Spis treści

1	Opis	Opis zagadnienia omawianego w dokumencie2					
2	Kom	Komunikacja Modbus TCP					
3	Wstę	p do nowego projektu	4				
	3.1	Nowy projekt	3				
	3.2	Dodanie nowego urządzenia	5				
4	Mode	bus Serwer	6				
	4.1	Konfiguracja dwóch połączeń w jednym sterowniku	6				
	4.2	Wyjaśnienie parametrów wejściowych "MB_SERVER"	8				
5	Mod	bus	9				
	5.1	Konfiguracja dwóch połączeń w jednym sterowniku "MB_CLIEN	T"9				
	5.2	Wyjaśnienie parametrów wejściowych "MB_CLIENT"	13				
6	Watc	h tables	13				
7	Zakro	es adresów	16				
8	Kody	warunkowe	17				
	8.1	Wartość "STATUS" dla "MB_SERVER"	17				
	8.2	Wartość "STATUS" dla "MB_CLIENT"	17				

1 Opis zagadnienia omawianego w dokumencie

W tym dokumencie opisano sposób uruchomienia komunikacji w protokole modbus TCP dla sterowników S7-1200.

Minimalna konfiguracja składa się z dowolnego CPU S7-1200.

Wykaz urządzeń:

Urządzenie	Liczba	Numer katalogowy
Sterownik SIMATIC S7-1200, model	1	6ES7 214-1AE30-0XB0
CPU 1214C DC/DC/DC		
Sterownik SIMATIC S7-1200, model	1	6ES7 212-18D30-0XB0
CPU 1212C AC/DC/RLY		

Wykaz oprogramowania:

Oprogramowanie	Liczba	Numer katalogowy
STEP 7 Basic v11	1	6ES7 822-0AA01-0YA0

Dodatkowe akcesoria:

Urządzenie	Liczba	Numer katalogowy
Kabel SIMATIC NET PROFIBUS	1	6XV 1830-0EH10
Wtyczka 90 stopni PROFIBUS	2	6ES7 972-0BB52-0XA0
Switch ethernetowy	1	6GK7277-1AA10-0AA0

2 Komunikacja Modbus TCP

Modbus TCP (Transmission Control Protocol) jest standardowym protokołem komunikacyjnym wykorzystującym interfejs PROFINET w sterowniku dla komunikacji TCP/IP. Nie są potrzebne żadne dodatkowe moduły komunikacyjne.

Modbus TCP bazuje na połączeniach Open User Communication (OUC) jako protokół komunikacyjny Modbus. Można skonfigurować kilka połączeń klient-serwer, oprócz połączenia między S7-1200 i STEP 7. Możliwa jest jednoczesna obsługa różnych typów połączeń (S7-connection, HMI-connection...), których liczba jest ściśle określona (patrz FAQ:36932465 lub manual S7-1200).

Każde połączenie "MB_SERVER" musi posiadać przypisany unikalny blok danych typu instancje DB oraz numer portu IP. Możliwe jest tylko jedno połączenie na każdy port. "MB_SERVER" musi zostać wywołany indywidualnie dla każdego połączenia.

Uwaga:

Modbus TCP może poprawnie pracować tylko ze sterownikiem z firmware'm w wersji v2.1 i oprogramowaniem STEP 7 V11 SP1 lub wyższej. Próba wywołania instrukcji Modbus na sterowniku ze starszą wersją firmware'u spowoduje błąd.

Klient (master) Modbusa TCP musi kontrolować połączenie klient-serwer za pomocą parametru DISCONNECT. Podstawowe akcje klienta Modbusa wyszczególniono poniżej:

1. Inicjalizacja połączenia do poszczególnego serwera (slave'a) poprzez adres IP i numer portu IP.

2. Inicjalizacja klienta do transmisji wiadomości Modbus i otrzymywania odpowiedzi od serwera.

3. W razie potrzeby inicjalizacja odłączenia klienta i serwera w celu połączenia z innym serwerem.

Dostępne instrukcje Modbus TCP

- MB_CLIENT: tworzy połączenie TCP klient-serwer, wysyła polecenia, otrzymuje odpowiedzi i kontroluje przerwanie połączenia od serwera.

- MB_SERVER: łączy się z klientem Modbusa TCP na żądanie, otrzymuje wiadomości Modbus i wysyła odpowiedź.

3 Wstęp do nowego projektu

3.1 Nowy projekt

W celu utworzenia nowego projektu należy uruchomić:

Totally Integrated Automation Portal (TIA), a następnie przejść do **Project view.**



Tworzenie nowego projektu można rozpocząć wybierając **Project** > **New.** Należy nadać nazwę projektu **np. "FAQ_Modbus_TCP"**, żeby zakończyć tworzenie projektu należy kliknąć **Create.**

	AL MODBULA				
	Siemens - MODBUS				
	Project Edit View Inse	rt Online Options	Tools Wine	dow Help	
	📑 New	Ctrl+N	्रांक एम ह	Go online 🖉 G	o offline 🎝 🌆 🖪 📉 🖂 🔲
	Upen	Ctri+U	and the second second		
	Migrate project				
	Close	Ctrl+W		Create a new proj	ect. X
	📮 Save	Ctrl+S	1		LIADRIE
	Save as	Ctrl+Shift+S		Project name:	C/Decuments and Settings/Administrated/AuDecuments'
	Delete project		_	Path:	Nadrona
	Ellistic Card Baader		_	Comment.	inducted and a second s
	SIMATIC Card Keader	· · ·			
	Print	Ctrl+P	_		
	Documentation settings				
	Print preview				Create Cancel
	C:Documents and Settings	VAdmi\MODBUS			
	C:Documents and Settings	\sim\Project_PD			
	C:Documents and Settings	WAdWodbus RTU			
	F:WIREKImodbussimodbus	S Madanii Anna dhuan			
	CiDocuments and Settings	odbus slave test			
	CiDocuments and Settings	A MODBUS2poi			
	CiDocuments and Settings	VadmMODBUS2			
	5.3				
www.siemens.pl/sim	Eot				

Doradztwo techniczne: simatic.pi@siemens.com

Powinno pojawić się okno projektu sterownika SIMATIC S7-1200.

3.2 Dodanie nowego urządzenia

Z lewej strony okna w drzewie projektu "**Project tree**" należy kliknąć dwukrotnie na zakładkę **Add new device,** pojawi się okno o nazwie "**Add new device**", należy wpisać nazwę nowego urządzenia **np.** "**Modbus_Client".**

Następnie trzeba rozwinąć zakładkę **Unspecified CPU 1200,** zaznaczyć **6ES7 2xx-xxxx-xxxx.**

Ostatnim krokiem w tym oknie jest zaznaczenie **Open device view**, następnie należy kliknąć **OK**.

VA Siemens - FAQ_Modbus_TCP	
Project Edit View Insert Online Options Tools Window	Help
📑 🎦 🖬 Save project 📇 🐰 🗐 🗎 🗙 🏹 ± 🖓 ± 🏹	🗄 🛄 🕼 🖳 💋 Goonline 🖉 Gooffine 🏭 🌆 🖉 🛃 🛄
Project tree 🔲 🗸	Add new device
Devices	
1900 et al 19000 et al 1900 et al	Device name:
	PLC_1
▼ T FAQ_Modbus_TCP	
Add new device	PLC Device:
び 品 Devices & networks	▼ 🛄 SIMATIC \$7-1200
Gommon data	CPU CPU CPU CPU CPU CPU
G Documentation settings	PLC PLC PLC PLC PCDCDC
Caliguages aresources	CPU 1211 C DC/DC/R/y
SIMATIC Card Reader	CPU 1212C ACIDCRIy Unspecticed CPU 1200
	CPU 1212C DC/DC/DC
	Order no.: 6ES7 2000000000
	HMI FUT CPUT214C ACID CRIV Version: V2.1
	C CPU 1214C DODCRV Description
	▼ Unspecified CPU 1200
	6ES7 2XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
	FC surfaces
	► Systems In ATIC \$7-400
	• LI MARCEIZOU PEC
	Open device view OK Cancel

Po wprowadzonych zmianach automatycznie powinno otworzyć się okno projektu sterownika.

Najszybszym i najłatwiejszym sposobem konfiguracji sprzętowej jest bezpośrednie ściągnięcie jej z dostępnego fizycznie sterownika. Dokonuje się tego klikając opcję **detect.**

Otwiera się okno, w którym wyświetlone są dostępne, podłączone urządzenia. Wybieramy sterownik, z którym chcemy się połączyć i klikamy **detect.** Sprzęt zostanie wykryty automatycznie.

VA Siemens - FAQ_Modbus_TCP							
Project Edit View Insert Online Options Tools Window Help							
📑 🎦 🔒 Save project 💄 🐰 🛅 🗊 🗙 🍤 🛨 (주 🛨 🏹	🖥 🛄 🕼 🖳 🚿 Goonline 🖋 Gooffline 🏭 🖪 🧩 📃 🛄						
Project tree 🔲 🖣	FAQ_Modbus_TCP → Modbus_Client [Unspecified CPU 1200] _ ■ ■ ×						
Devices	🛃 Topology view 🛛 🛔 Network view 🛛 🛐 Device view						
. ™ O O M 🗎	🕌 Modbus_Client 💌 📰 🍊 🖽 🔍 🛨 100% 💌 📑						
Young Control of the second seco	57-1200 rack						

Hardware detection for	Modbus_Client				×
	Accessible devices in tar	get subnet:	ype of the PG/PC inte PG/PC inte	erface: 👤 PN/IE erface: 🔝 Intel(R	▼) 82577LM Gigabit ▼ 🔞
	Device	Device type	Туре	Address	MAC address
	MB_CLIENT	CPU 1214C DC/D	PN/IE	192.168.0.8	00-1C-06-00-2F-1E
Flash LED	MB_SERVER	CPU 1212C AC/DC	PN/IE	192.168.0.23	00-1C-06-02-9B-44
Online status information:					Kenesn
🛕 Found accessible dev	ice mb_server [192.168.0.	.23]			~
✓ Scanning ended.					~
					Detect <u>C</u> ancel

Dokładnie w ten sam sposób dodajemy drugi sterownik i nadajemy mu nazwę, np."Modbus_Server".

4 Modbus Serwer

4.1 Konfiguracja dwóch połączeń w jednym sterowniku "MB_SERVER"

Możliwe jest wykonanie kilku połączeń Modbus TCP server. Aby tego dokonać należy wywołać niezależnie dla każdego połączenia instrukcję MB_SERVER. Każde wywołanie musi posiadać przypisany indywidualny blok danych typu instancje DB, ID połączenia "CONNECT_ID" oraz numer portu IP "IP_PORT".

Zostaną stworzone dwa połączenia, z których jedno będzie służyło do przesyłu obszaru pamięci typu M, a drugie do przesyłu danych z bloku danych DB. Instrukcje "MB_SERVER" umieszczone zostaną w sterowniku, który został przez nas nazwany "Modbus_Server"

W pierwszej kolejności stworzony zostanie blok danych, w którym będą przechowywane zmienne, które mają być przesyłane.



W bloku danych tworzymy 10 zmiennych typu Int (razem 20 bajtów).

,T	СР	۲	Modbus_Server [C	PU 1212C AC/DC/R	lly] ▶ Pi	rogram	blocks → D	ata_block_1 [DB1] 🗕 🖬	∎×
101	2 2 2 4 5 1 E 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2									
	Da	ta_	block_1							
_		Na	me	Data type	0	ffset	Start value	Retain	Visible in	Comm
1	-	•	Static							
2		•	MB_1	Int	🔳 🔽 0.	.0			~	
3		•	MB_2	Int	2.	.0	0			
4		•	MB_3	Int	4.	.0	0			
5	-	•	MB_4	Int	6.	.0	0			
6		•	MB_5	Int	8.	.0	0		\checkmark	
7	-	•	MB_6	Int	10	0.0	0			
8		•	MB_7	Int	12	2.0	0			
9	-	•	MB_8	Int	14	4.0	0			
10		•	MB_9	Int	16	6.0	0			
11		•	MB_10	Int	18	8.0	0			
12		•	<add new=""></add>							

Następnie w bloku OB1 "MAIN" wywołujemy dwukrotnie instrukcję "MB_SERVER" i określamy parametry połączenia:

- "DISCONNECT" ustawiamy na zero. Oznacza to, że może zostać zainicjowane połączenie pasywne

- "CONNECT_ID" unikalny numer połączenia – "1" dla połączenia pierwszego, "2" dla drugiego
 - "IP PORT" numer portu IP – każde połączenie musi być przypisane do odrębnego portu.

Więcej na temat numerów portów znajduje się helpie STEP 7

- "MB_HOLD_REG" wskaźnik obszaru pamięci zawiera zmienne, do których dostęp będzie miał klient Modbusa używający funkcji 3 (read), 6 (write) oraz 16 (read)



4.2 Wyjaśnienie parametrów wejściowych "MB_SERVER"

Parametr	Typ danych	Opis
DISCONNECT	Bool	"MB_SERVER" próbuje nawiązać pasywne połączenie z partnerem. Oznacza to, że serwer jest w trybie pasywnym i nasłuchuje wywołania. Jeżeli "DISCONNECT" = 0 i połączenie jeszcze nie jest nawiązane, wtedy nowe połączenie może zostać zainicjowane Jeżeli "DISCONNECT" = 1 i połączenie jest nawiązane, wtedy połączenie jest przerwane. Pozwala to programowi kontrolować połączenie po jego nawiązaniu. Gdy parametr ma wartość "1", nie można zainicjować połączenia
CONNECT_ID	UInt	Parametr ten identyfikuje unikalne połączenia wewnątrz PLC. Każdy unikalny blok danych funkcji "MB_CLIENT" lub "MB_SERVER" musi posiadać unikalny numer ID połączenia
IP_PORT	UInt	Wartość domyślna = 502: Numer portu IP identyfikuje port IP który będzie monitorowany w celu wykrycia zapytania od klienta Modbusa. Poniższe numery TCP portu nie są dozwolone dla pasywnego połączenia "MB_SERVER":20, 21, 25, 80, 102, 123, 5001, 34962 oraz 34964

MB_HOLD_REG	Variant	Wskaźnik do rejestru pamięci serwera: Rejestr pamięci musi być blokiem danych o standardowym dostępie lub obszarem pamięci M. Ten obszar pamięci wykorzystywany jest do przechowywania danych, do których będzie miał dostęp klient Modbusa używając funkcji 3 (read), 6 (write) oraz 16 (write)
-------------	---------	--

5 Modbus Klient

5.1 Konfiguracja dwóch połączeń w jednym sterowniku "MB_CLIENT"

Zapytania klienta Modbusa mogą być rozsyłane za pośrednictwem kilku połączeń. Aby było to możliwe należy w programie umieścić instrukcję "MB_CLIENT" (po jednej dla każdego połączenia), jednocześnie należy pamiętać, że każde jej wywołanie musi mieć oddzielny blok instancji DB. Adres IP musi być różny, jeżeli łączymy się z różnymi sterownikami lub taki sam, jeżeli realizujemy kilka połączeń w obrębie tego samego sterownika. Jeżeli nawiązywanych jest kilka połączeń z tym samym sterownikiem należy pamiętać, aby odwołać się do odpowiedniego portu IP. Poniższy przykład pokazuje, w jaki sposób zrealizować dwa połączenia z jednym sterownikiem (jeden adres IP, dwa numery portu). Instrukcje "MB_Client" zostaną umieszczone w bloku OB1 w sterowniku, który nazwaliśmy "Modbus_Client".

Pierwszym krokiem jest określenie obszarów pamięci, do których mają zostać zapisane dane odczytane z serwera. W tym celu utworzymy dwa bloki danych. Pierwszy będzie składał się z dziesięciu zmiennych typu WORD (20 bajtów), a w drugim umieścimy tablicę składającą się ze 100 elementów o wielkości jednego bajtu. W tym celu klikamy dwukrotnie na "Add New block", wybieramy Data block i ustawiamy standardowe adresowanie (Block Access >> Standard - Compatible with S7-300/400). W pierwszym bloku danych umieszczamy 10 zmiennych o nazwie "MB" typu WORD. W tym celu wpisujemy nazwe zmiennej "MB", typ danych ustawiamy WORD i chwytając za prawy dolny róg pola z nazwą zmiennej przeciągamy w dół, dzięki czemu automatycznie utworzy nam się pożądana liczba kolejno ponumerowanych zmiennych.

FA	Q_I	Mo	dbus_TCP → Modbus_(Client [CPU 1	214C D	adadc]	 Program blo 		
🔹 🔹 💺 🎟 🚍 💖									
	Data_block_3								
		Na	me	Data type		Offset	Start value		
1		•	Static						
2	-00	•	МВ	Word			0		
3		•	<add new=""></add>						
	_								
				+					
FA	Q_	Mo	dbus_TCP → Modbus_	Client [CPU '	214C D	apapado]	Program blo		

3	¢ =	*	🎭 🋃 🧮 🗮			
	Da	ta_	_block_3			
		Na	me	Data type	Offset	Start value
1		•	Static			
2		•	MB	Word 🔳 💌]	0
3		•	MB_1	Word		0
4		•	MB_2	Word		0
5		•	MB_3	Word		0
6		•	MB_4	Word		0
7		•	MB_5	Word		0
8		•	MB_6	Word		0
9		•	MB_7	Word		0
10		•	MB_8	Word		0
11	-		MB_9	Word		0
12		•	⊲Add new>			

W drugim bloku danych umieszczamy tablicę "array" składającą się ze 100 elementów typu bajt. W tym celu w polu "Name" wpisujemy nazwę tablicy, np. "tablica". Następnie w polu "Data type" wpisujemy literę "a" i czekamy aż rozwinie się pole wyboru zmiennej", po czym wybieramy typ "Array [lo .. hi] of type". Wpisujemy zakres i określamy typ zmiennych.

Data black 4

	Da	ta_	block_4						
		Na	ime	Data type	Offs				
1	-00	•	Static						
2	-00	•	tablica	a					
З			<add new=""></add>	AOM_IDENT					
				Array [lo hi] of type					
				v					
					-				

	Da	ta_	<u>_DI</u>	оск_4			
	-	Na	me	1	Data type	Offset	Start value
1	-	•	St	atic			
2	-		•	tablica	Array [0 99] of Byte 🛛 🔳 📼	•	
3			•	tablica[0]	Byte		0
4	-		•	tablica[1]	Byte		0
5	-		•	tablica[2]	Byte		0
6	-		•	tablica[3]	Byte		0
7	-		•	tablica[4]	Byte		0
8	-		•	tablica[5]	Byte		0
9	-		•	tablica[6]	Byte		0
10	-		•	tablica[7]	Byte		0
11	-		•	tablica[8]	Byte		0
12	-		•	tablica[9]	Byte		0
13	-		•	tablica[10]	Byte		0
14	-		•	tablica[11]	Byte		0
15	-		•	tablica[12]	Byte		0
16	-		•	tablica[13]	Byte		0
17	-		•	tablica[14]	Byte		0
18	-		•	tablica[15]	Byte		0
19	-		•	tablica[16]	Byte		0
20			•	tablica[17]	Byte		0
21			•	tablica[18]	Byte		0
22			•	tablica[19]	Byte		0
23			•	tablica[20]	Byte		0
24			•	tablica[21]	Byte		0
25	-		•	tablica[22]	Rute		0

Z biblioteki instrukcji przeciągamy dwukrotnie do OB1 instrukcję "MB_CLIENT" (podobnie jak MB_SERVER) oraz przypisujemy odpowiednie parametry:

- "REQ" – żądanie połączenia z serwerem Modbusa (zbocze narastające)

- "DISCONNECT" – ustawiamy "0", jeżeli chcemy, aby nawiązanie połączenia było możliwe.

Po nawiązaniu połączenia można je zerwać za pomocą tego parametru ustawiając "1"

- "CONNECT_ID" - ustawiamy odpowiednio "1" i "2"

- "IP_OCTET_1..4" - wpisujemy adres IP sterownika, z którym będziemy się łączyć

- "IP_PORT" – numer portu nawiązuje do numeracji portów w serwerze

- "MB_MODE" – ustawiamy "0", dzięki czemu możliwy będzie odczyt z serwera

- "MB_DATA_ADDR" – wpisujemy 40001, co oznacza, że zaczniemy odczyt od początku zadeklarowanego w serwerze rejestru

- "MB_DATA_LEN" – parametr określa ilość danych jaka ma być odczytana z serwera. Wpisujemy odpowiednio 10 (10 słów) oraz 100 (100 słów)

- "MB_DATA_PTR" – jest to wskaźnik do lokalnego obszaru pamięci. Określa obszar danych, do którego mają zostać zapisane dane odczytane z serwera.

Opcjonalnie możemy dodać instrukcję "MOVE", dzięki czemu możliwe będzie zatrzaśnięcie statusu w momencie pojawienia się błędu.



Po wykonaniu wszystkich powyższych kroków ładujemy programy do sterowników. Jeżeli chcemy załadować program do wszystkich sterowników jednocześnie możemy je umieścić w jednej grupie.

W tym celu klikamy prawym przyciskiem myszy na nazwę projektu w drzewie projektu i wybieramy "Add group". Foldery poszczególnych sterowników przeciągamy do nowo utworzonej grupy, zaznaczamy ją lewym przyciskiem myszy i klikamy "Download to device"

Jeżeli w programie nie było żadnych błędów, zostanie on załadowany. Następnie zaznaczamy opcje "Start all", co spowoduje przejście sterowników w tryb "RUN"

Project tree	
Devices	
ڬ O O 🖆	≣
FAQ_Modbus_TCP	
💕 Add new device	
Add group	
💥 Cut	Ctrl+X
Сору	Ctrl+C
📋 Paste	Ctrl+V
🚬 🝠 Go online	Ctrl+K
🕨 🖣 🚮 Go offline	Ctrl+M
Receive alarms	
Cross-reference information	n Shift+F8
Cross-references	FS
📃 Print	Ctrl+P
Print preview	
國 Properties	Alt+Enter

V11	Siemens - FAQ_Modbus_TCP							
Pr	oject Edit View Insert Online Options	Tools	Windov	v <u>He</u>	lp			
	🛉 🎦 🔚 Save project 昌 🐰 💷 🗂 🗙 🍤) ± (2	1± 🖬	=	🖪 🚹 🖳 🔝 💋 Go online 🖉 G	o offline 🔚 🖪 🖪 🗶 🚽 🛄		
	Project tree 🛛 🔲 🗸	FAC) Mo	S TI	P ▶ Group 1 ▶ Modbus Client	[CPU 1214C DC/DC/DC] > Program blocks	▶ Main [OB1]	
	Devices		/		Download to device			
			¥					
		1 KQI	⊦g ≣∘	5	▝▌╘╘╜╝┋╸╝╘╵╝			
ing					- <u>J</u>		Block interface	
E	▼ ☐ FAQ_Modbus_TCP	I ▼ ∄	Rlack tit	· • م	"May Program Swaan (Ousla)"			X
Jrai	Add new device		Load pre	view				X
l e	Devices & networks		2 c	heck b	efore loading			
J.		•	•					
Ы	Modbus_client [CPU 1214C DC/DC/DC]	11	Status	1	Target	Message	Action	
	Common data	11	+I	0	 Modbus_Client 	Ready for loading.		^
	Documentation settings	11						=
	Canquages & resources	1.1		0	 Device configuration 	Delete and replace system data in target	Download to device	
	Online access	1.1						_
	SIMATIC Card Reader	11		0	▼ Software	Download software to device	Consistent download	
		11		2	 Download to device 	Objects that do not exist online.	_	
		11		2	Tags			
		11		2	Main [OB1]		Download to device	
		11		š	MB_CLIENT [PB1084]		Download to device	
				ă	MB_CLIENT_DB [DB1]		Download to device	
				ă	Data block 3 [DB3]		Download to device	
				ă	Data_block_5 [DB6]		Download to device	
				×	bata_brook_1[bb1]		e ponnou o ucrice	
		11	40	0	➡ Modbus_Server	Ready for loading.		
		1.1						
		1.1		0	 Device configuration 	Delete and replace system data in target	Download to device	
		11		0		Download software to device	Consistent download	
		11		0	 Download to device 	Objects that do not exist online.		
		11		0	Tags			
				S	Main [OB1]		Download to device	
		Ш		2	Data_block_1 [DB1]		Download to device	
				2	MB_SERVER [FB1085]		Download to device	
				2	MB_SERVER_DB [DB2]		Download to device	
		21		v	MB_SERVER_DB_T[DB3]		Download to device	~
								Retresh
		ŏ					Einish Load	Cancel

Status	ļ	Target	Message	Action	
4	<u> </u>	➡ Modbus_Client	Downloading to device completed without error.		^
	4	 Start modules 	Start modules after downloading to device.	🛃 Start all	
4	<u> </u>	➡ Modbus_Server	Downloading to device completed without error.		
	▲	 Start modules 	Start modules after downloading to device.	🛃 Start all	*

Oczywiście można również załadować program do każdego sterownika osobno bez tworzenia grupy urządzeń.

5.2 Wyjaśnienie parametrów wejściowych "MB_CLIENT"

Parametr	Typ danych	Opis
REQ	Bool	FALSE = brak żądania komunikacji Modbus
		TRUE = żądanie komunikacji z serwerem Modbus
DISCONNECT	Bool	Parametr ten pozwala na kontrolę połączenia z serwerem Modbus
		przez program.
		Jeżeli "DISCONNECT" = 0 i połączenie nie zostało jeszcze nawiązane,
		wtedy "MB_CLIENT" próbuje nawiązać połączenie z przypisanym
		adresem IP i numerem portu
		Jeżeli "DISCONNECT" = 1 i połączenie istnieje, wtedy następuje
		przerwanie połączenia. Jeżeli ten parametr ma wartość 1, wtedy nie
		zostanie nawiązane żadne połączenie
CONNECT_ID	UInt	Parametr ten identyfikuje unikalne połączenia wewnątrz PLC.
		Każdy unikalny blok danych funkcji "MB_CLIENT" lub "MB_SERVER"
		musi posiadać unikalny numer ID połączenia
IP_OCTET_1	USInt	Adres IP serwera Modbus TCP: Oktet 1
		8 bitowa część 32 bitowego adresu IP IPv4 serwera Modbus TCP
		z którym łączy się klient za pośrednictwem protokołu Modbus TCP
IP_OCTET_2	USInt	Adres IP serwera Modbus TCP: Oktet2
IP_OCTET_3	USInt	Adres IP serwera Modbus TCP: Oktet3
IP_OCTET_4	USInt	Adres IP serwera Modbus TCP: Oktet4
IP_PORT	UInt	Wartość domyślna = 502: Numer portu IP identyfikuje port IP serwera,
		do którego klient będzie się odwoływał przystępując do nawiązania
		połączenia używając protokołu TCP/IP.

6 Watch tables

Aby móc przetestować aplikację stworzymy tablice do obserwacji i modyfikacji zmiennych "Watch table". Klikamy "Add new watch table" i wpisujemy w niej zmienne, do których chcemy mieć dostęp. Tworzymy dwie takie tablice. Jedna w "Modbus_Client", a druga w "Modbus_Server". Aby móc obserwować dwie tablice jednocześnie, klikamy ikonkę "float".

Project tree 🛛 💷	<aq_№< th=""><th>lodbus_TCP → Modbus_Client</th><th>[CPU 1214C DC/D</th><th>C/DC] → Watch a</th><th>1d force tables</th><th>• Watch table_1</th><th>_.×</th></aq_№<>	lodbus_TCP → Modbus_Client	[CPU 1214C DC/D	C/DC] → Watch a	1d force tables	• Watch table_1	_ . ×
Devices							-
	· 😼 Lo	1 10 1 00 00					
	i	Name	Address	Display format	Monitor value	Modify value	9
▼ 🔄 FAQ_Modbus_TCP	1	"Client1_REQ"	🔳 %M10.0	Bool	-		
🗳 Add new device	2	"client1_STAT_ZATRZ"	%MW16	Hex			
📥 Devices & networks	3	"Data_block_3".MB	%DB3.DBW0	DEC_signed			
▼ 1 Modbus_Client [CPU 1214C DC/DC/D	C] 4	"Data_block_3".MB_1	%DB3.DBW2	DEC_signed			
T Device configuration	5	"Data_block_3".MB_2	%DB3.DBW4	DEC_signed			
😼 Online & diagnostics	6	"client2_REQ"	%M10.1	Bool			
🕨 🛃 Program blocks	7	"client2_STAT_ZATRZ"	%MW18	Hex			
🕨 🙀 Technology objects	8	"Data_block_4".tablica[0]	%DB4.DBB0	DEC_signed			
External source files	9	"Data_block_4".tablica[1]	%DB4.DBB1	DEC_signed			
🕨 🔚 PLC tags	10	"Data_block_4".tablica[2]	%DB4.DBB2	DEC_signed			
🕨 🔚 PLC data types 🛛 🎽	11		<add new=""></add>				
🗢 🥅 Watch and force tables 📕							
📑 Add new watch table							
Force table							
Watch table_1							
📴 Program info							
🛅 Text lists							
🕨 🥅 Local modules							
Modbus_Server [CPU 1212C AC/DC/F	R						
Project tree	AQ_₩	lodbus_TCP → Modbus_Client	[CPU 1214C DC/D	C/DC] → Watch aı	nd force tables 🛛	• Watch table_1	_ _ = ×
Devices							
M 0 0 N	· 🛛 🖌 🗛	9.9.9° an an					

Devices							
	19 🔓	1 10 1 P 🚏 📬					
	i	Name	Address	Display format	Monitor value	Modify value	9
▼ 🔄 FAQ_Modbus_TCP	1	"Client1_REQ"	%M10.0	Bool 💌			
📑 Add new device	2	"client1_STAT_ZATRZ"	%MW16	Hex			
📥 Devices & networks	3	"Data_block_3".MB	%DB3.DBW0	DEC_signed			
▼ 🛅 Modbus_Client [CPU 1214C DC/DC/DC]	4	"Data_block_3".MB_1	%DB3.DBW2	DEC_signed			
🕎 Device configuration	5	"Data_block_3".MB_2	%DB3.DBW4	DEC_signed			
😼 Online & diagnostics	6	"client2_REQ"	%M10.1	Bool			
🕨 🛃 Program blocks	7	"client2_STAT_ZATRZ"	%MW18	Hex			
🕨 🙀 Technology objects	8	"Data_block_4".tablica[0]	%DB4.DBB0	DEC_signed			
External source files	9	"Data_block_4".tablica[1]	%DB4.DBB1	DEC_signed			
🕨 🔚 PLC tags	10	"Data_block_4".tablica[2]	%DB4.DBB2	DEC_signed			
🕨 📴 PLC data types 🛛 🌽	11		<add new=""></add>				
🗢 🥅 Watch and force tables 📕							
📑 Add new watch table							
Force table							
Watch table_1							
📴 Program info							
🛅 Text lists							
🕨 🛅 Local modules							
Image: Modbus_Server [CPU 1212C AC/DC/R							

Klikamy podgląd (okularki), w kolumnie "Modify value" wpisujemy wartości, które chcemy przesłać i klikamy ikonkę "modify".

VA	N Siemens - FAQ_Modbus_TCP										
Pr	oject Edit View Insert Online Options 1	ools Win	dow Help								
	🕸 🍽 🛄 Save project 💻 🐰 🗐 🗎 🗙 🔊	+ (4+)		Go online 🛷 Go offline	🛵 🖪 🖪 🗙						
	Project tree 📖 🖣	FAQ_M0	abus_ICP > Group_I >	Modbus_Server [CPU]	I ZI ZC AUDURIYJ						
	Devices										
	NOO	ТСР	→ Group 1 → Modbus Cl	ient ICPU 1214C DC/DC	7DCl → Watch a	nd force tables	Watch table 1	_ E = X			
	- 5 E40 Modburg TCP										
	Add new device	lø Lo	91 96 17 🚏 🖺								
	Add new device A petworks	i	Name	Address	Display format	Monitor value	Modify value	9			
	The Group 1	1	"Client1_REQ"	%M10.0	Bool	FALSE					
	Modbus Client CPU 1214C DC/DC/	2	"client1_STAT_ZATRZ"	%MW16	Hex	16#0000					
	Modbus Server [CPU 1212C AC/DC.	3	"Data_block_3".MB	%DB3.DBW0	DEC_signed	0					
	▶ 🙀 Common data	4	"Data_block_3".MB_1	%DB3.DBW2	DEC_signed	0					
	Documentation settings	5	"Data_block_3".MB_4	%DB3.DBW8	Hex	 16#0000 					
	▶ 🐻 Languages & resources	6	"client2_REQ"	%M10.1	Bool	FALSE					
	Online access	7	"client2_STAT_ZATRZ"	%MW18	Hex	16#0000					
	SIMATIC Card Reader	8	"Data_block_4".tablica[0]	%DB4.DBB0	DEC_signed	0					
		9	"Data_block_4".tablica[1]	%DB4.DBB1	DEC_signed	0					
		10	"Data_block_4".tablica[99]	%DB4.DBB99	DEC_signed	0					
		11		<add new=""></add>							
		<						>			
		TCP >	Group 1 → Modbus Se	rver [CPU 1212C AC/DC	ZRIy] → Watch a	nd force tables	• Watch table 1				
		-									
		🕼 🕼	🕺 1 76 17 🙄 🙄 📬								
		i	Ne Modify all selected valu	es once and now.	Display format	Monitor value	Modify value	9			
		1	"Data_block_1".MB_1	%DB1.DBW0	DEC_signed	23543	23543	🗹 🔺			
		2	"Data_block_1".MB_2	%DB1.DBW2	DEC_signed	24332	24332	🗹 🔺			
		3		%DB1.DBW8	DEC_signed	▼ 1233	1233	🗹 🔺			
		4	"Tag_1"	%MB100	DEC_signed	23	23	🗹 🔔			
		5		%MB101	DEC_signed	21	21	🗹 🔺			
		6		%MB199	DEC_signed	16	16	🗹 🔺			
		7		<add new=""></add>							
		<						>			

W "Watch table" klienta uruchamiamy odczyt poprzez zmianę parametru REQ. Parametr REQ może być zmieniony np. poprzez kliknięcie prawym przyciskiem myszy na wiersz, w którym znajduje się parametr i wybraniu opcji "modify to 1". Następnie ustawiamy parametr powtórnie na "0". Operację wykonujemy najpierw dla "Client1_REQ", a następnie dla "Client2_REQ".

	TCP 🕨	Group_1 → Modbus_Client [Cl	PU 1214C DC/DC	VDC] → V	Vatch and forc	e tables		Watch table_1	_ IE
Þ		91 % # 📭 m							
	i	Name	Address	Display f	ormat Monit	tor value		Modify value	1
1		"Client1_REQ"	%M10.0	Bool	💌 🗖 FA	LSE		FALSE	🛛 🔼
2		"client1_STAT_ZATRZ"	%MW16	Hex	Modify	•		Modify to 0	Ctrl+Shift+0
3		"Data_block_3".MB	%DB3.DBW0	DEC_sig	🔭 Monitor all			Modify to 1	Ctrl+Shift+1
4		"Data_block_3".MB_1	%DB3.DBW2	DEC_sig	🔄 Monitor now	/	1	Modify now	
5		"Data_block_3".MB_4	%DB3.DBW8	Hex	M. Cut	real . V	%	Modify with trigge	er -
6		"client2_REQ"	%M10.1	Bool	a Conv	Ctrl+A	Z?	Enable periphera	loutputs
7		"client2_STAT_ZATRZ"	%MW18	Hex	Baste	Ctrl+V			
8		"Data_block_4".tablica[0]	%DB4.DBB0	DEC_sig		Cult+v			
9		"Data_block_4".tablica[1]	%DB4.DBB1	DEC_sig	🗙 Delete	Del			
10		"Data_block_4".tablica[99]	%DB4.DBB99	DEC_sig	Rename	F2			
11			<add new=""></add>		🗓 Expanded M	lode			
	<	III							>

Zmienne zostały odczytane, a parametry: "client1_STAT_ZATRZ" oraz "client2_STAT_ZATRZ" mają wartość zero (brak błędu podczas przesyłu).

FAQ	AQ_Modbus_TCP										
	тср 🔸	Group_1 → Modbus_Client [Cl	PU 1214C DC/DC	/DC] → Watch and	l force tables 🔸	Watch table_1	_ 12 _	IX			
Þ											
	i	Name	Address	Display format	Monitor value	Modify value	9				
1		"Client1_REQ"	%M10.0	Bool	FALSE	FALSE	Image: A start of the start				
2		"client1_STAT_ZATRZ"	%MW16	Hex	16#0000						
3		"Data_block_3".MB	%DB3.DBW0	DEC_signed	23543						
4		"Data_block_3".MB_1	%DB3.DBW2	DEC_signed	24332						
5		"Data_block_3".MB_4	%DB3.DBW8	DEC_signed 💌	1233						
6		"client2_REQ"	%M10.1	Bool	TRUE	TRUE	Ī 🗹 ,				
7		"client2_STAT_ZATRZ"	%MW18	Hex	16#0000						
8		"Data_block_4".tablica[0]	%DB4.DBB0	DEC_signed	23						
9		"Data_block_4".tablica[1]	%DB4.DBB1	DEC_signed	21						
10		"Data_block_4".tablica[99]	%DB4.DBB99	DEC_signed	16						
11			<add new=""></add>								
	<	1111						>			

7 Zakres adresów

Zamiast numeru funkcji instrukcja "MB_CLIENT" używa parametru "MB_MODE". Parametr "MB_DATA_ADDR" używany jest do określania adresu startowego obszaru danych, jaki chcemy przesłać. Kombinacja tych dwóch parametrów definiuje kod funkcji Modbus. Poniższa tabela zawiera zależność między parametrem "MB_MODE", kodem funkcji Modbus oraz zakresem adresów.

Parametr	Kod funkcji Modbus	Parametr	Funkcjonalność i typ danych	Parametr
0	01	1 do 2000	Czytanie bitów wyjściowych: 1 do 2000 bitów podczas jednego wywołania	1 do 9999
0	02	1 do 2000	Czytanie bitów wejściowych: 1 do 2000 bitów podczas jednego wywołania	10001 do 19999
0	03	1 do 125	Odczyt rejestrów pamiętających: 1 do 125 zmiennych typu WORD podczas jednego wywołania	40001 do 49999
0	04	1 do 125	Odczyt słów wejściowych: 1 do 125 słów podczas jednego wywołania	30001 do 39999
1	05	1	Zapis pojedynczego bitu wyjściowego: Jeden bit podczas jednego wywołania	1 do 9999
1	06	1	Zapis rejestru pamiętającego: Jedno słowo podczas jednego wywołania	40001 do 49999
1	15	2 do 1968	Zapis bitów wyjściowych: 2 do 1968 bitów podczas jednego wywołania	1 do 9999
1	16	2 do 123	Zapis rejestrów pamiętających: 2 do 123 słów podczas jednego wywołania	40001 do 49999

8 Kody warunkowe

8.1 Wartość "STATUS" dla "MB_SERVER"

- 8187 Nieprawidłowy wskaźnik dla parametru "MB_HOLD_REG": obszar pamięci jest zbyt mały
- 818C Parametr "DATA_PTR" wskazuje na zoptymalizowany blok danych (musi być standardowy blok DB lub obszar pamięci M)
- 8381 Nieobsługiwany kod funkcji
- 8382 Błąd długości danych
- 8383 Błąd adresu danych lub próba dostępu poza granice określone przez "MB"HOLD_REG"
- 8384 Błąd wartości danych
- 8385 Nieobsługiwana wartość kodu diagnostycznego (kod funkcji 08).

8.2 Wartość "STATUS" dla "MB_CLIENT"

- 80C8 Serwer nie odpowiada w określonym czasie
- 8188 Nieprawidłowa wartość parametru "MODE"
- 8189 Nieprawidłowa wartość parametru "DATA_ADDR"
- 818A Nieprawidłowa wartość parametru "DATA LEN"
- 818B Nieprawidłowy wskaźnik obszaru pamięci "DATA_PTR". Błąd może być spowodowany niepoprawną kombinacją parametrów "MB_DATA_ADDR" oraz "MB_DATA_LEN"
- 818C Parametr "DATA_PTR" wskazuje na zoptymalizowany blok danych (musi być standardowy blok DB lub obszar pamięci M)
- 8200 Port jest zajęty przetwarzaniem innego żądania połączenia
- 8380 Otrzymana ramka Modbusa jest zniekształcona lub otrzymano zbyt dużą ilość bajtów
- 8387 Przypisany numer ID połączenia jest różny od ID z poprzedniego zapytania. Może być tylko jeden numer ID połączenia użyty wewnątrz przypisanego do "MB_CLIENT" bloku daych DB
- 8388 Serwer Modbus zwrócił ilość danych inną niż był odpytany. Tyczy się tylko funkcji 15 i 16.