Rainfall Intensity Datalogger System. LoPy4-Based Design and Implementation

Sebastian Placencia
jose.placencial@ucuenca.edu.ec
Universidad de Cuenca
Cuenca, Ecuador

Fabian Astudillo-Salinas
fabian.astudillos@ucuenca.edu.ec
Universidad de Cuenca
Cuenca, Ecuador

Andres Vazquez-Rodas
andres.vazquezr@ucuenca.edu.ec
Universidad de Cuenca
Cuenca, Ecuador

Luis I. Minchala
ismael.minchala@ucuenca.edu.ec
Universidad de Cuenca
Cuenca, Ecuador

Jhonny Guaman
jho.guaman@gmail.com
Cuenca, Ecuador

ABSTRACT

The data record of the amount of rain is important for the monitoring and control of watersheds for water and soil conservation. Even more important is to be able to keep this information as up-to-date with a short latency and low cost. This article presents the design and implementation of a rainfall intensity data recording system based on the LoPy4 development board. This implementation integrates the acquisition, storage and wirelessly transmission of the data. There is a data recording system (Station Node) and a transmission forwarding system (Gateway node). In this way, a Wireless Sensors Network (WSN) architecture is achieved by using LoRa technology. This technology offers long range and low energy consumption. The results indicate a strong correlation of the data acquired by the system designed in this article and the data recorded by the reference DAVIS station.

ACM Reference Format:
Sebastian Placencia, Fabian Astudillo-Salinas, Andres Vazquez-Rodas, Luis I. Minchala, and Jhonny Guaman. 2019. Rainfall Intensity Datalogger System. LoPy4-Based Design and Implementation. In *16th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, Ubiquitous Networks (PE-WASUN '19), November 25–29, 2019, Miami Beach, FL, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3345860.3361513

1 INTRODUCTION

In recent years, the more visible effects of global warming, as well as the impact of the man on ecosystems, provoke more awareness in the conservation of ecosystems and especially the water sources. For this reason, the permanent monitoring of watersheds has been strengthened by registering different meteorological variables, such as: solar radiation, temperature, humidity, rain intensity, water level of rivers, speed and wind direction [5]. In addition, meteorological data together with artificial intelligence techniques can be used to predict the flow of rivers [9]. When real-time data is available, one of the applications consists in the development of early warning systems.

PROMAS (**PRO**grama para el **M**anejo del **A**gua y del **S**uelo) is an R+D+i center, oriented through its lines of action, to develop research, development and innovation initiatives related to water and soil management. This center monitors several meteorological parameters. The collected information is used for research purposes. The meteorological information is monitored by 130 stations located in the hydrographic basins of three provinces in Ecuador (Cañar, Azuay, and Chimborazo). There exists three types of stations: meteorological stations (measurement of all meteorological variables), rainfall stations (rainfall intensity measurement), and limnigraphic stations (river flow measurement). Stations from different types has been installed in the national park El Cajas, in southern Ecuador. The monitoring site was declared in 2002 a wetland of international importance. In 2014 was declared by UNESCO as one of the core areas of the biosphere reserve.

Currently, the data gathering from the weather stations including rainfall stations. The stations owned by PROMAS are of two commercial brands: HOBO and Davis. These are installed by PROMAS in several distant locations from its headquarters, is performed by using an old-fashioned methodology which consists in downloading the data *in-situ*. The data is collected from stations every 30 days on average. Unfortunately, this lack of autonomous operation of the stations implies high travelling costs as well as loses in the data when stations run out of power. Further, to improve the accuracy of the prediction system, the collected information has to upload to the data center at least in an hourly manner. The project "Application of wireless technologies to the flow prediction system in the Tomebamba river basin" win the XV contest of the research department of the University of Cuenca. This project seeks to get real-time information from the remote PROMAS meteorological stations. The development of stations are according to the requirements enumerated by [13]. The requirements taken into account for weather stations are the following: (1) real-time transmission; (2) low energy consumption; (3) presentation and ease handling of data; (4) programmable sampling period; (5) portability; (6) variable sampling frequency.

In this work we propose a system for data acquisition of rainfall intensity based on the LoPy4 development board of the Pycom company. The proposed system integrates the acquisition, storage and transmission of rainfall intensity data, hereinafter station. The

---

PE-WASUN '19, November 25–29, 2019, Miami Beach, FL, USA

ACM 978-1-4503-6908-4/19/11…$15.00
https://doi.org/10.1145/3345860.3361513

station is designed to work with double bucket raingauge sensor from any manufacturer.

This paper is organized as follows: Section 2 summarizes the related work. Section 3 describes the scenario and the different station types. Section 4 presents an overview of the proposed system. Section 5 presents practical results from the implementation. Finally, Section 6 presents the conclusions and future works.

2 RELATED WORK

There are several proposals for meteorological data acquisition and transmission systems based on microcontrollers. Some systems include a RF transmission system while others have a GPRS based transmission system[5][15][14]. Another approach is to access the datalogger using a proprietary software and then transmit the data to a server [8]. On the other hand, [4] proposes a meteorological station that has an integrated system in a Zolertia development platform. This platform is based on the system on chip CC2538 ARM Cortex-M3 for data acquisition and on the CC1200 RF transceiver chip for data transmission. In a previous work, the energy consumption of a Raspberry Pi is minimized in order to use it as a remote datalogger [3]. Additionally, it also has been developed a datalogger system for recording water level using the LoPy4 board [10]; this board allows a low energy consumption. As a complement of our previous works here we develop and evaluate a rainfall intensity datalogger system.

3 SCENARIO DESCRIPTION

PROMAS has a total of 130 stations located in the hydrographic basins of three provinces in Ecuador (Cañar, Azuay, and Chimborazo). From the total of stations we have several station clusters. One of the cluster is located in the area of "Loma Larga". This area has been chosen to implement the first WSN prototype. This area includes one meteorological station, one pluviometric (rainfall intensity) station and three limnigraphic stations.

It is important to note that this zone has not any GSM/GPRS coverage. The closest point to the stations with the availability of Internet access is the "Pinos" Camp. The average distance between stations and camp node is 5.33 Km. As we observe in Figure 1, there is not line of sight between camp node and stations. We try to communicate this two station points using one hop LoRa link; but, due to the complexity of the environment and the weather conditions, it was not possible to establish such a LoRa one hop link.

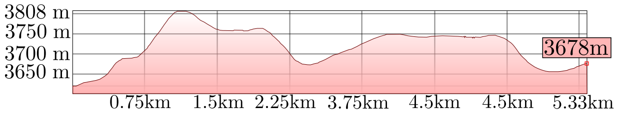

Figure 1: Terrain profile between the camp and the pluviometric station (This image is for test, it has to be update)

We solved this problem by adding an extra node in the network topology. This node acts just as a repeater. The average distance between the repeater node and the stations is 1.68 Km, and the distance between the camp node and the repeater node is 4 Km. Figure 2 shows the terrain profile between the camp node and the repeater node. In turn Figure 3 shows the terrain profile between repeater node and stations. And finally, Figure 4 illustrates the network topology.

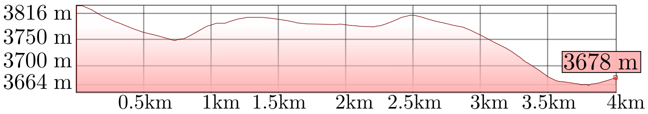

Figure 2: Terrain profile between the camp node and the repeater node (This image is for test, it has to be update)

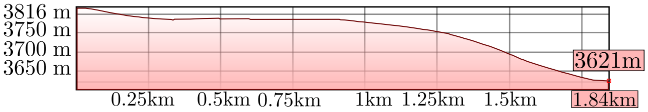

Figure 3: Terrain profile between the repeater node and the pluviometric station (This image is for test, it has to be update)

The limnimeter station has been developed in [10]. The meteorologic station is a work in progress in our project. Finally, the pluviometric station is described in the present work.

4 PROPOSED RAINFALL INTENSITY SYSTEM

The proposed system consists of three subsystems: the power supply subsystem, the datalogger subsystem, and the sensor subsystem. The datalogger subsystem is composed by the processes detailed in Fig. 5. The rainfall sensor used throughout this research is a Davis sensor which is a double bucket rain gauge sensor.

A LoPy4 board (PyCom manufacturer) is used for the system implementation. This board is composed by the Espressif ESP32 dual core chipset and a quadruple bearer radio (LoRa, Sigfox, WiFi, Bluetooth). The WiFi network is used to download the historical data. The data transmission is performed by the stations operating as network nodes, which are part of a wireless sensor network (WSN). The WSN is based on the LoRa technology. Data is transmitted from station nodes to Gateway nodes. The gateway node forwards the information to the data center. The network nodes use the SX1272 module integrated in the LoPy4 development board, this modules work in the 915 MHz ISM band. The embedded system of the gateway node uses a LoPy4 board and integrates a GPRS module for enabling the communication with the data server. In the following we present the datailed description of each subsystem.

4.1 Power supply subsystem

The consumption analysis is performed using the Nguyen methodology [12]. Active times per component (duty cycle) is used to calculate the overall consumption. The time period of the duty cycle is 5 minutes. This time is set by the requirements of PROMAS applications. During this period, the information is transmitted to the data center. The overall consumption is divided into four parts: the datalogger (includes all the components of the Fig. 5), the rainfall sensor, the LoRa module (used for data transmission) and a WiFi module (used for downloading the data *in-situ*).

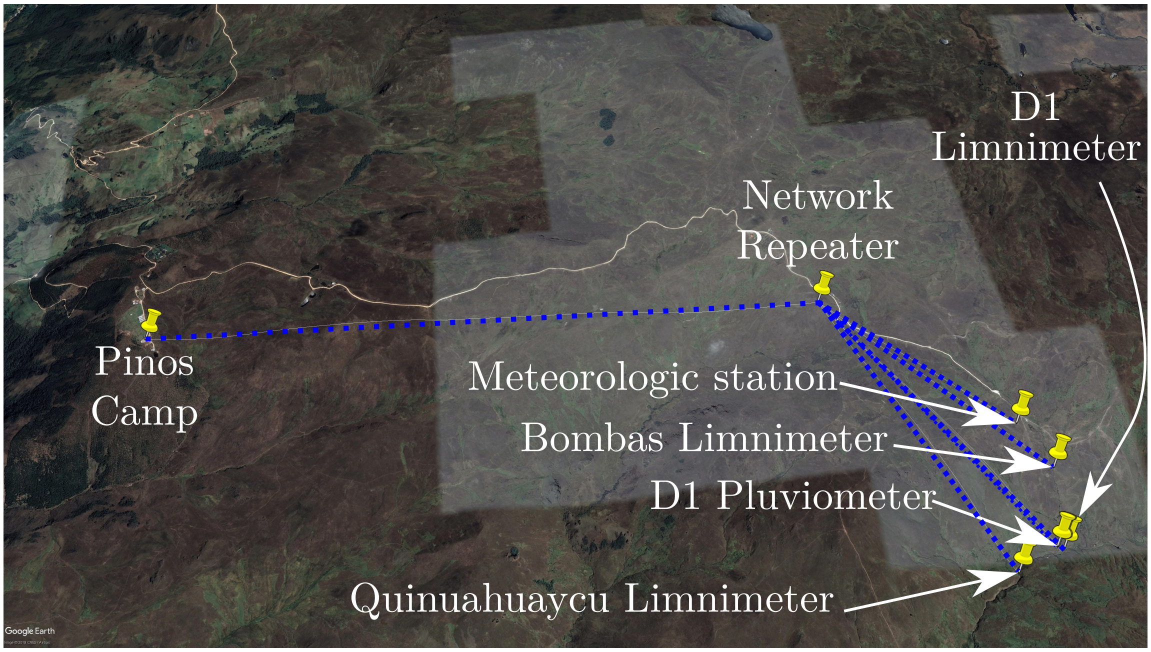

Figure 4: Network topology

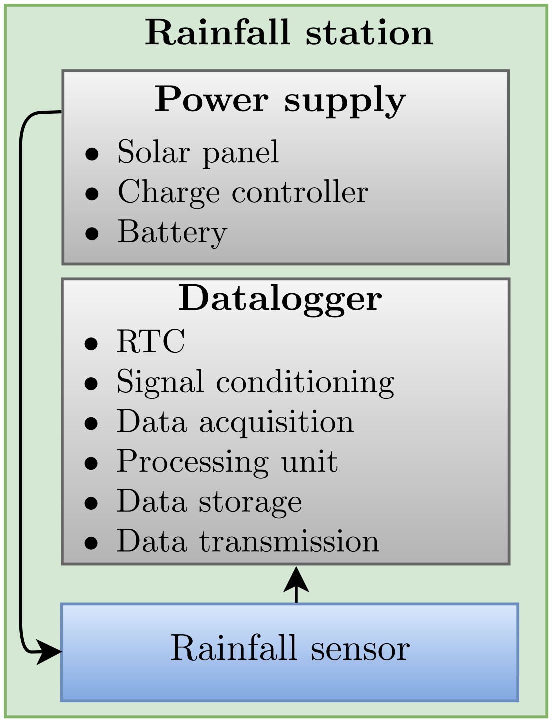

Figure 5: System overview

For the analysis, the datalogger is divided into two parts: acquisition and control. The acquisition process of the rainfall sensors remains active all the time. Therefore, the devices involved in this process have a duty cycle of 100%. The rainfall sensor consumption is 0.03 mA for the maximum measured value. The maximum rainfall at the area from the data registered by PROMAS is 22 mm in an hour (between the years 2015 and 2017), with this value the sensor consumes 0.088 mA. The control process is active during 10 seconds of the time period. It has a duty cycle of 3.34%.

The LoRa transceiver is set with a spreading factor of 12 and a bandwidth of 125 KHz (to maximize the range). The maximum data rate with this configuration is 244 bps. The data frame has 136 bits. The duty cycle for the transceiver is 0.19% by considering a transmission time of 560 ms. The consumption of the WiFi module is negligible, since it is activated when the data is downloaded *in-situ*. Table 1 presents the average consumption of each component of the datalogger and the overall consumption, which is 70.69 mW.

The purpose of the energy supply system is to be self-sustaining and to achieve an autonomy of at least 7 days. The system has a solar panel of 50 W, a load controller based on Maximum Power Point Tracking (MPPT) [6] to take advantage of all the available energy from solar panel and a solid acid battery battery of 12 V - 12 Ah. The solar energy is limited at the location due to adverse weather conditions. This fact implies few hours of solar energy for recharging batteries. Using a 7200 mAh–12 V solid acid battery (minimum load of 25%), the theoretically time for the next recharge cycle is 0.75×7200 mAh$\times 12$ V$/70.69$ mW≈ 916 hours ≈ 38 days.

4.2 Sensor subsystem

The rainfall sensor is based on a cubic spoon container with two ends. When one end is filled, it changes its position from top to bottom, like a seesaw. The process is repeated at the other end. The system delivers an output of electric current each time a container is filled. The electric current is captured as a voltage output. The

Table 1: Power Consumption Estimation

Devices	Mode	Current Draw (mA)	Duty Cycle (%)	Average Current (mA)	Power (mW) ($I \times 12\ V$)
Control	Active	23.04	3.33	3.72	44.64
	Idle	3.05	96.66		
Acquisition	Active	2.08	100	2.08	24.96
	Idle	0.00	0.00		
Sensor	Active	0.088	1.29	0.001	0.012
	Idle	0.00	98.71		
Transmission	Active	52.00	0.19	0.09	1.08
	Idle	0.00	99.81		
Overall Average Power Draw					70.69

output pulse captured corresponds to a measurement of 0.2 mm of rain [7].

4.3 Datalogger subsystem

Conditioning of the signal: There are two signals to be conditioned: the voltage pulses of the sensor and the battery signal. An RC low pass filter is used at the signal output. This configuration avoid false positives. The conditioning take into account the maximum voltage supported by the microcontroller (3.3 V). Meanwhile, the battery signal is conditioned using a voltage divider to the input voltage supported by the microcontroller (2.6 V-3.3 V).

Acquisition: The microcontroller reads the pulses delivered by the signal conditioning system. The reading of the signal is made on each falling flank using an interruption of the microcontroller. The values of the battery voltage, temperature, and humidity are acquired through the LoPy4 board.

Processing Unit: The system has two processing units. The main microcontroller is the ESP32. This is integrated on the LoPy4 board with a firmware developed by using Micropython programming language. It contains an application to send the information through LoRa or WiFi. An additional feature is the deep sleep mode, it is used to reduce the energy consumption. Since, the LoPy4 takes more than one second to restart from deep sleep, we decide to use another processing unit in charge of data acquisition and storage. This processing unit is the PIC18f26k20 microcontroller. The processing tasks are executed every 5 minutes. These tasks include: the calculation functions (maximum, minimum and total of the acquired data), external RTC configuration, LoRa modules configuration, WiFi modules configuration, and finally, the data storage, and generation of transmission frames.

Data storage: The information of the device is stored in three types of files: configuration, temporal and data files.

1. **Configuration file:** It contains the configuration information of the station. The file `rain.cnf` stores the period time value. This is the time interval used to send the information to the server; the time interval by default is 5 minutes.
2. **Data file:** For this data files, the first one stores data from the rain sensor; its prefix is *sensor* (`sensor-[date].dat`). The second file stores status information about battery level, temperature and humidity inside the station; its prefix is *status* (`status-[date].dat`). In the first case `date` is referred to year and month (i.e. `sensor-2019-01.dat`), and in the second case is only referred to the year (i.e. `status-2019.dat`). The records of the sensor are stored by default every 5 minutes (or how it is set in the `rain.cnf` file) and the status are stored every 60 minutes. The file structure of a data file is detailed in Section 4.4.
3. **Temporal file:** It stores the record prior to transmission (`sensor.tmp` and `status.tmp`). Once the record is successfully transmitted its content is deleted. The purpose of this type of file is to have a backup of the records that have not been transmitted due to any eventuality.

Data transmission: It is made using the SX1276 module. This module is integrated on the LoPy4 board. It works in the 915 MHz and 433 MHz bands. The power transmission is set up to 20 dBm. The Spreading Factor used is 12. This configuration uses a bandwidth of 125 Khz, a preamble of 8 symbols and a coding rate of 4/5, and LoRa modulation. An antenna of 2.2 dBi is used.

4.4 File structure for data storage

We define two types of records: data record (Fig. 6) and status record (Fig. 7). The data record has information of the acquired data; this data is stored each five minutes. The status record stores information about the internal condition of the datalogger; the status information is stored hourly. The field `type` (first bit) identifies the type of record. To minimize the amount of stored information, each record can be defined as *all* (0) or *delta* (1) type. The record of type all is stored once a day or when the number of bits of delta information is greater than defined: 8 bits for delta of total field, 5 bits for delta of battery, temperature and humidity fields.

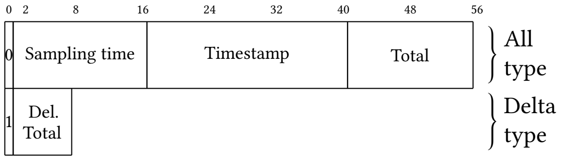

Figure 6: File structure for data file (sensor-*date*.dat)

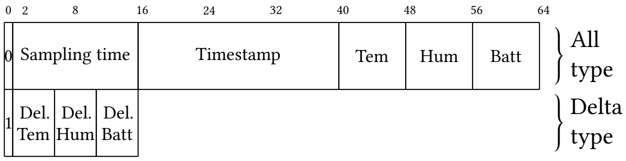

Figure 7: File structure for status file (status-*date*.dat)

4.5 Frame structure for data transmission

The frame is composed by the header, the payload and the cyclic redundancy check (CRC). The header has 16 bits and is composed by three fields: `version`, `StationId`, `SensorId` (see Fig. 8).

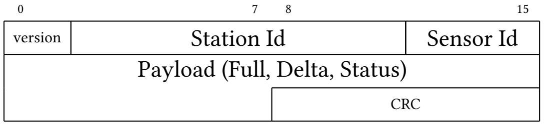

Figure 8: Frame structure

The system addresses up to 1022 stations by means of the gateway. The addressing is given by the field `StationId` (10 bits). The StationId 0 and 1023 are reserved. Each station has up to 14 sensors. These are managed using the field `SensorId` (4 bits). The `SensorId` 0 and 15 are reserved. `SensorId=0` (0000) is used to transmit status information (see Fig. 9). The status information to transmit is the temperature, the humidity and the battery voltage.

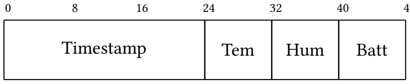

Figure 9: Frame structure for the status transmission

In this work the station has only the rain sensor. This sensor has reserved the `SensorId=2` (0010). The payload depends on the the record type stored in the data file. The payload for the full record contains the `timestamp` (3 bytes) and the total rainfall value (2 bytes). While the delta record includes only the `delta` value (1 byte).

Two types of frames are defined (field `type`). The first type (`0`) is the total rainfall during 5 minutes henceforth full frame. The second type (1) is delta rainfall during the last 5 minutes. Delta is calculated using the total rainfall of last full frame. Fig. 10 shows the structure of this types of frames.

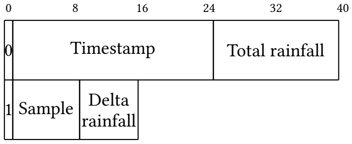

Figure 10: Frame structure for the data transmission.

To reduce the data transmission, a delta frame is sent every 5 minutes after a full frame. The full frame is sent every 255 (8 bits of `sample` field) delta frames, or when nodes are restarted. The maximum change of the rainfall is 12.8 mm (7 bits taking into account the sign bit). If delta is greater than 12.8 mm a full frame is sent.

Nodes are integrated into a WSN to carry out the data transmission to the server. Frames are sent from the nodes to the gateway. Then, these frames are processed by the gateway and forwarded to the server. The network scheme is presented in Fig. 11.

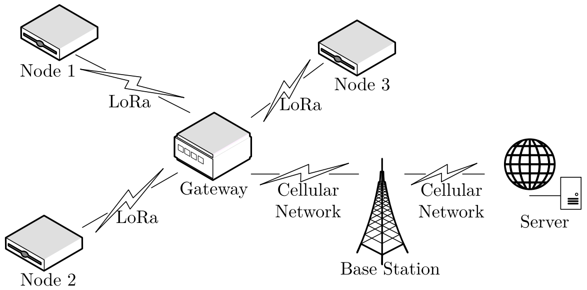

Figure 11: Network scheme for data transmission

5 IMPLEMENTATION AND RESULTS

Figure 12 shows the final design of the PCB. The implementation of the datalogger system is based on the LoPy4 development board of Pycom. The system integrates in one card: (1) the LoPy4 development card; (2) real time clock module (ds3231); (3) remperature-Humidity sensor (DHT11); (4) rainfall data acquisition: RC filter and PIC18f26k20 microcontroller [11]; (5) voltage divider (13.5 to 2.8 *V*); (6) step-Down converter Module of 5 *V*; (7) power Input (Battery).

In the PCB design, the tracks are not below the LoPy4 board to avoid problems with the radio frequency circuits. We had several problems that has been solved during the development. One problem is the malfunction of the WiFi interface; this problem has been solved by designing the hardware without lines and ground plane below the Lopy4 board. The final design of the PCB is available in github[1].

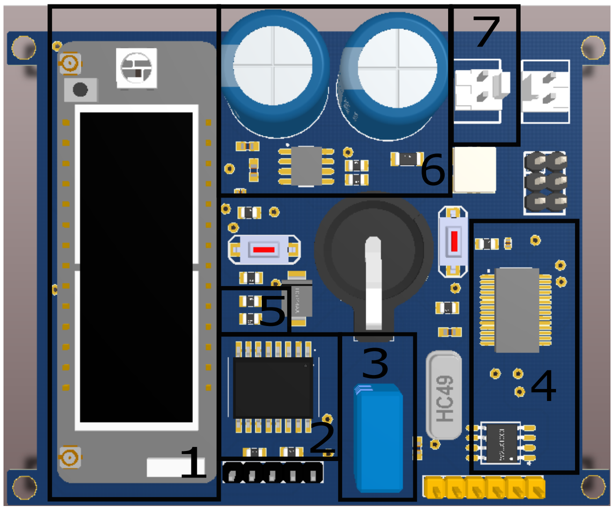

Figure 12: Datalogger prototype

For the implementation it is necessary to have two voltage levels, i.e., 5 V supply used in the LoPy and 3.3 V supply used in the ds3231, the PIC18f26k20, the DHT22[2] and the rainfall sensor. The voltage level supported by the board in its communication pins is lower than 3.3 V. The devices connected to the board must keep their voltage below the aforementioned level.

For the rainfall station implementation, we consider a housing for the elements that make up the system (battery, charge controller and datalogger). The housing is resistant to water and external agents in order to fully protect the system and to avoid errors.

The developed station has been compared to a commercial station. The two stations were located in the same place with a separation of 1 m. The tests were performed for 16 days, of which precipitation was registered in 8 days (see Figure 13). The data between the commercial station and the proposed system are statistically similar, which means that there are a high correlation. The resulting correlation coefficient is $r = 1.0$ (see Figure 14).

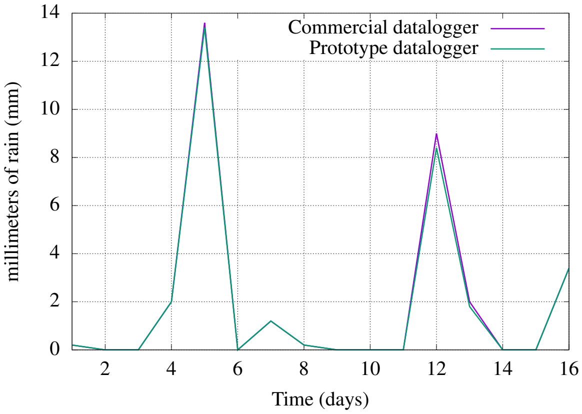

Figure 13: Comparative graph of the rain intensity measurement between a commercial datalogger and the prototype

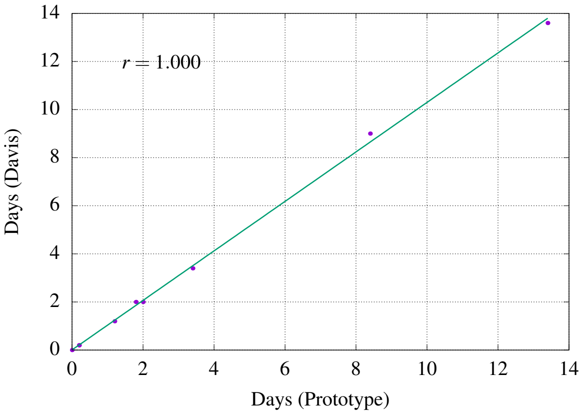

Figure 14: Dispersion chart

6 CONCLUSION

The design and implementation of the datalogger integrating the pluviometric sensor achieves the goal of the project. Different tests have been made and all of them were finished in a satisfactory way. The tests have been carried out both in the laboratory and in the field. In fact, this project uses a generic protocol at the data link layer. In the future version, we will evaluate and compare LoRaWAN and LoRaFABIAN at this layer to use one of them. Both protocols provide the automatic management of Spreading Factor of each station by means of the Adaptive Data Rate (ADR).

ACKNOWLEDGMENT

We acknowledge the financing of *Dirección de Investigación de la Universidad de Cuenca* (DIUC) and *PROgrama para el Manejo del Agua y el Suelo* (PROMAS) of the University of Cuenca.

REFERENCES

[1] 2019. PluviometricUC. https://github.com/dastudillo1983/PluviometerUC.git
[2] Ltd Aosong(Guangzhou) Electronics Co. [n.d.]. Digital-output relative humidity humidity & temperature sensor/module AM2303. , 11 pages.
[3] F. Astudillo-Salinas, D. Barrera-Salamea, A. Vázquez-Rodas, and L. Solano-Quinde. 2016. Minimizing the power consumption in Raspberry Pi to use as a remote WSN gateway. In *LATINCOM 2016*. 1–5. https://doi.org/10.1109/LATINCOM.2016.7811590
[4] Marc Fàbregas Bachs. 2016. Diseño e implementación de una estación meteorológica de bajo coste con conectividad a Internet. In *2017 29th Chinese Control And Decision Conference (CCDC)*.
[5] M. Benghanem. 2009. Measurement of meteorological data based on wireless data acquisition system monitoring. In *2009, Applied Energy*. 2651–2660. https://doi.org/10.1016/j.apenergy.2009.03.026
[6] A. Bidram, A. Davoudi, and R. S. Balog. 2012. Control and Circuit Techniques to Mitigate Partial Shading Effects in Photovoltaic Arrays. *IEEE Journal of Photovoltaics* 2, 4 (Oct 2012), 532–546. https://doi.org/10.1109/JPHOTOV.2012.2202879
[7] Onset Computer Corporation. 2005–2017. HOBO Data Logging Rain Gauge. , 6 pages.
[8] Leticia Rodríguez y Carlos Vionnet Emiliano López. 2014. Captura y transmisión automática de datos hidro-meteorológicos. *Maestrıa en computación aplicada a la ciencia y la ingenierıa, Facultad de Ingenierıa y Ciencias Hıdricas, UNL* (2014).
[9] Cisneros Felipe, Veintimilla Jaime, and Vanegas Pablo. 2013. Artificial neural networks applied to flow prediction scenarios in Tomebamba River-Paute watershed, for flood and water quality control and management at City of Cuenca Ecuador. In *EGU General Assembly Conference Abstracts*, Vol. 15.
[10] Jhonny Guaman, Fabian Astudillo-Salinas, Andres Vazquez-Rodas, Luis I Minchala, and Sebastian Placencia. 2018. Water Level Monitoring System Based on LoPy4 Microcontroller with LoRa technology. In *INTERCON 2018*. IEEE, 1–4.
[11] Microchip Technology Inc. 2010–2015. PIC18F2XK20/4XK20. , 440 pages.
[12] T. D. Nguyen. 2016. Energy efficient wireless sensor network and low power consumption station design for an urban water level monitoring system. In *3rd NICS*. 252–256. https://doi.org/10.1109/NICS.2016.7725660
[13] Fiona Regan, Antoin Lawlor, Brendan O Flynn, J Torres, Rafael Martinez-Catala, C O'Mathuna, and John Wallace. 2009. A demonstration of wireless sensing for long term monitoring of water quality. In *IEEE 34th LCN*. 819–825. https://doi.org/10.1109/LCN.2009.5355047
[14] A. Adane S. Ameur a, M. Laghrouche a. 2001. Monitoring a greenhouse using a microcontroller-based meteorological dataacquisition system. In *2001: Renewable Energy*. 19–30.
[15] F.J. Batlles * S. Rosiek. 2008. A microcontroller-based data-acquisition system for meteorological station monitoring. In *2008, Energy Conversion and Management*. 3746–3754. https://doi.org/10.1016/j.enconman.2008.05.029