Design and implementation of a LoRa based wireless control for drip irrigation systems

Conference Paper · December 2017

DOI: 10.1109/ICRAE.2017.8291389

Maqsudjon Usmonov
Politecnico di Torino

Design and Implementation of a LoRa Based Wireless Control for Drip Irrigation Systems

Maksudjon Usmonov, Francesco Gregoretti
Department of Electronics and Telecommunication
Politecnico di Torino
Corso Duca degli Abruzzi, 24, 10129, Torino, Italy
e-mail: francesco.gregoretti@polito.it, maksudjon.usmonov@polito.it

***Abstract*—The deployment of drip irrigation systems has been prevailing over the last few decades and it poses many difficulties to manual control them once they are installed on a large scale. A common approach to overcome this issue is to set up a wireless sensor and actuator network (WSAN). However, using wireless technologies in the field of agriculture presents several technical challenges, such as achieving long battery life, long-range capabilities and low-cost at the same time. In recent years, many technologies and protocols have been deployed trying to overcome those challenges. One of them is LoRa, proprietary wireless modulation technology. The network protocol (LoRaWAN), which is built on top of Lora modulation, has been studied and analyzed as part of this paper. Considering the cost and the complexity of setting up LoRaWAN, this work proposes a simpler and cost-effective protocol designed specifically to control drip irrigation systems. The project that is described in this paper includes the implementation results of both hardware and software for the wireless nodes, and development of GUI app for controlling drip irrigation systems.**

Keywords-component; drip irrigation; lora; LoraWAN; WSAN; low power

I. Introduction

A. Basics of Drip Irrigation

Drip irrigation is a type of microirrigation that has gained a great attention in recent years due to its potential to increase yields and to decrease water use. Water is distributed through a network of valves, pipes, tubings, and emitters and it is then dripped slowly, but directly to the root zone of the plant either from above or below the soil surface.

Figure 1 shows the general layout of typical drip irrigation system and its main components [1]. At first, a pump pushes the water to the distribution network at high pressure. Then water goes through the fertilizer solution tank and filtering takes place to prevent emitters clogging up. Filtered water flows through the main pipe and valves control its distribution to emitters of the particular area.

The typical lifetime of a drip irrigation system is about 5-10 years, once it is installed and annual maintenance cost is about 3 % of the installation cost on average [2]. This method of irrigation is preferred, because it reduces water usage substantially, works on almost any terrain and has the possibility to be automated eliminating human interaction completely or partially.

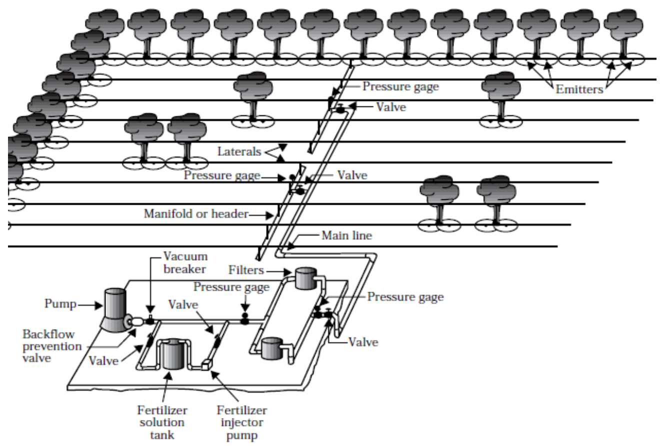

Figure 1. Typical drip irrigation system layout [1].

B. Related Work

The uses of computers and electronics in the field of agriculture, specifically, in the irrigation systems have created new engineering and research challenges. In particular, wireless control of actuators for agricultural purposes has some technical difficulties because of the limited budget and power resources. However, in recent years, many different technologies have been developed to efficiently set up WSANs (Wireless Sensor and Actuator Networks) [3]. And many studies are conducted to examine their impact on transforming the agriculture [4].

Over the years, technologies, such as ZigBee™ and Bluetooth, have been dominant to establish low-power, short-range, multihop networks [5,6,7], which exploits the mesh network topology. Although these standards are considered low-cost systems, their limited coverage (~100 meters) is a major drawback, that makes them difficult to be deployed in large-scale irrigation systems. On the other hand, cellular networks, such as GSM or LTE, are capable of providing long range transmission to form WSANs, and they have been successfully tested to control irrigation systems [8], but solar panels are required for each node to compensate higher power consumption of cellular network.

Also, its dependence on the availability of mobile network puts it under question for some remote rural areas.

An alternative solution for establishing long-range, low-power and low-cost WSANs is the low-rate transmission technology, referred to as LPWAN (Low Power Wide Area Network). The main differences between LPWANs and the previous technologies are the use of long-range radio links, deployment of the star network topologies and low-rate data transmissions. Sigfox, Ingenu, NB-IoT, DASH7, and LoRaWAN are examples of LPWAN [9]. All of those technologies have coverage distance of several kilometers and have their own advantages and shortcomings [10], in terms of the cost, scalability, power consumption, data rate and etc.

Since the wireless control of drip irrigation requires very small data exchange, any of these network types can be used. Among them, Lora is relatively new technology on top of which the LoRaWAN protocol operates. It has the highest radio link budget and the best "cost vs. range vs. power" tradeoff among its competitors [10].

That is why, for this project, LoRa modem has been chosen as a radio link.

II. Transmission Protocol

A. *Overview of Lora Physical Layer*

Lora is proprietary spread spectrum modulation technique that is based on Chirp Spread Spectrum modulation (CSS) and designed by Semtech [11]. LoRa modems work at unlicensed frequency bands, usually from sub-GHz category.

LoRa modulation uses frequency chirps with a linear variation of frequency over time in order to encode information and trades data rate for sensitivity within a fixed channel bandwidth [12-13]. It implements a variable data rate, utilizing the adjustable parameter, so-called spreading factor, which enables network performance to be optimized in a constant bandwidth by trading the data rate for range or power [14]. Thanks to the linearity of the chirp pulses, frequency offsets between the receiver and the transmitter are equivalent to timing offsets, and they are easily removed in the decoder. This also increases the immunity to the Doppler effect, which in turn is equivalent to a frequency offset. Decoding can tolerate an offset up to 20% of the bandwidth without impacting the performance. As a result, crystals in the transmitters do not need to be manufactured to extreme accuracy, and it leads to the cost reduction of LoRa transmitters. LoRa receivers can lock on to the frequency chirps received with a sensitivity of the order of -130 dBm [15]. The typical out-of-channel selectivity (the maximum ratio of power between an interferer in a neighboring band and the LoRa signal) is 90 dB and co-channel rejection (the maximal ratio of power between an interferer in the same channel and the LoRa signal) of LoRa receivers is 20 dB [11]. Compared to traditional modulation schemes, such as Frequency-Shift Keying (FSK), LoRa modulation excels at low-power and long-range transmissions. Also, LoRa exploits the variable error correction technique that improves the robustness of the transmitted signal at the expense of redundancy.

LoRa modulation bitrate R_b is defined as:

$$R_b = SF * \frac{\left[\frac{4}{4+CR}\right]}{\left[\frac{2^{SF}}{BW}\right]}\ bits/seconds$$

Where the different parameters are:

SF – spreading factor (7…12);

BW – bandwidth (Hz)

CR – coding rate (1…4)

All these parameters are adjustable in Semtech's LoRa modules, so that designer can choose the most suitable settings.

For low power application scenarios, duty cycled reception can be configured with Channel Activity Detection (CAD) mode of the LoRa modem. In CAD mode, modem alerts companion microcontroller (MCU) as soon as it detects the presence of the preamble. This permits to decide whether radio module should go back to sleep mode or remain awake and continue demodulation. Once MCU puts the module into CAD mode, LoRa switches to CAD receiver phase and starts to look for the correct preamble in the configured channel. After some time, it goes to CAD processing phase and triggers *CadDone* interrupt, then automatically puts itself to Standby mode. If the preamble has been detected, *CadDetected* interrupt is also triggered and microcontroller is responsible to put the module into receiver mode. In the case of preamble absence, it is put into sleep mode. Figure 2 shows current consumption level for one cycle of CAD operation [16].

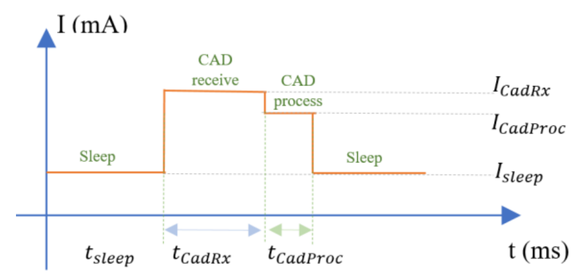

Figure 2. LoRa CAD timing.

Time and current values depend on the configured settings of the LoRa. They can be calculated as:

$$t_{CadRx} = \frac{32}{BW} + \frac{2^{SF}}{BW}$$

$$t_{CadProc} = \frac{SF * 2^{SF}}{1.75 * 10^6}$$

B. *LoRaWAN overview*

LoRaWAN is a medium access control (MAC) layer protocol that uses LoRa modules as a radio link between wireless nodes. Hence, LoRa is referred to as the "Physical layer" in the Open Systems Interconnection model (OSI model), whereas the LoRaWAN is the "Data-link layer".

LoRaWAN is implemented as "star-of-stars" topology, which consists of "end devices", "gateways (i.e. base stations)" and the "network server", as described in Figure 3.

End-devices are the low-power consumption nodes that communicate with gateways using LoRa.

A *Gateway* forwards packets coming from end-devices to a network server over an IP backhaul interface allowing a larger throughput, such as Ethernet or 3G. Multiple gateways can be deployed in a LoRaWAN, and the same data packet can be received (and forwarded) by more than one gateway.

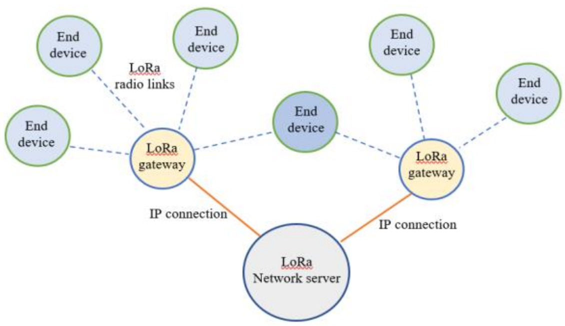

Figure 3. LoRaWAN system architecture.

The *Network Server* decodes the packets sent by the end-devices, removes duplicated ones, and generates reply packets though the proper gateway [17].

Gateways must be able to receive messages simultaneously from multiple channels, with different data-rate and variable payload length. This means that, gateways have to be equipped with multichannel receiver module, and this makes them more expensive than other parts of the system. By using additional gateways, network coverage can be extended, when single-hop transmission range is not sufficient.

LoRaWAN has three different classes of end-devices to address the various needs of applications named as A, B, C.

The most power efficient type is Class A, where each uplink (from end-device to gateway) transmission is followed by two short downlink receive windows (from gateway to end-device). End-devices of this class are suited for the use in the system which has been designed for irrigation control. However, using the standard LoRaWAN protocol [17] requires setting up gateways, and network servers. Moreover, in order to connect multiple gateways to the network server, another kind of high throughput connection like 3G, is needed and the overall system cost will increase.

Therefore, the cost is decreased by simplifying the data transfer protocol at the expense of a lower flexibility.

C. *The deployed data transfer protocol*

In order to eliminate the need for gateways and network servers, a master (central) station has been designed to relay packets between the control application in the PC and end-device nodes.

Figure 4 shows the overview of this simplified low-cost communication architecture.

Unlike gateways in LoRaWAN, the master station doesn’t have to handle simultaneous multichannel reception because transmissions are always initiated by the master, and the nodes send back data only when requested. If the irrigation control of larger areas is required, a longer communication range can be achieved by repeater nodes. In this architecture, all the nodes can act as a repeater, eliminating the need for separate repeater modules or additional gateways as in LoRaWAN. The routing path of the packet is decided in the central PC, and added to the packet itself, so that intermediate nodes may know where to forward the packet. The flow of the operations of the end-device is given in Figure 5.

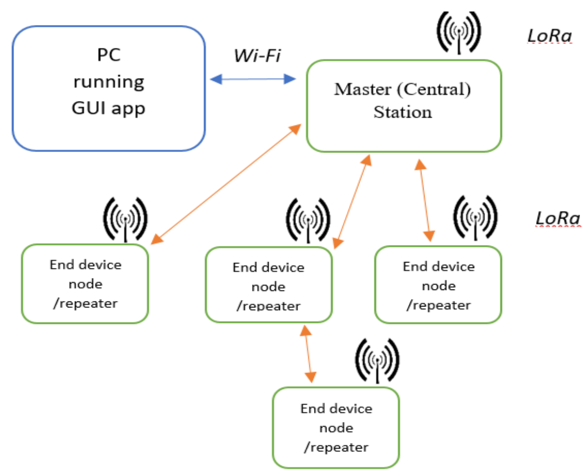

Figure 4. Communication architecture designed for the project.

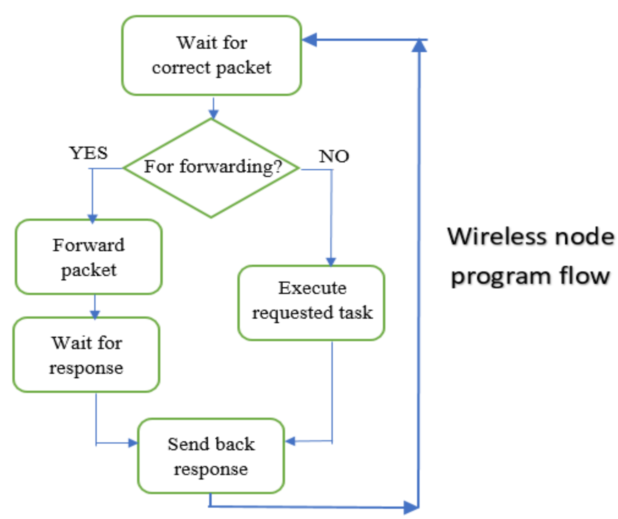

Figure 5. Wireless node program flow.

III. Objectives

The main goal of the project has been to design low-cost system for controlling a drip irrigation process such as the one shown in Figure 1. It is also crucial to maintain the reliable performance, easy installation and use, even for large-scale deployments. The main tasks have been:

- The design and implementation of the hardware wireless control nodes, capable of both monitoring the environmental data and controlling the actuators;
- The development of the firmware for the nodes, that satisfies low-power requirements.
- The development of the GUI application for controlling the irrigation process from one central point.

The main specifications were:

- A minimum of two years of battery life, assuming that each wireless node receives commands 4 times a day on average and have four D-size, typical alkaline, non-rechargeable batteries installed.
- A minimum 2 km range with single transmission, and its extension by deploying repeaters.
- A lower cost and an easier setup than a LoRaWAN network.
- A maximum of 1000 different nodes in one system
- Up to 4 actuators independently controlled in each node in order to reduce the cost for actuators which are placed close to each other.

A. *System components*

The control of the drip irrigation process includes operating the actuators, which in turn changes water distribution valves accordingly. Since high pressure and high-volume water flow should be controlled by battery powered, low-energy units, choosing the right type of valve is crucial.

Internally piloted solenoid valves [18], operated by latched solenoids, are the most common choice to control high pressure with low power. A latched solenoid usually contains a permanent magnet to keep the armature position once it has been changed, so that no power is required to maintain the state. Position can be changed by applying a short-pulsed voltage of opposite sign to the coil. Standard latched solenoids used in the system, require pulsed signal of 24V with a duration of 50 milliseconds. The coil resistance is of 9 Ω.

Hence, every wireless node should be able to generate stable 24 volts from a lower supply voltage, when necessary, providing sufficient current without sacrificing overall low-energy consumption.

IV. System Design

In order to meet with the requirements, components have been chosen considering their low-cost and low-power characteristics, without sacrificing the reliability. Each board consists of the following functional blocks, shown in Figure 6.

A. *Component Selection*

Ra-02 is a RF module that is based on the "Sx1278" LoRa chip designed by Semtech [14] and operates in the 433 MHz frequency band. It communicates with the microcontroller (MCU) through SPI interface. The MCU, is the *Atmega328P* and has been selected based on its low power consumption and its low-cost [19].

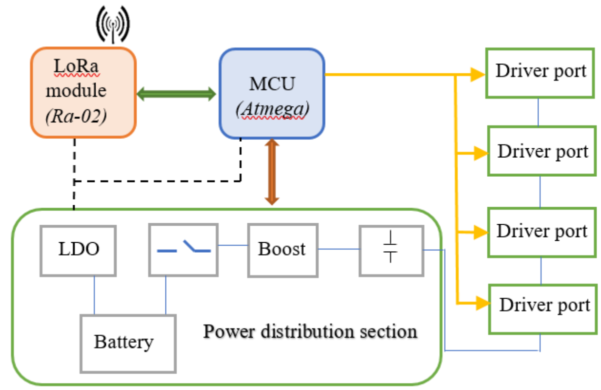

Figure 6. Board architecture of wireless end-device.

The power section comprises 4 D-size batteries in series which gives a 6V supply and a Low-dropout linear voltage regulator (LDO) [20] with 3.3V fixed output which powers the MCU and the LoRa module which may operate in the 1.8V ÷ 3.4V voltage range. In addition, a boost converter provides the 24 V supply to power the actuators. The energy required to change the position of latched solenoid is given, neglecting the reactive components, by:

$$E_{\substack{state\\change}} = \frac{V_s^{\,2}}{R_s} * T_s = \frac{(24V)^2}{9\ \Omega} * (50\ ms) = 3.2\ J$$

Providing this energy in a short time pulse requires a current I=24V/9Ω=2.7A and an electrical charge $q=I*T_s$=108 mC must be provided. Taking the power directly from the alkaline batteries would cause voltage drops leading to system malfunctions since they may not provide high peak currents. Therefore, a tank capacitor is placed at the output of the boost converter to be charged in a longer time and to release the current pulse when required. The minimum value of the capacitor is:

$$C = \frac{q}{V_s} = \frac{108\ mC}{24\ V} = 4.5\ mF$$

A large electrolytic capacitor with a nominal value of 4700µF has been chosen. However, another problem arises with boost converter circuit when large output capacitor is used. Start-up current of the converter will be very high, and most small boost converter ICs may fail. The solution which has been chosen was to use a converter with built-in soft-start functionality, such as the IC (*TPS61175)* from Texas Instruments [21] which increases output voltage gradually, so that output capacitor gets charged steadily. The MCU turns the converter off when not used, with the help of the high-side transistor switch.

The driver board has been designed separately, and it can be connected to the main board through any of the dedicated driver port. Basically, it is an H-bridge with at least 2.7A peak current, and operating with 3.3V logic level signals. The *DRV8801 IC* [22] fully satisfies the requirements.

B. *Component selection*

The same PCB board has been used both for the master and for the end nodes. The former mounts, in addition, the *ESP-01* (WiFi) module and the functions of the two types of boards are determined by the firmware written to the microcontroller.

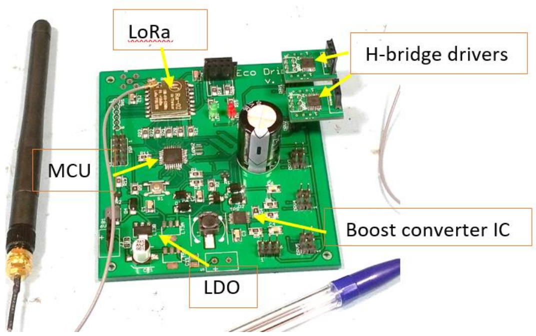

Figure 7. Assembled PCB of the wireless end-device (slave node).

Figure 7 shows an end-device (slave node) board with all components mounted, including two piggy-back driver modules, and figure 8 shows the master board with the WiFi module added and without the boost converter and the tank capacitor.

The Master module creates the WiFi Access Point (AP), receives the packets from computer, and relays them to the LoRa network, and vice versa.

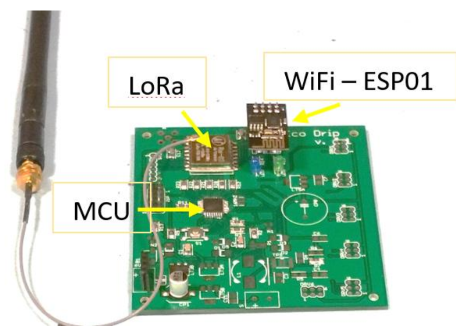

Figure 8. Assembled PCB for master (central) node.

C. *Estimated Energy Consumption*

Energy storage of one typical D-size alkaline battery is about *50000 J* [23]. However, considering self-discharge rate of approximately 3% per year, and 10% residual, only about 85% of initial energy can be used in two years of battery life, that is *42500 Joules* for one D-size battery. In this project, 4 batteries of this type has been deployed, so one node has *170000 J* of battery energy that should last for at least 2 years.

Assuming a number of daily activations of 4, and the overall efficiency of power section of roughly 80%, the total energy spent for controlling a single solenoid actuator every day is given by:

$$E_{control} = 4 * 1.25 * E_{\substack{state \\ change}} \approx 16\, J/day$$

In order to save power, end-devices use duty cycled CAD reception mode with pre-defined sleep period of *750ms*. Preamble duration of the packet must exceed sleep period for correct reception. Packet payload is maximum 9 bytes, containing network ID, destination address, source address, command, output port selection, and optionally at most 3 repeater addresses. Since the system may contain maximum 1000 nodes, 10 bits of address field is sufficient. Response packet *(4 bytes payload)* is sent back immediately after execution of given command. The sender radio does not go to sleep mode until it receives the response or gets a timeout and therefore minimum preamble length is sufficient for response packet.

As a result, total time-on-air for both command and response packets are *970ms* and *231ms* respectively.

Table 1 shows time durations and estimated energy consumption for single radio operations, with LoRa configured with *BW=125KHz, SF=10, CR=4/8, TX_power=+20 dBm.*

TABLE I. ENERGY CONSUMPTION FOR SINGLE CAD OPERATION, ONE RECEPTION AND ONE TRANSMISSION

Operation	Current draw	Duration	Energy consumption (with $V_{battery} = 6V$)
CAD RX	11.5 mA	8.5 ms	0.59 mJ
CAD Processing	6.5 mA	5.9 ms	0.23 mJ
LoRa standby	1.8 mA	1.6 ms	17 µJ
LoRa sleep	1 µA	750 ms	4.5 µJ
MCU awake	5 mA	16 ms	0.48 mJ
MCU sleep	10 µA	750 ms	45 µJ
Single RX	16.5 mA	970 ms	96 mJ
Single TX	125 mA	231 ms	173 mJ
LDO regulator	12 µA	*always*	6.22 J/day

Considering 4 interrogations per day, the estimated daily energy consumption is equal to :

$$E_{daily} \approx 177.5\, J/day$$

Hence, *170 kJ* of total energy available in the batteries lasts approximately 958 days, which means more than 2 years of battery life.

V. GUI APPLICATION

An application with graphical user interface has been developed specifically to control a drip irrigation process. Kivy, python framework for multitouch GUI application development [24], has been used to build it. Since it is a cross-platform framework, the application can be installed on different types of environment without changing the source code. The app has been tested on Linux and Windows. In this first version of the application, user can add, edit or remove the information of solenoid valves, such as its address and output port through which the valve is controlled. In the app, wireless end-devices are categorized into two types: Solenoid valve controller and Pump controller. Once the pump information is entered, it is controlled automatically according to solenoid valves states. For example, if user wants to turn off all the valves, first, app automatically turns off the pump, and sends "close" commands to other nodes. This ensures the safety of irrigation process, preventing pump operation when all valves are closed. Figures 9 and 10 shows some screenshots of the application.

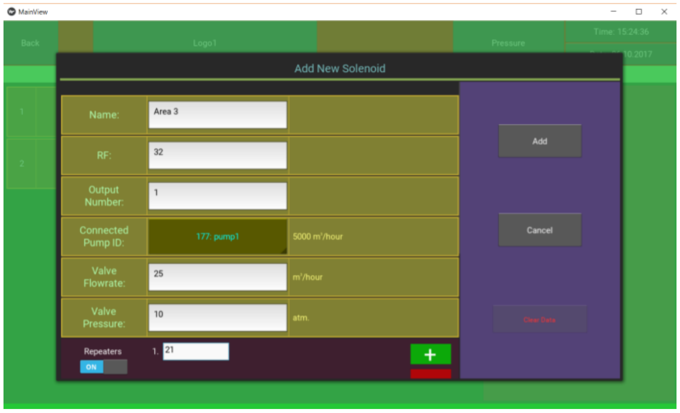

Figure 9. Adding new solenoid.

Figure 10. Control panel of all solenoid valves.

VI. Conclusion

In this paper, the solution using LoRa technology for cost-effective wireless control of drip irrigation systems has been presented. The system which utilizes LoRa modules to establish reliable radio link has been designed, and customized data transfer protocol that satisfies the requirements has been deployed. It is shown that this solution has the advantages over existing LoRaWAN protocol in terms of cost and complexity for this specific application.

In the future, it is planned to include sensors to monitor the environmental data, such as soil moisture and temperature, and develop the control system that is able to take automatic decisions based on the collected data.

Acknowledgment

This project has been supported by Politecnico di Torino, Italy and "Eco Drip Lux Ltd.", Tashkent, Uzbekistan.

References

[1] Irrigation Guide, United States Department of Agriculture (USDA), 1997, pp. 6-60, fig. 6-14

[2] Irrigation Guide, United States Department of Agriculture (USDA), 1997, pp. 5-5, table 5-2.

[3] Buratti, Chiara, et al. "An overview on wireless sensor networks technology and evolution." *Sensors* 9.9 (2009): 6869-6896.

[4] Wark, Tim, et al. "Transforming agriculture through pervasive wireless sensor networks." *IEEE Pervasive Computing* 6.2 (2007).

[5] Keshtgari, Manijeh, and Amene Deljoo. "A wireless sensor network solution for precision agriculture based on zigbee technology." *Wireless Sensor Network* 4.1 (2012): 25.

[6] Mackensen, Elke, Matthias Lai, and Thomas M. Wendt. "Bluetooth low energy (BLE) based wireless sensors." *Sensors, 2012 IEEE*.

[7] Khurshid Aliev, Rugiano Francesco, and Pasero Eros. "Smartphone and Bluetooth Smart Sensor Usage in IoT Applications." *Sensors & Transducers* 201.6 (2016): 27.

[8] Guti érez, Joaqu ín, et al. "Automated irrigation system using a wireless sensor network and GPRS module." *IEEE transactions on instrumentation and measurement* 63.1 (2014): 166-176.

[9] Centenaro, Marco, et al. "Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios." *IEEE Wireless Communications* 23.5 (2016): 60-67.

[10] de Carvalho Silva, Jonathan, et al. "LoRaWAN—A low power WAN protocol for Internet of Things: A review and opportunities." *Computer and Energy Science (SpliTech), 2017 2nd International Multidisciplinary Conference on. IEEE*, 2017.

[11] "LoRa™ Modulation Basics, Application Note." *Semtech, AN1200*, Rev.2, May 2015 http://www.semtech.com/images/datasheet/an1200.22.pdf

[12] Knight, Matthew, and Balint Seeber. "Decoding LoRa: Realizing a Modern LPWAN with SDR." *Proceedings of the GNU Radio Conference*. Vol. 1. No. 1. 2016.

[13] Knight, Matthew, and Balint Seeber. "Decoding LoRa: Realizing a Modern LPWAN with SDR." *Proceedings of the GNU Radio Conference*. Vol. 1. No. 1. 2016.

[14] "LoRa SX1276/77/78/79 Datasheet", Rev. 4. *Semtech*, 2015. http://www.semtech.com/images/datasheet/sx1276_77_78_79.pdf

[15] Augustin, Aloÿs, et al. "A study of LoRa: Long range & low power networks for the internet of things." *Sensors* 16.9 (2016): 1466

[16] "LoRa Modem: Low Energy Consumption Design" *Semtech, AN1200.17*, Rev.1, July, 2013. http://www.semtech.com/images/datasheet/LoraLowEnergyDesign_STD.pdf

[17] *Alliance, LoRa*. "A technical overview of LoRa and LoRaWAN." White paper, Nov (2015). https://www.lora-alliance.org/lorawan-white-papers

[18] "Technical Principles of Valves", Technical reference, *Omega*, https://www.omega.com/green/pdf/valve_tech_ref.pdf

[19] *Atmega328, Atmel*. "P datasheet complete, 2016-06 [Internet] URL: http://www.atmel.com/Images." Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_datasheet.pdf *[Hämtad: 2016-09-29]*.

[20] *Linear Technology*. "LT1521 low-dropout regulator with micropower quiescent current". http://cds.linear.com/docs/en/datasheet/1521335fb.pdf

[21] *Texas Instruments*, "TPS61175 – High voltage boost converter", SLVS892D, December 2008–Rev. April 2016 http://www.ti.com/lit/ds/symlink/tps61175.pdf

[22] *Texas Instruments*, "DRV8801 – DMOS Full-bridge motor drivers", SLVS855J, July 2008, Rev. March 2015. http://www.ti.com/lit/ds/symlink/drv8801.pdf

[23] " Datasheet of D-size alkaline-manganese dioxide battery", *Duracell Technical Library - MN1300*, https://www.duracell.com/en-us/techlibrary/product-technical-data-sheets

[24] "Kivy: Cross-platform Python Framework for NUI Development", https://kivy.org/