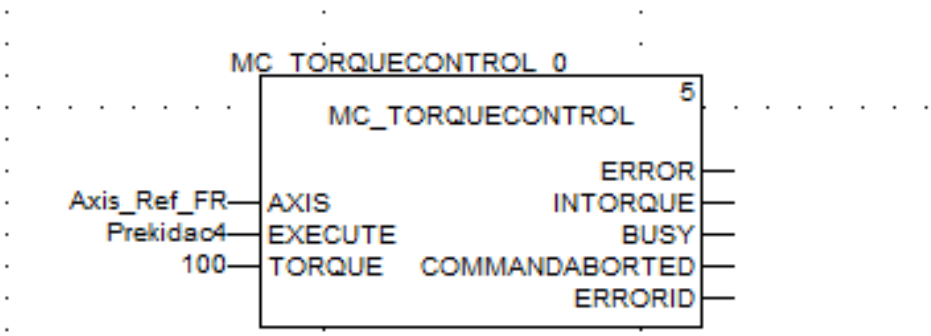
Улаз *INVERT* користи се за мијењање смјера мотора у зависности да ли се на улаз доведе логичка нула или логичка јединица. Када се на улаз *INVERT* доведе логичка јединица, смјер мотора се мијења.

Излаз *INVELOCITY* поставља се на вриједност логичке јединице када брзина обртања осе мотора достигне задату вриједност на улазу *VELOCITY.*

Функцијски блок *MC_TORQUECONTROL* користи се за регулацију обртног момента асинхроног мотора (Сл. 117).

На улаз *TORQUE* уноси се вриједност обртног момента у јединици *Nm*.

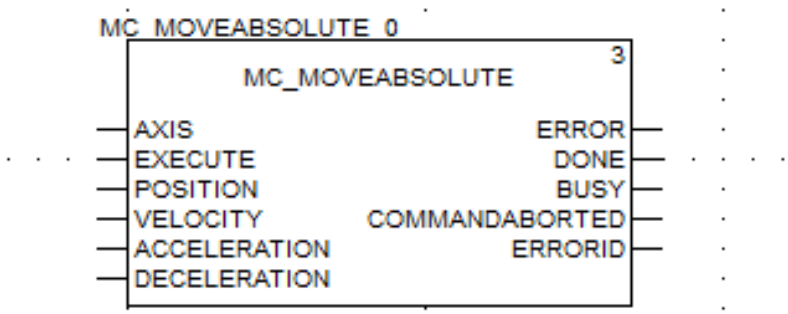
Излаз *INTORQUE* постаје логичка јединица у тренутку када обртни момент достигне задату вриједност. Функцијски блок *MC_TORQUECONTROL* приказан је на Сл. 117.



Слика 117 – Функцијски блок MC_TORQUECONTROL

Функцијски блок *MC_MOVEABSOLUTE* користи се за задавање апсолутне позиције вратила мотора, најчешће сервомотора (Сл. 118).

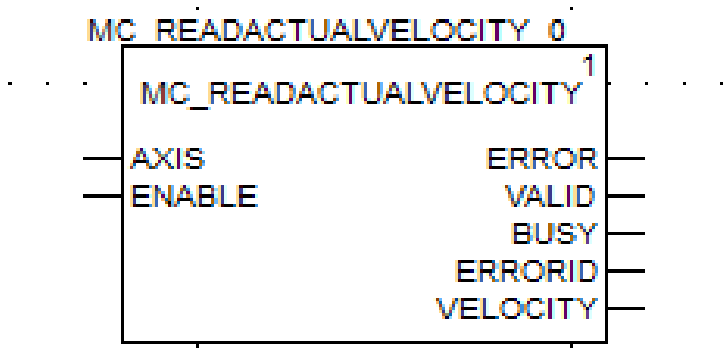
Улаз *POSITION* је типа *DINT*. Поставља апсолутну позицију коју треба да достигне вратило мотора. На улаз *VELOCITY* уноси се брзина вратила док не дође до жељене позиције. Улази *ACCELERATION* и *DECELERATION* користе се за задавање убрзања и успорења приликом кретања до задате позиције. Функцијски блок *MC_MOVEABSOLUTE* приказан је на Сл. 118. Излази из блока описани су кроз претходне блокове.



Слика 118 – Функцијски блок MC_MOVEABSOLUTE

Функцијски блок *MC_READACTUALVELOCITY* обавља функцију мјерења брзине обртања осе (Сл. 119).

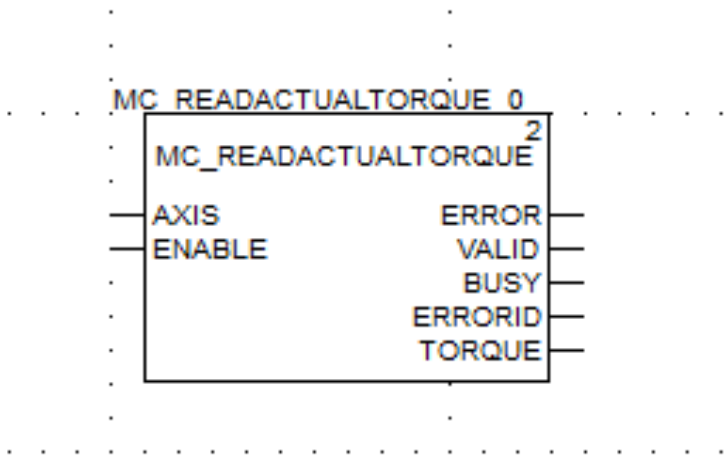
Излаз *VALID* је типа *BOOL* и постаје логичка јединица у тренутку када се блок успјешно изврши. Излаз *VELOCITY* је типа *DINT* и користи се за исписивање очитане брзине вратила. На Сл. 119 приказан је функцијски блок *MC_READACTUALVELOCITY.*



Слика 119 – Функцијски блок MC_READACTUALVELOCITY

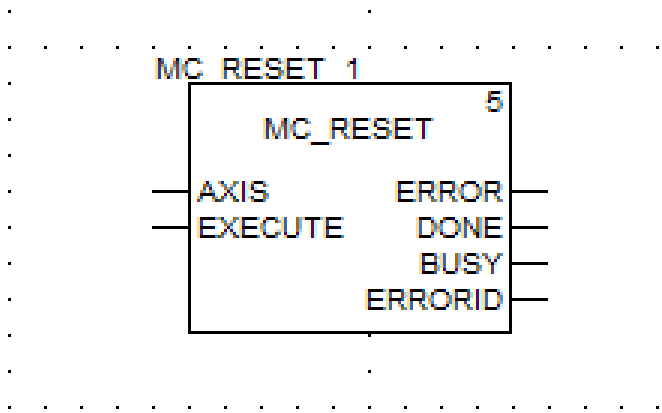
Функцијски блок *MC_READACTUALTORQUE* користи се за очитавање обртног момента мотора (Сл. 120).

Принцип рада, као и сви улази и излази су као и за функцијски блок *MC_READACTUALVELOCITY*. Једина разлика је што се очитава обртни момент мотора на излазу *TORQUE*. Приказ функцијског блока *MC_READACTUALTORQUE* приказан је на Сл. 120.



Слика 120 – Функцијски блок MC_READACTUALTORQUE

Посљедњи блок који припада функцијским блоковима за управљање фреквенцијским регулатором је функцијски блок *MC_RESET* (Сл. 121). Блок има функцију да ресетује садржај блока *ReadAxisError* који садржи дијагностичке информације.



Слика 121 – Функцијски блок MC_RESET

5. ПРИМЈЕРИ КОРИШЋЕЊА РАДНИХ СТАНИЦА

Вјежбе и рад са станицама почињу једноставним уводним задацима кроз које се увјежбава софтверска конфигурација, дефинисање промјенљивих, распоред и садржај развојног окружења, основне наредбе, отклањање грешака, симулација и пренос и тестирање програма на станици.

Како би се показала функционалност свих компоненти које су описане у практикуму, дата су два примјера која укључују улазно/излазне уређаје, фреквенцијски регулатор, асинхрони мотор и различите начине управљања асинхроним мотором.

5.1. Припремни задатак 1

Прекидач укључује лампицу.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

старт/стоп	Прекидач 1	%I0.1.0
лампица	Жута 1	%Q0.2.3

5.2. Припремни задатак 2

Лампица свијетли ако је бар један од тастера притиснут.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

тастер 1	Тастер 1	%I0.1.4
тастер 2	Тастер 2	%I0.1.5
лампица	Жута 1	%Q0.2.3

5.3. Припремни задатак 3

Лампица свијетли ако је прекидач укључен и ако је нужни стоп активиран.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

прекидач	Прекидач 1	%I0.1.0
нужни стоп	Нужни стоп	%I0.1.11
лампица	Жута 1	%Q0.2.3

5.4. Припремни задатак 4

Тастер Старт укључује лампицу, а тастер Стоп ју је искључује (самодржање).

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

старт	Тастер 1	%I0.1.4
стоп	Тастер 2	%I0.1.5
лампица	Жута 1	%Q0.2.3

5.5. Припремни задатак 5

Тастер Старт укључује лампицу, а тастер Стоп ју је искључује.

Задатак реализовати наредбама Сет и Ресет.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

старт	Тастер 1	%I0.1.4
стоп	Тастер 2	%I0.1.5
лампица	Жута 1	%Q0.2.3

5.6. Припремни задатак 6

Стона бушилица треба да се укључује и искључује прекидачем.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табелама.

старт/стоп	Прекидач 1	%I0.1.0
бушилица ради	Жута 1	%Q0.2.3

Додатак 1:

На стоној бушилици уграђен је оптички сензор којим је могуће детектовати да ли је бургија постављена и да ли је читава.

У случају да дође до лома бургије треба да се укључи аларм.

Аларм треба да остане укључен све док сензор не региструје да је у бушилицу постављена исправна бургија.

Ако је аларм активан, бушилица се зауставља и није ју могуће покренути док се бургија не промијени.

Слика 122 – Илустрација детекције лома бургије на машини

оптички сензор	Оптички сензор	%I0.1.8
аларм	Црвена 1	%Q0.2.2
старт/стоп	Прекидач 1	%I0.1.0
бушилица ради	Жута 1	%Q0.2.3

Додатак 2:

Стона бушилица се, умјесто прекидачем, укључује једним тастером, а искључује другим.

оптички сензор	Оптички сензор	%I0.1.8
аларм	Црвена 1	%Q0.2.2
старт	Тастер 1	%I0.1.4
стоп	Тастер 2	%I0.1.5
бушилица ради	Жута 1	%Q0.2.3

5.7. Припремни задатак 7

Лампица се укључује 3s након што је укључен прекидач.

Лампица се искључује када се прекидач искључи.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

прекидач	Прекидач 1	%I0.1.0
лампица	Жута 1	%Q0.2.3

5.8. Припремни задатак 8

Први дио задатка.

Када се тастер притисне 3 пута укључује се жута лампица.

Другим тастером се ресетује бројач.

Други дио задатка.

Ако се настави са притискањем тастера више од 5 пута, укључује се и црвена лампица.

Компоненте које су потребне за реализацију задатка и њихове адресе су дате у табели.

прекидач	Прекидач 1	%I0.1.0
лампица	Жута 1	%Q0.2.3

5.9. Управљање брзином асинхроног мотора помоћу *CANopen* протокола

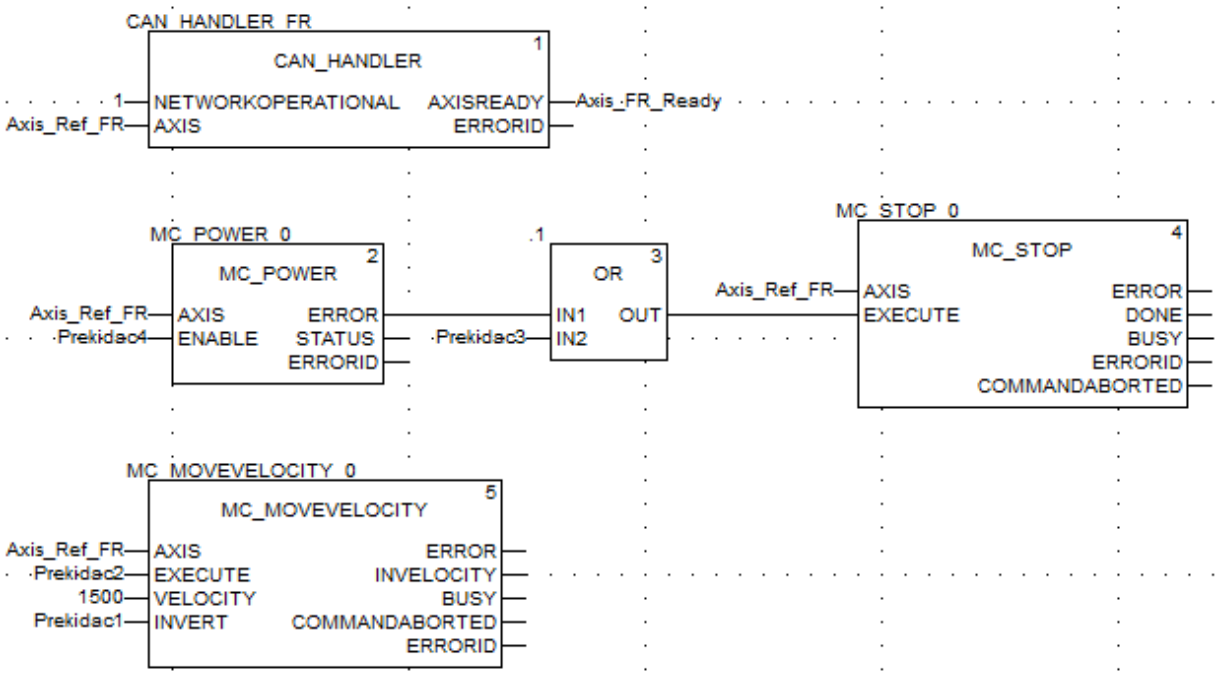
Примјер програма за управљање брзином обртања вратила асинхроног мотора путем *CANopen* протокола приказан је на Сл. 123.

Потребно је иницијализовати *CANopen* комуникацију путем функцијског блока *CAN_HANDLER.* Претходно је креирана оса, чија је промјенљива названа *Axis_Ref_FR.*

Да би се фреквенцијски регулатор припремио за примање команди са ПЛК-а, потребно је поставити функцијски блок *MC_POWER.* Укључивањем четвртог прекидача на контролној табли шаље се логичка јединица на улаз *ENABLE MC_POWER* блока. Фреквенцијски регулатор је спреман и чека наредбу од ПЛК-а за повећањем брзине вратила мотора.

Излаз *ERROR* из блока *MC_POWER_0* повезан је са улазом *OR* функцијског блока, на који је такође повезан улаз *Prekidac3*. Излаз из функцијског блока *OR* повезан је са улазом *EXECUTE MC_STOP_0* блока. Уколико би дошло до било какве грешке на блоку *MC_POWER_0*, излаз *ERROR* постаје логичка јединица која се шаље на *OR* функцијски блок. Излаз са блока *OR* користи се као команда за активирање блока *MC_STOP_0,* који се користи за заустављање мотора.

Коришћењем блока *MC_MOVEVELOCITY* уноси се жељена брзина, односно број обртаја вратила мотора. Укључивањем другог прекидача са контролне табле активира се блок *MC_MOVEVELOCITY* и број обртаја вратила мотора расте на заданих 1500 o/min. Укључивањем првог прекидача мијења се смјер обртања вратила асинхроног мотора.



Слика 123 – Примјер програма управљања асинхроним мотором помоћу CANopen протокола и ПЛК-а

5.10. Мјерење броја обртаја вратила мотора помоћу индуктивног сензора

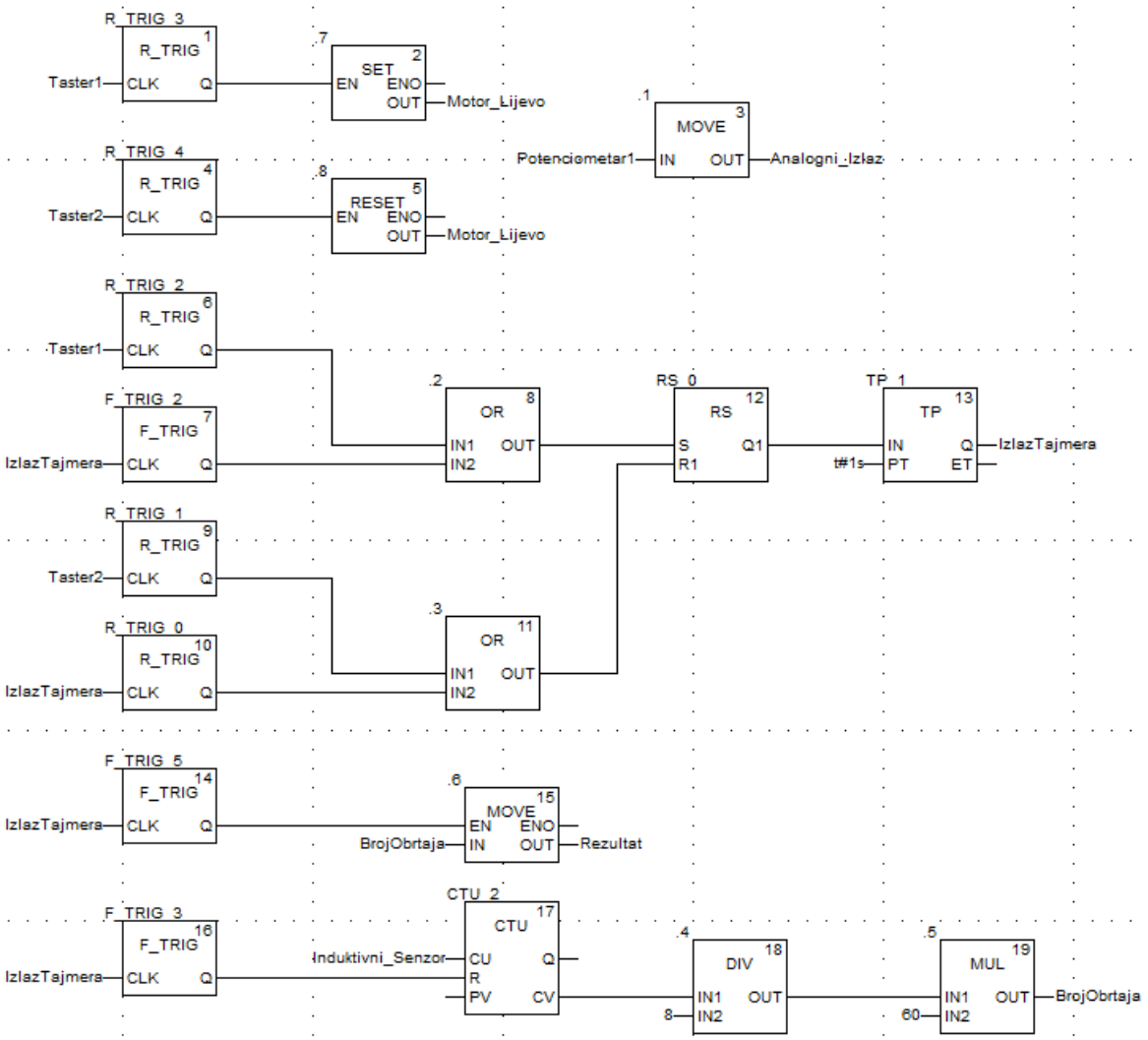
Други примјер односи се на програм који рачуна број обртаја вратила асинхроног мотора путем индуктивног сензора. С обзиром да је индуктивни сензор повезан на модул дигиталних улаза, а не на брзи бројач, потребно одредити до које брзине обртања вратила електромотора је могуће мјерити тачну брзину обртања.

У опису радне станице поменут је ланчаник постављен на вратило електромотора. Ланчаник има осам зуба, што имплицира да ће индуктивни сензор за један обртај вратила мотора генерисати осам импулса.

Са називне плочице електромотора може се уочити да је номиналан број обртаја при фреквенцији од 50 Hz, 2710 обртаја по минути. Након претварања минута у секунде и множења са бројем зуба по обртају вратила електромотора, индуктивни сензор би требао избројати 361 импулс у секунди за фреквенцију од 50 Hz.

Принцип рада програма заснива се на мјерењу броја импулса у току једне секунде. Импулси се затим претварају у број обртаја по минути.

Реализација задатка у програмском језику *FBD* приказана је на Сл. 124.



Слика 124 – Реализација програма за рачунање броја обртаја индуквним сензором

Притиском првог тастера на контролној табли сетује се дигитални излаз *Motor_Lijevo*. Дигитални излаз са ПЛК-а (*Motor_Lijevo*) повезан је директно на *DI1* улаз фреквенцијског регулатора и користи се као старт услов. Притиском на други тастер са контролне табле ресетује се дигитални излаз *Motor_Lijevo.*

Функцијским блоком *MOVE* управља се бројем обртаја вратила електромотора. Првим потенциометром са контролне табле задаје се референца од 0 до 10000 која представља опсег од 0 до 10 V. Излаз из блока *MOVE* представља аналогни излаз са ПЛК-а који је спојен директно на аналогни напонски улаз *AI2* фреквенцијског регулатора.

За прво покретање програма, поред сетовања дигиталног излаза *Motor_Lijevo*, користи се први тастер са контролне табле. Први тастер представља улаз у блок *R_TRIG_2* који региструје промјену стања са логичке нуле на логичку јединицу. Паралелно са функцијским блоком *R_TRIG_2* налази се функцијски блок *F_TRIG_2* на који се доводи излаз са тајмера *TP*. Функцијски блок *F_TRIG* детектује промјену улазног сигнала са логичке јединице на логичку нулу. Излази са блокова *R_TRIG* и *F_TRIG* остају активни у току једног програмског циклуса, а њихови излази се воде на функцијски блок *OR*. Функцијски блок *OR* на излазу даје логичку јединицу када је бар један од улаза логичка јединица.

Излаз из функцијског блока *OR* одводи се на *S* улаз *RS* блока (блок са доминантним ресетом). Након притиска првог тастера, сетује се излаз функцијског блока *RS* и на његовом излазу је логичка јединица. Логичка јединица на излазу *RS* блока користи се као окидачки импулс којим се управља тајмером.

На улазу *PT* тајмера уноси се вријеме (1 секунда). Тајмер генерише импулс на излазу, односно даје логичку јединицу на излазу за вријеме од једне секунде. За непрестано генерисање импулса тајмера, потребно је да тајмер добија окидачки импулс, како би наставио да генерише вријеме од једне секунде.

Како би се сваке секунде мијењала вриједност улаза у тајмер, потребно је вршити ресет и сет *RS* блока. Улаз ресет *RS* блока долази из другог *OR* блока. На улазе другог *OR* блока, доводе се излази са блокова *R_TRIG.* На улаз *R_TRIG_1* блока, доводи се улаз другог тастера на контролној табли, а на улаз другог *R_TRIG_0* блока, доводи се излаз са тајмера.

Другим тастером на контролној табли може се у било којем тренутку ресетовати *RS* блок. Након првог притиска првог тастера са контролне табле, почиње мјерење броја импулса индуктивним сензором. У тренутку када се излаз са тајмера активира, односно пређе из стања логичке нуле у стање логичке јединице, функцијски блок *R_TRIG_0* ресетује вриједност *RS* излаза. Сетовање *RS* блока се може извршити поновним притиском првог тастера или промјеном стања излаза са тајмера са логичке јединице на логичку нулу (*F_TRIG_2* блок).

За бројање импулса посредством индуктивног сензора користи се бројач унапријед, односно *CTU*. На улаз бројача *CU* доводи се индуктивни сензор. Ресет бројача врши се сваке секунде, односно преко блока *F_TRIG_3* (када излаз тајмера промијени стање са логичке јединице на логичку нулу - силазна ивица).

На излазу из бројача користи се прикључак *CV* који броји тренутну вриједност импулса. Број импулса се прво дијели са осам (8 импулса представља један обртај), а затим се множи са 60, како би се добио број обртаја по минути.

Другим функцијским блоком *MOVE* врши се упис промјенљиве *BrojObrtaja* у промјенљиву *Rezultat.* Упис се врши сваке секунде, након детекције силазне ивице излаза тајмера, како би број обртаја по минути био прегледан у току једног мјерења.

Мјерењем брзине формираним програмом у опсегу од нуле до номиналне брзине утврђено је да при фреквенцији већој од 15 Hz долази до неправилног бројања импулса, због дужине скен циклуса програмабилног логичког контролера. Дужина скен циклуса зависи од броја улаза и излаза, величине и захтјевности програма. Индуктивни сензор региструје промјене до 2.5 kHz. Да би се извршило правилно мјерење брзине окретања вратила електромотора, индуктивни сензор је потребно повезати на модул брзог бројача.

6. ЛИТЕРАТУРА

[1] Доступно на: http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/introtoplcs_SUPER.pdf, приступљено [10.10.2020]

[2] Хасан Смајић, Тони Дуспара, „Програмирање и примјена контролера“, Köln, 2020.

[3] Доступно на: https://www.se.com/ww/en/product/BMXXBP0600/rack-m340---6-slots---panel%2C-plate-or-din-rail-mounting/, приступљено [10.10.2020]

[4] Доступно на: https://www.se.com/ww/en/product/BMXCPS2000/power-supply-module-x80---100..240-v-ac---20-w/, приступљено [10.10.2020]

[5] Доступно на: https://gesrepair.com/what-is-an-eeprom/, приступљено [30.10.2020]

[6] Доступно на: https://www.javatpoint.com/ram, приступљено [30.10.2020]

[7] Доступно на: https://www.se.com/ww/en/product/BMXP3420302/processor-module-m340---max-1024-discrete-%2B-256-analog-i-o---canopen/, приступљено [11.10.2020]

[8] Доступно на: https://www.ni.com/en-rs/innovations/white-papers/13/the-basics-of-canopen.html, приступљено [11.10.2020]

[9] https://www.se.com/ww/en/product/BMXDDI1602/discrete-input-module-x80---16-inputs---24-v-dc-positive/, приступљено [11.10.2020]

[10] Доступно на: https://www.se.com/ww/en/product/BMXDDO1602/discrete-output-module-x80---16-outputs---solid-state---24-v-dc-positive/, приступљено [11.10.2020]

[11] Доступно на: https://www.se.com/ww/en/product/BMXAMM0600/mixed-analog-i-o-module-x80---4-inputs---2-outputs/, приступљено [12.10.2020]

[12] Доступно на: https://www.se.com/ww/en/product/BMXEHC0200H/high-speed-counter-module-m340---2-channels/, приступљено [12.10.2020]

[13] Доступно на: https://www.se.com/ww/en/product/ATV320U02M2C/variable-speed-drive%2C-atv320%2C-0.18-kw%2C-200%E2%80%A6240-v%2C-1-phase%2C-compact/, приступљено [12.10.2020]

[14] Доступно на: https://www.se.com/ww/en/product/ZB6AV4/red-pilot-light-head-%C3%B816-for-integral-led/, приступљено [13.10.2020]

[15] Доступно на : https://www.se.com/ww/en/product/ZB6AV3TQ/pilot-light-head-zb6av3-sold-by-100/, приступљено [1.11.2020]

[16] Доступно на: https://www.se.com/ww/en/product/ZB6AV5/yellow-pilot-light-head-%C3%B816-for-integral-led/, приступљено [1.11.2020]

[17] Доступно на: https://www.se.com/ww/en/product/ZB6AV1/white-pilot-light-head-%C3%B816-for-integral-led/, приступљено [1.11.2020]

[18] Доступно на: https://www.se.com/ww/en/product/ZB6EB4B/red-light-block-with-body-fixing-collar-with-integral-led-12...24v/, приступљено [13.10.2020]

[19] Доступно на: https://www.se.com/ww/en/product/ZB6EB3B/green-light-block-with-body-fixing-collar-with-integral-led-12...24v/, приступљено [1.11.2020]

[20] Доступно на: https://www.se.com/ww/en/product/ZB6EB5B/yellow-light-block-with-body-fixing-collar-with-integral-led-12...24v/, приступљено [1.11.2020]

[21] Доступно на: https://www.se.com/ww/en/product/ZB6EB1B/white-light-block-with-body-fixing-collar-with-integral-led-12...24v/, приступљено [1.11.2020]

[22] Доступно на: https://www.se.com/ww/en/product/ZB6CD25/black-square-selector-switch-head-%C3%B816-3-position-spring-return/, приступљено [1.11.2020]

[23] Доступно на: https://www.se.com/ww/en/product/ZB6DE1/white-rectang-projecting-illuminated-pushbutton-head-%C3%B816-spring-return-12...24v/, приступљено [1.11.2020]

[24] Доступно на: https://www.se.com/ww/en/product/XB6AS8349B/red-%C3%B830-emergency-stop-pushbutton-%C3%B816-trigger-latching-turn-release-1no%2B2nc/, приступљено [1.11.2020]

[25] Доступно на: https://www.se.com/ww/en/product/ZB6E1B/single-contact-block-for-head-%C3%B816-1no-faston-connector/, приступљено [1.11.2020]

[26] Доступно на: https://www.se.com/ww/en/product/ZB6E2B/single-contact-block-for-head-%C3%B816-1nc-faston-connector/, приступљено [1.11.2020]

[27] Доступно на: https://www.se.com/ww/en/product/ZB6Y009/body-fixing-collar-for-electrical-block/, приступљено [1.11.2020]

[28] Доступно на: https://eu.mouser.com/ProductDetail/Vishay-Sfernice/PE30L0FR222MAB?qs=Ub2JFBCWywKOKuJU8TGmTg%3D%3D, приступљено [14.10.2020]

[29] Доступно на: https://automation-insights.blog/2014/03/05/basic-operating-principle-of-an-inductive-proximity-sensor/, приступљено [17.11.2020]

[30] Доступно на: https://www.se.com/ww/en/product/XS612B1PAL2/inductive-sensor-xs6-m12---l54mm---brass---sn4mm---12..48vdc---cable-2, приступљено [14.10.2020]

[31] Доступно на: https://automation-insights.blog/2017/06/07/what-is-a-capacitive-sensor/, приступљено [17.11.2020]

[32] Доступно на: https://www.se.com/ww/en/product/XT230A1PAL2/capacitive-sensor---xt1---cylindrical-m30---plastic---sn-15-mm---cable-2-m/, приступљено [14.10.2020]

[33] Доступно на: https://robu.in/photoelectric-proximity-sensor-methods-advantages-applications/, приступљено [17.11.2020]

[34] Доступно на: https://www.se.com/ww/en/product/XUB4APANL2/photo-electric-sensor---xub---diffuse---sn-0.1m---12..24vdc---cable-2m/, приступљено [14.10.2020]

[35] Доступно на: https://www.se.com/ww/en/product/TSXCANTDM4/ip20-canopen-tap-junction-box---1-screw-terminal-block-and-4-sub-d9/?range=577-canopen&selected-node-id=12367564172, приступљено [14.10.2020]

[36] Доступно на: https://cselectric.co.in/blog/residual-current-circuit-breaker-rccb/?fbclid=IwAR3J6ZYXET7xign23PneghLGRxMOZdqUDC9EwowG-P3WNG5X14Ckwlp9AIk#1, приступљено [17.11.2020]

[37] Доступно на: https://www.se.com/ww/en/product/A9Z21225/acti-9-iid---rccb---2p---25a---30ma---type-a/, приступљено [15.10.2020]

[38] Доступно на: http://www.guilleviniag.com/downloads/Products/Schneider/Software/Unity/, приступљено [16.10.2020]

[39] W.Bolton, „Programmable Logic Controllers“, *Elsevier Newnes*, 2006.

[40] Schneider Electric, „Altivar Machine ATV320 Programming Manual“, 4.2018.

О ауторима

др БОЈАН З. КНЕЖЕВИЋ, дипл. инж. ел. тех., ванредни професор (26. октобар 1982, Бања Лука). Живи и ради у Бањој Луци гдје је завршио Основну школу „Вук С. Караџић“, средњу Електротехничку школу „Никола Тесла“ и Електротехнички факултет. Радио је у предузећу Конел, Лакташи. Мастер студиј завршио је на Електротехничком факултету у Београду. Докторирао је на Факултету техничких наука у Новом Саду. На Машинском факултету у Бањој Луци биран је за асистента, вишег асистента, доцента и ванредног професора. Обнаша неколико функција међу којима функцију Шефа Катедре за мехатронику и роботику и продекана за научноистраживачки рад и међународну сарадњу. Објавио је више од 40 научних радова у међународним и домаћим часописима и зборницима са конференција. Члан је уређивачког одбора неколико иностраних научних часописа и домаћих научних конференција. Добитник је награде Универзитета за остварене међународне резултате у научно-истраживачком раду 2017. године. Аутор је преко 100 стручних пројеката из области мехатронике, роботизације, аутоматизације и лифтоградње. Муж и отац.

др АЛМИР ОСМАНОВИЋ, дипл. инж. маш., ванредни професор (11. новембар 1980, Тузла). Живи и ради у Тузли гдје је завршио Основну школу и Мјешовиту средњу машинску школу и Машински факултет. 2004. године дипломирао је на Машинском факултету Универзитета у Тузли, одсјек Производно машинство. На истом факултету запослен је од 2005. године. На постдипломском студију Мехатронике магистрирао је 2009. године, а докторску дисертацију одбранио је 2014. године. У звање ванредног професора изабран је 2019. године на Машинском факултету Универзитета у Тузли, одсјек Мехатроника, област Мехатронички инжињеринг. Његов наставни и научно-истраживачки рад обухвата мехатронику, индустријску аутоматизацију и роботику, хидрауличке и пнеуматске системе, управљање мехатроничким системима. Аутор и коаутор је више од 30 научних и стручних радова и четири књиге. Учествовао је у реализацији домаћих и међународних истраживачких пројеката и стручних усавршавања у иностранству. Члан је два техничка комитета Института за стандардизацију Босне и Херцеговине и активно користи енглески језик.

ISBN XXX-XXXXX-XX-XX-X