

EFFICIENCY MAINTENANCE IN RENEWABLE ENERGY SYSTEM: MOTION-SENSOR LVDC CONTROL AND AUTOMATED SOLAR PANEL CONDITIONING

CDAC PROJECT

PROJECT REPORT

Submitted by

**ALAN GABRIEL
(B21EE1115, MBT21EE014)**



DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Mar Ivanios Vidyanagar, Nalanchira, Thiruvananthapuram, Kerala – 695 015, India

APRIL 2025

EFFICIENCY MAINTENANCE IN RENEWABLE ENERGY SYSTEM: MOTION-SENSOR LVDC CONTROL AND AUTOMATED SOLAR PANEL CONDITIONING

PROJECT REPORT

Submitted in partial fulfillment of the requirements of the award of Bachelor of Technology Degree in Electrical and Electronics Engineering of the APJ Abdul Kalam Technological University (KTU)

Submitted by

ALAN GABRIEL (B21EE1115, MBT21EE014)



DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

**MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY
(Autonomous)**

Mar Ivanios Vidyanagar, Nalanchira, Thiruvananthapuram, Kerala – 695 015, India

APRIL 2025

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

**MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY
(Autonomous)**

Mar Ivanios Vidyanagar, Nalanchira, Thiruvananthapuram, Kerala – 695 015, India.



CERTIFICATE

*This is to certify that this report entitled “Efficiency Maintenance in Renewable Energy: Motion-Sensor LVDC Control and Automated Solar Panel Conditioning” is a bonafide record of the project submitted by **ALAN GABRIEL (B21EE1115)** ,(VIII Semester) towards the partial fulfillment of the requirements for the award of B. Tech Degree in Electrical and Electronics Engineering of the APJ Abdul Kalam Technological University (KTU) during the year 2025.*

**Ms. Sandhya P
(Project Guide)
Assistant Professor
EEE Department**

**Ms. Surasmi N L
(Project Co Guide)
Assistant Professor
EEE Department**

**Mr. Nandagopan
(CDAC Guide)
PE Department**

**Ms. Aswin R B
(Project Coordinator)
Assistant Professor
EEE Department**

**Mr.Jayan P P
(CDAC Guide)
Scientist E
PE Department**

**Dr. Elizabeth Varghese
(Head of Department)
EEE Department**

ACKNOWLEDGEMENT

We express our heartfelt gratitude to God Almighty for His abundant grace throughout the completion of our Project Phase.

We are deeply thankful to our project guide, **Ms. Sandhya P.**, Assistant Professor in the Department of Electrical and Electronics Engineering, for her continuous support, insightful advice, and for providing the essential resources required for the successful completion of our Project Phases. We also extend our gratitude to **Ms. Surasmi N L** for her valuable assistance.

We wish to express our sincere appreciation to **Mr. Jayan P. P.**, Scientist at CDAC, Trivandrum, for his valuable suggestions, which greatly enhanced our presentation skills. His guidance and encouragement played a significant role in our growth during this phase of the project. Additionally, we thank **Mr. Nandagopan** from the Power Electronics Department at CDAC for his encouragement and support.

We extend our gratitude to **Mr. Aswin R. B.**, Project Coordinator and Assistant Professor of Electrical and Electronics Engineering department, for his guidance and unwavering support. Special thanks to **Dr. Soumya A. V.**, Associate Professor, our Class Advisor, for her encouragement throughout this endeavor.

Our sincere thanks go to **Dr. Elizabeth Varghese**, Head of the Department of Electrical and Electronics Engineering, for her leadership and support during our academic journey.

Lastly, we are immensely grateful to our parents and friends for their prayers, motivation, and moral support, which have been a source of strength throughout Project Phase.

ALAN GABRIEL

TABLE OF CONTENTS

ACKNOWLEDGEMENT	i
ABSTRACT	v
LIST OF FIGURES	vi
LIST OF TABLES	viii
NOMENCLATURE	ix
1. INTRODUCTION	1
1.1 INTRODUCTION	1
1.2 OBJECTIVE	1
1.3 SUMMARY	2
2. LITERATURE REVIEW	3
2.1 INTRODUCTION	3
2.2 SUMMARY	4
3. LVDC MOTION SENSOR	5
3.1 INTRODUCTION	5
3.2 COMPONENTS USED	6
3.3 DC SOURCE	6
3.4 VOLTAGE REGULATOR CIRCUIT	7
3.5 PIR SENSOR	9
3.6 DELAY CIRCUIT	9
3.6.1 Components Required	10
3.6.2 Circuit diagram	10
3.6.3 Working	11
3.6.4 Delay Calculation	11
3.7 NPN TRANSISTOR	12
3.7.1 Need for NPN Transistor	12
3.7.2 Selection and Testing of Transistors for Switching and Stabilization	12
3.8 SUMMARY	13
4. TESTING AND MODIFICATION OF LVDC MOTION SENSOR	14
4.1 INTRODUCTION	14
4.2 SELECTION OF BETTER TRANSISTORS	14
4.3 MODIFICATION OF TIMER CIRCUIT	15
4.4 PROTOTYPE IMAGE	15
4.5 PROTOTYPE COST ESTIMATION	16
4.6 FINAL SCHEMATIC DIAGRAM	17
4.7 GERBER FILE	17
4.8 SUMMARY	18
5. LVDC SENSOR CARD	19
5.1 INTRODUCTION	19
5.2 PCB MODULE	19
5.3 SUMMARY	20
6. AUTOMATED SOLAR PANEL CONDITIONING SYSTEM	21

6.1 INTRODUCTION	21
6.2 IMPORTANCE OF EFFICIENCY MAINTENANCE	21
6.3 OBJECTIVE	22
6.4 KEY BENEFITS	22
6.5 SUMMARY	23
7.CASE STUDY ON SOLAR PARK IN INDIA	24
7.1 INTRODUCTION	24
7.2 BHADLA SOLAR PARK	25
7.3 NOOR ABU DHABI SOLAR PLANT	26
7.4 MIT RESEARCH	27
7.5 K-DISC BUILDING	28
7.6 SUMMARY	28
8.SOLAR PANEL SURFACE CONDITIONING	30
8.1 INTRODUCTION	30
8.2 PRACTICAL SURVEY	30
8.2.1 Given Data	31
8.2.2 Calculate Standard Efficiency at 25°C	31
8.2.3 Efficiency at Nominal Operating Temperature (42°C)	32
8.2.4 Efficiency When Panel is Cool (e.g., 15°C)	32
8.2.5 Efficiency with Dust and Debris	32
8.3 GRAPHICAL REPRESENTATION	33
8.4 SUMMARY	34
9.SURFACE CONDITIONING SYSTEM ARCHITECTURE	35
9.1 INTRODUCTION	35
9.2 ESP32	35
9.3 HUB	36
9.3.1 Given Data	36
9.3.2 Electronic Components of the Hub	37
9.3.3 Programming & Software Implementation	37
9.3.5 Hardware	39
9.3.6. Schematic Diagram	40
9.3.7 ESP WIFI flowchart	41
9.3.8 ESP NOW flowchart	41
9.4 PANEL BOTS	41
9.4.1 Panel Bot Functionality	42
9.4.2 Electronic Components of the Panel Bot	42
9.4.3 Communication And Command Execution	43
9.4.4 Communication Flow of Panel Bots	44
9.4.5 Hardware	44
9.4.6 Schematic	48
9.4.7 Panel Bot FlowChart	48
9.5 WATER PUMP CIRCUIT	49
9.5.1 Electronic Components	49
9.5.2 Working	49

9.5.2 Hardware	50
9.5.3 Pump Relay Flow Chart	51
9.5.4 Schematic	52
9.6 MAX6675 TEMPERATURE SENSOR	52
9.7 H-BRIDGE MOTOR DRIVER	54
9.8 BLYNK IOT PLATFORM	56
9.9 ESP32, ESP8266, AND ESP-NOW	57
9.9.1 ESP32 and ESP8266 Specification	58
9.9.2 ESP NOW Radio Communication	58
9.10 LI-ION BATTERY 18650	59
9.11 FINAL PROTOTYPE	61
9.12 SUMMARY	61
10.COMPONENTS/ HARDWARE	62
10.1 TABLE OF COMPONENTS WITH PRODUCT LINK	62
11.CONCLUSION	64
REFERENCES	65
ANNEXURE	66
1) PROGRAM CODE FOR PANEL BOTS	66
2) PROGRAM CODE FOR HUB	71
3) PROGRAM CODE FOR WATER PUMP	84

ABSTRACT

This project introduces an intelligent energy optimization system leveraging Low-Voltage Direct Current (LVDC) technology integrated with motion-sensing automation. A 48V LVDC motion sensor system is developed to detect human presence and intelligently control BLDC fans and LVDC lights, enabling power usage only when necessary. This reduces unnecessary energy draw and promotes efficient electricity usage in modern infrastructures.

In addition to indoor energy savings, the project incorporates a Solar Panel Surface Conditioning System that enhances solar energy efficiency through automated panel cleaning. The system employs a mechanized cleaning unit and water pumping module that utilizes stored rainwater or filtered greywater. It is equipped with IoT capabilities for real-time monitoring and decision-making, and communicates through radio transmission between networked microcontrollers. An ESP-based microcontroller processes sensor data and commands the cleaning unit to operate based on environmental conditions, ensuring consistent cleanliness and optimal energy generation.

LIST OF FIGURES

Fig. 1.1 LVDC Scheme.....	1
Fig. 3.1 Block Diagram.....	5
Fig 3.2 DC source	7
Fig 3.3 Linear Voltage Regulator	8
Fig 3.4 Delay circuit	10
Fig 4.1 LVDC motion sensor prototype	15
Fig 4.2 LVDC motion sensor schematic	17
Fig 4.3 PCB Layout gerber file	17
Fig 5.2 LVDC Sensor Card.....	19
Fig 7.1 Bhadla Solar Park.....	25
Fig 7.2 Noor Abu Dhabi Solar Plant.....	26
Fig 7.3 MIT Research Graph.....	27
Fig 8.1: Before and After Cleaning Graph.....	33
Fig 9.1 Hub Hardware.....	40
Fig 9.2 Circuit for Hub.....	40
Fig 9.3: ESP WIFI.....	41
Fig 9.4: ESP NOW.....	41
Fig 9.5: CAD Drawing of Panel Bots.....	46
Fig 9.6.: CAD Drawing of Motor and Gear assembly unit.....	47
Fig 9.7: Panel bot Schematic.....	48
Fig 9.8: Panel bot Flowchart.....	48
Fig 9.9: Water Pump Hardware.....	51
Fig 9.10: Pump relay Flowchart.....	51

Fig 9.11: Water Pump Schematic.....	52
Fig 9.12:MAX6675 Temperature Sensor.....	54
Fig 9.13: H-Bridge Motor Driver.....	55
Fig 9.14: Blynk IOT Interface.....	57
Fig 9.15: ESP Now Radio Communication.....	59
Fig 9.16: LI-ION Battery 18650.....	60
Fig 9.17: Panel Bot Prototype.....	61

LIST OF TABLES

Table 4.1 Cost Estimation.....	16
Table 8.1: A sample table and its numbering.....	30
Table 9.1: MAX6675 Feature Table.....	53
Table 9.2: Key specifications of L298N Motor Driver.....	55
Table 9.3: Specification of ESP32 and ESP8266.....	58
Table 10.1: Components with Product Link.....	63

NOMENCLATURE

LVDC	Low Voltage Direct Current
AC	Alternating Current
DC	Direct Current
CADC	Compact and Adaptive Direct Current
KDISC	Kerala Development and Innovation Strategic Council
CDAC	Centre for Development of Advanced Computing
LiFePO ₄	Lithium Iron Phosphate (Battery Type)
PIR	Passive Infrared (Sensor)
NE555	555 Timer Integrated Circuit
IC	Integrated Circuit
PV	Photovoltaic
BLDC	Brushless Direct Current (Motor or Fan)
P _{max}	Maximum power output under STC
I _{mp}	Current at maximum power
V _{mp}	Voltage at maximum power
η	Efficiency
A	Area of solar panel
T	Panel temperature
G	Solar irradiance
I _{sc}	Short-circuit current
V _{oc}	Open-circuit voltage

1. INTRODUCTION

1.1 INTRODUCTION

- We plug the devices that internally operate on a DC.
- AC is converted to DC to operate the equipment. The process of conversion is not 100% efficient, due to power loss.
- By combining LVDC distribution with motion-based automation, the use of electricity can be optimized.

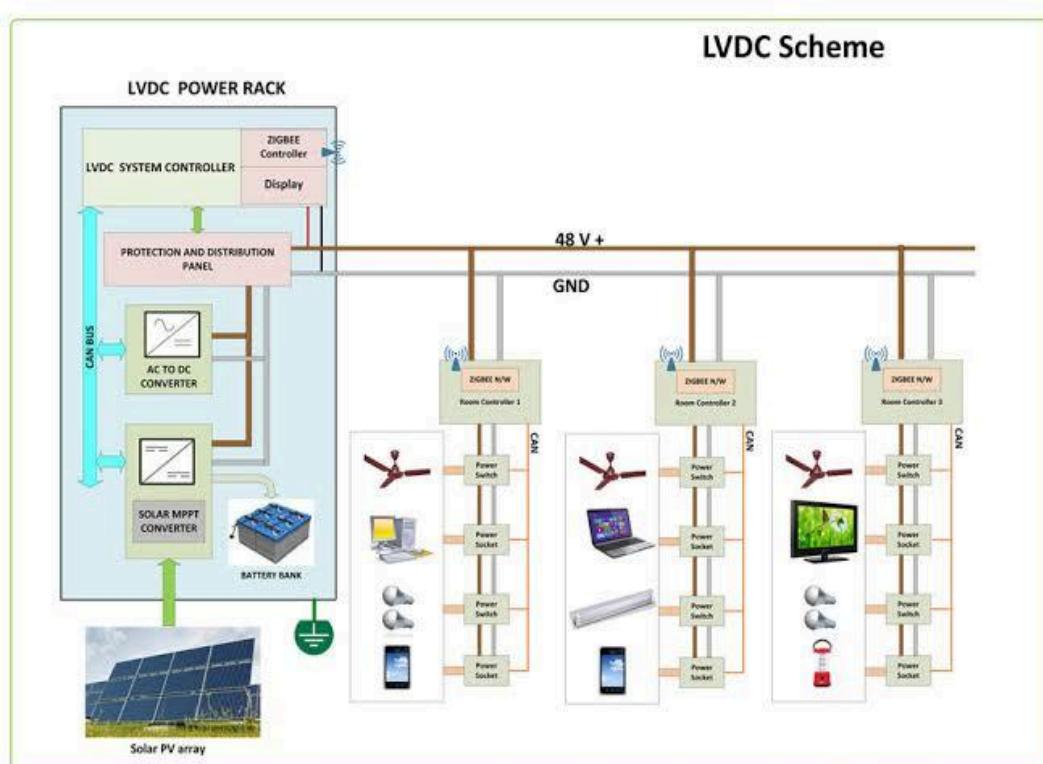


Fig 1.1: LVDC Scheme

1.2 OBJECTIVE

To implement an efficient motion sensor circuit for the LVDC Scheme shown in Fig 1.1 which regulates electricity consumption.

1.3 SUMMARY

In many buildings, most devices operate internally on DC, including PCs, LED lights, TVs, and mobile phones. However, renewable sources like solar PV produce DC power that is converted to AC, resulting in power losses. LVDC (Low Voltage Direct Current) home architecture can reduce these losses by minimizing conversion stages, improving energy efficiency, and allowing direct integration with renewable sources. LVDC homes are compatible with efficient devices like BLDC fans and LED lights, offer better power quality, and ensure safety. This project aims to develop an LVDC motion sensor system for BLDC fans and lights to reduce energy consumption.

2. LITERATURE REVIEW

2.1 INTRODUCTION

The growing demand for energy efficiency and renewable energy integration is driving interest in Low-Voltage Direct Current (LVDC) distribution systems. Traditional power systems rely on AC distribution, which requires multiple conversion stages when interfacing with DC-powered devices and renewable sources like solar panels, resulting in energy losses. LVDC distribution offers a solution by enabling direct power supply to DC-based equipment and integration with renewable energy sources. This approach not only minimizes conversion losses but also enhances system reliability, cost-effectiveness, and power quality—making LVDC a promising alternative for residential and commercial power distribution.

Low-Voltage DC Distribution—Utilization Potential in a Large Distribution Network Company Tomi Hakala, Tommi Lähdeaho, and Pertti Järventausta, Member, IEEE

The potential benefits of Low-Voltage Direct Current (LVDC) distribution in improving the power transfer capacity, cost efficiency, and reliability of electricity distribution systems.

T.-F. Wu, Y.-K. Chen, G.-R. Yu, and Y.-C. Chang, “Design and development of DC-distributed system with grid connection for residential applications,” in Proc. 8th Int. Conf. Power Electronics, ECCE Asia, 2011, pp. 235–241.

Aims to optimize the distribution of DC power in homes, reduce conversion losses that typically occur when AC power is converted to DC, and enable better integration of renewable energy sources (like solar panels) which naturally generate DC power.

A. Shrestha, D. Bista and B. Adhikary, "Current Practices of Solar Photovoltaic Panel Cleaning System and Future Prospects of Machine Learning Implementation," July 2020

Highlight the main factors affecting SPV system efficiency, including PV cell technology, ambient temperature, humidity, and soiling. Emphasize the role of cleaning in mitigating soiling issues to maintain SPV efficiency and longevity.

Shrihari Prasath.B and Dr. Vimalathithan Rathinasabapathy, "A Smart IoT System For Monitoring Solar PV Power Conditioning Unit" August 2016.

The remote monitoring system for the solar PV Power Conditioning Unit (PCU) in a greenhouse environment aims to address management and maintenance issues while reducing the time to repair.

2.2 SUMMARY

This literature review highlights the advantages of Low-Voltage Direct Current (LVDC) distribution for enhancing efficient energy consumption, cost efficiency, and reliability in large distribution networks, on optimizing DC power distribution within residential systems to reduce conversion losses and improve integration of renewable sources like solar panels, which generate DC power directly. Together, these studies emphasize LVDC's potential to increase efficiency and facilitate renewable energy usage in power systems.

3. LVDC MOTION SENSOR

3.1 INTRODUCTION

A motion sensor is an apparatus that senses movement in a space and initiates actions such as turning on or off lights or appliances. Motion sensors are frequently used in doorways and homes to automate security and lighting systems, making sure that lights only turn on when necessary. Motion sensors help cut down on wasteful energy use by shutting off lights and appliances when a room is empty, which lowers electricity costs and encourages energy efficiency.

Our motion sensor, which is integrated with the LVDC architecture used in buildings, runs directly on a 48V DC source. By detecting human intervention, this configuration enables effective control of LED lights and BLDC fans. This motion sensor improves energy efficiency by utilising the LVDC supply, which lowers monthly electricity consumption. Block diagram of the system is given in Fig 3.1

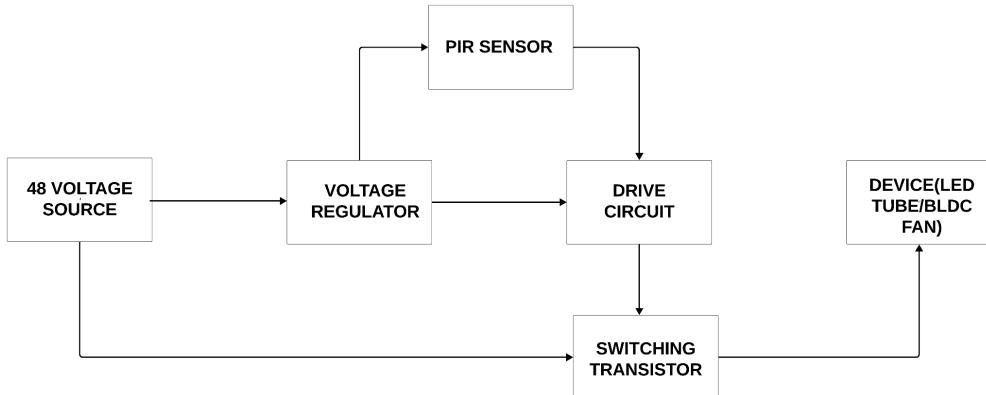


Fig 3.1, Block Diagram LVDC motion sensor

3.2 COMPONENTS USED

- DC Source
- Voltage regulator circuit
- PIR sensor
- Delay circuit
- 60V NPN-transistor

3.3 DC SOURCE

As illustrated in Fig. 3.2, CDAC and KDISC use an off-grid LVDC system that is powered by a 48V lithium-iron (LiFePO4) battery pack made by Hykon. With this configuration, LVDC lighting and fans can be powered by an efficient electrical distribution system throughout the building. A steady, low-voltage direct current supply is provided by the 48V battery system, minimising energy losses associated with AC-DC conversion. The system increases energy savings by delivering 48V DC directly to devices intended for LVDC operation.

This strategy was combined with renewable energy sources, like solar panels, which produce DC electricity. By using 48V DC directly for lighting and fan systems, the facility can use less electricity and develop a more sustainable power solution.



Fig 3.2: DC Source

3.4 VOLTAGE REGULATOR CIRCUIT

A voltage regulator circuit is required to step down the 48V DC supply to a stable lower voltage, such as 5V DC, to safely operate components like the NE555 timer IC and the PIR motion sensor module. Without regulation, the high 48V input could harm these components, which normally need lower operating voltages.

The regulator circuit reduces the voltage, guaranteeing a steady 5V supply that enables the NE555 to operate as an oscillator or timer in the motion sensor system and the PIR module to precisely detect motion. By allowing the use of low-voltage

components in a higher-voltage system, this regulated 5V output seamlessly integrates with the LVDC architecture and is essential for safe and effective operation.

The voltage was stepped down to the appropriate level using a voltage regulator circuit in the original design of the 48V DC motion sensor circuit. The output current of this circuit was insufficient to adequately power the load, which was a serious drawback. This restriction was mostly caused by a 5k resistor, which limited the current flow and decreased the circuit's efficiency. We switched to a different linear voltage regulator circuit in order to solve this problem. This alternate solution produced a simpler and more economical design by using fewer components while still providing an adequate output current. Because it was more cost-effective and efficient, the new regulator circuit in Fig. 3.3 was a better fit for the project's needs.

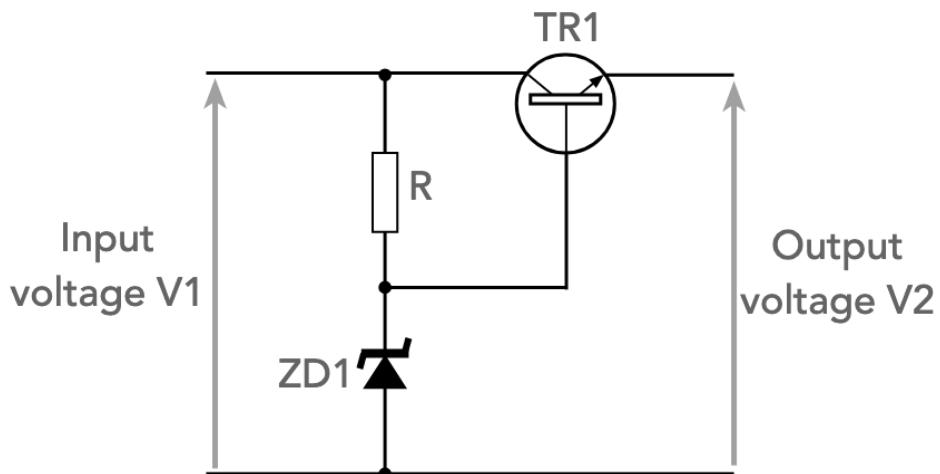


Fig 3.3 Linear Voltage Regulator

The circuit uses a single Zener diode or another type of voltage regulator diode, which is powered through a resistor connected to the unregulated supply, along with a pass transistor arranged in an emitter follower setup. This offers a straightforward feedback mechanism to guarantee that the Zener voltage is kept at the output, albeit with a voltage drop equivalent to the silicon transistor's base-emitter junction voltage of 0.6 volts. Such a circuit for a series pass voltage regulator is easy

to design. It is possible to determine the maximum emitter current by knowing the maximum current that the load requires. This is accomplished by dividing the transistor emitter current, or load current, by the transistor's β or h_{FE} . For a small Zener to maintain its regulated voltage, the Zener diode typically requires at least 10mA. Once the unregulated voltage, Zener voltage, and required current are known, the resistor value should be determined to supply both the base current for the transistor and the minimum current needed by the Zener diode. [Zener voltage minus unregulated voltage] / current. It is important to keep in mind that a tiny margin should be added to the current to guarantee that there is enough space for margin when the load is applied, causing the transistor base to absorb the entire current. When the load current is zero, the Zener diode's power dissipation capacity should be computed. In this scenario, the series resistor's entire current must be passed through the Zener diode. To help eliminate noise and potential voltage transients, a capacitor may occasionally be positioned across the Zener diode or voltage reference diode.

3.5 PIR SENSOR

A passive infrared (PIR) sensor is an electronic device that detects infrared radiation emitted by objects in its field of view. PIR sensors are commonly used in motion detectors, security alarms, and automatic lighting.

The 287-18001 module operates using infrared technology and features automatic control with a design that incorporates the high-sensitivity, high-reliability LHI778 sensor imported from Germany. It functions in an ultra-low-voltage mode and is commonly used in a range of motion-sensing electronic devices, particularly in battery-operated automatic control systems.

3.6 DELAY CIRCUIT

In this 48V DC motion sensor circuit, when motion is detected, the sensor outputs a 5V signal to an NPN transistor, activating the connected device. When motion ceases, the sensor's output drops to 0V, which would normally turn off the device immediately. However, to ensure the device remains on for a few additional

seconds after motion stops, a delay circuit is introduced. This delay circuit provides a buffer period, keeping the device in an active state temporarily even after the sensor signal is cut off. This feature is especially useful in applications requiring a consistent operation duration beyond the immediate motion detection, such as in lighting or alarm systems. The delay is achieved using a capacitor-resistor network or a timer IC, which sustains the transistor's base voltage momentarily, allowing for a smooth and reliable delay before deactivation. The circuit diagram of the Delay Circuit is shown in Fig 3.4.

3.6.1 Components Required

- 555 Timer IC
- LED / output device
- 470uF Capacitor
- Resistors: 68K, 10K, 220R (according to delay)

3.6.2 Circuit diagram

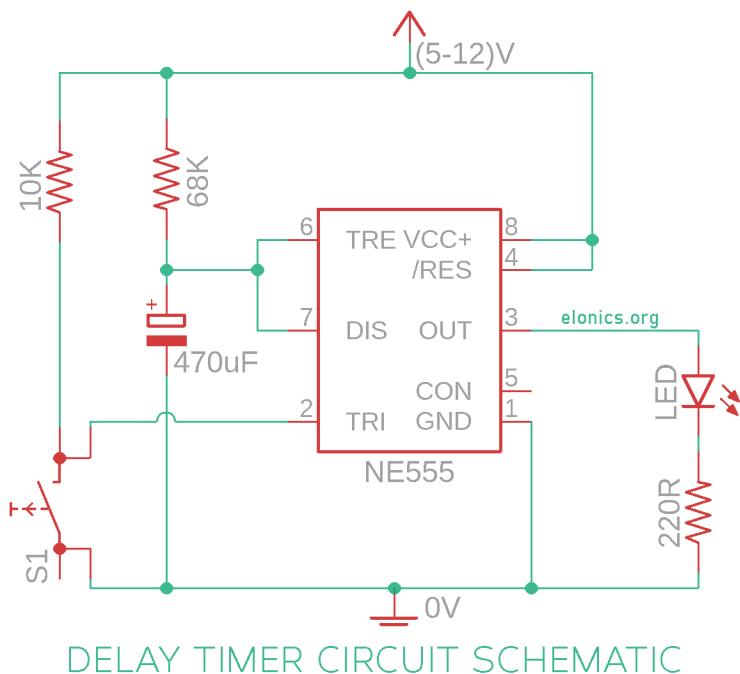


Fig 3.4 Delay Circuit

3.6.3 Working

- 0V is applied at the trigger pin (Pin-2)
- Since this applied voltage (0V) at Pin-2 is less than 1/3rd of the supply voltage, the output turns ON
- Simultaneously, the discharge Pin disconnects internally from 0V
- So now the capacitor starts charging via the resistor/potentiometer that connects it to positive rail
- Since the threshold input pin (Pin-6) is connected to positive terminal of the capacitor, it actively monitors the voltage across it
- As soon as the capacitor charges to 2/3rds of the supply voltage, Pin-6 turns OFF the output
- (This time period for which the capacitor charges from 0V to 2/3rds of supply voltage is the delay time)
- As soon as the output turns OFF, Pin-7 is internally re-connected to 0V and the capacitor is discharged completely
- The above steps are repeated each time when pin 2 get 0V

3.6.4 Delay Calculation

$$T=1.1RC \quad 30=1.1RC$$

let R=100000(to reduce current)

$$30=1.1*100000$$

$$C=30/(1.1*10000)$$

$$=2.7*10^{-4} \text{ (Selected } 470\mu\text{F})$$

3.7 NPN TRANSISTOR

3.7 NPN TRANSISTOR

3.7.1 Need for NPN Transistor

In this 48V DC motion sensor circuit, the output signal from the PIR (Passive Infrared) sensor is connected to the base of an NPN transistor, which serves as a switch for the 48V supply. When motion is detected, the PIR sensor provides a voltage signal to the transistor's base, allowing current to flow through the collector-emitter junction. This enables the transistor to complete the circuit, thereby powering the connected device.

Additionally, the linear voltage supply circuit utilizes another NPN transistor to maintain stable operation. By regulating the flow of current, this transistor helps to convert the 48V supply to the required lower voltage levels necessary for other components within the circuit, ensuring efficient operation without fluctuations.

3.7.2 Selection and Testing of Transistors for Switching and Stabilization

Initially, transistors such as the TIP3055 were chosen for switching and stabilizing the 48V DC supply in the circuit. However, during testing, numerous transistors (including CL100, MJE3055, TIP3055, and TIP35C) were damaged due to overheating. This issue was traced to the low DC gain (hFE) of these transistors, which required higher base currents, leading to excessive power dissipation and thermal stress.

To address this, transistors with higher DC gain values were selected. High-gain transistors, such as the BDX33D and BD677, with a DC gain of approximately 750 hFE , were used as replacements. The increased gain significantly reduced the base current requirements, thereby lowering thermal stress and enhancing circuit reliability. After implementing these higher-gain transistors, the circuit operated as expected, with improved stability and no further instances of overheating or component failure.

3.8 SUMMARY

The developed motion sensor features a PIR sensor, a NE555 timer, a voltage regulator, and a 48V switching transistor, designed to operate within a low-voltage DC system. This configuration enhances energy efficiency by automating device activation based on occupancy, further reducing power consumption in homes and optimizing the overall performance of connected devices.

4. TESTING AND MODIFICATION OF LVDC MOTION SENSOR

4.1 INTRODUCTION

This section details the process of testing and modifying the LVDC (Low Voltage DC) motion sensor circuit to ensure optimal functionality. Extensive testing was conducted on various components to identify those best suited for stable and efficient operation. Through this process, optimal working components were selected to handle the required voltage and current specifications, ensuring both reliability and durability. Additionally, a more efficient delay circuit was developed, allowing for smooth and controlled timing adjustments. The resulting design provides consistent activation and deactivation based on motion detection, making the circuit highly suitable for a wide range of low-voltage applications.

4.2 SELECTION OF BETTER TRANSISTORS

In optimizing the LVDC motion sensor circuit, the selection of transistors with high DC gain (hFE) was crucial to achieving stable and efficient operation. Initially, transistors with low DC gain were chosen for switching and stabilization. However, these components required higher base currents to achieve the necessary output, resulting in significant power dissipation and overheating. This led to repeated transistor failures, compromising the circuit's reliability.

To address this issue, transistors with higher DC gains, such as BDX33D and BD677 (with gains around 750 hFE), were selected. These high-gain transistors minimized the required base current, reducing thermal stress on the components and virtually eliminating overheating issues. This change not only enhanced the efficiency and reliability of the circuit but also extended the lifespan of the components, resulting in optimal circuit performance without heat-related failures.

4.3 MODIFICATION OF TIMER CIRCUIT

In the initial delay timer circuit design, pins 6 and 7 of the NE555 timer IC were connected, causing the timer to remain unresponsive to new inputs until the existing timing cycle completed. This configuration prevented the circuit from immediately responding to a new trigger signal, as the timer would only restart after fully turning off. To address this issue, pin 7 was disconnected, leaving it idle, and pins 2 and 6 were shorted together. This modification ensures that whenever a ground signal is applied to pin 2, the capacitor discharges, effectively resetting the timer and starting a new timing cycle. This change allows the circuit to respond promptly to each new input signal, making it more reliable for applications that require immediate re-triggering. The LVDC Motion Sensor Prototype Image is shown in Fig 4.1.

4.4 PROTOTYPE IMAGE

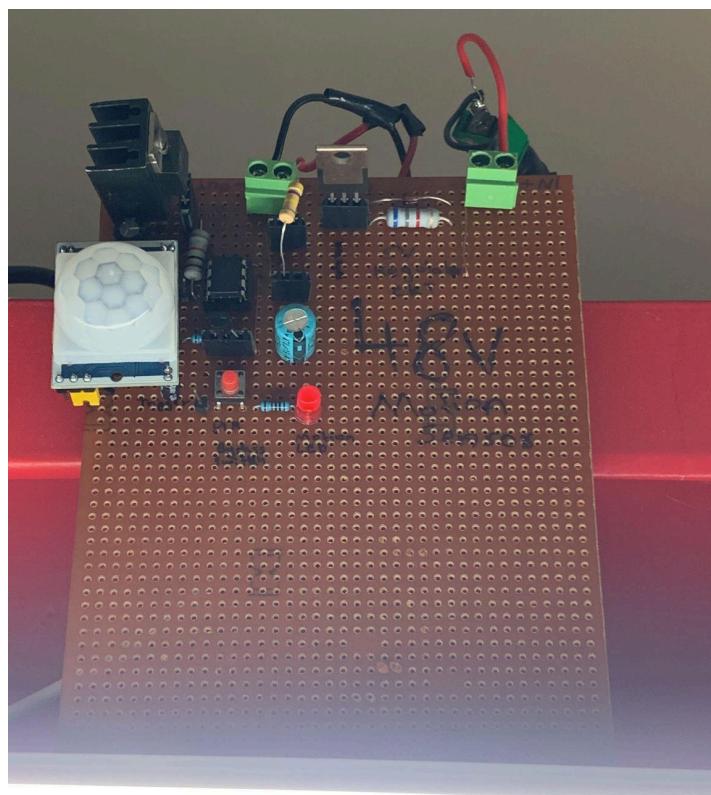


Fig 4.1 LVDC motion sensor prototype

4.5 PROTOTYPE COST ESTIMATION

SI NO.	COMPONENTS	NO. OF UNITS	TOTAL PRICE (Rs.)
1	PIR Motion Sensor	1	125
2	Heat Sink	1	10
3	555 Timer IC	1	15
4	Capacitors	2	4
5	Resistors	4	8
6	5V Zener Diode	1	5
7	Transistors	3	45
8	PCB Board	1	30
9	Voltage Regulator	2	60
	<i>Miscellaneous Expenses</i>		500
	<i>Contingency Expenses</i>		998
	TOTAL		1,800

Table 4.1 Cost Estimation

4.6 FINAL SCHEMATIC DIAGRAM

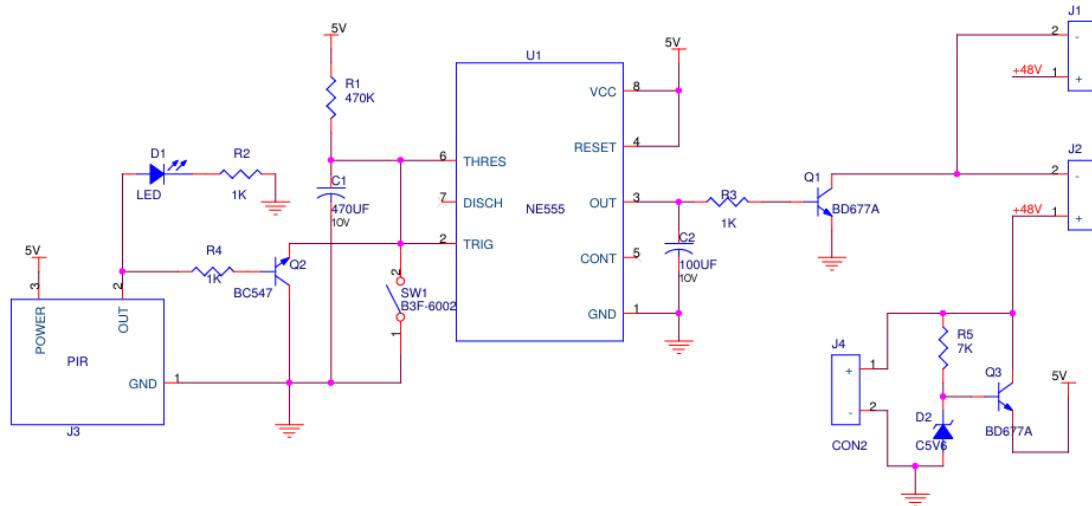
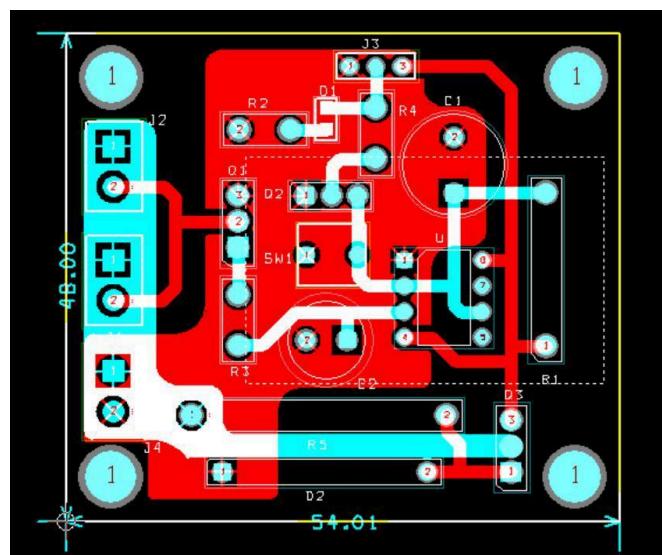


Fig 4.2 LVDC motion sensor schematic

4.7 GERBER FILE



4.3 PCB Layout Gerber file

4.8 SUMMARY

The testing and modification process for the LVDC motion sensor circuit focused on enhancing stability, efficiency, and reliability. Initial component testing revealed that low DC gain transistors led to overheating and frequent failures due to high base current requirements. To address this, high-gain transistors such as BDX33D and BD677 were selected, reducing base current needs and minimizing thermal stress, which significantly improved circuit durability and performance. Additionally, modifications to the NE555 timer circuit were made to enhance responsiveness. By disconnecting pin 7 and shorting pins 2 and 6, the timer resets immediately with each new trigger, allowing for consistent re-triggering and more reliable operation. These changes, along with the final schematic diagram shown in Fig 4.2 and Layout of Gerber file in Fig 4.3, provide a robust and efficient LVDC motion sensor design ideal for low-voltage applications.

5. LVDC SENSOR CARD

5.1 INTRODUCTION

The LVDC sensor card is an electronic circuit board labeled "BGF PE 02 1202A LVDC SENSOR CARD." The final PCB was manufactured by C-DAC based on the schematic diagram that we provided. This type of circuit board is typically used in low-voltage direct current (LVDC) applications, likely as a sensing module for monitoring electrical parameters such as voltage, current, or environmental factors. The presence of various electronic components suggests it plays a role in signal processing and conditioning.

5.2 PCB MODULE

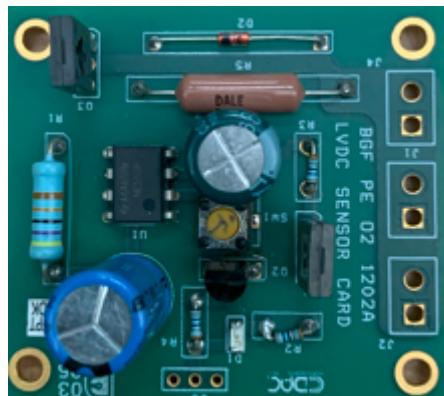


Fig 5.2 LVDC SENSOR CARD

5.3 SUMMARY

After the long term process of testing, evaluation and multiple iterations of our schematic design, C-DAC finalized and manufactured the LVDC Sensor Card with improved accuracy and reliability. The final PCB ensures precise monitoring and working of electrical equipment in a low-voltage DC environment. The LVDC sensor cards were mass-produced for large-scale deployment in the K-DISC building as part of the energy-efficient infrastructure initiative.

6. AUTOMATED SOLAR PANEL CONDITIONING SYSTEM

6.1 INTRODUCTION

The implementation of an automated solar panel surface conditioning system is crucial to maintaining optimal solar energy efficiency. Accumulation of dust and debris on the panel surface significantly reduces power generation. Studies indicate that every 1% increase in panel temperature reduces efficiency by approximately 0.5%. This efficiency drop occurs due to increased resistance and reduced voltage output.

Solar panels exposed to high temperatures experience energy losses, impacting overall system performance. Thus, developing an IoT-based automated cleaning mechanism is essential for ensuring steady and uninterrupted power supply while optimizing energy output.

6.2 IMPORTANCE OF EFFICIENCY MAINTENANCE

Solar energy is an abundant and renewable resource, but its effectiveness depends on maximizing energy conversion efficiency. The two main factors that lead to efficiency losses are as follows:

1. **Temperature Rise:** Every 1°C increase above standard test conditions (25°C) results in a 0.5% decrease in efficiency.
2. **Dust Accumulation:** Research indicates that unclean panels can lose up to 25% of their efficiency over time if not maintained properly.

To counteract these losses, regular cleaning and temperature regulation are necessary.

6.3 OBJECTIVE

The primary objective of this project is to develop an IoT-based automated solar panel surface conditioning system that enhances solar energy efficiency by addressing the efficiency losses due to dust accumulation and temperature rise. The system integrated an automated cleaning mechanism to remove dirt and debris, ensuring that the panels operated at optimal efficiency. In addition, the system incorporates an LVDC distribution network that minimizes energy conversion losses and improves power management. Another key goal is the integration of the Blynk IoT, enabling real-time monitoring and control of the cleaning system through a mobile application, allowing users to track system performance and manually initiate cleaning cycles when required. By ensuring continuous efficiency maintenance, this system contributes to the longevity and sustainability of solar panel installation. The project ultimately aims to enhance the energy output, reduce maintenance costs, and provide a scalable solution for large-scale solar farms and distributed solar power systems.

6.4 KEY BENEFITS

The implementation of an IoT-based solar panel cleaning system offers significant benefits in terms of energy efficiency, cost savings, and sustainability. By integrating an automated cleaning mechanism, the system ensures that dust and debris do not accumulate on the panel surface, thereby maintaining optimal power output. This results in enhanced energy management, where unnecessary power loss is minimized. Additionally, using an LVDC distribution network reduces conversion losses, making the overall system more efficient. The ability to remotely monitor and control cleaning cycles through Blynk IoT further optimizes energy utilization and reduces maintenance efforts, ensuring that panels operate at their highest possible efficiency.

Temperature regulation is another crucial advantage of the system. Excessive heat can cause a substantial drop in panel efficiency, and by implementing a cooling and automated cleaning system, overheating is prevented. This leads to improved

long-term performance and reduced wear on the panels. Moreover, since the automated cleaning mechanism reduces the reliance on manual labour, the system provides a cost-effective and sustainable solution for maintaining solar farms. Over time, the consistent performance of the panels ensures lower energy costs, increased return on investment, and improved adoption of renewable energy solutions for various applications.

6.5 SUMMARY

Ensuring the efficiency of solar panels is crucial for optimizing energy production and maintaining the system's reliability over time. Dust and debris buildup can greatly diminish power output, and rising temperatures further lower efficiency. Research indicates that for every 1°C increase above 25°C, efficiency drops by 0.5%. To tackle these issues, an IoT-driven automated system was created to improve solar panel efficiency by incorporating an automated cleaning feature and temperature control. This system keeps panels dust-free, preventing up to a 25% loss in efficiency over time, and includes an LVDC distribution network to reduce energy conversion losses. Utilizing Blynk IoT allows for real-time monitoring and remote management of cleaning cycles, ensuring peak performance with minimal manual effort. Moreover, temperature control prevents overheating, enhancing the panels' long-term performance and lifespan. This automated approach not only boosts energy output but also cuts down on maintenance expenses and labor reliance, offering a cost-effective and scalable solution for both large solar farms and distributed solar power systems. By maintaining consistent efficiency, the system increases return on investment and encourages the widespread adoption of sustainable solar energy solutions across various applications.

7.CASE STUDY ON SOLAR PARK IN INDIA

7.1 INTRODUCTION

Solar energy has become a crucial component of sustainable power generation, but maintaining the efficiency of solar panels remains a challenge due to environmental factors such as dust accumulation, temperature fluctuations, and humidity. Case studies from different regions highlight the impact of these challenges and the innovative solutions being implemented to mitigate efficiency losses. Bhadla Solar Park in Rajasthan, one of the largest solar farms in the world, faces severe dust accumulation due to its desert location, leading to frequent energy losses. To address this, automated robotic cleaning systems have been explored to ensure continuous dust removal and reduce operational costs. Similarly, the Noor Abu Dhabi Solar Plant in the UAE, operating in an arid environment with extreme water scarcity, utilizes robotic dry-cleaning technologies to maintain efficiency without relying on water-based cleaning methods. Additionally, MIT research has developed hydrophobic coatings that repel dust, reducing the need for frequent maintenance and extending panel lifespan. In Kerala's K-DISC Building, where humidity and airborne dust affect solar panel performance, an LVDC-powered automated cleaning system with IoT-based real-time monitoring has been introduced. Following a site inspection, a panel bot prototype was deployed to enhance long-term reliability and efficiency. These advancements emphasize the importance of smart, sustainable solutions in solar panel maintenance.

7.2 BHADLA SOLAR PARK

Bhadla Solar Park in Rajasthan Fig 7.1 is one of the largest solar farms in the world, situated in an extreme desert climate. The region experiences high levels of dust accumulation, significantly reducing the efficiency of solar panels. Due to the vast area covered by the park, manual cleaning methods are highly labour intensive and inefficient. This results in frequent energy losses, affecting the overall power output. To address these challenges, the implementation of automated robotic cleaning solutions has been explored, allowing for scheduled and efficient maintenance. These cleaning mechanisms ensure continuous dust removal, improving energy output and reducing operational costs while promoting sustainability in large-scale solar farms.



Fig 7.1 Bhadla Solar Park

7.3 NOOR ABU DHABI SOLAR PLANT

The Noor Abu Dhabi Solar Plant Fig 7.2, located in the UAE, is one of the world's largest single-site solar power plants. Given the arid environment, panels in this facility face severe dust accumulation, leading to substantial energy losses. Traditional water-based cleaning is impractical due to the region's water scarcity. Instead, the plant utilizes robotic dry-cleaning technologies that effectively remove dust without using water. These autonomous cleaning solutions operate in cycles, reducing downtime and maintaining a high level of efficiency. The integration of such innovative solutions highlights the importance of water-efficient cleaning mechanisms in sustaining the performance of large-scale solar farms in desert environments.



Fig 7.2 Noor Abu Dhabi Solar Plant

7.4 MIT RESEARCH

MIT research on solar panel maintenance has explored the effects of dust accumulation on energy efficiency. Studies conducted at MIT found that efficiency losses could reach up to 30% in extreme conditions if solar panels were not regularly cleaned. To mitigate these losses, researchers developed hydrophobic coatings that repel dust and minimize accumulation over time. These coatings reduce the need for frequent cleaning and extend the operational lifespan of solar panels. By integrating smart surface technologies with IoT-based monitoring, the research emphasizes new advancements in self-cleaning solar panel technologies, which could revolutionize the way solar energy systems are maintained globally.

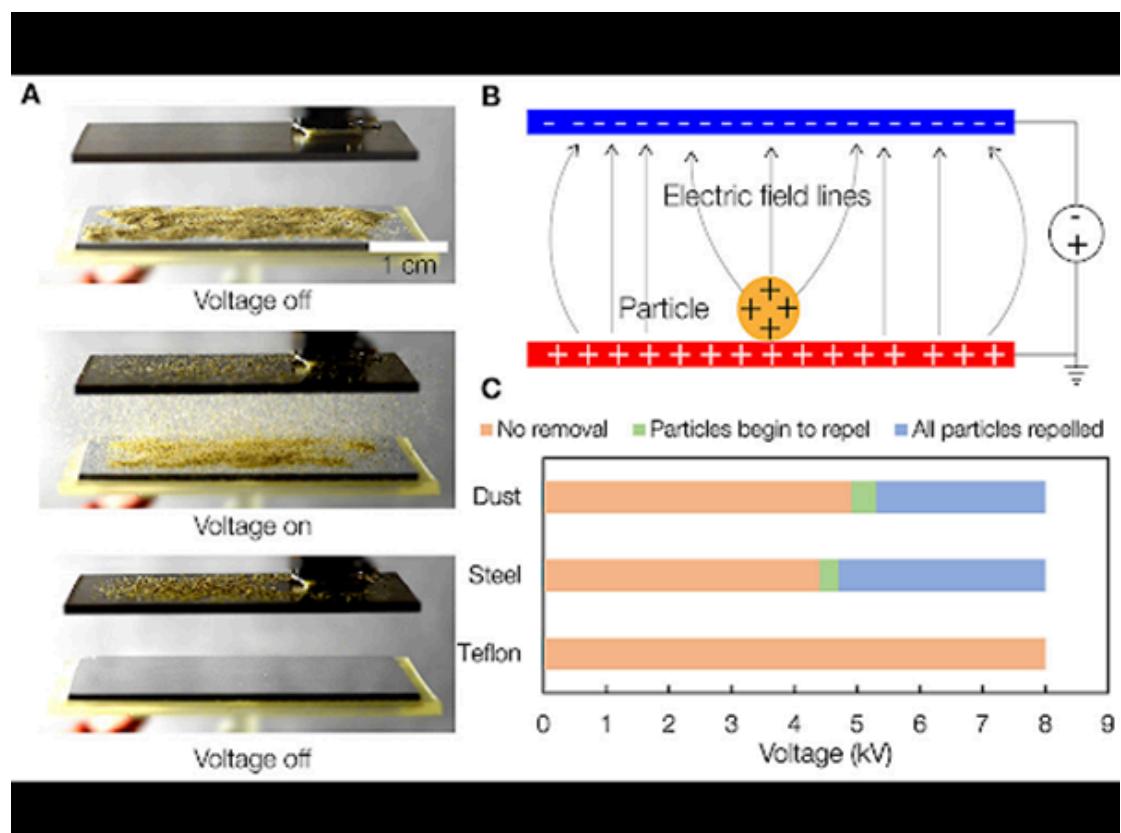


Fig 7.3 MIT Research Graph

7.5 K-DISC BUILDING

In the K-DISC Building in Trivandrum, Kerala, solar panels are frequently exposed to high humidity and airborne dust, leading to a decline in efficiency over time. Unlike desert environments, Kerala has a higher availability of water, making wet cleaning methods a viable solution. The proposed LVDC-powered automated cleaning system in the K-DISC building utilizes water-spraying techniques to keep the panels clean and cool, ensuring stable efficiency. By integrating real-time monitoring through IoT platforms, the system effectively manages energy usage while reducing the frequency of manual intervention, making it a more sustainable and cost-efficient solution. We have visited the K-DISC building for a site inspection and, based on our observations, we decided to deploy a panel bot prototype to maintain efficiency. This prototype will play a crucial role in automating the cleaning process, thereby optimizing energy output and enhancing the long-term reliability of the solar panels.

7.6 SUMMARY

Solar energy efficiency is often affected by environmental factors like dust accumulation, temperature fluctuations, and humidity, prompting innovative maintenance solutions worldwide. Bhadla Solar Park in Rajasthan, one of the world's largest solar farms, faces severe dust accumulation, leading to the adoption of automated robotic cleaning systems for continuous dust removal and improved energy output. Similarly, the Noor Abu Dhabi Solar Plant in the UAE, operating in an arid region with water scarcity, utilizes robotic dry-cleaning technologies to maintain efficiency without relying on water-based methods. MIT research highlights that dust accumulation can reduce solar panel efficiency by up to 30%, leading to the

development of hydrophobic coatings that repel dust, minimizing maintenance needs and extending panel lifespan. In Kerala's K-DISC Building, where humidity and airborne dust impact solar performance, an LVDC-powered automated cleaning system with IoT-based monitoring has been deployed, along with a panel bot prototype to ensure long-term efficiency. These case studies emphasize the significance of smart, sustainable technologies, demonstrating how automated cleaning, water-efficient methods, and advanced coatings can enhance solar panel performance and optimize energy generation in diverse environments.

8.SOLAR PANEL SURFACE CONDITIONING

8.1 INTRODUCTION

This report presents the efficiency estimation calculations conducted under both theoretical and practical approaches across various operating conditions. The study evaluates efficiency under different scenarios, including dirt accumulation, elevated temperature (hot condition), normal operating conditions, and nominal temperature settings. By comparing theoretical predictions with real-world practical results, this analysis provides valuable insights into performance variations due to environmental and operational factors. The findings highlight the impact of external conditions on efficiency and serve as a basis for optimizing performance.

8.2 PRACTICAL SURVEY

Parameters	Before Cleaning	After Cleaning
Temperature	68°C	42°C
Voltage	43 V	46 V
Current	5.64 A	7.9 A
Power	242.5 W	366 W
Efficiency	12.12 %	18.3 %

Table 8.1: A sample table and its numbering

8.2.1 Given Data

P_{max}: 449.77 W (maximum power output under standard test conditions, STC)

I_{mp}: 10.87 A (current at maximum power)

V_{mp}: 41.36 V (voltage at maximum power)

I_{sc}: 11.41 A (short-circuit current)

V_{oc}: 49.55 V (open-circuit voltage)

Nominal module operating temperature: 42°C ± 2°C

Standard test conditions (STC): 25°C, 1000 W/m² irradiance, clean panel

Efficiency drop: 0.5% per °C above 25°C

Dust/debris impact: Reduces efficiency by 20-25% (we'll assume 22.5% for this calculation)

8.2.2 Calculate Standard Efficiency at 25°C

The standard efficiency of the solar panel is calculated using the formula:

$$\text{Efficiency (\%)} = \frac{\text{Pmax (W)} \times \text{Area (m}^2\text{)}}{\text{Irradiance (W/m}^2\text{)}} \times 100$$

Panel dimensions: 2094 mm × 1039 mm = 2.094 m × 1.039 m = **2.175 m²**

Irradiance: 1000 W/m² (STC)

$$\text{Efficiency (\%)} = \frac{449.77 \times 2.175}{1000} \times 100 = 20.68\%$$

So, the **standard efficiency at 25°C** is **20.68%**.

8.2.3 Efficiency at Nominal Operating Temperature (42°C)

The nominal operating temperature is 42°C. Using the temperature impact formula:

$$\text{Efficiency Loss (\%)} = (T_{\text{actual}} - T_{\text{standard}}) \times 0.5\%$$

$$\text{Efficiency Loss (\%)} = (42 - 25) \times 0.5\% = 8.5\%$$

$$\text{Efficiency at } 42^\circ\text{C (\%)} = 20.68\% - 8.5\% = 12.18\%$$

So, the **efficiency at 42°C is 12.18%**.

8.2.4 Efficiency When Panel is Cool (e.g., 15°C)

If the panel is cooler than 25°C, the efficiency will increase. Using the same formula:

$$\text{Efficiency Gain (\%)} = (T_{\text{standard}} - T_{\text{actual}}) \times 0.5\%$$

$$\text{Efficiency Gain (\%)} = (25 - 15) \times 0.5\% = 5\%$$

$$\text{Efficiency at } 15^\circ\text{C (\%)} = 20.68\% + 5\% = 25.68\%$$

So, the **efficiency at 15°C is 25.68%**.

8.2.5 Efficiency with Dust and Debris

Dust and debris can reduce efficiency by 20-25%.

Assuming a **22.5% reduction**:

$$\text{Efficiency Loss (\%)} = 20.68\% \times 22.5\% = 4.65\%$$

$$\text{Efficiency with Dust (\%)} = 20.68\% - 4.65\% = 16.03\%$$

So, the **efficiency with dust/debris** is **16.03%**.

8.2.6 Combined Effects (Dust + High Temperature)

If the panel is both dirty and operating at 42°C:

$$\text{Efficiency at } 42^\circ\text{C with Dust (\%)} = 12.18\% - 4.65\% = 7.53\%$$

So, the **efficiency at 42°C with dust/debris** is **7.53%**.

8.3 GRAPHICAL REPRESENTATION

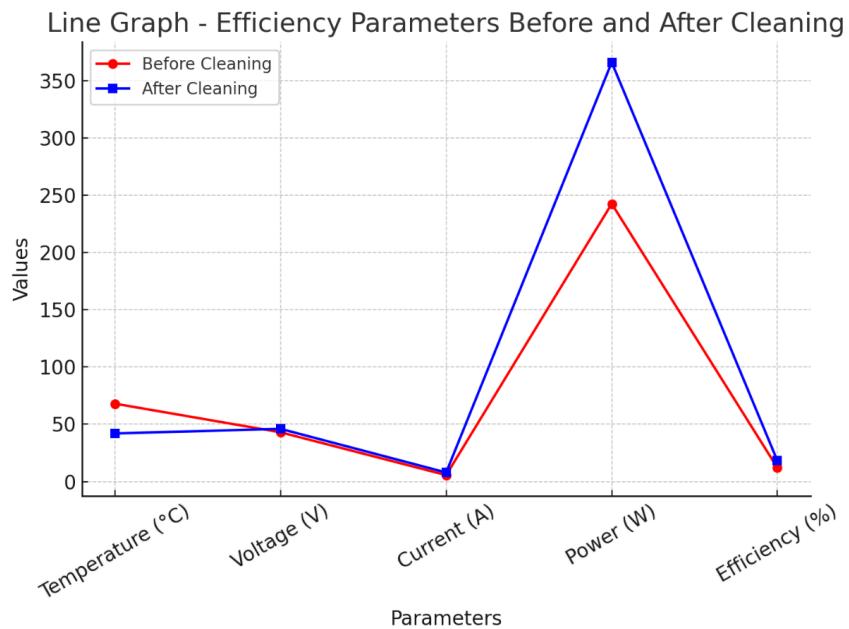


Fig 8.1: Before and After Cleaning Graph

8.4 SUMMARY

The prototype surface panel conditioning system is designed to maintain energy output while boosting long-term reliability and optimizing energy generation. The performance of solar panels is heavily affected by environmental conditions, especially temperature and cleanliness. Theoretical analysis shows that at 25°C with a clean panel, efficiency can reach 20.68%, but dust presence lowers it to 16.03%. Temperature changes also affect efficiency, with a maximum of 25.68% at 15°C, dropping to 12.18% at 42°C, and further decreasing to 7.53% when dust accumulates. Practical tests indicated that efficiency was 12.12% at 68°C before cleaning. However, after cleaning, efficiency improved to 18.3% at 42°C, highlighting the clear advantages of regular maintenance. These results stress the importance of routine cleaning to prevent performance degradation and maintain operational efficiency. Additionally, factors like humidity, shading, and panel aging can cause deviations from theoretical predictions. Effective temperature management, through methods like enhanced ventilation and reflective coatings, could further improve panel performance in hot conditions. This project highlights that with systematic cleaning, monitoring, and preventive maintenance, solar panels can achieve peak performance levels. The prototype surface panel conditioning system is crucial for maintaining energy output while enhancing long-term reliability and maximizing energy production.

9.SURFACE CONDITIONING SYSTEM ARCHITECTURE

9.1 INTRODUCTION

The Solar Panel Surface Conditioning System is designed to enhance solar panel efficiency by preventing dust accumulation through an automated cleaning mechanism. Its architecture consists of a hub, which acts as an intermediary between the panel bots, water pump circuit, and the Blynk dashboard. The hub processes commands from the user and coordinates the cleaning operation. The panel bots, mounted on the solar panels, execute the cleaning process based on the received instructions, ensuring efficient dust removal. The water pump circuit activates when commanded, spraying water onto the panels to aid in the cleaning process. The Blynk dashboard serves as the user interface, enabling remote monitoring and control of the system, ensuring seamless communication between all components for efficient and automated solar panel maintenance.

9.2 ESP32

The ESP32 is a family of cost-effective, energy-effective microcontrollers that integrate both Wi-Fi and Bluetooth capabilities. These chips feature colorful processing options, including the Tensilica Xtensa LX6 microprocessor in both binary-core and single-core configurations, the Xtensa LX7 dual-core processor, or a single-core RISC-V microprocessor.

Also, the ESP32 includes essential factors for wireless communication, similar as erected-in antenna switches, an RF balun, power amplifiers, low-noise receivers, pollutants, and power operation modules. Generally, the ESP32 is bedded in custom published circuit boards or included in development accoutrements with multiple GPIO legs and connectors, depending on the model and manufacturer.

Developed by Espressif Systems and manufactured by TSMC using a 40 nm process, the ESP32 serves as the successor to the ESP8266 microcontroller

9.3 HUB

The hub is the central control unit of the Solar Panel Surface Conditioning System, responsible for processing user commands, managing communication between different components, and ensuring the seamless operation of the system. It plays a crucial role in coordinating the panel bots and the water pump circuit while maintaining a stable connection with the Blynk dashboard for real-time monitoring and control.

9.3.1 Given Data

The hub primarily acts as an intermediary between the user interface (Blynk dashboard) and the system's hardware components. It performs the following key functions:

- Receives and processes commands from the Blynk dashboard via Wi-Fi.
- Communicates with the panel bots using an ESP-NOW mesh network for both broadcast and unicast signals.
- Synchronizes time and retrieves external data using an NTP (Network Time Protocol) server, ensuring scheduled operations.
- Controls the water pump circuit, activating or deactivating it based on received commands.

To efficiently execute these functions, the hub employs two ESP32 microcontrollers, each designated for a specific set of tasks.

9.3.2 Electronic Components of the Hub

Dual ESP32 Setup

The hub integrates two ESP32 microcontrollers to efficiently manage communication and processing tasks:

ESP32 for Internet Connectivity & Blynk Integration

- Establishes Wi-Fi connectivity to interface with the Blynk dashboard.
- Retrieves real-time data from the NTP server to synchronize system operations.
- Communicates with the user application, processing received commands.

ESP32 for ESP-NOW Mesh Networking

- Facilitates wireless communication with the panel bots via ESP-NOW protocol.
- Sends broadcast and unicast signals for controlling multiple panel bots simultaneously.
- Communicates with the main ESP32 via a UART (Universal Asynchronous Receiver-Transmitter) connection for command execution.

The ESP-NOW mesh network allows low-latency and direct peer-to-peer communication between the hub and panel bots, ensuring minimal power consumption and efficient data transfer.

9.3.3 Programming & Software Implementation

9.3.3.1 Development Environment – Arduino IDE

The hub is programmed using the Arduino IDE, an open-source platform that supports ESP32 microcontrollers. It provides an intuitive coding environment,

facilitating code compilation, debugging, and firmware uploads.

Key libraries used in programming the hub include:

- **WiFi.h** – To establish internet connectivity.
- **BlynkSimpleEsp32.h** – To communicate with the Blynk IoT platform.
- **WiFiUdp.h & NTPClient.h** – To synchronize real-time data from an NTP server.
- **esp_now.h** – To enable ESP-NOW communication between the hub and panel bots.
- **HardwareSerial.h** – To manage UART-based communication between the two ESP32 controllers.

9.3.3.2 Working of the Program

- The Wi-Fi ESP32 initializes the network connection and authenticates with the Blynk server.
- It fetches real-time data from an NTP server for system time synchronization.
- Based on the user's input from the Blynk dashboard, it sends control commands to the ESP-NOW ESP32 via UART.
- The ESP-NOW ESP32 transmits broadcast/unicast signals to the panel bots, triggering the cleaning process.
- The water pump is activated only when necessary, optimizing water usage.

9.3.4 Communication Flow within the Hub

- User sends a command via the Blynk App.
- The Wi-Fi ESP32 processes the command and sends it to the ESP-NOW ESP32 through UART.
- The ESP-NOW ESP32 transmits the command to the panel bots.

- Panel bots respond accordingly, initiating the cleaning mechanism.
- The hub updates the Blynk dashboard with the real-time status of the operation.

9.3.5 Hardware

Components required

- IP68 Water resistant box
- ESP32 5V LED
- 220 ohm resistor
- 10 Watts solar panel (loom solar panel)
- 22V CN3722 Battery charge controller
- 2s Battery
- Voltage Regulator LM7805
- Max6675 temperature sensor
- 30 Gauge wire

There are two esp32, first one is for WIFI connectivity and the second one is for ESP-NOW broadcast connectivity. TX2 of the first esp is connected to second esp Rx2, similarly the TX2 of the second esp is connected to the RX2 of first esp. Max 6675 temperature sensor having 3 data pins SO SCK CS are connected to D21, D22 and D23 of the second esp respectively. A 5V LED is connected to pin number D23 of the first esp32 through 220 ohm resistor. 10 Watts solar panel is connected through CL3722 solar battery charge controller which is connected to 186502s battery. A LM7805 voltage regulator is used to regulate 7.4V to 5V. The 5V is then supplied to Vin pins and the 5V of the temperature sensor. Fig 9.1 shows the Hub prototype image.



Fig 9.1 Hub Hardware

9.3.6. Schematic Diagram

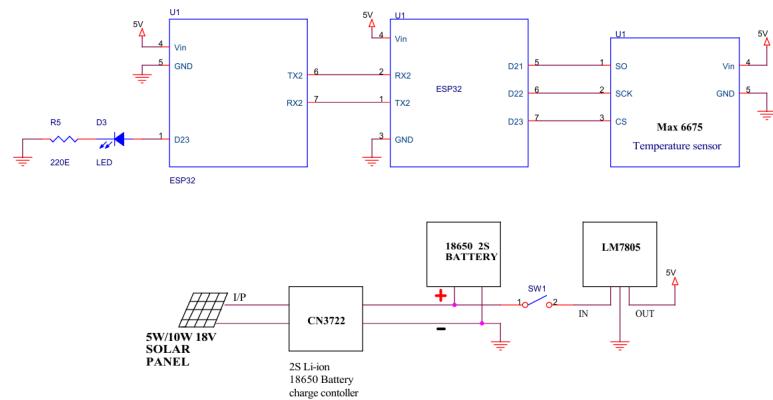


Fig 9.2 Circuit for Hub

9.3.7 ESP WIFI flowchart

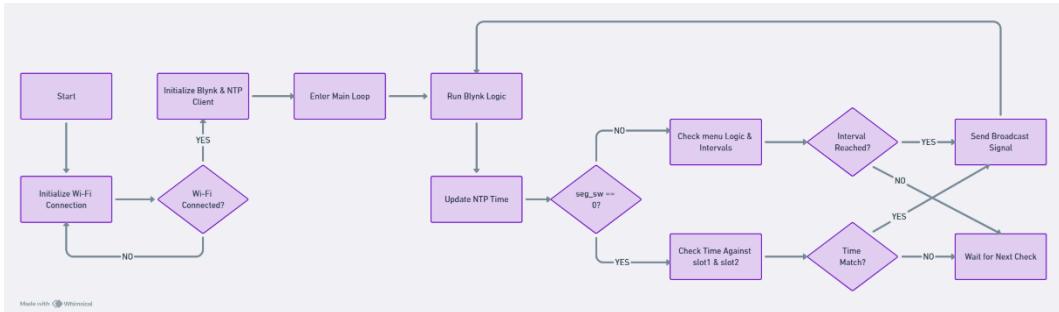


Fig 9.3: ESP WIFI

9.3.8 ESP NOW flowchart

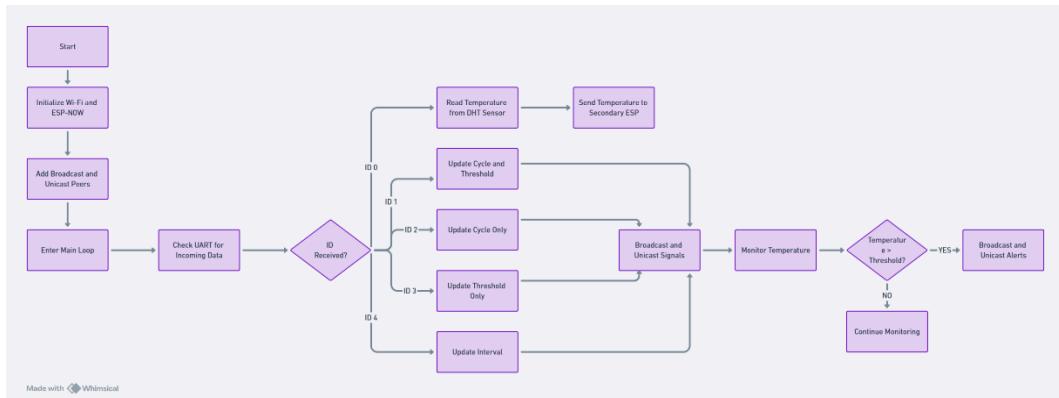


Fig 9.4: ESP NOW

9.4 PANEL BOTS

The panel bot is an autonomous cleaning unit responsible for removing dust and debris from the surface of solar panels. It operates based on commands received from the hub via an ESP-NOW mesh network and ensures optimized movement and

control for effective cleaning. Each panel bot is powered by an ESP8266 microcontroller, which facilitates wireless communication, motor control, and sensor integration for efficient operation.

9.4.1 Panel Bot Functionality

The panel bot is designed to:

- Receive commands from the hub via ESP-NOW communication.
- Activate the cleaning mechanism and move across the panel surface.
- Detects panel edges using limit switches to prevent overrunning.
- Detects inactivity and returns to its home position after a period of no command reception.
- Send status updates back to the hub after completing a cleaning cycle.

These functionalities ensure that the panel bot autonomously performs the cleaning task with minimal energy consumption and without requiring continuous user intervention.

9.4.2 Electronic Components of the Panel Bot

Each panel bot consists of the following hardware components:

- ESP8266 Microcontroller – Manages wireless communication, motor control, and sensor input processing.
- DC Motor with Motor Driver (L298N) – Controls the movement of the cleaning mechanism.
- Brush Assembly – Mechanically removes dust and debris from the panel surface.
- Limit Switches – Detects the edges of the panel and prevents the bot from moving beyond its designated area.

- Power Supply (Lithium-Ion Battery + Solar Charging Circuit) – Ensures continuous operation with efficient power management.

9.4.3 Communication And Command Execution

9.4.3.1 Receiving Broadcast and Unicast Commands

The panel bot communicates wirelessly with the hub using ESP-NOW, a low-latency communication protocol.

- Broadcast Commands: Sent by the hub to all panel bots simultaneously (e.g., "CLEAN_START" to initiate cleaning).
- Unicast Commands: Sent by the hub to a specific panel bot when individual control is required (e.g., "BOT_2_STOP" to halt a particular bot).

9.4.3.2 Command Execution Process

Listening for ESP-NOW Commands

- The ESP8266 continuously listens for messages from the hub.
- If a valid command (e.g., "CLEAN_START") is received, the bot initiates the cleaning cycle.

Cleaning Cycle Execution

- The bot moves across the panel, rotating the brush mechanism to remove dust.
- It continues operation until a limit switch is triggered, indicating the panel's edge.

Limit Switch Handling

- When a limit switch is activated, the bot either reverses direction or stops, depending on the command logic.
- This ensures the bot does not fall off or damage the solar panel.

Inactivity Detection & Auto Reset

- If the bot does not receive a new command for a set period (e.g., 5 minutes), it returns to its home position to conserve power.
- This prevents unnecessary operation and optimizes energy usage.

Status Update to Hub

- After completing the cleaning cycle, the bot sends a status message (e.g., "BOT_3_DONE") to inform the hub.

9.4.4 Communication Flow of Panel Bots

- The hub broadcasts a cleaning command ("CLEAN_START") to all panel bots.
- The panel bots begin moving and cleaning the panel surface.
- The bots detect edges using limit switches and adjust their movement accordingly.
- If no command is received for a predefined period, the bots return to their home positions.
- The panel bots send a completion status ("BOT_X_DONE") to the hub.

9.4.5 Hardware

- 60 rpm gate motor high torque
- Gear assembly
- Iron casing for gear assembly
- Synthetic brush
- Aluminium frame
- M3 and M5 nut and bolt
- Polycarbonate transparent sheet
- 4 inch wheel

- IP68 water resistant box
- Limiter switch
- PWM charge controller
- Lithium ion battery 12V
- L298N motor driver
- Esp 8266
- 220, 2.2k, 10k ohm resistor
- 5V LED
- LM7805 voltage regulator
- 10/20 watts solar panel

The 60 rpm gear motor is connected to a gear assembly of the panel bot. Motor is controlled by an L298N motor driver through esp32.In2, In3 and In1 and In4 are shorted. D2 and D5 pins of esp 8266 are connected to In1 and In2 respectively. Limiter switches S1 and S2 are connected to D6 and D7, these are active low and pulled up high using 10k and 2.2k resistors respectively. An LED is connected to pin number D1 through a 220 ohm resistor. Solar panel is connected to the battery through the PWM battery charge controller and the output is taken to LM7805 and 5V is obtained. The 5V is used to power ESP8266 through Vin pin. CAD designs of the panel bots are shown below in Fig 9.5 & 9.6.

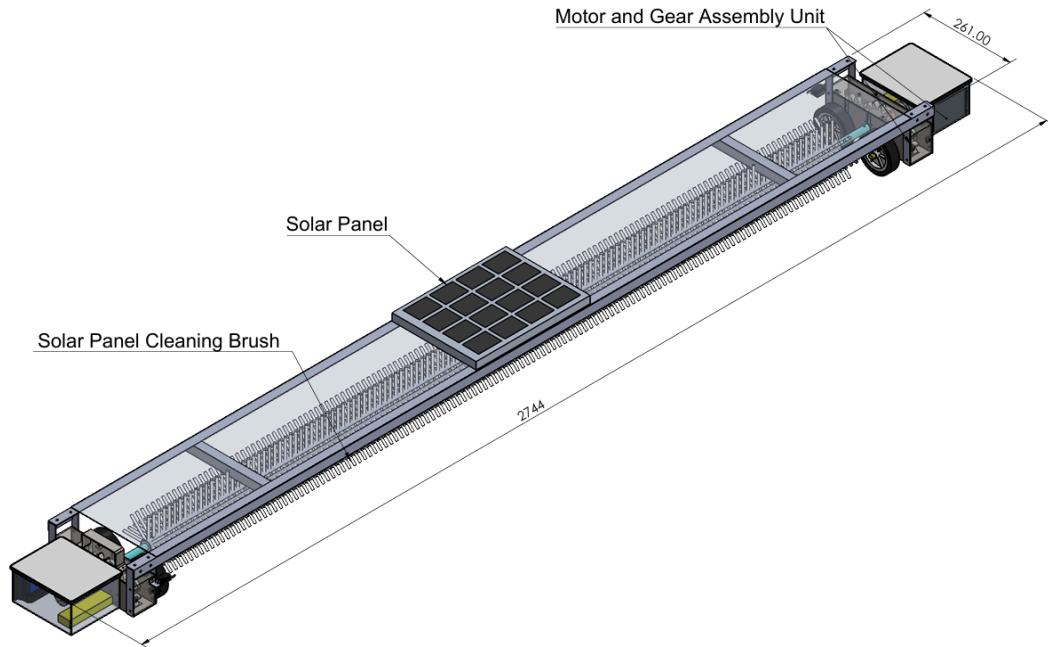
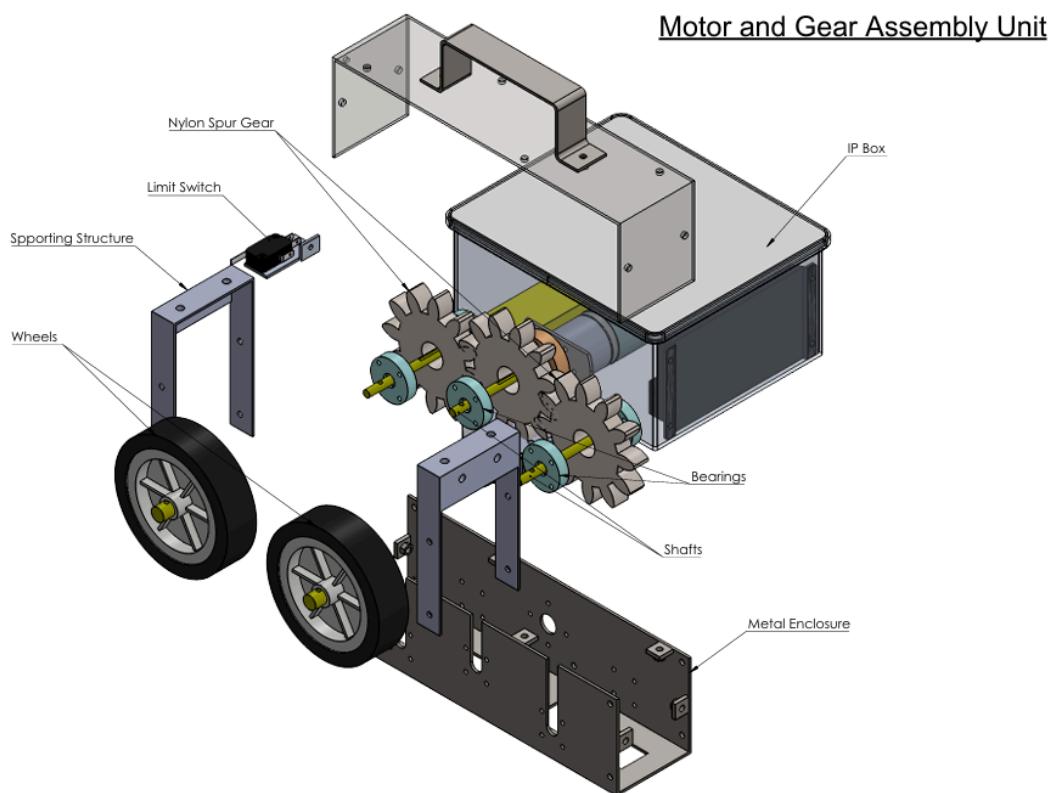


Fig 9.5: CAD Drawing of Panel Bots



Motor and Gear Assembly Unit

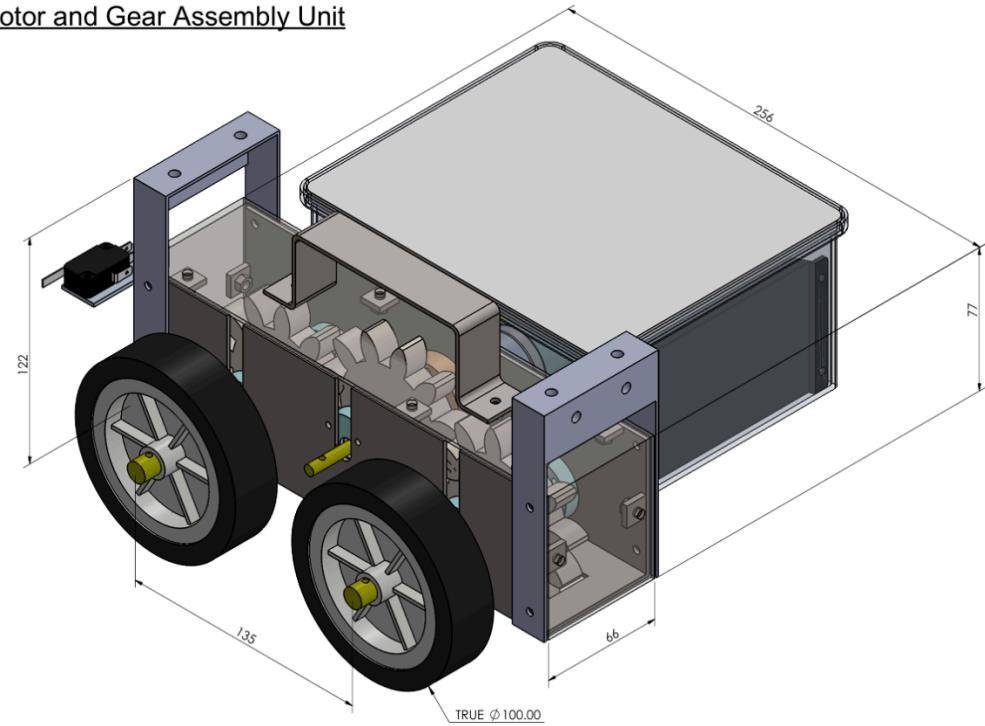


Fig 9.6: CAD Drawing of Motor and Gear assembly unit

9.4.6 Schematic

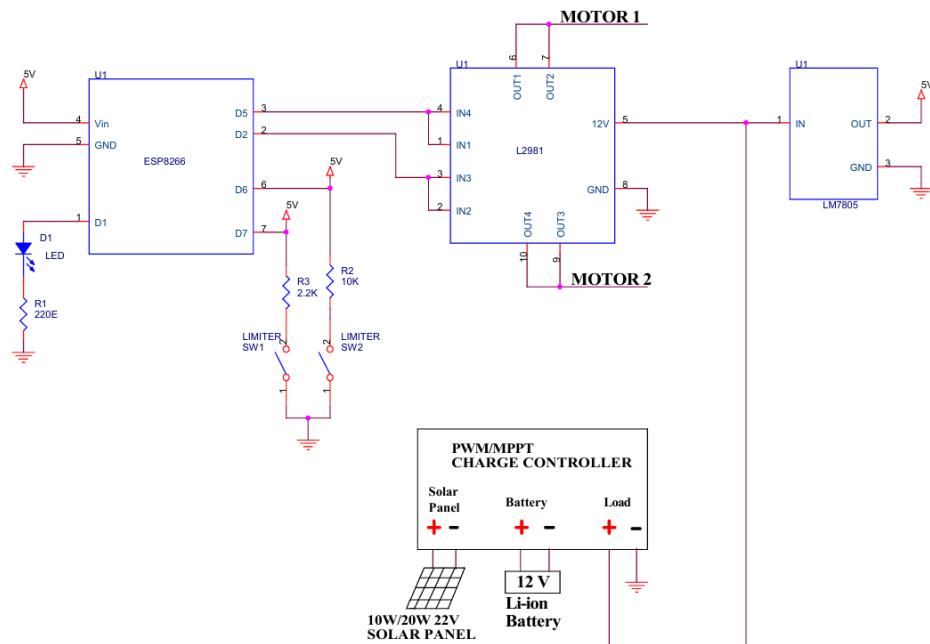


Fig 9.7: Panel bot Schematic

9.4.7 Panel Bot FlowChart

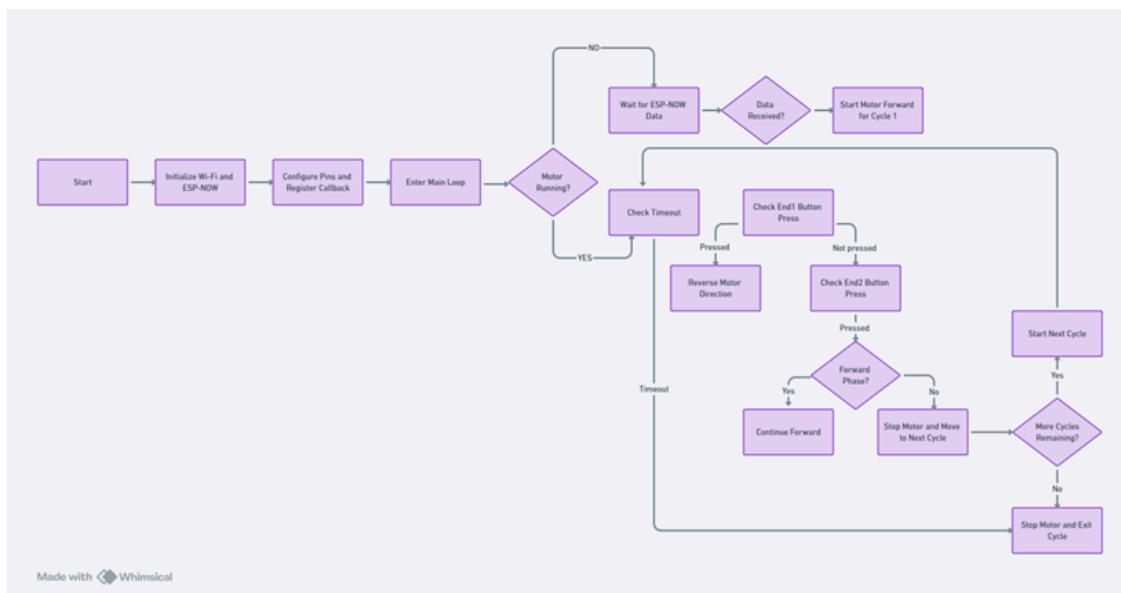


Fig 9.8: Panel Bots

9.5 WATER PUMP CIRCUIT

The water pump circuit is responsible for controlling an LVDC water pump, which sprays water onto the solar panel surface for cleaning. The circuit is triggered via a unicast command received from the hub over ESP-NOW communication. It uses an ESP8266 microcontroller to activate a relay module, which in turn controls the motor. This setup ensures efficient water usage while integrating seamlessly with the overall cleaning system.

9.5.1 Electronic Components

ESP8266 Microcontroller – Receives commands from the hub and controls the relay module.

Relay Module (5V or 12V, Single-Channel Relay) – Acts as a switch to control the LVDC motor.

LVDC Motor-Driven Water Pump – Pumps water onto the solar panel surface.

DC-DC Regulator (5V for ESP8266 & Relay) – Provides power for different components

9.5.2 Working

Receiving Unicast Command from Hub

- The hub sends a unicast message specifically to the ESP8266 in the water pump circuit.
- The ESP8266 listens for the command and processes the received instruction.

Activating the Relay Module

- When the command ("PUMP_ON") is received, the ESP8266 activates the relay module.
- The relay closes the circuit, allowing LVDC power to flow to the pump.

Water Pump Activation

- The pump starts spraying water onto the solar panels for cleaning.
- The system ensures water is sprayed only when necessary to optimize usage.

Automatic Pump Shutdown

- After a preset duration (e.g., 15–30 seconds) or when a stop command ("PUMP_OFF") is received, the ESP 8266 deactivates the relay.
- This cuts power to the LVDC motor, stopping the water pump.

9.5.2 Hardware

- Esp 8266
- 5V relay
- DC-DC buck converter (SZ-BKHV)
- 5V LED with 220 ohm resistor

The input pin of the relay is connected with the D1 pin of ESP8266. The 5V LED is connected to the D2 pin through a 220 ohm resistor. The positive terminal of the 48V LVDC is connected to the common pin of the relay. The output from the relay is taken from the normally closed pin. The DC-DC buck converter is used to regulate 48V DC to 5V DC which is used to power the microcontroller via Vin pin. Pump control system prototype is shown in fig 9.9 and schematic is shown in fig 9.11.



Fig 9.9: Water Pump Hardware

9.5.3 Pump Relay Flow Chart

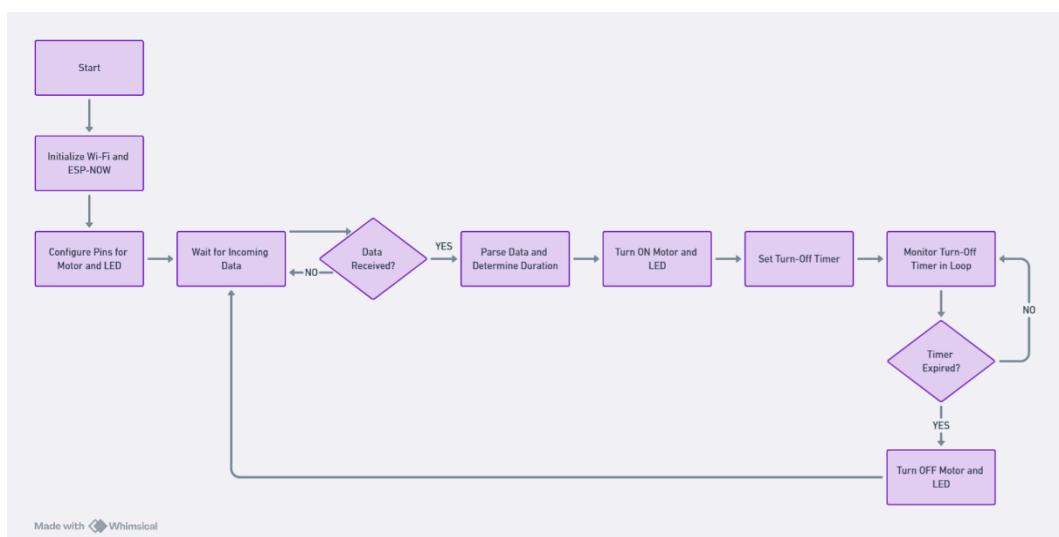


Fig 9.10: Pump Relay

9.5.4 Schematic

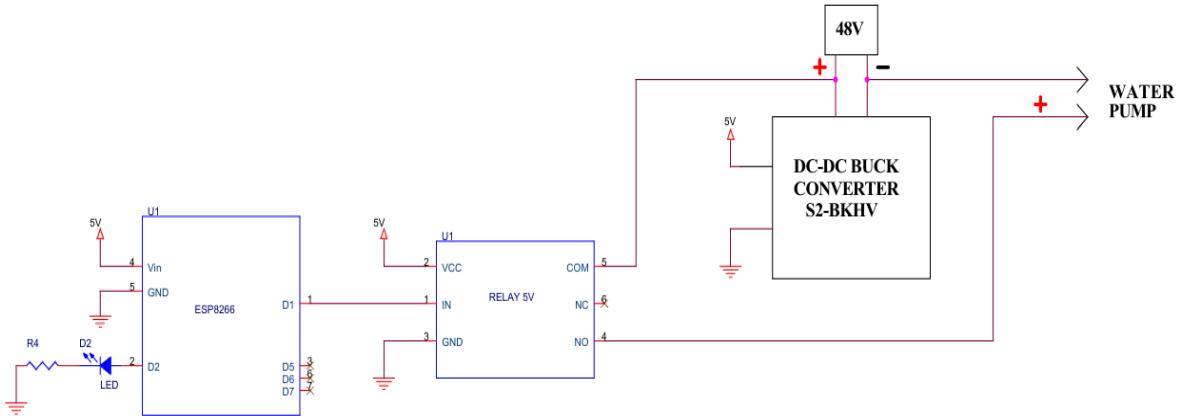


Fig 9.11: Water Pump Schematic

9.6 MAX6675 TEMPERATURE SENSOR

The MAX6675 performs cold-junction compensation and digitizes the signal from a type-K thermocouple. The data is in a 12-bit resolution, SPI-compatible, read-only format. This motor resolves temperatures to 0.25 °C, allows readings as high as 1024 °C, and exhibits thermocouple delicacy of 8 LSBs for temperatures ranging from 0 °C to 700 °C. The MAX6675 is available in a small, 8-leg SO package.

The MAX6675 temperature detector is a pivotal element in the Solar Panel Surface Conditioning System, responsible for covering the solar panel temperature in real-time. It's connected to the mecca, which continuously checks the panel's temperature. When the temperature exceeds a stoner-defined threshold, the panel bots are automatically activated to clean the panels, icing optimal effectiveness and life.

This intelligent robotization prevents inordinate dust accumulation, which can act as an insulator and increase panel temperature, reducing energy conversion effectiveness. By integrating temperature-ground activation, the system optimizes

energy consumption by icing that panel bots operate only when necessary, rather than at fixed intervals.

The MAX6675 Fig 4.8 was chosen over the others like the DHT11 due to its advanced temperature range, better out-of-door trustability, and fast response time, making it ideal for covering solar panels that can reach extreme temperatures. druggies can set custom temperature thresholds via the Blynk dashboard, making the system adaptable to different climate conditions. The mecca continuously processes temperature data from the MAX6675 and, if the threshold is exceeded, sends a unicast ESP- NOW command to spark the panel bots, which clean the panels and help dissipate heat. Once the cleaning cycle is completed, the mecca resumes monitoring, icing optimized performance. The advantages of the MAX6675 over standard sensors like the DHT11 are summarized below:

Feature	MAX6675 (Thermocouple)	Alternative Sensors (e.g., DHT11)
Temperature Range	-200°C to 700°C	0°C to 50°C
Accuracy	±1.5°C	±2°C to ±5°C
Outdoor Suitability	Excellent (Resistant to humidity & dust)	Limited (Affected by humidity)
Response Time	Fast	Slow
Integration with Hub	SPI Communication	Digital/Analog

Table 9.1: MAX6675 Feature Table

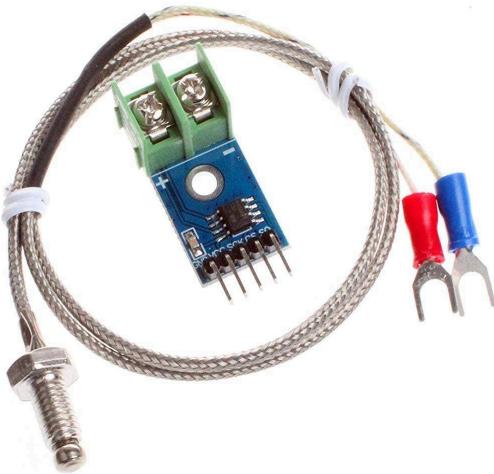


Fig 9.12: MAX6675 Temperature Sensor

9.7 H-BRIDGE MOTOR DRIVER

The H-Bridge Motor Driver [L298N](#) shown in Fig 9.13 is used to control the DC motors of the panel bots, enabling precise movement across the solar panel surface. This dual-channel motor driver allows the ESP8266 to regulate the direction and speed of the motors by controlling the voltage applied to them. The H-bridge configuration enables bidirectional movement, ensuring that the panel bots can move forward, reverse, and stop when required. The L298N module can drive two DC motors simultaneously and supports PWM (Pulse Width Modulation) speed control, making it ideal for the panel bot's movement control. Additionally, it includes built-in heat sinks to dissipate excess heat, enhancing durability and efficiency. The L298N motor driver is chosen for this application due to its high current-handling capability, wide voltage range, and reliability in outdoor conditions.

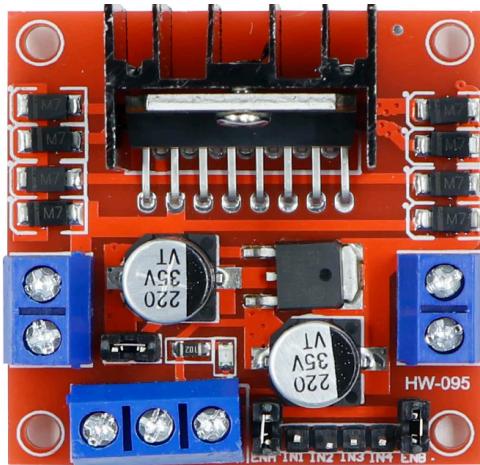


Fig 9.13: H-Bridge Motor Driver

Below are the key specifications from the L298N datasheet:

Specification	L298N Motor Driver
Control logic Voltage	5 V
Output Current (per channel)	2A(Continuous) and 3A(Peak)
Operating Voltage	5V – 35V
Number of Channels	2 (Dual H-Bridge)
PWM Control	Yes
Max Power Dissipation	25W

Built-in Protection	Thermal Shutdown
Communication	TTL Logic (Compatible with ESP8266)

Table 9.2: Key specifications of L298N Motor Driver

9.8 BLYNK IOT PLATFORM

The [Blynk](#) IoT platform plays a vital role in the Solar Panel Surface Conditioning System by enabling remote monitoring, automation, and control of the entire system through a mobile app. It allows users to send commands, monitor sensor data, and automate cleaning cycles based on real-time conditions. Through event-based triggers, the system can automatically activate panel bots and the water pump when the temperature threshold is exceeded, ensuring optimal cleaning without manual intervention. Additionally, Blynk's cloud-based data storage helps track system

performance over time, allowing users to analyse cleaning efficiency, energy savings, and environmental conditions. This seamless integration of remote access, automation, and data logging makes Blynk an essential component for enhancing operational efficiency and user convenience in maintaining solar panel performance. The Blynk iot interface is shown in fig 9.14.

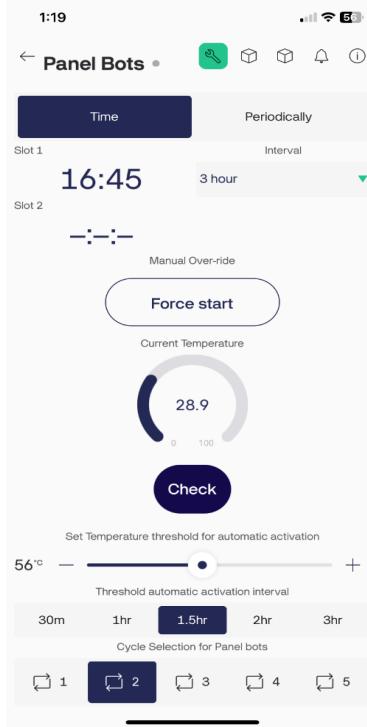


Fig 9.14: Blynk IOT Interface

9.9 ESP32, ESP8266, AND ESP-NOW

The Solar Panel Surface Conditioning System relies on ESP32 and ESP8266 microcontrollers for efficient data processing and wireless communication. The ESP32, serving as the hub, manages real-time monitoring, automation, and user commands via the Blynk IoT platform, while the ESP8266 is used in panel bots and the water pump circuit for executing cleaning tasks. These microcontrollers communicate using ESP-NOW, an optimized wireless protocol that enables low-latency, multi-peer data exchange without relying on an external Wi-Fi router.

9.9.1 ESP32 and ESP8266 Specification

Feature	ESP32	ESP8266
Processor	Dual-core Xtensa LX6 (240 MHz)	Single-core Tensilica L106 (80 MHz)
Connectivity	WIFI and Bluetooth	WIFI
GPIO Pins	34	17
Analog Inputs	18 (12-bit ADC)	1 (10-bit ADC)
PWM Support	Yes	Yes
Communication Interfaces	UART, SPI, I2C, I2S, CAN	UART, SPI, I2C
Power Consumption	Low	Ultra-low
Ideal Use Case	IoT applications with multiple sensors & high processing needs	Compact, cost-effective Wi-Fi-based automation

Table 9.3: Specification of ESP32 and ESP8266

9.9.2 ESP NOW Radio Communication

ESP-NOW is a fast, energy-efficient peer-to-peer wireless communication protocol developed by Espressif, enabling direct device-to-device communication without needing an active internet connection or Wi-Fi router.

- Low Latency (<2ms) – Ensures near-instant command execution.
- Multi-Peer Communication – Allows one-to-many and many-to-one communication for efficient coordination between the hub and multiple devices.
- Energy Efficient – Reduces power consumption compared to standard Wi-Fi communication.
- Independent Network – Works without an active Wi-Fi connection, ensuring seamless operation even in remote locations.

In the system shown in fig 9.15, ESP-NOW enables the ESP32 hub to send broadcast commands to all panel bots simultaneously or unicast signals to specific devices, ensuring synchronized, real-time control of cleaning operations and water spraying cycles. By leveraging ESP32, ESP8266, and ESP-NOW, the system achieves reliable, low-power automation for maintaining solar panel efficiency.

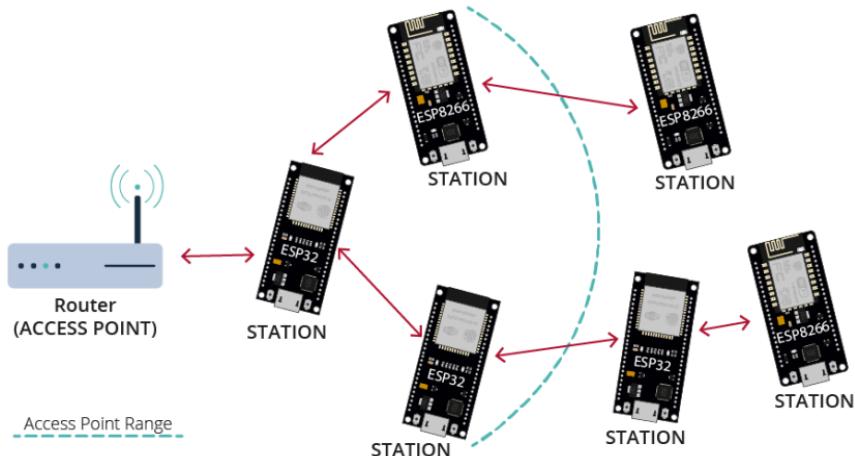


Fig 9.15: ESP Now Radio Communication

9.10 LI-ION BATTERY 18650

The Li-Ion 18650 shown in fig 9.16 battery is used as the primary power source for the panel bots and water pump circuit, ensuring reliable and energy-efficient operation. These batteries are chosen due to their high energy density, long cycle life, and low self-discharge rate, making them ideal for continuous outdoor applications.

The panel bots, powered by ESP8266 microcontrollers, require a stable power supply to drive the L298N motor driver and DC motors, which is efficiently handled by the 18650 battery pack. Additionally, the water pump circuit also utilizes the 18650 battery to power the relay and pump motor when triggered by the hub (ESP32). To ensure prolonged battery life, the system integrates solar charging and power management circuits, allowing batteries to recharge using solar energy, making the setup self-sustainable and reducing dependency on external power sources for uninterrupted operation.



Fig 9.16: LI-ION BATTERY 18650

9.11 FINAL PROTOTYPE



Fig 9.17: Panel Bot Prototype

9.12 SUMMARY

The Solar Panel Surface Conditioning System enhances solar panel efficiency through automated cleaning. The ESP32-based hub controls panel bots (ESP8266) and a water pump circuit, with Blynk IoT enabling remote monitoring. The MAX6675 temperature sensor ensures cleaning is activated when panel temperature exceeds a user-defined threshold. ESP-NOW facilitates real-time, low-latency wireless communication between the hub, panel bots, and pump circuit. The panel bots utilize the L298N motor driver for bidirectional movement, ensuring efficient cleaning. Powering the system, Li-Ion 18650 batteries provide a reliable, rechargeable energy source, supporting ESP8266, motors, and relays, with solar charging integration for sustainability. This energy-efficient, automated system is shown in fig 9.17 can be used to optimize panel cleaning, reducing manual intervention and enhancing long-term performance in diverse environmental conditions.

10.COMPONENTS/ HARDWARE

Updated GITHUB CODE: [Alan-GITHUB](#)

10.1 TABLE OF COMPONENTS WITH PRODUCT LINK

Sl. No.	Item No.	Supplier's Code / Description	Package / Doc. No.	Supplier / Mnfr.	Qty / set	Ckt. Ref./ Schedule	Product link
1		HUB components					
	1	220 Ohm 0.25W Metal Film Resistor		Generic	1	R5	Robu.in
	2	XL-504UBD- blue LED		XINGLIGHT	1	D3	Robu.in
	3	MAX6675 Module + K Type Thermocouple		Generic	1	U3	Robu.in
	4	ESP32 30pin			2	U1,U2	Robocraze
	5	Loom Solar 10 Wp		LOOM SOLAR	1		loomsolar
	6	CN3722		xcluma			xcluma
	7	18650 Lithium ion Rechargeable		HKN ENTERPRISES	1		Amazon
	8	Lm7805			1		Robu.in
2		Panel Bot components					
	1	XL-504UBD-blue LED		XINGLIGHT	1	D1	Robu.in
	2	220 Ohm 0.25W Metal Film Resistor		Generic	1	R1	Robu.in
	3	2.2k Ohm 0.25W Metal Film Resistor		Generic	1	R3	Robu.in
	4	MFR-25FBF52-10K-YAGEO-Through Hole Resistor		YAGEO		R2	Robu
	5	Z Series Roller Lever Limit Switch	Z-15GW2-B	Omron		SW1,SW2	twen.rs
	6	NodeMcu ESP8266					Robu.in
	7	L298N					Robu.in

Sl. No.	Item No.	Supplier's Code / Description	Package / Doc. No.	Supplier / Mnfr.	Qty / set	Ckt. Ref./ Schedule	Product link
	8	LM7805					Robu.in
	9	Loom Solar 20 Wp		LOOM SOLAR			loomsolar
	10	Solar Charge Controller Regulator PWM					xculma
	11	Li-Ion Battery Pack 11.1v 2.2Ah					Robu.in
3		Pump components					
	1	220 Ohm 0.25W Metal Film Resistor		Generic		R4	Robu.in
	2	XL-504UBD-blue LED		XINGLIGHT		D2	Robu.in
	3	5V Relay		Generic			Robu.in
	4	DC9V-120V 100V96V84V72V24V to 5V 3A		Generic			Robu.in
	5	NodeMcu ESP8266					Robu.in

Table 10.1:Components with Product Link

11.CONCLUSION

The distribution of DC power in homes, reduce losses that typically occur when AC power is converted to DC, and enable better integration of renewable energy sources which naturally generate DC power. Implementing an LVDC Motion-Sensor-Controlled Energy Management System and offers a robust approach to maintain energy usage in renewable systems. The LVDC system, combined with motion-sensor control, reduces unnecessary consumption and conversion losses. This approach promotes energy conservation and cost savings.

The Solar Panel Surface Conditioning System successfully addresses the issue of dust accumulation on solar panels, improving their efficiency and ensuring optimal energy output. This solution is particularly beneficial for regions with high dust levels and limited manual maintenance access. By preventing power loss due to dust accumulation, the system enhances the long-term performance of solar panels, making it an effective and energy-efficient solution for maintaining solar infrastructure.

REFERENCES

1. T. Hakala, P. Järventausta, and T. Lähdeaho, “The utilization potential of LVDC distribution,” in Proc. CIRED 22nd Int. Conf. Elect. Distrib., Stockholm, Sweden, 2013.
2. T. Kaipia, P. Salonen, J. Lassila, and J. Partanen, “Application of low voltage DC-distribution system—a techno-economical study,” in Proc. CIRED 19th Int. Conf. Elect. Distrib., Vienna, Austria, 2007.
3. T.-F. Wu, Y.-K. Chen, G.-R. Yu, and Y.-C. Chang, “Design and development of DC-distributed system with grid connection for residential applications,” in Proc. 8th Int. Conf. Power Electronics, ECCE Asia, 2011, pp. 235–241.
4. H. Qasem, T. R. Betts, H. Mällejans, H. AlBusairi, and R. Gottschalg, “Dust-induced shading on photovoltaic modules,” Prog. Photovolt., Res. Appl., vol. 22, no. 2, pp. 218–226, Feb. 2014
5. Nasib Khadka, Aayush Bista , Binmara Adhikari ,Ashish Shreshtha, Diwakar Bista and Brijesh Adhikary,” Current Practices of Solar Photovoltaic Panel Cleaning System and Future Prospects of Machine Learning Implementation” July 2020

ANNEXURE

Updated GITHUB CODE: <https://github.com/ALANGABRIEL007/Solar-Cleaning>

1) PROGRAM CODE FOR PANEL BOTS

```
#include <ESP8266WiFi.h>
#include <espnow.h>

// Define LED and motor driver pins
#define LED_PIN 5      // LED indicator
#define MOTOR_IN1 4    // GPIO for Motor IN1
#define MOTOR_IN2 14   // GPIO for Motor IN2

// Define input buttons for End1 and End2
#define END1_BUTTON_PIN 12 // GPIO for End1 button
#define END2_BUTTON_PIN 13 // GPIO for End2 button

// Variables for motor state and timing
bool motorRunning = false;
unsigned long lastActionTime = 0;
unsigned long timeoutPeriod = 30000; // 10-second timeout

int currentCycle = 0; // Current cycle count
int totalCycles = 0; // Total cycles to perform
bool forwardPhase = true; // Indicates forward or reverse motor state

// Callback function to handle incoming ESP-NOW data
void onDataRecv(uint8_t *mac, uint8_t *data, uint8_t len) {
    if (len > 0) {
```

```

int receivedData = data[0];
Serial.print("Data received: ");
Serial.println(receivedData);

// Determine total cycles based on received data
if (receivedData >= 5) {
    totalCycles = 5;
} else {
    totalCycles = max(receivedData, 1);
}

// Start first cycle
currentCycle = 1;
motorRunning = true;
lastActionTime = millis();

// Start motor forward
digitalWrite(MOTOR_IN1, HIGH);
digitalWrite(MOTOR_IN2, LOW);
forwardPhase = true;
Serial.println("Motor started forward for cycle 1");

}

}

void setup() {
    Serial.begin(115200);

    // Initialize pins

```

```

pinMode(LED_PIN, OUTPUT);
pinMode(MOTOR_IN1, OUTPUT);
pinMode(MOTOR_IN2, OUTPUT);
pinMode(END1_BUTTON_PIN, INPUT_PULLUP);
pinMode(END2_BUTTON_PIN, INPUT_PULLUP);

// Initialize Wi-Fi and ESP-NOW
WiFi.mode(WIFI_STA);
if (esp_now_init() != 0) {
    Serial.println("Error initializing ESP-NOW");
    return;
}
esp_now_set_self_role(ESP_NOW_ROLE_SLAVE);
esp_now_register_recv_cb(onDataRecv);
}

void loop() {
    unsigned long currentTime = millis();

    if (motorRunning) {
        // Check for timeout
        if (currentTime - lastActionTime > timeoutPeriod) {
            Serial.println("Timeout: Motor stop due to inactivity");
            motorRunning = false;
            stopMotor();
            return;
        }
    }
}

```

```

// Check for End1 button press (Reverse direction)

if (digitalRead(END1_BUTTON_PIN) == LOW) {
    if (forwardPhase) {
        Serial.println("End1 button pressed: Motor reverse");
        forwardPhase = false;
        digitalWrite(MOTOR_IN1, LOW);
        digitalWrite(MOTOR_IN2, LOW);
        delay(750);
        digitalWrite(MOTOR_IN1, LOW);
        digitalWrite(MOTOR_IN2, HIGH);
    } else {
        Serial.println("End1 button pressed: No action (already reversing)");
    }
    lastActionTime = currentTime;
    delay(100); // Basic debounce
}

// Check for End2 button press (Stop or move to next cycle)

if (digitalRead(END2_BUTTON_PIN) == LOW) {
    if (forwardPhase) {
        Serial.println("End2 button pressed: Motor continues forward");
        // Continue forward
        digitalWrite(MOTOR_IN1, LOW);
        digitalWrite(MOTOR_IN2, LOW);
        delay(750);
        digitalWrite(MOTOR_IN1, HIGH);
        digitalWrite(MOTOR_IN2, LOW);
    } else {
}

```

```

Serial.println("End2 button pressed: Motor stopped for this cycle");

forwardPhase = true; // Reset for the next cycle

currentCycle++;

if (currentCycle > totalCycles) {

    motorRunning = false;

    Serial.println("All cycles completed. Motor stopped.");

    stopMotor();

} else {

    Serial.print("Starting next cycle: ");

    Serial.println(currentCycle);

    // Start forward for next cycle

    digitalWrite(MOTOR_IN1, LOW);

    digitalWrite(MOTOR_IN2, LOW);

    delay(750);

    digitalWrite(MOTOR_IN1, HIGH);

    digitalWrite(MOTOR_IN2, LOW);

}

}

lastActionTime = currentTime;

delay(100); // Basic debounce

}

}

// LED indicator (ON during motor operation)

digitalWrite(LED_PIN, motorRunning);

}

// Function to stop the motor

```

```

void stopMotor() {
    digitalWrite(MOTOR_IN1, LOW);
    digitalWrite(MOTOR_IN2, LOW);
    motorRunning = false;
    digitalWrite(LED_PIN, LOW);
    Serial.println("Motor completely stopped.");
}

```

2) PROGRAM CODE FOR HUB

HUB ESPNOW

```

#include <esp_now.h>
#include <WiFi.h>
#include "max6675.h"

// MAX6675 sensor configuration

int ktcSO = 21;
int ktcCS = 23;
int ktcCLK = 22;

MAX6675 ktc(ktcCLK, ktcCS, ktcSO);

int cycle = 0;
int threshold = 35; // Default threshold value
int threshintv = 0;
unsigned long lastBroadcastTime = 0; // Stores the last time a broadcast was sent
unsigned long delayDuration = 0; // Duration to wait before the next broadcast
unsigned long lastTempRead = 0;

```

```

float temperature = 0;

struct Data {
    int ID;
    int cyc;
    int thres;
    int intv;
};

Data received;

// ESP-NOW addresses

uint8_t panelbots[] = {0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF}; // broadcast to all esp
uint8_t pump[] = {0x08, 0xF9, 0xE0, 0x6C, 0x4B, 0x07}; //water pump esp address

void onDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
    Serial.print("ESP-NOW Send Status: ");
    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Success" : "Fail");
}

void setup() {
    Serial.begin(115200);
    Serial.println("ESP-NOW ESP Initialized");
    Serial2.begin(9600, SERIAL_8N1, 16, 17);

    pinMode(ktcCS, OUTPUT);
    digitalWrite(ktcCS, HIGH);

    delay(500); // Give the MAX6675 time to stabilize
}

```

```

// Initialize ESP-NOW

WiFi.mode(WIFI_STA);

if (esp_now_init() != ESP_OK) {
    Serial.println("Error initializing ESP-NOW");
    return;
}

esp_now_register_send_cb(onDataSent);

esp_now_peer_info_t peerInfo;
memcpy(peerInfo.peer_addr, panelbots, 6);
peerInfo.channel = 0;
peerInfo.encrypt = false;
if (esp_now_add_peer(&peerInfo) != ESP_OK) {
    Serial.println("Failed to add broadcast peer");
}

memcpy(peerInfo.peer_addr, pump, 6);
peerInfo.channel = 0;
peerInfo.encrypt = false;
if (esp_now_add_peer(&peerInfo) != ESP_OK) {
    Serial.println("Failed to add unicast peer");
}
}

void readTemperature() {
    if (millis() - lastTempRead > 500) { // Ensure reading occurs every 500ms
        digitalWrite(ktcCS, LOW);

```

```

delay(10);

temperature = ktc.readCelsius();

digitalWrite(ktcCS, HIGH);

lastTempRead = millis();

//Serial.println(temperature);

}

}

void sendBroadcastAndUnicast(int cycleValue) {

uint8_t broadcastMessage[1] = { (uint8_t)cycleValue };

esp_now_send(panelbots, broadcastMessage, sizeof(broadcastMessage));

Serial.println("Broadcast signal sent to panelbots.");



uint8_t unicastMessage[1] = { (uint8_t)cycleValue };

esp_now_send(pump, unicastMessage, sizeof(unicastMessage));

Serial.println("Unicast signal sent to water pump.");

}

void loop() {

readTemperature();

if (Serial2.available()) {

Serial2.readBytes((uint8_t*)&received, sizeof(received));



if (received.ID == 0) {

if (temperature < 0) {

Serial.println("Failed to read from MAX6675 sensor");

} else {

```

```

Serial2.println(temperature);
Serial.print("Temperature sent to Blynk ESP: ");
Serial.println(temperature);
}

} else if (received.ID == 1) {

    cycle = received.cyc;
    threshold = received.thres;
    Serial.print("Cycle received: ");
    Serial.println(cycle);
    Serial.print("Threshold updated to: ");
    Serial.println(threshold);

    // Send immediate broadcast and unicast message
    sendBroadcastAndUnicast(cycle);
} else if (received.ID == 2) {

    cycle = received.cyc;
    Serial.print("Cycle received: ");
    Serial.println(cycle);
} else if (received.ID == 3) {

    threshold = received.thres;
    Serial.print("Threshold updated to: ");
    Serial.println(threshold);
} else if (received.ID == 4) {

    threshintv = received.intv;
    Serial.print("Threshold-Interval Updated and Saved: ");
    Serial.println(threshintv);
}
}

```

```

// Periodically check the temperature against the threshold

readTemperature();

if (temperature >= 0) {

    if (temperature > threshold && threshold != 0) {

        unsigned long currentMillis = millis();

        if (currentMillis - lastBroadcastTime >= delayDuration) {

            Serial.println("Temperature exceeded threshold!");

            sendBroadcastAndUnicast(cycle);

            lastBroadcastTime = currentMillis;

        }

    }

}

// Set delay duration based on threshintv

switch (threshintv) {

    case 0: delayDuration = 30L * 60 * 1000; break; // 30 minutes

    case 1: delayDuration = 60L * 60 * 1000; break; // 1 hour

    case 2: delayDuration = 90L * 60 * 1000; break; // 1.5 hours

    case 3: delayDuration = 120L * 60 * 1000; break; // 2 hours

    case 4: delayDuration = 180L * 60 * 1000; break; // 3 hours

    default: delayDuration = 30L * 60 * 1000; break; // Default to 30 minutes

}

}

}

} else {

    Serial.println("Failed to read from MAX6675 sensor");

}

}

```

ESP WIFI

```

#define BLYNK_TEMPLATE_ID "TMPL3O23XiYhT"
#define BLYNK_TEMPLATE_NAME "TEST"
#define BLYNK_AUTH_TOKEN "I_R4crrvU78DLGUI9UgMnbyn2VqBBDCO"

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <time.h>
#include <WiFiUdp.h>
#include <NTPClient.h>

char ssid[] = "AlaniPhone";
char password[] = "asdf1234";

// Define NTP Client settings
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 19800, 60000); // GMT+5:30 (IST), update
every minute

unsigned long lastTriggeredTime = 0;
unsigned long debounceInterval = 60000;

int seg_sw = 0;
int menu = 0;
String slot1 = "";
String slot2 = "";
int cycle = 1;
int threshold = 0;

```

```

unsigned long lastExecutionTime = 0;

unsigned long menuInterval = 1000;

int led = 23 ;
bool wifiConnected = false;
unsigned long previousMillis = 0;
const long blinkInterval = 500; // Blink interval in ms
bool ledState = false;
unsigned long wifiConnectedMillis = 0;
bool ledOnFor3Sec = false;

struct Data{
    int ID;
    int cyc;
    int thres;
    int intv;
};

Data senddata;

void setup() {
    Serial.begin(115200);
    Serial2.begin(9600, SERIAL_8N1, 16, 17);

    pinMode(led, OUTPUT);

    Serial.println("Connecting to Wi-Fi...");
    WiFi.begin(ssid, password);
    timeClient.begin();
}

```

```

Blynk.begin(BLYNK_AUTH_TOKEN, ssid, password);

}

void handleWiFi() {
    unsigned long currentMillis = millis();

    if (WiFi.status() != WL_CONNECTED) {
        if (wifiConnected) {
            wifiConnected = false;
            Serial.println("\nWi-Fi Connection Lost! Blinking LED...");
        }

        if (currentMillis - previousMillis >= blinkInterval) {
            previousMillis = currentMillis;
            ledState = !ledState;
            digitalWrite(led, ledState);
        }
    } else {
        if (!wifiConnected) {
            wifiConnected = true;
            wifiConnectedMillis = millis();
            ledOnFor3Sec = true;
            Serial.println("\nConnected to Wi-Fi!");
            digitalWrite(led, HIGH);
        }

        if (ledOnFor3Sec && (currentMillis - wifiConnectedMillis >= 3000)) {
            digitalWrite(led, LOW);
        }
    }
}

```

```

ledOnFor3Sec = false;

}

}

}

bool isTimeMatch(String targetTime) {
    timeClient.update();
    int currentHour = timeClient.getHours();
    int currentMinute = timeClient.getMinutes();
    int timevalue = (currentHour * 60 * 60 + currentMinute * 60);
    String currentTime = String(timevalue);
    Serial.println("Comparing Time: " + currentTime + " with " + targetTime);
    return currentTime == targetTime;
}

BLYNK_WRITE(V0) {
    int forcestart = param.toInt();
    if (forcestart == 1) {
        sendBroadcast();
    }
}

BLYNK_WRITE(V1) {
    seg_sw = param.toInt();
    Serial.println("seg_sw Updated: " + String(seg_sw));
}

BLYNK_WRITE(V2) {
}

```

```

menu = param.asInt();
Serial.println("menu Updated: " + String(menu));
}

BLYNK_WRITE(V3) {
slot1 = param.asString();
Serial.println("slot1 Updated: " + slot1);
}

BLYNK_WRITE(V4) {
slot2 = param.asString();
Serial.println("slot2 Updated: " + slot2);
}

BLYNK_WRITE(V7) {
senddata.cyc = param.asInt() + 1;
Serial.print("Cycle count Updated: ");Serial.println(senddata.cyc);
senddata.ID = 2;
Serial2.write((uint8_t*)&senddata,sizeof(senddata)); delay(100);
senddata.ID = 0;
}

BLYNK_WRITE(V8) {
senddata.thres = param.asInt();
Serial.print("Threshold Updated: ");Serial.println(senddata.thres);
senddata.ID = 3;
Serial2.write((uint8_t*)&senddata,sizeof(senddata)); delay(100);
senddata.ID = 0;
}

```

```
}
```

```
BLYNK_WRITE(V9) {
    senddata.intv = param.asInt();
    Serial.print("Threshold-Interval Updated: ");
    Serial.println(senddata.intv);
    senddata.ID = 4;
    Serial2.write((uint8_t*)&senddata,sizeof(senddata));
    delay(100);
    senddata.ID = 0;
}
```

```
BLYNK_WRITE(V5) {
    Serial.println("Blynk requests temperature");
    senddata.ID = 0;
    Serial2.write((uint8_t*)&senddata,sizeof(senddata));
    delay(100);

    if (Serial2.available()) {
        String receivedTemperature = Serial2.readStringUntil('\n');
        Serial.println("Temperature received: " + receivedTemperature);
        receivedTemperature.trim();
        float temperature = receivedTemperature.toFloat();
        Blynk.virtualWrite(V6, receivedTemperature);
    }
}
```

```
void sendBroadcast() {
    senddata.ID = 1;
    Serial2.write((uint8_t*)&senddata,sizeof(senddata));
    delay(100);
    senddata.ID = 0;
```

```

        Serial.print("Sent      via      UART      cycle      &      threshold      =
");Serial.print(senddata.cyc);Serial.print("& ");Serial.println(senddata.thres);
}

void checkLogic() {
    if (seg_sw == 0) {
        if ((millis() - lastTriggeredTime) >= debounceInterval) {
            if (isTimeMatch(slot1) || isTimeMatch(slot2)) {
                Serial.println("Specified Time Reached, sending signal");
                sendBroadcast();
                lastTriggeredTime = millis();
            }
        }
    } else if (seg_sw == 1) {
        if (millis() - lastExecutionTime >= menuInterval) {
            lastExecutionTime = millis();

            switch (menu) {
                case 0: menuInterval = 15 * 60 * 1000; Serial.println("15m interval completed, sending
signal to bots"); break;
                case 1: menuInterval = 30 * 60 * 1000; Serial.println("30m interval completed, sending
signal to bots"); break;
                case 2: menuInterval = 1 * 60 * 60 * 1000; Serial.println("1hr interval completed,
sending signal to bots"); break;
                case 3: menuInterval = 90 * 60 * 1000; Serial.println("1.5hr interval completed, sending
signal to bots"); break;
                case 4: menuInterval = 2 * 60 * 60 * 1000; Serial.println("2hr interval completed,
sending signal to bots"); break;
                case 5: menuInterval = 3 * 60 * 60 * 1000; Serial.println("3hr interval completed,
sending signal to bots"); break;
            }
        }
    }
}

```

```

default: return;

}

sendBroadcast();

}

}

}

}

void loop() {

handleWiFi();

if (wifiConnected) {

Blynk.run();

timeClient.update();

checkLogic();

}

delay(10); // Small delay to prevent excessive CPU usage

}

```

3) PROGRAM CODE FOR WATER PUMP

```

#include <ESP8266WiFi.h>

#include <espnow.h>

#define Motor 5 // Adjust according to your board

#define LED 4

bool On = false;

unsigned long TurnoffTime = 0;

```

```

void onDataRecv(uint8_t *mac, uint8_t *data, uint8_t len) {

    if (len > 0) {
        int d = data[0];
        int duration = 0; // Variable to hold motor run time

        // Set motor duration based on the value of d
        if (d == 0 || d == 1) {
            duration = 20;
        } else if (d == 2) {
            duration = 40;
        } else if (d == 3) {
            duration = 30;
        } else if (d == 4) {
            duration = 60;
        } else {
            duration = 60;
        }

        // Start motor and LED
        digitalWrite(Motor, LOW);
        digitalWrite(LED, HIGH);
        On = true;
        TurnoffTime = millis() + (duration * 1000); // Set turn-off time
        Serial.printf("Turning pump on for %d seconds\n", duration);
    }
    }
}

```

```
void setup() {
    Serial.begin(115200);
    pinMode(Motor, OUTPUT);
    pinMode(LED, OUTPUT);
    WiFi.mode(WIFI_STA);
    digitalWrite(Motor, HIGH);

    if (esp_now_init() != 0) {
        Serial.println("Error initializing ESP-NOW");
        return;
    }

    esp_now_set_self_role(ESP_NOW_ROLE_SLAVE);
    esp_now_register_recv_cb(onDataRecv);
}

void loop() {
    if (On && millis() > TurnoffTime) {
        digitalWrite(Motor, HIGH);
        digitalWrite(LED, LOW); // Turn LED OFF
        On = false;
    }
}
```

Alan Gabriel

MOTION-SENSOR LVDC.pdf

 Mar Baselios College of Engineering and Technology

Document Details

Submission ID

trn:oid:::3618:92355260

98 Pages

Submission Date

Apr 22, 2025, 4:09 PM GMT+5:30

14,674 Words

Download Date

Apr 22, 2025, 4:11 PM GMT+5:30

86,155 Characters

File Name

MOTION-SENSOR LVDC.pdf

File Size

7.9 MB

15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

-  **131** Not Cited or Quoted 14%
Matches with neither in-text citation nor quotation marks
-  **1** Missing Quotations 0%
Matches that are still very similar to source material
-  **5** Missing Citation 0%
Matches that have quotation marks, but no in-text citation
-  **0** Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

- 13%  Internet sources
- 10%  Publications
- 0%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

-  131 Not Cited or Quoted 14%
Matches with neither in-text citation nor quotation marks
-  1 Missing Quotations 0%
Matches that are still very similar to source material
-  5 Missing Citation 0%
Matches that have quotation marks, but no in-text citation
-  0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

- 13%  Internet sources
- 10%  Publications
- 0%  Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

Rank	Type	Source	Percentage
1	Internet	www.jetir.org	3%
2	Internet	elonics.org	1%
3	Internet	forum.arduino.cc	<1%
4	Internet	www.coursehero.com	<1%
5	Internet	vicinity2020.eu	<1%
6	Internet	www.maximintegrated.com	<1%
7	Internet	github.com	<1%
8	Internet	pluginhighway.ca	<1%
9	Internet	fabacademy.org	<1%
10	Publication	Hakala, Tomi, Tommi Lahdeaho, and Pertti Jarventausta. "Low-Voltage DC Distrib...	<1%

11 Publication

Nasib Khadka, Aayush Bista, Binamra Adhikari, Ashish Shrestha, Diwakar Bista, B... <1%

12 Internet

mbcet.ac.in <1%

13 Internet

create.arduino.cc <1%

14 Internet

www.bitsadmission.com <1%

15 Internet

eleceeasy.com <1%

16 Internet

ijjsrem.com <1%

17 Internet

khazna.ku.ac.ae <1%

18 Internet

www.tamimi.com <1%

19 Internet

researchr.org <1%

20 Publication

Neil Cameron. "Electronics Projects with the ESP8266 and ESP32", Springer Scienc... <1%

21 Internet

copperhilltech.com <1%

22 Internet

dspace.bracu.ac.bd <1%

23 Internet

www.theseus.fi <1%

24 Internet

ijarsct.co.in <1%

25 Internet

www.epj-pv.org <1%

26 Internet

archive.org <1%

27 Internet

www.researchgate.net <1%

28 Publication

Anudeep Juluru, Shriram K. Vasudevan, T. S. Murugesh. "Let's Get IoT! - 30 IoT Pr... <1%

29 Publication

Wajahat Khalid, Mohsin Jamil, Ashraf Ali Khan, Qasim Awais. "Open-Source Intern... <1%

30 Internet

archives.njit.edu <1%

31 Internet

content.instructables.com <1%

32 Internet

docs.blynk.io <1%

33 Internet

sparklyn.com.ng <1%

34 Internet

pt.scribd.com <1%

35 Internet

www.semanticscholar.org <1%

36 Publication

Gulshan Sharma, Pitshou N. Bokoro, Sudeep Tanwar. "Energy 4.0 - Trends, Challe... <1%

37 Internet

teddit.sdf.org <1%

38 Internet

pdfcookie.com <1%

39 Internet

solarquarter.com <1%

40 Publication

"IoT Based Solar Power Management Using Smart Switching", International Journ... <1%

41 Publication

Syed A.M. Said, Ghassan Hassan, Husam M. Walwil, N. Al-Aqeeli. "The effect of en... <1%

42 Internet

arduino.stackexchange.com <1%

43 Internet

documents.mx <1%

44 Internet

niuniv.com <1%

45 Internet

www.autoscripts.net <1%

46 Internet

www.instructables.com <1%

47 Publication

C. Tejashwini, Gajjala Somesh Kumar, R. Shekhar. "Chapter 23 Smart Pregnancy ... <1%

48 Publication

Hun-Chul Seo, Gi-Hyeon Gwon, Keon-Woo Park. "Development of New Protection ... <1%

49 Publication

Isyatur Raziah, Andri Novandri, Yuwaldi Away. "Real-Time Monitoring of Photovol... <1%

50 Publication

Li, Lulu, Jing Yong, Liqiang Zeng, and Xiaoyu Wang. "Investigation on the system ... <1%

51 Publication

Masashi Kawaguchi, Sachiyoshi Ichikawa, Masaaki Okuno, Takashi Jimbo, Naohir... <1%

52 Internet

digitalcollection.utm.edu.my <1%

53	Internet	
idoc.tips		<1%
54	Internet	
isroset.org		<1%
55	Internet	
repository.itny.ac.id		<1%
56	Internet	
turbocharged.in		<1%
57	Internet	
www.klikglodok.com		<1%
58	Publication	
, khalil rehman. "A Case Study of an Intelligent Traffic Control System in Smart Cit...		<1%
59	Publication	
Anjana Singh, Ravi Shankar, Amitesh Kumar. "A Comprehensive Review of Load F...		<1%
60	Internet	
circuitdigest.com		<1%
61	Internet	
docplayer.net		<1%
62	Internet	
dokumen.pub		<1%
63	Internet	
robu.in		<1%
64	Internet	
www.ersaelectronics.com		<1%
65	Internet	
www.hindawi.com		<1%
66	Internet	
www.slideshare.net		<1%

67

Publication

"Channel Characterization", Power Line Communications, 2016. <1%

68

Publication

Hao Lu, Wenjun Zhao. "Effects of particle sizes and tilt angles on dust deposition ... <1%

69

Publication

Jerez, Raiphy. "Novel Topology for Capacitively Isolated Switched Capacitor Conve... <1%

70

Publication

M. Gandomzadeh, A.A. Yaghoubi, A. Hoorsun, A. Parsay, A. Gholami, M. Zandi, R. ... <1%

71

Publication

Muneeb Ur Rehman, Muhammad. "Modular, Scalable Battery Systems with Integ... <1%

72

Publication

Ryan Stock, Benjamin K. Sovacool. "Left in the dark: Colonial racial capitalism and... <1%

73

Publication

Ton Duc Thang University <1%

74

Publication

Κορωνιός, Ευάγγελος. "Εξυπνη Υγεία : Δημιουργώντας Μια Συσκευή Μέτρησης", ... <1%

75

Internet

en.wikipedia.org <1%