

# Industrie-Automatisierung **System *HIMatrix***

## ***MODBUS*** ***Master/Slave***

### **Handbuch**



HIMA Paul Hildebrandt GmbH  
Industrie-Automatisierung

## **Wichtige Hinweise**

Alle in diesem Handbuch genannten HIMA-Produkte sind mit dem HIMA-Warenzeichen geschützt. Dies gilt ebenfalls, soweit nicht anders vermerkt, auch für andere genannte Hersteller und deren Produkte.

Alle technischen Angaben und Hinweise in diesem Handbuch wurden mit größter Sorgfalt erarbeitet und unter Einschaltung wirksamer Kontrollmaßnahmen zusammengestellt. Trotzdem sind Fehler nicht ganz auszuschließen.

HIMA sieht sich deshalb veranlasst, darauf hinzuweisen, dass weder eine Garantie noch die juristische Verantwortung oder irgend eine Haftung übernommen werden kann für die Folgen, die auf fehlerhafte Angaben zurückgehen. Für die Mitteilung eventueller Fehler ist HIMA dankbar.

Technische Änderungen vorbehalten.

Weitere Informationen sind in der Dokumentation auf der CD-ROM „ELOP II Factory“ und auf unserer Website unter [www.hima.de](http://www.hima.de) zu finden.

Informationsanfragen sind zu richten an:

HIMA Paul Hildebrandt GmbH  
Postfach 1261  
68777 Brühl

Tel: +49(6202)709 0  
Fax: +49(6202)709 107

e-mail: [info@hima.com](mailto:info@hima.com)

Inhaltsverzeichnis	Seite
<b>1 HIMA Modbus Master .....</b>	<b>7</b>
1.1 Benötigte Ausstattung und Systemanforderungen .....	7
1.2 Modbus Master Eigenschaften .....	7
1.3 Modbus-Funktionen .....	9
1.4 Überblick über die Anforderungstelegramme .....	11
1.5 Anforderungstelegramme zum Lesen .....	11
1.5.1 Signale verbinden .....	12
1.5.2 Validieren .....	12
1.5.3 Kopieren .....	12
1.5.4 Löschen .....	12
1.5.5 Eigenschaften .....	12
1.5.5.1 Anforderungstelegramm „Read Coils (01)“ .....	13
1.5.5.2 Anforderungstelegramm „Read Discrete Inputs (02)“ .....	13
1.5.5.3 Anforderungstelegramm „Read Holding Register (03)“ .....	13
1.5.5.4 Anforderungstelegramm „Read Input Registers (04)“ .....	14
1.6 Anforderungstelegramm zum Lesen und Schreiben .....	15
1.6.1 Signale verbinden .....	16
1.6.2 Validieren .....	16
1.6.3 Kopieren .....	16
1.6.4 Löschen .....	17
1.6.5 Eigenschaften .....	17
1.6.5.1 Anforderungstelegramm „Read Write Holding Registers (23)“ .....	17
1.7 Anforderungstelegramme zum Schreiben .....	18
1.7.1 Signale verbinden .....	18
1.7.2 Validieren .....	19
1.7.3 Kopieren .....	19
1.7.4 Löschen .....	19
1.7.5 Eigenschaften .....	19
1.7.5.1 Anforderungstelegramm „Write Multiple Coils (15)“ .....	19
1.7.5.2 Anforderungstelegramm „Write Multiple Registers (16)“ .....	20
1.7.5.3 Anforderungstelegramm „Write Single Coil (05)“ .....	20
1.7.5.4 Anforderungstelegramm „Write Single Register (06)“ .....	20
1.8 HIMA Modbus Master .....	21
1.8.1 Signale verbinden .....	21
1.8.2 Validieren .....	22
1.8.3 Neu .....	22
1.8.4 Import .....	22
1.8.5 Export .....	22
1.8.6 Kopieren .....	22
1.8.7 Löschen .....	22
1.8.8 Eigenschaften .....	23

1.8.8.1	Register „Allgemein“ .....	23
1.8.8.2	Register „CPU/COM“ .....	24
<b>1.9</b>	<b>Ethernet Slaves .....</b>	<b>25</b>
1.9.1	Neu .....	25
1.9.2	Einfügen .....	25
1.9.3	Eigenschaften .....	26
1.9.4	TCP/UDP Slave .....	26
1.9.4.1	Signale verbinden .....	26
1.9.4.2	Validieren .....	27
1.9.4.3	Neu .....	27
1.9.4.4	Import .....	27
1.9.4.5	Export .....	27
1.9.4.6	Eigenschaften .....	28
<b>1.10</b>	<b>Modbus Gateway .....</b>	<b>30</b>
1.10.1	Validieren .....	31
1.10.2	Neu .....	31
1.10.3	Eigenschaften .....	32
1.10.4	Gateway Slave .....	33
1.10.4.1	Signale verbinden .....	33
1.10.4.2	Validieren .....	34
1.10.4.3	Neu .....	34
1.10.4.4	Import .....	34
1.10.4.5	Export .....	34
1.10.4.6	Eigenschaften .....	34
<b>1.11</b>	<b>Serieller Modbus .....</b>	<b>36</b>
1.11.1	Validieren .....	37
1.11.2	Neu .....	37
1.11.3	Eigenschaften .....	37
1.11.4	Modbus Slave .....	39
1.11.4.1	Signale verbinden .....	39
1.11.4.2	Validieren .....	40
1.11.4.3	Neu .....	40
1.11.4.4	Import .....	40
1.11.4.5	Export .....	40
1.11.4.6	Eigenschaften .....	40
<b>2</b>	<b>HIMA Modbus Slave .....</b>	<b>42</b>
<b>2.1</b>	<b>Benötigte Ausstattung und Systemanforderungen .....</b>	<b>42</b>
<b>2.2</b>	<b>Modbus Slave Eigenschaften .....</b>	<b>42</b>
<b>2.3</b>	<b>Modbus Funktionen .....</b>	<b>43</b>
2.3.1	Signale verbinden .....	45
<b>2.4</b>	<b>Beispiel: Schreiben auf den Import-Bereich des <i>HIMatrix</i> Modbus Slaves .....</b>	<b>46</b>
2.4.1	Der Import-Bereich des <i>HIMatrix</i> Modbus Slaves .....	46

2.4.2	Konfiguration der Anforderungstelegramme im Modbus Master .....	47
2.4.2.1	Anforderungstelegramm „Write Multiple Coils (15)“ .....	47
2.4.2.2	Erstes Anforderungstelegramm „Write Multiple Register (16)“ .....	48
2.4.2.3	Zweites Anforderungstelegramm „Write Multiple Register (16)“ .....	49
2.4.3	Validation .....	50
2.4.4	Import .....	50
2.4.5	Export .....	50
2.4.6	Kopieren .....	50
2.4.7	Löschen .....	50
2.4.8	Eigenschaften .....	50
2.4.9	Register „Allgemein“ .....	51
2.4.9.1	Register „Serielle Schnittstelle“ .....	52
2.4.9.2	Register „TCP und UDP Ports“ .....	54
2.4.9.3	Register „CPU/COM“ .....	55
<b>3</b>	<b>Diagnose und Fehlercodes .....</b>	<b>56</b>
<b>3.1</b>	<b>Control Panel (Register „ModbusMs.“) .....</b>	<b>56</b>
3.1.1	Rahmen „Modbus Master“ .....	56
3.1.2	Rahmen „Modbus Slave“ .....	57
3.1.3	Diagnose Feldbus (FBx) LEDs der seriellen Schnittstellen.....	58
3.1.3.1	Funktion der FBx LED beim Modbus Master .....	58
3.1.3.2	Funktion der FBx LED beim Modbus Slave .....	59
3.1.4	Fehlercodes der TCP/IP Verbindung .....	59
<b>4</b>	<b>Anwendung mit Modbus Gateway.....</b>	<b>60</b>
<b>4.1</b>	<b>Eintrag der Modbus Master in die Routingtabelle .....</b>	<b>60</b>
<b>4.2</b>	<b>Konfiguration des Modbus Master 01 .....</b>	<b>62</b>
<b>4.3</b>	<b>Konfiguration des Modbus Master 02.....</b>	<b>67</b>
<b>4.4</b>	<b>Konfiguration des Modbus Slave 01 .....</b>	<b>70</b>

## Zu diesem Handbuch

Ziel dieses Handbuchs ist es, den Anwender mit den Menüfunktionen und den Dialogen des HIMA Modbus Master in **ELOP II Factory** vertraut zu machen.

Um den HIMA Modbus Master einzurichten, benötigt der Anwender das Programmierwerkzeug **ELOP II Factory**, das auf einem PC mit dem Betriebssystem Microsoft Windows NT®, Windows 2000® oder Windows XP® installiert sein muss.

Der Anwender sollte den Umgang mit dem Programmierwerkzeug **ELOP II Factory** und den HIMA HIMatrix Steuerungen beherrschen. Für ein Selbststudium wird das Handbuch „Erste Schritte **ELOP II Factory**“ und die Online-Hilfe von **ELOP II Factory** empfohlen. Darüber hinaus bietet HIMA Kundens Schulungen an.

Dieses Handbuch ist in vier Teile gegliedert:

- Der erste Teil erklärt die Menüfunktionen und Dialogfenster in **ELOP II Factory** zur Konfiguration des HIMA Modbus Master.
- Der zweite Teil erklärt die Menüfunktionen und Dialogfenster in **ELOP II Factory** zur Konfiguration des HIMA Modbus Slave.
- Der dritte Teil gibt Hinweise zur Diagnose und Fehlersuche.
- Im vierten Teil wird eine Konfiguration des HIMA Modbus Master beschrieben, die der Anwender in einer Schrittanleitung nachvollziehen kann.

Für weitere Informationen zu Modbus verweist HIMA auf die folgenden Spezifikationen:

- Modbus Application Protocol Specification
- Modbus over serial line specification and implementation guide V1.0

(Siehe [www.modbus.org](http://www.modbus.org))

Wir wünschen Ihnen viel Erfolg bei der Umsetzung Ihrer Modbus-Projekte. Sollten Sie weitere Fragen haben, wenden Sie sich bitte direkt an HIMA.

**Alle Rechte und technische Änderungen vorbehalten.**

© HIMA Paul Hildebrandt GmbH  
Postfach 1261  
D - 68777 Brühl bei Mannheim

# 1 HIMA Modbus Master

Die Datenübertragung zwischen dem HIMA Modbus Master und den Modbus Slaves kann sowohl über die serielle Schnittstelle (RS 485) als auch über TCP/UDP (Ethernet) erfolgen. Zudem kann der HIMA Modbus Master auch als Gateway dienen (siehe Kapitel 1.10).

## 1.1 Benötigte Ausstattung und Systemanforderungen

HIMA <b>ELOP II Factory</b>	Ab Version 5.2.0
HIMatrix Steuerungen	F30, F35 und F60 ab Hardware Revision: 02
Betriebssystemversionen der HIMatrix Steuerungen	<ul style="list-style-type: none"><li>- COM BS ab Version 6.22</li><li>- CPU BS ab Version 4.50</li></ul>
HIMA Modbus Modul	Siehe Datenblatt der jeweiligen HIMatrix Steuerung Kapitel „Anschlüsse für Feldbus Kommunikation.“
Lizenznummer	Jede der folgenden Modbus Master Funktionen muss mit einer Lizenznummer frei geschaltet werden. <ul style="list-style-type: none"><li>- Modbus Master (RS 485 Kommunikation)</li><li>- Modbus Master TCP (Ethernetkommunikation)</li></ul>

## 1.2 Modbus Master Eigenschaften

Modbus Master	Es kann nur ein Modbus Master pro Ressource konfiguriert werden. Ein Modbus Master kann gleichzeitig: <ul style="list-style-type: none"><li>- TCP/UDP Slaves,</li><li>- serielle Slaves auf mehreren seriellen Bussen bedienen und</li><li>- kann als Gateway für einen zweiten Modbus Master dienen.</li></ul>
Max. Anzahl Modbus Slaves	Ein Modbus Master kann bis zu 247 Slaves bedienen. <ul style="list-style-type: none"><li>- 122 Modbus Slaves pro serielle Schnittstelle <sup>1</sup></li><li>- 32 TCP/UDP Slaves über TCP/IP Verbindung</li><li>- 32 TCP/UDP Slaves über UDP/IP Verbindung</li></ul>
Max. Anzahl Anforderungstelegramme	Es können bis zu 988 Anforderungstelegramme pro Modbus Master konfiguriert werden.
Max. Länge Anforderungstelegramm	Siehe Spezifikation “Modbus Application Protocol Specification” <a href="http://www.modbus.org">www.modbus.org</a>
Max. Größe der	8192 Bytes (bis COM-BS V8.32)

Sendedaten 16384 Bytes (ab COM-BS V8.32)

Die tatsächliche Größe der Anwenderdaten die gesendet werden kann, muss wie folgt berechnet werden:  
Das Statusbyte des Masters und das Statusbyte von jedem zugeordneten Slave, muss von der max. Größe der Sendedaten (8192 Bytes bzw. 16384 Bytes) subtrahiert werden.

Max. Größe der Anwenderdaten (senden)  
= Max. Größe der Sendedaten in Byte - 1 Byte –  
(1 Byte \* Anzahl der Slaves)

Max. Größe der Empfangsdaten 8192 Bytes (bis COM-BS V8.32)  
16384 Bytes (ab COM-BS V8.32)

Die tatsächliche Größe der Anwenderdaten die empfangen werden kann, muss wie folgt berechnet werden:  
Die zwei Statusbytes des Masters und das Statusbyte von jedem zugeordneten Slave, muss von der max. Größe der Empfangsdaten (8192 Bytes bzw. 16384 Bytes) subtrahiert werden.

Max. Größe der Anwenderdaten (empfangen)  
= Max. Größe der Empfangsdaten in Byte - 2 Byte –  
(1 Byte \* Anzahl der Slaves)

Darstellungsformat der Modbus Daten Die *HIMatrix* Steuerungen verwenden das Big Endian Format für die Daten.  
Beispiel 32 Bit Daten (z.B. DWORD, DINT):

32 Bit Daten (hex)	0x12345678			
Speicher-Offset	0	1	2	3
Big Endian ( <i>HIMatrix</i> )	12	34	56	78
Middle Endian (H51q)	56	78	12	34
Little Endian	78	56	34	12

<sup>1</sup> Nach der Norm sind insgesamt drei Repeater zulässig, so dass maximal 122 Bus Teilnehmer pro serielle Schnittstelle eines Masters möglich sind. Der Modbus Master kann maximal 32 TCP/IP- und 32 UDP/IP-Slaves gleichzeitig bedienen.

**Hinweis:** Neben dem Modbus Protokoll können gleichzeitig noch weitere Protokolle (z.B. Profibus-DP, TCP S/R ...) auf der *HIMatrix*-Steuerung betrieben werden.  
Insgesamt können pro *HIMatrix*-Steuerung 16384 Bytes Daten gesendet und 16384 Bytes Daten empfangen werden.  
Die Sende- und Empfangsdaten können beliebig zwischen den Protokollen aufgeteilt werden. Die maximale Größe der Sende- und Empfangsdaten pro Protokoll ist vom COM-BS abhängig (siehe oben).



### 1.3 Modbus-Funktionen

Mit den bei HIMA realisierten Modbus-Funktionen ist es möglich, Signale in beide Richtungen zu schreiben oder zu lesen. Es können einzelne Signale (BOOL/WORD) oder mehrere aufeinander folgende Signale (BOOL/WORD) gelesen oder geschrieben werden.

Mit den Read-Funktionen können Signale aus dem Import- oder Export-Bereich eines Slave gelesen werden.

Mit den Write-Funktionen werden Signale nur in den Importbereich eines Slave geschrieben.

(Weitere Informationen zu Modbus finden Sie in der Spezifikation "Modbus Application Protocol Specification" [www.modbus.org](http://www.modbus.org))

Element	Code	Typ	Bedeutung
READ COILS	01	BOOL	Lesen mehrerer Signale (BOOL) aus dem Import- oder Export <sup>1)</sup> -Bereich des Slave.
READ DISCRETE INPUTS	02	BOOL	Lesen mehrerer Signale (BOOL) aus dem Export-Bereich des Slave.
READ HOLDING REGISTERS	03	WORD	Lesen mehrerer Signale beliebigen Typs aus dem Import- oder Export <sup>1)</sup> -Bereich des Slave.
READ INPUT REGISTERS	04	WORD	Lesen mehrerer Signale beliebigen Typs aus dem Export-Bereich des Slave.
READ WRITE HOLDING REGISTERS	23	WORD	Schreiben und Lesen mehrerer Signale beliebigen Typs in und aus dem Import- oder Export Bereich des Slave.
WRITE MULTIPLE COILS	15	BOOL	Schreiben mehrerer Signale (BOOL) in den Import-Bereich des Slave.
WRITE MULTIPLE REGISTERS	16	WORD	Schreiben mehrerer Signale beliebigen Typs in den Import-Bereich des Slave.
WRITE SINGLE COIL	05	BOOL	Schreiben eines einzelnen Signals (BOOL) in den Import-Bereich des Slave.
WRITE SINGLE REGISTER	06	WORD	Schreiben eines einzelnen Signals (WORD) in den Import-Bereich des Slave.

**Tabelle 1: Beschreibung der Modbus-Funktionen**

<sup>1)</sup>Export Bereich kann nur bei HIMA Slaves gewählt werden

---

**Hinweis:** Für die Funktionscodes (1 und 3) muss der Anwender den Import-/Export-Bereich im HIMA Modbus Slave konfigurieren, aus dem der Master die Daten lesen soll (Siehe Beschreibung des Slave).

Für die *HIMatrix*-Steuerungen mit COM BS Versionen V.3 bis V.6 gelten die folgenden Lesebereiche für Funktionscode 1 und 3:

COM BS V.3:	Importbereich
COM BS V.4:	Exportbereich (kompatibel zu H51q)
COM BS V.6:	Konfigurierbar

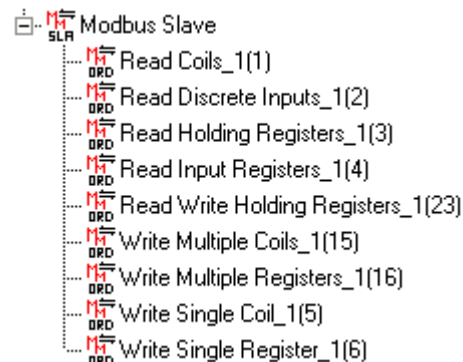
---

## 1.4 Überblick über die Anforderungstelegramme

Ein Anforderungstelegramm des Modbus Master enthält neben der Modbus-Funktion die Startadresse des Lese-/Schreibbereichs.

**Hinweis** Vor ELOP II Factory Version 6.42 muss die Anzahl der Signale in den „Eigenschaften“ der Anforderungstelegramme eingetragen werden.

Es können bis zu 988 Anforderungstelegramme pro Modbus Master konfiguriert werden.



**Bild 1: Die Anforderungstelegramme im HIMA Modbus Master**

## 1.5 Anforderungstelegramme zum Lesen

Zum Lesen von Signalen sendet der Modbus Master ein „Anforderungstelegramm zum Lesen“ an den Modbus Slave.

Der Modbus Slave sendet daraufhin ein Antworttelegramm mit den angeforderten Signalen an den Modbus Master zurück.

Die folgenden „Anforderungstelegramme zum Lesen“ stehen zur Verfügung:

- Read Coils (01)
- Read Discrete Inputs (02)
- Read Holding Registers (03)
- Read Input Registers (04)

Das Kontextmenü eines „Anforderungstelegramms zum Lesen“ enthält die folgenden Funktionen.

Anforderungstelegramm zum Lesen
Signale verbinden
Validieren
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.5.1 Signale verbinden

Im Dialog „Signale Zuordnungen“ werden die Signale eingefügt, die der Modbus Master vom Modbus Slave anfordert.

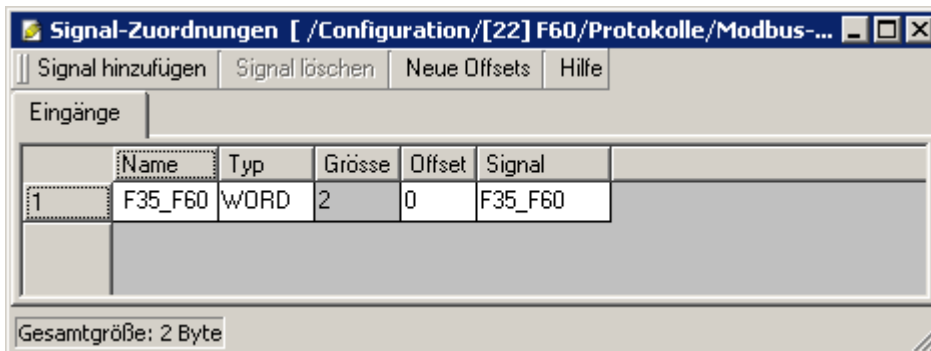


Bild 2: Dialogfenster „Signal Zuordnungen“ der Anforderungstelegramme zum Lesen

### 1.5.2 Validieren

Vor der Codegenerierung kann die Parametrierung des „Anforderungstelegramm zum Lesen“ getestet werden.

In der Strukturansicht wird das „Anforderungstelegramm zum Lesen“ selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegenerierung durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegenerierung abgebrochen.

### 1.5.3 Kopieren

Kopiert das „Anforderungstelegramm zum Lesen“ inklusive der Signale in die Zwischenablage.

### 1.5.4 Löschen

Löscht das gewählte „Anforderungstelegramm zum Lesen“ aus dem Projekt.

---

**Achtung** Mit der Menüfunktion „Löschen“ wird das „Anforderungstelegramm zum Lesen“ gelöscht. Vergewissern Sie Sich, dass Sie das „Anforderungstelegramm zum Lesen“ wirklich löschen wollen.

---

### 1.5.5 Eigenschaften

Eigenschaften der Anforderungstelegramme zum Lesen (siehe 1.5.5.1 bis 1.5.5.4).

### 1.5.5.1 Anforderungstelegramm „Read Coils (01)“

Element	Bedeutung
Typ	Modbus-Funktion „Read Coils (01)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Lesebereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8191 (bis COM-BS V8.32) 0 bis 16383 (ab COM-BS V8.32)

Tabelle 2: Anforderungstelegramm „Read Coils (01)“

### 1.5.5.2 Anforderungstelegramm „Read Discrete Inputs (02)“

Element	Bedeutung
Typ	Modbus-Funktion „Read Discrete Inputs (02)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Lesebereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8191 (bis COM-BS V8.32) 0 bis 16383 (ab COM-BS V8.32)

Tabelle 3: Anforderungstelegramm „Read Discrete Inputs (02)“

### 1.5.5.3 Anforderungstelegramm „Read Holding Register (03)“

Element	Bedeutung
Typ	Modbus-Funktion „Read Holding Registers (03)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Lesebereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)

Tabelle 4: Anforderungstelegramm „Read Holding Registers (03)“

#### 1.5.5.4 Anforderungstelegramm „Read Input Registers (04)“

Element	Bedeutung
Typ	Modbus-Funktion „Read Input Registers (04)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Lesebereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)

Tabelle 5: Anforderungstelegramm „Read Input Registers (04)“

## 1.6 Anforderungstelegramm zum Lesen und Schreiben

Zum Lesen und Schreiben von Signalen sendet der Modbus Master ein „Anforderungstelegramm zum Lesen und Schreiben“ an den Modbus Slave.

Zuerst schreibt der Modbus Master die definierten Schreib-Signale in den definierten Import-Bereich des Modbus Slave.

Anschließend liest der Modbus Master die definierten Lese-Signale aus dem definierten Import-Bereich des Modbus Slave.

---

**Hinweis:** Die Funktionen Schreiben und Lesen sind auch bei dem „Anforderungstelegramme zum Lesen und Schreiben“ voneinander unabhängig, sie werden nur in einem gemeinsamen Anforderungstelegramm gesendet.

Eine häufige Anwendung für das „Anforderungstelegramme zum Lesen und Schreiben“ ist jedoch, dass die geschriebenen Signale des Modbus Master wieder zurückgelesen werden. Damit wird überprüft, ob die gesendeten Signale korrekt geschrieben wurden.

---

Es steht das folgenden „Anforderungstelegramme zum Lesen und Schreiben“ zur Verfügung:

Read Write Holding Register (23)

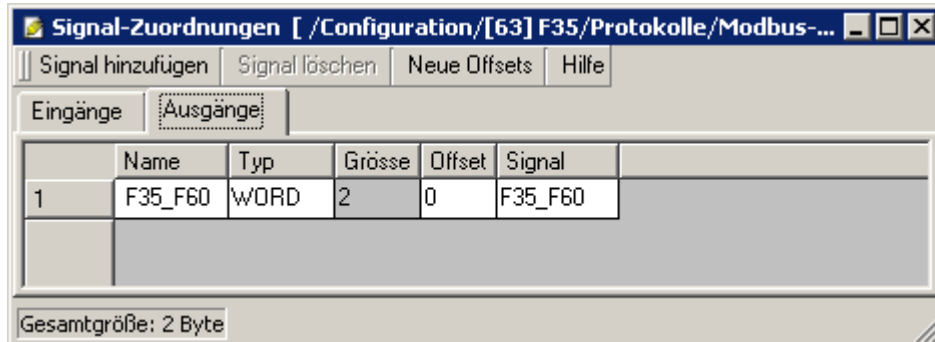
Das Kontextmenü des „Anforderungstelegramme zum Lesen und Schreiben“ enthält die folgenden Funktionen.

Anforderungstelegramm zum Lesen und Schreiben
Signale verbinden
Validieren
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.6.1 Signale verbinden

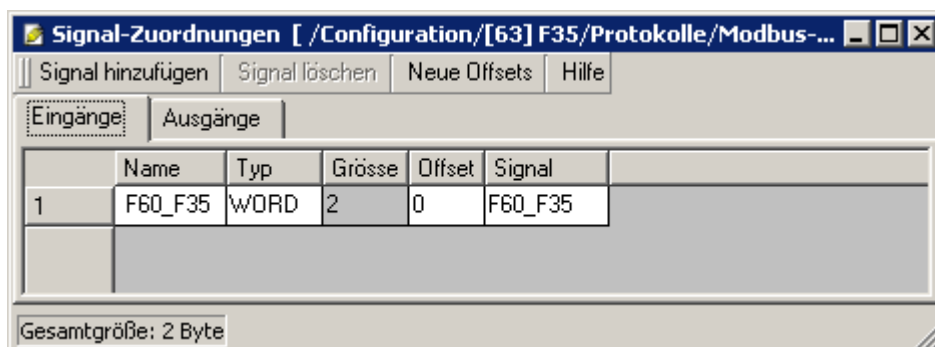
Im Dialog „Signale Zuordnungen“ werden:

- Die Signale die der Modbus Master an den Modbus Slave sendet, werden im Register „Ausgänge“ eingefügt.



**Bild 3: Register „Ausgänge“ des Dialog „Signal Zuordnungen“ der Read Write Funktion (23)**

- Die Signale die der Modbus Master vom Modbus Slave empfängt, werden im Register „Eingänge“ eingefügt.



**Bild 4: Register „Eingänge“ des Dialog „Signal Zuordnungen“ der Read Write Funktion (23)**

### 1.6.2 Validieren

Vor der Codegenerierung kann die Parametrierung des „Anforderungstelegramm zum Lesen und Schreiben“ getestet werden.

In der Strukturansicht wird das „Anforderungstelegramm zum Lesen und Schreiben“ selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 1.6.3 Kopieren

Kopiert das „Anforderungstelegramm zum Lesen und Schreiben“ inklusive Konfiguration in die Zwischenablage.



### 1.6.4 Löschen

Löscht das gewählte „Anforderungstelegramm zum Lesen und Schreiben“ aus dem Projekt.

---

**Achtung** Mit der Menüfunktion „Löschen“ wird das „Anforderungstelegramm zum Lesen und Schreiben“ gelöscht. Vergewissern Sie Sich, dass Sie das „Anforderungstelegramm zum Lesen und Schreiben“ wirklich löschen wollen.

---

### 1.6.5 Eigenschaften

Eigenschaften des Anforderungstelegramm.

#### 1.6.5.1 Anforderungstelegramm „Read Write Holding Registers (23)“

Element	Bedeutung
Typ	Modbus-Funktion „Read Write Holding Registers (23)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Lesebereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)
Startadresse des Schreibbereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)

Tabelle 6: Anforderungstelegramm „Read Write Holding Registers (23)“

## 1.7 Anforderungstelegramme zum Schreiben

Zum Schreiben von Signalen sendet der Modbus Master ein „Anforderungstelegramm zum Schreiben“ an den Modbus Slave.

Der Modbus Slave schreibt die empfangenen Signale in seinen Import-Bereich.

Die folgenden „Anforderungstelegramme zum Schreiben“ stehen zur Verfügung:

- Write Multiple Coils (15)
- Write Multiple Registers (16)
- Write Single Coil (05)
- Write Single Register (06)

Das Kontextmenü eines „Anforderungstelegramms zum Schreiben“ enthält die folgenden Funktionen.

Anforderungstelegramm zum Schreiben
Signale verbinden
Validieren
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.7.1 Signale verbinden

Im Dialog „Signale verbinden“ eines „Anforderungstelegramms zum Schreiben“ müssen die Signale eingefügt werden, die der Modbus Master zum Modbus Slave schreibt.

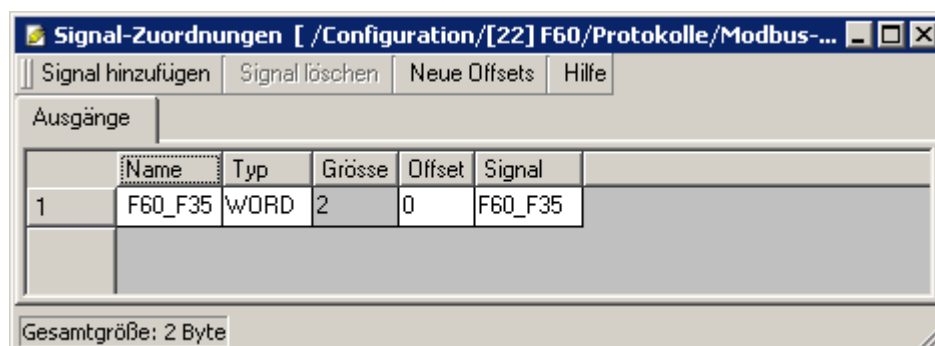


Bild 5: Dialogfenster „Signal Zuordnungen“ der Write Anforderungstelegramme

### 1.7.2 Validieren

Vor der Codegenerierung kann die Parametrierung des „Anforderungstelegramm zum Schreiben“ getestet werden.

In der Strukturansicht wird das „Anforderungstelegramm zum Schreiben“ selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 1.7.3 Kopieren

Kopiert das „Anforderungstelegramm zum Schreiben“ inklusive Konfiguration in die Zwischenablage.

### 1.7.4 Löschen

Löscht das gewählte „Anforderungstelegramm zum Schreiben“ aus dem Projekt.

---

**Achtung** Mit der Menüfunktion „Löschen“ wird das „Anforderungstelegramm zum Schreiben“ gelöscht. Vergewissern Sie Sich, dass Sie das „Anforderungstelegramm zum Schreiben“ wirklich löschen wollen.

---

### 1.7.5 Eigenschaften

Eigenschaften der „Anforderungstelegramme zum Schreiben“ (siehe 1.7.5.1 bis 1.7.5.4).

#### 1.7.5.1 Anforderungstelegramm „Write Multiple Coils (15)“

Element	Bedeutung
Typ	Modbus-Funktion „Write Multiple Coils (15)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Schreibbereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8191 (bis COM-BS V8.32) 0 bis 16383 (ab COM-BS V8.32)

Tabelle 7: Anforderungstelegramm „Write Multiple Coils (15)“

**1.7.5.2 Anforderungstelegramm „Write Multiple Registers (16)“**

Element	Bedeutung
Typ	Modbus-Funktion „Write Multiple Registers (16)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Schreibbereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)

Tabelle 8: Anforderungstelegramm „Write Multiple Registers (16)“

**1.7.5.3 Anforderungstelegramm “Write Single Coil (05)”**

Element	Bedeutung
Typ	Modbus-Funktion „Write Single Coil (05)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Schreibbereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8191 (bis COM-BS V8.32) 0 bis 16383 (ab COM-BS V8.32)

Tabelle 9: Anforderungstelegramm „Write Single Coil (05)“

**1.7.5.4 Anforderungstelegramm “Write Single Register (06)“**

Element	Bedeutung
Typ	Modbus-Funktion „Write Single Register (06)“
Name	Beliebiger, eindeutiger Name, für die Modbus-Funktion
Beschreibung	Beschreibung für die Modbus-Funktion
Startadresse des Schreibbereichs	0 bis 65535 Für <i>HIMatrix</i> Modbus Slaves 0 bis 8190 (bis COM-BS V8.32) 0 bis 16382 (ab COM-BS V8.32)

Tabelle 10: Anforderungstelegramm „Write Single Register (06)“

## 1.8 HIMA Modbus Master

Das Kontextmenü des Modbus Master enthält die folgenden Funktionen.

Modbus Master
Signale verbinden
Validieren
Neu
Import
Export
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.8.1 Signale verbinden

Das Dialogfenster „Signal-Zuordnungen“ stellt die drei Statussignale

- Bus-Fehler und
- Master-Status und
- Master-Zustandssteuerung

bereit, die es erlauben, den Zustand des Modbus Master im Anwenderprogramm auszuwerten und den Modbus Master zu steuern.

Eingänge	Bedeutung
Bus Fehler	Nicht genutzt
Master Status	Der Master Status zeigt den momentanen Protokollzustand an: bis COM BS-Version 8.0 0: OPERATE 1: OFFLINE ab COM BS-Version 8.0 1: OPERATE 0: OFFLINE

Tabelle 11: Dialogfenster „Signale verbinden“ von Modbus Master

Ausgänge	Bedeutung
Master-Zustandssteuerung	Hiermit kann der Modbus Master vom Anwenderprogramm gestoppt oder gestartet werden. 0: START 1: STOP

Tabelle 12: Dialogfenster „Signale verbinden“ von Modbus Master

### 1.8.2 Validieren

Vor der Codegenerierung kann die Parametrierung des Masters und der Slaves getestet werden. In der Strukturansicht wird der Modbus Master selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 1.8.3 Neu

Mit der Menüfunktion „Neu“ kann dem Modbus Master ein neuer

- „Modbus Gateway“ oder ein
  - „Serieller Modbus“
- hinzugefügt werden.

### 1.8.4 Import

Unter der Menüfunktion „Import“ im Kontextmenü befinden sich die Untermenüs:

- Verbindungsart
- Slaves
- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs können Konfigurierte \*.CSV Dateien zur Konfiguration des Modbus Master importiert werden.

### 1.8.5 Export

Unter der Menüfunktion „Export“ im Kontextmenü befinden sich die Untermenüs:

- Verbindungsart
- Slaves
- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs kann die Konfiguration des Modbus Master in \*.CSV Dateien exportiert werden.

### 1.8.6 Kopieren

Kopiert den Modbus Master inklusive Konfiguration in die Zwischenablage.

### 1.8.7 Löschen

Löscht den gewählten Modbus Master aus dem Projekt.

---

<b>Achtung</b>	Mit der Menüfunktion „Löschen“ wird der Modbus Master gelöscht. Vergewissern Sie Sich, dass Sie den Modbus Master wirklich löschen wollen. Archivieren Sie Ihr Projekt, bevor Sie „Löschen“ ausführen.
----------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

## 1.8.8 Eigenschaften

### 1.8.8.1 Register „Allgemein“

Mit *Eigenschaften* werden der Name und die Beschreibung für den Modbus Master eingegeben. Zudem werden hier die Parameter eingestellt, wenn der Modbus Master zusätzlich als TCP und/oder UDP Gateway arbeiten soll.

Element	Beschreibung
Typ	Modbus Master
Name	Beliebiger, eindeutiger Name, für Modbus Master
Beschreibung	Beliebige, eindeutige Beschreibung für Modbus Master
TCP-Gateway aktivieren	Ist das TCP Modbus Gateway aktiviert, muss mindestens eine Modbus-RS-485 Schnittstelle konfiguriert werden
TCP-Port	Standard: 502 Es können auch andere TCP-Ports konfiguriert werden. Dabei ist die Port-Belegung bei der Internet Corporation for Assigned Names and Numbers (ICANN) zu beachten.
Maximale Anzahl TCP-Verbindungen	Maximale Anzahl gleichzeitig offener TCP-Verbindungen als Server. Wertebereich: 1 bis 32 Vorgabewert: 5
UDP-Gateway aktivieren	Ist das UDP Modbus Gateway aktiviert, muss mindestens eine Modbus-RS-485 Schnittstelle konfiguriert werden
UDP-Port	Standard: 502 Es können auch andere UDP-Ports konfiguriert werden. Dabei ist die Port-Belegung bei der Internet Corporation for Assigned Names and Numbers (ICANN) zu beachten.
Maximale Länge der Queue für externe Anforderungstelegramme	Länge der Gateway-Warteschlange für noch nicht beantwortete Anforderungstelegramme von anderen Mastern ; wird nur beachtet, wenn ein Gateway aktiviert ist. Wertebereich: 1 bis 20 Vorgabewert: 3

Tabelle 13: Eigenschaften des Modbus Master

### 1.8.8.2 Register „CPU/COM “

Die Vorgabewerte für die Parameter sorgen für den schnellstmöglichen Datenaustausch der Modbus-Daten zwischen dem COM-Prozessor (COM) und dem CPU-Prozessor (CPU) in der *HIMatrix* Steuerung.

Diese Parameter sollten nur dann geändert werden, wenn eine Reduzierung der COM- und/oder CPU-Auslastung für eine Anwendung erforderlich sind und der Prozess dies zulässt.



**Die Änderung der Parameter wird nur für den erfahrenen Programmierer empfohlen.**

**Eine Erhöhung der COM und CPU Aktualisierungszeit bedeutet auch, dass die tatsächliche Aktualisierungszeit der Modbus-Daten erhöht wird. Die Zeitanforderungen der Anlage sind zu prüfen.**

Beachten Sie auch den Parameter „Master-Slave Datenaustausch [ms]“ (siehe 1.9.4.6 und 1.10.4.6), der die Aktualisierungszeit der Modbus Daten vom/zum Modbus-Slave festlegt.

Dieser kann entsprechend der COM/CPU-Aktualisierungszeit erhöht werden.

Element	Beschreibung
Refresh Rate [ms]	<p>Aktualisierungszeit in Millisekunden, mit der die Daten des Protokolls zwischen COM und CPU ausgetauscht werden.</p> <p>Ist die „Refresh Rate“ Null oder kleiner als die Zykluszeit der Steuerung, dann erfolgt der Datenaustausch so schnell wie möglich.</p> <p>Wertebereich: 0 bis (<math>2^{31}-1</math>)</p> <p>Vorgabewert: 0</p>
In einem Zyklus	<p><b>Aktiviert</b> Transfer der gesamten Daten des Protokolls von der CPU zur COM innerhalb eines Zyklus der CPU.</p> <p><b>Deaktiviert</b> Transfer der gesamten Daten des Protokolls von der CPU zur COM, verteilt über mehrere CPU Zyklen zu je 900 Byte pro Datenrichtung. Damit kann eventuell auch die Zykluszeit der Steuerung reduziert werden.</p> <p>Vorgabewert: Aktiviert</p>

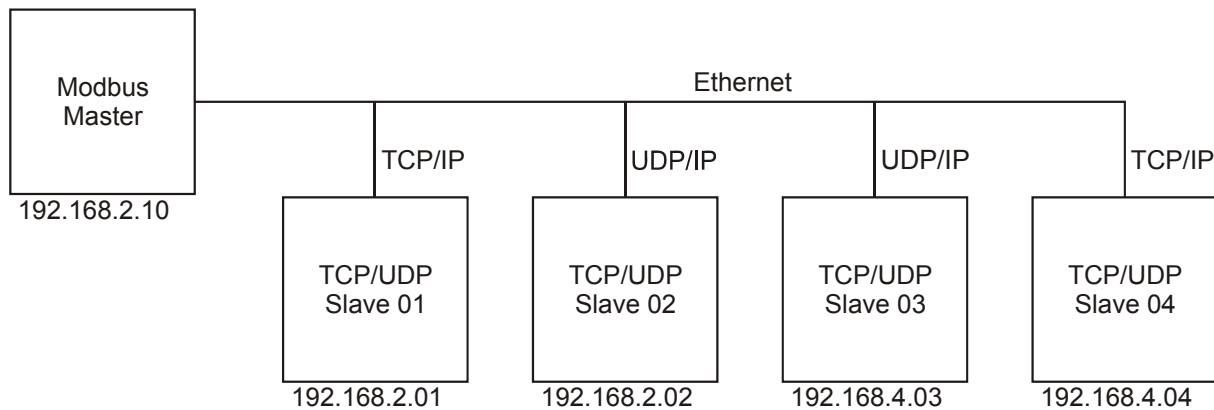
Tabelle 14: Eigenschaften des Modbus Master



## 1.9 Ethernet Slaves

Im Verzeichnis „Ethernet Slaves“ des Modbus Master werden die TCP/UDP Slaves angelegt.

Der Modbus Master kommuniziert mit seinen TCP/UDP Slaves über TCP/IP oder UDP/IP. Im Modbus Master können bis zu 32 TCP/IP und 32 UDP/IP Slaves konfiguriert werden.



**Bild 6: Modbus Kommunikation über TCP/IP**

**Hinweis:** Befinden sich die TCP/UDP Slaves und der Modbus Master in verschiedenen Subnetzen, müssen in der Routing-Tabelle die entsprechenden benutzerdefinierten Routen eingetragen werden.

Beachten Sie für das definieren der „Routing-Tabelle“ auch die **ELOP II Factory** Online Hilfe.

Das Kontextmenü des Verzeichnis „Ethernet Slaves“ enthält die folgenden Funktionen

Ethernet Slave
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.9.1 Neu

Mit der Menüfunktion „Neu“ im Kontextmenü des Verzeichnis „Ethernet Slaves“ wird ein neuer TCP/UDP Slave dem Verzeichnis „Ethernet Slaves“ hinzugefügt.

### 1.9.2 Einfügen

Mit der Menüfunktion „Einfügen“ im Kontextmenü des Verzeichnis „Ethernet Slaves“ kann ein TCP/UDP Slave aus einer anderen Konfiguration kopiert werden.

### 1.9.3 Eigenschaften

Über die Menüfunktion „Eigenschaften“ im Kontextmenü des Verzeichnis „Ethernet Slaves“ wird der Dialog „Eigenschaften“ geöffnet. Hier kann der Name des Verzeichnis „Ethernet Slaves“ geändert werden.

### 1.9.4 TCP/UDP Slave

Das Kontextmenü des TCP/UDP Slave enthält die folgenden Funktionen

TCP/UDP Slave
Signale verbinden
Validieren
Neu
Import
Export
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

#### 1.9.4.1 Signale verbinden

Das Dialogfenster „Signal-Zuordnungen“ aus dem Kontextmenü des TCP/UDP Slaves stellt die zwei Statussignale

- Slave-Zustand und
- Slave-Aktivierungssteuerung

bereit, die es erlauben, den Zustand des TCP/UDP Slave im Anwenderprogramm auszuwerten und den TCP/UDP Slave zu steuern.

Eingänge	Bedeutung
Slave-Zustand	<p>Verbindungsstatus des TCP/UDP Slave:</p> <p>bis COM BS-Version 8.0</p> <p>0: Verbunden</p> <p>1: Deaktiviert</p> <p>2: Nicht verbunden</p> <p>ab COM BS-Version 8.0</p> <p>0: Nicht verbunden</p> <p>1: Deaktiviert</p> <p>2: Verbunden</p>

Tabelle 15: Dialogfenster „Signale verbinden“ von TCP/UDP Slave

Ausgänge	Bedeutung
Slave-Aktivierungssteuerung	Hiermit kann der TCP/UDP Slave vom Anwenderprogramm deaktiviert oder aktiviert werden. 0: Aktivieren 1: Deaktivieren

Tabelle 16: Dialogfenster „Signale verbinden“ von TCP/UDP Slave

#### 1.9.4.2 Validieren

Vor der Codegenerierung kann die Parametrierung des TCP/UDP Slave getestet werden. In der Strukturansicht wird der TCP/UDP Slave selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

#### 1.9.4.3 Neu

Mit „Neu“ aus dem Kontextmenü des TCP/UDP Slave wird ein neues Anforderungstelegramm dem TCP/UDP Slave hinzugefügt.

#### 1.9.4.4 Import

Unter der Menüfunktion „Import“ im Kontextmenü befinden sich die Untermenüs:

- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs können Konfigurierte \*.CSV Dateien zur Konfiguration des TCP/UDP Slave importiert werden.

#### 1.9.4.5 Export

Unter der Menüfunktion „Export“ im Kontextmenü befinden sich die Untermenüs:

- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs kann die Konfiguration des TCP/UDP Slave in \*.CSV Dateien exportiert werden.

### 1.9.4.6 Eigenschaften

Mit *Eigenschaften* werden die Parameter für die TCP/IP oder UDP/IP Kommunikation mit dem TCP/UDP Slave eingestellt.

Element	Beschreibung
Typ	TCP/UDP Slave
Name	Beliebiger, eindeutiger Name, für den TCP/UDP Slave
Beschreibung	Beliebige, eindeutige Beschreibung für den TCP/UDP Slave
IP-Adresse	IP-Adresse des TCP/UDP Slave
Port	Standard: 502 Es können auch andere TCP/UDP-Ports konfiguriert werden. Dabei ist die Port-Belegung bei der Internet Corporation for Assigned Names and Numbers (ICANN) zu beachten.
Kommunikationsart IP-Protokoll	TCP oder UDP Vorgabewert: TCP
TCP-Verbindung nur bei Bedarf	Falls das Transportprotokoll TCP ist; kann man hier einstellen, ob die Verbindung zu diesem Slave nach jedem Datenaustausch automatisch abgebaut werden soll  TRUE: Die Verbindung soll abgebaut werden, FALSE Die Verbindung wird nicht abgebaut. Vorgabewert: FALSE
Master-Slave Datenaustausch [ms]	Intervall für den Datenaustausch mit diesem Slave 1 bis $(2^{31}-1)$ . Konnte der Slave nach „Maximale Anzahl Sendewiederholungen“ nicht erreicht werden, wird das Intervall „Master-Slave Datenaustausch“ um das vierfache hochgesetzt. Siehe auch Aktualisierungszeit zwischen CPU und COM 1.8.8.2.
Maximale Anzahl Sendewiederholungen	Maximale Anzahl an Sendewiederholungen, falls Slave nicht antwortet.  Die Anzahl der Sendewiederholungen kann beliebig eingestellt werden (0 bis 65535). Bei TCP/IP immer „0“, nicht änderbar. Empfohlen wird eine Anzahl von einer bis acht Sendewiederholungen.
Receive Timeout [ms]	Receive Timeout für diesen Slave [ms]. Nach dieser Zeit wird ein neuer Sendeversuch gestartet.

Tabelle 17: Eigenschaften des TCP/UDP Slave

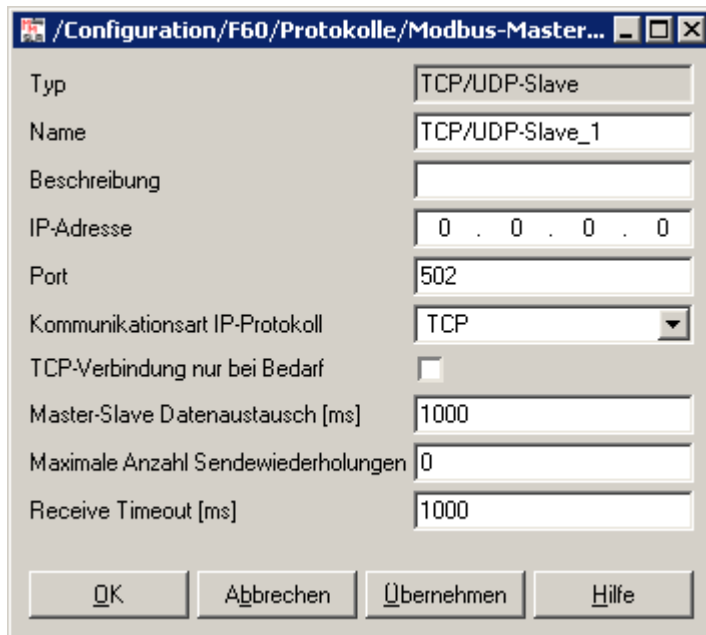
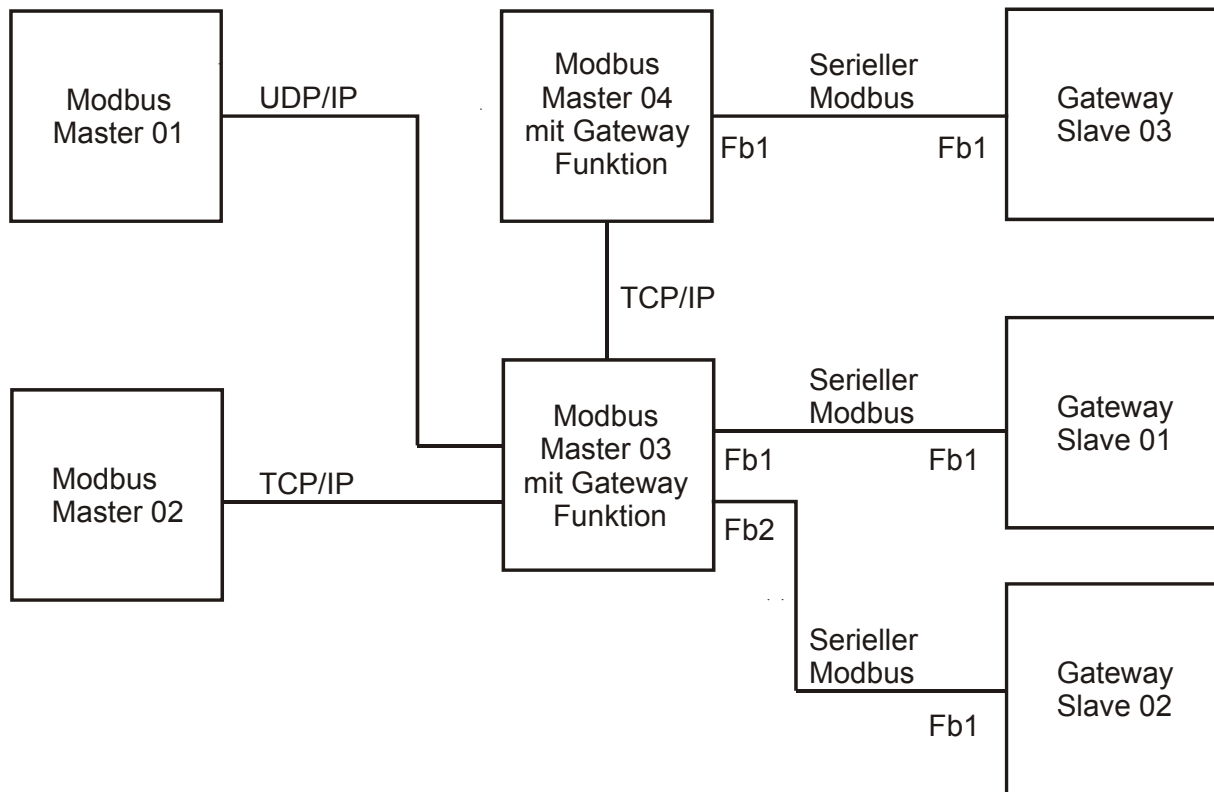


Bild 7: Dialogfenster „Eigenschaften“ des TCP/UDP Slave

### 1.10 Modbus Gateway

Ein Modbus Master kann zusätzlich als TCP und/oder UDP Gateway arbeiten, wenn die Gateway-Funktion konfiguriert wurde.



**Bild 8: Modbus Kommunikation über Modbus Master mit Gatewayfunktion**

Die Modbus Master 01, 02 und 03 kommunizieren über ein Gateway mit ihren seriellen Slaves. Aus diesem Grund müssen die seriellen Modbus Slaves 01 bis 03 in den zugehörigen Modbus Mastern als Gateway Slaves konfiguriert werden.

---

**Hinweis:** Die Receive Timeout [ms] der Modbus Slaves muss im Modbus Master höher eingestellt werden als im Modbus Master (Gateway).

---

- Der Modbus Master 01 kommuniziert mit dem Modbus Slave 01 über den Modbus Master 03 (mit Gateway-Funktion und serieller Schnittstelle fb1 und fb2).
- Der Modbus Master 02 kommuniziert mit dem Modbus Slave 02 über den Modbus Master 03 (mit Gateway-Funktion und serieller Schnittstelle fb1 und fb2).
- Gleichzeitig kommuniziert der Modbus Master 03 mit dem Modbus Slave 03 über den Modbus Master 04 (mit Gateway-Funktion und serieller Schnittstelle fb1).

---

**Hinweis:** Befinden sich der Modbus Gateway und der Modbus Master in verschiedenen Subnetzen, müssen in der Routing-Tabelle die entsprechenden benutzerdefinierten Routen eingetragen werden.  
Beachten Sie für das Definieren der „Routing-Tabelle“ auch die **ELOP II Factory** Online Hilfe.

---

Bis zu 122 serielle Modbus Slaves können über die serielle Schnittstelle adressiert werden. Der Bereich der Slave-Adresse ist 1 bis 247.

Maximal 32 TCP/IP- und 32 UDP/IP-Slaves kann der Modbus Master gleichzeitig bedienen.

Das Kontextmenü des Modbus Gateway enthält die folgenden Funktionen.

Modbus Gateway
Validieren
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.10.1 Validieren

Vor der Codegenerierung kann die Parametrierung des Modbus Gateway getestet werden. In der Strukturansicht wird der Modbus Gateway selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 1.10.2 Neu

Mit der Menüfunktion „Neu“ aus dem Kontextmenü des Modbus Gateway kann ein neuer Gateway Slave dem Modbus Master hinzugefügt werden.

### 1.10.3 Eigenschaften

Mit *Eigenschaften* werden die Parameter und die IP-Adresse für den Modbus Gateway eingestellt, über den der Modbus Master mit seinen Modbus Slaves kommunizieren soll.

Element	Beschreibung
Typ	Modbus Gateway
Name	Beliebiger, eindeutiger Name, für den Gateway
Beschreibung	Beliebige, eindeutige Beschreibung für den TCP/UDP Slave
Kommunikations IP-Protokoll	TCP oder UDP Vorgabewert: TCP
IP-Adresse	IP-Adresse des Gateways, über welches der Modbus Master mit seinen Modbus Slave kommunizieren soll. Vorgabewert: (0.0.0.0)
Port	Standard: 502

Tabelle 18: Eigenschaften des „Modbus Gateway“

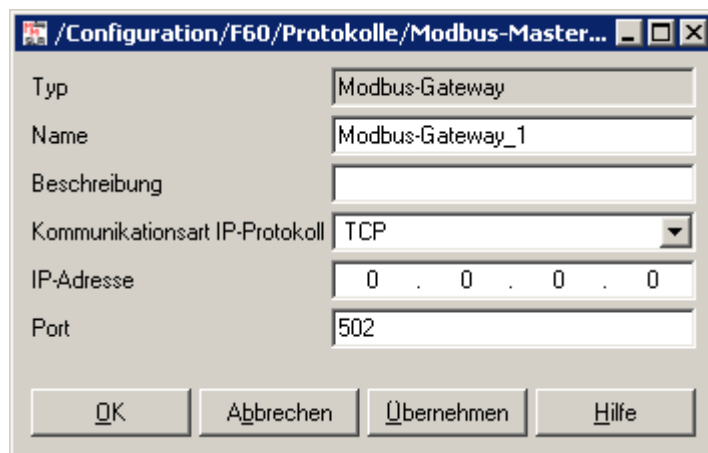


Bild 9: Dialogfenster „Eigenschaften“ des „Modbus Gateway“



### 1.10.4 Gateway Slave

Das Kontextmenü des Gateway Slave enthält die folgenden Funktionen.

Gateway Slave
Signale verbinden
Validieren
Neu
Import
Export
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

#### 1.10.4.1 Signale verbinden

Das Dialogfenster „Signal-Zuordnungen“ aus dem Kontextmenü des Gateway Slaves stellt die zwei Statussignale

- Slave-Zustand und
- Slave-Aktivierungssteuerung

bereit, die es erlauben, den Zustand des TCP/UDP Slave im Anwenderprogramm auszuwerten und den Gateway Slave zu steuern.

Eingänge	Bedeutung
Slave-Zustand	<p>Verbindungsstatus des Gateway Slave:</p> <p>bis COM BS-Version 8.0:</p> <p>0: Verbunden</p> <p>1: Deaktiviert</p> <p>2: Nicht verbunden</p> <p>ab COM BS-Version 8.0:</p> <p>0: Nicht verbunden</p> <p>1: Deaktiviert</p> <p>2: Verbunden</p>

Tabelle 19: Dialogfenster „Signale verbinden“ von Gateway Slave

Ausgänge	Bedeutung
Slave-Aktivierungssteuerung	<p>Hiermit kann der Gateway Slave vom Anwenderprogramm deaktiviert oder aktiviert werden.</p> <p>0: Aktivieren</p> <p>1: Deaktivieren</p>

Tabelle 20: Dialogfenster „Signale verbinden“ von Gateway Slave

#### 1.10.4.2 Validieren

Vor der Codegenerierung kann die Parametrierung des Gateway Slave getestet werden. In der Strukturansicht wird der Gateway Slave selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

#### 1.10.4.3 Neu

Mit der Menüfunktion „Neu“ aus dem Kontextmenü des Gateway Slave wird ein neues Modbus-Anforderungstelegramm dem Gateway Slave hinzugefügt.

#### 1.10.4.4 Import

Unter der Menüfunktion „Import“ im Kontextmenü befinden sich die Untermenüs:

- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs können Konfigurierte \*.CSV Dateien zur Konfiguration des Gateway Slave importiert werden.

#### 1.10.4.5 Export

Unter der Menüfunktion „Export“ im Kontextmenü befinden sich die Untermenüs:

- Orders der Slaves
- Signale der Slaves

Über diese Untermenüs kann die Konfiguration des Gateway Slave in \*.CSV Dateien exportiert werden.

#### 1.10.4.6 Eigenschaften

Mit „Eigenschaften“ werden die Parameter für den Gateway Slave eingestellt.

Element	Beschreibung
Typ	Gateway Slave
Name	Beliebiger, eindeutiger Name, für den Gateway Slave
Beschreibung	Beliebige, eindeutige Beschreibung für den Gateway Slave
Slave-Adresse	1 bis 247
TCP-Verbindung nur bei Bedarf	<p>Falls das Transportprotokoll TCP ist; kann man hier einstellen, ob die Verbindung zu diesem Slave nach jedem Datenaustausch automatisch abgebaut werden soll</p> <p>TRUE: Die Verbindung soll abgebaut werden, FALSE Die Verbindung wird nicht abgebaut.</p>

Element	Beschreibung
	Vorgabewert: FALSE
Master-Slave Datenaustausch [ms]	Aktualisierungsrate für diesen Slave. Konnte der Slave nach „Anzahl Sendeversuchen“ nicht erreicht werden, wird das Aktualisierungsintervall um das vierfache hochgesetzt. Vorgabewert: 1000 Siehe auch Aktualisierungszeit zwischen CPU und COM 1.8.8.2.
Maximale Anzahl Sendewiederholungen	Maximale Anzahl an Sendewiederholversuchen, falls Slave nicht antwortet. Die Anzahl der Sendeversuche kann beliebig eingestellt werden. Bei TCP/IP immer „0“, nicht änderbar Empfohlen wird jedoch eine Anzahl von null bis acht Sendeversuchen. Vorgabewert: 1
Receive Timeout [ms]	ReceiveTimeout für diesen Slave [ms]. Nach dieser Zeit wird ein neuer Sendeversuch gestartet. Vorgabewert: 1000

Tabelle 21: Eigenschaften des „Gateway Slave“

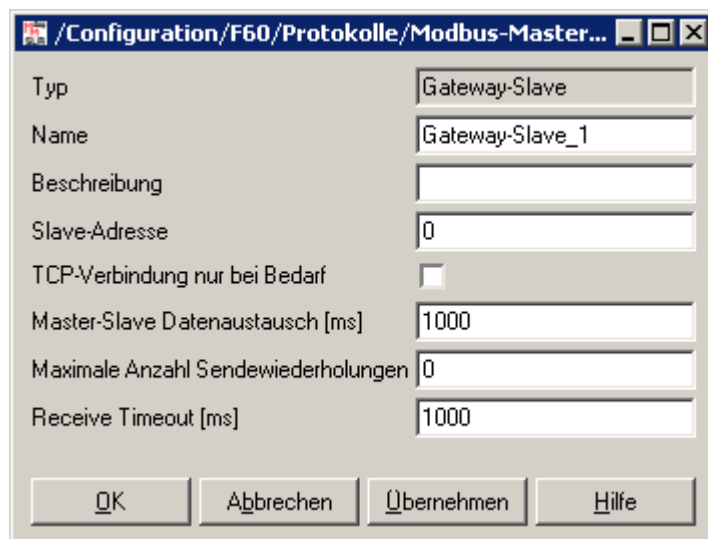
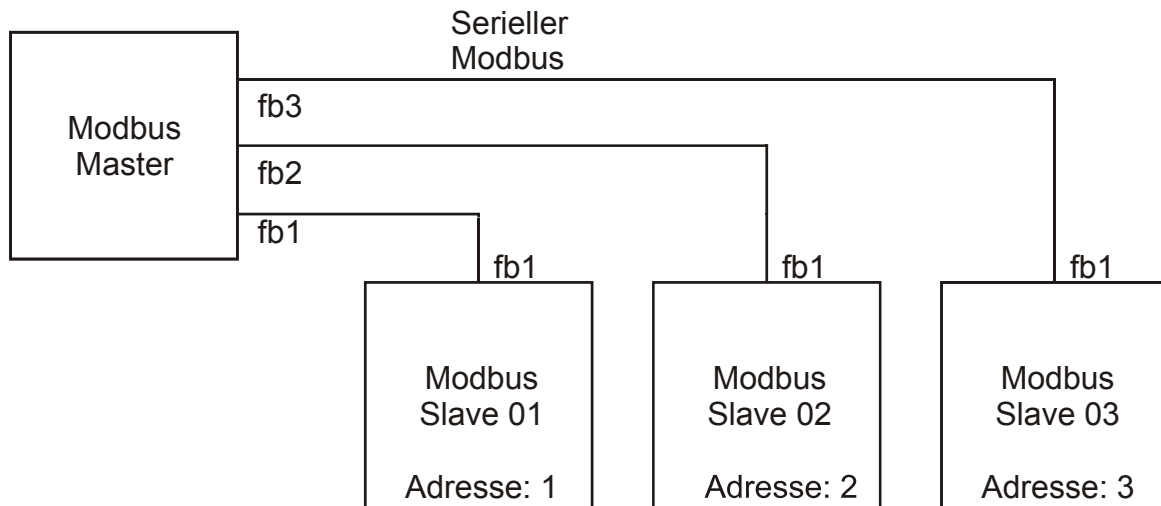


Bild 10: Dialogfenster „Eigenschaften“ des „Gateway Slave“

### 1.11 Serieller Modbus

Der Modbus Master kommuniziert mit seinen Seriellen Slaves über den seriellen Modbus.

**Hinweis** Die Pin-Belegung der SUB-D-Anschlüsse (fb1, fb2, fb3) wird in den Datenblättern der jeweiligen *HIMatrix* Steuerung beschrieben.



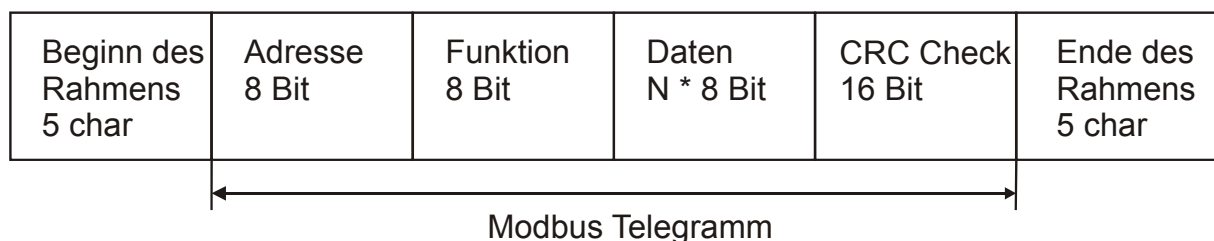
**Bild 11: Modbus Kommunikation über den seriellen Bus**

Pro Modbus Master können bis zu 247 Serielle Slaves konfiguriert werden. Jede Slave-Adresse (1 bis 247) muss eindeutig einem Seriellen Slave zugeordnet werden.

Nach der Norm sind insgesamt drei Repeater zulässig, so dass maximal 122 Busteilnehmer pro serielle Schnittstelle eines Masters möglich sind.

Der HIMA Modbus Master unterstützt die Datenübertragung im RTU-Format (Remote Terminal Unit).

Der RTU Telegrammrahmen bei HIMA beginnt und endet mit den vom Anwender vorgegebenen Idle-Zeichen (Vorgabewert: 5 Idle-Zeichen).



**Bild 12: RTU Telegrammrahmen bei HIMA**

Das Kontextmenü des Serieller Modbus enthält die folgenden Funktionen

Serieller Modbus
Validieren
Neu
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 1.11.1 Validieren

Vor der Codegenerierung kann die Parametrierung des Seriellen Modbus getestet werden. In der Strukturansicht wird der Seriellen Modbus selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 1.11.2 Neu

Mit der Menüfunktion „Neu“ aus dem Kontextmenü des Seriellen Modbus wird ein neuer Modbus Slave dem Seriellen Modbus hinzugefügt.

### 1.11.3 Eigenschaften

Mit *Eigenschaften* werden die Parameter für den seriellen Modbus eingestellt.

Element	Beschreibung
Typ	Serieller Modbus
Name	Name des seriellen Modbus ist durch den Anwender wählbar
Beschreibung	Beliebige, eindeutige Beschreibung für den seriellen Modbus
Schnittstelle	Die Feldbusschnittstelle, die für den Modbus Master benutzt werden soll (fb1, fb2, fb3).
Baudrate [bps]	Übertragungsgeschwindigkeit für RS-485 mögliche Werte: 115200 bps 76800 bps 62500 bps 57600 bps

Element	Beschreibung
	38400 bps 19200 bps 9600 bps 4800 bps 2400 bps 1200 bps 600 bps 300 bps Vorgabewert: 57600
Parität	keine ungerade gerade Vorgabewert: gerade
Stop Bits	Standard (passt die Anzahl der Stop Bits der Parität an: mit Parität = 1 Stop bit, keine Parität 2 Stop bit.) ein Stopbit zwei Stopbits Vorgabewert: Standard
Anzahl Idle-Zeichen	Die Anzahl der Idle-Zeichen am Beginn und Ende des RTU-Telegrammrahmens. Wertebereich: 0 bis 65535 Vorgabewert: 5 Zeichen

Tabelle 22: Eigenschaften des „Seriellen Modbus“

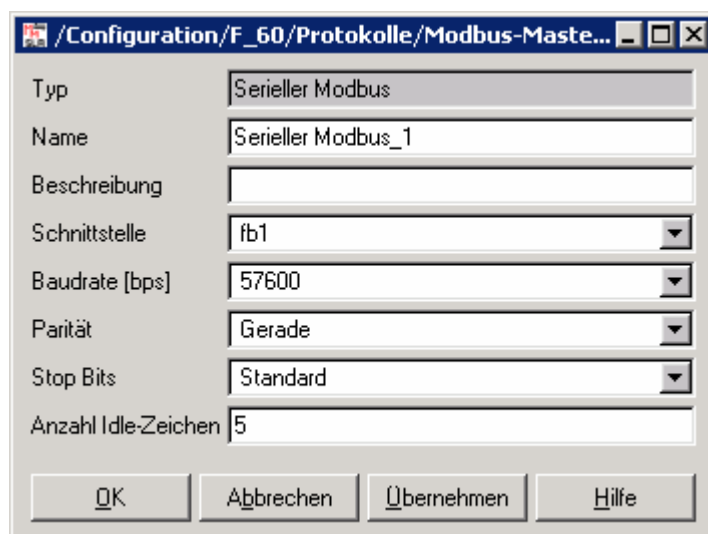


Bild 13: Dialogfenster „Eigenschaften“ des „Serieller Modbus“

### 1.11.4 Modbus Slave

Das Kontextmenü des Modbus Slave enthält die folgenden Funktionen.

Modbus Slave
Signale verbinden
Validieren
Neu
Import
Export
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

#### 1.11.4.1 Signale verbinden

Über die Menüfunktion „Signale verbinden“ aus dem Kontextmenü des Modbus Slave wird Das Dialogfenster „Signal-Zuordnungen“ geöffnet.

Eingänge	Bedeutung
Slave-Zustand	Verbindungsstatus des Modbus Slave: Bis COM BS-Version 8.0 0: Verbunden 1: Deaktiviert 2: Nicht verbunden Ab COM BS-Version 8.0 2: Verbunden 1: Deaktiviert 0: Nicht verbunden

Tabelle 23: Dialogfenster „Signale verbinden“ von Modbus-Slave

Ausgänge	Bedeutung
Slave-Aktivierungssteuerung	Hiermit kann der Modbus Slave vom Anwenderprogramm deaktiviert oder aktiviert werden. 0: Aktivieren 1: Deaktivieren

Tabelle 24: Dialogfenster „Signale verbinden“ von Modbus-Slave

#### 1.11.4.2 Validieren

Vor der Codegenerierung kann die Parametrierung des Modbus Slave getestet werden. In der Strukturansicht wird der Modbus Slave selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

#### 1.11.4.3 Neu

Mit der Menüfunktion „Neu“ aus dem Kontextmenü des Modbus Slave wird ein neues Modbus Anforderungstelegramm dem Modbus Slave hinzugefügt.

#### 1.11.4.4 Import

Unter der Menüfunktion „Import“ im Kontextmenü befinden sich die Untermenüs:

-Orders der Slaves

-Signale der Slaves

Über diese Untermenüs können Konfigurierte \*.CSV Dateien zur Konfiguration des Modbus Slave importiert werden.

#### 1.11.4.5 Export

Unter der Menüfunktion „Export“ im Kontextmenü befinden sich die Untermenüs:

-Orders der Slaves

-Signale der Slaves

Über diese Untermenüs kann die Konfiguration des Modbus Slave in \*.CSV Dateien exportiert werden.

#### 1.11.4.6 Eigenschaften

Mit *Eigenschaften* werden die Parameter für den Modbus Slave eingestellt.

Element	Beschreibung
Typ	Modbus Slave
Name	Name des Modbus Slave; durch den Anwender wählbar
Beschreibung	Beliebige, eindeutige Beschreibung für den Modbus Slave
Slave-Adresse	1 bis 247
Aktualisierungsintervall [ms]	Aktualisierungsrate für diesen Slave. Konnte der Slave nach „Anzahl Sendeversuchen“ nicht erreicht werden, wird das Aktualisierungsintervall um das vierfache hochgesetzt.



Anzahl Sendeveruche	<p>Maximale Anzahl an Sendewiederholversuchen, falls Slave nicht antwortet.</p> <p>Die Anzahl der Sendeveruche kann beliebig eingestellt werden (0 bis 65535).</p> <p>Empfohlen wird jedoch eine Anzahl von null bis acht Sendeveruchen.</p>
Receive Timeout [ms]	<p>ReceiveTimeout für diesen Slave [ms]. Nach dieser Zeit wird ein neuer Sendeveruch gestartet.</p> <p>Vorgabewert: 1000 ms</p>

Tabelle 25: Eigenschaften des „Modbus Slave“

**Hinweis:** Die *Receive Timeout* ist beim seriellen Modbus Slave abhängig von der eingestellten Übertragungsgeschwindigkeit. Ist die Baudrate 19200 [bps] oder höher, kann der Vorgabewert für die *Receive Timeout* verwendet werden. Bei niedrigeren Baudraten als 19200 [bps] muss die *Receive Timeout* erhöht werden.

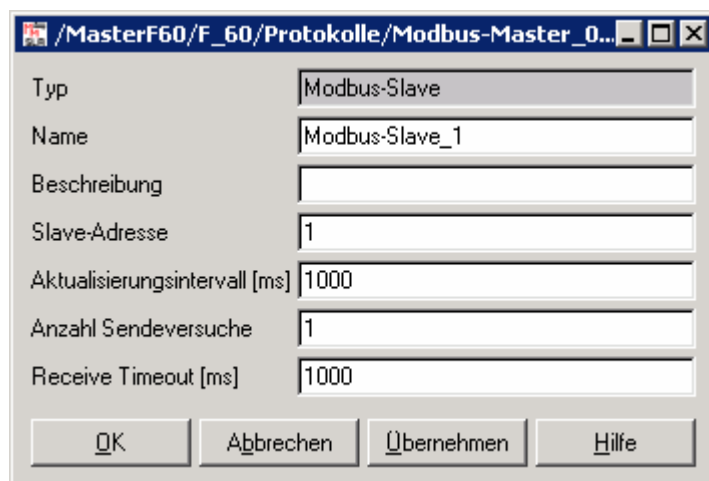


Bild 14: Dialogfenster „Eigenschaften“ des Modbus Slave

## 2 HIMA Modbus Slave

Der HIMA Modbus Slave kann gleichzeitig über die serielle Schnittstelle (RS 485) und über TCP/UDP (Ethernet) mit den Modbus Mastern kommunizieren.

### 2.1 Benötigte Ausstattung und Systemanforderungen

HIMatrix Steuerungen	F30, F35 und F60 ab Hardware Revision: 00
Modbus Slave seriell	<ul style="list-style-type: none"> <li>- HIMA <b>ELOP II Factory</b> ab Version 3.2.x</li> <li>- COM BS ab Version 3.14</li> <li>- CPU BS ab Version 3.14</li> </ul>
Modbus Slave TCP/IP	<ul style="list-style-type: none"> <li>- HIMA <b>ELOP II Factory</b> ab Version 5.2.0</li> <li>- COM BS ab Version 6.22</li> <li>- CPU BS ab Version 4.50</li> </ul>
HIMA Modbus Modul	Siehe Datenblatt der jeweiligen HIMatrix Steuerung Kapitel „Anschlüsse für Feldbus Kommunikation.
Lizenznummer	<p>Jede der beiden Modbus Slave Funktionen muss mit einer Lizenznummer frei geschaltet werden.</p> <ul style="list-style-type: none"> <li>- Modbus Slave seriell</li> <li>- Modbus Slave TCP/IP</li> </ul>

### 2.2 Modbus Slave Eigenschaften

Modbus Slave	Es kann nur ein Modbus Slave pro Ressource konfiguriert werden.
Max. Größe der Sendedaten	8192 Bytes (bis COM-BS V8.32) 16384 Bytes (ab COM-BS V8.32)
Max. Größe der Empfangsdaten	8192 Bytes (bis COM-BS V8.32) 16384 Bytes (ab COM-BS V8.32)
Darstellungsformat der Modbusdaten	Die HIMatrix Steuerungen verwenden das Big Endian Format für die Daten.

Beispiel 32 Bit Daten (z.B. DWORD, DINT):

32 Bit Daten (hex)	0x12345678			
Speicher-Offset	0	1	2	3
Big Endian (HIMatrix)	12	34	56	78
Middle Endian (H51q)	56	78	12	34
Little Endian	78	56	34	12

**Hinweis:** Neben dem Modbus Protokoll können gleichzeitig noch weitere Protokolle (z.B. Profibus-DP, TCP S/R ...) auf der *HIMatrix*-Steuerung betrieben werden.  
 Insgesamt können pro *HIMatrix*-Steuerung 16384 Bytes Daten gesendet und 16384 Bytes Daten empfangen werden.  
 Die Sende- und Empfangsdaten können beliebig zwischen den Protokollen aufgeteilt werden.  
 Die maximale Größe der Sende- und Empfangsdaten pro Protokoll ist vom COM-BS abhängig (siehe 2.2).

### 2.3 Modbus Funktionen

Die folgenden Modbus Funktionen werden vom HIMA Modbus Slave unterstützt.

Element	Code	Typ	Bedeutung
READ COIL	01	BOOL	Lesen mehrerer Signale (BOOL) aus dem Import- oder Export <sup>1)</sup> -Bereich des Slave.
READ DISCRETE INPUT	02	BOOL	Lesen mehrerer Signale (BOOL) aus dem Export-Bereich des Slave.
READ HOLDING REGISTER	03	WORD	Lesen mehrerer Signale beliebigen Typs aus dem Import- oder Export <sup>1)</sup> -Bereich des Slave.
READ INPUT REGISTER	04	WORD	Lesen mehrerer Signale beliebigen Typs aus dem Export-Bereich des Slave.
READ WRITE HOLDING REGISTER	23	WORD	Schreiben und Lesen mehrerer Signale beliebigen Typs in und aus dem Import-Bereich oder Export-Bereich des Slave.
WRITE MULTIPLE COIL	15	BOOL	Schreiben mehrerer Signale (BOOL) in den Import-Bereich des Slave.
WRITE MULTIPLE REGISTER	16	WORD	Schreiben mehrerer Signale beliebigen Typs in den Import-Bereich des Slave.
WRITE SINGLE COIL	05	BOOL	Schreiben eines einzelnen Signals (BOOL) in den Import-Bereich des Slave.
WRITE SINGLE REGISTER	06	WORD	Schreiben eines einzelnen Signals (WORD) in den Import-Bereich des Slave.
Diagnostics	08	x	Nur Subcode 0: Loopback-Funktion des Slave

Element	Code	Typ	Bedeutung
Read Device Identification	43	x	Liefern die Identifikationsdaten des Slaves an den Master.

**Tabelle 26: Modbus Funktionen des HIMA HIMatrix Modbus Slave**

<sup>1)</sup>Export Bereich kann nur bei HIMA Slaves gewählt werden

Hinweis zur Modbus Funktion: Read Device Identification (43)

Der HIMA Modbus Slave liefert die Identifikationsdaten an den Master und unterstützt die folgenden Object-Ids:

Basic:

0x00	VendorName	"HIMA Paul Hildebrandt GmbH"
0x01	ProductCode	"<Seriennummer>"
0x02	MajorMinorRevision	"<CPU Vx.y CRC / COM Vx.y CRC>"

Regular:

0x03	VendorUrl	"http://www.hima.de"
0x04	ProductName	"HIMatrix" oder "GuardPLC" je nach Gerätevariante
0x05	ModelName	"<RessourceTyp>" z.B. "F30", "F31", "F35", "F60"
0x06	UserApplicationName	"<Betriebsmittel-Name>[S.R.S]" aus ELOP-Projekt

Extended:

0x80	CPU BS Version/CRC	"<Vx.y / 0x234adcef>"
0x81	CPU OSL Version/CRC	"<Vx.y / 0x234adcef>"
0x82	CPU BL Version/CRC	"<Vx.y / 0x234adcef>"
0x83	COM BS Version/CRC	"<Vx.y / 0x234adcef>"
0x84	COM OSL Version/CRC	"<Vx.y / 0x234adcef>"
0x85	COM BL Version/CRC	"<Vx.y / 0x234adcef>"
0x86	Konfiguration-CRC	„< 0x234adcef>“

(Weitere Informationen zu Modbus finden Sie in der Spezifikation "Modbus Application Protocol Specification" [www.modbus.org](http://www.modbus.org))

Das Kontextmenü des HIMA Modbus Slave enthält die folgenden Funktionen.

Kontextmenü
Signale verbinden
Validieren
Neu
Import
Export
Kopieren
Einfügen
Löschen
Drucken
Eigenschaften

### 2.3.1 Signale verbinden

Über die Menüfunktion „Signale verbinden“ aus dem Kontextmenü des Modbus Slave wird Das Dialogfenster „Signal-Zuordnungen“ geöffnet.

Im Dialog „Signale Zuordnungen“ werden:

- alle Signale die der Modbus Slave an den Modbus Master sendet im Register „Ausgänge“ eingefügt.

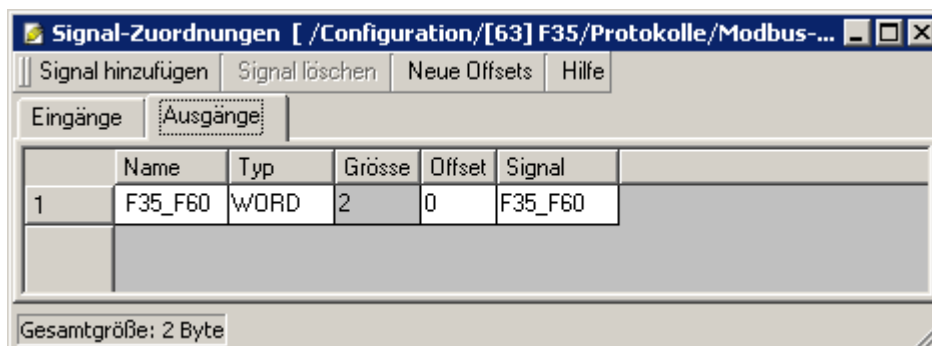


Bild 15: Register „Ausgänge“ im Dialog „Signal Zuordnungen“ des Modbus Slave

- alle Signale, die der Modbus Slave vom Modbus Master empfängt, werden im Register „Eingänge“ eingefügt.

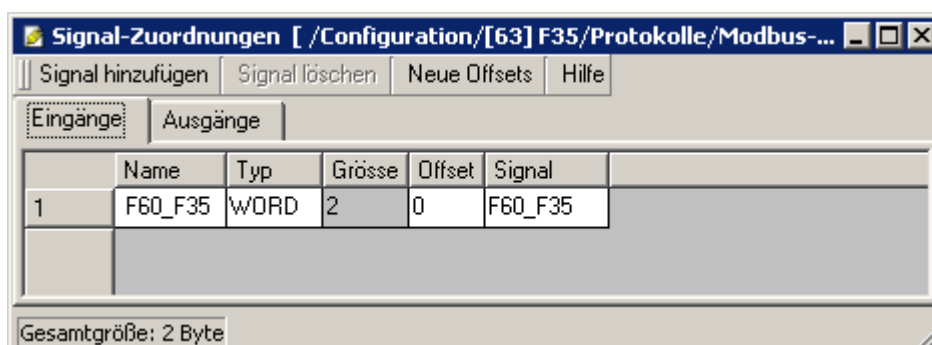


Bild 16: Register „Eingänge“ im Dialog „Signal Zuordnungen“ des Modbus Slave

## 2.4 Beispiel: Schreiben auf den Import-Bereich des HIMatrix Modbus Slaves

Das folgende Beispiel beschreibt, wie ein Modbus Master auf den Import-Bereich des HIMatrix Modbus Slaves zugreifen kann.

Um den Import-Bereich des HIMatrix Modbus Slaves zu beschreiben, werden in diesem Beispiel die folgenden Modbus Anforderungstelegramme benutzt:

- Ein Anforderungstelegramm „Write Multiple Coils (15)“
- Zwei Anforderungstelegramme „Write Multiple Registers (16)“

### 2.4.1 Der Import-Bereich des HIMatrix Modbus Slaves

Der Import-Bereich des HIMatrix Modbus Slaves kann wie im Bild 17 mit Signalen unterschiedlichen Typs angelegt werden.

Die Signale müssen nach der Spalte „Offset“ sortiert sein.

Die Adresse eines Signals im Adressbereich des HIMatrix Modbus Slaves wird aus dem Index eines Signals berechnet (Adresse eines Signals = Index – 1).

**Hinweis:** Beachten Sie, dass die Signale, die größer als ein Byte sind, in den HIMatrix Steuerungen im Big Endian Format gespeichert werden.

	Name	Typ	Grösse	Offset	Signal
1	IB0	BOOL	1	0	IB0
2	IB1	BOOL	1	1	IB1
3	IB2	BOOL	1	2	IB2
4	IB3	BOOL	1	3	IB3
5	IW0	WORD	2	4	IW0
6	IW1	WORD	2	6	IW1
7	IW2	WORD	2	8	IW2
8	IW3	WORD	2	10	IW3
9	IW4	WORD	2	12	IW4
10	IW5	WORD	2	14	IW5
11	IW6	WORD	2	16	IW6
12	IDW0	DWORD	4	18	IDW0
13	IDW1	DWORD	4	22	IDW1
14	IDW2	DWORD	4	26	IDW2
15	IDW3	DWORD	4	30	IDW3

Gesamtgröße: 34 Byte

Bild 17: Import-Bereich des Modbus-Slaves

**Hinweis:** Auf den Import-Bereich des Slaves kann der Master lesend und schreibend zugreifen.

Aus dem Export-Bereich des Slaves kann der Master nur lesen.

In der Spalte „Grösse“ des Dialogfensters „Signal-Zuordnungen“ wird die Größe des Signals in Bytes angezeigt.

## 2.4.2 Konfiguration der Anforderungstelegramme im Modbus Master

In den folgenden Abschnitten wird die Konfiguration der Anforderungstelegramme „Write Multiple Coils (15)“ und „Write Multiple Registers (16)“ beschrieben, die auf den *HIMatrix* Modbus Slave zugreifen sollen.

### 2.4.2.1 Anforderungstelegramm „Write Multiple Coils (15)“

#### Startadresse des Anforderungstelegramms ermitteln

Die Startadresse des Anforderungstelegramms muss aus dem Index des ersten Signals ermittelt werden, das mit diesem Anforderungstelegramm beschrieben wird.

Startadresse = Index – 1 (im *HIMatrix* Modbus Slave, siehe Bild 17)

Startadresse = 1 – 1 = 0

Tragen Sie **0** in das Feld „Address“ ein

#### Größe des Anforderungstelegramms ermitteln

Da es sich bei dem Typ des Anforderungstelegramms um „Write Multiple Coils (15)“ handelt, muss die Anzahl der Signale in Bytes eingegeben werden:

---

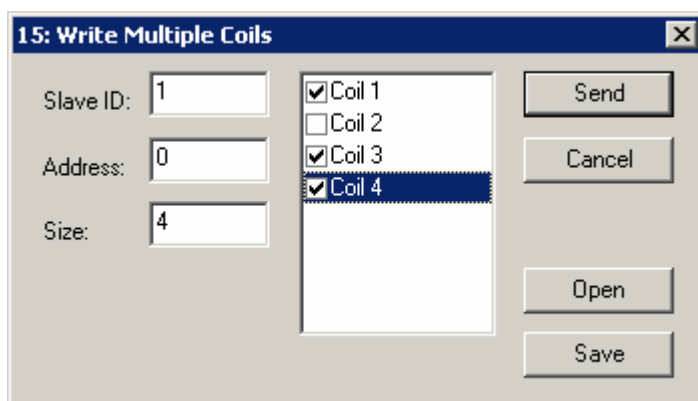
**Hinweis:** Eine BOOL Signal belegt ein Byte im Speicher einer *HIMatrix* Steuerung.

---

Anzahl der BOOL Signale = 4

Größe des Anforderungstelegramms = 4 Byte

Tragen Sie **4** in das Feld „Size“ ein.



**Bild 18:** Anforderungstelegramm „Write Multiple Coils (15)“ im Modbus Master

### 2.4.2.2 Erstes Anforderungstelegramm „Write Multiple Register (16)“

#### Startadresse des Anforderungstelegramms ermitteln

Die Startadresse des Anforderungstelegramms muss aus dem Index des ersten Signals ermittelt werden, das mit diesem Anforderungstelegramm beschrieben wird.

Startadresse = Index – 1 (im *HIMatrix* Modbus Slave, siehe Bild 17)

Adresse = 5 – 1 = 4

Tragen Sie **4** in das Feld „Address“ ein.

#### Größe des Anforderungstelegramms ermitteln

Da es sich bei dem Typ des Anforderungstelegramms um „Write Multiple Register (16)“ handelt, muss die Anzahl der Signale in Word eingegeben werden:

Anzahl der WORD Signale = 7 -> Größe = 7 Word

Anzahl der DWORD Signale = 1 -> Größe = 2 Word

Größe des Anforderungstelegramms = 9 Word

Tragen Sie **9** in das Feld „Size“ ein.

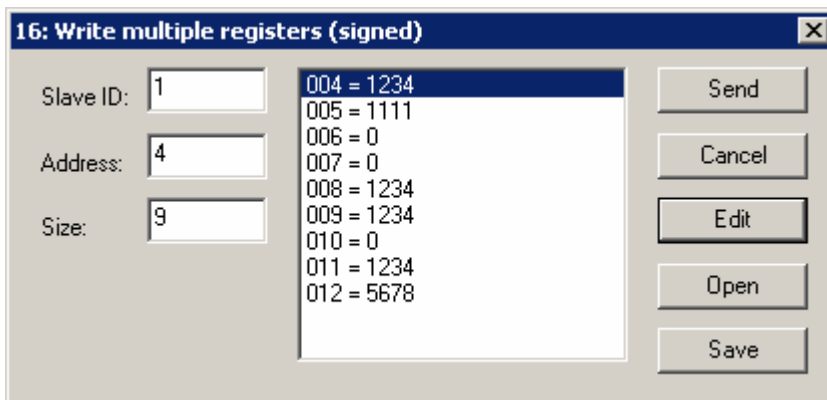


Bild 19: Anforderungstelegramm „Write Multiple Register(16)“ im Modbus Master

**Hinweis:** Beachten Sie, dass die Signale, die größer als ein Byte sind, in den *HIMatrix* Steuerungen im Big Endian Format gespeichert werden. Wenn Sie einen Modbus Master eines anderen Herstellers benutzen, dann müssen Sie darauf achten, dass Sie die Signal-Typen der geschriebenen und gelesenen Signal-Typen richtig interpretieren! (siehe auch 2.2 ).



### 2.4.2.3 Zweites Anforderungstelegramm „Write Multiple Register (16)“

#### Startadresse des Anforderungstelegramms ermitteln

Die Startadresse des Anforderungstelegramms muss aus dem Index des ersten Signals ermittelt werden, das mit diesem Anforderungstelegramm beschrieben wird.

Startadresse = Index – 1 (im *HIMatrix* Modbus Slave, siehe Bild 17)

Adresse = 13 – 1 = 12

Tragen Sie **12** in das Feld “Address” ein.

#### Größe des Anforderungstelegramms ermitteln

Da es sich bei dem Typ des Anforderungstelegramms um „Write Multiple Register (16)“ handelt, muss die Anzahl der Signale in Word eingegeben werden:

Anzahl der DWORD Signale = 3 -> Größe = 6 Word

Größe des Anforderungstelegramms = 6

Tragen Sie **6** in das Feld “Size” ein.

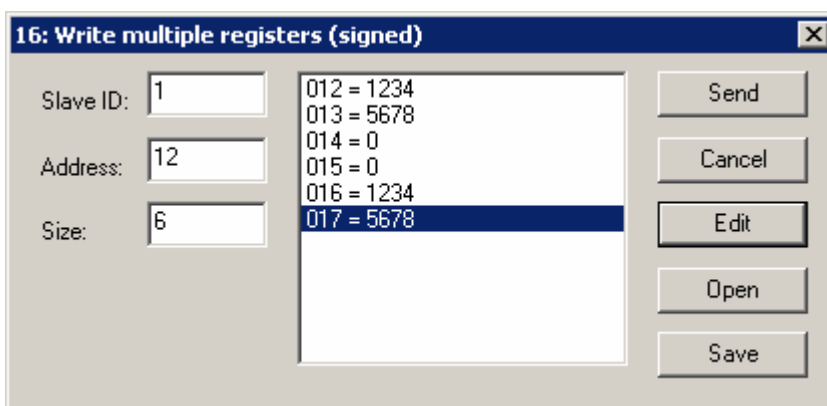


Bild 20: Anforderungstelegramm „Write Multiple Register(16)“ im Modbus Master

**Hinweis:** Beachten Sie, dass die Signale, die größer als ein Byte sind, in den *HIMatrix* Steuerungen im Big Endian Format gespeichert werden. Wenn Sie einen Modbus Master eines anderen Herstellers benutzen, dann müssen Sie darauf achten, dass Sie die Signal-Typen der geschriebenen und gelesenen Signal-Typen richtig interpretieren! (siehe auch 2.2 ).

### 2.4.3 Validation

Vor der Codegenerierung kann die Parametrierung des Modbus Slave getestet werden. In der Strukturansicht wird der Modbus Slave selektiert und im Kontextmenü wird *Validieren* gewählt. In der Fehler-Status-Anzeige werden dann eventuelle Fehler und Warnungen angezeigt.

Die Validation wird zudem automatisch vor jeder Codegeneration durchgeführt. Wird bei der Validation ein Fehler festgestellt, dann wird die Codegeneration abgebrochen.

### 2.4.4 Import

Unter der Menüfunktion „Import“ im Kontextmenü befindet sich das Untermenü:

-Signale

Über dieses Untermenü können über eine konfigurierte \*.CSV Datei, Signale für den Modbus Slave importiert werden.

### 2.4.5 Export

Unter der Menüfunktion „Export“ im Kontextmenü befindet sich das Untermenü:

-Signale

Über dieses Untermenü können Signale des Modbus Slave in \*.CSV Dateien exportiert werden.

### 2.4.6 Kopieren

Mit der Menüfunktion „Kopieren“ aus dem Kontextmenü des Modbus Slave wird der Slave mit allen Konfigurationen Kopiert.

### 2.4.7 Löschen

---

**Achtung:** Mit der Menüfunktion „Löschen“ wird der Modbus Slave gelöscht. Vergewissern Sie Sich, dass Sie den Modbus Slave wirklich löschen wollen. Archivieren Sie Ihr Projekt, bevor Sie „Löschen“ ausführen.

---

### 2.4.8 Eigenschaften

Über die Menüfunktion „Eigenschaften“ im Kontextmenü des Modbus Slave wird der Dialog „Eigenschaften“ geöffnet. Hier werden die folgenden Eigenschaften des Modbus Slave konfiguriert.

### 2.4.9 Register „Allgemein“

Im Register „Allgemein“ werden die folgenden Parameter für den Modbus Slave eingestellt.

Element	Beschreibung
Bereich zum Lesen der Funktionscodes 1 und 3	Hier kann der Anwender den Bereich des Modbus Slave festlegen aus dem die Funktionscodes (1 und 3) lesen sollen (Siehe Beschreibung des Slave).
	Importbereich
	Exportbereich (kompatibel zu 51q)
Bereich zum Lesen bei Funktionscode 23 (Ab COM BS V.6)	Hier kann der Anwender den Bereich des Modbus Slave festlegen aus dem der Funktionscode 23 lesen soll.
	Importbereich Der Master greift auf den Importbereich des Slaves lesend und schreibend zu.
	Exportbereich Der Master liest vom Exportbereich des Slaves und schreibt auf den Importbereich des Slaves.
	Standard: Importbereich

Tabelle 27: Register „Allgemein“ im Dialogfenster „Eigenschaften“ des Slave

**Hinweis:** Für die *HiMatrix*-Steuerungen mit COM BS Versionen V.3 bis V.6 gelten die folgenden Lesebereiche für Funktionscode 1 und 3:

COM BS V.3:	Importbereich
COM BS V.4:	Exportbereich (kompatibel zu H51q)
COM BS V.6:	Konfigurierbar

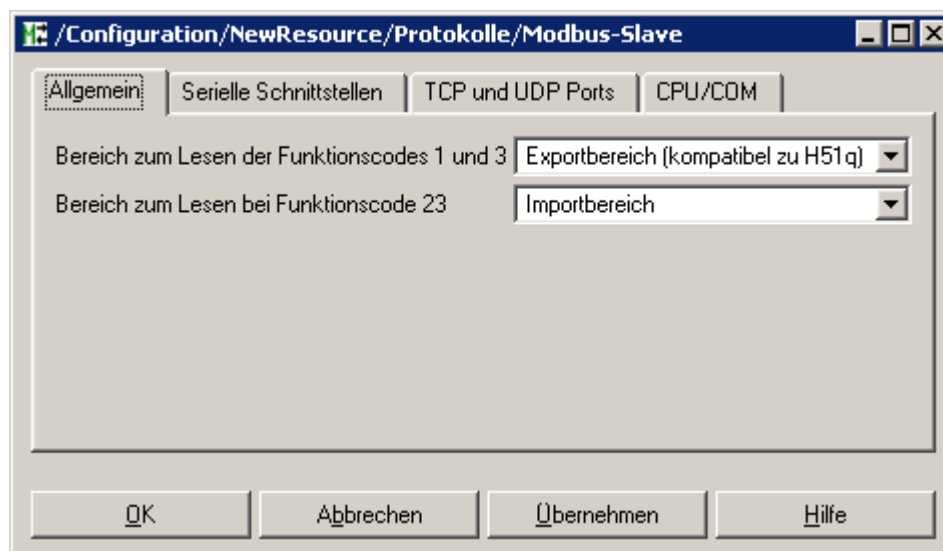


Bild 21: Register „Allgemein“ im Dialogfenster „Eigenschaften“ des Modbus Slave

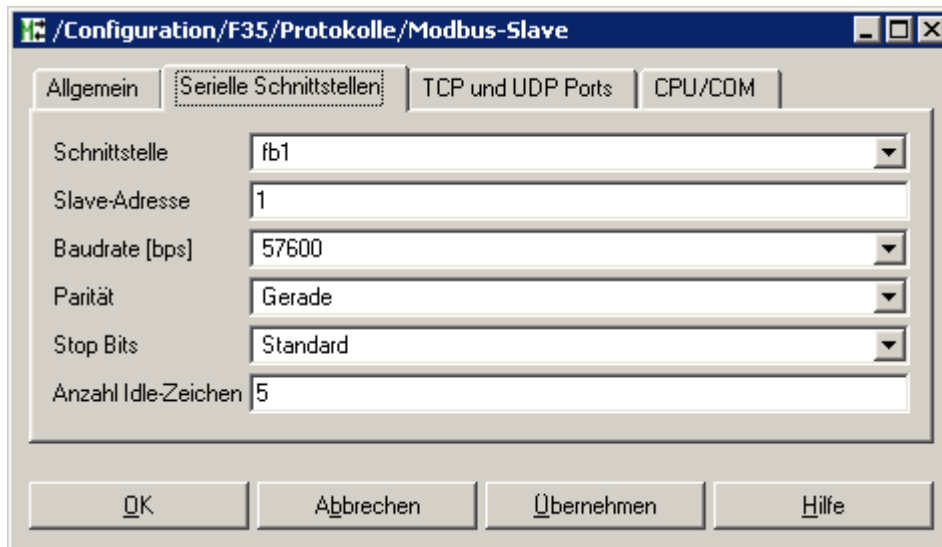
### 2.4.9.1 Register „Serielle Schnittstelle“

Im Register „Serielle Schnittstelle“ werden die Parameter der seriellen Schnittstelle des Modbus Slave eingestellt.

**Hinweis** Die Pin-Belegung der SUB-D-Anschlüsse (fb1, fb2, fb3) wird in den Datenblättern der jeweiligen *HIMatrix* Steuerung beschrieben.

Element	Beschreibung
Schnittstelle	Die Feldbusschnittstelle, die für den Modbus Slave benutzt werden soll (keine,fb1, fb2, fb3).
Slave-Adresse	Busadresse des Slave Wertebereich: 1 bis 247
Baudrate [bps]	Übertragungsgeschwindigkeit für RS-485 mögliche Werte: 115200 bps 76800 bps 62500 bps 57600 bps 38400 bps 19200 bps 9600 bps 4800 bps 2400 bps 1200 bps 600 bps 300 bps Vorgabewert: 57600
Parität	keine ungerade gerade Vorgabewert: gerade
Stop Bits	Standard (passt die Anzahl der Stop Bits der Parität an: mit Parität =1 Stop bit, keine Parität =2 Stop bit.) ein Stoppbit zwei Stoppbits Vorgabewert: Standard
Anzahl Idle-Zeichen	Die Anzahl der Idle-Zeichen am Beginn und Ende des RTU-Telegrammrahmens. Wertebereich: 0 bis 65535 Vorgabewert: 5 Zeichen

Tabelle 28: Register „Serielle Schnittstelle“ im Dialogfenster „Eigenschaften“ des Slave



**Bild 22:** Register „Serielle Schnittstelle“ im Dialogfenster „Eigenschaften“ des Modbus Slave

### 2.4.9.2 Register „TCP und UDP Ports“

Im Register „TCP und UDP Ports“ werden die Parameter für die TCP- und UDP-Ports des Modbus Slave eingestellt.

Element	Beschreibung
TCP aktivieren	TCP/IP Verbindung aktivieren Vorgabewert: Aktiviert
TCP-Port	Vorgabewert: 502
Maximale Anzahl TCP-Verbindungen	Maximale Anzahl gleichzeitig offener TCP-Verbindungen als Server. Wertebereich: 1 bis 20 Vorgabewert: 3
UDP aktivieren	UDP/IP Verbindung aktivieren Vorgabewert: Aktiviert
UDP-Port	Vorgabewert: 502

Tabelle 29: Register „TCP und UDP Ports“ im Dialogfenster „Eigenschaften“ des Slave

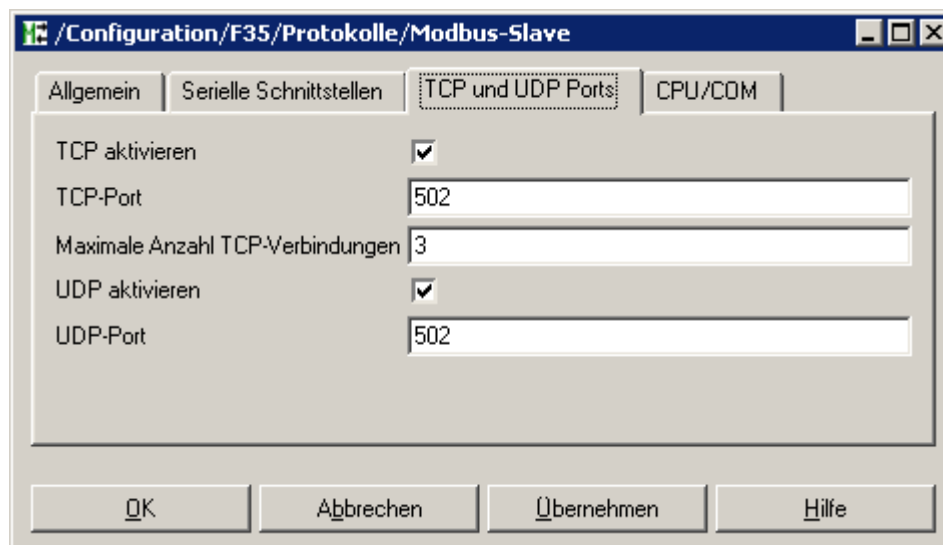


Bild 23: Register „TCP und UDP Ports“ im Dialogfenster „Eigenschaften“ des Slave

### 2.4.9.3 Register „CPU/COM“

Die vorgegebenen Parameter sorgen für den schnellstmöglichen Datenaustausch der Modbus-Daten zwischen dem COM-Prozessor (COM) und dem CPU-Prozessor (CPU) in der *HIMatrix* Steuerung.

Diese Parameter sollten nur dann geändert werden, wenn eine Reduzierung der COM- und/oder CPU-Auslastung für eine Anwendung erforderlich sind und der Prozess dies zulässt.



**Die Änderung der Parameter wird nur für den erfahrenen Programmierer empfohlen.**

**Eine Erhöhung der COM und CPU Aktualisierungszeit bedeutet auch, dass die tatsächliche Aktualisierungszeit der Modbus-Daten erhöht wird. Die Zeitanforderungen der Anlage sind zu prüfen.**

Beachten Sie auch den Parameter im Modbus-Master, der die Aktualisierungszeit der Modbus Daten vom/zum Modbus-Slave festlegt (siehe z.B. „Master-Slave Datenaustausch [ms]“ im HIMA Modbus-Master 1.9.4.6 und 1.10.4.6).

Dieser kann entsprechend der COM/CPU-Aktualisierungszeit erhöht werden.

Element	Beschreibung
Refresh Rate [ms]	<p>Aktualisierungszeit in Millisekunden, mit der die Daten im Modbus-Protokoll zwischen COM und CPU ausgetauscht werden.</p> <p>Ist die „Refresh Rate“ Null oder kleiner als die Zykluszeit der Steuerung, dann erfolgt der Datenaustausch so schnell wie möglich.</p> <p>Wertebereich: 0 bis (<math>2^{31}-1</math>)</p> <p>Vorgabewert: 0</p>
In einem Zyklus	<p><b>Aktiviert</b> Transfer der Modbus-Daten von der CPU zur COM innerhalb eines Zyklus der CPU.</p> <p><b>Deaktiviert</b> Transfer der Modbus-Daten (maximal 900 Byte pro Datenrichtung) von der CPU zur COM über mehrere Zyklen der CPU.</p> <p>Vorgabewert: Aktiviert</p>

**Tabelle 30: Register „COM/CPU“ im Dialogfenster „Eigenschaften“ des Slave**

## 3 Diagnose und Fehlercodes

### 3.1 Control Panel (Register „ModbusMs.“)

Im Register „Modbus“ kann der Anwender die Einstellungen des Master und der Slaves überprüfen. Zudem werden aktuelle Statusinformationen (z.B. Zykluszeit) des Masters und der Slaves angezeigt.

Öffnen Sie im Hardware-Management das Control Panel und wählen Sie das Register „ModbusMs.“.

Das Register „ModbusMs.“ ist in die Rahmen, „Modbus Master“ und „Modbus Slave“ unterteilt.

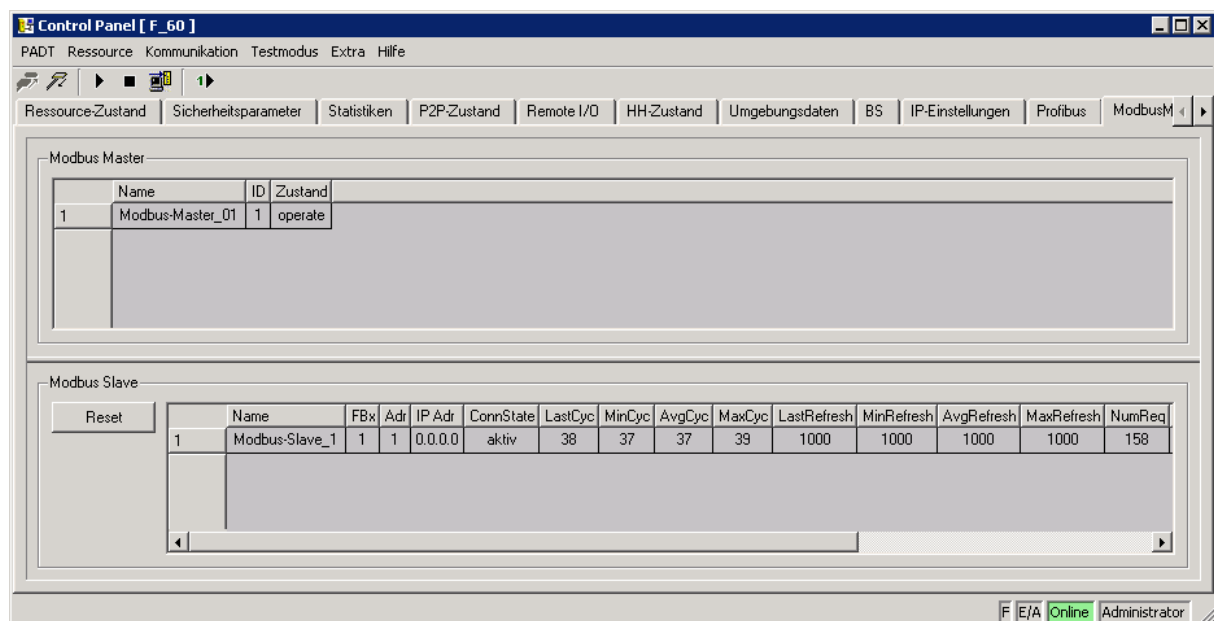


Bild 24: Dialogfenster „Control Panel“, Register „ModbusMs.“

#### 3.1.1 Rahmen „Modbus Master“

Im Rahmen „Modbus Master“ befindet sich ein Anzeigefeld in welchem der aktuelle Zustand des Master angezeigt wird.

Element	Erklärung
Name	Name des Masters
ID	Master-ID
Zustand	Aktueller Operationszustand OFFLINE, OPERATE, UNDEFINED

Tabelle 31: Werte im Anzeigefeld „Masterdaten“



### 3.1.2 Rahmen „Modbus Slave“

Im Rahmen „Modbus Slave“ befindet sich ein Anzeigefeld in welchem die aktuellen Statusinformationen (z.B. Zykluszeit min, max usw.) des selektierten Slave angezeigt werden.

#### **Reset:**

Setzt die statistischen Daten (Zykluszeit min, max usw.) auf null zurück.

Element	Erklärung
Name	Name des Slave
FBx	Nummer der Feldbusschnittstelle des Masters
Adr	Adresse des Slave auf dem seriellen Bus
IPAdr	IP-Adresse des Slave
ConnState	Verbindungszustand verbunden deaktiviert nicht verbunden
LastCyc	Letzte Zykluszeit: Zeit, in welcher der Slave die gesamten Anforderungs- telegramme des letzten Zyklus beantwortet hat.
MinCyc	Min. Zykluszeit
AvgCyc	Mittlere Zyklusdauer
MaxCyc	Längste aufgetretene Zykluszeit
LastRefresh	Letzte Zeitdauer eines Refreshintervalls
MinRefresh	Min. Zeitdauer eines Refreshintervalls
AvgRefresh	Mittlere Zeitdauer eines Refreshintervalls
MaxRefresh	Maximale bisher aufgetretene Refreshzeit
NumReq	Anzahl gesendeter Anforderungstelegramme
NumPosAnswer	Anzahl positiver Antworten
NumTimeouts	Anzahl Timeouts
LastRtt	Letzte Round-Trip Time (RTT) Zeit, in welcher der Slave das letzte Anforderungstele- gramm beantwortet hat.
MinRtt	Min. RTT
AvgRtt	Mittlere RTT
MaxRtt	Maximale bisher aufgetretene RTT

**Tabelle 32: Werte im Anzeigefeld „Slavedaten“**

### 3.1.3 Diagnose Feldbus (FBx) LEDs der seriellen Schnittstellen

Der Zustand der seriellen Modbus Kommunikation wird mit der FBx LED der jeweiligen konfigurierten seriellen Schnittstellen (fb1, fb2, fb3) angezeigt.

#### 3.1.3.1 Funktion der FBx LED beim Modbus Master

Funktion der FBx LED vor COM Betriebssystem V10.38

FBx LED	Beschreibung
AUS	In der Konfiguration des Modbus Master ist kein Modbus Slave zu dieser seriellen Schnittstelle konfiguriert.
Blinkt im Sekundentakt	Kein Datenaustausch! In der Konfiguration des Modbus Master ist mindestens ein Modbus Slave zu dieser seriellen Schnittstelle konfiguriert.
AN	Der Modbus Master befindet sich im Zustand OPERATE. Datenaustausch mit allen auf dieser seriellen Schnittstelle konfigurierten Modbus Slaves findet statt.
Blinkt im ¼ Sekunden-takt	Der Modbus Master befindet sich im Zustand OPERATE. Mindestens einer der auf dieser seriellen Schnittstelle konfigurierten Modbus Slaves ist nicht erreichbar.

Funktion der FBx LED ab COM Betriebssystem V10.38

FBx LED	Beschreibung
AUS	Kein Datenaustausch!
Blinkt	FBx LED blinkt, wenn der Modbus Master Datenpakete sendet oder empfängt.

### 3.1.3.2 Funktion der FBx LED beim Modbus Slave

Funktion der FBx LED vor COM Betriebssystem V10.38

LED FBx	Beschreibung
AUS	1. Die serielle Schnittstelle ist im Modbus Slave nicht konfiguriert. 2. Die <i>HIMatrix</i> Steuerung ist im Zustand STOP.
Blinkt im Sekundentakt	Kein Datenaustausch! Der Modbus Slave ist aber auf dieser seriellen Schnittstelle konfiguriert.
Blinkt in unregelmäßigen Zeitabständen	Datenaustausch ist aktiv. Bei jedem Empfang eines Modbus Telegramms wird die FBx LED ein- und ausgeschaltet.

Funktion der FBx LED ab COM Betriebssystem V10.38

FBx LED	Beschreibung
AUS	Kein Datenaustausch!
Blinkt	FBx LED blinkt, wenn der Modbus Master Datenpakete sendet oder empfängt.

### 3.1.4 Fehlercodes der TCP/IP Verbindung

Die Fehlercodes der TCP/IP Verbindung werden im Hardware-Management im Dialogfenster „Diagnose“ angezeigt.

Öffnen Sie das Dialogfenster „Diagnose“:

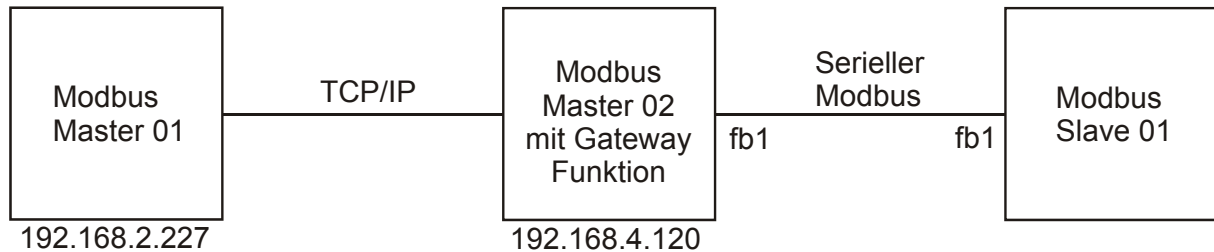
- ☐ Wechseln Sie in das Hardware-Management.
- ☐ Öffnen Sie über das „Hauptmenü“ *Online->Diagnose* das Dialogfenster „Diagnose“.
- ☐ Aktivieren Sie die „Com Kurzzeitdiagnose“, um die Fehlermeldungen anzuzeigen.

Fehlercode	Erklärung
35	Operation ist blockiert
48	Port ist bereits in Verwendung
50	Netzwerk läuft nicht
53	Verbindung durch Software abgebrochen
54	Verbindung durch Partner abgebrochen
55	Kein Pufferspeicher mehr verfügbar
60	Timeout aufgetreten; Verbindung geschlossen
61	Verbindung abgewiesen (durch Partner)
65	Kein Routingeintrag zum Partner vorhanden

**Tabelle 33: Fehlercodes der TCP/IP Verbindung**

## 4 Anwendung mit Modbus Gateway

Der Modbus Master 01 kommuniziert mit dem Modbus Slave 01 über den Modbus Master 02 (mit Gateway-Funktion).



**Bild 25: Modbus Kommunikation über einen Modbus Master mit Gateway Funktion**

---

**Hinweis:** Befinden sich der Modbus Gateway und der Modbus Master in verschiedenen Subnetzen, müssen in der Routing-Tabelle die entsprechenden benutzerdefinierten Routen eingetragen werden. Beachten Sie für das definieren der „Routing-Tabelle“ auch die **ELOP // Factory** Online Hilfe. Beachten Sie die Systemanforderungen und Ausstattung im Kapitel 1.1.

---

### 4.1 Eintrag der Modbus Master in die Routingtabelle

---

**Hinweis:** Wenn Modbus Master 01 und Modbus Master 02 im gleichen Subnet liegen, dann wird kein Routing Eintrag benötigt. Fahren Sie in diesem Fall mit Kapitel 4.2 fort.

---

Routing erlaubt den Datenaustausch zwischen Steuerungen in verschiedenen Subnetzen. In der Routing-Tabelle können maximal acht benutzerdefinierte Routen eingetragen werden.

Eingabe in die Routing-Tabellen

- IP-Adresse des Kommunikationspartners
- Subnet-Maske 255.255.255.255
- Bezeichnung des Ethernet Interface (für *HIMatrix* Steuerungen 0.0.0.1)

**Schritt 1:** Einträge in die Routing-Tabellen des Modbus Master 01:

- ❑ Öffnen Sie das Control Panel der Ressource von Modbus Master 01
- ❑ Öffnen Sie das Register „IP Settings“.
- ❑ Wählen Sie *Hinzufügen* um das Popup-Fenster „Routing-Eintrag hinzufügen“ zu öffnen.
- ❑ Tragen Sie die Parameter aus der folgenden Tabelle in die Routing-Tabelle ein.

Routing-Tabelle von Modbus Master 01		
Route IP	Route Subnet Mask	Gateway/IF
192.168.4.120	255.255.255.255	0.0.0.1

Tabelle 34: Routing-Eintrag für Modbus Master 01

**Schritt 2:** Einträge in die Routing-Tabellen des Modbus Master 02 mit Gateway:

- ❑ Öffnen Sie das Control Panel der Ressource von Modbus Master 02.
- ❑ Öffnen Sie das Register „IP Settings“.
- ❑ Wählen Sie *Hinzufügen* um das Popup-Fenster „Routing-Eintrag hinzufügen“ zu öffnen.
- ❑ Tragen Sie die Parameter aus der folgenden Tabelle in die Routing-Tabelle ein.

Routing-Tabelle von Modbus Master 02 Gateway		
Route IP	Route Subnet Mask	Gateway/IF
192.168.2.227	255.255.255.255	0.0.0.1

Tabelle 35: Routing-Eintrag für Modbus Master 02 (mit Gateway Funktion)

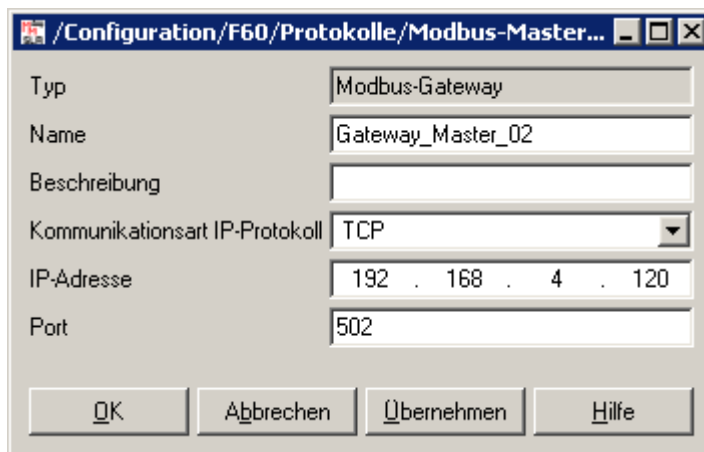
## 4.2 Konfiguration des Modbus Master 01

Im Modbus Master 01 wird der Modbus Gateway angelegt. In diesem werden die Parameter und die IP-Adresse für die Kommunikation über den Modbus Master 02 (mit Gateway Funktion) eingetragen.

Im Verzeichnis von „Modbus Gateway“ werden alle Gateway Slaves eingetragen.

**Schritt 1:** Konfigurieren Sie die Kommunikation zum Modbus Gateway:

- ❑ Öffnen Sie den Strukturbaum der Ressource von Modbus Master 01.
- ❑ Wählen Sie *Modbus Master->Neu->Modbus Gateway*.
- ❑ Öffnen Sie im Kontextmenü *Modbus Gateway->Eigenschaften* das Dialogfenster „Eigenschaften“.
- ❑ Übernehmen Sie die Parameter aus Bild 26.



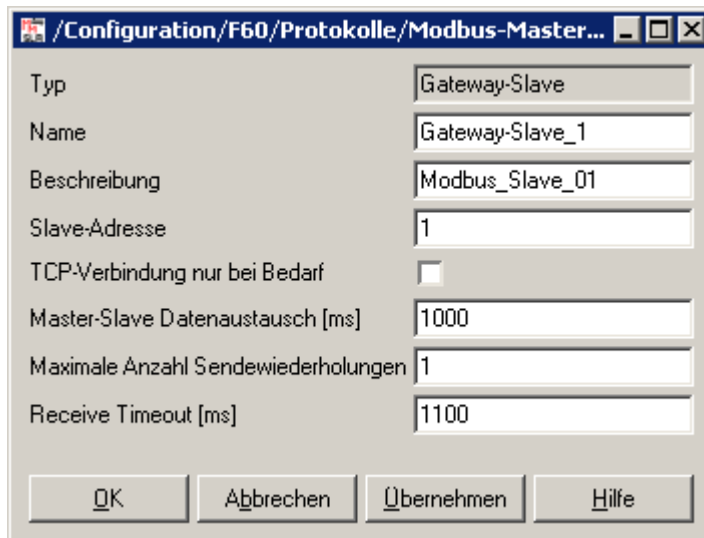
The image shows a configuration window titled "/Configuration/F60/Protokolle/Modbus-Master...". It contains several input fields and a dropdown menu. The 'Typ' field is set to 'Modbus-Gateway'. The 'Name' field contains 'Gateway\_Master\_02'. The 'Beschreibung' field is empty. The 'Kommunikationsart IP-Protokoll' dropdown menu is set to 'TCP'. The 'IP-Adresse' field is filled with '192 . 168 . 4 . 120'. The 'Port' field is filled with '502'. At the bottom of the window, there are four buttons: 'OK', 'Abbrechen', 'Übernehmen', and 'Hilfe'.

**Bild 26: IP-Adresse des Modbus Master 02 (Gateway) eintragen**

**Schritt 2:** Konfigurieren Sie den Gateway Slave im Master 01:

- ❑ Wählen Sie *Modbus Gateway->Neu->Gateway Slave*.
- ❑ Öffnen Sie im Kontextmenü *Gateway Slave->Eigenschaften* das Dialogfenster „Eigenschaften“.
- ❑ Übernehmen Sie die Parameter aus Bild 27

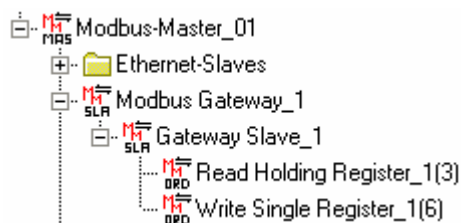
**Hinweis:** Die Receive Timeout [ms] der Modbus Slaves muss im Modbus Master höher eingestellt werden als im Modbus Master (Gateway).



**Bild 27: Serielle Adresse von Modbus Slave 01 eintragen**

**Schritt 3:** Erstellen Sie die Anforderungstelegramme:

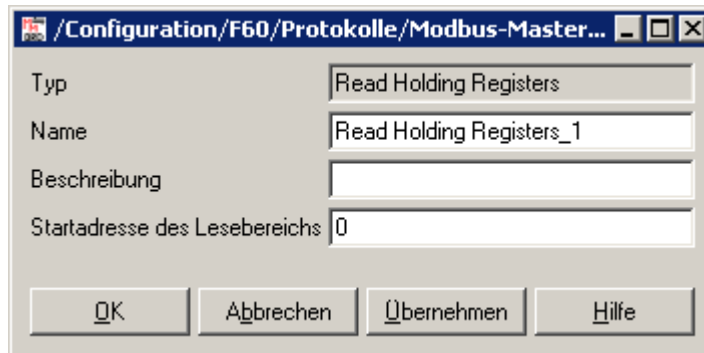
- ❑ Wählen Sie *Modbus Gateway->Gateway Slave->Neu*.
- ❑ Wählen Sie im Kontextmenü *Gateway Slave->Neu* die Anforderungstelegramme:
  - ❑ Ein „Read Holding Register (3) und
  - ❑ Ein „Write Single Register (6).



**Bild 28: Zwei Anforderungstelegramme anlegen**

**Schritt 4:** Konfigurieren Sie die Anforderungstelegramme:

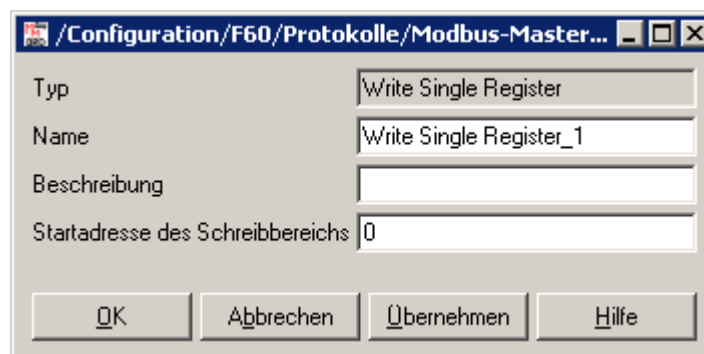
- ❑ Wählen Sie  
*Modbus Gateway->Gateway Slave-> Read Holding Register (3).*
- ❑ Öffnen Sie mit der Menüfunktion *Eigenschaften* im Kontextmenü das Dialogfenster „Eigenschaften“.
- ❑ Geben Sie für die Startadresse des Lesebereichs „0“ ein.



**Bild 29: Konfiguration des Anforderungstelegramm „Read Holding Register (03)“**

**Schritt 5:** Konfigurieren Sie die Anforderungstelegramme:

- ❑ Wählen Sie  
*Modbus Gateway->Gateway Slave-> Write Single Register (6).*
- ❑ Öffnen Sie mit der Menüfunktion *Eigenschaften* im Kontextmenü das Dialogfenster „Eigenschaften“.
- ❑ Geben Sie die für die Startadresse des Schreibbereichs „0“ ein.



**Bild 30: Konfiguration des Anforderungstelegramm „Write Single Register (06)“**

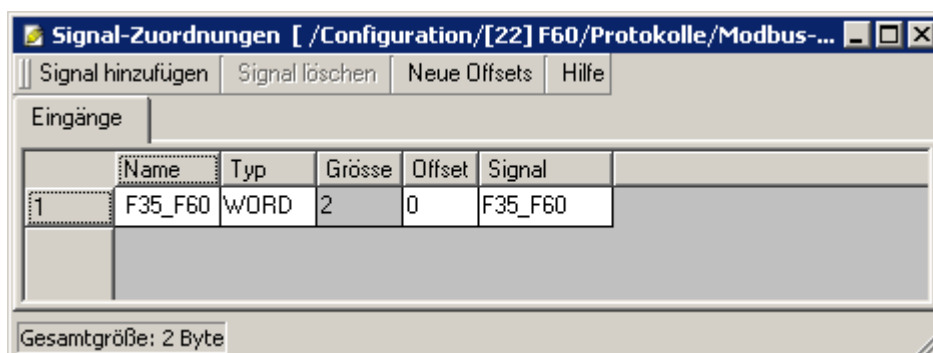


**Schritt 6:** Erstellen Sie die folgenden Signale im Signaleditor des Masters:

- ❑ Öffnen Sie im Hardware-Management den Signaleditor mit *Signale->Editor*.
- ❑ Erstellen Sie das Signal „F35\_F60“ vom Typ „WORD“. Dieses Signal wird als Eingangssignal für den Modbus Master verwendet.
- ❑ Erstellen Sie das Signal „F60\_F35“ vom Typ „WORD“. Dieses Signal wird als Ausgangssignal für den Modbus Master verwendet.

**Schritt 7:** Konfigurieren Sie die Eingangssignale:

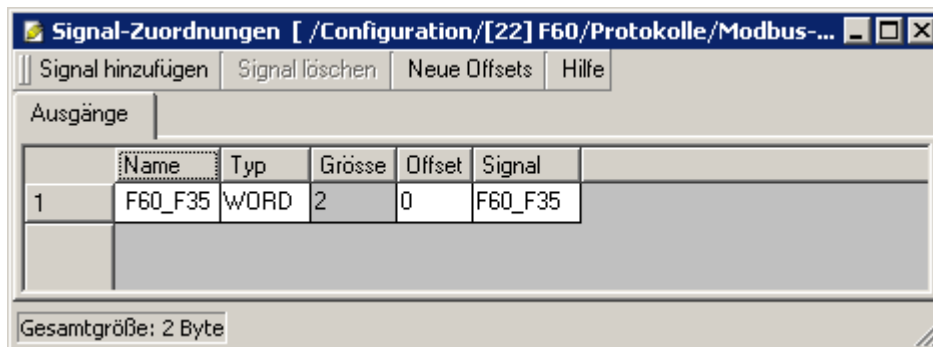
- ❑ Wählen Sie *Modbus Gateway->Gateway Slave-> Read Holding Register (3)*.
- ❑ Öffnen Sie mit *Signale verbinden* im Kontextmenü das Dialogfenster „Signal-Zuordnungen“.
- ❑ Öffnen Sie über das Hauptmenü *Signale -> Editor* den Signaleditor.
- ❑ Klicken Sie im Signaleditor auf den „Namen“ des Signals „F35\_F60“ und ziehen Sie den Signalnamen per Drag & Drop auf das Eingangssignal im Register *Eingang* des Dialogfensters "Signal-Zuordnungen".
- ❑ Klicken Sie im Dialogfenster „Signal-Zuordnungen“ auf die Schaltfläche *Neue Offsets*.
- ❑ Klicken Sie im Popup-Fenster „Offsets nummerieren“ auf die Schaltfläche *Nummerieren*.
- ❑ Schließen Sie das Dialogfenster.



**Bild 31:** Dialogfenster „Signale Zuordnungen“ des „Read Holding Register (03)“

**Schritt 8:** Konfigurieren Sie die Ausgangssignale:

- ❑ Wählen Sie  
*Modbus Gateway->Gateway Slave-> Write Single Register (6).*
- ❑ Öffnen Sie mit *Signale verbinden* im Kontextmenü das Dialogfenster „Signal-Zuordnungen“.
- ❑ Öffnen Sie über das Hauptmenü *Signale -> Editor* den Signaleditor.
- ❑ Klicken Sie im Signaleditor auf den „Namen“ des Signals „F60\_F35“ und ziehen Sie den Signalnamen per Drag & Drop auf das Ausgangssignal im Register *Ausgang* des Dialogfensters " Signal-Zuordnungen ".
- ❑ Klicken Sie im Dialogfenster „Signal-Zuordnungen“ auf die Schaltfläche *Neue Offsets*.
- ❑ Klicken Sie im Popup-Fenster „Offsets nummerieren“ auf die Schaltfläche *Nummerieren*.
- ❑ Schließen Sie das Dialogfenster.



**Bild 32:** Dialogfenster „Signal-Zuordnungen“ des „Write Single Register (06)“

**Schritt 9:** Laden Sie den Code in die Ressource:

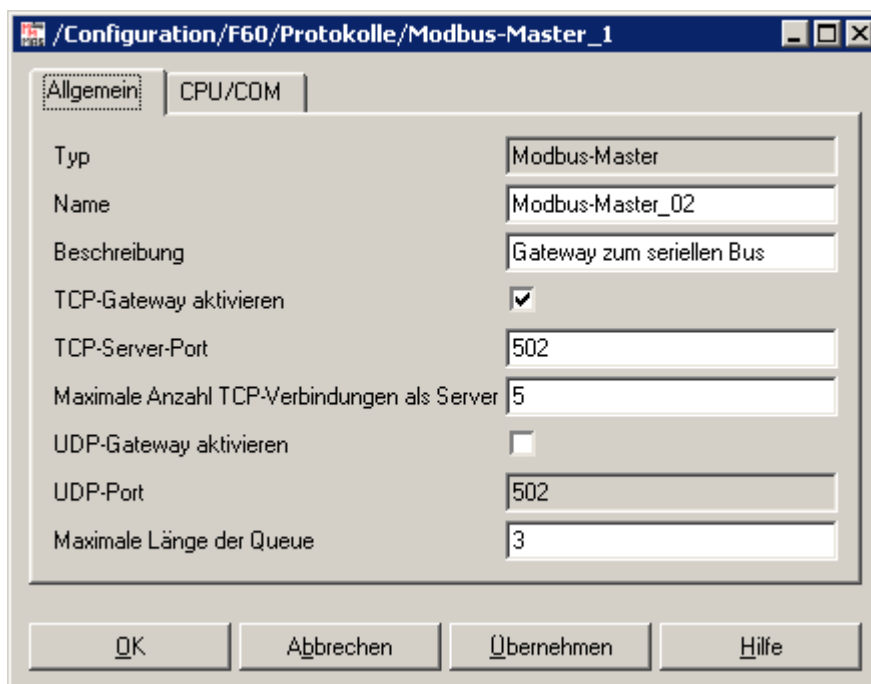
- ❑ Starten Sie den Code Generator für die Ressource.
- ❑ Stellen Sie sicher, dass der Code fehlerfrei generiert wurde (siehe Fehler-Status-Anzeige).
- ❑ Laden Sie den Code in die Ressource.

### 4.3 Konfiguration des Modbus Master 02

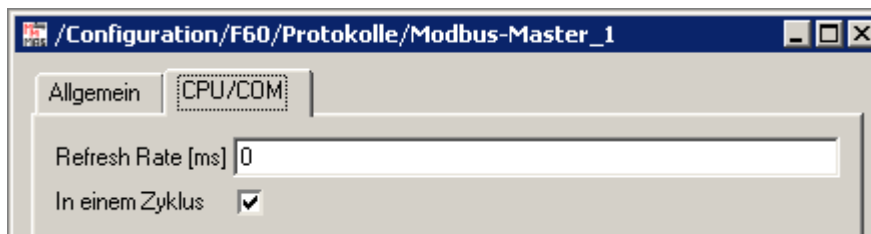
In den Eigenschaften des Modbus Master 02 muss die Gateway Funktion aktiviert werden. Damit werden die im Master 01 konfigurierten Gateway Slaves mit den seriellen Slaves verbunden.

**Schritt 1:** Konfigurieren Sie im Modbus Master 02 das TCP Gateway:

- Öffnen Sie den Strukturbaum der Ressource von Modbus Master 02
- Wählen Sie *Modbus Master->Eigenschaften*.
- Aktivieren Sie das TCP-Gateway und übernehmen Sie die Parameter aus Bild 33 und Bild 34.



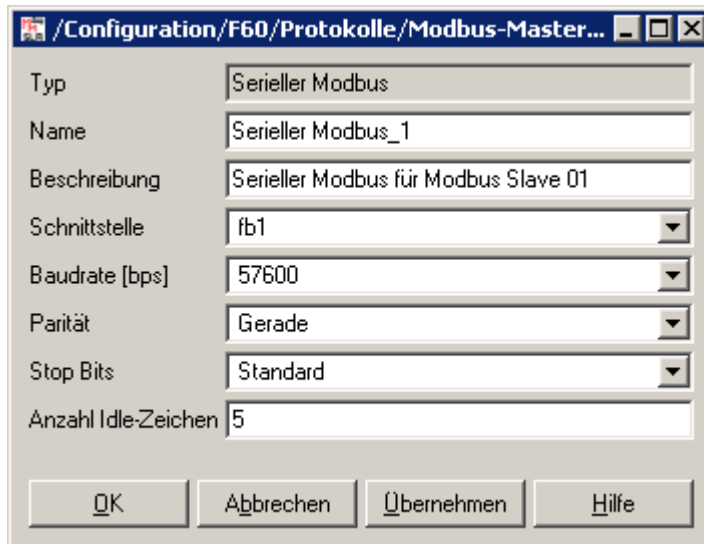
**Bild 33: Register „Allgemein“ im Dialogfenster „Modbus Master 02“**



**Bild 34: Register „CPU/COM“ im Dialogfenster „Modbus Master 02“**

**Schritt 2:** Konfigurieren Sie im Modbus Master 02 den seriellen Modbus:

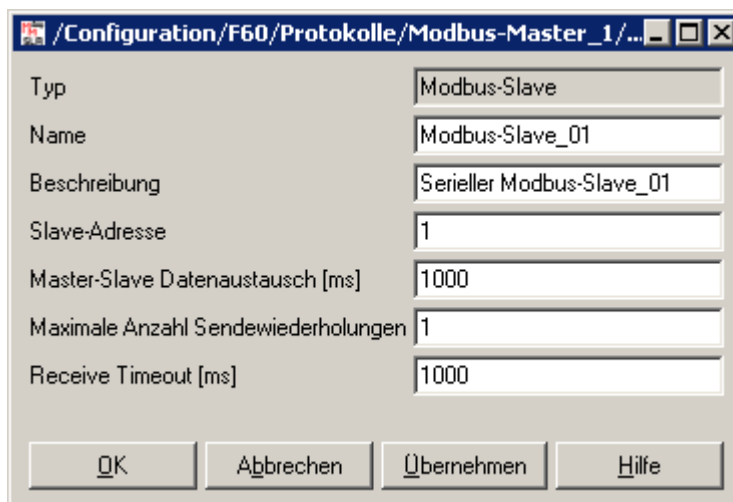
- ❑ Öffnen Sie den Strukturbaum der Ressource von Modbus Master 02
- ❑ Wählen Sie *Serieller Modbus->Eigenschaften*.
- ❑ Übernehmen Sie die Parameter aus Bild 35.



**Bild 35: Dialogfenster „Serieller Modbus 01“**

**Schritt 3:** Konfigurieren Sie den Modbus Slave:

- ❑ Wählen Sie *Serieller Modbus->Modbus Slave->Eigenschaften*
- ❑ Übernehmen Sie die Parameter aus Bild 36.



**Bild 36: Dialogfenster „Modbus Slave“**

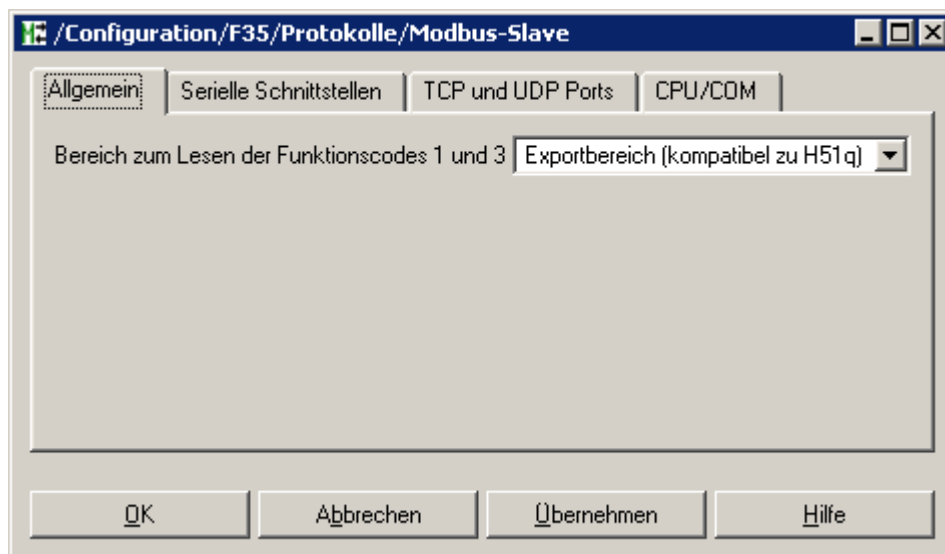
**Schritt 4:** Laden Sie den Code in die Ressource:

- ❑ Starten Sie den Code Generator für die Ressource.
- ❑ Stellen Sie sicher, dass der Code fehlerfrei generiert wurde (siehe Fehler-Status-Anzeige).
- ❑ Laden Sie den Code in die Ressource.

#### 4.4 Konfiguration des Modbus Slave 01

**Schritt 1:** Konfigurieren Sie den seriellen Modbus Slave 01:

- ❑ Öffnen Sie den Strukturbaum der Ressource von Modbus Slave 01.
- ❑ Wählen Sie *Protokolle->Modbus Slave*.
- ❑ Öffnen Sie im Kontextmenü *Modbus Slave->Eigenschaften* das Dialogfenster „Eigenschaften“.
- ❑ Wählen Sie das Register „Allgemein“.
- ❑ Übernehmen Sie die Parameter aus Bild 37.



**Bild 37:** Dialogfenster „Allgemein“

---

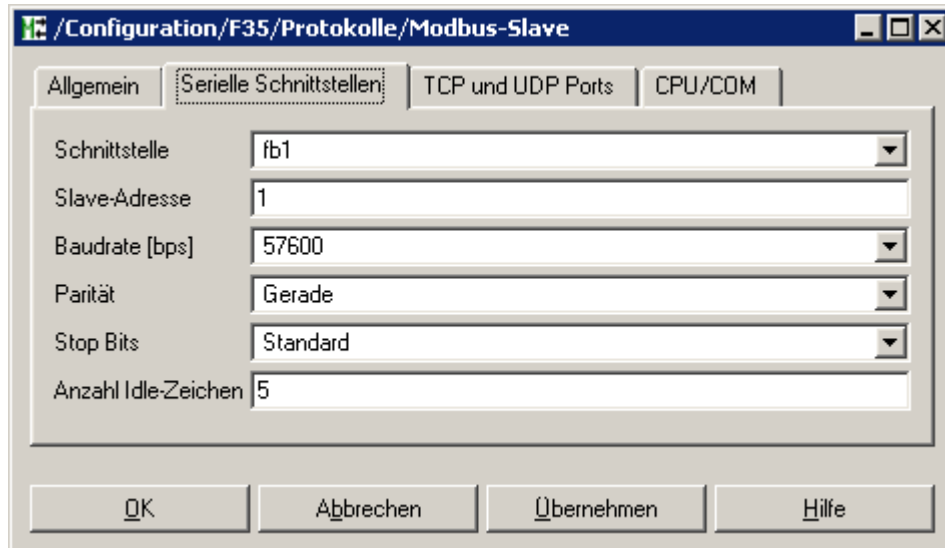
**Hinweis:** Für die *HIMatrix*-Steuerungen mit der COM BS Versionen V.3 bis V.6 gelten die folgenden Lesebereiche für die Funktionscodes 1 und 3:

COM BS V.3:	Importbereich
COM BS V.4:	Exportbereich (kompatibel zu H51q)
COM BS V.6:	Konfigurierbar

---

**Schritt 2:** Konfigurieren Sie die serielle Schnittstelle des Modbus Slave:

- ❑ Wählen Sie das Register „Serielle Schnittstellen“.
- ❑ Übernehmen Sie die Parameter aus Bild 38.



**Bild 38: Dialogfenster „Serielle Schnittstellen“**

**Schritt 3:** Erstellen Sie die folgenden Signale im Signaleditor des Modbus Slave:

- ❑ Öffnen Sie im Hardware-Management den Signaleditor mit *Signale->Editor*.
- ❑ Erstellen Sie das Signal „F35\_F60“ vom Typ „WORD“. Dieses Signal wird als Ausgangssignal für den Modbus Slave verwendet.
- ❑ Erstellen Sie das Signal „F60\_F35“ vom Typ „WORD“. Dieses Signal wird als Eingangssignal für den Modbus Slave verwendet.

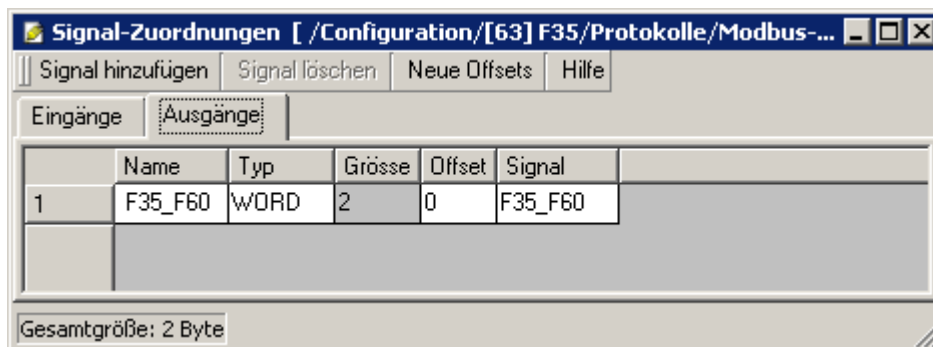
---

**Hinweis:** Wurde der Modbus Master in der gleichen Konfiguration angelegt, dann sind diese Signale bereits im Signaleditor vorhanden und können auch für den Modbus Slave verwendet werden.

---

**Schritt 4:** Verbinden Sie das Ausgangssignal des Modbus Slave:

- ❑ Öffnen Sie mit *Signale verbinden* im Kontextmenü des Modbus Slave das Dialogfenster „Signal-Zuordnungen“.
- ❑ Wählen Sie das Register „Ausgang“ des Dialogfensters „Signale Zuordnen“.
- ❑ Öffnen Sie über das Hauptmenü *Signale -> Editor* den Signaleditor.
- ❑ Klicken Sie im Signaleditor auf den „Namen“ des Signals „F35\_F60“ und ziehen Sie den Signalnamen per Drag & Drop auf das Ausgangssignal im Register „Ausgang“ des Dialogfensters „Signal-Zuordnungen“.
- ❑ Klicken Sie im Dialogfenster „Signal-Zuordnungen“ auf die Schaltfläche *Neue Offsets*.
- ❑ Klicken Sie im Dialogfenster „Offsets nummerieren“ auf die Schaltfläche *Nummerieren*.
- ❑ Schließen Sie das Dialogfenster.

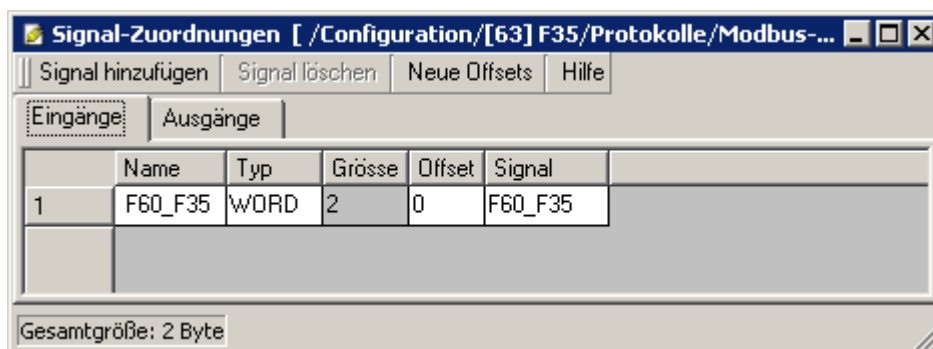


**Bild 39:** Register „Ausgänge“ des Dialogfenster „Signal-Zuordnungen“



**Schritt 5:** Verbinden Sie das Eingangssignal des Modbus Slave:

- ❑ Öffnen Sie mit *Signale verbinden* im Kontextmenü des Modbus Slave das Dialogfenster „Signal-Zuordnungen“.
- ❑ Wählen Sie das Register „Eingang“ des Dialogfensters „Signal-Zuordnungen“.
- ❑ Öffnen Sie über das Hauptmenü *Signale -> Editor* den Signaleditor.
- ❑ Klicken Sie im Signaleditor auf den „Namen“ des Signals „F60\_F35“ und ziehen Sie den Signalnamen per Drag & Drop auf das Eingangssignal im Register „Eingang“ des Dialogfensters „Signal-Zuordnungen“.
- ❑ Klicken Sie im Dialogfenster „Signal Zuordnen“ auf die Schaltfläche *Neue Offsets*.
- ❑ Klicken Sie im Dialogfenster „Offsets nummerieren“ auf die Schaltfläche *Nummerieren*.
- ❑ Schließen Sie das Dialogfenster.



**Bild 40:** Register „Eingänge“ des Dialogfenster „Signal-Zuordnungen“

**Schritt 6:** Laden Sie den Code in die Ressource:

- ❑ Starten Sie den Code Generator für die Ressource.
- ❑ Stellen Sie sicher, dass der Code fehlerfrei generiert wurde (siehe Fehler-Status-Anzeige).
- ❑ Laden Sie den Code in die Ressource.

**HIMA**  
**...die sichere Entscheidung.**



HIMA Paul Hildebrandt GmbH  
Industrie-Automatisierung  
Postfach 1261 • 68777 Brühl

Telefon: (06202) 709-0 • Telefax: (06202) 709-107  
E-mail: [info@hima.com](mailto:info@hima.com) • Internet: [www.hima.de](http://www.hima.de)

(0908)