Using Online Remote Laboratory in Agriculture Engineering and Electronic Training

https://doi.org/10.3991/ijoe.v15i06.9699

Abdelmoula Abouhilal(✉), Amine Moulay Taj,
Naîma Taifi, Abdessamad Malaoui
Sultan Moulay Slimane University, Béni Mellal, Morocco
a.abouhilal@usms.ma

Abstract—Practical manipulations are a core part of engineering training education systems. Remote labs are a new method used for teaching and practicing experimental manipulation using the performance of information and communication technologies. This paper presents a study of two remote labs architecture using low cost embedded systems that could be addressed to the 3rd year bachelor degree students on renewable energy and others on electronics courses. The first manipulation is based on Arduino microcontroller to monitor an irrigation system powered by photovoltaic panels. In addition, the second manipulation uses a powerful PcDuino, to control remotely a logic electronic experience. A simple interface is developed to allow students and instructors to access to these manipulations. This study is aimed to improve the present education systems in the Moroccan universities by managing the practical manipulation for a large number of students, especially in the open-access faculties. Finally, this architecture can be easily extended to other disciplines and courses.

Keywords—Remote labs, agriculture engineering, embedded systems, remote training, online manipulation, e-learning.

1 Introduction

Advancement of Information and Communication Technologies (ICT) has been open a new trend of online manipulations using the internet in recent years [1-3]. Web technologies and communication protocol play an important role to share resources in the net. Unlike hands-on manipulations, remote experiments allow students to manipulate via internet online experience that can be found anywhere in the world [4, 5]. Because of the large number of registered students in Moroccan universities, and due to lack of staff, remote labs become a real need to overcome the problems of hands-on PW that can go up to its absence in the open-access faculties [6]. Moreover, not only to the shortage of equipment and tutors, but also to the high expenses of the traditional practical works (PW). The proposed approach in this study is inspired by the European project EOLES [7, 8], and based on the implementation of low-cost Practical Work integrated into remote labs based embedded systems to the multidisciplinary faculty of Beni Mellal [9].

Remote laboratories can offer many advantages and benefits [10-12]:

- **Safety:** In the case of dangerous experimentations, remote labs provide a safe alternative to hands-on labs.
- **Accessibility:** Handicapped people have also the possibility to manipulate electrical experiences.
- **Availability:** Users can be connected from anywhere and at any time.
- **Observability:** The session of the experience can be watched by several students/tutors or even recorded.

In this paper, we propose two architectures of remote experiences, the first based on an Arduino board [13–15] to control a solar irrigation system pumping integrating artificial intelligence whose results is discussed above and it is developed to test the optimization of irrigation using solar pumping [16, 17]. We test 2 ways of irrigation through nonlinear algorithms. The second based on a mini PC Pcduino to control an experience of digital electronics. These manipulations can be used by a large number of students of the 3rd year bachelor on renewable energy; electronics and industrial systems without occupying any space or damage any material.

This paper is structured as follow. The implemented architectures of remote labs are presented in section 2. The developed software of each remote lab is described in section 3. Moreover, in section 4, used hardware is presented. Finally, section 5 provides and discusses obtained results.

2 Developed Architectures

2.1 3-tier Architecture

In order to implement the developed concept we used 3-tier architecture (users, server and irrigation system) (figure.1):

- **Users:** By using a web browser, the user can access to a java application (java applet) to control pumps, solenoid valves and supervise the data remotely in real time.
- **Server:** Stores outputs data of the embedded system in a database, then through two controllers (Fuzzy Logic (FL) and Artificial Neural Network (ANN)) processes and send data in real time to the web browser.
- **Irrigation systems:** The main system core, which based on Arduino board with the Ethernet shield and connected to the net. The system is powered by a solar panel and connected to three sensors (humidity, humidity soil and temperature), two solenoid valves and a pump via a relay module.

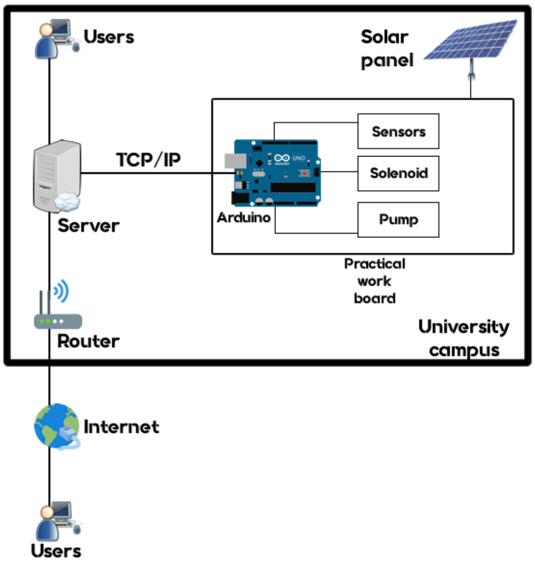


Fig. 1. General 3-tier architecture of remote lab.

2.2 2-tier Architecture

The design is based on a client-server architecture using open source software and low-cost embedded systems that allow the student, instructor or manager to access remotely to the manipulation (Fig. 2). The server part is equipped using Pcduino v3 board. It provides for the user a full environment for development based on embedded Linux as operating systems. The pcDuino is connected to the Internet through integrated Wi-Fi. This architecture uses more less peripherals and sophisticated materials.

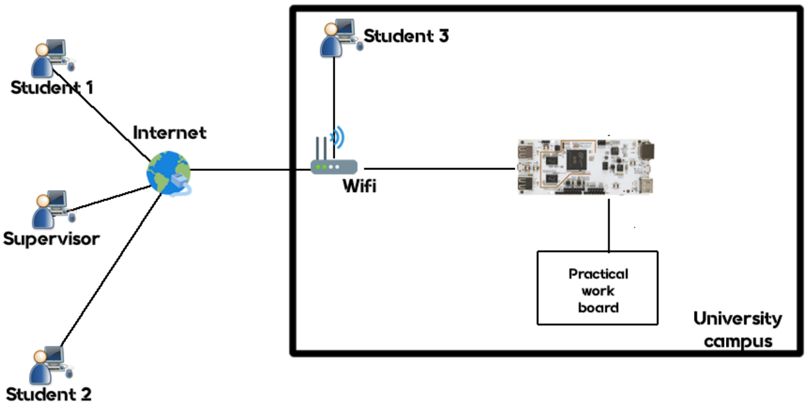


Fig. 2. General used architecture

3 Developed Software and Algorithms

3.1 Developed software for the irrigation control interface

The figure 3 presents the developed user interface, in this window five parts can be defined as follows:

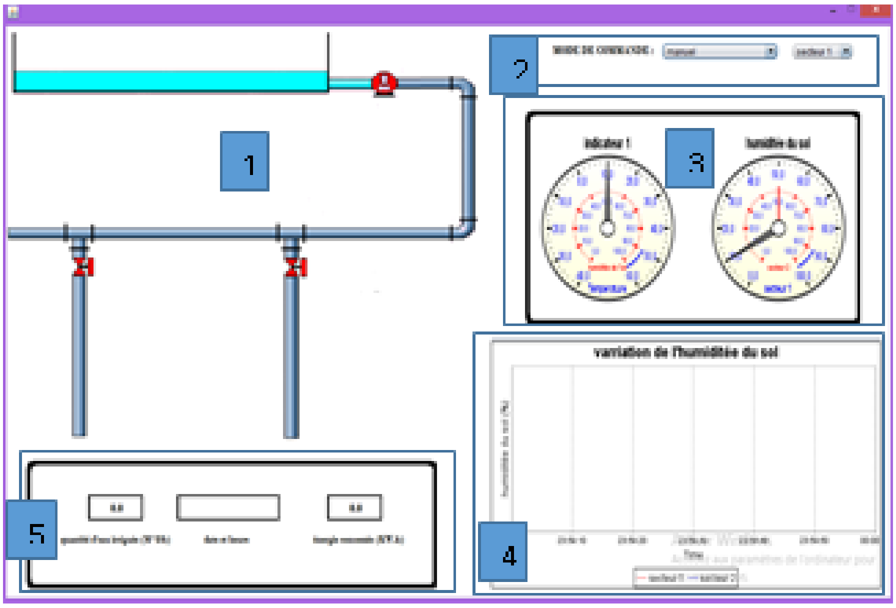


Fig. 3. Developed user interface.

- **Zone (1)** contains an interface to control a pump and two solenoid valves in case of control mode is manual. If using other modes, the pump and solenoid states were displayed.
- **Zone (2)** allows the user to irrigate the area 1 or 2 by choosing the control mode (manual, fuzzy logic controller, controller RNA, regulation).
- **Zone (3)** contains two double needle indicators that display the temperature, air humidity and soil moisture of sector 1 and 2.
- **Zone (4)** is used to plot the variation of the humidity versus real time.
- **Zone (5)** is containing two counters, the first one calculates the amount of irrigated water, and the second one is used to count the consumed energy by the pump.

In this manipulation students can control the pump remotely and manually, and the decision of irrigation can be taken based on the outputs data of sensors, voltage and current produced by the solar panel.

3.2 Developed algorithms

Fuzzy logic controller: The main function of this system is to manage irrigation and decide the quantity of water irrigated depending on the soil moisture, humidity of air and temperature. Therefore following variables were retained:

Input variables : Soil moisture: The first input was chosen by three qualify crowds subsets (dry, average and wet) defined as shown in figure 4:

Dry

$$\begin{cases} \mu_{dry}(x) = \frac{-1}{30}x + 1 \quad for\ x \leq 30 \\ \mu_{dry}(x) = 0 \qquad\qquad for\ x > 30 \\ \quad with\ \ \mu_{dry}(x) \in [0,1] \end{cases} \tag{1}$$

Average

$$\begin{cases} \mu_{average}(x) = 0 \qquad\qquad for\ \ x < 20 \\ \mu_{average}(x) = \frac{1}{30}x - \frac{2}{3} \quad for\ 20 \leq x < 50 \\ \mu_{average}(x) = \frac{-1}{30}x + \frac{8}{3} \quad for\ 50 \leq x < 80 \\ \mu_{average}(x) = 0 \qquad for\ \ 80 \leq x < 100 \\ \quad with\ \ \mu_{average}(x) \in [0,1] \end{cases} \tag{2}$$

Wet

$$\begin{cases} \mu_{wet}(x) = 0 \quad for \quad x < 70 \\ \mu_{wet}(x) = \frac{1}{30}x - \frac{7}{3} \quad for\ \ x \geq 70 \end{cases}$$

$$with\ \ \mu_{wet}(x) \in [0,1]¤ \tag{3}$$

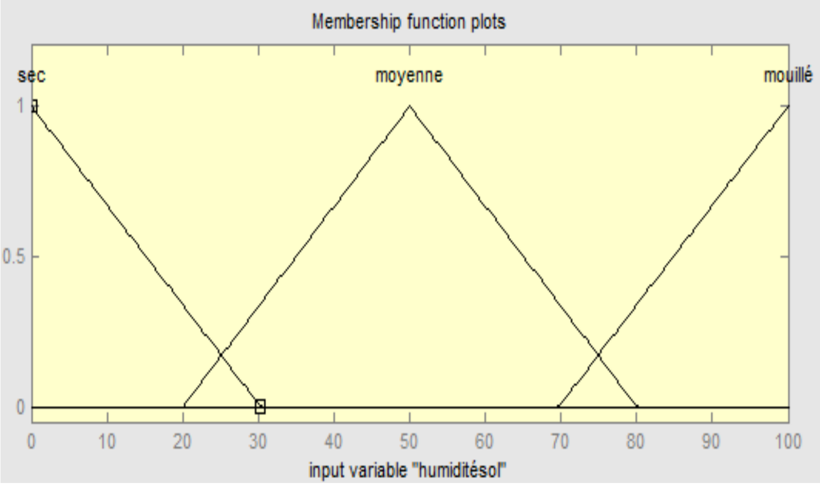


Fig. 4. Membership functions of soil moisture.

Input variables: Atmospheric humidity: Air humidity is treated as soil moisture: three subsets fuzzy (low, medium and high) (figure 5):

Low

$$\begin{cases} \mu_{\text{low}}(x) = \frac{-1}{30}x + 1 & for\ \ x \leq 30 \\ \mu_{\text{low}}(x) = 0 & for\ \ x > 30 \end{cases}$$
$$with\ \ \mu_{\text{low}}(x) \in [0,1]\text{¤} \tag{4}$$

Medium

$$\begin{cases} \mu_{\text{medium}}(x) = 0 & for\ x < 20 \\ \mu_{\text{medium}}(x) = \frac{1}{30}x - \frac{2}{3} & for\ 20 \leq x < 50 \\ \mu_{\text{medium}}(x) = \frac{-1}{30}x + \frac{8}{3} & for\ 50 \leq x < 80 \\ \mu_{\text{medium}}(x) = 0 & for\ 80 \leq x < 100 \end{cases}.$$
$$with\ \mu_{\text{medium}}(x) \in [0,1]\text{¤} \tag{5}$$

High

$$\begin{cases} \mu_{\text{high}}(x) = 0 & for\ x < 70 \\ \mu_{\text{high}}(x) = \frac{1}{30}x - \frac{7}{3} & for\ \ x \geq 70 \\ \quad with\ \mu_{\text{high}}(x) \in [0,1] & \end{cases} \tag{6}$$

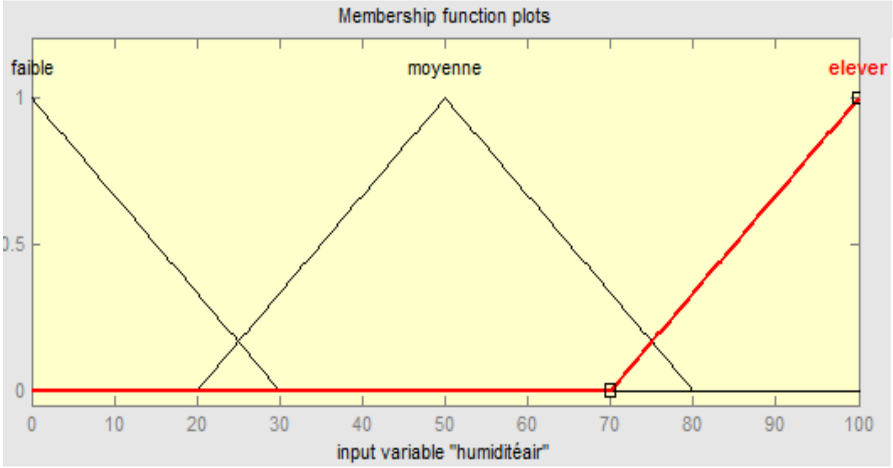


Fig. 5. Membership functions of soil moisture.

Input variables: Temperature: Temperature consists of three fuzzy sets: cool, normal, warm (figure 6):

Cool

$$\begin{cases} \mu_{\mathbf{cool}}(x) = \dfrac{-1}{35}x + \dfrac{3}{7} \quad for\ x \leq 15 \\ \mu_{\mathbf{cool}}(x) = 0 \qquad\qquad for\ \ x > 15 \\ \qquad with\ \mu_{\mathbf{cool}}(x) \in [0{,}1] \end{cases} \tag{7}$$

Normal

$$\begin{cases} \mu_{\mathrm{normal}}(x) = 0 \qquad\qquad\qquad for\ \ x < 10 \\ \mu_{\mathrm{normal}}(x) = \frac{1}{15}x - \frac{2}{3}\ for\ \ 10 \leq x < 25 \\ \mu_{\mathrm{normal}}(x) = \frac{-1}{15}x + \frac{8}{3}\ \ for\ \ 25 \leq x < 40 \\ \mu_{\mathrm{normal}}(x) = 0 \qquad\qquad for\ \ 40 \leq x < 50 \\ \qquad with\ \mu_{\mathrm{normal}}(x) \in [0,1] \end{cases} \tag{8}$$

Warm

$$\begin{cases} \mu_{\mathrm{warm}}(x) = 0 \qquad\qquad for\ x \leq 30 \\ \mu_{\mathrm{warm}}(x) = \frac{1}{20}x - \ \frac{3}{2} \quad for\ \ x > 30 \\ \qquad with\ \mu_{\mathrm{warm}}(x) \in [0,1] \end{cases} \tag{9}$$

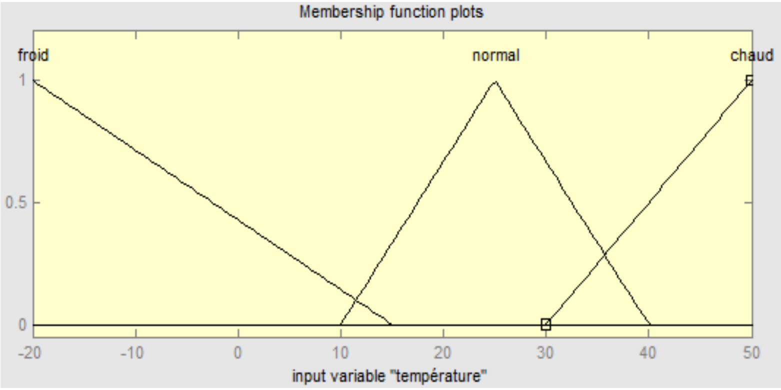


Fig. 6. Membership functions of the variable temperature.

Output variables: The output variable is mobilized as follows (figure 7):
Little

$$\begin{cases} \mu_{\mathrm{little}}(x) = \frac{-1}{35}x + 1 \quad for\ x \leq 35 \\ \mu_{\mathrm{litle}}(x) = 0 \qquad\qquad for\ \ x > 35 \\ \qquad with\ \mu_{\mathrm{little}}(x) \in [0,1] \end{cases} \tag{10}$$

Average

$$\begin{cases} \mu_{\text{average}}(x) = 0 & for\ x < 20 \\ \mu_{\text{average}}(x) = \frac{1}{25}x - \frac{4}{5} & for\ 20 \leq x < 45 \\ \mu_{\text{average}}(x) = \frac{-1}{25}x + \frac{14}{5} & for\ 45 \leq x < 70 \\ \mu_{\text{average}}(x) = 0 & for\ 70 \leq x < 96 \\ with\ \mu_{\text{average}}(x) \in [0,1] \end{cases} \tag{11}$$

Big

$$\begin{cases} \mu_{\text{big}}(x) = 0 & for\ x \leq 30 \\ \mu_{\text{big}}(x) = \frac{1}{46}x - \frac{25}{23} & for\ \ x > 30 \\ with\ \mu_{\text{big}}(x) \in [0,1] \end{cases} \tag{12}$$

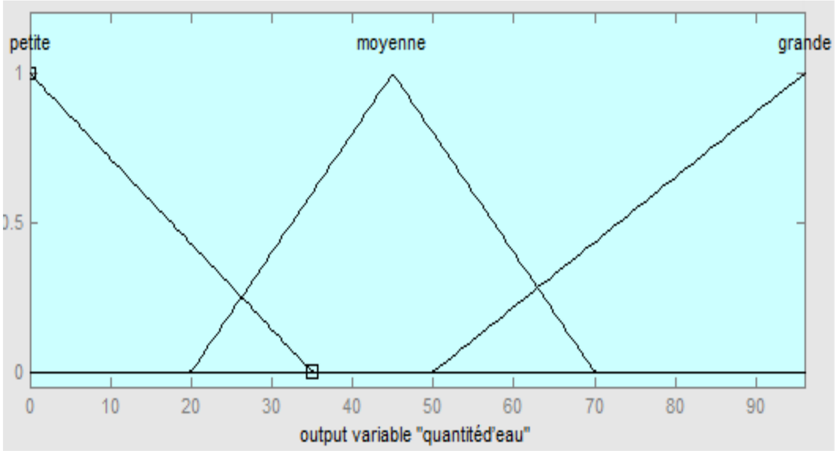


Fig. 7. Membership functions of the output variable.

3.3 Artificial neural network controller:

The irrigation system explained herein feat by a closed control loop. The control unit continuously receives feedback from various sensors on the ground [18]. The control unit decides the irrigation process based on data from sensors and preset parameters (soil type, nature of the plant, number of the days in the year).

This is a model for the irrigation system using the ANN [19].

ANN Controller is implemented using the following:

- **Topology:** Distributed Time Delay Neural Network is used
- **Training Function:** Bayesian Regulation function is used for training
- **Performance:** Sum of squared error is taken as performance of measurement
- **Goal:** The set goal is 0.0001
- **Learning Rate:** is set to 0.05

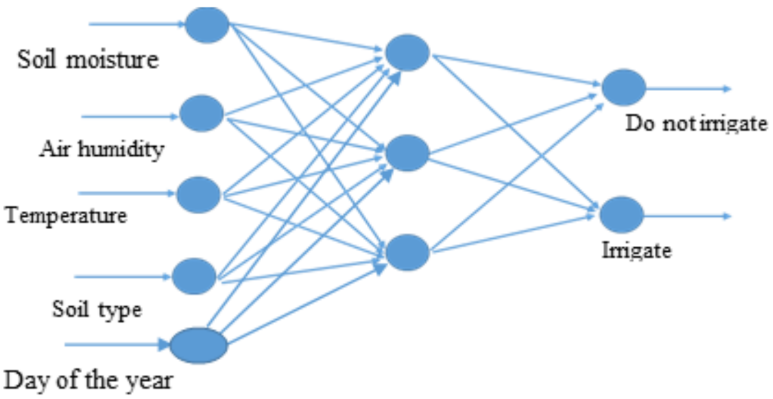


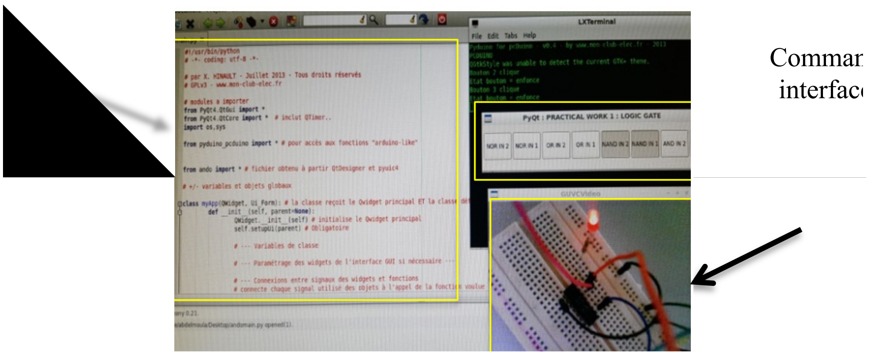
Fig. 8. Modeling of the irrigation system.

3.4 Description of the developed for electronic manipulation

In this work, Python programming language was employed to communication with the GPIO and system files of PcDuino. It provides a huge number of scientific libraries that make of it a robust server programming language. To facilitate the execution of functions related Pcduino we use an open source project PyDuino to develop programs using this library to send and get orders to and from GPIO, file system and multimedia.

In order to develop the interface we choose Pyqt4, a design language based on python. In the first step we design the window interface using Qt designer tools, after we get a python file. Two files were obtained:

- **PwMain.py:** Contains the main code to communicate with GPIO and import the design code from the second file.
- **Pw.py:** Contains the design code.



```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# par X. HINAULT - Juillet 2013 - Tous droits réservés
# GPLv3 - www.mon-club-elec.fr

# modules a importer
from PyQt4.QtGui import *
from PyQt4.QtCore import *  # inclut QTimer..
import os,sys

from pyduino_pcduino import * # pour accès aux fonctions "arduino-like"


from ando import * # fichier obtenu à partir QtDesigner et pyuic4

# +/- variables et objets globaux

class myApp(QWidget, Ui_Form): # la classe reçoit le Qwidget principal ET la classe dé
        def __init__(self, parent=None):
                QWidget.__init__(self) # initialise le Qwidget principal
                self.setupUi(parent) # Obligatoire

                # --- Variables de classe

                # --- Paramétrage des widgets de l'interface GUI si nécessaire ---

                # --- Connexions entre signaux des widgets et fonctions
                # connecte chaque signal utilisé des objets à l'appel de la fonction voulue
```

```
any 0.21.
e/abdelmoula/Desktop/andomain.py opened(1).
```

```
Pyduino for pcDuino - v0.4 - by www.mon-club-elec.fr - 2013
PCDUINO
QGtkStyle was unable to detect the current GTK+ theme.
Bouton 2 clique
Etat bouton = enfonce
Bouton 3 clique
Etat bouton = enfonce
```

Fig. 9. Developed user interface.

The Figure 9 shows the Qt application, part of the code and live video from the installed webcam module with PcDuino.

This work is also in progress and under test, for this reason we use Virtual Network Connect (VNC) to connect to the PcDuino for executing the PW.

4 Developed Hardware and Instrumentation

4.1 3-tier Architecture hardware description

We used Arduino kit to process outputs command and manage data received from sensors. Arduino is connected to the server through the Ethernet shield module that has an RJ45 port. The DHT100 and SEN0114 are sensors which measure respectively temperature, humidity and soil moisture via the Analog to Digital Converter (ADC) module of Arduino. To confirm the outputs data obtained from sensors, two sensors were used for the comparison in the developed prototype. The Arduino receives the commands of the server using our java application to control the pump and two solenoid valves during manual use of the developed system. Moreover, LCD was used to display the output data as shown in figures 10 and 11.

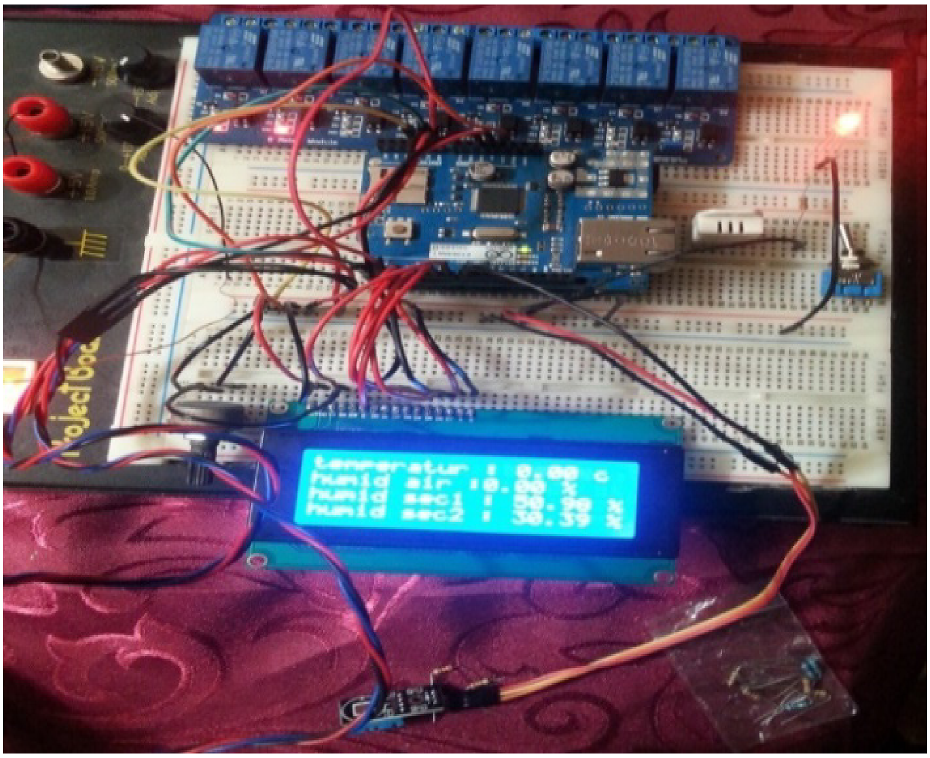


Fig. 10. Developed electronic circuit

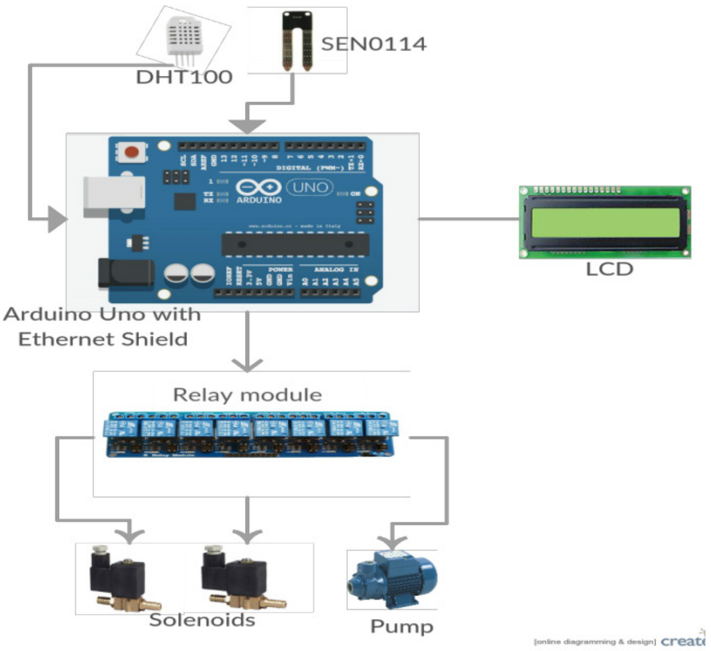


Fig. 11. Architecture of the electronic command.

4.2 2-tier Architecture hardware description

The principal hardware used in this architecture is the PcDuino v3 it has proven its applicability in various research [20], and the operating system used in this work was Linux. PcDuino contain many outputs and inputs:

- HDMI: interface to show the graphic desktop screen.
- Ethernet: to connect to the internet through RJ45 cable.
- Wi-Fi
- Audio out
- LCD : with a LVDS connector
- IR receiver : receive infrared signals
- Camera : CSI
- SATA host sockets to connect it to external hard drive.
- USB: 1 USB host and 1 USB OTG.
- Power: 5V and 2A.
- Arduino Headers : The PcDuino3 is 100% compatible with original Arduino shield and has the same pin :
  - 14 GPIO (General Purpose In/Out) Digital pins
  - 6 Analog Inputs connected to an ADC (Analog to Digital Converter) module.
  - 2 PWM : Pulse Width Modulation
  - 1 UART , 1 SPI and 1 I2C

Also, this embedded system has a CPU AllWinner A20 1GHz ARM cortex, a DRAM with 1GB and a storage of 4GB flash.

4.3 Manipulation description

The experience is destined to the 3rd year bachelor Electronics class, students can manipulate the integrated logic circuits such as AND, NAND and NO (Figure 12). This manipulation is aimed to obtain the truth table of each studied case by commanding the input ports of logic gate. Moreover, voltage and current of low and high state can be measured in the output using PcDuino's ADC.

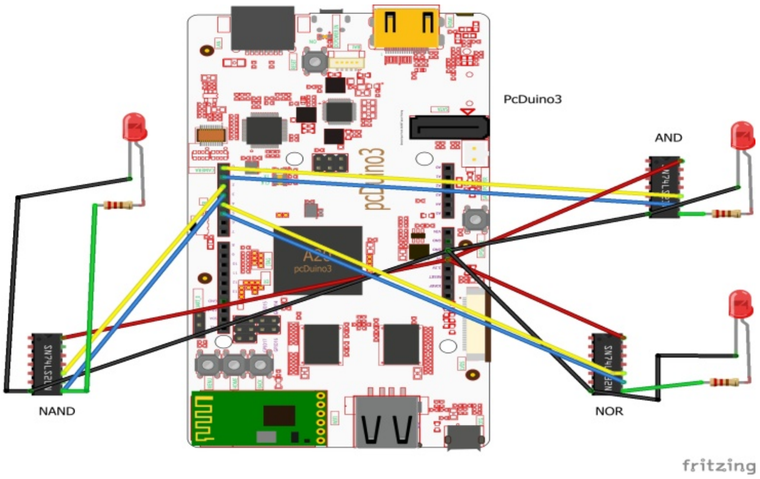


Fig. 12. Electronic schema of the logic experimentation.

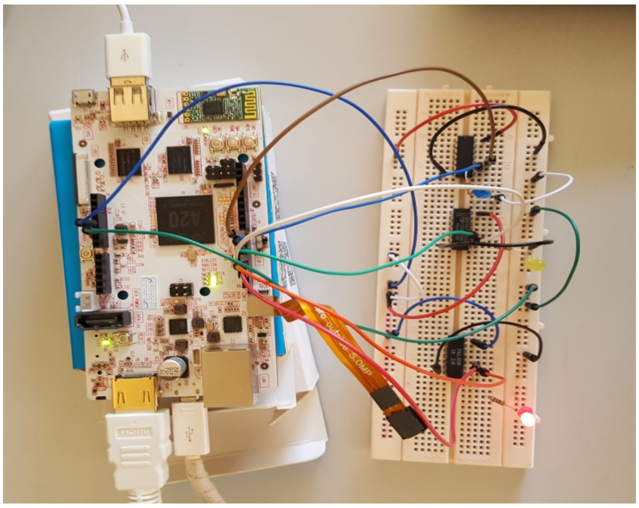


Fig. 13. Electronic schema of the logic experimentation.

5 Results, Analysis and Discussion

5.1 Experimental results

Concerning the Fuzzy Rules many combinations of the input parameters were used to fix fuzzy rules in order to find the optimal outputs results [21]. The table below summarizes all rules and logical relationships between the sub-functions of input and output.

Table 1. Rules and logical relationships between the sub-functions of input and output.

If Soil moisture =	And Air humidity=	And Temperature =	Then amount of water =
Dry	Low	Cool	Average
Dry	Low	Normal	Big
Dry	Low	Warm	Big
Dry	Medium	Cool	Average
Dry	Medium	Normal	Average
Dry	Medium	Warm	Big
Dry	High	Cool	Little
Dry	High	Normal	Average
Dry	High	Warm	Average
Average	Low	Cool	Little
Average	Low	Normal	Little
Average	Low	Warm	Average
Average	Medium	Cool	Little
Average	Medium	Normal	Little
Average	Medium	Warm	Average
Average	High	Cool	Little
Average	High	Normal	Little
Average	High	Warm	Little
Average	Low	Cool	Little
Wet	Low	Normal	Little
Wet	Low	Warm	Little
Wet	Medium	Cool	Little
Wet	Medium	Normal	Little
Wet	Medium	Warm	Little
Wet	High	Cool	Little
Wet	High	Normal	Little
Wet	High	Warm	Little

An experiment was carried out to compare the results obtained by the two algorithms. Two rows of the pumpkin were planted; the first line is irrigated using the fuzzy logic controller and the second line with the ANN controller. Figure 14 shows the used prototype.

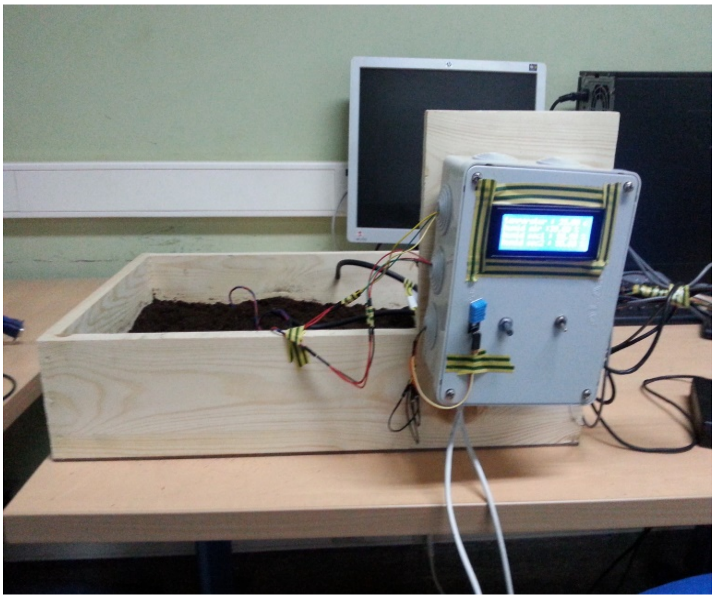


Fig. 14. Realized prototype.

Experience has unrolled for 11 days, for each 24 hours, calculates the quantity of water for each controller based on the output of the pump and the irrigation time.

- Pump flow rate = 0.3 L/min.
- Number of plants per line = 3.
- Irrigated area = 1800 mm^3

As presented in Figure 15, ANN controller uses an average of 0.68L / day unless the fuzzy logic controller using 0.83L / Days.

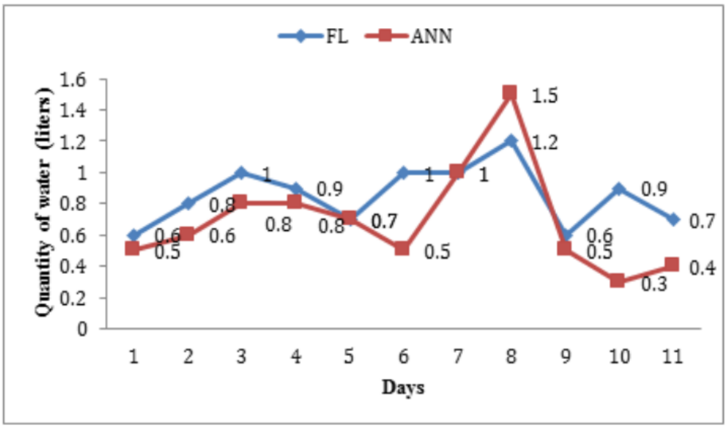


Fig. 15. Comparison between the controllers based on the water irrigated quantity.

For the manual mode irrigation, water quantity per day was used. Figure 11 shows that the fuzzy logic controller input irrigated water reduction of 16.08% compared to manual mode and ANN controller 32%.

5.2 Technical results

This paper we present two types of remote labs architecture, the first one is based on an Arduino and a server PC. The latter is powerful in terms of speed and capacity of data processing and storage. In this case, this architecture is reliable and the most suitable for this type of application. On the other hand, the second remote labs architecture is based on the use of an embedded system - PcDuino - which runs with an operating system. This case plays the role of one server and one control system at a time, but with limitations in processor speed and storage memory. With these performances, this type of architecture will be intended for applications that do not require a high-processing speed.

5.3 Pedagogical results

The time of realization of the PW: Unlike the hands-on PW, those remote are realized in months of time, since the student can use the manipulation quickly without any concern to damage the equipment. Where in the establishments those need labs, staff for the laboratory, with well-defined niches for the working hours, the remote lab is an interesting solution since they will gain additional time for other students can carry out their practical work.

The presence of the supervisor: The students who tested these applications have expressed that the presence of the teacher is not required since the student will follow a scenario and pre-defined steps to achieve the PW which does not require the intervention of a tutor.

Remote supervision: researchers: Who need to move periodically and each to the laboratory to monitor the progress of a phenomenon or measure quantities can also benefit from remote laboratories thanks to their benefits described in this paper which saves in terms of time, money and energy.

6 Conclusion

This study is aimed to improve the present education systems in the Moroccan universities by managing the practical manipulation for a large number of students, especially in the open-access faculties. In this work, two manipulations in renewable energy and electronic 3rd year bachelor classes were proposed. The approach was to use low-cost, high performance embedded systems as PcDuino and using free open source tools to connect and manipulate the experience using VNC.

However, a smart irrigation system was developed and used as remote manipulation for bachelor on renewable energy that is based on two closed loop controllers, the inputs of the first fuzzy logic controller are temperature, soil moisture and air humidity and the output is the amount of water to be irrigated, this has given satisfactory results because the use of water devoid of 16.08%. The second controller is based on the algorithms of the latter neural networks inputs such as temperature, soil moisture, air humidity, soil type and day of the year; the output is the decision to start irrigation or not irrigated, the results obtained by this controller are very satisfactory as it saves water consumption up to 32%.

The second step in this work is a remote practical work for electronic bachelor. Its objective is to manipulate the logic gate and obtain the truth table and measure physical quantities.

7 References

[1] J. Garcia-Zubia, I. Angulo, L. Rodriguez-Gil, P. Orduna, and O. Dziabenko, "Boole-WebLab-Deusto: Integration of a remote lab in a tool for digital circuits design," in 2013 IEEE Frontiers in Education Conference (FIE), 2013, pp. 848–854.

[2] C. A. Matarrita and S. Beatriz Concari, "Remote laboratories used in physics teaching: A state of the art," Proc. 2016 13th Int. Conf. Remote Eng. Virtual Instrumentation, REV 2016, no. February, pp. 385–390, 2016. https://doi.org/10.1109/REV.2016.7444509

[3] L. Gomes and S. Bogosyan, "Current Trends in Remote Laboratories," Ind. Electron. IEEE Trans., vol. 56, no. 12, pp. 4744–4756, 2009. https://doi.org/10.1109/TIE.2009.2033293

[4] A. Malaoui, "Low cost pedagogic device for practical works using embedded system.," Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA, vol. 2016–July, 2016. https://doi.org/10.1109/AICCSA.2015.7507160

[5] R. Jim, O. F. Avil, and M. F. Mauledoux, "Remote Lab for Robotics Applications," Int. J. Online Eng., vol. 14, no. 1, pp. 187–194, 2018. https://doi.org/10.3991/ijoe.v14i01.7674

[6] V. Reljić, B. Bajči, I. Milenković, J. Šulc, D. Šešlija, and S. Dudić, "Development of an experimental setup for remote testing of pneumatic control," Int. J. Online Eng., vol. 14, no. 1, pp. 195–202, 2018. https://doi.org/10.3991/ijoe.v14i01.7784

[7] A. V. Fidalgo et al., "The EOLES project remote labs across the mediterranean," Proc. 2014 11th Int. Conf. Remote Eng. Virtual Instrumentation, REV 2014, no. 530466, pp. 211–216, 2014. https://doi.org/10.1109/REV.2014.6784258

[8] A. Malaoui et al., "Implementation and validation of a new strategy of online practical works of power electronics for embedded systems," Int. J. Online Eng., vol. 13, no. 4, pp. 29–44, 2017. https://doi.org/10.3991/ijoe.v13i04.6659

[9] A. Malaoui, "Low cost pedagogic device for practical works using embedded system.," in Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, 2016.

[10] R. Heradio, L. De La Torre, D. Galan, F. J. Cabrerizo, E. Herrera-Viedma, and S. Dormido, "Virtual and remote labs in education: A bibliometric analysis," Comput. Educ., vol. 98, pp. 14–38, 2016. https://doi.org/10.1016/j.compedu.2016.03.010

[11] R. Heradio, L. de la Torre, and S. Dormido, "Virtual and remote labs in control education: A survey," Annu. Rev. Control, vol. 42, pp. 1–10, 2016. https://doi.org 10.1016/j.arcontrol.2016.08.001

[12] A. Malaoui, "Embedded Electronics Applied in Remote Labs and Practical Works: A technological Revolution in the Future, What Limits and Impacts?" ICSES Transactions on Computer Hardware and Electrical Engineering, vol. 2, no. 1, 2018.

[13] A. Malaoui, "Precise electric measurements with temperature using 10-bit embedded system: Application on photovoltaic junctions," Proc. Int. Conf. Microelectron. ICM, vol. 2016–March, pp. 265–268, 2016.

[14] V. M. Cvjetkovic and M. Matijevic, "Overview of architectures with Arduino boards as building blocks for data acquisition and control systems," Int. J. Online Eng., vol. 12, no. 7, pp. 10–17, 2016. https://doi.org/10.3991/ijoe.v12i07.5818

[15] M. Kalúz, L. Čirka,s R. Valo, and M. Fikar, ArPi Lab: A low-cost remote laboratory for control education, vol. 19, no. 3. IFAC, 2014. https://doi.org/10.3182/20140824-6-ZA-1003.00963

[16] I. Michaelides, P. Eleftheriou, and K. Economides, "Solar energy e-learning laboratory - Remote experimentation over the Internet," Rev 2005 2nd Int. Conf. Remote Eng. Virtual Instrum., pp. 1–10, 2005.

[17] A. Malaoui, "Implementation and tests of an automatic system to improve electrical energy in photovoltaic installations," Int. J. Innov. Appl. Stud., vol. 8, no. 1, pp. 328–340, 2014.

[18] A. Apostolakis, K. Wagner, I. N. Daliakopoulos, N. N. Kourgialas, and I. K. Tsanis, "Greenhouse Soil Moisture Deficit under Saline Irrigation and Climate Change," Procedia Eng., vol. 162, pp. 537–544, 2016. https://doi.org/10.1016/j.proeng.2016.11.098

[19] E. F. Poyen, A. Pal, B. Majumder, A. Ghosh, and R. Bandyopadhyay, "ANN vs. FUZZY in Automated Irrigation," Int. Conf. Innov. Eng. Technol. Dec. 28-29, 2014 Bangkok, Thail., no. December 2014, pp. 155–160, 2014.

[20] A. Abouhilal, A. M. Taj, R. Irkettou, M. Mejdal, and A. Malaoui, "Development and Testing of a Remote Laboratory for Practical Work Based on Embedded Electronics," J. Fundam. Appl. Sci., vol. 10, no. 4S, pp. 487–490, 2018.

[21] A. Malaoui, M. Ankrim, K. Quotb, and M. A. Fichouch, "Modelling and realization of a programmable thermic regulator using fuzzy logic algorithm: comparison with PI Controller," Assoc. Adv. Model. Simul. Tech. Entrep. J., vol. 64, pp. 1–2, 2009.

8 Authors

Abdelmoula Abouhilal is a PhD student with the research team in Electronic, instrumentation and measurement in the polydisciplinary faculty of Beni Mellal. His research interests include remote laboratories, instrumentation and embedded systems.

Amine Moulay Taj is a PhD student with the research team in Electronic, instrumentation and measurement in the polydisciplinary faculty of Beni Mellal. (mytaj.amine@gmail.com).

Naima Taifi is an Associate Professor in the physics department of the polydisciplinary faculty of Sultan Moulay Slimane University, Béni Mellal, Morocco. (taifinaima@yahoo.fr)

Abdessamad Malaoui is an Associate Professor in the physics department of the polydisciplinary faculty of Sultan Moulay Slimane University, Béni Mellal, Morocco. His research interests are in electronics, Industrial computer, µP & µC, renewable energy, PV system performance monitoring. (a.malaoui@usms.ma)

Article submitted 2018-10-11. Resubmitted 2018-12-29. Final acceptance 2018-12-30. Final version published as submitted by the authors.