

MODBUS RTU (Zero-based Addressing)

Supported Series: MODBUS RTU CONTROLLER

Website: <http://www.modbus.org>

HMI Setting:

Parameters	Recommended	Options	Notes
PLC type	MODBUS RTU (Zero-based Addressing)		
PLC I/F	RS485	RS232/RS485/USB	
Baud rate	9600	9600~115200	
Data bits	8	7, 8	
Parity	Even	Even, Odd, None	
Stop bits	1	1, 2	
PLC sta. no.	1	0-255	

Online simulator	YES	Broadcast command	YES
Extend address mode	YES		

Interval of block pack (words) : 5
Max. read-command size (words) : 120
Max. write-command size (words) : 120

Buttons: Address Range Limit..., Conversion...

[Address Range Limit]

The address range for device types 0x, 1x, and 0x_multi_coils can be configured. This option should be used when the controller's bit register address range is not a multiple of 16. For example, if the range is 0x1 ~ 0x6 or 0x9 ~ 0x14, communication will not be possible. To ensure effective communication between the HMI and the controller, please follow the configuration instructions below.

0x-1~6

Station no. : 1
Address type : 0x
Min. address : 1 Max. address : 6

* Use 0x_1~0x_32 (1x_1~1x_32) to define multiple different 0x (1x) address ranges.

Buttons: OK, Cancel

0x_1-9~14

0x/1x Address Range

Station no. : 1

Address type : 0x_1

Min. address : 9 Max. address : 14

* Use 0x_1~0x_32 (1x_1~1x_32) to define multiple different 0x (1x) address ranges.

OK Cancel

[Conversion]

The 3x_Double and 4x_Double address types are added. If [ABCD ->CDAB] check box is selected, please select 3x_Double and 4x_Double address types.

PLC Setting:

Communication mode

Modbus RTU protocol

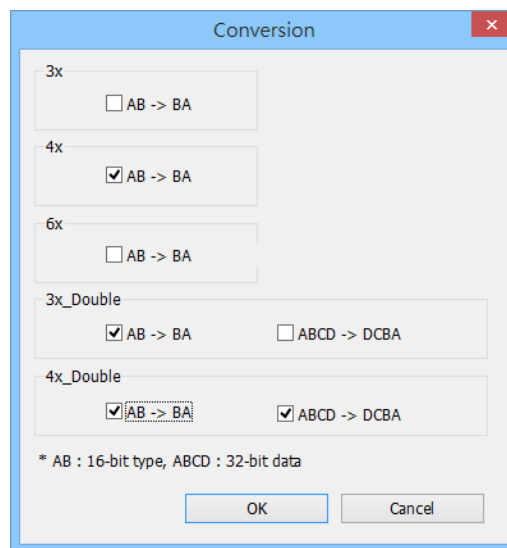
Device Address:

Bit/Word	Device type	Format	Range	Memo
B	0x	DDDDD	0 ~ 65535	Output bit
B	0x_multi_coils	DDDDD	0 ~ 65535	Write multiple coils
B	0x_single_bit	DDDDD	0 ~ 65535	*Note4
B	1x	DDDDD	0 ~ 65535	Input bit (read only)
B	1x_single_bit	DDDDD	0 ~ 65535	*Note4
B	3x_Bit	DDDDDdd	0 ~ 6553515	Input Register bit
B	4x_Bit	DDDDDdd	0 ~ 6553515	Output Register bit
B	6x_Bit	DDDDDdd	0 ~ 6553515	Output Register bit
B	0x_1	DDDDD	0 ~ 65535	
B	0x_2	DDDDD	0 ~ 65535	
B	0x_3	DDDDD	0 ~ 65535	
B	0x_4	DDDDD	0 ~ 65535	
B	0x_5	DDDDD	0 ~ 65535	
B	0x_6	DDDDD	0 ~ 65535	
B	0x_7	DDDDD	0 ~ 65535	
B	0x_8	DDDDD	0 ~ 65535	
B	0x_9	DDDDD	0 ~ 65535	
B	1x_1	DDDDD	0 ~ 65535	
B	1x_2	DDDDD	0 ~ 65535	
B	1x_3	DDDDD	0 ~ 65535	
B	1x_4	DDDDD	0 ~ 65535	
B	1x_5	DDDDD	0 ~ 65535	
B	1x_6	DDDDD	0 ~ 65535	
B	1x_7	DDDDD	0 ~ 65535	
B	1x_8	DDDDD	0 ~ 65535	
B	1x_9	DDDDD	0 ~ 65535	
W	3x	DDDDD	0 ~ 65535	Input Register (read
W	3x_MAX1W	DDDDD	0 ~ 65535	Display 32 bits *Note1
DW	3x_MAX2W	DDDDD	0 ~ 65535	*Note1
W	3x_Double	DDDDD	0 ~ 65535	*Note2
QW	3x_QWord	DDDDD	0 ~ 65535	64-bit
W	4x	DDDDD	0 ~ 65535	Output Register
W	4x_MAX1W	DDDDD	0 ~ 65535	Display 32 bits *Note1
DW	4x_MAX2W	DDDDD	0 ~ 65535	*Note1
W	4x_Double	DDDDD	0 ~ 65535	*Note2

Bit/Word	Device type	Format	Range	Memo
W	4x_32Bit	DDDDD	0 ~ 65535	Output Register *Note1
QW	4x_QWord	DDDDD	0 ~ 65535	64-bit
W	0x_single_coil	DDDDD	0 ~ 65535	*Note3
DW	5x	DDDDD	0 ~ 65535	4x double word swap
W	6x	DDDDD	0 ~ 65535	4x single word write

Note1: MAX1W and 4X_32Bit reads/writes 1 word for each packet and displays a 32-bit value, whereas MAX2W reads/writes 2 words for each packet.

Note2: Go the [System Parameter Settings] -> [Device Properties] and click [Conversion] to set the data format of device types 3x, 4x, 6x, 3x_double, 4x double.



Note3: Read/write a Bit at a time. Value 0 and 1 are used to represent ON or OFF status where 0 means set OFF and 1 means set ON.

Note4: The number of bits to read can be set in Macro; therefore, it is recommended to use 0x address type for functions such as GetData() and SetData(), in order to increase reading speed.

To read/write multiple 0x address type in macro, use 0x_multi_coils to increase write speed.

Note5: Please assign all the addresses to Even addresses, or all to Odd addresses, in order to prevent communication failure.

Note6: EBPro V6.03.02 or later supports 64 bits data type (**cMT Series only**), but please note that the address limit range is 48 bits in maximum..

NOTE:

Address type “5x” is mapped to Holding Register. The communication protocol of 5x is almost the same as “4x” except that “5x” swaps double word.

If 4x contains the following information:

Address	1	2	3	4	5	6	...
Data in word	0x1	0x2	0x3	0x4	0x5	0x6	
Data	0x0201		0x0403		0x0605		

For 5x, it will be:

Address	1	2	3	4	5	6	...
Data in word	0x1	0x2	0x3	0x4	0x5	0x6	
Data	0x0102		0x0304		0x0506		

Modbus RTU function code:

Device type	Read	Write
0x	0x01 Read coil	0x05 write single coil
0x_multi_coils	0x01 Read coil	0x0f write multiple coils
1x	0x02 Read discrete input	X
3x / 3x_Bit	0x04 Read input register	X
4x / 4x_Bit	0x03 Read holding register	0x10 write multiple registers
5x	0x03 Read holding register	0x10 write multiple registers
6x	0x03 Read holding register	0x06 write single register
0x_single_coil	0x01 Read coil	0x05 write single coil

Wiring Diagram:

Diagram 1

RS-232

The serial port pin assignments may vary between HMI models, please click the following link for more information.

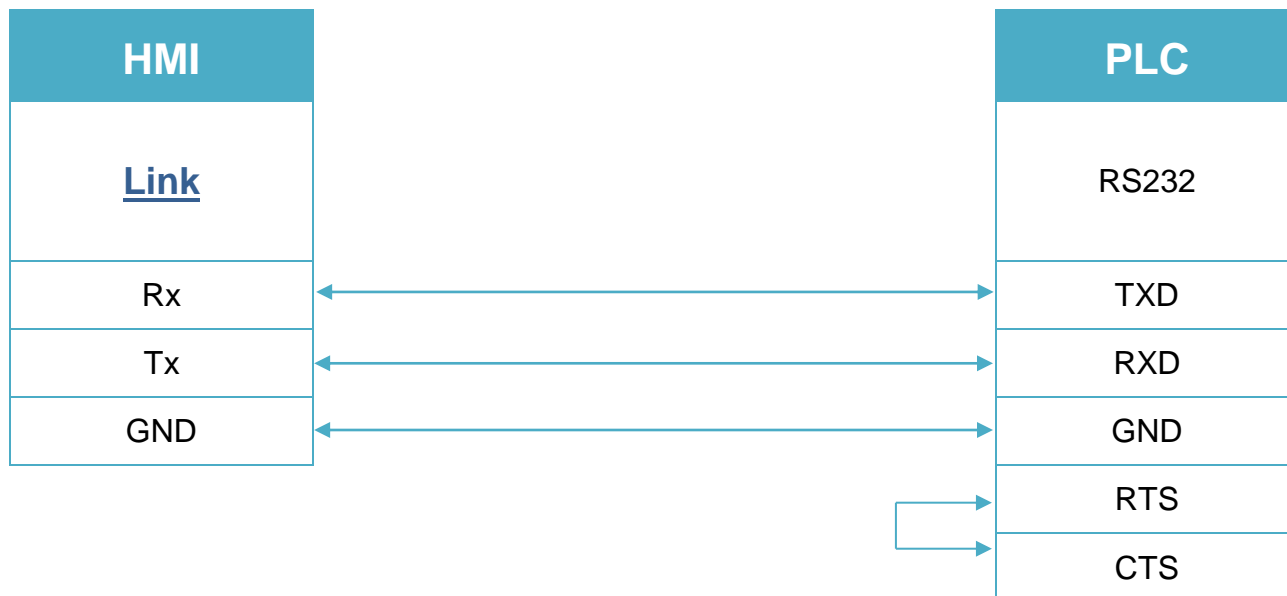


Diagram 2

RS-485 4W

The serial port pin assignments may vary between HMI models, please click the following link for more information.



Diagram 3

RS-485 2W

The serial port pin assignments may vary between HMI models, please click the following link for more information.

