International Journal of Recent Advances in Multidisciplinary Topics
Volume 3, Issue 5, May 2022
https://www.ijramt.com | ISSN (Online): 2582-7839

Real-Time Data Acquisition to Excel and Monitoring Characteristics of Solar Panel Using Arduino

N. Sreeramula Reddy[1], Chittiboina Gowthami[2*], E. Mounika[3], S. Harshad[4], R. V. Pavan Kumar[5]
[1]*Professor, Department of Electrical and Electronics Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, India*
[2,3,4,5]*Student, Department of Electrical and Electronics Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, India*

***Abstract*: This research presents a low-cost virtual instrumentation method for real-time PV panel monitoring of voltage, current, power, and temperature. The system is built around an inexpensive Arduino acquisition board. The data is presented in Excel using the PLX-DAQ data collection Macro, and the acquisition is done with a low-cost current and voltage sensor. The current and voltage sensors are used to determine the PV current and voltage. The output of the two sensors is then sent to the Arduino UNO board's microprocessor. The data collected throughout the acquisition process is recorded and plotted in real time in an Excel sheet. As a result, the I–V and P–V characteristics, which are processed in real-time, can be instantly acquired and plotted on an Excel sheet.**

***Keywords*: PV panel, Voltage, Current, Power, PLX-DAQ Spreadsheet, Serial connectivity, Arduino.**

1. Introduction

Because of their abundance, renewable energy resources are quickly becoming one of the most important sources of energy. Various difficulties confront the energy structure today as society evolves rapidly. The need for renewable energy supplies has increased as a result of growing awareness of pollution and global warming. Because of its availability, diversity, and environmental friendliness, solar energy is the most promising renewable energy source. The evolution and utilisation of this energy not only provide a means of utilising these resources, but also generate an efficient assessment that allows resources to be better adjusted in order to overcome energy resource shortages. Because varied environmental and geographical elements have an impact on resources, measuring solar resources under these conditions makes systems more cost-effective and increases the pace at which renewable solar energy is used. Under these conditions, data acquisition aids in determining the status of solar systems. Because many instruments are used to gather data from the system, the accuracy of the data acquisition system is also vital because different instruments can generate different findings, which can have a major influence on a large-scale solar system. The process of using a computer to measure an electrical or physical event such as voltage, current, temperature, pressure, or sound is known as data acquisition (DAQ). Sensors, DAQ measuring gear, and a computer with programmable software make up a DAQ system. PC-based DAQ systems are more powerful, flexible, and cost-effective than traditional measuring systems because they take advantage of the processing power, productivity, display, and networking capabilities of industry-standard PCs.

Fig. 1. View of solar panels in string

When solar panels are mounted outdoors, they are subjected to a variety of deterioration phenomena caused by factors such as weather (rain, hail, snow, gloomy days), vandalism, dust, bird droppings, shadows from nearby buildings, ageing, and so on. Each of these types of degradation can result in defects and failure, resulting in severe and irreversible damage such as fractures, hot spots, flames, and total solar panel failure. As a result, in order to achieve optimal performance in power generation, it is important to conduct supervision and maintenance to ensure dependability, efficiency, and extended useful life.

2. Software Requirements

Arduino IDE: Arduino IDE (Integrated Development Environment), where IDE stands for Integrated Development Environment, is a political candidate computer code created by Arduino.cc that is primarily used for writing, collecting, and uploading code to the Arduino Device. The majority Arduino modules are compatible with this open-source computer code, which is simple to install and start gathering code on the fly.

*Corresponding author: gowthami2912000@gmail.com

Introduction to Arduino IDE:

The Arduino IDE is a free computer programme that is mostly used for creating and collecting code for the Arduino Module.

- It's an Arduino computer code for a political candidate, making code compilation so simple that even a layperson with no prior technical knowledge may get their feet wet with the teaching method.
- It is available for operating systems such as Mac OS X, Windows, and UNIX and operates on the Java Platform, which has constitutional functions and commands that are useful for debugging, writing code, and collecting data in the environment.
- A wide range of Arduino modules are available, including the Arduino Uno, Arduino Mega, Arduino Technologist, Arduino Small, and many others.
- On the board of each of them is a microcontroller that has been fully coded and accepts the data contained within the code.
- The core code, also known as a sketch, written on the IDE platform eventually generates a Hex File, which is then copied and uploaded into the board's controller.
- The IDE environment consists of two basic parts: Editor and Compiler, with the former being used to write the specified code and the latter being used to collect and upload the code into the given Arduino Module.
- This environment is compatible with both C and C++.

PLX-DAQ Spread Sheet:

PLX-DAQ is a Microsoft Excel add-on utility for Parallax microcontroller data acquisition. Data may now be sent directly into Excel from any of our microcontrollers linked to any sensor and a PC's serial connection.

The PLX-DAQ Excel MacrHo collects data from an Arduino microcontroller and sends it to an Excel spreadsheet. All we have to do now is download it. Following installation, a folder entitled "PLX-DAQ" will be generated on the PC, containing a shortcut named "PLX-DAQ Spreadsheet." Then, in the PLX-DAQ window, we simply need to open the Spreadsheet and define the connection settings (Baud rate and port) to create communication between the board and Excel.

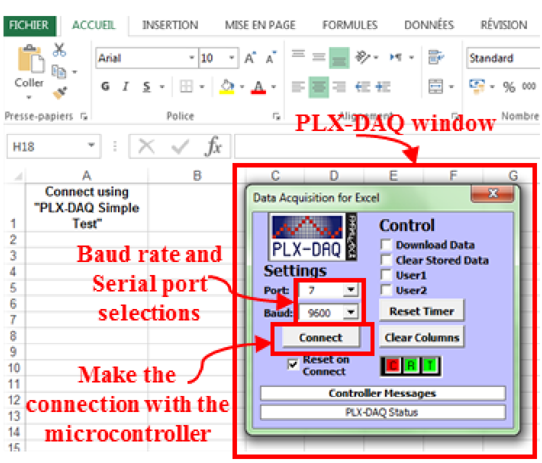


Fig. 2. PLX-DAQ is a Parallax microcontroller data acquisition

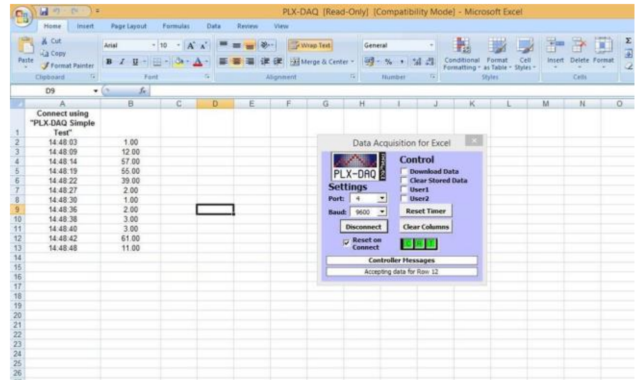


Fig. 3. Data from will be logged into the excel file every time

Every time, the data from will be logged into an excel file. You can collect as much data as you want with any other sensor (limit up to an excel column). We now require an application called PLQ-DAQ to collect real-time data into an excel file. This allows arduino to save the data to an excel spreadsheet.

3. Hardware Requirements

A. *Solar Panel*

A solar cell, also known as a photovoltaic cell, is an electrical device that uses the photovoltaic effect, a physical and chemical phenomenon, to convert light energy directly into electricity. It's a type of photoelectric cell, which is described as a device with electrical characteristics that change when exposed to light, such as current, voltage, or resistance. Modules, often known as solar panels, are made up of individual sun cell units.

Solar energy is the use of the sun's radiant light and heat in a variety of ways, including photovoltaics, solar thermal energy, solar architecture, and solar heating. It is a significant renewable energy source, and its technologies are classified as passive or active solar depending on how they gather and distribute solar energy or convert it to solar power. To harness the energy, active solar solutions include photovoltaic systems, concentrated solar power, and solar water heating. Orienting a structure to the Sun, selecting materials with favorable thermal mass or light-dispersing qualities, and designing rooms that naturally circulate air are all examples of passive solar approaches.

Fig. 4. Solar panel

At the upper atmosphere, the Earth gets 174 petawatts (PW) of solar radiation (insolation). The rest is absorbed by clouds, oceans, and land masses, with around 30% being reflected back to space. At the Earth's surface, solar light is predominantly in the visible and near-infrared bands, with a minor portion in the near-ultraviolet. The majority of the world's population lives in locations with insolation values of 150–300 watts persquare metre per day, or 3.5–7.0 kWh per square metreper day. "The development of affordable, limitless, and clean solar energy technology will have tremendous longer-term benefits," the International Energy Agency stated in 2011. It will improve countries' energy security 11 by relying on an indigenous, inexhaustible, and generally import-free resource, as well as raise sustainability, reduce pollution, minimise the costs of addressing global warming, and keep fossil fuel prices lower than they would be otherwise. These benefits are widespread. As a result, the higher expenses of early deployment incentives should be viewed as learning investments that must be wisely spent and widely shared.

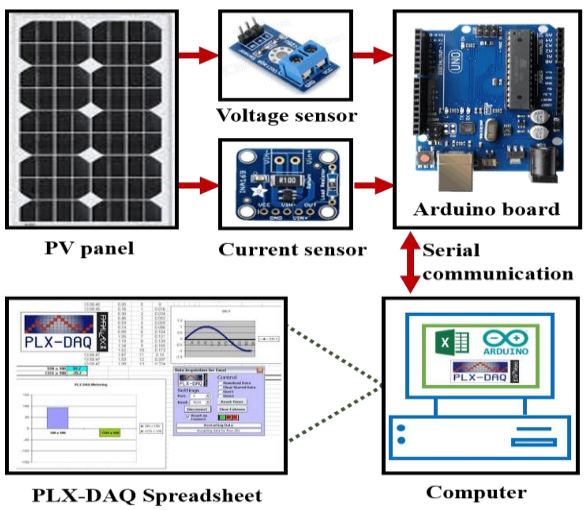


Fig. 5. Block diagram

This research presents a low-cost virtual instrumentation method for monitoring PV panel properties such as voltage, current, and power in real time. The system is built around an inexpensive Arduino acquisition board. The data is collected using low-cost current and voltage sensors, and the PLX-DAQ data acquisition macro is used to report the results in Excel.

The equipment's structure is depicted in the diagram below. The current and voltage sensors are used to determine the PV current and voltage. The output of the two sensors is then sent to the Arduino UNO board's microprocessor. The data collected throughout the acquisition process is recorded and visualised in real time in an Excel spreadsheet.

B. Current Sensor

A current sensor is a device that detects and generates a signal proportionate to the current flowing through a wire. An analogue voltage or current, or even a digital output, could be generated. The resulting signal can then be used to display the measured current in an ammeter, be stored in a data acquisition system for further analysis, or be utilised for control. The INA169 module (Figure (a)) is a current sensor that can detect a continuous current up to 5 A and is used to monitor the PV panel output current. The INA169 current sensor circuit is shown in Figure (b) (from INA169 Datasheet). The INA169 is a current monitor that monitors the voltage drop across a sensing resistor on the high side (Rs). The output resistor then generates a voltage level (Vo) (RL).

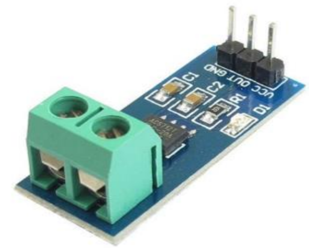


Fig. 6. Current sensor

C. Voltage Sensor

A voltage sensor is a device that measures and calculates the amount of voltage in an object. Voltage sensors can tell whether the voltage is AC or DC. The voltage is the sensor's input, while the switches, analogue voltage signal, current signal, or audible signal are the sensor's output.

Sensors are electronic or optical devices that can detect, recognise, and react to certain electrical or optical impulses. Voltage sensor and current sensor approaches have shown to be an excellent alternative to traditional current and voltage monitoring methods. The B25 voltage sensor module (Figure (a)) is used to measure the output voltage of the PV panel. The B25 voltage sensor works on the voltage divider principle; it's essentially a voltage divider with 30 k and 7.5 k resistances, as shown in Figure (b). Because the Arduino microcontroller's maximum analogue input voltage is 5 V, it is used to reduce the input voltage by up to 5 times compared to the original value. As indicated in Figure, the voltage sensor module will be connected to the PV panel load in parallel (c). The sensor value (V_{out}) is a digital value that ranges from 0 to 1023. The resolution of the PV panel module voltage is 0.00489 V (5/1023) since the microcontroller's ADC is written in 10 bits, and the input voltage of this module must be greater than 0.02445 V (0.00489 V 5). As a result, the PV panel's output voltage can be calculated using the following equation: $V=5*V_{out}*(5/1023)$ Voltage Sensors Have Several Advantages Over Traditional Measuring Techniques.

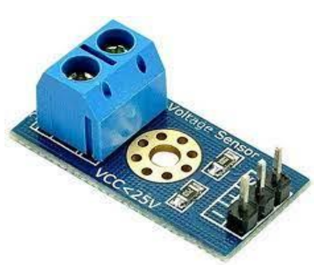


Fig. 7. Voltage sensor

4. Experimental Setup of the Virtual Instrumentation System

The Arduino board's microprocessor reads the output voltage and current from the PV panel, which are recorded by sensors, and then computes the output power. We open the PLX-DAQ Excel Macro after connecting the Arduino board to the computer via USB connection, and we define the serial port where the Arduino board is linked to the computer, as well as the Baud rate (9600 bits/sec) in the PLX-DAQ window following its display. It's worth noting that the Baud rate set in the PLX-DAQ window must match the one used in the Arduino board's software code. The output data will be collected and shown in real-time on the Excel Spreadsheet after selecting "connect." The light intensity is controlled by adjusting a variable resistance between 0 and 330 ohms manually (to trace the I-V and P-V characteristics). If necessary, a pyranometer is employed to detect light radiation. The microcontroller is set up to take measurements of PV current, voltage, and power every second.

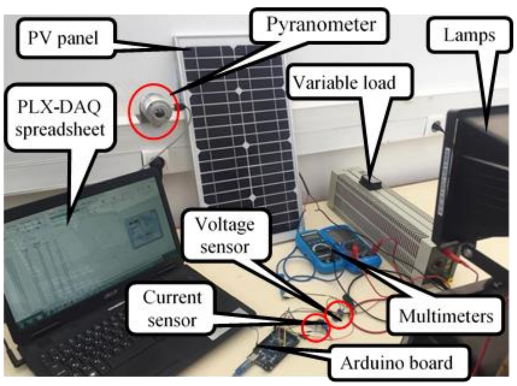
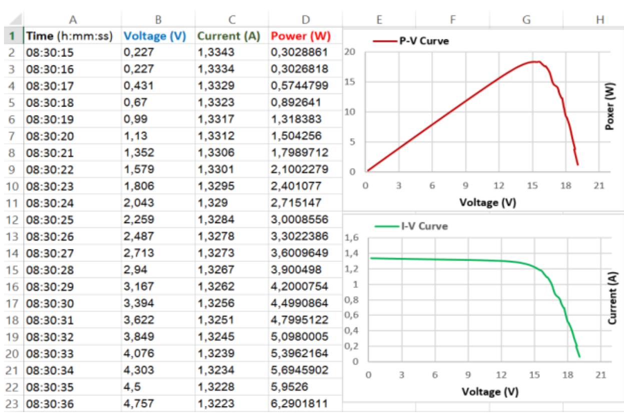


Fig. 8. Experimental results

The PV panel's I-V and P-V properties, as determined by our virtual instrumentation, are shown in the diagram below.



	A	B	C	D
1	Time (h:mm:ss)	Voltage (V)	Current (A)	Power (W)
2	08:30:15	0,227	1,3343	0,3028861
3	08:30:16	0,227	1,3334	0,3026818
4	08:30:17	0,431	1,3329	0,5744799
5	08:30:18	0,67	1,3323	0,892641
6	08:30:19	0,99	1,3317	1,318383
7	08:30:20	1,13	1,3312	1,504256
8	08:30:21	1,352	1,3306	1,7989712
9	08:30:22	1,579	1,3301	2,1002279
10	08:30:23	1,806	1,3295	2,401077
11	08:30:24	2,043	1,329	2,715147
12	08:30:25	2,259	1,3284	3,0008556
13	08:30:26	2,487	1,3278	3,3022386
14	08:30:27	2,713	1,3273	3,6009649
15	08:30:28	2,94	1,3267	3,900498
16	08:30:29	3,167	1,3262	4,2000754
17	08:30:30	3,394	1,3256	4,4990864
18	08:30:31	3,622	1,3251	4,7995122
19	08:30:32	3,849	1,3245	5,0980005
20	08:30:33	4,076	1,3239	5,3962164
21	08:30:34	4,303	1,3234	5,6945902
22	08:30:35	4,5	1,3228	5,9526
23	08:30:36	4,757	1,3223	6,2901811

Fig. 9. Virtual instrumentation

The results of a test similar to the previous one is displayed in Figure below, with the difference being the time step between each measurement, which was reduced from 1 second to only 100 milliseconds. Oscillations have formed on the I-V and P-V curves due to inaccuracy of data obtained by the instrument system, as illustrated in this figure, but only to a minor extent. A small step size, on the other hand, results in a large sample of measurements and, as a result, too many significant outcomes. As a result, there must be a compromise between small and large step sizes. It is often advised to choose a lower step size if you want to observe accurate changes in PV properties. Use a large step size if you don't care about the precision of the modifications and just want to get the instrument system running faster.

The results of a monitoring test for current, voltage and power of PV panel are presented in the Figure below. From the experimental results, it can be seen that the PV panel produced a maximum power of 17.07 W at "15h14min02s" when a voltage of 14.15 V and a current of 1.20 A appear. Subsequently, the output power is tends to a minimum value 822.2 mW when there is a voltage of 18.23 V and a current of 45.1 mA. Hence, as the present system is used such as a virtual instrument to acquire the PV panel characteristics under the real operation conditions, it can also be used on field periodical monitoring activities for PV systems.

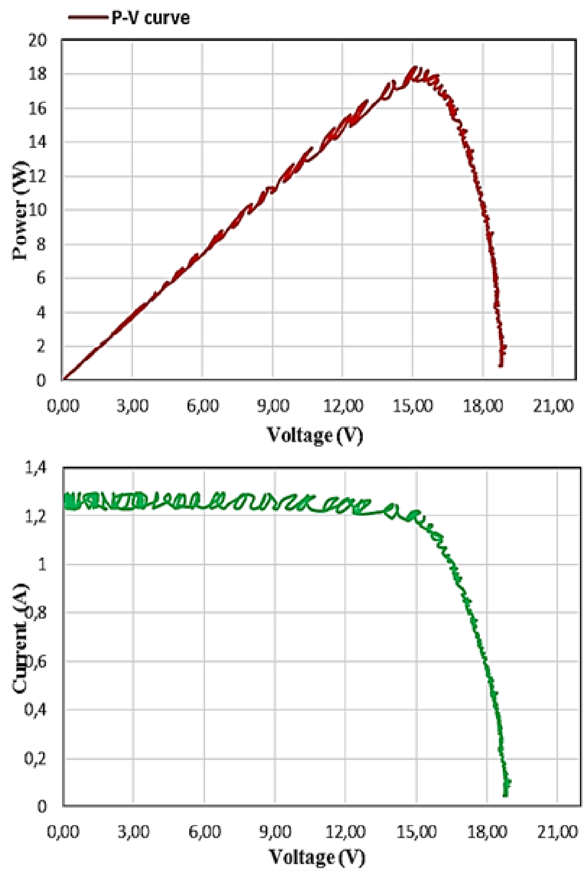


Fig. 10. The I-V and P-V characteristics of the PV panel

A. Code

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <dht11.h>
#include <LiquidCrystal.h>
#include <Arduino.h>
//#include <ESP8266WiFi.h>
//#include <BlynkSimpleEsp32.h>
#define DHT11PIN 4
// Define analog input
#define ANALOG_IN_PIN A0
//#define analogIn A1
const int analogIn = A1;
double mVperAmp = 185; // use 100 for 20A Module and 66 for 30A Module
double RawValue= 0;
```

```
double ACSoffset = 2500;
double Voltage = 0;
double Amps = 0;
double power = 0;
String Scroll_LCD_Left[120];

dht11 DHT11;
// Floats for ADC voltage & Input voltage
float adc_voltage = 0.0;
float in_voltage = 0.0;

// Floats for resistor values in divider (in ohms)
float R1 = 30000.0;
float R2 = 7500.0;

// Float for Reference Voltage
float ref_voltage = 5.0;

// Integer for ADC value
int adc_value = 0;
LiquidCrystal_I2C lcd(0x27, 16, 2); // I2C address 0x27, 16 column and 2 rows

void setup()
{
  // Setup Serial Monitor
  Serial.begin(9600);
  lcd.init(); // initialize the lcd
  lcd.begin(16, 2);
  lcd.backlight();
  lcd.setCursor(0,0);// move cursor to   (0, 0)
  lcd.print("RTDA SOLAR PANEL");
  //lcd.autoscroll();    // Set diplay to scroll automatically
  delay(3000);
  lcd.clear();
  lcd.setCursor(3,0);// move cursor to   (0, 0)
  lcd.print("AITS::RAJAMPET");
  lcd.setCursor(0,1);        // move cursor to   (0, 1)
  lcd.print("DEPARTMENT OF EEE");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);        // move cursor to   (0, 0)
  lcd.print("PROJECT BATCH 13");
  lcd.setCursor(0,1);        // move cursor to   (0, 1)
  lcd.print("1.C.GOWTHAMI(19705A0213)");
  delay(3000);
  lcd.clear();
lcd.setCursor(0,0);        // move cursor to   (0, 0)
  lcd.print("2.E.MOUNIKA");
  lcd.setCursor(0,1);        // move cursor to   (0, 1)
  lcd.print("3.RV PAVAN");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);        // move cursor to   (0, 0)
  lcd.print("4.S HARSHAD");
  delay(3000);
  lcd.clear();
  Serial.println("CLEARDATA");       // Clear all Excel sheet data
  Serial.println("LABEL,ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES::RAJAMPET");
  // Label columns: A for date, B for time, C for temperature and D for humidity
  Serial.println("LABEL,Date,Time,Humidity(h),Temperature Celsius(°C),DC Voltage(V),DC Current(A),DC Power(W)");
 }
void publish()
{
  lcd.setCursor(0,0);// move cursor to   (0, 1)
  lcd.print("Voltage(V):");
  lcd.print(in_voltage,1);
  lcd.setCursor(0,1);        // move cursor to   (0, 1)
  lcd.print("Current(A):");//
  lcd.print(Amps,1);
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);// move cursor to   (0, 1)
  lcd.print("Power(W):");
  lcd.print(power,1);
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);// move cursor to   (0, 1)
  lcd.print("Humidity:");
  lcd.print((float)DHT11.humidity);
  lcd.setCursor(0,1);// move cursor to   (0, 1)
  lcd.print("Temperature:");
  lcd.print((float)DHT11.temperature);
  delay(3000);
  lcd.clear();
}

void loop()
{
  Serial.println();
  int chk = DHT11.read(DHT11PIN);
  // Read the Analog Input
   adc_value = analogRead(ANALOG_IN_PIN);
   // Determine voltage at ADC input
   adc_voltage  = (adc_value * ref_voltage) / 1024.0;
   // Calculate voltage at divider input
   in_voltage = adc_voltage / (R2/(R1+R2));     //PV panel voltage
   RawValue = analogRead(analogIn);
   Voltage = (RawValue / 1024.0) * 5000;        // Gets you mV
   Amps = ((Voltage - ACSoffset) / mVperAmp);   //PV panel current
   power = in_voltage*Amps; //PV panel power
   Serial.print("DATA,DATE,TIME,"); // Write date and time on row A and row B respectively
   Serial.print((float)DHT11.humidity,2);
   Serial.print(",");  // Move to next column
   Serial.print((float)DHT11.temperature,2);
   Serial.print(",");  // Move to next column
   Serial.print(in_voltage, 2); // Print results to Serial Monitor to 2 decimal places
   Serial.print(",");    // Move to next column
   Serial.print(Amps,2); // the '2' after current allows you to display 2 digits after decimal point
   Serial.print(",");    // Move to next column
   Serial.print(power,2); // the '2' after power allows you to display 2 digits after decimal point
   Serial.println(",");  // Move to next column
   delay(5000);
  publish();
}
```

5. Conclusion

The modelling of real-time data gathering of solar panels is presented in this research. The real-time voltages of solar panels are imported from Arduino UNO to an excel sheet in the preceding study. This voltage varies with respect to atmospheric conditions such as solar light irradiation and ambient temperature. The graph above depicts the voltage variation and behaviour of the solar panel over time and days. The voltage of the solar panel fluctuates due to the low and high intensity of the sunrays. The aforementioned results clearly show this. The entire simulation was successfully simulated in LabVIEW with the help of Arduino. Every time the servomotor is started from its position, it must be calibrated.

References

[1] Camacho Olarte, J. (2020). Use of free tools to design a low-cost physical variables monitoring system based on embedded systems.

[2] Sharma, S, Jain, K. and Sharma, A. (2015) Solar Cells: In Research and Applications—A Review. Materials Sciences and Applications, 6, 1145-1155.

[3] Raja Paul, M.M.; Mahalakshmi, R.; Karuppasamypandiyan, M.; Bhuvanesh, A.; Ganesh, R.J. Classification and Detection of Faults in Grid Connected Photovoltaic System. Int. J. Sci. Eng. Res. 2016, 7, 149–154.

[4] Bedi, G.; Venayagamoorthy, G.K.; Singh, R.; Brooks, R.R.; Wang, K. Review of Internet of Things (IoT) in Electric Power and Energy Systems. IEEE Internet Things J. 2018, 5, 847–870.

[5] Shafique, K.; Khawaja, B.A.; Sabir, F.; Qazi, S.; Mustaqim, M. Internet of Things (IoT) for Next-Generation Smart Systems: A Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios. IEEE Access 2020, 8, 23022–23040.