

# FlyWire

## FireFly Communication Protocol for the Probe TRH



This document describes the commands available for the FireFly to communicate with a Probe TRH and how to change the settings by using additional factory, test and custom commands. See the FlyWire communication protocol document for additional information.

The following topics will be addressed in this document:

1. Principle of operation
2. Standard/factory/test commands
3. Custom commands
4. EEPROM map

## 1. Principle of operation

The Probe TRH is an Air Temperature (T) and Relative Humidity (RH) probe that provides accurate readings inside a stack of commodities. The Probe TRH is powered by the FireFly and automatically starts measuring after connecting to the FireFly. The Probe TRH data will be sent in addition to the data from the FireFly's internal sensors, using the FireFly's wireless transmitter.

The Probe TRH tube comprises of a metal tube with the sensor element located in the lower section of the tube. The FireFly is connected to the Probe TRH using the cable at the top of the tube.

The Probe TRH contains a microcontroller running the FireFly stack and a T/RH sensor, as described in more detail in the FlyWire protocol. All standard and optional commands are implemented, except for Set Values. The product is assigned to the Probe TRH is 5.

The Probe TRH will not tare any value when a magnet is used to reset the FireFly. So a Reboot command (0x03) or taring the fire will not have any effect on the reading.

## 2. Standard/factory/test commands

The following commands have been implemented:

Request Sensor Values:

0x5A | 0x04 | 0x02 | CHK

After receiving this command the Probe TRH replies with both the Temperature and Relative Humidity values in signed integer format. Both values are multiplied by 100 to allow a 2 decimal fractional value to be transmitted as a signed integer. The internal sensor has a Cyclic Redundancy Check (CRC) that is calculated for both temperature and relative humidity. If the CRC is invalid, the Solar Chimney will report a value of -100.00.

The Slave will respond with:

STF | 0x08 | 0x06 | MSBT | LSBT | MSBH | LSBH | CHK

MSB is the most significant byte and LSB is the least significant byte. xSBT is the temperature and xSBH the humidity.

For example, the Solar Chimney can give the following reply:

STF | 0x08 | 0x06 | 0x0A | 0x98 | 0x19 | 0xD5 | CHK

This can be decoded as follows:

Temperature: 0x0A98 = 2712 : 27.12 °C

Humidity: 0x19D5 = 6613 : 66.13 %RH

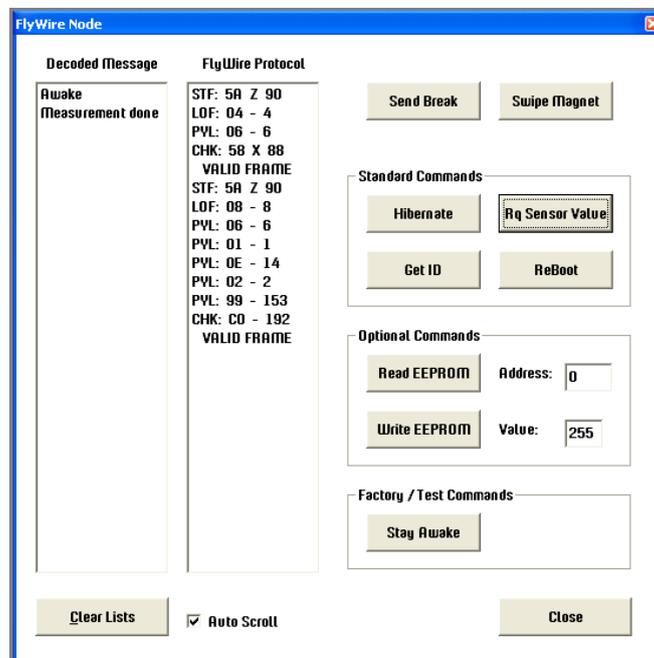


Figure 1: Solar Chimney reply to Get Sensor Values command.

Stay Awake:

0x5A | 0x04 | 0xE0 | CHK

This command will override the 1 second auto sleep feature of the Probe TRH. After issuing this command the Probe TRH will stay awake until a power cycle occurs or a Reboot or a Hibernate command is issued.

The Probe TRH will reply with an ACK.

STF | LOF | NAK | CHK

In **Error! Reference source not found.**, the Stay Awake command has been issued to prevent the Probe TRH from going to sleep. This allows the user to press the Get ID and Rq Sensor Value buttons without having to send a break condition to wake up the device after a 1 second delay.

### 3. Custom commands

Besides formatted temperature and relative humidity data, the Probe TRH can return the raw sensor values. For this, a custom command has been implemented compliant with the FlyWire protocol.

Get Raw Value:

0x5A | 0x04 | 0xF1 | CHK

This command returns the raw values as an unsigned integer

The Slave will respond with:

STF | 0x08 | 0x06 | MSBT | LSBT | MSBH | LSBH | CHK

MSB is the most significant byte and LSB is the least significant byte. xSBT is the temperature and xSBH the humidity.

SRH and ST denote the raw sensor output for humidity and temperature, respectively.

IMPORTANT NOTE: These formulas only apply when SRH and ST are represented as decimal values!

$$RH[\%] = 100 \cdot \frac{Sr_h}{2^{16} - 1}$$

$$T[^{\circ}C] = -45 + \frac{175 \cdot St}{2^{16} - 1}$$

## 4. EEPROM map

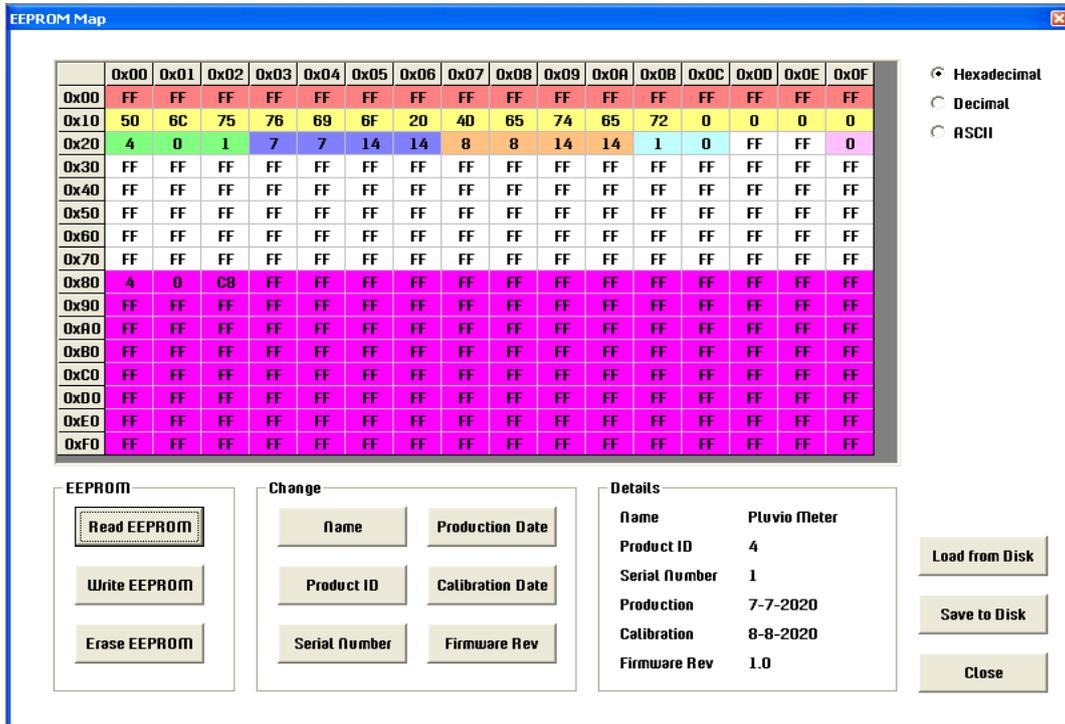


Figure 2: Example EEPROM memory snapshot.

0x00 – 0x0F Reserved

0x10 – 0x1F These locations hold the name of the device in ASCII. Unused locations at the end should be padded with 0x00. In the example above the EEPROM holds the name Probe TRH.

0x20 – 0x22 Product ID and Serial Number. The SN consists of two bytes forming an unsigned integer. High byte at 0x21, low byte at 0x22. In the example the Product ID is 4 for a Pluviometer. For a Probe TRH the Product ID is 5 (see list of supported devices in the FlyWire protocol document). The Serial Number is 0x0001 (1 in decimal)

0x23 – 0x26 Production date in DD/MM/YY/YY format. The year is split in two bytes. E.g. 2020 is stored as 20, 20 or 0x14, 0x14 in hex format. In the example both production date is set to July 7<sup>th</sup>, 2020.

0x27 – 0x2A Calibration date in DD/MM/YY/YY format. In the example the calibration date is August 8<sup>th</sup>, 2020.

0x2B – 0x2C Firmware version. Major value is stored at 0x2B, minor value at 0x2C. In the EEPROM map above the firmware version is 1.0

0x2F            Location 0x2F is used for debugging. If its value is non-zero, the on-board LED will be lit as long as the Solar Chimney is active, and switch off in hibernation. When set to zero, the LED will always be off to preserve power.

0x80 – 0xFF    This area is reserved for calibration data. It is not used in the Probe TRH.