Flowbit Hardware Prototype

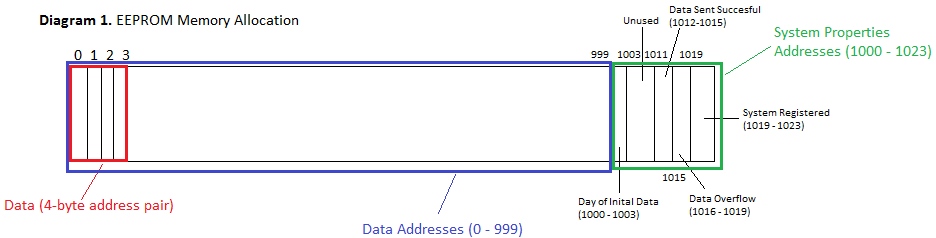
1. Hardware Overview  
The device uses the Arduino Uno MCU to drive the modules and components on the hardware. The hardware includes an Adafruit flow meter sensor, an Adafruit GPS, an Adafruit Real Time Clock module, and a Seedstudio SIM900 GPRS module. The device is powered using portable power sources such as a solar panel and a rechargeable lithium battery. More detailed specifications are described below.

2. Arduino Uno MCU

2.1 Overview  
The MCU uses an Arduino Uno Board (ATmega328P), a high performance low powered 8-bit RISC architecture chipset operating at 5V. The Arduino has an onboard flash memory of 32K bytes, SRAM of 2K bytes, and 1K bytes of non-volatile memory (EEPROM) with up to 100,000 write cycles. It supports 14 digital I/O pins that can be used for extra modules as needed.

2.2 EEPROM  
The EEPROM is a non-volatile storage solution that we are harnessing to store data and several crucial system properties. It has a storage size of 1024 bytes, a byte being 8-bits each. We currently define all data to be 32-bit values to maintain consistency internally. This means we are storing data in an address byte in a pair for of 4. Below is a diagram (diagram 1) detailing the storage allocation of the following data:

PULSE DATA – address byte 0 to 999  
DAY OF INITIAL DATA -address byte 1000 to 1003  
DATA SENT SUCCESS - address byte 1012 to 1015  
DATA OVERFLOW -address byte 1016 to 1019  
SYSTEM REGSITERED -address byte 1020 to 1023



2.3 Memory Allocation  
Memory allocation of large numbers such as date time and sensor pulses is crucial when working on an 8-bit architecture device. Power source reliability in extreme and rural environments was also factored in when dealing with memory allocation. The scenario where power would suddenly cut off was taken into account.

Data that is collected from the flow meter is ultimately recorded into the volatile section of the memory where there will remain with the retention of 20 years even in the absence of a power source. The collected data is stored in the EEPROM of 1kB. With a 1kB EEPROM we are able to store up to 250 slots of data points. This is about 250 days of data on a daily resolution. On an hourly resolution, this is about 10.4 days’ worth of data. As you can see, this is dependent on resolution of data. Data would be removed from each slot after it has successfully sent the data set the SMS and has been successfully parsed by the server. If the server does not receive the data, the data would continue to be stored and it will always attempt to upload everything that is still in the EEPROM. This data were to overflow, new incoming data would replace the oldest data from the beginning.

We are working on finding solutions to store more data offline. We are looking into MicroSD cards as a solution for local storage.

2.4 Pin Wiring

2.5 Runtime  
At runtime, the device goes through three different states where it loops in the last state until the system is powered off. When the system is powered on, it initializes all the serial ports to enable communications with all the modules attached to it. It initializes all the system properties on the EEPROM as well. Based on the EEPROM, it is decided if the GPS will need to be initialized are not. The system will determine if it has already registered. If so, it will not initialize the GPS.

As enters the second state, they will check if the system has registered again. If it is the first time that the device is turned on, it will enter the GPS to get a fix from the satellites on its location. After it has achieved a fix from more than three satellites to obtain acceptable accuracy, it will send its registration to the server via SMS. More details on how the registration works is explained in both the backend documentation and the GPRS module documentation below. They will not enter this state if the system is already registered.

It then proceeds to the last state where it monitors the flow meter for incoming water usage. The monitoring of water usage is handled using the concept of sessions. When the flow meter detects an incoming pulse, it will enter a session where the incoming pulses are first aggregated in the SRAM due to its volatility; direct writes into the EEPROM will wear out the lifespan of the material quickly.

A session is a set duration where all incoming pulses are stored into the SRAM variable first. A session is always resent whenever pulses are incoming. Whenever there are no change in pulses for that set duration, the session will end and it will store that aggregated number into the EEPROM. At the end of each session, and will also check the date to see if it is time to send the data to server.

3. GPRS Module

3.1 Overview  
The GPRS module is a Seedstudio SIM900. It is the primary module for communications in out of the future is to server. The SIM900 is a Quad-Band 850/900/1800/1900MHz, which should work on GSM networks in most countries. The module uses AT commands to control and send commands from the MCU to the module. With a SIM card attached, the GPRS module is capable of making calls, sending SMS, and receiving cellular data coverage.

3.2 SMS  
The SMS functionality is the main mode of communication that the device uses to send and receive data from. It can send up to 160 characters per SMS with varying limitations on you and characters. More is explained below. The cellular data uses the embedded TCP/UDP stack protocol for communications to work them read SMS that is sent to the SIM card’s number.

There are certain character limitations that the module has. For example, the character ‘[‘ will always be sent as ‘(‘ in the SMS. Other known characters that does this include the ‘{‘ / ‘}’ pair.

3.2.1 Sending SMS  
SMS is sent for two purposes. The first purpose is to register the device to the server on its first runtime. The data that is sent on the SMS is the following:

SYSTEM ID – the system ID  
SECRET KEY – the secret key unique to each system  
LATITUDE – the latitude determined by the GPS  
LONGITUDE – the longitude determined by the GPS  
FLOW METER SIZE – the literal flow meter size  
FLOW METER CONVERSION FACTOR – the conversion factor from pulse to liters based on flow meter size

More detail about how the registration system works will be described in the backend documentation.

The second purpose is to send data of the pulses stored in the EEPROM. Two crucial parts of the SMS is:

TIME – time in UTC associated with the data in the first index of the set  
DATA SET – data set of pulses at equally separated time interval as specified

The data set will be on average a length of 24 (at hourly resolution) to optimize number of SMS sent within the maximum character limit. If the amount of data that is needed to be sent exceeds the limit, a delimiter will be used to indicate that the remaining data will be sent in the remaining texts to follow.

3.2.2 Receiving SMS

3.3 Power Management  
The GPRS module draws power up to 2A bursts when trying to communicate and establish itself with the network. During idle mode it draws 50mA and 1.5mA at sleep mode. In order to save power, the module is only turned on when it is required. It is turned on only when it needs to send an SMS such as the registration or sending data. Otherwise it is always off.

4. GPS

4.1 Overview  
The GPS module used is the Adafruit Ultimate GPS Breakout V3. The GPS is only used during the first initialization of the device. It is used to track the location as required for the registration part of the device’s runtime. The GPS is turned off afterwards and never used again.

The positional tracking is accurate up to 1.8m. The device also turns altitude of two maximum of 27,000m. During tracking, the module draws a current of 25mA.

5. Flow Meter Sensor

5.1 Overview  
The sensor is an Adafruit Liquid Flow Meter.