



Instrukcja obsługi oprogramowania Mitsubishi Electric

Tworzenie i symulacja programów przy użyciu GX Works 3,
GT Designer 3 oraz GT Simulator 3

Spis treści

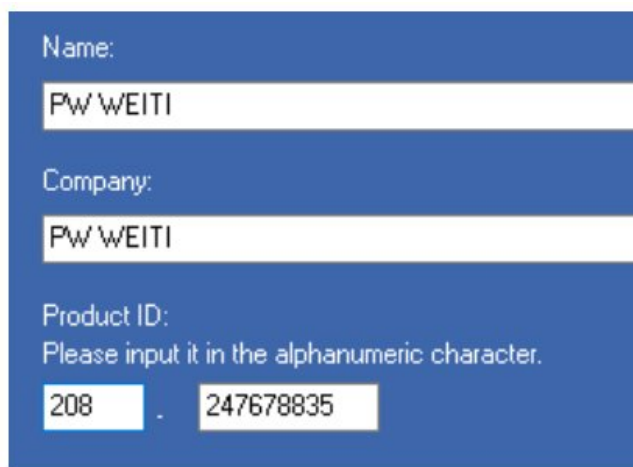
1	Instalacja oprogramowania	2
2	Oprogramowanie GX Works 3	3
2.1	Tworzenie nowego projektu	3
2.2	Zapisanie projektu	4
2.3	Okna projektu	5
2.4	Rodzaje dostępnych programów	7
2.5	Mapa pamięci	10
2.6	Etykiety	12
2.7	Okno Element Selection	14
2.8	Język drabinkowy (Ladder)	17
2.9	Symulacja i monitorowanie programu	19
2.10	E-Manual Viewer	22
3	Oprogramowanie GT Designer 3	23
3.1	Tworzenie nowego projektu	23
3.2	Zapisanie projektu	26
3.3	Okna projektu	26
3.4	Dodawanie obrazów	29
3.5	Projektowanie obrazów	30
3.6	Biblioteka	34
3.7	Symulacja	34
3.8	E-Manual Viewer	35
4	Oprogramowanie GT Simulator 3	36
4.1	Wybór symulatora	36
4.2	Symulacja	37

1. Instalacja oprogramowania

W niniejszej instrukcji następujące oprogramowanie zostanie omówione:

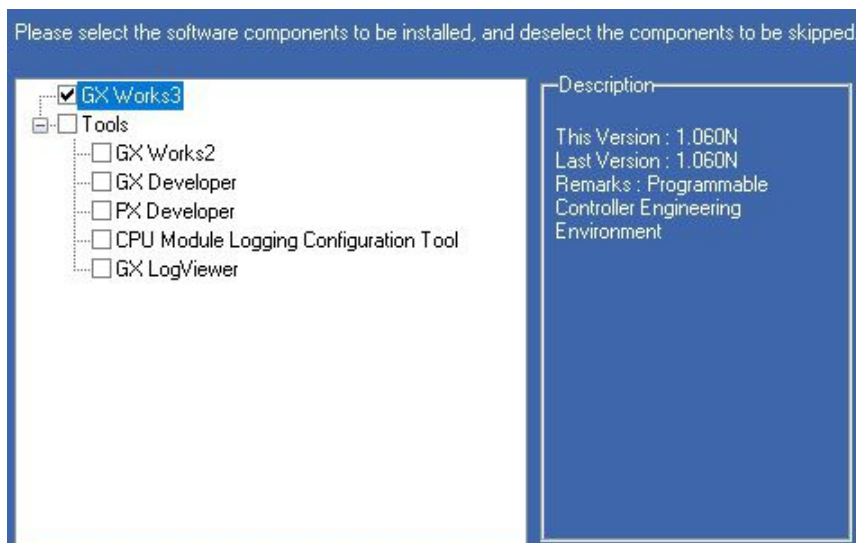
- GX Works 3 – oprogramowanie służące do tworzenia programu na sterownik PLC
- GT Designer 3 – oprogramowanie służące do tworzenia wizualizacji programu
- GT Simulator 3 – oprogramowanie służące do symulacji programu przy pomocy uprzednio utworzonej wizualizacji

Oprogramowanie należy pobrać z adresu: https://1drv.ms/f/s!AhrOuGQMt_7qhEJzn0Hd10uvJahv a następnie zainstalować. Instalację danego narzędzia przeprowadza się poprzez uruchomienie plików autorun.exe, a następnie wybranie nazwy narzędzia z listy. Instalator jest wspólny dla oprogramowania GT Designer 3 i GT Simulator 3. Po wybraniu nazwy narzędzia do zainstalowania należy przejść poszczególne kroki instalacji akceptując wszelkie zapytania. W momencie podania licencji należy wpisać licencję jak na rys.1.



Rysunek 1: Informacje o licencji

W przypadku instalacji GX Works 3 po wpisaniu licencji pojawi się lista z wyborem oprogramowania do zainstalowania. Zalecane jest nieinstalowanie dodatkowego oprogramowania poza GX Works 3, gdyż może spowodować to trudności w uruchomieniu symulacji napisanych programów. Na rys.2 przedstawiono właściwe zaznaczenie.



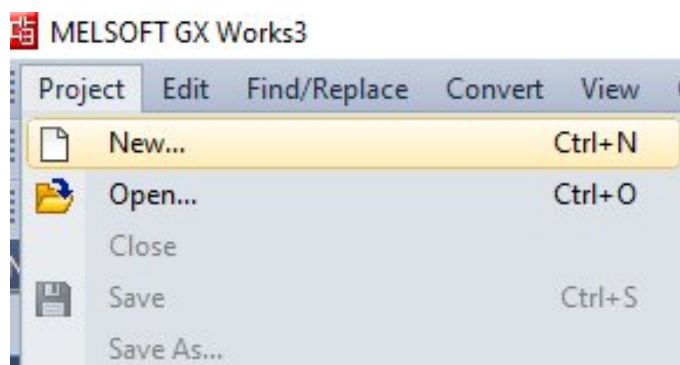
Rysunek 2: Wybór oprogramowania do instalacji

Następnie, będzie trwała pewien czas instalacja danego oprogramowania, po którym będzie ono gotowe do użytku. **Uwaga: oprogramowanie z wymienioną licencją należy odinstalować po zakończeniu semestru 2020L. Licencja przeznaczona jest tylko do użytku w ramach realizacji laboratoriów na Wydziale Elektroniki i Technik Informatycznych Politechniki Warszawskiej.**

2. Oprogramowanie GX Works 3

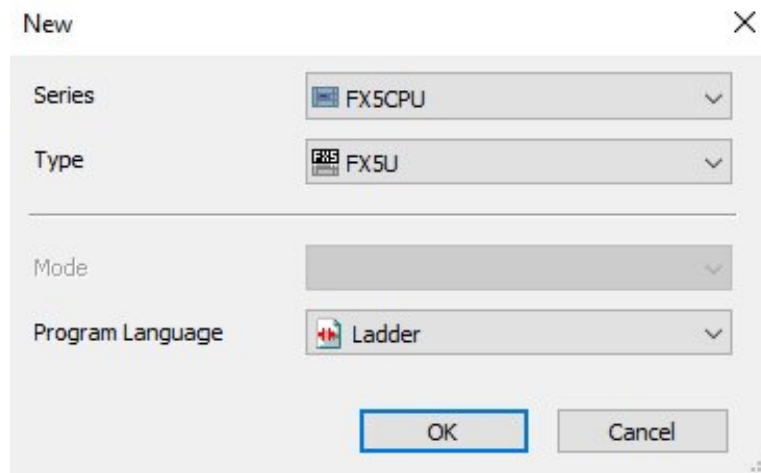
2.1. Tworzenie nowego projektu

Tworzenie nowego projektu w oprogramowaniu GX Works 3 zaczynamy od jego uruchomienia, a następnie wybrania opcji Project — > New jak pokazano na rys. 3.



Rysunek 3: Tworzenie nowego projektu

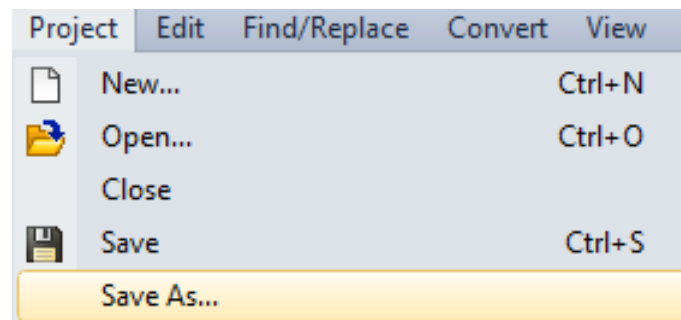
Wyświetli się okno z wyborem serii oraz typu sterownika jak i wybór języka programu. Wybrać należy sterownik FX5U oraz język drabinkowy jak na rys. 4.



Rysunek 4: Ustawienia projektu

2.2. Zapisanie projektu

Nowo utworzony projekt zapisujemy poprzez wybranie opcji Project —> Save As jak pokazano na rys. 5.



Rysunek 5: Zapisanie projektu

Kolejne zapisy można dokonywać skrótem klawiszowym Ctrl+S lub wybranie opcji Save znajdującej się nad opcją Save As. Zaleca się niestosowanie żadnych znaków specjalnych w nazwach projektów, bo może to uniemożliwić poprawny ich zapis.

2.3. Okna projektu

Projekt składa się z narzędzi i okien pozwalających tworzyć proste oraz również wyjątkowo złożone programy na sterownik PLC. Na początku zostaną omówione paski z narzędziami w górnej części projektu. Na rys. 6 ta część została przedstawiona.



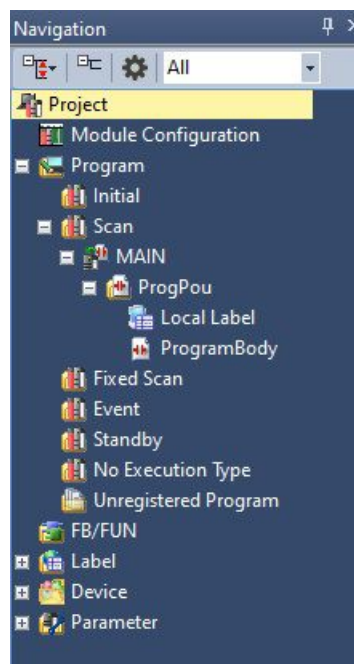
Rysunek 6: Paski narzędzi

Występują następujące rozwijane listy:

1. Project, gdzie występują m.in. opcje tworzenia, otwierania, zapisu czy eksportu biblioteki projektu.
2. Edit, gdzie występują m.in. opcje edycji projektu jak dodawanie wierszy, kolumn, elementów języka drabinkowego do okna głównego edycji programu czy też import/eksport plików.
3. Find/Replace, gdzie występują m.in. opcje znalezienia oraz zamiany istniejących elementów utworzonego programu na inne.
4. Convert, gdzie występują m.in. opcje przebudowy części lub całego projektu w celu przygotowania programu do symulacji.
5. View, gdzie występują m.in. opcje pokazania czy ukrycia poszczególnych okien projektu.
6. Online, gdzie występują m.in. opcje wgrania programu na sterownik PLC.
7. Debug, gdzie występują m.in. opcje symulacji programu bez wgrywania na sterownik PLC.
8. Diagnostics, gdzie występują m.in. opcje diagnostyki połączenia między oprogramowaniem a sterownikiem PLC.
9. Tool, gdzie występują m.in. opcje wgrania i odczytu parametrów z karty pamięci.
10. Window, gdzie występują m.in. opcje rozlokowania poszczególnych okien projektu.

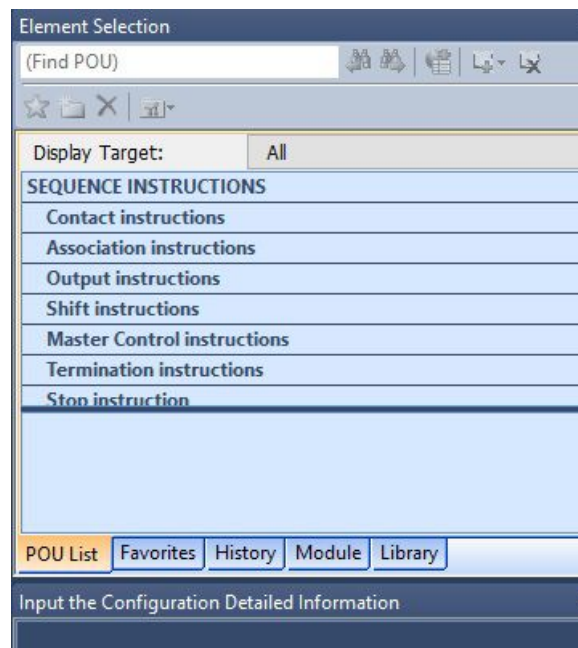
11. Help, gdzie można uzyskać informacje o oprogramowaniu oraz o wszystkich jego możliwościach.

Poniżej tych list znajdują się ikony umożliwiające szybki dostęp do popularnie wybieranych opcji z list. W pierwszym wierszu są głównie opcje przebudowy projektu, włączenia/zatrzymania symulacji oraz monitoringu programu. Natomiast w drugim wierszu znajdują się m.in. opcje z rozmieszczeniem okien. Paski te można pokazać lub schować w zależności od potrzeb w wcześniej wymienionej liście View. Po lewej stronie projektu znajduje się okno Navigation z drzewem naszego projektu. Okno Navigation pokazane jest na rys. 7.



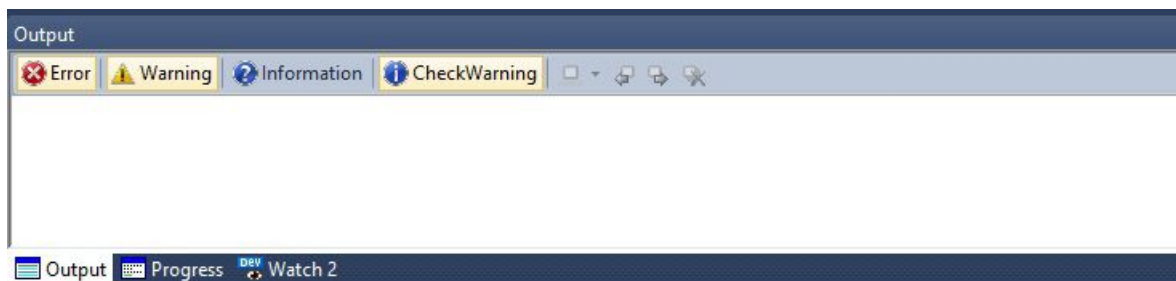
Rysunek 7: Okno Navigation

Znajdują się tam: aktualne informacje o konfiguracji modułu (gałąź Project –Module Configuration), nasze programy (gałąź Program), etykiety globalne (gałąź Label), podgląd pamięci (gałąź Device) oraz parametry systemu (gałąź Parameter). Po prawej stronie projektu znajduje się okno Element Selection, które m.in. zawiera wiele przydatnych instrukcji i funkcji do pisania programów na sterownik PLC. Okno zostało przedstawione na rys. 8.



Rysunek 8: Okno Element Selection

Funkcjonalność okna Element Selection jest rozwinięta w rozdziale 2.7. Na dole znajduje projektu znajduje się okno służące do monitoringu i weryfikacji poprawności działania napisanego programu. To okno zostało przedstawione na rys. 9.

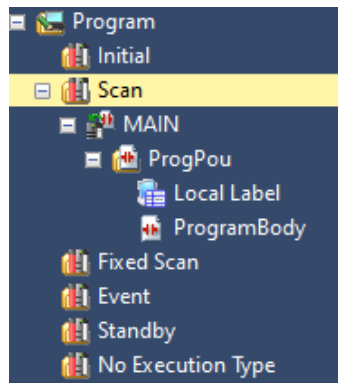


Rysunek 9: Okno informujące o aktualnym stanie programu

Pod kątem symulacji napisanego programu najważniejszymi sekcjami tego okna jest sekcja output informująca użytkownika o wykrytych błędach w programie oraz sekcja watch umożliwiająca kontrolę poszczególnych adresów pamięci sterownika.

2.4. Rodzaje dostępnych programów

W gałęzi projektu Program znajduje się lista możliwych do zaimplementowania rodzajów programów sterujących. Na rys. 10 przedstawiona została ta lista w drzewie projektu.

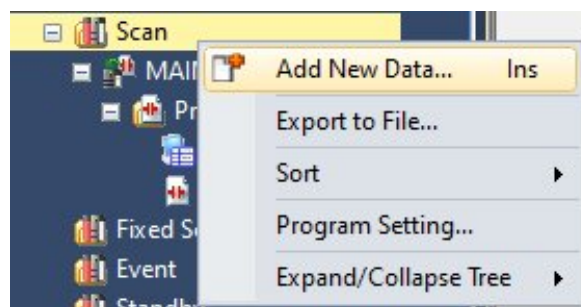


Rysunek 10: Rodzaje programów

Podział programów jest następujący:

1. Initial - program wykonywany tylko w pierwszym cyklu sterownika.
2. Scan - główny skan procesora, cyliczne wykonywanie programu z czasem cyklu zależnym od obciążenia procesora.
3. Fixed scan - skan z narzuconym okresem wykonania (czas konfigurowalny).
4. Event - obsługa zdarzeń.
5. No execution Type - magazyn kodu, programy tu przeniesione nie będą wykonywane.

Dodanie programu obojętnie którego rodzaju odbywa się poprzez wybranie opcji Add New Data jak pokazano na rys. 11.



Rysunek 11: Dodanie nowego programu

Następnie w otworzonym oknie należy przejrzeć i zaakceptować lub zmienić ustawienia nowego programu. Przykład okna pokazano na rys. 12.

New Data

Basic Setting

Data Type: Program Block

(Data Name): ProgPou1

Detail Setting

Program Configuration

Program Language: Ladder

Program file

Execution type: Scan

Program file for add destination: MAIN

OK Cancel

Rysunek 12: Ustawienia nowego programu

Ustawienie czasu skanu programu czy też zmianę ustawień poszczególnych przerwań można dokonać poprzez wejście z drzewa projektu do gałęzi FX5UCPU — > CPU Parameter. Na rys. 13 pokazano okno z możliwością obsługi przerwań.

Setting Item List

Input the Setting Item to Search

- Name Setting
- Operation Related Setting
 - Interrupt Settings
 - Fixed Interval Setting
 - Fixed Scan Execution Mode Setting
 - Interrupt Priority Setting from Module
- Service Processing Setting
- File Setting
- Memory/Device Setting
- RAS Setting
 - Scan Time Monitoring Time (WDT) Setting
 - Constant Scan Setting
 - Error Detections Setting
 - CPU Module Operation Setting at Error Detected

Item List Find Result

Setting Item

Item	Setting
Fixed Scan Interval Setting	
Interrupt Setting from Internal Timer	
I28	100 ms
I29	40 ms
I30	20 ms
I31	10 ms
Fixed Scan Execution Mode Setting	
Fixed Scan Execution Mode	Precede Fixed Scan
Priority Setting for Interrupt from Module	
Multiple Interruptions	Disable
Interrupt Priority	<Detailed Setting>
Index Register. Save/Restore Setting	Not Saved/Recovered

Explanation

Please set when change the interrupt fixed scan interval from the internal timer.

Check Restore the Default Settings

Apply

Rysunek 13: Obsługa przerwań

2.5. Mapa pamięci

Mapa pamięci sterownika złożona jest z wielu adresów, które to wyszczególniono na rys. 14. Najistotniejsze zostały zaznaczone na czerwone, są to:

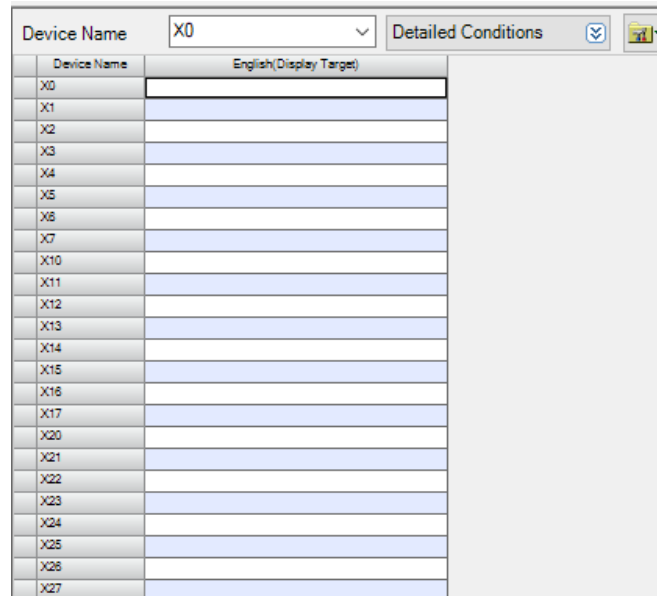
1. X - wejścia typu bit.
2. Y - wyjścia typu bit.
3. M - flagi (pomocnicze styki użytkownika) typu bit.
4. L - flagi typu bit nie tracące swojego stanu po zaniku zasilania.
5. T - odmierzacze czasu (timery) typu bit lub word (16bit).
6. D - rejestry typu word.
7. SM - flagi specjalne typu bit.
8. SD - rejestry specjalne typu word.

Przykładowo, adresowanie poszczególnych bitów wejść jest następujące: X0,X1,...X7; X10,X11,...,X17;... i analogiczne dla innych adresów.

Division	Type	Device name	Symbol	Notation
User device	Bit	Input	X	Octal
	Bit	Output	Y	Octal
	Bit	Internal relay	M	Decimal
	Bit	Latch relay	L	Decimal
	Bit	Link relay	B	Hexadecimal number
	Bit	Annunciator	F	Decimal
	Bit	Link special relay	SB	Hexadecimal number
	Bit	Step relay	S	Decimal
	Bit/word	Timer	T (Contact: TS, Coil: TC, Current value: TN)	Decimal
	Bit/word	Retentive timer	ST (Contact: STS, Coil: STC, Current value: STN)	Decimal
	Bit/word	Counter	C (Contact: CS, Coil: CC, Current value: CN)	Decimal
	Bit/Double word	Long counter	LC (Contact: LCS, Coil: LCC, Current value: LCN)	Decimal
	Word	Data register	D	Decimal
	Word	Link register	W	Hexadecimal number
	Word	Link special register	SW	Hexadecimal number
System device	Bit	Special relay	SM	Decimal
	Word	Special register	SD	Decimal

Rysunek 14: Mapa pamięci

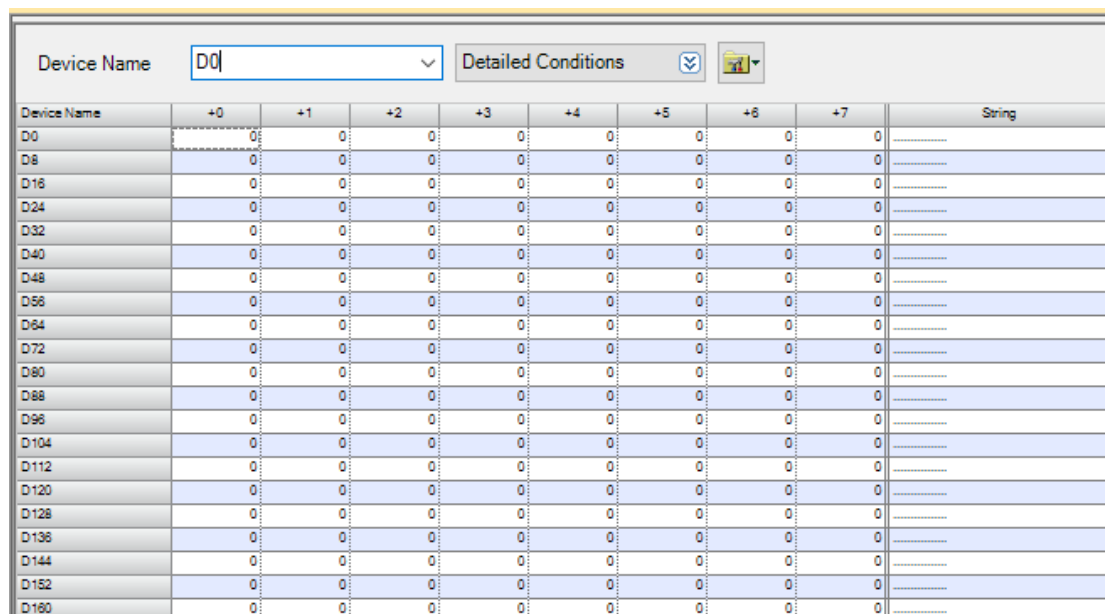
Poprzez wejście z drzewa projektu do gałęzi Device—>Device Comment—>Common Device Comment można podejrzeć dostępne adresy. Na rys. 15 przedstawiono podgląd adresów wejściowych.



Device Name	English(Display Target)
X0	
X1	
X2	
X3	
X4	
X5	
X6	
X7	
X10	
X11	
X12	
X13	
X14	
X15	
X16	
X17	
X20	
X21	
X22	
X23	
X24	
X25	
X26	
X27	

Rysunek 15: Podgląd dostępnych adresów

Natomiast poprzez przejście z drzewa projektu do gałęzi Device—>Device Memory—>Main można podejrzeć zawartość poszczególnych bitów rejestrów czy innych adresów jak pokazano na rys. 16.



Device Name	+0	+1	+2	+3	+4	+5	+6	+7	String
D0	0	0	0	0	0	0	0	0
D8	0	0	0	0	0	0	0	0
D16	0	0	0	0	0	0	0	0
D24	0	0	0	0	0	0	0	0
D32	0	0	0	0	0	0	0	0
D40	0	0	0	0	0	0	0	0
D48	0	0	0	0	0	0	0	0
D56	0	0	0	0	0	0	0	0
D64	0	0	0	0	0	0	0	0
D72	0	0	0	0	0	0	0	0
D80	0	0	0	0	0	0	0	0
D88	0	0	0	0	0	0	0	0
D96	0	0	0	0	0	0	0	0
D104	0	0	0	0	0	0	0	0
D112	0	0	0	0	0	0	0	0
D120	0	0	0	0	0	0	0	0
D128	0	0	0	0	0	0	0	0
D136	0	0	0	0	0	0	0	0
D144	0	0	0	0	0	0	0	0
D152	0	0	0	0	0	0	0	0
D160	0	0	0	0	0	0	0	0

Rysunek 16: Podgląd rejestrów

Na rys.17 przedstawiono dostępne flagi specjalne. Każda flaga specjalna posiada specjalną funkcję różniącą ją od zwykłej flagi, na przykład flaga SM402 załącza tylko podczas pierwszego cyklu programu.

Special relay	Name
SM400, SM8000	Always ON
SM401, SM8001	Always OFF
SM402, SM8002	After RUN, ON for one scan only
SM403, SM8003	After RUN, OFF for one scan only
SM409, SM8011	0.01 second clock
SM410, SM8012	0.1 second clock
SM411	0.2 second clock
SM412, SM8013	1 second clock
SM413	2 second clock
SM414	2n second clock
SM415	2n ms clock

Rysunek 17: Flagi specjalne

2.6. Etykiety

Etykiety są to pomocnicze nazwy elementów programu, które użytkownik może nadać w celu większej przejrzystości i kontroli nad napisanym programem. Rozróżnia się etykiety lokalne i globalne. Etykiety lokalne są wyłącznie dostępne w konkretnym programie i nie można im przydzielać konkretnych adresów. Natomiast etykiety globalne są dostępne w całym projekcie i można im przydzielać konkretne adresy z pamięci sterownika. Jeżeli został utworzony program w sekcji Scan to można otworzyć okno definiowania lokalnych etykiet dla tego programu poprzez przejście z drzewa projektu kolejno Program --> Scan --> Main --> Nazwa programu --> Local Label. Na rys. 18 przedstawiono to okno.

	Data Type	Class	Initial Value	Constant	English(Display Target)
1		...			

Rysunek 18: Okno definiowania etykiet lokalnych

Mamy tu możliwość określenia m.in. nazwy etykiety, jaki będzie miała typ danych (bit, word, double word, ...) oraz klasy (zmienna, stała,...). Przykład zdefiniowania etykiety lokalnej przedstawiono na rys. 19.

<Filter>		Easy Display <<		Display Setting		Check	
	Label Name	Data Type		Class		Initial Value	Constant
1	Wej3	Bit	...	VAR			
2			...				

Rysunek 19: Przykład zdefiniowania etykiety lokalnej

Aby zdefiniować etykiety globalne należy przejść z drzewa projektu kolejno do Label — >Global Label — >Global i otworzy się okno jak na rys. 20.

<Filter>

Easy Display <<

Display Setting

Check

	Label Name	Data Type		Class	Assign (Device/Label)	Initial Value	Constant	English(Display)	Remark
1									

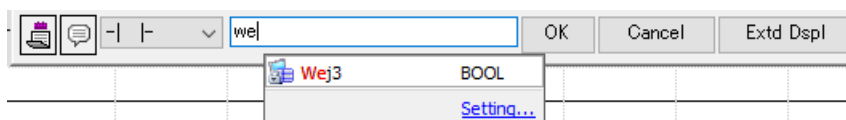
Rysunek 20: Okno definiowania etykiet globalnych

W przypadku definicji tych etykiet mamy te same możliwości określenia ich właściwości jak w przypadku etykiet lokalnych ale dodatkowo mamy również pole Assign(Device/Label) do wpisania konkretnego adresu z pamięci, któremu będzie nadana utworzona etykieta. Przykład utworzonej etykiety globalnej został przedstawiony na rys. 21.

<Filter>		Easy Display <<		Display Setting		Check	
	Label Name	Data Type		Class	Assign (Device/Label)		
1	labelX5	Bit	...	VAR_GLOBAL	X5		
2			...				

Rysunek 21: Przykład zdefiniowania etykiety globalnej

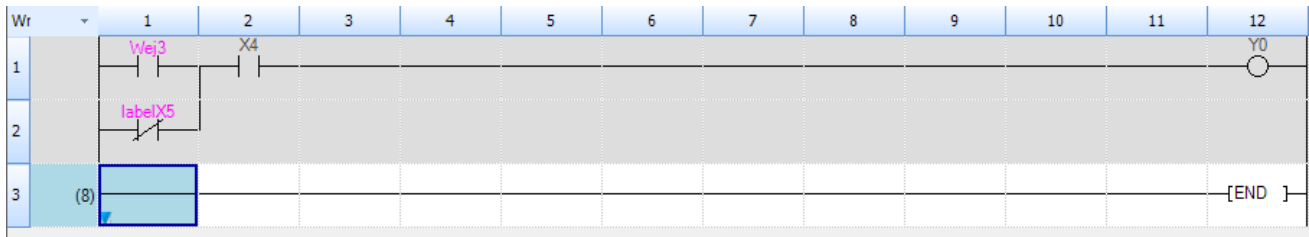
Jeśli chcemy nadać konkretnemu elementowi programu etykietę lokalną bądź globalną wystarczy kliknąć dwukrotnie w ten element i zacząć wpisywać nazwę etykiety. Po chwili powinna pojawić się na liście jak to pokazano na rys. 22.



Rysunek 22: Nadanie utworzonej etykiety elementowi w programie

Przypisanie jednemu elementowi w programie etykiety lokalnej, a drugiemu etykiety globalnej obrazuje rys. 23. Należy zwrócić uwagę, że w przeciwieństwie do styku z przypisaną etykietą globalną, gdzie styk będzie zmieniał swój stan w zależności od bitu adresu wejściowego, to

styk z przypisaną etykietą lokalną staje się stykiem pomocniczym i nie będzie uzależniony od żadnego adresu w pamięci.



Rysunek 23: Przykład programu z etykietami

2.7. Okno Element Selection

Okno Element Selection znajdujące się po prawej stronie projektu posiada wiele użytecznych instrukcji do pisania złożonych programów. Jeżeli jesteśmy w ProgramBody naszego programu to w zakładce POU List mamy długą listę instrukcji. Pierwszymi z nich są instrukcje sekwencyjne, które pokazano na rys. 24.

SEQUENCE INSTRUCTIONS
Contact instructions
Association instructions
Output instructions
Shift instructions
Master Control instructions
Termination instructions
Stop instruction

Rysunek 24: Instrukcje sekwencyjne

Są tu zarówno przydatne instrukcje łączenia styków jak i instrukcje dla wyjść np. instrukcje zliczające impulsy. Następnie znajdują się instrukcje podstawowe, które pokazano na rys. 25.

BASIC INSTRUCTIONS
Comparison Operation instructions
Arithmetic Operation instructions
Data transfer instructions
Logical Operation instructions
Data shift instructions
Bit processing instructions
Data Conversion instructions

Rysunek 25: Instrukcje podstawowe

Są tu podstawowe instrukcje logiczne, arytmetyczne czy też porównujące lub przenoszące poszczególne zawartości adresów. Kolejne na liście są instrukcje aplikacyjne, które to pokazano na rys. 26.

APPLICATION INSTRUCTIONS	
Program Branch instructions	Pulse related instructions
Program execution control instructions	Matrix input instruction
Rotation instructions	Indirect address read instructions
Data processing instructions	Timing check instructions
Structure creation instructions	Module access instructions
Data table operation instructions	Logging instructions
String processing instructions	Built-in Ethernet Function instructions
Data control instructions	Network common instruction
Clock instructions	Ethernet Module
Reading/writing data instructions	CC-Link IE Field Network Module
Real number instructions	Positioning Module
Random number instructions	Handy Instruction
Index register instructions	Dedicated Instruction(High-Speed Counter Function)
Special counter instructions	High-Speed I/O Control Instruction
Special timer instructions	Dedicated Instruction (Positioning Function)
Shortcut control instruction	External Device Communication Instruction
Ramp signal instruction	External Device I/O Instruction
	Step Ladder Instruction
	PID operation instruction
	Extended file register operation instruction
	Realtime monitor function instruction

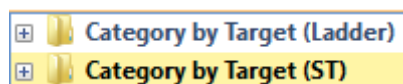
Rysunek 26: Instrukcje aplikacyjne

Są tu m.in instrukcje kontroli wykonywanego programu pod względem procesu jak i danych, odczytu oraz nadpisywania danych, specjalne instrukcje mierzące czas i impulsy czy też projektujące regulator PID. Ostatnią listą jest lista instrukcji bloków funkcyjnych, którą pokazano na rys. 27.

Standard Function/Function Block
Bit Type Boolean
Type Conversion
Arithmetic Operation
Comparison
Time Data Type
Single Numeric Variable
Selection
String
Bit Shift
Edge Detection
Counter
Timer
Bistable

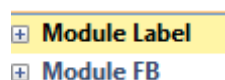
Rysunek 27: Instrukcje bloków funkcyjnych

Na tej liście są instrukcje tworzące bloki funkcyjne dla różnych operacji na bitach pamięci, operacji arytmetycznych, zliczających impulsy czy też mierzących upływający czas. W zakładce Favourites możemy dodać lub znaleźć najbardziej popularnie przez nas używane instrukcje jak pokazano na rys. 28.



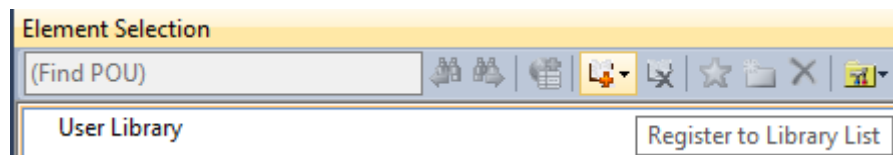
Rysunek 28: Zakładka Favourites

W zakładce Module możemy znaleźć etykiety systemowe przypisane dla naszego typu sterownika. Zakładkę przedstawiono na rys. 29.



Rysunek 29: Zakładka Module

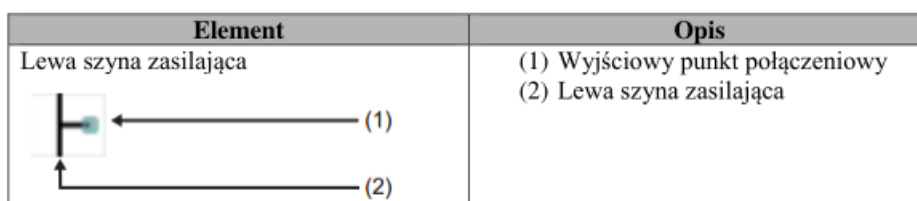
W zakładce History możemy znaleźć historię wybieranych przez nas instrukcji w ostatnich dniach użytkowania oprogramowania. Na rys. 30 przedstawiono zakładkę Library gdzie możemy zapisać lub wgrać do naszego projektu bibliotekę wykorzystanych instrukcji lub etykiet w celu szybszego tworzenia kolejnego projektu.



Rysunek 30: Zakładka Library

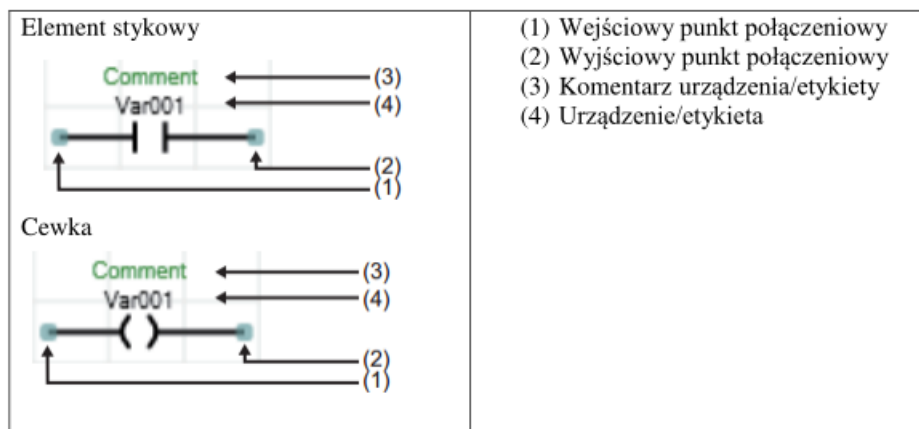
2.8. Język drabinkowy (Ladder)

Język drabinkowy (Ladder) jest popularnym językiem do pisania programów na sterownik PLC. Ideą języka jest przedstawienie graficzne połączenia między szyną zasilającą znajdującą się po lewej stronie w oknie edycji programu a drugą szyną znajdującą się naprzeciwko po prawej stronie. W zależności od przyjętych warunków połączenia można w ten sposób tworzyć proste jak i złożone programy sterujące. Na rys. 31 przedstawiono część szyny zasilającej.



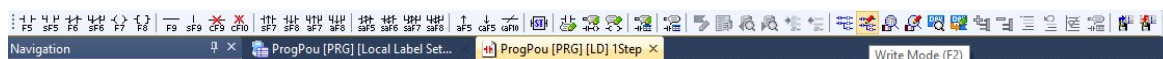
Rysunek 31: Szyna zasilająca

Styki są elementami programu, których stan definiuje czy istnieje połączenie między jedną szyną a drugą. Styk normalnie otwarty łączy szyny w przypadku pojawienia się na nim "1" logicznej, a styk normalnie zamknięty w przypadku pojawienia się na nim "0" logicznego. Cewki reprezentują dalsze wyprowadzenie sygnału jeśli spełniony jest warunek połączenia szyn. Na rys. 32 przedstawiono przykładowy element stykowy oraz cewkę. Są to podstawowe elementy służące do pisania programu typu Ladder. Natomiast istnieją elementy, które to posiadają specjalne funkcje, a uzależnione są często od przypisanych adresów z pamięci sterownika.



Rysunek 32: Element stykowy i cewka

Poprzez wejście do ProgramBody utworzonego programu, można edytować jego zawartość. Nad oknem do edycji programu znajduje się pasek z różnymi elementami umożliwiającymi tworzenie różnych programów drabinkowych. Można je wprowadzić do programu zarówno przeciągając do okna lub klikając na dowolną gałąź, a następnie stosując odpowiedni skrót klawiszowy. Skróty klawiszowe są napisane pod ikonami danych elementów na pasku z elementami. Należy pamiętać, że aby móc edytować program, trzeba mieć aktywną ikonę Write Mode znajdującą się na pasku elementów na rys. 33



Rysunek 33: Ustawienie trybu edycji programu

Dodawanie poszczególnych wierszy programu jest możliwe poprzez zaznaczenie istniejącego wiersza w programie, następnie kliknięcie go prawym przyciskiem i wybranie opcji Edit —> Insert Row. Również można to zrobić z rozwijanej listy Edit. Kasowanie wierszy i inne operacje związane z edycją wyglądu programu są możliwe w analogiczny sposób. Prosty przykładowy program przedstawiono na rys. 34.

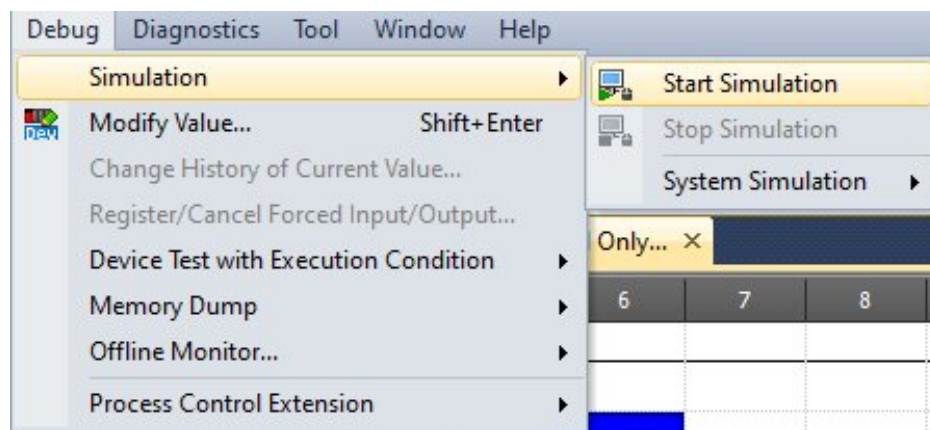
ProgPou [PRG] [Local Label Set...		ProgPou [PRG] [LD] (Read Only... X											
Read Mnt		1	2	3	4	5	6	7	8	9	10	11	12
1	(0)	X0	X16										Y0
2	(6)												[END]

Rysunek 34: Przykładowy program do symulacji

Wyjście Y0 będzie aktywne (pojawi się na nim "1" logiczna) pod warunkiem, że aktywne będzie wejście X0 (pojawi się na nim "1" logiczna) oraz nieaktywne wejście X16 (pojawi się na nim "0" logiczne). Aby zaktualizować projekt oraz sprawdzić program pod względem błędów należy kliknąć ikonę Rebuild All. Opcja ta jest również dostępna poprzez przejście z paska zadań Convert --> Rebuild All. W dolnym oknie pojawi się w zakładce Output informacja czy przebudowanie przebiegło pomyślnie lub czy wykryto błędy w programie.

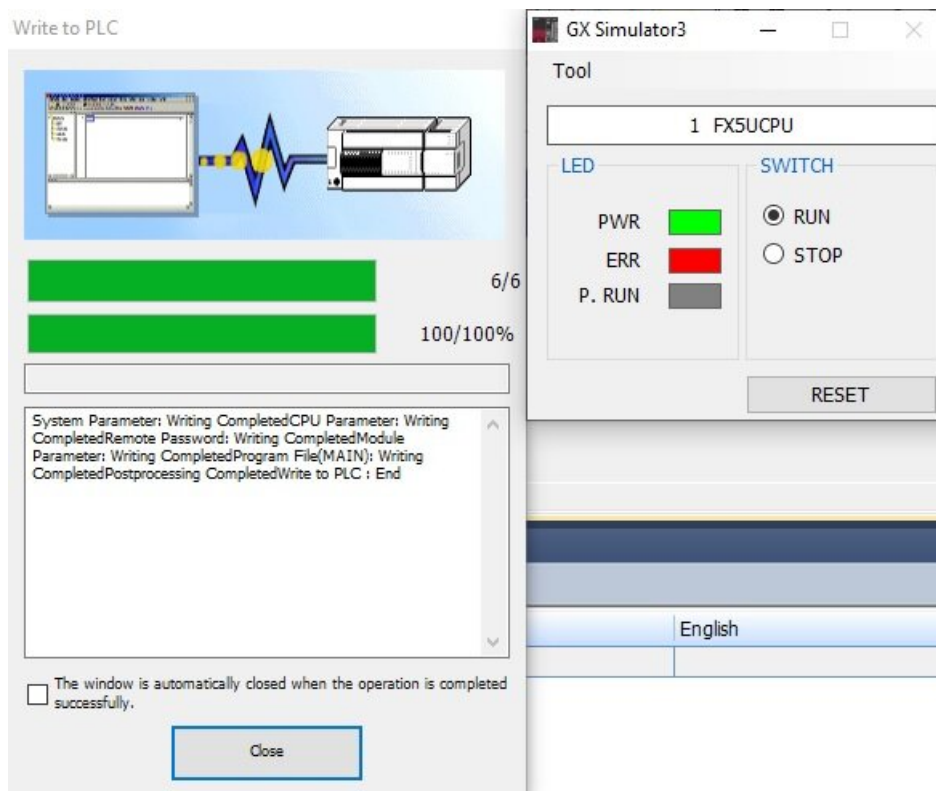
2.9. Symulacja i monitorowanie programu

Włączenie trybu symulacji napisanego programu następuje po wybraniu odpowiedniej opcji z listy Debug projektu jak to pokazano na rys. 35. Należy pamiętać, że uruchomienie symulacji będzie niemożliwe jeśli cały projekt nie został wcześniej przebudowany.



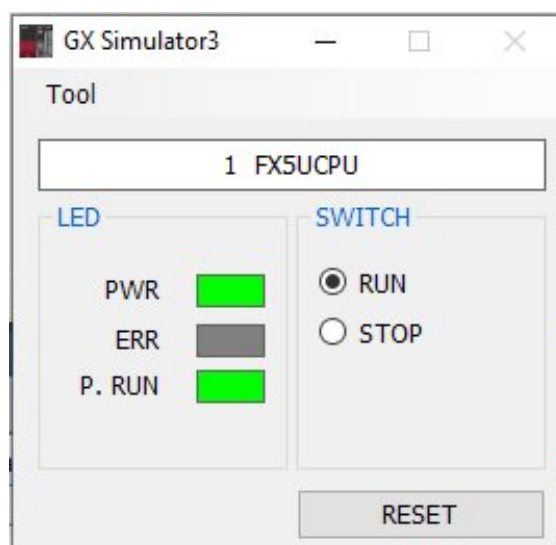
Rysunek 35: Uruchomienie symulacji w GXWorks3

Po kliknięciu wyświetla się okno z aktualnym stanem wgrywania programu do symulacji. Podczas tego momentu okno stanu symulacji GX Simulatora 3 posiada migającą czerwoną lampkę błędu jak na rys. 36.



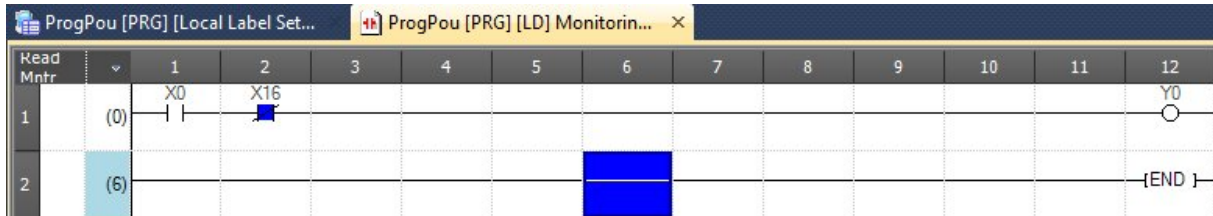
Rysunek 36: Wgrywania programu do symulacji

Należy zamknąć okno wgrywania programu gdy zostanie on w pełni wgrany co jest sygnalizowane wskaźnikiem. Gdy okno zostanie zamknięte to czerwona sygnalizacja błędu zniknie i zapali się zielona lampka P.Run sygnalizująca, że jest teraz aktywny tryb ciągłej symulacji programu jak pokazano na rys. 37.



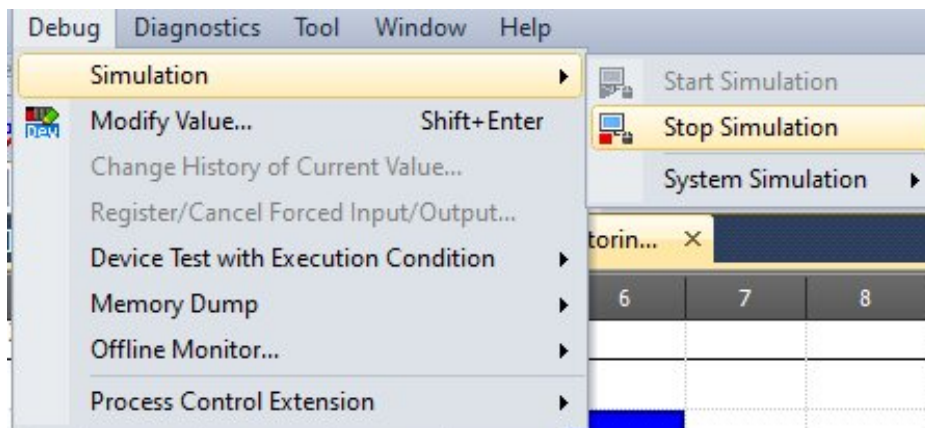
Rysunek 37: Włączony tryb symulacji programu

W trybie symulacji można obserwować aktywność lub brak aktywności elementów programu. Aktywność, sygnalizowana jest niebieskim podświetleniem danego elementu jak pokazano na rys. 38.



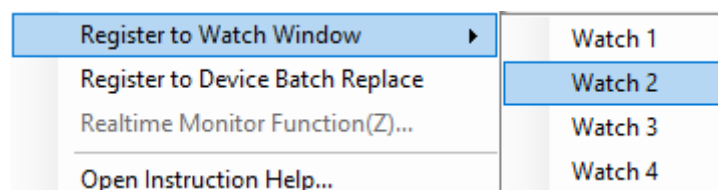
Rysunek 38: Aktywność elementu programu

W przedstawionym przykładzie styk normalnie zamknięty jest aktywny, bo bit X16 ma wartość 0. Symulację można przerwać poprzez kliknięcie Stop Simulation w sekcji Debug jak pokazano na rys. 39.



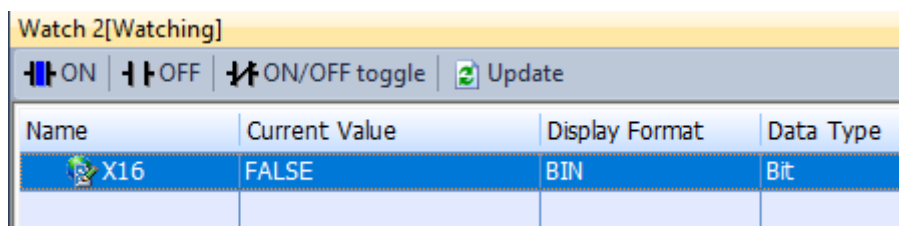
Rysunek 39: Zatrzymanie symulacji

Monitorować działanie programu można również przy użyciu zakładki Watch dolnego okna projektu. Kliknięcie na dany element programu z przypisanym adresem i wybranie opcji Register to Watch jak na rys. 40 spowoduje dodanie tego adresu do okna Watch. Również można w tym oknie wpisać adres ręcznie i monitorować jego zawartość.



Rysunek 40: Rejestracja adresu do okna Watch

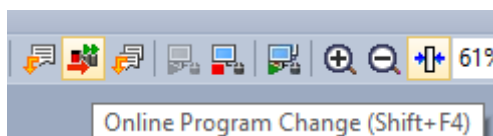
Podgląd adresu X16 przedstawiono na rys. 41. Również przy pomocy górnych opcji można zmieniać stan bitów tego adresu o ile nasz program na takie działanie zezwala.



Name	Current Value	Display Format	Data Type
X16	FALSE	BIN	Bit

Rysunek 41: Podgląd zawartości adresu w oknie Watch

Będąc w trybie symulacji możemy dokonywać drobnych modyfikacji naszego programu bez zatrzymywania tego trybu. Po dokonaniu modyfikacji w programie należy wybrać opcję Online Program Change jak pokazano na rys. 42 i po chwili zmiany zostaną wprowadzone.

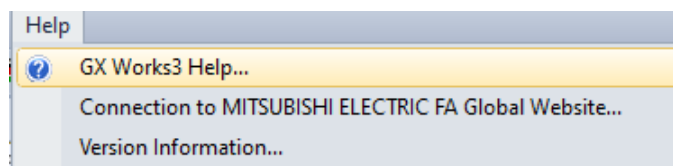


Rysunek 42: Tryb Online Program Change

Należy pamiętać, że uwzględnienie większych modyfikacji projektu jest możliwe wyłącznie przez zatrzymanie symulacji, dokonanie przebudowy projektu, a następnie ponowne włączenie symulacji.

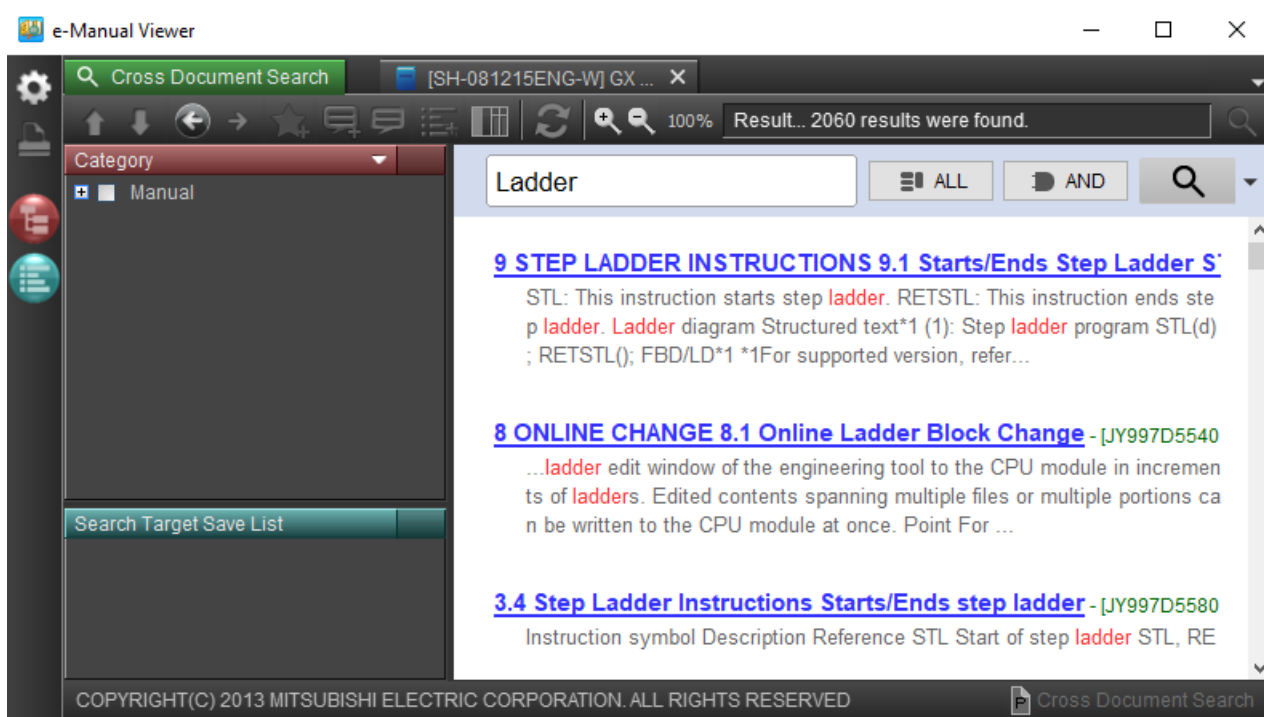
2.10. E-Manual Viewer

Oprogramowanie GX Works 3 ma ogromne możliwości, dlatego również bogata jest zawartość opisu wszystkich jego dostępnych funkcji. Wszystkie informacje są zamieszczone w E-Manual Viewer do którego dostęp jest poprzez wybranie odpowiedniej opcji jak to pokazano na rys. 43.



Rysunek 43: Otworzenie E-Manual Viwer

W otwartym dokumencie można przeszukiwać dowolną jego zawartość pod kątem wszystkiego co nas interesuje czy jest nam potrzebne do napisania efektywnego programu sterującego. Na rys. 44 przedstawiono przykładowe wyszukiwanie po słowie kluczowym.

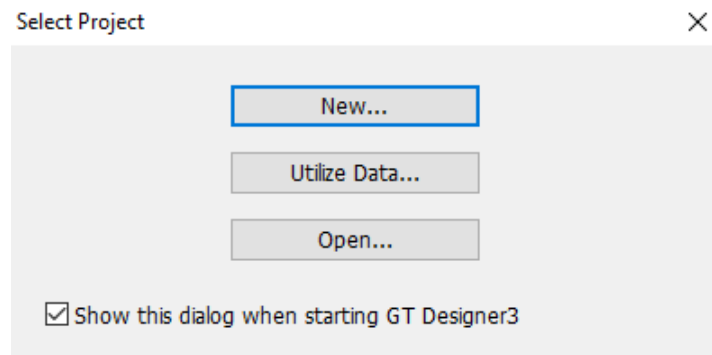


Rysunek 44: E-Manual Viwer

3. Oprogramowanie GT Designer 3

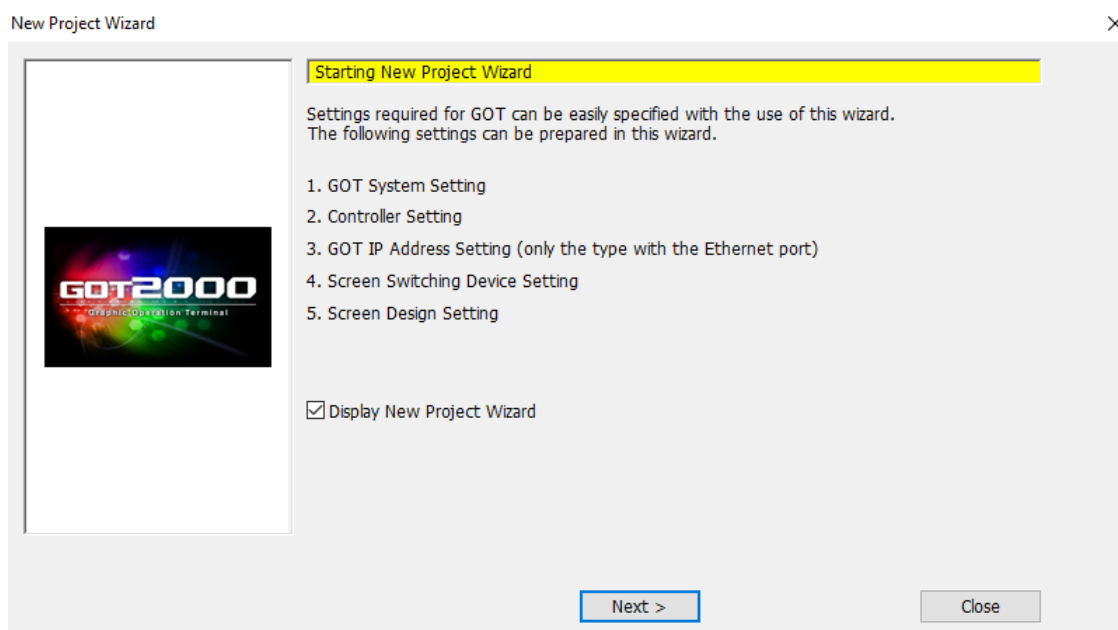
3.1. Tworzenie nowego projektu

Oprogramowanie GT Designer 3 jest niezbędne do tworzenia wizualizacji działania programu, napisanego w GX Works 3, na panel HMI. Tworzenie projektu zaczynamy od uruchomienia oprogramowania. Pojawi się okno jak na rys. 45. Jeśli okno nie zostanie automatycznie otwarte to należy wybrać opcję Project—>New, a jeżeli okno jest już otwarte to też wybieramy tą opcję w oknie.



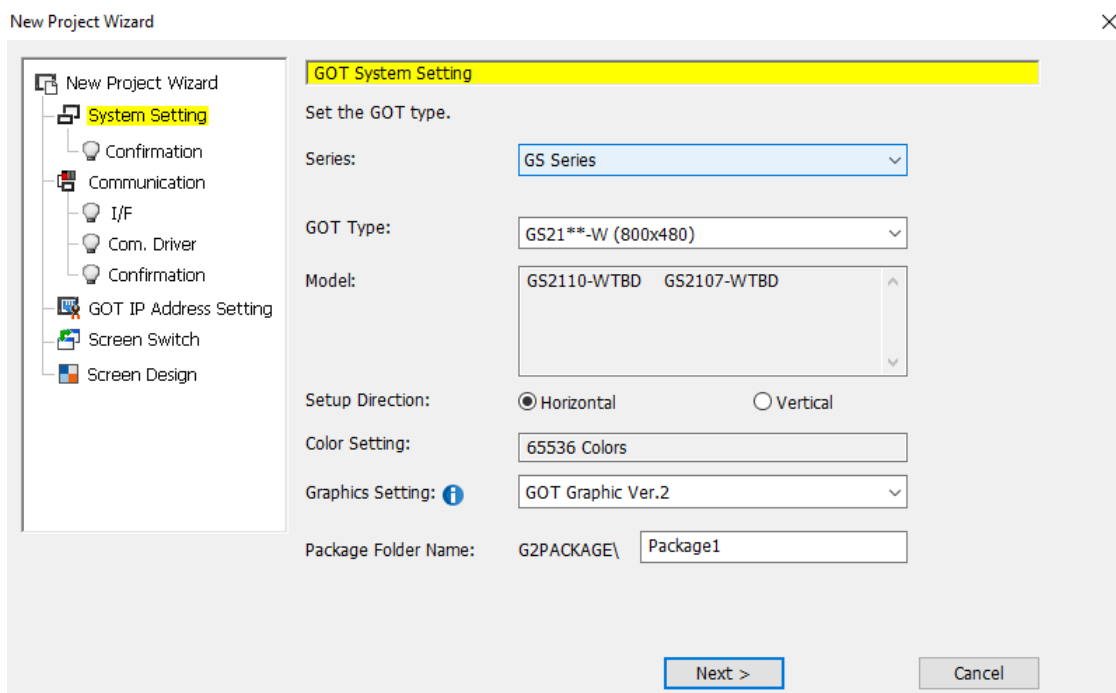
Rysunek 45: Okno do tworzenia nowego projektu

Następnie otworzy się okno konfiguracji nowego projektu jak to przedstawiono na rys. 46. Konfiguracja projektu jest podzielona na wiele kroków w związku z tym aby nawiązać właściwą komunikację między sterownikiem a panelem HMI.



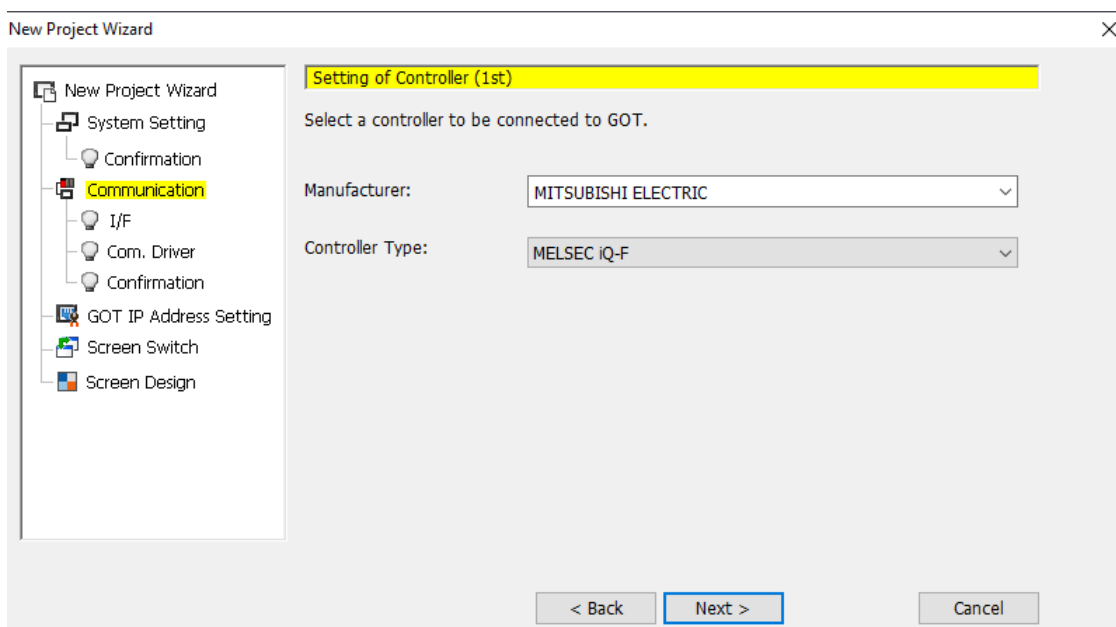
Rysunek 46: Tryb konfiguracji projektu

Podczas wyboru typu symulatora należy wybrać symulator GS21 z serii GS tak jak na rys. 47.



Rysunek 47: Wybór symulatora

Należy przejść do dalszych kroków akceptując ustawienia. W momencie pojawienia się okna do wpisania typu sterownika należy wybrać typ IQ-F jak pokazano na rys. 48.

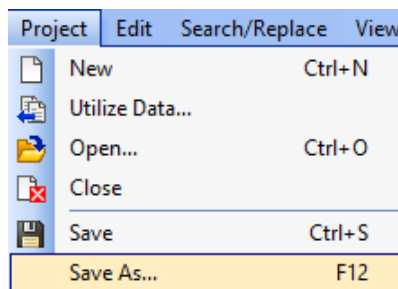


Rysunek 48: Wybór typu sterownika

Dalsze ustawienia należy zaakceptować bez zmian. Po wykonaniu wszystkich kroków nowy projekt zostanie utworzony i można rozpocząć jego edycję.

3.2. Zapisanie projektu

Pierwsze zapisanie projektu odbywa się poprzez wybranie opcji Project —> Save As jak na rys. 49. Identycznie jak w oprogramowaniu GX Works 3 można stosować skrót klawiszowy Ctrl+S do zapisu projektu w każdym momencie lub wybranie opcji Save. Należy również unikać stosowanie znaków specjalnych w nazwach projektów.



Rysunek 49: Zapisanie projektu

3.3. Okna projektu

Projekt ma wiele okien wartych omówienia. Górne okno podobnie jak w oprogramowaniu GX Works 3 ma rozwijane listy i ikony szybkiego dostępu do popularnych opcji jak pokazano na rys. 50. Zakładki są następujące:

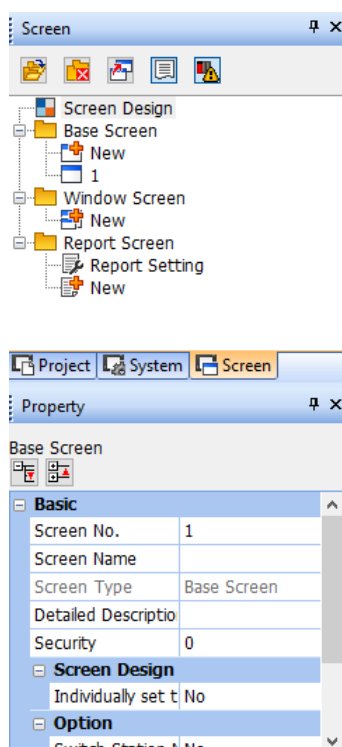
1. Project, gdzie występują m.in. opcje tworzenia, otwierania oraz zapisu projektu.
2. Edit, gdzie występują m.in. opcje edycji poszczególnych elementów graficznych.
3. Search/Replace, gdzie występują m.in. opcje znalezienia oraz zamiany istniejących elementów projektu.
4. View, gdzie występują m.in. opcje pokazania czy ukrycia poszczególnych okien projektu.
5. Screen, gdzie występują m.in. opcje tworzenia nowych obrazów projektu.
6. Common, gdzie występują m.in. ustawienia oprogramowania jak i sterownika.
7. Figure, gdzie występują m.in. rysowania i dodawania różnych figur do obrazów projektu.
8. Object, gdzie występują m.in. opcje tworzenia różnych przycisków, diód czy innych elementów graficznych.

9. Communication, gdzie występują m.in. ustawienia komunikacji z panelem HMI.
10. Diagnostics, gdzie występują informacje diagnostyczne o panelu HMI.
11. Tools, gdzie występują m.in. opcje związane z włączeniem symulacji programu.
12. Window, gdzie występują m.in. opcje z rozlokowaniem okien projektu.
13. Help, gdzie występują m.in. informacje o oprogramowaniu.



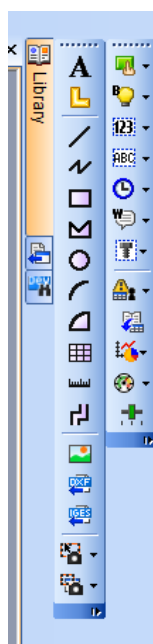
Rysunek 50: Rozwijane listy i ikony szybkiego dostępu do popularnych opcji

Po lewej stronie projektu znajduje się okno o nazwie Screen służące do dodawania nowych obrazów projektu. Poniżej jest okno Property w którym można obejrzeć i modyfikować właściwości zaznaczonych obiektów bądź obrazów. Okna zostały przedstawione na rys. 51.



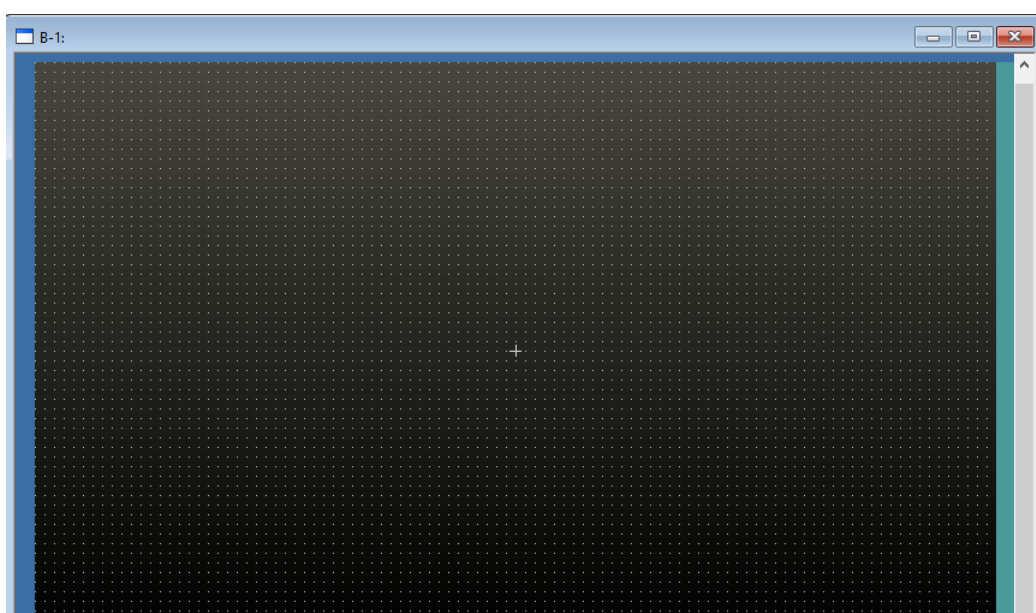
Rysunek 51: Okna Screen i Property

Po prawej stronie projektu znajduje się okno do wstawiania lub rysowania elementów graficznych. Zostało ono przedstawione na rys. 52.



Rysunek 52: Okno z wyborem elementów graficznych

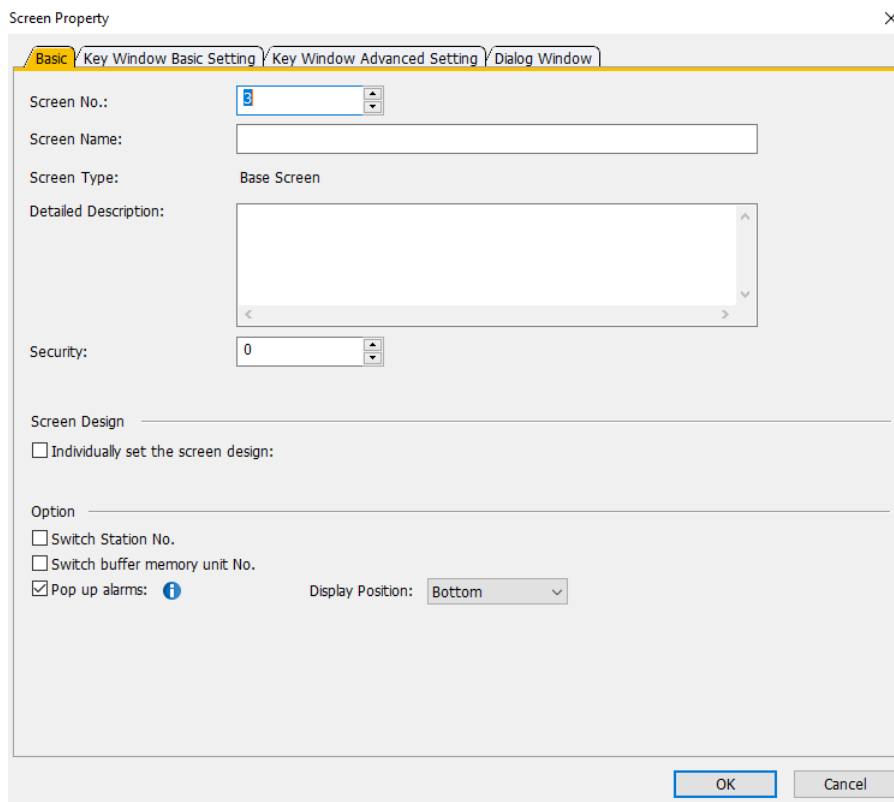
Na środku projektu znajduje się okno do edycji poszczególnych obrazów projektu. Przykładowe puste okno zostało przedstawione na rys. 53.



Rysunek 53: Okno edycji projektu

3.4. Dodawanie obrazów

Istnieją 3 typy obrazów Base Screen, Window Screen i Raport Screen. Dodawanie nowych obrazów do utworzonego projektu jest możliwe poprzez wybranie Opcji New z okna po lewej stronie projektu z odpowiedniej gałęzi drzewa. Obrazy typu Base Screen są na cały ekran symulacji. Obrazy typu Window Screen są obrazami o określonym rozmiarze i mogą być umieszczone w formie okienek na obrazie typu Base Screen. Obrazy Raport Screen są obrazami zbierającymi informacje do drukowania raportów. Po wybraniu opcji New danego typu obrazu można ustawić jego właściwości. Na rys. 54 pokazano okno ustawienia właściwości obrazu typu Base Screen. Należy pamiętać każdy obraz tego typu posiada swój unikalny numer identyfikacyjny, który można potem wykorzystywać do tworzenia ciekawych wizualizacji.



Rysunek 54: Właściwości obrazu

Po dodanie obrazu wyświetli się on w głównym oknie projektu. Kolejność wyświetlania obrazów można zmienić z poziomu drzewa czy też przesuwając zakładki obrazów w oknie głównym. W każdej chwili można zmienić właściwości obrazu poprzez kliknięcie prawym przyciskiem nazwy dodanego obrazu w drzewie projektu i wybrania opcji Screen Property.

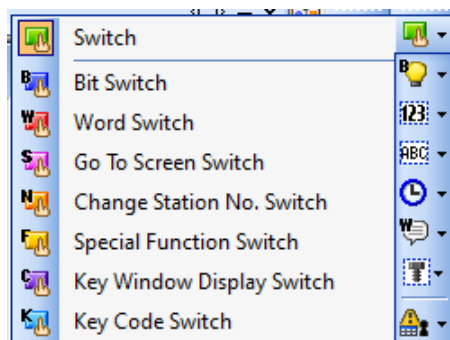
3.5. Projektowanie obrazów

Z okna po lewej stronie ekranu można w łatwy sposób używać różnych elementów graficznych poprzez przeciągnięcie do obrazu lub zaznaczenie obszaru w obrazie. Pasek na rys. 55 służy do rysowania ręcznie różnych figur. Poprzez kliknięcie dwukrotne na tak utworzoną figurę można przejść do jej właściwości i je zmodyfikować.



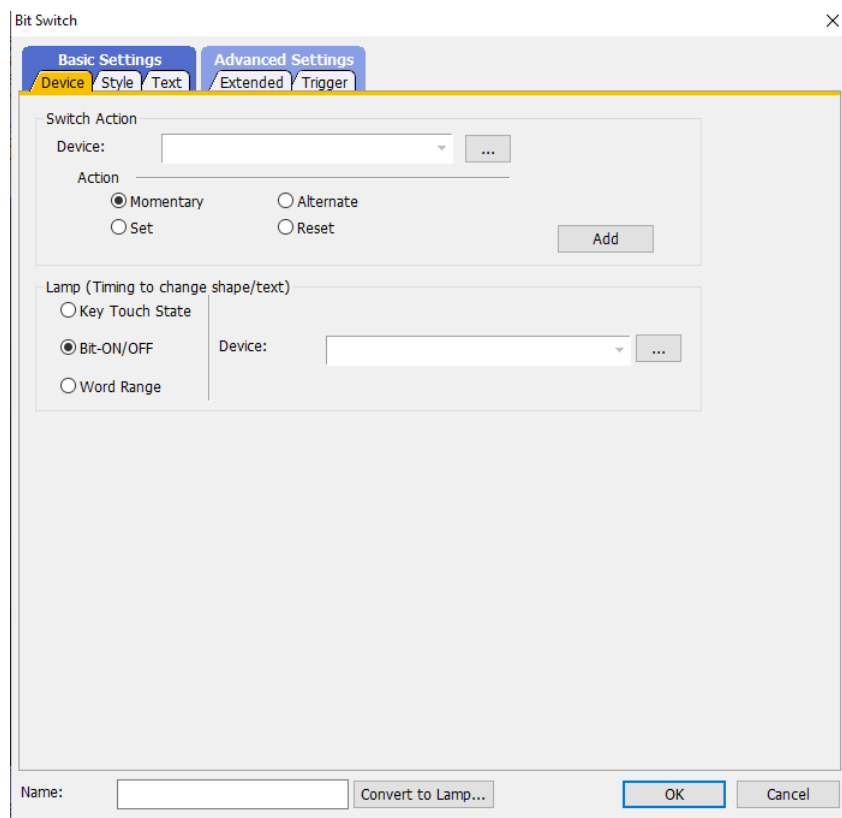
Rysunek 55: Pasek rysowania własnych elementów graficznych

Większość elementów graficznych, które można dodać do obrazów, posiada rozdzielone właściwości na podstawowe i zaawansowane. Właściwości podstawowe wiążą się z podstawowym i prostym zachowaniem się obiektu graficznego do którego należą. Właściwości zaawansowane dają większe możliwości w zachowaniu się elementu graficznego i sprawiają, że użytkownik może zaprojektować o wiele bardziej złożoną symulację swojego programu. Obok paska z rysowaniem różnych figur znajduje się pasek m.in z opcją dodawania różnych przycisków jak to pokazano na rys.56.



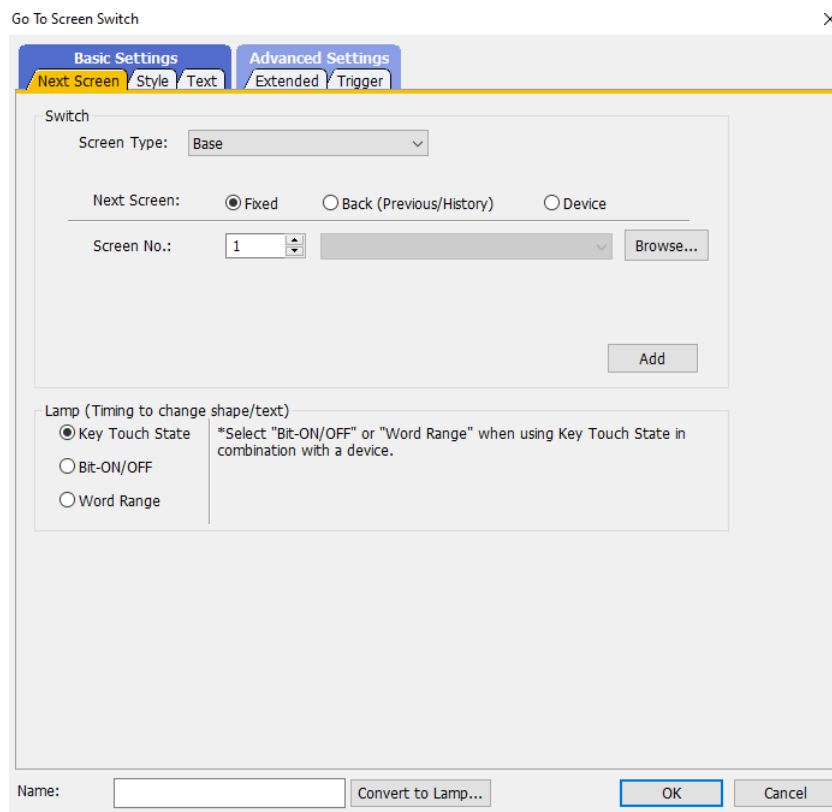
Rysunek 56: Typu przycisków

Do wyboru są m.in. zwykłe przyciski typu bit/word, przejścia do kolejnego obrazu czy posiadających specjalną funkcję. Na rys. 57 przedstawiono otwarte właściwości przycisku typu bit.



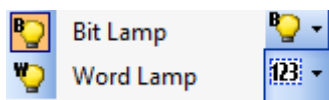
Rysunek 57: Właściwości przycisku typu bit

W oknie można m.in. ustawić czy przycisk ma być przyciskiem monostabilnym, bistabilnym, grafikę przycisku w dwóch stanach, co powoduje zmianę graficzną stanu i wiele innych opcji w zależności od potrzeb użytkownika. W oknie Device należy wpisać adres bitu który to będzie podłączony pod ten przycisk podczas symulacji.



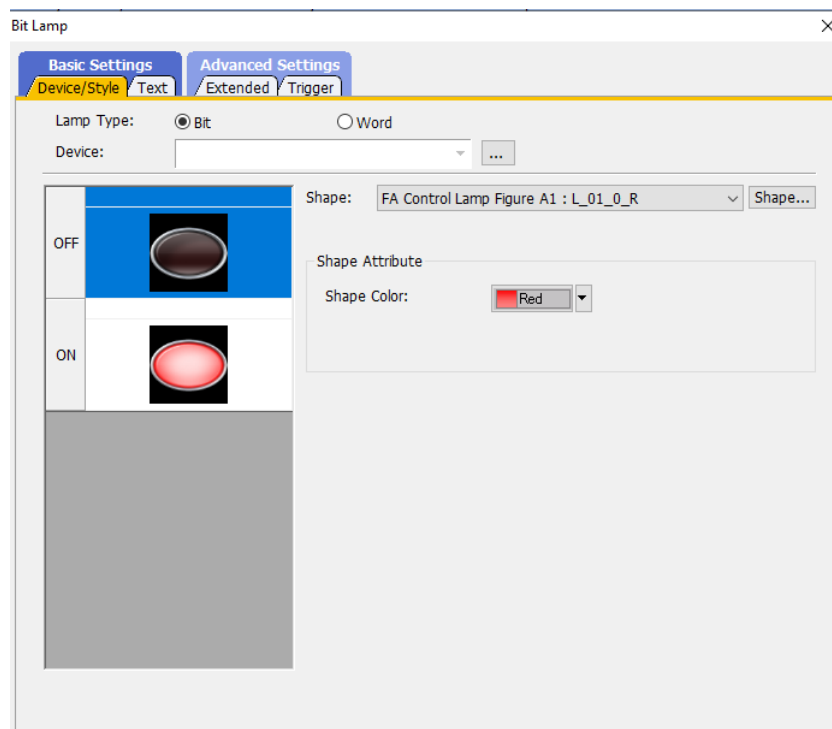
Rysunek 58: Właściwości przycisku typu Go To Screen

Na rys. 58 pokazano właściwości przycisku Go To Screen. Przycisk ten jest użyteczny podczas symulacji programu przy pomocy większej liczby obrazów. Po kliknięciu przycisku podczas symulacji pozwala na otworezenie innego obrazu. We właściwościach przycisku podaje się unikalny numer danego obrazu aby mógł to funkcję spełnić. Posiada również podobne możliwości ustawień właściwości co przycisk typu bit.



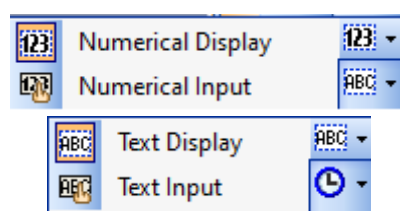
Rysunek 59: Typy diód

Do danego obrazu można również dodawać elementy typu dioda. Na rys. 59 pokazano typy dostępnych diód. W podobny sposób są skonstruowane właściwości tych elementów jak dla elementów typu przycisk.



Rysunek 60: Właściwości diody typu bit

Na rys. 60 przedstawiono właściwości diody typu bit. W polu Device również wpisuje się adres bitu, od którego zapalenie diody będzie uzależnione. Ponadto poniżej znajdują się elementy do wpisywania wartości liczbowych czy też tekstu, a także do ich wyświetlania. Listy pokazano na rys. 61.

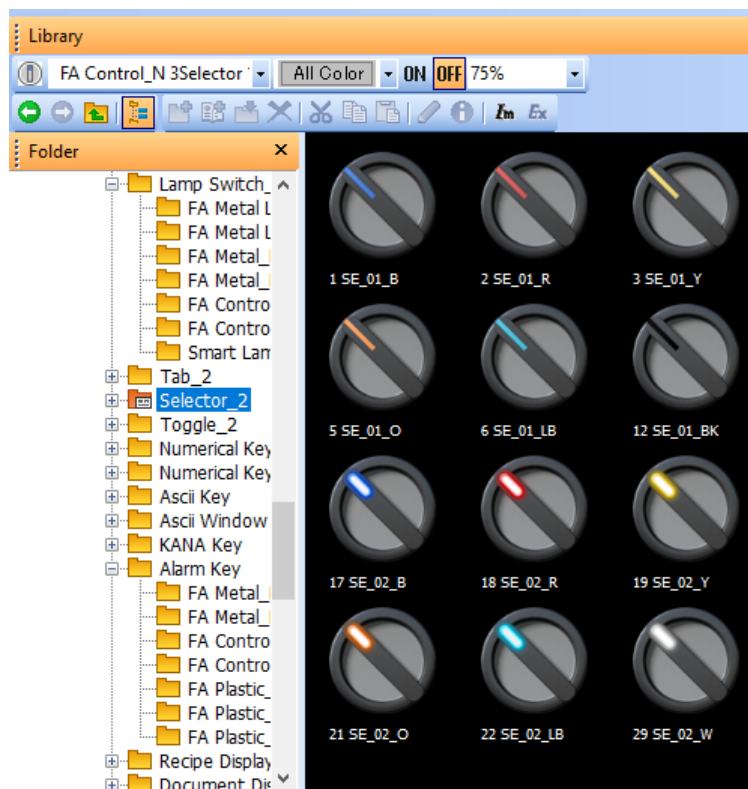


Rysunek 61: Elementy do wpisywania wartości liczbowych/tekstu i wyświetlania zawartości

Oprogramowanie GT Designer 3 posiada wiele różnych elementów graficznych do projektowania złożonych symulacji programów. Przedstawione elementy są elementami podstawowymi i najczęściej używanymi przy projektowaniu obrazów.

3.6. Biblioteka

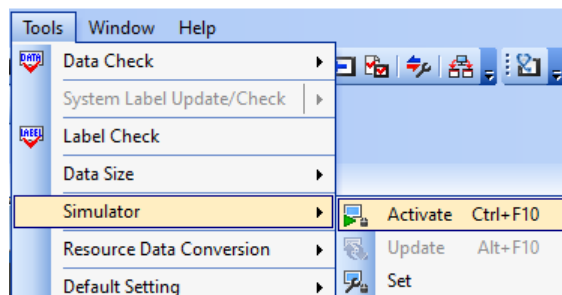
Oprogramowanie posiada również bogatą bibliotekę specjalnych elementów graficznych, które można przeciągnąć, modyfikować właściwości tych elementów i tworzyć bardziej interesujące obrazy. Biblioteka znajduje się przy prawym oknie i cała jej zawartość jest widoczna po rozciągnięciu paska Library jak pokazano na rys. 62.



Rysunek 62: Ustawienie symulatora

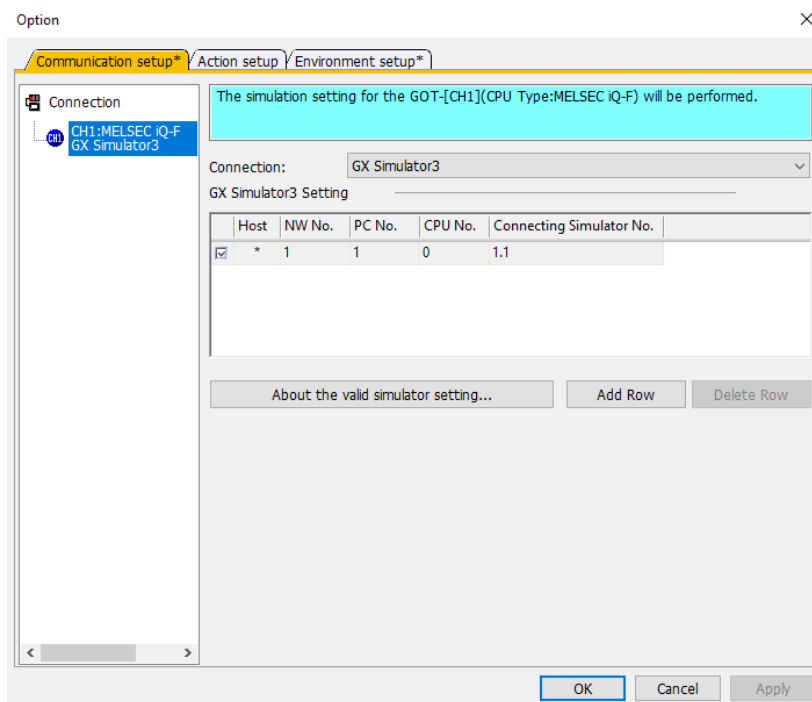
3.7. Symulacja

Symulację utworzonej wizualizacji można rozpocząć poprzez kolejne przejście Tools —> Simulator —> Activate jak to pokazano na rys. 63. Po wybraniu tej opcji po chwili rozpocznie się symulacja programu. Należy pamiętać, że symulacja jest niemożliwa w przypadku gdy program w GX Works 3 nie jest również w trybie symulacji.



Rysunek 63: Ustawienie symulatora

Jeżeli przez przypadek projekt został utworzony z złymi ustawieniami symulatora to należy przejść do opcji Set znajdującej się pod opcją Activate i wybrać z rozwijanej listy GX Simulator3.

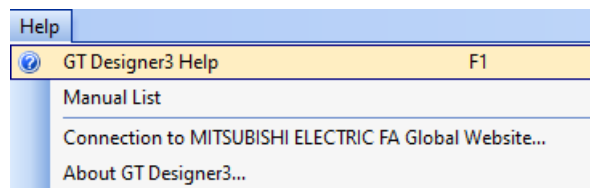


Rysunek 64: Ustawienie symulatora

Po zmianie symulatora symulacja powinna się rozpocząć bez jakichkolwiek przeszkód.

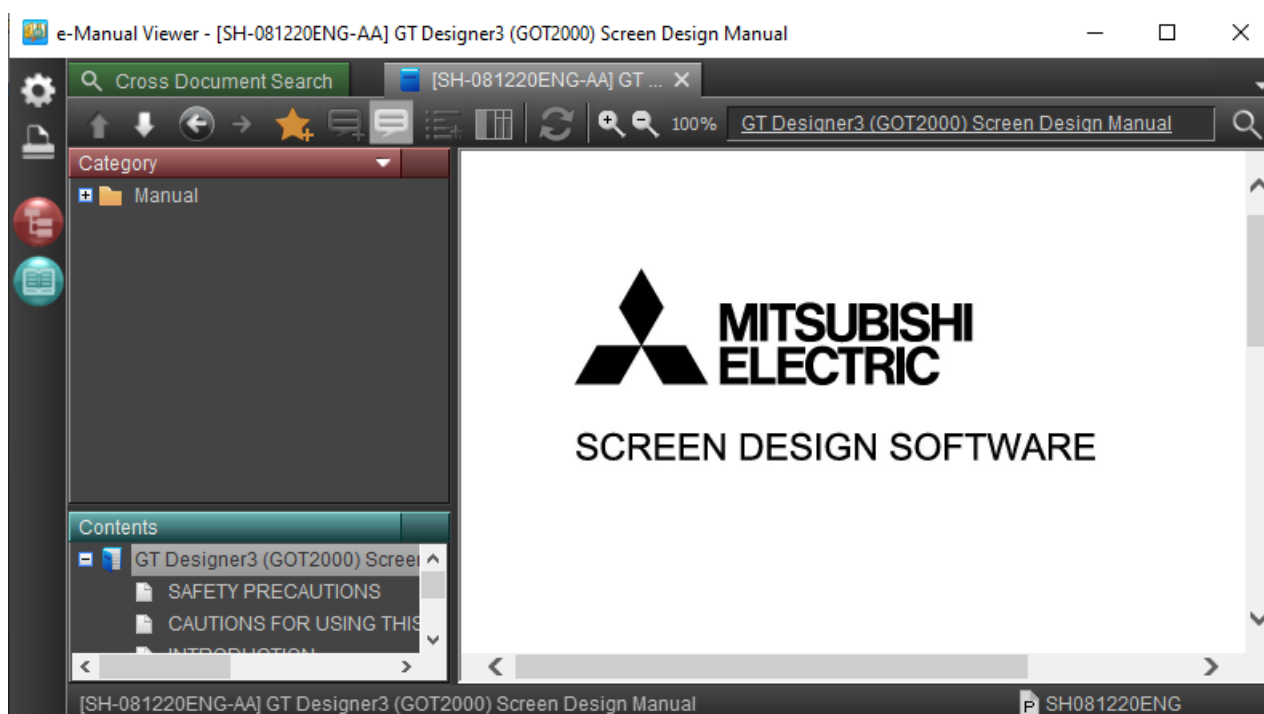
3.8. E-Manual Viewer

Dostęp do dokumentacji E-Manual Viewer jest również możliwy dla oprogramowania GT Designer 3. W tym celu należy przejść i wybrać opcję jak na rys. 65.



Rysunek 65: Przejście do dokumentacji

Po wybraniu opcji otworzy się E-Manual Viewer jak na rys. 66 dla tego oprogramowania i podobnie jak w oprogramowaniu GX Works 3 można przeszukiwać jego zawartość po słowach kluczowych.

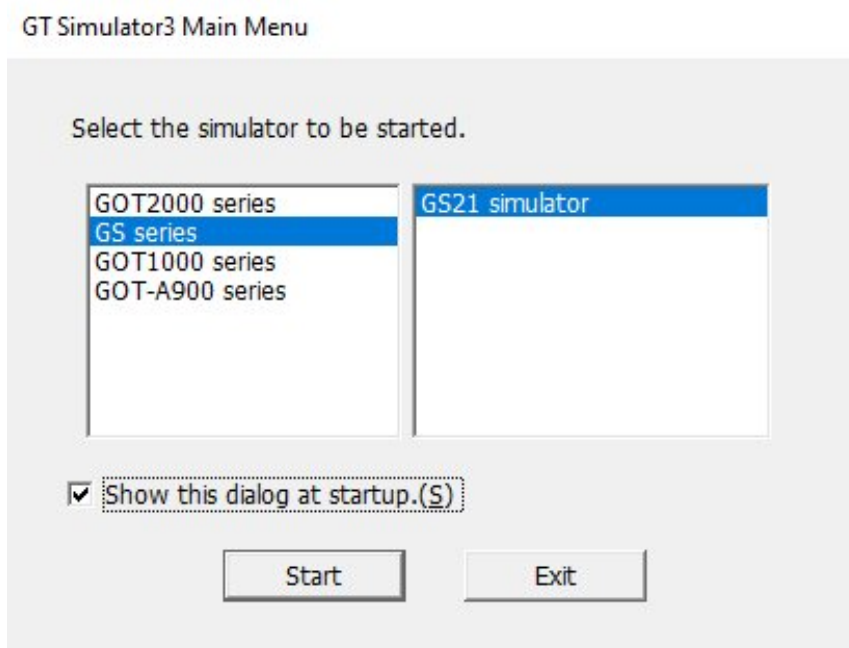


Rysunek 66: E-Manual Viewer

4. Oprogramowanie GT Simulator 3

4.1. Wybór symulatora

Graficzną symulację programu można również uruchomić bezpośrednio poprzez otwarcie GT Simulator 3. Po uruchomieniu oprogramowania pokaże się okno, w którym to należy wybrać GS21 simulator tak jak pokazano na rys.67.



Rysunek 67: Wybór symulatora

Wybranie tego symulatora jest niezbędne do otworzenia wizualizacji utworzonej uprzednio w GT Designer 3.

4.2. Symulacja

Symulacja graficzna programu jest tylko możliwa jeżeli program napisany w GX Works 3 jest w trybie symulacji. W przeciwnym razie oprogramowanie GT Simulator 3 zakomunikuje ten fakt i uniewmożliwi uruchomienie symulacji. Po wybraniu symulatora GS21 pojawi się okno gdzie należy wybrać opcję Open File jak pokazano na rys. 68 i otworzyć plik z utworzoną wcześniej wizualizacją.



Rysunek 68: Otwarcie projektu w GT Simulator 3

Po chwili uruchomi się symulator i będzie możliwa weryfikacja poprawności działania programu napisanego w GX Works 3.