

## GpsNeo converter

Converter GPS time and coordinates to Modbus

Real dimensions 1:1



### Contents:

1. Design
2. Hardware
  - 2.1 Hardware parameters
  - 2.2 Sending parameters to RS485 after RESET
3. Wiring and indication
  - 3.1 Wiring
  - 3.2 LED on PCB
  - 3.3 LED on BOX
4. ModBus RTU communication protocol
  - 4.1 Command 0x03 Read Registers
  - 4.2 Command 0x03 Read Configuration Registers
  - 4.3 Command 0x06 Write Registers
  - 4.4 Default parameters
  - 4.5 Range of addresses
5. Examples for Mous RTU
  - 5.1 Set the communication speed from 115200 Bd to 9600 Bd
  - 5.2 Set the communication speed from 9600 Bd to 115200 Bd
  - 5.3 Read 8 registers from register 100
  - 5.4 Getting the current address with universal address 0xff
  - 5.5 How to set the address
  - 5.6 Changing the address from 1h to 2h
  - 5.7 Reading measured values
6. Used sensors
  - 6.1 uBlox NEO-8M GPS receiver and OEM modul

## 1. Design

Name	Coverage	Comments
GPSNEO X	IP67	Exterior

## 2. Hardware

2.1 Hardware parameters	
Measured parameters	Time from GPS system, Coordinates from GPS system
Calculated parameters	Sun elevation and azimuth
Power	8 - 26V DC/ max 0,5W
Interface	RS485
Hardware	uBlox NEO-8M GPS receiver
Communication speed	9600 or 115200 Bd
Dimensions with Case	Ø40 x 135 mm without holder, Ø40 x 200 mm with holder
Dimensions only PCB	120 x 30 x 10 mm [w x d x h]
Design	Exterior, IP 67
Setup	Via software Bootloader or via ModBus directly

2.2 Sending parameters to RS485 after RESET		
	Parameter	Comments
1.row	112:RESET=4<cr><lf>	112 – address (dec), 4 – com. Protocol

## 3. Wiring, standard cable length: 3 m

Wire color	Comment
Green	Ground
White	12-24V DC
Yellow	RS485 +
Brown	RS485 -

## 4. ModBus RTU communication protocol

4.1 Command 0x03 Read registers			
Register	Register name	Description	Units/Notes
0	GPS Quality Indicator	1 – Locked to GPS, 0 - Internal time	
1	Year	Current year; UTC	
2	Month	Current month; UTC	
3	Day	Current day; UTC	
4	Hour	Current hour; UTC	
5	Minute	Current minute; UTC	
6	Second	Current second; UTC	
7	Sun Elevation	Tenth of Degrees	calculated
8	Sun Azimuth	Tenth of Degrees	calculated
9	Degrees of latitude		
10	Minutes of latitude		
11	Ten thousandths of a minute of latitude		
12	Degrees of longitude		
13	Minutes of longitude		
14	Ten thousandths of a minute of longit.		

<b>15</b>	Altitude	Above sea level in meters	1 m
-----------	----------	---------------------------	-----

**if the value is not available, the value is 0xff00h (65280)**

<b>4.2 Command 0x03 Read configuration registers</b>			
<b>Register</b>	<b>Register name</b>	<b>Description</b>	<b>Units/Notes</b>
<b>100</b>	Address	1 – 247	
<b>101</b>	Communication speed	0 – 115200, 1 – 9600	Bd
<b>102</b>	HWS version 0	Read Only	GP
<b>103</b>	HWS version 1	Read Only	SN
<b>104</b>	HWS version 2	Read Only	E*
<b>105</b>	HWS version 3	Read Only	:1
<b>106</b>	HWS version 4	Read Only	.0
<b>107</b>	Communication protocol		1 ÷ 5

<b>4.3 Command 0x06 Write Registers</b>			
<b>Register</b>	<b>Register name</b>	<b>Description</b>	<b>Units/Notes</b>
<b>100</b>	Address	1 – 247	
<b>101</b>	Communication speed	0 – 115200, 1 - 9600	Bd
<b>102-105</b>	Read Only		HWS version
<b>106</b>	Comm. Protocol	1 - INGSIMON 2 – HTML,3 - MODBUS ASCII 4 – MODBUS RTU 5 – MODBUS TCP	Default: Modbus RTU

<b>4.4 Default parameters</b>		
<b>Parameter</b>	<b>Value</b>	<b>Comment</b>
<b>Address</b>	0x70h (112d)	
<b>Communication speed</b>	115200, N, 8,1	
<b>Communication Protocol</b>	0x04	MODBUS RTU

<b>4.5 Range of address</b>	
<b>Address [dec]</b>	<b>Comment</b>
<b>1 - 247</b>	For sensors
<b>248 - 254</b>	Reserve
<b>255</b>	Universal address – used only to read registers Writing to registers does not work with this address

## 5. Examples for Modbus RTU

### Example 5.1

<b>Set the communication speed from 115200 Bd to 9600 Bd for Address 0x70 (112 dec)</b>		
<b>Poll</b>	70 06 00 65 00 01 52 F4	The response is at 115200 Bd. In the next communication will use 9600 Bd
<b>Response</b>	70 06 00 65 00 01 52 F4	

### Example 5.2

<b>Set the communication speed from 9600 Bd to 115200 Bd for Address 0x70 (112 dec)</b>		
<b>Poll</b>	70 06 00 65 00 00 93 34	The response is at 115200 Bd. In the next communication will use 9600 Bd
<b>Response</b>	70 06 00 65 00 00 93 34	

### Example 5.3

<b>Read 8 registers from 100 from Address 0x70 (112 dec)</b>		
<b>Poll</b>	70 03 00 64 00 08 0F 32	
<b>Response</b>	70 03 10 00 70 00 00 47 50 53 4E 45 2A 3A 31 2E 30 00 04 B7 E5	
<b>Meaning:</b>		
<b>Byte [hex]</b>	<b>Description</b>	<b>Comment</b>
70	Address	
03	function code	Read holding register
10	count of bytes (16 dec)	
00 70	Address	
00 00	communication speed	115200 Bd
47 50	GP	
53 4E	SN	
45 2A	E*	GPSNE*
3A 31	_1	
2E 30	.0	_1.0
00 04	communication protocol	4 - MODBUS RTU
B7 E5	Checksum	

### Example 5.4

<b>How to get the address from sensor with the unknow address with universal address 0xff</b> <b>Be aware, that only 1 equipment can be connected to the Modbus network.</b>		
<b>Poll</b>	FF 03 00 64 00 01 D0 0B	Read register 100
<b>Response</b>	FF 03 02 00 70 90 74	70 – equipment's address

### Example 5.5

<b>Changing the address from 70h to 1h.</b> <b>Be aware, that only 1 equipment can be connected to the Modbus network.</b>		
<b>Poll</b>	70 06 00 64 00 01 03 34	Write to register 100 value 1
<b>Response</b>	70 06 00 64 00 01 03 34	01 – equipment's new address
The next communication with equipment will be with address 1		

### Example 5.6

<b>Changing the address from 1h to 2h.</b>		
<b>Be aware, that only 1 equipment can be connected to the Modbus network.</b>		
<b>Poll</b>	<b>01 06 00 64 00 02 49 D4</b>	<b>Write to the register 100 value 2</b>
<b>Response</b>	<b>01 06 00 64 00 02 49 D4</b>	<b>02 – equipment’s new address</b>
<b>The next communication with equipment will be with address 2</b>		

<b>Example 5.7</b>		
<b>Reading measured values from 0. register, 16 registers. Address 70h.</b>		
<b>Poll</b>	<b>70 03 00 00 00 10 4E E7</b>	
<b>Response</b>	<b>70 03 20 00 01 00 15 00 07 00 06 00 07 00 38 00 13 01 CF 04 48 00 30 00 13 09 50 00 11 00 27 01 40 00 8B B8 8A</b>	
<b>Meaning:</b>		
<b>Byte [hex]</b>	<b>Description</b>	<b>Comment</b>
<b>70</b>	Address	
<b>03</b>	function code	Read holding register
<b>20</b>	count of bytes (32 dec)	
<b>00 01</b>	GPS Quality Indicator	Locked
<b>00 15</b>	Current year	21 (dec)
<b>00 07</b>	Current month	07 (dec)
<b>00 06</b>	Current day	06 (dec)
<b>00 07</b>	Current hour	07 (dec)
<b>00 38</b>	Current minute	56 (dec)
<b>00 13</b>	Current second	19 (dec)
<b>01 CF</b>	Sun elevation	479 (dec), 47.9 °
<b>04 48</b>	Sun azimuth	1123 (dec), 112.3 °
<b>00 30</b>	Degrees of latitude	48 (dec)
<b>00 13</b>	Minutes of latitude	19 (dec)
<b>09 50</b>	Ten thousandths of a minute of latitude	2384 (dec)
<b>00 11</b>	Degrees of longitude	17 (dec)
<b>00 27</b>	Minutes of longitude	39 (dec)
<b>01 40</b>	Ten thousandths of a minute of longitude	320 (dec)
<b>00 8B</b>	Altitude	139 (dec) m
<b>B8 8A</b>	Checksum	

## 6. Used sensors

### 6.1 uBlox NEO-8M GPS receiver and only fully functional PCB for Interior using

