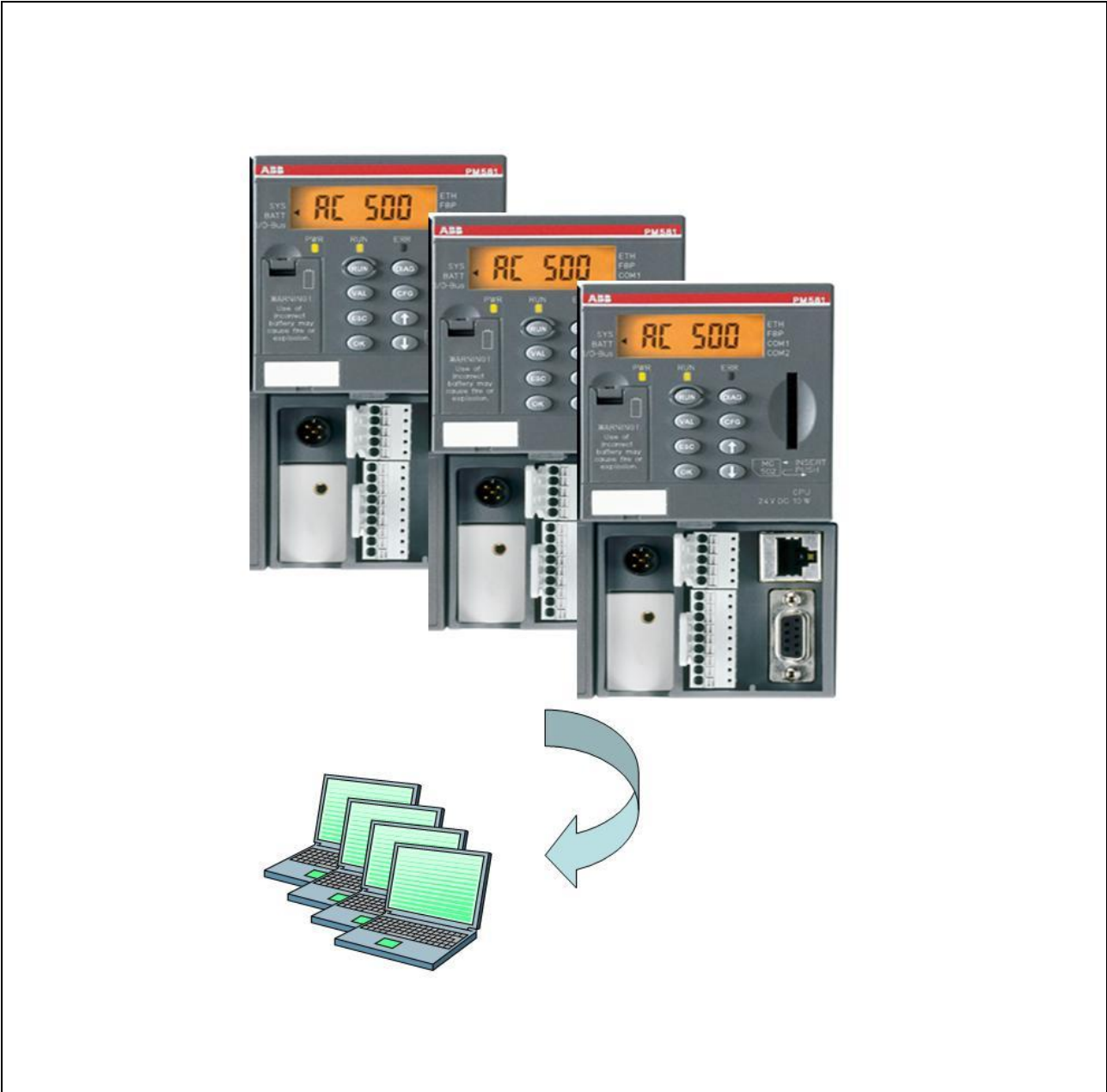


Scalable PLC  
for Individual Automation

AC500 Modbus RTU  
Communication - Multiple Jobs



# Content

<b>1</b>	<b>Disclaimer .....</b>	<b>2</b>
1.1	For customers domiciled outside Germany/ Für Kunden mit Sitz außerhalb Deutschlands.....	2
1.2	Nur für Kunden mit Sitz in Deutschland .....	2
<b>2</b>	<b>AC500 Modbus programming and testing .....</b>	<b>3</b>
2.1	Used PC software and Hardware.....	3
2.2	RS485 Connection.....	4
<b>3</b>	<b>PM581 Modbus RTU master – PC as Modbus RTU slave – multiple jobs .....</b>	<b>5</b>
3.1	PLC program .....	5
3.2	COM_MOD_MAST .....	5
3.3	Step Chain .....	5
3.4	Global variables.....	6
<b>4</b>	<b>PC modbus slave .....</b>	<b>7</b>
4.1	Data write.....	7
4.2	Data read.....	7
4.3	Data in the PLC.....	8
<b>5</b>	<b>Usefull tips .....</b>	<b>9</b>
5.1	Analyse the request telegrams sent by the AC500 CPU.....	9
5.2	Analyse the request telegrams sent by the CP400 operator panel.....	10
5.2.1	Serial line settings.....	11
5.2.2	Request telegrams sent by the CP400 master.....	11
<b>6</b>	<b>Related documents .....</b>	<b>13</b>

# 1 Disclaimer

## 1.1 For customers domiciled outside Germany/ Für Kunden mit Sitz außerhalb Deutschlands

### "Warranty, Liability:

The user shall be solely responsible for the use of this application example described within this file.

ABB shall be under no warranty whatsoever. ABB's liability in connection with this application example or the files included within this file, irrespective of the legal ground, shall be excluded. The exclusion of liability shall not apply in the case of intention or gross negligence. The present declaration shall be governed by and construed in accordance with the laws of Switzerland under exclusion of its conflict of laws rules and of the Vienna Convention on the International Sale of Goods (CISG)."

### "Gewährleistung und Haftung:

Der Nutzer ist allein für die Verwendung des in dieser Datei beschriebenen Anwendungsbeispiels verantwortlich.

ABB unterliegt keiner Gewährleistung. Die Haftung von ABB im Zusammenhang mit diesem Anwendungsbeispiel oder den in dieser Datei enthaltenen Dateien - gleich aus welchem Rechtsgrund - ist ausgeschlossen. Dieser Ausschluß gilt nicht im Falle von Vorsatz oder grober Fahrlässigkeit. Diese Erklärung unterliegt Schweizer Recht unter Ausschluß der Verweisungsnormen und des UN-Kaufrechts (CISG)."

## 1.2 Nur für Kunden mit Sitz in Deutschland

### "Gewährleistung und Haftung:

Die in diesem Anwendungsbeispiel enthaltenen Dateien beschreiben eine mögliche Anwendung der Steuerung AC500 bzw. zeigen eine mögliche Einsatzart der Steuerung. Sie stellen nur Beispiele für Programmierungen dar, sind aber keine fertigen Lösungen. Eine Gewähr kann nicht übernommen werden.

Der Nutzer ist für die ordnungsgemäße, insbesondere vollständige und fehlerfreie Programmierung der Steuerungen selbst verantwortlich. Im Falle der teilweisen oder ganzen Übernahme der Programmierbeispiele können gegen ABB keine Ansprüche geltend gemacht werden.

Die Haftung von ABB, gleich aus welchem Rechtsgrund, im Zusammenhang mit den Anwendungsbeispielen oder den in dieser Datei enthaltenen Dateien wird ausgeschlossen. Der Haftungsausschluß gilt jedoch nicht in Fällen des Vorsatzes, der groben Fahrlässigkeit, bei Ansprüchen nach dem Produkthaftungsgesetz, im Falle der Verletzung des Lebens, des Körpers oder der Gesundheit oder bei schuldhafter Verletzung einer wesentlichen Vertragspflicht. Im Falle der Verletzung einer wesentlichen Vertragspflicht ist die Haftung jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht zugleich ein anderer der in Satz 2 dieses Unterabsatzes erwähnten Fälle gegeben ist. Eine Änderung der Beweislast zum Nachteil des Nutzers ist hiermit nicht verbunden.

Es gilt materielles deutsches Recht unter Ausschluß des UN-Kaufrechts."

## 2 AC500 Modbus programming and testing

This document explains how to configure/program a PM581 CPU as Modbus master or modbus slave and how to check the basic modbus functionality with a PC modbus software.  
The modbus communication is RTU.

### 2.1 Used PC software and Hardware

<http://www.win-tech.com/>

Modbus Master PC software

[ModScan32.zip \(851K\)](#)  
Updated to Version 8.A00-03  
(Dec. 28, 2010)

Modbus slave PC software

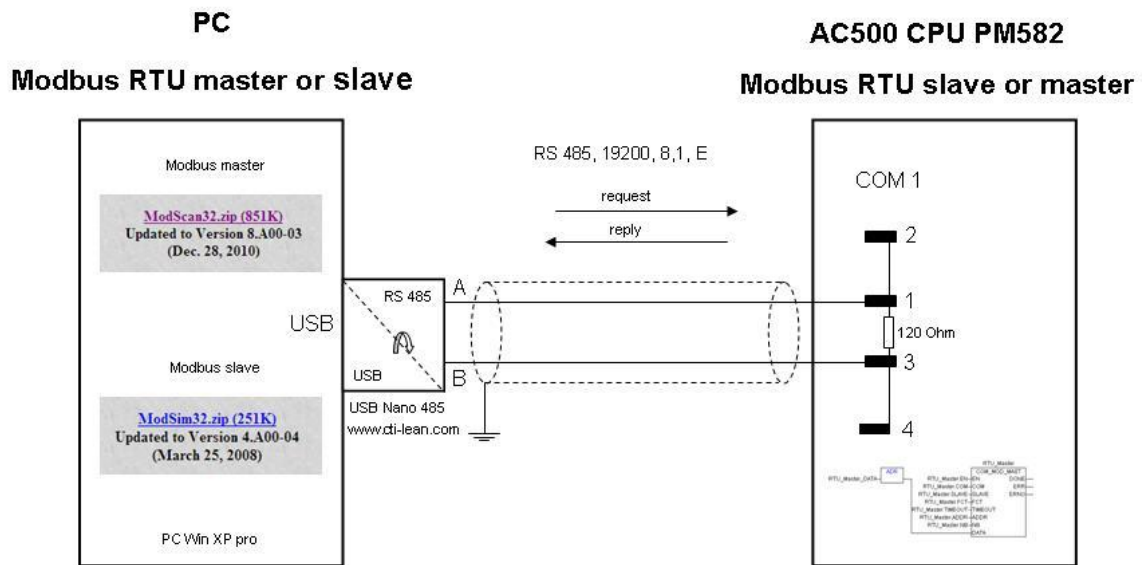
[ModSim32.zip \(251K\)](#)  
Updated to Version 4.A00-04  
(March 25, 2008)

USB – RS485 convertor: USB Nano from [www.cti-lean.com](http://www.cti-lean.com)

AC500 Programming software



## 2.2 RS485 Connection



PC as Modbus RTU Master - AC500 CPU PM582 as Modbus RTU slave or  
 AC500 CPU PM582 as Modbus RTU Master - PC as Modbus RTU slave

The USB Nano was installed as COM6 in PC



# 3 PM581 Modbus RTU master – PC as Modbus RTU slave – multiple jobs

## 3.1 PLC program

Example 2

Modbus data exchange over a ST step chain: write one word to the slave, read one word from slave.

## 3.2 COM\_MOD\_MAST

The program structure and call of the COM\_MOD\_MAST FB remain the same as in the example 1 (see Application Example “AC500 Modbus RTU Communication – One Job”), but

## 3.3 Step Chain

the number of steps in the chain increasing.

(\*step chain\*)

### CASE Chain OF

#### 0: (\*send request write one word to address 100\*)

```
IF NOT RTU_Master.DONE THEN (*COM_MOD_MAST has finished the previous job *)
RTU_Master.EN:=TRUE; (*set EN to TRUE *)
RTU_Master.COM:=1; (* send request over COM1 *)
RTU_Master.SLAVE:=3; (* slave number of the PC is 3 *)
RTU_Master.FCT:=16; (* FCT = write 1 word*)
RTU_Master.TIMEOUT:=1000; (* set TIMEOUT to 1000 ms*)
RTU_Master.ADDR:=100; (* register address in slave*)
RTU_Master.NB:=1; (* 1 word *)
RTU_Master_DATA:=WORD_WRITE;(* send WORD_WRITE*)
Chain:=10; (*next step in the chain*)
END_IF
```

#### 10: (\*reply from slave to write one word\*)

```
IF RTU_Master.DONE THEN (* reply telegram from slave is arrived*)
RTU_Master.EN:=FALSE; (* set EN to FALSE*)
Chain:=20; (* next step in the chain*)
END_IF
```

#### 20: (\*send request: read one word from address 101\*)

```
IF NOT RTU_Master.DONE THEN (*COM_MOD_MAST has finished the previous job *)
RTU_Master.EN:=TRUE; (*set EN to TRUE *)
RTU_Master.COM:=1; (* send request over COM1 *)
RTU_Master.SLAVE:=3; (* slave number of the PC is 3 *)
RTU_Master.FCT:=4; (* FCT = read 1 word*)
RTU_Master.TIMEOUT:=1000; (* set TIMEOUT to 1000 ms*)
RTU_Master.ADDR:=101; (* register address in slave*)
RTU_Master.NB:=1; (* 1 word *)
```

```
Chain:=30; (*next step in the chain*)  
END_IF
```

```
30: (*reply from slave to read one word*)
```

```
IF RTU_Master.DONE THEN (* reply telegram from slave is arrived*)  
WORD_READ:=RTU_Master_DATA; (*store read dat into WORD_READ*)  
RTU_master.EN:=FALSE; (* set EN to FALSE*)  
Chain:=0; (* next step in the chain*)  
END_IF  
END_CASE
```

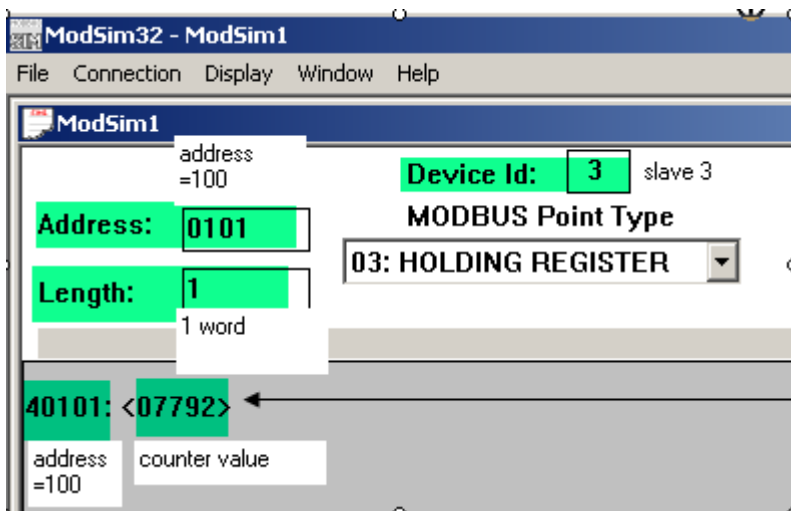
## 3.4 Global variables

```
(WORD_READ: WORD;)
```

## 4 PC modbus slave

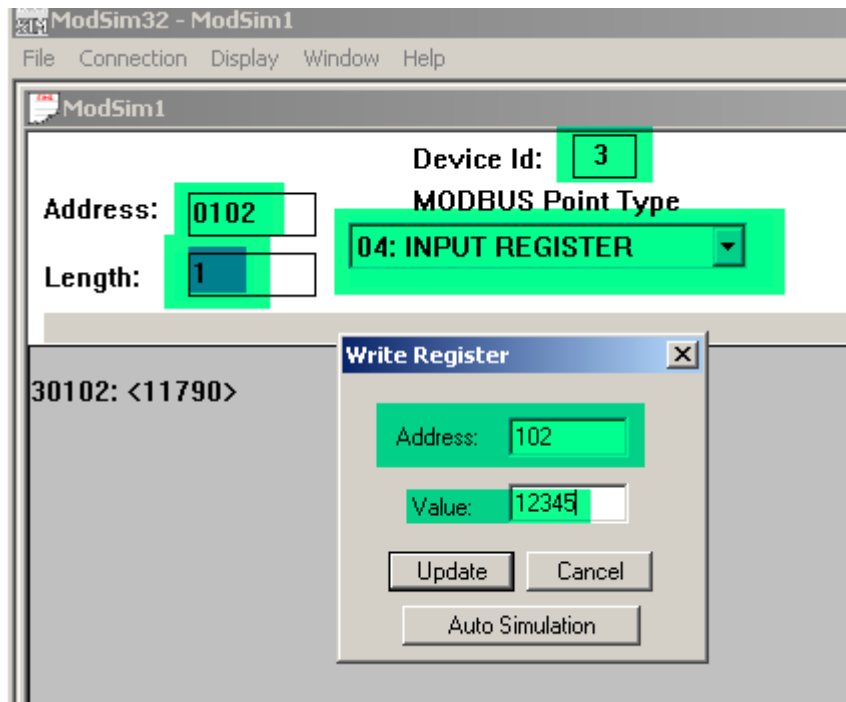
### 4.1 Data write

The data exchange can be observed in the PC modbus slave.  
The PLC master write data into the address 100:



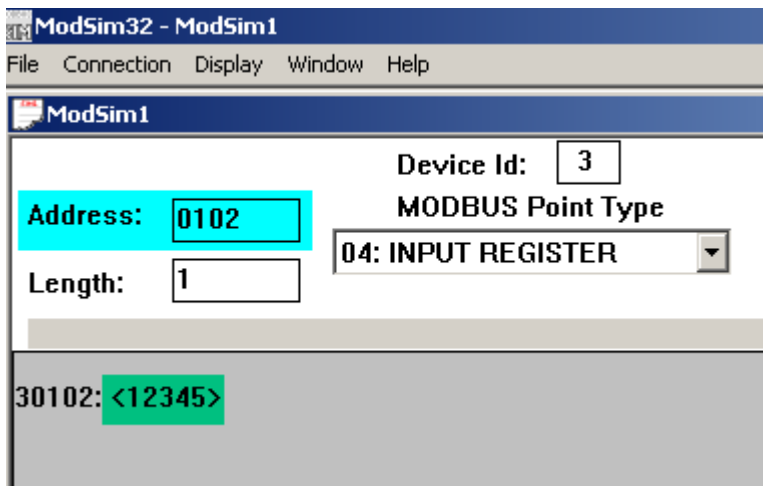
### 4.2 Data read

The PLC master is reading data from the address 101: input a variable to be read from address 102.



The value is=12345.





## 4.3 Data in the PLC

The value 12345 was read into the PLC

```

IF RTU_Master.DONE THEN
WORD_READ:=RTU_Master_DATA;
RTU_master.EN:=FALSE;
Chain:=0;
END_IF

END_CASE

*error handling*
F RTU_Master.ERR THEN
modbus_Error_NB:=RTU_Master.ERNO;
END_IF

```

```

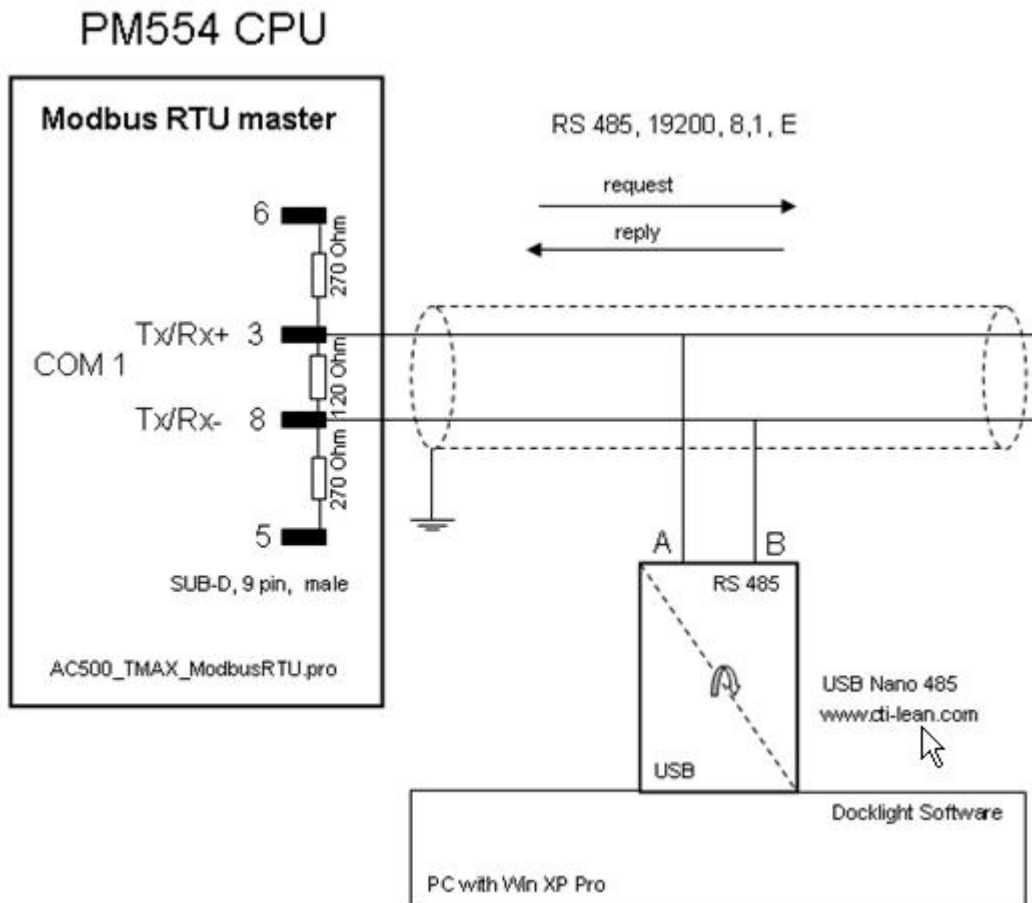
Chain = 20
RTU_Master.DONE = FALSE
WORD_READ = 12345
RTU_Master.EN = FALSE
Chain = 20
RTU_Master_DATA = 36006

```

## 5 Usefull tips

### 5.1 Analyse the request telegrams sent by the AC500 CPU

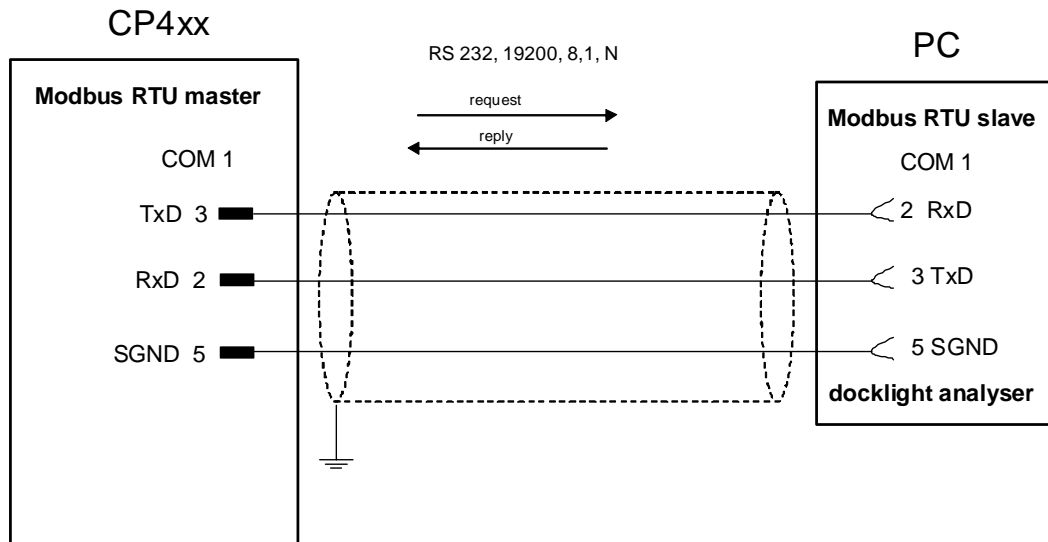
Disconnect the slave, you will get on the analyser the request telegrams sent by the AC500 CPU.



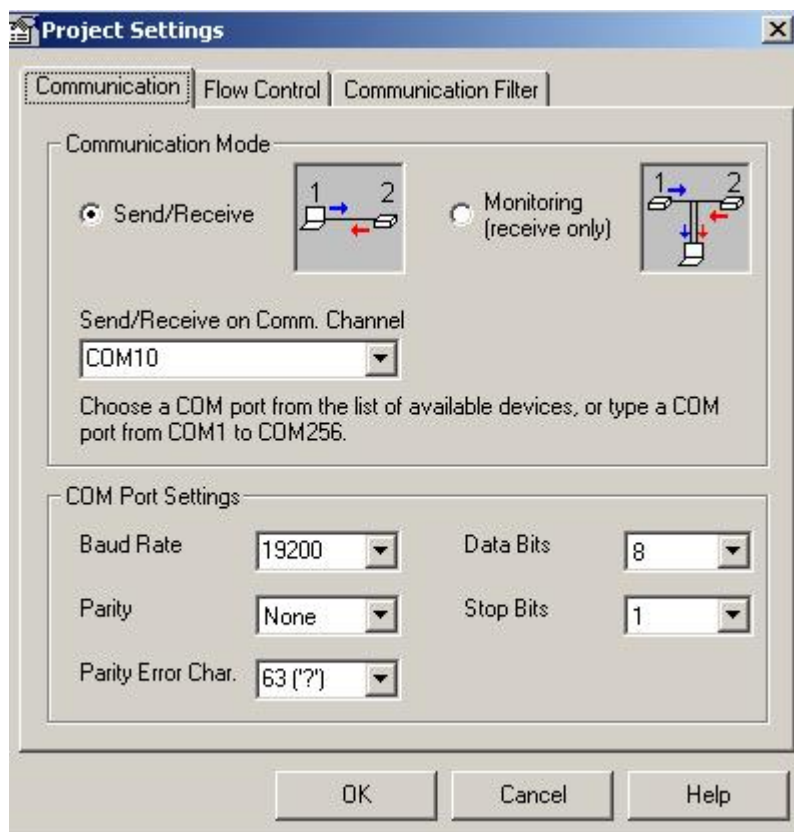
## 5.2 Analyse the request telegrams sent by the CP400 operator panel

The connection cable is the CP programming cable TK501.

Start the analyser software Docklight on the PC set the Baudrate, parity, data bits, start bits, according to the CP4xx configuration.

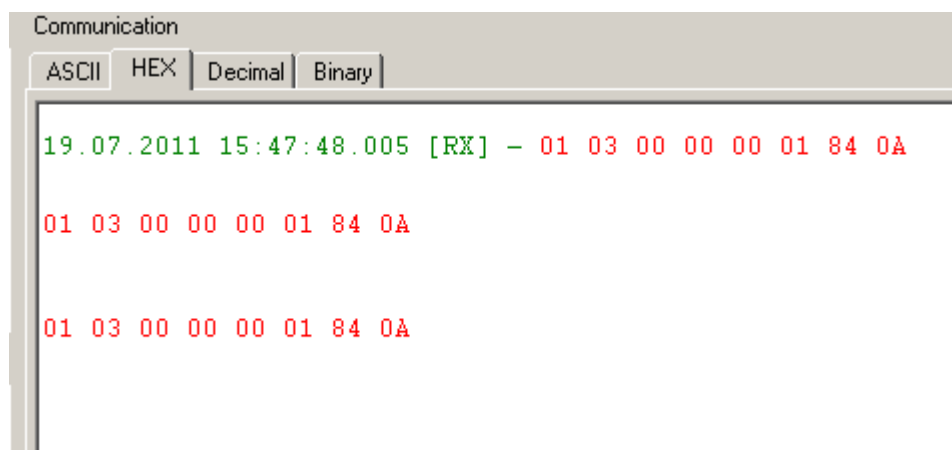


## 5.2.1 Serial line settings



## 5.2.2 Request telegrams sent by the CP400 master

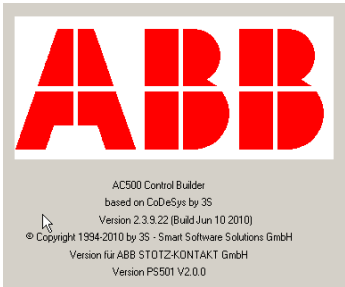
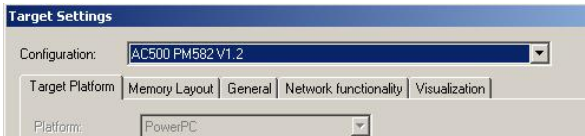
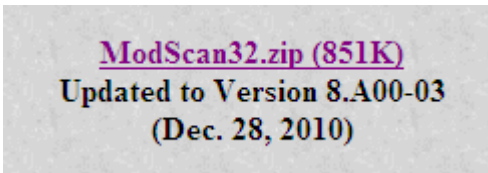
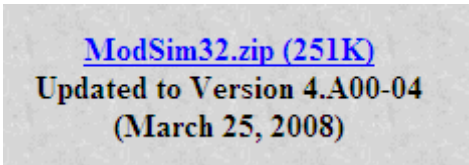


You will get the request telegrams set by the CP.



## Request telegram

01 = slave 1  
03 = FCT – read n words  
00 00 = address 0  
00 01 = one word  
84 0A = CRC

## 6 Related documents

Part	Document	DocNo / Rev
ABB Control Builder	 <p>AC500 Control Builder based on CoDeSys by 3S Version 2.3.9.22 (Build Jun 10 2010) © Copyright 1994-2010 by 3S - Smart Software Solutions GmbH Version für ABB STOTZ-KONTAKT GmbH Version PS501 V2.0.0</p>	
AC500 CPU	 <p>Target Settings Configuration: AC500 PM582 V1.2 Target Platform   Memory Layout   General   Network functionality   Visualization Platform: PowerPC</p>	
WinTech Modbus RTU Software	 <p><b>ModScan32.zip (851K)</b> <b>Updated to Version 8.A00-03</b> <b>(Dec. 28, 2010)</b></p>	Modbus master PC software from <a href="http://www.wintech.com/">http://www.wintech.com/</a>
WinTech Modbus RTU Software	 <p><b>ModSim32.zip (251K)</b> <b>Updated to Version 4.A00-04</b> <b>(March 25, 2008)</b></p>	Modbus slave PC software from <a href="http://www.wintech.com/">http://www.wintech.com/</a>
Modbus protocol description	 <p>Modbus-IDA MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1b</p>	 <a href="http://www.Modbus-IDA.org">http://www.Modbus-IDA.org</a>



<b>Doc. kind:</b> Helpline document	<b>No. o. p.:</b> 15
<b>Title:</b> AC500 Modbus RTU Communication – Multiple Jobs	<b>Lang.:</b> EN



Manual No. 2CDC 125 165 M0201

---

**ABB Automation Products GmbH**  
Wallstadter Str. 59  
D-68526 Ladenburg  
Tel. : +49 62 21 / 701-1444  
Fax : +49 62 21 / 701-1382  
E-Mail : [plc.support@de.abb.com](mailto:plc.support@de.abb.com)

[www.abb.com/plc](http://www.abb.com/plc)