NIU Journal of Humanities Copyright©2024
Nexus International University ISSN: *3007-1704;* 9(4): 51-60

Design and Implementation of an Arduino-Based MPPT Solar Charge Controller

MATTHEW B. OLAJIDE, STEPHEN ADEYOYE, KEHINDE OTESILE
Olabisi Onabanjo University, Ago-Iwoye, Nigeria

DAVID S. KUPONIYI
Gateway (ICT) Polytechnic Saapade, Nigeria

CHARITY S. ODEYEMI
Federal University of Technology Akure, Nigeria

Abstract. An effective solar charge controller's design is essential to maximizing the energy that photovoltaic (PV) systems can capture. This paper describes the design and implementation of an Arduino microcontroller-based Maximum Power Point Tracking (MPPT) charge controller. The purpose of this design is to improve solar energy systems' efficiency and dependability, especially in large-scale or high-current applications. The LM7815 regulator is used by the charge controller to control voltage, and high-power transistors (2SC5200) are used to boost current in order to provide a high load capacity. In order to guarantee optimal efficiency, MPPT controllers dynamically modify the solar panels' operating point to optimize the energy gathering process. The Arduino-based controller is open-source and inexpensive; it can be used in off-grid solar power systems. The technology can manage a maximum current of 60A and charge a battery bank effectively. The research analyzed the circuit design, control algorithms, and experimental findings, and real-world testing is used to assess the system's performance. The result was very impressive; the system proved to be efficient, reliable, and cost-effective, making it a viable solution for off-grid and high-power solar applications.

Keywords: MPPT, Arduino, charge controller, solar energy, photovoltaic systems, renewable energy, battery charging, Solar Power, Current Boosting

1. Introduction

The efficiency of solar charge controllers has grown in significance due to the rising demand for renewable energy sources, particularly solar energy, which is still gaining popularity. By optimizing the voltage and current at the operating point, MPPT charge controllers enable solar systems to derive the most power from photovoltaic arrays, even in the face of variable variables like temperature fluctuations and variations in sunlight intensity (Viraj Nijap, *et al.*, 2024). The system makes use of an LM7815 voltage regulator to supply a constant 15V supply, 2SC5200 NPN transistors for current amplification, and an Arduino microcontroller to carry out the MPPT algorithm. We used an Arduino Uno in the design to make the design more effective (Mairizwan, *et al.*, 2021). The system entails with multiple protections to shield the circuitry from aberrant conditions. Some of its features include an LCD display and LED indication. This design can be used to charge a standard 12V lead acid battery with a 50W to 600W solar panel. Accurately tracking the maximum power point (MPP) is crucial for the design of effective photovoltaic (PV) power generation systems, since the MPP varies with variations in atmospheric variables, such as temperature and sun radiation. To regulate the output, we have designed the most used MPPT method, Perturb and Observe (PO) (Nandini Pundir. 2020).

1.1 Maximum Power Point Transfer

A method for maximizing power output using wind turbines and photovoltaic (PV) solar systems is called maximum power point tracking, or MPPT. PV solar systems can be found in a variety of configurations. In its most simple form, power is sent straight from collector panels to the DC-AC solar inverter, which then feeds it into the power grid (Noor Hasliza Abdul Rahman, *et al.*, 2020). An alternative variant known as a hybrid inverter has the potential to divide the electricity at the inverter so that some power is sent to the grid and the rest is sent to a battery bank. The third version uses a specialized PV inverter with the MPPT but has no grid connectivity at all. A battery bank receives electricity directly in this design (Chowdhury *et al.*, 2016).

This research focuses exclusively on MPPT application for photovoltaic systems. As shown in figure 1. I-V curve can be used to evaluate the complex relationship between temperature and total resistance in solar cells, which results in a non-linear output efficiency. The MPPT system's main function is to sample the PV cells' output and apply the appropriate resistance (load) in order to maximize power under any given set of environmental variables. The product of MPP voltage (V_{mpp}) and MPP current (Impp) is MPP (Maximum Power Point) (Hohm D, Ropp M *et al.*, 2003).

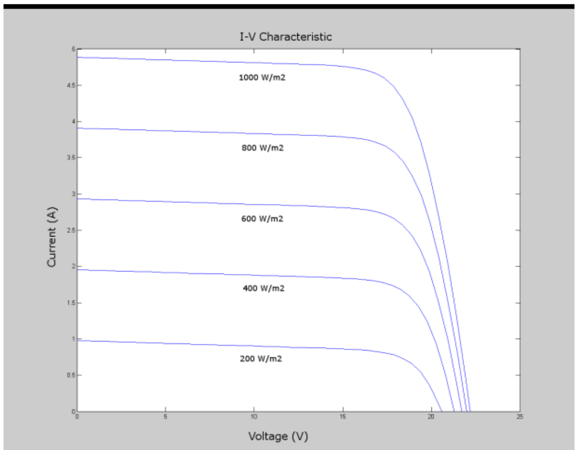

Figure 1a

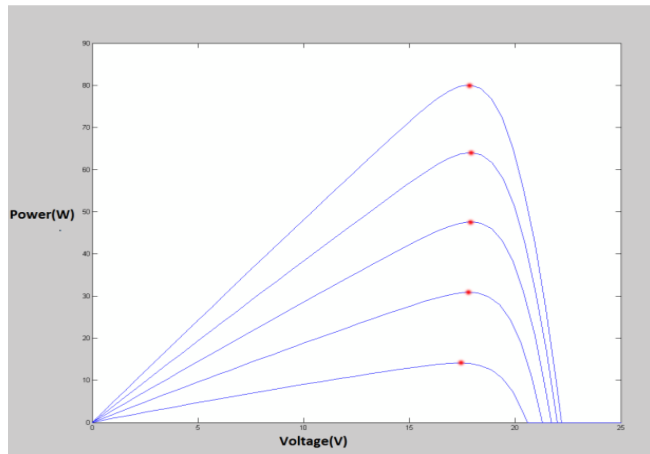

Figure 1b

Figure 1.(a) IV characteristics of PV panel for different irradiance level. (b) PV characteristics corresponding to IV characteristics in (a). Red dot shows the Maximum power point (MPP).

(Hussein KH, et al. 1995) compares and evaluates the percentage of power extraction with MPPT and without MPPT. It clearly shows that when we use MPPT with the PV system, the power extraction efficiency is increase to 97%. The study of developing a PV charging system for li-ion batteries by integrating MPPT and charging control for the battery is reviewed (Chowdhury, *et al.*, 2016).

2. Design Analysis

2.1 Arduino Microcontroller

At the core of the MPPT charge controller is the Arduino, which executes a Perturb and Observe (P&O) algorithm to track the maximum power point of the solar panel. Arduino is chosen for its simplicity, flexibility, and ability to handle real-time data processing necessary for MPPT.

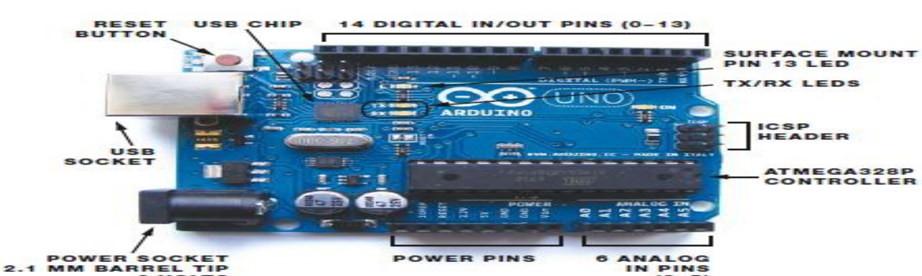

Figure 2: *Arduino UNO development board.*

2.2 Perturb & Observe Algorithm

Perturb & observe (P&O) algorithm's simplicity and ease to implement, is sometimes referred to as the "hill climbing" method and is the most widely used algorithm in practice. The P&O algorithm functions in this way in its most basic version. Assuming the PV module is functioning at a position that is away from the MPP, This algorithm modifies the PV module's working voltage by a tiny amount, and it then measures the change in power, or P. It is assumed that the operating point has been shifted closer to the MPP if the P is positive. The operational point should therefore advance toward the MPP as a result of additional voltage disturbances in the same direction. In the event that P is negative, the operating point has departed from the MPP; hence, the perturbation should be reversed in order to return to the MPP [8-11]. Figure 2, present the flowchart of the algorithm.

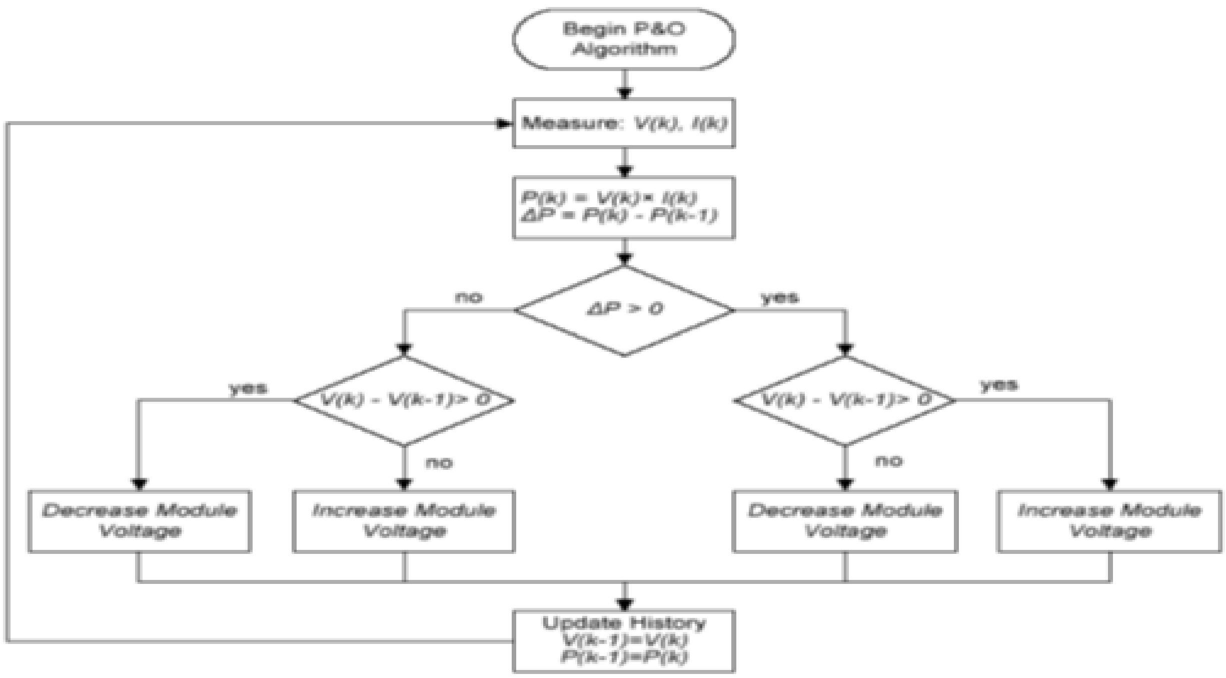

Figure 3: Flow Chart of PO algorithm used in our MPPT charge controller.

2.2.1 MPPT Algorithm

The P&O algorithm periodically adjusts the operating voltage and observes the corresponding change in power output. If the power increases, the adjustment continues in the same direction; if it decreases, the direction is reversed. The Arduino is programmed to manage this operation, with readings taken from the voltage and current sensors connected to the PV panel.

2.3 LM7815 Voltage Regulator

The LM7815 is a linear voltage regulator that ensures a stable 15V output to the control circuitry. The stability of this voltage is crucial for maintaining consistent operation, especially in high-power systems where voltage fluctuations can lead to component failure or inaccurate MPPT performance.

2.4 SC5200 Transistors for Current Boosting

The 2SC5200 NPN transistors are used to boost the current-handling capacity of the system. With a maximum collector current of 15A, multiple transistors are configured in parallel to handle the total current load of 60A. These transistors amplify the current while maintaining high efficiency, enabling the controller to manage large power outputs required by the load or battery bank.

2.5 Circuit Design

The schematic of the charge controller includes the following main sections:

Input Stage: The solar panel is connected to the input stage, where a voltage sensor reads the panel voltage. A current sensor measures the output current to the battery.
Power Regulation Stage: The LM7815 regulator provides a stable 15V supply for the Arduino and control logic. The Arduino executes the MPPT algorithm and adjusts the duty cycle of the MOSFET in the buck converter.
Current Amplification Stage: The 2SC5200 transistors are used in parallel to handle the large current demands of the system. Heat sinks are incorporated to dissipate heat generated during operation, ensuring thermal stability.
Output Stage: The output stage is connected to the battery bank, where the voltage and current are regulated to match the power requirements of the battery system.

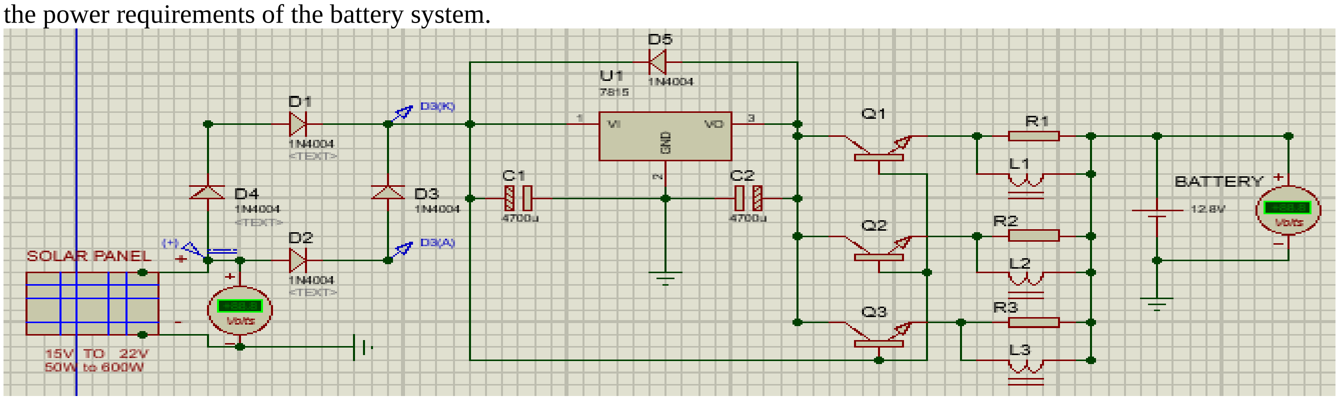

Figure 4: *Controller voltage regulation circuit simulation*

In the circuit above, there is a full bridge rectification arrangement of diode, then it is fed into the 15v regulator to regulate for the transistor to do the current amplification from collector pin to emitter, CE, and the base of the transistor is activated with unregulated voltage from the rectifier.

2.5.1 Voltage divider circuit for Arduino

The circuit below shows the voltage division for Arduino microcontroller which understand only 0v - 5v

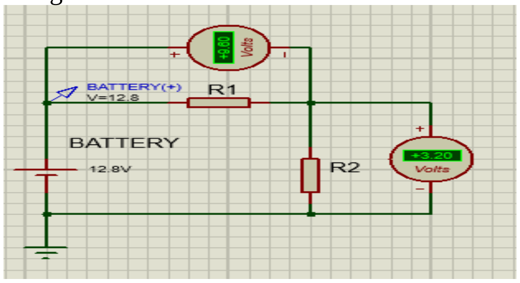

Figure 5: Voltage divider circuit

From figure above, the value of R_1 and R_2 is unknown, to calculate this, we will use the voltage divider formular,

$$V_{out} = V_{in} \left(\frac{R1}{R1+R2}\right) \quad (1)$$

From research, V_{out} can either be voltage across R1 (V_{R1}) or voltage across R2(V_{R2}), but in this work we need 5v maximum to be delivered to Arduino which is voltage $V_{R2,}$ therefore we will substitute for both V_{R1} and V_{R2} as output voltage while
V_{in} from the controller circuit is regulated 15v
Since 5v should be V_{R2} meant for Arduino and the resistors are arrange in series, so

$$V_{in} = V_{R1} + V_{R2} \quad (2)$$

$15v = V_{R1} + 5$

$V_{R1} = 10v$

Below are the parameters to be substituted

$V_{out1} = V_{R1} = 10v$, $V_{out2} = V_{R2} = 5v$, $V_{in} = 15V$

Therefore;

$V_{R1} = V_{in} \left(\frac{R1}{R1+R2}\right)$ (3)

$V_{R2} = V_{in} \left(\frac{R2}{R1+R2}\right)$ (4)

Making V_{in} the subject of formular in both equation (3) and (4), and there after equating them, gives
$V_{R1}\left(\frac{R1+R2}{R1}\right) = V_{R2} \left(\frac{R1+R2}{R2}\right)$

The ratio is;
$\frac{V_{R1}}{V_{R2}} = \frac{R_1}{R_2}$ (5)
Therefore by substitution,

$$\frac{10}{5} = \frac{R1}{R2} = 2$$

The ratio of R1 to R2 is 2
This means if
R1 = 1kΩ also R1 = 2 kΩ and so on with ratio of 2
R2 = 2 kΩ R2 = 4 kΩ
This is verified in the circuit simulation below

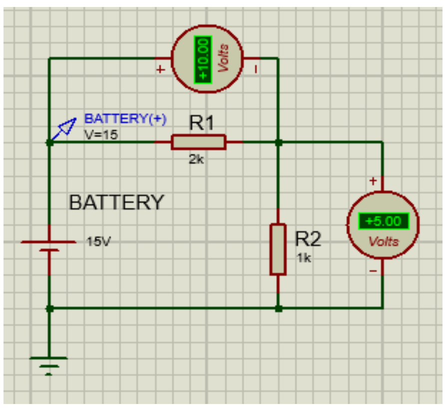

Figure 6a

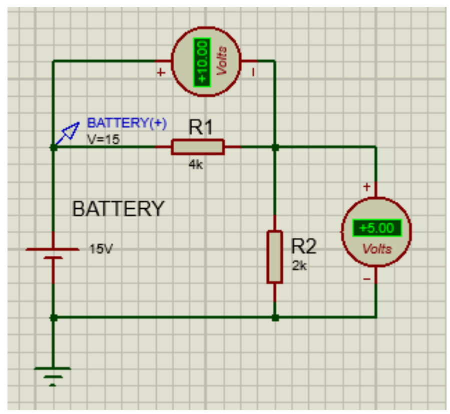

Figure 6b

Figure 6: *Verifying derived formular for voltage division ratio using proteus*

2.5.2 Software code algorithm

The brain of the system is the Arduino Uno ATme ga328P which is a programmable microcontroller, the code is written using Arduino idle in C++ language.
The execution code algorithm of the system is presented in Table.1

Table 1: Arduino code for the system

```
#include <LiquidCrystal.h>
#define BAUDRATE 9600
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
float cvtp1;
int vp1 = 0;
int lpin = 13;
int led1 = 7;
int led2 = 8;
int led3 = 9;
int led4 = 6;
int rl1 = 10;
int R1 = 2000;
int R2 = 1000;
int R3 = 100;
int R4 = 300;
float C1 = 0.00;
void setup() {
  // put your setup code here, to run once:
 Serial.begin(9600);
 lcd.begin(16, 2);
 lcd.setCursor(0,0);
  lcd.print("MPPT Charge Cntrl");
  lcd.setCursor(1,3);
  lcd.print("Made In Ibogun");
  delay(3000);
  lcd.clear();
  pinMode (led1, OUTPUT);
  pinMode (led2, OUTPUT);
  pinMode (led3, OUTPUT);
  pinMode (led4, OUTPUT);
  pinMode (rl1, OUTPUT);
}
  void loop() {
   // put your main code here, to run repeatedly:
  cvtp1 = analogRead(vp1);
  float DCvolt = cvtp1*(5.0/1024)*((R1+R2)/R2)+C1;
  float V=DCvolt;

  if (V >=10.4 && V <=11.0){
   digitalWrite(led1, HIGH);
   digitalWrite(led2, LOW);
   digitalWrite(led3, LOW);
   digitalWrite(led4, LOW);
   digitalWrite(rl1, LOW);
   Serial.print("Dc volt = ");
   Serial.println(DCvolt,1);
   Serial.println("Low Voltage");
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print("Batt volt:");
   lcd.print(DCvolt,1);
   lcd.print("v");
   lcd.setCursor(2,3);
   lcd.print("Low Voltage");
   delay(1000);
  }
  else if (V > 11.0 && V <=14.0){
   digitalWrite(led3, HIGH);
   digitalWrite(led1, LOW);
   digitalWrite(led2, LOW);
   digitalWrite(led4, LOW);
   digitalWrite(rl1, LOW);
   Serial.print("Dc volt = ");
   Serial.println(DCvolt,1);
   Serial.println("Full Voltage");
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print("Batt volt:");
   lcd.print(DCvolt,1);
   lcd.print("v");
   lcd.setCursor(2,3);
   lcd.print("Full Voltage");
   delay(1000);
  }
  else if (V >14.0){
  digitalWrite(led2, LOW);
  digitalWrite(led1, LOW);
  digitalWrite(led3, LOW);
  digitalWrite(led4, HIGH);
  digitalWrite(rl1, LOW);
   Serial.print("Dc volt = ");
   Serial.println(DCvolt,1);
   Serial.println("Normal Voltage");
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print("Batt volt:");
   lcd.print(DCvolt,1);
   lcd.print("v");
   lcd.setCursor(2,3);
   lcd.print("NO BATT/LOAD");
   delay(1000);
  }
  else if (V <10.4){
   digitalWrite(led1, HIGH);
   digitalWrite(led2, LOW);
   digitalWrite(led3, LOW);
   digitalWrite(led4, LOW);
   Serial.print("Dc volt = ");
   Serial.println(DCvolt,1);
   Serial.println("Batt too low");
     Serial.println("system shut down for 20min");
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print("Batt volt:");
   lcd.print(DCvolt,1);
   lcd.print("v");
   lcd.setCursor(0,3);
   lcd.print("Batt too low");
   delay(3000);
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print("battery too low");
   lcd.setCursor(0,3);
   lcd.print("Switch off load");
   lcd.autoscroll();
   digitalWrite(rl1, HIGH);
   delay(5000);
}
}
```

2.6 Implementation Stage

Getting the component in line with the appliance's specification was the most crucial task throughout the implementation stage. Additionally, there are two primary sub stages: the hardware implementation stage and the software implementation stage. Figure.7, present simulation circuit diagram of the complete system.

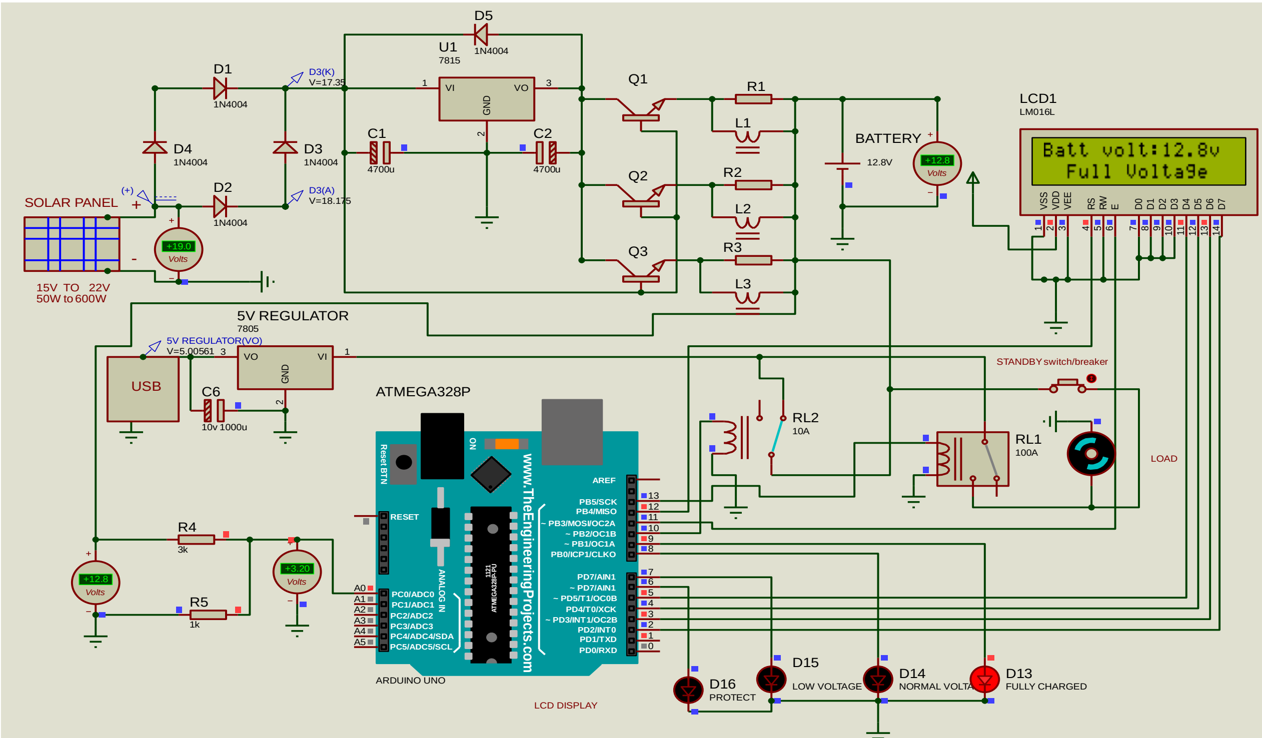

Figure 7: *Simulation of the system*

2.8 Circuit Overview

Solar Panel Input: The solar panel must provide a voltage higher than 15V to allow the LM7815 to regulate properly. Typically, this would be around 22V to account for voltage drops.

LM7815 Voltage Regulator: The LM7815 will regulate the voltage down to 15V. However, by itself, it can only supply up to 1.5A. To achieve 60A, we will need to use 2SC5200 transistors in parallel as current boosters.
2SC5200 Transistors: These transistors will take over the majority of the current load. Each 2SC5200 can handle a significant amount of current, depending on cooling, but for safety, we'll assume each transistor can handle around 15A.

2.9 Circuit Analysis

2.9.1: Calculations

From the circuit in Fig8 above;
Lm7815 and 2SC5200 will dissipate power as heat given by;
$P_{out} = (V_{in} - V_{out}) \times I_{out}$ (6)
Since V_{in} = 22v max
and V_{out} = 15v regulated
For Lm7815; without transistor yet
$I_{7815} = I_{out} = 1.5A$
$P_{dissipated} = (22 - 15) \times 1.5 = 10.5W$

The Lm7815 can drive the base of 2SC5200 transistor, allowing higher current through the C-E path
The output current can be significantly increased depending on how many transistor parallel
$P_{out} = V_{out} \times I_{out}$ (7)
$= 15 \times 1.5 = 22.5$
Note: 15v regulated voltage is fed into the transistor, and practically there is a drop along C-E known as V_{CE} and also across the diodes to prevent reverse voltage

Therefore, input voltage is 15v, while the output measured voltage is 14.2v
For 2SC5200; using transistor as current booster
Since the maximum current needed is 60A, but each transistor can give 15A, therefore

Number of transistor = $\frac{\textbf{total current}}{\textbf{individual current}}$ (8)

$= \frac{60}{15} = 4$

So four (4) to five can be used
With 15v input from regulator and 14.2V output, for 60A:
For individual transistor (2SC5200)
$P_{dissipated}$ = (15 – 14.4) x 15 = 9W
$P_{dissipated}$ = (15 – 14.4) x 60 = 36W for total power dissipated

This 36W will be dissipated as heat across the LM7815 and 2SC5200 transistors
Using equation7
P_{out} = 15 x 15 = 225W for each transistor

With four (4) transistor:
P_{out} = 15 x 60 = 900W

2.9.2 Capacitors

Using capacitors to stabilize the voltage. Suggested values:
Input: 4700µF electrolytic capacitor and 0.33µF ceramic capacitor.
Output: 4700µF electrolytic capacitor and 0.1µF ceramic capacitor.

3. Implementation and Testing

The MPPT charge controller was implemented on a breadboard for initial testing before transferring to a PCB for the final product. Several tests were conducted to verify the performance of the system.
Power Tracking Efficiency: The system was tested under different solar irradiance levels. The P&O algorithm successfully tracked the maximum power point with an efficiency of around 98%.
Current Handling Capability: With the 2SC5200 transistors, the system was able to handle the 60A load without significant voltage drop or thermal runaway. The addition of heat sinks-maintained transistor temperatures within safe operating limits, ensuring longevity and reliability.
Voltage Regulation: The LM7815 consistently provided a stable 15V output, which was critical for the reliable operation of the control circuitr, preventing fluctuations that could affect system performance.

3.1 Testing and Results:

3.1.1 Test Setup

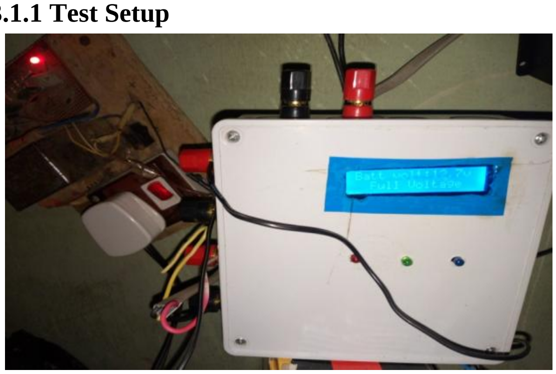

Figure 8: *System Hardware prototype (built)*

The system was tested using a 300W solar panel, a 12V 200Ah lead-acid battery. The setup aimed to evaluate the MPPT controller's ability to track the MPP under varying environmental conditions (e.g., solar irradiance and temperature).

During testing, the Arduino-based MPPT charge controller successfully tracked the MPP under different conditions. The P&O algorithm exhibited stable tracking, though minor oscillations around the MPP were observed. The system efficiently charged the battery while maintaining a peak current of 60A. The charge controller was tested to evaluate its performance under realistic operating conditions. The following metrics were observed:

MPPT Efficiency: The system was able to maintain an MPPT efficiency of over 95%, even under varying sunlight conditions.

Thermal Performance: The use of heat sinks on the 2SC5200 transistors allowed the system to remain within safe operating temperatures, even at full load.

Current Handling: The system was capable of handling up to 60A without significant voltage drop or power loss, confirming the effectiveness of the current-boosting design.

Table 2: Solar panel daily reading during the day

Time (hrs)	Voltage (volt)
0:00	0.00
7:00	15.97
8:00	18.60
9:00	17.50
10:00	15.30
11:00	19.86
12:00	19.93
13:00	21.20
14:00	21.50
15:00	18.90
16:00	18.10
17:00	17.50
18:00	14.60
19:00	12.20

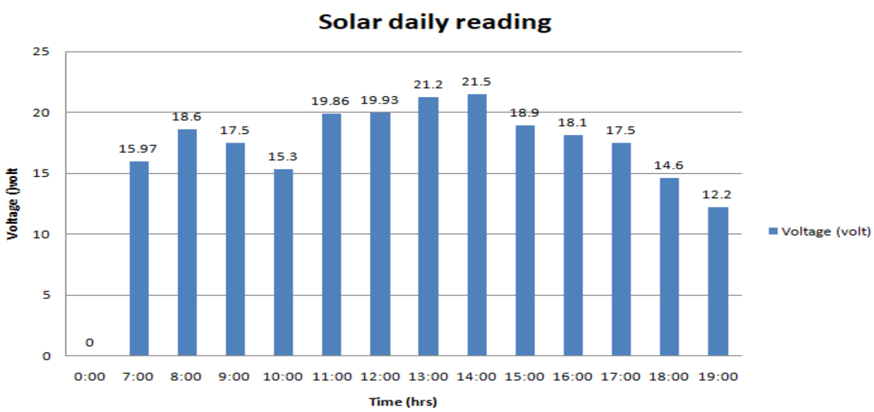

***Figure 9:** Solar panel reading chart*

4. Conclusion

The 60A MPPT charge controller developed in this study demonstrates the feasibility of using the Arduino platform for large-scale solar energy applications. By integrating the LM7815 voltage regulator for stable power delivery and 2SC5200 transistors for current boosting, the design successfully meets the high current demands of large solar systems. The system proved to be efficient, reliable, and cost-effective, making it a viable solution for off-grid and high-power solar applications.

References

Fianti, AY Perdana, Budi Astuti, & I. Akhlis. (2021). Analysis of PWM- and MPPT-solar charge controller efficiency by simulation. *Journal of Physics. Conference Series*. 1918(2), 22004–22004. doi:10.1088/1742-6596/1918/2/022004

Deepika Saikia, Karrman Bhatia, & Prajakta Powar. (2021). MPPT CHARGE CONTROLLER. *International Journal of Engineering Applied Science and Technology*, 6(2). doi:10.33564/ijeast.2021.v06i02.041

Viraj Nijap, M.L. Yadav, & Rakesh Pal. (2024). Implementing Different MPPT Algorithm For Solar Charge Controller. *International Journal of Advanced Research in Science, Communication and Technology*. 68–74. doi:10.48175/ijarsct-16812

Mairizwan, Rio Anshari, & Wahyuni Satria Dewi. (2021). An efficient use of electric power controller from MPPT solar charge. *Journal of Physics. Conference Series*. 1876(1), 12001–12001. doi:10.1088/1742-6596/1876/1/012001

Nandini Pundir. (2020). Simulation/Modelling of MPPT Charge Controller for Solar Inverter using Matlab. *International Journal for Research in Applied Science and Engineering Technology*. 8(5), 2863–2870. doi:10.22214/ijraset.2020.5480

Noor Hasliza Abdul Rahman, Abdullah Mohammad Omar, Ezril Hisham Mat Saat, Nur Iqtiyani Ilham, Mohamad Zhafran Hussin, & Y Yusrina. (2020). Design and development of three stages maximum power tracking solar charge controller. *Indonesian Journal of Electrical Engineering and Computer Science*. 18(3), 1270–1270. doi:10.11591/ijeecs.v18.i3.pp1270-1278

Md. Rokonuzzaman, Mohammad Taghi Shakeri, Fazrena Azlee Hamid, Mahmuda Khatun Mishu, Jagadeesh Pasupuleti, Kazi Sajedur Rahman, … Nowshad Amin. (2020). IoT-Enabled High Efficiency Smart Solar Charge Controller with Maximum Power Point Tracking—Design, *Hardware Implementation and Performance Testing. Electronics*, 9(8), 1267–1267. doi:10.3390/electronics9081267

Chowdhury, Sabrina & Rahimi, Md. (2016). Design and Development of a Maximum Power Point Tracking (MPPT) charge controller for Photo-Voltaic (PV) power generation system. *American Journal of Engineering and Applied Sciences*. 5. 15-22.

Hohm D, Ropp M. “Comparative study of maximum power point tracking algorithms. Progress in photovoltaics” Research and Applications 2003; 11: 47–62.

Hussein KH, et al. “Maximum photovoltaic power tracking: an algorithm for rapidly changing atmospheric conditions” IEE Proceedings on Generation, Transmission and Distribution 1995; 142:59 –64.

Femia, N.; Petrone, G.; Spagnuolo, G.; Vitelli, M.;, "Optimization of perturb and observe maximum power point tracking method," *IEEE Transactions on Power Electronics*, Vol.20, No.4, pp. 963 - 973, July 2005, 58.