An Open Source Hardware & Software online raingauge for real-time monitoring of rainwater harvesting in Sri Lanka

***Y. Chemin*[1,2] *, N. Sanjaya*[1,2] *, P.K.N.C. Liyanage*[1]**

[1]International Water Management Institute, [2]University of Moratuwa, Faculty of Architecture

Abstract

The rainwater harvesting network is in need of volumetric estimation of rainfall-runoff contributed to recharge of the aquifers. Regular reporting of high quality rainfall events and intensity is still a challenge to be addressed in Sri Lanka.
A combination of Open Source Hardware (OSHW) and Free & Open Source Software (FOSS) is being used to created a royalty-free, cheap raingauge design, with full control of on-board data collection and statistics.
Actual state-of-development has gone through the generic weather station prototype delivery to Irrigation Department for tank management and is in testing phase. A local manufacturer has delivered an all-in-one integrated board based on the prototype provided. Local manufacture of the sensors (tipping-bucket and other wind sensors) is under experimentation.

Introduction

Sri Lanka has already displayed its share of signs of climate change with global temperature warming over past century. Many researchers have concluded that no significant systematic trend of annual rainfall is evident in Sri Lanka but the pattern of the rainfall has been changed (De Costa, 2008 ; Basnayake, 2011 ; Chandrapala, 1996 ; Jayatillake et. al, 2005 ; Domroes, 1996 ; Manawadu and Fernando, 2008). During the last decade, Sri Lanka has experienced number of extreme rainfall events and severe drought in crop growing periods. Farmers are continuously complaining to planners about the uncertainty and inadequacy of rainfall. The country has seen flash floods, ponding, landslides and urban floods due to high intensity rainfalls frequently. Also, some provinces have seen by far and large the difficulties to manage erratic and insufficient rainfall for agriculture. The impact of the rainfall event depends on how it unfolds as much as on the final rainfall tally. 40mm of rainfall in 24 hours in an urban area (city) may not become a disaster, and is beneficial to rural areas as it permits proper infiltration to the crops root zone. But if the same rainfall amount falls within an hour it can change the entire economic environment of that city, or in agricutltural areas, it could be lost as runoff.

Open design is the development of physical products, machines and systems through use of publicly shared design information (https://en.wikipedia.org/wiki/Open_Design). Open Source HardWare (OSHW) is a direct consequence of the Open Design philosophy (Pearce, 2013a). It permits to share blueprints of open designs in formats ready for circuitry manufacturing tools. Such is the Arduino micro-controler. As the makers of Arduino describe it: "*Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists and anyone interested in creating interactive objects or environments*" (*www.arduino.cc*). Recently, a Sri Lankan version of Arduino was created under the name of "Lakduino" (Fig. 1 ; *www.lakduino.com*), is manufactured and distributed in Sri Lanka.

In the same line, many add-on board have been open designed to plug on Arduino and provide specialised functionalities. These add-on board are called "shields" in the Arduino jargon. Shields may provide sensors, GPS radio or communication devices, are *stackable*. They can be plugged on top of each other to provide the required types of combination of i.e. sensors, location services and online reporting.

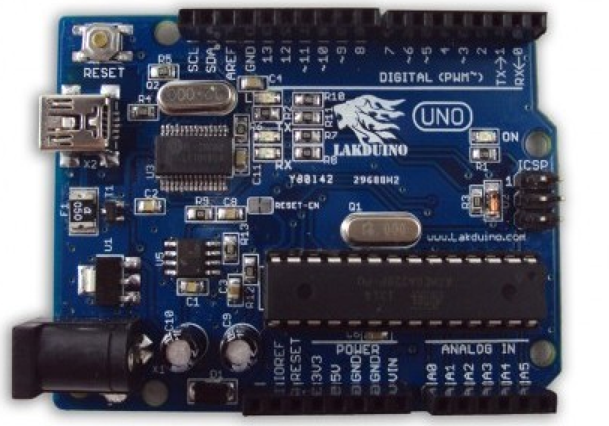

Figure 1: Sri Lanka's Arduino: The Lakduino

Since Arduino systems are open source software programmable, it belongs to the owner to generate the actual program that will process the Arduino *stack*. This is a strong advantage, as it is possible to custom build a generic tool, then customize it to conditions, environments or experiment. It has now become more prominent in several fields of science to design your own experiment to know/control more uncertainties using OSHW & FOSS (Pearce, 2013b ; Pearce, 2014).

This research explores the creation of fully customized meteorological station with a raingauge, based on the open design, open hardware, open source software corner stones. It has the benefits of being replicable royalty-free, customizable in terms of sensors and data logging analysis & on-board statistics. The aim of this study is to assess the Commodity Of The Shelf (COTS) possibilities to create a Mini Weather Station (MWS) that can be cost effective and customizable under the *Open* philosophy.

Methodology

The following flow-chart brings together the required elements for an effective sensing and data logging of weather information (generic basis). After designing this generic version, the targeted design as a raingauge was a subset to be identified and implemented.

The various components of this generic design were found in various online micro electronic stores in Sri Lanka and abroad. The lakduino (http://lakduino.com) website provides the basic information and an online store (in LKR) with postal delivery, along with the certification of full compatibility with the original Arduino design (http://www.arduino.cc), which we tested positively. The lakduino micro-controler is programmable with 32Kb ROM storage, which is compiled and uploaded through the Arduino IDE (http://arduino.cc/en/Main/Software). All software already online is following the Open Source Software philosophy, and thus is copylefted (https://en.wikipedia.org/wiki/Copyleft).

Three main components were used to build the weather station *circuitry stack*, the first one is the "Weather Shield", the second is the GPS, and finally the data logger.

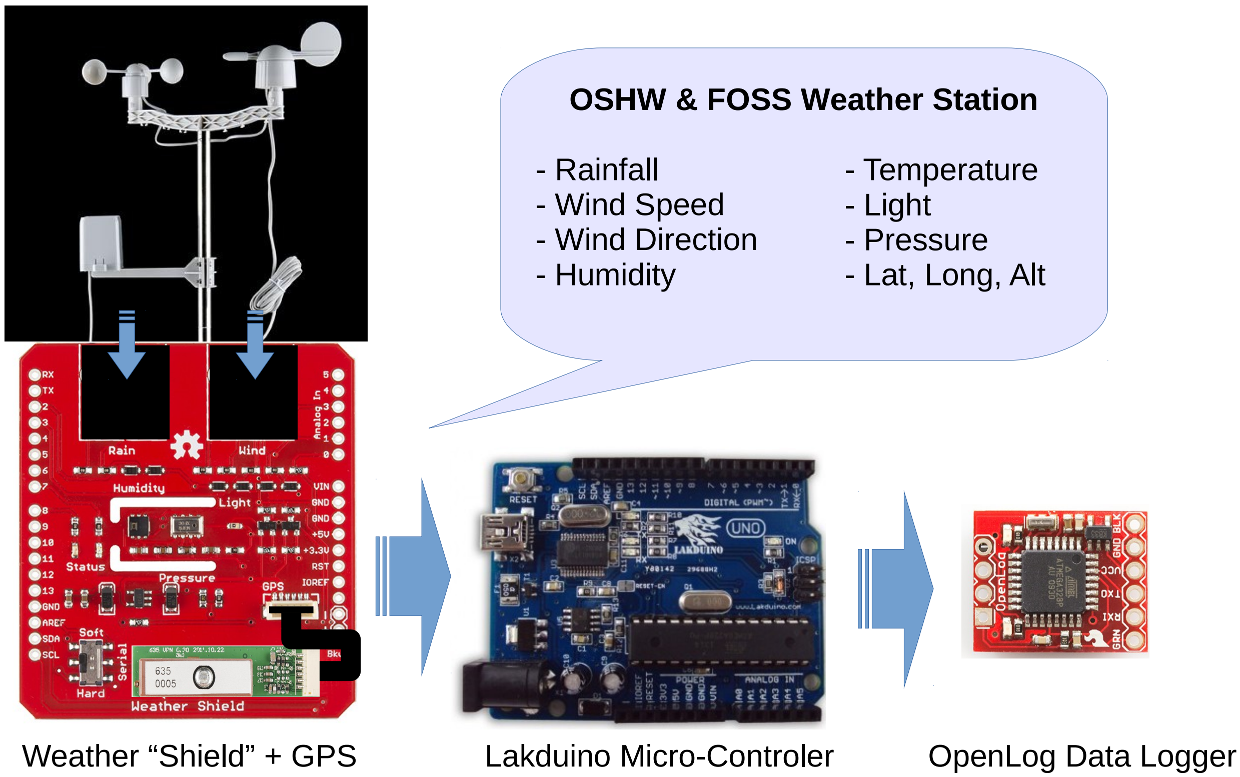

Figure 2: Generic design of a OSHW + FOSS weather station

The "Weather Shield" is an add-on board to the Arduino, and plugs directly on top of it. It is manufactured and distributed by Sparkfun (http://www.sparkfun.com), an online store of small electronics. On the board are temperature, humidity, light and atmospheric pressure sensors. Two additional plugs provide connectivity to a wind/rain weather kit (also from the same online store). The set once soldered, and mounted, can be programmed through the Arduino board to return the data from all sensors to the session terminal. The data is returned to the terminal in Comma Separated Values form (CSV ; file extension *.csv ; https://en.wikipedia.org/wiki/Comma-separated_values).

The GPS is a small form factor radio geolocation system, also from the same online store as the "Weather Shield" and the weather kit. It plugs directly on the "Weather Shield" through a small cable connector. Programming is simple through a specialised library that can be downloaded from a url in the same online site. Besides providing an accurate GMT time for each weather sensors reading, the GPS permits to geolocate the readings, without the need of external GPS at each installation. This is useful when using the raingauge station on a rotation experience to cover more sampling areas or evaluating the best grid network with less number of stations.

The data logger is an OpenLog Design (https://github.com/sparkfun/OpenLog/wiki) and receives a micro SD card, tested with up to 16Gb. Internally, it is using a fat16/fat32 file manipulation library for Arduino from Bill Greiman (https://code.google.com/p/sdfatlib/). The data coming from the Arduino program is printed to the terminal screen, and this same print of data is recorded as text files in the micro SD, without any configuration required.

Results

A generic meteorological weather station was designed and prototyped to fit several research and monitoring requirements within Sri Lanka. The first prototype is shown in Figure 3 and was delivered to Irrigation Department in Anuradhapura for testing in June 2014 after a first trial run and discussion with them where specific rainfall statistics were identified.

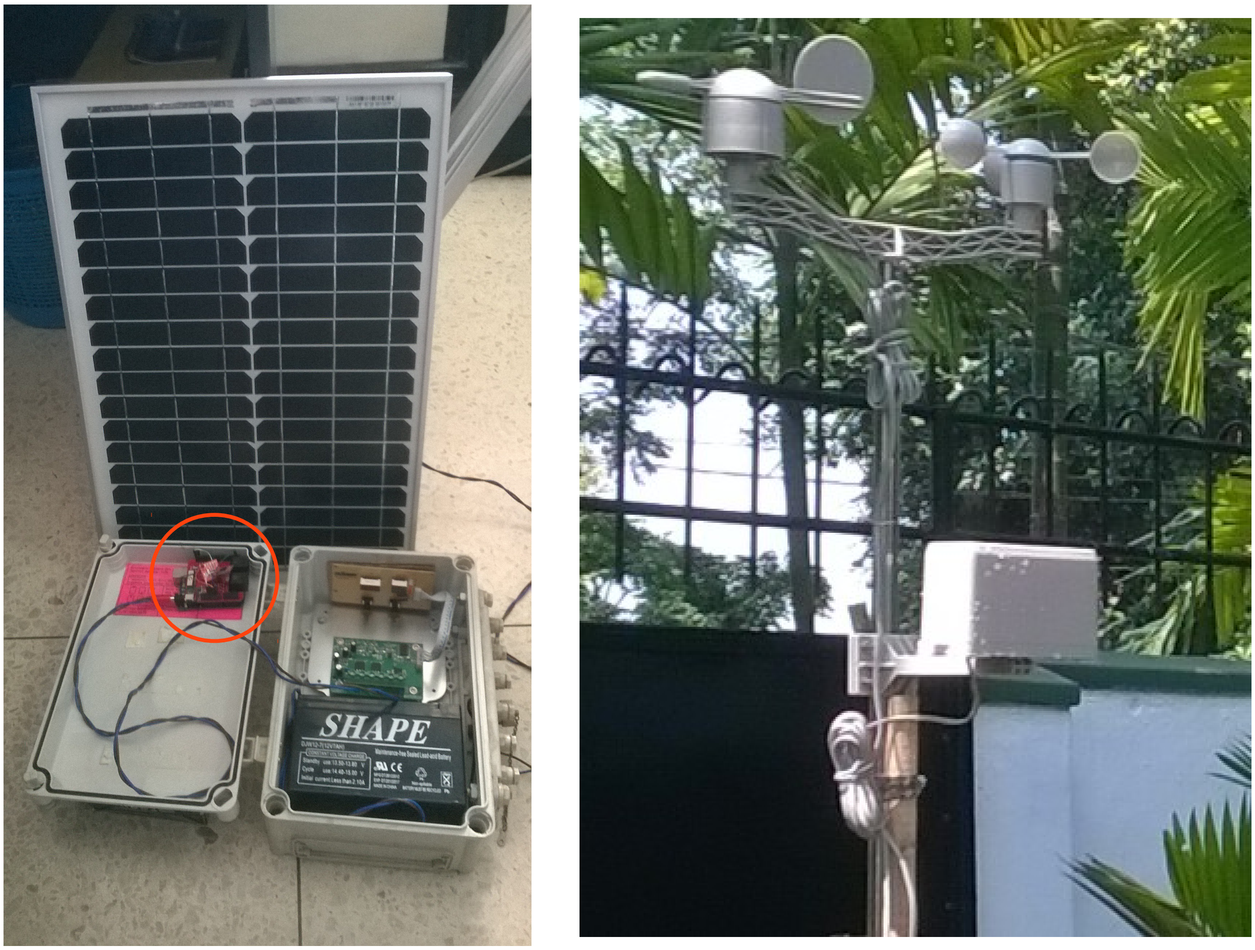

Figure 3: Left side: circuits (red circle) & power. Right side: wind sensors & raingauge

Specialized programming for the requirement of surface water flow from the irrigation department has been done in the form of 5 minutes statistics of rainfall intensity, with a detection flag for rainfall event detection and duration (Figure 4). The aim of the required customized analysis is to corner the rainfall types and their influence on the runoff pattern for improving tank management. Such programming is now available as a basis for the raingauge development for high intensity evaluation and monitoring of rainfall-runoff events and modeling.

The raingauge data programmed is analysed below (Figure 5). There is a rain detection flag (RainFlag, values are 0 or 1) operating on a 5 minutes time interval. Three statistical aggregations of Rainfall are computed, 5 minutes (Rain5 ; not shown), hourly (RainH) and daily (RainD ; not shown).

```
//Interrupt routines (these are called by the hardware interrupts, not by the main code)
//-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
void rainIRQ()
//Count rain gauge bucket tips as they occur
//Activated by the magnet and reed switch in the rain gauge, attached to input D2
{
        raintime = millis(); // grab current time
        raininterval = raintime - rainlast; // calculate interval between this and last event
        if (raininterval > 10) // ignore switch-bounce glitches less than 10mS after initial
        {
                dailyrainin += 0.011*25.4; //Each dump is 0.011" of water
                rainHour[minutes] += 0.011*25.4; //Increase this minute's amount
                rain5m[minutes_5m] +=0.011*25.4; // increase this 5 mnts amout
                rainlast = raintime; // set up for next event
        }
        //Rain or not (1 or 0)
        if(rainin_5m >0)
        {
                Rainindi=1;//RainFlag is ON
        }
        if(rainin_5m == 0)
        {
                Rainindi=0;//RainFlag is OFF
        }
}
```

Figure 4: Example of a specifically designed rainfall event code

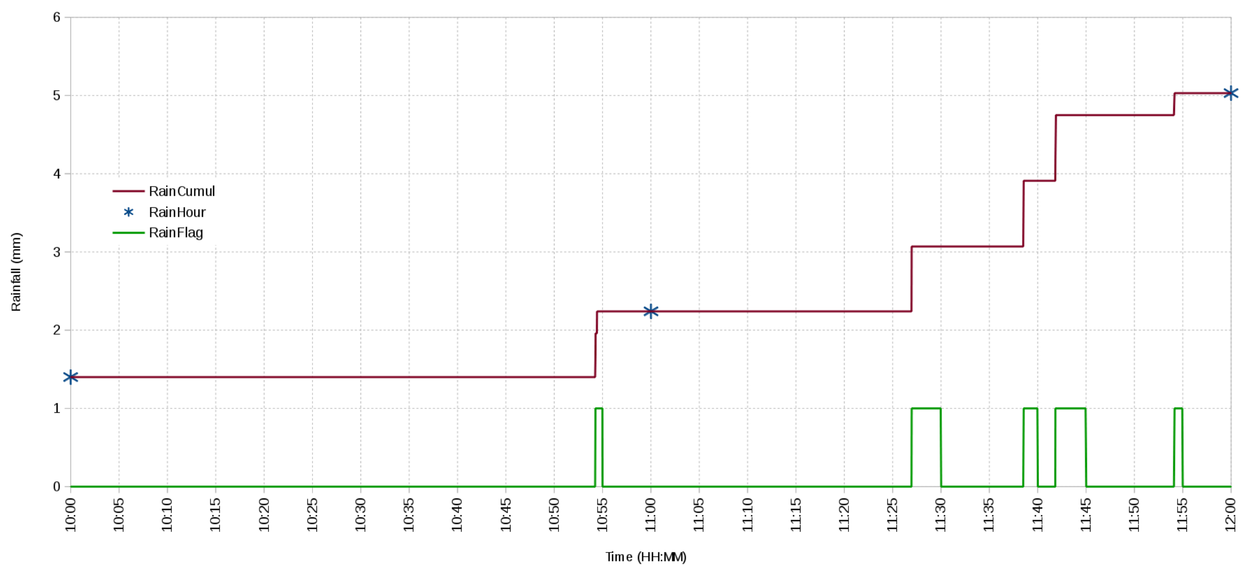

Figure 5: Raingauge Data Analysis

Discussion

Climatic monitoring observations of Sri Lanka use daily rain gauge accumulation, not rainfall intensity of duration. As a result authorized institutions are weak to predict upcoming issues and build up awareness. Daily accumulations are not enough to capture the rate of individual intensity of rainfall because high accumulation can result from a short period of high intensity rainfall, a higher frequency of lower rainfall or combination of these two. There is an arising need for a higher frequency data, to improve statistics for research and accelerate the accumulation of knowledge on extreme weather events. The edge of the interval of data recording is below 30 minutes: 10 minutes or 5 minutes.

Rainwater harvesting depends upon the frequency and amount of rainfall. Therefore detailed statistics of rainfall is essential in the initial stages of the rainwater harvesting system. For instance, the annual rainfall and monthly rainfall is not enough in sizing (storage capacity) a rainfall harvesting system especially with the extreme rainfall pattern due to climate change. The size of the rainwater harvesting system should be different area to area depending of the rainfall pattern, intensity and frequency of the unique locations.

For several weeks, the data was re-evaluated and feed-back converted into structural and software changes. Among the structural changes required is the shape of the raingauge cup, not designed for heavy rainfall, and prone to splashing, reducing the total rainfall recorded. We are in contact with a local manufacturer of micro-electronics to address those changes and increase the percentage of the weather station being made in Sri Lanka.

The requirement to have an efficient monitoring system of rainwater harvesting is for the network of raingauge to be online reporting to a central repository, preferrably open to the public. It turns out that on-going development of the circuits include tests with GSM modules. Online upload of rain data could be done 4 times per day, with each upload reporting hourly rainfall in a ftp site, ready to be used for GIS preparation and modeling, offline or online too.

Conclusion

This research endeavours to demonstrate that Open Design, Open Source Hardware and Free & Open Source Software can be used to taylor build a raingauge meteorological station, with Commodity Off The Shelf elements. The advantages are multiple, the cost is several times less than market options, it records what is of interest to the user only, and prepares analysis on-the-fly as programmed. Availability of parts of the COTS elements is now increasing locally as we speak. Full local manufacture of the whole raingauge is underway, along with an online reporting capability in testing phase.

References

Basnayake, B.R.S.B. (2011) Climate Change in Sri Lanka, Department of Metrology, Sri Lanka [available at http://www.meteo.gov.lk/index.php?option=com_content&view =article&id=73& Itemid=82&lang=en] [viewed on 20/05/2012].

Chandrapala, L. (1996) Long term trends of rainfall and temperature in Sri Lanka. Climate Variability and Agriculture, Y.P. Abrol, S. Gadgil and G.B. Pant, Eds., Narosa Publishing House, New Delhi, 153-162.

De Costa, 2008, Basnayake, 2011, Chandrapala, 1996, Jayatillake et. al, 2005 and Domroes, 1996, Manawadu and Fernando, 2008).

De Costa, W.A.J.M. (2008) Climate change in Sri Lanka: myth or reality? Evidence from long-term meteorological data, Journal of National Science Foundation of Sri Lanka, 36 Special Issue.

Domroes, M., (1996) Rainfall variability over Sri Lanka, Climate variability and Agriculture, 394-410.

Jayatillake, H.M., Chandrapala, L., Basnayake, B.R.S.B. and Dharmaratne, G.H.P. (2005) Water Resources and climate change Proceedings Workshop on Sri Lanka National Water Development Report, eds. Wijesekera, N.T.S., Imbulana, K.A.U.S. and Neupane, B., World Water Assessment Programme, Paris, France.

Manawadu, L. and Fernando, N. (2008) Climate Change in Sri Lanka, [Available at http://archive.cmb.ac.lk/research/bitstream/70130/1100/1/Manawadu_CLIMATE%20CHANGES%20IN%20SRI%20LANKA.pdf] [viewed on 26/05/2012].

Pearce, J.M. (2013a) Share, and We All Grow Richer. The Analytical Scientist. Issue 213, Article 301.

Pearce, J.M. (2013b) Open-source hardware for research and education. Physics Today. Vol. 66(11). pp. 8-9

Pearce, J.M. (2014) Open-Source Lab - How to Build Your Own Hardware and Reduce Research Costs. 1st Edition. Elsevier (Eds). ISBN: 9780124104624

Acknowledgement

The authors would like to acknowledge the CGIAR Research Program on Water, Land and Ecosystems (WLE) & the CGIAR Research Program on Climate Change, Agriculture and Food Security (CCAFS).