

SIEMENS



FAQ • 12/2014

# MODBUS-TCP mit den Anweisungen MB\_CLIENT und MB\_SERVER

S7-1200

<http://support.automation.siemens.com/WW/view/de/83130159>

---

Dieser Beitrag stammt aus dem Siemens Industry Online Support. Es gelten die dort genannten Nutzungsbedingungen ([www.siemens.com/nutzungsbedingungen](http://www.siemens.com/nutzungsbedingungen)).

### **Vorsicht**

Die in diesem Beitrag beschriebenen Funktionen und Lösungen beschränken sich überwiegend auf die Realisierung der Automatisierungsaufgabe. Bitte beachten Sie darüber hinaus, dass bei Vernetzung Ihrer Anlage mit anderen Anlagenteilen, dem Unternehmensnetz oder dem Internet entsprechende Schutzmaßnahmen im Rahmen von Industrial Security zu ergreifen sind. Weitere Informationen dazu finden Sie unter der Beitrags-ID 50203404.

<http://support.automation.siemens.com/WW/view/de/50203404>

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung .....</b>	<b>4</b>
<b>2</b>	<b>Anwenderprogramm Modbus TCP-Client .....</b>	<b>5</b>
2.1	FC1 „Read_HoldingRegister“ .....	5
2.2	FC4 „Write_HoldingRegister“ .....	7
2.3	Ein- und Ausgangsparameter der Anweisung „MB_CLIENT“ .....	9
2.4	Parameter MB_MODE und MB_DATA_ADDR .....	10
<b>3</b>	<b>Anwenderprogramm Modbus TCP-Server .....</b>	<b>12</b>
3.1	FC1 „Read_HoldingRegister“ .....	12
3.2	FC4 „Write_HoldingRegister“ .....	13
3.3	Ein- und Ausgangsparameter der Anweisung „MB_SERVER“ .....	15
<b>4</b>	<b>Bedienung des Programmbeispiels .....</b>	<b>17</b>
4.1	Bedienung des Anwenderprogramms im Modbus TCP-Client .....	17
4.2	Bedienung des Anwenderprogramms im Modbus TCP-Server .....	19

# 1 Einleitung

In diesem Beispiel wird die Modbus-TCP-Kommunikation zwischen zwei S7-1200 CPUs gezeigt. Im Anwenderprogramm der S7-1200 CPU werden die Anweisungen „MB\_CLIENT“ oder „MB\_SERVER“ aufgerufen und parametrisiert.

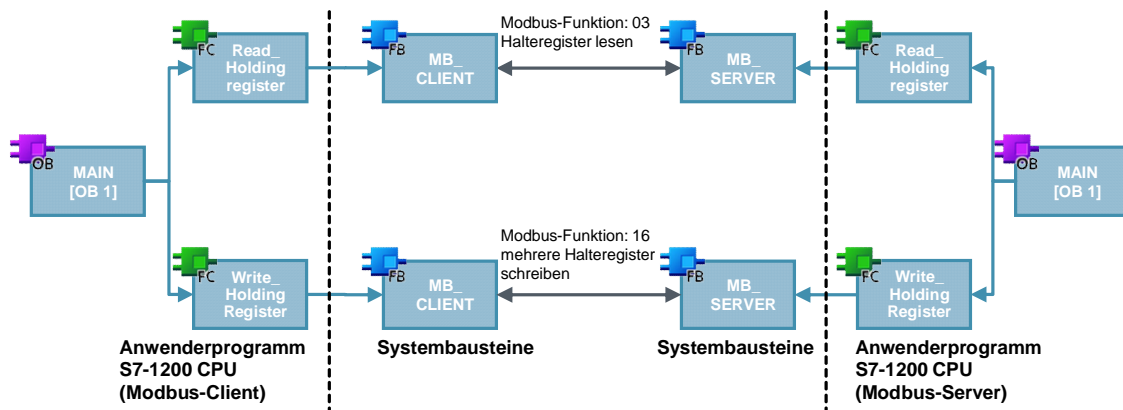
Die Anweisung "MB\_CLIENT" kommuniziert als MODBUS TCP-Client über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul. Über die Anweisung "MB\_CLIENT" bauen Sie eine Verbindung zwischen den Client und Server auf, senden Anfragen und empfangen Antworten und steuern den Verbindungsabbau des Modbus TCP-Servers.

Die Anweisung "MB\_SERVER" kommuniziert als MODBUS TCP-Server über die PROFINET-Verbindung der S7-1200 CPU. Zur Verwendung der Anweisung benötigen Sie kein zusätzliches Hardwaremodul. Die Anweisung "MB\_SERVER" verarbeitet Verbindungsanfragen eines Modbus TCP-Client, empfängt Anfragen von Modbus-Funktionen und sendet Antwort-Meldungen.

In diesem Beispiel werden 2 Modbus-Funktionen demonstriert. Für jede Modbus-Funktion wird über ein Modbus-Bausteinpärchen (MB\_CLIENT und MB\_SERVER) eine Modbus-TCP-Verbindung aufgebaut.

[Abbildung 1-1](#) zeigt eine Übersicht der demonstrierten Modbus-Funktionen und die Zuordnung der Bausteinpärchen.

Abbildung 1-1



## 2 Anwenderprogramm Modbus TCP-Client

Im Anwenderprogramm des Modbus TCP-Client wird die Anweisung "MB\_CLIENT" für jede Modbus-TCP-Verbindung mit einer eindeutigen ID und einem eigenen Instanz-Datenbaustein aufgerufen. Der Aufruf der Anweisung „MB\_CLIENT“ erfolgt jeweils in einer eigenen Funktion.

Tabelle 2-1

ID	Aufruf der Anweisung „MB_CLIENT“	Instanz-DB der Anweisung „MB_CLIENT“	Beschreibung
1	FC1 „Read_HoldingRegister“	DB2 „MB_CLIENT_DB“	Halteregister lesen
4	FC4 „Write_HoldingRegister“	DB5 „MB_CLIENT_DB_3“	Halteregister schreiben

### 2.1 FC1 „Read\_HoldingRegister“

Die Funktion FC1 „Read\_HoldingRegister“ ruft intern die Anweisung „MB\_CLIENT“ auf, um die Modbus-TCP-Verbindung mit der ID=1 aufzubauen und das Halteregister zu lesen.

Die Kommunikationsanfrage zum Lesen des Halteregisters wird über den Merker M1.0 am Eingang REQ gesteuert.

In diesem Beispiel wird die Modbus-TCP-Verbindung mit der Verbindungsnummer=1 zum Port 502 des Modbus TCP-Server aufgebaut. Der Modbus TCP-Server hat die IP-Adresse 192.168.0.30.

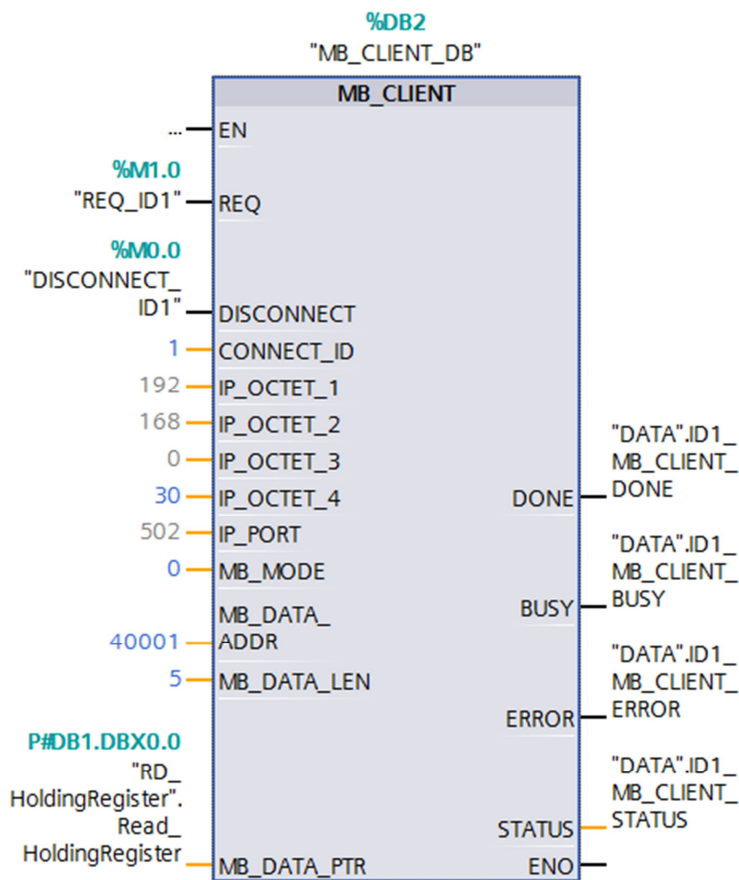
Es werden 5 Datenwörter aus dem Halteregister gelesen. Dafür parametrieren Sie die Eingangsparameter MB\_MODE, MB\_DATA\_ADDR und MB\_DATA\_LEN wie folgt:

- MB\_MODE = 0
- MB\_DATA\_ADDR = 40001
- MB\_DATA\_LEN = 5

**Hinweis** Im Kapitel [2.4 „Parameter MB\\_MODE und MB\\_DATA\\_ADDR“](#) finden Sie eine ausführliche Beschreibung der Parameter MB\_MODE und MB\_DATA\_ADDR.

Die [Abbildung 2-1](#) zeigt den Aufruf und die Parametrierung der Anweisung „MB\_CLIENT“ im FC1.

Abbildung 2-1



Copyright © Siemens AG 2014 All rights reserved

**Hinweis** Im Kapitel [2.3 Ein- und Ausgangsparameter der Anweisung „MB\\_CLIENT“](#) finden Sie eine Übersicht und Beschreibung der Ein- und Ausgangsparameter der Anweisung „MB\_CLIENT“.

### Empfangspuffer

Am Parameter MB\_DATA\_PTR geben Sie den Puffer für die vom Modbus TCP-Server empfangenen Daten an. Die aus dem Halteregeister gelesenen Daten werden im DB1 „RD\_HoldingRegister“ ab Adresse 0 gespeichert.

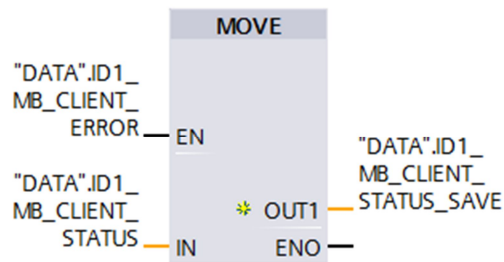
Tabelle 2-2

Variablenname	Datentyp	Adresse im DB1
Read_HoldingRegister	Array [0..4] of Word	0.0

### Fehlerauswertung

Wenn die Anweisung „MB\_CLIENT“ im FC1 mit Fehler beendet wird, dann wird der Fehlercode des Parameters STATUS für eine Fehlerauswertung im DB7 „DATA“ in der Variablen „ID1\_MB\_CLIENT\_STATUS\_SAVE“ gespeichert.

Abbildung 2-2



## 2.2 FC4 „Write\_HoldingRegister“

Die Funktion FC4 „Write\_HoldingRegister“ ruft intern die Anweisung „MB\_CLIENT“ auf, um die Modbus-TCP-Verbindung mit der ID=4 aufzubauen und in das Halteregeister des Modbus TCP-Server zu schreiben.

Die Kommunikationsanfrage zum Schreiben in das Halteregeister wird über den Merker M1.3 am Eingang REQ gesteuert.

In diesem Beispiel wird die Modbus-TCP-Verbindung mit der Verbindungsnummer=4 zum Port 505 des Modbus TCP-Server aufgebaut. Der Modbus TCP-Server hat die IP-Adresse 192.168.0.30.

Es werden 5 Datenwörter in das Halteregeister des Modbus TCP-Server geschrieben. Dafür parametrieren Sie die Eingangsparameter MB\_MODE, MB\_DATA\_ADDR und MB\_DATA\_LEN wie folgt:

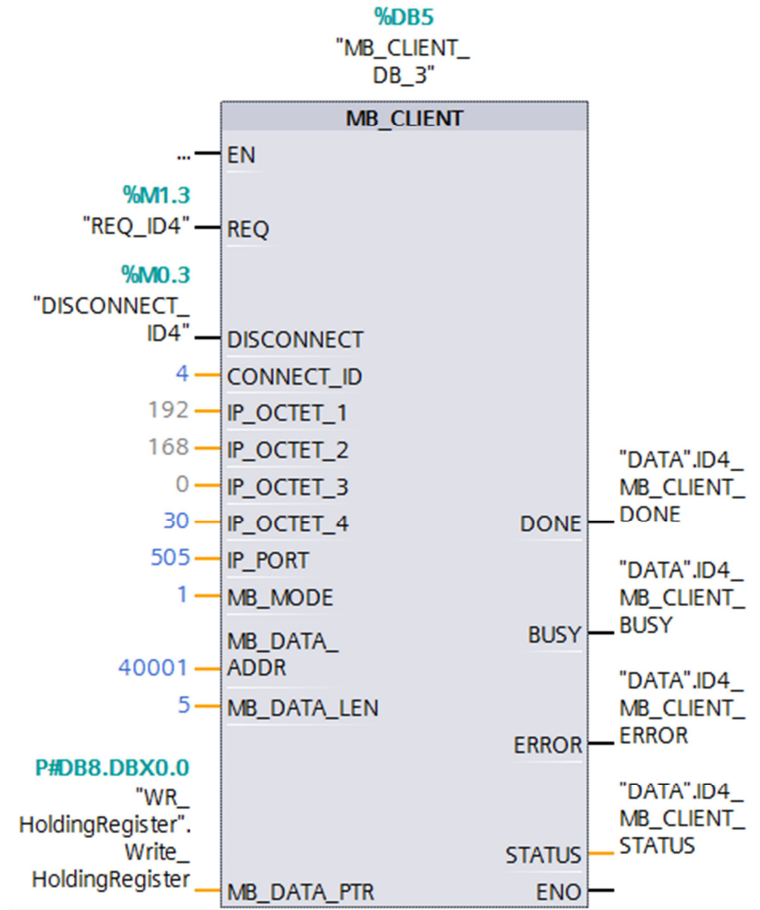
- MB\_MODE = 1
- MB\_DATA\_ADDR = 40001
- MB\_DATA\_LEN = 5

### Hinweis

Im Kapitel [2.4 „Parameter MB\\_MODE und MB\\_DATA\\_ADDR“](#) finden Sie eine ausführliche Beschreibung der Parameter MB\_MODE und MB\_DATA\_ADDR.

Die [Abbildung 2-3](#) zeigt den Aufruf und die Parametrierung der Anweisung „MB\_CLIENT“ im FC4.

Abbildung 2-3



Copyright © Siemens AG 2014 All rights reserved

**Hinweis** Im Kapitel [2.3 Ein- und Ausgangsparameter der Anweisung „MB\\_CLIENT“](#) finden Sie eine Übersicht und Beschreibung der Ein- und Ausgangsparameter der Anweisung „MB\_CLIENT“.

**Sendepuffer**

Am Parameter MB\_DATA\_PTR geben Sie den Puffer für die zum Modbus TCP-Server zu sendenden Daten an. Die in das Halteregeister zu schreibenden Daten sind im DB8 „WR\_HoldingRegister“ ab Adresse 0 gespeichert.

Tabelle 2-3

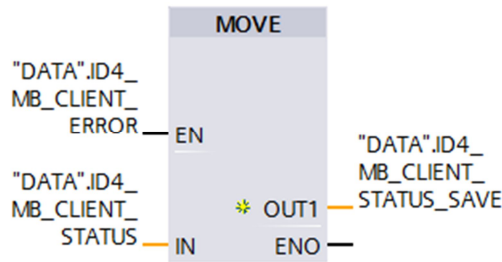
Variablenname	Datentyp	Adresse im DB1
Write_HoldingRegister	Array [0..4] of Word	0.0

**Fehlerauswertung**

Wenn die Anweisung „MB\_CLIENT“ im FC4 mit Fehler beendet wird, dann wird der Fehlercode des Parameters STATUS für eine Fehlerauswertung im DB7 „DATA“ in der Variablen „ID4\_MB\_CLIENT\_STATUS\_SAVE“ gespeichert.



Abbildung 2-4



## 2.3 Ein- und Ausgangsparameter der Anweisung „MB\_CLIENT“

### Eingangsparameter

Die Anweisung „MB\_CLIENT“ hat folgende Eingangsparameter.

Tabelle 2-4

Eingangsparameter	Datentyp	Beschreibung
REQ	BOOL	Kommunikationsanfrage mit dem Modbus TCP-Server bei steigender Flanke. <b>Hinweis</b> Mit der Kommunikationsanfrage wird der Instanz-DB für andere Clients gesperrt. Änderungen an den Eingangsparametern wirken sich erst aus, wenn eine Antwort des Servers erfolgt ist oder eine Fehlermeldung ausgegeben wurde. Wird während einer laufenden Anfrage der Parameter REQ erneut gesetzt, wird im Anschluss keine weitere Übertragung durchgeführt.
DISCONNECT	BOOL	Über den Parameter steuern Sie den Verbindungsaufbau und –abbau zu dem Modbus TCP-Server. <ul style="list-style-type: none"> <li>0: Kommunikationsverbindung zu der angegebenen IP-Adresse und Port-Nummer herstellen</li> <li>1: Kommunikationsverbindung trennen. Während des Verbindungsabbaus wird keine andere Funktion ausgeführt. Nach dem erfolgreichen Abbau der Verbindung wird am Parameter STATUS der Wert 7003 ausgegeben.</li> </ul> Ist der Parameter REQ beim Verbindungs-aufbau gesetzt, wird die Anfrage sofort gesendet.
CONNECT_ID	UINT	Eindeutige ID zur Identifizierung der Verbindung. Jeder Instanz der Anweisung „MB_CLIENT“ muss eine eindeutige Verbindungs- ID zugewiesen werden.
IP_OCTET_1	USINT	Erstes Oktett der IP-Adresse des Modbus TCP-Servers.
IP_OCTET_2	USINT	Zweites Oktett der IP-Adresse des Modbus TCP-Servers.
IP_OCTET_3	USINT	Drittes Oktett der IP-Adresse des Modbus TCP-Servers.

Eingangsparameter	Datentyp	Beschreibung
IP_OCTET_4	USINT	Viertes Oktett der IP-Adresse des Modbus TCP-Servers.
IP_PORT	UINT	IP-Port-Nummer des Servers, zu dem der Client die Verbindung herstellt und über das TCP/IP-Protokoll kommuniziert. (Standard-Wert: 502)
MB_MODE	USINT	Auswahl des Modus der Anfrage Im Kapitel <a href="#">2.4 „Parameter MB_MODE und MB_DATA_ADDR“</a> finden Sie eine ausführliche Beschreibung des Parameters MB_MODE.
MB_DATA_ADDR	UDINT	Anfangsadresse der Daten, auf welche die Anweisung „MB_CLIENT“ zugreift. Im Kapitel <a href="#">2.4 „Parameter MB_MODE und MB_DATA_ADDR“</a> finden Sie eine ausführliche Beschreibung des Parameters MB_DATA_ADDR.
MB_DATA_LEN	UINT	Datenlänge: Anzahl der Bits oder Wörter für den Datenzugriff.
MB_DATA_PTR	VARIANT	Zeiger auf das Modbus-Datenregister. Das Register ist ein Puffer für die vom Modbus TCP-Server empfangenen oder zum Modbus TCP-Server zu sendenden Daten. Der Zeiger muss auf einen globalen Datenbaustein mit Standardzugriff verweisen. In diesem Beispiel verweist der Zeiger auf den DB1.

### Ausgangsparameter

Die Anweisung „MB\_CLIENT“ hat folgende Ausgangsparameter.

Tabelle 2-5

Ausgangsparameter	Datentyp	Beschreibung
DONE	Bool	Das Bit am Ausgangsparameter DONE wird auf „1“ gesetzt, sobald der letzte Auftrag ohne Fehler ausgeführt wurde
BUSY	Bool	0: kein Auftrag von „MB_CLIENT“ in Bearbeitung 1: Auftrag von „MB_CLIENT“ in Bearbeitung
ERROR	Bool	0: kein Fehler 1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter STATUS angezeigt.
STATUS	WORD	Fehlercode der Anweisung

## 2.4 Parameter MB\_MODE und MB\_DATA\_ADDR

Die Anweisung „MB\_CLIENT“ verwendet anstelle eines Funktionscodes den Parameter MB\_MODE. Über den Parameter MB\_DATA\_ADDR legen Sie die Modbus-Startadresse der Daten fest, auf die Sie zugreifen möchten. Die Kombination der Parameter MB\_MODE und MB\_DATA\_ADDR legt den Funktionscode fest, der in der aktuellen Modbus-nachricht verwendet wird.

Die [Tabelle 2-6](#) zeigt den Zusammenhang zwischen dem Parameter MB\_MODE, der in diesem Beispiel verwendeten Modbus-Funktion und dem Adressraum.

Tabelle 2-6

MB_MODE	Modbus-Funktion	Datenlänge	Funktion und Datentyp	MB_DATA_ADDR
0	03	1 bis 125	Halteregister lesen 1 bis 125 WORD pro Aufruf	40001 bis 49999
1	16	2 bis 123	Mehrere Halteregister schreiben 2 bis 123 WORD pro Aufruf	40001 bis 49999

### 3 Anwenderprogramm Modbus TCP-Server

Im Anwenderprogramm des Modbus TCP-Server wird die Anweisung „MB\_SERVER“ für jede Modbus-TCP-Verbindung mit einer eindeutigen ID und einem eigenen Instanz-Datenbaustein aufgerufen. Der Aufruf der Anweisung „MB\_SERVER“ erfolgt jeweils in einer eigenen Funktion.

Tabelle 3-1

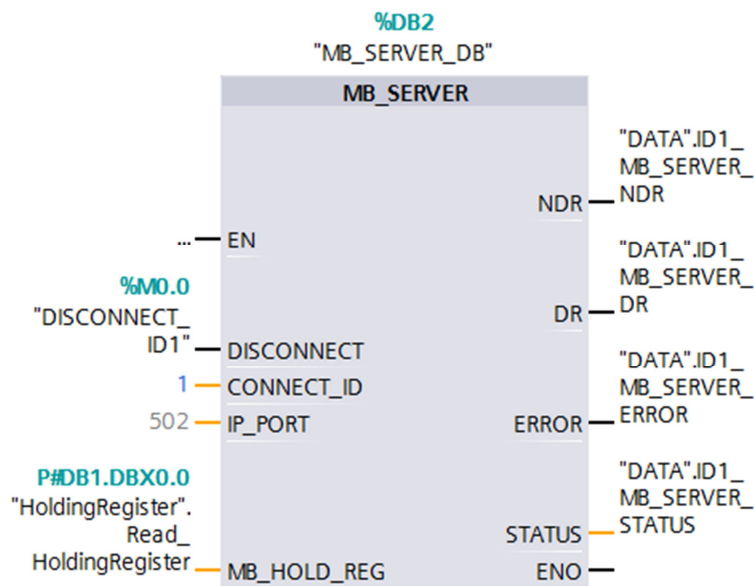
ID	Aufruf der Anweisung „MB_SERVER“	Instanz-DB der Anweisung „MB_SERVER“	Beschreibung
1	FC1 „Read_HoldingRegister“	DB2 „MB_SERVER_DB“	Halteregister lesen
4	FC4 „Write_HoldingRegister“	DB5 „MB_SERVER_DB_3“	Halteregister schreiben

#### 3.1 FC1 „Read\_HoldingRegister“

Die Funktion FC1 „Read\_HoldingRegister“ ruft intern die Anweisung „MB\_SERVER“ auf, um die Verbindungsanfrage zum Lesen des Halteregisters zu verarbeiten. Die Verbindungsanfrage erfolgt über die Modbus-TCP-Verbindung mit der ID=1 und den Port 502.

Die [Abbildung 3-1](#) zeigt den Aufruf und die Parametrierung der Anweisung „MB\_SERVER“ im FC1.

Abbildung 3-1



**Hinweis**

Im Kapitel [3.3 Ein- und Ausgangsparameter der Anweisung „MB\\_SERVER“](#) finden Sie eine Übersicht und Beschreibung der Ein- und Ausgangsparameter der Anweisung „MB\_SERVER“.

### Parameter MB\_HOLD\_REG

Der Parameter MB\_HOLD\_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Speicherbereich können Sie einen globalen Datenbaustein oder einen Merker verwenden. Die folgende Tabelle zeigt, wie die Modbus-Adressen auf das Haltereister für die Modbus-Funktion 3 (WORD lesen) abgebildet werden.

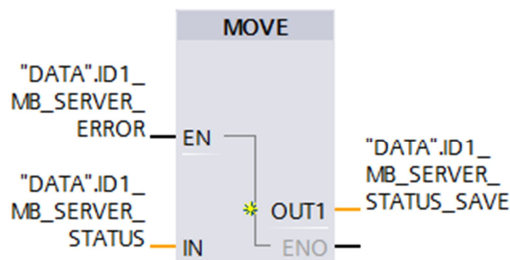
Tabelle 3-2

Modbus-Adresse	Absolute Adresse	Symbolischer Name
40001	DB1.DBW0	„HoldingRegister“.Read_HoldingRegister[0]
40002	DB1.DBW2	„HoldingRegister“.Read_HoldingRegister[1]
40003	DB1.DBW4	„HoldingRegister“.Read_HoldingRegister[2]
40004	DB1.DBW6	„HoldingRegister“.Read_HoldingRegister[3]
40005	DB1.DBW8	„HoldingRegister“.Read_HoldingRegister[4]

### Fehlerauswertung

Wenn die Anweisung „MB\_SERVER“ im FC1 mit Fehler beendet wird, dann wird der Fehlercode des Parameters STATUS für eine Fehlerauswertung im DB7 „DATA“ in der Variablen „ID1\_MB\_SERVER\_STATUS\_SAVE“ gespeichert.

Abbildung 3-2



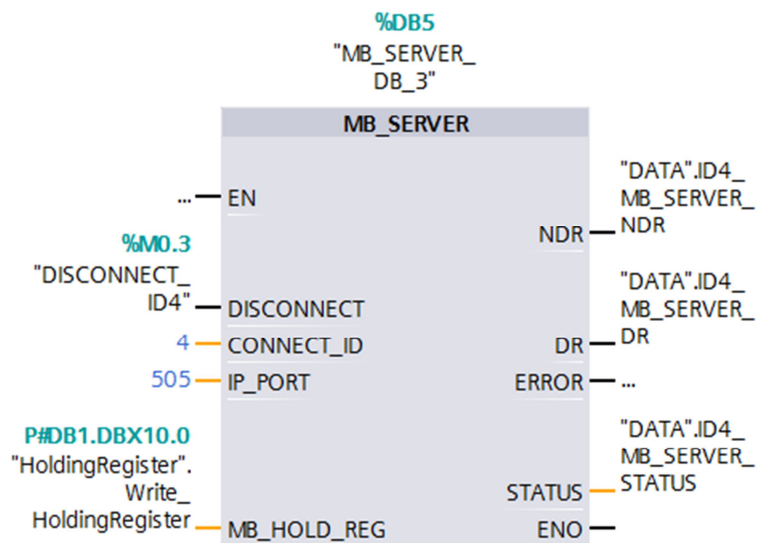
Copyright © Siemens AG 2014 All rights reserved

## 3.2 FC4 „Write\_HoldingRegister“

Die Funktion FC4 „Write\_HoldingRegister“ ruft intern die Anweisung „MB\_SERVER“ auf, um die Verbindungsanfrage zum Schreiben in das Haltereister zu verarbeiten. Die Verbindungsanfrage erfolgt über die Modbus-TCP-Verbindung mit der ID=4 und den Port 505.

Die [Abbildung 3-3](#) zeigt den Aufruf und die Parametrierung der Anweisung „MB\_SERVER“ im FC4.

Abbildung 3-3



**Hinweis**

Im Kapitel [3.3 Ein- und Ausgangsparameter der Anweisung „MB\\_SERVER“](#) finden Sie eine Übersicht und Beschreibung der Ein- und Ausgangsparameter der Anweisung „MB\_SERVER“.

Copyright © Siemens AG 2014 All rights reserved

**Parameter MB\_HOLD\_REG**

Der Parameter MB\_HOLD\_REG ist ein Zeiger auf einen Datenpuffer zur Ablage der Daten, die von oder auf den Modbus-Server gelesen oder geschrieben wurden. Als Speicherbereich können Sie einen globalen Datenbaustein oder einen Merker verwenden. Die folgende Tabelle zeigt, wie die Modbus-Adressen auf das Haltereister für die Modbus-Funktion 16 (mehrere WORD schreiben) abgebildet werden.

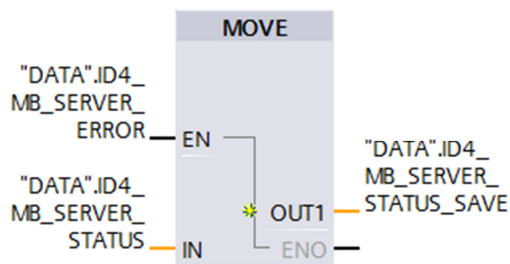
Tabelle 3-3

Modbus-Adresse	Absolute Adresse	Symbolischer Name
40001	DB1.DBW10	„HoldingRegister“.Write_HoldingRegister[0]
40002	DB1.DBW12	„HoldingRegister“.Write_HoldingRegister[1]
40003	DB1.DBW14	„HoldingRegister“.Write_HoldingRegister[2]
40004	DB1.DBW16	„HoldingRegister“.Write_HoldingRegister[3]
40005	DB1.DBW18	„HoldingRegister“.Write_HoldingRegister[4]

**Fehlerauswertung**

Wenn die Anweisung „MB\_SERVER“ im FC4 mit Fehler beendet wird, dann wird der Fehlercode des Parameters STATUS für eine Fehlerauswertung im DB7 „DATA“ in der Variablen „ID4\_MB\_SERVER\_STATUS\_SAVE“ gespeichert.

Abbildung 3-4



### 3.3 Ein- und Ausgangsparameter der Anweisung „MB\_SERVER“

#### Eingangsparameter

Die Anweisung „MB\_SERVER“ hat folgende Eingangsparameter.

Tabelle 3-4

Eingangsparameter	Datentyp	Beschreibung
DISCONNECT	BOOL	Die Anweisung „MB_SERVER“ geht eine passive Verbindung mit einer Partner-Baugruppe ein, d. h. der Server reagiert auf eine TCP-Verbindungsanfrage von jeder anfragenden IP-Adresse. Über diesen Parameter können Sie steuern, wann eine Verbindungsanfrage akzeptiert wird. <ul style="list-style-type: none"> <li>0: Wenn keine Kommunikationsverbindung besteht, wird eine passive Verbindung aufgebaut.</li> <li>1: Initialisierung des Verbindungsabbaus. Ist der Eingang gesetzt, werden keine anderen Vorgänge ausgeführt. Nach erfolgreichem Verbindungsabbau wird am Parameter STATUS der Wert 7003 ausgegeben.</li> </ul>
CONNECT_ID	UINT	Eindeutige ID zur Identifizierung der Verbindung. Jeder Instanz der Anweisung „MB_SERVER“ muss eine eindeutige Verbindungs- ID zugewiesen werden.
IP_PORT	UINT	Startwert 502 Die Nummer des IP-Ports legt fest, welcher IP-Port für Verbindungsanfragen des Modbus TCP-Clients überwacht wird.
MB_HOLD_REG	VARIANT	Zeiger auf das Modbus-Halteregister der Anweisung „MB_SERVER“. Verwenden Sie als Halteregister einen globalen Datenbaustein mit Standard-Zugriff. Das Halteregister beinhaltet die Werte, auf die ein Modbus-Client über die Modbus-Funktionen 3 (lesen), 6 (schreiben) und 16 (schreiben) zugreifen darf.

### Ausgangsparameter

Die Anweisung „MB\_SERVER“ hat folgende Ausgangsparameter.

Tabelle 3-5

Ausgangsparameter	Datentyp	Beschreibung
NDR	BOOL	New Data Read <ul style="list-style-type: none"><li>0: keine neuen Daten</li><li>1: neue Daten durch den Modbus-Client geschrieben</li></ul>
DR	BOOL	Data Read <ul style="list-style-type: none"><li>0: keine neuen Daten gelesen</li><li>1: Daten durch den Modbus-Client gelesen</li></ul>
ERROR		Tritt während eines Aufrufs der Anweisung „MB_SERVER“ ein Fehler auf, wird der Ausgang am Parameter ERROR auf „1“ gesetzt. Detailinformationen zur Fehlerursache werden am Parameter STATUS angezeigt.
STATUS	WORD	Detaillierte Statusinformation der Anweisung.



## 4 Bedienung des Programmbeispiels

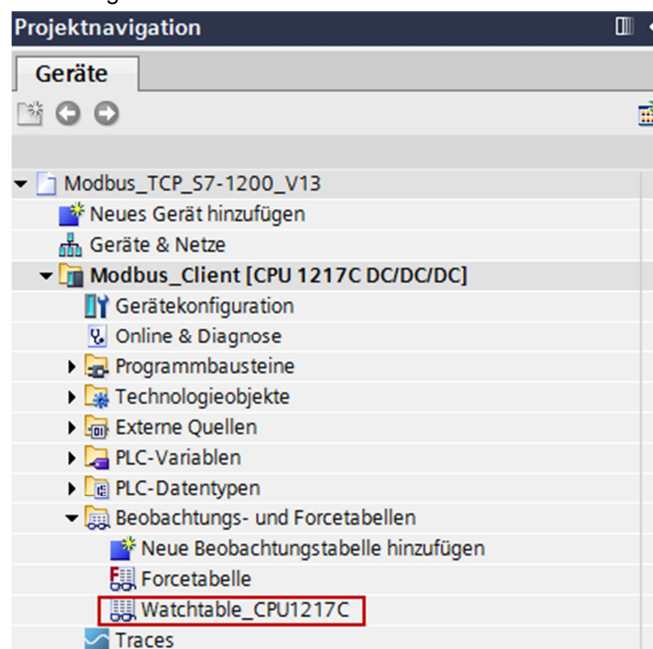
Über die mitgelieferte Beobachtungstabelle der CPU 1217C (Modbus-Client) sowie über die mitgelieferte Beobachtungstabelle der CPU 1211C (Modbus-Server) können Sie kontrollieren, ob die Daten erfolgreich aus dem Halteregeister des Modbus-Server gelesen bzw. in das Halteregeister des Modbus-Server geschrieben wurden.

### 4.1 Bedienung des Anwenderprogramms im Modbus TCP-Client

#### Beobachtungstabelle der CPU 1217C (Modbus-Client) öffnen

In der Projektnavigation navigieren Sie im Geräteordner der CPU 1217C (Modbus-Client) in den Ordner „Beobachtungs- und Forcetabellen“. Doppelklicken Sie auf die bereits angelegte Beobachtungstabelle „Watchtable\_CPU1217C“. Die Beobachtungstabelle „Watchtable\_CPU1217C“ wird im Arbeitsbereich geöffnet.

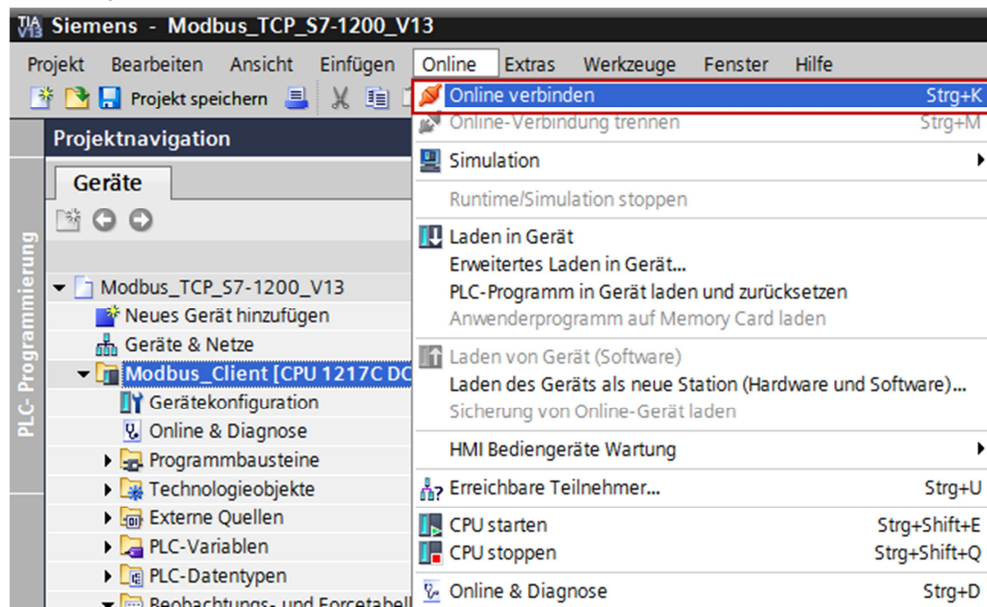
Abbildung 4-1



#### Online-Verbindung zur CPU 1217C aufbauen

In der Projektnavigation markieren Sie den Geräteordner der CPU 1217C (Modbus-Client). Wählen Sie das Menü „Online > Online verbinden“, um eine Online-Verbindung zur CPU 1217C aufzubauen.

Abbildung 4-2



**Werte aus dem Haltereister des Modbus-Servers lesen und in das Haltereister des Modbus-Servers schreiben**

Wenn die Online-Verbindung zur CPU 1217C erfolgreich aufgebaut ist, dann klicken Sie in der Beobachtungstabelle „Watchtable\_CPU1217C“ auf die Schaltfläche „Alle beobachten“. In der Spalte „Beobachtungswert“ werden die aktuellen Werte der Variablen angezeigt.

Abbildung 4-3

Modbus\_TCP\_S7-1200\_V13 > Modbus\_Client [CPU 1217C DC/DC/DC] > Beobachtungs- und Forcettabelle

	Name	Adresse	Anzeigeformat	Beobachtungswert
1	"REQ_ID1"	%M1.0	BOOL	<input checked="" type="checkbox"/> TRUE
2	"RD_HoldingRegister".Read_HoldingRegister[0]	%DB1.DBW0	Hex	16#0001
3	"RD_HoldingRegister".Read_HoldingRegister[1]	%DB1.DBW2	Hex	16#0002
4	"RD_HoldingRegister".Read_HoldingRegister[2]	%DB1.DBW4	Hex	16#0003
5	"RD_HoldingRegister".Read_HoldingRegister[3]	%DB1.DBW6	Hex	16#0004
6	"RD_HoldingRegister".Read_HoldingRegister[4]	%DB1.DBW8	Hex	16#0005
7	"REQ_ID4"	%M1.3	BOOL	<input checked="" type="checkbox"/> TRUE
8	"WR_HoldingRegister".Write_HoldingRegister[0]	%DB8.DBW0	Hex	16#0011
9	"WR_HoldingRegister".Write_HoldingRegister[1]	%DB8.DBW2	Hex	16#0012
10	"WR_HoldingRegister".Write_HoldingRegister[2]	%DB8.DBW4	Hex	16#0013
11	"WR_HoldingRegister".Write_HoldingRegister[3]	%DB8.DBW6	Hex	16#0014
12	"WR_HoldingRegister".Write_HoldingRegister[4]	%DB8.DBW8	Hex	16#0015
13	<Hinzufügen>			

1. Damit der Auftrag zum Lesen des Haltereisters ausgeführt wird, setzen Sie den Merker M1.0 „REQ\_ID1“ auf den Wert „1“. Dazu tragen Sie für den Merker M1.0 „REQ\_ID1“ den Steuerwert „TRUE“ ein.
2. Damit der Auftrag zum Schreiben in das Haltereisters ausgeführt wird, setzen Sie den Merker M1.3 „REQ\_ID4“ auf den Wert „1“. Dazu tragen Sie für den Merker M1.3 „REQ\_ID4“ den Steuerwert „TRUE“ ein.
3. In der CPU 1217C (Modbus-Client) werden die zu schreibenden Werte im DB8 „WR\_HoldingRegister“ gespeichert. In der Spalte „Steuerwert“ tragen Sie die Werte ein, die der Modbus-Client in das Haltereister des Modbus-Servers schreibt.
4. Klicken Sie auf die Schaltfläche „Steuert alle aktivierten Werte „einmalig und sofort““, um die eingetragenen Steuerwerte als aktuelle Werte zu setzen.
5. In der CPU 1217C werden die gelesenen Werte im DB1 „RD\_HoldingRegister“ gespeichert. In der Spalte „Beobachtungswert“ werden die gelesenen Werte angezeigt.

Abbildung 4-4

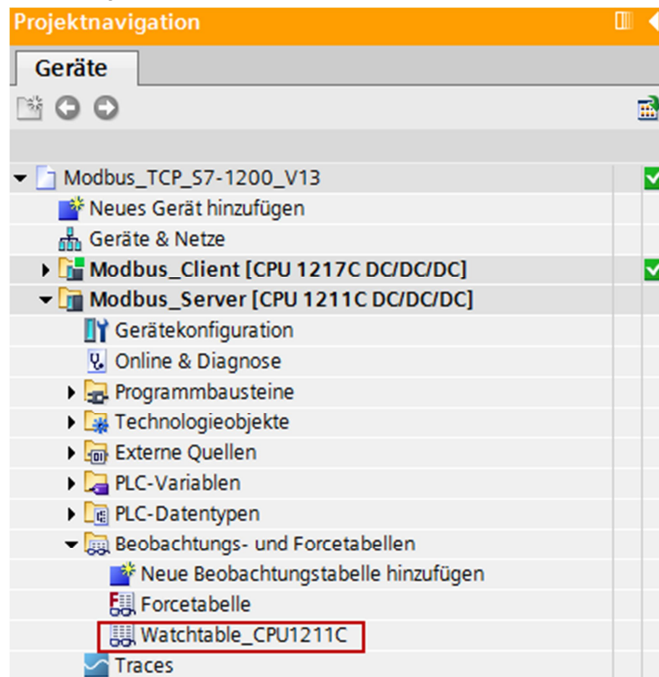
	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert
1	"REQ_ID1"	%M1.0	BOOL	TRUE	TRUE
2	"RD_HoldingRegister".Read_HoldingRegister[0]	%DB1.DBW0	Hex	16#0001	
3	"RD_HoldingRegister".Read_HoldingRegister[1]	%DB1.DBW2	Hex	16#0002	
4	"RD_HoldingRegister".Read_HoldingRegister[2]	%DB1.DBW4	Hex	16#0003	
5	"RD_HoldingRegister".Read_HoldingRegister[3]	%DB1.DBW6	Hex	16#0004	
6	"RD_HoldingRegister".Read_HoldingRegister[4]	%DB1.DBW8	Hex	16#0005	
7	"REQ_ID4"	%M1.3	BOOL	TRUE	TRUE
8	"WR_HoldingRegister".Write_HoldingRegister[0]	%DB8.DBW0	Hex	16#0011	16#0011
9	"WR_HoldingRegister".Write_HoldingRegister[1]	%DB8.DBW2	Hex	16#0012	16#0012
10	"WR_HoldingRegister".Write_HoldingRegister[2]	%DB8.DBW4	Hex	16#0013	16#0013
11	"WR_HoldingRegister".Write_HoldingRegister[3]	%DB8.DBW6	Hex	16#0014	16#0014
12	"WR_HoldingRegister".Write_HoldingRegister[4]	%DB8.DBW8	Hex	16#0015	16#0015
13	<Hinzufügen>				

## 4.2 Bedienung des Anwenderprogramms im Modbus TCP-Server

### Beobachtungstabelle der CPU 1211C (Modbus-Server) öffnen

In der Projektnavigation navigieren Sie im Geräteordner der CPU 1211C (Modbus-Server) in den Ordner „Beobachtungs- und Forcetabellen“. Doppelklicken Sie auf die bereits angelegte Beobachtungstabelle „Watchtable\_CPU1211C“. Die Beobachtungstabelle „Watchtable\_CPU1211C“ wird im Arbeitsbereich geöffnet“.

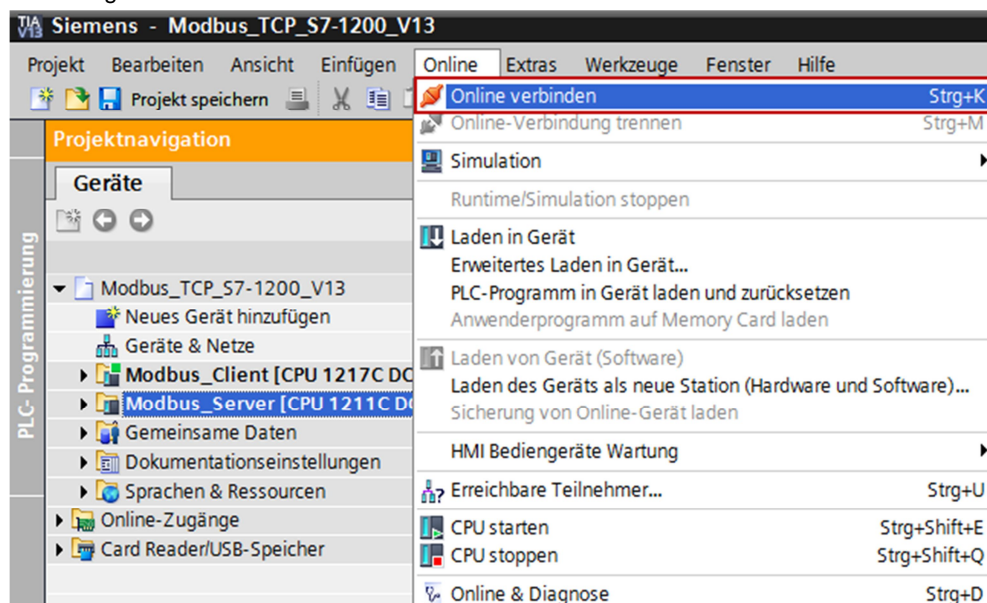
Abbildung 4-5



### Online-Verbindung zur CPU 1211C aufbauen

In der Projektnavigation markieren Sie den Geräteordner der CPU 1211C (Modbus-Server). Wählen Sie das Menü „Online > Online verbinden“, um eine Online-Verbindung zur CPU 1211C aufzubauen.

Abbildung 4-6



Wenn die Online-Verbindung zur CPU 1211C erfolgreich aufgebaut ist, dann klicken Sie in der Beobachtungstabelle „Watchtable\_CPU1211C“ auf die Schaltfläche „Alle beobachten“. In der Spalte „Beobachtungswert“ werden die aktuellen Werte der Variablen angezeigt.

Abbildung 4-7

	Name	Adresse	Anzeigeformat	Beobachtungswert
1	"HoldingRegister".Read_HoldingRegister[0]	%DB1.DBW0	Hex	16#0001
2	"HoldingRegister".Read_HoldingRegister[1]	%DB1.DBW2	Hex	16#0002
3	"HoldingRegister".Read_HoldingRegister[2]	%DB1.DBW4	Hex	16#0003
4	"HoldingRegister".Read_HoldingRegister[3]	%DB1.DBW6	Hex	16#0004
5	"HoldingRegister".Read_HoldingRegister[4]	%DB1.DBW8	Hex	16#0005
6				
7	"HoldingRegister".Write_HoldingRegister[0]	%DB1.DBW10	Hex	16#0011
8	"HoldingRegister".Write_HoldingRegister[1]	%DB1.DBW12	Hex	16#0012
9	"HoldingRegister".Write_HoldingRegister[2]	%DB1.DBW14	Hex	16#0013
10	"HoldingRegister".Write_HoldingRegister[3]	%DB1.DBW16	Hex	16#0014
11	"HoldingRegister".Write_HoldingRegister[4]	%DB1.DBW18	Hex	16#0015
12				

1. In der CPU 1211C (Modbus-Server) werden die zu lesenden Daten im DB1 ab Adresse 0 gespeichert. In der Spalte „Steuerwert“ tragen Sie die Werte ein, die der Modbus-Client aus dem Haltereister des Modbus-Servers liest.
2. Klicken Sie auf die Schaltfläche „Steuert alle aktivierten Werte „einmalig und sofort““, um die eingetragenen Steuerwerte als aktuelle Werte zu setzen.
3. In der CPU 1211C (Modbus-Server) werden die geschriebenen Daten im DB1 ab Adresse 10 gespeichert. In der Spalte „Beobachtungswert“ werden die geschriebenen Werte angezeigt.

Abbildung 4-8

	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert
1	"HoldingRegister".Read_HoldingRegister[0]	%DB1.DBW0	Hex	16#0001	16#0001
2	"HoldingRegister".Read_HoldingRegister[1]	%DB1.DBW2	Hex	16#0002	16#0002
3	"HoldingRegister".Read_HoldingRegister[2]	%DB1.DBW4	Hex	16#0003	16#0003
4	"HoldingRegister".Read_HoldingRegister[3]	%DB1.DBW6	Hex	16#0004	16#0004
5	"HoldingRegister".Read_HoldingRegister[4]	%DB1.DBW8	Hex	16#0005	16#0005
6					
7	"HoldingRegister".Write_HoldingRegister[0]	%DB1.DBW10	Hex	16#0011	
8	"HoldingRegister".Write_HoldingRegister[1]	%DB1.DBW12	Hex	16#0012	
9	"HoldingRegister".Write_HoldingRegister[2]	%DB1.DBW14	Hex	16#0013	
10	"HoldingRegister".Write_HoldingRegister[3]	%DB1.DBW16	Hex	16#0014	
11	"HoldingRegister".Write_HoldingRegister[4]	%DB1.DBW18	Hex	16#0015	