

A Low-Cost Educational Remotely Controlled Solar Energy Laboratory

Salaheddin Odeh*, Tarek Rishmawi, Rafat Al-Hawash

Faculty of Engineering, Al-Quds University, Abu Dies, Jerusalem, Palestine

*Corresponding author: salah.odeh@staff.alquds.edu

Abstract This paper proposes the hardware and software implementation of the system required to establish a low-cost educational remotely controlled solar energy laboratory. The system consists of two main parts, a Solar Energy System and a Remotely Controlled Laboratory. The Solar Energy System is a Photovoltaic system, which consists of multiple photovoltaic cells that convert solar radiation (sunlight) or normal lights into usable direct current (DC) electricity, and then it either charges a backup battery or uses an inverter circuit that changes direct current (DC) to alternating current (AC). The other part of the system is a Remotely Controlled Laboratory, aimed at enabling students to control solar energy experiments remotely.

Keywords: remote labs, instrumentation, photovoltaic system, solar energy

Cite This Article: Salaheddin Odeh, Tarek Rishmawi, and Rafat Al-Hawash, "A Low-Cost Educational Remotely Controlled Solar Energy Laboratory." *American Journal of Electrical and Electronic Engineering*, vol. 5, no. 4 (2017): 126-135. doi: 10.12691/ajeec-5-4-2.

1. Introduction

Laboratories make science come alive; they are considered the core of engineering learning as they provide better understanding of scientific theories. Laboratories are divided into three categories: proximal (Hands-on), simulators, and remote laboratories. Proximal labs are costly; they take up large space, and lecturer time. Simulators are emulation of real-world experiments with many drawbacks including the inaccuracy of the mathematical model that won't reflect the real-world problem.

Labs play a fundamental role in representing concepts and principles, providing the ability to design and explore, encouraging social skills and improving practical skills. These educational targets are proposed by ABET. ABET (Accreditation Board for Engineering and Technology) is a nonprofit, non-governmental organization that accredits college and university programs in the disciplines of applied science, computing, engineering, and engineering technology [1]. Students can have better understanding of the scientific theories by applying them in the labs, therefore, most courses in the engineering education require labs [2,3].

Remote labs provide the abilities to share the devices, and to relax time constraints. They can be considered an integral part to traditional hands-on [4], as they include many advantages, for example, relaxing geographical constraints, sharing of expensive devices and infrastructure, improving experiment precision, improving efficiency by repeating, improving safety and security as there is no risk of major failure. In contrast to hands-on, they lack trouble shooting, debugging and equipment setup. The boundaries among the three categories are blurred in the sense that

most labs are mediated by computers, and that the psychology of presence may be as important as technology [5].

In this paper, a remote lab system for teaching solar energy laboratory is presented; the idea is to demonstrate the work of a solar energy system that uses Photovoltaic cells to convert sunlight or normal light into electricity, and the system is remotely controlled and monitored using an Arduino microcontroller connected serially via USB, interfaced, and programmed using LabVIEW. LabVIEW (short for Laboratory Virtual Instrument Engineering Workbench) is a system-design platform and development environment for a visual programming language from National Instruments.

It is to note that similar implementations already exists in literature, for example [6,7]. One major issue of this contribution is not only being able to reproduce the lab in a small scale for several behaviors of a real solar system to allow the users to make experiences close to reality, but implementing a lab, which is considerably cheap, so it can be replicated easily, and it is flexible, so the remotely controlled part can be easily reused for different experiments. The purpose of the system is to provide an educational environment for the students of the parts and devices of the solar energy system, so they can learn the parts, remotely control them, and do some measurements.

This paper proposes the hardware and software implementation of the system required to integrate the physical system within a remote laboratory. The system, which is called "A Low-Cost Educational Remotely Controlled Solar Energy Laboratory", consists of two main parts, a Solar Energy System and a Remotely Controlled Laboratory. The Solar Energy System is a Photovoltaic system, which consists of multiple photovoltaic cells that convert solar radiation (sunlight) or normal lights into usable direct current (DC) electricity,

and then it either charges a backup battery or uses an inverter circuit that changes direct current (DC) to alternating current (AC). The system includes also a Charge Controller, which is used to limit the rate at which electric current is added to or drawn from the battery, it prevents overcharging and overvoltage, which can reduce battery performance or lifespan, and may pose a safety risk. The other part of the system is a Remotely Controlled Laboratory, aimed at enabling students to control solar energy experiments remotely. The solar energy system parts are mounted on a movable table that contains the following parts:

1. Arduino Mega microcontroller: used to remotely control the Relays Switching Matrix, which is connected via USB to a server running LabVIEW customized software.
2. Relays Switching Matrix: used to remotely connect, and disconnect devices.
3. Digital Multimeter: connected to LabVIEW, used as virtual instrument to remotely measure amps and voltages of the connected devices.

This allows the user to access the system remotely by visiting a Website running on the server and communicates with an Arduino microcontroller that completely controls the laboratory by controlling the relays switching matrix, so the user has the ability to send commands and receive the desired output.

2. A Brief Comparison between Hands-on, Virtual Labs and Remote Labs

Engineering curriculum contains a set of credit hours required for lab experiments. Engineering is distinguished by the fact of being an applied science, and to be able to apply scientific knowledge and theories to develop practical applications and inventions. Students must carry out experiments in order to gain better understanding of scientific theories, to learn how to deal efficiently with equipment and lab devices, and to cooperate with each other in a practical environment, which will be very important for their career.

Hands-on labs are distinguished from the other types by two characteristics [8]. The students and the lab equipment must be locally present in the same place, and the real devices and instruments are physically connected. Because of space and money limitations, hands-on labs sometimes cannot be afforded. Therefore, virtual and remote labs usage has increased in the engineering curriculum.

A virtual lab is a software emulation that imitates a real-world experiment specified by a mathematical model. It is an interactive environment for creating and conducting simulated experiments on a computer, that is, in fact imitate hands-on labs [9]. Unfortunately, this makes students unfamiliar with real equipment and components, making it harder for them to deal with this equipment in real work. Moreover, most of the mathematical models lack accuracy, which can be critical for lab experiments. Nowadays, modern technologies have

greatly improved the environment of labs; we can build simulations of complex processes of various sciences including biological, medical, chemical, physical, agriculture, engineering and educational practice by using the various multimedia content forms available. There are many virtual lab programs or simulators available to be used, such as LabVIEW, MATLAB, OrCAD family, and many more.

Remote labs use modern telecommunication and Web technologies to remotely perform real experiments with real devices and components at a distant physical location, whilst the researcher is in a separate geographical location. Many academic institutions provide a variety of remote lab experiments designated as Web-based labs: these labs support remotely controlled physical experiments [10]. In order to support such experiments, remote labs need a lab server; the lab server communicates between the user and the physical experiment in the lab [11]. This type of lab is suitable for distance learning courses where students do not need to be locally present on lab. The student or researcher can access the lab remotely and control the instruments using their favorite Web browser such as Firefox, Chrome or Internet Explorer. By using a suitable interface on the browser to send commands and receive real-time measurements and feedback from the lab components. Usually, users' access to the system is assigned by a specific time schedule and managed by a database.

3. The Stat-of-the-art Remote Labs for Teaching Renewable Energy

The demand for a remotely controlled solar energy laboratory in the engineering education is enormous, due to the cost of the equipment and devices to conduct real experiments and the restrictions on time and place. Modern computers and high speed Internet access are readily available, thus supporting a remote laboratory idea.

Currently, there is already an European project that involves an open access remote PV laboratory for educational purposes. The project is a partnership between five European Universities as well as the Technical Chamber of Greece. These Universities are: Technological Educational Institution of Athens, Greece, University of Bordeaux, France, Brunei University, UK, the Open University, UK and the University of Savoie, France (see Figure 1).

In Palestine, there are few educational solar energy systems, but they are not remotely controlled and students have limited time and access to such systems and instead of conducting real experiments, a variety of computer simulations are used to demonstrate fundamental principles of the solar energy system. Our system's main goal is to be used in education, so it can be included without hindrances in an Engineering curriculum. Remote experiments could be used for distance education. Thus, students from around the world, whose institutes do not have solar laboratories can perform experiments and then analyze the results according to the theory.

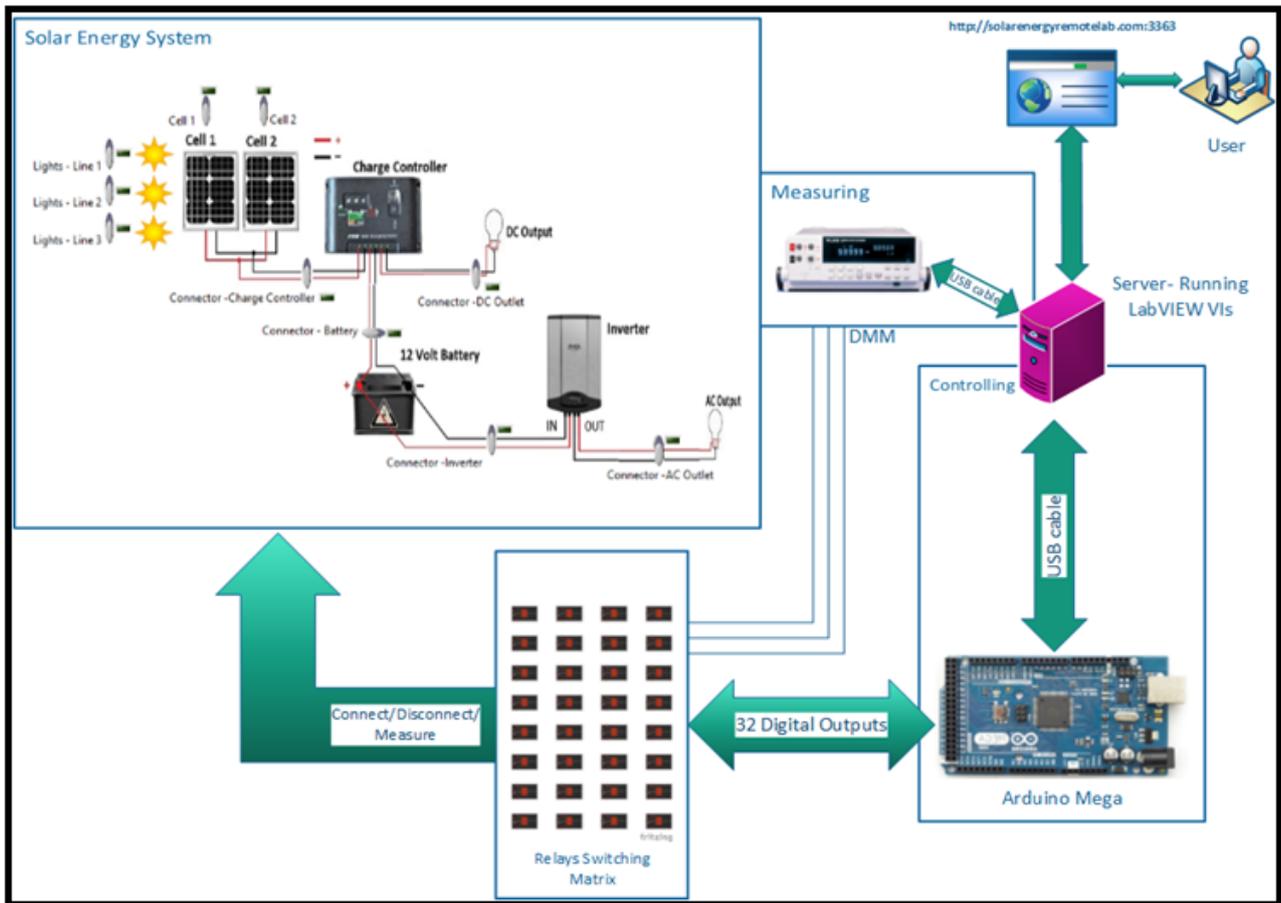


Figure 1. The System Block Diagram

4. System Requirements Analysis

During the process of requirements engineering, systems engineers establish the required system services and the constraints, under which the system must later operate. In many cases, the requirements themselves are the descriptions of the system services and constraints that are generated during the requirements engineering process. To discover the requirements, the system designer should, on the one hand, gather information about the proposed and existing systems and, on the other, distill the user and system requirements from this information, which can be in forms of documentation, system stakeholders and the specifications of similar systems. Several approaches are available to acquire the requirements such as interaction with stakeholders through interviews and observation, as well as using prototypes and scenarios [12]. To achieve the required goals of the system, we had to study the requirements of designing such a system very carefully, that includes the ability to remotely access, control, and monitor the devices in the system. Requirements are divided into two parts, the hardware part and the software part.

Building a complete Photovoltaic Solar Energy System requires each of the following components to accomplish its task:

1. Light Sources: three lines of 60-Watt light Bulbs (6 bulbs each line), used instead of sunlight for demonstration purpose and for the lab experiments with the ability that each line can be controlled.

2. Two Photovoltaic (PV) cells. A PV cell is an electrical device that converts the energy of light directly into Direct Current by the photovoltaic effect, which is the creation of electric current in a material when exposed to light. The specifications of the used PV cells are shown in Table 1.
3. A Charge Controller or a battery regulator is used to limit the rate at which electric current is added to or drawn from the 12 V battery, it prevents overcharging and overvoltage, which can reduce battery performance or lifespan, and may create a safety risk. It also prevents completely draining a battery, or performs controlled discharges, depending on the battery technology, to protect battery life. A 12 V charge controller with a current of 10 A is shown in Figure 2.
4. Inverter: it is a circuit that changes direct current (DC) to alternating current (AC).
5. 12V-7A DC Deep-Cycle battery: it is a suitable choice because of its reasonable price compared with its quality, efficiency and life cycle. Voltage bank must match the system requirement, which is determined by the inverter.

Table 1. PV Cell Specifications

Cell Property	Value
Max Power	10W
Max Power Current	0.58A
Open Circuit Voltage	24V
Short Circuit Current	0.64A



Figure 2. 12V 10A Charge Controller

The other part of the hardware system is a Remotely Controlled Laboratory; the solar energy system parts are mounted on a movable table controlled and monitored using the following parts:

1. Arduino Mega 2560 R3 (Figure 3): It's the brain of the system, it is a microcontroller board based on the ATmega2560. It was chosen because of the following reasons: its low cost price; since one of the goals of the system is to accomplish a low-cost remote laboratory, its scalability as it has many digital pins (54), its compatibility with LabVIEW's Graphical Language, and its ability to provide real-time control. Moreover, it can be interfaced and programmed serially using a USB cable. The specifications of the Arduino Mega 2560 R3 are shown in Table 2.
2. Fluke 45 Digital Multimeter (Figure 4): it is used as virtual instrument to remotely measure amps and voltages of the connected devices. It was chosen for many reasons including the interfacing port, its compatibility with LabVIEW. The interfacing of the Multimeter is done with a serial cable using the serial port (RS-232c) on the Multimeter.
3. A RS-232c Serial to USB 2.0 PL2303 Adapter is used to connect the Fluke 45 to the server. It provides over 1 Mbps data transfer rate. To interface between two DTE devices (Computer and DMM), it is required to make a null modem (Crossover) Female/Female cable instead of using Modem or DCE's. Crossover cable is usually used to connect two DTE devices as shown in Figure 5.
4. Relays Switching Matrix: The relay is an electro-mechanical device with an electromagnetic coil and a metal switch. When the coil is energized, the metal switch will move, completing the circuit. Normally-open (NO) contacts connect the circuit when the relay is activated; the circuit is disconnected when the relay is inactive. The system needs the Relays Switching Matrix to fully access and control the devices, to switch them on or off, to make a specific arrangement in the physical system, and to measure amps and voltages when needed. 32 relays are required to achieve the purpose of this system, 8 of these relays are used as backup relays.

It is required to interface each relay so it can be safely connected to the microcontroller digital channels to protect the microcontroller. The circuit shown in Figure 6 is used to interface each relay of the 32 relays to the 32 digital outputs channels.

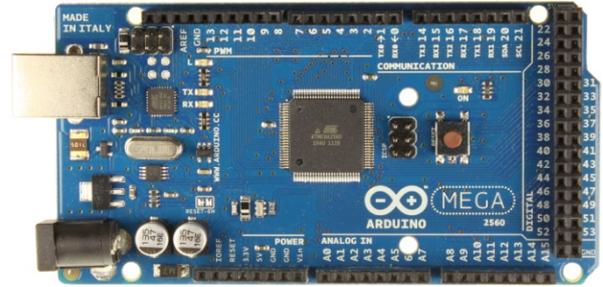


Figure 3. Arduino Mega 2560 R3

Table 2. Arduino Mega 2560 R3 Specifications

Feature	
Operating Voltage	5V
Digital I/O Pins	54
Analog Inputs	16
DC Current per I/O pin	40mA
Flash Memory	256KB
SRAM	8 KB
EEPROM	4KB
Clock Speed	16 MHz



Figure 4. Fluke 45 Digital Multimeter

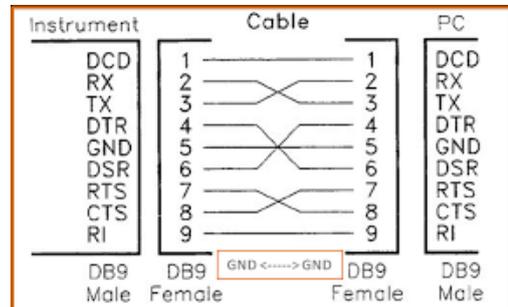


Figure 5. Crossover Cable

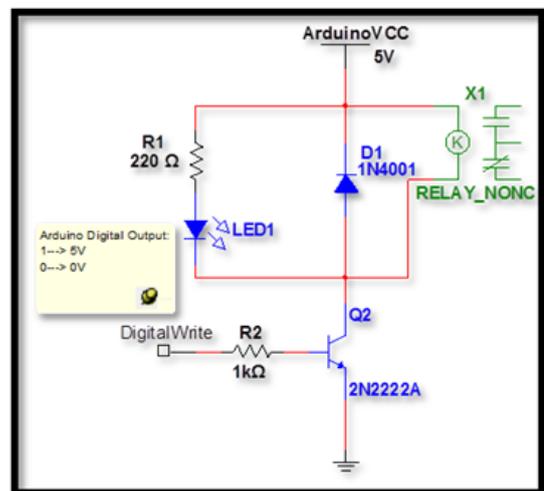


Figure 6. Relay Interfacing Circuit

Finally, the system is large with many devices and components so it is required to be installed on a movable table (width=96 cm, height=100cm, depth=50 cm) that it will be easy to move and maintain when needed. Shelves are used to hold devices inside the table, like the cells, the inverter, the battery, and the digital Multimeter. A control cabinet is used on the side of the table for safety and neat design to include the Arduino microcontroller, the charge controller, the Relays Switching Matrix, and all of the connections to the devices. Electrical wires are 2.5mm that hold currents up to 10 Amps, taking into account that we are dealing with DC and AC, and 32 Relays are used to ensure full control of the system, some of them are used as backup in case of failure. Finally, the connections around the table and inside the cabinet are arranged and terminated using Screw Terminals.

5. Software Design Requirements

Generally, it is difficult to resolve requirements and design uncertainties of research projects. For this reason, our system development follows the prototyping approach based on the iterative design and development approach [13]. In the iterative process, the stages: specification, design, development, and testing are not chained; rather interleaved and concurrent. The system containing software and hardware is developed in a series of increments with each including new system functionality. Unlike most remote labs mainly dedicated for a particular experiment, our system is reusable by both physical (electronic) and software configuration. In other words, after connecting the experiment components and equipment with lab server, the experiment designer maps the physical experiment structure into the configuration data-structure. Building reusable software-ergonomic components either as standalone or integral components for designing user-interfaces is

multidisciplinary which makes it difficult to design and to implement. It demands knowledge and skills from disparate science and engineering disciplines such as cognitive science in order to make remote environment usable and catch up with modern learning theories, education for the integration of remote laboratories activities in the students learning process, software engineering so that the imagined model can be implemented.

Software development is divided into two main parts: controlling and monitoring. Controlling involves interfacing and programming the Arduino microcontroller with LabVIEW, it is done using LINX toolkit by LVH [14]. To interface the Arduino to the LabVIEW, we must upload a firmware to its memory to open the connection between the computer's USB port and the USB adapter in the Arduino. Monitoring involves interfacing the digital Multimeter with LabVIEW to monitor and make measurements remotely. Fluke45 requires a specific driver to be installed in LabVIEW to give the system the ability to control the device, read, and write to it.

The development environment used is the LabVIEW software; the programming is done using its graphical language named "G". There are many editions of LabVIEW, each edition's cost differs depending on what bundles are added to its base software. The most suitable edition for the system is LabVIEW Student Edition, which is the academic edition and it is cost-effective.

To be able to connect both the Arduino and the Fluke45 DMM and to be used as Virtual Instruments, LabVIEW requires some important packages to be installed first including VISA [15] and JKI VI Package Manager [16]. The Virtual Instrument Software Architecture (VISA) is used to interface and program the Fluke45 Digital Multimeter with LabVIEW, it allows us to install the required driver for the Multimeter. The JKI VI Package Manager is used to download the required packages for the Arduino microcontroller. Figure 7 demonstrates the various use cases required to be achieved in this system.

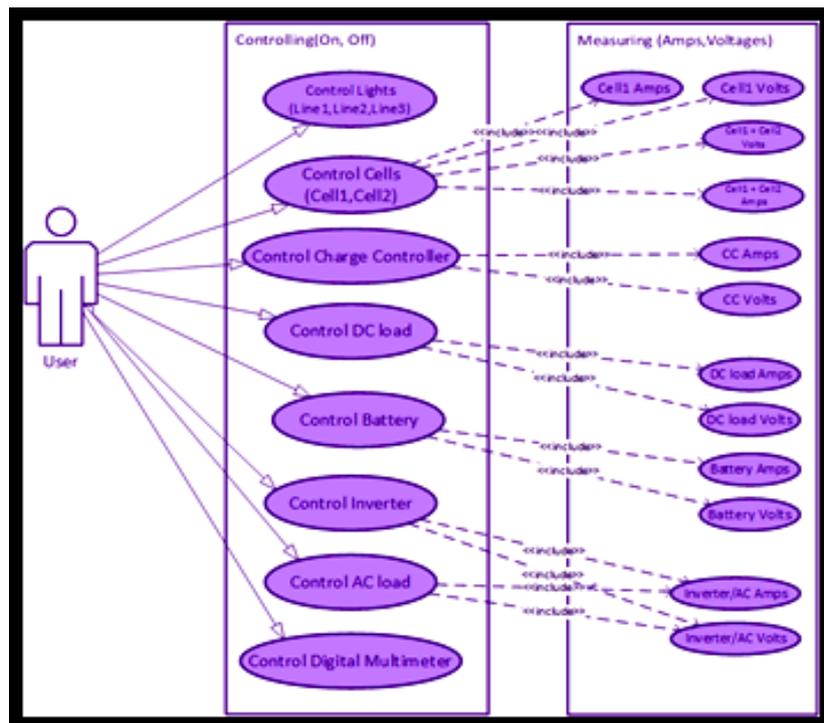


Figure 7. Use-Case Modeling

6. Realizations and Implementation

A. Implementing the Software

LabVIEW programs are called Virtual Instruments (VIs) as mentioned in previous sections, because they interface the physical instruments, such as the Arduino and the Digital Multimeter. Creating a new Virtual Instrument cause two windows to appear: the front panel window and the block diagram. The front panel window is the user-interface for the VI that includes the controls and the indicators. The Block diagram includes terminals, subVIs, functions, constants, structures, and wires, which transfer data among other block diagram objects.

By using LINX interface, the following four functions are needed to write to a digital channel in the block diagram window: *initialize.vi*, *digital write.vi*, *close.vi*, and *simple error handling.vi* (see Figure 8). All these functions must be included inside a while loop (super loop) that is shown in Figure 9.

The initialize function opens a connection to the Arduino (LINX device), the value of the digital output is written to the digital channel, the close function closes the connection to the Arduino, and the simple error function

handle errors. For the 32 relays, the above process is repeated to write to the 32 digital output channels.

To achieve modularity, the logic of the system is included inside the while loop of the Arduino as a “core” sub Virtual Instrument (subVI). A subVI is made using smaller sections of codes, they are the same as VIs because they contain front panels and block diagrams, and you can call them from a VI. They are similar to subroutines in text-based programming languages. By using combinational logic, which is implemented with Boolean circuits, we made the subVIs. Combinational logic is used to build circuits that produce specified outputs from certain inputs, so by using LabVIEW’s logic gates, we started by making a default subVI module for measuring amps and voltages to avoid damages and conflicts for safety, as shown in Figure 10.

This subVI is called “Relays Logic” that is included in the “core” logic subVI that will be included in the while loop of the Arduino and connected to the digital output channels. There is another subVI called “DCAC logic” (Figure 11) that is used to prevent the user from connecting the DC load and the AC load at the same time, and it is included also in the “core” logic subVI.

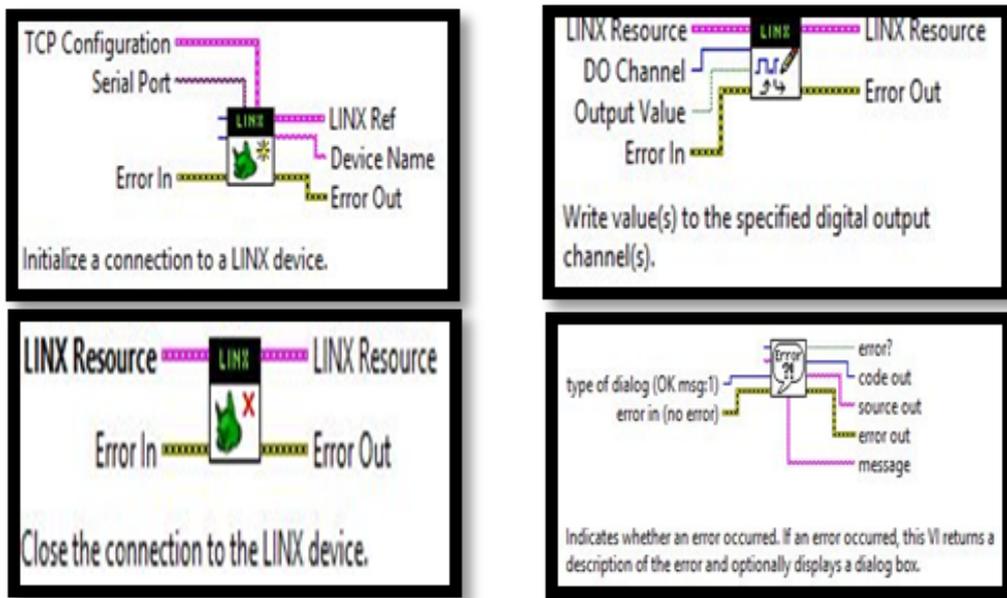


Figure 8. Functions needed to write to a digital channel

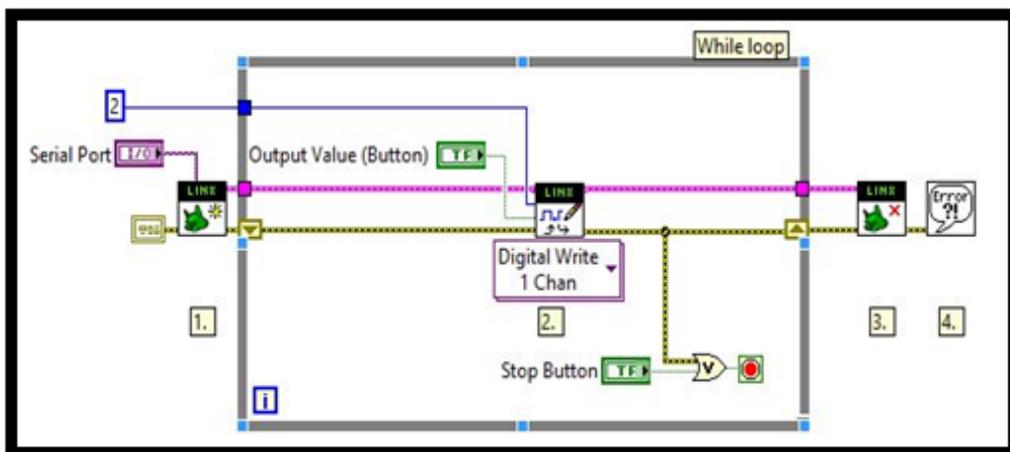


Figure 9. Digital Write to one Channel

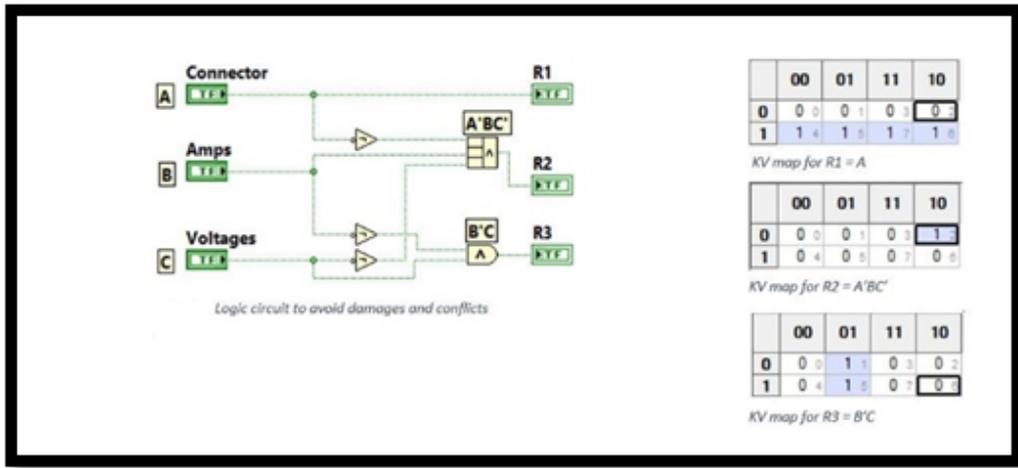


Figure 10. "Relays Logic" subVI

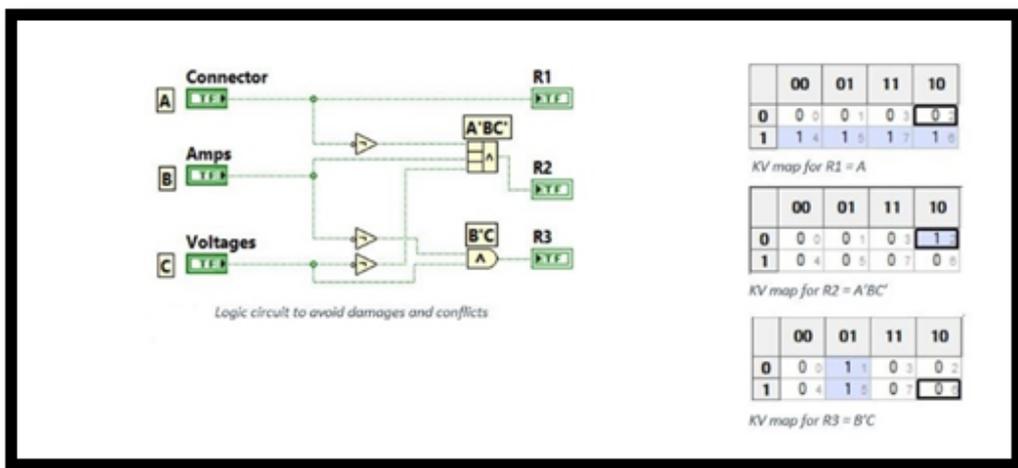


Figure 11. DCAC logic subVI

© 2014
Eng. Rafat Al-Havash,
Eng. Tarek Rihmani,
Dr. Salehuddin Odeh,
ALL RIGHTS RESERVED.

Welcome | Port Configurations

Welcome to the Solar Energy Remote Laboratory!

>In the left panel you can control the Solar Energy System.
>In options panel below, you can choose the Measurement mode.
>In the DMIM panel, you can choose options related to the DMIM and see the DMIM outputs and the system indicators.

Options
Connections
Error

Amperes
Voltages

DMIM - Power

DMIM - Options Errors

Primary Display (By AC+DC Vrms)
AC-DC rms Volts 5

Secondary Display (By None)
None 0

Trigger Mode (Internal)
Internal Rear Disabled 0

Timeout (25 sec)
25000

Range (Auto)
Auto 0

Rate (Fast)
Fast 2

Send Trigger (F-No)
Yes
No

Measurement
0.00000

Cells - Voltages
Cells - Amperes
CC - Voltages
CC - Amperes
Battery - Voltages
Battery - Amperes
AC - Voltages
AC - Amperes
DC - Voltages
DC - Amperes

Stop the System
STOP

Figure 12. The System User-Interface

To achieve compatibility between the physical system and the software system, we used the “Case Structure” in the “Core logic” subVI to reflect the physical system and separate the measurements part from the connections part. The case structure includes the following cases: Connections, Cells – Measurements, Charge Controller – Measurements, DC Outlet – Measurements, Battery – Measurements, and Inverter – Measurements. Each case includes its own logic, the “Relays logic” subVI, the “DCAC logic” subVI, and errors handling depending on the desired outcome. Finally, this subVI is included in the Arduino VI’s WHILE loop (main program), each output of the subVI is connected to a digital channel output in the Arduino VI.

Publishing the system to the World Wide Web is simple that can be achieved with a tool called “Web Publishing tool” in LabVIEW that makes it easy to configure a Web

server and publish the website. The user-interface of the system is shown in Figure 12.

B. Hard Implementation

The Photovoltaic Solar Energy System is physically connected as shown in Figure 1 above; each device can be connected or disconnected using a switch button implemented physically by using a relay connected to a digital output. Each relay is included in a 4-Relays Interfacing Module as shown in Figure 13 below. The block diagram in Figure 14 shows how the system is physically connected with the Relays Switching Matrix, R2 - R12.

For the measurements part, two disconnecting/connecting relays are used when measuring volts or amps to avoid any noise coming from the common wire. And in the case of amps measurements, the device must be disconnected, putting the Multimeter in series with the circuit.

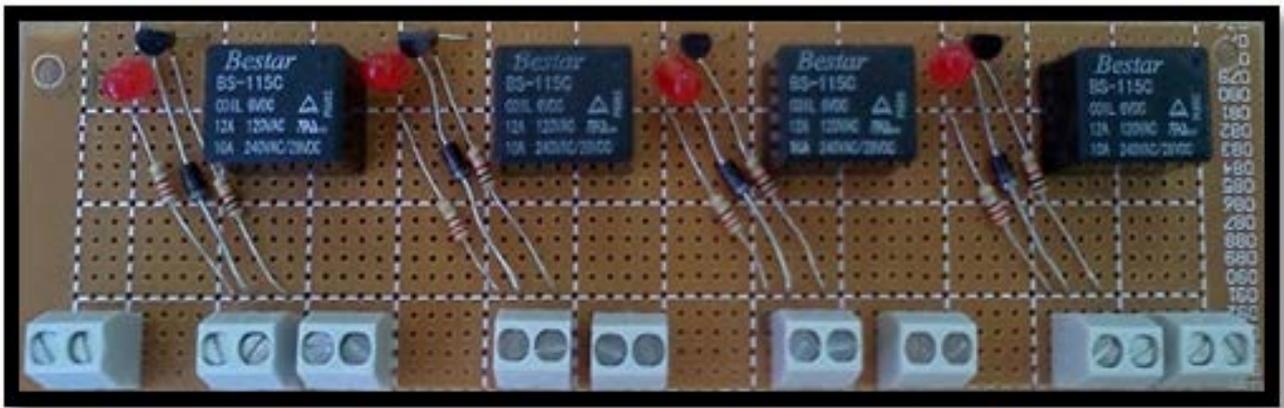


Figure 13. Relays Interfacing Module

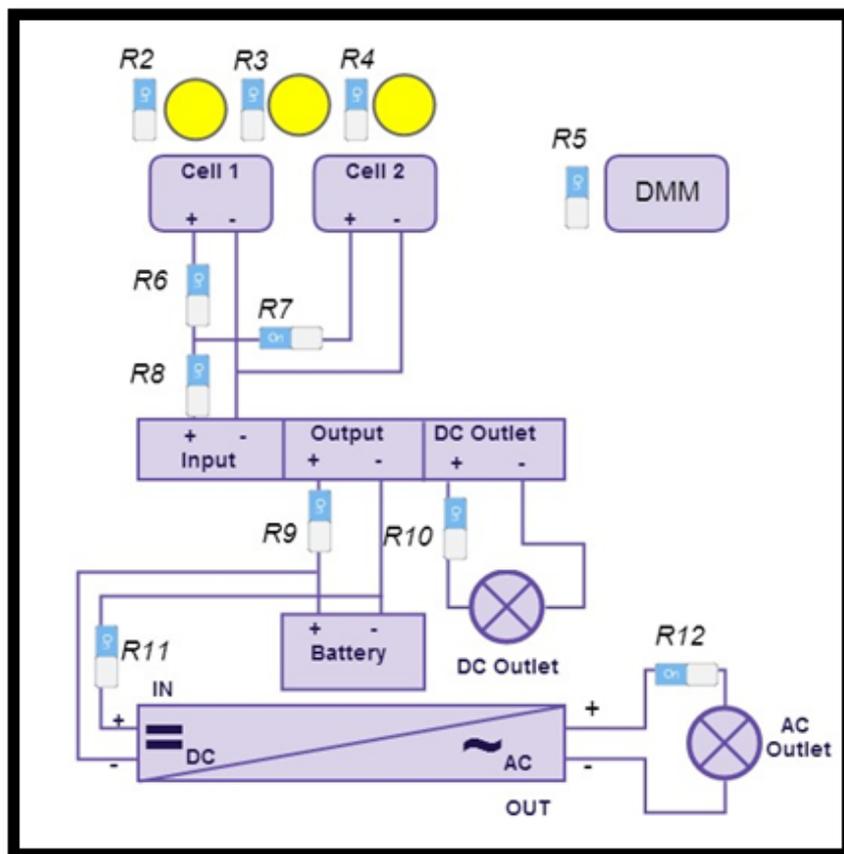


Figure 14. The System connections with the Relays Switching Matrix (R2 – R12)

7. Testing and Results

The results of the system were very satisfactory, meaning that the system is working as expected. The user can access the system remotely using his browser and connect/disconnect the devices depending on the experiment he wants to perform and gets the results immediately. The user-interface is simple and easy to learn and enjoy. The following figure shows the working system, the example demonstrate the measurements of the cell voltages.

8. A Comparative Study: Remote Lab versus Hands-On

One of our objectives of this research study is to find out how much effective the implemented remote lab for solar energy assist students in understanding and reinforcing their theoretical concepts. An obvious way for doing that would be a comparative evaluation, which compares our remote lab system, with its hand-on equivalence. Evaluation of new practical educational systems depends on student surveys to measure the achievement of the required practical skills of the students from these categories of labs, compared with other traditional ones. A survey questionnaire of closed end questions will be implemented and the raw data will be collected in order to investigate student perceptions of their experiences of both the hand-on and the remote lab. Closed-ended questions are questions in which all possible answers are identified, and the respondent is asked to choose one of the answers (Strongly disagreement, Disagreement, Neutral, Agreement, Strongly disagreement). According to Reja et al. [17], closed-ended questions have advantages: closed-ended questions are generally more straightforward and offer choices for respondents, closed questions guide respondents to specific information needed, closed questions permit to ask more questions in less time, and the data (answers) are easy to tabulate, and to analyze [18].

In order to achieve our, educational materials are to be added to shape the teaching part of the system. The teaching material will include the following:

- Introduction to Remote laboratories.
- Remote Laboratories: A new trending method.
- The Solar Energy as an alternative.
- Photovoltaics.
- Photovoltaics cells types and sizes.
- Making a Solar PV System.
- Stand-Alone Photovoltaic Systems
- Grid-connected PV systems.
- From sunlight to Batteries.
- From sunlight to AC.

The material will include various experiments and tasks to be performed on each principle mentioned. Electrical, Electronics, Energy, and Computer Engineering Curricula can include the material to be given to the students.

9. Conclusion and Future Work

To sum up, the main goal of our project was to build a low cost educational environment for the student to access and control the solar energy system remotely, without

restrictions on time and place. And that's what we have achieved by building a system with a simple but powerful user interface that makes it fun and simple to learn. Finally, we can say that we achieved the required goals of the system and the results are satisfactory and promising. Future work will include the following points:

- Modifying the software to handle more than one user (using time slots, Queues, session management and database).
- Modifying the switching matrix to be more efficient and flexible.
- Adding more experiment options such as light filters, a motor to change the angle of the cells.
- Adding more monitoring options to the system such as an Oscilloscope and a Web Camera.
- Expand the system to generate electricity from wind using turbines.
- Adding a temperature sensor to avoid unwanted heat that affects the whole system and the components.
- Changing the microcontroller to Raspberry PI allowing us to wirelessly control the microcontroller without the need for a USB cable, thus more remote environment.

References

- [1] ABET: Accreditation Board for Engineering and Technology, Inc. [Online]. Available at: <http://www.abet.org/about-abet/>.
- [2] S. Dormido, H.Vargas, J. Sánchez, N. Duro, R. Dormido, S. Dormido-Canto, and F. Esquembre, "Using Web-Based Laboratories for Control Engineering Education," International Conference on Engineering Education, Coimbra, Portugal, September, 2007.
- [3] Z. Nedic, J. Machotka, and A. Nafalski, "Remote laboratories versus virtual and real laboratories," Proc. 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, Colorado, USA, November, 2003.
- [4] W. G. Hutzl, A remotely accessed HVAC laboratory for distance education. *Int. J. Eng. Education* 18, 6, pp. 711-716, 2002.
- [5] S. Odeh, "Building reusable remote labs with adaptable client user-interfaces," *Journal of Computer Science and Technology*, vol. 25, no. 5, pp. 999-1015, 2010.
- [6] D. Assante, M. Tronconi, "Photovoltaic System as a Remote Didactic Laboratory for Electrical Engineering Courses", *International Journal of Online Engineering*, vol. 11 (4), 2015.
- [7] R. S. Herrera, M. A. Márquez, A. Mejías, R. Tirado, J. M. Andújar, "Exploring the usability of a remote laboratory for photovoltaic systems", *IFAC-PapersOnLine*, vol. 48(29), pp. 7-12, 2015.
- [8] J. Ma, and J. Nickerson, "Hands-On, Simulated, and Remote Laboratories: A Comparative Literature Review," *ACM Computer Survey*, Vol.38, Issue 3, Article No.7, 2006.
- [9] Z. Nedic, J. Machotka, A. Nafalski, "Remote laboratories versus virtual and real laboratories," in: Proc. of the 33rd ASEE/IEEE Frontiers in Education Conference, pp. T3E-1-6. Boulder, Colorado, USA, November (2003).
- [10] E. Valentin, A. Verbraeck, and H. Sol: Advantages and Disadvantages of Building Blocks in Simulation Studies: Laboratory Experiment with Simulation Experts, *Simulation in Industry 15th European Simulation Symposium*, SCS-European Publishing House, 2003.
- [11] B.Guimaraes, A. Souza, H. Gosmann, and A. Bauchspiess: "Internet Based Remote Laboratory: The Level Control of three coupled water Reservoirs", ACCA, Santiago de Chile, 2002.
- [12] Odeh, S. (2010) "Building reusable remote labs with adaptable client user-interfaces," *Journal of Computer Science and Technology*, vol. 25, no. 5, pp. 999-1015.
- [13] Sommerville I. *Software Engineering*. Addison Wesley, 2016, pp.415-438.
- [14] LINX: Available at: <https://labviewhacker.com>.

- [15] VISA: Available at <http://www.ni.com/visa/>.
- [16] JKI VI Package Manager: Available at <http://jki.net/vipm>.
- [17] U. Reja, K. L. Manfreda, V. Hlebec, and V. Vehovar1: "Open-ended vs. Close-ended Questions in Web Questionnaires", *Advances in Methodology and Statistics (Metodološki zvezki)*, 19, pp.159-177, 2003.
- [18] Abu Shanab, S., Odeh, S., Hodrob, R., Anabtawi, M. (2012). Augmented Reality Internet Labs Versus Hands-On and Virtual Labs: A Comparative Study. In: *The 2012 International Conference on Interactive Mobile and Computer Aided Learning (IMCL)*, 6-8 November, Amman, Jordan, pp. 17-21.