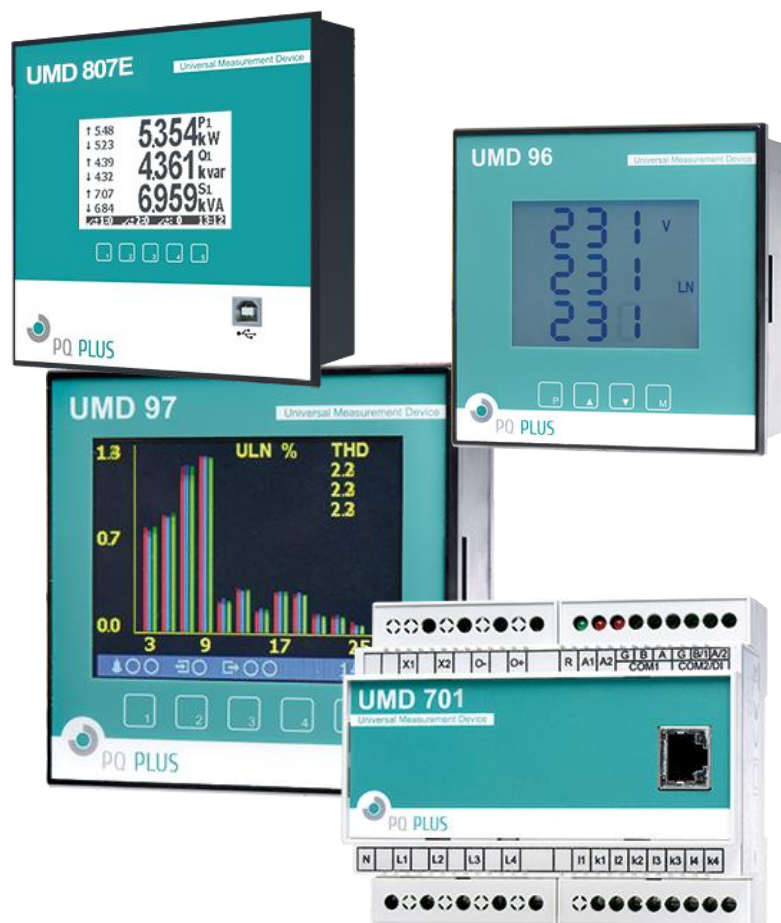


Softwarebeschreibung

Modbus Kopplung Simatic S7 TIA

mit Simatic S7 300/400 über Profinet (OnBoard Ethernet)

V2.0 - 1/2016



Inhaltsverzeichnis

1. Haftung und Gewährleistung.....	3
2. TIA Portal V12/13	4
2.1 Lieferumfang	4
2.2 Aufruf im Programm.....	5
2.3 FB 192 Parameter	5
2.3.1 Datenablage im RX_DB	7
2.4 Beispiel eines Programmaufrufes.....	8
2.5 INDEX	9

1. Haftung und Gewährleistung

Der Inhalt dieser Dokumentation ist nicht Teil einer früheren oder bestehenden Vereinbarung, Zusage oder eines Rechtsverhältnisses oder soll dieses abändern.

Die Fa. PQ Plus GmbH weist darauf hin, dass der Benutzer für den Einsatz der Software und deren programmtechnischen Einbindung selbst verantwortlich ist.

Die Verantwortung für die vorschriftsmäßige Errichtung der Gesamtanlage hat in diesem Fall der Anlagenbetreiber oder dessen Generalauftragnehmer.

Software

Da Daten unter bestimmten Bedingungen in jedem elektronischen Speicher verloren gehen oder geändert werden, übernimmt die Fa. PQ Plus GmbH keine Haftung für Daten, die aufgrund missbräuchlicher Verwendung, Reparaturen bzw. Defekten an der Hardware oder aus irgendwelchen anderen Gründen verloren gehen oder anderweitig unbrauchbar werden.

HINWEIS

Die Fa. PQ Plus GmbH übernimmt keine Haftung, direkt oder indirekt, für finanzielle Verluste oder Schadensansprüche Dritter, die aus der Nutzung dieser Software und seiner Funktionen entstehen.

Zur Nutzung der Software ist nur der jeweilige Vertragspartner berechtigt, eine Weitergabe an Dritte ist nicht gestattet. Diese Bedienungsanleitung wurde mit der gebotenen Sorgfalt erarbeitet; es wird keinerlei Haftung für Fehler oder Auslassungen übernommen. Ebenso wird für Schäden die sich aus der Nutzung von Informationen dieser Bedienungsanleitung ergeben nicht gehaftet.

2. TIA Portal V12/13

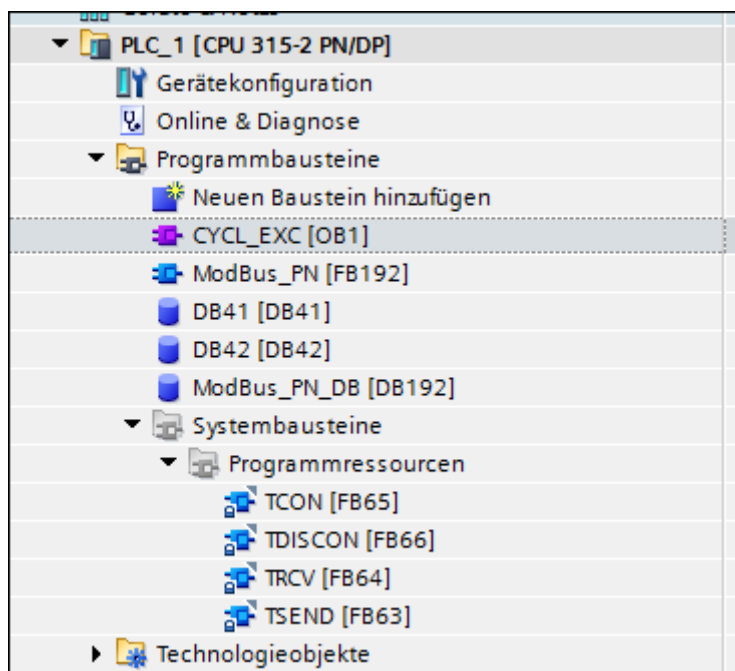
- FB 192 wird mit einem zugeordneten Instanz-DB aufgerufen. Die Nummer des DB's ist frei wählbar.
- FB 192 kann auch umbenannt werden.
- FB 192 ist Multi-Instanz fähig.
- Der Aufruf erfolgt im OB1.
- Es werden folgende umbenannte Siemens Standard-Bausteine verwendet (im Projektumfang enthalten):

FB 1063 – SendData	(FB63)
FB 1064 – ReceiveData	(FB64)
FB 1065 – Connect	(FB65)
FB 1066 – Disconnect	(FB66)

Die Standard FB's können nicht umbenannt werden

2.1 Lieferumfang

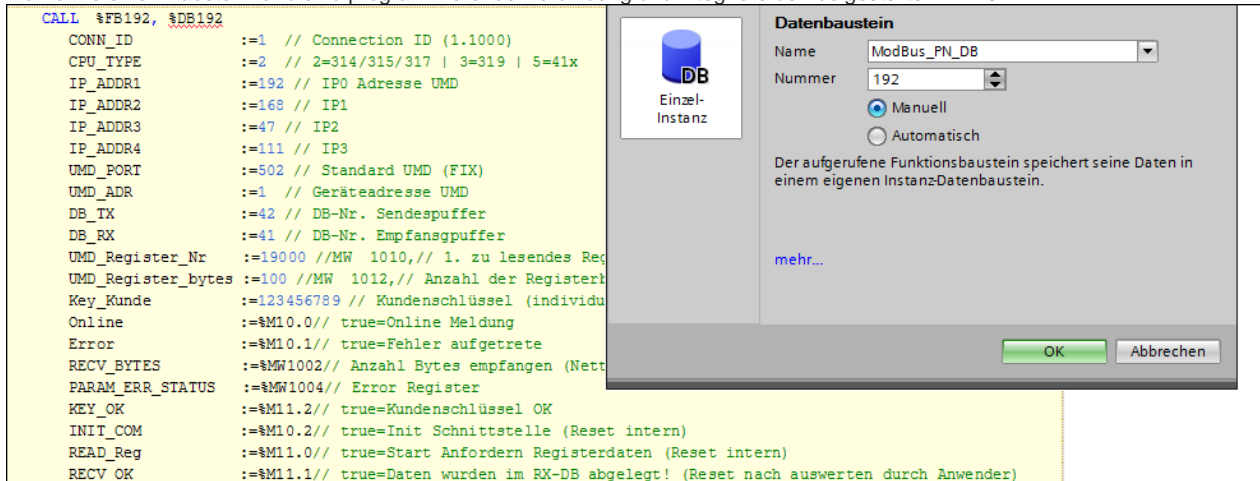
Der Lieferumfang besteht aus einem TIA Projekt (V13) „TIA_UMD20“.



Lieferumfang des TIA – Projektes

2.2 Aufruf im Programm

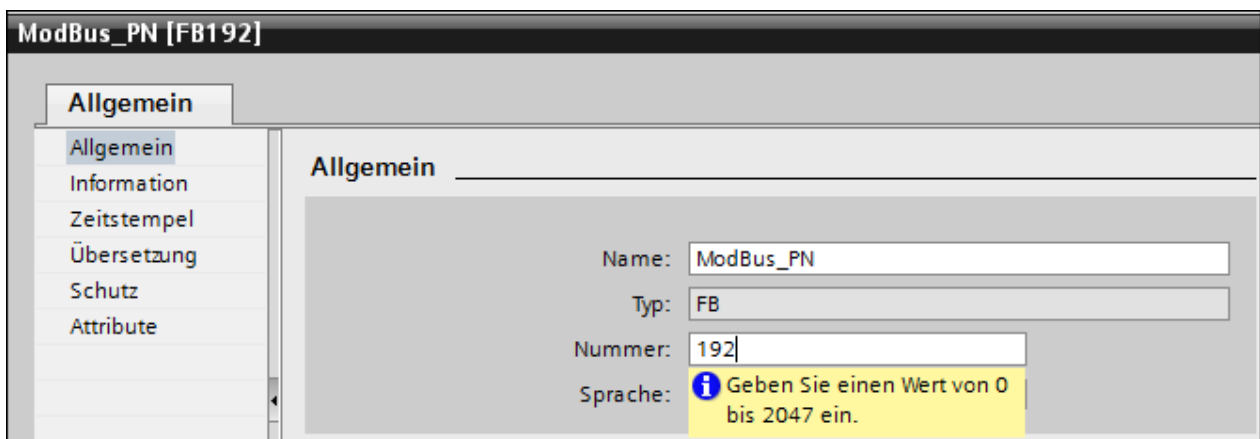
Man öffne einen Baustein für die zu programmierende Verbindung und integriere den beigestellten FB 192.



Beim Anlegen wird nach dem gewünschten Instanz DB gefragt.

Hinweis:

Die Bausteinnummer FB192 ist jederzeit änderbar



2.3 FB 192 Parameter

In der folgenden Tabelle werden die Input Parameter zusammengestellt:

ModBus_PN						
Name	Datentyp	Offset	Kommentar
Input						
CONN_ID	Int	0.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Connection ID (je Aufruf eigene ID) [1.....
CPU_TYPE	Int	2.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0=CP443 2=315/317 3=319 5=413/...
IP_ADDR1	Int	4.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	UMD IP1
IP_ADDR2	Int	6.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	UMD IP2
IP_ADDR3	Int	8.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	UMD IP3
IP_ADDR4	Int	10.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	UMD IP4
UMD_PORT	Int	12.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	PortNr. UMD (Standard 502)
UMD_ADR	Int	14.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Geräteadresse UMD (1...x)
DB_TX	Int	16.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DB Nr zum Senden
DB_RX	Int	18.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DB Nr zum Empfangen Datenangaben ...
UMD_Register_Nr	Int	20.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Registernummer zum Auslesen
UMD_Register_bytes	Int	22.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Anzahl Bytes zum Lesen
Key_Kunde	DInt	24.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Key des Kunden -----

Input	Datentyp	Beschreibung
CONN_ID	INT	Nummer der Verbindung [1..1000]
CPU_TYPE	INT	Beschreibt die verwendete CPU / CP Type: 0-CP 443 2-CPU 315/317 3-CPU 319 4-CPU 4xx
IP_ADDR1	INT	IP-Adresse Teil 1 (z.B. 192)
IP_ADDR2	INT	IP-Adresse Teil 2 (z.B. 168)
IP_ADDR3	INT	IP-Adresse Teil 3 (z.B. 104)
IP_ADDR4	INT	IP-Adresse Teil 4 (z.B. 47)
UMD_PORT	INT	Nummer des UMD Ports des Gerätes (Default: 502)
UMD_ADR	INT	Nummer der Modbusadresse des UMD Gerätes (z.B. 1)
DB_RX	INT	Nummer eines Datenbausteines zum Senden von Daten über den Modbus (Länge: mind. 150 Worte) Bereich: 1 – 16.000. Dieser DB wird intern verwendet und sollte vom Anwender nicht geändert werden.
DB_TX	INT	Nummer eines Datenbausteines zum Empfangen von Daten über den Modbus (Länge: mind. 12 Bytes). Bereich: 1 – 16.000 Nach Übergabe der Parameter und dem Start mit <i>READ_Reg</i> = TRUE werden bei fehlerfreier Ausführung die Daten in dem DB ab Datenwort DBW0 abgelegt. Je nach Format des/der gelesenen Register müssen die entsprechenden Formate im DB zur Auswertung gewählt werden (z.B. Simatic Gleitpunkt für Float Register).
UMD_Register	INT	Angabe des ersten zu lesenden Registers (Nummern siehe UMD Anleitung PQ Plus) (0..32767)
UMD_Register_bytes	INT	Anzahl der zu lesenden Bytes Bereich: 2 – 160 Es können somit auch Registerbereiche gelesen werden (z.B. 12 Byte entsprechen 3 Float-Register (für 3 Phasenströme o.ä.)).
Key_Kunde	DINT	Angabe des mitgelieferten Kundenschlüssels (Key)

In der folgenden Tabelle werden die OUTPUT Parameter zusammengestellt:

ModBus_PN						
Name	Datentyp	Offset	Kommentar
Input						
Output						
Online	Bool	28.0	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Online flag
Error	Bool	28.1	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Fehler bei Operation
RECV_BYTES	Int	30.0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Anzahl gelesene Bytes vom UMD
PARAM_ERR_STATUS	Word	32.0	16	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Erroranzeige bei Parameterübergabe
KEY_OK	Bool	34.0	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Key OK
InOut						

Output	Datentyp	Beschreibung
Online	BOOL	Zeigt den Status Verbindung zum UMD an: TRUE: Verbindung ist aufgebaut FALSE: keine Verbindung ist aufgebaut
Error	BOOL	Signalisiert einen Fehler bei der Parameterübergabe des Anwenders. TRUE: Fehler in den Parametern, Detailinformation findet man unter <i>PARAM_ERR_STATUS</i>
RECV_BYTES	INT	Anzahl der empfangenen Bytes im <i>DB_RX</i>
PARAM_ERR_STATUS	WORD	0h: kein Fehler aufgetreten 8xxxh: Fehler bei Parameterübergabe (Details siehe Liste im STEP7 Projekt) Fehler können sein: DB nicht vorhanden, DB zu kurz, CPU type falsch, IP Adresse ungültig usw.
KEY_OK	BOOL	TRUE: Schlüsselnummer OK

Vor einem Datenabruf muss das Flag „KEY_OK“ = true sein, ebenso muss das Flag „Online“ = true sein.

In der folgenden Tabelle werden die IN OUT Parameter zusammengestellt:

ModBus_PN						
Name	Datentyp	Offset	Kommentar
▶ Input				<input type="checkbox"/>	<input type="checkbox"/>	
▶ Output				<input type="checkbox"/>	<input type="checkbox"/>	
▼ InOut				<input type="checkbox"/>	<input type="checkbox"/>	
INIT_COM	Bool	36.0	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Anzeige das UMD Online ist
READ_Reg	Bool	36.1	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Start Lese Register vom UMD
RECV_OK	Bool	36.2	fa!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Daten vorhanden im DB_RX _____...

Output	Datentyp	Beschreibung
INIT_COM	BOOL	Muss vom Anwender auf TRUE gesetzt werden, wenn die Verbindung neu initialisiert werden soll (z.B. nach Netz-Aus oder beim Auftreten der Offline-Meldung)
READ_Reg	BOOL	Nach dem Eintragen der gewünschten Registernummer(n) und der Anzahl der Bytes wird der Lesevorgang durch Setzen auf TRUE gestartet. Nach Ausführung wird die Variable vom Treiber auf FALSE gesetzt. Wenn das Lesen der Register erfolgreich war, wird das Flag <i>RECV_OK</i> auf TRUE gesetzt, die Anzahl der Bytes wird in <i>RECV_BYTES</i> eingetragen und die Daten werden im DB_RX ab Datenwort 0 eingetragen. Wenn das Lesen der Register nicht erfolgreich war, wird das Flag <i>Error</i> auf TRUE gesetzt und die Fehlernummer wird in der Variable <i>PARAM_ERR_STATUS</i> eingetragen.
RECV_OK	BOOL	FALSE: kein Auftrag gelesen TRUE: Daten wurden empfangen. Das Bit sollte vom Anwender nach der Verarbeitung der Daten wieder auf FALSE gesetzt werden.

2.3.1 Datenablage im RX_DB

Die empfangenen Daten werden im **DB_RX** abgelegt (im Beispiel DB41) und es wird das Bit **RECV_OK** gesetzt. Die Nettodaten werden ab Byte 9 im DB abgelegt (DBB9 und folgende).

%DB41.DBD9	Gleitpunktzahl	228.7341	<input type="checkbox"/>	Netto Daten 1.DD
%DB41.DBD13	Gleitpunktzahl	0.06358746	<input type="checkbox"/>	Netto Daten 2.DD
%DB41.DBD17	Gleitpunktzahl	0.04129949	<input type="checkbox"/>	Netto Daten 3.DD
%DB41.DBD21	Gleitpunktzahl	228.6857	<input type="checkbox"/>	Netto Daten 4.DD

Beispiel: Empfang von 16 Bytes

Abgefragt wurden die Register 4352/4352/4356 [1100/1102/11104hex] (U1/U2/U3) des UMD97.

2.4 Beispiel eines Programmaufrufes

Im folgenden Beispiel wird der Treiber im FC2 aufgerufen.

```

CALL %FB192, %DB192
  CONN_ID      :=1 // Connection ID (1.1000)
  CPU_TYPE     :=2 // 2=314/315/317 | 3=319 | 5=41x
  IP_ADDR1     :=192 // IP0 Adresse UMD
  IP_ADDR2     :=168 // IP1
  IP_ADDR3     :=47 // IP2
  IP_ADDR4     :=111 // IP3
  UMD_PORT     :=502 // Standard UMD (FIX)
  UMD_ADR      :=1 // Geräteadresse UMD
  DB_TX        :=42 // DB-Nr. Sendepuffer
  DB_RX        :=41 // DB-Nr. Empfangspuffer
  UMD_Register_Nr :=19000 //MW 1010, // 1. zu lesendes Register (z.B. 19000)
  UMD_Register_bytes :=100 //MW 1012, // Anzahl der Registerbytes (2/4...100)
  Key_Kunde    :=123456789 // Kundenschlüssel (individuell)
  Online       :=%M10.0// true=Online Meldung
  Error        :=%M10.1// true=Fehler aufgetrete
  RECV_BYTES   :=%MW1002// Anzahl Bytes empfangen (Netto Daten)
  PARAM_ERR_STATUS :=%MW1004// Error Register
  KEY_OK       :=%M11.2// true=Kundenschlüssel OK
  INIT_COM     :=%M10.2// true=Init Schnittstelle (Reset intern)
  READ_Reg     :=%M11.0// true=Start Anfordern Registerdaten (Reset intern)
  RECV_OK      :=%M11.1// true=Daten wurden im RX-DB abgelegt! (Reset nach auswerten durch Anwender)
  
```

Zur Erklärung der Parameterversorgung:

CPU_TYPE: **2** steht für CPU315/317

Die IP-Adresse des UMD wird mit **192.168.47.111** eingestellt.

Der Port des UMD wird auf **502** eingestellt.

Die Geräteadresse Modbus UMD wird auf **1** eingestellt.

Als Sendebaustein wird der DB 42 verwendet, als Empfangsbaustein wird der DB 41 verwendet.

Das/die zu lesende(n) Register wird im Beispiel mit **4100 [auf MW1010 geschrieben]** (Istfrequenz) vom Anwender übergeben.

Die Anzahl zu lesender Registerbytes werden im Beispiel mit **4 [auf MW1012 geschrieben]** vom Anwender übergeben.

Eine Übergabe über Datenworte ode o.ä. ist ebenfalls möglich.

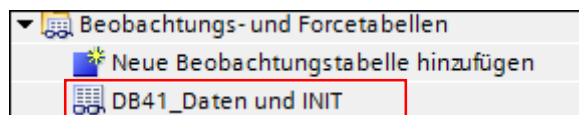
Die Anzahl der empfangenen Netto Bytes werden vom Treiber im MW1002 eingetragen.

Für die Status anzeigen (BOOL) werden Merker verwendet.

Key_Kunde:

Hier muss die mitgelieferte Schlüsselzahl eigegeben werden, um eine Kommunikation zu ermöglichen. Nach 10 Versuchen mit falschen Keycode wird die Verbindung blockiert. Danach muss man den Instanz-DB neu laden, um wieder eine Kommunikation mit dem korrekten Key aufbauen zu können.

Beispiel: Lese Register **4352 - 4356 – Voltage U1-N / U2-N / U3-N**



(Beobachtungstabelle siehe Projekt)

TIA_MODBx ▶ CPU 317-2 PN/DP [CPU 317-2 PN/DP] ▶ Beobachtungs- und Forcetabellen ▶ DB41_Daten und INIT

	i	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert		Kommentar
1		*ONLINE*	%M10.0	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	
2		*ERROR*	%M10.1	BOOL	<input type="checkbox"/> FALSE		<input type="checkbox"/>	
3		*REQU_DATA*	%M11.0	BOOL	<input type="checkbox"/> FALSE	TRUE	<input checked="" type="checkbox"/>	
4		*RX_DATAs*	%M11.1	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	
5		*Key_OK*	%M11.2	BOOL	<input checked="" type="checkbox"/> TRUE		<input type="checkbox"/>	
6		*INIT*	%M10.2	BOOL	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	
7		*Register-Nr*	%MW1010	DEZ+/-	19000	19000	<input checked="" type="checkbox"/>	
8		*Anzahl Bytes*	%MW1012	DEZ+/-	16	16	<input checked="" type="checkbox"/>	
9			%DB41.DBD9	Gleitpunktzah	231.1361		<input type="checkbox"/>	Netto Daten 1.DD
10			%DB41.DBD13	Gleitpunktzah	0.06207862		<input type="checkbox"/>	Netto Daten 2.DD
11			%DB41.DBD17	Gleitpunktzah	0.04567968		<input type="checkbox"/>	Netto Daten 3.DD
12			%DB41.DBD21	Gleitpunktzah	231.091		<input type="checkbox"/>	Netto Daten 4.DD
13		*RX_Bytes*	%MW1002	DEZ+/-	25		<input type="checkbox"/>	
14		*ERR_Nr*	%MW1004	Hex	16#0000		<input type="checkbox"/>	
15		<input type="text" value=""/>	<Hinzufügen>				<input type="checkbox"/>	

2.5 INDEX

B

Beispiel eines Programmaufrufes · 10
Beispiel Phasenspannung · 11

C

Connection ID · 6
CPU Typ · 6

E

Einbindung ins Anwenderprogramm · 5

F

FB192 Parameter_TIA · 6

H

Haftung · 3

I

In_Out parameter · 8

Input parameter · 6
IP Adresse · 6

L

Lieferumfang · 4

O

Output parameter · 7

P

Port Nummer · 6
 Programmaufruf · 5

T

TIA V12 · 4

U

UMG_Modbus