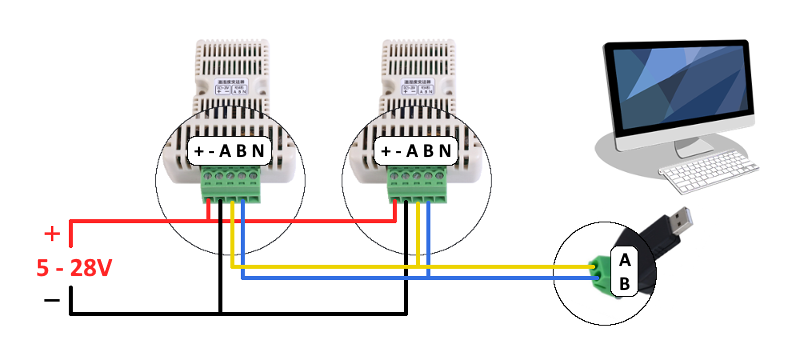
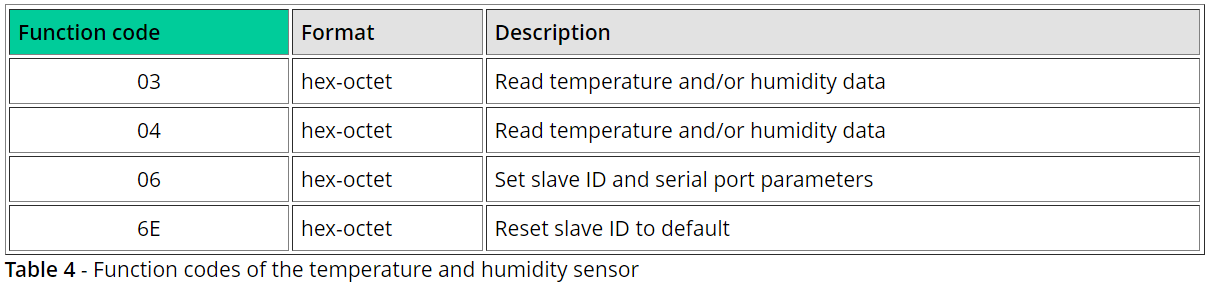
**Temperature sensor modbus command frame**



The [Modbus RTU](https://ozeki.hu/p_5854-modbus-rtu.html" \t "_blank) command frame **build up from 8 bytes** totally. It is constructed from **1 byte address** field which value can be set between 1 and 247 (broadcast is 0). This is the ID of the slave (or device). After that there is **1 byte function** field. This module supports 4 type of function codes: 2 ones for reading temperature and/or humidity data and 2 ones for setting/reseting device settings. Then it followed by **4 byte data** field. In this case, this field tells what you would like to read (for example temperature data and/or humidity data) or which setting value you would like to set to the slave (such as slave ID). So, the interpretation of this data field depends on the function code. Finally, **2 bytes for checksum** field that calculated from the previous 6 bytes using CRC16 checksum.



You can see some information about what function codes are supported. The function code 0x03 and 0x04 means the data reading. Function code 0x06 allows you to customize the device settings (for example slave ID, baud-rate, stop bit etc.). Finally, using 0x6E you can reset the sensor ID to default. The following **Table 4** summarizes these.

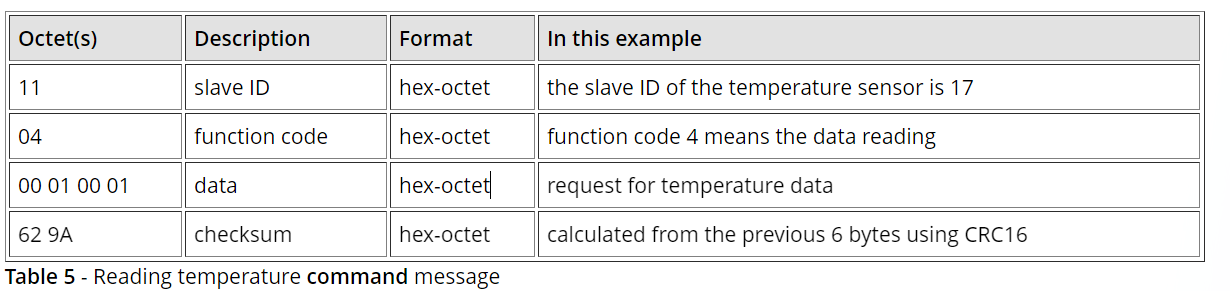


## Examples

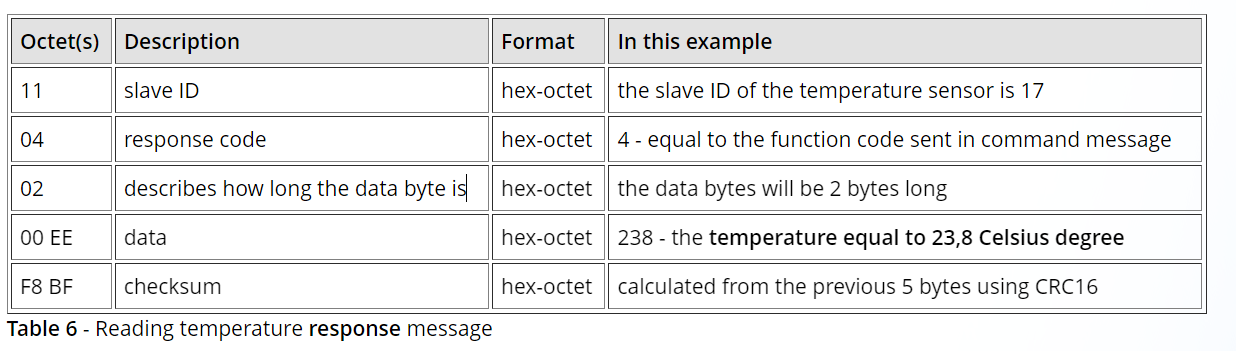
In this step, you will find some examples what frame you can send to the slave and what will you get from it. You will also get an answer what the sent command frames means, and how to interpret the response frame.

### **Read temperature**

To receive the temperature data measured by the sensor, the following byte array must be sent to the slave in hexadecimal format: **11 04 00 01 00 01 62 9A**. The first byte (0x11) means the **slave ID** to whom the message to be sent. The second byte is the **function code**. In this case, the 4 (0x04) means the data reading. So, the device will be know, you would like to read some data. The next 4 bytes will tell what kind of **data** you would like to read. To get temperature, the data is 0x00010001. The last two byte (0x629A) are the **checksum** calculated from the previous 6 bytes. The following **Table 6** explains how to interpret these bytes as a command message.



If you have sent the command message above to the sensor, you will get a response message like the below one. The first byte (0x11) is the **slave ID** from which the response message came from. The second byte is the **response code** that is the same as the function code in the command message. The third byte (0x02) is the data **length**. Now it is 2, so the length of data will be 2 bytes long. In this case, the **data bytes are 0x00 and 0xEE**. In decimal numeral system **this means 238 that is equal to 23,8 Celsius degree**. The last two bytes are the **checksum** calculated from the previous 5 bytes. The following table explains how to interpret these bytes as a response message.



### **Read humidity**

To request the humidity data value from the sensor, you need to send the following bytes as a command message if the slave ID of the sensor is 17: **11 04 00 00 00 01 33 5A**. The **Table 7** below shows how to interpret it.

