

---

# Communication Protocol MODBUS RTU used in TDS display

---

Comprehensive protocol description

---



# MODBUS RTU v TDS

## Datasheet

Created: 6.4.2009

Last update: 30.5 2011 10:54

Pages: 16

© 2011 Papouch s.r.o.

---

## Papouch s.r.o.

Address:

**Strašnická 3164/1a  
102 00 Praha 10  
Czech Republic**

Telephone:

**+420 267 314 267-8  
+420 602 379 954**

Fax:

**+420 267 314 269**

Internet:

**[www.papouch.com](http://www.papouch.com)**

E-mail:

**[info@papouch.com](mailto:info@papouch.com)**



## CONTENT

Description.....	4
Basic communication parameters .....	4
List of changes by individual versions of firmware.....	4
Switching between protocols.....	5
Spinel → MODBUS RTU .....	5
Configuration Permission .....	5
Switchover .....	5
MODBUS RTU → Spinel .....	6
Memory organization .....	6
Holding Register.....	6
Detailed information on instructions .....	7
Communication parameters.....	7
Configuration Permission .....	7
Device ID .....	7
Serial line communication speed.....	8
Data word format .....	9
Packet end distinction .....	9
Communication protocol .....	10
Display .....	11
Entering data on the display.....	11
Entering data as a number.....	11
Display brightness.....	12
Data validity .....	12
Remaining time of data validity .....	13

**DESCRIPTION**

This document describes MODBUS RTU communication protocol used in TDS display. The documentation of the hardware of the sensor and the description of its functionality is available from <http://www.papouch.com/> (detailed documentation also downloadable in PDF).

**Basic communication parameters**

---

- Communication line.....RS485
- Communication speed .....ranging 1.2 kBd to 115.2 kBd (*default: 9.6 kBd*)
- Number of data bits.....8
- Parity.....no parity
- Number of stop-bits.....1
- Delay before response .....2 ms<sup>1</sup>
- Starting address.....0x31
- Default protocol set by manufacturer .....Spinel

**List of changes by individual versions of firmware**

---

**version 02**

MODBUS RTU protocol added. TDS is only able to communicate via one protocol at a time. Switching between the protocols is possible using a function described below. (The default protocol, set by the manufacturer, is Spinel. Its description can be found in a separate document.)

---

<sup>1</sup> The delay has been incorporated to enable RS485 to switch the direction of communication.

## SWITCHING BETWEEN PROTOCOLS

The default protocol set by the manufacturer is Spinel. For switching into MODBUS, use the following instructions in the Spinel protocol.

### Spinel → MODBUS RTU

#### Configuration Permission

Enables to carry out a service instruction. This instruction must immediately precede the instruction for “Switchover”.

This instruction may not be used with a universal or broadcast address.

#### Request:

*Instruction code: E4H*

#### Response:

*Acknowledgement code: ACK 00H*

#### Examples:

<b>Request:</b>
2AH, 61H, 00H, 05H, 01H, 02H, E4H, 88H, 0DH
Configuration permission.
<b>Response:</b>
2AH, 61H, 00H, 05H, 01H, 02H, 00H, 6CH, 0DH
Instruction receipt acknowledged.

#### Switchover

The protocol can be switched over by a special instruction of the Spinel protocol, format 97. Only the address of a particular module can be used here (it is not possible to use a so called “broadcast” or universal address). This instruction must be intermediately preceded by the “Configuration Permission” instruction.

#### Request:

*Instruction code: EDH*

#### Response:

*Acknowledgement code: ACK 00H*

#### Examples:

<b>Request:</b>
2AH, 61H, 00H, 06H, 66H, 02H, EDH, 02H, 17H, 0DH
Instruction to switch from Spinel to MODBUS RTU.
<b>Response:</b>
2AH, 61H, 00H, 05H, 66H, 02H, 00H, 07H, 0DH
Instruction receipt acknowledged. After sending this response THT communicates via MODBUS RTU.

**MODBUS RTU → Spinel**

The method of switching from MODBUS RTU to Spinel is described on page 10 hereof.

**MEMORY ORGANIZATION**

**Holding Register**

<i>Address</i>	<i>Access</i>	<i>Function</i>	<i>Name</i>	<i>Page</i>
<b>Communication parameters</b>				
0x0000	read, write	0x03, 0x06, 0x10	Permission for configuration	5
0x0001	read, write	0x03, 0x06, 0x10	Address (ID)	7
0x0002	read, write	0x03, 0x06, 0x10	Communication speed	8
0x0003	read, write	0x03, 0x06, 0x10	Data word	9
0x0004	read, write	0x03, 0x06, 0x10	Packet end distinction	9
0x0005	read, write	0x03, 0x06, 0x10	Communication protocol	10
0x0064	read, write	0x03, 0x06, 0x10	Display entry	11
0x006E	read, write	0x03, 0x06, 0x10	Display brightness	12
0x006F	read, write	0x03, 0x06, 0x10	Data validity	12
0x0070	read, write	0x03, 0x06, 0x10	Remaining time of data validity	13
0x0071	read, write	0x03, 0x06, 0x10	Display entry as a number	11

## DETAILED INFORMATION ON INSTRUCTIONS

### Communication parameters

#### Configuration Permission

This instruction must precede all instructions writing into the Holding Register. It prevents any undesirable change of configuration.

Configuration Permission instruction must precede *any* configuration instruction. It is not allowed to enter Configuration Permission together with other parameters using Multiply Write.

#### Function codes:

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

#### Memory position and length:

Starting address	2 Bytes	0x0000
Register count	2 Bytes	1

#### Parameters:

Number of bytes	1 Byte	2
Result	2 Byte	0x00FF = Configuration Permission accepted

#### Device ID

Address (ID) of the device. Devices connected to one communication interface must hold unique addresses. The address unambiguously identifies the device within the network. The default address is 0x31.

#### Function codes:

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

#### Memory position and length:

Starting address	2 Bytes	0x0001
Register count	2 Bytes	1

#### Parameters:

Number of bytes	1 Byte	2
Address	2 Byte	Device address within the range 1 to 247

**Serial line communication speed**

To configure the speed of the communication line.

**Function codes:**

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x0002
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Speed code	2 Byte	Speed code: 1200 - 0003H 2400 - 0004H 4800 - 0005H 9600 - 0006H ( <i>default setting</i> ) 19200 - 0007H 38400 - 0008H 57600 - 0009H 115200 - 000AH



**Data word format**

To configure parameters of the data word (parity, number of bits and number of stop-bits).

**Function codes:**

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x0003
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Speed code	2 Byte	Code according to the following table. (default: 0x0000.)

Code	No of bits	Parity	No of stop-bits
0x0000 (default)	8	none (N)	1
0x0001	8	even (E)	1
0x0002	8	odd (O)	1
0x0003	8	none (N)	2
0x0004	8	even (E)	2
0x0005	8	odd (O)	2
0x0006 to 0x00FF	8	none (N)	1

**Packet end distinction**

To configure the delay between the bytes that will be understood as the end of each packet. The delay is entered as a number of bytes. It is possible to enter a value ranging from 4 to 100. The default value is 10.

**Function codes:**

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x0004
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Delay	2 Byte	The delay as a number of bytes. It is possible to enter a value ranging from 4 to 100.

**Communication protocol**

This function enables THT to be switched into Spinel protocol. After sending the response, THT switches to the selected protocol and communicates through it from this point on. (Every protocol contains an instruction for switching between protocols.)

**Function codes:**

- 0x03 – Read Holding Register
- 0x06 – Write Single Register
- 0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x0005
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Protocol code	2 Byte	Protocol code: Spinel - 0001H MODBUS RTU - 0002H

## Display

### Entering data on the display

Memory space for entering data on the display.

#### Function codes:

0x03 – Read Holding Register






0x06 – Write Single Register

0x10 – Write Multiple Registers

#### Memory position and length:

Starting address	2 Bytes	0x0064
Register count	2 Bytes	5

#### Parameters:

Number of bytes	1 Byte	2
Data on the display	2 Byte	<p>Each character is one 16bit register. ASCII characters ranging from &lt;0 to 9&gt;, &lt;a to z&gt;, space, dash and dot.</p> <p><i>Table of characters:</i></p> <p> .....0 to 9</p> <p> .....A to J</p> <p> .....K to U</p> <p> .....V to Z</p> <p> .....dash</p>

### Entering data as a number

Memory space for entering data on the display as one number within the range from 0 to 9999.

#### Function codes:

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

#### Memory position and length:

Starting address	2 Bytes	0x0071
Register count	2 Bytes	1

#### Parameters:

Number of bytes	1 Byte	2
Data on the display	2 Byte	A number ranging from 0 to 9999.

**Display brightness**

To set the brightness of the display in five steps.

**Function codes:**

- 0x03 – Read Holding Register
- 0x06 – Write Single Register
- 0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x006E
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Data on the display	2 Byte	Values ranging from 0 to 4, where 0 = switched off, 1 to 4 = brightness levels (4 is the maximum)

**Data validity**

This instruction defines how long the data is to be displayed. After this time has elapsed, the display will show four dashes (- - - -). The entered time is valid permanently, i.e. not only for the currently displayed value but also for all values received later. To cancel this function, enter 0.

(This function is suitable for periodical updating of the displayed value. After the entered time has elapsed, the dashes will advise the operator that data updating has encountered an error.)

**Function codes:**

- 0x03 – Read Holding Register
- 0x06 – Write Single Register
- 0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x006F
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Data on the display	2 Byte	Time in seconds; if 0, values will be displayed without any restriction.

**Remaining time of data validity**

---

Shows the remaining time of value displaying. (Can also be entered, but we recommend that you rather use the previous option.)

**Function codes:**

0x03 – Read Holding Register

0x06 – Write Single Register

0x10 – Write Multiple Registers

**Memory position and length:**

Starting address	2 Bytes	0x0070
Register count	2 Bytes	1

**Parameters:**

Number of bytes	1 Byte	2
Data on the display	2 Byte	Value in seconds showing the remaining time of value displaying.





# Papouch s.r.o.

Data transmission in industry, line and protocol conversions, RS232/485/422/USB/Ethernet/GPRS/WiFi, measurement modules, intelligent temperature sensors, I/O modules, and custom-made electronic applications.

Address:

**Strašnická 3164/1a  
102 00 Praha 10  
Czech Republic**

Telephone:

**+420 267 314 267-8  
+420 602 379 954**

Fax:

**+420 267 314 269**

Internet:

**[www.papouch.com](http://www.papouch.com)**

E-mail:

**[info@papouch.com](mailto:info@papouch.com)**

