Sprawozdanie - GE Fanuc

Kewin Gałuszka
nr indeksu 241624
Kacper Starościak
nr indeksu

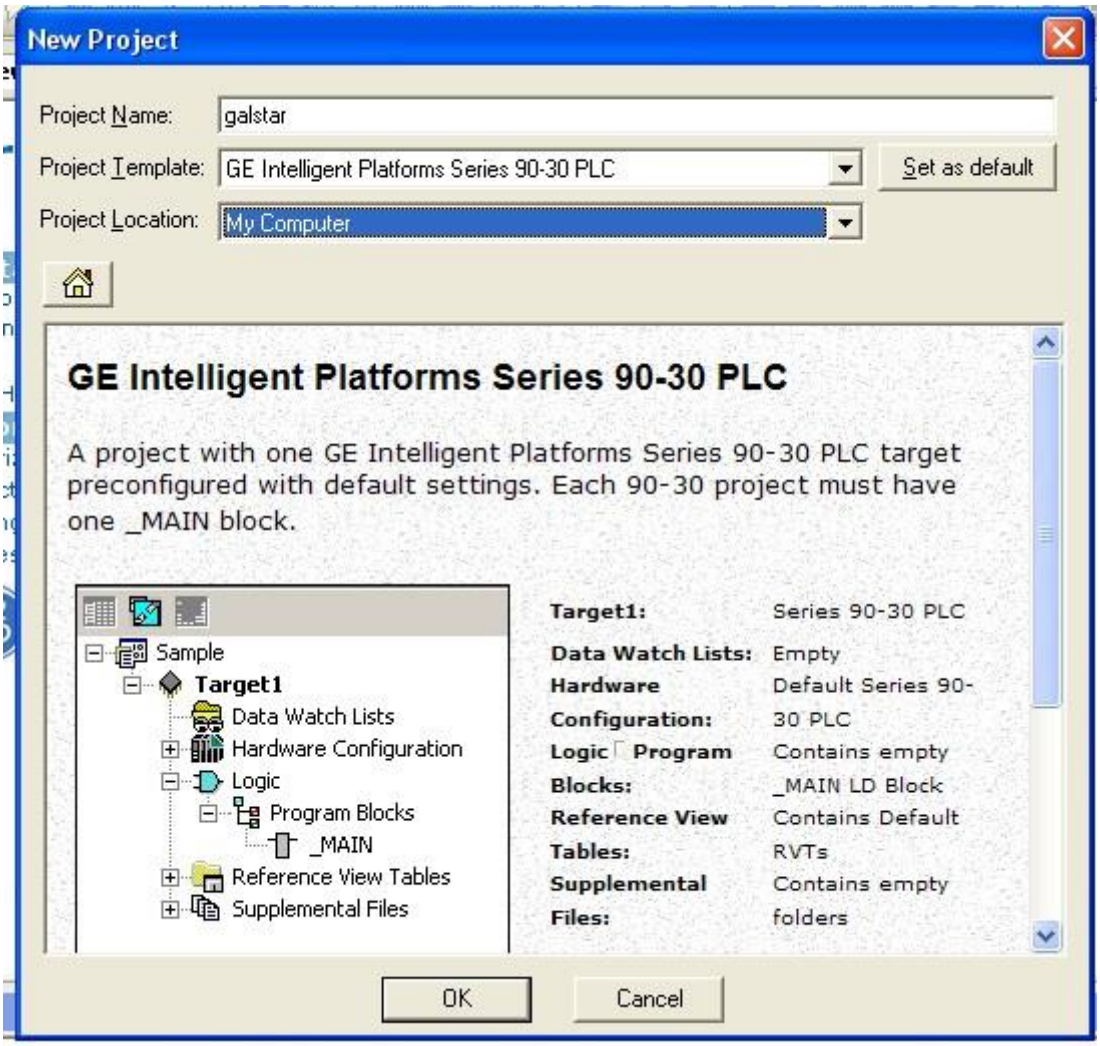
1. **Cel ćwiczenia**
   Celem ćwiczenia jest zapoznanie się z działaniem sterownika PLC GE Fanuc oraz nabycie podstawowych umiejętności programowania ww. sterownika.
2. **Wstęp teoretyczny**

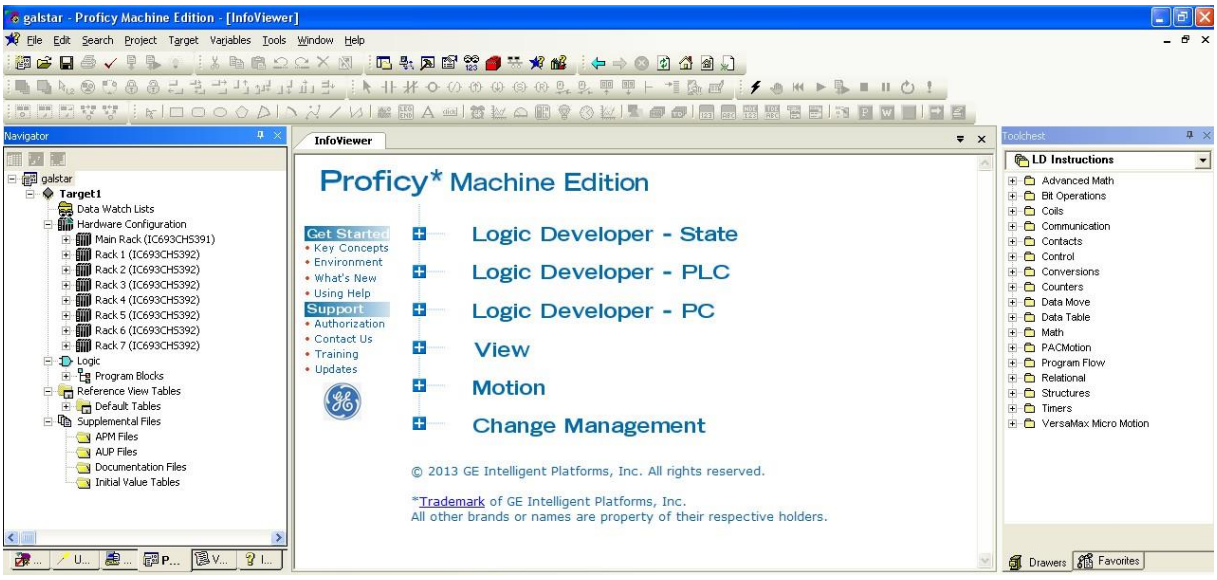
   Programowanie sterowników GE Fanuc obejmuje dwa etapy
   a) konfiguracja sterownika
   Konfigurowanie sterownika ma na celu między innymi poinformowanie sterownika o modułach, które zostały podłączone do kolejnych slotów płyty łączeniowej oraz podaniu adresów logicznych, których zostały przypisane fizycznym wejściom i wyjściom.
   b) tworzenie logiki programu
   Tworzenie logiki programu odbywa się w języku drabinkowym. Po stworzeniu programu sterowania, program wprowadzany jest do pamięci sterownika i uruchamiany.

   Moduły wykorzystywane w GE Fanuc
   - zasilacz PWR
   - jednostka CPU
   - moduły wejść i wyjść (cyfrowych i analogowych)
   - moduły sterowników sieci LAN
   - moduły komunikacyjne do sieci Ethernet
   oraz inne np. Moduł szybkiego licznika HSC

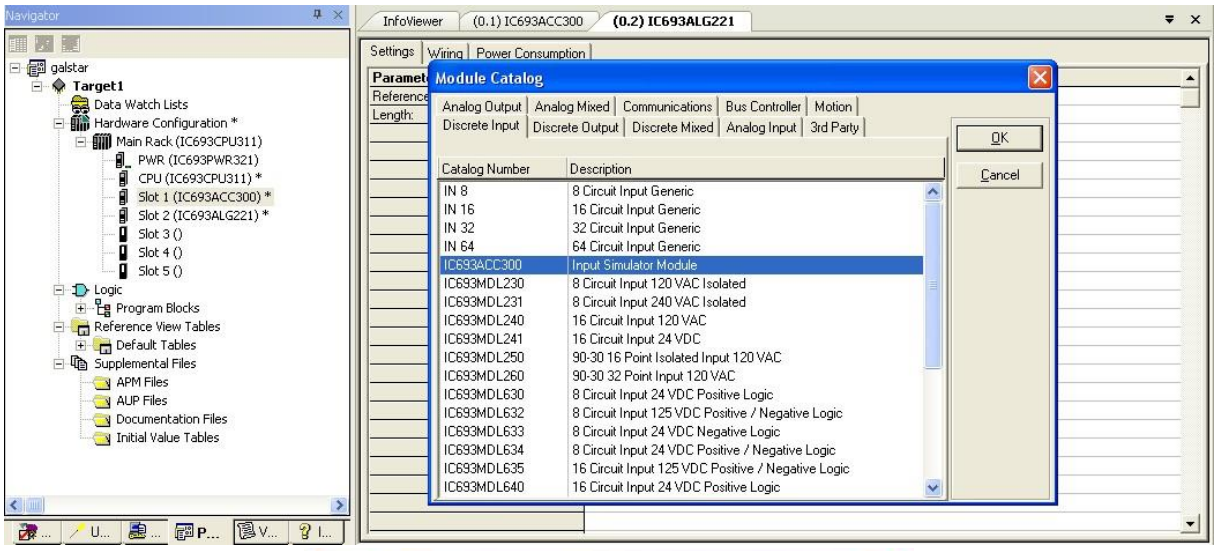
3. **Przebieg ćwiczenia**
   Ćwiczenie rozpoczęto od konfiguracji sterownika, zgodnie z przedstawioną instrukcją.

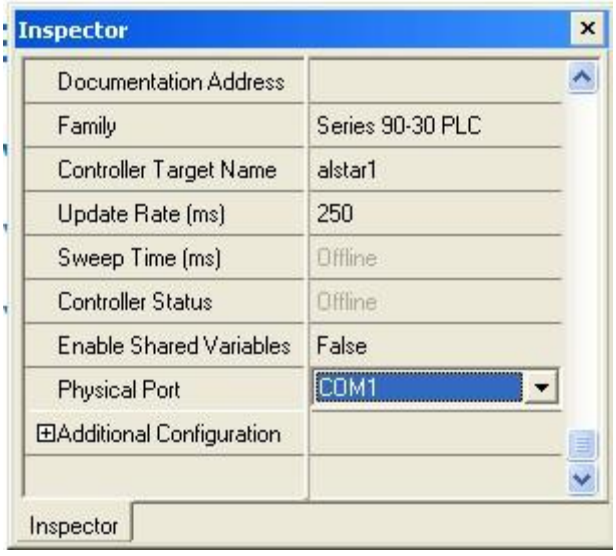
Stworzono nowy projekt

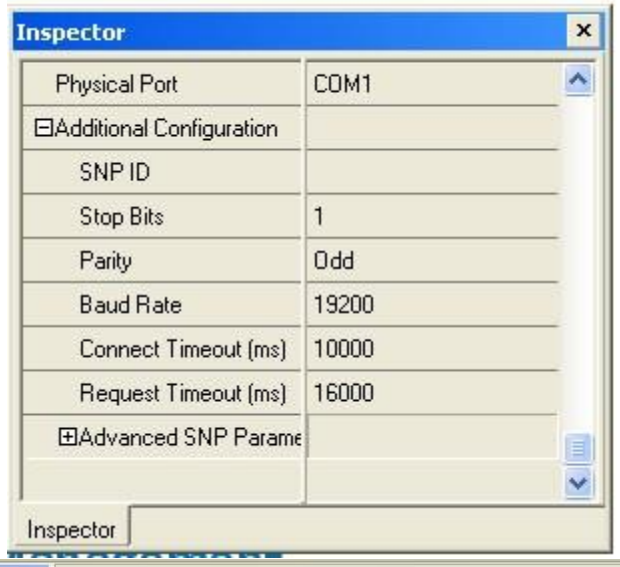


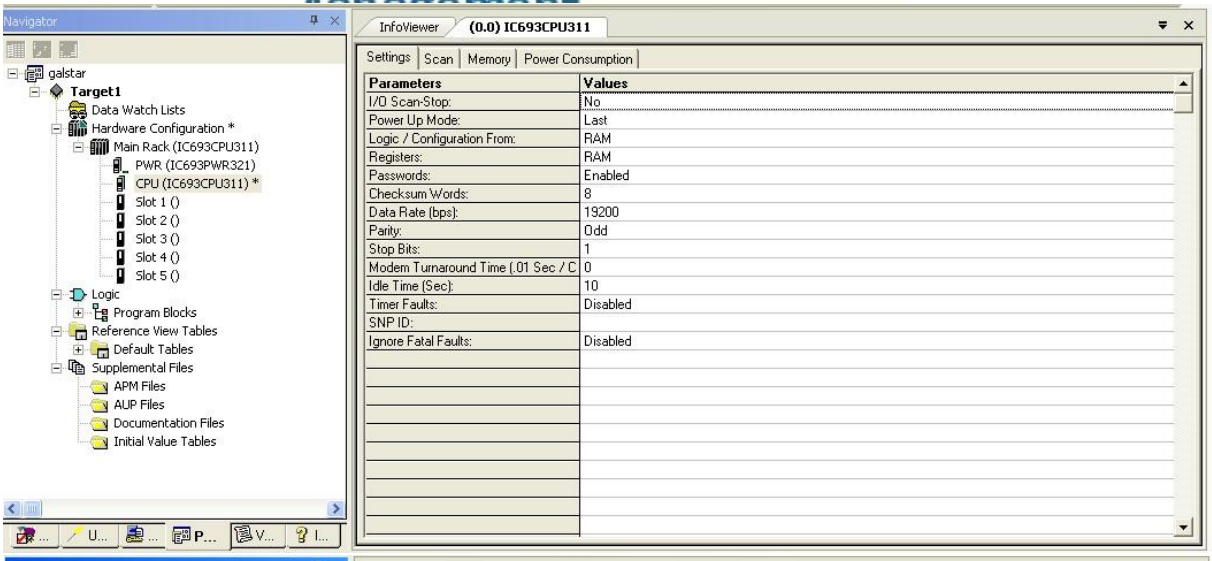


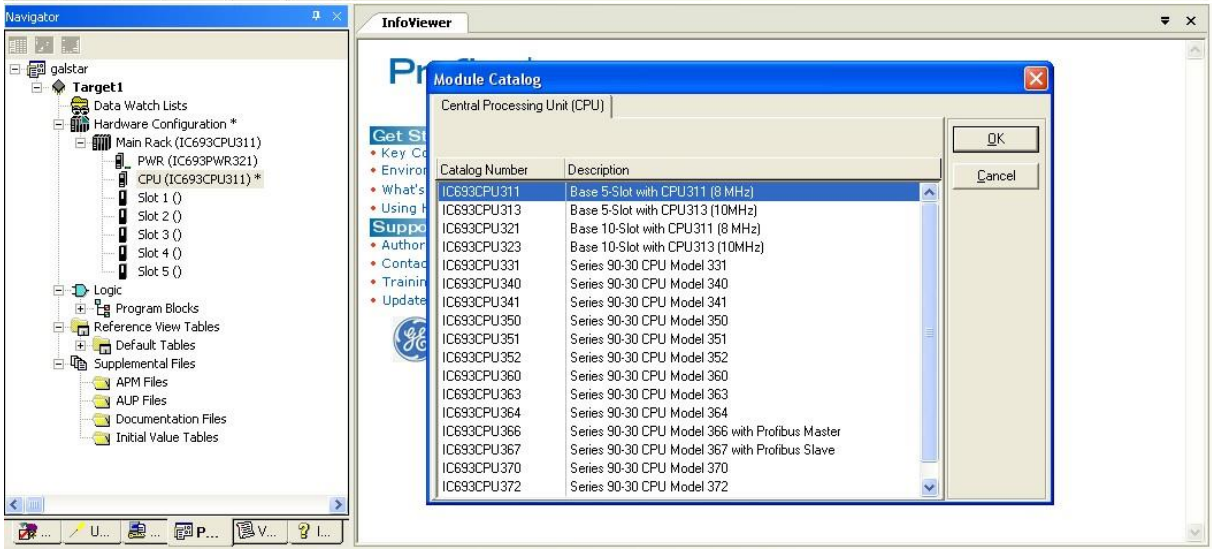
Kolejno przystąpiono do konfiguracji każdego z zamontowanych modułów. Modele odczytano z tabliczek umieszczonych na modułach.

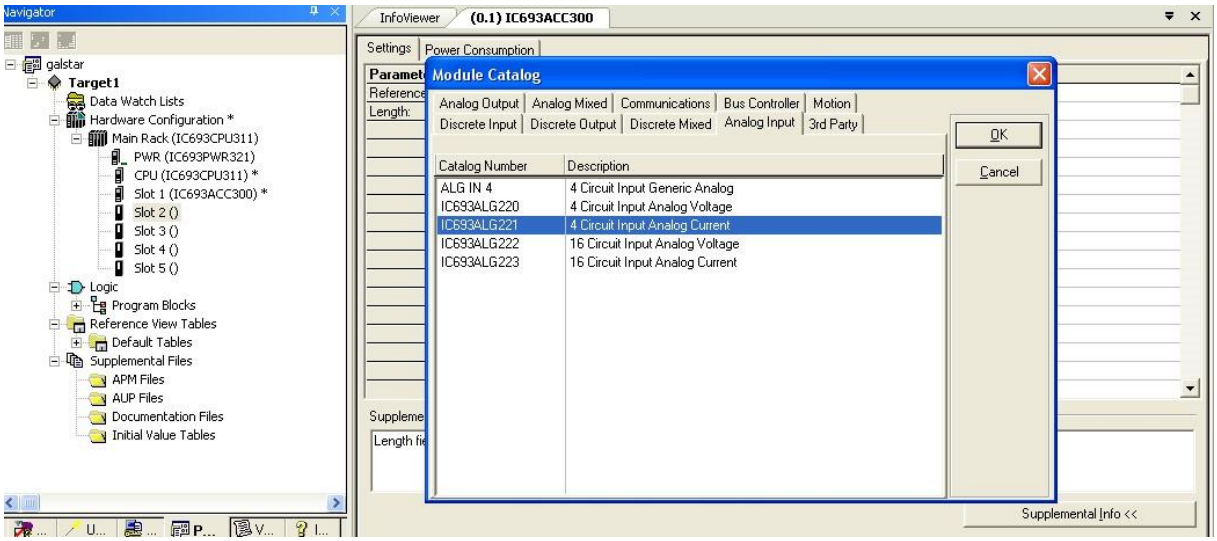


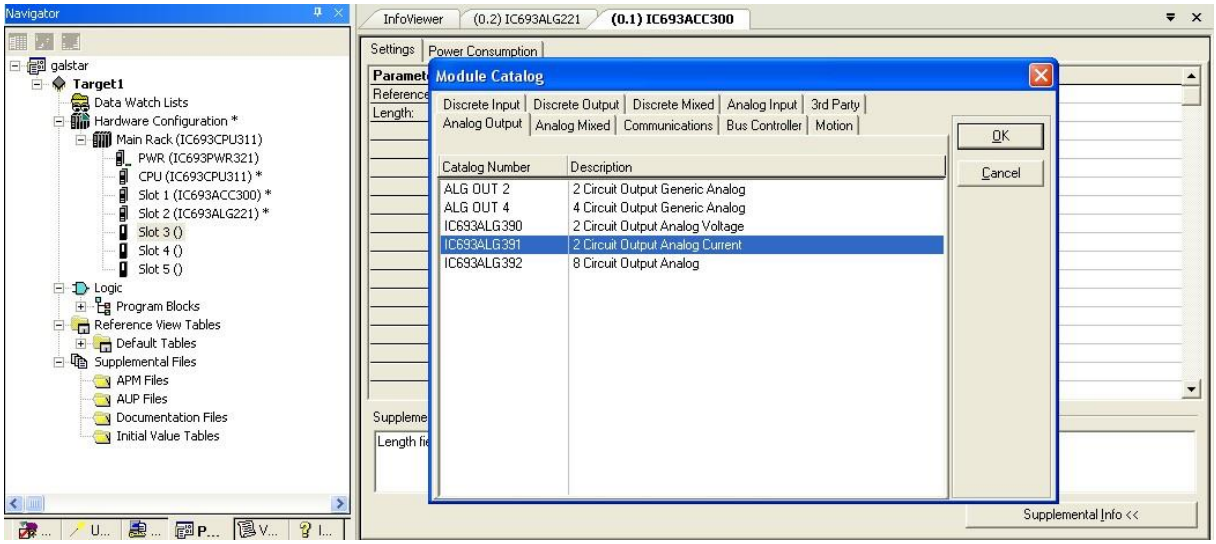


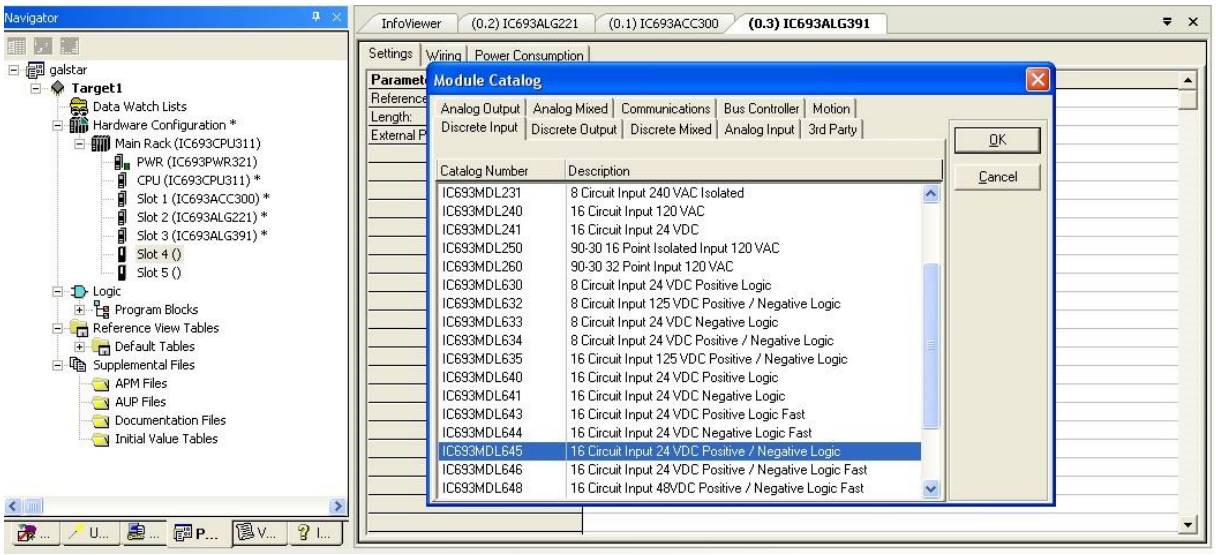

Inspector
Physical Port
COM1
Additional Configuration
SNP ID
Stop Bits
1
Parity
Odd
Baud Rate
19200
Connect Timeout (ms)
10000
Request Timeout (ms)
16000
Advanced SNP Parame
Inspector

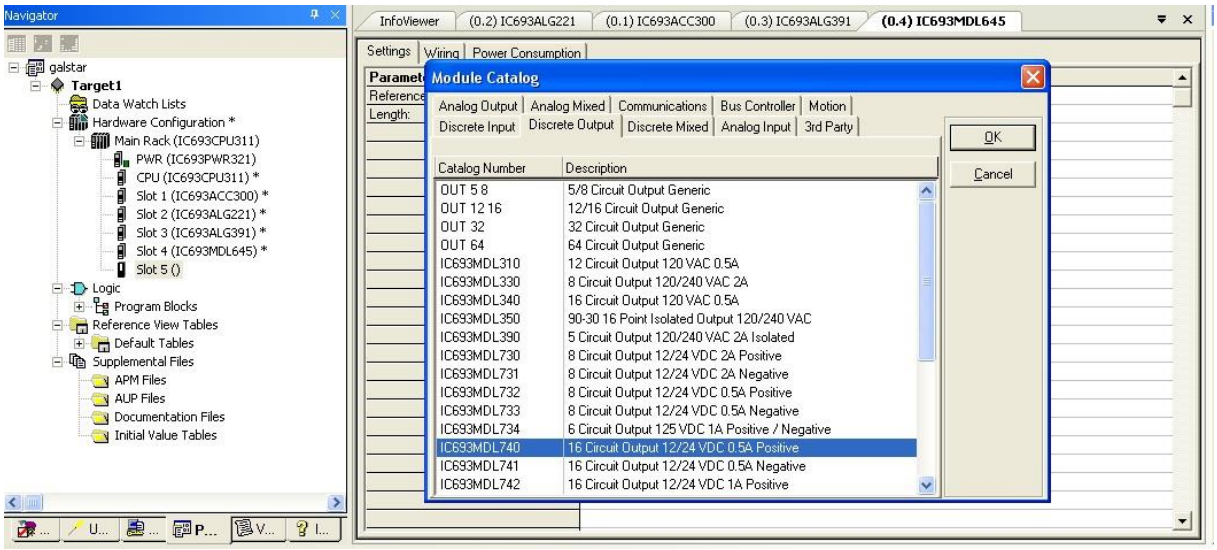

Navigator
galstar
Target1
Data Watch Lists
Hardware Configuration *
Main Rack (IC693CPU311)
PWR (IC693PWR321)
CPU (IC693CPU311) *
Slot 1 ()
Slot 2 ()
Slot 3 ()
Slot 4 ()
Slot 5 ()
Logic
Program Blocks
Reference View Tables
Default Tables
Supplemental Files
APM Files
AUP Files
Documentation Files
Initial Value Tables
InfoViewer
(0.0) IC693CPU311
Settings
Scan
Memory
Power Consumption
Parameters
Values
I/O Scan-Stop: No
Power Up Mode: Last
Logic / Configuration From: RAM
Registers: RAM
Passwords: Enabled
Checksum Words: 8
Data Rate (bps): 19200
Parity: Odd
Stop Bits: 1
Modem Turnaround Time (.01 Sec / C 0
Idle Time (Sec): 10
Timer Faults: Disabled
SNP ID:
Ignore Fatal Faults: Disabled


Navigator
galstar
Target1
Data Watch Lists
Hardware Configuration *
Main Rack (IC693CPU311)
PWR (IC693PWR321)
CPU (IC693CPU311) *
Slot 1 ()
Slot 2 ()
Slot 3 ()
Slot 4 ()
Slot 5 ()
Logic
Program Blocks
Reference View Tables
Default Tables
Supplemental Files
APM Files
AUP Files
Documentation Files
Initial Value Tables
InfoViewer
Module Catalog
Central Processing Unit (CPU)
OK
Cancel
Catalog Number
Description
IC693CPU311 Base 5-Slot with CPU311 (8 MHz)
IC693CPU313 Base 5-Slot with CPU313 (10MHz)
IC693CPU321 Base 10-Slot with CPU311 (8 MHz)
IC693CPU323 Base 10-Slot with CPU313 (10MHz)
IC693CPU331 Series 90-30 CPU Model 331
IC693CPU340 Series 90-30 CPU Model 340
IC693CPU341 Series 90-30 CPU Model 341
IC693CPU350 Series 90-30 CPU Model 350
IC693CPU351 Series 90-30 CPU Model 351
IC693CPU352 Series 90-30 CPU Model 352
IC693CPU360 Series 90-30 CPU Model 360
IC693CPU363 Series 90-30 CPU Model 363
IC693CPU364 Series 90-30 CPU Model 364
IC693CPU366 Series 90-30 CPU Model 366 with Profibus Master
IC693CPU367 Series 90-30 CPU Model 367 with Profibus Slave
IC693CPU370 Series 90-30 CPU Model 370
IC693CPU372 Series 90-30 CPU Model 372

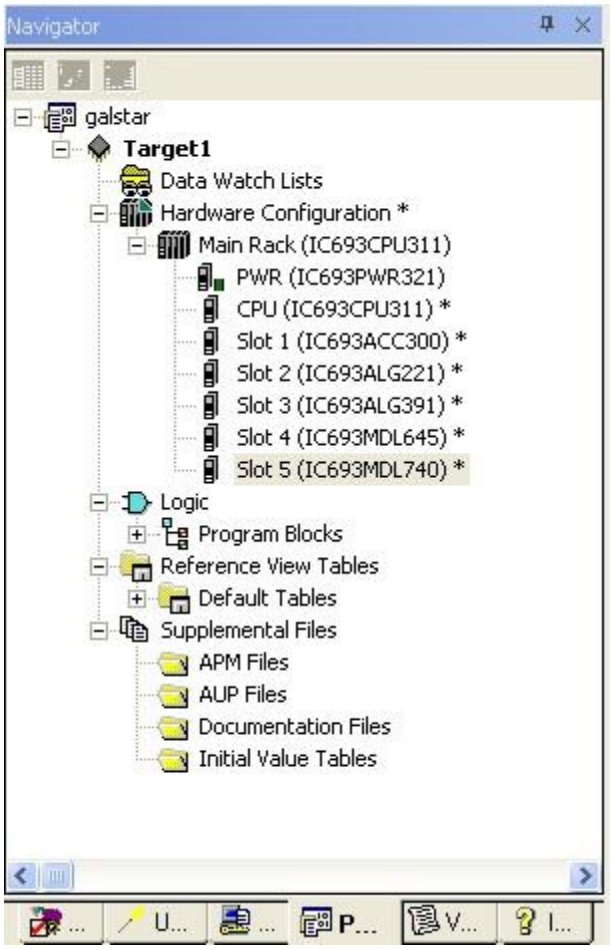

Navigator
galstar
Target1
Data Watch Lists
Hardware Configuration *
Main Rack (IC693CPU311)
PWR (IC693PWR321)
CPU (IC693CPU311) *
Slot 1 (IC693ACC300) *
Slot 2 ()
Slot 3 ()
Slot 4 ()
Slot 5 ()
Logic
Program Blocks
Reference View Tables
Default Tables
Supplemental Files
APM Files
AUP Files
Documentation Files
Initial Value Tables
InfoViewer
(0.1) IC693ACC300
Settings
Power Consumption
Module Catalog
Analog Output
Analog Mixed
Communications
Bus Controller
Motion
Discrete Input
Discrete Output
Discrete Mixed
Analog Input
3rd Party
OK
Cancel
Catalog Number
Description
ALG IN 4
4 Circuit Input Generic Analog
IC693ALG220
4 Circuit Input Analog Voltage
IC693ALG221
4 Circuit Input Analog Current
IC693ALG222
16 Circuit Input Analog Voltage
IC693ALG223
16 Circuit Input Analog Current
Supplemental Info <<


Navigator
galstar
Target1
Data Watch Lists
Hardware Configuration *
Main Rack (IC693CPU311)
PWR (IC693PWR321)
CPU (IC693CPU311) *
Slot 1 (IC693ACC300) *
Slot 2 (IC693ALG221) *
Slot 3 ()
Slot 4 ()
Slot 5 ()
Logic
Program Blocks
Reference View Tables
Default Tables
Supplemental Files
APM Files
AUP Files
Documentation Files
Initial Value Tables
InfoViewer
(0.2) IC693ALG221
(0.1) IC693ACC300
Settings
Power Consumption
Module Catalog
Discrete Input
Discrete Output
Discrete Mixed
Analog Input
3rd Party
Analog Output
Analog Mixed
Communications
Bus Controller
Motion
OK
Cancel
Catalog Number
Description
ALG OUT 2
2 Circuit Output Generic Analog
ALG OUT 4
4 Circuit Output Generic Analog
IC693ALG390
2 Circuit Output Analog Voltage
IC693ALG391
2 Circuit Output Analog Current
IC693ALG392
8 Circuit Output Analog
Supplemental Info <<

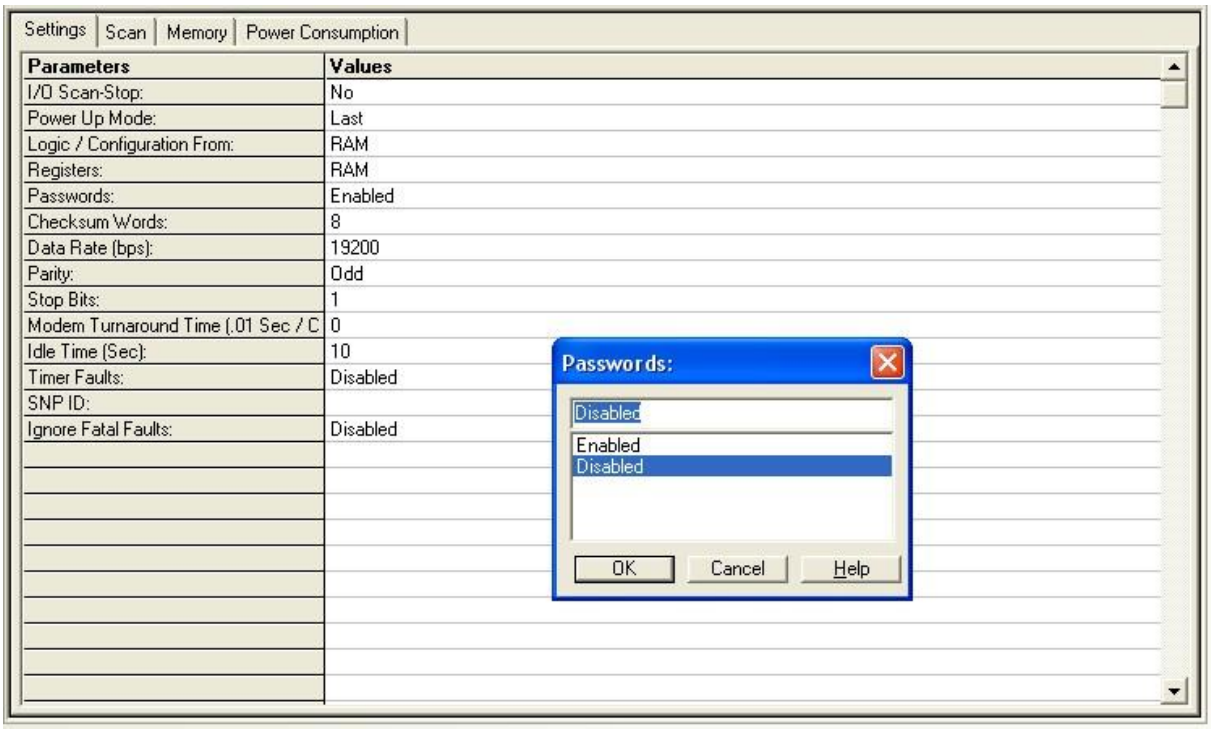

Navigator
galstar
Target1
Data Watch Lists
Hardware Configuration *
Main Rack (IC693CPU311)
PWR (IC693PWR321)
CPU (IC693CPU311) *
Slot 1 (IC693ACC300) *
Slot 2 (IC693ALG221) *
Slot 3 (IC693ALG391) *
Slot 4 ()
Slot 5 ()
Logic
Program Blocks
Reference View Tables
Default Tables
Supplemental Files
APM Files
AUP Files
Documentation Files
Initial Value Tables
InfoViewer
(0.2) IC693ALG221
(0.1) IC693ACC300
(0.3) IC693ALG391
Settings
Wiring
Power Consumption
Module Catalog
Analog Output
Analog Mixed
Communications
Bus Controller
Motion
Discrete Input
Discrete Output
Discrete Mixed
Analog Input
3rd Party
OK
Cancel
Catalog Number
Description
IC693MDL231
8 Circuit Input 240 VAC Isolated
IC693MDL240
16 Circuit Input 120 VAC
IC693MDL241
16 Circuit Input 24 VDC
IC693MDL250
90-30 16 Point Isolated Input 120 VAC
IC693MDL260
90-30 32 Point Input 120 VAC
IC693MDL630
8 Circuit Input 24 VDC Positive Logic
IC693MDL632
8 Circuit Input 125 VDC Positive / Negative Logic
IC693MDL633
8 Circuit Input 24 VDC Negative Logic
IC693MDL634
8 Circuit Input 24 VDC Positive / Negative Logic
IC693MDL635
16 Circuit Input 125 VDC Positive / Negative Logic
IC693MDL640
16 Circuit Input 24 VDC Positive Logic
IC693MDL641
16 Circuit Input 24 VDC Negative Logic
IC693MDL643
16 Circuit Input 24 VDC Positive Logic Fast
IC693MDL644
16 Circuit Input 24 VDC Negative Logic Fast
IC693MDL645
16 Circuit Input 24 VDC Positive / Negative Logic
IC693MDL646
16 Circuit Input 24 VDC Positive / Negative Logic Fast
IC693MDL648
16 Circuit Input 48VDC Positive / Negative Logic Fast



Widok okna po skonfigurowaniu jednostki CPU oraz wszystkich modułów

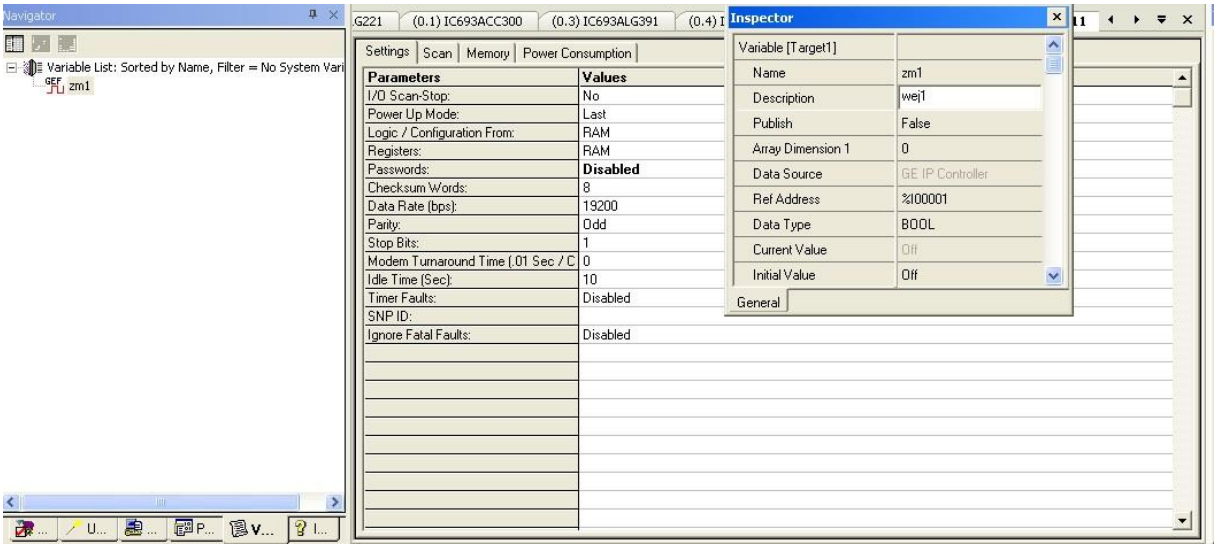


Zwrócono również uwagę na wyłączenie hasła.

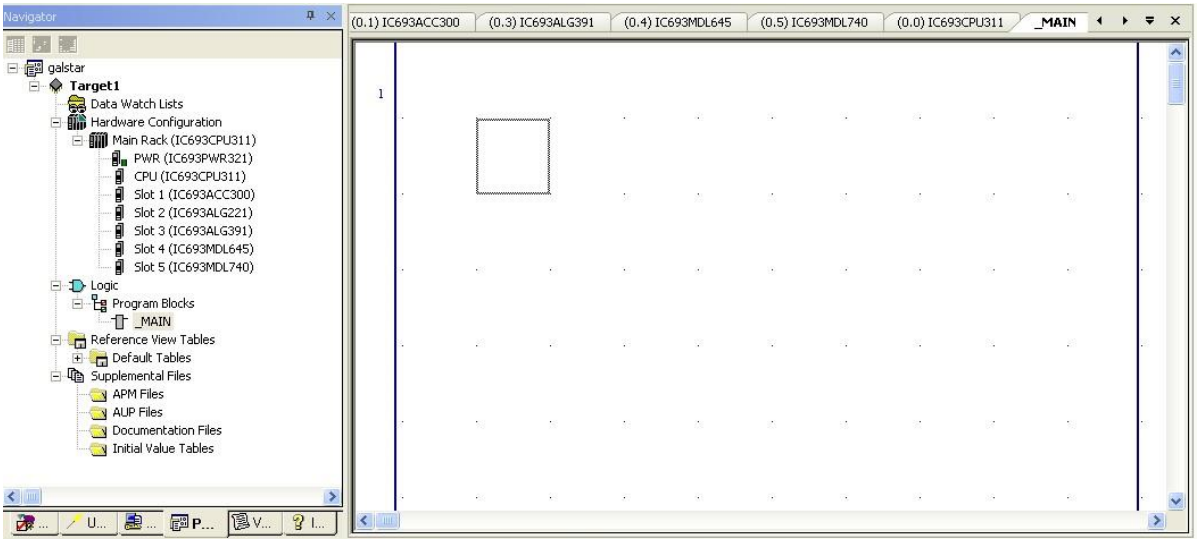


Rozpoczęto tworzenie testowego programu, który miał na celu sprawdzenie działania sterownika.

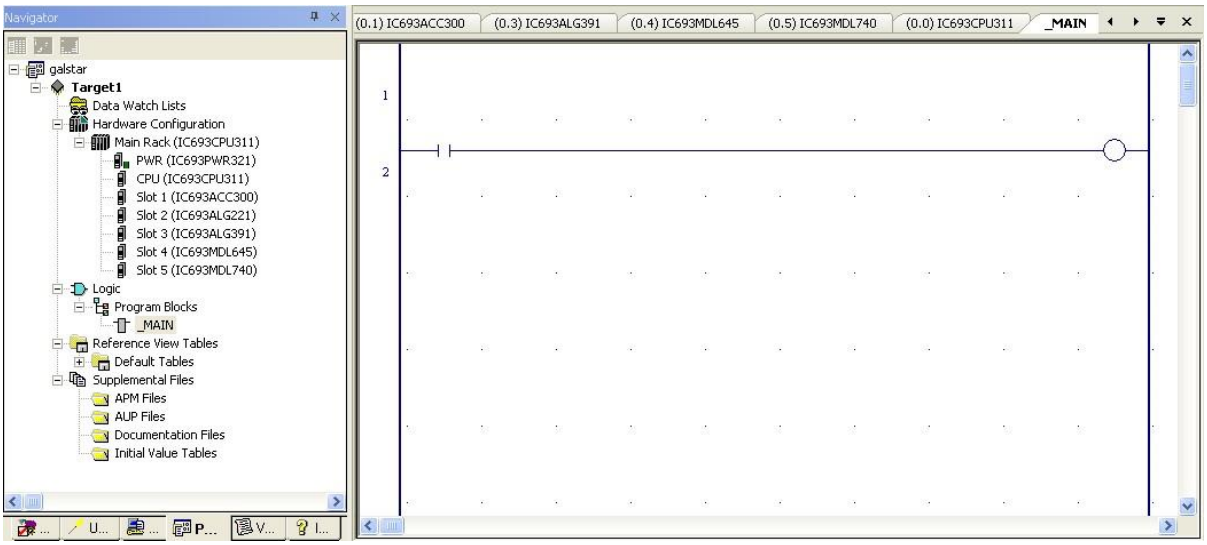
Dodano zmienną wejściową



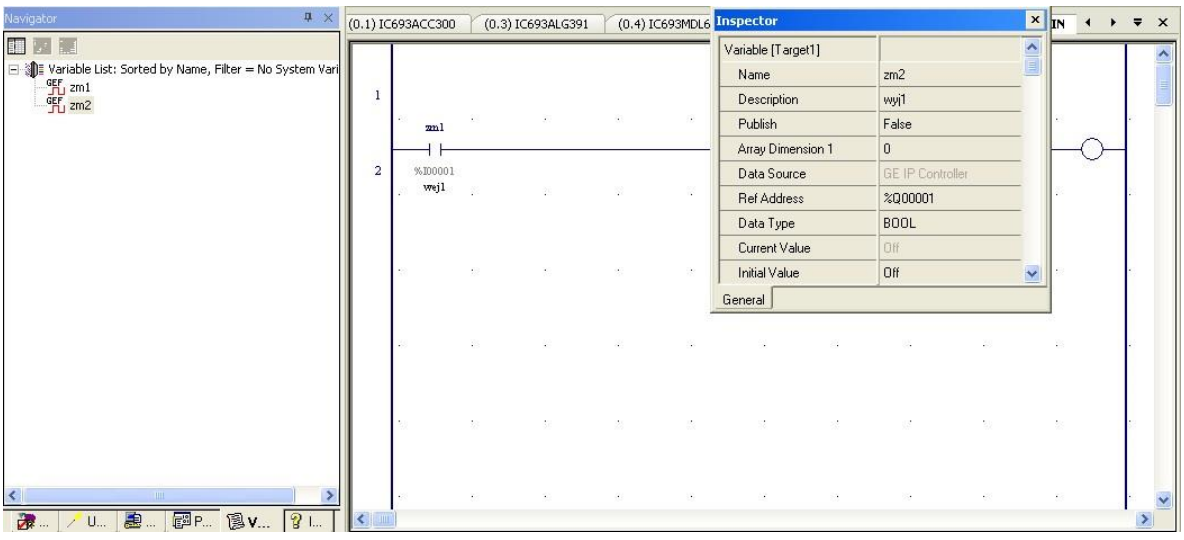
Widok okna edytora programu przed rozpoczęciem tworzenia logiki



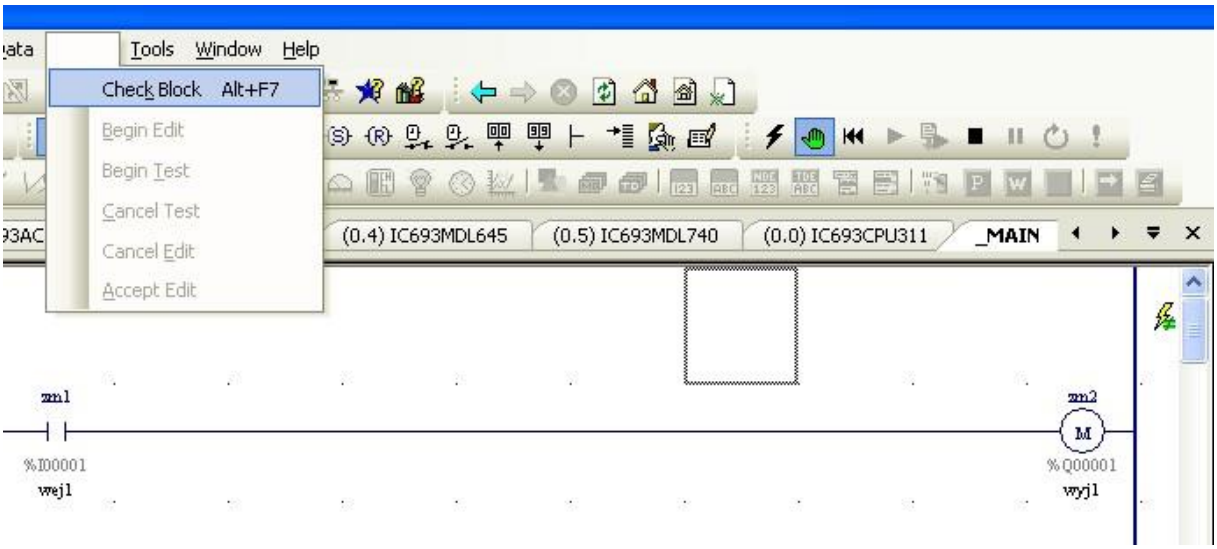
Stworzono logikę



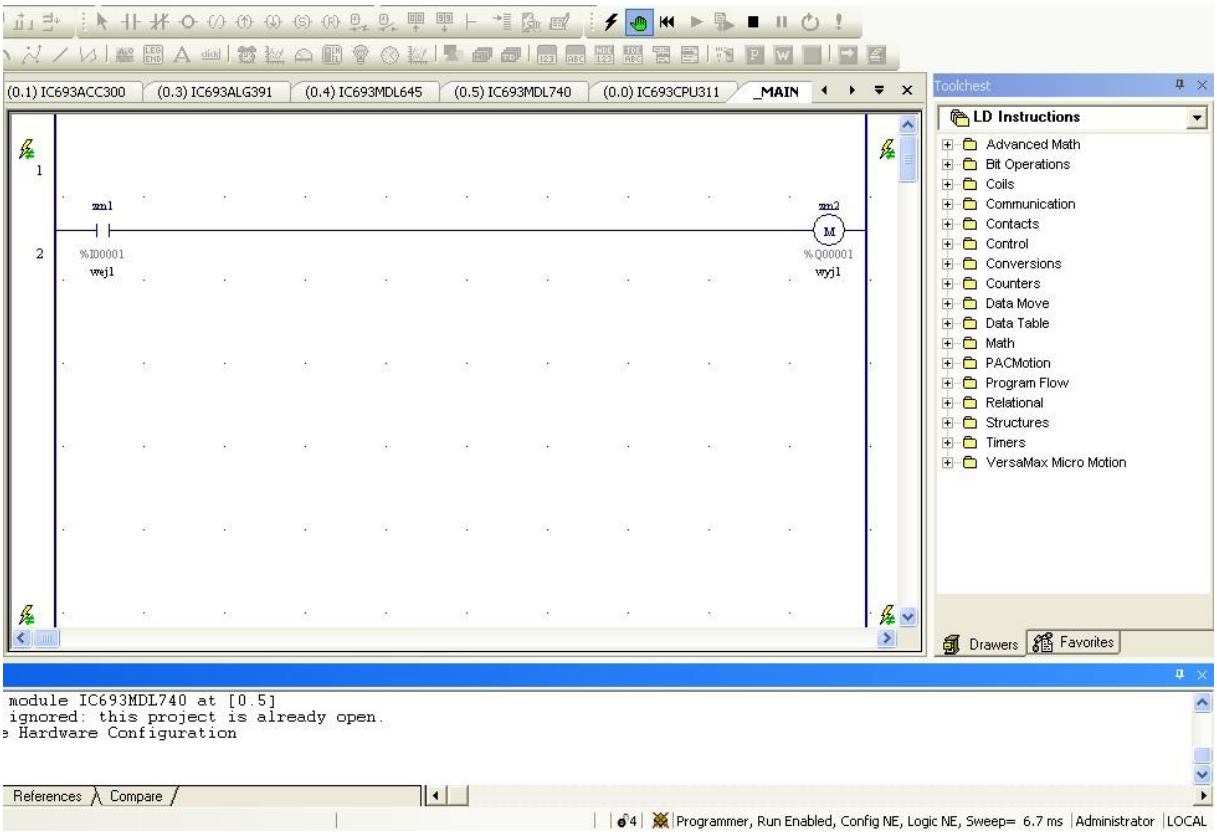
Oraz dodano drugą zmienną, tym razem wyjściową. Przypisano je do odpowiednio do cewki oraz styku.



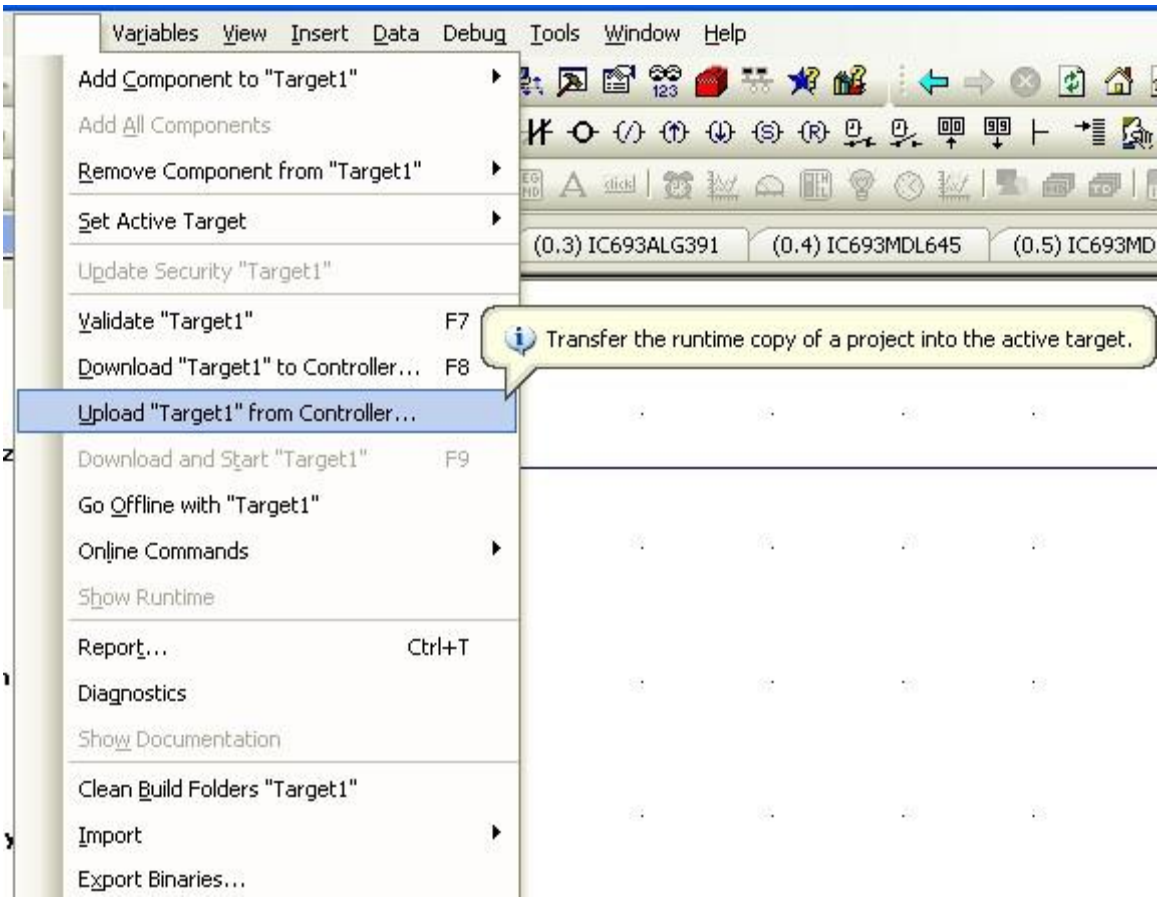
Sprawdzono, czy program nie zawiera błędów

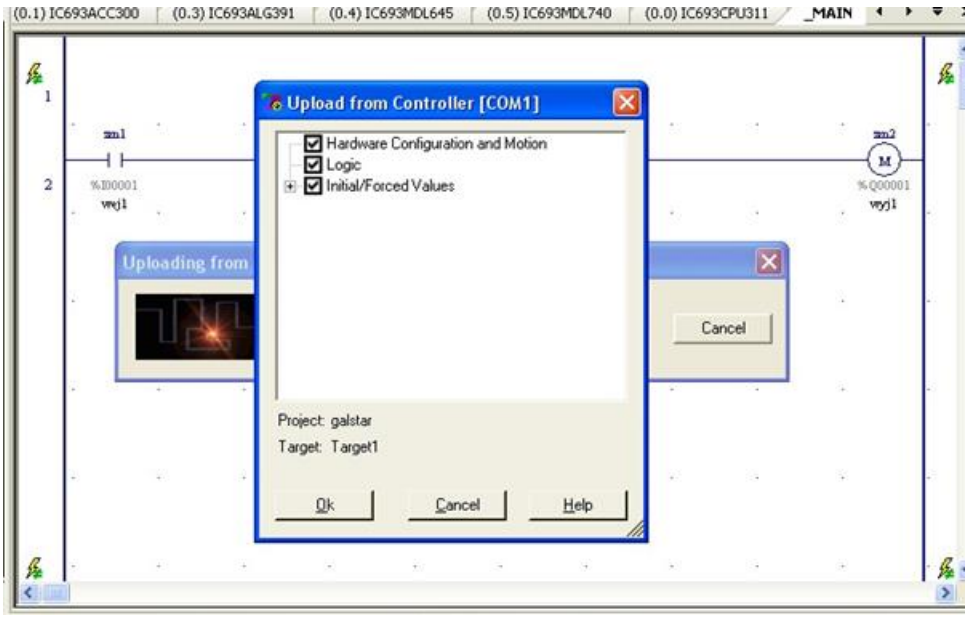


Widok okna edytora i konsoli po procesie debugowania

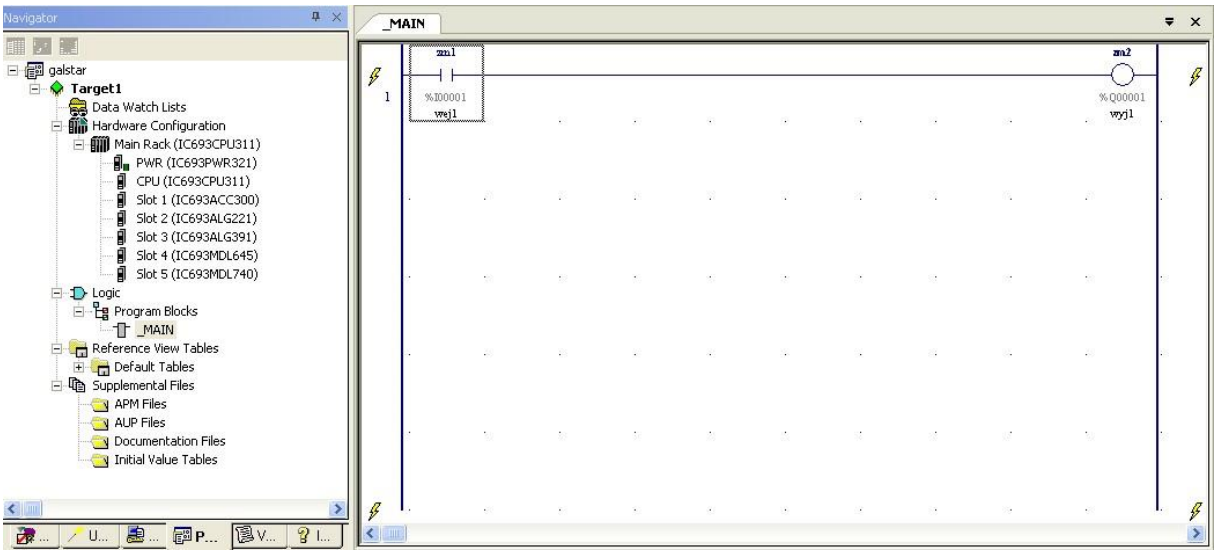


Rozpoczęto procedurę wgrania programu do pamięci sterownika PLC





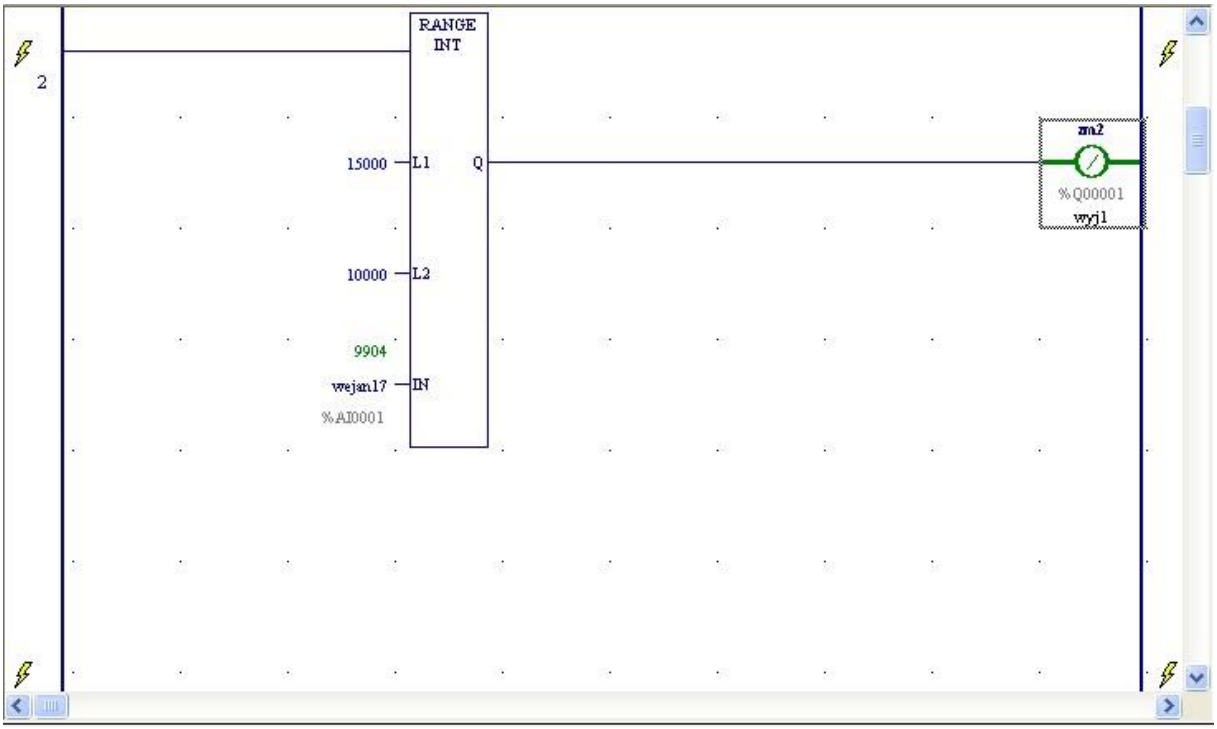
Gdy program został wgrany, można było w czasie rzeczywistym obserwować stany panujące na cewce i styku.



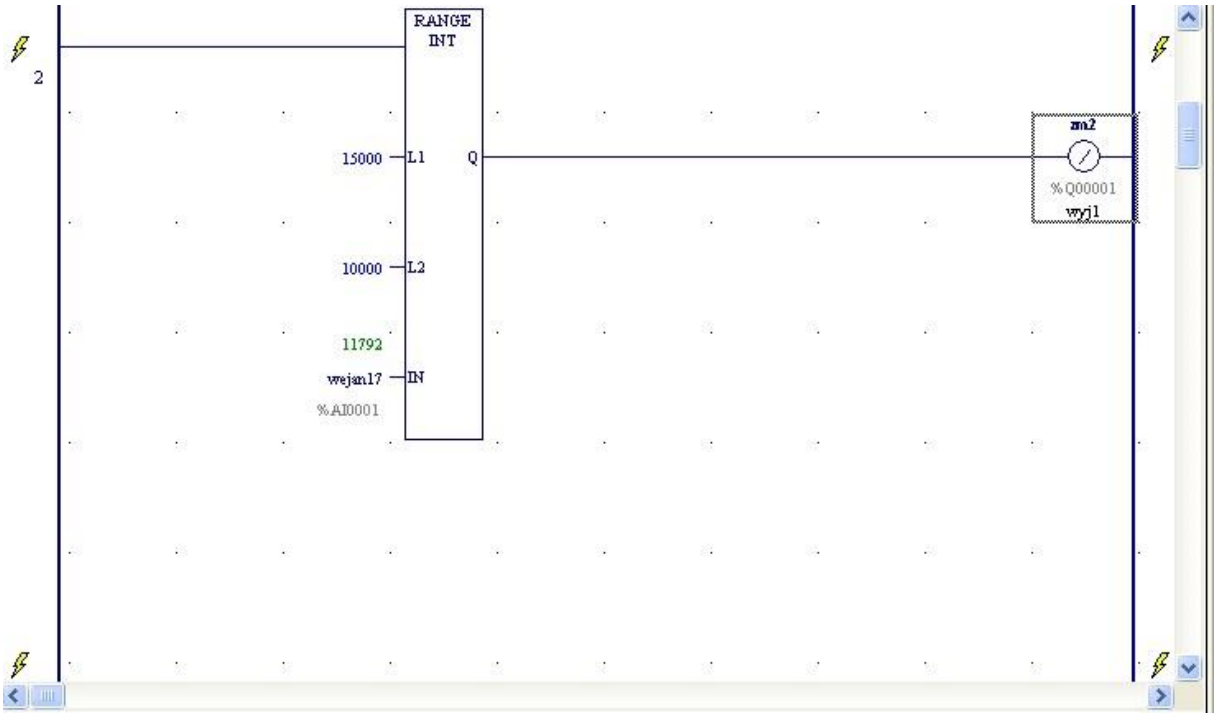
Przystąpiono do realizacji zadania 4.

1. Stworzono wskaźnik, informujący, czy wartość wejściowa znajduje się w bezpiecznym zakresie. Gdy wartość znajdowała się poza nim, następowała sygnalizacja tego zdarzenia. Obserwowano w czasie rzeczywistym działanie programu.

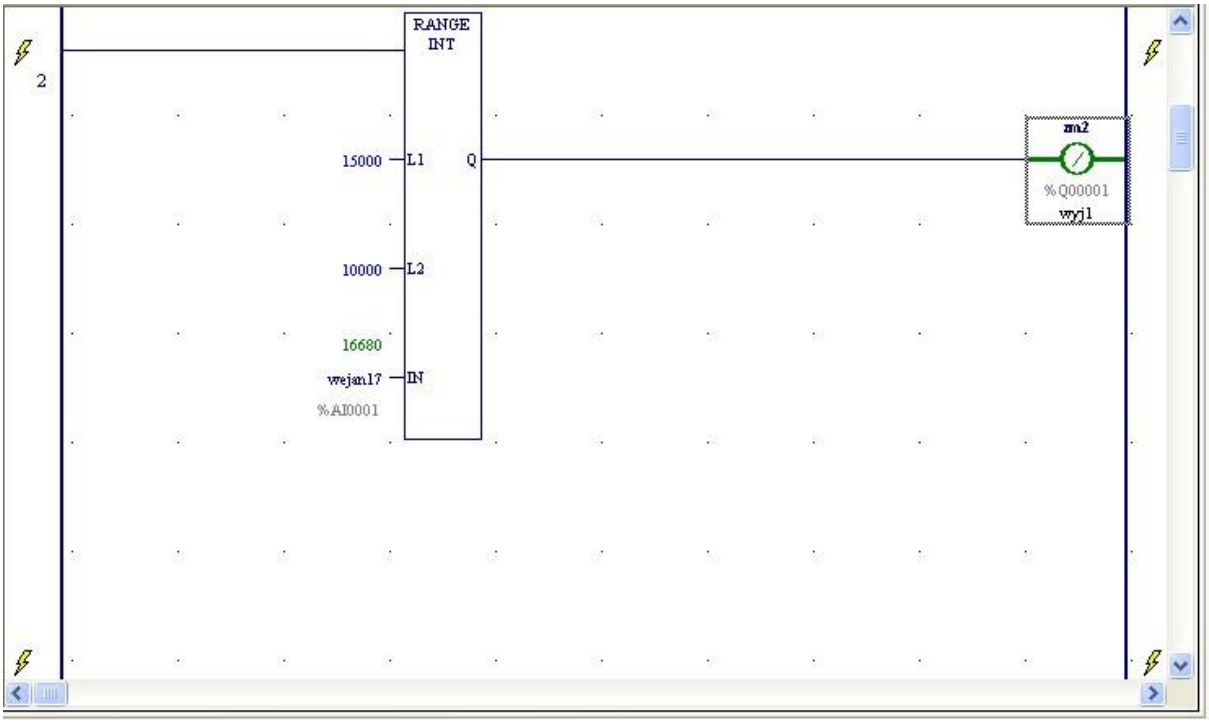
a) gdy wartość jest za niska następuje sygnalizacja



b) gdy wartość znajduje się w założonym przedziale, zgodnie z założeniem, nie uruchamia się sygnalizacja

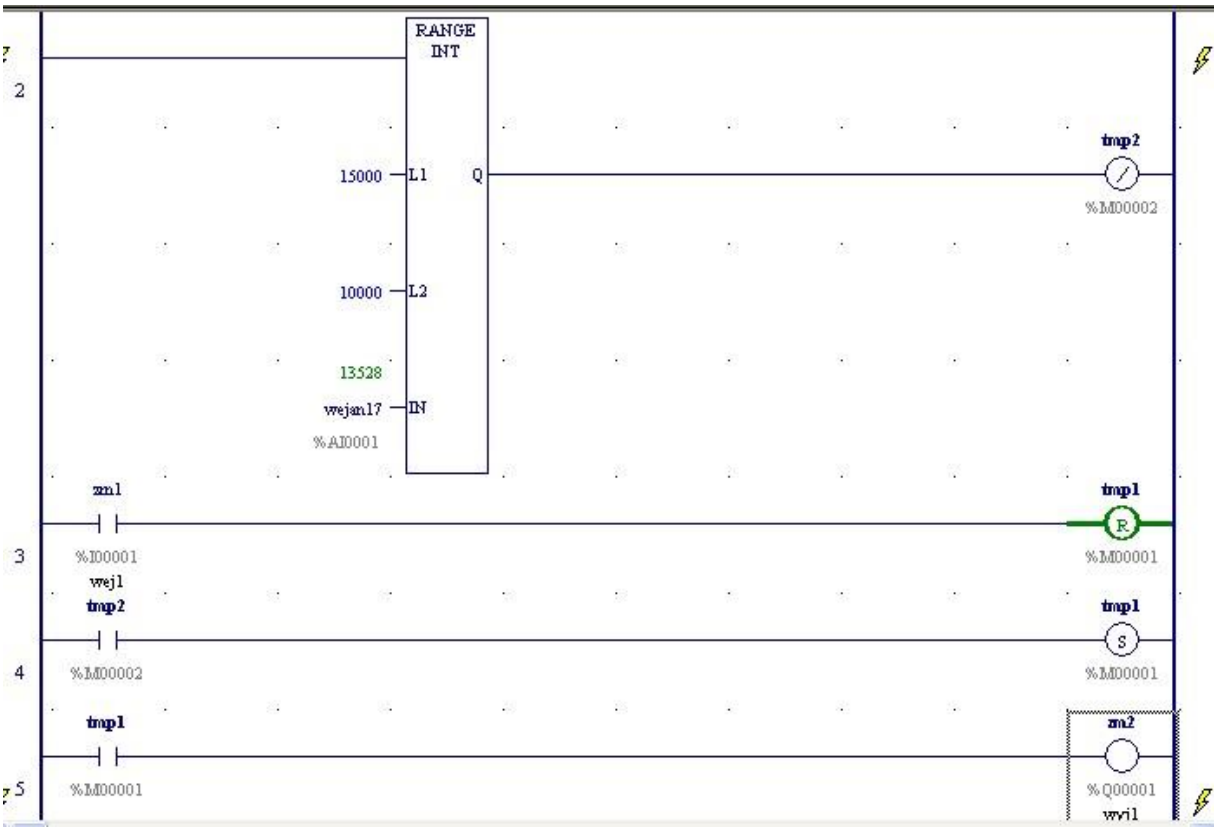


c) gdy wartość jest zbyt wysoka, również następuje alarm

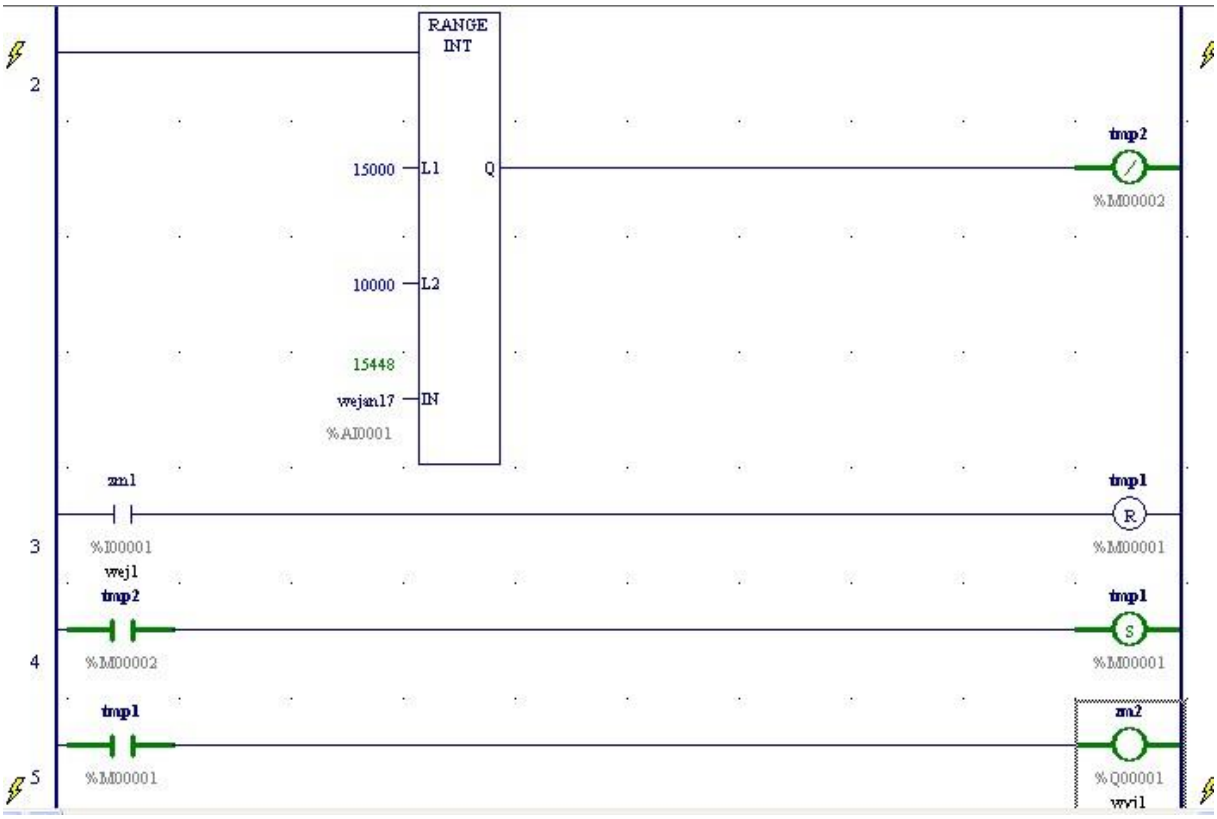


2. Do programu dodano funkcjonalność, w której po wyjściu z bezpiecznego zakresu następowała sygnalizacja tego zdarzenia i trwała ona aż do momentu wykasowania jej przez użytkownika przez fizyczny przełącznik. Podobnie jak w poprzednim przypadku, obserwowano w czasie rzeczywistym stany na wejściach i wyjściach.

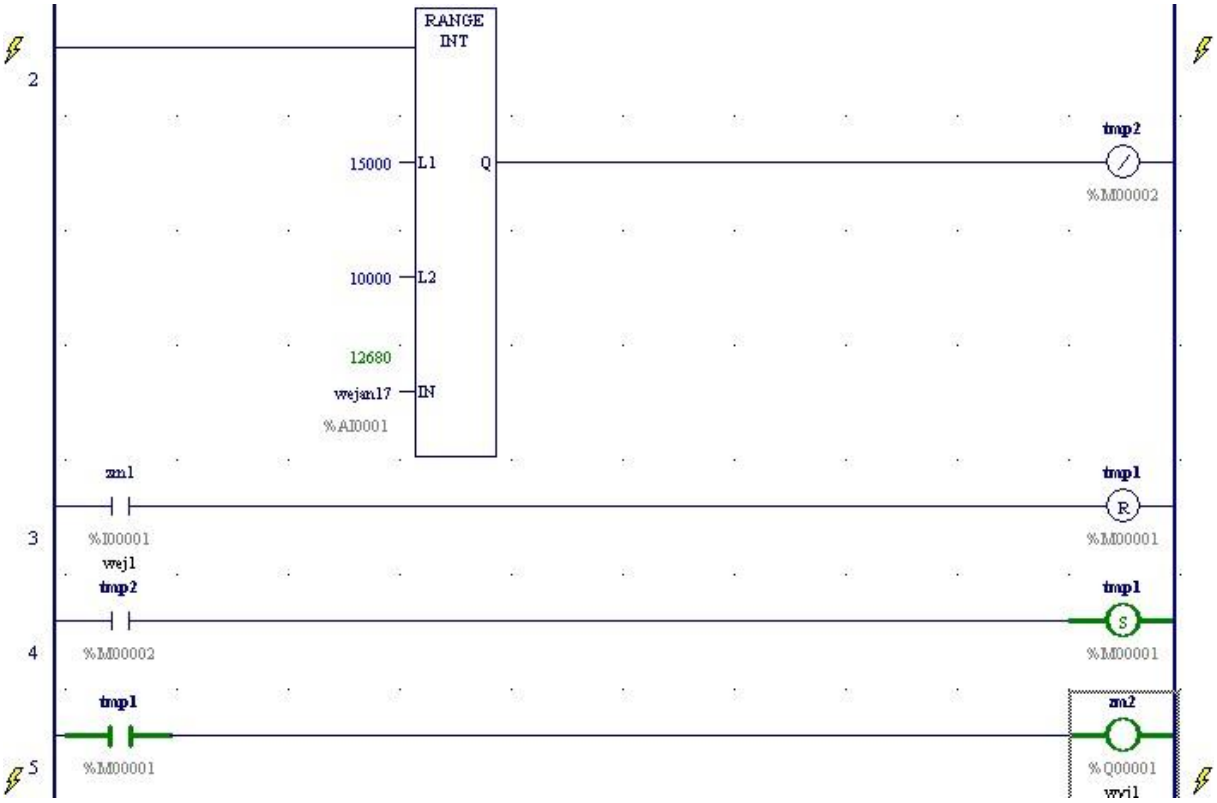
a) wartość znajduje się w bezpiecznym przedziale, nie następuje sygnalizacja.



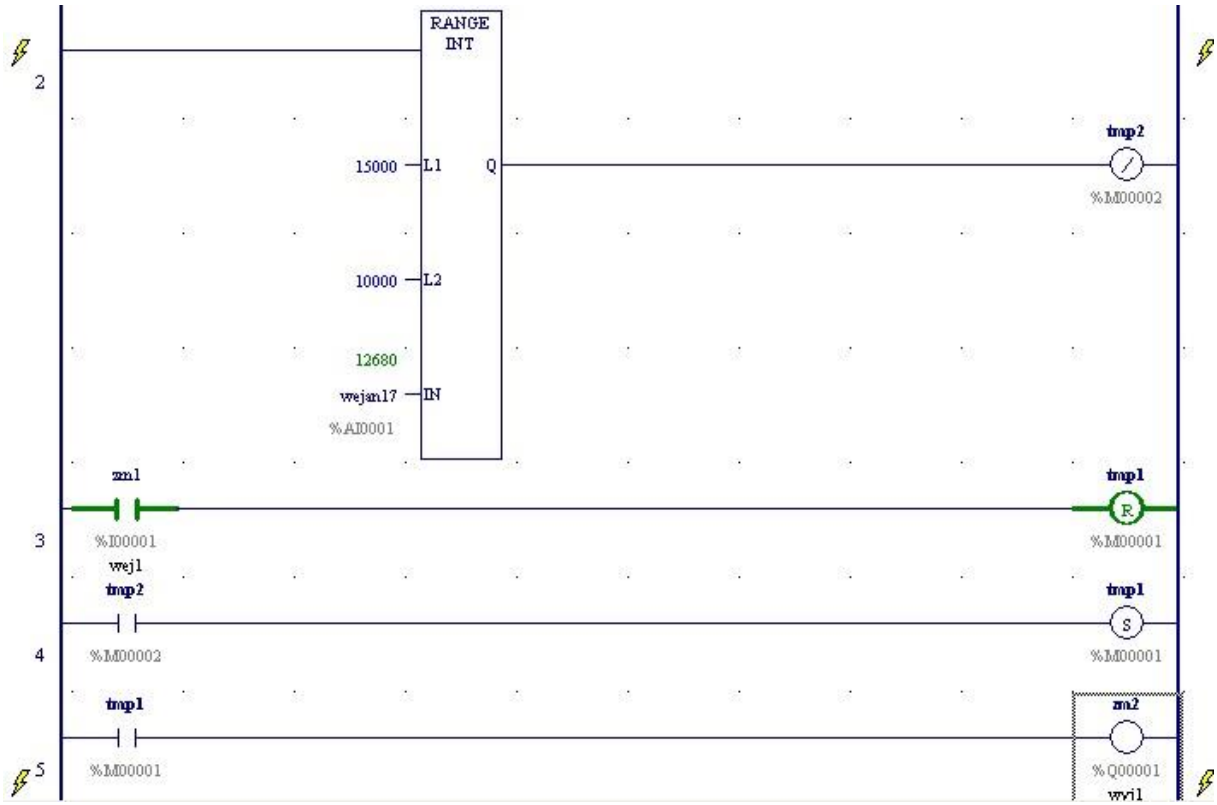
b) wartość jest zbyt duża, uruchamia się alarm stworzony w punkcie 1 (na czas trwania przekroczenia) oraz alarm stworzony w niniejszym punkcie

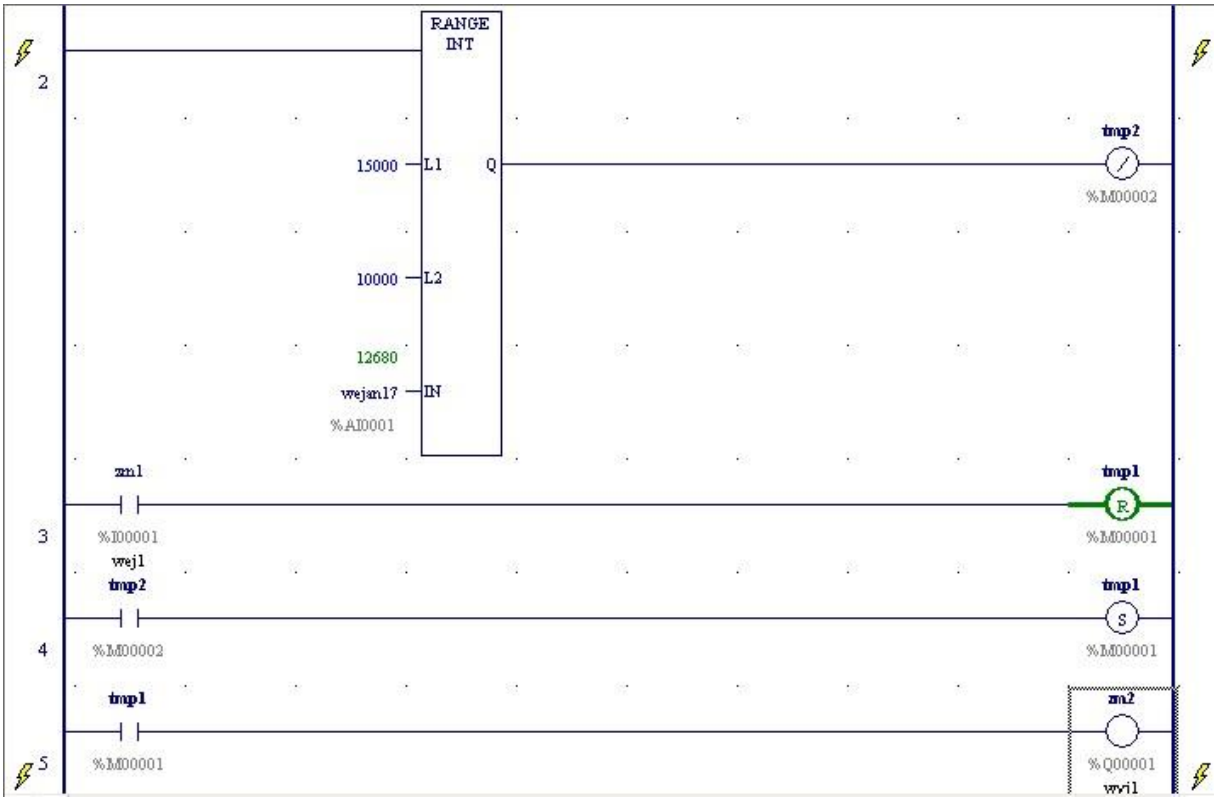


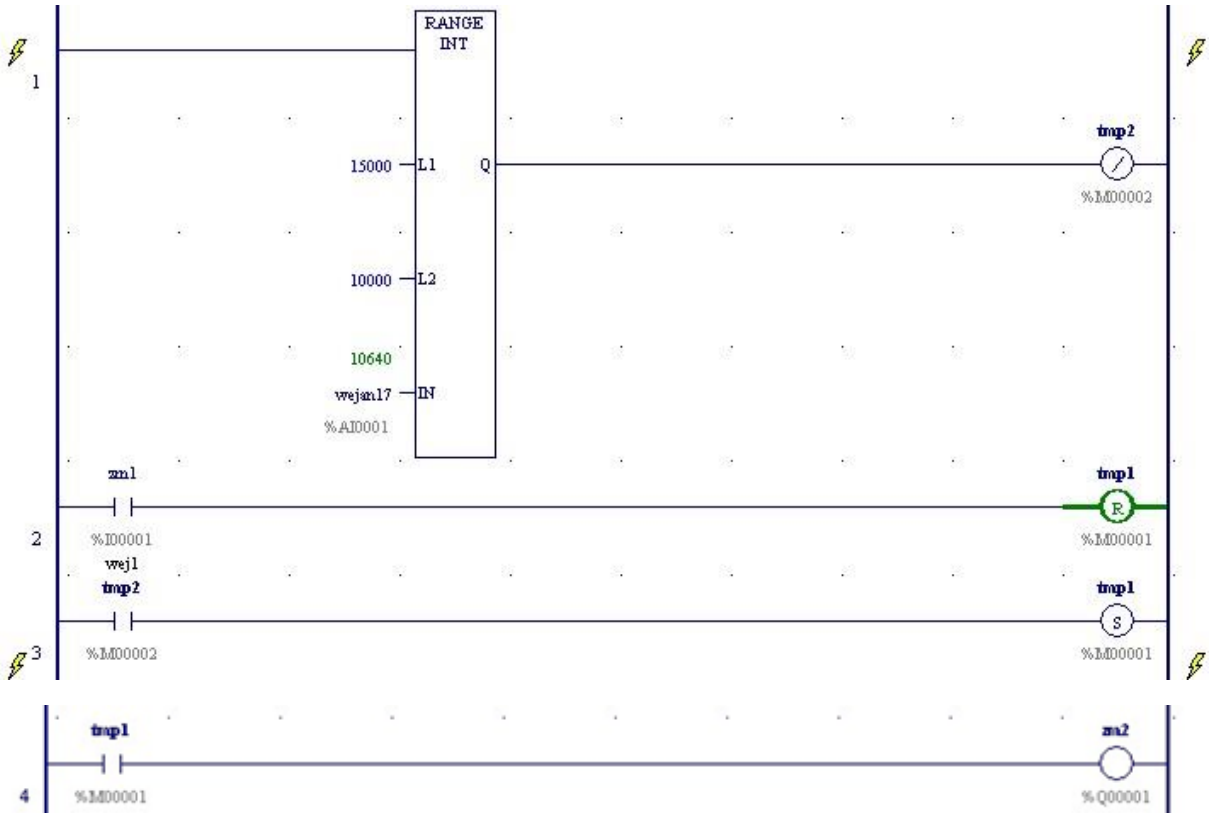
c) wartość znów znajduje się w bezpiecznym przedziale. Alarm działający jedynie w czasie wyjścia poza bezpieczny zakres przestał działać. Drugi alarm wymaga skasowania przez użytkownika



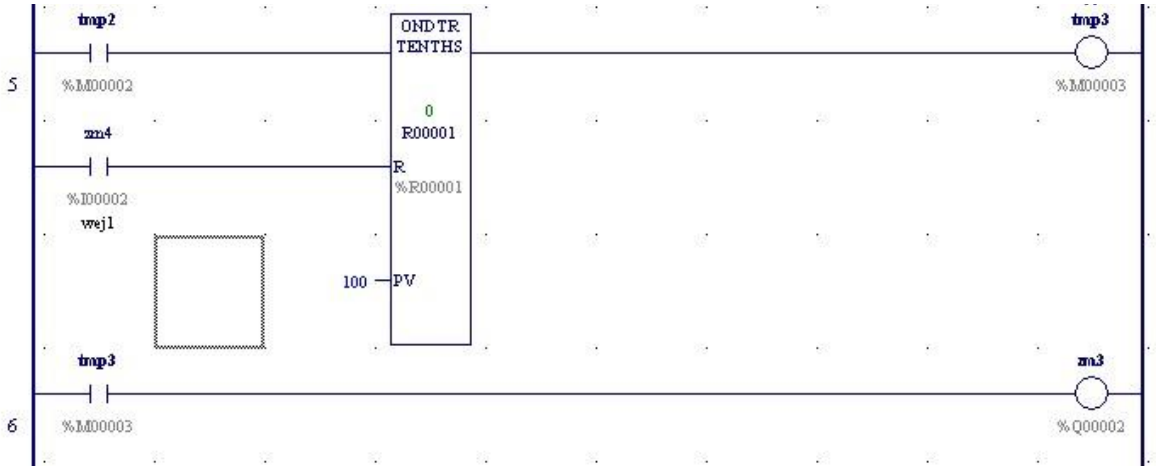
d) alarm jest kasowany przez użytkownika







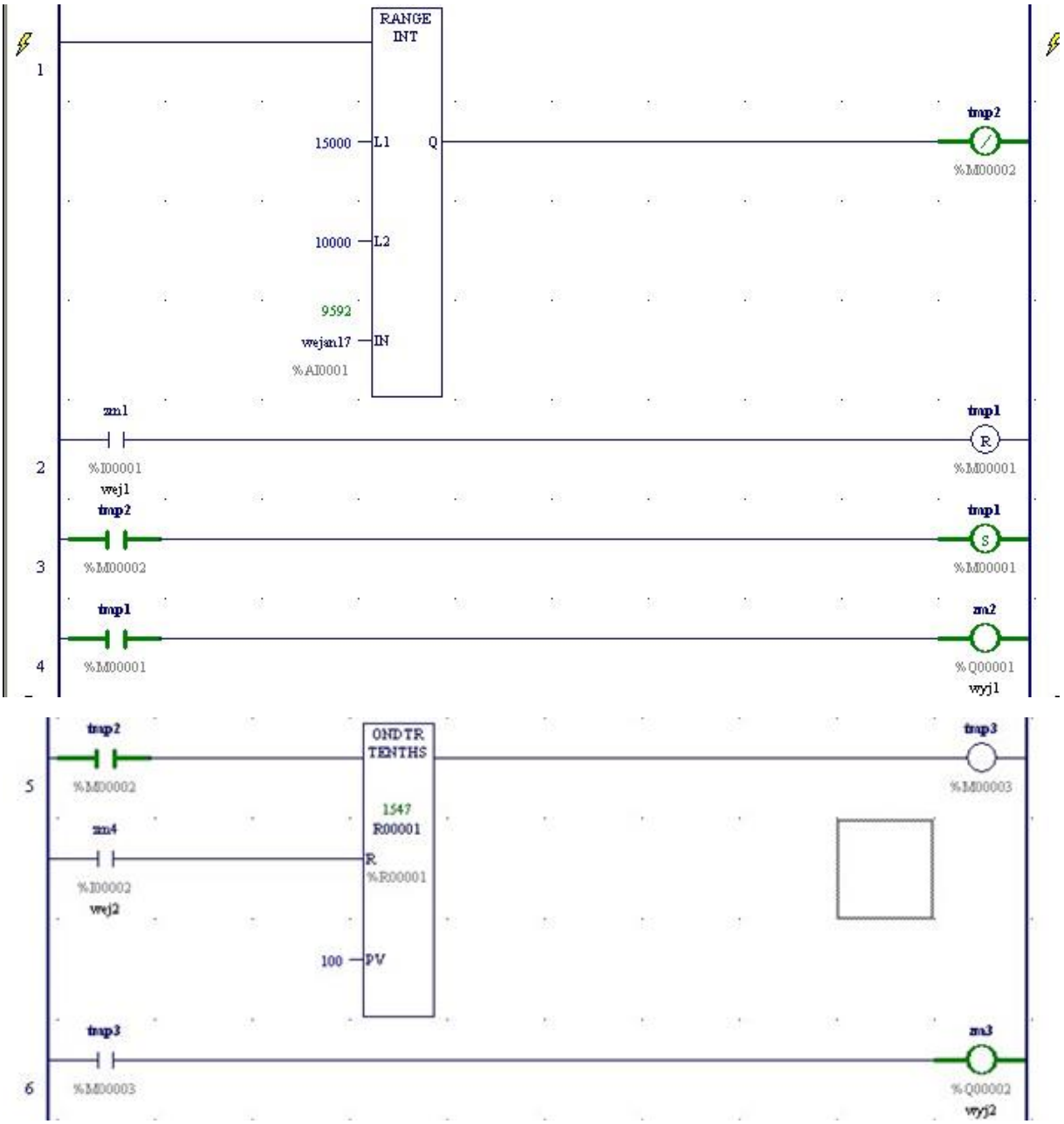
3. Dodano kolejny alarm, który był uruchamiany, gdy zmienna była poza zakresem przez dłużej niż 10 sekund. Po włączeniu alarmu, wymagał on skasowania przez użytkownika fizycznym przełącznikiem. Poniższy screen zawiera fragment logiki zawierający tą funkcjonalność.

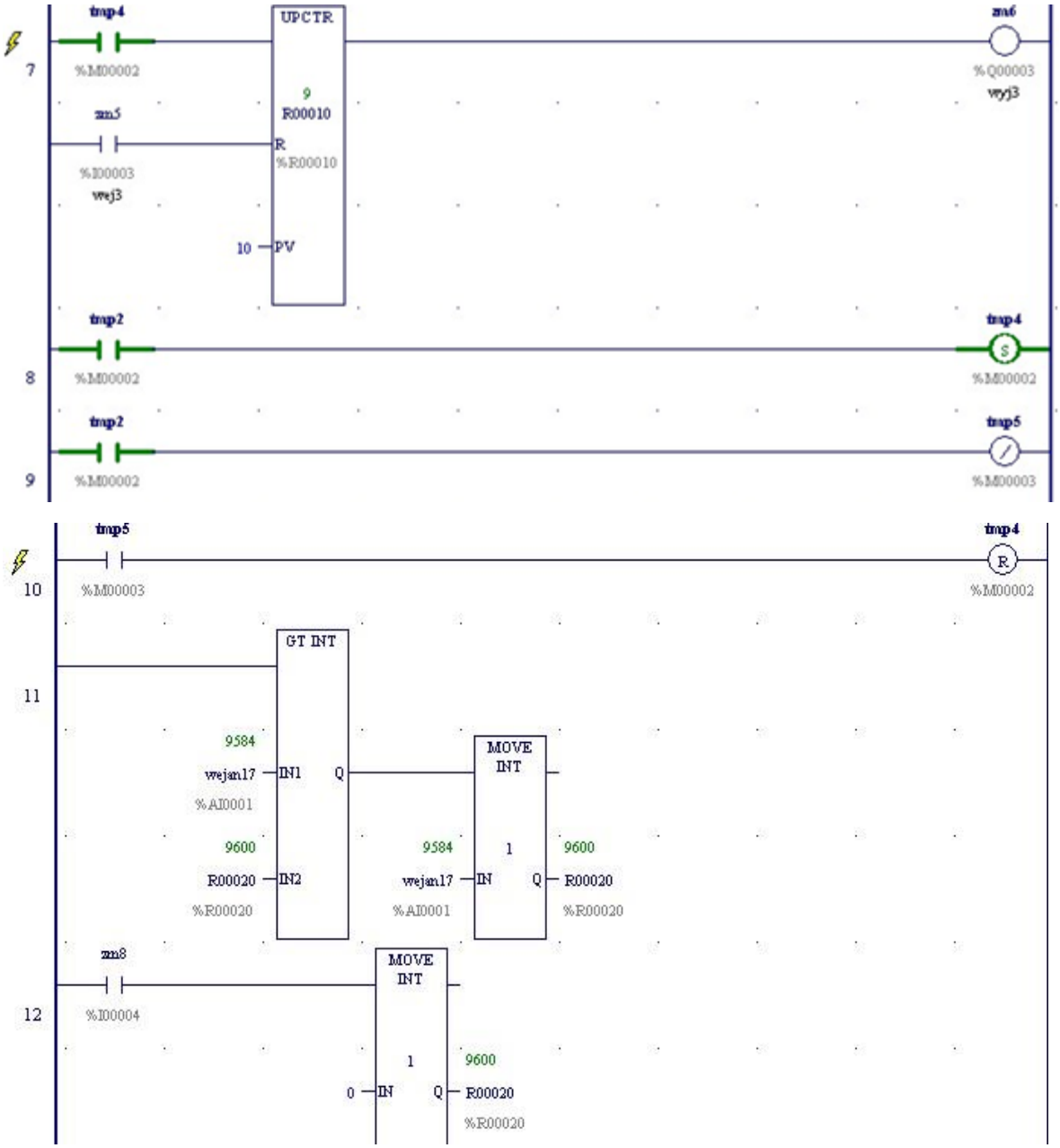


4. Do programu dodano kolejną funkcjonalność, która po dziesięciu włączeniach alarmu miała uruchomić kolejny znacznik alarmowy. Wyłączanie (i zerowanie licznika) było realizowane przez przełączenie przełącznika.

5. Dodano funkcjonalność zapisywania najwyższej wartości sygnału wejściowego była zapisywana do rejestru sterownika. Użytkownik za pomocą przełącznika miał możliwość wyzerowania rejestru.

Program, po dodaniu wszystkich powyższych funkcjonalności





Wnioski:
Programowanie w języku drabinkowym początkowo może wydawać się nieintuicyjne, jednakże po pokonaniu pierwszych trudności początkujący programista jest w stanie korzystać z kolejnych funkcjonalności języka, po zapoznaniu się z dokumentacją, bez większych problemów.

Sterowniki PLC dostarczają ogromne możliwości. Modułowa budowa sprawia, że nie musimy posiadać zbędnych elementów, z których nie korzystamy, a jedynie te, które będą wykorzystywane w projekcie.
Sterowniki te można stosować do np. realizacji procesów produkcyjnych w fabrykach lub automatyce domowej.