PANELE OPERATORSKIE WOP-2000

WebOP Designer 2.0 Instrukcja obsługi



Spis treści

1. Tworzenie projektu i konfiguracja połączeń	4 -
1.1 Tworzenie projektu	4 -
1.2 Port Ethernet	5 -
1.3 Transmisja Transparent Serwer	7 -
2. Odczyt i zapis wartości z/do rejestrów	8 -
2.1 Odczyt i zapis wartości rejestru	8 -
2.1.1 Utworzenie połączenia z modułem ADAM-4055 i interfejsem RS-485	8 -
2.1.2 Odczyt stanów binarnych z urządzenia na przykładzie modułu ADAM-4055	9 -
2.1.3 Odczyt przy pomocy makra	10 -
2.1.4 Zapis stanów logicznych na przykładzie modułu ADAM-4055	13 -
2.1.4.1 Obsługa kontrolki Bit Button	14 -
2.1.4.2 Obsługa kontrolki Toogle Switch	15 -
2.1.5 Odczyt wejść analogowych na przykładzie modułu ADAM-4017+	16 -
2.1.5.1 Odczyt bezpośredni rejestru	16 -
2.1.5.2 Odczyt rejestru przez makro	18 -
3. Elementy graficzne	18 -
3.1 Blok Bar Graph	18 -
3.2 Powiązanie bloku wskaźnika Meter z suwakiem typu Slider	20 -
3.3 Powiązanie bloku graficznego Picture Dispaly z blokiem Bar Graph i suwakiem Slider	21 -
4. Dodawanie Tagów	23 -
5. Receptury	24 -
6. Logowanie danych	27 -
6.1 Dodawanie i konfiguracja	27 -
6.2 Tabale i wykresy wykorzystujące Data Logger	30 -
6.2.1 Historic Data Table	30 -

	6.2.2 Historic Trend Graph	- 31 -
	6.2.3 Zapis danych typu Data Logger do pliku tekstowego	- 31 -
7.	Uruchamianie i testowanie	- 32 -
7	7.1 Kompilacja i wgrywanie projektu do sterownika	- 32 -
7	2.2 Testowanie i symulacja	- 33 -
8. F	oziomy dostępu do aplikacji panelu	- 35 -
9. L	anguages	- 37 -

1. Tworzenie projektu i konfiguracja połączeń

1.1 Tworzenie projektu



Rysunek 1. Tworzenie projektu

W pierwszej kolejności wchodzimy w menu projektu i wybieramy pole **New**. Następnie wybieramy model panelu operatorskiego w polu **Panel Type** oraz nadajemy nazwę projektu. Kolejno definiujemy połączenia z urządzeniami współpracującymi z panelem. W tym celu wybieramy interfejs oraz typ połączenia w polu "**Link Type**":

- "**Direct Link (COM)**" umożliwia komunikację poprzez port szeregowy (w przypadku RS-485 nie jest definiowany adres tak więc w późniejszym etapie do połączenia można przypisać kilka urządzeń np. przy protokole ModbusRTU).

- "Direct Link (Ethernet)" zapewnia komunikację poprzez sieć Ethernet ze zdefiniowanym adresem IP.
- "Gateway Service (COM)" jest to brama RS.
- "Gateway Service (Ethernet)" jest to brama Ethernet.

aa 1	
m PM Designer -	
III Elle View Iools Help	
<u>, b w ≓, ii, b H to te ¢ </u> ¢ ⇒	NELER, 2 A DE A C SAN REFEROR & E MAA EB."
Project manager Text Database	Link Number: 1 Link Name: Link II Link Type: Decide/Server: Decide/Server: Decide/Server: Link Port: Galeway Service [Ethernet]
	Record communication status in operation log The duration of showing a communication error message: 5 v second(s)
124, 23	100%

Rysunek 2. Wybór interfejsu w połączenia Link

W kolejnym kroku należy wybrać protokół komunikacyjny w polu "**Device/Server**" oraz numer interfejsu w polu "**Link Port**".

New Link	New Link
Link Number: 1 Link Name: Com2, 57:200 Link Type: Direct Link (COM) Device/Server: Simmatic 57:200 (PPI; 14:o 1) Link Port: COM2	Link Number: 1 Link Name: Modbus Device/Slave (TCP/IP) Link Type: Direct Link (Ethernet) Device/Slave: Modicon Corp. Modbus Device/Slave (TCP/IP) Link Port: Ethernet1
Record communication status in operation log	
The duration of showing a communication error message: 5 💌 second(s)	The duration of showing a communication error message: 5 💌 second(s)

Rysunek 3. Definiowanie połączenia Link

1.2 Port Ethernet

Aby skorzystać z interfejsu Ethernet, na etapie konfiguracji połączenia Link, w polu Link Type należy określić parametr Direct Link (Ethernet). Przy tym interfejsie w polu Device/Server mamy bogaty wybór protokołów, które

mogą posłużyć do komunikacji z urządzeniami wielu producentów. W naszym przykładzie skorzystamy z protokołu Modbus TCP niezbędnego do komunikacji z modułem wejść analogowych ADAM-6017 firmy Advantech. W tym celu wybieramy protokół **Modicon Corp**. oraz **Modbus Device/Slave (TCP/IP).** We właściwościach połączenia Link podajemy adres IP modułu, port komunikacyjny (domyślnie dla połączeń Modbus TCP wynosi 502) oraz **Node Address**, który dla serii ADAM-6000 zawsze wynosi 1.

Link Properties				
General Paramet	er			
Link Number:	1			
Link Name:	6017			
Link Type:	Direct Link (Ethernet)			~
Device/Server:	Modicon Corp.	ModBus Device/Slave (TC	CP/IP)	*
Link Port:	Ethernet1	*	📃 Sub-links	
	unication status in operation log			
The duration of s	howing a communication error me	ssage: 5 💌 second(s)		
			ОК	Cancel Help
Link Properties				X
Link Properties	2			X
General Parameter	a 100 - 2 - 101			×
Link Properties General Parameter IP Address: 19	त्र 2.168.2.191			X
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502	er 2.168.2.191 ort			×
Link Properties General Parameter IP Address: 19 Vuse Default Pr Port: 502	er 2.168.2.191 ort			X
Link Properties General Parameter IP Address: 19 ✓ Use Default Pr Port: 502 Node Address:	er 2 . 168 . 2 . 191 ort			
Link Properties General Parameter IP Address: 19 ♥ Use Default Port: 502 Node Address:	er 2 . 168 . 2 . 191 ort 1			X
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time:	ar 2 . 168 . 2 . 191 ort 1			X
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay:	er 2 . 168 . 2 . 191 ort 1 0 \$ (x 0.1 Sec.) 0 \$ (x 0.1 Sec.)			
Link Properties General Parameter IP Address: 19 ✓ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 1 0 ♀ (x 0.1 Sec.) 0 ♀ (x 0.1 Sec.) 0 ♀			
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2 . 168 . 2 . 191 ort 1 0 \$ (x 0.1 Sec.) 0 \$ (x 0.1 Sec.) 0 \$			
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 1 0 \$ (x 0.1 Sec.) 0 \$ (x 0.1 Sec.) 0 \$			
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 1 0 ♀ (x 0.1 Sec.) 0 ♀ (x 0.1 Sec.) 0 ♀			
Link Properties General Parameter IP Address: 19 ♥ Use Default Pu Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 11 0 \$ (x 0.1 Sec.) 0 \$ (x 0.1 Sec.) 0 \$			
Link Properties General Parameter IP Address: 19 ♥ Use Default Pr Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 1 0 ♥ (x 0.1 Sec.) 0 ♥ (x 0.1 Sec.) 0 ♥			
Link Properties General Parameter IP Address: 19 ♥ Use Default Port: 502 Node Address: Timeout Time: Command Delay: Retry Count:	ar 2.168.2.191 ort 11 0 ♀ (x 0.1 Sec.) 0 ♀ (x 0.1 Sec.) 0 ♀		ΟK	

Rysunek 4. Dodawanie połączenia Link z portem Ethernet przy pomocy protokołu Modbus TCP

1.3 Transmisja Transparent Serwer

Tryb pracy Transparent Serwer umożliwia utworzenie lustrzanego, przezroczystego połączenia portu RS za pośrednictwem innego portu szeregowego. W takiej konfiguracji dane mogą być odczytywane lub zapisywane do urządzeń podłączonych do panela operatorskiego za pośrednictwem narzędzi zewnętrznych, takich jak PC z zainstalowanym oprogramowaniem typu Scada.

W pierwszej kolejności konfigurujemy połączenie główne w panelu operatorskim "Link 1". Do transmisji wybieramy protokół Modbus RTU na porcie COM1.

	Link Properties			
	General Paramet	er		
	Link Number:	٥		
Project Manager Text Database	Link Name:	Link 1		
	Link Type:	Direct Link (COM)		~
Global	Device/Server:	Modicon Corp.	Modicon 984 Device/Slave (RTU)	v
	Link Port:	COM1 (Link 1)	Sub-links	
Font Templates				
Picture Database				
Sound Database				
In Text Database				
Global Macros				
	Becord comm	unication status in operation log		
🖨 😻 Links		and alon status in operation log		
Internal Memory				
	The duration of s	howing a communication error mes	ssage: 5 💌 second(s)	
Tags Rename				
			OK	Cancel Help

Rysunek 5. Zakładka General w ustawieniach połączenia Link 1

Link Properties		X
General Parameter		
Transmission Baud Rate: 9600 V Data Bits: 8 V Parity: None V Stop Bits: 1 V	Others Panel Address: 1 PLC Address: 1 Timeout Time: 0 Command Delay: 0 Retry Count:	
N.	ОК	Cancel Help

Rysunek 6. Zakładka Parameter w połączeniu Link 1

Następnie dodajemy nowe połączenie "Link 2", które ustawiamy w trybie "2-to-1 Transparent Serwer (COM)". W polu "Data Link" wybieramy port mapowany a w polu "Link Port" port na którym będą dostępne mapowane rejestry.

	Link Properties		
	General Paramet	er	
	Link Number:	E	
	Link Name:	Link 2	
	Link Type:	Communication Service (COM)	~
Project Manager Text Database	Device/Server:	PanelMaster 2-to-1 Transparent Server (COM)	~
🖃 🚰 test	Link Port:	COM2 (Link 2)	
Global	Data Link:	Link 1	~
The Font Templates			
Ricture Database			
- 🍊 Sound Database			
To Text Database			
Global Tags			
Global Macros			
	The duration of s	howing a communication error message: 5 💉 second(s)	
Add Link			
Driver List			
Tags		UK	Lancei Help

Rysunek 7. Ustawienia w połączeniu Link 2

2. Odczyt i zapis rejestrów

2.1 Odczyt i zapis wartości rejestru

2.1.1 Utworzenie połączenia z modułem ADAM-4055 z interfejsem RS-485

Klikamy prawym klawiszem na drzewie projektu na Link i wybieramy Add Link. Następnie podajemy nazwę połączenia i wybieramy odpowiedni port - tutaj Direct Link (COM). Z listy obsługiwanych protokołów wybieramy Modicorn Corp. oraz Modicon 984 Device/Slave (RTU), który odpowiada protokołowi Modbus RTU i zatwierdzamy. Po tym, klikamy prawym klawiszem myszy na utworzone połączenie i wchodzimy w ustawienia. W zakładce Parameter ustawiamy parzystość Parity jako None i zatwierdzamy.

New Link				X	
Link Number:	1				
Link Name:	Link 1				
Link Type:	Direct Link (COM)		~		
Device/Server:	Modicon Corp.	Modicon 984 Device/Slave (RTU)	~		
Link Port:	COM1	Sub-links			
Record comm	unication status in operation log				
The duration of s	The duration of showing a communication error message: 5 💌 second(s)				

Rysunek 8. Dodawanie połączenia Link

peral Para	ies ameter				
neral Para Transmissic Baud Rate Data Bits: Parity: Stop Bits:	ameter 500	Others Panel Address: PLC Address: Timeout Time: Command Delay: Retry Count:	1 1 0 0 0 0 0 0 0 0 0 0	(x 0.1 Sec.) (x 0.1 Sec.)	

Rysunek 9. Dodawanie połączenia Link

2.1.2 Odczyt stanów binarnych z urządzenia na przykładzie modułu ADAM-4055

Do wyświetlania stanów binarnych w programie WebOP Designer służy kontrolka **Bit Lamp**. W celu dodania kontrolki należy kliknąć lewym klawiszem myszy na ikonie **P** na pasku narzędziowym lub wybrać ją z górnego menu **Object->Bit Lamp**, i umieścić ją na polu ekranu.



Rysunek 10. Kontrolka Bit Lamp w obszarze roboczym projektu

Bit Lamp		٤	K
ID: BL0000	Visibility Note:		
NO_B	DR	Shape State: On Off Border Color: A FG Color: A Pattern: A BG Color: A	
Address Type:	Bit 2:00001		
Monitor Address.	2.00001		
			Address Input Keypad ? 🔀
			Link: Link 1 💌
			2 • : • 0001 C A 1 2 3 CLR D B 4 5 6 BS E : 7 8 9 ESC
		OK Cancel Help	F / . 0 ENT

Dwukrotne kliknięcie na dodanej kontrolce otwiera okno konfiguracji:

Rysunek 11. Właściwości kontrolki Bit Lamp

W przypadku modułu ADAM-4055 dane o stanie wejść binarnych pobierane są z rejestrów 00001-00008 kolejno dla kanałów DI0-DI7. W celu dodania monitorowanego bitu należy w polu **Address Type** wybrać wartość **Bit**. Następnie

kliknąć klawisz 📖 a po otwarciu się okna **Address Input Keypad** w polu **Link** wybierać Internal Memory lub inne zdefiniowane połączenie np. domyślnie Link 1. Teraz należy podać adres urządzenia, typ zmiennej oraz numer rejestru np. 00001. Po poprawnym wprowadzeniu danych należy kliknąć przycisk **ENT**.

2.1.3 Odczyt przy pomocy makra

Inną możliwością odczytu stanów rejestrów jest wykorzystanie makra czasowego wywoływanego np. co dwie sekundy. W takim przypadku generowany ruch w sieci jest mniejszy a dane uaktualniane są regularnie co z góry ustalony przedział czasowy. Dodatkowa dane mogą być przetrzymywane w pamięci panelu operatorskiego, co umożliwia ich zastosowanie w wielu procesach jednocześnie i zmniejsza obciążenie urządzenia.

a) Aby dodać makra czasowe w drzewie projektu klikamy prawym przyciskiem myszy na Macros i wybrać opcję Add Macro.



Rysunek 12. Dodawanie makra do projektu

W otwartym oknie **New Macro** podajemy nazwę makra i zatwierdzamy OK. Po tej czynności w drzewie Macro pojawi się nowe makro, które możemy edytować poprzez dwukrotne kliknięcie myszy. W edytorze makr, w polu **Command** wybieramy pozycję o określonym stanie logicznym - w naszym przypadku "=" z kolumny **Move/Set**.

	×	Properties		₽×
<mark>,c</mark> ∢	•_ # X	Command:	•	
	General Move/Set = = ! MOV SETM Arithmetc + *	Command: IF/ELIF Program Con Logical I & ^ Calculate MAX MIN BMAX	trol Utilities Convert BCD BIN DW W B2W W2B A2X X2A W2F DOT	
	/ %	BMIN SUM XSUM SWAP	D2F F2W F2D	

Rysunek 13. Dodawanie komendy do makra

Dalej w ustawieniach **Propertis** definiujemy zmienne:

Properties		₽	×
1.0=2:00001(B)		
Command:	P1 = P2		
Data Type:	(B) Bit 🔽		
Par. P1:	1.0		
Par. P2:	2:00001		

Rysunek 14. Wybór parametrów dla komendy makra

W polu **Data Type** wybieramy typ zmiennej jako **(B) Bit**. Do pola **Par. P1** wpisujemy zmienną lokalną np. \$U1.0 do której będzie zapisywany stan rejestru z urządzenia. Parametr **Par. P2** odpowiada za adres rejestru w urządzeniu. Adres należy dobrać analogicznie do poprzedniego przykładu (np. 00001 dla urządzenia o adresie 2 i rejestrze 00001).

W edytorze makr pojawi się linia: U\$1.0=2:00001(B)

b) Teraz pozostaje nam skonfigurowanie cykli w jakich będzie uruchamiane makro. W drzewie projektu dwukrotnie klikamy na nazwę projektu i wchodzimy w okno **General**. Zaznaczamy pozycję **Time Macro #1** z interwałem czasowym **Time Interval** np. 0.5 sekundy. W nowej zakładce **Time Macro #1** przypisujemy wcześniej zdefiniowane makro do pola **Macro name**. W takiej konfiguracji makro będzie uruchamiane cyklicznie co 0.5 sekundy.

c) Lampkę **Bit Lamp** dodajemy analogicznie jak w punkcie 2.1.2 z tymże do pola **Monitor Address** wpisujemy adres zmiennej lokalnej z pamięci Internal Memory czyli \$U1.0 jak w powyższym przykładzie.

eneral Label	Visibility	
ID: BL0000	Note:	
		Picture Shape 📃 External Label
		Shape
		State: On Off
		Border Color: FG Color:
NO_BE	PR	Pattern:
	Bit	~
Address Type:	12/	

Rysunek 15. Blok Bit Lamp

eneral Setup	
General Custom Time Macro #1	
Application Name: 2000 Model: WOP-2080V (Ethernet) Screen: 1 Screen 1 Delay Time: 0 Screen 1 Delay Time: 0 Screen 1 Delay Time: 0 Default User Level: 8 Default User Level	Battery Backed RAM: 128 KB Flash ROM: Standard Macro Startup Macro Kain Macro Event Macro #1 : Event Macro #2 : Event Macro #3 : Event Macro #4
Display Idle Screen	Event Macro #4 : Time Macro #1 : Time Interval : 0.5 v second Time Macro #2 : Time Macro #3 : Time Macro #4 :
Buzzer Sounding Time: 0.5 🗸 📚 seconds	Print Printer: (None)
Overlapped buttons can be activated in sequence by one touch Note:	
<	
	OK Cancel Help

Rysunek 16. Konfiguracja podstawowych parametrów panelu

2.1.4 Zapis stanów logicznych na przykładzie modułu ADAM-4055

W przypadku modułu ADAM-4055 do zapisu/odczytu stanu wyjść cyfrowych DO0-DO7 wykorzystywane są rejestry z zakresu od 00017 do 00024.

Do zapisu danych binarnych posłużyć mogą dwie kontrolki: **Bit Button B** oraz **Toggle Switch Wstawianie** i otwieranie okna konfiguracyjnego jest analogiczne do punktu 2.1.2



Rysunek 17. Widok kontrolek Bit Button i Toggle Switch

2.1.4.1 Obsługa kontrolki Bit Button

Umieszczamy przycisk Bit Button 🕒 na polu ekranu i wchodzimy w jego ustawienia (klikamy dwukrotnie myszką).

Bit Button 🛛 🔀	
General Label Advanced Visibility	
ID: BB0000 Note:	
Picture Shape	
Shape VFTA: Sunken 💌	
State: On Off Test VFTA	
Color: FG Color: FG Color:	
NO_BDR Pattern:	
Operation Set ON ○ Set OFF ○ Set ON Pulse ○ Set OFF Pulse	
Momentary DN Momentary DFF Invert	
Address Type: Bit	Address Input Keypad
Write Address: 2:00017	Link 1
Monitor Monitor Address identical to Write Address	
Monitor Address: 2:00017	2 💌 : 💌 🚥
	CA123CLR
🗌 ON Macro	D B 4 5 6 BS E 7 8 9 ESC
OK Cancel Help	F/.0ENT

Rysunek 18. Konfiguracja parametrów kontrolki Bit Button

W polu **Operation** wybieramy sposób działania przycisku:

Set ON, Set ON Pulse – ustawia bit Write Address na 1.

Set OFF, Set OFF pulse – ustawia bit Write Address na 0.

Momentary ON – ustawia bit **Write Address** na 1 przez czas trzymania przycisku, po puszczeniu przycisku bit przyjmuje wartość 0.

Momentary OFF – ustawia bit **Write Address** na 0 przez czas trzymania przycisku, po puszczeniu przycisku bit przyjmuje wartość 1.

Invert – ustawia bit Write Address na 1 gdy przycisk jest wyłączony (stan 0) i na odwrót.

W polu **Write Address** należy podać adres do którego będzie przypisany stan bitowy przycisku. W przypadku ADAM-4055 wartość 2:00017 oznacza że zostanie wysterowane wyjście 0 o adresie 00017 (urządzenie posiada adres 2).

2.1.4.2 Obsługa kontrolki Toogle Switch

ON ON

Kontrolka **Toggle Switch** and umożliwia pracę w systemie dwustanowym w którym ON odpowiada stanowi wysokiemu "załączony" oraz OFF niskiemu "Wyłączony". W poniższym przykładzie do kontrolki przypisany został rejestr 00017 powiązany z wyjściem cyfrowym DO0 w module ADAM-4055.

Toggle Switch	
General Label Advanced Visibility ON Macro OFF Macro	
ID: TS0000 Note:	
Picture Shape	
Shape VFTA: Sunken 💌	
State: On Off Test VFTA Border	
Write Address: 2:00017	
Monitor Address identical to Write Address	
Monitor Address: 2:00017	Address Input Keypad 🛛 🛛 🔀
✓ ON Macro	Link: Link 1
	2 🗸 : 🔽 🔽
	CA123CLR
	D B 4 5 6 BS E : 7 8 9 ESC
OK Cancel Help	F/.OENT

Rysunek 19. Konfiguracja parametrów kontrolki Toggle Switch

W przykładzie aktywowano również opcje ON i OFF Macro, pozwalające na wykonanie zaimplementowanych w makrach operacji przy każdej zmianie stanu przycisku. Zaznaczenie pola ON lub OFF Macro uaktywnia zakładkę o tej samej nazwie, w której należy dodać odpowiednie makro w polu Macro name. W naszym przykładzie przycisk został powiązany z wcześniej dodaną lampką Bit Lamp ze zdefiniowanym adresem lokalnym \$U1.0. I tak kolejno, w przypadku zmiany stanu (ON na OFF lub odwrotnie) zmienia się jednocześnie stan naszej lampki. W takiej sytuacji nie ma już dalszej potrzeby korzystania z makra czasowego ponieważ makro jest wywoływanie tylko w momentach zmiany stanu, a nie cyklicznie co zadany w makrze interwał czasowy.

ON Macro - \$U1.0 = 1(B)

OFF Macro - \$U1.0 = 0(B)

Toggle Switch			
General Label Advanced Visibility ON Macro OFF Macro			
Macro name: lamp on 🗸 New			
• \$U1.0=1(B)			
	Pro	operties	×
	\$	U1.0=1(B)	
		Command: P1 = P2	
		Data Type: (B) Bit	•
		Par. P1: \$U1.0	
		Par. P2: 1	
		Operation:	
× 1		Assigns the value of P2 to	3
		Parameters:	
		Type Description	~
	11		
	<		

Rysunek 20. Konfiguracja ON Macro

2.1.5 Odczyt wejść analogowych na przykładzie modułu ADAM-4017+

ADAM-4017+ to moduł wejść analogowych wyposażony w 8 niezależnych kanałów z wejściami napięciowymi i prądowymi o rozdzielczości 16 bitów. Urządzenie wyposażone zostało w interfejs RS-485 i współpracuje z protokołem Modbus RTU. Dane przechowywane są w rejestrach od 40001(kanał AI0) do 40008(kanał AI7).

2.1.5.1 Odczyt bezpośredni rejestru

a) W celu odczytania wartości rejestru bezpośrednio z urządzenia można skorzystać z przykładowej kontrolki **Numeric Display III**.



Rysunek 21. Widok kontrolki Numeric Display

Numeric Display	ŧ		×		
General Advance	ed Visibility		45		
ID: ND0000	Note:]		
	Tra	insparent Background			
	Shap	e External Label			
	Border	Color:			
NO_BE	BG Co	lor:			
Data Type:	16-Bit Unsigned Inte	eger 💌			
Display Type:	16-Bit Unsigned De	cimal 💌	Address	Input Kovpad	
Monitor Address:	40001		Audress	трискеурай	
Font: Font_1		v	Link: Lin	nk 1	*
Text Color:		nt Justification			ताह
Total Digits: 4	Cent	er O Leading Zeros			
Fractional Digits:	0 🗢 🔘 Righ	t O Leading Spaces			
			F		scl
L		Cancel Help			Ï

Rysunek 22. Konfiguracja parametrów kontrolki Numeric Display

Zakładka General:

Shape - klikając na ten przycisk można zdefiniować wygląd pola Numeric Display.

External Label - zaznaczenie pola pozwala na dodanie etykiety tekstowej przed polem **Numeric Display**. Parametry etykiety definiuje się w zakładce **External Label** pojawiającej się po zaznaczeniu tego pola.

Border Color i BG Color - opcje pozwalają określić kolor ramki oraz pola Numeric Display.

Text Color - opcja pozwala wybrać kolor tekstu.

Total Digits - opcja pozwala ustawić liczbę wyświetlanych cyfr.

Fractional Digits – opcja pozwala ustawić wyświetlaną liczbę cyfr po przecinku.

Alignment oraz **Justification** – opcje pozwalają ustawić sposób wyświetlania wartości w polu **Numeric Display**, takie jak wyrównanie tekstu lub niewyświetlanie nieznaczących zer.

b) W zakładce **Advanced** można ustawić opcję dodatkowe, jak np. zmiana koloru czcionki lub tła po przekroczeniu ustalonych poziomów wartości.

c) W zakładce **Visibility** można ustawić opcje dodatkowe, jak np. widoczność pola w zależności od poziomu zalogowanego użytkownika.

d) W zakładce General w polu Data type i Display Type wybieramy 16-Bit Unsigned Integer.

e) W polu **Monitor Address** wpisujemy odpowiadający kanałowi Al0 rejestr 40001 przypisany do urządzenia ADAM-4017+ o adresie 1.

2.1.5.2 Odczyt rejestru przez makro

a) W celu odczytu wartości rejestru przez makro skorzystać należy z makra czasowego (sposób tworzenia i konfiguracji podany w podpunkcie 2.1.3).

b) W makrze wpisujemy odpowiednią linijkę np.:

\$U10=40001

W tym przypadku wartość z rejestru 40017 z urządzenia o adresie 1 jest zapisywana do zmiennej lokalnej o adresie \$U10. Typ zmiennej należy ustalić jako Unsigned Integer.

c) W polu **Monitor Address** kontrolki **Numeric Display** wpisać należy adres zmiennej, która przechowuje odczytaną wartość, w tym przypadku \$U10. Można również skorzystać z kontrolek innych typów (np. **Bar Graph Meter** (Meter) postępowanie jest analogiczne do wyżej opisanych przypadków.

d) Odpowiednio skonfigurować makra czasowe w drzewie projektu – punkt 2.1.3 c).

3. Elementy graficzne

3.1 Blok Bar Graph

Z paska narzędziowego dodajemy bloczek **Bar Graph L** i dodajemy go na ekran.



Rysunek 23. Widok bloku Bar Graph

Klikamy dwukrotnie prawym klawiszem myszy i wchodzimy w ustawienia bloczku.

a) W zakładce głównej General wyróżniamy pola:

Shape - pozwala na zmianę wyglądu bloczki.

External label – pozwala na dodanie etykiety tekstowej w bloczku. W momencie aktywacji pojawia się odpowiednia zakładka umożliwiająca dalszą konfigurację.

Border Color i BG Color – odpowiadają za kolor ramki oraz pola Bar Graph.

Direction – określa kierunek zmiany wypełnienia.

ID: BG0000 Note: External Label Border Color:	
Shape External Label Border Color:	
BG Color:	
Direction Upward Downward Leftward Rightward	
Data type: 16-Bit Unsigned Integer	
Dynamic Range	
Min.: 0 Max.: 65535 Link: Link 1	~
Bipolar Bar Bar Pattern:	
Bar FG Color: Bar BG Color: CA 1 2 3 CLR D B 4 5 6 BS	
E : 7 8 9 ESC OK Carcel Help F / . 0 ENT	

Rysunek 24. Konfiguracja i dodawanie rejestru do bloku Bar Graph

Data Typ – definiuje typ zmiennej.

Monitor Address – w tym polu należy wpisać adres urządzenia i rejestru.

Min, Max – określają minimalną i maksymalną wartość słupka bargraphu. Zaznaczenie pola Dynamic Range umożliwia zdefiniowanie tych wartość poprzez przypisaną zmienną.

Bipolar Bar - pozwala na ustawienie skali zawierającej wartości ujemne i dodatnie, **Middle point** określa środek skali.

Bar Pattern, Bar FG Color, Bar BG Color – pozwalają zdefiniować wygląd słupka bargraphu.

b) Zakładka Scale umożliwia zdefiniowanie parametrów skali wyświetlanej obok słupka.

c) Zakładki **F. Marker, B. Marker** definiują markery w postaci strzałek, które są dodawane do przy osi słupka bargraph. We właściwościach można zdefiniować do 24 strzałek oraz określić ich kształt oraz kolor zmiany w przypadku przekroczenia przypisanej wartości.

d) W zakładce **Advanced** można uaktywnić oraz zmienić zakres wyświetlanych danych. Pole **Percentagle Display** uaktywnia procentowy wskaźnik wypełniania bloczku.

e) W zakładce **Visibility** można ustalić takie parametry jak położenie, uaktywnić widoczność w zależności od stanu przypisanego rejestru oraz określić dostęp użytkowników.

W podobny sposób należy skonfigurować bloczki typu Meter 🊈

3.2 Powiązanie bloku wskaźnika Meter z suwakiem typu Slider

Na ekranie dodajemy bloczek typu **Meter** $\stackrel{\text{res}}{\longrightarrow}$, którego konfiguracja odbywa się w podobny sposób jak w bloku Bar Graph w punkcie 3.1. W polu Monitor Address wstawiamy zmienną wewnętrzną Internal Memory \$U100, która powiązana zostanie ze zmienną z bloku **Slider**. Poprzez pole **Picture Shape** do bloku przypisujemy obraz graficzny, pobierany z dostępnej bazy programu lub z własnych zasobów (aby dodać poniższy rysunek należy zaznaczyć Picture Shape, nacisnąć przycisk Import from Liblary, wybrać Picture Liblary file jako Meters i zaznaczyć meter10). Położenie rysunku względem zarysu można ustalić w grupie pól **Swing Adjustment**.

	Meter 🔀
	General Range Scale Visibility
	ID: M0000 Note:
	Picture Shape
	meter10 CBB
	Shape Swing
	Border Color:
	BG Color:
	NO_BDR
	Direction: 💿 Clockwise
	Data Type: 16-Bit Unsigned Integer 💌
	Monitor Address: SU100
	Dynamic Range
Company of the local division of the local d	Min.: 0 Max.: 10
	Needle Color:
	Swing Adjustment
	Radius: -11 🗢 Angle: 0 📚
	Center X: 0. 📚 Center Y: 1 📚
	OK Cancel Help

Rysunek 25. Widok i konfiguracja bloku Meter

Następnie na ekranie wstawiamy blok **Slider Switch** i w polu **Monitor Adres** dodajemy zmienną wewnętrzną \$U100. Ustawiamy **Direction** jako Rightward oraz podajemy zakres zmiennej od 0 do 10. W zakładce **Scale** zaznaczamy pole Display. Po skompilowaniu i uruchomieniu wskaźnik Meter będzie liniowo reagował na zmiany położenia suwaka Slider.

Slide Switch General Scale Advanced Visibility ID: SW0000 Note: Shape External Label
Border Color: BG Color: NO_BDR
Direction Upward Downward Ceftward Rightward Data Type: 16-Bit Unsigned Integer Write Address: \$U100 Dynamic Range Minimum: 0 Knob Border Color: BG Color: Help

Rysunek 26. Konfiguracja bloku Slider Switch

3.3 Powiązanie bloku graficznego Picture Dispaly z blokiem Bar Graph i suwakiem Slider

Na ekranie wstawiamy blok **Picture Display**, **Slider Switch** oraz **Bar Graph**. Klikamy dwukrotnie na blok Picture Display i konfigurujemy:

State Type - typ zmiennych określających widoczność bloku.

Monitor Address - zmienna kontrolująca widoczność bloku (powiązana z polem State Type).

Picture - pole umożliwia wybór elementów graficznych z biblioteki programu lub z własnych zasobów zdjęć z rozszerzeniem bmp, gif lub wfm. W naszym przykładzie wybieramy element z rodziny Tanks o nazwie Storage Facility 4.

Transparent - jest opcja pozwalająca na wybranie koloru, który będzie traktowany jako przezroczysty – w naszym przypadku wybieramy biały.

cture Display 🛛 🛛 🔀	Picture Display
ieneral Picture Visibility Testing	General Picture Visibility Testing
ID: PD0000 Note:	S# Picture Name Border Spacing: 0 • 0 Storage Facility 4 Attribute • • 1 Picture: • <t< th=""></t<>
State Type Bit Value LSB Animation Data Type: Monitor Address: Total States: 2	Move Up Move Down Flip/Rotate: 0° V Tone Fit to Object Position 0
OK Cancel Help	OK Cancel Help

Rysunek 27. Konfiguracja bloku Picture Display

Następnie w utworzonym bloku **Bar Graph** wybieramy **Drection** jako **Upward** i podajemy zmienną lokalną \$U100, odpowiadającą za stan wypełnienia. Zakres ustalamy od 0 do 10. Tak skonfigurowany blok przesuwamy na blok Picture Display i dostosowujemy jego wymiary. Jeżeli chcemy przesunąć któryś z bloków na wierzch należy kliknąć prawym klawiszem myszy na bloku i wybrać z menu kontekstowego opcję **Bing to Top** lub **Bing to Forward**.

Teraz pozostaje już tylko dodać blok Slider ze zmienną lokalną \$U100 i uruchomić program.



Rysunek 28. Widok połączonych bloków Picture Display z Bar Graph

4. Dodawanie Tagów

Innym sposobem adresacji urządzeń są wbudowane w programie znaczniki Tag. Są one o tyle korzystne że zdefiniowany Link oraz adres urządzenia kojarzone są bezpośrednio z przypisaną programowo zmienną. W dalszym procesie tworzenia aplikacji zmienna ta może być wykorzystywana wielokrotnie bez konieczność każdorazowego definiowania adresu urządzenia.

W pierwszym kroku w drzewie projektu klikamy prawym klawiszem myszy na znacznik **Tags** i otwieramy listę dostępnych znaczników. Dwukrotnie klikamy na interesujące nas połączenie (u nas Link 1) i w polu **Name** wprowadzamy nazwę zmiennej. W polu **Data Type** określamy typ zmiennej a w **Address** podajemy odpowiedni adres urządzenia.



Rysunek 29. Widok tabeli Tagów

W poniższym przykładzie dodajemy dwie zmienne połączone z wejściem i wyjściem modułu ADAM-4055 o adresie 2. Zmienna ADAM-4055DI0 powiązana zostanie z wejściem DI0 o rejestrze 00001 a zmienna ADAM-4055DO0 z wyjściem DO0 o rejestrze 00017.

AP_1								
Screen 1 (#1) Tags								
Internal Memory		Name	Data Type	Addr	Length	Scan Rate	Comment	
	1	ADAM-4055DIO	Bit	💽 2:00)	n/a	Normal		
	2	Address Input	Keypad	? 🗙	n/a	Normal]
	3							1
		Link: Link 1		~				
		2 💌 :	✓ Щ	001				
		CA	1230	ïLR				
		DB	4 5 6 8	3S				
		E:	789E	SC				
		EZ	. 0 EN1					
					1			
1								

Rysunek 30. Definiowanie adresu zmiennej Tags

5. Receptury

Receptury pozwalają na przechowywanie danych w zarezerwowanym obszarze pamięci panelu operatorskiego. Odczyt oraz zapis danych dokonywany jest ze wskazanych na etapie konfiguracji adresów i może być rozszerzany o dowolne urządzenia podłączone do WOPa np. sterowniki PLC lub moduły ADAM.

W drzewie projektu klikamy prawym klawiszem myszy na Recipes i dodajemy nową recepturę. Następnie dwukrotni klikamy na powstałą zmienną i wchodzimy jej ustawienia **General**. W poniższym przykładzie do receptury przypisano zmienną Tags A6017 powiązaną z modułem wejść analogowych ADAM-6017.



Rysunek 31. Położenie Recptur w drzewie projektu

cipe Block		
General Data Item		
Name: Ricept6017	ID: 0	
Recipe Size: 1 😂	words Number Of Recipes: 1	
Memory Required: 1	words	
Write Recipe To PLC		
Write Address: 1\A6	3017	
Notification		
Read Recipe From PL	.C 🛛 🗹 Read Address Identical To Write Address	
Read Address: 1\46	i017	
Notification		
Read/Write Size: (defau	ult) 👽 words	
Reverse the order of t	the high word and low word of 32-bit data	
Recipe Memory		
Bit Address Range:	\$R0:0.0 - \$R0:0.f	
Word Address Range:	\$R0:0 - \$R0:0	
Current Recipe		
Bit Address Range:	\$CR0:0.0 - \$CR0:0.f	
Word Address Range:	\$CR0:0 - \$CR0:0	
Current Recipe Number	r Register: \$RN0	
Need space in flash B	ROM to save backup	

Rysunek 32. Zakładka General w Recipe Block

W zakładce General wyróżniamy pola:

Name - nazwa recepty.

ID - identyfikator recepty.

Recipe size - wielkość recepty (0 - 1023).

Number of recipes - liczba recept (0 - 65535).

Memory Required - automatycznie przypisana pamięć do recepty.

Write Recipe To PLC - adres rejestru sterownika PLC do którego ma być zapisana recepta.

Read Recipe From PLC - adres rejestru z którego ma być odczytana receptura (będzie taki sam co rejestr

zapisywany w przypadku zaznaczenia pola Read Address Identical To Write Address)

Read Address - adres rejestru sterownika PLC, z którego ma być odczytana recepta.

Recipe Memory - obszar pamięci zarezerwowany dla receptury.

Current Recipe - obszar pamięci aktualnej receptury.

Need space In flash ROM to save backup – opcja ta umożliwia zapis danych z receptury do pamięci zewnętrznej poprzez port USB

Zapis receptury może zostać zrealizowany w połączeniu z blokiem funkcyjnym **Function Button** II z wybraną funkcją **Save Recipe Data to Flash ROM** lub **Save Recipe Data to File** (zapis do zdefiniowanego pliku csv lub txt w pamięci zewnętrznej na USB np. pendrive).

Odczyt receptury odbywa się przy pomocy bloku funkcyjnego z funkcją **Read Recipe from Controller** (każdorazowy odczyt ze wszystkich receptur po kliknięciu przycisku).

W zakładce Data Item pojawiają się kolejne przebiegi uzależnione od wielkości recepty Ricept Size.

Addr. Name CR0:0 R6017 Address: CR0:0 Name: B6017 Language: Language 1 Data Type: 16-Bit Unsigned Integer Display Type: 16-Bit Unsigned Decimal Total Digits: 5 Fractional Digits: 0 Scaling Gain: 1 Offset: 0 Range Check. Min.: 0			
Min.:	Addr. CR0:0	Name R6017	Address: CR0:0 Name: R6017 Language: Language 1 Data Type: 16-Bit Unsigned Integer Display Type: 16-Bit Unsigned Decimal Total Digits: 5 Fractional Digits: 0 Scaling Gain: 1 Offset: 0 Range Check
Max: 9999	Alt+Lin: Mo	e item un Alt+Down Move item down	Min.: 0 Max.: 9999

Rysunek 33. Zakładka Data Item w Recipe Block

W zakładce Data Item wyróżniamy pola:

Name - nazwa przebiegu.
Language - wybór języka.
Data Type - typ zmiennej.
Display Type - typ zmiennej wyświetlanej.
Total Digits - liczba wyświetlanych cyfr.
Fractional Digits - Liczba wyświetlanych cyfr po przecinku.
Scaling - skalowanie według równania Y=aX+b gdzie Y to wartość wyświetlana, X to wartość odczytana, a to wzmocnienie, b to Offset.

ID: FB0003 Note:	Visibility
	Picture Shape
	Shape
	Test VFTA
	Border EG Color
NO_BDR	Pattern:
Operation: Save Becine Dal	ta To File
Ele Eutonsion:	
Recipe Block:) LOV (01X1
Ricept6017 (0)	~
Default Filename:	
test.txt	Filename Selectable

Rysunek 34. Wykorzystanie Bloku Function Button umożliwiającego zapis receptury do zdefiniowanego pliku.

6. Logowanie danych

6.1 Dodawanie i konfiguracja

Moduł **Data Logger** umożliwia wprowadzanie danych do pamięci wewnętrznej panelu operatorskiego. Aby skorzystać z tej funkcji w pierwszej kolejności w drzewie projektu wyszukujemy pole **Data Logger**. Klikając prawym klawiszem myszy dodajemy moduł do archiwizacji danych do projektu – wybieramy **Add Data Logger**. Następnie dwukrotne klikając myszą wchodzimy do ustawień modułu. W poniższym przykładzie do modułu przypisano zmienną Tags A6017 powiązaną z modułem wejść analogowych ADAM-6017.



Rysunek 35. Dodawanie modułu Data Loggera do projektu

Name: Loggers6017 ID: 0 💌	
Use battery backed RAM Sample Size: 1 vords Number of Samples: 300 vords Logging Buffer Size: 1800 words Latest Sampled Data Word Address: \$L0:0 - \$L0:0 Read Address: 1\46017	Sample Full Processing ♥ Stop Sampling ♥ Notify Bit: Full Limit: 90< % External Control ♥ Clear Buffer Trigger Bit: ♥ Enable Sampling Enabling Bit: ♥ Save Data to File File Type: .CSV ●.TXT .LDF
 Triggered Clocked Timed (sub-second) Load From .LDF File 	Operation Type: Create Append or Create Time to Save: Every hour on the hour Filename: adamek.txt Filename Suffix: _YYMMDD_hhmmss

Rysunek 36. Ustawienia modułu Data Logger w zakładce General

W zakładce General wyróżniamy pola:

Name - nazwa modułu.

ID - identyfikator.

Use battery backed RAM - zapis danych do pamięci z podtrzymaniem bateryjnym.

Sample Size - wielkość próbki (jedna próbka ma 16 bitów).

Number of Samples - ilość próbek.

Logging Buffer Size - ilość pamięci potrzebnej dla danych archiwalnych (obliczana automatycznie).

Word Address - obszar pamięci przydzielany automatycznie.

Read address - adres pamięci z którego będą pobierane dane.

Sampling Method:

Timed - próbkowanie w określonych odstępach czasu (w sekundach od 0 do 65535).

Triggered - próbkowanie wyzwalane przez bit Trigger Bit zawarty w module Command & Status

Clocked - wyzwalanie przez zegar systemowy (w minutach od 1 do 60).

Timed - próbkowanie z określonym interwałem czasowym (w sekundach od 0.1 do 0.9s).

Sample Full Processing:

Stop Sampling - zatrzymanie próbkowania w przypadku przepełnienia bufora danych.

Notify - w przypadku przepełniana bufora zmienia się stan w określonym bicie lub zmiennej. **External Control:**

Clear Buffet - czyszczenie bufora danych wyzwalane przez bit Trigger Bit zawarty w module Command & Status Enable sampling - wyzwolenie próbkowania przez bit Enabling Bit zawarty w module Command & Status Save to file - zapisuje dane do wybranego pliku w zdefiniowanych cyklach czasowych.

Addr. L0:0	Name	
Nulls: Me		Address: L0:0 Name: L6017 Language: Language 1 • • • Data Type: 16-Bit Unsigned Integer • Display Type: 16-Bit Unsigned Decimal • Total Digits: 5 • Fractional Digits: 0 • Scaling Gain: 1 Offset: 0

Rysunek 37. Ustawienia modułu Data Logger w zakładce Data Item

W zakładce Data Item wyróżniamy pola:

Name - nazwa przebiegu.
Language - wybór języka.
Data Type - typ zmiennej.
Display Type - typ zmiennej wyświetlanej.
Total Digits - liczba wyświetlanych cyfr.
Fractional Digits - Liczba wyświetlanych cyfr po przecinku.
Scaling - skalowanie według równania Y=aX+b gdzie Y to wartość wyświetlana, X to wartość odczytana, a to wzmocnienie, b to Offset.

6.2 Tabele i wykresy wykorzystujące dane typu Data Logger

Do obsługi modułu Data Logger można wykorzystać bloki Historic Data Table 🕮 oraz Historic Trend Graph 🗠

6.2.1 Historic Data Table

Blok Historic Data Table pozwala na wyświetlanie danych w tabeli ze zdefiniowanego moduły Data Loggers z

określonym interwałem czasowym. W pierwszej kolejności wstawiamy blok 🕮 do naszej aplikacji i wchodzimy w ustawienia poprzez dwukrotne kliknięcie myszą. W polu **Assiociated Data Loggers** wybieramy odpowiedni moduł Data Loggers. Po uruchomieniu na ekranie uzyskamy cykliczny odczyt z wejścia analogowego AIO z modułu ADAM-6017, przy pomocy protokołu Modnus TCP i sieci Ethernet.

			Historic Data Table	
			General Data Item Visibility	
			ID: HDD00000 Note: Shape Border Color:	- test
			BG Color: NO_BDR	
			Associated Data Logger: Loggers6017 (0)	
			Data Source: 💿 Data Logger 🔿 File	
			☐ Title Data	
			Language: Language 1 Display Oldest Data First	
			Font Font Font Font	
			Color: Default Color: Default Color:	
			Date: Date Set Default Color To All Data Items	
			Time: Time Time/Date Display	
			Background Color: DD/MM/YY V	
Date	Time	L6017	Grid	
31/12/09	23:59	99999	V Horizontal Color:	
a 31/12/09	23:59	99999		
31/12/09	23:59	99999	Line Spacing: 2 🗘 Item Spacing: 2 🗘	
31/12/09	23:59	99999	OK Cancel Help	j

Rysunek 38. Ustawienia bloku Historic Data Table

6.2.2 Historic Trend Graph

Blok Historic Trend Graph pozwala na wyświetlanie danych w postaci przebiegu ze zdefiniowanego moduły Data

Loggers. W pierwszej kolejności wstawiamy blok do naszej aplikacji i wchodzimy w ustawienia poprzez dwukrotne kliknięcie myszą. W polu Assiociated Data Loggers wybieramy odpowiedni moduł Data Loggers. W zakładkach **Curve** i **Axis** możemy zdefiniować zakres odczytu danych oraz wygląd przebiegu. Po uruchomieniu na ekranie uzyskamy cykliczny odczyt z wejścia analogowego AIO z modułu ADAM-6017 przy wykorzystaniu protokołu Modnus TCP i sieci Ethernet.

 listoric Trend Graph	\mathbf{X}
Ib: If the display of the second s	
Dynamic Range Support Zoom Display Relative Time	OK Cancel Help
0. Ustavija da blaku Uštata da Tara d Corak	

Rysunek 39. Ustawienia bloku Historic Trend Graph

6.2.3 Zapis danych typu Data Logger do pliku tekstowego

Aby zapisać dane typu Data Logger należy skorzystać z bloku funkcyjnego **Function Button** i wybrać funkcję **Save Logged Data To File**. W ustawieniach należy określić odpowiedni Data Logger, typ pliku tekstowego oraz podać jego nazwę. Dane będą zapisywane do pliku w pamięci zewnętrznej (np. Pendrive) przy każdorazowym aktywowaniu przycisku Function Button.

	Function Button
	General Label Advanced Visibility ID: FB0000 Note:
	□ Picture Shape Shape VFTA: Sunken Test VFTA Border □ Color: □ Pattern: □ BG Color: □ Pattern: □ BG Color: □ File Extension: ○ CSV Operation: □ Save Logged Data To File ▼ File Extension: ○ CSV Operation: □ Coperation: □ Coperation: ○ Coperation: □ Coperation: □ Coperation: ○ Coperation: ○ File Extension: ○ Coperation: □ Coperation: □ Coperation: □
	Loggers6017 (0)
	Default Filename: logged.txt Filename Selectable
Save Data Loggeers	OK Cancel Help

Rysunek 40. Blok funkcyjny umożliwiający zapis danych z modułu Data Logger do pliku tekstowego

7. Uruchamianie i testowanie

7.1 Kompilacja i wgrywanie projektu do sterownika

Procesu kompilacji można dokonać z poziomu paska skrótów naciskając guzik **Compile b** lub z poziomu menu projektu wchodząc w **Panel->Compile**. W przypadku wystąpienia błędów w czasie kompilacji odpowiednie adnotacje pojawią się w polu **Build List**. Po prawidłowym skompilowaniu należy wgrać projekt do sterownika

naciskając guzik **Download** ilub wybrać z menu projektu **Panel->Download**. Tutaj w ustawieniach **Link Settings** wybieramy port COM, w przypadku transmisji po porcie szeregowym, lub port Ethernet podając adres IP panelu operatorskiego.





Rysunek 42. Download projektu

7.2 Testowanie i symulacja

Środowisko WebOP Designer udostępnia również bardzo wygodne funkcje symulacyjne, które mogą być użyte przed załadowaniem projektu do paneli z serii WOP-2000. Wyróżniamy tu dwa tryby symulacyjne Online i Offline.

Aby wejść w tryb symulacyjny **Offline** należy uaktywnić przycisk 😥 lub wybrać z menu programu **Tool->Run Offline Symulation**. W tym trybie symulacja będzie odbywała się bez komunikacji ze zdefiniowanymi połączeniami typu Link. W przypadku wyboru symulacji **Online** (należy wybrać z menu **Tool->Run Online Symulation**) możliwe jest również przetestowanie połączeń z urządzeniami zewnętrznymi - pełna symulacja pracy tak jak na panelu operatorskim.

🗷 PanelSim	
General Link 1 General Link 1 Panel: test_AP_1 Model: WOP-2070V (Ethernet) Version: 1.2.85 Simulation Off-line Off-line On-line	Open Run Exit

Rysunek 43. Symulacja w trybie Offline



Rysunek 44. Widok aplikacji w trybie symulacyjnym

8. Poziomy dostępu do aplikacji panelu

WebOP Designer definiuje 9 (0-8) poziomów dostępu użytkowników, zróżnicowanych pod względem przypisanych uprawnień takich jak dostęp do bloków czy ekranów. Do każdego z poziomów przypisywane jest unikatowe hasło o różnym rozmiarze (poziom 0 nie ma zdefiniowanego hasła i ma najmniejsze możliwości kontroli uprawnień). Aby zmienić domyślnie zdefiniowane hasło należy wejść w menu programu **Panel->Passwords**. Zaznaczenie pola **Automatic login for operation requiring a higher user level** spowoduje że uaktywnienie bloku o wyższym poziomie dostępu automatyczne uruchomi okno autoryzacji **Password Entry**.

			Passwords			? 🛛
			User Level	Password	Comment	ОК
			1	1		Cancel
			2	22		
			3	333		
Password En	try		4	4444		
	***	C1.	5	55555		
		Cir	6	666666		
1	\mathbf{a}	\sim	7	777777		
		3	8	88888888		
			9 🗹		Use developer password	
4	5	6				
			Automatic login for operations requiring a higher user level			
7	8	9	Login Trigger Bit: None 💌			
			Logout Trigger Bit: None 💌			
Esc	0	Ent	Login Timed	out: 60	seconds	

Rysunek 45. Hasła przypisane do poszczególnych poziomów użytkownika oraz pole Password Entry

Ustawienia oraz aktywację poziomów dokonuje się z poziomu menu projektu **Panel->General Setup**. Zaznaczając pole **Default User Level** ustalamy domyślnie poziom projektu. W przypadku zaznaczenia pola **Login Required** przy starcie aplikacji wymagana będzie autoryzacja poprzez podanie przypisanego danemu poziomowi hasła.

neral Custom Time Macro #1	
pplication Name: AR_1	
fodel: WDP-2070V (Ethernet)	Battery Backed RAM: 128 KB
Start Up	
Screen: 1 Screen 1	Startup Macro
Delay Time: 0 🔿 second(s) 👘 Display Countdown	Main Macro
	. Event Macro #1 :
Language 1	Event Macro #2 :
🗌 Login Required 🛛 Default User Level: 🛛 😒 🤤	Event Macro #3:
Idle Processing	Event Macro #4 :
Display Idle Screen	✓ Time Macro #1 : Time Interval : 0.5 v second
	Time Macro #2
Screen Saver Time: 20 😴 minutes	
	Print
Buzzer Sounding Time: 0.5 💉 🤤 seconds	Printer: (None)
Overlapped buttons can be activated in sequence by one tou	ch
lote:	
	2
<u><</u>	2

Rysunek 46. Konfiguracja poziomu użytkownika w projekcie

Jeżeli chcemy zróżnicować uprawnienia danym użytkownikom należy odpowiednio wcześniej skonfigurować poszczególne bloki w ich ustawieniach w zakładce **Advanced** (o ile jest możliwa) oraz **Visibility**. Aby zmienić prawa dostępu do danego bloku wchodzimy w ustawienia **Advanced** i zaznaczamy opcję **Enable by User Level**, a w polu **Lowest Available User Level** podajemy poziom dostępu do bloku. W przypadku, gdy zalogowany użytkownik ma niższe prawa niż zdefiniowany blok, po jego uaktywnianiu konieczne będzie podanie ponownie hasła o wyższym priorytecie. W każdej chwili można zmienić poziom przy pomocy bloku funkcyjnego z funkcją **Log In**.

W każdym bloku możliwe jest ustalenie widoczności w zależności od poziomu użytkownika. Konfiguracja odbywa się w ustawieniach bloku w zakładce **Visibility**. Poprzez zaznaczenie pola **Controlled by User Level** i wybranie liczby w polu **Lowest Available User Level** ustalamy najniższy poziom zezwalajmy na widoczność bloku.

Toggle Switch	Toggle Switch
General Label Advanced Visibility	General Label Advanced Visibility
✓ Touch Operation Control Image: Touch Operation Contrelevel </td <td>Invisible Controlled by Bit</td>	Invisible Controlled by Bit
Enabled by User Level Lowest Enabling User Level:	Controlled by User Level Lowest Visible User Level: 2
Minimum Hold Time: 0 v second(s)	Dimension Left: 52 🗢 Width: 100 🗢 Top: 88 🗢 Height: 50 🜩 Redraw
Operation Logging	
OK Cancel Help	OK Cancel Help
Off	οτέδη ω ριοκη τοββιε 2Μιττμ



Rysunek 48. Widok zablokowanego bloku

9. Languages

Jedną z podstawowych funkcji WebOP Designer jest możliwość definiowania różnych języków w ramach jednego projektu. Jest to o tyle wygodne, że pozwala na szybkie przetłumaczenie wszystkich słów występujących w danej aplikacji, przy pomocy jednego przycisku funkcyjnego. I tak w jednym projekcie można utworzyć do 10 różnych języków z przypisaną do niego unikalną bazą słow.

Po utworzeniu projektu domyślnie przypisany zostaje tylko język angielski. Aby dodać kolejne języki należy w drzewie projektu wybrać **Languages**, podać ilość języków w polu **Number of languages** i zdefiniować je w polu **Character Set**. Aby ustawić style danego języka należy wejść w **Font Templates** w drzewie projektu.

🖬 PM Designer - C:\aaa\test.pm	2-[Screen 1 (#1)]	
File Edit View Screen Draw	Object Project Panel Tools Wir	ndow <u>H</u> elp
, D 🗳 🛛 🕺 b 🖻 -	? ?	Q Q, 0# 0h 🛛 🖓 0 🖾 (
• B W 🖽 🛅 FI 🗄	-0- 💬 🤹 🗧 (III de	🖽 🖽 😼 🗣 🤮 🔛 🗃 🗃
Project Manager Text Database		# × AP_1
	Languages Number of languages: 2 V Language Name 1 English 2 Polish	Character Set Process Default Language
다 그러고 Loggers 다일 Operation Logging 관 อ Macros	Import Export	OK Cancel

Rysunek 49. Konfiguracja dostępnych języków w projekcie.



Rysunek 50. Przypisanie stylu do danego języka.

Tłumaczenie słów odbywa się z poziomy **Text Database** umieszczonym w drzewie projektu. W ustawieniach klikamy **Export Panel Application Text** i wprowadzamy kolejne tłumaczenia słów występujących w naszym projekcie. Po przetłumaczeniu klikamy **Import Panel Application Text** co spowoduje aktualizacje językowe.



Rysunek 51. Tłumaczenie słów występujących w projekcie na język Polski

Project Mar	nager	Text Da	atabase			₽×	
Table A		~	5 🖉 🕓 T) <u>.</u>			Scr
String ID		Englis	h		Polish	^	
A00004		Off			Włącz	_	' <u>E</u>
A00003		On			Wyłacz		
A00002		Off			Włącz		
A00001		On			Wyłacz		
A20000		Englis	n		Angielski		
A19999		Polish			Polski		
A19998						6	
A19997	lmp	ort Pan	el Applio	ation T	ext	2	' XI
A19996							
A19995	_			T			
A19994	Fro	m/Table:		Table A	1		~
A19993							
A19992	To//	Applicatio	n:	AP_1			~
A19991							
A19990	Ref	erence La	anguage:	English			~
A19989							
A19988	Sele	ection:					
A19987							
A19986	I	mport	Languag	e			
A19985		1	Polish				
A19984							
A19983							_
A19982							
A19981							
A19980							
A19979							
A19978							_
A19977							
A19976							
A19975	<						>
A19974							
A19973							
A10072		_		_			
		C	Import		Cancel		
uild List							
and cloc							

Rysunek 52. Import tłumaczeń do bloków

W naszym przykładzie skorzystamy już z wcześniej dodanych bloków Bit Lamp i Toggle Switch . Klikamy dwukrotnie na blok Bit Lamp i wchodzimy w zakładkę **Label**. Tam możemy wprowadzić odpowiedni tekst pojawiający się na bloku w zależności stanu. W zależności od wybranego w polu **Language** języka możemy wprowadzić odpowiednie tłumaczenia oraz styl języka (my zrobiliśmy to już wcześniej z poziomu **Text Database**).

Podobnie postępujemy w przypadku bloku Toggle Switch. Następnie dodajemy dwa bloki funkcyjne **D** i w polu **Operation** wybieramy **Change language**. W jednym ustawiamy zdefiniowany język Language jako Polish a w drugim English. Po skompilowaniu i uruchomieniu w naszej aplikacji będą zmieniały się tłumaczenia w zależności od wybranej wersji językowej przycisków funkcyjnych.

Lamp 🛛 🛛	
Label Visibility Language: English Source of the state of the	
OFF Text ON Text OFF Picture ON Picture	
Off 🔄 🔳	
Font Font_1	
Color: Blink: No	Włącz
Char. Spacing: 0 🗢 Line Spacing: 0 🗢	117
Position	vy tącz
Shape BG Color: O O O Copy Attributes to	
ON State	Angielski
	Polski
OK Cancel Help	

Rysunek 53. Konfiguracja językowa w bloku Bit Lamp