# Komunikacja paneli Astraada HMI ze sterownikami Siemens S7-1200

### WSTĘP

Panele operatorskie Astraada HMI Panel mają możliwość wymiany danych procesowych ze sterownikami Siemens S7 w sieci Ethernet. Pozwala na to odpowiedni driver komunikacyjny wbudowany w oprogramowanie narzędziowe Astraada HMI CFG. W celu uruchomienia komunikacji konieczna jest odpowiednia konfiguracja drivera po stronie panelu i sterownika.

W instrukcji opisano zagadnienia związane z:

- Konfiguracją panelu Astraada HMI i sterownika Siemens S7-1200
- Adresowaniem zmiennych
- Importem własnych typów danych
- Importem zmiennych zorganizowanych w Data Blockach

Uwaga – akapity poświęcone importowaniu zmiennych w Data Blockach dotyczą tylko wersji oprogramowania Astraada HMI CFG 4.0.x.x, obsłudze importu zmiennych w starszych wersjach oprogramowania poświęcona jest osobna instrukcja dostępna w <u>zakładce Wsparcie na stronie</u> internetowej firmy Astor.

#### **KONFIGURACJA STEROWNIKA**

Po podstawowej konfiguracji sterownika w oprogramowaniu TIA Portal należy przejść do zakładki Properties w kontekście określonego urządzenia, a następnie w zakładce Protection zaznaczyć opcję Permit access with PUT/GET communication...

Astraada_HMI > PLC_1 [CPU 1212C AC/DC/Rly]								₽∎×
			2	Topology view	🔒 Netwo	rk view	Devic	e view
🕌 PLC_1 💌 🖽 🔣 🔍 ±			Devic	ce overview				
		^	- <b>*</b>	Module		Slot	I address	Q address
						103		
						102		
C)						101		
w v				PLC_1		1		
				DI 8/DQ 6_	1	11	0	0
103 102 101 1	2 3			AI 2_1		12	6467	
Rack 0						13		
SIEMENS SHERE SHERE				HSC_1		116	100010	
				HSC_2		1 17	100410	
				HSC_3		118	100810	
1 4 3 ceu auto Acecesy			4	HSC_4		119	101210	
				HSC_5		1 20	101610	
			•	HSC_6		1 21	102010	
				Pulse_1		1 32		10001
				Pulse_2		1 33		10021
				Pulse_3		1 34		10041
				Pulse_4		1 35		10061
				► PROFINET II	nterface_1	1 X1		
						2		
						3		
			_					
			_					
			_					
			_					
	_	~	_					
× III > 100%	▼ <u></u> Ÿ.	. 1	<			Y.		>
PLC_1 [CPU 1212C AC/DC/Rly]				Roperties	1 Info	🖁 Diagi	nostics	
General IO tags System constants Texts								
Pulse generators (PTO/PWM)								^
Startup								
Cycle								
Communication load								
System and clock memory								
Webserver								_
User interface languages Connection mechanisms								
Time of day ≣ ►		an a fals man					~ ``	
Protection	rermit acce	ss with PU	I/GE I COMI	munication from rem	ote partner (P	LC, HIMI, OF	· (,)	
Configuration control								Ξ
Connection resources								
Overview of addresses 🗸								
< III >								~

Po wykonaniu tych czynności, skompilowaniu programu i pobraniu aplikacji do sterownika możliwa będzie komunikacja ze sterownikiem z panelu Astraada HMI.

### KONFIGURACJA LINKU W PANELU OPERATORSKIM

W trakcie tworzenia nowego projektu na panel operatorski w oprogramowaniu Astraada HMI CFG należy skonfigurować link komunikacyjny odpowiedni dla urządzenia S7-1200 w sposób przedstawiony poniżej. Jeśli nie zrobiono tego na etapie tworzenia projektu można dodać nowy link klikając PPM na obiekcie Links w Project Managerze i wybierając Add Link...





W zakładce General należy wybrać link Ethernet, Device/Server Siemens AG -> SIMATIC S7-1200

Link Properties						×
General Paramet	ter					
Link Number:	1					
Link Name:	Link 1					
Link Type:	Direct Link (Ethernet)				$\sim$	
Device/Server:	Siemens AG	~	SIMATIC S7-1200 (Ethernet)		$\sim$	
			LOGO (Ethernet) SIMATIC S7 (Ethernet)			
			SIMATIC S7-1200 (Ethernet) SIMATIC S7-200 (Ethernet) SIMATIC S7-200 SMART (Ethernet) SIMATIC S7-300 (Ethernet) Simatic S7-300 Ethernet Module (CP343)			
Record comm	unication status in operation	log				
The duration of s	showing a communication err	or message:	5 v second(s)			
				ОК	Anuluj	Pomoc



W zakładce Parameter należy wpisać adres IP sterownika i ustawić niezerowy Timeout Time.

Link Properties	$\times$
General Parameter	
IP Address: 192.168.1.51 Use Default Port Port: 102 Use Default Rack and Slot Numbers Connection type: OP Rack Number: 0 Slot Number: 1	
Sidt Number:	
Timeout Time:       5       (x 0.1 Sec.)         Command Delay:       3       (x 1 ms)         Retry Count:       0       •         Try reconnecting immediately after communication break (Not recommended; the normal way will wait 3~60 seconds before reconnecting)	
OK Anuluj Pom	ос

Po zatwierdzeniu ustawień link zostanie skonfigurowany.

#### **ADRESOWANIE ZMIENNYCH**

Po podstawowej konfiguracji urządzeń przestawionej powyżej można już odczytywać i zapisywać zmienne w pamięci wewnętrznej sterownika. Dla przykładu utworzono w oprogramowaniu Astraada HMI CFG obiekt Numeric Entry, Bit Button i Bit Lamp. Obiekt Numeric Entry zaadresowano wpisując adres zmiennej QW0 i typ danych 16-Bit Unsigned Integer.



			Address Input Keypad ?	×
umeric Entry General Advance	ed Visibility	Gradient	Link: Link 1	~
ID: NE0000	Note:		Type: Bit Of Word OBit Of Word	
		Transparent	513 V: QW V	_
		Shape	C A 1 2 3 CLR	ł
		Border Color:	D B 4 5 6 BS	
NO_BE	DR	BG Color:	E : 7 8 9 ESC	:
Data Type:	16-Bit Unsig	ned Integer		٦
Display Type:	16-Bit Unsig	ned Decimal		
Write Address:	QW0			
Monitor addre	ess identical to	write address	Macro	

Do obiektu Bit Button w analogiczny sposób przypisano adres Q0.0, a do obiektu Bit Lamp adres I0.0.

	)efa	ult tag table									
		Name	Da	ata type	Address		Retain	Visibl	Acces	Monitor value	Comment
1	-00	0wp	Int	t	%QW0			<b>~</b>	$\checkmark$	281	
2	-00	iO	Bo	ool	%10.0			<b></b>	<b></b>	FALSE	
3	-00	qО	Bo	ool 🔳	%Q0.0	•			<b></b>	TRUE	
			Q0.0 ON QW0	81	IO.0 Off						



### ADRESOWANIE ZMIENNYCH W DATA BLOCKACH

W programie w TIA Portal utworzono dla przykładu nowy Data Block DB1 o nazwie Data\_block\_1.

Aby możliwy był odczyt i zapis zmiennych wewnątrz Data Blocków z poziomu panelu operatorskiego podczas tworzenia nowego bloku należy odznaczyć opcję Symbolic access only,

Add new block				×
Namo				
Data block 1				
Data_DIOCK_1				
	Туре:	🧧 Global DB	•	
	Language:	DB 💌		
Organization	Number:	1		
(OB)		🔵 Manual		
		Automatic		
		Symbolic access only		
Function	Description:			
block (FB)	Data blocks (DBs)	are data areas in the program that cor ollowing types:	itain user data.	
	- A global data blo	ck		
-	- An instance data	DIOCK		
_				
Function				
Data block				
(DB)	more			
Further information	ation			
✓ Add new and oper	n		ОК	Cancel

bądź w zależności od wersji oprogramowania wejść w ustawienia Data Blocku i w zakładce Attributes odznaczyć opcję Optimized block access.

Devices		
	Data_block_1 [DB1]	
	General	
🔻 📄 Astraada_HMI	General	
💣 Add new device	Information	Attributes
📩 Devices & networks	Time stamps	
🔻 🚹 PLC_1 [CPU 1212C A	Compilation	Only store in load memory
时 Device configurat	Protection	Data block write-protected in the device
😵 Online & diagnos	Attributes	
🔻 🔂 Program blocks	Download with	Optimized block access
💣 Add new blocl		
💶 Main [OB1]		
j Data_block_1		
🕨 🙀 Technology objec		
🕨 🔚 External source fi		
🔻 浸 PLC tags		
lags 🖏 🖏 🖏		
📩 Add new tea t		

Dzięki temu po kompilacji zmiennym wewnątrz Data Blocków nadane zostaną odpowiednie offsety co później umożliwi ich adresowanie.

As	Astraada_HMI → PLC_1 [CPU 1212C AC/DC/Rly] → Program blocks → Data_block_1 [DB1]														
1	÷ 💣			· 📭 🖬 🗛 🚯 I	B) 🗮 🔢 🚏								<b>=</b>		
Data_block_1															
	1	Nam	e		Data type	Offset	Start value	Monitor value	Retain	Accessible f	Visible in	Setpoint	Comment		
1		• s	tatic												
2		•	r Va	r_1	"Some_type_1"	0.0				$\checkmark$	<b></b>				
З				Some_string_1	String	0.0				<b>V</b>	<b>~</b>				
4				Some_bool_1	Bool	256.0	false	FALSE		<b>V</b>	<b>~</b>				
5	-			Some_int_1	Int	258.0	0	0		$\checkmark$	$\checkmark$				
6	-			Some_real_1	Real	260.0	0.0	0.0		$\checkmark$	$\checkmark$				
7	-		•	Some_struct_1	Struct	264.0				$\checkmark$	$\checkmark$				
8	-			Some_string_2	String	0.0		"		$\checkmark$	$\checkmark$				
9	-			Some_int_2	Int	256.0	0	0		$\checkmark$	$\checkmark$				
10	-			Some_real_2	Real	258.0	0.0	0.0		$\checkmark$	$\checkmark$				
11	-			Some_array_1	Array[020] of Int	262.0				$\checkmark$	$\checkmark$				
12		•	Va	r_2	String	568.0		"		<b></b>	<b></b>				
13		•	Va	r_3	Int	824.0	0	0		<b></b>					
14		•	Va	r_4	String[30]	826.0		"		<b></b>	<b></b>				
15		•	Va	r_5	Int	858.0	0	0		<b></b>	<b></b>				
16		•	<a></a>	dd new>											

Zmienne wewnątrz data blocków adresować można w oprogramowaniu Astraada HMI CFG "ręcznie", przykładowo chcąc wymieniać ze sterownikiem wartość zmiennej Var\_3 należy użyć składni DB1.DBW824:

9999		NO_BDR	Borger Color: BG Color:			
DB1.DBW82	4 Data	Type: 16-Bit Sign	ed Integer	$\sim$		
	Displa	ay Type: 16-Bit Sign	ed Decimal	$\sim$		
-9999	9 Write	Address: DB1.DBW	324			
<b>o</b> o.		onitor address identical	to write address		Macro	
	Monit	or Address: DB1.DBW	824	<b>~</b> ⊔	Use	
	Font:	Font 1	~ Te	ext Color:		

W nieco bardziej skomplikowanym przykładzie – chcąc odczytywać zmienną Some\_real\_2 będącą częścią struktury w ramach własnego typu danych należy obliczyć offset dodając offsety kolejnych poziomów, czyli w tym przypadku offset struktury + offset zmiennej w ramach tej struktury: 264 + 258 = 522. Dodatkowo w tym przypadku odczytywana zmienna jest zmienną 32-bitową typu floating point, zatem w ustawieniach obiektu należy wybrać odpowiedni typ zmiennej:

9999	NO_BD	DR BG Color:	
	Data Type:	32-Bit Floating Point $$	
DB1:DBD522	Display Type:	32-bit Floating Point $\sim$	
-99.99 📭	Write Address:	DB1.DBD522	
	Monitor addre	ss identical to write address	Macro
D	Monitor Address:	DB1.DBD522	Use
	Font: Font 1	V Text Color:	
	Total Digits: 4	Fractional Digits: 2	

1	-	•	Sta	atic							0.940
2	-	•	•	Va	r_1	"Some_type_1"	0.0				QVVO
3			•		Some_string_1	String	0.0		"		
4			•		Some_bool_1	Bool	256.0	false	FALSE		281
5	-		•		Some_int_1	Int	258.0	0	0		
6	-		•		Some_real_1	Real	260.0	0.0	0.0		
7	-		•	•	Some_struct_1	Struct	264.0				DB1.DBD522
8	-			•	Some_string_2	String	0.0		"		
9	-00			•	Some_int_2	Int	256.0	0	0		25.50
10				•	Some_real_2	Real	258.0	0.0	25.5		25.50
11				•	Some_array_1	Array[020] of Int	262.0				
12		•		Va	r_2	String	568.0		"		
13	-	•		Va	r_3	Int	824.0	0	55		
14	-	•		Va	r_4	String[30]	826.0		"		
15		•		Va	r_5	Int	858.0	0	0		

Dodatkowo oprogramowanie Astraada HMI CFG wspiera dynamiczne adresowanie zmiennych w ramach tego drivera w oparciu o Index registers, przykładowo możliwe jest użycie poniższej adresacji:

DB[\$I3].DBW100

DB[\$I5+2].DBD[\$I10+16]

DB[\$I0+2].DBX[\$I1].5

### **OBSŁUGA WŁASNYCH TYPÓW DANYCH**

Oprogramowanie Astraada HMI CFG w wersjach 4.0.x.x umożliwia import własnych typów danych tworzonych w oprogramowaniu TIA Portal. Dla przykładu utworzono nowy typ o nazwie Some\_type\_1.

Project tree		Astraad	ia_HMI → PLC_1 [CPU 12	12C AC/DC/Rly] 🕨	PLC data types	Some_type	<u>∟</u> 1			_ <b>=</b> = ×
Devices										
1400	🔲 📑	22	<b>■ ■                    </b>	s 🖿 🔢 😤						<b></b>
		Som	e type 1							
<ul> <li>Astraada HMI</li> </ul>	0	N	lame	Data type	Default value	Accessible f	Visible in	Setnoint	Comment	
Add new device		1 📶	Some string 1	String					comment	
Devices & networks		2 -	Some bool 1	Bool	false			i iii		
✓ PLC 1 [CPU 1212C AC/DC/R]v]		3 - 60	Some int 1	Int	0					
Device configuration		4 -	Some real 1	Real	0.0					
Q Online & diagnostics		5 40	Some struct 1	Struct						
Program blocks	-	6 61	Some string 2	String						
Add new block	=	7 - 671 -	Some int 2	Int	0			i iii		=
Main [OB1]		8 - 57 -	Some real 2	Real	0.0					
Data block 1 [DB1]		9 - 61 -	<ul> <li>Some array 1</li> </ul>	ArravI0201 of Int						
Technology objects		10 📶	Some array 1[0]	Int	0					
External source files		11 🖘	Some array 1[1]	Int	0					
✓ PLC tags		12 📲	Some array 1[2]	Int	0					
Show all tags		13 📲	Some array 1[3]	Int	0					
Add new tag table		14 📶	Some array 1[4]	Int	0					
😤 Default tag table [31]		15 📶	<ul> <li>Some array 1[5]</li> </ul>	Int	0					
▼ B PLC data types		16 📶	<ul> <li>Some array 1[6]</li> </ul>	Int	0					
Add new data type		17 📲	Some array 1[7]	Int	0					
Some_type_1		18 🕣	Some_array_1[8]	Int	0					
Watch and force tables		19 📶	Some array 1[9]	Int	0					
Online backups		20 📶	<ul> <li>Some array 1[10]</li> </ul>	Int	0					
🕨 🔀 Traces		21 📶	<ul> <li>Some array 1[11]</li> </ul>	Int	0					
Device proxy data		22 📶	<ul> <li>Some array 1[12]</li> </ul>	Int	0					
Program info		23 📲	<ul> <li>Some_array_1[13]</li> </ul>	Int	0					
Text lists		24 🕣	Some_array_1[14]	Int	0					
Local modules		25 📶	<ul> <li>Some array 1[15]</li> </ul>	Int	0					*
Common data		<				111				>

Aby wyeksportować typ danych należy kliknąć PPM na nowym typie i wybrać opcję Generate source from blocks, a następnie zapisać w dowolnym miejscu na dysku komputera plik z rozszerzeniem .udt.

	🧠 Delault tag	tabi	e [51]	15 🔩	U
	🔻 🛅 PLC data types			16 🕣	1
	📑 Add new da	ta t	ype	17 🕣	1
	🖪 Some_type		Open		1
	Watch and forc		open		1
	🕨 📴 Online backup	Ж	Cut	Ctrl+X	1
	🕨 🔀 Traces		Сору	Ctrl+C	1
	Device proxy da		Paste	Ctrl+V	1
	Program info		Copy as text		1
	Text lists	×	Delete	Del	1
	Local modules		Rename	F2	1
	🕨 🏹 Common data		Compile	•	H
	🕨 🛅 Documentation se		Go online	Ctrl+K	
	🕨 🐻 Languages & reso	5	Go offline	Ctrl+M	2
•	🔚 Online access				H
_	🕈 Displav/hide interf	Ð	Generate source from blocks		Þ
<b>×</b>	Details view		Cross-reference information	Shift+F11	p
		×	Cross-references	F11	a
	Name	Ē	Call structure		•
1	Some_string_1		Assignment list		
-	Some_bool_1		Print	Ctrl+P	
•	Some_int_1	ł	Print preview		
-	Some_real_1	Q	Properties	Alt+Enter	
•	Some struct 1		Struct		1

W oprogramowaniu Astraada HMI CFG należy otworzyć okno tagów (klikając dwukrotnie PPM na obiekcie Tags w drzewie projektu) i wybrać z listy po lewej stronie link powiązany ze sterownikiem, kliknąć PPM na tym linku i wybrać opcję Import Data Type...



w 🗗 🔂 🖪 🖪			프 🛋 🎬 🛏	12 ab	📑 💡 🔏 123 abc 📖
~	Tag Table (A	AP_1)			
		1 🗙 🛛		a.	
* * * *	🔤 İnte	rnal Memory		Link 1	Data Type
~		Add Su	ıbgroup		Name
t Manager		Insert (	Group Above		
Siemens_HMI_2		Insert	Group Below		
Global		Delete	Group	_	
AP_I		Renam	e Group		
The Cloud		1000		_	
📲 Data Exchang		Preview	w/Import Tags		
MQTT Client		Export	Tags in the Group		
		Export	Group with Its Subgrou	ps	
		Import	Data Type		
Sound Table		Export	Data Type		
🖻 🙅 Setup				_	
General Setup		Load T	ags From OPC Server		
-Q. Clock		Defaul	t Setting of Exposed Tag	s	
🔏 User Access C		<u>E</u> xport	Exposed Tags	•	
🞸 Event Process					
Due Date Pass					
Screen					

Następnie po zaznaczeniu checkboxa w kolumnie Use należy wybrać odpowiedni plik z dysku i wybrać polecenie Convert.

Use	File path		Data type name
✓ 1	C:\Users\wojciechta\Desktop\Siem_HMI_test\Some_type_1.udt	Browse	Some_type_1
2		Browse	
3		Browse	
4		Browse	
5		Browse	
6		Browse	
7		Browse	
8		Browse	
PM Desi File pat	gner exported tag file (*.csv) h:		Browse

Po tej operacji w części okna po prawej stronie powinna pojawić się kompletna lista zawierająca typ zmiennej, całość należy zatwierdzić przyciskiem OK. Na liście w zakładce Data Type w ramach Tag Table powinny pojawić się zaimportowane typy danych.



### **IMPORT DATA BLOCKÓW**

Po utworzeniu Data Blocku zgodnie z wymienionymi wcześniej zaleceniami należy skompilować projekt i wykonać operację Generate source from blocks. Wygenerowany zostanie w ten sposób plik z rozszerzeniem .db, który należy zapisać w dowolnym miejscu na dysku komputera.



W oprogramowaniu Astraada HMI CFG w oknie Tag Table należy po kliknięciu PPM na odpowiednim linku komunikacyjnym wybrać opcję Preview/Import Tags...



Następnie należy zaznaczyć checkbox w kolumnie Use i wybrać wcześniej wygenerowany plik z rozszerzeniem .db dla Data Blocku z odpowiednim numerem.

Use	DB name	File path		Tag name prefix	
<b>∠</b> 1	DB1	C:\Users\wojciechta\Desktop\Siem_HMI_test\Data_block_1.db	Browse	Data_block_1	]
2	DB2		Browse		
3	DB3		Browse		
4	DB4		Browse		
5	DB5		Browse		
6	DB6		Browse		
7	DB7		Browse		
8	DB8		Browse		
TIA Port	al exported [	DB file (*.TIA)			
File pat	h:			Brow	/se
TIA Port	al exported t	ag file (*.XLSX)			
Cile anti	h.			Prov	



Po zatwierdzeniu przyciskiem Import powinna pojawić się kompletna lista zmiennych, należy kliknąć Import po raz drugi.

- 🔊 Tags	Name	Alias For	Data Type	Address	Scan Rate
	1 📮 tagDB1		DB_Data_block_1	DB1.DBB0	Normal
	1.1 HE Var_1		Some_type_1	DB1.DBB0	
	1.2 Uar_2		String	DB1.DBB568	
	1.3 Var_3		Int	DB1.DBW824	
	1.4 - 🖽 Var_4		String	DB1.DBB826	
	1.5 Var. 5		Int	DR1 DRW858	
	AND AND AND A			001.0000000	

Zmienne zostały dodane do projektu, można już zamknąć okno Tag Table.

Tag Table (AP_1)								×
🗃 🛱 🗙 🛋 🖻 🖻					💰 🗙	ኤ 🋍 👜		
Internal Memory	Link 1	Data Type						
Link 1		Name	Alias For	Data Type	Address S	can Rate Exposed	Description	^
-	1	📮 tagDB1	Í	DB_Data_block_1	DB1.DBB0 N	lormal		
1	1.1	- Var_1		Some_type_1	DB1.DBB0			
1	1.1.1	Some_string_1		String	DB1.DBB0			
1	1.1.1.1	- MaxLength		USInt	DB1.DBB0	N/A		
1	1.1.1.2	StringLength		USInt	DB1.DBB1	N/A		
1	1.1.1.3	Data		USInt[0253]	DB1.DBB2	N/A		
3	1.1.2	Some_bool_1		Bool	DB1.DBX256.0	N/A		
1	1.1.3	Some_int_1		Int	DB1.DBW258	N/A		
1	1.1.4			Real	DB1.DBD260	N/A		
1	1.1.5	Some_struct_1		Some_struct_1	DB1.DBB264			
1	1.1.5.1	Some_string_2		String	DB1.DBB264			
	1.1.5.1.1	MaxLength		USInt	DB1.DBB264	N/A		
1	1.1.5.1.2			USInt	DB1.DBB265	N/A		
1	1.1.5.1.3	Data		USInt[0253]	DB1.DBB266	N/A		
9	1.1.5.2	Some_int_2		Int	DB1.DBW520	N/A		
5	1.1.5.3	Some_real_2		Real	DB1.DBD522	N/A		
t	1.1.5.4	Some_array_1		Int[020]	DB1.DBW526	N/A		
1	1.2	- Var_2		String	DB1.DBB568			
1	1.2.1	- MaxLength		USInt	DB1.DBB568	N/A		
1	1.2.2			USInt	DB1.DBB569	N/A		
	1.2.3	Data		USInt[0253]	DB1.DBB570	N/A		
	1.3	Var_3		Int	DB1.DBW824	N/A		
1	1.4	Var_4		String	DB1.DBB826			
	1.4.1	MaxLength		USInt	DB1.DBB826	N/A		~
	<							>

Od tego momentu w obiektach w Astraada HMI CFG można odwoływać się do nazw zmiennych, obok pola do wpisywania adresu w ramach obiektów w oprogramowaniu znajduje się przycisk z etykietą:

Data Type:	16-Bit Unsigned Integer $\checkmark$	
Display Type:	16-Bit Unsigned Decimal $\qquad \qquad \lor$	
Write Address:		

Po jego użyciu pojawi się okno, w którym będzie można wybrać zmienną z listy, przykładowo chcąc odczytywać wartość z tablicy Some\_array\_1 o indeksie 2 w ramach zmiennej Var\_1 wybrano:

				Select Tag					×
				Constant					Q
				Search:					~
Jumeric	Entry			Link:	Link 1				~
General	Advance	d Visibility	Gradie	Tag Name:	tagDB	1.Var_1.Some_struct_1.Som	e_array_1[2]		New Tag >>
	Advances	a visibility	Gradie	Base Tag/Ad	dress:	DB1.DBW530	Bit Number:	$\sim$	List Options
ID:	1E0000	Note:		🖃 🗰 Link	1				
			🗌 Tra		tagDB1	L			
- 60			Shap	÷	🕮 Var	_1			
					·····	Some_string_1 Some_bool_1			
- 82		_	Borde			Some_int_1			
	NO_BD	R	BG Co			Some_real_1			
Data T	VDet	16-Bit Unsig	ned Int			······································			
Data i	ype.	To bit onsig	neu inc		-	Some_int_2			
Display	/Type:	16-Bit Unsig	ned De		_	Some_real_2			
Write /	Address:				₽.		1		
Mor	nitor addres	ss identical to	o write a						
Monito	r Address:								
						- [4]			
Font:	Font 1		~			- C [5]			
Total D	Digits: 4	-	Fractio						
Nur	mbers Of To	otal And Frac	tional D						
						[9] [10]			
- Alian	ment	Justification							
	off	Zero Sun	press			<b>[12]</b>			
			7			- [13] - [14]			
00	enter		zeros			<b>[15]</b>			
ORi	ight	O Leading !	Spaces			<b>[16]</b>			¥
							Select		Cancel

Po skompilowaniu i wgraniu programów do urządzeń komunikacja zostanie nawiązana:

7	-	-	•	Some_struct_1	Struct	264.0		
8	-	-	•	Some_string_2	String	0.0		n
9	-	-	•	Some_int_2	Int	256.0	0	0
1	0 -		•	Some_real_2	Real	258.0	0.0	25.5
1	1 -	-	•	<ul> <li>Some_array_1</li> </ul>	Array[020] of Int	262.0		
1	2 -			<ul> <li>Some_array_1[0]</li> </ul>	Int	0.0	0	0
1	3 -			<ul> <li>Some_array_1[1]</li> </ul>	Int	2.0	0	0
1	4 -	-		<ul> <li>Some_array_1[2]</li> </ul>	Int	4.0	0	15
1	5 -			<ul> <li>Some_array_1[3]</li> </ul>	Int	6.0	0	0

15