FlyWire
FireFly Communication Protocol for the Pluviometer

This document describes the commands available for the FireFly to communicate with a Pluviometer 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. Factory/test commands
3. Custom commands
4. EEPROM map

1. Principle of operation

The Quantified Pluviometer is an ultra-low power wireless rain gauge. It deploys the tipping bucket principle to measure the quantity and rate of rainfall. The Quantified Pluviometer is designed to be powered by a Quantified FireFly sensor node, and to transmit data through the FireFly transmitter.

The Pluviometer is available in a small (S) and in a large (L) version. The Pluviometer S is a tiny device, ideal for applications where space is limited. The Pluviometer L has a higher resolution thanks to a larger funnel, and comes with bird spikes and a leaf grid.

The Pluviometer contains a microcontroller running the FireFly stack and a magnetic switch that counts the number of times the spoon tips and multiplies that by the volume of the spoon.

The Pluviometer measures the total amount of rain. The output value can be tared (i.e., the output can be set to zero) by resetting the FireFly using the magnetic switch, by cycling the power, or by issuing a Reboot command (0x03).

2. Factory/test commands

The following test command has been implemented:

Stay Awake:
0x5A | 0x04 | 0xE0 | CHK
This command will override the 1 second auto sleep feature of the Pluviometer. After issuing this command the Pluviometer will stay awake until a power cycle occurs, a Reboot or a Hibernate command. This command is useful during testing without having to send a break each time to wake-up the device.

The Pluviometer will reply with an ACK:
STF | LOF | NAK | CHK

In the snapshot below, the Stay Awake command has been issued to prevent the Pluviometer 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.
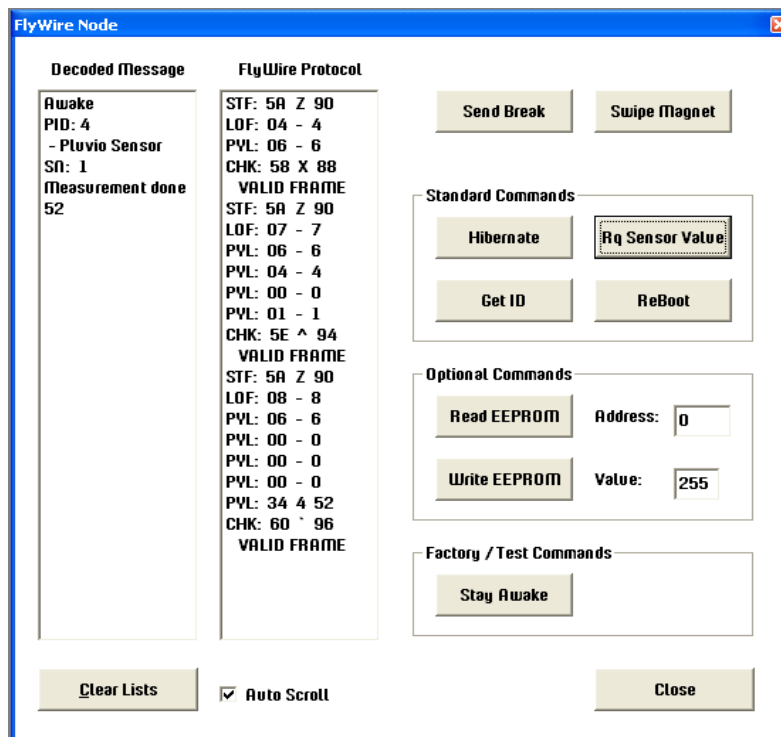

Figure 1: Example of the response of a Pluviometer L to a Get ID and Get Sensor values command.

3. Custom commands

Compliant with the FlyWire protocol, one device-specific command has been implemented.
The Get Raw Value command can be used to obtain the number of times the spoon has tipped. As opposed to the default Request Sensor Values command, which returns the number of tips multiplied with the volume per tip, the Get Raw Value command does not multiply the number of tips with the tipping volume.

Get Raw Value:
0x5A | 0x04 | 0xF1 | CHK
This command returns the value as an unsigned long.

The Slave will respond with:
STF | 0x08 | 0x06 | MMSB | MSB | LSB | LLSB | CHK

MMSB is the most significant byte and LLSB is the least significant byte. As an example a value of 0x00000133 (307 decimal) will be returned as follows:

STF | 0x08 | 0x06 | 0x00 | 0x00 | 0x01 | 0x33 | CHK
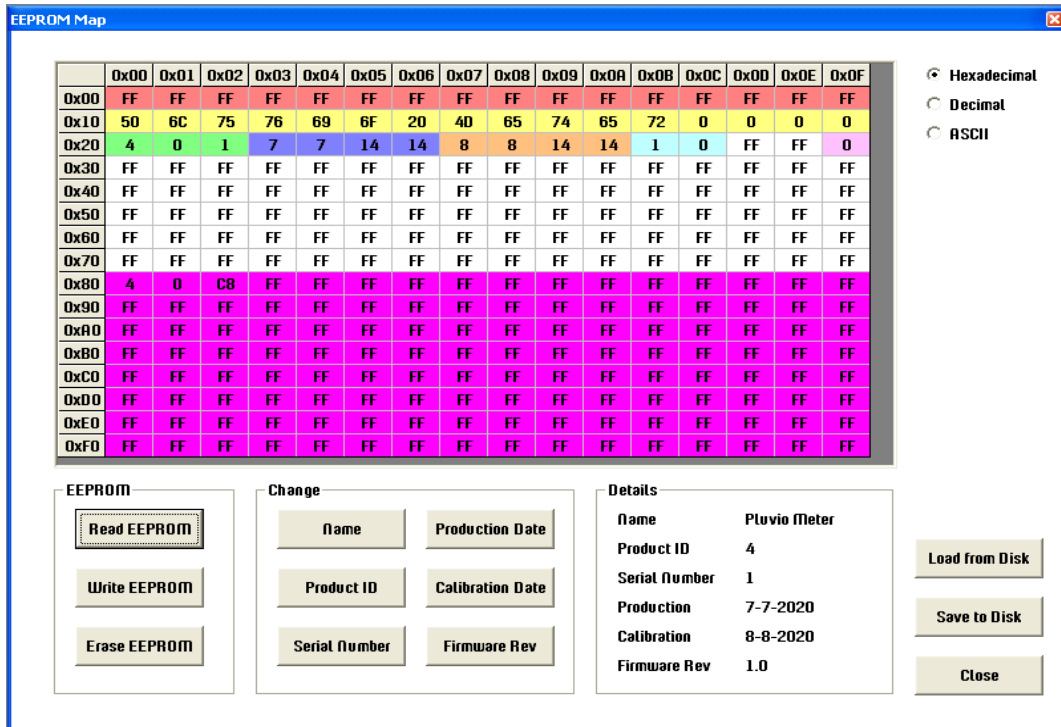
4. EEPROM map

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 "Pluvio Meter".

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 L and 7 for a Pluviometer S (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 7th, 2020.

0x27 – 0x2A   Calibration date in DD/MM/YY/YY format. In the example the calibration data is august 8th, 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   Used for debugging. If its value is non-zero, the on-board LED will be lit as long as the Pluviometer is active, and Switch off in hibernation. When set to zero, the LED will always be off to preserve power.

0x80 – 0x8F   This area is reserved for calibration data.